

Chapitre 05

Résultats

Introduction

Afin de mettre en valeur, ou de mieux expliquer, les aspects importants de notre système, différents résultats illustrant le système de segmentation d'images de synthèse et d'images médicales sont montrés dans ce chapitre. L'aboutissement à ces résultats est facilité par l'utilisation d'un *PC PIII 500MH* d'une *RAM 128 Mo*, ainsi par l'utilisation du *Builder C++*. Ainsi l'application de la méthode sous NetLogo, qui a donnée des résultats encourageants.

5.1 Initialisation du système

Avant de démarrer le système, un ensemble de paramètres doivent être initialisés. Ces paramètres présentent :

- *Nombre d'agents contour et d'agents région (N_{region} , $N_{contour}$)* : ces paramètres sont définis avant de placer les agents dans l'image originale. Ils représentent respectivement le nombre d'agents contour qui démarrent la croissance du contour, et le nombre d'agents région qui démarrent la croissance de régions. Ces deux paramètres varient au fur et à mesure de l'évolution des agents, ceci est expliqué par le comportement de reproduction.
- *Seuil région* : Il est nécessaire seulement dans le cas où une région est arrêtée par un contour, il permet de déterminer les candidats (les 4-voisins) à agréger au région.

5.2 Application aux images de synthèse

Afin de valider notre approche, nous avons l'appliquée sur des images de synthèse (voir l'exemple de la figure 5.1).

Dans l'exemple ci-dessous, nous avons utilisé une image de synthèse en niveau de gris de taille 256*256. Initialement, nous avons défini $N_{region}=05$ (nombre de germes région) et $N_{contour}=02$ (nombre de germes contour).

A $t=t_0$, les agents région cherchent les candidats en 4-voisins et les marquent comme appartenant à la région homogène trouvée. Les agents contours, n'apparaissent pas, puisque ils cherchent les germes initiaux qui démarrent la croissance de contour en se basant sur les 8-voisins du germe courant.

A $t=t_1$ les agents région et contour en compétition pour segmenter l'image ; les agents contour évoluent de telle façon à interdire les agents région de dépasser les limites des régions homogènes. Dans ce cas, le comportement de reproduction est apparaît. Il est évident aussi, que deux agents de même type permettent une fusion.

A $t=t_2$, deux agents région meurent. Les autres agents continuent la segmentation. L'image résultat est donnée dans la figure ci-dessous.

