World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:10, No:5, 2016

Edge Detection Using Multi-Agent System: Evaluation on Synthetic and Medical MR Images

A. Nachour, L. Ouzizi, Y. Aoura

***Abstract*—**Recent developments on multi-agent system have brought a new research field on image processing. Several algorithms are used simultaneously and improved in deferent applications while new methods are investigated. This paper presents a new automatic method for edge detection using several agents and many different actions. The proposed multi-agent system is based on parallel agents that locally perceive their environment, that is to say, pixels and additional environmental information. This environment is built using Vector Field Convolution that attract free agent to the edges. Problems of partial, hidden or edges linking are solved with the cooperation between agents. The presented method was implemented and evaluated using several examples on different synthetic and medical images. The obtained experimental results suggest that this approach confirm the efficiency and accuracy of detected edge.

***Keywords*—**Edge detection, medical MR images, multi-agent systems, vector field convolution.

I. Introduction

EDGE detection is useful in several areas such as 3D reconstruction, pattern recognition, robotics, and medical imaging. Use of edge makes analyses and identification simpler. Numerous traditional detectors have been proposed over the years including derivative computation of the image pixels to assess any change in intensity profile of neighboring pixels: Robert, Sobel, Prewitt, Canny, and so on. These approximations of the gradient intensity have been used to reduce gradient computation times [1], but most of them include a stage where the local properties of a pixel and its neighborhood are evaluated. Consequently, it follows that the larger the neighborhood, the greater the number of possible solutions, but a higher complexity of the algorithm and a longer processing time.

Active Contour Model (ACM) is another type to solve problems of edge detection. The basic idea, originally proposed by Kass et al. [2], is attracting or repulsing an evolving contour toward the nearest salient edge. The active models deform on the image domain and capture a desired feature by minimizing the energy functional subject to certain constraints. Many existing models have been applied to various applications that showing their superior results [3].

Another type of methods has been set in images processing which are based on multi-agent systems [4], [5]. Recently, the image segmentation based agents has become quite popular, since these tools make possible to apply more advanced algorithms and to perform computationally demanding tasks quickly. Parallel agents can analyze problems that they are locally confronted with, and then use the algorithm which seems to be the more suitable for their local context [6]-[9]. The cooperation enables agents to share different collected information with others agents in their environment. The cooperative approaches increase the quality of the segmentation by confronting the information provided by the different algorithms.

This paper presents a method for the automatic edge detection using several agents and many different actions. First, initialized agents named ‘scouts’ look for edges by minimizing the energy functional attached to its position according to a Vector Field Convolution (VFC) [10] that attracts them towards the existing edges. Then, agents named ‘edge agent’ moves to neighbor pixel in this edge which grows in length. ‘edge agent’ negotiate with others agent the closing of the edge it is representing. All agents in the system cooperate with other agents to come to an overall edge detection. Cooperation involves fusion between agents or adding an intermediate agent as nodes agents. The global agent named ‘Observer’ loads image, creates, initializes and kills agents.

The rest of the paper is organized as follows: Section II presents a related work. Section III provides the illustration of the proposed approach. Section IV analyzes the results obtained and at the same time comparing it to other techniques. Finally, Section V concludes the work of this paper.

II. Related Work

MAS are used in several applications for segmentation which remains a central task in image processing. Rodin et al. [11] present a parallel agent for detection of age rings of tree. They use a set of agents which follow either the light rings or the dark rings and act on the image. The agent’s actions aim at the reinforcement of the rings by stressing the contrasts thus allowing a reliable detection of these rings, even if they are discontinuous. The system has an important over detection of the edges than traditional method that use Canny filter. Liu et al. [12] present a parallel image processing system based on simple reactive agents for homogeneous region detection for

Abdelhafid Nachour is with the Modelisation, Optimisation and Logistics Laboratory in ENSAM, Ecole Nationale Superieure d’Arts et Meiters, Moulay Ismail University Meknes, Morocco (e-mail: nachour.abdel@gmail.com).

Latifa Ouzizi is with the Department of Industrial Engineering and Production in ENSAM, Ecole Nationale Supérieure d’Arts et Métiers, Moulay Ismail University Meknes, Morocco (e-mail: l.ouzizi@ensam-umi.ac.ma).

