Genetic Approaches for the Automatic Division of Topological Active Volumes

J. Novo, N. Barreira, M.G. Penedo, and J. Santos

Computer Science Department, University of A Coruña, Spain
{jnovo,nbarreira,mgpenedo,santos}@udc.es

Abstract. The Topological Active Volumes is an active model focused on 3D segmentation tasks. It is based on the 2D Topological Active Nets model and provides information about the surfaces and the inside of the detected objects in the scene. This paper proposes new optimization approaches based on Genetic Algorithms combined with a greedy local search that improve the results of the 3D segmentations and overcome some drawbacks of the model related to parameter tuning or noise conditions. The hybridization of the genetic algorithm with the local search allows the treatment of topological changes in the model, with the possibility of an automatic subdivision of the Topological Active Volume. This combination integrates the advantages of the global and local search procedures in the segmentation process.

Keywords: Topological Active Volumes, Genetic Algorithms, 3D segmentation.

1 Introduction and Previous Work

The active nets model was proposed by Tsumiyama and Yamamoto [1] as a variant of the deformable models [2] that integrates features of region–based and boundary–based segmentation techniques. To this end, active nets distinguish two kinds of nodes: internal nodes, related to the region–based information, and external nodes, related to the boundary–based information. The former model the inner topology of the objects whereas the latter fit the edges of the objects.

The Topological Active Net model and its extension to 3D, that is, the Topological Active Volume (TAV) model [3], were developed as an extension of the original active net model. It solves some intrinsic problems to the deformable models such as the initialization problem. The model deformation is controlled by energy functions in such a way that the mesh energy has a minimum when the model is over the objects of the scene. The TAV model is an active model focused on segmentation tasks that makes use of a volumetric distribution of nodes. It integrates information of edges and regions in the adjustment process and allows to obtain topological information inside the objects found. This way, the model, not only detects surfaces as any other active contour model, but also segments the inside of the objects. The model has a dynamic behavior by means

J. Mira et al. (Eds.): IWINAC 2009, Part II, LNCS 5602, pp. 20–29, 2009.

of topological changes in its structure, that enables accurate adjustments and the detection of several objects in the scene.

There is very little work in the optimization of active models with genetic algorithms (GA), mainly in edge or surface extraction [4][5] in 2D tasks. For instance, Ballerini [4] has developed the "genetic snakes", this is, snakes that minimize their energy by means of genetic algorithms. In [6] the authors have proved the superiority of a global search method by means of a Genetic Algorithm (GA) in the optimization of the Topological Active Nets in 2D images. The results showed that the GA is less sensitive to noise than the usual greedy optimizations and does not depend on the parameter set or the mesh size.

Regarding 3D images, Jones and Metaxas [7] have used deformable contours to estimate organ boundaries. They integrate region-based and physics-based boundary estimation methods. Starting from a single voxel within the interior of an object, they make an initial estimate of the objects boundary using fuzzy affinity, which measures the probability of two voxels belonging to the same object, together with clustering. Qiu et al. [8] have used two deformable models: a deformable surface model (SMD) and a Deformable Elastic Template (DET). The main drawback of these models, as the authors indicate, is that in both models an initial shape (surface or ellipsoid) is needed as well as it must be manually positioned in the data/image. They used both models to the analysis of the 3D shape of mouse embryo from 3D ultrasound images. The same drawback can be associated with adaptive deformable models, typically with only surface modelling, which use a reparameterization mechanism that enables the evolution of surfaces in complex geometries. McInerney and Terzopoulos [9] have used a model of this type with complex anatomic structures from medical images.

Bro-Nielsen [10] has used 3D "active cubes" to segment medical images, where the automatic net division is a key issue. Since the greedy energy-minimization algorithm proposed is sensitive to noise, an improved greedy algorithm inspired by a simulated annealing procedure is also incorporated. In [11] the author also uses the model to modelling elastic deformations in 3D solid objects. In [12] the authors propose a genetic algorithm with new defined operators for the segmentation process using TAV structures. The genetic approach overcomes some drawbacks, basically images with different types of noise, with regard to the work proposed in [3].