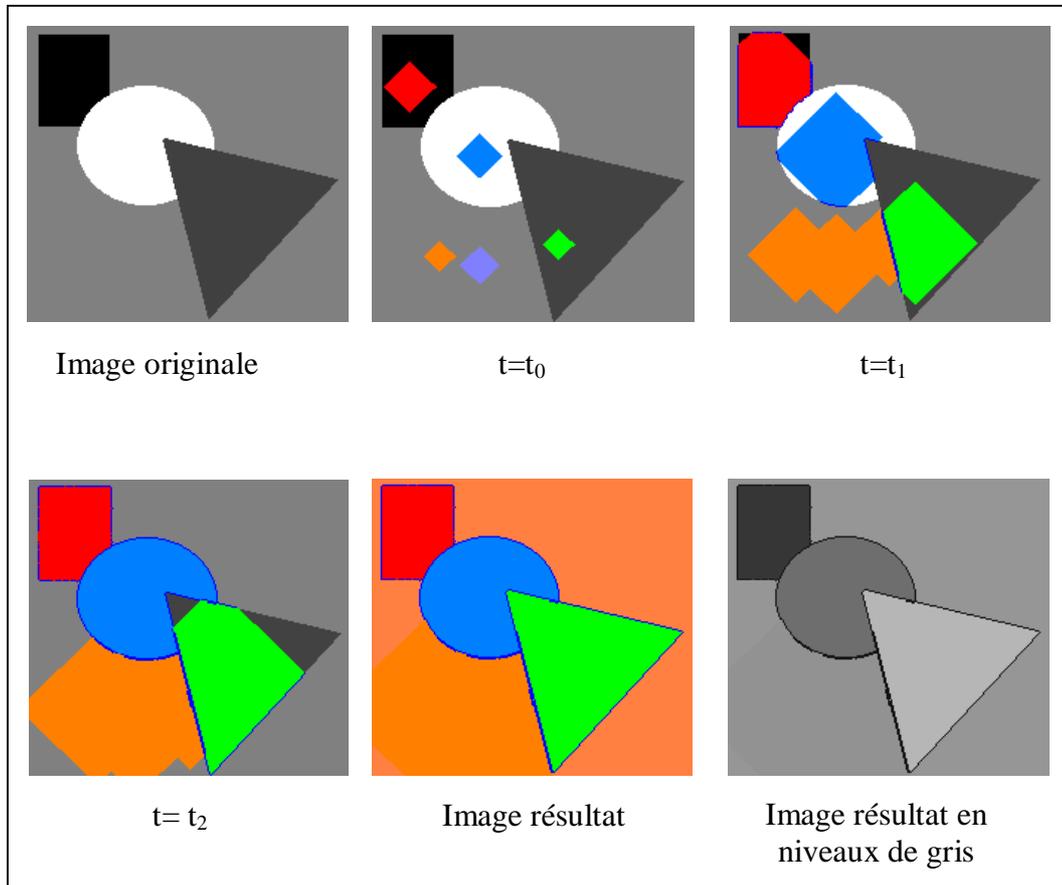


Figure 5.1 *Processus de coevolution d'agents (contour et région) pour la segmentation d'une image de synthèse (256*256)*

Durant l'évolution de processus de segmentation, on a acquérir le nombre de pixels représentant les régions marquées ainsi le nombre de pixels des contours détectés. La figure 5.2, représente la croissance du nombre de pixels régions et contours dans le temps. La courbe bleue représente le nombre de pixels des régions, et la courbe mauve représente le nombre de pixels contours récoltés par les agents au fur et à mesure de processus de coevolution.

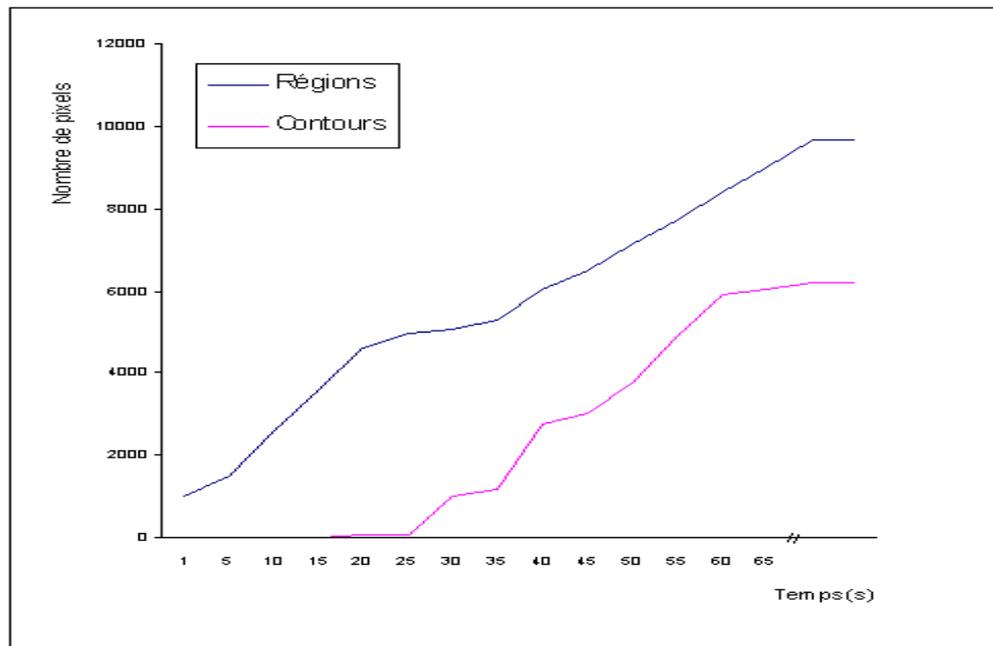


Figure 5.2 Evolution du nombre de pixels régions et contours en fonction du temps

Nous avons remarqué, qu'il y'a une stabilisation après un certain temps du nombre de pixels détectés soit par les agents région ou bien par les agents contour. Ceci est expliqué par le fait que les agents terminent leurs travaux et qu'il ne reste pas de candidats à agréger aux primitives.

