IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 27, 2020, accepted August 23, 2020, date of publication August 27, 2020, date of current version September 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019800

Learning Contour-Based Mid-Level Representation for Shape Classification

CHENGZHUAN YANG[1], LINCONG FANG[1], AND HUI WEI[2,3]
[1]School of Information, Zhejiang University of Finance and Economics, Hangzhou 310018, China
[2]Laboratory of Cognitive Algorithm and Model, Fudan University, Shanghai 201203, China
[3]Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University, Shanghai 201203, China

Corresponding author: Chengzhuan Yang (chengzhuanyang@zufe.edu.cn)

This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Project LQ19F020003 and Project LY18F020023, in part by the National Natural Science Foundation of China under Project 61375122 and Project 61902159, and in part by the Shanghai Science and Technology Development Funds under Project 13dz2260200 and Project 13511504300.

ABSTRACT This article proposes a novel contour-based mid-level shape description method for shape classification. This method resolves the shortcomings of low-level shape descriptors in dealing with the shapes of objects with large intra-class changes and non-linear deformation (articulation, occlusion and noise), thus improving the accuracy of shape classification. First, we extract the outer contour of an object and sample it. We next describe each sampling point on the shape contour with a triangular feature and regard it as a local feature. Then, a shape codebook is learned, and the Fisher vector encoding method is used to produce a compact mid-level shape feature. Finally, the learned mid-level shape features are sent to the linear support vector machine (SVM) classifier for shape recognition. The proposed method has been extensively tested on several standard shape datasets, and the experimental results show that our approach attains high accuracy of shape classification. Comparisons to other state-of-the-art shape classification approaches further prove the superiority and effectiveness of our method.

INDEX TERMS Shape classification, triangular feature, fisher vector, mid-level shape representation.

I. INTRODUCTION

Shape is an important visual information of a target object, and it is also an important basis for the human visual system to recognize and classify the target object. Unlike other features of an image, shape can remain unchanged under illumination and other deformation conditions. Because of these advantages, shape-based object recognition has always been a hot topic in computer vision and image processing [1]–[6], and it has been widely used in text analysis, neuroscience, agriculture, biomedicine and engineering technology [7]. Shape recognition is generally referred to as a classification task. The goal of this task is to predict the class label of the test shape when providing a set of training shapes. The main challenge in shape recognition is how to extract the discriminative shape feature of the target object, and it is invariant to the geometrical changes (rotation, translation and scale), the intra-class variation and the nonlinear deformation of the shape.

The associate editor coordinating the review of this manuscript and approving it for publication was Zahid Akhtar.

Although many shape representation approaches have been presented over the past decade, they can mainly be divided into contour-based and region-based approaches [8]–[10]. The contour-based methods use the sampling points on the contour of an object to represent the shape, such as curvature scale space [11], the shape context (SC) [1], the shape tree [12], hierarchical string cuts (HSCs) [13], and multiscale triangular centroid distance (MTCD) [14]. The region-based methods employ the interior region of the contour to represent the shape, such as Zernike moment [15], generic Fourier descriptors [16], polar harmonic transforms [17], and multi-scale bisector integrals [18]. In the past few years, there has been more research activity on contour-based shape description methods than the region-based ones. There are two main reasons for this: human beings can readily recognize the object only with the contour information of an object; and the contour of a shape is more important than the inner region in many applications, such as contour-based object detection [19], [20]. Thus, this article primarily constructs the mid-level representation based on the contour information of a shape.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

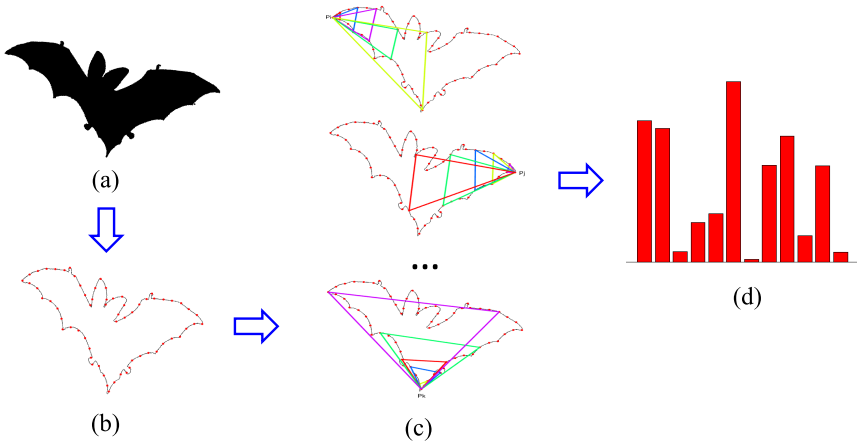

FIGURE 1. The pipeline of building mid-level shape representation by the proposed approach: (a) original shape; (b) the outer contour (black boundary) of the shape (a) and the sampling points of the shape contour (red points), which are usually taken as the input of a contour-based shape recognition approach; (c) the triangular feature of each point of the shape contour; and (d) the final mid-level shape representation.

However, the above shape descriptors are low-level representation methods, and the main disadvantage of these methods is that they cannot handle the shapes of an object with large intra-class variation and nonlinear deformation. In addition, in shape recognition, these methods use one-to-one matching, such as the dynamic programming (DP) approach, to obtain the matching cost of every two shapes. They then adopt the nearest neighbor (NN) classifier to identify the shapes. The main drawback of this recognition method is that it is extremely time consuming when the number of training samples is very large. Unlike the above low-level shape description methods, we develop a mid-level shape descriptor to deal with the large changes of an object's shape. In recent years, there has been some related work on the construction of mid-level features for shape-based object recognition, such as literature [21]–[23]. Of these, the most similar method to the work of this article is literature [22], which is also based on the contour information of a shape, and then uses the bag of words (BOW) model to construct the mid-level shape representation. However, there are four main differences between the proposed method and the method in [22]. First, they need to normalize the shapes using the major axis to keep the rotation of the shape unchanged because the method in [22] is based on the spatial pyramid matching (SPM) strategy. However, due to the large difference in the shape of an object, the major axis method is unstable for shape normalization and will therefore affect the accuracy of shape recognition. Second, our method does not need to divide the contour segments. We use each sample point in the shape contour as a local feature. This is because the shape of an object has an emergent property that becomes only apparent once all the object boundary contours have been grouped. If we divide a complete shape contour into a number of contour segments, so that a large number of redundant contour segments will be generated, the accuracy of shape recognition will be affected. Third, the local feature description method is different from the method of shape context used in [22]. We adopt a triangular feature description method, which can describe the global and local information of the shape contour very well, and this method has a lower feature dimension. Finally, unlike the BOW model used in [22] to construct mid-level features, the proposed method adopts the Fisher Vector (FV) encoding method [24], [25], which can describe the first-order and second-order statistical information of data. The shape descriptor obtained by this coding method has better shape discrimination ability, thereby improving the performance of shape classification. In brief, the proposed mid-level shape description method is simpler and more straightforward and can describe the shape information of objects very effectively.

The pipeline of the proposed contour-based mid-level shape representation method is shown in Fig. 1. First, for a binary image, we extracted the outer contour of an object's shape and sampled it, as shown in Fig. 1(b). Then, we used the triangular feature to characterize each point of the shape boundary, as shown in Fig. 1(c). This shape descriptor can adequately capture the fine details and global information of a shape, and it is also invariant to translation, scale and rotation of a shape. Third, the learning method of the Gaussian mixture model (GMM) was employed to obtain a shape codebook, and the FV encoding method was adopted to obtain the mid-level shape representation, and power normalization and L_2 normalization were subsequently performed on it. The final mid-level shape representation is shown in Fig. 1(d). Finally, the learned mid-level shape feature was sent into the linear support vector machine (SVM) classifier for shape recognition.

The rest of the article is structured as follows. We briefly review some related work in Section II. In Section III, we introduce the proposed contour-based mid-level shape

representation method in detail. Section IV provides the experimental results of our approach on some shape datasets. In Section V, we present our conclusions.

II. RELATED WORK

Many shape classification approaches have been presented over the past decade [2], [22], [23], [26]–[35]. These methods are divided into exemplar-based and model-based methods. Below, we briefly review the most important of these methods.

The exemplar-based methods [2], [26], [28], [30], [31], [33], [35] mainly extract informative and robust shape descriptors, then adopt the shape matching method to compute the similarity between two shapes, and finally classify them according to the NN classifier. For example, Ling [2] proposed an inner distance shape context (IDSC) descriptor for shape classification. This descriptor is an extension of the SC descriptor, which uses the inner distance instead of the Euclidean distance of the SC to better capture the articulation changes of objects. Daliri and Torre [28] converted the contour points of a shape into symbolic representations, adopted the edit distance to compare the similarity between two symbols, and finally classified the shapes according to the distance values. Wang *et al.* [31] proposed a height function descriptor for shape matching and classification. This method adopted a dynamic programming matching approach to calculate the dissimilarity between two shapes and then classified the shapes based on the calculated distance. To better deal with the large deformation of objects, some researchers used the skeleton description method to identify shapes. For instance, Sebastian *et al.* [26] presented a shape recognition framework based on a shock graph description. This method is very effective for detecting the visual deformation of shapes. Macrini *et al.* [30] proposed a shape matching method based on bone graphs and achieved good shape recognition performance. To further improve the performance of shape classification, some new exemplar-based methods have been proposed. For example, Bicego and Lovato [33] presented a bioinformatics method for 2D shape classification. They first converted the 2D shape matching into a sequence matching problem, and then used the sequence matching approach in bioinformatics to calculate the dissimilarity between the two shapes. Finally, the classification task of shapes was completed using the NN framework. In [35], a hexagonal grid-based triangulated feature descriptor was proposed for shape retrieval and classification. This method achieved better shape retrieval and classification performance and had lower computational complexity. However, because the exemplar-based methods use low-level shape representation, they cannot handle complexly deformed objects very well, and at the same time they have very high matching cost when the amount of data in the training set is quite large.

