

N° d'ordre :



جامعة محمد بوضياف - المسيلة
Université Mohamed Boudiaf - M'sila
كلية التكنولوجيا
Faculté de Technologie
قسم الإلكترونيك
Département d'Electronique



MEMOIRE DE MASTER

Présenté par : Telli Elhamlaoui

DOMAINE : SCIENCES ET TECHNOLOGIES

FILIERE : ELECTRONIQUE

OPTION : contrôle industriel

Thème

Efficacité de détection de défilement de texte sur trames continues

Soutenue le : 09/06/2015 devant le jury composé de :

Guer matt Nobil

M.C.B - Université Mohamed Boudiaf - M'sila

Président

Lalaoui Lahouaoui

M.A.A - Université Mohamed Boudiaf - M'sila

Encadreur

YESSAD Dalila

M.A.B - Université Mohamed Boudiaf - M'sila

Examineur

OUALI Mohammed Assam

M.A.A - Université Mohamed Boudiaf - M'sila

Examineur

Promotion : Juin 2015

Sommaire

Remerciements.....	IV
Liste des figures.....	VI
Liste des abréviations.....	VII
Introduction Générale.....	1
CHAPITRE I : Traitement d'Image et Vidéo	
I.1 Introduction.....	3
I.2 Traitement D'image.....	3
I.2.1 Définition d'une image.....	4
I.2.1.1 Pixel.....	4
I.2.1.2 Résolution.....	4
I.2.2 Opérateurs ponctuels.....	5
I.2.2.1 L'histogramme.....	5
I.2.2.2 L'histogramme cumulé.....	5
I.2.3 Espace de couleur.....	6
I.2.3.1 Le système RGB.....	6
I.2.3.2 Le système de représentation YCrCb.....	6
I.2.4 Arithmétique de l'image.....	7
I.2.4.1 Soustraction.....	7
I.2.4.2 Le ET logique.....	8
I.2.5 Amélioration des images.....	8
I.2.6 Opérateurs locaux.....	8
I.2.6.1 Filtrage spatiale.....	8
I.2.6.2 Convolution.....	9
I.2.6.3 Filtres linéaires.....	9
I.2.6.4 Lissage.....	9
I.2.6.4.1 Filtre moyennneur.....	9
I.2.6.4.2 Filtre médian.....	10
I.2.7 Binarisation.....	11
I.2.8 Opérateur morphologique pour l'amélioration des images binaires.....	13
I.2.8.1 L'élément structurant.....	13
I.2.8.2 Dilatation.....	13
I.2.8.3 Erosion.....	14

I.2.9 Format d'images.....	14
I.2.9.1 JOEG (Joint Photo Expert Group).....	14
I.2.9.2 JPEG2000 (Joint Photo Expert Group 2000).....	14
I.2.9.3 PNG (Portable Network Graphics).....	15
I.2.9.4 TIFF (Tag Image File Format).....	15
I.2.9.5 GIF (Compuserve Graphics).....	15
I.3 Application de traitement d'image.....	15
I.4 Vidéo.....	16
I.4.1 La vision.....	16
I.4.2 La couleur.....	16
I.4.3 Le signal.....	16
I.4.4 L'affichage	12
I.4.5 Le son.....	18
I.4.6 Les formats d'enregistrement.....	19
I.4.7 Le format DV (Digital Video)	19
I.4.8 Formats standard vidéo	20
I.5 Conclusion.....	20
CHAPITRE II : Les filtres de traitement d'images	
II.1 Introduction.....	21
II.2 Filtres de convolution	22
II.3 Filtrage spatial	24
a- Filtres passe haut	24
b- Filtres passe bas	24
II.4 Bruits	24
a- Bruit linéaire	24
b- Bruit non linéaire	25
II.5 Filtres linéaires	25
II.5.1 Lissage	25
a- Filtre Moyenneur.....	25
b- Filtre Gaussien.....	26
II.5.2 Amélioration du contraste	27
II.5.3 Détection de contours	27
a- Filtres contours verticaux.....	27
b- Filtres contours horizontaux.....	27

c- Filtre Laplacien.....	28
II.6 Filtres non linéaires	28
II.6.1 Lissage	29
a- Filtre Médian.....	29
b- Filtre Max.....	30
c- Filtre Min.....	30
II.6.2 Les filtre de détection des contours	30
a- Filtre de Sobel.....	30
b- Filtre de Prewitt.....	31
c- Filtre de Robert.....	32
II.7 Conclusion	33
 CHAPITRE III : Algorithme Efficace pour la Détection de Défilement du Texte sur une séquence Vidéo	
III.1 Introduction.....	34
III.2 Algorithme proposée.....	34
III.2.1 Lissage d'image et détection de contour	35
III.2.2 Différentiel Temporel Adaptative.....	37
III.2.3 Processus de dilation et de localisation du texte	41
III.3 Conclusion.....	43
 CHAPITRE IV : Simulation & Résultats	
IV.1 Introduction.....	44
IV.2 Simulation.....	44
IV.3 Conclusion.....	47
Conclusion Générale.....	48
Bibliographie.....	49

Remerciements

*Nous adressons avant tout mes remerciements à notre dieu le créateur
de l'univers pour son aide et son courage pour se travailler.*

*Nous tenons à exprimer nos vifs remerciements à notre encadreur
Mr LALAOUI Lahouaoui
pour son entière disposition et ses judicieux conseils.*

*Nous exprimons nos remerciements aux membres de jury de nous
avoir honoré en acceptant de juger notre travail.*

*Nous tenons à exprimer nos vifs remerciements à tout les enseignants
de département d'Electronique d'Université de M'sila*

.

*Enfin nous remercions toute personne ayant participé de près ou de
loin à l'élaboration de ce travail.*

Dédicaces

*Je dédie cet humble travail
À mes très chers parents pour leur grand
amour et leurs sacrifices ;
À tout mes frères ;
À mes très chères sœurs ;
À ma très chère épouse ;
À toute ma famille ;
À mes fidèles amis
A tous mes amis.*

Liste des figures

Figure I.3: Schéma illustratif d'une image couleur.....	4
Figure I.4 : Représentation de l'histogramme.....	5
Figure I.5 : Représentation graphique de l'espace de couleurs RGB.....	6
Figure I.6 : Espace de couleur YCbCr.....	7
Figure I.7: Binarisation basée sur l'histogramme.....	11
Figure I.8 : présente une illustration d'une image avant binarisation.....	12
Figure I.9 : présente une illustration d'une image après binarisation.....	12
Figure I.10 : Image sémantique de binairisation.....	13
Figure I.11: Les trois couleurs primaires.....	16
Figure I.12: Intensité de la lumière blanche.....	17
Figure II.1 : Exemple de forme de masque.....	21
Figure II.2: Le produit de convolution entre l'image I et le masque K	22
Figure II.3: Exemple de masques de convolution de taille (3×3)	23
Figure II.4: Exemple de filtre Médian.....	29
Figure II.5: Exemple du filtre Sobel.....	31
Figure II.6 : Exemple du filtre Prewitt.....	32
Figure II.7 : Exemple du filtre Robert.....	33
Figure III.1 : Le schéma synoptique de l'algorithme proposé pour défilement de texte.....	35
Figure III.2 : illustration de détecteur de Robert.....	36
Figure III.3 : illustration de détecteur de Sobel.....	36
Figure III.4 : illustration de détecteur de Prewitt.....	37
Figure III.5 : Le texte défilant vérifié avec 30% de frontière.....	38
Figure III.6 : calcul de Différentiel Temporel Adaptative.....	40
Figure III.7 opération de dilation.....	43
Figure IV.1 illustration du processus de l'algorithme étape par étape.....	46

Liste des tableaux

Tableau II.1 Formats standard vidéo.....	20
---	----

Liste des abréviations

DSP: Digital Signal Processing

Pixel: Picture Element

RGB: Red Green Blue

YCrCb: Y (rouge+vert+bleu) et Cr (Y-rouge) et Cb (Y-bleu)

NG: Niveau de Gris

bit: Binary Digit

JPEG: Joint Photo Expert Group

PNG: Portable Network Graphics

TIFF: Tag Image File Format

GIF: Compuserve Graphics

USA: United Status of America

CCD: Charge Coupled Device

NTSC: National Television Standard Committee

PAL: Phase Alternation by Line

SECAM: Séquentiel Couleur à Mémoire

DV: Digital Video

TVN: Télévision Numérique

MPEG: Moving Picture Experts Group

YUV: Y (luminance) et U, V (chrominance)

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Depuis quelques années, les documents audiovisuels numérisés sont de plus en plus fréquents. Des grandes bases de données audiovisuelles ont été créées par des entreprises, des organisations et aussi par des personnes privées. Aujourd'hui, la télévision (TV) est la forme la plus populaire de divertissement et la principale source d'information dans le monde. Défilement de texte est largement utilisé pour intégrer des données supplémentaires au sein de trames vidéo. Pour éviter les interférences entre les programmes de vidéo et des informations complémentaires, texte défilant est généralement situé à la limite des cadres. Pour répondre à cette exigence, de nombreux pays ont défini des règles de radiodiffusion désignant l'emplacement du texte de défilement. Texte défilant peut être situé sur la partie supérieure ou inférieure de cadres, se déplaçant de droite à gauche ou de gauche à droite. Alternativement, texte défilant peut être situé sur le côté gauche ou droit déplaçant vers le haut ou vers le bas. Si la position du texte défilant peut être détecté, le contenu du texte peut fournir des données importantes et de fournir une source d'information supplémentaire de l'indice ainsi que des indices pour la classification de la vidéo. Les informations de texte peuvent être enregistrées sur le disque et fouillé à l'aide des indices pour obtenir des informations relatives au contenu vidéo.

En raison de la grande complexité de contenu vidéo, la précision de détection tendance à se dégrader lorsque le texte apparaît sur un fond riche. Pour augmenter le contraste entre le texte et l'information de fond, le texte peut être présenté comme un corps noir avec un fond blanc ou comme un corps blanc avec un fond noir. Toutefois, dans les situations de télévision les plus pratiques, le fond est rarement noir pur ou blanc pur. Le faible contraste causé lorsque le niveau de gris de l'arrière-plan est proche de celle du texte peut rendre la détection de texte difficile.

La plupart des travaux traite le problème de la détection et de la localisation du texte (nommé “détection”) ainsi, il existe très peu de recherche récente ont été présenté des algorithmes pour la détection du texte dans les images fixes. En revanche, cette étude a examiné le problème de la détection de défilement de texte dans les vidéos.

Texte défilant se déplace dans différents cadres, ce qui permet l'utilisation de corrélation temporelle d'identifier les caractéristiques différentielles entre inter-cadres.

Malheureusement, la vitesse de déplacement ne correspond pas nécessairement parmi les chaînes de télévision. Ainsi, cette étude propose sélection du cadre d'adaptation pour localiser des informations liées de texte supplémentaire. Nous avons également développé de pré-filtrage et méthodes d'amélioration de frontière de texte pour augmenter la précision de la détection et la suppression adaptative du bruit a été utilisée pour différencier les pixels de bruit à partir de pixels de texte.

Le travail présenté dans ce mémoire est organisé en quatre chapitres structurés comme suit: Le premier chapitre décrit quelques outils de base utilisés pour le traitement d'image et vidéo. Le second chapitre présente les différentes techniques de détection du contour à base des opérateurs de calcul de gradient, produit de convolution et dérivé de premier ordre, afin d'extraire des zones du texte dans une image. Le troisième chapitre présente un algorithme efficace pour la détection de défilement du texte sur une trame continue. Dans le dernier chapitre, nous présentons notre simulation et les résultats obtenus en appliquant l'algorithme proposé.

Et enfin notre travail se clôture par une conclusion générale qui présente un résumé du travail accompli, et les perspectives à donner à ce présent travail.

CHAPITRE I

Traitement d'Image et Vidéo

I.1 Introduction

On désigne par traitement d'images et vidéo l'ensemble des techniques permettant de modifier une image numérique dans le but de l'améliorer (qualitativement ou quantitativement avec le codage à compression) ou de détecter et d'extraire les zones de texte liées à la vidéo.

Le traitement d'images et vidéo est une discipline nouvelle qui s'est développée rapidement grâce à l'émergence des nouvelles technologies de l'information. Il s'appuie notamment sur les mathématiques liées à l'information, le traitement du signal, les systèmes électroniques, et sur l'avancée des capacités de calcul des microprocesseurs, notamment ceux qui sont développés exclusivement pour le traitement de signal et qui offrent de grandes capacité et vitesse de calculs (DSP, ...). La jeunesse du traitement d'images numériques et le vaste champ d'application qu'il s'est vu attribué ont fortement contribué à la nécessité de créer une méthodologie et de séparer les domaines d'applications. Le traitement d'image s'est ainsi distingué en fonction de quatre domaines d'applications majeurs : l'analyse, la synthèse, le codage et l'amélioration.

Ce chapitre comporte la partie théorique du traitement d'images et vidéo. Cette opération a pour objectif d'extraire d'un texte à partir d'une séquence des trames continues permettant de les représenter sous une forme plus condensée et facile.

I.2 Traitement D'image

Les images constituent l'un des moyens les plus importants qu'utilise l'homme pour communiquer avec ses semblables. Les efforts récents ont élargi même cette communication aux machines. Le traitement d'images est l'ensemble des méthodes qui cherchent à rendre cette opération possible, plus simple, plus efficace et plus agréable. Le traitement d'images est né de l'idée et de la nécessité de remplacer l'observateur humain par la machine. Les images ont alors été numérisées pour pouvoir être traitées par l'ordinateur à travers l'extraction et l'analyse de caractéristiques à partir des pixels. On peut dire que l'opération de traitement d'images, qui a pour objectif d'extraire à partir d'une image numérique des indices visuels ou des primitives pertinentes permettant de les représenter sous une forme plus condensée et facilement exploitable, afin de permettre un suivi de cible en mouvement. Parmi les objectifs du traitement d'image [1] :

- Améliorer la qualité de l'image : subjectivement (la rendre plus agréable à l'œil), objectivement (améliorer son contraste, ressortir certaines formes ou zones...etc.).

