

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'enseignement supérieur et de la recherche scientifique
UNIVERSITE AMMAR TELIDJI DE LAGHOUAT



INSTITUT DE GENIE INFORMATIQUE

PROJET DE FIN D'ETUDES

POUR L'OBTENTION DU DIPLOME
D'INGENIEUR D'ETAT EN INFORMATIQUE
OPTION : Systèmes Parallèles Et Distribués

THEME

Approche de sélection automatique d'un
détecteur de contours

Réalisé Par :
BOUSSEBCI SAFIA
FERHAT KHADIDJA

Encadré Par :
M^m: GUIBADJ FATNA

N° d'ordre: /2008-PFE/DGI

Remerciements

Nous aimerions remercier le Dieu en premier lieu qui nous a donné la force et la volonté d'achever ce projet.

Nous passons nos gratitude à Mme Guibadj Fatna pour nous avoir encadré et proposé ce sujet, pour avoir bien voulu nous aider à réaliser ce travail, et surtout avec ses orientations et ses conseils prodigués.

Nous tenons à remercier les membres du jury Mr Ahfir et Mr Biranne, pour l'honneur qu'ils nous ont fait en acceptant de présider et d'examiner notre mémoire.

On passe Nos gratitude à Mme Karouche, Mme chetieh, Mlle Abd el Hafidi, Mr Mokhatari, Mr Boukhalfa, Mr Djoudi, Mr Yagoubi, Mr Ouniten, Mr Lagraa, Mr Ben toubat, Mr Derradji, Mr Tahari, Mr Allawi, Mr Ben saad, Mr Boukhalkhal et l'ensemble des enseignants pour leurs aides et soutiens durant nos années d'études.

On' exprime gratitude envers les deux ingénieurs en informatique Mr H. Mourad et Mr B. Kuidar, pour leurs critiques et conseils avisés, et pour leurs encouragements.

Enfin, nous remercions toutes personnes qui nous ont aidé et encouragé de près ou de loin, à la réalisation de ce travail et au long de notre chemin universitaire.

Dédicaces

Je dédie ce modeste travail à :

- Mon mari, Sofiane, mes parents et à toute ma famille pour leur soutien infaillible durant toutes mes années d'étude.

- Mon binôme Khadidja.

- Sans oublier ma promo de DEUA informatique 2005 et ma promo 2008.

Safia

Je dédie ce modeste travail à :

- Mes très chers parents et mes grands parents qui ont sacrifié pour moi et qui m'ont soutenu tout au long de mon chemin d'étude.

- Mes frères et mes sœurs surtout mon petit frère Lamine.

- Mes tantes et oncles leurs enfants.

- Toute ma famille sans exception.

- Toute mes chère amies que je l'aime beaucoup.

- Tout ce qui m'aime et aime.

- Toute la promo 2008, et les étudiants de l'informatique.

- Mon binôme Safia.

Khadidja

Sommaire

Résumé

No table of contents entries found.

Chapitre II : Prétraitement des images numérique

No table of contents entries found. Chapitre III : Méthodologie de détection de contours

No table of contents entries found.

Chapitre IV : Paramètres de sélection

No table of contents entries found.

Chapitre V : Présentation du logiciel

No table of contents entries found. **Conclusion générale.....68**

Bibliographie.....69

Table des figures

Chapitre I : Traitement numérique des images

No table of contents entries found.5

Chapitre II : Prétraitement des images numérique

No table of contents entries found.

No table of contents entries found.

Chapitre III : Méthodologie de détection de contours

No table of contents entries found.

Chapitre IV : Paramètres de sélection

[No table of contents entries found.](#)

Chapitre V : Présentation du logiciel

No table of contents entries found. Figure V.7 : Le menu Edition.....	61
Figure V.8 : Le menu Prétraitement	61
Figure V.9 : Sous menu filtrage linéaire	61
Figure V.10 : Sous menu filtrage non linéaire	62
Figure V.11 : Filtrage médian.....	62
Figure V.12 : Binarisation.....	63
Figure V.13 : Histogramme	64
Figure V.14 : Le menu Extraction de contours.....	64
Figure V.15 : Extraction de contours par le détecteur Sobel	65
Figure V.16 : Le menu Extraction automatique de contours	66
Figure V.17 : Extraction automatique de contours	66
Figure V.18 : Le menu Aide	67

Liste des tableaux

Chapitre I : Traitement numérique des images

No table of contents entries found.

Chapitre II : Prétraitement des images numérique

No table of contents entries found.

No table of contents entries found.

Chapitre IV : Paramètres de sélection

No table of contents entries found.

Résumé

La détection des contours d'une image est une opération non triviale puisqu'il existe un grand nombre de détecteurs et que chaque détecteur a des performances différentes suivant les caractéristiques de l'image, Les caractéristiques retenues dans cette étude sont le bruit, le flou, l'homogénéité, le niveau de détail et l'entropie.

Ce mémoire propose une approche de sélection automatique d'un détecteur de contours qui applique le détecteur le plus approprié aux caractéristiques de chaque image. La première étape fait appel à des estimateurs des caractéristiques de l'image. La deuxième étape est rendue possible grâce à un travail préliminaire d'analyse et d'expérimentation d'une sélection de détecteurs de contours et d'identification d'un ensemble de règles heuristiques de détermination du détecteur le plus approprié aux caractéristiques de l'image.

L'approche est validée sur un banc d'essai d'images et s'est avérée plus performante que l'application directe de n'importe quel détecteur sur l'image.

Introduction générale

Introduction :

Le traitement de l'image est considéré comme un sous-domaine du traitement du signal, science au carrefour des mathématiques, de la physique et de l'informatique. Les années vingt voient le balbutiement dans le domaine de l'image sans véritable évolution jusqu'à la fin de la seconde guerre mondiale.

Les techniques de traitement du signal, et par conséquence celles de l'image, évoluèrent grâce à l'arrivée du radar ainsi que la prospection pétrolière. Dans les années quatre-vingt, le milieu médical s'intéresse aux images et particulièrement aux techniques d'imageries afin d'améliorer les diagnostics. De plus, dans les années quatre-vingt-dix, le public se familiarise au traitement de l'image grâce au logiciel Photoshop ou Gimp sur le système UNIX.

Dans ce mémoire, nous nous intéresserons uniquement aux méthodes de détection de contours qui est une opération fondamentale en traitement d'image car elle est à la base de la reconnaissance de formes.

Face à un nombre important de techniques et filtres de détection de contours, il peut être difficile de choisir l'approche la plus adaptée à une collection spécifique d'images d'autant plus qu'il n'existe pas de technique optimale pour tous les cas de figures. Par ailleurs, les techniques de détection de contours analysent souvent une image dans sa globalité sans tenir compte des spécificités des composantes de l'image.

Pour mener à bien cette opération de détection de contours et remédier à cet état de fait, il est utile d'étudier dans un premier temps un ensemble de techniques et d'analyser leur performance et adaptabilité selon les caractéristiques des images dans leur globalité (image réelle, floue ou nette, bruitée). Ensuite, une approche automatique de détection de

contours sera proposée. Finalement, les résultats de l'analyse seront présentés sous forme de règles heuristiques de sélection de détecteurs.

L'extraction des contours est utilisée, dans la majorité des cas, pour faciliter l'étude des détails de l'image. L'application la plus simple est le rehaussement des contours lorsque ceux-ci sont trop flous. Cependant, l'utilisation de ces techniques sert aussi à la recherche d'image selon le contenu, au repérage d'objets spécifiques dans des images ou encore à reconstruire des objets en trois dimensions. Pour ce genre d'application, il est indispensable d'extraire les contours des objets présents afin de pouvoir les reconnaître ou les reconstruire. Extraire les contours est donc souvent un prétraitement indispensable de l'image avant la réalisation d'autres traitements.

Lorsque l'on souhaite extraire des contours, nous remarquons très rapidement que le nombre de techniques existantes est très grand et que chaque détecteur possède ses propres caractéristiques. Cela est d'autant plus contraignant que les performances de ces détecteurs varient suivant les caractéristiques de l'image. Il faut donc essayer d'identifier le détecteur le plus adéquat à une image donnée. Pour cela, il est nécessaire de déterminer pour chaque image les caractéristiques de bruit, de flou ou encore de contraste d'entropie, niveau de détail et d'homogénéité. Autant de paramètres qu'il peut paraître subjectif de déterminer à l'oeil nu.

Dans ce mémoire de fin d'étude, nous présentons le problème de détection automatique de contours, ensuite nous abordons le sujet des caractéristiques des images. Enfin, nous faisons un état de l'art des différents détecteurs que nous avons choisi pour faire nos comparaisons. Finalement, le but de notre travail est de déterminer les caractéristiques des images et le ou les détecteurs le(s) plus adapté(s) à une image donnée. Une étude empirique complète le mémoire et sert de validation de l'approche proposée.

Dans le premier chapitre, nous abordons brièvement, une introduction sur le traitement d'images, les caractéristiques des images numériques ainsi que quelques techniques de traitement d'images utilisées afin d'analyser le contenu d'une image à des fins de calibration, reconnaissance et de localisation d'objets polyédriques.

Dans le deuxième chapitre, on va voir que le terme de prétraitement regroupe souvent toutes les opérations qui sont appliquées aux images, indépendamment de leur usage futur, pour leur assurer une bonne qualité.

Elles concernent donc essentiellement les corrections de contraste et la suppression du bruit.

Dans le troisième chapitre, nous décrivons quelques méthodes de détection de contours, les plus connus et les plus représentatifs tel que Robert, Sobel, Prewitt, Kirsch, Laplacien, Marr et Hildreth , Canny et Deriche.

Dans le quatrième chapitre, nous visons à identifier les détecteurs et les paramètres les plus appropriés à divers cas d'images. Pour cela, on s'intéresse à identifier des caractéristiques de l'image par une estimation du flou du bruit, de contraste, d'homogénéité, de niveau de détail et d'entropie. Puis, par des expérimentations sur une variété d'images et de détecteurs, on essaie d'établir des règles de choix de détecteurs.

En fin dans le cinquième chapitre, on représente le logiciel qui a été réalisé pour la sélection automatique de contours.

Chapitre I :

Traitement numérique des images

Nous abordons brièvement, dans ce chapitre, une introduction sur le traitement d'images, les caractéristiques des images numériques ainsi que quelques techniques de traitement d'images utilisées afin d'analyser le contenu d'une image à des fins de calibration, reconnaissance et de localisation d'objets polyédriques.

1. Introduction : [1]

Le traitement d'images désigne une discipline des mathématiques appliquées qui étudie les images numériques et leurs transformations, dans le but d'améliorer leur qualité ou d'en extraire de l'information.

Il s'agit donc d'un sous-ensemble du traitement du signal dédié aux images et aux données dérivées comme la vidéo (par opposition aux parties du traitement du signal consacrées à d'autres types de données : son et autres signaux monodimensionnels notamment), tout en opérant dans le domaine numérique (par opposition aux techniques analogiques de traitement du signal, comme la photographie ou la télévision traditionnelles).

Dans le contexte de la vision artificielle, le traitement d'images se place après les étapes d'acquisition et de numérisation, assurant les transformations d'images et la partie de calcul permettant d'aller vers une interprétation des images traitées. Cette phase d'interprétation est d'ailleurs de plus en plus intégrée dans le traitement d'images, en faisant appel notamment à l'intelligence artificielle pour manipuler des connaissances, principalement sur les informations dont on dispose à propos de ce que représentent les images traitées (connaissance du domaine).

La compréhension du traitement d'images commence par la compréhension de ce qu'est une image. Le mode et les conditions d'acquisition et de numérisation des images traitées conditionnent largement les opérations qu'il faudra réaliser pour extraire de l'information. En effet, de nombreux paramètres entrent en compte, les principaux étant :

- La résolution d'acquisition et le mode de codage utilisé lors de la numérisation, qui déterminent le degré de précision des éventuelles mesures de dimensions,
- Les réglages optiques utilisés, (dont la mise au point) qui déterminent par exemple la netteté de l'image,
- Les conditions d'éclairage, qui déterminent une partie de la variabilité des images traitées,
- Le bruit de la chaîne de transmission d'image.

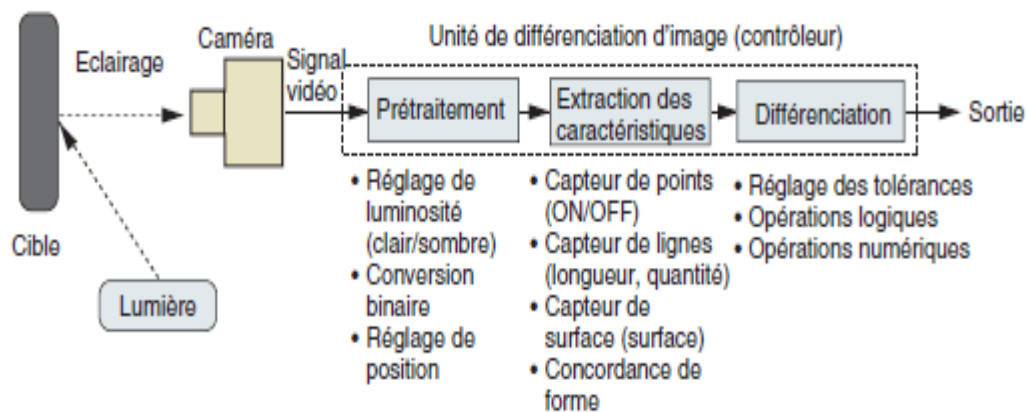


Figure I.1 : Organigramme du traitement des images.

2. Image numérique :

On désigne sous le terme d'image numérique toute image (dessin, icône, photographie ...) acquise, créée, traitée ou stockée sous forme binaire (suite de 0 et de 1) :

- Acquisée par des Convertisseurs Analogique-Numérique situés dans des dispositifs comme les scanners, les appareils photo ou caméscopes numériques, les cartes d'acquisition vidéo (qui numérisent directement une source comme la télévision).
- Créée directement par des programmes informatiques, via la souris, les tablettes graphiques ou par la modélisation 3D (ce que l'on appelle par abus de langage les « images de synthèse »).
- Traitée grâce à des outils informatiques. Il est facile de la transformer, modifier en taille, en couleur, d'ajouter ou supprimer des éléments, d'appliquer des filtres variés, etc.
- Stockée sur un support informatique (disquette, disque dur, CD-ROM, ...).

3. Caractéristiques d'une image numérique :

3.1. Pixel : [2]

Le *pixel* (abréviation venant de l'anglais : *picture element*) est l'élément de base d'une image, ou d'un écran, c'est-à-dire un point. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image.

À chaque pixel est associée une couleur, usuellement décomposée en trois composantes primaires (Rouge, vert, bleu).

3.2. Voisinage : [3]

Chaque point –image $I(x, y)$ a quatre voisins horizontaux-verticaux et quatre voisins diagonaux. Tableau(I.1) :

$(x-1, y-1)$	$(x-1, y)$	$(x-1, y+1)$
$(x, y-1)$	(x, y)	$(x, y+1)$
$(x+1, y-1)$	$(x+1, y)$	$(x+1, y+1)$

Tableau I.1 : les 8 voisins d'un pixel.

Les *voisins* horizontaux-verticaux de $I(x, y)$ sont appelés les quatre voisins et on dit qu'ils sont quatre adjacents à $I(x, y)$. Les voisins de $I(x, y)$ définissant le reste du voisinage 3×3 de la figure ci-dessus sont appelés les 8 voisins et on dit qu'ils sont 8-adjacents à $I(x, y)$.

3.3. Définition : "Le poids de l'image" [4]

La *définition* d'une image est le nombre de pixels total, c'est-à-dire sa "dimension informatique" (le nombre de colonnes de l'image que multiplie son nombre de lignes). Une image possédant 640 pixels en largeur et 480 pixels en hauteur aura une définition de 640 pixels par 480, notée 640x480.

3.4. Résolution numérique : "la finesse de l'image"

La *résolution* d'une image composée de points est définie par la densité des points par unité de surface. Elle permet de définir la finesse de l'image. Plus la résolution est grande, plus la finesse de l'image est grande.

Donc la *résolution* définit le nombre de pixels par unité de longueur (centimètre ou pouce). La résolution d'une image numérique s'exprime en **PPI** (**P**ixel **P**er **I**nch) ou **PPP** (**P**ixels **P**ar **P**ouce).

3.5. La luminance :

C'est le degré de *luminosité* des points de l'image .elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface. Souvent le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet.

3.6. Niveau de gris : [1]

Le *niveau de gris*, en imprimerie, désigne la concentration des points de trame et est donc directement en rapport avec le rendu de l'image. Un niveau de gris va alors varier du blanc au noir. Cela est également valable pour toutes les autres couleurs (cyan, magenta, jaune). On fera alors varier la densité du cyan du blanc jusqu'au cyan foncé.

Dans le cas d'une image numérique, le niveau de gris représente l'intensité lumineuse d'un pixel, lorsque ses composantes de couleur sont identiques en intensité lumineuse.

3.7. Le contraste :

Le *contraste* est une propriété intrinsèque à une image qui permet de quantifier, la capacité de distinguer deux régions distinctes. Il s'agit dans ce cas de distinguer deux régions suffisamment grandes d'après l'intensité des points représentés par des niveaux de gris en image numérique.

3.8. La netteté :

Définition 1 : Une image de clarté en termes de concentration et de contraste. Quand une image a beaucoup de petits détails, on dit avoir bon contraste. Contraste est une partie intégrante de netteté de l'image, on peut dire qu'elle est la distinction entre les renseignements dans le cadre d'une image qui peut vous aider à voir les détails fins.

Définition 2 : La *netteté* est une impression visuelle qui n'est pas liée à la résolution. Il semble que ce soit la précision des contours qui est l'élément déterminant dans l'impression de netteté. Or, le terme scientifique qui exprime la précision des contours est l'"acutance": elle se mesure au niveau des transitions

entre les zones sombres et les zones claires: plus les transitions sont rapides (sur une courte distance), plus l'acutance est élevée, et avec elle l'impression de netteté.

Définition 3 : La *netteté* est une qualité subjective, elle désigne la capacité d'un matériel à distinguer la nuance de volume ou de couleur la plus subtile, sans nuire à sa lisibilité.

On distingue :

- **La netteté OPTIQUE :** Qui est conditionnée par la nature des objectifs, des lentilles, et des mécaniques de l'appareil. C'est bien entendu un paramètre sur lequel nous n'avons aucune latitude de choix, si ce n'est celui d'envisager l'acquisition d'un appareil plus performant (Cette qualité devrait être déterminante lors d'un achat).
- **La netteté ÉLECTRONIQUE :** Qui est un procédé d'accentuation artificielle des contours et des contrastes au stade de la numérisation, personnalisable en fonction des documents et des opérateurs. On le désigne sous le terme de SHARPNESS ou d'USM. Il consiste en une accentuation des écarts de densité du document sur la frange de transition.

3.9. Texture :

La *texture* est le terme pour caractériser la distribution spatiale des éléments d'une surface donnée dans une image.

En imagerie, elle peut être définie par des relations spatiales que possèdent les niveaux de gris des pixels voisins, qui contribuent à l'apparence globale de l'image.

3.10. Régions :

Partie de l'image où les pixels possèdent une forte connexité entre eux et ayant des propriétés communes qui les différencient des pixels des autres régions.

