

Algorithme Génétique et Critère de la Trace pour l'Optimisation du Vecteur Attribut : Application à la Classification Supervisée des Images de Textures

M. Nasri et M. EL Hitmy

Ecole Supérieure de Technologie, B.P 473, OUJDA, MAROC.

Email : nasri@est.univ-oujda.ac.ma

Abstract

Selecting the parameters for the classification is a delicate procedure. We present in this paper a new method for selecting the parameters based on the genetic algorithm which optimizes the choice of parameters by minimizing a cost function. This function is defined by a Trace criterion. Our approach is validated on some texture images. The proposed algorithm gives a fast convergence towards the optimal solution.

Résumé

La sélection des paramètres est une procédure très délicate pour la classification. Nous présentons dans cet article une nouvelle méthode basée sur une approche génétique qui optimise le choix des paramètres par la minimisation d'une fonction coût. La fonction coût est choisie d'après le critère de la Trace. Cette approche est validée sur des images de textures. L'algorithme proposé converge rapidement vers la solution optimale.

1 Introduction

Une texture est une répétition spatiale d'un motif de base dans différentes directions [1]. L'analyse des textures joue un rôle très important dans l'interprétation automatique des scènes. Un tel problème est rencontré dans l'imagerie médicale, le contrôle de qualité dans l'industrie, la télédétection, la compression des images TV, etc.

Plusieurs méthodes de caractérisation de texture sur des images ont été proposées dans la littérature. Elles définissent des paramètres discriminants de la texture.

La classification des images de textures consiste à partitionner un ensemble d'images en groupes ou classes de telle sorte que les images appartenant à une même classe soient plus semblables entre elles que celles appartenant à des classes différentes [2]. Cette approche nécessite la sélection des paramètres qui permettent de bien séparer les classes représentatives des différentes textures dans l'espace des paramètres. La classification est dite

supervisée lorsque le nombre de classes est connu *a priori*. Dans le cas contraire, la classification est dite *non supervisée*.

La sélection des paramètres est une procédure très délicate pour la classification [3]. Ce problème demande toute l'expérience du traicteur d'images [1,4]. Il est très important de sélectionner les paramètres les plus pertinents et donc représentatifs en vue de concevoir une bonne classification. Ce choix "*optimal*" conduit à une *rapidité de décision* et peut jouer, pour la classification, le rôle de *filtre* face au bruit apporté par les paramètres non représentatifs. Il existe actuellement plusieurs méthodes et outils de sélection des paramètres [1].

Nous présentons dans ce travail une *nouvelle* méthode basée sur une *approche génétique* qui optimise le choix des paramètres par la minimisation d'une *fonction coût*. Cette dernière est choisie d'après le *critère de la Trace*. Les paramètres sont regroupés dans un vecteur appelé *vecteur attribut*. La sélection des paramètres les plus pertinents revient à *optimiser* ce vecteur.

Dans la section 2, nous présentons quelques paramètres discriminants de la texture. Dans la section 3, nous présentons une introduction aux algorithmes génétiques. Dans la section 4, nous donnons quelques définitions, ensuite nous présentons le critère de la Trace. Dans la section 5, nous formalisons le problème de sélection des paramètres, puis, nous définissons une *stratégie* pour l'application de l'algorithme génétique à la résolution de ce problème. Dans la section 6, nous évaluons les performances de notre méthode par quelques résultats expérimentaux. En fin, nous donnons une conclusion.

2 Paramètres de textures

De nombreuses méthodes de caractérisation de texture sur des images ont été proposées dans la littérature (HARALICK [5], GALLOWAY [6], etc.). Ces méthodes considèrent la texture comme une organisation spatiale des niveaux de gris les uns par rapport aux autres. Elles définissent des paramètres discriminants de la texture en se

basant sur des outils statistiques. Parmi ces méthodes, nous citons les matrices de *cooccurrence* [5] et les matrices des *longueurs de plage* [6].

HARALICK a proposé quatorze paramètres descriptifs des textures à partir des matrices de cooccurrence, les paramètres les plus fréquemment utilisés sont [5] : *homogénéité (Hom)*, *homogénéité locale (HomL)*, *entropie (Ent)*, *uniformité (Uni)*, *directivité (Dir)*, *contraste (Cont)* et *corrélacion (Cor)*.

