



**HAL**  
open science

# Une approche multi-agent pour la segmentation d'images de profondeur

Smaine Mazouzi, Zahia Guessoum, Fabien Michel

## ► To cite this version:

Smaine Mazouzi, Zahia Guessoum, Fabien Michel. Une approche multi-agent pour la segmentation d'images de profondeur. Reconnaissances des Formes et Intelligence Artificielle, Jan 2008, Amiens, France. pp.121-133. <hal-01302992>

**HAL Id: hal-01302992**

**<https://hal.science/hal-01302992v1>**

Submitted on 12 Dec 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Une approche multi-agent pour la segmentation d'images de profondeur

## A Multi-agent Approach for Range Image Segmentation

S. Mazouzi<sup>1</sup>

Z. Guessoum<sup>2</sup>

F. Michel<sup>1</sup>

<sup>1</sup>Equipe MODECO-CReSTIC, Université de Reims, Reims, France

<sup>2</sup>LIP6, Université de Paris 6, Paris, France

Laboratoire LERI, IUT de Reims, B.P. 1035, 51687, Reims, France

mazouzi@leri.univ-reims.fr

### Résumé

*Dans cet article, nous présentons et nous évaluons une approche multi-agent pour la segmentation d'images de profondeur. L'approche consiste en l'utilisation d'une population d'agents autonomes pour la segmentation d'une image de profondeur en ses différentes régions planes. Les agents s'adaptent aux régions sur lesquelles ils se déplacent, puis effectuent des actions coopératives et compétitives produisant une segmentation collective de l'image. Un champ de potentiel artificiel est introduit afin de coordonner les mouvements des agents et de leur permettre de s'organiser autour des pixels d'intérêt. Les résultats expérimentaux obtenus par des images réelles montrent le potentiel de l'approche proposée pour l'analyse des images de profondeurs, et ce vis-à-vis de l'efficacité de segmentation et de la fiabilité des résultats.*

### Mots Clef

Segmentation d'images, Multi-agents, Coordination d'agents, Champ de potentiel, Image de profondeur.

### Abstract

*In this paper, we present and evaluate a multi-agent approach for range image segmentation. The approach consists in using a population of autonomous agents to segment a range image in its planar regions. The agents adapt to the regions on which they move, then perform cooperative and competitive actions allowing a collective segmentation of the image. An artificial potential field is introduced in order to coordinate agent movements, and allow them to be gathered around pixels of interest. Experimental results obtained with real images show the potential of the proposed approach for range image analysis, regarding both segmentation reliability and computation efficiency.*

### Keywords

Image segmentation, Multi-agent, Agent coordination, Potential field, Range Image.

## 1 Introduction

La segmentation d'une image est souvent nécessaire pour fournir une description compacte et opportune de son contenu. Elle consiste en l'affectation des pixels à des sous-ensembles homogènes et disjoints, formant une partition de l'image. Les pixels qui appartiennent à une même région partagent une propriété commune, dite critère d'homogénéité de région. A l'instar des méthodes générales de segmentation d'images, les méthodes de segmentation en imagerie de profondeur se divisent en deux catégories distinctes : les méthodes de segmentation en contours, et les méthodes de segmentation en régions. Pour la première catégorie, les pixels qui correspondent aux discontinuités de profondeur (contour en marche) ou d'orientation de surface (contour en toit) sont sélectionnés et chaînés dans le but de délimiter les régions de l'image [4, 9]. Les méthodes orientées contour sont bien connues pour leur coût de calcul faible ; cependant elles sont très sensibles au bruit et nécessitent des post-traitements complexes pour le chaînage et la fermeture de contours. Les méthodes orientées région utilisent les propriétés de surface pour grouper les pixels ayant les mêmes propriétés, dans des parties connexes et disjointes [10, 3, 1]. Comparées aux méthodes orientées contour, les méthodes orientées région sont plus stables et moins sensibles au bruit. Cependant, leur efficacité dépend fortement de la sélection des graines initiales de régions. De plus, cette approche ne facilite pas la parallélisation et la distribution des traitements. Ceci rend le coût de calcul très élevé et ne permet pas à ces méthodes d'être utilisées pour des applications temps réel.

Dans les deux approches, il est souvent nécessaire d'effectuer des tâches de prétraitements, qui consistent principalement en un lissage de l'image ou un filtrage de bruit. Cependant, dans le cas d'images hautement bruitées, telles que les images de profondeur [7], un fort lissage peut conduire à l'effacement des contours, notamment les contours en toit et les contours lisses. Ces derniers correspondent à la discontinuité de la courbure de surface, et leur

détection demeure un déficit en segmentation d'images [9]. Cependant, si l'image est sous-lissée, les distorsions qui restent engendrent des résultats inexacts ou erronés. Ce problème est dû à la restriction du calcul et de la décision au voisinage local du pixel traité. En imagerie de profondeur, plusieurs méthodes récentes échouent parce qu'elles n'adressent et ne résolvent pas correctement ce problème [8, 1].

Pour remédier à cette difficulté, certains auteurs ont proposé des systèmes multi-agents pour la segmentation d'images. Dans ces systèmes, un agent, représentant une entité logicielle autonome, possède des capacités de perception et d'action limitées, et n'est pas conçu pour résoudre un problème dans sa totalité. Les agents coopèrent donc pour fournir une solution collective. Contrairement aux systèmes conventionnels, les solutions dans les systèmes multi-agents, émergent des actions collectives effectuées au sein de la population des agents [11]. Cependant, la plupart des systèmes multi-agents qui ont été proposés pour la segmentation d'images procèdent selon une approche supervisée et sont spécifiques aux contenus des images traitées. Ils sont dédiés en majorité aux images à niveau de gris, avec le nombre et les propriétés des régions préalablement connus.