3.11. Bruits :

Le *bruit* dans une image numérique est considéré comme un phénomène de brusque variation d'un pixel isolé par rapport à ses voisins. Il s'agit de parasites disgracieux, et particulièrement sensibles sur les images prises dans de mauvaises conditions d'éclairage. Et on distingue deux types de bruits:

3.11.1. Le bruit additif : [5]

Le bruit additif peut être défini de la façon suivante :

Etant donnée une image non bruitée R et I la même image avec un bruit additif A , alors chaque pixel j est caractérisé par la relation : $I_j = A_j + R_j$

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

3.11.2. Le bruit multiplicatif : [5]

Le bruit multiplicatif se définit de la façon analogue :

Etant donnée une image non bruitée R et I la même image avec un bruit multiplicatif B , alors chaque pixel j est caractérisé par la relation : $I_j = B_j * R_j$

Où B_j est une variable aléatoire de moyenne égale à 1.

La principale caractéristique de ce bruit est que les pixels d'une zone homogène seront d'autant plus bruités quand leur niveau de gris est élevé.

3.12. Contours et frontières :

La *frontière* est la notion associée à une variation d'intensité ou à une discontinuité entre les propriétés de deux ensembles connexes de points.

Les *contours* représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste à identifier dans l'image les points qui séparent deux textures différentes.

4. Acquisition de l'image :

L'*acquisition* d'une image se fait au travers d'une caméra *CCD* (Charge Coupled Device) qui est composée d'une matrice de photodiode (512x512). Chaque photodiode traduit la luminance par l'émission d'un signal électrique positif compris entre 0,3v et 1v (0,3v : noir, 1v : blanc et entre les deux se situent les différents niveaux de gris). Le dispositif électronique d'acquisition se charge de récolter les signaux, émis par les photodiodes, ligne par ligne de la matrice *CCD* espacée d'un bref temps de synchronisation.

Viens ensuite la conversion des signaux analogiques en données numériques positives comprises entre 0 et 255 (0 : noir, 255 : blanc et entre les deux se situent les différents niveaux de gris).

L'image, après acquisition, est représentée sous une autre forme, que son simple affichage, afin de mettre en évidence ses différentes caractéristiques.

L'image issue de la caméra *CCD* (caméra à transfert de charge) est composée de 3x256 couleurs ou de 256 niveaux de gris.

5. Prétraitement :

L'image brute est une structure simple dans son organisation mais complexe dans son contenu .ceci est du à plusieurs facteurs, d'une part la grande quantité d'informations qui y réside, d'autre part le nombre important de processus de l'utilisateur qui tentent à introduire des distorsions indésirables, donc à la dégrade.

On peut citer comme élément de perturbation : l'éclairage qui peut être trop faible et rend l'image trop sombre, le bruit électronique inhérent au capteur qui engendre une imprécision dans la restitution des niveaux de gris et la qualité optique de l'objectif qui peut engendrer des aberrations ainsi que la mise au point .l'amélioration d'images consiste en un ensemble de méthodes destinées à améliorer l'aspect visuel d'une image. Il existe différentes approches visant à améliorer une image :

- La modification de l'histogramme.
- La réduction de bruit (filtrage).

6. Segmentation de l'image :

Le traitement d'images s'intègre dans le processus de la vision par ordinateur comme un outil et n'a donc pas de pouvoir décisionnel. Le traitement d'images se définit comme un ensemble de tâches destinées à extraire de l'image brute des informations qualitatives et quantitatives.

Le processus de *segmentation* est le passage obligé pour la vision par ordinateur, qui peut être définie ainsi : "La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image A en sous-ensembles R_i , appelés *régions* tel qu'aucune région ne soit vide, l'intersection entre deux régions soit vide et l'ensemble des régions recouvre toute l'image.

En effet la littérature abonde sur le sujet et le nombre de travaux de recherche est important sur la segmentation des images numériques.

Nous distinguons deux entités de segmentation :

- L'une est basée sur la détection de contours des objets (**discontinuité**).
- L'autre est basée sur la mise en région de la couleur (**homogénéité**).

7. Conclusion :

On peut voir le traitement d'image comme un type des traitements numériques des signaux et comme un préalable à la reconnaissance des formes, à l'analyse de scènes et à l'intelligence artificielle. Ces relations riches avec des domaines variés font que la diversité des traitements que l'on peut effectuer par voie numérique sur des images est très grande. Pour illustrer cette diversité, on peut citer quelques exemples :

On peut chercher à améliorer sa qualité, il s'agit donc de techniques d'améliorations "*rehaussement*" qui font appel à des procédés variés (modification de l'histogramme, filtrage linéaire et non linéaire).

1. Améliorer une la qualité d'une image peut aussi signifier qu'on cherche à retrouver une image idéale de l'objet qui a été dégradée par divers processus : on veut corriger les distorsions géométrique d'un capteur,...etc.il s'agit donc de techniques de restauration.
2. On peut chercher à détecter la présence de certaines formes, certains contours ou certaines textures de modèle connu, sans vouloir préserver les autres informations contenues dans l'image. Il s'agit là de problème de détection.

Dans le chapitre suivant on va voir en détail les méthodes de prétraitement des images numériques.

Chapitre II :

Prétraitements des images numériques

Dans ce chapitre, on va voir que le terme de prétraitement regroupe souvent toutes les opérations qui sont appliquées aux images, indépendamment de leur usage futur, pour leur assurer une bonne qualité.

Elles concernent donc essentiellement les corrections de contraste et la suppression du bruit.

1. Introduction :

Le traitement d'images (ou le pré-traitement) regroupe l'ensemble des processus visant à améliorer les caractéristiques d'une image.

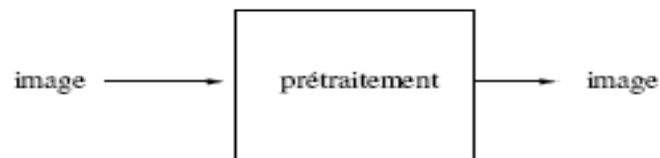


Figure II.1 : Le prétraitement.

Ce processus de bas niveaux regroupe :

- Le lissage local : il s'agit de supprimer le bruit, ou les petites variations, présent dans une image. L'intensité d'un pixel est transformée en fonction des intensités sur un petit voisinage du pixel.
- L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image (contraste, ...) pour faciliter son interprétation par l'œil humain.
- La restauration d'images a pour but de supprimer les dégradations subies par une image à l'aide de connaissance a priori sur ces dégradations.

2. Le filtrage (lissage local) :

Les opérations de filtrage ont pour objet d'atténuer l'effet du bruit et faire disparaître les défauts et les perturbations contenus dans une image. Cela fait que les images ont souvent un aspect brouillé à la réception. Il est alors nécessaire de filtrer ces images avant de les traiter. Il existe des filtres linéaires où la transformation d'un pixel est le fruit d'une combinaison linéaire des pixels voisins, et des filtres non linéaires où les pixels voisins interviennent suivant une loi non linéaire.

2.1. Le filtrage linéaire :

Ce type de filtrage permet de diminuer sensiblement les effets du bruit .une méthode simple consiste à considérer chaque pixel de l'image et d'en faire la moyenne avec les huit pixels qui lui sont voisins.

Il peut être mis en œuvre en utilisant un masque de la forme suivante :

$$H1 = \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Ensuite, on déplace ce masque sur toute l'image.

On distingue généralement les types de filtres linéaires les plus connus :

2.1.1. Filtre passe-bas :

Consistant à atténuer les composantes de l'image ayant une fréquence haute (pixels foncés). Ce type de filtrage est généralement utilisé pour atténuer le bruit de l'image, c'est la raison pour laquelle on parle habituellement de lissage. Les filtres moyenneurs sont un type de filtres passe-bas dont le principe est de faire la moyenne des valeurs des pixels avoisinants. Le résultat de ce filtre est une image plus floue. [6]

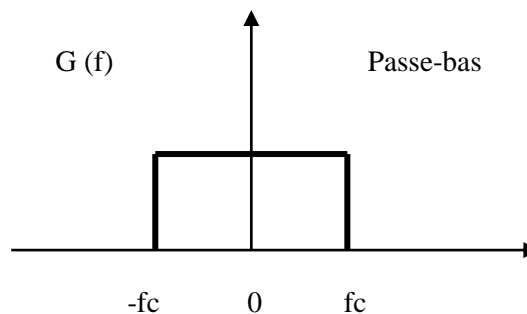


Figure II.2 : Filtre passe-bas.

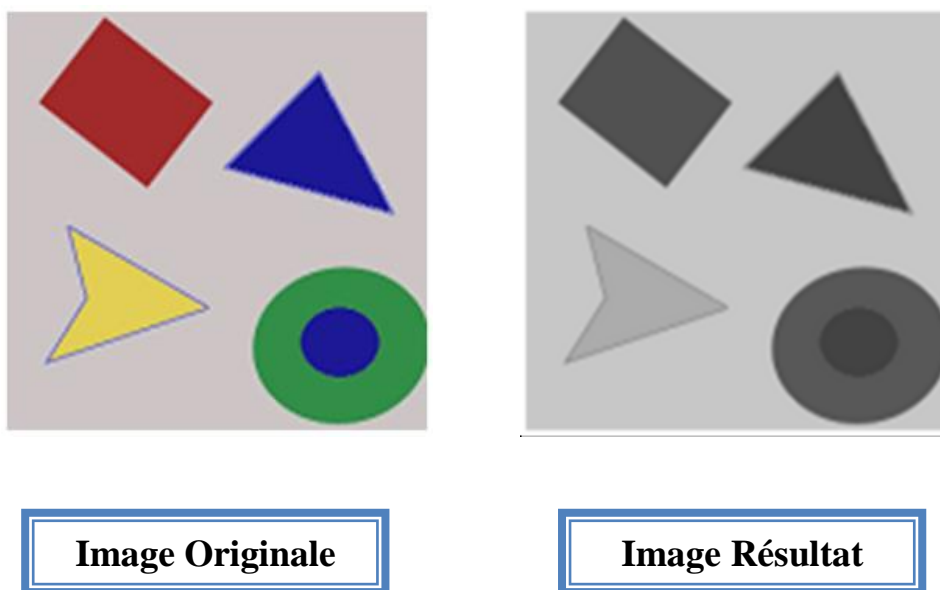


Figure II.3 : Résultat de filtrage Passe-bas.

2.1.2. Filtre passe- haut :

Les filtres passe-haut, à l'inverse des passe-bas, atténuent les composantes de basse fréquence de l'image et permettent notamment d'accentuer les détails et le contraste, c'est la raison pour laquelle le terme de "filtre d'accentuation" est parfois utilisé. [5]

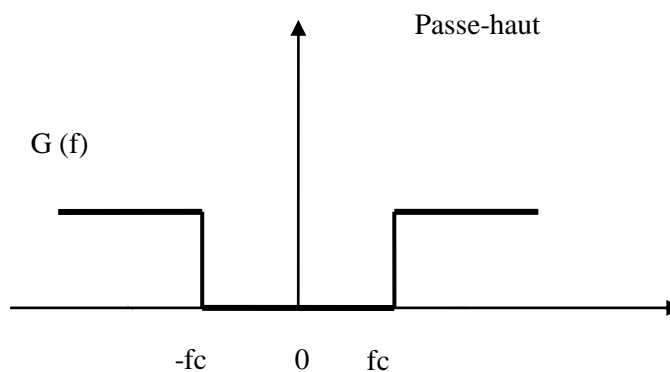


Figure II.4 : Filtre passe-haut.

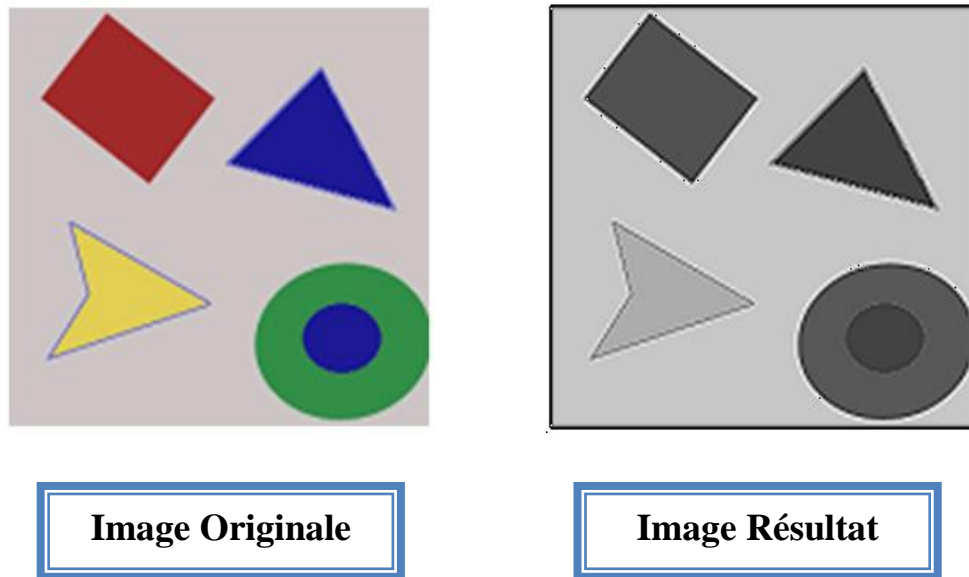


Figure II.5 : Résultat de filtrage Passe-haut.

2.1.3. Filtre passe bande :

Cette opération est une dérivée du filtre passe-bas .Les filtres passe-bande permettant d'obtenir la différence entre l'image originale et celle obtenue par application d'un filtre passe-bas. C'est à dire elle consiste à éliminer la redondance d'informations entre l'image d'origine et l'image obtenue par filtrage passe-bas.

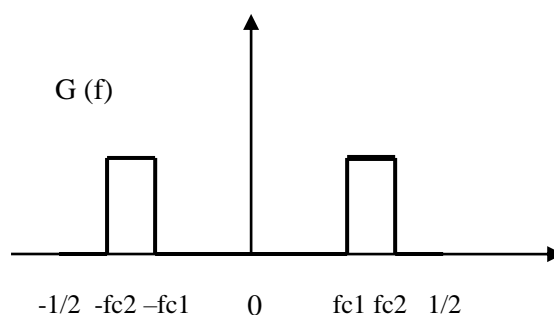


Figure II.6 : Filtre passe bande.

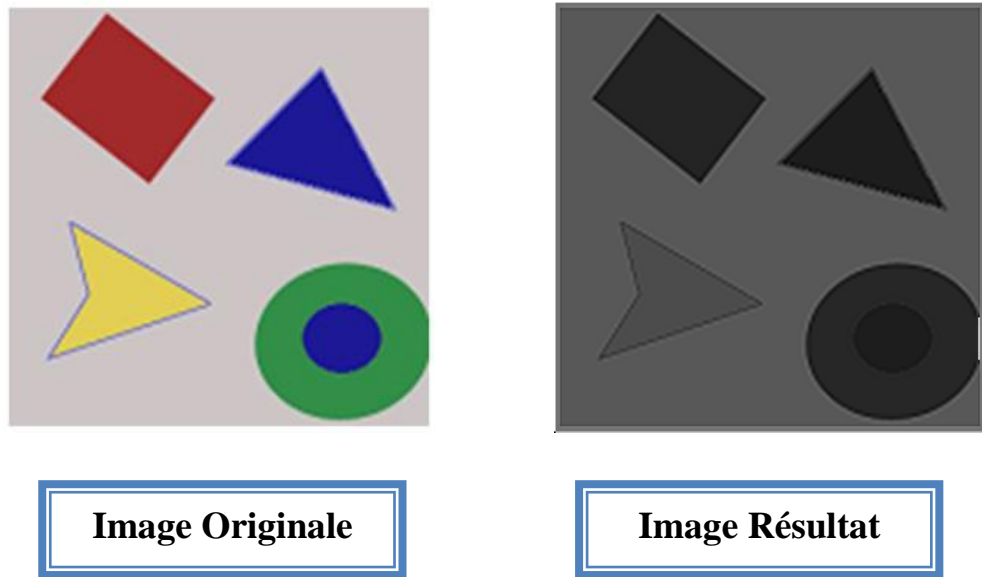


Figure II.7 : Résultat de filtrage Passe bande.

2.1.4. Moyenne : [3]

Le pixel de l'image filtrée est une combinaison de ses proches voisins. Leur moyenne en l'occurrence, est la convolution de l'image avec le masque suivant :

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Son intérêt est qu'il ne change pas trop les contours mais en revanche il intègre un flou dans l'image.

Si on veut avoir une plus grande netteté, on choisit un filtre qui attribue au point central le poids le plus élevé comme il est représenté par les masques suivants :

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

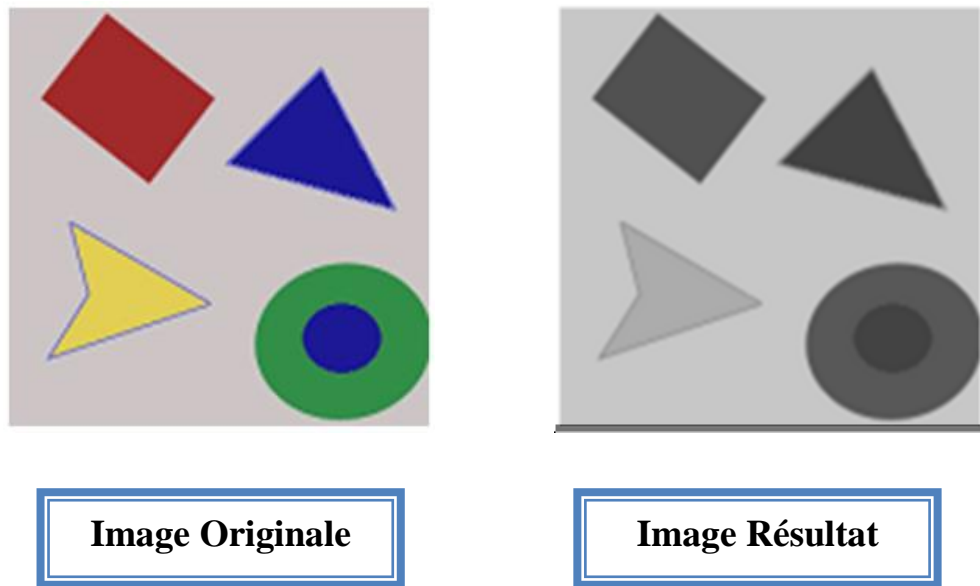


Figure II.8 : Résultat de filtrage Moyenne.

2.2. Le filtrage non linéaire :

Ces opérateurs ont été développés pour pallier aux insuffisances des filtres linéaires. Dans ces filtres le niveau d'un pixel de l'image filtrée est toujours fonction du niveau des pixels environnants, mais le calcul n'est plus basé sur des combinaisons linéaires.

Les filtres les plus connus sont le filtre médian, le filtre maximum, le filtre minimum et le filtre morphologique.

2.1.1. Filtre médian:

Le filtrage médian est plus adapté que le filtrage linéaire pour réduire le bruit : car une image comporte des pixels aberrants (par exemple un seul pixel blanc au milieu d'une zone noire ou des pixels isolés répartis aléatoirement dans l'image qui dégradent la qualité de l'image) on dit qu'elle est "bruitée".

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.