The model-based approach [22], [23], [27], [29], [36], [37] is not based solely on a single match of every two shapes but instead on learning a classification model to identify the shape. For example, Sun and Super [27] presented a Bayesian approach that employed the normalized contour segments of a shape for shape recognition. Bai *et al.* [29] used a Gaussian mixture model to integrate the skeleton and contour information of a shape, and then trained a generative model for shape recognition. Daliri and Torre [36] presented a kernel-based shape recognition method. They first converted the sample points on the contour into symbol strings, then used the edit distance to evaluate the dissimilarity between the symbol strings. Next, they converted the distance to the appropriate kernel before adopting the SVM classifier for shape recognition. Wang *et al.* [22] presented a mid-level shape description method called bag of contour fragments (BCF) for shape classification. This method uses the BOW model to build mid-level shape representation and attains better recognition accuracy in shape classification. Ramesh *et al.* [23] proposed a shape classification method based on invariant characteristics and context information in the BOW model. First, they used the spectral magnitude of the log-polar transformation as a local feature. Next, they integrated the context information into the Bow framework and proposed a method to select the appropriate codebook size in the BOW model. Finally, they trained the SVM classifier to identify shapes. Shen *et al.* [37] proposed a learnable pooling function to effectively combine the features of the BCF and bag of skeleton-associated contour parts (BoSCP) methods. Then, they adopted the SVM classifier for shape recognition based on this combination of methods.

Recently, the deep learning method has attained significant success in computer vision [38], [39], speech recognition [40], [41], natural language processing [42] and other fields [43]. Some researchers have begun to adopt the deep learning approach for 2D shape analysis and recognition [32], [34], [44]. Eslami *et al.* [44] presented a Shape Boltzmann Machine method for shape recognition. To model binary shapes, they used the Deep Boltzmann Machine, which can produce very realistic shape samples that are different from training samples. As a result, this method has a strong generalization ability. Ke and Li [32] proposed a convolution neural network (CNN) method to learn robust high-level shape features and used the method to handle the rotation issue of objects in shape recognition. Li *et al.* [34] introduced a deep neural network optimization method that uses the stochastic gradient Markov chain Monte Carlo (SG-MCMC) to learn the weights uncertainty in a deep neural network, which provides accurate estimates of the uncertainty of the model. They used this method and achieved excellent performance in both 2D and 3D shape classifications. However, the deep learning method also has several shortcomings, such as the need for a great deal of training data, the need for more computing resources for model learning and the poor interpretability of the learned model. The proposed method overcomes these shortcomings: it requires only a small amount of data to learn, the model learning requires few computing resources, and the learned model has better interpretability. Furthermore, subsequent experiments showed that the deep learning method is not the most effective method for 2D shape

recognition through subsequent experiments, and that our proposed method achieved the best performance in shape classification task, superior to not only the traditional shape classification method but also to the shape classification method based on deep learning.

III. PROPOSED CONTOUR-BASED MID-LEVEL SHAPE REPRESENTATION METHOD

This section describes the proposed contour-based mid-level representation method for shape classification in detail. First, we define the triangular feature description of the shape contour sampling point and regard it as a local feature. Next, we describe how the FV is used to build the mid-level shape representation based on triangular features. Finally, we use the linear SVM classifier for shape recognition via the built-in mid-level features.

A. DEFINITION OF TRIANGULAR FEATURE REPRESENTATION

The triangular feature description of the contour sampling points introduced in this subsection is based on our previous work [45]. But unlike the method in [45], we do not need to use the final multiscale Fourier descriptor which is obtained using the Fourier transform. We only use the triangular features of the shape contour sampling points. A brief explanation of the triangular features of the shape contour sampling points is as follows.

For a binary shape image, we use the edge tracking algorithm to obtain the outer contour S of an object, and then we uniformly sample it to acquire the contour sampling point of the object as $S = \{P_1, \ldots, P_N\}$. Here, N represents the number of sampling points, and P_1 and P_N represent the starting and end points of the contour S, respectively. For each sample point $P_i = (x_i, y_i)$, $(i \in [1, N])$ of the shape contour S, we can use T_s triangles at different scales to represent it. T_s is the number of triangles, which takes $T_s = \lfloor log_2(N/2) \rfloor$, and $\lfloor log_2(N/2) \rfloor$ is the lower bound of $log_2(N/2)$. Fig. 2 shows this more specifically.

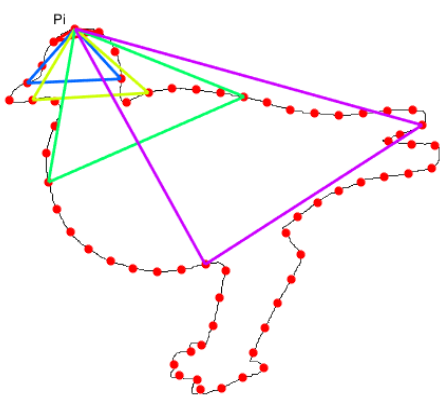

FIGURE 2. The multi-scale triangular feature representation of sampling point P_i for shape contour S. We sample $N = 64$ points for the contour S and the sampling point P_i can be represented by five triangles ($T_s = 5$).

As shown below, each sampling point P_i of the shape contour S can be represented by Ts triangles $Tr_i^1, Tr_i^2, \ldots, Tr_i^{T_s}$, where the triangle Tr_i^k is formed by the contour points P_i, $P_{i+l(k)}$, $P_{i-l(k)}$ $(i \in [1, N], k \in [1, T_s])$. Further, $l(k)$ is the logarithmic distance between sample points of the contour S, while l represents an increasing function where $l(T_s) < N/2$. In our experiment, we set $l(k) = 2^{k-1}$ (cf. Fig. 2). Three triangular features are used to characterize the sampling point P_i, which is defined as follows:

$$\begin{aligned} TAR(P_i) &= (TAR(Tr_i^1), \ldots, TAR(Tr_i^{T_s})), \\ TCD(P_i) &= (TCD(Tr_i^1), \ldots, TCD(Tr_i^{T_s})), \\ TASL(P_i) &= (TASL(Tr_i^1), \ldots, TASL(Tr_i^{T_s})). \end{aligned} \tag{1}$$

Here, $TAR(Tr_i^k)$ denotes the signed area of triangle Tr_i^k, $TCD(Tr_i^1)$ represents the distance from point P_i to the centroid of triangle Tr_i^k, $TASL(Tr_i^1)$ is composed of two side lengths and intersection angle at point P_i of triangle Tr_i^k. Let $(x_{i-l(k)}, y_{i-l(k)})$, (x_i, y_i), and $(x_{i+l(k)}, y_{i+l(k)})$ represent the coordinates of the points $P_{i-l(k)}$, P_i and $P_{i+l(k)}$, respectively. Initially, we compute the signed area of triangle $TAR(Tr_i^k)$, which is given by

$$TAR(Tr_i^k) = \frac{1}{2} \begin{vmatrix} x_{i-l(k)} & y_{i-l(k)} & 1 \\ x_i & y_i & 1 \\ x_{i+l(k)} & y_{i+l(k)} & 1 \end{vmatrix}. \tag{2}$$

The TAR value is capable of characterizing the concave and convex properties of a shape. Next, we calculate the central distance $TCD(Tr_i^1)$ of Tr_i^k, which is given by the following expression:

$$TCD(Tr_i^k) = \sqrt{(x_i - x_{g_{ik}})^2 + (y_i - y_{g_{ik}})^2}. \tag{3}$$

Here, $(x_{g_{ik}}, y_{g_{ik}})$ is the coordinates of the centroid point g_{ik}, with $i \in [1, N]$ and $k \in [1, T_s]$ of triangle $\triangle P_{i-l(k)} P_i P_{i+l(k)}$, which is defined as follows:

$$\begin{cases} x_{g_{ik}} = (x_{i-l(k)} + x_i + x_{i+l(k)})/3 \\ y_{g_{ik}} = (y_{i-l(k)} + y_i + y_{i+l(k)})/3. \end{cases} \tag{4}$$

Lastly, we calculate the two side lengths and intersection angle at P_i of Tr_i^k. Let L_i^{1k}, L_i^{2k} and L_i^{3k} denote the lengths of the three sides of the triangle Tr_i^k, and the lengths are arranged from small to large $(L_i^{1k} \le L_i^{2k} \le L_i^{3k})$. Let $M_i^k = \frac{L_i^{1k}}{L_i^{3k}}$ and $N_i^k = \frac{L_i^{2k}}{L_i^{3k}}$. α_i^k is the intersection angle at point P_i of Tr_i^k. Therefore, $TASL(Tr_i^k)$ is defined as follows:

$$TASL(Tr_i^k) = (M_i^k, N_i^k, \alpha_i^k). \tag{5}$$

The intersection angle α_i^k at point P_i of Tr_i^k is given by

$$\alpha_i^k = acos\left(\frac{\overrightarrow{P_iP_{i-l(k)}} \cdot \overrightarrow{P_iP_{i+l(k)}}}{|\overrightarrow{P_iP_{i-l(k)}}| \cdot |\overrightarrow{P_iP_{i+l(k)}}|}\right). \tag{6}$$

