

وزارة التربية الوطنية
MINISTERE DE L'EDUCATION NATIONALE

ECOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

DEPARTEMENT : Génie Electrique.
OPTION : Automatique.

PROJET DE FIN D'ETUDES

SUJET

ETUDE COMPARATIVE ENTRE
LE TRAITEMENT D'IMAGES PAR LES METHODES
CLASSIQUES ET PAR LES RESEAUX DE NEURONES.

Proposé par :
M.C. SOUAMI.

Etudié par :
SENOUCI-BRIKSI Djelloul.
YEDDOU Yazid M'hamed.

Dirigé par
M.C. SOUAMI.

PROMOTION

Juin 1994.

DEPARTEMENT : Génie Electrique.
OPTION : Automatique.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

PROJET DE FIN D'ETUDES

SUJET

ETUDE COMPARATIVE ENTRE
LE TRAITEMENT D'IMAGES PAR LES METHODES
CLASSIQUES ET PAR LES RESEAUX DE NEURONES.

Proposé par :
M.C. SOUAMI.

Etudié par :
SENOUCI-BRIKSI Djelloul.
YEDDOU Yazid M'hamed.

Dirigé par
M.C. SOUAMI.

PROMOTION

Juin 1994.

المدرسة الوطنية المتعددة التقنيات
المكتبة — BIBLIOTHEQUE
Ecole Nationale Polytechnique

DEDICACES



A mon Père, pour son soutien.

A ma Mère, pour son amour.

A mes frères et soeurs.

A ma petite nièce Manel.

A mon petit neveu Mohamed Yacine.

A tous ceux qui me sont chers.

Je dédie ce travail.

Djelloul.



A ma Mère et mon Père ,

*simple témoignage de mon profond et affectueux
attachement.*

A mes frères et soeurs.

A mes proches et tous ceux qui me sont très chers.

Yazid.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

REMERCIEMENTS

Nous remercions particulièrement notre promoteur Monsieur M.C.SOUAMI pour ces précieux conseils et l'intérêt constant avec lequel il a suivi le progrès de notre travail.

Nous exprimons notre reconnaissance aux responsables des centres de calcul Messieurs H.DJOUZI et L.YAHI qui ont mis à notre disposition tous les moyens sans lesquels notre travail n'aurait pas pu être mené à bon terme.

Que Madame HAMMAMI trouve ici notre profonde gratitude pour l'aide et le soutien qu'elle nous a apporté .

Nous remercions également Monsieur M.BENABADJI pour la bibliographie qu'il a mis à notre disposition.

Enfin, Nos remerciements s'adressent à tous les membres du jury pour nous avoir honoré en acceptant de juger notre travail.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

RESUME

Le traitement d'images est une discipline très importante qui a connu un grand essor cette dernière période, grâce à ses applications qui deviennent de plus en plus grandissantes.

Dans le présent projet, nous proposons deux techniques différentes de traitement d'images destiné à la robot-vision :

- Le traitement d'image basé sur les méthodes classiques.
- Le traitement d'image en utilisant les réseaux de neurones.

Dans la première partie, nous étudions et nous mettons en oeuvre plusieurs techniques de traitement, afin de constater les résultats qu'elles peuvent donner.

Dans la seconde partie, l'élaboration de réseaux de neurones pour remplir différentes fonctions, nous permettra de juger sur les grandes aptitudes des réseaux de neurones dans le domaine de l'imagerie .

La mise en oeuvre du traitement de plusieurs images par les deux techniques, nous amènera à établir une comparaison et de conclure sur la puissance des réseaux de neurones dans ce domaine.

Enfin, ce travail nous conduira à déduire les perspectives que peut ouvrir la technique de la vision neurale.

SOMMAIRE

INTRODUCTION.

1

CHAPITRE I : GENERALITES SUR L'IMAGERIE

I.1. Définition d'une image analogique.	5
I.2. Acquisition de l'image.	5
I.2.1. La caméra.	5
I.2.2. Le scanner.	6
I.3. Numérisation de l'image.	6
I.4. Image et robot-vision.	7
I.4.1. La vision de bas niveau.	7
I.4.2. La vision de haut niveau.	8

CHAPITRE II : TRAITEMENT D'IMAGES PAR LES METHODES CLASSIQUES.

II.1. Traitement de bas niveau (low level vision).	10
II.1.1. Introduction.	10
II.1.2. Lissage de l'image.	11
II.1.3. Amélioration de l'image.	14
II.1.4. Détection de contours.	18
II.1.5. Seuillage de l'image.	25
II.2. Traitement de haut niveau (high level vision).	27
II.2.1. Introduction.	27
II.2.2. Edge linking (établissement de liaisons).	28
II.2.3. Suivi de contours.	29
II.3. Conclusion.	31

CHAPITRE III : PRESENTATION DES RESEAUX DE NEURONES ARTIFICIELS.

III.1. Introduction.	34
III.2. Les réseaux de neurones artificiels.	34
III.3. Architecture des réseaux de neurones.	35
III.4. Apprentissage des réseaux de neurones.	36
III.4.1. Backpropagation.	37
III.4.2. M.O.A.	38

CHAPITRE VI : TRAITEMENT D'IMAGES PAR LES RESEAUX DE NEURONES.

VI.1. Introduction.	42
VI.2. Extraction des contours.	42
VI.2.1. Objectif.	42
VI.2.2. Elaboration du réseau.	42
VI.2.3. Applications.	44
VI.3. Reconnaissance de formes.	52
VI.4. Détection des axes.	56
VI.5. Sélection d'objets couleur.	59
VI.6. Conclusion.	61

CHAPITRE V : COMPARAISON ENTRE LA VISION CLASSIQUE ET LA VISION NEURALE.

V.1. Introduction.	63
V.2. Présentation des exemples.	63
V.3. Interprétation des résultats et comparaison.	68
V.4. Conclusion.	69
CONCLUSION GENERALE.	71
BIBLIOGRAPHIE.	

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

INTRODUCTION GENERALE

INTRODUCTION GENERALE

Durant les dernières années, le développement de l'intelligence artificielle et de l'informatique industrielle a permis d'introduire la technique des " réseaux de neurones artificiels ", qui connaît un très grand intérêt dans les milieux scientifiques.

D'autre part, de toutes les disciplines déjà existantes, le traitement d'images est celle qui a connu le plus grand essor depuis la dernière décennie.

La robot-vision est justement une partie très importante de cette discipline. Son importance réside dans ces applications qui découleraient d'une machine qui saurait tenir compte de son environnement et agir en conséquence à partir des scènes visuelles présentes devant elle.

De là, une association très intéressante est celle liant le traitement d'images et les réseaux neuronaux, par l'importance et les applications diverses de la première, et la puissance et les capacités reconnues du second.

C'est le travail que nous proposons d'effectuer dans le présent projet.

Dans un premier temps, nous aborderons l'imagerie par son aspect classique. Nous mettrons en oeuvre plusieurs opérations sur des images afin d'apprécier leurs performances.

Nous introduirons ensuite les réseaux de neurones artificiels, et nous entamerons la " neural vision ". La majeure partie sera ici consacrée à l'étude des capacités des réseaux neuronaux à remplir les fonctions classiques de traitement d'images, et particulièrement la segmentation, c'est à dire l'extraction de contours à partir de différentes images brutes.

Nous proposerons également des applications importantes pour la construction d'un système complet de robot-vision.

Au terme de ce travail, et après avoir étudié les aptitudes des réseaux de neurones dans ce domaine, nous présenterons nos conclusions et les perspectives ouvertes par cette étude.

CHAPITRE I :

GENERALITES SUR L'IMAGERIE

I.1. Définition d'une image analogique :

Une fonction image est une fonction réelle à quatre variables $L(x, y, t, \lambda)$, associée à une scène visuelle, traduisant l'énergie lumineuse présente en un point de coordonnées (x, y) , à un instant t et sur une longueur d'onde λ .

Dans de nombreux cas, cette fonction peut être fixée à une fonction $L(x, y)$, en la considérant monochrome (λ fixe) et figée dans le temps (t fixe) [8, 3].

La fonction $L(x, y)$ associe une valeur réelle comprise entre 0 et M , dans un domaine D , et une valeur nulle à l'extérieur de ce domaine, soit :

$$\begin{aligned} L(x, y) &\in [0, M] \text{ pour } (x, y) \in D. \\ L(x, y) &= 0 \text{ ailleurs.} \end{aligned}$$

I.2. Acquisition de l'image:

Cette opération nous permet de convertir une scène réelle en données numériques pouvant être traitées par ordinateur.

Il existe actuellement deux procédés de saisie :

I.2.1. La caméra :

Les plus pratiques sont les caméra CCD (Charged Coupled Device) [8]. Elles sont constituées d'un assemblage de photodiodes correspondant chacune à un pixel de l'image; elles délivrent ainsi une tension proportionnelle à l'énergie lumineuse présente en ce point.

I.2.2. Le scanner :

C'est un dispositif permettant l'acquisition et la numérisation d'une image analogique à partir d'un support papier.

L'image à scanner est placée sur une vitre transparente. Cette image reçoit une lumière constante émise par des tubes luminescents. La lumière réfléchiée par les différents points de l'image est recueillie par une barrette CCD, via un miroir et une lentille focalisatrice, qui effectue le balayage de l'image. Les cellules photosensibles de la barrette CCD émettent alors des tensions proportionnelles à la lumière réfléchiée, donnant ainsi le niveau de gris de chaque pixel.

Le scanner que nous allons utiliser au cours de ce travail est le " Scan Jet " de Hewlett Packard, disponible au laboratoire d'Automatique.

I.3. Numérisation de l'image:

Afin d'être traitée par ordinateur, une image analogique doit être numérisée.

Le processus de numérisation peut être décomposé en trois opérations : [8]

- Balayage de l'image par un spot de dimensions finies.
- Echantillonnage spatial bidimensionnel.
- Quantification de l'image.

L'image obtenue est une matrice à deux dimensions $f(i, j)$, dont chaque élément (i, j) constitue ce qu'on appelle un " pixel " (figure I.1). La taille du pixel définit la résolution de l'image. La valeur de cet élément représente le niveau de gris en ce point. Cette valeur est comprise entre 0 et M, où 0 correspond au blanc et M correspond au noir.

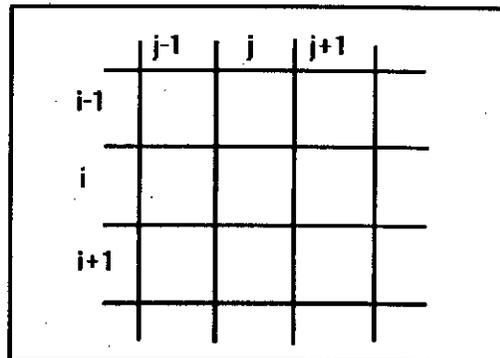


Figure I.1.

I.4. Image et robot-vision:

Le traitement d'image destiné à la robot-vision est un processus d'extraction et d'interprétation d'informations à partir d'une image, permettant à une machine d'appréhender son environnement d'une manière intelligente. [1].

Ce processus appelé encore " vision par ordinateur " ou " computer vision " est généralement subdivisé en deux grandes parties : [1]

- La vision de bas niveau (low level vision).
- La vision de haut niveau (high level vision).

I.4.1. La vision de bas niveau :

Les opérations de bas niveau commencent à partir de l'acquisition. A cette étape, l'image est représentée sous forme brute, avec plusieurs niveaux de gris et, éventuellement, des distorsions, des redondances, et des bruits qui peuvent être enregistrés.

Les opérations principales à ce niveau ont pour finalité de réduire toutes informations jugées inutiles, telles que les bruits.

Théoriquement, afin de parvenir à cet objectif, une série d'opérateurs mathématiques sont automatiquement appliqués. Ces opérateurs nous amènent de l'image initiale brute à une image prête à être exploitée.

I.4.2. La vision de haut niveau :

Celle-ci intervient immédiatement après les opérations précédentes et permet l'exploitation des informations extraites. En effet, la vision de haut niveau doit opérer sur des images déjà pré traitées.

A ce stade, le système de vision doit pouvoir interpréter les résultats afin de prendre les décisions adéquates, et ainsi arriver au bout du processus défini par la robot-vision.

Chapitre II :

TRAITEMENT D'IMAGES PAR
LES METHODES CLASSIQUES

II.1. Traitement de bas niveau:

II.1.1. Introduction :

Il existe deux approches principales dans le traitement d'images : la première est basée dans le domaine spatial, la seconde dans le domaine fréquentiel. [3, 7, 11, 8]. Dans notre cas, nous nous intéresseront à la première approche.

Le traitement d'images dans le domaine spatial consiste à opérer directement sur le pixel lui-même, en lui appliquant une fonction de traitement $h (.)$:

$$g (i , j) = h (f (i , j))$$

où:

$f (i , j)$: est le pixel image d'entrée.

$g (i , j)$: est le pixel image de sortie.

$h (.)$: est un opérateur défini sur quelques pixels adjacents au pixel (i , j) .

Cette approche consiste donc à utiliser une sub-image carrée ou rectangulaire, centrée en (i , j) . Le centre est déplacé pixel par pixel.

La plus simple des formes de $h (.)$ est celle agissant sur une fenêtre (1×1) , $g (.)$ ne dépend alors que de la valeur de $f (.)$ au point (i , j) .

La plus fréquente des opérations dans le domaine spatial est ce qu'on appelle " masque de convolution " (fenêtre de convolution, filtre de convolution).

Un masque est une matrice $(M \times N)$ dont les coefficients sont choisis de telle manière à faire apparaître certaines propriétés bien définies de l'image. La fonction $g (i , j)$ est le résultat de la convolution du masque $(M \times N)$ par la fenêtre définissant le neighborhood du point (i , j) : [3, 8]

$g = h \otimes f$ où " \otimes " symbolise l'opérateur de convolution.

$$g(i, j) = \sum_{k=-M}^{M-1} \sum_{l=-N}^{N-1} h(k, l) \cdot L(i-k, j-l).$$

II.1.2. Lissage de l'image:

Le lissage (smoothing) consiste à réduire une grande partie du bruit ou tout effet parasite présent dans une image. En effet, ceux-ci ont généralement une fréquence plutôt élevée par rapport au reste de l'image, d'où la nécessité d'un filtrage passe-bas. [18].

