

Université de Montréal

**Évaluation d'un modèle *a priori* basé sur un seuillage de la TCD en  
super-résolution et comparaison avec d'autres modèles a priori**

par  
Philippe St-Onge

Département d'informatique et de recherche opérationnelle  
Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures  
en vue de l'obtention du grade de Maître ès sciences (M.Sc.)  
en informatique

Septembre, 2008

© Philippe St-Onge, 2008.

Université de Montréal  
Faculté des études supérieures

Ce mémoire intitulé:

**Évaluation d'un modèle *a priori* basé sur un seuillage de la TCD en  
super-résolution et comparaison avec d'autres modèles a priori**

présenté par:

Philippe St-Onge

a été évalué par un jury composé des personnes suivantes:

Pierre Poulin  
président-rapporteur

Max Mignotte  
directeur de recherche

Philippe Langlais  
membre du jury

**Mémoire accepté le**

## RÉSUMÉ

Le problème de la super-résolution entre dans le cadre des problèmes inverses mal posés en traitement d'images. C'est en fait un problème de restauration ou de reconstruction qui consiste en l'obtention d'images de résolutions plus grandes que celles fournies par le capteur utilisé. L'objectif visé est de sur-échantillonner l'image tout en réduisant les erreurs d'aliassage, le bruit et le flou. Ce projet de recherche a pour but de proposer et réaliser une procédure bayésienne pour résoudre ce problème. Dans celle-ci, nous modélisons les dégradations subies par l'image originale théorique par une distribution appelée vraisemblance, combinée avec un modèle *a priori* (ou terme de régularisation) qui résume les connaissances des propriétés de l'image haute résolution que l'on cherchera ensuite à inférer. À cette fin, nous nous proposons de tester différents modèles *a priori* pour ce problème de reconstruction ; en particulier, le plus simple et déjà utilisé dans la littérature, le terme de régularisation de Tikhonov, et le modèle dit de GGMRF, mis au point afin de répondre à l'exigence de préservation concurrente des régions uniformes et des contours des objets contenus dans l'image, seront comparés avec le modèle *a priori* médian de Hsiao *et al.* [14] (utilisé en tomographie), un autre basé sur une segmentation Markovienne [16], et un troisième basé sur un seuillage de la transformée en cosinus discrète [51].

**Mots clés:** traitement d'images, reconstruction, restauration, super-résolution, modèle bayésien, modèle *a priori*.

## ABSTRACT

Super-resolution is an ill-posed inverse problem that consists in the reconstruction of an image at a resolution that is higher than the available data otherwise permits. This generally requires the availability of multiple low-res images taken from slightly shifted points of view, an imaging model describing the degradation the ideal hi-res image undergoes during low-res image acquisition, and which includes aliasing, noise and blur. This paper aims to describe and implement a bayesian procedure designed to solve this problem. To this end, the degradation process is modelled by a distribution called likelihood, combined to an *a priori* model (or regularization term) which embodies our expectations of what properties the original, undegraded hi-res image should possess. In this context, we propose to implement and test a few *a priori* models to examine their properties. The first one, the simplest, and probably the best-known in the literature, is the Tikhonov regularisation term. We also examine another, more advanced model, the Generalized Gaussian Markov Random Fields model (or GGMRF) proposed by Bouman and Sauer in tomography. We then compare these two staples of image processing literature, first to a median prior proposed by Hsiao *et al.* [14]; then to another based on a segmentation [16]; and finally to a model based on thresholding of the DCT [51].

**Keywords:** Image processing, reconstruction, restoration, Bayesian model, *a priori* model.

## TABLE DES MATIÈRES

<b>RÉSUMÉ</b> . . . . .	<b>iii</b>
<b>ABSTRACT</b> . . . . .	<b>iv</b>
<b>TABLE DES MATIÈRES</b> . . . . .	<b>v</b>
<b>LISTE DES TABLEAUX</b> . . . . .	<b>vii</b>
<b>LISTE DES FIGURES</b> . . . . .	<b>viii</b>
<b>LISTE DES SIGLES</b> . . . . .	<b>x</b>
<b>NOTATION</b> . . . . .	<b>xi</b>
<b>CHAPITRE 1 : INTRODUCTION</b> . . . . .	<b>1</b>
<b>CHAPITRE 2 : ACQUISITION ET FORMATION D'IMAGES DIGITALES</b> . . . . .	<b>6</b>
2.1 La lumière et le spectre électromagnétique . . . . .	6
2.2 La caméra CCD . . . . .	7
2.2.1 Le module CCD . . . . .	10
2.2.2 L'interface digitale et la représentation de l'image digitale . . . . .	17
<b>CHAPITRE 3 : ACQUISITION ET TRAITEMENT DES DONNÉES</b> <b>20</b>	
3.1 Le modèle d'observation . . . . .	20
3.1.1 Le recalage . . . . .	22
3.2 La fonction de dispersion de points . . . . .	26
3.3 Le bruit . . . . .	29
3.3.1 Alternatives et implantation . . . . .	30
3.4 Reconstruction MV . . . . .	33
3.5 Reconstruction MAP . . . . .	36

3.6	Le choix du terme <i>a priori</i> . . . . .	37
3.7	Autres approches . . . . .	37
3.8	Modèles <i>a priori</i> étudiés . . . . .	40
3.8.1	Régularisation Tikhonov . . . . .	40
3.8.2	Modèle <i>a priori</i> GGMRF . . . . .	40
3.8.3	Modèle médian . . . . .	44
3.8.4	Modèle basé sur une segmentation . . . . .	47
3.8.5	Modèle de débruitage par seuillage de coefficients de DCT . . . . .	56
3.9	Stratégie de minimisation et implantation . . . . .	67
<b>CHAPITRE 4 : PRÉSENTATION DES RÉSULTATS ET DISCUS-</b>		
<b>SION . . . . .</b>		<b>71</b>
4.1	Méthodologie de test . . . . .	71
4.1.1	Génération synthétique de séquences d'images BR . . . . .	71
4.1.2	La variance du bruit de l'image . . . . .	74
4.1.3	Le nombre d'images BR pour la super-résolution . . . . .	74
4.1.4	Les paramètres . . . . .	76
4.1.5	La mesure de performance . . . . .	78
4.2	Présentation des résultats . . . . .	79
4.3	Discussion . . . . .	80
<b>CHAPITRE 5 : CONCLUSION . . . . .</b>		<b>98</b>
<b>BIBLIOGRAPHIE . . . . .</b>		<b>100</b>

## LISTE DES TABLEAUX

3.1	Algorithme d'appariement de blocs . . . . .	24
3.2	Algorithme ECI . . . . .	52
3.3	Algorithme des k-moyennes . . . . .	53
3.4	Algorithme ICM . . . . .	55
3.5	Algorithme de super-résolution . . . . .	68
3.6	Algorithme de super-résolution par seuillage de la DCT . . . . .	70
4.1	Séquences d'images pour tests . . . . .	76
4.2	Modèles et paramètres menant aux meilleurs résultats pour la séquence <b>baboon</b> tous modèles confondus . . . . .	80
4.3	Modèles et paramètres menant aux meilleurs résultats pour la séquence <b>barbara</b> tous modèles confondus . . . . .	81
4.4	Modèles et paramètres menant aux meilleurs résultats pour la séquence <b>f16</b> tous modèles confondus . . . . .	81
4.5	Modèles et paramètres menant aux meilleurs résultats pour la séquence <b>lena</b> tous modèles confondus . . . . .	90
4.6	Paramètres menant aux meilleurs résultats pour la séquence <b>baboon</b> pour chaque modèle . . . . .	91
4.7	Paramètres menant aux meilleurs résultats pour la séquence <b>barbara</b> pour chaque modèle . . . . .	92
4.8	Paramètres menant aux meilleurs résultats pour la séquence <b>f16</b> pour chaque modèle . . . . .	93
4.9	Paramètres menant aux meilleurs résultats pour la séquence <b>lena</b> pour chaque modèle . . . . .	94
4.10	Paramètres produisant les meilleurs résultats en moyenne pour chaque modèle, toutes variances et nombres d'images confondues . . . . .	95

## LISTE DES FIGURES

1.1	Exemple de super-résolution . . . . .	3
1.2	Intuition derrière la super-résolution . . . . .	4
2.1	Le spectre électromagnétique . . . . .	7
2.2	Système optique de lentille mince . . . . .	8
2.3	La diffraction . . . . .	9
2.4	Exemples d'aberration . . . . .	10
2.5	Systèmes d'acquisition d'images . . . . .	11
2.6	Structure d'un élément CCD . . . . .	12
2.7	Transfert CCD pleine trame . . . . .	12
2.8	Acquisition d'images couleurs avec un CCD . . . . .	14
2.9	Échantillonnage et quantification . . . . .	15
2.10	L'effet de moiré . . . . .	16
2.11	Bruit gaussien . . . . .	17
2.12	Blooming . . . . .	18
3.1	Modèle d'observation de l'image . . . . .	22
3.2	Pyramide multi-résolution d'une image . . . . .	25
3.3	Champ de vecteur . . . . .	27
3.4	Illustration de l'influence de la PSF . . . . .	28
3.5	La relation entre les pixels BR et HR . . . . .	32
3.6	Interpolation non-uniforme . . . . .	38
3.7	Cliques d'ordre 1 et 2 . . . . .	42
3.8	Relation entre $f_j$ et $m_j$ . . . . .	46
3.9	Approximation de la valeur absolue par la $\log \cosh(\cdot)$ . . . . .	47
3.10	Matrices de base de la DCT . . . . .	59
3.11	Carrelage pour la DCT . . . . .	60
3.12	Intuition derrière la méthode par seuillage de la DCT . . . . .	63

4.1	Les quatre images tests . . . . .	73
4.2	16 images BR . . . . .	75
4.3	Graphique des performance des différents modèles, paramètres fixes	82
4.4	Exemples de résultats : <b>baboon</b> . . . . .	83
4.5	Exemples de résultats : <b>barbara</b> . . . . .	84
4.6	Exemples de résultats : <b>lena</b> . . . . .	85
4.7	Région d'intérêt dans l'image <b>lena</b> . . . . .	86
4.8	Région d'intérêt dans l'image <b>f16</b> . . . . .	87
4.9	Région d'intérêt (pantalons) dans l'image <b>barbara</b> . . . . .	88
4.10	Région d'intérêt (détail de la nappe) dans l'image <b>barbara</b> . . . . .	89

## LISTE DES SIGLES

ARSB	Amélioration du rapport signal sur bruit
BR	Basse-résolution
CCD	Charge-Coupled Device
DCT	Transformée en cosinus discrète
ECI	Estimation conditionnelle itérative
GGMRF	Modèle “Generalized Gaussian Markov Random Fields”
HR	Haute-résolution
ICM	Iterated Conditonal Modes
MAP	Maximum <i>a posteriori</i>
MED	Modèle <i>a priori</i> médian
MV	Maximum de la vraisemblance
PSF	Point Spread Function
RSBP	Rapport signal sur bruit de pointe
SEG	Modèle <i>a priori</i> basé sur une segmentation
TM	Modèle <i>a priori</i> Tikhonov-Miller

## NOTATION

$\mathbf{x}$	Image HR idéale
$\hat{\mathbf{x}}$	Image HR reconstruite
$\mathbf{y}_k$	Image BR
$\mathbf{y}_r$	Image BR de référence pour le recalage
$L_1$	Facteur de sous-échantillonnage horizontal
$L_2$	Facteur de sous-échantillonnage vertical
$L$	Facteur de sous-échantillonnage lorsque $L_1 = L_2$
$N_1$	Dimension horizontale d'une image BR
$N_2$	Dimension verticale d'une image BR
$N$	Nombre de pixels dans l'image HR $\mathbf{x}$
$M$	Nombre de pixels dans une image BR $\mathbf{y}_k$
$k$	Nombre d'images BR $\mathbf{y}_k$
$\mathbf{D}$	Matrice de décimation
$\mathbf{F}_k$	Matrice de flou correspondant à $\mathbf{y}_k$
$\mathbf{M}_k$	Matrice de déformation correspondant à $\mathbf{y}_k$
$\mathbf{n}_k$	Vecteur du bruit correspondant à $\mathbf{y}_k$
$\mathbf{W}_k$	Matrice combinant les effets de $\mathbf{F}_k$ et $\mathbf{M}_k$
$\mathbf{y}$	Vecteur des $k$ images BR ordonné lexicographiquement
$\mathbf{W}$	Matrice formée par la combinaison des $\mathbf{W}_k$
$\mathbf{n}$	Vecteur du bruit des $k$ images BR
$\hat{\mathbf{x}}_{ML}$	Estimation HR obtenu par reconstruction MV
$\hat{\mathbf{x}}_{MAP}$	Estimation HR obtenu par reconstruction MAP
$\Omega(\mathbf{x})$	Terme <i>a priori</i> , fonction de coût
$\Omega_{\text{méd}}(\mathbf{x})$	Terme <i>a priori</i> médian

# CHAPITRE 1

## INTRODUCTION

Les images digitales forment une source de données importante pour un large éventail de domaines technologiques. Qu'il s'agisse d'imagerie médicale, de télé-surveillance, d'observation par satellite, de radioastronomie, ou de divertissement multimédia, les images digitales occupent une position centrale au sein d'une foule d'applications de pointe, ce qui confère un intérêt particulier aux techniques d'amélioration de leur qualité et leur expressivité. Un critère important de cette expressivité est la résolution spatiale, une quantité que nous chercherons donc à maximiser pour la plupart des applications.

La résolution spatiale est communément définie comme la taille du plus petit détail perceptible dans une image [1]. Pour illustrer, imaginons deux images identiques en tous points hormis leur résolution. Nous appellerons l'image présentant la résolution la plus haute image haute-résolution, et l'autre image basse-résolution<sup>1</sup>. Les affirmations suivantes sont alors équivalentes : 1) l'image haute-résolution présentera des dimensions en pixels<sup>2</sup> supérieures à celles de l'image basse-résolution ; 2) l'image haute-résolution aura une plus haute densité de pixels par unité de surface du plan image ; 3) un pixel de l'image basse-résolution couvrira une plus grande surface du plan image que celle occupée par un pixel de l'image haute-résolution. Puisque le pixel est l'élément de base d'une image digitale, tout détail de la scène dont la projection sur le plan image présente une surface inférieure ou similaire à celle d'un pixel sera estompé, et il en découle logiquement que l'image haute-résolution pourra préserver des détails imperceptibles dans l'image basse-

---

<sup>1</sup>Aucune hypothèse n'est émise quant aux dimensions des images concernées : il suffit qu'une image présente une résolution supérieure à une autre pour que ces qualificatifs s'appliquent en toute logique.

<sup>2</sup>Les dimensions en pixels d'une image sont généralement déclinées sous la forme largeur  $\times$  hauteur ; par exemple les dimensions  $1280 \times 1024$  décrivent une image présentant 1280 colonnes et 1024 rangées de pixels.

résolution.

Bien qu'il soit possible d'augmenter la résolution des images digitales acquises en développant des senseurs plus performants, cette démarche implique des coûts importants et se butte de toute façon à des limites physiques [2, 3] qui devront être contournées autrement une fois celles-ci atteintes. Une approche prometteuse pour atteindre cet objectif est la méthode dite de super-résolution [2, 12, 13]. Il s'agit d'une approche apparentée à la restauration d'images qui consiste à calculer une estimation haute-résolution à partir d'une série d'images basse-résolution (typiquement une séquence vidéo). Cette méthode présente plusieurs avantages attrayants. Tout d'abord elle permet d'obtenir des images haute-résolution sans exiger l'usage d'équipement haute précision, ce qui présente un intérêt immédiat au point de vue économique et permet d'envisager son utilisation pour des applications qui ne peuvent justifier le prix d'achat de tel matériel. Ensuite, la technique ne requiert dans la plupart des cas que la disponibilité de plusieurs images d'une même scène, prises à des points de vue légèrement différents, pour générer une image digitale haute-résolution de bonne qualité, prérequis généralement aisément rempli pour la plupart des applications où l'acquisition se fait sous la forme de séquence vidéo.

Le processus qui conduit à l'estimation d'images haute-résolution par super-résolution est fondé sur l'intuition suivante : deux images basse-résolution d'une même scène, présentant un déplacement de l'une vers l'autre comportant un élément fractionnel, contiennent chacune de l'information différente sur la scène originale, qu'il est possible d'exploiter afin de construire une estimation haute-résolution. Pour illustrer cette idée à l'aide d'un exemple, imaginons deux images  $y_1(i, j)$  et  $y_2(i, j)$  de dimensions arbitraires, prises avec une caméra dont le point de vue a été légèrement modifié entre chaque acquisition, dans ce cas-ci par une simple translation<sup>3</sup>. Si le déplacement en pixel d'une image à l'autre peut s'exprimer en nombre entier, par exemple, une translation vers la gauche de l'ordre d'un pixel de  $y_2$  par rapport à  $y_1$ , les deux images contiennent la même information :

---

<sup>3</sup>Nous dénotons ici les images sous une notation fonctionnelle :  $y_1$  identifie l'image,  $i$  identifie la colonne du pixel et  $j$  sa rangée.

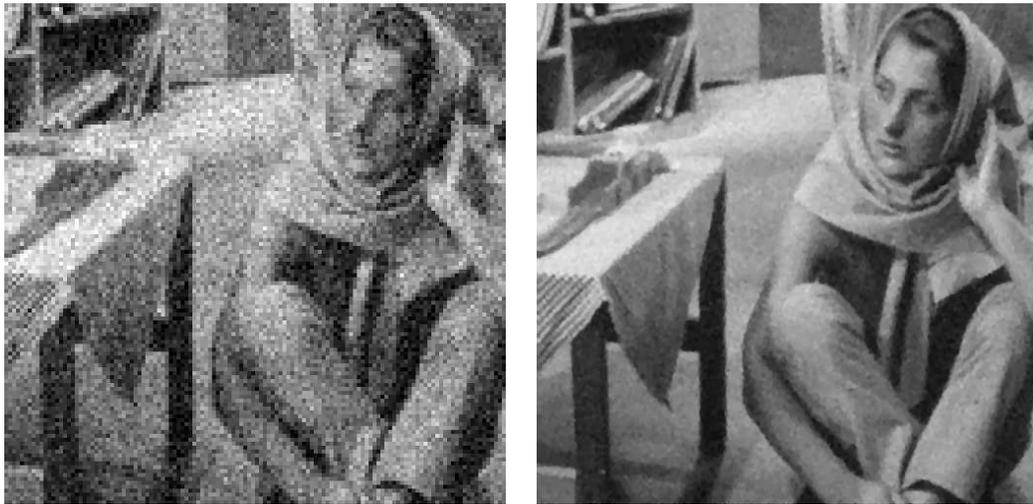


FIG. 1.1 – Comparaison entre la première d’une série de 16 images basse-résolution bruitées et une estimation haute-résolution générée à l’aide du modèle *a priori* GGMRF.

$y_2(i, j) = y_1(i+1, j)$  (nous faisons ici abstraction du problème des bords de l’image). Cependant, si ce déplacement contient un élément fractionnel, une telle équivalence entre les deux images ne peut être établie et chaque image contient donc de l’information différente<sup>4</sup>. Intuitivement, il paraît possible d’utiliser cette information (c’est-à-dire la valeur des pixels basse-résolution et une correspondance entre les pixels des deux images telle que calculée par une étape d’estimation du mouvement) pour recouvrer la valeur des pixels d’une image haute-résolution théorique. Cette image haute-résolution théorique constitue l’image à être générée par super-résolution : elle est de résolution arbitraire, choisie en fonction des données disponibles. Chaque pixel de chacune des images basse-résolution correspondra à un groupe de pixels haute-résolution de l’image à reconstruire. Si nous pouvons estimer le mouvement entre les images basse-résolution, nous pouvons établir ces correspondances sous la forme d’une somme pondérée de pixels haute-résolution pour chaque pixel basse-résolution. En inversant les équations ainsi générées, par

---

<sup>4</sup>Nous assumons ici que la scène comporte des éléments haute fréquence qui ont été atténués par le processus d’acquisition, hypothèse presque universellement valide en pratique.

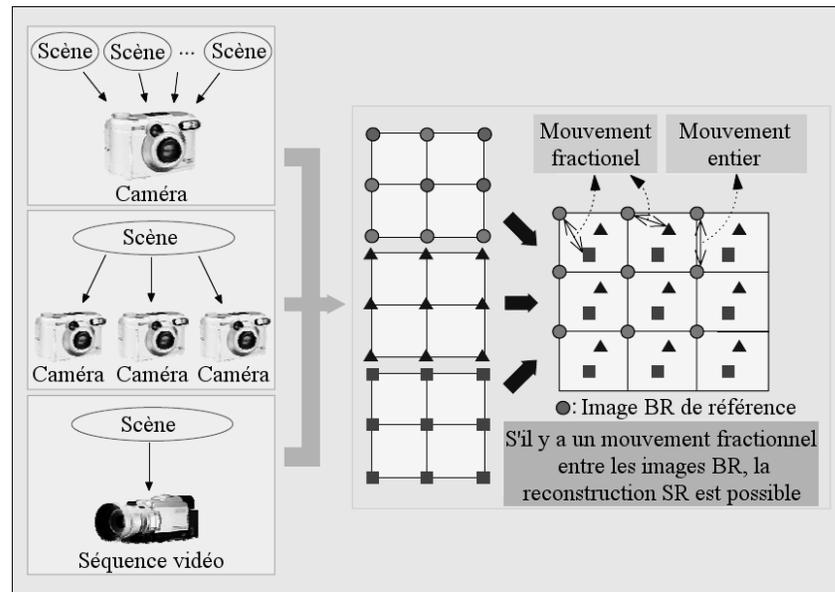


FIG. 1.2 – Illustration graphique de l'intuition derrière la super-résolution (source : [2]).

exemple à l'aide de méthodes linéaires, il est possible de recouvrer une estimation de la valeur des pixels haute-résolution et ainsi reconstruire une image haute-résolution agrégeant l'information des images basse-résolution.

Dans le cadre de ce mémoire, nous nous proposons de présenter une méthode pour arriver à la reconstruction d'une estimation haute-résolution à partir d'une série d'images basse-résolution. Dans ce contexte, nous avons implanté l'algorithme de reconstruction régularisée décrit au chapitre 3. Cette implantation a exigé la mise au point d'un processus itératif visant le raffinement progressif d'une estimation initiale à l'aide de méthodes linéaires. La contribution du présent mémoire réside en l'application au problème de la super-résolution des modèles *a priori* présentés plus loin dans le texte, ainsi que l'évaluation comparatives de leurs performances.

Le présent travail se conforme ainsi à une structure permettant de bien exposer les éléments du problème. Dans un premier temps, le chapitre 2 présente le processus d'acquisition et de formation de l'image, par lequel l'image idéale

haute-résolution théorique est dégradée pour nous parvenir sous la forme d'image basse-résolution. Le chapitre 3 se penche sur la théorie de la super-résolution, en commençant par un énoncé détaillé du problème. Nous enchaînons avec une présentation de plusieurs modèles *a priori*, dont les trois qui sont l'objet du présent travail, c'est-à-dire le modèle médian de Hsiao *et al.* [14] et un modèle basé sur une segmentation [16] et un modèle basé sur un seuillage de la DCT [51]. Nous poursuivons avec une description de l'algorithme de super-résolution que nous utilisons dans notre implantation et les adaptations requises, le cas échéant, pour chaque modèle. Finalement, le chapitre 4 s'attarde à la présentation et l'analyse de résultats, et nous présentons nos conclusions au chapitre 5.

## CHAPITRE 2

### ACQUISITION ET FORMATION D'IMAGES DIGITALES

Une image digitale typique est la représentation bidimensionnelle d'une scène telle que révélée par une source d'illumination. Cette dernière peut revêtir plusieurs formes, comme par exemple la lumière visible, les rayons X, les ondes radar, les ultrasons, etc. [1]. Dans ce chapitre, nous traitons de l'acquisition d'images digitales captées dans le spectre visible. Nous fournissons tout d'abord une brève description du spectre électromagnétique. Nous enchaînons avec une description de la structure d'un type commun de caméra digitale, la caméra CCD, et nous exposons en parallèle les effets de l'acquisition et de la digitalisation sur la qualité de l'image digitale obtenue par observation d'une scène donnée. Cette discussion du processus de digitalisation nous permettra de formuler un modèle d'imagerie au chapitre suivant, modèle que nous utiliserons pour décrire et résoudre le problème de la super-résolution.

#### 2.1 La lumière et le spectre électromagnétique

Le rayonnement électromagnétique désigne la propagation dans l'espace d'ondes présentant des éléments électriques et magnétiques, nommées ondes électromagnétiques [1, 4]. On peut aborder ces ondes sous deux aspects : l'aspect d'ondes sinusoïdales, ou celui de particules d'énergie sans masse appelées photons ; il s'agit de la dualité onde-particule des ondes électromagnétiques. Les ondes électromagnétiques sont dotées d'une longueur, d'une fréquence et d'une énergie, avec la longueur d'onde inversement proportionnelle à la fréquence et à l'énergie. Ces propriétés permettent de classer les ondes électromagnétiques sur une échelle appelée spectre électromagnétique, que l'on peut voir à la figure 2.1. Le spectre est divisé en régions désignant la "famille" d'ondes correspondant à un intervalle donné dans le spectre. La lumière visible occupe une région étroite allant de  $0,43\mu\text{m}$  (violet) à  $0,79\mu\text{m}$

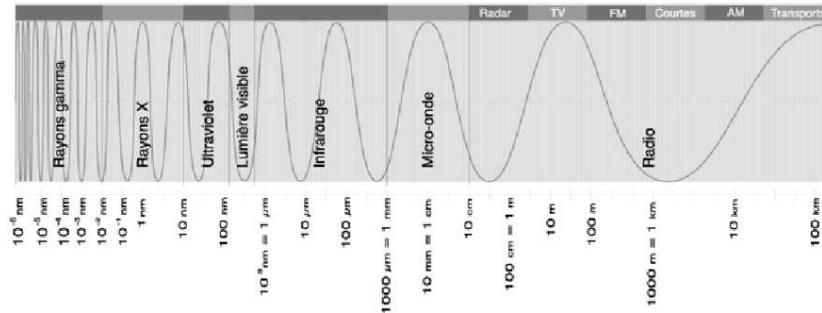


FIG. 2.1 – Le spectre électromagnétique (source : Musée virtuel Canada).

(rouge), et la lumière “blanche” est obtenue par combinaison de photons de toutes les longueurs d’onde dans le spectre visible.

## 2.2 La caméra CCD

Le détecteur à couplage de charge, communément appelé CCD (de l’anglais, *Charge-Coupled Device*), est le résultat des travaux de Willard Boyle et George E. Smith aux laboratoires Bell à la fin des années 1960 [5]. Conçu à l’origine comme un nouveau type de mémoire informatique, le CCD s’est rapidement fait remarquer pour son potentiel en imagerie en raison de sa grande sensibilité à la lumière, et il constitue l’une des technologies les plus répandues dans la construction de caméras digitales modernes [1].

Une caméra digitale typique est composée des éléments principaux suivants [6] : un appareil optique, un module CCD et une interface digitale.

L’appareil optique est composé d’une ou plusieurs lentilles et permet de concentrer les rayons lumineux en provenance d’un point de la scène vers un point (idéalement) unique du plan focal (figure 2.2). L’acquisition d’images nettes requiert que ce point se retrouve sur ou près de la grille de photosenseurs du module CCD : le système de lentilles est donc conçu pour que tous les points dans une région de l’espace, appelée profondeur de champ, respectent cette règle [7]. La fidélité de

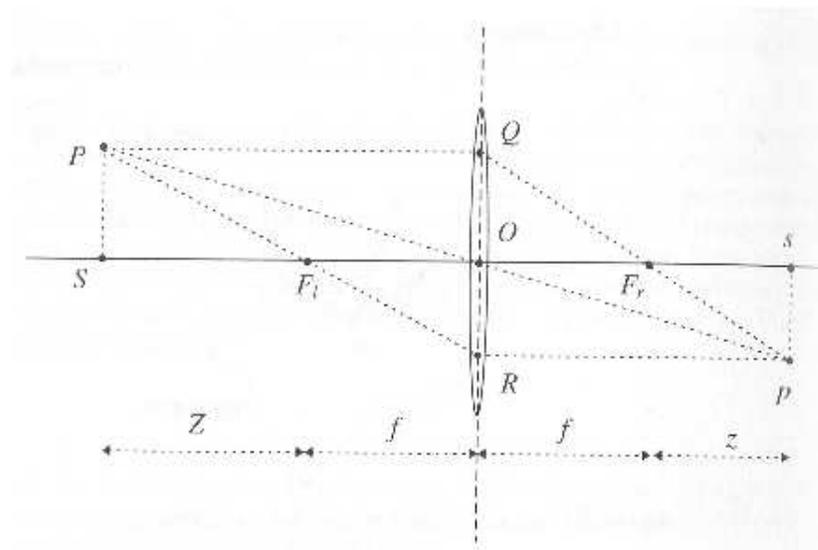


FIG. 2.2 – Système optique de lentille mince. Cette configuration idéale montre les rayons en provenance du point  $P$  tous focalisés au même point  $p$  (source : [7]).

l'image projetée sur la grille CCD par rapport à la scène est déterminée par la qualité de l'appareil optique. Deux facteurs sont d'intérêt particulier : la résolution du système de lentilles et les dégradations optiques de l'image.

### 2.2.0.1 Dégradations dues à l'appareil optique

La résolution du système de lentilles, sa capacité à projeter clairement des détails individuels de la scène, est limitée par le phénomène de diffraction [4] : une source lumineuse ponctuelle traversant une ouverture ne produit non pas un point sur le plan image mais un disque appelé tache d'Airy. Ce phénomène fixe une distance minimale entre deux sources ponctuelles en deçà de laquelle il est impossible d'en distinguer l'image, tel qu'illustré à la figure 2.3. Cette influence du système optique, qui a l'effet sur l'image projetée d'un filtre passe-bas (effet de flou), peut être mesurée grâce au critère de Rayleigh [4] et exprimée sous la forme d'une fonction de dispersion de points [6].

