



HAL
open science

Agrandissement d'images par synthèse de similarités et par induction sur un ensemble

Didier Calle

► **To cite this version:**

Didier Calle. Agrandissement d'images par synthèse de similarités et par induction sur un ensemble. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 1999. Français. NNT : . tel-00004813

HAL Id: tel-00004813

<https://theses.hal.science/tel-00004813>

Submitted on 18 Feb 2004

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

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

THÈSE

présentée par

Didier Calle

pour obtenir le titre de
DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER - GRENOBLE 1

Spécialité : Informatique

Agrandissement d'images par synthèse
de similarités et par induction sur un ensemble

Date de soutenance : 25 Novembre 1999

Composition du Jury :

Alain Chéhikian	PRESIDENT
Michel Barlaud	RAPPORTEUR
Henri Maître	RAPPORTEUR
Franck Davoine	EXAMINATEUR
Annick Montanvert	EXAMINATEUR (Directrice de thèse)
Bernard Rougé	EXAMINATEUR

Thèse préparée au sein du Laboratoire **TIMC - Institut IMAG**

à ma famille
à Karen

Remerciements

Je tiens à remercier Monsieur Alain Chéhikian, Professeur à l'Université Joseph Fourier de Grenoble, pour l'honneur qu'il m'a fait en présidant mon jury de thèse.

Je remercie vivement Monsieur Michel Barlaud, Professeur à l'Université de Nice-Sophia Antipolis, ainsi que Monsieur Henri Maître, Professeur à l'ENST Paris, pour avoir accepté la responsabilité d'être rapporteurs de ma thèse. Je les remercie également pour l'intérêt qu'ils ont manifesté à l'égard de mes recherches.

Je remercie chaleureusement Monsieur Franck Davoine, Maître de Conférence à l'Université Technologique de Compiègne, pour le plaisir qu'il m'a fait en acceptant d'être membre de mon jury.

Je remercie Monsieur Bernard Rougé, chercheur au CNES de Toulouse, pour avoir accepté d'être membre de mon jury et avoir apporté ses compétences en restauration d'images.

J'exprime ma profonde reconnaissance à Madame Annick Montanvert pour avoir dirigé mon travail durant ces trois années malgré ses nombreuses responsabilités. Je tiens à la remercier pour sa bonne humeur constante et à lui témoigner toute mon amitié.

Cette thèse a été réalisée dans l'équipe INFODIS dirigée par Monsieur Jean-Marc Chassery, Directeur de Recherche au CNRS. Je le remercie très sincèrement pour ses contributions au développement et à la diffusion de mes recherches.

Je tiens également à exprimer ma sympathie aux personnes que j'ai pu côtoyer lors de ma thèse et qui ont contribué à rendre agréable mon séjour à Grenoble.

Enfin, j'exprime ma plus profonde amitié à Fabrice et à Myriam pour les bons moments passés ensemble.

Résumé

Ce mémoire porte sur l'agrandissement des images numériques fixes en niveaux de gris dans un contexte général sans connaissance a priori. Il est constitué de trois parties. La première porte sur une description détaillée des méthodes d'agrandissement que l'on peut trouver dans la littérature. Nous commençons par présenter les méthodes d'interpolation classiques ayant pour objectif de préserver les fréquences de l'image à agrandir, puis nous détaillons des méthodes récentes de préservation structurelle produisant une meilleure netteté. La deuxième partie constitue la contribution majeure de ce travail en proposant deux nouvelles méthodes d'agrandissement. La première méthode est basée sur la synthèse de similarités détectées sur une représentation pyramidale de l'image. Elle reprend à la base le zoom fractal classique en apportant de nombreuses modifications et améliorations aussi bien dans la phase d'analyse que dans celle de synthèse. Nous vérifions expérimentalement l'hypothèse de préservation des similarités. La deuxième méthode d'agrandissement que nous proposons s'intéresse à l'ensemble admissible des images agrandies d'une image initiale. La condition d'admissibilité repose ici sur la notion de réduction : une image agrandie appartient à l'ensemble des solutions si sa réduction est identique à l'image initiale. Nous étudions différents algorithmes de projection sur cet ensemble. La troisième partie concerne des améliorations et des applications de nos deux méthodes. Tout d'abord, nous améliorons la qualité de l'image agrandie par synthèse de similarités en recherchant celles-ci sur une pyramide en quinconce. Ensuite, nous exploitons la méthode d'agrandissement par induction pour régulariser, vis-à-vis de la contrainte de réduction, les images agrandies par synthèse de similarités. Enfin, nous exploitons également cette méthode pour réaliser un codage hiérarchique de l'image permettant sa transmission progressive sur réseau.

Abstract

This thesis deals with the enlargement of still digital images in a general context. It is made of three parts. The first one is a detailed description of the enlargement methods which can be found in the literature. First of all, we present classic interpolation methods which aim at retaining the frequencies of the image to be enlarged. Then, we detail more recent methods which retain the structures of images and give better results. The second part is the main contribution of this work, as it presents two original enlargement methods. The first one synthesizes similarities detected on a pyramidal representation of the image, and it modifies and improves both the analysis and the synthesis phases of the classical fractal zoom. In this part, the assumption of the similarities retaining is experimentally checked. The second method which is here presented deals with the admissible set of enlarged images of an original image. This admissibility is closely linked to the reduction of the enlarged image, i.e, an image belongs to the solutions set only if its reduction is identical to the original image. We study different algorithms to project onto this set. In the third part, we present several ways of improving and applying our methods. First, we improve the quality of an image enlarged thanks to the similarities synthesis method: to do so, we detect these similarities on a quincunx pyramid. Then, we use the induction enlarging method to regularize in reduction the images enlarged with the similarities synthesis method. We eventually use the induction enlarging method to obtain a hierarchical encoding of the image which allows us to make its progressive transmission.

Table des matières

1	Introduction générale	1
2	Méthodes d'agrandissement préservant les fréquences	7
2.1	Introduction	7
2.2	Notions sur la transformée de Fourier	8
2.2.1	La transformée de Fourier	8
2.2.2	Les propriétés de la transformée de Fourier	9
2.2.3	L'analyse spectrale	10
2.3	La préservation fréquentielle	12
2.3.1	La méthode du <i>zero-padding</i>	12
2.3.2	Interpolation avec des fonctions polynomiales	17
2.3.3	Interpolation avec la transformée B-spline	22
2.4	Amélioration de la netteté	28
2.4.1	Utilisation d'une équation différentielle partielle	28
2.4.2	Diffusion anisotropique	30
2.5	Conclusion	32
3	Méthodes d'agrandissement préservant les structures	35
3.1	Introduction	35
3.2	L'interpolation adaptative	36
3.2.1	Bloc homogène et bloc à contour simple	39
3.2.2	Autres types de blocs	41
3.2.3	Améliorations globales de l'interpolation adaptative	42
3.3	L'extrapolation fréquentielle	43
3.3.1	La transformée en ondelettes	43
3.3.2	La régularité et les ondelettes	44
3.3.3	La détection et la synthèse des singularités	46
3.4	Le zoom fractal	48
3.4.1	Transformation fractale d'une image	49
3.4.2	Agrandissement avec un code fractal	51
3.5	Autres méthodes	53
3.5.1	Interpolation avec un filtre médian	53
3.5.2	Interpolation avec un filtre de Wiener	54

3.5.3	Interpolation avec un modèle probabiliste	55
3.5.4	Interpolation basée sur la variation totale	56
3.6	Mesure de la qualité de l'agrandissement	58
3.7	Conclusion	61
4	Réduction et analyse pyramidale d'une image	63
4.1	Introduction	63
4.2	Analyse fréquentielle d'une réduction	64
4.2.1	Décimation spatiale et décimation fréquentielle	64
4.2.2	Réduction de facteur 2 d'un maillage carré	65
4.2.3	Réduction d'un maillage carré à un maillage en quinconce	68
4.2.4	Réduction d'un maillage en quinconce à un maillage carré	71
4.3	Analyse pyramidale	74
4.3.1	La pyramide de Meer	76
4.3.2	La pyramide de Burt	77
4.3.3	La pyramide de Chin	80
4.3.4	Comparaison expérimentale	82
4.3.5	Pyramide de différences	84
4.4	Pyramide en quinconce	86
4.4.1	Construction d'un filtre paramétré	87
4.4.2	Contraintes fréquentielles	89
4.4.3	Mise en œuvre	90
4.4.4	Pyramide de différences	90
4.5	Conclusion	92
5	Aggrandissement par synthèse de similarités sur une pyramide	95
5.1	Introduction	95
5.2	Généralités sur la méthode	96
5.2.1	Le principe de base	96
5.2.2	Algorithme général	98
5.3	La mesure de similarité entre deux blocs	103
5.3.1	Le modèle de similarité	103
5.3.2	Calcul des coefficients de la transformation massique	105
5.3.3	Exemples	105
5.4	Construction et organisation primaire d'un domaine de recherche	107
5.4.1	La sélection et l'ordre des blocs	108
5.4.2	La sélection avec une pyramide de différences	109
5.5	Classification par la mesure d'angles entre les espaces de transformation	111
5.5.1	Principe général de la classification	111
5.5.2	La mesure d'angle entre deux espaces de transformation	114
5.5.3	Un algorithme mesurant l'angle entre deux plans de transformation	115
5.5.4	Mise en œuvre	119
5.6	La recherche d'une similarité dans le domaine	121

5.6.1	Conditions pour exécuter une recherche	121
5.6.2	La recherche globale	122
5.6.3	La recherche locale	124
5.7	Agrandissement par synthèse des similarités	125
5.7.1	Agrandissement de facteur deux	126
5.7.2	Agrandissement de facteur supérieur à deux	130
5.8	Résultats expérimentaux	131
5.8.1	Configuration standard des paramètres	132
5.8.2	Etude des paramètres	134
5.8.3	Validation du modèle	148
5.8.4	Agrandissement de facteur supérieur à deux	156
5.8.5	Comparaisons avec d'autres méthodes d'agrandissement	157
5.9	Conclusion	160
6	Agrandissement par induction sur un ensemble	165
6.1	Introduction	165
6.2	Formulation de l'induction	168
6.2.1	Généralités	168
6.2.2	La réduction d'une image	168
6.2.3	Caractérisation de l'ensemble induit d'une réduction	169
6.2.4	Projection sur l'ensemble induit d'une réduction	172
6.2.5	Interprétation fréquentielle	173
6.3	Les méthodes de projections partielles	175
6.3.1	Les projections élémentaires	176
6.3.2	Projections alternées <i>vs</i> projections parallèles	177
6.3.3	Projection approchée par des projections partielles alternées	177
6.3.4	Projection approchée par des projections partielles parallèles	179
6.3.5	Projection approchée par des projections partielles parallèles avec extrapolation	183
6.3.6	Projection exacte par des projections partielles alternées	186
6.3.7	Projection exacte par des projections partielles parallèles	187
6.4	Etude expérimentale des méthodes de projections partielles	189
6.4.1	Induction sur un ensemble consistant	190
6.4.2	Induction sur un ensemble inconsistant	196
6.4.3	Induction sans connaissance <i>a priori</i>	200
6.4.4	Résumé de l'étude	201
6.5	Induction avec un facteur d'agrandissement supérieur à deux	202
6.6	Conclusion	205
7	Améliorations et applications	207
7.1	Introduction	207
7.2	Synthèse de similarités sur une pyramide en quinconce	208
7.2.1	Particularités	209
7.2.2	Résultats avec une image de référence	212

7.2.3	Résultats sans image de référence	218
7.3	Régularisation d'agrandissements par induction	228
7.3.1	Résultats avec une image de référence	228
7.3.2	Résultats sans image de référence	231
7.4	Compression réversible et transmission progressive d'images	231
7.4.1	La pyramide laplacienne et ses améliorations	233
7.4.2	Amélioration avec le principe d'induction	237
7.4.3	Résultats	239
7.5	Conclusion	246
8	Conclusion générale	249
	Bibliographie	255

Chapitre 1

Introduction générale

L'agrandissement d'images numériques est une opération courante, nécessaire et utile dans de nombreuses applications liées à l'imagerie. Tout d'abord, on peut avoir besoin d'agrandir une image pour accéder à ses détails, comme par exemple dans le domaine de l'infographie en général ou dans celui de l'imagerie médicale. Les photocopieurs sont également souvent équipés d'une fonction d'agrandissement. L'agrandissement peut aussi être employé pour augmenter la résolution de l'image sur un même support spatial afin d'accroître le confort visuel de l'observateur. On pensera en particulier aux imprimantes ou encore à la conversion entre le format américain NTSC et le format haute définition HDTV. L'agrandissement d'images peut aussi être utile pour dépasser les limites technologiques des systèmes d'acquisition, comme par exemple en imagerie satellitaire. Il est également nécessaire dans des domaines comme l'imagerie de synthèse pour le placage de textures sur des surfaces mathématiques (*warping*) ou le mosaïquage de photos. L'agrandissement d'images peut également s'avérer très utile pour réaliser une compression et une transmission progressive de l'image.

Il existe différents contextes de l'agrandissement en fonction du type d'image à agrandir ou des connaissances que l'on a sur celle-ci. En effet, il peut s'agir d'une image binaire [54], en niveaux de gris ou en couleurs [66]. Celle-ci peut également être représentée sur deux dimensions ou sur trois dimensions, comme c'est le cas notamment en tomographie [60]. Par ailleurs, on peut disposer d'informations multiples sur le système d'acquisition ayant servi à numériser l'image que l'on souhaite agrandir. Dans le contexte de la vidéo, on peut également exploiter une séquence d'images pour obtenir une seule image haute-résolution. Ceci permet d'obtenir de l'information au-delà de la bande passante du système vidéo, et l'on parle dans ce cas de *super-résolution* [102], [125]. Le contexte peut également être celui d'une application temps réel comme par exemple la conversion NTSC-HDTV : dans ce cas, l'agrandissement est câblé et la méthode utilisée doit être très rapide [81].

Selon le type d'image et les connaissances que l'on possède, la méthodologie de l'agrandissement diffère. Dans le cadre de cette thèse, nous nous intéressons uniquement aux images numériques fixes en niveaux de gris et représentées sur deux

dimensions. De plus, nous ne disposons d'aucune connaissance sur le système d'acquisition¹. En dehors de l'image à agrandir, nous n'exploitons aucune information extérieure. Le contexte adopté se veut donc très général.

L'agrandissement d'une image numérique consiste à augmenter le nombre total des échantillons. Dans le contexte adopté, les valeurs des échantillons de l'image agrandie sont estimées en considérant celles des échantillons initiaux. Les méthodes d'agrandissement se répartissent alors en deux grandes catégories : les méthodes de préservation fréquentielle et les méthodes de préservation structurelle.

Les méthodes de préservation fréquentielle réalisent une interpolation de l'image initiale en exploitant le théorème d'échantillonnage de Shannon [126], [127]. Plus précisément, elles font l'hypothèse que l'acquisition numérique de l'image à agrandir a été réalisée avec une fréquence d'échantillonnage deux fois supérieure à la plus haute fréquence contenue dans l'image continue (fréquence de Nyquist). Dans ce cas, le théorème de Shannon stipule que l'image continue peut être reconstruite exactement à partir de l'image numérique. La méthode théorique permettant de reconstruire l'image continue est la méthode du *zero-padding* [148]. Elle consiste à placer dans le spectre fréquentiel de l'image des valeurs nulles au-delà de la fréquence de Nyquist. En pratique, l'image agrandie obtenue avec cette méthode présente un effet d'ondulation visuellement gênant dû au phénomène de Gibbs. En outre, elle nécessite l'utilisation d'une transformée de Fourier directe pour obtenir le spectre fréquentiel et pouvoir introduire les zéros, puis une transformée de Fourier inverse pour obtenir le résultat de l'agrandissement. De nombreuses autres méthodes ont été proposées dans la littérature, ayant pour objectif de diminuer à la fois l'effet d'ondulation et la complexité algorithmique de la méthode du *zero-padding*. Parmi celle-ci, la méthode la plus convaincante est celle de la transformée B-spline qui unifie les méthodes d'interpolation sur une base mathématique très solide [136].

Les méthodes de préservation structurelle exploitent des caractéristiques communes aux images naturelles. Une caractéristique exploitée tout particulièrement est celle de l'invariance en résolution des contours représentant des discontinuités [70](p.154). Ainsi, l'objectif de la plupart des méthodes est de préserver la qualité des contours afin de maintenir la netteté de l'image. Malheureusement, les contours présents dans une image ne sont pas toujours l'expression d'une discontinuité. En effet, certains contours peuvent être l'expression d'une déformation locale sur un objet. Le critère de qualité d'une méthode de préservation structurelle porte donc notamment sur sa capacité à distinguer les différents types de contours. La méthode d'interpolation adaptative fait partie des méthodes les plus utilisées [46], [132]. En présence d'un contour, elle réalise une interpolation directionnelle. Par contre, en l'absence de contours, elle réalise une interpolation classique. Elle considère que tous les contours détectés sont des discontinuités parfaites et l'image agrandie n'est généralement pas très réaliste. L'analyse réalisée par les méthodes d'extrapolation

1. Nous verrons dans la conclusion générale de ce mémoire que les méthodes d'agrandissement que nous proposons peuvent facilement prendre en compte de l'information liée au système d'acquisition.

fréquentielle est beaucoup plus probante [29], [33]. En exploitant une représentation espace-échelle de l'image obtenue par une transformée en ondelettes [91], ces méthodes distinguent plus facilement les discontinuités véritables, notamment à travers les maxima d'ondelettes. En marge de ces méthodes, on trouve la méthode du *zoom fractal* [55], [106]. Son originalité est de préserver les structures en préservant les similarités entre les structures à des résolutions différentes.

Les méthodes d'agrandissement par préservation fréquentielle sont adaptées lorsque les conditions du théorème de Shannon sont vérifiées. Malheureusement, ces conditions sont très rarement satisfaites pour des images naturelles. En effet, la fréquence d'échantillonnage est très rarement suffisante en raison de la limitation technologique des systèmes d'acquisition. De plus, les scènes naturelles contiennent très souvent des discontinuités qui repoussent la fréquence de Nyquist à l'infini et rendent caduc le théorème de Shannon. Ainsi, les méthodes d'agrandissement par préservation fréquentielle génèrent un effet de flou assez caractéristique, notamment sur les contours. L'absence de hautes fréquences ne permet ni de maintenir la qualité des contours, ni de reconstituer les discontinuités.

En général, il est donc préférable d'employer des méthodes de préservation structurelle pour agrandir une image naturelle. En introduisant de nouvelles fréquences au niveau des contours, elles permettent de maintenir une certaine netteté de l'image. Toute la difficulté est de distinguer les contours représentant des discontinuités de ceux représentant des déformations continues. Si la résolution de l'image originale n'est pas suffisante, il n'est pas possible de certifier qu'un contour traduit une véritable discontinuité. Il faudrait pour cela réaliser une interprétation sémantique du contenu de l'image, et cela dépasse largement les possibilités actuelles. En revanche, il est souvent possible de distinguer des contours qui expriment des déformations d'objets en réalisant une analyse multi-résolution.

Nous présentons dans le cadre de cette thèse deux méthodes d'agrandissement complémentaires. La première est basée sur la synthèse de similarités détectées sur une représentation pyramidale de l'image et la deuxième réalise une projection sur un ensemble d'images agrandies qui vérifient la contrainte de réduction.

Notre première méthode peut être considérée comme un approfondissement du zoom fractal [26]. L'intérêt de celui-ci est de préserver les structures de manière implicite en cherchant à préserver les similarités entre ces dernières. Le zoom fractal que l'on trouve dans la littérature exploite directement le code de similarités produit par les méthodes de compression fractale [47], [73]. Or les exigences en compression et en agrandissement sont plutôt opposées. Dans notre approche, nous reprenons à la base le principe de recherche et de préservation des similarités en considérant avant tout les exigences de l'agrandissement. Nous apportons de nombreuses modifications et améliorations au zoom fractal classique aussi bien dans la phase d'analyse que dans celle de synthèse des similarités. L'hypothèse de préservation des similarités qui est à la base de la méthode du zoom fractal n'a jamais été vérifiée dans la littérature. Nous l'étudions de manière approfondie dans le cadre de cette thèse.

Dans notre deuxième méthode, le problème de l'agrandissement d'une image

est posé comme un problème inverse de réduction [27], [28]. Les solutions admissibles d'un agrandissement sont généralement multiples. Il semble donc assez naturel d'aborder l'agrandissement d'une image avec une approche ensembliste. Une première contrainte assez simple que doit vérifier l'ensemble des solutions d'un agrandissement est celle de la réduction. Plus précisément, la réduction de l'image agrandie doit aboutir à une image équivalente à celle de départ. Une image agrandie appartenant à l'ensemble des solutions peut finalement être obtenue en réalisant la projection d'une image quelconque sur cet ensemble. Dans le cadre de cette thèse, l'ensemble des solutions est soumis uniquement à la contrainte de réduction. Remarquons que cette approche de l'agrandissement par la résolution d'un problème inverse a des points communs avec certaines méthodes de super-résolution produisant une image haute-résolution à partir d'une séquence d'images basse-résolution [102]. Dans ces méthodes, l'image haute-résolution est obtenue par projection sur des contraintes de réduction associées aux images basse-résolution. Toutefois, dans ce cadre, des notions supplémentaires sont introduites, comme celle du mouvement de la caméra. Dans notre approche, nous étudions divers algorithmes de projection sur l'ensemble des solutions qui sont issus de la théorie d'estimation des ensembles convexes [39]. Certains d'entre eux permettent d'obtenir une projection exacte. Les algorithmes de projection utilisés dans les méthodes classiques de super-résolution permettent seulement d'obtenir une projection approchée (non-orthogonale).

Notre deuxième méthode d'agrandissement se veut complémentaire de notre première méthode, et plus généralement des autres méthodes de préservation structurelle. En effet, ces méthodes génèrent une image agrandie qui ne respecte généralement pas la contrainte de réduction. On peut régulariser cette image en la projetant sur l'ensemble des solutions vérifiant le problème inverse de réduction. En outre, cette seconde méthode permet de faire le lien avec les méthodes de préservation fréquentielle. En effet, la projection permet de restaurer les fréquences de l'image originale qui sont modifiées lors du processus d'agrandissement. En d'autres termes, l'image issue de la projection préserve les fréquences de l'image originale tout en préservant les hautes fréquences produites par les méthodes de préservation structurelle.

L'organisation générale du mémoire est décrite ci-dessous.

Le chapitre 2 est consacré à la présentation des méthodes d'agrandissement par préservation fréquentielle. La méthode d'interpolation par transformée B-spline sera utilisée au cours de ce mémoire comme la méthode de référence de ce chapitre. Nous l'exploiterons en particulier dans notre approche de l'agrandissement par synthèse de similarités pour interpoler les blocs non synthétisés ainsi que les résidus de similarités lors de la synthèse (chapitre 5). Cette méthode fera également l'objet d'une comparaison avec notre propre approche.

Le chapitre 3 décrit les principales méthodes d'agrandissement préservant les structures : l'interpolation adaptative, l'extrapolation fréquentielle et le zoom fractal. Cette dernière méthode est celle qui a inspiré notre méthode de synthèse de similarités. Elle constituera donc la méthode de référence à notre approche. En

marge de ces trois méthodes, nous en présentons également d'autres, dont le filtrage médian qui sera utilisé pour comparer nos résultats dans le chapitre 5.

Dans le chapitre 4, nous étudions l'opération de réduction d'une image d'un point de vue fréquentiel, que ce soit avec un facteur de réduction 2 ou $\sqrt{2}$. Nous présentons également diverses approches proposées dans la littérature pour construire une pyramide passe-bas à facteur de réduction 2. Nous étudions également des pyramides de différences construites à partir des pyramides passe-bas précédentes. Enfin, nous proposons un algorithme de construction rapide de pyramides passe-bas et de différences en quinconce (facteur de réduction $\sqrt{2}$). L'opération de réduction sera exploitée dans notre méthode d'agrandissement par induction sur un ensemble (chapitre 6). Les pyramides à facteur de réduction 2 seront utilisées dans le chapitre 5, et celles en quinconce seront exploitées dans le chapitre 7.

Le chapitre 5 présente notre approche de l'agrandissement par synthèse de similarités sur une pyramide à facteur de réduction 2. Nous apportons diverses contributions dans les phases d'analyse (recherche des similarités) et de synthèse. Nous donnons de nombreux résultats expérimentaux illustrant les divers paramètres de notre méthode. Nous vérifions également l'hypothèse de préservation des similarités à deux résolutions différentes. Enfin, nous comparons des images agrandies par notre méthode à celles obtenues avec les méthodes d'interpolation par transformée B-spline et de filtrage médian.

Dans le chapitre 6, nous décrivons notre approche de l'agrandissement par induction sur un ensemble. Nous caractérisons précisément l'ensemble des images agrandies soumises à la contrainte de réduction et nous donnons une interprétation fréquentielle des solutions. Nous présentons ensuite diverses méthodes de projection qui permettent d'obtenir une projection soit exacte (orthogonale), soit approchée (non orthogonale) dans les cas de consistance et d'inconsistance de l'ensemble des solutions. L'efficacité des méthodes de projection est ensuite étudiée à travers de nombreux résultats expérimentaux.

Le chapitre 7 présente des améliorations et des applications de nos deux méthodes d'agrandissement présentées dans les chapitres 5 et 6. Tout d'abord, nous améliorons la qualité de l'image agrandie par synthèse de similarités en recherchant les similarités sur une pyramide en quinconce à facteur de réduction $\sqrt{2}$. A partir d'images très diverses, nous montrons que notre méthode produit des agrandissements de meilleure qualité que ceux produits par une transformée B-spline, notamment en préservant la netteté des images. Ensuite, nous exploitons la méthode d'agrandissement par induction pour régulariser, vis-à-vis de la contrainte de réduction, les images agrandies par synthèse de similarités. A l'aide de résultats expérimentaux, nous mettons en évidence la restauration des basses fréquences. Enfin, nous exploitons également cette même méthode pour réaliser la compression réversible et la transmission progressive d'une image [23], [24], [25]. La comparaison effectuée entre les résultats obtenus avec une pyramide laplacienne et ceux obtenus grâce à notre méthode montre la supériorité de cette dernière, tant sur le taux de compression que sur la qualité des images intermédiaires générées lors de la reconstruction.

Enfin, nous concluons ce mémoire dans le chapitre 8 en dégagant les contributions apportées à l'agrandissement d'images numériques. Nous discutons des limites de nos approches, ainsi que des perspectives de recherche envisagées.

Chapitre 2

Méthodes d'agrandissement préservant les fréquences

2.1 Introduction

Le théorème d'échantillonnage (Shannon) précise qu'un signal continu peut-être reconstruit exactement à partir d'échantillons, si la fréquence d'échantillonnage est deux fois supérieure à la plus haute fréquence contenue dans le signal (fréquence de Nyquist), [126], [127]. En supposant que le signal discret que l'on souhaite agrandir vérifie cette condition, on dispose d'une méthode d'agrandissement relativement simple. Il suffit de sur-échantillonner les fonctions continues (sinusoïdes) associées à chaque fréquence, le pas d'échantillonnage étant fixé en fonction du facteur d'agrandissement souhaité. Plus ce facteur est important, plus le pas d'échantillonnage est petit, et plus le signal agrandi est proche du signal continu hypothétique. Le signal de départ et le signal agrandi ont le même contenu fréquentiel, la seule différence porte sur le pas d'échantillonnage. On peut donc parler d'une préservation fréquentielle dans le processus d'agrandissement.

Dans la suite, nous commençons par rappeler des notions fondamentales sur la transformée de Fourier et sur l'analyse spectrale. Nous étudions ensuite la méthode du *zero-padding* qui permet d'agrandir une image en préservant idéalement les fréquences. Nous étudions également d'autres méthodes d'agrandissement comme l'interpolation avec des fonctions polynomiales et l'interpolation avec la transformée B-spline qui préservent plus ou moins bien les fréquences originales. Nous terminons ce chapitre en décrivant quelques approches qui corrigent le problème d'apparition de flou qui est caractéristique des méthodes d'agrandissement agissant par préservation fréquentielle.

2.2 Notions sur la transformée de Fourier

Le domaine de représentation habituel d'une image est le *domaine spatial*. C'est celui auquel est sensible notre système visuel, et par voie de conséquence, c'est aussi celui qui nous délivre le plus d'informations sur le contenu d'une image.

Le fait de disposer de plusieurs représentations de la même image permet d'analyser et de mettre en œuvre plus facilement certains traitements. Des opérations difficiles à appliquer dans un domaine peuvent devenir plus faciles dans un autre. Dans cette perspective, le *domaine fréquentiel* s'avère très complémentaire du domaine spatial. Il représente une image par un ensemble de sinusoides de fréquences différentes, ayant chacune une amplitude et un décalage de phase particuliers. Le passage du domaine spatial au domaine fréquentiel se fait par l'intermédiaire de la *transformée de Fourier*.

Nous donnons ci-après les éléments nécessaires à ce chapitre ; pour plus de détails sur la transformée de Fourier, le lecteur peut se reporter à [58].

2.2.1 La transformée de Fourier

L'*analyse de Fourier* consiste à déterminer l'amplitude et le décalage de phase des sinusoides à partir d'une représentation spatiale du signal. Inversement, la *synthèse de Fourier* consiste à ajouter les sinusoides entre elles pour obtenir à nouveau une représentation spatiale du signal. Ces deux opérations sont rendues possibles par le couple de transformées suivant :

$$F(u) = \int_{-\infty}^{\infty} f(x)e^{-2\pi iux} dx \quad (2.1)$$

$$f(x) = \int_{-\infty}^{\infty} F(u)e^{2\pi iux} du \quad (2.2)$$

La fonction $F(u)$ est appelée le *spectre fréquentiel* du signal. C'est une fonction complexe qui associe une amplitude et une phase à chaque fréquence u . L'équation (2.1) définit la *transformée de Fourier directe*, tandis que l'équation (2.2) définit la *transformée de Fourier inverse*. Le passage du domaine spatial au domaine fréquentiel est complètement spécifié par ces deux équations. Celles-ci sont valables pour des signaux continus. Dans le cas particulier d'un signal périodique, on peut obtenir des simplifications et aboutir aux séries de Fourier : le spectre a alors la remarquable propriété de devenir discret puisque les fréquences non nulles sont des entiers multiples d'une fréquence fondamentale.

Pour obtenir la représentation fréquentielle d'une image, il faut redéfinir le couple de transformées (2.1) et (2.2) avec une dimension supplémentaire. Il faut également tenir compte du support fini de l'image et de sa discrétisation. Pour cela, on remplace les intégrales par des sommes partielles. On obtient alors le couple de transformées

discrètes suivant :

$$F(u, v) = \sum_{x,y=0}^{N-1} f(x, y) e^{\frac{-2\pi i}{N}(ux+vy)} \quad (2.3)$$

$$f(x, y) = \frac{1}{N^2} \sum_{u,v=0}^{N-1} F(u, v) e^{\frac{2\pi i}{N}(ux+vy)} \quad (2.4)$$

Elles correspondent à une image de taille $N \times N$ pixels avec $N = 2^n$. On remarquera notamment que l'échantillonnage périodique, associé à l'image dans le domaine spatial, a pour effet de rendre périodique le spectre fréquentiel $F(u, v)$. De même, la discrétisation du domaine fréquentiel conduit à une périodicité de l'image $f(x, y)$ dans le domaine spatial. Ceci génère deux artefacts. Le premier est la génération de hautes fréquences dues aux discontinuités du pavage de l'image dans le domaine spatial. Pour y remédier, une solution simple consiste à atténuer les valeurs sur les bords de l'image. Une autre solution consiste à former une nouvelle image à partir de quatre copies de l'image originale, ceci en les positionnant de manière à avoir une symétrie centrale. On effectue alors l'analyse de Fourier sur cette nouvelle image. Du fait de l'annulation des termes en sinus, la transformée est appelée *transformée en cosinus discret*. Le deuxième artefact concerne la taille de l'image discrète. Lorsque N est trop petit, l'approximation de la transformée de Fourier continue est très grossière.

La transformée de Fourier discrète nécessite une quantité de calculs non négligeable. On voit assez facilement avec l'équation (2.3) que, pour une image de taille $N \times N$ pixels, la complexité algorithmique est $O(N^4)$. L'analyse de Fourier sur des images de grande taille devient possible en utilisant un algorithme dû à James Cooley et John Tukey (1965), dont la complexité est $O(N^2 \log_2 N)$. Cet algorithme est communément désigné par le terme *FFT (Fast Fourier Transform)*. L'économie de calcul vient d'un regroupement astucieux des termes pour éviter les calculs redondants.

2.2.2 Les propriétés de la transformée de Fourier

La transformée de Fourier a des propriétés qui la rendent très attrayante pour de nombreux traitements. Parmi ces qualités, nous en retenons trois : les propriétés de linéarité, d'échelle et de convolution. La *propriété de linéarité* est importante car elle permet d'utiliser l'analyse de Fourier pour l'étude des systèmes linéaires. Elle s'écrit de la manière suivante :

$$\alpha f(x) + \beta g(x) \xrightarrow{TF} \alpha F(u) + \beta G(u) \quad (2.5)$$

La *propriété d'échelle* établit qu'une compression (dilatation) du signal dans le domaine spatial conduit à une dilatation (compression) dans le domaine fréquentiel. On remarquera que la dilatation (compression) est accompagnée d'une diminution

(augmentation) de l'amplitude de manière à ce que l'aire demeure constante :

$$\begin{aligned} f(\alpha x) &\xleftrightarrow{TF} \frac{1}{\alpha} F\left(\frac{u}{\alpha}\right) \\ \frac{1}{\alpha} f\left(\frac{x}{\alpha}\right) &\xleftrightarrow{TF} F(\alpha u) \end{aligned} \quad (2.6)$$

La *propriété de convolution* est particulièrement intéressante pour l'étude des filtres. Elle indique que le produit de convolution dans le domaine spatial (fréquentiel) se traduit par une multiplication dans le domaine fréquentiel (spatial) :

$$\begin{aligned} f * g(x) &\xleftrightarrow{TF} F(u) \cdot G(u) \\ f(x) \cdot g(x) &\xleftrightarrow{TF} F * G(u) \end{aligned} \quad (2.7)$$

Pour bien comprendre toute l'importance de cette propriété, imaginons que $g(x)$ soit un filtre et que $f(x)$ soit le signal à analyser. Le filtrage du signal se traduit par le produit de convolution $f * g(x)$. Peu intuitive dans le domaine spatial, cette opération devient beaucoup plus claire dans le domaine fréquentiel. Le filtre agit sur les fréquences du signal en atténuant ou en renforçant leur amplitude. En pratique, il est particulièrement commode de construire un filtre en lui donnant les propriétés voulues dans le domaine fréquentiel, puis de le mettre en œuvre dans le domaine spatial en utilisant sa réponse impulsionnelle.

2.2.3 L'analyse spectrale

Dans l'équation (2.3), le spectre fréquentiel $F(u, v)$ est de nature complexe. Pour apprécier plus facilement l'information qu'il contient, on s'intéresse à son amplitude $|F(u, v)|$ qui a l'avantage d'être une fonction réelle. Elle est souvent désignée comme étant le *spectre d'amplitude*. En notant $F_{re}(u, v)$ et $F_{im}(u, v)$ les parties réelle et imaginaire de $F(u, v)$, nous avons :

$$|F(u, v)| = \sqrt{F_{re}(u, v)^2 + F_{im}(u, v)^2} \quad (2.8)$$

La fonction définie par $|F(u, v)|^2$ est appelée le *spectre de puissance*. Le décalage de phase associé aux fréquences peut également s'exprimer sous une forme spectrale. On parle dans ce cas du *spectre de phase*. Il se définit de la manière suivante :

$$\Phi(u, v) = \tan^{-1} \left(\frac{F_{im}(u, v)}{F_{re}(u, v)} \right) \quad (2.9)$$

En pratique, l'information pertinente sur l'image se retrouve essentiellement dans l'amplitude des fréquences. On préfère donc généralement le spectre d'amplitude au spectre de phase.

Pour afficher le spectre d'amplitude sur une sortie graphique, on calcule le logarithme de l'amplitude et on normalise le résultat pour avoir un affichage sur 256 niveaux de gris. On obtient la fonction suivante :

$$D(u, v) = \frac{\log(1 + |F(u, v)|)}{\log(1 + \max_{s,t} |F(s, t)|)} \cdot 255 \quad (2.10)$$

L'intensité lumineuse du spectre est proportionnelle à l'amplitude de $|F(u, v)|$ sachant que la valeur 255 correspond au blanc et que la valeur nulle correspond au noir.

Un exemple de spectre d'amplitude est donné sur la figure 2.1(b). L'image originale représente une prise de vue par un cameraman, elle a une taille de 128×128 pixels. Le spectre étant périodique, on préfère l'afficher en le centrant en $(0, 0)$ afin d'améliorer le confort visuel. Ainsi, dans cet exemple, u et v appartiennent à l'intervalle $[-64, 64[$. En affichant le spectre de cette manière, on tend vers les hautes fréquences de l'image en s'éloignant du centre et vers les basses fréquences en s'en rapprochant.

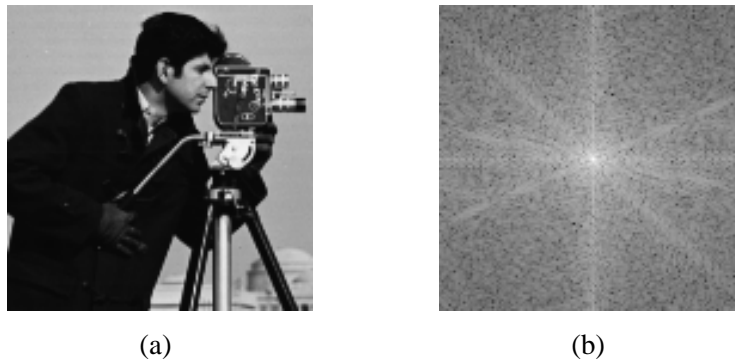


Figure 2.1 – (a) Image du cameraman. (b) Affichage de l'image spectrale $D(u, v)$.

Plusieurs observations se dégagent du spectre montré sur la figure 2.1(b). Tout d'abord, la fréquence fondamentale (au centre) possède l'amplitude la plus élevée et l'amplitude décroît globalement en s'éloignant du centre. Cette observation est une propriété générale des spectres d'amplitude des images naturelles : l'amplitude est inversement proportionnelle à la fréquence [141]. On peut également observer que le spectre est symétrique par rapport à son centre. Ceci est une propriété que l'on peut facilement démontrer à partir des expressions (2.3) et (2.8). En fait, l'utilisation de nombres complexes dans la transformée de Fourier discrète introduit une redondance de près de la moitié des coefficients. On observe également sur le spectre des traces linéaires. Elles correspondent à des orientations privilégiées comme celles des pieds de la caméra ou du tronc du cameraman : les traces sont orthogonales à ces structures. Il faut également remarquer que la transformée de Fourier discrète introduit une périodicité implicite de l'image. Autrement dit, des différences importantes de niveaux de gris sur les bords de l'image introduisent obligatoirement des fausses discontinuités horizontales et verticales. Elles se traduisent sur le spectre par une amplification des coefficients associés aux fréquences horizontales et verticales, c'est-à-dire par des traces horizontales et verticales. Cet artefact est partiellement visible dans l'exemple de la figure 2.1.

2.3 La préservation fréquentielle

Le fait d'augmenter le nombre d'échantillons d'une image se traduit dans le domaine fréquentiel par un élargissement du spectre dans les hautes fréquences. Réciproquement, l'élargissement du spectre fréquentiel s'accompagne d'un sur-échantillonnage dans le domaine spatial (cf. propriété (2.6)), c'est-à-dire d'un agrandissement de l'image initiale. Comme nous l'avons dit dans la section 2.1, nous supposons dans ce chapitre que l'image à agrandir vérifie la condition du théorème d'échantillonnage. En élargissant le spectre de l'image, il faut donc assigner des valeurs nulles aux hautes fréquences introduites. On effectue ainsi un sur-échantillonnage, tout en préservant le spectre fréquentiel initial.

Dans la suite de cette section, nous étudions plus en détail cette méthode d'agrandissement qui est souvent référencée dans la littérature comme la méthode du *zero-padding*. Nous étudions également des approches qui visent à optimiser la complexité algorithmique et à corriger les artefacts de cette méthode.

2.3.1 La méthode du *zero-padding*

La méthode du *zero-padding* se résume par trois étapes illustrées sur la figure 2.2. Dans la première, il s'agit de passer du domaine spatial au domaine fréquentiel avec la transformée de Fourier. Dans la deuxième, le spectre est élargi avec des zéros. Enfin dans la troisième étape, on utilise la transformée de Fourier inverse pour revenir dans le domaine spatial.

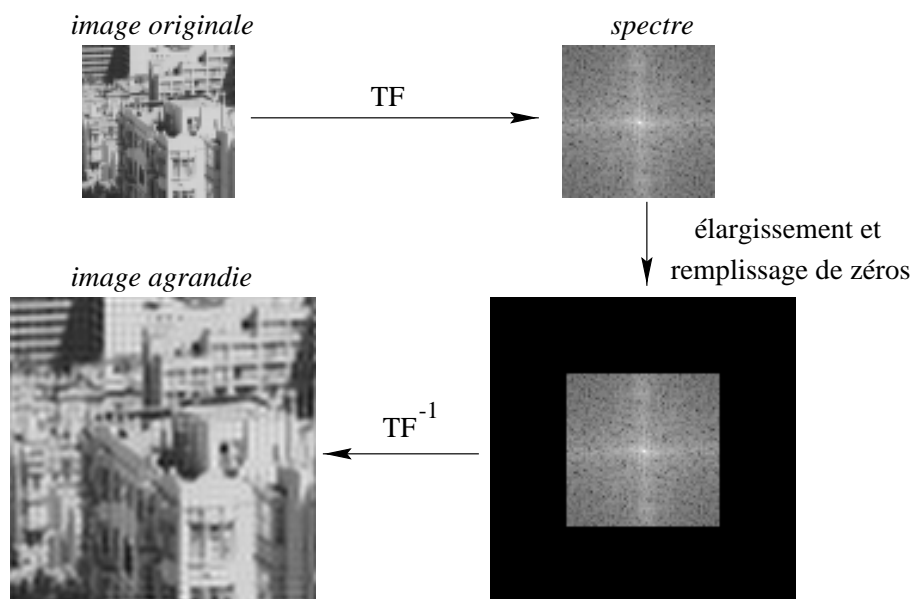


Figure 2.2 – Agrandissement de facteur $a = 2$ avec la méthode du *zero-padding*.

Cherchons à formaliser le lien qui existe entre l'image originale et l'image agrandie par un facteur entier a avec la méthode du *zero-padding*. Pour cela, désignons par $f(x, y)$ l'image originale de $N \times N$ pixels, et par $g(x, y)$ l'image agrandie de $aN \times aN$

pixels. D'après (2.4), cette dernière s'exprime en fonction de son spectre fréquentiel par :

$$g(x, y) = \frac{1}{(aN)^2} \sum_{u, v = -aN/2}^{aN/2-1} G(u, v) e^{\frac{2\pi i}{aN}(ux+vy)} \quad (2.11)$$

Le spectre $G(u, v)$ est lié au spectre $F(u, v)$ par :

$$(-aN/2 \leq u, v < aN/2) \quad \begin{cases} G(u, v) = a^2 F(u, v) & \text{lorsque } -N/2 \leq u, v < N/2 \\ G(u, v) = 0 & \text{sinon} \end{cases} \quad (2.12)$$

La multiplication de $F(u, v)$ par le coefficient a^2 permet de garder la dynamique des niveaux de gris de l'image originale $f(x, y)$. En substituant $G(u, v)$ dans (2.11), on trouve :

$$g(x, y) = \frac{1}{N^2} \sum_{u, v = -N/2}^{N/2-1} F(u, v) e^{\frac{2\pi i}{aN}(ux+vy)} \quad (2.13)$$

Et finalement, on peut exprimer l'image agrandie en fonction de l'image originale en substituant $F(u, v)$ à l'aide de (2.3) :

$$\begin{aligned} g(x, y) &= \frac{1}{N^2} \sum_{u, v = -N/2}^{N/2-1} \sum_{m, n = -N/2}^{N/2-1} f(m, n) e^{\frac{-2\pi i}{N}(um+vn)} e^{\frac{2\pi i}{aN}(ux+vy)} \\ &= \frac{1}{N^2} \sum_{m, n = -N/2}^{N/2-1} f(m, n) \sum_{u, v = -N/2}^{N/2-1} e^{\frac{2\pi i u}{N}(\frac{x}{a}-m)} e^{\frac{2\pi i v}{N}(\frac{y}{a}-n)} \end{aligned} \quad (2.14)$$

On peut réécrire l'expression (2.14) d'une manière plus intelligible :

$$g(x, y) = \sum_{m, n = -N/2}^{N/2-1} f(m, n) h\left(\frac{x}{a} - m, \frac{y}{a} - n\right) \quad (2.15)$$

$$h(x, y) = \frac{1}{N^2} \sum_{u, v = -N/2}^{N/2-1} e^{\frac{2\pi i}{N}(ux+vy)} \quad (2.16)$$

On définit l'image $f^{\uparrow a}$ comme étant un agrandissement de facteur a de l'image f obtenu en intercalant des zéros entre les pixels initiaux :

$$(-aN/2 \leq m, n < aN/2) \quad \begin{cases} f^{\uparrow a}(m, n) = f\left(\frac{m}{a}, \frac{n}{a}\right) & \text{lorsque } \frac{m}{a} \text{ et } \frac{n}{a} \text{ sont entiers} \\ f^{\uparrow a}(m, n) = 0 & \text{sinon} \end{cases} \quad (2.17)$$

En utilisant cette nouvelle image, on réécrit (2.15) par :

$$g(x, y) = \sum_{m, n = -aN/2}^{aN/2-1} f^{\uparrow a}(m, n) h\left(\frac{x-m}{a}, \frac{y-n}{a}\right) = f^{\uparrow a} * h\left(\frac{x}{a}, \frac{y}{a}\right) \quad (2.18)$$

Avec cette dernière expression, il apparaît que la méthode du *zero-padding* revient à intercaler des zéros dans l'image initiale, puis à convoluer l'image obtenue avec la fonction h dilatée d'un facteur a . Intéressons-nous maintenant aux propriétés de la fonction h . L'expression (2.16) montre qu'il s'agit d'une série géométrique. Après quelques développements, on trouve l'expression analytique suivante :

$$h(x, y) = \frac{1}{N^2} \cdot \cos \frac{\pi(x+y)}{N} \cdot \frac{\sin \pi x \sin \pi y}{\sin \frac{\pi x}{N} \sin \frac{\pi y}{N}} \quad (2.19)$$

Lorsque N tend vers l'infini, on peut faire une approximation continue de la somme dans (2.16) et obtenir la limite de la fonction h :

$$\tilde{h}(x, y) = \frac{1}{N^2} \int_{-\frac{N}{2}}^{\frac{N}{2}} \int_{-\frac{N}{2}}^{\frac{N}{2}} e^{\frac{2\pi i}{N}(ux+vy)} dudv = \text{sinc } \pi x \cdot \text{sinc } \pi y \quad (2.20)$$

où la fonction sinc est la fonction *sinus cardinal* définie par $\text{sinc}(x) = \sin(x)/x$. La figure 2.3(a) illustre la fonction limite \tilde{h} sur un support de largeur 16. Les lignes sur le plan inférieur montrent les passages par zéro. Du fait de la séparabilité de \tilde{h} , on la représente également avec une seule dimension sur la figure 2.3(b), et ceci sans perte de généralité. La fonction passe par zéro pour toutes les valeurs entières de x , sauf pour $x = 0$ où la fonction vaut 1. Ceci signifie que la convolution décrite par (2.18) préserve les pixels de l'image originale. La méthode du *zero-padding* est donc bien une méthode d'interpolation où la fonction d'interpolation limite est donnée par la fonction \tilde{h} .

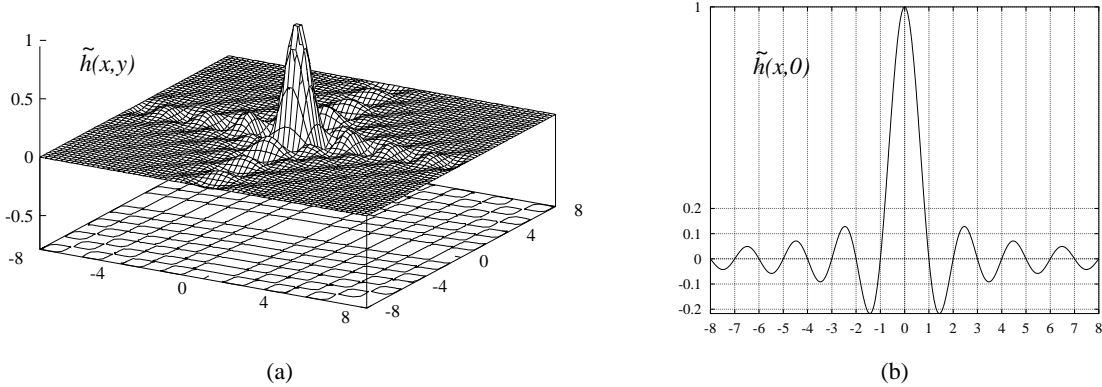


Figure 2.3 – (a) Fonction limite $\tilde{h}(x, y)$. (b) Illustration de $\tilde{h}(x, 0)$.

Une comparaison entre $h(x, 0)$ et sa limite $\tilde{h}(x, 0)$ est montrée sur la figure 2.4 pour quelques valeurs de N . On remarque que $h(x, 0)$ réalise une bonne approximation de $\tilde{h}(x, 0)$ dans l'intervalle $[0, N/4]$, et plutôt mauvaise dans l'intervalle $[N/4, N/2]$. En fait, plus on se rapproche de $N/2$ et plus l'approximation est mauvaise. Il est intéressant d'interpréter la relation (2.18) dans le domaine fréquentiel. D'après la propriété (2.7), le produit de convolution devient un produit de multiplication et on a :

$$G(u, v) = a^2 F^{\uparrow a}(u, v) \cdot H(u, v) \quad (2.21)$$

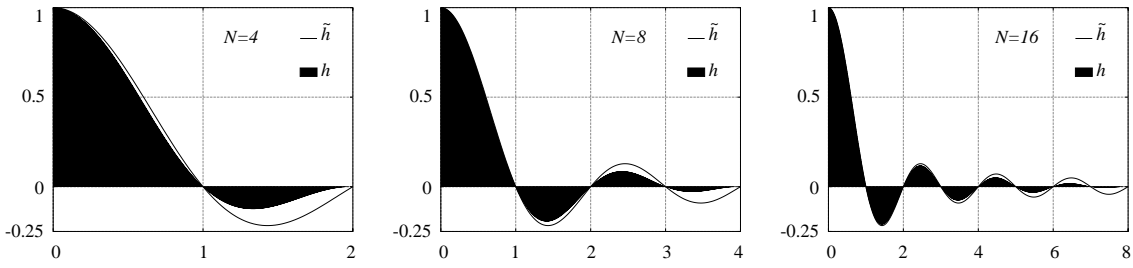


Figure 2.4 – Comparaison entre $h(x, 0)$ et sa limite $\tilde{h}(x, 0)$ pour différentes valeurs de N .

On peut identifier facilement $H(u, v)$ en interprétant la définition de $h(x, y)$ comme une transformée de Fourier inverse, c'est-à-dire :

$$h(x, y) = \frac{1}{N^2} \sum_{u, v=-aN/2}^{aN/2-1} H(u, v) e^{\frac{2\pi i}{N}(ux+vy)} \quad (2.22)$$

$$(-aN/2 \leq u, v < aN/2) \quad \begin{cases} H(u, v) = 1 & \text{lorsque } -N/2 \leq u, v < N/2 \\ H(u, v) = 0 & \text{sinon} \end{cases} \quad (2.23)$$

On peut également facilement montrer que le spectre $F^{\uparrow a}(u, v)$ résulte d'une périodisation de $F(u, v)$ sur l'intervalle $[-aN/2, aN/2 - 1]$, ce qui s'écrit :

$$-aN/2 \leq u, v < aN/2 \quad F^{\uparrow a}(u, v) = F(u, v) \quad (2.24)$$

Autrement dit, le fait d'intercaler des zéros dans l'image originale se traduit d'un point de vue fréquentiel par un élargissement du spectre, où les hautes fréquences introduites sont obtenues en dupliquant le spectre initial. On vérifie facilement qu'en substituant $F^{\uparrow a}(u, v)$ et $H(u, v)$ dans (2.21), on retrouve bien l'expression initiale (2.12) de $G(u, v)$. On voit que la fonction $H(u, v)$ permet de préserver le spectre initial $F(u, v)$ en bloquant les hautes fréquences introduites par $F^{\uparrow a}(u, v)$. Elle constitue un *filtre passe-bas* dont $h(x, y)$ est l'expression dans le domaine spatial.

La mise en place du *zero-padding* par filtrage fréquentiel ou par interpolation spatiale est illustrée sur la figure 2.5. L'image agrandie révèle un artefact de la méthode du *zero-padding*. La préservation exacte du spectre initial entraîne un effet d'ondulation des contours de l'image. La figure 2.6 montre plus clairement cet effet en utilisant une mire.

Ce problème d'ondulation est dû au filtrage «tout ou rien» introduit par $H(u, v)$ et qui se traduit par des discontinuités spectrales. Une localisation précise dans le domaine fréquentiel est forcément répartie sur tout le domaine spatial. On peut vérifier sur la figure 2.4 que la fonction d'interpolation $h(x, y)$, qui est l'équivalent spatial du filtre $H(u, v)$, possède bien un comportement oscillatoire. Celui-ci décroît très lentement et il est davantage marqué dans les directions horizontale et verticale. Ceci signifie que tous les pixels de l'image originale contribuent plus ou moins dans l'interpolation d'un seul pixel de l'image agrandie. Un pixel interpolé dans une région

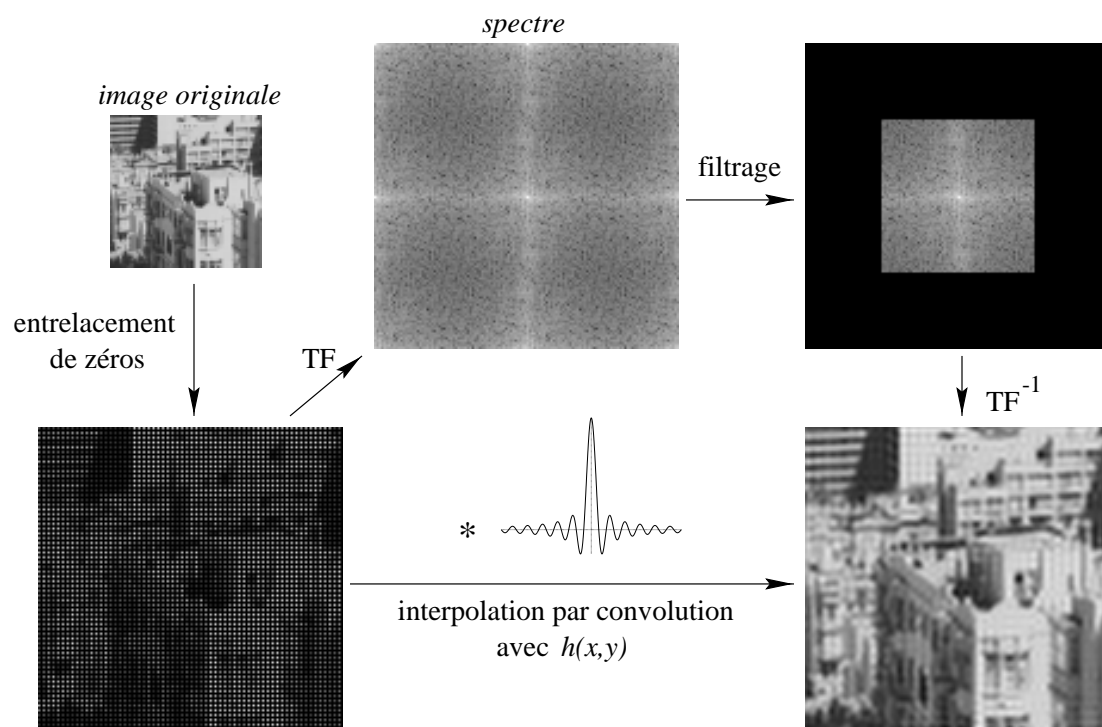


Figure 2.5 – Mise en œuvre du *zero-padding* par interpolation spatiale ou par filtrage fréquentiel.

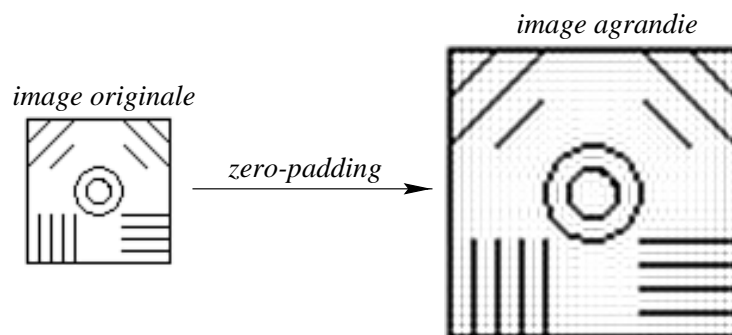


Figure 2.6 – Illustration de l'effet d'ondulation dans la méthode du *zero-padding*.

homogène est influencé par les contours avoisinants. Les contours, qui traduisent des variations brutales des niveaux de gris, sont donc répercutés sur leur voisinage.

Pour diminuer l'effet d'ondulation des contours, il faut augmenter la convergence vers zéro de la fonction d'interpolation idéale h . Pour cela, différentes approches ont été proposées dans la littérature. Yaroslavsky remplace la fonction h par une combinaison de fonctions sinus cardinal en opérant un léger décalage de phase pour que la fonction d'interpolation résultante converge plus rapidement vers zéro [148]. Une autre approche proposée par Brouaye consiste à décomposer le signal avec des fonctions orthogonales de type ondelettes plutôt qu'avec des sinusoides. Les

ondelettes étant mieux localisées dans le temps, la fonction d'interpolation résultante a l'avantage de décroître beaucoup plus rapidement que le sinus cardinal [20].

Pour diminuer considérablement l'effet d'ondulation, on peut envisager de réaliser une approximation de la fonction d'interpolation idéale sur un support de petite taille. Non seulement l'étendue des ondulations se trouve limitée au support choisi, mais la convolution est aussi beaucoup plus rapide. Evidemment, si on veut garder une certaine qualité dans l'interpolation, il ne faut pas que le support soit trop petit. La méthode d'approximation la plus simple consiste à couper la fonction sinus cardinal en effectuant un fenêtrage avec une fonction rectangle, cependant il y a un renforcement des ondulations sur le support préservé. Pour corriger ce problème, on peut utiliser une fonction gaussienne pour adoucir le fenêtrage précédent, ou utiliser différentes fenêtres comme celle de Hann, Hamming, Blackman, ou encore Kaiser. Plutôt que de fenêtrer le sinus cardinal sur un petit support, on peut déterminer sur ce même support la fonction d'interpolation optimale dont la réponse en fréquence approche au mieux le filtre rectangle idéal. Dans ce cas, il s'agit de réaliser l'interpolation avec un filtre FIR (*finite impulse response*) optimal [34], [35], [43], [56], [121].

Pour supprimer complètement l'effet d'ondulation, Liu *et al.* proposent une approche originale de l'interpolation où l'image agrandie doit minimiser une fonction d'énergie globale tout en préservant la valeur des pixels d'origine [87]. Ce problème est résolu par les moindres carrés en utilisant un algorithme de descente en gradient. Néanmoins, cette approche est itérative et nécessite une quantité importante de calculs.

Notes bibliographiques

La méthode du *zero-padding* a été proposée pour la première fois par Schafer et Rabiner en 1973, comme une méthode d'interpolation idéale mais avant tout théorique [121]. Plus tard, elle a été mise en œuvre avec la FFT sur des signaux 1D [53], [108], puis étendue aux signaux 2D [118]. Cette approche a ensuite été optimisée en évitant les calculs inutiles où interviennent les zéros, que ce soit dans le cas 1D [1], ou dans le cas 2D [32], [129]. La méthode du *zero-padding* a également été mise en œuvre avec d'autres transformées que la FFT. On peut citer notamment la *transformée en cosinus discret* (DCT) [3], [145], ou encore la *transformée de Hartley discrète* (DHT) qui évite le calcul des valeurs complexes [2].

2.3.2 Interpolation avec des fonctions polynomiales

Nous avons vu dans la section 2.3.1 que l'agrandissement d'une image avec une préservation exacte des fréquences conduit à interpoler cette image avec une fonction limite de type sinus cardinal. Nous avons relevé deux problèmes inhérents au fait que cette fonction s'étend sur tout le support de l'image. Le premier est une complexité algorithmique élevée $O(N^4)$, et le deuxième est un effet d'ondulation des contours dû à la participation de tous les pixels dans l'interpolation d'un seul. Pour corriger ces deux problèmes à la fois, il faut utiliser une fonction d'interpolation à support

compact, au détriment de la qualité de la préservation des fréquences. Dans cette section, nous étudions des fonctions d'interpolation qui ont cette propriété. Elles sont construites en raccordant aux points d'échantillonnage initiaux des morceaux de polynômes de degrés plus ou moins élevés [89], [101].

Pour simplifier la présentation de cette méthode d'agrandissement, nous nous intéressons à l'interpolation sur une seule dimension. Elle peut facilement être étendue sur deux dimensions en admettant la séparabilité de la fonction d'interpolation 2D. D'un point de vue pratique, l'interpolation est réalisée avec un produit de convolution, de la même manière que nous l'avons vu avec la relation (2.18) dans le cas d'une interpolation sur deux dimensions. Plus précisément, l'interpolation d'un signal f à une dimension avec un facteur d'agrandissement a est définie par :

$$g(x) = \sum_{m=0}^{aN-1} f^{\uparrow a}(m) h\left(\frac{x-m}{a}\right) = f^{\uparrow a} * h\left(\frac{x}{a}\right) \quad (2.25)$$

où le nombre N représente le nombre d'échantillons de f et la fonction $f^{\uparrow a}$ est obtenue en intercalant $a-1$ zéros entre les échantillons de f , c'est-à-dire :

$$(0 \leq m < aN) \quad \begin{cases} f^{\uparrow a}(m) = f\left(\frac{m}{a}\right) & \text{lorsque } \frac{m}{a} \text{ est entier} \\ f^{\uparrow a}(m) = 0 & \text{sinon} \end{cases} \quad (2.26)$$

Dans la suite, nous étudions les cas où la fonction d'interpolation h est construite à partir de morceaux de polynômes de degrés 0, 1, 2 ou 3.

Interpolation au plus proche voisin

La fonction d'interpolation la plus simple est illustrée sur la figure 2.7(a). Elle est obtenue à partir d'un polynôme de degré 0. On la désigne souvent comme une *interpolation au plus proche voisin*, puisque seul le point le plus proche intervient dans l'interpolation. Elle se formalise de la manière suivante :

$$\begin{cases} h(x) = 1 & \text{lorsque } |x| \leq \frac{1}{2} \\ h(x) = 0 & \text{sinon} \end{cases} \quad (2.27)$$

Sa réponse en fréquence est donnée par :

$$H(u) = \int_{-\frac{1}{2}}^{\frac{1}{2}} e^{-2\pi i u x} dx = \text{sinc } \pi u \quad (2.28)$$

En raison de sa simplicité extrême, l'interpolation au plus proche voisin est très appréciée dans les applications en temps réel. Elle a par exemple longtemps été utilisée dans les moteurs 3D pour interpoler les textures dans les mondes virtuels. Elle a cependant le défaut de générer un effet de pixelisation de l'image (voir figure 2.8), qui laisse une impression d'agrandissement «synthétique» à l'observateur. La figure 2.7(b) montre un spectre en fréquence relativement éloigné du filtre idéal

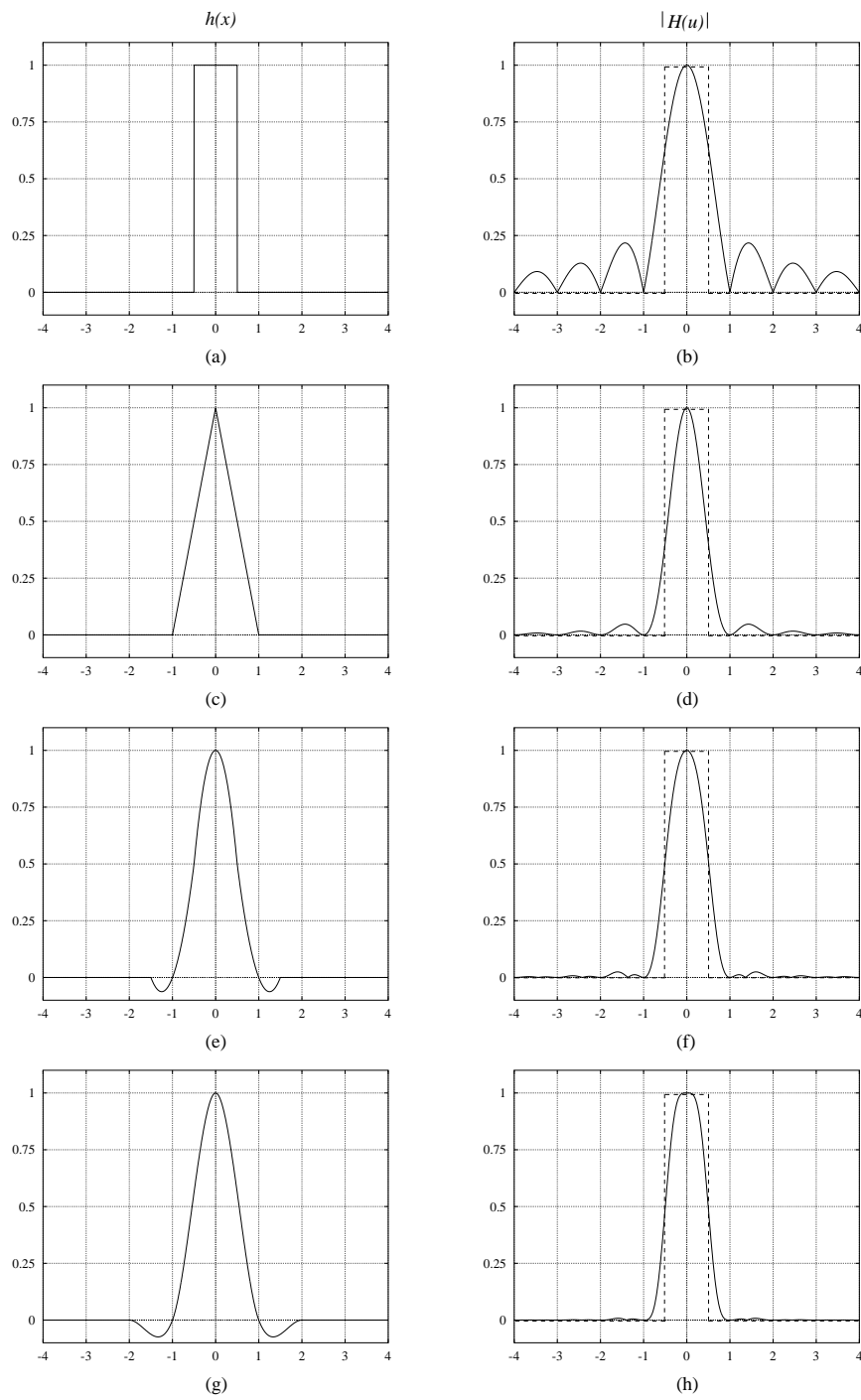


Figure 2.7 – Fonctions d'interpolation polynomiale et spectres associés: (a-b) interpolation au plus proche voisin, (c-d) interpolation linéaire, (e-f) interpolation quadratique, (g-h) interpolation cubique.

illustré en pointillés. La bande stoppante contient des lobes importants qui atténuent partiellement les hautes fréquences introduites par $f^{\uparrow a}$ lors de l'agrandissement.

On peut trouver dans [12] et [85] une approche particulière où l'effet de pixelisation est fortement atténué grâce à une sorte de lissage de l'image agrandie par interpolation au plus proche voisin. Dans cette approche, chaque groupe de pixels issu de l'agrandissement d'un pixel original est circonscrit par un disque auquel on attribue le niveau de gris du groupe de pixels. La valeur de chaque pixel de l'image agrandie est alors recalculée en effectuant une moyenne pondérée entre les niveaux de gris des disques qui recouvrent le pixel. Le poids associé à chaque disque est fonction de la surface du pixel qui est recouverte par le disque.

Interpolation linéaire

Une interpolation entre les deux points les plus proches peut être réalisée en reliant les échantillons initiaux entre eux par des segments de droite. La fonction d'interpolation correspondante, illustrée sur la figure 2.7(c) est obtenue en utilisant deux polynômes de degré 1 :

$$\begin{cases} h(x) = 1 - |x| & \text{lorsque } |x| \leq 1 \\ h(x) = 0 & \text{sinon} \end{cases} \quad (2.29)$$

Sa réponse en fréquence est donnée par :

$$H(u) = \int_{-1}^1 (1 - |x|) e^{-2\pi i u x} dx = \text{sinc}^2 \pi u \quad (2.30)$$

Ce type d'interpolation donne globalement de meilleurs résultats que l'interpolation au plus proche voisin avec peu de calculs supplémentaires. On la retrouve de plus en plus souvent dans les applications en temps réel. La figure 2.7(d) montre une réponse en fréquence où les lobes sont nettement moins proéminents dans la bande stoppante. Toutefois les fréquences sont davantage atténuées dans la bande passante, ce qui se traduit dans le domaine spatial par un flou visuel assez important (voir figure 2.8).

Interpolation quadratique

L'interpolation entre les trois points les plus proches a été étudiée par Dodgson en utilisant trois polynômes de degré 2 [48]. Pour éviter un déphasage, les polynômes quadratiques sont définis sur des morceaux décalés de la moitié du pas d'échantillonnage initial. Pour déterminer les coefficients des trois polynômes, Dodgson utilise diverses contraintes : symétrie, normalisation, continuité aux points de raccord, et préservation des échantillons initiaux. Il aboutit ainsi à la fonction d'interpolation quadratique suivante :

$$\begin{cases} h(x) = -2x^2 + 1 & \text{lorsque } |x| \leq \frac{1}{2} \\ h(x) = x^2 - \frac{5}{2}|x| + \frac{3}{2} & \text{lorsque } \frac{1}{2} \leq |x| \leq \frac{3}{2} \\ h(x) = 0 & \text{sinon} \end{cases} \quad (2.31)$$

Dont la réponse en fréquence est donnée par :

$$H(u) = \frac{6\text{sinc } \pi u - 6\text{sinc } 3\pi u - \cos \pi u + \cos 3\pi u}{(2\pi u)^2} \quad (2.32)$$

La fonction d'interpolation quadratique et sa réponse en fréquence sont illustrées respectivement sur les figures 2.7(e) et 2.7(f). Par rapport à l'interpolation linéaire, les fréquences sont moins atténuées dans la bande passante, et par conséquent, l'image interpolée est moins floue (voir figure 2.8). Cependant, l'effet de pixelisation est encore visible puisque les lobes dans la bande stoppante ne sont pas suffisamment atténués.

Lorsque le facteur d'agrandissement est $a = 2$, on remarque à partir de l'expression (2.32) que l'interpolation quadratique est identique à l'interpolation linéaire.

(b) *interpolation au plus proche voisin*



(c) *interpolation linéaire*



(a) *image originale*



(d) *interpolation quadratique*



(e) *interpolation cubique*

Figure 2.8 – Agrandissement de facteur $a = 4$ par interpolation avec des fonctions polynomiales.

Interpolation cubique

L'interpolation entre les quatre points les plus proches peut être réalisée à partir de trois polynômes de degré trois [78], [89], [100]. Pour réduire l'espace des solutions,

on applique diverses contraintes sur ces polynômes : symétrie autour de zéro, continuité et première dérivée continue aux points de raccord. On aboutit finalement à une fonction d'interpolation cubique définie par un seul paramètre :

$$\begin{cases} h(x) = (A + 2)|x^3| - (A + 3)x^2 + 1 & \text{lorsque } |x| \leq 1 \\ h(x) = A(|x^3| - 5x^2 + 8|x| - 4) & \text{lorsque } 1 \leq |x| \leq 2 \\ h(x) = 0 & \text{sinon} \end{cases} \quad (2.33)$$

Le choix du paramètre A a beaucoup été étudié dans la littérature. La figure 2.7(g) illustre la fonction cubique obtenue avec $A = -1/2$. Pour cette valeur, la fonction d'interpolation possède de bonnes caractéristiques fréquentielles [89]. La réponse en fréquence est donnée par :

$$H(u) = \frac{18 - 24 \cos 2\pi u + 6 \cos 4\pi u - \pi u(2 \sin 2\pi u - \sin 4\pi u)}{(2\pi u)^4} \quad (2.34)$$

La figure 2.7(h) montre un spectre beaucoup plus proche du filtre idéal. On peut d'ailleurs observer sur la figure 2.8 que l'effet de pixelisation, caractéristique de l'interpolation au plus proche voisin, a presque disparu. Pour supprimer complètement cet effet de pixelisation, Morse *et al.* proposent de lisser les lignes de niveaux qui sont associées aux iso-valeurs [98].

2.3.3 Interpolation avec la transformée B-spline

La transformée B-spline fournit une représentation continue d'un signal discret $f(k)$ [137]. Le degré de continuité dépend directement du choix d'une fonction polynomiale de base : la fonction B-spline. Avec une fonction B-spline continue d'ordre n , on peut obtenir un signal continu $\phi^n(x)$ dérivable $n - 1$ fois de la manière suivante :

$$\phi^n(x) = \sum_{m=-\infty}^{\infty} c(m)\beta^n(x - m) = c * \beta^n(x) \quad (2.35)$$

où les $c(m)$ représentent les coefficients B-spline. La fonction B-spline d'ordre 0 est une fonction rectangle. Pour obtenir une fonction B-spline continue d'ordre n , il faut convoluer n fois la fonction rectangle par elle-même (voir figure 2.9). Cette propriété se décrit simplement par :

$$\beta^n(x) = \beta^{n-1} * \beta^0(x) = \underbrace{\beta^0 * \dots * \beta^0(x)}_{n \text{ fois}} \quad (2.36)$$

L'agrandissement de facteur a d'un signal discret $f(k)$ se fait en deux étapes [136] :

1. *Transformée B-spline directe* : Déterminer les coefficients $c(m)$ à partir de $f(k)$ et d'une fonction B-spline discrète d'ordre n .
2. *Transformée B-spline indirecte* : Calculer le signal interpolé à partir des coefficients $c(m)$ et de la fonction B-spline dilatée d'un facteur a .

Dans la suite, nous décrivons ces deux étapes.

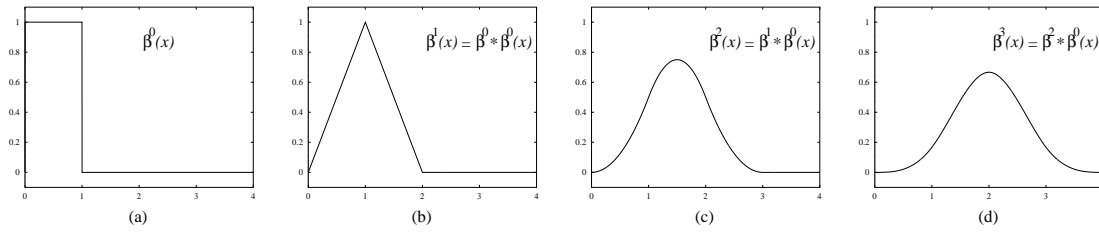


Figure 2.9 – Fonctions B-spline continues : (a) ordre 0, (b) ordre 1, (c) ordre 2, (d) ordre 3.

Transformée B-spline directe

Les coefficients $c(m)$ sont déterminés de manière à ce que la fonction $\phi^n(x)$ passe exactement par les échantillons de $f(k)$, c'est-à-dire :

$$f(k) = \phi^n(k) = \sum_{m=-\infty}^{\infty} c(m)b^n(k-m) \quad (2.37)$$

où $b^n(k)$ représente la fonction B-spline discrète définie par $b^n(k) = \beta^n(k)$. La relation (2.37) fait apparaître un produit de convolution. En passant dans le domaine fréquentiel avec la transformée de Fourier, on obtient :

$$f(k) = c * b^n(k) \xleftrightarrow{TF} F(u) = C(u) \cdot B^n(u) \quad (2.38)$$

On voit avec cette dernière expression que les coefficients B-spline peuvent être obtenus par un simple filtrage inverse du signal original :

$$C(u) = F(u) \cdot B^n(u)^{-1} = \frac{F(u)}{\sum_{k=0}^{n+1} b^n(k)e^{-2\pi iuk}} \quad (2.39)$$

Dans [138], Unser *et al.* donnent l'expression de $B^n(u)^{-1}$ pour n variant de 0 à 7, et étudient différentes techniques pour réaliser efficacement le filtrage.

Transformée B-spline indirecte

L'approche la plus simple pour agrandir $f(k)$ d'un facteur a à partir de sa représentation B-spline, consiste à intercaler $a - 1$ zéros entre les coefficients B-spline et à interpoler avec la fonction B-spline dilatée d'un facteur a :

$$g(k) = \phi^n\left(\frac{k}{a}\right) = \sum_{m=-\infty}^{\infty} c(m)b_a^n(k-am) = c^{\uparrow a} * b_a^n(k) \quad (2.40)$$

où b_a^n est définie par $b_a^n(k) = \beta^n(k/a)$ et $c^{\uparrow a}$ est définie de manière analogue à (2.26).

Une alternative à cette approche est proposée par Unser *et al.* [140]. Les coefficients B-spline de l'image agrandie $g(k)$ sont déterminés pour que l'erreur d'approximation entre la représentation B-spline continue de $f(k)$ et celle de $g(k)$ soit

minimum au sens des moindres carrés. Ceci revient à faire une projection orthogonale de la représentation B-spline continue de $f(k)$ sur l'espace des fonctions splines dilatées d'un facteur a . Plus l'ordre de la fonction B-spline est élevé, plus cette approche devient complexe à mettre en œuvre. Lee *et al.* proposent de remplacer la projection orthogonale par une projection oblique plus facile à évaluer [82]. En fait, l'approche par projection a surtout un intérêt lorsque l'on souhaite réduire une image avec sa représentation B-spline. Elle permet dans ce cas d'éviter l'effet d'aliasing qui pourrait apparaître avec l'approche par rééchantillonnage qui est illustrée avec l'expression (2.40). Dans le cas d'un agrandissement, l'approche par projection est beaucoup moins justifiée. Lorsque n est assez élevé ($n \geq 3$), l'image agrandie par projection est très semblable à celle obtenue par l'approche du rééchantillonnage [140]. Dans la suite, nous nous intéressons uniquement à cette dernière approche.

Unser *et al.* ont montré que la fonction B-spline discrète d'ordre n et dilatée d'un facteur a , pouvait s'obtenir simplement dans l'espace discret par [136] :

$$b_a^n(k) = \frac{1}{a^n} \underbrace{(b_a^0 * \dots * b_a^0)}_{n \text{ fois}} * b_1^n(k) \quad (2.41)$$

En substituant $b_a^n(k)$ dans (2.40) et en passant dans le domaine fréquentiel, on obtient :

$$G(u) = C^{\uparrow a}(u) \cdot B_a^0(u) \cdot \left(\frac{B_a^0(u)}{a} \right)^n \cdot B_1^n(u) \quad (2.42)$$

Cette expression montre que l'interpolation peut-être réalisée à partir d'une cascade de $n + 1$ filtres de largeur a réalisant une moyenne et d'un post-filtrage avec une fonction B-spline discrète d'ordre n .

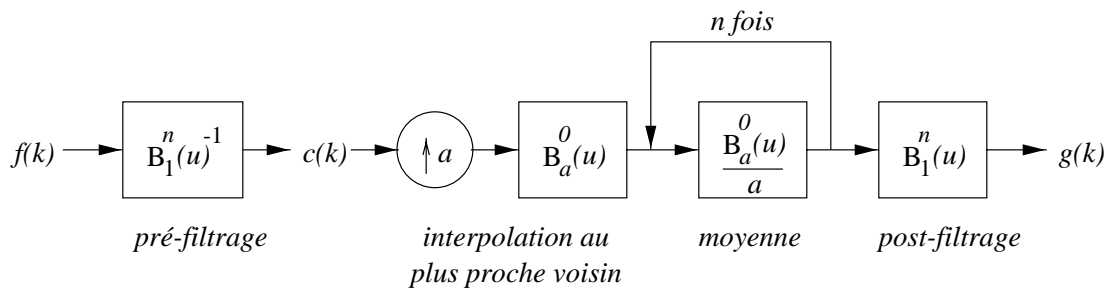


Figure 2.10 – Système global pour un agrandissement de facteur a avec la transformée B-spline d'ordre n .

Le système global d'interpolation avec la transformée B-spline est illustré sur la figure 2.10. Le signal initial $f(k)$ est d'abord filtré avec $B_1^n(u)^{-1}$ (transformée B-spline directe). Le signal $c(k)$ en sortie représente les coefficients B-spline. Il est ensuite agrandi en intercalant des zéros entre les échantillons, puis en convoluant le résultat avec une fonction rectangle. Cette opération revient à faire une interpolation au plus proche voisin de $c(k)$. Le signal est ensuite lissé en itérant n fois un filtre

réalisant une moyenne. Enfin, un dernier filtrage avec $B_1^n(u)$ (transformée B-spline indirecte) est effectué.

Pour analyser l'interpolation avec la transformée B-spline, Unser *et al.* s'intéressent à la fonction d'interpolation équivalente $h^n(x)$ qui s'applique directement sur le signal original :

$$\phi^n(x) = \sum_{m=-\infty}^{\infty} c(m)\beta^n(x-m) = \sum_{k=-\infty}^{\infty} f(k)h^n(x-k) \quad (2.43)$$

La fonction h^n est appelée *fonction spline cardinale* d'ordre n . Unser *et al.* ont montré qu'elle converge vers la fonction d'interpolation idéale (sinus cardinal) lorsque n tend vers l'infini. La figure 2.11(a) illustre la fonction spline cardinale d'ordre 3. Celle-ci est égale à zéro pour toutes les valeurs entières de x , sauf pour $x = 0$ où elle vaut 1. La méthode d'agrandissement par B-spline est donc bien une méthode d'interpolation.

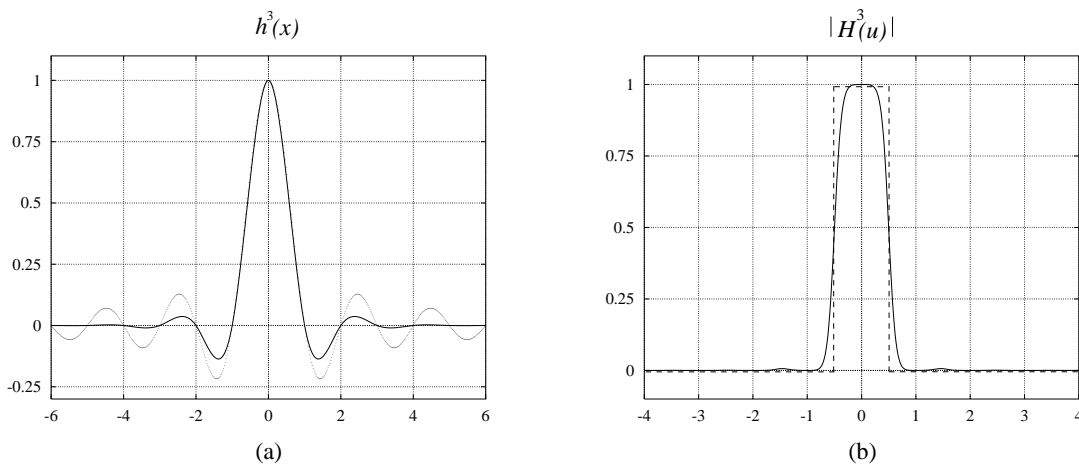


Figure 2.11 – (a) Fonction spline cardinale d'ordre $n = 3$. (b) Spectre d'amplitude.

La fonction spline cardinale d'ordre 3 réalise une bonne approximation locale de la fonction d'interpolation idéale tracée en pointillé. D'ailleurs, sur la figure 2.11(b), le spectre de h^n apparaît très proche du filtre rectangle idéal. La fonction spline cardinale décroît beaucoup plus vite vers zéro que la fonction sinus cardinal. Elle est par conséquent beaucoup moins sensible aux données distantes. Le signal interpolé avec la transformée B-spline présente donc beaucoup moins l'effet d'ondulation que celui obtenu avec la méthode du *zero-padding* (cf. section 2.3.1).

Pour appliquer aux images (signaux 2D) le système d'agrandissement décrit sur la figure 2.10, on suppose que les fonctions B-spline bidimensionnelles sont séparables. Il suffit donc d'appliquer successivement les traitements à une dimension sur les lignes puis sur les colonnes [137].

La figure 2.12 illustre les transformées B-spline¹ directe et indirecte appliquées sur l'image de San-Francisco ($n = 3$). On remarque que la transformée B-spline

1. Le code source a été écrit par Philippe Thévenaz qui fait partie de la même équipe que Michael Unser à l'EPFL de Lausanne. Ce code est disponible à l'adresse <http://bigwww.epfl.ch/thevenaz/affine.html>.

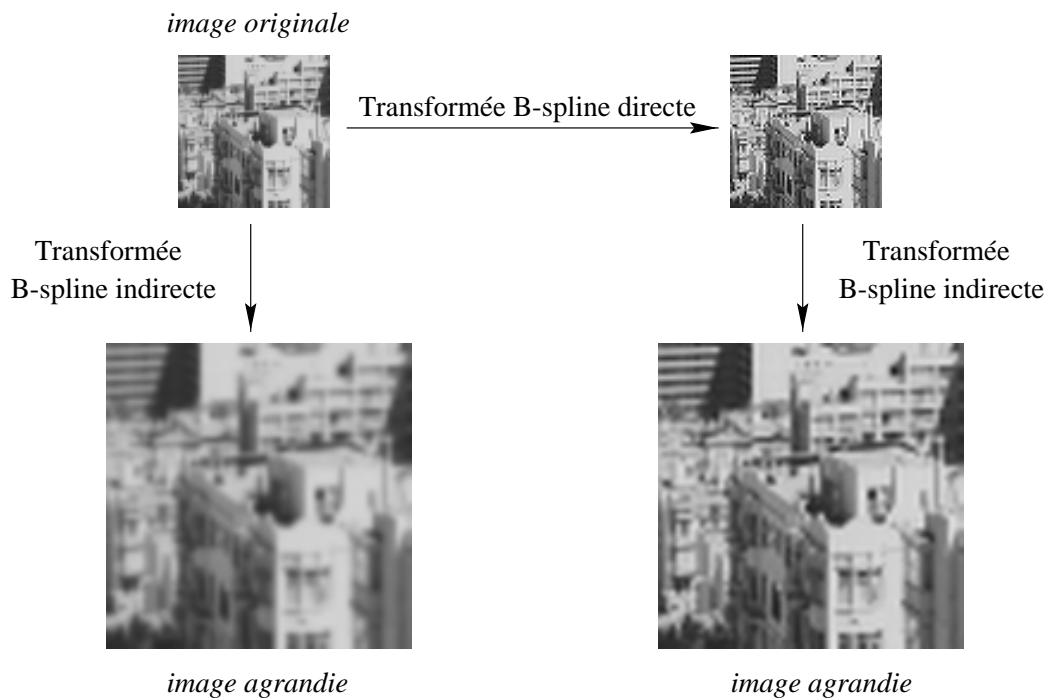


Figure 2.12 – Transformées B-spline directe et indirecte ($n = 3$).

directe amplifie les hautes fréquences de l'image originale. En fait, plus l'ordre de la B-spline est élevé et plus les hautes fréquences sont amplifiées. Cette observation s'explique par le fait que la fonction B-spline est de plus en plus lisse et aplatie lorsque n augmente (cf. figure 2.9). L'amplification des valeurs permet donc de réaliser l'équilibre avec la fonction B-spline.

Lorsque la transformée B-spline directe ne précède pas la transformée B-spline indirecte, l'image agrandie apparaît floue sur la figure 2.12. On peut donc voir la méthode d'agrandissement avec la transformée B-spline comme une méthode qui amplifie la dynamique des niveaux de gris de l'image, réalise une interpolation au plus proche voisin, puis un lissage de l'image.

La figure 2.13 montre les agrandissements de facteur 4 obtenus en faisant varier l'ordre n de la fonction B-spline. Lorsque n augmente, on tend de plus en plus vers l'interpolation idéale (fonction sinus cardinal), et l'effet d'ondulation dans l'image agrandie est de plus en plus important (voir figure 2.13(e)). Lorsque $n = 0$ et $n = 1$, le pré-filtrage et le post-filtrage montrés sur la figure 2.10 n'ont aucun effet sur l'image. Pour $n = 0$, le système revient à faire une interpolation au plus proche voisin. Et pour $n = 1$, il revient à faire une interpolation linéaire puisque la convolution de deux fonctions rectangle génère une fonction triangle (cf. figure 2.9). Il faut noter que ce résultat provient de l'approche du rééchantillonnage adoptée dans la transformée B-spline indirecte et qui est illustrée par l'expression (2.40). Avec l'approche des moindres carrés (ou projection orthogonale) [140], on obtiendrait des fonctions d'interpolation plus complexes pour $n = 0$ et $n = 1$. L'image agrandie avec $n = 3$ réalise un bon compromis entre l'effet de flou et l'effet d'ondulation.



Figure 2.13 – Agrandissement de facteur $a = 4$ avec la transformée B-spline d'ordre n .

Notes bibliographiques

La méthode d'interpolation par transformée B-spline a été introduite par Hou et Andrews en 1978 [68]. Ne sachant pas comment trouver efficacement les coefficients B-spline, ils ont proposé d'agrandir l'image en appliquant directement la fonction B-spline sur l'image originale. Cette approche revient à supprimer la transformée B-spline directe qui permet d'évaluer les coefficients B-spline. Comme on peut le voir sur la figure 2.12, elle aboutit à une image floue. Parker *et al.* ont toutefois considéré cette approche comme une méthode d'interpolation à part entière, et l'ont comparée aux méthodes d'interpolation avec des fonctions polynomiales [101]. Ils arrivent finalement à la conclusion que la méthode B-spline est moins intéressante car elle génère un flou excessif. En fait, leur comparaison n'est pas valide puisque la fonction B-spline ne préserve pas les points initiaux et ne peut pas être considérée comme une fonction d'interpolation. Pour être valide, la comparaison aurait dû se faire avec la fonction spline cardinale. Maeland a été le premier à dénoncer la «fausse interpolation» avec la fonction B-spline et à rectifier les conclusions erronées de Parker *et al.* en effectuant la comparaison avec la fonction spline cardinale [89].

Sa conclusion est que cette dernière réalise une meilleure interpolation que la fonction polynomiale cubique (cf. section 2.3.2). Sankar *et al.* ont proposé en 1988 une approche pour obtenir efficacement les coefficients B-spline et réaliser une véritable interpolation [117]. En fait, Unser *et al.* ont relevé une erreur dans leur modèle qui a finalement conduit à un résultat proche de celui de Hou et Andrews [136]. En définitive, il a fallu attendre 1991 pour que Unser *et al.* donnent un algorithme valide et efficace pour interpoler une image avec la transformée B-spline [136]. C'est cette approche que nous avons présentée précédemment.

2.4 Amélioration de la netteté

Dans la section précédente, nous avons étudié différentes méthodes d'agrandissement qui préservent plus ou moins bien les fréquences de l'image originale. Nous avons vu que cette préservation pouvait se réaliser simplement par un filtrage linéaire des hautes fréquences lors de l'agrandissement. Cependant, même avec la méthode d'interpolation idéale, on peut observer que l'image agrandie présente toujours un effet de flou. Celui-ci est d'autant plus important que le facteur d'agrandissement est élevé. Ce problème est typique des filtres linéaires qui ont tendance à lisser les contours. Pour le corriger, il faut effectuer un post-traitement de l'image pour rétablir la netteté des contours. Pour cela, quelques approches ont été proposées dans la littérature. Nous étudions dans cette section deux approches caractéristiques. La première restaure les contours à partir de l'expression d'une équation différentielle partielle. L'algorithme de restauration exploite la représentation B-spline de l'image originale (cf. section 2.3.3). La deuxième approche procède par diffusion anisotrope de l'information depuis l'intérieur des régions homogènes vers les frontières. On peut voir cette approche comme une sorte de rehaussement des contrastes réalisée sur une image floue.

2.4.1 Utilisation d'une équation différentielle partielle

Prades-Nebot *et al.* ont proposé récemment un algorithme de restauration des contours d'une image agrandie à partir d'une représentation B-spline [107]. L'algorithme est basé sur l'équation différentielle partielle suivante :

$$\frac{\partial f(x, t)}{\partial t} = - \left| \frac{\partial f(x, t)}{\partial x} \right| \operatorname{sgn} \left(\frac{\partial^2 f(x, t)}{\partial x^2} \right) \quad (2.44)$$

où la fonction $\operatorname{sgn}(x)$ est la fonction *signe* qui renvoie -1 si $x < 0$, 0 si $x = 0$ et 1 si $x > 0$. La partie droite de l'équation (2.44) est composée d'un *terme de propagation* $|f_x|$ et d'un *terme de détection du contour* $\operatorname{sgn}(f_{xx})$. Le terme de propagation détermine le degré d'amplification de la valeur en chaque point de f . Le rôle du terme de détection du contour est de changer le signe du terme de propagation pour que le flux local soit dirigé vers le contour.

En supposant que la fonction initiale $f(x, 0)$ est une fonction continue, alors l'application itérée de l'expression (2.44) sur cette fonction conduit à une fonction

stable $f(x, \infty)$ comportant des discontinuités partout où $\partial^2 f(x, 0)/\partial x^2 = 0$. La figure 2.14 montre un exemple emprunté à Prades-Nebot *et al.* qui illustre l'équation différentielle partielle (2.44) sur une itération de l'algorithme. On voit notamment que la fonction tracée sur la figure 2.14(d) permet de restaurer les contours lissés du signal original f montré sur la figure 2.14(a)

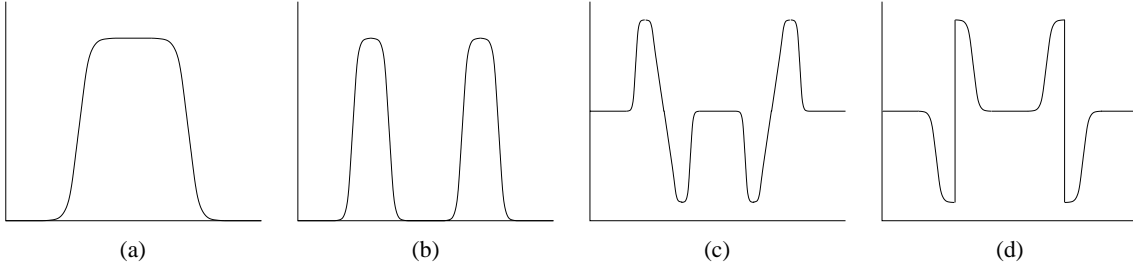


Figure 2.14 – (a) Signal f contenant deux contours lissés. (b) Illustration de $|f_x|$. (c) Illustration de f_{xx} . (d) Illustration de $-|f_x|\text{sgn}(f_{xx})$.

Prades-Nebot *et al.* appliquent l'algorithme de restauration des contours sur la représentation B-spline continue $\phi^n(x)$ définie par (2.35), du signal discret $f(k)$ à agrandir. Ainsi, en prenant $f(x, 0) = \phi^n(x)$, on obtient [107] :

$$\frac{\partial f(x, 0)}{\partial t} = - \left| \sum_{m=-\infty}^{\infty} c_1(m) \beta^{n-1} \left(x - m + \frac{1}{2}\right) \right| \text{sgn} \left(\sum_{m=-\infty}^{\infty} c_2(m) \beta^{n-2} (x - m) \right) \quad (2.45)$$

avec $c_1(m) = c(m) - c(m-1)$ et $c_2(m) = c(m-1) - 2c(m) + c(m+1)$.

En approximant $\partial f(x, 0)/\partial t$ par $(f(x, t + \Delta t) - f(x, t))/\Delta t$ et en discrétisant x , on obtient $f(k, \Delta t) = f(k, 0) - e(k)$ où $e(k)$ est un terme de correction défini par :

$$e(k) = \Delta t \left| \sum_{m=-\infty}^{\infty} c_1(m) \beta^{n-1} \left(k - m + \frac{1}{2}\right) \right| \text{sgn} \left(\sum_{m=-\infty}^{\infty} c_2(m) \beta^{n-2} (k - m) \right) \quad (2.46)$$

Ainsi, en soustrayant le terme de correction $e(k)$ à l'image interpolée avec la représentation B-spline $f(k, 0)$, on obtient un signal $f(k, \Delta t)$ avec des contours plus nets. Pour réaliser la correction uniquement sur les contours, Prades-Nebot *et al.* préconisent d'utiliser un seuil sur le terme de propagation $|f_x|$.

Il faut remarquer que le signal stable $f(k, \infty)$ est ici remplacé par le signal $f(k, \Delta t)$ qui est obtenu en une seule itération. Le degré de restauration des contours est donc directement lié au choix de Δt . On remarquera que l'algorithme ne peut s'appliquer que si l'ordre de la B-spline initiale est supérieur ou égal à trois afin de garantir les dérivées partielles première et seconde. Pour une B-spline cubique, le terme de correction défini par (2.46) se simplifie pour donner [107] :

$$e(k) = -\frac{\Delta t}{2} (c(k+1) - c(k-1)) \text{sgn}(c_2(k)) \quad (2.47)$$

Pour étendre l'algorithme aux images (signaux 2D), Prades-Nebot *et al.* considèrent l'équation différentielle partielle suivante :

$$f_t = -\sqrt{f_x^2 + f_y^2} \text{sgn}(L(f)) \quad (2.48)$$

où $L(f)$ est un terme permettant de détecter les contours. Pour obtenir rapidement le terme de correction, Prades-Nebot *et al.* appliquent des masques de type Sobel ou Prewitt directement sur les coefficients B-spline. Les résultats expérimentaux obtenus dans [107] semblent plutôt concluants. Le terme de correction améliore la netteté des images obtenues à partir d'une représentation B-spline. Le seul problème notable de la méthode concerne les ondulations caractéristiques des fonctions B-splines cardinales d'ordre élevé (cf. figure 2.13). Les ondulations peuvent être considérées comme des contours à restaurer alors qu'elles sont dues uniquement à un artefact de la préservation fréquentielle. Dans ce cas, la méthode de Prades-Nebot *et al.* amplifie les ondulations.

2.4.2 Diffusion anisotropique

Le flou apparaissant sur les contours d'une image peut être supprimé en utilisant les principes d'une *diffusion anisotropique*, modèle initié par Perona et Malik [103], [116]. Il s'agit d'une approche itérative où l'image est convoluée avec un filtre réalisant un lissage adaptatif. En fait, les poids du filtre évoluent dans l'espace de manière à ce que le lissage soit maximum dans une zone homogène et minimum (voire nul) sur un contour. Après convergence, l'image obtenue est composée de régions uniformes séparées par des contours nets et précis.

Dans cette section, nous étudions une approche proposée par Albiol *et al.* qui adaptent le concept de diffusion anisotropique à la suppression du flou dans une image agrandie par préservation fréquentielle [8]. Les auteurs ont appliqué leur méthode sur une image obtenue par interpolation linéaire (cf. section 2.3.2), mais *a priori* rien n'empêche de l'appliquer sur une image obtenue avec une méthode plus évoluée, comme par exemple la méthode d'interpolation avec la transformée B-spline (cf. section 2.3.3).

Contrairement à la méthode classique de diffusion anisotropique [116], la méthode d'Albiol *et al.* n'est pas itérative: chaque pixel n'est traité qu'une seule fois. La diffusion de l'information est réalisée ici par un traitement ordonné des pixels de l'image. En fait, les pixels sont traités successivement en prenant en priorité les pixels centrés sur une région homogène et en terminant par ceux qui sont proches des contours. On réalise ainsi une diffusion de l'information depuis l'intérieur des régions vers leurs frontières, comme illustré sur la figure 2.15(a).

L'algorithme général peut se résumer par les étapes suivantes :

1. Agrandissement de l'image originale avec une méthode d'interpolation quelconque.
2. Calcul du gradient en chaque pixel de l'image agrandie et constitution d'une liste de pixels ordonnée selon la valeur du gradient.
3. Extraction du pixel de plus faible gradient de la liste et calcul de sa nouvelle valeur.
4. Retour à l'étape 3 tant que la liste n'est pas vide.

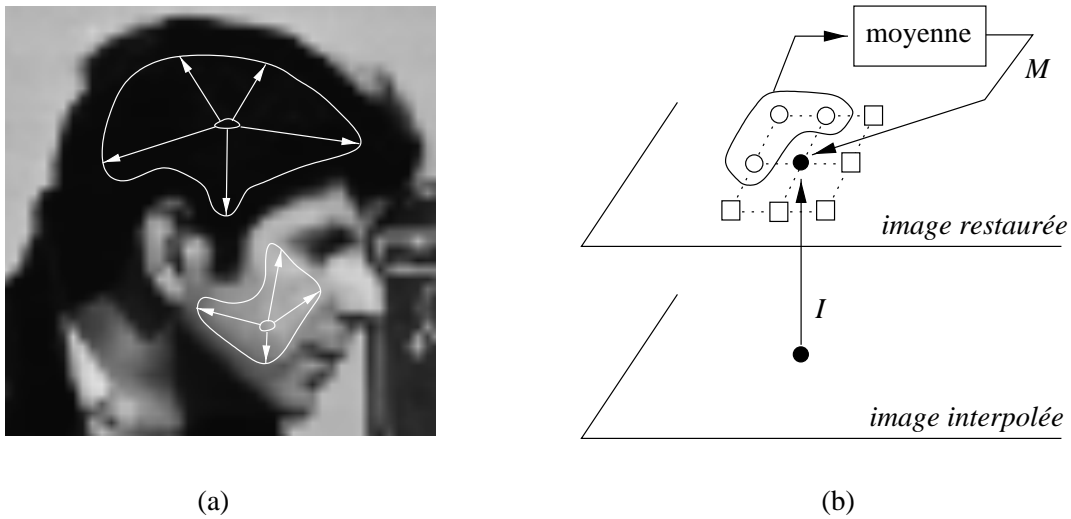


Figure 2.15 – (a) Principe de diffusion anisotropique. (b) Traitement d'un pixel (\bullet) : sa nouvelle valeur est une combinaison de sa valeur I dans l'image interpolée et de la moyenne M de ses pixels voisins déjà traités (\circ).

Dans l'étape 2, la mesure du gradient agit un peu comme une mesure de probabilité d'appartenance du pixel à un contour. Le gradient est maximum lorsque le pixel appartient à un contour, et il est nul lorsque le pixel appartient à une zone uniforme. Le classement des pixels selon la valeur du gradient doit donc permettre de réaliser la diffusion de l'information depuis l'intérieur des régions vers leurs frontières. En pratique, le gradient peut-être calculé par morphologie mathématique ou plus simplement à l'aide d'un opérateur de type Sobel.