Dans cet article, une approche non supervisée de segmentation d'images de profondeur, basée sur le paradigme agent est présentée et discutée. Elle consiste en l'utilisation d'une population d'agents réactifs qui se déplacent sur l'image et agissent sur les pixels situés sur les pourtours des régions. Les agents s'adaptent aux régions planes de l'image, puis lissent les pixels appartenant à ces régions. Cependant, ils se trouvent en compétition sur les bordures entre les régions. L'alignement alternatif des pixels des bordures qui résulte des actions des groupes compétitifs d'agents préserve ses bordures contre l'effacement. Les régions de bruit qui sont caractérisées par de faibles tailles ou par des profondeurs aberrantes ou aléatoires, ne permettent pas aux agents de s'adapter. Ces régions se contractent continuellement, et ce par l'alignement de leurs pixels aux régions planes qui les entourent.

L'objectif du travail présenté dans cet article est de surmonter la difficulté due à la restriction de la perception au seul voisinage immédiat du pixel traité. En effet, selon notre approche, un pixel est traité non uniquement en fonction de son voisinage, mais aussi en fonction des états des agents qui visitent ce pixel. La mémoire d'un agent représente une perception plus large qui, lorsqu'elle est combinée avec l'information locale de l'image, permet une décision plus fiable, en prenant en compte des données globales. Dans le but d'optimiser les déplacements des agents, un champ de potentiel inspiré du champ électrostatique est utilisé. Il permet aux agents de s'auto-organiser et de rationaliser leurs mouvements en se regroupant autour des régions d'intérêt (contours et bruit) et à concentrer leurs actions autour de ces régions.

Nous montrons dans ce travail que, malgré la simplicité du

modèle utilisé pour la représentation des données image, les résultats obtenus sont meilleurs que ceux obtenus selon des approches conventionnelles. Nous croyons que l'interaction entre les agents fournit une solution alternative pour la segmentation d'images, par rapport aux méthodes de segmentation basées sur des modèles complexes et coûteux en temps de calcul [12].

L'utilisation d'agents réactifs et faiblement couplés permet de plus à notre approche d'être parallélisable et ainsi d'être utilisée dans des applications temps réel. Des expérimentations intensives ont été effectuées en utilisant des images réelles provenant de la base de données standard ABW [7]. Les résultats expérimentaux obtenus montrent le potentiel de l'approche proposée pour une segmentation efficace et exacte des images de profondeur.

La suite du papier est organisée comme suit : dans la section 2, nous passons en revue quelques travaux ayant proposé des systèmes multi-agents pour la segmentation d'images. La section 3 présente la méthode de modélisation des propriétés des surfaces. La section 4 est réservée à l'approche proposée. Elle décrit le comportement des agents, et montre le mécanisme collectif sous-jacent permettant la détection de contours et l'élimination de bruit. Les résultats expérimentaux sont présentés à la section 5, dans laquelle nous discutons la sélection des paramètres, et où nous présentons et commentons les résultats obtenus. Finalement, une conclusion résume notre contribution.

## 2 Etat de l'art

Plusieurs systèmes multi-agents ont été proposés dans le domaine du traitement d'images et de la reconnaissance des formes. Ces systèmes introduisent des solutions intéressantes pour remédier à différents problèmes tels que la manipulation de connaissances hétérogènes, le contrôle de haut niveau sur les traitements de bas niveau, et la parallélisation et la distribution des traitements. Dans cet état de l'art, nous ne considérons que les travaux qui ont adressé une solution basée agent pour la segmentation d'images.

Liu et al. [13] ont présenté un système basé sur des agents réactifs pour la segmentation d'IRM du cerveau. Quatre types d'agents sont utilisés pour étiqueter les pixels de l'image en fonction de leur degré d'appartenance aux différentes régions. Les agents qui réussissent à trouver des pixels d'une région homogène spécifique, créent des agents clones à l'intérieur de leurs régions voisines. Les auteurs affirment que leur système est plus robuste et plus efficace que les algorithmes classiques d'éclatement et de fusion de régions. Cependant, dans ce système les agents n'interagissent pas directement entre eux, et n'agissent pas sur l'image. Leurs actions dépendent uniquement de leur perception locale de l'image. Néanmoins, ils sont créés et placés de telle sorte qu'ils soient susceptibles de trouver plus de pixels de la région pour laquelle ils ont été créés.

Pour le même type d'image, Richard et al. [15] ont proposé une architecture hiérarchique d'agents situés et coopératifs pour la segmentation d'images. Trois types d'agents ont

été utilisés : agent de contrôle global, agent de contrôle local, et agent dédié au tissu. Le rôle de l'agent de contrôle global est de partitionner le volume de données en territoires adjacents et d'affecter à chaque territoire un agent de contrôle local. Le rôle de ce dernier est de créer les agents dédiés au tissu dont le rôle est d'effectuer un accroissement de région à l'intérieur du volume local. Les paramètres de distribution des données sont mis à jour par coopération entre les agents voisins. Par l'utilisation de différents types d'agents, le système proposé a pris en compte les traitements d'images de bas niveau, ainsi que le contrôle sur les tâches de segmentation. Cependant, il résulte des patterns complexes d'interaction dont la gestion est épineuse et coûteuse.

Les deux systèmes précédents ont été bien optimisés pour la segmentation des IRM du cerveau. Ils peuvent produire de bons résultats, car les caractéristiques de régions dans ce type d'image sont régulières du fait qu'elles correspondent aux différentes structures anatomiques du cerveau. En plus, la plupart des contours dans de telles images sont de type marche. Ces contours sont faciles à détecter, comparés aux contours de toit et contours lisses.

En utilisant le langage multi-agents oRis [6], Rodin et al. [16] ont proposé un système multi-agents formé d'agents réactifs pour la détection de contours dans des images biologiques. En fonction de certains a priori sur le contenu des images, le système vise à produire une détection de contours meilleure que celle obtenue par les détecteurs traditionnels. Deux groupes d'agents dits respectivement agents de noircissement et agents de luminance suivent les régions sombres et les régions claires. Leurs actions visent à renforcer les régions par l'accroissement de leur contraste, permettant ainsi une détection fiable de ces régions. Selon cette approche, les agents parcourant les différents contours sont totalement indépendants sans aucune d'interaction entre eux. Le système paraît comme un algorithme parallèle de segmentation qui était bien optimisé pour la détection des contours de type toit dans certains types d'image. Cependant, il peut échouer à détecter des contours plus simples tels que les contours de type marche. De plus, le nombre et la topologie des régions attendues doivent aussi être préalablement connus et codés dans les agents.