GALLOWAY a défini cinq paramètres de caractérisation de textures à partir de la matrice de longueurs de plage. Ces paramètres sont [6] : *short run emphasis (SRE)*, *long run emphasis (LRE)*, *gray level nonuniformity (GLN)*, *run length nonuniformity (RLN)* et *run percentage (RP)*.

Il existe d'autres paramètres permettant de caractériser des textures, on peut citer les moments et moyennes d'espace, les paramètres *géométriques*, *fractals* et les paramètres *inter-régions* [1].

3 Algorithmes génétiques

3.1 Introduction

Les algorithmes génétiques (AG) sont des méthodes d'optimisation de fonctions. Ces algorithmes s'inspirent de l'évolution génétique des espèces, schématiquement, ils copient de façon extrêmement simplifiée certains comportements des populations naturelles. Ainsi, ces techniques reposent toutes sur l'évolution d'une population de solutions qui sous l'action de règles précises optimisent un comportement donné, exprimé sous forme d'une fonction, dite *fonction sélective (fitness function)* ou *adaptation à l'environnement* [7].

3.2 Concepts de base

Un algorithme génétique est un algorithme itératif de recherche d'optimum, il manipule une *population* de taille constante. Cette population est formée de points candidats appelés *chromosomes*. La taille constante de la population entraîne un phénomène de compétition entre les chromosomes. Chaque chromosome représente le codage d'une solution potentielle au problème à résoudre, il est constitué d'un ensemble d'éléments appelés *gènes*, pouvant prendre plusieurs valeurs appartenant à un *alphabet* non forcément numérique [8].

A chaque itération, appelée *génération*, est créée une nouvelle population avec le même nombre de chromosomes. Cette génération consiste en des chromosomes mieux "*adaptés*" à leur environnement tel qu'il est représenté par la fonction sélective. Au fur et à mesure des générations, les chromosomes vont tendre vers l'optimum de la fonction sélective. La création d'une nouvelle population à partir de la précédente se fait par application des opérateurs génétiques que sont : la

sélection, le *croisement* et la *mutation*. Ces opérateurs sont *stochastiques* [9].

La sélection des meilleurs chromosomes est la première opération dans un algorithme génétique. Au cours de cette opération l'algorithme sélectionne les éléments pertinents qui optimisent mieux la fonction. Le croisement permet de générer deux chromosomes nouveaux "*enfants*" à partir de deux chromosomes sélectionnés "*parents*" (figure 1), tandis que la mutation réalise l'inversion d'un ou plusieurs gènes d'un chromosome (figure 2) [9].

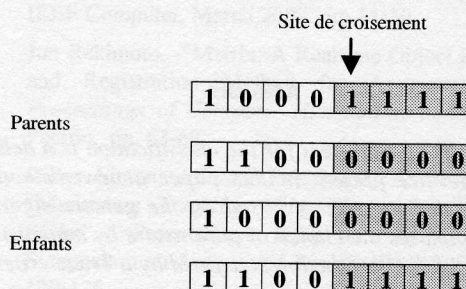


Figure 1 : Opérateur de Croisement.

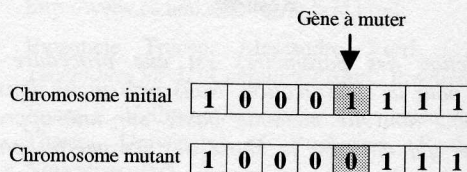


Figure 2 : Opérateur de Mutation.

La figure 3 illustre les différentes opérations qui interviennent dans un algorithme génétique de base [10]:

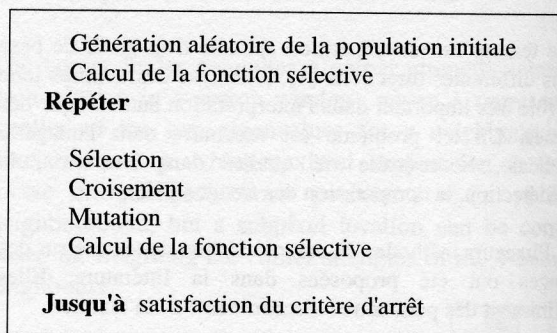


Figure 3 : Algorithme génétique de base.