Dans l'étape 3, la nouvelle valeur du pixel x résulte d'une combinaison entre sa valeur $I(x)$ dans l'image interpolée et la moyenne $M(x)$ des valeurs de ses pixels voisins déjà traités (voir figure 2.15(b)). Les poids dans la combinaison dépendent directement du gradient $G(x)$. Plus le gradient est faible et plus la valeur du pixel tend vers $I(x)$; plus il est fort est plus la valeur est proche de $M(x)$. La nouvelle valeur du pixel dans l'image restaurée est donc obtenue par l'expression :

$$R(x) = N(G(x)) \cdot M(x) + (1 - N(G(x))) \cdot I(x) \quad (2.49)$$

où la fonction N permet de normaliser la valeur du gradient dans l'intervalle $[0, 1]$. Il s'agit d'une fonction linéaire qui est définie par :

$$\begin{cases} N(x) = \alpha \frac{G(x)}{\max_y G(y)} & \text{lorsque } G(x) < \frac{\max_y G(y)}{\alpha} \\ N(x) = 1 & \text{sinon} \end{cases} \quad (2.50)$$

où α est un paramètre dont le but est de compresser plus ou moins la fonction de normalisation.

A travers l'expression (2.49), on remarque qu'un pixel situé à l'intérieur d'une région homogène garde la valeur donnée par l'image interpolée, tandis qu'un pixel

proche d'un contour dépend plutôt de ses voisins. On voit donc qu'il y a une diffusion de la valeur des pixels centrés à l'intérieur des régions, et que cette diffusion est dirigée vers les contours. Son importance est contrôlée par le paramètre α qui agit directement sur les poids de la combinaison dans (2.49). Plus ce paramètre est élevé et plus les régions dans l'image restaurée deviennent uniformes avec des contours nets et précis. Albiol *et al.* préconisent pour α une valeur située entre 2 et 4 afin d'obtenir des contours nets dans la majorité des cas.

La méthode de diffusion anisotropique permet d'améliorer considérablement la netteté des images agrandies par préservation fréquentielle. Cependant, elle souffre de deux défauts. Le premier est qu'elle a tendance à supprimer les structures fines de l'image originale. Par exemple, une ligne blanche sur un fond noir va être attaquée par deux fronts d'ondes qui propagent une valeur identique. Le résultat est la disparition complète de la ligne. Pour remédier à ce problème, les auteurs mettent en place une détection des structures fines à l'aide d'outils morphologiques. Les pixels qui appartiennent à ces structures sont traités en priorité avec les pixels de gradient minimum dans l'algorithme de restauration donné plus haut. Le deuxième défaut, plus problématique que le premier, est que la méthode de diffusion anisotropique améliore la netteté de *tous* les contours, y compris ceux qui sont déjà lissés dans l'image originale. De plus, la netteté d'un contour dépend davantage des propriétés des pixels «germe» (gradient minimum) que des propriétés locales de ce contour.

Pour terminer, il faut noter que dans les zones de l'image qui sont fortement texturées, la valeur du gradient est généralement faible. Ainsi, les pixels issus de ces zones sont traités en priorité avec ceux issus des zones homogènes. Par conséquent, la méthode de diffusion anisotropique a plutôt tendance à lisser les textures présentes dans l'image.

2.5 Conclusion

Nous avons présenté dans ce chapitre différentes méthodes d'agrandissement qui préservent plus ou moins bien les fréquences de l'image originale. La méthode d'agrandissement du *zero-padding* préserve exactement les fréquences de l'image originale, mais elle possède deux défauts. Le principal est qu'elle introduit un effet d'ondulation des contours dans l'image agrandie. Le second est qu'elle nécessite une quantité de calculs non négligeable puisqu'il faut effectuer une transformée de Fourier directe, insérer des coefficients nuls pour les hautes fréquences, puis effectuer une transformée de Fourier inverse. Toutefois, nous avons vu que cette opération pouvait être réalisée de manière équivalente dans le domaine spatial en convoluant l'image avec une fonction sinus cardinal.

Cette équivalence spatiale nous a permis de montrer que l'on pouvait limiter l'effet d'ondulation et diminuer fortement la quantité de calculs en convoluant l'image avec une fonction à petit support. Cette amélioration se fait au détriment de la préservation exacte des fréquences. Ceci nous a amené à présenter deux nouvelles méthodes : la méthode d'interpolation avec des fonctions polynomiales et la méthode

d'interpolation avec la transformée B-spline. D'un point de vue théorique cette dernière méthode est la plus aboutie des deux puisqu'elle fait le lien avec la méthode du *zero-padding*. Lorsque l'ordre de la B-spline augmente, la fonction spline cardinale tend vers la fonction sinus cardinal. Avec une B-spline cubique, le support de la fonction spline cardinale est suffisamment réduit pour que l'agrandissement soit rapide et que l'effet d'ondulation ne soit plus perceptible. La méthode d'interpolation avec la transformée B-spline cubique apparaît donc clairement comme la méthode de référence de ce chapitre.

L'agrandissement par préservation fréquentielle est adapté lorsque le théorème d'échantillonnage est vérifié. En pratique, c'est très rarement le cas pour les images naturelles. La conséquence est l'apparition d'un effet de flou, notamment sur les contours. Nous avons présenté deux approches pour améliorer la netteté des images agrandies : la première est basée sur l'expression d'une équation différentielle partielle et la seconde sur un procédé de diffusion anisotropique. Bien que ces approches améliorent effectivement la netteté, celle-ci est plutôt artificielle. De plus, les dégradations engendrées par les méthodes d'agrandissement qui préservent les fréquences sont souvent irréversibles. En fait, il faut corriger le problème à sa source. La préservation unique des fréquences initiales est insuffisante pour les images naturelles, il faut également créer des hautes fréquences pour maintenir les contours et les textures. Dans le chapitre suivant, nous étudions des méthodes d'agrandissement qui ont cet objectif.

Chapitre 3

Méthodes d'agrandissement préservant les structures

3.1 Introduction

Les images de scènes naturelles contiennent généralement des objets dont les contours représentent des discontinuités dans la scène réelle. D'un point de vue fréquentiel, ces discontinuités se traduisent par une infinité de fréquences. Les méthodes d'agrandissement que nous avons présentées dans le chapitre 2 ne font que préserver (dans le meilleur des cas) les fréquences de l'image originale. Elles ne peuvent donc pas créer les hautes fréquences manquantes qui sont nécessaires à la représentation des discontinuités dans l'image agrandie. Le résultat de cette incapacité est la génération d'un flou visuellement gênant à l'emplacement des contours. Le flou n'apparaît pas uniquement au niveau des contours, il peut aussi apparaître au niveau des textures. Celles-ci se caractérisent également très souvent par une infinité de fréquences.

Pour dépasser les limites des méthodes classiques, comme la méthode du *zero-padding*, de nouvelles méthodes permettant la création de hautes fréquences ont été proposées depuis une dizaine d'années. Généralement, ces méthodes laissent de côté le problème de la synthèse des textures, très difficile à appréhender, pour ne prendre en compte que celui des contours. Toutefois, même le problème de la préservation des contours n'est pas aussi facile à résoudre qu'on pourrait le penser. En effet, les contours de l'image originale ne traduisent pas forcément tous une discontinuité entre des objets. Certains contours peuvent résulter d'une déformation continue sur la surface même d'un objet. Ce type de contour ne doit pas être préservé comme s'il s'agissait d'une discontinuité. Dans le cas contraire, on obtiendrait des fausses discontinuités dans l'image agrandie aussi sémantiquement gênantes que le flou occasionné par les vraies discontinuités non préservées. Nous verrons plus loin que l'analyse multi-résolution est une voie de recherche intéressante et prometteuse

pour discerner et synthétiser les différents types de contours.

Parmi les méthodes d'agrandissement qui essaient de préserver les structures, on peut distinguer deux grandes catégories de méthodes. La première catégorie opère directement sur les images floues produites par les méthodes d'agrandissement qui préservent les fréquences originales (cf. chapitre 2). On ne peut pas réellement parler de méthodes d'agrandissement mais plutôt de méthodes d'amélioration de la netteté. Nous les avons étudiées dans le chapitre précédent en section 2.4.

La deuxième catégorie de méthodes utilise des modèles d'agrandissement originaux. Parmi celles-ci, on peut distinguer essentiellement trois approches. La première, qui est la plus répandue, consiste à préserver les contours en faisant de l'interpolation adaptative. Le modèle le plus représentatif de ce type de méthode consiste à réaliser une interpolation linéaire classique à l'intérieur d'une région homogène et à réaliser une interpolation directionnelle au voisinage du contour. Nous étudions cette approche particulière de l'interpolation dans la section 3.2. La deuxième approche exploite les corrélations qui existent entre les bandes de fréquences d'une image (principalement sur les contours) pour extrapoler les hautes fréquences de l'image agrandie. Cette approche par extrapolation fréquentielle est étudiée dans la section 3.3. La troisième approche, très différente des deux premières, est issue de la compression d'images par fractale. En exploitant l'indépendance du code fractal vis-à-vis de la résolution, on peut réaliser facilement un agrandissement de l'image. Cette technique, souvent désignée par le terme *zoom fractal*, est étudiée dans la section 3.4. En marge de ces trois principales approches, il en existe d'autres, moins répandues, que nous résumons dans la section 3.5.

Une procédure simple pour juger une méthode d'agrandissement consiste à réduire une image puis à l'agrandir. On peut alors effectuer des comparaisons entre l'image originale, qui sert de référence, et l'image réduite agrandie. Puisque ces deux images ont la même taille, on peut facilement effectuer des mesures mathématiques qui relèvent les différences entre les deux images.

Lorsque l'on effectue réellement un agrandissement, on ne dispose pas de référence. Dans ce cas, pour évaluer la qualité de l'image agrandie, on peut en faire une analyse spectrale. À partir du spectre fréquentiel, on peut savoir si une méthode d'agrandissement produit des hautes fréquences et si celles-ci sont corrélées avec les basses fréquences. Les mesures mathématiques sur le spectre ne sont pas forcément significatives, aussi une analyse spectrale visuelle se révèle souvent plus pertinente. Ces différentes techniques qui permettent d'évaluer la qualité d'un agrandissement sont présentées dans la section 3.6.

3.2 L'interpolation adaptative

Dans la section 2.3.2, nous avons vu que l'interpolation d'une image pouvait être réalisée en intercalant des échantillons nuls entre les pixels originaux et en effectuant une convolution avec un filtre spatial à petit support (cf. expression (2.25)). Ce

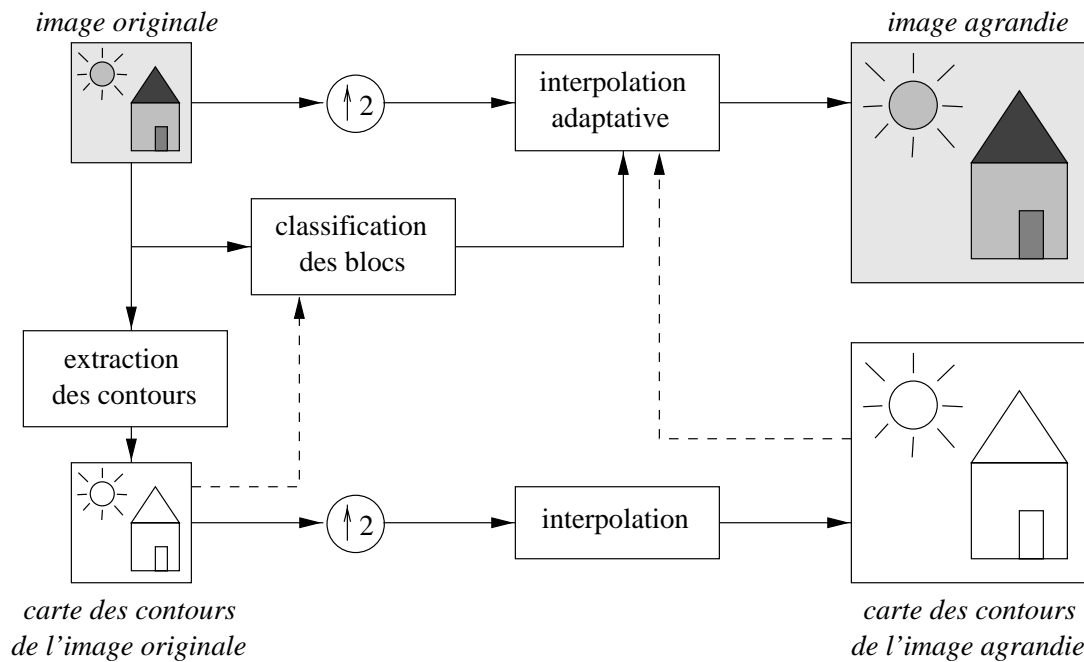


Figure 3.1 – Schéma standard des méthodes d'interpolation adaptative.

filtre a pour objectif de préserver le contenu fréquentiel de l'image originale. Il est particulièrement adapté dans les zones de faible variation d'intensité, comme par exemple dans celles qui sont homogènes. Dans ce cas, il permet de préserver leurs basses fréquences caractéristiques. Par contre, il est particulièrement inadapté dans les zones où la variation de l'intensité est importante, comme par exemple dans celles comportant des contours. Dans ces zones qui contiennent des hautes fréquences, il faut créer des hautes fréquences supplémentaires dans l'image agrandie sous peine d'y faire apparaître du flou. Les filtres linéaires décrits dans le deuxième chapitre ne peuvent pas réaliser cette opération puisqu'ils sont construits uniquement pour préserver des fréquences existantes.

Les méthodes d'interpolation adaptative proposent de résoudre ce problème en adaptant le filtre spatial au contenu de la zone filtrée. On peut trouver deux approches. La première consiste à utiliser un filtre non-linéaire unique qui va réellement changer ses poids en fonction du contenu de l'image¹. La seconde approche, beaucoup plus répandue, consiste à disposer de plusieurs modèles d'interpolation et à utiliser celui qui convient le mieux dans la zone d'interpolation traitée². Dans la suite, nous nous intéressons plus particulièrement à cette deuxième approche de l'interpolation adaptative. Son schéma standard est décrit sur la figure 3.1.

Le modèle d'interpolation adapté au bloc traité (voisinage pris en compte dans l'interpolation d'un pixel) est choisi en fonction d'une certaine classification. De manière générale, on peut comptabiliser cinq blocs caractéristiques qui sont illustrés sur

1. Voir [17], [30], [111], [130], [133], [147].

2. Voir [9], [10], [15], [18], [45], [46], [52], [74], [75], [131], [132], [142], [143], [144].

la figure 3.2 avec leurs spectres fréquentiels. Il y a tout d'abord le *bloc homogène* qui rassemble les blocs ayant une variation d'intensité plutôt faible. Comme le montre la figure 3.2(b), l'énergie spectrale dans ce type de blocs est plutôt concentrée dans les basses fréquences³. Une interpolation classique qui se contente de préserver les fréquences originales est dans ce cas adaptée. Les méthodes d'interpolation adaptative s'intéressent en particulier aux *blocs à contour simple* tels qu'on peut le voir sur la figure 3.2(c). Ce type de bloc est composé de deux régions significatives avec des niveaux de gris relativement différents. En général, les méthodes simplifient sa modélisation en considérant que la séparation est rectiligne. Cette hypothèse est à peu près vérifiée en prenant des blocs de taille suffisamment petite (par exemple 4×4 pixels) : un contour curviligne peut localement être approximé par un segment de droite. Le spectre sur la figure 3.2(d) montre une trace fréquentielle orientée caractéristique d'une direction privilégiée dans le bloc (la trace est orthogonale à cette direction). La majorité des méthodes d'interpolation adaptative ne considère que ces deux premiers types de blocs : bloc homogène et bloc à contour simple [74], [75], [142].

Il existe cependant des méthodes qui distinguent d'autres types de blocs comme les *blocs à contours multiples* et les *blocs texturés* [131], [132], [143]. Un bloc à contours multiples est un bloc qui contient plus de deux régions significatives (voir figure 3.2(e) pour un exemple de bloc à trois régions). Le spectre montre des traces associées aux différents contours qui traduisent les frontières entre les régions. On peut dire que dans ce type de bloc, une certaine organisation se dégage de l'image spectrale. Au contraire, dans un bloc texturé comme sur la figure 3.2(g), on a plutôt une sorte de chaos dans les structures et le spectre ne contient pas de trace particulière (voir figure 3.2(h)). On peut quand même noter que l'énergie spectrale est ici répartie sur tout le spectre, contrairement au spectre du bloc homogène sur la figure 3.2(b).

Enfin, on peut également trouver des méthodes qui distinguent une cinquième catégorie de blocs : le bloc à contour isolé [45], [46]. Ce type de bloc est illustré sur la figure 3.2(i). La différence entre celui-ci et le bloc à contour simple représenté sur la figure 3.2(c) est que dans le premier cas, le contour a une existence réelle, tandis que dans le second, il marque une frontière entre deux régions ayant des niveaux de gris différents. Sur le spectre d'un bloc à contour isolé (voir figure 3.2(j)), on trouve toujours une trace orthogonale au contour.

Nous décrivons la classification et l'interpolation de ces différents types de blocs dans les sections 3.2.1 et 3.2.2.

Comme on peut le voir sur la figure 3.1, certaines méthodes d'interpolation adaptative exploitent une carte des contours haute-résolution (estimation sous-pixel) pour classifier et interpoler les blocs [10], [18], [130], [147]. Les changements d'intensité dans une image peuvent être détectés en évaluant le laplacien. Dans [147], le

3. En général, les traces verticale et horizontale d'un spectre ne doivent pas être prises en compte. La périodisation implicite de l'image dans la transformée de Fourier engendre obligatoirement des discontinuités artificielles situées sur les bords (cf. section 2.2.3). Cet artefact n'est cependant pas visible lorsque le pourtour de l'image a un niveau de gris constant, ce qui est rarement le cas.

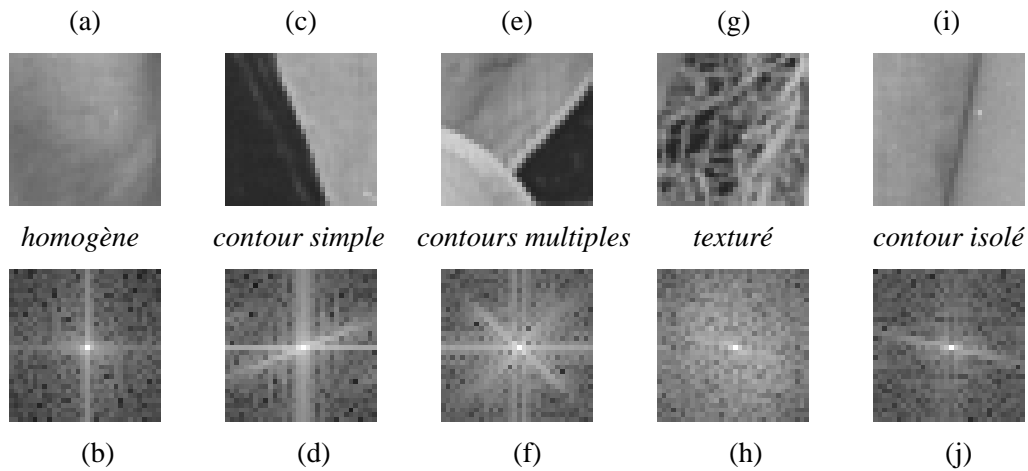


Figure 3.2 – Blocs caractéristiques dans une image.

laplacien de l'image originale est obtenu par une convolution avec un filtre dont les poids sont calculés en discrétisant le laplacien d'une fonction gaussienne. Dans [10], le filtre utilisé a un poids positif au centre et des poids négatifs en périphérie (*center-on surround-off*), ce qui constitue une approximation grossière du laplacien. Dans [18], ce filtre est remplacé par un autre qui satisfait mieux la propriété d'invariance en rotation et qui est plus proche du laplacien d'une gaussienne.

Pour obtenir la carte des contours haute-résolution à partir de celle de l'image originale, certains réalisent une estimation par interpolation linéaire [10], d'autres une estimation par interpolation cubique [18], [147].

3.2.1 Bloc homogène et bloc à contour simple

Les méthodes d'interpolation adaptative s'intéressent principalement à deux types de blocs : le bloc homogène et le bloc à contour simple. Dans la suite, nous synthétisons les différentes manières de les classifier et de les interpoler rencontrées dans la littérature (cf. figure 3.2).

Classification

Un bloc est qualifié d'homogène selon différents critères. Le plus répandu est celui de l'amplitude du gradient qui est généralement évaluée à partir de l'opérateur de Sobel [9], [15], [45], [46], [52]. Si l'amplitude est inférieure à un certain seuil, alors la variation de l'intensité à l'intérieur du bloc est jugée trop faible et le bloc est classé comme étant homogène. D'autres mesures semblables à celle du gradient sont utilisées. On trouve par exemple dans [142] une approche où la différence normalisée entre les valeurs extrêmes du bloc est utilisée comme critère de classement. Tandis que dans [143] et [144], c'est le rapport entre la différence des valeurs extrêmes du bloc et celle des valeurs extrêmes de l'image qui est employé.

Pour classer un bloc dans la catégorie des blocs à contour simple, le premier critère

est que celui-ci ne soit pas homogène. Le second critère qui détermine la décision varie selon les méthodes d'interpolation adaptative. Ainsi, dans [131] et [132], le gradient est calculé sur tous les pixels du bloc. Si une direction est dominante, alors on suppose qu'un contour est présent dans le bloc. Dans [142], on trouve un critère assez similaire puisque c'est la différence des valeurs maximales sur les lignes orientées dans une même direction qui est exploitée. Si ces différences sont inférieures à un certain seuil, qui dépend de la différence entre les valeurs extrêmes du bloc, alors le bloc est supposé orienté. Les méthodes d'interpolation adaptative qui exploitent une carte des contours utilisent également celle-ci pour détecter la présence d'un contour simple dans un bloc [10], [18], [147].

Interpolation

Pour les blocs homogènes, une interpolation classique qui préserve les fréquences initiales du bloc est suffisante (cf. chapitre 2). Bien que l'interpolation linéaire ait des caractéristiques relativement pauvres d'un point de vue spectral, la plupart des méthodes l'utilise pour interpoler un bloc homogène. Il existe néanmoins une exception où l'interpolation cubique est utilisée [45], [46].

Pour interpoler les blocs à contour simple, on trouve de nombreux modèles. On peut tout d'abord distinguer les méthodes qui exploitent une carte des contours de l'image à un niveau «sous-pixel». Celles-ci réalisent le plus souvent soit une interpolation linéaire [10], soit une interpolation cubique [130], [147]. La seule différence avec l'interpolation classique est que l'interpolation d'un pixel est ici réalisée uniquement avec les pixels voisins qui sont du même côté du contour. Dans [18], la valeur du pixel à interpoler est obtenue en utilisant un modèle d'interpolation polynomiale relativement original. Ce modèle est basé sur l'utilisation de deux polynômes qui sont raccordés au niveau du contour estimé (haute-résolution) et dont les pentes respectives dépendent de contraintes de continuité imposées par les pixels du bloc. L'avantage de ce modèle plus complexe est de préserver les caractéristiques des contours qui ne sont pas forcément nets dans l'image originale : un contour n'est pas toujours l'expression d'une discontinuité dans la scène réelle.

Parmi les méthodes qui n'exploitent pas de carte des contours, on peut discerner celles qui appliquent un modèle de surface adapté aux contours dans un espace continu et celles qui procèdent plus classiquement par filtrage discret directionnel. Dans la première catégorie de méthodes, on trouve une approche où le bloc est modélisé par deux niveaux de gris séparés par un segment de droite continu [74], [75]. Les deux niveaux de gris et la position de la droite sont déterminés par une technique des moindres carrés dans un espace continu. Suivant sa position vis-à-vis du segment de droite, le pixel à interpoler est affecté de l'un des deux niveaux de gris. Si celui-ci est traversé par le segment, alors sa valeur résulte d'une combinaison linéaire des deux niveaux de gris. Dans [143] et [144], on trouve une généralisation du modèle précédent où les niveaux de gris du bloc sont modélisés avec des polynômes plus ou moins complexes. Comme dans [18], cette approche permet de préserver l'aspect des contours qui ne sont pas nets. Une autre approche exploite le gradient pour estimer

l'orientation de la ligne de niveau des iso-valeurs qui traverse le pixel à interpoler [9], [52]. La valeur de ce dernier est alors obtenue à partir d'une estimation de l'iso-valeur associée à la ligne de niveau. Cette approche est également exploitée dans [15] sur des bandes de fréquences particulières de l'image originale (voir en section 3.2.2). Enfin, on peut trouver une approche basée sur un polynôme d'extrapolation (formule de Gregory-Newton) [45], [46]. Il ne s'agit plus d'une interpolation le long du contour, mais d'une extrapolation dirigée vers le contour.

Dans la deuxième catégorie de méthodes (filtrage discret), on trouve notamment une approche basée sur l'utilisation de douze filtres directionnels [131], [132]. Le filtre est choisi en fonction de l'orientation du contour estimée à partir du gradient calculé en chaque pixel du bloc : douze orientations sont définies dans un bloc de taille 4×4 pixels. Les filtres sont construits de manière à réaliser une interpolation parallèle au contour. On retrouve à peu près la même idée dans [142] avec une interpolation selon les droites discrètes parallèles à l'orientation estimée du contour. L'interpolation se fait en réalisant une moyenne des deux ou des quatre plus proches voisins selon l'orientation des droites et de la position du pixel à interpoler.

3.2.2 Autres types de blocs

En dehors du bloc homogène et du bloc à contour simple, les méthodes d'interpolation adaptative considèrent parfois d'autres types de blocs. De manière globale, on peut distinguer trois blocs supplémentaires : le bloc à contours multiples, le bloc texturé et le bloc à contour isolé (cf. figure 3.2).

Bloc à contours multiples et bloc texturé

Dans quelques méthodes d'interpolation adaptative, les blocs texturés et les blocs à contours multiples sont classés par défaut dans la même catégorie que les blocs homogènes [45], [46], [74], [75], [130], [142], [144]. Dans ce cas, le modèle d'interpolation utilisé est le plus souvent une simple interpolation linéaire.

Dans d'autres méthodes, les blocs à contours multiples et les blocs à contour simple sont indifférenciés et un unique modèle est utilisé dans les deux cas. Par exemple, dans [10] et [18], la notion de bloc à contours multiples disparaît en raison de la taille minimale des blocs d'analyse (2×2 pixels).

Dans [15], les blocs à contours multiples sont décomposés en blocs à contour simple. Le principe de la méthode est de décomposer initialement l'image en huit composantes directionnelles grâce à huit filtres fréquentiels : chacun conserve une bande orientée particulière du spectre. Les huit composantes sont alors agrandies par interpolation directionnelle et recombinaées pour former l'image agrandie finale.

On retrouve dans [17] une méthode qui est dans le fond assez proche de la précédente. Son but est d'éviter la classification en déterminant les poids du filtre spatial adapté au bloc. Le principe est d'assigner, à chaque pixel de l'image originale, huit poids directionnels évalués à partir d'opérateurs dédiés. Ces poids sont ensuite interpolés pour générer les poids directionnels associés aux pixels de l'image agrandie.

Un pixel de l'image agrandie est finalement obtenu en combinant les valeurs interpolées dans chaque direction affectées de leur poids directionnel respectif. En présence d'un contour simple, le poids directionnel associé à la direction du contour domine et l'interpolation se fait parallèlement au contour. Si le bloc est homogène, tous les poids sont à peu près semblables et l'interpolation se réduit à une simple moyenne des pixels voisins. Pour un bloc à contours multiples, on a une moyenne des pixels situés dans les directions des divers contours.

Dans toutes ces méthodes, un bloc texturé est interpolé de la même manière qu'un bloc homogène (interpolation linéaire ou cubique).

Il existe d'autres méthodes où les blocs texturés et les blocs à contours multiples sont interpolés avec un modèle particulier. En pratique, ces deux types de blocs ne sont pas dissociés : le modèle de détection et celui d'interpolation sont les mêmes dans les deux cas. On parle alors plutôt de *blocs irréguliers* pour les désigner.

Un bloc est dit irrégulier si moins de la moitié des gradients, calculés pour chaque pixel du bloc, indique une direction particulière [131], [132], [143]. Dans ce cas, le modèle d'interpolation le plus simple consiste à réaliser une interpolation directionnelle orthogonalement au gradient local [131], [132]. Dans [143] le modèle d'interpolation est plus complexe. Il s'agit de modéliser le bloc irrégulier avec une surface polynomiale non-orientée. L'interpolation est réalisée en rééchantillonnant cette surface continue.

Bloc à contour isolé

Les blocs à contour isolé sont généralement détectés et traités de la même manière que les blocs à contour simple. C'est notamment le cas dans les méthodes qui exploitent le gradient comme modèle de détection et qui réalisent une interpolation parallèle au contour [9], [15], [17], [52], [131], [132], [142], [143], [144].

Dans [74] et [75], la modélisation du bloc avec deux niveaux de gris séparés par un segment de droite ne permet pas de traiter un contour isolé comme un contour simple. Dans ce cas, le contour isolé est interpolé linéairement.

Les contours isolés disparaissent généralement avec les méthodes d'extrapolation dirigées vers les contours [10], [18], [130]. Pour éviter ce problème, dans [45] et [46], on trouve un modèle de détection des contours isolés. Celui-ci consiste à tester l'extrapolation de chaque côté du contour. Si dans les deux directions, l'extrapolation génère une erreur importante, alors on suppose la présence d'un contour isolé. Dans ce cas, le modèle d'interpolation utilisé est l'interpolation au plus proche voisin (cf. section 2.3.2)

3.2.3 Améliorations globales de l'interpolation adaptative

Le bruit dans l'image peut perturber sérieusement l'interpolation adaptative. Un bloc peut ne pas être classé dans la bonne catégorie et l'estimation des paramètres, comme par exemple l'orientation du bloc, peut être faussée. Pour corriger ce problème, Wang *et al.* suggèrent de réaliser une diffusion anisotropique préalablement

à l'étape d'interpolation [52]. La diffusion anisotrope permet de filtrer le bruit dans l'image tout en améliorant la netteté de ses contours (cf. section 2.4.2).

Pour améliorer globalement la netteté et les contrastes dans l'image agrandie par interpolation adaptative, Thurnhofer *et al.* proposent d'ajouter une image de hautes fréquences estimées à partir de celles de l'image originale [132], [131].

Le modèle comporte trois phases. La première consiste à extraire les hautes fréquences de l'image originale avec un filtre *Volterra* quadratique. Ce filtre rentre dans la catégorie des filtres passe-haut et permet d'écarter une bonne partie du bruit que l'on trouve généralement dans les hautes fréquences. La deuxième phase consiste à interpoler l'image des hautes fréquences avec une méthode d'interpolation adaptative. Enfin dans la dernière phase, on réalise un amincissement des contours dans l'image des hautes fréquences interpolée. L'amincissement est basé sur l'amplitude du gradient : en dessous d'un certain seuil de l'amplitude, le niveau de gris du pixel est divisé par la racine carrée de l'amplitude.

3.3 L'extrapolation fréquentielle

Dans le domaine fréquentiel, un contour se caractérise par une multitude de fréquences corrélées [70](p.154). On peut utiliser cette corrélation pour synthétiser les fréquences supérieures de l'image originale et ainsi préserver la netteté des contours dans l'image agrandie. En pratique, la détection et la synthèse des fréquences associées aux contours est un problème extrêmement difficile à maîtriser dans l'espace de Fourier, dès lors que l'image est complexe. Ceci provient du fait qu'une information localisée spatialement se répartit sur l'ensemble du spectre fréquentiel. On a donc un mélange de l'information fréquentielle associée aux contours, aux textures, au bruit, etc., qui est particulièrement difficile à démêler. Pour surmonter cette difficulté, il faut utiliser une représentation de l'image qui permet une localisation aussi bien spatiale que fréquentielle. La transformée en ondelettes permet justement d'obtenir une représentation espace-échelle de l'image (le changement d'échelle entraîne un changement de la fréquence).

Dans la suite, nous décrivons une approche qui réalise la synthèse des coefficients d'ondelettes à une échelle supérieure à celle de l'image originale [29], [33]. Elle est basée sur la détection des singularités à travers les échelles et sur l'extrapolation de ces singularités à une échelle supérieure. Nous commençons par donner quelques notions élémentaires sur la transformée en ondelettes [70].

3.3.1 La transformée en ondelettes

La transformée en ondelettes permet d'obtenir une représentation multi-échelles d'un signal. Comme la transformée de Fourier, la transformée en ondelettes décompose le signal sur une base de fonctions orthogonales. La différence est que les sinusoides sont ici remplacés par des fonctions qui sont localisées aussi bien spatialement que fréquentiellement. Ces fonctions, appelées *ondelettes*, sont générées par

dilatations et translations d'une ondelette mère ψ :

$$(a, b \in \mathbb{R}, a \neq 0) \quad \psi_{a,b}(x) = \frac{1}{\sqrt{|a|}} \psi\left(\frac{x-b}{a}\right) \quad (3.1)$$

où a représente un facteur d'échelle et b un facteur de translation. Dans l'expression (3.1), la multiplication par $|a|^{-\frac{1}{2}}$ permet de maintenir l'aire constante. L'ondelette mère doit être de carré intégrable et doit satisfaire la condition d'admissibilité suivante :

$$\int_{-\infty}^{\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty \quad (3.2)$$

où $\hat{\psi}$ exprime la transformée de Fourier de ψ . Dans le cas discret où $a = 2^m$ et $b = n2^m$ avec $m, n \in \mathbb{Z}$, il est montré qu'il existe des fonctions ψ telles que l'ensemble $\{\psi_{m,n}\}_{m,n \in \mathbb{Z}}$ constitue une base orthogonale de fonctions de carré intégrable [95]. La famille d'ondelettes générée par l'ondelette mère est dans ce cas donnée par :

$$(m, n \in \mathbb{Z}) \quad \psi_{m,n}(x) = 2^{-\frac{m}{2}} \psi(2^{-m}x - n) \quad (3.3)$$

La transformée en ondelettes continue d'une fonction $f \in L^2(\mathbb{R})$ est alors définie par :

$$(\mathcal{W}_\psi f)(m, n) = \langle \psi_{m,n}, f \rangle = \int_{-\infty}^{\infty} \psi_{m,n}(x) f(x) dx \quad (3.4)$$

Et la transformée inverse, qui permet de reconstruire la fonction f initiale, est définie par :

$$f(x) = \sum_{m,n} (\mathcal{W}_\psi f)(m, n) \psi_{m,n}(x) \quad (3.5)$$

La décomposition en ondelettes de la fonction f va dépendre du choix de l'ondelette mère et de l'aptitude de cette dernière à analyser le signal. La figure 3.3 illustre deux exemples d'ondelettes mères. La première est l'ondelette la plus simple que l'on puisse utiliser : l'*ondelette de Haar*. La seconde est le «chapeau mexicain» défini par $\psi(x) = (1 - 2x^2)e^{-x^2}$.

Comme pour la transformée de Fourier, il existe une version rapide de la transformée en ondelettes. Celle-ci a été proposée par Mallat en 1989 à partir des concepts de la multi-résolution [91]. L'algorithme de Mallat fait intervenir une fonction supplémentaire, la *fonction d'échelle* ϕ , dont le rôle est de lisser le signal. Le principe de l'algorithme est d'utiliser la fonction d'échelle pour diminuer progressivement la résolution du signal, tout en utilisant les ondelettes pour relever les détails supprimés par la fonction d'échelle à chaque itération. En pratique, cet algorithme revient à utiliser un filtre passe-bas lié à la fonction d'échelle et un filtre passe-haut lié aux ondelettes.

3.3.2 La régularité et les ondelettes

Lorsque l'on décompose un signal avec une transformée en ondelettes, on constate des similitudes entre les coefficients d'ondelettes à travers les échelles et notamment

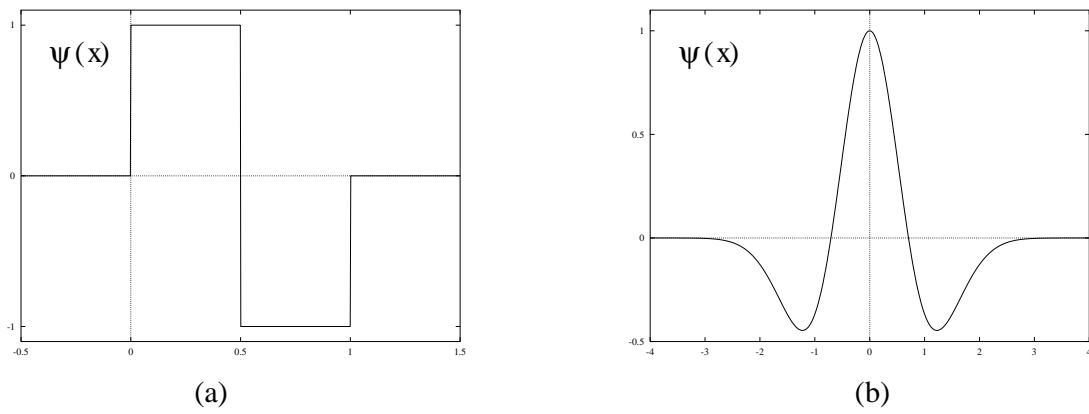
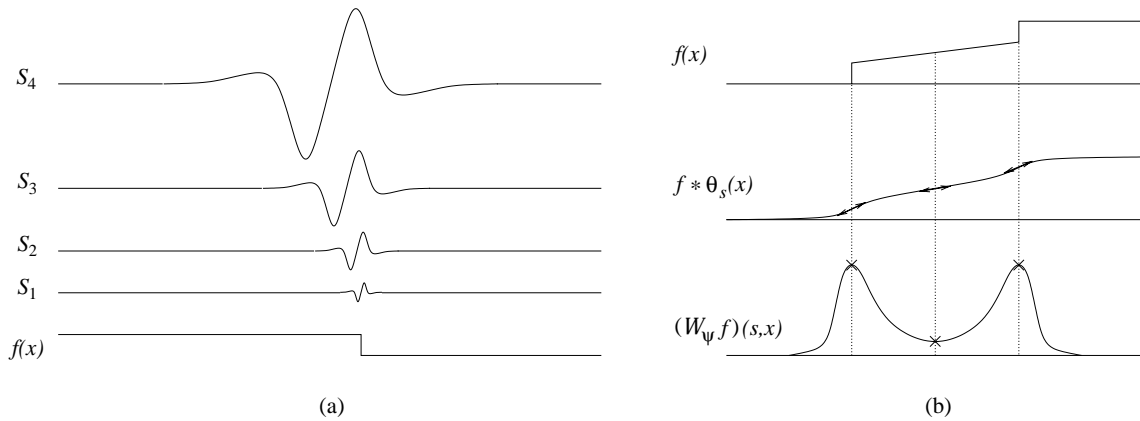


Figure 3.3 – Exemples d'ondelettes mères: (a) ondelette de Haar, (b) le «chapeau mexicain».

Figure 3.4 – (a) Décomposition en ondelettes d'une discontinuité. (b) Détection des singularités avec les maximum locaux de $(\mathcal{W}_\psi f)(s, x)$.

au niveau des singularités (contours), comme le montre la figure 3.4(a). Ces dernières peuvent être caractérisées à partir de la notion de régularité qui indique si une fonction est plus ou moins lisse. Nous allons voir, grâce au théorème 3.1 donné ci-après, que la régularité locale d'une fonction peut être estimée à partir de sa transformée en ondelettes. Ce théorème nous permettra dans la section 3.3.3 de synthétiser, à une échelle supérieure, les coefficients d'ondelettes associés aux singularités. Tout d'abord, nous commençons par donner une définition de la régularité locale d'une fonction.

Définition 3.1 (Régularité lipschitzienne [92]) Soit $0 \leq \alpha \leq 1$. Une fonction $f : \mathbb{R} \rightarrow \mathbb{R}$ est localement α -lipschitzienne en x_0 s'il existe deux constantes K et $h > 0$ telles que $\forall x |x - x_0| \leq h$:

$$|f(x) - f(x_0)| \leq K|x - x_0|^\alpha \quad (3.6)$$

La régularité lipschitzienne est adaptée pour détecter les singularités. Si la fonction f est dérivable en x_0 , alors elle possède une régularité $\alpha \geq 1$. Au contraire, si f est

discontinue, mais limitée dans le voisinage de x_0 , alors sa régularité est $\alpha = 0$. Le théorème qui suit montre que l'exposant α dans (3.6) peut-être estimé à partir de l'évolution des coefficients d'ondelettes à travers les échelles.

Théorème 3.1 (Régularité et transformée en ondelettes [92]) *Soit $0 < \alpha < 1$. Une fonction f est α -lipschitzienne au voisinage de x_0 , le voisinage étant défini par une constante $h > 0$, s'il existe une constante $K > 0$ telle que $\forall x |x - x_0| \leq h$ et $\forall s > 0$:*

$$|(\mathcal{W}_\psi f)(s, x)| \leq K s^\alpha \quad (3.7)$$

Pour une singularité, l'évolution des coefficients à travers les échelles se traduit par une quasi-égalité dans (3.7). En fait, comme l'illustre la figure 3.4(a), on trouve des motifs assez similaires aux différentes échelles. En estimant K et α à partir des échelles inférieures, on peut donc facilement synthétiser les coefficients d'ondelettes liés à une singularité aux échelles supérieures. Pour cela, il faut tout d'abord détecter les singularités.

3.3.3 La détection et la synthèse des singularités

Soit θ une fonction régularisante, comme par exemple une fonction gaussienne, et soit ψ l'ondelette définie par $\psi(x) = \frac{d\theta(x)}{dx}$. La transformée en ondelettes par rapport à ψ est donnée par [92] :

$$(\mathcal{W}_\psi f)(s, x) = f * \psi_s(x) = s \frac{d}{dx}(f * \theta_s)(x) \quad (3.8)$$

où la fonction θ_s est définie par $\theta_s = (1/s)\theta(x/s)$. A partir de (3.8), on observe que la transformée en ondelettes $(\mathcal{W}_\psi f)(s, x)$ est proportionnelle à la première dérivée de la régularisée de f par θ_s . Ceci signifie que les singularités de f correspondent aux maxima locaux de $(\mathcal{W}_\psi f)(s, x)$. En considérant uniquement le maximum du module de $(\mathcal{W}_\psi f)$, on peut donc détecter facilement les singularités de f , comme le montre la figure 3.4(b).

En pratique, seules sont retenues les singularités pour lesquelles est obtenue une quasi-égalité dans (3.7) aux différentes échelles. On supprime ainsi les contours qui appartiennent aux zones texturées, pour finalement ne garder que les contours exprimant des frontières entre objets. Pour les singularités retenues, il ne reste plus qu'à estimer la régularité α et le nombre K à partir des maxima d'ondelettes aux différentes échelles. Ainsi, supposons qu'une singularité soit détectée au voisinage de x_0 . Dans le cas discret où $s = 2^i$ et en notant simplement a_i le coefficient en x_0 de la bande S_i (échelle 2^i), on obtient à partir de (3.7) :

$$|a_i| \leq K(2^i)^\alpha \quad (3.9)$$

$$\log_2 |a_i| \leq \log_2 K + i\alpha \quad (3.10)$$

On suppose que l'on dispose de $L - 1$ bandes de coefficients d'ondelettes, notées S_2, \dots, S_L (L représente le niveau maximum d'analyse). On souhaite extrapoler la

bande S_1 en x_0 , point auquel on a détecté une singularité forte. Une estimation de α et K en x_0 est obtenue en résolvant le problème aux moindres carrés suivant :

$$(K, \alpha) = \arg \min_{(c, \beta)} \sum_{i=2}^L (\log_2 |a_i| - \log_2 c - \beta) \quad (3.11)$$

Le coefficient a_1 est alors calculé en appliquant la relation :

$$|a_1| = K2^\alpha \quad (3.12)$$

Pour estimer les autres coefficients de la bande S_1 qui ne correspondent pas à des points de singularité, on trouve différentes techniques. Dans [29], les coefficients de la bande S_3 sont utilisés comme valeurs initiales : la bande S_2 est écartée car elle contient le plus souvent du bruit. Ces coefficients sont multipliés par un facteur 1/4 pour prendre en compte le changement d'échelle. L'amplitude locale est réajustée en fonction des coefficients estimés précédemment à partir des singularités. Les coefficients manquant sont finalement interpolés avec une spline cubique. Dans [33], on trouve une minimisation fonctionnelle qui exploite les coefficients initiaux des singularités. La monotonie est utilisée comme contrainte sur les coefficients.

Notes bibliographiques

L'approche présentée précédemment sur l'extrapolation d'une bande d'ondelettes, prend en compte la décomposition aux différentes échelles par l'intermédiaire des maximum d'ondelettes et notamment des singularités [29], [33]. Il est noté dans [8], que cette approche préserve la netteté des contours mais produit par contre de nombreux artefacts comme des effets d'auréoles. Ceci peut provenir d'un déphasage ou d'une amplitude trop importante pour certains coefficients prédits.

Dans la littérature, on trouve d'autres approches plus simples qui extrapolent à partir d'une ou deux bandes d'ondelettes. C'est par exemple le cas dans [128], où deux filtres d'interpolation sont utilisés pour synthétiser la bande supérieure. Les poids de ces filtres sont obtenus en optimisant l'erreur de prédiction en utilisant deux bandes d'ondelettes calculées à partir de l'image originale. Le premier filtre est optimisé sur les zones de contours et le deuxième sur les zones de faibles variations de l'intensité. Dans [59], l'extrapolation est réalisée à partir d'une pyramide laplacienne [22] qui peut être vue comme un cas particulier de décomposition en ondelettes non-orthogonales. En pratique, seule la base de la pyramide laplacienne est utilisée. En l'interpolant, puis en lui appliquant un opérateur non-linéaire qui va supprimer les basses fréquences, on génère un niveau supplémentaire de la pyramide laplacienne. Celui-ci est ajouté à l'image originale interpolée pour créer finalement l'image agrandie. On retrouve une idée assez similaire dans [14], où une image de différences (hautes fréquences) est ajoutée à l'image agrandie par interpolation à l'ordre 0. L'image de différences est ici obtenue par une procédure d'apprentissage avec un réseau neuronal sur des blocs d'images. Dans [6], on retrouve également une procédure d'apprentissage par réseau neuronal. Elle est ici réalisée à partir d'une transformée en cosinus discrète ou d'une transformée de Walsh discrète et consiste à prédire les hautes fréquences à partir des basses fréquences.

3.4 Le zoom fractal

La méthode que nous présentons dans cette section est très différente des deux méthodes précédentes puisqu'elle n'intègre pas explicitement les notions classiques de contours ou de régions. Elle est basée sur la recherche des similarités structurelles de l'image et sur l'hypothèse de leur invariance par changement de résolution. Ce type d'approche est directement lié à la compression par fractale [47], [50], dont l'idée fondamentale est de représenter une image uniquement avec des relations d'auto-similarités locales (voir figure 3.5). L'ensemble de ces relations constitue ce que l'on appelle un *code fractal*. En itérant ce code sur une image quelconque, on peut montrer qu'il y a convergence vers un attracteur proche de l'image originale. Le code fractal d'une image est donc une représentation approximative de cette image. Du point de vue du stockage, il requiert beaucoup moins de place que l'image originale, et à ce titre il permet de compresser efficacement (avec pertes) une image. L'utilisation du code fractal pour l'agrandissement d'une image est récente et s'appuie sur son indépendance vis-à-vis de la résolution de l'image originale. Dans la littérature, cette méthode est souvent désignée par le terme *zoom fractal* [55], [62], [96], [106]. C'est actuellement l'une des méthodes qui fournit les résultats visuels les plus probants. Elle permet à la fois de respecter la nature discontinue des images naturelles, et d'introduire des détails supplémentaires grâce au principe d'auto-similarité. Nous verrons néanmoins qu'elle n'est pas exempte de défauts, le principal étant d'introduire souvent des fausses discontinuités.



Figure 3.5 – Exemples d'auto-similarités locales dans l'image de Lenna.

Nous commençons par étudier le modèle standard de transformation fractale d'une image, puis nous décrivons la procédure d'agrandissement d'une image à partir de son code fractal.

3.4.1 Transformation fractale d'une image

Etant donnée une image \mathcal{A} de résolution $M^2 = 2^m \times 2^m$ pixels, on souhaite trouver un opérateur W tel que $W(\mathcal{A}) = \mathcal{A}$ et tel que ce point fixe ou *attracteur* soit unique. Un tel opérateur représente un code exact de l'image \mathcal{A} , puisque la seule connaissance de W permet de retrouver exactement \mathcal{A} . La transformation de l'image \mathcal{A} par l'opérateur W est qualifiée de *transformation fractale*.

Le problème inverse de recherche de W est particulièrement difficile. En pratique, on se contente d'une certaine classe d'opérateurs simples et on réduit W à un système de transformations élémentaires $(\omega_i)_{i \in \mathcal{I}}$ opérant chacune sur une sous-partie de l'image \mathcal{A} . Cette deuxième restriction permet de limiter l'approximation $W(\mathcal{A}) \approx \mathcal{A}$ due à la première restriction. Plus précisément, à chaque transformation ω_i est associé un bloc source $d_{\varphi(i)} \subset \mathcal{A}$ et un bloc destination $r_i \subset \mathcal{A}$ de manière à ce que l'ensemble $(r_i)_{i \in \mathcal{I}}$ forme une partition de \mathcal{A} . L'ensemble $(d_{\varphi(i)})_{i \in \mathcal{I}}$ est un sous-ensemble de $(d_j)_{j \in \mathcal{J}}$ composé lui-même de blocs de \mathcal{A} *a priori* quelconques. La fonction $\varphi : \mathcal{I} \rightarrow \mathcal{J}$ associe un bloc source $d_{\varphi(i)}$ à un bloc destination r_i de telle sorte que $\omega_i(d_{\varphi(i)}) \approx r_i$. Ainsi, le problème de recherche d'un opérateur W se ramène à un problème de recherche d'une fonction φ et d'un système de transformations élémentaires $(\omega_i)_{i \in \mathcal{I}}$ tels que :

$$W(\mathcal{A}) = \bigcup_{i \in \mathcal{I}} \omega_i(d_{\varphi(i)}) \approx \bigcup_{i \in \mathcal{I}} r_i = \mathcal{A} \quad (3.13)$$

Dans le modèle standard, l'ensemble $(r_i)_{i \in \mathcal{I}}$ forme une partition régulière à géométrie carrée avec des blocs de taille égale à $N^2 = 2^n \times 2^n$ pixels, et la collection $(d_j)_{j \in \mathcal{J}}$ est composée de blocs de taille $2^{n+1} \times 2^{n+1}$ pixels. Cette situation est décrite par la figure 3.6(a).

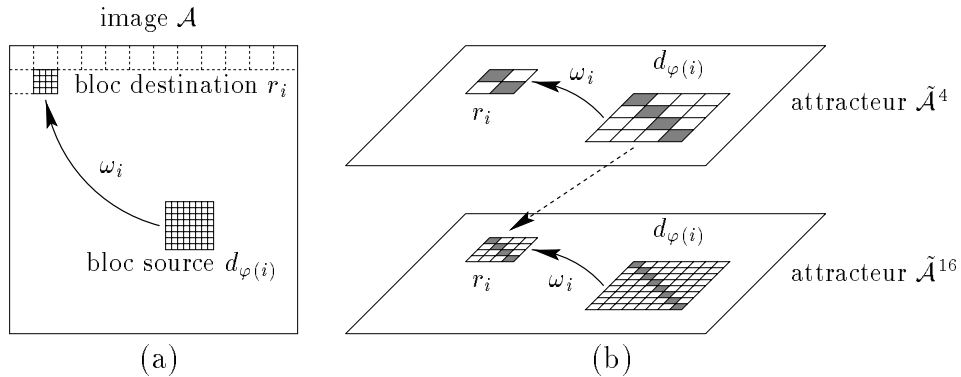


Figure 3.6 – (a) Transformation élémentaire $\omega_{i \in \mathcal{I}}$. (b) Principe du zoom fractal.

Généralement, on restreint l'ensemble $(\omega_i)_{i \in \mathcal{I}}$ à un ensemble de transformations affines de la forme :

$$\begin{aligned} \omega_i &: \mathbb{R}^{2N \times 2N} \rightarrow \mathbb{R}^{N \times N} \\ d_{\varphi(i)} &\mapsto s_i \psi(d_{\varphi(i)}) + o_i[\mathbf{1}]_{N \times N} \end{aligned} \quad (3.14)$$

où $[\mathbf{1}]_{N \times N}$ est une matrice de taille $N \times N$ composée uniquement de 1. La fonction ψ est une fonction de contraction spatiale ramenant la taille d'un bloc source à celle d'un bloc destination. Classiquement, elle réalise une convolution du bloc source avec un filtre opérant une moyenne, puis décime le bloc résultant d'un facteur 2, c'est-à-dire $(\forall (x, y) \in [1, \dots, N]^2)$:

$$\psi(d_{\varphi(i)})(x, y) = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{bmatrix} * \begin{bmatrix} d_{\varphi(i)}(2x-1, 2y-1) & d_{\varphi(i)}(2x-1, 2y) \\ d_{\varphi(i)}(2x, 2y-1) & d_{\varphi(i)}(2x, 2y) \end{bmatrix} \quad (3.15)$$

Les coefficients s_i et o_i sont respectivement appelés *coefficient d'échelle* et *coefficient de décalage*. Ils agissent sur la luminance des pixels du bloc réduit $\psi(d_{\varphi(i)})$. Résoudre le problème (3.13) avec une métrique induite par une norme ℓ^2 revient finalement à trouver pour chaque bloc destination $r_i \in \mathcal{I}$ un triplet $(s_i, o_i, \varphi(i)) \in \mathbb{R}^2 \times \mathcal{J}$ tel que :

$$(s_i, o_i, \varphi(i)) = \arg \min_{(a, b, j)} \| a\psi(d_j) + b[\mathbf{1}]_{N \times N} - r_i \|^2 \quad (3.16)$$

Cette fonction étant convexe, on peut résoudre le problème de minimisation en parcourant $(d_j)_{j \in \mathcal{J}}$ et en calculant à chaque fois les paramètres optimaux (a, b) donnés par l'annulation des dérivées partielles correspondantes. Il s'agit alors pour chaque bloc source $d_j \in \mathcal{J}$ de résoudre le système :

$$\begin{bmatrix} \sum_{x,y} \psi(d_j)(x, y)^2 & \sum_{x,y} \psi(d_j)(x, y) \\ \sum_{x,y} \psi(d_j)(x, y) & N^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{x,y} \psi(d_j)(x, y) \cdot r_i(x, y) \\ \sum_{x,y} r_i(x, y) \end{bmatrix} \quad (3.17)$$

Nous obtenons de cette manière un opérateur W complètement spécifié et fournissant un *code fractal* de l'image \mathcal{A} . Du fait de la simplicité du modèle, l'image transformée $W(\mathcal{A})$ n'est qu'une approximation de \mathcal{A} et l'attracteur obtenu (s'il existe) est dans ce cas donné par :

$$\lim_{\xi \rightarrow \infty} W^{\circ \xi}(\mathcal{A}) = \tilde{\mathcal{A}} \quad (3.18)$$

Si l'opérateur W est contractant dans l'espace des images, on peut montrer que l'attracteur $\tilde{\mathcal{A}}$ existe et qu'il est unique. Pour cela, une condition suffisante est que les coefficients d'échelle associés aux transformations élémentaires $(\omega_i)_{i \in \mathcal{I}}$ soient tels que $|s_i| < 1$ pour tout $i \in \mathcal{I}$. Ainsi, avec la seule connaissance de W , on peut retrouver approximativement \mathcal{A} en itérant W sur une image quelconque \mathcal{B} de même résolution que \mathcal{A} , c'est-à-dire :

$$(\forall \mathcal{B} \in \Xi^{res(\mathcal{A})}) \lim_{\xi \rightarrow \infty} W^{\circ \xi}(\mathcal{B}) = \tilde{\mathcal{A}} \approx \mathcal{A} \quad (3.19)$$

où $res(\mathcal{A})$ désigne la résolution de l'image \mathcal{A} , c'est-à-dire son nombre de pixels, et Ξ^r désigne l'espace des images de résolution r .

Nous avons présenté dans cette section le modèle standard de transformation fractale d'une image. Des modèles plus évolués ont été proposés dans la littérature

spécialisée. Ils sont notamment basés sur des partitionnements de l'image originale plus complexes que le simple partitionnement carré. On peut citer notamment le partitionnement rigide comme le *quadtree*, semi-rigide comme le *H-V* (*horizontal-vertical*) constitué de rectangles, ou encore le partitionnement adaptatif obtenu à partir d'une triangulation de Delaunay [47].

La transformation fractale d'une image est une opération particulièrement coûteuse en temps de calcul. Pour l'accélérer, il faut optimiser la recherche des similarités dans l'image. De nombreuses stratégies ont été proposées dans ce sens, comme par exemple la classification ou la quantification des blocs sources, la recherche des similarités par exploration locale, ou encore des techniques pour ramener le problème de minimisation (3.16) à un problème de recherche du plus proche voisin [50].

3.4.2 Agrandissement avec un code fractal

En faisant varier la résolution des blocs destination $(r_i)_{i \in \mathcal{I}}$, on fait varier proportionnellement la résolution de l'attracteur de W . On peut de cette manière réaliser un agrandissement de l'image \mathcal{A} à une résolution quelconque. On parle alors d'un *zoom fractal*. Pour signifier la dépendance de la résolution de l'attracteur vis-à-vis de la résolution des blocs destination, nous notons $\tilde{\mathcal{A}}^{res(r_i \in \mathcal{I})}$ cet attracteur. Le principe du zoom fractal se décrit alors par l'équation :

$$(\forall \mathcal{B} \in \Xi^{\frac{M^2}{N^2} \cdot res(r_i \in \mathcal{I})}) \quad \lim_{\xi \rightarrow \infty} W^{\circ \xi}(\mathcal{B}) = \tilde{\mathcal{A}}^{res(r_i \in \mathcal{I})} \quad (3.20)$$

Cette équation traduit finalement une pyramide d'images (attracteurs) où chaque niveau de résolution possède des relations de similarités locales strictement identiques à celles des autres niveaux. En notant $\tilde{\mathcal{A}}_{|r_i}$ la restriction de l'attracteur $\tilde{\mathcal{A}}$ au bloc r_i , ces relations de similarités locales s'expriment par :

$$(\forall i \in \mathcal{I}) \quad \tilde{\mathcal{A}}_{|r_i}^{res(r_i)} = s_i \psi(\tilde{\mathcal{A}}_{|d_{\varphi(i)}}^{res(r_i)}) + o_i[\mathbf{1}]_{res(r_i)} \quad (3.21)$$

En d'autres termes, les relations de similarités inter-blocs extraites de l'image \mathcal{A} par la résolution du problème (3.16), sont invariantes par changement de résolution spatiale. C'est cette hypothèse qui justifie le zoom fractal.

Un théorème important dû à Baharav *et al.*, permet de réaliser le zoom fractal en évitant le principe itératif décrit par (3.20).

Théorème 3.2 (Zoom fractal [13]) *Si la fonction de contraction spatiale ψ est du même type que (3.15), alors les attracteurs $\tilde{\mathcal{A}}^{res(d_{\varphi(i) \in \mathcal{I}})}$ et $\tilde{\mathcal{A}}^{res(r_i \in \mathcal{I})}$ sont liés par la relation suivante :*

$$(\forall i \in \mathcal{I}) \quad \tilde{\mathcal{A}}_{|r_i}^{res(d_{\varphi(i)})} = s_i \tilde{\mathcal{A}}_{|d_{\varphi(i)}}^{res(r_i)} + o_i[\mathbf{1}]_{res(d_{\varphi(i)})} \quad (3.22)$$

Ce théorème signifie qu'un attracteur peut-être directement calculé à partir de l'attracteur de niveau de résolution inférieur. En supprimant la fonction de contraction

spatiale dans l'expression (3.14), on obtient le bloc destination à un niveau de résolution double. Nous avons illustré ceci sur la figure 3.6(b) en exprimant par une flèche en pointillé la relation existant entre les blocs $\tilde{\mathcal{A}}_{r_i}^{16}$ et $\tilde{\mathcal{A}}_{d_{\varphi(i)}}^4$.

On peut donc obtenir plus rapidement le zoom fractal d'une image \mathcal{A} en calculant d'abord l'attracteur initial $\tilde{\mathcal{A}}^{N^2}$, puis en construisant la suite d'attracteurs $\tilde{\mathcal{A}}^{4N^2}, \tilde{\mathcal{A}}^{16N^2}, \tilde{\mathcal{A}}^{64N^2}, \dots$ avec le théorème 3.2, jusqu'à obtenir l'attracteur de résolution désirée. Pour augmenter davantage la rapidité du zoom, il est préférable en pratique d'appliquer le théorème 3.2 directement sur l'image originale \mathcal{A} . On peut montrer que la suite d'images construite à partir de \mathcal{A} converge asymptotiquement vers la suite d'attracteurs obtenue à partir de $\tilde{\mathcal{A}}^{N^2}$. Remarquons que cette technique n'est utilisable que si la fonction de contraction spatiale réalise une moyenne des pixels.

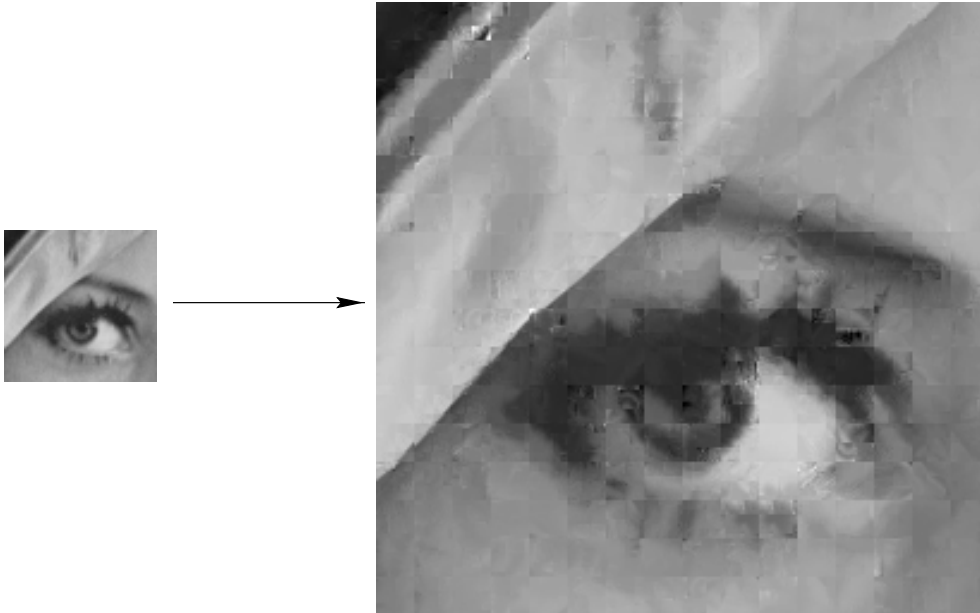


Figure 3.7 – Zoom fractal de facteur 4.

Nous montrons un exemple de zoom fractal de facteur 4 sur la figure 3.7 ($N=4$). Nous observons une nette amélioration de la résolution, mais également l'apparition de discontinuités artificielles situées aux frontières des blocs destination. Pour diminuer cet artefact, Polidori *et al.* ont proposé une technique d'interpolation adaptative opérant sur quatre images agrandies issues de quatre partitions différentes de l'image initiale [106]. Cependant, cette méthode ne supprime pas complètement les effets de blocs et produit un léger effet de flou.

3.5 Autres méthodes

En dehors des trois principales méthodes que nous avons présentées dans les sections précédentes, il en existe d'autres qui sont plus marginales mais qui présentent néanmoins un intérêt certain. Dans cette section, nous présentons quatre de ces méthodes. Les deux premières sont basées sur l'utilisation de filtres d'interpolation particuliers : le filtre médian qui est un filtre non-linéaire et le filtre de Wiener qui est un filtre d'optimisation. Les deux suivantes réalisent l'agrandissement en respectant une contrainte de réduction et en minimisant une fonctionnelle qui préserve les contours. Dans le premier cas, la fonctionnelle est issue d'une modélisation probabiliste avec une distribution de Gibbs sur un champ markovien. Dans le deuxième cas, la fonctionnelle à minimiser est la variation totale liée au gradient de l'image.

3.5.1 Interpolation avec un filtre médian

Le *médian* d'une liste de valeurs correspond à l'élément central de la liste ordonnée. Lorsque le nombre des éléments de la liste est pair, le médian est défini comme la moyenne des deux valeurs associées aux éléments centraux. Un filtre médian est un filtre non-linéaire qui produit comme résultat le médian des valeurs traitées. On peut utiliser ce type de filtre pour interpoler une image et remplacer ainsi le filtre linéaire classique qui lisse les contours, par un filtre non-linéaire qui améliore la netteté.

Zeng *et al.* réalisent l'interpolation en deux étapes [149] : la première consiste à interpoler les pixels situés sur les diagonales et la deuxième consiste à interpoler les pixels restants, situés sur les lignes horizontales et verticales. Les figures 3.8(a) et 3.8(b) illustrent les pixels voisins X_1, X_2, X_3, X_4 pris en compte pour interpoler le pixel Y dans les deux étapes respectives. Zeng *et al.* proposent deux filtres pour la première étape et trois autres pour la deuxième, ce qui par combinaison produit en tout six filtres. Le filtre qui semble donner les meilleurs résultats est le suivant [149] :

$$\begin{cases} \text{Etape 1: } Y = \text{median}(X_1, X_2, X_3, X_4, X_{FIR1}) \\ \text{Etape 2: } Y = \text{median}(X_1, X_2, X_3, X_4, \bar{X}_H, \bar{X}_V, X_{FIR2}) \end{cases} \quad (3.23)$$

où $\bar{X}_H = (X_1 + X_2)/2$ représente la moyenne horizontale et $\bar{X}_V = (X_3 + X_4)/2$ la moyenne verticale. Les valeurs X_{FIR1} et X_{FIR2} sont issues de filtres FIR (*finite impulse response*) permettant respectivement l'interpolation linéaire d'un maillage carré et d'un maillage en quinconce. Les poids de ces deux filtres sont montrés sur les figures 3.8(c) et 3.8(d).

La figure 3.9(a) donne un exemple d'interpolation avec le filtre médian défini par l'expression (3.23). L'image utilisée est une portion de l'image de San-Francisco. On peut comparer ce résultat à celui obtenu par la méthode de la transformée B-spline cubique qui est montré sur la figure 2.12. L'image obtenue avec le filtre médian possède des contours plus nets que celle obtenue avec la transformée B-spline. Ceci est surtout vrai pour les contours proches d'une inclinaison diagonale. Pour les contours horizontaux et verticaux, l'effet de flou est toujours présent puisque dans

L'interpolation avec le filtre de Wiener donne un résultat correct lorsque l'image à interpoler a des caractéristiques très simples. Ce type de filtre produit des résultats sensiblement meilleurs que le filtre médian [149]. Cependant, en raison de la phase d'apprentissage, il nécessite beaucoup plus de calculs que ce dernier.

3.5.3 Interpolation avec un modèle probabiliste

Schultz *et al.* ont posé le problème de l'interpolation d'un point de vue probabiliste en modélisant l'image agrandie avec un champ markovien [123], [124]. Tout d'abord, l'image originale et l'image agrandie sont liées par une contrainte de réduction simple : la valeur d'un pixel de l'image originale est égale à la moyenne des quatre valeurs associées aux quatre pixels correspondants dans l'image agrandie (pour un agrandissement de facteur 2 dans chaque direction). L'ensemble des images agrandies qui respectent cette contrainte est défini par :

$$\mathcal{Z} = \{\vec{z} \mid \vec{y} = D\vec{z}\} \quad (3.24)$$

où \vec{y} est le vecteur associé à l'image originale, \vec{z} est celui associé à l'image agrandie et D est une matrice qui traduit les contraintes de dépendance linéaire (moyennes).

L'ensemble \mathcal{Z} n'est jamais vide, il comporte au moins une solution : l'image agrandie par interpolation au plus proche voisin de \vec{y} (cf. section 2.3.2). Pour obtenir une solution particulière dans \mathcal{Z} qui respecte des contraintes de régularité, Schultz *et al.* modélisent l'image agrandie par un champ aléatoire de Markov avec une distribution de Gibbs. La maximisation de la probabilité *a priori* (MAP) est utilisée pour calculer une estimation \hat{z} de l'image agrandie. Dans ce cas, on aboutit à un problème de minimisation d'une fonctionnelle d'énergie U et l'image estimée \hat{z} est donnée par :

$$\hat{z} = \arg \min_{\vec{z} \in \mathcal{Z}} \left\{ U(\vec{z}) = \sum_{c \in \mathcal{C}} V_c(\vec{z}) \right\} \quad (3.25)$$

où la fonction V_c est une fonction potentielle associée à une clique⁴ c . Les cliques sont définies sur un voisinage de 3×3 pixels et correspondent aux quatre directions possibles. On a donc quatre types de cliques prenant chacune en compte trois pixels alignés dans une direction. Schultz *et al.* définissent la fonction potentielle par l'expression suivante :

$$\sum_{c \in \mathcal{C}} V_c(\vec{z}) = \sum_{k,l} \left(\sum_{i=1}^4 \rho(D_{c_i}^2 z_{k,l}) \right) \quad (3.26)$$

où $z_{k,l}$ représente le niveau de gris de l'image z au pixel de coordonnées (k, l) et $D_{c_i}^2 z_{k,l}$ représente la dérivée seconde dans la direction associée à la clique c_i . L'utilisation de la dérivée seconde permet de générer une image solution de \mathcal{Z} relativement lisse. Dans l'expression (3.26), la fonction ρ est une fonction de Huber qui permet de préserver les discontinuités de l'image originale dans l'image agrandie. Elle a également pour intérêt de rendre convexe la fonctionnelle U . Cette propriété permet

4. Une clique est un sous-graphe complet défini sur un certain voisinage.

d'utiliser des méthodes rapides pour trouver le minimum global de U . En pratique, Schultz *et al.* utilisent un algorithme de descente en gradient et prennent comme configuration initiale de \hat{z} l'image agrandie par interpolation au plus proche voisin de \vec{y} . Cette image appartient effectivement à l'ensemble \mathcal{Z} défini par (3.24), mais ne minimise pas la fonction d'énergie U .

Nous montrons sur la figure 3.10 l'exemple⁵ donné par Schultz *et al.* dans [124].

Dans cet exemple, l'image originale est réduite d'un facteur 4 et le problème est d'agrandir d'un facteur 4 l'image réduite. Pour cela, l'image interpolée au plus proche voisin de l'image réduite est utilisée comme image agrandie initiale dans l'algorithme de descente en gradient. On peut remarquer que le résultat final de cette descente en gradient, montré sur la figure 3.10(d), récupère un peu d'information sur l'image originale. Malgré tout, l'image agrandie par la méthode de Schultz *et al.* demeure relativement floue.

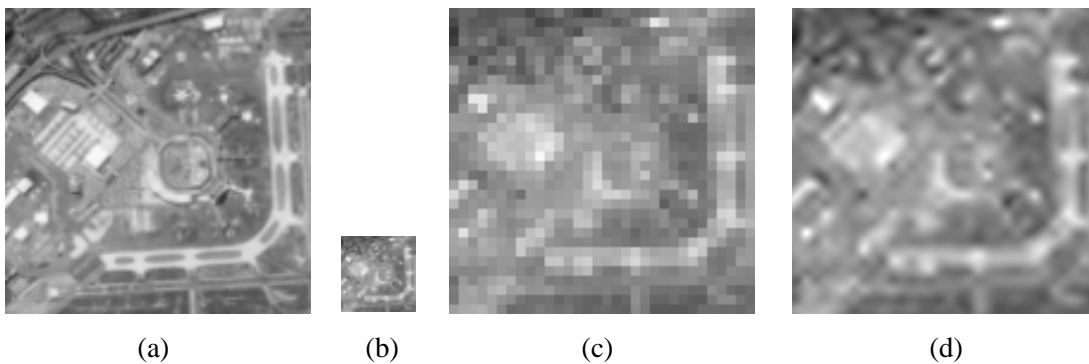


Figure 3.10 – (a) Image originale. (b) Image réduite d'un facteur 4 avec un filtre spatial réalisant une moyenne. (c) Image initiale utilisée par la descente en gradient : elle est obtenue par interpolation au plus proche voisin de l'image réduite. (d) Résultat de la descente en gradient.

Dans [67] et [99], l'interpolation est également réalisée en modélisant l'image agrandie par un champ de Markov. La différence est que l'image originale et l'image agrandie ne sont pas liées par une contrainte de réduction. Cependant, des paramètres de textures sont évalués sur différentes régions de l'image originale grâce à un algorithme de descente en gradient. Ces paramètres sont ensuite utilisés pour déterminer la distribution de Gibbs du champ markovien associé à l'image agrandie. La minimisation de l'énergie sur ce champ est finalement réalisée avec une opération originale de filtrage non-linéaire.

3.5.4 Interpolation basée sur la variation totale

La méthode de minimisation de la variation totale a été proposée initialement par Rudin *et al.* pour débruiter une image tout en préservant les contours [112]. Celle-ci a ensuite été adaptée par Guichard *et al.* pour interpoler une image [64].

⁵. Les images montrées dans cet exemple ont été trouvées sur internet à l'adresse suivante <http://lisa.ee.nd.edu/rls/papers/J9/>.

Comme dans la méthode probabiliste de Schultz *et al.*, que nous avons présentée précédemment, l'image agrandie est soumise à une contrainte de réduction. Celle-ci est cependant plus générale puisque le simple filtre réalisant une moyenne est ici remplacé par un filtre normalisé de taille et de poids quelconques. De plus, une opération de sous-échantillonnage est réalisée après le filtrage. L'ensemble des images agrandies qui respectent la contrainte de réduction est donc donné par :

$$\mathcal{W} = \{w \mid u = QS(w)\} \quad (3.27)$$

où u est l'image originale, w est une image agrandie, Q symbolise l'opérateur de sous-échantillonnage et S représente le filtre spatial. Si u et w ont une taille respective N et $2N$, alors la contrainte de réduction s'écrit dans le domaine fréquentiel :

$$(\forall i \in] - N/2, N/2]) \quad 2\tilde{u}_i = \tilde{s}_i\tilde{w}_i + \tilde{s}_{i+N}\tilde{w}_{i+N} \quad (3.28)$$

où \tilde{u} , \tilde{w} et \tilde{s} représentent respectivement les transformées de Fourier de u , w et s . Le facteur 2 dans le membre de gauche permet de préserver la dynamique du signal. A travers la relation (3.28), on perçoit les degrés de liberté dont on dispose pour interpoler une image de manière à ce que le résultat appartienne à l'ensemble \mathcal{W} . Pour chaque fréquence \tilde{u}_i de l'image u , on peut répartir l'information entre deux fréquences \tilde{w}_i et \tilde{w}_{i+N} de l'image agrandie w . Lorsque toute l'information est placée dans les coefficients basse-fréquence \tilde{w}_i , $\forall i \in] - N/2, N/2]$, l'image w obtenue est équivalente à celle obtenue avec la méthode du *zero-padding* que nous avons étudiée dans la section 2.3.1. On remarque également, avec la relation (3.28), qu'une condition suffisante pour que l'ensemble \mathcal{W} ne soit pas vide est que la réponse fréquentielle du filtre s soit non-nulle sur l'intervalle $] - N/2, N/2]$.

Pour avoir une «bonne solution» appartenant à l'ensemble \mathcal{W} , Guichard *et al.* proposent de choisir l'image agrandie de \mathcal{W} qui minimise une mesure de régularité basée sur la norme de la variation totale du gradient spatial :

$$E(w) = \int |\nabla w| dx \approx \sum_{i=0}^{2N-1} |w_{i+1} - w_i| \quad (3.29)$$

Cette mesure de régularité a l'avantage de préserver les contours de l'image originale.

Finalement, Guichard *et al.* obtiennent une solution de \mathcal{W} qui minimise la norme de la variation totale en utilisant un algorithme de descente en gradient. L'image initiale, à partir de laquelle est réalisée la descente en gradient, est l'image interpolée par la méthode du *zero-padding*. Cette image appartient à l'ensemble \mathcal{W} , mais ne minimise pas en général la mesure de régularité E . Nous avons déjà vu dans la section 2.3.1 que la méthode du *zero-padding* produit des ondulations dans l'image interpolée. La minimisation de la variation totale va justement supprimer ces ondulations tout en préservant les contours.

Nous donnons un exemple⁶ d'agrandissement de facteur 4 sur la figure 3.11. L'image originale est une portion de l'image de Lenna (256×256 pixels). La figure 3.11(c) montre le résultat de l'algorithme de descente en gradient effectuée à

6. Les images de cet exemple sont celles qui sont montrées dans l'article de Guichard *et al.* [64]. Elles ont été récupérées directement sur le site WEB de l'auteur à l'adresse <http://www.ceremade.dauphine.fr/fguichar/images/zoom1.gif>.



Figure 3.11 – (a) Interpolation au plus proche voisin. (b) Interpolation avec la méthode du *zero-padding*. (c) Interpolation par minimisation de la variation totale.

partir de l'image de la figure 3.11(b) obtenue par la méthode du *zero-padding*. On remarque que la minimisation de la variation totale supprime effectivement l'effet d'ondulation et que les contours sont un peu plus nets. Cependant, la netteté demeure partielle et les contours sont quand même lissés. De plus, on peut remarquer un effet de peinture qui se traduit par des plateaux d'intensité constante.

3.6 Mesure de la qualité de l'agrandissement

L'évaluation de la qualité d'un agrandissement est une tâche difficile. En principe, on ne dispose pas d'une image de référence à partir de laquelle on pourrait effectuer des mesures de comparaison. De plus, la création de détails peut parfois entraîner des incohérences au niveau des structures (on peut par exemple penser à l'interpolation avec un filtre médian, cf. section 3.5.1) et ce type d'artefact est difficile à déceler uniquement par des mesures mathématiques. Pour cette raison, l'estimation visuelle demeure le meilleur moyen d'évaluer la qualité d'un agrandissement. Cette estimation peut néanmoins être aidée par un examen du spectre fréquentiel de l'image agrandie [8], [59], [64], [131], [132]. De cette manière, on peut facilement savoir si la méthode produit des hautes fréquences et si celles-ci sont corrélées avec les basses fréquences, notamment au niveau des contours. Une bonne méthode d'agrandissement devrait normalement prolonger le spectre dans les hautes fréquences. Par exemple, on peut voir sur la figure 3.12(a) que la méthode d'interpolation par minimisation de la variation totale prolonge un peu le spectre initial (bloc central). La trace diagonale qui est prolongée correspond au bord du chapeau de Lenna et traduit une discontinuité. La création de hautes fréquences dans cet exemple est néanmoins partielle, puisque l'énergie fréquentielle est surtout concentrée au niveau du spectre initial.

Cependant, cette aide est souvent insuffisante dans le sens où l'interprétation visuelle du spectre n'est pas toujours facile. Par exemple, lorsque l'image possède des structures avec des orientations multiples, on ne distingue plus vraiment de

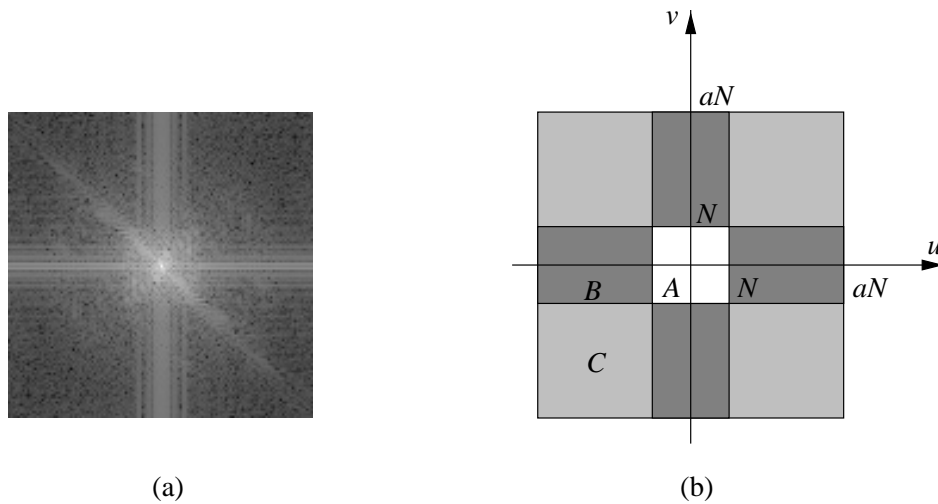


Figure 3.12 – (a) Spectre de l'image agrandie par minimisation de la variation totale qui est illustrée sur la figure 3.11(c). (b) Les trois zones A , B , C à distinguer sur le spectre d'une image agrandie par facteur a .

traces spectrales : l'information fréquentielle est mélangée. Dans ce cas, les hautes fréquences peuvent être l'expression d'un bruit spatial et pas nécessairement celui d'une amélioration de la résolution.

Pour faciliter l'analyse spectrale de l'image agrandie, Xue *et al.* ont proposé trois mesures mathématiques effectuées sur le spectre de l'image agrandie. Ces trois mesures sont associées à trois zones distinctes du spectre de l'image agrandie. Elles sont représentées sur la figure 3.12(b). La zone A correspond à la gamme de fréquences associée au spectre de l'image originale. La zone B est une zone de hautes fréquences horizontales et verticales qui contient le plus souvent de l'énergie non significative. En fait, il s'agit d'un artefact de la transformée de Fourier. Cette dernière périodise implicitement l'image dans le domaine spatial. Cette périodisation entraîne la création de discontinuités artificielles verticales et horizontales sur les bords de l'image (cf. section 2.2.3). La zone C correspond aux hautes fréquences restantes de l'image agrandie. La première mesure consiste à calculer la différence entre le spectre de l'image originale et la zone spectrale A de l'image agrandie. Pour une bonne méthode d'agrandissement, la distorsion doit être minimale. Les deux autres mesures évaluent la quantité d'énergie présente dans les zones B et C . Lorsque la méthode d'agrandissement crée des hautes fréquences, l'énergie fréquentielle dans ces zones augmente. Ces deux dernières mesures ne sont malheureusement pas toujours significatives. Par exemple, l'interpolation au plus proche voisin crée des hautes fréquences qui sont seulement dues à une duplication du spectre initial. Ces hautes fréquences sont indésirables et dans ce cas les mesures d'énergie spectrale dans les zones B et C ne sont pas significatives. Il faut donc interpréter ces mesures avec précaution et une analyse de la provenance des hautes fréquences s'avère indispensable.

L'estimation de la qualité d'un agrandissement sans référence reste donc généralement difficile par un autre moyen que l'appréciation visuelle. Cependant, on peut

tout de même analyser une méthode d'agrandissement à partir de cas particuliers où l'on dispose d'une image de référence. Pour cela, on utilise la méthode d'agrandissement à partir d'une version réduite d'une image originale. Cette dernière peut alors servir de référence pour effectuer des mesures de comparaison avec le résultat de l'agrandissement. Dans la littérature, on retrouve souvent cette procédure simple de comparaison pour analyser le comportement expérimental d'une méthode d'agrandissement [29], [46], [124], [149]. Une des mesures les plus utilisées est l'erreur quadratique moyenne MSE (*mean-square error*):

$$MSE = \frac{1}{N^2} \sum_{i,j=0}^{N-1} (R(i,j) - A(i,j))^2 \quad (3.30)$$

où R désigne l'image de référence et A l'image agrandie, toutes deux de taille $N \times N$ pixels. C'est parfois la racine carrée de l'erreur MSE qui est considérée. Dans ce cas, on parle d'erreur $RMSE$ (*root-mean-square error*). Deux autres mesures très courantes sont le rapport signal à bruit SNR (*signal-to-noise ratio*):

$$SNR = 10 \log_{10} \frac{\sum_{i,j=0}^{N-1} R(i,j)^2}{N^2 \cdot MSE} \quad (3.31)$$

et le rapport signal à bruit de crête $PSNR$ (*peak signal-to-noise ratio*):

$$PSNR = 10 \log_{10} + \frac{255^2}{MSE} \quad (3.32)$$

En fait, ces mesures sont issues du domaine de la compression non réversible des images [47]. On peut en trouver de nouvelles dans [7], comme par exemple une mesure évaluant l'effet de bloc, une évaluant la dégradation des contours ou encore une autre évaluant les faux-contours. Cependant, il ne faut pas oublier que ces différentes mesures (MSE , SNR , $PSNR$, *etc.*) ne sont pas toujours significatives et qu'elles restent des mesures globales alors que l'œil humain est quant à lui également sensible aux détails. Une erreur locale est tout de suite détectée par l'œil, alors qu'elle passe inaperçue avec une mesure globale.

L'analyse spectrale peut également être utilisée pour évaluer la qualité de reconstruction de la méthode d'agrandissement. Dans ce cas, on peut comparer directement le spectre de l'image agrandie à celui de l'image de référence [111].

Il faut cependant remarquer que cette approche de l'évaluation de la qualité d'une méthode d'agrandissement assimile le problème de l'agrandissement à un problème de reconstruction qui est pourtant relativement différent. En effet, le nombre de solutions possibles et visuellement acceptables pour un agrandissement n'est pas unique mais en général multiple. Les comparaisons avec un seul représentant des solutions peuvent indiquer à tort que l'agrandissement n'est pas acceptable. Autrement dit, l'appréciation visuelle doit toujours intervenir pour compléter les mesures mathématiques trop rigides.

3.7 Conclusion

Nous avons décrit dans ce chapitre trois grandes approches ayant pour objectif de préserver les structures de l'image originale afin de maintenir une certaine netteté.

La première est l'interpolation adaptative qui exploite toujours le principe de filtrage développé par les méthodes de préservation fréquentielle du chapitre 2, mais qui adapte désormais le filtre aux caractéristiques locales de l'image. Cette approche préserve les contours détectés comme s'il s'agissait de discontinuités parfaites. Cependant, les contours ne traduisent pas toujours des discontinuités mais souvent des déformations continues à la surface des objets. Le fait de traiter tous les contours comme des discontinuités procure un effet artificiel dans l'image agrandie.

La deuxième approche est basée sur l'extrapolation fréquentielle réalisée à partir d'une représentation espace-échelle de l'image originale obtenue par une transformée en ondelettes. Ce type de représentation est particulièrement adapté pour discerner les véritables discontinuités des fausses en exploitant les maxima d'ondelettes. En réalité, on ne peut jamais être certain qu'une discontinuité apparente à une certaine échelle d'analyse soit une véritable discontinuité. Plus le facteur d'agrandissement est important, et plus le risque d'erreur augmente. Cependant, l'analyse espace-échelle permet tout de même d'écarter les contours qui n'apparaissent pas comme des discontinuités à l'échelle d'analyse de l'image originale. Bien que cette approche améliore la netteté de l'image agrandie, elle génère aussi des artefacts dans l'image agrandie lorsque la prédiction est mauvaise [8].

La troisième approche qualifiée de zoom fractal est basée sur les développements réalisés en compression d'images par fractales. Elle exploite le code de similarités structurelles produit à partir de l'image originale : l'image agrandie doit respecter ce code de similarités. L'image agrandie par cette approche présente des fausses discontinuités introduites par les erreurs de similarités. De plus, la phase de recherche des similarités entraîne une complexité algorithmique relativement importante.

Le zoom fractal constitue une approche particulièrement originale de l'agrandissement et demeure encore peu étudié. En effet, le code de similarités est toujours construit selon des critères se référant au domaine de la compression, or les préoccupations en agrandissement en sont assez éloignées. De plus, l'hypothèse de préservation des similarités qui est à la base de la méthode n'a jamais été vérifiée de manière expérimentale. Dans le chapitre 5, nous développons notre propre approche en reprenant à la base le principe de préservation des similarités. Afin de prendre en compte les préoccupations spécifiques à l'agrandissement, nous apportons de nombreuses modifications et améliorations au zoom fractal classique dans les phases d'analyse et de synthèse des similarités. Nous étudions également de manière approfondie l'hypothèse de préservation de similarités entre différentes résolutions de l'image originale.

Le problème de l'agrandissement est en fait un problème inverse de réduction dont les solutions sont multiples. Dans le chapitre 6, nous développons une seconde approche de type ensembliste en imposant des contraintes sur l'ensemble des images

agrandies. Elle se veut complémentaire de notre première approche et plus généralement des autres approches de l'agrandissement, comme par exemple l'interpolation adaptative. Nous verrons dans le chapitre 7 que cette complémentarité est obtenue en projetant l'image agrandie sur l'ensemble des solutions vérifiant le problème inverse de réduction. Cette seconde approche permet également de faire le lien entre les méthodes d'agrandissement qui préservent les fréquences initiales (chapitre 2) et celles qui préservent les structures (chapitre 3). En effet, cette deuxième catégorie de méthodes produit des hautes fréquences sans toutefois se préoccuper de la préservation des basses fréquences. Celles-ci sont justement restaurer en prenant en compte la contrainte de réduction de l'image agrandie.

La réduction d'images est particulièrement exploitée dans nos deux approches de l'agrandissement, que ce soit de manière itérée pour produire une pyramide d'images (chapitres 5 et 7), ou de manière élémentaire (chapitres 6 et 7). Aussi, le chapitre suivant est-il consacré à l'étude de cette opération élémentaire.

Chapitre 4

Réduction et analyse pyramidale d'une image

4.1 Introduction

La réduction d'une image tient un rôle très important dans ce travail. Les deux méthodes d'agrandissement complémentaires que nous proposons dans les chapitres 5 et 6 sont basées sur la notion de réduction. Dans la première méthode, nous exploitons l'information produite par des réductions successives de l'image pour inverser le processus et extrapoler un niveau de résolution supérieur (chapitre 5). Dans la deuxième méthode, nous posons le problème de l'agrandissement comme un problème inverse de réduction et nous nous intéressons à l'ensemble des images agrandies qui génèrent l'image initiale par réduction (chapitre 6).

Pour identifier les traitements à mettre en œuvre pour réduire une image, nous effectuons une analyse fréquentielle de la réduction dans la section 4.2. Il s'agit avant tout d'un cadre théorique pour développer des méthodes de réduction rapides. Nous nous intéressons ici à deux types de réduction. La première est une réduction classique où l'on préserve un échantillon sur quatre et la deuxième est une réduction en quinconce où l'on préserve un échantillon sur deux. Dans le premier cas, le facteur de réduction dans chaque direction est 2 et dans le deuxième, il est de $\sqrt{2}$.

En itérant la réduction d'une image, on obtient une représentation multi-résolution que l'on désigne par le terme *pyramide*. Ce type de représentation facilite beaucoup la mise en place de certains traitements. Nous l'utilisons dans le chapitre 5 pour inverser la réduction et agrandir l'image initiale. Nous étudions différentes approches dans la section 4.3 pour construire rapidement une pyramide avec un facteur de réduction 2.

Pour obtenir une représentation pyramidale avec un facteur de réduction moins brutal, nous étudions également la construction rapide d'une pyramide en quinconce dans la section 4.4. En pratique, on obtient un facteur de réduction $\sqrt{2}$ avec des

niveaux intermédiaires par rapport aux pyramides où le facteur de réduction est 2.

4.2 Analyse fréquentielle d'une réduction

Réduire la résolution d'une image, c'est diminuer le nombre d'échantillons nécessaires à sa représentation. Une décimation pure et simple des échantillons revient à sélectionner arbitrairement des échantillons. Ceci conduit inévitablement à faire disparaître des structures et à en créer de nouvelles dans l'image réduite. En infographie, ce phénomène est couramment appelé *aliasing*. Intuitivement, on imagine que la solution à ce problème est de combiner les valeurs des échantillons entre elles avant d'effectuer la décimation. Cependant, on peut s'interroger sur la manière de réaliser cette combinaison. L'analyse spectrale permet d'avoir une vision claire sur ce problème. En fait, pour éviter l'aliasing, nous allons voir que la *décimation spatiale*¹ doit être précédée d'une *décimation fréquentielle*².

4.2.1 Décimation spatiale et décimation fréquentielle

La suppression d'échantillons d'une image peut se modéliser simplement à l'aide d'une *fonction de décimation spatiale* définie par :

$$\begin{cases} d_s(x, y) = 1 \text{ pour un échantillon conservé} \\ d_s(x, y) = 0 \text{ pour un échantillon supprimé} \end{cases} \quad (4.1)$$

Décimer spatialement une image consiste alors à multiplier la fonction $f(x, y)$ de l'image par la fonction de décimation $d_s(x, y)$, c'est-à-dire :

$$g(x, y) = f(x, y) \cdot d_s(x, y) \quad (4.2)$$

Grâce à la propriété de convolution décrite par la relation (2.7), l'opération précédente peut s'interpréter dans le domaine fréquentiel comme une convolution du spectre de la fonction de décimation par le spectre de l'image :

$$G(u, v) = D_s * F(u, v) \quad (4.3)$$

Si la fonction $d_s(x, y)$ comporte des valeurs nulles, alors la convolution décrite dans l'équation (4.3) entraîne nécessairement des recouvrements du spectre initial $F(u, v)$. Ceci génère en pratique de «fausses» hautes fréquences dans le spectre $G(u, v)$, c'est-à-dire de l'aliasing dans l'image décimée $g(x, y)$. Pour éviter ce problème, il faut supprimer les hautes fréquences de $F(u, v)$ qui sont concernées par les recouvrements. Pour cela, préalablement à la décimation spatiale, on multiplie $F(u, v)$ par une *fonction de décimation fréquentielle* définie de la manière suivante :

$$\begin{cases} D_f(u, v) = 1 \text{ pour une fréquence conservée} \\ D_f(u, v) = 0 \text{ pour une fréquence supprimée} \end{cases} \quad (4.4)$$

1. Décimation spatiale : processus d'annulation périodique de la valeur image.

2. Décimation fréquentielle : processus d'annulation d'une certaine bande de fréquences.

Dans le domaine fréquentiel, la décimation totale s'exprime alors par :

$$G(u, v) = D_s * (F(u, v) \cdot D_f(u, v)) \quad (4.5)$$

ou de manière équivalente dans le domaine spatial :

$$g(x, y) = d_s(x, y) \cdot (f * d_f(x, y)) \quad (4.6)$$

La fonction $D_f(u, v)$ a pour but de laisser passer ou de bloquer des fréquences particulières de l'image à réduire. Il s'agit donc d'un *filtre fréquentiel* et sa réponse impulsionnelle $d_f(x, y)$ est appelée *filtre spatial*, ou tout simplement *filtre*, s'il n'y a pas d'ambiguïté. En pratique, la fonction $D_f(u, v)$ est construite en observant quelques règles. La principale est que la convolution $D_s * D_f(u, v)$ ne doit générer que des valeurs nulles ou unitaires. Cette règle assure l'absence de recouvrements du spectre $D_f(u, v)$. Ce dernier doit également posséder les propriétés de symétries centrale et axiale pour ne pas privilégier une direction. En outre, il faut sauvegarder en priorité les basses fréquences, c'est-à-dire que la décimation doit se faire des hautes vers les basses fréquences.

Dans la suite, nous étudions les fonctions de décimation fréquentielle (ou filtres) à utiliser avec diverses fonctions de décimation spatiale. Nous étudions le cas d'une réduction de facteur 2 sur un maillage carré, le cas d'une réduction d'un maillage carré à un maillage en quinconce et son inverse (facteur de réduction $\sqrt{2}$).

4.2.2 Réduction de facteur 2 d'un maillage carré

La réduction d'une image par un facteur 2 en horizontal et en vertical, consiste simplement à supprimer un échantillon sur deux dans chaque direction. Dans ce cas, la fonction de décimation spatiale s'exprime simplement par :

$$\begin{cases} d_s(x, y) = 1 & \text{lorsque } x \text{ et } y \text{ sont pairs} \\ d_s(x, y) = 0 & \text{sinon} \end{cases} \quad (4.7)$$

En utilisant la transformée de Fourier définie par l'équation (2.3) et en supposant que l'image à réduire a une taille de $2N \times 2N$ pixels, on trouve le spectre suivant :

$$\begin{cases} D_s(u, v) = N^2 & \text{lorsque } u \bmod N = 0 \text{ et } v \bmod N = 0 \\ D_s(u, v) = 0 & \text{sinon} \end{cases} \quad (4.8)$$

Les figures 4.1(a) et 4.1(b) illustrent respectivement la fonction de décimation spatiale et son spectre fréquentiel pour $N = 4$.

La fonction de décimation fréquentielle optimale $D_f(u, v)$ permettant d'éviter le recouvrement spectral issu de la convolution de $F(u, v)$ avec $D_s(u, v)$ est donnée par :

$$(-N \leq u, v < N) \begin{cases} D_f(u, v) = 1 & \text{lorsque } -\frac{N}{2} \leq u, v < \frac{N}{2} \\ D_f(u, v) = 0 & \text{sinon} \end{cases} \quad (4.9)$$

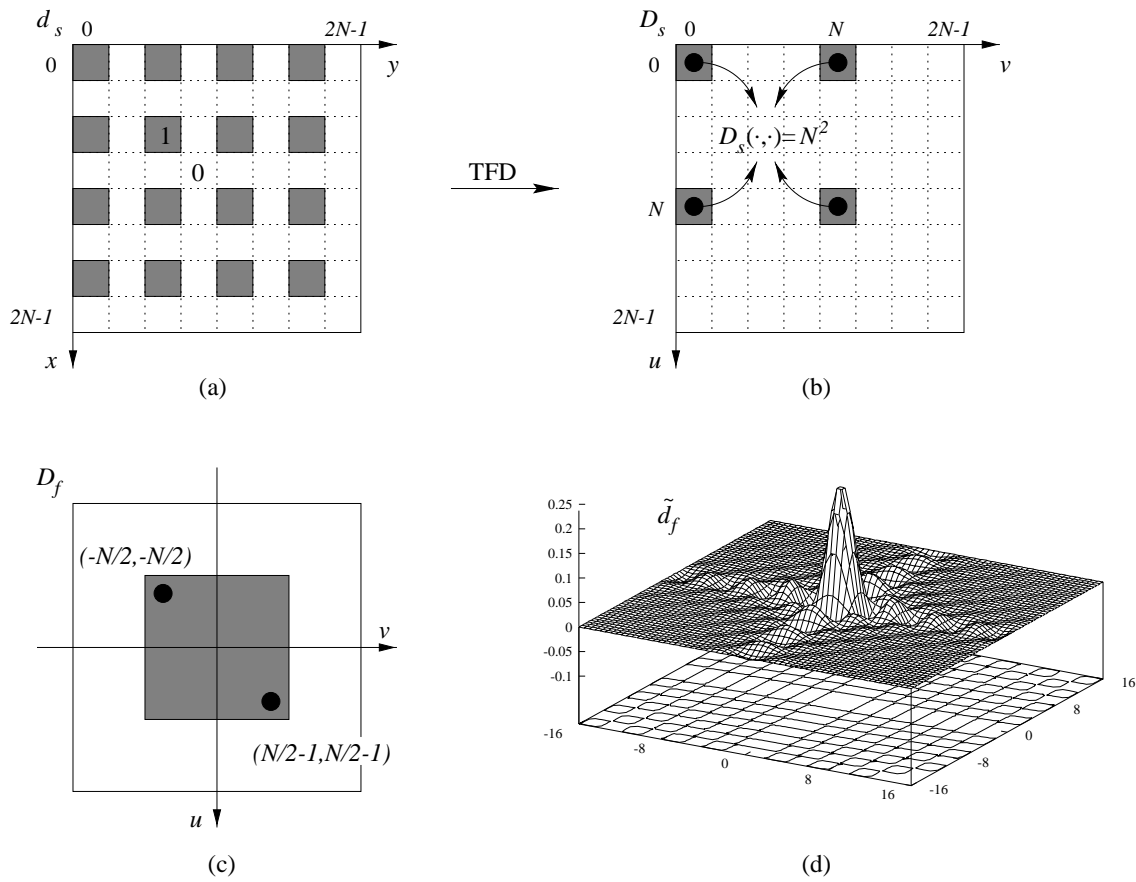


Figure 4.1 – Réduction de facteur 2 d'un maillage carré: (a) fonction de décimation spatiale $d_s(x, y)$, (b) spectre fréquentiel associé, (c) fonction de décimation fréquentielle $D_f(u, v)$, (d) limite de sa réponse impulsionnelle.

La figure 4.1(c) illustre cette fonction. Sa réponse impulsionnelle dans le domaine spatial s'exprime par :

$$d_f(x, y) = \frac{1}{4N^2} \sum_{u, v = -\frac{N}{2}}^{\frac{N}{2}-1} e^{\frac{\pi i}{N}(ux+vy)} = \frac{1}{4N^2} \cdot \cos \frac{\pi(x+y)}{2N} \cdot \frac{\sin \frac{\pi x}{2} \sin \frac{\pi y}{2}}{\sin \frac{\pi x}{2N} \sin \frac{\pi y}{2N}} \quad (4.10)$$

Lorsque N tend vers l'infini, on peut faire une approximation continue de la somme et obtenir la limite de la réponse impulsionnelle :

$$\tilde{d}_f(x, y) = \frac{1}{4N^2} \int_{-\frac{N}{2}}^{\frac{N}{2}} \int_{-\frac{N}{2}}^{\frac{N}{2}} e^{\frac{\pi i}{N}(ux+vy)} dudv = \frac{1}{4} \cdot \text{sinc} \frac{\pi}{2} x \cdot \text{sinc} \frac{\pi}{2} y \quad (4.11)$$

où la fonction sinc est la fonction *sinus cardinal* définie par $\text{sinc}(x) = \sin(x)/x$. Remarquons que nous retrouvons là les fonctions h et \tilde{h} du chapitre 2, section 2.3.1, avec $d_f(x, y) = h(x/2, y/2)/4$ et $\tilde{d}_f(x, y) = \tilde{h}(x/2, y/2)/4$.

D'après l'équation (4.6), réduire d'un facteur 2 une image de taille $2N \times 2N$ pixels consiste d'abord à convoluer celle-ci par le filtre discret $d_f(x, y)$ défini par (4.10),

puis à sous-échantillonner le résultat en utilisant la fonction de décimation spatiale donnée par (4.7). Dans le cas où N est suffisamment grand, on peut approximer $d_f(x, y)$ par sa limite $\tilde{d}_f(x, y)$. Cette dernière est illustrée sur la figure 4.1(d). Son comportement oscillatoire dans les directions horizontale et verticale, provient des discontinuités de la fonction de décimation fréquentielle. Une localisation précise dans le domaine fréquentiel est forcément répartie sur tous les échantillons du domaine spatial.

