

Chapitre

03**Généralités sur le traitement d'images****Introduction**

Le traitement d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années.

On désigne par *traitement d'images numériques* l'ensemble des techniques permettant de modifier une image numérique afin d'améliorer ou d'en extraire des informations.

De ce fait, le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dans le but de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes.

Dans ce chapitre, nous abordons les notions de base nécessaires à la compréhension des techniques de traitement d'images. Ensuite, nous allons donner un aperçu sur les différents techniques connues dans ce domaine.

3.1 Définition de l' image

Une image est plutôt difficile à décrire d'une façon générale. Une image est une représentation du monde. En traitement d'image, la majorité du temps, on considère qu'il s'agit d'une fonction mathématique de $R \times R$ dans R où le couplet d'entrée est considéré comme une position spatiale, le singleton de sortie comme l'intensité (couleur ou niveaux de gris) du phénomène physique. Il arrive cependant que l'image soit dite "3D" donc la fonction est de $R \times R \times R$ dans R . Les images couleurs peuvent être représentées soit par trois images représentant les trois couleurs fondamentales, soit par une image de $R \times R$ dans $R \times R \times R$ [wikipedia 04].

L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter[chakib 99].

3.2 Acquisition d'une image

L'acquisition d'images constitue un des maillons essentiels de toute chaîne de conception et de production d'images. Pour pouvoir manipuler une image sur un système informatique, il est avant tout nécessaire de lui faire subir une transformation qui la rendra lisible et manipulable par ce système. Le passage de cet objet externe (l'image d'origine) à sa représentation interne (dans l'unité de traitement) se fait grâce à une procédure de numérisation. Ces systèmes de saisie, dénommés optiques, peuvent être classés en deux catégories principales [chakib 99] [Bénallal 03]:

- les caméras numériques ,
- et les scanners.

A ce niveau, notons que le principe utilisé par le scanner est de plus en plus adapté aux domaines professionnels utilisant le traitement de l'image comme la télédétection, les arts graphiques, la médecine, etc. Le développement technologique a permis l'apparition de nouveaux périphériques d'acquisition appelés cartes d'acquisition, qui fonctionnent à l'instar des caméras vidéo, grâce à un capteur *C.C.D.* (Charge Coupled Device). La carte d'acquisition reçoit les images de la camera, de la T.V. ou du scanner afin de les convertir en

informations binaires qui seront stockées dans un fichier[chakib 99]. L'une des caractéristiques intéressantes de ces capteurs est la taille du plus petit élément (*pixel*), mais aussi l'intercorrélation de deux éléments voisins : plus cette intercorrélation est faible, meilleure est l'image[wikipedia 04].

3.3 Caractéristiques d'une image numérique

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants:

3.3.1 Dimension

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image[chakib 99].

3.3.2 Résolution

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour désigner le nombre total de pixels affichables horizontalement ou verticalement sur un moniteur; plus grand est ce nombre, meilleure est la résolution [chakib 99].

3.3.3 Bruit

Un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur [chakib 99].

3.3.4 Histogramme

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (couleur) et de voir entre quelles bornes est répartie la majorité des niveaux de gris (couleur) dans le cas d'une image trop claire ou d'une image trop foncée.

Il peut être utilisé pour améliorer la qualité d'une image (Rehaussement d'image) en introduisant quelques modifications, pour pouvoir extraire les informations utiles de celle-ci.

Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents, ou encore pour mesurer certaines propriétés sur une image, on modifie souvent l'histogramme correspondant [chakib 99] [Pillou 04].

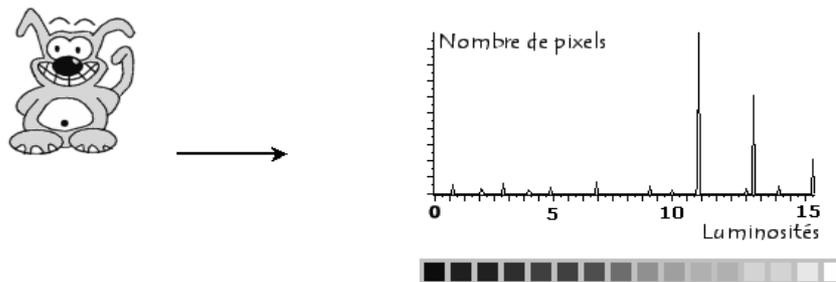


Figure3.1 Histogramme et palette associés à une image [Pillou 04]

3.3.5 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, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. Une bonne luminance se caractérise par [chakib 99] [Nicolas 01] :

- Des images lumineuses (brillantes);
- Un bon contraste : il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir; ces images entraînent des pertes de détails dans les zones sombres ou lumineuses.
- L'absence de parasites.

3.3.6 Contraste

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images. Si L_1 et L_2 sont les degrés de luminosité respectivement de deux zones voisines A_1 et A_2 d'une image, le contraste C est défini par le rapport [chakib 99] :

$$C = \frac{L_1 - L_2}{L_1 + L_2}$$

3.3.7 Images à niveaux de gris

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255. Chaque pixel n'est donc plus représenté par un bit, mais par un octet. Pour cela, il faut que le matériel utilisé pour afficher l'image soit capable de produire les différents niveaux de gris correspondant.

Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la "couleur" de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux [chakib 99].

3.3.8 Images en couleurs

Même s'il est parfois utile de pouvoir représenter des images en noir et blanc, les applications multimédias utilisent le plus souvent des images en couleurs. La représentation des couleurs s'effectue de la même manière que les images monochromes avec cependant quelques particularités. En effet, il faut tout d'abord choisir un modèle de représentation. On peut représenter les couleurs à l'aide de leurs composantes primaires. Les systèmes émettant de la lumière (écrans d'ordinateurs,...) sont basés sur le principe de la synthèse additive : les couleurs sont composées d'un mélange de rouge, vert et bleu (modèle R.V.B.) [chakib 99].

3.4 Système de traitement d'images

Un système de traitement numérique d'images est composé de :

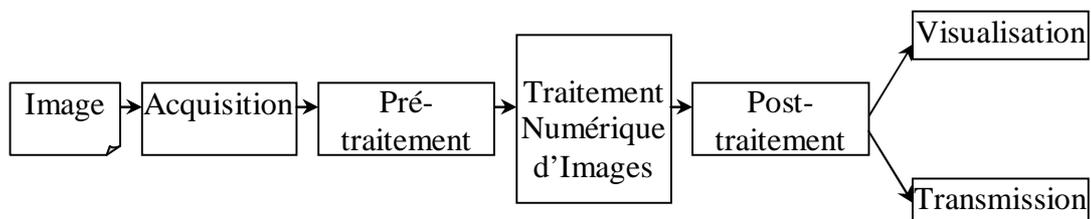


Figure 3.2 Schéma d'un système de traitement d'images [chakib 99]

3.5 Pré-traitement d'images

Cette phase a lieu juste après l'acquisition des images et a pour objectif d'améliorer la qualité de l'image en vue de sa segmentation.