L'image projetée sur la grille CCD est de plus sujette à plusieurs types de

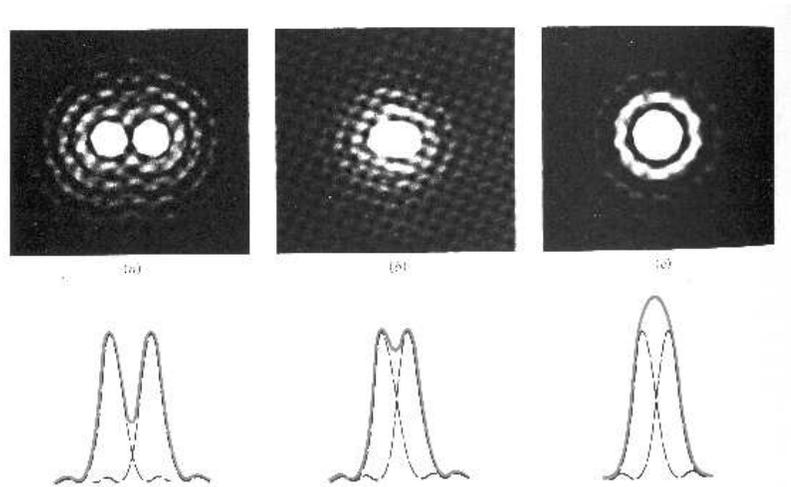


FIG. 2.3 – Diffraction démontrée à l’aide de deux sources lumineuses ponctuelles. On remarque qu’en deçà d’une certaine distance, il devient impossible de distinguer les deux taches d’Airy (source : [4]).

dégradations causées par l’appareil optique. Une liste incomplète inclut :

**l’aberration sphérique** qui survient lorsque les rayons incidents aux bords de la lentille sont focalisés à un endroit autre que le point où les rayons incidents au centre convergent [4] ;

**l’astigmatisme** qui désigne un défaut par lequel la clarté de la projection d’un objet sur le plan image varie selon son orientation [8] ;

**l’aberration chromatique** qui est un phénomène où les rayons de lumière sont projetés à un point légèrement différent dépendant de leur longueur d’onde, ce qui se solde par un effet de “prisme”, c’est-à-dire la décomposition de l’image d’un objet en plusieurs images de couleurs différentes et légèrement décalées les unes des autres [4] ;

**le vignettage** causé par l’illumination non-uniforme de la grille de photosenseurs par l’appareil optique et qui se solde par un assombrissement graduel de l’image plus on s’approche des coins [6].

Ces défauts peuvent être atténués, par exemple en ajoutant des lentilles présentant



FIG. 2.4 – Deux exemples d’aberrations : à gauche, des rayons lumineux de couleurs différentes sont focalisés à des points différents dans un exemple d’aberration chromatique ; à droite, l’aberration sphérique pousse les rayons à être redirigés vers des points différents selon leur point d’intersection avec la lentille (source : [4]).

une aberration inverse [4], et les appareils optiques modernes sont généralement conçus pour donner une qualité d’image optimale dans des conditions typiques, selon l’usage auquel la caméra est destinée. Bien qu’il soit possible d’inclure la contribution de certaines de ces aberrations au modèle de formation d’image, il est plus commun dans la littérature de super-résolution de les exclure pour simplifier la notation [2], et c’est un choix que nous partageons pour les fins de ce mémoire.

### 2.2.1 Le module CCD

Le module CCD est composé de photosenseurs organisés en éléments simples, en bande d’éléments ou grille d’éléments (figure 2.5). Cette dernière configuration est celle rencontrée dans les caméras digitales et est la plus conforme à la structure d’une image digitale, c’est-à-dire une matrice bidimensionnelle de pixels, tandis qu’une disposition en bande correspond, par exemple, au fonctionnement d’un scanner, et suppose un mouvement perpendiculaire à la bande d’éléments pour capter des images à deux dimensions.

Un élément CCD est constitué à partir d’un matériau conducteur (un polysilicium dopé) superposé à une couche de semiconducteur (silicium), les deux étages étant séparés par un matériau hautement isolant (dioxyde de silicium). Cette structure est reproduite à la figure 2.6. Lorsque l’élément est exposé à la lumière, l’énergie des photons est absorbée par le substrat de silicium, ce qui entraîne l’apparition

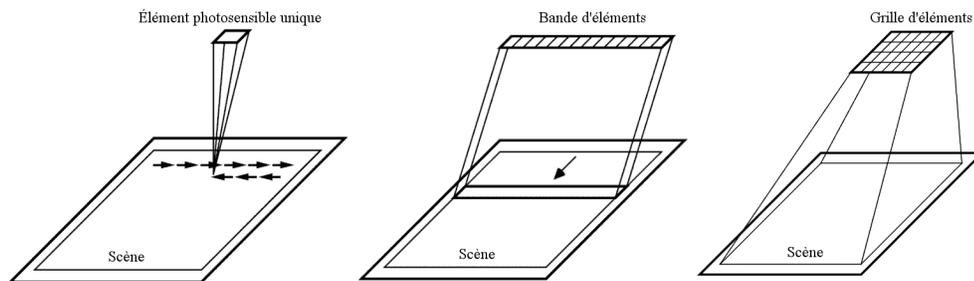


FIG. 2.5 – Illustration des trois configurations d’éléments photosensibles. Les flèches dénotent le mouvement du/des senseurs dans le cas d’un élément sensible ou d’une bande de senseurs (source : [5]).

d’une charge électrique, sous la forme d’électrons, linéairement proportionnelle à la quantité de lumière reçue par l’élément. Lorsqu’un voltage approprié est appliqué à l’étage de polysilicium, un “puits” de potentiel peut être formé dans le silicium vers lequel les électrons peuvent être canalisés [5]. Pour permettre la propagation de l’information lumineuse de chaque élément de la grille CCD vers l’interface digitale, une stratégie de transfert de charge doit être utilisée pour déplacer les électrons ainsi rassemblés en paquets vers un amplificateur de signal. Une méthode populaire, dite pleine trame (*full-frame* en anglais), implique le transfert rangée par rangée des paquets d’électrons vers un registre horizontal, qui redirige l’information vers l’amplificateur sous la forme de signal séquentiel. La simplicité de cette stratégie permet la construction de modules CCD avec une résolution maximale [5].

Chaque élément CCD communique une mesure de l’intensité reçue, c’est-à-dire une valeur exprimant la quantité de lumière absorbée par l’élément durant sa période d’exposition, toutes longueurs d’ondes confondues [5]. L’image ainsi générée est dite monochrome, ou en niveaux de gris : elle ne comporte aucune information sur la couleur, on l’appellera donc image d’intensité. Pour permettre la prise d’images couleurs, il existe trois techniques possibles, toutes visant l’acquisition de l’image selon les trois canaux de l’espace-couleur habituel RVB : le rouge, le vert et le bleu.

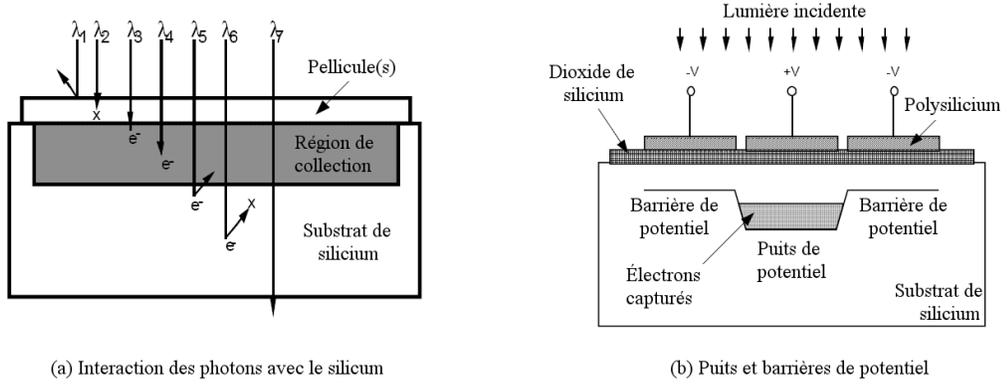


FIG. 2.6 – Illustration du fonctionnement d’un élément photosensible CCD (source : [5]).

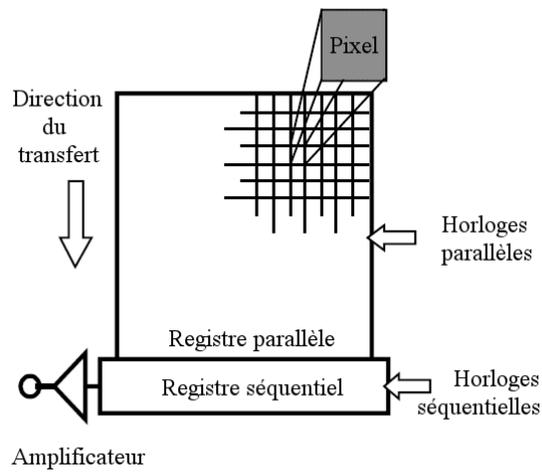


FIG. 2.7 – Transfert CCD pleine trame (source : [5]).

La première (figure 2.8a), conceptuellement très simple, consiste en la triple exposition de la grille CCD à la même scène, en appliquant successivement un filtre différent pour chacun des canaux. Cette technique permet de capter des images où chaque canal est de résolution égale à la grille CCD, mais limite la durée de chaque exposition à moins du tiers d’une exposition normale, tout en augmentant la complexité mécanique de la caméra.

Une seconde méthode (figure 2.8b) implique l’utilisation d’un système optique pour séparer la lumière entrante en trois faisceaux redirigés chacun vers une grille CCD dédiée exclusivement à un canal parmi les trois canaux RVB, mais cette approche est encore plus complexe que la précédente, notamment en raison de la difficile calibration des trois grilles nécessaire pour reconstruire l’image RVB sans décalage d’un canal à l’autre.

La dernière méthode requiert l’application d’un filtre individuel à chaque élément de la grille CCD selon une configuration conçue pour permettre la reconstruction subséquente de chaque canal. La figure 2.8c présente plusieurs configurations possibles<sup>1</sup>. Chaque canal présentant des “trous” là où se trouvent les éléments des deux autres couleurs dans la grille, cette méthode requiert l’interpolation des valeurs manquantes dans un calcul suivant l’acquisition de l’image. Ce désavantage est contrebalancé par la complexité grandement réduite du module CCD.

### 2.2.1.1 Dégradations dues à l’électronique du CCD

La scène projetée sur la grille CCD par l’appareil optique est continue. Or la grille est composée d’éléments discrets et l’image acquise est destinée à être digitale, c’est-à-dire qu’elle est emmagasinée en tant que structure de données comportant un nombre fini d’éléments. Cet état de fait implique deux choses. Dans un premier lieu, comme on l’a décrit plus haut, l’image de la scène, projetée sur la grille d’éléments du capteur, subit un échantillonnage : le signal continu est échantillonné

---

<sup>1</sup>La densité supérieure des éléments verts s’explique parce que le système visuel humain base démesurément sa perception de l’intensité dans la portion verte du spectre lumineux, tandis que sa perception de la couleur s’étale plus dans les régions du rouge et du bleu [6].

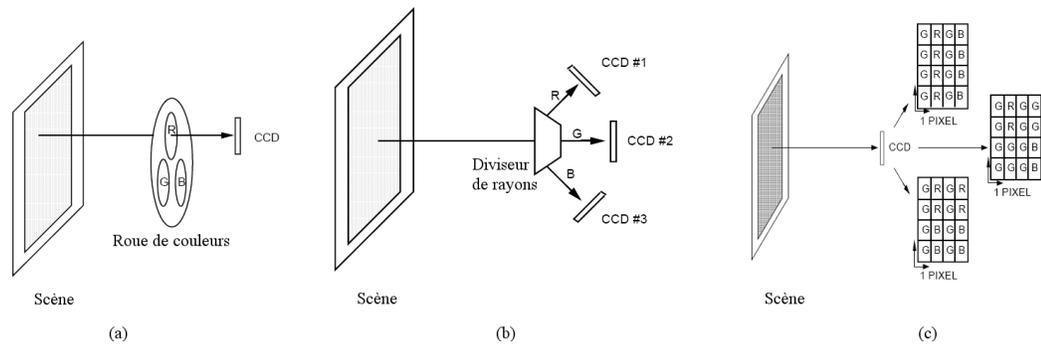


FIG. 2.8 – Trois méthodes de capter des images couleurs avec une caméra CCD (source : [5]).

à intervalles réguliers, selon une fréquence imposée par les dimensions de la grille de photosenseurs, et ce sont ces échantillons qui forment l'image<sup>2</sup>. Dans un second temps, la valeur de chaque pixel de l'image subit une quantification : l'intensité mesurée par l'élément est arrondie à une valeur entière sur une échelle allant du noir au blanc. Dans le cas d'images digitales monochromes stockées par un système informatique moderne, ces valeurs d'intensité vont de 0 pour le noir jusqu'à 255 pour le blanc. La figure 2.9 illustre l'opération.

Le théorème de Nyquist stipule que l'échantillonnage d'un signal doit être effectué à une fréquence deux fois supérieure à la plus haute fréquence de celui-ci pour qu'une reconstruction fidèle soit possible [9]. Si la fréquence d'échantillonnage est insuffisante, les composantes haute fréquence du signal original seront perdues et leur reconstruction se fera sous la forme de composantes plus basse fréquence absentes du signal original. Ce signal basse fréquence “fantôme” sera appelé un alias du signal haute fréquence, et on réfère au phénomène sous le terme d'aliassage [10]. Une solution à ce problème est l'augmentation de la fréquence d'échantillonnage, le moyen le plus direct étant l'augmentation de la densité des éléments dans la grille CCD, mais il est impossible dans les faits d'enrayer le problème : la résolution

<sup>2</sup>Trucco et Verri notent dans [7] que la grille de senseurs n'est pas nécessairement de dimensions égales à celles de l'image acquise, mais qu'il s'agit d'une simplification utile.

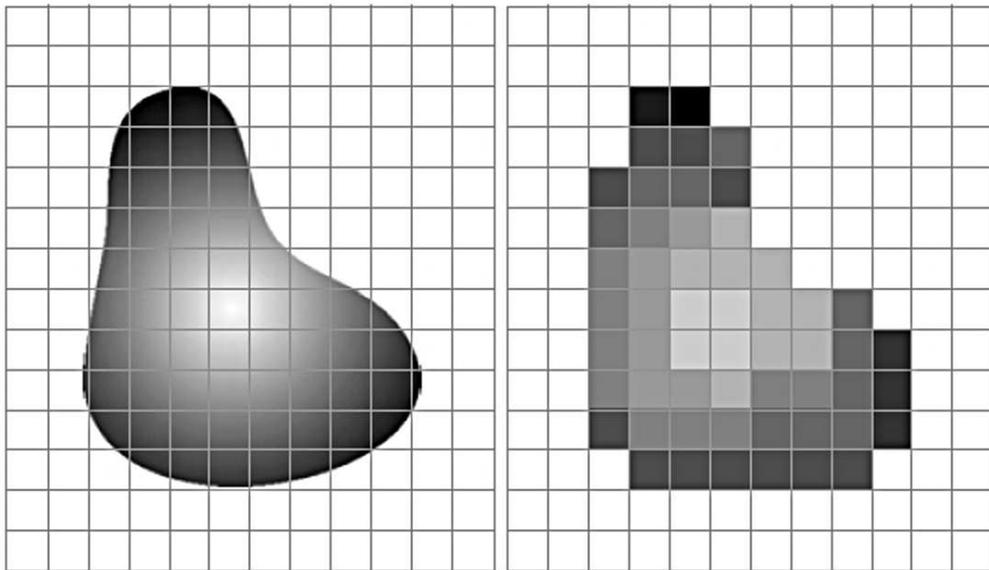


FIG. 2.9 – À gauche, scène continue superposée à la grille de senseurs. À droite, l'image résultante après échantillonnage et quantification (source : [1]).

des systèmes d'acquisition présente des limitations physiques d'une part, et d'autre part il existe des signaux de fréquence très haute, voir infinie, dans le monde qui nous entoure. L'aliassage est donc une réalité avec laquelle il faut composer, et les images utilisées dans ce mémoire sont considérées comme en présentant les effets.

Outre les effets dus à l'échantillonnage et à la quantification, un autre facteur contribuant à la dégradation de l'image digitale est le bruit, causé dans le cas d'une caméra CCD par trois sources principales [6]. Tout d'abord, la différence de sensibilité d'un élément à l'autre de la grille CCD contribue un premier élément de bruit, dit bruit à motif fixé (*fixed-pattern noise*) parce qu'il ne varie pas d'image en image. Ensuite vient le bruit grenaille (*shot noise*), causé par l'incertitude entourant la charge réelle emprisonnée dans le puits de potentiel du photosenseur. Parce que la mesure de l'intensité reçue par l'élément peut varier d'une évaluation à l'autre pour la même exposition, la valeur du pixel dans l'image présentera également une incertitude, et donc un élément de bruit. Finalement, la nature de la grille CCD ajoute un élément de bruit relié à la manipulation des charges électriques requise

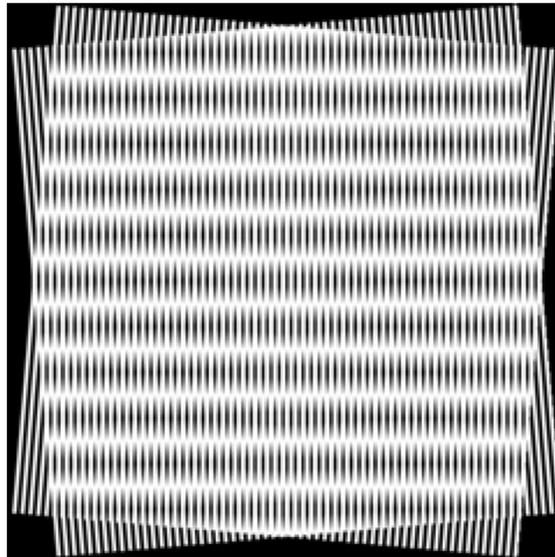


FIG. 2.10 – Le moiré, un exemple d’aliassage. (source : [1]).

par le processus d’acquisition.

Le blooming est une autre source de dégradation de l’image qui peut survenir durant l’acquisition. Ce phénomène se manifeste quand un élément CCD est sur-exposé, entraînant la formation d’une charge électrique supérieure à ce que le puits de potentiel de l’élément peut contenir [5]. Dans ce cas, les électrons surnuméraires débordent vers les éléments voisins, causant une mesure faussement élevée de l’intensité. Ce problème peut être corrigé en utilisant une caméra dotée d’une fonction “anti-blooming”, qui permet le drainage du surplus d’électrons sans qu’il y ait contamination du voisinage.

Finalement, le module CCD participe à l’effet de flou de la caméra initié dans un premier temps par l’appareil optique. Les électrons générés dans un élément CCD sous l’effet de la lumière ne sont pas nécessairement tous canalisés vers le puits de potentiel de cet élément après l’exposition : le champ électrique qui collecte les électrons ne pénètre pas uniformément la totalité de l’élément, et certains électrons peuvent demeurer dans ces régions dénuées de champ durant la phase de canalisation. Ces électrons “libres” peuvent diffuser vers les puits voisins aléatoirement,



FIG. 2.11 – Image originale (à gauche) et image comportant un bruit gaussien blanc d'écart-type  $\sigma = 25$ .

provoquant une contamination de la valeur d'un pixel par ses voisins, et donc un effet de flou. Comme le flou dû à l'appareil optique, ce fait est intégré au modèle de formation d'image au sein de la fonction de réponse impulsionnelle (PSF) de la caméra.

### 2.2.2 L'interface digitale et la représentation de l'image digitale

L'interface digitale transforme le signal vidéo transmis par le module CCD en image bidimensionnelle [7]. La représentation interne de l'image digitale ainsi créée étant arbitraire, nous pouvons directement décrire la représentation théorique que nous utiliserons pour la suite de ce travail. Nous définissons donc une image  $I$  sous la forme d'une matrice  $L \times W$ , où  $L$  dénote la hauteur et  $W$  dénote la largeur.

$$\mathbf{I} = \begin{pmatrix} i_{0,0} & i_{0,1} & \dots & i_{0,W-1} \\ i_{1,0} & i_{1,1} & \dots & i_{1,W-1} \\ \vdots & \vdots & \ddots & \vdots \\ i_{L-1,0} & i_{L-1,1} & \dots & i_{L-1,W-1} \end{pmatrix}$$



FIG. 2.12 – Exemple de blooming (source : <http://www.imaging-resource.com>).

Les valeurs contenues dans  $I$  sont des valeurs entières mesurant l'intensité du pixel correspondant, allant du noir (0) au blanc (255) et passant par 254 niveaux de gris. Cette représentation correspond exactement à la représentation informatique d'une image digitale monochrome. Bien que nous ne traitons pas d'images couleurs dans ce texte pour éviter d'en alourdir le contenu inutilement, toutes les méthodes décrites plus loin s'appliquent tout aussi bien aux images couleurs.

## CHAPITRE 3

### ACQUISITION ET TRAITEMENT DES DONNÉES

Ce chapitre développe les concepts et méthodes nécessaires à la super-résolution. Dans un premier lieu, nous décrivons un modèle génératif de l'image qui tient compte du processus d'acquisition présenté au chapitre précédent, en abordant les notions d'estimation du mouvement, de fonction de dispersion des points, et de bruit. Nous poursuivons avec la présentation d'une approche bayésienne menant à l'élaboration d'une stratégie de super-résolution. Nous procédons ensuite à un survol d'autres approches rencontrées dans la littérature pour arriver aux mêmes fins. Nous enchaînons avec la présentation des modèles *a priori* à l'étude dans ce mémoire et dont nous examinons les performances au chapitre suivant. Finalement, nous décrivons l'algorithme, basé sur notre approche bayésienne, avec lequel nous procédons à la super-résolution.

#### 3.1 Le modèle d'observation

On suppose une image haute-résolution de dimension  $L_1 N_1 \times L_2 N_2$  ordonnée lexicographiquement<sup>1</sup> sous la forme du vecteur  $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$  où  $N = L_1 N_1 L_2 N_2$ . L'image  $\mathbf{x}$  représente l'image haute-résolution idéale non-dégradée, échantillonnée à partir d'une scène continue à ou au-delà de la limite de Nyquist. Les paramètres  $L_1$  et  $L_2$  dénotent le facteur de sous-échantillonnage du modèle d'observation dans les dimensions horizontale et verticale respectivement. Les images basse-résolution, au nombre de  $p$ , seront donc de dimension  $N_1 \times N_2$ , et chaque pixel basse-résolution correspondra à une fenêtre de taille  $L_1 \times L_2$  dans l'image haute-résolution<sup>2</sup>. On dénote par  $\mathbf{y}_k = [y_{k,1}, y_{k,2}, \dots, y_{k,M}]^T$  le vecteur ordonné

---

<sup>1</sup>On entend par cette formule le fait d'ordonner les pixels de l'image en fonction de leurs coordonnées, de telle façon que le vecteur résultant soit composé des rangées de l'image juxtaposées l'une à la suite de l'autre.

<sup>2</sup>Cette fenêtre sera généralement déphasée de la grille de pixels de  $\mathbf{x}$  par un mouvement d'ordre fractionnel. Les dimensions de la fenêtre ne dénotent pas une correspondance directe d'un pixel

lexicographiquement de la  $k$ -ième image basse-résolution pour  $k = 1, 2, \dots, p$ , avec  $M = N_1 \times N_2$ .

Nous devons maintenant décrire le processus par lequel une image basse-résolution  $\mathbf{y}_k$  est générée à partir de l'image haute-résolution  $\mathbf{x}$ . Ce modèle servira à spécifier la méthode à suivre pour recouvrer une image haute-résolution reconstruite à partir de la séquence d'images basse-résolution.

Si on considère que  $\mathbf{x}$  demeure constante durant le processus d'acquisition, les images basse-résolution dépendent du mouvement de la caméra, d'une fonction de dispersion des points, de la décimation de l'image haute-résolution vers une résolution inférieure, et du bruit, et elles ne diffèrent qu'en raison du mouvement<sup>3</sup> et du bruit. Le flou (filtre passe-bas) peut être causé par le système optique (diffraction, aberration, focus, etc.), le mouvement (flou cinétique), les conditions atmosphériques et la réponse impulsionnelle du senseur. Le mouvement peut être obtenu par mouvement libre de la caméra, par mouvement contrôlé (vibration [11], sous-grille de senseurs [12]) ou dans le cas général, en utilisant des points de vue non restreints d'une même scène [13]. La décimation (c'est-à-dire le sous-échantillonnage par rapport à  $\mathbf{x}$ ) est le fruit de la grille d'éléments photosensibles de la caméra et produit des images arborant un élément d'aliasage. Le modèle génératif de l'image peut donc être formulé comme suit [2, 17] :

$$\mathbf{y}_k = \mathbf{D}\mathbf{F}_k\mathbf{M}_k\mathbf{x} + \mathbf{n}_k \quad \text{pour } 1 \leq k \leq p \quad (3.1)$$

où  $\mathbf{M}_k$  est la matrice de déformation (liée au déplacement), de taille  $L_1N_1L_2N_2 \times L_1N_1 \times L_2N_2$ ,  $\mathbf{F}_k$  est la matrice de dispersion des points (le flou), également de taille  $L_1N_1L_2N_2 \times L_1N_1 \times L_2N_2$ , et  $\mathbf{D}$  est une matrice de sous-échantillonnage de taille  $(N_1N_2)^2 \times L_1N_1L_2N_2$  opérant la décimation de l'image haute-résolution  $\mathbf{x}$  vers

---

de basse-résolution avec un groupe de pixels de  $\mathbf{x}$  mais bien sa taille en pixels dans  $\mathbf{x}$ . De plus, dépendant de la déformation due au mouvement et la perspective, cette fenêtre peut ne pas avoir une forme rectangulaire.

<sup>3</sup>C'est ce mouvement d'ordre fractionnel qui donne les différents points de vue qui nous permettront de calculer une estimation haute-résolution.

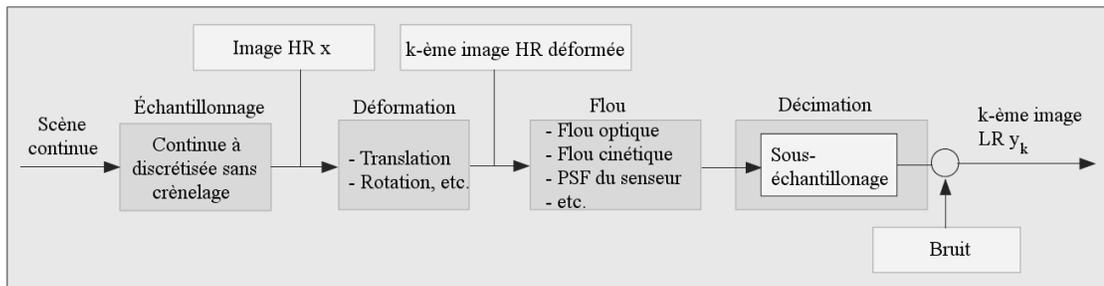


FIG. 3.1 – Le modèle d’observation de l’image (source : [2]).

l’image basse-résolution  $\mathbf{y}_k$ . Le vecteur  $\mathbf{n}_k$ , de longueur  $M = N_1 N_2$ , dénote quant à lui le bruit de l’image. Nous posons ci-après  $L_1 = L_2$ , laissant tomber l’indice et ne considérant que le facteur de magnification  $L$ , afin d’alléger la notation.

L’élaboration d’une stratégie de super-résolution, qui vise l’inversion du processus d’imagerie décrit en (3.1) (c’est-à-dire la reconstruction d’une estimation  $\hat{\mathbf{x}}$  de  $\mathbf{x}$ ), requiert dans un premier temps l’évaluation des matrices  $\mathbf{M}_k$  et  $\mathbf{F}_k$ , ainsi que du vecteur  $\mathbf{n}_k$ , et nous résumons dans les sections suivantes les démarches requises pour y arriver. L’estimation précise de ces paramètres a un impact important sur la qualité de la reconstruction super-résolution, et le facteur de magnification  $L$  de  $\hat{\mathbf{x}}$  par rapport aux dimensions de  $\mathbf{y}_k$  est limité par la justesse de ces mesures [18, 19].

### 3.1.1 Le recalage

Un problème commun en traitement d’images est celui d’établir une correspondance entre les pixels de deux ou plusieurs images prises dans des conditions différentes. Ces dernières peuvent renvoyer à la dimension temporelle (images saisies à des instants différents), spatiale (images acquises à partir de points de vue différents), ou modale (images prises avec des senseurs de type différent). Cet intérêt répandu s’est soldé par la mise au point au fil des ans d’un large éventail de méthodes pour y arriver. On peut retrouver une vue d’ensemble des méthodes répertoriées dans la littérature dans [20] et [21].

Dans le contexte de la super-résolution, nous disposons d’une série d’images

basse-résolution  $\mathbf{y}_k$ , qui constituent une collection de points de vue différents de l'image théorique  $\mathbf{x}$ , et dont nous désirons faire correspondre les pixels : il s'agit de l'estimation du mouvement, ou recalage. En pratique  $\mathbf{x}$  est inconnue, et nous devons choisir dans la série  $\mathbf{y}_k$  une image de référence vers laquelle les autres seront recalées. Un choix intuitif en présence d'images issues d'une séquence vidéo [22] est celui de l'image  $[\frac{p}{2}]$ , l'image du milieu, que nous dénoterons  $\mathbf{y}_r$ . Ce choix d'une image de référence fixe le point de vue de l'image reconstruite  $\hat{\mathbf{x}}$  ; autrement dit, on suppose que  $\mathbf{x}$  et  $\mathbf{y}_r$  partagent le même point de vue de la scène.

On distingue traditionnellement deux types de méthodes de recalage [7, 21] : les méthodes basées sur l'aire et les méthodes basées sur les caractéristiques. De ces deux types d'approches, les premières sont les plus souvent rencontrées en super-résolution. Cette prépondérance s'explique par leur simplicité et leur bonne performance pour recaler des images présentant des déplacements principalement translationnels, hypothèse commune en super-résolution. Ces méthodes procèdent à la mise en correspondance en comparant des blocs de pixels, et c'est pourquoi on les appelle également méthodes d'appariement de blocs. Cette approche requiert que l'on détermine la taille des blocs (ou fenêtres) à utiliser, la mesure de similarité entre les blocs à employer, et la stratégie à suivre pour conduire la recherche. L'algorithme 3.1 [7] présente une version simple qui opère l'appariement des blocs au sens de l'erreur quadratique moyenne en limitant l'amplitude du déplacement en pixels à  $d = (d_1, d_2)$ . Une variante fréquemment rencontrée est l'appariement de blocs hiérarchique, qui se fonde sur une démarche multi-résolution : une représentation pyramidale de chaque image est calculée (figure 3.2), et on procède à la mise en correspondance en commençant par la résolution la plus faible. Chaque "étage" de la pyramide permet de raffiner les mesures prises à l'étape précédente.