- Retrouver une image idéale : corriger les distorsions géométriques ou photométriques d'un capteur, corriger le flou dû à l'instabilité de l'appareil de prise de vue... etc.
- Chercher à détecter la présence de certaines formes (contours, textures de modèles connus) en préservant les autres informations contenues dans l'image.
- Réduire la quantité d'information par codage ou compression.
- Compréhension de l'image pour doter la machine de capacités visuelles.

I.2.1 Définition d'Une image

Il existe plusieurs définition d'une image numérique, elle peut être définie comme suit: une image numérique est définie comme un signal fini bidimensionnel échantillonné et composée d'unités élémentaires (appelé pixel) représentant chacun une portion de l'image.

I.2.1.1 Pixel

Un pixel (contraction de "*Picture element*") est le nom associé à une unité de base de l'image qui correspond à un pas de discrétisation. Un pixel est caractérisé par sa position et sa valeur (*i.e.* son niveau de gris).

I.2.1.2 Résolution

La résolution spatiale est la taille de l'image en pixels et correspond au degré de finesse des détails distinguables dans l'image. En théorie, la plus grande résolution spatiale est préférable, puisqu'elle correspond à une image plus fine. En pratique, la taille de l'image sera limitée par :

- La capacité des systèmes de transmission.
- Les systèmes de stockage.
- Les systèmes de calcul.

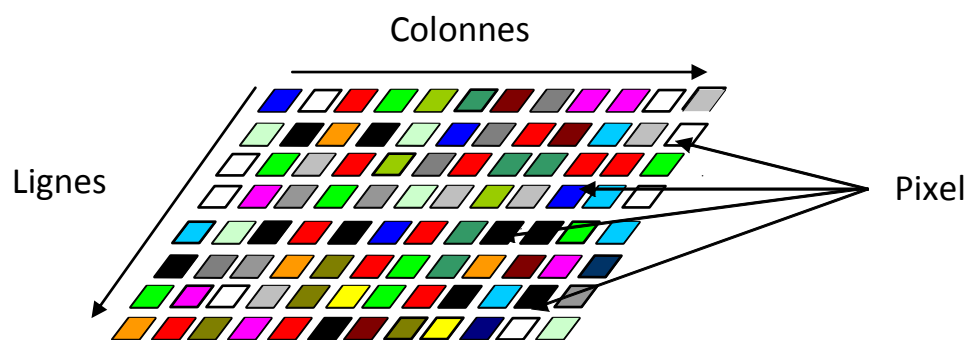


Figure I.3: Schéma illustratif d'une image couleur.

I.2.2 Opérateurs ponctuels

I.2.2.1 L'histogramme

Si on travaille sur une image en niveaux de gris avec 256 niveaux alors on obtient 256 nombres entiers.

$$I \rightarrow H(i): i = 0, 1, \dots, 255 \quad (I.5)$$

H_i : Nombre d'occurrences du niveau de gris « i » dans l'image.

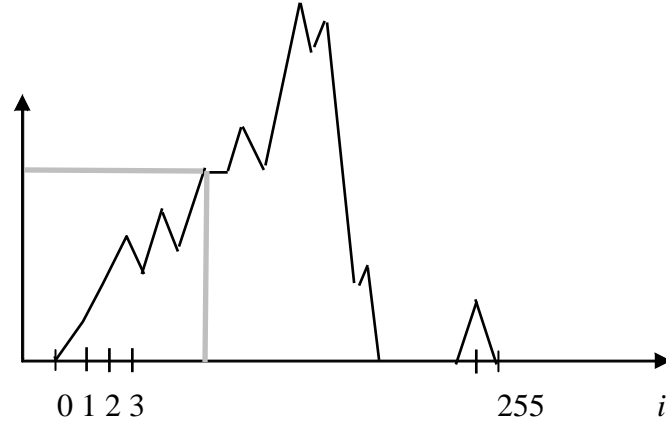


Figure I.4 : Représentation de l'histogramme.

L'étude d'une image numérique débute le plus souvent par le calcul et l'analyse de son histogramme. On obtient ainsi des informations sur la dynamique réelle de l'image (nombre de niveaux de gris réellement utilisés) et donc le contraste disponible, le caractère aléatoire ou non de la distribution des niveaux de gris et la similitude de l'histogramme (ou de morceaux) avec une distribution statistique connue (par exemple une gaussienne), la présence de pics significatifs,....

Tous les paramètres statistiques sur les niveaux de gris peuvent être calculés à partir de l'histogramme :

$$\text{Moyenne : } \mu = \frac{1}{256} \sum_{i=0}^{255} H(i) \quad (I.6)$$

$$\text{Variance : } \sigma^2 = \frac{1}{256} \sum_{i=0}^{255} H(i) \times (i - \mu)^2 \quad (I.7)$$

I.2.2.2 L'histogramme cumulé

C'est une fonction du niveau de gris i telle que :

$$I \rightarrow H_c(i) = \sum_{k=0}^i H(k) \text{ Avec } H(k) \text{ valeur de l'histogramme} \quad (I.8)$$

$$H_c(0) = H(0) \text{ et } H_c(i) = H_c(i-1) + H(i) \quad i > 0$$

$H_c(255) = \text{nombre total de pixel de l'image.}$

Réciproquement on peut passer de $H_c(i)$ à $H(i)$ par la formule suivante :

$$H(0) = H_c(0) \dots \dots \dots H(i) = H_c(i) - H_c(i - 1) \quad (\text{I.9})$$

I.2.3 Espace de couleur

Il existe plusieurs systèmes pour la définition des couleurs projetées sur le moniteur d'un ordinateur. On se limite, dans ce cas, aux deux espaces de couleurs utilisés dans notre travail (RGB, et YCrCb) et aux relations qui les lient.

I.2.3.1 Le système RGB

C'est un mode de composition des couleurs basé sur le principe des couleurs additives. Le rouge, le vert et le bleu sont les trois primaires utilisés dans la constitution de couleurs à partir de la source lumineuse. Une image RGB est composée de trois couches où chacune est codée comme une image en niveau de gris avec des valeurs allant de 0 à 255.

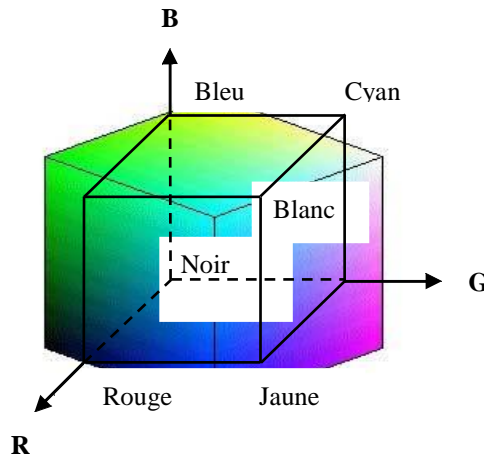


Figure I.5 : Représentation graphique de l'espace de couleurs RGB.

I.2.3.2 Le système de représentation YCrCb

La représentation des couleurs dans l'espace YCbCr est une transformation linéaire donnée par les équations (I.10). Ce système appartenant aux systèmes luminance chrominance, a été développé à l'origine pour assurer la compatibilité entre téléviseurs couleur et téléviseurs noir et blanc [1], puisque la composante luminance correspond aux nuances de gris, et les composantes de chrominance correspondent aux informations sur la

couleur. Une simple transformation linéaire permet de passer d'un système RGB à un système YCrCb :

$$\begin{aligned} Y &= rR + vG + bB \\ C_b &= a_1(R - Y) + b_1(B - Y) \\ C_r &= a_2(R - Y) + b_2(B - Y) \end{aligned} \quad (\text{I.10})$$

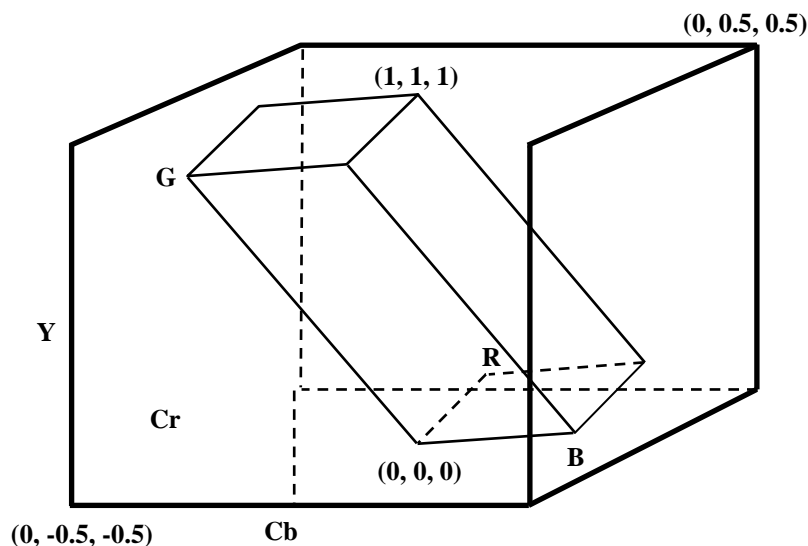


Figure I.6 : Espace de couleur YCbCr.

I.2.4 Arithmétique de l'image

L'arithmétique d'images consiste à appliquer l'opération arithmétique standard et les opérations logiques à une ou plusieurs images. Les opérations sont en fait appliquées pixel par pixel. L'image résultat d'une opération arithmétique ou logique est donc telle que la valeur de chaque pixel résultat ne dépend que des valeurs des pixels opérande correspondants dans les images d'entrée. Les images opérandes doivent donc être de même taille, de plus une des images opérande peut être une image constante.

I.2.4.1 Soustraction

$$I(x, y) = I_1(x, y) - I_2(x, y) \quad (\text{I.11})$$

La soustraction peut aussi se faire en valeur absolue (par exemple deux images temporelles et la différence donne ce qui a bougé).

$$I(x, y) = \text{abs}(I_1(x, y) - I_2(x, y)) \quad (\text{I.11})$$

La façon d'implanter cet opérateur est liée à la manière de prendre en compte les résultats négatifs ou bien supérieurs à 255 (si l'on travaille sur des octets).

I.2.4.2 Le ET logique

L'opération de et logique entre deux images peut s'appliquer sur des images binaires : Le et logique est calculé entre les deux pixels directement ou bien sur des images de niveaux de gris; l'opération est réalisée bit par bit sur les huit bits représentant le pixel.

I.2.5 Amélioration des images

L'étape d'amélioration peut être décomposée en deux sous approches, la première est celle appliquée sur d'images brutes (RGB...etc.), c'est l'amélioration par les méthodes de filtrage, l'objectif de ces méthodes est l'élimination du bruit introduit pendant la transmission des images ou par le matériel lui-même ...etc. Parmi les filtres les plus utilisés on cite: les filtres linéaires, le filtre moyen (sur de petites fenêtres de 3x3 ou 5x5 pixels), le filtre gaussien, et les filtres non linéaires. La deuxième approche est : lorsque l'image est passée par l'étape de binarisation, elle est suivie par des opérateurs morphologiques, tel que: l'érosion, la dilatation, l'ouverture et la fermeture.

I.2.6 Opérateurs locaux

Ces techniques permettent de supprimer le bruit (opération de lissage) et/ou de mettre en évidence et détecter les points frontières. Les méthodes correspondantes s'appuient sur des *filtrages* de l'image. Ce filtrage peut être fréquentiel (passage par la transformée de Fourier par exemple) ou spatial (convolution de l'image avec un filtre).

L'image possède dans sa nature une redondance spatiale en information, les pixels voisins en 4-voisinage ou en 8-voisinage possèdent les mêmes caractéristiques de point de vue niveau de gris. Une présence de bruit dans une image provoque une variation brutale de niveau de gris d'un pixel par rapport aux autres voisins.

Il existe différents types de filtres et la taille et le contenu du filtre vont correspondre à divers types d'opérations :

- Lissage d'images : éliminer les pixels isolés considérés comme bruit.
- Calcul de gradient (dérivée première), de Laplacien (dérivée seconde).

I.2.6.1 Filtrage spatiale

Le filtrage spatial consiste à remplacer la luminosité d'un pixel de l'image par une valeur qui a été déterminée grâce au voisinage de ce pixel. Les filtres peuvent être séparés en deux catégories, filtres linéaires et filtres non linéaires [2].

Le filtrage spatial d'une image I consiste à convoluer cette image considérée comme une fonction $I(x,y)$ avec une fonction $w(x,y)$ qui représente la réponse impulsionnelle du filtre (le masque). Le résultat est une nouvelle image dite image du filtrée [3].

I.2.6.2 Convolution

Pour mettre en œuvre un filtrage avec des filtres linéaires, on utilise un opérateur mathématique nommé convolution (noté \otimes). La convolution consiste donc à prendre un voisinage du pixel que l'on veut modifier, le plus souvent ce voisinage est compris dans une fenêtre carrée de côté de longueur impair, de multiplier chaque valeur du voisinage par des coefficients appartenant à un masque de convolution et de faire la somme qui deviendra la nouvelle valeur du pixel [2].

Ce traitement est effectué pour chaque pixel de l'image, mis à part les bords de l'image pour lesquels, soit on met les pixels à 0, soit on ne les filtre pas, soit on les enlève,...

$$I'(x,y) = I(x,y) \otimes W(i,j) = \sum_{m=-d}^d \sum_{n=-d}^d W(m,n)I(x+m,y+n) \quad (I.12)$$

Où la taille du masque de convolution est $(2d+1) \times (2d+1)$.

I.2.6.3 Filtres linéaires

Un filtre est linéaire s'il vérifie :

$$\text{Filtre}(c1.\text{image1}+c2.\text{image2}) = c1\text{Filtre}(\text{image1}) + c2\text{Filtre}(\text{image2}) \quad (I.13)$$

Les filtres linéaires les plus connus sont les filtres Passe-haut, passe-bas, Sobel, Prewitt et Laplacien. Dans tous les cas, ces filtres sont représentés sous forme de masque auquel est associé un coefficient diviseur qui permet de ramener les valeurs calculées entre 0 et 255 pour une image à 256 niveaux de gris [2].

