



**Universit Ibn Zohr**  
**Facult des Sciences Agadir**  
**Dpartement d'Informatiques**  
**Filire Sciences Mathmatiques et Informatique**

## **Réduction de bruit de la vidéo par filtre non local**

Projet tutor préparé par  
**Chaima Elmejjari et Nouhaila Hassni**

Sous la direction de  
Prof : M. El Oufdi Ahmed Fouad

Soutenu le : ... Mai 2020

Devant le jury :

Prof. .... Professeur la Facult des sciences Agadir  
Prof. .... Professeur la Facult des sciences Agadir

# Introduction

# Remerciements

# Table des matières

<b>1</b>	<b>Généralisation</b>	<b>5</b>
1.1	Introduction . . . . .	5
1.2	Définition d'une image . . . . .	5
1.2.1	Les images binaires (noir ou blanc) . . . . .	6
1.2.2	Les images en niveaux de gris . . . . .	6
1.2.3	Les images couleurs . . . . .	6
1.3	Formats d'images . . . . .	7
1.4	Traitement et acquisition d'image . . . . .	8
1.4.1	Processus du capteur . . . . .	8
1.4.2	Acquisition d'image Utilisation d'un seul capteur . . . . .	8
1.4.3	Acquisition d'image Utilisation Capteur matriciel . . . . .	8
1.5	Les sources de bruit . . . . .	10
1.5.1	Le bruit additif . . . . .	10
1.5.2	Le bruit multiplicatif . . . . .	10
1.6	Types de bruit . . . . .	11
1.6.1	Bruit périodique . . . . .	11
1.6.2	Bruit de chrominance . . . . .	11
1.6.3	Bruit de luminance . . . . .	11
1.6.4	Bruit Poivre et sel . . . . .	12
1.7	Conclusion . . . . .	13
<b>2</b>	<b>Filtrage des images</b>	<b>14</b>
2.1	Introduction . . . . .	14
2.2	les Filtres local . . . . .	14
2.2.1	filtre local linéaire . . . . .	14
2.2.2	filtre local non linéaire . . . . .	15
2.3	le Filtre non local . . . . .	16
2.4	Conclusion . . . . .	17
2.5	Méthode du bruit . . . . .	18
2.5.1	Neighborhood filters . . . . .	18
2.5.2	The Total Variation minimization . . . . .	18

<b>3</b>		<b>19</b>
3.1	. . . . .	19
3.2	Conclusion . . . . .	20
3.3	Bibliographie . . . . .	21
3.4	ANNEXES . . . . .	22

# Chapitre 1

## Généralisation

### 1.1 Introduction

Les images numériques sont présentées sous forme de matrices, cependant les opérations, en particulier arithmétique, se font entre pixels, des deux images, situés aux mêmes positions. De ce fait, nous allons réserver ce chapitre à la présentation des notions générales liées à l'image.

### 1.2 Définition d'une image

Une image numérique est un signal numérique composé d'unités élémentaires (appelées pixels) qui représentent chacun une portion de l'image. Contrairement au cas unidimensionnel, nous étudierons uniquement les images numériques (discrètes).

Une image numérique est définie par :

- le nombre de pixels qui la composent en largeur et en hauteur,
- l'étendue des teintes de gris ou des couleurs que peut prendre chaque pixel (on parle de dynamique de l'image).

