An Agent-Based Approach for Range Image Segmentation

Smaine Mazouzi[1], Zahia Guessoum[2], Fabien Michel[1], and Mohamed Batouche[3]

[1] MODECO-CReSTIC, Université de Reims, B.P. 1035, 51687, Reims, France
{mazouzi,fmichel}@leri.univ-reims.fr
[2] LIP6, Université de Paris 6, 104, av. du Président Kennedy, 75016, Paris, France
zahia.guessoum@lip6.fr
[3] Département d'informatique, Université de Constantine, 25000, Algérie
batouche@wissal.dz

Abstract. In this paper an agent-based segmentation approach is presented and evaluated. The approach consists in using a high number of autonomous agents for the segmentation of a range image in its different planar regions. The moving agents perform cooperative and competitive actions on the image pixels allowing a robust extraction of regions and an accurate edge detection. An artificial potential field, created around pixels of interest, allows the agents to be gathered around edges and noise regions. The results obtained with real images are compared to those of some typical methods for range image segmentation. The comparison results show the potential of the proposed approach for scene understanding in range images regarding both segmentation efficiency, and detection accuracy.

Keywords: Image segmentation, Multi-agent systems, Range image, Artificial potential field.

1 Introduction

The segmentation of an image is often necessary to provide a compact and convenient description of its content, suitable for high level image analysis and understanding. It consists in assigning image pixels to homogenous and disjoint subsets, which form an image partition. The pixels which belong to the same region share a common feature called the region homogeneity criterion. In range images, segmentation methods can be divided into two distinct categories: edge-based segmentation methods and region-based segmentation methods. In the first category, pixels which correspond to discontinuities in depth or in surface normals are selected and chained in order to delimit the regions in the image [8,4,10]. Edge-based methods are well known for their low computational cost, however they are very sensitive to noise.

Region-based methods use geometrical surface descriptors to group pixels with the same proprieties in disjoint regions [22,11,13,3,1]. Compared to edge-based methods, region-based methods are more stable and less sensitive to noise. However, their efficiency depends strongly on the selection of the region seeds. Most

N. Jamali et al. (Eds.): CCMMS 2007, MMAS 2006, LSMAS 2006, LNAI 5043, pp. 146–161, 2008.

of the time, the segmentation results in an over-partition of the image. So, it is necessary to perform an iterative fusion of homogenous regions. Such an approach does not facilitate the distribution of the used algorithms, and leads to high computational costs [7].

Furthermore, most of the proposed methods model surface proprieties by computing image derivatives of different orders. Such techniques result in a highly noise-sensitive detection. It is then necessary to perform some preprocessing tasks, which consist mostly in image smoothing and noise filtering. However, in the case of highly noisy images such as range images [7], a strong noise smoothing can erase the roof edges (at discontinuities of surface normals), and the smooth edges (at discontinuities of curvature) whose detection remains a challenge. Moreover, if the noise is under-smoothed the distortions, which remain in the image, result in inaccurate or erroneous segmentation results. This difficulty, which is an open issue in image segmentation [14], results from the restriction of computation and decision to the local neighborhood of the processed pixel. In range images, several recent segmentation methods fail because they do not correctly address and resolve this problem [9].

To deal with this problem, some authors have proposed agent-based solutions for 2-D image segmentation. Agent-based solutions inherit the advantages of the agent-oriented systems for collective problem solving. In such systems a single agent has a limited perception and limited capabilities, and it is not designed to solve an entire problem. Agents cooperate thus in order to provide a collective solution. Contrary to conventional systems, solutions in agent-based systems emerge from collective action of interactive agents [12].

In this paper, a new agent-based solution for range image segmentation is presented and discussed. It consists in the use of reactive agents, which move over the image, and act on the visited pixels. While moving over the image, an agent adapts to the planar region on which it moves, and memorizes its proprieties. When an agent encounters a pixel which does not belong to its current region, an agent alters this pixel in order to align it to its current region. At the boundaries between the regions the agents will be in competition to align the pixels of the boundaries to their respective regions. The resulting alternative alignment of the boundary pixels preserves the region boundaries against erasing. Noise regions that are characterized by small sizes or by aberrant depths (outliers) prevent the agents from adapting. Thus, these regions continuously contract by aligning their pixels to the planar regions which surround them.

Our aim is to overcome the difficulty related to the local perception around the processed pixel. A pixel is therefore processed according to both its neighborhood, and the agents that visit this pixel. An agent acts on pixels with more certainty, acquired from its move on large areas on the image regions. The combination of the global information memorized within the agent, and the local information, provides more reliable decisions. To optimize the agent movements, an artificial potential field inspired from the electrostatic potential field is used. It allows to rationalize the movements of the agents by directing them to be gathered around the regions of interest (edges and noise regions) and to

concentrate their actions around these regions. The utilization of a large number of reactive and weakly coupled agents provides a massively multi-agent system, allowing a parallel and distributed image segmentation. Extensive experimentations have been performed using real images from the ABW database [7]. The obtained results show the high potential of the proposed approach for an efficient and accurate segmentation of range images.

The remainder of the paper is organized as follows: In Section 2, we review the agent-based approaches for image segmentation. Section 3 introduces the surface proprieties modeling. Section 4 is devoted to the proposed approach. It describes the behavior of the agents and shows the underlying collective mechanism to deal with the edge detection and the noise removal. The experimental results are introduced in Section 5, in which we discuss the selection of the used parameters, and we analyse and comment the obtained results. Finally, a conclusion summarizes our contribution.

