Table des matières

Table des matières

Table des figures

Table des figures

Introduction générales

Introduction générale

L'image numérique est omniprésente dans notre quotidien, les web-cames, les appareils photo, les caméscopes numériques et les scanners, tous ces dispositifs électroniques désormais usuels permettent de produire massivement de telles images. Au-delà de ces utilisations domestiques, la médecine, l'astronomie, la météorologie, etc ; sont aussi de grands générateurs et consommateurs d'images numériques. Le passage de la scène réelle à la forme digitale possède l'avantage de transporter la chaine de traitement de l'image vers le monde de l'informatique, qui est simple, puissant et facile d'accès. Cela a permet d'envisager plusieurs opérations sur l'image comme le zoom, la réduction, la rotation, la détection des contours, etc. L'amélioration de la qualité de l'image fait partie des opérations qu'on peut appliquer sur l'image numérique, comme le deflouage, le débruitage pour les images qui subissent des dégradations au moment de la prise de la scène, ou lors du processus de numérisation ou de transmission.

Parmi aussi, les dégradations que peut subir une image, lors d'un traitement numérique, il y a la perte totale de l'information dans une zone de l'image. Dans cette thèse on s'intéresse au problème de la reconstructions de données manquantes dans une image, Ce processus, couramment nommé :"Inpainting", considéré comme un problème inverse, où partant d'une observation I_0 qui représente l'image dégradée, nous essayons de trouver l'image originale I, avec $I_0 = A(I)$ où A est l'opérateur qui provoque la dégradation. L'inpainting s'étend aussi à l'opération de supprimer des parties indésirables dans l'image : des écritures, des logos, des taches sur des photographies, pour des effets spéciaux [8, 57] sur l'image, on peut enlever des parties dans la scène elle même. La correction des documents anciens [13, 15], la réparation des rayures de microfilms [51]. L'inpainting nécessite une interpolation spatiale intelligente des pixels connus de l'image sur la zone abimée. Cette restauration de la partie détériorée doit être faite d'une façon plausible.

Dans la littérature, plusieurs approches ont été proposées pour réaliser l'opération d'inpainting sur les images numériques voir [8, 57, 51, 16]. Il s'avère qu'une grande partie de ces approches et techniques nécessite l'utilisation d'équations aux dérivées partielles (**EDP**). Toutefois, les limites de chacune de ces **EDP** furent très rapidement atteintes à cause des effets d'oscillations, et de la dégradation des résultats observés prés des discontinuités, en l'occurrence, les contours des objets formant l'image. Le problème est de proposer une nouvelle **EDP** assurant cette opération de remplissage avec un contenu en harmonie avec le reste de l'image, tout en préservant le plus possible les continuités dans l'image, et obtenir, non pas l'image originale, mais une image visuellement très proche de l'image originale.

L'objectif de cette thèse est de développer de nouveaux modèles et de nouveaux algorithmes qui permettront une avancée significative, et qui améliorent les résultats d'inpainting pour restaurer de façon plausible des zones manquantes ou détériorées d'une image. Notre approche s'inspire de la mécanique des fluides pour générer un algorithme qui détermine, de la manière la plus automatique possible, l'intensité de pixels considérés comme manquants dans l' image, en propageant les lignes de niveaux continument à l'intérieur de la zone d'inpainting, sans que l'opérateur intervienne.

Comme nous le savons tous, les équations de Navier-Stokes dans $\mathbb{R}^2$, pour un fluide newtonien incompressible, contrôle et modélise le déplacement des substances fluides (Écoulements, tourbillons, turbulences ...). Dans cette thèse on élabore une analogie entre la mécanique des fluides et le traitement d'image, en s'inspirant principalement du travaux de Bertalmio et al dans [14, 13] pour transporter l'information de l'extérieur vers l'intérieur de la zone d'inpainting.

Cette thèse comporte trois chapitre :

Dans le chapitre 1, intitulé Image numérique et traitement, on initie le lecteur au monde de l'image numérique, par définir ce qu'est une image numérique, et en donnant ses principales propriétés. Dans le même chapitre nous introduisons les différents modèles de traitement d'image qu'on peut rencontrer dans la littérature, ces différents modèles sont exposés selon les outils mathématiques qui sont utilisés. L'application de ces différents modèles est illustrée par des exemples.

Dans le chapitre 2. on aborde les principales propriétés mathématiques d'une image, en particulier celles qui sont importantes dans le choix du modèle approprié pour réaliser une meilleure restauration de l'image dégradée, en l'occurrence le gradient d'une image, la courbure d'une ligne de niveaux et d'autres notions importantes. Des notions mathématiques qui sont indispensables pour le développement théorique de notre modèle sont aussi rappelées dans ce chapitre.

La méthode que nous proposons pour l'inpainting est présentée dans le chapitre 3. On commence par définir rigoureusement l'opération de l'inpainting comme exemple de problème inverse dans le traitement d'images et on donne quelques exemples sur le champ d'application de cette opération. Par la suite, les principales approches pour l'inpainting sont présentées, suivies d'une discussions sur les inconvénients et les limites de chaque approche. Notre modèle ainsi que les différentes étapes de l'algorithme seront présentés, et consolidés, d'une part, par une étude théorique sur l'existence de la solution pour notre modèle ainsi que la stabilité des schémas numériques développés, et d'autre part, par des applications sur des images synthétiques et réelles. Les résultats obtenus sont comparés à d'autres méthodes et la performance de notre modèle est illustrée en faisant usage de certains critères classiques dans le traitement d'image, en l'occurrence le PSNR (Peak Signal-to-Noise Ratio).

Une conclusion sur notre approche est donnée à la fin, dans laquelle on rappelle les performances de notre modèle, et on mentionne ses quelques inconvénients, qui feront l'objet des futurs travaux, comme l'extension de notre modèle pour les images texturées.

Chapitre 1

Chapitre 1

Image numérique et traitement

1.1 Image numérique

Avant de définir la problématique de notre étude, nous allons rappeler brièvement ce que l'on désigne par image numérique, comment on peut l'obtenir et nous présenterons les transformations dictées par le souci d'adopter des traitements ultérieurs comme le filtrage, la restauration, le rehaussement , l'inpainting ...

1.1.1 De l'analogique vers le numérique

Les images numériques sont des clichés électroniques d'une scène réelle ou de documents tels que photographies, manuscrits, textes imprimés et œuvres d'art. Ces clichés sont obtenus, lors de la phase d'acquisition, par des capteurs appelés spécifiquement CCD (Charge Couple Device). L'image numérique une fois acquise, subit un échantillonnage [1]. Cette tache est l'une des toutes premières étapes de traitement numérique des images qui réduit l'ensemble continu du monde réel observable en une série de valeurs discrètes. Cette étape s'accompagne d'une quantification des niveaux de gris et débouche ensuite

1. En traitement d'images, l'échantillonnage apparait en de nombreuses autres occasions que lors de l'acquisition. En effet, les images sont très souvent appelées à être échantillonnées, par exemple pour des conversions de formats ou pour des transformations de géométrie, voir [5].

sur une grille de points ou éléments d'images (en anglais : picture element ou pixel), voir figure 1.1. A chaque pixel correspond une valeur tonale (noir, blanc, niveaux de gris ou couleur).

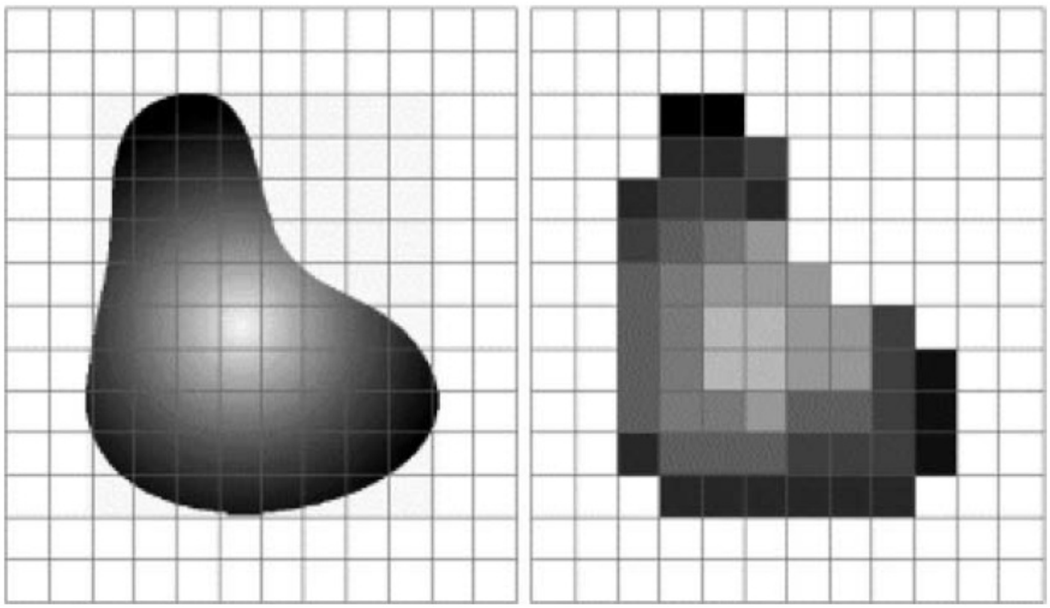


FIGURE 1.1 – Numérisation d'une scène réelle, par échantillonnage et quantification.

Mathématiquement, toute image numérique n'est qu'un tableau de pixels, voir figure 1.2, c'est-à-dire une matrice I, où chaque pixel est décrit par :

- ses coordonnées dans l'image (i, j).
- sa valeur $I(i, j)$.

La taille d'un pixel définit la résolution de l'image numérique par rapport à l'image analogique, c'est-à-dire la finesse de la grille. Plus le nombre de pixels dans l'image diminue, plus la résolution baisse, et plus la qualité de l'image numérique se dégrade. La résolution est, donc la capacité à distinguer les détails fins dans l'image. La fréquence spatiale d'échantillonnage est le meilleur indicateur de la résolution. C'est pourquoi les termes points par pouce ou pixels par pouce (dpi et ppi en anglais) sont les expressions courantes et synonymes indiquant la résolution des images numériques, voir figure 1.3. On appelle définition de l'image le nombre de points (pixels) constituant une image, c'est le

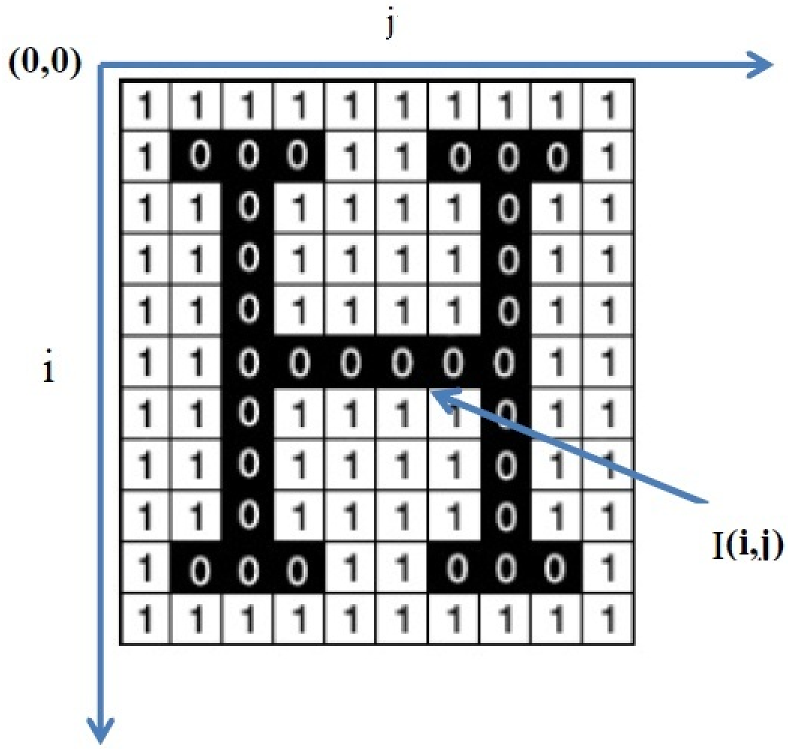


FIGURE 1.2 – L'image et les pixels : dans cette image binaire, chaque pixel se voit assigné une valeur tonale, dans ce cas 0 pour le noir et 1 pour le blanc.

nombre de colonnes de l'image multiplie par le nombre de lignes. Une image possédant 10 colonnes et 11 lignes aura une définition de $10 * 11$. En pratique, c'est le rapport entre la définition et la résolution de l'image qui va déterminer sa qualité.

1.1.2 Propriétés d'image numérique

Après avoir défini l'image numérique, on propose dans cette sous section ces quelques propriétés, du point de vue qualité, contenu et représentation :

Contenu de l'image

Parmis les plus importants composants définissant une image, voir figure 1.4 on peut citer :

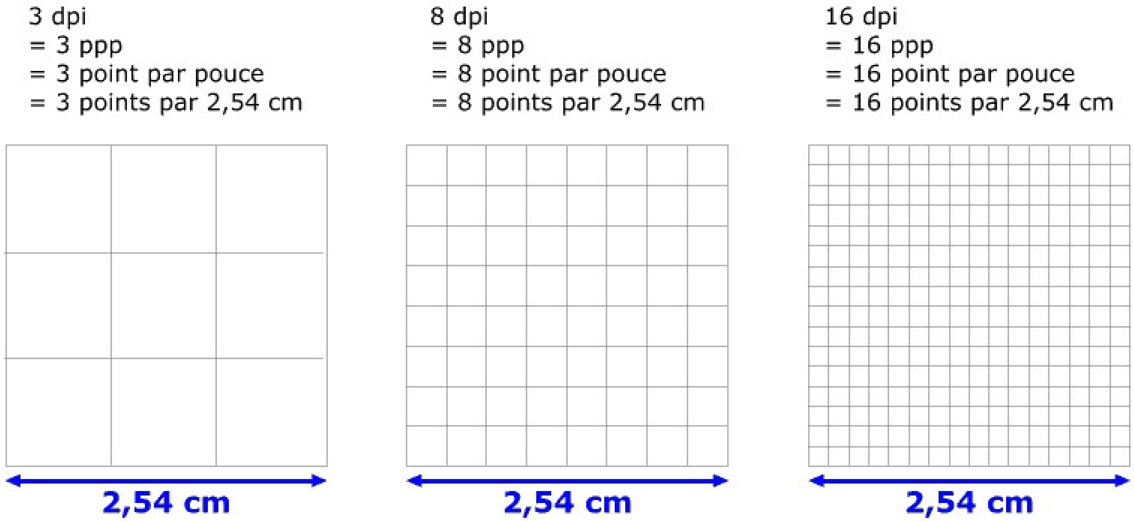


FIGURE 1.3 – Résolution d'une image : image avec des résolutions différentes,1pouce=2.54 cm.

- **Texture** : la répartition statique ou géométrique des intensités dans l'image.
- **Contour** : limite entre deux pixels (ou groupe de pixels) dont la différence de niveau de gris (ou couleur) est significative,
- **Région** : groupe de pixels présentant des caractéristiques similaires (intensité, mouvement, etc.),
- **Objet** : région (ou groupe de région) entièrement délimitée par un contour, possèdant une indépendance dans l'image.

Qualité d'image

Parmi les qualités d'une image on peut citer :

- **Contraste** : qualité de la dynamique des intensités de l'image,
- **Brillance** : moyenne des niveaux de gris de l'image, ou intensité moyenne,
- **Bruit** : signal parasite dont la distribution dans l'image est aléatoire,
- **Déformations géométriques** : défauts dus à la différence d'axe entre le capteur d'acquisition et le centre de la scène observée.

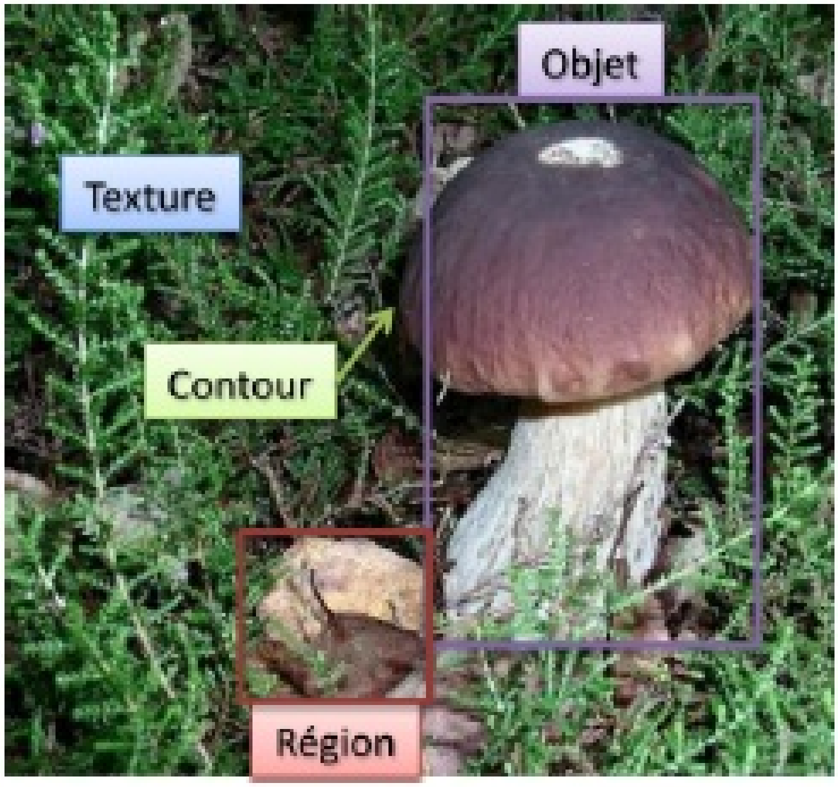


FIGURE 1.4 – Les contenus d'une image.

Représentation d'image

Une image numérique peut être représenté numériquement sous forme :

- **Matricielle** : les images sous cette forme sont les images que nous utilisons généralement pour restituer des photos numériques, on l'appelle aussi une image en mode point ; elles reposent sur une grille de plusieurs pixels formant une image avec une définition bien précise, lorsqu'on les agrandi trop, on perd de la qualité de la dynamique des intensités de l'image.
- **Vectorielle** : ce sont des images dont la particularité est que chaque forme qui la compose est décrite mathématiquement à partir de points et de tangentes. Elles ne peuvent pas décrire une image trop complexe comme une photographie, mais sont tout à fait adaptées au rendu typographique, aux logos et autre formes composées de tracés simple. voir figure 1.5

FIGURE 1.5 – Image matricielle et image vectoriel.

1.1.3 Évaluation de la qualité d'image numérique

En traitement d'images, il est important de pouvoir décrire précisément les dégradations liées à la compression, au transfert et à la reconstruction des images. Des mesures ont été développées pour remplir cette tache.

D'autre part, en traitement d'images, on se soucie plus de la fidélité de l'image reconstruite après traitement à l'originale, que de la qualité perceptuelle. On propose quelques mesures qui quantifie cet aspect :

MSE(Mean squard error)

Le **MSE** mesure la moyenne des carrés des erreurs entre une image référence (l'image d'origine) et une image dégradée (ou l'image reconstruite après traitement). Il est calculé pixel par pixel en ajoutant les carrées des différences de valeur entre pixels et en divisant le résultat par le nombre total de pixel :

$$\mathbf{MSE} = \frac{\sum_{1 \leq m \leq M, 1 \leq n \leq N} [I_1(m,n) - I_2(m,n)]^2}{M \times N},$$

avec $M \times N$: la taille de l'image, I_1 et I_2 : l'image d'origine et l'image reconstituée.

Un **MSE** nul indique une parfaite identité entre l'image d'origine et l'image reconstituée, c'est à dire aucune perte d'information.

PSNR(Peak signal to noise ratio)

Le **PSNR** exprime le rapport entre la puissance d'un signal et celle du bruit affectant la fidélité de sa représentation. Il est utilisé pour exprimer la qualité de reconstruction d'un algorithme de traitement d'image avec perte, sa valeur étant indéfinie lorsque les deux images comparées sont identiques. Il est obtenu à travers cette formule :

$$\mathbf{PSNR} = 10 log_{10} \left(\frac{R^2}{MSE} \right),$$

avec R la profondeur de l'image qui représente la différence entre la plus grande et la plus petite valeur du niveau de gris.

Son avantage par rapport au **MSE** est d'intégrer dans son calcul la profondeur R de l'image.

SSIM(Structural similarity)

Le **SSIM** indice de similarité structurelle, permet de mesurer la similarité entre deux images d'une manière plus proche à la perception subjective humaine. Il se base sur le constat que la vision humaine est plus adaptés à l'analyse d'information structurelle et a donc pour vocation de mesurer efficacement les altérations de cette information entre l'image source et l'image reconstituée, voir [21].

Il est définit par trois termes : un terme de luminance, un terme de contraste et un terme de structure :

$$SSIM = l^{\alpha} c^{\beta} s^{\gamma},$$

avec α, β et γ trois paramètres qui définissent l'importance de chaque terme : luminance l, contraste c et structure s, voir [22] pour plus de détails. La figure 1.6 donne un exemple d'application sur une image en niveau de gris.

