CONVEX SHAPE RETRIEVAL FROM EDGE MAPS BY THE USE OF AN EVOLUTIONARY ALGORITHM

A. Nezhinsky, J. Kruisselbrink and F. Verbeek
Leiden University, LIACS, Niels Bohrweg 1, The Netherlands

Keywords: Evolutionary algorithm, Model based segmentation, Shape retrieval.

Abstract: There is a need for a high-throughput approach for extracting biological shapes from images. The approach for automated extraction of convex biological shapes presented in this paper is an Evolutionary Algorithm. As opposed to existing model based segmentation methods this approach is uniform for different images, needs no training set and is initialized automatically. The process of finding the shape is considered an optimization problem and for that reason an Evolutionary Algorithm was a good candidate for a solution. The results show that the proposed Evolutionary Algorithm gives a fast solution for pattern recognition and shape extraction.

1 INTRODUCTION

Pattern recognition is crucial for the analysis of large image datasets in bio–imaging and many successful applications have been described in the literature. The first step in many applications is to segment a meaningful biological structure from images and remove the background, despite its position, rotation and scale. Images in biology are acquired under different conditions which affects quality, brightness, contrast, lighting conditions and the amount of noise. Consequently, extraction as well as processing of shapes from the images might be hampered. For our analysis we want to find an uniform way to overcome these problems.

Since many biological shapes (cell, embryo etc.) can be approximated by a convex shape we will propose a method for extraction of convex-shaped structures from images. Our approach is not meant for recognition and discrimination between certain predefined shapes, but for the recognition and the extraction of the most prominent convex shape within the image.

The Evolutionary Algorithm (EA) will extract the object with its shape closest to a convex shape and with the best edge gradient. In most cases this is the biological structure.

Though many model based segmentation techniques (T.F.Cootes et al., 2009), (M.Kass et al., 1987), (F.Leymarie and M.D.Levine, 1993), (Y.N.Wu et al., 2007), (J.A.Sethian and A.James, 1999) are already successfully used, we want to investigate the possibilities of an EA being able to extract convex shapes from images. Thereby we want to look at the advantages of an EA over other approaches and evaluate the difficulties.

Figure 1 shows the general image process flow in which the proposed method and the preprocessing step are incorporated for shape extraction. Edge maps are used as input images, since edges can be used as distinctive features for shapes. An EA is used to extract the convex shapes from these edge maps.

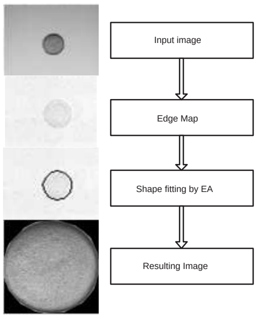


Figure 1: Data flowchart.

The paper is organized as follows. In section 2 existing approaches are discussed. Section 3 gives a short overview of our approach. Section 4 addresses the proposed EA used for shape recognition. In sections 5 experimental results are shown and they are discussed in section 6.

Nezhinsky A., Kruisselbrink J. and Verbeek F. (2010).
CONVEX SHAPE RETRIEVAL FROM EDGE MAPS BY THE USE OF AN EVOLUTIONARY ALGORITHM.
In *Proceedings of the First International Conference on Bioinformatics*, pages 221-225
DOI: 10.5220/0002742502210225

2 EXISTING APPROACHES

Most widely used methods are: active shape models (ASM) (T.F.Cootes et al., 2009); (F.J.Verbeek et al., 2004), active snakes (or contours) (M.Kass et al., 1987) (F.Leymarie and M.D.Levine, 1993), deformable templates (Y.N.Wu et al., 2007) and levelset based methods (LSM) (J.A.Sethian and A.James, 1999).
Most model based segmentation methods are concerned with the retrieval of certain pre–defined shapes from the images.
A classic ASM approach needs a large training set. Instances not matching the training set may fail. In our approach no training set is needed.
With the ASM, deformable templates and the active snake approach the initialization of the shape model must be located close to the shape we are looking for. The LSM approach and some other approaches also suffer from high computational complexity.