Le temps de traitement (temps CPU) est très important. C'est un facteur décisif, il doit être le plus petit possible. Ceci implique que les opérateurs doivent être locaux, c'est-à-dire qu'ils doivent intervenir sur un nombre limité de pixels et essentiellement sur des pixels dans le voisinage proche du pixel courant.

Les techniques de pré-traitements les plus courantes qu'on va présenter sont [Bensrhair02,a] [Cocquerez 95,+]:

- ↳ La modification d'histogramme,
- ↳ La réduction du bruit par filtrage,

3.5.1 Modification d'histogramme

On cherche à améliorer l'image en appliquant une transformation ponctuelle d'intensité; à tout pixel $a_s(x,y)$ on associe une intensité par la transformation T croissante de façon à conserver des contrastes entre régions (claire sur fond sombre, apparaît plus claire après transformation). Le pixel après transformation est le pixel $a'_s(x,y)$:

$$a'_s(x,y) = T(a_s(x,y))$$

Dans ce qui suit, on va présenter deux types de modification d'histogramme:

- Expansion de dynamique
- Egalisation d'histogramme

Expansion de dynamique

Cette méthode consiste à utiliser au mieux l'échelle des niveaux de gris disponible sur le système d'acquisition d'images.

Soit $A[i,j]$ l'image de départ, $A'[i,j]$ image après transformation et $[a_0, a_1]$ l'intervalle des intensités présentes dans l'image et soit $[a_{\min}, a_{\max}]$ l'intervalle disponible.

L'expansion de dynamique correspond à la transformation: $a'_s = \alpha + \beta a_s$ telle que:

$$\forall a \in [a_0, a_1] \quad a_s \xrightarrow{T} a'_s \in [a_{\min}, a_{\max}]$$

$$\alpha = \frac{a_{\min} \cdot a_1 - a_{\max} \cdot a_0}{a_1 - a_0}, \quad \beta = \frac{a_{\max} - a_{\min}}{a_1 - a_0}$$

Egalisation d'histogramme

Cette transformation consiste à rendre le plus plat possible l'histogramme de niveaux de gris de l'image. L'image $A[i,j]$ est considérée comme un ensemble de réalisation d'une variable aléatoire A , admettant une densité de probabilité $f(a)$ et une fonction de répartition $F(a)$, f représente l'histogramme normalisé de $A[i,j]$.

On cherche une transformation T continue dérivable par morceaux et strictement croissante telle que la variable aléatoire $B=T(A)$ soit uniformément répartie dans l'intervalle $[b_{min}, b_{max}]$.

Soit $g(b)$ la densité de probabilité de B , et T' la transformation dérivée de T :

$$g(b) = \begin{cases} f(a) \frac{1}{T'(a)} = \frac{1}{b_{max} - b_{min}} & \text{si } b_{min} < b < b_{max} \\ 0 & \text{ailleurs} \end{cases}$$

$$T'(a) = (b_{max} - b_{min}) \cdot f(a) \quad , \quad b = T(a) \quad , \quad a \in [a_{min}, a_{max}]$$

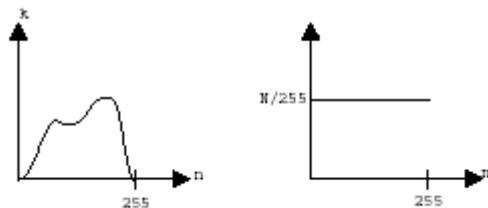


Figure3.3 Egalisation d'histogramme [Bensrhair 02,a]

3.5.2 Réduction du bruit par filtrage

Le bruit peut provenir de différentes causes :

- Environnement lors de l'acquisition,
- Qualité du capteur,
- Qualité de l'échantillonnage.

En général, le bruit d'images est considéré comme un champ aléatoire caractérisé par sa densité de probabilité f et sa fonction de répartition F . La nature plus ou moins impulsionnelle du bruit peut être décrite à l'aide de densité de probabilités de la forme [Cocquerez 95]:

$$f(a) = C \cdot \exp(-K|a|^\alpha)$$

Plusieurs filtres sont utilisés pour la réduction de bruit, ils sont divisés en deux catégories :

- Filtres linéaires stationnaires ,
- Filtres non linéaires stationnaires.

La première catégorie comprend tous les opérateurs pouvant exprimer leur résultat comme une combinaison linéaire des niveaux de gris d'un voisinage de l'image. Ces filtres possèdent des caractéristiques spectrales, on parle ainsi de filtre passe-bas (l'image devient floue) ou de filtre passe-haut (les contours ressortent).

Voici quelques exemples de filtres réducteurs de bruit :

☞ *Filtre moyenneur*

☞ *Filtre gaussien*

☞ *Filtre exponentiel de Shen*

Le principale inconvénient des filtres linéaire est que la réduction de bruit s'accompagne d'un étalement des transitions entre régions. Ce problème peut être surmonté par l'utilisation des filtres non linéaires.

Le domaine de filtrage non linéaire est extrêmement vaste. Les opérateurs les plus courants en pré-traitement d'images sont :

- Les filtres d'ordre,
- Les filtres homomorphique,
- Les filtres morphologique.

3.6 Extraction de contours [Cocquerez 95,+] [Monga 95,+] [UFRIMA 03][Girard 99,+]

La détection de contour est une étape préliminaire à de nombreuses applications de l'analyse d'images. Les contours constituent en effet des indices riches, au même titre que les points d'intérêts, pour toute interprétation ultérieure de l'image. Les contours dans une image proviennent des :

- discontinuités de la fonction de réflectance (texture, ombre),
- discontinuités de profondeur (bords de l'objet),

et sont caractérisés par des discontinuités de la fonction d'intensité dans les images.

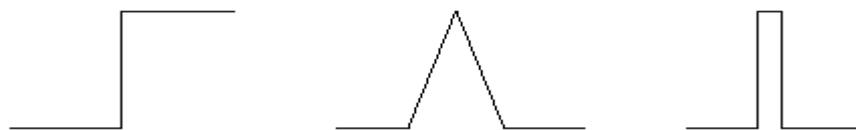


Figure 3.4 Différents types de contours : marche, toit et pointe [Bensrhair 02,b]

Le principe classique de la détection de contours repose sur l'étude des dérivées de la fonction d'intensité dans l'image : les extréma locaux du gradient de la fonction d'intensité et les passages par zéro du laplacien.