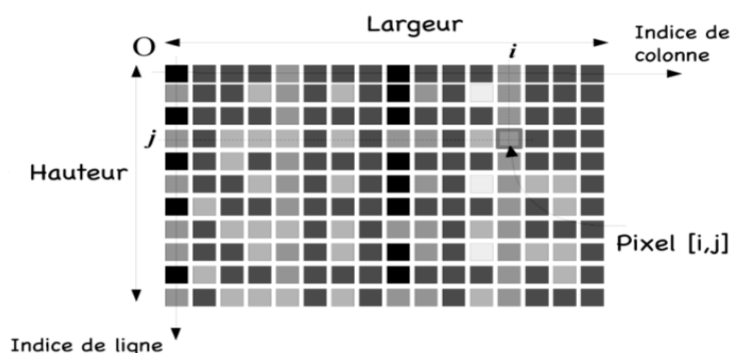


FIGURE 1.1 – Capteurs

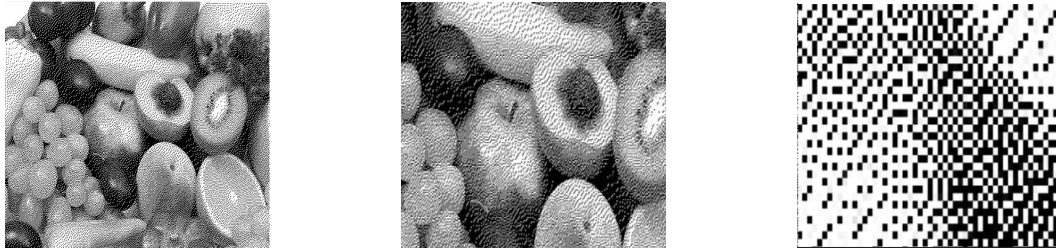


FIGURE 1.2 – Les images binaires (noir ou blanc)

### 1.2.1 Les images binaires (noir ou blanc)

Est une image en noir et blanc, dans lequel chaque pixel est soit noir, soit blanc.

### 1.2.2 Les images en niveaux de gris



FIGURE 1.3 – Les images en niveaux de gris

images en niveaux de gris renferment 256 teintes de gris. Par convention la valeur zéro représente le noir et la valeur 255 le blanc. En effet chaque entier représentant un niveau de gris est codé sur 8 bits.

Il est donc compris entre 0 et  $2^8 - 1 = 255$ . On peut coder une image en niveaux de gris sur 16 bits ( $0 \leq n \leq 2^{16} - 1$ ) ou sur 2 bits, dans ce dernier cas le niveau de gris vaut 0 ou 1 : il s'agit alors d'une image binaire (Noir et Blanc).

### 1.2.3 Les images couleurs



FIGURE 1.4 – Les images couleurs

Pour chaque pixel, on code en binaire chacune des 3 valeurs des composantes R, V, B (Rouge, Vert, Blue) de la couleur du pixel. Le code binaire de l'image est obtenu en indiquant successivement pour chaque pixel le code binaire des 3 composantes. Si on code chaque composante sur 8 bits, chaque pixel sera donc représenté par 24 bits.

## 1.3 Formats d'images

Un format d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler.

Il existe un grand nombre de formats. Voici les plus utilisés :

format	Compression des données	Nb de couleurs	Affichage progressif	Usage
BMP	Non compressé	de 2 à 16 millions	non	Image non dégradée mais très lourde.Stokage
JPEG	Réglable,avec perte de qualité.Plus la compression est importante,plus l'image est dégradée.	16millions	oui	Tous usages,selon compression.Images"naturelles".
GIF	Oui,sans perte de qualité	de 2 à 256 avec palette	oui	Logos et Internet.Supporte les animations et la transparence
TIFF	Réglable, au choix sans perte ou avec perte de qualité	16 millions	non	Tous sauf Internet
PNG	Oui, sans perte de qualité	de 2 à 256 ou 16 millions	oui	Tous, recommandé Internet mais incompatible avec les navigateurs anciens. Supporte la transparence.



## 1.4 Traitement et acquisition d'image

- Il existe trois agencements de capteurs principaux utilisés pour transformer l'énergie d'éclairage en images numériques.

- Capteur d'imagerie unique.
- Capteur de ligne.
- Capteur matriciel.

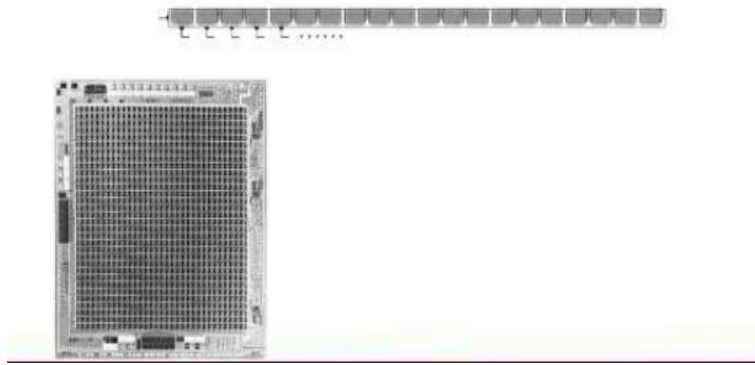


FIGURE 1.5 – Capteurs

### 1.4.1 Processus du capteur

L'énergie entrante est transformée en tension par la combinaison de l'énergie électrique d'entrée et d'un matériau de capteur sensible au type particulier d'énergie détecté.