5.3 Application aux images médicales

Notre système est appliqué sur des images médicales (IRM), un des exemples est présenté dans la figure 5.2.

Dans cet exemple, nous avons utilisé une image cérébrale 140*170. Le système est initialisé par $N_{région}=8$ et $N_{contour}=5$. Durant le processus de coevolution compétitive d'agents, il y'a une reproduction d'agents qui continuent la recherche des régions à segmenter.

A $t=t_0$, les agents commencent leur évolution. A $t=t_1$, reproduction de nouveaux agents qui cherchent des régions à segmenter.

A $t=t_2$, les régions qui segmentent une même composante sont fusionnées. Un seul des deux agents continuera le travail, l'autre se terminant pour permettre de diminuer la charge du système.

A $t=t_3$, le résultat est présenté sous formes de régions colorées par les couleurs d'agents. Le résultat en niveaux de gris est montré dans la figure ci dessous.

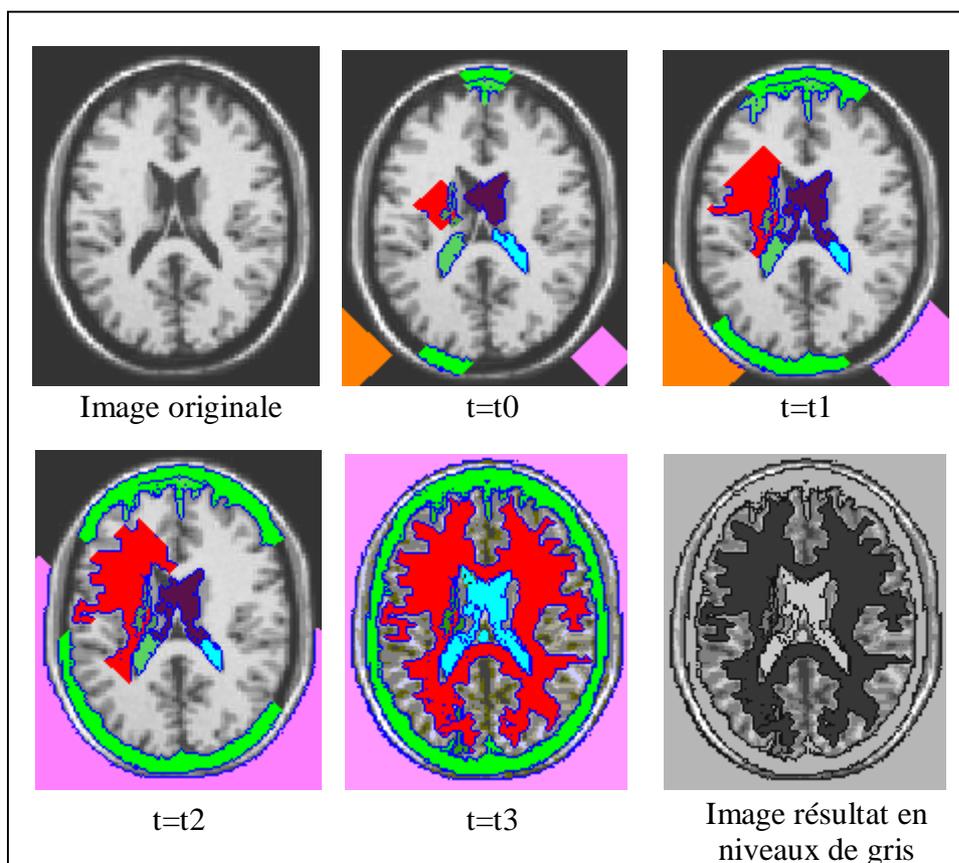


Figure 5.3 *Processus de segmentation par coévolution d'une image cérébrale (140*170)*

5.4 Utilisation du NetLogo (1)

NetLogo est un environnement de programmation multi agents écrit en Java, il a plusieurs versions. Vu les grands avantages (parallélisme...) qu'offre NetLogo, nous avons l'exploiter pour l'application de notre méthode sur différents types d'images. Des résultats satisfaisants sont démontrés dans la figure (5.4).

NetLogo permet de simuler des phénomènes naturels et sociaux. Il est bien convenu pour modeler des systèmes complexes qui se développent avec le temps en particulier. Les modeleurs peuvent donner des directives à centaines ou milliers d'agents " *indépendants* " qui fonctionnent en concurrence.