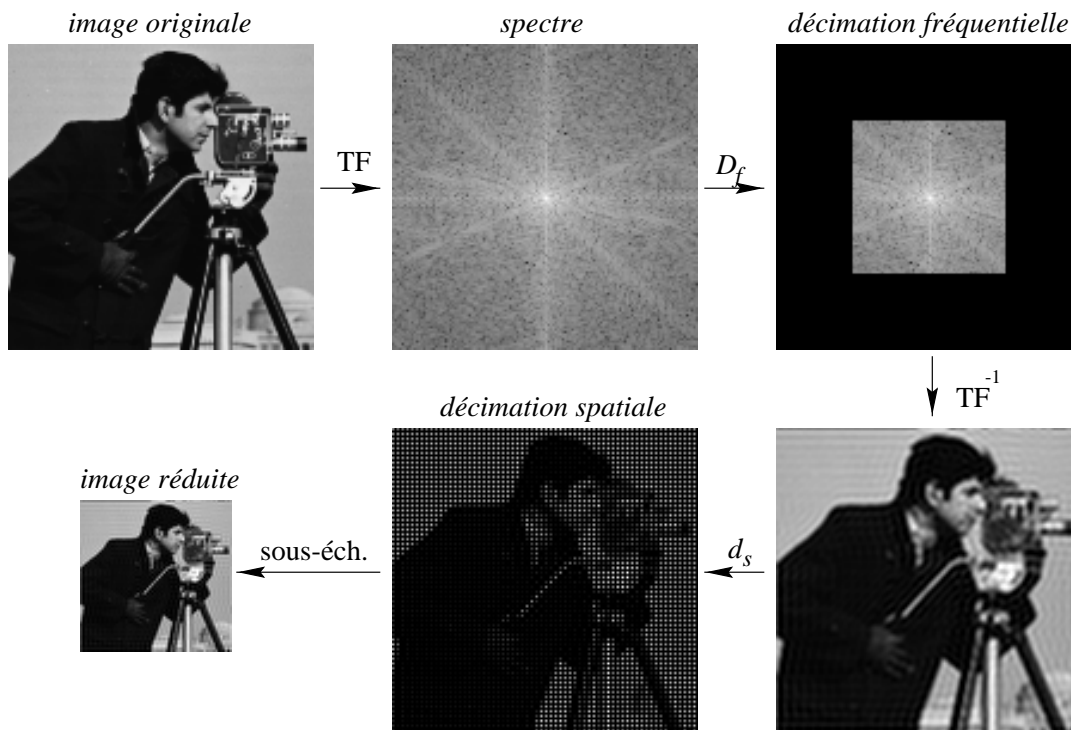


Figure 4.2 – Réduction de facteur 2 par analyse spectrale.

La figure 4.2 montre les différentes étapes d'une réduction de facteur 2 de l'image du cameraman par analyse spectrale. La première étape consiste à calculer le spectre fréquentiel de l'image avec la transformée de Fourier discrète. Dans la seconde étape, les hautes fréquences du spectre sont supprimées en utilisant la fonction de décimation fréquentielle définie par (4.9). Dans la troisième étape, l'image équivalente dans le domaine spatial est calculée avec la transformée de Fourier inverse. Enfin, dans la quatrième et dernière étape, l'image est sous-échantillonnée en utilisant la fonction de décimation spatiale définie par (4.7).

La réduction illustrée sur la figure 4.2 revient à exploiter les deux expressions (4.5) et (4.6) à la fois. La décimation fréquentielle est effectuée dans le domaine fréquentiel et la décimation spatiale est réalisée dans le domaine spatial. Cette solution est plus facile à mettre en œuvre. On peut cependant obtenir un résultat identique en effectuant toutes les opérations dans le domaine spatial, ce qui revient à exploiter uniquement l'expression (4.6). Pour cela, il suffit de remplacer les trois premières

étapes de la figure 4.2 par une convolution avec le filtre discret défini par (4.10), ou par sa limite illustrée sur la figure 4.1(d), lorsque N est suffisamment grand.

L'image réduite révèle un artefact de la méthode. La suppression des hautes fréquences du spectre entraîne un effet d'ondulation des contours de l'image. C'est une conséquence directe du comportement oscillatoire des fonctions sinusoidales. Cet artefact est connu sous le nom de *phénomène de Gibbs*. Pour l'éliminer, on peut utiliser une fonction d'apodisation pour adoucir les discontinuités présentes dans la fonction de décimation fréquentielle. Cette solution revient à diminuer les oscillations du filtre illustré sur la figure 4.1(d).

4.2.3 Réduction d'un maillage carré à un maillage en quinconce

La réduction d'un maillage carré à un maillage en quinconce consiste à supprimer un point sur deux du signal $f(x, y)$ de manière à ce que le nombre d'échantillons préservés sur une ligne soit égal à celui préservé sur une colonne. Cette situation est décrite sur la figure 4.3. La fonction de décimation spatiale se définit donc de la manière suivante :

$$\begin{cases} d_s(x, y) = 1 & \text{lorsque } x + y \text{ est pair} \\ d_s(x, y) = 0 & \text{sinon} \end{cases} \quad (4.12)$$

Pour une image de $2N \times 2N$ pixels, le spectre fréquentiel de $d_s(x, y)$ est donné par :

$$\begin{cases} D_s(u, v) = 2N^2 & \text{lorsque } (u, v) = (0, 0) \text{ et } (u, v) = (N, N) \\ D_s(u, v) = 0 & \text{sinon} \end{cases} \quad (4.13)$$

La fonction de décimation spatiale et son spectre sont illustrés respectivement sur les figures 4.4(a) et 4.4(b) pour $N = 4$.

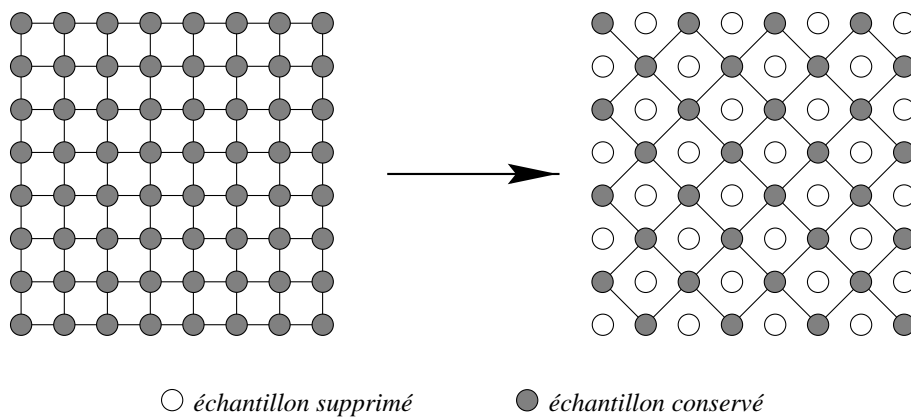


Figure 4.3 – Réduction d'un maillage carré à un maillage en quinconce (facteur de réduction racine de 2).

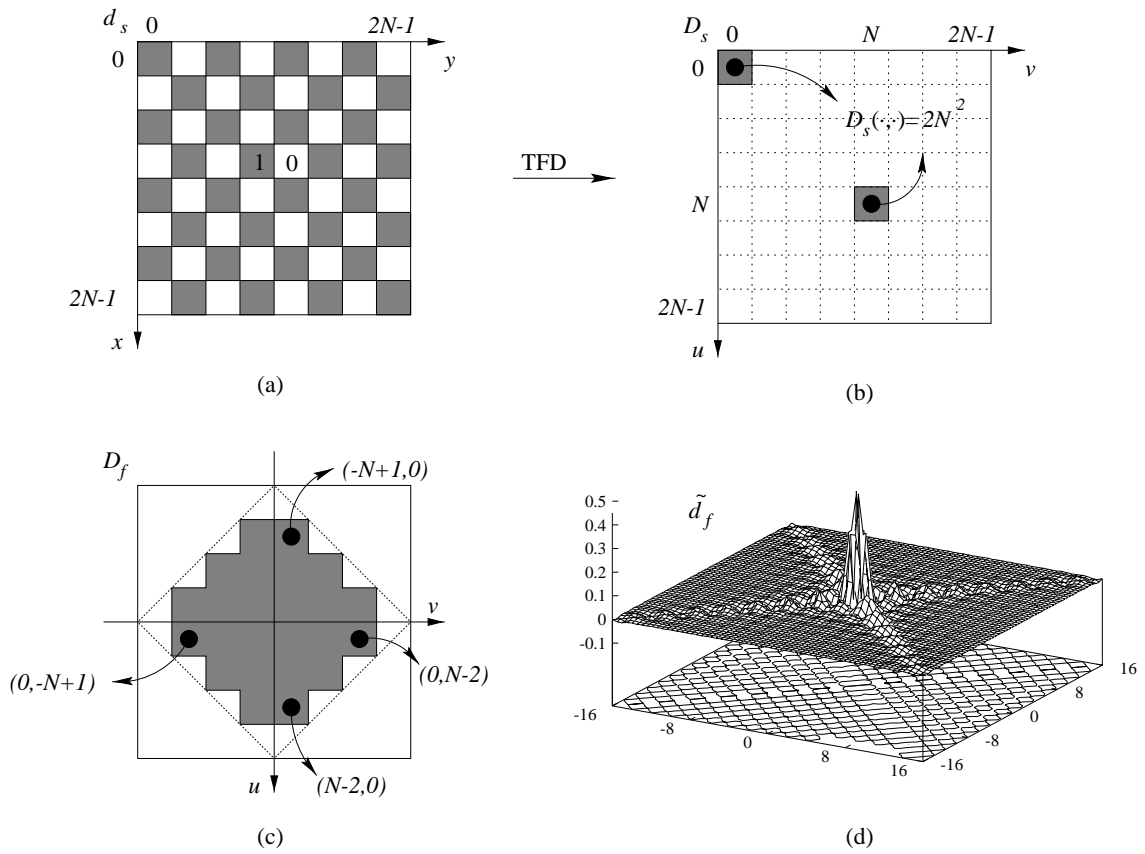


Figure 4.4 – Réduction d'un maillage carré à un maillage en quinconce : (a) fonction de décimation spatiale $d_s(x,y)$, (b) spectre fréquentiel associé, (c) fonction de décimation fréquentielle $D_f(u,v)$, (d) limite de sa réponse impulsionnelle.

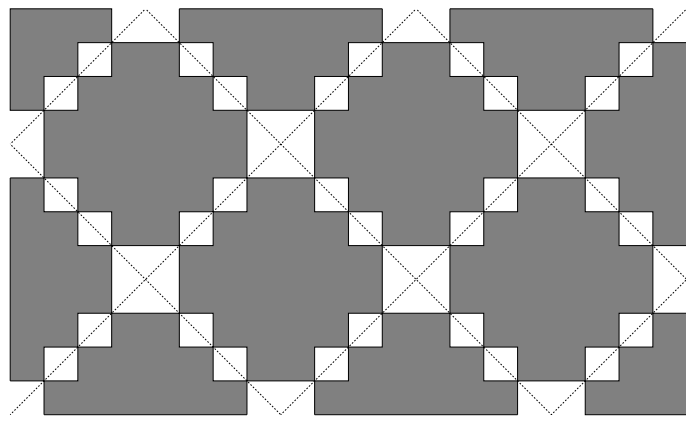


Figure 4.5 – Pavage du spectre discret avec la fonction de décimation fréquentielle optimale ($N = 4$).

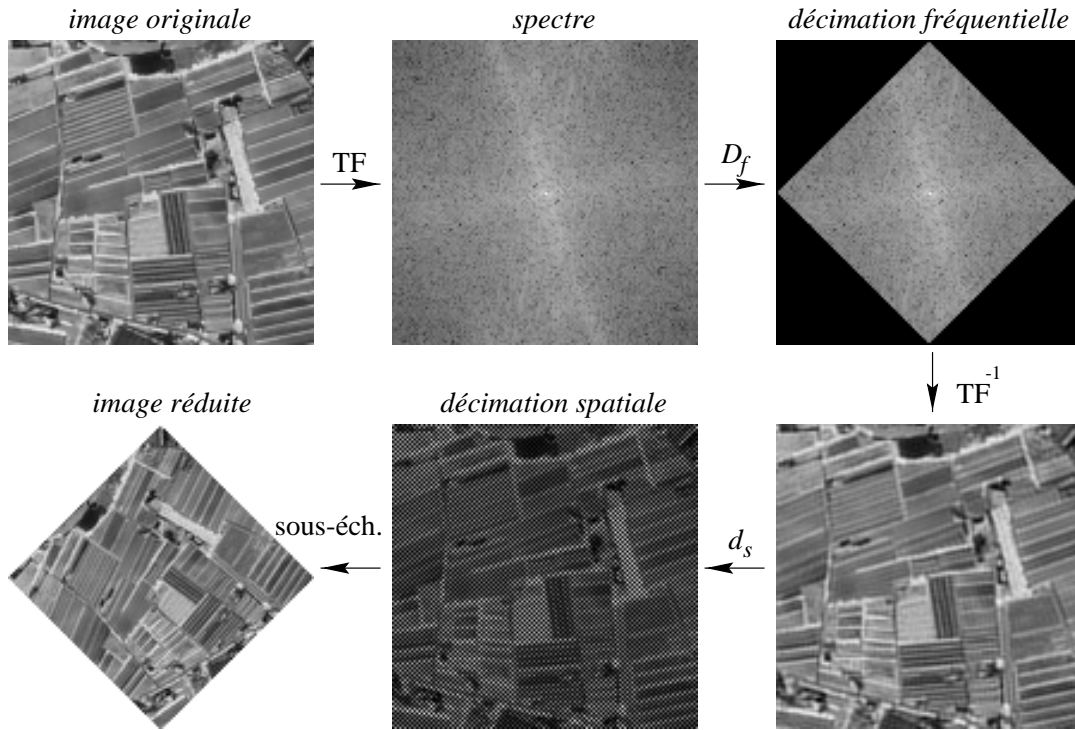


Figure 4.6 – Réduction d'un maillage carré à un maillage en quinconce par analyse spectrale.

La fonction de décimation fréquentielle optimale $D_f(u, v)$ associée à la fonction de décimation spatiale $d_s(x, y)$ est donnée par :

$$\begin{cases} D_f(u, v) = 1 & \text{lorsque} \begin{cases} 0 \leq u < N - 1 & \wedge & -N + u < v < N - u - 1 \\ -N < u < 0 & \wedge & -N - u < v < N + u \end{cases} \\ D_f(u, v) = 0 & \text{sinon} \end{cases} \quad (4.14)$$

La figure 4.4(c) illustre cette fonction. Avec un signal $f(x, y)$ qui possède un nombre d'échantillons dyadique (puissance de 2), on remarque que la fonction $D_f(u, v)$ ne pave pas complètement le plan (voir la figure 4.5). En pratique, ce problème n'est pas très important puisque N est suffisamment grand pour minimiser les effets de bord, et les coefficients non recouverts sont généralement très faibles.

La réponse impulsionnelle de la fonction $D_f(u, v)$ définie par (4.14) s'exprime par :

$$d_f(x, y) = \frac{1}{4N^2} \sum_{u=0}^{N-2} \sum_{v=-N+1+u}^{N-2-u} e^{\frac{\pi i}{N}(ux+vy)} + \frac{1}{4N^2} \sum_{u=-N+1}^{-1} \sum_{v=-N-u}^{N-1+u} e^{\frac{\pi i}{N}(ux+vy)} \quad (4.15)$$

En supposant que N est suffisamment grand, on peut faire une approximation continue de cette somme et obtenir la limite de $d_f(x, y)$ suivante :

$$\begin{aligned} \tilde{d}_f(x, y) &= \frac{1}{4N^2} \int_0^N \int_{-N+u}^{N-u} e^{\frac{\pi i}{N}(ux+vy)} dudv + \frac{1}{4N^2} \int_{-N}^0 \int_{-N-u}^{N+u} e^{\frac{\pi i}{N}(ux+vy)} dudv \\ &= \frac{\cos \pi x - \cos \pi y}{\pi^2(y^2 - x^2)} \end{aligned} \quad (4.16)$$

Cette fonction est illustrée sur la figure 4.4(d). Son apparence est proche de celle de la fonction sinus cardinal 2D montrée sur la figure 4.2(d). Le passage de l'une à l'autre se faisant par une rotation de 45° . En fait, les ondulations sont orthogonales aux discontinuités de la fonction de décimation fréquentielle illustrée sur la figure 4.4(c).

La figure 4.6 illustre l'utilisation des fonctions de décimation fréquentielle et spatiale (définies respectivement par (4.14) et (4.12)) pour obtenir la réduction en quinconce d'une image de Carpentras (128×128 pixels). La fonction de décimation fréquentielle en losange supprime uniquement les hautes fréquences diagonales de l'image originale. Toutes les fréquences horizontales et verticales sont préservées. Cette opération de décimation fréquentielle peut être réalisée de manière équivalente dans le domaine spatial en convoluant l'image originale soit avec le filtre discret défini par (4.15), soit par le filtre continu défini par (4.16) lorsque N est suffisamment grand. Comme dans l'exemple de la section 4.2.2 (figure 4.2), on peut relever un phénomène d'ondulation des contours, principalement selon l'axe diagonal. Cet effet apparaît plus clairement sur la figure 4.10 où l'image initiale est une mire noire et blanche de 128×128 pixels.

4.2.4 Réduction d'un maillage en quinconce à un maillage carré

La réduction d'un maillage en quinconce à un maillage carré revient à supprimer une ligne sur deux ou, ce qui est équivalent, une colonne sur deux du signal $f(x, y)$. Ce principe de décimation est montré sur la figure 4.7.

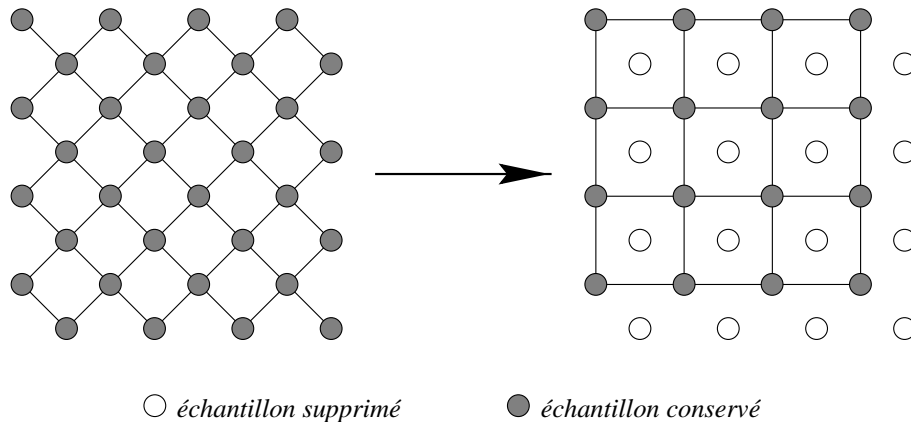


Figure 4.7 – Réduction d'un maillage en quinconce à un maillage carré (facteur de réduction racine de 2).

De manière à simplifier l'analyse, il est pratique de remplacer le maillage en quinconce par un maillage carré ayant un point sur deux à valeur nulle. Ainsi, la fonction de décimation spatiale s'exprime simplement par :

$$\begin{cases} d_s(x, y) = 1 & \text{lorsque } x \text{ ou } y \text{ est pair} \\ d_s(x, y) = 0 & \text{sinon} \end{cases} \quad (4.17)$$

Pour une image de $2N \times 2N$ pixels, le spectre fréquentiel de $d_s(x, y)$ est donné par :

$$\begin{cases} D_s(u, v) = 3N^2 & \text{lorsque } (u, v) = (0, 0) \\ D_s(u, v) = N^2 & \text{lorsque } (u, v) = (N, 0) \text{ et } (u, v) = (0, N) \\ D_s(u, v) = -N^2 & \text{lorsque } (u, v) = (N, N) \\ D_s(u, v) = 0 & \text{sinon} \end{cases} \quad (4.18)$$

Les figures 4.8(a) et 4.8(b) montrent respectivement la fonction de décimation spatiale et son spectre pour $N = 4$. La fonction de décimation fréquentielle optimale $D_f(u, v)$ associée à la fonction de décimation spatiale $d_s(x, y)$ est donnée par :

$$\begin{cases} D_f(u, v) = 1 & \text{lorsque } \begin{cases} -N \leq u, v \leq -\frac{N}{2} - 1 \\ -\frac{N}{2} \leq u, v \leq \frac{N}{2} - 1 \\ \frac{N}{2} \leq u, v \leq N - 1 \\ -N \leq u \leq -\frac{N}{2} - 1 \wedge \frac{N}{2} \leq v \leq N - 1 \\ \frac{N}{2} \leq u \leq N - 1 \wedge -N \leq v \leq -\frac{N}{2} - 1 \end{cases} \\ D_f(u, v) = 0 & \text{sinon} \end{cases} \quad (4.19)$$

Cette fonction est illustrée sur la figure 4.8(c). Sa réponse impulsionnelle s'écrit de la manière suivante :

$$\begin{aligned} d_f(x, y) = & \frac{1}{4N^2} \left(\sum_{u=-N}^{-\frac{N}{2}-1} \sum_{v=-N}^{-\frac{N}{2}-1} e^{\frac{\pi i(ux+vy)}{N}} + \sum_{u=-\frac{N}{2}}^{-1} \sum_{v=-\frac{N}{2}}^{-1} e^{\frac{\pi i(ux+vy)}{N}} + \sum_{u=\frac{N}{2}}^{N-1} \sum_{v=\frac{N}{2}}^{N-1} e^{\frac{\pi i(ux+vy)}{N}} \right. \\ & \left. + \sum_{u=-N}^{-\frac{N}{2}-1} \sum_{v=\frac{N}{2}}^{N-1} e^{\frac{\pi i(ux+vy)}{N}} + \sum_{u=\frac{N}{2}}^{-1} \sum_{v=-N}^{-\frac{N}{2}-1} e^{\frac{\pi i(ux+vy)}{N}} \right) \end{aligned} \quad (4.20)$$

En supposant que N est suffisamment grand, on peut faire une approximation continue de cette somme et obtenir la limite de $d_f(x, y)$:

$$\begin{aligned} \tilde{d}_f(x, y) = & \frac{1}{4N^2} \left(\int_{-N}^{-\frac{N}{2}} \int_{-N}^{-\frac{N}{2}} e^{\frac{\pi i(ux+vy)}{N}} dudv + \int_{-\frac{N}{2}}^{\frac{N}{2}} \int_{-\frac{N}{2}}^{\frac{N}{2}} e^{\frac{\pi i(ux+vy)}{N}} dudv \right. \\ & + \int_{\frac{N}{2}}^N \int_{\frac{N}{2}}^N e^{\frac{\pi i(ux+vy)}{N}} dudv + \int_{-N}^{-\frac{N}{2}} \int_{\frac{N}{2}}^N e^{\frac{\pi i(ux+vy)}{N}} dudv \\ & \left. + \int_{\frac{N}{2}}^N \int_{-N}^{-\frac{N}{2}} e^{\frac{\pi i(ux+vy)}{N}} dudv \right) \\ = & \frac{1}{2} \left(\text{sinc } \pi x \cdot \text{sinc } \pi y + (\text{sinc } \pi x - \text{sinc } \frac{\pi}{2} x)(\text{sinc } \pi y - \text{sinc } \frac{\pi}{2} y) \right) \end{aligned} \quad (4.21)$$

Cette fonction est illustrée sur la figure 4.8(d). Elle apparaît plus complexe que les deux autres fonctions de décimation fréquentielle que nous avons déjà pu voir.

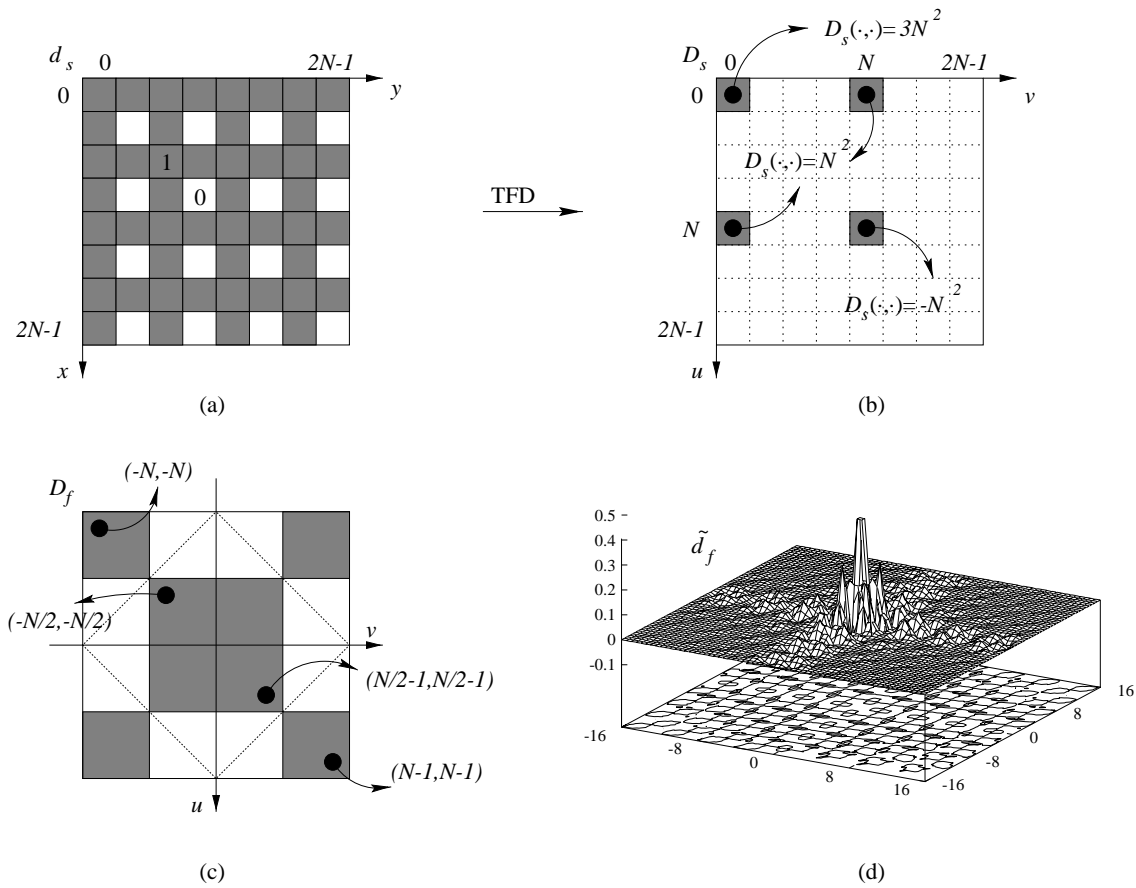


Figure 4.8 – Réduction d'un maillage en quinconce à un maillage carré : (a) fonction de décimation spatiale $d_s(x, y)$, (b) spectre fréquentiel associé, (c) fonction de décimation fréquentielle $D_f(u, v)$, (d) limite de sa réponse impulsionnelle.

La figure 4.9 illustre l'utilisation des fonctions de décimation fréquentielle et spatiale définies respectivement par (4.19) et (4.17) pour réduire l'image en quinconce montrée sur la figure 4.6. La fonction de décimation fréquentielle supprime les hautes fréquences horizontales et verticales de l'image en quinconce et laisse intactes ses fréquences diagonales. Dans ce cas, l'effet d'ondulation se retrouve dans les directions horizontales et verticales.

La fonction de décimation fréquentielle de la figure 4.9 complète celle illustrée sur la figure 4.6. La réduction de facteur 2 obtenue par l'enchaînement de ces deux fonctions est comparable à celle obtenue avec la fonction de décimation fréquentielle illustrée sur la figure 4.2. Cependant, la réduction de facteur 2 qui procède en deux étapes (carré-quinconce-carré) donne une image réduite légèrement moins bonne que celle qui procède en une seule étape (carré-carré). Ceci provient des ondulations diagonales qui s'ajoutent aux ondulations verticales et horizontales. On peut observer ce phénomène sur la figure 4.10.

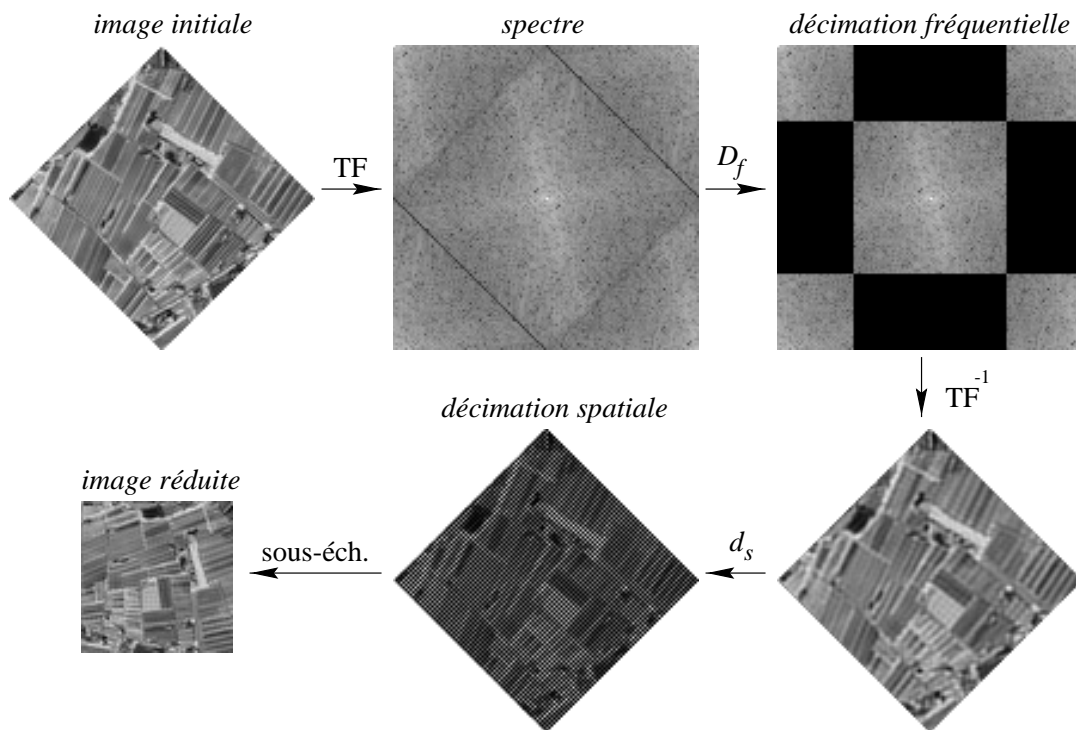


Figure 4.9 – Réduction d'un maillage en quinconce à un maillage carré par analyse spectrale.

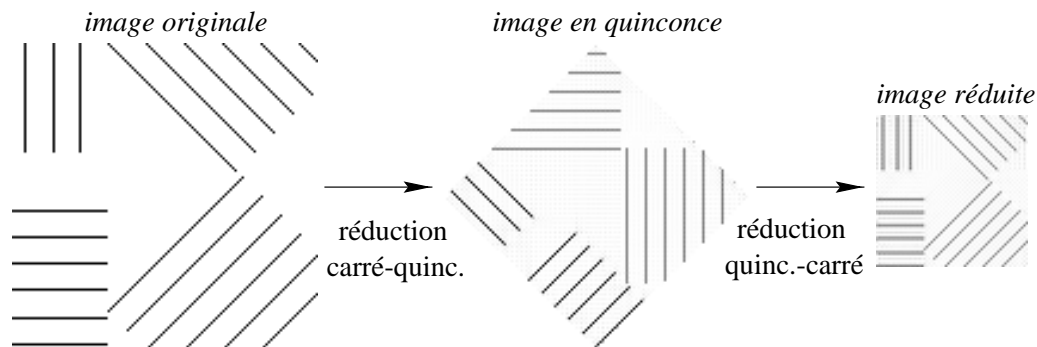


Figure 4.10 – Illustration du phénomène d'ondulation.

4.3 Analyse pyramidale

En itérant la réduction d'une image, on obtient une représentation pyramidale où chaque niveau exprime une résolution particulière de l'image. Ce modèle de représentation facilite beaucoup certains traitements de l'image. Il permet une analyse progressive de l'information en parcourant les niveaux de la pyramide depuis l'apex jusqu'à la base. Les niveaux supérieurs représentent les structures générales de l'image, tandis que les niveaux inférieurs décrivent plutôt les détails.

Les applications de l'analyse pyramidale sont nombreuses. On peut citer notamment la mise en correspondance, la détection des contours, la segmentation, ou

encore la reconnaissance d'objets [76], [77]. Dans le chapitre 5, nous utilisons la représentation pyramidale pour agrandir l'image.

Dans cette section, nous nous intéressons uniquement aux pyramides à facteur de réduction 2. Le cas d'une pyramide en quinconce à facteur de réduction $\sqrt{2}$ est étudié dans la section suivante.

La construction d'une pyramide s'appuie sur l'opération de réduction d'une image. En théorie, la réduction de facteur 2 par analyse fréquentielle décrite dans la section précédente permet de préserver idéalement l'information pour la construction des différents niveaux de la pyramide. Cette approche a été étudiée par Watson pour construire des pyramides dites *idéales* [146]. En pratique, le phénomène de Gibbs, qui se traduit par des ondulations dans l'image, limite l'intérêt de l'approche fréquentielle. Pour corriger cet artefact, une solution envisageable est d'adoucir les discontinuités du filtre fréquentiel (ou fonction de décimation fréquentielle) par une fonction d'apodisation. Une autre solution est de réaliser la réduction uniquement dans le domaine spatial en convoluant l'image avec la réponse impulsionnelle du filtre fréquentiel idéal. En prenant un support suffisamment petit, par exemple 9×9 pixels, on arrive à limiter globalement l'effet d'ondulation. Ceci revient à tronquer le filtre spatial idéal $\tilde{d}_f(x, y)$ défini par (4.11). Meer *et al.* proposent une autre approche qui consiste à construire un filtre à petit support dans le domaine spatial et à optimiser son spectre fréquentiel vis-à-vis du filtre fréquentiel idéal $D_f(u, v)$ donné par (4.9) [94]. La solution d'une réduction par convolution avec un petit filtre se révèle également très efficace pour ce qui est des calculs. La complexité algorithmique est dans ce cas $O(N^2)$, alors qu'elle est $O(N^2 \log_2 N)$ dans le cas d'une réduction par analyse spectrale.

Chaque niveau d'une pyramide représente l'image originale moins une certaine bande de fréquences supprimée par le filtrage passe-bas réalisé dans les réductions successives. Pour cette raison, ce type de pyramide est communément désignée par le terme *pyramide passe-bas*.

Dans la suite, nous étudions l'approche de Meer *et al.*, ainsi que celles de Burt [21] et de Chin *et al.* [38] où le filtre est construit uniquement sous des contraintes spatiales. Dans ces trois approches, le filtre 2D est *séparable*, c'est-à-dire qu'il s'exprime comme le produit de deux filtres 1D. Ceci permet de simplifier la définition du filtre 2D, mais aussi de diminuer la quantité de calculs. De plus, comme il n'y a pas de raison de privilégier une orientation particulière, les deux filtres 1D sont en fait identiques. Dans ce cas, le filtre spatial 2D de largeur K s'exprime de la manière suivante :

$$h(x, y) = w(x) \cdot w(y) \quad 0 \leq x, y < K \quad (4.22)$$

Le spectre fréquentiel a la propriété d'être également séparable. Si le filtrage se fait sur une image de taille $2N \times 2N$ pixels, alors on obtient :

$$H(u, v) = W(u) \cdot W(v) \quad 0 \leq u, v < 2N \quad (4.23)$$

Dans la suite, on s'intéresse uniquement au filtre spatial $w(x)$ et à son spectre $W(u)$.

Cette étude se termine par une comparaison expérimentale des trois approches dans la section 4.3.4.

Dans la construction d'une pyramide passe-bas, l'opération de réduction supprime à chaque niveau une bande de fréquences supplémentaire de l'image originale. On peut relever ces bandes de fréquences pour construire un nouveau type de pyramide : la *pyramide de différences* qui décompose l'image originale selon des bandes de fréquences. Nous étudions la construction de cette pyramide dans la section 4.3.5. Ce type de pyramide nous servira dans le chapitre 5 pour sélectionner les blocs qui, dans une pyramide passe-bas, contiennent de l'information fréquentielle pertinente.

4.3.1 La pyramide de Meer

Dans l'approche de Meer *et al.*, les poids du filtre spatial sont déterminés pour optimiser la réponse en fréquence vis-à-vis du filtre fréquentiel idéal [94]. La figure 4.11(a) illustre le spectre typique du filtre spatial optimal. On peut distinguer trois zones : la bande passante, la bande de transition et la bande stoppante. Dans la bande passante ($0 < u < u_p$), l'amplitude est proche de la valeur optimale 1. Dans la bande de transition ($u_p < u < u_s$), l'amplitude décroît vers une valeur nulle. Dans la bande stoppante, l'amplitude demeure proche de 0. Les amplitudes des oscillations présentes dans la bande passante et dans la bande stoppante sont notées respectivement δ_p et δ_s .

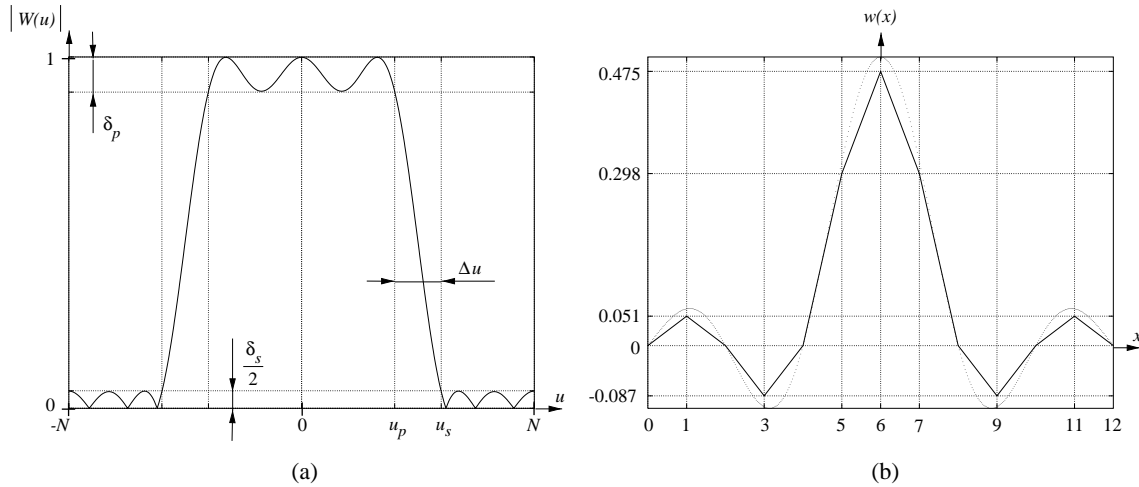


Figure 4.11 – (a) Spectre d'amplitude d'un filtre optimal. (b) Filtre optimal pour $K = 13$.

En prenant comme référence le filtre idéal $D_f(u)$ dont l'expression sur 2 dimensions est définie par (4.9), on peut montrer que le filtre $W(u)$ représenté sur la figure 4.11(a) minimise le critère d'erreur de Chebyshev [94] :

$$E = \min_{W(u)} \max_{\substack{0 \leq u \leq u_p \\ u_s \leq u < N}} |W(u) - D_f(u)| \quad (4.24)$$

Pour diminuer cette erreur, il faut réduire la bande de transition $\Delta u = u_s - u_p$. En fait, cette dernière est liée à la largeur K du filtre spatial par la formule empirique

suivante [43], [94] :

$$K \approx \frac{2N \cdot D_\infty(\delta_p, \delta_s)}{\Delta u} \quad (4.25)$$

où $D_\infty(\delta_p, \delta_s)$ est une fonction tabulée avec des valeurs variant entre 1.25 et 4. Cette fonction croît lorsque la tolérance sur les oscillations diminue, c'est-à-dire lorsque δ_p et δ_s sont de plus en plus petits. La relation (4.25) montre que la bande de transition diminue lorsque la taille du filtre augmente. Autrement dit, pour que le filtre soit de bonne qualité, il faut que sa largeur soit suffisamment importante. Il faut donc faire un compromis entre qualité et complexité algorithmique. Meer *et al.* proposent pour le filtre optimal $W(u)$ les paramètres suivants :

$$K = 13 \quad , \quad u_p + u_s = N \quad , \quad \Delta u = \frac{N}{5} \quad , \quad \delta_p = \delta_s = 0.1 \quad (4.26)$$

Le filtre spatial $w(x)$ correspondant est montré sur la figure 4.11(b). Le filtre spatial idéal, représenté par un sinus cardinal, est superposé en pointillé pour permettre la comparaison. On remarque que les deux filtres sont approximativement similaires pour les valeurs entières de x . L'approche de Meer *et al.* est donc très semblable à l'approche qui consiste à tronquer le filtre idéal.

4.3.2 La pyramide de Burt

Dans l'approche de Burt, les poids du filtre spatial sont déterminés uniquement par des contraintes spatiales [21]. Outre la séparabilité, Burt établit quatre contraintes qu'un filtre doit respecter. Il s'agit des contraintes de *normalisation*, de *symétrie*, d'*équicontribution* et d'*unimodalité*.

La contrainte de normalisation impose que toute valeur calculée varie dans le même intervalle que celui des valeurs initiales de l'image :

$$\sum_{i=0}^{K-1} w(i) = 1 \quad (4.27)$$

La contrainte de symétrie assure que les éléments situés à une même distance du centre de $h(x, y)$ interviennent avec un poids identique dans le calcul. Ainsi, il n'y a pas de direction privilégiée :

$$w(x) = w(K - 1 - x) \quad 0 \leq x < K \quad (4.28)$$

La contrainte d'équicontribution intervient pour que chaque pixel du niveau courant contribue avec le même poids dans la construction du niveau supérieur :

$$\sum_{i=0}^{K-1} w(m + 2i) = \frac{1}{2} \quad 0 \leq m \leq 1 \quad (4.29)$$

La contrainte d'unimodalité impose que le poids d'un élément soit d'autant plus fort qu'il est proche du centre de $h(x, y)$. Si de plus, tous les poids sont positifs, alors

les variations de l'image initiale sont approximativement conservées. Il s'agit d'une condition pour éviter de créer de nouvelles structures (contours) lors de la réduction [37], [86]. Cette contrainte s'énonce de la manière suivante :

$$0 \leq w(x) \leq w(y) \quad x \leq y < \frac{K}{2} \quad (4.30)$$

Pour effectuer un filtrage rapide et pouvoir construire rapidement une pyramide, Burt préconise $K = 4$ ou $K = 5$. En respectant les contraintes de normalisation, de symétrie et d'équicontribution, on obtient les poids suivants pour le filtre pair :

$$w(1) = w(2) = \alpha \quad , \quad w(0) = w(3) = \frac{1}{2} - \alpha \quad (4.31)$$

De même, on obtient les poids suivants pour le filtre impair :

$$w(2) = \beta \quad , \quad w(1) = w(3) = \frac{1}{4} \quad , \quad w(0) = w(4) = \frac{1}{4} - \frac{\beta}{2} \quad (4.32)$$

La forme finale du filtre dépend du paramètre α lorsque $K = 4$ ou du paramètre β lorsque $K = 5$. Le respect de la contrainte d'unimodalité permet d'encadrer ces paramètres par les valeurs suivantes :

$$\frac{1}{4} \leq \alpha, \beta \leq \frac{1}{2} \quad (4.33)$$

Burt détermine les paramètres optimaux α et β pour que la convolution itérée du filtre converge vers une gaussienne³. Il trouve $\alpha = 0.37$ et $\beta = 0.4$. La figure 4.12(a) illustre le filtre impair ($K = 5$) pour différentes valeurs de β .

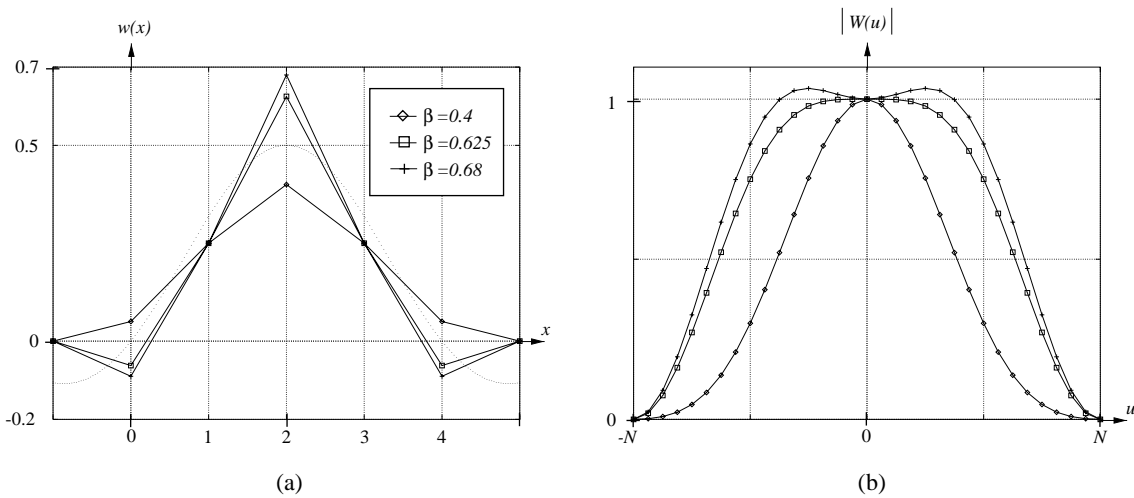


Figure 4.12 – (a) Filtrés obtenus pour différentes valeurs de β ($K = 5$). (b) Spectres associés.

3. En raison de cette propriété du filtre, la pyramide de Burt est généralement appelée *pyramide gaussienne*.

Meer *et al.* ont étudié les effets des paramètres α et β dans le domaine fréquentiel [94]. Pour cela, il faut calculer la réponse en fréquence des filtres en fonction du paramètre α ou β . Si l'image est de taille $2N \times 2N$ pixels, on obtient pour le filtre pair :

$$W(u) = 2\alpha \cos \frac{\pi u}{2N} + (1 - 2\alpha) \cos \frac{3\pi u}{2N} \quad (4.34)$$

De même, la réponse en fréquence pour le filtre impair est donnée par :

$$W(u) = \beta + \frac{1}{2} \cos \frac{\pi u}{N} + \left(\frac{1}{2} - \beta\right) \cos \frac{2\pi u}{N} \quad (4.35)$$

Comme le montre la figure 4.12(b), Meer *et al.* ont constaté à partir de mesures spectrales que, pour $K = 5$, les valeurs de α ou β préconisées par Burt permettent d'obtenir un filtre ayant de bonnes caractéristiques d'antialiasing, c'est-à-dire que les hautes fréquences sont suffisamment atténuées pour éviter un recouvrement spectral. Cependant, le filtre atténue aussi de manière importante les moyennes et basses fréquences. Le filtrage passe-bas est donc loin d'être optimal et génère un effet de flou gênant pour les post-traitements. Pour améliorer la qualité fréquentielle du filtrage, Meer *et al.* suggèrent d'assouplir la contrainte d'unimodalité en autorisant les poids négatifs. Cette contrainte, exprimée par (4.30), est reformulée par :

$$|w(x)| \leq |w(y)| \quad x \leq y < \frac{K}{2} \quad (4.36)$$

Ceci revient en fait à supprimer la borne supérieure de (4.33). Dans ce cas, les poids $w(0)$ et $w(3)$ pour le filtre pair et les poids $w(0)$ et $w(4)$ pour le filtre impair deviennent négatifs lorsque $\alpha, \beta > 1/2$. D'après Meer *et al.*, le filtrage passe-bas est théoriquement optimal pour $\alpha = 0.58$ et $\beta = 0.68$. Cependant, comme le montre la figure 4.12(b) pour $\beta = 0.68$, le filtre offre de moins bonnes caractéristiques d'antialiasing et les basses fréquences sont amplifiées (coefficients supérieurs à 1) au lieu d'être seulement préservées. Pour éviter ce problème, nous proposons d'ajouter des contraintes sur le spectre du filtre spatial. Il faut que celui-ci soit unimodal et que les coefficients soient compris dans l'intervalle $[0, 1]$:

$$0 \leq W(v) \leq W(u) \leq 1 \quad 0 \leq u \leq v < N \quad (4.37)$$

Pour respecter cette contrainte fréquentielle, on trouve de manière empirique que α et β doivent être compris dans les intervalles suivants :

$$0.375 \leq \alpha \leq 0.562 \quad 0.375 \leq \beta \leq 0.625 \quad (4.38)$$

Lorsque α et β sont tous deux inférieurs à 0.375, certains coefficients associés aux hautes fréquences du spectre $W(u)$ deviennent négatifs. Lorsque $\alpha > 0.562$ et $\beta > 0.625$, certains coefficients associés aux basses fréquences deviennent supérieurs à 1. Pour la borne supérieure $\beta = 0.625$, on peut observer sur la figure 4.12(a) que les coefficients $w(0)$ et $w(4)$ sont toujours négatifs. Cependant, la figure 4.12(b) montre que le spectre respecte bien les contraintes données par (4.37).

Pour finir, on peut remarquer que les valeurs préconisées par Burt ($\alpha = 0.37$ et $\beta = 0.4$) sont comprises dans les intervalles définis par (4.38). Cependant, la borne inférieure dans (4.33) est trop faible puisqu'elle est en dehors de ces intervalles.

4.3.3 La pyramide de Chin

Chin *et al.* considèrent la réduction d'une image avant tout comme un problème de lissage d'une surface [38]. Un signal numérique de N pixels peut-être interpolé en ajustant $N - 1$ morceaux de polynômes. La réduction du signal consiste alors à diminuer le nombre de morceaux d'ajustement tout en minimisant un certain critère d'erreur. A partir de cette approche, Chin *et al.* déterminent le filtre équivalent en fonction de l'ordre du polynôme d'ajustement. Dans la suite, nous étudions le cas simple où les raccords se font par des segments linéaires (polynôme d'ordre 1).

Le critère d'erreur à minimiser lors de la réduction correspond à la distance qui sépare le signal original du signal réduit. Chin *et al.* utilisent la métrique classique induite par la norme ℓ_2 . Ainsi, l'erreur entre le signal original f de $2N + 1$ pixels et le signal réduit \bar{f} est donnée par :

$$E = \|f - \bar{f}\|^2 = \sum_{i=0}^{2N} (f(x_i) - \bar{f}(x_i))^2 = \sum_{i=0}^{2N} (y_i - \bar{y}_i)^2 \quad (4.39)$$

Dans le cas où le polynôme d'ajustement est linéaire, le signal réduit est défini par :

$$0 \leq i \leq 2N \quad \begin{cases} \bar{f}(x_i) = \bar{y}_i & \text{lorsque } i \text{ est pair} \\ \bar{f}(x_i) = \frac{1}{2}(\bar{y}_{i-1} + \bar{y}_{i+1}) & \text{sinon} \end{cases} \quad (4.40)$$

En faisant une simple substitution dans (4.39), on obtient :

$$E = \sum_{i=0}^N (y_{2i} - \bar{y}_{2i})^2 + \sum_{i=1}^N \left[y_{2i-1} - \frac{1}{2}(\bar{y}_{2i-2} + \bar{y}_{2i}) \right]^2 \quad (4.41)$$

Le signal réduit qui minimise cette erreur annule également les dérivées partielles :

$$\frac{\partial E}{\partial \bar{y}_{2i}} = \begin{cases} 2y_0 + y_1 - \frac{5}{2}\bar{y}_0 - \frac{1}{2}\bar{y}_2 = 0 & i = 0 \\ y_{2i-1} + 2y_{2i} + y_{2i+1} - \frac{1}{2}\bar{y}_{2i-2} - 3\bar{y}_{2i} - \frac{1}{2}\bar{y}_{2i+2} = 0 & 0 < i < N \\ y_{2N-1} + 2y_{2N} - \frac{1}{2}\bar{y}_{2N-2} - \frac{5}{2}\bar{y}_{2N} = 0 & i = N \end{cases} \quad (4.42)$$

Ce système d'équations peut se noter sous la forme matricielle $AY = B\bar{Y}$, où A et B sont des matrices de tailles respectives $(N + 1) \times (N + 1)$ et $(N + 1) \times (2N + 1)$ définies par :

$$A = \begin{bmatrix} 4 & 2 & & & 0 \\ & 2 & 4 & 2 & \\ & & & \dots & \\ & & & & 2 & 4 & 2 \\ 0 & & & & & & 2 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 5 & 1 & & & 0 \\ 1 & 6 & 1 & & \\ & & \dots & & \\ & & & 1 & 6 & 1 \\ 0 & & & & 1 & 5 \end{bmatrix} \quad (4.43)$$

Pour obtenir le signal réduit qui minimise l'erreur E , il suffit de calculer $\bar{Y} = B^{-1}AY$. En supposant que N est pair et qu'il tend vers l'infini, on peut effectuer ce calcul

en convoluant Y par la $(N/2 + 1)$ ième ligne de $B^{-1}A$. Cette ligne définit le filtre 1D à utiliser pour réduire le signal original. Finalement, Chin *et al.* obtiennent une formulation simple pour calculer les poids de ce filtre :

$$0 \leq i \leq \frac{N}{2} \quad \begin{cases} w(2i) = w(-2i) = \frac{1}{\gamma^i \sqrt{2}} \\ w(2i + 1) = w(-2i - 1) = \frac{1}{2}(w(2i) + w(2i + 2)) \end{cases} \quad (4.44)$$

où γ est défini par $\gamma = -(3 + 2\sqrt{2})$. Pour des raisons de simplicité, le filtre est défini par un centrage sur 0. En pratique, les coefficients du filtre décroissent assez rapidement, et l'on peut donc prendre une largeur de filtre relativement petite. La figure 4.13(a) illustre le filtre linéaire obtenu pour une largeur $K = 13$. On peut relever quelques différences avec le filtre sinus cardinal illustré en pointillé. Le poids central est notamment beaucoup plus élevé et les poids aux extrémités sont plus faibles. Le spectre montré sur la figure 4.13(b) révèle des informations intéressantes sur les propriétés du filtre linéaire. Tout d'abord, on peut remarquer que la bande passante n'atténue pas du tout les basses fréquences, mais les amplifie. On remarque également que la fonction $W(u)$ passe exactement par 1 lorsque u est égale à la fréquence de coupure ($u = N/2$), puis décroît. Il y a donc une atténuation des hautes fréquences dans la bande stoppante. D'après les caractéristiques fréquentielles du filtre, on peut supposer que le filtre rehausse les contrastes des moyennes fréquences.

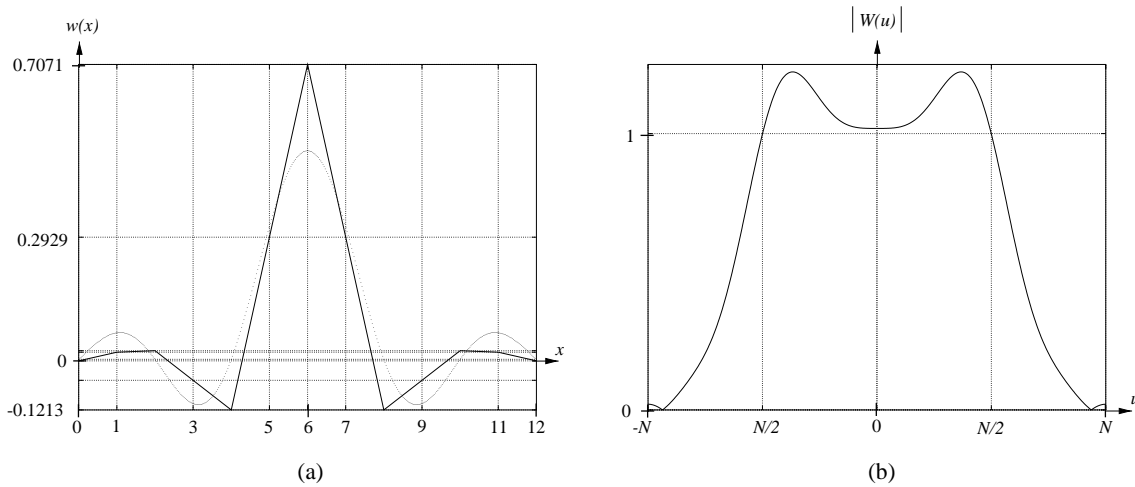


Figure 4.13 – (a) Filtre linéaire ($K = 13$). (b) Spectre associé.

La figure 4.14(a) illustre le filtre obtenu (d'après [38]) avec un polynôme d'ajustement cubique et une largeur de filtre $K = 15$. Par rapport au filtre linéaire, on peut remarquer que les poids du filtre cubique sont plus proches de la fonction sinus cardinal. Le spectre sur la figure 4.14(b) montre d'ailleurs que les hautes fréquences sont davantage atténuées dans la bande stoppante. On remarque que l'amplification des basses fréquences est toujours présente dans la bande passante, avec toutefois un déplacement notable vers les moyennes fréquences. On observe une légère atténuation des basses fréquences autour de $u = 0$. On peut supposer qu'en augmentant le

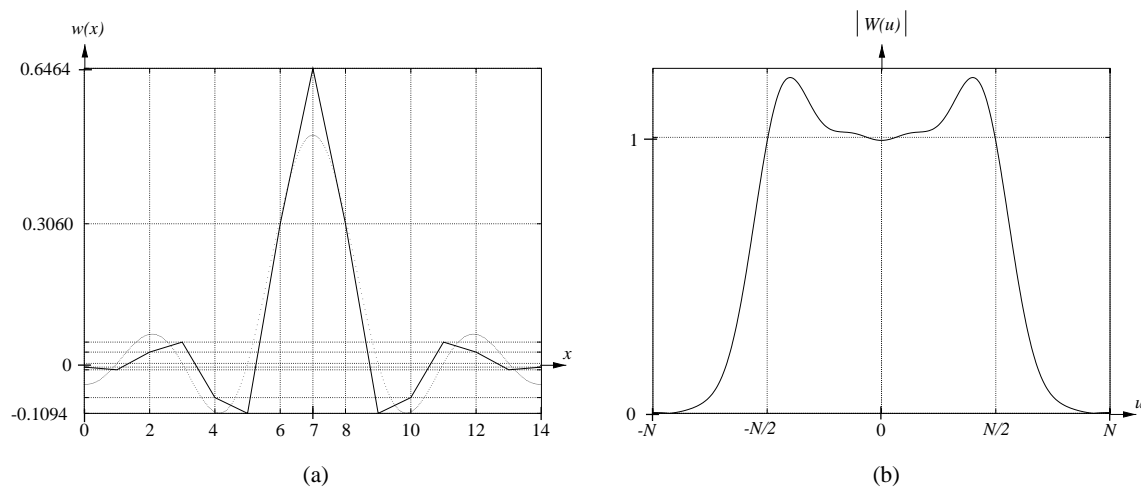


Figure 4.14 – (a) Filtre cubique ($K = 15$). (b) Spectre associé.

degré du polynôme d'ajustement, le filtre correspondant se rapproche du filtre idéal sinus cardinal.

4.3.4 Comparaison expérimentale

La figure 4.15 illustre les quatre niveaux supérieurs des pyramides de Meer, Chin et Burt obtenues à partir de l'image du cameraman représentée sur la figure 4.2. La pyramide située en haut est la pyramide de référence obtenue en itérant la réduction fréquentielle idéale que nous avons étudiée dans la section 4.2.2. On peut remarquer que l'effet d'ondulation (phénomène de Gibbs) est de plus en plus significatif au fur et à mesure que la résolution décroît. De plus, on relève des erreurs en bordure des différents niveaux qui sont dues à la périodisation implicite de l'image avec la transformée de Fourier discrète. Cette pyramide que l'on peut qualifier de *pyramide idéale* d'un point de vue fréquentiel a été développée pour la première fois par Watson [146].

La pyramide de Meer est une amélioration de cette dernière. En procédant uniquement dans le domaine spatial par convolution avec un filtre à petit support (13×13 pixels), l'approche de Meer *et al.* limite l'étendue des ondulations et supprime les erreurs sur les bordures des niveaux. La pyramide de Chin, que ce soit avec le filtre linéaire ou cubique, génère des structures assez semblables à celles de la pyramide de Meer sur les différents niveaux. Comme nous l'avons déjà précisé en analysant les spectres fréquentiels de ces deux filtres sur les figures 4.13(b) et 4.14(b), on observe un rehaussement des contrastes, surtout avec le filtre linéaire.

On remarque que dans les approches de Meer et de Chin, l'effet d'ondulation subsiste dans le voisinage proche des contours. Ce problème provient du fait que la largeur des filtres utilisés n'est pas suffisamment petite. Avec des filtres de largeur $K = 4$ et $K = 5$, l'approche de Burt supprime complètement les ondulations. Avec le filtre pair, on remarque un déphasage de l'information dans la direction haut-gauche. Il est particulièrement significatif sur le niveau 4. Avec le filtre impair, on



Figure 4.15 – Illustration de différentes pyramides à facteur de réduction 2. Nous montrons les quatre niveaux supérieurs des pyramides obtenues à partir de l'image originale du cameraman de 128×128 pixels illustrée sur la figure 4.2. Pour faciliter l'analyse visuelle, les niveaux 2, 3 et 4 ont été redimensionnés à l'échelle du niveau 1.

évite ce déphasage. Lorsque $\beta = 0.375$ le filtre approche une gaussienne et atténue fortement les moyennes et basses fréquences (cf. figure 4.12), ce qui se traduit par un flou visuel sur tous les niveaux de la pyramide. Lorsque $\beta = 0.625$, le filtre préserve la netteté des niveaux et la pyramide obtenue est relativement proche de celle de Meer, l'effet d'ondulation en moins.

Tant du point de vue de la complexité algorithmique que de la qualité visuelle de la pyramide générée, l'approche de Burt avec $K = 5$ et $\beta = 0.625$ semble être la plus intéressante. Dans la suite, nous l'utiliserons comme approche de référence pour réduire une image ou construire une pyramide passe-bas avec un facteur de réduction 2.

4.3.5 Pyramide de différences

Pour obtenir chaque niveau d'une pyramide passe-bas, on effectue successivement une opération de filtrage (décimation fréquentielle) et une opération de sous-échantillonnage (décimation spatiale). Le filtrage retient les hautes fréquences du niveau courant pour ne laisser passer que les basses fréquences. En relevant à chaque niveau de la pyramide les hautes fréquences supprimées par l'opération de filtrage, on construit une pyramide qui traduit une décomposition de l'image originale en bandes de fréquences successives. On peut voir cette pyramide comme une pyramide où les niveaux expriment les différences entre deux niveaux successifs de la pyramide passe-bas.

Ce type de pyramide a été étudié initialement par Crowley *et al.* dans le contexte de la vision artificielle [44], et par Burt *et al.* dans le contexte de la compression d'images [22]. Dans l'algorithme de Crowley *et al.*, le filtre utilisé est un filtre binomial qui approche une distribution gaussienne. La décomposition en bandes de fréquences est obtenue en effectuant une différence entre chaque niveau de la pyramide passe-bas et sa version filtrée. Dans l'algorithme de Burt *et al.*, le filtre utilisé est le filtre de taille 5×5 que nous avons présenté dans la section 4.3.2. Chaque niveau de la pyramide de différences, appelée *pyramide laplacienne*⁴ par ses auteurs, est ici obtenu en effectuant la différence entre deux niveaux successifs de la pyramide passe-bas. Pour que les deux niveaux aient la même taille, le niveau inférieur est interpolé avec le même filtre utilisé pour construire la pyramide passe-bas. La décomposition en bandes de fréquences est moins bonne que dans l'algorithme de Crowley *et al.*, puisque chaque bande est obtenue par une différence entre un niveau de la pyramide et sa version filtrée, non pas une fois, mais deux fois. Dans la pyramide laplacienne, il y a donc beaucoup plus de corrélation entre les niveaux que dans la pyramide de différences de Crowley *et al.* Comme nous le verrons dans le chapitre 7, l'approche de Burt *et al.* se justifie dans un contexte de compression et de transmission progressive des images.

Dans cette section, nous sommes plutôt intéressés par l'algorithme de Crowley

4. Cette appellation provient du filtre utilisé par Burt *et al.* qui approxime une gaussienne. Marr et Hildreth ont montré que le laplacien d'une gaussienne était la limite d'une différence de deux gaussiennes [93].

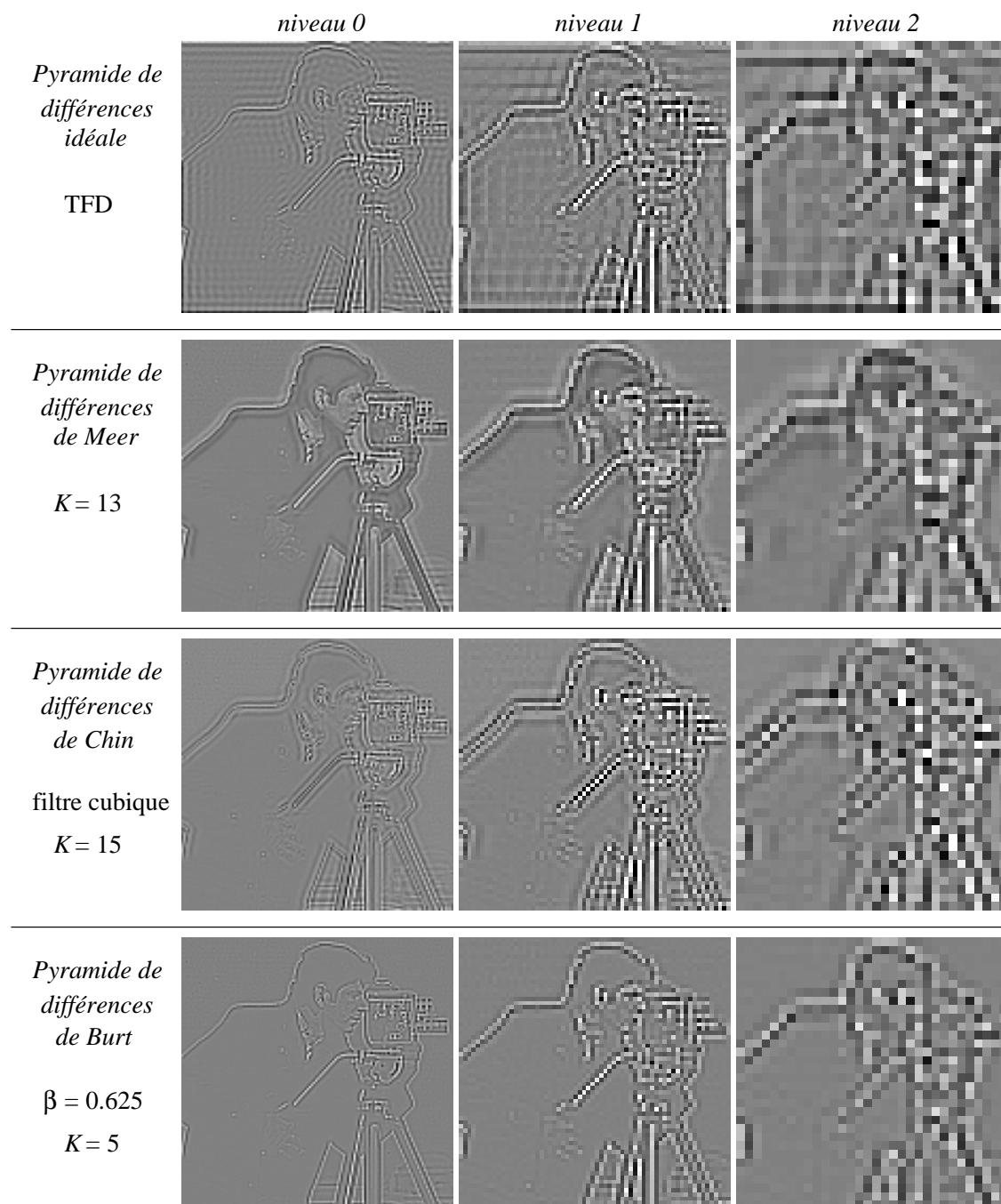


Figure 4.16 – Illustration de différentes pyramides de différences (sur trois niveaux) à facteur de réduction 2. L'image originale est celle du cameraman de 128×128 pixels illustrée sur la figure 4.2. Pour faciliter l'analyse visuelle, les niveaux 1 et 2 ont été redimensionnés à l'échelle du niveau 0 (base de la pyramide), et les niveaux de gris sont amplifiés d'un facteur 2.

et al. qui permet d'obtenir une meilleure décomposition fréquentielle de l'image originale. Cependant, nous généralisons leur approche en n'imposant pas de filtre particulier. Pour construire la pyramide de différences, on utilise le même filtre que celui utilisé dans la construction de la pyramide passe-bas. Ainsi, la construction de la pyramide passe-bas et celle de la pyramide de différences peuvent se résumer de la manière suivante :

$$(k \geq 0) \quad \begin{cases} p_{k+1} = (w * p_k) \downarrow 2 \\ d_k = p_k - w * p_k \end{cases} \quad (4.45)$$

où w est le filtre passe-bas, p_k représente le k -ième niveau de la pyramide passe-bas, sachant que p_0 représente la base de la pyramide, c'est-à-dire l'image originale. Le niveau d_k de la pyramide de différences est donc obtenu en effectuant la différence entre le niveau de la pyramide passe-bas p_k et sa version filtrée par w . Dans le domaine fréquentiel, cette opération se traduit par :

$$(k \geq 0) \quad D_k = P_k \cdot (1 - W) \quad (4.46)$$

Avec cette dernière expression, on voit assez facilement que lorsque le filtre W est un filtre passe-bas idéal, $1 - W$ est un filtre passe-haut idéal. Le niveau d_k correspond donc aux hautes fréquences du niveau p_k . En d'autres termes, on vérifie que la pyramide de différences exprime une décomposition en bandes de fréquences de l'image originale et qu'en théorie, la décomposition fréquentielle est parfaite lorsque le filtre W est un filtre passe-bas idéal. Ce cas particulier a été étudié par Watson pour construire une pyramide de différences idéale [146]. Evidemment, l'effet de Gibbs produit des ondulations sur toute la pyramide, ce qui limite fortement l'utilisation de celle-ci.

La figure 4.16 illustre les pyramides de différences obtenues à partir des pyramides de Meer, Chin et Burt et des filtres associés. La pyramide située en haut est la pyramide de différences de référence obtenue à partir de la pyramide idéale où le filtre passe-bas w est le filtre fréquentiel idéal (cf. section 4.2.2). En d'autres termes, il s'agit de la pyramide de différences étudiée par Watson que nous avons mentionnée précédemment. La pyramide de différences de Burt est la seule qui soit vraiment exempte de l'effet de Gibbs. De plus, la décomposition est relativement proche de la pyramide de différences idéale, l'effet d'ondulation en moins.

4.4 Pyramide en quinconce

Pour construire leur pyramide de différences, Crowley *et al.* exploitent un filtre passe-bas gaussien réalisant une réduction de facteur $\sqrt{2}$ (cf. section 4.3.5). Ce même filtre pourrait être exploité pour construire une pyramide passe-bas à facteur de réduction $\sqrt{2}$. Kropatsch a également étudié la construction d'une telle pyramide [80].

Toutefois, son approche repose avant tout sur des considérations d'ordre géométrique. L'approche que nous développons dans cette section est basée à la fois sur des considérations spatiales et sur des considérations fréquentielles.

En itérant l'emploi des fonctions de décimations fréquentielle et spatiale (développées dans les sections 4.2.3 et 4.2.4), on peut générer une succession d'images à résolution décroissante avec un facteur de réduction $\sqrt{2}$. Cependant, l'analyse fréquentielle a montré que les fonctions de décimation fréquentielle utilisées étaient trop brutales et introduisaient un effet d'ondulation nuisible dans la réduction. Dans le contexte d'une pyramide, ces effets sont encore plus gênants puisqu'ils sont transmis et amplifiés niveau après niveau, et peuvent engendrer de «fausses structures». Un autre inconvénient est que l'utilisation répétée des transformées de Fourier inverses pour construire chaque niveau de la pyramide augmente considérablement la complexité algorithmique. Pour résoudre ces deux problèmes à la fois, on peut utiliser un filtre spatial à support réduit comme nous l'avons vu dans la section précédente pour les pyramides à facteur de réduction 2. Pour construire une pyramide en quinconce, nous élaborons dans la suite un filtre spatial paramétré en nous inspirant de l'approche de Burt que nous avons étudiée dans la section 4.3.2.

4.4.1 Construction d'un filtre paramétré

La taille et le support du filtre déterminent directement l'efficacité de celui-ci. Son support doit être suffisamment petit pour que la convolution soit rapide, mais assez grand pour intégrer l'information nécessaire. Pour la réduction du maillage carré au maillage en quinconce, un support de taille 5×3 (*horizontal* \times *diagonal*) illustré sur la figure 4.17(a) satisfait à ces deux contraintes. Pour la réduction du maillage en quinconce au maillage carré, il est avantageux d'utiliser le même support pivoté de 45 degrés, celui-ci étant ainsi de taille 3×5 comme le montre la figure 4.17(b). De cette manière, il est possible d'employer un filtre unique dans les deux

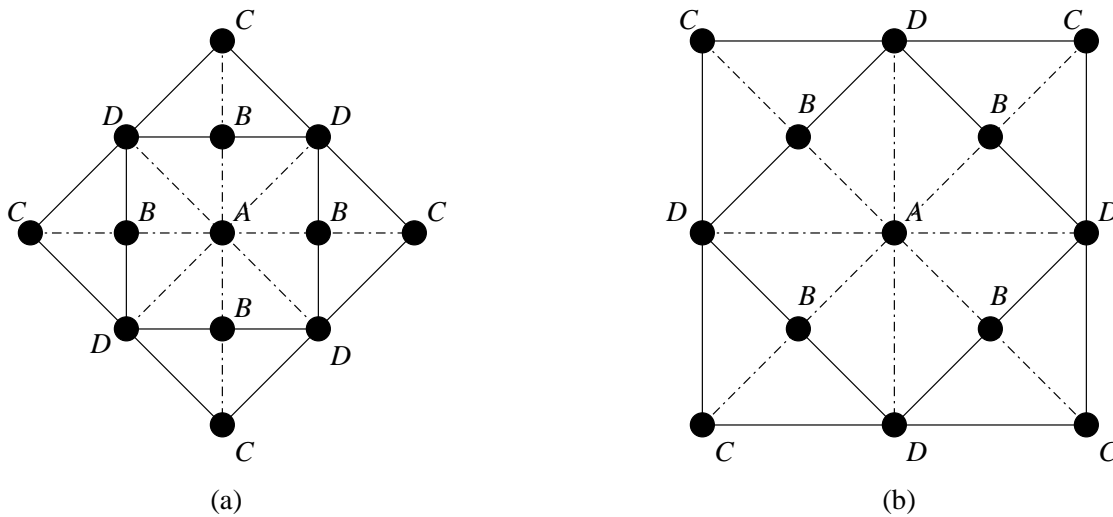


Figure 4.17 – (a) Support du filtre paramétré. (b) Rotation de 45° du support.

types de réduction. Pour établir les 13 poids de ce filtre, on utilise les contraintes spatiales de Burt [21] : symétrie, normalisation, équicontribution et unimodalité. La contrainte de symétrie permet de fixer le nombre de variables à 4 : A , B , C et D (cf. figure 4.17). Les contraintes de normalisation et d'équicontribution établissent respectivement les deux équations suivantes :

$$\begin{cases} A + 4B + 4C + 4D = 1 \\ A + 4C + 4D = 4B \end{cases} \quad (4.47)$$

On en déduit que $B = \frac{1}{8}$, et que la variable C est reliée aux variables A et D par la relation suivante :

$$C = \frac{1}{8} - \frac{A}{4} - D \quad (4.48)$$

Le filtre se définit par conséquent à partir des deux variables A et D . Pour simplifier son analyse, on désire le définir avec une variable unique. Pour cela, A , B , C et D sont redéfinies à partir de 3 variables de la manière suivante :

$$A = a^2 \quad , \quad B = b^2 \quad , \quad C = c^2 \quad , \quad D = ac \quad (4.49)$$

D est donc remplacée par une combinaison des variables A et C . Cette reformulation se justifie dans le cadre de la séparabilité partielle du filtre. En effet, celui-ci peut se redéfinir à partir des deux sous-filtres 1D suivants :

$$S = [c \ 0 \ a \ 0 \ c]^t \quad , \quad T = [0 \ b \ 0 \ b \ 0]^t \quad (4.50)$$

Le filtre R défini par $R = S \cdot S^t + T \cdot T^t$ correspond au filtre 3×5 avec des valeurs nulles intercalées. A partir de la reformulation donnée dans (4.49), la définition du filtre repose sur une seule variable a , sachant que :

$$b = \frac{1}{2\sqrt{2}} \quad , \quad c = \frac{1}{2\sqrt{2}} - \frac{a}{2} \quad (4.51)$$

Pour ne pas créer de fausses structures lors du filtrage, il faut que le filtre soit unimodal. Pour cela, il faut réunir les cinq contraintes suivantes : $a^2 \geq b^2 \geq ac \geq c^2 \geq 0$. On vérifie facilement que :

$$a \geq c \geq 0 \Rightarrow ac \geq c^2 \quad , \quad a \geq b \Rightarrow b > c \quad , \quad (\forall a) \quad b^2 > ac \quad (4.52)$$

La propriété d'unimodalité est donc vérifiée lorsque $a \geq b$ et $c \geq 0$. La condition sur la variable a est alors la suivante :

$$\frac{1}{2\sqrt{2}} \leq a \leq \frac{1}{\sqrt{2}} \quad (4.53)$$

Dans la suite, nous cherchons à définir l'intervalle des valeurs permises de a dans le domaine fréquentiel.

4.4.2 Contraintes fréquentielles

Afin d'exprimer les contraintes fréquentielles sur le filtre, intéressons-nous à sa réponse en fréquence. Ainsi, dans le cas d'une réduction Carré-Quinconce, nous avons :

$$W_{CQ}(u, v) = a^2 + 2b^2 \left(\cos \frac{2\pi u}{N} + \cos \frac{2\pi v}{N} \right) + 2c^2 \left(\cos \frac{4\pi u}{N} + \cos \frac{4\pi v}{N} \right) + 4ac \cos \frac{2\pi u}{N} \cdot \cos \frac{2\pi v}{N} \quad (4.54)$$

Dans le cas d'une réduction Quinconce-Carré, sa réponse en fréquence est donnée par :

$$W_{QC}(u, v) = a^2 + 4b^2 \cos \frac{2\pi u}{N} \cdot \cos \frac{2\pi v}{N} + 4c^2 \cos \frac{4\pi u}{N} \cdot \cos \frac{4\pi v}{N}$$

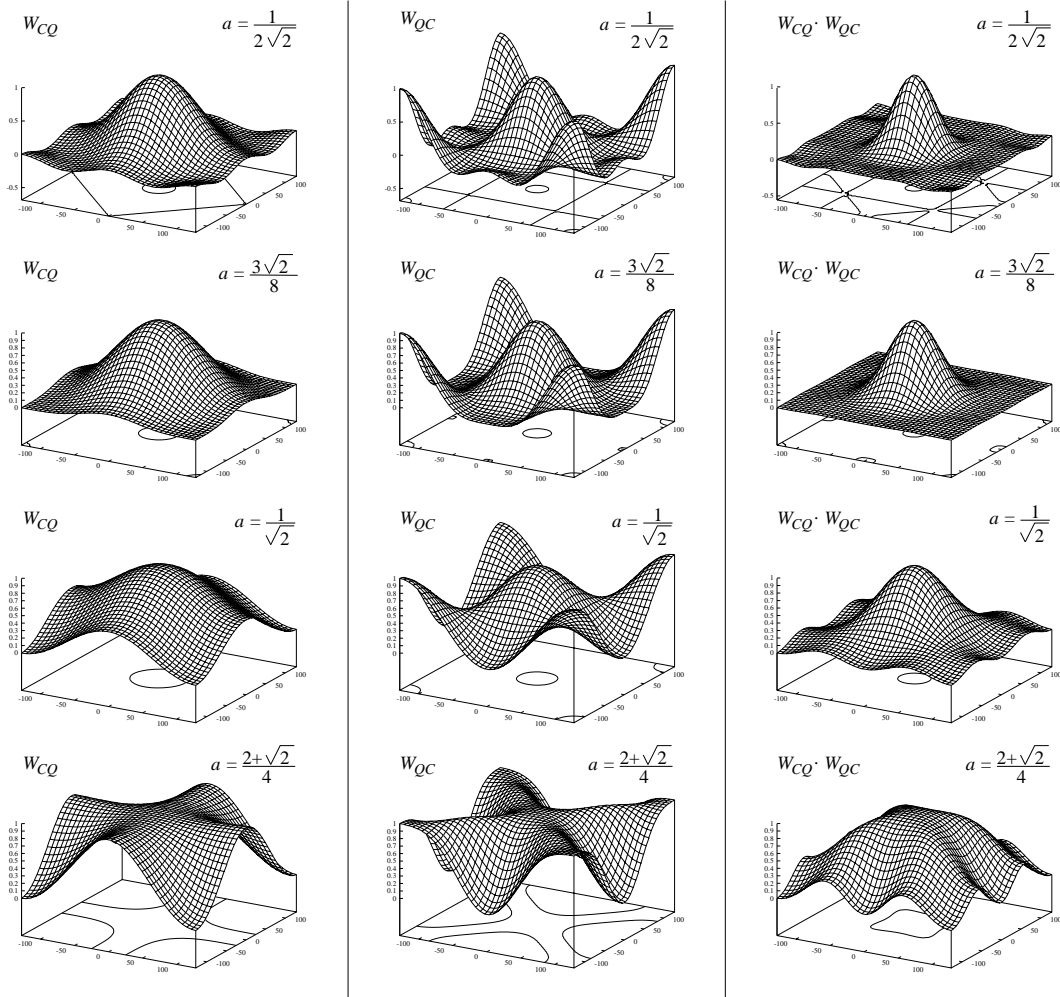


Figure 4.18 – Représentation des fonctions W_{CQ} , W_{QC} et $W_{CQ} \cdot W_{QC}$ pour différentes valeurs du paramètre a .

$$+2ac \left(\cos \frac{4\pi u}{N} + \cos \frac{4\pi v}{N} \right) \quad (4.55)$$

Dans le domaine fréquentiel, il faut que le filtre soit unimodal et que ses coefficients soient compris dans l'intervalle $[0, 1]$. Pour respecter ces deux contraintes, on trouve empiriquement que le paramètre a doit être compris dans l'intervalle suivant :

$$\frac{3\sqrt{2}}{8} \leq a \leq \frac{2 + \sqrt{2}}{4} \quad (4.56)$$

La figure 4.18 illustre les fonctions W_{CQ} , W_{QC} et $W_{CQ} \cdot W_{QC}$ pour des valeurs de a égales aux bornes supérieures et inférieures des intervalles donnés par (4.53) et (4.56). Les lignes projetées sur les bases traduisent les courbes de niveaux pour les valeurs 0 et 0.9 des fonctions. Lorsque $a = 1/2\sqrt{2}$, on observe que la moitié des coefficients sont négatifs. Avec $a > 3\sqrt{2}/8$ tous les coefficients deviennent positifs.

D'un point de vue théorique, la réduction Carré-Quinconce doit laisser intactes les fréquences horizontales et verticales. De même, la réduction Quinconce-Carré doit laisser intactes les fréquences diagonales. Cette propriété du filtre est respectée uniquement lorsque $a = (2 + \sqrt{2})/4$. Lorsque a est inférieur à cette valeur, on a nécessairement une atténuation de ces fréquences. Par contre, le recouvrement spectral devient moins important. En pratique, ce recouvrement dépend évidemment de l'image à filtrer.

4.4.3 Mise en œuvre

La figure 4.19 montre différentes pyramides obtenues à partir de l'image de Carpentras (128×128 pixels) en faisant varier le paramètre a selon trois valeurs : $a = 1/(2\sqrt{2})$, $a = 1/\sqrt{2}$ et $a = (2 + \sqrt{2})/4$. La pyramide située en haut a été obtenue en appliquant itérativement les fonctions de décimation spatiale et de décimation fréquentielle étudiées dans les sections 4.2.3 et 4.2.4. Il s'agit donc d'une pyramide de référence. En comparaison à celle-ci, on remarque que dans les deux pyramides suivantes où $a = 1/(2\sqrt{2})$ et $a = 1/\sqrt{2}$, le filtre retient trop d'informations et les niveaux sont de plus en plus flous. Par contre, la dernière pyramide ($a = (2 + \sqrt{2})/4$) semble donner des résultats très proches de ceux de la pyramide de référence. Cette valeur particulière du paramètre a correspond au cas où le filtre laisse passer idéalement les fréquences horizontales et verticales pour une réduction Carré-Quinconce, et les fréquences diagonales pour une réduction Quinconce-Carré.

4.4.4 Pyramide de différences

On peut construire une pyramide de différences en quinconce suivant le même principe que celui décrit dans la section 4.3.5. Pour cela, on relève les bandes de fréquences supprimées à chaque étape de réduction. De par les propriétés fréquentielles du filtre utilisé, les bandes expriment de manière alternée des fréquences diagonales, puis des fréquences horizontales et verticales de l'image originale. On a donc une décomposition fréquentielle plus progressive que celle obtenue avec la pyramide de différences à facteur de réduction 2.

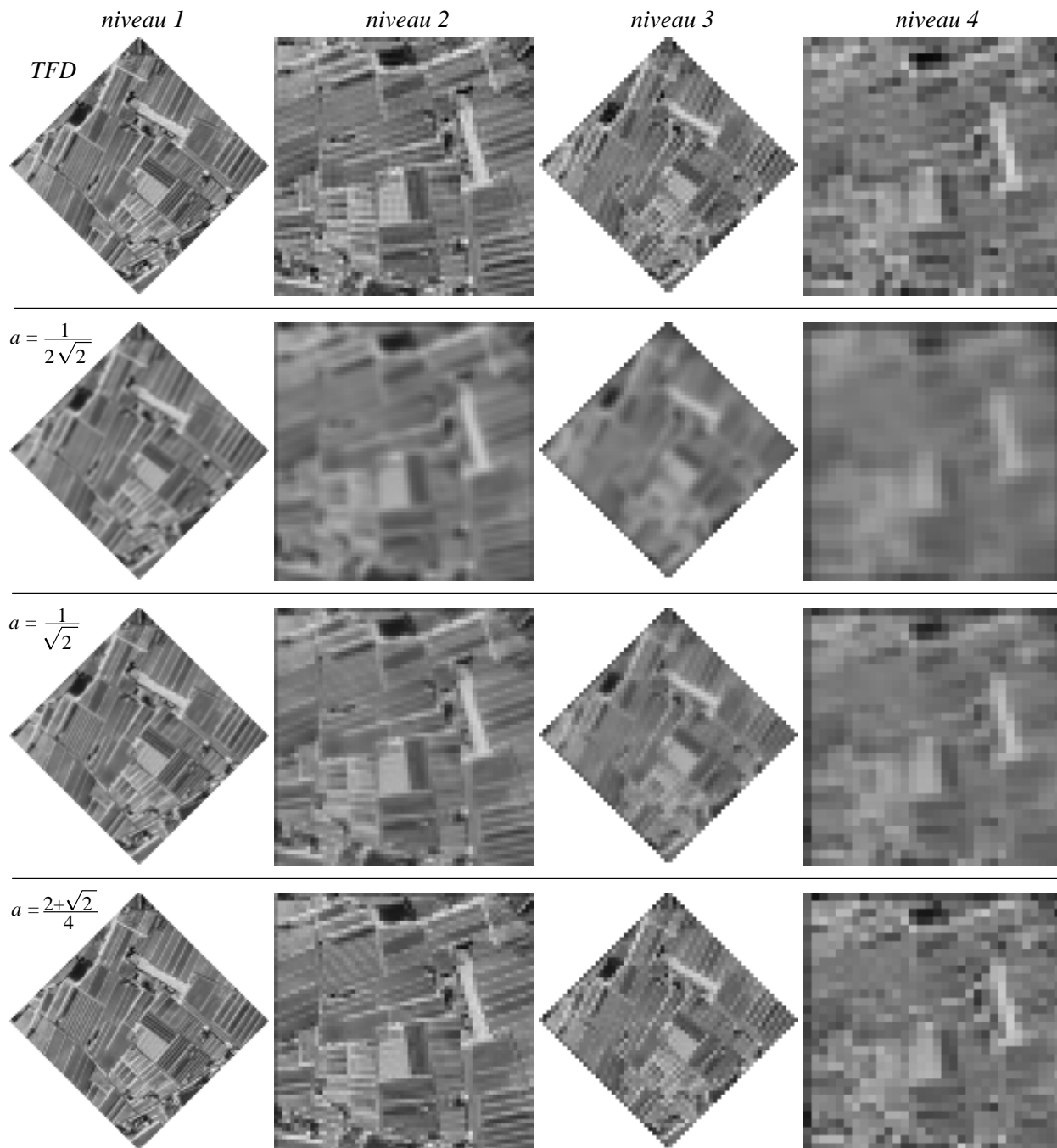


Figure 4.19 – Pyramides à facteur de réduction racine de 2. La pyramide située en haut est la pyramide de référence obtenue par analyse spectrale. Les trois pyramides suivantes ont été obtenues en utilisant le filtre spatial et en faisant varier le paramètre a .

La figure 4.20 illustre deux pyramides de différences en quinconce construites à partir de l'image de Carpentras de 128×128 pixels. La première est la pyramide de différences idéale obtenue avec les filtres fréquentiels idéaux étudiés dans les sections 4.2.3 et 4.2.4. La deuxième pyramide de différences est celle obtenue avec le filtre paramétré que nous avons développé dans la section 4.4.1 en prenant comme paramètre $a = (2 + \sqrt{2})/4$. Cette dernière présente beaucoup moins d'effet d'ondulation que la première. Les niveaux pairs expriment les fréquences diagonales et les niveaux impairs les fréquences horizontales et verticales. On peut voir dans cet exemple particulier que l'image originale de Carpentras est principalement composée de fréquences horizontales et verticales.

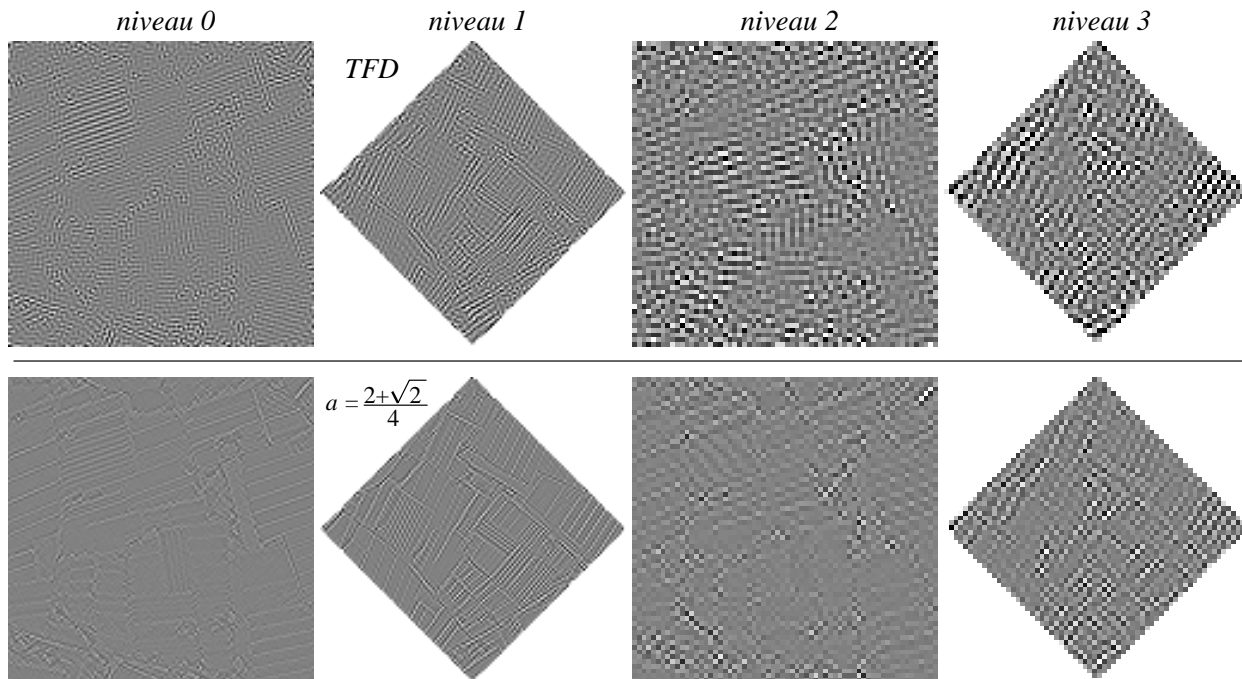


Figure 4.20 – Pyramides de différences à facteur de réduction racine de 2. La pyramide située en haut est la pyramide de référence obtenue par analyse spectrale. La pyramide du bas est obtenue avec le filtre paramétré développé en section 4.4.1. Les niveaux de gris sont amplifiés d'un facteur 4.

4.5 Conclusion

Nous avons étudié dans ce chapitre l'opération de réduction d'une image, ainsi que la construction de pyramides d'images à facteur de réduction 2 ou $\sqrt{2}$. Nous avons tout d'abord analysé l'opération de réduction d'un point de vue fréquentiel. Ceci nous a permis d'identifier clairement les traitements à mettre en œuvre et en particulier les filtres passe-bas idéaux à réaliser pour chaque type de réduction.

A partir de cette analyse théorique, nous avons introduit les travaux de différents

chercheurs ayant pour objectifs de réaliser efficacement l'opération de réduction et de construire rapidement des pyramides passe-bas d'images à facteur de réduction 2. L'approche de Burt *et al.* apparaît comme la plus intéressante tant sur le plan de la qualité que sur celui de la complexité algorithmique. En nous inspirant de cette approche, nous avons proposé une méthode de construction rapide d'une pyramide en quinconce (facteur de réduction $\sqrt{2}$). Enfin, nous avons étudié la construction de pyramides de différences à facteur de réduction 2 et $\sqrt{2}$ pour décomposer l'image originale en bandes fréquentielles.

L'opération de réduction de facteur 2 sera utilisée dans le chapitre 6 comme une contrainte dans le cadre de notre approche de l'agrandissement par induction sur un ensemble. Les pyramides passe-bas et de différences à facteur de réduction 2 seront exploitées dans le chapitre 5 pour la recherche de similarités inter-résolutions. Celles à facteur de réduction $\sqrt{2}$ (pyramides en quinconce) seront utilisées dans le chapitre 7 pour améliorer la méthode du chapitre 5.

Chapitre 5

Agrandissement par synthèse de similarités sur une pyramide

5.1 Introduction

Nous proposons dans ce chapitre une approche d'agrandissement originale qui procède par analyse-synthèse de similarités à partir d'une représentation pyramidale de l'image [26]. Notre approche s'inscrit parmi les méthodes d'agrandissement du chapitre 3 préservant les structures de l'image originale. Le principe général est d'extraire des relations locales de similarité (ressemblances) entre l'image originale et ses résolutions inférieures décrites par les différents niveaux de la pyramide. L'agrandissement est finalement obtenu en effectuant la synthèse de ces relations à une résolution supérieure à celle de l'image originale.

Le modèle d'analyse-synthèse repose sur une définition précise de la notion de similarité. Nous dirons que deux blocs¹ sont similaires s'ils sont reliés par une transformation analytique simple, la nature de cette transformation étant à préciser. De plus, il faut mettre en place une mesure de similarité pour évaluer le degré de similarité entre deux blocs. Ces deux points sont développés dans la section 5.3.

Une fois que la notion de similarité est précisément définie, la principale difficulté du modèle d'analyse réside dans la recherche des similarités. Une recherche par exploration exhaustive est laborieuse du point de vue de la complexité algorithmique. Il faut donc mettre en place des stratégies de recherche efficaces. De plus, on peut s'interroger sur l'intérêt d'une recherche globale par rapport à celui d'une recherche locale à travers les résolutions. Nous étudions ces différents aspects dans les sections 5.4 à 5.6.

Pour extrapoler un niveau inférieur à la base de la pyramide, il faut mettre en place un modèle de synthèse des similarités. Pour un agrandissement de facteur 2, le

1. Dans notre approche, un bloc est un ensemble de pixels formant un carré. Cette forme géométrique permet de simplifier de nombreux traitements.

modèle exploite directement la transformation analytique des relations de similarité ainsi que leur résidu. Pour un agrandissement de facteur supérieur à 2, deux solutions s'offrent à nous. Soit on généralise le modèle de synthèse utilisé pour l'agrandissement de facteur 2, et dans ce cas l'analyse est faite une fois pour toutes, soit on recommence l'analyse à partir de l'agrandissement de facteur 2, c'est-à-dire que l'on effectue à nouveau une recherche de similarités, et on applique le modèle de synthèse qui produit une résolution double. En itérant ce principe, on obtient l'agrandissement souhaité. Le modèle de synthèse et les différentes possibilités d'agrandissement sont étudiés en détail dans la section 5.7.

A partir de résultats expérimentaux, nous montrons dans la section 5.8 que notre approche produit de meilleurs agrandissements que les approches classiques qui préservent uniquement les fréquences de l'image originale. Nous montrons également que l'hypothèse de préservation d'une similarité à une résolution supérieure se vérifie dans la majorité des cas.

Notre approche est liée au zoom fractal (cf. section 3.4) et plus généralement à la compression fractale, par la notion commune de similarité inter-résolutions. Les techniques développées dans ce domaine servent donc naturellement de cadre de référence à notre approche. Cependant, les besoins en compression et en agrandissement étant relativement différents, le modèle que nous développons dans ce chapitre s'en différencie par de nombreux points, que ce soit dans la phase d'analyse ou dans celle de synthèse. Nous les détaillons dans la conclusion de ce chapitre.

5.2 Généralités sur la méthode

Dans cette section, nous présentons le principe de base de notre approche, puis nous décrivons les grandes étapes de l'algorithme qui sont ensuite approfondies dans les sections suivantes.

5.2.1 Le principe de base

L'idée générale de notre modèle d'agrandissement peut se comprendre assez facilement à l'aide de la figure 5.1. Celle-ci représente une pyramide à trois niveaux obtenue à partir de l'image de Lenna. Pour faciliter l'appréciation visuelle, les niveaux supérieurs de la pyramide ont été redimensionnés à la même échelle que la base. Pour les besoins de l'explication du modèle, supposons que l'image centrale soit l'image originale et que l'image du bas soit l'agrandissement que l'on cherche à obtenir. Dans ce cas, les deux images supérieures constituent une pyramide à deux niveaux dont l'image centrale constitue la base.

Intéressons-nous pour le moment au bloc X situé sur cette base et au bloc Y situé au premier niveau (image du haut). Supposons que le bloc Y soit similaire par une transformation analytique simple τ au bloc X . Dans ce cas, celui-ci peut être obtenu en appliquant la transformation τ au bloc Y et en ajoutant un résidu ε , c'est-à-dire $X = \tau(Y) + \varepsilon$. Si la similarité est de bonne qualité, le résidu est faible,

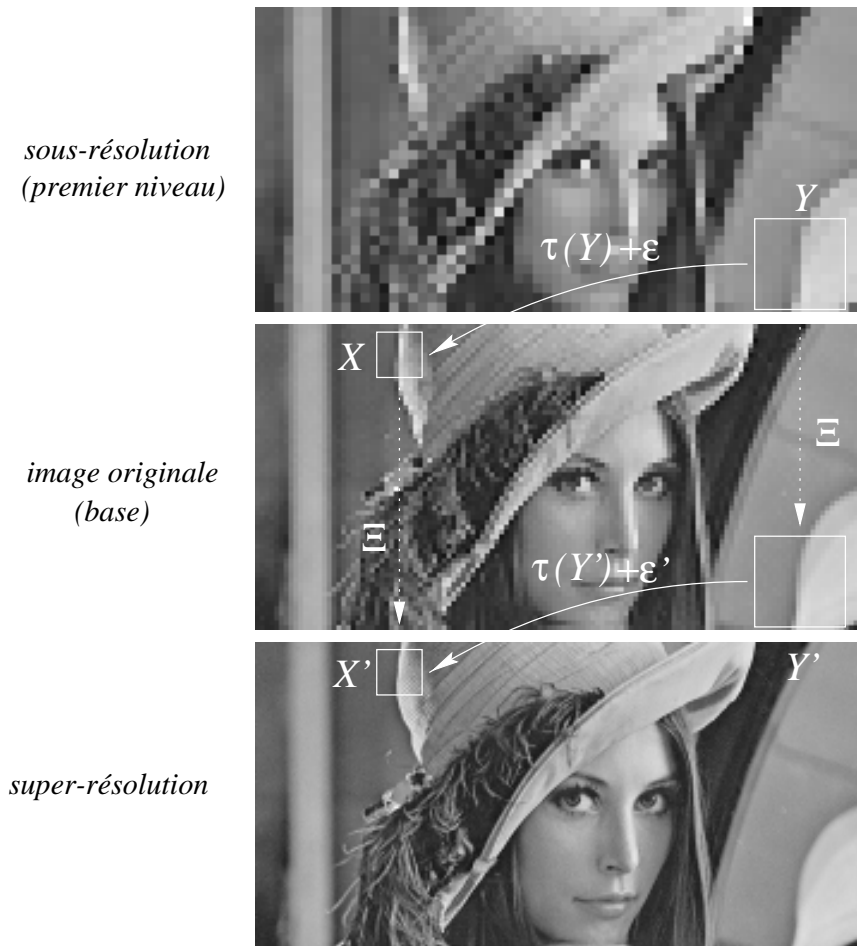


Figure 5.1 – Synthèse d'une similarité à une résolution supérieure.

voire négligeable.

Considérons maintenant les blocs fils X' et Y' situés sur les niveaux immédiatement inférieurs dans la pyramide. Le bloc X' est le bloc que l'on cherche à synthétiser sur le niveau de super-résolution de la pyramide. On peut l'exprimer en fonction du bloc Y' par la relation $X' = \tau(Y') + \varepsilon'$ où ε' est le résidu inconnu de la transformation. En supposant que la relation de similarité entre X et Y est préservée idéalement entre leurs blocs fils, on peut considérer que le résidu ε' est du même ordre que le résidu ε . Dans ce cas, on peut faire l'approximation $X' = \tau(Y') + I(\varepsilon)$ où la fonction I est une fonction d'interpolation qui double la résolution du résidu ε .

En résumé, nous avons synthétisé localement la fonction d'agrandissement Ξ qui permet de passer d'un bloc X à son bloc fils X' , par la suite de relations suivante :

$$X' = \Xi(X) = \Xi(\tau(Y) + \varepsilon) = \tau(Y') + \varepsilon' \approx \tau(Y') + I(\varepsilon) \quad (5.1)$$

En d'autres termes, ne connaissant pas le résultat de la fonction d'agrandissement Ξ pour la configuration du bloc X , nous l'estimons à partir d'un résultat connu $Y' =$

$\Xi(Y)$ situé à une résolution inférieure, et ceci en exploitant la relation de similarité qui existe entre X et Y . Pour obtenir la totalité du niveau de super-résolution avec ce principe, on le partitionne² en blocs élémentaires (X'_i) et on applique l'expression (5.1) à chaque bloc. Evidemment, en pratique, on ne connaît pas les blocs (Y_i) qui sont similaires aux blocs pères (X_i). Toute la difficulté du modèle est en fait de trouver ces similarités sur les niveaux supérieurs de la pyramide.

5.2.2 Algorithme général

Sur la base du principe que nous avons décrit précédemment, nous donnons l'algorithme général du modèle d'agrandissement dans la table 5.1. Chaque étape est détaillée dans la suite.

1. Construire une pyramide.
2. Construire et organiser un domaine de recherche des similarités \mathcal{D} .
3. Partitionner le niveau de super-résolution de la pyramide en blocs (X'_i).
4. Rechercher dans le domaine \mathcal{D} les blocs (Y_i) similaires aux blocs pères (X_i) associés aux blocs (X'_i).
5. Extrapoler le niveau de super-résolution de la pyramide en synthétisant les blocs (X'_i) à partir des relations de similarité trouvées et des blocs fils (Y'_i) associés aux blocs (Y_i).

