DETECTING CONTOUR SALIENCES USING TENSOR SCALE

Fernanda A. Andaló, Paulo A. V. Miranda, Ricardo da S. Torres, and Alexandre X. Falcão

Institute of Computing – State University of Campinas (UNICAMP)
CEP 13084-851, Campinas, SP, Brazil

ABSTRACT

Tensor Scale is a morphometric parameter that unifies the representation of local structure thickness, orientation, and anisotropy, which can be used in several image processing tasks. This paper introduces a new application for tensor scale, which is the detection of saliences on a given contour, based on the tensor scale orientations computed for the entire object and mapped to its contour. For validation purposes, we present a shape descriptor that uses the detected contour saliences. Experimental results are provided, comparing the proposed method with our previous Contour Salience Descriptor (CS). We show that the proposed method can be not only faster and more robust in the detection of salience points than the CS method, but also more effective as a shape descriptor.

***Index Terms*—** Image processing, Image shape analysis, Information retrieval

1. INTRODUCTION

The saliences of a shape are defined as the higher curvature points along the shape contour [1], or vertex points along the contour with first derivative discontinuity [2]. Their detection is the key to various applications in image processing (e.g., image registration, polygonal approximation, motion analysis, and shape description [3]).

A salience detector should satisfy import criteria [4], such as: all true saliences should be detected; no false saliences should be detected; salience points should be well localized; robustness with respect to noise (e.g., rounded corners or peaks on the object's contour); and efficient computation.

In this paper, we extend the application of tensor scale for salience detection on an object's contour obtained from a binary image. Tensor Scale [5] is a morphometric parameter yielding a unified representation of local structure thickness, orientation, and anisotropy. That is, at any image point, its tensor scale is represented by the largest ellipse centered at that point and within a homogeneous region.

Other methods for salience detection based on derivatives present instability problems due to points with infinity or very large curvature. Among the existing solutions, we selected the Contour Salience (CS) [3] method for comparison, because of its interesting previous results. We show that our method can be not only faster and more robust in the detection of salience points, but also more effective as a shape descriptor, using the same experiment set used for the CS evaluation.

2. BACKGROUND

In [5], Punam et al. introduced a local scale method – Tensor Scale – represented by the largest ellipse within a homogeneous region and centered at a given pixel p. This method defines the ellipse uniquely by three factors: orientation (angle of the major axis with the horizontal axis), anisotropy (a relation between the major and the minor axes), and thickness (length of the minor axis).

A tensor scale ellipse is calculated from sample lines that are traced around a given pixel, from 0 to 180 degrees (Figure 1(a)). The axes of the ellipse are determined by computing the intensities along each of the sample lines and the location of two optimum edge points on these lines (Figure 1(b)). The next step consists of repositioning the edge locations to points equidistant to the given pixel, following the axial symmetry of the ellipse (Figure 1(c)). The computation of the best-fit ellipse to the repositioned edge locations is done by Principal Component Analysis (PCA) (Figure 1(d)).

These computations are performed for every pixel of the image. A critical drawback is that the algorithm proposed in [5] is computationally expensive and quite prohibitive for some image processing tasks. For this reason, Miranda et al. [6] proposed an efficient implementation of the original method, which differs in the following aspects.

The first change is in the edge location phase. The adopted approach is to go along each pair of opposite segments, alternately at the same time, instead of going along one entire segment. By doing this, the reposition phase is no longer necessary. The second change is the use of two connected thresholds to simplify the method of detecting edges. The third and final change is the improvement of the ellipse computation phase. They proposed a function that gives the angle of the ellipse directly, instead of using PCA.

Work supported by Microsoft Escience and Tablet PC Technology and Higher Education projects, CNPq, FAPESP, CAPES, and FAEPEX.

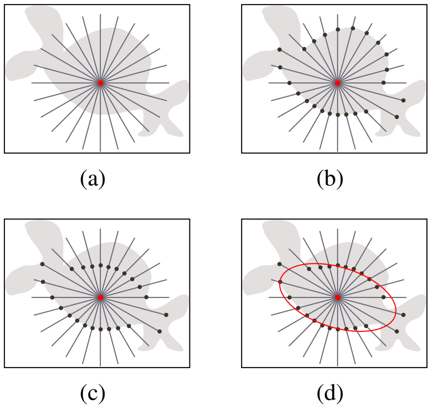


Fig. 1. Original method for Tensor Scale Computation.