En se basant sur l'architecture cognitive Soar [14], Bovenkamp et al. [2] ont développé un système multi-agents pour la segmentation des images intravasculaire ultrasonores (IVUS). Leur objectif est d'établir un sous-système basé connaissances, sur les algorithmes de traitement de bas niveau. Dans leur système un seul agent est affecté à chaque objet prévu dans l'image. Les agents coopèrent et adaptent dynamiquement les algorithmes de segmentation, en se basant sur la connaissance contextuelle, l'information locale, et les croyances personnelles. Dans ce travail, le problème du contrôle des algorithmes de traitement d'image était bien posé et résolu. Cependant aucun agent ni comportement n'ont été proposés pour remédier au

problème d'ambiguïté des données dans l'image. Le couplage fort entre les traitements de haut niveau (spécifiques au domaine) et les traitements de bas niveau (spécifiques à l'image), adopté dans ce système, ne permet pas la réutilisation des méthodes proposées pour d'autres applications ou d'autres types d'images.

Nous pouvons constater que la plupart des systèmes basés agent, proposés pour la segmentation d'images, suivent une approche supervisée, et sont spécifiques au contenu des images traitées. Ces systèmes visent à segmenter les images en régions connues correspondant aux différents objets préalablement prévus. L'approche que nous proposons dans ce papier, se veut générale et non supervisée. Elle vise à segmenter une image en ses différentes régions en se basant sur certaines propriétés géométriques des régions. Le comportement à la fois adaptatif et compétitif des agents contribue à surmonter la contrainte du voisinage local du pixel inhérente au traitement d'images.

### 3 Modélisation et calcul des propriétés locales de surface

Une image de profondeur est une image 3-D où à chaque pixel  $(x, y)$  correspond la distance  $Z(x, y)$  entre le capteur télémétrique et le point correspondant de la scène. Dans une telle image, les régions représentent les parties visibles des surfaces des différents objets figurant dans la scène. Dans le but d'atténuer le bruit Gaussien ainsi que le bruit impulsif, un filtre Gaussien et un filtre médiane sont appliqués aux données brutes de profondeur. Soit  $Z^*(x, y)$  les paramètres de l'équation du plan tangent à la surface en  $(x, y)$ . L'équation du plan tangent en  $(x, y)$  est obtenue par la méthode de régression multiple en utilisant les pixels appartenant au voisinage du pixel en question. Le voisinage du pixel  $(x, y)$  est constitué des pixels  $(x', y')$  situés dans une fenêtre de taille  $3 \times 3$  centrée en  $(x, y)$ , et dont les profondeurs  $Z(x', y')$  sont, à un seuil donné de profondeur ( $Tr_D$ ), proches. L'équation d'un plan dans un référentiel cartésien 3-D peut être exprimée comme suit :

$$ax + by + cz = d \quad (1)$$

où  $(a, b, c)^T$  est le vecteur unitaire normal au plan ( $a^2 + b^2 + c^2 = 1; c < 0$ ) et  $d$  est la distance orthogonale du plan à l'origine du référentiel.

Les traitements effectués sur l'image se basent sur la comparaison de plans. En effet, deux plans d'équations respectives :  $ax + by + cz = d$  et  $a'x + b'y + c'z = d'$  sont considérés égaux s'ils ont à des seuils près, la même orientation et la même distance à l'origine du référentiel. Soit  $\theta$  l'angle entre les deux vecteurs normaux, et  $D$  la distance entre les deux plans :  $\sin(\theta) = \|(a, b, c)^T \otimes (a', b', c')^T\|$  et  $D = |d - d'|$ . Les deux plans sont donc considérés égaux si  $\sin(\theta) \leq Tr_\theta$  et  $D \leq Tr_D$ , où  $Tr_\theta$  et  $Tr_D$  sont les seuils respectivement d'angle et de profondeur, obtenus par apprentissage (voir sous-section 5.2). Le test d'égalité de

deux plans est utilisé d'abord pour tester si le pixel à la position  $(x, y)$  appartient à une région plane, étant donnée son équation de plan. Il est également utilisé pour tester si le pixel en  $(x, y)$  est un pixel d'intérêt (de contour ou de bruit) ou non. Dans ce cas, le pixel en question est considéré comme un pixel d'intérêt si au moins un de ses voisins a une équation de plan différente, en se référant aux seuils précédents.

## 4 Segmentation collective d'image par agents réactifs

### 4.1 Comportement des agents

L'image est considérée comme un environnement dans lequel des agents sont initialisés à des positions arbitraires. Les agents utilisés ont une granularité faible, et leur nombre doit être suffisamment grand. Un agent cherche une région plane autour de sa position courante et s'adapte à cette région en mémorisant son équation de plan. Ensuite, il effectue des actions qui dépendent à la fois de son état et de l'état du pixel sur lequel il se situe. A chaque instant  $t$ , un agent est caractérisé par sa position  $(x_t, y_t)$  dans l'image, et par sa capacité  $C_t$  d'agir sur les pixels rencontrés. Au début du processus, tous les agents sont incapables d'altérer les pixels de l'image. Un agent devient capable de modifier un pixels ( $C_t=true$ ) lorsque il détecte une région plane autour de sa position courante. Si un agent vient d'altérer un pixel, il perd sa capacité d'altération ( $C_t=false$ ) et recommence à chercher une nouvelle région plane. Chaque agent ayant modifié un pixel, enregistre dans une structure appropriée  $I$ , dite matrice d'états des pixels, à la position  $(x_t, y_t)$  le dernier état du pixel visité :  $I(x_t, y_t) \in \{\text{lissé, aligné, non changé}\}$ .

