

Deformable Object Segmentation and Contour Tracking in Image Sequences Using Unsupervised Networks

Ana-Maria Cretu, Emil M. Petriu, Pierre Payeur and Fouad F. Khalil

*School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada
[acretu, ppayeur, petriu, fkhalil]@site.uottawa.ca*

Abstract

The paper discusses a novel unsupervised learning approach for tracking deformable objects manipulated by a robotic hand in a series of images collected by a video camera. The object of interest is automatically segmented from the initial frame in the sequence. The segmentation is treated as clustering based on color information and spatial features and an unsupervised network is employed to cluster each pixel of the initial frame. Each pixel from the clustering results is then classified as either object of interest or background and the contour of the object is identified based on this classification. Using static (color) and dynamic (motion between frames) information, the contour is then tracked with an algorithm based on neural gas networks in the sequence of images. Experiments performed under different conditions reveal that the method tracks accurately the test objects even for severe contour deformations, is fast and insensitive to smooth changes in lighting, contrast and background.

1. Introduction

The analysis of image sequences is a basic task in many computer vision applications. The increasing need for fully automated video analysis algorithms and the availability of good quality and relatively cheap video cameras has generated lately a lot of interest in the topic of object tracking. Object tracking involves in general three steps: the segmentation of moving objects in an image, the tracking of such objects from frame to frame and the analysis of object tracks in order to evaluate their behavior. Many approaches have been explored for both color image segmentation and object tracking. Among them, neural network based solutions have received considerable attention due to the fact that these methods are usually more effective and efficient than traditional ones [1]. The object segmentation problem can be viewed as a clustering problem in a feature space derived from the color and motion information and therefore is well

suited for unsupervised modeling approaches. Unsupervised learning is in general preferred to supervised learning because the latter requires a set of training samples, which may not be available, especially when the image features are unknown or when a certain degree of automation is desired.

There are various neural network-inspired solutions proposed for the segmentation of objects based on color from images. One approach is to classify image pixels into object of interest (or foreground) and background. Maddalena and Petrosino [2] use a self-organizing network for background and foreground modeling from video sequences in which for each image pixel a neuron map is built based on its color components and the information stored in each pixel is updated only if its best matching weight vector is close enough to the background model based on a predefined distance; otherwise the pixel is considered as belonging to a moving object. A similar idea is employed by Luque *et al.* [3], who design an unsupervised competitive neural network to serve as a classifier of pixels as background or foreground in a video sequence of traffic surveillance. Both of these procedures are costly in computation time as they require a pixel-by-pixel processing of each frame.

Another approach for object segmentation from images uses color information. A hierarchical two-stage self-organizing network is employed in [4] to segment images based on color, where the first stage deals with the unsupervised grouping of input color image pixel values in order to identify groups of typical colors and the second stage determines the dominant colors from the group of typical colors. Jiang *et al.* [5] perform image segmentation based on a self-organizing map having as input five-dimensional feature vectors (representing the x, y coordinates of a pixel and its RGB values). The result is further refined with a gliding window; if an isolated pixel is detected in the gliding window, its associated class is substituted by the most frequent class that occurred in that window. Blocks with the least number of pixels

are incorporated into their largest neighboring block and also blocks with the least mean of RGB values are merged into their nearest neighboring block. Dong and Xie [6] propose a system for color image segmentation that employs both unsupervised and supervised segmentation. For the unsupervised segmentation, the LUV color components of an image are projected into a small set of prototypes using a self-organizing map from which optimal clusters are found using simulated annealing. The supervised segmentation includes color learning and pixel classification. All of these solutions are limited because the network parameters (most importantly the network size) are determined heuristically, the number of clusters in which the image is divided is set *a priori* and it is not obvious how to select these parameters for a certain image and color.