Table 5.1 – Algorithme général pour obtenir le niveau de super-résolution d'une pyramide.

Etape 1 : La pyramide

La construction de la pyramide est basée sur l'étude réalisée dans le chapitre 4. Pour créer une représentation pyramidale d'une image avec un facteur de réduction de 2, il faut itérer un filtrage passe-bas, un sous-échantillonnage et une quantification à partir de l'image originale. Le filtrage passe-bas permet d'éviter le repliement du spectre causé par l'opération de sous-échantillonnage. Nous avons étudié différents filtres dans la section 4.3 : le filtre 13×13 de Meer, les filtres 4×4 et 5×5 de Burt, et les filtres 13×13 et 15×15 de Chin. La construction de la pyramide repose donc avant tout sur le choix de l'un de ces filtres. Nous renvoyons le lecteur à la section 4.3.4 pour une comparaison expérimentale de ceux-ci.

2. Lorsque les blocs fils et les blocs pères ont le même support, partitionner le niveau de super-résolution revient à partitionner la base de la pyramide. Cependant, nous verrons plus loin que la synthèse des similarités peut être améliorée en choisissant un support plus petit pour les blocs fils, ce qui entraîne un recouvrement des blocs pères. De manière générale, c'est donc bien le niveau de super-résolution qui doit être partitionné et non pas la base de la pyramide.

Remarquons que le modèle d'agrandissement est indépendant du choix du filtre. Cependant, comme nous le verrons dans la suite, la parité de la fenêtre de similarité est liée à celle du filtre utilisé pour générer la pyramide. Un filtre pair (resp. impair) induit le choix d'une fenêtre de similarité de largeur paire (resp. impaire). Cette contrainte permet d'éviter un déphasage local lors de la synthèse de la similarité.

Par rapport au filtre impair, le filtre pair a l'inconvénient d'introduire un déphasage lors de la construction de la pyramide (cf. figure 4.15). Par ailleurs, une fenêtre de similarité de largeur paire simplifie beaucoup la mise en place du modèle d'agrandissement. Dans la suite, le modèle d'agrandissement est développé avec les deux types de parité. Le cas d'une fenêtre de largeur paire a surtout un intérêt «pédagogique», tandis que celui d'une fenêtre de largeur impaire a plutôt un intérêt expérimental.

Outre le filtre, il faut également choisir le nombre de niveaux dans la pyramide. Plus la pyramide possède de niveaux, plus on augmente la possibilité de trouver de bonnes similarités. Cependant, on peut s'interroger sur la pertinence d'une similarité située à une résolution très grossière de l'image originale, surtout si celle-ci est très complexe. Il faut donc limiter le nombre de niveaux de la pyramide en fonction de la complexité de l'image à agrandir.

Etape 2 : Le domaine de recherche

Le domaine de recherche des similarités \mathcal{D} est composé d'un ensemble de blocs extraits de la pyramide. Par souci de simplification, ces blocs sont des blocs carrés et ont tous la même largeur. Celle-ci correspond à la largeur de la fenêtre de similarité choisie. D'un point de vue expérimental, une taille de 4×4 pixels ou de 5×5 pixels selon la parité du filtre, semble la plus appropriée. Avec une largeur de fenêtre trop grande, on diminue la qualité des similarités tandis qu'avec une largeur trop petite, les similarités ne sont plus significatives.

La première condition pour qu'un bloc appartienne au domaine \mathcal{D} est qu'il soit situé sur les niveaux supérieurs de la pyramide, c'est-à-dire sur toute la pyramide sauf la base. Une similarité avec un bloc Y_i qui appartiendrait à la base ne présenterait aucun intérêt puisque son fils Y'_i serait inconnu et la synthèse de similarité au niveau de super-résolution ne pourrait pas être réalisée.

Une autre condition pour que le bloc appartienne au domaine \mathcal{D} est qu'il soit porteur d'une information pertinente. Un bloc homogène ou un bloc composé uniquement de basses fréquences (relativement au niveau de la pyramide auquel il se trouve) ne présente aucun intérêt pour une similarité. Les aspects de sélection des blocs sont traités en section 5.4.

Pour simplifier la recherche des similarités dans l'étape 4, il faut organiser le domaine \mathcal{D} . Une première organisation relativement simple consiste à ordonner les blocs selon leur variance. Une seconde consiste à regrouper les blocs possédant des caractéristiques communes. Dans ce sens, nous développons dans la section 5.5 une approche qui consiste à rassembler les blocs similaires par projection (cette notion est précisée dans la section 5.5.1).

Etape 3 : Le partitionnement

Comme nous l'avons déjà précisé dans l'étape précédente, les blocs pères ont une taille de 4×4 pixels ou de 5×5 pixels selon la parité du filtre choisi pour construire la pyramide. Le support spatial des blocs fils ne peut pas excéder celui des blocs pères, c'est-à-dire qu'ils ont une taille maximale de 8×8 pixels ou de 10×10 pixels. Le support d'un bloc fils doit être centré sur celui du bloc père et l'ensemble des blocs fils doit former une partition du niveau de super-résolution. Pour améliorer la qualité de la super-résolution, on a intérêt à diminuer le support des blocs fils par rapport à celui des blocs pères (ceci est confirmé par les résultats expérimentaux). De cette manière, on obtient des recouvrements partiels entre les blocs pères.

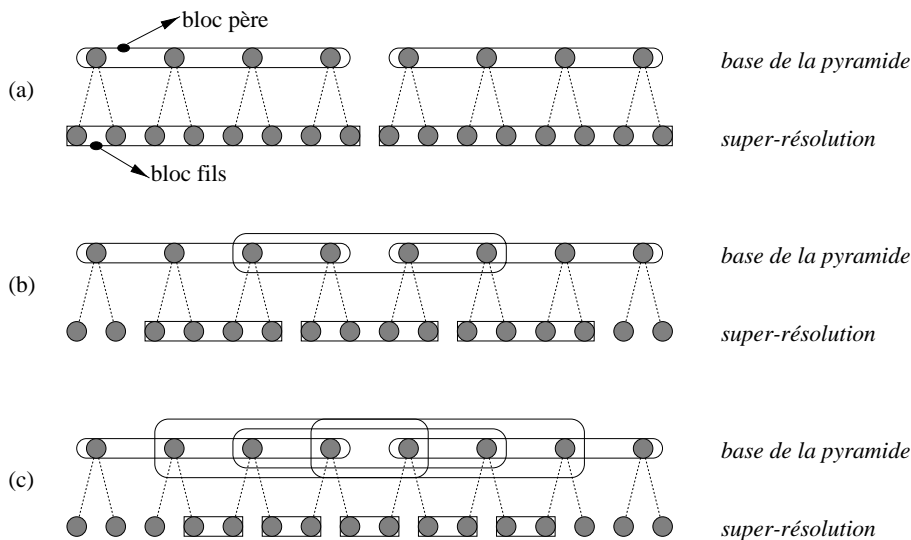


Figure 5.2 – Représentation unidimensionnelle des différents partitionnements du niveau de super-résolution lorsque les blocs pères ont une taille de 4×4 pixels: (a) blocs fils de 8×8 pixels, (b) blocs fils de 4×4 pixels, (c) blocs fils de 2×2 pixels.

Lorsque les blocs pères ont une taille de 4×4 pixels, on définit trois types de partitionnement du niveau de super-résolution associés à trois tailles différentes des blocs fils: 8×8 , 4×4 ou 2×2 pixels. Ces trois partitionnements sont illustrés sur une seule dimension sur la figure 5.2. On voit qu'en diminuant la taille des blocs fils, on augmente en proportion leur nombre pour garantir le pavage du niveau de super-résolution, et on génère des recouvrements entre les blocs pères puisque leur taille demeure constante. La complexité algorithmique augmente avec le nombre de blocs à synthétiser. Avec des blocs fils de taille 4×4 et 2×2 pixels, on a également un problème de pavage sur les bords. En pratique, les pixels concernés seront simplement interpolés par une transformée B-spline (cf. section 2.3.3).

Dans le cas où les blocs pères ont une taille de 5×5 pixels, on définit un seul partitionnement du niveau de super-résolution. Comme le montre la figure 5.3 sur deux dimensions, ce partitionnement est composé de quatre tailles de blocs fils: 5×5 , 5×3 , 3×5 ou 3×3 pixels. Ce partitionnement permet de paver complètement

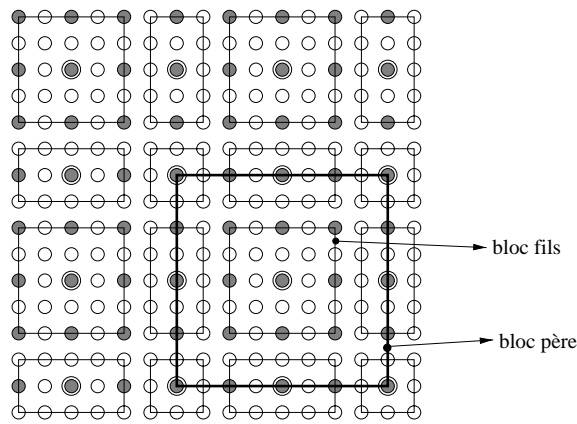


Figure 5.3 – Partitionnement du niveau de super-résolution lorsque les blocs pères ont une taille de 5×5 pixels. Il y a quatre supports différents pour les blocs fils : 5×5 , 5×3 , 3×5 ou 3×3 pixels.

une image dont la largeur est un multiple de huit. Il permet également d’obtenir des recouvrements partiels entre les blocs pères.

Les partitions que nous avons choisies sont relativement simples et rigides, dans le sens où elles ne s’adaptent pas à l’image. Ce choix s’explique au moins par deux raisons. Tout d’abord, lorsqu’il y a des recouvrements entre les blocs pères, on peut difficilement envisager de faire un partitionnement adaptatif du niveau de super-résolution. Ensuite, la partition du niveau de super-résolution est liée à la géométrie des blocs pères situés à la base de la pyramide. En prenant un support à géométrie carrée pour ces derniers, on simplifie la construction et l’organisation du domaine de recherche \mathcal{D} , et les similarités sont donc plus faciles à rechercher.

Étape 4 : La recherche des similarités

La recherche des similarités dans le domaine \mathcal{D} tire parti de la sélection des blocs et de leur organisation faites dans l’étape 2. Pour trouver une similarité d’un bloc X_i , le schéma de recherche le plus évident consiste à parcourir le domaine \mathcal{D} en évaluant le degré de similarité de chaque bloc. Le bloc retenu est le bloc du domaine qui minimise la mesure de similarité au bloc X_i . On peut parler dans ce cas d’une *recherche globale*. Avec ce schéma de recherche, on exploite uniquement la sélection des blocs réalisée dans l’étape 2. Le domaine \mathcal{D} étant en pratique relativement vaste, une recherche globale nécessite de nombreux calculs. Pour accélérer la recherche, on exploite l’organisation du domaine réalisée dans l’étape 2. Comme nous l’avons déjà précisé, le domaine est organisé en classes rassemblant les blocs similaires par projection (voir section 5.5). Chacune de ces classes contient un bloc particulier appelé *le représentant* de la classe. Ainsi, pour exploiter l’organisation du domaine, la recherche d’une similarité ne se fait plus parmi l’ensemble des blocs du domaine \mathcal{D} mais parmi l’ensemble des représentants des classes du domaine. Cette recherche est ensuite poursuivie à l’intérieur de la classe retenue. Avec cette optimisation de la recherche globale, on ne garantit pas que le bloc Y_i soit le bloc le plus similaire

à X_i dans le domaine \mathcal{D} . Cependant, on réalise un bon compromis entre la qualité de la similarité trouvée et la complexité de la recherche globale. Nous étudions en détail l'algorithme de recherche globale dans la section 5.6.2.

Une alternative à la recherche globale d'une similarité du bloc X_i est de faire une recherche sur un certain voisinage autour de l'axe des résolutions centré sur le bloc X_i . Dans ce cas, la recherche d'une similarité dans le domaine \mathcal{D} est contrainte par la localisation du bloc, celui-ci ne devant pas être situé à plus d'une certaine distance du bloc X_i (paramètre de l'algorithme). On peut donc parler d'une *recherche locale* au bloc X_i . Une recherche très locale convient surtout pour les blocs (X_i) qui sont traversés par un contour simple. En effet, il est reconnu que la majorité des contours existe à toutes les résolutions d'une image (cf. section 3.3). Pour un bloc X_i situé sur un contour, il y a donc une forte probabilité de trouver une bonne similarité autour de l'axe des résolutions qui le traverse. L'algorithme d'une recherche locale est approfondi dans la section 5.6.3.

La recherche d'une similarité pour un bloc homogène ou porteur d'une information basse-fréquence ne présente pas d'intérêt. Les conditions pour lancer la recherche d'une similarité sont étudiées dans la section 5.6.1.

Etape 5 : La synthèse des similarités

Les blocs (X'_i) à synthétiser forment une partition du niveau de super-résolution de la pyramide (voir étape 3). Pour un bloc X'_i dont le père X_i est homogène ou porteur d'une information basse-fréquence, on réalise une simple interpolation. Pour les autres blocs X'_i , on réalise une synthèse en exploitant la similarité qui relie le bloc père X_i au bloc Y_i selon le principe que nous avons présenté dans la section 5.2.1. Ce modèle de synthèse est décrit en détail dans la section 5.7.1.

Pour obtenir un agrandissement de facteur supérieur à 2, on peut envisager deux approches. La première consiste à recommencer la phase d'analyse-synthèse en considérant que le niveau de super-résolution est la nouvelle base de la pyramide. En itérant ce principe, on produit toutes les résolutions intermédiaires à l'agrandissement souhaité. A chaque itération, le nombre des blocs (X'_i) est quadruplé et le temps de recherche des similarités des blocs pères est multiplié par quatre. Pour éviter une croissance démesurée de la complexité algorithmique, on ne reconstruit pas le domaine de recherche \mathcal{D} à chaque itération. Autrement dit, les similarités sont toujours recherchées dans les niveaux supérieurs de la pyramide construite à partir de l'image originale. Finalement, cette approche consiste à itérer uniquement les étapes 3 à 5 de l'algorithme défini dans la table 5.1 en doublant à chaque fois la résolution de la base de la pyramide.

La deuxième approche consiste à réutiliser uniquement le modèle de synthèse à partir de la première analyse. Comme pour la première approche, on produit les résolutions intermédiaires à l'agrandissement souhaité, mais le nombre de blocs (X'_i) reste inchangé. En fait, seule la taille des blocs (X'_i) et (Y'_i) double à chaque itération. Les blocs (X'_i) sont synthétisés à partir des blocs (Y'_i) avec les relations de similarité trouvées lors de la première analyse. Le résidu de la transformation analytique est

obtenu en interpolant le résidu initial ε_i avec le facteur d'agrandissement approprié. Cette approche a l'avantage sur la première d'être beaucoup plus rapide, surtout lorsque le facteur d'agrandissement est important. Cependant, comme le montrent les résultats expérimentaux de la section 5.8, elle génère des artefacts dans l'image agrandie. Les deux approches d'agrandissement sont développées dans la section 5.7.2.

5.3 La mesure de similarité entre deux blocs

On dit qu'un bloc Y est *similaire* à un bloc X si en déformant le bloc Y avec une transformation simple sur les niveaux de gris, on arrive à minimiser suffisamment la distance au bloc X . L'avantage de cette définition de la similarité réside dans la possibilité de synthétiser la relation existante entre les deux blocs grâce à la fonction de transformation. On voit que le degré de similarité entre deux blocs va dépendre du choix d'une transformation élémentaire et de celui d'une métrique pour exprimer la distance. En compression par fractale, on utilise une métrique induite par la norme ℓ_2 et la transformation est généralement affine [50]. Le choix de ce type de transformation peut s'expliquer par son rapport avec la réalité physique. On peut en effet relier une transformation linéaire à une modification de la luminosité et du contraste du bloc Y . De plus, dans une optique de compression, le stockage de la transformation est crucial. Le choix d'une transformation linéaire se révèle alors très avantageux puisqu'il n'y a que deux paramètres à stocker : le coefficient d'échelle et le coefficient de décalage. On peut également améliorer le modèle de similarité en augmentant la transformation avec des rotations et des réflexions du bloc Y , ce qui fait un total de huit isométries possibles [72], [73]. Ceci améliore la qualité de l'image compressée, mais affaiblit le taux de compression avec un paramètre supplémentaire à stocker.

Dans le cadre de l'agrandissement d'images, le nombre de paramètres de la transformation n'est pas crucial. Nous pouvons donc généraliser le concept de similarité en utilisant des transformations plus complexes. Dans la suite, nous décrivons un modèle de similarité basé sur des transformations polynomiales. Nous montrons ensuite que les paramètres de la transformation peuvent être obtenus en résolvant un système d'équations. Nous terminons cette section par des exemples de mesures de similarité réalisées avec différents polynômes.

5.3.1 Le modèle de similarité

Nous définissons la mesure de similarité d'un bloc Y à un bloc X de même taille $N \times N$, par la fonction suivante :

$$\chi(X, Y) = \min_{\tau_{(p,q)}, k} \| X - \tau_{(p,q)} \circ \theta_k(Y) \|^2 \quad (5.2)$$

où la fonction $\tau_{(p,q)}$ est une fonction de transformation massique définie par un *polynôme d'échelle* de degré p et un *polynôme fixe* de degré q :

$$\tau_{(p,q)}(Y_{i,j}) = \sum_{n=1}^p a_n Y_{i,j}^n + \sum_{n=0}^q (b_n i^n + c_n j^n) \quad (5.3)$$

Et la fonction θ_k est une fonction de transformation spatiale qui réalise une *isométrie* particulière du bloc selon la valeur de k . Les huit isométries possibles sont définies de la manière suivante :

$$\left\{ \begin{array}{ll} \text{Identité} & \theta_0(Y_{i,j}) = Y_{i,j} \\ \text{Rotation de } -90^\circ \text{ autour du centre} & \theta_1(Y_{i,j}) = Y_{N-1-j,i} \\ \text{Rotation de } -180^\circ \text{ autour du centre} & \theta_2(Y_{i,j}) = Y_{N-1-i,N-1-j} \\ \text{Rotation de } +90^\circ \text{ autour du centre} & \theta_3(Y_{i,j}) = Y_{j,N-1-i} \\ \text{Réflexion autour de l'axe horizontal} & \theta_4(Y_{i,j}) = Y_{N-1-i,j} \\ \text{Réflexion autour de l'axe vertical} & \theta_5(Y_{i,j}) = Y_{i,N-1-j} \\ \text{Réflexion autour de l'axe diagonal } -45^\circ & \theta_6(Y_{i,j}) = Y_{j,i} \\ \text{Réflexion autour de l'axe diagonal } +45^\circ & \theta_7(Y_{i,j}) = Y_{N-1-j,N-1-i} \end{array} \right. \quad (5.4)$$

Dans l'expression (5.3), le polynôme d'échelle agit directement sur les niveaux de gris du bloc Y , alors que le polynôme fixe ajoute un bloc constant plus ou moins complexe selon la valeur de q . Les coefficients (a_n) , (b_n) et (c_n) permettent de contrôler le degré de transformation finale du bloc Y . Par exemple, en prenant $p = 1$, $q = 0$, $a_1 = 1$ et $b_0 = c_0 = 0$, on obtient une transformation identique. En supprimant les contraintes sur les coefficients a_1 , b_0 et c_0 , on obtient une transformation linéaire du bloc Y . En augmentant les valeurs de p et q , on obtient des transformations polynomiales de plus en plus complexes.

Dans la littérature, la transformation linéaire ($p = 1$ et $q = 0$) est le modèle de transformation le plus répandu. L'utilisation d'une transformation plus complexe est rare. On peut toutefois trouver l'expérimentation des polynômes d'échelle quadratique et cubique avec $q = 0$ [88], [150], et celle des polynômes fixe linéaire, quadratique et cubique avec $p = 1$ [49], [62], [96], [97]. Dans le cadre de notre étude, nous généralisons ces approches en faisant varier à la fois p et q avec $p \in \{1, 2, 3\}$ et $q \in \{0, 1, 2\}$. De plus, nous nous affranchissons des contraintes liées à la contraction de la transformation fractale (cf. section 3.4.1). En effet, dans la phase de décodage en compression par fractale, la transformation est itérée sur une image quelconque et il faut assurer la convergence vers un attracteur unique proche de l'image originale. Comme nous le verrons plus loin en détail, la notion d'attracteur est absente de notre approche. La transformation est appliquée une seule fois à une résolution supérieure. Nous n'avons donc pas besoin d'ajouter de contraintes particulières sur les coefficients de la transformation.

Le degré p du polynôme d'échelle et le degré q du polynôme fixe étant choisis, la mesure de similarité du bloc Y au bloc X passe par la résolution du problème

de minimisation dans l'expression (5.2). Cette minimisation se fait en ajustant les variables réelles (a_n) , (b_n) , et (c_n) dans la transformation massique $\tau_{(p,q)}$, ainsi que la variable entière k liée à la transformation spatiale θ_k . La procédure de minimisation est réalisée en optimisant la transformation massique pour les huit valeurs de k . La mesure de similarité finale correspond alors à la mesure minimale obtenue en faisant varier k .

5.3.2 Calcul des coefficients de la transformation massique

Pour une valeur fixée de k , le problème de minimisation qui apparaît en (5.2) revient à minimiser la fonction suivante :

$$\xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q) = \sum_{i,j=0}^{N-1} \left(X_{i,j} - \tau_{(p,q)}(Y_{i,j}) \right)^2 \quad (5.5)$$

Cette fonction étant convexe, on peut obtenir les coefficients optimaux (a_n) , (b_n) , et (c_n) en annulant les dérivées partielles de la fonction ξ pour chaque coefficient, c'est-à-dire :

$$\begin{cases} \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial a_1} = \dots = \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial a_p} = 0 \\ \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial b_0} = \dots = \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial b_q} = 0 \\ \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial c_0} = \dots = \frac{\partial \xi(a_1, \dots, a_p, b_0, \dots, b_q, c_0, \dots, c_q)}{\partial c_q} = 0 \end{cases} \quad (5.6)$$

Ce système d'équations peut se mettre sous la forme matricielle $Ax = B$ suivante :

$$\begin{bmatrix} \overline{Y^2} & \dots & \overline{Y^{p+1}} & \overline{i^0 Y} & \dots & \overline{i^q Y} & \overline{j^0 Y} & \dots & \overline{j^q Y} \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \overline{Y^{p+1}} & \dots & \overline{Y^{2p}} & \overline{i^0 Y^p} & \dots & \overline{i^q Y^p} & \overline{j^0 Y^p} & \dots & \overline{j^q Y^p} \\ \hline \overline{Y i^0} & \dots & \overline{Y^p i^0} & \overline{i^0} & \dots & \overline{i^q} & \overline{j^0 i^0} & \dots & \overline{j^q i^0} \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \overline{Y i^q} & \dots & \overline{Y^p i^q} & \overline{i^q} & \dots & \overline{i^{2q}} & \overline{j^0 i^q} & \dots & \overline{j^q i^q} \\ \hline \overline{Y j^0} & \dots & \overline{Y^p j^0} & \overline{i^0 j^0} & \dots & \overline{i^q j^0} & \overline{j^0} & \dots & \overline{j^q} \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \overline{Y j^q} & \dots & \overline{Y^p j^q} & \overline{i^0 j^q} & \dots & \overline{i^q j^q} & \overline{j^q} & \dots & \overline{j^{2q}} \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_p \\ b_0 \\ \vdots \\ b_q \\ c_0 \\ \vdots \\ c_q \end{bmatrix} = \begin{bmatrix} \overline{XY} \\ \vdots \\ \overline{XY^p} \\ \overline{X i^0} \\ \vdots \\ \overline{X i^q} \\ \overline{X j^0} \\ \vdots \\ \overline{X j^q} \end{bmatrix} \quad (5.7)$$

où la notation \overline{X} est définie par $\overline{x^n y^m} = \sum_{i,j} x^n y^m / N^2$.

La matrice A est une matrice carrée symétrique de largeur $p + 2(q + 1)$. On peut facilement obtenir la matrice inverse A^{-1} et résoudre le système d'équations avec l'algorithme de Gauss-Jordan [109].

5.3.3 Exemples

Nous illustrons le modèle de similarité à partir d'une pyramide à deux niveaux qui est montrée dans l'encadré du haut sur la figure 5.4. Le bloc de référence X

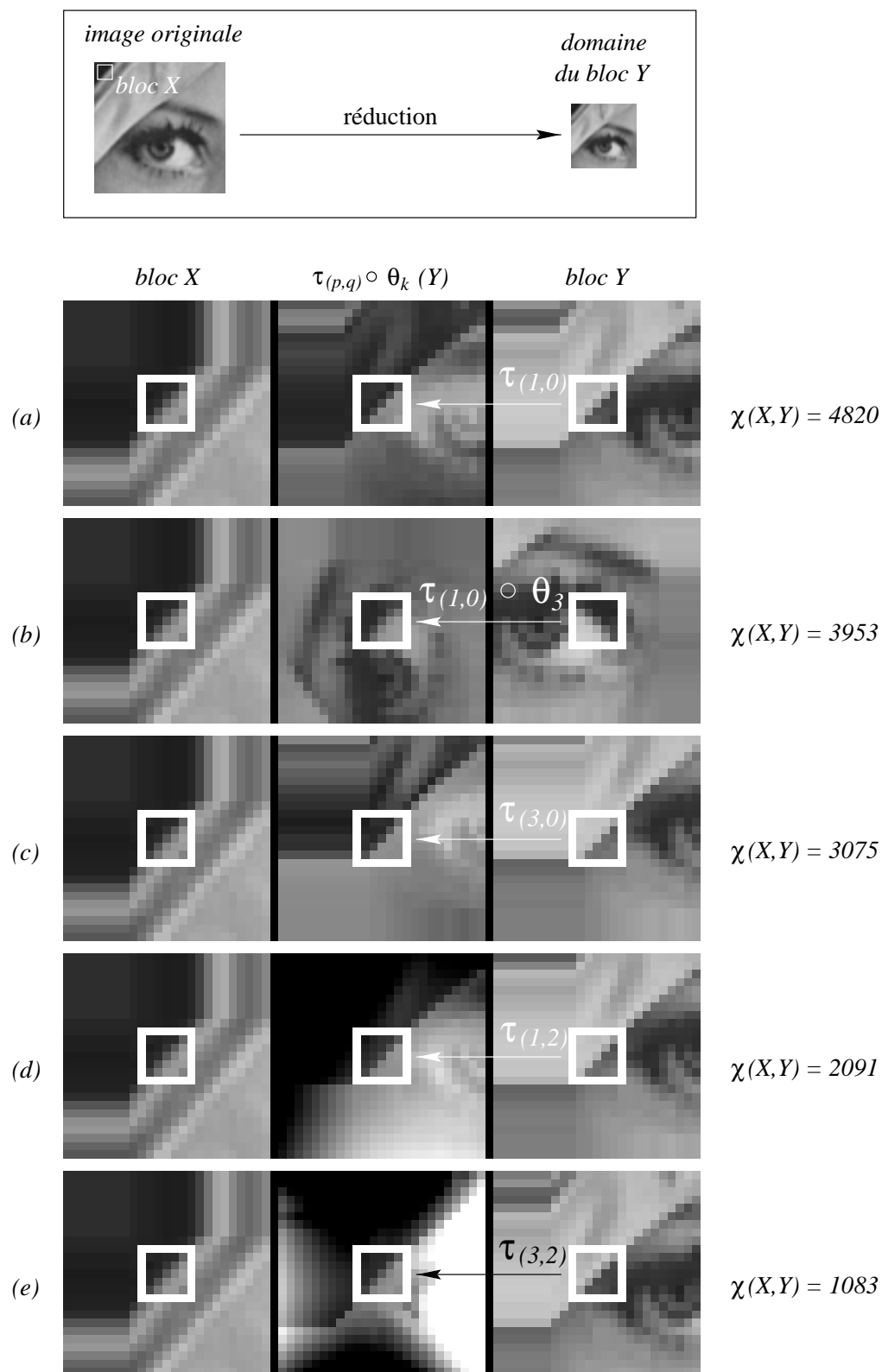


Figure 5.4 – Exemples de similarités avec différentes fonctions de transformation : (a) transformation linéaire $\tau_{(1,0)}$, (b) transformation linéaire $\tau_{(1,0)}$ avec isométrie (rotation $+90^\circ$), (c) transformation $\tau_{(3,0)}$, (d) transformation $\tau_{(1,2)}$, (e) transformation $\tau_{(3,2)}$.

est un bloc de 5×5 pixels situé à la base de la pyramide dans le coin supérieur du chapeau de Lenna. Dans les exemples qui suivent, le bloc Y est le bloc du premier niveau (image réduite) qui minimise la mesure de similarité au bloc X avec une certaine transformation.

La figure 5.4(a) illustre le cas où la transformation massique est linéaire. Le bloc Y le plus similaire au bloc X est situé dans le coin inférieur du chapeau. Dans cet exemple, la transformation $\tau_{(1,0)}$ réalise une inversion des niveaux de gris (le coefficient a_1 est négatif), comme le montre l'image du centre.

Les figures 5.4(c), 5.4(d) et 5.4(e) illustrent des transformations massiques plus complexes avec respectivement $\tau_{(3,0)}$, $\tau_{(1,2)}$ et $\tau_{(3,2)}$. Dans chaque cas, la meilleure similarité est également située dans le coin inférieur du chapeau. Cependant, plus la transformation est complexe et plus la mesure de similarité diminue. Ceci est normal, puisque l'espace de transformation s'agrandit lorsque p et q augmentent. Remarquons que le polynôme fixe permet plus de liberté que le polynôme d'échelle, puisqu'il ajoute un bloc analytique indépendant du bloc X .

La figure 5.4(b) montre le cas où la transformation est composée d'isométries. Dans cet exemple, le bloc Y le plus similaire au bloc X est situé sur l'œil de Lenna. Il est rendu similaire au bloc X après une rotation de 45° . On remarque que la transformation massique $\tau_{(1,0)}$ agit très peu sur les niveaux de gris.

5.4 Construction et organisation primaire d'un domaine de recherche

Dans l'algorithme décrit en section 5.2, la phase d'analyse consiste à trouver des similarités entre la base de la pyramide et ses niveaux supérieurs. Comme la complexité algorithmique de cette opération est particulièrement élevée, il faut mettre en place une stratégie d'optimisation des recherches. Pour cela, les similarités ne sont pas recherchées directement dans la pyramide, mais dans une structure organisée appelée *domaine de recherche*. Celui-ci est composé d'un ensemble de blocs extraits des niveaux supérieurs de la pyramide. La taille de ces blocs est la même que celle de la fenêtre de similarité. Comme nous l'avons déjà précisé en section 5.2, cette taille est de 4×4 pixels lorsque le filtre est pair et de 5×5 pixels lorsqu'il est impair.

Pour construire un domaine de recherche optimal, il faut sélectionner uniquement des blocs porteurs d'une information réellement pertinente. Cependant, la sélection ne doit pas être trop restrictive, sous peine de diminuer la probabilité de trouver de bonnes similarités. Dans la suite de cette section, nous décrivons deux schémas de sélection basés sur la variance. Le premier exploite directement la pyramide construite à partir de l'image originale et le second une pyramide de différences.

5.4.1 La sélection et l'ordre des blocs

La construction initiale du domaine de recherche consiste simplement à sélectionner des blocs de la pyramide. Pour cela, on déplace une fenêtre de sélection sur tous les niveaux de la pyramide sauf la base, comme l'illustre la figure 5.5(a). Pour sélectionner un bloc, la seule contrainte que nous imposons est que sa variance soit supérieure ou égale à un seuil limite σ_{min}^2 . Un bloc homogène ou presque homogène ne présente pas suffisamment d'informations pour faire l'objet d'une similarité. De plus, il est très probable que le peu d'informations relevées dans ce type de bloc ne soit finalement que du bruit. Si ce bloc était sélectionné, il y aurait un risque pour que la transformation analytique amplifie considérablement le bruit et que la similarité ne soit plus significative. Une sélection basée sur la variance apparaît donc comme un excellent moyen pour écarter ce type de bloc.

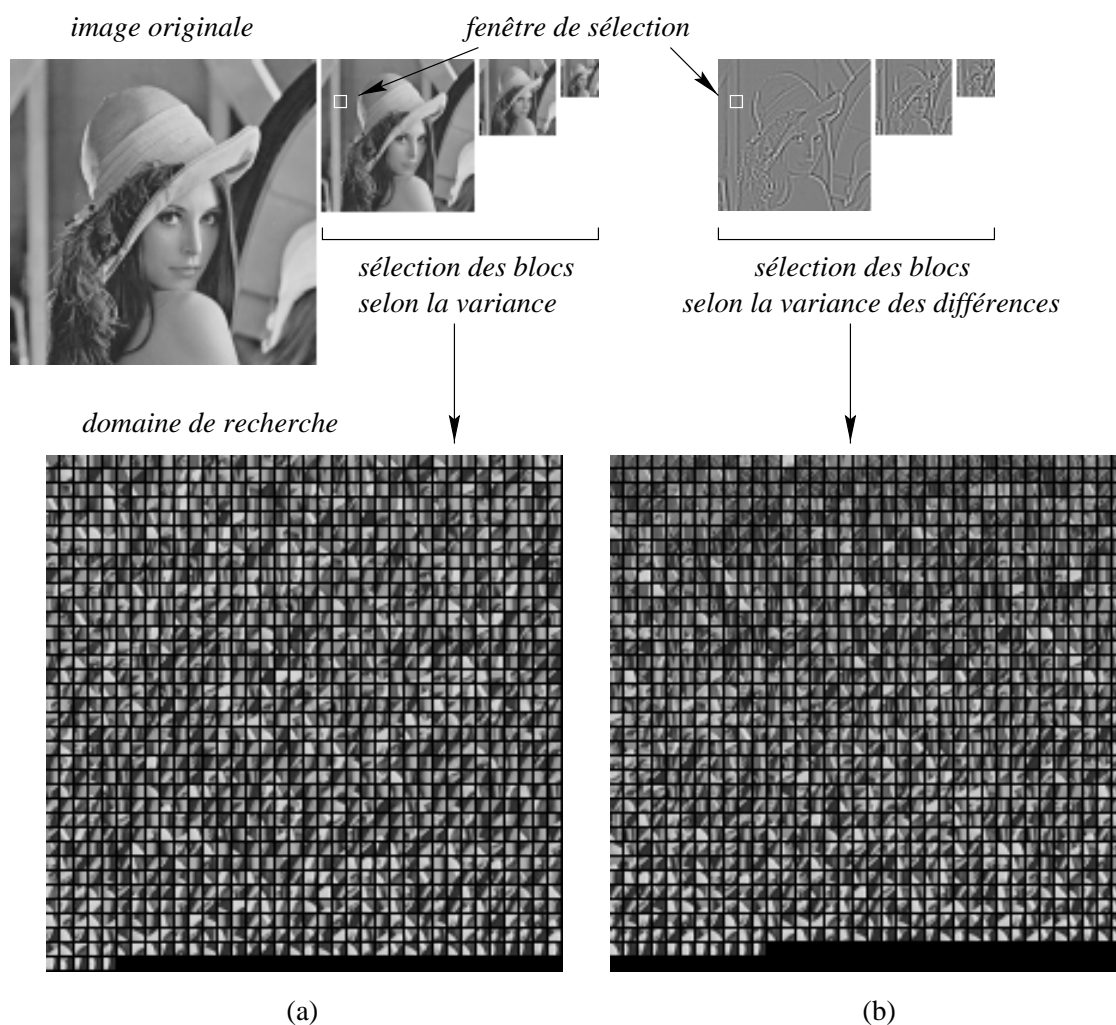


Figure 5.5 – Construction et organisation primaire d'un domaine de recherche (blocs de taille 5×5 pixels) : (a) sélection des blocs sur la pyramide passe-bas, (b) sélection des blocs sur la pyramide de différences (les niveaux de gris sur la pyramide sont amplifiés d'un facteur 2).

Une organisation primaire des blocs sélectionnés peut être réalisée en les ordonnant selon un certain critère mathématique. En référence à un article de Lee *et al.* [83], nous prenons également la mesure de la variance pour ordonner les blocs. Nous verrons dans la section 5.6 comment on peut limiter l'espace de recherche en exploitant cet ordre.

La figure 5.5(a) illustre le domaine de recherche obtenu en ordonnant les blocs selon la variance : le bloc qui a la variance la plus faible est dans le coin haut-gauche et le bloc qui a la variance la plus forte est dans le coin bas-droit. Ces blocs ont été sélectionnés à partir d'une pyramide de 4 niveaux construite avec le filtre 5×5 de Burt ($\beta = 0.6$). La base de la pyramide est une image de Lenna de 128×128 pixels. Le filtre étant impair, on prend une fenêtre de sélection de 5×5 pixels. Dans cet exemple, l'écart-type limite qui a été choisi est $\sigma_{min} = 35$. Cette valeur correspond à une sélection de 1265 blocs. La figure 5.6 montre le nombre de blocs sélectionnés en fonction de la valeur de σ_{min} . On obtient un maximum de 4528 blocs dans le domaine de recherche lorsque $\sigma_{min} = 0$. L'écart-type maximum que l'on peut obtenir pour un bloc de 5×5 pixels avec 256 niveaux de gris est $\sigma \approx 127$. Il correspond au cas où 12 ou 13 pixels sont blancs et 13 ou 12 pixels sont noirs. Pour l'image de Lenna, l'écart-type maximum que l'on peut trouver est $\sigma \approx 75$. Dans cet exemple, la courbe de sélection est à peu près linéaire. La fixation du seuil n'est donc pas facile. Avec une image originale plus simple, comme celle du cameraman, 25% des blocs des trois niveaux supérieurs de la pyramide ont un écart-type inférieur à 2 et 40% ont un écart-type inférieur à 5. Dans ce cas, le choix du seuil est plus simple puisqu'en choisissant $\sigma_{min} = 5$, on supprime une majorité de blocs inintéressants.

5.4.2 La sélection avec une pyramide de différences

Une alternative à la sélection précédente est de sélectionner un bloc en fonction de son contenu fréquentiel. Un bloc de type haute-fréquence est porteur d'une information pertinente puisqu'il apporte une information supplémentaire par rapport à son bloc père dans la pyramide. Au contraire, un bloc de type basse-fréquence ne contient aucune information supplémentaire et ses descendants non plus. Ces derniers ne doivent donc pas être sélectionnés pour construire le domaine de recherche.

On peut obtenir facilement l'information fréquentielle d'un bloc en exploitant une pyramide de différences (cf. section 4.3.5). Cette pyramide se construit directement à partir de la pyramide passe-bas en deux étapes. La première étape consiste à filtrer chaque niveau de la pyramide passe-bas avec le même filtre qui a servi pour sa construction. On obtient ainsi une pyramide filtrée où chaque niveau exprime les basses fréquences des niveaux associés dans la pyramide passe-bas. La deuxième étape consiste à soustraire les niveaux de la pyramide filtrée de ceux de la pyramide passe-bas, afin de générer la pyramide de différences. Chaque niveau de cette pyramide exprime les hautes fréquences des niveaux associés dans la pyramide passe-bas. Il s'agit de l'information perdue lors des réductions successives. La figure 5.5(b) illustre la pyramide de différences obtenue avec la pyramide passe-bas de la figure 5.5(a). La base n'est pas montrée puisqu'elle n'est pas utilisée pour la sélection des

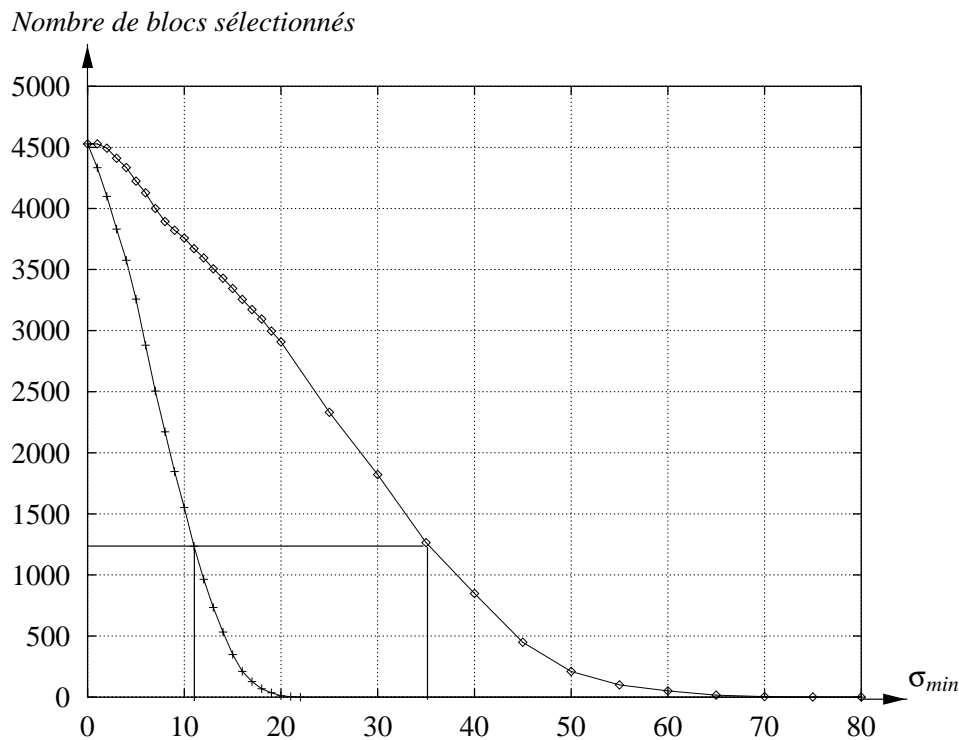


Figure 5.6 – Courbe illustrant le nombre de blocs sélectionnés en fonction de l'écart-type σ_{min} : (◇) sélections sur la pyramide passe-bas, (+) sélections sur la pyramide de différences.

blocs.

Comme dans le schéma de sélection de la section 5.4.1, un bloc est sélectionné si sa variance est supérieure ou égale à un seuil limite σ_{min}^2 . La seule différence est que le calcul de la variance est réalisé sur la pyramide des différences. L'entropie des niveaux de gris étant plus faible, il faut diminuer σ_{min}^2 . Comme précédemment, on peut fixer le seuil en s'aidant d'une courbe qui exprime le nombre de blocs sélectionnés en fonction de la valeur de σ_{min} . Comme le montre la figure 5.6, la courbe obtenue est également linéaire, mais elle est comprimée par rapport à la courbe trouvée en sélectionnant les blocs directement dans la pyramide passe-bas. Pour garder à peu près le même nombre de blocs, c'est-à-dire 1265 blocs, il faut que le seuil ait la valeur $\sigma_{min} = 11$. Dans ce cas, le domaine est composé de 1235 blocs. Le domaine de recherche obtenu avec ce seuil est illustré sur la figure 5.5(b). Par rapport au domaine obtenu sur la figure 5.5(a), on peut observer que les différences se situent surtout au niveau des blocs de faible variance. Avec la pyramide de différences, les blocs de faible variance ont une dynamique moins élevée sur les niveaux de gris. Cependant, ils sont porteurs d'une information fréquentielle plus riche et donc plus pertinente pour la synthèse des similarités. Cependant, le risque de sélectionner des blocs qui sont porteurs de bruit augmente, puisque celui-ci se situe surtout dans les hautes fréquences. Pour profiter des deux schémas de sélection, on peut envisager un schéma hybride où un bloc est sélectionné en fonction de sa variance sur la pyramide passe-bas et de celle sur la pyramide de différences. On a dans ce cas deux variances

limites à respecter.

5.5 Classification par la mesure d'angles entre les espaces de transformation

Pour faciliter la recherche d'une similarité dans le domaine, il est préférable de l'organiser en regroupant les blocs qui ont des caractéristiques communes. En compression par fractale, de nombreuses méthodes ont été proposées dans ce but. De manière générale, on trouve des méthodes qui classifient les blocs en utilisant des mesures mathématiques simples (moyenne, variance, gradient, etc.) [51], [72], et d'autres qui procèdent par agrégation autour de blocs caractéristiques [19], [47], [84]. Cependant, toutes ces méthodes ne sont pas optimales puisque la construction des classes est indépendante du modèle de similarité.

Nous proposons dans cette section de regrouper les blocs du domaine qui ont un espace de transformation relativement proche. La distance entre ces espaces est évaluée par la mesure d'angle. Les groupes ainsi construits contiennent des blocs ayant une mesure de similarité à peu près identique, quel que soit le bloc de référence. Pour évaluer la similarité d'un groupe de blocs, il est donc suffisant de mesurer la similarité avec un seul de ses blocs. Ce principe est décrit en détail dans la section 5.5.1. Dans la section 5.5.2, nous étudions la mesure d'angle entre deux espaces de transformation lorsque le modèle de similarité est basé sur une transformation polynomiale quelconque. Dans la section 5.5.3, nous donnons un algorithme pour calculer l'angle entre deux espaces de transformation dans le cas particulier où la transformation est linéaire. La section 5.5.4 contient des résultats expérimentaux.

5.5.1 Principe général de la classification

Dans la section 5.3.1, nous avons vu que pour mesurer la similarité d'un bloc Y à un bloc X de même taille $N \times N$ pixels, il fallait résoudre un problème de minimisation dans l'expression (5.2). En considérant les blocs X et Y sous une forme vectorielle dans un espace à N^2 dimensions, l'opération de minimisation équivaut à une projection du bloc X sur le sous-espace engendré par la transformation du bloc Y . Chaque dimension de l'espace représente la variation d'intensité associée à un pixel du bloc.

Pour simplifier la présentation de la méthode, prenons le cas d'une transformation massique linéaire $\tau_{(1,0)}$ et d'une transformation spatiale identité θ_0 . La mesure de similarité dans l'espace vectoriel devient alors :

$$\chi(X, Y) = \min_{a_1, b_0, c_0} \| \vec{X} - a_1 \vec{Y} - (b_0 + c_0) \vec{1} \|^2 \quad (5.8)$$

où $\vec{1}$ représente le vecteur unité. La minimisation revient dans ce cas à projeter le vecteur \vec{X} sur le sous-espace à deux dimensions engendré par les vecteurs \vec{Y} et $\vec{1}$, comme l'illustre la figure 5.7 pour le cas simple où les blocs sont composés de trois

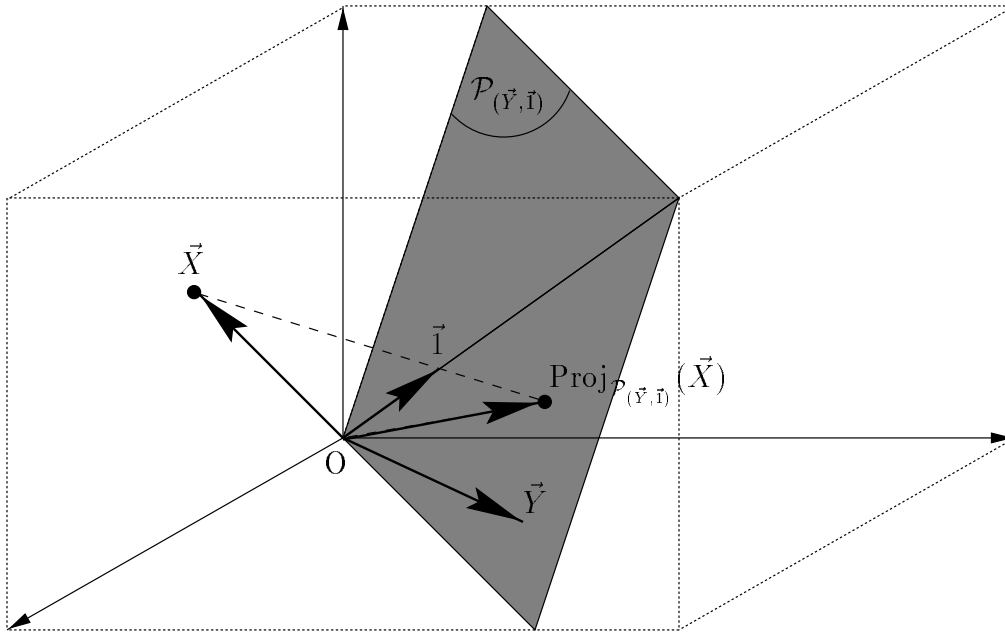


Figure 5.7 – Interprétation géométrique de la mesure de similarité avec une transformation linéaire.

pixels. En notant $\mathcal{P}_{(\vec{Y}, \vec{I})}$ le plan engendré par \vec{Y} et \vec{I} , et $\text{Proj}_{\mathcal{P}_{(\vec{Y}, \vec{I})}}(\vec{X})$ le vecteur issu de la projection orthogonale de \vec{X} sur $\mathcal{P}_{(\vec{Y}, \vec{I})}$, on peut alors réécrire l'expression (5.8) sous la forme suivante :

$$\chi(X, Y) = \|\vec{X} - \text{Proj}_{\mathcal{P}_{(\vec{Y}, \vec{I})}}(\vec{X})\|^2 \quad (5.9)$$

Avec une transformation massique toujours linéaire, considérons deux blocs A et B appartenant au domaine de recherche primaire. Supposons que l'on cherche à mesurer leur similarité à un bloc de référence X . L'espace de transformation associé au bloc A est le plan $\mathcal{P}_{(\vec{A}, \vec{I})}$ et celui associé au bloc B est le plan $\mathcal{P}_{(\vec{B}, \vec{I})}$. Remarquons qu'avec une transformation massique linéaire, les espaces de transformation ont toujours le vecteur unitaire \vec{I} en commun. Soient $\vec{N}_A = \vec{A} \wedge \vec{I}$ et $\vec{N}_B = \vec{B} \wedge \vec{I}$, les vecteurs normaux aux plans de transformation associés respectivement aux vecteurs \vec{A} et \vec{B} . Notons $\alpha = (\vec{N}_A, \vec{N}_B)$ l'angle des deux vecteurs normaux. Lorsque $\alpha = 0$, les deux plans de transformation sont confondus et le résultat de la projection de \vec{X} est identique pour les deux plans. Dans ce cas, nous disons que ces deux blocs sont *similaires par projection*. Quel que soit le vecteur \vec{X} , le résultat de la projection sur les deux plans donne un vecteur unique $\text{Proj}_{\mathcal{P}_{(\vec{A}, \vec{I})}}(\vec{X}) = \text{Proj}_{\mathcal{P}_{(\vec{B}, \vec{I})}}(\vec{X})$. Lorsque deux blocs du domaine sont similaires par projection, on a tout intérêt à les regrouper au sein d'une même classe. Ainsi, on peut mesurer la similarité des blocs d'une classe à un bloc X par une seule projection en choisissant un bloc quelconque de cette classe. On voit donc que la mesure de l'angle α apparaît comme un critère de regroupement des blocs du domaine de recherche primaire. L'espace vectoriel étant en pratique relativement grand (25 dimensions pour des blocs de 5×5 pixels), l'angle

nul est très peu probable. Pour regrouper les blocs, il faut donc accorder une certaine tolérance sur l'angle α . L'angle maximum à ne pas dépasser sera noté α_{max} . Si cet angle-limite est suffisamment petit, on peut considérer que la projection de \vec{X} sur les plans $\mathcal{P}_{(\vec{A}, \vec{I})}$ et $\mathcal{P}_{(\vec{B}, \vec{I})}$ génère deux vecteurs assez proches pour que les blocs A et B soient regroupés au sein d'une même classe.

Lorsque $\alpha_{max} = 0$, la construction d'une classe avec plus de deux blocs ne pose aucun problème particulier. On regroupe simplement les blocs qui partagent le même espace de transformation. Par contre, lorsque $\alpha_{max} > 0$, il existe différentes solutions pour construire une classe avec plus de deux blocs. Une première solution est d'inclure un bloc A dans une classe si pour tous les blocs B appartenant à cette classe on a $\alpha = (\vec{N}_A, \vec{N}_B) \leq \alpha_{max}$. Une deuxième solution est d'inclure un bloc A dans une classe si pour au moins un bloc B appartenant à cette classe on a $\alpha = (\vec{N}_A, \vec{N}_B) \leq \alpha_{max}$. Une troisième solution est d'avoir un unique bloc de référence R dans la classe et d'inclure un bloc A dans cette classe si $\alpha = (\vec{N}_A, \vec{N}_R) \leq \alpha_{max}$. En pratique, nous avons choisi cette dernière solution. Elle permet de limiter la propagation des regroupements et sa complexité algorithmique est plus réduite que celle des deux premières solutions. Avec cette troisième solution, la classification du domaine primaire avec la mesure d'angle va dépendre à la fois du choix de l'angle-limite α_{max} et de celui des blocs de référence pour la construction des classes. On peut voir cette classification comme une sorte d'algorithme d'agrégation.

1. Construire le domaine de recherche primaire \mathcal{D}_1 .
2. Extraire le bloc A de \mathcal{D}_1 qui a la plus grande variance.
3. Extraire tous les blocs B de \mathcal{D}_1 pour lesquels :

$$\alpha = (\vec{N}_A, \vec{N}_B) \leq \alpha_{max}$$

4. Regrouper le bloc A et les blocs extraits B au sein d'une unique classe dans le domaine secondaire \mathcal{D}_2 .
5. Si le domaine \mathcal{D}_1 n'est pas vide, retourner à l'étape 2.

Table 5.2 – Algorithme général de classification du domaine de recherche primaire.

Le principe général de l'algorithme de classification est résumé dans la table 5.2. La construction du domaine de recherche primaire (étape 1) a été étudiée dans la section 5.4. Pour construire une classe du domaine secondaire, nous avons choisi de prendre comme bloc de référence le bloc qui possède la plus grande variance parmi les blocs restants du domaine primaire (étape 2). Ce choix s'explique par le fait que le bloc qui possède la plus grande variance a un rapport information/bruit qui est en probabilité plus élevé que les autres. De plus, il est préférable de créer les

classes à partir de blocs de référence qui contiennent de l'information pertinente par rapport au contenu de l'image. L'élimination progressive des blocs du domaine primaire (étape 3) garantit que tout bloc du domaine primaire appartient à une et une seule classe du domaine secondaire. Elle permet également d'optimiser la complexité algorithmique de la classification. L'algorithme général de classification étant donné, il nous reste à mesurer l'angle α formé par les deux vecteurs normaux aux espaces de transformation associés à deux blocs A et B .

5.5.2 La mesure d'angle entre deux espaces de transformation

Pour mesurer l'angle α entre les deux vecteurs normaux aux espaces de transformation associés aux blocs A et B , il faut effectuer un plongement dans un espace à deux dimensions. Quelle que soit la dimension de l'espace initial, on peut montrer que α peut toujours être obtenu par la formule analytique suivante :

$$\cos \alpha = \frac{\langle \vec{N}_A, \vec{N}_B \rangle}{\| \vec{N}_A \| \cdot \| \vec{N}_B \|} \quad (5.10)$$

La problème de la mesure d'angle étant résolu, il nous reste à déterminer précisément les vecteurs normaux \vec{N}_A et \vec{N}_B . Supposons que les blocs aient une taille de $N \times N$ pixels, l'espace vectoriel associé est dans ce cas de dimension N^2 . Supposons également que la dimension de l'espace de transformation soit de dimension $d < N^2$. Celle-ci dépend en fait de la complexité de la transformation massique dans le modèle de similarité. Pour que le vecteur normal à l'espace de transformation ait une existence, il faut effectuer un plongement dans un espace de dimension $d + 1$. Une fois ce plongement réalisé, il faut effectuer le produit vectoriel des d vecteurs associés à l'espace de transformation pour obtenir le vecteur normal.

Prenons le cas le plus simple où la transformation massique est une transformation linéaire $\tau_{(1,0)}$ avec $b_0 = c_0 = 0$. Dans ce cas, le polynôme fixe n'est pas pris en compte dans la transformation et l'espace de transformation d'un bloc est simplement une droite dont le vecteur directeur est le bloc lui-même. Notons $\mathcal{D}_{\vec{A}}$ et $\mathcal{D}_{\vec{B}}$ les droites de transformation associées aux blocs A et B . En théorie, pour obtenir les vecteurs normaux aux droites $\mathcal{D}_{\vec{A}}$ et $\mathcal{D}_{\vec{B}}$, il faut faire un plongement dans l'espace à deux dimensions engendré par les vecteurs \vec{A} et \vec{B} . En pratique, cette procédure peut être évitée. En effet, dans l'espace à deux dimensions, l'angle entre \vec{N}_A et \vec{N}_B est égal à l'angle entre \vec{A} et \vec{B} . Dans ce cas particulier, on peut donc obtenir α en calculant directement l'angle entre les vecteurs \vec{A} et \vec{B} dans l'espace initial à N^2 dimensions. Pour cela, il suffit d'utiliser la formule (5.10) en substituant \vec{N}_A par \vec{A} et \vec{N}_B par \vec{B} .

Prenons maintenant le cas un peu plus compliqué d'une transformation massique linéaire $\tau_{(1,0)}$ où les coefficients b_0 et c_0 ne sont pas contraints. Comme nous l'avons déjà précisé précédemment, les espaces de transformation associés aux blocs A et B sont définis par deux plans $\mathcal{P}_{(\vec{A}, \vec{I})}$ et $\mathcal{P}_{(\vec{B}, \vec{I})}$. Dans ce cas, pour obtenir les vecteurs normaux \vec{N}_A et \vec{N}_B , il faut effectuer un plongement dans un espace à trois dimensions. Ce plongement est réalisé dans l'espace défini par les trois vecteurs \vec{A} , \vec{B} et

$\vec{1}$. On détermine ensuite les vecteurs normaux en effectuant les produits vectoriels $\vec{N}_A = \vec{A} \wedge \vec{1}$ et $\vec{N}_B = \vec{B} \wedge \vec{1}$. Un algorithme détaillé est donné dans la section 5.5.3.

Pour généraliser cette approche à une transformation massique $\tau_{(p,q)}$ définie par (5.3), on considère la transformation équivalente dans l'espace vectoriel des blocs. On obtient l'expression suivante :

$$\tau_{(p,q)}(\vec{Y}) = \sum_{n=1}^p a_n \vec{Y}^{\vec{n}} + \sum_{n=0}^q (b_n \vec{i}^{\vec{n}} + c_n \vec{j}^{\vec{n}}) \quad (5.11)$$

Dans le cas général, on voit que l'espace de transformation associé à un bloc est engendré par un ensemble de $p + 2q + 1$ vecteurs. Pour simplifier l'écriture, nous notons cet espace $\mathcal{E}_{(\vec{Y}^{\vec{p}}, \vec{i}^{\vec{q}}, \vec{j}^{\vec{q}})}$. Pour obtenir les vecteurs normaux \vec{N}_A et \vec{N}_B , il faut nécessairement effectuer un plongement dans un espace à $p + 2q + 2$ dimensions. L'espace de plus petites dimensions dans lequel on peut effectuer un plongement est l'espace engendré par la totalité des vecteurs qui définissent l'espace $\mathcal{E}_{(\vec{A}^{\vec{p}}, \vec{i}^{\vec{q}}, \vec{j}^{\vec{q}})}$ et la totalité de ceux qui définissent l'espace $\mathcal{E}_{(\vec{B}^{\vec{p}}, \vec{i}^{\vec{q}}, \vec{j}^{\vec{q}})}$. Ces deux espaces ont en commun les vecteurs associés au polynôme fixe dans l'expression (5.11), c'est-à-dire les vecteurs $\vec{i}^{\vec{q}}, \dots, \vec{i}, \vec{j}^{\vec{q}}, \dots, \vec{j}$ et $\vec{1}$. Ainsi, la dimension minimale du plongement est $2p + 2q + 1$. Pour que les vecteurs normaux aient une existence dans un espace commun, il faut forcément que l'on ait $p = 1$. Cependant, il n'y a pas de restriction sur le degré q du polynôme fixe. En conclusion, la mesure de l'angle α n'est possible que si le polynôme d'échelle est linéaire. Dans ce cas, on peut réaliser le plongement dans l'espace à $2q + 3$ dimensions. Les vecteurs normaux sont ensuite obtenus par les produits vectoriels $\vec{N}_A = \vec{A} \wedge \vec{i}^{\vec{q}} \wedge \dots \wedge \vec{i} \wedge \vec{j}^{\vec{q}} \wedge \dots \wedge \vec{j} \wedge \vec{1}$ et $\vec{N}_B = \vec{B} \wedge \vec{i}^{\vec{q}} \wedge \dots \wedge \vec{i} \wedge \vec{j}^{\vec{q}} \wedge \dots \wedge \vec{j} \wedge \vec{1}$. Dans la suite, nous nous intéressons uniquement au cas où $q = 0$.

5.5.3 Un algorithme mesurant l'angle entre deux plans de transformation

Dans cette section, nous développons un algorithme pour mesurer l'angle entre deux espaces de transformation associés à deux blocs A et B lorsque la transformation massique est linéaire (cf. équation (5.8)). Dans ce cas, les deux espaces de transformation sont décrits par les deux plans $\mathcal{P}_{(\vec{A}, \vec{1})}$ et $\mathcal{P}_{(\vec{B}, \vec{1})}$ dans l'espace vectoriel des blocs. Plus rigoureusement, l'angle entre les deux plans de transformation correspond à l'angle α formé par leurs vecteurs normaux respectifs \vec{N}_A et \vec{N}_B . Pour que ceux-ci aient une existence, la première opération à réaliser est de faire un plongement de l'espace initial vers un espace à trois dimensions.

L'espace initial, noté E , est l'espace vectoriel des blocs de taille $N \times N$ pixels. La dimension de l'espace E est donc N^2 . On munit cet espace d'un repère orthonormal $(O, \vec{e}_1, \vec{e}_2, \dots, \vec{e}_{N^2})$. Les vecteurs \vec{A} , \vec{B} et $\vec{1}$ sont initialement définis dans cet espace. Le plongement de ces trois vecteurs dans un espace à trois dimensions est toujours possible. Pour cela, il faut trouver un nouveau repère orthonormal $(O, \vec{k}_1, \vec{k}_2, \dots, \vec{k}_{N^2})$ dans l'espace E ayant la propriété de rendre nul les produits scalaires des vecteurs \vec{A} , \vec{B} et $\vec{1}$ avec les vecteurs $\vec{k}_4, \vec{k}_5, \dots, \vec{k}_{N^2}$. Une fois ce repère

1. Annulation des composantes du vecteur \vec{I} :

Pour i allant de N^2 à 2

$$\begin{cases} x_1^{(N^2-i+1)}[\vec{I}] = \sqrt{x_1^{(N^2-i)}[\vec{I}]^2 + x_i^{(N^2-i)}[\vec{I}]^2} \\ x_i^{(N^2-i+1)}[\vec{I}] = 0 \end{cases}$$

$$\cos \beta = \frac{x_1^{(N^2-i)}[\vec{I}]}{x_1^{(N^2-i+1)}[\vec{I}]} \quad \sin \beta = \frac{x_i^{(N^2-i)}[\vec{I}]}{x_1^{(N^2-i+1)}[\vec{I}]}$$

$$\begin{cases} x_1^{(N^2-i+1)}[\vec{A}] = x_1^{(N^2-i)}[\vec{A}] \cdot \cos \beta + x_i^{(N^2-i)}[\vec{A}] \cdot \sin \beta \\ x_i^{(N^2-i+1)}[\vec{A}] = x_i^{(N^2-i)}[\vec{A}] \cdot \cos \beta - x_1^{(N^2-i)}[\vec{A}] \cdot \sin \beta \end{cases}$$

$$\begin{cases} x_1^{(N^2-i+1)}[\vec{B}] = x_1^{(N^2-i)}[\vec{B}] \cdot \cos \beta + x_i^{(N^2-i)}[\vec{B}] \cdot \sin \beta \\ x_i^{(N^2-i+1)}[\vec{B}] = x_i^{(N^2-i)}[\vec{B}] \cdot \cos \beta - x_1^{(N^2-i)}[\vec{B}] \cdot \sin \beta \end{cases}$$

2. Annulation des composantes du vecteur \vec{A} :

Pour i allant de N^2 à 3

$$\begin{cases} x_2^{(N^2-i+2)}[\vec{A}] = \sqrt{x_2^{(N^2-i+1)}[\vec{A}]^2 + x_i^{(N^2-i+1)}[\vec{A}]^2} \\ x_i^{(N^2-i+2)}[\vec{A}] = 0 \end{cases}$$

$$\cos \beta = \frac{x_2^{(N^2-i+1)}[\vec{A}]}{x_2^{(N^2-i+2)}[\vec{A}]} \quad \sin \beta = \frac{x_i^{(N^2-i+1)}[\vec{A}]}{x_2^{(N^2-i+2)}[\vec{A}]}$$

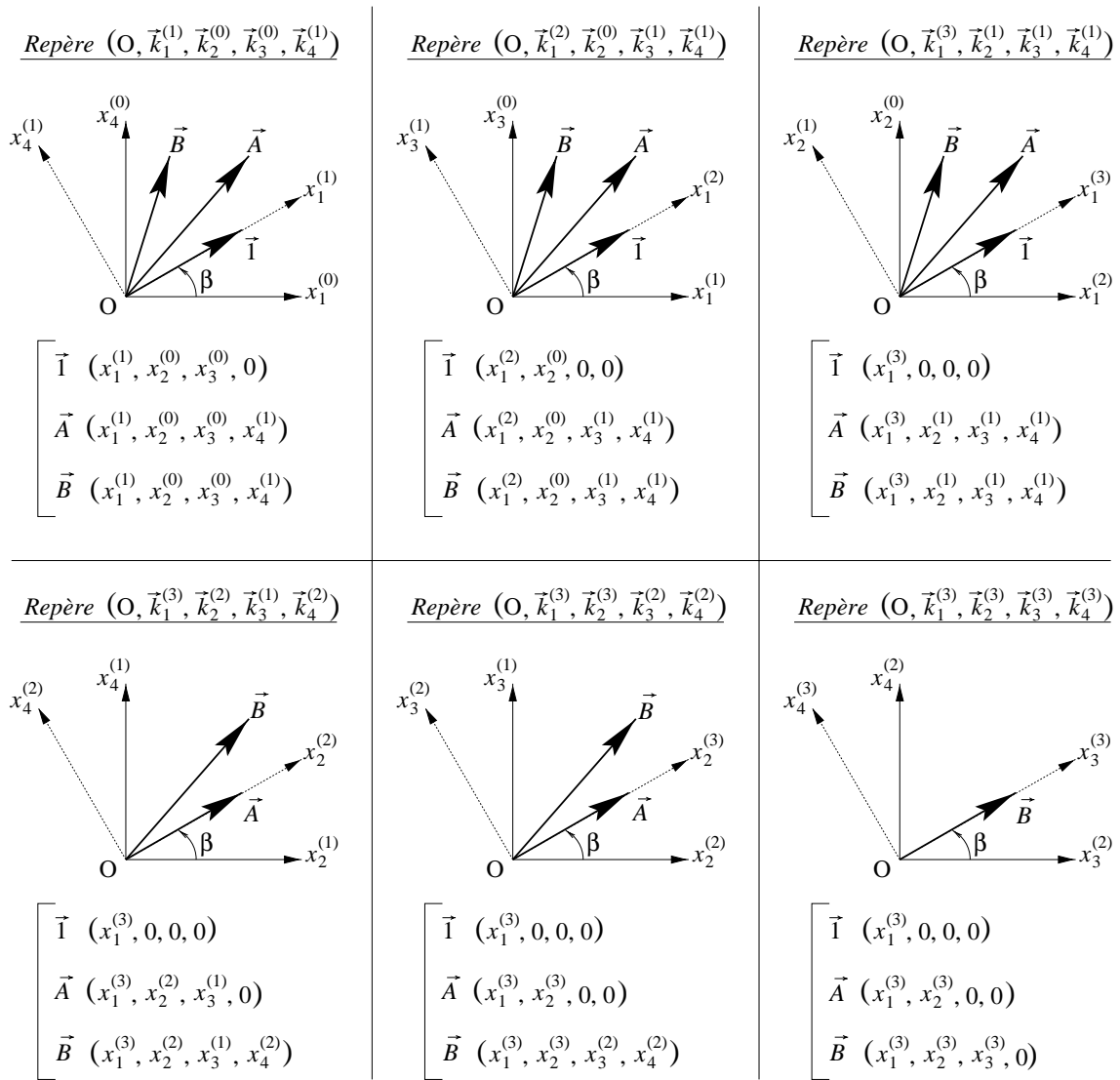
$$\begin{cases} x_2^{(N^2-i+2)}[\vec{B}] = x_2^{(N^2-i+1)}[\vec{B}] \cdot \cos \beta + x_i^{(N^2-i+1)}[\vec{B}] \cdot \sin \beta \\ x_i^{(N^2-i+2)}[\vec{B}] = x_i^{(N^2-i+1)}[\vec{B}] \cdot \cos \beta - x_2^{(N^2-i+1)}[\vec{B}] \cdot \sin \beta \end{cases}$$

3. Annulation des composantes du vecteur \vec{B} :

Pour i allant de N^2 à 4

$$\begin{cases} x_3^{(N^2-i+3)}[\vec{B}] = \sqrt{x_3^{(N^2-i+2)}[\vec{B}]^2 + x_i^{(N^2-i+2)}[\vec{B}]^2} \\ x_i^{(N^2-i+3)}[\vec{B}] = 0 \end{cases}$$

Table 5.3 – Algorithme de plongement des vecteurs \vec{A} , \vec{B} et \vec{I} dans un espace H à trois dimensions.


 Figure 5.8 – Illustration de l'algorithme de la table 5.3 pour $N = 2$.

déterminé, on peut redéfinir facilement les vecteurs \vec{A} , \vec{B} et \vec{I} dans l'espace à trois dimensions H muni du repère $(O, \vec{k}_1, \vec{k}_2, \vec{k}_3)$.

La table 5.3 illustre l'algorithme que nous utilisons pour obtenir les composantes des vecteurs \vec{A} , \vec{B} et \vec{I} dans le repère $(O, \vec{k}_1, \vec{k}_2, \dots, \vec{k}_{N^2})$. Initialement, ces trois vecteurs sont exprimés dans le repère $(O, \vec{k}_1^{(0)}, \vec{k}_2^{(0)}, \dots, \vec{k}_{N^2}^{(0)}) = (O, \vec{e}_1, \vec{e}_2, \dots, \vec{e}_{N^2})$. Les composantes des vecteurs \vec{A} , \vec{B} et \vec{I} sont exprimées dans une succession de repères jusqu'à obtenir un repère final $(O, \vec{k}_1^{(N^2-1)}, \vec{k}_2^{(N^2-1)}, \vec{k}_3^{(N^2-1)}, \vec{k}_4^{(3)}, \dots, \vec{k}_{N^2}^{(3)}) = (O, \vec{k}_1, \vec{k}_2, \dots, \vec{k}_{N^2})$. Chacun des repères successifs est obtenu par une rotation élémentaire du repère précédent qui modifie au plus deux composantes des vecteurs \vec{A} , \vec{B} et \vec{I} . Plus précisément, l'algorithme est composé de trois étapes. Pour faciliter sa compréhension, nous avons illustré sur la figure 5.8 la succession des repères utilisés

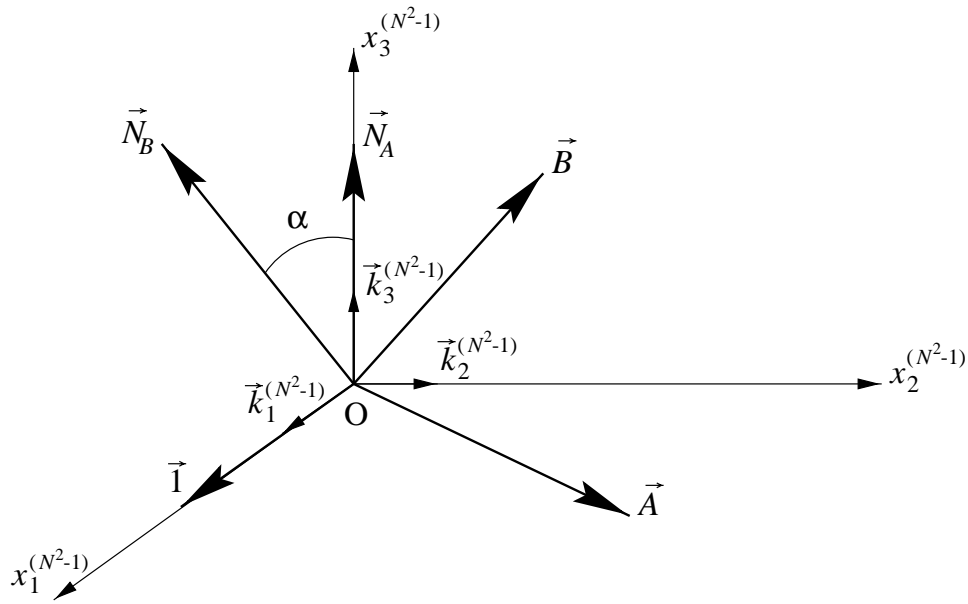


Figure 5.9 – Mesure de l'angle α entre les vecteurs normaux \vec{N}_A et \vec{N}_B après un plongement des vecteurs \vec{A} , \vec{B} et \vec{I} dans l'espace H à trois dimensions : la situation illustrée est celle obtenue au terme de l'algorithme de la table 5.3.

dans le cas où $N = 2$.

La première étape de l'algorithme consiste à annuler toutes les composantes du vecteur \vec{I} sauf la première, par une succession de rotations du repère initial. A la fin de cette première étape, on obtient le repère $(O, \vec{k}_1^{(N^2-1)}, \vec{k}_2^{(1)}, \dots, \vec{k}_{N^2}^{(1)})$ dans lequel \vec{I} est colinéaire à $\vec{k}_1^{(N^2-1)}$. Les composantes du vecteur \vec{I} associées aux autres vecteurs de ce repère sont donc nulles.

Dans la deuxième étape, toutes les composantes du vecteur \vec{A} , sauf les deux premières, sont annulées par une succession de rotations du repère précédent. A la fin de cette deuxième étape, on obtient le repère $(O, \vec{k}_1^{(N^2-1)}, \vec{k}_2^{(N^2-1)}, \vec{k}_3^{(2)}, \dots, \vec{k}_{N^2}^{(2)})$ dans lequel \vec{A} est exprimé uniquement avec les deux vecteurs $\vec{k}_1^{(N^2-1)}$ et $\vec{k}_2^{(N^2-1)}$.

Enfin, dans la troisième étape, on annule toutes les composantes du vecteur \vec{B} sauf les trois premières, toujours par une succession de rotations élémentaires. On obtient finalement le repère $(O, \vec{k}_1^{(N^2-1)}, \vec{k}_2^{(N^2-1)}, \vec{k}_3^{(N^2-1)}, \vec{k}_4^{(3)}, \dots, \vec{k}_{N^2}^{(3)})$ dans lequel \vec{B} est exprimé uniquement à partir des trois vecteurs $\vec{k}_1^{(N^2-1)}$, $\vec{k}_2^{(N^2-1)}$ et $\vec{k}_3^{(N^2-1)}$.

La situation finale de l'algorithme est illustrée sur la figure 5.9. Les trois vecteurs \vec{A} , \vec{B} et \vec{I} sont plongés dans l'espace H à trois dimensions muni du repère $(O, \vec{k}_1^{(N^2-1)}, \vec{k}_2^{(N^2-1)}, \vec{k}_3^{(N^2-1)}) = (O, \vec{k}_1, \vec{k}_2, \vec{k}_3)$.

Une fois que le plongement des trois vecteurs \vec{A} , \vec{B} et \vec{I} dans un espace à trois dimensions est réalisé, on peut facilement obtenir les vecteurs normaux aux plans

de transformation $\mathcal{P}_{(\vec{A}, \vec{I})}$ et $\mathcal{P}_{(\vec{B}, \vec{I})}$ en effectuant les produits vectoriels suivants :

$$\begin{cases} \vec{N}_A = \vec{A} \wedge \vec{I} = x_2[\vec{A}] \cdot \vec{k}_3 \\ \vec{N}_B = \vec{B} \wedge \vec{I} = x_2[\vec{B}] \cdot \vec{k}_3 - x_3[\vec{B}] \cdot \vec{k}_2 \end{cases} \quad (5.12)$$

En utilisant la formule (5.10), on obtient finalement l'angle α :

$$\cos \alpha = \frac{x_2[\vec{B}]}{\sqrt{x_2[\vec{B}]^2 + x_3[\vec{B}]^2}} \quad (5.13)$$

5.5.4 Mise en œuvre

Dans cette section, nous appliquons l'algorithme de classification donné dans la table 5.2 en utilisant une transformation massique linéaire $\tau_{(1,0)}$. L'angle α formé par les vecteurs normaux aux plans de transformation est calculé en appliquant l'algorithme de la section 5.5.3. Le domaine de recherche primaire que l'on cherche à classifier est celui qui est illustré sur la figure 5.5(b). Il est composé de 1235 blocs.

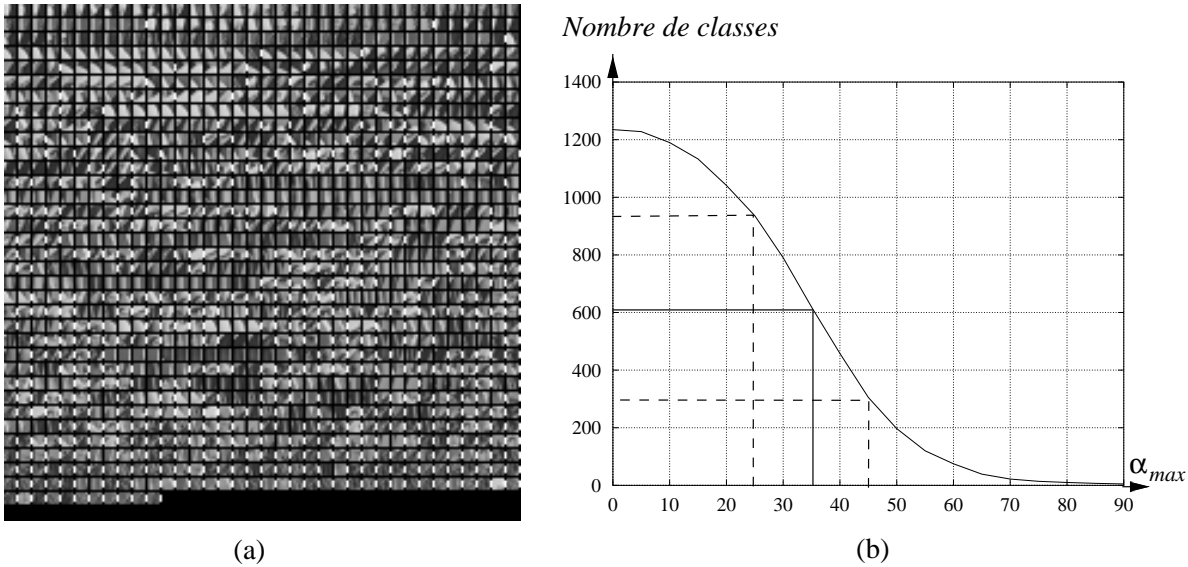


Figure 5.10 – (a) Domaine de recherche classifié avec une transformation massique linéaire $\tau_{(1,0)}$ et un angle-limite $\alpha_{max} = 35^\circ$. La suite de blocs encadrée par deux traits blancs verticaux délimite une même classe. (b) Courbe illustrant le nombre de classes générées en fonction de la valeur de l'angle α_{max} (en degrés).

La figure 5.10(b) montre le nombre de classes obtenues en faisant varier l'angle-limite α_{max} . Avec un angle nul, le nombre de classes est égal au nombre de blocs du domaine, soit 1235. Il est très rare que des blocs issus d'une image naturelle partagent le même espace de transformation. D'après la courbe, le nombre de classes commence à diminuer significativement à partir de 20° . Entre 25° et 45° la courbe est à peu près linéaire. A partir de 50° , la progression devient logarithmique. La classe unique est finalement obtenue pour $\alpha_{max} \approx 160^\circ$.

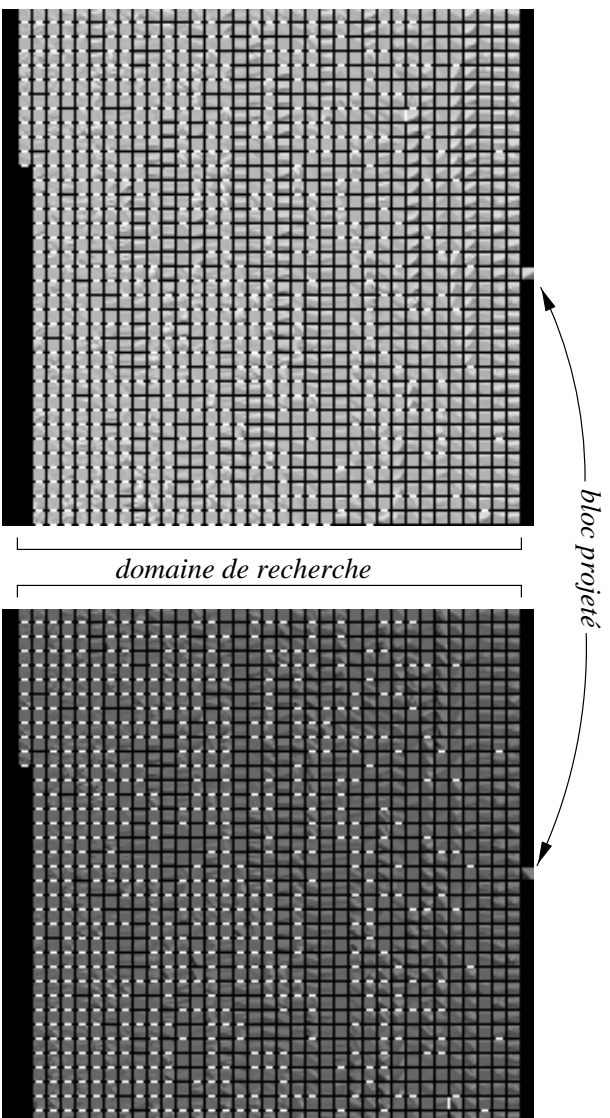


Figure 5.11 – Exemple de projection de deux blocs sur le domaine de recherche classifié. Le bloc le plus similaire est souligné par un trait blanc.

La figure 5.10(a) illustre le domaine obtenu pour $\alpha_{max} = 35^\circ$. Le bloc de référence A de plus grande variance que l'on utilise pour construire la première classe (cf. algorithme de la table 5.2) est situé dans le coin supérieur gauche. Cette première classe est constituée des blocs suivants, considérés de gauche à droite puis de haut en bas jusqu'au premier segment vertical blanc. À l'intérieur d'une classe, les blocs sont ordonnés selon la variance dans l'ordre décroissant de la gauche vers la droite. Cet ordre découle directement de l'organisation du domaine primaire réalisée dans la section 5.4.1, il n'est pas fait explicitement lors de la classification. Le représentant d'une classe (bloc de référence) est toujours le bloc qui possède la plus grande variance dans la classe. Les représentants des classes du domaine sont donc toujours situés immédiatement après un segment blanc vertical.

Le domaine obtenu sur la figure 5.10(a) montre que notre algorithme de classification permet de regrouper facilement les blocs approximativement similaires par projection. La figure 5.11 renforce cette idée en montrant le résultat de la projection de deux blocs différents sur tout le domaine. Pour chaque bloc du domaine, nous montrons le résultat de la projection sur le plan de transformation associé à ce bloc. La projection qui ressemble le plus au bloc projeté détermine le bloc du domaine qui lui est le plus similaire. Ce bloc est souligné par un trait blanc sur la figure 5.11. On peut remarquer que les blocs regroupés au sein d'une même classe engendrent des projections à peu près identiques.

Sur la figure 5.10(a), il est intéressant de noter que les classes regroupant le plus d'éléments sont généralement des classes qui contiennent des blocs à contour simple. Les blocs texturés sont plutôt isolés. Ceci corrobore l'idée selon laquelle on trouve plus facilement des similarités pour les blocs qui représentent des contours simples.

5.6 La recherche d'une similarité dans le domaine

Dans cette section, nous étudions la recherche d'une similarité dans le domaine \mathcal{D} pour un bloc X situé à la base de la pyramide. On peut envisager deux stratégies de recherche diamétralement opposées.

La première est une stratégie de *recherche globale*. Dans ce cas, on souhaite trouver la meilleure similarité au bloc X , sans autre condition que le bloc similaire appartienne au domaine \mathcal{D} . Une recherche exhaustive qui consisterait à mesurer la similarité de chaque bloc du domaine serait beaucoup trop longue en temps de calcul. Pour accélérer significativement ce type de recherche, nous proposons dans la section 5.6.2 un algorithme de recherche qui exploite la classification du domaine étudiée dans la section précédente.

La seconde stratégie est une stratégie de *recherche locale*. Dans ce cas, on recherche une similarité locale au bloc X sur l'axe des résolutions. Ce type de recherche est basé sur l'idée de régularité des contours le long de l'axe des résolutions. On sait que les discontinuités que l'on trouve généralement dans une scène naturelle génèrent des contours à tous les niveaux de résolution. De manière plus pratique, on peut observer ce phénomène de régularité sur une pyramide comme celle représentée sur la figure 5.5(a). Par exemple, le contour associé à l'épaule de Lenna est présent sur tous les niveaux de la pyramide. Pour un bloc X qui contient un contour simple, on a donc une forte probabilité de trouver une similarité locale de bonne qualité le long de l'axe des résolutions. La notion de localité est définie en pratique par un rayon de recherche autour du bloc X sur les niveaux supérieurs de la pyramide. Pour enrichir cette stratégie, nous délocalisons partiellement la recherche en exploitant les classes du domaine \mathcal{D} . L'algorithme de recherche locale est étudié en détail dans la section 5.6.3.

La stratégie de recherche locale est surtout adaptée pour les blocs qui contiennent des contours simples. Pour les blocs plus complexes, comme les blocs contenant des contours multiples ou des textures, on augmente la probabilité de trouver une bonne similarité en effectuant une recherche globale.

La recherche d'une similarité n'est pas toujours utile. Par exemple, si le bloc X est homogène, une simple interpolation linéaire peut être suffisante pour synthétiser son bloc fils situé au niveau de super-résolution. Dans la section 5.6.1, nous étudions les conditions à une recherche de similarité.

5.6.1 Conditions pour exécuter une recherche

Les conditions que nous imposons au bloc X pour lancer une recherche de similarité sont comparables aux contraintes de sélection des blocs pour construire le domaine \mathcal{D} que nous avons présentées dans la section 5.4.

On effectue une recherche de similarité si la variance du bloc X est supérieure ou égale à un seuil limite σ_{min}^2 . Comme dans la section 5.4, la variance du bloc peut être calculée en considérant soit la base de la pyramide passe-bas, soit la base de la pyramide des différences (voir figure 5.12). Dans le premier cas, on prend en compte

l'homogénéité du bloc, tandis que dans le deuxième, on prend plutôt en compte le contenu fréquentiel du bloc. Cette deuxième solution est *a priori* plus justifiée que la première. En effet, lorsque le bloc X ne contient pas de hautes fréquences, il est peu probable que son bloc fils en contienne au niveau de super-résolution de la pyramide. Quoi qu'il en soit, un tel bloc X ne fournit pas suffisamment d'informations pour permettre une synthèse par similarité. Pour ce type de bloc, il est donc inutile de faire une recherche de similarité. Pour synthétiser son bloc fils, on se contente de préserver le contenu fréquentiel du bloc X avec la méthode d'interpolation par transformée B-spline étudiée dans la section 2.3.3.

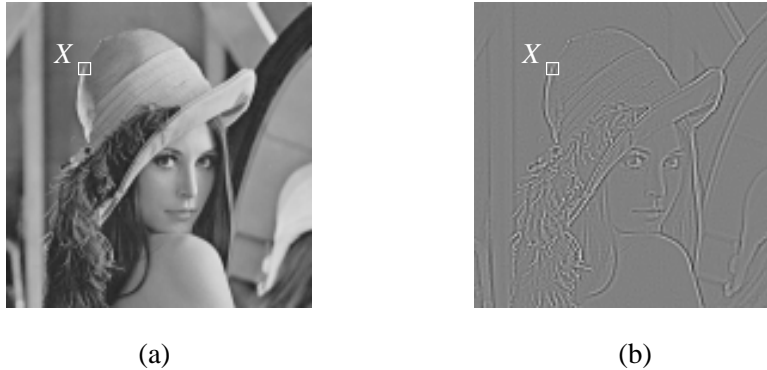


Figure 5.12 – Calcul de la variance du bloc X sur : (a) la base de la pyramide passe-bas, (b) la base de la pyramide des différences (les niveaux de gris sont amplifiés d'un facteur 2).

5.6.2 La recherche globale

En compression par fractale, de nombreuses méthodes ont été proposées pour trouver rapidement une similarité à un bloc parmi un ensemble de blocs (domaine). En général, ces méthodes accélèrent la recherche soit en tirant partie d'une organisation particulière du domaine de recherche, soit en remplaçant les mesures de similarité par des mesures de comparaison rapides. Dans la première catégorie de méthodes, on retrouve les méthodes de classification et d'agrégation que nous avons déjà présentées brièvement dans la section 5.5. On peut également trouver la méthode de Lepsøy *et al.* qui organise les blocs dans une structure arborescente [84]. Dans la deuxième catégorie de méthodes, on peut citer les méthodes fonctionnelles qui consistent à faire des comparaisons indirectes avec un vecteur unitaire indépendant [120]. Il y a aussi la méthode de Saupe, qui en associant à chaque bloc un vecteur caractéristique orthonormal au bloc unitaire, aboutit à un problème de recherche du plus proche voisin [119]. A partir de ces deux catégories de méthodes, on peut évidemment envisager des méthodes hybrides.

Dans cette section, nous proposons une méthode de recherche globale qui exploite le modèle de classification du domaine \mathcal{D} que nous avons développé dans la section 5.5. Le principe de recherche initial est relativement simple. On peut discerner deux phases. Dans la première, on mesure la similarité des représentants des classes avec le bloc X . En considérant le représentant le plus similaire, on mesure dans la deuxième

phase la similarité des blocs de la classe associée à ce représentant. Pour accélérer la première phase, nous exploitons également une méthode proposée par Lee *et al.* qui consiste à limiter l'exploration du domaine en se basant sur la variance des blocs [83]. Nous appliquons cette méthode sur le sous-domaine engendré par la totalité des représentants des classes. L'idée développée par Lee *et al.* est qu'en pratique, la probabilité est faible pour que la variance de deux blocs similaires soit très éloignée. On peut donc limiter la recherche en considérant en priorité les blocs qui ont une variance proche du bloc X et en limitant la profondeur de la recherche.

- Les N classes du domaine issues de l'algorithme de la table 5.2 et les représentants associés sont désignés respectivement par (C_i) et (R_i) avec $1 \leq i \leq N$.
1. Parmi les représentants (R_i) , trouver celui qui a la variance la plus proche de celle du bloc X . Soit R_D ce représentant.
 2. Choisir le pourcentage w de représentants à explorer autour de R_D . Soit W le nombre de représentants qui en résulte ($W = (N * w)/100$).
 3. Calculer les indices i_{min} et i_{max} des représentants à évaluer :

$$\begin{cases} i_{min} = N - W & , & i_{max} = N & \text{si } D + W - N > 0 \\ i_{min} = 1 & , & i_{max} = W & \text{si } D - W < 0 \\ i_{min} = D - W/2 & , & i_{max} = D + W/2 & \text{sinon} \end{cases}$$

4. Mesurer la similarité au bloc X des représentants R_i pour lesquels $i_{min} \leq i \leq i_{max}$. Soit R_S le représentant qui minimise la mesure de similarité.
5. Mesurer la similarité au bloc X de tous les blocs appartenant à la classe C_S . Le bloc le plus similaire est choisi comme similarité finale du bloc X .

Table 5.4 – Algorithme de recherche globale d'une similarité du bloc X dans le domaine de recherche classifié \mathcal{D} .

L'algorithme que nous utilisons est résumé dans la table 5.4. La recherche de similarité parmi les représentants (étape 4) s'organise autour du bloc R_D (étape 1). Nous rappelons que les représentants sont ordonnés selon la variance : le bloc R_1 a la variance la plus forte et le bloc R_N a la variance la plus faible. La profondeur de recherche dépend du paramètre W (étape 2). En théorie, cette profondeur est symétrique autour de R_D , c'est-à-dire que l'on évalue $W/2$ représentants à sa gauche et autant à sa droite. En pratique, si le bloc R_D est suffisamment proche des extrêmes, on risque de sortir de l'intervalle permis $[1..N]$ et dans ce cas, le nombre total des représentants évalués ne correspond plus au choix initial W . Pour éviter

ce problème, lorsque la profondeur à gauche ne peut pas être atteinte, on augmente en proportion celle de droite et *vice versa*. Les indices i_{min} et i_{max} des représentants à évaluer sont calculés en respectant ce principe (étape 3).

5.6.3 La recherche locale

Pour rechercher une similarité localement au bloc X , on prend uniquement en compte les blocs du domaine \mathcal{D} pour lesquels le test de la table 5.5 est positif. L'algorithme de recherche correspondant est donné dans la table 5.6(a).

Le test de la table 5.5 indique si un bloc Y situé sur les niveaux supérieurs de la pyramide appartient à la localité du bloc X , situé quant à lui sur la base de la pyramide. Lorsque le paramètre de tolérance T est nul, un test positif indique que la projection spatiale du bloc Y sur la base de la pyramide recouvre complètement le bloc X . Lorsque $0 < T \leq L$, un test positif peut éventuellement indiquer un recouvrement partiel. Lorsque $T > L$, le test peut être positif même si la projection de Y ne recouvre pas du tout X . On voit donc qu'en augmentant le paramètre T , on augmente la localité du bloc X .

- Le bloc X appartient au niveau 0 de la pyramide (base).
- Le bloc Y appartient au niveau $niv(Y)$ de la pyramide.
- $i(X)$ et $j(X)$ définissent les coordonnées respectives du pixel situé dans le coin haut-gauche du bloc X .
- Le paramètre $T \in \mathbb{N}$ définit la tolérance sur le recouvrement.
- La largeur de la fenêtre de similarité est définie par L .

Si au moins une des quatre inégalités suivantes est vérifiée, alors Y n'est pas dans la localité de X (le test est négatif) :

1. $i(Y) \cdot 2^{niv(Y)} > i(X) + T$
2. $[i(Y) + L - 1] \cdot 2^{niv(Y)} < i(X) + L - 1 - T$
3. $j(Y) \cdot 2^{niv(Y)} > j(X) + T$
4. $[j(Y) + L - 1] \cdot 2^{niv(Y)} < j(X) + L - 1 - T$

Table 5.5 – Test d'appartenance du bloc Y à la localité du bloc X .

D'un point de vue complexité algorithmique, une recherche directe sur les niveaux de la pyramide peut éventuellement être plus avantageuse qu'une recherche dans le domaine \mathcal{D} . Ceci est surtout vrai lorsque T est faible et que le domaine contient beaucoup de blocs. Dans ce cas, en effectuant une recherche directe sur la pyramide, on remplace les tests d'appartenance sur tous les blocs du domaine par des tests de

variance (cf. section 5.4.1) sur quelques blocs situés à proximité de l'axe de résolution qui traverse le bloc X .

a. Recherche locale:

Pour chaque classe \mathcal{C}_i du domaine \mathcal{D}

1. Faire le test d'appartenance de la table 5.5 pour chaque bloc de la classe \mathcal{C}_i .
2. Pour les blocs qui rendent ce test positif, mesurer leur similarité au bloc X .

Le bloc le plus similaire est choisi comme similarité finale de X .

b. Recherche locale avec délocalisation partielle:

Pour chaque classe \mathcal{C}_i du domaine \mathcal{D}

1. Faire le test d'appartenance de la table 5.5 pour chaque bloc de la classe \mathcal{C}_i .
2. Si au moins un de ces blocs rend ce test positif, mesurer la similarité au bloc X de tous les blocs de \mathcal{C}_i .

Le bloc le plus similaire est choisi comme similarité finale de X .

Table 5.6 – Algorithme de recherche locale d'une similarité du bloc X dans le domaine de recherche classifié \mathcal{D} .

On peut délocaliser partiellement la recherche en exploitant les classes du domaine de recherche. Il s'agit de prendre en compte tous les blocs du domaine pour lesquels au moins un bloc de la classe associée vérifie le test d'appartenance de la table 5.5. L'algorithme de recherche locale d'une similarité dans le domaine avec une délocalisation partielle est donné dans la table 5.6(b).

5.7 Agrandissement par synthèse des similarités

Pour agrandir une image d'un facteur 2, nous proposons un modèle de synthèse des similarités que nous avons déjà présenté dans la section 5.2.1. Nous le détaillons dans la section 5.7.1 et nous donnons quelques exemples de synthèses réelles.

Pour effectuer un agrandissement de facteur supérieur à 2, nous proposons deux approches. Dans les deux cas, toutes les résolutions intermédiaires entre la base de la pyramide initiale et l'agrandissement souhaité sont produites. En d'autres termes, le processus d'agrandissement est itératif et à chaque itération, l'image agrandie vient compléter la pyramide pour constituer sa nouvelle base. Dans la première approche, on applique le modèle d'analyse-synthèse pour produire un agrandissement de facteur 2 de la base courante de la pyramide. Cet agrandissement constitue la nouvelle

base de la pyramide et le procédé est reproduit jusqu'à obtenir la résolution souhaitée. Avec cette approche, on quadruple le nombre de similarités à rechercher à chaque itération. Dans la deuxième approche, on recommence uniquement la phase de synthèse avec les similarités trouvées initialement par la phase d'analyse réalisée sur la pyramide de l'image originale. A chaque itération, on produit la nouvelle base de la pyramide en appliquant le modèle de synthèse sur le niveau immédiatement inférieur. Avec cette approche, on double la taille des blocs à chaque itération et le nombre de similarités demeure constant. Ces deux approches sont développées dans la section 5.7.2.

5.7.1 Agrandissement de facteur deux

Pour illustrer nos propos, reprenons la figure 5.1 de la section 5.2.1. Le bloc X , appartenant à la base de la pyramide, est le bloc père du bloc X' que l'on cherche à synthétiser. Le bloc Y est un bloc situé sur l'un des niveaux supérieurs de la pyramide (pas nécessairement sur le premier niveau) sauf la base. Nous supposons que le bloc Y est obtenu par minimisation de la mesure de similarité (cf. expression (5.2)) avec l'un des algorithmes de recherche que nous proposons dans la section 5.6. Le bloc Y est donc lié au bloc X par la relation $\tau_{(p,q)} \circ \theta_k(Y) + \varepsilon = X$ où la transformation $\tau_{(p,q)} \circ \theta_k$ est complètement spécifiée par la minimisation effectuée dans l'étape de recherche. Le bloc résiduel ε de la transformation est donné par la différence suivante :

$$\varepsilon = X - \tau_{(p,q)} \circ \theta_k(Y) \quad (5.14)$$

Le bloc fils de X est synthétisé en appliquant la transformation au bloc fils de Y et en interpolant le bloc ε avec un opérateur I :

$$X' = \tau_{(p,q)} \circ \theta_k(Y') + I(\varepsilon) \quad (5.15)$$

L'application de la transformation spatiale θ_k ne pose pas de problème particulier. Il suffit d'effectuer l'isométrie définie par (5.4) qui est indiquée par la valeur de k . L'application de la transformation massique $\tau_{(p,q)}$ nécessite un sur-échantillonnage de facteur 2 du polynôme fixe pour prendre en compte le changement de résolution. De plus, lorsque le filtre qui a servi à construire la pyramide est de largeur paire, il faut sur-échantillonner avec un déphasage de $-\frac{1}{4}$ dans chaque direction. Ainsi, lorsque le filtre a une largeur impaire, la transformation massique du bloc $Z = \theta_k(Y')$ est donnée par :

$$\tau_{(p,q)}(Z_{i,j}) = \sum_{n=1}^p a_n Z_{i,j}^n + \sum_{n=0}^q b_n \left(\frac{i}{2}\right)^n + c_n \left(\frac{j}{2}\right)^n \quad (5.16)$$

Lorsque le filtre a une largeur paire, la transformation massique est donnée par :

$$\tau_{(p,q)}(Z_{i,j}) = \sum_{n=1}^p a_n Z_{i,j}^n + \sum_{n=0}^q b_n \left(\frac{2i-1}{4}\right)^n + c_n \left(\frac{2j-1}{4}\right)^n \quad (5.17)$$

Pour l'opérateur d'interpolation I dans (5.15), nous avons choisi la transformée B-spline cubique étudiée dans la section 2.3.3. Cette méthode d'interpolation permet

d'approcher la méthode d'interpolation idéale du *zero-padding* sans introduire d'effet d'ondulations lorsque la B-spline est cubique. L'opérateur I agrandit donc le résidu ε en préservant ses fréquences.

La synthèse d'une similarité à une résolution supérieure est donc réalisée en supposant que les paramètres de la transformation $\tau_{(p,q)} \circ \theta_k$ ne changent pas et que le résidu de transformation initial est préservé. D'un point de vue pratique, l'ajout du terme $I(\varepsilon)$ dans l'expression (5.15) est très important, il permet de corriger globalement l'erreur (ou résidu) de similarité qui existe entre les blocs pères X et Y .

La figure 5.13 illustre quelques exemples de synthèses de similarités. Nous reprenons ici la pyramide à deux niveaux de la figure 5.4. Le bloc X appartient à la base et le bloc Y au premier niveau. La figure 5.13(a) montre le cas où le modèle de similarité est basé uniquement sur une transformation massique linéaire $\tau_{(1,0)}$. On voit dans cet exemple que la transformation rehausse légèrement le contraste du bloc Y . On peut remarquer visuellement une certaine différence entre X et $\tau_{(1,0)}(Y)$: il s'agit du résidu ε de la transformation. Le bloc fils du bloc Y , c'est-à-dire le bloc Y' , est situé sur la base de la pyramide. La transformation $\tau_{(1,0)}(Y)$ rehausse également son contraste. On voit que l'ajout du résidu interpolé $I(\varepsilon)$ permet de corriger l'erreur qui existe entre X et $\tau_{(1,0)}(Y)$. La correction intervient principalement dans le triangle inférieur du bloc X' .

La figure 5.13(b) illustre le cas d'un modèle de similarité plus complexe avec une transformation massique $\tau_{(3,2)}$. Les blocs X et Y sont exactement les mêmes que dans l'exemple de la figure 5.13(a). La mesure de similarité donne une valeur beaucoup plus petite que dans le premier cas : la transformation $\tau_{(3,2)}$ permet de déformer davantage le bloc Y . Le résidu est donc beaucoup plus faible et la différence entre X et Y est difficilement perceptible. Si maintenant on considère le bloc fils X' , on peut observer que le résidu interpolé $I(\varepsilon)$ apporte très peu de modifications.

On peut faire à ce niveau une observation importante à partir des deux exemples précédents, qui sera confirmée par les résultats expérimentaux de la section 5.8. L'ajout du résidu interpolé $I(\varepsilon)$ dans la synthèse du bloc fils X' apporte très peu de modifications lorsque la fonction de transformation est suffisamment complexe. Par contre, plus celle-ci est simple, plus la correction apportée par $I(\varepsilon)$ est importante. En fait, l'ajout de ce terme permet de compenser la simplicité de la transformation. On peut voir facilement avec les deux exemples des figures 5.13(a) et 5.13(b) que le résultat de la synthèse réalisée avec une transformation simple est finalement rendu similaire à celui d'une transformation complexe par l'ajout du résidu $I(\varepsilon)$. Cette observation incite à penser qu'il est préférable d'utiliser une transformation simple pour deux raisons. La première est que la complexité algorithmique est moins importante : l'inversion de la matrice A définie par l'expression (5.7) est plus rapide. La deuxième est qu'une similarité trouvée avec une fonction de transformation simple est souvent visuellement plus significative.