Un agent s'adapte à la région de l'image dans laquelle il se déplace en mémorisant les caractéristiques de cette région, et en adoptant le comportement adéquat en fonction des données de l'image. Nous montrons dans la suite de la section, que le comportement simple et réactif des agents permet de faire émerger les lignes de contour et d'éliminer efficacement le bruit qui existe dans l'image. La figure 1 schématise le comportement d'un agent en fonction de son état courant et du pixel sur lequel il est situé.

**Recherche d'une région plane.** Après sa création, un agent se déplace aléatoirement dans l'image et cherche une région plane autour de sa position courante. L'agent utilise les  $L$  derniers pixels parcourus pour tester s'il est à l'intérieur d'une région plane ou non.  $L$  est appelée la longueur d'adaptation, et représente le degré de confiance que l'agent soit à l'intérieur d'une région plane. L'agent considère qu'il est à l'intérieur d'une région plane si les derniers  $L$  pixels parcourus appartiennent tous au même plan. L'agent mémorise la région trouvée et la considère désormais comme sa région plane courante. Après cela, il devient capable d'altérer le premier pixel rencontré qui n'appartient pas à sa région plane ( $C_t=true$ ).

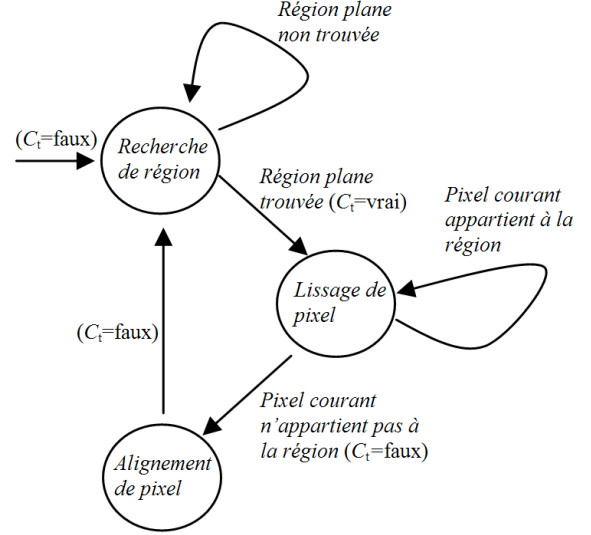


FIG. 1 – Le comportement d'un agent en fonction de son état et de sa position

**Lissage des pixels à l'intérieur d'une régions.** Lors de son déplacement dans une région plane, l'agent lisse l'image au niveau du pixel sur lequel il se situe, en mettant à jour les équations respectives du plan de la région mémorisée, et du plan à la position du pixel. Ceci est fait en remplaçant les deux équations par leur moyenne pondérée. Soit  $(a, b, c, d)$  et  $(a', b', c', d')$  les paramètres respectivement du plan de la région mémorisée, et du plan au pixel courant, les paramètres résultats  $(a'', b'', c'', d'')$  du plan moyen pondéré sont obtenus comme suit :

$$(a'', b'', c'', d'') = \frac{1}{1+l} (a' + la, b' + lb, c' + lc, d' + ld) \quad (2)$$

où  $l$  est la longueur, en pixels, du chemin parcouru par l'agent dans la région plane.

**Alignement des pixels d'intérêt.** Les pixels d'intérêt sont les pixels de contours ou les pixels à l'intérieur des régions de bruit. Un pixel d'intérêt est reconnu en testant l'égalité du plan au niveau de ce pixel, et celui de la région mémorisée au sein de l'agent (voir section 3). Quand un agent rencontre un pixel d'intérêt (qui n'appartient pas à sa région plane courante), il l'altère pour l'aligner partiellement à la région sur laquelle il se déplace. Les paramètres  $(a'', b'', c'', d'')$  de la nouvelle équation de plan à la position du pixel sont obtenus par la combinaison linéaire des anciens paramètres  $(a, b, c, d)$  et des paramètres de l'équation de plan de la région mémorisée  $(a', b', c', d')$  :

$$(a'', b'', c'', d'') = \frac{1}{1+\xi} (a + \xi a', b + \xi b', c + \xi c', d + \xi d') \quad (3)$$

où  $\xi$  est la force d'altération.

Après cela, l'agent perd sa capacité d'altération ( $C_t=false$ ) et recommence à chercher une nouvelle région plane. Suite à l'altération d'un pixel, l'agent peut passer dans une autre région ou rester dans la région précédente. Si le pixel altéré est un pixel de contour, il est plus probable que l'agent passe dans une autre région plane. Par contre, si le pixel altéré appartient à la bordure d'une région de bruit, l'agent traverse cette région et plus probablement retourne à la région plane précédente, sauf si la région de bruit est située entre deux régions planes différentes. La force d'altération  $\xi$  est un paramètre critique qui influe sur la qualité des résultats et le temps de calcul. En effet, le choix de valeurs élevées pour  $\xi$  conduit à une détection rapide de régions. Cependant, les bordures des régions qui en résultent sont déformées. Le choix de valeurs basses de  $\xi$  conduit à une détection lente. Néanmoins, les bordures des régions, dans ce cas, sont bien détectées et correctement localisées. Pour accélérer le processus de segmentation sans déformation des contours, un agent choisit la force d'altération parmi  $\xi_{min}$  et  $\xi_{max}$  en fonction de l'information enregistrée dans la matrice des états de pixels  $I$ . En effet, l'agent suppose que la région courante est adjacente à une région de bruit, et alors utilise  $\xi_{max}$  comme force d'altération, s'il existe au moins 3 pixels  $(x', y')$  étiquetés "non changé" ( $I(x', y')="non\ changé"$ ) autour de l'agent. Sinon, l'agent suppose que la région plane courante est adjacente à une autre région plane, où des agents ont étiqueté les pixels comme "lissés" ou "alignés". Dans ce cas, l'agent utilise la force d'altération  $\xi_{min}$ . Les constantes  $\xi_{min}$  et  $\xi_{max}$  sont deux paramètres du système dont les valeurs sont déterminées par apprentissage (voir sous section 5.2).