After the identification of the object of interest, the aim of an object tracker is to generate the trajectory of the object in time by locating its position in every frame of the video [7]. Several neural network solutions have been proposed for tracking of rigid objects in image sequences. Luque *et al.* [8] track rigid objects (e.g. cars in traffic sequences) with a growing competitive network. Each object in a scene is assigned to a neuron, and that neuron represents and identifies that particular object. Neurons are added or deleted when new objects enter or exit the scene. In [9], the authors apply a variation of growing neural gas, to track basic circular computer-generated rigid objects in video sequences. Rodriguez *et al.* [10] use the topology of a growing neural gas network to determine the posture of a hand in an image sequence, and the adaptation dynamics of the same neural network in time to determine the gesture performed by the hand. Angelopoulou *et al.* [11] describe contours of hands by an active growing neural gas network where each node of the network is described by its geometrical position, the underlying local feature structure of the image, and the distance vector between the reference image and any successive images. An unsupervised solution for tracking of the outer boundary of the lips in an image sequence is proposed in [12]. The mouth contour is determined based on skin color and spatial knowledge (location in the lower half of the detected face). The lip motion is tracked by a time adaptive self-organizing map [13], a modified self-organizing map network where each neuron has an individual learning rate and neighborhood. Apart from the last solution, all the other ones refer to rigid object tracking and cannot be adapted to deformable object tracking. Some are limited by factors such as uniform uncluttered backgrounds and some by their ability to handle grey level images only.

In order to enable the automatic recuperation of elastic parameters and/or the learning of elastic behavior of deformable objects that would allow for their manipulation using robotic hands, research can greatly benefit from the development of automated segmentation and tracking procedures for such objects. The work proposed in this paper is a continuation of our research on the topic of 3D deformable object sensing and modeling. In [14] we have successfully modeled the elastic behavior of a deformable object as a complex relationship from measured forces to deformation profiles (captured by a laser range finder) using neural networks. The work in this paper aims at improving the accuracy of the elastic behavior modeling by capturing the full contour of the object during deformation, not only a single deformation profile. While the paper addresses the problem of automatic deformable object segmentation and tracking in image sequences, it aims further than the classical segmentation/tracking solutions. The neural-network solution proposed does not only segment and track, but also models the elasticity of the object, by correlating measured forces at the level of the robotic fingers with the deformation. As such, it generates maps toward the learning and recognition of the nature of an object's elastic behavior. Such a description will be useful to control the hand during manipulation of deformable objects. It is also important to mention that the experimentation takes place in a relatively controlled environment. The accent is on a fast algorithm that is insensitive to smooth changes in lighting, contrast and background. The solution does not have to deal with multiple moving objects and severe changes in the environment, but rather with accurately tracking severe contour deformations.

The proposed solution for color-based segmentation relies on an unsupervised ontogenic network that is a network that builds by itself, by incrementally adding node by node to a neural map. In this way the constraint imposed by the fixed map size of standard unsupervised networks (e.g. self-organizing maps) is eliminated. Moreover, the adaptation parameters of the network are constant, therefore eliminating the need for selecting them heuristically. The tracking is also based on a neuro-inspired algorithm that models the contour of a deformable object. In other words, the model predicts the new position of its neurons and readjusts their position to map this contour. This adaptation is executed automatically as well. The solution proves to be fast and robust against severe contour deformations, smooth changes in lighting, contrast and background.

2. Proposed framework

The proposed approach for deformable object segmentation and tracking can be summarized as follows: the object of interest is automatically segmented from the initial frame in a sequence of images. The segmentation is treated as clustering based on color information (HSV color components) and spatial features (x, y coordinates of each pixel). The HSV color space is chosen because it represents better the color similarities and is able to better identify pixels on the same surface in spite of some differences between their colors due to non-uniform illumination or shading effects. An unsupervised ontogenic network, namely a growing neural gas [15], is employed to cluster each pixel of the initial frame. The clustering results are then classified as one of two categories: object of interest or background based on the mean HSV value. The contour of the object is identified based on this classification and a second growing neural gas is applied this time to represent the position of each point over the contour. This compact growing neural gas description of the contour is then used as an initial configuration for a sequence of neural gas networks that will track the contour in the image sequence. To speed up the procedure, the training takes place only for those frames where a motion occurs with respect to the previous frame.

2.1. Deformable object segmentation

A growing neural gas (GNG) is employed to cluster each pixel of an HSV coded image. The growing neural gas algorithm can be described as follows [15]: For each input vector presented to the network, two best matching nodes are selected, whose weights are the closest to the input, based on their Euclidean distance. A neighborhood connection is created between them if the connection does not already exist and its age is set to 0. The position of these nodes and the ones of the topological neighbors of the winner unit are updated such that they better fit the input. All edges that are not used increase in age and if the age exceeds a threshold, the corresponding edges are deleted. Any node that has no edge connection is removed as well. After a defined number of iterations, a new node is added to support the node that has accumulated the highest error in the previous steps. The new node is placed between the node with the highest error and one of its neighbors that has the next highest error. A global decrease of errors is then performed. The algorithm continues until some stopping criterion is met. The full mathematical formulation is presented in [15].