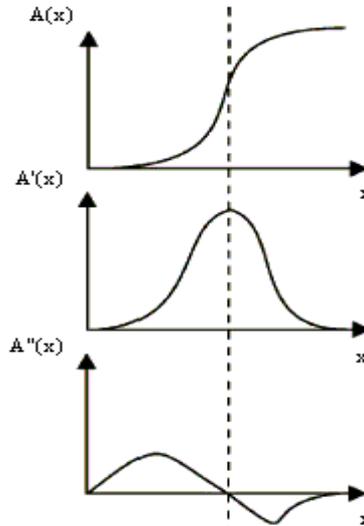


Figure 3.5 Cas bidimensionnel [Bensrhair 02,b]

3.6.1 Approche Gradient [Bensrhair 02,b]

La recherche de la transition peut être réalisée en recherchant les maxima locaux du gradient. Dans le cas bidimensionnel, le vecteur gradient est défini au point $M(x, y)$ par :

$$\vec{\nabla}A(x, y) = \begin{pmatrix} \frac{\partial A}{\partial x} \\ \frac{\partial A}{\partial y} \end{pmatrix}$$

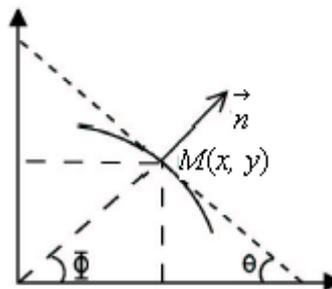


Figure 3.6 Orientation du contour [Bensrhair 02,b]

$\vec{\nabla}A(x, y)$ est normal à la courbe de niveau donnée par $A(x, y) = cste$ qui passe par M .

Au point $M(x, y)$ l'orientation Φ du gradient est donnée par : $\Phi = \arctan \left(\frac{\frac{\partial A}{\partial y}}{\frac{\partial A}{\partial x}} \right)$

et le vecteur unitaire \vec{n} normal au contour : $\vec{n} = \begin{pmatrix} \cos \Phi \\ \sin \Phi \end{pmatrix}$

L'orientation du contour a pour expression : $\theta = \frac{\pi}{2} - \Phi$

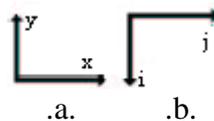


Figure 3.7 Orientation des axes. a) cas continu. b) cas discret

Pour une image numérique, on a :

$$\frac{\partial A}{\partial y} = \frac{\Delta A}{\Delta i} = A_i[i, j] = A_i[i+1, j] - A_i[i, j]$$

$$\frac{\partial A}{\partial x} = \frac{\Delta A}{\Delta j} = A_j[i, j] = A_j[i, j+1] - A_j[i, j]$$

La norme du gradient est :

$$|\nabla A[i, j]| = \sqrt{A_j^2[i, j] + A_i^2[i, j]}$$

ou :

$$|\nabla A[i, j]| = \max\{A_j[i, j], A_i[i, j]\}$$

ou bien

$$|\nabla A[i, j]| = |A_j[i, j]| + |A_i[i, j]|$$

Opérateurs Prewitt et Sobel

$$\begin{aligned}
 A_j[i, j] &= h_j * A[i, j] & h_j &= \begin{pmatrix} 1 & 0 & -1 \\ c & 0 & -c \\ 1 & 0 & -1 \end{pmatrix} \\
 A_i[i, j] &= h_i * A[i, j] & h_i &= \begin{pmatrix} 1 & c & 1 \\ 0 & 0 & 0 \\ -1 & -c & -1 \end{pmatrix}
 \end{aligned}$$

Les matrices h_j et h_i sont appelées masques. Elles sont les noyaux de convolution de filtres à réponse impulsionnelle finie. Pour Prewitt, on prend $c=1$. Pour Sobel, on prend $c=2$.

Opérateur de Kirsh

C'est un opérateur à 8 masques correspondant chacun à une direction préférentielle et obtenue par rotation de $\frac{\pi}{4}$ de l'opérateur de base h_0 . (la numérotation des masques est faite dans l'ordre des directions de Freeman en 8-connextité).

$$h_0 = \begin{pmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{pmatrix}$$

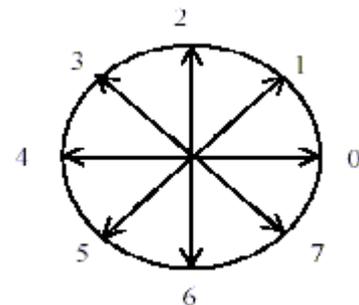


Figure 3.8 Les directions de Freeman d'après [Cocquerez 95]

3.6.2 Approche Laplacien

La recherche des contours dans l'image $A(x, y)$ peut aussi être réalisée en recherchant les passages par zéro du laplacien. On dispose donc, de la valeur du laplacien en chaque point de l'image soit la fonction Δ .

$$\Delta A(x, y) = \frac{\partial^2 A}{\partial x^2} + \frac{\partial^2 A}{\partial y^2}$$

On considère que les points de contours sont localisés aux passages par zéro de laplacien. Si le calcul du laplacien était exact il suffirait de sélectionner les points M tels que $\Delta(M)=0$.

3.6.3 Approche par filtrage optimal : Approche de Canny

Filtre optimal de réponse impulsionnelle $h(x)$ qui satisfait les trois contraintes suivantes :

- ↪ Bonne détection
- ↪ Bonne localisation
- ↪ Réponse unique

Soit $A(x)$ un signal mono-dimensionnel représentant un saut d'amplitude U_0 noyé dans un bruit blanc stationnaire $N(x)$ de moyenne nulle et de densité spectrale N_0^2 .

$$A(x) = U_0 U(x) + N(x)$$

Le signal de sortie est :

$$C(x) = A * h(x) = \int_{-\infty}^{\infty} A(t)h(x-t)dt$$

Bonne détection : Le rapport RSB (Rapport Signal/Bruit) doit être maximisé.

$$RSB = \frac{U_0 \int_0^{\infty} h(x-t)dt}{\left[E \left(\left| \int_{-\infty}^{\infty} N(t)h(x-t)dt \right|^2 \right) \right]^{1/2}} = \frac{U_0 \int_0^{\infty} h(x-t)dt}{N_0 \left[\int_{-\infty}^{\infty} h^2(t)dt \right]^{1/2}}$$

Bonne localisation : Les points détectés doivent être aussi près que possible du centre du contour véritable (à maximiser)

$$\frac{U_0}{N_0} \frac{|h'(0)|}{\left[\int_{-\infty}^{\infty} h'^2(t)dt \right]^{1/2}}$$

Unicité de la réponse : On utilise le critère de Canny. On veut minimiser la densité $d0$ des passages par 0 de la réponse du bruit.