2 Related Work

Several agent-based systems have been proposed for image analysis and object recognition. They propose interesting solutions to deal with several problems, such as multiple domain knowledge handling, control automation over the image interpretation tasks, collective segmentation, and distributed and parallel image processing. In this review we consider only works which have addressed a solution in image segmentation.

Liu et al. [16] introduce a reactive agent-based system for brain MRI segmentation. They underline that the employed agents are more robust and more efficient than the classical region-based algorithms. Four types of agents are used to label the pixels of the image according to their membership grade to the different regions. When finding pixels of a specific homogenous region an agent creates offspring agents into its neighborhood. In this system, the agents neither interact directly between them nor act on the image. Their actions depend only on their local perception. Nevertheless, each agent is created so that it becomes more likely to meet more homogenous pixels. For the same type of images, Richard et al. [18] propose a hierarchical architecture of situated and cooperative agents for brain MRI segmentation. Three types of agents have been used: global control agent, local control agent, and tissue dedicated agent. The role of the global control agent is to partition the volume of data into adjacent territories and to assign one local control agent to each territory. The role of a local control agent is to create the tissue dedicated agents which perform a local region growing. The statistical parameters of the image data distribution, needed to perform region growing are updated according to the interaction between neighboring agents. Using several types of agents has allowed to deal with both the control over the high-level segmentation tasks and the low-level image processing tasks.

The two previous systems are well optimized to brain MRI segmentation. They can provide interesting results because region characteristics are regular in the different brain anatomic parts. In addition, most of the edges in such images are

jump edges (at discontinuities of image data) which are easy to detect, compared to roof and smooth edges.

Rodin et al. [19] using the oRis langauge [6], have presented a reactive agent-based system for edge detection in biological images. According to some prior on image content, the system provides an edge detection which is better than that provided by traditional detectors. Two groups of agents, called darkening agents and lightning agents follow respectively the dark regions and the light regions. Their actions aim at reinforcing regions by stressing their contrast, allowing a reliable detection of these regions. In this system, agents are fully independent from each other, and never interact. The system can be considered as a parallel segmentation algorithm which was well optimized for the detection of roof edges in some biological images. However, agents were not designed to detect discontinuities of image data. So, the system may fail to detect jump edges. Furthermore, the number and the topology of the expected regions must be known, and hard coded within the agents.

Based on the cognitive architecture Soar [17], Bovenkamp et al. [2] have developed a multi-agent system for IntraVascular UltraSound (IVUS) image segmentation. They aim to elaborate a high knowledge-based control over the algorithms of low-level image processing. In this system, an agent is assigned to every expected object in the image. Agents cooperate and dynamically adapt the segmentation algorithms, according to contextual knowledge, local information and their personal believes. In this work, the problem of the control over segmentation algorithms seems to be well resolved. However, no agent or even behavior has been proposed to deal with the problem of uncertain and noisy data.

The proposed agent-based systems for image segmentation are specific to image contents. Following a supervised approach, these systems aim at segmenting images in known and previously expected regions. The system proposed in this paper claims to be general and unsupervised. It aims to segment an image into its several regions by using some invariant surface proprieties. The adaptive and competitive behavior of the agents allow overcoming the constraint related to the restriction of the treatments to the local neighborhood of pixels. We show in this work that despite the simplicity of the model used to represent surfaces, the obtained results are better than those provided by conventional approaches. We believe that interactions between agents provide an alternative way for image segmentation to that of methods based on complicated and costly models [15].

3 Surface Modeling

A range image is a discretized two-dimensional array where at each pixel (x, y) is recorded the distance $Z(x, y)$ between the range finder and the corresponding point of the scene. Regions in such an image represent the visible patches of object surfaces. To attenuate the white and the impulsive noise contained in the image, a Gaussian filter and a median filter are applied to the depth raw data. A new image $Z^*(x, y)$, called plane image is then derived from the range image.

Each pixel (x, y) of the new image records the tangent plane to the surface at (x, y). The best tangent plane at (x, y) is obtained by the multiple regression method using the set of neighboring pixels $\chi(x, y)$. The neighborhood $\chi(x, y)$ is made up of pixels (x', y') situated within a 3×3 window centred at (x, y), and whose depths $Z(x', y')$ are close, according to a given threshold (Tr_D).

The plane equation in a $3-D$ coordinate system may be expressed as follows:

$$ax + by + cz = d \tag{1}$$

where $(a, b, c)^T$ is the unit normal vector to the plane $(a^2 + b^2 + c^2 = 1; c < 0)$ and d is the orthogonal distance between the plane and the coordinate origin. First, Parameters α, β and γ of the surface $z = \alpha x + \beta y + \gamma$ at (x_0, y_0) are obtained by the minimization of the function Φ, defined as follows:

$$\Phi(\alpha, \beta, \gamma) = \sum_{(x_i, y_i) \in \chi(x_0, y_0)} [\alpha x_i + \beta y_i + \gamma - Z(x_i, y_i)]^2 \tag{2}$$

with

$\chi(x_0, y_0) = \{(x_0 + i, y_0 + j); (i, j) \in \{-1, 0, +1\}$ and $|Z(x_0 + i, y_0 + j) - Z(x_0, y_0)| < Tr_D\}$