In the context of this work, the growing neural gas network receives as input a vector of the form $\mathbf{X} = \{x_1, \dots, x_n\}$, with x_i being a five-dimensional vector of the form $[H_i, S_i, V_i, X_i, Y_i]$, with $i = 1 \dots n$, where n represents the number of pixels in the image. The input vector \mathbf{X} is therefore a set of given HSV color coded pixels together with their coordinates in an image. The inclusion of the coordinates of the pixels in the image aside from the color information improves the clustering results by providing spatial information on the position of the pixel without adding computational burden. The computation time is proportional to the number of adaptation steps and not to the size of the input vector. However, it is worth mentioning that generally the clustering results depend more on the first components of the input vector. The addition of too many inputs will not bring improvements in the learning procedure on the contrary it might lead to larger errors in the clusters. The growing neural gas maps the \mathbf{X} input on a smaller array of nodes \mathbf{M} such that pixels with similar color and position in \mathbf{X} are projected to nearby nodes in \mathbf{M} . Initially \mathbf{M} is empty. The growing gas algorithm adds supplementary nodes into the network structure at the position where the accumulated error is the highest, and when the number of learning iterations performed is an integer multiple of some predefined value. The growth of the network is terminated when a predefined stopping criterion is met (e.g. a minimum error is reached). \mathbf{M} has the same form as \mathbf{X} , $\mathbf{M} = \{m_1, \dots, m_r\}$, $m_j = [H_j, S_j, V_j, X_j, Y_j]$, with $j = 1 \dots r$, with $r < n$.

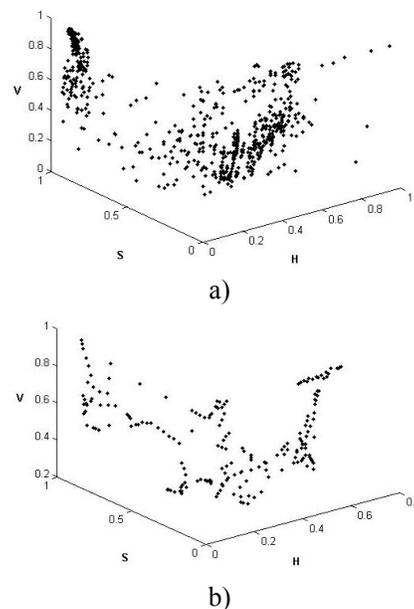


Figure 1: a) Example of distribution of HSV color values in the HSV color space, b) the corresponding growing neural gas map, M.

Figure 1a shows an example of an initial distribution of HSV color values in the HSV color space, while Figure 1b shows the corresponding map \mathbf{M} obtained when the color coded HSV values in Figure 1a and the spatial coordinates of the corresponding pixels are presented to a growing neural gas network. The X and Y dimensions are also taken into account during clustering, but are not displayed explicitly due to the limitation of graphical display in five dimensions.

In order to segment an object or multiple objects with similar color in a selected image, \mathbf{M} is split into two clusters: foreground (object(s) of interest) denoted $C_f = \{m_1, \dots, m_{rf}\}$, where m_f belong to \mathbf{M} with $f = 1 \dots rf$, and background, denoted $C_b = \{m_1, \dots, m_{rb}\}$, where m_b belongs to \mathbf{M} , with $b = 1 \dots rb$ and $rb + rf = r$. The classification is based on the knowledge that generally the background is darker in color than the object of interest (the distinction can be made based on the mean HSV value). If this assumption doesn't stand up, a swap of the two clusters can be performed to ensure that the algorithm works correctly. Figure 2 presents the foreground (denoted by circles) and background (denoted by dots) for the growing neural gas map \mathbf{M} in Figure 1b. In order to select the color of interest, which is the color of the object to be segmented, the mean is computed for all HSV values in the foreground cluster C_f , $O_c = \text{mean}(H_k, S_k, V_k)$, with $k = 1 \dots rf$. The mean HSV value, O_c , for the object in the presented example is marked with a larger square in Figure 2.

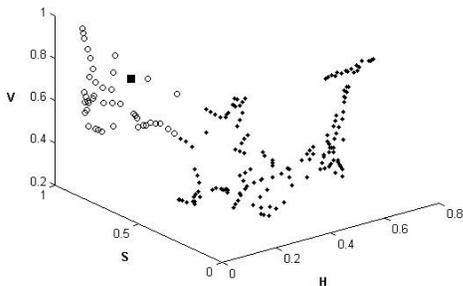


Figure 2: Identified background (dots) and object (circles) clusters in the HSV space, and mean HSV color (larger square).