La forme d'onde de la tension de sortie est la réponse du ou des capteurs, et une quantité numérique est obtenue à partir de chaque capteur en numérisant sa réponse.

### 1.4.2 Acquisition d'image Utilisation d'un seul capteur

Le plus familier Capteur de ce type est la photodiode, qui est construite en matériaux de silicium et dont la forme d'onde de tension de sortie est proportionnelle à la lumière. L'utilisation d'un filtre devant un capteur améliore la sélectivité. Afin de générer une image 2D à l'aide d'un seul capteur, il doit y avoir des déplacements relatifs dans les directions X et Y entre le capteur et la zone à imager.

### 1.4.3 Acquisition d'image Utilisation Capteur matriciel

L'acquisition d'image est la création d'une représentation codée numériquement des caractéristiques visuelles d'un objet, comme une scène physique ou la structure intérieure d'un objet.

chaque appareil photo numérique possède un capteur qui convertit la lumière en charges électriques. Une façon simplifiée de penser à ces capteurs est de penser à un tableau 2D de milliers

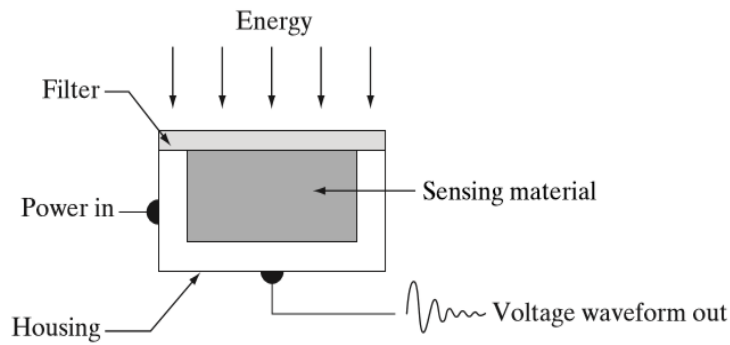


FIGURE 1.6 – Capteur d'imagerie unique

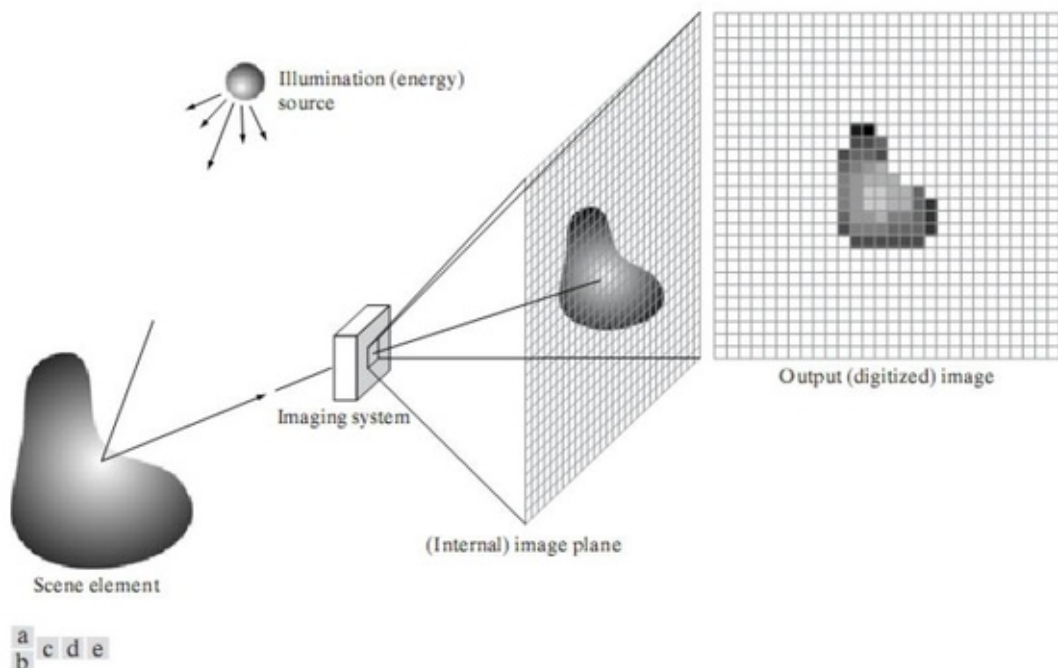


FIGURE 1.7 – le processus d'acquisition d'images numériques

ou des millions de minuscules cellules solaires.