Critère généralisé :

$$d_G = \frac{1}{\pi} \left[\frac{\int_{-\infty}^{\infty} h'^2(t) dt}{\int_{-\infty}^{\infty} h^2(t) dt} \right]^{\frac{1}{2}}$$

Critère de Canny :

$$d_G = \frac{1}{\pi} \left[\frac{\int_{-M}^M h'^2(t) dt}{\int_{-M}^M h^2(t) dt} \right]^{\frac{1}{2}}$$

$h(x)$ est un *RIF* défini sur l'intervalle $[-M, M]$ Canny a défini les conditions aux limites suivantes :

- $h(0) = 0$
- $h(M) = 0$
- $h'(0) = S$ (pente à l'origine)
- $h'(M) = 0$

3.7 Segmentation en régions [Baillie 03] [Franck 01] [Duchesnay 01,a]

Fondamentalement, la segmentation est un processus qui consiste à découper une image en régions connexes présentant une homogénéité selon un certain critère, comme par exemple la couleur. L'union de ces régions doit redonner l'image initiale

On regroupe généralement les algorithmes de segmentation en trois grandes classes (voir [Baillie 03]) :

- Segmentation basée sur les pixels
- Segmentation basée sur les régions
- Segmentation basée sur les contours

La première catégorie travaille sur des histogrammes de l'image. Par seuillage, clustering ou clustering flou, l'algorithme construit des classes de couleurs qui sont ensuite projetées sur l'image. La segmentation est implicite puisqu'on suppose que chaque cluster de l'histogramme correspond à une région dans l'image. En pratique, ce n'est pas forcément le cas et il faut séparer les régions de l'image qui sont disjointes bien qu'appartenant au même cluster de couleur. Ces algorithmes sont assez proches des algorithmes de réduction de couleur.

La deuxième catégorie correspond aux algorithmes d'accroissement ou de découpage de région. L'accroissement de région est une méthode bottom-up : on part d'un ensemble de petites régions uniformes dans l'image (de la taille d'un ou de quelques pixels) et on regroupe les régions adjacentes de même couleur jusqu'à ce qu'aucun regroupement ne soit plus

possible. Le découpage de région est le pendant top-down des méthodes d'accroissement : on part de l'image entière que l'on va subdiviser récursivement en plus petites régions tant que ces régions ne seront pas suffisamment homogènes.

La troisième catégorie s'intéresse aux contours des objets dans l'image. La plupart de ces algorithmes sont locaux, c'est à dire fonctionnent au niveau du pixel.

Les algorithmes que nous allons présenter sont :

- Accroissement de région fonctionnant au niveau du pixel.
- *Split and merge* classique.

Des études détaillées de ces algorithmes et d'autres sont données dans [Baillie 03].

3.7.1 Accroissement de région

Les méthodes d'accroissement de régions sont les méthodes de segmentation les plus simples. Le principe est basé sur une approche bottom-up : l'algorithme part de petits éléments de l'image qu'il va tenter de regrouper en éléments plus importants.

Nous présentons ici la version de base de l'algorithme d'accroissement de région qui fonctionne en agrégeant des pixels. Supposons une région de couleur homogène R . Initialement, $R = 1 \text{ pixel}$. On va étendre la région R en incluant les pixels situés sur la frontière et dont la couleur est proche de celle de R (la variation de couleur est inférieure à un seuil δ caractéristique de ce type d'algorithmes). En répétant cette procédure jusqu'à ce qu'il n'y ai plus de pixels de couleur assez proche sur la frontière, on obtient une région de couleur homogène maximale autour du pixel de départ. La région initiale "gonfle" en absorbant des pixels de la frontière, jusqu'à stabilité par rapport à la propriété d'homogénéité. Afin d'assurer un critère de connexité, on utilise les voisins en 4-connexités pour l'accroissement de région et la 8-connexités pour l'accroissement de contour (cas de détection de contours).

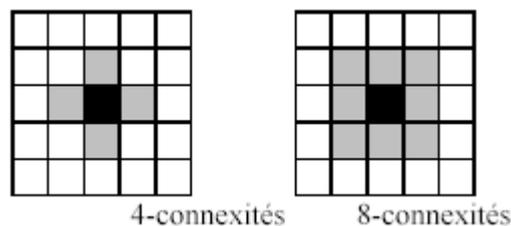


Figure 3.9 Voisinage d'un pixel [Achard 03]

Cette méthode présente deux limitations:

- Les régions obtenues dépendent fortement des pixels d'amorçage choisis et de l'ordre dans lequel les pixels de la frontière sont examinés.
- Le résultat final est très sensible à la valeur du seuil δ .

Cet algorithme fait parti de la classe d'algorithmes de segmentation dit "locaux". L'opération élémentaire consiste à manipuler des pixels adjacents et l'algorithme n'a aucune vision globale du résultat qu'il obtient.

3.7.2 Méthodes du type "split and merge"[Baillie 03] [Duchesnay 01,a]

L'algorithme "split and merge" a été présenté la première fois en 1974 par Pavlidis et Horowitz [Baillie 03]. Cet algorithme s'apparente dans son principe à l'algorithme d'accroissement de région que nous venons de présenter. La différence principale provient de la nature des régions élémentaires agrégées. Dans l'algorithme "split and merge", les régions agrégées proviennent d'une première phase (split) de traitement de l'image qui construit de manière récursive des régions carrées de taille variable mais homogènes.

☞ *Split*

La méthode de découpage de l'image utilisée dans cet algorithme est basée sur la notion de "quadtree". Cette structure de données est un arbre quaternaire qui permet de stocker l'image à plusieurs niveaux de résolution. On part d'une région initiale qui est l'image tout entière. Si cette image vérifie un critère d'homogénéité de couleur, l'algorithme s'arrête. Sinon, on découpe cette région en quatre parties de même taille et on lance la procédure récursivement dans ces quatre parties. La région initiale va être stockée comme un nœud dans un graphe et les sous parties comme des fils de ce nœud. La figure 3.10 montre une image en noir et blanc 8x8 et le découpage correspondant à chaque niveau, ainsi que la structure d'arbre associée à ce découpage.