La figure 5.13(c) illustre un cas de synthèse réalisée avec une transformation linéaire $\tau_{(1,0)}$ où l'on permet les isométries du bloc Y le but d'augmenter la probabilité de trouver une bonne similarité. Encore une fois, on observe l'importance de l'ajout de $I(\varepsilon)$ dans la synthèse de X' pour rattraper le résidu de similarité qui existe entre

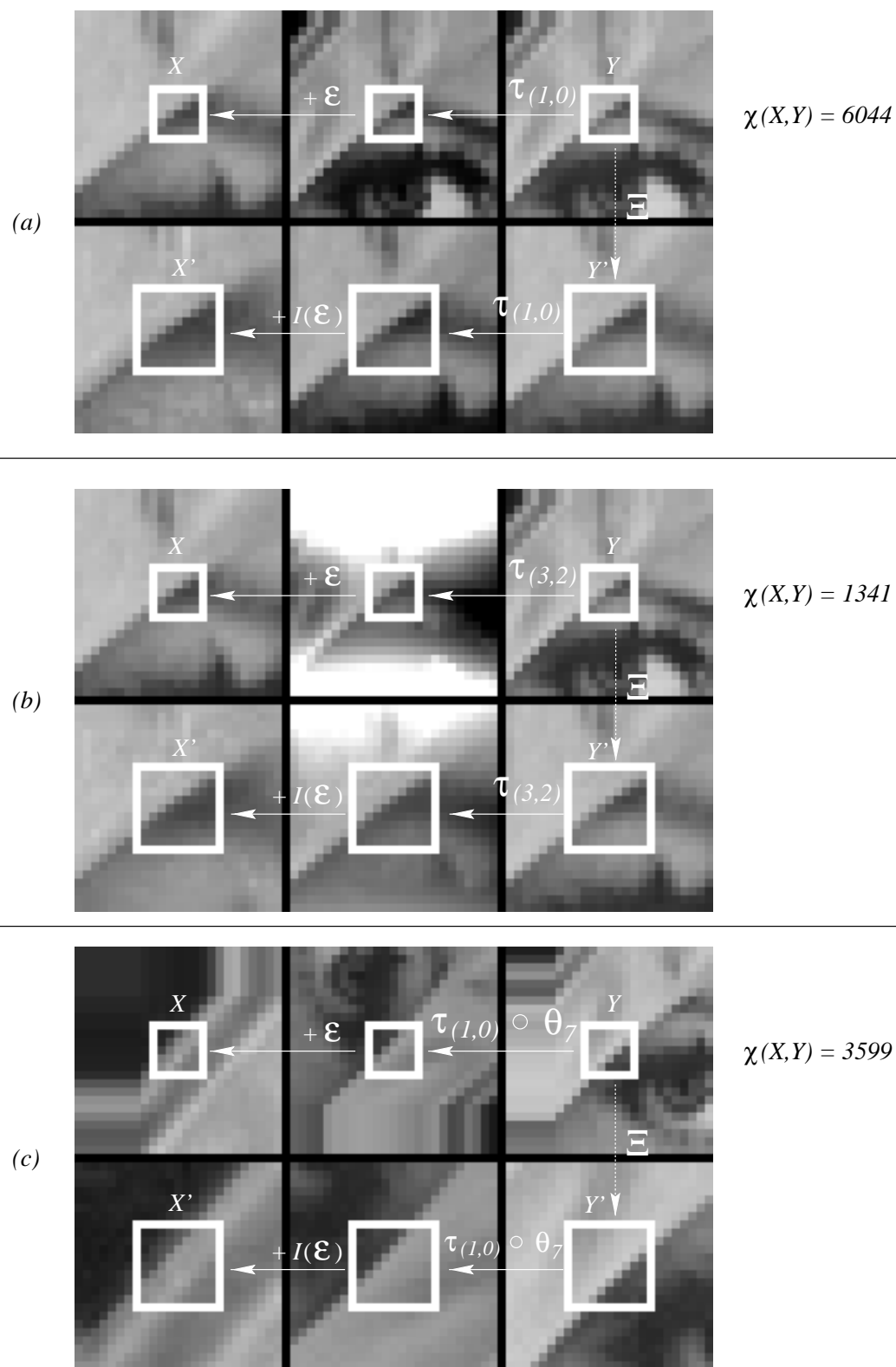


Figure 5.13 – Exemples de synthèses de similarités réalisées à partir de la pyramide à 2 niveaux de la figure 5.4, avec différentes fonctions de transformation : (a) transformation linéaire $\tau_{(1,0)}$, (b) transformation $\tau_{(3,2)}$, (c) transformation linéaire $\tau_{(1,0)}$ avec isométrie (réflexion autour de l'axe diagonal $+45^\circ$).

X et Y .

Nous avons présenté le modèle de synthèse d'un bloc fils en considérant le cas particulier où son support était le même que celui de son bloc père. Dans ce cas, le niveau de super-résolution entier peut être obtenu en partitionnant la base de la pyramide en blocs pères et en synthétisant les blocs fils avec le principe décrit précédemment.

De manière plus générale, ce n'est pas la base de la pyramide qui doit être partitionnée mais le niveau de super-résolution. Les supports des blocs fils doivent respecter les contraintes données par le partitionnement choisi dans l'étape 3 de l'algorithme général figurant dans la table 5.1. Ces supports sont soit égaux soit plus petits que ceux des blocs pères. Dans tous les cas, on peut supposer en premier lieu que le support du bloc fils est le même que celui du bloc père, puis faire éventuellement une restriction du support du bloc fils afin de respecter le partitionnement du niveau de super-résolution. Ceci revient tout simplement à prendre en compte un sous-ensemble des pixels du blocs fils.

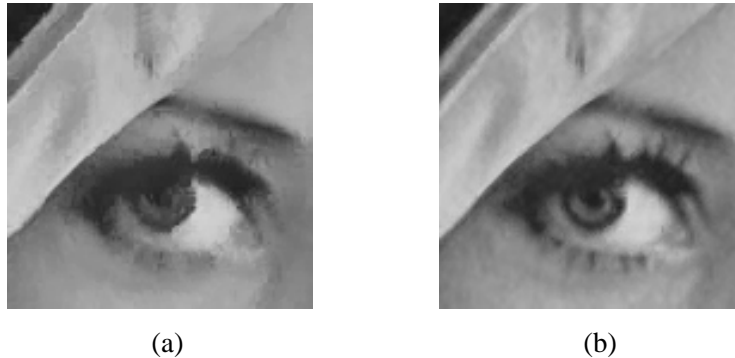


Figure 5.14 – Agrandissement de facteur 2 à partir de la pyramide à 2 niveaux de la figure 5.4. La synthèse des similarités est réalisée: (a) sans ajouter les résidus de similarités, (b) en ajoutant les résidus de similarités.

Nous montrons sur la figure 5.14 un exemple de synthèse complète du niveau de super-résolution de la pyramide à deux niveaux illustrée sur la figure 5.4. Dans cet exemple, les blocs pères ont une taille de 5×5 pixels et le partitionnement du niveau de super-résolution est illustré sur la figure 5.3. La fonction de transformation utilisée par le modèle de similarité est la fonction linéaire $\tau_{(1,0)}$ et la recherche des similarités est réalisée sans contrainte particulière sur le premier niveau de la pyramide. La figure 5.14(a) montre le cas où l'on ne prend pas en compte la correction apportée par l'ajout des résidus de similarité interpolés, tandis que la figure 5.14(b) illustre le cas où ceux-ci sont pris en compte. L'agrandissement obtenu dans le second cas montre une qualité nettement supérieure à celui obtenu dans le premier cas. L'intérêt pratique d'ajouter $I(\varepsilon)$ dans l'expression (5.15) se confirme par conséquent d'un point de vue global.

5.7.2 Agrandissement de facteur supérieur à deux

On suppose qu'une première phase d'analyse-synthèse de la pyramide initiale a eu lieu afin de produire un premier niveau de super-résolution qui double la résolution de la base de la pyramide (cf. section précédente). Le niveau de super-résolution est intégré dans la pyramide et constitue désormais sa nouvelle base.

Pour produire un agrandissement de facteur supérieur à deux, il faut synthétiser à nouveau un niveau de super-résolution qui double la base de la pyramide, l'intégrer dans la pyramide comme sa nouvelle base, puis recommencer jusqu'au facteur d'agrandissement souhaité. Pour synthétiser le niveau de super-résolution à chaque itération, on peut envisager deux approches.

Première approche : analyse-synthèse

La première approche est la plus simple à mettre en œuvre. Elle consiste à doubler dans chaque direction la partition du niveau de super-résolution à chaque itération : la taille en pixels des blocs fils et des blocs pères est constante, mais leur nombre quadruple à chaque fois. Cette approche nécessite de recommencer à chaque nouvelle itération la phase d'analyse pour rechercher dans le domaine \mathcal{D} des similarités aux blocs pères. Pour éviter d'augmenter démesurément la complexité algorithmique, le domaine \mathcal{D} n'est pas reconstruit. En d'autres termes, les similarités sont toujours recherchées sur la pyramide initiale construite à partir de l'image originale.

La synthèse des similarités avec cette approche obéit donc au même schéma que celui présenté dans la section 5.7.1.

Deuxième approche : synthèse

La seconde approche est un peu plus complexe à mettre en œuvre, mais elle est beaucoup plus rapide puisque la phase d'analyse y est supprimée. Dans cette approche, on suppose que les relations de similarité trouvées pour les blocs pères lors de la première analyse sont toujours vérifiées pour leurs blocs fils aux niveaux de résolution supérieurs. Selon cette hypothèse, on synthétise à chaque itération le niveau de super-résolution avec un partitionnement identique à celui utilisé lors de la première analyse. En d'autres termes, les blocs fils ont toujours le même support, mais leur nombre de pixels quadruple à chaque itération.

Pour décrire de manière plus formelle le modèle de synthèse utilisé, nous nous intéressons à la synthèse d'un bloc fils particulier en reprenant les conditions initiales décrites au début de la section 5.7.1. Le bloc X appartient toujours à la base de la pyramide initiale (image originale) et le bloc Y est un bloc similaire lié à X par la relation $\tau_{(p,q)} \circ \theta_k(Y) + \varepsilon = X$. Le bloc résiduel de la similarité ε est toujours donné par l'expression (5.14). En supposant que le facteur d'agrandissement souhaité est 2^M (dans chaque direction), il faut produire M niveaux de super-résolution. La synthèse du bloc fils de X au m -ième niveau de super-résolution est donnée par la relation suivante :

$$(0 < m \leq M) \quad X^{\uparrow 2^m} = \tau_{(p,q)} \circ \theta_k(Y^{\uparrow 2^m}) + I^{\uparrow 2^m}(\varepsilon) \quad (5.18)$$

où la notation $\uparrow 2^m$ symbolise un agrandissement de facteur 2^m . L'opérateur d'interpolation $I^{\uparrow 2^m}$ est toujours celui de la méthode d'interpolation par transformée B-spline cubique. Comme le montre la relation (5.18), la synthèse de $X^{\uparrow 2^m}$ dépend en partie du bloc $Y^{\uparrow 2^m}$ qui est un bloc fils de Y . Tant que $Y^{\uparrow 2^m}$ appartient à la pyramide initiale, on peut synthétiser $X^{\uparrow 2^m}$ sans difficulté particulière puisque $Y^{\uparrow 2^m}$ est connu. Par contre, lorsque le bloc $Y^{\uparrow 2^m}$ n'appartient plus à la pyramide initiale, il faut le calculer en synthétisant les blocs qui le recouvrent. On voit qu'en augmentant le facteur d'agrandissement M , une récursivité se met en place de telle sorte que la synthèse d'un bloc dépend de la synthèse d'un deuxième, qui dépend lui-même de la synthèse d'un troisième, etc. En conclusion, plus l'agrandissement est grand et plus l'hypothèse de préservation des similarités initiales se fragilise par la propagation des erreurs. Nous verrons à l'aide de résultats expérimentaux (section 5.8) que cette conclusion se traduit en pratique par une dégradation de la qualité de l'agrandissement qui est proportionnelle au facteur d'agrandissement.

L'application de la transformation spatiale θ_k dans l'expression (5.18) ne pose pas de problème particulier. Comme dans le modèle de synthèse d'un agrandissement de facteur 2, il suffit de réaliser l'isométrie indiquée par k (cf. expression (5.4)).

L'application de la transformation massique $\tau_{(p,q)}$ nécessite un sur-échantillonnage de facteur 2^m du polynôme fixe. Ainsi, lorsque le filtre utilisé pour construire la pyramide est de largeur impaire, la transformation massique du bloc $Z = \theta_k(Y^{\uparrow 2^m})$ est donnée par :

$$\tau_{(p,q)}(Z_{i,j}) = \sum_{n=1}^p a_n Z_{i,j}^n + \sum_{n=0}^q b_n \left(\frac{i}{2^m}\right)^n + c_n \left(\frac{j}{2^m}\right)^n \quad (5.19)$$

Lorsque le filtre a une largeur paire, il faut sur-échantillonner le polynôme fixe en introduisant un déphasage de $\frac{1-2^m}{2^{m+1}}$. Dans ce cas, la transformation massique est donnée par :

$$\tau_{(p,q)}(Z_{i,j}) = \sum_{n=1}^p a_n Z_{i,j}^n + \sum_{n=0}^q b_n \left(\frac{2i+1-2^m}{2^{m+1}}\right)^n + c_n \left(\frac{2j+1-2^m}{2^{m+1}}\right)^n \quad (5.20)$$

Dans la présentation du modèle de synthèse précédent, nous avons supposé que les supports des blocs fils $X^{\uparrow 2^m}$ ($0 < m < M$) étaient identiques à ceux de leur bloc père X . En pratique, les supports des blocs fils doivent respecter le partitionnement des niveaux de super-résolution choisi au départ (cf. étape 3 de l'algorithme de la table 5.1). Pour cela, il suffit de faire une restriction au véritable support du bloc fils qui est indiqué par la partition.

5.8 Résultats expérimentaux

Cette section est consacrée à l'étude expérimentale de notre méthode d'agrandissement. Cette étude s'intéresse tout d'abord aux paramètres de la méthode, et notamment aux effets obtenus sur le résultat d'un agrandissement de facteur 2 en

faisant varier ces différents paramètres. Ceci nous permet de mieux comprendre le rôle de chacun d'entre eux et nous aide ainsi à choisir plus facilement leur valeur. Pour faciliter cette étude, nous commençons dans la section 5.8.1 par fixer une configuration standard de l'ensemble des paramètres. Ensuite, dans la section 5.8.2, nous les faisons varier en nombre réduit, souvent de manière unique, mais toujours dans le contexte de la configuration standard que nous avons définie. Nous nous intéressons surtout aux effets générés par la variation de ces paramètres. Il semble raisonnable de supposer que ces effets sont communs à une majorité d'images.

La section 5.8.3 est consacrée à la validation de notre méthode d'agrandissement. Nous étudions la qualité de l'information produite par le modèle de synthèse. Nous étudions en particulier dans quelle mesure l'hypothèse de préservation idéale des similarités exploitée pour créer le niveau de super-résolution de la pyramide est validée en pratique. Pour cela, nous utilisons une image de référence et l'agrandissement de facteur 2 est réalisé à partir de la réduction de cette image de référence.

Dans la section 5.8.4, nous montrons des agrandissements de facteur supérieur à deux. Nous comparons en particulier les deux approches que nous avons proposées dans la section 5.7.2.

Enfin, la section 5.8.5 compare la qualité d'agrandissements obtenus par notre méthode à ceux produits par la transformée B-spline (chapitre 2) et le filtrage médian (chapitre 3).

5.8.1 Configuration standard des paramètres

Les différents paramètres de notre méthode d'agrandissement sont présentés dans la table 5.7. Pour les étudier, nous utilisons deux images différentes de 64×64 pixels. La première est une image simple extraite de l'image de Lenna de 512×512 pixels. La deuxième image est un peu plus complexe, elle est extraite de l'image du cameraman de 256×256 pixels.

Certains de ces paramètres dépendent beaucoup des caractéristiques de l'image à agrandir. C'est principalement le cas de l'écart-type minimum d'un bloc pour sélectionner dans le domaine \mathcal{D} et aussi celui de l'écart-type minimum d'un bloc pour exécuter une recherche de similarité. Nous adaptons ces deux paramètres à chaque image traitée, les autres paramètres étant communs.

Concernant la pyramide, nous choisissons d'utiliser une pyramide de quatre niveaux et d'utiliser le filtre de Burt de taille 5×5 avec $\beta = 0.625$ (cf. section 4.3.2). La comparaison expérimentale entre différents filtres a montré la supériorité du filtre 5×5 de Burt, tant au niveau de la complexité algorithmique qu'au niveau de la qualité visuelle par l'absence d'ondulations et de déphasage (cf. section 4.3.4).

Le filtre choisi pour construire la pyramide étant un filtre de largeur impaire, les blocs du domaine de recherche \mathcal{D} ont donc une taille de 5×5 pixels. Ceux-ci sont sélectionnés à partir de la pyramide de différences. Ceci permet de faire la sélection des blocs en fonction de leur contenu fréquentiel (cf. section 4.3.5). Un bloc est sélectionné si son écart-type est supérieur à 4 pour l'image de Lenna et à 10 pour l'image du cameraman. Chaque valeur est fixée pour avoir un nombre suffisant de

1. **La pyramide**
 - Nombre de niveaux (*entier* > 1).
 - Choix d'un filtre et en particulier de sa parité.
2. **Le domaine de recherche \mathcal{D}**
 - Sélection des blocs avec une pyramide de différences (*oui/non*).
 - Ecart-type minimum d'un bloc du domaine (*réel* > 0).
 - Classification du domaine (*oui/non*): choix d'un angle-limite (*réel* > 0) entre les plans de transformations.
3. **La recherche des similarités**
 - Choix du modèle de similarité: degré p du polynôme d'échelle, degré q du polynôme fixe et isométries permises (*identité, rotations* ou *rotations+réflexions*).
 - Ecart-type minimum d'un bloc pour exécuter une recherche (*réel* > 0): calcul à partir de la pyramide de différences (*oui/non*).
 - Profondeur de la recherche dans le domaine \mathcal{D} (*pourcentage*).
 - Modèle de recherche (*local* ou *global*) et choix de la tolérance T sur le recouvrement du bloc père dans le cas d'une recherche locale.
4. **Le partitionnement du niveau de super-résolution**

Table 5.7 – Les différents paramètres à choisir pour agrandir une image par synthèse de similarités sur une pyramide.

blocs dans le domaine \mathcal{D} . Par la suite, nous serons parfois amenés à utiliser un filtre à largeur paire pour construire la pyramide. Un tel filtre introduit un déphasage sur la pyramide de différences qui génère des écart-types de blocs beaucoup plus importants (voir figure 5.17(a)). Pour tenir compte de ce phénomène, il faut redéfinir l'écart-type minimum σ_{min} d'un bloc du domaine \mathcal{D} lorsque le filtre a une largeur paire. Dans ce cas, nous prendrons $\sigma_{min} = 9$ pour Lenna et $\sigma_{min} = 18$ pour le cameraman. Par défaut, le domaine \mathcal{D} n'est pas organisé en classes. Les blocs du domaine sont seulement ordonnés selon leur variance (cf. section 5.4.1).

En ce qui concerne la recherche des similarités, on utilise le modèle de recherche global avec l'algorithme d'exploration du domaine \mathcal{D} basé sur la variance des blocs qui est décrit dans la table 5.4. Cet algorithme est défini en exploitant les représentants des différentes classes du domaine. Le domaine \mathcal{D} n'étant pas classifié dans la configuration standard des paramètres, nous supposerons que chaque bloc du domaine définit une classe dont il est le représentant. La profondeur de recherche w , qui s'exprime comme un pourcentage du nombre de blocs présents dans le domaine \mathcal{D} , est fixée à 50%. La recherche d'une similarité est réalisée uniquement pour les

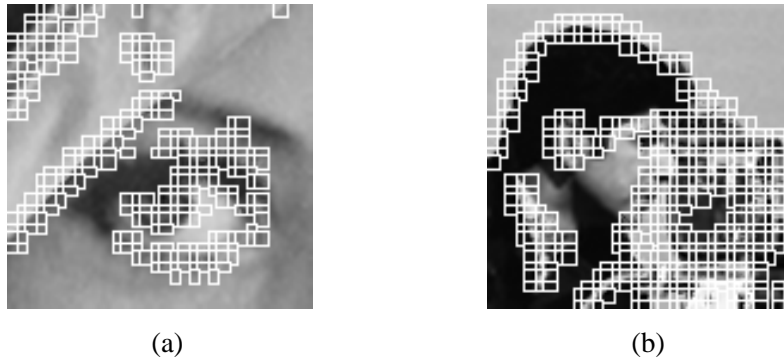


Figure 5.15 – Localisation des blocs synthétisés par similarité dans les images de Lenna et du cameraman en prenant respectivement $\sigma_{min} = 3$ et $\sigma_{min} = 8$.

blocs qui ont un écart-type supérieur ou égal à 3 pour l'image de Lenna et à 8 pour l'image du cameraman. Celui-ci est calculé à partir de la pyramide de différences afin de prendre en compte le contenu fréquentiel des blocs. Ces deux seuils ont été fixés pour que les blocs synthétisés sur le niveau de super-résolution recouvrent les zones de la base de la pyramide de différences qui ont des variations d'intensité non négligeables (voir les pyramides de différences illustrées sur la figure 5.16). Les figures 5.15(a) et 5.15(b) illustrent la localisation des blocs qui sont synthétisés par des similarités avec les seuils choisis pour l'image de Lenna et celle du cameraman. Comme nous l'avons déjà expliqué précédemment, lorsque le filtre a une largeur paire, l'écart-type σ_{min} d'un bloc est beaucoup plus grand sur la pyramide des différences. Dans ce cas, nous prendrons $\sigma_{min} = 6$ pour Lenna et $\sigma_{min} = 14$ pour le cameraman. Le modèle de similarité utilisé par défaut est basé sur une transformation massique linéaire $\tau_{(1,0)}$. De plus, les isométries permises sont l'identité et les rotations des blocs.

Enfin, le partitionnement du niveau de super-résolution est celui qui est illustré sur la figure 5.3. C'est le seul partitionnement que nous avons défini pour les filtres à largeur impaire. En fait, le changement de partition sera étudié dans le cadre d'un filtre à largeur paire avec les trois partitions illustrées sur la figure 5.2.

La figure 5.16 illustre la pyramide passe-bas, la pyramide de différences, le domaine de recherche \mathcal{D} et l'image agrandie qui sont obtenus pour l'image de Lenna et celle du cameraman, ceci avec la configuration standard des paramètres que nous avons définie précédemment.

5.8.2 Etude des paramètres

Dans cette section, nous faisons varier certains paramètres de la configuration standard et nous étudions leur influence sur le résultat de l'agrandissement.

La profondeur de la pyramide

Le premier paramètre lié à la pyramide est le nombre de ses niveaux. On peut en particulier s'interroger sur la localisation des similarités trouvées. Est-ce que ces



Figure 5.16 – Agrandissement de facteur 2 des images de Lenna et du cameraman par synthèse de similarités sur une pyramide. Les paramètres utilisés sont ceux de la configuration standard. Les niveaux de gris de la pyramide de différences sont amplifiés d'un facteur 2.

similarités sont réparties équitablement sur l'ensemble des niveaux?

Tout d'abord, la table 5.8 indique le nombre de blocs du domaine de recherche \mathcal{D} qui appartiennent à chaque niveau de la pyramide. Le deuxième nombre indique à chaque fois le nombre maximum de blocs qui peuvent être extraits sur le niveau correspondant. On voit en particulier qu'en montant dans la pyramide, le rapport entre les deux nombres tend vers 1. Ceci est normal puisqu'en montant dans la pyramide, la taille du niveau diminue tandis que celle des blocs est constante.

Image	Niveau 1	Niveau 2	Niveau 3
<i>Lenna</i>	505 / 784	141 / 144	16 / 16
<i>Cameraman</i>	572 / 784	141 / 144	16 / 16

Table 5.8 – Niveau d'origine des blocs du domaine de recherche \mathcal{D} .

La table 5.9 montre quant à elle le nombre de similarités détectées sur chaque niveau. Au vu de ces résultats, il apparaît clairement que les similarités sont essentiellement trouvées sur le premier niveau, un peu sur le deuxième et pratiquement pas sur le troisième. Cependant, ce constat est parfaitement normal puisque la table 5.8 montre que la majorité des blocs du domaine de recherche est issue du premier niveau.

Image	Niveau 1	Niveau 2	Niveau 3
<i>Lenna</i>	265	11	2
<i>Cameraman</i>	418	78	5

Table 5.9 – Niveau d'origine des similarités.

Pour savoir si la répartition des similarités sur l'ensemble des niveaux est équitable, il faut considérer les résultats de la table 5.9 relativement à la quantité de blocs du domaine \mathcal{D} associée à chaque niveau. Pour cela, nous définissons un indice P_k pour chaque niveau k de la manière suivante :

$$P_k = \frac{S_k \sum_i D_i}{D_k \sum_i S_i} \quad (5.21)$$

où D_k est le nombre de blocs du domaine \mathcal{D} qui appartiennent au niveau k et S_k est le nombre de similarités détectées sur le niveau k . Les résultats sont présentés dans la table 5.10. Pour l'image de Lenna, on peut observer que les niveaux 2 et 3 ne sont pas vraiment exploités dans la recherche des similarités. Pour l'image du cameraman, la répartition est plus équitable. Cependant, dans les deux cas, on constate que plus on monte dans la pyramide et plus la probabilité qu'un bloc soit similaire à un bloc de l'image originale diminue. Ceci semble indiquer que la recherche de similarités dans les résolutions faibles de l'image originale (niveaux supérieurs de la pyramide) est généralement inutile. Cette conclusion est surtout vraie pour l'image de Lenna. Pour l'image du cameraman, la meilleure répartition des similarités sur les niveaux indique qu'il y a tout de même un intérêt à effectuer des recherches de similarité sur les niveaux supérieurs de la pyramide. En fait, cette meilleure répartition s'explique par les contours de la tête du cameraman qui ont des propriétés remarquables de similarité à tous les niveaux de la pyramide (cf. figure 5.16).

Image	Niveau 1	Niveau 2	Niveau 3
<i>Lenna</i>	1.25	0.19	0.30
<i>Cameraman</i>	1.06	0.80	0.45

Table 5.10 – Indice de répartition des similarités sur les niveaux de la pyramide.

Le filtre de la pyramide

Le second paramètre lié à la pyramide est le filtre permettant de la construire. Comme nous l'avons déjà dit précédemment dans le choix de la configuration standard des paramètres, le filtre 5×5 de Burt avec $\beta = 0.625$ est le filtre qui présente les meilleures propriétés pour construire une pyramide sans introduire d'ondulations et de déphasage (cf. étude expérimentale de la section 4.3.4). On peut cependant s'interroger sur l'influence de l'utilisation d'un filtre à largeur paire sur le résultat de l'agrandissement. Dans ce cas, la fenêtre de similarité est de taille 4×4 pixels. On peut supposer qu'avec cette taille les similarités sur la pyramide sont plus faciles à trouver qu'avec celle de 5×5 pixels associée au filtre de largeur impaire. Cependant, on a également plus de chance d'avoir des similarités moins significatives puisque l'espace de transformation se réduit.

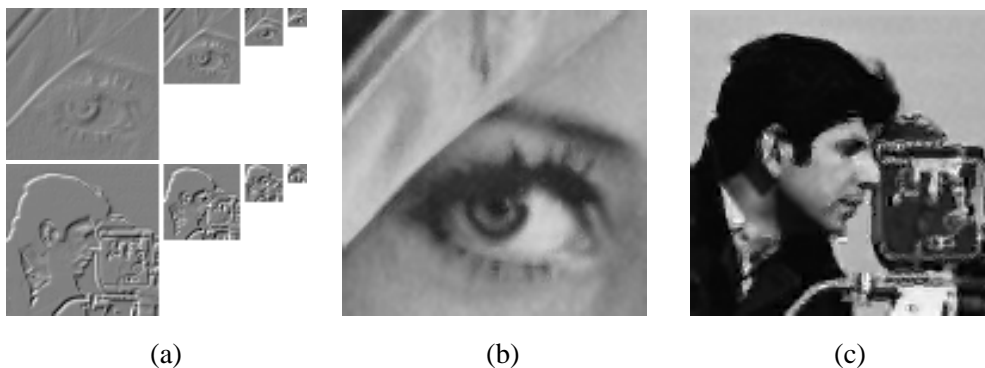


Figure 5.17 – Utilisation d'un filtre à largeur paire: (a) pyramides de différences, (b) agrandissement de l'image de Lenna, (c) agrandissement de l'image du cameraman.

Les figures 5.17(b) et 5.17(c) illustrent les agrandissements des images de Lenna et du cameraman, obtenus dans le contexte où le filtre est de largeur paire. Nous avons utilisé ici le filtre 4×4 de Burt avec $\alpha = 0.562$ (cf. section 4.3.2). Les agrandissements obtenus semblent de moins bonne qualité que ceux obtenus avec le filtre 5×5 de Burt sur la figure 5.16, surtout pour l'agrandissement du cameraman. En fait, la dégradation s'explique essentiellement par le déphasage introduit par le filtre de largeur paire qui dénature les structures, et en particulier les contours. Ce déphasage est particulièrement visible dans les pyramides de différences présentées sur la figure 5.17(a).

Construction du domaine de recherche \mathcal{D}

Pour construire le domaine de recherche \mathcal{D} , il faut sélectionner des blocs de la pyramide. Cette sélection est liée à deux paramètres, le premier est l'écart-type minimum que doit avoir un bloc pour faire partie du domaine \mathcal{D} et le second est le choix de la pyramide utilisée pour calculer l'écart-type des blocs. Nous avons vu dans la section 5.4.2 que l'utilisation de la pyramide de différences pour effectuer ce calcul permettait de sélectionner des blocs qui contenaient des hautes fréquences. C'est une

propriété nécessaire pour pouvoir prolonger le spectre initial lors de la synthèse des similarités. La valeur de l'écart-type minimum d'un bloc est donc liée à l'importance de ses hautes fréquences. Cette valeur agit sur le nombre de blocs présents dans le domaine \mathcal{D} . Elle doit être assez élevée pour éliminer les blocs qui ne contiennent pas de hautes fréquences. Cependant, en prenant une valeur trop grande, on diminue la probabilité de trouver des similarités par la suite. Le choix de cette valeur naît d'un compromis entre ces deux contraintes. Pour la partie expérimentale, nous renvoyons le lecteur à l'exemple présenté dans la section 5.4.

Classification du domaine de recherche \mathcal{D}

Les blocs du domaine de recherche \mathcal{D} peuvent être classifiés en utilisant l'algorithme développé dans la section 5.5. Pour cela, il faut choisir un angle-limite α_{max} entre les plans de transformation. Cet angle agit sur l'importance du regroupement des blocs et, par conséquent, sur le nombre de classes du domaine. On peut supposer que lorsque le nombre de classes diminue, le temps de recherche des similarités et la qualité des similarités diminuent également. Pour le moment, nous étudions expérimentalement ces différents aspects avec l'image du cameraman.

Les figures 5.18(a) et 5.18(b) illustrent respectivement l'évolution du nombre de classes en fonction de l'angle α_{max} et le temps (en secondes) requis pour classer

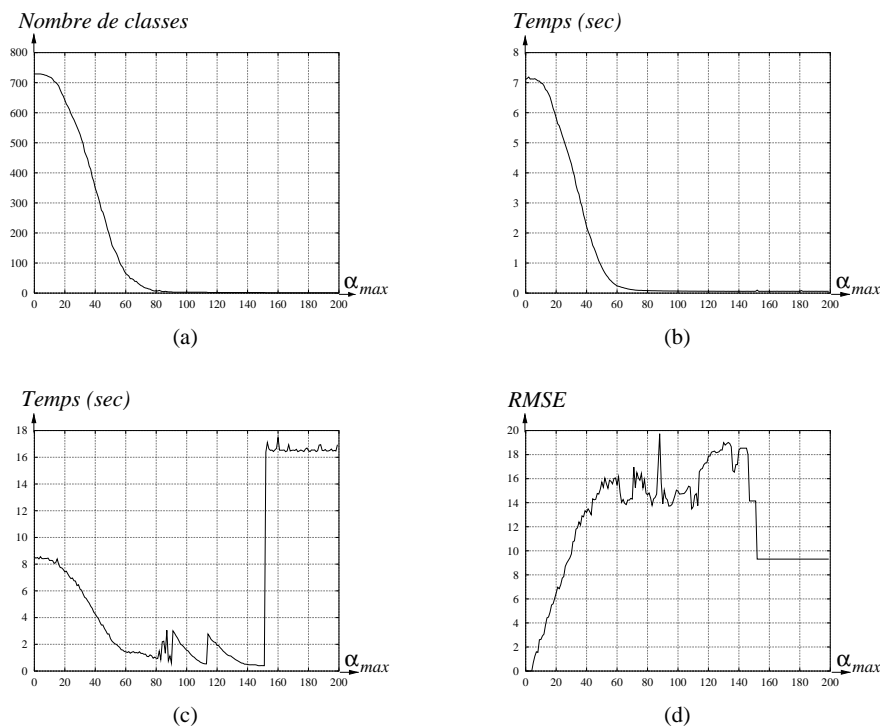


Figure 5.18 – Classification du domaine de recherche pour l'image du cameraman. Influence de l'angle α_{max} (en degrés) sur : (a) le nombre de classes du domaine, (b) le temps mis pour classer le domaine, (c) le temps total mis pour rechercher les similarités, (d) l'erreur $RMSE$ entre l'agrandissement obtenu avec classification et celui obtenu sans classification.

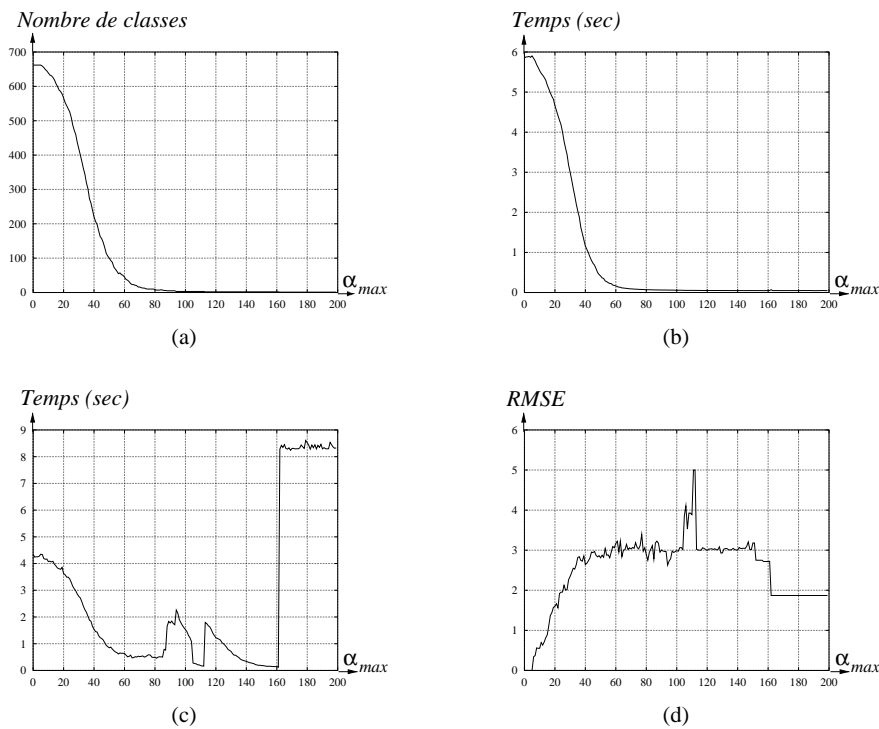


Figure 5.19 – Classification du domaine de recherche pour l’image de Lenna. Influence de l’angle α_{max} (en degrés) sur : (a) le nombre de classes du domaine, (b) le temps mis pour classifier le domaine, (c) le temps total mis pour rechercher les similarités, (d) l’erreur $RMSE$ entre l’agrandissement obtenu avec classification et celui obtenu sans classification.

les blocs du domaine. Dans les deux cas, on trouve une courbe qui obéit à une loi gaussienne.

La figure 5.18(c) montre le temps mis pour rechercher l’ensemble des similarités dans le domaine de recherche. Globalement, la courbe semble également obéir à une loi gaussienne. On peut cependant observer une petite zone d’oscillations lorsque α_{max} est compris entre 80° et 86° , puis deux «décrochements» très nets pour $\alpha_{max} = 91^\circ$ et $\alpha_{max} = 114^\circ$, et enfin un «plateau» lorsque $\alpha_{max} > 151^\circ$. La zone d’oscillations correspond à une zone d’instabilité du nombre de classes, celui-ci varie dans l’intervalle $[5, 9]$ de manière chaotique. Le premier décrochement correspond au passage de 4 à 3 classes et le deuxième à celui de 3 à 2 classes. Enfin, le plateau final apparaît avec la classe unique. Dans ce cas, le temps de recherche des similarités nécessaire est deux fois plus important que lorsque le nombre de classes est le même que celui des blocs du domaine. En effet, dans le second cas, on bénéficie d’une profondeur de recherche de 50% tandis que dans le premier cas, la recherche se fait parmi tous les blocs de la classe unique, c’est-à-dire parmi la totalité des blocs du domaine.

Pour évaluer la qualité de l’agrandissement en fonction de l’angle α_{max} , nous utilisons l’image agrandie sans classification comme image de référence. La figure 5.18(d) montre l’erreur $RMSE$ (cf. section 3.6) générée entre l’image de référence et

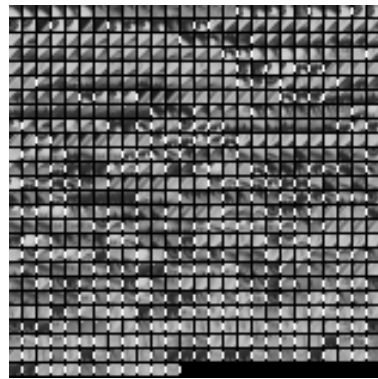
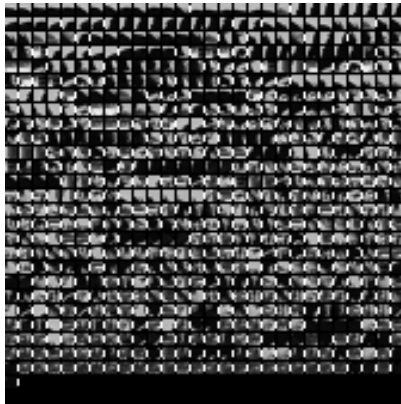
*domaine de recherche classifié**image agrandie**domaine de recherche classifié**image agrandie*

Figure 5.20 – Agrandissement de facteur 2 des images de Lenna et du cameraman en classifiant le domaine de recherche : $\alpha_{max} = 35^\circ$ pour l'image de Lenna et $\alpha_{max} = 30^\circ$ pour l'image du cameraman.

l'agrandissement obtenu en augmentant α_{max} . L'erreur croît rapidement de manière linéaire jusqu'à $\alpha_{max} \approx 50^\circ$. Ensuite, elle oscille horizontalement jusque $\alpha_{max} = 150^\circ$. Au-delà, le domaine ne contient plus qu'une seule classe et la recherche des similarités se fait sur la totalité du domaine \mathcal{D} . C'est ce qui explique la brusque chute de l'erreur.

La figure 5.19 montre les mêmes courbes d'expérimentation pour l'image de Lenna. Elles ont à peu près les mêmes propriétés que celles décrites précédemment pour l'image du cameraman.

La figure 5.20 illustre les agrandissements obtenus en classifiant les domaines de recherche. Les valeurs de l'angle-limite α_{max} ont été choisies pour réaliser un compromis intéressant entre la rapidité de recherche des similarités et la qualité de l'agrandissement. Pour l'image de Lenna, on accélère significativement la recherche des similarités sans que la dégradation du résultat de l'agrandissement ne soit vraiment perceptible. L'image du cameraman étant plus complexe, il est beaucoup plus

difficile de diminuer le nombre de classes sans trop dégrader la qualité de l'agrandissement. On voit donc, à travers ces deux exemples, que l'intérêt de la classification du domaine dépend avant tout de la complexité de l'image traitée. Il dépend également de la taille de l'image car plus elle est grande, plus les blocs sont nombreux et plus la nécessité de les regrouper en classes se fait ressentir.

Le modèle de similarité

Pour rechercher des similarités dans le domaine \mathcal{D} , il faut tout d'abord fixer les paramètres du modèle de similarité que nous avons présenté dans la section 5.3.1. Ces paramètres sont au nombre de trois.

Les deux premiers sont liés à la fonction de transformation massique $\tau_{(p,q)}$. Plus précisément, il faut fixer le degré p du polynôme d'échelle et le degré q du polynôme fixe. La figure 5.21 illustre différents agrandissements de l'image de Lenna obtenus en faisant varier le couple (p, q) dans l'intervalle $\{1, 2, 3\} \times \{0, 1, 2\}$. Pour montrer plus nettement l'influence de la transformation massique sur le résultat de l'agrandissement, l'interpolation du bloc résiduel de la transformation n'est pas ajoutée lors de la synthèse des similarités (cf. expression (5.15)). Il s'agit en quelque sorte d'agrandissements bruts. En augmentant p et en laissant $q = 0$, les contours de l'image deviennent plus saillants. Ce phénomène s'explique par le fait que plus p est grand, et plus le polynôme d'échelle a la possibilité d'amplifier le contraste du bloc traité. En augmentant q et en laissant $p = 1$, l'image devient moins bruitée et les contours semblent plus lisses, surtout pour $q = 1$. En fait, lorsque $q = 1$, le polynôme fixe se traduit par un plan d'inclinaison variable, ce qui produit des transitions plus continues. Avec $q = 2$, on ajoute une parabolöide au plan précédent, ce qui produit des transitions un peu plus brusques. La diminution du bruit observée en augmentant q est due au polynôme fixe qui prend plus d'importance que le polynôme d'échelle dans la transformation du bloc traité. Plus précisément, le bloc artificiel produit par le polynôme fixe prend plus de poids que le bloc naturel traité par le polynôme d'échelle. Le bruit qui est introduit par le bloc naturel est donc plus faible. En augmentant à la fois p et q , on bénéficie en même temps d'un meilleur contraste et d'une diminution du bruit dans l'image agrandie. C'est ce que l'on observe pour les transformations $\tau_{(2,1)}$ et $\tau_{(3,2)}$ sur la figure 5.21. Cependant, la figure 5.22 montre que les différences entre les agrandissements obtenus avec $\tau_{(1,0)}$, $\tau_{(2,1)}$ et $\tau_{(3,2)}$ sont beaucoup moins perceptibles lorsque l'interpolation du résidu de transformation est ajoutée lors de la synthèse des similarités. Il apparaît nettement que les limitations de la transformation $\tau_{(1,0)}$ sont compensées par l'utilisation du résidu de la transformation au moment de la synthèse. La conséquence est que le rapport «qualité/quantité de calculs» est beaucoup plus intéressant avec $\tau_{(1,0)}$ qu'avec $\tau_{(2,1)}$ ou $\tau_{(3,2)}$. D'un point de vue pratique, il semble donc préférable d'utiliser la transformation $\tau_{(1,0)}$ dès lors que les résidus de transformation sont exploités dans la synthèse des similarités. Cette conclusion est renforcée par le fait que l'utilisation de polynômes de degrés élevés introduit parfois une instabilité dans la synthèse des similarités. Ce phénomène se traduit par l'apparition d'un bruit assez gênant que

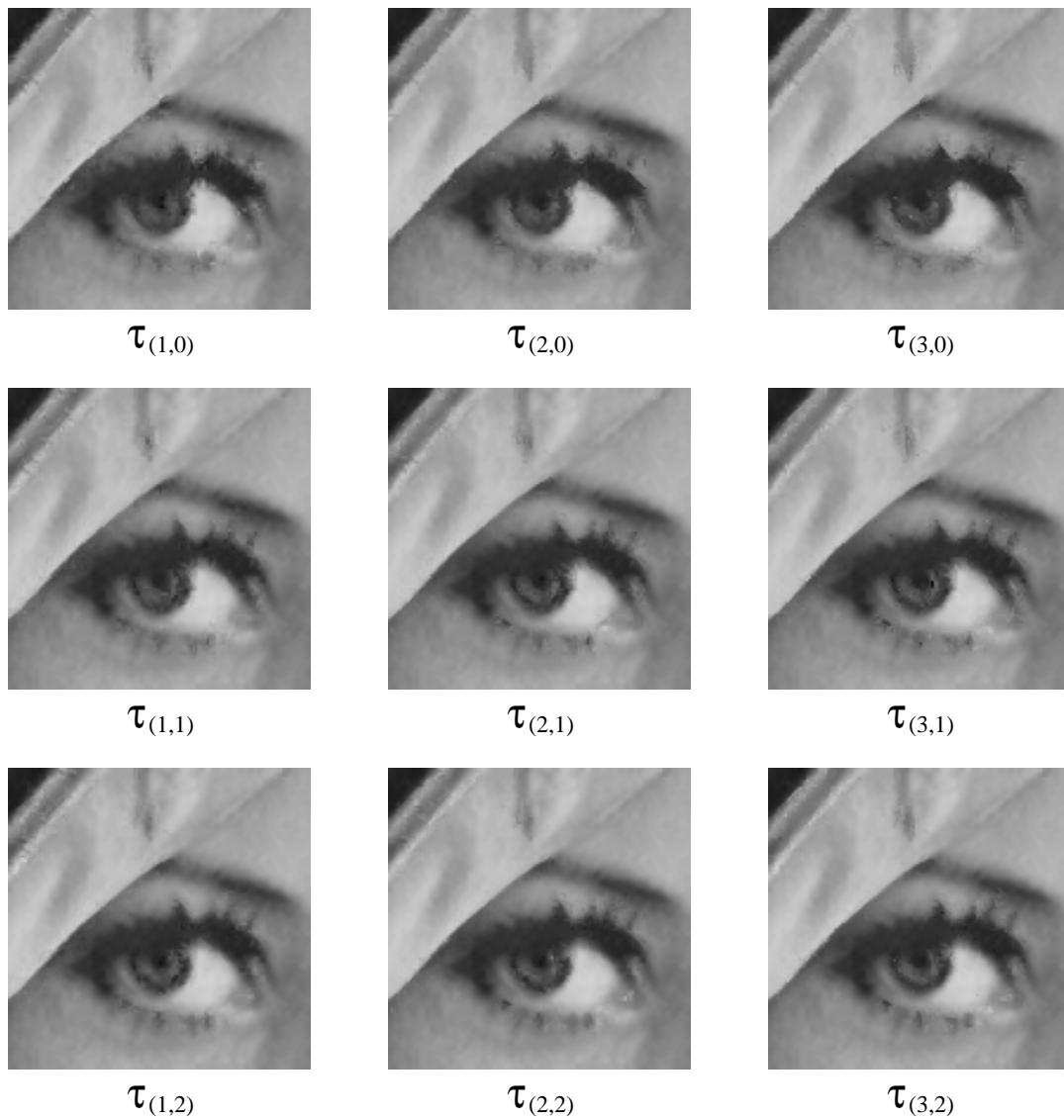


Figure 5.21 – Agrandissements de facteur 2 de l’image de Lenna obtenus en variant les paramètres p et q de la fonction de transformation massique $\tau_{(p,q)}$. Le résidu de transformation n’est pas pris en compte lors de la synthèse de similarité.

l’on peut distinguer sur les agrandissements de l’image du cameraman présentés sur la figure 5.22.

Le troisième paramètre est lié à la fonction de transformation spatiale θ_k qui agit sur le bloc traité en lui appliquant l’une des huit isométries définies par (5.4). Ce paramètre consiste à choisir les isométries qui sont permises lors de la recherche de similarités. Dans le cadre de notre application, nous avons défini les trois possibilités suivantes : identité ($k = 0$), identité et rotations ($0 \leq k \leq 3$), identité, rotations et réflexions ($0 \leq k \leq 7$). En augmentant le nombre d’isométries permises, on augmente la possibilité de trouver des similarités intéressantes, mais on augmente également de manière proportionnelle la quantité de calculs. La figure 5.23



Figure 5.22 – Agrandissements de facteur 2 des images de Lenna et du cameraman en utilisant les fonctions de transformation massique $\tau_{(1,0)}$, $\tau_{(2,1)}$ ou $\tau_{(3,2)}$. Le résidu de transformation est pris en compte lors de la synthèse de similarité.

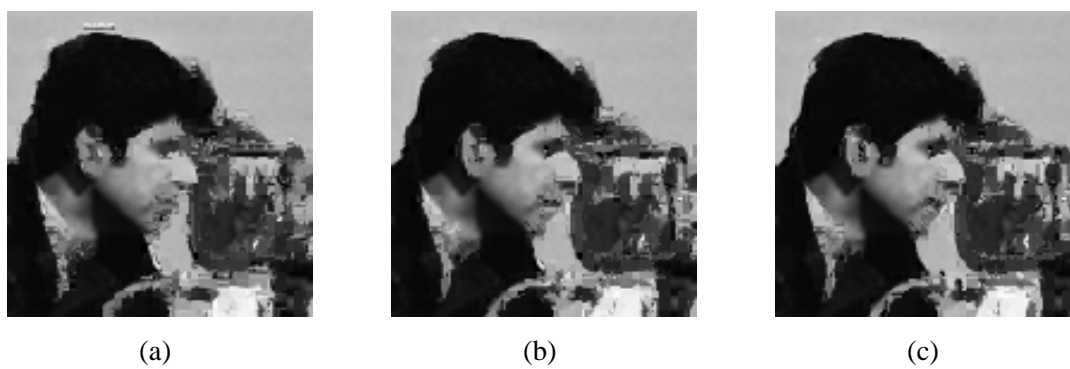


Figure 5.23 – Agrandissements de facteur 2 de l'image du cameraman obtenus avec différents groupes d'isométries: (a) identité, (b) identité et rotations, (c) identité, rotations et réflexions. Le résidu de transformation n'est pas pris en compte lors de la synthèse de similarité.

illustre les agrandissements de l'image du cameraman obtenus avec l'une des trois combinaisons d'isométries définies précédemment. Pour faciliter les observations, les résidus de transformation ne sont pas exploités lors de la synthèse des similarités. On constate que l'utilisation des rotations conjointement à l'identité permet d'améliorer significativement la qualité de l'agrandissement. L'amélioration apportée par l'utilisation supplémentaire des réflexions est beaucoup moins perceptible. En fait, les blocs générés par réflexions peuvent souvent être obtenus de manière assez équivalente par la fonction de transformation massique avec des rotations de bloc. Ceci est surtout vrai pour les blocs qui contiennent des contours. Les transformations supplémentaires apportées par les réflexions sont donc généralement redondantes avec les transformations déjà permises par les rotations. Par contre, les blocs générés par rotations ne peuvent généralement pas être obtenus par une transformation massique seule. Les isométries basées sur les rotations apportent donc un gain de qualité non négligeable. En pratique, un excellent compromis entre rapidité et qualité est réalisé en utilisant uniquement comme isométries l'identité et les rotations.

La profondeur de recherche dans le domaine \mathcal{D}

La recherche globale d'une similarité d'un bloc X dans le domaine \mathcal{D} est réalisée par une exploration autour du bloc $R_{\mathcal{D}}$ du domaine qui a la variance la plus proche de celle de X (cf. table 5.4). Nous rappelons que les blocs du domaine sont ordonnés en fonction de leur variance. La profondeur de recherche autour du bloc $R_{\mathcal{D}}$ dépend du paramètre w . Celui-ci s'exprime comme un pourcentage du nombre de blocs présents dans le domaine \mathcal{D} . Le nombre de blocs explorés de chaque côté du bloc $R_{\mathcal{D}}$ est donné par $W/2$ avec $W = (N * w)/100$, N étant le nombre de blocs présents dans le domaine \mathcal{D} .

En diminuant w , le temps nécessaire à la recherche diminue en proportion, mais la probabilité de trouver une bonne similarité devient plus faible. La limitation de la profondeur de recherche permet également d'effectuer les transformations massiques sur des blocs qui ont une variance assez proche de celle de X , et donc de limiter l'importance de ces transformations. Ceci permet de privilégier des similarités entre des blocs visuellement assez proches.

Pour nous aider à choisir plus précisément la valeur de w , on s'intéresse à l'histogramme des profondeurs des meilleures similarités du domaine \mathcal{D} trouvées pour les blocs de références. Pour chaque bloc de référence X , on mesure la profondeur entre le bloc $R_{\mathcal{D}}$, qui a la variance la plus proche de celle de X , et le bloc le plus similaire à X dans le domaine \mathcal{D} . Les histogrammes de profondeur pour les images de Lenna et du cameraman sont illustrés respectivement sur les figures 5.24(a) et 5.24(b). Dans les deux cas, on remarque clairement que la distribution n'est pas uniforme. En fait, la probabilité de trouver la meilleure similarité d'un bloc de référence diminue avec la profondeur de recherche. Ce résultat montre que l'on peut limiter la profondeur de recherche sans trop diminuer la probabilité de trouver la meilleure similarité d'un bloc dans le domaine \mathcal{D} . En prenant par exemple $w = 50\%$ pour l'image du cameraman, on trouve pour une majorité de blocs de référence leur meilleure similarité

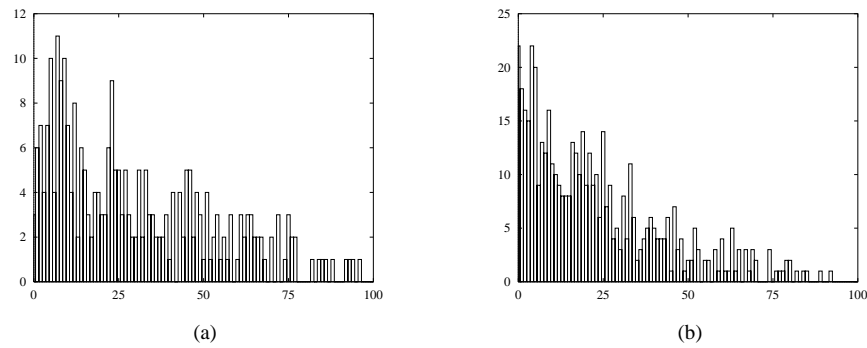


Figure 5.24 – Histogramme de profondeur des meilleures similarités pour : (a) l’image de Lenna, (b) l’image du cameraman. (voir texte pour les explications)

dans le domaine \mathcal{D} en explorant seulement la moitié de ce domaine.

La recherche locale de similarités

La recherche d’une similarité localement à un bloc X dépend du paramètre T (cf. table 5.5). Il s’agit d’un paramètre de tolérance sur le recouvrement du bloc X par le bloc similaire. Lorsque T est nul, le bloc similaire recouvre complètement le bloc X . Lorsque $0 < T \leq 5$, le recouvrement peut être partiel, et lorsque $T > 5$, il n’y a plus forcément de recouvrement.

La figure 5.25 illustre les agrandissements des images de Lenna et du cameraman obtenus en faisant varier le paramètre T . Pour les deux images, on remarque que la recherche locale permet de synthétiser des similarités de bonne qualité pour les contours simples, comme le bord du chapeau de Lenna ou le pourtour de la tête du cameraman. En fait, ce résultat est dû au caractère auto-similaire des contours à travers les résolutions (cf. pyramides de différences de la figure 5.16). Pour les blocs qui contiennent des motifs plus complexes, la recherche locale de similarité n’est pas convaincante. Pour ces blocs, il est préférable de faire une recherche globale sur toute la pyramide. On observe également sur la figure 5.25 que la recherche locale avec recouvrement complet ($T = 0$) est souvent trop restrictive. Une recherche avec un recouvrement partiel ($T = 5$) améliore beaucoup la qualité de l’agrandissement. Par contre, l’absence éventuelle de recouvrement ($T = 10$) n’améliore pas de manière significative la qualité. En fait, plus on augmente T , et plus on tend vers une recherche globale des similarités. Ceci permet d’améliorer progressivement la synthèse des blocs complexes, mais cette progression demeure assez lente.

Le partitionnement du niveau de super-résolution

Pour simplifier, nous étudions les effets du partitionnement du niveau de super-résolution sur la qualité de l’agrandissement dans le contexte d’un filtre à largeur paire. La figure 5.26 illustre les agrandissements obtenus avec les différents partitionnements représentés sur la figure 5.2. Lorsque les blocs fils ont une taille de 8×8 pixels, ils ont le même support que leurs blocs pères de taille 4×4 pixels. Dans ce



Figure 5.25 – Agrandissements de facteur 2 des images de Lenna et du cameraman obtenus en effectuant une recherche locale des similarités et en faisant varier le paramètre T . Pour chaque couple d'agrandissements, celui du haut est obtenu sans utiliser les résidus de transformation lors de la synthèse des similarités, et celui du bas est obtenu en les utilisant.

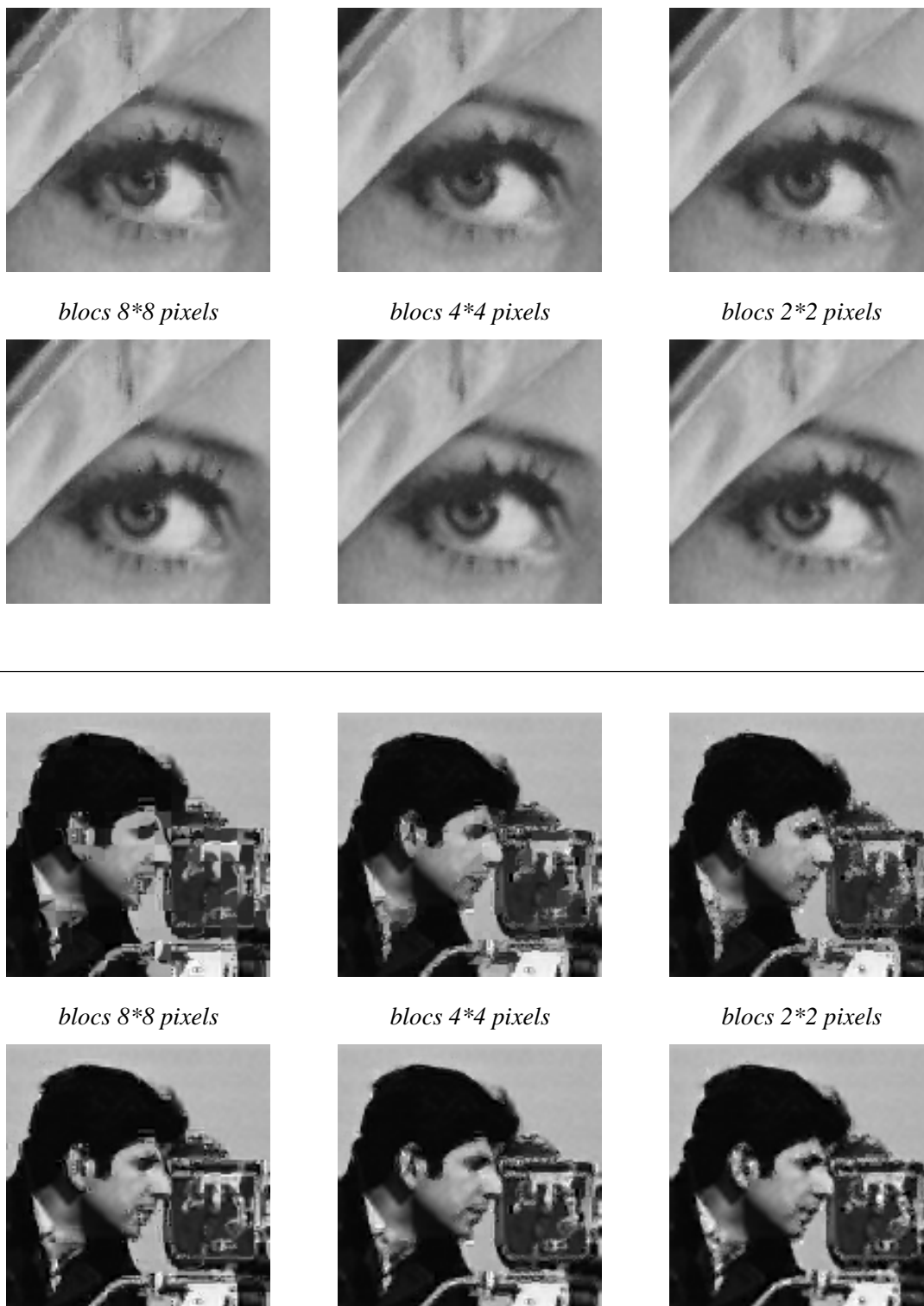


Figure 5.26 – Agrandissements de facteur 2 des images de Lenna et du cameraman obtenus avec différentes partitions du niveau de super-résolution. Pour chaque couple d'agrandissements, celui du haut est obtenu sans utiliser les résidus de transformation lors de la synthèse des similarités, et celui du bas est obtenu en les utilisant.

cas, les blocs pères forment une partition de la base de la pyramide. L'absence de recouvrement des blocs pères produit lors de la synthèse des similarités un effet de bloc qui est partiellement estompé en utilisant les résidus de transformation. Lorsque les blocs fils ont une taille de 4×4 pixels, leurs blocs pères se recouvrent et l'effet de bloc n'est plus perceptible. L'utilisation de blocs fils de taille 2×2 pixels permet d'augmenter le taux de recouvrement des blocs pères. Cependant, on remarque que les améliorations apportées sont beaucoup moins significatives. De plus, on constate sur l'agrandissement du cameraman un effet «pointilliste».

En diminuant la taille des blocs fils, on augmente proportionnellement leur nombre, ainsi que celui des similarités à rechercher dans le domaine \mathcal{D} . Pour faire un compromis intéressant entre rapidité et qualité, on voit qu'il est nettement préférable d'utiliser des blocs fils de taille 4×4 pixels. Le partitionnement équivalent dans le contexte d'un filtre à largeur impaire est illustré sur la figure 5.3.

5.8.3 Validation du modèle

Notre méthode d'agrandissement est basée sur la préservation de similarités inter-blocs à travers deux résolutions consécutives de l'image (cf. section 5.7). Dans cette section, nous nous intéressons à la validité de cette hypothèse. Pour cela, nous réalisons l'agrandissement de facteur 2 d'une image obtenue par réduction d'une image originale. Cette procédure nous permet d'effectuer des comparaisons entre l'image agrandie et l'image originale qui sert de référence.

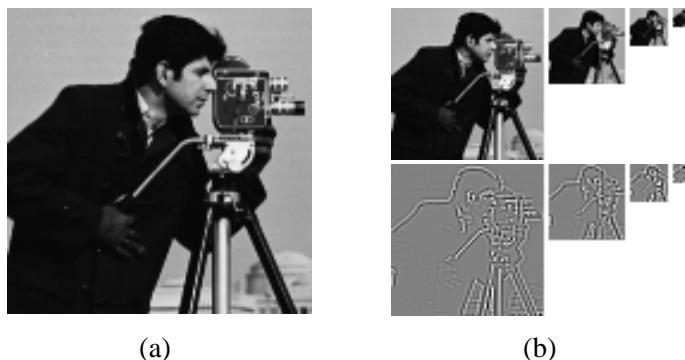


Figure 5.27 – (a) Image de référence. (b) Pyramide passe-bas obtenue à partir de la réduction de l'image de référence et pyramide de différences associée (les niveaux de gris sont amplifiés d'un facteur 2).

Nous utilisons l'image du cameraman de 128×128 pixels comme image de référence. L'image réduite est obtenue par un filtrage avec le filtre de Burt de taille 5×5 ($\beta = 0.625$) et une décimation de facteur 2. L'agrandissement est réalisé par une synthèse de similarités sur la pyramide générée à partir de l'image réduite du cameraman. Les paramètres de l'agrandissement sont ceux de la configuration standard que nous avons définie dans la section 5.8.1. La figure 5.27 montre l'image de référence, la pyramide passe-bas de 4 niveaux générée à partir de l'image réduite et la pyramide de différences associée.

Dans la suite, nous nous intéressons à l'ensemble des blocs (X'_i) du niveau de super-résolution qui sont synthétisés par similarité. Pour chacun d'entre eux, nous associons les deux blocs résiduels suivants :

$$\begin{cases} \varepsilon_i = X_i - \tau_{(p,q)} \circ \theta_k(Y_i) \\ \varepsilon'_i = R_i - X'_i = R_i - \tau_{(p,q)} \circ \theta_k(Y'_i) - I(\varepsilon_i) \end{cases} \quad (5.22)$$

où X_i est le bloc père de X'_i , Y_i est le bloc similaire à X_i par la transformation $\tau_{(p,q)} \circ \theta_k$, Y'_i est le bloc fils de Y_i et R_i est le bloc situé sur l'image de référence au même emplacement que X'_i . En d'autres termes, ε_i représente le bloc résiduel de la similarité entre les blocs pères X_i et Y_i , et ε'_i représente le bloc résiduel de la similarité entre les blocs fils R_i et Y'_i . La valeur moyenne du résidu représenté par les blocs ε_i et ε'_i peut être mesurée par la fonction :

$$Q(x) = \frac{\sum_{m,n=0}^{N-1} |x(m, n)|}{N^2} \quad (5.23)$$

où x est un bloc de taille $N \times N$ pixels. Les mesures $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ représentent donc des mesures d'erreur moyenne de similarité entre (respectivement) les blocs pères et les blocs fils. Pour que la mesure $Q(\varepsilon'_i)$ soit significative par rapport à $Q(\varepsilon_i)$, il faut que les supports des blocs fils soient identiques à ceux des blocs pères. En d'autres termes, pour calculer $Q(\varepsilon'_i)$ on prend pour le support de X'_i celui de son bloc père X_i et non pas celui qui est imposé par le partitionnement du niveau de super-résolution. Ainsi, dans le contexte d'un filtre à largeur impaire, le bloc ε_i a une taille de 5×5 pixels et le bloc ε'_i a une taille de 9×9 pixels. De même, dans le contexte d'un filtre à largeur paire, le bloc ε_i a une taille de 4×4 pixels et le bloc ε'_i a une taille de 8×8 pixels.

D'après l'hypothèse de préservation des similarités à travers les résolutions, $Q'(\varepsilon_i)$ devrait tendre vers zéro lorsque $Q(\varepsilon_i)$ tend vers zéro. Dans la suite, nous nous intéressons tout d'abord à la relation entre $Q'(\varepsilon_i)$ et $Q(\varepsilon_i)$ dans le contexte où le filtre est de largeur impaire et les paramètres sont ceux de la configuration standard. Nous étudions également l'évolution de $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ en fonction de la complexité (écart-type) des blocs pères (X_i). Ensuite, nous nous intéressons à l'influence de certains paramètres de la configuration standard sur la relation entre $Q'(\varepsilon_i)$ et $Q(\varepsilon_i)$. Nous étudions en particulier les influences de la transformation massique du modèle de similarité, du filtre à largeur paire, et de la recherche locale des similarités.

La préservation des similarités

La figure 5.28(a) illustre le résidu de similarité entre les blocs fils en fonction du résidu de similarité entre leurs blocs pères, sans que $I(\varepsilon_i)$ soit pris en compte pour synthétiser X'_i dans (5.22). La barre verticale en pointillé indique la valeur moyenne de $Q(\varepsilon_i)$ et la barre horizontale celle de $Q(\varepsilon'_i)$. Tout d'abord, on remarque que plus $Q(\varepsilon_i)$ diminue et plus $Q(\varepsilon'_i)$ diminue. Ceci conforte notre hypothèse de préservation des similarités. On remarque également que $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ ont une valeur souvent

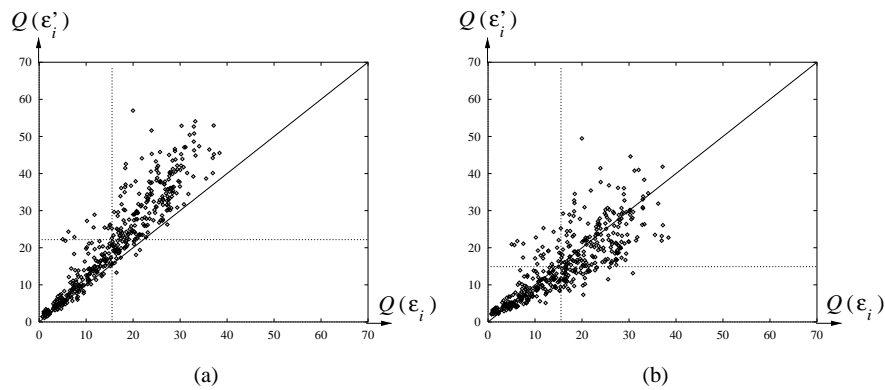


Figure 5.28 – Résidu de similarité entre les bloc fils en fonction du résidu de similarité entre les blocs pères. Le résidu de similarité entre les blocs fils est calculé : (a) sans prendre en compte $I(\varepsilon_i)$, (b) en prenant en compte $I(\varepsilon_i)$.

très proche, de sorte que globalement, le nuage de points est assez proche de la fonction $f(x) = x$.

Comme le montre la figure 5.28(b), lorsque les blocs (X'_i) sont synthétisés en corrigeant la transformation avec $I(\varepsilon_i)$, le nuage de points se rapproche nettement de l'axe des abscisses. En fait, le résidu de similarité entre les blocs fils devient globalement plus faible. Celui-ci devient même majoritairement plus petit que le résidu de similarité entre les blocs pères. Cette observation montre clairement l'importance de rajouter $I(\varepsilon_i)$ lors de la synthèse des similarités.

En résumé, la figure 5.28 montre clairement que l'hypothèse de préservation des similarités se vérifie pour la majorité des blocs. Il existe toujours des blocs pour lesquels cette hypothèse ne s'applique pas, mais ceux-ci sont très minoritaires. En d'autres termes, la probabilité pour qu'une similarité inter-blocs soit préservée à une résolution supérieure est en pratique relativement élevée.

Relations entre le résidu de similarité et l'écart-type des blocs

On peut se demander si la validité de la préservation des similarités dépend de la complexité des blocs pères (X_i). On peut également se demander si $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ n'augmentent pas proportionnellement à cette complexité. C'est ce que nous vérifions maintenant en évaluant la complexité d'un bloc par la mesure de l'écart-type de ses niveaux de gris.

La figure 5.29 illustre le rapport entre $Q(\varepsilon'_i)$ et $Q(\varepsilon_i)$ en fonction de $\sigma(X_i)$. On remarque que ce rapport n'augmente pas lorsque $\sigma(X_i)$ augmente et qu'il demeure plutôt constant avec une valeur proche de 1. Il a même tendance à être plus grand lorsque $\sigma(X_i)$ est plus faible. En fait, ce phénomène s'explique par le fait qu'il y a plus de probabilités pour que des blocs qui ont un écart-type faible aient un résidu de similarité important par rapport à la faible quantité d'informations qu'ils contiennent. On remarque que l'utilisation de $I(\varepsilon_i)$ lors de la synthèse permet de diminuer globalement le rapport $Q(\varepsilon'_i)/Q(\varepsilon_i)$. En fait, ceci est surtout vrai lorsque

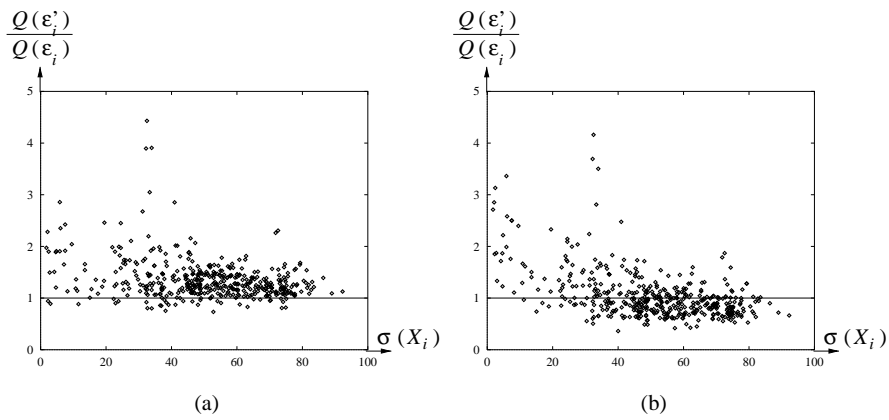


Figure 5.29 – Rapport entre $Q(\varepsilon'_i)$ et $Q(\varepsilon_i)$ en fonction de l'écart-type du bloc père X_i . Le résidu de similarité des bloc fils est calculé : (a) sans prendre en compte $I(\varepsilon_i)$, (b) en prenant en compte $I(\varepsilon_i)$.

l'écart-type des blocs pères est important ($\sigma(X_i) > 40$). Lorsque l'écart-type est faible ($\sigma(X_i) < 40$), le rapport reste inchangé, voire même moins bon lorsque $\sigma(X_i)$ est très faible.

La figure 5.30 montre l'influence de l'écart-type des blocs pères (X_i) sur la mesure $Q(\varepsilon_i)$ et sur la mesure $Q(\varepsilon'_i)$. On voit clairement que l'amplitude de ces deux mesures augmente avec la valeur de $\sigma(X_i)$. On pourrait supposer *a priori* que ces deux mesures sont proportionnelles à la complexité des blocs pères. Cependant, les nuages de points de la figure 5.30 montrent que ce n'est pas exactement le cas. Lorsque $\sigma(X_i)$ augmente, on trouve toujours des blocs pour lesquels le résidu de similarité est proche de zéro.

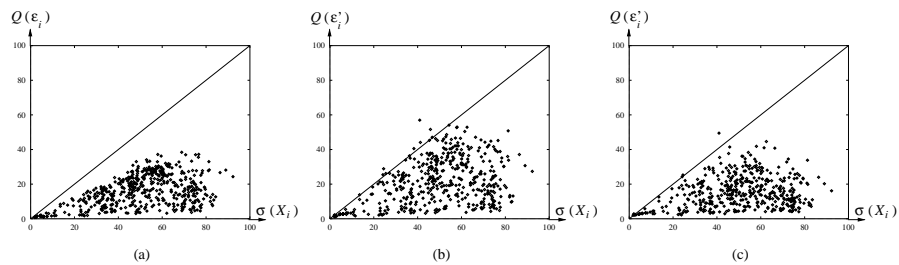


Figure 5.30 – (a) Résidu de similarité entre les blocs pères en fonction de l'écart-type du bloc père X_i . (b) Résidu de similarité entre les blocs fils en fonction de l'écart-type du bloc père X_i sans prendre en compte $I(\varepsilon_i)$. (c) Résidu de similarité entre les blocs fils en fonction de l'écart-type du bloc père X_i en prenant en compte $I(\varepsilon_i)$.

Ces différents résultats montrent que l'hypothèse de préservation idéale des similarités est validée indépendamment de la complexité (écart-type) des blocs traités. Quelle que soit cette complexité, lorsque le résidu de similarité entre les blocs pères tend vers zéro, le résidu de similarité entre les blocs fils tend également vers zéro.

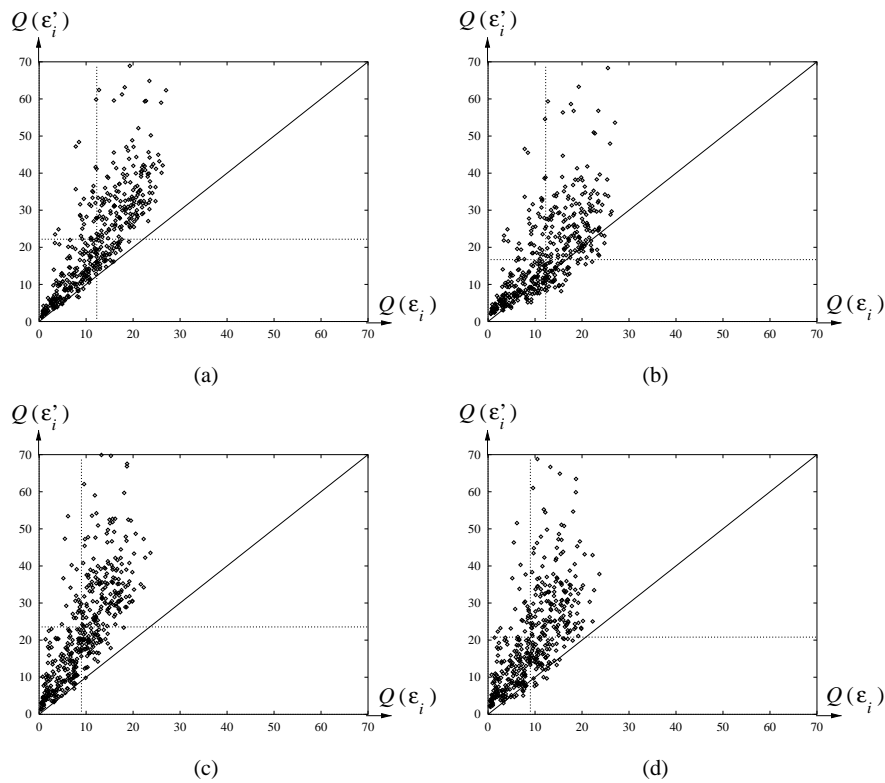


Figure 5.31 – Résidu de similarité entre les bloc fils en fonction du résidu de similarité entre les blocs pères pour les transformations massiques $\tau_{(2,1)}$ (a-b) et $\tau_{(3,2)}$ (c-d). Le résidu de similarité des bloc fils est calculé : (a-c) sans prendre en compte $I(\varepsilon_i)$, (b-d) en prenant en compte $I(\varepsilon_i)$.

Influence de la transformation massique

La figure 5.31 illustre les relations entre $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ obtenues avec les transformations massiques $\tau_{(2,1)}$ et $\tau_{(3,2)}$. On voit qu'en augmentant les degrés du polynôme d'échelle et du polynôme fixe, les résidus de similarité entre les blocs pères diminuent. Par contre, les résidus de similarité entre les blocs fils ne diminuent pas en conséquence. Au contraire, en moyenne, ils augmentent. De plus, on remarque que les améliorations apportées par $I(\varepsilon_i)$ lors de la synthèse sont de moins en moins significatives. Finalement, en comparant les nuages de points sur les figures 5.28(b), 5.31(b) et 5.31(d), il apparaît clairement que l'hypothèse de préservation des similarités est surtout vérifiée lorsque le modèle de similarité est basé sur une transformation massique linéaire $\tau_{(1,0)}$. En augmentant les paramètres p et q de $\tau_{(p,q)}$, cette hypothèse est de plus en plus fragilisée.

Filtre à largeur paire

La figure 5.32 illustre la relation entre $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ lorsque le filtre est de largeur paire. La fenêtre de similarité de largeur 4×4 pixels produit des résidus de similarité qui sont en moyenne plus faibles pour les blocs pères. En revanche, les résidus de

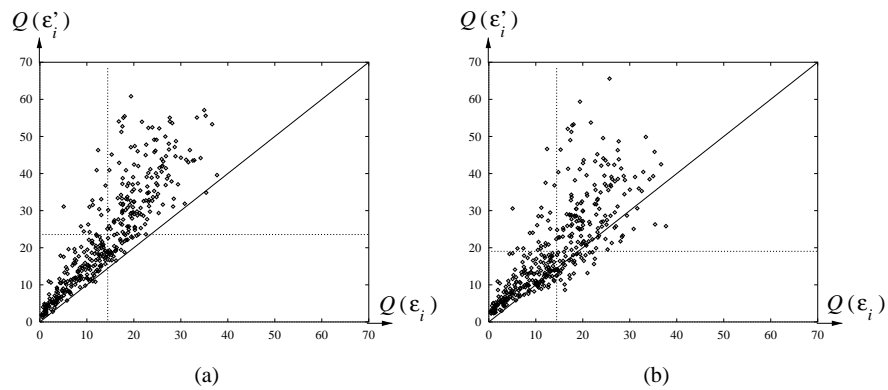


Figure 5.32 – Résidu de similarité entre les bloc fils en fonction du résidu de similarité entre les blocs pères dans le cas d'un filtre à largeur paire. Le résidu de similarité des bloc fils est calculé : (a) sans prendre en compte $I(\varepsilon_i)$, (b) en prenant en compte $I(\varepsilon_i)$.

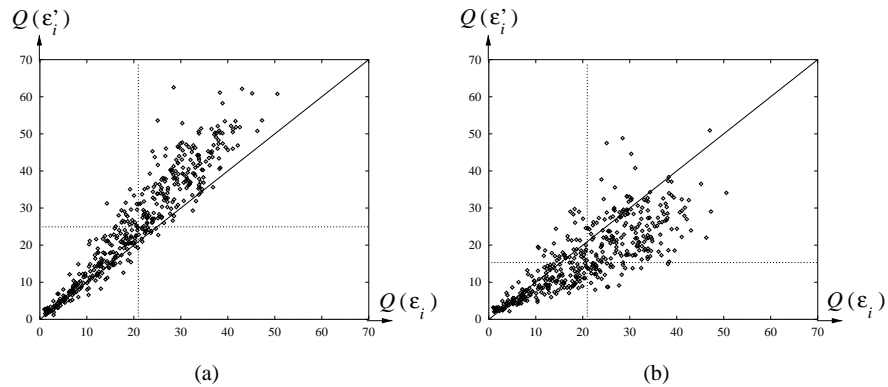


Figure 5.33 – Résidu de similarité entre les bloc fils en fonction du résidu de similarité entre les blocs pères dans le contexte d'une recherche locale des similarités ($T = 5$). Le résidu de similarité des bloc fils est calculé : (a) sans prendre en compte l'interpolation du résidu des blocs pères $I(\varepsilon_i)$, (b) en prenant en compte l'interpolation du résidu des blocs pères $I(\varepsilon_i)$.

similarité associés aux blocs fils sont en moyenne plus élevés. Ces résultats sont en accord avec la dégradation de l'agrandissement constatée visuellement sur la figure 5.17 lorsque l'on utilise un filtre à largeur paire.

La recherche locale des similarités

La figure 5.33 illustre la relation entre $Q(\varepsilon_i)$ et $Q(\varepsilon'_i)$ dans le contexte d'une recherche locale des similarités ($T = 5$). Les nuages de points ont à peu près les mêmes caractéristiques que ceux montrés sur la figure 5.28 dans le contexte d'une recherche globale des similarités. Les résidus moyens de similarité sont cependant plus élevés, surtout pour les blocs pères.

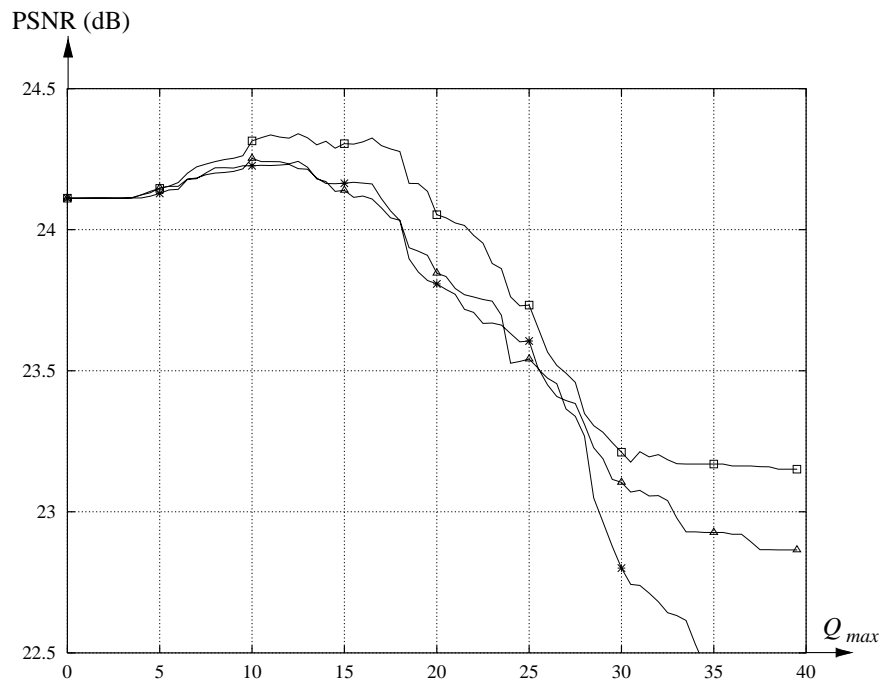


Figure 5.34 – Rapport signal à bruit en fonction de la valeur limite Q_{max} du résidu de similarité entre les blocs pères dans le contexte d'une : (*) recherche locale des similarités ($T = 5$), (Δ) recherche globale des similarités avec isométries rotations, (\square) recherche globale des similarités avec isométries rotations+réflexions.

Choix d'une limite sur le résidu de similarité

Lorsque le résidu d'une similarité $Q(\varepsilon_i)$ est important, cela signifie que le bloc Y_i est peu similaire à X_i . Dans ce cas, il est préférable de ne pas réaliser la synthèse de la similarité à une résolution supérieure afin de ne pas dégrader la qualité de l'agrandissement. Il faut donc fixer une valeur limite Q_{max} sur le résidu de similarité. Dans le contexte de la configuration standard des paramètres, la figure 5.28 montre que le résidu moyen des similarités entre les blocs pères se situe autour d'une valeur de 16. Le choix de $Q_{max} = 16$ peut donc constituer un premier choix intéressant. Toutefois, pour mieux cerner l'influence de Q_{max} sur la qualité de l'agrandissement, on peut analyser la mesure du $PSNR$ entre l'image de référence et l'image agrandie en fonction de la valeur de Q_{max} . Le résultat de cette mesure est présenté sur le graphique de la figure 5.34. Lorsque $Q_{max} = 0$, aucun bloc fils n'est synthétisé par similarité et le $PSNR$ est de 24.11 dB. Dans ce cas, l'image agrandie est en fait identique au résultat d'une interpolation par transformée B-spline cubique. Ensuite, le $PSNR$ augmente pour atteindre la valeur maximale 24.25 dB pour $Q_{max} = 10$, puis diminue pour repasser en dessous de 24.11 dB aux alentours de $Q_{max} = 16$. Au-delà de $Q_{max} = 16$, le $PSNR$ se dégrade très rapidement.

En résumé, le choix $Q_{max} = 16$ semble être relativement intéressant. Avec une valeur plus faible, le $PSNR$ est meilleur, mais le nombre de blocs synthétisés par similarité diminue, ce qui limite l'intérêt de l'approche. En revanche, avec une valeur

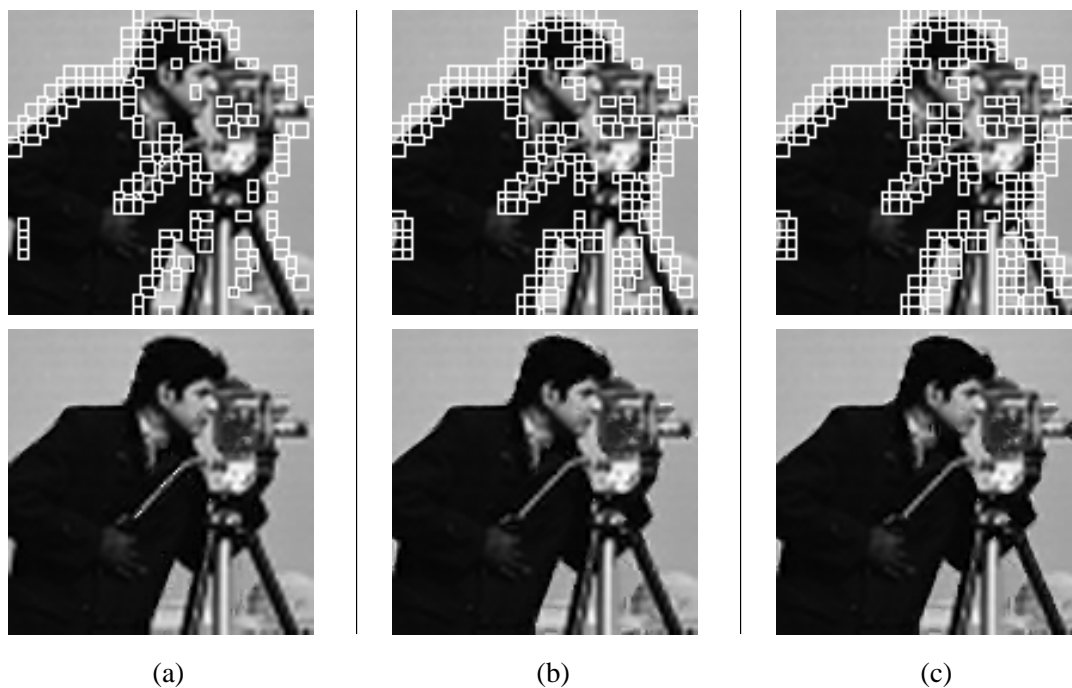


Figure 5.35 – Localisation des blocs fils synthétisés et agrandissements obtenus avec $Q_{max} = 16$ dans le contexte d'une : (a) recherche locale des similarités ($T = 5$), (b) recherche globale des similarités avec isométries *rotations*, (c) recherche globale des similarités avec isométries *rotations+réflexions*.

plus forte, le nombre de blocs synthétisés augmente, mais le *PSNR* se dégrade très rapidement. La figure 5.35(b) illustre la localisation des blocs synthétisés, ainsi que l'agrandissement obtenu en choisissant $Q_{max} = 16$. Les blocs synthétisés sont situés principalement sur le pourtour du cameraman. Il y a très peu de blocs positionnés sur la caméra : les similarités sont difficiles à trouver dans les niveaux supérieurs de la pyramide.

La figure 5.34 illustre également les mesures du *PSNR* dans les cas d'une recherche locale et d'une recherche globale avec des isométries supplémentaires de type *réflexion*. La courbe obtenue dans le premier cas est relativement proche de la courbe associée à la recherche globale avec isométries de type *rotation*. Cependant, pour une valeur de Q_{max} donnée, le nombre de blocs synthétisés est beaucoup plus faible lorsque la recherche est locale. Ceci est illustré par la figure 5.35(a) avec $Q_{max} = 16$.

La courbe obtenue dans le deuxième cas présente une valeur du *PSNR* toujours plus élevée que celle de la recherche globale effectuée uniquement avec des isométries de type *rotation*. La valeur maximale du *PSNR* est 24.34 dB pour $Q_{max} = 12.5$. Non seulement la valeur du *PSNR* est plus élevée, mais le nombre de blocs synthétisés est également plus important, comme le montre la figure 5.35(c). Les améliorations sont surtout situées au niveau du pied droit de la caméra.

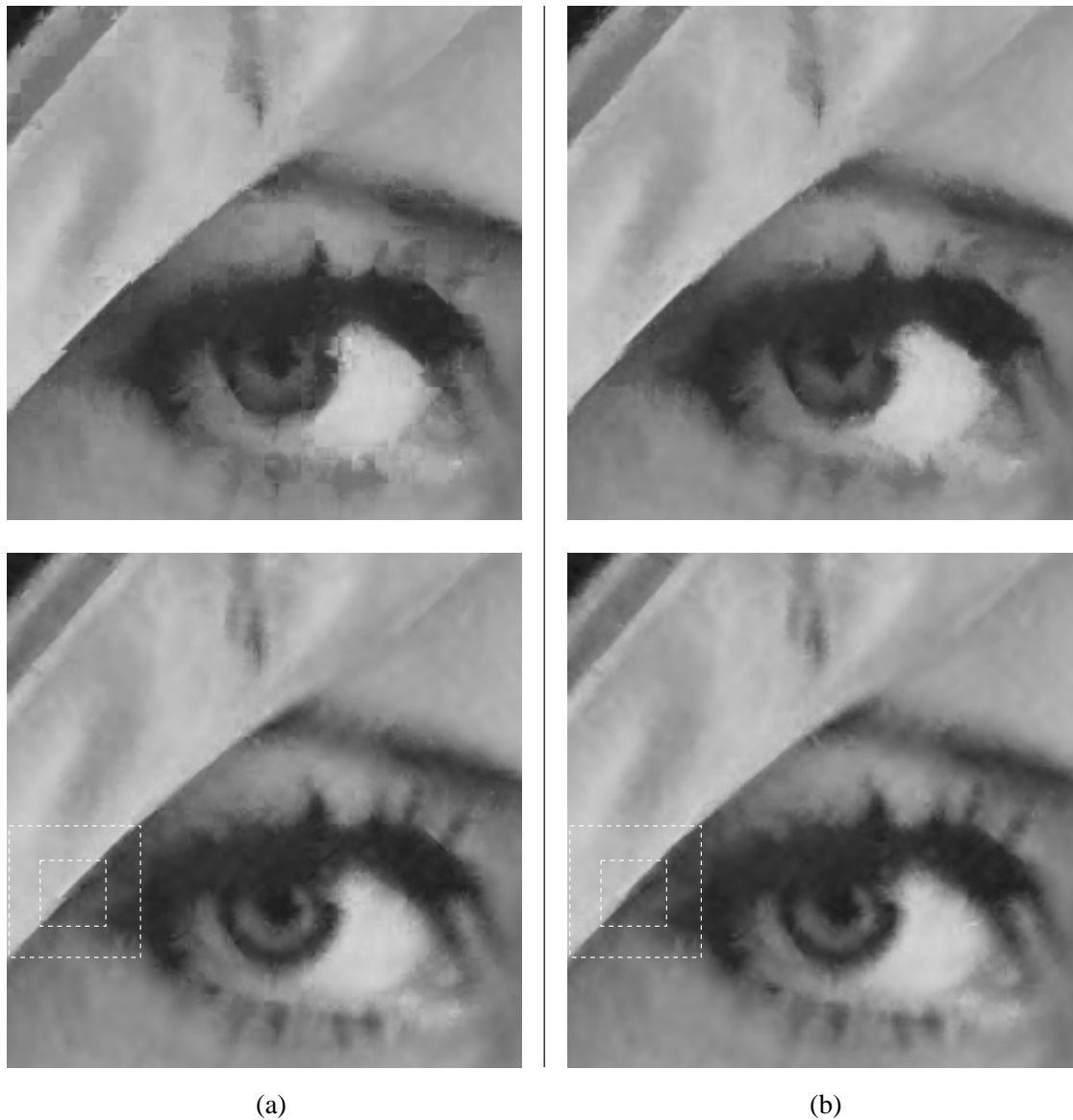


Figure 5.36 – Agrandissements de facteur 4 de l’image de Lenna : (a) approche par synthèse unique, (b) approche par analyse-synthèse. L’agrandissement du haut est obtenu sans utiliser les résidus de transformation lors de la synthèse des similarités, et celui du bas est obtenu en les utilisant.

5.8.4 Agrandissement de facteur supérieur à deux

Pour agrandir une image avec un facteur plus grand que deux, nous avons proposé deux approches dans la section 5.7.2. La première réalise la synthèse des similarités à toutes les résolutions en exploitant uniquement la phase d’analyse réalisée pour produire l’agrandissement de facteur deux. La seconde approche effectue quant à elle une synthèse à partir d’une analyse renouvelée à chaque résolution intermédiaire.

Le résultat de ces deux approches est présenté sur la figure 5.36 dans le cadre d’un agrandissement de facteur 4 de l’image de Lenna. Les différences sont surtout

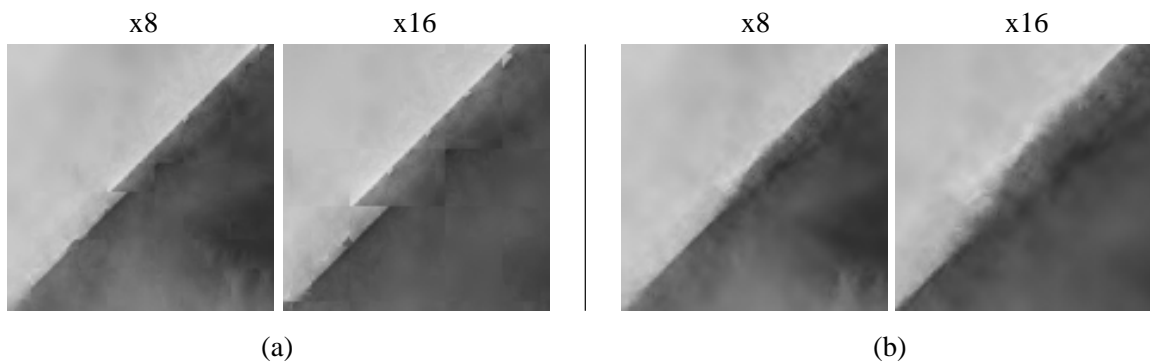


Figure 5.37 – Détails des agrandissements de facteur 8 et 16 de l'image de Lenna : (a) approche par synthèse unique, (b) approche par analyse-synthèse.

perceptibles au niveau des discontinuités comme le bord du chapeau dans la zone encadrée. La figure 5.37 illustre cette zone particulière dans le cadre d'agrandissements de facteur 8 et 16 de l'image de Lenna. Les différences entre les deux approches sont ici flagrantes. Dans le cas de l'approche par synthèse unique, on observe l'apparition d'un léger effet de bloc avec des décrochements. Cet effet n'est pas présent dans le cas de l'approche par analyse-synthèse. Toutefois, on peut observer un effet de gommage.

En fait, la première approche réalise un zoom auto-similaire sur l'agrandissement de facteur 2. Elle révèle donc des différences entre les blocs adjacents qui n'étaient pratiquement pas perceptibles à l'échelle de la première synthèse, d'où l'effet de bloc. Avec la deuxième approche, ces différences sont lissées à chaque nouvelle résolution en recommençant l'analyse sur une partition plus fine. Le recouvrement entre les blocs pères permet de prévenir l'apparition d'un effet de bloc.

5.8.5 Comparaisons avec d'autres méthodes d'agrandissement

Dans cette section, nous comparons les agrandissements produits par notre méthode à ceux produits par l'interpolation par transformée B-spline et l'interpolation par filtrage médian.

La figure 5.38 présente les résultats obtenus par agrandissement de facteur 2 de l'image du cameraman réduite qui apparaît sur la figure 5.27(b). L'image de référence et son spectre d'amplitude sont montrés sur la figure 2.1. L'interpolation par transformée B-spline se contente de préserver les fréquences de l'image réduite, d'où l'apparition d'un effet de flou assez prononcé. Malgré l'utilisation d'une B-spline cubique, on observe l'apparition d'un effet d'ondulation localisé sur le pourtour du cameraman. L'interpolation par filtrage médian améliore un peu la qualité de l'image agrandie mais l'effet de flou persiste sur le pourtour du cameraman. On remarque que le filtre non-linéaire est particulièrement efficace au niveau du bras de direction de la caméra. On observe d'ailleurs sur le spectre que la trace diagonale associée au bras de direction se prolonge dans les hautes fréquences comme sur le spectre de référence sur la figure 2.1(b). L'image agrandie par synthèse de similarités est

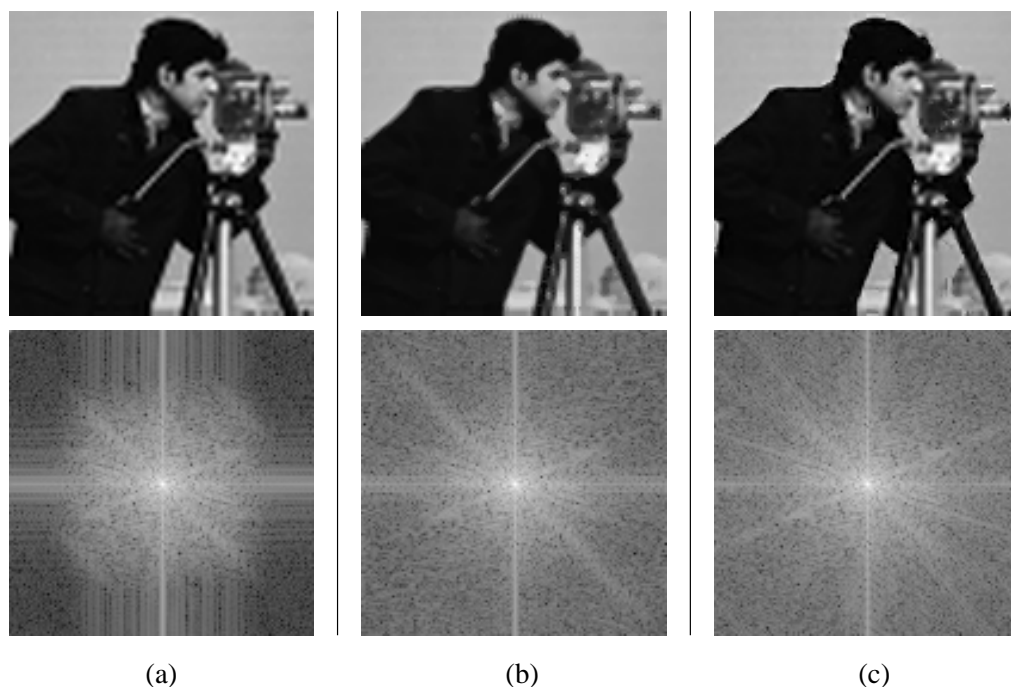


Figure 5.38 – Agrandissement de facteur 2 de l'image obtenue par réduction de l'image du cameraman et spectre associé : (a) interpolation par transformée B-spline, (b) interpolation par filtrage médian, (c) synthèse de similarités.

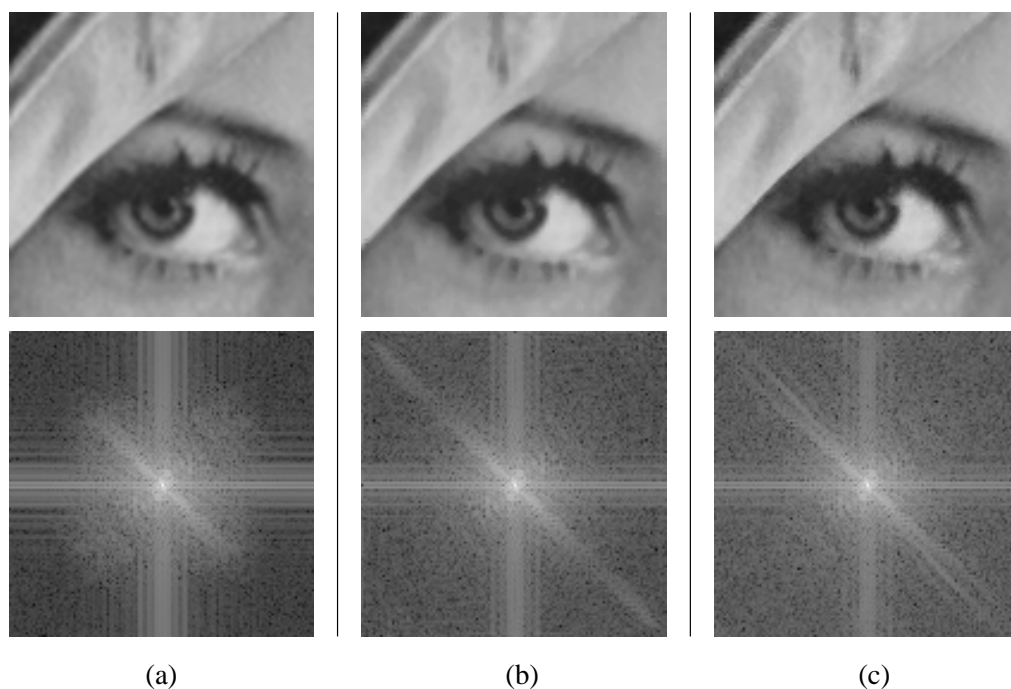


Figure 5.39 – Agrandissement de facteur 2 de l'image de Lenna et spectre associé : (a) interpolation par transformée B-spline, (b) interpolation par filtrage médian, (c) synthèse de similarités.