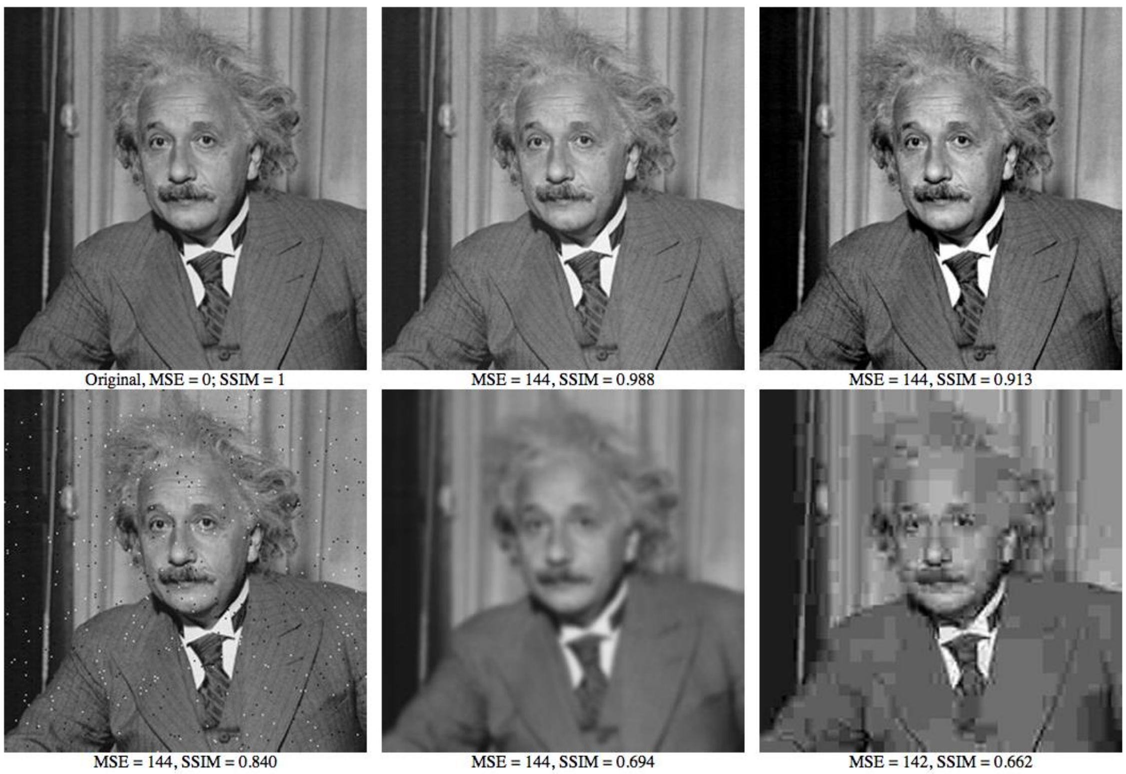


FIGURE 1.6 – Évaluation de la qualité d'image numérique

1.2 Le traitement d'images numériques et ses outils

Un traitement d'image est nécessaire, par exemple, pour améliorer la perception de l'information contenue dans l'image, pour détecter des objets, pour réaliser des comparaisons d'informations, pour extraire de l'image des indicateurs.

Les outils de traitement d'image les plus courants peuvent être organisés selon les principaux modèles mathématiques qui ont été employés pour traiter cette image. Aux différents modèles présentés correspond un certain nombre d'outils fondamentaux et incontournables que ce soit d'un point de vue pratique ou théorique. Citons : la convolution, la transformée de Fourier, l'histogramme, la corrélation, les ondelette, etc.

Dans la suite de cette section, on présente quelques exemples de modèles utilisés dans le traitement d'image ; on associe à chaque modèle proposé les outils couramment utilisés. Notons que les différents modèles proposés ne sont ni exclusifs ni cloisonnés, pour plus de détails on renvoie le lecteur vers [16, 46] :

1.2.1 Modèle linéaire

Dans un modèle linéaire, les opérateurs mathématiques utilisés sont ceux qui préservent la structure d'espace vectoriel, c'est-à-dire que si on considère deux images numériques I et J, et si D désigne l'opérateur utilisé, alors :

$$D(\alpha I + \beta J) = \alpha D(I) + \beta D(J).$$

Parmi ces opérateurs, le plus fondamental est l'opérateur de convolution(voir [43]).

Convolution

Pour réduire un bruit b dans une image, on peut avoir recours aux opérations de filtrage, qu'on peut réaliser en faisant une convolution entre l'image dégradée I et un filtre D, appelé aussi masque de convolution. La convolution est une opération de voisinage qui effectue une combinaison linéaire de pixels d'image I, produisant une nouvelle image I'.

Mathématiquement, la convolution d'une image I de taille $N * M$ avec un filtre D est donné par :

$$I'(i,j) = (I * D)(i,j) = \sum_{n=-\frac{d-1}{2}}^{\frac{d-1}{2}} \sum_{m=-\frac{d-1}{2}}^{\frac{d-1}{2}} I(i,j)D(n-i, m-j),$$

D est choisi comme un masque de taille d. Pour conserver la moyenne des pixels dans l'image originale, on normalise les coefficients du filtre. Quand le masque recouvre des zones en dehors de l'image, on considère que l'image est entourée de noir, c'est-à-dire de pixels avec une intensité nulle.

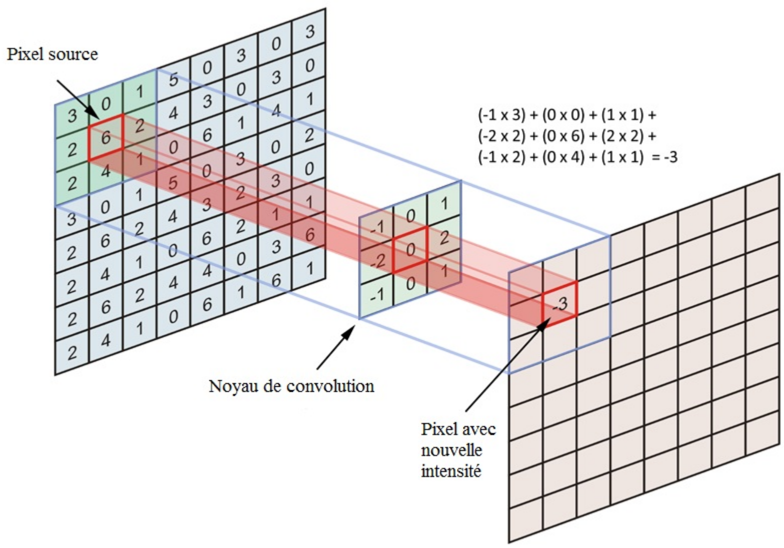


FIGURE 1.7 – Convolution de l'image avec un noyau.

1.2.2 Modèle fréquentiel

Le modèle fréquentiel tend à décrire l'image I en terme de structure périodique. En la décomposant dans une base de fonctions périodiques simples, comme des sinusoïdes, ce qui fournit une nouvelle représentation de l'image dans l'espace des fréquences ; l'un des outils le plus utilisé pour réaliser ce transfert est la transformée de Fourier discrète.

Transformée de Fourier discrète (TFD)

Pour une image de taille $N \times M$ la TFD est donnée par :

$$I(u,v) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} I(n,m) e^{-2\pi i (\frac{un}{N} + \frac{vm}{M})},$$

avec : $-\frac{N}{2} \leq u \leq \frac{N}{2} - 1$ et $-\frac{M}{2} \leq v \leq \frac{M}{2} - 1$.

Une fois le spectre obtenu, on considère ou on supprime les fréquences indésirables à l'aide d'un filtre adéquat. On peut distinguer trois familles de filtres :

- **Filtre passe-bas** : est un filtre qui laisse passer les bases fréquences et qui atténue les hautes fréquences,
- **Filtre passe-haut** : est un filtre qui laisse passer les hautes fréquences et qui atténue les basses fréquences,
- **Filtre passe bande** : appelé aussi filtre éjecteur de bande, c'est un filtre qui empêche le passage d'un intervalle de fréquences.

FIGURE 1.8 – Application du filtre à une image, on a appliqué respectivement un filtre passe bas, un filtre passe haut et un filtre passe bande sur l'image originale à droite.

Une fois l'annulation des fréquences faite, l'image traitée est reconstruite à l'aide de la **TFD** inverse (**TFDI**) donnée par :

$$I(n,m) = \frac{1}{N \times M} \sum_{u=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{v=-\frac{M}{2}}^{\frac{M}{2}-1} I(u,v) e^{2\pi i(\frac{un}{N}+\frac{vm}{M})},$$

avec : $0 \leq n \leq N-1$ et $0 \leq m \leq M-1$.

Pour réaliser des traitements dans l'espace fréquentiel des images, on peut aussi avoir recours aux transformées en z, transformées en ondelette,etc. (Voir[66]).

1.2.3 Modèle statistique

Dans ce genre de modèle, on s'intéresse aux propriétés statistiques des images comme la distribution des valeurs prises par les pixels, la corrélation existante entre des pixels voisins, la fréquence d'occurrence de certaines structures spatiales. Ces statistiques fournissent des grandeurs et des fonctions sur lesquelles peuvent s'appuyer des modèles probabilistes utilisés pour le traitement d'image.

1.2.4 Modèle différentiel

Dans les modèles différentiels, on considère l'image comme une fonction continue I ; on étudie son comportement local à l'aide de ses dérivées. Une telle étude impose certaines régularités [2] sur la fonction I ; les modèles différentiels permettent de mettre en évidence certaines variations spatiales de l'image. Ils sont utilisés comme traitements de base dans de nombreuses opérations comme le rehaussement de contraste, la détection de contours, le zoom sur une partie de l'image, etc.

Un exemple classique du modèle différentiel est l'équation de la chaleur, pour une fonction I de $\Omega \subset \mathbb{R}^2 \times \mathbb{R}^+$ dans $\mathbb{R}$:

$$\begin{cases} \frac{\partial I}{\partial t} = \Delta I \\ I(x, y, 0) = I_0(x, y) \end{cases} \tag{1.1}$$

I_0 est la donnée initiale sur I, tout en complétant le système (1.1) par des conditions aux bords de Ω. La solution de cette équation est une fonction de $\Omega \times \mathbb{R}^+$ dans $\mathbb{R}$, qui décrit la propagation au cours du temps t de la chaleur dans un milieu isotrope. Pour le traitements d'images, partant d'une image I_0, en résolvant cette équation, on obtient une série d'images $I(., t)$ traitée à l'échelle t, (voir figure 1.9). L'EDP de la chaleur permet une diffusion isotrope. Cette diffusion s'opère ainsi de manière identique dans toutes les directions sans avoir aucune direction privilégiée. Pour des taches de restauration d'images bruitées, ceci présente des inconvénients. En effet dans des zones avec intensité homogène,

2. Ce n'est pas toujours le cas, car une image noir et blanc est discontinue aux contours par exemple.

(a) Original (b) Image bruitée (c) Image filtrée

FIGURE 1.9 – Filtrage d'une image par l'**EDP** de la chaleur.

ce processus permettra de diminuer effectivement l'effet du bruit, mais dans les zones présentant des discontinuités au niveau de l'intensité en niveau de gris, celles-ci seront aussi lissées et le contraste des objets formant ces zones sera sensiblement réduit. Pour traiter ce problème, des idées de diffusion anisotrope ont été proposées. Pour plus de détails sur ces idées voir [30, 29].

Chapitre 2

Chapitre 2

Outils mathématiques

2.1 Structure locale d'images

Dans cette section, on considère une image I de Ω dans $\mathbb{R}$, avec Ω un borné de $\mathbb{R}^2$; on suppose que $I \in C^2(\overline{\Omega})$ et nous nous plaçons en un point (x, y) de Ω.

2.1.1 Ligne de niveau

En ce point (x, y) passe la ligne de niveau associée au paramètre $c = I(x, y)$, c'est-à-dire la frontière de $\chi_c(I)$, courbe deux fois différentiable. Le long de cette ligne, la fonction I est constante et égale à c. En bref la ligne de niveau associée au paramètre c est l'ensemble donné par :

$$\{(x, y) \in \Omega, I(x, y) = c\}.$$

2.1.2 Gradient

Le gradient d'une image mesure la façon dont l'image varie. Il contient deux types d'information. Le module du gradient qui nous informe sur la vitesse de variation de l'image, tandis que la direction du gradient nous indique en quelle direction le changement s'effectue. Comme le gradient a un module et une direction, il est naturel de l'écrire de la façon suivante :

$$\nabla I(x,y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right).$$

Un premier résultat est que le gradient de I en (x,y), c'est-à-dire le vecteur $\nabla I(x,y)$, est perpendiculaire à la ligne de niveau passant par le point (x,y), ce qui confirme que l'intensité de l'image ne change pas le long d'une ligne de niveau.

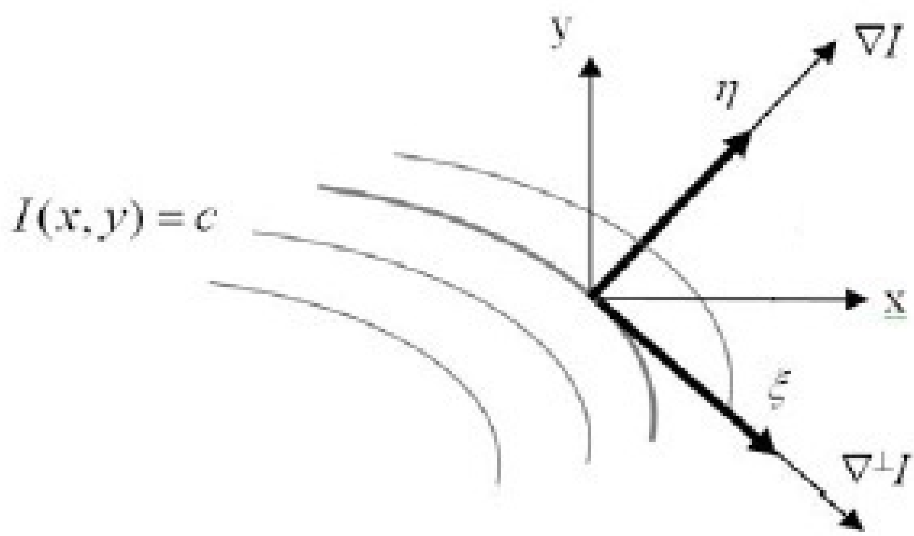


FIGURE 2.1 – Illustration de la structure locale d'une image

Si on suppose de plus que $|\nabla I(x,y)| \neq 0$, on peut définir un système de coordonnées locales en ce point (voir la figure 2.1) :

$$\eta = \frac{\nabla I}{|\nabla I(x,y)|} \quad \text{et} \quad \xi = \frac{\nabla^{\perp} I}{|\nabla I(x,y)|},$$

avec : $\nabla^{\perp} I = (-\frac{\partial I}{\partial y}, \frac{\partial I}{\partial x})$.

2.1.3 Matrice Hessienne

Nous définissons la matrice Hessienne d'une image I par :

$$HI = \begin{pmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial y^2} \end{pmatrix}.$$

et nous donnons également la forme quadratique associée à HI, c'est-à-dire que pour deux vecteurs X et Y, nous avons :

$$HI(X,Y) = X^t HI\,Y,$$

où X^t est le transposé de X.

Pour calculer les dérivées secondes d'une image I selon les deux directions η et ξ, précédemment définies, on utilise la matrice hessienne :

$$I_{\eta\eta} = HI(\frac{\nabla I}{|\nabla I|}, \frac{\nabla I}{|\nabla I|}) \quad \text{et} \quad I_{\xi\xi} = HI(\frac{\nabla I^{\perp}}{|\nabla I|}, \frac{\nabla I^{\perp}}{|\nabla I|}),$$

pour la démonstration voir [33].

La dérivée seconde $I_{\eta\eta}$ de I dans la direction du gradient η localise d'une manière précise les contours, lorsque $I_{\eta\eta} = 0$. La dérivée seconde $I_{\xi\xi}$ de I dans la direction des tangentes ξ au ligne de niveau mesure la courbure d'iso-intensité. On obtient une valeur nulle de $I_{\xi\xi}$ pour les lignes de niveaux droites, et des grandes valeurs pour les lignes de niveaux ayant une forme concave ou convexe.

2.1.4 Courbure

En supposant que $\nabla I \neq 0$, on note souvent par κ la courbure d'une ligne de niveau donnée par :

$$\kappa(x,y) = \frac{1}{|\nabla I|^3} HI(\nabla I^{\perp}, \nabla I^{\perp})(x,y).$$

Ceci est équivalent à :

$$\kappa(x,y) = div(\frac{\nabla I}{|\nabla I|}) = \frac{I_{\xi\xi}}{|\nabla I|}(x,y),$$

où *div* désigne l'opérateur défini par :

$$div(A) = \frac{\partial A_1}{\partial x} + \frac{\partial A_2}{\partial y},$$

où A est une fonction vectorielle : $A(x,y) = (A_1(x,y), A_2(x,y))$. On peut montrer que $\kappa(x,y)$ est la courbure de la ligne de niveau passant par (x,y), c'est-à-dire l'inverse du rayon du cercle osculateur à la courbe $\{I(x,y) = c\}$ au point (x,y). Pour une présentation détaillé sur la structure locale de l'image, voir [46].

2.2 Espaces fonctionnels

Produit scalaire

Un produit scalaire sur un espace vectoriel réel V est la fonction :

$$(.,.) : V \times V \to \mathbb{R},$$

tel que $\forall x, y \quad \text{et} \quad z \in V$ et pour tout $\lambda \in \mathbb{R}$ cette application vérifie :

- $(x, y) = (y, x)$,
- $(x, y + z) = (x, y) + (x, z)$,
- $\lambda(x, y) = (\lambda x, y)$,
- $(x, x) > 0 \quad \text{si} \quad x \neq 0$.

Un espace pré-hilbertien réel est la donnée $(V; (.,.))$ où V est un espace vectoriel réel et $(.,.)$ est un produit scalaire défini sur V.

Si $(.,.)$ est un produit scalaire, la norme associée à ce produit scalaire est donné par :

$$|x| = (x, x)^{\frac{1}{2}}.$$

La métrique associée à ce produit scalaire est donné par :

$$d(x, y) = |x - y| = (x - y, x - y)^{\frac{1}{2}}.$$

2.2.1 Espace de Hilbert

Un espace de Hilbert réel, est un espace pré-hilbertien réel complet par rapport à la métrique induite par le produit scalaire. Nous pouvons dire qu'un espace de Hilbert est un espace de Banach équipé d'un produit scalaire;

Espace des fonctions L^p

Soit Ω un ouvert de $\mathbb{R}^n$. On définit l'espace L^p de tous les fonctions mesurables $u : \Omega \to \mathbb{R}$ tel que $||u||_{L^p(\Omega)} < \infty$, en d'autre terme :

$$L^p(\Omega) = \{u : \Omega \to \mathbb{R}, u \text{ est Lebesgue mesurable}, ||u||_{L^p(\Omega)} < \infty\},$$

avec :

$$||u||_{L^p(\Omega)} = \begin{cases} (\int_\Omega |u|^p \,\mathrm{d}x)^{\frac{1}{p}} & \text{si} \quad 1 \leq p < \infty \\ \underset{\Omega}{ess\, sup}|u| & \text{si} \quad p = \infty \end{cases} \tag{2.1}$$

2.2.2 Espace des fonctions continûment différentiables

Soit Ω est un sous-espace de $\mathbb{R}^n$, nous définissons $C^k(\Omega)$ l'espace des fonctions $k-$fois continûment différentiables :

$$C^k(\Omega) = \{u : \Omega \to \mathbb{R} : u \,\text{est}\, k - \text{fois continûment différentiables}\}.$$

Le support d'une fonction continues $u(x)$ définie sur $\mathbb{R}^n$ est la fermeture de l'ensemble des points x où $u(x)$ ne s'annule pas :

$$supp(u) = \overline{\{x \in (R)^n : u(x) \neq 0\}}.$$

Un ensemble A est bornée s'il est contenu dans une boule $B_R(O)$ de centre O et de rayon R suffisamment large. Si le support de u est borné, nous disons que u à un support compact. $C_0^k(\Omega)$ désigne les fonctions $C^k(\Omega)$ à support compact.

2.2.3 Espace de Sobolev

Soit Ω un ouvert de $\mathbb{R}^n$. On pose

$$H^1(\Omega) = \{u \in L^2(\Omega) : \partial u / \partial x_i \in L^2(\Omega), \forall i = 1, ..., n\},$$

la dérivation est à comprendre au sense des distributions, l'espace $H^1(\Omega)$ est muni de la norme :

$$\|u\|_{H^1} = \left(\sum_{i=1}^{n} \int_\Omega \left| \frac{\partial u}{\partial x_i}(x) \right|^2 dx + \int_\Omega |u(x)|^2 dx \right)^{1/2},$$

la norme de $H^1(\Omega)$ est issue d'un produit scalaire noté $(u, v)_{H^1}$ tel que :

$$(u, v)_{H^1} = \sum_{i=1}^{n} \int_\Omega \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_i}(x) dx + \int_\Omega u(x) v(x) dx.$$

L'espace $H^1(\Omega)$ muni de la norme $\|.\|_{H^1}$ est un espace de Hilbert.

De la même façon, on définit les espaces de Sobolev $H^m(\Omega)$ où m est un entier strictement positif par :

$$H^m = \{u \in L^2(\Omega) : D^\alpha u \in L^2(\Omega), \alpha \in \mathbb{N}^n, |\alpha| \leq m\},$$

on le munit de la norme naturelle :

$$\|u\|_{H^m} = \left(\sum_{|\alpha| \leq m} \int_\Omega |D^\alpha u|^2 dx \right)^{1/2}.$$

De façon plus générale, pour tout $1 \leq p < \infty$ et pour tout $m \in \mathbb{N}$, $m \geq 1$, on peut définir les espaces de Sobolev :

$$W^{m,p}(\Omega) = \{u \in L^p(\Omega) : D^\alpha u \in L^p(\Omega), \forall \alpha \in \mathbb{N}^n, |\alpha| \leq m\},$$

que l'on munit de la norme :

$$\|u\|_{W^{m,p}} = \left(\sum_{|\alpha| \leq m} \int_\Omega |D^\alpha u|^p dx \right)^{1/p}.$$

Les espaces $W^{m,p}(\Omega)$ sont des espaces de Banach.

2.2.4 Espace des fonctions à variation bornée[3]

Dans les problèmes liés à la vision par ordinateur les discontinuités dans l'image ont une importance significative et présentent des détails indispensables pour tout traitement. Ainsi nous devons être capable de présenter les fonctions discontinues. L'espace $BV(\Omega)$ des fonctions à variation bornée est le mieux adapté pour rendre compte de cet aspect, mais avant de le définir ; on introduit le concept de la variation totale.

La variation totale

Soit $\Omega \in \mathbb{R}^n$ un ouvert ; la variation totale d'une fonction u n'est que la norme L^1 du gradient de cette fonction ; rigoureusement, on peut la définir par :

$$||u||_{TV(\Omega)} = \int_\Omega |\nabla u| \, \mathrm{d}x \quad \text{pour tout} \quad u \in C^1(\Omega).$$

Cette norme mesure les éventuelles oscillations représentant le bruit dans une image.

On peut redéfinir la variation totale d'une fonction donnée par la forme duale :

$$||u||_{TV(\Omega)} = sup\left\{ \int_{\Omega} u.\nabla g \, \mathrm{d}x : g \in C_0^1(\Omega, \mathbb{R}^n), |g(x)| \leq 1, \forall x \in \Omega \right\}.$$

Espace des fonctions à variation bornée

Une fonction $u \in L^1$ avec une variation totale finie, est dite une fonction à variation bornée. L'ensemble des fonctions à variation bornée, noté par $BV(\Omega)$ est défini par :

$$BV(\Omega) = \{u \in L^1(\Omega) : ||u||_{TV} < \infty\}.$$

Rappelons que $BV(\Omega)$ est un espace de Banach équipé de la norme :

$$||u||_{BV} = ||u||_{L^1(\Omega)} + ||u||_{TV(\Omega)}.$$

Pour en savoir d'avantage sur ce sujet, le lecteur pourra se référer au livre [3].