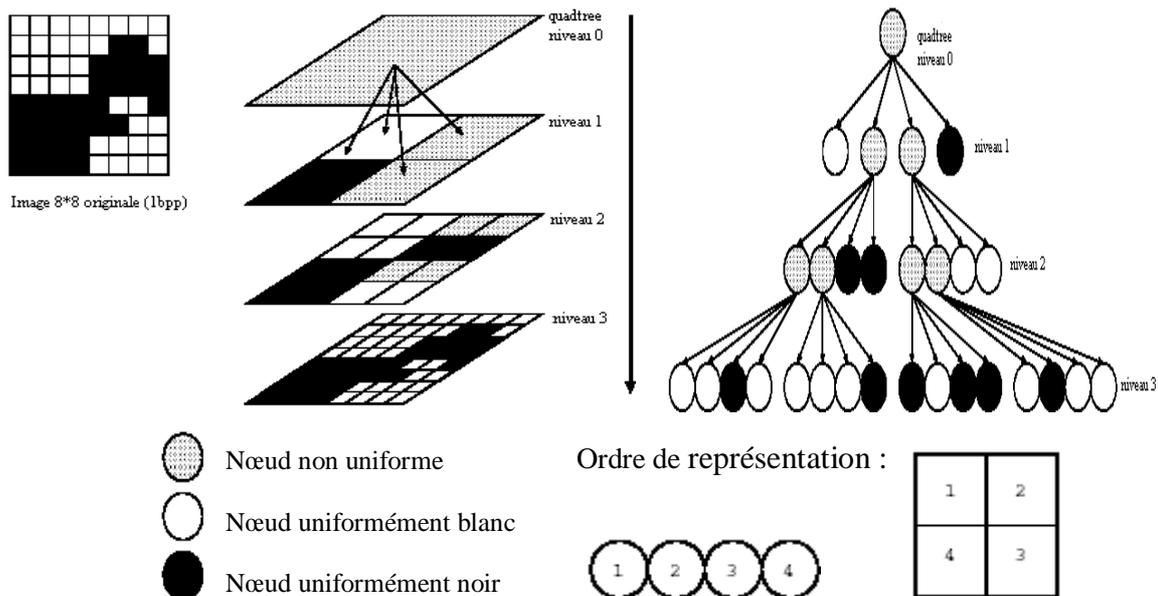


Figure 3.10 *Découpage par quadtree d'une image 8x8. Traitement avec seuil=100% [Baillie 03]*

☞ Merge

La procédure de découpage décrite précédemment aboutit à un nombre de régions trop élevé. La cause fondamentale de cette sur-segmentation est que l'algorithme découpe les régions de manière arbitraire. Il se peut qu'il coupe de cette façon une zone homogène en deux ou quatre parties (Figure 3.11).

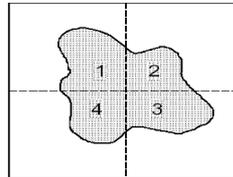


Figure 3.11 *Problème de découpage arbitraire de régions dans la phase de "split"*[Baillie 03]

Les parties 1,2,3 et 4 de la (Figure 3.11) par exemple, appartiennent à des branches différentes du quadtree. Elles sont donc considérées comme des régions différentes bien que leur couleur soit identique.

La solution, qui correspond à la phase "merge" de l'algorithme, est de procéder à une fusion de régions après le découpage. L'implémentation la plus simple de cette fusion cherche tous les couples de régions adjacentes dans l'arbre issu du découpage et cherche à les fusionner si leur couleur est assez proche.

3.8 Approches coopératives en traitement d'images

La résolution collective de problème se préoccupe de développer des systèmes d'agents *simples* capables de résoudre collectivement un problème complexe ; par exemple construire un chemin reliant tous les nœuds d'un graphe et dont la longueur est minimale (problème dit du voyageur de commerce ou TSP).

La question principale qui se pose alors est de déterminer le système de représentation de l'environnement, les comportements individuels, et la dynamique qui va régir le fonctionnement du système afin qu'il produise une réponse au problème qui lui est posé.

Plusieurs modèles inspirés des insectes sociaux ont déjà été élaborés et utilisés pour effectuer de la résolution de problème. L'une des applications de ces modèles est le traitement d'images.

Dans ce type d'approches, l'image y est vue comme un environnement ou territoire analysé par une société d'agents (ou processus) distribués spatialement. Cette approche est novatrice car elle va à l'encontre de la conception classique centralisée et planifiée des tâches d'analyse d'images. Ces architectures mettent en relief la distribution de calcul, l'adaptation

locale des traitements et la richesse des coopérations possibles entre les aspects contour et région.

Dans ce qui suit, on va présenter quelques approches appliquées aux traitement d'images.

3.8.1 Système de Bellet

Bellet [Bellet 98] a développé un système coopératif pour la segmentation de bas niveau des images. Ce système est multi-processus, puisqu'une image est segmentée par une multitude de processus indépendants, qui sont basés sur des modèles de croissance de région ou de suivi de contour. Chaque processus est initialisé à un endroit précis de l'image, avec un germe, et il fait croître sa primitive à partir de cet endroit. La croissance (région ou contour) s'effectue par une évaluation multi-critères de pixels avoisinants et sélection des meilleurs candidats(figure 3.12).(le lecteur intéressé peut se référé à [Bellet98]).

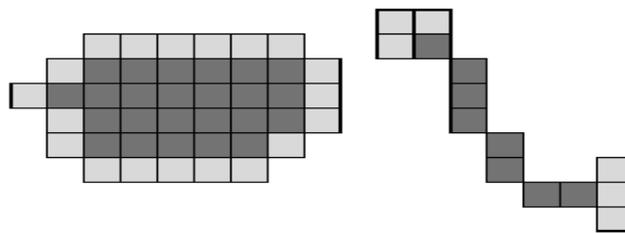


Figure3.12 Principes de la croissance de région et du suivi de contour. Pour faire croître la primitive (pixels foncés), différents candidats sont possibles (pixels clairs). Le pixel ayant la meilleure évaluation sera ajouté à la primitive.

Ce système est basé sur une nouvelle forme de coopération entre différents processus de segmentation de natures différentes. Lorsqu'un processus manque d'informations pour mener à bien les décisions qu'il doit prendre, il crée de nouveaux processus, qualifiés de fils pour segmenter de nouvelles primitives et ainsi, récolter plus d'informations sur l'environnement local. Un processus peut également s'appuyer sur les informations récoltées par ses voisins pour mener à bien sa tâche. C'est sur ce modèle de coopération que s'effectue toute la segmentation de l'image.