La méthode basée sur le filtre médian est particulièrement adaptée lorsque le bruit est constitué de points isolés ou de lignes fines. Elle n'est applicable qu'aux images en niveaux de gris, ce filtre permet de prendre toutes les valeurs de niveaux de gris des pixels formant la valeur du pixel considéré et son voisinage, puis trie pour mettre la valeur *médiane* est celle situé au milieu de la liste obtenue à la place du pixel concerné.

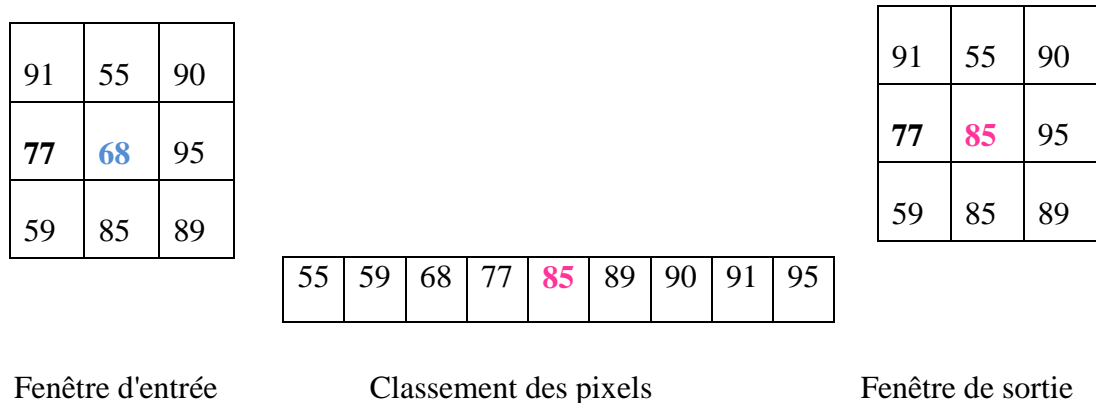


Figure II.9 : Application d'un filtre médian 3*3 sur une fenêtre 3*3.

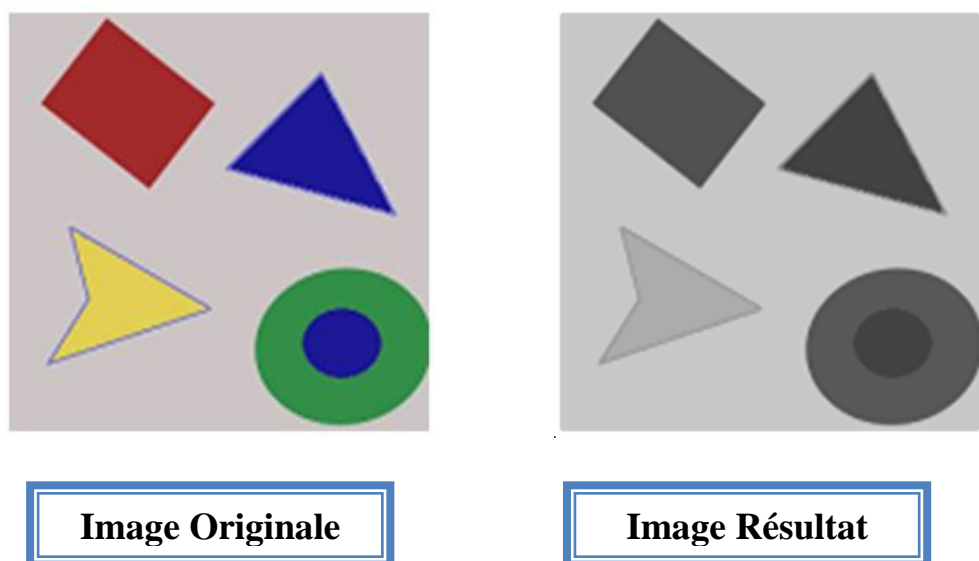


Figure II.10 : Résultat de filtrage Médian.

2.1.2. Filtre maximum:

On applique le même principe que le filtre médian mais le pixel central dans ce cas sera remplacé par la valeur maximale selon le filtre de l'ensemble de pixels de la fenêtre de voisinage.

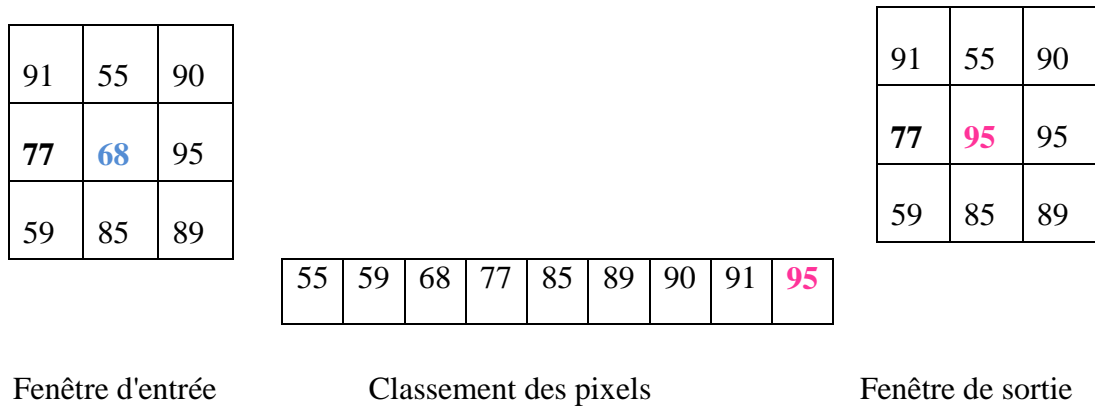


Figure II.11 : Application d'un filtre maximum 3*3 sur une fenêtre 3*3.

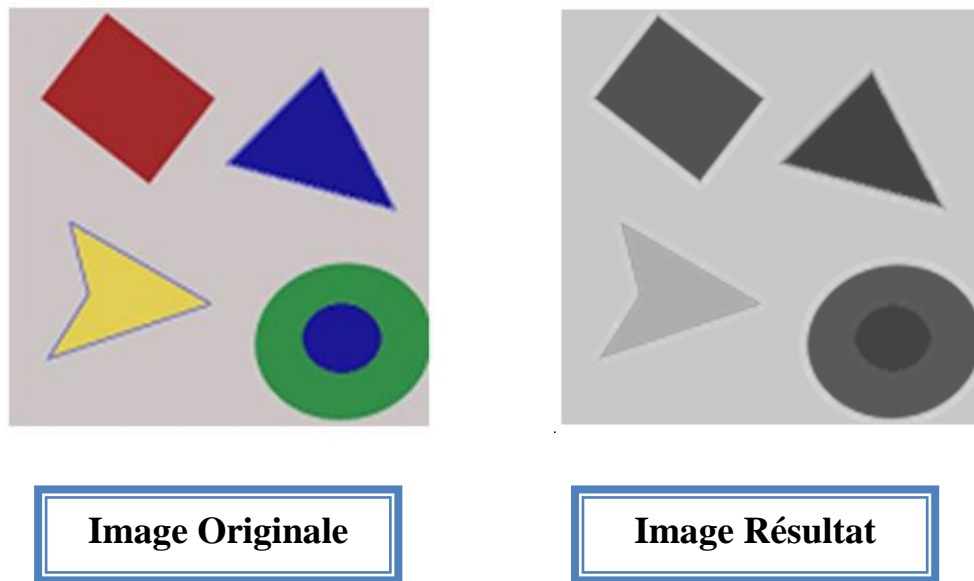


Figure II.12 : Résultat de filtrage Maximum.

2.1.3. Filtre minimum:

Pour ce cas la valeur minimale prend la place du pixel concerné.

91	55	90
77	68	95
59	85	89

55	59	68	77	85	89	90	91	95
----	----	----	----	----	----	----	----	----

91	55	90
77	55	95
59	85	89

Fenêtre d'entrée

Classement des pixels

Fenêtre de sortie

Figure II.13 : Application d'un filtre minimum 3*3 sur une fenêtre 3*3.

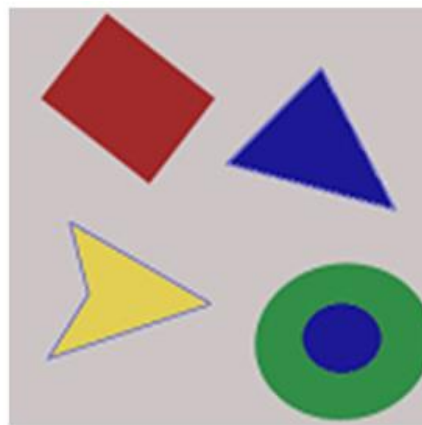


Image Originale

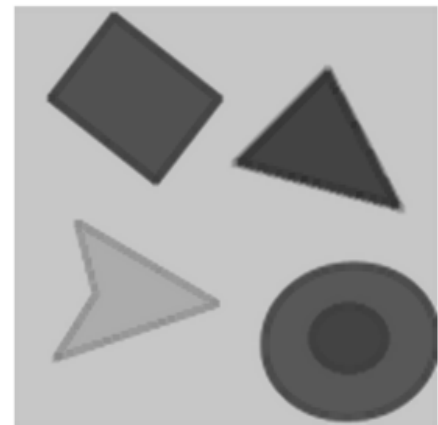


Image Résultat

Figure II.14 : Résultat de filtrage Minimum.

2.1.4. Filtre morphologique :

Ce type de filtrage est utilisé pour éliminer des pixels isolés dans une image binaire (deux niveaux de gris : noir (0) et blanc (1)), qui sont considérés comme du bruit, il met en correspondance chaque pixel et ses voisins par une fonction logique (*ET*, *OU*), mais dans le cas d'une image à plusieurs niveaux de gris, le pixel, concerné est traité avec ses voisins par une fonction mathématique (*maximum*, *minimum*) respectivement. Parmi les opérateurs morphologiques nous citons :

a) La dilatation :

La *dilatation* a pour effet d'augmenter la taille des pixels. Elle effectue le *OU* logique pour des images binaires, ou la recherche de la plus grande valeur d'intensité pour des images à plusieurs niveaux de gris des huit voisins (au plus) du pixel considéré.

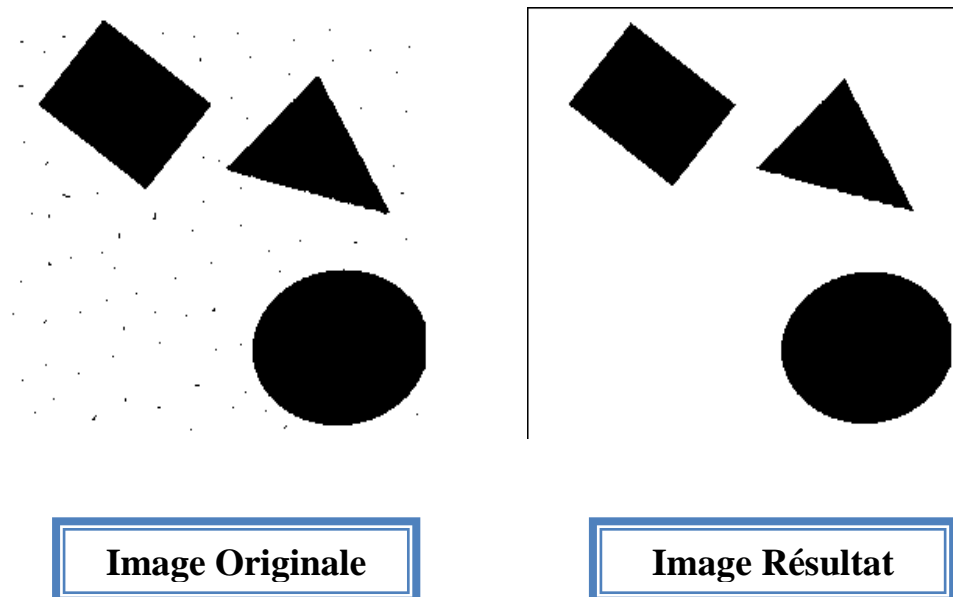


Figure II.15 : Résultat de filtrage Morphologie (Dilatation).

b) L'érosion :

L'*érosion* consiste à réduire les pixels. Elle effectue le *ET* logique pour des images binaires, ou la recherche de la plus petite valeur d'intensité pour des images à plusieurs niveaux de gris des huit voisins (au plus) du pixel considéré.

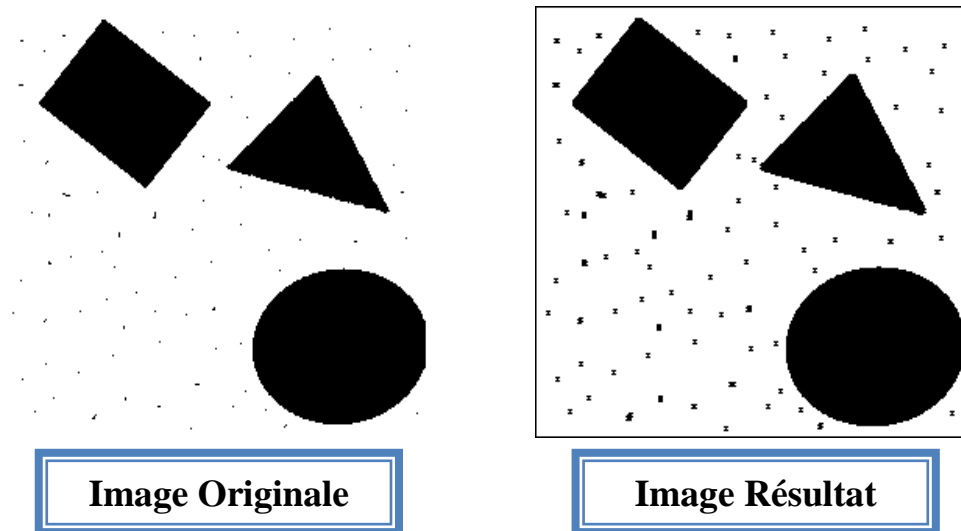


Figure II.16 : Résultat de filtrage Morphologie (Erosion).

c) L'ouverture :

L'*ouverture* consiste à éliminer des presque îles étroites des pixels. En appliquant d'abord une *érosion*, et ensuite une *dilatation*.

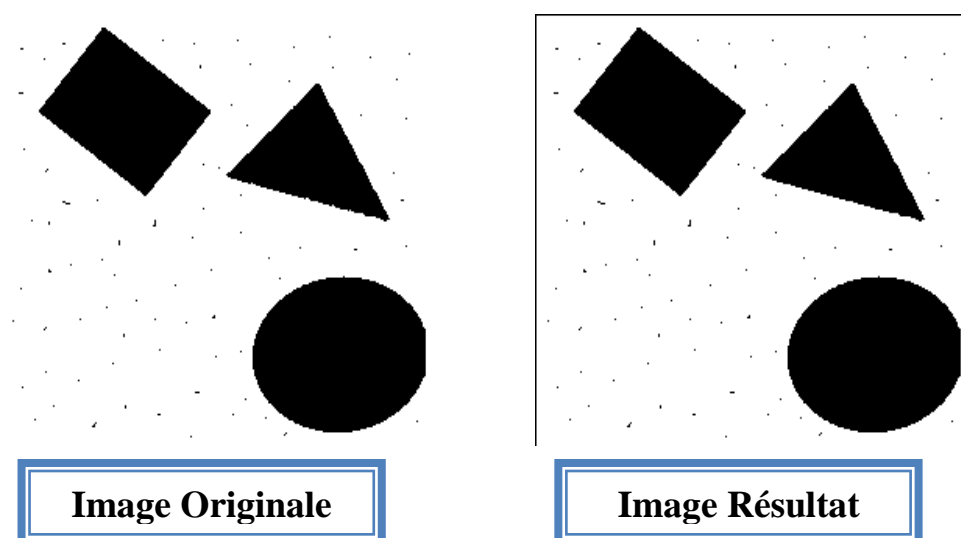


Figure II.17 : Résultat de filtrage Morphologie (Ouverture).

d) Fermeture :

A l'inverse de l'ouverture, la *fermeture* est une opération qui ferme les golfes et les petits trous. Pour faire une fermeture, on applique d'abord une *dilatation*, et ensuite une *érosion*.

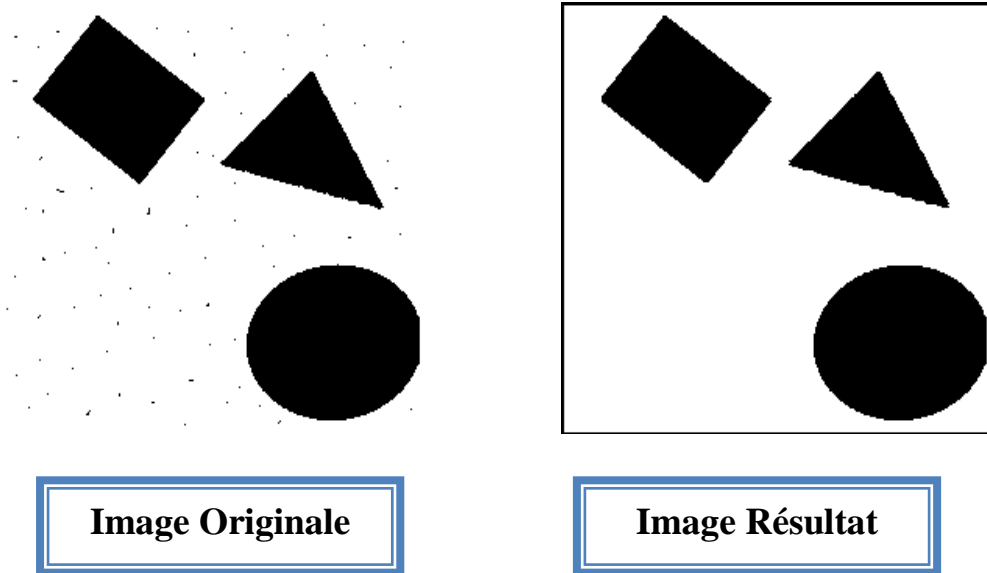


Figure II.18 : Résultat de filtrage Morphologie (Fermeture).

Interprétation des résultats :

Nous constatons que les images traitées avec le filtre moyen présentent un lissage de l'image, ce qui introduit un flou sur les contours des objets. Cependant, le filtre médian préserve les principales transitions entre régions sombres et claires.

En ce qui concerne le filtrage morphologique nous avons constaté que :

- L'érosion élimine les composantes connexes, élargit les trous, sépare les formes proches.
- La dilatation bouche les trous, élargit les caps, soude les formes proches.
- L'ouverture lisse les formes, élimine les composantes connexes, conserve souvent la taille et la forme, ne conserve pas la topologie.
- La fermeture bouche les trous, conserve souvent la taille et la forme, ne conserve pas la topologie (soude les éléments proches).

D'après ces résultats, nous avons opté pour le filtre médian car il préserve les contours et il donne des images nettes et fait disparaître les points parasites.

3. Amélioration d'images :

L'amélioration d'images consiste à modifier les caractéristiques visuelles de l'image de manière à en faciliter son interprétation par l'œil humain. Il peut s'agir de rehausser les contrastes, d'accentuer certaines intensités pour mettre en valeur une région, ... Les histogrammes sont fréquemment utilisés pour effectuer ce type d'opérations.