Notons que si on utilise un algorithme du type décrit ci-haut, il est nécessaire de précéder le recalage par l'interpolation des images basse-résolution par un facteur de  $L$  [3] : en effet, l'appariement se faisant sur des blocs de pixels entiers, cette interpolation est nécessaire pour mesurer le mouvement avec une précision fractionnelle. Une méthode utilisée dans [22] s'inspire de la variante hiérarchique

TAB. 3.1 – Algorithme d'appariement de blocs

<p><b>Algorithme d'appariement de blocs</b></p> <p><b>Entrées :</b> une paire d'images, <math>I_1</math> et <math>I_2</math>.</p> <p><b>Sorties :</b> un tableau des vecteurs du mouvement de <math>I_1</math> vers <math>I_2</math>.</p>
<p>Soit :</p> <ul style="list-style-type: none"> <li>– <math>p_1(i, j)</math>, un pixel de <math>I_1</math>.</li> <li>– <math>L = 2W + 1</math>, la largeur du bloc.</li> <li>– <math>R(p_1)</math>, la région de recherche dans <math>I_2</math> pour le pixel <math>p_1</math>.</li> <li>– <math>d = (d_1, d_2)</math>, un déplacement de <math>d_1</math> pixels sur la verticale et <math>d_2</math> pixels sur l'horizontale.</li> </ul> <p>Pour tout <math>p_1(i, j)</math> :</p> <p style="padding-left: 2em;">Pour tout <math>d \in R(p_1)</math> :</p> $c(d) = \sum_{k=-W}^W \sum_{l=-W}^W (I_1(i+k, j+l) - I_2(i+l+d_1, j+l+d_2))^2$ <p><math>\bar{d} = \arg \min_{d \in R(p_1)} c(d)</math> est le déplacement observé au pixel <math>p_1</math> de <math>I_1</math> à <math>I_2</math>.</p> <p>Retourner le tableau des <math>\bar{d}</math>.</p>

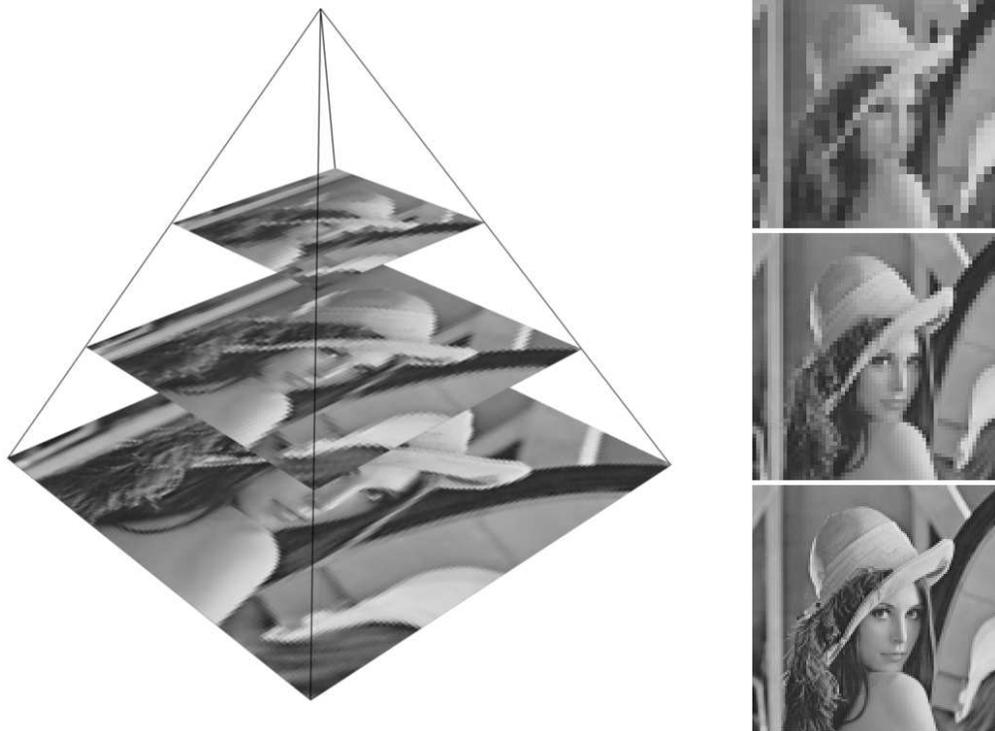


FIG. 3.2 – Pyramide multi-résolution utilisée pour l'appariement de blocs hiérarchique

de l'appariement de blocs afin de permettre une précision arbitraire : la pyramide multi-résolution est augmentée avec des versions interpolées à des facteurs itérativement plus grands, ce qui permet un raffinement subpixel de la mise en correspondance avec une précision choisie en fonction des besoins.

Le recalage est fréquemment effectué en prétraitement, c'est-à-dire durant une phase de calcul précédant la reconstruction super-résolution elle-même [2]. Le cadre particulier de la super-résolution permet d'aborder le problème de l'estimation du mouvement sous un angle différent. Au lieu de procéder au recalage pendant la phase de prétraitement, un nombre croissant de techniques de super-résolution abordent le champ de vecteurs du mouvement comme une quantité inconnue à reconstruire parallèlement à l'estimation haute-résolution  $\hat{\mathbf{x}}$ , notamment à l'aide d'itérations alternées de descente du gradient. Une méthode utilisée dans [11] s'apparente à l'appariement de blocs, mais diffère en ce que les images basse-résolution  $\mathbf{y}_k$  sont comparées non pas à l'image de référence  $\mathbf{y}_r$  mais à  $\hat{\mathbf{y}}_k = \mathbf{D}\mathbf{F}_k\mathbf{M}_k\hat{\mathbf{x}}$ , l'image basse-résolution simulée en appliquant le modèle d'observation en (3.1) à l'estimation haute-résolution  $\hat{\mathbf{x}}$ . Cette approche tire avantage de la restauration obtenue à chaque itération et permet d'envisager une amélioration de la performance par rapport aux méthodes conventionnelles.

Pour les techniques de super-résolution qui ne procèdent pas à l'estimation du mouvement parallèlement à la restauration, le recalage fait souvent partie de l'information présupposée connue. Dans notre implantation, nous empruntons cette approche : les images basse-résolution sont simulées à partir d'une image haute-résolution selon un mouvement choisi par nous pour générer les matrices du système linéaire. Cette simplification permet d'isoler la performance des modèles *a priori* que nous désirons évaluer de celle du recalage.

### 3.2 La fonction de dispersion de points

La détermination d'une fonction de dispersion de point a pour but de définir formellement la relation entre un pixel basse-résolution et les pixels haute-résolution

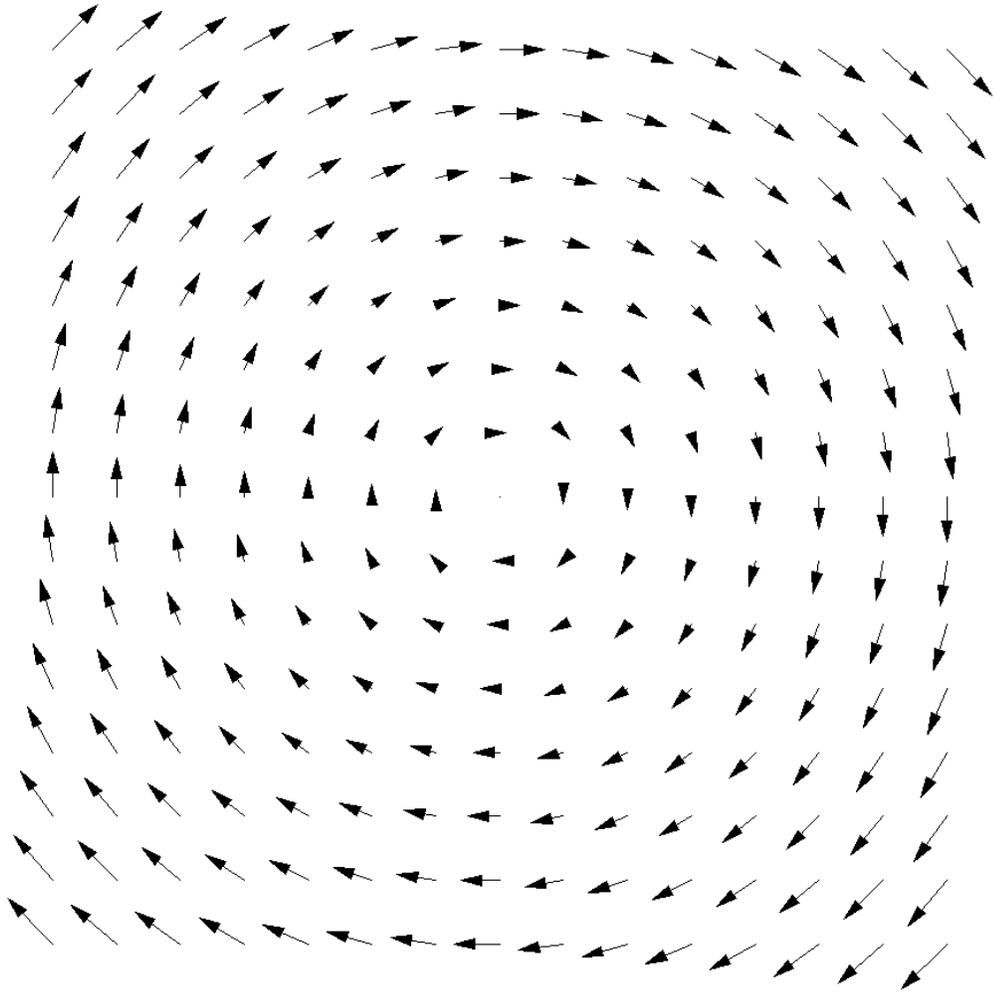


FIG. 3.3 – Un champ de vecteurs, le résultat du recalage.

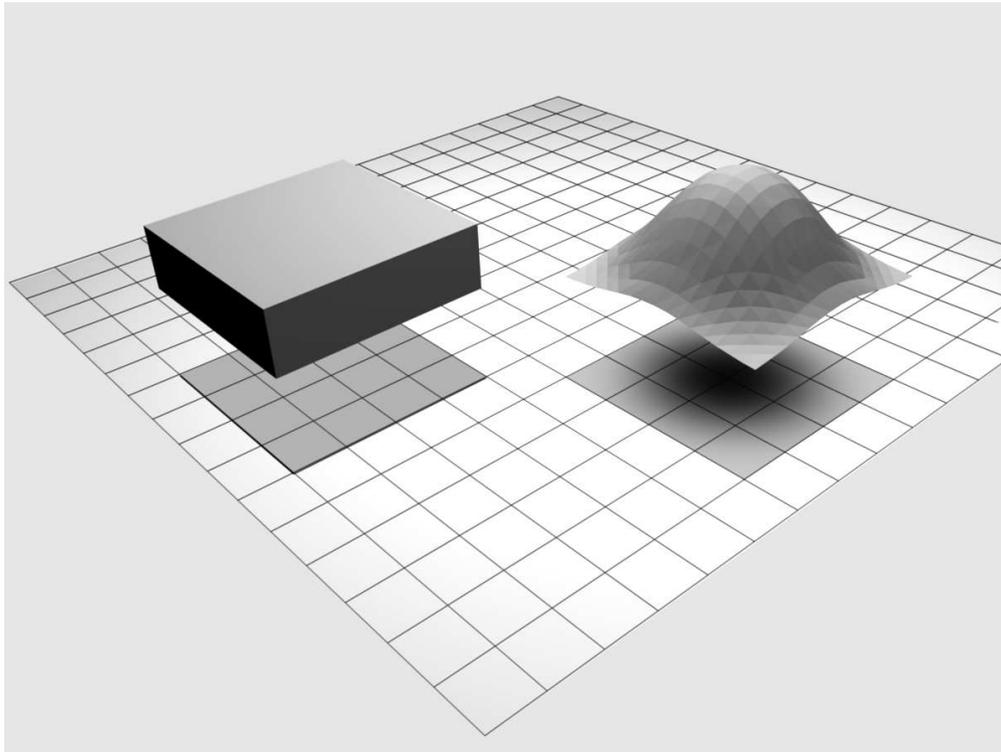


FIG. 3.4 – Illustration de l'influence des pixels HR sur un pixel BR : à gauche, une fonction boîte assigne une importance égale à tous les pixels HR ; à droite, une fonction gaussienne accorde un plus grand poids aux pixels HR qui sont proches du centre du pixel BR.

qui l'influencent (voir figure 3.4).

La fonction de dispersion de points pour la  $k$ -ième image peut être exprimée sous la forme [18, 23] :

$$F_k = \text{PSF}_{\text{senseur}} * \text{PSF}_{\text{lentille}} \quad (3.2)$$

où  $\text{PSF}_{\text{senseur}}$  dénote la fonction de dispersion de points du senseur et  $\text{PSF}_{\text{lentille}}$  la fonction de dispersion de points du système de lentilles. La fonction  $\text{PSF}_{\text{lentille}}$  peut être de plus divisée en deux facteurs [23] modélisant la défocalisation et la diffraction dues au système optique. Ces facteurs réfèrent aux imperfections de l'image énoncées au chapitre précédent. Il est également possible d'inclure à la fonction de dispersion de points la modélisation du flou cinétique [13].

Les recherches en restauration d'images ont permis la mise au point de nombreuses méthodes d'identification du flou [24–29]. Ces méthodes se divisent en deux approches principales [28] : l'identification *a priori* du flou, qui procède à l'identification de la fonction de dispersion lors d'une opération préalable, par exemple en mesurant la réponse de la caméra lorsque exposée à une source de lumière ponctuelle ; et l'identification du flou dans un processus parallèle à la restauration d'image. Cette dernière approche est celle empruntée par exemple dans [30], qui propose une méthode de super-résolution qui raffine itérativement le modèle du flou à l'aide d'une minimisation alternée de fonctions de coûts dans un sens bayésien. En pratique, en super-résolution, il est plus commun [31] de supposer la fonction de dispersion de points connue au préalable, et cette dernière est fréquemment [18,23] modélisée à l'aide d'une fonction boîte :

$$F_k(x, y) = \begin{cases} L^{-2} & |x|, |y| \leq \frac{L}{2} \\ 0 & \text{autrement} \end{cases}$$

où  $L$  dénote la grandeur du côté du pixel basse-résolution dans l'image haute-résolution et les coordonnées  $(x, y)$  réfèrent aux pixels haute-résolution correspondant au pixel basse-résolution à l'étude.

### 3.3 Le bruit

Le bruit de l'image est le résultat de l'électronique de l'acquisition tel que décrit au chapitre précédent. La quantité et la nature du bruit introduit par une caméra peuvent être évaluées en prenant plusieurs images d'une scène statique et en mesurant la variation entre les images [7]. Le bruit est presque universellement présumé être gaussien blanc additif dans la littérature, et c'est une convention que nous respecterons. Ce type de bruit présente le double avantage d'être facile à manipuler mathématiquement et de permettre de bien modéliser le bruit habituellement rencontré en traitement d'images [1]. Sa distribution s'énonce comme

suit :

$$P_Z(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp \frac{-(z - \mu)^2}{2\sigma^2} \quad (3.3)$$

où  $z$  dénote la valeur observée du pixel,  $\mu$  dénote sa valeur moyenne (ou réelle) et  $\sigma^2$  la variance du bruit.

À noter que le terme  $\mathbf{n}_k$  dans l'équation (3.1) peut également dénoter la différence  $\mathbf{n}_k = \mathbf{y}_k - \mathbf{D}\mathbf{F}_k\mathbf{M}_k\hat{\mathbf{x}}$  entre l'image basse-résolution  $\mathbf{y}_k$  et l'image basse-résolution obtenue en simulant le processus d'acquisition sur l'estimation haute-résolution  $\hat{\mathbf{x}}$ . Cette interprétation concentre dans  $\mathbf{n}_k$  la variation due au bruit ainsi que celle due aux erreurs d'estimation des paramètres (recalage et PSF), et est utile dans la justification des méthodes de super-résolution présentées plus loin dans le texte.

### 3.3.1 Alternatives et implantation

Le modèle présenté en (3.1) est fondé sur l'hypothèse que l'image  $\mathbf{y}_k$  est le résultat, dans l'ordre, d'une déformation (ou mouvement), d'un filtre passe-bas, d'une décimation et d'un élément de bruit. Ce modèle se calque sur le processus d'acquisition typique d'une caméra CCD et s'applique fidèlement aux images acquises dans des conditions où le filtre passe-bas est dominé par la caméra, c'est-à-dire où le système optique est la cause principale du flou. Un modèle alternatif [2] interchange la position de la déformation et du flou dans l'équation, comme au modèle suivant :

$$\mathbf{y}_k = \mathbf{D}\mathbf{M}_k\mathbf{F}_k\mathbf{x} + \mathbf{n}_k \quad \text{pour } 1 \leq k \leq p. \quad (3.4)$$

Cette variation du modèle est appropriée dans les cas où la source principale du flou est extérieure à la caméra, comme par exemple en présence de conditions atmosphériques défavorables. Ce modèle alternatif présente un intérêt supplémentaire soulevé dans [32]. En effet, bien que le modèle (3.1) soit celui qui exprime le plus fidèlement le processus d'acquisition typique d'une caméra CCD, les travaux dans [32] tendent à indiquer que les méthodes de reconstruction basées sur le modèle (3.4) permettent une reconstruction de qualité supérieure quand les pa-

ramètres de mouvement (c'est-à-dire le recalage) sont évalués à partir des images basse-résolution. Ce comportement peu intuitif doit néanmoins être pris en compte dans l'implantation d'applications de super-résolution. Cependant, puisque nous faisons usage dans ce texte d'images basse-résolution simulées et dont le mouvement est donc parfaitement déterminé, cette nuance n'est mentionnée que par souci de précision.

Le modèle utilisé dans notre implantation est une reformulation populaire [2,18] du modèle en (3.1), duquel il diffère en ce que les matrices  $\mathbf{D}$ ,  $\mathbf{F}_k$  et  $\mathbf{M}_k$  sont fusionnées dans la matrice  $\mathbf{W}_k$  :

$$\mathbf{y}_k = \mathbf{W}_k \mathbf{x} + \mathbf{n}_k. \quad (3.5)$$

Cette matrice condense la totalité du processus d'acquisition en une seule matrice (à l'exception du bruit) et permet d'exprimer l'influence des pixels haute-résolution sur les pixels basse-résolution dans un format relativement compact. Le reste de cette section décrit la construction de cette matrice  $\mathbf{W}_k$  et clarifie la représentation interne de l'influence entre pixels dont nous avons discuté jusqu'ici plutôt abstraitement.

La matrice  $\mathbf{W}_k$  exprime la relation entre l'image haute-résolution  $\mathbf{x}$  et l'image basse-résolution  $\mathbf{y}_k$  selon un calcul que l'on peut décrire comme suit. Soit  $y_{k,m}$ , le  $m$ -ième pixel de l'image vectorisée  $\mathbf{y}_k$  dans l'ordre lexicographique défini plus haut ; la valeur de  $y_{k,m}$  est le résultat du produit scalaire de la  $m$ -ième rangée de  $\mathbf{W}_k$  avec l'image haute-résolution vectorisée  $\mathbf{x}$  (on fait ici abstraction de la contribution du bruit à la valeur de  $y_{k,m}$ ). Nous construisons la  $m$ -ième rangée de  $\mathbf{W}_k$  en nous basant sur l'information acquise lors du recalage ainsi que sur le modèle du flou défini par la fonction de dispersion de points. Une méthode simple pour déterminer  $w_{k,m,r}$ , le poids à accorder au  $r$ -ième pixel  $x_r$  de  $\mathbf{x}$  dans la valeur de  $y_{k,m}$  est de calculer la proportion de la fenêtre de dimension  $L_1 \times L_2$  du pixel  $y_{k,m}$  projetée dans l'image  $\mathbf{x}$  qui est recouverte par  $x_r$ . Nous pouvons noter deux choses au sujet de cette méthode : tout d'abord, ce calcul correspond

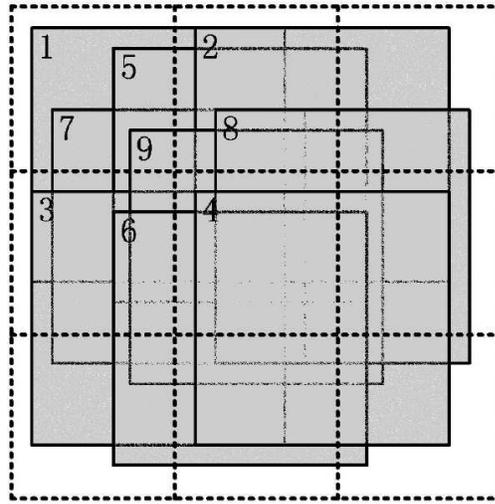


FIG. 3.5 – Illustration de la relation entre les pixels BR et HR. Les pointillés dénotent la grille HR comportant 9 pixels. Les rectangles numérotés représentent des pixels BR. Le facteur de magnification  $L = \frac{25}{16}$  (source : [18]).

au comportement de la fonction de dispersion de type boîte énoncée plus haut ; ensuite, hormis circonstances exceptionnelles, un très faible nombre des pixels  $x_r$  de  $\mathbf{x}$  influenceront  $y_{k,m}$ , et la matrice  $\mathbf{W}_k$  sera donc creuse, c'est-à-dire qu'elle comportera une large majorité d'éléments de valeur nulle. La matrice  $\mathbf{E}$  et la figure 3.5 illustrent la structure de  $\mathbf{W}_k$ .

$$\mathbf{E} = \frac{1}{625} \begin{pmatrix} 196 & 154 & 0 & 154 & 121 & 0 & 0 & 0 & 0 \\ 0 & 196 & 154 & 0 & 154 & 121 & 0 & 0 & 0 \\ 0 & 0 & 0 & 196 & 154 & 0 & 154 & 121 & 0 \\ 0 & 0 & 0 & 0 & 196 & 154 & 0 & 154 & 121 \\ 72 & 192 & 36 & 78 & 208 & 39 & 0 & 0 & 0 \\ 0 & 0 & 0 & 72 & 192 & 36 & 78 & 208 & 39 \\ 72 & 78 & 0 & 192 & 208 & 0 & 36 & 39 & 0 \\ 0 & 72 & 78 & 0 & 192 & 208 & 0 & 36 & 39 \\ 16 & 64 & 20 & 64 & 256 & 80 & 20 & 80 & 25 \end{pmatrix} \quad (3.6)$$

Le modèle (3.5) nous fournit une équation linéaire par image. Avec  $p$  images, nous pouvons “empiler” l’équation pour chaque image afin d’obtenir [13] :

$$\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_p \end{bmatrix} = \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_p \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \vdots \\ \mathbf{n}_p \end{bmatrix} \quad (3.7)$$

que nous dénotons :

$$\mathbf{y} = \mathbf{W}\mathbf{x} + \mathbf{n}. \quad (3.8)$$

### 3.4 Reconstruction MV

Une première piste de solution pour résoudre l’équation (3.8) pour  $\mathbf{x}$  est la reconstruction au sens du maximum de la vraisemblance (MV) [13]. Cette approche cherche à maximiser la fonction de densité de probabilité conditionnelle des données connaissant l’image haute-résolution  $\mathbf{x}$  :

$$\hat{\mathbf{x}}_{ML} = \arg \max_x P_{Y|X}(y|x). \quad (3.9)$$

En isolant  $\mathbf{n}$  et en substituant  $\mathbf{x}$  par  $\hat{\mathbf{x}}$  dans (3.8), nous obtenons :

$$\mathbf{y} - \mathbf{W}\hat{\mathbf{x}} = \mathbf{n}. \quad (3.10)$$

Cette équation renvoie à l’interprétation de  $\mathbf{n}$  qui en fait non seulement le vecteur du bruit dû à l’électronique du capteur, mais également le terme incarnant l’erreur due au recalage et à la modélisation de la fonction de dispersion des points. En suivant la convention dans la littérature, laquelle présuppose un bruit gaussien blanc, nous pouvons exprimer la probabilité du vecteur d’échantillons  $\mathbf{y}$  connaissant

une estimation  $\hat{\mathbf{x}}$  de  $\mathbf{x}$  (c'est-à-dire la vraisemblance) comme suit :

$$P_{Y|X}(\mathbf{y}|\hat{\mathbf{x}}) = \prod_{i=1}^{pM} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y_i - \hat{y}_i)^2}{2\sigma^2}\right) \quad (3.11)$$

où  $\hat{y}_i$  est le  $i$ -ème élément du vecteur  $\hat{\mathbf{y}} = \mathbf{W}\hat{\mathbf{x}}$ ,  $p$  est le nombre d'images et  $M = N_1 \times N_2$  dénote le nombre de pixels dans chacune. La fonction de log-vraisemblance correspondante se décline comme suit :

$$\begin{aligned} \mathcal{L}(\mathbf{y}) &= -\sum_{i=1}^{pM} (y_i - \hat{y}_i)^2 \\ &= -\|\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}\|^2. \end{aligned} \quad (3.12)$$

L'estimation MV  $\mathbf{x}_{MV}$  peut donc être exprimée ainsi :

$$\begin{aligned} \mathbf{x}_{MV} &= \arg \max_x -\|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 \\ &= \arg \min_x \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2. \end{aligned} \quad (3.13)$$

Or, l'équation (3.13) correspond à résoudre au sens de l'erreur des moindres carrés le système  $\mathbf{W}\mathbf{x} = \mathbf{y}$  [18], avec une solution de la forme  $\mathbf{x} = \mathbf{W}^+\mathbf{y}$  où  $\mathbf{W}^+ = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T$  est la pseudo-inverse Moore-Penrose de  $\mathbf{W}$ .

La matrice  $\mathbf{W}$  dans l'équation (3.8) est creuse et de taille proportionnelle au nombre d'échantillons (pixels), et la charge de calcul requise pour calculer sa pseudo-inverse est généralement trop lourde pour être envisagée en pratique [2, 13]. Il sera donc nécessaire de faire appel à des méthodes indirectes pour recouvrer une estimation de l'image haute-résolution  $\mathbf{x}$ , notamment par le biais de méthodes itératives.

La reconstruction d'une image haute-résolution à partir d'une séquence basse-résolution est un problème inverse, c'est-à-dire un problème qui cherche à recouvrer les données initiales d'un mécanisme fonctionnel direct (dans notre cas, la génération d'une image basse-résolution à partir d'une image haute-résolution dans

l'équation (3.5)) à partir de sa solution. Dans un sens, nous désirons renverser le processus d'imagerie.

Hadamard définit un problème comme étant bien posé si trois conditions sont réunies [33] :

- une solution au problème *existe* ;
- la solution est *unique* ;
- la solution dépend des données de façon *continue*.

Le problème de l'estimation d'une image haute-résolution à partir d'une séquence d'images, à l'instar de la majorité des problèmes inverses, est dit mal posé parce qu'il contrevient à ces règles de plusieurs façons. Tout d'abord, plusieurs images haute-résolution (légèrement) différentes peuvent être à l'origine d'une même image basse-résolution donnée [22]. D'autre part, l'estimation de  $\mathbf{x}$  est trop sensible au bruit et aux erreurs de modélisation du mouvement et du flou, ce qui entraîne de larges variations dans l'estimation  $\hat{\mathbf{x}}$  sous l'influence de faibles changements dans  $\mathbf{W}$  et  $\mathbf{y}$  [34], ce qui déroge au troisième critère d'Hadamard. Cette sensibilité aux paramètres provoque une dégradation rapide de la qualité de  $\hat{\mathbf{x}}$  proportionnellement au facteur de magnification  $L$ , comportement à proscrire de toute méthode de super-résolution que l'on désire implanter.

Devant cet état de fait, nous devons contraindre le problème afin de rejeter les solutions sous-optimales à notre sens en faveur d'une solution présentant des qualités que nous recherchons. Nous appelons ceci la régularisation du problème, et une approche populaire pour y arriver est par le biais d'une reconstruction MAP (Maximum *A Posteriori*) dans le cadre bayésien, qui est l'objet de la prochaine section.

### 3.5 Reconstruction MAP

La reconstruction maximum a posteriori (MAP) cherche à maximiser la probabilité *a posteriori* de  $\mathbf{x}$  sachant  $\mathbf{y}$  :

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} P_{X|Y}(\mathbf{x}|\mathbf{y}). \quad (3.14)$$

Or, la loi de Bayes pose :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (3.15)$$

où  $P(B|A)$  pour un  $B$  connu est la vraisemblance de  $A$ ,  $P(A)$  dénote la probabilité *a priori* de  $A$  et  $P(B)$  la probabilité *a priori* de  $B$ , ce qui nous permet d'écrire :

$$\begin{aligned} \hat{\mathbf{x}} &= \arg \max_{\mathbf{x}} P_{X|Y}(\mathbf{x}|\mathbf{y}) \\ &= \arg \max_{\mathbf{x}} \frac{P_{Y|X}(\mathbf{y}|\mathbf{x})P_X(\mathbf{x})}{P_Y(\mathbf{y})} \\ &= \arg \max_{\mathbf{x}} P_{Y|X}(\mathbf{y}|\mathbf{x})P_X(\mathbf{x}) \end{aligned} \quad (3.16)$$

$$= \arg \max_{\mathbf{x}} \{ \ln P_{Y|X}(\mathbf{y}|\mathbf{x}) + \ln P_X(\mathbf{x}) \}. \quad (3.17)$$

Dans l'équation (3.16),  $P_{Y|X}(\mathbf{y}|\mathbf{x})$  renvoie à l'équation (3.13) et la fonction de log-vraisemblance correspondante tandis que  $P_X(\mathbf{x})$  est appelée probabilité *a priori* de  $\mathbf{x}$  et représente une contrainte sur  $\hat{\mathbf{x}}$ . Ce terme requiert que nous émettions une hypothèse sur les propriétés de  $\hat{\mathbf{x}}$  qui nous permettra de régulariser le problème afin d'aboutir à une seule solution optimale. Nous obtenons la minimisation suivante :

$$\hat{\mathbf{x}}_{MAP} = \arg \min_{\mathbf{x}} \{ \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 + \alpha\Omega(\mathbf{x}) \} \quad (3.18)$$

où le terme  $\|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2$  est communément appelé le terme de fidélité aux données,  $\Omega(\mathbf{x})$  est une fonction qui mesure la conformité de  $\hat{\mathbf{x}}$  à une propriété déterminée au préalable : il s'agit du terme de pénalisation ou terme *a priori*.  $\alpha$  est un paramètre de

régularisation contrôlant le poids accordé à  $\Omega(\mathbf{x})$  dans la reconstruction : un poids faible laisse dominer le terme de fidélité aux données tandis qu'un poids fort permet à  $\Omega(\mathbf{x})$  d'exercer une influence importante sur les résultats de la reconstruction. La fonction  $\Omega(\mathbf{x})$  doit également permettre de régulariser le problème en contraignant le problème à une solution unique.

La qualité de la reconstruction  $\hat{\mathbf{x}}$  dépend donc d'un choix judicieux de  $\Omega(\mathbf{x})$ .

