3D Texton Spaces for color-texture retrieval

Susana Alvarez, Anna Salvatella, Maria Vanrell, and Xavier Otazu

Universitat Rovira i Virgili, Department of Computer Science & Mathematics,
Campus Sescelades, Avinguda dels Països Catalans, 26, 43007 Tarragona, Spain,
susana.alvarez@urv.es
Computer Vision Center,
Edifici O, Campus UAB, 08193 Bellaterra, Barcelona, Spain,
{anna.salvatella,maria.vanrell,xavier.otazu}@uab.es

Abstract. *Color and texture are visual cues of different nature, their integration in an useful visual descriptor is not an easy problem. One way to combine both features is to compute spatial texture descriptors independently on each color channel. Another way is to do the integration at the descriptor level. In this case the problem of normalizing both cues arises. In this paper we solve the latest problem by fusing color and texture through distances in texton spaces. Textons are the attributes of image blobs and they are responsible for texture discrimination as defined in Julesz's Texton theory. We describe them in two low-dimensional and uniform spaces, namely, shape and color. The dissimilarity between color texture images is computed by combining the distances in these two spaces. Following this approach, we propose our TCD descriptor which outperforms current state of art methods in the two different approaches mentioned above, early combination with LBP and late combination with MPEG-7. This is done on an image retrieval experiment over a highly diverse texture dataset from Corel.*

Key words: color-texture descriptors, retrieval, Corel dataset

1 Introduction

In the literature there are several works dealing with color and texture in different applications, however the integration of both features is still an open problem [6]. The different nature of these two visual cues has been studied from different points of view. While texture is essentially a spatial property, color has usually been studied as a property of a point.

Computational approaches have proposed several algorithms to integrate these features. Some approaches [2, 3] process texture and color separately, using different descriptors, they combine both descriptions at a similarity measure level afterwards. This means that for every visual cue a dissimilarity measure is obtained, each one in a different space and then they are combined to obtain a final similarity that needs to be scaled in order to be comparable. Other approaches [6, 11, 12] use the same descriptor over the three components on the

chosen color space. The final descriptor is composed by the concatenation of the three feature vectors obtained separately from each color channel.

In this paper we propose a perceptual approach to combine color and texture in order to define a compact color-texture descriptor. Our combination is based on two low-dimensional spaces that describe color textures through the texton concept. Here we use the original definition of texton given by Julesz in his Texton Theory [4]. Textons are defined as the attributes of image blobs. The differences of their first order statistics are the responsibles for texture discrimination. We use two different spaces, one to represent shape textons and a second one to represent color textons. In this way we obtain a combination of cues directly from the attributes of the blobs.

The paper is organized as follows: in section 2 we review the perceptual considerations justifying the attribute spaces and describe the computational method to obtain the important blobs of an image. In section 3 we describe the two texton spaces where the descriptor, *Texture Component Descriptor* (TCD), we propose is derived from the fusion of similarities computed in each of these spaces. Section 4 contains the experiment that evaluates our approach, showing that our descriptor achieves better performance than current descriptors in retrieval. We compare our TCD with MPEG-7 and LBP descriptors in standard Corel datasets. In the last section we sum up our proposal of a perceptual integration of color and texture descriptors.

2 Texture and Blobs

Texton theory [4] was originally introduced as the basis for the first steps in texture perception. This theory states that preattentive vision directs attentive vision to the location where differences in density of textons occur, ignoring positional relationships between textons and defines the concept of textons as the attributes of elongated blobs, terminators and crossings. From several psychophysical experiments they conclude that preattentive texture discrimination is achieved by differences in first-order statistics of textons, which are defined as line-segments, blobs, crossings or terminators; and their attributes, width, length, orientation and color.

Inspired by this idea, we consider hereby that a texture can be defined as a set of blobs, but we will not consider terminators or crossings, since it is not clear whether they would be necessary for natural images. Therefore, we propose a texture descriptor based on the attributes of the image blobs or *perceptual blobs*.

Thus, following the assumption that a texture can be described by their blobs then we are also assuming that a texture is provided by the existence of groups of similar blobs. This is the basis of the repetitiveness nature of texture images. Some examples of this proposal can be seen in Fig. 1. In image (a) a striped texture is described by two different types of blobs: blue elongated blobs and grey elongated blobs. In the same figure, texture (b) can be described in terms of 6 different types of blobs, which are blue, green and orange, of different sizes and shapes. The groups of blobs sharing similar features (size, orientation and

color) are called *texture components* (TC). This description of a textured image in terms of the attributes of blobs or textons is the basis of our descriptor.