3.1. L'histogramme :

L'histogramme est une fonction permettant de donner la fréquence d'apparition des différents niveaux de gris qui composent l'image. En abscisse on représente les niveaux de gris et en ordonnées leurs fréquences d'apparition. L'histogramme des niveaux de gris nous informe sur la concentration de l'image.

On compte chaque pixel selon son niveau de gris par exemple pour l'image suivante :

1	1	2	2	3	1	4	12	10	1	3
3	2	2	2	4	11	10	9	9	8	3
5	2	2	6	9	0	0	15	1	7	7
9	2	6	1	5	5	11	10	7	7	7
3	3	11	8	9	10	10	5	9	7	7
8	1	1	3	4	8	9	9	8	7	3
8	9	0	0	1	1	0	6	6	7	8
6	8	5	1	3	5	3	6	1	7	9

Tableau II.1 : Les pixels avec son niveau de gris.

Le pixel de valeur 2 apparaît 8 fois.

Le pixel de valeur 7 apparaît 10 fois.

		8					10								
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Tableau II.2 : Regroupement de la répétition de valeur de chaque pixel.

Algorithme :

Soit H un tableau de taille égale au nombre de niveaux de gris de l'image (n)

- Pour tout i de 0 à n-1, on initialise H[i] à 0
- Pour tout pixel (x, y) de l'image I, $H[I[x, y]] = H[I[x, y]] + 1$

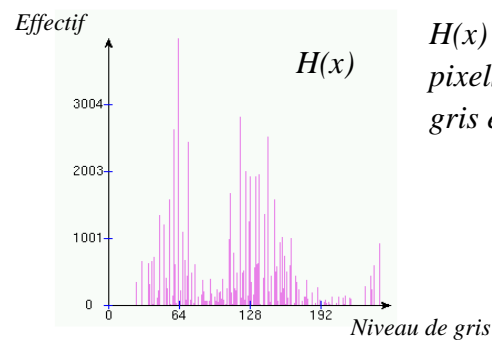


Figure II.19 : Image et son histogramme.

3.2. Modification d'histogramme :

Pour modifier les caractéristiques de l'image (accentuer les contrastes en général), une approche générale consiste à appliquer une fonction qui associe à chaque valeur d'intensité dans l'image une nouvelle valeur. Cette fonction va modifier l'histogramme de l'image.

Cette modification a pour but d'optimiser la dynamique des événements à analyser.

3.2.1. Linéarisation de l'histogramme : [7]

Pour améliorer l'information contenue dans l'image, on utilise une technique appelée linéarisation de l'histogramme ou encore **égalisation d'histogramme**. La transformation est construite de telle façon que le nouvel histogramme soit plat (distribution uniforme des niveaux de gris); cette opération est équivalente à maximiser l'entropie de l'image, on obtient alors théoriquement, une image ayant une information maximale. Pour une image qui présente un pic dans les niveaux sombres, cela revient à étendre la dynamique de la zone sombre au détriment de celle de la zone claire.

La transformation qui permet de s'approcher au mieux d'une distribution uniforme peut être obtenue par une fonction de distribution cumulative (FDC) ou histogramme cumulé, c'est-à-dire l'intégrale de l'histogramme original.

3.2.2. Amincissement itératif d'histogramme : [7]

A l'inverse de la linéarisation, le procédé appelé amincissement itératif d'histogramme se propose d'accroître les différences entre pics de l'histogramme par une réduction des niveaux de gris. Le principal inconvénient de cette modification d'histogramme est le non prise en compte de la répartition spatiale des niveaux de gris dans l'image. Pour tenir compte de cette répartition spatiale, il est nécessaire de faire appel à des techniques plus complexes relevant du domaine de la segmentation en régions et de l'analyse d'images.

3.3. Interprétation d'histogramme :

L'histogramme permet de donner un grand nombre d'informations sur la distribution des niveaux de gris et de voir entre quelles bornes est répartie la majorité des niveaux de gris dans le cas d'une image trop clair ou d'une image trop foncée. (Voir figure II.20).

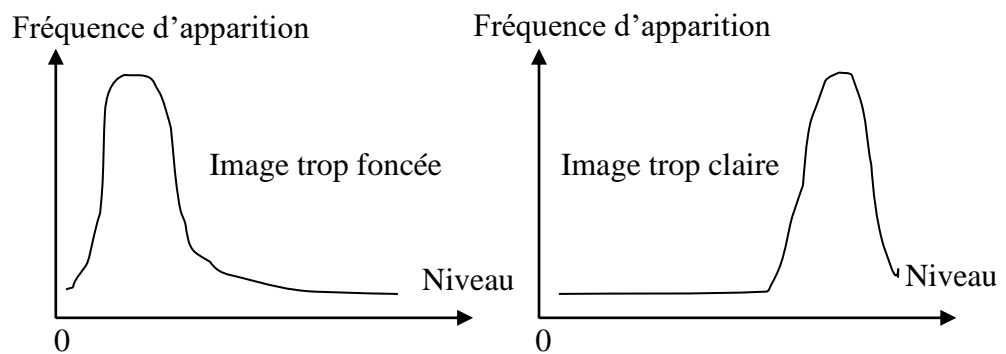


Figure II.20 : Histogramme d'image.

La forme de l'histogramme nous donne des informations sur les caractéristiques de l'image :

- Un histogramme étroit indique un manque de contraste dans l'image.
- Un histogramme concentré dans un intervalle des plus bas niveaux de gris indique une image trop sombre.
- Un histogramme concentré dans un intervalle des plus hauts niveaux de gris indique une image trop claire.
- Un histogramme peut aider à choisir les niveaux de seuils dans l'opération du seuillage.

Application et résultat :

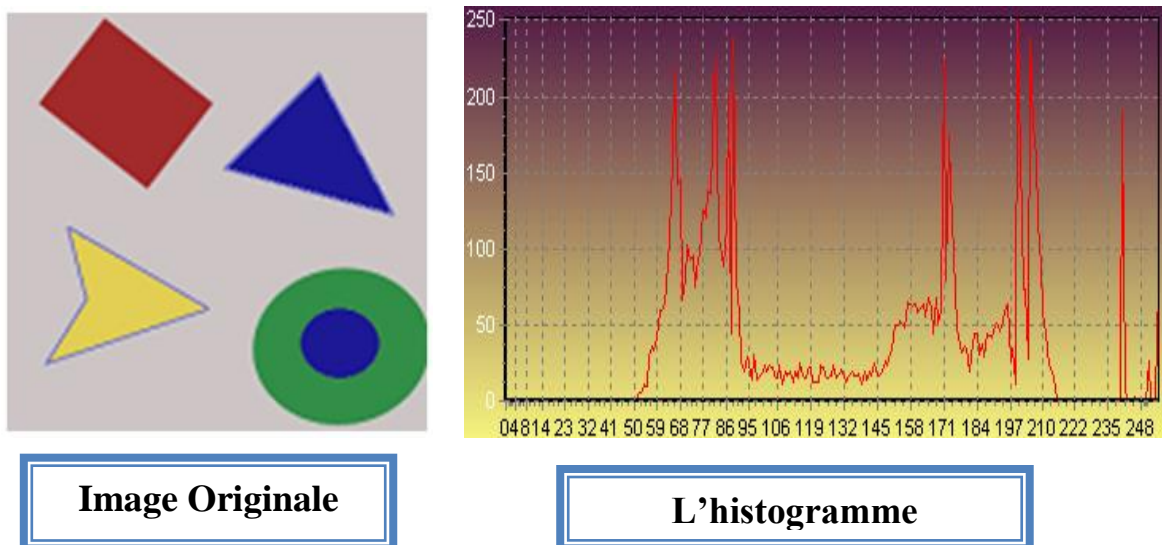


Figure II.21 : Histogramme.

4. La binarisation :

Binariser une image revient à trouver un seuil aux niveaux de gris. Un pixel dont la valeur est en dessous de ce seuil va alors prendre une des deux valeurs possibles et inversement un pixel dont la valeur est supérieure au seuil prendra la valeur opposée. La matrice représentant cette image ne va donc contenir que deux valeurs, typiquement des 0(noir) et des 1(blanc). Donc son but est de transformer une image multi-niveaux en une image binaire.

Pourquoi la binarisation ?

- La plupart des méthodes existantes travaillent sur une image simplifiée noire et blanche.
- Une bonne binarisation est une étape importante pour la reconnaissance.
- Une binarisation est un traitement irréversible car c'est le résultat d'une interprétation suivant un critère et une méthode.
- Les images non binarisées conservent toute l'information.

Il existe de très nombreuses techniques de binarisation toutes basées sur le principe de seuillage.

Algorithme de binarisation :

Début

Pour chaque pixel de l'image

Faire

Si le niveau de gris de pixel $> S$

Alors pixel = blanc (225)

Si non pixel = noir (0)

Fin si

Fait

Fin

Où S est le seuil qui est la valeur d'intensité entre deux classes, il est appliqué sur l'image selon ces trois méthodes, où dans les deux premières méthodes, les seuils sont choisis par l'utilisateur.

- Seuillage globale.
- Seuillage adaptatif.
- Seuillage automatique.

5. Les techniques de seuillage :

5.1. Le seuillage globale :

Cette technique traite d'une manière identique tous les points de l'image à partir d'un seuil déterminé préalablement.

5.2. Le seuillage adaptatif:

Il consiste à fixé deux seuils S_B et S_H tel que $S_B < S_H$.

Les pixels dont le niveau de gris est inférieur à S_B sont noirs, et ceux qui sont supérieures à S_H sont blanc tant dis que les pixels des niveaux de gris compris entre S_B et S_H , sont soit noir soit blanc selon la couleur de leurs voisins. Le choix de seuil peut être fait empiriquement, mais aussi d'une manière automatique.

5.3. Le seuillage automatique:

Dans cette technique le seuil est déterminé automatiquement par la méthode des différences cumulées qui est décrite comme suit :

Soit le voisinage suivant :

1	2	3
8	0	4
7	6	5

On définit la différence positive par :

$$S_p = \sum (y_0 - y_i) * 1(y_0 - y_i)$$

Et la différence négative par :

$$S_N = \sum (y_i - y_0) * 1(y_i - y_0)$$

Avec : $1|y| = 1$ si $y > 0$ et $1|y| = 0$ si $y < 0$

Si on cumule soit les différences positives, soit les différences négatives en parcourant toute l'image pour chaque niveau de gris, on obtient un tableau dont l'indice du maximum des cumules correspond au seuil recherché.

Application et résultat :

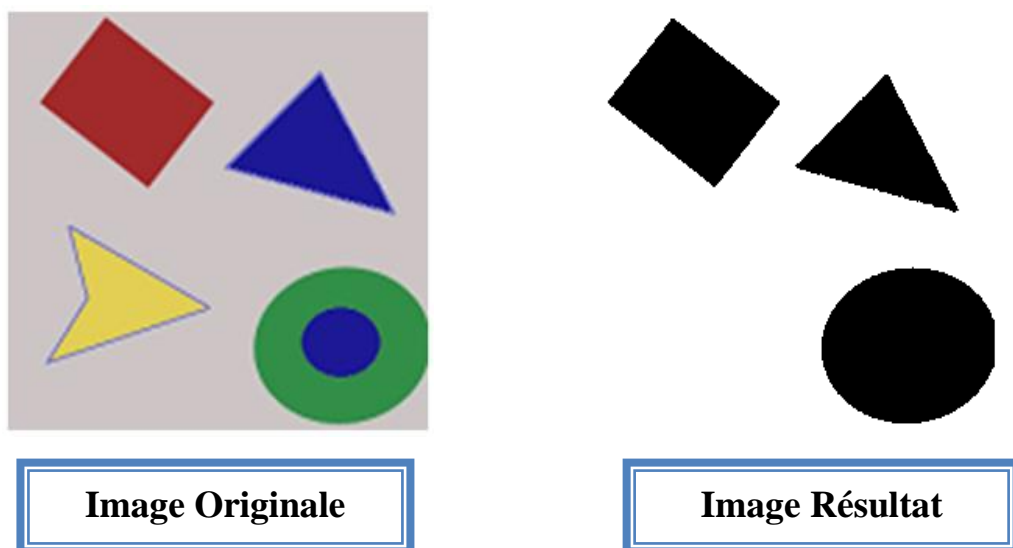


Figure II.22 : Binarisation.

6. Conclusion :

Le prétraitement est une étape très importante pour l'extraction des caractéristiques d'image, c'est un processus de bas-niveau, qui nécessite très peu d'informations sur le contenu des images. Il s'agit des processus de filtrage, d'amélioration et de restauration d'images.

Chapitre III :

La méthodologie de détection de contours

Nous décrivons quelques méthodes de détection de contours, les plus connus et les plus représentatifs tel que Robert, Sobel, Prewitt, Kirch, Laplacien, Marr et Hildreth, Canny et Deriche.

1. Introduction :

L'extraction de contours joue un rôle primordial dans tout système de vision par ordinateur. Beaucoup d'efforts ont été faits pour extraire les contours d'une image et plusieurs techniques ont été proposées à ce jour. Les contours correspondent généralement à des changements brusques de propriétés physiques ou géométriques de l'image perçue et forment ainsi des attributs très importants pour l'analyse.

Dans la plupart des techniques développées à ce jour, plusieurs opérateurs locaux (directionnels ou pas) de dérivation du premier ordre et du second ordre sont utilisés. Ils sont alors suivis respectivement d'une recherche de maxima locaux et de passage par zéro. Toutefois, ces opérateurs ne donnent souvent pas, à eux seuls, de bons résultats sur une image réelle où les changements d'intensité sont rarement nets et abrupts. Une opération de seuillage et un suivi de contours sont en effet nécessaires pour éliminer le bruit et affiner les contours à une épaisseur d'un pixel.

Depuis quelques années, on assiste cependant à de nouvelles approches en extractions de contours qui procèdent par optimisation de critères prenant en compte un modèle prédéfini du contour à détecter. Ces approches qui permettent une meilleure formulation du problème se sont avérées nettement plus performantes que les toutes premières techniques proposées.

Mais avant d'entamer les détails d'extraction des contours, il serait nécessaire de connaître les différents types des contours et quelles que caractéristiques de ces derniers.

2. Différents types de contours :

Dans une image, un contour peut être considéré de différentes manières. Nous allons décrire ici trois principales manières de considérer un contour :

Premièrement, un contour peut être vu comme un changement brusque de l'intensité de l'image. Il existe plusieurs types de variations comme le montre la figure III.1.

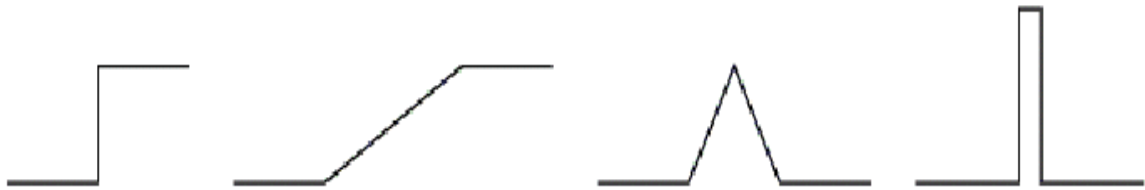


Figure III.1 : Profil de contours : marche, rampe, toit, pic.

Deuxièmement, une façon très proche de celle citée ci-dessus est de considérer les contours comme une différence sur la couleur.

Troisièmement, si on considère l'image comme étant un signal 2D, on peut passer dans le domaine fréquentiel (par transformée de Fourier par exemple). Dans ce cas, un contour peut être vu comme représentant les hautes fréquences du signal.

Cependant, ces représentations sont parfois insuffisantes pour représenter un contour. À titre d'exemple, nous remarquons que parfois notre cerveau est capable de repérer des contours invisibles sur l'image car il a la capacité d'utiliser ses connaissances pour extraire des formes d'une image. Par exemple, le cerveau humain est capable de voir le triangle dans l'image suivante :

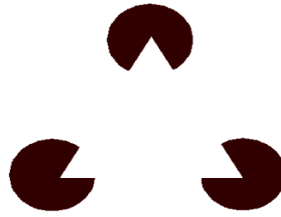


Figure III.2 : Triangle invisible pour un ordinateur

Cependant, dans certains cas ce triangle pouvait quand même être détecté grâce à des techniques spécifiques comme par exemple les algorithmes de Hough ou des algorithmes d'objets saillants.

3. Caractéristiques des contours :

Les caractéristiques d'un contour sont:

- **La continuité des points contours d'un objet :** Un contour est représenté par une chaîne connexe de points sur l'image. Il est possible de caractériser la forme des contours et d'utiliser des méthodes de reconnaissance de forme pour les comparer entre eux. L'utilisation du voisinage ou contexte d'un point contour donne des informations de forme et d'orientation.
- **Epaisseur d'un pixel:** Les contours représentent les frontières des objets. Il est important d'évaluer avec précision la position du contour de l'objet et de le représenter par un seul point qui correspond à la zone de variation maximale de la luminance.
- **La direction du contour est orthogonale à celle du gradient :** Le contour correspond à la frontière de l'objet et la variation de luminance est perpendiculaire au contour sur l'image.
- **Robustesse au bruit:** Il faut éliminer les points qui du fait des imperfections du système de prise de vue présentent les caractéristiques d'un contour, afin d'obtenir un contour aussi proche que celui obtenu sans influence du bruit.

4. Les méthodes de détection de contour :

Deux approches :

1. **Approche gradient** : détermination des extrema locaux dans la direction du gradient.
2. **Approche laplacien** : détermination des passages par zéro du laplacien.

Ces approches reposent sur le fait que les contours correspondent des discontinuités d'ordre 0 de la fonction d'intensité.

Le calcul de dérivée nécessite un pré-filtrage des images. Filtrage linéaire pour les bruits de moyenne nulle (par exemple bruit blanc Gaussien, filtre Gaussien). Filtrage non-linéaire pour les bruits impulsionnels (filtre médian par exemple).

4.1. Méthodes dérivatives du premier ordre :(Approche Gradient)

Étant donné que l'on définit un contour comme une discontinuité locale de l'intensité lumineuse, nous nous basons sur la méthode du gradient pour détecter les contours.

Le gradient correspond à la normale d'un contour. Si la valeur du gradient est différente de zéro on en déduit que l'on se situe dans une zone de transition entre une partie claire et une partie sombre.

Si $f(x, y)$ représente une image, le gradient n'est autre que l'application directe de la dérivée première et il s'écrit:

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x}(x, y) \\ \frac{\partial f}{\partial y}(x, y) \end{bmatrix}$$

Le Gradient a pour amplitude :

$$\|\nabla f(x, y)\| = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2}$$

Bien que cette amplitude soit souvent approchée par une des deux formules suivantes :

$$\|\nabla\| \approx |\nabla_x| + |\nabla_y| \quad \text{ou} \quad \|\nabla f\| = \max(|\nabla_x f|, |\nabla_y f|)$$

Sa direction est donnée par :

$$\phi = \text{Arc tan} \frac{\frac{\partial f(x, y)}{\partial y}}{\frac{\partial f(x, y)}{\partial x}}$$

L'analyse du vecteur gradient permet de mettre en évidence un contour. En effet l'amplitude du gradient indique une discontinuité plus ou moins forte et sa direction est par définition normale au contour. Le gradient est un opérateur directionnel.