3 APPROACH OVERVIEW

We want to develop an approach which is very flexible. While existing approaches like active snakes are very *flexible* models, we want an even more flexible one, where the initialization shape does not need to be set (besides that it is a convex shape) and no training set is needed. An approach that does not require predefined shapes for initialization could be applied to different kind of images without any significant parameter change.

3.1 Image Preprocessing

Edge location within each image is efficiently represented by the magnitude of the gradient, which will be the basis for the shape retrieval. Therefore we transform the input image into a gray scale or a binary representation from which the edge map is computed. Any state of the art edge detector can be used for this purpose.
Thus for each (pixel) position in the image we have: $\mid \bigtriangledown f(x,y) \mid$ as the magnitude of the gradient (R.Gonzales and R.Woods, 2001) on the edge map. We normalize this value between 0 and 1 and will denote the normalized magnitude of gradient as g'. The normalized edge map will be used as the input for our algorithm.

3.2 Structure of Candidate Solutions

The convex shape C is represented as minimal set of points making up the convex hull:

$$C = \{(x_1,y_1),\ldots,(x_n,y_n)\} \qquad (1)$$

The points are represented by xy–coordinates of pixels in the edge map. n denotes the number of points used to approximate the shape.
For C we aim to find a convex shape with the largest edge strength (as high as possible g') in the edge map, cf Figure 2.

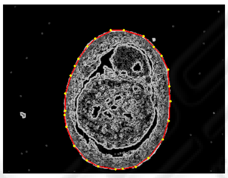


Figure 2: An example edge map of a light microscope image of the embryo of *Emys orbicularis*. C is denoted as a red contour. Yellow dots are the points making up C.

The process of finding C is considered an optimization problem; given a large number of candidate solutions (shapes) we are trying to find the optimal one. For that reason an EA was a good candidate for a solution.
When a solution (a contour) is found, the coordinates of C are transferred to the initial image and the found shape is cut out.

4 AN EA FOR CONVEX SHAPE RECOGNITION

Evolutionary Algorithms (EA) (T.Bäck, 1996) are typically suitable for solving global optimization problems (K.Deb, 2001), in particular when the solution space is very large and no straightforward analytical solution exists or can be found. In an EA, a population of candidate solutions is evolved toward better solutions by introducing computer analogues for recombination, mutation and selection.
The outline of a generic EA reads:

```
t = 0
Initialize P(t)
Evaluate P(t)
while not terminate do
    P′(t) = SelectMates (P(t))
    P′′(t) = Recombine (P′(t))
    P′′′(t) = Mutate (P′′(t))
    P(t + 1) = Select (P(t) \cup P′′′(t))
    t = t + 1
end while
```

Here, t denotes the generation counter, $P(t) = \{A_1, \ldots, A_\mu\}$ is the population [1] of μ candidate solutions (or individuals), and the functions *SelectMates*, *Recombine*, *Mutate*, and *Select* represent the different genetic operators. At the end of each iteration the fitness $\Phi(A(t+1))$ of each individual in the new population is generated and the individuals of $P(t+1)$ are sorted by their fitness.

For the application of the principles of evolutionary computation to the problem at hand one needs a suitable representation for the candidate solutions and a method to determine the quality of the candidate solutions which can serve as a fitness function for the optimization.

4.1 Individual Representation

An *individual* A is represented by a collection of points (cf.3.2). Each point has a value which is the normalized gradient $g'(x,y)$ (cf.3.1). An individual can be represented as a finite set of points on a planar space:

$$A = \{(x_1,y_1), \ldots, (x_n,y_n)\} \tag{2}$$

The convex hull shape is determined by the boundary points of a point collection. We will denote this collection of convex boundary points C_j for individual A_j. The subset of points in A_j construct the convex shape C_j for individual j as shown in Figure 3. Thus $C_j \supseteq A_j$.