On distingue deux types de filtrage passe-bas:

- Le filtrage passe-bas linéaire.
- Le filtrage passe-bas non linéaire.

II.1.2.1. Le filtrage passe-bas linéaire :

Il consiste à réduire les fluctuations rapides des niveaux de gris. L'opération la plus courante est le " neighborhood averaging ": le pixel considéré est remplacé par la valeur moyenne des 9 pixels de la fenêtre étudiée. On utilise donc un masque (3 x 3) : [8, 14]

$$M = (1/9) \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$g(i, j) = M \otimes f(i, j) = (A + B + C + D + E + F + G + H + L(i, j)) / 9$$

où les coefficients (A, B, C, D, E, F, G, H, et L (i , j)) sont les éléments de la matrice définie par :

$$\begin{bmatrix} A & B & C \\ H & L(i,j) & D \\ G & F & E \end{bmatrix}$$

Un exemple de l'application du filtrage passe-bas linéaire est présenté dans la figure 2.1.

Cependant, ce filtre élimine les composantes H.F. du bruit, mais aussi celles du signal. Un possible brouillage (blurring) des contours peut alors apparaître. Ce phénomène peut être réduit, dans quelques cas par un filtrage non linéaire appelé filtre de la médiane [8, 1].

II.1.2.2. Le filtrage passe-bas non linéaire : (filtre de la médiane) :

Il consiste à remplacer la valeur du pixel considéré par la valeur médiane des pixels présents dans un voisinage (neighborhood) pré défini. Le plus commun est d'utiliser une fenêtre (3 x 3). [8, 11].

$$L (i , j) = \text{Médiane} (A, B, C, D, E, F, G, H, L (i , j))$$

où les coefficients (A, B, C, D, E, F, G, H, et L (i , j)) sont les éléments de la matrice définie par :

$$\begin{bmatrix} A & B & C \\ H & L(i,j) & D \\ G & F & E \end{bmatrix}$$

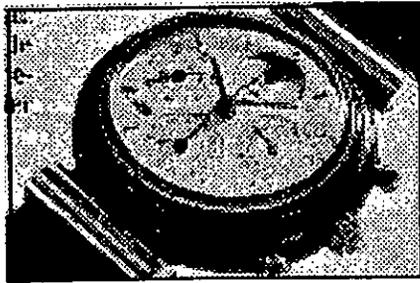
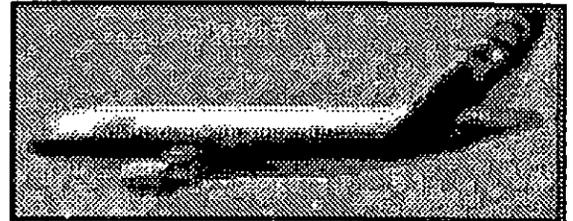
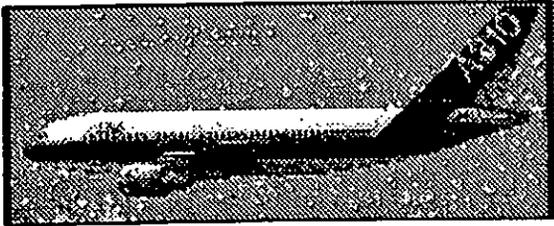


Figure 2.1. : Application du filtrage passe-bas linéaire.

L'effet du " blurring " est très apparent sur l'image de sortie.

II.1.3. Amélioration de l'image :

L'objectif de l'amélioration (enhancement) est de fournir une image (quelque fois différentes de l'image initiale) d'apparence visuelle meilleure pour l'observateur, ou pour mettre en évidence certains traits caractéristiques.

Le problème de l'amélioration d'images est un problème subjectif, qui dépend de chaque cas. Les principaux traitements liés à l'amélioration des images sont :

- La modification du contraste.
- La modification de l'histogramme.

II.1.3.1. Modification du contraste :

On améliore le contraste d'une image par une regraduation de l'intensité de chaque pixel. Ceci est effectué à travers " la fonction de transfert de l'amélioration du contraste ".

La figure 2.2. représente l'image, avant et après le passage par la fonction d'amélioration du contraste. On remarque bien que si l'image originale possède k niveaux de gris et occupe une bande étroite, l'image améliorée possède également k niveaux de gris, mais occupe une bande plus large.

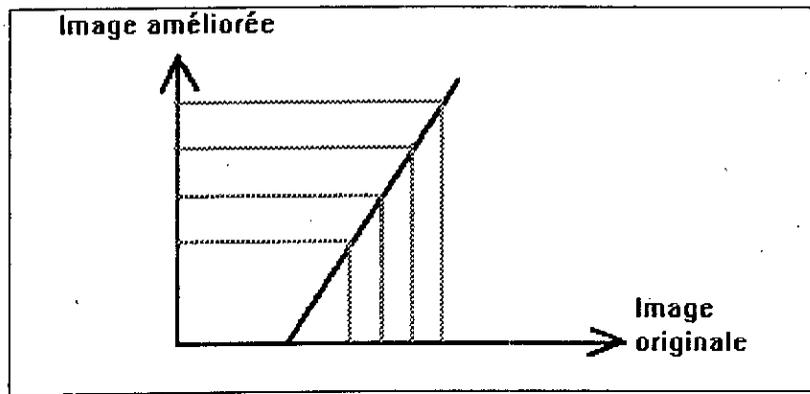


Figure 2.2.

S'il existe un nombre restreint de points non désirés dans l'image (dont la luminosité dépasse généralement celle des points voisins), on préfère utiliser , une fonction linéaire à seuil (voir figure 2.3.) : [3]

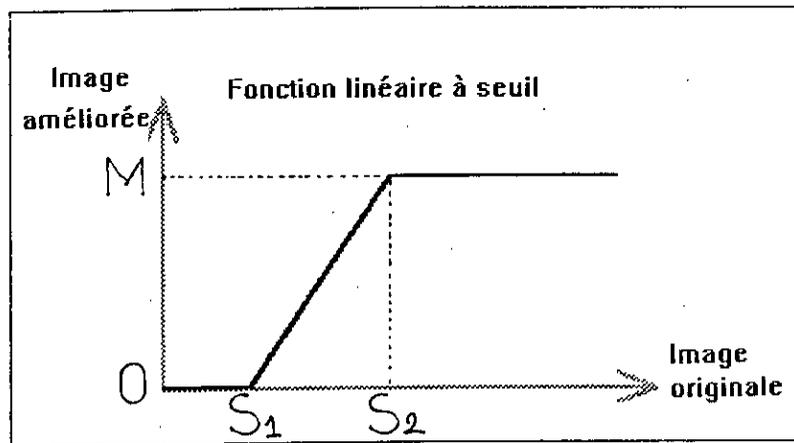


Figure 2.3.

Remarque:

La fonction de transfert d'amélioration du contraste n'est pas obligatoirement monotone et croissante (comme celles vues précédemment). Dans d'autres applications, on peut utiliser d'autres fonctions; par exemple (figure 2.4.) : [3]

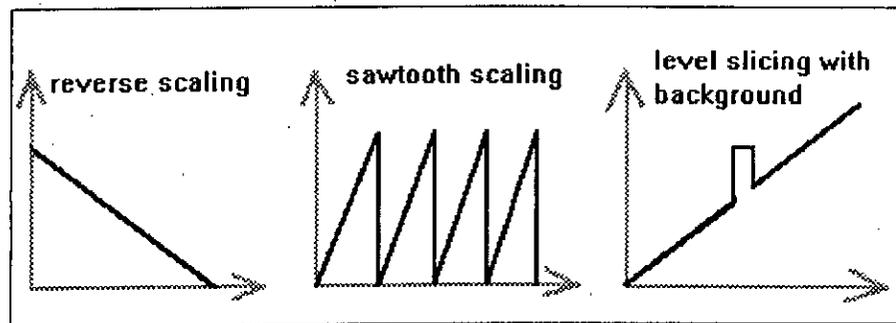


Figure 2.4.

La figure 2.5. représente l'effet de la modification du contraste d'une image.

II.1.3.2. Modification d'histogramme :

Certaines images ont un certain nombre de pixels dont la luminance est inférieure à la luminance moyenne, d'où imperception de certains détails dans les zones sombres.

La modification d'histogramme a pour objectif de mettre ce dernier sous une autre forme désirée. Cette modification peut être considérée comme une transformation monotone $T(\cdot)$, telle que :

$$g_k = T(f_k)$$

où :

f_k : est l'intensité de l'image originale.

g_k : est l'intensité de l'image améliorée.

Nous avons :

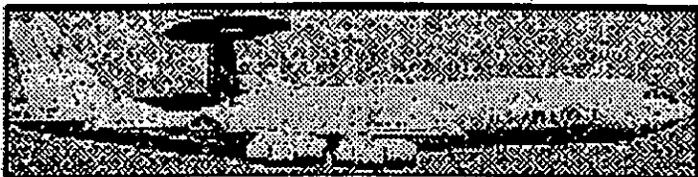
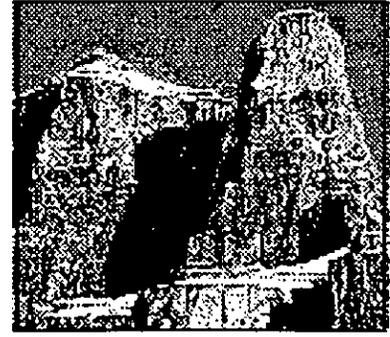


Figure 2.5. : Modification du contraste d'une image.

L'image de sortie est relativement meilleur en son aspect " visuel ".

$$\sum_{k=0}^m \Pr(f_k = a_k) = 1$$

$$\sum_{k=0}^m \Pr(g_k = b_k) = 1$$

avec: $\Pr(f_k = a_k)$: loi de probabilité de l'histogramme de l'image originale.

$\Pr(g_k = b_k)$: loi de probabilité de l'histogramme de l'image améliorée.

$$\sum_{k=0}^m \Pr(f_k = a_k) = \sum_{k=0}^m \Pr(g_k = b_k)$$

En passant au cas continu, on trouve :

$$\int_{g_{\min}}^g P_g(g) \cdot dg = P_f(f)$$

$P_f(f)$ étant connue (distribution de l'image originale), on peut imposer la distribution désirée pour l'image améliorée, et ainsi tirer $g(\cdot)$. [3]

$P_g(g)$ peut être, par exemple, une distribution uniforme, gaussienne, ect.

II.1.4. Détection de contours :

La détection des contours (Edge Detection) joue un rôle essentiel dans la robot-vision. En effet, c'est généralement la première opération que l'on applique avant l'opération de reconnaissance. [1]

La détection de contours consiste en un filtrage passe-haut qui permet d'obtenir une image plus " aiguisée " (sharpened), faisant apparaître les contours des objets et réduisant tout brouillage (blurring) contenu dans l'image.

Un contour est défini par une variation rapide d'intensité lumineuse. Cette variation peut donc être détectée par le maximum d'une première dérivée (gradient), ou par le passage à zéro d'une seconde dérivée (laplacien).

II.1.4.1. Opérateur gradient : [3, 7, 8]

Le gradient d'une image en un point (x , y) est le vecteur dont la phase indique le sens dans lequel le taux de changement de niveaux de gris est le plus grand, et dont le module est une mesure de ce taux de changement maximal.

Le gradient de l'image au point (x , y) sera donc défini par le vecteur :

$$G [L (x , y)] = \begin{bmatrix} \frac{dL}{dx} \\ \frac{dL}{dy} \end{bmatrix} = \begin{bmatrix} G_x \\ G_y \end{bmatrix}$$

dont l'amplitude est : $|G(I(x,y))| = \sqrt{(G_x^2 + G_y^2)} \approx |G_x| + |G_y|$

et la phase : $\langle G(I(x,y)) \rangle = \arctg(G_y/G_x)$

Cette opération revient à convoluer l'image avec les masques :

$$M_1 = (-1, +1) \quad \text{et} \quad M_2 = \begin{bmatrix} -1 \\ +1 \end{bmatrix}$$

appelés masques de Roberts.

Nous aurons :

$$G_x = \frac{dl}{dx} = L(x+1, y) - L(x, y) \quad \text{représentant la convolution de } M_1 \text{ avec l'image.}$$

et:

$$G_y = \frac{dL}{dy} = L(x, y+1) - L(x, y) \quad \text{représentant la convolution de } M_2 \text{ avec l'image.}$$

Il existe d'autres formes de gradients, qui ont l'avantage d'être moins sensibles aux bruits, par rapport à l'opérateur précédent.

Ces gradients opèrent sur une fenêtre (3 x 3), centrée au point étudié (x , y).

On définit ci-dessous les deux opérateurs les plus connus :

- **L'opérateur de Sobel** : [3, 7, 8, 13]

Il est défini par :

$$G_x = \frac{dl}{dx} = [L(x+1, y-1) + 2.L(x+1, y) + L(x+1, y+1)] - [L(x-1, y-1) + 2.L(x-1, y) + L(x-1, y+1)]$$

$$G_y = \frac{dL}{dy} = [L(x-1, y+1) + 2.L(x, y+1) + L(x+1, y+1)] - [L(x-1, y-1) + 2.L(x, y-1) + L(x+1, y-1)]$$

on tire des équations précédentes les masques de Sobel :

$$M_1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \quad \text{et} \quad M_2 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

La figure 2.6 représente le résultat de l'application de l'opérateur de Sobel sur quelques images scannées.