### 3.6 Le choix du terme *a priori*

Si le terme de fidélité aux données pénalise les solutions qui ne concordent pas au modèle d'acquisition des images basse-résolution, le terme *a priori*  $\Omega(\mathbf{x})$  contraint la solution à des images haute-résolution présentant des propriétés jugées souhaitables pour  $\hat{\mathbf{x}}$ . Ce terme reflète une hypothèse sur la nature de l'image  $\mathbf{x}$  idéale recherchée : il favorise les solutions qui présentent une certaine propriété, que celle-ci soit présente dans l'image  $\mathbf{x}$  ou non. Il est donc nécessaire de faire un choix judicieux dans la forme de  $\Omega(\mathbf{x})$  afin d'obtenir une image  $\hat{\mathbf{x}}$  proche de  $\mathbf{x}$ . Ce choix est non-trivial et est l'objet d'un volume important de littérature ; [2], [13], [16], [37] ne sont que quelques articles qui présentent une variété de modèles *a priori*.

### 3.7 Autres approches

La méthode de super-résolution utilisée dans ce mémoire est désignée dans la littérature sous l'appellation reconstruction régularisée [2] ou algorithme basé sur la reconstruction [18]. Cette approche est l'une des plus populaires en super-résolution ; cependant, d'autres méthodes visant le même objectif ont été proposées, et nous présentons ici brièvement quelques-unes des mieux connues. Pour un panorama plus détaillé et une bibliographie étoffée, nous référons le lecteur à la synthèse de Park *et al.* [2].

La méthode la plus "intuitive" de super-résolution, celle qui s'inscrit le plus naturellement dans la manipulation directe d'images digitales, est celle dite d'interpolation non-uniforme. Cette méthode s'opère en trois étapes : 1) le recalage,

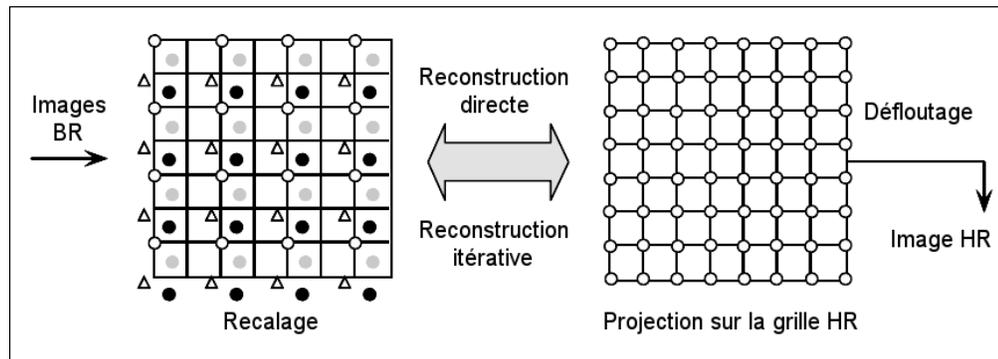


FIG. 3.6 – Exemple d’interpolation non-uniforme : les points foncés et les triangles représentent des pixels non-alignés sur la grille HR.

dont nous avons présenté un aperçu dans le contexte de la super-résolution plus haut dans ce texte ; 2) l’interpolation non-uniforme proprement dite ; et 3) la restauration de l’estimation composite générée à l’étape précédente, qui peut inclure, dépendant du modèle de génération de l’image, un débruitage et une suppression du flou. La figure 3.6 illustre l’intuition derrière l’interpolation non-uniforme : les pixels des images recalées sont utilisés pour interpoler la valeur des pixels de la grille HR, selon une stratégie qui pourra, par exemple, faire appel à divers types de pavages [35], tels entre autres la triangulation de Delaunay [36]. Cette approche présente d’une part l’avantage de l’aisance d’implantation et d’un coût de calcul relativement faible, au détriment de la flexibilité du modèle d’imagerie. Cette méthode implique aussi une super-résolution en deux étapes (calcul de l’apport des échantillons et restauration) qui rend plus complexe le raisonnement sur les résultats de la super-résolution.

Une autre méthode est celle dite de projection vers des ensembles convexes, ou POCS (*Projection onto Convex Sets*). Il s’agit d’une approche fondée dans la théorie des ensembles qui émane du raisonnement suivant : intégrer la contribution du modèle *a priori* à la solution finale équivaut à contraindre celle-ci à une famille d’ensembles convexes  $C_i$ , chacun desquels étant défini comme l’ensemble des vecteurs-solution qui satisfont une certaine propriété prescrite dans le modèle

*a priori*. La solution, si elle existe, est donc nécessairement comprise dans l'intersection convexe des ensembles  $C_i$ , dénoté  $C_s = \cap_{i=1}^m C_i$ , qui représente l'ensemble des solutions qui satisfont toutes les propriétés voulues par le modèle. La solution peut alors être calculée par projection alternée de la solution sur les ensembles  $C_i$ , selon la règle suivante :

$$\hat{x}^{n+1} = P_m P_{m-1} \dots P_2 P_1 \hat{x}^n \quad (3.19)$$

où  $x^0$  est un vecteur de départ arbitraire et  $P_i$  est un opérateur de projection qui projette un vecteur donné vers l'ensemble  $C_i$ . Les contraintes imposées par projection vers les ensembles  $C_i$  doivent idéalement favoriser la fidélité aux données et appliquer les effets du modèle *a priori* concurremment, de façon analogue à la méthode de reconstruction régularisée telle que présentée à l'équation (3.18). La méthode POCS présente des avantages similaires à ceux de la méthode par reconstruction régularisée, c'est-à-dire la simplicité théorique et l'aisance d'application du modèle *a priori* ; cependant, elle souffre d'une convergence lente, un coût de calcul élevé et une solution non-unique.

Une autre approche aborde le problème dans le domaine fréquentiel. Cette méthode s'appuie sur les propriétés de la transformée de Fourier et sur un modèle d'imagerie simplifié, qui permettent la formulation d'un système d'équations mettant en relation les transformées de l'image HR et des images BR, et qui peut être exploité pour arriver à la super-résolution. L'avantage principal de cette approche est l'élégance et la simplicité théorique ; cependant, sa dépendance sur un modèle de mouvement purement translationnel est une faiblesse, et la difficulté d'usage de modèle *a priori* pour raffiner la solution pose un écueil sérieux au raffinement des résultats.

Les recherches en super-résolution ont donné lieu à la proposition de plusieurs méthodes et techniques. Certaines constituent des approches différentes au problème [3, 38], tandis que d'autres font usage de méthodes reconnues de super-résolution modifiées pour application dans des cadres particuliers, comme les vi-

sages [39], les images satellites [40], les séquences vidéos [41–44, 46], ou la reconnaissance optique de caractères [45].

### 3.8 Modèles *a priori* étudiés

Cette section présente les différents modèles *a priori* étudiés dans le cadre de ce mémoire. Nous commençons par une description de deux modèles-témoins, le modèle Tikhonov et le modèle GGMRF. Nous présentons ensuite les trois modèles à l'étude, soient le modèle médian, un modèle basé sur une segmentation et un modèle basé sur un seuillage de la DCT.

#### 3.8.1 Régularisation Tikhonov

Un choix classique de modèle *a priori* [2], et l'un des plus simples, est de prendre  $\Omega(\mathbf{x}) = \|\mathbf{C}\mathbf{x}\|^2$ , ce qui correspond à la régularisation Tikhonov [13, 16]. Si  $\mathbf{C}$  est la matrice identité  $\mathbf{I}$ ,  $\Omega(\mathbf{x}) = \|\mathbf{x}\|^2$  pénalise les solutions présentant une forte variance de niveaux de gris. Ce terme présente le défaut d'introduire parfois des artefacts ("oscillations") dans la solution. Si  $C$  modélise une filtre passe-haut, par exemple un filtre laplacien, la présence de contours nets ou bien définis dans la solution sera pénalisée, et les images résultantes auront un aspect plus ou moins flou dépendant du poids accordé à  $\Omega(\mathbf{x})$ . Un poids élevé pour  $\Omega(\mathbf{x})$  promouvoit des changements graduels dans l'image au détriment des contours, tandis qu'un poids faible laisse dominer le terme de fidélité aux données au risque de produire une solution bruitée ou dégradée. Nous voudrions donc un  $\Omega(\mathbf{x})$  qui conserve les contours tout en atténuant les méfaits du bruit et des erreurs d'estimation de mouvement. Cet object est difficile à atteindre dans le cas général.

#### 3.8.2 Modèle *a priori* GGMRF

Plusieurs modèles ont été proposés afin de résoudre ce problème de l'accommodation simultanée de régions homogènes et de contours. L'un d'entre eux est le modèle gaussien généralisé (ou GGMRF, de l'anglais *Generalized Gaussian Mar-*

*kov Random Field*) proposé dans [37] et basé sur la notion de champ markovien. Nous commençons tout d'abord par présenter cette structure [47].

Soit  $S = \{s_1, s_2, \dots, s_N\}$  un ensemble de  $N$  sites et un système de voisinage  $V = \{v_s, s \in S\}$  définissant un ensemble de sous-ensembles de  $S$  répondant aux propriétés suivantes :  $\forall s, r \in S$ , 1)  $s \notin v_s$  et 2)  $s \in v_r \iff r \in v_s$ .  $v_s$  est appelé *voisinage* de  $s$  et les sites contenus dans  $v_s$  sont dits les *voisins* de  $s$ . Un sous-ensemble  $C \subseteq S$  est appelé une *clique* si  $C$  contient un seul site ou si  $\forall s, r \in C$ ,  $s$  et  $r$  sont voisins.

Dans le contexte d'une image digitale,  $S$  désigne l'ensemble des pixels de l'image, et en utilisant les coordonnées des pixels de la façon habituelle, nous pouvons décrire une famille de systèmes de voisinages qui présente un intérêt particulier comme suit :

$$V = \{(k, l) \in S \mid 0 < (k - i)^2 + (l - j)^2 \leq c\}. \quad (3.20)$$

Le cas où  $c = 1$  donne le système de voisinage de premier ordre, qui se limite aux voisins directement verticaux ou horizontaux d'un pixel : pour le pixel  $(i, j)$ , les quatre voisins seront  $(i - 1, j)$ ,  $(i, j - 1)$ ,  $(i, j + 1)$  et  $(i + 1, j)$ . Le cas  $c = 2$  est celui qui nous intéresse plus particulièrement. Il s'agit du voisinage de second ordre, qui ajoute au voisinage de premier ordre les voisins diagonaux :  $(i - 1, j - 1)$ ,  $(i - 1, j + 1)$ ,  $(i + 1, j - 1)$  et  $(i + 1, j + 1)$ . On peut voir les cliques possibles dans ces deux systèmes de voisinage à la figure 3.7. Le nombre de cliques augmente rapidement avec  $c$ , mais il s'avère [47] qu'un voisinage de second ordre utilisant seulement les cliques de deux pixels permet une flexibilité suffisante pour modéliser un éventail varié de comportements locaux.

Ces "comportements locaux" de l'image sont définis à l'aide d'une distribution, dite de Gibbs, qui s'exprime de la façon suivante :

$$\begin{aligned} P_X(\mathbf{x}) &= \frac{1}{Z} \exp -U(\mathbf{x}) \\ &= \frac{1}{Z} \exp \left\{ - \sum_{c \in C} \phi_c(\mathbf{x}) \right\} \end{aligned} \quad (3.21)$$

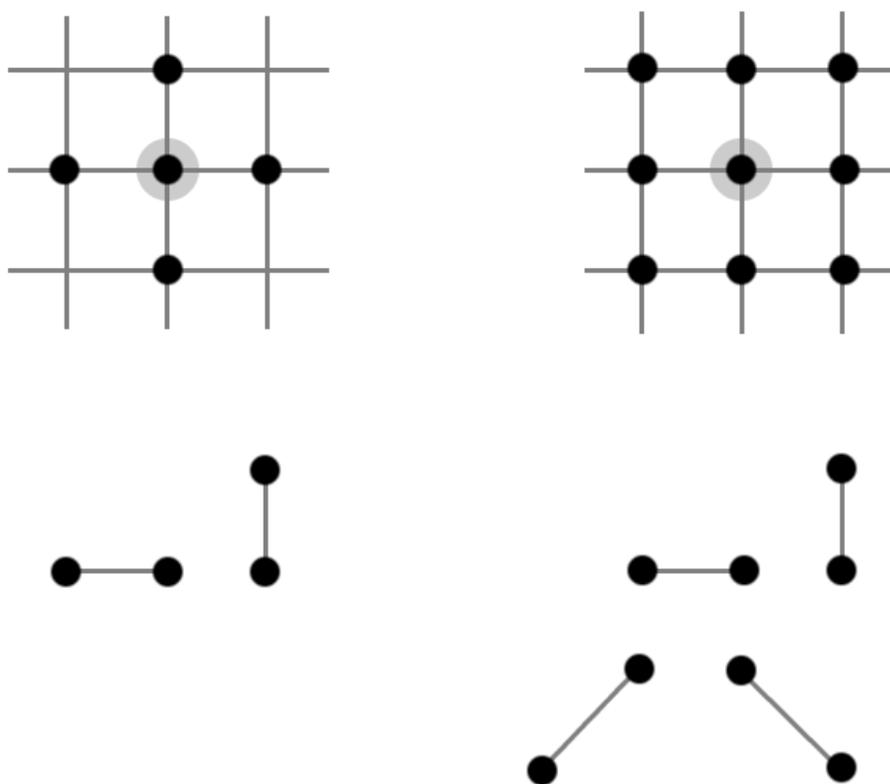


FIG. 3.7 – Structure des systèmes de voisinage d'ordres 1 et 2, et les cliques de deux sites possibles.

où  $U(\mathbf{x})$  est une fonction d'énergie,  $\phi_c(\mathbf{x})$  est une fonction de potentiel dont la valeur dépend de celle des pixels membres de la clique  $c$ ,  $C$  dénote l'ensemble des cliques de l'image et  $Z$  est une constante normalisante.

Cette formulation permet d'établir la propriété caractéristique des champs markoviens suivante :

$$\forall s \in S \ P(x_s | x_r, r \neq s) = P(x_s | x_r, r \in v_s), \quad (3.22)$$

c'est-à-dire que la distribution d'un pixel  $x_s$  dans l'image ne dépend que des pixels membres de son voisinage  $v_s$ . Cette propriété nous permet de modéliser des hypothèses sur la nature de  $\mathbf{x}$  à un niveau local et facilite la construction de  $\Omega(\mathbf{x})$  (qui serait autrement difficile à conceptualiser [47]).

La notion de champ markovien nous permet donc de modéliser la probabilité de rencontrer une réalisation de  $\mathbf{x}$  en pénalisant la valeur d'un pixel en fonction de son voisinage immédiat. Ce coût lié au voisinage est modélisé par la fonction  $U(x)$ , et sera plus ou moins élevé dépendant de la vraisemblance de ce voisinage. Il importe de choisir la fonction de potentiel en fonction de notre objectif de conservation concurrente des zones homogènes et des contours tel qu'énoncé plus haut. Or celle-ci est fréquemment choisie afin de pénaliser les différences marquées dans la valeur des pixels du voisinage [37], ce qui tend à pénaliser les contours et par le fait même est contraire à notre but. La contribution de [37] en cette matière est l'apport d'une fonction qui permet d'harmoniser ces deux préoccupations divergentes. Cette fonction prend la forme :

$$P_X(\mathbf{x}) \propto \exp \left\{ -\lambda^q \sum_{s,r \in v_s} \beta_{sr} |x_s - x_r|^q \right\} \quad (3.23)$$

où  $\beta_{sr} = (2\sqrt{2} + 4)^{-1}$  si  $x_s$  et  $x_r$  sont des voisins horizontaux/verticaux ou  $\beta_{sr} = (4 + 4\sqrt{2})^{-1}$  s'ils forment une clique diagonale, et  $1 \leq q \leq 2$ . Le paramètre  $q$  détermine l'attitude du modèle envers les contours : une valeur élevée les décourage tandis qu'une valeur faible (par exemple, de l'ordre de  $q = 1, 2$ ) les admet.

Le modèle *a priori* GGMRF, démontré dans [37], présente de bonnes performances dans la régularisation de la formulation MAP et mène donc à une solution unique dans le cadre d'une restauration d'image. Nous l'avons adapté pour les besoins de cette étude à un contexte de super-résolution et nous utiliserons les résultats comme point de comparaison avec ceux des modèles qui constituent le sujet de ce mémoire.

### 3.8.3 Modèle médian

Le modèle *a priori* médian proposé par Hsiao *et al.* [14] s'inspire d'un modèle antérieur suggéré par Alenius *et al.* [48]. Ce dernier a pour effet d'inciter les pixels de l'estimation  $\hat{\mathbf{x}}$  à graviter vers la médiane locale à chaque étape de la reconstruction, comportement qui permet de préserver les régions monotones de l'image sans en estomper les contours. Cependant, le caractère heuristique du modèle d'Alenius *et al.* rend difficile son analyse et impossible la formulation de  $\hat{\mathbf{x}}$  en tant que minimisation d'une fonction de coût tel qu'énoncé dans l'équation (3.18). L'apport de Hsiao *et al.* est la mise au point d'un modèle présentant les mêmes qualités et se prêtant bien à la minimisation. Cette section décrit ce modèle médian et expose le raisonnement motivant sa conception.

Le modèle *a priori* médian appartient à une classe de fonctions de coût qui pénalisent la divergence de  $\mathbf{x}$  par rapport à un vecteur auxiliaire  $\mathbf{m}$ , selon la relation :

$$\Omega(\mathbf{x}, \mathbf{m}) = \sum_j \sum_{j' \in \mathcal{N}(j)} \omega_{jj'} \phi(x_j - m_{j'}) \quad (3.24)$$

où  $\phi(\cdot)$  est une fonction de potentiel spécifiant le comportement de la fonction de coût et  $w_{jj'} \geq 0$  est un facteur exprimant le poids à accorder au potentiel en fonction de  $j$  et  $j'$ .  $\mathcal{N}(j)$  dénote l'ensemble de pixels formé par  $j$  et son système de voisinage d'ordre un, et  $j'$  réfère aux éléments situés aux coordonnées correspondantes dans  $\mathbf{m}$ . La particularité de cette classe de modèles *a priori* réside en ce que la fonction de potentiel  $\phi(\cdot)$  relate  $x_j$  non pas à ses voisins dans  $\mathbf{x}$  mais aux éléments  $m_{j'}$  dans le vecteur auxiliaire  $\mathbf{m}$ .

Pour expliquer le raisonnement derrière le modèle médian  $\Omega_{\text{méd}}(\mathbf{x}, \mathbf{m})$ , il est utile de définir en premier lieu un modèle connexe qui a la forme :

$$\Omega_{\text{abs}}(\mathbf{x}, \mathbf{m}) = \sum_j \sum_{j' \in \mathcal{N}(j)} |x_j - m_{j'}|. \quad (3.25)$$

Cette équation pose  $\phi(\cdot) = |\cdot|$  et  $\omega_{jj'} = 1 \forall j' \in \mathcal{N}(j)$ . L'usage de la valeur absolue pour  $\phi(\cdot)$  s'explique par l'équivalence suivante [49] :

$$c_M = \arg \min_{\zeta} \sum_i |c_i - \zeta|$$

où  $c_M$  est la valeur médiane de l'ensemble de valeurs  $c_i$ . Autrement dit, la minimisation par rapport à  $\zeta$  de la fonction de potentiel  $|c_i - \zeta|$  sur un ensemble de valeurs retourne la médiane de cet ensemble. Si l'on choisit  $\mathbf{m}$  de telle sorte que  $m_{j'}$  soit la médiane locale de l'ensemble  $\mathcal{N}(j)$ , l'équation (3.25) favorisera l'attraction de  $x_j$  vers cette médiane, ce qui est l'objectif déclaré du modèle *a priori* médian.

Dans ce contexte, nous pouvons reformuler la minimisation (3.18) sous la forme de l'estimation jointe suivante :

$$\hat{\mathbf{x}}, \hat{\mathbf{m}} = \arg \min_{\mathbf{x}, \mathbf{m}} \{ \|\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}\|^2 + \alpha \Omega_{\text{abs}}(\mathbf{x}, \mathbf{m}) \} \quad (3.26)$$

où  $\hat{\mathbf{m}}$  dénote l'estimation au sens MAP de  $\mathbf{m}$ . Cette minimisation s'opère de façon itérative comme suit :

$$\hat{\mathbf{x}}^{k+1} = \arg \min_{\mathbf{x}} \{ \|\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}\|^2 + \alpha \Omega_{\text{abs}}(\mathbf{x}, \hat{\mathbf{m}}^k) \}, \quad (3.27)$$

$$\hat{\mathbf{m}}^{k+1} = \arg \min_{\mathbf{m}} \{ \Omega_{\text{abs}}(\hat{\mathbf{x}}^{k+1}, \mathbf{m}) \}. \quad (3.28)$$

Tel que mentionné plus haut, nous désirons choisir  $\mathbf{m}$  de façon à ce que  $\hat{\mathbf{m}}$  corresponde à la médiane locale de  $\mathcal{N}(j)$ . Nous pouvons donc reformuler (3.27) et (3.28) ainsi :

$$\hat{\mathbf{x}}^{k+1} = \arg \min_{\mathbf{x}} \{ \|\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}\|^2 + \alpha \Omega_{\text{abs}}(\mathbf{x}, \hat{\mathbf{m}}^k) \}, \quad (3.29)$$

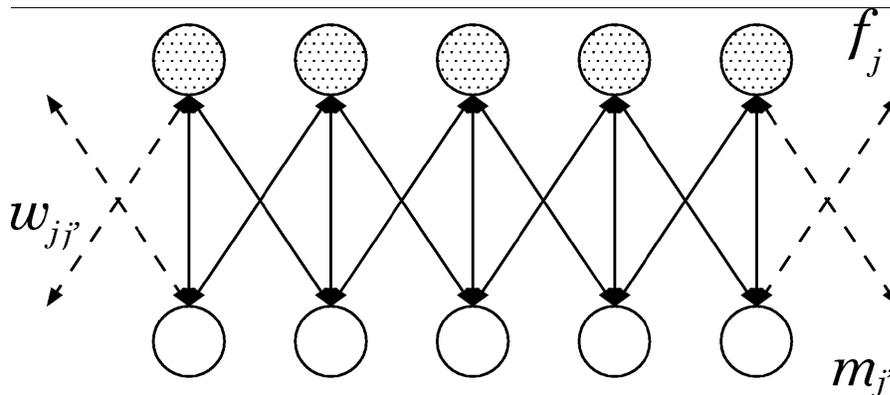


FIG. 3.8 – Illustration unidimensionnelle de la relation de voisinage entre  $f$  et le vecteur auxiliaire  $m$  (source : [14]).

$$\hat{\mathbf{m}}^{k+1} = \text{médiane} (\hat{x}_j^{k+1} | j \in \mathcal{N}(j')). \quad (3.30)$$

À remarquer la notation  $j \in \mathcal{N}(j')$ , qui désigne les pixels de  $x_j$  correspondant au voisinage de  $m_{j'}$  dans  $\mathbf{m}$  (voir figure 3.8). Cette itération d'estimations alternées de  $\hat{\mathbf{f}}$  et  $\hat{\mathbf{m}}$  provoque un comportement un peu ésoérique :  $\Omega_{\text{abs}}(\mathbf{x}, \mathbf{m})$  pousse  $\hat{\mathbf{x}}^{k+1}$  vers la médiane incarnée par  $\hat{\mathbf{m}}^k$ , mais  $\hat{\mathbf{m}}^k$  résulte elle-même d'un calcul de la médiane de  $\hat{\mathbf{x}}^k$ . Ce phénomène est interprété par Hsiao *et al.* comme la gravitation de  $\hat{\mathbf{x}}$  vers la médiane de ses propres médianes locales. Bien que nébuleux en théorie, ce mécanisme s'avère présenter les mêmes avantages (préservation des contours et des régions monotones) que le modèle médian d'Alenius *et al.* [14].

La fonction  $\Omega_{\text{abs}}(\mathbf{x}, \mathbf{m})$  telle que décrite en (3.25) souffre cependant du défaut de ne pas être dérivable en tout point, ce qui empêche sa minimisation par une méthode du gradient. Cet état de fait pousse à faire appel à une approximation de la valeur absolue :

$$\frac{1}{\eta} \log \cosh(\eta z) \approx |z| \quad (3.31)$$

où  $\eta$  est un paramètre contrôlant la fidélité de l'approximation (voir figure 3.9), qui croît avec des valeurs de  $\eta$  plus grandes. Nous pouvons maintenant présenter la fonction de coût du modèle médian incorporant cette approximation de la valeur

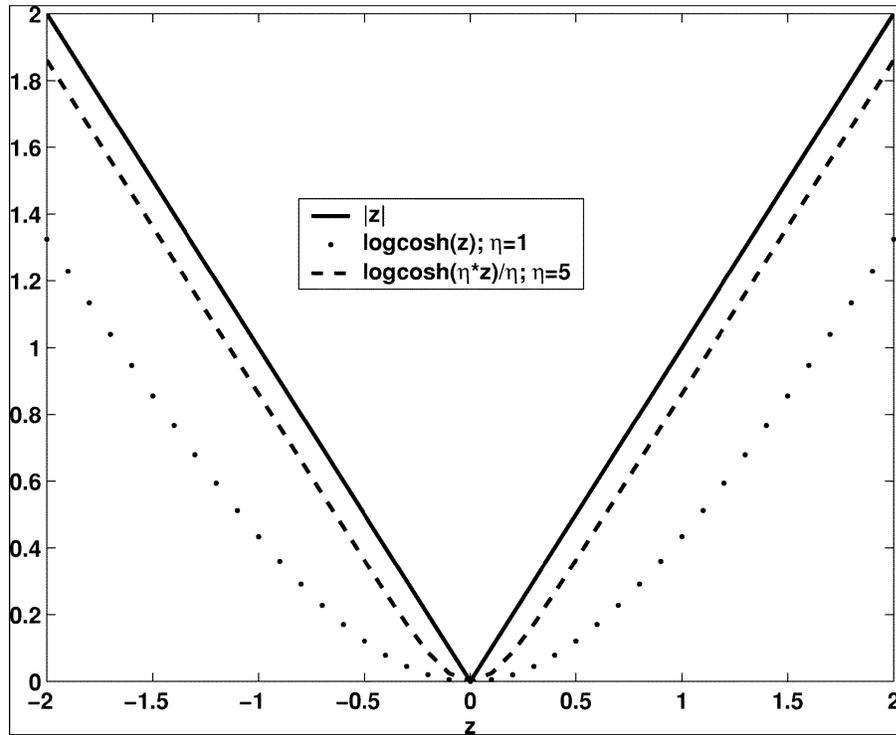


FIG. 3.9 – Illustration de l’approximation de la fonction de valeur absolue par la fonction  $\log \cosh(\cdot)$ , avec deux valeurs de  $\eta$  (source : [14]).

absolue :

$$\Omega_{\text{méd}}(\mathbf{x}, \mathbf{m}) = \frac{1}{\eta} \sum_{j=1}^N \sum_{j' \in \mathcal{N}(j)} \log \cosh(\eta(x_j - m_{j'})). \quad (3.32)$$

La minimisation itérative devient alors :

$$\hat{\mathbf{x}}^{k+1} = \arg \min_{\mathbf{x}} \{ \|\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}\|^2 + \alpha \Omega_{\text{méd}}(\mathbf{x}, \hat{\mathbf{m}}^k) \}, \quad (3.33)$$

$$\hat{\mathbf{m}}^{k+1} = \arg \min_{\mathbf{m}} \{ \Omega_{\text{méd}}(\hat{\mathbf{x}}^{k+1}, \mathbf{m}) \}. \quad (3.34)$$

### 3.8.4 Modèle basé sur une segmentation

Ce modèle, présenté dans [16], tire sa justification dans le constat suivant : alternativement à l’élaboration d’un modèle harmonisant les critères de conservation

concurrente des contours et des zones homogènes, approche empruntée par exemple par le modèle GGMRF cité plus haut, une méthode présentant les mêmes qualités peut être envisagée si, en présence d'une connaissance préalable de ces zones homogènes, on applique un simple critère de douceur à l'intérieur de celles-ci. L'affirmation est intuitive d'une part : la perte de netteté des contours, par exemple dans le cadre d'une restauration selon la régularisation de Tikhonov, résulte de la pénalisation de ceux-ci en faveur d'une solution où les régions à forte énergie sont adoucies. Si nous possédons une connaissance *a priori* des contours, nous pouvons nous garder de les éliminer dans la solution finale.

Le point crucial de cette approche réside dans notre capacité à estimer les contours dans l'image  $\mathbf{x}$  à partir de la série d'images basse-résolution  $\mathbf{y}_k$ . La tâche de diviser une image en régions de pixels apparentés selon un critère donné (en l'occurrence la valeur de l'intensité) est un problème bien connu en traitement d'images, que l'on désigne sous l'appellation de segmentation. Si une telle segmentation de  $\mathbf{x}$  est possible à partir des données disponibles, une méthode simple d'imposer une contrainte de douceur dans les régions homogènes consiste à pénaliser les solutions présentant une grande variance dans l'intensité des régions. Cette section présente ce modèle *a priori* sous les deux angles qui s'imposent : tout d'abord, la segmentation non-supervisée d'une estimation initiale de  $\mathbf{x}$  ; ensuite, l'utilisation de la segmentation afin de compartimenter l'image et imposer la contrainte de douceur à l'intérieur des régions qui en résultent.

#### 3.8.4.1 La segmentation markovienne non-supervisée

Tel que souligné plus haut, la segmentation de  $\mathbf{x}$  nous pose un problème certain :  $\mathbf{x}$  nous est inconnu ; qui plus est, nous recherchons une segmentation de  $\mathbf{x}$  précisément dans le but de recouvrer  $\mathbf{x}$ . Pour remédier à la situation, il nous faut commencer par générer une estimation initiale de  $\hat{\mathbf{x}}$ , que nous dénotons  $\hat{\mathbf{x}}_p$  (à ne pas confondre l'un avec l'autre) à l'aide de quelques itérations de super-résolution selon le modèle *a priori* GGMRF. Nous utilisons pour ce faire une valeur élevée de  $\alpha$  afin de générer une image largement débruitée. Cette solution préliminaire  $\hat{\mathbf{x}}_p$

servira de fondement pour la phase de segmentation décrite à la section suivante.