From the definition of the above three triangular features, we can see that the sizes of these triangular features are T_s, T_s and $3 \times T_s$. Therefore, we can obtain the multiscale triangular

feature (MTF) of point P_i, which can be expressed by the following equation:

$$MTF(P_i) = (TAR(P_i), TCD(P_i), TASL(P_i)). \tag{7}$$

We can see that the size of $MTF(P_i)$ is $1 \times L$, where $L = 5 \times T_s$. As a result, we obtain the multiscale triangular feature of shape S, which is given by

$$\begin{aligned} MTF(S) &= (MTF(P_1), \ldots, MTF(P_N))^T \\ &= \begin{pmatrix} TAR(P_1) & TCD(P_1) & TASL(P_1) \\ \vdots & \ddots & \vdots \\ TAR(P_N) & TCD(P_N) & TASL(P_N) \end{pmatrix}. \end{aligned} \tag{8}$$

We can see that the size of $MTF(S)$ is $N \times L$, where the row i represents the feature $MTF(P_i)$ of the sampling point P_i of S. Next, we analyze the geometric invariant properties (translation, scale, and rotation) of the MTF. We find that the *TAR* and *TCD* descriptors remain unchanged for translation and rotation of a shape, except for changes in scale. The *TASL* descriptor remains unchanged for translation, rotation and scale of a shape. At the same time, to keep the included angle of the TASL descriptor in the interval [0, 1], we normalized the included angle of the TASL on π. Finally, we normalize the *TAR* and *TCD* using the local normalization method given in [46]. The calculation formula is

$$\begin{aligned} TAR(P_i, Tr_i^k) &= \frac{TAR(P_i, Tr_i^k)}{max_{i=1}^N |TAR(P_i, Tr_i^k)|}, \\ TCD(P_i, Tr_i^k) &= \frac{TCD(P_i, Tr_i^k)}{max_{i=1}^N |TCD(P_i, Tr_i^k)|}. \end{aligned} \tag{9}$$

This calculation formula indicates that we normalize the *TAR* and *TCD* descriptors by dividing the maximum absolute value at each scale. All values of the MTF matrix after normalization are between -1 and 1. Therefore, we obtain the triangular features of each sample point of the shape boundary, which is employed as a local feature for the subsequent construction of the mid-level shape representation. Moreover, the descriptor has four advantages: (1) the geometric transformation of the shape remains unchanged; (2) the multi-scale representation structure, which can describe the local and global features of a shape; (3) the compactness of the feature description, which requires less memory to store the descriptor; (4) the descriptor is obtained by adjacency contour points, which can represent the spatial layout information between sampling points. This is also the reason why we do not need to use the SPM method to add spatial layout information in the subsequent construction of mid-level shape representation.

B. CONSTRUCTION OF THE CONTOUR-BASED MID-LEVEL SHAPE REPRESENTATION

After we extract the local features of the shape, we then use the FV to build the mid-level shape representation. The FV is first proposed for the recognition of natural images [24], [25], which achieves good recognition accuracy in natural image classification. Below, we briefly describe how to use this method in the field of 2D shape analysis.

For a given shape contour S, we can obtain the local features of its multi-scale triangle as $S = \{f_t \in R^L, t = 1, 2, .., N\}$. Here, we assume that S is generated via a probability density function u_λ whose parameter is λ. According to literature [24], [25], we choose u_λ as a GMM: $u_\lambda(f) = \sum_{i=1}^K w_i u_i(f)$, where K denotes the number of components in GMM. The parameter λ of the GMM is expressed as $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1, \ldots, K\}$. Here, w_i, μ_i and Σ_i represent the mixture weight, the average vector and the covariance matrix of Gaussian u_i, respectively. At the same time, we also suppose that Σ_i is a diagonal matrix and σ_i^2 is used to represent the variance vector. The GMM u_λ is obtained by the maximum likelihood (ML) estimation, which is trained on a set of local shape features from the training shapes. Let $\gamma_t(i)$ be the soft assignment of local feature f_t to Gaussian i:

$$\gamma_t(i) = \frac{w_i u_i(f_t)}{\sum_{j=1}^K w_j u_j(f_t)}. \tag{10}$$

Since the dimension of local feature f_t is L, we can obtain two gradients of L dimension in regard to mean μ_i and standard deviation σ_i, respectively. Their computational expressions are

$$\mathcal{G}_{\mu,i}^S = \frac{1}{N\sqrt{w_i}} \sum_{t=1}^N \gamma_t(i)(\frac{f_t - \mu_i}{\sigma_i}), \tag{11}$$

$$\mathcal{G}_{\sigma,i}^S = \frac{1}{N\sqrt{2w_i}} \sum_{t=1}^N \gamma_t(i)[\frac{(f_t - \mu_i)^2}{\sigma_i^2} - 1]. \tag{12}$$

Among them, the division between vectors is item-by-item arithmetic. We then concatenate the two gradient vectors $\mathcal{G}_{\mu,i}^S$ and $\mathcal{G}_{\sigma,i}^S$ $(i = 1, \ldots, K)$ to form the mid-level shape representation FS_λ^S, whose characteristic dimension is $2KL$.

Next, we use the power normalization [24] through the element-wise nonlinear operation to suppress the problem of burstiness. For each component fs_i $(i = 1, \ldots, 2KL)$ of the mid-level shape representation FS_λ^S, the element-wise nonlinear operation is defined as:

$$fs_i := |fs_i|^\alpha \times sign(fs_i), \tag{13}$$

where the value of α is in the interval [0, 1]. In the experiment, we set the α value to 0.2. Then we employ the L_2 normalized method to obtain the final mid-level shape representation that is used for shape classification.

C. SHAPE CLASSIFICATION USING MID-LEVEL FEATURE

The resulting contour-based mid-level shape feature is a vector, so we can employ the linear SVM to classify shapes directly. We adopt the method introduced by Crammer and Singer *et al.* [47] for multi-class SVM. Let $\{(f_i, y_i)\}_{i=1}^M$ be the training set, which is composed of M shapes from C categories, where f_i and y_i represent the contour-based mid-level feature and the category label of i-th shape, respectively. Next, we train a multi-class SVM classifier, which is

TABLE 1. Classification accuracy of our method on the Animal dataset before and after dimension reduction with principal component analysis (PCA).

Method	N	K	D	$\hat{D} = D$	$\hat{D} = 2048$	$\hat{D} = 1024$	$\hat{D} = 512$	$\hat{D} = 256$	$\hat{D} = 128$
Our Method	1024	256	23,040	88.85%	89.26%	88.52%	88.83%	88.68%	88.44%

defined as

$$\min_{w_1,\ldots,w_C} \sum_{j=1}^{M} ||w_j||^2 + \beta \sum_i \max(0, 1 + w_{l_i}^T f_i - w_{y_i}^T f_i), \quad (14)$$

where $l_i = \arg\max_{l\in[1,2,\ldots,C], l\neq y_i} w_l^T f_i$. In Eq. (14), the left part of the equation represents the regularization term, and the right part represents the hinge-loss. The parameter β is used to control the relative weight between the left and right parts in Eq. (14). To solve Eq. (14), we employ the off-the-shelf SVM, LibLinear, introduced by Fan *et al.* [48]. For a test shape feature f, its class label is predicted by

$$\hat{y} = \arg\max_{l\in[1,\ldots,C]} w_l^T f. \quad (15)$$

Here, we select the linear SVM classifier because the efficiency of this method is relatively high. At the same time, the method can be extended to large-scale shape retrieval and recognition when the linear SVM classifier is used. In addition, the accuracy of shape classification is further improved when the other nonlinear kernels are used, such as the intersection kernel or the Gaussian kernel function.

IV. EXPERIMENTAL RESULTS

We evaluated the classification performance of our method on several standard shape datasets and compared it with the state-of-the-art shape classification approaches discussed in this section. First, we described the experimental setup. Then, we provided the classification result of our method on four standard shape datasets, including the Animal dataset [29], the MEPG-7 dataset [49], the Swedish leaf dataset [50], and the ETH-80 dataset [51]. Next, the robustness of our method was studied, and we then analyzed the influence of the parameters introduced in our method on the performance of shape classification. Finally, we tested the performance of our method for object classification in real-world images.

A. EXPERIMENTAL SETUP

For a given shape S, we sampled it for $N = 1024$ points. Thus, we obtained that the size of each sampling point in the shape was $L = 5 \times T_s = 5 \times \lfloor log_2(N/2) \rfloor = 5 \times 9 = 45$. We regarded the feature of each sampling point as a local feature of the shape. When learning the shape codebook, the number of clustering center K was set to 256. We studied the effect of different numbers of clustering centers on the performance of shape classification. Then, we employed the FV encoding method to obtain the size of each shape as $D = 2 \times L \times K = 2 \times 45 \times 256 = 23,040$. Finally, we used the linear SVM for shape classification, where the weight of the parameter β was set to 10 in all experiments. The algorithm was implemented using the MATLAB language, and the experiment was conducted on a PC with an Intel Core i7-8700K 3.7GHz CPU and 32 GB RAM. We only needed around 28 ms to calculate the triangular feature descriptor of a shape. The learning of shape codebook took around 5 min, and it took about 10 ms to encode the feature vector for a shape. It took around 22 ms to test a shape.

We also used the principal component analysis (PCA) method to decrease the dimensions of shape features, thereby reducing memory usage. The PCA method is one of the most widely used data dimensionality algorithms in computer vision. In the experiment, we directly adopted the PCA method on the contour-based mid-level shape feature, thereby reducing the dimension of the feature (from D to $\hat{D}$ as shown in Table 1). The influence of different feature dimensions on shape classification performance was then analyzed. As is shown later, in the experiment, the classification rate of the proposed contour-based mid-level feature was not greatly affected, but sometimes it is beneficial to the classification result due to the feature selection ability of PCA.

