

Chapitre 04

Traitement d'images par écosystèmes

Introduction

Le domaine de la simulation d'écosystèmes requiert la contribution de plusieurs disciplines notamment certaines liées à la biologie et à l'informatique. Le but de ce type de simulation est de comprendre les formes complexes d'interactions qui caractérisent les organismes d'écosystèmes. L'inspiration la plus reconnue de ces interactions est la coevolution (*voir §2.1.4*), où l'évolution d'une espèce affecte l'évolution des espèces avec lesquelles elle interagit, par exemple par des relations de type chaîne alimentaire, prédateur proies, symbiose, parasitisme ...

Dans ce chapitre, nous présentons une approche de segmentation par *coevolution* d'images en niveaux de gris. Pour cela on a utilisé deux types hétérogènes d'agents réactifs : agent région et agent contour, qui contribuent à la segmentation de l'image qui est considérée comme leur environnement d'évolution.

Afin de détailler cette méthode, ce chapitre est organisé de la façon suivante :

Premièrement, les différentes approches concernant l'application de coevolution ainsi des définitions de base seront présentés. Le paragraphe suivant s'attardera sur la description de notre approche suivi des détails d'évolution des différents agents. Les comportements des agents seront abordés à la fin de ce chapitre. Ce dernier sera achevé par une conclusion.

4.1 Travaux récents dans les algorithmes coevolutionnaires

Dans ces dernières années, plusieurs recherches qui s'intéressent à l'application de la coevolution ont été publiées dans la littérature. Ces recherches utilisent, un type d'algorithmes qui constitue une extension des *algorithmes evolutionnaires* est reconnu par : *Algorithmes coevolutionnaires* [Kirley 02].

Dans [Drezewski 03], un système multi agent coévolutionnaire (*co-evolutionary multi-agent system* : CoEMAS) a été introduit ainsi un modèle formel de CoEMAS est présenté. Dans un tel système, deux ou plusieurs espèces d'agents coévolvent afin de résoudre un problème donné. L'application de CoEMAS à l'Optimisation de la Fonction Multi – Modale a été présentée dans ce travail.

Une autre application qui utilise les algorithmes coevolutionnaires est présentée dans [Zaritsky 03]. Cette application concerne la recherche de la plus courte ficelle qui contient toutes les ficelles d'un ensemble donné. Ce problème est reconnu sous l'appellation (SCS: Shortest Common Superstring). Trouver le plus court ficelle commun a des candidatures dans la compression de données. SCS a aussi des candidatures importantes dans la biologie computationnelle (computational biology) concernant le problème de séquençement d'ADN...

Dans le domaine de traitement d'images, le travail de [Veenman 03,+], présente un modèle pour le problème de segmentation, qui n'a pas besoin de connaissance au sujet du nombre de groupes a priori. Le système proposé est basé sur un algorithme du coevolution cellulaire pour l'optimisation du modèle. Dans ce modèle des agents sont placés dans une grille 2D régulière qui représente l'image et qui impose des relations avoisinantes sur eux. Les agents, d'une manière coopérative, considèrent la migration de pixel d'un agent à l'autre pour améliorer l'homogénéité de l'ensemble des régions de l'image qu'ils représentent. Si l'union des régions d'agents avoisinantes est homogène alors les agents forment des alliances. De l'autre côté, si un agent découvre un sujet déviant, il isole le sujet.

4.2 Définitions de base

4.2.1 Gradient

Le gradient de l'image donne une information sur les variations existantes dans l'image. Il existe un nombre considérable de techniques pour calculer le gradient discret. Dans la plupart des cas, le gradient est calculé grâce à un produit de convolution. L'évaluation des différentes méthodes permet de constater qu'elles produisent des résultats similaires pour la majorité des applications [Salotti 94]. Nous avons choisi une méthode reposant sur l'utilisation de gradient discret (voir §4.5). La norme du gradient est calculée en utilisant les directions représentées par le code de Freeman.