3.3 Résolution d'un problème par AG

L'application de l'AG à la résolution d'un problème nécessite de coder les solutions potentielles à ce problème en des chaînes finies de bits afin de constituer les chromosomes, de trouver une fonction sélective permettant une bonne discrimination entre les chromosomes et de définir les opérateurs génétiques qui seront utilisés [8].

4 Eléments descriptifs et critère de la Trace

4.1 Définitions

Considérons un ensemble de M images de textures $\{I_1, I_2, \dots, I_i, \dots, I_M\}$ caractérisées par N paramètres de textures regroupés sous la forme d'un vecteur ligne $V = (P_1, P_2, \dots, P_j, \dots, P_N)$. Soit $R_i = (P_{ij})_{1 \leq j \leq N}$ un vecteur de \mathbf{R}^N dont la $j^{\text{ème}}$ composante P_{ij} est la valeur prise par le paramètre de texture P_j sur l'image de texture I_i . Soit $E_V = \{P_1, P_2, \dots, P_j, \dots, P_N\}$. Soit mat_va la matrice de M lignes (représentant les images I_i) et de N colonnes (représentant les paramètres P_j), définie par :

$$mat_va = \left(P_{ij} \right)_{\substack{1 \leq i \leq M \\ 1 \leq j \leq N}}$$

On appelle V vecteur attribut, R_i l'observation associée à l'image I_i ou réalisation du vecteur attribut V pour cette image, \mathbf{R}^N l'espace d'observation ou l'espace des paramètres [1,2,4], E_V l'ensemble associé à V et mat_va matrice d'observation associée à V . la $i^{\text{ème}}$ ligne de mat_va est la transposée de R_i , notée R_i^t . Chaque réalisation R_i appartient à une classe C_s , $s=1, \dots, N_c$.

4.2 Matrice de variance totale, matrice intraclasse et matrice interclasse

Soit col_{P_j} le vecteur de dimension M associé au paramètre P_j , défini par :

$$col_{P_j} = \left(P_{ij} \right)_{1 \leq i \leq M} \\ = j^{\text{ème}} \text{ colonne de } mat_va$$

Soit m_{P_j} la moyenne du paramètre P_j , définie par :

$$m_{P_j} = \frac{1}{M} \sum_{i=1}^M P_{ij}$$

Soit $m_{(P_j, C_s)}$ la moyenne du paramètre P_j des réalisations (individus) de la classe C_s , définie par :

$$m_{(P_j, C_s)} = \frac{1}{l_s} \sum_{\substack{i \\ R_i \in C_s}} P_{ij} \\ l_s \text{ est le cardinal de } C_s$$

Soit $V_{(m)}$ le vecteur des moyennes de chaque paramètre, défini par :

$$V_{(m)} = \left(m_{P_j} \right)_{1 \leq j \leq N}$$

Soit $V_{(m, C_s)}$ le vecteur des moyennes de chaque paramètre des réalisations (individus) de la classe C_s , défini par :

$$V_{(m, C_s)} = \left(m_{(P_j, C_s)} \right)_{1 \leq j \leq N}$$

La matrice de variance totale T_V associée au vecteur attribut V est définie par [1,11] :

$$T_V = \frac{1}{M} \sum_{i=1}^M (R_i - V_{(m)}) (R_i - V_{(m)})^t$$

La matrice intraclasse W_V associée au vecteur attribut V est définie par [1,11] :

$$W_V = \frac{1}{M} \sum_{s=1}^{N_c} \sum_{\substack{i \\ R_i \in C_s}} (R_i - V_{(m, C_s)}) (R_i - V_{(m, C_s)})^t$$

W_V est la somme des matrices de covariance de chacune des classes et représente une mesure de compacité des classes. Elle est appelée aussi matrice de compacité [11].

La matrice interclasse B_V associée au vecteur attribut V est définie par [1,11] :

$$B_V = T_V - W_V$$

B_V représente une mesure de séparabilité des classes. Elle est appelée aussi matrice de séparabilité [11].