B. ANIMAL DATASET

We tested the proposed approach on the Animal dataset, which was introduced by Bai *et al.* [29]. This dataset is composed of 2,000 shapes, which are divided into 20 categories, each containing 100 shapes. Several samples of the Animal dataset are presented in Fig. 3. Shape classification is challenging on this dataset because the shapes in the dataset contain very large intra-class variations and gesture changes. For direct comparison with other methods, we used the same performance evaluation criteria as in [22], [23], [27], [29], [33]. We randomly chose 50 shape images per class for training, and the remaining shapes were used for testing. To avoid the deviations caused by randomness, we performed 10 experiments and averaged them for classification accuracy. We first analyzed the effect of PCA on the performance of shape classification, as shown in Table 1, then we compared our approach to the other shape classification approaches. Table 2 compares the experimental results of our approach and other state-of-art shape classification approaches.

We can observe from Table 1 that the classification accuracy of our method remains stable after using the PCA dimension reduction algorithm. At the same time, we can see that our method obtains the highest classification performance when the feature dimension is reduced $\hat{D} = 2048$. A possible reason for this is that the feature selection ability of PCA is amplified by the SVM classifier. In addition, we can see that there is almost no loss of classification accuracy when the dimension is reduced $\hat{D} = 512$. In the subsequent experiments, we only give the classification accuracy when

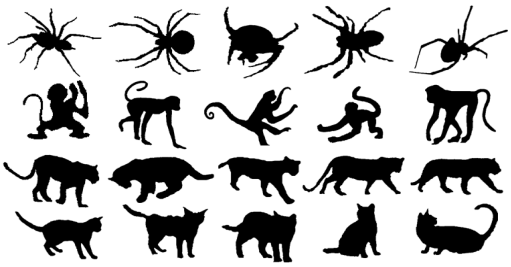

FIGURE 3. Samples of the Animal dataset. The first row shows five samples of the class of spiders, the second row shows five samples of the class of monkeys, the third row represents five samples of the class of leopards, and the last row represents five samples of the class of cats.

TABLE 2. Classification results of different methods on the Animal dataset.

Method	Classification accuracy
Class segment set [27]	69.70%
IDSC [2]	73.60%
Bag of SIFT [52]	74.90%
Contour segments [29]	71.70%
Skeleton paths [29]	67.90%
ICS [29]	78.40%
Multiscale Fourier descriptor [45]	67.35%
Distinctive Parts [53]	80.70%
BCF [22]	$83.40 \pm 1.30\%$
Bioinformatics [33]	83.70%
pSGLD [34]	$84.80 \pm 0.35\%$
Contextual BOW [23]	86.00%
BoSCP-LP [37]	$\mathbf{89.77 \pm 0.65\%}$
Our Method	$88.85 \pm 0.66\%$
Our Method ($\hat{D} = 2048$)	$\mathbf{89.26 \pm 0.74\%}$
Our Method ($\hat{D} = 512$)	$88.83 \pm 0.48\%$

using PCA to reduce the dimension to $\hat{D} = 512$. Therefore, the PCA dimension reduction algorithm reduces the memory usage and improves the efficiency of shape classification.

We can see from Table 2 that our approach achieves the highest classification performance among all of the competing approaches except for the BoSCP-LP method [37]. The classification accuracy of our method is 4% higher than the deep learning method of pSGLD [34]. In addition, the classification accuracy of our approach is nearly 6% higher than the popular model-based shape classification method of BCF [22] and nearly 3% higher than the Contextual Bow method [23]. Our method also achieves comparable classification performance with the recently proposed model-based BoSCP-LP method [37]. The classification accuracy of our approach is 5% higher than the advanced exemplar-based method of Bioinformatics [33]. We also see that the recognition accuracy of our method is 21% higher than that of the multiscale Fourier descriptor [45]. Although our mid-level shape feature is built based on the multiscale Fourier descriptor, the high-level semantic information of an object's shape can be obtained using the FV encoding method. Therefore, the built-in mid-level shape feature is better at shape discrimination. Low-level shape features are very difficult to deal with large intra-class variations and inter-class similarities of objects. Because the shape of objects in the Animal dataset varies greatly, the recognition accuracy of the low-level multiscale Fourier descriptor on this dataset is low. We adopt the 1-NN approach to obtain the classification result of the multiscale Fourier descriptor method. If not specified otherwise, the subsequent experimental results of the multiscale Fourier descriptor are obtained by the 1-NN classification method.

We also provided the classification accuracy rate of each class of the Animal dataset, as shown in Table 3. It can be seen that our approach greatly improves the recognition accuracy of the cat, leopard and monkey classes. The results indicate that our approach can capture the objects with large intra-class variations and non-linear deformations. At the same time, our approach achieves the highest recognition accuracy in most categories of the Animal dataset. Therefore, we deem our method more appropriate for shape recognition than the other methods.

C. MPEG-7 DATASET

The MPEG-7 dataset [49] is extensively used to evaluate the recognition accuracy of various shape classification methods. It is made up of 1400 shapes, which are divided into 70 categories, each containing 20 shapes. Fig. 4 shows several examples of this dataset. Some shapes of the MPEG-7 dataset have very large intra-class variation and complex deformation, so the dataset is very challenging for shape classification.

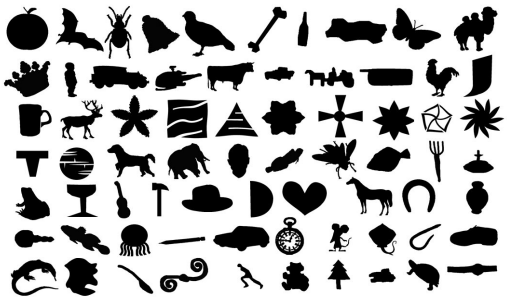

FIGURE 4. Typical examples from the MPEG-7 dataset. One image per class is shown.

There are several shape classification methods that have reported experimental results on this dataset [22], [27], [33]. In order to compare our approach to these methods, we employed the same performance assessment criteria, namely the half training and the leave-one-out. Both are methods commonly used to assess classification performance. The half training evaluation criteria indicate that we randomly select half of the shapes in each category (i.e., 10 shapes per category) to train the model and the other half of the shapes to test the performance of the model. The experiment is repeated 10 times, and the average classification accuracy and standard deviation are given. The leave-one-out evaluation criteria indicate that we adopt all shapes except the current shape for training and employ the rest of the shapes for testing, and the average shape recognition

TABLE 3. Classification accuracy of our method on the Animal dataset before and after dimension reduction with PCA.

Method	Bird	Butterfly	Cat	Cow	Crocodile	Deer	Dog	Dolphin	Duck	Elephant
CS [29]	0.760	0.890	0.390	0.700	0.540	0.690	0.690	0.870	0.830	0.950
ICS [29]	0.760	0.930	0.480	0.800	0.660	0.790	0.750	0.890	0.890	**0.970**
BCF [22]	0.876	0.922	0.738	0.774	0.768	0.904	**0.826**	0.890	0.870	0.952
Our Method	**0.912**	**0.964**	**0.820**	**0.820**	**0.858**	**0.922**	0.820	**0.950**	**0.936**	0.956
Method	Fish	Flyingbird	Hen	Horse	Leopard	Monkey	Rabbit	Rat	Spider	Tortoise
CS [29]	0.700	0.570	0.890	0.960	0.560	0.210	0.810	0.520	0.980	0.810
ICS [29]	0.740	0.650	0.940	0.970	0.650	0.330	0.870	**0.840**	**1.000**	0.900
BCF [22]	0.798	0.720	0.942	0.954	0.664	0.584	0.858	0.706	0.992	0.936
Our Method	**0.812**	**0.848**	**0.976**	**0.992**	**0.802**	**0.750**	**0.970**	0.796	0.996	**0.952**

accuracy is given. Table 4 compares the classification results of our method with those of the other shape classification approaches.

TABLE 4. Classification results of different approaches on the MPEG-7 dataset.

Method	Half training	Leave-one-out
Class segment set [27]	90.9%	97.93%
IDSC+DP [27]	88.43%	90.71%
Contour segments [29]	91.10%	-
Skeleton paths [29]	86.70%	-
ICS [29]	96.60%	-
HGTT [35]	-	95.09%
Manifold kernel SVM [54]	-	96.57%
Polygonal multi-resolution [55]	-	97.57%
String of symbols [56]	-	97.36%
Robust symbolic [28]	-	98.57%
Kernel-edit distance [36]	-	98.93%
BCF [22]	97.16%	98.93%
Bioinformatics [33]	96.10%	98.10%
BoSCP-LP [37]	$98.72 \pm 0.42\%$	-
Multiscale Fourier descriptor [45]	96.27%	97.93%
Our Method	$\mathbf{98.77 \pm 0.65\%}$	99.14%
Our Method ($\hat{D} = 512$)	$98.75 \pm 0.70\%$	**99.21%**

We can see from Table 4 that our approach achieves the highest classification accuracy of all the approaches when using the half training evaluation criteria. At the same time, our approach also achieves the best classification performance among all the competing approaches using the leave-one-out evaluation criteria. In addition, the performance of our method remains stable even when the PCA is used to reduce shape features to 512 dimensions. Similarly, the classification accuracy of our method exceeds that of all the other methods in the half training evaluation criteria and in the leave-one-out evaluation criteria.