The procedure is applied only on the first frame of the sequence. The identified HSV color code O_c is then searched in the initial image and over all images in the sequence where movement occurs. All pixels with the mean HSV color code or a very similar code (within a tolerance level) are replaced with 1 and the rest with 0 to segment the object of interest. The tolerance level allows for the fine tuning of areas on the object surface with different lighting conditions and around the object edges that are usually perceived darker because of the

shadow effect. A median filter is finally applied on the result to reduce isolated patches of color.

2.2. Deformable object tracking

2.2.1. Contour identification in the initial frame.

The initial image is down-sampled in order to speed up the color identification procedure and is transformed to HSV color coding. The procedure in Section 2.1 is applied to obtain the color of interest and this color is replaced with 1s and the background with 0s in the initial frame. The contour of the object is then identified in this resulting binary image with the Sobel edge detector and a second growing neural gas is applied this time on the two-dimensional vector that represents the position of each point in the contour, $\mathbf{X}_c = \{x_{c1}, \dots, x_{cn}\}$, with $x_{ci} = [X_i, Y_i]$, $i = 1 \dots cn$. The main purpose of this network is to detect the optimum number of points (stored in \mathbf{M}_{GNG}) on the contour that accurately represent its geometry. Based on the identified HSV value and on the dynamic information in form of motion between frames in the sequence, this compact representation of the contour of the object obtained by the second growing neural gas is tracked with an algorithm based on a third neural architecture, namely the neural gas, in the sequence of images.

2.2.2. Contour tracking with neural gas. The main purpose of neural gas (NG) networks [16] is to cluster multi-dimensional vectors. It consists of nodes which independently move over the input data space while learning. The algorithm starts by initializing the set of network nodes with a predefined number of units whose corresponding reference vectors are chosen randomly according to a probability density function or from a finite set. Each unit has an associated reference vector that indicates its position in the input space. At each training step, the winning neuron that best matches an input vector is identified using the minimum Euclidean distance criterion. The neurons to be adapted in the learning procedure are selected according to their rank in an ordered list of distances between their weights and the input vector. A full description of the algorithm is available in [16].

In the context of the present work, the neural gas is employed to track the contour of an object from a sequence of images by predicting and readjusting the position of its neurons to follow the contour. Similar to the second growing neural gas described above, it receives as input vectors x_{ci} of the form $[X_i, Y_i]$, $i = 1 \dots cn$. It starts with an initial configuration of nodes, in this case set to the initial contour points obtained using the growing neural gas (from \mathbf{M}_{GNG}). These nodes move over the data space during adaptation and

the model contracts asymptotically towards the points in the input space, taking the shape of the contour.

While somewhat similar to growing neural gas, the main difference is the fact that the number of nodes in the output map \mathbf{M}_{NG} of the neural gas is predefined. In this case they are set equal to the number of points in \mathbf{M}_{GNG} . In the context of this application, as the optimum number of points to represent the contour is known (from the initial frame analyzed with the growing neural gas) the proposed algorithm will use this number of nodes to track the contour over the image sequence. The use of a fixed size network, with a predefined number of nodes ensures a unified representation of the contour in all frames and makes easier the study of the trajectory of points in the contour as each point can be referred to its previous position. As well, the capability of the network to start with a set of predefined positions allows the tracking algorithm to always continue from the shape of the contour in the previous frame. Therefore it increases the speed of the training procedure as the movement is in most cases smooth from one frame to the other.

2.2.3. Tracking over a sequence of images. At the time a new frame is presented to the algorithm, an intensity difference is computed between the gray-scale representation of the new frame and the previous one. The purpose is to identify if motion occurred in the scene. This motion can be either the result of the manipulation of the object which leads to its deformation, or the result of some external factors, such as a change in the background. To capture deformations of the object of interest, the frame is transformed to HSV color coding and the object of interest and its contour are identified based on the mean HSV color as identified from the first frame. A neural gas network (initialized with the contour of the object in the previous frame) is used to predict the new position of its neurons and to readjust them to fit the new contour. This new contour will be used to initialize the next neural gas network in the sequence when a new frame in which motion occurs is presented again to the algorithm. The procedure is repeated until the last frame of the sequence.

By computing intensity differences between frames, the proposed solution can handle smooth changes in the contrast and lighting conditions. Moreover, due to the fact that the algorithm searches for the color of interest in the frame each time a motion occurs, the proposed solution can also handle cluttered and moving backgrounds as long as it does not interfere with the mean HSV color of the object.