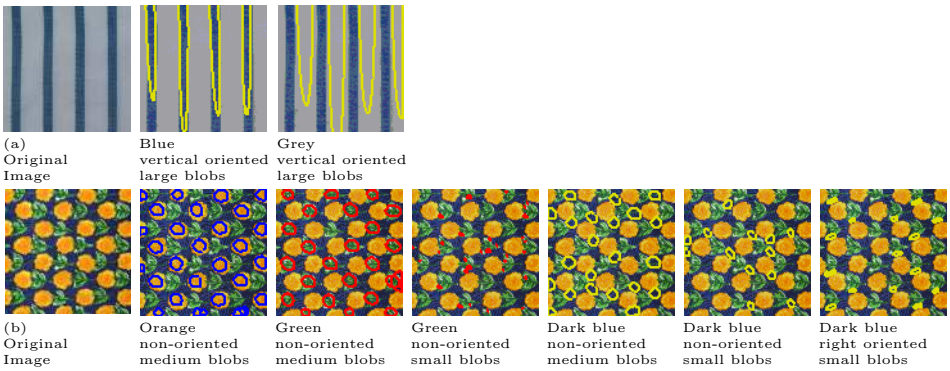


Fig. 1. Texture components and their description

2.1 Blob detection

To obtain the attributes of the image blobs we use the differential operators in the scale-space representation proposed in [5]. We use the normalised differential Laplacian of Gaussian operator to detect the blobs of the image ($\nabla^2_{norm} L_\sigma$). This operator also allows us to obtain the scale and the location of the blobs. The aspect-ratio and orientation of non-isotropic blobs are obtained from the eigenvectors and eigenvalues of the windowed second moment matrix [5].

Since blob information emerge from both intensity and chromaticity variations, this procedure is applied to each component in the opponent color space in order to obtain the blobs of a color image. Previously, all the components were normalized to be invariant to intensity changes and then a perceptual filtering was carried out. This perceptual filtering is performed with a winner-take-all mechanism that selects the blobs of higher response of $\nabla^2_{norm} L_\sigma$ from those that overlap in different channels. This last step provides us with a list of *perceptual blobs* and their attributes, that we refer as Blob Components (BC), which are given in matrix form as:

$$\mathbf{B} = [\mathbf{B_{sha}} \mathbf{B_{col}}] \tag{1}$$

where $\mathbf{B}$ is formed by joining two matrices: $\mathbf{B_{sha}}$ that contains blob shape attributes and $\mathbf{B_{col}}$ contains blob color attributes. These matrices can be defined as:

$$\mathbf{B_{sha}} = [\mathbf{WL\Theta}],\ \mathbf{B_{col}} = [\overline{\mathbf{I}}\,\overline{\mathbf{RG}}\,\overline{\mathbf{BY}}] \tag{2}$$

where $\mathbf{W^T} = [w_1 \dots w_n]$, $\mathbf{L^T} = [l_1 \dots l_n]$, $\mathbf{\Theta^T} = [\theta_1 \dots \theta_n]$ being (w_j, l_j, θ_j) shape attributes of the j-th blob (width, length and the orientation respectively), and

$\overline{\mathbf{I}}^{\mathbf{T}} = [\bar{i}_1 \dots \bar{i}_n]$, $\overline{\mathbf{RG}}^{\mathbf{T}} = [\overline{rg}_1 \dots \overline{rg}_n]$, $\overline{\mathbf{BY}}^{\mathbf{T}} = [\overline{by}_1 \dots \overline{by}_n]$ being $(\bar{i}_j, \overline{rg}_j, \overline{by}_j)$ color attributes of the j-th blob (median of the intensity and chromaticities of the pixels forming the winner blob respectively).