Parameters a, b, c and d are thus obtained as follows:

$$(a, b, c, d)^T = \frac{1}{\sqrt{\alpha^2 + \beta^2 + 1}} (\alpha, \beta, -1, \gamma)^T \tag{3}$$

The tasks performed on the plane image are based on the comparison of planes. Indeed, we consider that two planes $ax+by+cz = d$ and $a'x+b'y+c'z = d'$ are equal if they have, according to given thresholds, the same orientation and the same distance to the coordinate origin. Let θ be the angle between the two normal vectors, and D the distance between the two planes: $\sin(\theta) = \|(a, b, c)^T \otimes (a', b', c')^T\|$ and $D = |d-d'|$. So, the two planes are considered equal if $\sin(\theta) \leq Tr_\theta$ and $D \leq Tr_D$, where Tr_θ and Tr_D are respectively the angle and the distance thresholds. Plane comparison is first used to test if a given pixel belongs to a planar region, given its plane equation. It is also used to test if the pixel is, or is not, a pixel of interest (edge or noise pixel). In this case, the pixel in question is considered as a pixel of interest if at least one of its neighbors has a different plane equation, according the previous thresholds.

4 Multi-agent Range Image Segmentation

The plane image is considered as the environment in which agents are initialized at random positions. An agent checks if it is situated within a planar region, and adapts this region if planar, by memorizing its plane equation. Next, the agent performs actions which depend on both its state and the state of the pixel

on which it is located. At each time t, an agent is characterized by its position (x_t, y_t) over the image, and by its ability A_t to act on the encountered pixels. At the beginning of the process, all the agents are unable to alter the image pixels. After having been adapted to a planar region, an agent becomes able to modify the first pixel not belonging to this region (A_t=true). When an agent alters a pixel, it loses its alteration ability (A_t=false) and starts again searching for a new planar region. An agent having modified a pixel records in an appropriate array I at the position (x_t, y_t) the last state of the visited pixel: $I(x_t, y_t) \in$ {smoothed, aligned, unchanged}. We show next, that this simple behavior of the agents allow both the detection of the image edges, and the removal of the noise regions.

4.1 Agent Behavior

An agent adapts to the region of the image on which it is moving by computing and storing the proprieties of this region, and by adopting the suited behavior to the local image data. Fig. 1 depicts the behavior of an agent according to its state and the state of the pixel on which it is located.

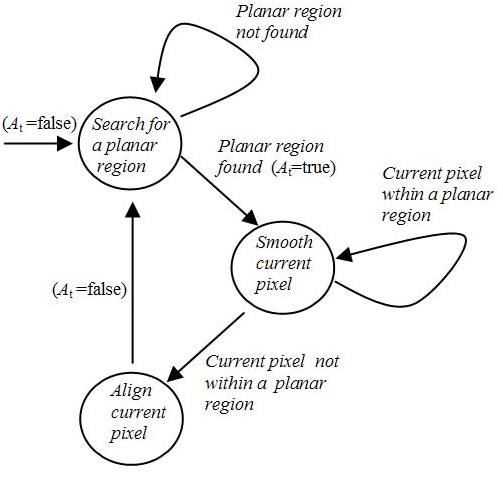


Fig. 1. Agent behavior according to its state and position

Searching for a Planar Region. After its creation, an agent randomly moves within the image and searches for a planar region around its current position. The seed of the searched region is formed of the last L pixels visited by the agent. L is called the adaptation path-length. It represents the confidence degree that the agent is situated within a planar region. The agent considers that it is within a planar region if the pixels of the seed form a planar surface. The agent memorizes the new region and considers it as its current planar region. It agent becomes then able to alter the first encountered pixel which does not belong to its planar region (A_t=true).

Moving on a Planar Region. While moving inside a planar region, an agent smoothes the pixel on which it is located by updating the equations of both the memorized plane and the plane at the pixel position. This is done by replacing the two equations by their weighted average. Let (a', b', c', d') and (a, b, c, d) be the parameters respectively of the memorized plane and the plane at the current pixel. Resulting parameters of the weighted average plane, before normalization, are obtained as follows:

$$(a'', b'', c'', d'') = \frac{1}{1+l}(a + la', b + lb', c + lc', d + ld') \tag{4}$$

where l is the length of the path crossed by the agent on the planar region.

Pixel Alignment. Pixels of interest are edge pixels or pixels within noise regions. When an agent meets a pixel of interest (i.e. not belonging to its current planar region) it alters it in order to partially align it to the planar region on which it is moving. Parameters (a'', b'', c'', d'') of the new plane equation at the pixel position are obtained by linear combination of the current parameters (a, b, c, d) and the parameters of the memorized plane equation (a', b', c', d'):

$$(a'', b'', c'', d'') = \frac{1}{1+\xi}(a + \xi a', b + \xi b', c + \xi c', d + \xi d') \tag{5}$$

where ξ is the alteration strength.