3. TENSOR SCALE COMPUTATION AND CONTOUR MAPPING

The proposed method begins with the tensor scale computation for all pixels inside an object, using the algorithm proposed by Miranda et al. [6], summarized in the previous section. Next, it uses the Euclidean Distance Transform (EDT) (Section 3.1) to map the tensor scale orientations onto the object's contour (Section 3.2).

3.1. Euclidean Distance Transform

The Euclidean Distance Transform (EDT) is calculated using the Image Foresting Transform (IFT) [7]. For a given object O, the IFT-Euclidean Distance Transform [3] (IFT-EDT) computes at the same time, for every object's pixel, its closest pixel on the object's contour S and the squared Euclidean distance between them. S is a set of contour pixels following a given order along the contour of O. The first information resulted by the IFT-EDT is stored into a root map R while the second is stored in a cost map C. The root map is used in the next step, to map the orientation of the ellipses onto the contour S.

3.2. Tensor Scale Contour Mapping

After computing the IFT-EDT, each contour pixel s in S is root of a region (influence zone) formed by pixels in O which are closest to s than to any other root in S. Every pixel q in this region will have $R(q) = s$. The idea is to map to s the orientation of the ellipse with highest anisotropy in its influence zone (Algorithm 1).

Algorithm 1 outputs two vectors ($MapOri$ and $MapAni$) that are updated so that $MapOri(s)$ and $MapAni(s)$, for all pixels $s \in S$, contain the orientation of the ellipse with the highest anisotropy in the influence zone of s and the value of this anisotropy, respectively. Contour

Algorithm 1 Mapping orientations to the object's contour

Input: An binary image I with a single object O, a set S of contour pixels of O, the root map R resulting from IFT-EDT, and $Anisotropy$ and $Orientation$ vectors that contain the tensor scale anisotropy and orientation, respectively, computed for all pixels in O (Section 2, algorithm by Miranda et al. [6]).

Output: $MapAni$ and $MapOri$ vectors.

for all pixel $p \in S$ **do**
    $MapOri(p) \leftarrow 0$;
    $MapAni(p) \leftarrow 0$;
end for
for all pixel $p \in$ object O **do**
    if $MapAni(R(p)) < Anisotropy(p)$ **then**
        $MapAni(R(p)) \leftarrow Anisotropy(p)$;
        $MapOri(R(p)) \leftarrow Orientation(p)$;
    end if
end for

points with no influence zone borrow the orientations of the neighbors.

4. SALIENCES DETECTION

In order to locate the salience points, we calculate the differences between adjacent mapped orientations in S. This is possible because high curvature points cause abrupt change of orientation along the contour.

The difference value at pixel $p \in S$ is $Difference(p) = AngularDistance(MapOri(p-1), MapOri(p+1))$, where the function $AngularDistance(\alpha, \beta)$ gives the smallest angle between the orientations α and β.

Now, we can use a threshold value to eliminate low values of difference along the contour. Figure 2 shows the detected saliences (dots) using threshold $16°$, i.e, saliences related to angle differences lower than $16°$ were not represented.

5. SHAPE DESCRIPTION BY SALIENCES

Corners and high curvature points concentrate more information than other points of the shape [8]. For this reason, it is intuitive to conceive that curvature is an important key for the identification of many geometric aspects. Based on this, we use the saliences as key points for shape description. After the salience detection phase, we need to determine the salience value of the points. It is known that the influence zones of salience points are greater than the influence zones of other points along the contour and the influence area of a convex point is greater outside the contour than inside, and vice-versa for concave points [3].

For each detected salience, the salience value are estimated using the influence zone of the pixels in S, which is

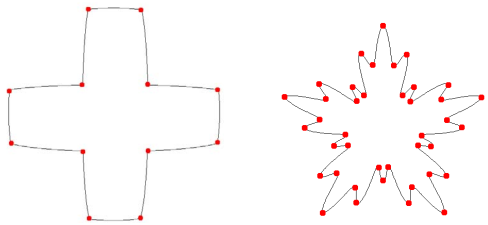


Fig. 2. Visualization of salience points.

computed from the root map R generated by IFT-EDT. Negative values are used for concave points and positive values for convex points.