D. SWEDISH LEAF DATASET

The third dataset used in our shape classification experiment was the Swedish leaf [50]. This dataset is composed of 15 species of leaf images, each with 75 leaves for a total of 1125 leaf images. Fig. 5 shows several sample images from this dataset. Because the leaves in this dataset have very large intra-class variations and inter-class similarities, it is very challenging for leaf classification. We can see from Fig. 5 that several species are very similar in terms of leaf shape, like the first, third and ninth species. The last image represents an example of a mask image that we used to extract the shape contour of the leaf.

FIGURE 5. Sample images of the Swedish leaf dataset, one image per species. The final leaf is an example of a mask image.

To compare our approach to the existing methods, we followed the same experimental setup as literature [2], [12], [14], [22], [45], [50]. For the 75 leaf images in each species, we randomly chose 25 leaves for training and used the remaining 50 leaves for testing. We conducted the experiment 10 times and reported the recognition accuracy and standard deviation on this dataset. Table 5 presents the recognition results of our method and the other shape classification approaches. The recognition results of other approaches come directly from the published results.

TABLE 5. Classification results of different methods on the Swedish leaf dataset.

Method	Recognition rate
Soderkvist [50]	82.40%
SC+DP [2]	88.12%
Fourier Descriptor [2]	89.60%
IDSC+DP [2]	94.13%
Shape-tree [12]	96.28%
Spatial PACT [57]	90.61%
MDM [58]	93.60%
IDSC+Morphological strategy [59]	94.80%
Manifold MKL SVM [54]	94.40%
Robust symbolic [28]	95.47%
Hierarchical feature using CNN [32]	95.53%
MTCD [14]	96.31%
BCF [22]	96.56%
Pattern Counting [60]	97.07%
Multiscale Fourier descriptor [45]	97.27%
Our Method	$\mathbf{98.31 \pm 0.50\%}$
Our Method ($\hat{D} = 512$)	$\mathbf{98.33 \pm 0.62\%}$

We can see from Table 5 that our method achieves the highest classification accuracy of all the methods compared. The recognition accuracy of our approach is nearly 3% higher than the deep learning method of hierarchical feature using CNN [32], is nearly 2% higher than the model-based method of BCF [22] and is nearly 4% higher than the model-based Manifold MKL SVM method [54]. Furthermore, the recognition accuracy of our approach is higher than the two exemplar-based methods of pattern counting [60] and multiscale Fourier descriptor [45] by 1.2% and 1%, respectively. We also see that our approach still achieves very high classification accuracy when the PCA is used to reduce shape features to 512 dimensions. Even though the shapes of the different species are highly similar, our approach still shows a strong ability for shape discrimination.

E. ETH-80 DATASET

The final dataset utilized in our experiment was the ETH-80 [51], which contains 80 high-resolution 3-D objects from eight classes. Each object contains 41 images captured from various angles, so the dataset is composed of 3,280 images. The segmentation masks of all images are given, so it is very easy to assess shape classification approaches on the ETH-80 dataset. Fig. 6 presents several samples of this dataset.

FIGURE 6. Some examples from the ETH-80 dataset. Each category is shown in a separate row.

The leave-one-object-out performance assessment standard is utilized on the ETH-80 dataset, which is introduced in [51]. It means that images from 79 objects are used to learn the classifier, and the images of the rest of the objects are employed for testing. We compared the recognition result of our method with other shape classification methods, and Table 6 lists the classification results of the different approaches on this dataset.

It can be seen that our approach achieves the highest classification accuracy of all the methods. The classification accuracy of our approach is 6% higher than the best example-based method of Bioinformatics [33]; 4% higher than the best model-based manifold MKL SVM method [54]; and more than 2% higher than the advanced deep learning-based hierarchical feature using the CNN method [32]. We can also see that the classification accuracy of our approach exceeds that of all other shape classification approaches when using PCA to reduce the shape feature to 512 dimensions. Therefore, it can be deduced that our method has a very strong ability for shape recognition.

TABLE 6. Classification rates of different approaches on the ETH-80 dataset.

Method	Classification accuracy
Color histogram [51]	64.86%
PCA gray [51]	82.99%
PCA masks [51]	83.41%
MTCD [14]	82.29%
SC+DP [51]	86.40%
Multiscale Fourier descriptor [45]	86.91%
IDSC+DP [2]	88.11%
IDSC+Morphological strategy [59]	88.04%
Height function [31]	88.72%
Robust symbolic [28]	90.28%
Kernel-edit distance [36]	91.33%
BCF [22]	91.49%
Bioinformatics [33]	91.50%
Manifold MKL SVM [54]	93.75%
Hierarchical feature using CNN [32]	95.80%
Our Method	**97.93%**
Our Method ($\hat{D} = 512$)	**97.38%**

F. ROBUSTNESS TO NOISE

We also tested the effect of our method on shape classification performance in the presence of noise. We added noise to the shape contours and performed shape classification on these noise shape contours. We used the entire MPEG-7 dataset [49] as the initial shape contours. These noises were appended by perturbing the x- and y- coordinates of the shape contour. The noise values are derived from a Gaussian random variable with zero mean and standard derivation σ. Fig. 7 shows an example of noise contours with different σ. As σ becomes larger, the shape of an object is affected more and the shape contour becomes rougher. Fig. 8 shows the contour-based mid-level features under different noise conditions. We can see that the contour-based mid-level features can remain relatively stable, and the increasing of σ has very little effect on our method.

To measure the dissimilarity of their features at different noise levels, we use the L_1 distance to calculate the dissimilarity between noise shape and original shape at different noise levels. Fig. 9 shows the L_1 distance at five different noise levels. We can see that the proposed mid-level feature can maintain the overall stability under different noise levels. At the same time, we also analyzed the mid-level features of different classes of object shapes to measure the stability of the proposed method. We use L_1 distance to compute the shape difference between mid-level features. Fig. 10 shows a total of five shapes, in which Fig. 10(a) shows a beetle image and Fig. 10(b) represents its scale- and rotation-changed version. Fig. 10(c) shows a beetle image which has a large intra-class change compared with that of Fig. 10 (a). Fig. 10(d) and 10(e)

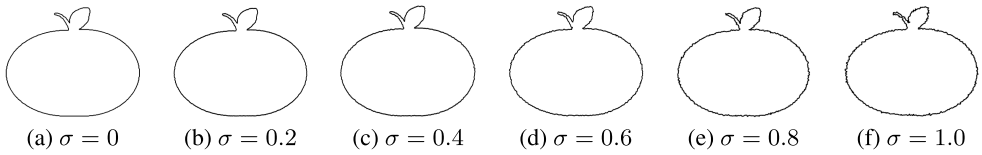

FIGURE 7. An instance of shape contours using different Gaussian noises.

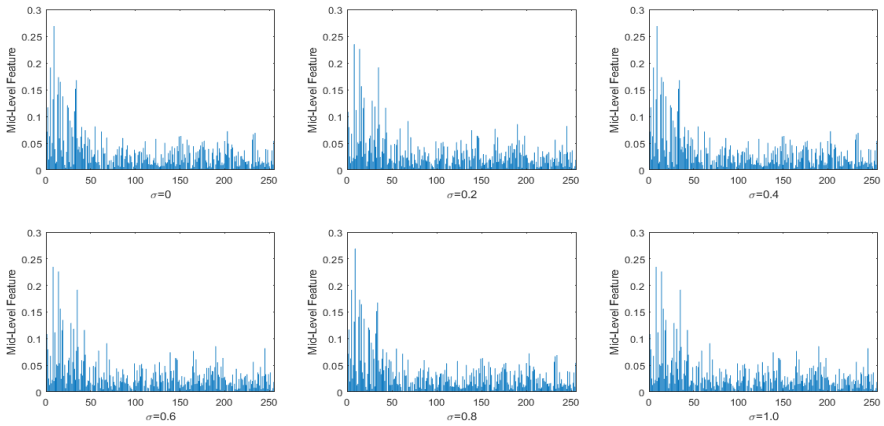

FIGURE 8. Mid-level features with different Gaussian noise levels σ.

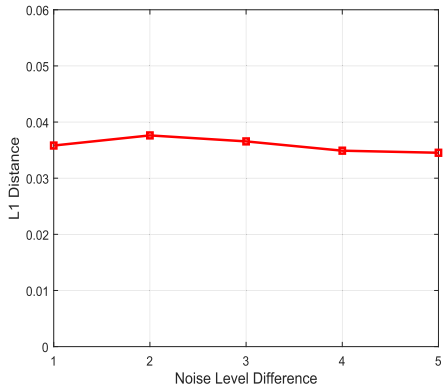

FIGURE 9. The L_1 distances between five noise shapes and original shape.

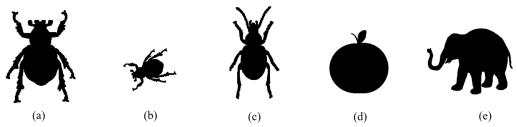

FIGURE 10. An instance of five shapes with different changes.

show the shapes of apple and elephant objects, respectively. Then we calculate the L_1 distances between the shape in Fig. 10(a) and the other four shapes in Fig. 10 (b) to 10 (e). We obtain the distances $L_1(a, b)$=0.0310, $L_1(a, c)$=0.0391, $L_1(a, d)$=0.0601, $L_1(a, e)$=0.0620. We can see that the L_1 distance between the shapes of the same class is much smaller than between the shapes of different classes. Therefore, the proposed mid-level feature can significantly distinguish objects of different classes and achieves a strong ability to distinguish shapes.

Finally, we also analyzed the classification accuracy of our method under different noise levels σ. We used the evaluation criteria of the MPEG-7 shape dataset, which adopts half of the shapes in each category of the dataset for training and the remaining shapes for testing. Fig. 11 shows the classification results of our method when the noise level σ varies from 0 to 1. It can be seen that the classification accuracy of our approach is almost unaffected when the noise level $\sigma = 0.2$. At the same time, the classification accuracy of our method is only slightly reduced under the noise from $\sigma = 0.4$ to $\sigma = 1.0$. The classification accuracy of our method is only reduced by 0.8% when σ increases from 0 to 1. This experimental result further shows the robustness of our approach to noise.