In this paper, Genetic Algorithms are hybridized with a greedy method, so we can join the advantages of the global and local search methods. Moreover, the hybrid combination allows topological changes in the segmentation model. The model and hybrid procedure allow to perform segmentations when the image contains several objects. In this case, the TAV net should be divided to segment them. To this end, a net reconfiguration mechanism must be developed in order to perform multiple object detection and segmentation. We tested the hybrid approach in artificial and real images, specifically with images of the medical domain.

This paper is organized as follows. Section 2 introduces the basis of the TAV model. Section 3 briefly explains the GA used in the model optimization. Section 4 details the combination with the greedy methodology to perform a link cutting procedure and an automatic net division procedure. In Section 5 representative examples are included to show the capabilities of the different approaches. Finally, Section 6 expounds the conclusions.

2 Brief Description of Topological Active Volumes

The Topological Active Volumes (TAV) model is an active contour model focused on extraction and modelization of volumetric objects in a 3D scene [3].

A Topological Active Volume is a three-dimensional structure composed of interrelated nodes where the basic repeated structure is a cube. There are two kinds of nodes: the external nodes, that fit the surface of the object, and the internal nodes, that model its internal topology. The state of the model is governed by an energy function defined as follows:

$$E(v) = \int_0^1 \int_0^1 \int_0^1 E_{int}(v(r,s,t)) + E_{ext}(v(r,s,t)) dr ds dt \tag{1}$$

where E_{int} and E_{ext} are the internal and the external energy of the TAV, respectively. The internal energy controls the shape and the structure of the net. Its calculation depends on first and second order derivatives that control contraction and bending, respectively. It is defined by the following equation:

$$\begin{aligned} E_{int}(v(r,s,t)) = \alpha(&|v_r(r,s,t)|^2 + |v_s(r,s,t)|^2 + |v_t(r,s,t)|^2) + \\ &\beta(|v_{rr}(r,s,t)|^2 + |v_{ss}(r,s,t)|^2 + |v_{tt}(r,s,t)|^2) + \\ &2\gamma(|v_{rs}(r,s,t)|^2 + |v_{rt}(r,s,t)|^2 + |v_{st}(r,s,t)|^2) \end{aligned} \tag{2}$$

where subscripts represents partial derivatives and α, β and γ are coefficients controlling the first and second order smoothness of the net.

E_{ext} represents the features of the scene that guide the adjustment process and is different for external and internal nodes. It is defined as:

$$\begin{aligned} E_{ext}(v(r,s,t)) = \omega f[I(v(r,s,t))] + \\ \tfrac{\rho}{\aleph(r,s,t)} \textstyle\sum_{n \in \aleph(r,s,t)} \tfrac{1}{\|v(r,s,t) - v(n)\|} f[I(v(n))] \end{aligned} \tag{3}$$

where ω and ρ are weights, $I(v(r,s,t))$ is the intensity value of the original image in the position $v(r,s,t)$, $\aleph(r,s,t)$ is the neighborhood of the node (r,s,t) and f is a function of the image intensity, which is different for both types of nodes. For example, if the objects to detect are bright and the background is dark, the function f is defined as follows in order to minimize the energy value of external and internal nodes when they are on the surface or inside the objects, respectively:

$$f[I(v(r,s,t))] = \begin{cases} h[I_{max} - \overline{I_N(v(r,s,t))}] \text{ for internal nodes} \\ h[\overline{I_N(v(r,s,t))} + \xi(G_{max} - G(v(r,s,t)))] \\ \quad + \ \delta DG(v(r,s,t)) \text{ for external nodes} \end{cases} \tag{4}$$

ξ is a weighting term, I_{max} and G_{max} are the maximum intensity values of image I and the gradient image G, respectively, $I(v(r,s,t))$ and $G(v(r,s,t))$ are the intensity values of the original image and gradient image in the position $v(r,s,t)$, $\overline{I_N(v(r,s,t))}$ is the mean intensity in a $N \times N \times N$ cube and h is an appropriate scaling function (not used in this work). $DG(v(r,s,t))$ is the distance from the position $v(r,s,t)$ to the nearest position in the gradient image that points out an edge.