### 3.8.4.2 Formulation du problème

Nous commençons par formuler le problème de la segmentation de  $\hat{\mathbf{x}}_p$  comme suit. Soit une paire de champs aléatoires  $(X, Z)$  définis sur  $\hat{\mathbf{x}}_p$ .  $X$  dénote le champ des valeurs d'intensités observées (en l'occurrence les pixels de  $\hat{\mathbf{x}}_p$ ) mesurées selon les 256 niveaux de gris habituels, et  $Z$  représente le champ des étiquettes relié à l'image segmentée recherchée  $\mathbf{z}$  et prenant des valeurs dans l'ensemble  $\{e_0, e_1, \dots, e_{K-1}\}$  correspondant aux  $K$  classes de la segmentation. Notre objectif ici est donc l'estimation du champ  $Z$ , c'est-à-dire l'assignation à chacun des pixels de  $\hat{\mathbf{x}}_p$  d'une étiquette correspondant à l'une des  $K$  classes prescrites. Pour y parvenir, nous faisons appel à une approche bayésienne que nous décrivons ici. Rappelons la loi de Bayes dans le contexte présent :

$$P_{Z|X}(z|x) = \frac{P_{(X|Z)}(x|z)P_Z(z)}{P_X(x)}. \quad (3.35)$$

On démontre plus loin dans ce chapitre que la segmentation optimale de  $\hat{\mathbf{x}}_p$  sera celle qui maximise  $P_{Z|X}(z|x)$ . Or la vraisemblance  $P_{X|Z}(x|z)$ , qui décrit la variation de l'intensité à l'intérieur de chaque classe, peut être modélisée à l'aide d'une loi statistique choisie en fonction des données ; dans notre implantation, nous faisons appel à une distribution gaussienne  $\mathcal{N}(\mu, \sigma^2)$ . La distribution  $P_Z(z)$  de  $Z$ , quant à elle, modélise les propriétés escomptées du champ des étiquettes et est modélisée à l'aide d'une distribution de Gibbs. Nous revenons ci-après sur ces deux distributions en les déclinant plus formellement.

Tout d'abord, tel que mentionné au paragraphe précédent, la vraisemblance est modélisée sous la forme d'une distribution gaussienne, avec le formulation bien connue :

$$\mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right), \quad (3.36)$$

où  $\mu$  dénote la moyenne de la distribution et  $\sigma^2$  sa variance. On constate donc que

$P_{X|Z}(x|z)$  dépend des paramètres  $\mu$  et  $\sigma$  et qu'il faudra dans un premier temps arriver à estimer ceux-ci. Cette estimation se fait selon une méthode qui est l'objet de la section suivante.

Ensuite, nous désirons fixer la forme de  $P_Z(z)$  de façon à tirer avantage de la nature markovienne du problème de segmentation. Cette propriété de l'étiquetage nous permet de formuler une hypothèse sur la nature de  $Z$  qui favorisera certaines solutions présentant des propriétés désirables. La distribution *a priori*  $P_Z(z)$  que nous utilisons prend la forme :

$$P_Z(z) = \frac{1}{D} \exp \left( - \sum_{\langle s,t \rangle} \beta_{st} [1 - \delta(z_s, z_t)] \right) \quad (3.37)$$

où  $D$  est une constante de normalisation,  $\langle s, t \rangle$  dénote une clique du voisinage d'ordre 2 du pixel  $s$  dans  $\hat{\mathbf{z}}$ ,  $\delta(\cdot)$  est le delta de Kronecker et  $\beta_{st} = \beta_1, \beta_2, \beta_3$  et  $\beta_4$  dépendent de la nature de la clique, respectivement : horizontale, verticale, diagonale droite, ou diagonale gauche.

Dans ce contexte, la probabilité *a posteriori*  $P_{Z|X}(z|x)$  peut s'exprimer en tant que distribution de Gibbs du type  $P_{Z|X}(z|x) \propto \exp\{-U(z, x)\}$ . La fonction d'énergie à minimiser afin d'arriver à une estimation  $\hat{\mathbf{z}}$  de la segmentation idéale de  $\hat{\mathbf{x}}_p$  revêt alors la forme :

$$U(z, x) = \sum_{s \in S} \Psi_s(z_s, x_s) + \sum_{\langle s,t \rangle} \beta_{st} [1 - \delta(z_s, z_t)] \quad (3.38)$$

où  $\Psi_s(z_s, x_s) = -\ln P_{X_s|Z_s}(x_s, z_s)$ . Les deux termes de cette formulation correspondent respectivement au terme de fidélité aux données et au terme du modèle *a priori*, selon la même logique utilisée dans le contexte de la restauration super-résolution. Cette minimisation, et une description plus détaillée de sa justification, est traitée plus loin dans ce chapitre.

### 3.8.4.3 Estimation des paramètres : $\Phi$

La première étape de ce calcul est l'estimation du vecteur  $\Phi = [(\mu_0, \sigma_0), \dots, (\mu_{K-1}, \sigma_{K-1})]$  des paramètres de la représentation gaussienne de la vraisemblance pour chaque classe. Pour ce faire nous faisons appel à l'algorithme ECI (*estimation conditionnelle itérative*), qui permet de converger vers la meilleure estimation de  $\Phi$  au sens de l'erreur quadratique moyenne. Cet algorithme repose sur l'approche itérative suivante :

$$\hat{\Phi}^{p+1} = E^p[\hat{\Phi}(Z, X)|X = x] \quad (3.39)$$

$$= \int \hat{\Phi} P_{Z|X}(z|x) dz \quad (3.40)$$

où  $E^p$  dénote l'espérance conditionnelle telle qu'évaluée à l'aide du vecteur  $\hat{\Phi}^p$  obtenu à l'itération précédente et  $\hat{\Phi}(Z, X)$  est l'estimateur des données complètes. Cette approche se base sur le fait que la meilleure estimation de  $\Phi$  au sens de l'erreur quadratique moyenne correspond à l'espérance conditionnelle. Puisque cette dernière dépend du vecteur  $\Phi_{opt}$  des paramètres idéaux (inconnu et dans les faits impossible à évaluer avec exactitude), l'approche itérative est toute indiquée pour converger vers une bonne estimation. Le tableau 3.2 présente les étapes de la procédure.

L'algorithme ECI permet d'arriver à une estimation  $\hat{\Phi}$  du vecteur des paramètres de la distribution de chaque classe si nous disposons de trois éléments :

- une estimation initiale  $\hat{\Phi}^0$  suffisamment proche de  $\Phi$  ;
- un estimateur  $\hat{\Phi}(Z, X)$  des données complètes ;
- une méthode permettant de simuler des réalisations de  $Z$  selon la distribution de chaque classe et sachant  $X$ .

Ici, l'estimation initiale  $\hat{\Phi}^0$  est le résultat de l'algorithme des k-moyennes, décliné au tableau 3.3 [50]. Cet algorithme procède à la classification des pixels en posant  $K$  centres initiaux. Un pixel est assigné à une classe en fonction de la proximité au centre correspondant, et à la fin de chaque itération, les centres sont actualisés

TAB. 3.2 – Algorithme ECI

<p><b>Algorithme ECI</b></p> <p><b>Entrées :</b> une image <math>\mathbf{x}</math> et une estimation initiale <math>\hat{\Phi}^0</math> des paramètres</p> <p><b>Sorties :</b> une estimation <math>\hat{\Phi}</math> des paramètres</p> <p>– Itérer jusqu'à ce que <math>\hat{\Phi}^{p+1} \approx \hat{\Phi}^p</math> :</p> <ul style="list-style-type: none"> <li>- Simuler <math>n</math> réalisations <math>(z_{(1)}, z_{(2)}, \dots, z_{(n)})</math> de <math>Z</math> de la distribution <i>a posteriori</i> <math>P_{Z X}(z x, \hat{\Phi}^p)</math> à l'aide de l'échantillonneur de Gibbs et considérant <math>\hat{\Phi}^p</math>.</li> <li>- Calculer <math>\hat{\Phi}^{p+1}</math> à l'aide des estimateurs MV appliqués sur les réalisations simulées à l'étape précédente selon la relation :</li> </ul> $\hat{\Phi}^{p+1} = \frac{1}{n}[\hat{\Phi}(z_{(1)}, x) + \dots + \hat{\Phi}(z_{(n)}, x)] \quad (3.41)$
--

afin de représenter le centre gravitationnel des pixels associés.

Les estimateurs MV des données complètes sont, dans le cas d'une distribution gaussienne :

$$\hat{\mu}_{MV} = \frac{1}{N} \sum_{i=1}^N x_i \quad (3.45)$$

et

$$\hat{\sigma}_{MV}^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \hat{\mu}_{MV})^2 \quad (3.46)$$

c'est-à-dire les estimateurs empiriques calculés à partir des données.

Finalement, nous pouvons simuler des réalisations de  $Z$  à l'aide de méthodes itératives de la famille de l'échantillonneur de Gibbs [47].

Ainsi donc, la procédure détaillée dans cette section permet d'arriver à une estimation  $\hat{\Phi}$  du vecteur des paramètres que nous espérons proche de  $\Phi$ . Nous utiliserons cette information pour procéder à la classification des pixels de  $\hat{\mathbf{x}}_p$  dans la section suivante.

TAB. 3.3 – Algorithme des k-moyennes

<p><b>Algorithme k-moyennes</b>  <b>Entrées :</b> une image <math>\mathbf{x}</math>  <b>Sorties :</b> une estimation des paramètres des <math>K</math> classes</p>
<p>1. Choisir <math>K</math> pixels qui serviront de centres <math>(c_0^0, \dots, c_{K-1}^0)</math> des <math>K</math> groupements :</p> $c_i^0 = x_i \quad 0 \leq i \leq K - 1 \quad (3.42)$ <p>2. À l'itération <math>m</math>, on associe chaque pixel <math>x_i</math> de <math>\mathbf{x}</math> au groupement <math>C_i^m</math> dont le centre lui est le plus proche :</p> $\ x_i - c_i^m\  < \ x_i - c_j^m\  \quad \forall j \neq i \quad (3.43)$ <p>3. À la fin de l'itération <math>m</math>, on réévalue le centre de chaque groupement :</p> $c_i^{m+1} = \frac{1}{N_i} \sum_{x \in C_i^m} x \quad \forall i \quad (3.44)$ <p>où <math>N_i</math> dénote le nombre d'échantillons dans le groupement <math>C_i^m</math>.</p> <p>4. Si les centres sont demeurés les mêmes entre l'itération <math>m</math> et <math>m + 1</math>, arrêter, sinon retourner à l'étape 2.</p>

### 3.8.4.4 Segmentation markovienne

La segmentation bayésienne considère comme critère d'estimation le critère de minimisation de l'espérance d'une fonction de coût comme suit :

$$\hat{Z} = \arg \min_z E[C(Z, z)|X = x] \quad (3.47)$$

où  $Z$  dénote un vecteur aléatoire représentant la solution idéale et  $C(Z, z)$  associe un coût à la classification  $z$ .

L'estimateur MAP est associé à la fonction de coût :

$$C(Z, z) = 1 - \delta_Z(z) \quad (3.48)$$

où  $\delta_Z(z)$  est la masse de Dirac en  $Z$ . Cette fonction impose le même coût (1) à toutes les configurations  $z$  différentes de  $Z$ . On peut donc reformuler (3.47) comme suit :

$$\hat{z}_{MAP} = \arg \min_z \left\{ \sum_{z' \in E} C(Z, z') \cdot P_{Z|X}(z'|x) \right\} \quad (3.49)$$

$$= \arg \min_z \{1 - P_{Z|X}(z|x)\} \quad (3.50)$$

$$= \arg \max_z \{P_{Z|X}(z|x)\} \quad (3.51)$$

où  $E$  dénote l'ensemble des classifications possibles. L'estimation MAP revient donc à déterminer  $\hat{z}$  qui maximise  $P_{Z|X}(z|x)$ . Cette opération s'avère cependant impraticable en raison de la taille de l'espace de recherche (une image de dimensions  $N^2$  requiert le calcul de  $2^{N^2}$  configurations différentes). C'est ici que le caractère bayésien de notre approche montre son utilité, en permettant la reformulation :

$$\hat{z}_{MAP} = \arg \max_z \{P_Z(z) \cdot P_{X|Z}(x|z)\}. \quad (3.52)$$

En dénotant par  $U(z)$  la somme des potentiels locaux sur les cliques d'ordre 2, on pose  $P_Z(z) = \frac{1}{D} \exp(-U(z))$  où  $D$  est une constante de normalisation. Nous

TAB. 3.4 – Algorithme ICM

<p><b>Algorithme ICM</b></p> <p><b>Entrées :</b> une image <math>\mathbf{x}</math> et une estimation <math>\hat{\Phi}</math> des paramètres</p> <p><b>Sorties :</b> une segmentation <math>\hat{\mathbf{z}}</math></p>
<p>– Initialisation de <math>\hat{z}^0</math> à une valeur proche de la solution optimale, en l'occurrence la segmentation au sens MV :</p> $\hat{z}^0 = \arg \max \{P_{X Z}(x z)\} \quad (3.55)$ <p>– Itérer jusqu'à ce que <math>\hat{z}^{k+1} \approx \hat{z}^k</math> :</p> <p>– En balayant chaque site <math>s</math> selon une stratégie de visite de sites :</p> <p>- On calcule la probabilité d'appartenir à chaque classe selon la probabilité :</p> $P_{Z_s Z_{\nu_s}, X_s}(z_s \nu_s, x_s) = \frac{1}{D_s} \exp \left( - \left( \sum_{c \in C} V_c(z_s) + V(z_s, x_s) \right) \right) \forall z_s \in \Omega \quad (3.56)$ <p>- On sélectionne la classe <math>z_s</math> qui correspond à la probabilité maximale :</p> $\hat{z}_s = \arg \max_{z_s \in \Omega} \{P_{Z_s Z_{\nu_s}, X_s}(z_s \nu_s, x_s)\} \quad (3.57)$

arrivons alors à la minimisation mentionnée en (3.38) :

$$\hat{z}_{MAP} = \arg \max_z \left\{ \frac{1}{D} \exp - (U(z) + V(z, x)) \right\} \quad (3.53)$$

$$= \arg \min_z \{U(z) + V(z, x)\} \quad (3.54)$$

où  $V(z, x) = -\ln P_{X|Z}(x|z)$  et  $U(z) = \sum_{\langle s,t \rangle} \beta_{st}(1 - \delta(z_s, z_t))$ .

La minimisation de (3.54) peut se faire de plusieurs méthodes. Nous faisons ici usage de l'algorithme ICM (*iterated conditional modes*), décrit au tableau 3.4. Cet algorithme est une méthode déterministe permettant d'obtenir de bons résultats de segmentation lorsqu'un estimé initial proche de la solution finale est disponible.

### 3.8.4.5 Partition en régions

La segmentation  $\hat{\mathbf{z}}$  ainsi obtenue est subséquentement partitionnée en régions de 100 pixels ou moins afin d'estomper l'effet de transition abrupte entre les régions

lors de l'application du modèle *a priori*, qui impose autrement une contrainte de douceur au sein de celles-ci.

Le caractère stochastique de l'algorithme ECI permet la génération de plusieurs segmentations différentes pour une image donnée, et nous prenons avantage de cette opportunité dans la conception du terme de pénalité décrit à la prochaine section.

#### 3.8.4.6 Terme de pénalité

La fonction de coût associé au modèle basé sur une segmentation à l'étude ici prend la forme suivante :

$$\Omega(\mathbf{x}_s) = \|\rho(\mathbf{x}_s)\|^2 = \|\mathbf{x}_s - \frac{1}{N_{seg}} \sum_{g=1}^{N_{seg}} \Gamma_g(\mathbf{x}_s)\|^2 \quad (3.58)$$

où  $N_{seg}$  dénote le nombre de segmentations effectuées et l'opérateur  $\Gamma_g(\mathbf{x})$  retourne pour un pixel donné l'intensité moyenne de la région auquel celui-ci appartient dans la  $g$ -ième segmentation. Il est désirable de se baser sur plus d'une segmentation pour atténuer l'impact sur le modèle d'une mauvaise segmentation. Si on assigne à chaque région d'une segmentation l'intensité moyenne des pixels qu'elle contient, on peut imaginer le second terme du côté droit de l'équation (3.58) comme la moyenne des segmentations, et il apparaît clairement que l'effet de la fonction de coût est de pénaliser la déviation de  $\hat{\mathbf{x}}$  de celle-ci.

#### 3.8.5 Modèle de débruitage par seuillage de coefficients de DCT

Le dernier modèle [51] étudié dans le présent mémoire se base sur la notion de décomposition d'un signal, en l'occurrence une image, en composantes distinctes. Un exemple bien connu de cette opération est la transformée de Fourier, qui permet de représenter un signal périodique sous la forme d'une somme de fonctions sinus et cosinus selon des amplitudes et fréquences variées [1]. D'autres transformées existent et sont utilisées en traitement d'images, comme par exemple la transformée en cosinus discrète (DCT), la transformée en ondelettes, etc.

Cette approche permet d’aborder l’image sous un angle différent : en effet, l’étude de l’image par le biais des coefficients obtenus après transformation nous permet d’effectuer des opérations qui seraient beaucoup plus complexes si on tentait de les appliquer directement aux pixels de l’image. On peut retrouver une foule d’exemples des mérites de cette démarche dans la littérature sur les applications de la transformée de Fourier discrète : celles-ci ont démontré leur efficacité pour le débruitage, la détection de contours, et plus généralement tout filtrage appliqué dans le domaine fréquentiel. Pour une discussion plus approfondie de ces techniques nous pouvons nous référer à la théorie présentée dans [1].

Le modèle qui nous intéresse ici s’inscrit dans la lignée des techniques de débruitage par application de seuillage des coefficients de la transformée [52–56]. La présente section commence par une présentation de la transformée en cosinus discrète, que nous utilisons dans l’implantation de ce modèle, et se poursuit avec une description du cadre et des méthodes liées à ce type de débruitage. Nous achevons cette section sur une description de l’adaptation de cette approche à notre modèle de super-résolution.

### 3.8.5.1 La DCT

La transformée en cosinus discrète est l’une des transformées les plus communément rencontrées en traitement d’images, notamment en raison de son utilisation importante en compression d’image [1]. La norme de compression utilisée dans les formats JPEG et MPEG en sont l’application la plus omniprésente.

Soit une image  $f(i, j)$  de taille  $N \times N$ , la transformée en cosinus discrète  $T(u, v)$  peut être exprimée comme suit :

$$T(u, v) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i, j)g(i, j, u, v) \quad (3.59)$$

où

$$g(i, j, u, v) = \alpha(u)\alpha(v) \cos \left[ \frac{(2i+1)u\pi}{2N} \right] \cos \left[ \frac{(2j+1)v\pi}{2N} \right] \quad (3.60)$$

avec

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{pour } u = 0 \\ \sqrt{\frac{2}{N}} & \text{pour } u = 1, 2, \dots, N-1 \end{cases}$$

et avec  $\alpha(v)$  défini similairement. La transformée inverse est quant à elle obtenue selon la formule :

$$f(i, j) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} T(u, v) h(i, j, u, v) \quad (3.61)$$

et il s'avère que dans le cas de la DCT,  $g(i, j, u, v) = h(i, j, u, v)$ . On constate que  $g(i, j, u, v)$  et  $h(i, j, u, v)$  ne dépendent pas de l'image  $f(i, j)$  ou de sa transformée  $T(u, v)$ , et il est donc possible d'en faire le calcul au préalable pour une valeur de  $N$  donnée. La figure 3.10 présente l'ensemble des matrices de base correspondant à  $N = 8$ .

Il est important de noter qu'en pratique les images étudiées auront une dimension de beaucoup supérieure à  $N \times N$  : le signal sera alors divisé en fenêtres ou blocs de taille  $N \times N$  (une valeur de  $N = 8$  tel que suggéré précédemment étant une valeur typique) et traité séquentiellement (voir figure 3.11).

### 3.8.5.2 Cadre et motivation

Soit une matrice  $x$  de dimension  $N \times N$  représentant un bloc dans l'image à reconstruire et  $n$  la matrice du bruit correspondant, nous dénotons alors par  $y = x + n$  la matrice du signal bruité. Posons  $\mathbf{H}$  un opérateur matriciel qui effectue une transformée vers un domaine orthogonal [52], et  $\mathbf{H}^{-1}$ , l'opérateur inverse. En l'occurrence, la transformée en cosinus discrète sera utilisée.

La technique classique de débruitage par seuillage de coefficients se décline comme suit :

$$c = \mathbf{H}y \quad (3.62)$$

$$\hat{c} = \mathcal{T}(c) \quad (3.63)$$

$$\hat{x} = \mathbf{H}^{-1}\hat{c} \quad (3.64)$$

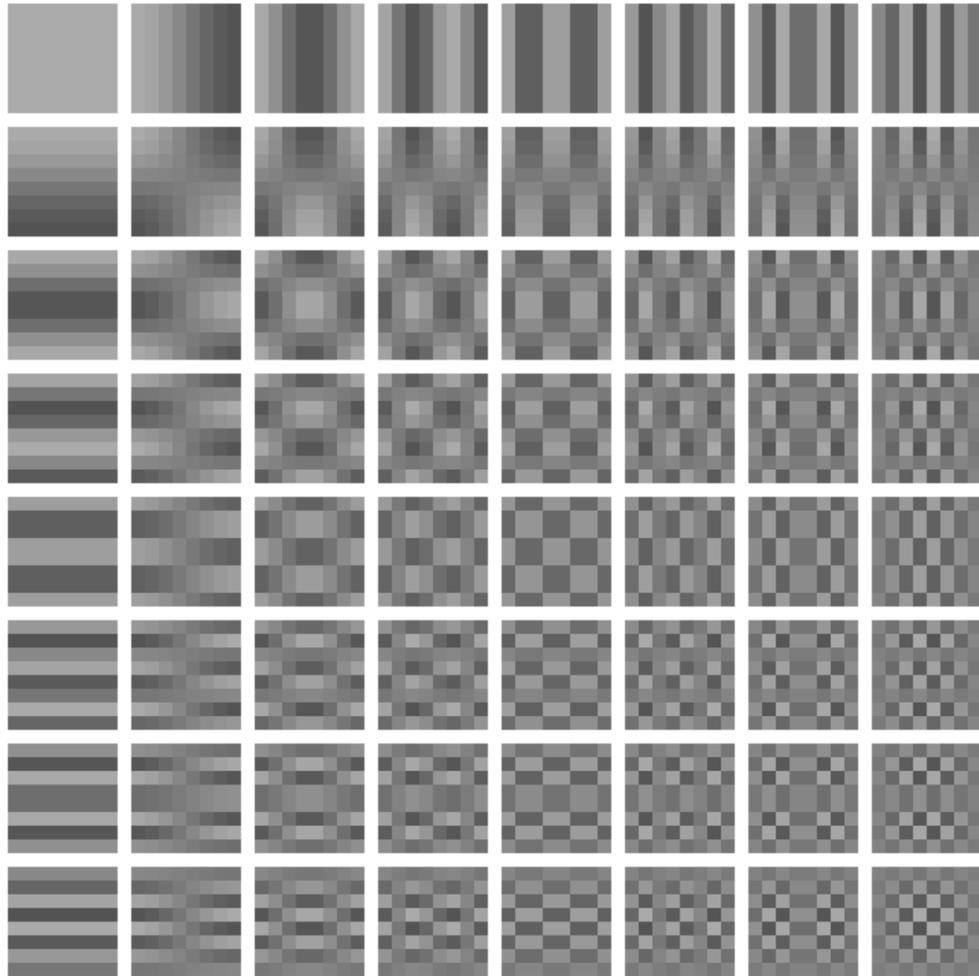


FIG. 3.10 – Les 64 matrices de base de la transformée en cosinus discrète.

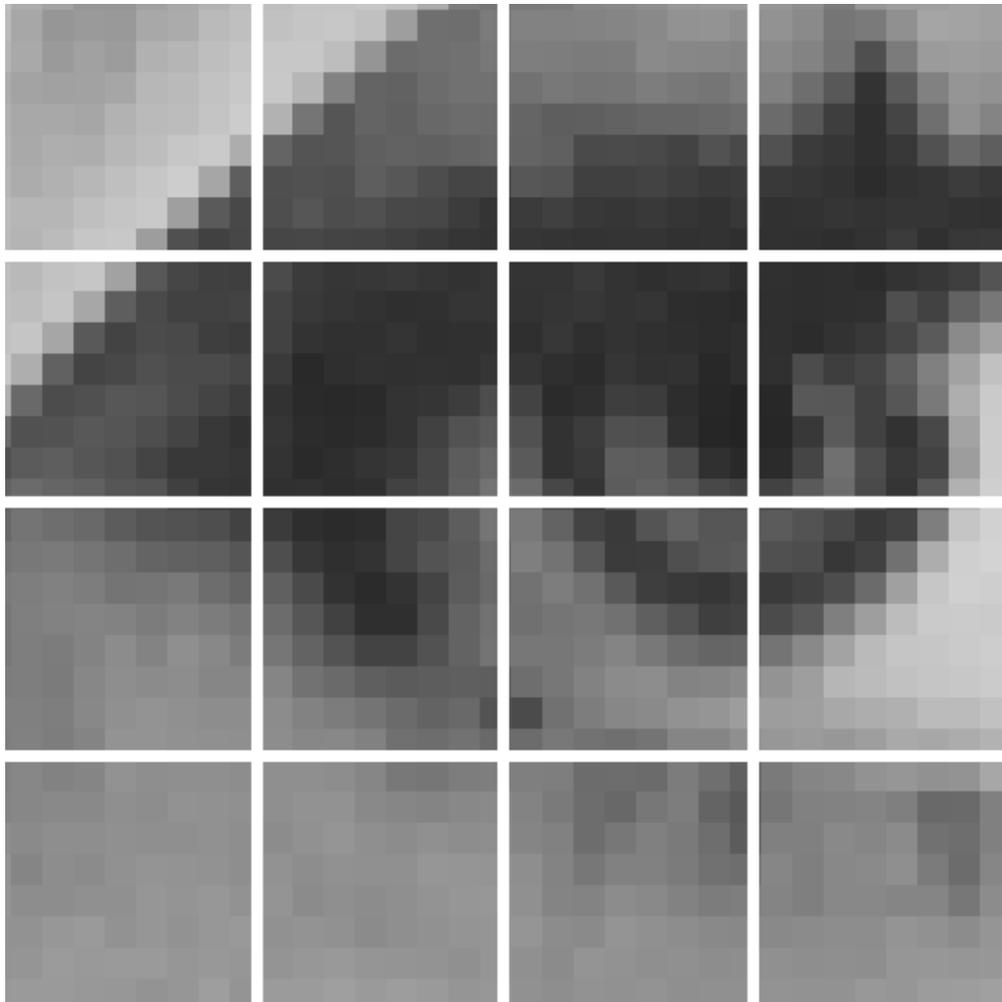


FIG. 3.11 – Carrelage d'une imagerie  $32 \times 32$  en 16 matrices  $8 \times 8$ .

où  $c$  dénote la matrice des coefficients de la transformée,  $\hat{c}$  la matrice des coefficients après seuillage,  $\mathcal{T}(\cdot)$  la fonction de seuillage et  $\hat{x}$  la matrice du signal débruité. La restauration dépend ici du seuillage des coefficients dans le domaine de la transformée, et celui-ci est opéré à la lumière d'un constat : les coefficients dont la valeur est faible représentent essentiellement du bruit (dans un modèle où l'image a des propriétés de sparsité, c'est-à-dire lorsque celle-ci peut être représentée par un nombre peu élevé de coefficients DCT) et peuvent être ramenés à zéro, résultant en un débruitage du signal original. Une fonction de seuillage typique (ici, une fonction à seuil fixe) prendra alors la forme :

$$\hat{c}(k) = \begin{cases} c(k) & |c(k)| > S \\ 0 & |c(k)| \leq S \end{cases} \quad (3.65)$$

où  $c(k)$  dénote le  $k$ -ième coefficient,  $\hat{c}(k)$  le  $k$ -ième coefficient après seuillage, et  $S$  la valeur du seuil utilisée pour l'opération.

La technique présentée au paragraphe ci-haut a démontré son efficacité dans la pratique. Cependant, les travaux dans [52] dirigent l'attention vers un phénomène d'artefacts qui en contamine parfois les résultats. Ces artefacts s'expliquent de plusieurs façons dépendant de la nature de la transformée et du seuillage appliqué. Par exemple, le débruitage par seuillage d'ondelettes peut engendrer des artefacts au niveau des discontinuités du signal, dépendant également en partie de la position de la discontinuité dans l'élément, tandis que le seuillage des coefficients de la DCT entraîne l'apparition d'artefacts aux bords des éléments du pavage du signal (c'est-à-dire les subdivisions en fenêtres  $N \times N$  tel que décrit plus haut). Pour remédier à ce défaut, une amélioration de la technique a été suggérée dans [52] qui consiste en l'application du débruitage par seuillage de coefficients obtenus selon plusieurs transformations selon des grilles décalées. Autrement dit, plusieurs reconstructions sont générées et combinées en faisant la moyenne, dans l'espoir que pour chaque pixel du signal, la moyenne des reconstructions présentera une valeur restaurée.

Concrètement :

$$\hat{x} = \frac{1}{M} \sum_{i=1}^M \hat{x}_i, \quad (3.66)$$

où  $M$  dénote le nombre de reconstructions.

L'innovation proposée dans [51] consiste en un raffinement de cette dernière méthode. L'amélioration proposée implique un choix plus judicieux du poids accordé à chacune des  $M$  reconstructions : au lieu d'une pondération uniforme résultant en une simple moyenne, nous recherchons une fonction de poids plus appropriée aux données. L'équation (3.66) devient alors :

$$\hat{x} = \sum_{i=1}^M \gamma_i \hat{x}_i, \quad (3.67)$$

avec  $\gamma_i$  le poids accordé à la reconstruction  $\hat{x}_i$ .

La motivation derrière cette décision est illustrée à la figure 3.12. On y constate que les blocs des carrelages DCT1 et DCT2 qui correspondent au site  $n_e$  du signal recouvrent également une discontinuité. Intuitivement, nous pouvons en déduire que ces blocs produiront un seuillage moins épars que ceux des pavages DCT3 et DCT4, et qu'il serait avantageux de donner une importance plus grande aux reconstructions de ces deux dernières.

### 3.8.5.3 Dérivation

Nous supposons par souci de simplicité un signal 2D (c'est-à-dire un vecteur) prenant des valeurs centrées sur zéro.