I.2.6.4 Lissage

I.2.6.4.1 Filtre moyenneur

Cette méthode permet de lisser les images, c'est à dire de diminuer les différences de niveaux de gris entre pixels voisins. Le filtrage par la moyenne consiste à remplacer chaque pixel par la valeur moyenne de ses voisins (le pixel lui-même y compris). Cette méthode a pour effet de modifier les niveaux de gris trop différents de leurs voisins. En ce sens on peut penser supprimer le bruit, c'est-à-dire des niveaux de gris anormaux. Suivant l'effet du lissage que l'on veut réaliser, on choisira une taille de filtre plus ou moins grande (3x3, 5x5,...) mais on doit comprendre que les contours de l'image de départ deviendront alors plus flous.

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Fenêtre 3×3

1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25

Fenêtre 5×5

Plus le filtre grossit (la taille du masque agrandis), plus le lissage devient important et plus le flou s'accroît.

1-Un pixel isolé avec un niveau de gris anormal pour son voisinage va perturber les valeurs moyennes des pixels de son voisinage.

2-Sur une frontière de régions le filtre va estomper le contour et le rendre flou, ce qui est gênant en visualisation bien sûr mais éventuellement aussi pour un traitement ultérieur qui nécessiterait des frontières nettes.

Il est possible de moduler ces effets néfastes en réalisant en chaque pixel une convolution conditionnelle, par exemple en un pixel de niveau de gris NG1, on applique le filtre, supposons obtenir une valeur NG2, alors on décidera d'appliquer le filtre que si :

$$|NG1 - NG2| \geq \text{seuil}. \quad (\text{I.14})$$

I.2.6.4.2 Filtre médian

Le filtre médian réalise un lissage de l'image un peu plus performant que le filtre moyenne en ce qui concerne les détails dans l'image.

Ce filtre est très utilisé pour éliminer le bruit sur une image qui peut être de différentes origines (poussières, petits nuages, baisse momentanée de l'intensité électrique sur les capteurs,...) et qui se traduit par des taches de faible dimension dont la distribution sur l'image est aléatoire. L'avantage de ce filtre réside dans le fait qu'il conserve les contours alors que les autres types de filtres ont tendance à les adoucir.

P1	P2	P3
P4	P5	P6
P7	P8	P9

Si on prend une fenêtre de l'image donnant les niveaux de gris comme représentés par le tableau précédent, on classe les valeurs des pixels (P_1, P_2, P_3, \dots) par ordre croissant et on prend celui du milieu (la médiane). Le pixel P_5 sera alors remplacé par cette valeur médiane [2].

I.2.7 Binarisation

Une image binaire est celle pour laquelle chaque pixel ne peut avoir pour valeur que 0 ou 1. Elle offre plusieurs avantages tels que : la réduction de l'espace mémoire (pixel codé sur 1 bit), isolation des objets sur un fond...etc. Pour l'obtention d'une image binaire à partir d'une image en niveaux de gris, des techniques de seuillage sont utilisées.

Le seuillage est une opération qui ramène une image brute à plusieurs niveaux à une autre à deux niveaux seulement. La méthode la plus adoptée est celle basée sur l'histogramme de l'image. Le problème principal de cette méthode est le choix du seuil (ou de l'intervalle de binarisation). Avec un intervalle trop large, on obtient une image binaire contenant des pixels qui ne font pas partie des objets que l'on veut extraire. Généralement il s'agit de bruit, ou des structures d'une autre nature, qui ont un niveau de gris proche de celui des objets recherchés. Avec un intervalle trop étroit, on obtient une image binaire dont certains objets d'intérêt n'apparaissent pas où apparaissent partiellement.

Une binarisation basée sur l'histogramme est efficace si les pixels de l'objet d'intérêt ont un niveau de gris proche d'une même valeur s , tandis que ceux des autres structures ont presque toujours des niveaux de gris éloignés de s .

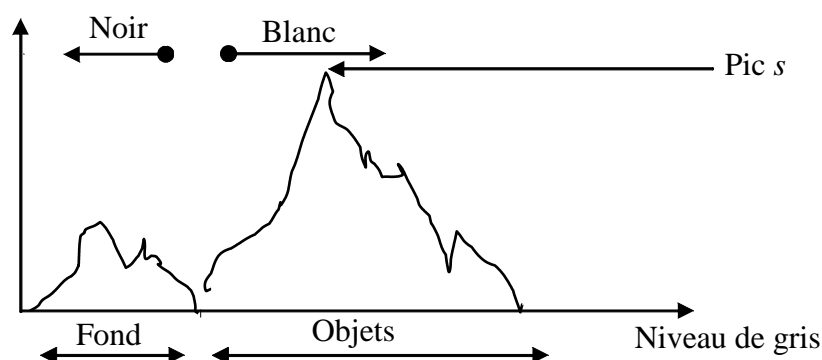


Figure I.7: Binarisation basée sur l'histogramme.

Les avantages qu'offrent les images binaires par rapport aux images multi-niveaux sont multiples :

1. Espace mémoire réduit (pixel codé sur 1 bit au lieu de 8 bits) ;

2. Simplicité des opérateurs appliqués à leur traitement ;
3. Isolation des objets sur un fond (applications industrielles) ;
4. La binarisation est considérée comme étape de segmentation.

Exemple :

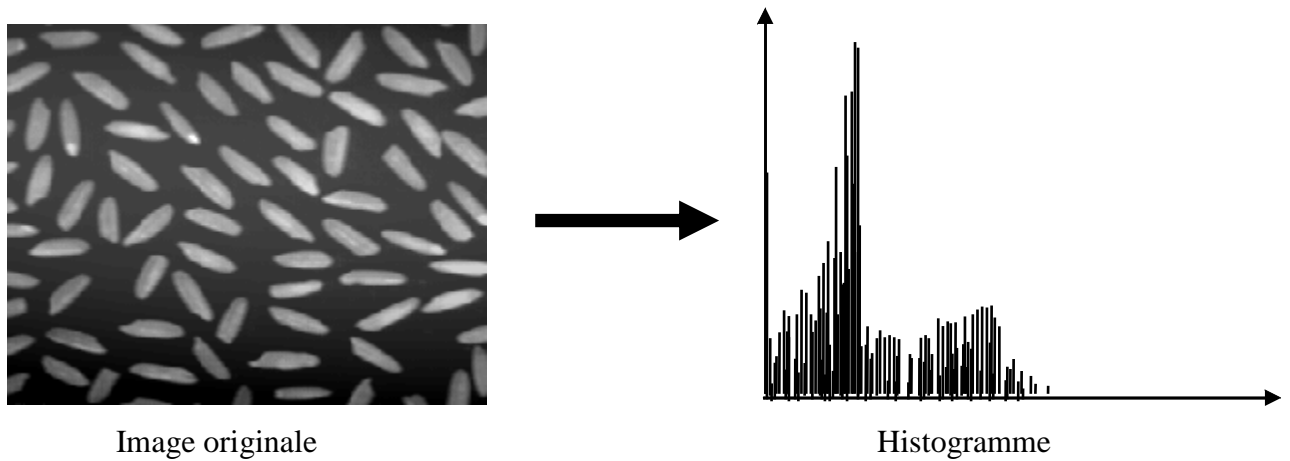


Figure I.8 : présente une illustration d'une image avant binarisation.

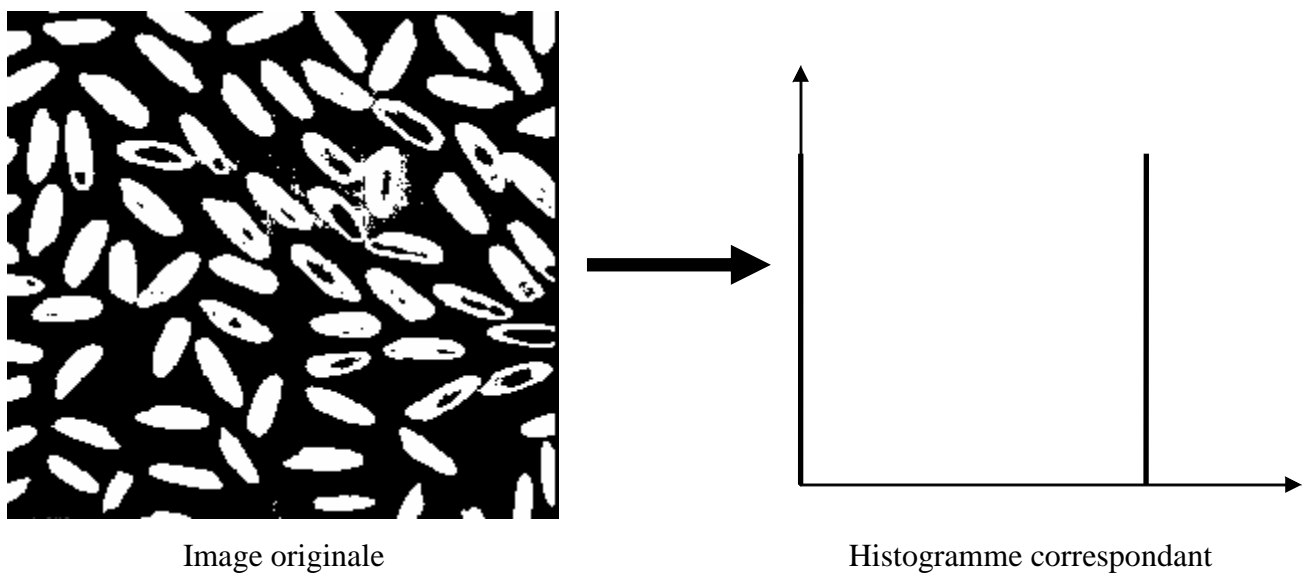


Figure I.9 : présente une illustration d'une image après binarisation.

Par suite, une image binaire peut être représentée par une matrice booléenne dont chaque élément signifie vrai (1 = blanc) ou faux (0 = noir).

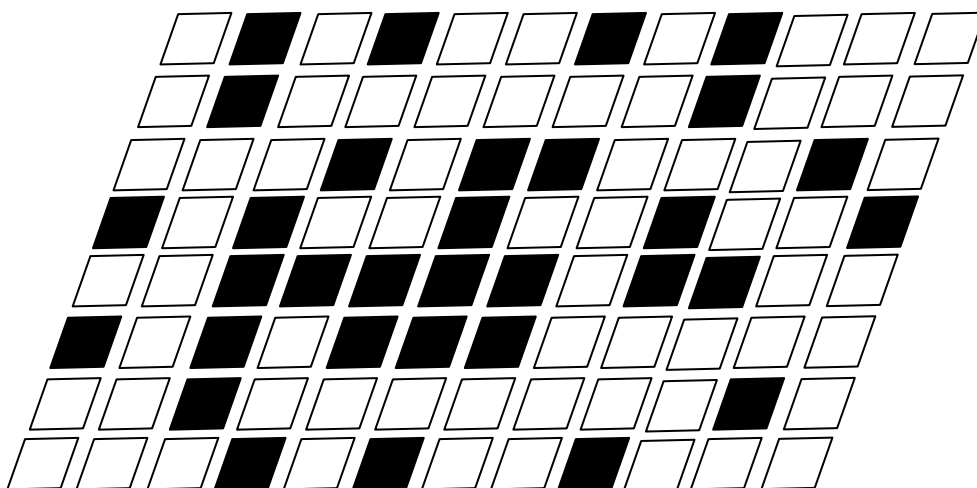


Figure I.10 : Image sémantique de binairisation.

I.2.8 Opérateur morphologique pour l'amélioration des images binaires

Il existe deux opérateurs morphologiques de base : l'érosion et la dilatation.

L'ouverture et la fermeture ne sont que des combinaisons de ces opérateurs. Ils sont utilisés pour l'amélioration des images binaires telle que l'élimination des points de petites tailles considérés comme des bruits.

I.2.8.1 L'élément structurant

L'élément structurant est un masque binaire (composé de 1 et de 0) utilisé par la plupart des fonctions morphologiques pour déterminer quel voisin d'un pixel contribuera à sa nouvelle valeur. Il est destiné à pondérer l'effet d'une transformation morphologique en fonction de la forme des contours des objets concernés.

I.2.8.2 Dilatation

La dilatation consiste à éliminer les points noirs isolés : on dilate les parties blanches ce qui élimine les points noirs. L'opération de dilatation est définie par [4]:

$$I \otimes S = \{x[(\hat{S})_x \cap I] \neq \Phi\} \quad (\text{I.15})$$

Où I est l'image qui sera dilatée, S est un ensemble de points de coordonnées connus. L'effet de base de l'opérateur de dilatation sur une image binaire est d'agrandir progressivement les limites des régions noires de l'image. Donc les régions blanches de l'image grandissent en dimension.

I.2.8.3 Erosion

Il s'agit ici d'éliminer les points blancs isolés. La méthode est assez similaire à celle de la dilatation. L'opération d'érosion est définie par [4] :

$$I \ominus S = \{x[(\hat{S})_x \subseteq I]\} \quad (\text{I.16})$$

L'effet de base de l'opérateur de l'érosion est d'éroder les limites des régions d'images élémentaires de couleur blanche. Donc les régions d'images élémentaires de couleur blanche diminuent en dimension, et les régions élémentaires de couleur noire deviennent plus grandes.

La dilatation et l'érosion peuvent se combiner pour créer d'autres opérateurs morphologiques plus complexes pouvant résoudre une variété de problèmes [4].

Par définition, l'ouverture est une opération d'érosion suivie d'une dilatation, elle est définie par [5] :

$$B \circ I = (B \ominus I) \oplus I \quad (\text{I.17})$$

La fermeture est une opération de dilatation suivie d'une érosion, elle est donnée par [4]:

$$A \bullet I = (A \oplus I) \ominus I \quad (\text{I.18})$$

I.2.9 Format d'images

Un format d'image comprend en général un en-tête qui contient des données sur l'image (taille de l'image en pixels par exemple) suivie des données de l'image. La structuration des données est différente pour chaque format d'image [6].

I.2.9.1 JPEG (Joint Photo Expert Group)

Ce format est la représentation d'une norme ISO. Il devrait subir une prochaine évolution nommée JPEG 2000 avec un mode de compression différent, qui devrait permettre de stocker plusieurs résolutions d'une même image dans un même fichier. De plus, l'adjonction de métadonnées devrait être possible [6].