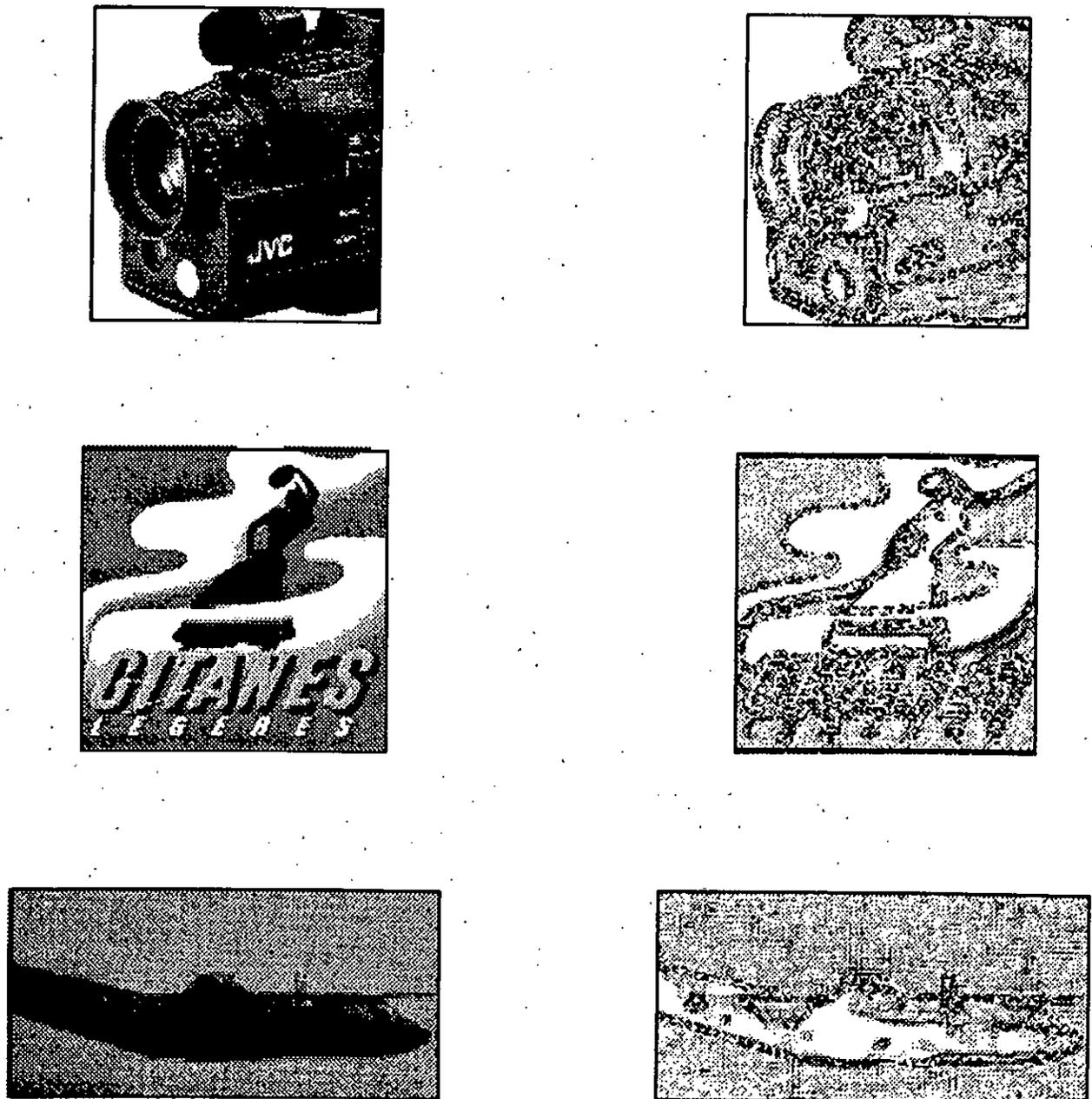


Figure 2.6. : Application de l'opérateur de Sobel.

Les contours des objets sont bien extraits, mais il apparaît encore quelques reflets. Il ya donc présence de bruits.

- L'opérateur de Prewitt: [3, 7, 8, 13]

$$G_x = \frac{dL}{dx} = [L(x+1, y-1) + L(x+1, y) + L(x+1, y+1)] - [L(x-1, y+1) + L(x-1, y) + L(x-1, y-1)]$$

$$G_y = \frac{dL}{dy} = [L(x-1, y+1) + L(x, y+1) + L(x+1, y+1)] - [L(x-1, y-1) + L(x, y-1) + L(x+1, y-1)]$$

ces équations aboutissent aux masques de Prewitt :

$$M_1 = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ +1 & +1 & +1 \end{bmatrix} \quad \text{et} \quad M_2 = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix}$$

Le résultat de l'application de l'opérateur de Prewitt sur quelques images est présenté dans la figure 2.7.

Remarque :

L'application de l'opérateur gradient de Roberts fait apparaître les zones frontières comme des zones à gradient maximums, et les zones homogènes comme des zones à gradient minimums.

Cependant, la présence d'un point bruité dans une zone homogène résulterait en un gradient élevé. Ceci constitue l'inconvénient principal de cet opérateur.

Afin de pallier à cet handicap, il existe un autre opérateur: l'opérateur gradient amélioré. [14]

II.1.4.2. Opérateur gradient amélioré :

L'image gradient est ici obtenue en calculant quatre gradients simple dans une fenêtre (4 x 4). La valeur du gradient amélioré est donnée par la moyenne géométrique des gradients calculés :

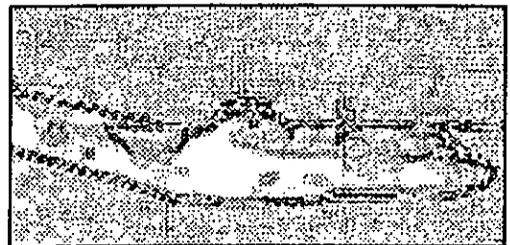
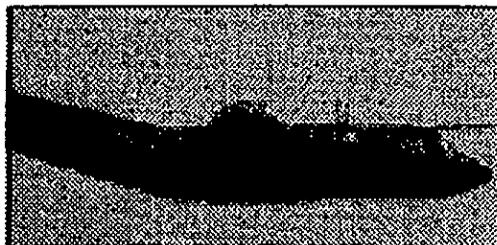
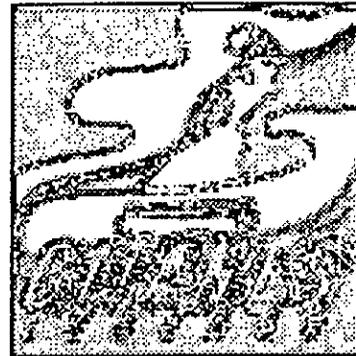
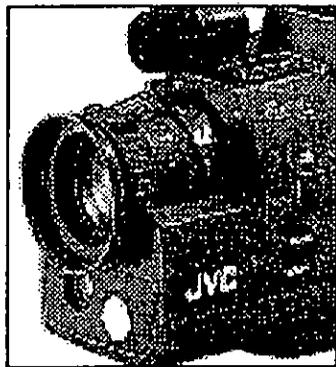


Figure 2.7. : Application de l'opérateur de Perwitt.

Les contours des objets sont plus épais, donc maquent de précision.
 Les zones de reflets sont toujours présents.

$$\text{Grad} (a) = (A.B.C.D)^{1/4}$$

où:

$$A = | f - k | + | j - g |$$

$$B = | a - p | + | m - d |$$

$$C = | b - o | + | i - h |$$

$$D = | c - n | + | e - l |$$

La fenêtre étudiée étant:

$$\begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

L'ensemble des points isolés dus aux bruits sont considérablement réduits, car si l'un au moins des 4 gradients devient faible, le produit devient alors également faible.

II.1.4.3. Opérateur Laplacien :

Le laplacien est la seconde dérivée de l'image :

$$P[L(x,y)] = \frac{d^2 L}{d x^2} + \frac{d^2 L}{d y^2}$$

$$P[L(x,y)] = [L(x+1,y) + L(x-1,y) + L(x,y+1) + L(x,y-1)] - 4.L(x,y)$$

On tire ainsi l'opérateur (ou le masque) de Laplace :

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

Vu qu'il s'agit encore d'une seconde dérivée, cet opérateur est très sensible aux bruits. Son application n'est utile que pour savoir si le pixel contour étudié se trouve du côté sombre ou éclairé de la variation de niveau.

L'application de ce masque sur quelques images est présentée dans la figure 2.8.

II.1.5. Seuillage de l'image: (binarisation) :

L'opération de seuillage est l'une des principales méthodes utilisées pour la détection d'objets. [1]

L'histogramme d'une image type est généralement constitué de deux modes principaux de niveaux de gris (figure 2.9).

On peut alors définir une valeur de seuillage S , telle que:

$$L(x,y) = \begin{cases} 1 & \text{Si } L(x,y) > T \\ 0 & \text{Si } L(x,y) < T \end{cases}$$

Une valeur optimale du seuil est la valeur minimale entre les deux pics de l'histogramme.

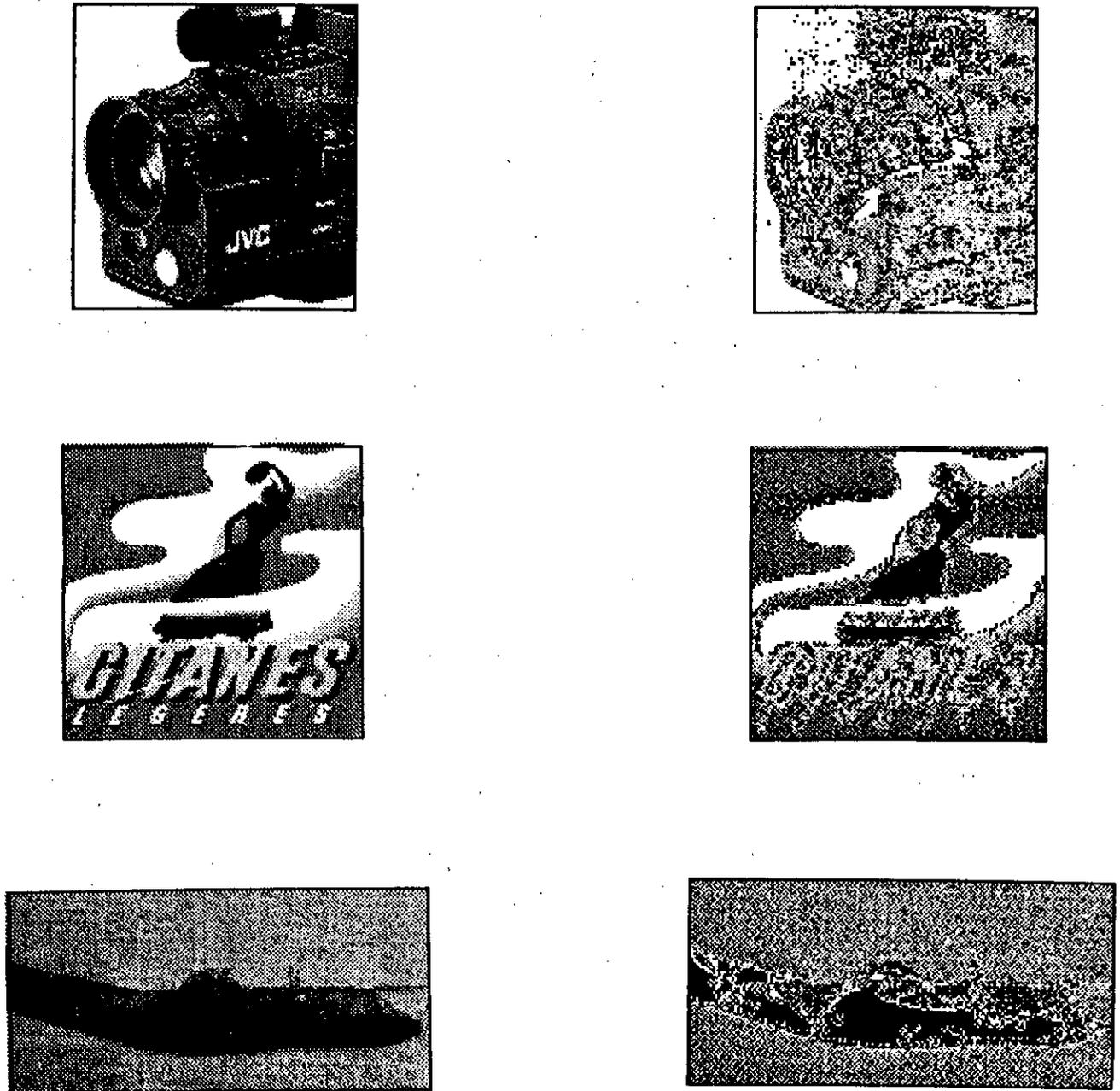


Figure 2.8. : Application de l'opérateur de Laplace.

Les contours ne sont pas apparents. Cet opérateur est très sensible aux bruits.

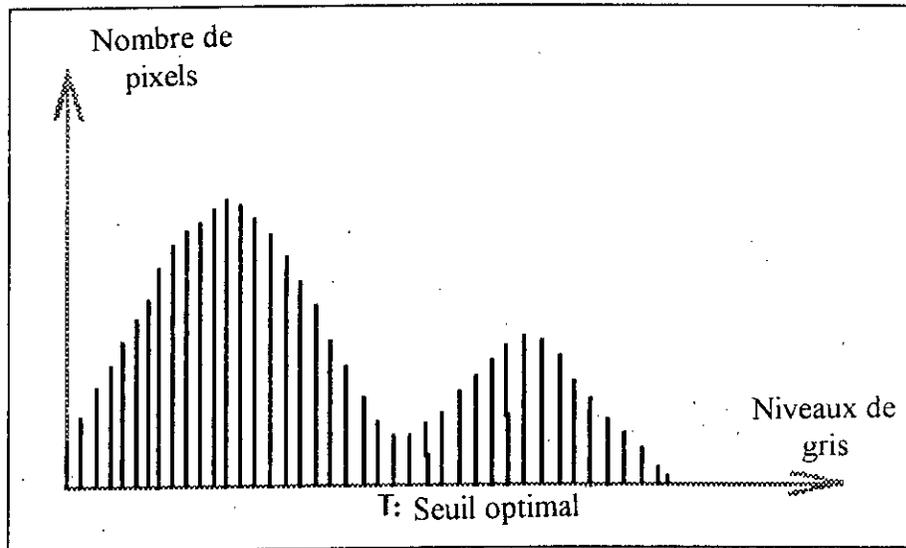


Figure 2.9.

La binarisation de l'image présente l'avantage (très important) du faible espace mémoire requis par une image binaire. En effet, l'image ne possédant que deux niveaux (0 et 1), il suffit d'un seul bit pour coder un pixel.