The agent loses then its alteration ability (A_t=false) and starts again to search a new planar region. Further to the alteration of a pixel, the agent can pass into another region, or remain in the current region. If the altered pixel is an edge pixel, the agent likely pass in an other planar region. However, if the altered pixel is on the boundary of a noise region, the agent crosses the noise region and most likely end up in the previous planar region, except if the noise region is situated between two planar regions. The alteration strength ξ is a critical parameter which affects the quality of the results and the time of computation. Indeed, high values of ξ lead to a fast detection of regions. However the resulting region boundaries are distorted and badly localized (Fig. 2c). Low values of ξ result in a slow detection, but region boundaries in this case are well detected and correctly localized (Fig. 2d). In order to speed up the segmentation process, and avoid edge distortion, an agent chooses the alteration strength among ξ_{min} and ξ_{max} according to the information recorded by other agents in the array I. So, an agent assumes that the current planar region is adjacent to a noise region and thus uses ξ_{max} as alteration strength, if the number of "unchanged" pixels (situated in a noisy region) around the agent is greater than a given threshold (fixed to 3 in our experimentations). Indeed, pixels labeled "unchanged" in the adjacent region mean that this latter is a noise region for which agents have not adapted and consequently have not smoothed its pixels. Otherwise the agent assumes that the current planar region is adjacent to another one, where other agents have labeled the pixels as "smoothed" or "aligned". In this case the agent uses the alteration strength ξ_{min}.

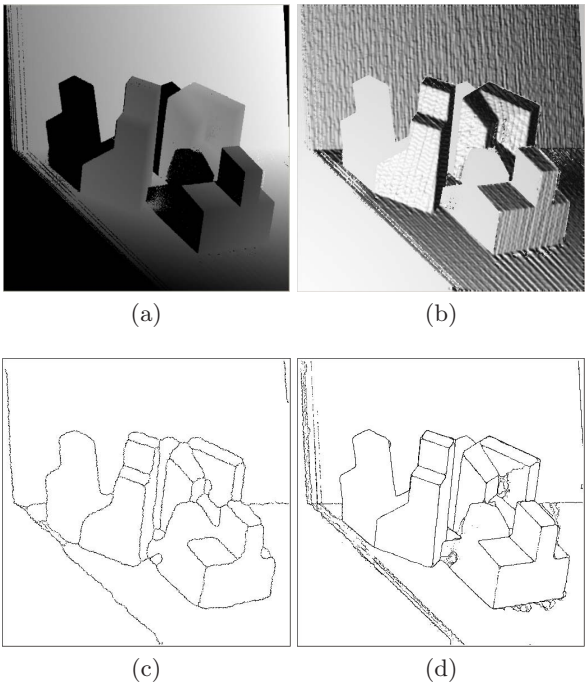


Fig. 2. The impact of the alteration strength on the segmentation results: (a) Range image (abw.test.8); (b) Rendered range image; (c) Segmentation results with $\xi_{min} = \xi_{max} = 4$ at t=2500; (d) Segmentation results with $\xi_{min} = 0.3$ and $\xi_{max} = 5$ at t=13000

4.2 Agent Coordination by Artificial Potential Field

To endow the agents with a self-organization mechanism, an artificial electrostatic like potential field is used. It is created and updated around the aligned pixels. It allows agents to be gathered around pixels of region boundaries, and concentrate their actions at these pixels. Contrary to other works, where the potential field is created at known positions of objects (goals and obstacles) [5,21,20], the potential field in our case results from the interaction of agents with the objects in the environment (pixels). The intensity $\Psi(x, y)$ of the potential field at position (x, y) created by a set of P pixels beforehand aligned $\{(x_i, y_i), i = 1..P \wedge I(x_i, y_i)$=aligned$\}$ is given by:

$$\Psi(x, y) = \sum_{i=1}^{P} \frac{k}{\sqrt{(x - x_i)^2 + (y - y_i)^2}}, k \in R^+ \tag{6}$$

where k is the constant of the electrostatic force, set to 1.

An agent which is able to alter pixels (A_t=true) and situated at position (x_t, y_t) undergoes an attractive force $\overrightarrow{F}$. This force is expressed by the gradient vector of the potential field:

$$\overrightarrow{F} = \begin{cases} -\overrightarrow{\nabla}\Psi(x_t, y_t) \text{ if } A_t{=}\text{true} \\ \overrightarrow{0} \qquad\qquad\quad \text{otherwise} \end{cases}$$

So, the agent movements, which are stochastic in nature, are weighted by the attractive force applied by the potential field. Agents are influenced to head for the pixels of interest, while keeping random moves. The random component of the agent moves allows the exploration of all regions of the image.

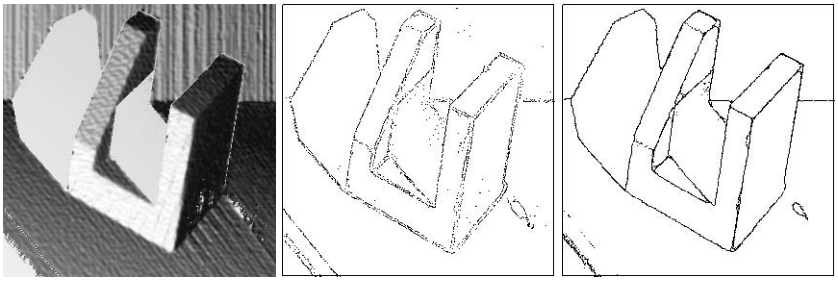
A Relaxation mechanism of potential field is also introduced. It allows the agents gathered around pixels of interest to be released and thus to explore other regions of the image. Around a given pixel, the field intensity decreases after every alteration of this pixel. The equation of the relaxation dynamic is expressed as follows:

$$\Psi_{t+1}(x, y) = \mu \times \Psi_t(x, y), \mu < 1 \tag{7}$$

$\Psi_0(x, y)$ corresponds to the created field after the first alteration of the pixel. The constant μ set to 0.9, represents the decrease rate of the field intensity. After several alignements of a given pixel, the field intensity around this pixel decreases, and tends to zero. This situation represents the final state of the process, after which the system can be stopped.