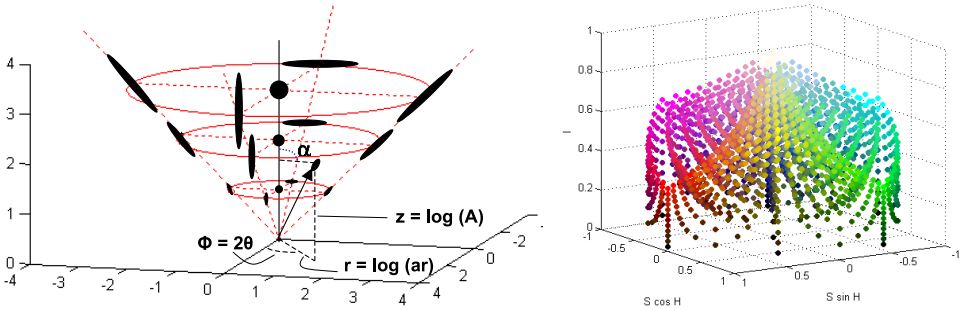
3 Texton spaces

At this point we have to deal with the problem of the different nature of the attributes we have computed for the Blob Components that is given by $\mathbf{B}$. We will use two different texton spaces to represent the two sets of attributes, $\mathbf{B_{sha}}$ and $\mathbf{B_{col}}$. The first one is the shape texton space and the second one is the color texton space. Both need to be perceptual spaces since the fusion of color and texture is done through the Euclidean distances in these two spaces separately.

The uniform space used to represent shape is a three dimensional cylindrical space where two axes represent the shape of the blob (aspect-ratio and area) and the third axis represents its orientation. The space we have used is shown in Fig. 2.(a). This perceptual shape space is obtained by performing a non linear transformation U,

$$\begin{array}{rccc} U: & \mathbb{R}^3 & \rightarrow & \mathbb{R}^3 \\ & (w,l,\theta) & \rightarrow & (r,z,\phi) \end{array} \tag{3}$$

where $r = \log(ar), z = \log(A)$ and $\phi = 2\theta$, being ar the blob aspect ratio ($ar = w/l$), A its area ($area = w \cdot l$) and θ its orientation.



(a) (b)

Fig. 2. (a) Shape Texton Space in cylindrical coordinates. (b) Color Texton space (HSI).

To represent the color attributes of blobs we use the HSI color space corresponding to the transform given in [1]. This space is shown in Fig. 2.(b). Although this color space is not perceptually uniform, our choice is based on the fact that is close to an uniform space when we need to represent non-calibrated color.

Following our initial assumption that a texture is provided by the existence of groups of similar blobs (*Texture Components*), in the next section we propose a color-texture descriptor based on clustering blob attributes in these two texton spaces.

3.1 Texture Component descriptor (TCD)

Considering the properties of the texton spaces we can state that similar blobs are placed on different unidimensional varieties such as lines, rings or arcs. To group blobs of similar shapes and colors we use a clustering method that groups data with these points distributions and, at the same time, makes it possible to combine spaces with different characteristics, specifically color and shape. The clustering algorithm which has these properties is the Normalized Cut (N-cut) [9], that obtains the clusters by partitioning a graph. In the graph the nodes are the points of the feature space and the edges between the nodes have a weight equal to the similarity between nodes. To determine the similarity between nodes we need to define a distance. Since the shape space has been designed to be uniform and the HSI color space is almost uniform, it is reasonable to use the Euclidean distance.

The N-Cut clustering algorithm can be defined as

$$NCUT([U(\mathbf{B_{sha}}), HSI(\mathbf{B_{col}})], \mathbf{\Omega}) = \{\hat{\mathbf{B}}^{\mathbf{1}}, \hat{\mathbf{B}}^{\mathbf{2}}, \ldots, \hat{\mathbf{B}}^{\mathbf{k}}\} \tag{4}$$

where, $\mathbf{\Omega}$ is the weight matrix, and its elements define the similarity between two nodes through the calculation of the distance in each one of the texton spaces (shape and color) in an independent way. These weights are defined as,

$$\omega_{pq} = e^{-\frac{\|U(\mathbf{B_{sha}})_p - U(\mathbf{B_{sha}})_q\|_2^2}{\sigma_{sha}^2}} \cdot e^{-\frac{\|HSI(\mathbf{B_{col}})_p - HSI(\mathbf{B_{col}})_q\|_2^2}{\sigma_{col}^2}} \tag{5}$$

This weight represents the similarity between blob p and blob q that depends on the similarity of its shape features and the similarity of its color features. $U(\mathbf{B_{sha}})_p$ and $HSI(\mathbf{B_{col}})_p$ are the p-th row of the matrices $U(\mathbf{B_{sha}})$ and $HSI(\mathbf{B_{col}})$ respectively. As in [9], σ_{sha} and σ_{col} are defined as a percentage of the total range of each feature distance function, the first one in the shape space and the second one in the color space.