## 4.2 Coordination d'agents par un champ de potentiel

Afin de doter les agents d'un mécanisme d'auto-organisation spatiale, un champ de potentiel artificiel inspiré du champ électrostatique est créé et mis à jour autour des pixels alignés. Il permet aux agents de se regrouper autour des régions d'intérêt dans l'image et de concentrer leurs actions sur les contours de ces régions. Contrairement à d'autres travaux où le champ de potentiel est créé à des positions prédéfinies correspondant aux objets de l'environnement (buts et obstacles) [5, 18, 17], dans notre cas, le champ de potentiel résulte de l'interaction des agents avec les objets de l'environnement (pixels de l'images). L'intensité  $\Psi(x, y)$  du champ à la position  $(x, y)$ , créé par un ensemble de  $P$  pixels préalablement alignés  $\{(x_i, y_i), i = 1..P \wedge I(x_i, y_i)=aligné\}$  est donnée par :

$$\Psi(x, y) = \sum_{i=1}^P \frac{k}{\sqrt{(x-x_i)^2 + (y-y_i)^2}}, k \in R^+ \quad (4)$$

où  $k$  est la constante de la force électrostatique.

Un agent, capable d'altérer des pixels ( $C_t=true$ ), et situé à la position  $(x, y)$ , subit une force d'attraction  $\vec{F}$ . Cette

force s'exprime à l'aide du vecteur gradient du champ de potentiel :

$$\vec{F} = \begin{cases} -\vec{\nabla}\Psi(x, y) & \text{si } C_t=true \\ \vec{0} & \text{sinon} \end{cases}$$

Ainsi, les déplacements de l'agent qui sont de nature stochastique sont pondérés par la force attractive exercée par le champ de potentiel. Les agents sont ainsi poussés à se diriger vers les pixels d'intérêt, tout en maintenant un déplacement stochastique qui permet d'explorer toutes les régions de l'image.

Un mécanisme de relaxation du champ de potentiel est introduit. Il permet aux agents groupés autour des pixels d'intérêt de se libérer et d'explorer d'autres régions dans l'image. Autour d'un pixel donné, l'intensité du champ décroît après chaque altération de ce pixel. L'équation de la dynamique de relaxation est exprimée comme suit :

$$\Psi_{t+1}(x, y) = \mu \times \Psi_t(x, y), \mu < 1 \quad (5)$$

$\Psi_0(x, y)$  correspond au champ créé après la première altération du pixel. La constante  $\mu$  initialisée à 0.9, représente le taux de décroissance de l'intensité du champ. Après plusieurs altérations d'un pixel donné, le champ généré autour de ce pixel tend à s'annuler. A la limite de la dynamique, le champ de potentiel est complètement relaxé dans toute l'image. Ceci représente l'état final du processus pour lequel l'intérieur des régions est totalement lissé et les contours sont précisément détectés.

## 4.3 Détection de contours

Lors de son déplacement sur l'image, un agent lisse les pixels qui appartiennent approximativement à la région plane sur laquelle il se situe actuellement. Un agent considère les pixels n'appartenant pas à cette région comme des pixels de bruit. Ces derniers sont donc automatiquement alignés à la région plane qui les entoure. Cependant, les pixels des frontières entre les régions planes sont de vrais pixels de contours et ne devraient donc pas être alignés. Néanmoins, sur le segment de contour entre deux régions adjacentes, deux groupes concurrents d'agents se forment. Chaque groupe est formé d'agents qui transitent d'une région à une autre. Les agents de chaque groupe alignent les pixels de la frontière à leur région plane courante. Ainsi, ces pixels se retrouvent continuellement permutés entre les deux régions adjacentes, et demeurent ainsi distingués dans l'image. Cette compétition entre les agents permet également l'amincissement des contours de type toit et des contours lissés qui existent entre les régions adjacentes d'un même objet. Au début, un tel contour est gras, formé d'une bande de pixels qui n'appartient à aucune des deux régions adjacentes (Fig. 2a). Durant le processus, les pixels sur chaque coté de la bande sont continuellement alignés à la région dont ils forment la bordure. Les agents qui

passent à l'intérieur de cette bande ne peuvent pas s'adapter et quittent la bande sans altérer ses bordures. La bande continue donc à se rétrécir jusqu'au point où il ne reste que les pixels qui sont en son centre. Ces pixels représentent la ligne de compétition entre les deux groupes d'agents, qui ne sera alignée à aucune des deux régions (Fig. 2b). Au bout d'un certain temps, il ne reste distingué dans l'image que les pixels de bordure entre les régions adjacentes qui continuent à se permuter entre les régions. Ce schéma d'action compétitif entre les agents permet de faire émerger les lignes de contours. Ce résultat qui n'est codé dans aucun agent, émerge de l'action collective de tous les agents sur l'image.

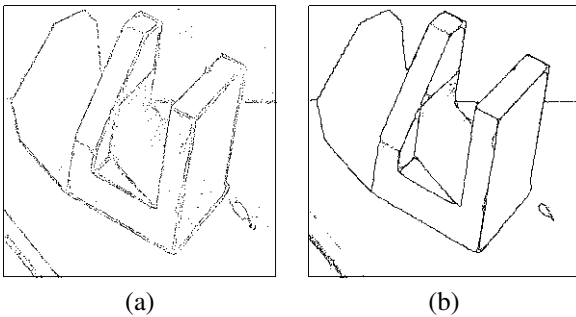


FIG. 2 – Amincissement de contours (image abw.test.22) : (a) Pixels de contour à  $t=800$  ; (b) Pixels de contour à  $t=8000$