4.3 Edge Detection and Noise Removal

While moving over the image, agents smooth pixels that approximatively belong to their respective planar regions. An agent considers pixels that do not belong to its current region as noise pixels. The agent aligns thus automatically these pixels to its current planar region (Fig. 3b). However, pixels on the boundaries of planar regions are true-edge pixels and thus should not be aligned. Nevertheless, the competition between agents preserves these pixels against an inappropriate smoothing. Indeed, around the edge between two adjacent planar regions, two groups of agents are formed on the two sides of this edge. Each group is formed of agents passing from one region to the other. Agents of each group align the pixels of the boundary to their respective region. So, pixels of the boundary are continuously swapped between the two adjacent regions. This allows these pixels to remain emergent in the image (Fig. 3c). This pattern of competitive actions between agents allows the emergence of image edges. The edge map is not coded in any agent, it results from the collective actions of the agents. Unlike true regions of the image, which remain preserved against erasing, noise regions continuously narrow, and they finally disappear. Borders of these regions are continuously aligned to the true planar regions, that surround them. An agent, having aligned a pixel which belongs to the border of a noise region and having moved inside this region, will not be able to adapt. Consequently, it cannot align any pixel when leaving the noise region. This occurs in two distinct situations: 1) when a region is planar but insufficiently large to allow agents to cross the minimal path-length L necessary to be able to adapt; 2) when a region is sufficiently



(a) (b) (c)

Fig. 3. Underlying edge detection (image abw.test.22): (a) Rendered range image; (b) Aligned pixel at t=800 ; (c) Only edge pixels are emergent at t= 8000

large but not planar, or made of random depths (noise). In both situations, the agent leaves the noise region and will adapt inside the surrounding planar region. The true regions have large sizes sufficient to allow agent to adapt and then align boundary pixels when leaving these regions. However, noise regions, which are non planar, or having weak size, prevent agents from adapting. Consequently, agents will be unable to align pixels on the boundaries of these regions when leaving them. As a result, boundaries of these regions are continuously aligned from outside by including their pixels in the true surrounding regions. After several execution steps, these regions will be completely erased.

After Several iterations, all image regions are well delimited by the detected boundaries. A simple region growing steered by the detected boundaries allows to extract the regions of the image.

5 Experiments and Analysis

5.1 Evaluation Framework

Hoover et al. have proposed a dedicated framework for the evaluation of range image segmentation algorithms [7], which has has been used in several related works [10,9,13,3,1]. The framework consists of a set of real range images, and a set of objective performance metrics. It allows to compare a machine-generated segmentation (MS) with a manually-generated segmentation, supposed ideal and representing the ground truth (GT). The most important performance metrics are the numbers of instances respectively of correctly detected regions, over-segmented regions, under-segmented regions, missed regions, and noise regions. Region classification is performed according to a compare tool tolerance T; $50\% < T \leq 100\%$ which reflects the strictness of the classification. The 40 real images of ABW database are divided into two subsets: 10 training images, and 30 test images. The training images are used to estimate the parameters of a given segmentation method. Using these parameters, the method is applied to the test images. The Performance metrics are computed and stored in order to

be used to compare the involved methods. In our case, four methods, namely USF, WSU, UB and UE, cited in [7] are involved in the result comparison.

5.2 Parameter Selection

Since the evaluation framework provides a set of training images with ground truth segmentation (GT), we have opted to a supervised approach for the selection of parameters. For our System, named 2ARIS for Agent-based Approach for Range Image Segmentation , six parameters should be set: ξ_{min}, ξ_{max}, Tr_θ , Tr_D, N, and L. These parameters are divided into two subsets: 1) ξ_{min}, ξ_{max}, Tr_θ, and Tr_D represent respectively the two alignment strengths, the angle threshold, and the depth threshold. These parameters are used for testing and aligning image pixels, and 2) N and L represent respectively the number of agents, and the adaptation path-length. These two parameters control the dynamic of the multi-agent system. For the first parameter subset, 256 combinations namely (ξ_{min}, ξ_{max}, Tr_θ, Tr_D) $\in \{0.5, 0.3, 0.1, 0.05\} \times \{1.0, 3.0, 5.0, 7.0\} \times \{15, 18, 21, 24\} \times \{12, 16, 20, 24\}$ were run on the training images. The performance criterion for this parameters is the average number of correctly detected regions with the compare tool tolerance T set to 80%. The two alignment strengths ξ_{min} and ξ_{max} are set respectively to 0.3 and 5.0. These values have provided a good edge detection in a reasonable execution time. The threshold Tr_θ was set to 20. We have observed that higher values of this parameter under-differentiate regions regarding their orientations, and lead to an under-segmentation of the image. However, lower values over-differentiate regions, and lead to an over-segmentation. It results in a high number of false and small regions, which should be merged in the true neighboring regions. Finally, the threshold Tr_D is set to 16. Note that values significantly greater than 16 can lead to wrongly merge some parallel overlapped regions. However, if Tr_D is significantly less than 16, highly sloping regions cannot be detected as planar regions [10]. This results in a high rate of missed regions.