Localement à un point de l'image, les dérivées partielles font l'objet d'une approximation par une méthode de différentiation.

$$\frac{\partial f(x, y)}{\partial x} \quad \text{Est évalué par} \quad \Delta_x f(x, y) = f(x + 1, y) - f(x, y)$$

$$\frac{\partial f(x, y)}{\partial y} \quad \text{Est évalué par} \quad \Delta_y f(x, y) = f(x, y + 1) - f(x, y)$$

A cette méthode on succède les opérateurs de **Roberts**, **Prewitt** et **Sobel** ou **Kirsch** qui donnent une estimation de la dérivée directionnelle de l'image par rapport à un axe. Que nous représentons sous la forme de masque de convolution :

4.1.1. Présentation des opérateurs de Roberts :

Roberts approxime les dérivées directionnelles suivant les axes orientés à 45 degrés à l'aide d'une convolution de l'image $f(x, y)$ avec les masques 2×2 suivant :

$$H_0 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad H_1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Soient $f_x = f * H_0$ et $f_y = f * H_1$ les images obtenues par convolution de l'image originale f avec les deux masques de H_0 et H_1 . L'image avec les contours accentués f_a est alors obtenue comme étant l'image approximant la norme du gradient :

$$f_a = \|\vec{\nabla} f\| = \sqrt{f_x^2(x, y) + f_y^2(x, y)}$$

Ce détecteur est malheureusement très localisé et donc très sensible au bruit, ce qui nuit grandement à ses performances. Les autres sont moins sensibles au bruit et offrent de meilleurs résultats.

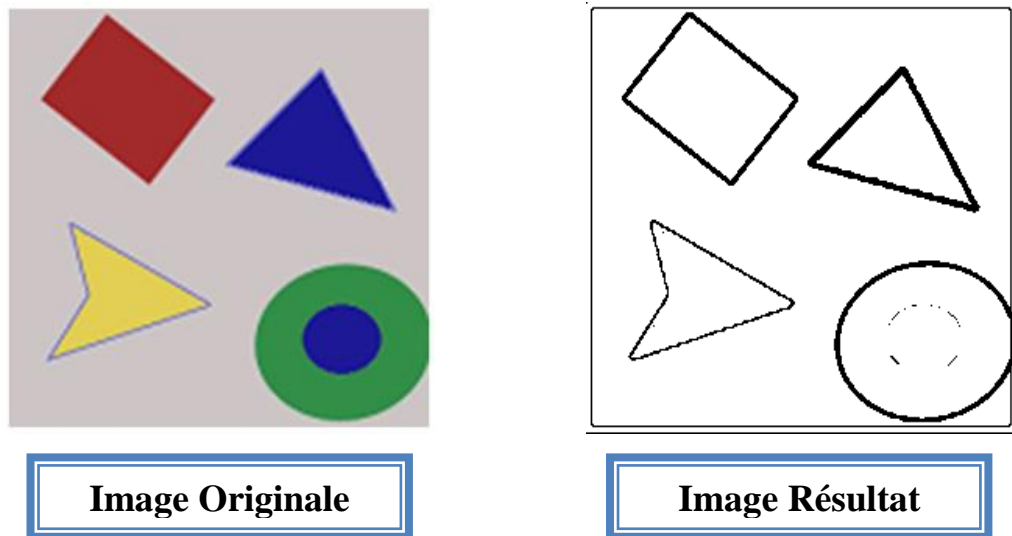


Figure III.3 : Extraction de contour par le détecteur Robert.

4.1.2. Présentation des opérateurs de Sobel et Prewitt :

Ces détecteurs utilisent un filtre linéaire qui a l'avantage d'effectuer deux opérations en même temps, soit un lissage de l'image et soit une dérivation. Ils restent sensibles au bruit mais donnent de bons résultats sur des images non bruitées. Voici les deux filtres impliquant un opérateur horizontal et un opérateur vertical.

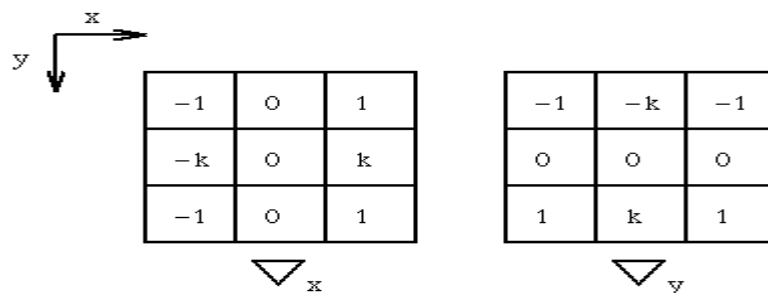


Figure III.4 : Masques de convolution des opérateurs de *Prewitt* ($k=1$) et de *Sobel* ($k=2$)

L'opérateur de *Prewitt* est défini pour une valeur de $k=1$ et *Sobel* pour $k=2$. Ces opérateurs calculent les variations dans seulement deux directions et ont

tendance à donner des contours légèrement épais, ce qui dans certains cas peut être gênant. Cependant, il existe des détecteurs qui font des rotations sur les coefficients des filtres pour inclure des dérivées dans d'autres directions. C'est ce que fait le détecteur suivant.

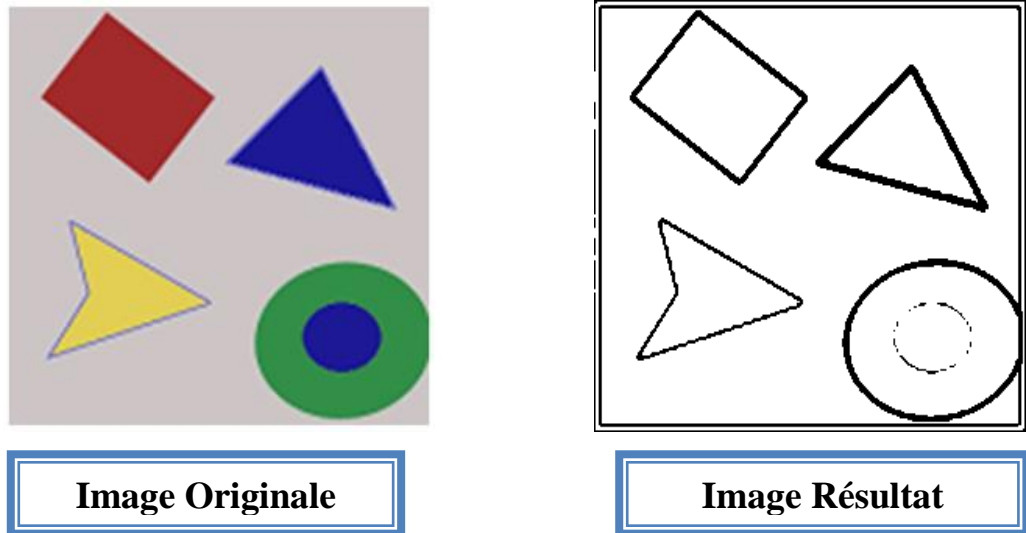


Figure III.5 : Extraction de contour par le détecteur Sobel.

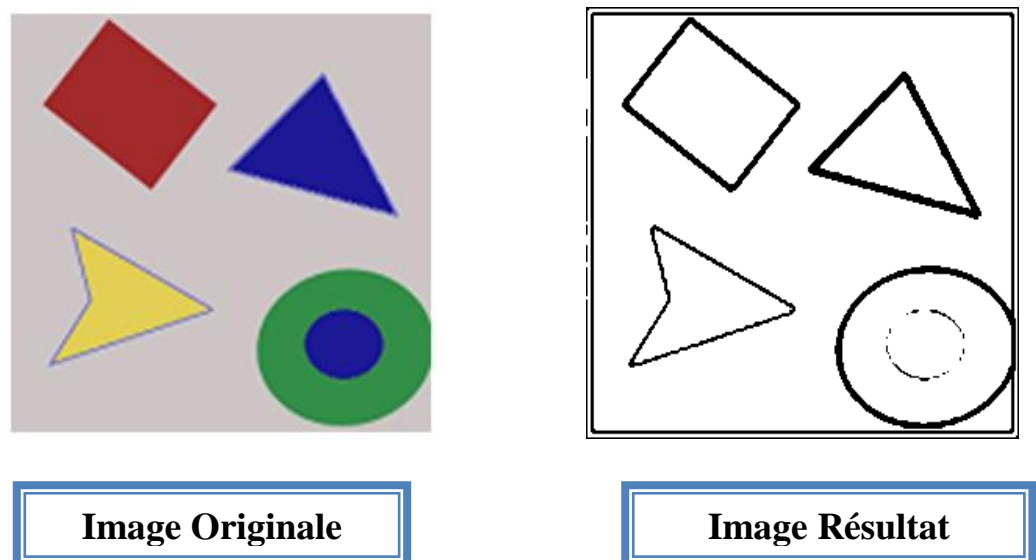


Figure III.6 : Extraction de contour par le détecteur Prewitt.

4.1.3. Présentation des opérateurs de Kirsch :

Lorsque plusieurs contours se rejoignent en un point, la détection sur seulement deux directions peut être insuffisante. L'opérateur de **Kirsch** est un opérateur à huit masques correspondant chacun à une direction préférentielle et obtenue par rotation de $\pi/8$ de l'opérateur de base H_0 :

$$H_0 = \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & 3 \\ -3 & -3 & -3 \end{bmatrix} \quad H_1 = \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} \dots$$

Cependant, les faibles résultats et la lourdeur de ce détecteur font qu'il est peu utilisé.

Le gradient retenu sera celui correspondant à la valeur maximum donnée

$$\text{par :} \quad \max_{i=1, \dots, 8} |H_i * f|$$

L'orientation retenue pour le contour sera celle correspondante au masque ayant permis d'obtenir le gradient maximum.

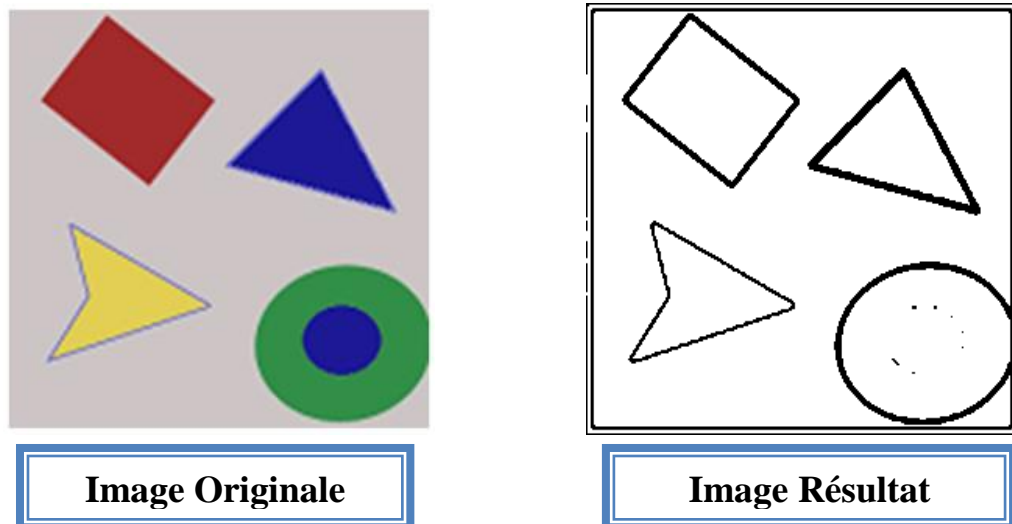


Figure III.7 : Extraction de contour par le détecteur Kirsch.

4.2. Méthodes dérivatives du deuxième ordre : (Approche Laplacien)

4.2.1. La méthode du Laplacien :

L'opérateur *Laplacien* est un opérateur de filtrage linéaire. Le contour se trouve là où le Laplacien change de signe, c'est à dire là où la dérivée seconde passe par zéro. Ce passage par zéro étant unique le Laplacien fournit directement un contour d'un pixel d'épaisseur.

Le Laplacien de la fonction Image est défini par :

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2}(x, y) + \frac{\partial^2 f}{\partial y^2}(x, y)$$

Ainsi en utilisant également les propriétés de différentiation l'opérateur Laplacien peut s'écrire sous la forme suivante :

$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

Généralement écrit par le masque de convolution suivant :

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Nous remarquons que l'opérateur Laplacien est symétrique quelque soit l'orientation choisie, il n'est donc pas directionnel. L'opérateur Laplacien est fortement sensible aux bruits. Ce qui conduit à essayer d'atténuer les bruits par l'emploi de filtre de lissage.

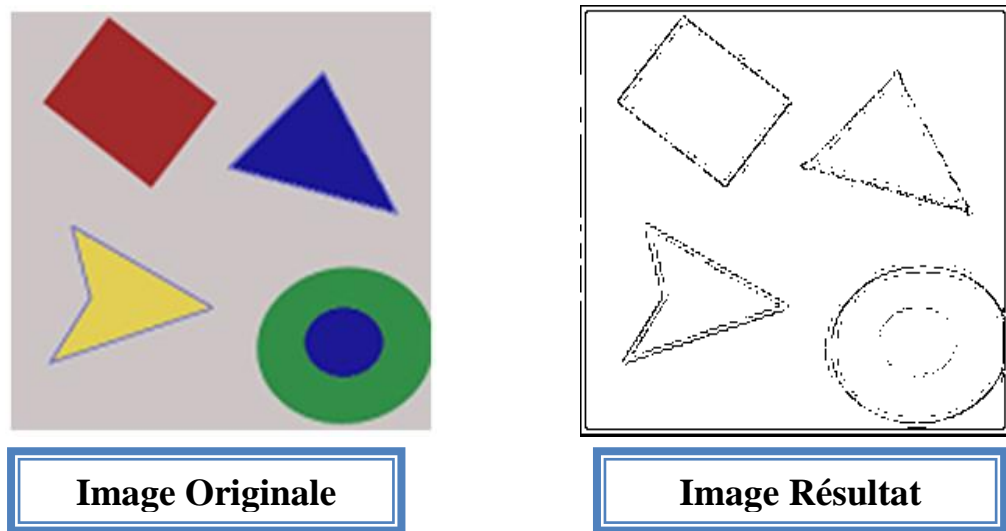


Figure III.8 : Extraction de contour par le détecteur Laplacien.

4.2.2. La méthode de Marr et Hildreth : [8]

Marr et Hildreth ont développé une technique permettant de régler le problème du bruit dans les images dont on veut détecter les contours. Ils ont fait une convolution de l'image avec un Laplacien d'une gaussienne (LoG filter). Cela permet de supprimer du bruit qui aurait été détecté par le Laplacien seul. Mais il faut cependant ajuster la variance de la Gaussienne en fonction du type d'image à traiter. Un filtre de variance trop large dégrade trop l'image (en la rendant trop floue et en éliminant les contours) et une variance trop faible conduit à une détection de bruit trop importante.

Dans le cas continu, l'opérateur de *Marr et Hildreth* correspond à la convolution de l'image de coordonnées x et y par l'opérateur dérivée seconde d'une Gaussienne de variance σ .

En deux dimensions, le filtre Gaussien s'écrit :

$$G_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

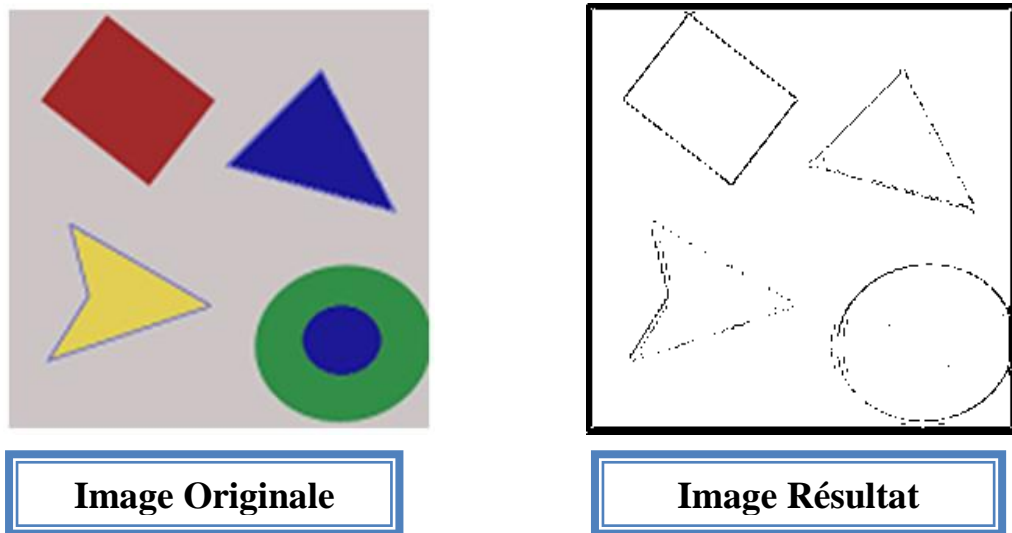


Figure III.9 : Extraction de contour par le détecteur Marr et Hildreth.

4.3. Méthodes par filtrage optimal : (Approche optimale)

4.3.1. La méthode de Canny : [9]

Comme dans le cas de la détection de points d'intérêt la détection des contours consiste essentiellement à repérer les changements locaux d'intensité sur l'image en niveau de gris.

Le détecteur de *Canny* repose sur quatre étapes :

- Un filtrage Gaussien pour limiter les fausses détections dues au bruit.
- Un calcul de gradient d'intensité pour chaque pixel afin de faire ressortir les variations d'intensité.
- Une suppression des non-maximas par un seuillage sur le module du gradient.
- Un seuillage hystérésis qui détermine et relie les pixels où le gradient est suffisamment grand pour qu'ils figurent sur un contour.

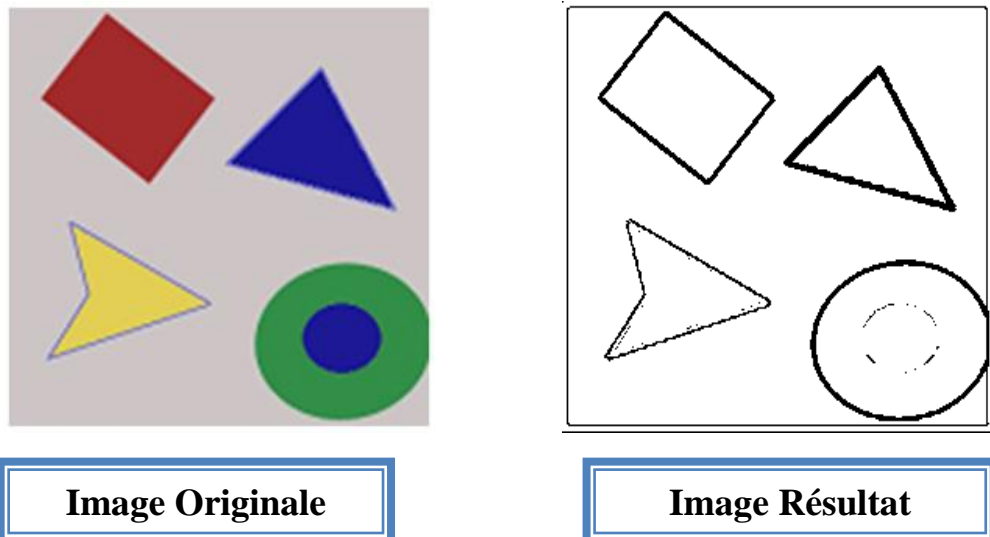


Figure III.10 : Extraction de contour par le détecteur Canny.