Youssef Aoura is with the Department of Materials and Process Engineering in ENSAM, Ecole Nationale Supérieure d’Arts et Métiers, Moulay Ismail University Meknes, Morocco (e-mail: y.aoura@ensam-umi.ac.ma).

brain scan images segmentation. Agents act according to a perception–action model without problem solving or deliberation. Each agent can achieve tests on pixels around it in a circular neighborhood. Bovenkamp [13] elaborate a high-level knowledge-based control over low-level image segmentation algorithms. The agents dynamically adapt segmentation algorithms based on knowledge about global constraints, contextual knowledge, local image information and personal beliefs. The system has been applied to IntraVascular UltraSound (IVUS) images which are segmented by agents, each of them is specialized for segmenting a predefined primitive. Settache et al. [14] use multi-agent system to share the result of a quad tree algorithm used to identify the primitive regions, and a Shen filter to determine edges in the MRI brain part's detection. The solution consists of merging the regions according to their homogeneity, and the edge agent can prohibit the fusion between two regions if it is located in between. Fleury [15] proposes a segmentation method for multi-object detection, semi-interactive and generic nature, applied to the extraction of cardiac structures in CT imaging. Bellet et al. [16] present an incremental processes of region growing and edge detection where the cooperation between two types of agent is dynamic and allows to transmit information when it becomes necessary to take a decision. This cooperative approach increases the segmentation's quality by confronting the information provided from different algorithms. Yanai et al. [17] use a MAS to extract the primitive information like lines, edges or regions using different types of algorithms. Each agent is located on the image and builds a set of coherent primitives. Then, the agents interact to negotiate their local primitives.

III. Proposed Methodology

A. Pre-Processing VFC

Vector Field Convolution is proposed as a new external force for active contours by Bing Li [10]. VFC is calculated by convolving the edge map generated from the image with the user defined vector field kernel $k(x,y)=[u_k(x),v_k(y)]$ in which all the vectors point to the kernel origin

$$k(x,y)=m(x,y)n(x,y) \tag{1}$$
$$k(x,y)=[u_k(x),v_k(y)]$$

where $m(x,y)$ is the magnitude of the vector at (x,y) and $n(x,y)$ is the unit vector pointing to the kernel origin.

$$n(x,y)=[-\frac{x}{r},-\frac{y}{r}] \tag{2}$$

where $r=\sqrt{x^2+y^2}$, and $n(0,0)=[0,0]$ at the origin [10].

The VFC force $f_{VFC}=[u_{VFC}(x),v_{VFC}(y)]$ is given by calculating the convolution of the vector field kernel $k(x,y)$ and the edge map $f_G(x,y)$ generated from the image $I(x,y)$:

$$f_{VFC}=f_G(x,y)*k(x,y) \tag{3}$$

The VFC is just the filtering result of the edge map, which does not depend on the origin of the kernel [10]. The VFC field highly depends on the magnitude of the vector field kernel $m(x,y)$. By considering the fact that the influence from the edge should decrease as the particles are further away, the magnitude should be a decreasing positive function of distance from the origin. We use in this work a magnitude function, given as [10]:

$$m(x,y)=(r-\varepsilon)^{-\gamma} \tag{4}$$

where γ is a positive parameter to control the decrease, and ε is a small positive constant to prevent division by zero at the origin.

B. Implementation of Multi-Agent System

To build a MAS, we use JADE (Java Agent Development Framework), created by TILAB laboratory described by Bellifemine et al. [18]. JADE allows the development of multi-agent systems and applications conforming to FIPA standards [19]. It is implemented in JAVA and provides classes that implement "JESS" for the definition of agents' behavior. JADE has three main modules (necessary to FIPA standards):

- DF "Director Facilitor" provides services "yellow pages" to the platform.
- ACC "Communication Channel Agent" manages the communication between agents.
- AMS "Agent Management System" oversees the registration of agents, their authentication, access and use of the system.

These three models are activated each time when starting the platform.

The interest of a multi-agent architecture lies in the collaboration between agents. Therefore, these agents have several ways of communication: synchronous, asynchronous and broadcasting. The synchronous communication corresponds to simple calls of methods as in JAVA, C++, and so forth. On the other hand, a higher level communication requires that the agents have the ability to exchange messages on an asynchronous way, with the management of a message box, in order to increase the liability of the service demands, propositions, negotiations, etc. In our system, first, each agent has a very simple behavior which allows it to take a decision (find out an edge) according to its position in the image and to the information enclosed in it. Second, each agent can send messages using the FIPA ACL that provides the types of required message. A simple example of JAVA code implementation is as follow:

- ACLMessage Message = new ACLMessage(ACLMessage.INFORM);
- Message.addReceiver(new AID("Scout12", AID.ISLOCALNAME));
- Message.setContent("Hellow");
- send(message).