3. Experimental Results

In order to validate the proposed modeling framework, experimentation was conducted on a set of deformable objects with different shapes and made from different materials (namely a round foam ball, a oval rubber ball and a rectangular foam sponge and a long foam sponge) manipulated with a 3-fingered Barrett robotic hand, as shown in Figure 3a-d. The deformations of the object as a result of the interaction with the robotic hand's fingers are recorded in a series of image sequences collected with a camera located perpendicularly over the object.

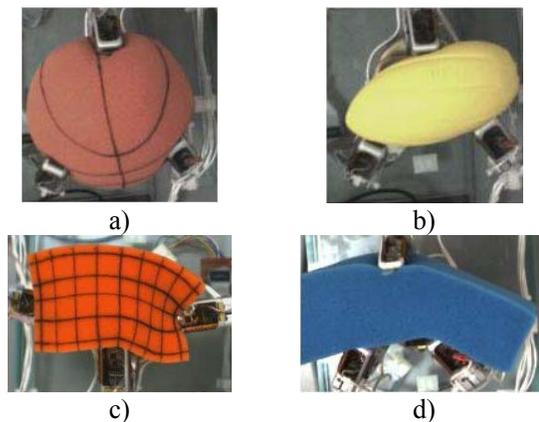


Figure 3: a) Round foam ball, b) oval rubber ball, c) rectangular foam sponge, and d) long foam sponge.

The color-based segmentation steps for the initial frame in case of the round foam ball in Figure 3a are illustrated in Figure 4. The distribution of HSV values in the HSV color space, the growing neural gas map, its division into foreground and background and the mean HSV value of the object's color are shown in Figures 1 and 2. Figure 4a shows the down-sampled image of the round foam ball by 50%, from 160×160 pixels to 80×80 pixels in the window of interest. Figure 4b shows that by only replacing the color of interest with 1 and the rest of the image with 0, the result is rather rough and sensitive to non-uniform lighting conditions and shadow effects. Several parts of the object that are slightly darker are erroneously classified as background. The inclusion of a tolerance brings an improvement in the results obtained, as illustrated in Figure 4c. The tolerance is the maximum distance for each component of the HSV coding from the mean HSV code that identifies the color of interest. Overall, for all the examples presented, the level of tolerance is set to low values (less than 0.09 for H and S values and less than 0.02 for the V values). Its role is

to compensate for slight shading and color variations resulting from the lighting conditions and/or shadow effects. A median filter is finally applied to reduce isolated patches and smooth the edges. The filtered image is shown in Figure 4d.

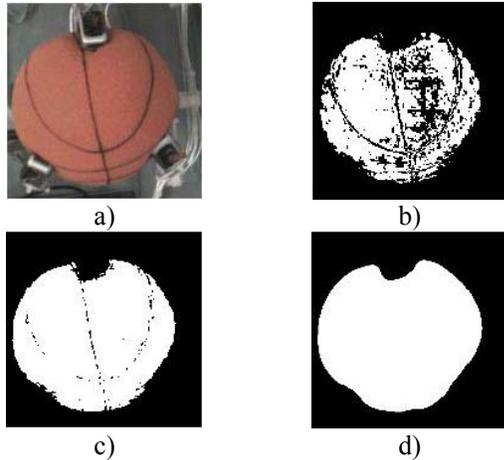


Figure 4: a) Down-sampled image of round foam ball and results obtained b) when replacing the color of interest with 1 and background with 0, c) when a tolerance level of 0.07 for H and S and 0.01 for V value is accepted, and d) after median filtering.

In order to show the benefit of using an unsupervised neural network to segment an image, the proposed method is compared for all the objects under study with a standard color-based segmentation. The latter is based on mean HSV values computed in a user selected frame that samples the object color. The same tolerance level and median filtering are applied as in the case of the proposed solution. The ground truth is obtained by a manual segmentation of the object of interest and the average error rate is computed using:

$$E = \frac{\sum_{i=1}^f N_{ow}^i + N_{bw}^i}{N * f}$$

where N_{ow}^i is the number of pixels that belong to the foreground but are wrongly classified as background in the i -th frame, N_{bw}^i is the number of pixels that belong to the background but are wrongly classified as foreground at the i -th frame, N is the total number of pixels in a frame and f is the number of frames in the image sequences. The results are presented in Figure 5. Apart from being automated, the proposed solution shown in blue reaches slightly lower errors than the standard color-based segmentation shown in red.