Il existe deux grandes familles des capteurs CCD et CMOS :

- >Les capteurs CCD créent des images de haute qualité et à faible bruit.
- >CMOS les capteurs sont généralement plus sensibles au bruit.

Un CCD transporte la charge à travers la puce et le lit dans un coin du tableau. Un analogique-numérique convertisseur (ADC) transforme ensuite la valeur de chaque pixel en une valeur numérique en mesurant le montant de la charge à chaque photosite et convertit cette mesure en forme binaire.

Les appareils CMOS utilisent plusieurs transistors à chaque pixel pour amplifier et déplacer la charge en utilisant plus traditionnellement des fils.

## 1.5 Les sources de bruit

### 1.5.1 Le bruit additif

Le bruit additif peut être défini de la façon suivante : étant données une image non bruitée R et I la même image avec bruit additif A.

Alors chaque pixel j est caractérisé par la relation :

$$I_j = A_j + R_j$$

Où  $A_j$  est une variable aléatoire de moyenne à 0.

### 1.5.2 Le bruit multiplicatif

Le bruit multiplicatif se définit de façon analogue : étant données une image non bruitée R et I la même image avec bruit multiplicatif Alors chaque pixel j est caractérisé par la relation :

$$I_j = R_j * B_j$$

Où  $B_j$  est une variable aléatoire de moyenne égale à 1. La principale caractéristique de ce bruit est que les pixels d'une zone homogène seront d'autant plus bruités que leur niveau de gris est élevé.

## 1.6 Types de bruit

### 1.6.1 Bruit périodique

Le bruit périodique dans une image est un bruit généré par un composant électronique , une machine ou un processus cyclique, la périodicité du bruit se constate par un motif de bruit se superposant sur des régions de l'image avec une fréquence spatiale déterminée.



FIGURE 1.8 – bruit périodique

### 1.6.2 Bruit de chrominance

C'est la composante colorée des pixels bruités : il est visible sous la forme de taches de couleurs aléatoires.

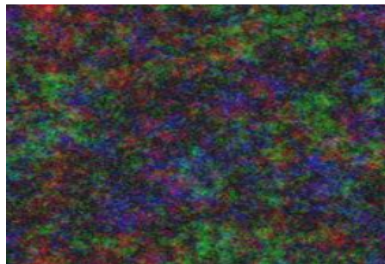


FIGURE 1.9 – bruit de chrominance

### 1.6.3 Bruit de luminance

C'est la composante lumineuse des pixels bruités : il est visible sous la forme de taches plus foncées ou plus claires donnant un aspect granuleux à l'image.

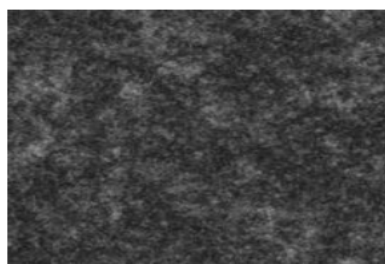


FIGURE 1.10 – bruit de luminance

#### 1.6.4 Bruit Poivre et sel

Le bruit poivre et sel également appelé bruit impulsif, est une dégradation de l'image sous la forme de pixels noirs et blancs (d'où le nom poivre et sel) répartis d'une manière aléatoire dans l'image. Ce bruit est dû soit à des erreurs de transmission de données, soit aux dysfonctionnement ou à la présence de particules fines sur les éléments du capteur de la camera ou a des emplacements mémoire défectueux dans le matériel.



FIGURE 1.11 – bruit Poivre et sel

## 1.7 Conclusion

Le bruit est une information indésirée qui pose un risque de perdre du contenu de l'image. Dans ce chapitre nous avons présenté les différents types de bruits, avec leurs sources. Dans le chapitre suivant nous présentons les méthodes utilisées pour supprimer le bruit d'image avec une préservation des données utiles.

# Chapitre 2

## Filtrage des images

### 2.1 Introduction

Le filtrage est une approche qui consiste à éliminer la présence d'information parasite qui s'ajoute de façon aléatoire à l'image, ce qui conduit à améliorer sa qualité. Comme nous pouvons dire, le filtrage est une opération fondamentale en traitement d'image, il permet d'améliorer la perception de certains détails, de réduire le bruit, de compenser certains défauts du capteur, etc...