The reception of message is assured using the following JAVA code.

- ACLMessage message_Recu = receive();
- String Text = message_Recu.getContent();.

C. Multi-Agent Model

The used MAS model is constituted by the agents and their environment and many different actions (Fig. 1). The environment contains the image. Each pixel of this image characterizes a gray level and contains a Boolean value which defines if the pixel has already been explored by an agent. Agents have different behaviors according to their current state and perception:

- The agent named ‘Observer’ is the control center of our system (Fig. 1). This agent creates and initializes all agents in the system and decides to remove inactive agents.
- The agents named ‘scouts’ are responsible to Checking the edge.
- The ‘edge agent’ for exploration step.
- Others agents is also created as the agents named ‘node agent’ that represent the start position of ‘edge agents’ in different direction and ‘end edge agent’ representing the end of specific edge detection.

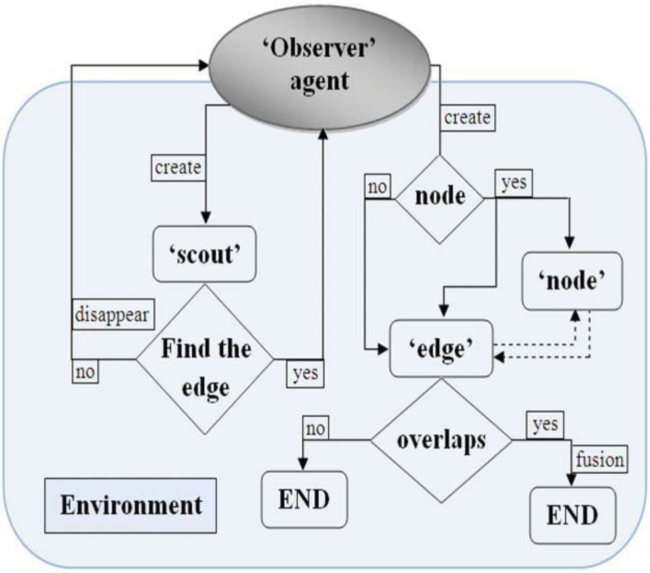

Fig. 1 MAS model used in this approach

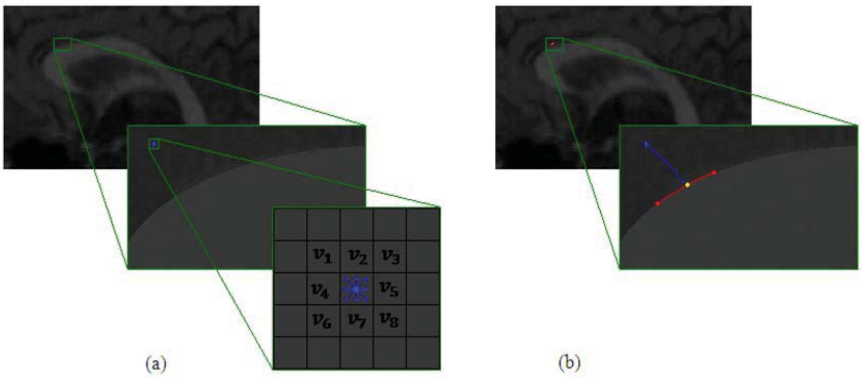

Fig. 2 Checking process: (a) ‘scout’ looks for the edge. (b) Find the edge and initialize the Exploration step. ‘scout’ pixel is colored by bleu, ‘node agent’ by yellow and ‘edge agent’ by red

The process of edge detection is divided in three steps:

- Checking step: Each ‘scout’ moves from its position to its neighbor that minimizes the VFC energy functional attached to its position.
- Exploring step: The ‘scouts’ check if it is on an edge, not already explored. If it is the case, each ‘scout’ determines the possible directions of the edge and communicate whit the ‘Observer’ to create an ‘edge agent’. Each ‘edge agent’ is initialized in one direction and marks every pixel by following the edge.
- Finally, ‘end edge agent’ negotiates for closing the edge with other agents to come to an overall detection.