Les trois matrices présentées T_V , W_V et B_V sont de dimension $N \times N$ et servent souvent à établir un critère qui mesure le pouvoir discriminant d'un ensemble de paramètres [11]. Dans notre cas nous limiterons au critère de la Trace qui est largement utilisé.

4.3 Critère de la Trace

Pour estimer le pouvoir discriminant de q paramètres parmi les N qui existent, on utilise les matrices de variance totale, intraclasse et interclasse mais restreintes aux q paramètres considérés [1]. Soit V_k ($k \in N^*$) un vecteur attribut arbitraire, de dimension q , extrait de V et soit T_{V_k} ,

W_{V_k} et B_{V_k} les matrices de variance totale, intraclasse et interclasse associées à V_k . Ces trois matrices sont de dimension $q \times q$.

Le pouvoir discriminant de V_k est estimé par [1,11] :

$$J_{V_k} = \text{Trace}(T_{V_k}^{-1} B_{V_k}) \\ \text{Trace}(X) \text{ est la trace de la matrice } X$$

Ce critère est à maximiser : plus la valeur de J_{V_k} est grande plus V_k est discriminant [1,11].

5 Pour une résolution génétique du problème d'optimisation du vecteur attribut

5.1 Formalisation du problème d'optimisation

Nous disposons d'un vecteur attribut initial $V_{init} = (P_1 P_2 .. P_j .. P_N)$ et de M réalisations $(R_i)_{1 \leq i \leq M}$ de ce vecteur. Chaque réalisation R_i est associée à l'image I_i . Chaque image I_i appartient à une classe connue (*classification supervisée*), dans notre cas nous nous limiterons à trois classes C_1, C_2 et C_3 ($N_c = 3$) définies comme suit :

$$\begin{aligned} (R_i)_{1 \leq i \leq l_1} &\in C_1 \\ (R_i)_{l_1+1 \leq i \leq l_2} &\in C_2 \\ (R_i)_{l_2+1 \leq i \leq M} &\in C_3 \end{aligned}$$

l_1 et l_2 : deux entiers délimitants les trois classes.

Notre objectif est d'extraire parmi les N paramètres $(P_j)_{1 \leq j \leq N}$ du vecteur V_{init} les q ($q \ll N$) paramètres les plus pertinents au sens du critère de la Trace. Les q paramètres concernés constituent le vecteur attribut optimal noté V_{op} .

5.2 Stratégie d'application de l'AG à la résolution du problème d'optimisation du vecteur attribut.

Codage proposé

Soit V_k ($k \in N^*$) un vecteur attribut arbitraire extrait de V_{init} , V_k peut avoir une à N composantes, donc à chaque V_k est associé le chromosome chr_k de N gènes binaires g_{kj} :

$$\begin{aligned} chr_k &= (g_{kj})_{1 \leq j \leq N} \\ \text{avec } g_{kj} &= \begin{cases} 1 & \text{si } P_j \in E_{V_k} \\ 0 & \text{sinon} \end{cases} \end{aligned}$$

chr_k est une matrice ligne binaire de dimension N . Chaque gène g_{kj} code la présence ou l'absence du paramètre P_j dans le vecteur attribut V_k . Le vecteur V_k codé par chr_k , ne peut être une solution possible au problème à résoudre que si :

$$Dim(V_k) = \sum_{j=1}^N g_{kj} = q$$

Dans l'algorithme génétique nous devons éliminer tout chromosome codant un vecteur attribut ne respectant pas cette contrainte.

Fonction sélective proposée

Soit V_k ($k \in N^*$) un vecteur attribut extrait de V_{init} , de dimension q , codé par chr_k , pour calculer la valeur sélective de chr_k codant V_k , nous définissons la fonction sélective F par :

$$F(chr_k) = \frac{1}{Trace(T_{V_k}^{-1} B_{V_k})}$$

V_k est la solution optimale au problème à résoudre si $F(chr_k)$ est minimale.