4.3.2. La méthode de Deriche :

Les opérateurs de Deriche sont utilisés dans deux grandes méthodes de détection de contours : l'une basée sur les maxima locaux du gradient, l'autre sur les passages par zéro du laplacien. Les méthodes récentes combinent le gradient et le laplacien. Les opérateurs implémentés sont ceux proposés par Rachid Deriche.

5. Conclusion :

Dans ce chapitre, nous avons présenté un aperçu des détecteurs les plus connus et les plus représentatifs. La majorité des détecteurs sont basés sur des principes similaires et proposent des convolutions de l'image avec des filtres de lissage et de dérivation. Nous avons représenté des détecteurs issus des techniques de dérivation des images (Robert, Sobel, Prewitt, Kirsch, Laplacien, Marr et Hildreth) ainsi que deux détecteurs dits optimaux (Canny et Deriche) en expliquant leurs différences au niveau des filtres utilisés.

Chapitre IV :

Paramètres de sélection

Nous visons à identifier les détecteurs et les paramètres les plus appropriés à divers cas d'images. Pour cela, on s'intéresse à identifier des caractéristiques de l'image par une estimation du flou du bruit, d'homogénéité, de niveau de détail et d'entropie. Puis, par des expérimentations sur une variété d'images et de détecteurs, on essaie d'établir des règles de choix de détecteurs.

1. Introduction :

Nous allons dans cette partie expliquer les algorithmes et techniques que nous avons utilisées pour extraire les caractéristiques des images comme le bruit, le flou, l'homogénéité, le niveau de détail et l'entropie. Ensuite nous présenterons la méthode de sélection d'un détecteur en fonction de ces caractéristiques.

2. Le banc d'essai :

Afin de tester notre application de manière efficace, nous avons conçu un banc d'essai contenant plusieurs images qui ont été sélectionnées suivant les critères suivants :

- Les images contiennent des objets de domaines très différents (vie courante, paysages, dessins, radiographies, images de musée, ...)
- Certaines images contiennent du flou ou des zones floues
- Certaines images contiennent du bruit ou des zones bruitées.

Nous avons aussi ajouté à notre banc d'essai des images très connues en détection de contours (Lena, Muscle) car ces images ont l'avantage d'avoir servi de nombreuses fois pour tester des détecteurs de contours. Elles nous offrent donc certaines références.

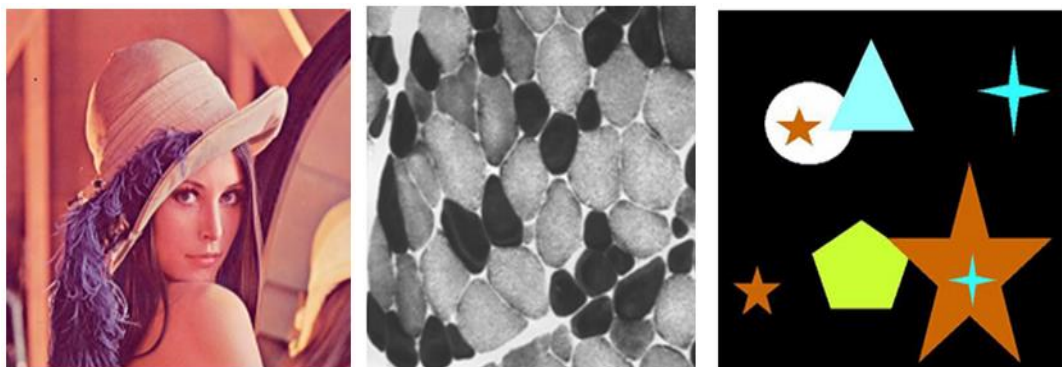


Figure IV.1 : Lena, Muscle, Etoile.

3. Estimation du bruit:

Il sera essentiel dans notre travail d'avoir une certaine estimation du bruit. Cela nous permettra, entre autres, de choisir la variance la plus appropriée à l'élimination du bruit dans l'image. Le bruit étant dans les hautes fréquences de l'image, nous avons utilisé un filtre passe haut afin de laisser passer uniquement les hautes fréquences, c'est à- dire le bruit et les contours. Nous avons utilisé le filtre Robert pour calculer la dérivée de l'image. Ce dernier laisse effectivement passer les contours et le bruit de l'image. Ensuite nous avons utilisé des formules de moyenne et d'écart type pour calculer la variance des variations du signal.

$$\bar{x} = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} D(x, y)}{NM}$$

Où \bar{x} est la moyenne des variations du signal D (signal dérivé), x et y sont les coordonnées d'un pixel dans l'image, N est le nombre de lignes de l'image et M est le nombre de colonnes. À partir de cette moyenne nous allons calculer la variance du signal avec la formule suivante :

$$\sigma^2 = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (D(x, y) - \bar{x})^2}{NM}$$

σ^2 représente la variance du signal D . Enfin, on va normaliser notre estimation en divisant cette variance par la variance maximale qui est :

$$\sigma_{\max}^2 = (|\bar{x}| + 255)^2$$

Dans cette dernière, le 255 représentent le nombre de niveaux de gris possibles dans l'image.

4. Estimation du flou : (EF)

L'estimation que nous avons utilisé se base sur une remarque très simple et permet d'estimer globalement la quantité de flou gaussien dans une image.

Le principe que nous avons utilisé est le suivant : Plus une image est floue, moins elle sera affectée par un filtre gaussien. Où encore, plus un signal n'est lisse moins le lissage aura d'effet sur lui.

Supposons que $f(x, y)$ représente l'image originale, $f'(x, y)$ représente l'image filtrée par une gaussienne et $e(x, y)$ est la différence entre f et f' . Nous calculons ensuite le rapport entre l'image originale $f(x, y)$ et l'effet d'une gaussienne $e(x, y)$ comme suit :

$$\text{diff}_{f,f'} = \frac{\sum_{x,y} f(x,y)^2}{\sum_{x,y} e(x,y)^2} \text{ avec } e(x,y) = f'(x,y) - f(x,y)$$

Nous rappelons que plus les images sont similaires, plus le $e(x, y)$ tendra vers zéro et donc **EF** sera grande.

Enfin plutôt que d'estimer le flou directement sur l'image originale, nous avons préféré estimer l'évolution du flou de l'image entre deux variances, ceci pour mesurer l'évolution de l'effet de la gaussienne par rapport aux mêmes références quelque soit l'image à analyser. Ainsi pour estimer le flou d'une image I , nous allons la filtrer par une gaussienne de variance 1 ce qui donnera une image I_1 , puis nous filtrerons I par une gaussienne de variance 4 qui donnera l'image I_2 . Notre estimateur de flou est alors calculé par :

$$\text{Estimation du flou : } \mathbf{EF} = \mathbf{diff} \ I_1, I_2$$

Il est important de retenir que cet estimation est globale et ne permet pas de savoir où se situent les zones floues. Elle permet juste de savoir si l'image dans sa globalité contient du flou. L'estimation peut donc donner des résultats difficiles à analyser dans le cas où l'image contient des zones très nettes et des zones très floues.

5. L'entropie et le niveau de détail :

Il est important lorsque nous allons étudier nos images de pouvoir savoir si elles contiennent beaucoup ou peu de détails. Une image contenant beaucoup de détails sera traitée différemment d'une image très homogène. Pour cela nous allons calculer deux coefficients, l'entropie et le niveau de détail.

5.1. L'entropie : (H)

L'entropie est très utilisée en compression de données. Elle permet de quantifier la quantité d'informations contenue dans une donnée. Si on prend une image de N pixels où chaque pixel varie entre 0 et 255 niveaux de gris, on appelle n_i le nombre de pixels pour le niveau de gris i . Alors la probabilité qu'un pixel x soit de la couleur i est:

$$P(x \in i) = \frac{n_i}{N}$$

On définit l'entropie de l'image par la formule suivante :

$$H(\text{Image}) = - \sum_{i=0}^{255} P(x \in i) * \log(P(x \in i))$$

En analysant l'entropie des images étoiles, Lena et people.

On a trouvé les résultats suivants :

$H(\text{étoiles}) = 0.8696$, $H(\text{Lena}) = 2.1685$ et $H(\text{Muscle}) = 2.2196$.

On en déduit donc facilement que l'image (étoile) contient beaucoup moins d'informations que les images (Lena) et (Muscle). Ceci nous servira dans le choix des détecteurs de contours, puisque lorsqu'une image contient très peu d'information, il suffira d'utiliser un détecteur de contour très simple (comme Robert par exemple).

5.2. Niveau de détails : (D)

Le niveau de détails de l'image est une donnée supplémentaire que nous utilisons pour estimer la quantité de contours présents dans l'image. Pour cela, nous examinons l'histogramme des niveaux de gris de l'image. Plus l'image contient de niveaux de gris différents plus le niveau de détail va être important. On calcule donc cette valeur avec la formule suivante :

$$D(\text{Image}) = \frac{\text{Nombre de niveaux de gris non nuls dans l'histogramme de l'image}}{\text{Nombre de niveau de gris}}$$

On obtient les résultats suivants sur les trois images précédentes :

$$D(\text{étoiles}) = 1.0039 \quad , \quad D(\text{Lena}) = 0.8509 \quad \text{et} \quad D(\text{Muscle}) = 0.8549 .$$

6. Homogénéité : (Homog)

L'homogénéité de l'image est une donnée qui va nous permettre de savoir si les niveaux de gris présents dans l'image sont très dispersés dans l'image ou s'ils forment des zones de couleur homogène. Pour cela, on va analyser chaque pixel avec ses 8 plus proches voisins. S'il existe une différence de niveau de gris supérieure à un seuil entre le pixel et l'un des ses voisins, alors on incrémente le résultat de 1. On divise le résultat final par le nombre de pixels afin d'obtenir un coefficient entre 0 et 1.

Les résultats obtenus sur les trois images sont :

$$\text{Homog}(\text{étoiles}) = 0.0759 \quad , \quad \text{Homog}(\text{Lena}) = 0.0192 \quad \text{et} \quad \text{Homog}(\text{Muscle}) = 0.01 .$$

Le premier avantage de ces calculs est de pouvoir déterminer approximativement le type d'image à laquelle nous avons affaire. Ainsi, nous pouvons à partir de ces valeurs déterminer que l'image (étoiles) contient peu de détails et des zones très homogènes tandis que l'image (Muscle) contient beaucoup de détails et peu de zones homogènes (donc un grand nombre de contours ou beaucoup de bruit).

7. Choix d'un détecteur :

Il va falloir choisir un détecteur en fonction des caractéristiques de l'image. Pour cela, nous allons tout d'abord effectuer une série de tests sur un banc d'essai d'images de divers types afin de repérer les principaux avantages de chaque détecteur. Le banc d'essai comporte dix images de même taille 277*254.

Mais avant de les soumettre à tous les détecteurs implémentés dans le cadre de notre travail (Robert, *Sobel*, *Prewitt*, *Kirsch*, *Laplacien*, *Marr et Hildreth*, *Canny*), elles seront traitées de plusieurs façons, par l'ajout de bruit et de flou à l'aide de quelques logiciels qu'on a téléchargé du Web.

Nous obtenons donc un tableau récapitulatif des performances des détecteurs suivant différents cas de figure (Tableau IV.1). Les résultats expérimentaux permettent d'émettre les observations suivantes:

- 1) L'inconvénient majeur de Robert et Laplacien est leur extrême sensibilité au bruit du fait de la « petite taille » de Robert et de l'utilisation de la dérivée seconde pour le Laplacien. Par contre Prewitt donne évidemment de meilleurs résultats mais avec une complexité de calcul nettement plus grande.
- 2) Puisque Marr et Hildreth est un Laplacien d'une gaussienne (LoG filter), il permet de supprimer du bruit qui aurait été détecté par le Laplacien seul. Mais il est peu fiable.
- 3) En analysant ces résultats, on constate que certains détecteurs sont plus efficaces quelque soit le cas. Par exemple, dans le cas des images nettes, le détecteur Prewitt est toujours meilleur que Robert, Sobel ou Kirsch. Ceci va en partie simplifier le choix du détecteur par la suite.

4) De même, dans le cas des images bruitées, le détecteur Canny se comporte mieux que celui de Marr et Hildreth et Laplacien et sera donc préféré à ces deux détecteurs.

image	Les caracteristiques						détecteur	Comentaire
	Bruit	Flou	Contraste	Entropie	Niveau de détail	Homoginité		
Lena	0,0116	-2,7025	51,3723	2,1685	0,8509	0,0192	Robert	des manques
							Sobel	meilleur
							Priwitt	épais
							Kirsch	des manques
							Laplacien	non fermé
							Canny	des manques
Marr et hildreth	des manques							
Lenabruité	0,046	-8,2703	67,3479	1,8645	0,8509	0,2277	Robert	bruité
							Sobel	bruité
							Priwitt	bruité
							Kirsch	bruité
							Laplacien	bruité
							Canny	meilleur
Marr et hildreth	bruité							
Lenaflou	0,0099	11,0704	45,2713	2,1537	0,7333	0,0127	Robert	des manques
							Sobel	épais
							Priwitt	meilleur
							Kirsch	des manques
							Laplacien	non fermé
							Canny	des manques
Marr et hildreth	discontinue							
Logo	0,0083	7,8862	54,664	0,9691	0,7333	0,0052	Robert	des manques
							Sobel	non fermé
							Priwitt	meilleur
							Kirsch	des manques
							Laplacien	discontinue
							Canny	non fermé
Marr et hildreth	des manques							
Etoile	0,01933	-0,8489	86,8082	0,8696	1,0039	0,0759	Robert	meilleur
							Sobel	épais
							Priwitt	épais
							Kirsch	épais
							Laplacien	bruité
							Canny	épais
Marr et hildreth	des manques							

							Robert	épais
							Sobel	épais
Etoilebruité	0,1046	-0,057	114,2668	0,8592	1,0039	0,3501	Priwitt	épais
							Kirsch	épais
							Laplacien	bruité
							Canny	meilleur
							Marr et hildreth	bruité
							Robert	des manques
							Sobel	meilleur
Papier	0,0136	0,5597	89,0182	2,0785	1,0039	0,0972	Priwitt	épais
							Kirsch	épais
							Laplacien	non fermé
							Canny	épais
							Marr et hildreth	des manques
							Robert	épais
							Sobel	meilleur
Muscle	0,0108	-3,7794	59,7278	2,2196	0,8549	0,01	Priwitt	épais
							Kirsch	épais
							Laplacien	discontinue
							Canny	des manques
							Marr et hildreth	des manques
							Robert	épais
							Sobel	épais
Domino	0,0153	-3,4673	69,9196	1,874	0,9529	0,0157	Priwitt	épais
							Kirsch	épais
							Laplacien	discontinue
							Canny	meilleur
							Marr et hildreth	non fermé
							Robert	épais
							Sobel	meilleur
Cellules	0,0129	2,9894	57,6429	2,123	1,0039	0,0052	Priwitt	épais
							Kirsch	épais
							Laplacien	des manques
							Canny	non fermé
							Marr et hildreth	interrompu
							Robert	épais
							Sobel	épais
Cellules flou	0,0086	8,0394	50,4765	2,0516	0,945	0,0052	Priwitt	meilleur
							Kirsch	épais
							Laplacien	discontinue
							Canny	non fermé
							Marr et hildreth	discontinue
							Robert	bruité
							Sobel	meilleur
interieur	0,0134	-1,346	78,9779	2,1373	1,0039	0,0525	Priwitt	épais
							Kirsch	épais
							Laplacien	non fermé
							Canny	des manques
							Marr et hildreth	interrompu

Tableau IV.1 : tableau d'analyse.

8. Règles de sélection du détecteur :

D'après les remarques précédentes et en nous référant au tableau IV.1 relatif aux résultats expérimentaux, nous avons établi les règles heuristiques de sélection suivantes :

- Si l'image contient peu de bruit (estimation inférieure à 0.01) et un flou moyen (supérieure à 5) alors nous utilisons le détecteur de Prewitt.
- Si l'image contient beaucoup de bruit (estimation supérieure à 0.02) et une homogénéité supérieure à 0.1 alors nous utilisons le détecteur Canny.
- Si l'image contient du bruit (estimation inférieure ou égale à 0.02) et une homogénéité inférieure ou égale à 0.1 on a trois cas à vérifier :
 - Si le niveau de détail de l'image est supérieur ou égale à 0.9 et strictement inférieur à 1 alors nous utilisons le détecteur Canny.
 - Si le niveau de détail de l'image est supérieur ou égale à 1 et l'entropie de l'image est inférieure à 1 alors nous utilisons le détecteur Robert.
 - Si l'entropie de l'image est supérieure ou égale à 1 alors nous utilisons le détecteur Sobel.

9. Conclusion :

La détection des contours d'une image numérique est un traitement très délicat, car chaque image a un ensemble de caractéristiques différentes par rapport à une autre image tel que le bruit, le flou, l'homogénéité, le niveau de détails, et l'entropie. Il existe un grand nombre de détecteurs et chaque détecteur a des performances différentes suivant les caractéristiques de l'image.

Certains détecteurs sont très efficaces pour les images nettes mais sensibles au bruit, alors que d'autres détecteurs sont très adaptés aussi bien aux images nettes qu'aux images bruitées mais sensibles au flou.

Chapitre V:

Présentation du logiciel

Nous présentons dans ce chapitre la description détaillée de l'application (les différentes composantes de l'application, ainsi que les diverses techniques implémentées).

1. Introduction :

Le but de notre projet est de réaliser un logiciel qui permet la sélection automatique d'un détecteur de contours, et aussi d'effectuer des opérations de prétraitement tel que filtrage et binarisation...

2. Présentation du logiciel :

Ce logiciel a été réalisé dans l'environnement WINDOWS XP, avec l'utilisation de *Delphi 7* comme un langage de programmation, car en plus de sa facilité de programmation, il offre une interface conviviale à notre logiciel.

Notre application utilise tous les moyens offerts par WINDOWS XP à savoir :

- Le système de menus déroulant donnant un accès simple à toutes les fonctions du système.
- La barre de boutons à manipulation pratique permettant un accès rapide aux principales fonctions du système.