2.3 Les équations aux dérivées partielles (EDP)

Dans cette section on essaye d'introduire brièvement les **EDP** d'évolution souvent utilisées en traitement d'image et en vision par ordinateur, car les approches à base d'**EDP** ont pour intérêt qu'elles permettent d'obtenir dans de nombreux cas des résultats d'existence et d'unicité pour la solution recherchée, et qu'elles peuvent se mettre en œuvre à l'aide de puissants schémas numériques.

2.3.1 Définition d'une EDP d'évolution

Soit Ω l'ensemble de tous les points (x, y) formant une image I, défini par :

$$\begin{aligned} I : \quad & \Omega \times \mathbb{R}^+ \to \mathbb{R} \\ & (x, y, t) \to I(x, y, t) \end{aligned}$$

.

La plus part des **EDP** qui sont souvent utilisées dans le traitement d'image sont données par la relation F entre la dérivée $\frac{\partial I}{\partial t}$ et les dérivées successives de l'image I suivant x et y jusqu'à l'ordre n qui désignera l'ordre de l'**EDP**.

$$\begin{cases} I(x,y,0) & = I_0(x,y) \\ \frac{\partial I}{\partial t} & = F\big(I, \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial^2 I}{\partial x^2}, \frac{\partial^2 I}{\partial y^2}, \frac{\partial^2 I}{\partial x \partial y} \cdots \frac{\partial^n I}{\partial x^p \partial y^{n-p}}\big). \end{cases} \tag{2.2}$$

La condition initiale, à l'instant $t = 0$, est fournie par l'image dégradée $I(x,y,0) = I_0(x,y)$. Avec la donnée des conditions sur le bord de Ω, on peut assurer que le système (2.2) converge vers une solution stable. La résolution du système (2.2) nous permet d'avoir une série d'images $\{I_t\}_{t \geq 0}$, d'où une analyse multi-échelle de l'image I, avec un paramètre d'échelle t. Pour plus de détails et d'exemples sur les **EDP** et l'analyse multi-échelle de l'image, on renvoie le lecteur vers [30, 29].

2.3.2 EDP et images discrètes

Sachant qu'une **EDP** est définie sur un domaine continu, une discrétisation est nécessaire pour que l'**EDP** puisse mieux modéliser le traitement qu'on désire effectuer sur l'image [8, 16, 33, 28, 37]. on approxime la dérivée première en temps $\frac{\partial I}{\partial t}$ au point (x,y) via le développement de Taylor par :

$$\frac{\partial I}{\partial t} = \frac{I(t+\Delta t, x, y) - I(t,x,y)}{\Delta t},$$

par l'application d'un schéma de résolution d'Euler (Dans [36, 24, 23], le lecteur trouvera, en détail, d'autres méthodes pour la discrétisation des **EDO** et **EDP**) sur l'équation (2.2), on obtient :

$$\begin{cases} I(x,y,0) & = I_0(x,y) \\ I(x,y,t+\Delta t) & = I(x,y,t) + \Delta t F\big(I\big). \end{cases} \tag{2.3}$$

Ce qui génère le système itératif suivant :

$$\begin{cases} I(x,y,0) & = I_0(x,y) \\ I^{n+1}(x,y) & = I^n(x,y) + \Delta t F\big(I^n\big). \end{cases} \tag{2.4}$$

Le système (2.4) génère une série d'images $I^n(x, y)$ construites par récurrence à partir de l'image initialement fournie pour le traitement. L'incrément dt ne doit pas être trop petit pour que le nombre d'itérations nécessaires ne pas être trop grand, mais dt ne doit pas être non plus trop grand si on veut éviter d'osciller autour de la solution et rendre le système (2.4) numériquement instable.

D'autre part, la discrétisation des dérivées spatiales de l'image doit être minutieusement choisie pour assurer une bonne convergence.

2.3.3 Equation de Navier-Stokes

Les équation de Navier-Stokes modélisent le comportement des fluides (écoulements, tourbillons, turbulence ...). Elles ont été utilisées avec profit en traitement d'image [14, 31, 67] pour restaurer des films anciens et réparer les parties détériorées, pour zoomer dans une image, pour supprimer du texte,etc.

Ces équations de Navier-Stokes, qui seront utilisées dans le chapitre suivant, décrivent les déplacements des substances fluides en général, en fonction de deux variables primaires qui sont la vitesse d'écoulement et la pression. On s' intéresse au cas où le fluide est Newtonien (dans [63] on trouve des définitions exactes d'un fluide Newtonien, ainsi que ces propriétés physiques).

Les équations de Navier-Stokes sont :

$$\begin{cases} \rho(\frac{\partial u}{\partial t} + (u.\nabla u) - \nu\Delta u + \nabla p = f & \text{dans} \quad \Omega \times \mathbb{R}^+, \\ \operatorname{div} u = 0 & \text{dans} \quad \Omega \times \mathbb{R}^+, \end{cases} \tag{2.5}$$

où $\Omega \subset \mathbb{R}^n$ est un domaine qui représente une région de l'espace. Le domaine Ω sera toujours supposé borné et régulier, $u = u(x, t)$ est la vitesse de l'écoulement, $p = p(x, t)$ représente la pression du fluide, au point $x = (x_1, \dots, x_n) \in \Omega$ et à l'instant t. La vitesse est une fonction vectorielle $u = (u_1, \dots, u_n) \in \mathbb{R}^n$ avec : $u_i = u_i(x_1, \dots, x_n, t)$ et la pression p est une fonction scalaire. La densité $\rho > 0$ du fluide est choisie constante et $\nu > 0$ désigne la viscosité dynamique du fluide. Enfin, $f = (f_1, \dots, f_n)$ représente une densité massique de la force extérieure, la gravité par exemple. Les différents opérateurs différentiels intervenant dans les équations de Navier-Stokes sont définis de la façon suivante :

$$\nabla = (\frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n}), \quad u.\nabla u = \sum_{i=1}^{n} u_i \frac{\partial u}{\partial x_i}$$

$$\Delta u = \sum_{i=1}^{n} \frac{\partial^2 u}{\partial x_i^2}, \quad \mathrm{div} = \sum_{i=1}^{n} \frac{\partial u_i}{\partial x_i}$$

Aux équations du système (2.5), on peut ajouter une condition aux limites de type Dirichlet sur le bord $\partial\Omega$, c'est-à-dire :

$$u(x,t) = g(x,t) \quad \text{pour} \quad x \in \partial\Omega,\, t > 0,$$

plus une donnée initiale sur la vitesse :

$$u(x,0) = u_0(x) \quad \text{pour} \quad x \in \partial\Omega,$$

où u_0 est une fonction donnée.

2.4 Mise en œuvre numérique

Dans cette sous-section on s'intéresse à la résolution numérique des équations aux dérivées partielles par les différences finies.

2.4.1 Différences finies

Dans la littérature, plusieurs approches sont proposées, en l'occurrence les méthodes spectrales, les éléments finis, les volumes finis. La méthode des différences finies se révèle la plus adaptée pour les **EDP** utilisées dans le traitement des images. Le succès de cette approche dans le traitement d'images est dû à la structure de l'image numérique qu'on peut associer à un maillage régulier naturel.

Principe général

Le principe de la méthode de différences finies est du aux travaux de plusieurs mathématiciens du $18^{\text{ème}}$ siècle Euler, Taylor, Leibniz $\ldots$, il consiste à approximer les dérivées partielles par des quotients. Pour mieux présenter le principe de cette méthode, nous considérons le problème suivant à une dimension ; pour le cas de dimension supérieure, nous renvoyons aux [8, 28, 2, 27] :

$$\begin{cases} \mathcal{L}v = F, & t > 0 \,\text{et}\, x \in \mathbb{R} \\ v(0,x) = f(x), & x \in \mathbb{R} \end{cases} \tag{2.6}$$

où v et F sont définies sur $\mathbb{R}^2$, $\mathcal{L}$ est un opérateur différentiel linéaire et la fonction v désigne la solution exacte de (2.6).

Comme exemple pratique, nous considérons le long de cette section, l'équation de la chaleur à une dimension :

$$\frac{\partial v}{\partial t} = \nu \frac{\partial v^2}{\partial x^2}, \quad t > 0, \quad x \in \mathbb{R}, \tag{2.7}$$

où $\nu > 0$ est une constante ; ce problème est équivalent à :

$$\mathcal{L}v = 0 \quad \text{avec} \quad \mathcal{L}v = \frac{\partial v}{\partial t} - \nu \frac{\partial^2 v}{\partial x^2},$$

la condition initiale du problème précédent est $v(0,x) = f(x)$.

Pour résoudre le problème (2.7), nous discrétisons le domaine, en plaçant une grille uniforme de pas d'espace Δx couvrant le domaine entier ; de même, l'espace temps sera discrétisé avec un pas Δt (voir figure 2.2).

Résoudre le problème numériquement revient à trouver une fonction discrète u définie aux nœuds $(n\Delta t, i\Delta x)$. Nous désignons par v_i^n la valeur de v en ces nœuds.

La fonction u sera obtenue comme solution de l'équation discrète :

$$\begin{cases} L_i^n u_i^n = F_i^n & , i = -\infty \ldots + \infty, \\ u_i^0 = f(i), \end{cases} \tag{2.8}$$

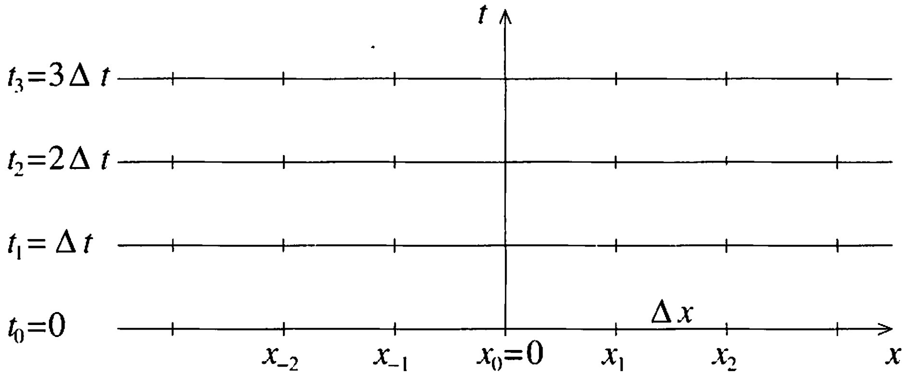


FIGURE 2.2 – Grille de discrétisation

où L_i^n, F_i^n correspondent aux approximations discrètes de $\mathcal{L}$ et F .

Revenons sur notre exemple pour l'équation de la chaleur, on obtient des dérivées approchées, si Δx est petit, grâce au développement de Taylor (Voir [23, 24]) de $v(t, x + \Delta x)$ au voisinage de x :

$$v(t, x + \Delta x) = v(t, x) + \Delta x \frac{\partial v}{\partial x}(t, x) + \frac{(\Delta x)^2}{2!} \frac{\partial^2 v}{\partial x^2}(t, x) + \ldots + \frac{(\Delta x)^n}{n!} \frac{\partial^n v}{\partial x^n}(t, x) + \ldots$$

En tronquant cette série au premier ordre Δx, on obtient l'approximation de la première dérivée spatiale :

$$\frac{\partial v}{\partial x}(n\Delta t, i\Delta x) = \frac{v_{i+1}^n - v_i^n}{\Delta x} + O(\Delta x)$$

L'approximation de la dérivée première en espace est alors d'ordre 1, et l'erreur de troncature $O(\Delta x)$ tend vers 0 linéairement.

Pour la dérivée seconde, le principe est identique, et repose sur les développements de Taylor au voisinage de $i\Delta x$. On écrit :

$$v(t + \Delta t, x) = v(t, x) + \Delta t \frac{\partial v}{\partial x}(t, x) + \frac{\Delta t^2}{2}(\frac{\partial^2 v}{\partial x^2})(t, x) + \frac{\Delta t^3}{6}(\frac{\partial^3 v}{\partial x^3})(t, x) + \ldots$$

$$v(t - \Delta t, x) = v(t, x) - \Delta t \frac{\partial v}{\partial x}(t, x) + \frac{\Delta t^2}{2}(\frac{\partial^2 v}{\partial x^2})(t, x) - \frac{\Delta t^3}{6}(\frac{\partial^3 v}{\partial x^3})(t, x) + \ldots$$

En faisant la somme de ces deux égalités, on aboutit à une approximation de la dérivé seconde en espace :

$$\frac{\partial^2 v}{\partial x^2}(n\Delta t, i\Delta x) = \frac{v_{i+1}^n - 2v_i^n + v_{i-1}^n}{\Delta x^2} + O(\Delta x)$$

Le schéma précédent est d'ordre deux ; il est dit "centrée" pour approximer la dérivée seconde en espace. Si Δt est petit, par le même procédé, on peut obtenir :

$$\frac{\partial v}{\partial t}(n\Delta t, i\Delta x) = \frac{v_i^{n+1} - v_i^n}{\Delta t} + O(\Delta t)$$

comme une approximation de la dérivée première temporelle.

On obtient donc une approximation de l'équation 2.7 donnée par :

$$L_i^n u_i^n = 0 \quad \text{avec} \quad L_i^n u_i^n = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \nu \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}. \tag{2.9}$$

Cette équation (2.9) aux différences finies peut être réécrite comme suivant :

$$u_i^{n+1} = (1 - 2r)u_i^n + r(u_{i+1}^n + u_{i-1}^n) \quad \text{avec} \quad r = \nu \Delta t / \Delta x^2. \tag{2.10}$$

Ce schéma est dit explicite, ce qui signifie que les valeurs de la solution approchée, à l'instant $(n + 1)\Delta t$ sont obtenues uniquement en utilisant les valeurs de la solution à l'instant $n\Delta t$.

La question qui se pose maintenant, est : quelle est la relation entre l'équation discrétisée et l' **EDP** initiale ainsi que la relation entre leurs solutions respectives ? Pour répondre à cette question nous définirons la convergence, la consistance et la stabilité :

Consistance

Le schéma (2.9) est dit ponctuellement consistant si pour toute fonction régulière $\phi = \phi(t, x)$,

$$(\mathcal{L}\phi - F)|_i^n - [L_i^n \phi(n\Delta t, i\Delta x) - F_i^n] \to 0$$

quand Δx, $\Delta t \to 0$ et $((n + 1)\Delta t, i\Delta x) \to (t, x)$.

Stabilité

Pour un problème d'évolution, un schéma numérique est stable si les erreurs (d'arrondi, de troncature, ...) ne peuvent pas croitre pendant la procédure numérique d'un pas de temps au suivant. Un schéma peut être :

- **Inconditionnellement stable** : si quelque soient Δt et Δx les erreurs n'explosent pas au fil des itérations ;
- **Conditionnellement stable** : on doit poser une condition sur Δt et Δx pour que la solution n'explose pas.
- **inconditionnellement instable** : si quelque soient Δt et Δx les erreurs s'amplifient au fil des itérations.

Stabilité et analyse de Fourier

Pour prouver la stabilité d'un schéma numérique par la méthode de l'analyse de Fourier, on injecte dans le schéma un mode de Fourier

$$v_j^n = A(k)^n exp(2i\pi k x_j) \qquad \text{avec} \qquad x_j = j\Delta x,$$

et on déduit la valeur du facteur d'amplification $A(k)$. On appelle condition de stabilité de Von Neumann l'inégalité

$$\|A(k)\| \leq 1 \quad \text{pour tout mode} \quad k \in \mathbb{Z}.$$

Si la condition de stabilité de Von Neumann est satisfaite (avec éventuellement des restrictions sur Δt et Δx), alors le schéma est stable, si non il est instable. Pour le schéma discrétisant l'équation de la chaleur on peut vérifie qu'il est stable si et seulement si la condition $2\nu\Delta t \leq (\Delta x)^2$ est satisfaite.

Convergence

Une condition nécessaire et suffisante pour qu'un schéma discrétisant une EDP linéaire, soit convergent est d'avoir, simultanément, la stabilité et la consistance.

2.4.2 Résolution des systèmes linéaires

Considérons un système d'équations linéaires de la forme

$$AX = B, \tag{2.11}$$

où A une matrice inversible de dimension $N \times N$, B est un vecteur connu et X le vecteur inconnu. Il existe deux grandes familles de méthodes de résolution :

- **Les méthodes directes** : qui permettent de résoudre le système (2.11) soit par triangularisation ou soit par factorisation de la matrice A. Les principales méthodes sont :

  1. **Méthode de Gauss** : nous transformons le système (2.11) en un système équivalent $A_1X = B_1$, où A_1 est une matrice triangulaire, et on procède à la résolution.

  2. **Méthode de Gauss-Jordan** : consiste à transformer A en une matrice identité.

  3. **La factorisation** LU on décompose la matrice A sous la forme LU où L est une matrice triangulaire unitaire inférieur et U une matrice triangulaire supérieure, et le système (2.11) devient :

  $$L\underbrace{UX}_{=Y} = b \Leftrightarrow \begin{cases} LY = b \\ UX = Y \end{cases}$$

  4. **La Méthode de Cholesky** dans la cas où A est matrice symétrique définie positive, nous optons pour une décomposition de la forme $A = LL^T$, avec L matrice triangulaire inférieure.

  Pour plus de détails sur les méthodes qui nous permet l'obtention de chaque décomposition le lecteur peut consulter [54, 42]

- **Les méthodes itératives** : qui introduisent une notion de convergence vers la solution. On commence par décomposer la matrice A sous la forme $G - H$, avec G

inversible, alors :

$$\begin{aligned} Ax = b \Leftrightarrow \quad & Gx = Hx + b \\ \Leftrightarrow \quad & x = Mx + N \end{aligned} \tag{2.12}$$

Avec $M = G^{-1}H$ et $N = G^{-1}B$. Ce qui nous ramène à un problème de recherche de point fixe, on définie une suite récurrente :

$$\begin{cases} X^0 \quad \text{vecteur quelconque de} \quad \mathbb{R}^+ \\ X^{n+1} = MX^n + N \end{cases}$$

Si cette suite est convergente, sa limite X est solution du système. Pour le choix de la décomposition, nous distinguons plusieurs méthodes :

1. **Méthode de Jacobi** : On écrit $A = D - E - F$ où D est la partie diagonale de A, $-E$ (resp. $-F$) la partie triangulaire inférieur (resp. supérieur) de A. On choisit alors $G = D$ et $H = E + F$, et on a alors $M = D^{-1}(E + F) = D^{-1}(E + F) = I - D^{-1}A$ et $N = D^{-1}B$

   Dans la pratique, on inverse pas D, le calcule effectif se fait en résolvant directement le système :

   $$DX^{k+1} = (E + F)X^k + B$$

2. **Méthode de Gauss-Seidel sans ou avec relaxation** : Pour cette méthode, on opte pour la décomposition : $G = D - E$ et $H = F$, on a alors $M = (D - E)^{-1}F$ et $N = (D - E)^{-1}B$, le calcule effectif se fait en résolvant :

   $$(D - E)X^{k+1} = FX^k + B$$

   . Pour relaxer la méthode, on considéré $G = \frac{1}{\omega}D - E$ et $H = \frac{1-\omega}{\omega}D + F$ où ω est le paramètre de relaxation, et M sera donnée par : $M = (\frac{1}{\omega}D - E)^{-1}(\frac{\omega-1}{\omega}D + F)$

3. **Méthode du gradient** : Soit un paramètre réel $\alpha = 0$. On appelle méthode du gradient la méthode itérative basée sur la décomposition suivante : $G = \frac{1}{\alpha}Id$ et $H = \frac{1}{\alpha}Id - A$, la grande applicabilité de cette méthode provient principalement du fait qu'elle peut être interprétée en tant que méthode de minimisation de la fonction :$f(x) = \frac{1}{2}Ax.x - b.x$.

4. **Méthode du gradient conjugué avec ou sans pré-conditionnement** : La méthode du gradient conjugué est souvent utilisée pour résoudre des systèmes linéaires dont la matrice est symétrique réelle définie positive. Combinée avec un préconditionnement, cette méthode est une amélioration de la méthode du gradient. La construction de l'algorithme de la méthode du gradient conjugué, se fait à partir des espace de Krylov. Pour un exposé détaillé de cette méthode, nous renvoyons le lecteur vers [2, 54, 42, 56].

Chapitre 3

Chapitre 3

Inpainting par un modèle P-Laplacien

3.1 Introduction et motivation

Dans cette section, on décrit le processus d'inpainting et on présente un état de l'art de l'inpainting, où on montre comment les différentes techniques agissent, à savoir la formulation variationnelle [19, 17, 11, 10, 12] et l'approche à base d' **EDP** [14, 13, 15, 31, 67, 18, 26, 58], qui peuvent être appliquées avec succès à ce problème.

3.1.1 Problématique de l'inpainting en imagerie

La transmission d'une image réelle, le stockage ou une mauvaise manipulation, peuvent être l'origine de plusieurs dégradation au niveau de l'image ce qui cause une perte de l'information dans une partie de l'image, voir figure 3.1, L'inpainting a pour but de restaurer ces images endommagées.

L'inpainting peut être aussi un meilleur moyen pour enlever artificiellement des parties indésirables dans l'image. La restauration doit être faite pour que l'image finale semble inchangée à un observateur qui ne connaît pas l'image originale au point de rendre les zones reconstruites non décelables ou du moins difficilement ? voir figure 3.2 et 3.3

Étant donnés un domaine $D \subset \mathbb{R}^2$, un sous ensemble $\Omega \subset D$, et une image I_0 définie

FIGURE 3.1 – Inpainting sur une image mal conservée

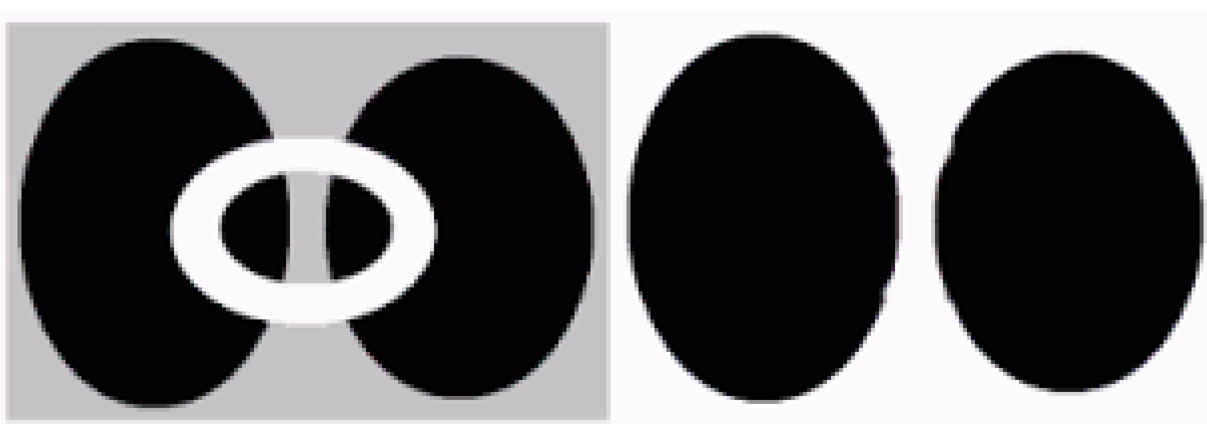


FIGURE 3.2 – Inpainting pour supprimer une partie d'une image synthétique

sur $D \backslash \Omega$, on cherche une image I, qui est l' image reconstruite à partir de l'image I_0 , qui correspond à I_0 à l'extérieur du Ω et qui a un contenu plus significatif à l'intérieur de Ω, ce contenu semble naturel pour l'œil d'un observateur humain, qui est le seul juge final de l'efficacité d'un tel traitement. Ceci peut être obtenu, en examinant l'information disponible autour du domaine Ω, et la propager à l'intérieur de Ω, voir 3.4.