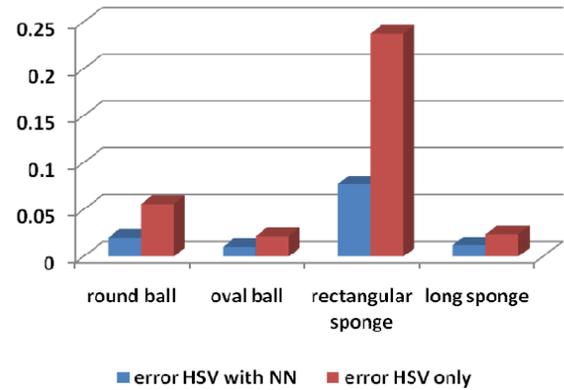


Figure 5: Average error rate for segmentation.

The image obtained after median filtering (as in Figure 4d) is then used to detect the contour of the object of interest, as shown in Figure 6a. As seen in Figure 6b where the contour is overlapped over the initial frame, the contour is accurately identified. The growing neural gas model, M_{GNG} , trained to capture the contour is shown with red dots in Figure 6c.

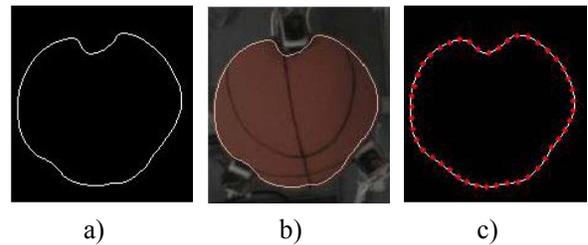


Figure 6: a) Contour detection, b) contour displayed over the initial frame, and c) growing neural gas model marked with red.

Starting from this model, a series of neural gas networks is employed to track the contour of the object as it changes under the action of the robotic hand. Each neural gas network starts with the configuration of nodes from the previous frame in order to minimize the movement of nodes during adaptation. This assumption is valid as the movement is usually smooth from one frame to the other.

Figures 7 to 10 illustrate different frames and the corresponding configuration of tracking points (points in M_{NG} , shown with red dots) for each the four test objects. It can be seen that the tracker follows accurately the contour of each deformed object. As a pilot study, the algorithm is tested on the Matlab platform running on a Pentium 1.3GHz machine with 512MB memory. It achieves a speed of between 2 and 5 fps depending on the speed at which the

deformations of the object occur, which impacts the frequency of the updates for the neural gas nodes.

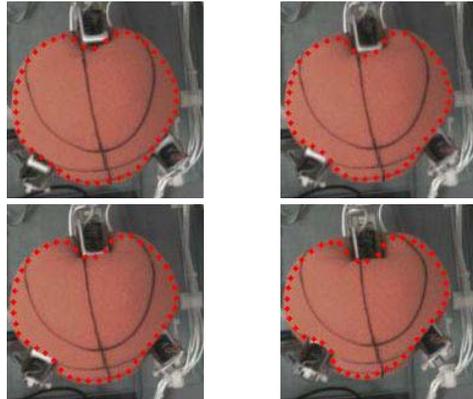


Figure 7: Contour tracking on the foam ball.

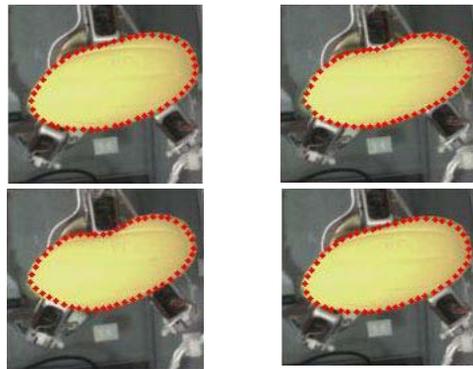


Figure 8: Contour tracking on the rubber ball.

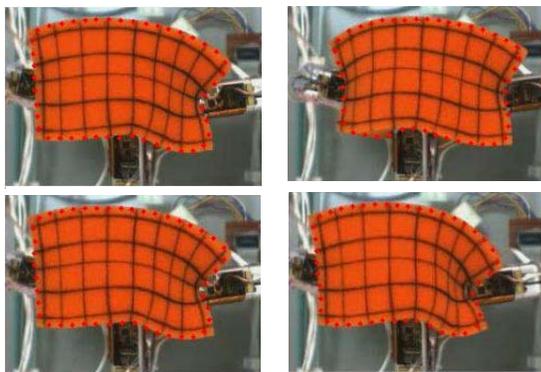


Figure 9: Contour tracking on the rectangular foam sponge.