Notre logiciel détient le menu suivant :

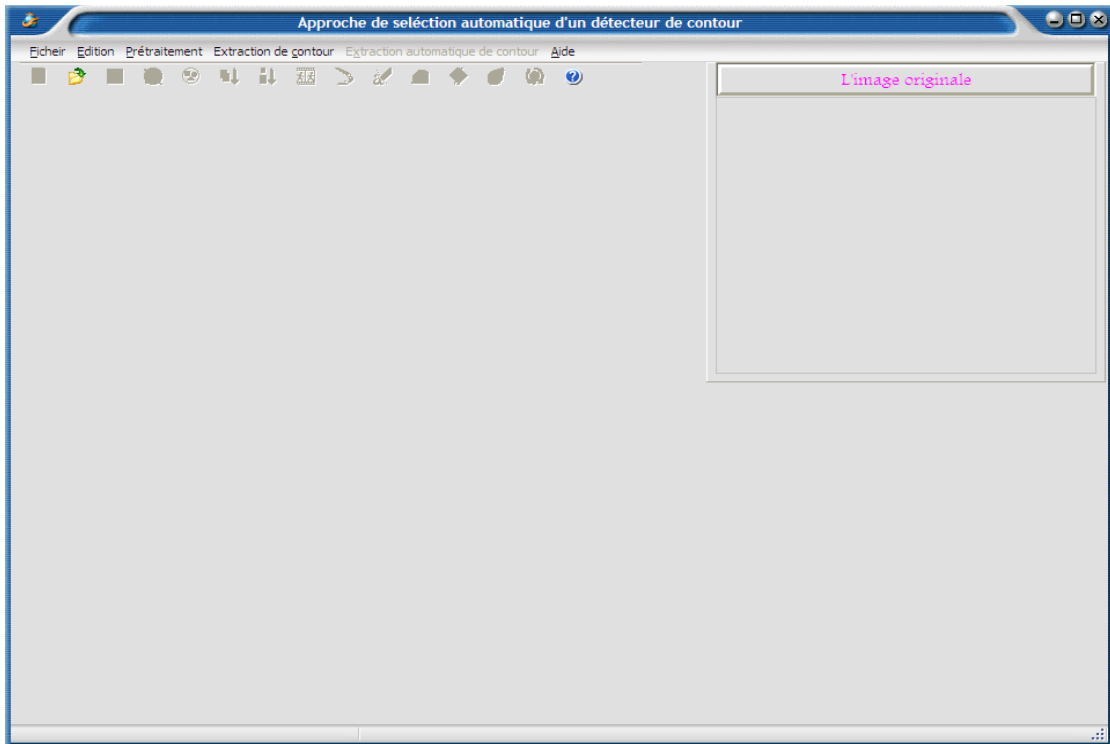


Figure V.1 : La fenêtre principale du logiciel.

3. Menus du logiciel et barre d'icône :

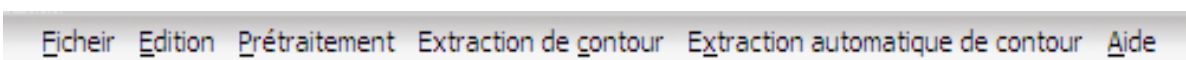


Figure V.2 : Le menu du logiciel.

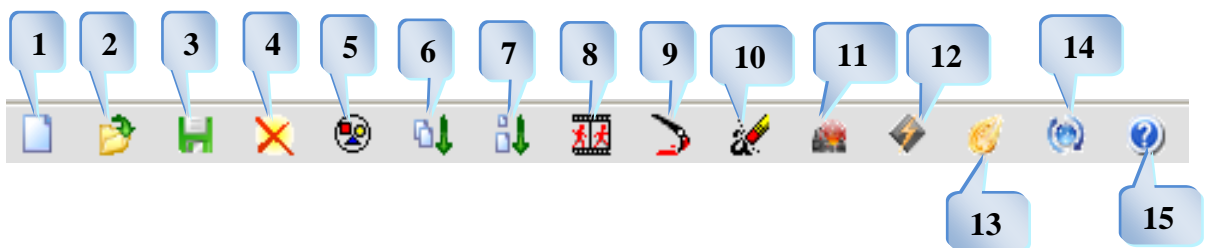


Figure V.3 : La barre d'icône.

- 1 : Nouveau, permet de réinitialiser le logiciel.
- 2 : Ouvrir, permet l'ouverture d'une image.
- 3 : Enregistrer, permet d'enregistrer l'image traitée (résultante).
- 4 : Supprimer, permet de supprimer l'image traitée.
- 5 : Propriété de l'image.
- 6 : Filtrage linéaire.
- 7 : Filtrage non linéaire.
- 8 : Binarisation.
- 9 : Gardien.
- 10 : Laplacien.
- 11 : Marr et Hildreth.
- 12 : Canny.
- 13 : Extraction automatique de contours.
- 14 : Inverser les images.
- 15 : Aide.

3.1. Menu Fichier :

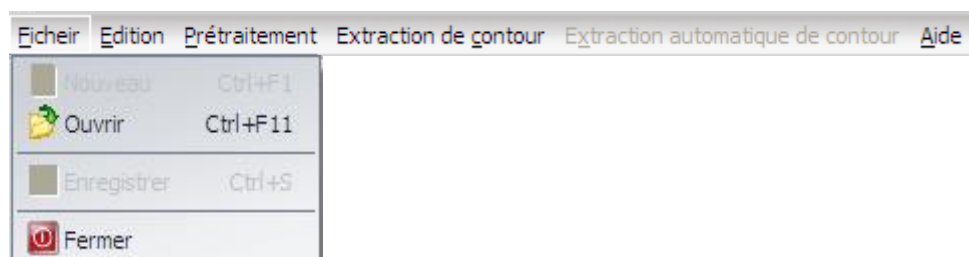


Figure V.4 : Le menu Fichier.

3.1.1. Nouveau : réinitialisation du traitement.

3.1.2. Ouvrir : permet d'ouvrir un fichier image de type BMP.

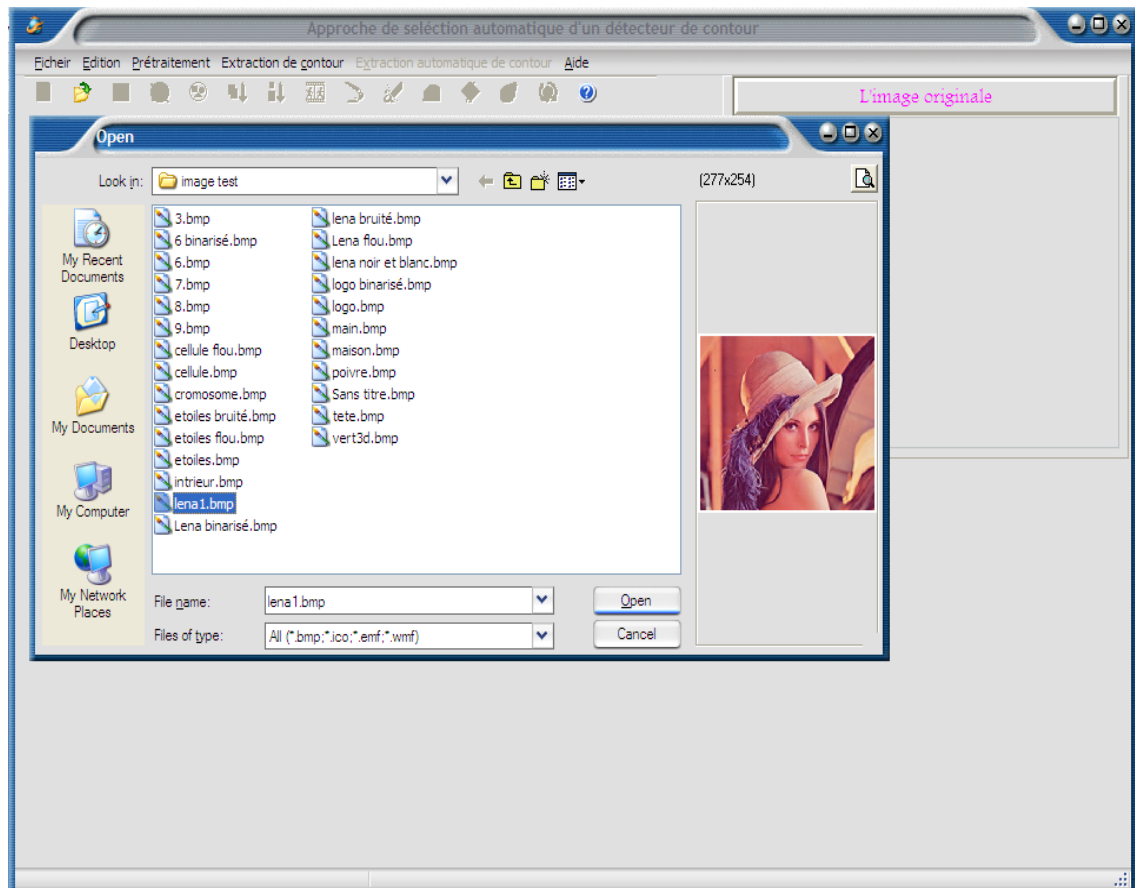


Figure V.5 : Ouvrir une image.

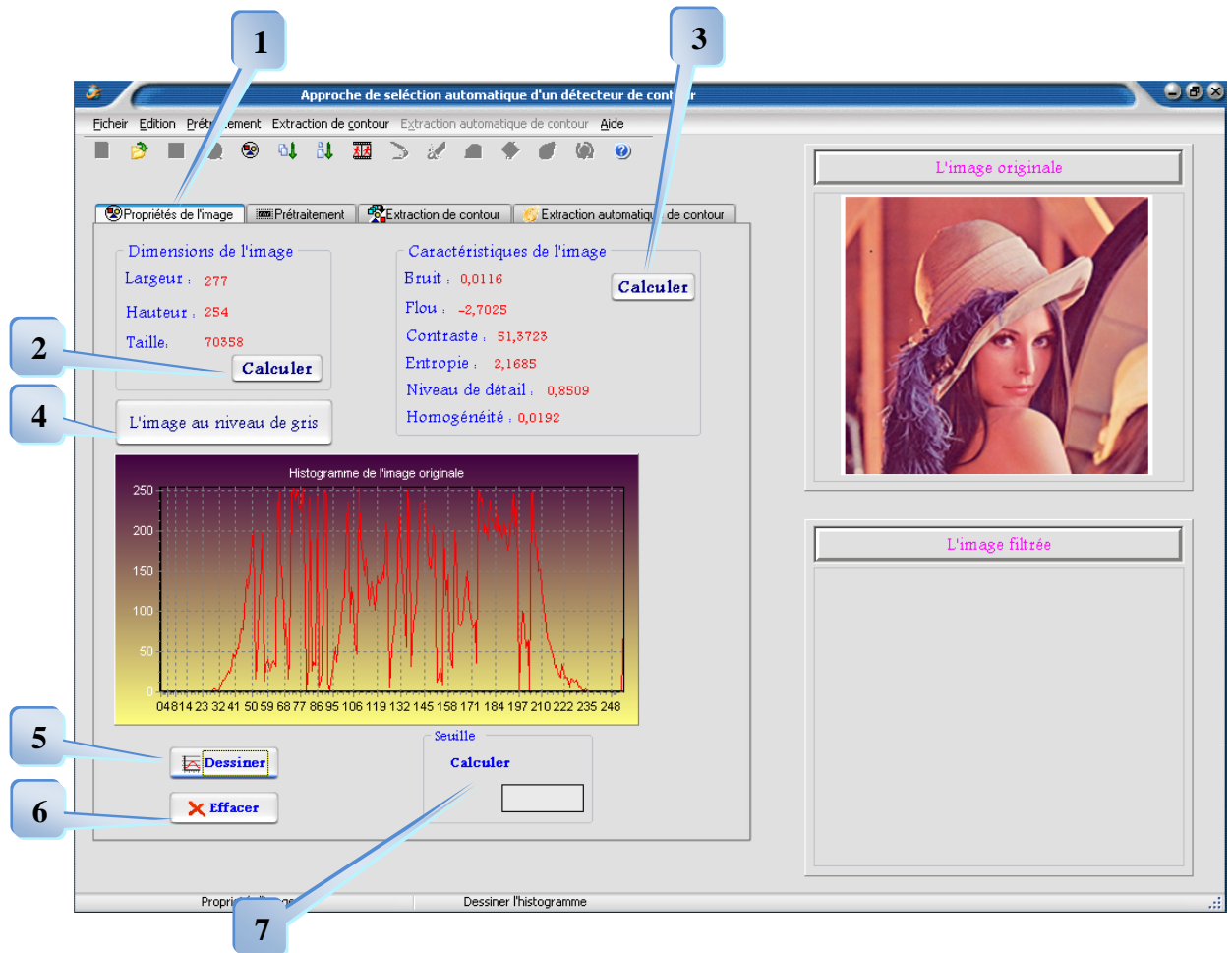


Figure V.6 : Après l'ouverture de l'image.

- 1 : Propriété de l'image originale.
- 2 : permet de calculer les dimensions de l'image originale.
- 3 : permet de calculer les caractéristiques de l'image originale.
- 4 : permet de traiter l'image originale au niveau de gris.
- 5 : permet de dessiner l'histogramme de l'image originale.
- 6 : permet d'effacer l'histogramme de l'image originale.
- 7 : permet de calculer le seuil à partir de l'histogramme.

3.1.3. Enregistrer : enregistrer l'image sous un notre nom dans la fenêtre active.

3.1.4. Fermer : permet de quitter l'application.

3.2. Menu Edition :

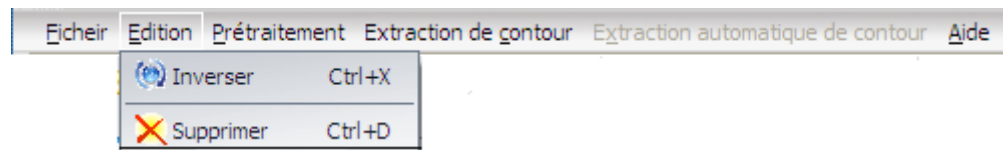


Figure V.7 : Le menu Edition.

3.2.1. Inverser : permet d'inverser l'image originale avec l'image traitée.

3.2.2. Supprimer : permet de supprimer l'image traitée.

3.3. Menu Prétraitement :

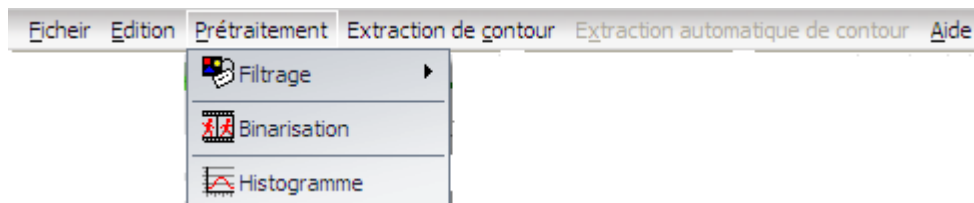


Figure V.8 : Le menu Prétraitement.

3.3.1. Filtrage : il y a deux types de filtrages :

a) **Filtrage linéaire:** il y a quatre types de filtrage linéaire:

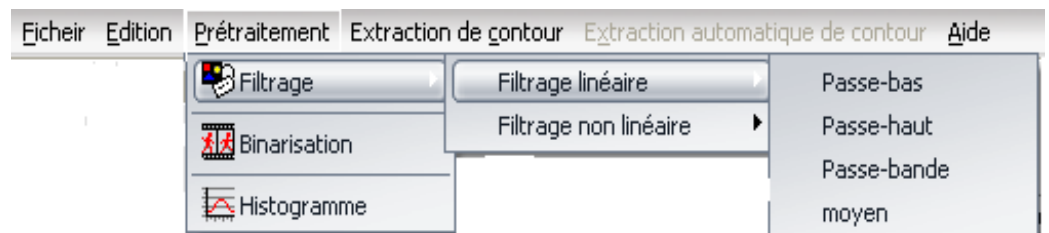


Figure V.9 : Sous menu filtrage linéaire.

a.1) Filtre passe-bas : permet de filtrer l'image avec le filtre passe-bas.

a.2) Filtre passe-haut : permet de filtrer l'image avec le filtre passe-haut.

a.3) Filtre passe bande : permet de filtrer l'image avec le filtre passe bande.

a.4) Filtre moyen: permet de filtrer l'image avec le filtre moyen.

b) Filtrage non linéaire : il y a quatre types de filtrage non linéaire:

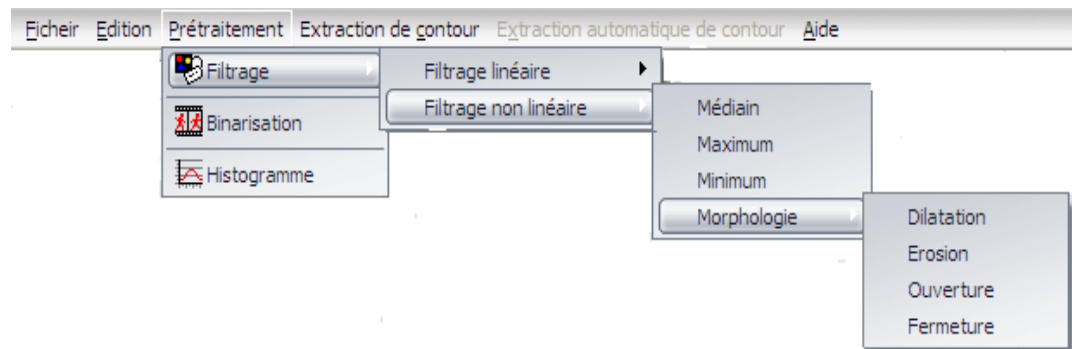


Figure V.10 : Sous menu filtrage non linéaire.

b.1) Filtre médian: permet de filtrer l'image avec le filtre médian.

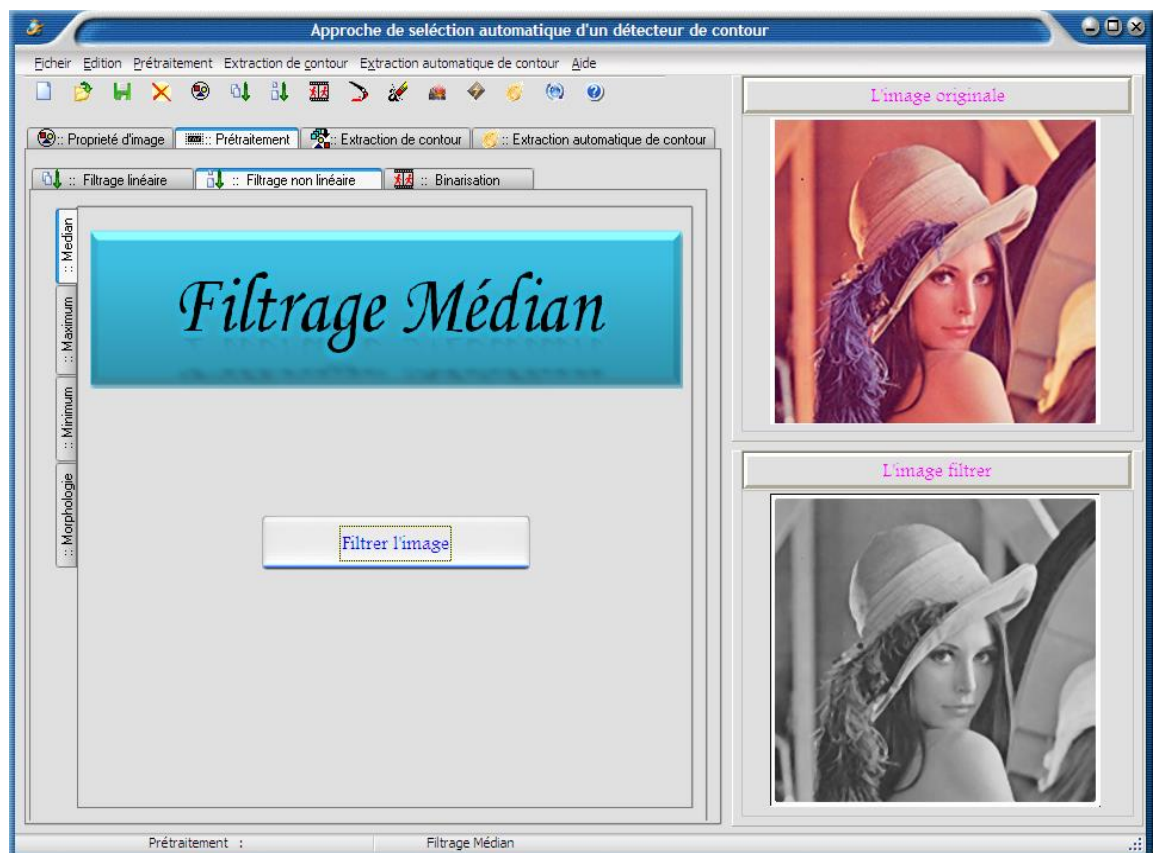


Figure V.11 : Filtrage médian.

b.2) Filtre maximum: permet de filtrer l'image avec le filtre maximum.

b.3) Filtre minimum: permet de filtrer l'image avec le filtre minimum.

b.4) Filtre morphologie:

- ❖ **Dilatation** : permet de dilater l'image.
- ❖ **Erosion** : permet de d'éroder l'image.
- ❖ **Ouverture** : permet de calculer l'ouverture morphologique de l'image.
- ❖ **Fermeture** : permet de calculer la fermeture morphologique de l'image.