II.2. Traitement de haut niveau :

II.2.1. Introduction :

Les techniques de détection de contours ne font apparaître, en fait, que les points de discontinuité présents dans une image. Or ceci ne caractérise pas un contour complet (présence de bruits, interruption dans les contours,...). Ces techniques doivent alors être suivies par des procédures d'établissement de liaison, ou autres algorithmes d'extraction de contours. [1, 14].

II.2.2. Edge linking : (établissement de liaisons)

La méthode la plus utilisée afin de restituer un contour est celle de " l'analyse locale " [1]. Celle-ci opère sur un voisinage (neighborhood) généralement petit (3 x 3 ou 5 x 5) de pixels qui ont déjà été traités par les méthodes de bas niveau.

les deux principaux critères sur lesquels repose cette méthode sont l'amplitude et la direction du gradient.

On dit que le pixel (x , y) est similaire en amplitude au pixel (x' , y') si :

$$| | G[L (x , y)] | - | G[L (x' , y')] | | \leq T$$

Où T est une valeur seuil pré définie.

On dit que le pixel (x , y) a un angle similaire au pixel (x' , y') si :

$$| \arg [G (L (x , y))] - \arg [G (L (x' , y'))] | \leq \alpha$$

Ou α est une valeur seuil pré-définie.

On relie le point (x , y) au point (x' , y') faisant partie de son voisinage (neighborhood) si les deux critères d'amplitude et d'angle sont satisfaits.

La restitution du contour se fait en affectant à ses points un certain niveau de gris et un autre au reste de l'image (binarisation). [1].

Remarques sur l'application de la méthode:

Cette méthode permet de relier efficacement tous les points constituant un seul contour. Or celle-ci est basée sur le critère de similarité qui dépend des valeurs seuils

(α et T). Le choix expérimental de ces paramètres peut donc conduire à de très longs temps recherche.

II.2.3. Suivi de contours :

Ces algorithmes consistent à extraire un contour sur une image déjà seuillée. Leur principe consiste généralement à faire un balayage de l'image afin de détecter les points marquant la transition du fond sombre (background) vers l'objet (clair).

Deux algorithmes sont proposés:

II.2.3.1. L'algorithme de Duda & Bart : [14, 3, 1]

Nous le résumons par les étapes suivantes :

1. balayer l'image jusqu'à rencontre du premier point objet.
2. Tourner à gauche.
3. Tester le point suivant :
 - si c'est un point objet, tourner à gauche.
 - sinon tourner à droite.
4. s'arrêter à la rencontre du point de départ.

Remarques sur l'application de l'algorithme:

L'application de cet algorithme montre que celui-ci est très limité. En effet il ne peut faire ressortir qu'un seul contour sur une image qui peut en contenir plus.

De ce fait un second algorithme est proposé.

II.2.3.2. Algorithme de Chottera & Shridar :[19]

Cet algorithme est capable, de détecter les contours de tous les éléments de l'image (objets, trous).

Il est constitué des étapes suivantes:

1. Balayer l'image jusqu'à rencontre du premier point contour (I_1 , J_1) le point précédent est noté (I_2 , J_2).

2. A partir du point (I_2 , J_2) en tournant autour du point (I_1 , J_1) de façon à balayer ses voisins dans le sens horaire, on numérote les 7 autres voisins du point (I_1 , J_1).

3. évaluer les coordonnées $L_x (k)$ et $L_y (k)$ du k ème voisin.

4. Si le k ème voisin est un point objet, il sera le point suivant du contour et on définit (I_1 , J_1) en ce point; (I_2 , J_2) sera le ($k-1$)ème du voisinage de (I_1 , J_1).
aller à l'étape 2.

sinon, prendre $k = k+1$ et aller à l'étape 3.

6. Continuer jusqu'à rencontre du premier point contour détecté.

7. affecter à tous les contours détectés une valeur I_c différente des autres afin d'éviter la détection du même contour une autre fois.

8. Prendre $I_c = I_c + 1$, aller à l'étape 1 pour détecter de nouveaux contours.

9. s'arrêter après balayage de toute l'image.

Remarques sur l'application de l'algorithme:

Cet algorithme est meilleur que le précédent dans le sens que celui-ci est capable de détecter tous les contours sur une même image. Néanmoins le grand inconvénient de cette méthode est le temps qu'elle nécessite. En effet, celui-ci parcourt chaque contour séparément, pendant le balayage de l'image.

II.3. Conclusion :

A travers cette étude, la principale conclusion à tirer est la nécessité de la présence d'un opérateur humain pour apprécier l'état de l'image brute et définir les différentes opérations à effectuer et ceci, afin d'extraire l'information utile particulière à une application donnée.

Cela nécessite donc le passage par plusieurs étapes (voir figure 2.10), avec toutes les implications évidentes sur le temps de calcul.

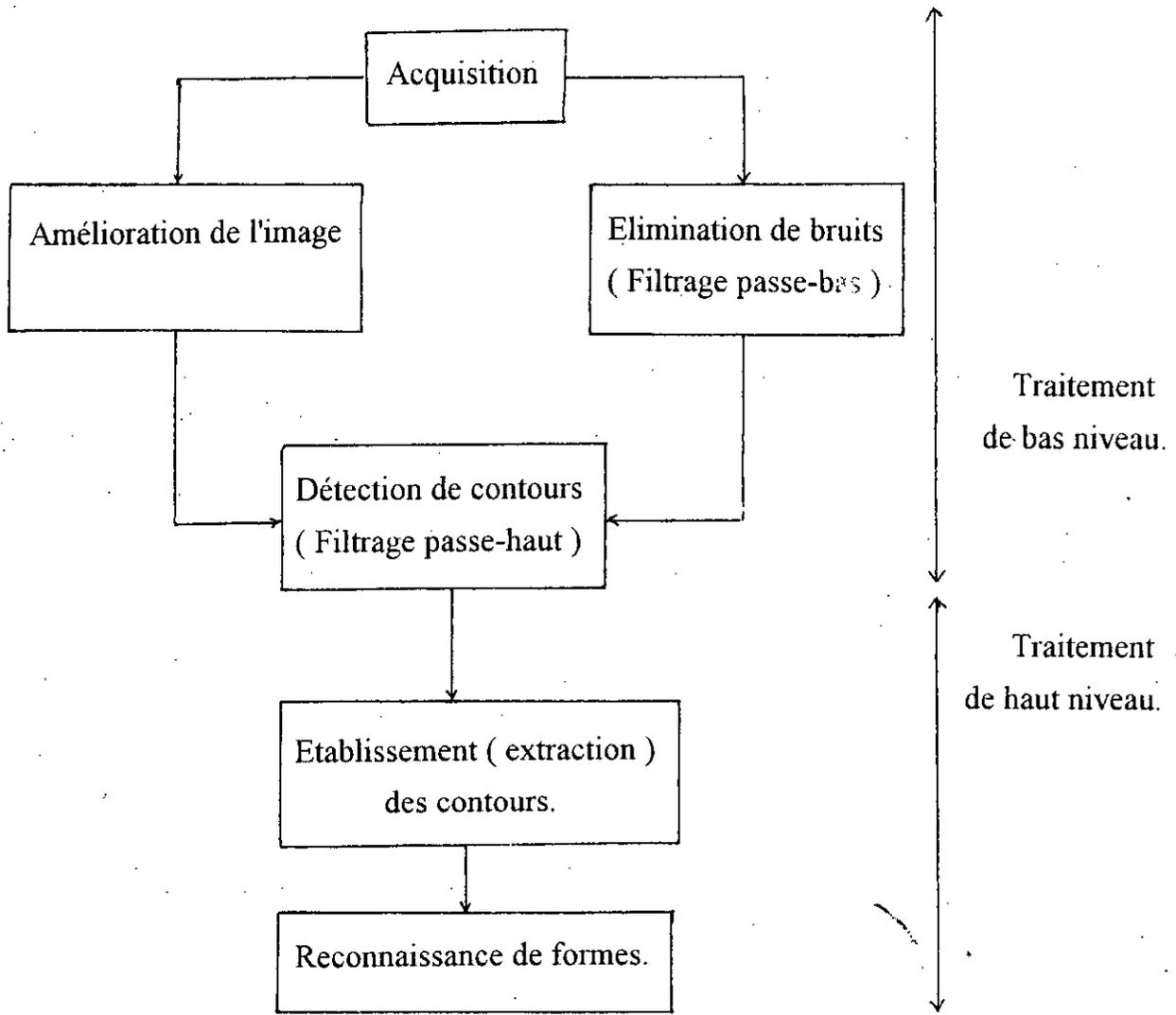


Figure 2.10.

Chapitre III :

PRESENTATION DES RESEAUX
DE NEURONES ARTIFICIELS

III.1. introduction:

Le concept de réseaux de neurones artificiels a été introduit , pour la première fois , durant les années quarante , par deux chercheurs Mc Culloch et Pitts [2, 9], afin d'essayer d'imiter les capacités de mémorisation, classification, apprentissage et généralisation du cerveau humain. L'approche de ces deux chercheurs a été de modéliser ce dernier (le cerveau) en reproduisant son constituant de base : le neurone.

Cependant, cette audacieuse tentative de simulation du cerveaux a été freinée à cause de ses limites théoriques. L'étude des réseaux de neurones n'a repris son élan qu'à partir des années quatre vingts, après l'avènement de puissants moyens mathématiques d'études, tel que l'algorithme de backpropagation, présenté en 1986 par Rumelhart, Hinton et Williams [9], qui a beaucoup contribué au regain d'intérêt pour ce domaine.

III.2. Les réseaux de neurones artificiels:

III.2.1. Neurone biologique et neurone artificiel:

Le neurone est l'élément de base du système nerveux central. Un neurone est constitué de trois éléments: [2]

- Le corps cellulaire : qui effectue toutes les transformations biochimiques nécessaires.
- Les dendrites : qui sont les récepteurs principaux du neurone pour capter les signaux qui lui parviennent.
- L'axone: qui représente le "bus" sur lequel le neurone émet ses signaux.

Le neurone formel (neurone artificiel) a été proposé par Mc Culloch et Pitts en 1943. Il s'inspire essentiellement du neurone biologique . Celui-ci fait une somme pondérée des signaux qui lui parviennent , puis s'active en appliquant à cette somme une fonction dite fonction d'activation (figure 3.1) .

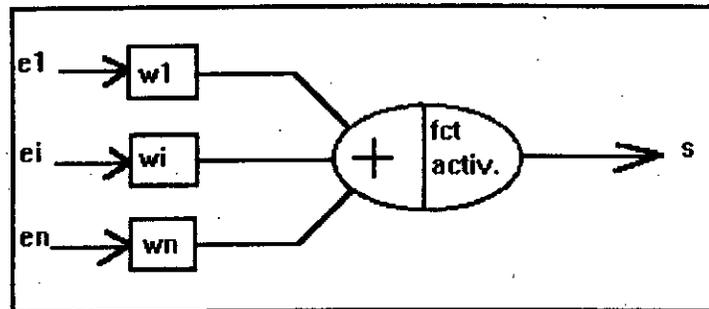


Figure 3.1.

La fonction d'activation $f(\cdot)$ peut prendre différentes formes d'un modèle à un autre, suivant l'application .

Cette fonction doit refléter le seuil ou la saturation du signal de sortie du neurone.[2]

Elle peut être une fonction sigmoïde, une fonction seuil, une fonction de heaviside, ect.

III.3. Architecture des réseaux de neurones :

On peut concevoir deux organisations principales de neurones :

-Les réseaux entièrement connectés ou chaque neurone reçoit l'information de la part de tous les autres neurones , la traite et leur renvoie sa sortie . Ainsi , tous les

neurones communiquent entre eux .

-Les réseaux multicouches où les neurones sont organisés en couches. Au sein d'une même couche , les neurones ne communiquent pas entre eux . Il reçoivent l'information de la couche précédente, la traite et la renvoie à la couche immédiatement suivante. L'information ne circule donc que dans un seul sens, de l'entrée vers la sortie .

S'inspirant du système visuel qui lui même organisé en couche [2], c'est ce dernier modèle qui sera utilisé dans le présent travail.

III.4. Apprentissage des réseaux de neurones:

L'apprentissage d'un réseau de neurone consiste à faire varier ses poids synaptiques , afin de le faire évoluer vers un état où chaque information en entrée produit la sortie désirée. La méthode d'apprentissage la plus courante est l'apprentissage supervisé qui est basé sur la présentation au réseau d'un ensemble d'exemple représentant chacun une entrée et la sortie correspondante. L'apprentissage consiste donc à minimiser l'erreur entre la réponse du réseau et la sortie désirée pour les couples (exemples) d'entraînement. Cette minimisation se fera en agissant sur les poids.

Se basant sur des méthodes mathématique de minimisation de fonction, plusieurs techniques d'entraînement ont été développées .

Nous présentant ci-dessous deux algorithmes d'entraînement :

III.4.1. Backpropagation: [4, 10, 2]

Présenté pour la première fois en 1986 par Rumelhart , Hinton et Williams [9, 2], cet algorithme a donné de très bon résultats et reste la méthode d'entraînement la plus usitée. Cette technique a pour but de minimiser l'erreur quadratique entre la sortie désirée et celle du réseau en se basant sur la méthode de "descente de gradient".

Les étapes de cet algorithme sont les suivantes:

1. Initialiser aléatoirement les poids du réseau .
2. Injecter le vecteur d'entrée X_p .
3. Calculer les entrées des neurones des couches internes :

$$Z_{ip}(W) = \sum_j W_{ij} \cdot Y_{jp}$$

4. Calculer les sorties des neurones des couches internes :

$$X_{ip} = f_i(Z_{ip}(W))$$

5. Calculer les sorties du réseaux :

$$\overline{X}_{kp}(W) = f_k(Z_{kp}(W))$$

6. Calculer les termes d'erreurs de la couche de sortie :

$$\delta_{kp} = -f'_k(Z_{kp}(W)) \cdot (\overline{X}_{kp}(W) - d_{kp})$$

7. Calculer les terme d'erreur des couches internes :

$$\delta_{ip} = f'_i(Z_{ip}(W)) \cdot \sum_k \delta_{kp} \cdot W_{ki}$$

8. Changer les poids du réseaux :

$$\Delta W_{ki} = \eta \cdot \delta_{kp} \cdot Y_{ip}(W) \quad \text{pour la couche de sortie.}$$

$$\Delta W_{ij} = \eta \cdot \delta_{ip} \cdot Y_{jp}(W) \quad \text{pour les couches internes.}$$

où η est le pas d'entraînement.