NetLogo est doté de trois types d'agent : tortues, patchs, et observateur. Les tortues sont des agents qui peuvent prendre différentes formes et qui installent sur le monde. Le monde est une grille de deux dimensions divisé en cases représentant les patchs. Ils constituent l'environnement des tortues qui peuvent se déplacer sur ces patchs. L'observateur n'a pas d'emplacement, vous pouvez l'imaginer en dehors du monde de tortues et patchs.

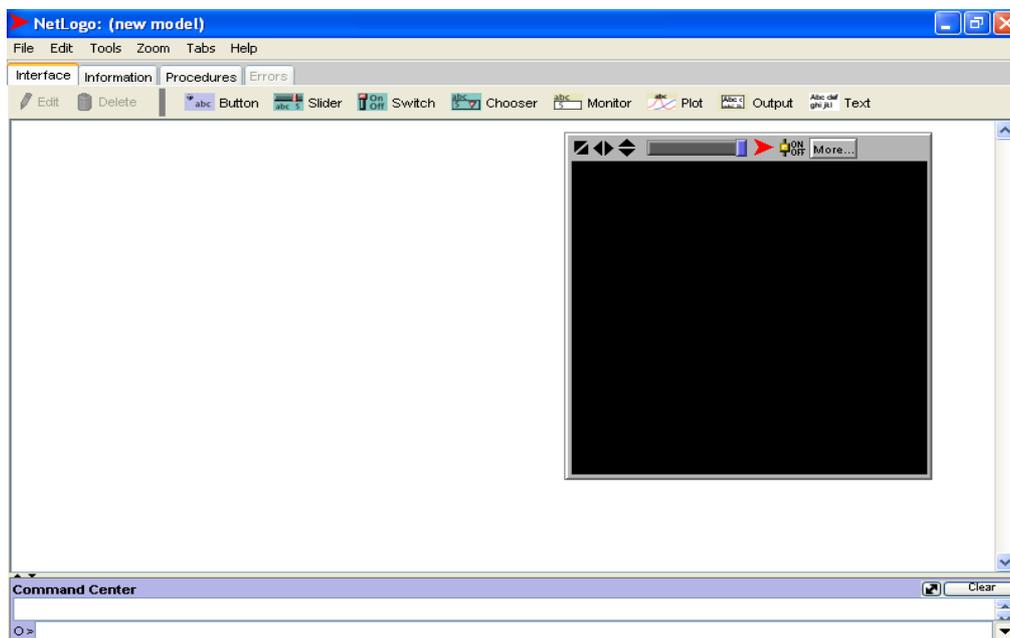


Figure 5.4 Interface du NetLogo (Version 2.1)

(1) <http://ccl.northwestern.edu/netlogo/>

Notre méthode est appliquée sous NetLogo et nous avons abouti aux résultats présentés dans la figure ci-dessous (images de synthèse et réelles).