I.2.9.2 JPEG2000 (Joint Photo Expert Group 2000)

Le format JPEG représente un inconvénient majeur : la qualité de l'image après modification diminue. C'est pourquoi, les membres du groupe JPEG, se sont remis à travailler sur l'élaboration d'un nouveau format de codage d'images plus performant : le JPEG2000 [6]. Entre autres possibilités, le format JPEG2000 permet, dans une zone définie, des

manipulations géométriques tels que la rotation, la permutation du paysage par rapport au portrait. Le format JPEG2000 permet de définir une ou plusieurs zones d'une image afin d'en préserver la qualité. En effet, ce système permet de dégrader les régions les moins stratégiques de l'image tout en gardant intact la zone sélectionnée.

De plus, un code d'erreur, permet de détecter si une erreur est survenue lors de la transmission de l'image. Malheureusement, le JPEG2000 ne permet pas la correction de ses erreurs [6].

Le format JPEG 2000 permet d'incorporer des métadonnées (un dossier est associé à l'image contenant des informations relatives à la sécurité, à l'auteur...).

I.2.9.3 PNG (Portable Network Graphics)

Ce format est l'émanation de recommandations du consortium W3 (1996), c'est dire s'il est recommandé de l'utiliser même si son principal inconvénient est de ne pas être reconnu par les navigateurs d'ancienne génération (niveau 3 et inférieurs).

I.2.9.4 TIFF (Tag Image File Format)

Attention aux différentes (pour ne pas dire multiples) versions existantes de ce format d'images qui rendent sa diffusion parfois difficile [6].

I.2.9.5 GIF (Compuserve Graphics)

Ce format est sous brevet Unisys. Il s'agit donc d'un format "propriétaires", avec tous les inconvénients des formats "propriétaires"...

I.3 Application de traitement d'image

L'utilisation du traitement d'images touche des domaines très variés :

- **La robotique:** dont les tâches usant de l'imagerie sont principalement le contrôle de qualité ainsi que la robotique mobile et la robotique aérienne.
- **Les technologies biomédicales (médecine) :** dont l'exemple le plus connu utilisant le traitement d'images est le scanner.
- **L'imagerie aérienne et spatiale:** dont les traitements concernent l'amélioration des images satellitaires, l'analyse de ressources terrestres et les analyses météorologiques.

Le traitement numérique des images se résume en un ensemble de techniques permettant la transformation des images numériques. Il commence par un système d'acquisition (caméra) suivi par une partie traitement, analyse afin de comprendre la scène acquise.

I.4 Vidéo

Un flux vidéo est composé d'une succession d'images, 25 par seconde en Europe et 30 par seconde aux USA. Chaque image est décomposée en lignes horizontales, chaque ligne pouvant être considérée comme une succession de points.

La lecture et la restitution d'une image s'effectue donc séquentiellement ligne par ligne comme un texte écrit : de gauche à droite puis de haut en bas.

I.4.1 La vision

Dans le phénomène de la vision humaine, un flux lumineux composé de photons frappe l'œil. Dans l'œil, les bâtonnets vont réagir à l'intensité de la lumière (la luminance ou Y) et les cônes vont réagir à la couleur (la chrominance ou C). Le cerveau effectue la synthèse de ces informations pour composer une image. L'œil humain est davantage sensible à l'intensité lumineuse (Y) qu'à la couleur (C).

I.4.2 La couleur

La vidéo utilise ce qui est identifié comme la synthèse additive de la couleur. Dans le système colorimétrique additif, les trois couleurs primaires sont le rouge, le vert et le bleu (RVB ou RGB dans la terminologie anglophone). En combinant ces trois couleurs primaires, il est possible de reproduire tout le spectre visible par l'œil humain. L'écran de visualisation vidéo sera donc composé d'une série de triades rouge-vert-bleu. L'activation de l'ensemble de ces triades formera l'image.

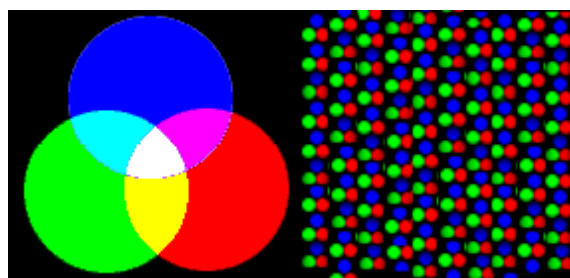


Figure I.11: Les trois couleurs primaires.

I.4.3 Le signal

En vidéo, c'est la caméra qui transforme l'information lumineuse (photons) en signal électrique (électrons). En vidéo analogique, l'intensité de ce signal électrique varie de façon continue. Le processus de transformation de l'information lumineuse en signal électrique est le

suivant :

- L'élément optique de la caméra, l'objectif, sépare la lumière en trois composants le rouge, le vert, le bleu. Cette opération est réalisée en faisant passer le flux lumineux par une succession de filtres dichroïques réfléchissant certaines couleurs et en laissant passer d'autres.
- En vidéo professionnelle, les trois images sont projetées sur trois capteurs photosensibles distincts formés chacun de centaines de milliers de points, généralement entre 400 000 et 700 000. Ces capteurs sont nommés CCD (charge coupled device) ou dispositifs à transfert de charge. Les caméras domestiques ne sont généralement équipées que d'un seul capteur CCD.
- Pour chacun des points de chacun des capteurs, l'énergie lumineuse sera transformée en énergie électrique. Ainsi, à la sortie des capteurs, trois signaux électriques d'intensité variable, un signal pour chacune des trois composantes. La lumière blanche est formée par la somme des trois composantes RVB. Toutefois, elle n'est pas composée des trois couleurs primaires en quantité égale. Les proportions sont suivantes :

$$100\% \text{ Y} = 29,9\%(R) + 58,7\%(V) + 11,4\%(B) \quad (\text{I.1})$$

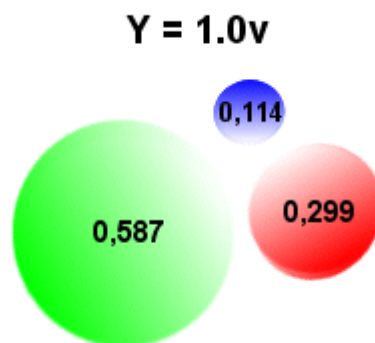


Figure I.12: Intensité de la lumière blanche.

- Une information supplémentaire devra être ajoutée à ces signaux. Comme nous l'avons dit, une valeur doit être établie pour chacun des points des capteurs. C'est cette valeur pour chacun des points de chacun des pixels qui permettra de recomposer

l'image sur un écran de visualisation, chacun des points de la surface d'affichage pouvant alors prendre la même valeur qu'au moment de la capture par la caméra. La restitution de l'information pour chacun des pixels se fait par balayage ligne par ligne de la surface de l'écran de reproduction, les valeurs enregistrées pour chacun des pixels au moment de la capture étant transférées aux pixels de la surface de reproduction. Pour synchroniser ces balayages, un signal de fin de ligne est ajouté pour chacune des lignes. Un signal de fin d'image est également inséré.

I.4.4 L'affichage

La technique de la composition d'image par balayage a une conséquence : contrairement à l'image cinéma, l'image vidéo ne s'affiche pas d'un seul coup, elle s'affiche ligne par ligne, en commençant par le haut à gauche pour finir en bas à droite. Afin d'assurer une uniformité dans la luminosité de cet affichage, en d'autres termes afin que l'intensité lumineuse des pixels du haut de l'écran ne commence pas à diminuer alors que ceux du bas viennent d'être activés, une technique spécifique a été développée. Il s'agit de la technique d'entrelacement. Un premier balayage est effectué sur les lignes impaires, puis un deuxième sur les lignes paires. Il faut donc deux passages pour recomposer une image complète. La fréquence d'entrelacement s'appuie sur la fréquence du réseau électrique. Comme en Amérique du Nord la fréquence des réseaux est de 60Hz, le système de balayage est donc de 60 demi-images, soit 30 images par seconde. De façon précise, le taux de rafraîchissement est plutôt de 29,97 images par seconde.

La norme Nord-Américaine en télévision est le NTSC (National Television Standard Committee). L'image NTSC est formée de 525 lignes. De ces lignes, seulement 486 sont utilisées pour afficher l'image. Les deux autres principales normes sont européennes. Il s'agit des normes PAL (Phase Alternation by Line) et SECAM (Séquentiel Couleur à Mémoire). L'image PAL ou SECAM est formée de 625 lignes dont 576 sont utilisées pour l'image. Comme la fréquence des réseaux électriques européens est de 50 Hz, le système de balayage est de 50 demi-images, soit 25 images par seconde.

I.4.5 Le son

Le son fera l'objet d'un traitement distinct. Le signal sonore sera distinct du signal vidéo. Il sera également enregistré sur des pistes distinctes sur le ruban vidéo.

I.4.6 Les formats d'enregistrement

En vidéo analogique, les signaux vidéo et audio seront enregistrés sur ruban magnétoscopique. Il existe différents formats d'enregistrement.

Le format RVB : Les trois informations pour chacune des trois couleurs sont encodées distinctement, véhiculées sur des câbles séparés et accompagnées d'un signal de synchronisation.

Le format composante (ou YUV ou Y, B-Y, R-Y ou Y, Cr, Cb) : Les informations de luminance (noir et blanc) (Y) et les informations de chrominance (couleur, R-Y et B-Y) sont enregistrées sur des pistes différentes. Lors de l'enregistrement, on diminue la quantité d'information concernant la couleur. Pour y arriver, puisque nous savons que la luminance (Y) est composée de la somme des 3 chrominances (rouge, vert et bleu), il est donc possible de supprimer une des composantes couleurs et de la reconstruire à partir de la valeur Y et des deux autres valeurs, le bleu (B) et le rouge (R) :

$$Y = (0,299 \times R + 0,587 \times V + 0,114 \times B) \times 1 \quad (I.2)$$

$$B-Y = (-0,299 \times R - 0,587 \times V + 0,886 \times B) \times 0,306 = U \quad (I.3)$$

$$R-Y = (0,299 \times R - 0,587 \times V + 0,886 \times B) \times 0,877 = V \quad (I.4)$$

I.4.7 Le format DV (Digital Video)

Afin de permettre que les fichiers et les supports pour les enregistrer demeurent de taille raisonnable, les formats d'enregistrement vidéo numérique effectuent une opération de compression des données dans la caméra, au moment de la prise de vue. Il existe plusieurs formats de vidéo numérique. Le format sans le plus répandu en ce moment est le format DV pour *Digital Video*. Ce format est fondé sur une norme internationale créée par un consortium de 10 sociétés en 1993. Il existe des variantes du format DV, par exemple le format DVCAM ou DVCPRO. Mais les différences sont davantage liées au processus d'enregistrement des données sur un support physique qu'au processus de traitement et de compression du signal.

L'enregistrement se déroule selon les séquences suivantes [6] :

- L'élément optique de la caméra sépare la lumière vers trois composants : rouge, vert et bleu (RVB) et transmet l'information à un convertisseur. Le taux de transfert de la donnée est alors de 248Mops.



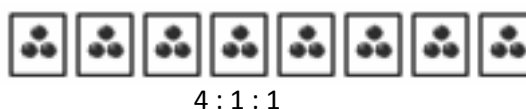
4 : 4 : 4

- Un convertisseur transforme le signal RGB en signal YUV (Y pour la luminance, U et V pour la couleur). Comme l'œil humain perçoit mieux la lumière que la couleur, lors

de l'échantillonnage chaque valeur Y d'un pixel est échantillonnée quatre fois, alors que les valeurs U et V ne le sont que deux fois, ce que l'on nomme la compression YUV 4:2:2. Ce premier échantillonnage réduit la taille de la donnée d'un tiers sans dégradation perceptible par l'œil. Le taux de transfert est alors réduit à 165Mops.



- Afin de réduire encore davantage le volume d'information, on utilise un convertisseur YUV 4:2:2 - YUV 4:2:0/YUV 4:1:1. Chaque pixel du signal résultant garde sa même valeur de lumière (Y), alors que quatre pixels se partagent la même valeur de couleur (U et V). Le débit est alors de 124Mops.



I.4.8 Formats standard vidéo

Analogiques	Numériques
-VHS	-Digital 8 -HDV
-VHSC	-DV ou Digital Vidéo - XDCAM HD
-SVHS	-DVCAM -DVCPRO HD
-8mm ou vidéo 8	-Betacam SX -DVD
-Hi8	-Digital S -Avchd
-Betacam/BetacamSP	-IMX
-UMATIC / U MATIC SP	-XDCAM
-Betamax	-DVCPRO
-V2000	-Betacam numérique
	-HDCam

Tableau I.1 : Formats standard vidéo.

I.5 Conclusion

Dans ce chapitre, nous avons présenté les notions de base du traitement d'images, et points de vu dans la partie théorique l'extrême importance du traitement d'images pour l'extraction des primitives dans une application basé sur la vidéo, pour cela les différentes approches possibles pour cette extraction sont détaillées. Nous avons commencé par des brefs théoriques sur la vidéo puis les méthodes de traitement d'images ainsi que les différents espaces d'images utilisés dans notre application, Dans le chapitre suivant, nous allons étudier les différentes techniques de détection du contour à base des opérateurs de calcule de gradient, produit de convolution et dérivé de premier ordre, afin d'extraire des zones du texte dans une image.

CHAPITRE II

Les filtres de traitement d'images

II.1 Introduction

Le principe du filtrage est de modifier la valeur des pixels d'une image, généralement dans le but d'améliorer son aspect. En pratique, il s'agit de créer une nouvelle image en se servant des valeurs des pixels de l'image d'origine. [7]

L'objectif principal du filtrage est d'améliorer la qualité visuelle de l'image (ex : atténuer et/ou éliminer le bruit) ou extraire des attributs de l'image (ex : contours), en modifiant la valeur de niveau du gris d'un pixel en fonction de la valeur de ses voisins (figure II.1), les éléments significatifs de l'image doivent être préservés au mieux. [8] [9] [10]

Les voisinages du pixel à traiter est fonction de la forme du masque choisi par le filtre.

La figure II.1 suivante donne quelques exemples de masques (les cases en vert : le pixel central, les cases oranges : les pixels voisins).