The choice of a fixed size neural gas network allows for an easy and fast study of the trajectory of the points on the contour between successive frames during adaptation. The knowledge of these trajectories enables the description of the deformation behavior under the force exercised with the robotic fingers. This representation provides a solid basis for the

anticipation of the deformation of the object under manipulation.

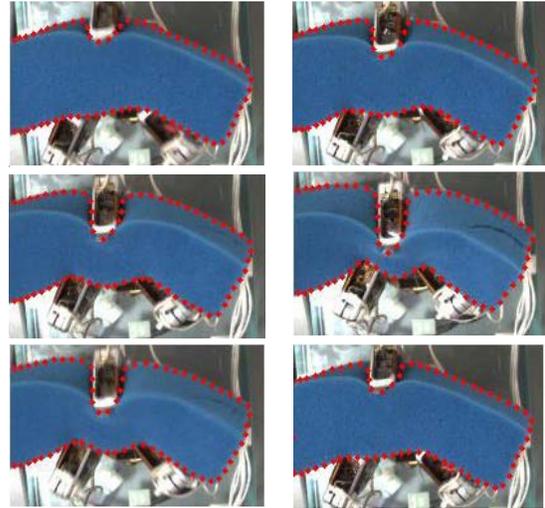


Figure 10: Contour tracking on the long sponge.

Figure 11 shows a part of the complex trajectory that the nodes of the long sponge in Figure 10 follow from the first image to the last image in the sequence (the sponge is squeezed and released after). The movement of nodes occurs from the exterior towards the interior contours. The trajectory is marked with arrows for only a few nodes not to overload the figure. One can observe that the nodes retain their correspondence throughout the deformation due to the choice of a fixed number of nodes used in the neural gas network and that a one-to-one correspondence of the points in the trajectory avoids the mismatch of the points during the deformation. It also allows for the tracking of the lateral motion along the sides of the object (that occurs beyond its compression), as depicted in the enlarged Figures 11b and c.

It can be noted that the proposed solution, by imposing continuity in the displacement of shape points, brings an additional geometrical constraint over the object deformation. As such, the technique goes beyond a standard color-based approach for tracking. Given that contour tracking is dedicated here to deformable objects, one cannot rely on other information, such as shape constraints for example, that could be otherwise used as a basis for tracking. All these properties contribute to the advantages of the proposed solution over classical and neural-network based solutions for segmentation and tracking, and provide an approach to estimate the elasticity characteristics of an object based on its contour deformation when submitted to external forces.

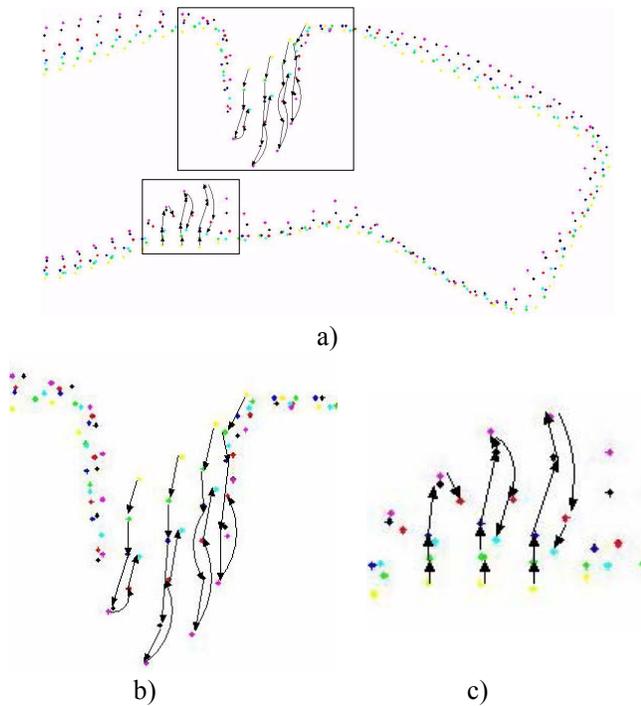


Figure 11: Long sponge: a) trajectory of nodes between successive configurations, b) and c) parts of enlarged trajectory from a).