9. Refaire les étapes (2 à 8) pour chaque couple d'entraînement.

10. Calculer l'erreur globale :

$$E(W) = 0,5 \cdot \sum_k (\overline{X_{kp}} - d_{kp})^2$$

11. Refaire les étapes (2 à 11) tant que $E(W)$ n'a pas atteint le seuil fixé.

Remarque sur la méthode :

- Théoriquement, le pas d'entraînement doit être très petit. Or ceci nécessiterait un temps de calcul trop élevé. Un pas assez grand peut conduire à des oscillations compromettant ainsi la convergence du réseau.

- D'autre part, dès que le réseau se stabilise dans un minimum, il ne peut plus en sortir. Il y a donc risque que celui-ci converge vers un minimum local.

III.4.2. La méthode d'optimisation aléatoire : (M.O.A) [4, 5, 6]

La technique d'optimisation aléatoire proposée par Matys en 1961 assure une convergence vers un minimum global avec une probabilité égale à 1 [4, 5].

En 1981, Solis & Wets [5] se sont inspirés de cette méthode pour développer un algorithme de minimisation d'une fonction $f(x)$.

Les étapes de cet algorithme sont les suivantes :

1. Initialiser aléatoirement le vecteur x à $x(0)$, mettre $b(0) = 0$, donner une valeur à v , fixer le nombre d'itérations M .

2. Générer un vecteur aléatoire gaussien $\xi(k)$ de moyenne $b(k)$ et de variance v

3. Si : $f(x(k) + \xi(k)) < f(x(k))$

alors : $x(k+1) = x(k) + \xi(k)$

$b(k+1) = 0.4 \cdot x(k) + 0.2 \cdot b(k)$

Si : $f(x(k) + \xi(k)) \geq f(x(k))$ et $f(x(k) - \xi(k)) < f(x(k))$

alors : $x(k+1) = x(k) - \xi(k)$

$b(k+1) = x(k) - 0.4 \cdot \xi(k)$

Sinon : $x(k+1) = x(k)$

$b(k+1) = 0.5 \cdot b(k)$

4. Refaire les étapes (2 à 3) tant que $k < M$.

En 1988, Norio Baba [4] a proposé l'application de cette technique pour l'apprentissage des réseaux de neurones artificiels. La fonction à minimiser dans ce cas est l'erreur globale :

$$E(W) = 0.5 \sum_k (y_{kp} - d_{kp})^2$$

Les étapes de cet algorithme sont les suivantes :

1. Initialiser aléatoirement le vecteur W à $W(0)$, mettre $b(0) = 0$, donner une valeur à v , fixer le nombre d'itérations M .

2. Générer un vecteur aléatoire gaussien $\xi(k)$ de moyenne $b(k)$ et de variance v

3. Calculer $E(W(k))$ et $E(W(k) + \xi(k))$

Si : $f(W(k) + \xi(k)) < E(W(k))$

alors : $W(k+1) = W(k) + \xi(k)$

$b(k+1) = 0.4 \cdot W(k) + 0.2 \cdot b(k)$

Sinon : calculer $E(W(k) - \xi(k))$

Si : $E(W(k) + \xi(k)) \geq E(W(k))$ et $E(W(k) - \xi(k)) < E(W(k))$

alors : $W(k+1) = W(k) - \xi(k)$

$b(k+1) = W(k) - 0.4 \cdot \xi(k)$

Sinon : $W(k+1) = W(k)$

$b(k+1) = 0.5 \cdot b(k)$

4. Refaire les étapes (2 à 3) tant que $k < M$.

Chapitre VI :

TRAITEMENT D'IMAGES PAR
LES RESEAUX DE NEURONES

IV.1. Introduction :

A travers l'étude qui va suivre, nous allons essayer d'utiliser les capacités des réseaux de neurones pour le traitement d'images, et particulièrement pour la segmentation ou l'extraction de contours à partir d'images brutes.

Nous proposerons également des applications importantes pour l'établissement d'un système complet de robot-vision

IV.2. Extraction des contours :

IV.2.1. Objectif :

L'objectif de cette partie est la détection et la restitution des contours de tous les éléments existant sur une image. Pour cela, nous avons choisi de faire déplacer un réseau sur une image de dimension quelconque pour faire ressortir les contours.

IV.2.2. Elaboration du réseau :

L'approche que nous avons élaborée est celle du masque centré sur un pixel central.

La couche d'entrée du réseau est une matrice (3 x 3) représentant la portion d'image couverte (voir figure 4.1.). La couche de sortie est constituée d'un seul neurone représentant le pixel central de la matrice d'entrée, dont l'appartenance à un contour dépendra de l'état de ces pixels voisins (neighborhood).

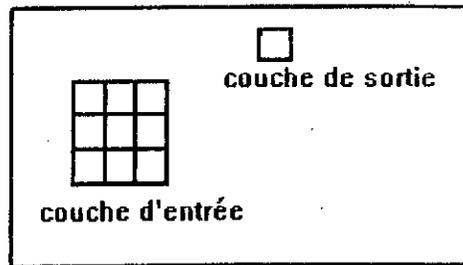


Figure 4.1.

Un " 1 " en sortie signifiera que ce dernier fait partie d'un contour, un " 0 " indiquera le contraire.

L'entraînement du réseau est effectué à l'aide d'exemples traduisant les possibilités de contours (rectilignes, ronds, pointus,...).

Les résultats de l'apprentissage sont résumés dans le tableau 4.1.

Architecture de réseau	Méthode utilisée	Résultats
9 - 7 - 5 - 1 26 couples d'entraînement	M.O.A. $v = 0,1$	$k = 1670$ (nombre d'itérations) erreur = 0,001
9 - 7 - 5 - 1 26 couples d'entraînement	B.P. $\eta = 0,1$	$k = 3000$ erreur = 0,01

Tableau 4. 1.

Nous constatons que la M.O.A. donne de meilleurs résultats.

D'autre part, les essais de généralisation n'ont pas été satisfaisants. Ce qui nous a conduit à augmenter le nombre d'exemples d'entraînement pour améliorer les performances du réseau.

Nous avons continué cet apprentissage avec la M.O.A.

Les résultats de ce second entraînement sont présentés dans le tableau 4.2.

Architecture de réseau	Méthode utilisée	Résultats
9 - 7 - 5 - 1 40 couples entraînement	M.O.A. $v = 1$	$k = 2000$ erreur = 0,000165

Tableau 4.2.

IV.2.3. Applications :

Afin d'apprécier les performances de réseau nous l'avons testé sur différents types d'images.

IV.2.3.1. Application sur des images binaires générées par ordinateur:

Les premiers tests ont été effectués sur des images que nous avons synthétisées par micro ordinateur. Ces images représentent des objets couleurs placés dans un background nul.

La figure 4.2 présente les résultats obtenus pour quelques images.

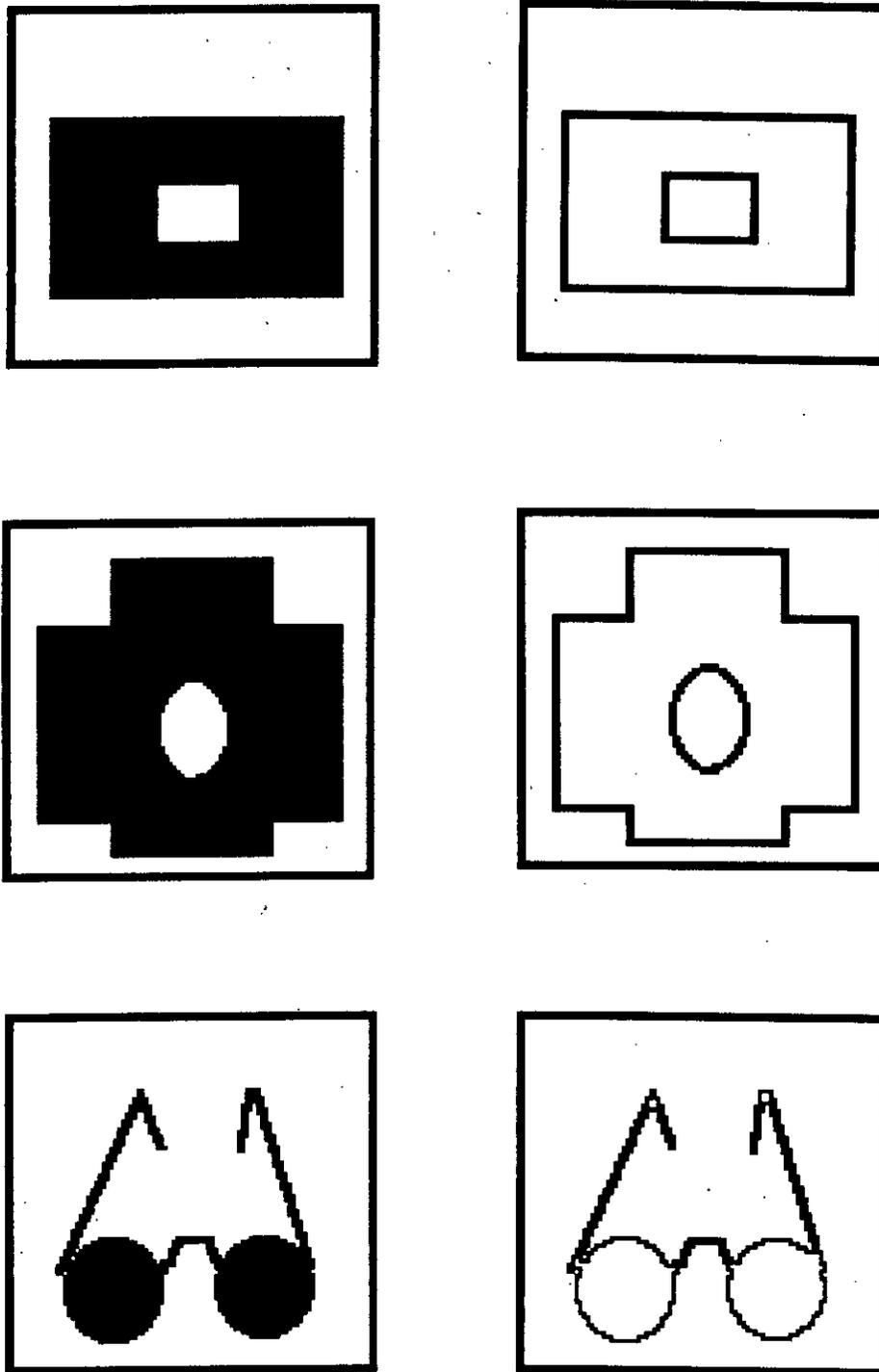


Figure 4.2. : Application du réseau sur des images binaires générées par ordinateur

Nous constatons que le réseau est arrivé à extraire le contour de tous les objets avec un temps relativement court.

IV.2.3.2. Application sur des images binaires bruitées générées par ordinateur:

Les tests conduits sur des images bruitées ont montré que le réseau est robuste et élimine celles-ci.

Les résultats de l'application des images incrustées de bruits aléatoires sont présentés dans la figure 4.3.

IV.2.3.3. Application sur des images multiniveaux générées par ordinateur:

Les images testées ici représentent des objets de différentes couleurs placés dans un background de couleur différente.

Les résultats de ces tests sont présentés dans les figures 4.4, 4.5, et 4.6.

Le réseau est arrivé à isoler le contour de chaque objet du background. Il s'est encore une fois révélé très efficace.

IV.2.3.4. Application sur des images multiniveaux bruitées générées par ordinateur :

L'application du réseau sur des images multiniveaux bruitées a confirmé son efficacité à éliminer celles-ci. Le bruit est représenté par un ensemble de pixels, dont le niveau de gris est différent par rapport au voisinage environnant (neighborhood).

La figure 4.7 représente les images bruitées et les images de sortie du réseau.

On remarque bien que le réseau fait ressortir le contour tout en éliminant les bruits.