The result of the clustering obtained by the N-cut algorithm is represented by $\hat{\mathbf{B}}^{\mathbf{i}}$, $\forall i = 1, ..., k$ (where k is the total number of clusters). The prototype of each cluster i becomes our *Texture Component Descriptor* (TCD^i). This is computed by estimating the median of all the blob attributes in the i cluster, $[\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{sha}} \hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{col}}]$. This give a 6-dimensional description for each cluster or Texture Component (TC):

$$TCD^i = (r^i, z^i, \phi^i, h^i, s^i, i^i) \tag{6}$$

In this way the descriptors of an image are the shape (3D) and color attributes (3D) of its TC. In figure 3 we show the over all scheme to obtain the TCD.

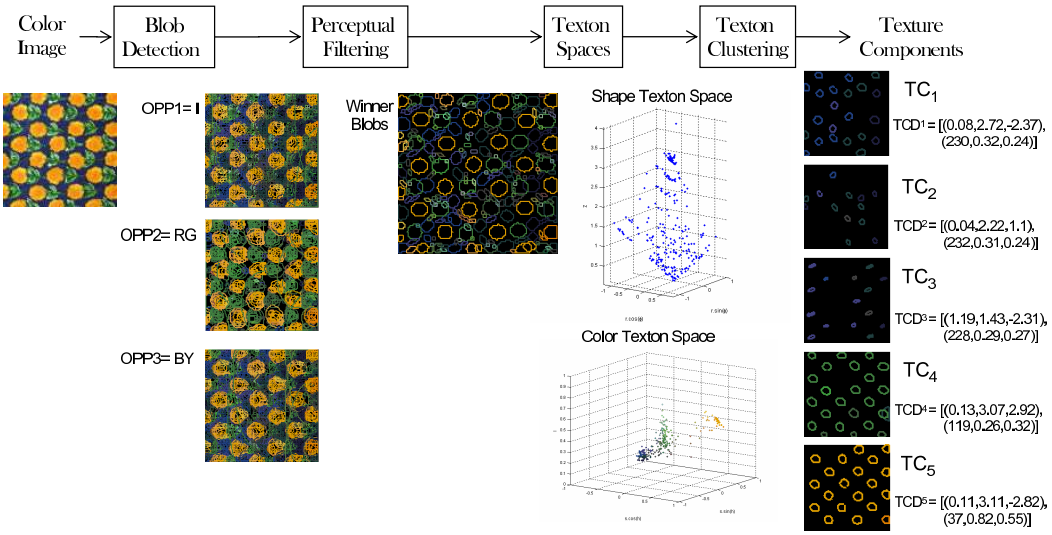


Fig. 3. Stages of TCD Computation.

4 Experiment

This experiment evaluates the performance of our TCD descriptor in an image retrieval application. In order to compute the similarity between two textures we need to define an adequate measure which considers that the TCD of images can have different number of texture components. For a given image, the number of texture components in its TCD depends on the complexity of the texture content. A metric presenting this property is the Earth Mover's Distance [8]. In our case this distance adapts perfectly because our feature spaces are bounded independently of the image content. Shape space has the limits of blob attributes and color space is bounded by the maximum luminance. Therefore we define the ground distance between two TCD and the weighting parameters as

$$d(TCD^i, TCD^j) = \alpha \cdot d_{shape}(TCD^i, TCD^j) + \beta \cdot d_{color}(TCD^i, TCD^j) \quad (7)$$

where d_{shape} and d_{colour} are Euclidean distances in the shape space and color space, respectively. The shape space has been built taking into account perceptual considerations allowing it to be considered as a uniform space, therefore distances are correctly estimated. This is not the case in the HSI color space that is not real uniform space, therefore the distances are not accurate. The parameters α and β are the weights of these two distances.