2.1 Greedy Optimization

The TAV model is automatic, so the initialization does not need any human interaction as other deformable models. As a broad outline, the greedy adjustment process consists of the minimization of the energy of an initial mesh that covers all the image and, after that, the link cutting between external nodes badly placed, this is, the external nodes that are not on the surfaces of the objects. The breaking of connections allows a perfect adjustment to the surfaces and the detection of holes and several objects in the 3D scene [3]. This procedure is explained in Section 4.

3 Brief Description of the Adapted Genetic Algorithm

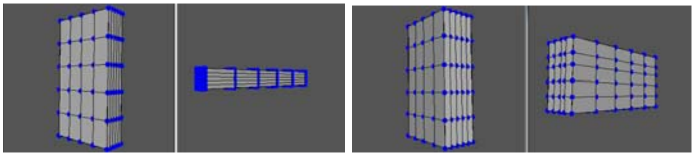
As the greedy algorithm, the GA minimizes the energy components. To this end, the genotypes code the Cartesian coordinates of the TAV nodes. Nevertheless, the key issues are the features of the genetic operators developed, as defined in [12]. Their use is briefly explained in this section.

3.1 Genetic Operators

Crossover operator. It is used an arithmetical crossover instead of the classical crossover operator because the latter produces a great number of incorrect offspring genotypes, this is, TAVs with crossings in their nodes. The new genes are defined as a weighted mean between the corresponding values in the two parent chromosomes. Figure 1 shows an example with two new individuals (TAVs) with average topologies between the selected parents.

Mutation operator. It moves any node to another random position, with a restricted movement in order to avoid crossings. To this aim, the operator computes the limits of the node mutation, taking into account its 26 neighboring nodes. With these limits, the node's coordinates are mutated at random positions within them. In the case of external nodes, virtual nodes are defined at mirrored distances of the opposite nodes in the same axis.

Spread operator. The aim of this operator is to maintain the diversity of sizes in the population since the proposed crossover operator tends to produce individuals with progressively similar sizes. The spread operator stretches a TAV in any given direction.



(a) (b)

Fig. 1. Arithmetical crossover operator. (a) Selected parents. (b) Offspring after the crossover.

Group mutation. A group of neighboring nodes randomly selected is mutated simultaneously in the same direction and with the same value. Performing a group mutation is generally more useful than mutate only a node since the internal energy is minimum when nodes are equidistant so, in most cases, a single mutation could not reduce the TAV energy.

Shift operator. It moves the TAV mesh to another position in the image. This movement allows that external and internal nodes can get into the object to segment at the same time approximately. This way, the position of the objects in the image does not affect the final node distribution.

The selection operator used is a tournament selection with small window sizes to have low selective pressures. Finally, elitism is used in the evolutions in order to preserve the best individual through generations.

4 Hybrid Approach

In this work we combine the GA global search and the greedy local search, by means of a Lamarckian strategy. This is, the greedy search is applied to each individual of the genetic population, typically a short number of greedy iterations. As a result, the fitness of the individuals is changed. A combination using a Lamarckian strategy means that the changes in the TAV structures provided by the greedy search revert to the original genotypes.

One of the advantages of the hybrid approach is that overcomes the limitation of the implemented GA related to its inability to perform topological changes in the TAV structure. The mixed model uses the procedures of the local search to cut links between adjacent external nodes after the minimization process.

4.1 Link Cutting Procedure

The link cutting procedure requires the identification of the external nodes wrongly located to break connections. Hence, the flexibility of the net in these areas will be increased, and the net will be able to improve the adjustment. These nodes wrongly located are the nodes more distant to the object edges. To this end we use the Tchebycheff's theorem. This way, an external node v_{ext} is wrongly located if its gradient distance fulfils the following inequality:

$$GD(v_{ext}) > \mu_{GD} + 3\sigma_{GD} \tag{5}$$

where μ_{GD} is the average gradient distance of the whole set of external nodes and σ_{GD} is their standard deviation.