4.2.2 Seuillage

Le critère de seuillage permet d'accepter ou non un pixel selon qu'il est au-dessus ou en-dessous d'un seuil calculé. La particularité de ce seuillage est dans le calcul du seuil et dans l'évolution dans le temps de ce seuil. En effet, le seuil peut être calculé selon n'importe laquelle des caractéristiques de l'environnement. De plus, ce calcul est adapté en tenant compte des informations récoltées par la segmentation en cours. Comme cette segmentation évolue, ainsi que les informations récoltées sur celle-ci, le seuil évolue tout au cours de la vie du comportement [Boucher 99]. Dans notre méthode, le seuil n'est utilisé que s'il est nécessaire (C'est à dire, lorsque l'agent région rencontre un agent contour).

4.3 Description de l'approche

L'approche proposée dans ce mémoire concerne la segmentation d'images en niveaux de gris par coevolution. Pour cela, deux types d'agents réactifs sont définis : *Agents contour* et *Agent Région*. Initialement, les agents sont initialisés aléatoirement comme des germes dans l'image, qui représente l'environnement d'évolution de ces agents.

Nous avons utilisé la construction incrémentale (pour le contour et la région) qui permet de prendre en compte les configurations locales de l'environnement, tant en termes photométriques qu'en termes de primitives préexistantes dans le système. L'incrémentalité dans un système de vision apporte donc une richesse de fonctionnement à travers une

évolution temporelle, des possibilités de prises de décision dynamiques, et plus globalement une adaptabilité des méthodes.

Après initialisation, les agents régions et contours se déplacent et marquent leur environnement suivant le principe de l'incrémentalité.

Dans ce qui suit, on va détailler le processus d'évolution de chaque type d'agents.

4.4 Evolution d'agent région

A partir d'un germe (pixel) initial, l'agent région commence à se déplacer d'un pixel à un autre afin de marquer une région homogène. Le déplacement de l'agent se fait aléatoirement à l'un de ses 4-voisins qui représentent les candidats à ajouter à la région. L'illustration est montrée dans la (figure 4.1).

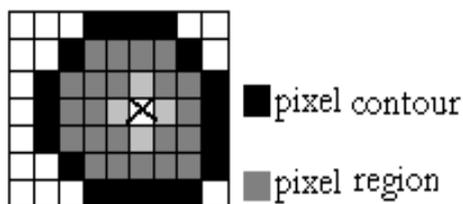


Figure 4.1 Evolution d'un agent region à partir d'un germe initial.

Considérons une méthode de croissance incrémentale de régions utilisant le schéma générique décrit au chapitre 3, dont le but est de fournir des régions correspondant à des zones de luminosité homogène de l'image. L'algorithme d'évaluation des pixels devra donc logiquement privilégier les points candidats dont l'intensité des niveaux de gris est la plus proche du niveau de gris moyen de la région courante. Définissons ainsi $P(x,y)$ le pixel candidat, ayant une intensité $I(x,y)$. La région R courante contient n pixels référencés $P(x_0,y_0), P(x_1,y_1), \dots, P(x_{n-1}, y_{n-1})$, et la moyenne des niveaux de gris de R est définie par :

$$\bar{I}_R = \frac{1}{n} \sum_{i=0}^{n-1} I(x_i, y_i)$$

L'évaluation E_1 du pixel P par rapport à la région R se définit donc par :

$$E_1(x, y) = 1 - \min\left(1, \frac{|I(x, y) - \bar{I}_R|}{k}\right)$$

où k est une constante

L'évaluation E_1 est maximum pour les points dont le niveau de gris correspond exactement à celui moyen de la région, et décroît lorsque cette différence devient plus importante. Le seuil k permet de normaliser l'expression dans l'intervalle $[0..1]$.