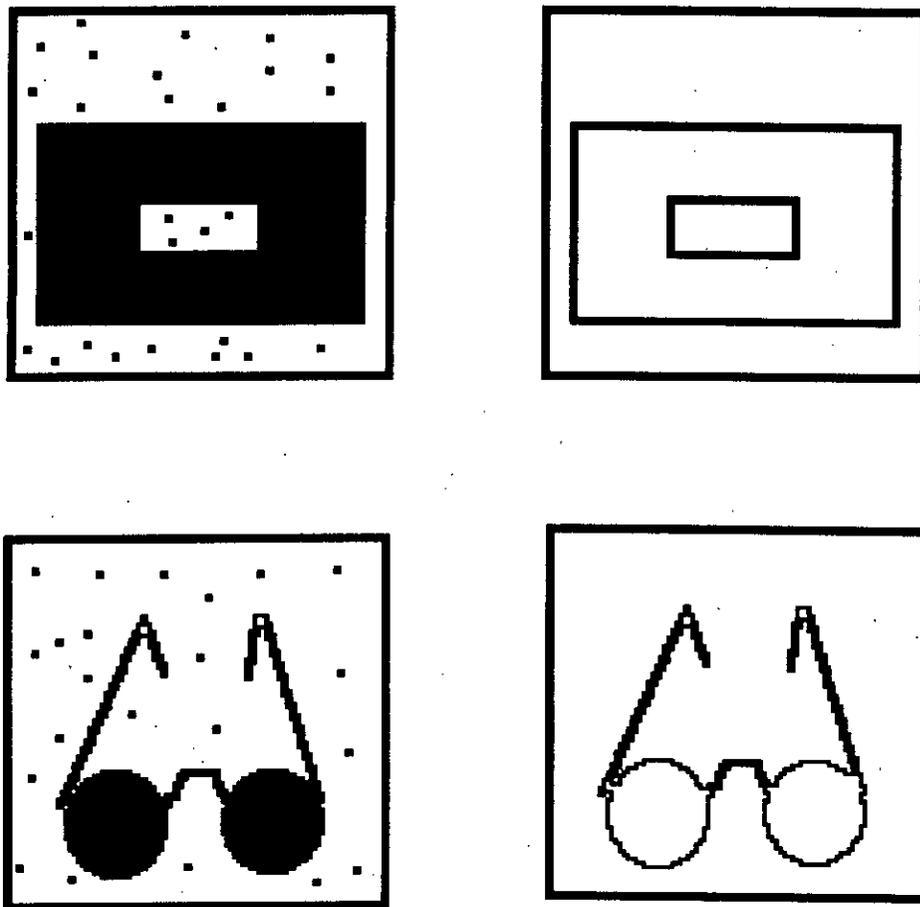


Figure 4.3. : Application du réseau sur des images binaires bruitées générées par ordinateur

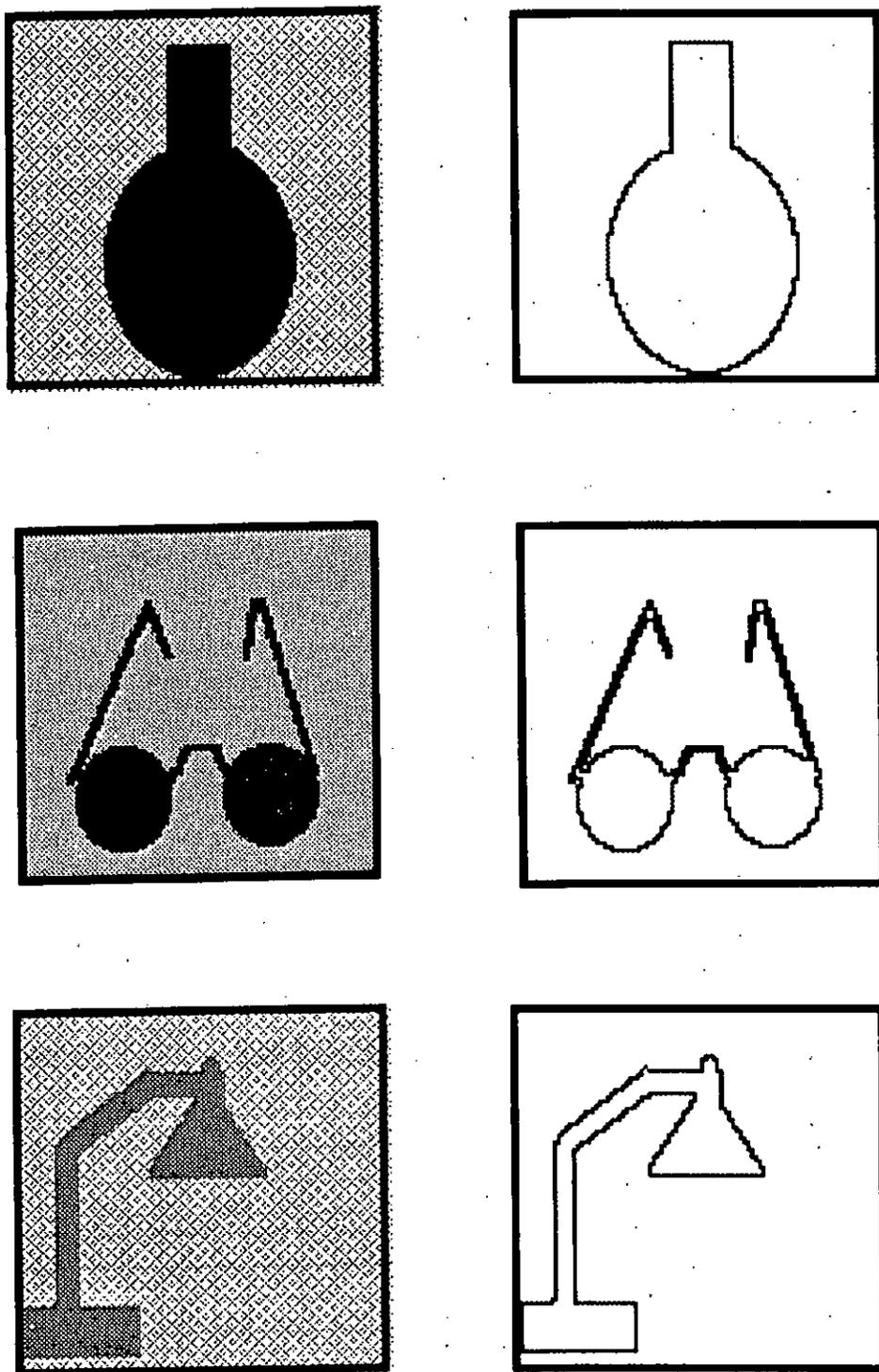


Figure 4.4. : Application du réseau sur des images multiniveaux générées par ordinateur

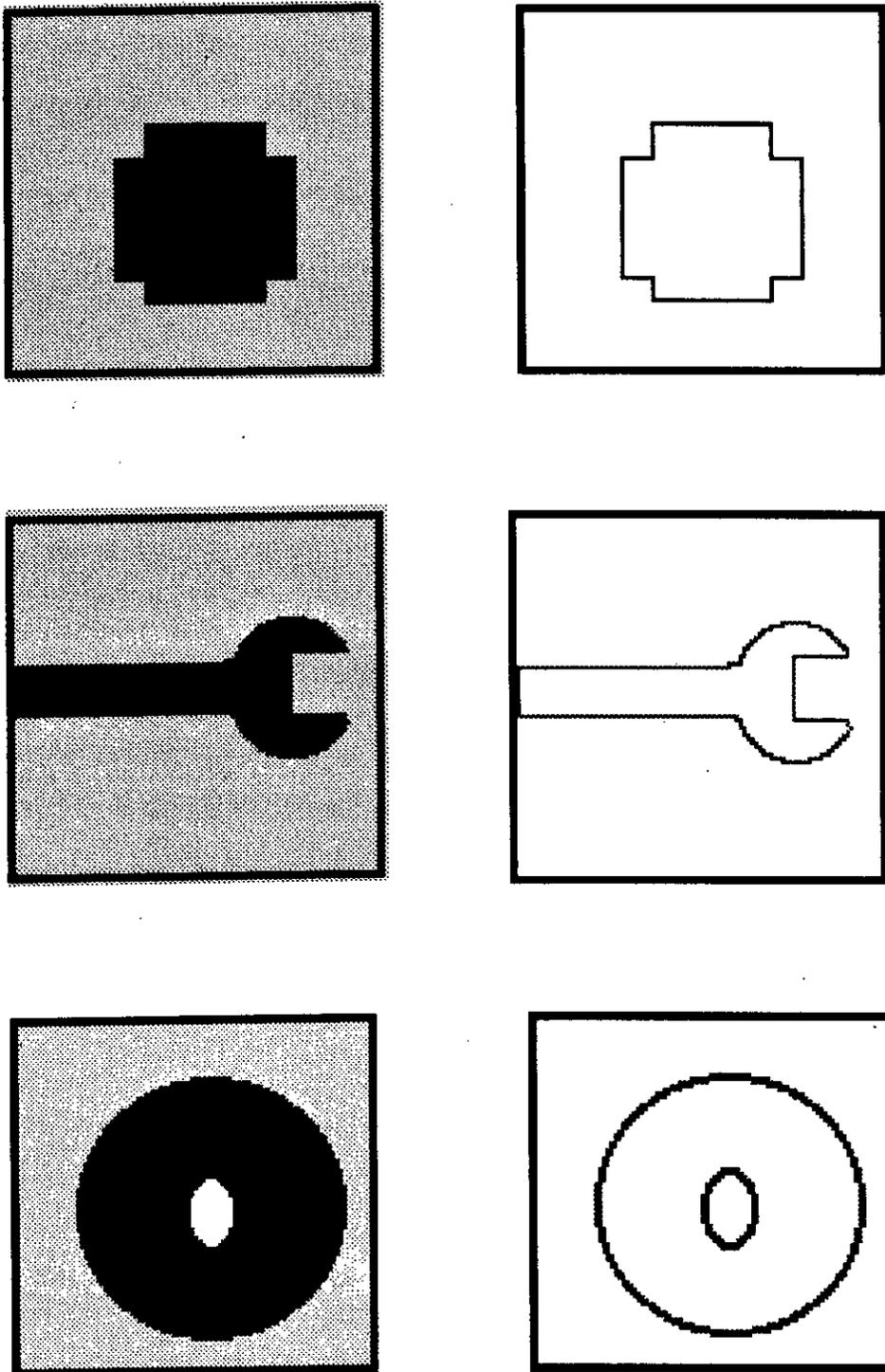


Figure 4.5. : Application du réseau sur des images multiniveaux générées par ordinateur

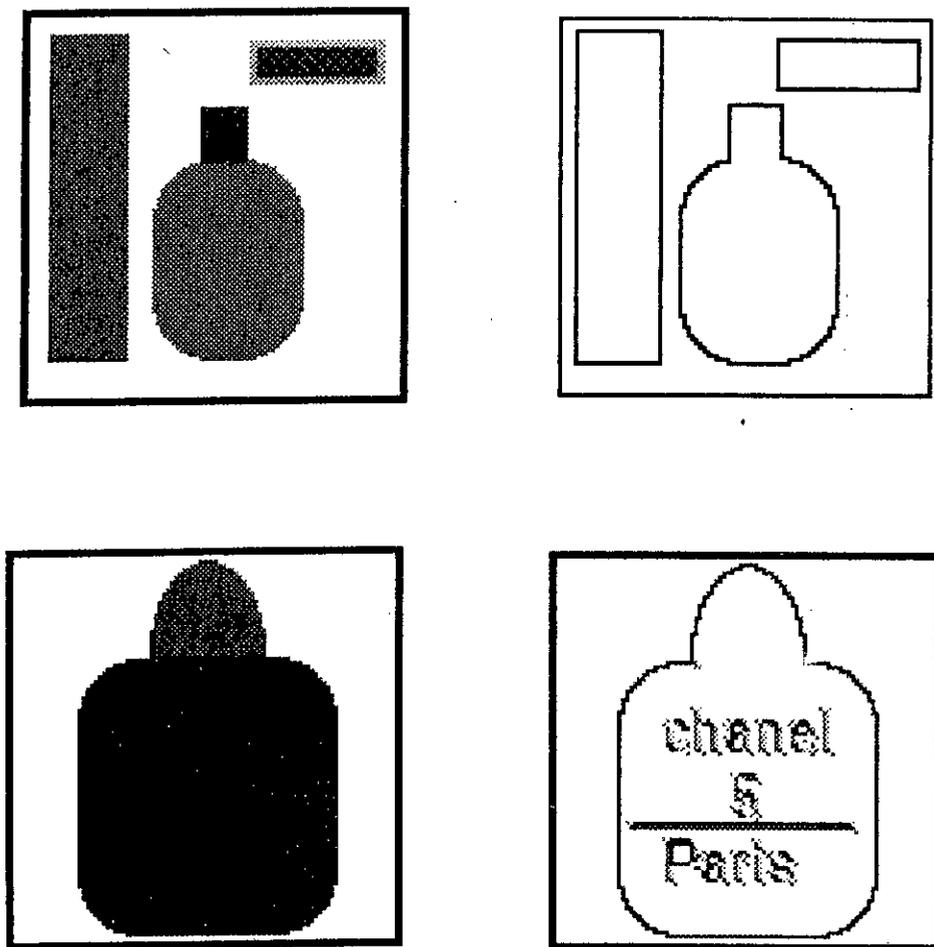


Figure 4.6. : Application du réseau sur des images multiveaux générées par ordinateur

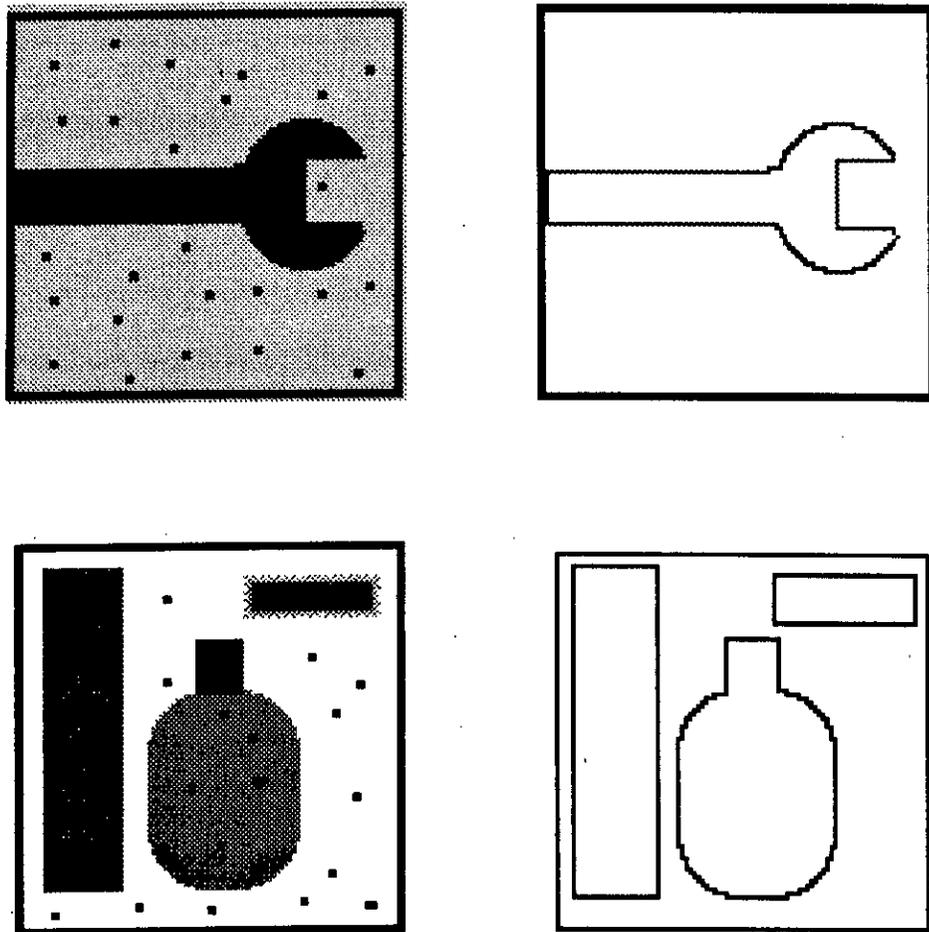


Figure 4.7. : Application du réseau sur des images multiniveaux bruitées générées par ordinateur

IV.2.3.5. Application sur des images réelles obtenues par scanner :

Vu les performances du réseau vis à vis des tests précédents, nous avons présenté au réseau des images réelles, obtenues à partir du scanner Scan Jet.