After the identification of the outlier set, the link to remove is selected. It is the link between the node with the highest gradient distance and its worst neighbor in the outlier set. Once the link is cut, some internal nodes become external since they are on the boundaries of the net. The increase of the number of external nodes allows a better adjustment to the object boundaries. Throughout the generations, the topology of the best individual is considered in the rest of the population of the next generation. Additionally, in our implementation, the iterations of the greedy search are only performed in particular generations of the evolutionary process, typically a random number between 1 and 6.

4.2 Automatic Net Division

Since the link cutting process breaks the net topology to improve the adjustment, when the image has several objects, the net should be divided to segment them. To this end, a net reconfiguration mechanism must be developed in order to perform multiple object detection and segmentation.

The net division is performed by the link cutting algorithm. However, this algorithm cannot be applied directly to the automatic division. Since the TAV topology must be preserved, problems arise when cutting a link implies leaving isolated planes. In such case, these links cannot be cut so a "thread" composed by cubes will appear between two subnets. If one connection in one of these cubes is broken, the net topology is not preserved. Figure 2 shows these ideas in 2D for a better visualization. Figure 2(a) presents an example with a "thread". Figure 2(b) depicts a case that leads to threads. If the labelled link is removed, there will be two threads since no other link can be cut. The 3D case is equivalent.

However, this problem can be overcome if we consider a direction in the cutting process, as done by Bro-Nielsen [10]. Thus, a cutting priority is associated to each node which connections are removed. A higher priority is assigned to the nodes in

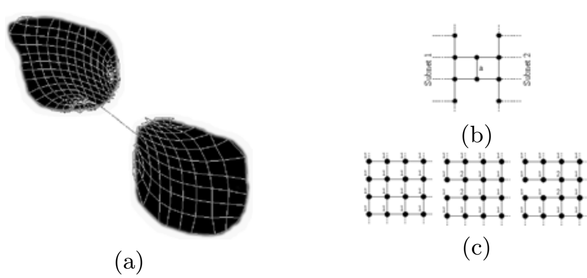


Fig. 2. Threads and cutting priorities in 2D. (a) Image segmentation with threads. (b) If link "a" is removed, no other link can be removed in order to preserve the TAN topology. (c) Recomputation of cutting priorities. When a link is broken in a direction, the neighborhood in this direction increases its priorities.

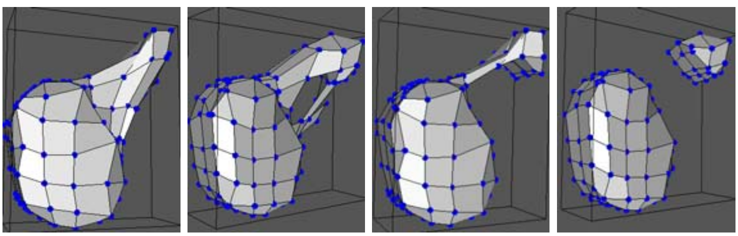
the cutting direction whereas a lower priority is assigned to the nodes involved in the cut. Figure 2(c) shows the recomputation of the node priorities after several cuts in the 2D case. The extension for the 3D case is straightforward.

The cutting priority weights the gradient distance of each node. Thus, once the set of badly placed external nodes is obtained using equation 5, the link to remove consist of two neighboring nodes within this set, n_1 and n_2, that fulfill:

$$\begin{array}{l} GD_{v_{ext}}(n_1) \times P_{cut}(n_1) > GD(n) \times P_{cut}(n),\ \forall n \neq n_1 \\ GD_{v_{ext}}(n_2) \times P_{cut}(n_2) > GD_{v_{ext}}(m) \times P_{cut}(m), \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall m \neq n_2, m \in \aleph(n_1) \end{array} \tag{6}$$

where $P_{cut}(x)$ is the cutting priority of node x, $GD_{v_{ext}}(x)$ is the distance from the position of the external node x to the nearest edge, and $\aleph(n_1)$ is the set of neighboring nodes of n_1.

Figure 3 shows an illustrative example of a breaking process in a net division.



(a) (b) (c) (d)

Fig. 3. Example of segmentation with a breaking sequence. (a) Individual before breaking. (b) and (c) Intermediate steps. (d) Final result after the automatic net division.

5 Results with the Hybrid Approach

This section presents some representative examples. In all of these, the same image was used as the external energy for both internal and external nodes, and all the test images had 256 gray levels. The examples show the capabilities and main difficulties of the alternatives developed for the energy minimization.