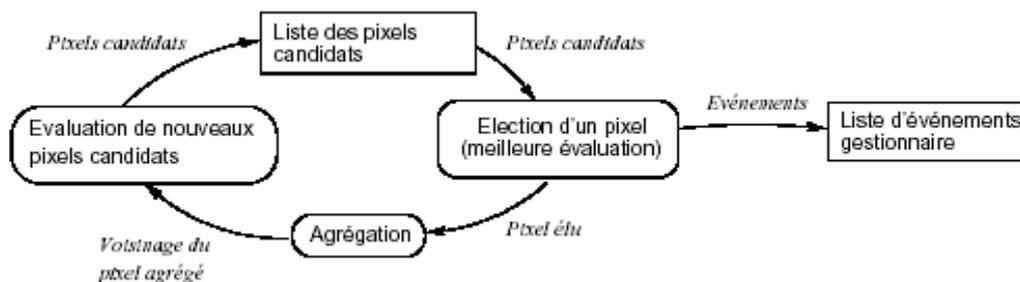


Figure3.13 Boucle de contrôle du comportement de perception. Ce comportement boucle sur une liste de pixels tant que celle-ci n'est pas vide. [Boucher 99]

Le système est généraliste et ne bénéficie d'aucune connaissance sur les applications et les images qu'il segmente. Il a été utilisé pour différents types d'images.

3.8.2 Approche de Boucher

Alain Boucher [Boucher 99] étend les processus de Bellet [Bellet 98] à des agents permettant de bien identifier les structures internes aux agents et leurs interactions. Les agents sont spécialisés dans la segmentation et l'interprétation d'images cytologiques; un des aspects les plus intéressants concerne le traitement de séquences d'images et la manière de lancer des agents dans l'image suivante de la séquence, en fonction de ce qui est en train de se dérouler dans l'image courante. Les agents disposent de quatre comportements :

- Le comportement de perception explore son environnement et sélectionne des pixels candidats à l'agrégation.
- Le comportement d'interaction gère les fusions et les échanges de pixels entre agents.
- Le comportement de différenciation correspond à une interprétation à valeur sémantique. Les agents représentent au départ des primitives région ou contour ; les informations accumulées sur eux-même et dans un environnement commun vont leur permettre de déterminer la nature sémantique de la primitive traitée.
- Le comportement de reproduction, permet de créer au besoins des agents à certains endroits de l'image. Il conditionne l'exploration de cette image et donc le contrôle de système.

3.8.3 Approche de Duchesnay

Duchesnay [Duchesnay 01] s'appuie sur la structure de pyramide irrégulière (voir la figure 3.14) pour gérer le processus de fusion de régions et assurer la convergence de la segmentation; une coopération région-région assez sophistiquée est mise en œuvre, mais qui ne tire pas suffisamment parti de l'information contour. Un des aspects intéressants de son approche est l'utilisation d'une procédure de décimation (récursive) pour le passage du niveau k au niveau $k+1$. La pyramide se construit en partant de la base qui représente l'image pré segmentée (par exemple avec l'algorithme des quadrees) jusqu'au dernier niveau de la pyramide comportant le minimum d'information. Les niveaux de cette pyramide sont des graphes d'adjacence de régions.

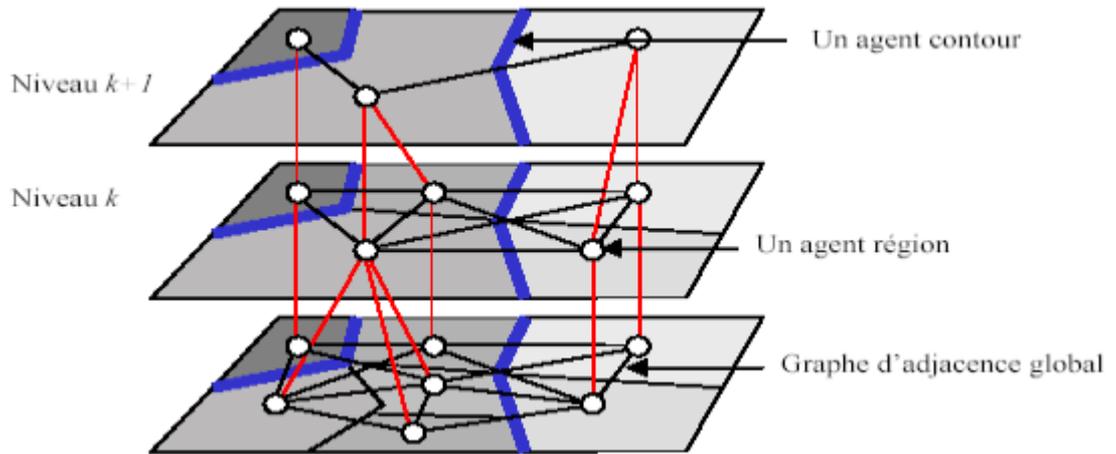


Figure 3.14 La pyramide de graphe d'adjacence des régions d'après [Duchesnay 01]

3.8.4 Approche de Liu & Tang : Segmentation par « agents migrants »

Dans cette approche [Liu 99,+], des agents répartis aléatoirement dans l'image vont migrer (se diffuser) de pixel en pixel jusqu'à ce qu'ils trouvent dans leur environnement local un motif recherché. Une fois fixé sur le pixel, l'agent l'étiquette puis se reproduit en initialisant de nouveaux agents dans son voisinage et devient inactif.

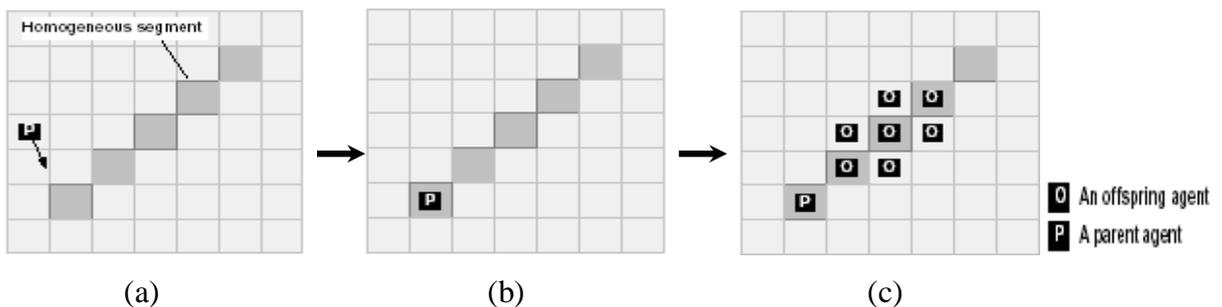


Figure 3.15 Comportement de reproduction. (a) Au temps t , un agent, indiqué par le symbole 'P', est prêt à déplacer selon une direction. (b) Au temps $t+1$, l'agent est déplacé à un pixel de segment homogène. (c) Au temps $t+2$, l'agent s'auto-produit un nombre d'agents descendants, indiqués par le symbole 'O', en une direction spécifique calculée à partir de son vecteur propre. [Liu 99,+]