To perform this experiment we have used three different datasets, these are Texture images from the Corel stock photography collection[1]: Textures (137000), Various Textures I (593000) and Textures II (404000). In the experiment we

[1] Corel data are distributed through http://www.emsps.com/photocd/corelcds.htm

refer to them as *Corel*, *Corel1* and *Corel2* respectively. Each Corel group has 100 textures (768 x 512 pixels) and every texture is divided into 6 subimages, then the total number of images is $6x100 = 600$ for each Corel dataset. In figure 4 we show some textures of the three Corel datasets.

(a) *Corel*

(b) *Corel1*

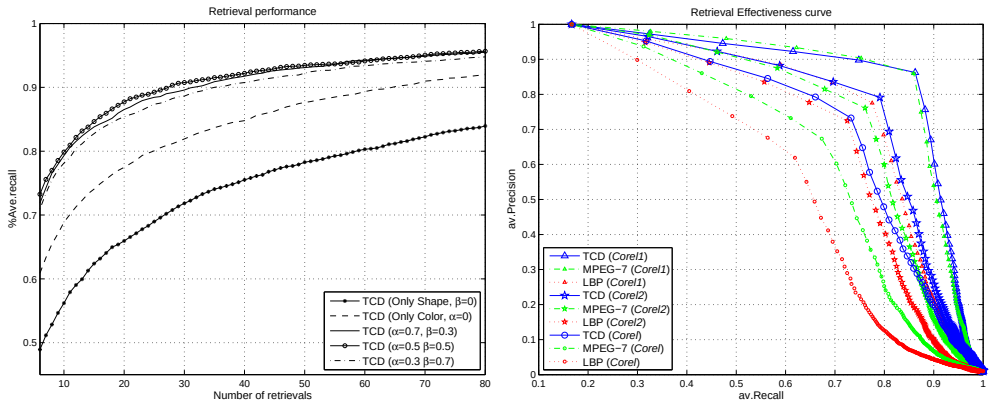
(c) *Corel2*

Fig. 4. Corel datasets.

We use the Recall measure [10] to evaluate the performance of the retrieval and the precision-recall curves. The results have been computed by using all the images in each dataset as query images. In the ideal case of the retrieval, the top 6 retrieved images would be from the same original texture.

We find that using similar weights in the combination of shape and color descriptors to compute the distance (α, β in equation 7) do not have a relevant influence on the average recall measure. This is because color and texture information are already integrated at the blob level, before building the descriptor *TCD*. This fact is illustrated in Fig.5.(a) for *Corel* datasets. Best results in all datasets are obtained when both color and shape are combined (when using only color or shape the average rate substantially decreases).

For comparing purposes, in table 1 we show the Recall rates for the 3 datasets using our *TCD* and two different descriptors that combine color and texture in different ways. These two descriptors are the standard MPEG-7 descriptors [7] (HTD and SCD as they are combined in [2]) and the color extension of the LBP descriptor proposed in [6]. The computed Average Retrieval rate shows how our *TCD* overcomes both the $LBP_{8,1}RGB$ descriptor and the MPEG-7 descriptors for the three Corel datasets. The LBP parameters we have chosen



(a) (b)

Fig. 5. (a) Retrieval performance of TCD with different weights on the *Corel* dataset. (b) Precision-Recall curves of TCD, MPEG-7 and LBP descriptors for different datasets.

Table 1. Average Recall Rates

Descriptor	*Corel*	*Corel1*	*Corel2*
TCD	**73.25%**	**86.25%**	**79.11%**
MPEG-7 (SCD+HTD)	67.33%	85.94%	76.11%
$LBP_{8,1}RGB$	61.89%	77.53%	72.5%
TCD(Only Color)	60.89%	78.56%	69.25%
TCD(Only Shape)	48.92%	49.33%	49.33%
MPEG-7 (SCD)	48.5%	64.56%	61.58%
MPEG-7 (HTD)	55.56%	74.22%	63.64%

are those that produce the best results over the Corel datasets. In Fig.5.(b) there are the precision-recall curves that confirms the previous results. The last four rows of Table 1 show the retrieval rates using either color or texture for *TCD* and MPEG-7 descriptors respectively, showing the contribution of each separate feature on the discrimination experiment.

The best results of the *TCD* are achieved with *Corel1* dataset because it has more homogeneous textures than the other Corel datasets. That is, any subimage of the given texture preserves the same appearance of the texture. The *TCD* is a good descriptor to model the repetitive properties of textures.