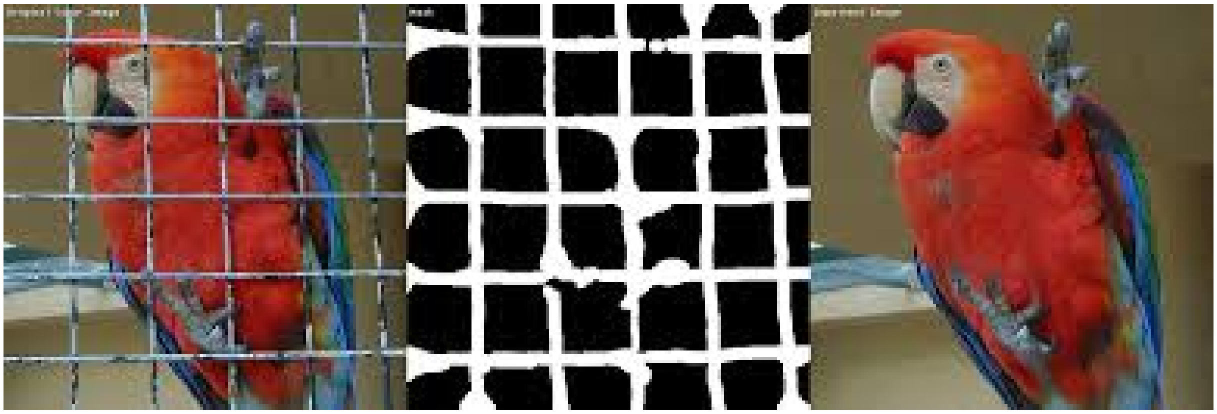


FIGURE 3.3 – Inpainting pour supprimer une partie d'une image réelle

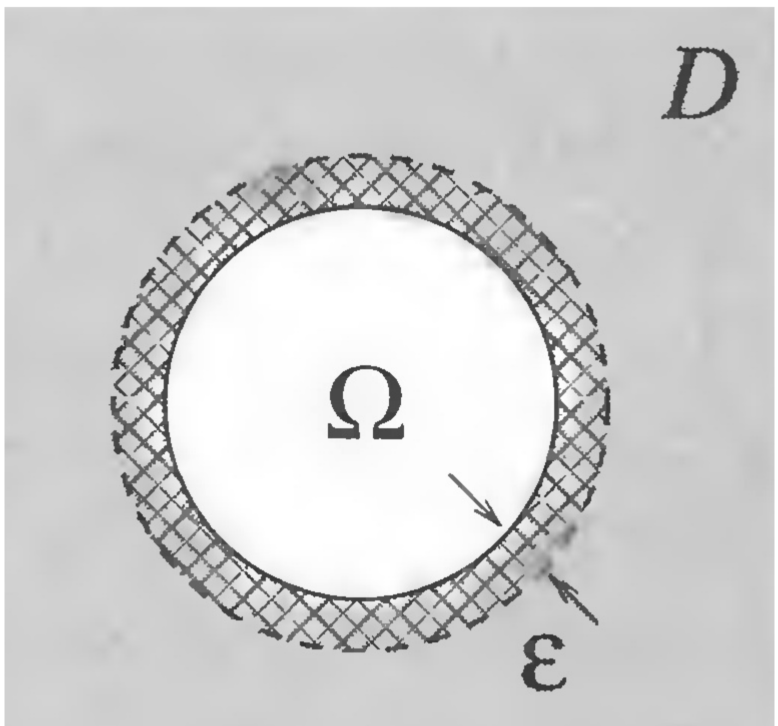


FIGURE 3.4 – L'inpainting, mathématiquement décrit.

Du point de vue mathématique, l'inpainting n'est qu'un problème d'interpolation dans Ω, c'est-à-dire une interpolation spatiale intelligente des pixels de l'image bordant les régions à reconstruire, qui permet de bien recomposer les géométries globales des structures de l'image et d'estimer des valeurs de pixels à l'intérieur de ces régions. Les parties suivantes, seront consacrées à la présentation de certaines approches d'inpainting :

3.1.2 Approche par formulation variationnelle

Les approches traditionnelles développées pour l'inpainting d'images réelles incluent souvent des opérateurs agissant dans le domaine spatiale ou spectrale, où leur forme est généralement déterminée en fonction des critères à minimiser et des informations a priori que l'on peut avoir sur le problème à résoudre ; voici quelques exemples, dans leurs classement chronologique d'apparition :

L'Approche de Masnou et Morel

Le point de départ pour plusieurs travaux d'inpainting est le travail réalisé par Nitzberg, Mumford et Shiota [52] sur l'élasticité, qui ont essayé d'identifier les objets d'occlusion et occlus. On commence par noter que expérimentalement, le système visuel détecte les occlusions même à des niveaux très bas, et il est capable de compléter partiellement les contours occlus en respectant tout simplement le principe de bonne continuité [41].

Sachant que, lorsqu'un objet occlut d'autres objets, les frontières d'occlusion et occlues forment des T-jonctions particulières ; l'idée est de les détecter et de les relier d'une manière continue par la suite, voir figure 3.5.

Commençons par définir la notion de T-jonction précisément, en utilisant les ensembles de niveau associé au valeur c du niveau de gris donnés par :

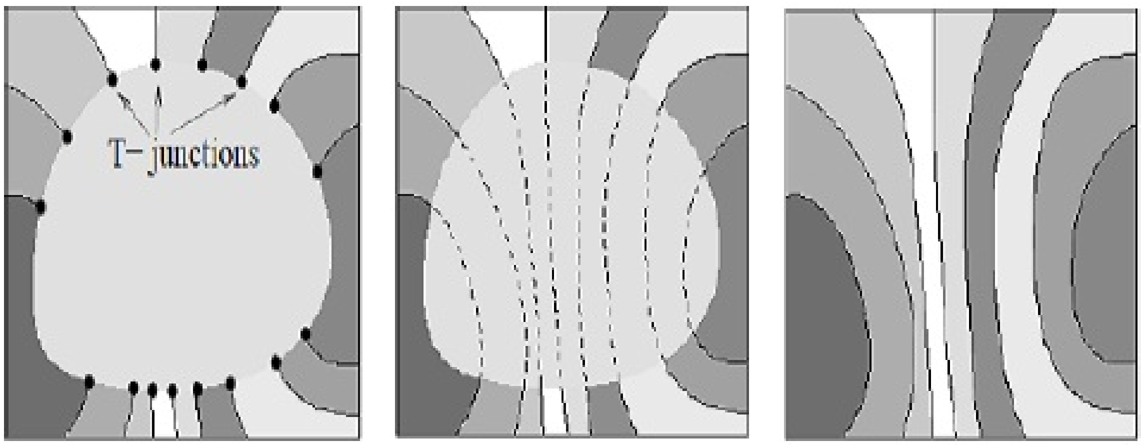
$$\chi_c(I) = \{(x, y) \in \Omega : I(x, y) \geq c\}.$$

La famille $\chi_c(I)$ nous fournit une représentation complète de l'image I grace à la formule :

$$I(x, y) = sup\{c : (x, y) \in \chi_c(I)\}, \quad \text{pour tout} \quad (x, y) \in \Omega.$$

Nous définissons les lignes de niveau comme étant les frontières des ensembles de niveau. Si on suppose que $I_0 \in BV(D \backslash \Omega)$, une T-jonction est définie comme étant le point d'intersection de $\partial\Omega$ avec une ligne de niveau. Deux T-jonctions sont dites compatibles si elles sont issues du même ensemble de niveau avec une même orientation de ∇I_0 aux deux points définissant ces deux T-jonctions.

La question maintenant est : étant donné deux T-jonctions T_1 et T_2 sur $\partial\Omega$ avec deux orientations θ_1 et θ_2 de ∇I_0, voir figure 3.5, comment peut-on les relier avec une courbe Γ dans Ω, tout en respectant le plus possible le principe de bonne continuité ?



(a) (b) (c)

FIGURE 3.5 – Illustration de l'approche de Masnou et Morel, où les lignes de niveau sont reconstruites en premier puis les niveaux de gris en suite.

Masnou et Morel [47], inspirés par les travaux de Nitzberg et al [52], proposent de chercher la courbe Γ en minimisant la fonctionnelle :

$$\int (\alpha + \beta\kappa^p) dH^1 + (\theta_1.N_1) + (\theta_2.N_2),$$

où α et β sont deux constantes positives, $p \geq 1$, κ est la courbure de Γ, tandis que $(\theta_1.N_1)$ et $(\theta_2.N_2)$ désignent les angles entre θ_i et N_i, avec $i = 1, 2$ où N_i est la normale de Γ à la jonction T_i, où Γ est la courbe reliant les deux jonctions T_1 et T_2. Par application de la minimisation sur l'ensemble de toutes les T-jonctions admissibles, on obtient :

$$\int \left(\sum_{\Gamma \in F_c} \int (\alpha + \beta\kappa^p) dH^1 + (\theta_1.N_1) + (\theta_2.N_2) \right) dx,$$

où F_c représente la famille de toutes les courbes reliant les jonctions associées à l'ensemble de niveaux $\chi_c(I_0)$. Une fois toutes les courbes sont reconstruites, les auteurs procèdent à l'opération de remplissage entre les lignes de niveaux établies sur Ω avec le niveau de gris approprié, celui qui est sur le bord de Ω, voir figure 3.5. Cette méthode propose une reconstruction invariante au changement de contraste, en accord avec le principe de bonne continuité. L'existence et l'unicité de la solution sont données en détail dans [47].

L'Approche de Ballester et al [11, 10, 12]

Dans le même esprit des travaux de Nitzberg et al [52] sur l'élasticité, Ballester et al [11, 10, 12] dans une série de travaux, proposent d'attaquer le problème d'inpainting en interpolant conjointement les champs de vecteurs θ de direction orthogonal des lignes de niveau, et les niveaux de gris de l'image dans la zone manquante Ω de l'image. Rappelons d'abord que le champ de vecteur θ est lié à l'image I en s'accordant à la contrainte :

$$\theta.\nabla I = |\nabla I| \quad \text{et} \quad |\theta| \leq 1,$$

puis les auteurs cherchent, ensuite (I, θ) comme le minimum de la fonctionnelle :

$$\int_\Omega |div(\theta)|^p (a + b|\nabla k \star I|) dx,$$

où $I = I_0$ dans $D \backslash \Omega$, et sur le bord de Ω on prend $\theta.N = \theta_0.N$, de plus les paramètres p, a et b vérifient : $p > 1$, $a > 0$ et $b \leq 0$. k est un noyau de régularisation de classe C^1 avec $k(I) > 0$ et $N(x, y)$ représente la normale au point $x \in \partial\Omega$. Dans ce modèle, les informations investies pour remplir la partie manquante sont : I_0 sur $\partial\Omega$ et θ_0 la normale sur les lignes de niveau. Un exemple d'application de cette méthode est donné dans la figure 3.6 sur une image synthétique, et sur une image réelle dans la figure 3.7.

La Minimisation de la variation totale de Chan et Shen [19]

Étant donnée une image $I(x, y)$, définie sur une région D bornée, ouverte et convexe, sa variation totale est définie comme suit :

$$TV(I) = \int_D |\nabla I| dx = \int_D \sqrt{I_x^2 + I_y^2} dx.$$

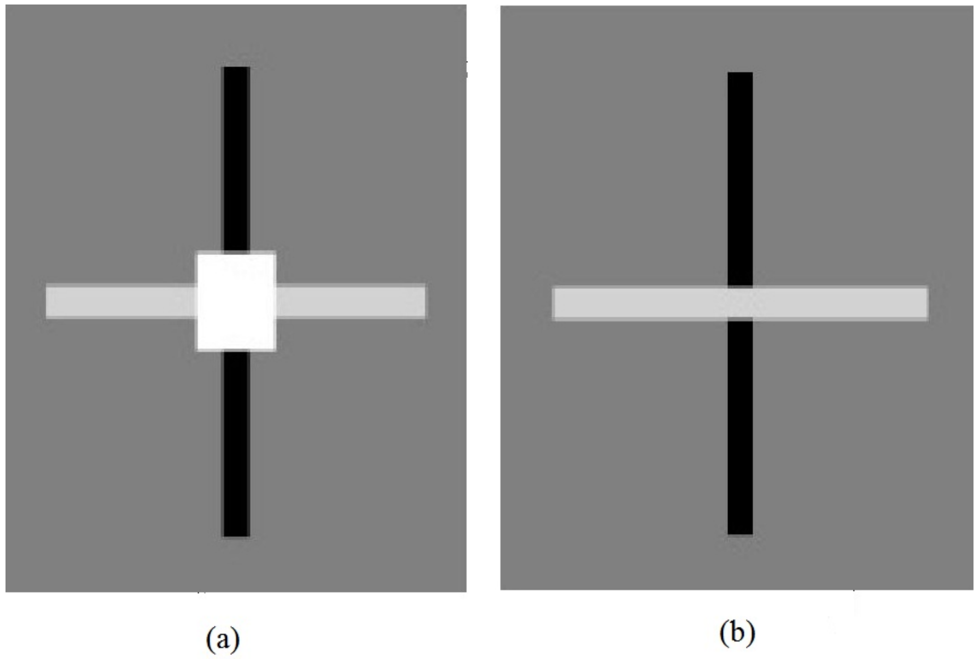


FIGURE 3.6 – Exemple illustrant l'application de l'approche de Ballester et al sur une image synthétique.

FIGURE 3.7 – Exemple illustrant l'application de l'approche de Ballester et al sur l'image de Lena.

Chan et Shen [19] ont introduit une approche basée sur la variation totale, en minimisant, sur la zone à reconstruire, l'énergie suivante :

$$\int_D |\nabla I| d\,x,$$

où on impose que $I = I_0$ sur $D\backslash\Omega$. Pour opérer un lissage de bruit sur l'image à reconstruire, on remplace la contrainte $I = I_0$ sur $D\backslash\Omega$ par :

$$\frac{1}{|D\backslash\Omega|}\int_{D\backslash\Omega}(I - I_0)^2 d\,x = \sigma^2.$$

σ est la variance du bruit. Les auteurs proposent de minimiser sur $BV(D)$ la formulation sans contrainte suivante :

$$\lambda\int_{D\backslash\Omega}(I - I_0)^2 \mathrm{d}\,x + \int_D |\nabla I| \mathrm{d}\,x,$$

où λ est le multiplicateur de Lagrange.

Des résultats d'existence et d'unicité de la solution sont donnée dans[19], la figure 3.8 donne un exemple de l'application de cette approche pour la suppression des taches sur une image réelle.

(a) (b)

FIGURE 3.8 – Exemple d'application de la variation totale à l'inpainting.

En fin, nous notons que pour résoudre les problèmes variationnels précédemment exposés, on utilise souvent une technique basée sur les multiplicateurs de Lagrange ; une technique qui cherche le minimum d'une énergie comme une solution d'une **EDP**, pour plus de détails voir [26, 58, 50].

3.1.3 Approche par (EDP)

Il existe dans la littérature plusieurs modèles d'inpainting qui utilisent des **EDP** pour diffuser ou transporter l'information qui avoisine la zone d'inpainting ; parmi ces **EDP**, un grand nombre ne provient pas d'une approche variationnelle ; dans ce qui suit, on présentera deux modèles, et dans [16, 8, 57], le lecteur trouvera plus de modèles.

L'approche de Bertalmio et al [14]

L'une des premières **EDP** dans le domaine d'inpainting à été proposée par Bertalmio et al [14], inspiré par les techniques de base utilisées dans le monde artistiques ; leur **EDP** propage des informations situées sur le bord de la zone d'inpainting en utilisant une équation de transport d'ordre 3. L'information à transporter est un détecteur de contours, par exemple le Laplacien de l'image, la direction dans laquelle l'information sera transportée est la direction des lignes de niveau, c'est-à-dire la direction orthogonale au gradient de l'image. Les auteurs modélisent la propagation du Laplacien dans la direction des isophotes en résolvant l'**EDP** :

$$\frac{\partial I}{\partial t}(t,x,y) = \nabla(\Delta I(t,x,y)).\nabla I^{\perp}(t,x,y), \forall (x,y) \in \Omega. \tag{3.1}$$

A cette **EDP**, les auteurs, dans [14], ajoutent la condition initiale $I(0,x,y) = I_0^{ext}(x,y)$, où I_0^{ext} est une quelconque extension continue de la donnée I_0 sur Ω. Comme condition au bord, ils prennent : $I(t,x,y) = I_0(x,y)$ pour tout $(x,y) \in \partial\Omega$ et pour tout $t \geq 0$.

Nous pouvons aussi transporter le niveau de gris de l'image I dans la direction des isophotes du Laplacien de I, l'**EDP** résultante est :

$$\frac{\partial I}{\partial t}(t,x,y) = \nabla(\Delta I(t,x,y))^{\perp}.\nabla I,$$

une fois que l'état stationnaire d'évolution est atteint ($\frac{\partial I}{\partial t} = 0$), la solution vérifie la même équation (3.1).

Les auteurs dans [13] propose d'appliquer un processus de diffusion en alternance avec l'équation de transport, pour assurer une meilleure évolution de l'image. On termine par donner un exemple d'opération d'inpainting sur une image ancienne, obtenu par cette **EDP**, voir figure 3.9.

(a) (b)

FIGURE 3.9 – Restauration d'une image ancienne par le modèle de Bertalmio et al.

L'Approche de Chan et Shen : Diffusion Entrainée par Courbure

Dans le but d'améliorer la performance du processus d'inpainting proposé dans [17], un processus qui est basé principalement sur la minimisation de la variation totale d'une image I. Chan et Shen remarque que l'inpainting par la variation totale souffre d'un inconvénient majeur car le modèle ne restaure pas bien l'objet qui a des parties déconnectées et éloignées par la zone d'inpainting, voir la figure 3.10 pour un exemple typique.

Donc, dans certains cas le résultat d'inpainting par variation totale viole même le principe de la connectivité de la perception humaine, la plupart des humains préfèrent avoir les deux parties disjointes reliées, même quand elles sont éloignées [52, 41, 40].

Pour remédier à cette non cohérence avec le principe de la connectivité, Chan et Shen propose dans [18] un nouveau modèle d'inpainting, utilisant l' **EDP** d'inpainting suivante :

$$\frac{\partial I}{\partial t} = div\left(\frac{\nabla I}{|\nabla I|}\right) + \lambda_e(I - I_0, \qquad (3.2)$$

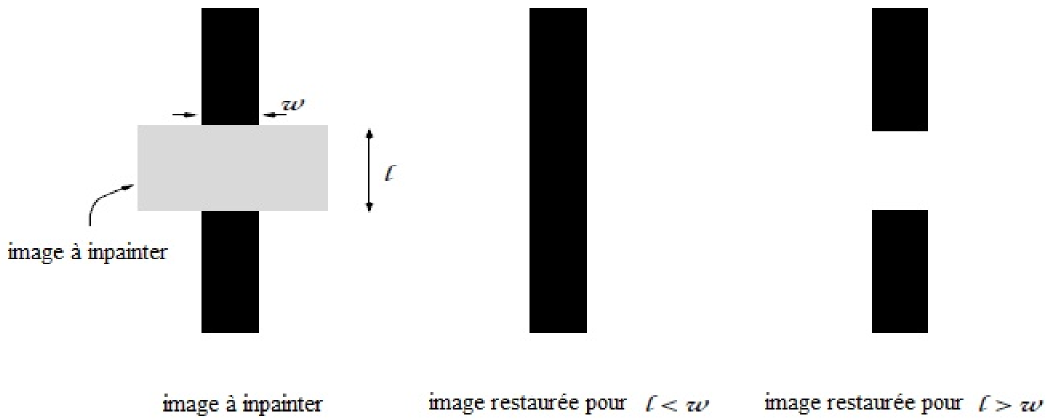


FIGURE 3.10 – L'inpainting par la variation totale qui dépend de la zone d'inpainting et ses dimensions.

où $\lambda_e(x) = \lambda(1 - \chi_\Omega(x))$ est le multiplicateur de Lagrange étendu, χ_Ω est une fonction caractéristique sur la zone d'inpainting Ω et λ est une constante. Le modèle de Chan et Sheh [18] implique la courbure κ des lignes de niveau, et remplace le coefficient de diffusion $\frac{1}{|\nabla u|}$ dans l'équation (3.2) qui dépend uniquement du contraste des isophotes ce qui ne prend pas en compte le cas où la courbure prend des valeurs infinis (Dans les coins par exemple). Le modèle proposé par Chan et Sheh est :

$$\begin{cases} \frac{\partial I}{\partial t} = div(\frac{g(\kappa)\nabla I}{|\nabla I|}) \quad \text{dans} \quad \Omega, \\ I = I_0 \quad \text{dans} \quad D/\Omega, \end{cases} \tag{3.3}$$

où g est une fonction croissante telle que $g(0) = 0$ et $g(+\infty) = +\infty$. Typiquement nous pouvons choisir $g(s) = s^p$ avec $p \geq 1$. Pour incorporer un processus de lissage pour le cas où l'image est en plus bruitée, les auteurs combinent l'opération d'inpainting et le lissage dans le modèle de la variation totale dans une même équation et proposent :

$$\frac{\partial I}{\partial t} = div\left(\frac{g(x,\kappa)\nabla I}{|\nabla I|}\right) + \lambda_e(I - I_0).$$

Avec :

$$g(x, \lambda_e) = \begin{cases} (1, \lambda) & \text{si} \quad x \in D/\Omega, \\ (g(s), 0) & \text{si non.} \end{cases} \tag{3.4}$$

Les résultats obtenu par le modèle de Chan et Sheh montrent la cohérence de cette méthode avec la perception humaine et les lois de vision (voir figure 3.11).