The number of employed agents N depends on the size of the image, while the adaptation path-length L depends on the level of detail of the image. These two parameters are critical and must be carefully selected. Inappropriate values of these two parameters can lead to a high rate of segmentation errors. Indeed, an insufficient number of agents lead to an under-processing of the image. So, resulting regions are deprived of a set of pixels which should be included in these regions. A low value of the adaptation path-length L leads to take into account small planar regions which should be considered as noise regions. However, higher values of L can result in missing some true planar regions which are insufficiently large (see section 4.3). In order to set the parameters N and L, 25 combinations of these parameters, namely $(N, L) \in \{1500, 2000, 2500, 3000, 3500\} \times \{3, 5, 7, 9, 11\}$ were run on the training set. In this case, the performance criterion is the average number of noise regions, with the compare tool tolerance set to 80%. Obtained optimal values of N and L are respectively 2500 and 7.

5.3 Experimental Results

Fig. 4 shows an instance of segmentation progression within time. The time t represents the number of steps performed by each agent since the beginning of the process. Displaying a range image by a simple rendering algorithm (Fig. 4a), allows observing the high level of noise in the used images. Figures 4b, 4c, 4d and 4e show the set of pixels of interest (edge or noise pixels) respectively at t=1000, 5000, 9000 and 13000. Regions are progressively smoothed by aligning noise pixels to the surrounding planar regions. Edges between adjacent regions are also progressively thinned. At the end of the process, region borders consist of thin lines of one pixel wide (Fig. 4e). Fig. 4f shows the segmentation result obtained by displaying the borders of the extracted regions.

Fig. 5 shows the segmentation results of the image abw.test.8, with the compare tool tolerance T set to 80%. This image was considered as a typical image to compare the involved methods [7],[3]. Fig. 5a shows the range image, and Fig. 5b shows the ground truth segmentation (GT). Fig. 5c, 5d 5e and 5f are segmentation results obtained respectively by USF, WSU, UB and UE methods. Fig. 5g presents the segmentation result obtained by our method. Metrics in table 1 show that all image regions detected by the best-referenced segmenter

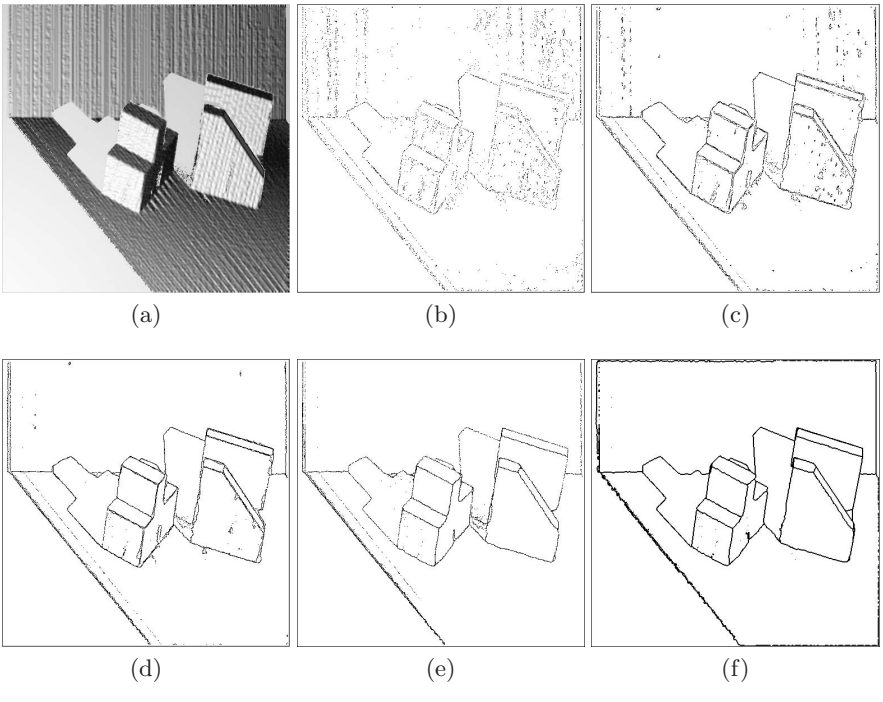


Fig. 4. Segmentation progression. (a) Rendered range image (abw.test.6); (b) at t=1000, (c) at t=5000 ; (d) at t=9000 ; (e) at t=13000 ; (f) Segmentation result (Extracted regions)