G. PARAMETER STUDY

In this section, we describe our study of the influence of the parameters introduced by our method on shape classification performance. The two important parameters in our approach are the number of the shape contour sampling points N, and the number of clustering center K. We employed the Animal dataset for our parameter analysis. In the experiment, one parameter serves as a variable, and the other parameter remains unchanged. The performance evaluation criteria are the same as the Animal dataset, where half of shapes in each category are used for training, and the rest of the shapes are used for testing. Table 7 lists the classification results for different N and M values.

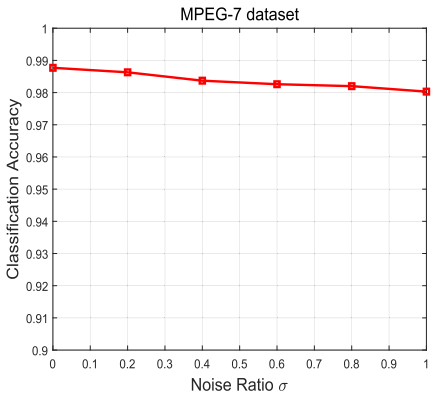

FIGURE 11. Classification results of the MPEG-7 shape dataset are given under the noise level σ varying from 0 to 1.

TABLE 7. Classification results of the Animal dataset for different values of N and K used in the proposed method.

	$K = 64$	$K = 128$	$K = 256$	$K = 512$	$K = 1024$
$N = 64$	0.7981	0.8085	0.8212	0.8158	0.8051
$N = 128$	0.8406	0.8499	0.8575	0.8508	0.8488
$N = 256$	0.8706	0.8711	0.8799	0.8738	0.8737
$N = 512$	0.8742	0.8806	0.8869	0.8847	0.8762
$N = 1024$	0.8778	0.8874	**0.8885**	0.8853	0.8830

We can see from Table 7 that our method obtains the highest classification accuracy on the Animal dataset when $N = 1024$ and $K = 256$. We also see that the classification result of our method increases with the increase of N. In addition, the increase in the classification accuracy gets smaller and smaller when the value of N is further increased, as we can see from $N = 512$ to $N = 1024$. This is mainly because increasing the number of sampling points N can enhance the ability of triangular features to capture the local details and spatial information of a shape. This will result in further enhancement of the discriminative ability of the final mid-level shape feature and improvement of the performance of shape classification. We can also see that the recognition results of our method first increase and then decrease with the increase of K. The best classification result is obtained when $K = 256$. Therefore, we set $N = 1024$ and $K = 256$ in all the experiments.

To analyze the influence of local shape features on the final shape classification performance, we studied the classification accuracy of different local shape features on the Animal dataset. First, we analyzed the influence of different combinations of the proposed triangular feature on shape classification performance. Next, we further compared the proposed triangular feature with other classical local shape descriptors. Table 8 shows the classification results of different combinations of the triangular features on the Animal dataset for $N = 1024$ and $K = 256$. Each row in Table 8 indicates that only one or several feature combinations are used. For instance, the first row of Table 8 indicates that only the TAR feature is used as the local shape feature, and the last row indicates that the TAR, TCD and TSAL are combined as

TABLE 8. Classification results of different combinations of our approach on the Animal dataset ($N = 1024$, $K = 256$).

Situations	Classification accuracy
TAR	$78.84 \pm 1.17\%$
TCD	$74.61 \pm 1.15\ \%$
TASL	$85.55 \pm 1.01\ \%$
TAR+TCD	$85.81 \pm 0.76\ \%$
TAR+TASL	$\mathbf{89.04 \pm 0.86\ \%}$
TCD+TASL	$85.53 \pm 1.19\ \%$
TAR+TCD+TASL	$88.85 \pm 0.66\ \%$

the local shape feature. It can be seen from Table 8 that the best classification result is obtained on the Animal dataset when the TAR is combined with TSLA. The classification accuracy is slightly less when the TCD feature is added to the local shape feature, but the overall classification performance is still relatively stable. The classification accuracy of a single triangular feature is low. For example, the TCD feature obtains only a classification accuracy of 74.61% on the Animal dataset, which is 14% lower than the best classification performance. Combining multiple triangular features can achieve higher shape classification performance. For example, combining TCD and TAR features achieves 85.81% classification accuracy on the Animal dataset, which is higher than only TAR and only TCD features by 7% and 11%, respectively. Therefore, the experimental results indicate that local shape feature has a significant influence on shape classification performance.

To further verify the effect of local shape features on shape classification performance, we compared our proposed triangular feature with the classical shape context (SC) [1], inner-distance shape contexts (IDSC) [2], and height function [31] local shape descriptors. In the experiment, all the parameters were set the same except the local shape feature. We first used the FV method to encode these local shape features and to obtain the corresponding mid-level shape features. Then, we use the mid-level shape feature for shape classification. Table 9 shows the comparison results between our method and these classical shape descriptors. Our method achieves the best classification accuracy on the Animal dataset among all the comparison methods. The recognition rate of our method is higher than that of SC [1], IDSC [2], and height function [31] by 13.9%, 13.2% and 24.2%, respectively. The height function method obtains the lowest recognition accuracy of all the approaches because of its inability to characterize the complex shape changes of an object. The recognition accuracy of the IDSC method is slightly higher than that of the SC method, primarily because it can deal with the articulation changes of an object's shape better. This experiment further shows that the multiscale triangular feature can better deal with the complex change of an object's shape, and the accuracy of shape recognition can be improved by combining it with the FV encoding method.

Next, we also analyze the relationship between the shape complexity and the contour sampling point N. Here, we select five categories from the Animal dataset: birds, cats, deer, fish,

TABLE 9. Classification results of different local shape features on the Animal dataset ($N = 1024$, $K = 256$).

Method	Classification accuracy
SC+FV [1]	$75.15 \pm 0.67\%$
IDSC+FV [2]	$75.83 \pm 1.34\%$
Height function +FV [31]	$65.89 \pm 1.55\%$
Our Method(TAR+TASL)	$\mathbf{89.04 \pm 0.86}$ **%**
Our Method(TCD+TAR+TASL)	88.85 ± 0.66 %

and monkeys. For each category, we choose six representative images. Fig. 12 shows a total of 30 shapes for five categories. The shape complexity (SC) is defined as

$$SC = Avg(SC_{TAR}, SC_{TCD}, SC_{TASL}), \tag{16}$$

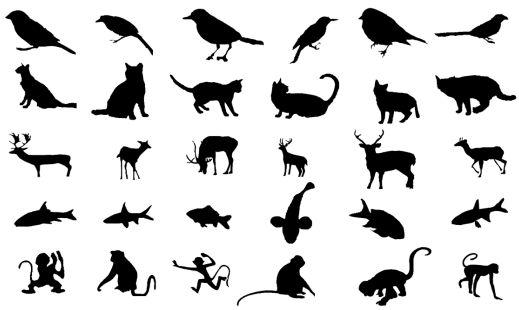

FIGURE 12. Some shapes for the five classes from the Animal dataset. The first, second, third, fourth and last row show six shapes from Bird, Cat, Deer, Fish, and Monkey class from this dataset, respectively.

where $Avg(.)$ takes the average value. SC_{TAR}, SC_{TCD} and SC_{TASL} are defined as

$$SC_{TAR} = \frac{1}{N}\sum_{n=1}^{N} |{}^{max}_{ts}\{TAR(n, ts)\} - {}^{min}_{ts}\{TAR(n, ts)\}|,$$

$$SC_{TCD} = \frac{1}{N}\sum_{n=1}^{N} |{}^{max}_{ts}\{TCD(n, ts)\} - {}^{min}_{ts}\{TCD(n, ts)\}|,$$

$$SC_{TASL} = \frac{1}{N}\sum_{n=1}^{N} |{}^{max}_{ts}\{TASL(n, ts)\} - {}^{min}_{ts}\{TASL(n, ts)\}|. \tag{17}$$

When the complexity of the contour is low, humans are usually more sensitive to the deformation of shape contour. The shape complexity here is defined as the average, over all sampling points, of the absolute difference between the maximum and minimum TAR, TCD, and TASL values at all scale levels. Then, we take their average value as the final shape complexity. First, we calculate the complexities of each shape for different sampling points N, which is shown in Fig. 13. We can see that the shape complexity of each shape becomes larger as the number of contour sampling points N increases. For different shapes, shape complexity is also different, which shows that the shape complexity is unique and can reflect the complexity of each shape. Next, we calculate the average complexity of each category, as shown in Fig. 14. We can see that the complexities of different categories are different, and their complexities also become larger as N increases.

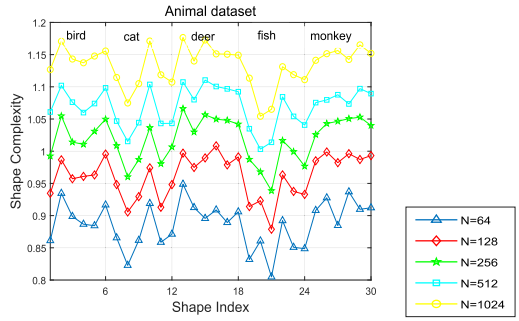

FIGURE 13. Shape complexities of 30 shapes for five classes at different sampling points value N.