Opérateurs génétiques utilisés

Nous avons adopté pour la phase de sélection, la *stratégie de sélection par rangement* (qui consiste à ranger les chromosomes par ordre croissant de leur valeur sélective et à leur attribuer une probabilité de sélection selon leurs rangs) et pour la phase de reproduction (croisement et mutation) la *stratégie élitiste* (qui consiste à garder intact le meilleur chromosome de la population lors du passage d'une génération à la suivante) [9].

6 Résultats expérimentaux et évaluations

6.1 Introduction

La sélection des paramètres les plus pertinents au sens des performances de la classification, dépend des textures à classer et de la vitesse d'exécution recherchée [4]. Le nombre de paramètres à sélectionner q doit être comparable au nombre de classes [1].

Selon ces trois recommandations citées ci-dessus, nous avons envisagé deux expérimentations tenant compte de la nature des textures à classer. Ces expérimentations sont réalisées pour $q = 2$. La valeur de q retenue est d'une part comparable au nombre de classes que nous avons (trois classes) et d'autre part traduit que nous nous intéressons à la vitesse d'exécution et que nous désirons capter le caractère intrinsèque des textures en peu de paramètres. Nous avons retenu dans les deux expérimentations le vecteur attribut initial suivant :

$$V_{init} = (\text{Hom HomL Ent Uni Dir Cont SRE LRE GLN RLN RP})$$

Afin d'évaluer les performances de notre méthode, nous avons utilisé un *classifieur supervisé* utilisant comme méthode de classification, la méthode des *k-means* [2].

6.2 Première expérimentation

Nous avons choisi pour ce test trente images de textures (figure 4) réparties en trois classes Sol_1, Sol_2 et Sol_3 . Les trois textures représentent trois types de sol différents.



I_1 I_{11} I_{21}
 $1 \leq i \leq 10, I_i \in C_1$ $11 \leq i \leq 20, I_i \in C_2$ $21 \leq i \leq 30, I_i \in C_3$

Figure 4 : Trois images test représentant les trois classes.

L'exécution de l'AG s'effectue rapidement. La figure 5 montre l'évolution de la valeur sélective du meilleur chromosome de la population courante au fur et à mesure des générations. Le vecteur attribut optimal V_{op} donné par l'algorithme génétique est :

$$V_{op} = (HomL, Ent)$$

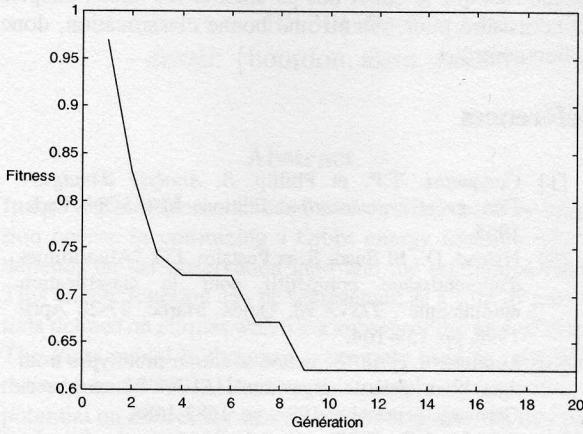


Figure 5 : Evolution de la valeur sélective en fonction de la génération.

Les résultats de la classification des images test dans l'espace $(HomL, Ent)$ sont résumés par la figure 6. Les classes données par le classifieur sont les suivantes :

$$C_1 = \{I_1, I_2, I_3, I_4, I_5, I_7, I_8, I_{10}\}$$

$$C_2 = \{I_6, I_9, I_{11}, I_{12}, I_{13}, I_{14}, I_{15}, I_{16}, I_{17}, I_{18}, I_{19}, I_{20}\}$$

$$C_3 = \{I_{21}, I_{22}, I_{23}, I_{24}, I_{25}, I_{26}, I_{27}, I_{28}, I_{29}, I_{30}\}$$

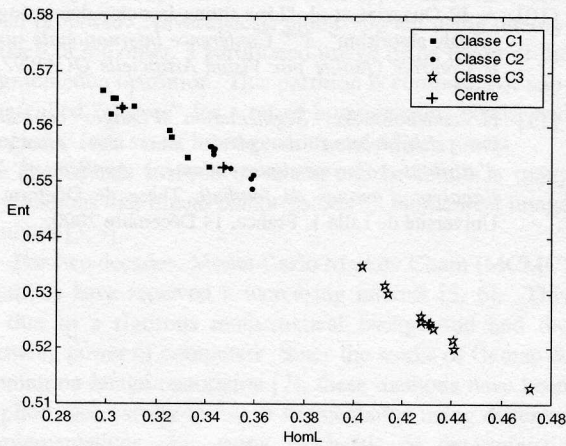


Figure 6 : Résultats de la classification dans l'espace $(HomL, Ent)$.