#### 4.4 Elimination du bruit

Contrairement aux vraies régions de l'image, qui restent préservées contre l'effacement, les régions de bruit se contractent continuellement, et se finissent par disparaître. Les bordures de ces dernières sont continuellement alignées aux vraies régions planes environnantes. Chaque agent ayant aligné un pixel appartenant à la bordure d'une région de bruit, et s'est déplacé à l'intérieur de cette région, ne sera pas capable de s'adapter. Par conséquent, il ne peut aligner aucun pixel en quittant cette région. Ceci se produit dans deux cas différents : 1) lorsque la région de bruit est suffisamment large mais non plane, ou formée de pixels ayant des profondeurs aléatoires ; 2) lorsque la région est plane mais insuffisamment large pour permettre aux agents de traverser la longueur minimale  $L$  nécessaire pour pouvoir s'adapter. Dans les deux cas, l'agent quitte la région de bruit et s'adapte dans la région plane qui entoure cette dernière. A l'intérieur d'une vraie région de l'image, un agent traverse une longueur suffisante lui permettant de s'adapter à cette région. Cependant, un agent ne peut pas s'adapter à l'intérieur d'une région dont la surface n'est pas plane, et ce, quelle que soit la taille de cette région. Il ne peut également pas s'adapter à l'intérieur d'une région plane si la taille de cette dernière ne lui permet pas de traverser une longueur supérieure ou égale à la longueur minimale  $L$ . Dans les deux derniers cas, les agents ne peuvent pas s'adapter, et par conséquent

ne seront pas capables d'aligner les pixels de bordure en quittant ces régions. Autrement dit, les vraies régions ont des tailles suffisamment grandes qui permettent aux agents qui sont à l'intérieur de s'adapter et d'aligner les pixels de bordure en quittant ces régions. Par contre, les régions de bruit, qui sont des régions non planes ou ayant de faibles tailles ne permettent pas aux agents de s'adapter. Par conséquent les agents n'alignent pas les pixels de bordure de ces régions en les quittant. Dans ce cas, les bordures de ces régions sont continuellement alignées de l'extérieur par inclusion de leurs pixels dans les vraies régions environnantes. Après plusieurs pas d'exécution, ces régions seront complètement effacées.

## 5 Expérimentation et analyse

### 5.1 Le framework d'évaluation

Un framework d'évaluation et de comparaison, dédié aux méthodes de segmentation d'images de profondeur, a été proposé par Hoover et al. dans [7] et est largement utilisé par les auteurs d'algorithmes [9, 8, 3, 1]. Le framework consiste en une base d'images de profondeur, et un ensemble de métriques objectives de performance. Il permet de comparer une segmentation générée par une machine de segmentation (MS) à une segmentation générée manuellement, supposée idéale, et représentant la réalité terrain (GT). Les métriques de performance les plus importantes sont le nombre de régions correctement détectées, de régions sur-segmentées, de régions sous-segmentées, de régions omises et de régions de bruit. La classification de régions est effectuée pour un ensemble de valeurs d'un seuil de tolérance  $T$  ;  $50\% < T \leq 100\%$ , qui reflète la rigueur de la classification. Les 40 images de la base ABW, de tailles de  $512 \times 512$  pixels, ont été divisées en deux sous-ensembles : un sous-ensemble de 10 images d'apprentissage, et un sous-ensemble de 30 images de test [7]. Les images d'apprentissage sont utilisées pour estimer les paramètres d'une méthode de segmentation donnée. Suite à cela, la méthode est appliquée à toutes les images de test. Les différentes valeurs des métriques de performances sont calculées et stockées pour être utilisées pour la comparaison des différentes méthodes. Nous avons utilisé ce framework pour évaluer notre approche et comparer nos résultats aux quatre méthodes : USF, WSU, UB et UE, référencées dans [7]. Avant de présenter les résultats expérimentaux, nous montrons la phase d'apprentissage qui permet la sélection des valeurs optimales des différents paramètres.

### 5.2 Sélection de paramètres

Pour le système proposé, nommée SIBA pour Segmentation d'Image à Base d'Agent, six paramètres doivent être fixés :  $\xi_{min}$ ,  $\xi_{max}$ ,  $Tr_{\theta}$ ,  $Tr_D$ ,  $N$ , et  $L$ . Cet ensemble de paramètres est divisé en deux sous-ensembles.  $\xi_{min}$ ,  $\xi_{max}$ ,  $Tr_{\theta}$ , et  $Tr_D$  représentent respectivement les forces d'alignement et les seuils d'angle et de profondeur, et sont utilisés dans les opérations de test et d'ali-

gnement des pixels de l'image. Les paramètres  $N$  et  $L$  représentent respectivement le nombre d'agents, et la longueur d'adaptation. Ces deux paramètres influencent la dynamique des agents. Pour le premier sous-ensemble de paramètres, nous considérons 256 combinaisons à savoir  $(\xi_{min}, \xi_{max}, Tr_{\theta} \text{ et } Tr_D) \in \{0.5, 0.3, 0.1, 0.05\} \times \{1.0, 3.0, 5.0, 7.0\} \times \{15, 18, 21, 24\} \times \{12, 16, 20, 24\}$ , à exécuter sur les images d'apprentissage. Le critère de performance choisi pour ces quatre paramètres est le nombre moyen de régions correctement détectées dans les images d'apprentissage, avec le seuil de tolérance  $T=80\%$ . Les forces d'alignement  $\xi_{min}$  et  $\xi_{max}$  ont été initialisées respectivement à 0.3 et 5.0. Ces valeurs assurent une bonne localisation des pixels de contour en un temps d'exécution raisonnable. Le seuil  $Tr_{\theta}$  est initialisé à  $21^\circ$ . Nous avons constaté que des valeurs plus grandes de ce paramètre sous différencient les pixels selon leurs vecteurs normaux, et conduisent à une sous-segmentation. Cependant, des valeurs significativement inférieures à  $21^\circ$  sur différencient les pixels et conduisent à une sur-segmentation de l'image. Ceci se traduit par la génération d'un nombre élevé de fausses petites régions. Finalement, le seuil  $Tr_D$  est initialisé à 16. Les valeurs significativement supérieures à 16 peuvent conduire à une fausse fusion de régions parallèles et proches. Cependant, si la valeur de ce paramètre est significativement inférieure à 16, les régions hautement inclinées ne peuvent pas être détectées comme régions planes [9]. Ceci conduit à un taux élevé de régions omises.