3.3.2. Binarisation : permet de binariser l'image.

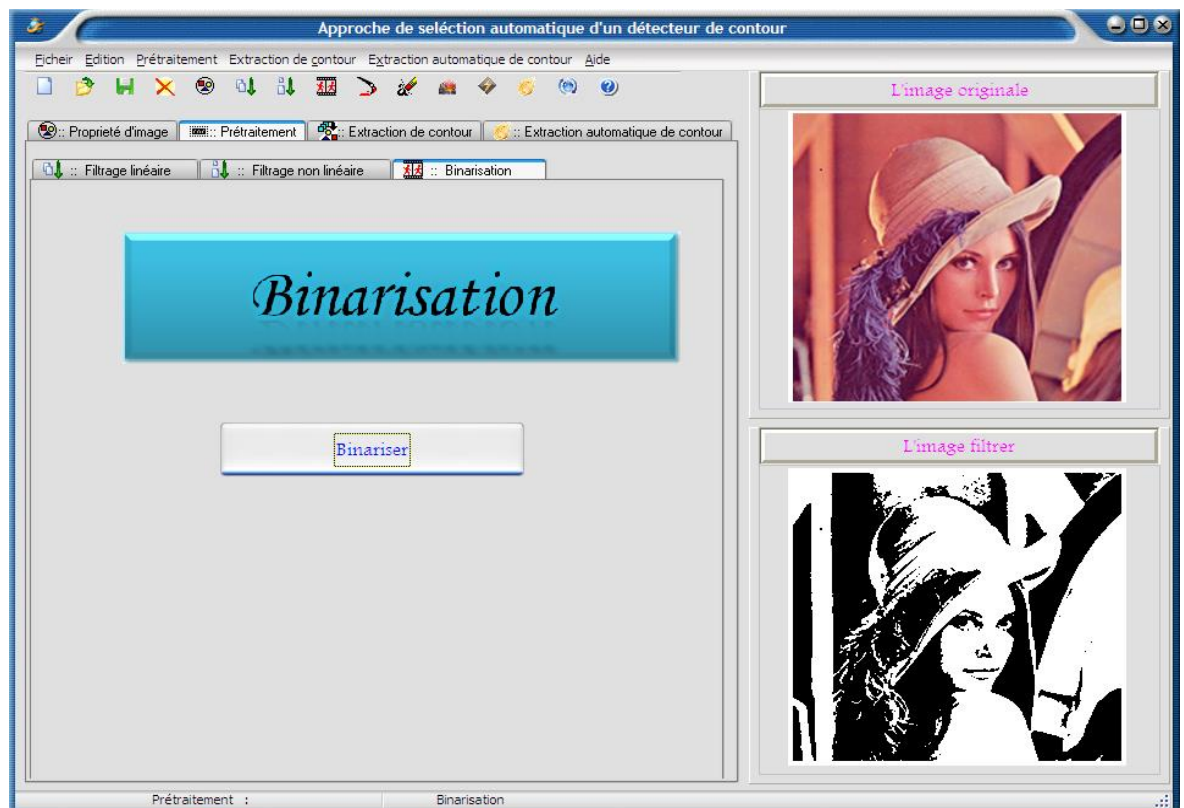


Figure V.12 : Binarisation.

3.3.3. Histogramme : trace l'histogramme de l'image.

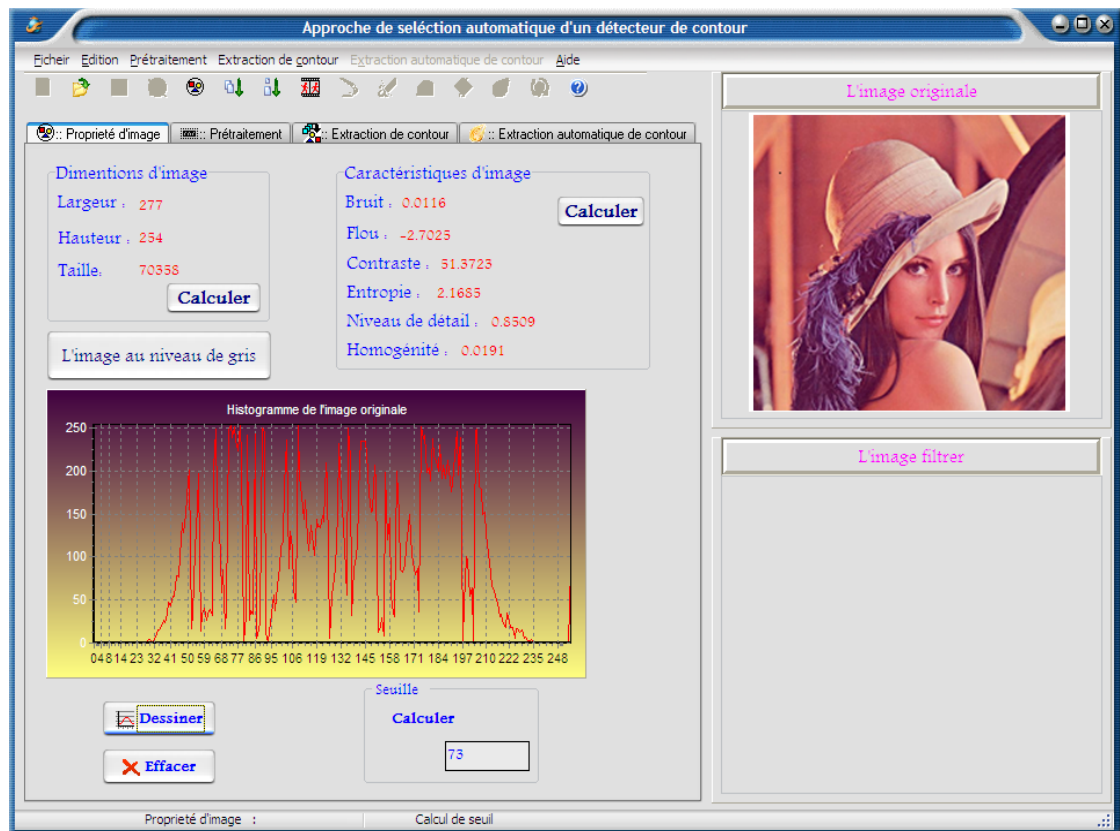


Figure V.13 : Histogramme.

3.4. Menu Extraction de contours :

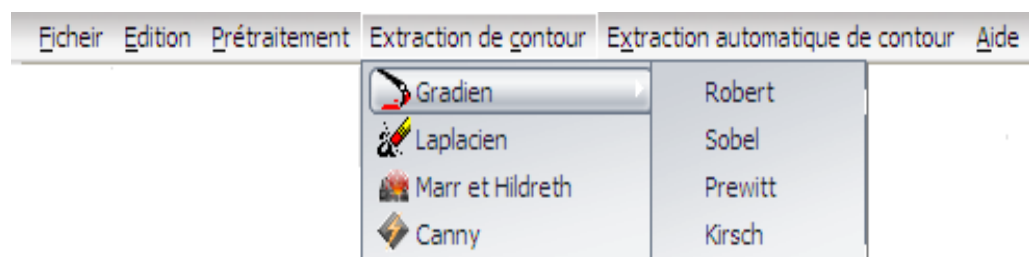


Figure V.14 : Le menu Extraction de contours.

3.4.1. Gradient : contient quatre opérateurs :

- a) **Robert** : permet la détection de contours de l'image filtrée par l'opérateur de Robert.

- b) **Sobel** : permet la détection de contours de l'image filtrée par l'opérateur de Sobel.

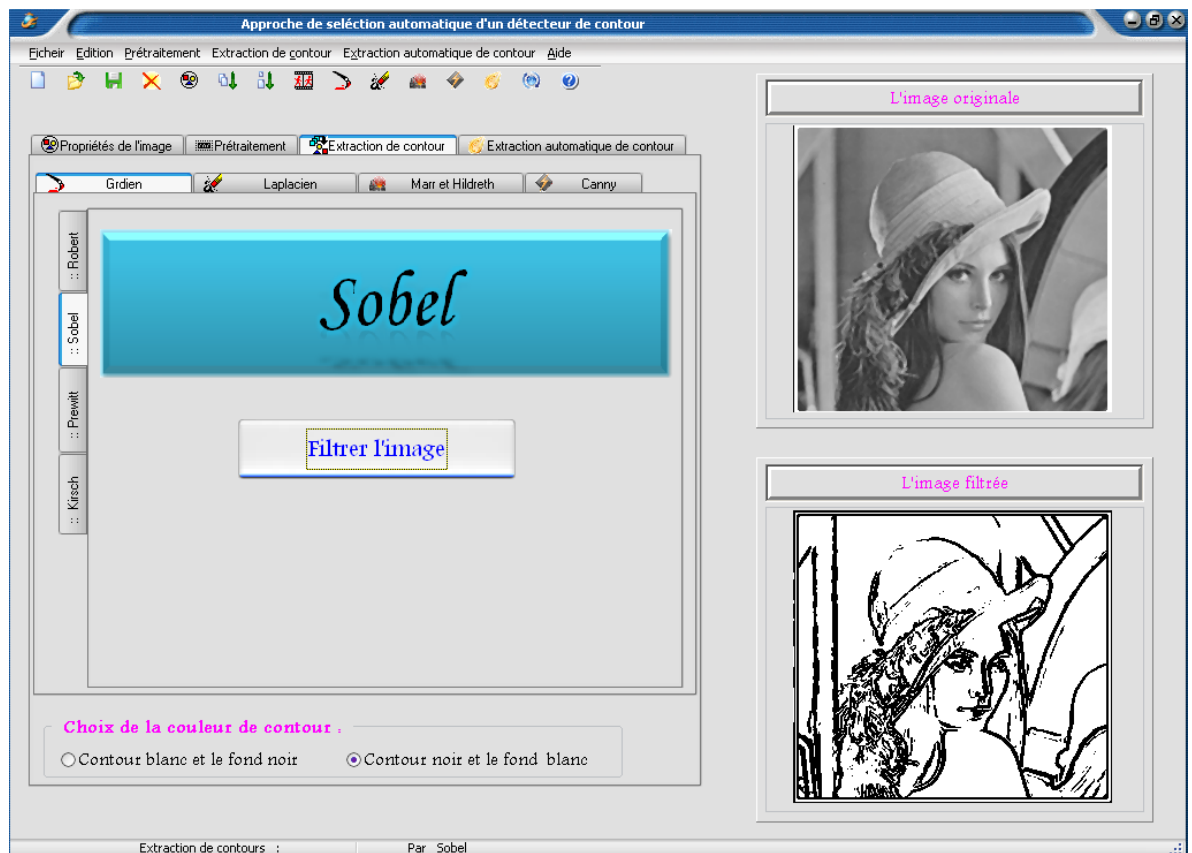


Figure V.15 : Extraction de contours par le détecteur Sobel.

- c) **Prewitt** : permet la détection de contours de l'image filtrée par l'opérateur de Prewitt.
- d) **Kirsch** : permet la détection de contours de l'image filtrée par l'opérateur de Kirsch.
- 3.4.2. Laplacien** : permet la détection de contours de l'image filtrée par l'opérateur de Laplacien.
- 3.4.3. Marr et Hildreth** : permet la détection de contours de l'image filtrée par l'opérateur de Marr et Hildreth.
- 3.4.4. Canny** : permet la détection de contours de l'image filtrée par l'opérateur de Canny.

3.5. Menu Extraction automatique de contours :

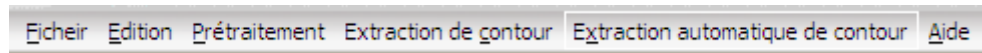


Figure V.16 : Le menu Extraction automatique de contours.

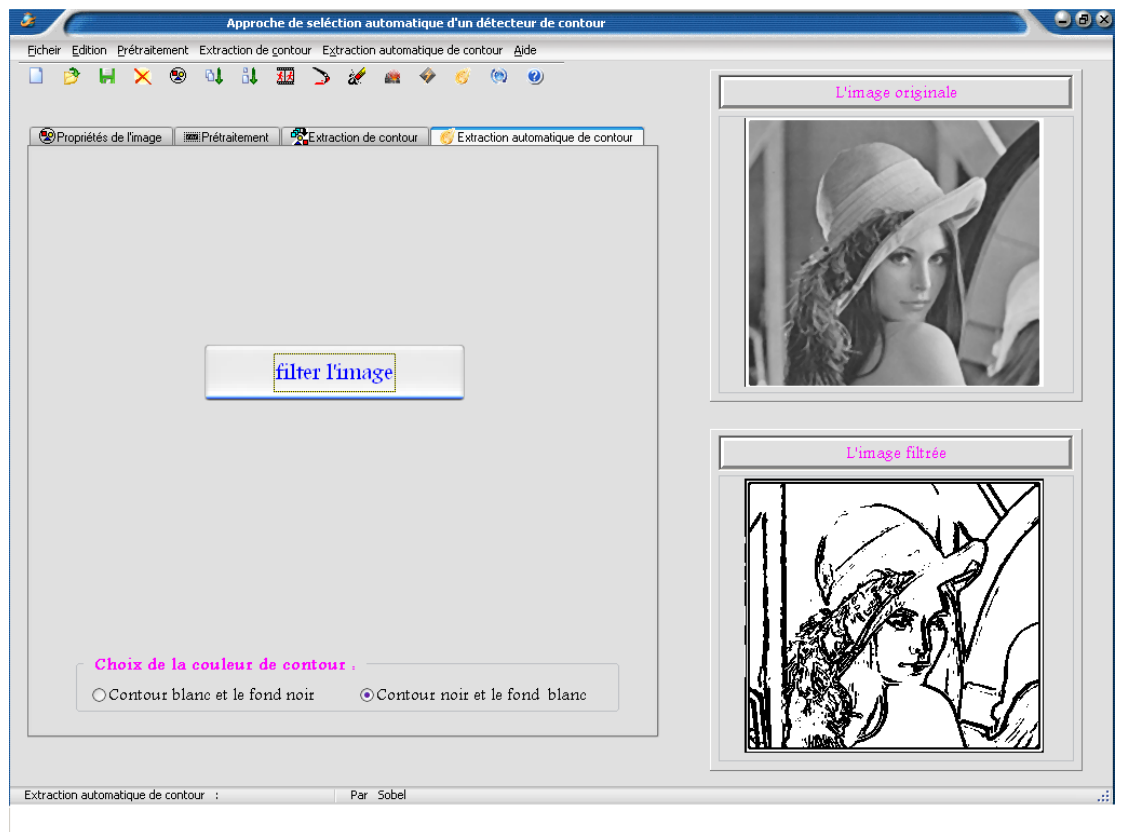


Figure V.17 : Extraction automatique de contours.

3.6. Menu Aide:

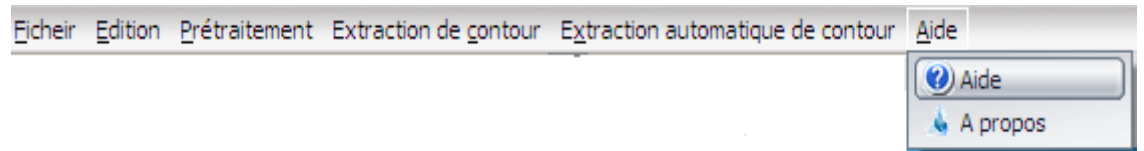


Figure V.18 : Le menu Aide.

4. Conclusion :

Notre logiciel comporte tous les traitements et algorithmes étudiés et présentés dans ce mémoire, dont principalement : le prétraitement (filtrage, binarisation, histogramme...), l'extraction de contours avec tous les détecteurs présentés, sans oublier aussi les fonctions de bases de gestions des fichiers d'images, tel que les différentes commandes attachées aux menu du fichier et barre d'outils, afin d'avoir une interface simple et souple à l'utilisation.

Conclusion générale

Conclusion générale

A l'issue de ce travail, nous avons réussi à développer une application capable de choisir automatiquement les détecteurs de contours les plus adéquats et les plus efficaces pour les caractéristiques de chaque image. Les caractéristiques utilisées sont le bruit blanc, le flou Gaussien, le niveau de détails, l'homogénéité et l'entropie. Cependant, il est possible d'améliorer encore les résultats en étudiant des caractéristiques plus spécifiques ou en développant de meilleurs détecteurs de contours ou de meilleurs estimateurs.

Nous pourrions aussi envisager d'utiliser certaines techniques spécifiques et récentes pour traiter les images avant de détecter les contours, comme par exemple effectuer un débruitage à l'aide de l'outil Contourlet. Cependant, ces techniques nécessitent un travail à part entière et n'ont donc pas pu être traitées dans ce mémoire.

Finalement, notre travail présente des avantages certains en détection de contours mais les possibilités d'extension sont encore très vastes, que ce soit au niveau des caractéristiques ou au niveau des détecteurs de contours.

BIBLIOGRAPHIE



[3] Djamila TARFAYA et Mebarka DIB, système de reconnaissance des empreintes digital. Mémoire d'ingénieur de l'université de Laghouat, 2006.

[8] Guesdon VINCENT, détection efficace de contours d'images. Mémoire présenté comme exigence partielle de la maîtrise en informatique de l'université du québec en outaouais, 12 octobre 2004.

[4] Rachid DERICHE, Techniques d'extraction de contours. INRIA Sophia-Antipolis, 2004.

[9] Raphaele BALTER, Construction d'un maillage 3D évolutif et scalable pour le codage Vidéo. Thèse de docteur de l'université de rennes 1, 27 mai 2005.

WEBOGRAPHIE



[1] <http://www.wikipédia.com>

[2] http://duson.club.fr/Telechargt/f21_def_resol_pf.

[4] <http://tecfa.unige.ch/staf/staf-k/benetos/staf13/per1/tache5/resolution.html>

[6] <http://www.cours.photophiles.com/index.php?option=com.contient&task=view>

[7] <http://etudiant.univ-mlv.fr/sblanche/imgNum.html>

[5] <http://www.tsi.enst.fr/sti/enseignement/ressources/mti/bruit/index.html#mozTocId307043>