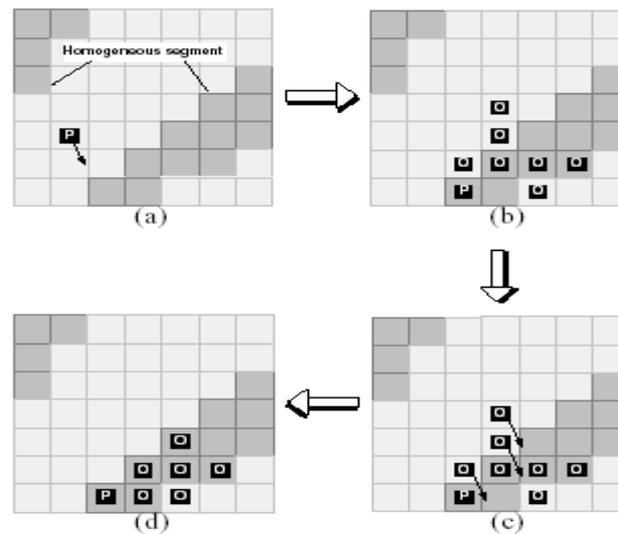


Figure 3.16 Comportement de diffusion (a) Au temps t , un agent 'P', est prêt à se déplacer à un emplacement voisin dans une direction comme indiqué. (b) Dans le temps $t+1$ et $t+2$, l'agent rencontre un pixel de segment homogène et par la suite, il s'auto-reproduit un nombre d'agents descendants 'O'. (c) Les agents descendants 'O' dans des régions non homogènes sont prêts à entreprendre des mouvements de diffusion dans leurs régions voisines, dans une direction comme héritée de leur parent et les agents de mêmes parents. Les agents qui se trouvent aux pixels de segment homogène, produiront plus d'agents descendants (bien qu'il ne soit pas montré dans la figure). (d) Au temps suivant, certains des agents se répandant comme indiqué dans (c) rencontreront plus de pixels de segment homogène. [Liu 99 ,+]

Au cours de sa migration, l'agent va vieillir et peut mourir si le nombre de cycles de recherche excède son temps de vie. Les auteurs ont aussi introduit la notion d'évolution associée aux directions de diffusion et de reproduction. En effet, en cas de succès d'un des comportements dans une direction, la pondération associée à cette direction sera renforcée. Ainsi les comportements de migration et reproduction de ses descendants favoriseront cette direction.

Les auteurs appliquent leur approche à la détection de segment de contour ; les agents migrent jusqu'à trouver un contour, ils s'y fixent, l'étiquettent, et initialisent de nouveaux agents qui auront de fortes chances de trouver un contour recherché dans un voisinage proche.

3.8.5 Une plate-forme pour la segmentation des images IRM

L'objectif de cette plate-forme [Settache 02] multi-agent est la segmentation d'images appliquée aux IRM cérébrales 2D. Dans ce travail, le parallélisme de segmentation permet d'envisager de réaliser la fusion simultanée de couples de régions voisines quelle que soit leur position dans l'image, et quels que soient les critères de candidature à la fusion utilisés.

Dans [Settache 02], Il existe différents protocoles de coopération :

- Coopération région-région pour la fusion de régions.
- Coopération contour-région pour la fusion de régions : il ne doit pas exister de contour entre les deux régions qui souhaitent fusionner
- Coopération contour-région pour la division de région : un contour qui traverse une région peut inciter cette région à se diviser.
- Coopération contour-contour pour le raccordement de contours.
- Coopération région-contour pour le raccordement de contours.
- Coopération région-contour pour le prolongement de contours : on peut prolonger un contour à condition que les régions voisines ne s'y opposent pas. Par exemple, si le contour suit un morceau de la frontière entre deux régions, on pourra le prolonger le long de la frontière.

La figure 3.17 illustre la coopération région-région et région contour pour la fusion de régions. Au départ toutes les régions calculent leur désir de fusion avec chacun de leur voisins et ce désir sont communiqués au meilleur voisin par un message (par exemple une flèche sur la figure 3.17a entre R10 et R110). Une opération de fusion est exécutée quand le désir de fusion est mutuel (figure 3.17.b entre R160 et R122) et la région se met alors dans un état d'attente jusqu'à exécution de la fusion de son meilleur voisin (par exemple la région R110 dans la figure 3.17.b). Comme on peut voir sur la figure 3.17.d la fusion des deux régions R210-R130 est stoppée par l'agent contour qui se situe aux frontières de ces deux régions.

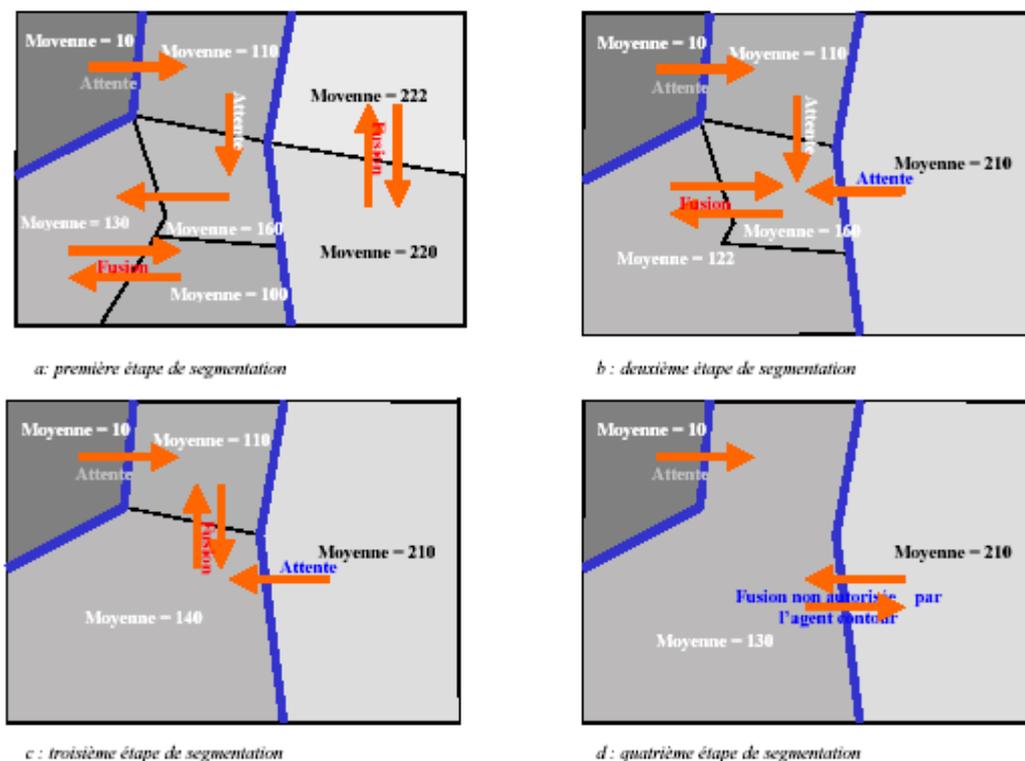


Figure 3.17 Exemple de coopération région-contour et région-région pour la fusion des régions [Settache 02]