4. Conclusion

The paper demonstrates the benefit of using unsupervised neural networks for the purpose of deformable object segmentation and contour tracking in image sequences. It is relatively fast, robust and eliminates the constraints of standard unsupervised networks with respect to the choice of parameters and the predetermined size of the network. As a future development, the trajectory of the deformation contours obtained by the proposed algorithm will be correlated with force data measured at the robotic hand's fingers in order to allow for the description and prediction of elastic behavior of objects. Such knowledge will enable safe robotic hand manipulation of deformable objects.

5. References

[1] N.C. Yeo, K.H. Lee, Y.V. Venkatesh, and S.H. Ong, "Colour Image Segmentation Using the Self-Organizing Map and Adaptive Resonance Theory", *Image and Vision Computing*, vol. 23, pp. 1060-1079, Elsevier, 2005.

[2] L. Maddalena, and A. Petrosino, "A Self-Organizing Approach to Detection of Moving Patterns for Real-time Applications", *Advances in Brain, Vision, and Artificial Intelligence*, LNCS 4729, pp. 181-190, Springer, 2007.

[3] R.M. Luque, E. Dominguez, E.J. Palomo, and J. Munoz, "A Neural Network Approach for Video Object Segmentation in Traffic Surveillance", *ICIAR*, LNCS 5112, pp. 151-158, Springer, 2008.

[4] S.H. Ong, N.C. Yeo, K.H. Lee, Y.V. Venkatesh, and D.M. Cao, "Segmentation of Color Images Using a Two-Stage Self-Organizing Network", *Image and Vision Computing*, vol. 20, pp. 279-289, Elsevier, 2002.

[5] Y. Jiang, K.-J. Chen, and Z.H. Zhou, "SOM Based Image Segmentation", *Lecture Notes in Artificial Intelligence*, G. Wang, Q. Liu, Y. Yao, A. Skowron (Eds.), vol. 2639, pp. 640-643, Springer, 2003.

[6] G. Dong, and M. Xie, "Color Clustering and Learning for Image Segmentation Based on Neural Networks", *IEEE Trans. Neural Networks*, vol. 16, no.4, pp. 925-936, 2005.

[7] A. Yilmaz, O. Javed, and M. Shah, "Object Tracking: A Survey", *ACM Computing Surveys*, vol. 38, no. 4, pp. 1-45, ACM Press, 2006.

[8] R.M. Luque, J.J. Ortiz-de-Lazcano-Lobato, E. Lopez-Rubio, and E.J. Palomo, "Object Tracking in Video Sequences by Unsupervised Learning", *CAIP*, LNCS 5702, pp. 1070-1077, Springer, 2009.

[9] G.A. Drumea and H. Frezza-Buet, "Tracking Fast Changing Non-stationary Distributions with a Topologically Adaptive Neural Network", *Proc. Symp. Artificial Neural Networks*, pp. 43-48, Belgium, 2007.

[10] J.G. Rodriguez, F. Flórez-Revuelta, and J.M. García-Chamizo, "Learning and comparing trajectories with a GNG-based architecture", *DCAI*, J.M. Corchado et al. (Eds.), pp. 644-652, Springer, 2009.

[11] A. Angelopoulou, A. Psarrou, G. Gupta, and J.G. Rodriguez, "Robust Modeling and Tracking of Nonrigid Objects using Active-GNG", *Proc. IEEE Int. Conf. Computer Vision*, pp. 1-7, Rio de Janeiro, Brazil, 2007.

[12] M.K. Moghaddam and R. Safabakhsh, "TASOM-Based Lip Tracking Using the Color and Geometry of the Face", *Proc. Conf. Machine Learning and Applications*, pp. 63-68, 2005.

[13] H. Shahhosseini and R. Safabakhsh, "A TASOM Based Algorithm for Active Contour Modeling", *Pattern Recognition Letters*, 24, pp. 1361-1373, Elsevier, 2003.

[14] A.-M. Cretu, P. Payeur and E.M. Petriu, "Neural Network Mapping and Clustering of Elastic Behavior from Tactile and Range Imaging for Virtualized Reality Applications", *IEEE Trans. Instrumentation and Measurement*, vol. 57, no. 9, pp. 1918-1928, Aug. 2008.

[15] B. Fritzsche, "Unsupervised Ontogenic Networks", *Handbook of Neural Computation*, E. Fiesler, IOP Publishing Ltd and Oxford University Press, 1997.

[16] T.M. Martinetz, "Competitive Hebbian Learning Rule Forms Perfectly Topology Preserving Maps", *Proc. Int. Conf. Artificial Neural Networks*, Springer, Amsterdam, pp. 427-434, 1993.