Le nombre d'agents utilisés  $N$ , et la longueur d'adaptation  $L$  sont critiques et doivent être minutieusement sélectionnés. Un mauvais choix de ces deux paramètres peut conduire à un taux élevé d'erreurs de segmentation. En effet, un nombre insuffisant d'agents, pour une taille donnée des images traitées ( $512 \times 512$  pixels), conduit à un sous-lissage de l'image et à un sous-amincissement des contours. Les régions qui en résultent sont amputées d'un nombre important de pixels qui auraient dû être inclus dans ces régions. Une faible valeur de la longueur d'adaptation  $L$  conduit à la prise en compte de petites régions planes qui devraient être considérées comme régions de bruit. Par contre, une valeur élevée de  $L$  peut conduire à l'omission de certaines vraies régions planes et qui sont relativement étroites (voir section 4.4). Pour fixer les paramètres  $N$  et  $L$ , 25 combinaisons de ces paramètres, soit  $(N, L) \in \{1500, 2000, 2500, 3000, 3500\} \times \{3, 5, 7, 9, 11\}$  ont été exécutées sur les images d'apprentissage. Dans ce cas, le critère de performance choisi est le nombre moyen des régions de bruit, avec le seuil de tolérance  $T=80\%$ . Le calcul a abouti aux valeurs de  $N=2500$  et  $L=7$ .

### 5.3 Résultats expérimentaux

La figure 3 montre un exemple de l'évolution du processus de segmentation dans le temps. Le temps  $t$  représente le nombre de pas effectués par chaque agent depuis le début du processus. L'affichage d'image de profondeur par un simple algorithme de rendu réaliste (Fig. 3a), permet de

constater le niveau élevé de bruit dans les images utilisées. Les figures 3b, 3c, 3d et 3e montrent l'ensemble des pixels d'intérêt (pixels de contours ou de bruit) respectivement à  $t=1000, 5000, 9000$  et  $13000$ . Les régions sont progressivement lissées par l'alignement des pixels de bruits qui sont à l'intérieur de ces régions. Les contours entre les régions adjacentes sont progressivement amincis. A la fin, ces contours sont formés de lignes minces de 1 pixel de largeur (Fig. 3e).

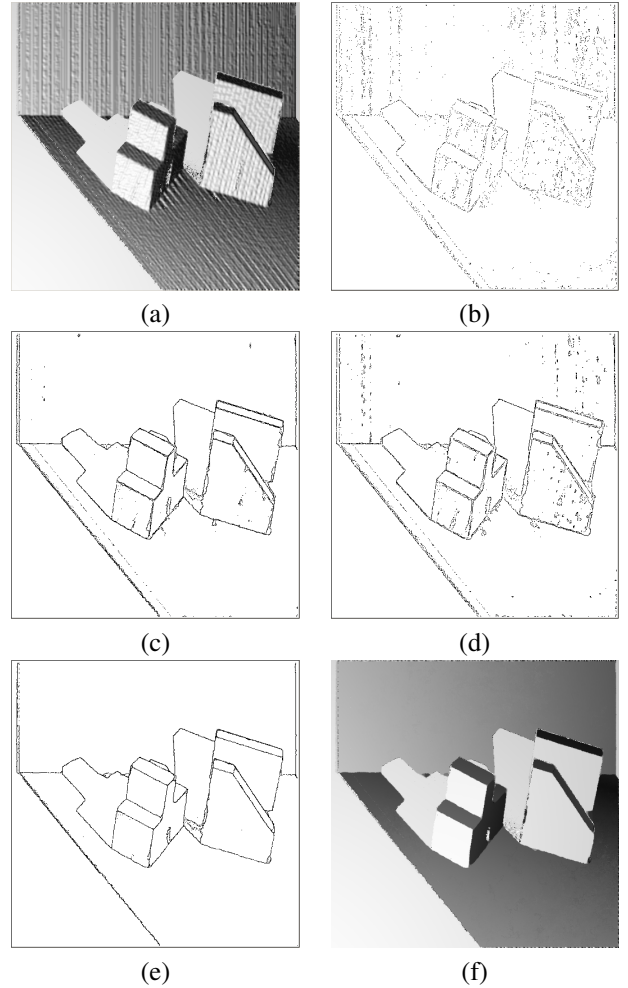


FIG. 3 – Progression de segmentation. (a) Image de profondeur rendue réaliste (abw.test.6); (b) à  $t=1000$ , (c) à  $t=5000$ ; (d) à  $t=9000$ ; (e) à  $t=13000$ ; (f) Image reconstruite rendue réaliste

La table 1 présente les résultats moyens obtenus avec toutes les images de test et pour toutes les métriques de performances. Le seuil de tolérance étant fixé à la valeur typique 80%. La figure 4 montre les nombres moyennes des nombres de régions correctement détectées pour toutes les images de test aux différentes valeurs du seuil de tolérance  $T$ ;  $T \in \{51\%, 60\%, 70\%, 80\%, 90\%, 95\%\}$ . Les résultats obtenus montrent que le nombre de régions correctement détectées par notre méthode est en moyenne meilleur que celui de USF, UB et WSU. Par exemple, notre mé-

TAB. 1 – Résultats moyens des cinq méthodes au seuil de tolérance  $T=80\%$

Mét.	Rég. GT	Dét. Cor.	Sur- seg.	Sou- seg.	Omis	Bruit
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
SIBA	15.2	13.0	0.5	0.1	1.7	0.9

thode enregistre un score meilleur que celui de WSU pour toutes les valeurs du seuil de tolérance  $T$ . Elle enregistre un score meilleur que celui de la méthode USF pour le seuil de tolérance  $T \in \{80\%, 90\%, 95\%\}$ , et meilleur que celui de UB pour  $T \in \{50\%, 60\%, 70\%, 80\%\}$ . Pour toutes les métriques de détection incorrecte, notre méthode a enregistré des scores équivalents à ceux de UE et USF. Ces dernières ont enregistré des scores meilleurs que ceux de UB et WSU.