Les poids  $\gamma_i$  dans l'équation (3.66) doivent être choisis afin de minimiser la valeur attendue de l'erreur moyenne carrée de  $\hat{x}$  par rapport à  $x$  en fonction de  $y$  :

$$E[|x - \hat{x}|^2 | y]. \quad (3.68)$$

Nous supposons tout d'abord que chaque reconstruction  $\hat{x}_i$  diffère de  $x$  princi-

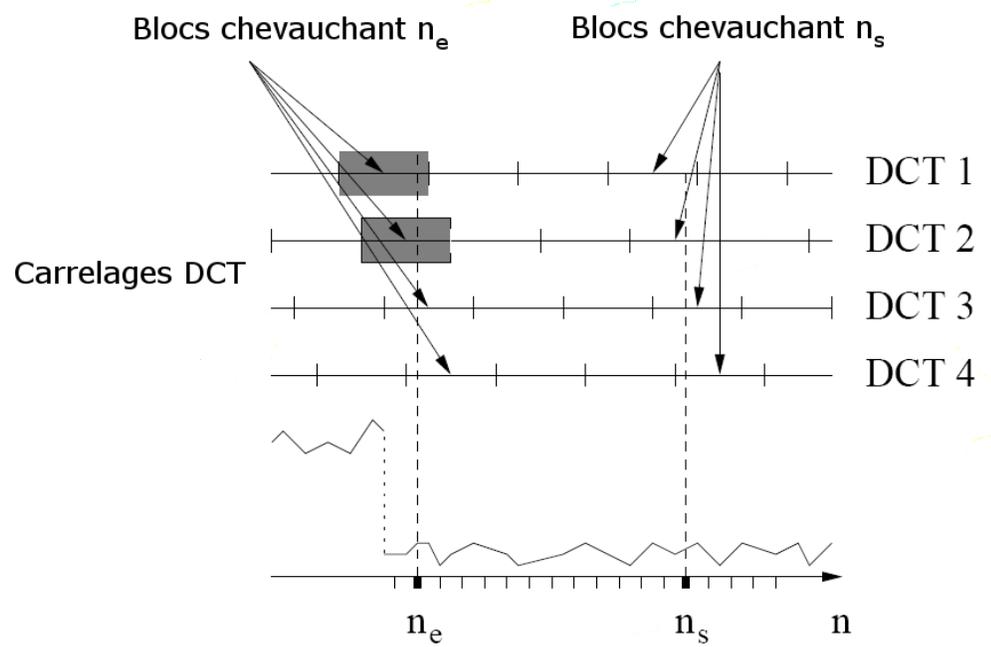


FIG. 3.12 – Schéma illustrant l'intuition derrière le modèle basé sur un seuillage de la DCT. Les blocs des transformées DCT3 et DCT4 qui chevauchent  $n_e$  présenteront une sparsité plus favorable que les blocs correspondants dans DCT1 et DCT2.

palement par le bruit, dénoté  $\gamma_i$  :

$$\hat{x}_i \approx x + \omega_i. \quad (3.69)$$

De plus, nous définissons l'ensemble d'index  $\mathcal{V}_i$  comme l'ensemble des sites dans  $c_i$  qui ne sont pas ramenés à 0 lors du seuillage :

$$\mathcal{V}_i = \{k \mid |c_i(k)| > \mathcal{T}\}. \quad (3.70)$$

Nous émettons alors l'hypothèse que l'équation (3.68) est équivalente à :

$$E[||x - \hat{x}||^2 | \mathcal{V}_1, \dots, \mathcal{V}_M], \quad (3.71)$$

c'est-à-dire que la valeur attendue de  $|x - \hat{x}|$  dépend exclusivement du degré de sparsité des différents seuillages. Nous reformulons donc le problème sous la forme suivante :

$$E[||x - \sum_{i=1}^M \gamma_i \hat{x}_i||^2 | \mathcal{V}_1, \dots, \mathcal{V}_M]. \quad (3.72)$$

En dénotant par  $\mathcal{E}[\dots]$  la valeur conditionnelle  $E[\dots | \mathcal{V}_1, \dots, \mathcal{V}_M]$  et en définissant  $\Gamma_n = [\gamma_1, \dots, \gamma_M]^T$  et  $\chi_n = [\hat{x}_1, \dots, \hat{x}_M]^T$ , l'équation (3.72) devient :

$$\min_{\Gamma_n} \mathcal{E}[|x - \Gamma_n^T \chi_n|^2]. \quad (3.73)$$

Le bruit de l'image étant indépendant, nous avons :

$$\mathcal{E}[\omega_i x] = 0, \quad (3.74)$$

et grâce à (3.69), nous pouvons déduire :

$$\mathcal{E}[\hat{x}_i x] \cong \mathcal{E}[x^2], \text{ et } \mathcal{E}[\hat{x}_i \hat{x}_j] \cong \mathcal{E}[x^2] + \mathcal{E}[\omega_i \omega_j]. \quad (3.75)$$

En dérivant (3.73) par rapport aux composantes de  $\Gamma_n$ , nous obtenons :

$$\sigma_x^2 \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \left( \sigma_x^2 \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{bmatrix} + \mathbf{W}_n \right) \Gamma_n, \quad (3.76)$$

où  $\sigma_x^2 = \mathcal{E}[|x|^2]$  et la matrice  $M \times M$   $\mathbf{W}_n$  est composée des éléments  $\mathbf{W}_n(i, j) = \mathcal{E}[\omega_i \omega_j]$  pour  $i, j = 1, \dots, M$ .

En dénotant par  $u_n$  un vecteur dont les valeurs sont des 0 partout sauf à l'index  $n$ , qui a une valeur de 1, nous pouvons exprimer  $\omega_i$  et  $\mathbf{W}_n$  comme suit :

$$\omega_i = \mathbf{H}_i^T \mathbf{S}_i \mathbf{H}_i \omega, \quad (3.77)$$

et

$$\mathbf{W}_n(i, j) = \mathcal{E}[\omega_i \omega_j] \quad (3.78)$$

$$= u_n^T \mathbf{H}_i^T \mathbf{S}_i \mathbf{H}_i \mathcal{E}[\omega \omega^T] \mathbf{H}_j^T \mathbf{S}_j \mathbf{H}_j u_n \quad (3.79)$$

$$= \sigma_\omega^2 u_n^T \mathbf{H}_i^T \mathbf{S}_i \mathbf{H}_i \mathbf{H}_j^T \mathbf{S}_j \mathbf{H}_j u_n \quad (3.80)$$

$$= \sigma_\omega^2 \mathbf{G}_n(i, j) \quad (3.81)$$

où  $\sigma_\omega^2$  représente la variance du bruit et  $\mathbf{G}_n$  est une matrice  $M \times M$  qui dépend uniquement de l'ensemble de transformées et l'ensemble d'index  $\mathcal{V}_i$ .

Cette dernière équation nous permet de reformuler (3.76) sous la forme :

$$\lambda_n \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \mathbf{G}_n \Gamma_n, \quad (3.82)$$

où  $\lambda_n$  est un facteur d'échelle. En posant  $\sigma_x^2 \gg \sigma_\omega^2$ , nous obtenons  $\sum_i \gamma_i \approx 1$ , ce qui nous permet d'omettre les paramètres statistiques et fixe  $\lambda_n$  à une valeur que nous dénotons  $\lambda_n^*$ .

### 3.8.5.4 Solution

Plusieurs solutions sont proposées dans l'article de Guleryuz [51], dépendant des simplifications utilisées dans la résolution. La solution présentée ici est celle préconisée par Guleryuz et celle que nous utilisons dans ce mémoire.

Nous supposons que les fonctions de base de la transformée ont une valeur constante dans leur support spatial. Considérons la  $m$ -ième fonction de base de la  $i$ -ème transformée que l'on retrouve à la  $m$ -ième rangée de  $\mathbf{H}_i$ , notée  $\mathbf{H}_i(m, n)$ . Soit  $\nu_i$  le nombre de composantes non-nulles dans  $\mathbf{H}_i(m, \cdot)$ . Pour cette valeur de  $m$  et pour  $n = 1, \dots, N$ , nous définissons :

$$\mathbf{F}_i(m, n) = \begin{cases} \frac{1}{\sqrt{\nu_i(m)}} & \text{si } \mathbf{H}_i(m, n) \neq 0 \\ 0 & \text{autrement.} \end{cases}$$

Dans le cas de transformée par blocs comme la DCT, les fonctions de base de la transformée ont un support en bloc. En supposant que toutes les transformées dans l'ensemble de pavages ont la même taille de bloc,  $\nu_i(m)$  devient indépendant de  $i$  et  $m$ , c'est-à-dire que  $\nu_i(m) = \nu$ . Puisque les transformées prennent la forme de blocs, chaque pixel de l'image est recouvert par un seul bloc de chaque transformée. Considérons le pixel  $n$  du bloc  $b_{n,i}$  de la  $i$ -ème transformée, qui après seuillage de la  $i$ -ème transformée comporte  $\mathcal{N}(b_{n,i})$  coefficients non-nuls. Nous avons alors :

$$u_n^T \mathbf{F}_i^T \mathbf{S}_i \mathbf{F}_i u_n = \sum_{k \in \mathcal{V}_i} (\mathbf{F}_i(k, n))^2 \quad (3.83)$$

$$= \sum_{k \in \{\text{coefficients non-nuls dans } b_{n,i}\}} \frac{1}{\nu} \quad (3.84)$$

$$= \frac{\mathcal{N}(b_{n,i})}{\nu}. \quad (3.85)$$

Nous obtenons donc au pixel  $n$  le poids de l'estimation restaurée de la  $i$ -ème transformée :

$$\gamma_i = \frac{C_n}{\mathcal{N}(b_{n,i})}, \quad (3.86)$$

où  $C_n$  est une constante indépendante de  $i$  choisie afin que  $\sum_i \gamma_i = 1$ . Il apparaît donc que cette solution implique des poids  $\gamma_i$  qui dépendent uniquement de la sparsité des blocs transformés dans les estimations restaurées.

### 3.9 Stratégie de minimisation et implantation

Nous avons présenté cinq modèles *a priori* pouvant nous aider à procéder à la super-résolution. Il nous reste maintenant à énoncer une méthode clairement définie pour atteindre l'objectif ultime d'une estimation  $\hat{\mathbf{x}}$ . La minimisation (3.18) peut s'effectuer selon plusieurs techniques d'optimisation ; dans notre implantation, nous faisons appel à la méthode de descente du gradient, qui recherche le minimum de la fonction de manière itérative en progressant dans le sens du gradient négatif :

$$\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma \Delta E(\mathbf{x}) \quad (3.87)$$

où  $E(\mathbf{x}) = \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 + \alpha\Omega(\mathbf{x})$  et  $\gamma$  dénote la grandeur du pas. La procédure itérative peut donc être développée comme suit :

$$\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma [\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n) - \alpha\Omega'(\hat{\mathbf{x}}^n)]. \quad (3.88)$$

L'algorithme 3.5 formalise le processus par lequel nous procédons à la super-résolution dans ce mémoire.

La minimisation de Tikhonov s'effectue donc comme suit :

$$\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma [\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n) - \alpha\mathbf{C}^T\mathbf{C}\hat{\mathbf{x}}^n], \quad (3.89)$$

et la minimisation GGMRF selon la formule :

$$\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma \left[ \mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n) - \alpha q \lambda^q \sum_{s,r} \beta_{sr} \|x_s - x_r\|^{q-1} \text{signe}(x_s - x_r) \right]. \quad (3.90)$$

où la fonction  $\text{signe}(a)$  retourne 1 ou  $-1$  dépendant de si son argument est positif

TAB. 3.5 – Algorithme de super-résolution

<p><b>Algorithme de super-résolution</b>  <b>Entrées :</b> une liste d'images BR <math>\mathbf{y}_k</math>  <b>Sorties :</b> une image restaurée HR <math>\hat{\mathbf{x}}</math></p> <ol style="list-style-type: none"> <li>1. Estimation du mouvement des <math>k - 1</math> images BR par rapport à une image de référence <math>\mathbf{y}_r</math> (par exemple, la première image de la liste).</li> <li>2. Construction de la matrice <math>\mathbf{W}</math> à l'aide du recalage effectué à l'étape précédente et en tenant compte du modèle de PSF, en l'occurrence une fonction boîte.</li> <li>3. <math>\hat{\mathbf{x}}^0 \leftarrow \mathbf{y}'_r</math>, où <math>\mathbf{y}'_r</math> dénote une version HR de <math>\mathbf{y}_r</math> obtenue par simple interpolation.</li> <li>4. Itérer : <math display="block">\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma [\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n) - \alpha\Omega'(\hat{\mathbf{x}}^n)]</math> jusqu'à ce que <math>\hat{\mathbf{x}}^{n+1} \approx \hat{\mathbf{x}}^n</math>.</li> </ol> <p>Retourner l'image <math>\hat{\mathbf{x}}</math>.</p>
---

ou négatif.

En ce qui a trait au modèle *a priori* médian, la dérivée de  $\Omega_{\text{méd}}(\mathbf{x}, \mathbf{m})$  a la forme suivante :

$$\Omega'_{\text{méd}}(\mathbf{x}, \mathbf{m}) = \frac{1}{\eta} \sum_{j=1}^N \sum_{j' \in \mathcal{N}(j)} \tanh(\eta(x_j - m_{j'})) \quad (3.91)$$

et la minimisation (3.34) est calculée dans notre implantation par un filtre médian comme en (3.30). À noter que cette méthode ne correspond pas à la minimisation optimale (3.34), mais il s'agit d'une approximation pratique.

La minimisation de la fonction de coût du modèle basé sur une segmentation prend la forme [16] :

$$\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma [\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n) - \alpha\rho'(\hat{\mathbf{x}}_s^n)\rho(\hat{\mathbf{x}}_s^n)] \quad (3.92)$$

avec  $\rho'(\hat{\mathbf{x}}_s^n)$  comme suit :

$$\frac{\partial \rho(\mathbf{x}_s)}{\partial \mathbf{x}_{seg}} = 1 - \frac{1}{N_{seg}} \sum_{g=1}^{N_{seg}} \frac{1}{\mathcal{N}_g} \quad (3.93)$$

où  $\mathcal{N}_g$  est le nombre de pixels compris dans la région à laquelle un pixel donné appartient dans la  $g$ -ième segmentation.

Le cas du modèle basé sur un seuillage de la DCT requiert une dérogation à l'algorithme 3.5 : en effet, il est impossible d'exprimer son influence sous la forme d'une fonction de coût, et il nous faut donc appliquer le modèle sous une autre forme. Le tableau 3.6 présente ce nouvel algorithme. Il s'agit d'une itération alternant entre une exécution de l'algorithme de Landweber, suivie d'une application du seuillage de la DCT décrit ci-haut. Bien que la structure soit différente, on perçoit que l'itération de Landweber fait ici office de terme de fidélité aux données, alors que le seuillage "pénalise" les solutions trop bruitées. La fonction  $\Psi(\cdot)$  dans l'algorithme opère le seuillage de la DCT sur l'image  $\hat{\mathbf{x}}^{n+1}$  selon le seuil  $S$ . Soixante-quatre (64) carrelages de l'image sont générés, selon des décalages horizontaux et verticaux de 0 pixels à 7 pixels. Le seuillage est opéré sur chacun d'entre eux et la restauration est le résultat de la pondération de ces 64 débruitages selon la technique énoncée plus haut.

TAB. 3.6 – Algorithme de super-résolution par seuillage de la DCT

<b>Algorithme de super-résolution par seuillage de la DCT</b>	
<b>Entrées :</b>	une liste d'images BR $\mathbf{y}_k$
<b>Sorties :</b>	une image restaurée HR $\hat{\mathbf{x}}$
<ol style="list-style-type: none"> <li>1. Estimation du mouvement des <math>k - 1</math> images BR par rapport à une image de référence <math>\mathbf{y}_r</math> (par exemple, la première image de la liste).</li> <li>2. Construction de la matrice <math>\mathbf{W}</math> à l'aide du recalage effectué à l'étape précédente et en tenant compte du modèle de PSF, en l'occurrence une fonction boîte.</li> <li>3. <math>\hat{\mathbf{x}}^0 \leftarrow \mathbf{y}'_r</math>, où <math>\mathbf{y}'_r</math> dénote une version HR de <math>\mathbf{y}_r</math> obtenue par simple interpolation.</li> <li>4. Itérer : <div style="text-align: center; margin: 10px 0;"> <math display="block">\hat{\mathbf{x}}^{n+1} = \hat{\mathbf{x}}^n - \gamma [\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n)]</math> <math display="block">\hat{\mathbf{x}}^{n+1} \leftarrow \Psi(\hat{\mathbf{x}}^{n+1}, S)</math> </div> <p style="text-align: center; margin: 10px 0;">jusqu'à ce que <math>\hat{\mathbf{x}}^{n+1} \approx \hat{\mathbf{x}}^n</math>.</p> </li> </ol>	
Retourner l'image $\hat{\mathbf{x}}$ .	

## CHAPITRE 4

### PRÉSENTATION DES RÉSULTATS ET DISCUSSION

Nous présentons dans ce chapitre l'étude de la performance des différents modèles *a priori* dans le cadre de la super-résolution. À cette fin nous décrivons la nature des tests effectués et la mesure de qualité utilisée sur les résultats. Nous poursuivons avec une explication, pour chaque modèle, des paramètres étudiés, ceux qui sont gardés fixes durant les tests, et dans chaque cas le pourquoi de ce choix. Nous enchaînons avec la présentation des résultats proprement dits pour finir avec une discussion de ceux-ci.

#### 4.1 Méthodologie de test

Nous nous sommes fixé comme objectif au début de ce mémoire de comparer les performances, dans le cadre de la super-résolution, de trois modèles *a priori* par rapport à deux modèles témoins, le modèle TM et le modèle GGMRF.

##### 4.1.1 Génération synthétique de séquences d'images BR

La phase de test requiert que nous ayons à notre disposition une ou plusieurs séquences d'images basse-résolution. Nous avons opté pour l'utilisation d'images BR générées par nous à partir d'une image HR selon un schéma de décalage prédéterminé. Cette approche présente des avantages évidents. Tout d'abord l'image originale constitue la restauration idéale contre laquelle la performance des différents modèles sera mesurée : les séquences BR étant dérivées de l'image HR, cette dernière présente le maximum d'information qu'il est possible d'extraire de la séquence. De plus, la génération synthétique de séquence nous donne une connaissance parfaite de la grille de décalage (c'est-à-dire le mouvement) entre les images de la séquence. Cette connaissance nous évite de dépendre pour nos résultats de techniques de recalage : notre étude portant sur les modèles *a priori*, l'estimation

du mouvement à partir d'une séquence vidéo ajouterait un élément de dégradation des résultats superflu à notre objectif. Finalement la génération synthétique nous permet de contrôler le nombre et la qualité des images recouvrant la zone d'intérêt (une amplitude de décalage trop élevée entre les images BR réduit la taille de la région dans l'image HR où plusieurs des images BR se superposent – et donc où la super-résolution peut être opérée).

Nous avons généré nos séquences BR à partir de quatre images en niveaux de gris et de résolution  $512 \times 512$ , que l'on retrouve à la figure 4.1. Ces images sont l'objet d'un usage fréquent en traitement d'image en raison de leurs attributs différents. L'image `f16.pgm` présente un degré de complexité plutôt faible avec un arrière-plan aux variations douces et un avant-plan présentant des arêtes nettes. À l'opposé nous retrouvons l'image `baboon.pgm`, où le motif de fourrure fournit un élément haute-fréquence sur presque toute l'image. L'image `lena.pgm` est un exemple d'image de complexité moyenne ; il en va de même pour l'image `barbara.pgm`, à l'exception du motif de la nappe, qui dans ses différentes orientations apporte un degré de complexité supérieur. Les différents degrés de complexité nous permettent de mesurer les différents modèles selon leur performance quant à la préservation simultanée des zones unies et des arêtes de l'image, tout en présentant des éléments dont la qualité de la restauration nous renseignera qualitativement sur l'acuité de chaque modèle.

Pour nos tests nous avons généré pour chacune des images HR mentionnées ci-haut une séquence d'images BR comportant 16 images de résolution  $128 \times 128$ . Chaque image est générée à partir de l'image HR selon la méthode simple suivante : le cadre de l'image est décalé par rapport à l'image de départ selon un mouvement purement translationnel (c'est-à-dire sans rotation ou transformation perspective) de 0 à 3 pixels dans les dimensions verticale et horizontale (nous adoptons le système de coordonnées qui situe le point d'origine  $(0, 0)$  au coin supérieur gauche de l'image, avec la première coordonnée référant à l'axe horizontal et la seconde à l'axe vertical). Les pixels des côtés droit et inférieur de l'image sont répétés lorsque que ce cadre dépasse ceux de l'image originale. L'image BR est en-



FIG. 4.1 – Les quatre images  $512 \times 512$  utilisées pour les tests de super-résolution : a) baboon.pgm ; b) barbara.pgm ; c) f16.pgm ; d) lena.pgm.

suite obtenue en réduisant la résolution par un facteur de 4 : chaque pixel BR est obtenu simplement en faisant la moyenne des quatre pixels HR correspondants. Il est ensuite possible d'ajouter un bruit à l'image BR ; lorsque prescrit par les tests, un bruit gaussien blanc d'une variance de 10, 25 ou 100 a été ajouté aux images des séquences.

À partir d'une image  $512 \times 512$ , nous générons donc seize images  $128 \times 128$ , bruitées ou non, et présentant l'une par rapport aux autres un décalage d'ordre subpixel, c'est-à-dire d'une amplitude inférieure à un pixel BR.

La batterie de tests que nous avons élaborée se penche sur trois facettes du problème de la super-résolution. Dans un premier temps nous faisons fluctuer la variance du bruit dans l'image. Dans un second, nous varions le nombre d'images BR utilisées. Finalement, certains des modèles comportent des paramètres qu'il est possible de modifier pour en étudier les effets. Nous reviendrons sur ce dernier aspect à la section suivante.

#### 4.1.2 La variance du bruit de l'image

Pour chaque séquence, quatre versions sont produites avec des variances de bruit gaussien différentes : 0 (aucun bruit), 10, 25 et 100. Cette variation de l'amplitude du bruit permet de mesurer les propriétés de restauration de chaque modèle au-delà de la super-résolution. Lorsqu'un grand nombre d'images BR sont disponibles, l'apport en information de celles-ci permet d'"éliminer" l'effet du bruit, et la performance de la super-résolution domine ; cependant, si peu d'images sont utilisées, le bruit est un facteur important dans dégradation et le modèle *a priori* doit composer avec ses effets.

#### 4.1.3 Le nombre d'images BR pour la super-résolution

Le nombre d'images BR disponibles a un impact important sur les résultats de la super-résolution. Outre l'effet d'atténuation du bruit qu'un grand échantillon d'images permet, la quantité de pixels BR à notre disposition a un impact direct



FIG. 4.2 – Les 16 images  $128 \times 128$  générées à partir de `lena.pgm`, avec un bruit de variance 100.

Nombre d'images	Décalages présents dans la séquence
16	Tous
8	(0, 0), (1, 1), (2, 2), (3, 3), (0, 3), (1, 2), (2, 1), (3, 0)
4	(0, 0), (1, 1), (2, 2), (3, 3)
2	(0, 0), (2, 2)

TAB. 4.1 – Les décalages, mesurés en pixels HR, des images utilisées dépendant du nombre d'images dans chaque séquence BR

sur la qualité de la super-résolution qu'il est possible d'obtenir théoriquement et en pratique. Les tests sont donc exécutés sur des séries de 16, 8, 4 ou 2 images, prélevées de la séquence de seize images BR décrites plus haut. Le tableau 4.1 présente pour chaque séquence les décalages utilisés.

#### 4.1.4 Les paramètres

Chaque modèle *a priori* comporte des paramètres qui influent sur la qualité des résultats. Au cours de notre investigation nous avons isolé certains paramètres qui nous semblaient avoir un impact plus important sur les résultats : concrètement, l'optimisation de ces paramètres permet d'améliorer la qualité de la reconstruction sous des conditions plus ou moins spécifiques de bruit et du nombre d'images disponibles. Ces paramètres ont donc été l'objet d'expérimentations dans notre batterie de tests. La variation des autres paramètres s'est démontrée présenter un potentiel d'optimisation moindre, et par conséquent ceux-ci ont été fixés à des valeurs jugées bonnes en ce qui a trait à notre cadre d'imagerie, ici aussi après expérimentation empirique. Les sections qui suivent présentent pour chaque modèle ces paramètres et les valeurs fixées ou testées. Pour ce faire nous nous référons aux équations de descente du gradient élaborées au chapitre précédent. Dans tous les cas sauf pour le modèle basé sur une segmentation de [16], le pas du gradient est fixé à  $\gamma = 1$ .

#### 4.1.4.1 Modèle Tikhonov-Miller

Le modèle *a priori* de Tikhonov-Miller (éq. (3.89)) comporte, hormis le pas  $\gamma$  et le filtre passe-haut  $\mathbf{C}$  (modélisé par un opérateur Laplacien), un seul paramètre, le paramètre  $\alpha$  qui module l'influence du terme de pénalité par rapport au terme de fidélité aux données. Trois valeurs ont fait l'objet d'essais : 0, 0,001 et 0,01. Ces valeurs sont choisies après expérimentation :  $\alpha = 0$  n'impose aucune contrainte et entraîne une grande sensibilité au bruit et au manque d'images BR ;  $\alpha > 0,01$  résulte en des images au bruit atténué mais avec un flou qui contrevient à l'exigence de préservation des arêtes. À noter que lorsque  $\alpha = 0$ , le modèle *a priori* est sans influence et l'opération est réduite à une minimisation de Landweber.

#### 4.1.4.2 Modèle GGMRF

Le modèle GGMRF (Eq. (3.90)) comporte un paramètre variable, le facteur de forme  $q$ , qui est testé aux valeurs 1.2, 1.5 et 2.0. Ces valeurs sont le fruit de l'expérimentation et concordent avec la prescription dans [37] qui prône  $1 \leq q \leq 2$ . Le paramètre  $\lambda$  est fixé à 0.5, et le coefficient  $\beta_{sr}$  est décrit à la section 3.8.2.

#### 4.1.4.3 Modèle médian

Le modèle médian (éqs. (3.33) et (3.91)) comporte deux paramètres variables :  $\alpha$ , qui module l'influence du terme de pénalité, et  $\eta$ , qui contrôle le degré de raffinement de l'approximation de la valeur absolue dans le modèle. Des valeurs de 0.5, 1.0 et 1.5 sont testées pour  $\alpha$  ; 1, 5 et 20 pour  $\eta$ . Ces valeurs se basent sur des données empiriques et concordent avec les valeurs citées dans [14] pour des images comportant un bruit faible (et, rappelons-le, dans le cadre d'une restauration conventionnelle à partir d'une image unique). Les valeurs suggérées pour les images fortement bruitées ( $\alpha = 80$  par exemple) se sont révélées inadaptées à notre cadre.

#### 4.1.4.4 Modèle basé sur une segmentation

Le modèle *a priori* basé sur une segmentation proposé dans [16] ne comporte aucun paramètre qu'il soit utile de moduler manuellement. De plus tous les paramètres fixes se retrouvent en argument pour la segmentation markovienne : en l'occurrence, nous avons fixé le nombre de segmentations  $N_{Seg}$  à 8, le nombre de classes  $K$  à 20 et le nombre maximal de pixels dans chaque région de segmentation à 10. Pour ce modèle nous avons implanté un paramètre  $\alpha$  évalué automatiquement au début du calcul, selon l'équation :

$$\alpha = \zeta \cdot \frac{\|\mathbf{W}^T(\mathbf{y} - \mathbf{W}\hat{\mathbf{x}}^n)\|^2}{\Omega(\hat{\mathbf{x}}^0)} \quad (4.1)$$

où  $\hat{\mathbf{x}}^0$  dénote la restauration GGMRF initiale et où  $\zeta = 0.1$ .

#### 4.1.4.5 Modèle basé sur un seuillage de la DCT

Le seuillage du résultat de la transformée cosinus discrète dans l'algorithme 3.6 est effectué à des valeurs de seuil de 1.0, 1.2, 1.4, 2.5, 5.0 et 7.5. Cette grande amplitude dans les valeurs s'explique par la qualité remarquable de ce modèle *a priori* à tous les niveaux de bruit et de nombre d'images, pour peu qu'un seuil adapté soit utilisé, tel que révélé lors de nos essais. Un seuil fixe a été utilisé pour tous les tests.

#### 4.1.5 La mesure de performance

Nous récapitulons ici en quelques mots la structure de la batterie de tests exécutés dans le cadre du présent travail. Pour chacune des quatre images HR de départ, une séquence de 16 images a été générée, sur lesquelles cinq modèles *a priori* ont été appliqués. Chacun de ceux-ci a été testé sur toutes les permutations possibles de quatre variances de bruit (0, 10, 25, 100) et de quatre longueurs de séquence d'images (16, 8, 4, 2). Cette opération a été répétée pour chacune des valeurs des paramètres énumérés dans la section précédente. Au total, près de 2000

tests ont été effectués dans la génération des résultats finaux, sans compter les évaluations empiriques préalables.

La qualité de l'image reconstruite lors de chaque test individuel a été mesurée à l'aide du RSBP (rapport signal sur bruit de pointe, ou PSNR, de l'anglais *Peak Signal to Noise Ratio*), défini comme suit :

$$\text{RSBP} = 20 \cdot \log_{10} \left( \frac{255}{\sqrt{\text{EQM}}} \right) \quad (4.2)$$

où EQM dénote l'erreur quadratique moyenne sur toute l'image, calculée par :

$$\text{EQM} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|\hat{\mathbf{x}}_{i,j} - \mathbf{x}_{i,j}\|^2. \quad (4.3)$$

Notre choix s'est arrêté sur cette mesure en raison d'une part de son usage fréquent en traitement d'images, et d'autre part de la multiplicité des images BR  $\mathbf{y}_k$  et la différence de dimensions entre l'image reconstruite  $\hat{\mathbf{x}}$  et ces dernières, qui rendent l'utilisation de l'ARSB (amélioration du rapport signal sur bruit, en anglais ISNR – *Improvement in Signal to Noise Ratio*) moins rigoureuse.

## 4.2 Présentation des résultats

La présente section décline les résultats présentés dans le cadre de ce mémoire. Il s'agit de données, sélectionnées parmi la masse de résultats produits lors de la batterie de tests, qui illustrent bien les capacités des différents modèles appliqués à la super-résolution. La section qui suit procèdera à l'interprétation de ces résultats. Les tableaux 4.2 à 4.5 présentent les modèles et paramètres donnant les meilleurs résultats, tous modèles confondus, pour chaque séquence d'images et pour chaque combinaison de bruit et de longueur de séquence. Les tableaux 4.6 à 4.9 fournissent les paramètres qui conduisent aux meilleurs résultats pour chaque modèle individuellement. Le tableau 4.10 décline, pour chaque modèle et chaque séquence d'images, le ou les paramètres qui donnent les meilleurs résultats en moyenne, pris

	Nombre d'images				
		16	8	4	2
Variance	0	TM $\alpha = 0.0$ 23.033825	TM $\alpha = 0.0$ 22.332108	DCT $S = 1.0$ 21.888016	DCT $S = 1.0$ 21.324421
	10	TM $\alpha = 0.0$ 22.282444	DCT $S = 1.0$ 22.055510	DCT $S = 1.0$ 21.808300	DCT $S = 1.2$ 21.156620
	25	DCT $S = 1.0$ 22.167210	DCT $S = 1.0$ 22.005617	DCT $S = 1.0$ 21.651451	DCT $S = 1.4$ 20.932926
	100	DCT $S = 1.0$ 21.983284	DCT $S = 1.2$ 21.427805	DCT $S = 2.5$ 20.917763	GGMRF $q = 1.5$ 20.278242

TAB. 4.2 – Modèles et paramètres menant aux meilleurs résultats pour la séquence **baboon** tous modèles confondus

sur toutes les variances et longueurs de séquence. La figure 4.3 illustre la performance de chaque modèle, évalué avec les paramètres cités au tableau 4.10, sur la séquence **lena**. Deux modèles *a priori* ont deux choix de paramètres possibles dans ce tableau. Pour le modèle médian, nous avons choisi d'illustrer le résultat des paramètres  $\alpha = 1.5$ ,  $\eta = 20$ . Pour le modèle basé sur un seuillage de la DCT, nous avons opté pour  $S = 1.4$ . Les figures 4.4 à 4.6 montrent des images résultant de l'application des différents modèles à la super-résolution de séquences d'images BR. Les figures 4.7 et 4.8 fournissent des régions grossies afin de mettre en évidence les effets de chaque modèle sur la super-résolution. Les figures 4.9 et 4.10 attirent l'attention vers le comportement des différents modèles en présence de détails périodiques.