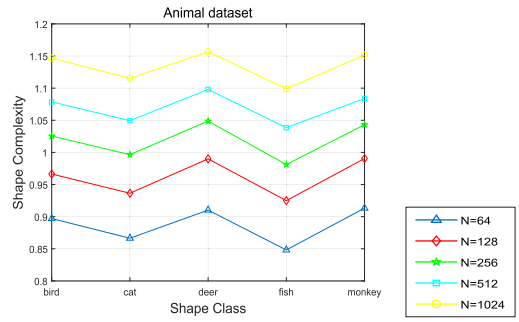

FIGURE 14. Shape complexities of five classes for different sampling points value N.

Finally, we analyze the relationship between different image sizes and shape complexity. Fig. 15 shows the bird image with different image sizes. Table 10 presents the shape complexities of different image sizes for different contour sampling points N. It can be seen that the shape complexities of different image sizes increase gradually with the increase of contour sampling points N. The shape complexities increase as the image size increases when $N = 64$, $N = 128$, and $N = 1024$. For $N = 256$ and $N = 512$, the shape complexities first increase and then decrease as the image size increases. At the same time, we can observe that the shape complexity remains stable as the image size increases when N gets increasingly larger.

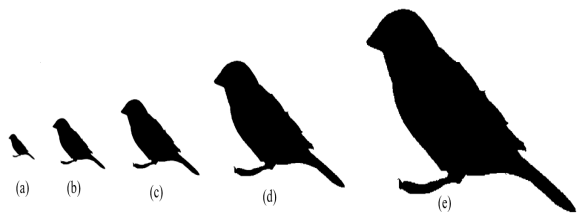

FIGURE 15. Bird image with different resolutions. (a) image size = [44,64]; (b) image size = [87,128]; (c) image size = [260,384]; (d) image size = [432,640]; (e) image size = [519,768].

We can see from the experimental results that the PCA method can sometimes improve the accuracy of shape classification. The possible reason is that the PCA method has

TABLE 10. Shape complexities of different image sizes *imgsize* at different contour sampling points *N*.

	$N = 64$	$N = 128$	$N = 256$	$N = 512$	$N = 1024$
$imgsize = [44, 64]$	0.8267	0.9176	0.9675	1.0299	1.1018
$imgsize = [87, 128]$	0.8420	0.9116	0.9844	1.0334	1.1158
$imgsize = [260, 384]$	0.8635	0.9286	0.9831	1.0488	1.1181
$imgsize = [432, 640]$	0.8612	0.9347	0.9925	1.0609	1.1268
$imgsize = [519, 768]$	0.8751	0.9373	0.9918	1.0436	1.1351

the ability of feature selection, which is enhanced by the SVM classifier. We may want to know whether we can select some distinguishing features through other feature selection methods. On the one hand, it can reduce the dimension of features and improve the efficiency of shape classification; on the other hand, it can eliminate some noise data and improve the accuracy of shape classification. Next, we adopted three feature selection methods, including two classic Relief-F [61] and Laplacian score [62] feature selection methods and the recently proposed infinite latent feature selection (ILFS) [63] method, to study the influence of different features on the performance of shape classification. We still use the Animal dataset for experimental analysis, and we select five feature dimensions: 128, 256, 512, 1024 and 2048. Table 11 shows the classification results of the three feature selection methods on the Animal dataset for five different feature dimensions.

TABLE 11. Classification results of the three feature selection methods on the Animal dataset for five different feature dimensions.

Method	$\hat{D} = 128$	$\hat{D} = 256$	$\hat{D} = 512$	$\hat{D} = 1024$	$\hat{D} = 2048$
Relief-F [61]	45.30%	57.08%	65.40%	70.67%	77.66%
Laplacian score [62]	50.79%	60.66%	67.21%	74.18%	80.23%
ILFS method [63]	40.91%	53.94%	62.09%	71.94%	79.83%

It can be seen from the experimental result that the Laplacian score method achieves the best classification accuracy among the three feature selection methods. The classification accuracy of the ILFS method is better than the classic Relief-F method when the feature dimension is greater than or equal to 1024. At the same time, the classification accuracy of the Relief-F method is better than the ILFS method when the feature dimension is less than or equal to 512. However, the classification accuracy of these three feature selection methods is far lower than that of the PCA method. The possible reason is that the PCA method obtains some better low-dimensional features through feature transformation, and these three feature selection methods select a number of features from high-dimensional data. Therefore, some important features will be lost when the selected feature dimensions are less, which affects the classification result. This experimental result shows that the PCA is a highly effective dimension reduction method.

H. OBJECT CLASSIFICATION ON REAL-WORLD IMAGES

All of the above experiments were performed on manually extracted shapes, and one might wonder how the proposed method detects and recognizes objects in real-world images. To explain this, we combined the Weizmann Horse [64], which as 328 horse images, with the ETHZ Cow dataset [51], which has 111 cow images, for a total of 439 images. We adopted an unsupervised game-theoretic saliency detection approach [65] to extract the foreground region for each image. Fig. 16 shows the saliency detection results of several images. The first two columns in Fig. 16 show several typical results, while the other columns show difficult images with severe segmentation errors.

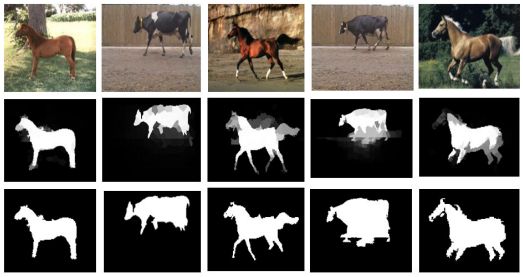

FIGURE 16. Foreground extraction results of test images from the Weizmann Horse and ETHZ Cow datasets. The first row shows several test images, the second row represents the salient detection results by the unsupervised game-theoretic method and the third row represents the binarization result of the salient maps. The first two columns indicate several typical results, and the other columns represent several unsatisfactory results of foreground extraction.

To evaluate the performance of the various shape classification methods, we randomly selected 60 images in each category to train the classification model and used the remaining images to test the model. We carried out 10 experiments and averaged them as to the classification accuracy. We compared the proposed method with the BCF method [22] and the multiscale Fourier descriptor method [45]. These two methods are representative methods of mid-level shape feature and low-level shape descriptor, and our method is based on multiscale triangular feature descriptor. The comparison results of these methods are shown in Table 12. We can see that our method achieves the highest classification accuracy of all the methods, including 10% higher than the BCF method [22] and 11% higher than the multiscale Fourier descriptor method [45]. It is worth noting that the addition of the TCD descriptor in this experiment can improve the overall classification. The possible reason for this is that the noise in foreground extraction has an effect on the dataset, but the overall classification performance of our method remains relatively stable. This experiment proves that our method can be applied to object classification of real-world images and is sufficiently robust to foreground extraction errors and local noisy boundaries.

TABLE 12. Classification results of the Weizmann Horse and ETHZ Cow datasets.

Method	Classification accuracy
Multiscale Fourier descriptor [45]	$82.48 \pm 2.05\%$
BCF [22]	$83.23 \pm 1.50\%$
Our Method(TAR+TASL)	$91.54 \pm 1.51\ \%$
Our Method(TCD+TAR+TASL)	$\mathbf{93.32 \pm 1.58}\ \%$

V. CONCLUSION

We have proposed a novel contour-based mid-level shape description method for shape classification. This method can deal with the intra-class variation and non-linear deformation of an object's shape, and enhance the distinguishing ability of the shape. Compared with the previous work, we offer three contributions. First, we regard each point of the shape boundary as a local feature, unlike previous methods using the contour fragment as a local feature. Second, we present an effective local feature description method named the triangular feature, which can describe the details and global characteristics of a shape and represent the spatial layout information between sampling points. This is why our method does not require the spatial pyramid matching (SPM) to append the spatial layout information. Finally, we build a mid-level shape descriptor with good discrimination performance based on the FV encoding method for shape classification. The performance of our method has been extensively evaluated on a large number of shape benchmarks, and the experimental results show that our approach exceeds existing state-of-the-art shape classification approaches. In future work, we will focus on how to apply this method to the recognition of target objects in complex scenes.

REFERENCES

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[2] H. Ling and D. W. Jacobs, "Shape classification using the inner-distance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 2, pp. 286–299, Feb. 2007.

[3] V. Ferrari, F. Jurie, and C. Schmid, "From images to shape models for object detection," *Int. J. Comput. Vis.*, vol. 87, no. 3, pp. 284–303, May 2010.

[4] Y. Su, Y. Liu, B. Cuan, and N. Zheng, "Contour guided hierarchical model for shape matching," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1609–1617.

[5] H. Wei, C. Yang, and Q. Yu, "Efficient graph-based search for object detection," *Inf. Sci.*, vols. 385–386, pp. 395–414, Apr. 2017.

[6] C. Yang, H. Wei, and Q. Yu, "A novel method for 2d nonrigid partial shape matching," *Neurocomputing*, vol. 275, pp. 1160–1175, Jan. 2018.

[7] L. da Fontoura Costa and R. M. Cesar, Jr., *Shape Classification and Analysis: Theory and Practice*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2009.

[8] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognit.*, vol. 37, no. 1, pp. 1–19, Jan. 2004.

[9] M. Yang, K. Kpalma, and J. Ronsin, "A survey of shape feature extraction techniques," *Pattern Recognit.*, vol. 15, no. 7, pp. 43–90, 2008.

[10] L. Kurnianggoro and K.-H. Jo, "A survey of 2D shape representation: Methods, evaluations, and future research directions," *Neurocomputing*, vol. 300, pp. 1–16, Jul. 2018.