The descriptor is formed by the same feature extraction and metric functions used in CS [3]. The only difference between our approach and the original CS descriptor is the quality of the salience points detected along the contour.

The feature vector extraction works as follows. One arbitrary salience point is taken as reference and the method computes the relative position of each salience point with respect to the reference point, following the order of points in S. The vector is composed by the salience values and the their relative positions.

There are two drawbacks in this kind of method that have to be considered for matching two feature vectors: the reference point may not be the same for different vectors and feature vector of distinct objects may differ in size. Therefore, the matching is a heuristic algorithm that registers the vectors using the reference points and computes their similarity taking into account the different sizes. This algorithm is based on the matching algorithm described in [3].

6. EXPERIMENTAL RESULTS

For the experiments, two databases (Fish-shape and the MPEG-7 Part B) were used in part or entirely.

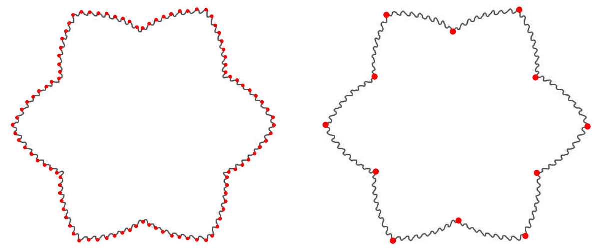
The Fish-shape[1] database consists of 1100 fish shapes. The classes were formed by ten variations of each original image with rotation and scaling, resulting in 1100 classes with 10 images each one.

The MPEG-7[2] Part B database consists of 1400 shapes divided in 70 classes of various shapes (20 images in each class).

We evaluated our method with respect to two aspects. First, the quality of the estimated salience points (Section 6.1). Second, the impact of a better estimation in the results of the CS descriptor (Section 6.2).

The CS approach of detecting saliences (skeleton-based approach) begins with the calculation of multiscale internal and external skeletons by label propagation. Then, the

[1] http://www.ee.surrey.ac.uk/research/vssp/imagedb/demo.html
[2] http://www.chiariglione.org/mpeg/



(a) TS. (b) Skeleton.

Fig. 3. Saliences granularity for the TS- and Skeleton-based approaches.

saliences are detected by matching each salience point of the internal skeleton to one convex point of the contour and each salience point of the external skeleton to one concave point of the contour.

6.1. Salience Detection

The first consideration is concerned with performance issues. The Tensor Scale based approach (TS-based approach) was twice faster (speedup of 2.04), on average, than the skeleton-based approach used in CS [3], when executed for the entire Fish database. Experiments considered that the methods were executed on a AMD 64 3000+ Processor, with 1GB of RAM memory.

The second consideration is that the TS-based method is computed locally, looking for each mapped orientation and for its neighbors along the contour. The skeleton-based method is more global, because it uses the internal and external skeletons of the whole shape for salience detection. This difference in granularity also makes the detection of saliences less robust in the skeleton-based approach, because the multiscale skeletons have to be thresholded to obtain salience points. This threshold represents a smoothing of the contour and, consequently, loss of some important saliences. In order to detect these saliences, we would have to reduce the threshold. The TS-based method is also dependent of a threshold, but it is much easier to fix a single threshold for the entire database, which is the case of TS-based approach, than to find the best threshold for every single image in the database, which is the case of the skeleton-based approach.

Figure 3 shows the differences between the two methods. While the skeleton-based approach has higher granularity, detecting only the twelve global saliences presented in the object, the TS-based approach detects more abrupt differences of orientation that exist on the contour.

For effectiveness comparison, we constructed a database consisting of 42 shapes of the Fish-shape database and 112 shapes of the MPEG-7 Part B database, resulting in 2835

Measures	Skeleton	TS 10	TS 12	TS 14	TS 16	TS 18
Recall	0.956	0.964	0.964	0.963	**0.963**	0.962
Precision	0.903	0.889	0.923	0.946	**0.963**	0.968
Accuracy	0.718	0.840	0.862	0.874	**0.875**	0.867

Table 1. Effectiveness measures for skeleton- and TS-based approaches.

saliences. The images were chosen by taking into account the obviousness of the contour salience points location, in order to not favor any method. Then, a set of ground truth images were constructed with the location of the salience points.