Dans ce chapitre nous nous intéressons avant tout aux généralités sur les catégories de filtrage, puis nous étudions les filtres utilisés dans notre travail.

### 2.2 les Filtres local

#### 2.2.1 filtre local linéaire

##### a • filtre moyennneur

Appelé également mean filtering , averaging ou box filtering . Ce filtrage remplace un pixel par la moyenne de lui meme et de ses voisins.

Le filtre moyennneur permettant d'éliminer les hautes fréquences ,correspondant au bruit .son inconvénient est qu'il élimine également les hautes fréquences correspondants aux détailss de l'image : il rend ainsi l'image moins bruitée mais plus floue.

Exemple :

$$\begin{bmatrix} g & c & h \\ b & a & d \\ f & e & i \end{bmatrix} = \begin{bmatrix} 79 & 54 & 47 \\ 192 & 190 & 153 \\ 166 & 189 & 203 \end{bmatrix}$$

Dans notre exemple , cette moyenne vaut :

$$(190+192+79+54+47+153+203+189+166)/9=141.4$$

### a • Filtre gaussien

Appelé également gaussian filtering . le principe de ce filtrage est une convolution avec une gaussien.

Nous rappelons l'expression d'une gaussienne s'écrit sous forme :

$$G_h(\mathbf{x}) = \frac{1}{(4\pi h^2)} e^{-\frac{|\mathbf{x}|^2}{4h^2}}$$

le lissage gaussien permet de corriger le bruit dans les parties homogènes des images mais est moins efficace que le lissage moyenneur .

## 2.2.2 filtre local non linéaire

### a • le filtre médian

Ce filtre vise à remplacer la valeur du pixel central par la valeur médiane de la répartition (on trie les luminances dans l'ordre croissant) des niveaux de gris des pixels situés à l'intérieur de cette fenêtre, comme l'illustre la figure suivante :

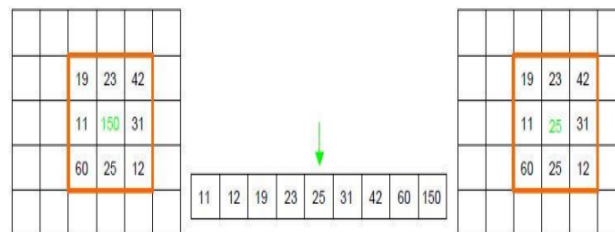


FIGURE 2.1 – FILTRE MEDIAN

### algorithme

1. trier les valeurs du voisinage par ordre croissant.
2. sélectionner la médiane des valeurs (La médiane est la valeur "milieu" : 50 plus claires)
3. attribuer cette valeur médiane au niveau de sortie. (Pixel centrale)

### a • filtre minimum

Ce filtre consiste pour chaque pixel à remplacer sa valeur par celle du minimum sur une fenêtre carrée centrée sur le pixel.

Algorithme :

1. Choisir une fenêtre de taille impaire ( $3 \times 3$ ;  $5 \times 5$ ; etc... )



2. Centrer la fenêtre sur chaque pixel et regrouper les valeurs des pixels voisins qui se trouvent à l'intérieur de la fenêtre.
3. Remplacer la valeur du pixel courant par le niveau de gris le minimum parmi ses pixels voisins.

#### **b • filtre maximum**

Consiste pour chaque pixel à remplacer sa valeur par celle du maximum sur une fenêtre carrée centrée sur le pixel.

Algorithme :

1. Choisir une fenêtre de taille impaire ( $3 \times 3$ ;  $5 \times 5$ ; etc... )
2. Centrer la fenêtre sur chaque pixel et regrouper les valeurs des pixels voisins qui se trouvent à l'intérieur de la fenêtre.
3. Remplacer la valeur du pixel courant par le niveau de gris le maximum parmi ses pixels voisins.

Les filtres minimum et maximum ordonnent l'ensemble des pixels du voisinage, et sélectionnent soit la plus petite ou la plus élevée. Cette famille de filtre permet de supprimer des petits détails très lumineux ou très sombres, mais affecte fortement la taille des objets.