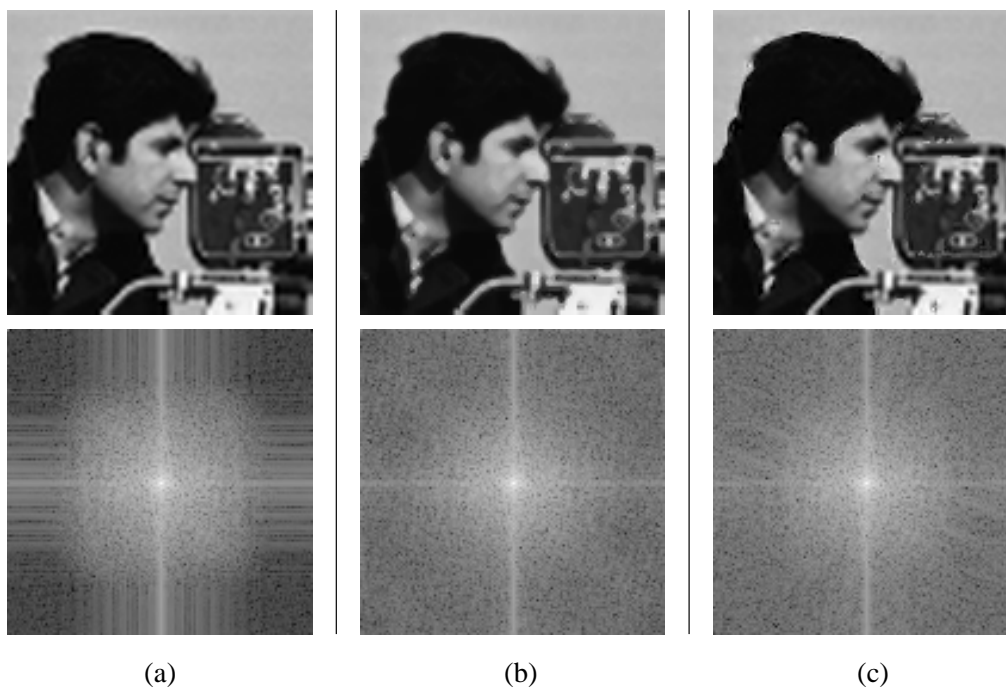


Figure 5.40 – Agrandissement de facteur 2 de l’image du cameraman et spectre associé : (a) interpolation par transformée B-spline, (b) interpolation par filtrage médian, (c) synthèse de similarités.

la même que celle qui est montrée sur la figure 5.35. L’effet de flou est absent du pourtour du cameraman et du pied droit de la caméra : ce sont essentiellement ces zones qui contiennent des blocs synthétisés (cf. figure 5.35(c)). Le reste de l’image est identique à l’image agrandie par transformée B-spline et présente donc un effet de flou important notamment au niveau de la caméra. Sur le spectre, on remarque que plusieurs traces se prolongent dans les hautes fréquences. Elles correspondent à l’orientation globale du cameraman, ainsi qu’au pied droit de la caméra. Toutefois, contrairement au résultat observé avec le filtrage médian, la trace diagonale associée au bras de direction n’est pas prolongée. En fait, la figure 5.35(c) montre que les blocs synthétisés au niveau du bras de direction entourent celui-ci mais ne le recouvrent presque pas.

La figure 5.39 illustre les résultats obtenus en agrandissant l’image de Lenna. L’interpolation par transformée B-spline produit toujours du flou, surtout sur le bord inférieur du chapeau. Le filtre médian améliore très peu la qualité de l’image agrandie. On peut observer sur le spectre un prolongement de la trace diagonale. Cependant, celui-ci est artificiel : il est comparable à ce que l’on obtient avec une interpolation au plus proche voisin. Il est dû à une duplication du spectre dans la direction diagonale. L’image agrandie par synthèse de similarités possède plus de netteté, surtout au niveau des contours du chapeau. On observe d’ailleurs sur le spectre un prolongement de la trace diagonale qui se scinde en deux traces distinctes. En fait cette scission, dont on perçoit l’amorce sur le spectre de l’image agrandie

par transformée B-spline, est due aux deux contours du chapeau qui prennent des directions légèrement différentes.

La figure 5.40 montre les résultats obtenus en agrandissant l'image du cameraman. Les résultats sont assez similaires à ceux de la figure 5.38. Toutefois, en raison de l'absence de directions privilégiées, on ne perçoit pas de traces particulières sur le spectre. Remarquons que l'agrandissement par synthèse de similarités est obtenu en prenant $Q_{max} = 16$, comme dans l'exemple illustré sur la figure 5.38.

5.9 Conclusion

Nous avons décrit dans ce chapitre notre première approche de l'agrandissement d'images basée sur la synthèse de similarités à partir d'une pyramide à facteur de réduction 2. Le principe global est de rechercher des similarités entre des blocs situés sur la base de la pyramide et des blocs situés sur les niveaux supérieurs. En supposant que ces liens de similarité se préservent à une résolution supérieure, les blocs fils des blocs situés sur la base de la pyramide sont synthétisés à partir de ces liens de similarité et des blocs fils des blocs similaires situés sur les niveaux supérieurs de la pyramide. Cette synthèse permet finalement d'extrapoler un niveau supplémentaire de la pyramide, et donc d'obtenir l'agrandissement de facteur 2 de l'image originale. Nous avons développé et étudié différents aspects de ce modèle, que ce soit dans la phase d'analyse (recherche des similarités) ou dans la phase de synthèse.

Phase d'analyse

La notion de similarité entre deux blocs repose sur le choix d'un modèle de transformation. Pour cela, nous avons proposé d'utiliser des fonctions polynomiales. D'après nos expérimentations, la transformation linéaire est celle qui génère globalement les meilleurs résultats lors d'une synthèse à une résolution supérieure. En fait, lorsque le degré du polynôme augmente, l'espace de transformation devient plus important et des blocs visuellement très différents peuvent devenir similaires. Dans ce cas, l'hypothèse de préservation d'une similarité à une résolution supérieure se fragilise et le risque d'erreur est plus important.

Nous avons également étudié l'utilisation d'isométries. Les rotations de blocs améliorent nettement la qualité de l'image agrandie. En revanche, les réflexions de blocs sont moins convaincantes : les améliorations apportées sont insuffisantes par rapport à la quantité de calculs nécessaire. En fait, la fonction de transformation massique permet dans la plupart des cas et notamment sur les contours simples, d'obtenir le même résultat qu'une réflexion.

En ce qui concerne la recherche des similarités, nous avons étudié deux stratégies : locale ou globale. La recherche locale est relativement adaptée pour des blocs positionnés sur des contours simples qui se conservent sur les niveaux supérieurs de la pyramide. Quel que soit le type de bloc, la recherche globale augmente la probabilité de trouver une similarité significative. Cependant, elle nécessite une quantité

de calculs nettement supérieure. Pour accélérer la recherche globale, nous avons mis en place une classification des blocs de la pyramide. Pour cela, nous avons développé un modèle original de classification basé sur la mesure d'un angle entre deux espaces de transformation. Les blocs qui partagent plus ou moins le même espace de transformation sont regroupés au sein d'une même classe. De cette manière, quel que soit le bloc de référence, la mesure de similarité est à peu près identique pour tous les blocs d'une même classe. Ainsi, lors de la recherche d'une similarité, il est suffisant de considérer un représentant par classe. Notre modèle de classification se démarque de ceux que l'on trouve en compression fractale par sa prise en compte de la transformation massique. La classification est surtout adaptée lorsque beaucoup de blocs dans l'image partagent le même espace de transformation. Dans ce cas, elle permet de diminuer considérablement la quantité de calculs sans dégrader la qualité de l'image agrandie.

Pour sélectionner les blocs de la pyramide, nous avons proposé d'utiliser une pyramide de différences. Ceci permet de prendre en compte le contenu fréquentiel des blocs et, en particulier, de réaliser la synthèse des similarités à partir de blocs qui contiennent des hautes fréquences. De plus, ceci permet de mettre en œuvre une recherche de similarité uniquement pour des blocs qui contiennent des hautes fréquences. Pour les autres blocs, l'absence de hautes fréquences autorise l'utilisation d'une méthode d'agrandissement qui préserve les fréquences initiales. Nous avons choisi d'utiliser pour cela une interpolation par transformée B-spline cubique.

Phase de synthèse

Les similarités trouvées sont très rarement parfaites. Pour compenser le résidu de similarité lors de la synthèse, celui-ci est interpolé par une transformée B-spline cubique puis ajouté au résultat. Nous avons montré que la prise en compte du résidu de similarité lors de la synthèse permettait d'obtenir des agrandissements de meilleure qualité en corrigeant en partie les défauts de similarités.

Lorsque le résidu de similarité entre deux blocs pères est trop important, il est préférable de ne pas réaliser la synthèse. Ainsi, en limitant le résidu de similarité admissible, la qualité de l'image agrandie s'améliore. La valeur-limite est issue d'un compromis entre le nombre de blocs synthétisés par similarité et la qualité de l'image agrandie.

Nous avons vérifié l'hypothèse de préservation des similarités à une résolution supérieure. Les résultats ont montré que l'hypothèse était validée pour une majorité de blocs synthétisés. Il semble que les similarités qui contredisent l'hypothèse de préservation aient un résidu non négligeable. Une solution pour éviter la synthèse de ces similarités est donc de diminuer la valeur-limite du résidu de similarité admissible pour une synthèse. Malheureusement, cette solution écarte également la synthèse de similarités vérifiant l'hypothèse de préservation.

En diminuant les supports des blocs fils par rapport à ceux des blocs pères, on génère des recouvrements entre ces derniers. Ceci permet de supprimer les fausses discontinuités entre les blocs synthétisés. La qualité de l'image agrandie est donc

meilleure, mais cette amélioration se fait au détriment du temps de calcul puisque le nombre de blocs fils augmente.

Pour réaliser un agrandissement de facteur plus grand que deux, nous avons proposé deux approches : synthèse unique ou analyse-synthèse. La seconde approche donne un agrandissement de meilleure qualité puisque la phase d'analyse est recommencée à chaque résolution intermédiaire. Par contre, la première approche est beaucoup plus rapide puisque c'est la phase d'analyse qui est coûteuse en temps de calcul.

Nous avons comparé les images agrandies par notre méthode à celles obtenues par des méthodes classiques : interpolation par transformée B-spline et filtrage médian. Les résultats montrent que notre méthode permet d'obtenir des images agrandies plus nettes. De plus, en effectuant des mesures à partir d'une image de référence, nous avons montré que notre approche donnait une valeur du *PSNR* plus élevée que celle issue de la méthode d'interpolation par B-spline cubique.

Liens et différences avec le zoom fractal

Notre approche est basée sur la notion de similarité, comme la méthode du zoom fractal (cf. section 3.4). Toutefois, le zoom fractal classique exploite directement le code de similarités produit par les méthodes de compression fractale. Notre approche prend davantage en compte les exigences de l'agrandissement. A ce titre, elle peut être considérée comme une amélioration du zoom fractal.

Les principales différences entre notre approche et le zoom fractal sont les suivantes :

- Dans notre approche, les similarités sont recherchées sur une pyramide construite selon la théorie de l'échantillonnage. Dans le zoom fractal, les similarités sont recherchés sur l'image originale en augmentant la taille des blocs. De plus, pour que les blocs sources et destinations (cf. section 3.4 pour la terminologie) aient une taille identique, les blocs sources sont soit décimés directement sans filtrage, soit filtrés avec un opérateur simple réalisant une moyenne puis décimés. En d'autres termes, le zoom fractal recherche des similarités à des résolutions inférieures de l'image, sans prendre en compte la théorie de l'échantillonnage.
- Le code de similarités produit par une méthode de compression fractale est basé sur la notion d'opérateur contractant. Ceci impose des conditions spécifiques sur les transformations élémentaires entre les blocs, ce qui limite la qualité des similarités. Dans notre approche, cette notion est complètement absente puisque l'image originale est connue et que, par conséquent, il n'est pas nécessaire d'itérer le code de similarités. Remarquons que nous avons montré dans la section 3.4, grâce au théorème 3.2 dû à Baharav *et al.*, que le zoom fractal pouvait être obtenu directement à partir de l'image originale sans itérer le code de similarités.
- Dans notre approche, nous prenons en compte les résidus de similarités pour effectuer une correction des blocs synthétisés. Nous prenons également en compte

le contenu fréquentiel des blocs en exploitant une pyramide de différences.

Limite de l'approche

La méthode d'agrandissement par synthèse de similarités a une limite bien définie. Pour fonctionner correctement, il est nécessaire de trouver des similarités sur la pyramide, or il n'est pas possible de garantir leur existence avec une image quelconque. En d'autres termes, notre méthode n'est pas adaptée lorsque les structures dans l'image sont trop complexes et multiples. En revanche, elle convient parfaitement lorsque des structures claires se dégagent de l'image. Pour améliorer la méthode, il faut donc augmenter la probabilité de trouver des similarités sur la pyramide. Pour cela, nous étudierons dans le chapitre 7 l'utilisation d'une pyramide à facteur de réduction $\sqrt{2}$ où les transitions entre niveaux sont moins brutales.

Chapitre 6

Agrandissement par induction sur un ensemble

6.1 Introduction

Nous proposons dans ce chapitre de considérer l'agrandissement d'une image I d'un point de vue ensembliste [27], [28]. Une telle approche de l'agrandissement se justifie parce que la solution idéale d'un agrandissement à une résolution¹ particulière est rarement unique. De manière générale, on peut même supposer que l'ensemble \mathcal{K} des solutions visuellement admissibles est très grand quoique non infini.

Dans ce travail, nous nous intéressons à la propriété la plus évidente de cet ensemble : la propriété de réduction qui établit que toutes les images de \mathcal{K} doivent générer par réduction une image unique et que celle-ci doit être identique à l'image I . La propriété de réduction peut être vue comme une contrainte de réversibilité de l'agrandissement. On note Δ l'opérateur de réduction qui associe une image de résolution² $k - 1$ à une image de résolution k et I_{k-1} l'image originale que l'on souhaite agrandir à la résolution k . La propriété de réduction de \mathcal{K} s'écrit alors :

$$(\forall X_k \in \mathcal{K}) \quad \Delta(X_k) = I_{k-1} \quad (6.1)$$

et l'ensemble des images de résolution k qui ont la même propriété est donné par :

$$\Delta^{-1}(I_{k-1}) = \{X_k \mid \Delta(X_k) = I_{k-1}\} \quad (6.2)$$

En pratique, l'opérateur de réduction utilisé réalise un filtrage passe-bas qui supprime les hautes fréquences de X_k , puis un sous-échantillonnage qui ramène la résolution à celle de l'image originale I_{k-1} et enfin une quantification des valeurs réelles

1. Afin de simplifier les développements de ce chapitre, nous nous intéressons essentiellement à un agrandissement de facteur 2. Nous verrons dans la section 6.5 comment adapter ces développements pour réaliser un agrandissement de facteur supérieur à 2.

2. Une image est considérée comme une matrice de taille $2^k \times 2^k$. On dit alors qu'il s'agit d'une image de *résolution* k et cette information est placée en indice (ex: I_k).

sur l'ensemble des niveaux de gris à valeurs entières. Pour que la théorie demeure aussi générale que possible, le filtre passe-bas est une matrice carrée de taille et de poids quelconques.

Avec l'opérateur de réduction adopté, l'égalité stricte dans (6.2) est souvent trop contraignante puisque l'ensemble $\Delta^{-1}(I_{k-1})$ peut être vide. C'est par exemple le cas lorsque I_{k-1} est une image de type échiquier (un pixel sur deux est blanc, un pixel sur deux est noir) et que le filtre passe-bas possède une distribution gaussienne. En effet, une image de ce type ne peut pas être obtenue par réduction avec ce filtre puisqu'il lisse obligatoirement l'image X_k . Il s'agit d'un cas extrême assez évident, mais l'ensemble $\Delta^{-1}(I_{k-1})$ peut aussi être vide pour des images naturelles et des filtres plus proches du filtre idéal. Dans le cadre général que l'on s'est fixé, il faut pouvoir traiter ce problème. Pour cela, lorsque l'ensemble $\Delta^{-1}(I_{k-1})$ est vide, on s'intéresse aux images proches de I_{k-1} pour lesquelles l'ensemble n'est plus vide. De cette manière, la contrainte d'égalité dans (6.2) est assouplie et on obtient un ensemble plus général qui contient dans tous les cas au moins une solution. Ainsi, nous définissons l'ensemble induit d'une image I_{k-1} par un opérateur de réduction Δ comme l'ensemble des images de résolution k dont la réduction fournit la meilleure approximation de I_{k-1} :

$$,^k = \{X_k \mid \Delta(X_k) \approx I_{k-1}\} \quad (6.3)$$

L'ensemble $,^k$ est l'ensemble des images agrandies de I_{k-1} qui nous intéresse désormais. Lorsqu'il existe au moins une image de résolution k qui, par réduction avec Δ , donne exactement I_{k-1} , on dit que l'ensemble $,^k$ est *consistant*. Dans le cas contraire, on dit qu'il est *inconsistant*.

Pour réaliser l'agrandissement de I_{k-1} par induction sur l'ensemble $,^k$, on projette sur celui-ci une image particulière de résolution k (voir figure 6.1). Etant donnée la nature linéaire de l'opérateur de réduction Δ , l'ensemble $,^k$ a la propriété d'être convexe lorsqu'il est consistant, ce qui garantit l'unicité de la projection. Cette propriété n'est malheureusement pas conservée en cas d'inconsistance. Pour que $,^k$ soit également convexe dans le cas inconsistant, nous apporterons des modifications dans sa formulation en section 6.2.3. Finalement, on vérifie bien qu'en variant l'image projetée, on peut obtenir tous les éléments de $,^k$.

Remarquons que dans la méthode d'interpolation probabiliste de Schultz *et al.* (cf. section 3.5.3) et surtout dans celle basée sur la variation totale de Guichard *et al.* (cf. section 3.5.4), on retrouve également la notion d'ensemble d'agrandissement contraint en réduction. Ces deux méthodes se distinguent cependant de notre approche car leurs auteurs s'intéressent à un sous-ensemble de solutions qui minimisent une certaine fonctionnelle liée à la régularité de l'image (cette notion de régularité est liée à celle de dérivée). Par ailleurs, dans la méthode de Schultz *et al.*, l'opérateur de réduction est très simpliste puisque le filtre passe-bas est un filtre réalisant une moyenne et qu'il n'y a pas de sous-échantillonnage. Dans la méthode de Guichard *et al.*, l'opérateur de réduction est plus général mais le filtre doit tout de même avoir une réponse fréquentielle non-nulle pour éviter de générer un ensemble de solutions vide (inconsistant). Dans les deux cas, le problème de la quantification des niveaux de gris n'est pas pris en compte. Enfin, dans la méthode de Schultz *et al.* et dans

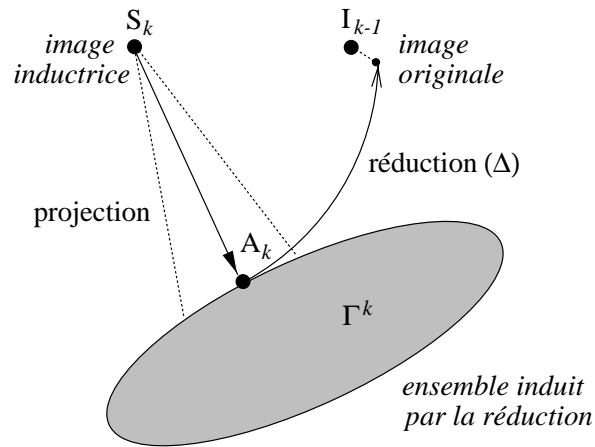


Figure 6.1 – Induction de I_{k-1} sur l'ensemble Γ^k par projection d'une image inductrice S_k .

celle de Guichard *et al.*, une solution du sous-ensemble minimisant la fonctionnelle est obtenue par un algorithme de descente en gradient. Cet algorithme exploite une solution préalable qui doit déjà satisfaire la contrainte de réduction. Dans le premier cas, cette solution est obtenue par une interpolation au plus proche voisin (cf. section 2.3.1) et dans le deuxième cas, elle est obtenue avec la méthode du *zero-padding* (cf. section 2.3.2).

Dans notre approche, nous nous intéressons à toutes les solutions de l'ensemble induit d'une réduction et il n'y a pas de contrainte particulière sur l'image projetée sur cet ensemble : elle est *a priori* quelconque. Ceci nous amène notamment à introduire la notion de projection exacte (orthogonale) ou approchée (non-orthogonale) sur cet ensemble. De plus, dans notre approche, le filtre peut être quelconque puisque nous traitons le cas d'inconsistance de l'ensemble, k . Il faut aussi souligner que nous prenons en compte la quantification des niveaux de gris sur des valeurs entières.

Dans la section 6.2, nous caractérisons précisément l'ensemble, k , que ce soit dans le cas consistant ou dans le cas inconsistant. Nous formulons ensuite deux types de projection possibles sur, k : une projection exacte et une projection approchée. Dans la section 6.3, nous développons cinq méthodes de projections partielles itérées dont certaines ne peuvent être utilisées que dans le cas consistant. Ces méthodes réalisent soit une projection exacte, soit une projection approchée sur, k . Dans la section 6.4, nous étudions le comportement de ces méthodes de projections partielles d'un point de vue expérimental à la fois dans les cas de consistance et d'inconsistance de, k . Enfin, dans la section 6.5, nous généralisons les développements précédents pour réaliser une induction avec un facteur d'agrandissement supérieur à deux.

6.2 Formulation de l'induction

Dans cette section, nous mettons en place le cadre théorique relatif à l'induction d'une image sur l'ensemble \mathcal{I}^k défini par (6.3). Tout d'abord, nous définissons mathématiquement l'opérateur de réduction Δ . Ensuite, nous caractérisons l'ensemble \mathcal{I}^k dans le cas consistant comme l'intersection d'ensembles élémentaires qui exercent chacun une contrainte locale sur l'ensemble des solutions. Nous le caractérisons dans le cas inconsistant comme étant l'ensemble des images de résolution k qui minimisent la distance à tous les ensembles élémentaires (au sens des moindres carrés). Nous formulons par la suite les propriétés d'une séquence de projections partielles sur les ensembles élémentaires pour réaliser soit une projection exacte (orthogonale), soit une projection approchée (non orthogonale) sur \mathcal{I}^k . Enfin, nous terminons cette section en donnant une interprétation fréquentielle de l'ensemble induit d'une réduction.

6.2.1 Généralités

Soit Ξ un espace de Hilbert réel, doté du produit scalaire $\langle \cdot | \cdot \rangle$, de la norme $\| \cdot \|$ et de la distance d . On définit Ξ^k comme étant un sous-espace vectoriel de Ξ de dimension finie $2^k \times 2^k$. On définit $\Omega^k \subset \Xi^k$ comme étant l'espace des images de résolution k à valeurs dans l'intervalle³ $[0, 255] \subset \mathbb{R}^+$. Il s'agit donc d'un hypercube de l'espace Ξ^k . Pour simplifier les écritures, nous notons $\mathcal{I}_k = \{0, \dots, 2^k - 1\}$ un ensemble d'indices.

6.2.2 La réduction d'une image

Les propriétés de l'ensemble \mathcal{I}^k dépendent directement du choix de l'opérateur de réduction Δ . Celui-ci joue donc un rôle fondamental dans le principe d'induction. Notre définition de l'opérateur de réduction repose sur l'analyse fréquentielle de la réduction que nous avons réalisée dans la section 4.2. Nous avons pu voir en particulier que la décimation spatiale de l'image, c'est-à-dire la suppression d'échantillons, devait être précédée d'une opération de convolution avec un filtre passe-bas. Cette opération permet de supprimer les hautes fréquences de l'image, ce qui évite un repliement du spectre et donc empêche la déstructuration de l'image.

Définition 6.1 (Opérateur de réduction) *Un opérateur de réduction Δ convolue l'image avec un filtre passe-bas, décime l'image d'un facteur 2 (dans chaque direction) et réalise une quantification sur des valeurs entières (niveaux de gris). Ceci se traduit par :*

$$\begin{aligned}
 (\forall k > 0) \quad \Delta : \Omega^k &\rightarrow \Omega^{k-1} \\
 X_k &\mapsto [W * X_k] \downarrow 2 = X_{k-1}
 \end{aligned}
 \tag{6.4}$$

3. La nature discrète des valeurs des pixels sera prise en compte uniquement pour le passage de Ω^k vers Ω^{k-1} (voir la définition d'un opérateur de réduction dans la section 6.2.2).

où W représente le filtre passe-bas, $*$ le produit de convolution, $[\cdot]$ la partie entière inférieure et $\downarrow 2$ la décimation de facteur 2.

Il n'y a pas de restriction particulière sur la taille et les poids du filtre passe-bas W . Toutefois, pour simplifier l'écriture mathématique, nous considérons que le filtre est de taille $(2M + 1) \times (2M + 1)$. On pourrait très facilement étendre les résultats qui suivent au cas d'un filtre de largeur paire. En pratique, on peut prendre pour le filtre W un des filtres que nous avons étudiés dans la section 4.3.

Pour faciliter le développement des méthodes de projections partielles dans la section 6.3, nous reformulons l'opérateur de réduction avec la notation matricielle suivante :

$$(\forall k > 0) \quad \Delta(\vec{X}_k) = [\mathcal{H} \cdot \vec{X}_k] = \vec{X}_{k-1} \quad (6.5)$$

où \vec{X}_k explicite une écriture vectorielle de la matrice X_k , c'est-à-dire : $(\forall (i, j) \in \mathcal{I}_k^2) \vec{X}_k(2^k i + j) = X_k(i, j)$. La matrice \mathcal{H} est une matrice de taille $2^{2(k-1)} \times 2^{2k}$, que nous appelons *matrice de réduction* et que nous définissons par :

$$\mathcal{H} = \begin{bmatrix} {}^t\vec{H}_{k,(0,0)} \\ \vdots \\ {}^t\vec{H}_{k,(0,2^{k-1}-1)} \\ \vdots \\ {}^t\vec{H}_{k,(2^{k-1}-1,0)} \\ \vdots \\ {}^t\vec{H}_{k,(2^{k-1}-1,2^{k-1}-1)} \end{bmatrix} \quad (6.6)$$

où $(H_{k,(i,j)})_{(i,j) \in \mathcal{I}_{k-1}^2}$ est une famille de matrices de taille $2^k \times 2^k$ engendrée par le filtre passe-bas W . Plus précisément, pour tout (i, j) appartenant à \mathcal{I}_{k-1}^2 , une matrice $H_{k,(i,j)}$ positionne le centre du filtre en $(2i, 2j)$. L'algorithme de génération d'une matrice de la famille $(H_{k,(i,j)})$ est donné dans la table 6.1.

La fonction φ permet de valider les calculs sur les bords de l'image. Elle est définie par :

$$\varphi(x, y) = \begin{cases} 0 & \text{si } x \leq 0 \\ x & \text{si } 0 < x < y \\ y - 1 & \text{si } x \geq y \end{cases} \quad (6.7)$$

Ce traitement des bords revient à dupliquer vers l'extérieur les pixels situés en bordure de l'image. Lorsque le filtre est appliqué au niveau de la bordure, on a donc une superposition d'une partie de ses poids.

6.2.3 Caractérisation de l'ensemble induit d'une réduction

Soit une image originale I_{k-1} et un opérateur de réduction Δ tel que nous l'avons défini précédemment. Nous cherchons à présent à caractériser l'ensemble I_k décrit

$$\begin{aligned}
& \mathbf{H}_{k,(i,j)} = \mathbf{0} \\
& \text{pour } n \text{ allant de } -M \text{ à } M \\
& \quad \text{pour } m \text{ allant de } -M \text{ à } M \\
& \quad \quad s = \varphi(2i + m, 2^k) \text{ et } t = \varphi(2j + n, 2^k) \\
& \quad \quad \mathbf{H}_{k,(i,j)}(s, t) = \mathbf{H}_{k,(i,j)}(s, t) + \mathbf{W}(m, n)
\end{aligned}$$

Table 6.1 – Construction d'une matrice $\mathbf{H}_{k,(i,j)}$.

par (6.3). Pour cela, nous différencions le cas consistant où $\Delta^{-1}(\mathbf{I}_{k-1}) \neq \emptyset$ du cas inconsistant où $\Delta^{-1}(\mathbf{I}_{k-1}) = \emptyset$.

Ensemble consistant

Lorsque l'ensemble $,^k$ est consistant, on peut le caractériser plus précisément dans l'espace Ξ^k en utilisant (6.5) et (6.6) :

$$,^k = \Delta^{-1}(\mathbf{I}_{k-1}) = \left\{ \mathbf{X}_k \in \Omega^k \mid (\forall (i, j) \in \mathcal{I}_{k-1}^2) [\langle \vec{\mathbf{H}}_{k,(i,j)} \mid \vec{\mathbf{X}}_k \rangle] = \mathbf{I}_{k-1}(i, j) \right\} \quad (6.8)$$

Avec cette formulation, on remarque que chaque pixel (i, j) de \mathbf{I}_{k-1} exerce une contrainte locale sur les éléments de $,^k$. Chacune de ces contraintes engendre un ensemble de solutions localement admissibles. A partir de ces ensembles, nous construisons une famille $(\Lambda_{(i,j)}^k)_{(i,j) \in \mathcal{I}_{k-1}^2}$ définie par :

$$(\forall (i, j) \in \mathcal{I}_{k-1}^2) \quad \Lambda_{(i,j)}^k = \left\{ \mathbf{X}_k \in \Xi^k \mid [\langle \vec{\mathbf{H}}_{k,(i,j)} \mid \vec{\mathbf{X}}_k \rangle] = \mathbf{I}_{k-1}(i, j) \right\} \quad (6.9)$$

Chaque ensemble $\Lambda_{(i,j)}^k$ contient toutes les images de Ξ^k satisfaisant localement l'équation $\Delta(\mathbf{X}_k) = \mathbf{I}_{k-1}$ au pixel (i, j) . Ces ensembles sont des sous-espaces de Ξ^k délimités par deux hyperplans et possèdent par conséquent la propriété de convexité. Nous notons Λ^k l'intersection de cette famille d'ensembles. Nous pouvons alors redéfinir $,^k$ de la manière suivante :

$$,^k = \Delta^{-1}(\mathbf{I}_{k-1}) = \Lambda^k \cap \Omega^k = \left(\bigcap_{(i,j) \in \mathcal{I}_{k-1}^2} \Lambda_{(i,j)}^k \right) \cap \Omega^k \quad (6.10)$$

Dans le cas consistant, $,^k$ représente donc l'intersection d'ensembles convexes élémentaires, comme le montre la figure 6.2(a).

Ensemble inconsistant

Lorsque l'ensemble $,^k$ est inconsistant (figure 6.2(b)), on peut le redéfinir dans l'espace Ξ^k par :

$$,^k = \left\{ \mathbf{X}_k \in \Omega^k \mid (\forall \mathbf{Y}_k \in \Omega^k) d(\Delta(\mathbf{X}_k), \mathbf{I}_{k-1})^2 \leq d(\Delta(\mathbf{Y}_k), \mathbf{I}_{k-1})^2 \right\} \quad (6.11)$$

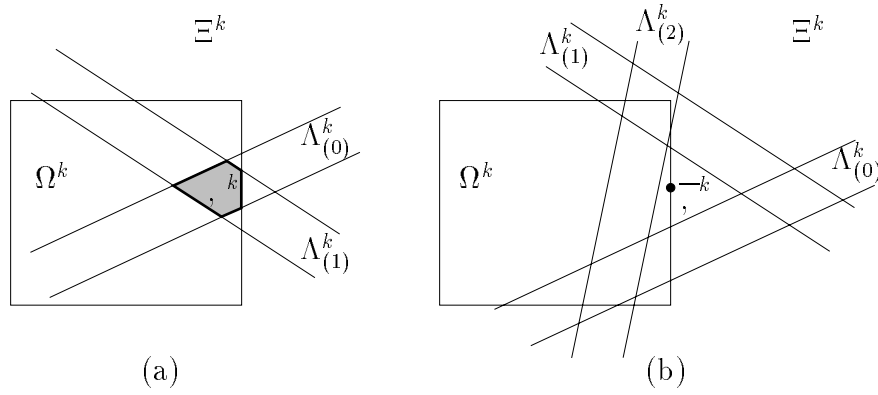


Figure 6.2 – (a) L'ensemble Γ^k est consistant. (b) L'ensemble Γ^k est inconsistant.

Pour caractériser plus précisément cet ensemble dans Ξ^k , nous avons à résoudre un problème de minimisation dans Ξ^{k-1} . Ce problème est difficile, d'autant que l'ensemble des solutions n'est pas forcément convexe dans Ξ^k , ce qui ne garantit plus l'unicité d'une projection. Pour simplifier, nous le remplaçons par un problème de minimisation dans Ξ^k dont les solutions ont l'avantage de former un ensemble convexe. Plus exactement, nous cherchons à obtenir l'ensemble des images de résolution k qui minimisent la distance à $(\Lambda_{(i,j)}^k)$ et Ω^k . Pour cela, nous construisons une fonction $\Phi : \Xi^k \rightarrow \mathbb{R}^+$ qui mesure le degré de proximité d'une image de Ξ^k à la famille $(\Lambda_{(i,j)}^k)$ et à l'hypercube des images Ω^k :

$$(\forall X_k \in \Xi^k) \quad \Phi(X_k) = \frac{1}{2} \left(\sum_{(i,j) \in \mathcal{I}_{k-1}^2} d(X_k, \Lambda_{(i,j)}^k)^2 + d(X_k, \Omega^k)^2 \right) \quad (6.12)$$

L'ensemble \checkmark^k des images de Ξ^k qui minimisent la fonction Φ est donné par :

$$\checkmark^k = \{X_k \in \Xi^k \mid (\forall Y_k \in \Xi^k) \Phi(X_k) \leq \Phi(Y_k)\} \quad (6.13)$$

D'un point de vue théorique, il est possible que \checkmark^k n'ait pas d'intersection commune avec l'espace des images Ω^k . Pour être certain d'obtenir des solutions qui appartiennent à l'espace des images, il faut finalement projeter \checkmark^k sur Ω^k . L'ensemble issu de cette projection est alors défini par :

$$\bar{\checkmark}^k = \{X_k \in \Omega^k \mid (\forall Y_k \in \checkmark^k)(\forall Z_k \in \Omega^k) d(Y_k, X_k)^2 \leq d(Y_k, Z_k)^2\} \quad (6.14)$$

Les trois cas suivants peuvent se présenter :

1. Il existe une image X_k de Ω^k telle que $\Phi(X_k) = 0$.
Dans ce cas, $\bar{\checkmark}^k = \checkmark^k = \checkmark^k = \Delta^{-1}(I_{k-1})$ et $\bar{\checkmark}^k$ est consistant.
2. Il n'existe pas de solution rendant nulle Φ et $\checkmark^k \cap \Omega^k \neq \emptyset$.
Dans ce cas, $\bar{\checkmark}^k = \checkmark^k \neq \checkmark^k$.

3. Il n'existe pas de solution rendant nulle Φ et $\checkmark^k \cap \Omega^k = \emptyset$.
 Dans ce cas, $\overline{\checkmark}^k \neq \checkmark^k \neq \checkmark^k$.

Dans le cas inconsistant, les solutions décrites par \checkmark^k ne minimisent pas la fonction Φ dans l'espace Ξ^k et les réductions associées aux solutions de $\overline{\checkmark}^k$ ne minimisent pas la distance à I_{k-1} dans l'espace Ξ^{k-1} . Bien qu'étant un ensemble approximatif de \checkmark^k dans le cas inconsistant, l'ensemble $\overline{\checkmark}^k$ a sur lui l'avantage pratique d'être mathématiquement caractérisé. Il a de plus la propriété remarquable d'être convexe, ce qui d'un point de vue calculabilité est un avantage considérable.

Proposition 6.1 (Convexité) *Les ensembles \checkmark^k et $\overline{\checkmark}^k$ sont convexes :*

$$(i) \quad \checkmark^k \neq \emptyset \text{ et } \overline{\checkmark}^k \neq \emptyset$$

$$(ii) \quad (\forall \alpha \in]0, 1[) \quad (\forall (X_k, Y_k) \in \checkmark^k \times \checkmark^k) \quad \alpha \cdot X_k + (1 - \alpha) \cdot Y_k \in \checkmark^k$$

$$(\forall \alpha \in]0, 1[) \quad (\forall (X_k, Y_k) \in \overline{\checkmark}^k \times \overline{\checkmark}^k) \quad \alpha \cdot X_k + (1 - \alpha) \cdot Y_k \in \overline{\checkmark}^k$$

Dans la suite, nous délaissions \checkmark^k au profit des ensembles \checkmark^k et $\overline{\checkmark}^k$ qui sont plus faciles à caractériser dans le cas inconsistant.

6.2.4 Projection sur l'ensemble induit d'une réduction

Dans le cas général (consistant ou inconsistant), l'induction de l'image I_{k-1} sur l'ensemble $\overline{\checkmark}^k$ revient d'abord à projeter une image quelconque $A_k^{(0)}$ sur \checkmark^k , puis à projeter l'image résultante sur Ω^k . Cette dernière projection est élémentaire, elle permet d'obtenir dans tous les cas une solution qui appartient à l'espace des images. La projection sur \checkmark^k est beaucoup moins évidente. Nous verrons dans la section 6.3 de quelle manière on peut réaliser cette projection en itérant des combinaisons de projections partielles sur la famille $(\Lambda_{(i,j)}^k)$ et sur l'espace des images Ω^k . Pour le moment, nous nous intéressons uniquement aux propriétés de la séquence d'images $(A_k^{(n)})_{n \geq 0}$ engendrée par cette suite de projections partielles.

L'image $A_k^{(0)}$ est appelée *image inductrice*. L'image de \checkmark^k vers laquelle converge la séquence est appelée *image induite* (voir figure 6.3). Si l'image induite A_k est l'image de \checkmark^k qui minimise la distance à l'image inductrice $A_k^{(0)}$, alors l'image induite est la *projection exacte* (orthogonale) de cette dernière sur \checkmark^k .

Définition 6.2 (Projection exacte) *La projection de $A_k^{(0)}$ sur \checkmark^k est exacte si elle génère une séquence $(A_k^{(n)})_{n \geq 0}$ ayant les propriétés suivantes :*

- (i) *La séquence converge vers une image de \checkmark^k :*

$$\lim_{n \rightarrow \infty} A_k^{(n)} = A_k \implies A_k \in \checkmark^k \implies \Delta(\text{Proj}_{\Omega^k}(A_k)) \approx I_{k-1}$$

- (ii) *L'image induite minimise la distance à l'image inductrice :*

$$(\forall X_k \in \checkmark^k) \quad \|A_k - A_k^{(0)}\| \leq \|X_k - A_k^{(0)}\| \iff \text{Proj}_{\checkmark^k}(A_k^{(0)}) = A_k$$

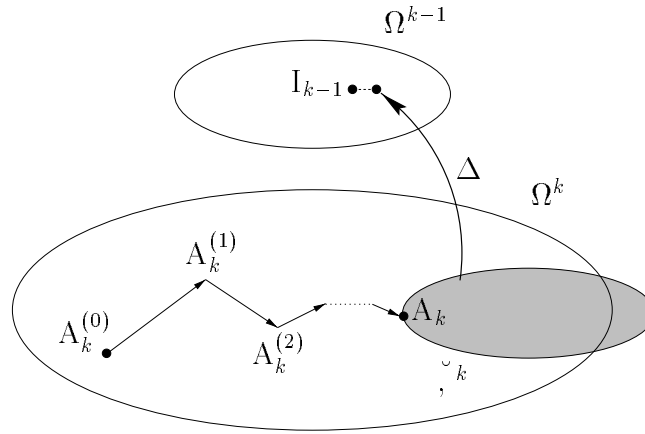


Figure 6.3 – Induction de I_{k-1} sur l'ensemble $\check{\Gamma}^k$: $A_k^{(0)}$ est l'image inductrice et A_k est l'image induite.

Nous définissons un deuxième modèle de projection plus souple où la séquence $(A_k^{(n)})$ est Fejér-monotone [122], ce qui lui donne la propriété de tendre vers tous les éléments de $\check{\Gamma}^k$ à la fois. Généralement, l'image induite par ce modèle n'est pas une projection exacte de l'image inductrice sur $\check{\Gamma}^k$, mais seulement une projection approchée. L'image induite appartient encore à l'ensemble $\check{\Gamma}^k$, mais elle ne minimise plus forcément la distance à l'image inductrice $A_k^{(0)}$.

Définition 6.3 (Projection approchée) La projection de $A_k^{(0)}$ sur $\check{\Gamma}^k$ est approchée si elle génère une séquence $(A_k^{(n)})_{n \geq 0}$ ayant les propriétés suivantes :

- (i) La séquence converge vers une image de $\check{\Gamma}^k$:

$$\lim_{n \rightarrow \infty} A_k^{(n)} = A_k \implies A_k \in \check{\Gamma}^k \implies \Delta(\text{Proj}_{\Omega^k}(A_k)) \approx I_{k-1}$$
- (ii) La séquence est $\check{\Gamma}^k$ -Fejér-monotone :

$$(\forall n \geq 0) (\forall X_k \in \check{\Gamma}^k) \|A_k^{(n+1)} - X_k\| \leq \|A_k^{(n)} - X_k\|$$

Comme nous le verrons dans la partie expérimentale (section 6.4), la projection approchée sur $\check{\Gamma}^k$ est obtenue plus rapidement que la projection exacte.

6.2.5 Interprétation fréquentielle

L'interprétation fréquentielle des projections partielles successives est plutôt difficile. En effet, une projection partielle est par définition localisée dans le domaine spatial. Dans le domaine fréquentiel, elle modifie donc l'ensemble des coefficients du spectre. L'interprétation fréquentielle de la projection globale sur $\check{\Gamma}^k$ n'est pas plus simple. Même lorsque celle-ci est exacte (orthogonale), l'interprétation fréquentielle demeure difficile puisque la minimisation d'une distance dans le domaine spatial n'a pas d'équivalent simple dans le domaine fréquentiel.

Cependant, l'interprétation fréquentielle de l'ensemble $\overline{\cdot}^k$ est déjà plus aisée, du moins lorsque celui-ci est consistant. Dans ce cas, les contraintes imposées aux images X_k de $\overline{\cdot}^k$ peuvent être réparties en deux groupes.

Le premier groupe est déterminé par l'opérateur de réduction Δ défini par l'expression (6.4). Les opérations de filtrage et de décimation dans le domaine spatial et dans le domaine fréquentiel s'écrivent :

$$X_k * W(x, y) = G_k(x, y) \xleftrightarrow{TF} \hat{W}(u, v) \cdot \hat{X}_k(u, v) = \hat{G}_k(u, v) \quad (6.15)$$

$$G_k(x, y) \cdot H_k(x, y) = R_k(x, y) \xleftrightarrow{TF} \hat{G}_k * \hat{H}_k(u, v) = \hat{R}_k(u, v) \quad (6.16)$$

où \hat{X}_k représente la transformée de Fourier de X_k . La décimation spatiale $\downarrow 2$ est réalisée par une multiplication avec la fonction H_k définie par (4.7). En utilisant son équivalent fréquentiel \hat{H}_k donné par (4.8) avec $N = 2^{k-1}$, on obtient l'expression de $\hat{R}_k(u, v)$ suivante⁴ :

$$\begin{aligned} \hat{R}_k(u, v) = & \frac{1}{4} \left[\hat{G}_k(u, v) + \hat{G}_k(u, v + 2^{k-1}) \right. \\ & \left. + \hat{G}_k(u + 2^{k-1}, v) + \hat{G}_k(u + 2^{k-1}, v + 2^{k-1}) \right] \end{aligned} \quad (6.17)$$

Le changement de résolution de R_k à R_{k-1} est donné simplement par :

$$R_{k-1}(x, y) = R_k(2x, 2y) \xleftrightarrow{TF} \hat{R}_{k-1}(u, v) = \hat{R}_k(u, v) \quad (6.18)$$

Enfin, l'opération de quantification des niveaux de gris dans le domaine spatial s'écrit :

$$(\forall (x, y) \in \mathcal{I}_{k-1}^2) \quad I_{k-1}(x, y) \leq R_{k-1}(x, y) < I_{k-1}(x, y) + 1 \quad (6.19)$$

Ces contraintes étant très localisées, on ne dispose pas d'équivalents simples dans le domaine fréquentiel et chaque contrainte est répartie sur l'ensemble du spectre. On peut tout au plus écrire la relation suivante :

$$I_{k-1}(x, y) \leq \frac{4}{2^{2k}} \sum_{u,v=0}^{2^{k-1}-1} \hat{R}_{k-1}(u, v) e^{\frac{2\pi i}{2^{k-1}}(ux+vy)} < I_{k-1}(x, y) + 1 \quad (6.20)$$

Le deuxième groupe de contraintes est lié à l'intervalle des niveaux de gris associé aux pixels de l'image X_k . Chaque valeur doit appartenir à l'intervalle $\{0, \dots, 255\}$. Cette contrainte s'écrit :

$$(\forall (x, y) \in \mathcal{I}_k^2) \quad 0 \leq X_k(x, y) \leq 255 \quad (6.21)$$

Comme la contrainte de quantification des niveaux de gris exprimée par (6.19), cette contrainte n'a pas d'expression simple dans le domaine fréquentiel :

$$0 \leq \frac{1}{2^{2k}} \sum_{u,v=0}^{2^{k-1}-1} \hat{X}_k(u, v) e^{\frac{2\pi i}{2^k}(ux+vy)} \leq 255 \quad (6.22)$$

4. On peut remarquer que l'on obtient ici une généralisation sur deux dimensions de l'expression (3.28).

En définitive, seules les contraintes de filtrage et de décimation sont relativement faciles à interpréter dans le domaine fréquentiel. A partir des expressions (6.15) et (6.17), on voit que chaque fréquence de \hat{R}_{k-1} est en partie liée à quatre fréquences de \hat{X}_k . De plus, si on omet les contraintes (6.19) et (6.21), on remarque que chaque fréquence \hat{X}_k ne dépend que d'une fréquence unique de \hat{R}_{k-1} . Il s'agit évidemment d'une simplification du problème, mais cela montre que les projections sur les contraintes de filtrage et de décimation sont directes dans le domaine fréquentiel. Cet attrait pour le domaine fréquentiel est cependant compromis par les contraintes imposées sur les niveaux de gris puisque celles-ci deviennent beaucoup plus difficiles à appréhender.

Lorsque \hat{W} est le filtre idéal rectangulaire défini par (4.9) avec $N = 2^{k-1}$, l'expression (6.17) se simplifie par :

$$(-2^{k-2} \leq u, v < 2^{k-2}) \quad 4\hat{R}_k(u, v) = \hat{G}_k(u, v) \quad (6.23)$$

Ce cas particulier est intéressant. Il montre qu'avec le filtre idéal, les images X_k de $\check{\cdot}^k$ ont toutes le même contenu fréquentiel dans les basses fréquences et que les hautes fréquences ne sont pas contraintes autrement que par l'espace des images lui-même.

Dans le cas inconsistant, l'interprétation fréquentielle de $\check{\cdot}^k$ est très difficile. En effet, dans ce cas, les images X_k doivent minimiser la fonction Φ , définie par (6.12), qui mesure la distance globale aux contraintes locales de réduction et à l'espace des images. Cette fonction n'a pas d'expression simple dans le domaine fréquentiel.

Cependant, on voit qu'une condition pour que $\check{\cdot}^k$ ne soit pas inconsistant est :

$$(\forall (u, v) \in \mathcal{I}_{k-1}^2) \quad \sum_{s,t=0}^1 \hat{W}(u + s2^{k-1}, v + t2^{k-1}) \neq 0 \quad (6.24)$$

Cette condition est nécessaire mais non suffisante. L'inconsistance peut également être provoquée par l'incompatibilité entre les contraintes (6.17), (6.20) et (6.22).

6.3 Les méthodes de projections partielles

Pour réaliser la projection de $A_k^{(0)}$ sur l'ensemble $\check{\cdot}^k$, nous utilisons des méthodes de projections partielles sur la famille $(\Lambda_{(i,j)}^k)$ et sur l'hypercube Ω^k . Ces méthodes sont dérivées de méthodes générales d'estimation d'une intersection de convexes [16], [39]. Elles sont basées sur des projections alternées ou des projections parallèles. Dans la suite, nous donnons cinq méthodes de projections partielles qui réalisent soit une projection approchée, soit une projection exacte sur $\check{\cdot}^k$. Trois de ces méthodes sont utilisables uniquement lorsque $\check{\cdot}^k$ est consistant, les deux autres le sont dans tous les cas.

Avant de présenter les différentes méthodes de projections partielles, nous commençons par définir les projections élémentaires sur les ensembles de la famille $(\Lambda_{(i,j)}^k)$ et sur l'espace des images Ω^k . Nous discutons ensuite des propriétés de convergence

des projections partielles alternées et de celles des projections partielles parallèles en fonction de la consistance de \checkmark^k .

6.3.1 Les projections élémentaires

La projection d'une image quelconque X_k sur un ensemble convexe \mathcal{C}^k , notée $\text{Proj}_{\mathcal{C}^k}$, donne une image unique définie par :

$$\text{Proj}_{\mathcal{C}^k}(X_k) = \arg \min_{Y_k \in \mathcal{C}^k} d(Y_k, X_k) \quad (6.25)$$

L'image issue de cette projection est l'image du convexe \mathcal{C}^k qui minimise la distance à l'image projetée X_k . Les projections élémentaires suivantes sont basées sur cette définition.

Projection sur un ensemble particulier de la famille $(\Lambda_{(i,j)}^k)$

Les ensembles de la famille $(\Lambda_{(i,j)}^k)$ définis par (6.9) peuvent être redéfinis de manière équivalente par :

$$\Lambda_{(i,j)}^k = \left\{ X_k \in \Xi^k \mid I_{k-1}(i, j) \leq \langle \vec{H}_{k,(i,j)} \mid \vec{X}_k \rangle \leq I_{k-1}(i, j) + 1^- \right\} \quad (6.26)$$

où 1^- est un nombre réel inférieur à 1, mais aussi proche que possible de 1.

Les ensembles donnés par (6.26) sont tous enveloppés par deux hyperplans. D'après (6.25), projeter une image X_k sur l'un de ces ensembles revient à projeter cette image sur l'hyperplan le plus proche.

La projection de X_k sur un hyperplan $\mathcal{H}^k = \{Z_k \in \Xi^k \mid (\delta \in \mathbb{R}, Y_k \in \Xi^k) \langle \vec{Y}_k \mid \vec{Z}_k \rangle = \delta\}$ est donnée par l'équation suivante [39] :

$$(\forall X_k \in \Xi^k) \quad \text{Proj}_{\mathcal{H}^k}(X_k) = X_k + \frac{\delta - \langle \vec{Y}_k \mid \vec{X}_k \rangle}{\|\vec{Y}_k\|^2} \vec{Y}_k \quad (6.27)$$

Nous définissons alors la projection d'une image quelconque $X_k \in \Xi^k$ sur un ensemble de la famille $(\Lambda_{(i,j)}^k)$ par :

$$(\forall (i, j) \in \mathcal{I}_{k-1}^2) \quad \text{Proj}_{\Lambda_{(i,j)}^k}(X_k) = X_k + \frac{\psi(X_k, i, j)}{\|\vec{H}_{k,(i,j)}\|^2} \vec{H}_{k,(i,j)} \quad (6.28)$$

où la fonction ψ est donnée par la relation suivante :

$$\psi(X_k, i, j) = (I_{k-1}(i, j) - \langle \vec{H}_{k,(i,j)} \mid \vec{X}_k \rangle)^+ - (\langle \vec{H}_{k,(i,j)} \mid \vec{X}_k \rangle - I_{k-1}(i, j) - 1^-)^+ \quad (6.29)$$

où $(\forall r \in \mathbb{R}) r^+ = \max\{r, 0\}$ est appelée partie positive de r .

Projection sur l'espace des images Ω^k

En prenant des niveaux de gris compris dans l'intervalle $[0, 255]$, l'espace des images se décrit par :

$$\Omega^k = \left\{ X_k \in \Xi^k \mid (\forall (i, j) \in \mathcal{I}_k^2) 0 \leq X_k(i, j) \leq 255 \right\} \quad (6.30)$$

La projection d'une image quelconque $X_k \in \Xi^k$ sur l'espace Ω^k donne une image $\text{Proj}_{\Omega^k}(X_k) = Y_k$ telle que :

$$(\forall (i, j) \in \mathcal{I}_k^2) \quad Y_k(i, j) = \begin{cases} 0 & \text{si } X_k(i, j) < 0 \\ 255 & \text{si } X_k(i, j) > 255 \\ X_k(i, j) & \text{sinon} \end{cases} \quad (6.31)$$

6.3.2 Projections alternées vs projections parallèles

Lorsque l'ensemble \checkmark^k est consistant, les projections alternées ou parallèles sur les ensembles élémentaires convergent (sous certaines conditions) vers un élément de l'intersection de ces ensembles.

Lorsque l'ensemble \checkmark^k est inconsistant, les projections alternées génèrent une séquence $(A_k^{(n)})$ stationnaire qui oscille entre les ensembles convexes élémentaires donnés par la famille $(\Lambda_{(i,j)}^k)$ et Ω^k . On peut trouver pour chacun de ces ensembles élémentaires, une sous-séquence de $(A_k^{(n)})$ qui converge vers une image de cet ensemble, comme l'illustre la figure 6.4(a). Cependant, la séquence ne converge pas vers une solution qui minimise la distance à tous les ensembles élémentaires, c'est-à-dire qui minimise la fonction Φ . En fait, seules les projections parallèles (sans extrapolation) ont cette propriété, comme le montre la figure 6.4(b).

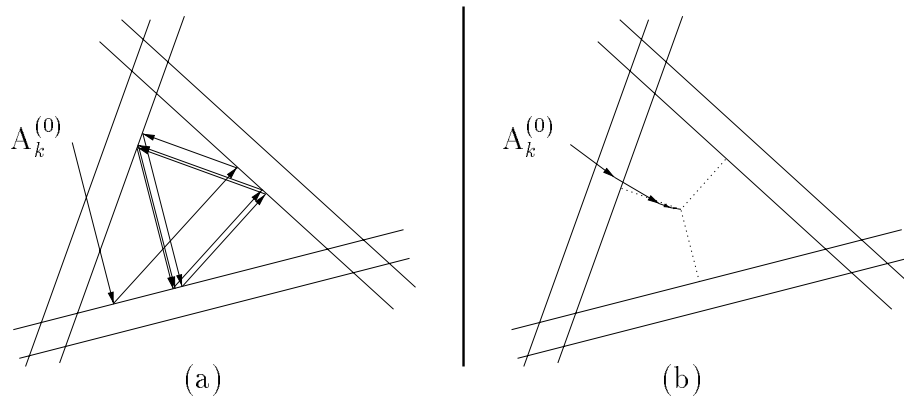


Figure 6.4 – Cas inconsistant : (a) convergence des projections alternées, (b) convergence des projections parallèles.

6.3.3 Projection approchée par des projections partielles alternées

Lorsque \checkmark^k est consistant, la méthode de projection la plus simple consiste à faire des projections partielles alternées entre les ensembles de la famille $(\Lambda_{(i,j)}^k)$ et l'ensemble Ω^k . Plus précisément, l'image inductrice $A_k^{(0)}$ est projetée successivement sur tous les ensembles de la famille $(\Lambda_{(i,j)}^k)$ et l'image résultante est projetée sur Ω^k .

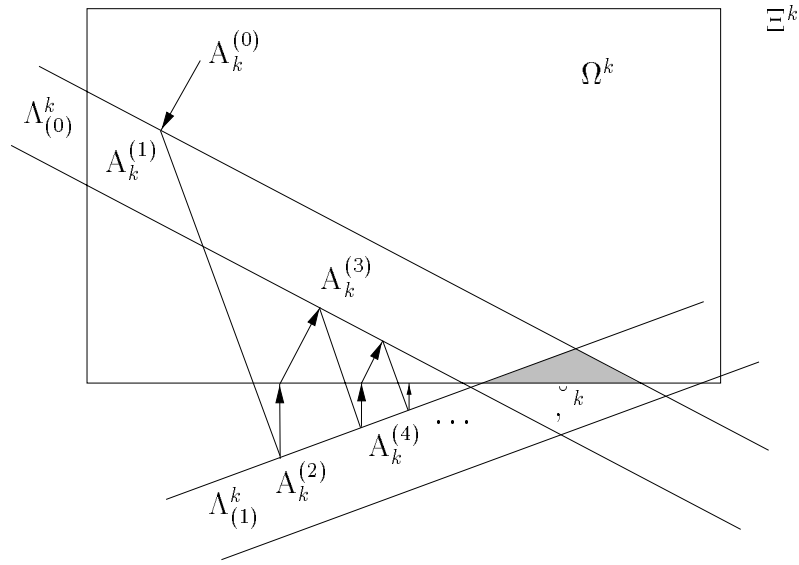


Figure 6.5 – Projection approchée sur un ensemble consistant : méthode des projections partielles alternées (non relaxée dans cet exemple).

Cette série de projections est itérée jusqu'à ce qu'un certain critère de convergence soit satisfait. Ce principe est illustré par un exemple sur la figure 6.5. L'ensemble \checkmark^k est ici représenté par l'intersection de trois ensembles élémentaires (domaine grisé) : $\Lambda_{(0)}^k$, $\Lambda_{(1)}^k$ et Ω^k . Dans cet exemple simplificateur, l'image induite minimise la distance à l'image inductrice. Toutefois, tel n'est pas toujours le cas. On pourrait obtenir un contre-exemple en gardant le même ordre dans les projections et en déplaçant suffisamment $A_k^{(0)}$ vers la droite.

La méthode des projections alternées est définie par la séquence $(A_k^{(n)})_{n \geq 0}$ suivante :

$$(\forall n \in \mathbb{N}) \quad B_k^{(n+1)} = A_k^{(n)} + \lambda^{(n)} \left(\text{Proj}_{\Lambda_{(i_n, j_n)}^k} (A_k^{(n)}) - A_k^{(n)} \right) \quad (6.32)$$

$$(\forall n \in \mathbb{N}) \quad \begin{cases} A_k^{(n+1)} = \text{Proj}_{\Omega^k} (B_k^{(n+1)}) & \text{si } i_n = j_n = 2^{k-1} - 1 \\ A_k^{(n+1)} = B_k^{(n+1)} & \text{sinon} \end{cases} \quad (6.33)$$

$$i_n = n \bmod 2^{k-1} \quad j_n = \lfloor \frac{n}{2^{k-1}} \rfloor \bmod 2^{k-1}$$

Le coefficient $\lambda^{(n)}$ est un paramètre de relaxation qui agit sur la vitesse de convergence. Il est simplement soumis à la contrainte :

$$(\forall n \in \mathbb{N}) \quad 0 < \lambda^{(n)} < 2 \quad (6.34)$$

On parle de sous-relaxation lorsque $\lambda^{(n)} < 1$ et de sur-relaxation lorsque $\lambda^{(n)} > 1$. Le cas où $\lambda^{(n)} = 1$ correspond à un modèle non relaxé. La méthode des projections alternées a l'inconvénient de converger lentement lorsque les angles formés par

les hyperplans de la famille $(\Lambda_{(i,j)}^k)$ sont trop faibles. Une sur-relaxation maximale ($\lambda^{(n)} \approx 2$) permet de corriger ce problème. Cependant, pour se rapprocher de la projection exacte de $\mathbf{A}_k^{(0)}$ sur $\check{\cdot}^k$, on a plutôt intérêt à faire une sous-relaxation : plus $\lambda^{(n)}$ est proche de 0, et plus on minimise la distance $d(\mathbf{A}_k^{(\infty)}, \text{Proj}_{\Gamma^k}(\mathbf{A}_k^{(0)}))$. Ceci signifie que la qualité et la rapidité de la projection varient en sens contraire suivant le choix du paramètre de relaxation $\lambda^{(n)}$. Pour contenter au mieux ces deux contraintes, nous ferons en pratique une sur-relaxation progressive (voir les résultats expérimentaux en section 6.4).

Proposition 6.2 (Convergence [63]) *La méthode des projections alternées définie par (6.32), (6.33) et (6.34), produit une séquence $(\mathbf{A}_k^{(n)})_{n \geq 0}$ Fejér-monotone qui converge vers une image de l'ensemble $\check{\cdot}^k$ lorsque celui-ci est consistant.*

6.3.4 Projection approchée par des projections partielles parallèles

Pour le moment, supposons que $\check{\cdot}^k$ est consistant. Dans ce cas, l'intersection de n ensembles convexes peut être vue comme l'intersection engendrée par le produit cartésien de ces n ensembles avec un convexe diagonal de même dimension. Dans l'espace produit, créé en dupliquant n fois l'espace original, une suite de projections alternées entre l'ensemble produit et le convexe diagonal permet d'obtenir un élément de leur intersection. L'opération équivalente dans l'espace original revient à faire des projections parallèles sur les n ensembles élémentaires. Ce formalisme permettant d'étudier les méthodes de projections parallèles est dû à Pierra [104], [105].

Soit $\Xi^{k-1} \times \Xi^k$ l'espace créé par le produit cartésien de $2^{k-1} \times 2^{k-1} + 1$ copies de l'espace Ξ^k . On dote cet espace du produit scalaire $\langle\langle \cdot | \cdot \rangle\rangle = \sum \langle \cdot | \cdot \rangle$, de la norme $\|\cdot\|$ et de la distance \mathbf{d} . Le produit cartésien des ensembles de la famille $(\Lambda_{(i,j)}^k)$ est donné dans l'espace Ξ^{k-1} par :

$$\Lambda^{k-1} = \left\{ \mathbf{X}_{k-1} \in \Xi^{k-1} \mid (\forall (i,j) \in \mathcal{I}_{k-1}^2) \mathbf{X}_{k-1}(i,j) \in \Lambda_{(i,j)}^k \right\} \quad (6.35)$$

On appelle *convexe diagonal* \mathcal{D} , le sous-espace vectoriel de $\Xi^{k-1} \times \Xi^k$ défini par :

$$\mathcal{D} = \left\{ (X_k, X_k, \dots, X_k) \in \Xi^{k-1} \times \Xi^k \mid X_k \in \Xi^k \right\} \quad (6.36)$$

Trouver une image dans $\check{\cdot}^k$ revient dans l'espace $\Xi^{k-1} \times \Xi^k$ à trouver une image dans $\check{\Gamma} = (\Lambda^{k-1} \times \Omega^k) \cap \mathcal{D}$. Pour cela, on utilise une méthode de projections alternées sur les ensembles $(\Lambda^{k-1} \times \Omega^k)$ et \mathcal{D} (voir figure 6.6).

En fixant l'image inductrice par $\mathbf{A}^{(0)} \in \mathcal{D}$, la méthode des projections alternées produit une séquence dans $\Xi^{k-1} \times \Xi^k$ définie par ($\forall n \in \mathbb{N}$) :

$$\mathbf{A}^{(n+1)} = \mathbf{A}^{(n)} + \lambda^{(n)} \left(\text{Proj}_{\mathcal{D}} \circ \text{Proj}_{\Lambda^{k-1} \times \Omega^k} (\mathbf{A}^{(n)}) - \mathbf{A}^{(n)} \right) \quad (6.37)$$

où $\lambda^{(n)}$ est un paramètre de relaxation défini comme dans (6.34).

Proposition 6.3 (Convergence) *La méthode des projections alternées définie par (6.37) et (6.34) produit une séquence $(\mathbf{A}^{(n)})_{n \geq 0}$ qui converge vers l'ensemble $\check{\Gamma}$, lorsque celui-ci est consistant.*

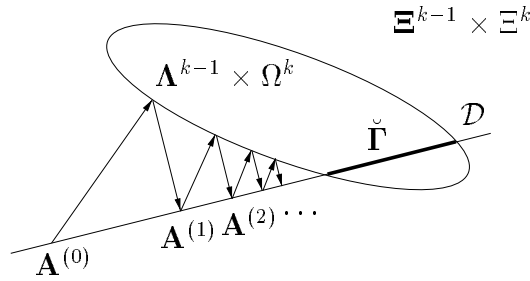


Figure 6.6 – Projection approchée sur un ensemble consistant : méthode des projections partielles alternées dans l'espace $\Xi^{k-1} \times \Xi^k$.

Maintenant, exprimons cette méthode de projections dans l'espace original Ξ^k . D'après [104], nous avons :

$$\begin{cases} (\forall \mathbf{X} \in \mathcal{D}) & \text{Proj}_{\Lambda^{k-1} \times \Omega^k}(\mathbf{X}) = \left((\text{Proj}_{\Lambda_{(i,j)}^k}(X_k))_{(i,j) \in \mathcal{I}_{k-1}^2}, \text{Proj}_{\Omega^k}(X_k) \right) \\ (\forall (\mathbf{X}_{k-1}, Y_k) \in \Lambda^{k-1} \times \Omega^k) & \text{Proj}_{\mathcal{D}}(\mathbf{X}) = \left(\frac{1}{2^{2(k-1)} + 1} \sum_{(i,j) \in \mathcal{I}_{k-1}^2} \mathbf{X}_{k-1}(i,j) + Y_k \right) \end{cases} \quad (6.38)$$

Ainsi, la relation (6.37) s'exprime dans Ξ^k par :

$$\Lambda_k^{(n+1)} = \Lambda_k^{(n)} + \frac{\lambda^{(n)}}{2^{2(k-1)} + 1} \left[\sum_{(i,j) \in \mathcal{I}_{k-1}^2} \left(\text{Proj}_{\Lambda_{(i,j)}^k}(\Lambda_k^{(n)}) - \Lambda_k^{(n)} \right) + \left(\text{Proj}_{\Omega^k}(\Lambda_k^{(n)}) - \Lambda_k^{(n)} \right) \right] \quad (6.39)$$

Nous obtenons donc une séquence $(\Lambda_k^{(n)})$ construite en itérant une combinaison linéaire de projections sur la famille $(\Lambda_{(i,j)}^k)$ et sur l'espace des images Ω^k . Pour cette raison, la méthode associée à (6.37) est appelée *méthode des projections parallèles* dans l'espace original Ξ^k .

Proposition 6.4 (Convergence [104]) *La méthode des projections parallèles définie par (6.39) et (6.34) produit une séquence $(\Lambda_k^{(n)})_{n \geq 0}$ Fejér-monotone qui converge vers une image de l'ensemble \cdot, \checkmark^k , lorsque celui-ci est consistant.*

Pour analyser le cas inconsistant de \cdot, \checkmark^k , on construit la fonction $\Phi : \mathcal{D} \rightarrow \mathbb{R}^+$ qui mesure le degré de proximité d'un élément de \mathcal{D} à l'ensemble $\Lambda^{k-1} \times \Omega^k$:

$$\Phi(\mathbf{X}) = \frac{1}{2} \mathbf{d}(\mathbf{X}, \Lambda^{k-1} \times \Omega^k)^2 \quad (6.40)$$

Définissons également \mathcal{F} comme l'ensemble des points fixes de $\text{Proj}_{\mathcal{D}} \circ \text{Proj}_{\Lambda^{k-1} \times \Omega^k}$ (voir figure 6.7), c'est-à-dire :

$$\mathcal{F} = \left\{ \mathbf{X} \in \mathcal{D} \mid \text{Proj}_{\mathcal{D}}(\text{Proj}_{\Lambda^{k-1} \times \Omega^k}(\mathbf{X})) = \mathbf{X} \right\} \quad (6.41)$$

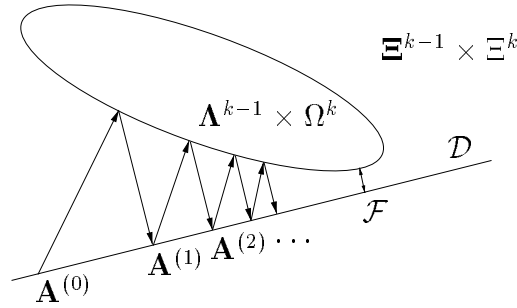


Figure 6.7 – Projection approchée sur un ensemble inconsistant : méthode des projections partielles alternées dans l'espace $\Xi^{k-1} \times \Xi^k$.

Proposition 6.5 (Equivalences [41]) *L'ensemble des points fixes \mathcal{F} coïncide avec l'ensemble des éléments de \mathcal{D} qui minimisent la fonction Φ . De plus, minimiser la fonction Φ dans l'espace original Ξ^k revient à minimiser la fonction Φ dans l'espace produit $\Xi^{k-1} \times \Xi^k$.*

Cette proposition signifie que l'ensemble \checkmark^k a comme équivalent dans l'espace $\Xi^{k-1} \times \Xi^k$ l'ensemble des points fixes \mathcal{F} .

Proposition 6.6 (Convergence [41]) *La méthode des projections alternées donnée par (6.37), produit une séquence $(\mathbf{A}^{(n)})_{n \geq 0}$ qui converge vers l'ensemble des points fixes \mathcal{F} dans l'espace $\Xi^{k-1} \times \Xi^k$.*

Dans l'espace original Ξ^k , cette proposition se traduit par la proposition de convergence suivante.

Proposition 6.7 (Convergence [41]) *La méthode des projections parallèles définie par (6.39) et (6.34) produit une séquence $(\Lambda_k^{(n)})_{n \geq 0}$ Fejér-monotone qui converge vers une image de l'ensemble \checkmark^k , lorsque celui-ci est inconsistant.*

Amélioration de la convergence

Le coefficient $\frac{1}{2^{2(k-1)} + 1}$ qui apparaît dans (6.38) et (6.39) traduit le barycentre équilibré des $2^{2(k-1)} + 1$ projections faites sur la famille $(\Lambda_{(i,j)}^k)$ et sur Ω^k . En pratique, ce coefficient n'est pas optimal pour les deux raisons suivantes : la première est que le nombre de projections effectives, pour une itération donnée, peut être inférieur au nombre total de projections ($2^{2(k-1)} + 1$) ; la seconde raison est que les matrices de la famille $(H_{k,(i,j)})$ sont creuses lorsque la taille du filtre W est beaucoup plus petite que celle de l'espace Ξ^k (cf. table 6.1). Dans ce cas, les projections se font dans des espaces partiels.

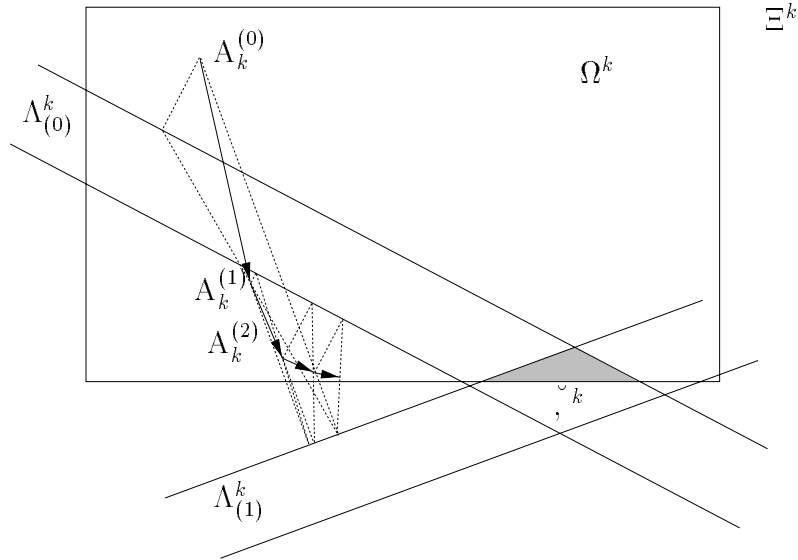


Figure 6.8 – Projection approchée sur un ensemble consistant : méthode des projections partielles parallèles dans Ξ^k (non relaxée dans cet exemple).

Pour être optimal, le coefficient barycentrique doit prendre en compte de ces deux observations. Pour cela, nous construisons deux matrices : une matrice $\mathcal{R}_k^{(n)}$ de dimensions $2^k \times 2^k$ et une matrice $\mathcal{R}_{k-1}^{max(n)}$ de dimensions $2^{k-1} \times 2^{k-1}$. La première matrice est appelée *matrice de recouvrements*. Elle identifie les recouvrements générés par la superposition des matrices $(H_{k,(i,j)})$: l'élément $\mathcal{R}_k^{(n)}(s, t)$ traduit le nombre de fois où $H_{k,(i,j)}(s, t)$ n'est pas nul quand on fait varier (i, j) dans l'intervalle \mathcal{I}_{k-1}^2 . Nous donnons l'algorithme de construction de $\mathcal{R}_k^{(n)}$ dans la table 6.2. La deuxième matrice est appelée *matrice de recouvrement maximal*. Elle se construit à partir de la matrice de recouvrements, en calculant le recouvrement maximal associé à chaque matrice de la famille $(H_{k,(i,j)})$. L'élément $\mathcal{R}_{k-1}^{max(n)}(i, j)$ représente le coefficient barycentrique associé à la projection de $A_k^{(n)}$ sur l'ensemble $\Lambda_{(i,j)}^k$. L'algorithme de construction de $\mathcal{R}_{k-1}^{max(n)}$ est donné dans la table 6.3.

A partir de la matrice de recouvrement maximal, nous obtenons cette version améliorée de la méthode des projections parallèles :

$$A_k^{(n+1)} = A_k^{(n)} + \lambda^{(n)} \left[\sum_{(i,j) \in \mathcal{I}_{k-1}^2} \frac{1}{\sigma_{(i,j)}^{(n)}} \left(\text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)}) - A_k^{(n)} \right) + \frac{1}{\sigma'^{(n)}} \left(\text{Proj}_{\Omega^k} (A_k^{(n)}) - A_k^{(n)} \right) \right] \quad (6.42)$$

$$\left\{ \begin{array}{l} \sigma_{(i,j)}^{(n)} = \mathcal{R}_{k-1}^{max(n)}(i, j) \text{ si } \text{Proj}_{\Omega^k}(A_k^{(n)}) - A_k^{(n)} = 0 \\ \sigma_{(i,j)}^{(n)} = \mathcal{R}_{k-1}^{max(n)}(i, j) + 1 \text{ sinon} \end{array} \right. \quad \sigma^{(n)} = \max_{(i,j) \in \mathcal{I}_{k-1}^2} \mathcal{R}_{k-1}^{max(n)}(i, j) + 1 \quad (6.43)$$

Dans la partie expérimentale, c'est cette version de la méthode des projections parallèles que nous utiliserons. Son principe est illustré par la figure 6.8.

$$\begin{array}{l} \mathcal{R}_k^{(n)} = \mathbf{0} \\ \text{pour } j \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\ \quad \text{pour } i \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\ \quad \quad \text{si } \text{Proj}_{\Lambda_{(i,j)}^k}(A_k^{(n)}) - A_k^{(n)} \neq \mathbf{0} \\ \quad \quad \quad \text{pour } t \text{ allant de } \varphi(2j - M, 2^k) \text{ à } \varphi(2j + M, 2^k) \\ \quad \quad \quad \quad \text{pour } s \text{ allant de } \varphi(2i - M, 2^k) \text{ à } \varphi(2i + M, 2^k) \\ \quad \quad \quad \quad \quad \mathcal{R}_k^{(n)}(s, t) = \mathcal{R}_k^{(n)}(s, t) + 1 \end{array}$$

Table 6.2 – Construction de la matrice de recouvrements $\mathcal{R}_k^{(n)}$.

$$\begin{array}{l} \mathcal{R}_{k-1}^{max(n)} = \mathbf{0} \\ \text{pour } j \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\ \quad \text{pour } i \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\ \quad \quad \text{si } \text{Proj}_{\Lambda_{(i,j)}^k}(A_k^{(n)}) - A_k^{(n)} \neq \mathbf{0} \\ \quad \quad \quad \text{pour } t \text{ allant de } \varphi(2j - M, 2^k) \text{ à } \varphi(2j + M, 2^k) \\ \quad \quad \quad \quad \text{pour } s \text{ allant de } \varphi(2i - M, 2^k) \text{ à } \varphi(2i + M, 2^k) \\ \quad \quad \quad \quad \quad \text{si } \mathcal{R}_k^{(n)}(s, t) > \mathcal{R}_{k-1}^{max(n)}(i, j) \\ \quad \quad \quad \quad \quad \quad \mathcal{R}_{k-1}^{max(n)}(i, j) = \mathcal{R}_k^{(n)}(s, t) \end{array}$$

Table 6.3 – Construction de la matrice de recouvrement maximal $\mathcal{R}_{k-1}^{max(n)}$.

6.3.5 Projection approchée par des projections partielles parallèles avec extrapolation

En pratique, lorsque \checkmark^k est consistant, la méthode des projections parallèles présentée précédemment converge plus lentement que la méthode des projections alternées. Cependant, l'expérience montre que la convergence peut-être nettement améliorée en prenant un paramètre de relaxation $\lambda^{(n)}$ supérieur à 2. En fait, si l'on veut garder la propriété de Fejér-monotonie de la séquence $(A_k^{(n)})$ avec une méthode de

projections alternées, il faut obligatoirement que $\lambda^{(n)}$ demeure dans l'intervalle $]0, 2[$. Cependant, cette contrainte est suffisante mais non nécessaire dans le cas d'une méthode de projections parallèles. Ceci est essentiellement dû à la combinaison linéaire des projections qui donne localement davantage de liberté et qui, par conséquent, permet d'élargir l'intervalle de définition de $\lambda^{(n)}$.

Pour déterminer le paramètre de relaxation $\lambda^{(n)}$ optimal, nous nous basons sur les travaux de Pierra [104], [105], [42]. Reconsidérons l'espace $\Xi^{k-1} \times \Xi^k$ que nous avons présenté dans la section 6.3.4. Définissons $\mathbf{G}^{(n)} = \text{Proj}_{\Lambda^{k-1} \times \Omega^k}(\mathbf{A}^{(n)})$ et $\mathbf{D}^{(n)} = \text{Proj}_{\mathcal{D}}(\mathbf{G}^{(n)})$ comme l'illustre la figure 6.9. L'hyperplan $\mathcal{H}^{(n)}$ prenant appui sur le convexe $\Lambda^{k-1} \times \Omega^k$ en $\mathbf{G}^{(n)}$ traverse le convexe diagonal \mathcal{D} en un point $\mathbf{E}^{(n)}$ qui définit l'extrapolé.

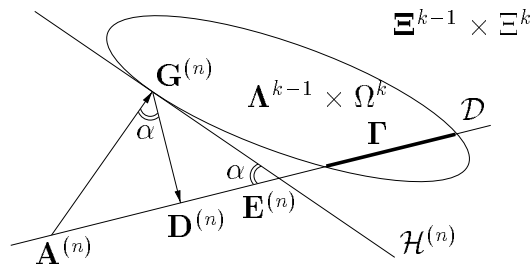


Figure 6.9 – Projection sur un ensemble consistant : Méthode des projections partielles alternées avec extrapolation dans l'espace $\Xi^{k-1} \times \Xi^k$.

Cherchons maintenant à exprimer le coefficient d'extrapolation nécessaire pour que l'on ait finalement $\mathbf{A}^{(n+1)} = \mathbf{E}^{(n)}$. Nous avons les relations suivantes :

$$\frac{1}{\sin \alpha} = \frac{\| \mathbf{E}^{(n)} - \mathbf{A}^{(n)} \|}{\| \mathbf{G}^{(n)} - \mathbf{A}^{(n)} \|} = \frac{\| \mathbf{G}^{(n)} - \mathbf{A}^{(n)} \|}{\| \mathbf{D}^{(n)} - \mathbf{A}^{(n)} \|} \quad (6.44)$$

Nous définissons le coefficient d'extrapolation $\beta^{(n)}$ par :

$$\beta^{(n)} = \frac{\| \mathbf{E}^{(n)} - \mathbf{A}^{(n)} \|}{\| \mathbf{D}^{(n)} - \mathbf{A}^{(n)} \|} = \frac{\| \mathbf{G}^{(n)} - \mathbf{A}^{(n)} \|^2}{\| \mathbf{D}^{(n)} - \mathbf{A}^{(n)} \|^2} = \frac{\| \text{Proj}_{\Lambda^{k-1} \times \Omega^k}(\mathbf{A}^{(n)}) - \mathbf{A}^{(n)} \|^2}{\| \text{Proj}_{\mathcal{D}} \circ \text{Proj}_{\Lambda^{k-1} \times \Omega^k}(\mathbf{A}^{(n)}) - \mathbf{A}^{(n)} \|^2} \quad (6.45)$$

Il se formule dans l'espace original Ξ^k par la relation suivante :

$$\beta^{(n)} = (2^{2(k-1)} + 1) \frac{\sum_{(i,j) \in \mathcal{I}_{k-1}^2} \left(\left\| \text{Proj}_{\Lambda_{(i,j)}^k}(\mathbf{A}_k^{(n)}) - \mathbf{A}_k^{(n)} \right\|^2 + \left\| \text{Proj}_{\Omega^k}(\mathbf{A}_k^{(n)}) - \mathbf{A}_k^{(n)} \right\|^2 \right)}{\left\| \sum_{(i,j) \in \mathcal{I}_{k-1}^2} \left(\text{Proj}_{\Lambda_{(i,j)}^k}(\mathbf{A}_k^{(n)}) - \mathbf{A}_k^{(n)} \right) + \left(\text{Proj}_{\Omega^k}(\mathbf{A}_k^{(n)}) - \mathbf{A}_k^{(n)} \right) \right\|^2} \quad (6.46)$$

La méthode des projections parallèles extrapolée est définie par (6.39) avec un paramètre de relaxation $\lambda^{(n)}$ soumis à la contrainte :

$$(\forall n \in \mathbb{N}) \quad 0 < \lambda^{(n)} < 2\beta^{(n)} \quad (6.47)$$

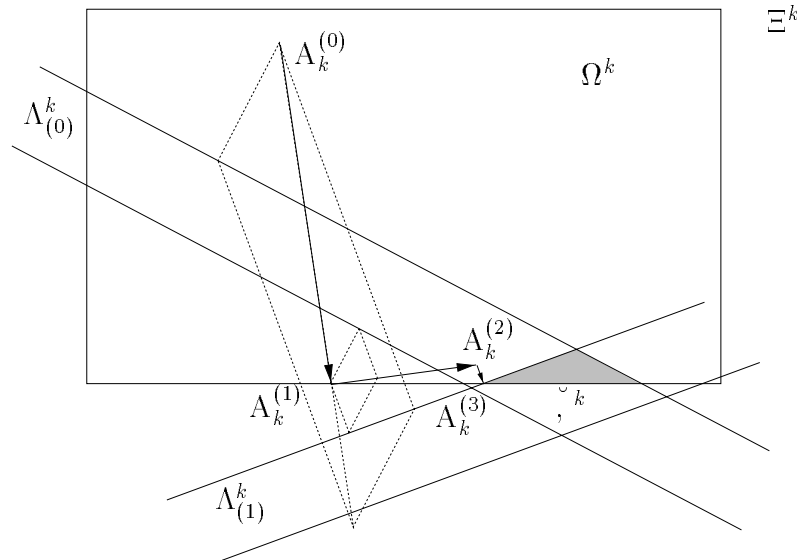


Figure 6.10 – Projection approchée sur un ensemble consistant : méthode de projections partielles parallèles extrapolée dans Ξ^k (non relaxée dans cet exemple).

Le principe de la méthode est illustré dans l'espace Ξ^k sur la figure 6.10.

On remarque que le coefficient barycentrique $\frac{1}{2^{2(k-1)+1}}$ qui apparaît dans (6.39) est finalement remplacé par un coefficient d'extrapolation optimal. Il est donc ici inutile d'améliorer le coefficient barycentrique comme nous l'avons fait pour la méthode des projections parallèles de la section 6.3.4.

En raison de son principe de relaxation par extrapolation, la méthode des projections parallèles extrapolée converge beaucoup plus vite, surtout dans les premières itérations. Cependant, comme Pierra [105], nous remarquons à partir d'expérimentations que lorsque $A_k^{(m)}$ est très proche de \checkmark_k , la séquence $(A_k^{(n)})_{n \geq m}$ a tendance à osciller entre les ensembles convexes élémentaires : on obtient un comportement analogue à celui d'une méthode de projections alternées, sans toutefois avoir des projections totales. Pour limiter ce comportement oscillatoire qui ralentit la convergence finale, Pierra suggère de faire un recentrage toutes les trois itérations en divisant de moitié le coefficient d'extrapolation, c'est-à-dire :

$$(\forall n \in \mathbb{N}) \quad \lambda^{(n)} = \begin{cases} \beta^{(n)}/2 & \text{si } n = 2 \bmod 3 \\ \beta^{(n)} & \text{sinon} \end{cases} \quad (6.48)$$

Proposition 6.8 (Convergence [104]) *La méthode des projections parallèles extrapolée définie par (6.39), (6.46) et (6.47) produit une séquence $(A_k^{(n)})_{n \geq 0}$ Fejér-monotone qui converge vers \checkmark_k , lorsque celui-ci est consistant.*

6.3.6 Projection exacte par des projections partielles alternées

Les trois méthodes précédentes ont l'inconvénient de converger vers une image de \mathcal{C}^k qui ne correspond pas à une projection exacte de l'image inductrice $A_k^{(0)}$. Pour obtenir une projection exacte sur une intersection non vide de convexes, Han propose une méthode de projections alternées où une composante normale au convexe courant est ajoutée avant de calculer chaque projection [40], [65]. Sur un modèle équivalent, nous proposons la méthode des projections alternées suivante afin de réaliser une projection exacte sur un ensemble \mathcal{C}^k consistant :

$$(N = 2^{k-1} \times 2^{k-1}) \quad \begin{cases} A_k^{(n+1)} = \text{Proj}_{\Lambda_{(i_n, j_n)}^k} (A_k^{(n)} + B_k^{(n)}) & \text{si } n \pmod{N+1} \neq N \\ A_k^{(n+1)} = \text{Proj}_{\Omega^k} (A_k^{(n)} + B_k^{(n)}) & \text{sinon} \end{cases} \quad (6.49)$$

$$i_n = n \pmod{N+1} \pmod{2^{k-1}} \quad j_n = \lfloor \frac{n \pmod{N+1}}{2^{k-1}} \rfloor \pmod{2^{k-1}}$$

La composante $B_k^{(n)}$ est un vecteur normal à l'ensemble $\Lambda_{(i_n, j_n)}^k$ lorsque $n \pmod{N+1} \neq N$ et normal à l'ensemble Ω^k dans les autres cas. Il correspond au vecteur de projection calculé au cycle précédent sur le même ensemble, mais de direction opposée. Durant le premier cycle, ce vecteur est nul. Ainsi, nous avons :

$$(N = 2^{k-1} \times 2^{k-1}) \quad \begin{cases} B_k^{(n)} = 0 & \text{pour } 0 \leq n \leq N \\ B_k^{(n)} = B_k^{(n-N-1)} + A_k^{(n-N-1)} - A_k^{(n-N)} & \text{pour } n > N \end{cases} \quad (6.50)$$

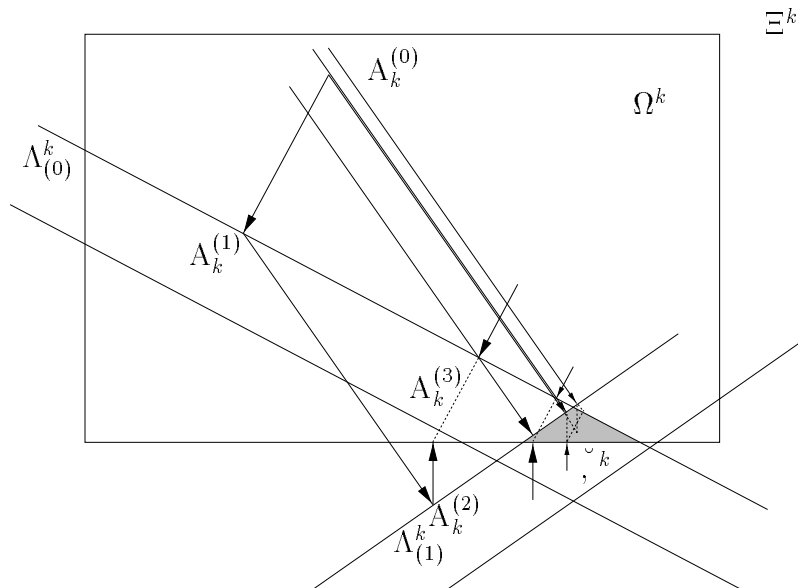


Figure 6.11 – Projection exacte sur un ensemble consistant : méthode des projections partielles alternées.

Le comportement de la séquence $(A_k^{(n)})$ est illustré par un exemple sur la figure 6.11. On remarquera notamment la propriété suivante [65] :

$$n \pmod{N+1} = 0 \implies A_k^{(n)} + \sum_{i=n}^{n+N} B_k^{(i)} = A_k^{(0)} \quad (6.51)$$

La suite de composantes $(B_k^{(n(N+1)+i)})_{i \in \{0, \dots, N\}}$ tend, avec l'accroissement de n , vers le vecteur de projection de $A_k^{(0)}$ sur \checkmark^k , c'est-à-dire vers le vecteur $\text{Proj}_{\Gamma^k}(A_k^{(0)}) - A_k^{(0)}$.

Proposition 6.9 (Convergence [65]) *La méthode des projections alternées définie par (6.49) et (6.50) produit une séquence $(A_k^{(n)})_{n \geq 0}$ qui converge vers $\text{Proj}_{\checkmark^k}(A_k^{(0)})$, lorsque l'ensemble \checkmark^k est consistant.*

6.3.7 Projection exacte par des projections partielles parallèles

La méthode de Han peut-être appliquée dans l'espace produit $\Xi^{k-1} \times \Xi^k$ en considérant les deux convexes $\Lambda^{k-1} \times \Omega^k$ et \mathcal{D} (cf. section 6.3.4). L'opération équivalente dans l'espace original Ξ^k est une suite de projections parallèles qui génère une séquence $(A_k^{(n)})$ convergeant vers la projection exacte de $A_k^{(0)}$ sur \checkmark^k , [40], [71].

Ainsi, la méthode des projections alternées, définie par (6.49) et (6.50), s'écrit dans l'espace $\Xi^{k-1} \times \Xi^k$ de la manière suivante :

$$\mathbf{A}^{(n+1)} = \text{Proj}_{\mathcal{D}} \circ \text{Proj}_{\Lambda^{k-1} \times \Omega^k} (\mathbf{A}^{(n)} + \mathbf{B}^{(n)}) \quad (6.52)$$

$$\begin{cases} \mathbf{B}^{(0)} = \mathbf{0} \\ \mathbf{B}^{(n+1)} = \mathbf{B}^{(n)} + \mathbf{A}^{(n)} - \text{Proj}_{\Lambda^{k-1} \times \Omega^k} (\mathbf{A}^{(n)} + \mathbf{B}^{(n)}) \text{ pour } n \geq 0 \end{cases} \quad (6.53)$$

Proposition 6.10 (Convergence) *La méthode des projections alternées définie par (6.52) et (6.53) produit une séquence $(\mathbf{A}^{(n)})_{n \geq 0}$ qui converge vers $\text{Proj}_{\checkmark^k}(\mathbf{A}^{(0)})$, lorsque l'ensemble \checkmark^k est consistant.*

En utilisant (6.38), on peut exprimer (6.52) et (6.53) dans Ξ^k par :

$$A_k^{(n+1)} = \frac{1}{2^{2(k-1)} + 1} \left[\sum_{(i,j) \in \mathcal{I}_{k-1}^2} \text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)} + B_{k,(i,j)}^{(n)}) + \text{Proj}_{\Omega^k} (A_k^{(n)} + O_k^{(n)}) \right] \quad (6.54)$$

$$(\forall (i,j) \in \mathcal{I}_{k-1}^2) \begin{cases} B_{k,(i,j)}^{(0)} = 0 \\ B_{k,(i,j)}^{(n+1)} = A_k^{(n)} + B_{k,(i,j)}^{(n)} - \text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)} + B_{k,(i,j)}^{(n)}) \text{ pour } n \geq 0 \end{cases} \quad (6.55)$$

$$\begin{cases} O_k^{(0)} = 0 \\ O_k^{(n+1)} = A_k^{(n)} + O_k^{(n)} - \text{Proj}_{\Omega^k} (A_k^{(n)} + O_k^{(n)}) \text{ pour } n \geq 0 \end{cases} \quad (6.56)$$

Proposition 6.11 (Convergence) *La méthode des projections parallèles définie par (6.54), (6.55) et (6.56) produit une séquence $(A_k^{(n)})_{n \geq 0}$ qui converge vers $\text{Proj}_{\checkmark^k}(A_k^{(0)})$, lorsque l'ensemble \checkmark^k est consistant.*

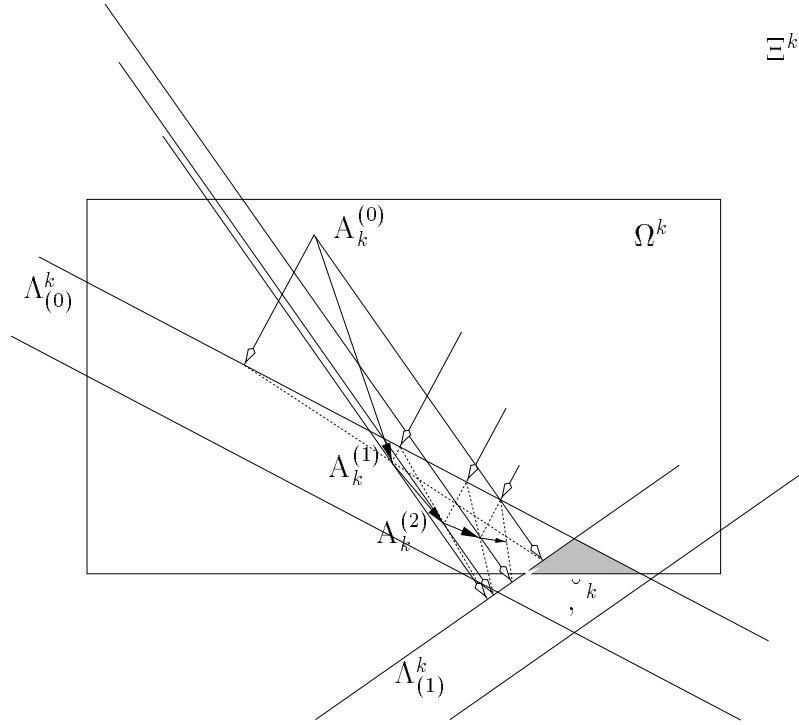


Figure 6.12 – Projection exacte sur un ensemble consistant : méthode des projections parallèles. Les vecteurs à flèches blanches représentent les vecteurs de projection partielle dont la combinaison génère les vecteurs à flèches noires représentant la séquence $A_k^{(n)}$.

Pour augmenter la vitesse de convergence de la séquence générée par (6.54), on peut utiliser des améliorations analogues à celles développées dans la section 6.3.4. Ainsi, on peut remplacer (6.54) par :

$$\begin{aligned}
 A_k^{(n+1)} &= A_k^{(n)} + \frac{1}{\sigma^{(n)}} \left(\text{Proj}_{\Omega^k} (A_k^{(n)} + O_k^{(n)}) - A_k^{(n)} \right) \\
 &+ \sum_{(i,j) \in \mathcal{I}_{k-1}^2} \frac{1}{\sigma_{(i,j)}^{(n)}} \left(\text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)} + B_{k,(i,j)}^{(n)}) - A_k^{(n)} \right) \quad (6.57)
 \end{aligned}$$

$$\begin{cases}
 \sigma^{(n)} = \max_{(i,j) \in \mathcal{I}_{k-1}^2} \mathcal{R}_{k-1}^{max(n)}(i, j) + 1 \\
 \sigma_{(i,j)}^{(n)} = \mathcal{R}_{k-1}^{max(n)}(i, j) \text{ si } \text{Proj}_{\Omega^k} (A_k^{(n)} + O_k^{(n)}) - A_k^{(n)} = 0 \\
 \sigma_{(i,j)}^{(n)} = \mathcal{R}_{k-1}^{max(n)}(i, j) + 1 \text{ sinon}
 \end{cases} \quad (6.58)$$

Nous obtenons ainsi une méthode de projections parallèles qui est illustrée par la figure 6.12. Les algorithmes de construction de $\mathcal{R}_k^{(n)}$ et $\mathcal{R}_{k-1}^{max(n)}$ sont donnés respectivement dans les tables 6.4 et 6.5.

$$\begin{aligned}
& \mathcal{R}_k^{(n)} = \mathbf{0} \\
& \text{pour } j \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\
& \quad \text{pour } i \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\
& \quad \quad \text{si } \text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)} + B_{k,(i,j)}^{(n)}) - A_k^{(n)} \neq \mathbf{0} \\
& \quad \quad \quad \text{pour } t \text{ allant de } \varphi(2j - M, 2^k) \text{ à } \varphi(2j + M, 2^k) \\
& \quad \quad \quad \quad \text{pour } s \text{ allant de } \varphi(2i - M, 2^k) \text{ à } \varphi(2i + M, 2^k) \\
& \quad \quad \quad \quad \quad \mathcal{R}_k^{(n)}(s, t) = \mathcal{R}_k^{(n)}(s, t) + 1
\end{aligned}$$

Table 6.4 – Construction de la matrice de recouvrements $\mathcal{R}_k^{(n)}$.

$$\begin{aligned}
& \mathcal{R}_{k-1}^{max(n)} = \mathbf{0} \\
& \text{pour } j \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\
& \quad \text{pour } i \text{ allant de } 0 \text{ à } 2^{k-1} - 1 \\
& \quad \quad \text{si } \text{Proj}_{\Lambda_{(i,j)}^k} (A_k^{(n)} + B_{k,(i,j)}^{(n)}) - A_k^{(n)} \neq \mathbf{0} \\
& \quad \quad \quad \text{pour } t \text{ allant de } \varphi(2j - M, 2^k) \text{ à } \varphi(2j + M, 2^k) \\
& \quad \quad \quad \quad \text{pour } s \text{ allant de } \varphi(2i - M, 2^k) \text{ à } \varphi(2i + M, 2^k) \\
& \quad \quad \quad \quad \quad \text{si } \mathcal{R}_k^{(n)}(s, t) > \mathcal{R}_{k-1}^{max(n)}(i, j) \\
& \quad \quad \quad \quad \quad \quad \mathcal{R}_{k-1}^{max(n)}(i, j) = \mathcal{R}_k^{(n)}(s, t)
\end{aligned}$$

Table 6.5 – Construction de la matrice de recouvrement maximal $\mathcal{R}_{k-1}^{max(n)}$.

6.4 Etude expérimentale des méthodes de projections partielles

Dans cette section, nous étudions expérimentalement les méthodes de projections partielles que nous avons présentées dans la section précédente. Cette étude est composée de trois parties. Dans la première, l'ensemble $\overline{\cdot}^k$ est consistant et dans la deuxième, il est inconsistant. La troisième partie correspond au cas général où l'on n'a aucune connaissance *a priori* sur la consistance de $\overline{\cdot}^k$. Nous résumons les principaux résultats de cette étude dans un tableau en section 6.4.3.

Pour simplifier la présentation des résultats, nous utilisons un symbole pour désigner chaque méthode de projections :

Symbole	Méthode
<i>MAPA</i>	projection approchée par projections alternées
<i>MAPP</i>	projection approchée par projections parallèles
<i>MAPPE</i>	projection approchée par projections parallèles extrapolées
<i>MEPA</i>	projection exacte par projections alternées
<i>MEPP</i>	projection exacte par projections parallèles

Pour chacune de ces méthodes, nous précisons les choix que nous avons faits concernant les paramètres. Dans la méthode *MAPA*, nous faisons une sur-relaxation progressive. Plus précisément, nous utilisons la règle suivante :

$$\lambda^{(0)} = 1 \quad , \quad \lambda^{(n+1)} = \begin{cases} \lambda^{(n)} + 0.1 & \text{si } \lambda^{(n)} < 1.9 \\ 1.99 & \text{sinon} \end{cases} \quad (6.59)$$

Dans la méthode *MAPP*, nous faisons une sur-relaxation maximale, c'est-à-dire que pour tout n nous adoptons $\lambda^{(n)} = 1.99$. Nous utilisons ici la version améliorée donnée par (6.42). Pour la méthode *MAPPE*, nous utilisons la version non-relaxée avec $\lambda^{(n)} = \beta^{(n)}$. Toutefois, nous adoptons la règle du recentrage donnée par (6.48). Pour la méthode *MEPP*, nous utilisons la version améliorée décrite par (6.57).

Ces différents choix correspondent à une utilisation presque optimale des méthodes de projections partielles. Enfin, précisons que nous adoptons comme filtre passe-bas le filtre de Burt de taille 5×5 avec $\beta = 0.35$ (cf. section 4.3.2), sauf indication contraire.

6.4.1 Induction sur un ensemble consistant

Pour évaluer les méthodes de projections partielles développées dans la section 6.3 dans le cadre d'une induction sur un ensemble consistant, nous utilisons une image initiale I_{k-1} de 64×64 pixels ($k = 7$). Pour être certain que $\overline{I_{k-1}}$ soit consistant, l'image initiale I_{k-1} est générée par une réduction d'une image de 128×128 pixels extraite de l'image de Lenna de 512×512 pixels. L'image de 128×128 pixels et sa version réduite I_{k-1} sont montrées sur la figure 6.13.