3.8.6 Détection de régions dans des images à niveaux de gris : l'inspiration des araignées sociales

Ce travail [Bourjot 01,+] a exposé une démarche de transposition pour passer d'un modèle de simulation à un modèle de résolution. Le cadre biologique initial concerne un processus de construction collective de toile chez les araignées sociales. Sa transposition concerne la détection de régions dans des images à niveau de gris.

L'environnement est un carré de $N \times N$ piquets dans lequel évoluent les araignées et correspond à l'image à analyser (figure 3.18.a). Un piquet correspond à un ou plusieurs pixels de l'image d'origine, selon l'échantillonnage réalisé. La hauteur d'un piquet correspond au niveau de gris du ou des pixels correspondants (figure 3.18.b). Il peut donc y avoir 256 hauteurs différentes. Initialement, il n'y a pas de fils dans l'environnement, ils apparaîtront au fur et à mesure du déplacement des araignées.

Les items comportementaux d'une araignée sont le *déplacement*, la *pose* d'un fil et, en plus, un comportement de *retour* sur la toile. L'araignée est dotée d'un état interne pour permettre une pose contextuelle.

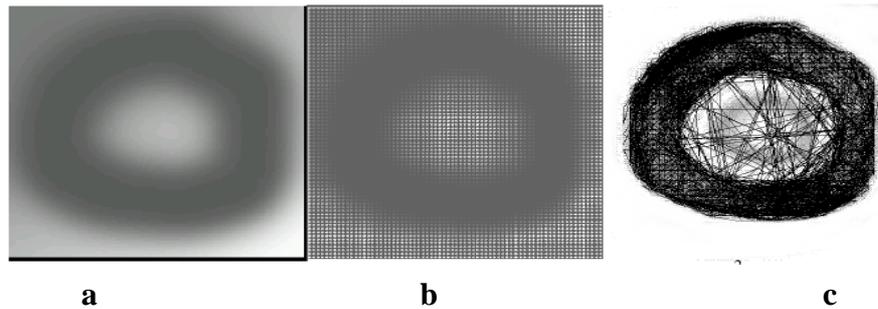


Figure 3.18 Exemple de résultat, d'après [Bourjot01+]

La dynamique du système est la même que dans le cas de la simulation et est de type stigmergique. Ce processus construit des toiles qui sont interprétées comme des régions de l'image.

3.8.7 Traitement d'images par vie artificielle

Carden [Carden 02] propose une approche de détection des contours des régions dans une image simple en niveaux de gris par vie artificielle. Ce travail est basé sur les travaux de Reynolds et MacGill [MacGill 00] où les agents représentent les « géoboids », qui permettent de calculer le gradient.

Les agents ont différents comportements de direction trois inspirés de Boids de Reynolds : (voir figure 3.19) cohésion, alignement et séparation plus deux autres inspirés de Géoboids de MacGill : chercher et fuir.

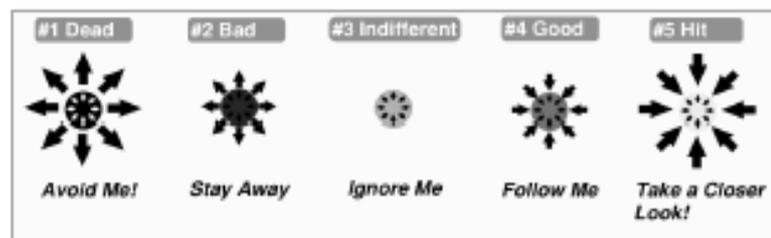


Figure 3.19 Cinq états de geoBoid, utilisés pour les boids de traitement d'images [MacGill 00]

Ces deux derniers comportements visent à chercher : se diriger vers des agents satisfaits (qui sont dans une position proche du contour), et à fuir : s'éloigner des agents insatisfaits (position plus loin du contour).

Un résultat d'application de cette méthode est indiqué dans la (figure 3.20).

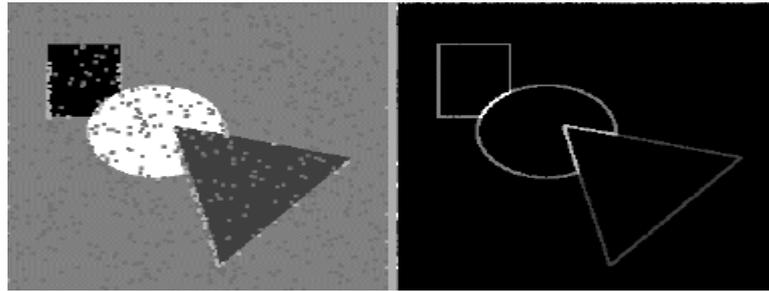


Figure 3.20 Exemple de résultat (image de synthèse) d'après [Carden 02]

Conclusion

Nous avons introduit dans ce rapport les notions de base qui servent de fondement à la compréhension de différentes techniques de traitement d'images. Plusieurs méthodes classiques de traitement ont été proposées dans la littérature, nous avons présenté quelques unes qui nous semblent les plus courantes dans le processus du traitement et analyse d'image.

Les pré-traitements d'images permettent d'améliorer la qualité de l'image en vue de traitements ultérieurs. Deux grandes approches peuvent être envisagées pour extraire les zones pertinentes des images :

On cherche les discontinuités dans la scène, c'est l'approche contour.

On cherche des zones de niveaux de gris homogènes, c'est l'approche région

De nouvelles recherches s'intéressent aux systèmes complexes et leurs caractéristiques ainsi l'application de ces approches sur le traitement d'images.

Les phénomènes collectifs naturels sont une source d'inspiration pour proposer des méthodes multi-agents de résolution de problèmes. Ils permettent l'élaboration de comportements individuels simples produisant collectivement des phénomènes complexes.

A la fin de ce chapitre, nous avons présenté les différentes approches qui ont été élaborées pour le traitement d'images en niveaux de gris soit par agents simples situés dans l'image, ou bien, par inspiration des systèmes naturels.