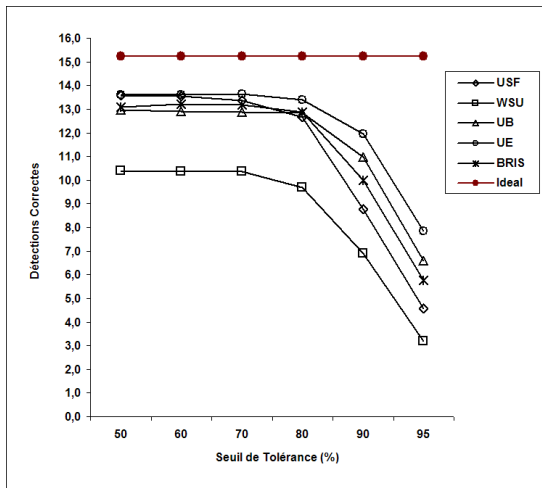


FIG. 4 – Résultats moyens des régions correctement détectées de toutes les méthodes au différents seuils de tolérance  $T$ ;  $0.5 < T \leq 1.0$

## 6 Conclusion

Dans cet article, nous avons proposé une approche multi-agents pour la segmentation d'images de profondeur. Les interactions indirectes entre les agents autonomes qui se déplacent dans l'image permettent à la fois, une élimination efficace de bruit, et une extraction fiable de régions. Les compétitions entre les agents qui s'auto-organisent autour des bordures des régions ont permis de faire émerger les lignes de contours dans l'image. Ces contours, dont aucune détection explicite n'est codée dans les agents, émergent des interactions et des actions collectives au sein de la population d'agents. Les résultats obtenus se-

lon notre approche sont meilleurs par rapport à ceux obtenus par des algorithmes conventionnels de segmentation, tel que l'accroissement de région, ou la détection explicite de contours. Les agents utilisés sont faiblement couplés et communiquent indirectement via leur environnement (l'image). Ceci permet des implémentations parallèles souhaitables pour une haute efficacité de calcul, nécessaire pour l'interprétation en temps-réel d'images, notamment en vision robotique. Les résultats d'expérimentation obtenus avec les images réelles de la base ABW ont été comparés à ceux obtenus par quatre algorithmes typiques de segmentation d'images de profondeur. Les résultats de comparaison révèlent une excellente performance de la méthode proposée et ce, à la fois, en terme d'efficacité de détection et d'exactitude des résultats. Notre approche s'apprête à être généralisée à la segmentation images contenant des surfaces courbées. Pour ce faire, il est nécessaire de modéliser les surfaces dans ces images, et d'adapter le comportement des agents en fonction de la nature de ces surfaces.

## Références

- [1] A. Bab Hadiashar and N. Gheissari. Range image segmentation using surface selection criterion. *IEEE Transactions on Image Processing*, 15(7) :2006–2018, 2006.
- [2] E. G. P. Bovenkamp, J. Dijkstra, J. G. Bosch, and J. H. C. Reiber. Multi-agent segmentation of IVUS images. *Pattern Recognition*, 37(4) :647–663, 2004.
- [3] Y. Ding, X. Ping, M. Hu, and D. Wang. Range image segmentation based on randomized hough transform. *Pattern Recognition Letters*, 26(13) :2033–2041, 2005.
- [4] T.J. Fan, G.G. Medioni, and R. Nevatia. Segmented description of 3-D surfaces. *IEEE Journal of Robotics and Automation*, 3(6) :527–538, December 1987.
- [5] Jacques Ferber. *Les systèmes multi-agents : vers une intelligence collective*. Informatique, Intelligence Artificielle. InterÉditions, 1995.
- [6] F. Harrouet, J. Tisseau, P. Reignier, and P. Chevaillier. oRis : un environnement de simulation interactive multi-agents. *Technique et Science Informatiques*, 21(4) :499–524, 2002.
- [7] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. W. Bowyer, D. W. Eggert, A. W. Fitzgibbon, and R. B. Fisher. An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7) :673–689, 1996.
- [8] X. Jiang, K. W. Bowyer, Y. Morioka, S. Hiura, K. Sato, S. Inokuchi, M. Bock, C. Guerra, R. E. Loke, and J. M. Hans du Buf. Some further results of experimental comparison of range image segmentation algorithms. In *15th International Conference on Pat-*

*tern Recognition*, volume 4, pages 4877–4882, Barcelona, Spain, 2000.

- [9] X. Jiang and H. Bunke. Edge detection in range images based on Scan Line approximation. *Computer Vision and Image Understanding*, 73(2) :183–199, 1999.
- [10] S.B. Kang and K. Ikeuchi. The complex EGI : A new representation for 3-D pose determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(7) :707–721, 1993.
- [11] E. V. Krishnamurthy and V. K. Murthy. Distributed agent paradigm for soft and hard computation. *Journal of Network and Computer Applications*, 29(2) :124–146, 2006.
- [12] S. Z. Li. *Markov random field modeling in image analysis*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2001.
- [13] J. Liu and Y. Y. Tang. Adaptive image segmentation with distributed behavior-based agents. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(6) :544–551, 1999.
- [14] A. Newell. *Unified theories of cognition*. Harvard University Press, Cambridge, MA, USA, 1990.
- [15] N. Richard, M. Dojat, and C. Garbay. Automated segmentation of human brain MR images using a multi-agent approach. *Artificial Intelligence in Medicine*, 30(2) :153–176, 2004.
- [16] V. Rodin, A. Benzinou, A. Guillaud, P. Ballet, F. Harrouet, J. Tisseau, and J. Le Bihan. An immune oriented multi-agent system for biological image processing. *Pattern Recognition*, 37(4) :631–645, 2004.
- [17] O. Simonin. Construction of numerical potential fields with reactive agents. In *AAMAS '05 : Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 1351–1352, New York, NY, USA, 2005. ACM Press.
- [18] T. Tsuji, Y. Tanaka, P. Morasso, V. Sanguineti, and M. Kaneko. 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.