Les résultats obtenus montrent que les deux paramètres sélectionnés $HomL$ et Ent permettent une bonne séparation des trois classes dans l'espace des paramètres. En effet,

nous remarquons que toutes les images sont correctement affectées aux classes correspondantes, à l'exception des deux images I_6 et I_9 qui sont mal classées (affectées à la classe C_2 au lieu C_1). La séparation des trois classes est donc réalisée avec un taux d'erreur faible.

D'après la figure 6, les deux paramètres sélectionnés sont corrélés. Avec un seul paramètre nous avons obtenu le même taux d'erreur.

$$\text{Taux d'erreur} = \frac{2}{30} = 6.67\%$$

6.3 Deuxième expérimentation

Nous avons utilisé dans ce deuxième test quinze images de textures (figure 7) réparties en trois classes Sol_1 , $Laine$ et Sol_3 .

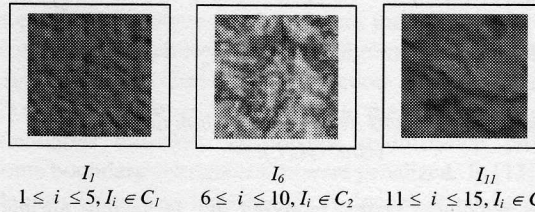


Figure 7 : Trois images test représentant les trois classes.

L'AG converge rapidement. La figure 8 illustre l'évolution la valeur sélective du meilleur chromosome de la population courante au fil des générations. Le vecteur attribut optimal V_{op} donné par l'AG dans ce cas est :

$$V_{op} = (Uni, GLN)$$

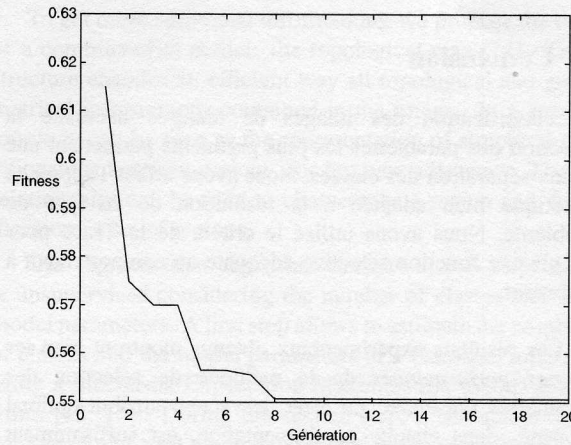


Figure 8 : Evolution de la valeur sélective en fonction de la génération.

Les résultats de la classification dans l'espace (Uni, GLN) sont donnés dans la figure 9. Les classes que nous donne le classifieur sont les suivantes :

$$C_1 = \{I_1, I_2, I_3, I_4, I_5\}$$

$$C_2 = \{I_7, I_8, I_9, I_{10}\}$$

$$C_3 = \{I_6, I_{11}, I_{12}, I_{13}, I_{14}, I_{15}\}$$

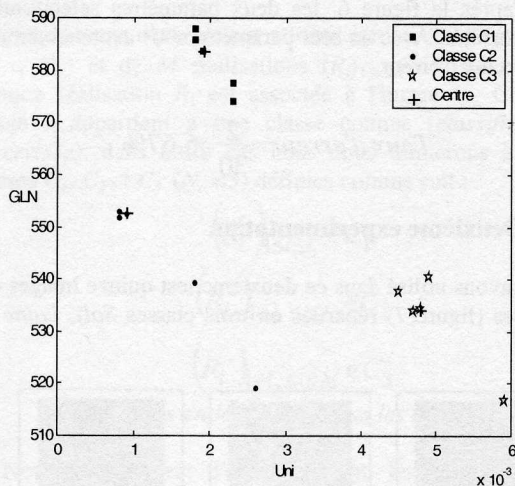


Figure 9: Résultats de la classification dans l'espace (Uni, GLN).