5 Conclusions

This paper proposes a perceptual integration of color and texture in an unified descriptor. To this end, we propose a computation procedure to implement the original definition of texton given in the Julesz's perceptual theory [4]. It is done by using two spaces to represent shape and color attributes of the image blobs. Both spaces show two important properties, they are low-dimensional and have perceptual transformations over the axes in order to easily derive similarities from distances.

Although blobs are initially computed separately in the channels of an opponent color space, they are fused with a winner-take-all mechanism over different spatially coincident responses. The shape attributes of these winner blobs (aspect-ratio, area and orientation) that we call *perceptual blobs* are uniformly transformed in the shape space and their median color is represented in a perceptual HSI space. Similarities in these two spaces are combined a posteriori to obtain a final similarity between blobs which is the input of a clustering algorithm. Clusters of blobs are coping with the inherent repetitive property of the image texture. Therefore, the fusion of texture and color is done at the level of their attributes independently of their spatial location.

By combining previous spaces we propose a high level color-texture descriptor, the Texture Component Descriptor (TCD), that arises from the decomposition of the image in its *textural components*, which are the clusters of the blob attributes. Each cluster is defined by a 6-dimensional vector and our TCD will be a list of these vectors, depending on the inherent complexity of the texture. To sum up, the TCD is compact, low-dimensional and it inherits the semantic derived from the blob attributes.

In order to test the efficiency of the proposed descriptor we have performed a retrieval experiment on a highly diverse dataset of Corel Texture images. We compared our descriptor with a late combination of two MPEG-7 descriptors [7] (HTD and SCD) and an early combinations with the LBP *RGB* [6] descriptor in a retrieval experiment. Our descriptor overcomes both in the three Corel datasets of textures analysed.

6 Acknowledgments

This work has been partially supported by projects TIN2007-64577 and Consolider-Ingenio 2010 CDS2007-35100018 of Spanish MEC (Ministry of Science).

References

1. Carron, T. & Lambert, P.: Color Edge Detector Using Jointly Hue, Saturation and Intensity, Int. Conference on Image Processing. vol. 3, pp. 977-981 (1994).
2. Dorairaj, R. & Namuduri, K.R.: Compact combination of MPEG-7 color and texture descriptors for image retrieval, Conference on Signals, Systems and Computers. Conference Record of the Thirty-Eighth Asilomar. vol. 1, pp. 387-391 (2004).
3. Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q. , Dom, B., Gorkani, M., Hafner M. J., Lee D., Petkovic, D. , Steele, D. , Yanker, P.: Query by Image and Video Content: The QBIC System, Computer, vol. 28, no. 9, pp. 23-32 (1995).
4. Julesz, B. & Bergen, J.R.: Textons, the fundamental elements in preattentive vision and perception of textures, Bell.Syst, Tech. Journal vol.62, no. 6, pp.1619-1645 (1983).
5. Lindeberg, T.: Scale-space theory in computer vision, Kluwer Academic Publishers (1994).
6. Maënpää, T. & Pietikäinen, M.: Classification with color and texture: jointly or separately? Pattern Recognition, No. 37. pp. 1629-1640 (2004).
7. Manjunath, B.S., Salembier, P. & Sikora, T.: Introduction to MPEG-7, John Wiley & Sons (2003).
8. Rubner, Y., Tomasi, C. & Leonidas, J.G.: The earth mover's distance as a metric for image retrieval, Int. Journal of Computer Vision vol. 40, no. 2, pp. 99-121, 2000.
9. Shi, J. & Malik, J.: Normalized cuts and image segmentation, IEEE Trans. on PAMI, vol. 22, no. 8, pp. 888-905, 2000.
10. Smith, J.R.: Image Retrieval Evaluation, Proc. IEEE Workshop on Content - Based Access of Image and Video Libraries, pp. 112-113, 1998.
11. Yu, H., Li, M., Zhang, H.J. & Feng, J.: Color Texture Moments for Content-Based Image Retrieval, International Conference on Image Processing, pp. 24-28. 2003.
12. Zhong, Y. & Jain, A.: Object localization using color, texture and shape, Pattern Recognition, vol. 33, no. 4, pp. 671-684, 2000.