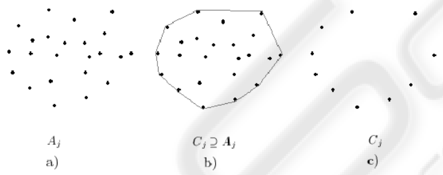


Figure 3: a) The representation of an individual A_j as a collection of points, b) an individual A_j and its convex hull C_j, c) convex boundary points C_j.

4.2 Genetic Operators

For this specific application we only use mutation. For each individual the following applies: every point in the individual is mutated with a probability p_{mut}. The general rule for the approximation of p_{mut} is the formula $p_{mut} = 1/l$, where l is the length of a bit string of properties of an individual; in our case the number of points of A. When mutation is applied to a point (x,y) its coordinates are mutated within some (small) mutation area O_{mut} around the old position of the point. A mutation fails if the mutated pixel is outside the image of if $g'(x,y) = 0$ (there is no gradient at all). Then O_{mut} is enlarged enabling a more global mutation.

4.3 Utility Function

The quality (or fitness) of a candidate solution is composed by the following:

1. i_C is the fraction of all points of A which are present on convex hull C:
$$i_C = \frac{|C|}{|A|} \rightarrow max \tag{3}$$
2. $\bar{g}$ is the average gradient value of all points of A:
$$\bar{g} = \frac{\sum_{i=1}^{i=n} g'(x_i,y_i)}{n} \rightarrow max \tag{4}$$
3. b_C is the normalized average gradient value of points of the convex hull C (of individual A). We calculate the average gradient value of the image between these points. We perform a 8-connected Bresenham-style walk (J.D.Foley et al., 1990) between all subsequent points in C. Let the function $B((x_i,y_i),(x_j,y_j))$ denote the average gradient value taken over the euclidean distance between consecutive points (x_i,y_i) and (x_j,y_j):
$$b_C = \frac{\sum_{i=0}^{i=i_C-1} B((x_i,y_i),(x_j,y_j))}{i_C} \rightarrow max \tag{5}$$
4. σ_C is the standard deviation of the convex hull border segment length, normalized by the use of average side distance. Let $d((x_i,y_i),(x_j,y_j))$ denote the euclidean distance between consecutive points (x_i,y_i) and (x_j,y_j), d_{avg} is the average distance of all these points.
$$\sigma_C = e^{(-1.0/d_{avg})\cdot\sqrt{\frac{\sum_{i=0}^{i=i_C-1} d((x_i,y_i),(x_{i+1},y_{i+1}))-d_{avg}}{i_C}}} \rightarrow max \tag{6}$$

Since parameters $(i_C, \bar{g}, b_C, \sigma_C)$ need be optimized simultaneously we are looking at a multi-objective optimization problem. The objectives can be conflicting, for example, increase of the points on convex hull can lead to lower gradient values. In our approach we aggregate the multiple objective functions into an univariate problem by making use of the concept of Desirability Functions (DF) and Desirability Index (DI) (D.Steuer, 1999). We also make use of a simple DF where for each parameter i we make use of a power function $f(i) = i^a$ (all parameters are in a normalized representation, it remains in the interval $[0,1]$ for any exponent a. To combine the DF's into one overall quality value we use a aggregation of all the parameters. The following product fitness function is parametrized as follows:

$$\Phi(\vec{a}_j(t)) = i_C{}^{a_0} \cdot \bar{g}^{a_1} \cdot b_C{}^{a_2} \cdot \sigma_C{}^{a_3} \rightarrow max \tag{7}$$

[1] We use notation P, not for probability, but for the population in an EA as is done in (T.Bäck, 1996).