1. Checking Step

The process of checking the edge is carried out by calculating the VFC values of the neighbors v_k (k=8), and moving the ‘scout’ to the neighbor's coordinate that has the lowest energy (Fig. 2 (a)). When the ‘scout’ attaints the edge and not moves, it determines various possible directions of the edge. If there is one direction to flow, Observer replaces the ‘scout’ by an ‘edge agent’. Otherwise, if there is more than one direction to follow, the ‘scout’ communicates with the Observer to create a ‘node agent’ and initiate each ‘edge agent’ for each direction (Fig. 2 (b)). If the ‘scout’ does not find any edge before a deadline, it disappears.

2. Exploring Step

This step is initialized by ‘edge agent’ to follow the considered edge starting from the ‘edge node’ or the position where it is to its neighbor. If this neighbor is valid, then the agent stores its position in its list of positions and advances

(Fig. 3). 'edge agent' verifies every time if the edge pixel is already exploited by the others agents. If it is the case, it asks a fusion with these agents and disappears. The fusion process is carried out when an 'edge agent' is linked with another one which explores the same edge in the opposite direction. In this case, 'Observer' replaces the two agents by another that becomes the 'end agent' (Fig. 4).

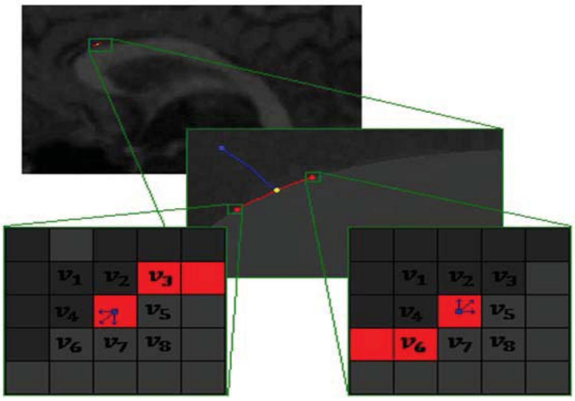

Fig. 3 Two 'edge agent' in red initialized for exploring the detected edge. The 'node agent' is represented in yellow

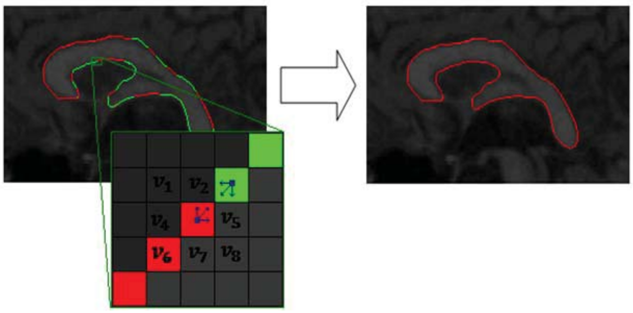

Fig. 4 Fusion process

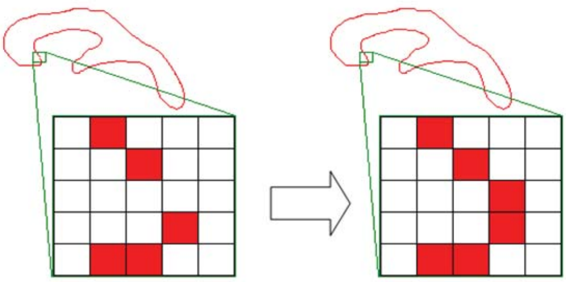

Fig. 5 Closing of the edge

3. Closing the Edge

The closing of the edge is a difficult step especially in medical images. In this system, an 'end edge agent' aims to be connected with the nearest 'end edge agent'. Nevertheless, it is connected only if the distance between itself and this nearest agent is lower than a threshold (Fig. 5).

IV. Results and Discussion

In this section, the experimental results are presented. The MAS has been evaluated by an extensive range of tests. Some images are employed for evaluation as synthetic images (Fig. 4) and MR Images (Fig. 5). The program is written using JAVA language and use MATLAB package. Experiments is done on a core duo (2.2 GHz, 2 GB), using the Windows XP. We have used a single square window 3×3 which is kept constant through all the experiments, $\gamma = 0{,}17$ and $\varepsilon = 0{,}01$. For more precise edge detection and to overcome some limit in this approach, such as edge closing, more than two directions to follow and sensitivity to noise, Observer agent divides the image into small rectangles in which 'scouts' are initialized in the middle. Traditionally, the edge detection was performed manually, which is a time-consuming procedure with results affected by high inter- and intra-user variability. However, in clinical practice the most attractive approaches are the fully automated ones. The proposed method is validated against manual edge detection, which is considered as the reference technique.