## **2.3 le Filtre non local**

#### **a • Débruitage par patches**

Le débruitage par morceaux (patches) est une technique de débruitage d'image utilisant l'algorithme de réduction du bruit numérique appelé en Anglais "non-local means". Contrairement aux filtres habituels qui réalisent une moyenne des valeurs du groupe de pixels localisés autour d'un pixel cible afin de réduire le bruit, le filtre "non-local means" réalise une moyenne de la totalité des valeurs des pixels contenus dans l'image, pondérées en fonction de leur similarité avec le pixel cible.

#### **b • Principe**

Comme montré par Buades, le principe des moyennes non-locales (NL Means), consiste à tirer profit d'une redondance de l'information à longue distance que l'on peut trouver dans les images. Cette approche est basée sur l'auto-similarité existant dans l'image elle même, alors il s'agit de trouver les pixels (patches) similaires dans l'image, ensuite calculer leur moyenne pondérée en fonction de leur similarité avec le pixel à débruiter.

Soit  $S(x)$  l'ensemble des pixels semblables à  $x(x \in \Omega)$ . Ainsi :

$$u(x) = \sum_{x=\Omega_y} w(x,y)v(y)$$

$u(x)$  est la valeur débruitée d'un pixel  $x$ .

$v(y)$  la valeur à débruiter au point  $y$  ( $y \in \Omega_2$ ).  
 $w(x, y)$  les poids.

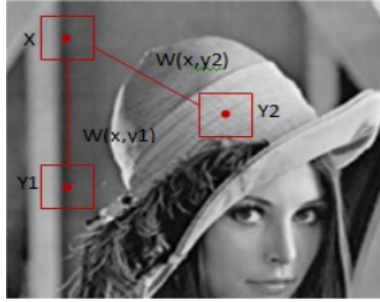


FIGURE 2.2 – Principe de l'algorithme NL-Means.

### c • Algorithme NL-Means

L'algorithme NLM est un algorithme de débruitage par moyenne pondérée, dont le noyau qui forme les poids évalué sur la distance entre des voisinages de ces pixels (patches). Les poids  $w(x, y)$  étant normalisés, nous obtenons :

$$\sum_{y \in \Omega_x} w(x, y) = 1$$

Dans l'algorithme N-L means, l'image débruitée est estimée selon :

$$NLM(x) = \sum_{y \in \Omega_x} \frac{1}{Z(x)} e^{-\frac{\|P(x) - P(y)\|_{2,a}^2}{2h^2}} v(y)$$

avec

$Z$  : le facteur de normalisation,

$P(x)$  : le path au pixel  $x$  fixé,

$P(y)$  : les patches similaires à  $P(x)$ ,

$h$  : le paramètre de contrôle du degré de lissage.

## 2.4 Conclusion

Le filtrage représente une opération importante dans le traitement d'images, car il permet de restaurer l'image originale après une détérioration (bruitage).

Comme plusieurs types de bruits peuvent affecter une image, les traiteurs d'images ont développé plusieurs types de filtres à cet effet.

## 2.5 Méthode du bruit

### 2.5.1 Neighborhood filters

Neighborhood filters sont basés sur l'idée que tous les pixels appartenant au même objet ont une valeur de niveau de gris similaire.

Neighborhood filters prennent donc une moyenne des valeurs des pixels qui sont à la fois proches en valeur de niveau de gris et en distance spatiale.

L'expression s'écrit sous forme :

$$YNF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_{B_\rho(\mathbf{x})} u(\mathbf{y}) e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}} d\mathbf{y}$$

$B_\rho(\mathbf{x})$  est une boule de centre  $\mathbf{x}$  et de rayon  $\rho$ .

$h$  est le paramètre de filtrage .

$C(\mathbf{x})$  est le facteur de normalisation .

### 2.5.2 The Total Variation minimization

La minimisation de la variation totale a été introduite par Rudin, Osher et Fatemi. Étant donné une image bruyante  $v(\mathbf{x})$ , ces auteurs ont proposé de récupérer l'image d'origine  $u(\mathbf{x})$ .

L'expression s'écrit sous forme :

$$TVF_\lambda(v) = \arg \min_u TV(u) + \lambda \int |v(\mathbf{x}) - u(\mathbf{x})|^2 d\mathbf{x}$$

Et le but de cette méthode est de trouver une image qui minimise la longueur des contours, ce qui tend à se débarrasser du bruit, tout en restant proche des données initiales.

# Chapitre 3

## 3.1

## **3.2 Conclusion**

### **3.3 Bibliographie**

## **3.4 ANNEXES**