(a) (b)

FIGURE 3.11 – Restauration de la barre complète même pour une largeur relativement grande., contrairement à l'inpainting par la variation totale qui restaure la même image, mais en deux barres séparées.

3.2 P-Laplacien

3.2.1 Introduction

Comme on a vu précédemment, pour s'affranchir de dégradation dans une image, certaines techniques se basent sur l'utilisation des **EDP** avec des termes de diffusion qui ont tendance à atténuer les contours présents dans l'image. Cet effet de bord est particulièrement pénalisant pour une future analyse d'image, par exemple pour les algorithmes de segmentation dont la finalité est de partager l'image en régions homogènes ou de retrouver la géométrie des objets présents dans l'image.

3.2.2 Définition du P-Laplacien

L'équation de Laplace $\Delta I = 0$ qui s'écrit classiquement :

$$\frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} = 0, \tag{3.5}$$

est l'équation d'Euler-Lagrange de l'intégrale de Dirichlet définie par :

$$\begin{aligned} D_2 I) &= \int_\Omega |\nabla I|^2 \, dx \, dy, \\ &= \int_\Omega \left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2 \, dx \, dy. \end{aligned}$$

Si on considère une puissance p telle que :

$$D_p(I) = \int_\Omega |\nabla I|^p \, dx \, dy. \tag{3.6}$$

$$= \iint_\Omega \left(\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2\right)^{\frac{p}{2}} dx \, dy, \tag{3.7}$$

l'équation d'Euler-Lagrange correspondante est :

$$div(|\nabla I|^{p-2} \nabla I) = 0, \tag{3.8}$$

que nous appelons l'équation du P-Laplacien. L'opérateur associé du P-Laplacien sera défini par :

$$\Delta_p = div(|\nabla I|^{p-2} \nabla I).$$

Deux cas limites se présentent :

- Pour p=2 : On retrouve le Laplacien, qui rend compte d'une diffusion isotrope qui ne prend pas en compte la structure de l'image et qui engendre la disparition des contours et qui par conséquence rend l'image de plus en plus floue.
- Pour p=1 : On retrouve un terme qui rend compte d'une diffusion anisotrope et qui dépend de l'amplitude totale des oscillations d'une image. Cette diffusion a l'avantage de ne pas pénaliser les discontinuités qui caractérise les contours dans l'image, et de supprimer les oscillations, avec l'inconvénient de créer l'effet d'escalier connus sous le nom "staircase-effect".

L'expression du P-Laplacien dans des coordonnées locales permet d'analyser ces propriétés et de les comparer à d'autre termes de diffusion ; nous choisissons un système de coordonnées orthogonales locales (ξ, η) de sorte que η soit parallèle au gradient et ξ soit perpendiculaire :

$$\eta = \frac{(I_x, I_y)}{|\nabla I|} = \frac{\nabla I}{|\nabla I|} \quad \text{et} \quad \xi = \frac{(-I_y, I_x)}{|\nabla I|} = \frac{\nabla^{\perp} I}{|\nabla I|}.$$

En se basant sur la définition de la dérivée seconde directionnelle on obtient :

$$\Delta_p I = |\nabla I|^{p-2} I_{\xi\xi} + (p-1)|\nabla I|^{p-2} I_{\eta\eta}. \tag{3.9}$$

L'équation (3.9) est interprétée comme somme d'une diffusion dans la direction tangentielle et d'une diffusion dans la direction du gradient, $|\nabla I|^{p-2}$ et $(p-1)|\nabla I|^{p-2}$ agissent comme des coefficients qui contrôlent la performance de la diffusion. De cette décomposition, on remarque que la diffusion dans le modèle de variation totale est uniquement dans la direction tangentielle ξ avec un coefficient $|\nabla u|^{-1}$, tandis que le modèle du P-Laplacien diffuse dans les deux directions ce qui permet de minimiser considérablement l'effet d'escalier.

3.3 Description du modèle proposé

Un certain nombre d'approches visant à améliorer le processus d'inpainting et la reconstruction des parties endommagées dans l'image ont été publiées [14, 13, 15, 31, 67, 19, 17, 11, 10, 12, 18, 26, 58]. Ces méthodes basées sur des techniques différentes les unes des autres fournissent des résultats satisfaisants, mais la minimisation de l'effet d'escalier dans l'image reste un point délicat.

Dans cette section, on présente un modèle basé sur l'utilisation d'une analogie entre les équations de Navier-Stokes et le traitement d'image :

3.3.1 Modèle proposé

On propose le modèle suivant, basé sur la résolution de l'EDP suivante :

$$\frac{\partial \omega}{\partial t} + u.\nabla\omega = \nu\Delta_p\omega, \quad \text{pour} \quad 1 < p < 2, \tag{3.10}$$

il est à noter que cette équation est l'équation de Navier Stokes en variables secondaires, en l'occurrence tourbillon-courant qui est assurée par la condition de divergence nulle sur u, voir[35] . En (3.10), on a remplacé la diffusion isotrope dans [14] par une diffusion P-Laplacienne, avec :

$$\omega = \Delta I, \tag{3.11}$$

et

$$u = \nabla^{\perp} I.$$

Comme l'image I à traiter est donnée à l'instant $t = 0$, on la note par I_0, donc on peut compléter notre système d'équations par la donnée initiale :

$$\omega_0 = \Delta I_0 \quad \text{sur} \quad \Omega, \tag{3.12}$$

de même pour les conditions aux bords :

$$\omega|_{\partial\Omega} = \Delta I|_{\partial\Omega} = g \quad \text{sur} \quad \partial\Omega. \tag{3.13}$$

Notre technique consiste à faire évoluer l'équation (3.10) jusqu'à l'équilibre, c-à-d l'état stationnaire, puis on procède à une résolution simultanée de (3.11). L'existence de la solution sera donnée dans la section suivante, ainsi que la discrétisation et la stabilité du schéma numérique utilisé.

Remarque 3.3.1. *Dans ce modèle, on peut remplacer l'équation (3.11), qui nécessite une inversion de l'opérateur Laplacien, par l'équation d'évolution suivante :*

$$\frac{\partial I}{\partial t} = \alpha(\Delta I - \omega). \tag{3.14}$$

Avec comme condition initiale I_0 et au bord $I|_{\partial\Omega}$, on fait évoluer (3.14) vers son état stationnaire, et l'image I solution de (3.11) sera obtenue sans passer par l'inversion du Laplacien.

3.3.2 Existence de la solution

On considère le problème (P_1) avec conditions non homogène aux limites :

$$\begin{cases} \frac{\partial \omega}{\partial t} + u.\nabla\omega - \nu\Delta_p\omega &= 0 \quad \text{sur} \quad Q_T, \\ \omega(0,.) &= \omega_0 \quad \text{sur} \quad \Omega, \\ \omega(t,x) &= g \quad \text{sur } \Gamma, \end{cases} \tag{3.15}$$

avec : $Q_T = (0,T)\times\Omega$ et $\Gamma = (0,T)\times\partial\Omega$, et on considère le problème (P_2) avec condition homogène aux limites :

$$\begin{cases} \frac{\partial \omega}{\partial t} + u.\nabla\omega - \nu\Delta_p\omega &= 0 \quad \text{sur} \quad Q_T, \\ \omega(0,.) &= \omega_0 \quad \text{sur} \quad \Omega, \\ \omega(t,x) &= 0 \quad \text{sur } \Gamma. \end{cases} \tag{3.16}$$

On a le thèorème suivant :

Théorème 3.3.1. *Soit $w_0 \in L^2(\Omega)$,où Ω un ouvert borné de $\mathbb{R}^2$ de frontière $\partial\Omega$ lipchitzienne, si $|\nabla w| < 1$. Le problème (P_2) admet une solution unique $\omega \in L^p(0,T,W_0^{1,p}(\Omega))$, telle que $\omega(.,0) = \omega_0$ pour $1 < p < +\infty$.*

Preuve. On définie la solution faible du problème (P_2) par :

$$(\frac{\partial\omega}{\partial t}, v) + (u.\nabla\omega, v) - \nu(\Delta_p\omega, v) = 0, \tag{3.17}$$

où $L^P(0,T,W_0^{1,p})' = L^q(0,T,W_0^{-1,q})$ avec $W_0^{-1,q} = (W_0^{1,p})'$ et $\frac{1}{q} + \frac{1}{p} = 1$.
Nous cherchons la suite $\{\omega_m\}$ vérifiant pour tout $v \in L^P(0,T,W_0^{1,p})$:

$$(\frac{\partial\omega_m}{\partial t}, v) + (u.\nabla\omega_m, v) - \nu(\Delta_p\omega_m, v) = 0, \tag{3.18}$$

c'est-à-dire :

$$\int_{Q_T} \frac{\partial\omega_m}{\partial t} v dxdt + \int_{Q_T} u.\nabla\omega_m v dxdt - \nu\int_{Q_T} \Delta_p\omega_m v dxdt = 0. \tag{3.19}$$

On prend $v = \omega_m$, on obtient :

$$\int_{Q_T} \frac{\partial\omega_m}{\partial t}\omega_m dxdt + \int_{Q_T} u.\nabla\omega_m\omega_m dxdt - \nu\int_{Q_T} \Delta_p\omega_m\omega_m dxdt = 0. \tag{3.20}$$

Par intégration par partie, et comme $\omega_m \in L^p(0,T,W_0^{1,p})$, on obtient :

$$\int_{Q_T} \frac{\partial \omega_m}{\partial t}\omega_m dxdt + \int_{Q_T} (u.\nabla\omega_m)\omega_m dxdt + \nu \int_{Q_T} |\nabla\omega_m|^{p-2}\nabla\omega_m.\nabla\omega_m dxdt = 0. \quad (3.21)$$

Et comme :

$$\begin{array}{rcl} \int_{Q_T} \frac{\partial \omega_m}{\partial t}\omega_m dxdt & = & \frac{1}{2}\int_{Q_T} 2\frac{\partial \omega_m}{\partial t}\omega_m dxdt \\ & = & \frac{1}{2}\int_{Q_T} \frac{\partial}{\partial t}(\omega_m)^2 dxdt \\ & = & \frac{1}{2}\int_{\omega}\int_{(0,T)} \frac{\partial}{\partial t}(\omega_m)^2 dt)dx \\ & = & \frac{1}{2}\int_{\Omega} \omega_m^2(x,\tau)dx - \frac{1}{2}\int_{\Omega}\omega_m^2(x,0)dx, \end{array} \quad (3.22)$$

et :

$$\begin{array}{rcl} \int_{\Omega}(u.\nabla\omega_m)\omega_m dx & = & \int_{\Omega}(\omega_m u.\nabla\omega_m dx \\ & = & \int_{\Omega} u.\omega_m\nabla\omega_m dx \\ & = & \frac{1}{2}\int_{\Omega} u.\nabla(\omega_m)^2 dx \\ & = & -\frac{1}{2}\int_{\Omega} div(u)(\omega_m)^2 dx \\ & = & 0, \end{array} \quad (3.23)$$

d'où :$\int_{Q_t}(u.\nabla\omega_m)\omega_m dxdt = 0$. Pour le dernier terme dans l'équation (3.21), on a :

$$\begin{array}{rcl} \nu\int_{Q_T} |\nabla\omega_m|^{p-2}\nabla\omega_m.\nabla\omega_m dxdt & = & \nu\int_{Q_T}|\nabla\omega_m|^{p-2}|\nabla\omega_m|^2 dxdt \\ & = & \nu\int_{Q_T}|\nabla\omega_m|^p dxdt, \end{array} \quad (3.24)$$

de ce qui précède, résulte que :

$$\frac{1}{2}\int_{\Omega}\omega_m^2(x,\tau)dx + \nu\int_{Q_T}|\nabla\omega_m|^p dxdt = \frac{1}{2}\int_{\Omega}\omega_m^2(x,0)dx \quad (3.25)$$

d'où : $|\omega_m|_{L^2(\Omega)} = (\int_{\Omega}\omega_m^2(x,\tau)dx)^{\frac{1}{2}} \leq C_1$, et du fait que $|\nabla\omega_m| < 1$, on obtient :

$$\begin{array}{rcl} |\nabla\omega_m| < 1 & \Rightarrow & |\nabla\omega_m|^p < 1 \\ & \Rightarrow & \int_{Q_T}|\nabla\omega_m|^p dxdt < \int_{Q_t} 1dxdt \\ & \Rightarrow & \int_{Q_T}|\nabla\omega_m|^p dxdt < mes(Q_t) \\ & \Rightarrow & \int_{Q_T}|\nabla\omega_m|^p dxdt < T.mes(\Omega) \\ & \Rightarrow & \int_{Q_T}|\nabla\omega_m|^p dxdt < C \end{array} \quad (3.26)$$

d'où : $\|\nabla\omega_m\|_{L^p(Q_T)} = (\int_{Q_T} |\nabla\omega_m|^p dxdt)^{\frac{1}{p}} \leq C_2$, ce qui permet de de conclure que la suite $\{\omega_m\}$ demeure bornée dans $L^\infty(0, T, W_0^{1,p}(\Omega))$ et $L^p(0, T, W_0^{1,p}(\Omega))$, donc on peut extraire une sous suite de $\{\omega_m\}$, noté encore $\{\omega_m\}$, qui converge faiblement vers ω solution du problème (P_2) dans $L^p(0, T, W_0^{1,p}(\Omega))$. Pour avoir la convergence forte, on a pour $q < p$:

$$\begin{aligned} \int_{Q_T} |\nabla\omega_m|^q dxdt &= \int_{|\nabla\omega_m|<1} |\nabla\omega_m|^q dxdt + \int_{|\nabla\omega_m|\geq 1} |\nabla\omega_m|^q dxdt \\ &\leq mes(Q_T) + C \leq M \end{aligned} \tag{3.27}$$

ce qui implique que $\{\nabla\omega_m\}$ converge $p.p$ par application du théorème de Vitalli (voir[45, 32]), et on conclut que :

$$\{\omega_m\} \to \omega \text{ dans } L^p(0, T, W_0^{1,p}(\Omega))$$
$$\text{et } \{\nabla\omega_m\} \to \nabla\omega \text{ dans } L^q(0, T, W_0^{-1,q}(\Omega)).$$

Et pour montrer que $\omega(.,0) = \omega_0$ sur Ω, on a $\omega \in L^\infty(0, T, L^2(\Omega))$ et on vérifie que $\frac{\partial\omega}{\partial t} \in L^\infty(0, T, L^2(\Omega))$, d'où : $\omega \in C(0, T) \to L^2$. $\square$

Pour prouver l'existante de la solution pour le problème (P_1), on considère la fonction ζ vérifiant :

$$\begin{cases} \zeta \in L^p(0, T; W^{1,p}(\Omega)), \quad \frac{\partial\zeta}{\partial t} \in L^q(0, T; W^{-1,q}(\Omega)) \\ \zeta = g \quad \text{sur} \quad \Gamma. \end{cases} \tag{3.28}$$

On introduit la fonction $\Theta = \omega - \zeta$ qui vérife $\Theta = 0 \quad \text{sur} \quad \Gamma$, donc tout revient à charcher ξ solution du problème homogène (P_3) donné par :

$$\begin{cases} \frac{\partial\Theta}{\partial t} + u.\nabla\Theta - \nu\Delta_p(\Theta + \zeta) = -\frac{\partial\zeta}{\partial t} - u.\nabla\zeta = f(\in L^q(0, T; W^{-1,q}(\Omega))), \\ \Theta \in L^p(0, T; W_0^{1,p}) \\ \Theta(0) = \omega_0 - \zeta_0 \in L^2(\Omega). \end{cases} \tag{3.29}$$

On adaptera le raisonnement précédent sur le problème (P_2) pour montrer l'existence de la solution du problème (P_3).

Remarque 3.3.2. – *Sur les conditions que doit vérifier g pour que ζ vérifiant (3.29) existe, on peut consulter [38].*

– *Pour l'unicité de la solution de (3.11) sous les conditions du* ***Théorème 3.1.1****, on peut utiliser le théorème de Lions dans [44], l'existence ainsi que l'unicité de la solution de l'équation de poisson (3.11) est un résultat classique.*

3.4 Mise en œuvre numérique

3.4.1 Discrétisation du modèle

On suppose que la région d'inpainting Ω est rectangulaire, et on considère une discrétisation basée sur les différences finies, on considère également un schéma centré en espace et un schéma explicite en temps (Forward-Time Central-Space **FTCS**), avec un choix approprié du pas spatiale $dx = dy = 1$, qu'on lui associe un pas de temps dt convenable, selon des critère de stabilité.

Discrétisation de l'équation principale (3.10)

Si on discrétise séparément le terme de diffusion dans l'équation (3.10), on obtient :

$$\omega_{i,j}^{n+1} = \omega_{i,j}^{n} + \Delta t\Big[- u_x^n D_x^0 \omega_{i,j}^n - u_y^n D_y^0 \omega_{i,j}^n + \nu\{\widetilde{D}_P \omega_{i,j}^n\}\Big],$$

où :

$$\omega_{i,j}^n = \omega(ndt, i, j)$$

D_x^0 et D_y^0 sont les approximations centrées de la première dérivée par rapport à x et y, données par :

$$D_x^0 \omega_{i,j}^n = \frac{(\omega_{i+1,j}^n - \omega_{i-1,j}^n)}{2},$$

et

$$D_y^0 \omega_{i,j}^n = \frac{(\omega_{i,j+1}^n - \omega_{i,j-1}^n)}{2}.$$

$\widetilde{D}_P$ est l'approximation de Δ_p , pour l'obtenir on commence par réécrire $\Delta_p\omega$ comme suit :

$$\Delta_p \omega = \partial_x(|\nabla\omega|^{p-2}\omega_x) + \partial_y(|\nabla\omega|^{p-2}\omega_y), \tag{3.30}$$

en discrétisant (3.30), on obtient :

$$\widetilde{D}_P\omega_{i,j} = \begin{cases} D_x^- \frac{D_x^+\omega_{i,j}}{(|(D_x^+\omega_{i,j})|^2+|(D_y^+\omega_{i,j})|^2+\delta)^{\frac{2-p}{p}}} \\ +D_y^- \frac{D_y^+\omega_{i,j}}{(|(D_x^+\omega_{i,j})|^2+|(D_y^+\omega_{i,j})|^2+\delta)^{\frac{2-p}{p}}}; \end{cases} \tag{3.31}$$

où δ est un paramètre de régularisation qui permet d'éviter la division par zéro quand $|D_x^+\omega_{i,j}|^2 + |D_y^+\omega_{i,j}|^2$ tend vers zéro ; en d'autres termes, au voisinage des pixels qui ont un gradient nul.

Les opérateurs D_x^-, D_x^+, D_y^- et D_y^+ sont les différences en avant et en arrière, par rapport à x et y respectivement, données par :

$$D_x^-\omega_{i,j} = \omega_{i,j} - \omega_{i-1,j},$$

$$D_x^+\omega_{i,j} = \omega_{i+1,j} - \omega_{i,j},$$

$$D_y^-\omega_{i,j} = \omega_{i,j} - \omega_{i,j-1},$$

et

$$D_y^+\omega_{i,j} = \omega_{i,j+1} - \omega_{i,j}.$$

Pour résoudre l'équation (3.11), on applique une formule de différence finie centrée standard, ce qui nous permet d'obtenir :

$$(I_{i+1,j}^n + I_{i-1,j}^n) + (I_{i,j+1}^n + I_{i,j-1}^n) - 4I_{i,j}^n = \omega_{i,j}^n, \tag{3.32}$$

qui s'écrit sous la forme :

$$A\overrightarrow{I^n} = \overrightarrow{\omega^n}, \tag{3.33}$$

où A est la matrice associée à (3.32).

3.4.2 Les conditions aux bords

Pour achever la discrétisation du système, des conditions aux bords sur I et ω sont nécessaires. Ces conditions sont différentes, dès que l'image I est connue dans $\partial\Omega$, on utilise une condition de Dirichlet pour I sur $\partial\Omega$; de plus l'image I est connue même au delà de $\partial\Omega$ et non seulement sur les frontières de la zone d'inpainting, étant donnée que la zone d'inpainting est à l'intérieur de l'image et n'intersecte pas ses frontières. Pour discrétiser les valeurs de ω sur $\partial\Omega$, on considère les étapes suivantes :

- Discrétisation de $U = -I_y$ et $V = I_x$ sur $\partial\Omega$ et dans Ω, en utilisant une formule de différence finie centrée si c'est possible et une formule décentrée à trois points le cas contraire, où I_x et I_y sont les dérivées premières par rapport à x et y respectivement.
- Discrétisation de U_y et V_x en utilisant les valeurs de U et V obtenues précédemment, où U_y et V_x sont les dérivées premières de U et V par rapport à y et x respectivement.
- La discrétisation de ω est obtenue en remarquant que :

$$\begin{aligned}\omega = \Delta I &= \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \\ &= \frac{\partial(\partial I)}{\partial x^2} - \frac{\partial(-\partial I)}{\partial y^2} \\ &= \frac{\partial V}{\partial x} - \frac{\partial U}{\partial y} \\ &= V_x - U_y;\end{aligned}$$

Notons que l'utilisation d'une formule décentrée à trois points avec une formule de différences finies centrée maintient un second ordre d'approximation dans les conditions aux bords. Par " formule décentrée à trois points" on désigne :

$$U_j' = 3U_j - 4U_{j-1} + U_{j-2},$$

ou :

$$U_j' = -3U_j + 4U_{j+1} - U_{j+2}.$$

3.4.3 Stabilité et implémentation numérique

Stabilité

En accord avec le choix simplifié du pas spatiale égale à $\Delta x = \Delta y = 1$, on adoptera une condition de stabilité sur Δt (Voir [7]) donné par :

$$\Delta t \leq min\big(\sqrt{2}/3, \sqrt{8\nu}/3\big)$$

Implémentation numérique

L'implémentation numérique de notre schéma est faite selon l'algorithme suivant : **Algorithme** :

- On initie ω sur Ω et $\partial\Omega$.
- On applique des formules de différences finies pour évaluer les approximations des dérivées premières ω_x et ω_y en chaque point intérieur de la zone d'inpainting Ω.
- Nous évaluons $\partial_x(|\nabla\omega|^{p-2}\omega_x) + \partial_y(|\nabla\omega|^{p-2}\omega_y)$ selon la formule (3.31).
- On détermine ω selon (3.10). Les étapes sont itérées jusqu'à l'obtention de l'état stationnaire de l'équation (3.10).
- Nous résolvons le système (3.33) pour obtenir I.