L'application de ce critère unique fournira certainement un résultat adéquat, mais au prix d'une forte sensibilité au bruit et aux éléments texturés de l'image. Dans de telles conditions, l'algorithme de croissance génère une région à la frontière déchiquetée, et peu exploitable en l'état. En effet, le pixel recherché est le plus homogène, il peut donc être située n'importe où à la frontière de la région. Aucune contrainte de forme ne vient forcer la région à conserver un aspect régulier. D'autres considérations et contraintes d'implantation peuvent suggérer de construire des régions aux formes plus régulières, sans pour autant renoncer au critère d'homogénéité initial.

L'idée est d'introduire une évaluation supplémentaire, notée E_2 , dont le rôle est d'évaluer le degré de compacité du pixel $P(x,y)$, comparé à la région R : Définissons des relations de voisinage $N_8(P(x_1,y_1),P(x_2,y_2)) = 1$ si et seulement si $\max(|x_2-x_1|,|y_2-y_1|) = 1$, ensuite l'évaluation :

$$\begin{aligned} E_2(x, y) &= \frac{1}{8} \sum_{p(x_i, y_i) \in R} N_8(P(x_i, y_i), P(x, y)) \\ &= \frac{1}{8} (\text{Nombre de pixels du voisinage immédiat de } P) \end{aligned}$$

Cette seconde évaluation E_2 fournira de fortes valeurs pour les pixels les plus entourés par une région.

Chaque évaluation considérée séparément correspond à un critère élémentaire. E_1 est le critère *d'homogénéité* tandis que E_2 reflète ce que l'on peut qualifier de critère de *compacité*, dont le comportement seul sera de combler les petits trous et aspérités au cours de la croissance de la région.

4.5 Evolution d'agent contour

Nous utilisons le terme contour au sens d'une structure linéaire regroupant des points d'un type donné. Ces points peuvent devenir une frontière, ou plus généralement, un indice visuel linéaire.

La construction du contour suit le même principe de construction de région basé sur le modèle incrémental de croissance. La *figure 4.2*, illustre le principe d'évolution d'un contour. Après initialisation aléatoire dans l'image, l'agent contour commence à se déplacer d'un pixel

à un autre afin de déterminer le germe de départ de la croissance du contour. A partir d'un germe initial, l'évaluation de pixels candidats aux extrémités du contour permet de le suivre en agrégeant les pixels ayant la plus forte probabilité d'appartenir à ce contour. Le déplacement de l'agent se fait à l'un des cases de ses 8-voisins qui représentent les candidats de contour.

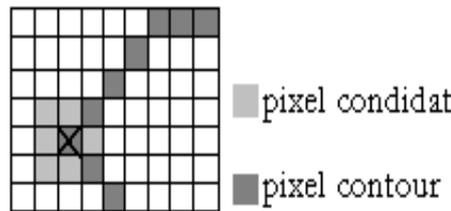


Figure 4.2 Evolution d'un agent contour à partir d'un germe initial. La case marquée par un croix représente le germe initial du contour, les 8-voisins de ce germe sont marquée par des cases candidats (gris clair).

Les critères d'évaluation lors de la construction de contour utilisent les informations récoltées par l'agent contour. Ces informations concerne : les extrémités du contour, le calcul du gradient et sa direction... Dans ce qui suit on va expliquer le principe de calcul du gradient ainsi sa direction pour la détection de contour. Ce principe est inspiré de [Bellet 98].