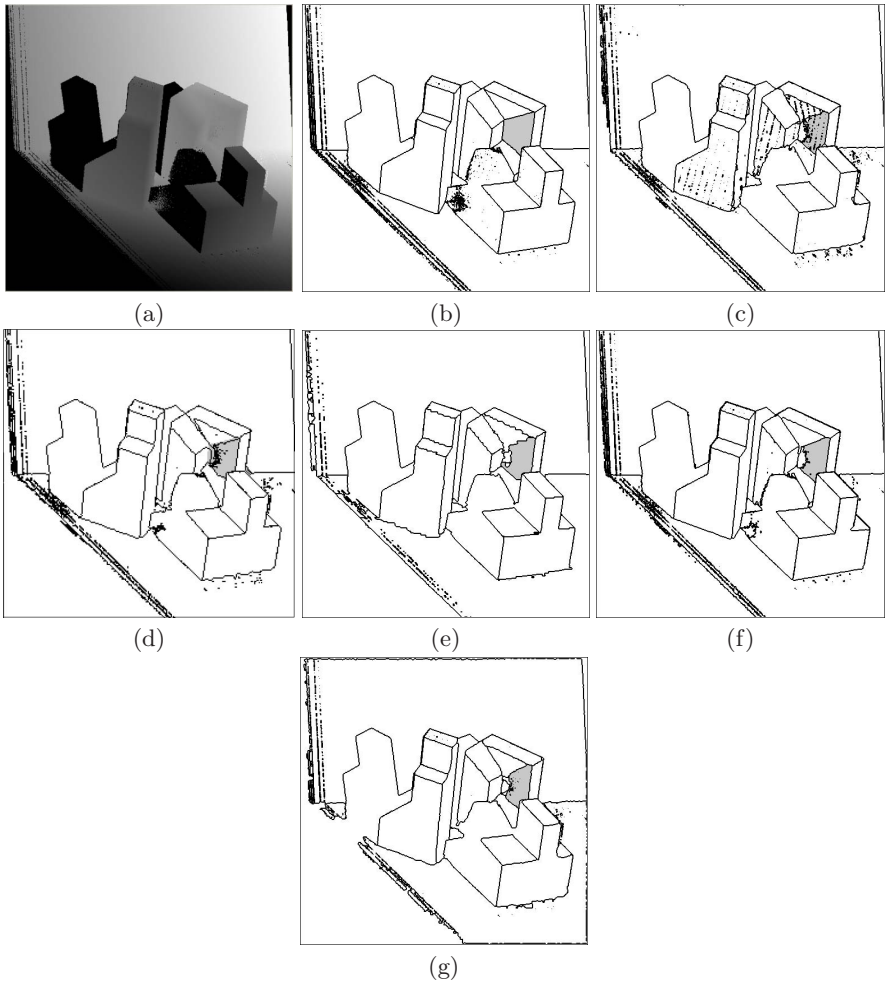


Fig. 5. Segmentation results of abw.test.8 image. (a) Range image; (b) Ground truth segmentation (GT); (c) USF result; (d) WSU result; (e) UB result; (f) UE result; (g) 2ARIS result

(UE) were detected by our method. Except the shadowed region, where all methods fail to detect, all object regions were detected. The incorrectly detected regions are those with small sizes, and situated on the horizontal support. Compared to the other methods, values of incorrect detection metrics are also good. Our method is equivalent to UE, and scored higher that the others.

Table 2 contains the average results obtained with all test images, and for all performance metrics. The compare tool tolerance was set to the typical value 80%. By considering both correct detection and incorrect detection metrics, obtained results show the good efficiency of our method. Fig. 6 shows the average numbers of correctly detected regions for all test images, and according

Table 1. Comparison results with abw.test.8 image for T=80%

Method	GT region	Correct detection	Over-segmentation	Under-segmentation	Missed	Noise
USF	21	17	0	0	4	3
WSU	21	12	1	1	6	4
UB	21	16	2	0	3	6
UE	21	18	1	0	2	2
2ARIS	21	18	2	0	1	1

Table 2. Average results of the different involved methods with T=80%

Method	GT region	Correct detection	Over-segmentation	Under-segmentation	Missed	Noise
USF	15.2	12.7	0.2	0.1	2.1	1.2
WSU	15.2	9.7	0.5	0.2	4.5	2.2
UB	15.2	12.8	0.5	0.1	1.7	2.1
UE	15.2	13.4	0.4	0.2	1.1	0.8
2ARIS	15.2	13.0	0.5	0.1	1.4	0.9

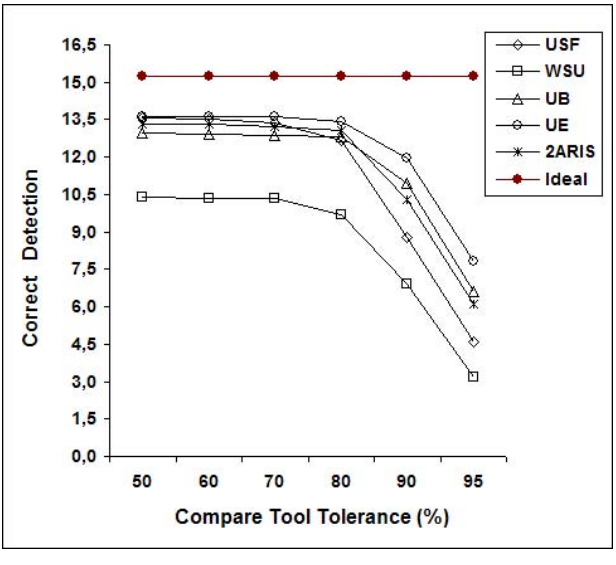


Fig. 6. Average results of correctly detected regions of all methods, according to the compare tool tolerance T ; $0.5 < T \leq 1.0$

to the compare tool tolerance T; $T \in \{51\%, 60\%, 70\%, 80\%, 90\%, 95\%\}$. Results show that the number of correctly detected regions by our system is in average better than those of USF, UB and WSU. For instance, our system scored higher than WSU for all the values of the compare tool tolerance T. It

scored higher than USF for $T \in \{80\%, 90\%, 95\%\}$, and better than UB for $T \in \{50\%, 60\%,70\%, 80\%\}$. For all incorrect detection metrics (instances of Over-segmentation, Under-segmentation, Missed Region, Noise Region), our system has equivalent scores to those of UE and USF. The two latter scored higher than UB and WSU.