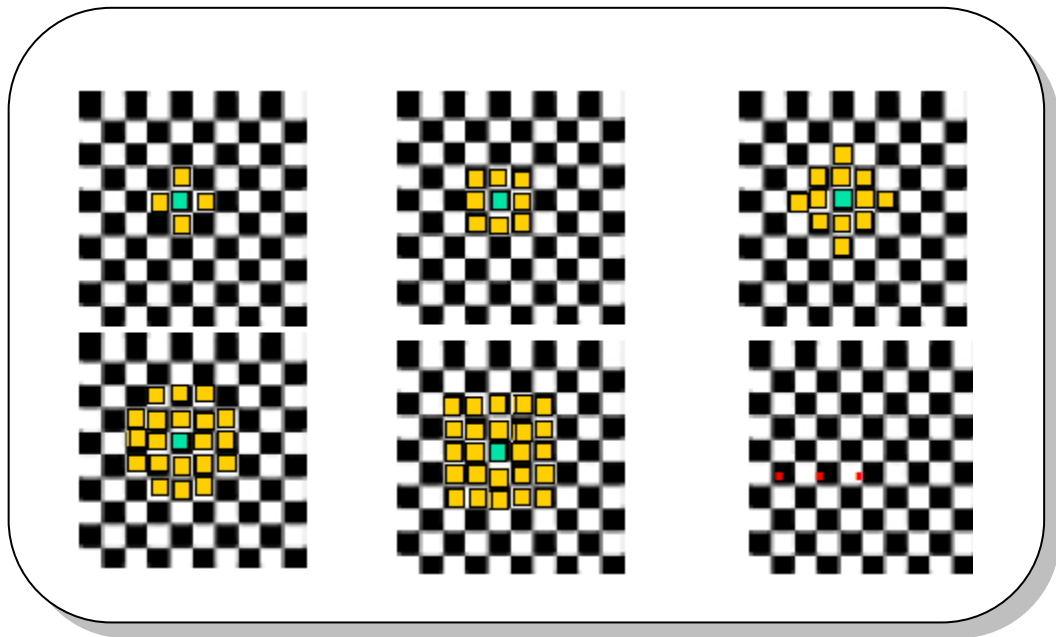


Figure II.1 : Exemple de forme de masque. [10]

Différentes méthodes de filtrage ont été développées suivant le type et l'intensité du bruit, ou les applications auxquelles on destine l'image.

Les premières et les plus simples de ces méthodes sont basées sur le filtrage linéaire stationnaire (invariant des transitions), mais les limitations de ces techniques (en particulier leur mauvaise conservations) a conduit au développement des filtres non linéaire.

Dans la plupart des cas présentés ici, le filtrage consiste à balayer l'image par une fenêtre d'analyse de taille finie. Le calcul du nouveau niveau de gris du pixel considéré ne prend en compte que les plus proches voisins de celui-ci. Et on les appelle les filtres de convolution [9].

II.2 Filtres de convolution

Plusieurs opérations de filtrages d'images utilisent la convolution. une convolution est caractérisée par un masque qui est une matrice de coefficients [11].

L'image numérique étant en quelque sorte une carte de pixels, on peut identifier chaque pixel par ses coordonnées X et Y et lui affecter une valeur liée à sa luminosité. On peut utiliser dans le cadre des images numériques une sorte un tableau de X colonnes et Y lignes qui réserve une place pour ranger la valeur de chaque pixel de l'image. En mathématique ce genre de tableau s'appelle une matrice, et les mathématiciens disposent d'outils pour effectuer des calculs sur les matrices, comme additionner deux matrices, les multiplier, etc...

Un produit de convolution, est un opérateur mathématique qu'on utilise pour multiplier les matrices entre elles.

Dans le cas qui nous intéresse, nous mettons en jeu deux matrices très différentes : la matrice image, très grande (par exemple $512 * 512$, ce qui représente 262144 pixels) et une matrice plus petite qu'on appelle le masque ou le noyau parce que c'est le "cœur" de tous les changements qui vont affecter l'image.

Le noyau ou le masque va donc agir sur chacun des pixels, c'est-à-dire sur chacun des éléments de la matrice "image".

I(1,1)	I(1,2)	.								*	K(1,1)	K(1,1)	K(1,1)
I(2,1)	.										K(2,1)	K(2,1)	K(2,1)
.		.									K(3,1)	K(3,1)	K(3,1)
.			I(x,y)										
									I(n,m)				

Figure II.2: Le produit de convolution entre l'image I et le masque K

Dans la figure ci-dessus, l'image est représentée par la matrice [i] composée de n*m éléments. Le masque est quand à lui composé de la matrice carrée [k] de 3*3 éléments.

Appliquer un filtre de convolution consiste à multiplier chacun des pixels de la matrice [i] par le masque [K] ; pour calculer la valeur d'un pixel I(x, y) de la matrice image, on multiplie sa valeur par celle du pixel central du masque k(2, 2) et on additionne ensuite la valeur des produits des pixels adjacents. Il reste ensuite qu'à diviser le résultat par le nombre d'éléments du masque, cette dernière opération n'appartient pas au produit de convolution proprement dit, mais elle est nécessaire pour maintenir la dynamique de l'image (différence entre le niveau du pixel le plus élevé et le plus faible) ainsi que sa linéarité.

L'application principale des produits de convolution est la création des filtres passe haut et passe bas [12].

Remarque

Pour toutes les opérations de convolution, la convolution sera destinée à créer une nouvelle image constituée des pixels reflétant une "mesure". Pour que l'image résultante une "mesure". Pour que l'image résultante soit valide, il faudra normaliser les mesures calculées afin qu'elles soient dans l'intervalle [0,255].

L'étude du masque de convolution permet de connaître la nature et les effets du filtre mis en œuvre. Par exemple :

$$\begin{array}{ccc}
 \textcircled{1} \quad \left\{ \begin{array}{l} h = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} & h = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} & h = \begin{bmatrix} 1 & -3 & 1 \\ -3 & 9 & -3 \\ 1 & -3 & 1 \end{bmatrix} \\
 \text{Moyenueur} & \text{Binomial-gaussien} & \text{Rehausseur de contraste}
 \end{array} \right. \\
 \\
 \textcircled{2} \quad \left\{ \begin{array}{l} h = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} & h = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} & h = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \\
 \text{Sobel horizontal} & \text{Gradient oblique} & \text{Laplacien}
 \end{array} \right. \\
 \text{Filtres } \textcircled{1} : & H(0,0) = \sum_i \sum_j h(i, j) = 1 \\
 \text{Filtres } \textcircled{2} : & H(0,0) = \sum_i \sum_j h(i, j) = 1
 \end{array}$$

Figure II.3: Exemple de masques de convolution de taille (3×3)

II.3 Filtrage spatial

C'est une technique très répandue pour améliorer les images. On utilise un filtre sous forme d'une matrice carré (3×3 ou 5×5 le plus souvent).

On centre le filtre sur chaque pixel, et on effectue la somme des produits des intensités des pixels par les valeurs des filtres [13]. Le filtrage spatial est divisé en 2 catégories :

a- Filtres passe haut

Un filtre passe haut, favorise les hautes fréquences spatiales, comme les détails, et de ce fait, il améliore le contraste, un filtre passe haut est caractérisé par un noyau comportant des valeurs négatives autour du pixel central.

Le filtrage passe haut met en évidence les variations de luminance qui caractérisent traditionnellement les contours des objets ou de la texture d'une image. Un filtre passe haut avec un gain unitaire permet par exemple de rehausser le contraste de l'image originale.

b- Filtres passe bas

Les filtres passe bas agissent en sens inverse des filtres passe haut et les résultats est un adoucissement des détails, ainsi qu'une réduction du bruit granuleux.

Le filtrage passe-bas qui atténue les variations de luminance. Il a donc pour effet de lisser le contenu de l'image et de limiter les variations brusques d'intensité.

On utilise typiquement afin de réduire les effets du bruit, et de supprimer le contenu haute fréquence (détails de l'image) avant un sous-échantillonnage (afin de limiter le risque d'apparition de fréquences parasites).

Remarque

On parle de sous-échantillonnage lorsque l'image est déjà discrétisée et qu'on diminue le nombre d'échantillons.

II.4 Bruits

Le bruit est un phénomène parasite généralement aléatoire provenant de phénomène divers : bruit thermique, transmission, éclairage [16]. Il existe 2 principaux types de bruits :

a- Bruit linéaire

Chaque pixel de l'image est modifié par l'ajout d'une valeur aléatoire, et on parle aussi un bruit uniforme.

b- Bruit non linéaire

Quelques pixels de l'image, prennent des valeurs complètement aléatoire, et on appelle aussi un bruit impulsif [10].

Il existe des lois de probabilité qui permettent de créer le bruit dans une image.

Remarque :

Le bruit de type gaussien est plus difficile à supprimer et le bruit généré par les appareils d'acquisition se rapproche généralement du bruit gaussien, donc pour cela il est nécessaire de concevoir des algorithmes de suppression de bruits, efficaces [15].

II.5 Filtres linéaires

La classe des filtres linéaires est utilisée couramment en traitement du signal. Un opérateur f de traitement d'image est dit linéaire si : $f(a.I+b.J)=f(a.I)+f(b.J)$. Autrement dit, filtrer la somme arithmétique de deux images revient au même que de filtrer les deux images séparément, puis effectuer la somme arithmétique des résultats [7].

Mais un filtre linéaire donné sera le plus souvent caractérisé par son masque, c'est-à-dire la matrice $[h(i, j)]$ et la relation entrée-sortie étant donnée par :

$$S[i, j] = \text{Somme}_{u,v} (E[i, j] \times h[i-u, j-v])$$

Pour u, v variant de moins l'infini à plus l'infini, E la matrice de l'image et h le masque du filtre.

Dans le cas de filtrage linéaire, on considère du bruit linéaire ou uniforme qui peut être un inconvénient [9].

II.5.1 Lissage

Lisser une image consiste à remplacer la valeur de chaque pixel (x, y) par la moyenne des valeurs de ses pixels voisins. Le résultat est une image de même nature que l'image d'origine (couleur ou niveau de gris) [11].

But de ces filtres rendre plus homogène les niveaux de gris et supprimer le bruit.

Exemples : Filtre gaussien, filtre moyenneur.

a) Filtre Moyenneur

C'est un cas particulier de filtre de convolution passe-bas, qui remplace chaque pixel par la moyenne des valeurs des pixels voisins et du pixel central. La taille du masque dépend de l'intensité du bruit et de la taille des détails significatifs de l'image traitée.

Son masque pour un filtre 3×3 est : $\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$.

Les effets du filtre moyenneur, varient avec la taille du masque plus gros comme un 5×5 ou plus grand encore plus le bruit sera éliminé, mais en contrepartie peut créer une apparence artificielle, des détails fins seront eux-aussi effacés et les contours.

Le filtre gaussien présenté dans le paragraphe suivant est un exemple de filtre linéaire dont les coefficients, choisis avec plus de soins, permettent un traitement moins grossier de l'image.

b) Filtre Gaussien

Les coefficients du masque pour un filtre 3×3 sont : $\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$,

Les coefficients sont calculés en utilisant des pondérations gaussiennes. Des itérations successives permettent d'obtenir le gaussien 5×5 (2 itérations) et le gaussien 7×7 (3 itérations) [9].

Ce filtre très populaire utilise la loi de probabilité de Gauss. Soient $U(x)$ le niveau de gris en un point x de l'image à traiter et G_σ la gaussienne d'écart type σ donnée par la formule suivant :

$$G_\sigma = \frac{1}{\sqrt{2\pi}} \text{EXP} \left(-\frac{|x||x|}{2\sigma\sigma} \right) \quad (\text{II.1})$$

Le filtrage gaussien de l'image résulte de la convolution de cette fonction avec des gaussiennes en chaque point de l'image :

$$U(x,\sigma)=(G_\sigma \times U_0)(x)=\int \sigma(x-y)U_0(y)dy \quad (\text{II.2})$$

Dans ce cas aussi, l'effet du filtre augmente avec la taille de son masque. Les contours et les détails fins sont cependant mieux conservés qu'avec le moyenneur.

En effet, en utilisant une pondération gaussienne, le filtre gaussien ou smooth prend mieux en compte les corrélations entre pixels, notamment pour une texture d'image (la fonction de corrélation des niveaux de gris pour une texture est fréquemment modélisé par une gaussienne).

Le filtre gaussien est un bon exemple pour les performances qu'on peut obtenir avec un filtre linéaire à réponse impulsionnelle finie. Le gros avantage de ces filtres, c'est leur facilité de conception et d'implémentation, mais ils ne peuvent être utilisés pour des travaux

trop fins (la détérioration des contours qu'ils induisent par exemple, empêchera une segmentation fine des images).

Ces limitations ont donc conduit à la conception de filtres non-linéaires présentés dans la partie suivante [9].

II.5.2 Amélioration du contraste

Amélioration du contraste : ce traitement ne s'applique qu'aux images en niveau de gris. On considère qu'une image est bien contrastée si les luminosités des pixels de l'image sont au mieux réparties sur l'axe [0 ; 255]. La façon la plus directe consiste à normaliser les luminosités des pixels à partir des maximums et minimums des luminosités dans l'image originale.

Soit m la luminosité minimale et M la luminosité maximal dans l'image originale. Il s'agit d'effectuer une transformation affine de chaque luminosité pour ramener la dynamique de $[m ; M]$ à $[0 ; 255]$. [13]

Et le but du contraste aussi est d'évaluer les modifications apportées à l'image.

Son masque est :

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

II.5.3 Détection de contours

La détection de contours, consiste à exhiber les points d'une image en niveaux de gris qui sont susceptibles de se situer à la frontière entre deux régions se traduit dans l'image par une importante variation de luminosité. [11]

Le but de ces filtres est d'extraire les contours dans une image, Exemple : filtre contours verticaux, filtre contours horizontaux, filtre laplacien [10].

a) Filtres contours verticaux

Ce filtre de détection des contours verticaux, utilise la symétrie verticale. On effectue la symétrie des pixels par rapport à l'axe vertical qui partage l'image en deux parties égales

[18], son masque est :

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

b) Filtres contours horizontaux

Ce filtre de détection des contours horizontaux, est le contraire du filtre contours verticaux. Il utilise la symétrie horizontale. On effectue la symétrie des pixels par rapport à l'axe horizontal qui partage l'image en deux parties égales [16].