Les résultats et commentaires sont réunis dans les figures 4.8 et 4.9.

IV.3. Reconnaissance de formes:

On se propose dans cette partie d'exploiter les capacités de mémorisation et de généralisation des réseaux de neurones pour classer un certain nombre d'objets suivant leurs formes.

Les formes à reconnaître dans le cas du présent travail seront les formes industrielles primaires :

- circulaires.
- rectangulaires.
- hexagonales.

Un premier réseau a été élaboré pour dissocier les formes circulaires des formes rectangulaires.

L'entrée de ce réseau est une image de dimensions (11 x 11) représentant un contour déjà extrait à l'aide du réseau précédent.

La sortie est constituée de deux neurones représentant les deux formes à identifier.

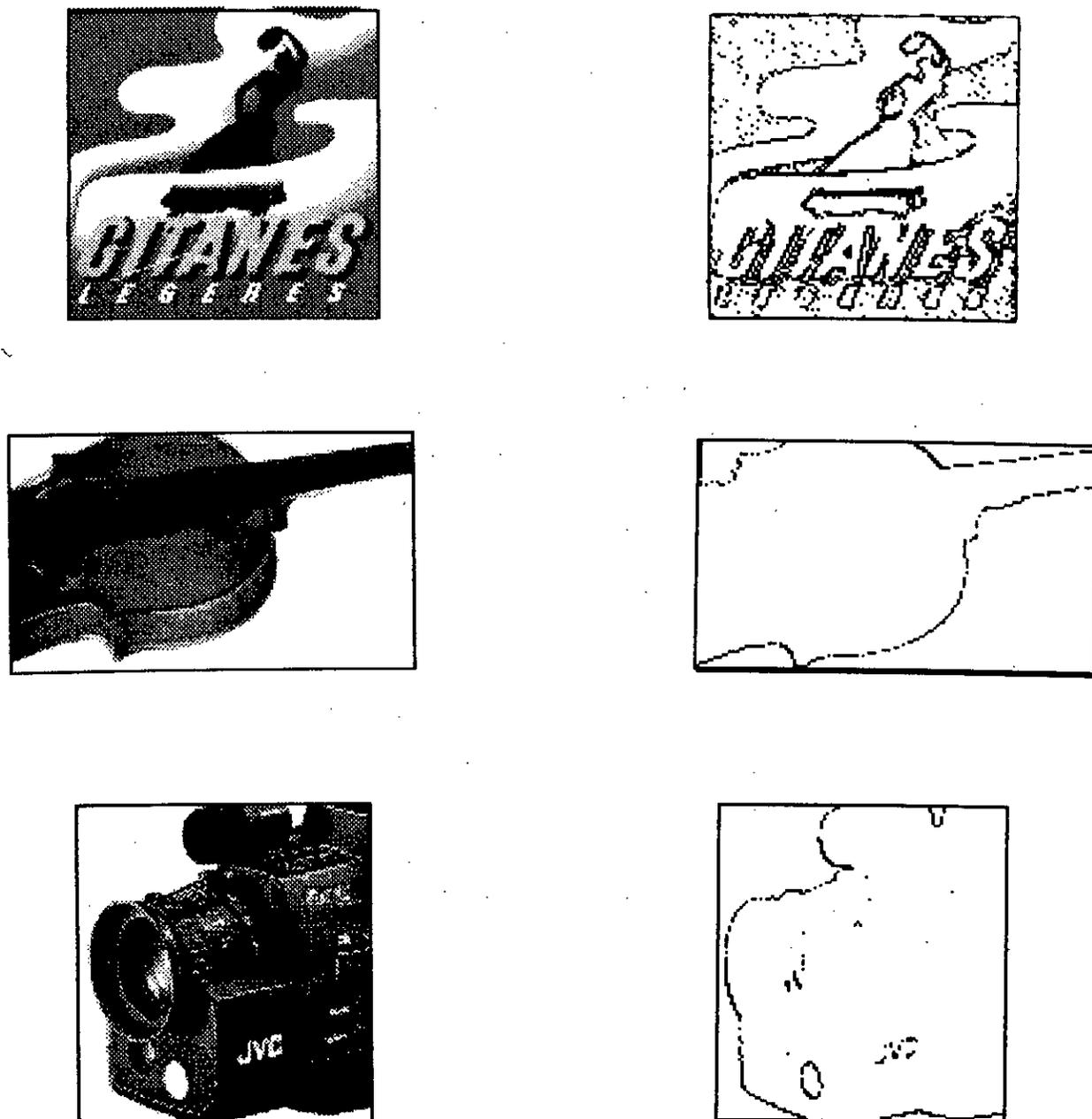


Figure 4.8. : Application du réseau sur des images réelles obtenues par scanner.

Les contours extraits par le réseau sont meilleurs que ceux extraits par les méthodes classiques. Le réseau donne en sortie une image relativement nette, même dans les zones à très faible contraste. Ces images doivent néanmoins subir des opérations de suivi de contours, pour une exploitation future.

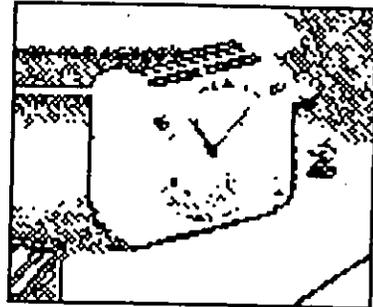
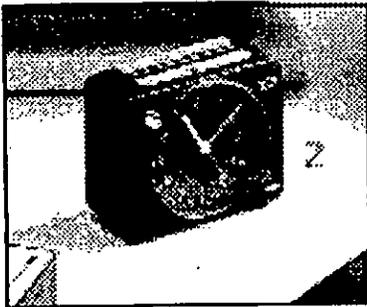


Figure 4.9. : Application du réseau sur des images réelles obtenues par scanner.

Les phénomènes de reflets, d'ombres, ou d'un mauvais éclairage n'ont pas eu la même influence sur le réseau que sur les méthodes classiques. Le réseau se révèle robuste vis à vis de ces problèmes qui peuvent fausser l'extraction du contour exact.

L'entraînement du réseau est effectué à l'aide de contours de différentes dimensions. Le résultat obtenu est le suivant :

Architecture de réseau	Méthode utilisée	Résultats
121 - 8 - 8 - 2	M.O.A. $v = 0,1$ à $0,000001$	$k = 6000$ erreur = $4,21.e-10$

Tableau 4.3.

Ce réseau fût testé sur d'autres objets, décrivant des formes rectangulaires et circulaires de dimensions différentes. Nous avons constaté que celui-ci différencie entre les deux formes avec une très bonne précision.

Ce résultat nous a amené à concevoir un autre réseau qui reconnaîtra, en plus des deux formes précédentes, la forme hexagonale.

L'entrée de ce réseau est une matrice (17 x 17) représentant le contour d'une forme quelconque.

La sortie est constituée de 3 neurones représentant les trois formes à dissocier.

Les résultats de l'apprentissage figurent dans le tableau 4.4.

Architecture du réseau	Méthode utilisée	Résultats
289 - 10 - 10 - 3	M.O.A. v = 0.1 à 0.000001	k = 13250 erreur = 0.00001235

Tableau 4.4.

Les tests de généralisation ont été encore une fois très satisfaisants et surtout très prometteurs pour d'autres applications, telle que la reconnaissance de caractères.

Les réseaux de neurones ont confirmé leurs très fortes capacités de mémorisation et de généralisation.

IV.4. Détection des axes :

La détection des axes d'un objet consiste en la détermination de l'angle de rotation de l'axe principal par rapport à l'axe horizontal.

Cette information est largement utilisée en robot-vision. C'est en effet la principale information qui permet à un bras de robot de se saisir d'un objet après l'avoir repéré, à partir de son image vidéo (figure 4.10).

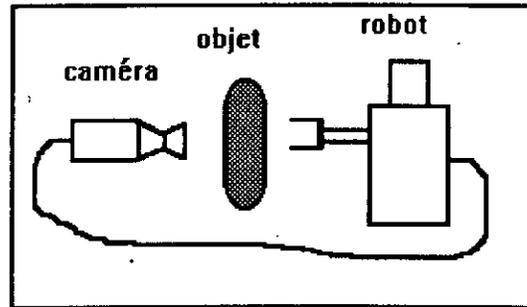


Figure 4.10.

L'objet que nous avons pris comme base d'étude est de forme rectangulaire de dimensions finies.

L'entrée du réseau est une matrice (16×16) contenant le contour de l'objet centré.
La couche de sortie est constituée de deux neurones représentant le sinus et le cosinus de l'angle de rotation.

Entraînement est effectuée à l'aide de 12 exemples représentant l'inclinaison de l'objet avec un incrément de 15° .

Remarque :

L'objet étudié dans ce cas étant d'une forme symétrique, la plage de variation de l'angle de rotation est donc comprise entre 0° et 180° .

Le résultat de l'apprentissage est résumé dans le tableau 4.5 :

Architecture du réseau	Méthode utilisée	Résultats
256 - 4 - 2 12 couples entraînement.	B.P. n=1	k = 5000 erreur = 0.0000713

Tableau 4.5.

Les tests de généralisation, présentés dans le tableau ci-dessous ont donné une erreur maximale (entre la l'angle de sortie du réseau et l'angle désiré) de 7°.

Ceci constitue un bon résultat, car cette erreur est en fait inhérente à la précision de l'image dont la tolérance (vu les dimensions de l'objet étudié) peut se calculer facilement. Celle-ci est de 8.5°.

Nous avons donc un système qui travaille avec une " sub-pixel precision ". [17]

Comparé à cette valeur de tolérance, nous pouvons dire que notre réseau a fourni de très bonnes performances.

Angle désiré	Angle à la sortie du réseau	Erreur
20°	18.23°	1.77°
37°	30.15°	6.85°
63°	60.38°	2.62
97°	98.09°	1.09°
127°	134.01°	7.01°

VI.5. Sélection d'objets couleurs :

L'objectif dans cette partie est la sélection d'objets suivant leurs couleurs. Une telle opération est appliquée dans le cas où l'on veut isoler un objet d'une certaine couleur, par rapport à d'autres ayant des couleurs différentes.

L'entrée du réseau est composé de onze neurones; trois d'entre eux représentent le code RGB du pixel image, les huit autres représentent les couleurs de sélection.

La sortie est constituée d'un seul neurone qui sera activé dans le cas où la couleur du pixel en entrée correspond à celle sélectionnée. L'image en sortie ne fera apparaître que l'objet dont la couleur est sélectionnée.

Remarques :

1. l'image d'entrée de ce réseau doit être préalablement filtrée par les filtres primaires RGB.

2. Les couleurs de sélection sont les huit couleurs principales. Leurs codes RGB sont indiqués sur le tableau suivant : [8]

Couleur	Code RGB		
	R	G	B
Noir	0	0	0
Bleu	0	0	1
Vert	0	1	0
Cyan	0	1	1
Rouge	1	0	0
Pourpre	1	0	1
Jaune	1	1	0
Blanc	1	1	1

Les résultats de l'apprentissage du réseau sont présentés sur le tableau 4.6

Architecture du réseau	Méthode utilisée	Résultats
11 - 10 - 5 - 1	M.O.A. $v = 0.1$	$k = 5000$ erreur = 0.000325

Tableau 4.6.

Cette application trouve toute son importance en robot-vision, ainsi que dans tous domaines où il est nécessaire de sélectionner des détails suivant leurs couleurs, notamment dans le domaine médical, géologique, météorologique, ect.

VI.6. Conclusion :

Il apparaît à travers les résultats obtenus que le réseau destiné à l'extraction de contours a réussi à remplacer plusieurs fonctions classiques, en passant de la vision de bas-niveau à celle de haut niveau.

Ainsi, à partir d'une image multiniveaux, ce réseau délivre directement les contours d'objets présents sur cette image. D'autre part, ce réseaux a fourni des résultats satisfaisants sur des images bruitées ou à luminance variable, en arrivant à extraire les contours d'une manière très visible.

Ces opérations sont effectué en un temps relativement meilleur. En effet, le traitement a duré, parfois, moins d'une demi seconde.

En outre, les réseaux de reconnaissance de formes, de détection des axes et de couleurs ont prouvé la capacité des réseaux de neurones à mémoriser, classifier et à généraliser.

Chapitre V :

COMPARAISON ENTRE LA VISION
CLASSIQUE ET LA VISION NEURALE

V.1. Introduction :

Après avoir étudié le traitement d'images par les méthodes classiques et par la méthode neurale, nous proposons dans cette partie de tester les capacités et limites de chaque technique.

Nous proposerons à chaque méthode des exemples significatifs par la qualité de l'image, présentant tous les problèmes connus dans le domaine de l'imagerie pour l'extraction des contours.

Nous pourrons par la suite faire une comparaison entre les deux approches.

V.2. Présentation des exemples :

Exemple 1 : (figure 5.1)

La première image représente un objet (réveil) ayant plusieurs petits détails, et placé dans un environnement qui contient d'autres objets. L'image est peu contrastée et présente des zones d'ombre.

Exemple 2 : (figure 5.2)

Le second objet (violon) est d'une bonne qualité visuelle, mais il présente cependant quelques reflets de lumière.

Exemple 3 : (figure 5.3)

Le troisième exemple (avion) a été acquis dans de très mauvaises conditions. On constate que l'image est très sombre, le contraste est par conséquent très faible, et de plus, il y a présence de reflets (mauvaise répartition de la lumière).