4.4 Implementation

The number of points m in each individual can be set depending on the precision of the prediction we want to retrieve. Obviously a lower m gives less accurate shape, while a higher m gives a more precise shape but is computationaly more expensive.
The individuals are initialized with:

$$
\begin{aligned}
x_{a_j} &= U(0, ImageX), j = (1, \ldots, n) \\
y_{a_j} &= U(0, ImageY), j = (1, \ldots, n) \qquad (8)
\end{aligned}
$$

The convex shape C is calculated using the Graham scan (R.L.Graham, 1972) algorithm with a time complexity of $O(m)$ (m is the number of points). Since the algorithm needs a sorted list the points are sorted on the x-coordinate by *insertion sort*.
Parent selection is repeated for each individual in the new population (μ times). For the selection, tournament size $q = 5$ is chosen as one of the common settings. A larger q results in less selective pressure. No crossover is used, as it did not yield better results. As a mutation rate of $1/m$ is most commonly used in EA, the mutation rate for this implementation is chosen accordingly, hence $p_{mut} = 1/m$.
By means of empirical testing we found the following settings fit for our problem: $a_0 = a_1 = 1$, $a_2 = 2$, $a_3 = 0.5$.

5 EXPERIMENTS AND RESULTS

For all tests we use a standard gradient operator – the *Sobel* (R.Gonzales and R.Woods, 2001) edge detector for the edge map creation.

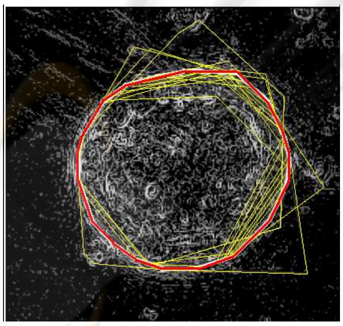


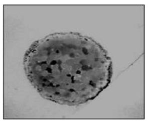
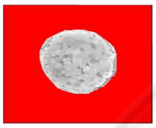
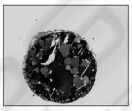
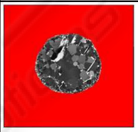
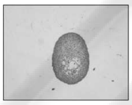
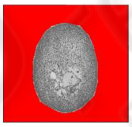
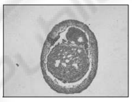
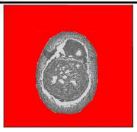
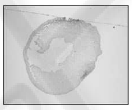
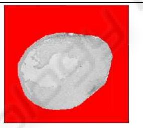
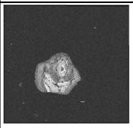
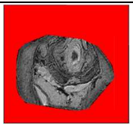
Figure 4: Snapshot of algorithm in search of best solution.

The process of finding the best solution is shown in Figure 4. Thin yellow lines represent candidate solutions. The bright red line represents the best solution (with highest Φ).

5.1 Results: Finding a Convex Hull

Images from various fields in biology, containing convex shapes were used for this test. The results are shown in Table 1. As can be seen the shapes that might be of interest are accurately retrieved. On a*Intel(R) Pentium(R) 4 CPU 3.2GHz* this took on average 3 seconds.

Table 1: Some results of our algorithm applied to images from different fields in biology after running for 70 generations and 2800 function–evaluations.

Input	output	input	output
			
			
			

5.2 Results: Finding a Convex Hull within a Convex Hull

In this section we will show that it is also possible to retrieve a convex shape from another convex shape.
We have used and tested the approach for our particular data - images of zebrafish in the early development. The images were acquired using Leica MZ16FA light microscope in 24-bit color at a resolution of 2592 × 1944 pixels. Time–lapse acquisition is accomplished in automated fashion over time $T(5 - 10h)$.

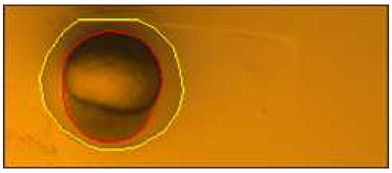


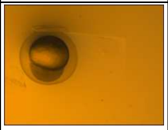
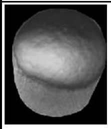
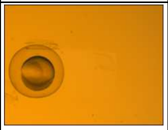
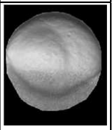
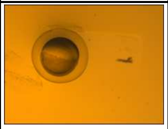
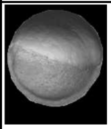
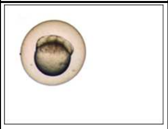
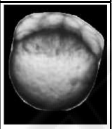
Figure 5: Zebrafish embryo in early stage.