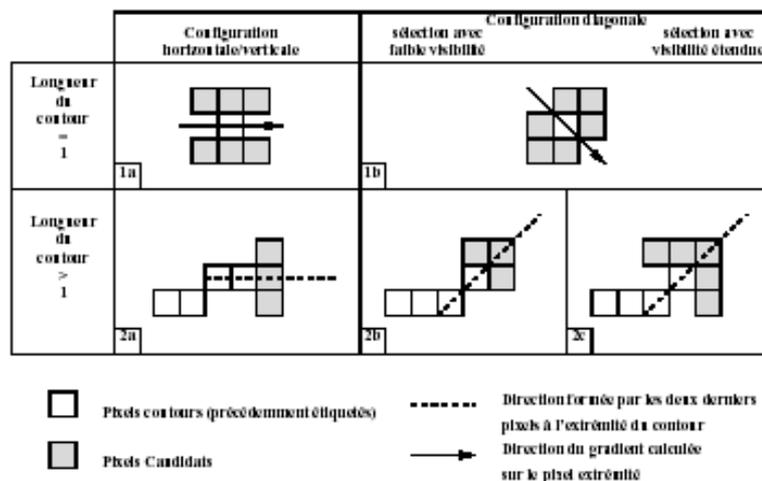


Figure 4.3 Le mode de sélection des pixels candidats pour la croissance de contour d'après [Bellet 98]. La mesure de la direction du gradient sert lorsque le contour ne contient qu'un seul pixel. Dans les autres cas, la configuration des pixels de l'extrémité et la nécessité de préserver la connexité permettent de retenir trois ou cinq pixels candidats

Si l'on appelle $P_1=P(x_1,y_1)$, et $P_2 = P(x_2,y_2)$ respectivement le dernier et l'avant dernier pixel du contour, P_1 et P_2 sont des pixels voisins, ce qui se traduit par la relation :

$$\max (|x_1-x_2|, |y_1-y_2|) =1$$

C'est à dire que la distance de P_1 à P_2 vaut 1 pour la norme ∞ . Le déplacement (dx,dy) entre les pixels P_2 et P_1 se définit alors par $(dx,dy) = (x_2-x_1,y_2-y_1)$, avec comme contrainte : $dx,dy \in \{-1,0,1\}$.

On définit alors la direction de Freeman entre les pixels P_2 et P_1 par : $D = F(dx,dy)$, avec :

$$\begin{array}{lll} F(-1,-1)=3 & F(0,-1)=2 & F(1,-1)=1 \\ F(-1,0)=4 & F(0,0) \text{ indéfini} & F(1,0)=0 \\ F(-1,1)=5 & F(0,1)=6 & F(1,1)=7 \end{array}$$

Dans le cas où la direction de Freeman F est paire (figure 4.4, cas 2a), la sélection des trois pixels grises est effectuée, correspondant aux directions F , $(F + 1) \bmod 8$ et $(F + 7) \bmod 8$. Dans le cas où la direction de Freeman est impaire (figure 4.4, cas 2b et 2c), il convient alors de sélectionner en plus les pixels correspondant aux directions $(F + 2) \bmod 8$ et $(F + 6) \bmod 8$.

La principale évaluation relative à un pixel contour candidat sera la norme du gradient, les autres types d'évaluation ne serviront qu'à pondérer cette évaluation principale. Considérons le pixel candidat $P(x,y)$ pour le contour C dans l'image P . Le gradient est calculé à partir des valeurs d'intensité dans le voisinage $\overrightarrow{G(x,y)}$, sous la forme de sa norme $G(x,y) = \|\overrightarrow{G(x,y)}\|$, et de sa direction G_{d8} , échantillonnée suivant les huit directions de Freeman $G_{d8} \in \{0..7\}$.

- La première évaluation repose sur une mesure locale du gradient :

$$E_1 = \frac{G(x, y)}{\max(x_i, y_i)_{\in P} G(x_i, y_i)}$$

Cette évaluation privilégie les points avec les plus fortes valeurs de la norme du gradient.

- $E_2 = 1$, la norme du gradient est localement maximum dans la direction du gradient.

$E_2 = 0$ sinon.

$$E_2 = 1 \quad - \quad \left\{ \begin{array}{ll} G(x-1, y) < G(x, y) & \text{et } G(x+1, y) < G(x, y) \\ & \text{si } G_{d8} \in \{0, 4\} \\ G(x, y-1) < G(x, y) & \text{et } G(x, y+1) < G(x, y) \\ & \text{si } G_{d8} \in \{2, 6\} \\ G(x-1, y+1) < G(x, y) & \text{et } G(x+1, y-1) < G(x, y) \\ & \text{si } G_{d8} \in \{1, 5\} \\ G(x+1, y+1) < G(x, y) & \text{et } G(x-1, y-1) < G(x, y) \\ & \text{si } G_{d8} \in \{3, 7\} \end{array} \right.$$