Table 1. TAV parameter sets in the segmentation processes of the examples

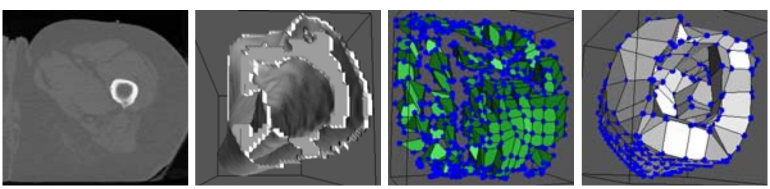
Figure	*Size*	α	β	γ	ω	ρ	ξ	δ
3	$6 \times 6 \times 4$	3.5	0.5	0.5	20.0	4.5	5.0	10.0
4	$8 \times 8 \times 8$	7.5	2.5	1.5	10.0	2.5	3.0	5.0
5(a)	$6 \times 6 \times 4$	3.5	0.5	0.5	20.0	4.5	5.0	10.0
5(b)	$8 \times 8 \times 8$	5.5	0.5	1.0	10.0	4.5	6.0	7.0
6	$8 \times 8 \times 8$	9.0	0.1	0.1	10.0	2.5	3.0	4.0

Table 1 includes the TAV parameters used in the segmentation examples. The TAV parameters were experimentally set as the ones in which the genetic algorithm gave good results, although it is very less sensitive to changes in parameters than the greedy procedure. We have used a tournament selection with a window size of 3% of the population and elitism of the best individual. The probabilities of the operators were experimentally set, too, taking values in the range where the best test results were obtained. The probabilities used were: crossover= 0.5; mutation= 0.0005; spread= 0.01; shift= 0.05 and group mutation= 0.001.

The execution time, with around 2,600 individuals, 1,200 generations and a $8 \times 8 \times 8$ TAV, is usually between 14 and 15 hours in an Intel Core 2 2.4 GHz. Nevertheless, the process can be faster maintaining acceptable results with fewer generations. The processing time of the GA process depends only on the size of the net and the population. The image size is not relevant.

5.1 Segmentation of Images with Noise That Require Topological Changes

We tested the hybrid method in real images that require topological changes. Figure 4 presents an example in the medical domain. In this case, the hybrid approach uses the link cutting procedure explained in Section 4.1. The example corresponds to a 3D image of a humerus composed by CT slices, as the one shown in figure 4(a). Figure 4(b) is a 3D reconstruction from the 2D slices. In this case, the greedy algorithm could not achieve a fine segmentation (Figure 4(c)) meanwhile the hybrid algorithm obtains a good result in this type of real images with noise and fuzzy contours (Figure 4(d)).

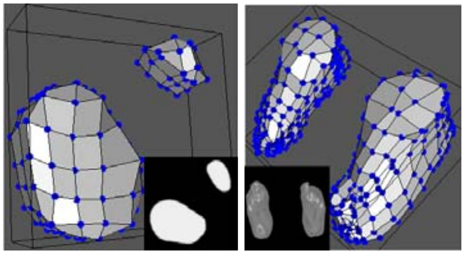


(a) (b) (c) (d)

Fig. 4. (a) Slice of the CT images set. (b) 3D representation of the humerus. (c) Segmentation with the greedy approach. (d) Segmentation with the hybrid algorithm.

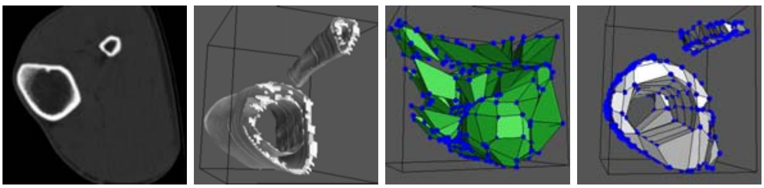
5.2 Segmentation of Images with Several Objects

The hybrid approach uses now the automatic net division procedure of the local search. We tested the hybrid version in artificial and real images with several objects. Figure 5 shows examples of segmentation that require the net division procedure. Figure 5(a) shows an example with an artificial image whereas Figure 5(b) shows the result with a real CT image of the feet.