Son masque est :
$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Remarque

Les deux filtres de contours verticaux et de contours horizontaux peuvent être utilisés pour créer un autre filtre. C'est le filtre de Prewitt, qui utilise par exemple les deux masques précédents 3*3, l'un pour l'axe horizontal (X) et l'autre pour l'axe vertical (Y). Chacun des masques est en fait un filtre gradient, qui est tous les deux combinés pour créer l'image finale. Il existe également le filtre Sobel, Freeman et Kirsch.

c) Filtre Laplacien

Le filtre Laplacien, est un filtre de convolution particulier, utilisé pour mettre en valeur les détails qui ont une variation rapide de luminosité. Le Laplacien est donc idéal pour rendre visible les contours des objets, d'où son utilisation dans la reconnaissance de formes dans des applications militaires d'abord, puis civiles ensuite. Il existe trois noyaux typiques de taille

3×3 qui peuvent remplir ce rôle :
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}; \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}; \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}.$$

Le Laplacien est un bon détecteur de contours, mais sensible au bruit. Il est donc nécessaire d'effectuer un lissage de l'image avant de calculer l'image de Laplacien [11].

II.6 Filtres non linéaires

Ces opérateurs ont été développés pour pallier aux insuffisances des Filtres linéaires et principalement la mauvaise conservation des contours. Ils ont le défaut d'infliger des déformations irréversibles à l'image résultante. La théorie des filtres non-linéaires est que chacun fondé sur des bases mathématiques ou empiriques différentes [9], ils permettent donc de supprimer totalement le bruit dans une image, exemple : le filtre Médian [10].

Les filtres linéaires, ne sont pas des algorithmes prédéfinis puisque ils ne reposent pas sur une théorie mais ils sont plus variés. Souvent, ils sont plus coûteux en temps de calculs [10].

Un de leur avantage, est de permettre de s'adapter à la nature du point considéré (pixel utile ou pixel de bruit) [15].

Deux aspects du lissage sont concernés par le filtrage non linéaire :

- **Le bruit impulsionnel** : les filtres linéaires éliminent mal les valeurs aberrantes.

- **L'intégrité des frontières** : on souhaiterait éliminer le bruit sans le rendre flou les frontières des objets [8].

II.6.1 Lissage

Il existe un filtre non linéaire qui rend l'image moins flou que le filtre gaussien ou moyennneur. C'est le filtre Médian et il est plus adéquat pour le bruit impulsif [10].

a) Filtre Médian

Le filtre Médian est utilisé pour atténuer les pixels isolés, d'une valeur très différente de leur entourage. Le filtre Médian n'est pas à proprement parler un produit de convolution, mais sa mise en œuvre sur l'image est assez similaire puisqu'un masque est appliqué sur l'image et collecte les valeurs des pixels. Sur l'exemple ci-dessous le masque est appliqué sur l'image et collecte les valeurs des pixels. Sur l'exemple ci-dessous le masque est un $3 \times 3 = 9$ éléments. Les neuf éléments extraits de l'image sont ensuite triés dans l'ordre croissant.

La valeur médiane d'une série est par définition celle qui sépare l'échantillon en deux parties de population égale ici on voit que c'est la valeur (41) facilement repérable à cause du tri :

L'algorithme va donc remplacer la valeur originale par la valeur médiane qui vaut 41 [10].

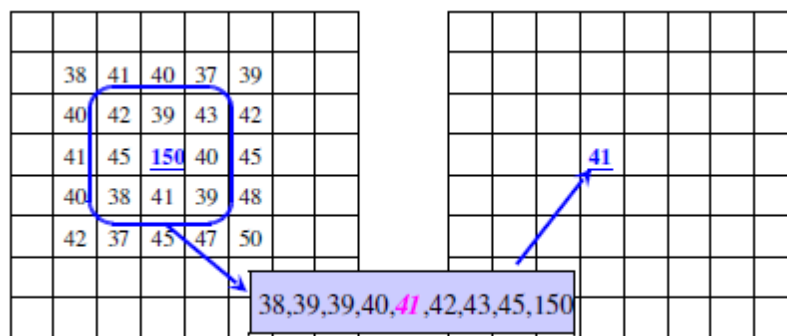


Figure II.4: Exemple de filtre Médian

Remarque

Le filtre Médian garde la netteté de l'image pour les éléments de dimensions importantes (par rapport au masque du filtre), mais élimine les détails fins de manière irrémédiable [9].

b) Filtre Max

Le filtre Max, est utile pour éliminer le "poivre". Sur l'exemple ci-dessous le masque est un $3 \times 3 = 9$ éléments. les neufs éléments extraits de l'image sont ensuite triés dans l'ordre croissant la valeur max d'une série est définie (facilement repérable à cause du tri).

L'algorithme va donc remplacer la valeur originale par la valeur max. la formule est :

$R = \max \{Z_k \mid k = 1, 2, \dots, 9\}$, pour trouver les points clairs [17].

c) Filtre Min

Le filtre Min est le contraire du filtre Max, il est utile pour éliminer le "sel". Il fonctionne, comme le filtre Max sauf que dans son cas, le filtre Min prend la valeur minimale et dont la formule est $R = \min \{Z_k \mid k = 1, 2, \dots, 9\}$, pour trouver les points clairs. [17]

II.6.2 Les filtre de détection des contours

L'objectif de cette famille de filtre n'est plus de lisser l'image mais au contraire de mettre en valeur ses limites et ses discontinuités. Une méthode simple consiste à soustraire une image filtrée par un filtre médian à l'image originale.

La détection de contour revient à détecter les points à forte variation dans l'image. Donc si on parle d'une bonne détection de contour, celle-ci doit être capable de différencier entre le bruit et la texture des objets (tous deux ont une forte variation).

Il existe plusieurs méthodes pour la détection du contour. On peut citer trois classes selon leur approche :

Méthodes différentielles (Gradient, Laplacien) ; Méthodes par templates (Roberts, Prewitt, Sobel, Kirsh) ; Méthodes par optimisation des contours, de bruits, (Marr, Canny, Deriche).

a) Filtre de Sobel

Le principe de ce filtre est que l'opérateur calcule le gradient de l'intensité de chaque pixel. Ceci indique la direction de la plus forte variation du clair au sombre, ainsi que le taux de changement dans cette direction. On connaît alors les points de changement soudain de luminosité, correspondant probablement à des bords, ainsi que l'orientation de ces bords.

L'opérateur utilise des matrices de convolution. La matrice (généralement de taille 3×3) subit une convolution avec l'image pour calculer des approximations des dérivées horizontale et verticale [18].

Ce filtre fait appel aux 2 masques défini ci-dessous et que on appel opérateurs de Sobel.

$$C1 = \frac{\partial I}{\partial x} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}; C2 = \frac{\partial I}{\partial y} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (\text{II.3})$$

Ces 2 masques C1 et C2 sont successivement appliqués à tous les points de l'image et le plus grand des 2 produits de convolution est retenu.

Le filtre de Sobel combine à la fois le lissage par le filtre monodimensionnel [1, 2, 1] et la dérivée selon une direction perpendiculaire au lissage, obtenue par le filtre [1, 0, -1]. On peut voir le résultat comme la moyenne des dérivées des moyennes. Détecteur de Roberts, l'image des contours est obtenue en calculant, pour chaque pixel le module du gradient. On peut appliquer ce détecteur sur des images couleurs en le même algorithme sur les différentes composantes RGB prises séparément.

Propriétés :

- Détection de renforcement des contours.
- Détection des régions uniforme ou homogène.



Figure II.5: Exemple du filtre Sobel

b) Filtre de Prewitt

La détection du contour par la méthode de Prewitt utilise les masques de convolution suivants :

$$C1 = \frac{\partial I}{\partial x} = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}; C2 = \frac{\partial I}{\partial y} = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad (\text{II.4})$$

La matrice qui correspond au filtrage horizontal, faisant ressortir essentiellement les contours verticaux, selon l'opérateur de Prewitt, s'écrit $h_x = [-1 \ 0 \ 1]$ tandis que la matrice verticale h_y est sa transposée. Les deux convolutions avec le tableau de valeurs initiales créent

deux tableaux G_x et G_y à l'origine du tableau G sur lequel on peut localiser les maximums [18].



Figure II.6 : Exemple du filtre Prewitt

b) Filtre de Robert

Les filtres de Roberts sont une approche discrète de la dérivée de pas 1 d'une fonction: le gradient de cette fonction. Le détecteur de Roberts recherche les dérivées selon les directions diagonales. Il est décomposé en deux masques de convolution pour chacune des composantes du gradient

$$\frac{\partial I}{\partial x} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}; \frac{\partial I}{\partial y} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (\text{II.5})$$

Le calcul du module du gradient permet alors d'obtenir le tracé de contour :

$$\sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \quad (\text{II.6})$$

Ce qui peut également être réalisé par :

$$\max\left(\left(\frac{\partial I}{\partial x}\right), \left(\frac{\partial I}{\partial y}\right)\right) \quad (\text{II.7})$$

Le détecteur de Roberts est un des plus vieux détecteurs de contours en traitement d'image numériques, mais il est à la fois le plus simple. Ce détecteur est bien moins utilisé que ceux de Sobel et Prewitt en raison de ses fonctionnalités limitées. En effet il bénéficie des mêmes inconvénients que notre exemple basique. Cependant il est fréquemment utilisé dans des implémentations matérielles où la simplicité et la vitesse sont des facteurs dominants [18]. Or le bruit peut aussi être une brusque variation locale des niveaux de gris : ces filtres sont

donc très sensibles au bruit car ils accentuent, par dérivation, le bruit présent dans l'image. De plus, ces filtres donneront un contour épais si celui-ci est un contour de type "rampe".



Figure II.7 : Exemple du filtre Robert

II.7 Conclusion

Dans ce chapitre, nous avons étudié les différents filtres de traitement d'images, dans la partie théorique l'extrême importance de détection des contours pour l'extraction des textes dans une image, pour cela les différentes approches ou opérateurs à savoir ; Sobel, Prewitt et Robert sont détaillées. Nous avons commencé par des brefs théoriques sur les filtres puis lissage et les méthodes de détection des contours, Dans le chapitre suivant, nous proposons un algorithme efficace pour la détection de défilement du texte dans une trame de vidéo.

CHAPITRE III

Algorithme Efficace pour la Détection de Défilement du Texte
sur une séquence Vidéo

III.1 Introduction

La plupart des documents précédents dans ce domaine ont été présentés des algorithmes pour la détection du texte dans les images fixes. En revanche, cette étude a examiné le problème de la détection de défilement de texte dans les vidéos. Texte défilant se déplace dans différents cadres, ce qui permet l'utilisation de corrélation temporelle d'identifier les caractéristiques différentielles entre inter-cadres. Malheureusement, la vitesse de déplacement ne correspond pas nécessairement aux chaînes de télévision. Ainsi, cette étude propose sélection du cadre d'adaptation pour localiser des informations liées au texte supplémentaire. Nous avons également étudié des méthodes de pré-filtrage de texte pour l'amélioration de bord et d'augmenter la précision de la détection et la suppression du bruit adaptative qui a été utilisée pour différencier les pixels de bruit à partir de pixels de texte.

III.2 Algorithme proposée

La Figure III.1 présente un schéma synoptique de l'algorithme proposé. Afin de réduire les coûts de calcul, l'algorithme proposé est utilisé pour convertir les images en couleur à échelle de gris, de telle sorte que seul le signal Y est utilisé pour le traitement. L'objectif du prétraitement est d'améliorer l'image lisse, réduisant ainsi le niveau de bruit afin de faciliter la précision de détection des informations de texte.

Des différences temporelles sont souvent trouvés entre les trames courante et précédente dans la région du texte défilant. Ainsi, cette étude a adopté une approche différentielle temporelle pour identifier les caractéristiques de base du texte de défilement. Parce que l'information liée au texte de défilement est limitée à des endroits le long de l'un des quatre frontières de chaque trame, les pixels dans la région centrale peuvent être rejetées d'emblée. Un degré élevé de contraste doit être maintenu entre les pixels de fond et texte pixels pour assurer que le texte est clairement visible. Contraste élevé conduit inévitablement à une forte variance, qui peut être utilisé pour calculer la variance dans les quatre frontières afin de détecter des informations de texte. Défilement de texte ne peut être trouvée dans des zones de forte variance; par conséquent, les zones frontalières avec faible variance peuvent être rejetées d'emblée. Il est possible que certains pixels de texte puissent être perdus après le traitement dans l'étape précédente. Par conséquent, cette étude a adopté une opération de dilatation pour récupérer texte pixels. Enfin, une grande ligne peut être tirée pour définir la zone rectangulaire dans laquelle le défilement du texte est plus susceptible d'être trouvé.

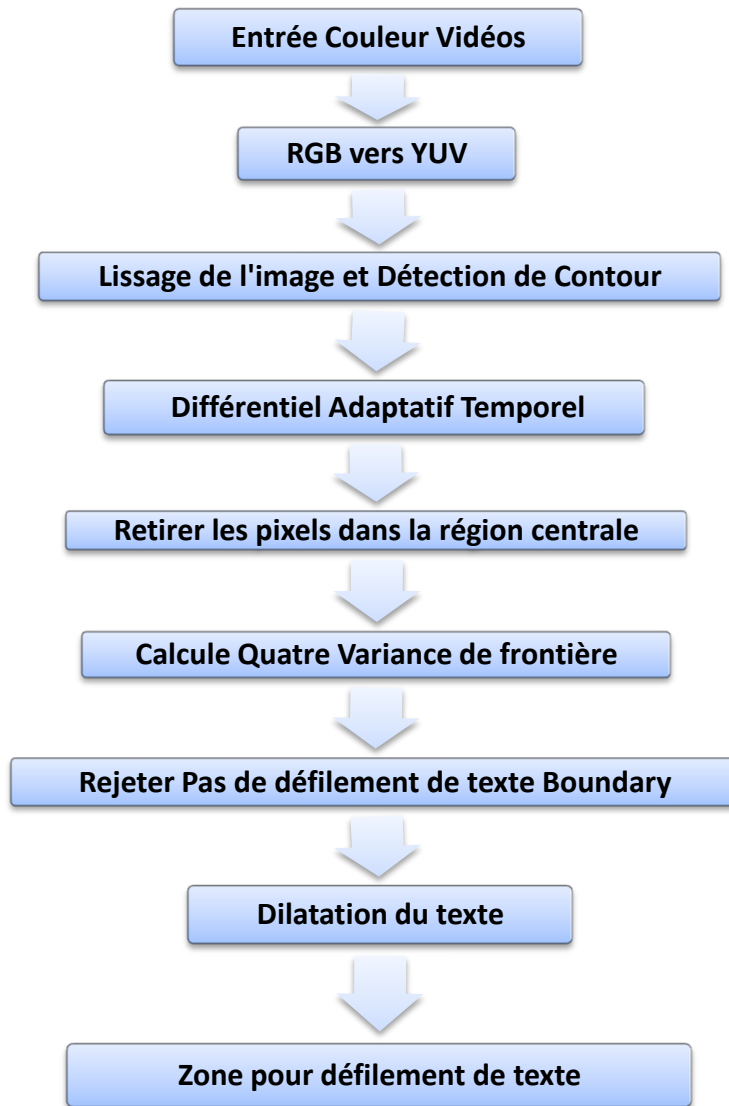


Figure III.1 : Le schéma synoptique de l'algorithme proposé pour défilement de texte.