6 Conclusion

In this paper we have introduced an agent-based approach for range image segmentation. Indirect interaction between autonomous agents moving over the image allows reliable edge detection and efficient noise removal. Competitive actions between agents that are self-gathered around region boundaries have allowed the emergence of image edges. Image edges, for which no explicit detection was coded in any agent, result from the collective action of all the agents. The proposed approach aims to improve efficiency and to deal with the problem of result accuracy. Indeed, obtained results are better than those provided by traditional algorithms, based on region growing techniques. Moreover, employed agents are weakly coupled, and indirectly communicate via the environment (image). This allows parallel or distributed implementations, necessary to obtain a high computational efficiency. Experimental results obtained with real images from ABW database were compared to those provided by four typical algorithms for range image segmentation. Comparison results show a good potential of the proposed method for both segmentation efficiency and accuracy. The proposed approach can be extended to deal with more complex surfaces by defining their specific proprieties, and endowing the agents with the appropriate behavior.

References

1. Hadiashar, A.B., Gheissari, N.: Range image segmentation using surface selection criterion. IEEE Transactions on Image Processing 15(7), 2006–2018 (2006)
2. Bovenkamp, E.G.P., Dijkstra, J., Bosch, J.G., Reiber, J.H.C.: Multi-agent segmentation of IVUS images. Pattern Recognition 37(4), 647–663 (2004)
3. Ding, Y., Ping, X., Hu, M., Wang, D.: Range image segmentation based on randomized hough transform. Pattern Recognition Letters 26(13), 2033–2041 (2005)
4. Fan, T.J., Medioni, G.G., Nevatia, R.: Segmented description of 3-D surfaces. IEEE Journal of Robotics and Automation 3(6), 527–538 (1987)
5. Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1999)
6. Harrouet, F., Tisseau, J., Reignier, P., Chevaillier, P.: oRis: un environnement de simulation interactive multi-agents. Technique et Science Informatiques 21(4), 499–524 (2002)
7. Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P.J., Bunke, H., Goldgof, D.B., Bowyer, K.W., Eggert, D.W., Fitzgibbon, A.W., Fisher, R.B.: An experimental comparison of range image segmentation algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 673–689 (1996)

8. Inokuchi, S., Nita, T., Matsuda, F., Sakurai, Y.: A three dimensional edge-region operator for range pictures. In: 6th International Conference on Pattern Recognition, Munich, pp. 918–920 (1982)
9. Jiang, X., Bowyer, K.W., Morioka, Y., Hiura, S., Sato, K., Inokuchi, S., Bock, M., Guerra, C., Loke, R.E., Hans du Buf, J.M.: Some further results of experimental comparison of range image segmentation algorithms. In: 15th International Conference on Pattern Recognition, Barcelona, Spain, vol. 4, pp. 4877–4882 (2000)
10. Jiang, X., Bunke, H.: Edge detection in range images based on Scan Line approximation. Computer Vision and Image Understanding 73(2), 183–199 (1999)
11. Kang, S.B., Ikeuchi, K.: The complex EGI: A new representation for 3-D pose determination. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(7), 707–721 (1993)
12. Krishnamurthy, E.V., Murthy, V.K.: Distributed agent paradigm for soft and hard computation. Journal of Network and Computer Applications 29(2), 124–146 (2006)
13. Li, S., Zhao, D.: Gradient-based polyhedral segmentation for range images. Pattern Recognition Letters 24(12), 2069–2077 (2003)
14. Li, S.Z.: Roof-edge preserving image smoothing based on MRFs. IEEE Transactions on Image Processing 9(6), 1134–1138 (2000)
15. Li, S.Z.: Markov random field modeling in image analysis. Springer, New York, Inc., Secaucus, NJ, USA (2001)
16. Liu, J., Tang, Y.Y.: Adaptive image segmentation with distributed behavior-based agents. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(6), 544–551 (1999)
17. Newell, A.: Unified theories of cognition. Harvard University Press, Cambridge, MA, USA (1990)
18. Richard, N., Dojat, M., Garbay, C.: Automated segmentation of human brain MR images using a multi-agent approach. Artificial Intelligence in Medicine 30(2), 153–176 (2004)
19. Rodin, V., Benzinou, A., Guillaud, A., Ballet, P., Harrouet, F., Tisseau, J., Le Bihan, J.: An immune oriented multi-agent system for biological image processing. Pattern Recognition 37(4), 631–645 (2004)
20. Simonin, O.: Construction of numerical potential fields with reactive agents. In: AAMAS 2005: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, pp. 1351–1352. ACM Press, New York (2005)
21. Tsuji, T., Tanaka, Y., Morasso, P., Sanguineti, V., Kaneko, M.: Bio-mimetic trajectory generation of robots via artificial potential field with time base generator. IEEE Transactions on Systems, Man, and Cybernetics, Part C 32(4), 426–439 (2002)
22. Yang, H.S., Kak, A.C.: Determination of the identity, position and orientation of the topmost object in a pile. Computer Vision, Graphics, and Image Processing 36(2-3), 229–255 (1986)