Fig. 6 Synthetic images used in the test

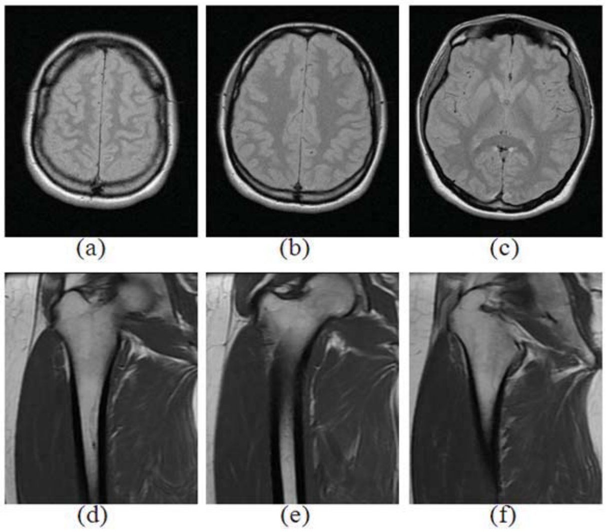

Fig. 7 IRM used for test: (a)-(c) human brain in axial view and (d)-(f) human femur in sagittal view

The results (Figs. 8 and 9) show that the proposed method is in good agreement compared with others works that use the Gradient Vector Fellow (GVF) [20], [21]. The comparisons with GVF approach show the advantages of this system, including superior noise robustness and reduced computational cost. Cooperation between agents overcomes the problem of strong, weak, partial, hidden or edges linking. Moreover, increasing the number of parallel agents increases the probability of finding the best detection and decreases the run time process.

Fig. 8 Results of detection for synthetic images

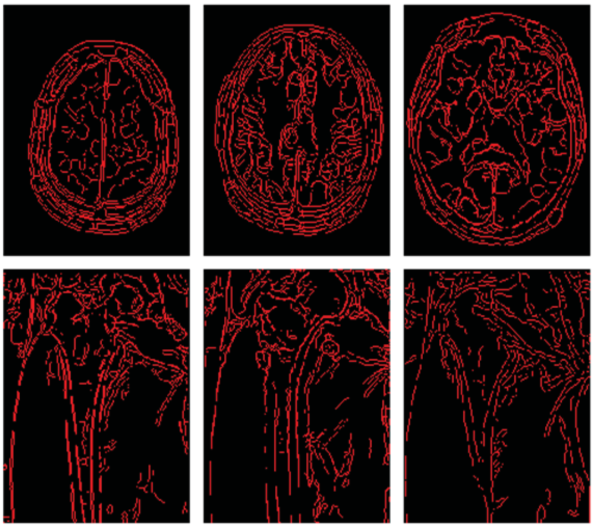

Fig. 9 Results of detection for MR images

TABLE I
MAXIMUM RMSE IN NOISY SYNTHETIC IMAGES COMPARED WITH THE SAME IMAGES SEGMENTED MANUALLY

tests	Maximum RMSE (pixel)	
	Our approach	GVF Model [21]
1	5	8
2	6	9
3	6	7
4	4	6
5	4	6
6	5	8
7	4	7
8	5	10
9	5	6
10	5	10

To evaluate the noise sensitivity, we add impulse noise to the synthetic test images. The results reveal the superior robustness to noise afforded by the VFC field. To quantify the accuracy of the results, the Root Mean Square error (RMSE) of the edge result is calculated. The error of an edge pixel is defined by the minimum distance between this pixel and the object pixels in the image. Table I shows the performance criterion using this metric. The results of RMSE are smallest compared with others reported in the literature and the total time detection was approximately 2X to 4X faster.

V. CONCLUSION

We have presented in this paper a multi-agent approach for edge detection in image using VFC. The VFC field is calculated by convolving a vector field kernel with the edge map generated from the image. As a distributed system, the MAS contains a set of agents working together and can generate an overall segmentation.

The system has been applied in different images which were segmented by agents. The contours which were detected automatically can represent complex shapes and overcome the difficulties caused by noise.