III.2.1 Lissage d'image et détection de contour ;

L'opération de lissage dans le présent document implique l'application d'un filtre passe-bas pour supprimer les niveaux de bruit. Typiquement, cela implique l'utilisation d'un filtre gaussien, le filtre moyen, ou filtre médian [19] [20]. Gaussiennes filtres sont utilisés pour conserver les informations de l'image originale et de réduire tous les types de bruit. Les filtres moyens peuvent également être utilisés pour réduire le bruit vidéo; Toutefois, le processus de la moyenne des pixels d'image peut détruire des informations de texte important. Filtres médians peut efficacement

traiter le bruit impulsif; cependant, ils sont mal adaptés à ce type de demandes. Cette étude a adopté un filtre gaussien pour la réduction du bruit dans le but d'améliorer la détection de texte.

Afin d'améliorer l'information de texte, la détection des contours [20] est utilisée. Les Figure III.2 - III.3- III.4 présentes les résultats de détection de contours en utilisant une variété de méthodes. Le détecteur Roberts est la méthode la plus efficace pour rejeter bords de non-texte et de texte en préservant les bords, comme le montre la Figure III.2. Bien que certains procédés soient capables de détecter des informations de bord, les résultats ne sont pas nécessairement idéals pour la détection de texte, comme le montre les Figure III.2 - III.3. Ainsi, cette étude a adopté l'opérateur Roberts pour la détection de texte.



Figure III.2 : illustration de détecteur de Robert.



Figure III.3 : illustration de détecteur de Sobel.



Figure III.4 : illustration de détecteur de Prewitt.

III.2.2 Différentiel Temporel Adaptative

Texte défilant se déplace rapidement au niveau des limites du cadre. Ainsi, les différences temporelles tendent à être relativement grandes lorsque le texte défile sur des trames continues. Différences temporelles peuvent être calculées à l'aide

$$Diff_{jk} = |f_{jk}^t - f_{jk}^{t-N}| \quad (III.1)$$

Pour permettre le calcul pixel par pixel entre la trame courante f_{jk}^t et la Nième trame précédente f_{jk}^{t-N} . La valeur du différentiel $Diff_{jk}$ peut être utilisée pour déterminer si le défilement de texte apparaît dans une séquence. Les quatre limites de trame (haute, basse, gauche et droite) sont vérifiées indépendamment en fonction de la valeur différentielle moyenne. L'écart moyen ($Diff$) est calculé par

$$Diff = \frac{\sum_{j=0}^N \sum_{k=0}^M Diff_{JK}}{N \times M} \quad (III.2)$$

Où $M \times N$ est la taille de bloc de la frontière. Règles de radiodiffusion définissent la zone de défilement de texte que 25% de l'image. Ainsi, cette étude vérifie chaque frontière à l'aide de 30% de la trame, comme représenté sur la Figure III.5, ce qui élimine ainsi les pixels dans la région centrale pour permettre une détection rapide. Les limites sont vérifiées pour faire défiler le texte en se référant à la

valeur Diff. Si la valeur limite Diff à tout dépasse un seuil donné, les pixels de bord sont préservés et la région de défilement de texte est associée à cette limite. Dans le cas contraire, les pixels sont éliminés en tant que candidats pour la détection d'un texte défilant.

	30%	
30%		30%
	30%	

Figure III.5 : Le texte défilant vérifié avec 30% de frontière.

La vitesse de défilement du texte sur le cadre n'est pas constante. Moins défilement rapide présente de grandes différences temporelles entre des trames adjacentes. Texte lent défilement peut entraîner les mêmes informations de texte apparaissant sur plusieurs trames, entraînant de légères différences temporelles entre les images en continu, ce qui peut entraver la détection de texte. Pour améliorer l'efficacité de détection, un certain nombre de cadres peut être ignorée pour fournir un plus grand déplacement entre inter-cadres. Cette étude a introduit la méthode de calcul différentiel adaptatif temporel représenté sur la Figure III.6. Parce que la vitesse du texte mobile varie entre les chaînes de télévision, nous avons commencé avec $N = 1$ comme valeur initiale pour les nouveaux canaux. La différence temporelle initiale entre trames adjacentes est déterminée par l'équation. (1). Une différence accumulée

(Diff) dépassant le seuil est une indication de déplacement rapide texte, tel que le châssis adjacent peut fournir un déplacement suffisant pour la détection d'un texte défilant. Toutefois, si le Diff est inférieure au seuil, N est progressivement augmentée en utilisant un traitement adaptatif, dans lequel un certain nombre de cadres sont ignorés afin d'élargir la différence inter-trame. Cette procédure est répétée jusqu'à ce que N est constaté de façon adaptative lorsque $\text{Diff} > \text{seuil}$. Ce résultat indique que la Nième trame peut être utilisé comme référence pour la détection d'un texte défilant sur la chaîne de télévision en cours. Si le nombre d'images sautées atteint le paramètre maximal (N_{max}) et la Diff reste en dessous du seuil, cela indique que pas moins de défilement se trouve le long de cette limite.

Le nombre de trames sautées de référence N dépend de la vitesse de défilement du texte. Une vitesse plus faible que nécessite plusieurs trames sont ignorées pour fournir le déplacement nécessaire entre la trame courante et la trame de référence. Ainsi, la valeur de N est inversement proportionnelle à la vitesse du texte défilant. En d'autres termes, N correspond au temps nécessaire pour que le texte de défilement à se déplacer. Le nombre de trames a sauté peut être estimée par

$$N = \text{taux de trame} \times \text{temps de marche} \times k \quad (\text{III.3})$$

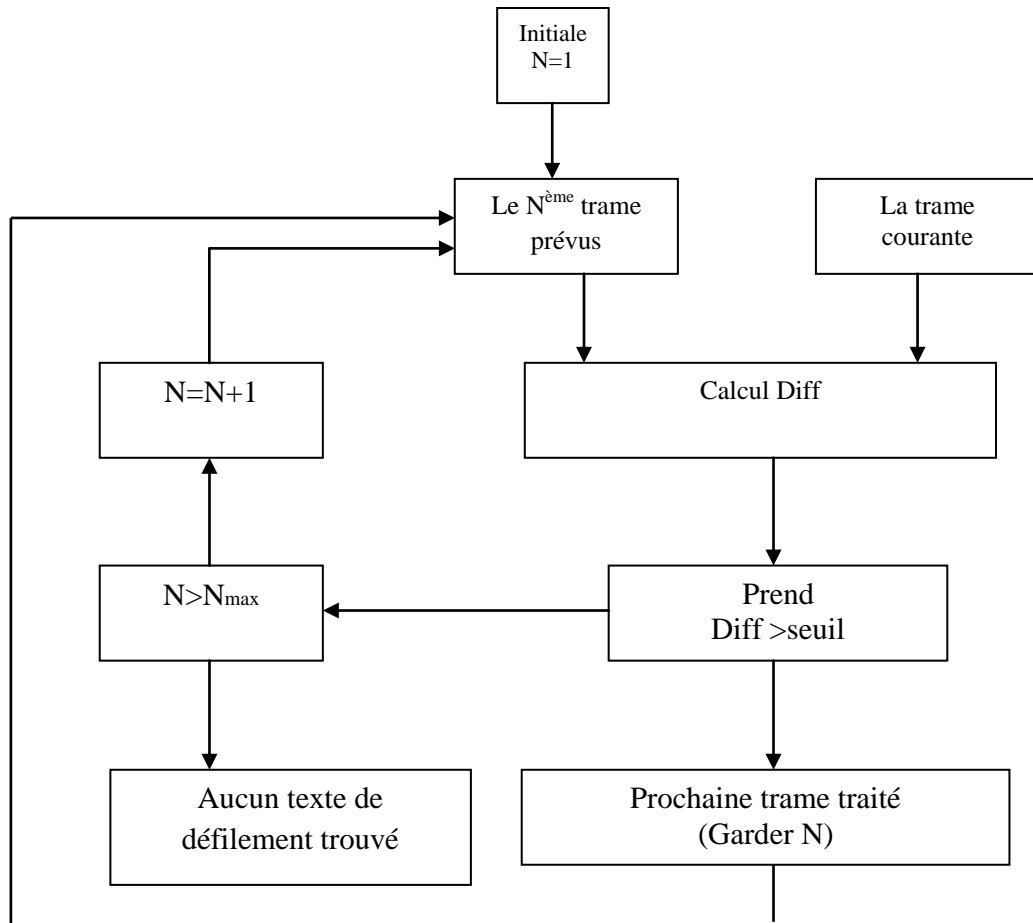


Figure III.6 : calcul de Différentiel Temporel Adaptative.

Où le temps de marche (Moving_Time) désigne le temps nécessaire pour le défilement de texte à déplacer d'une distance donnée. Frame_Rate indique le nombre d'échantillons obtenus à partir du signal de télévision par seconde, et k est constant. Lorsque la vitesse de défilement de texte est plus élevée, le temps de déplacement nécessaire pour le mouvement du texte sera plus court; par conséquent, la valeur de N va être plus petite. Lorsque le taux de trame est plus élevé, le même texte apparaît dans plusieurs cadres. Par conséquent, plusieurs trames doivent être sautées pour atteindre la différence suffisante dans le mouvement de défilement de texte; Par conséquent, N sera plus élevé.

III.2.3 Processus de dilation et de localisation du texte

Localisation du texte exige que les coordonnées du rectangle le défilement moins être définis. Après le calcul de l'écart temporel, des pixels de bruit sans rapport avec texte défilant peuvent présenter une grande valeur différentielle entre inter-cadres. Ainsi, ces pixels doivent être éliminés pour éviter les dégrader la précision de détection. Généralement, le texte pixels a une moyenne locale de haut niveau lors de l'utilisation du texte blanc sur un fond noir parce que les pixels de texte sont regroupés comme une région. Cependant, le bruit pixels est distribué au hasard et donc discrètement sur une image donnée. Le moyen local de pixels de bruit doit être inférieur à celle des pixels de texte, ce qui permet leur différenciation. Tout d'abord, la moyenne locale est calculée à l'intérieur d'une fenêtre $K \times L$ en tant que

$$LM_{ij} = \frac{\sum_{i=-K}^K \sum_{j=-L}^L Diff_{i+K,j+L}}{K \times L} \quad (III.4)$$

Où LM_{ij} désigne la moyenne locale pour le pixel f_{ij} . La valeur différentielle temporelle calculée en utilisant (III.1) peut être tronqué par

$$If Diff_{(j,k)} < Noise_Threshold \quad then T_t^{(i,j)} = 0 \quad (III.5)$$

$$Else T_t^{(i,j)} = 255$$

De telle sorte qu'un pixel utilise une résolution de 8 bits, où $T_t^{(i,j)}$ est une image binaire tronquée désignant des informations textuelles. Texte binaire est formé lorsque la superposition d'un objet de texte blanc sur un fond noir. Pour réduire le bruit en arrière-plan, le seuil de bruit (*Noise_Threshold*) dans (III.5) peut être calculé de manière adaptative selon la moyenne locale à l'aide

$$Noise_Threshold = C_{NT} - 2 \times LM_{ij} \quad (III.6)$$

Où la valeur du C_{NT} est constante. Afin d'améliorer le facteur moyen local, la valeur LM_{ij} est amplifiée par un facteur de 2. Une faible valeur de la moyenne locale indique que le pixel courant a une forte probabilité d'être un pixel de bruit. Ainsi, le bruit seuil est relevé permettant la plupart du bruit pour être tronquée à zéros. Après le calcul (III.5), la majeure partie du bruit est éliminé; Toutefois, une partie du bruit de haut niveau reste. Heureusement, la distribution aléatoire de bruit résiduel permet l'application du traitement de l'isolement pour sa suppression à l'aide

```

{
  for w1= -k to k
    for w2= -k to k

      if ( $S_{(i+w1,j+w2)} = 255$ )      Isolation-Counter++;
    }

  If Isolation-Counter > Isolation-Threshold       $S_{(i,j)} = 255$ 
  Else  $S_{(i,j)} = 0$ 

```

(III.7)

Où k a la valeur 1 pour l'élimination complète du bruit dans les expériences suivantes. Lorsque plus d'un tiers (1/3) des pixels dans une fenêtre de texte sont des points, pixels en cours de traitement peuvent être traités comme un pixel de texte; autrement, elle est considérée comme un bruit. Ainsi, un isolement de trois (3) seuil est utilisé pour le traitement d'une fenêtre de 3x3. Si les pixels transformés sont situés loin du groupe de texte, ils peuvent être considérés comme du bruit et donc éliminés par traitement via Eq. (III.7).

Opérateurs différentiels précédents et la suppression du bruit peut entraîner le rejet de quelques pixels de texte. Ainsi, cette étude a utilisé une opération de dilatation pour élargir la plage de texte afin de récupérer pixels perdus, comme le montre la Figure III.7.