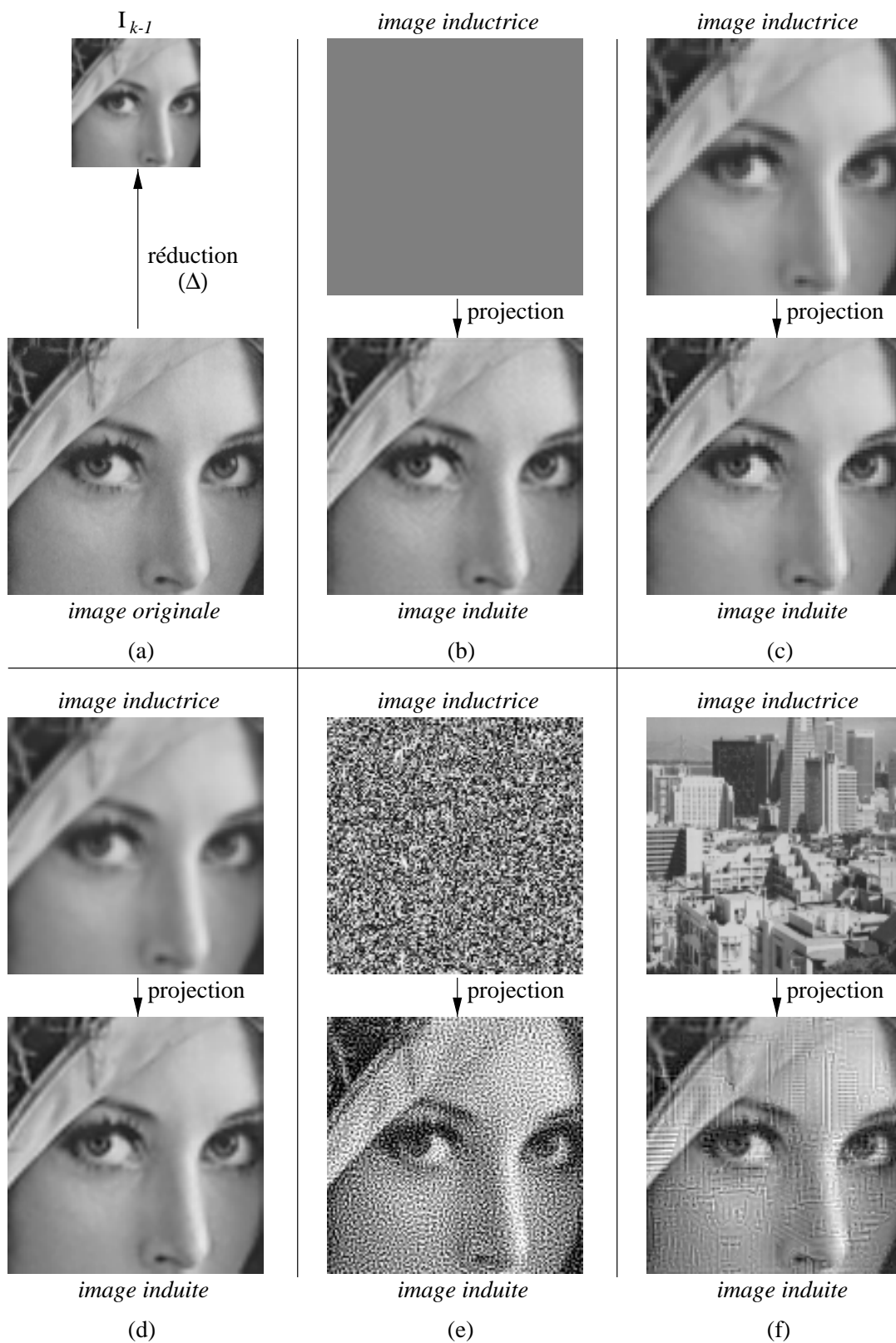


Figure 6.13 – Exemples d’inductions sur un ensemble consistant. (a) L’image initiale I_{k-1} de 64×64 pixels est obtenue par réduction d’une image originale de 128×128 pixels. Les couples d’images suivants (b-c-d-e-f) montrent à chaque fois une image inductrice particulière et l’image induite correspondante.

Influence de l'image inductrice sur l'image induite

Pour commencer, nous montrons sur la figure 6.13 une induction de I_{k-1} en utilisant diverses images inductrices :

Exemple	Image inductrice
(a)	image grise
(b)	interpolation au plus proche voisin de I_{k-1}
(c)	interpolation linéaire de I_{k-1}
(d)	image obtenue par un générateur aléatoire
(e)	image de San-Francisco

Les images induites correspondantes ont été obtenues par une projection exacte sur $\bar{\cdot}^k$ avec la méthode *MEPA*. On remarque notamment que l'induction permet de récupérer beaucoup d'informations (*a priori* perdues) sur l'image originale dont est issue I_{k-1} . Ceci est très clair avec l'image inductrice dans l'exemple (b). On voit également que les effets de pixelisation et de flou (respectivement) des images inductrices dans les exemples (c) et (d) ne sont pas supprimés après une projection sur $\bar{\cdot}^k$ mais sont tout de même atténués. Les exemples (e-f) illustrent l'influence que peut avoir l'image inductrice sur l'image induite : on remarque notamment que les hautes fréquences (contours) de l'image inductrice se retrouvent dans l'image induite.

Convergence des méthodes de projections partielles

Pour évaluer la vitesse de convergence des méthodes de projections partielles, nous employons la fonction de proximité Φ définie par (6.12). Ainsi, pour chaque méthode, nous calculons $\Phi(A_k^{(n)})$ à chaque itération globale⁵ n . Afin d'avoir une certaine «indépendance» vis-à-vis de l'image inductrice, nous avons normalisé les résultats avec $\Phi(A_k^{(0)})$. Les courbes correspondantes se trouvent sur la figure 6.14. Nous avons utilisé une image noire comme image inductrice.

Tout d'abord, nous remarquons que les méthodes *MAPA*, *MAPPE* et *MEPA* convergent vers une solution de $\bar{\cdot}^k$ en un nombre fini d'itérations (environ une trentaine). Les méthodes *MAPP* et surtout *MEPP* ont quant à elles une convergence beaucoup plus lente et semblent ne jamais atteindre une solution de $\bar{\cdot}^k$ (observation sur 1000 itérations). Ce premier constat est en accord avec la théorie, puisque les méthodes *MAPA*, *MEPA* et surtout *MAPPE* exploitent le fait que $\bar{\cdot}^k$ est consistant.

Ensuite, on remarque que *MAPPE* est la méthode qui converge le plus rapidement dans les dix premières itérations. On observe dans les itérations suivantes des pics d'oscillations. En fait, l'extrapolation est bien moins efficace lorsque $A_k^{(n)}$ est proche de $\bar{\cdot}^k$. Le recentrage donné par (6.48) a permis de limiter la plage d'oscillations. La

5. Pour les méthodes de projections alternées, une itération globale correspond à un cycle complet de projections locales sur toute l'image.

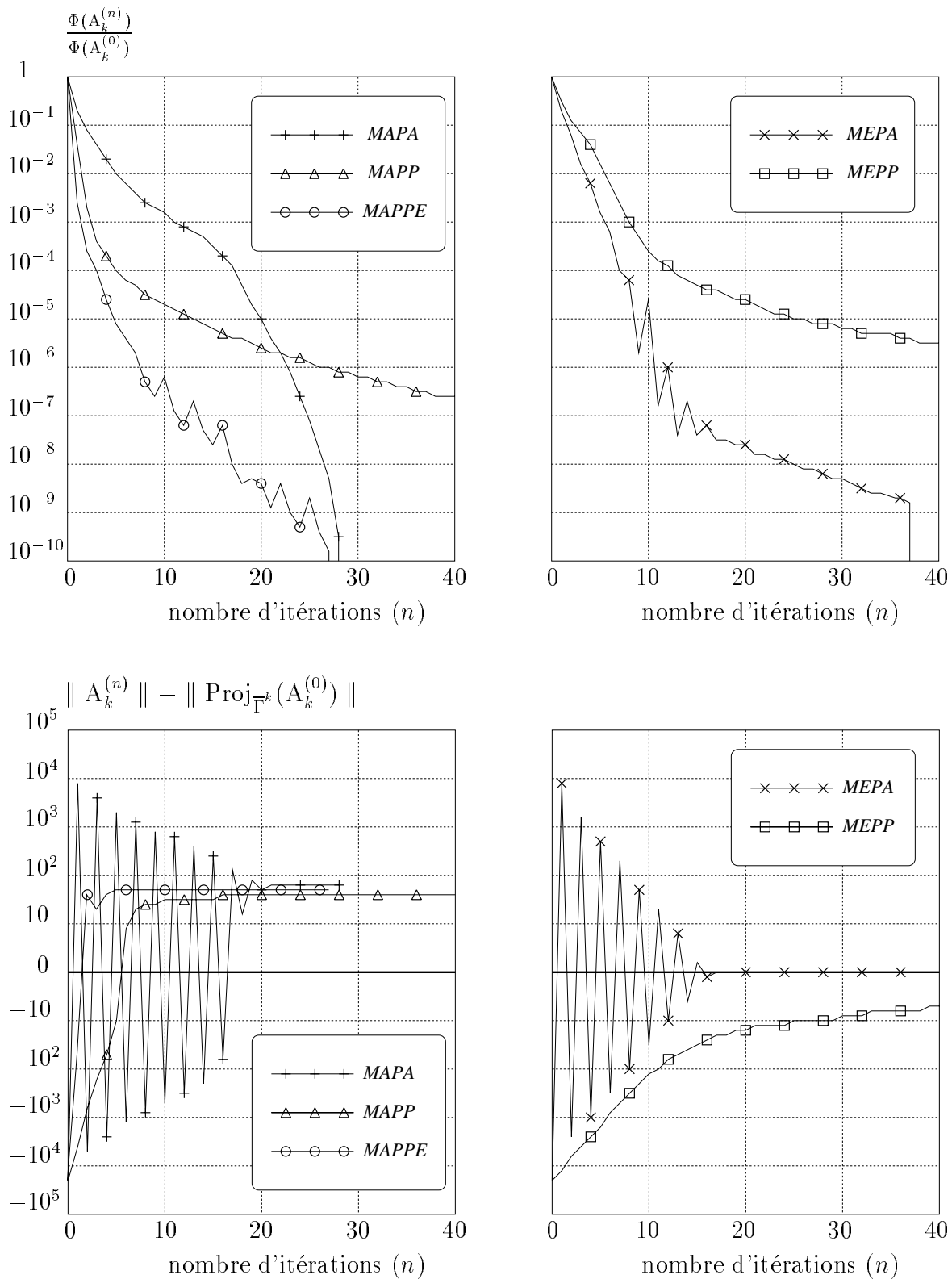


Figure 6.14 – Ensemble consistant : convergence des méthodes de projections partielles (haut) et mesure de la distance normalisée à l'image inductrice (bas).

méthode *MAPP* a également un très bon taux de convergence dans les premières itérations, mais celui-ci faiblit très rapidement. Ceci est dû à l'utilisation de la sur-relaxation maximale dont les effets sont d'autant plus visibles que $A_k^{(n)}$ est éloigné de $\bar{\cdot}^k$.

Nous remarquons également que la méthode *MEPA* semble converger plus vite que *MAPA* dans les dix premières itérations. Dans la deuxième dizaine d'itérations, *MEPA* souffre d'oscillations alors que *MAPA* montre un point d'inflexion. Dans la troisième dizaine d'itérations, la courbe de *MEPA* se stabilise avec un taux de convergence très moyen alors que le taux de convergence de *MAPA* augmente très rapidement. Le comportement de *MAPA* est en fait dû à la sur-relaxation progressive. Celui de *MEPA* peut s'expliquer par l'effet de recentrage vis-à-vis de $A_k^{(0)}$ qui est inhérent à cette méthode.

Enfin, nous remarquons que la méthode *MEPP* est celle qui offre le taux de convergence le plus faible, que ce soit dans les premières itérations ou dans les suivantes. Ceci est certainement dû à l'absence de relaxation dans la méthode.

Distance à l'image inductrice

Nous avons également mesuré à chaque itération globale n , la distance séparant l'image courante $A_k^{(n)}$ de l'image inductrice $A_k^{(0)}$. Cette dernière étant une image noire, sa distance à l'image $A_k^{(n)}$ revient à calculer la norme de $A_k^{(n)}$. D'après la théorie, les méthodes de projection exacte *MEPA* et *MEPP* minimisent cette norme lorsque n tend vers l'infini, celle-ci est alors équivalente à $\| \text{Proj}_{\bar{\Gamma}^k}(A_k^{(0)}) \|$. Nous avons pris l'image induite par *MEPA* comme estimation de $\text{Proj}_{\bar{\Gamma}^k}(A_k^{(0)})$. Aussi, pour simplifier la lecture des courbes, $\| \text{Proj}_{\bar{\Gamma}^k}(A_k^{(0)}) \|$ est soustrait de $\| A_k^{(n)} \|$. En d'autres termes, la méthode *MEPA* sert de méthode de référence. Les résultats sont présentés sur la figure 6.14.

Tout d'abord, nous remarquons que les courbes associées aux méthodes de projections alternées *MAPA* et *MEPA* marquent de fortes oscillations dans les premières itérations. Leurs amplitudes semblent diminuer très rapidement pour finalement devenir négligeables (à notre échelle d'étude) aux alentours d'une vingtaine d'itérations. Les courbes associées aux méthodes de projections parallèles *MAPP*, *MAPPE* et *MEPP* ne sont pas affectées par des oscillations. Au contraire, l'éloignement de l'image inductrice est ici très progressif.

Ensuite, nous remarquons que les images induites obtenues par les méthodes de projection approchée *MAPA*, *MAPP* et *MAPPE* ont toutes une norme supérieure à celle de l'image induite obtenue par *MEPA*. Ceci est en accord avec la théorie puisque *MEPA* est censée produire l'image induite la plus proche de l'image inductrice. Parmi les méthodes de projection approchée, c'est *MAPP* qui possède la norme la plus faible, puis viennent *MAPPE* et *MAPA*.

Nous observons également que la méthode de projection exacte *MEPP* génère une séquence qui semble converger en norme vers l'image induite par *MEPA*. Ce fait est en accord avec la théorie puisque la projection de $A_k^{(0)}$ sur $\bar{\cdot}^k$ est unique.

Qualité des méthodes de projection approchée

Pour évaluer la qualité des méthodes de projection approchée *MAPA*, *MAPP* et *MAPPE*, nous calculons la distance séparant les images induites obtenues par ces méthodes de l'image induite par *MEPA*. Nous obtenons les résultats suivants :

Méthode	$d(A_k^{(\infty)}, \text{Proj}_{\overline{T}^k}(A_k^{(0)}))$
<i>MAPA</i>	339.29
<i>MAPP</i>	187.89
<i>MAPPE</i>	188.77

D'après ce tableau, les méthodes *MAPP* et *MAPPE* produisent des images induites de qualité comparable. Il semble donc que les extrapolations faites par *MAPPE* n'aient pas altéré la qualité de l'image induite. Nous remarquons également que les méthodes parallèles *MAPP* et *MAPPE* génèrent une image induite de meilleure qualité que celle générée par la méthode des projections alternées *MAPA*. Ceci s'explique par le fait que les projections alternées prennent des directions plutôt «aveugles» vis-à-vis de l'ensemble $\overline{,}^k$, alors que les méthodes de projections parallèles s'orientent bien mieux vers $\overline{,}^k$.

La norme de l'image induite avec *MAPPE* étant supérieure à celle de l'image induite avec *MAPP* (cf. figure 6.14), on en déduit que l'image induite par *MAPPE* est plutôt à l'intérieur de $\overline{,}^k$ que sur sa surface. On explique ceci par l'extrapolation de *MAPPE* et par le fait que *MAPP* génère une séquence qui tend vers la surface de $\overline{,}^k$ sans la traverser, même avec une sur-relaxation.

Influence du filtre passe-bas sur l'induction

Pour achever nos expérimentations dans le contexte d'un ensemble $\overline{,}^k$ consistant, nous nous intéressons à l'influence du filtre passe-bas sur le résultat de l'induction. Par une simple variation du paramètre β du filtre de Burt, nous pouvons changer le graphe du filtre passe-bas. Sur la figure 6.15, nous montrons les images induites avec la méthode de projection exacte *MEPA* à partir d'une image inductrice grise (cf. exemple (b) sur la figure 6.13), ceci en faisant varier le paramètre β dans l'intervalle $[0.35, 0.6]$. Pour faciliter l'interprétation, nous avons dessiné le graphe du filtre passe-bas pour chaque valeur de β . Par exemple : pour $\beta = 0.40$, le filtre est unimodal et proche d'une gaussienne, pour $\beta = 0.50$, le filtre est triangulaire et pour $\beta = 0.60$, le filtre est trimodal et plus proche du filtre passe-bas idéal sinus cardinal.

Nous observons que plus la valeur de β est importante, plus l'image induite présente un effet de crénelage. Nous remarquons également que la convergence de l'induction est d'autant plus rapide que la valeur de β est grande : 35 itérations pour $\beta = 0.35$, 15 it. pour $\beta = 0.40$, 9 it. pour $\beta = 0.45$, 7 it. pour $\beta = 0.50$, 5 it. pour $\beta = 0.55$ et 2 it. pour $\beta = 0.60$. Ceci peut s'expliquer par le fait qu'un filtre passe-bas unimodal et aplati ($\beta = 0.35$) génère des contraintes locales dans l'induction beaucoup plus difficiles à satisfaire que lorsque le filtre est trimodal et étiré

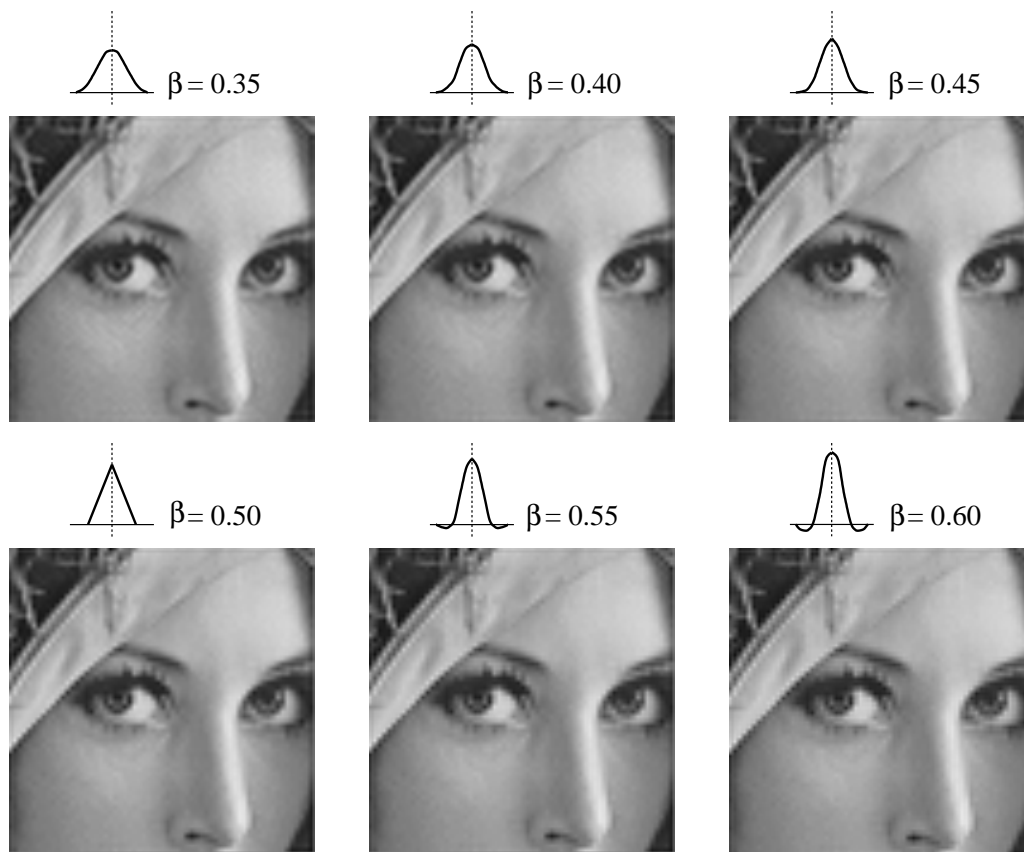


Figure 6.15 – Ensemble consistant : influence du filtre passe-bas sur l'induction. Nous montrons ici les images induites à partir d'une image grise en faisant varier le paramètre β du filtre de Burt.

($\beta = 0.60$). D'un point de vue géométrique, les angles entre les ensembles de la famille $(\Lambda_{(i,j)}^k)$ augmentent avec la valeur de β , ce qui a pour effet de diminuer le nombre des projections élémentaires nécessaires pour atteindre l'ensemble $\overline{\Lambda}^k$.

6.4.2 Induction sur un ensemble inconsistant

Pour évaluer les méthodes de projections partielles dans le cadre d'une induction sur un ensemble inconsistant, nous utilisons une image initiale I_{k-1} de type échiquier noir&blanc de taille 64×64 pixels ($k = 7$). Le filtre de Burt ($\beta=0.35$) que nous utilisons garantit dans ce cas l'inconsistance de $\overline{\Lambda}^k$: un tel filtre engendre nécessairement un lissage dans l'opération de réduction. L'image initiale I_{k-1} est montrée dans le coin supérieur gauche de la figure 6.16. Précisons qu'il s'agit ici d'un cas extrême d'inconsistance qui a pour intérêt de faciliter l'analyse expérimentale des caractéristiques de chaque méthode.

Qualité des méthodes de projections partielles

Nous montrons sur la figure 6.16 l'induction de I_{k-1} sur $\bar{\cdot}^k$ en utilisant l'image de San-Francisco (128×128 pixels) comme image inductrice. Nous avons testé les cinq méthodes de projections partielles. Les images induites que nous montrons correspondent à une centaine d'itérations globales.

Les méthodes de projections parallèles *MAPP* et *MEPP* produisent des images induites visuellement identiques dont la réduction donne une version «lissée» de I_{k-1} . Pour ces deux images, nous remarquons également que les pixels de coordonnées paires sont soit noirs, soit blancs et que les pixels intermédiaires ont un niveau de gris plus ou moins proche du gris moyen. C'est d'ailleurs sur l'ensemble de ces pixels intermédiaires que se retrouvent certains contours de l'image inductrice.

La méthode *MAPA* produit une séquence d'images instable (cf. figure 6.4(a)) et l'image réduite présente des artefacts très visibles. La méthode *MEPA* produit une image étonnamment proche de celles produites par les méthodes de projections parallèles *MAPP* et *MEPP*. Des différences sont tout de même perceptibles sur les bords de l'image. Enfin, la méthode *MAPPE* produit une séquence chaotique qui semble converger vers différents attracteurs, comme le montre la figure 6.17.

Convergence des méthodes de projections partielles

Pour évaluer la convergence des méthodes de projection dans le cas inconsistant, nous avons utilisé la fonction Φ de la même manière que dans le cas consistant (cf. section 6.4.1). Les méthodes de projections parallèles devant théoriquement minimiser la fonction Φ , nous avons pris l'image induite par la méthode *MEPP* comme référence ($\text{Proj}_{\bar{\cdot}^k}(A_k^{(0)})$). Les courbes se trouvent sur la figure 6.18(a). La courbe associée à *MAPPE* étant trop chaotique, nous avons jugé qu'il était inutile de la présenter. L'image inductrice est une image noire.

Nous remarquons notamment que les méthodes de projections alternées *MAPA* et *MEPA* convergent rapidement dans les premières itérations pour se stabiliser presque aussitôt. Les méthodes de projections parallèles *MAPP* et *MEPP* convergent moins vite mais minimisent davantage la fonction Φ , conformément à la théorie. Nous remarquons également que la méthode *MEPP* a un taux de convergence un peu plus faible que celui de *MAPP*. Cette observation est la même que dans le cas consistant.

Distance à l'image inductrice

De la même manière que dans le cas consistant, nous avons mesuré la distance entre l'image induite et l'image inductrice en prenant comme référence l'image induite par la méthode d'induction exacte *MEPP*. Les résultats se trouvent sur la figure 6.18(b).

Sur cette figure, nous observons tout d'abord que les méthodes de projections alternées *MAPA* et *MEPA* produisent des images induites plus proches de l'image inductrice. Toutefois, comme l'illustre la figure 6.18(a), ces deux méthodes ne minimisent pas la fonction Φ . Notons que la séquence produite par *MEPA* se stabilise

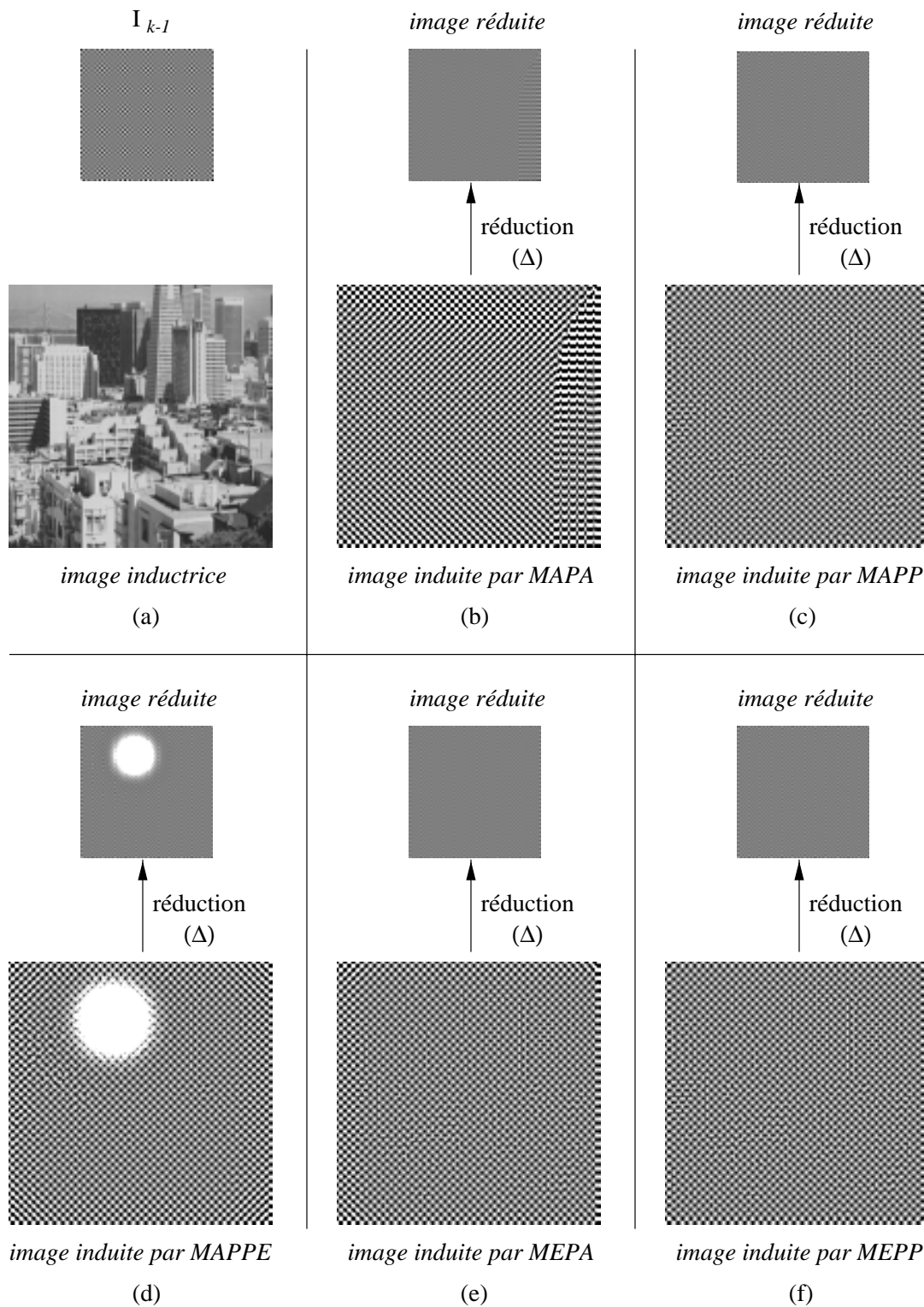


Figure 6.16 – Exemples d’inductions sur un ensemble inconsistant. (a) L’image initiale de 64×64 pixels est une image de type échiquier noir&blanc et l’image inductrice est une image de San-Francisco. Les couples d’images suivants (b-c-d-e-f) montrent à chaque fois l’image induite et l’image réduite associée obtenues avec les différentes méthodes de projections partielles.

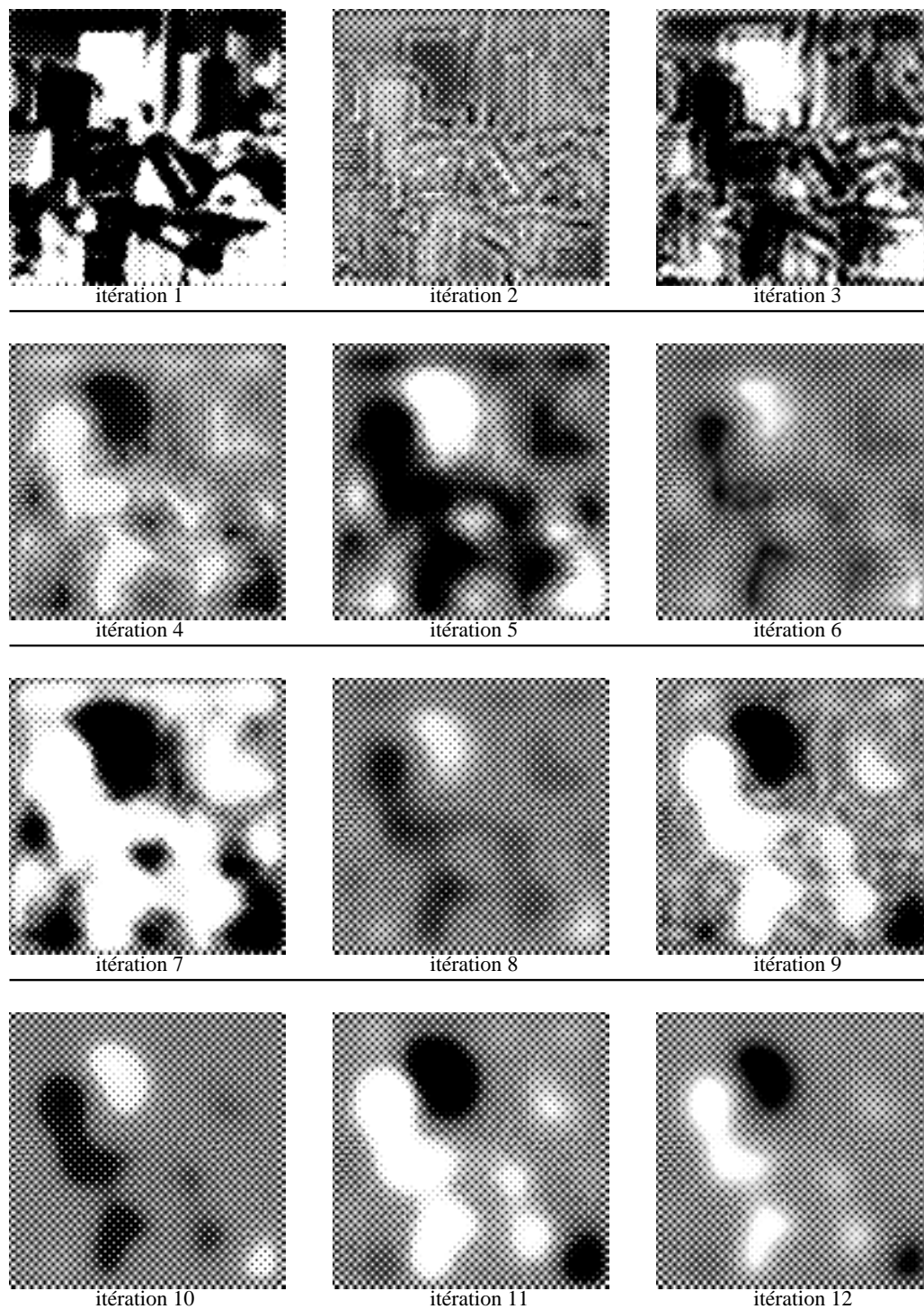


Figure 6.17 – Séquence d'images générée par projection approchée sur un ensemble inconsistant avec la méthode des projections parallèles extrapolée (*MAPPE*).

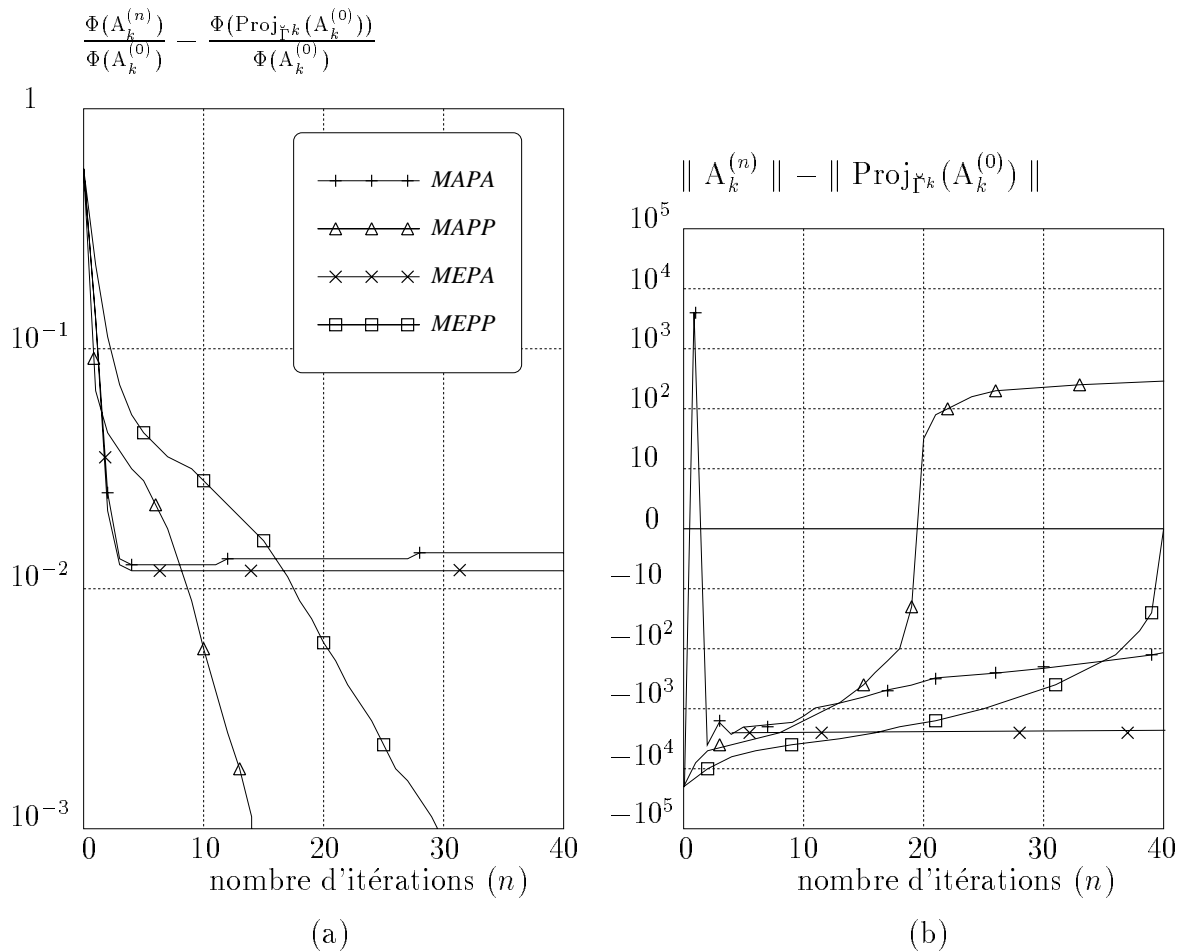


Figure 6.18 – Ensemble inconsistant : (a) convergence des méthodes de projections partielles, (b) distance normalisée à l’image inductrice.

alors que celle produite par *MAPA* semble s’éloigner lentement de l’image inductrice.

Enfin, conformément à la théorie, nous observons que l’image induite par *MEPP* est toujours plus proche de l’image inductrice que celle produite par *MAPP*.

6.4.3 Induction sans connaissance *a priori*

Lorsque l’on n’a aucune connaissance *a priori* sur la consistance de $\bar{\Gamma}^k$, on suppose que celui-ci est inconsistant. L’induction se fait alors avec les méthodes de projections parallèles *MAPP* ou *MEPP*. Pour réaliser l’induction dans ce contexte, nous avons utilisé l’image de 64×64 pixels extraite de l’image de Lenna de 512×512 déjà employée dans la partie expérimentale du chapitre 5.

Comme sur la figure 6.15, nous avons réalisé plusieurs inductions en faisant varier le paramètre β du filtre de Burt. Sur la figure 6.19, nous montrons les images induites avec la méthode de projection exacte *MEPP* à partir d’une image inductrice grise (cf. exemple (b) sur la figure 6.13). Lorsque $\beta = 0.60$, nous observons un léger effet de flou. Celui-ci semble s’atténuer avec la décroissance de β , mais il est remplacé

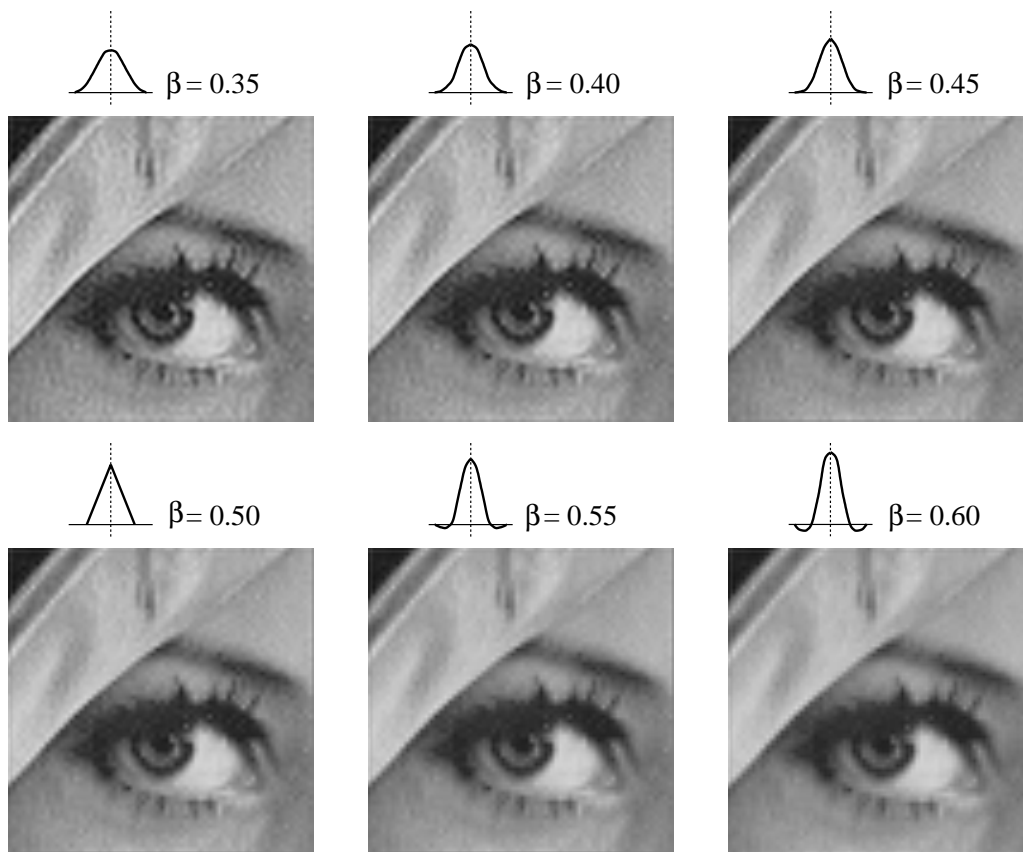


Figure 6.19 – Induction de l'image de Lenna sur Γ^k avec la méthode MEPP. Nous montrons ici les images induites en faisant varier le paramètre β du filtre de Burt. L'image inductrice est une image grise.

par un effet granuleux particulièrement visible pour $\beta = 0.35$.

Lorsque $\beta = 0.6$, le filtre de Burt est assez proche des caractéristiques du filtre passe-bas idéal. Nous avons montré dans la section 6.2.5 que lorsque le filtre W est le filtre passe-bas idéal, les images appartenant à $\overline{\Gamma}^k$ ont approximativement le même contenu fréquentiel que l'image initiale I_{k-1} dans les basses fréquences. Les hautes fréquences ne sont pas soumises à des contraintes : elles sont liées à celles de l'image inductrice. Lorsque celle-ci est une image grise, les hautes fréquences sont nulles. Ainsi, avec $\beta = 0.6$ et une image inductrice grise, on peut considérer que l'image induite illustrée sur la figure 6.19 possède sensiblement le même contenu fréquentiel que l'image initiale de Lenna. Le résultat est en fait assez comparable à celui obtenu avec la méthode d'interpolation cubique présentée dans la section 2.3.2.

6.4.4 Résumé de l'étude

Pour résumer l'étude des méthodes de projections partielles, nous avons rassemblé dans un tableau leurs principales propriétés, ainsi que leur classement respectif selon des critères de convergence et de qualité dans les cas consistant et inconsistant.

Méthode	<i>MAPA</i>	<i>MAPP</i>	<i>MAPPE</i>	<i>MEPA</i>	<i>MEPP</i>
Ensemble $\overline{\cdot}^k$	consistant	quelconque	consistant	consistant	quelconque
Induction	approchée	approchée	approchée	exacte	exacte
Particularité	relaxation	relaxation	extrapolation	-	-
<i>cas consistant</i>					
convergence	2	4	1	3	5
qualité	5	3	3	1	1
<i>cas inconsistant</i>					
convergence	4	1	-	3	2
qualité	4	2	-	3	1

Table 6.6 – Propriétés et classements des méthodes de projections partielles.

Lorsque $\overline{\cdot}^k$ est consistant, on peut utiliser toutes les méthodes de projections partielles : *MAPA*, *MAPP*, *MAPPE*, *MEPA* et *MEPP*. Deux de ces méthodes sont particulièrement efficaces : la méthode des projections parallèles extrapolée *MAPPE* et la méthode des projections alternées *MEPA*. La méthode *MAPPE* a l'avantage de converger très rapidement vers $\overline{\cdot}^k$ grâce à son principe d'extrapolation, tandis que la méthode *MEPA* a l'avantage de garantir une projection exacte sur $\overline{\cdot}^k$ avec un taux de convergence raisonnable.

Lorsque $\overline{\cdot}^k$ est inconsistant ou que l'on a pas de connaissance *a priori* sur la consistance, il faut utiliser les méthodes de projections parallèles *MAPP* ou *MEPP*. La méthode *MAPP* est plus rapide, mais la méthode *MEPP* a l'avantage sur celle-ci de garantir une projection exacte.

6.5 Induction avec un facteur d'agrandissement supérieur à deux

Dans ce chapitre, nous nous sommes intéressés pour l'instant à un agrandissement de facteur 2 de l'image I_{k-1} . Pour produire un agrandissement de facteur $a > 2$, il faut réaliser une induction de I_{k-1} sur l'ensemble des images de résolution $k + \log_2(a/2)$ dont la réduction de facteur a fournit la meilleure approximation de I_{k-1} . Cet ensemble est défini par :

$$\overline{\cdot}^{k+b} = \left\{ X_{k+b} \mid \Delta^{\circ(b+1)}(X_{k+b}) \approx I_{k-1} \right\} \quad (6.60)$$

où $b = \log_2(a/2)$.

Comme l'illustre la figure 6.20 dans le cas consistant avec $b = 1$, on ne peut pas se contenter de réaliser l'induction de I_{k-1} sur l'ensemble $\overline{\cdot}^{k+b}$ en effectuant successivement une induction de I_{k-1} sur $\overline{\cdot}^k$, puis une induction de A_k sur $\overline{\cdot}^{k+1}$ en poursuivant

jusqu'à faire une induction de A_{k+b-1} sur $,^{k+b}$. Ce découpage de l'induction en plusieurs étapes où le facteur d'agrandissement est 2 permet d'obtenir une solution de $,^{k+b}$ en utilisant les développements réalisés dans les sections 6.2 et 6.3. Toutefois, il permet seulement d'obtenir un sous-ensemble très réduit de solutions dans $,^{k+b}$. En d'autres termes, cette procédure ne réalise pas une induction de I_{k-1} sur l'ensemble $,^{k+b}$ mais sur un sous-ensemble de celui-ci qui dépend des images induites A_k, \dots, A_{k+b-1} aux résolutions intermédiaires.

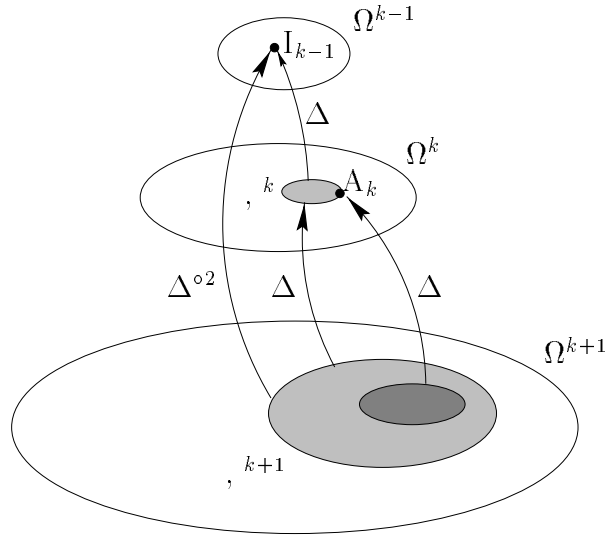


Figure 6.20 – Induction avec un facteur d'agrandissement supérieur à deux.

Pour réaliser l'induction de I_{k-1} directement sur l'ensemble $,^{k+b}$, il est nécessaire de traiter avec l'opérateur de réduction $\Delta^{\circ(b+1)}$ qui apparaît dans la définition de $,^{k+b}$. En reprenant littéralement la définition 6.1 de l'opérateur Δ , on obtient :

$$\Delta^{\circ(b+1)}(X_{k+b}) = \underbrace{[(W * \dots (W * (W * X_{k+b}) \downarrow 2) \downarrow 2 \dots) \downarrow 2]}_{b+1 \text{ fois}} = X_{k-1} \quad (6.61)$$

Les différentes opérations de convolution par W et de décimation de facteur 2 peuvent être réalisées de manière équivalente par une seule opération de convolution avec un filtre \mathbf{W} et une décimation de facteur a , c'est-à-dire :

$$\Delta^{\circ(b+1)}(X_{k+b}) = [\mathbf{W} * X_{k+b}] \downarrow a = X_{k-1} \quad (6.62)$$

Pour construire le filtre \mathbf{W} , il faut convoluer b fois le filtre W par des copies agrandies de lui-même. Cette opération s'écrit :

$$\mathbf{W} = W * W^{\uparrow 2} * W^{\uparrow 4} * \dots * W^{\uparrow a} \quad (6.63)$$

où la définition de la notation $W^{\uparrow a}$ est donnée par :

$$(0 \leq m, n < 2aM + 1) \quad \begin{cases} W^{\uparrow a}(m, n) = W\left(\frac{m}{a}, \frac{n}{a}\right) & \text{lorsque } \frac{m}{a} \text{ et } \frac{n}{a} \text{ sont entiers} \\ W^{\uparrow a}(m, n) = 0 & \text{sinon} \end{cases} \quad (6.64)$$

En d'autres termes, le filtre $W^{\uparrow a}$ est obtenu en intercalant $a - 1$ coefficients nuls entre les coefficients initiaux de W . La largeur de ce dernier étant $2M + 1$, on voit assez facilement à partir de l'expression (6.63) que la largeur du filtre \mathbf{W} est $2M(2a - 1) + 1$.

De la même manière que nous l'avons fait pour l'opérateur Δ dans la section 6.2.2, nous pouvons reformuler l'opérateur $\Delta^{\circ(b+1)}$ avec la notation matricielle suivante :

$$\Delta^{\circ(b+1)}(\vec{X}_{k+b}) = [\mathcal{H} \cdot \vec{X}_{k+b}] = \vec{X}_{k-1} \quad (6.65)$$

où la matrice de réduction \mathcal{H} est une matrice de taille $2^{2(k+b-1)} \times 2^{2k+b}$ définie par :

$$\mathcal{H} = \begin{bmatrix} {}^t\vec{H}_{k+b,(0,0)} \\ \vdots \\ {}^t\vec{H}_{k+b,(0,2^{k-1}-1)} \\ \vdots \\ {}^t\vec{H}_{k+b,(2^{k-1}-1,0)} \\ \vdots \\ {}^t\vec{H}_{k+b,(2^{k-1}-1,2^{k-1}-1)} \end{bmatrix} \quad (6.66)$$

où $(H_{k+b,(i,j)})_{(i,j) \in \mathcal{I}_{k-1}^2}$ est une famille de matrices de taille $2^{k+b} \times 2^{k+b}$ engendrée par le filtre \mathbf{W} . Plus précisément, pour tout (i, j) appartenant à \mathcal{I}_{k-1}^2 , une matrice $H_{k+b,(i,j)}$ positionne le centre du filtre en (ai, aj) . L'algorithme de génération d'une matrice de la famille $(H_{k+b,(i,j)})$ est donné dans la table 6.7.

$H_{k+b,(i,j)} = \mathbf{0}$ <p style="margin-left: 20px;">pour n allant de $-M(2a - 1)$ à $M(2a - 1)$</p> <p style="margin-left: 20px;">pour m allant de $-M(2a - 1)$ à $M(2a - 1)$</p> <p style="margin-left: 40px;">$s = \varphi(ai + m, 2^{k+b})$ et $t = \varphi(aj + n, 2^{k+b})$</p> $H_{k+b,(i,j)}(s, t) = H_{k+b,(i,j)}(s, t) + \mathbf{W}(m, n)$
--

Table 6.7 – Construction d'une matrice $H_{k+b,(i,j)}$.

Grâce à la formulation donnée par (6.65), on peut généraliser assez facilement tous les développements réalisés dans les sections 6.2 et 6.3. Un exemple d'induction avec un facteur d'agrandissement $a = 4$ est montré sur la figure 6.21. Il s'agit d'un cas

d'induction sur un ensemble consistant. L'image originale du cameraman (128×128 pixels) est d'abord réduite d'un facteur 4 par l'opérateur $\Delta^{\circ 2}$. Pour le filtre W , nous utilisons toujours le filtre de Burt 5×5 avec $\beta = 0.35$. Ensuite, l'induction de l'image réduite est réalisée avec la méthode *MEPA* en utilisant comme image inductrice l'image obtenue en interpolant l'image réduite par une transformée B-spline cubique. Il apparaît que l'image induite récupère un peu d'information spectrale sur l'image originale. En effet, l'ensemble des solutions étant convexe, l'image induite qui résulte de la projection est forcément plus proche de l'image originale que ne l'est l'image inductrice projetée. On peut également noter la présence d'un effet d'ondulation des contours dans l'image induite.

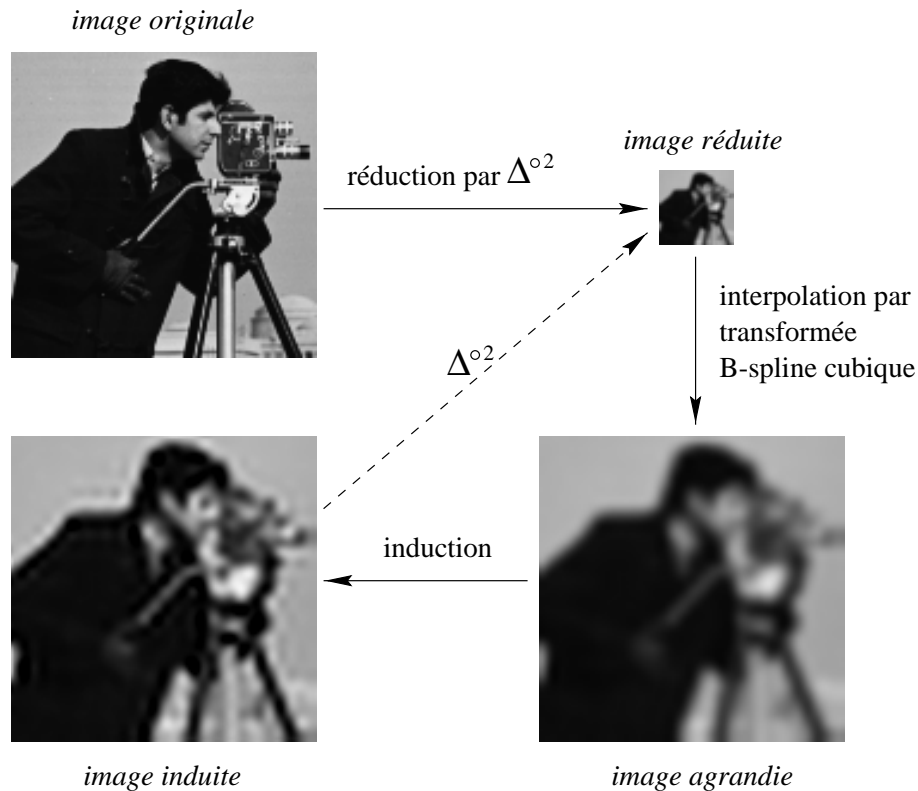


Figure 6.21 – Exemple d'induction avec un facteur d'agrandissement $a = 4$.

6.6 Conclusion

Nous avons présenté dans ce chapitre notre deuxième méthode d'agrandissement de type ensembliste. Le problème de l'agrandissement d'une image est ici posé comme un problème inverse de réduction.

Dans un premier temps, nous avons défini précisément l'ensemble des solutions du problème en nous appuyant sur l'opération de réduction telle qu'elle est définie par la théorie de l'échantillonnage. Nous nous sommes particulièrement attachés à

ce que l'ensemble des solutions soit convexe quels que soient l'image originale et le filtre utilisé dans l'opération de réduction.

Dans un deuxième temps, nous avons proposé différentes méthodes de projection sur cet ensemble, afin d'obtenir une solution particulière. Ces méthodes de projection ont été inspirées par des méthodes générales que l'on trouve dans la théorie d'estimation d'ensembles convexes [39]. Certaines permettent d'obtenir une projection exacte, et d'autres une projection approchée mais plus rapide. De plus, certaines ne peuvent être utilisées que lorsque l'ensemble des solutions est consistant. En dernier lieu, nous avons étudié le comportement expérimental de ces différentes méthodes. Lorsque l'ensemble des solutions est consistant, la méthode *MEPA* est la plus intéressante puisqu'elle permet d'obtenir une projection exacte. Dans un contexte où la rapidité de la projection est cruciale, la méthode *MAPPE* est plus adaptée puisqu'elle permet d'obtenir une projection approchée très rapidement. Lorsque l'ensemble des solutions est inconsistant, la méthode *MEPP* permet d'obtenir une projection exacte tandis que la méthode *MAPP* permet d'obtenir plus rapidement une projection approchée.

Cette méthode d'agrandissement par induction sur un ensemble est complémentaire des méthodes d'agrandissement du chapitre 3, ainsi que de notre première méthode présentée dans le chapitre 5. Elle permet de régulariser les images agrandies vis-à-vis de la contrainte de réduction. Ceci revient en particulier à restaurer l'information, relative à l'image originale, éventuellement perdue lors de son agrandissement. Cette méthode peut également être utilisée dans un contexte de compression réversible et de transmission progressive des images. Ces deux applications de l'induction seront étudiées dans le chapitre 7.

Chapitre 7

Améliorations et applications

7.1 Introduction

Ce chapitre est consacré aux améliorations et aux applications des deux méthodes d'agrandissement que nous avons proposées dans les chapitres 5 et 6.

Tout d'abord, nous proposons dans la section 7.2 d'améliorer la méthode d'agrandissement par synthèse de similarités sur une pyramide (cf. chapitre 5). La pyramide classique à facteur de réduction 2 est remplacée par une pyramide en quinconce à facteur de réduction $\sqrt{2}$. La suppression de l'information niveau par niveau étant moins brutale, on augmente la probabilité de trouver des similarités intéressantes sur la pyramide. La synthèse de meilleures similarités doit apporter une amélioration de la qualité de l'image agrandie.

D'un point de vue pratique, la méthode d'agrandissement par induction (cf. chapitre 6) permet de régulariser une image agrandie quelconque vis-à-vis de la contrainte de réduction. Pour cela, il suffit de la projeter sur l'ensemble, k des solutions. Cette régularisation permet en particulier de restaurer l'information associée à l'image originale qui est perdue lors de l'agrandissement. Nous discutons de cet aspect de l'induction dans la section 7.3. Nous donnons également quelques résultats qui mettent en évidence l'utilité d'une régularisation.

Pour terminer, nous proposons dans la section 7.4 d'employer la méthode d'agrandissement par induction pour compresser et transmettre progressivement une image sur réseau [23], [24], [25]. Le principe d'induction est utilisé dans un contexte pyramidal en modifiant l'algorithme de construction d'une pyramide laplacienne. Nous montrons en particulier que l'agrandissement par induction permet d'augmenter le taux de compression de manière significative. Il permet également au destinataire de donner aux niveaux successifs de la pyramide reçus par réseau la même résolution que l'image originale.

7.2 Synthèse de similarités sur une pyramide en quinconce

La table 5.10 montre clairement que les similarités ne sont pas réparties de manière équitable entre les différents niveaux de la pyramide. Les meilleures similarités des blocs de la base de la pyramide sont généralement trouvées sur les niveaux proches de cette base. En fait, plus on monte dans la pyramide et plus la probabilité de trouver une similarité diminue. Ce résultat montre que les similarités entre les structures ont en général peu de chance d'exister lorsque les résolutions respectives sont très différentes. Toutefois, ceci dépend beaucoup de la complexité des structures elles-mêmes. Pour illustrer ce propos, reprenons la pyramide générée à partir de l'image du cameraman qui est montrée sur la figure 5.16. Il apparaît clairement que le pourtour assez simple de la tête offre un remarquable degré d'auto-similarité, au moins jusqu'au niveau 2 de la pyramide. Par contre, les détails complexes de la caméra disparaissent en majorité dès la première réduction. On conçoit alors assez facilement que les similarités structurelles associées à la caméra sont très difficiles, voire impossibles à trouver sur les niveaux supérieurs de la pyramide.

Ceci nous conduit aux deux conclusions suivantes. La première est qu'en général, il est inutile d'avoir une différence de résolution importante entre la base et le dernier niveau de la pyramide. La deuxième conclusion est qu'il est préférable de diminuer le facteur de réduction entre les différents niveaux de la pyramide. Afin de mettre ceci en pratique, nous exploitons la pyramide en quinconce proposée dans la section 4.4. La figure 7.1 illustre la pyramide en quinconce obtenue à partir de l'image du cameraman, ainsi que la pyramide de différences associée (cf. section 4.4.4). On constate que le taux de réduction plus faible entre les niveaux de la pyramide a pour effet de supprimer plus progressivement les détails de l'image. De ce fait, les similarités structurelles entre les résolutions sont plus faciles à détecter.

Dans la suite de cette section, nous adaptons le modèle d'analyse-synthèse des similarités développé dans le chapitre 5 à une pyramide en quinconce.



Figure 7.1 – (a) Pyramide passe-bas en quinconce obtenue à partir de l'image du cameraman ($a = 0.85$). (b) Pyramide de différences associée (les niveaux de gris sont amplifiés d'un facteur 4).

7.2.1 Particularités

Comme le montre la figure 7.2, pour réaliser un agrandissement de facteur 2 avec une pyramide en quinconce, il faut produire deux niveaux de super-résolution successifs. Le premier est obtenu sur un maillage en quinconce et le deuxième sur un maillage carré. Hormis cette obligation de produire deux niveaux de super-résolution successifs au lieu d'un seul, l'algorithme global de l'agrandissement est à peu près identique à celui présenté dans la table 5.1.

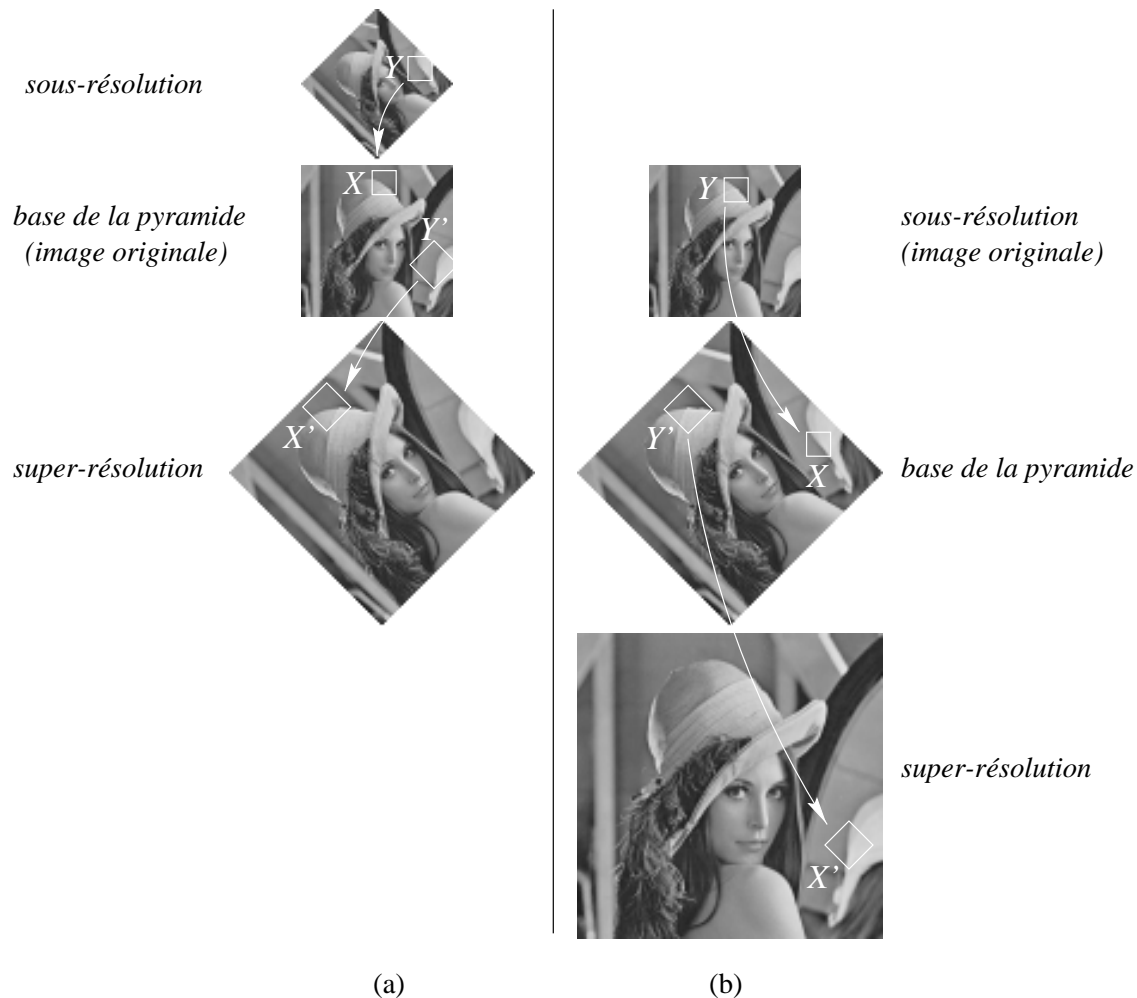


Figure 7.2 – Illustration des deux niveaux de super-résolution à produire pour agrandir une image de facteur 2 à partir d'une pyramide en quinconce. Cette figure illustre également un exemple de synthèse d'un bloc X' réalisée à partir d'un bloc Y' et de la relation de similarité entre les blocs pères X et Y .

Les étapes 1 et 2 demeurent assez semblables. La construction de la pyramide en quinconce (étape 1) est basée sur les développements que nous avons réalisés dans la section 4.4. La construction du domaine de recherche des similarités \mathcal{D} (étape 2) est réalisée par extraction de blocs de taille 5×5 pixels sur tous les niveaux de la pyramide en quinconce, sauf la base. Comme dans la section 5.4, les blocs sont

sélectionnés en fonction de leur écart-type et le calcul peut être réalisé soit à partir de la pyramide en quinconce, soit à partir de la pyramide de différences associée. Dans la suite, nous supposons que ce calcul est toujours réalisé à partir de la pyramide de différences afin de prendre en compte le contenu fréquentiel des blocs. Dans ce cas, la fixation de l'écart-type minimum σ_{min} doit prendre en compte le fait que les amplitudes des différences sont en moyenne beaucoup plus petites (environ deux fois) que dans le cas d'une pyramide de différences à facteur de réduction 2. Une fois les blocs sélectionnés, l'organisation du domaine et, en particulier, la classification des blocs, exploitent sans modification les algorithmes du chapitre 5.

Les étapes 3, 4 et 5 sont exécutées deux fois. En ce qui concerne la première fois, le niveau de super-résolution est disposé sur un maillage en quinconce. Son partitionnement (étape 3) est illustré sur la figure 7.3(a). On discerne quatre supports différents pour les blocs fils qui sont plus petits que les blocs pères, afin de générer des recouvrements entre ceux-ci. La recherche des similarités dans le domaine \mathcal{D} (étape 4) demeure identique à celle proposée dans le chapitre 5 lorsque celle-ci est globale (cf. section 5.6.2). Dans le cas d'une recherche locale, il est inutile d'exploiter les niveaux en quinconce. En effet, les blocs pères pour lesquels on recherche une similarité sont situés sur la base de la pyramide qui est disposée sur un maillage carré. Les blocs situés sur les niveaux en quinconce qui recouvrent les blocs pères sont porteurs des mêmes structures à une rotation de 45° près. En raison de cette rotation, il n'est pas possible de trouver localement un bloc similaire sur un niveau en quinconce. La synthèse des similarités (étape 5) est basée sur le même principe que celui décrit dans la section 5.7.1. La seule différence réside dans l'opérateur

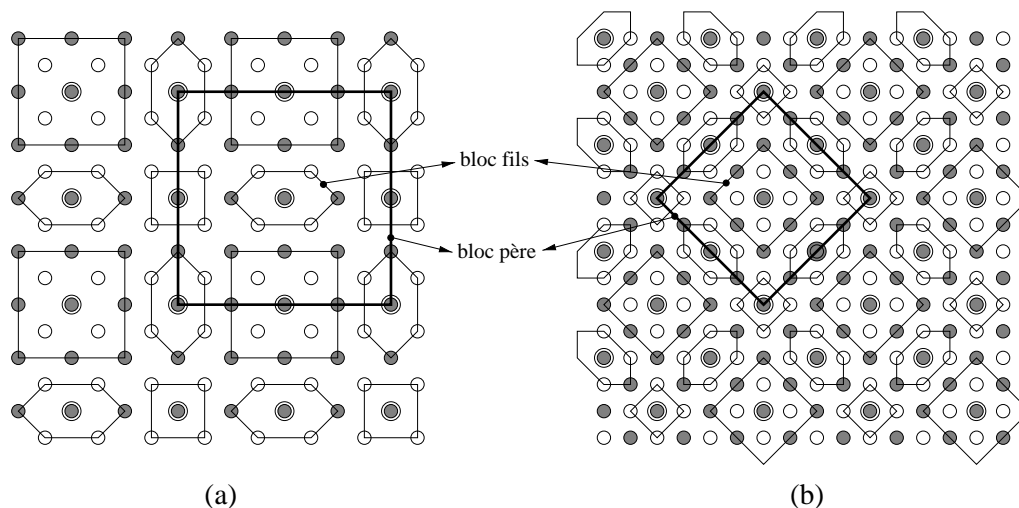


Figure 7.3 – Partitionnement du niveau de super-résolution de la pyramide en quinconce. La base de la pyramide est représentée par les points • et le niveau de super-résolution est représenté par les points • + ◦. Les doubles cercles symbolisent les centres des blocs fils. Le niveau de super-résolution est disposé : (a) sur un maillage en quinconce, (b) sur un maillage carré.

d'interpolation I qui doit réaliser l'interpolation du bloc résiduel ε de la transformation sur un maillage en quinconce. Nous exploitons toujours la transformée B-spline cubique pour réaliser cette interpolation. De plus, lorsque le bloc similaire Y est trouvé sur un niveau en quinconce, il faut effectuer une rotation de $+90^\circ$ du bloc X' . Ceci permet de compenser la première rotation à -45° imposée par le niveau en quinconce sur lequel est situé Y , puis de réaliser la deuxième rotation à $+45^\circ$ imposée par le niveau de super-résolution, également en quinconce. Ce phénomène est illustré par la synthèse du bloc X' sur la figure 7.3(a).

Lorsque l'on reconduit une deuxième fois les étapes 3, 4 et 5, le niveau de super-résolution en quinconce généré la première fois devient la nouvelle base de la pyramide. Le niveau de super-résolution à extrapoler est désormais disposé sur un maillage carré, comme le montre la figure 7.2(b). Son partitionnement (étape 3) est illustré sur la figure 7.3(b). Comme pour le partitionnement montré sur la figure 7.3(a), on discerne quatre supports différents pour les blocs fils. En fait, ces deux partitionnements sont identiques à une rotation de 45° près, et à un facteur d'échelle $\sqrt{2}$ près. La recherche des similarités (étape 4) demeure inchangée, sauf dans le cas d'une recherche locale où cette fois-ci, il devient inutile d'exploiter les niveaux disposés sur un maillage carré. Pour la synthèse des similarités (étape 5), le principe est le même que précédemment, hormis que le bloc X' doit cette fois-ci subir une rotation de -90° , comme illustré sur la figure 7.3(b). Par rapport à la production du premier niveau de super-résolution, la taille des blocs pères est inchangée mais leur nombre est doublé. La complexité algorithmique est donc deux fois plus élevée.

Que ce soit pour extrapoler le premier ou le deuxième niveau de super-résolution, précisons que les blocs non synthétisés sont toujours interpolés par une transformée B-spline cubique.

Sélection de blocs à partir d'une pyramide de différences en quinconce

Pour sélectionner les blocs du domaine \mathcal{D} , nous exploitons la pyramide de différences en quinconce que nous avons proposée dans la section 4.4.4. Un bloc appartenant à un niveau k de la pyramide passe-bas en quinconce est sélectionné si le bloc de même support sur le niveau $k+1$ de la pyramide de différences possède un écart-type supérieur à un seuil limite σ_{min} . Contrairement au cas d'une pyramide à facteur de réduction 2, le calcul de l'écart-type n'est donc pas réalisé sur le niveau k mais sur le niveau $k+1$ de la pyramide de différences. Ce décalage provient de la décomposition de l'image originale en fréquences diagonales d'une part et en fréquences verticales et horizontales d'autre part. En effet, un bloc situé sur un niveau impair (quinconce) est intéressant lorsqu'il possède des hautes fréquences diagonales, et celles-ci sont représentées sur le niveau supérieur pair de la pyramide de différences (cf. figure 7.1(b)). De même, un bloc situé sur un niveau pair est intéressant lorsqu'il possède des hautes fréquences verticales et horizontales, et celles-ci sont représentées sur le niveau supérieur impair de la pyramide de différences (cf. figure 7.1(b)).

De même, la mise en œuvre de la recherche d'une similarité pour un bloc père repose sur les conditions précédentes. Plus précisément, une recherche de similarité

à un bloc père est effectuée si le bloc de même support sur le niveau supérieur de la pyramide de différences possède un écart-type plus grand qu'un seuil limite σ_{min} prédéfini.

Agrandissement de facteur supérieur à deux

Comme dans la section 5.7.2, on peut réaliser un agrandissement de facteur supérieur à deux selon deux approches. La première et la plus simple consiste à itérer le modèle d'analyse-synthèse que l'on a décrit précédemment (étapes 3, 4 et 5) jusqu'à obtenir la résolution souhaitée. Le domaine de recherche des similarités \mathcal{D} est construit une fois pour toute à partir de la pyramide en quinconce initiale, ceci pour éviter une complexité algorithmique démesurée. Avec cette approche, la taille des blocs est maintenue constante mais leur nombre est doublé de niveau en niveau.

La deuxième approche consiste à effectuer la synthèse de chaque niveau en exploitant uniquement la première analyse réalisée pour obtenir l'agrandissement de facteur 2. La principale différence avec le modèle décrit dans la section 7.7.2 (deuxième approche: synthèse) est qu'en réalité, on bénéficie de deux codes de similarités au lieu d'un seul. Le premier est associé à la production du niveau de super-résolution disposé sur un maillage en quinconce, et le deuxième est associé à la production du niveau de super-résolution disposé sur un maillage carré. La synthèse des similarités est donc itérée en appliquant à chaque niveau de super-résolution le code de similarité associé et en doublant à chaque fois la taille en pixels des blocs afin de maintenir leur support constant. Pour interpoler le résidu de la transformation ε , on exploite toujours la méthode d'agrandissement par transformée B-spline cubique avec le facteur d'agrandissement correspondant au niveau extrapolé.

7.2.2 Résultats avec une image de référence

Dans cette section, nous comparons l'image agrandie par synthèse de similarités sur une pyramide en quinconce à une image de référence. Pour cela, nous réalisons l'agrandissement à partir d'une version réduite de l'image du cameraman. Dans le contexte d'une pyramide en quinconce, l'agrandissement de facteur 2 est réalisé en deux étapes. La première consiste à produire un niveau de super-résolution disposé sur un maillage en quinconce et la deuxième, à produire un niveau de super-résolution disposé sur un maillage carré. La figure 7.4 illustre les deux images de référence correspondant à ces deux étapes, ainsi que la pyramide en quinconce produite à partir de l'image réduite du cameraman. En ce qui concerne les paramètres de la méthode d'agrandissement, les écart-types minimum d'un bloc du domaine \mathcal{D} et d'un bloc pour lequel on recherche une similarité sont $\sigma_{min} = 5$. Ces valeurs correspondent à peu près à la moitié de celles utilisées dans le cadre d'une pyramide à facteur de réduction 2. Les autres paramètres sont définis par la configuration standard de la section 5.8.1.

Dans la suite de cette section, nous étudions tout d'abord la validité de l'hypothèse de préservation des similarités à travers les deux niveaux de super-résolution successifs. Ensuite, nous examinons l'influence de la valeur limite Q_{max} du résidu de

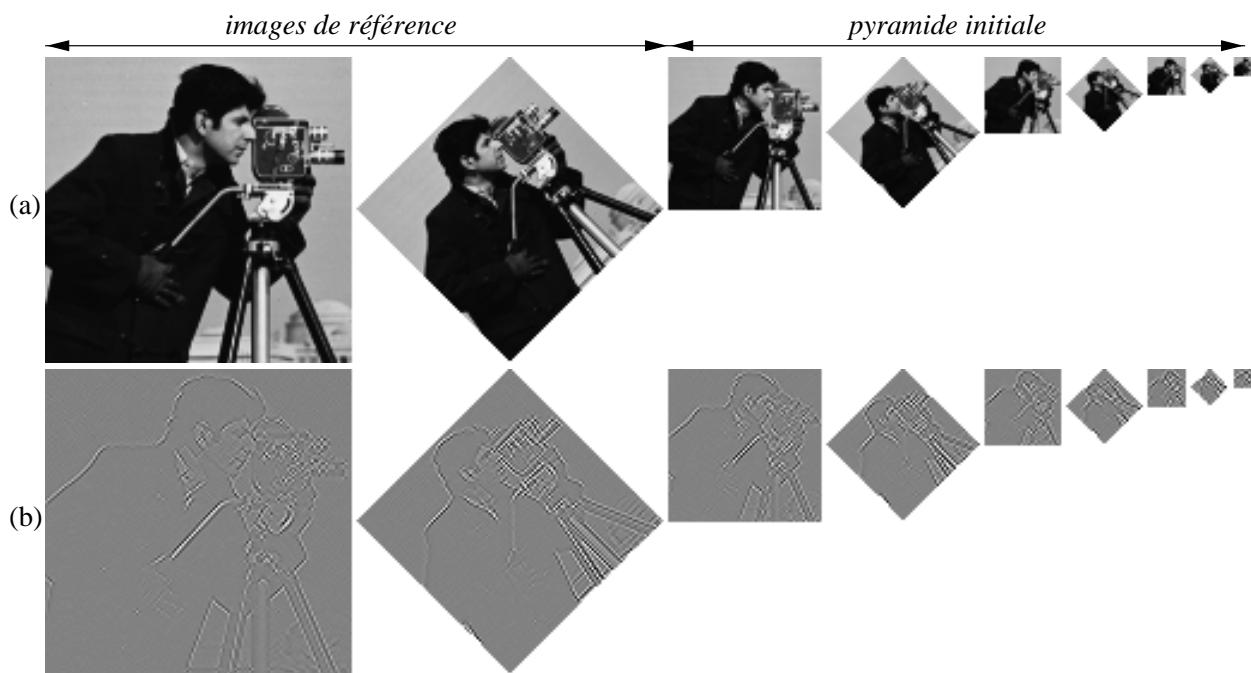


Figure 7.4 – (a) Pyramide passe-bas générée à partir de l'image du cameraman ($a = 0.85$). (b) Pyramide de différences associée (les niveaux de gris sont amplifiés d'un facteur 4).

similarité sur la qualité de l'agrandissement.

La préservation des similarités

Pour étudier la préservation des similarités à travers les deux niveaux de super-résolution, nous effectuons le même type de mesures que celles que nous avons réalisées dans la section 5.8.3. Plus précisément, la valeur moyenne $Q(\varepsilon'_i)$ du résidu de similarité entre les blocs fils R_i et Y'_i est exprimée en fonction de celle du résidu de similarité $Q(\varepsilon_i)$ entre les blocs pères X_i et Y_i .

Les figures 7.5(a-c-e) montrent les nuages de points obtenus pour le premier niveau de super-résolution disposé sur un maillage en quinconce, ceci selon différents contextes de recherche de similarités. Dans le cas d'une recherche locale (figure 7.5(a)) le nuage est relativement étiré, tandis que dans le cas d'une recherche globale (figures 7.5(c-e)), le nuage est plus compact : les résidus de similarité entre les blocs pères sont en moyenne plus faibles. Dans les deux cas, les résidus de similarité entre les blocs fils sont en moyenne plus faibles que lorsque la synthèse est réalisée avec une pyramide à facteur de réduction 2. Par exemple, en comparant les figures 5.28(b) et 7.5(c), on relève des résidus moyens respectivement de 15.5 et de 11.4. En fait, l'amplitude maximale d'une différence entre l'image agrandie et l'image de référence augmente nécessairement avec le facteur d'agrandissement.

Les nuages de points obtenus pour le deuxième niveau de super-résolution sont présentés sur les figures 7.5(b-d-f). Le résidu moyen entre les blocs pères est à chaque fois assez proche de celui calculé pour le premier niveau de super-résolution. En

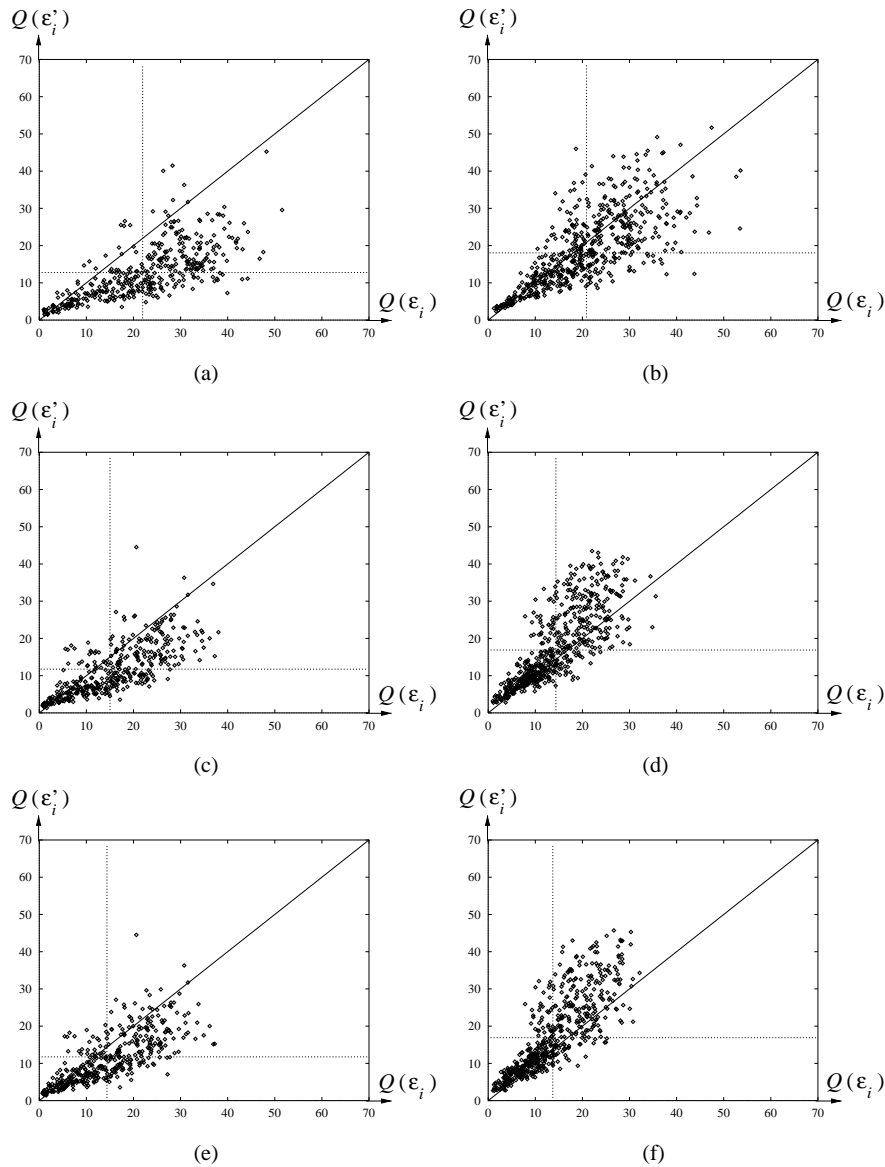


Figure 7.5 – Résidu de similarité entre les blocs fils en fonction du résidu de similarité entre les blocs pères dans le contexte d'une : (a-b) recherche locale de similarités, (c-d) recherche globale de similarités avec isométries *rotations*, (e-f) recherche globale de similarités avec isométries *rotations+réflexions*. Ces mesures sont effectuées sur le niveau de super-résolution disposé sur un maillage en quinconce (a-c-e) et sur celui disposé sur un maillage carré (b-d-f).

revanche, on remarque une nette dégradation du résidu moyen entre les blocs fils. Ceci s'explique par le fait que les résidus de similarité associés au deuxième niveau de super-résolution s'ajoutent à ceux du premier niveau.

En résumé, les différents nuages de points sur la figure 7.5 montrent que l'hypothèse de préservation des similarités se justifie pour la majorité des blocs synthétisés. Toutefois, lorsque le résidu de similarité entre deux blocs pères est trop important, la préservation de la similarité n'a pas vraiment d'intérêt. Il faut donc déterminer une valeur-limite au-delà de laquelle la synthèse n'est pas réalisée. C'est ce que nous allons étudier désormais.

Choix d'une limite sur le résidu de similarité

La figure 7.6 illustre l'évolution du $PSNR$ en fonction de la valeur-limite Q_{max} du résidu de similarité entre les blocs pères. La valeur maximale du $PSNR$ est atteinte avec $Q_{max} = 10$ lorsque la recherche est globale, et $Q_{max} = 15$ lorsqu'elle est locale. Contrairement au résultat illustré par la figure 5.34 dans le cas d'une recherche globale, l'ajout d'isométries de type *réflexion* n'améliore pas le $PSNR$.

Les résultats des agrandissements obtenus avec la valeur limite $Q_{max} = 13$ dans les différents contextes de recherche de similarités sont présentés sur les figures 7.7 et 7.8. Tout d'abord, on observe sur la figure 7.7 que les blocs synthétisés sont beaucoup moins nombreux lorsque la recherche est locale. Par ailleurs, dans le cas

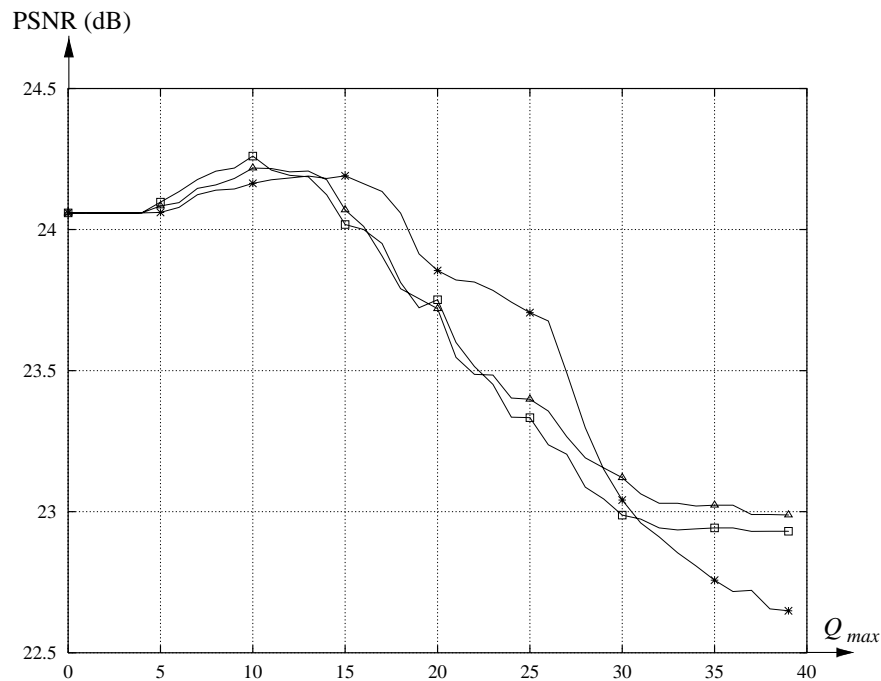


Figure 7.6 – Rapport signal à bruit en fonction de la valeur limite Q_{max} du résidu de similarité entre les blocs pères dans le contexte d'une : (*) recherche locale des similarités ($T = 5$), (Δ) recherche globale des similarités avec isométries *rotations*, (□) recherche globale des similarités avec isométries *rotations+réflexions*.

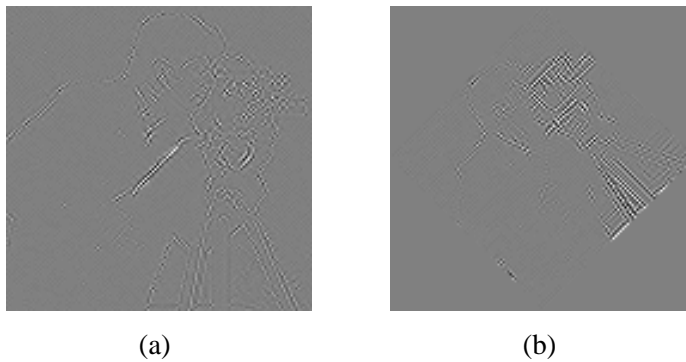


Figure 7.9 – Soustraction entre les images de différences originale et agrandie pour le niveau de super-résolution disposé : (a) sur un maillage carré, (b) sur un maillage en quinconce.

recherche de similarités. On observe à chaque fois que les traces associées au pourtour du cameraman se prolongent dans les hautes fréquences. Les traces associées au pied droit de la caméra et au bras de direction sont prolongées uniquement dans le cas d'une recherche globale des similarités.

La figure 7.9 montre les deux images générées en effectuant la soustraction des images de différences illustrées sur la figure 7.7(b) à celles de référence illustrées sur la figure 7.4(b). On remarque que les hautes fréquences produites sur les pourtours du cameraman et le pied droit de la caméra sont très proches de celles de l'image de référence. En effet, il y a très peu d'amplitude dans ces zones, que ce soit sur la figure 7.9(a) ou sur la figure 7.9(b). Les maxima d'amplitude sont en fait principalement situés au niveau des structures non synthétisées : col, caméra, pieds gauche et central. On remarque que la synthèse du bras droit de direction n'est pas très bonne.

7.2.3 Résultats sans image de référence

Dans cette section, nous présentons des résultats d'agrandissements obtenus dans le contexte général où on l'on ne dispose pas d'image de référence. Pour estimer la qualité des images agrandies, on ne peut plus utiliser de mesures comme celle du *PSNR*. Cependant, on peut toujours réaliser une analyse spectrale et exploiter des images de différences pour estimer visuellement la qualité des hautes fréquences produites par notre méthode d'agrandissement.

Dans la suite, nous commençons par nous intéresser à la répartition des similarités sur la pyramide en quinconce. Nous nous interrogeons en particulier sur l'équitabilité de cette répartition afin de connaître le nombre optimal de niveaux d'une pyramide en quinconce. Nous présentons par la suite quelques agrandissements de facteur 2 et 4 des images de Lenna et du cameraman. Enfin, nous terminons cette section en montrant des agrandissements de facteurs 2 de diverses images.

Répartition des similarités sur la pyramide

Les paramètres utilisés pour l'image du cameraman sont identiques à ceux utilisés précédemment. Pour l'image de Lenna, les écart-types minimum d'un bloc du

domaine \mathcal{D} et d'un bloc pour lequel on recherche une similarité sont $\sigma_{min} = 2$.

La table 7.1 indique le nombre de blocs du domaine \mathcal{D} extraits de chaque niveau de la pyramide en quinconce et la table 7.2 montre le nombre de similarités trouvées sur ces niveaux. Pour connaître la répartition des similarités entre les niveaux de la pyramide, nous calculons l'indice P_k défini par (5.21) pour chaque niveau k de la pyramide. Les résultats sont présentés dans la table 7.3 pour les deux niveaux de super-résolution générés.

Dans le cas du premier niveau de super-résolution (C-Q), les blocs pères sont situés sur la base de la pyramide (niveau 0). Que ce soit pour l'image de Lenna où pour celle du cameraman, ce sont surtout sur les niveaux pairs que sont trouvées les similarités, c'est-à-dire sur les niveaux disposés sur un maillage carré comme celui de la base. Pour l'image de Lenna, il est assez frappant de constater que le niveau 5 n'est pas du tout exploité. De plus, parmi les niveaux pairs, la majorité des similarités est trouvée sur le niveau 2. Ce constat rejoint celui effectué¹ à partir du tableau 5.10 dans le contexte d'une pyramide à facteur de réduction 2. La répartition des similarités entre tous les niveaux est nettement meilleure pour l'image du cameraman.

Dans le cas du deuxième niveau de super-résolution (Q-C), les blocs pères sont situés sur le premier niveau de super-résolution (en quinconce). Pour l'image de Lenna, P_k décroît lorsque k augmente et pour l'image du cameraman, la répartition est à peu près équitable.

En résumé, les deux exemples étudiés montrent bien que le nombre optimal de niveaux de la pyramide en quinconce dépend beaucoup de l'image à la base et que ce nombre est plutôt difficile à déterminer *a priori*.

Image	Niv. 1	Niv. 2	Niv. 3	Niv. 4	Niv. 5
<i>Lenna</i>	796 / 1568	423 / 784	266 / 288	142 / 144	32 / 32
<i>Cameraman</i>	854 / 1568	570 / 784	254 / 288	140 / 144	32 / 32

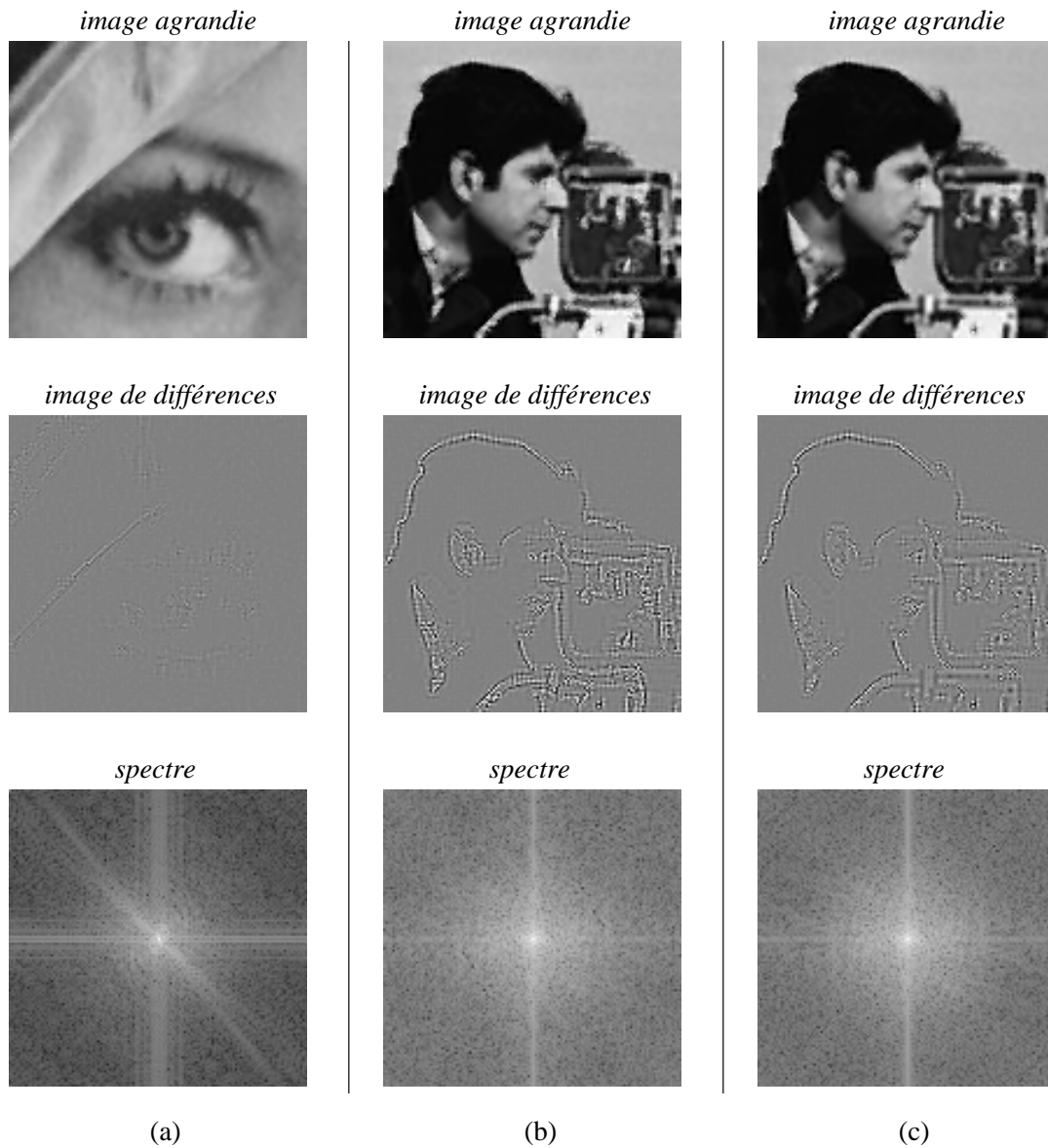
Table 7.1 – Niveau d'origine des blocs du domaine de recherche \mathcal{D} .

Image	Niv. 1	Niv. 2	Niv. 3	Niv. 4	Niv. 5
<i>Lenna</i> (C→Q)	113	69	27	4	0
<i>Lenna</i> (Q→C)	215	45	9	1	0
<i>Cameraman</i> (C→Q)	207	194	45	40	5
<i>Cameraman</i> (Q→C)	264	166	67	57	8

Table 7.2 – Niveau d'origine des similarités.

Image	Niv. 1	Niv. 2	Niv. 3	Niv. 4	Niv. 5
<i>Lenna</i> (C→Q)	1.11	1.27	0.79	0.22	0
<i>Lenna</i> (Q→C)	1.66	0.65	0.21	0.04	0
<i>Cameraman</i> (C→Q)	0.92	1.30	0.67	1.09	0.59
<i>Cameraman</i> (Q→C)	1.02	0.96	0.87	1.34	0.82

Table 7.3 – Indice de répartition des similarités sur les niveaux de la pyramide.

Figure 7.10 – Agrandissements de facteur 2, images de différences (les niveaux de gris sont amplifiés d'un facteur 4) et spectres associés: (a) image de Lenna, (b) image du cameraman, (c) image du cameraman avec $Q_{max} = 20$.

Agrandissement de facteur 2

Les figures 7.10(a-b) présentent les images agrandies respectivement de Lenna et du cameraman, sans limiter le résidu de similarité. Les images de différences permettent à la fois de localiser les hautes fréquences introduites et de visualiser leur amplitude.

L'image de différences associée à Lenna montre que les hautes fréquences sont créées essentiellement sur le bord inférieur du chapeau. D'ailleurs, le spectre montre que seule la trace associée au bord inférieur du chapeau est prolongée. Contrairement au résultat présenté sur la figure 5.39(c), la trace associée au bord supérieur du chapeau n'est donc pas prolongée. En fait, le résultat illustré sur la figure 7.10(a) est davantage en accord avec ce que l'on observe à travers les différents niveaux de la pyramide de différences montrée sur la figure 5.16. Seul le bord inférieur du chapeau traduit une véritable discontinuité. La diminution de l'amplitude constatée sur la pyramide de différences quand on descend les niveaux est confirmée par l'image de différences de la figure 7.10(a).

L'image de différences associée au cameraman sur la figure 7.10(b) montre que de nombreuses hautes fréquences sont créées dans l'image agrandie et que leur amplitude est relativement importante. Ceci est en accord avec ce que l'on peut observer sur la pyramide de différences illustrée sur la figure 5.16. L'image agrandie semble de meilleure qualité que celle obtenue par synthèse sur une pyramide à facteur de réduction 2 : les défauts sont moins nombreux et l'image semble plus naturelle. On peut diminuer l'importance des défauts restants en limitant le résidu de similarité des blocs synthétisés. Ainsi, la figure 7.10(c) montre le résultat obtenu avec $Q_{max} = 20$. Malgré une plus grande tolérance sur le résidu de similarité, on observe une meilleure qualité de l'image agrandie de la figure 7.10 que celle présentée sur la figure 5.40(c).

Agrandissement de facteur supérieur à 2

La figure 7.11 présente différents agrandissements de facteur 4 de l'image du cameraman. L'interpolation au plus proche voisin produit un effet de pixelisation, tandis que l'interpolation par transformée B-spline génère une image beaucoup plus naturelle, mais avec un flou excessif. Les images agrandies par synthèse de similarités avec $Q_{max} = 20$ ont des contours globalement plus nets. L'image obtenue par l'approche de synthèse unique (figure 7.11(c)) comporte de nombreux petits défauts qui sont dus à des décalages de structures. Comme le montre la figure 7.11(d), le renouvellement de l'analyse permet de corriger ces décalages. Malgré l'augmentation globale de la résolution constatée sur cette dernière image, il existe toujours des défauts structurels qui perturbent l'analyse visuelle, surtout au niveau de la caméra. Ceci s'explique notamment par l'accumulation des résidus de similarités à chaque niveau de super-résolution produit. Pour diminuer cet effet, une solution simple est de diminuer la valeur de Q_{max} . Cependant, ceci a pour effet de diminuer le nombre de blocs synthétisés, et donc d'augmenter le nombre de blocs interpolés

1. Le niveau 1 du tableau 5.10 correspond au niveau 2 du tableau 7.3.



Figure 7.11 – Agrandissements de facteur 4 de l'image du cameraman : (a) interpolation au plus proche voisin, (b) interpolation par transformée B-spline cubique, (c) approche par synthèse unique de similarités ($Q_{max} = 20$), (d) approche par analyse-synthèse de similarités ($Q_{max} = 20$).

par la transformée B-spline.

Résultats comparatifs avec diverses images

Les figures 7.12 à 7.15 illustrent des agrandissements de facteur 2 obtenus avec des images variées. L'image agrandie par synthèse de similarités sur une pyramide en quinconce peut à chaque fois être comparée à celle obtenue par une transformée B-spline cubique. Pour mieux évaluer l'information «haute-fréquence» produite par notre méthode, nous montrons à chaque fois une image de différences.

Pour simplifier, nous ne détaillons pas les paramètres utilisés pour chaque image agrandie. Nous précisons seulement que le résidu de similarité est limité par $Q_{max} = 30$ afin d'éviter des défauts importants. Notre objectif est de montrer les capacités de notre approche sur des images relativement différentes. L'image du babouin (figure 7.12) est très texturée : elle comporte beaucoup de hautes fréquences. L'image du pont (figure 7.13) possède des structures géométriquement simples. L'image du poivron (figure 7.14) comporte des structures moins géométriques et plus naturelles. L'image de Lenna (figure 7.12) est plutôt hétérogène, avec des textures au niveau des plumes du chapeau et des contours simples au niveau du pourtour du chapeau et de l'épaule. À chaque fois, l'image agrandie par synthèse de similarités présente une qualité supérieure aux images agrandies par transformée B-spline, l'amélioration portant surtout sur la netteté.



image originale



image agrandie par transformée B-spline cubique

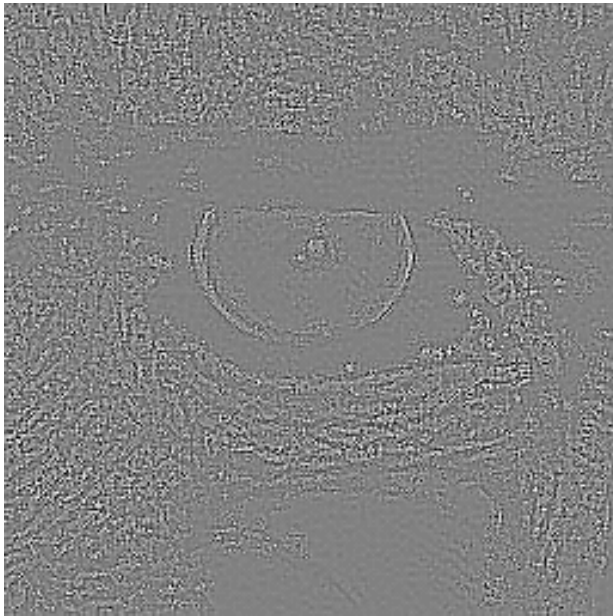


image de différences



image agrandie par synthèse de similarités

Figure 7.12 – Agrandissement de facteur 2 de l'image du babouin.



image originale



image agrandie par transformée B-spline cubique



image de différences



image agrandie par synthèse de similarités

Figure 7.13 – Agrandissement de facteur 2 de l'image du pont.

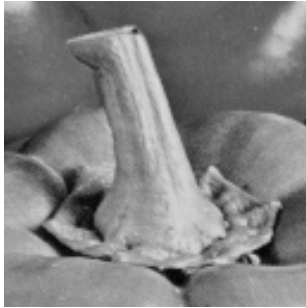


image originale

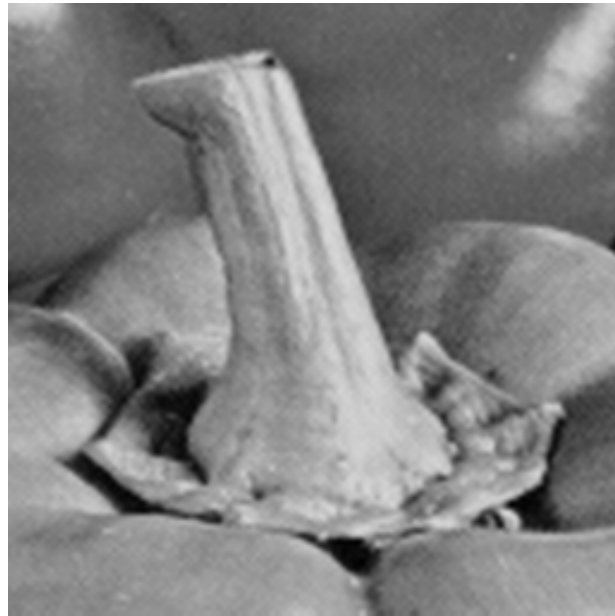


image agrandie par transformée B-spline cubique

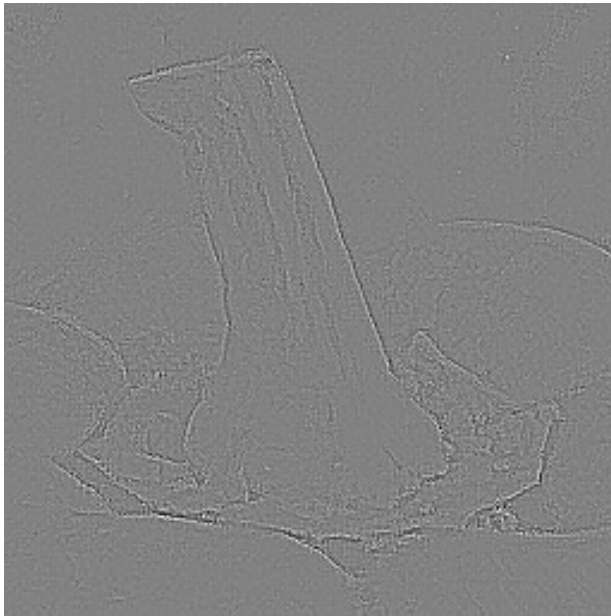


image de différences



image agrandie par synthèse de similarités

Figure 7.14 – Agrandissement de facteur 2 de l'image du poivron.



image originale



image agrandie par transformée B-spline cubique



image de différences



image agrandie par synthèse de similarités

Figure 7.15 – Agrandissement de facteur 2 de l'image de Lenna.