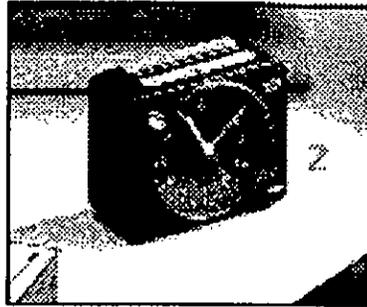
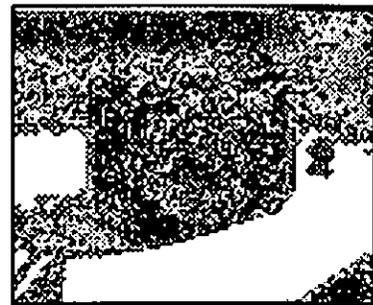


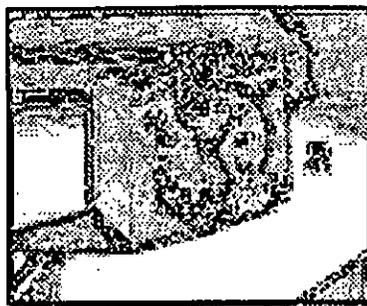
Image initiale



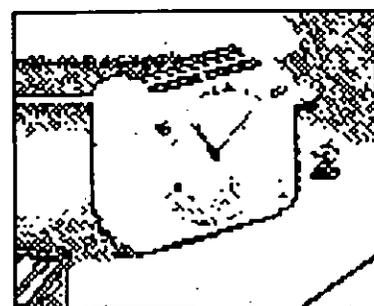
Sobel



Laplace



Prewitt



R.N.

Figure 5.1. : Exemple 1.

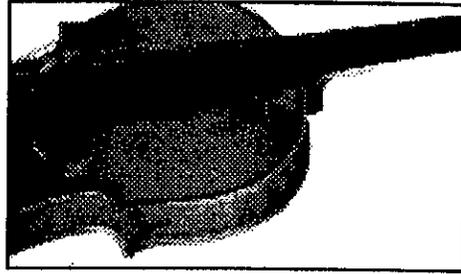
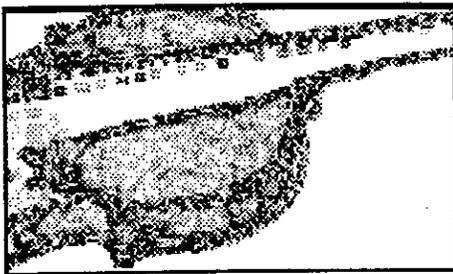
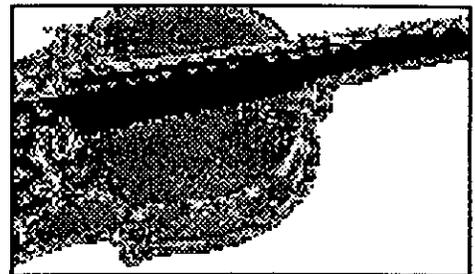


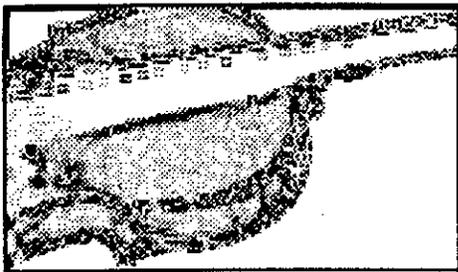
Image initiale



Sobel



Laplace



Prewitt



R.N.

Figure 5.2. : Exemple 2.

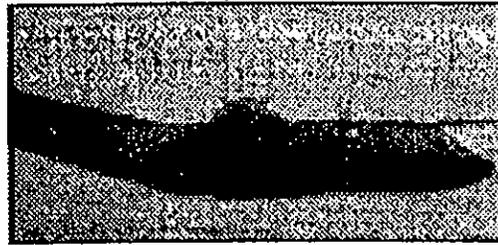
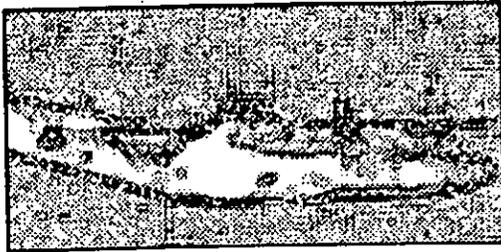


Image initiale



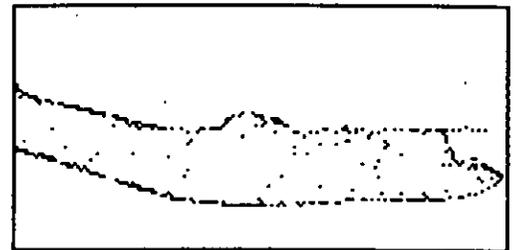
Sobel



Laplace



Prewitt

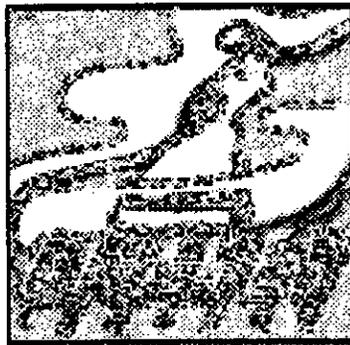


R.N.

Figure 5.3. : Exemple 3.



Image initiale



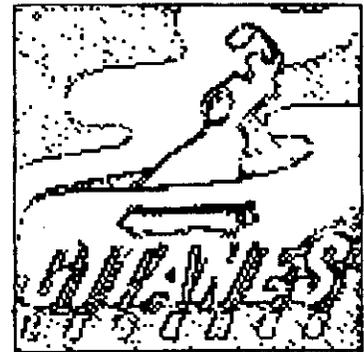
Sobel



Laplace



Prewitt



R.N.

Figure 5.4. : Exemple 4.

Exemple 4 : (figure 5.4)

La dernière image (boîte de cigarette) est une image de bonne qualité, ayant un bon contraste, mais aussi des détails fins proches l'un de l'autre.

V.3. Interprétation des résultats et comparaison :

Nous constatons en premier que les images issues de l'application des opérateurs classiques présentent des contours trop épais, donc manquant de précision.

D'autre part, nous remarquons que dans les endroits peu contrastés, les images en traitées donnent des zones confuses, ne pouvant être directement exploitées.

En outre, les zones d'ombre ou de reflet rendent parfois la qualité de l'image en sortie beaucoup trop mauvaise.

Concernant les images traitées par la méthode neurale, nous voyons très bien que la qualité de l'image est nettement meilleure.

Ainsi sur les images de qualité médiocre (exemple 3), le réseau a donné en sortie des résultats relativement meilleurs que ceux donnés par les techniques classiques.

Néanmoins, nous pensons qu'un suivi de contours peut être utile dans certains cas.

V.4. Conclusion :

A travers les résultats, nous constatons que les phénomènes de mauvaise répartition de la lumière, de reflet, de présence de zone d'ombre ou de mauvais contraste, nécessitent plusieurs opérations en traitement d'image classique avant d'effectuer la détection des contours. Par contre, le réseau est arrivé à faire l'extraction de ces contours à partir de l'image brute, écartant ainsi un grand nombre de ces problèmes. Les cas où l'effet de ces phénomènes est apparent peuvent être suivis par une opération de suivi de contours.

CONCLUSION GENERALE

CONCLUSION GENERALE

Au terme de cette étude, nous sommes arrivés aux résultats suivants:

- L'approche neurale peut remplacer efficacement plusieurs opérateurs classiques, notamment:
 - Le filtrage passe-bas.
 - Le filtrage passe-haut
 - La segmentation.

Elle passe, donc, de la vision de bas niveau à celle de haut niveau, avec tous les avantages de rapidité d'exécution associées au réseaux de neurones.

Ce résultat trouve toute son importance dans les applications en temps réel, notamment les systèmes d'inspection d'objets industriels sur les chaînes de production automatiques à hautes vitesses.

-L'efficacité du réseau entraîné en reconnaissance de formes ouvre la voie à plusieurs applications telles que la reconnaissance de caractères. Ce réseaux est à la base de tout système de commande intelligente ayant une capacité visuelle.

-Les réseaux déjà construits peuvent être assemblés pour former un système complet de robot-vision comportant :

- L'extraction de contours.
- La détection des axes.
- La reconnaissance de formes .

ce système peut être réaliser dans un circuit VLSI, intégré dans le système de vision d'un robot qui pourra remplir parfaitement ces fonctions en temps réel.

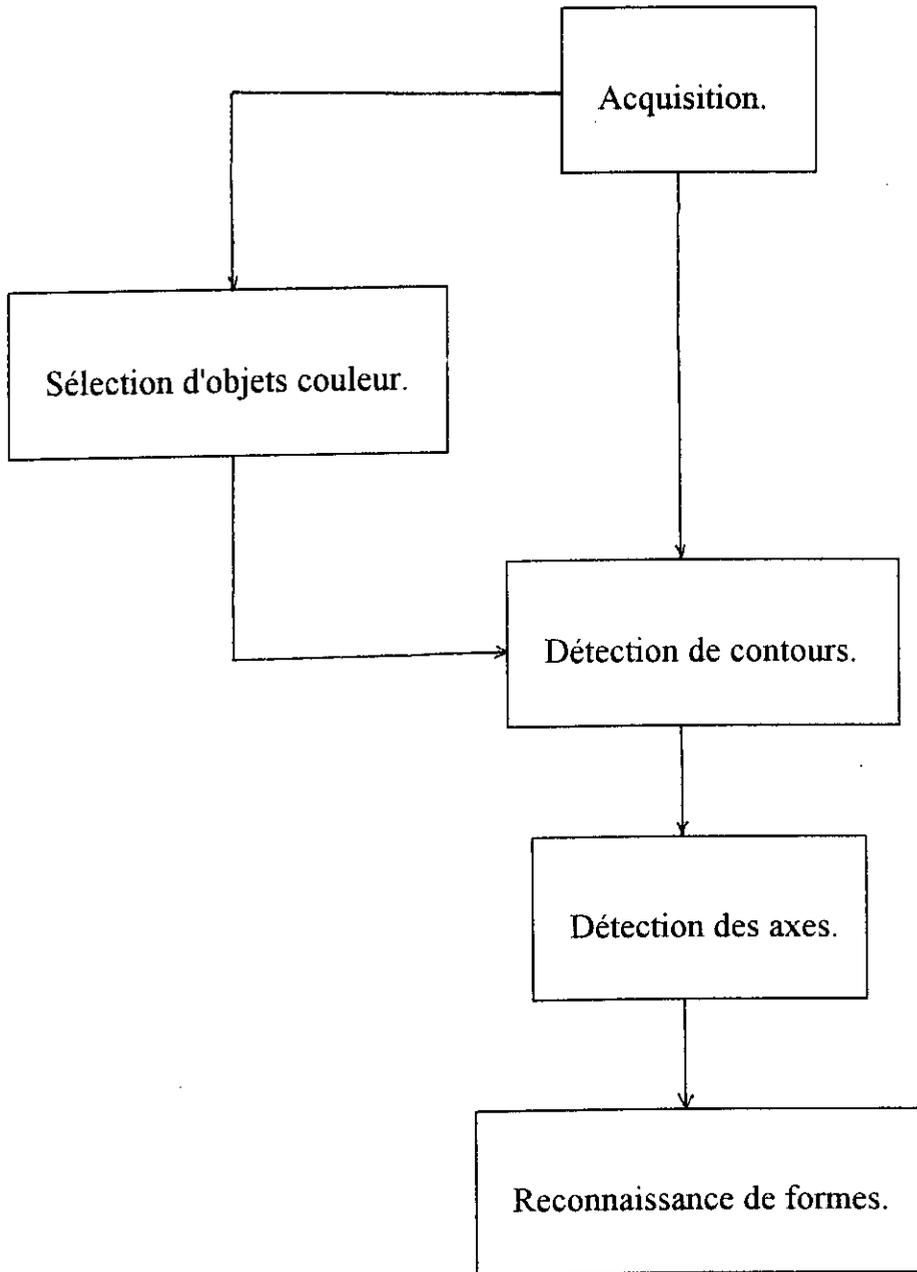


Schéma bloc d'un système complet de robot-vision.

BIBLIOGRAPHIE

- [1] : K. S. Fu, R. C. Gonzales, C. S. G. Lee : " Control, Sensing, Vision and Intelligence ". Mc Graw Hill Book Company. 1987.
- [2] : E. Davalo, P. Naïm : " Des réseaux de neurones ". Editions Eyrolles. 1990.
- [3] : W. K. Pratt : " Digital Image Processing ". Wiley Interscience publications. 1978.
- [4] : N. Baba : " A new approach for finding the global minimum of error function of Neural Networks ". IEEE on Neural Networks. 1989.
- [5] : F. J. Solis, R. J-B. Wets : " Minimisation by random search techniques ". Mathematics of operation research. 1981.
- [6] : S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi : " Optimization by simulated annealing ". IBM research center report. 1982.
- [7] : R. C. Gonzales, P. Wintz : " Digital Image Processing ". Adison-Wesley publishing company. 1977.
- [8] : A. Marion : " Introduction aux techniques de traitement d'images ". Editions Eyrolles. 1987.
- [9] : B. Widrow, M. A. Lehr : " 30 years of adaptive Neural Networks ". Proceedings of the IEEE. 1990.
- [10] : J. A. Freeman : " Neural Networks : algorithms, applications & programming ". Adison-Wesley publishing company. 1992.
- [11] : M. P. Ekstrom : " Digital Image Processing Technics ". Academic Press. New York. 1984.
- [12] : K. R. Castelman : " Digital Image Processing ". Prentice-Hall, INC. 1979.
- [13] : S. Castan : " Vision par ordinateur ". Le Haut Parleur. 1980.

- [14] : L. Saadaoui : " Traitement et acquisition d'images ".
Thèse d'ingénieur. ENP. 1987.
- [15] : A. Bijaoui : " Introduction au traitement numérique de l'image ".
Editions Masson. 1984.
- [16] : S. Geman, C. Wang : " Diffusions for global optimisation ".
SIAM Journal of Control Optimisation. 1986.
- [17] : Pugh : " Robot vision ".
M.I.T. Press. 1986.
- [18] : D. Clayssen, D. Lobstein, J. Zeintoun : " Les nouvelles images ".
Dunod. 1987.
- [19] : L. Saadaoui : "Traitement d'images en vue de la reconnaissance de formes ".
Thèse de Magister. ENP. 1994.