### 4.3 Discussion

Dans un premier temps les données rassemblées lors de l'expérimentation ont généralement confirmé l'intuition énoncée précédemment, à savoir que l'augmentation du bruit et/ou la réduction du nombre d'images BR dans la séquence im-

	Nombre d'images				
		16	8	4	2
Variance	0	DCT $S = 1.0$ 24.608608	DCT $S = 1.0$ 24.579742	DCT $S = 1.0$ 24.456303	DCT $S = 1.0$ 24.240257
	10	DCT $S = 1.0$ 24.579927	DCT $S = 1.0$ 24.546061	DCT $S = 1.0$ 24.398672	DCT $S = 1.4$ 24.094372
	25	DCT $S = 1.0$ 24.557646	DCT $S = 1.0$ 24.504702	DCT $S = 1.0$ 24.249908	DCT $S = 2.5$ 23.950974
	100	DCT $S = 1.0$ 24.402018	DCT $S = 1.4$ 24.079903	DCT $S = 2.5$ 23.811058	DCT $S = 5.0$ 23.497025

TAB. 4.3 – Modèles et paramètres menant aux meilleurs résultats pour la séquence *barbara* tous modèles confondus

	Nombre d'images				
		16	8	4	2
Variance	0	DCT $S = 1.0$ 31.166519	DCT $S = 1.0$ 31.071877	DCT $S = 1.0$ 30.861071	DCT $S = 1.2$ 29.368763
	10	DCT $S = 1.0$ 31.024054	DCT $S = 1.0$ 30.798471	DCT $S = 1.0$ 30.426809	DCT $S = 1.4$ 28.940369
	25	DCT $S = 1.0$ 30.862019	DCT $S = 1.0$ 30.496655	DCT $S = 1.4$ 29.906010	DCT $S = 2.5$ 28.560780
	100	DCT $S = 1.2$ 30.094307	DCT $S = 2.5$ 29.250435	DCT $S = 2.5$ 28.673500	DCT $S = 5.0$ 26.739588

TAB. 4.4 – Modèles et paramètres menant aux meilleurs résultats pour la séquence *f16* tous modèles confondus

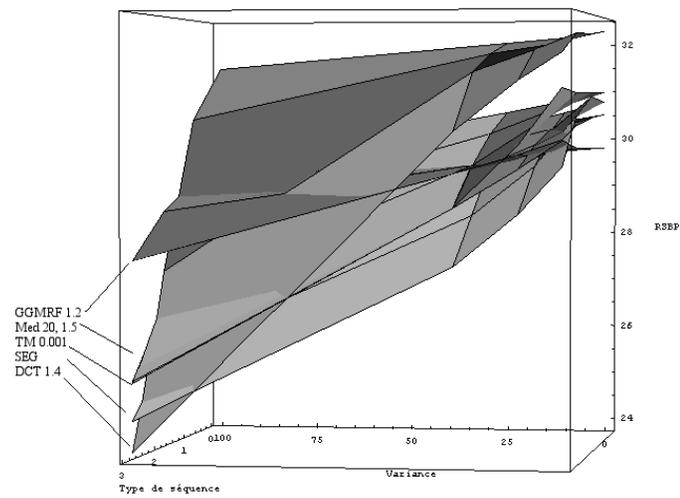
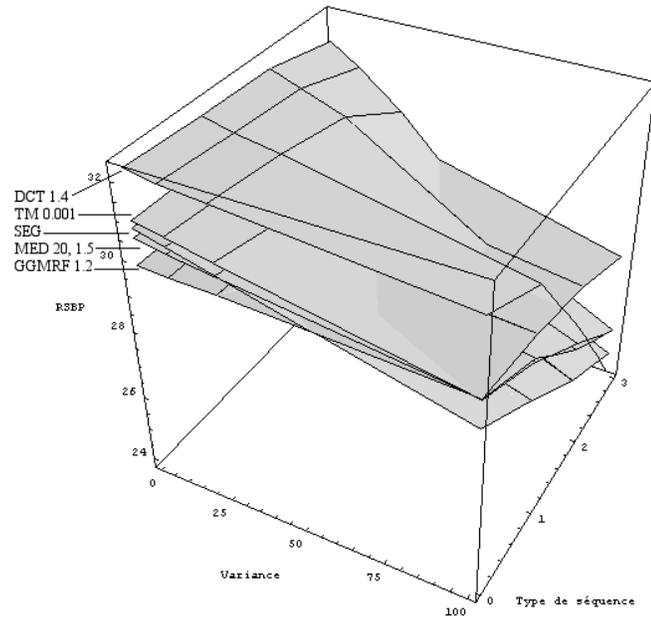


FIG. 4.3 – Représentation graphique des performances des différents modèles *a priori* selon des paramètres jugés bons en général au tableau 4.10, lorsqu'appliqués à la séquence **lena**

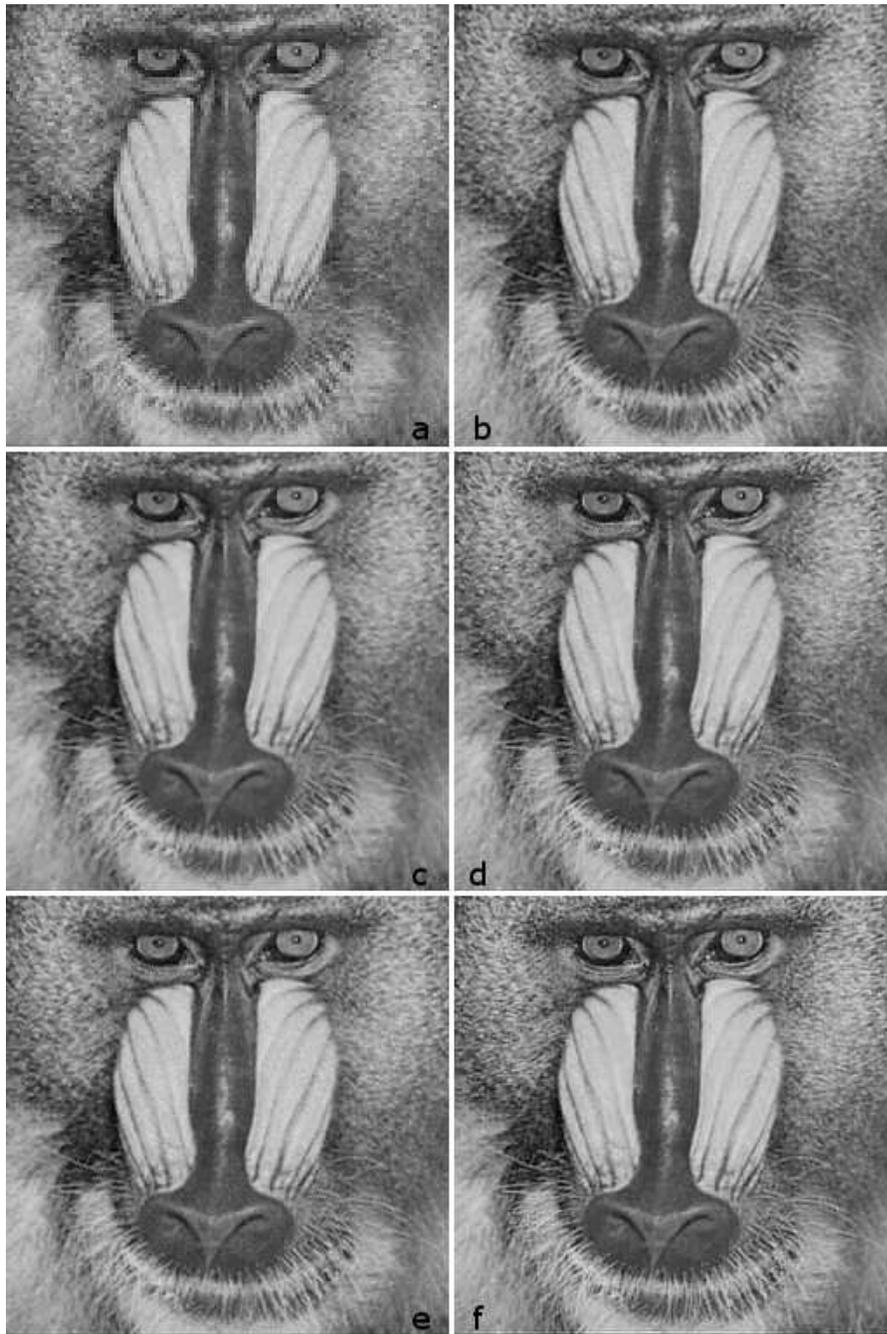


FIG. 4.4 – Exemples de résultats pour la séquence baboon (4 images,  $\sigma^2 = 25$ ) : a) image BR avec un décalage de (0,0); b) modèle TM :  $\alpha = 0.001$ ; c) modèle GGMRF :  $q = 1.0$ ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 1.0$ ; e) modèle basé sur une segmentation; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$



FIG. 4.5 – Exemples de résultats pour la séquence **barbara** (16 images,  $\sigma^2 = 0$ ) : a) image BR avec un décalage de (0,0) ; b) modèle TM :  $\alpha = 0.0$  ; c) modèle GGMRF :  $q = 1.2$  ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 0.5$  ; e) modèle basé sur une segmentation ; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$



FIG. 4.6 – Exemples de résultats pour la séquence **lena** (16 images,  $\sigma^2 = 0$ ) : a) image BR avec un décalage de (0,0); b) modèle TM :  $\alpha = 0.001$ ; c) modèle GGMRF :  $q = 1.2$ ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 0.5$ ; e) modèle basé sur une segmentation; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$

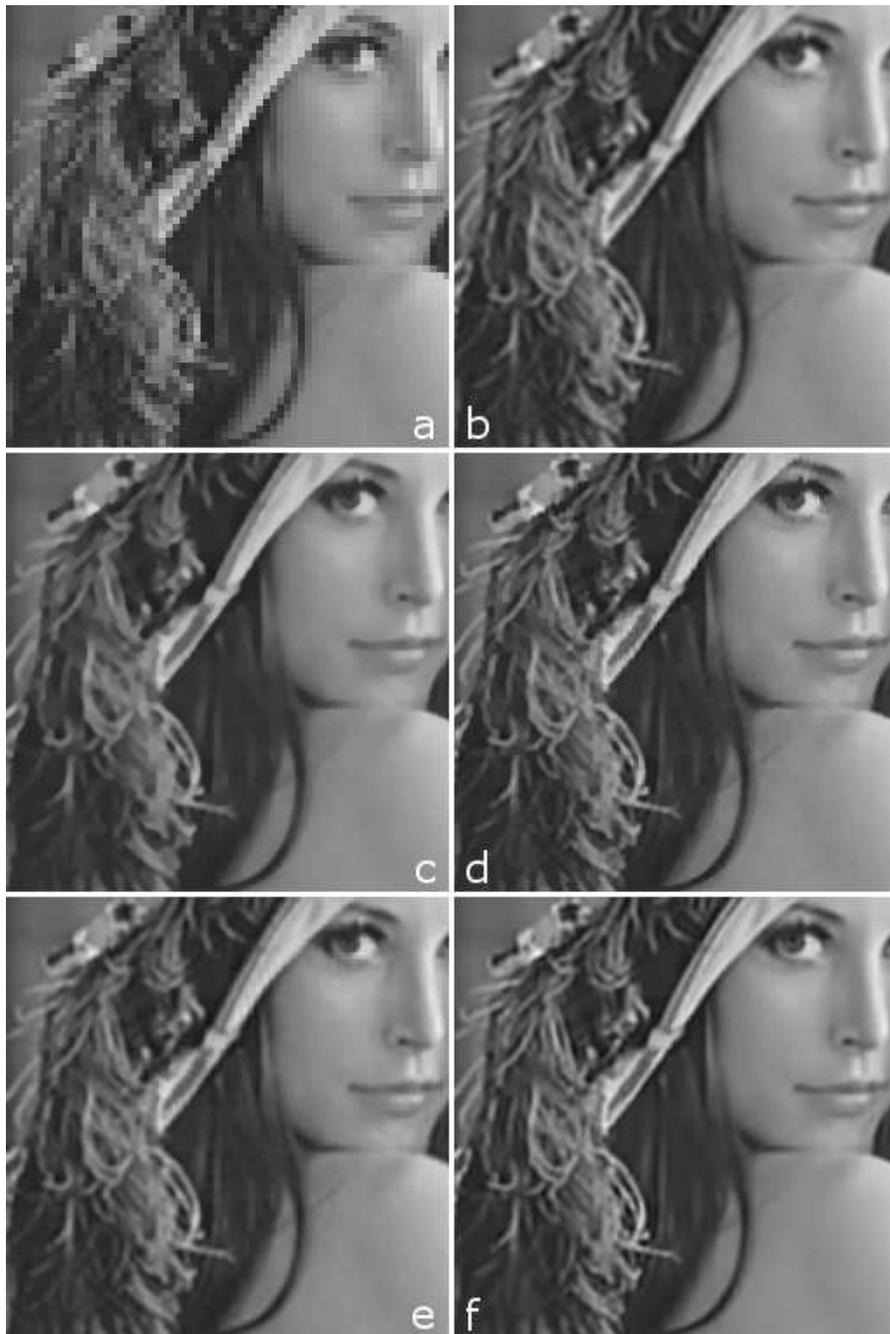


FIG. 4.7 – Région d'intérêt, séquence `lena` (16 images,  $\sigma^2 = 0$ ) : a) image BR avec un décalage de  $(0,0)$  ; b) modèle TM :  $\alpha = 0.001$  ; c) modèle GGMRF :  $q = 1.2$  ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 0.5$  ; e) modèle basé sur une segmentation ; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$

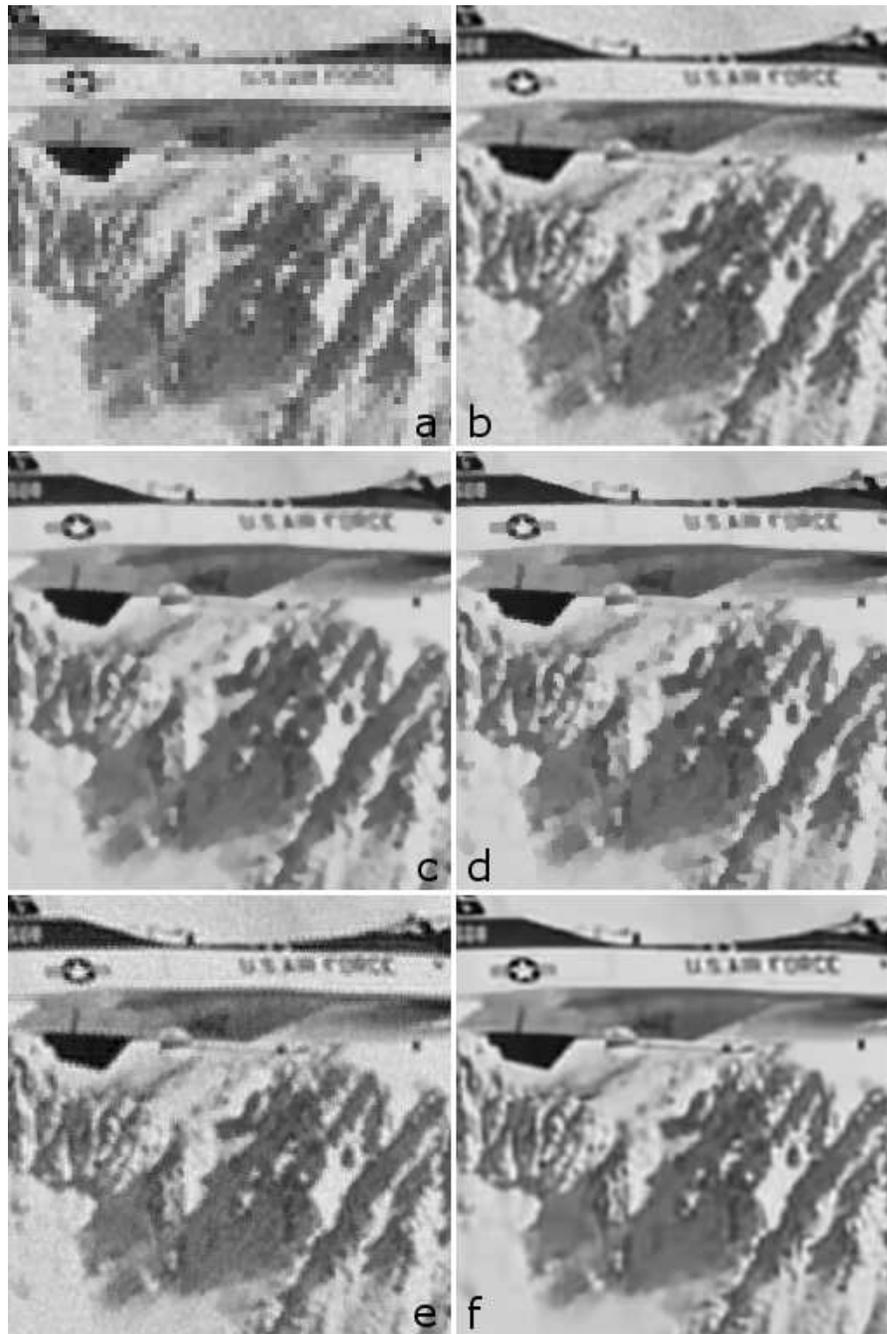


FIG. 4.8 – Région d'intérêt, séquence f16 (8 images,  $\sigma^2 = 25$ ) : a) image BR avec un décalage de (0,0) ; b) modèle TM :  $\alpha = 0.001$  ; c) modèle GGMRF :  $q = 1.2$  ; d) modèle médian :  $\eta = 5$ ,  $\alpha = 0.5$  ; e) modèle basé sur une segmentation ; f) modèle basé sur un seuillage de la DCT :  $S = 1.4$

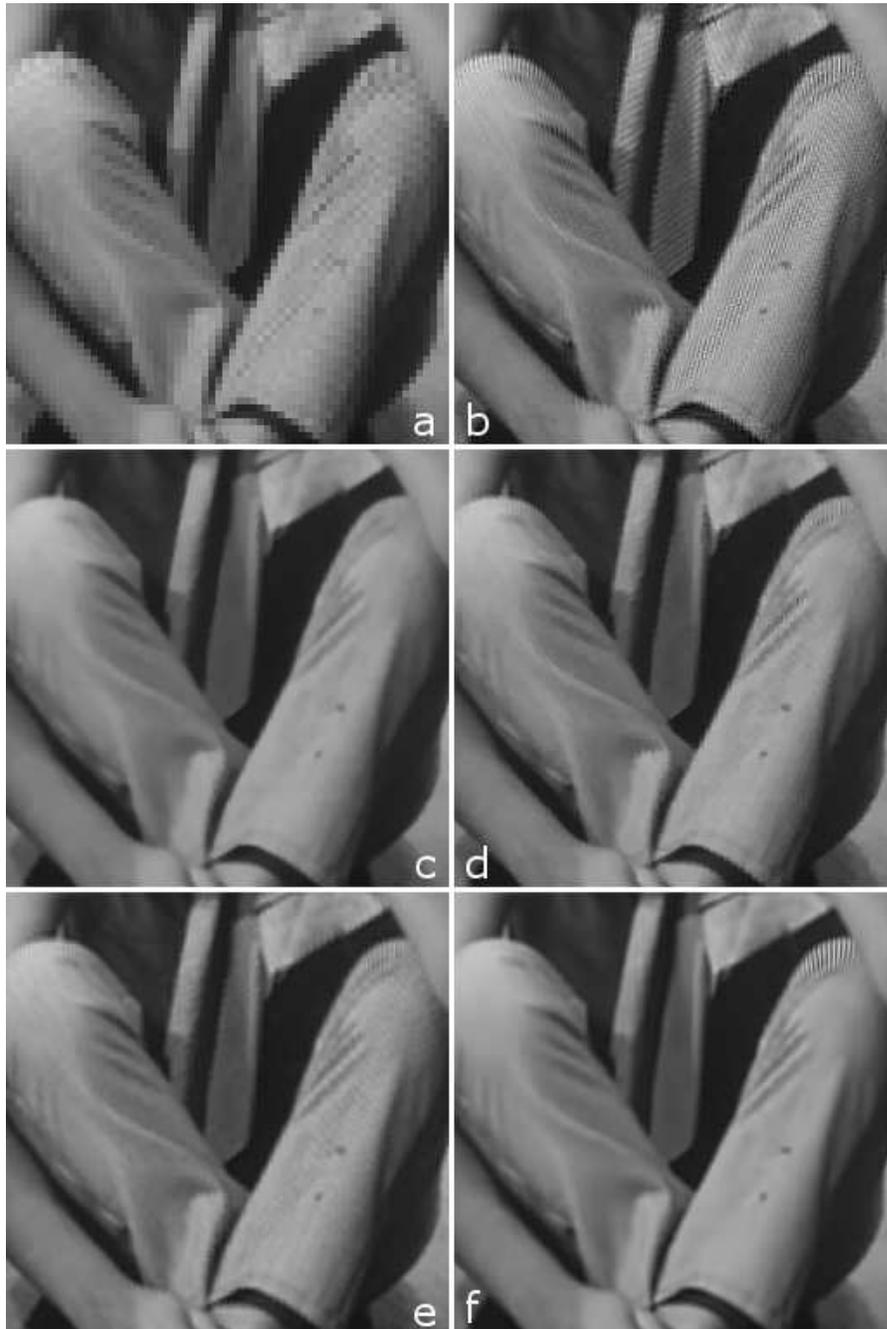


FIG. 4.9 – Détail des pantalons, séquence *barbara* (16 images,  $\sigma^2 = 0$ ) : a) image BR avec un décalage de (0,0) ; b) modèle TM :  $\alpha = 0.0$  ; c) modèle GGMRF :  $q = 1.2$  ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 0.5$  ; e) modèle basé sur une segmentation ; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$

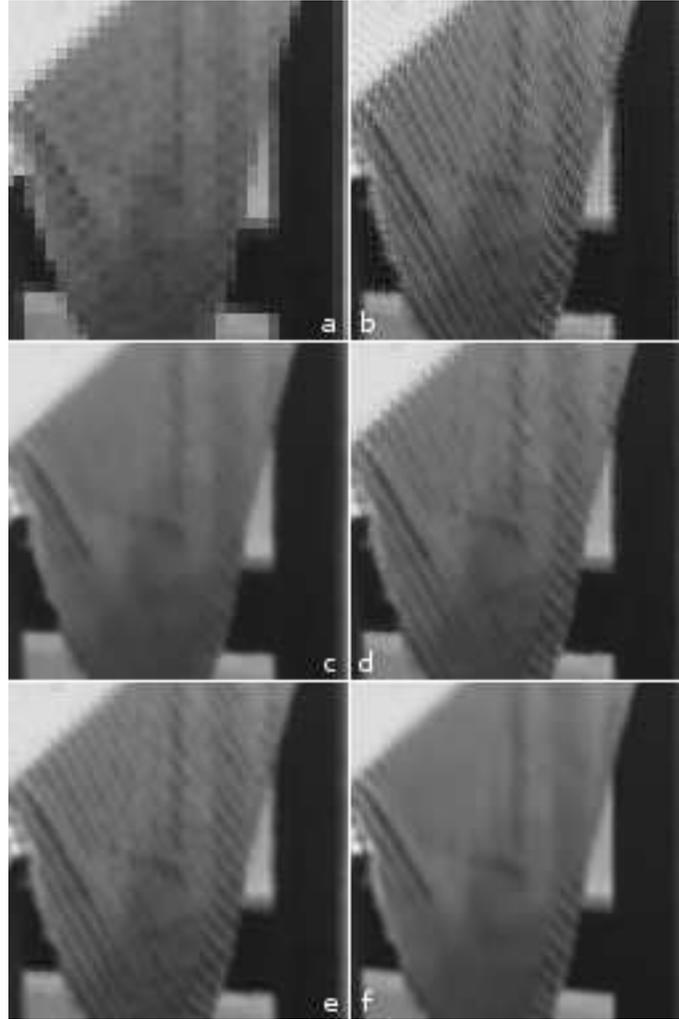


FIG. 4.10 – Détail de la nappe, séquence *barbara* (16 images,  $\sigma^2 = 0$ ) : a) image BR avec un décalage de (0,0) ; b) modèle TM :  $\alpha = 0.0$  ; c) modèle GGMRF :  $q = 1.2$  ; d) modèle médian :  $\eta = 20$ ,  $\alpha = 0.5$  ; e) modèle basé sur une segmentation ; f) modèle basé sur un seuillage de la DCT :  $S = 1.0$

	Nombre d'images				
		16	8	4	2
Variance	0	DCT $S = 1.0$ 32.925705	DCT $S = 1.0$ 32.749470	DCT $S = 1.0$ 32.612068	DCT $S = 1.0$ 31.861307
	10	DCT $S = 1.0$ 32.709705	DCT $S = 1.0$ 32.438019	DCT $S = 1.0$ 32.085709	DCT $S = 1.4$ 31.142033
	25	DCT $S = 1.0$ 32.532066	DCT $S = 1.0$ 32.080151	DCT $S = 1.2$ 31.367348	DCT $S = 2.5$ 30.290194
	100	DCT $S = 1.2$ 31.543137	DCT $S = 2.5$ 30.420776	DCT $S = 2.5$ 29.801193	DCT $S = 5.0$ 28.518902

TAB. 4.5 – Modèles et paramètres menant aux meilleurs résultats pour la séquence **lena** tous modèles confondus

plique qu'une influence supérieure soit donnée au modèle *a priori* afin d'obtenir de meilleurs résultats. Concrètement, cela signifie que  $\alpha$  varie inversement au nombre d'images et proportionnellement à l'amplitude du bruit pour les modèles où ce paramètre était modulé lors des tests. Pour le modèle médian, la valeur optimale pour  $\eta$  suit un comportement contraire, c'est-à-dire qu'elle descend lorsque  $\alpha$  augmente. Ce phénomène s'explique par le fait que l'augmentation de la valeur de  $\eta$  se solde par une amplification de la préservation des arêtes [14]; or en présence d'un bruit important et/ou d'un nombre faible d'images, ce comportement risque de préserver des artefacts externes au signal lui-même. Donc dans ce cas il est préférable que la contrainte de conservation des arêtes soit relaxée. Ceci reflète l'action des autres modèles dans la même situation, soit une augmentation générale du flou dans les résultats. En ce qui a trait au modèle basé sur un seuillage de la DCT, le seuil  $S$  augmente, ce qui apparaît logique : l'augmentation du bruit et des artefacts est mieux contrecarrée par un seuil élevé pour la DCT, qui estompe le bruit – et les détails – dans la solution.