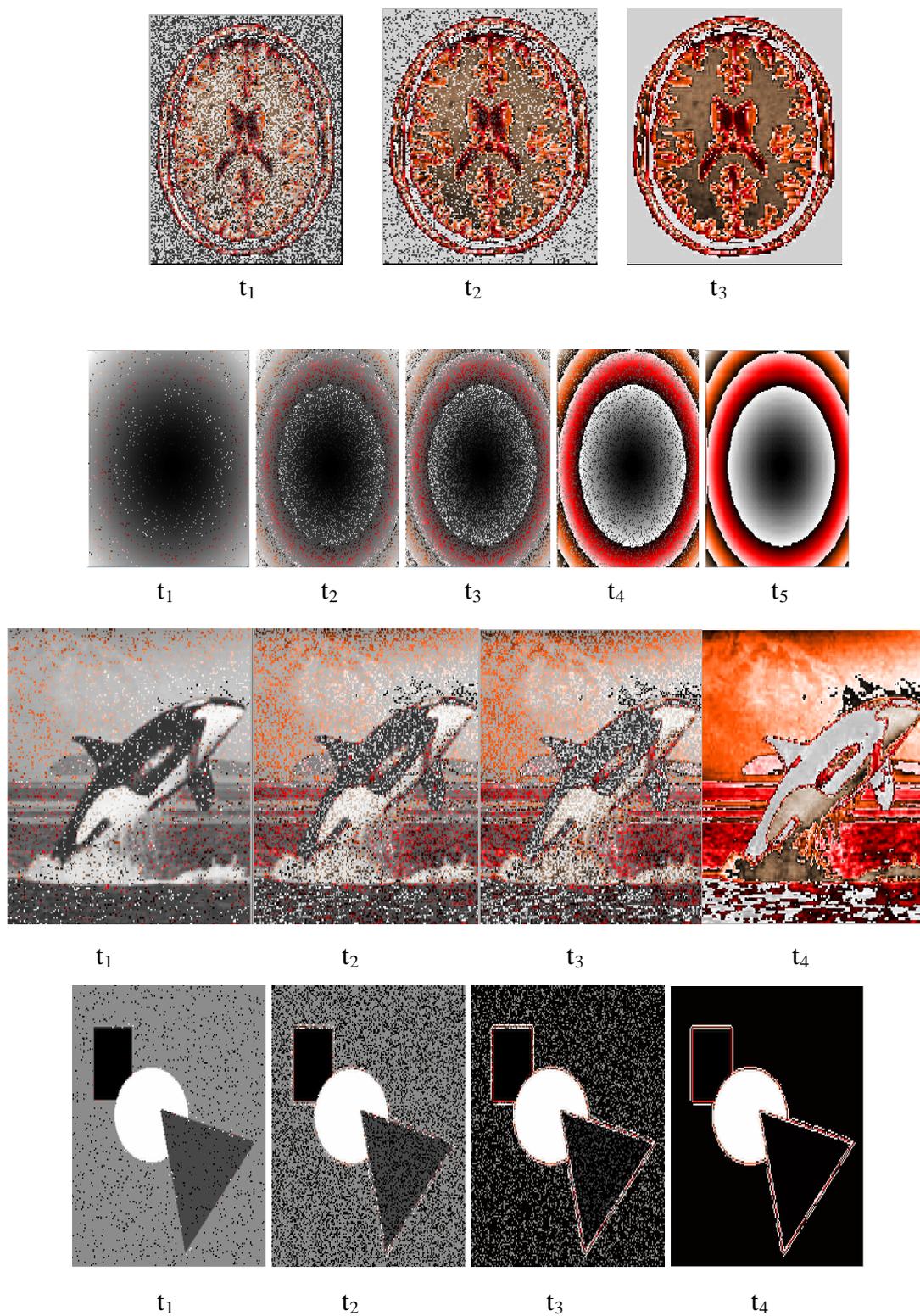


Figure 5.5 Ensemble de résultats obtenus à partir du NetLogo

5.4 Discussion

Les résultats présentés dans ce chapitre, montrent le processus de coevolution entre les deux types d'agents : région et contour pour la segmentation d'images de différents types. Nous avons remarqué que, plus le nombre de germes contours et région est élevé, plus que la segmentation est converge.

Le nombre important des agents dans l'image, mène par la suite, à des fusions entre ceux qui travaillent sur la même composante. En effet, tel qu'explique avec le comportement de reproduction.

Un agent peut également s'appuyer sur les informations récoltées par ses voisins pour mener à bien sa tâche. C'est sur ce modèle de coopération que s'effectue toute la segmentation de l'image. Le système est généraliste et ne bénéficie d'aucune connaissance sur les applications et les images qu'il segmente.

Notre approche suit le modèle de croissance de primitives (région et contour) dans leur évolution de même que le travail de [Djerou 02] qui présente une approche de segmentation par émergence. Concernant les interactions entre agents, notre système, est inspiré des travaux mentionnés dans le chapitre 04.

Au niveau des critères d'évaluation du contour, il est important de tenir en compte en plus les fermetures ainsi que l'utilisation des méthodes plus puissantes pour vérifier et choisir les maxima locaux du contour.

Conclusion

Ce chapitre a montré quelques résultats de notre système, en illustrant autant les différents aspects de conception de notre approche que les résultats de segmentation.

Les résultats obtenus sont encourageants puisque nous avons appliqué notre approche sur différents types d'images et nous avons remarqué que les régions détectées sont satisfaisantes.