Future work is directed towards more investigations on evolutionary computation techniques and the development of new approach to be applied to a wider range of problems with a higher reliability.

REFERENCES

[1] S. J. Ros, N. Andarawis-Puri, E. L. Flatow, ''Tendon extracellular matrix damage detection and quantification using automated edge detection analysis'' Journal of Biomechanics 46, pp: 2844–2847, 2013
[2] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models. Computer Vision (1), pp: 21–31, 1988.
[3] P. P. R. Filho, P. C. Cortez, A. C. S. Barros, V. H. Albuquerque ''Novel Adaptive Balloon Active Contour Method based on internal force for image segmentation – A systematic evaluation on synthetic and real images'' Expert Systems with Applications, pp: 7707—7721, 2014.
[4] A. K. Mohanty, M. R. Senapati, S. K. Lenka, A novel image mining technique for classification of mammograms using hybrid feature selection. Neural Computing and Applications, 22(6), pp: 1151–1161, 2013.
[5] A. Nachour, L. Ouzizi, Y. Aoura, Multi-Agent 3D Reconstruction of Human Femur from MR Images. 15th International Conference on Intelligent Systems Design and Applications, 2015.
[6] T. Arai, H. Ogata, T. Suzuki, Collision avoidance among multiple robots using virtual impedance. In Proceedings of the IEEE/RSJ international conference on intelligent robots and systems. pp: 479–485, 1989.
[7] Z. Li, J. Liu. A multi-agent genetic algorithm for community detection in complex networks. Physica A: Statistical Mechanics and its Applications, 449. Pp: 336-347, 2016.
[8] A. Kasaiezadeh, A. Khajepour, Multi-agent stochastic level set method in image segmentation. Computer Vision and Image Understanding 117 pp: 1147–1162, 2013.
[9] A. Guillaud et al, ''A Multiagent Systemfor Edge Detection and Continuity Perception on Fish Otolith Images'' Journal on Applied Signal Processing, 7, pp: 746–753, 2002.
[10] B. Li, ''Active Contour External Force Using Vector Field Convolution for Image Segmentation'' IEEE transaction on image processing, 16, pp: 8, 2007.
[11] V. Rodin, F. Harrouet, P. Ballet, J. Tisseau, oRis: Multiagents Approach for Image Processing, in: H. Shi, P.C. CoJeld (Eds.), SPIE Conference on Parallel and Distributed Methods for Image Processing II, Vol. 3452, SPIE, San Diego, CA, pp. 57–68, 1998.
[12] J. Liu, Y.Y. Tang, Adaptative image segmentation with distributed behavior based agents, IEEE Trans. Pattern Anal. Mach. Intell. 6, pp: 544–551,1999.
[13] E.G.P. Bovenkamp, J. Dijkstra, J.G. Bosch, J.H.C. Reiber, Multi-agent segmentation of IVUS images. Pattern Recognition 37, pp: 647 – 663, 2004.
[14] H. Settache, C. Porquet, S. Ruan. Une plate-forme multi agents pour lasegmentation d'images: application dans le domaine des IRM cérébrales 2D. Technical report, Université de Caen, 2002.
[15] J. Fleureau, M. Garreau, D. Boulmier, C. Leclercq and A. Hernandez, Segmentation 3D multi-objets d'images scanner cardiaques: une approche multi-agent. In IRBM 30 pp:104-11, 2009.

[16] F. Bellet, Une approche incrémentale, coopérative et adaptative pour la segmentation des images en niveau de gris. Institut National Polytechnique de Grenoble, France, 1998.
[17] K. Yanai. An image understanding system for various images based on multi-agent architecture, December 1999.
[18] F. Bellifemine, A. Poggi, G. Rimassa, JADE A FIPA compliant agent framework, CSELT internal technical report. Proceedings of PAAM'99, London, pp: 97—108, 1999.
[19] FIPA: Foundation for Intelligent Physical Agents, Agent Communication Language, FIPA 99 Specification Draft, 1999.
[20] N. Paragios, O. Gotardo, V. Ramesh, Gradient vector flow fast geodesic active contours. pp 67–75.
[21] L. Je. Prince, C. Xu. Gradient Vector Flow: A New External Force Model for Snakes. In IEEE Image and Multidimensional Signal Processing Workshop, pp 30–31, 1996.