3.5 Résultats numériques

Dans cette section, on présente les résultats numériques obtenus avec le modèle proposé. Pour commencer, on applique le modèle à deux images synthétiques de taille 400×400, l'une est masquée avec un carrée blanc de taille 28×28, qui illustre le cas ou le domaine est régulier, voir figure 3.13 et l'autre image est masquée avec un quart de cercle de même taille, qui illustre le cas où le domaine est irrégulier, voir figure 3.12.

Dans le but de tester le modèle proposé et montrer l'impacte du choix du paramètre p sur les résultats obtenus, nous visualisons l'amélioration sur le PSNR pour différentes valeurs de p, voir figure 3.14 Dans le deuxième cas, on a appliqué le modèle sur des images

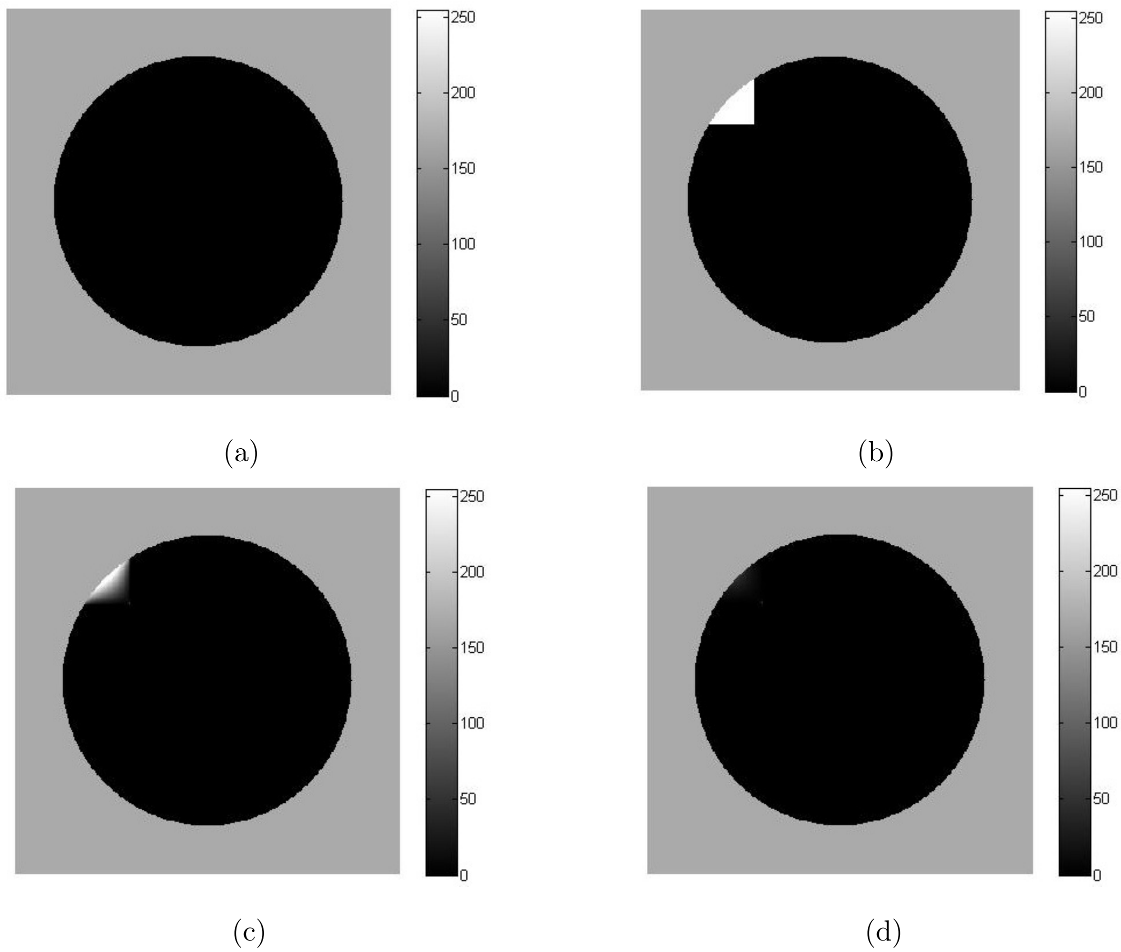
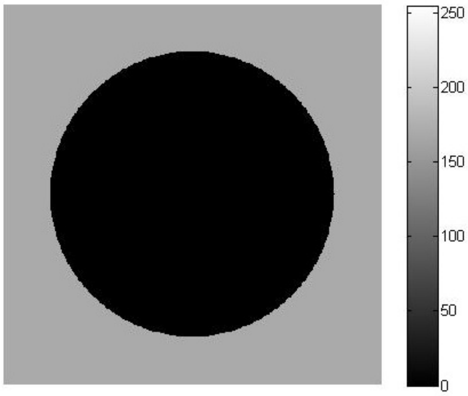


FIGURE 3.12 – Résultat d'inpainting dans un domaine irrégulier, l'algorithme est lancé avec un pas de temps $\Delta t = 10^{-4}$, viscosité $\nu = 2$, $p = 1.8$: (a) l'image originale , (b) image masquée avec un quart de disque blanc, (c) et (d) représentent les images restaurées après 5 et 20 itérations, respectivement, on a : (cputime= 55.2712 s, PSNR=34.4707 pour 20 itérations).

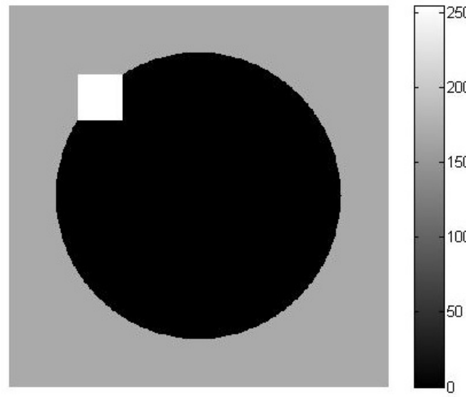
réelles pour supprimer un texte indésirable dans l'image, voir figure (3.15).

Remarque 3.5.1. *Toutes les compilations ont été fait en utilisant Matlab 7.0 sur un micro personnel 3GHz, Core I3 avec 3Ghz de RAM.*

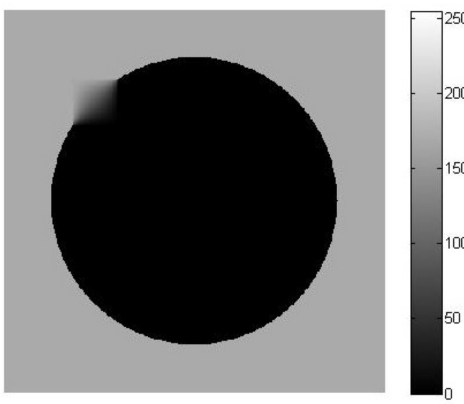
Deux cas limites se présentent :



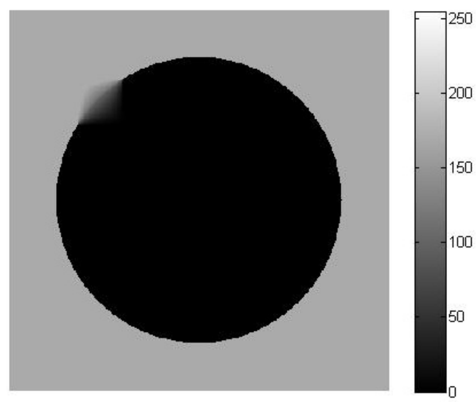
(a)



(b)



(c)



(d)

FIGURE 3.13 – Résultat d'inpainting dans un domaine régulier, l'algorithme est lancé avec un pas de temps $\Delta t = 10^{-4}$, viscosité $\nu = 2$, $p = 1.8$: (a) l'image originale, (b) image masquée avec un carré blanc, (c) et (d) représentent les images obtenues après 5 et 20 itérations, respectivement. (cputime=68.3908 s, PSNR=33.2645 pour 20 itérations).

Quand p tend vers 1

Dans le cas où p tend vers 1, le P-Laplacien sera donné par : $\Delta_1 u = |\nabla|^{-1} u_{\xi\xi}$, ce qui montre que pour la variation totale (TV), le processus de remplissage, ou de reconstruction des parties endommagées se fait uniquement le long des contours ; c-à-d parallèlement à la tangente, ce qui préserve bien les contours, mais dans ce cas le modèle souffre de l'inconvénient d'une diffusion directionnelle dans les régions qui ont seulement un fort gradient, ce qui ne permet pas de réduire l'effet d'échelle dans l'image, qui peut être éliminé seulement dans le cas ou la diffusion est dans toutes les directions.

Quand p tend vers 2

Dans le cas où p tend vers 2, le P-Laplacien s'écrit : $\Delta_2 u = u_{\xi\xi} + u_{\eta\eta}$, donc les coefficients de diffusion selon les deux directions, tangentielle et perpendiculaire, sont égales ce qui génère un processus de diffusion isotrope ce qui présente un inconvénient majeur, en effet dans les zones d'une intensité homogène, le processus permet de réduire efficacement les effets du bruit, mais pour les zones où des discontinuités se présentent, le processus génère des zones floues et les contours désignant les limites des éventuels objets dans l'image seront considérablement abimés ; donc une perte dans les propriétés de l'image comme dans Bertalmio et al (2001) pour $p = 2$ et $g = 1$, figure (3.14b). En effet, on remarque que pour un choix de valeur du paramètre p compris entre 1 et 2, nous pouvons préserver les discontinuités intéressantes et significatives dans l'image, tout en évitant la création des blocs avec une intensité constante dans l'image résultante ; en d'autre terme diminuer le plus possible l'effet d'échelle dans l'image après le processus d'inpainting. Ces résultats ont fait l'objet de deux publications (Voir [49] et [48]) . D'autres modèles ont été étudiés, en l'occurrence une équation intégro-différentielle :

3.6 Décomposition de l'image par une équation intégro-différentielle

Le modèle qu'on vient de présenter dans ce chapitre, assure un processus d'inpainting sur une classe d'images purement géométriques, qui ne contiennent pas de texture. Afin d'élargir le champs d'applicabilité du modèle sur des images texturées, on propose dans cette section un modèle basé sur l'utilisation d'une équation intégro-différentielle, qui sépare la partie texturée de l'image de la partie géométrique afin de pouvoir appliquer le procédé d'inpainting séparément sur chaque composante (Dans l'annexe A, le lecteur trouvera un bref état d'art sur les méthodes de décomposition) :

3.6.1 Nouvelle équation intégro-différentielle

On s'inspire de [59, 61, 62], on propose une nouvelle équation intégro-différentielle qui assure une décomposition d'images sous la forme $[BV(\Omega), G(\Omega)]$, où $G(\Omega)$ est l'espace de Meyer, qui est prouvé dans [50] d'être le mieux adapté pour l'extraction de texture.

Approche proposée

On considère le problème de minimisation :

$$\inf_{u \in BV(\Omega)} \int_\Omega |\nabla u| dx + \lambda \|f - u\|^2_{H^{-1}(\Omega)}, \tag{3.34}$$

et sachant que : $\|u\|^2_{H^{-1}(\Omega)} = \int_\Omega |\nabla(\Delta^{-1} v) dx$, la minimisation de (3.34) nous conduit a la résolution de l'équation d'Euler-Lagrange associée suivante :

$$u = f - \frac{1}{2\lambda} \Delta \left[div \left(\frac{\nabla u(x,t)}{|\nabla u(x,t)|} \right) \right],$$

avec comme conditions initiales :

$$\frac{\partial u}{\partial n}|_{\partial\Omega} = 0,$$

et

$$\frac{\partial}{\partial n} \left(div \left(\frac{\nabla u(x,t)}{|\nabla u(x,t)|} \right) \right) |_{\partial\Omega} = 0.$$

Par application d'une analyse hiérarchique similaire à [59, 61, 62], on obtient l'équation intégro-différentielle suivante :

$$\int_0^t u(x,s)ds = f(x) - \frac{1}{2\lambda(t)}\Delta\left[div\left(\frac{\nabla u(x,t)}{|\nabla u(x,t)|}\right)\right], \tag{3.35}$$

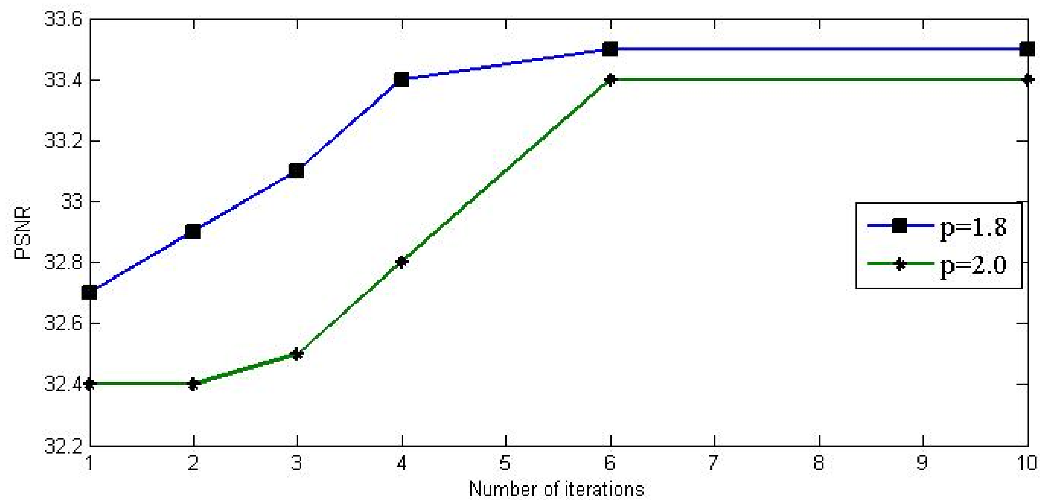
Implémentation numérique et résultat

Soit Δt le pas de temps et U^{n+1} la solution approchée de (3.35) à l'instant $t = (n+1)\Delta$, on a :

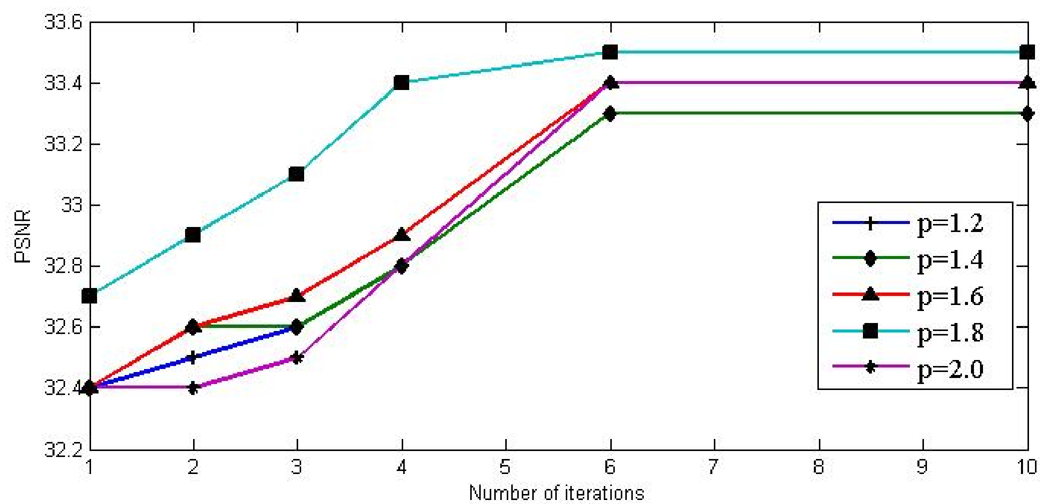
$$U^{n+1} = U^{n+1} + \omega^{n+1} \quad \text{et} \quad \omega_{i,j}^{n+1} = u_{i,j}^{n+1}\Delta t,$$

avec : $u^{n+1}(i,j) = u_{i\Delta x, j\Delta y}^{n+1}$ est l'approximation de u^{n+1} au point $(i\Delta x, j\Delta y)$.

Pour le schéma numérique, on adopte la discrétisation proposée dans [62]. Les figures : 3.19, 3.20 et 3.21 présentent des applications de notre modèle sur des images réelles.

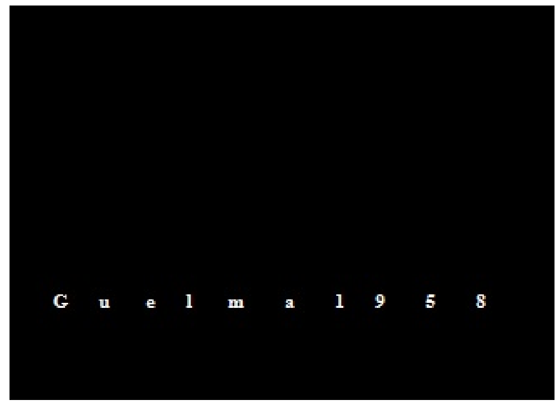


(a)



(b)

FIGURE 3.14 – Le PSNR en fonction de nombre d'itérations (a) le résultat optimum (expérimentalement) pour $p = 1.8$ (b) Comparaison avec les résultats de l'approche proposée dans [14] caractérisée par $p = 2$



(a) (b)

(c) (d)

FIGURE 3.15 – Suppression d'une écriture sur une image réelle : (a) image originale, (b) masque appliqué à l'image, (c) image détériorée et (d) image obtenue après inpainting.


(a) (b)

(c)

FIGURE 3.16 – Suppression des grandes taches sur une image réelle : (a) image originale, (b) image détériorée et (c) image reconstruite par l'algorithme d'inpainting proposé.

FIGURE 3.17 – Un zoom sur les parties reconstruites de l'image par l'inpainting.

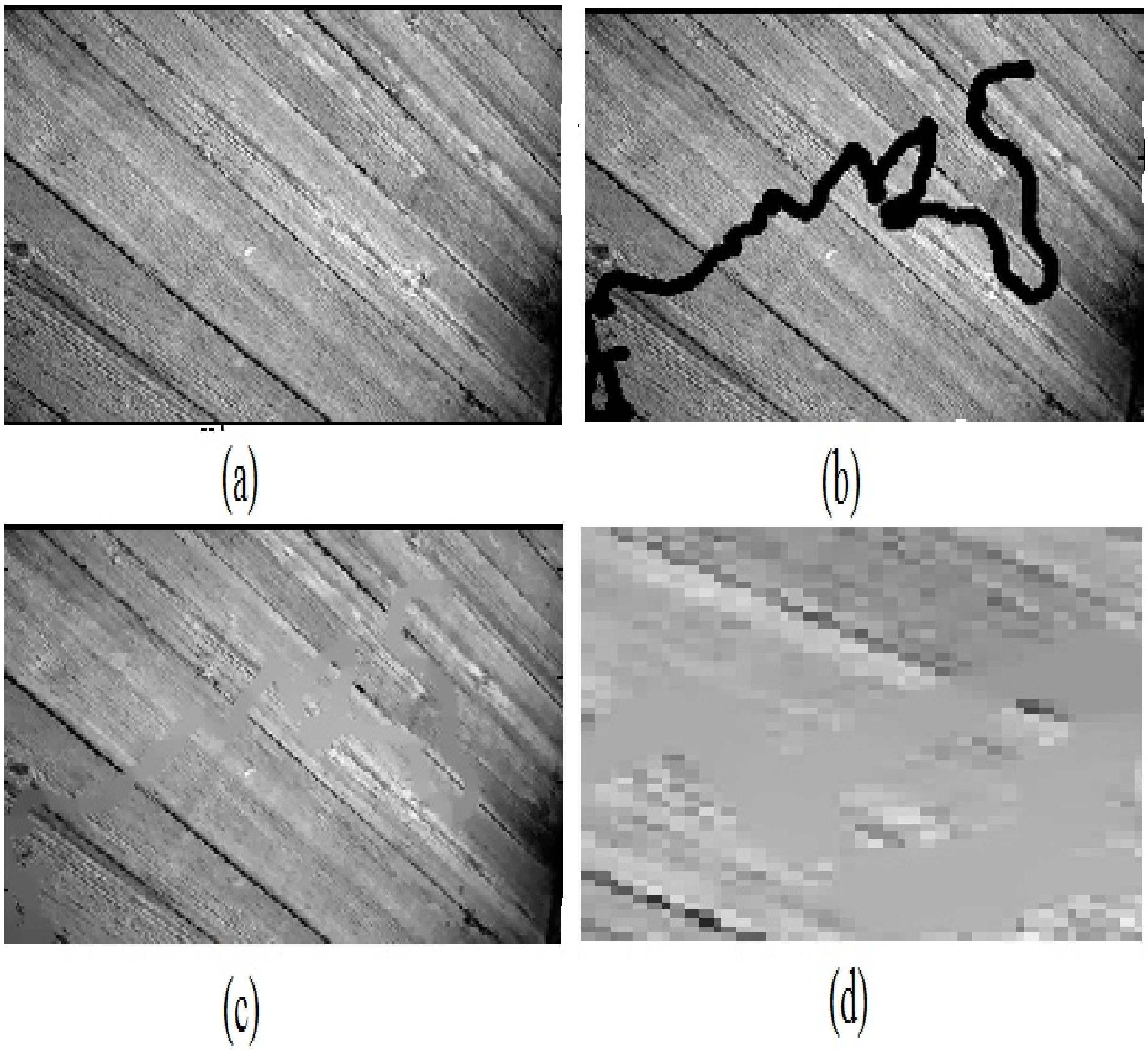


Figure 3.18 – Suppression d'un griffonnage sur une image réelle : (a) image originale, (b) image griffonnée, (c) image obtenue après inpainting et (d) un zoom de 400% sur une partie de la zone reconstruite de l'image (c)

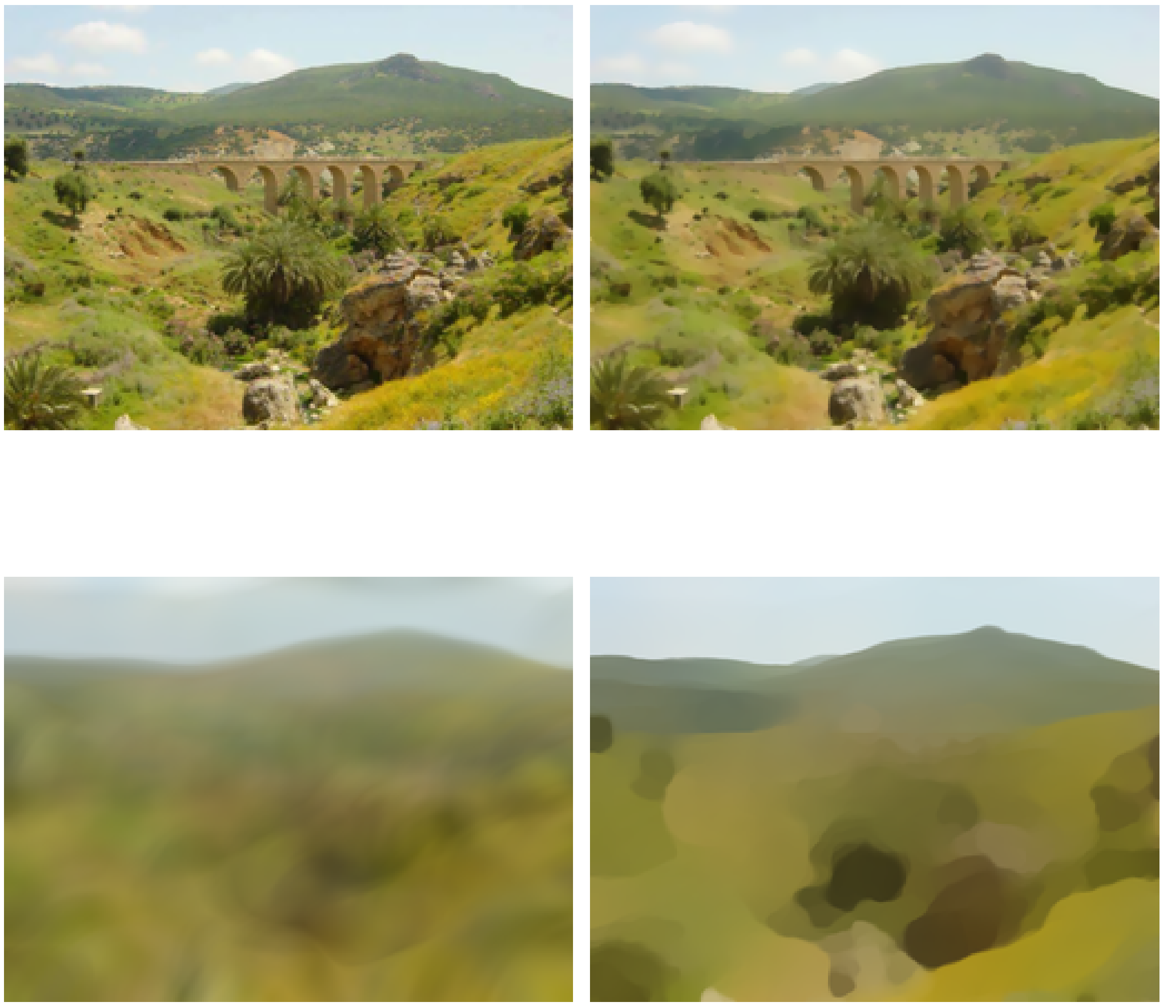


FIGURE 3.19 – Exemple 1 de l'application de la décomposition par l'équation intégro-différentielle (3.35)

FIGURE 3.20 – Exemple 2 de l'application de la décomposition par l'équation intégro-différentielle (3.35).