Les résultats obtenus montrent que le vecteur attribut optimal proposé dans ce cas, rend très efficace la discrimination. Les deux paramètres sélectionnés *Uni* et *GLN* permettent de bien séparer les trois classes dans l'espace des paramètres. En effet, nous observons que toutes les images test sont bien classées, sauf l'image I_6 qui affectée à la classe C_3 au lieu C_2 . Ceci est traduit par un taux d'erreur faible.

$$\text{Taux d'erreur} = \frac{1}{15} = 6.67\%$$

7 Conclusion

La classification des images de textures nécessite la sélection des paramètres les plus pertinents permettant une bonne séparation des classes. Nous avons utilisé l'approche génétique bien adaptée à la résolution de ce type de problème. Nous avons utilisé le critère de la Trace pour définir une fonction sélective adéquate au comportement à optimiser.

Les résultats expérimentaux obtenus montrent bien les *bonnes performances* de la méthode de sélection des paramètres présentée. En effet, le vecteur attribut optimal proposé, dans chaque expérimentation, est suffisamment représentatif puisque ses composantes permettent une *bonne séparation* des classes représentatives des différentes textures dans l'espace des paramètres, ce qui est traduit par des *taux d'erreurs faibles*. L'algorithme proposé assure une *convergence rapide* vers la solution optimale. Par ailleurs, nous avons montré que dans la première expérimentation un paramètre aurait suffi pour obtenir le

même taux d'erreur, alors que dans la deuxième expérimentation, les deux paramètres sont nécessaires. Ces deux expérimentations confirment que la dimension du vecteur attribut optimal dépend des images à traiter. L'optimisation de cette dimension peut faire l'objet d'une étude par la suite.

Notre approche peut être utilisée concrètement dans les problèmes de contrôle de qualité par vision artificielle à chaque fois que le choix des paramètres les mieux adaptés est nécessaire pour obtenir une bonne classification, donc un bon contrôle.

Références

- [1] Cocquerez, T.P. et Phillip, S. *Analyse d'images : Filtrage et segmentation*. Editions MASSON, Paris, 1995.
- [2] Hamad, D., El Saad, S. et Postaire, J.G. "Algorithmes d'apprentissage compétitif pour la classification automatique", *TISVA '98*, Oujda, Maroc, 27-28 April 1998, pp. 159-164.
- [3] Ludimla, I. et al. "Selection of cluster prototypes from data by a genetic algorithm", *EUFIT'97*, Aachen, Germany, September 1997, pp. 1683-1688.
- [4] Postaire, J.G., *De l'image à la décision : Analyse des images numériques et théorie de la décision*. Editions DUNOD, 1987.
- [5] Haralick, R.M. "Statistical and structural approaches to textures". *IEEE*, Vol. 67, N° 5, 1979, pp. 786-804.
- [6] Galloway, M.M. "Texture analysis using gray level run lengths". *Computer Graphics Image Processing*, 1975, Vol. 4, pp. 172-179.
- [7] Lutton, E., *Algorithmes génétiques et Fractales*. Dossier d'habilitation à diriger des recherches, Université Paris XI Orsay, 11 Février 1999.
- [8] Ludovic, Mé., *Audit de sécurité par algorithmes génétiques*. Thèse de Doctorat, Université de Rennes 1, France, 7 Juillet 1994.
- [9] Renders, J.M., *Algorithmes génétiques et Réseaux de Neurones*. Editions HERMES, 1995.
- [10] A. El Ouazizi et al. "Line fitting in noisy data using genetic algorithm". *3^{ème} Conférence Internationale sur le Contrôle Qualité par Vision Artificielle QCAV'97*, 1997.
- [11] N.Vandenbroucke. *Segmentation d'images couleur par classification de pixels dans des espaces d'attributs colorimétriques adaptés : Application à l'analyse d'images de football*. Thèse de Doctorat, Université de Lille 1, France, 14 Décembre 2000.