7.3 Régularisation d'agrandissements par induction

Dans le chapitre 6, nous avons proposé de considérer le problème de l'agrandissement comme un problème inverse de réduction. Ceci nous a amené à définir un ensemble admissible de solutions à l'agrandissement d'une image. Pour qu'une image agrandie par une méthode d'agrandissement quelconque appartienne à cet ensemble, il faut réaliser une projection sur celui-ci. L'interprétation fréquentielle de cet ensemble, réalisée dans la section 6.2.5, montre clairement que la projection revient à restaurer les basses fréquences de l'image agrandie. Plus précisément, lorsque le filtre passe-bas est idéal, la projection a pour effet de remplacer approximativement les basses fréquences de l'image agrandie par les fréquences de l'image originale. En d'autres termes, la projection sur l'ensemble induit d'une réduction permet de restaurer dans l'image agrandie l'information relative à l'image originale éventuellement perdue lors du processus d'agrandissement.

La majorité des méthodes d'agrandissement présentées dans le chapitre 3 est concernée par la régularisation vis-à-vis de la contrainte de réduction. En effet, ces méthodes ont pour objectif de produire de l'information haute-fréquence pour préserver une certaine netteté de l'image agrandie, mais elles ne se préoccupent absolument pas de préserver l'information donnée par l'image originale. La régularisation d'une image agrandie par ces méthodes permet finalement de faire le lien avec les méthodes de préservation fréquentielle décrites dans le chapitre 2.

Dans la suite de cette section, nous nous intéressons uniquement à la régularisation d'images agrandies avec notre méthode d'agrandissement par synthèse de similarités. Nous étudions cette régularisation à la fois dans le cas où l'on dispose d'une image de référence, et dans le cas réel où celle-ci n'existe pas. Précisons que la méthode de projections partielles est la méthode *MEPP* qui permet d'obtenir une projection exacte sur l'ensemble des solutions quelle que soit sa consistance.

7.3.1 Résultats avec une image de référence

Le résultat d'une régularisation dépend beaucoup du filtre passe-bas utilisé dans l'opération de réduction. D'un point de vue théorique, le meilleur résultat devrait être obtenu avec le filtre passe-bas idéal (sinus cardinal). Cependant, nous avons vu dans le chapitre 4 que ce filtre introduisait un effet d'ondulation important. En pratique, il est préférable d'utiliser un filtre spatial à petit support, comme par exemple le filtre de Burt de taille 5×5 pixels.

La figure 7.16 montre le résultat d'une régularisation réalisée avec différents filtres passe-bas. L'image initiale et l'image agrandie à régulariser sont illustrées dans la partie supérieure de la figure. L'image initiale est l'image réduite du cameraman que nous avons déjà utilisée à maintes reprises, et l'image agrandie est celle qui apparaît sur la figure 7.8(b). La régularisation de cette image en utilisant le filtre de Burt permet d'améliorer sa qualité, puisque l'on remarque un gain de 0.44 dB du *PSNR*. L'utilisation du filtre de Meer entraîne un gain nettement moins significatif puisque

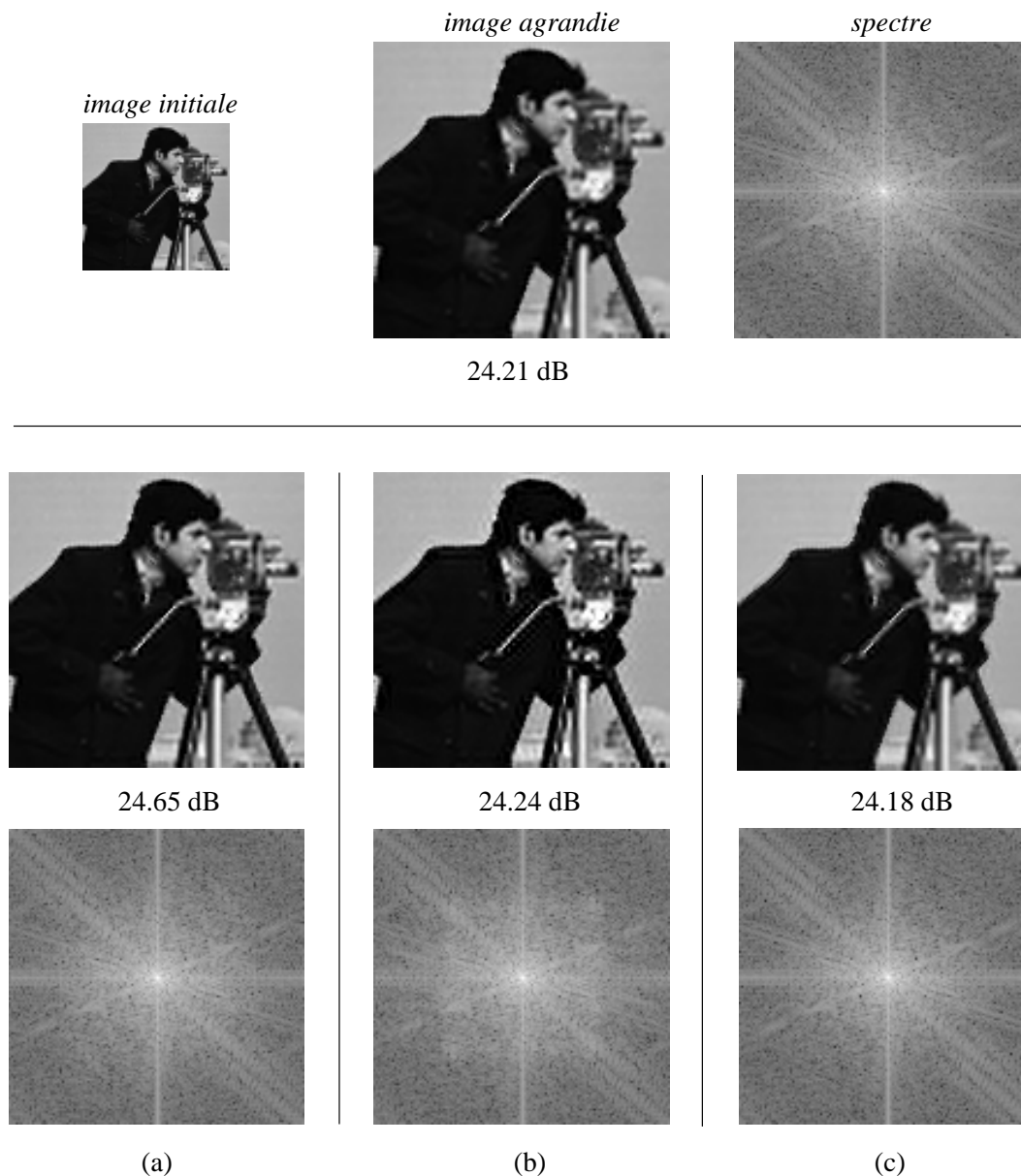


Figure 7.16 – Régularisation de l'image agrandie à partir d'une version réduite de l'image du cameraman en utilisant : (a) le filtre de Burt ($\beta = 0.625$), (b) le filtre de Meer, (c) le filtre cubique de Chin.

celui-ci est seulement de 0.03 dB. De plus, l'image régularisée présente un effet d'ondulation assez important qui se caractérise sur le spectre par l'apparition d'une fenêtre au niveau des basses fréquences. Ce résultat n'est pas vraiment surprenant puisque nous avons pu voir dans la section 4.3.1 que le filtre de Meer était une approximation du sinus cardinal sur un support 13×13 . Enfin, l'utilisation du filtre de Chin entraîne un léger effet d'ondulation beaucoup moins perceptible malgré son support 15×15 . Toutefois, on observe une dégradation du *PSNR* avec -0.03 dB.

En résumé, parmi les trois filtres utilisés, celui de Burt ($\beta = 0.625$) semble le plus

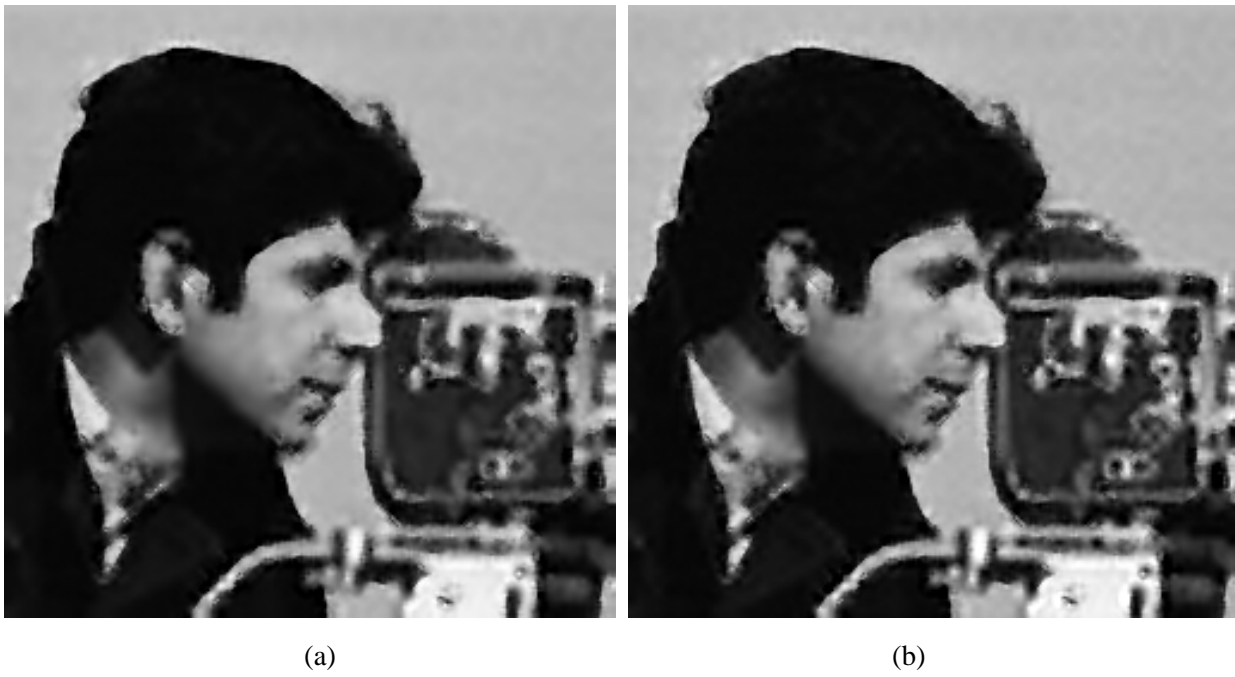


Figure 7.17 – Régularisation de l'image agrandie par un facteur 4 de l'image du cameraman. (a) Image à régulariser. (b) Image régularisée. (c) Soustraction entre (a) et (b).

adapté à la régularisation de l'image agrandie. Sans introduire d'artefacts, il permet de préserver les hautes fréquences introduites dans l'image agrandie et de restaurer les basses fréquences associées à l'image originale.

7.3.2 Résultats sans image de référence

La figure 7.17 illustre la régularisation de l'image du cameraman agrandie d'un facteur 4 qui apparaît sur la figure 7.11(d). Nous exploitons ici les développements réalisés dans la section 6.5 dans le cadre d'une induction avec un facteur d'agrandissement supérieur à 2. Le filtre passe-bas élémentaire que nous employons est le filtre de Burt ($\beta = 0.625$). Le filtre utilisé pour la régularisation est obtenu en convoluant le filtre 5×5 de Burt par une copie agrandie d'un facteur 2 de lui-même (cf. expression (6.63)). Le filtre ainsi construit est défini sur un support 13×13 .

Le résultat de la régularisation est montré sur la figure 7.17. Les corrections apportées par cette régularisation ne sont pas faciles à évaluer au premier coup d'œil. Pour faciliter l'analyse visuelle, une image de soustraction est présentée sur la figure 7.17(c). Les différences que l'on perçoit sur cette image montrent bien que les corrections sont réalisées au niveau des basses fréquences. De manière locale, les différences maximales se trouvent sur les points d'échantillonnage de l'image originale. Ceci s'explique par le fait que le poids maximal du filtre est au centre et que lors de la régularisation, le filtre est centré sur les points d'échantillonnage de l'image originale.

7.4 Compression réversible et transmission progressive d'images

Dans cette section, nous exploitons le principe d'induction du chapitre 6 pour effectuer un codage hiérarchique de l'image. Ce type de codage a notamment pour application la transmission progressive d'images sur réseaux [90]. Le destinataire reçoit tout d'abord une version grossière de l'image à bas débit qui est ensuite affinée par des transmissions successives de l'information restante. Ce destinataire a la possibilité d'interrompre la transmission à tout moment, ce qui permet de limiter la quantité d'informations transitant sur le réseau. Le codage hiérarchique peut également être utile pour rechercher une image dans une base de données. La consultation peut se faire très rapidement en exploitant les résolutions grossières des images de la base, à condition que celles-ci soit pré-calculées et stockées. Enfin, ce codage permet de transmettre facilement une résolution de l'image adaptée au périphérique utilisé : écran de télévision haute-définition (HDTV) avec incrustation d'écrans basse-définition, imprimantes à résolutions variables, etc. Pour cette dernière application, divers constructeurs ont proposé récemment le format FlashPix qui adopte une structure pyramidale très rudimentaire de l'image [79].

Une propriété que l'on recherche tout particulièrement pour un codage hiérarchique est sa capacité à compresser l'image originale [11], [115], [134]. Pour simplifier, nous nous intéressons uniquement à une compression réversible de l'image. Ce type de compression est surtout recherché dans des domaines où chaque pixel révèle une information importante qui ne doit pas être modifiée. On pensera en particulier à l'imagerie médicale ou satellitaire.

Dans le contexte d'une compression sans perte d'information, le taux de compression est limité par l'entropie de l'image. Pour que celle-ci soit la plus faible possible, il faut trouver une représentation de l'image qui réalise une décorrélation des pixels. Parmi les méthodes les plus utilisées, on trouve notamment le codage RLE (*run length encoding*) qui code les longueurs des séquences de pixels ayant un même niveau de gris. Cette méthode est surtout efficace lorsque l'image a peu de niveaux de gris. On trouve également le codage par plans de bits où l'image de $2^8 = 256$ niveaux de gris est vue comme la superposition de 8 plans d'épaisseur 1 bit. Chaque plan est donc une image binaire qui est codée efficacement à l'aide d'un codage RLE. Le codage par plans de bits est assez efficace mais il est néanmoins sensible aux erreurs de transmission. Enfin, il existe aussi le codage prédictif DPCM (*differential pulse code modulation*) qui consiste à prédire la valeur d'un pixel en fonction de ses pixels voisins, puis à coder l'erreur de prédiction par un codage entropique.

Le codage prédictif DPCM est particulièrement performant pour compresser une image sans perte d'information, mais il n'est pas du tout adapté à la transmission progressive. En fait, seul le codage par plans de bits peut éventuellement être utilisé pour cette application. On parle alors d'un *codage hiérarchique à résolution fixe*. L'image est reconstruite avec une taille identique à celle de l'image originale en affinant progressivement chaque pixel. Cependant, ce type de codage demeure assez limité pour une transmission progressive et la qualité des images successives au cours du décodage est loin d'être optimale. Pour cette application, les méthodes les plus intéressantes sont basées sur un *codage hiérarchique à résolution variable*. L'image est reconstruite en augmentant progressivement sa résolution. Ces méthodes exploitent une structure pyramidale de l'image. D'un point de vue historique, la pyramide de référence est la pyramide laplacienne proposée en 1983 par Burt et Adelson [22]. Elle permet de décorrélérer efficacement les pixels à des résolutions variables de telle sorte que chaque niveau possède une variance et une entropie plus faibles que celles de l'image originale. Depuis 1983, de nombreuses recherches se sont portées sur la décomposition pyramidale d'images. On peut citer en particulier la pyramide *spline* de *Unser et al.* qui reprend le principe de construction de la pyramide laplacienne, mais où les niveaux de la pyramide gaussienne sont obtenus par une transformée B-spline [138], [139]. De manière plus générale, ces approches s'intègrent dans la théorie des ondelettes de Mallat [91].

Dans la suite, nous commençons par rappeler le principe de construction d'une pyramide laplacienne, puis nous interprétons son contenu dans le domaine fréquentiel. Nous discutons ensuite de quelques améliorations apportées dans la littérature. Enfin, nous proposons une amélioration personnelle de la pyramide laplacienne en exploitant le principe d'induction du chapitre 6.

7.4.1 La pyramide laplacienne et ses améliorations

La construction d'une pyramide laplacienne est basée sur l'exploitation de la pyramide de Burt avec le filtre w de taille 5×5 que nous avons étudié dans la section 4.3.2. Cette pyramide, souvent appelée *pyramide gaussienne*, est une collection d'images G_K, G_{K-1}, \dots, G_0 où G_K est l'image originale de taille $2^K \times 2^K$ et G_{K-1}, \dots, G_0 représentent les versions à résolution décroissante de G_K obtenue selon la procédure itérative suivante :

$$(0 < k \leq K) \quad G_{k-1} = \Delta(G_k) = [w * G_k] \downarrow 2 \quad (7.1)$$

La différence entre un niveau de la pyramide gaussienne et l'agrandissement du niveau supérieur conduit à la construction d'un niveau de la pyramide laplacienne. Ce principe se décrit par :

$$(0 < k \leq K) \quad L_k = G_k - \Xi(G_{k-1}) \quad (7.2)$$

où Ξ désigne l'opérateur d'agrandissement défini par :

$$\Xi(G_{k-1}) = [4w * G_{k-1}^{\uparrow 2}] \quad (7.3)$$

où la notation $G_k^{\uparrow 2}$ est décrite par (2.26). Le facteur 4 permet de normaliser le filtre w . L'agrandissement de G_{k-1} est donc obtenu en intercalant des valeurs nulles entre ses échantillons, puis en convoluant l'image par le filtre w et en quantifiant les niveaux de gris sur des valeurs entières. On voit que l'on retrouve ici la formulation de l'interpolation par des fonctions polynomiales à travers l'expression (2.25). En réalité, l'opérateur Ξ fait une véritable interpolation uniquement lorsque $\beta = 0.5$, puisque dans ce cas les valeurs des pixels de G_{k-1} sont conservées. On remarque que le filtre correspondant est celui d'une interpolation linéaire illustrée sur la figure 2.7(c).

En faisant abstraction de la quantification des niveaux de gris, on peut obtenir une expression relativement simplifiée de L_k en fonction de G_k et disposer d'une interprétation fréquentielle d'une pyramide laplacienne :

$$L_k = G_k - 4w * (H_k \cdot (w * G_k)) \xleftrightarrow{TF} \hat{L}_k = \hat{G}_k - 4\hat{w} \cdot (\hat{H}_k * (\hat{w} \cdot \hat{G}_k)) \quad (7.4)$$

où la fonction H_k réalise la décimation spatiale de facteur 2 symbolisée par $\downarrow 2$ dans (7.1). Cette fonction est définie par (4.7). En utilisant l'expression de \hat{H}_k donnée par (4.8), on obtient :

$$\hat{L}_k(u, v) = \hat{G}_k(u, v) - \hat{w}(u, v) \sum_{s,t=0}^1 \hat{w}(u + s2^{k-1}, v + t2^{k-1}) \cdot \hat{G}_k(u + s2^{k-1}, v + t2^{k-1}) \quad (7.5)$$

Cette dernière expression peut être présentée de manière plus intelligible par :

$$\hat{L}_k = \hat{G}_k \cdot (1 - \hat{w}^2) - \hat{E}_k \quad (7.6)$$

où \hat{E}_k se déduit facilement de (7.5) et représente le repliement spectral issu de l'opération de décimation dans l'expression (7.1).

Dans l'expression (7.6), lorsque \hat{w} est le filtre passe-bas idéal, $1 - \hat{w}^2$ est un filtre passe-haut idéal et \hat{E}_k est nul. Dans ce cas particulier, chaque niveau de la pyramide laplacienne L_k représente les hautes fréquences du niveau G_k associé dans la pyramide gaussienne. En d'autres termes, la pyramide laplacienne peut être vue comme une décomposition de l'image originale en bandes de fréquences. En pratique, avec le filtre 5×5 de Burt, $1 - \hat{w}^2$ approxime grossièrement le filtre passe-haut idéal et la composante du repliement spectral \hat{E}_k n'est pas nulle. La figure 7.18 illustre le filtre $1 - \hat{w}^2$ pour quelques valeurs du paramètre β . On remarque que la corrélation fréquentielle entre les niveaux de la pyramide laplacienne est plus faible lorsque la valeur de β est importante (environ 0.6). Cependant, en considérant la figure 4.12(b), on observe que la composante du repliement spectral \hat{E}_k diminue avec la valeur de β .

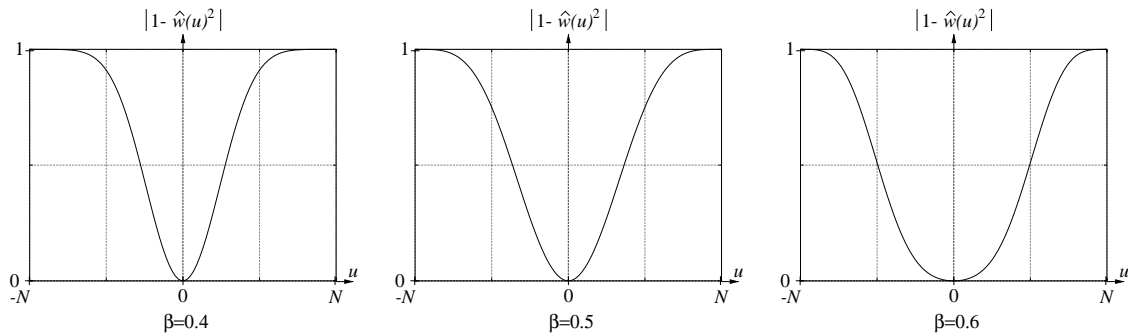


Figure 7.18 – Illustration du filtre $1 - \hat{w}^2(u)$ pour différentes valeurs du paramètre β .

La pyramide laplacienne est relativement proche de la pyramide de différences que nous avons présentée dans le chapitre 4. En comparant les expressions (4.46) et (7.6), il apparaît clairement que pour un filtre donné, la décomposition fréquentielle obtenue avec la pyramide laplacienne est forcément moins bonne qu'avec une pyramide de différences. En particulier, la corrélation fréquentielle entre les niveaux de la pyramide laplacienne est plus importante. Cependant, celle-ci a l'avantage sur la pyramide de différences de permettre une reconstruction exacte de chaque niveau de la pyramide gaussienne à partir du niveau supérieur et du niveau associé dans la pyramide laplacienne :

$$(0 < k \leq K) \quad G_k = \Xi(G_{k-1}) + L_k \quad (7.7)$$

Ainsi, à partir d'une pyramide laplacienne et du niveau le plus haut de la pyramide gaussienne, il est possible de reconstruire exactement l'image originale. En résumé, la pyramide laplacienne permet tout à la fois de représenter une image selon ses résolutions décroissantes et de coder uniquement les différences entre les résolutions successives. Elle est donc particulièrement adaptée à la transmission progressive d'images avec une compression réversible des données. Pour augmenter le taux de compression global de la pyramide laplacienne, des chercheurs ont apportés certaines

améliorations que nous présentons rapidement dans la suite avant de présenter notre propre amélioration.

Améliorations de Unser

A partir de l'analyse précédente, on remarque que la pyramide laplacienne a quelques insuffisances. Tout d'abord, l'opérateur d'agrandissement Ξ défini par (7.3) réalise une véritable interpolation uniquement lorsque $\beta = 0.5$. Pour les autres valeurs de β , les échantillons initiaux ne sont pas conservés :

$$\beta \neq 0.5 \implies \Xi(G_{k-1}) \downarrow 2 \neq G_{k-1} \quad (7.8)$$

De plus, l'opérateur d'agrandissement Ξ n'est pas vraiment l'inverse de l'opérateur de réduction Δ , c'est-à-dire :

$$\Delta(G_k) \neq \Delta \circ \Xi \circ \Delta(G_k) \quad (7.9)$$

Enfin, la variance des niveaux de la pyramide laplacienne n'est pas forcément minimale puisque la corrélation fréquentielle entre les niveaux est relativement importante (cf. figure 7.18).

Pour améliorer ces divers points, Unser a proposé d'utiliser deux filtres correcteurs agissant chacun, soit sur l'opérateur de réduction, soit sur l'opérateur d'agrandissement [135], [90]. La stratégie de Unser est donc de ne pas modifier les opérateurs Δ et Ξ mais d'opérer un pré-traitement et un post-traitement adaptés. Plus précisément, le problème (7.8) est corrigé en opérant une pré-convolution de G_{k-1} avec un filtre chargé de compenser l'erreur d'interpolation. Pour minimiser l'énergie de L_k , il faut minimiser la perte d'information lors de la réduction de G_k afin d'avoir une corrélation minimale entre L_k et G_{k-1} . Pour cela, Unser place un filtre correcteur après l'opérateur Δ . Il est évalué pour minimiser l'énergie de L_k par les moindres carrés. Ce filtre a également la propriété de corriger le problème (7.9).

Pyramide laplacienne réduite

D'un point de vue compression, le principal défaut de la pyramide laplacienne réside dans le nombre total des coefficients à coder. Celui-ci est supérieur au nombre de pixels de l'image originale et le rapport entre les deux tend vers 4/3.

Pour amener ce rapport à l'unité, Houlding *et al.* ont proposé de supprimer le filtrage dans l'opération de réduction pour ne conserver que la décimation spatiale de facteur 2 [69]. L'agrandissement des niveaux est quant à lui réalisé par interpolation linéaire, c'est-à-dire que le filtre w de Burt est remplacé par un filtre triangulaire dans (7.3). Grâce à cette simplification extrême, un coefficient sur quatre de la pyramide laplacienne devient nul et le nombre de coefficients restants est identique au nombre de pixels de l'image. De plus, on remarque que les problèmes (7.8) et (7.9) sont résolus d'emblée.

Cette optimisation permet de diminuer l'entropie globale de la pyramide laplacienne, ce qui améliore le taux de compression global. Par contre, la décimation

brutale sans filtrage réalisée itérativement pour construire la pyramide pseudo-gaussienne détériore considérablement les structures de l'image originale. Cette détérioration est d'ailleurs de plus en plus importante lorsque l'on se rapproche de l'apex de la pyramide. Lors de la reconstruction progressive de l'image, on a donc une qualité d'image moins bonne avec les premiers niveaux transmis. En résumé, l'optimisation de Houlding *et al.* se révèle surtout intéressante dans un contexte de compression réversible. Par contre, dans un contexte de transmission progressive, elle génère une baisse significative de la qualité au début de la reconstruction, ce qui limite son intérêt pour cette application.

Aiazzi *et al.* ont généralisé l'approche précédente en remplaçant l'interpolation linéaire par une interpolation cubique pour l'agrandissement des niveaux [4]. Remarquons qu'en raison de la propagation de l'effet d'aliasing à travers la pyramide pseudo-gaussienne, il n'est pas du tout prouvé que l'utilisation d'un filtre approchant le filtre fréquentiel idéal pour l'agrandissement des niveaux permette de minimiser l'entropie de la pyramide laplacienne.

Il existe également une structure pyramidale proposée par Goldberg *et al.*, appelée *pyramide de différences réduite* [57], dont le principe de construction est très proche de celui de la pyramide laplacienne réduite de Houlding *et al.* La principale différence est que la construction de la pyramide de différences réduite est basée sur un codage arithmétique astucieux et non pas sur les notions classiques du filtrage.

Quantification rétroactive

Pour améliorer le taux de compression d'une pyramide laplacienne, Burt et Adelson réalisent une quantification uniforme des niveaux de gris [22]. Cependant, pour éviter d'avoir une trop grande perte lors de la reconstruction de l'image originale, le pas de quantification diminue avec la résolution des niveaux de la pyramide. Avec ce modèle de quantification, la compression devient irréversible.

Aiazzi *et al.* ont proposé récemment un modèle de quantification qui permet de reconstruire exactement l'image originale [5]. Pour obtenir cette propriété, la quantification est réalisée progressivement depuis l'apex jusqu'au niveau 1 de la pyramide laplacienne, la base (niveau 0) n'étant pas quantifiée pour garantir la réversibilité de la compression. Chaque niveau est recalculé pour prendre en compte la quantification du niveau supérieur. Cette opération se décrit par les deux relations suivantes :

$$(0 < k \leq K) \quad \tilde{L}_k = G_k - \Xi(\tilde{G}_{k-1}) \quad , \quad \tilde{G}_k = Q_k(\tilde{L}_k) + \Xi(\tilde{G}_{k-1}) \quad (7.10)$$

où Q_k est l'opérateur de quantification associé au niveau k .

La pyramide $\tilde{L}_K, \tilde{L}_{K-1}, \dots, \tilde{L}_0$ constitue la pyramide laplacienne recalculée en prenant en compte l'opération de quantification. L'entropie de chaque niveau \tilde{L}_k est donc plus grande que celle du niveau initial L_k . La transmission progressive de l'image originale est réalisée en transmettant successivement les niveaux quantifiés $Q_0(\tilde{L}_0), Q_1(\tilde{L}_1), \dots, Q_{K-1}(\tilde{L}_{K-1})$, puis finalement \tilde{L}_k . La baisse significative de l'entropie des niveaux quantifiés compense largement l'augmentation de l'entropie de

\tilde{L}_k .

En résumé, le modèle de quantification d'Aiazzi *et al.* permet de diminuer la quantité d'information nécessaire à la transmission des niveaux supérieurs de la pyramide laplacienne. Evidemment, cette amélioration se fait au détriment de la qualité de l'image dans les premières phases de la reconstruction.

Décorrélacion inter-niveaux

Même avec une décomposition parfaite en bandes de fréquences de l'image originale, on observe toujours une corrélation importante entre les niveaux d'une pyramide, notamment au niveau des contours. Ce phénomène est particulièrement visible sur une pyramide laplacienne, comme le montre l'exemple de la figure 7.21 obtenu à partir de l'image de Lenna.

Pour diminuer la corrélation inter-niveaux dans une pyramide quelconque, Qiu a proposé de placer des prédicteurs entre chaque niveau de la pyramide laplacienne [110]. Chaque niveau L_k est prédit en fonction du niveau supérieur L_{k-1} . Le résultat de la prédiction est alors soustrait à L_k pour générer un niveau d'entropie plus faible. En itérant ce principe, on obtient une nouvelle pyramide laplacienne :

$$(2 < k \leq K) \quad L_k^* = L_k - P_k(L_{k-1}) \quad (7.11)$$

où P_k est l'opérateur de prédiction du niveau L_k . Il est obtenu selon un procédé d'estimation réalisée à partir des deux niveaux supérieurs à L_k pour que $P_k(L_{k-2}) \approx L_{k-1}$. Plus précisément, les niveaux L_{k-1} et L_{k-2} sont découpés en blocs de sorte que chaque bloc de L_{k-1} soit lié à un bloc père dans L_{k-2} . Le prédicteur P_k est estimé par optimisation linéaire avec un algorithme classique des moindres carrés à partir de l'ensemble des couples de blocs père-fils.

La méthode de décorrélacion de Qiu permet d'augmenter le taux de compression de la pyramide laplacienne tout en préservant la réversibilité de cette compression. De plus, il n'est pas nécessaire de transmettre les prédicteurs lors de la transmission progressive de la pyramide, puisque ceux-ci sont directement estimés à partir des premiers niveaux reçus. Ceci se fait évidemment au prix d'une plus grande complexité algorithmique.

Cependant, on remarque que la décorrélacion des niveaux de la pyramide laplacienne ne peut s'appliquer qu'à partir du quatrième niveau en partant de l'apex. En effet, l'apex est le niveau le plus haut de la pyramide gaussienne et il ne peut pas être utilisé pour décorrélacion les niveaux inférieurs de la pyramide laplacienne. En outre, l'apprentissage du prédicteur requiert deux niveaux successifs de la pyramide laplacienne. En d'autres termes, pour que l'algorithme de prédiction soit vraiment intéressant dans le cadre de la transmission progressive d'images, il faut que la pyramide laplacienne ait un nombre suffisant de niveaux.

7.4.2 Amélioration avec le principe d'induction

Comme Unser [135], nous remarquons que l'opérateur d'agrandissement Ξ utilisé par Burt et Adelson n'est pas l'inverse de l'opérateur de réduction Δ (cf. expression

(7.9)). Ceci empêche l'entropie de la pyramide laplacienne d'être optimale. Pour corriger ce problème, Unser place un premier filtre correcteur après la réduction et un deuxième avant l'agrandissement pour obtenir une interpolation parfaite. Son approche est un peu particulière : pour que Ξ soit l'inverse de Δ , il modifie les deux opérateurs à la fois.

Dans la suite, nous proposons une approche beaucoup plus générale que celle de Unser, où seul l'opérateur d'agrandissement Ξ est changé. La pyramide gaussienne demeure donc intacte.

Ainsi, pour corriger le problème (7.9), nous proposons de remplacer l'agrandissement $\Xi(G_{k-1})$ dans (7.2) par une induction de G_{k-1} sur l'ensemble $,^k$ défini par :

$$(0 < k \leq K) \quad ,^k = \Delta^{-1}(G_{k-1}) = \{X_k \mid \Delta(X_k) = G_{k-1}\} \quad (7.12)$$

En fait, on retrouve l'expression (6.2) à différentes résolutions. Cet ensemble a la propriété d'être non vide puisque pour toute résolution k , il contient au moins un élément :

$$(0 < k \leq K) \quad G_k \in ,^k \quad (7.13)$$

L'ensemble $,^k$ est donc consistant et on peut réaliser l'induction de G_{k-1} sur $,^k$ en utilisant l'une des méthodes de projections partielles développées dans le chapitre 6.

La construction de la nouvelle pyramide laplacienne générée avec le principe d'induction est donnée par :

$$(0 < k \leq K) \quad R_k = G_k - \text{Proj}_{\Gamma^k}(\Xi(G_{k-1})) \quad (7.14)$$

Pour ne pas confondre cette pyramide avec celle de Burt et Adelson, nous la désignons désormais *pyramide résiduelle*. Dans (7.14), la projection exacte de $\Xi(G_{k-1})$ sur $,^k$ est obtenue avec la méthode de projections partielles *MEPA* (cf. section 6.3.6). En utilisant à chaque niveau $\Xi(G_{k-1})$ comme image inductrice, on garantit une baisse de l'énergie de R_k vis-à-vis de L_k .

Proposition 7.1 (Minimisation de l'énergie) *Pour toute résolution k , R_k a une énergie inférieure ou égale à celle de L_k si :*

(i) *L'image inductrice est $\Xi(G_{k-1})$.*

(ii) *La projection sur $,^k$ est exacte.*

Cette proposition est assez facile à démontrer. En effet, en considérant à la fois (7.13) et le fait que $,^k$ est convexe, on a nécessairement :

$$(0 < k \leq K) \quad d(G_k, \text{Proj}_{\Gamma^k}(\Xi(G_{k-1}))) \leq d(G_k, \Xi(G_{k-1})) \quad (7.15)$$

Comme pour la pyramide laplacienne, la transmission progressive est réalisée en transmettant successivement R_0, R_1, \dots, R_K . Sachant que $R_0 \equiv G_0$, le destinataire reconstruit la pyramide gaussienne initiale et, finalement, l'image originale, en appliquant la relation suivante :

$$(0 < k \leq K) \quad G_k = R_k + \text{Proj}_{\Gamma^k}(\Xi(G_{k-1})) \quad (7.16)$$

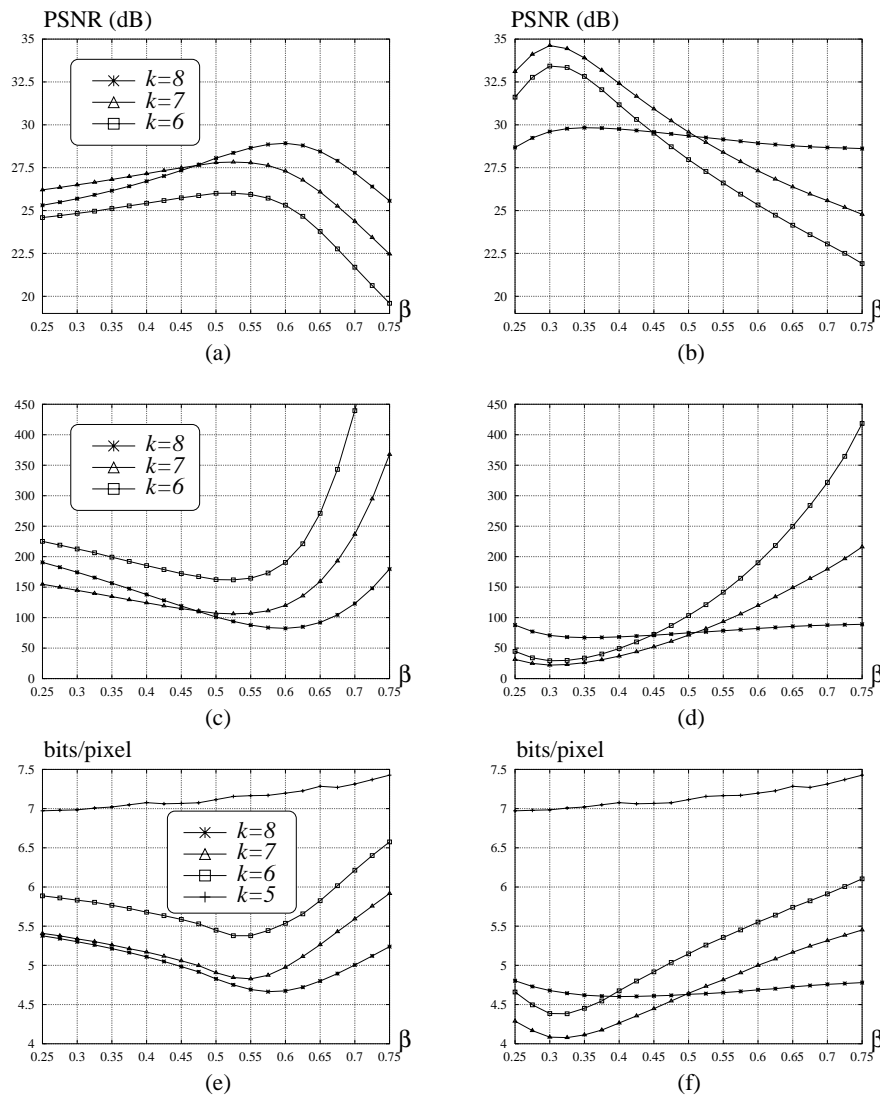


Figure 7.19 – Résultats obtenus pour l'image de Lenna (256×256 pixels). (a-b) Rapport signal à bruit de crête (*PSNR*) des niveaux de la pyramide gaussienne agrandis par : (a) convolution avec le filtre de Burt, (b) induction. (c-d) Variance des niveaux de : (c) la pyramide laplacienne, (d) la pyramide résiduelle. (e-f) Entropie des niveaux de : (e) la pyramide laplacienne, (f) la pyramide résiduelle.

On peut faire également une observation plutôt inattendue : la valeur maximale du *PSNR* est ici obtenue lorsque $\beta \approx 0.3$, c'est-à-dire lorsque le filtre w est encore plus « aplati » qu'une gaussienne. Cette observation se vérifie avec d'autres images que celle de Lenna et c'est toujours pour une valeur proche de 0.3 que la valeur optimale du *PSNR* est atteinte. Ceci est particulièrement important et nous reviendrons sur ce point dans la conclusion de ce chapitre. Outre ce décalage important de la valeur de β , on remarque que le gain du *PSNR* est très important pour les niveaux $k = 7$ et $k = 6$, puisqu'il est respectivement de 6.8 dB et de 7.4 dB. Ce phénomène est certainement dû au filtre qui supprime une bande de fréquence très large lorsque

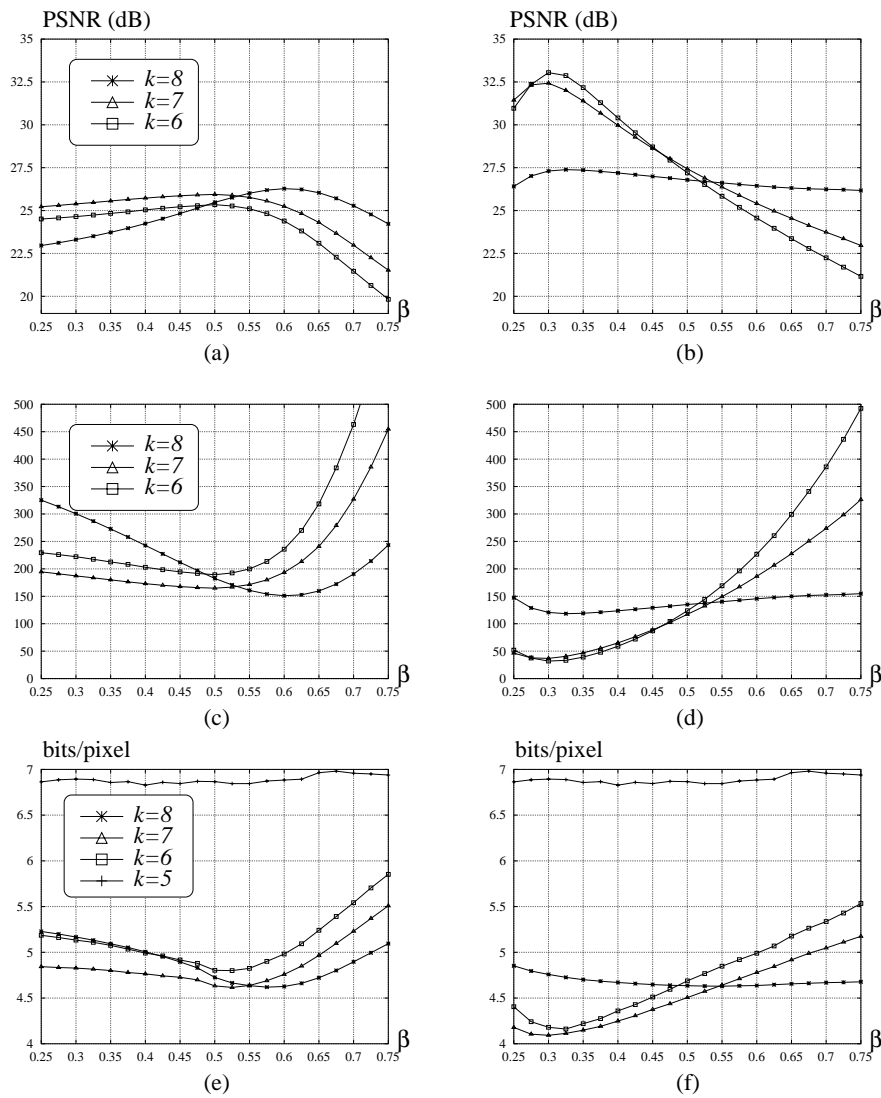


Figure 7.20 – Résultats obtenus pour l'image du cameraman (256×256 pixels). (a-b) Rapport signal à bruit de crête (PSNR) des niveaux de la pyramide gaussienne agrandis par : (a) convolution avec le filtre de Burt, (b) induction. (c-d) Variance des niveaux de : (c) la pyramide laplacienne, (d) la pyramide résiduelle. (e-f) Entropie des niveaux de : (e) la pyramide laplacienne, (f) la pyramide résiduelle.

$\beta \approx 0.3$. Ainsi, après la première réduction, la bande de fréquences restante occupe un espace très réduit par rapport à la capacité des niveaux successifs. En d'autres termes, il y a moins d'informations à prédire pour les niveaux $k = 7$ et $k = 6$, et par conséquent, l'agrandissement par induction est plus performant pour ces niveaux.

Les figures 7.19(c) et 7.19(d) montrent la variance des trois premiers niveaux de la pyramide laplacienne et de la pyramide résiduelle. Les observations sont assez similaires à celles effectuées sur les courbes liées au *PSNR*, avec toutefois une inversion au niveau de la progression. Ces deux figures confirment expérimentalement la proposition 7.1.

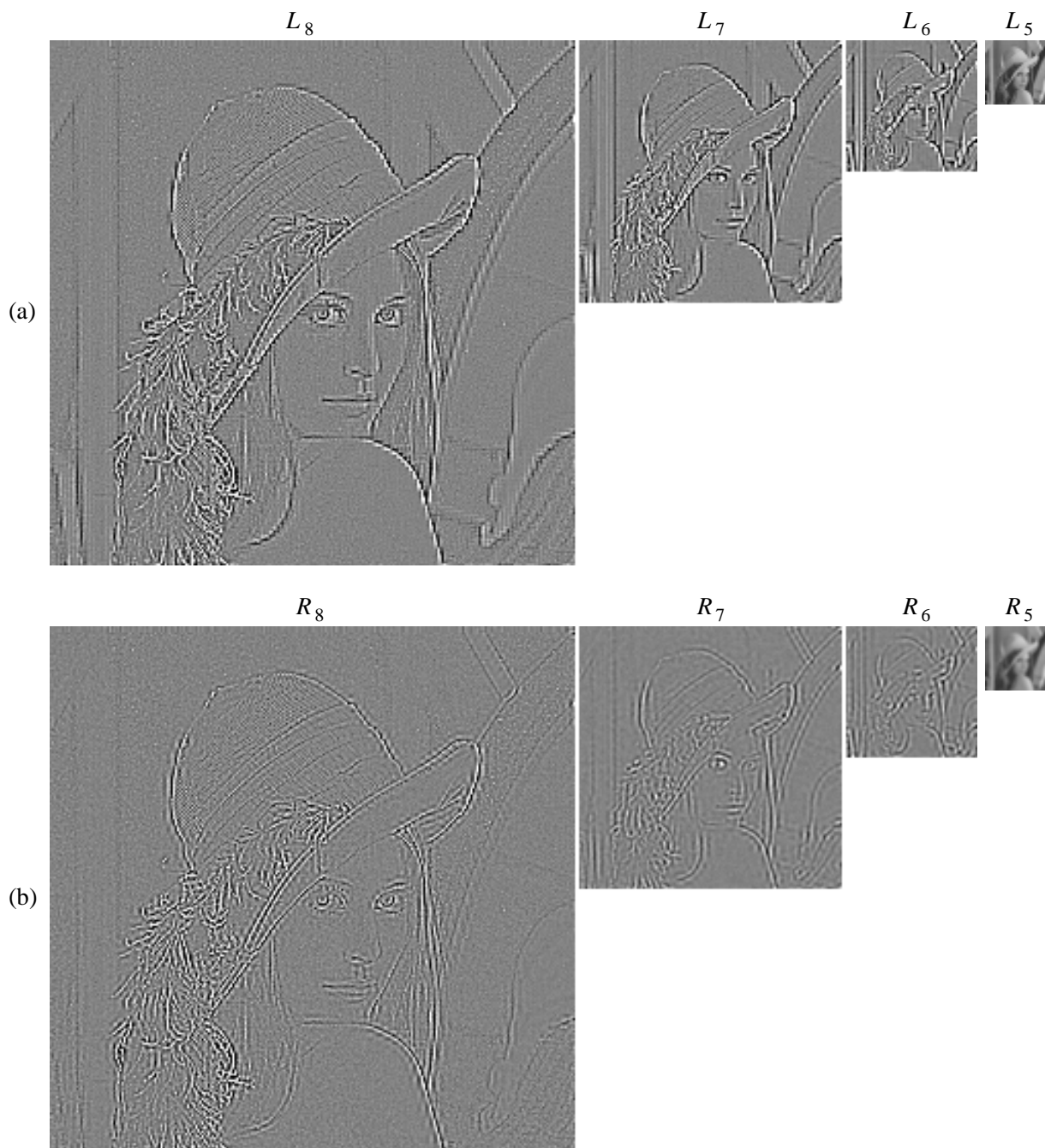


Figure 7.21 – Image de Lenna (256×256 pixels) : (a) pyramide laplacienne avec $\beta = 0.575$, (b) pyramide résiduelle avec $\beta = 0.35$. Les niveaux de gris des pyramides laplacienne et résiduelle sont amplifiés d'un facteur 4.

Les figures 7.19(e) et 7.19(f) montrent l'entropie des niveaux de la pyramide laplacienne et de la pyramide résiduelle. Il s'agit de l'entropie calculée à partir de l'histogramme de l'image selon la formule :

$$H = - \sum_{i=0}^{255} p_i \log_2(p_i) \quad (7.18)$$

où la probabilité p_i est le nombre de pixels ayant le niveau de gris i , divisé par le nombre total de pixels de l'image.

L'entropie du niveau le plus haut ($k = 5$) est nettement supérieure à celle des autres niveaux puisqu'il s'agit du niveau G_5 de la pyramide gaussienne. L'importance de la compression est directement liée au calcul de l'entropie de la pyramide. En ramenant l'entropie des niveaux supérieurs à celle de la base de la pyramide et en effectuant la somme, on obtient une entropie totale fournissant une estimation de la compression-limite du codage. Pour la pyramide laplacienne, l'entropie totale est minimale lorsque $\beta = 0.575$: dans ce cas, on obtient 6.33 bits/pixel. Pour la pyramide résiduelle, la valeur minimale est atteinte lorsque $\beta = 0.35$, et la compression est dans ce cas 6.03 bits/pixel. A titre de comparaison, précisons qu'un codage entropique directement réalisé sur l'image originale de Lenna permet d'obtenir une compression maximale de 7.44 bits/pixel.

La figure 7.20 illustre des résultats très comparables obtenus avec l'image du cameraman³ de 256×256 pixels.

Les pyramides laplacienne et résiduelle obtenues respectivement avec $\beta = 0.575$ et $\beta = 0.35$ sont illustrées sur la figure 7.21. On voit assez facilement que l'amplitude des différences est globalement moins importante sur la pyramide résiduelle. Celle-ci présente également moins de corrélation entre les différents niveaux que la pyramide laplacienne.

Les amplitudes des différences sur le niveau R_7 , et surtout sur le niveau R_6 , sont beaucoup plus faibles que celles situées sur le niveau R_8 . Cette propriété particulièrement intéressante dans le contexte d'une transmission progressive est due au filtre de Burt qui supprime une bande très large de hautes fréquences lorsque $\beta = 0.35$. C'est également pour cette raison que le dernier niveau R_5 est visuellement plus flou que le niveau L_5 de la pyramide laplacienne.

Transmission et réception progressives

La figure 7.22(a) montre les niveaux de la pyramide gaussienne reconstruits à partir de la pyramide laplacienne illustrée sur la figure 7.21(a). Il s'agit en fait de la reconstruction progressive de l'image originale réalisée par le destinataire. Pour avoir un confort visuel maximum, nous supposons que le destinataire agrandit les niveaux successifs de la pyramide gaussienne à une taille identique à celle de l'image originale. Les différents niveaux sont alors agrandis avec la méthode d'interpolation

3. Cette image est disponible à l'adresse <http://www.dip.ee.uct.ac.za/imageproc/stdimages/greyscale/camera.tif>.

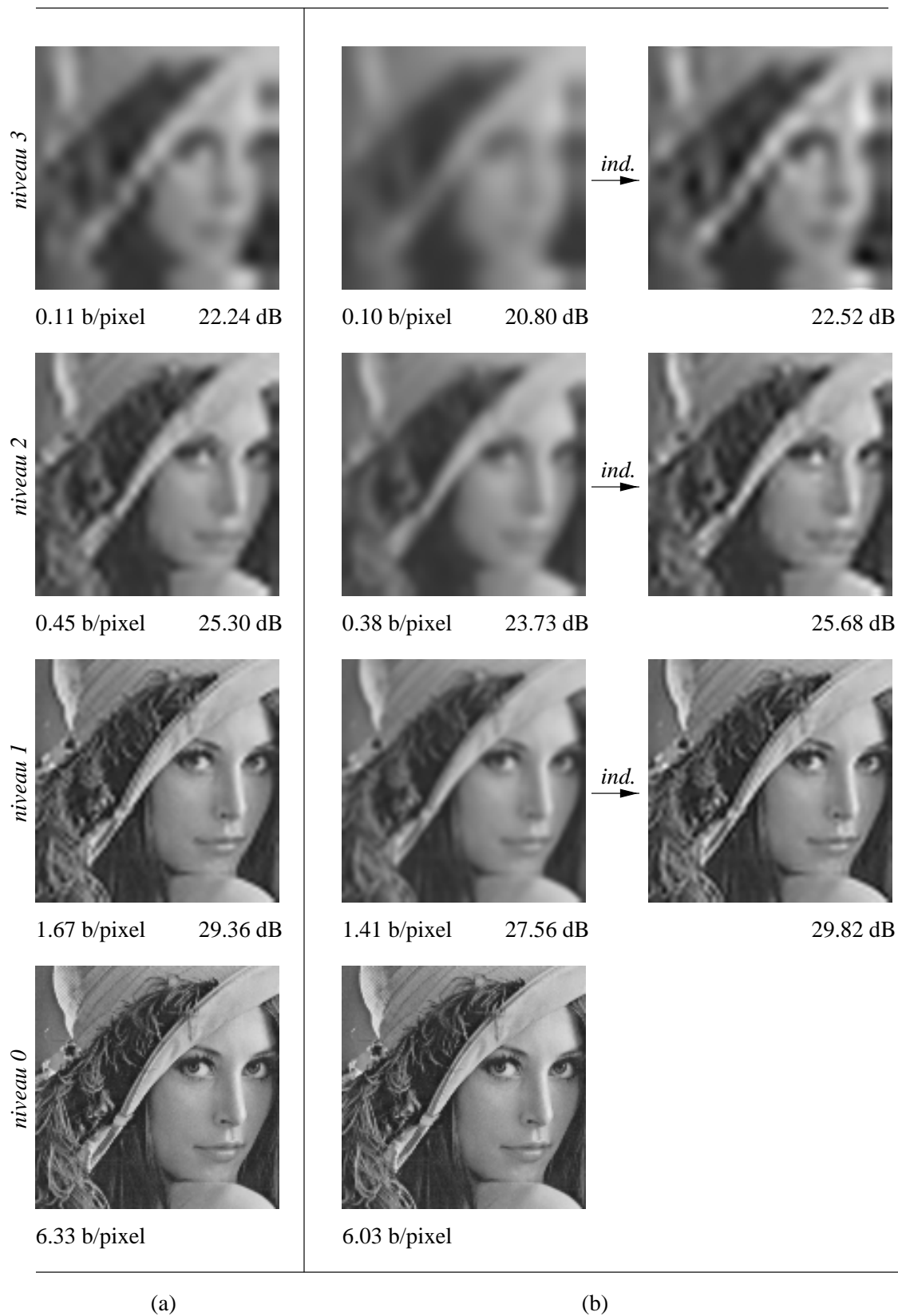


Figure 7.22 – Transmission et réception progressives de l'image de Lena (256×256 pixels) : (a) par utilisation de la pyramide laplacienne ($\beta = 0.575$), (b) par utilisation de la pyramide résiduelle ($\beta = 0.35$). Le débit est estimé par un calcul d'entropie.

par transformée B-spline cubique que l'on peut considérer comme la méthode la plus intéressante parmi celles décrites dans le chapitre 2. Le destinataire visualise donc successivement les niveaux représentés de haut en bas sur la figure 7.22(a) et peut interrompre la transmission à tout moment. Le nombre de bits par pixel de l'image originale nécessaire pour achever la construction de chaque niveau est indiqué en-dessous (l'estimation du codage entropique est réalisée avec la formule (7.18)). La qualité des images successives par rapport à l'image originale est évaluée par la mesure du *PSNR* qui est également indiquée en-dessous de chaque niveau (cf. formule (3.32)). La reconstruction illustrée sur la figure 7.22(a) montre le meilleur résultat que l'on peut espérer avec une pyramide laplacienne, tant sur le plan du taux de compression que sur celui de la qualité des niveaux successifs.

La colonne de gauche, sur la figure 7.22(b) présente les niveaux de la pyramide gaussienne reconstruits à partir de la pyramide résiduelle illustrée sur la figure 7.21(b). Le nombre de bits par pixel associé à chaque niveau est inférieur à celui des niveaux correspondants sur la figure 7.22(a). En d'autres termes, la pyramide résiduelle permet de réaliser une économie sur la quantité de bits nécessaires à la transmission des niveaux. Toutefois, on remarque que la qualité des images reconstruites est nettement moins bonne, puisqu'elles sont beaucoup plus floues et que le *PSNR* accuse une baisse importante. Ce problème est dû au paramètre $\beta = 0.35$ utilisé pour la construction de la pyramide résiduelle. Pour le corriger, il suffit de réaliser une projection de chaque image agrandie successive sur l'ensemble induit par le niveau correspondant sur la pyramide gaussienne comme nous l'avons vu dans la section 7.4.2. Par exemple, le niveau 3 est projeté sur l'ensemble \mathcal{G}_5^8 induit par le niveau G_5 , puis le niveau 2 est projeté sur l'ensemble \mathcal{G}_6^8 induit par le niveau G_6 , etc. La colonne de droite sur la figure 7.22(b) présente le résultat de chaque projection. On observe une nette amélioration de la qualité des images agrandies. Dans chaque cas, le *PSNR* est même supérieur à celui obtenu sur les niveaux correspondants de la figure 7.22(a). Ainsi, que ce soit au niveau du taux de compression ou au niveau de la qualité de la reconstruction progressive, la pyramide résiduelle donne de meilleurs résultats que la pyramide laplacienne.

Les tables 7.4 et 7.5 montrent les résultats obtenus avec l'image de Lenna⁴ de 512×512 pixels utilisée comme référence dans la littérature.

	Niv. 4	Niv. 3	Niv. 2	Niv. 1	Niv. 0
<i>Entropie</i> (bits/pixel)	0.028	0.11	0.41	1.47	5.47
<i>PSNR</i> (dB)	22.19	25.19	29.05	34.68	-

Table 7.4 – Transmission et réception progressives de l'image de Lenna de 512×512 pixels par utilisation de la pyramide laplacienne ($\beta = 0.575$).

4. Cette image est disponible à l'adresse http://www.icsl.ucla.edu/~ipl/psnr_images.html.

	Niv. 4	Niv. 3	Niv. 2	Niv. 1	Niv. 0
<i>Entropie</i> (bits/pixel)	0.027	0.09	0.34	1.21	5.15
<i>PSNR</i> avant induction (dB)	20.75	23.59	27.0	31.98	-
<i>PSNR</i> après induction (dB)	22.43	25.55	29.5	35.29	-

Table 7.5 – Transmission et réception progressives de l’image de Lenna de 512×512 pixels par utilisation de la pyramide résiduelle ($\beta = 0.35$).

7.5 Conclusion

Nous avons proposé dans ce chapitre diverses améliorations et applications de nos deux méthodes d’agrandissement.

Tout d’abord, pour améliorer la méthode d’agrandissement par synthèse de similarités, nous avons proposé de remplacer la pyramide à facteur de réduction 2 par une pyramide à facteur de réduction $\sqrt{2}$. La suppression des détails étant plus progressive, les similarités structurelles sont plus nombreuses. Les résultats obtenus montrent d’ailleurs une meilleure qualité dans l’agrandissement d’images complexes (cf. figure 7.10(b)). Cette amélioration entraîne néanmoins une complexité algorithmique plus grande, puisque pour obtenir un agrandissement de facteur 2, il faut estimer une résolution intermédiaire en quinconce.

Dans un deuxième temps, nous avons exploité la méthode d’agrandissement par induction pour régulariser les images agrandies vis-à-vis de la contrainte de réduction. Nous avons vu que le filtre de Burt $\beta = 0.625$ était bien adapté pour cette régularisation et permettait d’augmenter le *PSNR* de manière significative (+0.44 dB) en restaurant efficacement l’information perdue sur l’image originale.

Enfin, la méthode d’agrandissement par induction a également été appliquée dans un contexte de compression réversible et de transmission progressive d’images. Elle a été utilisée dans la construction d’une pyramide laplacienne pour remplacer l’opérateur d’agrandissement classique basé sur un filtrage. Les résultats ont montré que la pyramide construite, baptisée pyramide résiduelle, permettait d’obtenir une meilleure compression que la pyramide laplacienne classique (-0.3 bits/pixel). Le gain est surtout très important sur les niveaux supérieurs de la pyramide résiduelle (+7 dB et +0.6 bits/pixel), ce qui est particulièrement intéressant pour la transmission progressive sur réseau. Lors de la reconstruction progressive de l’image, la méthode d’agrandissement par induction peut également être utilisée pour agrandir les images intermédiaires à une même résolution que l’image originale. Nous avons montré que dans ce cas, la qualité de ces images était meilleure que celle des images intermédiaires générées à partir de la pyramide laplacienne (entre +0.28 dB et +0.46 dB).

En ce qui concerne le filtre optimal pour la compression, nous avons noté que le meilleur taux de compression était obtenu avec $\beta = 0.575$ dans le cas de la pyramide laplacienne. Avec ce paramètre, le filtre de Burt est localement assez proche du filtre idéal (sinus cardinal). Ceci s’explique assez bien par la théorie de l’échantillonnage :

le filtre idéal est celui qui minimise l'énergie sur la pyramide lorsque l'on procède par réduction puis interpolation. En revanche, dans le cas de la pyramide résiduelle, le meilleur taux de compression est obtenu lorsque $\beta = 0.35$. Avec ce paramètre, le filtre est proche d'une distribution gaussienne, et par conséquent, il est relativement éloigné du sinus cardinal. Ce constat est particulièrement intéressant. Pour le moment, nous n'avons aucune certitude quant à l'explication de ce phénomène. Toutefois, nous remarquons que lorsque $\beta = 0.35$, l'ensemble $,^k$ est plutôt étiré dans l'espace, tandis que pour $\beta = 0.575$, il est plutôt compact. Il semble donc que l'étirement de $,^k$ a pour effet de diminuer la distance entre la projection de l'image inductrice et l'image originale qui appartient à $,^k$.

Chapitre 8

Conclusion générale

Nous avons proposé dans ce mémoire deux méthodes d'agrandissement complémentaires. La première présentée dans le chapitre 5 a pour objectif de maintenir la netteté de l'image en introduisant des hautes fréquences. La deuxième présentée dans le chapitre 6 permet de restaurer dans l'image agrandie les fréquences de l'image originale qui sont éventuellement modifiées par la première méthode d'agrandissement. Une application de cette deuxième méthode pour la compression réversible et la transmission progressive d'images a également été étudiée dans le chapitre 7.

Le début de ce mémoire traite de différentes méthodes d'agrandissement d'images qui existent dans la littérature.

Nous avons tout d'abord présenté les méthodes classiques dont l'objectif est d'interpoler l'image initiale en préservant ses fréquences. Ces méthodes sont basées sur le théorème d'échantillonnage de Shannon et notamment sur l'hypothèse que la fréquence d'échantillonnage utilisée pour l'acquisition de l'image est deux fois supérieure à la fréquence de Nyquist. En pratique, cette hypothèse est très rarement vérifiée et l'application du théorème de Shannon conduit à une image agrandie visuellement floue.

Nous avons ensuite présenté des méthodes d'agrandissement ayant pour objectif de préserver les structures de l'image afin de maintenir une certaine netteté dans l'image agrandie. Pour cela, ces méthodes produisent de l'information fréquentielle au-delà de la bande passante du système d'acquisition. De manière générale, elles s'intéressent principalement à la préservation des contours de l'image en leur ajoutant des hautes fréquences. Parmi l'ensemble de ces méthodes, nous avons décrit en détail trois méthodes représentatives : l'interpolation adaptative, l'extrapolation fréquentielle et le zoom fractal. L'interpolation adaptative a pour inconvénient de préserver idéalement tous les contours de l'image comme s'il s'agissait de discontinuités parfaites. Ceci procure un effet de netteté artificielle dans l'image agrandie. En procédant par une analyse simultanée en espace et en échelle, l'extrapolation fréquentielle permet de distinguer les véritables discontinuités à travers les maxima d'ondelettes et d'ajouter des hautes fréquences uniquement au niveau des contours

associés. Par rapport aux deux méthodes précédentes, le zoom fractal est particulièrement original puisqu'il ne cherche pas directement à préserver les structures, mais plutôt à préserver les similarités entre les structures. L'avantage de cette approche est que la modélisation est indépendante des structures elles-mêmes. Ainsi, le zoom fractal peut aussi bien préserver les contours que les textures. De plus, il est possible de préserver les caractéristiques d'un contour, que celui-ci soit l'expression d'une discontinuité ou non. En fait, ceci dépend de la qualité de la similarité trouvée. Le zoom fractal a donc pour principal intérêt de simplifier la modélisation : il n'est plus nécessaire d'adapter un modèle spécifique à chaque type de structure.

Notre contribution s'est portée essentiellement sur l'étude de deux nouvelles méthodes d'agrandissement.

La première méthode est apparentée au zoom fractal. En reprenant le principe de préservation des similarités à sa base et en prenant en compte le contexte de l'agrandissement plutôt que celui de la compression, nous avons apporté de nombreuses modifications et améliorations au zoom fractal classique aussi bien dans la phase d'analyse que dans celle de synthèse.

Concernant la phase d'analyse, la recherche des similarités est réalisée sur une pyramide d'images afin de prendre en compte la théorie de l'échantillonnage. Nous avons étudié l'utilisation d'une pyramide à facteur de réduction deux, puis celle d'une pyramide à facteur de réduction $\sqrt{2}$ en ajoutant des niveaux en quinconce. Cette dernière pyramide augmente la probabilité de trouver des similarités de bonne qualité en supprimant l'information plus progressivement. Pour construire efficacement cette pyramide, nous avons proposé une approche utilisant un filtre paramétré à petit support soumis à des contraintes aussi bien spatiales que fréquentielles. Par ailleurs, nous avons étudié différents modèles de similarités inter-blocs en agissant sur les paramètres d'une transformation massique polynomiale et sur une transformation spatiale réalisant des isométries. D'après nos expérimentations, le modèle de similarité basé sur une transformation massique linéaire et des isométries de type *réflexion* est celui qui donne globalement les meilleurs résultats avec un coût algorithmique raisonnable. Remarquons que ce modèle est celui utilisé généralement par le zoom fractal. Cependant, la notion d'opérateur contractant étant absente de notre modèle, les coefficients de la transformation linéaire ne sont soumis à aucune contrainte particulière. Nous avons également proposé d'exploiter une pyramide de différences pour sélectionner les blocs admissibles à une similarité. Ceci nous a permis de prendre en compte l'aspect fréquentiel des blocs et en particulier de considérer uniquement ceux qui sont porteurs de hautes fréquences. Nous avons ensuite étudié deux stratégies de recherche des similarités sur la pyramide : une recherche locale et une recherche globale. La recherche locale est surtout adaptée pour des blocs positionnés sur des contours simples qui se conservent sur les niveaux supérieurs de la pyramide. La recherche globale permet de trouver des similarités plus significatives quel que soit le type de bloc. Toutefois, elle nécessite une quantité de calcul beaucoup plus élevée. Pour diminuer cette dernière, nous avons proposé une méthode originale de classification des blocs de la pyramide. Contrairement aux méthodes de

classification utilisées en compression fractale, notre méthode prend en compte le modèle de similarité pour regrouper les blocs. En se basant sur la notion d'angle, les blocs partageant plus ou moins le même sous-espace de transformation sont regroupés au sein d'une même classe. Cette méthode de classification permet d'accélérer la recherche globale des similarités sans diminuer la qualité des similarités trouvées lorsque beaucoup de blocs sont similaires sur la pyramide.

Concernant la phase de synthèse, nous avons proposé de corriger le résidu de similarité entre les blocs pères en l'ajoutant à la similarité entre les blocs fils. Pour cela, le résidu est interpolé par une transformée B-spline cubique. Nous avons montré par de nombreux résultats expérimentaux que la correction du résidu lors de la synthèse permettait d'améliorer considérablement la qualité globale de l'image agrandie. Par ailleurs, la diminution des supports des blocs fils par rapport à ceux des blocs pères, nous a permis de générer des recouvrements entre ces derniers, et par conséquent, d'améliorer la qualité des transitions entre les blocs fils. Toutefois, le nombre de blocs fils à synthétiser étant plus grand, le temps de calcul augmente. En outre, pour réaliser l'agrandissement des blocs porteurs de basses fréquences, nous avons utilisé la méthode d'interpolation par transformée B-spline cubique.

A partir de résultats expérimentaux, nous avons vérifié la validité de l'hypothèse de préservation des similarités. Soulignons que ceci n'a jamais été effectué dans le cadre du zoom fractal. Les résultats ont montré que l'hypothèse est validée pour la majorité des blocs synthétisés même lorsque la recherche est réalisée globalement. Plus précisément, lorsque le résidu de similarité entre les blocs pères tend vers zéro, le résidu de similarité entre les blocs fils tend également vers zéro. De plus, ce dernier est globalement proportionnel au premier. En conclusion, la probabilité de préservation d'une similarité est suffisamment élevée pour que la synthèse donne la plupart du temps un résultat valable. Finalement, les images agrandies par notre méthode ont été comparées à celles obtenues par un filtrage médian et plus particulièrement à celles d'une transformée B-spline cubique. Les résultats montrent que notre méthode produit des images agrandies plus nettes. De manière quantitative, nous avons montré à partir d'une image de référence que notre méthode donnait une valeur du *PSNR* plus élevée que celle issue de la transformée B-spline cubique.

Pour réaliser un agrandissement de facteur supérieur à deux, nous avons proposé un modèle de synthèse unique et un modèle d'analyse-synthèse. En réitérant la phase d'analyse, le deuxième modèle permet d'obtenir un agrandissement de meilleure qualité mais ceci au détriment d'une quantité de calcul bien plus élevée.

Notre première méthode d'agrandissement a une limite bien établie. En effet, la préservation d'une similarité a un intérêt lorsque le résidu de similarité entre les blocs pères est suffisamment petit. Lorsque ce n'est pas le cas, il est préférable d'agrandir le bloc avec une autre méthode. Pour simplifier, nous avons utilisé une transformée B-spline cubique pour interpoler ce type de bloc, mais d'autres méthodes pourraient être utilisées. En fait, nous pensons que la méthode d'agrandissement idéale n'existe pas et qu'il faut plutôt s'orienter vers des méthodes hybrides. Finalement, notre méthode est intéressante lorsqu'il existe des similarités de bonne qualité sur la pyramide. Leur existence n'étant pas garantie, il n'est pas possible d'être certain

de l'efficacité de la méthode. Toutefois, d'après les divers agrandissements effectués dans le chapitre 7, il semble qu'en pratique la méthode fonctionne sur des images assez variées.

La deuxième méthode d'agrandissement s'intéresse à l'ensemble admissible des images agrandies d'une image initiale. La condition d'admissibilité repose ici sur la notion de réduction : une image appartient à l'ensemble des solutions si sa réduction est identique à l'image initiale. Pour éviter que l'ensemble des solutions soit vide, nous avons redéfini la condition d'admissibilité dans le cas inconsistant pour que des images ayant une réduction proche de l'image initiale soient considérées. Dans tous les cas, nous nous sommes attachés à ce que l'ensemble des solutions soit convexe quels que soient l'image initiale et le filtre utilisé dans l'opération de réduction. Par ailleurs, afin d'obtenir une solution quelconque de cet ensemble, nous avons proposé cinq méthodes de projection inspirées de méthodes plus générales issues de la théorie d'estimation des ensembles convexes [39]. Deux de ces méthodes permettent d'obtenir une projection exacte : *MEPA* et *MEPP*. La première méthode est adaptée lorsque l'ensemble des solutions est consistant, tandis que la seconde est adaptée lorsque celui-ci est inconsistant. Les trois autres méthodes fournissent une projection approchée plus rapidement : *MAPA*, *MAPP* et *MAPPE*. Dans le cas consistant, la méthode *MAPPE* est la plus rapide, tandis que dans le cas inconsistant, seule la méthode *MAPP* peut être employée.

Nous avons exploité cette deuxième méthode pour régulariser vis-à-vis de la contrainte de réduction les images agrandies par notre première méthode d'agrandissement. L'interprétation fréquentielle de l'ensemble des solutions a montré que lorsque le filtre passe-bas est idéal (sinus cardinal), la projection revient à restaurer les basses fréquences de l'image agrandie et à laisser intact ses hautes fréquences. Ceci a été mis en évidence à travers des résultats expérimentaux en utilisant un filtre à petit support (filtre de Burt, $\beta = 0.625$). De manière plus générale, cette régularisation peut être réalisée sur une image agrandie par une méthode de préservation structurelle quelconque et restaurer ainsi l'information perdue lors de l'agrandissement. En outre, elle permet de faire le lien avec les méthodes de préservation fréquentielle.

Nous avons également exploité cette deuxième méthode d'agrandissement pour réaliser la compression réversible et la transmission réversible d'une image. Pour cela, nous l'avons utilisé dans la construction d'une pyramide laplacienne pour remplacer l'opération d'agrandissement classique par filtrage. La pyramide ainsi construite a été baptisée *pyramide résiduelle*. Nous avons montré de manière théorique que lorsque l'image inductrice est l'image agrandie par filtrage et que la méthode utilisée pour la projection est la méthode *MEPA*, alors la pyramide résiduelle a une énergie nécessairement plus petite que celle d'une pyramide laplacienne. En pratique, les résultats ont montré que le gain en compression est significatif. De plus, en mesurant le *PSNR*, nous avons mis en évidence que l'agrandissement par induction permet d'obtenir des images intermédiaires de meilleure qualité lors de la reconstruction de l'image de départ. Ceci est particulièrement intéressant dans un contexte de transmission progressive sur réseau.

Les perspectives de recherche portent à la fois sur des améliorations et sur des applications de nos deux méthodes d'agrandissement.

Une amélioration de notre première méthode d'agrandissement serait de prendre en compte, lors de la synthèse d'un bloc fils, le résidu de similarité entre deux blocs pères par un modèle de déformation des structures. Pour le moment, le résidu est pris en compte en réalisant une interpolation par une transformée B-spline cubique et en ajoutant le résultat lors de la synthèse du bloc. Pour améliorer cette synthèse, on pourrait appliquer un modèle de déformation au bloc similaire après l'utilisation de la transformation massique. Il s'agirait tout d'abord de détecter les structures à la fois dans le bloc de référence et dans le bloc similaire, puis d'appliquer un modèle de déformation des structures du bloc similaire pour que celles-ci soient identiques à celles du bloc de référence. Le même modèle de déformation serait alors appliqué pour synthétiser le bloc fils. Pour simplifier, on pourrait commencer par s'intéresser à un modèle de déformation des contours simples et réaliser un *warping* des textures [31].

Une autre amélioration de notre première méthode concernerait l'agrandissement de facteur supérieur à deux. Pour le moment, nous avons étudié deux approches : synthèse unique et analyse-synthèse. Nous avons noté que la deuxième approche permet d'améliorer la qualité de l'image agrandie en réitérant la phase d'analyse à chaque résolution intermédiaire. Toutefois, nous avons également noté que la complexité algorithmique est beaucoup plus élevée. Pour remédier à ce problème, on pourrait envisager une approche intermédiaire où la première analyse effectuée pour réaliser l'agrandissement de facteur 2 serait exploitée pour accélérer les analyses suivantes. Pour cela, une solution simple serait de rechercher chaque similarité localement à celle trouvée lors de la première analyse.

Une amélioration de notre deuxième méthode d'agrandissement serait d'ajouter des contraintes sur l'ensemble des images agrandies admissibles. Pour le moment, la seule contrainte exploitée est celle de réduction. Une autre contrainte pourrait porter sur la corrélation inter-niveaux que l'on observe sur une représentation espace-échelle obtenue par une transformée en ondelettes [33], [113]. En exploitant les maxima d'ondelettes, on pourrait localement contraindre les coefficients d'ondelettes de l'image agrandie. De même, les minima d'ondelettes pourraient également être exploités pour annuler les coefficients d'ondelettes de l'image agrandie non corrélés avec ceux de l'image initiale. Après avoir caractérisé le nouvel ensemble des solutions, il faudrait redéfinir des méthodes de projection prenant en compte les contraintes sur les coefficients d'ondelettes de l'image agrandie.

Les applications de nos deux méthodes d'agrandissement sont multiples. Tout d'abord, en ce qui concerne la méthode par synthèse de similarités, on pourrait envisager d'appliquer notre méthode de classification des blocs du domaine de recherche dans le contexte d'une compression d'image par fractale. La prise en compte du modèle de similarité lors de la classification devrait permettre d'obtenir une valeur du *PSRN* plus élevée à un taux de compression constant et ceci sans augmenter la quantité de calculs. Ensuite, en ce qui concerne la méthode d'agrandissement par

induction, on pourrait l'exploiter pour réaliser une compression avec perte d'information (non réversible). Pour cela, il faudrait appliquer un modèle de quantification des niveaux de la pyramide résiduelle. Dans ce cas, il est possible que les ensembles $,^k$ associés à chaque niveau deviennent inconsistants. Il faudrait étudier les effets de la quantification sur la qualité de l'image reconstruite par les inductions successives. Pour limiter les effets d'une quantification, on pourrait réaliser une quantification rétroactive [5].

L'application de l'agrandissement par induction à la compression réversible et la transmission progressive d'images pourrait également être améliorée. Une amélioration possible serait de rechercher le filtre optimal permettant d'obtenir une énergie minimale sur les niveaux de la pyramide résiduelle. Nous avons noté dans les résultats expérimentaux que le paramètre β du filtre de Burt permettant d'obtenir le meilleur taux de compression avec la pyramide résiduelle était compris dans l'intervalle $[0.3, 0.35]$. Ceci correspond à un filtre où les poids sont proches d'une distribution gaussienne, ce qui est plutôt éloigné du filtre idéal (sinus cardinal) préconisé par la théorie de l'échantillonnage. L'amélioration du filtre passe par une meilleure compréhension de ce phénomène. Une voie possible serait d'étudier les propriétés de l'ensemble des solutions $,^k$ en fonction de la distribution du filtre. La méthode des ellipsoïdes devrait permettre d'obtenir des informations comme le volume, le diamètre, ou encore l'épaisseur [61], [113], [114]. Enfin, une amélioration du taux de compression pourrait également être obtenue en appliquant un modèle de décorrélation entre les niveaux de la pyramide résiduelle [110].

De manière plus générale, nos deux méthodes d'agrandissement pourraient être exploitées dans le contexte de la vidéo et plus particulièrement dans celui de la télévision haute-définition avec la conversion NTSC-HDTV. Dans ce cas, les méthodes devraient être optimisées pour être ensuite câblées. L'accélération des méthodes passerait notamment par la prise en compte des images agrandies précédentes pour l'agrandissement de l'image courante. En outre, nos deux méthodes peuvent être exploitées dans le cadre de l'agrandissement des images satellites en quinconce. Pour cela, la synthèse des similarités est réalisée à partir d'une pyramide en quinconce où la base est elle-même en quinconce. L'image agrandie d'un facteur $\sqrt{2}$ est ensuite régularisée par la méthode d'agrandissement par induction en exploitant un filtre de type quinconce.

Enfin, nous avons supposé dans le cadre de cette thèse que nous ne disposions d'aucune connaissance sur le système d'acquisition. Toutefois, remarquons que des informations liées à la fonction de réduction pourraient facilement être prises en compte par nos deux méthodes d'agrandissement pour améliorer les résultats. Pour cela, il s'agirait de déterminer les poids du filtre associé à la caméra en utilisant deux images de résolutions différentes. La difficulté se situe surtout au niveau de la calibration de la caméra. Le filtre ainsi obtenu serait utilisé dans la première méthode pour construire la pyramide d'images et dans la deuxième méthode pour régulariser l'image agrandie.

Bibliographie

- [1] J. W. Adams. A subsequence approach to interpolation using the fft. *IEEE Transactions on Circuits and Systems*, 34(5):568–570, May 1987.
- [2] J. I. Agbinya. Fast interpolation algorithm using fast hartley transform. *Proceedings of the IEEE*, 75(4):523–524, April 1987.
- [3] J. I. Agbinya. Interpolation using the discrete cosine transform. *Electronics Letters*, 28(20):1927–1928, September 1992.
- [4] B. Aiazzi, L. Alparone, and S. Baronti. A reduced laplacian pyramid for lossless and progressive image communication. *IEEE Transactions on Communications*, 44(1):18–22, January 1996.
- [5] B. Aiazzi, L. Alparone, S. Baronti, and F. Lotti. Lossless image compression by quantization feedback in a content-driven enhanced laplacian pyramid. *IEEE Transactions on Image Processing*, 6(6):831–843, June 1997.
- [6] I. Aizenberg, N. Aizenberg, J. Astola, and K. Egiazarian. Solution of the super-resolution problem through extrapolation of the orthogonal spectra using multi-valued neural technique. *EUSIPCO*, III:1761–1764, 1998.
- [7] H. M. Al-Otum. Evaluation of reconstruction quality in image vector quantisation using existing and new measures. *IEE Proceedings Vision, Image and Signal Processing*, 145(5):349–356, October 1998.
- [8] A. Albiol and J. Serra. Morphological image enlargements. *Journal of visual communication and image representation*, 8(4):367–383, December 1997.
- [9] V. R. Algazi, G. E. Ford, and R. Potharlanka. Directional interpolation of images based on visual properties and rank order filtering. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:3005–3008, May 1991.
- [10] J. Allebach and P. W. Wong. Edge-directed interpolation. *IEEE International Conference on Image Processing*, III:707–710, September 1996.
- [11] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Transactions on Image Processing*, 1:205–220, 1992.

- [12] E. Atsumi, M. Yoshida, and F. Ono. Unified interpretation of projection-method and linear-interpolation and the inductive introduction of improved reduction and enlargement methods. *SPIE Very High Resolution and Quality Imaging II*, 3025:100–111, February 1997.
- [13] Z. Baharav, D. Malah, and E. Karnin. Hierarchical interpretation of fractal image coding and its applications. *Fractal Image Compression: Theory and Application (Y. Fisher, Ed.)*, New-York: Springer-Verlag, pages 91–117, 1995.
- [14] H. Bakircioglu and E. Gelenbe. Enhanced image enlargement with the random neural network. *Proceedings of the Third Joint Conference on Information Sciences*, March 1997.
- [15] R. H. Bamberger. A method for image interpolation based on a novel multirate filter bank structure and properties of the human visual system. *SPIE Image Processing Algorithms and techniques III*, 1657:351–362, February 1992.
- [16] H. H. Bauschke and J. M. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3):367–426, September 1996.
- [17] A. D. Bayrakeri and R. M. Mersereau. A new method for directional image interpolation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:2383–2386, May 1995.
- [18] A. Biancardi, L. Lombardi, and V. Pacaccio. Improvements to image magnification. *LNCS 1310: International Conference on Image Analysis and Processing*, 1:142–149, September 1997.
- [19] R. D. Boss and E. W. Jacobs. Archetype classification in an iterated transformation image compression algorithm. *Fractal Image Compression: Theory and Application (Y. Fisher, Ed.)*, New-York: Springer-Verlag, pages 79–90, 1995.
- [20] F. Brouaye. Echantillonnage et ondelettes : la méthode du zero-padding revisitée. *Traitement du signal*, 9(2):193–200, 1992.
- [21] P. J. Burt. Fast filter transforms for image processing. *Computer Graphics and Image Processing*, 16:20–51, 1981.
- [22] P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, April 1983.
- [23] D. Calle and A. Montanvert. Pyramide d'images résiduelle par expansion inductive. *Rapport de recherche ENS Lyon N95-36*, Novembre 1995.
- [24] D. Calle and A. Montanvert. Pyramide d'images résiduelle par expansion inductive. *10ième congrès Reconnaissance des Formes et Intelligence Artificielle (RFIA '96)*, 1:278–287, Janvier 1996.

- [25] D. Calle and A. Montanvert. Pyramide résiduelle pour la compression d'images. *16ième Colloque Gretsi sur le traitement du Signal et des Images*, 1:187–190, Septembre 1997.
- [26] D. Calle and A. Montanvert. Augmentation de la résolution spatiale d'une image par des recherches de similarités dans une pyramide. *Actes de CORESA-98, Compression et représentation des signaux audiovisuels*, pages 331–338, Juin 1998.
- [27] D. Calle and A. Montanvert. Augmentation de la résolution spatiale d'une image par induction sur un ensemble de super-résolutions. *Rapport de recherche IMAG RR-997-I*, Mai 1998.
- [28] D. Calle and A. Montanvert. Super-resolution inducing of an image. *IEEE International Conference on Image Processing*, III:232–236, Octobre 1998.
- [29] W. K. Carey, D. B. Chuang, and S. S. Hemami. Regularity-preserving image interpolation. *IEEE International Conference on Image Processing*, I:901–904, October 1997.
- [30] S. Carrato, G. Ramponi, and S. Marsi. A simple edge-sensitive image interpolation filter. *IEEE International Conference on Image Processing*, III:711–714, September 1996.
- [31] K. H. Chan and R. Lau. Contour-based warping. *Graphical Models and Image Processing*, 60:331–348, 1998.
- [32] S. C. Chan, K. L. Ho, and C. W. Kok. Interpolation of 2-d signal by subsequence fft. *IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing*, 40(2):115–118, February 1993.
- [33] S. G. Chang, Z. Cvtkovié, and M. Vetterli. Resolution enhancement of images using wavelet transform extrema extrapolation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:2379–2382, May 1995.
- [34] H. Chen and G. E. Ford. An fir image interpolation filter design method based on properties of human vision. *IEEE International Conference on Image Processing*, III:581–585, November 1994.
- [35] T. C. Chen and R. J. P. De Figueiredo. Image decimation and interpolation techniques based on frequency domain analysis. *IEEE Transactions on Communications*, 32(4):479–484, April 1984.
- [36] G. M. Cheung and B. Zeng. Image interpolation using wiener filters and neural networks. *IEEE Workshop on nonlinear signal and image processing*, June 1995.

- [37] A. Chéhikian. Algorithmes optimaux pour la génération de pyramides d'images passe-bas et laplaciennes. *Traitement du signal*, 9(4):297–307, 1992.
- [38] F. Chin, A. Choi, and Y. Luo. Optimal generating kernels for image pyramids by piecewise fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1191–1198, December 1992.
- [39] P. L. Combettes. The foundations of set theoretic estimation. *Proceedings of the IEEE*, 81(2):182–208, February 1993.
- [40] P. L. Combettes. Signal recovery by best feasible approximation. *IEEE Transactions on Image Processing*, 2(2):269–271, April 1993.
- [41] P. L. Combettes. Inconsistent signal feasibility problems: least-squares solutions in a product space. *IEEE Transactions on Signal Processing*, 42(11):2955–2966, November 1994.
- [42] P. L. Combettes. Convex set theoretic image recovery by extrapolated iterations of parallel subgradient projections. *IEEE Transactions on Image Processing*, 6(4):493–506, April 1997.
- [43] R. E. Crochiere and L. R. Rabiner. Interpolation and decimation of digital signals – a tutorial review. *Proceedings of the IEEE*, 69(3):300–331, March 1981.
- [44] J. L. Crowley and R. M. Stern. Fast computation of the difference of low-pass transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:212–222, 1984.
- [45] A. M. Darwish and M. S. Bedair. An adaptive resampling algorithm for image zooming. *SPIE Image and Video Processing IV*, 2666:131–144, February 1996.
- [46] A. M. Darwish, M. S. Bedair, and S.I. Shaheen. Adaptive resampling algorithm for image zooming. *IEE Proceedings Vision, Image and Signal Processing*, 144(4):207–212, August 1997.
- [47] F. Davoine. Compression d'images par fractales basée sur la triangulation de delaunay. *Thèse, Institut National Polytechnique de Grenoble*, Décembre 1995.
- [48] N. A. Dodgson. Quadratic interpolation for image resampling. *IEEE Transactions on Image Processing*, 6(9):1322–1326, September 1997.
- [49] F. Dudbridge. Least-squares block coding by fractal functions. *Fractal Image Compression: Theory and Application (Y. Fisher, Ed.)*, New-York: Springer-Verlag, pages 229–241, 1995.
- [50] Y. Fisher (editor). Fractal image compression: Theory and application. New-York: Springer-Verlag, 1995.

- [51] Y. Fisher. Fractal image compression with quadtrees. *Fractal Image Compression: Theory and Application (Y. Fisher, Ed.)*, New-York: Springer-Verlag, pages 55–77, 1995.
- [52] G. E. Ford, R. R. Estes, and H. Chen. Space scale analysis for image sampling and interpolation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, III:165–169, 1992.
- [53] D. Fraser. Interpolation by the fft revisited – an experimental investigation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(5):665–675, May 1989.
- [54] E. Fussfeld and Y. Y. Zeevi. Super-resolution estimation of edge images. *International Conference on Pattern Recognition*, I:11–15, October 1994.
- [55] M. Gharavi-Alkhansari, R. DeNardo, Y. Tenda, and T. S. Huang. Resolution enhancement of images using fractal coding. *SPIE Visual Communications and Image Processing*, 3024, 1997.
- [56] G. Goertzel, F. Mintzer, G. R. Thompson, and Y-H. Chen. The design of interpolation filters and their application to image scaling and rotation. *Research Report RC 20175 (53115) - IBM*, April 1986.
- [57] M. Goldberg and L. Wang. Comparative performance of pyramid data structures for progressive image transmission. *IEEE Transactions on Communications*, 39(4):540–548, April 1991.
- [58] R. C. Gonzalez and R. E. Woods. Digital image processing. *Addison Wesley*, 1993.
- [59] H. Greenspan and C. H. Anderson. Image enhancement by non-linear extrapolation in frequency space. *SPIE Image and Video Processing II*, 2182:2–13, February 1994.
- [60] G. J. Grevera and J. K. Udepa. An objective comparison of 3-d image interpolation methods. *IEEE Transactions on Medical Imaging*, 17(4):642–652, August 1998.
- [61] M. Grötschel, L. Lovsz, and A. Schrijver. Geometric algorithms and combinatorial optimization. *Berlin: Springer-Verlag*, 1988.
- [62] D. Götting, A. Ibenthal, and R. Grigat. Fractal image coding and magnification using invariant features. *NATO Advanced Study Institute on Fractal Image Encoding and Analysis*, July 1995.
- [63] L. G. Gubin, B. T. Polyak, and E. V. Raik. The method of projections for finding the common point of convex sets. *U.S.S.R. Computational Math. Math. Physics*, 7(6):1–24, 1967.

- [64] F. Guichard and F. Malgouyres. Total variation based interpolation. *EU-SIPCO*, III:1741–1744, September 1998.
- [65] S. P. Han. A successive projection method. *Mathematical Programming*, 40(1):1–14, January 1988.
- [66] N. Herodotou and A. N. Venetsanopoulos. Colour image interpolation for high resolution acquisition and display devices. *IEEE Transactions on Consumer Electronics*, 41(4):1118–1126, November 1995.
- [67] N. Herodotou, A. N. Venetsanopoulos, and L. Onural. Image interpolation using a simple gibbs random field model. *IEEE International Conference on Image Processing*, I:494–497, October 1995.
- [68] H. S. Hou and H. C. Andrews. Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26:508–517, December 1978.
- [69] D. Houlding and J. Vaisey. Low entropy image pyramids for efficient lossless coding. *IEEE Transactions on Image Processing*, 4(8):1150–1153, August 1995.
- [70] B. Burke Hubbard. Ondes et ondelettes - la saga d'un outil mathématique. *Sciences d'Avenir - Pour la science, Diffusion Belin*, Novembre 1995.
- [71] A. N. Iusem and A. R. De Pierro. On the convergence of han's method for convex programming with quadratic objective. *Mathematical Programming*, 52(2):265–284, August 1991.
- [72] A. E. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Transactions on Image Processing*, 1(1):18–30, January 1992.
- [73] A. E. Jacquin. Fractal image coding: a review. *Proceedings of the IEEE*, 81(10):1451–1465, October 1993.
- [74] K. Jensen and D. Anastassiou. Spatial resolution enhancement of images using nonlinear interpolation. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:2045–2048, April 1990.
- [75] K. Jensen and D. Anastassiou. Subpixel edge localization and the interpolation of still images. *IEEE Transactions on Image Processing*, 4(3):285–295, March 1995.
- [76] J. M. Jolion. Analyse d'images : le modèle pyramidal. *Traitement du signal*, 7(1):5–17, 1990.
- [77] J. M. Jolion and A. Rosenfeld. A pyramid framework for early vision. *Eds. Kluwer*, 1994.

- [78] R. G. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 29:1153–1160, December 1981.
- [79] Kodak. Flashpix format and architecture white paper. <http://www.kodak.com>, June 1996.
- [80] W. G. Kropatsch. A pyramid that grows by powers of 2. *Pattern Recognition Letters*, 3:315–322, September 1985.
- [81] C. J. Kuo, C. Liao, and C. C. Lin. Adaptive interpolation technique for scanning rate conversion. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):317–321, June 1996.
- [82] C. Lee, M. Eden, and M. Unser. High-quality image resizing using oblique projection operators. *IEEE Transactions on Image Processing*, 7(5):679–692, May 1998.
- [83] C. K. Lee and W. K. Lee. Fast fractal image block coding based on local variances. *IEEE Transactions on Image Processing*, 7(6):888–891, June 1998.
- [84] S. Lepsøy and G. E. Øien. Fast fractal image encoding by adaptive codebook clustering. *Fractal Image Compression: Theory and Application (Y. Fisher, Ed.)*, New-York: Springer-Verlag, pages 177–197, 1995.
- [85] J. G. Leu. Image size reduction and enlargement based on circular apertures. *SPIE Visual Communications and Image Processing*, 2501(1):231–242, May 1995.
- [86] T. Lindeberg. Scale space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):234–254, March 1990.
- [87] P. C. Liu, W. T. Chang, and W. Z. Shen. Recursive minimum energy algorithm for image interpolation. *IEEE Transactions on Consumer Electronics*, 44(1):187–191, February 1998.
- [88] M. H. Loew, D. Li, and R. L. Pickholtz. An adaptive pifs model in fractal image compression. *SPIE Medical imaging : Image Display*, 2707:284–293, February 1996.
- [89] E. Maeland. On the comparison of interpolation methods. *IEEE Transactions on Medical Imaging*, 7(3):213–217, September 1988.
- [90] N. De Maistre and C. Labit. Codage hiérarchique d’images pour la transmission progressive : un état de l’art. *Publication interne de l’IRISA N953*, Octobre 1995.

- [91] S. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.
- [92] S. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2):617–643, March 1992.
- [93] D. Marr and H. Hildreth. Theory of edge detection. *Proc. of Royal Soc. London B*, 207:187–219, 1980.
- [94] P. Meer, E. S. Baugher, and A. Rosenfeld. Frequency domain analysis and synthesis of image pyramid generating kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(4):512–522, July 1987.
- [95] Y. Meyer. Principe d’incertitude, bases hilbertiennes et algèbres d’opérateurs. *Séminaire Bourbaki*, 662:28–37, 1986.
- [96] D. M. Monro and P. D. Wakefield. Zooming with implicit fractals. *IEEE International Conference on Image Processing*, I:913–916, 1997.
- [97] D. M. Monro and S. J. Wooley. Fractal image compression without searching. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, V:557–560, 1994.
- [98] B. S. Morse and D. Schwartzwald. Isophote-based interpolation. *IEEE International Conference on Image Processing*, III:227–231, October 1998.
- [99] L. Onural, N. Herodotou, and A. N. Venetsanopoulos. Interpolation by regeneration using gibbs random field models. *IEEE Workshop on nonlinear signal and image processing*, June 1995.
- [100] S. K. Park and R. A. Schowengerdt. Image reconstruction by parametric cubic convolution. *Computer Vision, Graphics, and Image Processing*, 23(3):258–272, September 1983.
- [101] J. A. Parker, R. V. Kenyon, and D. E. Troxel. Comparison of interpolating methods for image resampling. *IEEE Transactions on Medical Imaging*, MI-2(1):31–39, March 1983.
- [102] A. J. Patti, M. I. Sezan, and A. M. Tekalp. Superresolution video reconstruction with arbitrary sampling lattices and nonzero aperture time. *IEEE Transactions on Image Processing*, 6(8):1064–1076, August 1997.
- [103] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, July 1990.
- [104] G. Pierra. Méthodes de projections parallèles extrapolées relatives à une intersection de convexes. *Rapport de recherche INPG*, Septembre 1975.

-
- [105] G. Pierra. Decomposition through formalization in a product space. *Mathematical Programming*, 28:96–115, January 1984.
- [106] E. Polidori and J-L. Dugelay. Zooming using iterated function systems. *FRAC-TALS*, 5:111–123, April 1997.
- [107] J. Prades-Nebot, A. Albiol, and C. Bachiller. Enhanced b-spline interpolation of images. *IEEE International Conference on Image Processing*, III:289–293, October 1998.
- [108] K. P. Prasad and P. Satyanarayana. Fast interpolation algorithm using fft. *Electronics Letters*, 22(4):185–187, February 1986.
- [109] W. H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. Numerical recipes in c: the art of scientific computing. *Cambridge university press, second edition*, 1992.
- [110] G. Qiu. A progressively predictive image pyramid for efficient lossless coding. *IEEE Transactions on Image Processing*, 8(1):109–115, January 1999.
- [111] G. Ramponi. Warped distance for space-variant linear image interpolation. *IEEE Transactions on Image Processing*, 8(5):629–639, May 1999.
- [112] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [113] L. Saas. Agrandissement d’image par induction : évaluation d’ensemble par la méthode des ellipsoïdes et induction par ondelettes. *Rapport du DEA de Mathématiques Appliquées de l’UJF*, Juin 1999.
- [114] A. Sabharwal and L. Potter. Set estimation via ellipsoidal approximations. *IEEE Transactions on Signal Processing*, 45(12):3107–3112, December 1997.
- [115] A. Said and W.A. Pearlman. A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3):243–250, June 1996.
- [116] P. Saint-Marc, J. Chen, and G. Medioni. Adaptive smoothing: a general tool for early vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):514–529, June 1991.
- [117] P. V. Sankar and L. A. Ferrari. Simple algorithms and architectures for b-spline interpolation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):271–276, March 1988.
- [118] P. Sathyanarayana, P. S. Reddy, and M. N. Swamy. Interpolation of 2-d signals. *IEEE Transactions on Circuits and Systems*, 37(5):623–625, May 1990.

- [119] D. Saupe. Accelerating fractal image compression by multi-dimensional nearest neighbor search. *Proceedings DCC'95 Data Compression Conference*, J. A. Storer, M. Cohn (eds.), IEEE Computer Society Press, March 1995.
- [120] D. Saupe, R. Hamzaoui, and H. Hartenstein. Fractal image compression - an introductory overview. *Fractal Models for Image Synthesis, Compression, and Analysis*, D. Saupe, J. Hart (eds.), ACM SIGGRAPH'96 Course Notes 27, August 1996.
- [121] R. W. Schafer and L. R. Rabiner. A digital signal processing approach to interpolation. *Proceedings of the IEEE*, 61(6):692–702, June 1973.
- [122] D. Schott. Basic properties of fejer monotone sequences. *Rostock. Math. Kolloq*, 49(12):57–74, 1995.
- [123] R. R. Schultz and R. L. Stevenson. Improved definition image expansion. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, III:173–176, 1992.
- [124] R. R. Schultz and R. L. Stevenson. A bayesian approach to image expansion for improved definition. *IEEE Transactions on Image Processing*, 3(3):233–242, May 1994.
- [125] R. R. Schultz and R. L. Stevenson. Extraction of high-resolution frames from video sequences. *IEEE Transactions on Image Processing*, 5(6):996–1011, June 1996.
- [126] C. E. Shannon. A mathematical theory of communication. *The Bell System Tech. J.*, 27:379–423, 623–656, July and October 1948.
- [127] C. E. Shannon. Communication in the presence of noise. *Proc. Inst. Radio Eng.*, 37(1):10–21, January 1949.
- [128] B. Simon, J. Y. Mertès, P. Ciblat, and B. Macq. Local interpolation in multi-resolution decomposition of images. *IEEE International Conference on Image Processing*, I:287–290, September 1996.
- [129] T. Smit, M. R. Smith, and S. T. Nichols. Efficient sinc function interpolation technique for center padded data. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(9):1512–, September 1990.
- [130] J. Stokes. Resampling satellite scenes. *SPIE*, 2357:776–783, August 1994.
- [131] S. Thurnhofer, M. Lightstone, and S. K. Mitra. Adaptive interpolation of images with application to interlaced-to-progressive conversion. *SPIE Visual Communications and Image Processing*, 2094:614–625, November 1993.
- [132] S. Thurnhofer and S. K. Mitra. Edge-enhanced image zooming. *Optical Engineering*, 35(7):1863–1870, July 1996.

- [133] H-C. Ting and H-M. Hang. Spatially adaptive interpolation of digital images using fuzzy inference. *SPIE*, 2727:1206–1217, 1996.
- [134] M. J. Tsai, J. Villasenor, and F. Chen. Stack-run image coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 6:519–521, October 1996.
- [135] M. Unser. An improved least squares laplacian pyramid for image compression. *Signal Processing*, 27(2):187–203, May 1992.
- [136] M. Unser, A. Aldroubi, and M. Eden. Fast b-spline transforms for continuous image representation and interpolation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):277–285, March 1991.
- [137] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part i – theory. *IEEE Transactions on Signal Processing*, 41(2):821–833, February 1993.
- [138] M. Unser, A. Aldroubi, and M. Eden. B-spline signal processing: Part ii – efficient design and applications. *IEEE Transactions on Signal Processing*, 41(2):834–848, February 1993.
- [139] M. Unser, A. Aldroubi, and M. Eden. The 12 polynomial spline pyramid. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):364–379, April 1993.
- [140] M. Unser, A. Aldroubi, and M. Eden. Enlargement or reduction of digital images with minimum loss of information. *IEEE Transactions on Image Processing*, 4(3):247–258, March 1995.
- [141] A. van der Schaaf and J. H. van Hateren. Modelling the power spectra of natural images: statistics and information. *Vision Research*, 36(17):2759–2770, 1996.
- [142] Y. Wang and S. K. Mitra. Edge preserved image zooming. *Proceedings European Signal Processing Conference (EURASIP)*, IV:1445–1448, 1988.
- [143] Y. Wang and S. K. Mitra. Motion/pattern adaptive interpolation of interlaced video sequences. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4:2829–2832, May 1991.
- [144] Y. Wang and S. K. Mitra. Image representation using block pattern models and its image processing applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):321–336, April 1993.
- [145] Z. Wang. Interpolation using type i discrete cosine transform. *Electronics letters*, 26(15):1170–1172, July 1990.

-
- [146] A. B. Watson. Ideal shrinking and expansion of discrete sequences. *Technical Memorandum 88202 - NASA*, January 1986.
- [147] K. Xue, A. Winans, and E. Walowit. An edge-restricted spatial interpolation algorithm. *Journal of Electronic Imaging*, 1(2):152–161, April 1992.
- [148] L. P. Yaroslavsky. Efficient algorithm for discrete sinc interpolation. *Applied optics*, 36(2):460–463, January 1997.
- [149] B. Zeng and A. N. Venetsanopoulos. Image interpolation based on median-type filters. *Optical Engineering*, 37(9):2472–2482, September 1998.
- [150] Y. Zhao and B. Yuan. A new affine transformation: its theory and application to image coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(3):269–274, June 1998.