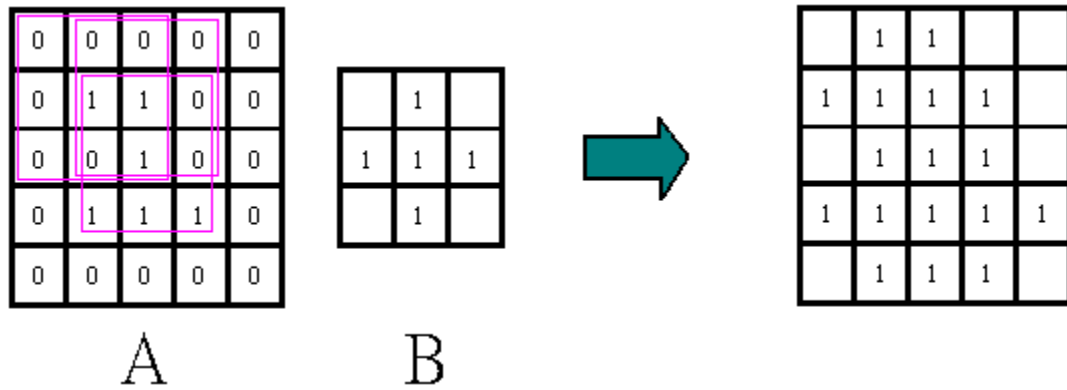


Figure III.7 opération de dilation.

Les coordonnées du rectangle de texte défilant est déterminé en fonction de la valeur de coordonnées maximale dans chaque direction. Les coordonnées maximales peuvent être facilement trouvées par balayage. Par exemple, pour la détection de texte défilant dans la direction verticale, balayage est effectué de la ligne supérieure à la ligne de fond rangée par rangée, et les premiers et derniers pixels sont désignés comme les coordonnées supérieure et inférieure d'un rectangle. Pour la détection de texte défilant dans la direction horizontale, la numérisation est effectuée colonne par colonne, de telle sorte que la position des premier et dernier pixels de texte sont désignés comme étant la droite et les coordonnées d'un rectangle à gauche. Une fois que les coordonnées du rectangle sont identifiées, les limites du texte défilant sont marquées au moyen de ligne gras pour la localisation.

III.3 Conclusion

Dans ce chapitre, nous avons présenté un algorithme efficace pour la détection de défilement du texte dans une séquence vidéo, cet algorithme est constitué par trois étapes nécessaire et important, la première étape est le lissage et détection de contours d'image, la deuxième nous avons calculés la différence temporel adaptive et à la dernière nous avons appliqués l'opération de dilation et de localisation de texte, dans le chapitre suivant, nous allons présenter les résultats de simulations des algorithmes proposés.

CHAPITRE IV

Simulation & Résultats

IV.1 Introduction

Dans ce chapitre nous présentons notre simulation des résultats obtenus en appliquant l'algorithme proposé sur les différents échantillons des séquences vidéos afin d'extraire et de détecter le défilement du texte sur la vidéo.

IV.2 Simulation

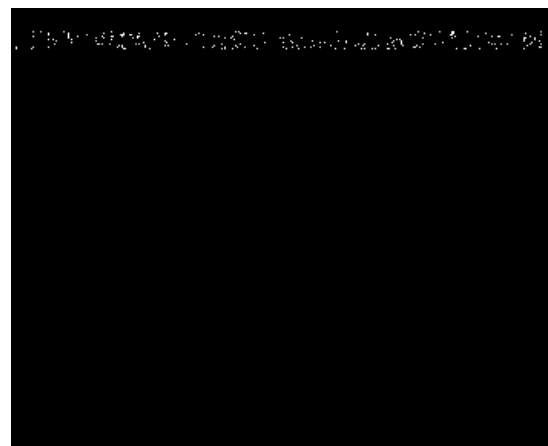
Dans la partie simulation, cette étude a adopté des échantillons de vidéo de télévision numérique (TVN) programmes. La taille de l'image est de 352x240 à 30 images par seconde. Les échantillons ont été prélevés en continu et stockés sur le disque dur dans le format MPEG-2. Chaque séquence a été stockée pendant environ une minute, après quoi un décodeur MPEG-2 a été utilisé pour reconstruire les premiers pixels RGB. Pour la détection de texte défilant, RGB des pixels ont été converties au format YUV, de telle sorte que seul le signal Y a été généré, conformément à l'algorithme proposé dans la Figure III.1. Pour activer adaptative temporelle informatique différentiel, nous fixons le seuil(Threshold) =15 et $N_{max}=10$ dans les expériences. Ainsi, lorsque la fréquence de trame est de 30 images / s, et le temps nécessaire pour que le texte de défilement à se déplacer d'une distance donnée est de 0,1 sec. Dans ces conditions, le même texte apparaît dans trois trames contiguës. Pour extraire le différentiel de déplacement du texte défilant, le nombre d'images sautées (N) doit être réglé à au moins 2. Toutefois, la portée de détection devient trop grande lors de l'utilisation d'un N supérieur. Le nombre de trames sautées augmente avec une augmentation de la vitesse de défilement du texte. En cas de mouvement très lent, lorsque $N = N_{max}$, faire défiler le texte ne peut pas être trouvé. Toutefois, lorsque la trame précédente a été marquée avec un rectangle de texte défilant à indiquer, l'image courante peut être marquée en utilisant le même rectangle.

Étape par étape les résultats du traitement sont présentés dans la Fig. 6 pour illustrer les performances de l'algorithme proposé.

1. Les Figure IV.1 (a) & (b) montrent la conversion de l'image originale de couleur à échelle de gris utilisant une matrice de transformation.
2. L'algorithme de détection Robert est ensuite utilisé pour détecter l'image en demi, comme le montre la Figure IV.1 (c).
3. Une image des bords avec une différence temporelle plus grande entre les pixels est présentée dans la Figure IV.1 (d).
4. Les côtés du texte de défilement sont sélectionnés par calcul différentiel moyenne en utilisant l'équation. (2), comme le montre la Figure IV.1 (e).
5. La Figure IV.1 (f) illustre les opérations de troncature et de suppression de la Figure IV.1 (e) utilisé pour améliorer l'information de texte.
6. Enfin, le texte résultant pixels peut être extrait à l'aide des résultats de dilatation en tant que masque pour l'image de gris d'origine, comme le montre la Figure IV.1 (g).
7. La Figure IV.1 (h) présente un rectangle sur l'image couleur originale indiquant la portée maximale des pixels associés avec le texte de défilement.



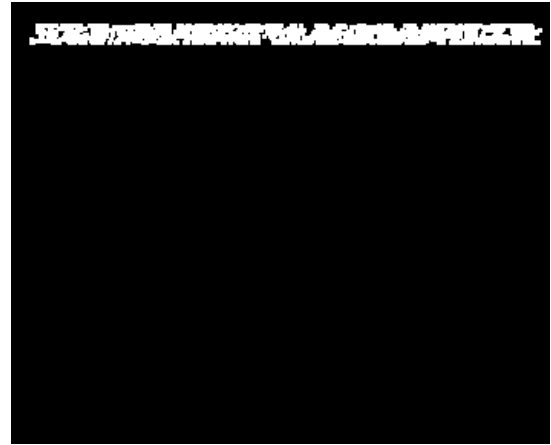
(a)



(e)



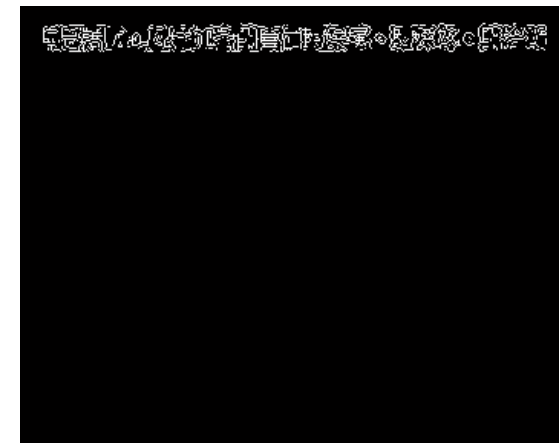
(b)



(f)



(c)



(g)



(d)



(h)

Figure IV.1 illustration du processus de l'algorithme étape par étape.

IV.3 Conclusion

L'algorithme proposé est capable d'identifier avec précision le défilement de texte à partir des vidéos grâce à l'application de la différenciation temporelle basée sur le traitement de trames multiples. Aucun cas de fausse détection ou la détection manquante ont été observées dans les expériences. Dans certains cas, avec les milieux complexes, la gamme de détection a eu tort. Cependant, l'algorithme proposé a été en mesure d'identifier la position de défilement du texte avec un degré élevé de précision.

CONCLUSION GENERALE

CONCLUSION GENERALE

Ce mémoire présente un algorithme rapide et efficace pour la détection de texte défilant dans des cadres vidéo TV. L'algorithme est capable de détecter le texte défilant et sauter encore de texte par l'adoption d'un traitement spatial et temporel. Pré-filtrage est utilisé pour améliorer l'information de texte et d'adaptation calcul différentiel temporel est utilisé pour identifier le défilement du texte. Le post-traitement est utilisé pour enlever pixels liées au bruit et à élargir la gamme de texte défilant. L'efficacité de l'algorithme a été vérifiée à l'aide d'une vaste base de données de vidéos obtenues à partir de programmes de télévision réels. Les résultats démontrent l'exactitude de l'algorithme proposé dans la différenciation de texte défilant, sans fausses détections ou détections manquées. L'algorithme proposé surperformé toutes les méthodes concurrentes dans des expériences réelles.

PERSPECTIVES

Dans l'avenir, l'algorithme proposé de détection de défilement du texte pourrait également être utilisé pour cacher un texte défilant sans compromettre l'information du fond de vidéo. Les pixels de défilement de texte peuvent aussi être cachés en utilisant des techniques d'interpolation. Enfin, le texte défilant pourrait être utilisé pour produire un indice pour le classement et l'archivage des vidéos.

BIBLIOGRAPHIE

- [1] R.E.-Cummings, P.Pouliquen, M. Anthony Lewis “A Vision Chip for Color Segmentation and Pattern Matching” EURASIP Journal on Applied Signal Processing, pp 703–712, 2003.
- [2] DEVAUX Frédéric ; « Filtrage d’images par Réseaux de Neurones » ; Edition Université Paris 8 Saint Denis, 1997, France.
- [3] DESACHY Jacky ; « Analyse d’images » ; Edition Université des Antilles et de la Guyane Pointe à Pitre, Janvier 2001, France.
- [4] Z. Hai-bo, Y. Kui, L. Jin-dong “A Fast and Robust Vision System for Autonomous Mobile Robots” Proc. Int. Conf. on Robotics, Intelligent Systems and Signal Processing, China, 2003.
- [5] D.Khadraoui “La Commande Référencée Vision Pour Le Guidage Automatique De Véhicules” Thèse doctorat, université de Renne-I, 1996.
- [6] <http://www.ariadne.ac.uk/issue8/electronic-formats/intro.html>, les différents formats d’images.
- [7] OGE MARQUES Florida « PRACTICAL IMAGE AND VIDEO PROCESSING USING MATLAB »; Edition Atlantic University 2011 by John Wiley & Sons.
- [8] Frédéric DROLET « Introduction au Traitement des images», Cours, Université Laval, Hiver 2006.
- [9] Eric Favier. L'analyse et le traitement des images. ENISE ; Edition Université de Lyon.
- [10] INB1 - T. Boudier1 Traitement d’images II Transformations géométriques Filtrages linéaires et non-linéaire.
- [11] Digital Image Processing: An Algorithmic Introduction using Java (Burger & Burge).
- [12] Astroate CCD Image processing, cours d'astrométrie et de compositage pour images CCD.
- [13] MICHEL BEAUDOUIN « Informatique Graphique II : Traitement d’image», Cours, Université de paris-sud, 2005.
- [14] Lingrand Diane « Cours de traitement d’image » ; université Nice sophia antipolis, janvier 2004.
- [15] Benoit Naegel « Traitement d’image », Cours Master 1 ISI, Université de strasbourg, 2008.
- [16] Dana.H.Ballard & Christopher.M.Brown Computer vision, prentice Hall, Engle wood Cliffs, New Jersey, 1982.
- [17] Jean-Luc Bloechle «TRAITEMENT DE L'IMAGE NUMÉRIQUE» UNIVERSITÉ DE FRIBOURG, SUISSE, Mai 2008.
- [18] Introduction au traitement d’images Détection de contours et segmentation, Yoann Sculo Semestre de Printemps 2009 ; Branche SIT.
- [19] C. L. Zitnick, Member, IEEE, and William T. Freeman “Automatic estimation and removal of noise from a single Image”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 30, No.2, pp. 299-314. Feb, 2008.
- [20] R. C. Gonzalez, R. E. Woods, *Digital Image Processing*, Addison Wesley, 1992.

Résume :

Ce travail présente un algorithme efficace pour la détection de défilement de texte sur des trames en continu. Le système proposé adopte à la fois spatiale et temporelle de calcul ainsi que de prétraitement de différencier les bruits, du texte et l'information arrière plan.

Défilement du texte est détecté en utilisant différenciation temporelle entre les inter-images, puis la région dans laquelle le texte défilant apparaît est identifiée comme un rectangle.

Mots clé : Texte défilant, suppression du bruit, la dilatation, la différenciation temporelle, détection de texte.

Abstract:

This paper presents an efficient algorithm for the detection of scrolling-text on continuous frames. The proposed scheme adopts both spatial and temporal computation as well as pre-processing to differentiate among noise, text, and background information.

Scrolling text is detected using temporal differentiation among inter-frames and then the region in which the scrolling text appears is identified as a rectangle.

Keywords: Scrolling text, noise removal, dilation, temporal differentiation, text detection.

ملخص:

يقدم هذا العمل خوارزمية فعالة للكشف عن تمرير النص على إطارات مستمرة. المخطط المقترح يعتمد على كلا من الحساب المكاني و الحساب الزمني , فضلا عن المعالجة المسبقة للتمييز بين الضوضاء , النص والمعلومات الأساسية. تمرير النص يتم الكشف عنه باستخدام الفارق الزمني بين الإطارات ثم المنطقة التي يتم التعرف على تمرير النص على شكل مستطيل.

الكلمات المفتاح: تمرير النص , إزالة الضوضاء , تمدد , الفارق الزمني , كشف النص.