This experiment relies on counting the true positive saliences (T_+) and false positive saliences (F_+) for the ground truth images, using both methods. After this counting, three effectiveness measures were calculated: recall, precision, and accuracy. Recall (Rec) and precision (Prec) are computed as $Rec = \frac{T_+}{T_+ + T_-}$ and $Prec = \frac{T_+}{T_+ + F_+}$, where T_- is the number of true negatives, and ($T_+ + T_-$) represents the total number of points. The accuracy (Acc) is calculated as $Acc = \frac{T_+ + T_-}{T_+ + T_- + F_+ + F_-}$, where F_- (false negatives) represents the number of miss-detections.

The results with different threshold values for TS-based approach are presented in Table 1. The threshold for the skeleton-based approach was fixed in 5%, that is the value recommended in [3] for this database.

The TS-based method has better accuracy than the skeleton-based, for all tested thresholds. In the TS-based method, the accuracy was maximized with threshold value 16 and this is the value adopted for this method in the experiments described in the next section.

6.2. Shape Descriptor

A good effectiveness measure should capture the concept of separability. Separability indicates the discriminatory ability between objects that belong to distinct classes. This concept was introduced for CBIR in [3].

Both descriptors were computed for Fish-shape database and the separability curves for the two evaluated descriptors are showed in Figure 4.

The TS and CS approaches have equivalent performance for search radii less than 25% of their maximum distance. From this point to 65%, the TS is more robust and effective then CS. By analyzing Figure 4, we observe that TS is more effective or equal to CS in 80% of the search radii.

7. CONCLUSIONS AND FUTURE WORK

This paper introduces a salience detector based on Tensor Scale. For this purpose, it uses the differences between adjacent tensor scale orientations mapped onto the object's contour. The experimental results showed that this method is faster and more robust than saliences detection proposed in [3].

We also proposed a new version of CS descriptor, using our approach to detect saliences. The experiments indicate that the new approach is more effective than CS up to 80% of the search radii, according to multiscale separability measure.

We are currently comparing the proposed method with other shape descriptors and using more image databases.

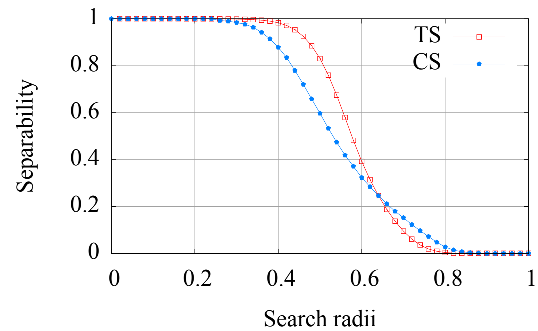


Fig. 4. Multiscale separability curve for Fish database.

8. REFERENCES

[1] R. da S. Torres, A. X. Falcão, and L. da F. Costa, "A graph-based approach for multiscale shape analysis," *PR*, vol. 37, no. 6, pp. 1163–1174, 2004.

[2] L. da F. Costa, A. G. Campos, and E. T. M. Manoel, "An integrated approach to shape analysis: results and perspectives," in *Proc. of the QCAV2001*, 2001, vol. 1, pp. 23–24.

[3] R. da S. Torres and A. X. Falcão, "Contour salience descriptors for effective image retrieval and analysis," *IVC*, vol. 25, no. 1, pp. 3–13, 2007.

[4] F. Mokhtarian and R. Suomela, "Robust image corner detection through curvature scale space," *PAMI*, vol. 20, no. 12, pp. 1376–1381, 1998.

[5] P. K. Saha, "Tensor Scale: A local morphometric parameter with applications to computer vision and image processing," in *CVIU*, 2005, vol. 99, pp. 384–413.

[6] P. A. V. Miranda, R. da S. Torres, and A. X. Falcão, "TSD: A shape descriptor based on a distribution of tensor scale local orientation," in *Proc. of the SIBGRAPI 2005*, 2005, pp. 139–146.

[7] A. X. Falcão, J. Stolfi, and R. A. Lotufo, "The image foresting transform: theory, algorithms, and applications," *PAMI*, vol. 26, no. 1, pp. 19–29, 2004.

[8] F. Attneave, "Some informational aspects of visual perception," *Psych. Rev.*, vol. 61, no. 3, pp. 183–193, 1954.