[11] S. Abbasi, F. Mokhtarian, and J. Kittler, "Curvature scale space image in shape similarity retrieval," *Multimedia Syst.*, vol. 7, no. 6, pp. 467–476, Nov. 1999.

[12] P. F. Felzenszwalb and J. D. Schwartz, "Hierarchical matching of deformable shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[13] B. Wang and Y. Gao, "Hierarchical string cuts: A translation, rotation, scale, and mirror invariant descriptor for fast shape retrieval," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 4101–4111, Sep. 2014.

[14] C. Yang, H. Wei, and Q. Yu, "Multiscale triangular centroid distance for shape-based plant leaf recognition," in *Proc. Eur. Conf. Artif. Intell.*, 2016, pp. 269–276.

[15] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 489–497, May 1990.

[16] D. Zhang and G. Lu, "Shape-based image retrieval using generic Fourier descriptor," *Signal Process., Image Commun.*, vol. 17, no. 10, pp. 825–848, Nov. 2002.

[17] P.-T. Yap, X. Jiang, and A. C. Kot, "Two-dimensional polar harmonic transforms for invariant image representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1259–1270, Jul. 2010.

[18] B. Wang and Y. Gao, "Multi-scale bisector integrals: An invariant descriptor for accurate shape retrieval," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 765–769.

[19] F. Ghadiri, R. Bergevin, and G. A. Bilodeau, "Carried object detection based on an ensemble of contour exemplars," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 852–866.

[20] Q. Yu, H. Wei, and C. Yang, "Local part chamfer matching for shape-based object detection," *Pattern Recognit.*, vol. 65, pp. 82–96, May 2017.

[21] J. J. Lim, C. L. Zitnick, and P. Dollár, "Sketch tokens: A learned mid-level representation for contour and object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3158–3165.

[22] X. Wang, B. Feng, X. Bai, W. Liu, and L. Jan Latecki, "Bag of contour fragments for robust shape classification," *Pattern Recognit.*, vol. 47, no. 6, pp. 2116–2125, Jun. 2014.

[23] B. Ramesh, C. Xiang, and T. H. Lee, "Shape classification using invariant features and contextual information in the bag-of-words model," *Pattern Recognit.*, vol. 48, no. 3, pp. 894–906, Mar. 2015.

[24] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 143–156.

[25] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, Dec. 2013.

[26] T. B. Sebastian, P. N. Klein, and B. B. Kimia, "Recognition of shapes by editing shock graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 71–550, May 2004.

[27] K. B. Sun and B. J. Super, "Classification of contour shapes using class segment sets," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 727–733.

[28] M. R. Daliri and V. Torre, "Robust symbolic representation for shape recognition and retrieval," *Pattern Recognit.*, vol. 41, no. 5, pp. 1782–1798, May 2008.

[29] X. Bai, W. Liu, and Z. Tu, "Integrating contour and skeleton for shape classification," in *Proc. IEEE 12th Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Sep. 2009, pp. 360–367.

[30] D. Macrini, S. Dickinson, D. Fleet, and K. Siddiqi, "Object categorization using bone graphs," *Comput. Vis. Image Understand.*, vol. 115, no. 8, pp. 1187–1206, Aug. 2011.

[31] J. Wang, X. Bai, X. You, W. Liu, and L. J. Latecki, "Shape matching and classification using height functions," *Pattern Recognit. Lett.*, vol. 33, no. 2, pp. 134–143, Jan. 2012.

[32] Q. Ke and Y. Li, "Is rotation a nuisance in shape recognition?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4146–4153.

[33] M. Bicego and P. Lovato, "A bioinformatics approach to 2D shape classification," *Comput. Vis. Image Understand.*, vol. 145, pp. 59–69, Apr. 2016.

[34] C. Li, A. Stevens, C. Chen, Y. Pu, Z. Gan, and L. Carin, "Learning weight uncertainty with stochastic gradient MCMC for shape classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5666–5675.

[35] P. Govindaraj and M. S. Sudhakar, "Hexagonal grid based triangulated feature descriptor for shape retrieval," *Pattern Recognit. Lett.*, vol. 116, pp. 157–163, Dec. 2018.

[36] M. R. Daliri and V. Torre, "Shape recognition based on kernel-edit distance," *Comput. Vis. Image Understand.*, vol. 114, no. 10, pp. 1097–1103, Oct. 2010.

[37] W. Shen, C. Du, Y. Jiang, D. Zeng, and Z. Zhang, "Bag of shape features with a learned pooling function for shape recognition," *Pattern Recognit. Lett.*, vol. 106, pp. 33–40, Apr. 2018.

[38] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.

[39] F. Porikli, S. Shan, C. Snoek, R. Sukthankar, and X. Wang, "Deep learning for visual understanding: Part 2 [From the guest Editors]," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 17–19, Jan. 2018.

[40] Z.-H. Ling, S.-Y. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. M. Meng, and L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 35–52, May 2015.

[41] T. Tan, Y. Qian, H. Hu, Y. Zhou, W. Ding, and K. Yu, "Adaptive very deep convolutional residual network for noise robust speech recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 8, pp. 1393–1405, Aug. 2018.

[42] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [Review Article]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, Aug. 2018.

[43] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[44] S. M. A. Eslami, N. Heess, C. K. I. Williams, and J. Winn, "The shape Boltzmann machine: A strong model of object shape," *Int. J. Comput. Vis.*, vol. 107, no. 2, pp. 155–176, Apr. 2014.

[45] C. Yang and Q. Yu, "Multiscale Fourier descriptor based on triangular features for shape retrieval," *Signal Process., Image Commun.*, vol. 71, pp. 110–119, Feb. 2019.

[46] N. Alajlan, I. El Rube, M. S. Kamel, and G. Freeman, "Shape retrieval using triangle-area representation and dynamic space warping," *Pattern Recognit.*, vol. 40, no. 7, pp. 1911–1920, Jul. 2007.

[47] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2001.

[48] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[49] L. J. Latecki, R. Lakamper, and T. Eckhardt, "Shape descriptors for non-rigid shapes with a single closed contour," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2000, pp. 424–429.

[50] O. J. O. Söderkvist, "Computer vision classification of leaves from Swedish trees," M.S. thesis, Dept. Elect. Eng., Linköping Univ., Linköping, Sweden, Sep. 2001.

[51] B. Leibe and B. Schiele, "Analyzing appearance and contour based methods for object categorization," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2003, pp. II-409–II-415.

[52] K.-L. Lim and H. K. Galoogahi, "Shape classification using local and global features," in *Proc. 4th Pacific-Rim Symp. Image Video Technol. (PSIVT)*, Nov. 2010, pp. 115–120.

[53] C. Li, X. You, A. Ben Hamza, W. Zeng, and L. Zhou, "Distinctive parts for shape classification," in *Proc. Int. Conf. Wavelet Anal. Pattern Recognit.*, Jul. 2011, pp. 97–102.

[54] S. Jayasumana, M. Salzmann, H. Li, and M. Harandi, "A framework for shape analysis via Hilbert space embedding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1249–1256.

[55] E. Attalla and P. Siy, "Robust shape similarity retrieval based on contour segmentation polygonal multiresolution and elastic matching," *Pattern Recognit.*, vol. 38, no. 12, pp. 2229–2241, Dec. 2005.

[56] M. Daliri and V. Torre, "Shape recognition and retrieval using string of symbols," in *Proc. 5th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2006, pp. 101–108.

[57] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1489–1501, Aug. 2011.

[58] R. Hu, W. Jia, H. Ling, and D. Huang, "Multiscale distance matrix for fast plant leaf recognition," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4667–4672, Nov. 2012.

[59] R.-X. Hu, W. Jia, Y. Zhao, and J. Gui, "Perceptually motivated morphological strategies for shape retrieval," *Pattern Recognit.*, vol. 45, no. 9, pp. 3222–3230, Sep. 2012.

[60] C. Zhao, S. S. F. Chan, W.-K. Cham, and L. M. Chu, "Plant identification using leaf shapes—A pattern counting approach," *Pattern Recognit.*, vol. 48, no. 10, pp. 3203–3215, Oct. 2015.

[61] H. Liu and H. Motoda, *Computational Methods of Feature Selection*. Boca Raton, FL, USA: CRC Press, 2007.

[62] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 507–514.

[63] G. Roffo, S. Melzi, U. Castellani, and A. Vinciarelli, "Infinite latent feature selection: A probabilistic latent graph-based ranking approach," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1398–1406.

[64] E. Borenstein, E. Sharon, and S. Ullman, "Combining top-down and bottom-up segmentation," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, 2004, p. 46.

[65] Y. Zeng, M. Feng, H. Lu, G. Yang, and A. Borji, "An unsupervised game-theoretic approach to saliency detection," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4545–4554, Sep. 2018.

CHENGZHUAN YANG received the Ph.D. degree from the Department of Computer Science, Fudan University, China, in 2017. He is currently an Assistant Professor with the School of Information, Zhejiang University of Finance and Economics. His current research interests include computer vision, pattern recognition, and artificial intelligence.

LINCONG FANG received the Ph.D. degree in mathematics from Zhejiang University, Hangzhou, in 2009. He is currently an Associate Professor with the Zhejiang University of Finance and Economics. His research interests include geometric design and computation, geometric modeling, and interval arithmetic.

HUI WEI received the Ph.D. degree from the Department of Computer Science, Beijing University of Aeronautics and Astronautics, in 1998. From 1998 to 2000, he was a Postdoctoral Fellow with the Department of Computer Science and the Institute of Artificial Intelligence, Zhejiang University. In November 2000, he has joined the Department of Computer Science and Engineering, Fudan University. His research interests include artificial intelligence and cognitive science.