4.6 Coevolution d'agents pour la segmentation d'images

On a présenté dans le chapitre 2 (§4), le principe de coevolution ainsi que ses formes (coevolution coopérative et coevolution compétitive). Dans notre travail, nous avons exploité le principe de coevolution compétitive pour la segmentation d'images en niveaux de gris en s'inspirant des travaux de [Veenman 03,+]. Pour cela on a défini deux types d'agents : agent contour et agent région présentés ci-dessus. Ces agents contribuent à la segmentation de l'image qui constitue leur environnement. Le principe de ce processus est illustré dans la figure ci-dessous.

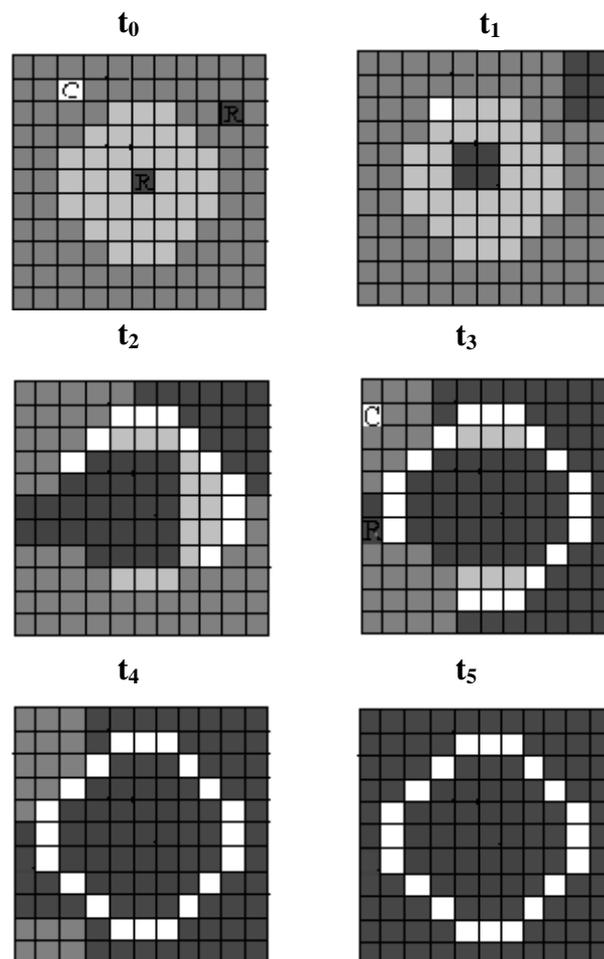


Figure 4.4 Principe de segmentation d'une image par coevolution entre agents région et agents contour. A l'instant t_0 , trois agents sont initialisés dans l'image, agent contour (case marquée par la couleur blanche et avec une lettre C) et deux agents région (cases grises foncées avec une lettre R). à l'instant t_1 , lancement des trois agents ; l'agent contour se déplace à l'un des pixels voisins il se trouve à un pixel qui représente le germe initial ; les deux agents région commencent à marquer leurs régions suivant le principe incrémental de région. À l'instant t_2 , il y a une compétition entre agent contour et agent région, l'agent région dépasse le contour avant l'arrivée d'agent contour. A l'instant t_3 , l'agent contour arrête l'agent région de se dépasser sa région et reproduction de deux agents : agent région qui continue l'évolution de la nouvelle région et un agent contour pour évoluer un contour s'il existe. A l'instant t_4 , la mort d'un agent région ainsi les agents contour et la fusion des deux agents région. A l'instant t_5 , image résultat, définition de deux régions.