As can be seen in Figure 5 the shape of zebrafish embryos in early stages can be described as a deformed circle being the embryo (red shape in Figure 5), encapsulated in a larger (deformed) circle being the membrane (yellow shape in Figure 5).

In our case to extract the embryo shape from the image we run the convex hull extraction algorithm twice. First are looking for the outer convex shape. The moment the approximation of the outer shape (the membrane) is complexed we try to find the inner shape (the embryo), which is then the convex shape with the largest edge strength within the membrane shape.

We have run the algorithm on a set of 600 images of the zebrafish embryo. A few results shown in Table 2. As can be seen the embryo shapes were retrieved accurately. The algorithm failed in cases where the edge map was very weak (out of focus), which was 2% of our test set. On an *Intel(R) Pentium(R) 4 CPU 3.2GHz* this took up to 10 seconds.

Table 2: Some results of our algorithm applied to images of zebrafish embryos after running for 300 generations and 24000 function–evaluations.

Input	output	Input	images
			
			

6 DISCUSSION AND CONCLUSION

In this paper, we have described the use of the EA for the recognition of convex shaped objects and tested it for different biological images.

We have demonstrated the application of the EA approach to different images and the possibility to retrieve both single convex shape (cf 5.1) as the possibility to retrieve a convex shape from another convex shape (cf 5.2).

In our approach the shape is not predefined like in most models based segmentation methods, but can be any convex shape within an image (assuming there is one convex shape in the image, we are looking for). Also, our approach does not need a training set like ASM does and can be used without significant changes for different applications within bio–imaging.

The results are promising and our future direction is to apply this strategy in high-throughput applications for screening and compare it against other methods. This form of fast automated image analysis is indispensable for such approaches.

ACKNOWLEDGEMENTS

This work was supported by the Smartmix program of the Netherlands Ministry of Economic Affairs. Special thanks to G. Lamers for assistance with image acquisition and L. Bertens for providing images.

REFERENCES

D.Steuer (1999). Multi-criteria-optimization and desirability indices. *Technical Report 20/99, University of Dortmund, Statistics Department,*.

F.J.Verbeek, D.D.Rodrigues, H.Spaink, and A.Siebes (2004). Data submission of 3d image sets to a bio-molecular database using active shape models and a 3d reference model for projection. *Proceedings SPIE /5304/, Internet Imaging V. 13-23*.

F.Leymarie and M.D.Levine (1993). Tracking deformable objects in the plane using an active contour model. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 15, No. 6*.

J.A.Sethian and A.James (1999). Level set methods and fast marching methods : Evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. *Cambridge University Press*.

J.D.Foley, Dam, A., S.K.Feiner, and J.F.Hughes (1990). *Computer Graphics: Principles and Practice*. Addison-Wesley.

K.Deb (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley Interscience Series in systems and Optimization.

M.Kass, A.Witkin, and D.Terzopoulos (1987). Snakes: Active contour models. *International Journal of Computer Vision. Vol. 1, No. 4*.

R.Gonzales and R.Woods (2001). *Digital Image Processing*. Addison-Wesley, London, 2nd edition.

R.L.Graham (1972). An efficient algorithm for determining the convex hull of a finite. *Information Processing Letters 1*.

T.Bäck (1996). *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford Univ. press.

T.F.Cootes, C.J.Taylor, D.H.Cooper, and J.Graham (2009). Active shape models - their training and application. *Computer Vision and Image Understanding, Vol. 61, No. 1*.

Y.N.Wu, Z.Si, C.Fleming, and Zhu, S. (2007). Deformable template. *IEEE International Conference on Computer Vision*.