(a) (b)

Fig. 5. Example of segmentation with several objects in the scene. (a) Segmentation with two artificial objects from CT images. (b) Segmentation of two feet. The insets show examples of 2D slices used as input to the segmentation process.



(a) (b) (c) (d)

Fig. 6. Image with several objects. (a) Slice of the CT images set. (b) 3D representation of the tibia and fibula. (c) Segmentation with the greedy approach. (d) Segmentation with the hybrid algorithm.

Genetic algorithm vs. greedy results. Figure 6 shows the comparison of results with both algorithms in a real domain. The image is composed of a sequence of CT images that contain two bones, a tibia and a fibula. Figure 6(a) represents a slice of this CT image set. Figure 6(b) shows the 3D reconstruction from the 2D slices. In addition of the fuzziness of the contours of the two bones, the external contour of the leg introduces a contrast in the background gray level that the algorithms must overcome. Due to this, the greedy approach cannot achieve a good segmentation (Figure 6(c)) meanwhile the hybrid algorithm overcomes the external contour and the image noise to provide a correct division of the subnets (Figure 6(d)). Note that the bigger bone requires the link cutting procedure to segment the hole of its internal part.

6 Conclusions

We have presented new approaches to the energy minimization task in the Topological Active Volume model. We have used a hybrid Lamarckian combination of the greedy local search with the global search of the genetic algorithm.

The hybrid combination developed was tested with several images, from the artificial domain to a real one. The new approach achieved a good adjustment to the objects and improved the results of the greedy algorithm. The hybrid approach is not sensitive to noise and it obtains good segmentations in images with fuzzy contours. It is useful in images that require the use of the topological changes provided by the local search, together with the advantages of the global search. Specially, the approach obtains correct segmentation results in images with several objects or complex surfaces.

References

1. Tsumiyama, K.S.Y., Yamamoto, K.: Active net: Active net model for region extraction. IPSJ SIG. notes 89(96), 1–8 (1989)
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1(2), 321–323 (1988)
3. Barreira, N., Penedo, M.G.: Topological Active Volumes. EURASIP Journal on Applied Signal Processing 13(1), 1937–1947 (2005)
4. Ballerini, L.: Medical image segmentation using genetic snakes. In: Proceedings of SPIE: Application and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation II, vol. 3812, pp. 13–23 (1999)
5. Séguier, R., Cladel, N.: Genetic snakes: Application on lipreading. In: International Conference on Artificial Neural Networks and Genetic Algorithms (2003)
6. Ibáñez, O., Barreira, N., Santos, J., Penedo, M.G.: Genetic approaches for topological active nets optimization. Pattern Recognition 42, 907–917 (2009)
7. Jones, T.N., Metaxas, D.N.: Automated 3D segmentation using deformable models and fuzzy affinity. In: Duncan, J.S., Gindi, G. (eds.) IPMI 1997. LNCS, vol. 1230, pp. 113–126. Springer, Heidelberg (1997)
8. Qiu, B., Clarysse, P., Montagnat, J., Janier, M., Vray, D.: Comparison of 3D deformable models for in vivo measurements of mouse embryo from 3D ultrasound images. In: Ultrasonics Symposium, 2004, vol. 1, pp. 748–751. IEEE, Los Alamitos (2004)
9. McInerney, T., Terzopoulos, D.: Topology adaptive deformable surfaces for medical image volume segmentation. IEEE Transactions on Medical Imaging 18(10), 840–850 (1999)
10. Bro-Nielsen, M.: Active nets and cubes. Technical Report 13, IMM, Technical University of Denmark (1994)
11. Bro-Nielsen, M.: Modelling elasticity in solids using active cubes - application to simulated operations. In: Ayache, N. (ed.) CVRMed 1995. LNCS, vol. 905, pp. 533–541. Springer, Heidelberg (1995)
12. Novo, J., Barreira, N., Santos, J., Penedo, M.G.: Topological active volumes optimization with genetic approaches. In: XII Conference of the Spanish Association for the Artificial Intelligence, vol. 2, pp. 41–50 (2007)