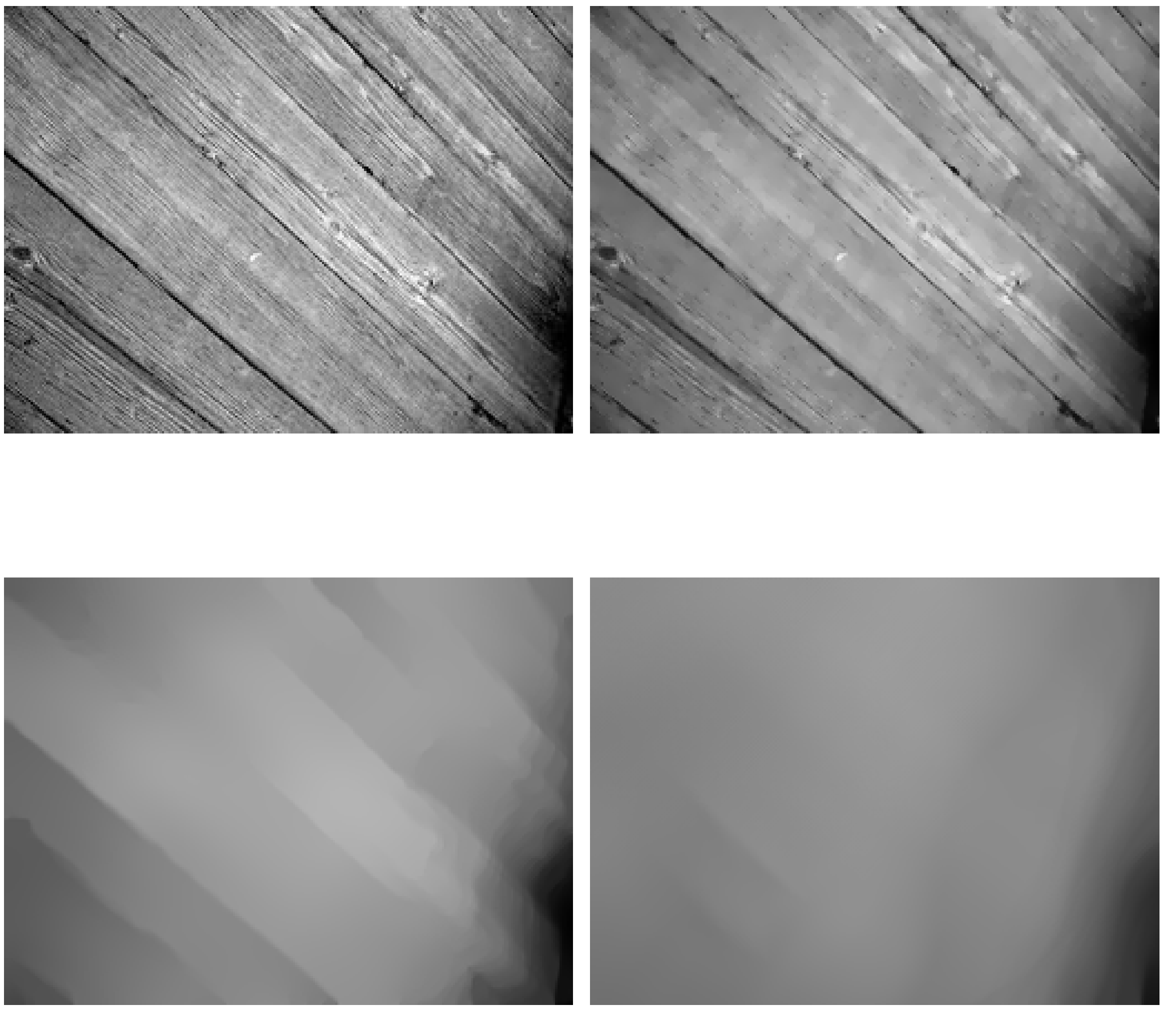


FIGURE 3.21 – Exemple 3 de l'application de la décomposition par l'équation intégro-différentielle (3.35).

Conclusion et perspectives

Conclusion et perspectives

Conclusion

Dans ce travail, on a proposé un nouveau modèle est basée principalement sur une version des équations de Naviers-Stokes en formulation courant-tourbillon, avec un terme de diffusion non-linéaire basé sur l'utilisation du p-Laplacien. Un algorithme numérique est proposée et élaboré, pour réaliser d'une manière automatique, la tâche d'inpainting sur des images réelles et synthétiques en niveau de gris.

L'algorithme numérique utilise les schémas de différences finies pour la discrétisation du système d'équations obtenues, une méthode qui permet d'effectuer des opérations d'inpainting sur des régions arbitraire de l'image, rectangulaire ou non rectangulaire.

Un résultat de stabilité du schémas numérique pour le modèle proposé a été donnée. Elle nous a permet de déterminer un maximum sur le pas de temps utilisé. L'existence de la solution a été démontré.

On a implémenté la méthode pour certaines images en niveau de gris (réelles et synthétiques). Les résultats obtenus confirment certains avantages de l'approche proposée :

1. l'absence de toute restriction sur la régularité du domaine d'inpainting, ;
2. le temps d'exécution "Cputime" est faible en le comparant avec d'autre méthodes.
3. Pour évaluer la qualité de l'image obtenue, on a comparé avec l'image originale supposée disponible, le PSNR montre la performance de la méthode numérique utilisée ;

4. Un aspect unificateur est présent dans ce modèle, car il regroupe les deux méthodes classiques, celle de la variation totale (pour $p = 1$) et le modèle de Bertalmio (pour $p = 2$). Pour $p = 1.8$, on a constaté, d'une manière heuristique, qu'il s'agit du meilleur compromis entre les deux méthodes signalées précédemment.

Il est essentiel de noter que la méthode proposée, diffuse localement l'information vers la zone d'inpainting, ce qui n'est pas souhaitable pour les images texturées voir figure (3.18). Par une diffusion locale, nous pouvons prolonger uniquement les lignes de niveaux qui touchent la zone d'inpainting, ce qui est suffisant pour les images géométriques, mais ce processus ne permet pas de reconstruire les contours fermés à l'intérieur de la zone d'inpainting elle même ; d'où la nécessité de développer des techniques non-locales plus élaborées. Ce qui nous a conduit à proposer une équation intégro-différentielle qui permet de décomposer l'image en deux parties distinctes : partie texturée et partie géométrique. Les premiers résultats montre la performance de cette décomposition.

Perspectives

Les deux résultats obtenus par les deux modèles proposés dans cette thèse ont aboutit à des questions ouvertes, on peut citer par exemple l'application de ces modèles sur des images texturées, où l'opération d'inpainting avec des EDP engendre des dégradations sur la qualité de l'image. Développer des techniques spécifique aux images texturées révèle très important pour une meilleur restauration d'images. Une piste à explorer consiste à combiner la méthode de décomposition telle qu'elle est présentée dans cette thèse, et à appliquer le processus d'inpainting sur chaqu'une des deux composantes obtenues. Notons que l'élaboration d'un modèle d'inpaiting pour la partie texturée de l'image décomposée est cruciale, utiliser les blocs des similarité est piste aussi à investir.

Annexe A

Annexe

A.1 État de l'art pour la décomposition de l'image

Il existe plusieurs modèles permettant l'extraction de texture, nous nous contentons de présenter ici quelques modèles :

A.1.1 Modèle de Rudin, Osher et Fatemi

Le modèle de Rudin et al [55] est une alternatif à la régularisation de Tychonov, ce modèle consiste à diviser l'image f en deux parties $u+v$ où u est la partie régulière et v le bruit, la solution dans ce modèle est obtenue en minimisant :

$$\int_{\Omega} |\nabla u| dx + \lambda \|v\|_{L^2}^2, \tag{A.1}$$

avec $\lambda > 0$. Si on pose $v = f - u$, on obtient le problème de minimisation suivant :

$$\underset{u \in BV}{inf} \{F(u) = \int_{\Omega} |\nabla u| + \lambda \|f-u\|_{L^2}^2\}, \tag{A.2}$$

ce qui conduit à une décomposition $[BV(\Omega), L^2(\Omega)]$ de l'image f.

Théorème A.1.1. *Pour une image $f \in L^2(\Omega)$, le problème (A.2) admet une solution unique.*

Preuve. Voir [55]. □

Remarque A.1.1. *La solution de (A.2) est obtenu en résolvant l'équation d'Euler-Lagrange suivante :*

$$u = f + \frac{1}{\lambda} div \left(\frac{\nabla u}{|\nabla u|} \right). \tag{A.3}$$

Pour des valeurs différentes de λ, l'équation (A.3) génère une série d'images $(u_\lambda)_{\lambda>0}$ formant une représentation multi-échelle de l'image f.

A.1.2 Modèle de Tadmor et al

Tadmor et al, dans [59, 61, 62] présentent une nouvelle représentation multi-échelle de l'image observée f en considérant le modèle A.2 comme point de départ :

$$f = \tau u_{\lambda_0} + v_{\lambda_0}, \quad [u_{\lambda_0}, v_{\lambda_0}] := \underset{f=\tau u+v}{inf} \{\|u\|_{BV} + \frac{\lambda_0}{\tau}\|v\|_{L^2}^2\}, \tag{A.4}$$

Comme la partie v_{λ_0} peut contenir encore de la texture, Tadmor et al essaye de la décomposer par le même procédé, mais avec une nouvelle valeur de $\lambda = \lambda_1 > \lambda_0$, ce qui permet d'obtenir :

$$v_{\lambda_0} = \tau u_{\lambda_1} + v_{\lambda_1}, \quad [u_{\lambda_1}, v_{\lambda_1}] := \underset{v_{\lambda_0}=\tau u+v}{inf} \{\|u\|_{BV} + \frac{\lambda_1}{\tau}\|v\|_{L^2}^2\}. \tag{A.5}$$

On répète le même procédé pour $\lambda_0 < \lambda_1 < \lambda_2...$:

$$v_{\lambda_{j-1}} = \tau u_{\lambda_j} + v_{\lambda_j}, \quad [u_{\lambda_j}, v_{\lambda_j}] := \underset{v_{\lambda_{j-1}}=\tau u+v}{inf} \{\|u\|_{BV} + \frac{\lambda_j}{\tau}\|v\|_{L^2}^2\}, \tag{A.6}$$

d'où :

$$\begin{aligned} f &= \tau u_{\lambda_0} + v_{\lambda_0} \\ &= \tau u_{\lambda_0} + \tau u_{\lambda_1} + v_{\lambda_1} \\ &= ... \\ &= \tau u_{\lambda_0} + \tau u_{\lambda_1} + ... + \tau u_{\lambda_N} + v_{\lambda_N}. \end{aligned}$$

De ce qui précède, on en déduit :

$$\sum_{j=0}^{N} u_{\lambda_j}\tau = f - v_{\lambda_N}, \tag{A.7}$$

de (A.3) et (A.7), on obtient

$$\sum_{j=0}^{N} u_{\lambda_j}\tau = f + \frac{1}{2\lambda_N} div\left(\frac{\nabla u}{|\nabla u|}\right). \tag{A.8}$$

La description hiérarchique (A.8) motive les auteurs d'adopter une nouvelle représentation multi-échelle de l'image f comme solution de l'équation intégro-différentielle suivante :

$$\int_0^t u(x,s)ds = f(x) + \frac{1}{2\lambda_t} div\left(\frac{\nabla u(x,t)}{|\nabla u(x,t)|}\right). \tag{A.9}$$

La représentation multi-échelle dans ce cas est :

$$\left(U(.,t) := \int_0^t u(.,s)ds\right)_{t\geq 0}$$

$u(x,t)$ représente la vitesse avec laquelle l'image $U(t)$ varie avec le temps. La figure $A.1$ donne un exemple de l'application de cette décomposition.

A.1.3 Modèle de Meyer

Yves Meyer a analysé de nombreux modèles de décomposition $f = u+v$, et notamment le modèle de Rudin et al [55], mettant en avant le fait que la norme L^2 n'est pas le meilleur choix pour la capture de la partie oscillante d'une image. C'est pourquoi il a introduit dans [50] un espace a priori mieux adapté : l'espace G des fonctions oscillantes.

A.1.4 L'espace G des fonctions oscillantes

$G(\mathbb{R}^2)$ est l'espace des distributions f qui peuvent être écrites sous la forme :

$$f = \frac{\partial g_1}{\partial x_1} + \frac{\partial g_2}{\partial x_2} = div(g),$$

avec : g_1 and g_2 dans $L^\infty(\mathbb{R}^2)$ et où $\frac{\partial g_1}{\partial x_1}$ et $\frac{\partial g_2}{\partial x_2}$ sont des dérivées au sens de distributions. $G(\mathbb{R}^2)$ est de plus muni de la norme :

$$\|v\|_{G(\mathbb{R}^2)} = inf\big(\|g\|_{L^\infty(\mathbb{R}^2)} | v = div(g)\big),$$

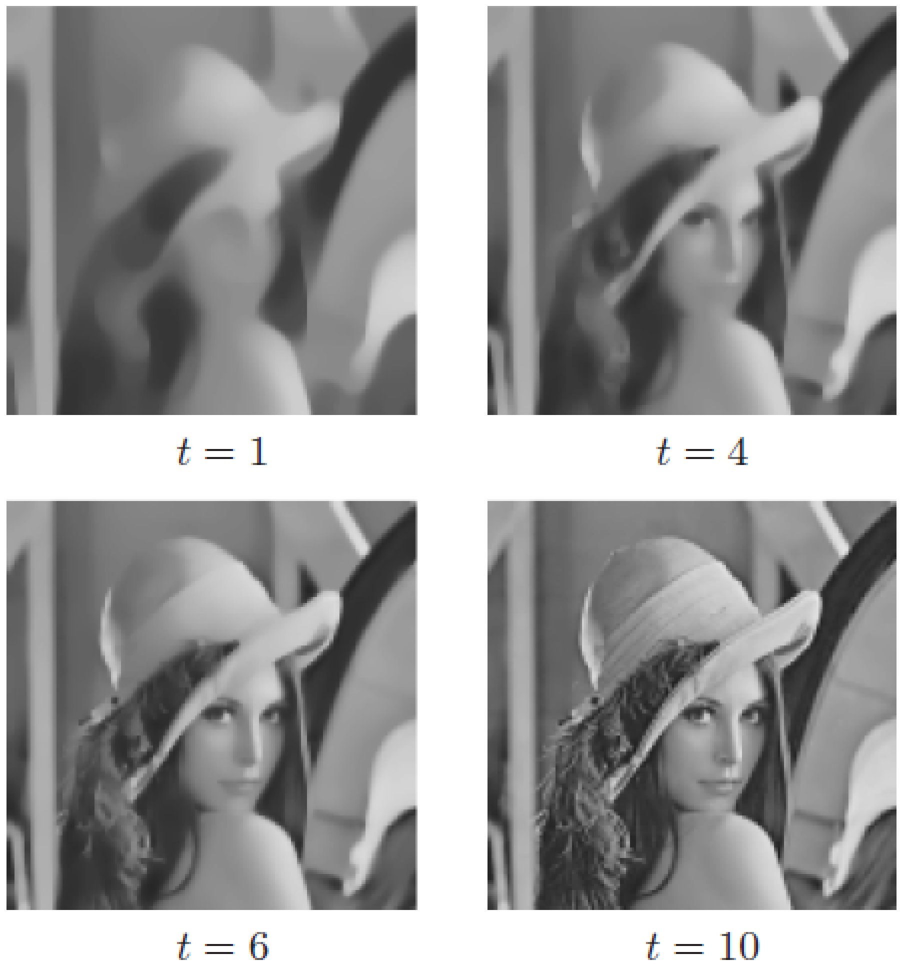


FIGURE A.1 – Les images $U(t) = \int_0^t u(., s)ds$ de Lena sont obtenues par décomposition de Tadmor et al [60] à l'intant $t = 1, 2, 6, 10$. Ici $\lambda(t) = 0.002 \times 2^t$.

où

$$\|g\|_{L^\infty(\mathbb{R}^2)} = sup\|g(x)\|,$$

avec :

$$\|g(x)\| = \sqrt{|g_1|^2 + |g_2|^2}(x).$$

A.1.5 Modèle de Meyer

Meyer propose dans [50] le modèle suivant :

$$\underset{(u,v)\in BV\times G/f=u+v}{inf}\left(\int_\Omega |\nabla u|dx + \lambda\|v\|_G^2\right). \tag{A.10}$$

Toutefois, il ne propose pas dans [50] une méthode numérique permettant de résoudre le problème (A.10).

A.1.6 Modèle de Vese et Osher

Vese et Osher [64] ont proposé une méthode pour minimiser (A.10), la méthode est basée sur une version discrète de l'espace $G(\mathbb{R}^2)$.

A.1.7 Espace de Meyer discret

On suppose que l'image est de taille $N \times N$, l'espace de Meyer discret est donné par :

$$G = \{v \in X | g \in Y \quad \text{tel que} \quad v = div(g)\},$$

et

$$\|v\|_G = inf\{\|g\|_\infty | v = div(g), g = (g^1, g^2) \in X^2\},$$

où

$$\|g\|_\infty = \underset{1\le i,j\le N}{sup}\|g_{i,j}\|_{\mathbb{R}^2},$$

avec

$$\|g_{i,j}\|_{\mathbb{R}^2} = \sqrt{|g^1_{i,j}|^2 + |g^2_{i,j}|^2}.$$

A.1.8 Modèle de Vese et Osher

Pour résoudre le problème de Meyer et décomposer une image f, Vese et al [64] proposent le modèle suivant :

$$\underset{(u,v)\in BV(\Omega)\times G(\Omega)/f=u+v}{inf} \left(\int |\nabla u| + \lambda \|f - u - v\|^2_{L^2(\Omega)} + \mu \|v\|_{G(\Omega)} \right), \tag{A.11}$$

où Ω est un ouvert borné, u la composante structure et v la composante oscillante, ils remplacent le terme $\|v\|_{G(\Omega)}$ par $\|\sqrt{g_1^2 + g_2^2}\|_{L^1}$, puis calculent les équations d'Euleur-Lagrange associées. Pour plus de détails, nous renvoyons le lecteur à [64].

A.1.9 Modèle de Osher, Solé et Vese

Les auteurs dans [65], modifient le modèle (A.11) en remplacant l'espace G de Meyer par $H^{-1}(\Omega)$. Nous rappelons que pour un ouvert Ω de $\mathbb{R}^2$, $H^{-1}(\Omega)$ est le dual de la fermuture de $C^1(\Omega)$ dans $H^1(\Omega)$. Les auteurs considère ainsi le problème suivant :

$$\underset{u\in BV(\Omega)}{inf} \int_\Omega |\nabla u| dx + \lambda \|f - u\|^2_{H^{-1}(\Omega)}, \tag{A.12}$$

qu'il résolvent en utilisant la théorie des équations aux dérivées partielles.