4.7 Comportements de l'agent

Un comportement est une action que peut mettre en oeuvre l'agent, dans le but d'explorer l'environnement ou de modifier celui-ci. Chaque comportement correspond à une étape nécessaire du processus de segmentation. Trois comportements sont définis dans notre méthode concernant les deux types d'agents (région et contour):

- **perception** : Ce comportement sert à créer la primitive contour ou région pour laquelle l'agent est spécialisé. Il permet à l'agent de construire, de façon incrémentale, une primitive en évaluant l'environnement et les différentes informations accumulées par lui et par les autres agents. Il est le comportement vital de l'agent, qui détermine sa survie, car c'est sur la construction de la primitive qu'est défini le but (implicite) de l'agent. C'est ce comportement, de par les informations qu'il récoltera, qui permettra aux autres comportements de s'activer.
- **reproduction** : Sert à créer des agents en certains lieux de l'image. Ce comportement permet à un agent d'explorer son environnement local et de lancer des germes d'agents qui continueront le travail de segmentation.
- **interaction** : Le comportement d'interaction gère tous les processus d'interactions entre deux agents. Il s'agit d'actions qui résultent en des communications directes entre deux agents concernés par une situation précise comme la fusion de primitives décrivant la même entité. L'interaction par fusion permet à deux agents, qui travaillent sur la même composante, de réunir leurs primitives (région ou contour). En effet, tel qu'expliqué avec le comportement de reproduction, la stratégie employée par le système est de lancer plus d'agents que nécessaire, pour être certain de ne manquer aucune zone de l'image. Les agents ainsi lancés commencent par segmenter leurs composantes et vont se rencontrer au fur et à mesure qu'ils croissent. Deux agents qui se rencontrent et découvrent qu'ils travaillent tous les deux sur la même composante de l'image peuvent fusionner leurs efforts afin de diminuer le nombre d'agents redondants. Un seul des deux agents continuera le travail, l'autre se terminant pour permettre de diminuer la charge du système. La fusion ne concerne donc que des agents possédant la même spécialisation, c'est à dire de même type.

Le principe de base dans l'action de ces comportements est l'évaluation de l'information afin de prendre des décisions. La coevolution entre les agents est le résultat de ces actions, où l'évolution d'un agent affecte l'évolution d'un autre agent. Ceci est évident pour le cas d'une évolution d'un agent région par exemple, et qui dépasse les limites de celui-ci, l'agent contour va arrêter l'évolution de l'agent région sans utilisation d'un seuil sur l'ensemble de la région.

Le schéma général du système est représenté dans la figure (4.5) :

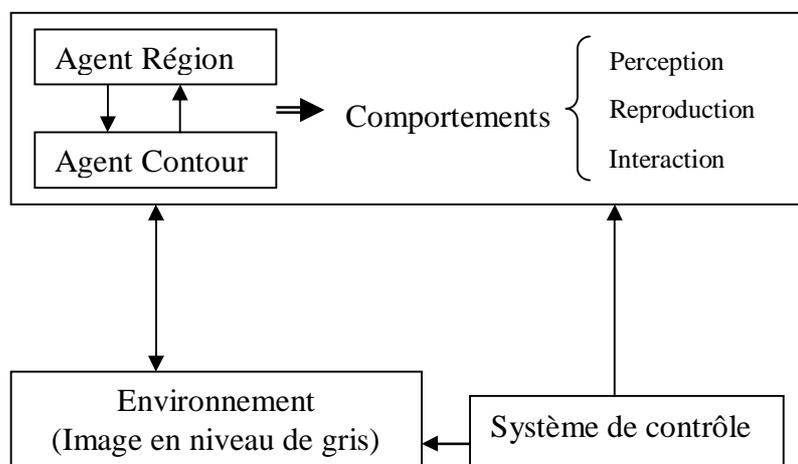


Figure 4.5 Schéma général du système

Conclusion

Dans ce chapitre, on a présenté une description détaillée d'une nouvelle méthode pour la segmentation d'images en niveaux de gris en se basant sur le principe de coevolution. On a vu que deux types d'agents (région et contour) contribuent à la segmentation d'une image par une coevolution compétitive. Les résultats obtenus par application de cette méthode seront abordés dans le chapitre suivant.