Dans un second temps, on constate la prépondérance du modèle basé sur un

Nombre d'images				
$\sigma^2$	16	8	4	2
Modèle Tikhonov-Miller				
0	$\alpha = 0.0$ 23.033825	$\alpha = 0.0$ 22.332108	$\alpha = 0.0$ 21.482559	$\alpha = 0.001$ 21.137907
10	$\alpha = 0.0$ 22.282444	$\alpha = 0.0$ 21.499134	$\alpha = 0.001$ 21.251572	$\alpha = 0.001$ 21.007870
25	$\alpha = 0.0$ 21.474350	$\alpha = 0.001$ 21.312105	$\alpha = 0.001$ 21.136456	$\alpha = 0.001$ 20.830801
100	$\alpha = 0.001$ 21.164955	$\alpha = 0.001$ 20.948849	$\alpha = 0.001$ 20.578327	$\alpha = 0.01$ 20.144548
Modèle GGMRF				
0	$q = 1.2$ 21.328871	$q = 1.2$ 21.268806	$q = 1.2$ 21.178169	$q = 1.2$ 20.949068
10	$q = 1.2$ 21.292200	$q = 1.2$ 21.236500	$q = 1.2$ 21.127171	$q = 1.2$ 20.874434
25	$q = 1.2$ 21.256552	$q = 1.2$ 21.191391	$q = 1.2$ 21.056736	$q = 1.2$ 20.769817
100	$q = 1.2$ 21.069885	$q = 1.2$ 20.929058	$q = 1.2$ 20.646801	$q = 1.5$ 20.278242
Modèle médian				
0	$\eta = 20, \alpha = 0.5$ 22.243069	$\eta = 20, \alpha = 0.5$ 21.983200	$\eta = 20, \alpha = 0.5$ 21.601177	$\eta = 20, \alpha = 0.5$ 21.026133
10	$\eta = 20, \alpha = 0.5$ 22.113453	$\eta = 20, \alpha = 0.5$ 21.837379	$\eta = 20, \alpha = 0.5$ 21.374073	$\eta = 20, \alpha = 0.5$ 20.846956
25	$\eta = 20, \alpha = 0.5$ 21.952206	$\eta = 20, \alpha = 0.5$ 21.588818	$\eta = 20, \alpha = 1.0$ 21.121269	$\eta = 20, \alpha = 1.0$ 20.637985
100	$\eta = 20, \alpha = 1.0$ 21.216486	$\eta = 20, \alpha = 1.5$ 20.845612	$\eta = 5, \alpha = 1.0$ 20.413034	$\eta = 5, \alpha = 1.0$ 20.077211
Modèle basé sur une segmentation				
0	N/A 22.073946	N/A 21.784143	N/A 21.465565	N/A 20.842274
10	N/A 21.958920	N/A 21.613983	N/A 21.291679	N/A 20.688591
25	N/A 21.827696	N/A 21.460499	N/A 21.084423	N/A 20.485935
100	N/A 21.201492	N/A 20.775282	N/A 20.242292	N/A 19.596430
Modèle basé sur un seuillage de la DCT				
0	$S = 1.0$ 22.204950	$S = 1.0$ 22.089581	$S = 1.0$ 21.888016	$S = 1.0$ 21.324421
10	$S = 1.0$ 22.182369	$S = 1.0$ 22.055510	$S = 1.0$ 21.808300	$S = 1.2$ 21.156620
25	$S = 1.0$ 22.167210	$S = 1.0$ 22.005617	$S = 1.0$ 21.651451	$S = 1.4$ 20.932926
100	$S = 1.0$ 21.983284	$S = 1.2$ 21.427805	$S = 2.5$ 20.917763	$S = 2.5$ 20.250572

TAB. 4.6 – Paramètres menant aux meilleurs résultats pour la séquence baboon pour chaque modèle

Nombre d'images				
$\sigma^2$	16	8	4	2
Modèle Tikhonov-Miller				
0	$\alpha = 0.0$ 24.369400	$\alpha = 0.001$ 24.136057	$\alpha = 0.001$ 24.100723	$\alpha = 0.001$ 24.080387
10	$\alpha = 0.001$ 24.120436	$\alpha = 0.001$ 24.060076	$\alpha = 0.001$ 23.957197	$\alpha = 0.001$ 23.838789
25	$\alpha = 0.001$ 24.063854	$\alpha = 0.001$ 23.954229	$\alpha = 0.001$ 23.762291	$\alpha = 0.01$ 23.588640
100	$\alpha = 0.001$ 23.764580	$\alpha = 0.01$ 23.505838	$\alpha = 0.01$ 23.347609	$\alpha = 0.01$ 23.086283
Modèle GGMRF				
0	$q = 1.2$ 24.022484	$q = 1.2$ 24.014006	$q = 1.2$ 24.020569	$q = 1.2$ 24.034576
10	$q = 1.2$ 24.005787	$q = 1.2$ 23.981428	$q = 1.2$ 23.955900	$q = 1.2$ 23.920135
25	$q = 1.2$ 23.981905	$q = 1.2$ 23.937134	$q = 1.2$ 23.861710	$q = 1.2$ 23.749743
100	$q = 1.2$ 23.829100	$q = 1.2$ 23.646948	$q = 1.5$ 23.450436	$q = 1.5$ 23.175299
Modèle médian				
0	$\eta = 20, \alpha = 0.5$ 24.232176	$\eta = 20, \alpha = 0.5$ 24.162777	$\eta = 20, \alpha = 0.5$ 24.069138	$\eta = 20, \alpha = 1.0$ 23.865835
10	$\eta = 20, \alpha = 0.5$ 24.145048	$\eta = 20, \alpha = 0.5$ 24.000401	$\eta = 20, \alpha = 1.0$ 23.868252	$\eta = 20, \alpha = 1.5$ 23.709564
25	$\eta = 20, \alpha = 0.5$ 24.020004	$\eta = 20, \alpha = 1.0$ 23.866718	$\eta = 20, \alpha = 1.5$ 23.703690	$\eta = 20, \alpha = 1.5$ 23.491590
100	$\eta = 20, \alpha = 1.5$ 23.698788	$\eta = 5, \alpha = 1.0$ 23.462334	$\eta = 5, \alpha = 1.0$ 23.248363	$\eta = 1, \alpha = 0.5$ 23.063534
Modèle basé sur une segmentation				
0	N/A 24.258375	N/A 24.186081	N/A 24.057486	N/A 23.733042
10	N/A 24.170168	N/A 24.003979	N/A 23.810177	N/A 23.482389
25	N/A 24.037769	N/A 23.809547	N/A 23.514042	N/A 23.154999
100	N/A 23.553444	N/A 23.065250	N/A 22.414370	N/A 21.797745
Modèle basé sur un seuillage de la DCT				
0	$S = 1.0$ 24.608608	$S = 1.0$ 24.579742	$S = 1.0$ 24.456303	$S = 1.0$ 24.240257
10	$S = 1.0$ 24.579927	$S = 1.0$ 24.546061	$S = 1.0$ 24.398672	$S = 1.4$ 24.094372
25	$S = 1.0$ 24.557646	$S = 1.0$ 24.504702	$S = 1.0$ 24.249908	$S = 2.5$ 23.950974
100	$S = 1.0$ 24.402018	$S = 1.4$ 24.079903	$S = 2.5$ 23.811058	$S = 5.0$ 23.497025

TAB. 4.7 – Paramètres menant aux meilleurs résultats pour la séquence barbara pour chaque modèle

Nombre d'images				
$\sigma^2$	16	8	4	2
Modèle Tikhonov-Miller				
0	$\alpha = 0.001$ 28.457344	$\alpha = 0.001$ 28.438816	$\alpha = 0.001$ 28.446112	$\alpha = 0.001$ 28.444672
10	$\alpha = 0.001$ 28.318882	$\alpha = 0.001$ 28.221701	$\alpha = 0.001$ 28.031906	$\alpha = 0.001$ 27.772444
25	$\alpha = 0.001$ 28.161121	$\alpha = 0.001$ 27.924442	$\alpha = 0.001$ 27.532089	$\alpha = 0.001$ 27.015211
100	$\alpha = 0.001$ 27.411774	$\alpha = 0.001$ 26.673225	$\alpha = 0.01$ 26.122345	$\alpha = 0.01$ 25.672457
Modèle GGMRF				
0	$q = 1.2$ 28.384180	$q = 1.2$ 28.319418	$q = 1.2$ 28.256458	$q = 1.2$ 28.029957
10	$q = 1.2$ 28.301521	$q = 1.2$ 28.215546	$q = 1.2$ 28.039511	$q = 1.2$ 27.733757
25	$q = 1.2$ 28.221302	$q = 1.2$ 28.061558	$q = 1.2$ 27.770767	$q = 1.2$ 27.337212
100	$q = 1.2$ 27.761734	$q = 1.2$ 27.243050	$q = 1.5$ 26.637499	$q = 1.5$ 26.015757
Modèle médian				
0	$\eta = 20, \alpha = 0.5$ 28.585590	$\eta = 20, \alpha = 1.0$ 28.225533	$\eta = 5, \alpha = 0.5$ 27.967316	$\eta = 20, \alpha = 1.5$ 27.310928
10	$\eta = 20, \alpha = 1.0$ 28.329971	$\eta = 20, \alpha = 1.0$ 28.005953	$\eta = 5, \alpha = 0.5$ 27.661116	$\eta = 5, \alpha = 0.5$ 26.959843
25	$\eta = 20, \alpha = 1.0$ 28.150379	$\eta = 5, \alpha = 0.5$ 27.725286	$\eta = 5, \alpha = 0.5$ 27.299355	$\eta = 5, \alpha = 1.0$ 26.611012
100	$\eta = 20, \alpha = 1.5$ 27.399633	$\eta = 5, \alpha = 1.0$ 26.850067	$\eta = 5, \alpha = 1.0$ 26.353466	$\eta = 1, \alpha = 0.5$ 25.705730
Modèle basé sur une segmentation				
0	N/A 28.716127	N/A 28.387852	N/A 27.895126	N/A 26.893747
10	N/A 28.397297	N/A 27.909634	N/A 27.260454	N/A 26.301096
25	N/A 28.022253	N/A 27.376348	N/A 26.629660	N/A 25.669928
100	N/A 26.836494	N/A 25.695879	N/A 24.516804	N/A 23.375381
Modèle basé sur un seuillage de la DCT				
0	$S = 1.0$ 31.166519	$S = 1.0$ 31.071877	$S = 1.0$ 30.861071	$S = 1.2$ 29.368763
10	$S = 1.0$ 31.024054	$S = 1.0$ 30.798471	$S = 1.0$ 30.426809	$S = 1.4$ 28.940369
25	$S = 1.0$ 30.862019	$S = 1.0$ 30.496655	$S = 1.4$ 29.906010	$S = 2.5$ 28.560780
100	$S = 1.2$ 30.094307	$S = 2.5$ 29.250435	$S = 2.5$ 28.673500	$S = 5.0$ 26.739588

TAB. 4.8 – Paramètres menant aux meilleurs résultats pour la séquence f16 pour chaque modèle

Nombre d'images				
$\sigma^2$	16	8	4	2
Modèle Tikhonov-Miller				
0	$\alpha = 0.001$ 30.985537	$\alpha = 0.001$ 30.951593	$\alpha = 0.001$ 30.934334	$\alpha = 0.001$ 31.014078
10	$\alpha = 0.001$ 30.811649	$\alpha = 0.001$ 30.593082	$\alpha = 0.001$ 30.283863	$\alpha = 0.001$ 29.906473
25	$\alpha = 0.001$ 30.563898	$\alpha = 0.001$ 30.126646	$\alpha = 0.001$ 29.495260	$\alpha = 0.01$ 28.845232
100	$\alpha = 0.001$ 29.406681	$\alpha = 0.01$ 28.632294	$\alpha = 0.01$ 28.122492	$\alpha = 0.01$ 27.341642
Modèle GGMRF				
0	$q = 1.2$ 30.670698	$q = 1.2$ 30.650318	$q = 1.2$ 30.646984	$q = 1.2$ 30.736813
10	$q = 1.2$ 30.600662	$q = 1.2$ 30.495737	$q = 1.2$ 30.365173	$q = 1.2$ 30.235088
25	$q = 1.2$ 30.510103	$q = 1.2$ 30.297775	$q = 1.2$ 29.975246	$q = 1.2$ 29.551094
100	$q = 1.2$ 29.924536	$q = 1.2$ 29.236542	$q = 1.5$ 28.548607	$q = 1.5$ 27.663151
Modèle médian				
0	$\eta = 20, \alpha = 0.5$ 31.172525	$\eta = 20, \alpha = 0.5$ 30.969719	$\eta = 20, \alpha = 0.5$ 30.503426	$\eta = 20, \alpha = 1.0$ 30.066341
10	$\eta = 20, \alpha = 0.5$ 30.814817	$\eta = 20, \alpha = 1.0$ 30.356949	$\eta = 20, \alpha = 1.0$ 29.896147	$\eta = 20, \alpha = 1.5$ 29.461006
25	$\eta = 20, \alpha = 1.0$ 30.405144	$\eta = 20, \alpha = 1.0$ 29.914400	$\eta = 20, \alpha = 1.5$ 29.374271	$\eta = 5, \alpha = 1.0$ 28.750858
100	$\eta = 20, \alpha = 1.5$ 29.419533	$\eta = 5, \alpha = 1.0$ 28.722752	$\eta = 5, \alpha = 1.0$ 28.109024	$\eta = 1, \alpha = 0.5$ 27.591021
Modèle basé sur une segmentation				
0	N/A 30.784283	N/A 30.878975	N/A 30.485151	N/A 29.489670
10	N/A 30.564865	N/A 30.170229	N/A 29.470694	N/A 28.591263
25	N/A 30.243904	N/A 29.469696	N/A 28.496384	N/A 27.584455
100	N/A 28.631496	N/A 27.216877	N/A 25.696827	N/A 24.535444
Modèle basé sur un seuillage de la DCT				
0	$S = 1.0$ 32.925705	$S = 1.0$ 32.749470	$S = 1.0$ 32.612068	$S = 1.0$ 31.861307
10	$S = 1.0$ 32.709705	$S = 1.0$ 32.438019	$S = 1.0$ 32.085709	$S = 1.4$ 31.142033
25	$S = 1.0$ 32.532066	$S = 1.0$ 32.080151	$S = 1.2$ 31.367348	$S = 2.5$ 30.290194
100	$S = 1.2$ 31.543137	$S = 2.5$ 30.420776	$S = 2.5$ 29.801193	$S = 5.0$ 28.518902

TAB. 4.9 – Paramètres menant aux meilleurs résultats pour la séquence lena pour chaque modèle

baboon		
	Paramètres	RSBP
Modèle Tikhonov-Miller	$\alpha = 0.001$	21.120058
Modèle GGMRF	$q = 1.2$	21.023436
Modèle médian	$\eta = 20, \alpha = 0.5$	21.124304
Modèle basé sur une segmentation	N/A	21.149572
Modèle basé sur un seuillage de la DCT	$S = 1.0$	21.466831
barbara		
	Paramètres	RSBP
Modèle Tikhonov-Miller	$\alpha = 0.001$	23.750885
Modèle GGMRF	$q = 1.2$	23.812677
Modèle médian	$\eta = 20, \alpha = 1.5$	23.637018
Modèle basé sur une segmentation	N/A	23.565552
Modèle basé sur un seuillage de la DCT	$S = 1.4$	24.016088
f16		
	Paramètres	RSBP
Modèle Tikhonov-Miller	$\alpha = 0.001$	27.559324
Modèle GGMRF	$q = 1.2$	27.716410
Modèle médian	$\eta = 5, \alpha = 0.5$	27.234152
Modèle basé sur une segmentation	N/A	26.867756
Modèle basé sur un seuillage de la DCT	$S = 1.4$	29.339346
lena		
	Paramètres	RSBP
Modèle Tikhonov-Miller	$\alpha = 0.001$	29.638861
Modèle GGMRF	$q = 1.2$	29.912048
Modèle médian	$\eta = 20, \alpha = 1.5$	29.321898
Modèle basé sur une segmentation	N/A	28.894386
Modèle basé sur un seuillage de la DCT	$S = 1.4$	30.863478

TAB. 4.10 – Paramètres produisant les meilleurs résultats en moyenne pour chaque modèle, toutes variances et nombres d’images confondues

seuillage de la DCT parmi les meilleurs résultats. Cet état de fait se mesure quantitativement au niveau du RSBP des résultats ; il se perçoit aussi qualitativement au niveau des images. La seule exception à cette suprématie survient en présence d'images présentant un niveau de complexité supérieur, comme c'est le cas avec la séquence **baboon**. On constate au tableau 4.2 qu'en présence d'un bruit faible et d'un grand nombre d'images, le modèle TM présente le meilleur RSBP. Ce résultat, surprenant à prime abord, s'explique par le fait qu'un nombre important d'images dans la séquence tend à déterminer le problème : plus nombreuses sont les relations qu'il est possible d'établir entre les pixels BR et HR, plus les données elles-mêmes permettent d'arriver à la solution. En d'autres termes, plus le terme de fidélité aux données dans (3.18) prend de l'importance, au détriment de l'influence du modèle *a priori*. À l'autre extrémité du tableau 4.2, on retrouve le modèle GGMRF. Nous expliquons ceci par le fait que dans ces conditions, il devient difficile, voire impossible, de différencier le bruit dans la séquence des détails à reconstruire, et alors le flou opéré par une application du GGMRF avec une valeur élevée de  $q$  – et possiblement avec une application du modèle TM avec un  $\alpha$  supérieur à ce que nous avons utilisé dans nos tests – s'avère la meilleure solution.

Il est bon de noter que bien que le modèle DCT tende à l'emporter dans toutes les catégories, cet avantage repose sur une variation du seuil  $S$  en réponse à tout changement de condition. La figure 4.3 illustre bien ce constat : la valeur de  $S = 1.4$ , valeur donnant les meilleurs résultats en moyenne pour le modèle DCT tel que noté au tableau 4.10, présente la courbe descendante sur la diagonale (16 images,  $\sigma^2 = 0$ )-(2 images,  $\sigma^2 = 100$ ) la plus abrupte parmi tous les modèles. Fait également corroboré par les tableaux précédents, où l'on remarque que  $S$  tend à varier de 1.0 à 5.0 sur cette même diagonale. Il est donc logique de conclure que lorsque les circonstances permettent une réévaluation au cas par cas du seuil, le modèle basé sur un seuillage de la DCT constitue le meilleur choix parmi ceux étudiés dans ce mémoire. Dans le cas contraire, une étude plus approfondie pour déterminer deux ou trois valeurs de paramètres, chacune jugée plus adaptée à des circonstances de super-résolution changeantes, pourrait être salutaire.

Un dernier élément mérite d'être soulevé, et pour ce faire nous nous référons à la figure 4.10. Cette image présente une région d'intérêt de la séquence **barbara**, nommément le pan visible de la nappe qui est oblique au plan image. On y remarque une reconstruction du motif de la nappe d'une qualité très variable d'un modèle à l'autre. Premier élément à noter : le motif est à toutes fins pratiques absent de l'image BR, du moins est-il impossible à identifier autrement que par une espèce de quadrillé de pixels. Constat second : les reconstructions produites par les modèles TM, médian et basé sur une segmentation présentent un motif clairement identifiable. Par contre, le modèle GGMRF et celui basé sur un seuillage de la DCT échouent complètement ou en majeure partie à ce recouvrement. Clairement on assiste ici à la conséquence du mode de fonctionnement propre à chaque modèle. Le modèle TM avec une valeur faible de  $\alpha$  s'appuie principalement sur les données pour en venir à une image restaurée. Le modèle médian et celui basé sur une segmentation sont tous deux conçus avec la contrainte du maintien des arêtes de l'image, un élément crucial de leur influence. L'action du GGMRF par contre est moins explicite, et, on le voit ici, tend à estomper les détails fins en même temps que le bruit. Le même problème afflige le modèle DCT, qui travaille sur l'image dans le domaine fréquentiel. Un signal d'une amplitude similaire dans ce domaine au bruit et artefacts que l'on désire éliminer entraîne de toute évidence la perte de ce signal.

## CHAPITRE 5

### CONCLUSION

Depuis l'avènement de la photographie au XIX<sup>e</sup> siècle, les images ont pris une place toujours croissante dans tous les domaines. Avec l'explosion des applications comportant un élément de visuel depuis les quelques dernières décennies, et une place critique des images digitales au sein des divers champs de savoir humain, rien ne laisse présager une diminution de leur importance. Il est donc d'un intérêt évident de continuer à mettre au point de nouvelles méthodes qui nous permettent de manipuler et traiter les images qui contiennent l'information qui nous intéresse.

Dans ce mémoire, nous avons présenté l'une de ces techniques, la super-résolution, et nous avons argumenté en faveur de son efficacité afin de reconstruire des images haute-résolution à partir de séquence d'images de résolution moindre. Dans cette optique nous avons présenté trois modèles *a priori*, à être testés au sein d'un algorithme de reconstruction régularisée et comparés à deux autres modèles-témoins ayant déjà fait l'objet d'applications en traitement d'images, soient le modèle Tikhonov-Miller et le modèle GGMRF.

Nos résultats, issus d'une batterie de tests exhaustive portant sur un éventail de paramètres et de variables, ont démontré l'efficacité de chacun d'entre eux, avec des performances généralement supérieures à celles des modèles-témoins. Cet avantage s'est avéré être particulièrement favorable en ce qui a trait au modèle basé sur un seuillage de la DCT ; cependant, les modèles médian et basé sur une segmentation se sont montrés plus aptes à la conservation de détails périodiques fins.

Plusieurs voies visant l'amélioration de nos résultats s'offrent à nous. Dans un premier temps, le modèle d'imagerie utilisé dans notre implantation s'est révélé être un facteur limitatif à la qualité de nos résultats. Un modèle plus raffiné, notamment au niveau de la pondération des pixels HR en relation avec les pixels BR, serait salutaire. De plus, les excellentes performances du modèle basé sur un seuillage de la DCT, résultant en des images largement débruitées qui conservent

la structure originale de l'image permet d'envisager l'implantation d'un procédé de recalage raffiné de façon concurrente avec la super-résolution (voir section 3.1.1). Il serait également intéressant d'investiguer la possibilité de conjuguer les bonnes performances du modèle DCT avec l'action auxiliaire d'un autre modèle, choisi celui-là en raison de ses qualités de rétention des détails fins autrement occultés par le modèle DCT.

## BIBLIOGRAPHIE

- [1] R. C. GONZALEZ & R. E. WOODS, *Digital Image Processing*, Prentice Hall, seconde éd., 2002.
- [2] S. C. PARK, M. K. PARK & M. G. KANG, “Super-Resolution Image Reconstruction : A Technical Overview”, *IEEE Signal Processing Magazine*, vol. 20, n° 3, mai 2003, p. 21-36.
- [3] T. KOMATSU, K. AIZAWA, T. IGARASHI & T. SAITO, “Signal-processing based method for acquiring very high resolution images with multiple cameras and its theoretical analysis”, *IEEE Proceedings*, vol. 140, n° 1, février 1993, p. 19-25.
- [4] H. BENSON, *University Physics*, John Wiley & Sons, Inc., 1991.
- [5] *Charge-Coupled Device (CCD) Image Sensors*, Eastman Kodak Company, révision n° 1, mai 2001.
- [6] W.-H. CHAN, “CCD Camera Design and ASIC Applications”, *Proceedings of the Eighth Annual IEEE International ASIC Conference*, septembre 1995, p. 403-408.
- [7] E. TRUCCO & A. VERRI, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, 1998.
- [8] P. VAN WALREE, *Photographic optics*, <http://www.vanwalree.com/optics.html>, dernière édition le 21 janvier 2007.
- [9] K. SAYOOD, *Introduction to Data Compression*, Morgan Kaufmann, seconde éd., 2000.
- [10] D. S. EBERT, F. K. MUSGRAVE, D. PEACHY, K. PERLIN & S. WORLEY, *Texture & Modeling, A Procedural Approach*, Morgan Kaufmann, troisième éd., 2003.
- [11] R. C. HARDIE, K. J. BARNARD & E. E. ARMSTRONG, “Joint MAP Registration and High-Resolution Image Estimation Using a Sequence of Un-

- dersampled Images”, *IEEE Transactions of Image Processing*, vol. 6, n° 12, décembre 1997, p. 1621-1633.
- [12] M. K. NG & N. K. BOSE, “Mathematical Analysis of Super-Resolution Methodology”, *IEEE Signal Processing Magazine*, vol. 20, n° 3, mai 2003, p. 62-74.
- [13] D. CAPEL & A. ZISSERMAN, “Computer Vision Applied to Super Resolution”, *IEEE Signal Processing Magazine*, vol. 20, n° 3, mai 2003, p. 75-86.
- [14] I.-T. HSIAO, A. RANGARAJAN & G. GINDI, “A New Convex Edge-Preserving Median Prior with Applications to Tomography”, *IEEE Transactions on Medical Imaging*, vol. 22, n° 5, mai 2003, p. 580-585.
- [15] M. MIGNOTTE, “An adaptive segmentation-based regularization term for image restoration”, *12th IEEE International Conference on Image Processing, ICIP’05*, vol. 1, septembre 2005, p. 901-904.
- [16] M. MIGNOTTE, “A Segmentation-Based Regularization Term for Image Deconvolution”, *IEEE Transactions on Image Processing*, vol. 15, n° 7, juillet 2006, p. 1973-1984.
- [17] S. FARSIU, D. ROBINSON, M. ELAD & P. MILANFAR, “Fast and Robust Multiframe Super Resolution”, *IEEE Transactions on Image Processing*, vol. 13, n° 10, octobre 2004, p. 1327-1344.
- [18] Z. LIN & H.-Y. SHUM, “Fundamental Limits of Reconstruction-Based Super-resolution Algorithms under Local Translation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, n° 1, janvier 2004, p. 83-97.
- [19] D. ROBINSON & P. MILANFAR, “Statistical Performance Analysis of Super-Resolution”, *IEEE Transactions on Image Processing*, vol. 15, n° 6, juin 2006, p. 1413-1428.
- [20] L. GOTTESFELD BROWN, “A Survey of Image Registration Techniques”, *ACM Computing Surveys*, vol. 24, n° 4, décembre 1992, p. 326-376.
- [21] B. ZITOVA & J. FLUSSER, “Image Registration Methods : a Survey”, *Image and Vision Computing*, vol. 21, 2003, p. 977-1000.

- [22] R. R. SCHULTZ & R. L. STEVENSON, "Extraction of High-Resolution Frames from Video Sequences", *IEEE Transactions on Image Processing*, vol. 5, n° 6, juin 1996, p. 996-1011.
- [23] S. BAKER & T. KANADE, "Limits on Super-Resolution and How to Break Them", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, n° 9, septembre 2002, p. 1167-1183.
- [24] A. K. KATSAGGELOS & K. T. LAY, "Maximum Likelihood Blur Identification and Image Restoration Using the EM Algorithm", *IEEE Transactions on Signal Processing*, vol. 39, n° 3, mars 1991, p. 729-733.
- [25] R. L. LAGENDIJK, J. BIEMOND & D. E. BOEKEE, "Hierarchical Blur Identification", *1990 International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, 1990, p. 1889-1892.
- [26] S. S. MAKHANOV, S. VANNAJRAIROJU & S. KONDO, "A New Blur Identification Scheme", *TENCON 99. Proceedings of the IEEE Region 10 Conference*, vol. 2, décembre 1999, p. 1315-1318.
- [27] G. B. GIANNAKIS & R. W. HEATH JR., "Blind Identification of Multichannel FIR Blurs and Perfect Image Restoration", *IEEE Transactions on Image Processing*, vol. 9, n° 11, novembre 2000, p. 1877-1896.
- [28] D. KUNDUR & D. HATZINAKOS, "Blind Image Deconvolution", *IEEE Signal Processing Magazine*, mai 1996, p. 43-64.
- [29] D. KUNDUR & D. HATZINAKOS, "Blind Image Deconvolution Revisited", *IEEE Signal Processing Magazine*, novembre 1996, p. 61-63.
- [30] Y. HE, K.-H. YAP, L. CHEN & L.-P. CHAU, "Blind Super-Resolution Image Reconstruction Using a Maximum a Posteriori Estimation", *2006 IEEE International Conference on Image Processing*, octobre 2006, p. 1729-1732.
- [31] M. ELAD & A. FEUER, "Restoration of a Single Superresolution Image from Several Blurred, Noisy, and Undersampled Measured Images", *IEEE Transactions on Image Processing*, vol. 6, n° 12, décembre 1997, p. 1646-1658.

- [32] Z. WANG & F. QI, “On Ambiguities in Super-Resolution Modeling”, *IEEE Signal Processing Letters*, vol. 11, n° 8, août 2004, p. 678-681.
- [33] M. BERTERO, T. A. POGGIO & V. TORRE, “Ill-Posed Problems in Early Vision”, *Proceedings of the IEEE*, vol. 76, n° 8, août 1988, p. 869-889.
- [34] A. NEUMAIER, “Solving Ill-Conditioned and Singular Linear Systems : A Tutorial on Regularization”, *SIAM Review*, vol. 40, n° 3, 1998, p. 636-666.
- [35] J. CLARK, M. PALMER & P. LAWRENCE, “A Transformation Method for the Reconstruction of Functions from Nonuniformly Spaced Samples”, *IEEE Transactions on Signal Processing*, vol. 33, n° 5, octobre 1985, p. 1151-1165.
- [36] S. LETTRATTANAPANICH & N. K. BOSE, “High Resolution Image Formation from Low Resolution Frames Using Delaunay Triangulation”, *IEEE Transactions on Image Processing*, vol. 11, n° 12, décembre 2002, p. 1427-1441.
- [37] C. BOUMAN & K. SAUER, “A Generalized Gaussian Image Model for Edge-Preserving MAP Estimation”, *IEEE Transactions on Image Processing*, vol. 2, n° 3, juillet 1993, p. 296-309.
- [38] M. IRANI & S. PELEG, “Super Resolution from Image Sequences”, *International Conference on Pattern Recognition*, vol. 90, p.115-120.
- [39] B. K. GUNTURK, A. Z. BATUR, Y. ALTUNBASAK, M. H. HAYES & R. M. MERSEREAU, “Eigenface-Domain Super-Resolution for Face Recognition”, *IEEE Transactions on Image Processing*, vol. 12, n° 5, mai 2003.
- [40] P. CHEESEMAN, B. KANEFSKY, R. KRAFT & J. STUTZ, “Super-Resolved Surface Reconstruction From Multiple Images”, *Maximum Entropy and Bayesian Methods*, Kluwer Academic Publishers, 1996, p. 293-308.
- [41] M. BEN-EZRA, A. ZOMET & S. K. NAYAR, “Video Super-Resolution Using Controlled Subpixel Detector Shifts”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, n° 6, juin 2005.
- [42] C. A. SEGALL, A. K. KATSAGGELOS, R. MOLINA & J. MATEOS, “Bayesian Resolution Enhancement of Compressed Video”, *IEEE Transactions on Image Processing*, vol. 13, n° 7, juillet 2004.

- [43] M. V. W. ZIBETTI & J. MAYER, “Simultaneous Super-Resolution for Video Sequences”, IEEE International Conference on Image Processing 2005, vol. 1.
- [44] E. SHECHTIMAN, Y. CASPI & M. IRANI, “Space-Time Super-Resolution”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, n° 4, avril 2005.
- [45] D. CAPEL & A. ZISSERMAN, “Super-Resolution Enhancement of Text Image Sequences”, 15th International Conference on Pattern Recognition 2000 Proceedings, vol. 1, p. 600-605.
- [46] B. K. GUNTURK, Y. ALTUNBASAK & R. M. MERSEREAU, “Super-Resolution Reconstruction of Compressed Video Using Transform-Domain Statistics”, IEEE Transactions on Image Processing, vol. 13, n° 1, janvier 2004, p. 33-43.
- [47] S. GEMAN & D. GEMAN, “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images”, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 6, n° 6, novembre 1984, p. 721-741.
- [48] S. ALENIUS & U. RUOTSALAINEN, “Bayesian Image Reconstruction for Emission Tomography Based on Median Root Prior”, *European Journal of Nuclear Medicine*, vol. 24, n° 3, mars 1997, p. 258-265.
- [49] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING & B. P. FLANNERY, *Numerical Recipes in C : The Art of Scientific Computing*, Cambridge University Press, 1992.
- [50] M. MIGNOTTE, *Segmentation d’images sonar par approche Markovienne hiérarchique non-supervisée et classification d’ombres portées par modèles statistiques*, thèse doctorale défendue en juillet 1998, École Navale, Brest, France.
- [51] O. G. GULERYUZ, “Weighted Overcomplete Denoising”, *Proc. Asilomar Conf. on Signals and Systems*, novembre 2003.
- [52] R. COIFMAN & D. DONOHO, “Translation Invariant De-noising”, *Lecture Notes in Statistics : Wavelets and Statistics*, 1995, p. 125-150.

- [53] C. TASWELL, “Experiments in Wavelet Shrinkage Denoising”, *Journal of Computational Methods in Science and Engineering*, vol. 1, n° 2-3, 2001, p. 315-326.
- [54] L. KAUR, S. GUPTA & R. C. CHAUGAN, “Image Denoising using Wavelet Thresholding”, <http://citeseer.ist.psu.edu/556198.html>.
- [55] S. MARUSIC, G. DENG, D. B. H. TAY, “Image Denoising Using Over-Complete Wavelet Representations”, *Proceedings of the European Signal Processing Conference, EUSIPCO 2005*, 2005.
- [56] D. L. DONOHO, “De-Noising by Soft-Thresholding”, *IEEE Transactions on Information Theory*, vol. 41, n° 3, 1995, p. 613-627.