A.2 Application de la méthode Fourier pour l'étude de stabilité du modèle (3.10)

Pour étudier la stabilité par la méthode de Fourier, On considère l'équation linéarisée suivante :

$$\frac{\partial \omega}{\partial t} + u.\nabla \omega - \nu \Delta \omega = 0 \tag{A.13}$$

discrétisée par différence finie selon le schéma explicite suivant :

$$\begin{aligned} \tfrac{\omega^{n+1}_{i,j} - \omega^n_{i,j}}{\Delta t} &= -u_1 . \tfrac{\omega^n_{i+1,j} - \omega^n_{i-1,j}}{2} - u_2 \tfrac{\omega^n_{i,j+1} - \omega^n_{i,j-1}}{2} \\ &+ \nu(\omega^n_{i+1,j} + \omega^n_{i-1,j} - 4\omega^n_{i,j} + \omega^n_{i,j+1} + \omega^n_{i,j-1}) \end{aligned} \tag{A.14}$$

ce qui donne :

$$\begin{aligned} \omega_{i,j}^{n+1} &= -\Delta t u_1.\frac{\omega_{i+1,j}^n-\omega_{i-1,j}^n}{2} - \Delta t u_2\frac{\omega_{i,j+1}^n-\omega_{i,j-1}^n}{2} \\ &+ \nu(\omega_{i+1,j}^n + \omega_{i-1,j}^n - 4\omega_{i,j}^n + \omega_{i,j+1}^n + \omega_{i,j-1}^n) + \omega_{i,j}^n \end{aligned} \tag{A.15}$$

On pose $\alpha = \frac{\Delta t U}{2}$ et $\beta = \Delta t \nu$, avec : $U = max\{u_1, u_2\}$ d'où :

$$\begin{aligned} \omega_{i,j}^{n+1} &= (1-4\alpha)\omega_{i,j}^n + (-\alpha+\beta)\omega_{i+1,j}^n + (\alpha+\beta)\omega_{i-1,j}^n \\ &+ (-\alpha+\beta)\omega_{i,j+1}^n + (\alpha+\beta)\omega_{i,j-1}^n \end{aligned} \tag{A.16}$$

On injecte dans l'équation (A.16) un mode de Fourier, en posant : $\omega_{i,j}^n = \lambda^n e^{ksi}e^{krj}$ avec $k^2 = -1$, on obtient :

$$\begin{aligned} \widehat{\omega_{i,j}^{n+1}} &= [(1-4\beta) + (-\alpha+\beta)e^{ks} + (\alpha+\beta)e^{-ks} \\ &+ (-\alpha+\beta)e^{kr} + (\alpha+\beta)e^{-kr}]\widehat{\omega_{i,j}^n} \end{aligned} \tag{A.17}$$

Mais :

$$\begin{aligned} (-\alpha+\beta)e^{ks} + (\alpha+\beta)e^{-ks} &= \alpha(e^{-ks} - e^{ks}) + \beta(e^{ks} + e^{-ks}) \\ &= 2\beta cos(s) - 2\alpha k sin(s) \end{aligned} \tag{A.18}$$

et :

$$\begin{aligned} (-\alpha+\beta)e^{kr} + (\alpha+\beta)e^{-kr} &= \alpha(e^{-kr} - e^{kr}) + \beta(e^{kr} + e^{-kr}) \\ &= 2\beta cos(r) - 2\alpha k sin(r) \end{aligned} . \tag{A.19}$$

donc :

$$\begin{aligned} \lambda &= 1 - 4\beta + 2\beta(cos(r) + cos(s)) - 4k\alpha(sin(s) + sin(r)) \\ &= 1 - 4\beta sin^2(\tfrac{r}{2}) - 4\beta sin^2(\tfrac{s}{2}) - 4k\alpha(sin(s) + sin(r)) \end{aligned} \tag{A.20}$$

d'où :

$$|\lambda| = \sqrt{(1 - 4\beta sin^2(\frac{r}{2}) - 4\beta sin^2(\frac{s}{2}))^2 + (4\alpha(sin(s) + sin(r)))^2}$$

Comme le schéma est stable pour $|\lambda| < 1$, cela est vérifie dés qu'on a :

$$Re(\lambda) < \frac{\sqrt{2}}{2} \quad \text{et} \quad Im(\lambda) < \frac{\sqrt{2}}{2}$$

Avec $Re(\lambda)$, $Im(\lambda)$ sont : la partie réel et imaginaire du nombre complexe Λ.

On vérifie sans peine que pour $\beta < \frac{1}{8}$ et $\alpha < \frac{\sqrt{2}}{4}$, on obtient $|\lambda| < 1$, d'où la stabilité du schéma. Exprimée en fonction du pas du temps Δt, la condition de stabilité sera donnée par :

$$\Delta t < min\{\frac{1}{8\nu};\frac{\sqrt{2}}{4U}\}$$

A.3 Code matlab pour le modèle d'inpainting

Le code suivant sous matlab assure l'implémentation numérique de la discrétisation de l'EDP du modèle d'inpainting proposé dans le chapitre 3 :

```
function Ainpainted=NSInpainting(IO,Ad,Bd,n,T,dt)
initialisation de u, v , w, ux, vx
[N,M]=size(IO)
for i=1 :N
for j=1 :M
u(i,j)=0 ; v(i,j)=0 ; w(i,j)=0 ; ux(i,j)=0 ; vx(i,j)=0 ;     end
end
A=lapdisc(n) ;
évaluation des composantes de la vitesse sur le bord de oméga
b,a,BF,AF les coordonnées de la zone d'inpainting dans l'image initiale
Bf=n+Bd-1
Af=n+Ad-1
for i=-1 :1 :n+2
u(Ad-1+i,Bd-1+n)=-(1/2)*(-IO(Ad-1+i,Bf+2)+4*IO(Ad-1+i,Bf+1)-3*IO(Ad-1+i,Bf)) ;
u(Ad-1+i,Bd-1+1)=-(1/2)*(3*IO(Ad-i+i,Bd)-4*IO(Ad-1+i,Bd-1)+IO(Ad-1+i,Bd-2)) ;
end ;
for j=-1 :1 :n+2
v(Ad-1+n,Bd-1+j)=(1/2)*(-IO(Af+2,Bd-1+j)+4*IO(Af+1,Bd-1+j)-3*IO(Af,Bd-1+j)) ;
v(Ad-1+1,Bd-1+j)=(1/2)*(3*IO(Ad,Bd-1+j)-4*IO(Ad-1,Bd-1+j)+IO(Ad-2,Bd-1+j)) ;
end ;
```

évaluation des composantes de la vitesse dans oméga

```
for i=2 :n
for j=2 :n
v(Ad-1+i,Bd-1+j)=(1/2)*(IO(Ad-1+i+1,Bd-1+j)-IO(Ad-1+i-1,Bd-1+j)) ;
u(Ad-1+i,Bd-1+j)=-(1/2)*(IO(Ad-1+i,Bd-1+j+1)-IO(Ad-1+i,Bd-1+j-1)) ;
end ;
end ;
```

évaluation des composantes de la dérivé de la vitesse

```
for i=1 :n
vx(Ad-1+i,Bd-1+n)=-(1/2)*(-v(Ad-1+i,Bf+2)+4*v(Ad-1+i,Bf+1)-3*v(Ad-1+i,Bf)) ;
vx(Ad-1+i,Bd-1+1)=-(1/2)*(3*v(Ad-i+i,Bd)-4*v(Ad-1+i,Bd-1)+v(Ad-1+i,Bd-2)) ;
end ;
for j=1 :n
uy(Ad-1+n,Bd-1+j)=(1/2)*(-u(Af+2,Bd-1+j)+4*u(Af+1,Bd-1+j)-3*u(Af,Bd-1+j)) ;
uy(Ad-1+1,Bd-1+j)=(1/2)*(3*u(Ad,Bd-1+j)-4*u(Ad-1,Bd-1+j)+u(Ad-2,Bd-1+j)) ;
end ;
```

évaluation des composantes de w sur le bord

```
for j=1 :n
w(Ad,Bd-1+j)=-uy(Ad,Bd-1+j)+vx(Ad,Bd-1+j) ;
w(Af,Bd-1+j)=-uy(Af,Bd-1+j)+vx(Af,Bd-1+j) ;
end ;
for i=1 :n
w(Ad-1+i,Bf)=-uy(Ad-1+i,Bf)+vx(Ad-1+i,Bf) ;
w(Af,Bd-1+j)=-uy(Ad-1+i,Bd)+vx(Ad-1+i,Bd) ;
end ;
```

évaluation des composantes de w dans oméga

```
for i=2 :n-1
for j=2 :n-1
```

```
w(Ad-1+i,Bd-1+j)=IO(Ad-1+i+1,Bd-1+j)+IO(Ad-1+i-1,Bd-1+j)+IO(Ad-1+i,Bd-1+j+1)-

IO(Ad-1+i,Bd-1+j-1)-4*IO(Ad-1+i,Bd-1+j) ;
end
end
programme principal
Ti représente le compteur d'itération
T le temps de traitement
dt= le pas du temps
Ti=0
while Ti<=T
for i=2 :n-1
for j=2 :n-1
w(Ad-1+i,Bd-1+j)=w(Ad-1+i,Bd-1+j)+dt*(-(1/2)*u(Ad-1+i,Bd-1+j)*(w(Ad-1+i+1,Bd-
1+j)-
w(Ad-1+i-1,Bd-1+j))-(1/2)*v(Ad-1+i,Bd-1+j)*(w(Ad-1+i,Bd-1+j+1)-w(Ad-1+i,Bd-1+j-
1))) ;
end
end
Ti=Ti+dt ;
end
restriction sur les valeurs de la zone d'inpainting
for i=1 :n
for j=1 :n
winp(i,j)=w(Ad-1+i,Bd-1+j) ;
end
end
winpvect=veccol(winp) ;
```

résolution de l'équation du poison

Iinpvect=linsolve(A,winpvect') ;

reconstruction de la zone d'inpainting

Iinp=vec2mat(Iinpvect,n) ;

reconstruction de l'image entière

for i=1 :n

for j=1 :n

IO(Ad-1+i,Bd-1+j)=Iinp(i,j) ;

end

end

Ainpainted=IO ;

A.4 Code matlab pour le modèle de décomposition de l'image

Le code suivant sous matlab assure l'implémentation numérique de la discrétisation de l'équation intégro-différentielle du modèle de décomposition proposé dans le chapitre 3 :

function [imgeometrique,imtexture]=Imdecomp(IO,Ad,Bd,n,T,dt) Initialisation de Uold

Uold=double(zeros(numrows,numcols)) ;

Boucle principale

for m=1 :T

Choix de lambda

limbda=0.002 $* 2^(m)$;

Image originale

wold=double(f) ;

wnew=double(zeros(numrows,numcols)) ;

erreur pour le teste d'arrêt. difference = wnew - wold ;

e=norm(difference) ;

jac=0 ;

Teste d'arrêt

while $e \geq= 1$ and $jac \leq 200$

for i=2 :numrows-1

for j=2 :numcols-1

$Dplusxw = wold(i+1,j) - wold(i,j)$;

$Dminexw = wold(i,j) - wold(i-1,j)$;

$Dcenterx1w = (wold(i+1,j) - wold(i-1))/2$;

$Dcenterx2w = (wold(i+1,j-1) - wold(i-1,j-1))/2$;

$Dplusyw = wold(i,j+1) - wold(i,j)$;

$Dmineyw = wold(i,j) - wold(i,j-1)$;

$Dcentery1w = (wold(i,j+1) - wold(i,j-1))/2$;

$Dcentery2w = (wold(i-1,j+1) - wold(i-1,j-1))/2$;

$CE = 1/sqrt(eps^2 + (Dplusxw)^2 + (Dcentery1w)^2)$;

$CW = 1/sqrt(eps^2 + (Dminexw)^2 + (Dcentery2w)^2)$;

$CS = 1/sqrt(eps^2 + (Dcenterx1w)^2 + (Dplusyw)^2)$;

$CN = 1/sqrt(eps^2 + (Dcenterx2w)^2 + (Dmineyw)^2)$;

$d = 2 * limbda * h^2 * (f(i,j) - Uold(i,j)) + CE * wold(i,j) + CW * wold(i-1,j) + CS * wold(i,j+1) + CN * wold(i,j-1)$;

$n = 2 * limbda * h^2 + CE + CW + CS + CN$;

wnew(i,j)=d/n ;

end

end

différence = wnew - wold ;

Actualisation de l'erreur

e=norm(difference) ;

wold=wnew ;

jac=jac+1 ;

end

Superposition de la partie texture. Unew=Uold+wnew ;

Uold=Unew ;

Imtexture=Uold ;

Imgeometrique=f-Uold ; end

.

Bibliographie

Bibliographie

[1] S. Agmon and L. Niremberg, Lower Bounds and Uniqueness Theorems for Solutions of Differential Equations in Hilbert Spaces, *Comm. Pure Applied math*, (20), pages : 207-229, (1960).

[2] G. Allaire, Analyse Numérique et Optimisation, *Ellipses*,(2006).

[3] L. Ambrosio, N. Fusco and D. Pallara, Functions of Bounded Variation and Free Discontinuity Problems, *Oxford Science Publications*, (2000).

[4] K. Amoura and F.Z. Nouri, Problème de Stokes et La Technique de Glowinski, *Journal of Sciences and Technology*, pages : 34-38, (2006).

[5] J. Argence, D. Ghazanfarpour, D. Michelucci and B. Peroché, La Synthèse d'Images, *Hermes Science Publication*, (1990).

[6] P. Arias, V. Caselles, G. Facciolo, and G. Sapiro, Variational Framework for Exemplar-Based Image Inpainting, *Int. J. Comput. Vis*, 3(23), pages : 1-29, (2001).

[7] M.B. Artzi, J.P. Croisille, D. Fishlov and S. Trachtenberg, A Pure-Compact Scheme For the Streamfunction Formulation of Navier-Stokes Equations, *Journal of computational Physics* , (205), pages : 640-664, (2005).

[8] G. Aubert and P. Komprobst, Mathematical Problems in Image Processing :Partial Differential Equations and the Calculus of Variations, *Apllied Mathematical Sciences*, (174), (2006).

[9] J.F. Aujol, S. Ladjal and S. Masnou, Exemplar-Based Inpainting From a Variational Point of View, *SIAM J.Math. Anal*, 3(42), pages 1246-1285, (2010).

[10] B. Ballester, M. Bertalmio, V. Caselles G. Sapiro and J. Verdera, Filling-in by Joint Interpolation of Vector Fields and Gray Levels, *IEEE Transactions on Image Processing*, 8(10), pages : 1200-1211, (2001).

[11] B. Ballester, V. Caselles, J. Verdera, M. Bertalmio and G. Sapiro, A Variational Model for Filling-in Gray Level and Color Images, *IEEE Computer Society Press*, (1), pages : 10-16, (2001).

[12] B. Ballester, V. Caselles and J. Verdera, Disocclusion by Joint Interpolation of Vector Fields and Gray levels, *Multiscale Modeling and Simulation*,(2), pages : 80-123, (2003).

[13] M. Bertalmio, V. Caselles, C. Ballester and G. Sapiro, Computer Graphics, *SIGGRAPH*, pages : 417-424, July(2000).

[14] M. Bertalmio, A.L. Bertozzi and G. Sapiro, Navier Stokes, Fluid Dynamics, and Image and Vedio Inpainting, *Proceedings of the 2001 IEEE Computer Society Conference*, pages : 355-362, (2001)

[15] M. Bertalmio, V. Caselles, G. Haro and G. Sapiro, PDE-based Image and Surface Inpainting, *Springer*, pages : 32-36, (2005).

[16] M. Bertero and P. Boccacci, Introduction to Inverse Problems in Imaging, *CRC Press*, (1997).

[17] T. Chan and J. Shen, Mathematical Models for Local Deterministic Inpaintings, *Technical report 00-11, Department of Mathematics, UCLA*, LosAngeles (2000).

[18] T. Chan and J. Shen, Non-Texture Inpainting by Curvature-Driven Diffusion, *Journal of visual communication and image representation*, pages : 436-449, (2001).

[19] T. Chan, S.H. Kang, and J. Shen, Euler's Elastica and Curvature Based Inpainting, *SIAM J. Appl. Math*, (2002).

[20] T. Chan and J. Shen, Mathematical Models for Local Non-Texture Inpainting, *SIAM J. Appl. Math*, (62), pages : 1019-1043, (2002).

[21] G.H. Chen, C.L. Yang, L.M. Po and S.L. Xie, Image Quality Assessment : from Error Measurement to Structural Similarity, *Proc. IEEE Transactions on Image Processing*, pages : 600-612, (2004).

[22] G.H. Chen, C.L. Yang, L.M. Po and S.L. Xie, Edge-Based Strucural Similarity for Image Quality Assessment, *Proc. IEEE Internatianal Conference on Acoustics, Speech and Signal Processing*, (2006).

[23] P.G. Ciarlet, Introduction à l'Analyse Numérique Matricielle et à l'Optimisation, *Masson*, (1982).

[24] P.G. Ciarlet and J.L. Lions, Handbook of Numerical Analysis, *North-Holland*, (1990).

[25] M. Coster and J.L. Chermant, Précis d'Analyse d'Images,*Presse de CNRS*, (1989).

[26] A. Criminisi, P. Pérez and K. Toyama, Region Filling and Object Removal by Examplar Based Image Inpainting, *IEEE Trasactions on Image Processing*, 9(13), (2004).

[27] R. Dautray, and J.L. Lions, Analyse Mathématique et Calcul Numérique pour les Sciences et les Techniques, *Masson*, (1998).

[28] J.P. Demailly, Anallyse Numériqe et Equations Différentielles, *EDP Sciences*, (2006).

[29] R. Deriche, Fast Algorithms for Low-Level Vision, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12(1), pages : 78-88, (1990).

[30] R. Deriche and O. Faugeras, Les Équations aux Dérivées Partielles en Traitement des Images et Vision par Ordinateur, *Traitement du Signal*, (13), pages : 551-577, (1996).

[31] M.A. Ebrahimi, M. Holst and E. Lunasin, The Navier-Stokes-Voight Model for Image Inpainting, (2009).

[32] A. Elmahi and D. Meskine, Strongly Nonlinear Parabolic Equations With Natural Growth Terms in Orliez Space, *Nonlinear Analysis*, (60), pages : 1-35, (2005).

[33] F. Drira, Contributions à la Restauration des Images de Documents Anciens, *Thése de Doctorat, l'Institut National des Sciences Appliquées de Lyon*, (2007).

[34] G.P. Galdi, An Introduction to the Mathematical Theory of the Navier-Stokes Equation, *Springer*, (2011).

[35] V. Girault and P. Raviart, Finite Element Methods for the Navier-Stokes Equations, Theory and Algorithms, *Springer Verlag*, (1989).

[36] E. Goncalvès, Résolution Numérique, Discrétisation des EDO et EDP, *Cours dans l'Institut National Polytechnique de Grenoble*, (2008).

[37] R.C. Gonzalez, and R.E. Woods, Digital Image Processing, *Prentice Hall*, (2008).

[38] P. Grisvard, Caractérisation de Quelques Espaces d'Interpolation, *Archiv for Rat*, pages : 40-63, (1967).

[39] R. Horaud and O. Monga, Vision par Ordinateur, *Hermes Science Publication*, (1993).

[40] G. Kanizsa, Organization in Vision, Praeger, New York (1979).

[41] G. Kanizsa, Grammatica Del Vedere, Bologna, Il Mulino (1980).

[42] M. Lakrib, Cours l'Analyse Numérique, *Office des Publications Universitaires* , (2005).

[43] D. Lingrand, Introduction au traitement d'images, *Vuibert*, (2008).

[44] J.L. Lions, Quelques Méthodes de Résolution pour des Problèmes aux Limites Non Linéaires, *Dunod Gauthier-Villars*, (1969).

[45] M. Maouni and F.Z. Nouri, Image Restoration Based on P-Gradient Model, *International Journal of Applied Mathematics*, (41), pages : 48-57, (2013).

[46] H. Maitre, Le traitement des images, *Hermes science publication*, (2002).

[47] S. Masnou and J.M. Morel, Level Lines Based Disocclusion, *International Conference on Image Processing*, (3), pages : 259-263, (1998).

[48] D. Mezhoud, F.Z. Nouri and P. Spiteri, Inpainting by an Inverse Problem Resolution, *2014 18th International Conference on Information Visualisation*, pages : 298-301, (2014).

[49] D. Mezhoud, F.Z. Nouri and P. Spiteri, An Inpainting Result Based On P-Laplace Operator, *International Journal of Computing Science and Mathematics*, 6(7), pages : 554-567, (2016).

[50] Y. Meyer, Oscillating Patterns in Image Processing and Nonlinear Evolution Equations, *American Mathematical Science*, (22), (2001).

[51] A. Mitiche and I.B. Ayed, Variational and Level Set Methods in Image Segmentation, *Springer Topics in Signal Processing* , (2011).

[52] M. Nitzberg, D. Mumford and T. Shiota, Filtering, Segmentation and Depth, *Lecture Notes in Computer Science*, (662), (1993).

[53] W.K. Pratt, Digital Image Processing, *Wiley Inter Science*, (2001).

[54] J. Rappaz and M. Picasso, Introduction à l'Analyse Numérique,*Presse Polytechniques et Universitaires Romandes*, (2004).

[55] L. Rudin, S. Osher and E. Fatmi, Nonlinear Total Variation Based Noise Removal Algorithm, *Physica de Nolinear Phenomina*, 60(25), pages 1072–1076, American Physical Society, Dec (1992).

[56] Y. Saad, Iterative Methods for Sparse Linear Systems, *PWS*, (1996).

[57] F.Y. Shih, Image Processing and Mathematical Morphology : Fundamentals and Applications, *CRC Press*, March (2006).

[58] J. Sun, L. Yuan, J. Jia and H.Y. Shum, Image Completion With Structure Propagation, *ACM SIGGRAPH*, (24), pages : 861–868, (2005).

[59] E. Tadmor, S. Nezzar and L. Vese, A Multi-scale Image Representation Using Hierarchical Decompositions, *Multiscale Model Simul*, 4(2), pages : 554-579, (2004).

[60] E. Tadmor, S. Nezzar and L. Vese, Multiscale Hierarchical Decomposition of Images With Applications to Deblurring, Denoising and Segmentation, *Commun. Math. Sci*, 6(2), pages : 1-26, (2008).

[61] E. Tadmor and P. Athavale, Multi-scale Image Representation Using Integro-Differential Equations, *Inverse Problems and Imaging*, 4(3), pages :693-710, (2009).

[62] E. Tadmor and P. Athavale, Integro-Differential Equations Based on Image Decomposition, *SIAM Journal on Imaging Sciences*, 4(1), pages : 300-312, (2011).

[63] R. Temam, Navier-Stokes Equations and Nonlinear Functional Analysis, *CBMS-NSF Regional Conference Series in Applied Mathematics*, (1995).

[64] L. Vese and S. Osher, Modeling Textures With Total Variation Minimization and Oscillatory Functions, *Journal of Scientific Computing*, 1(19), pages : 553-572, (2003).

[65] L. Vese, S. Osher, and A. Sole, Image Decomposition and Restoration Using Total Variation Minimization, *Multi-scale Model Simulation*, pages : 349-370, (2003).

[66] D.F. Walnut, An Introduction to Wavelet Analysis, *Birkhauser*, (2004).

[67] B. A. Youcef and E. H. Atta, Digital Image Inpainting Using Finite Volume Approach and The Navier-Stokes Equations, *Proc. Int Conference in Mathematical and Computational Methods in Science and Engineering*, (2007).