

Shape-based Object Retrieval and Classification with Supervised Optimisation

Cong Yang¹, Oliver Tiebe¹, Pit Pietsch², Christian Feinen¹, Udo Kelter² and Marcin Grzegorzec¹

¹Research Group for Pattern Recognition, University of Siegen, Siegen, Germany

²Software Engineering and Databases Group, University of Siegen, Siegen, Germany

Keywords: Object Retrieval, Object Classification, Optimisation, Shape Features.

Abstract: In order to enhance the performance of shape retrieval and classification, in this paper, we propose a novel shape descriptor with low computation complexity that can be easily fused with other meaningful descriptors like shape context, etc. This leads to a significant increase in descriptive power of original descriptors without adding to much computation complexity. To make the proposed shape descriptor more practical and general, a supervised optimisation strategy is introduced. The most significant scientific contributions of this paper includes the introduction of a new and simple feature descriptor with supervised optimisation strategy leading to the impressive improvement of the accuracy in object classification and retrieval scenario.

1 INTRODUCTION

Shape retrieval and classification are very important topics in computer vision. In order to improve the accuracy of shape matching a number of new shape descriptors have been proposed in recent years (Zhang and Lu, 2004) to effectively find perceptually similar shapes from a database. In addition, some learning methods (Bai et al., 2010) have been employed on the top of shape matching algorithms to involve more shape context information.

However, it is a difficult task to develop appropriate shape descriptors and matching algorithms. Firstly, a desirable shape descriptor should be invariant to shape rotation, translation and scaling. Though skeleton-based matching approaches (Goh, 2008; Hedrich et al., 2013; Bai and Latecki, 2008) can effectively classify shapes, most of them require heavy calculation for skeleton pruning (Bai et al., 2007) and for determining correspondences (Bai and Latecki, 2008). Secondly, for both shape descriptors and matching algorithms, there are many uncertain parameters that are involved in the feature generation and matching processes which are hard to be optimised by experiences among different datasets. Over the past decade or so, computer scientists have proposed many meaningful shape descriptors like Inner Distance (Ling and Jacobs, 2007), Shape Context (Belongie et al., 2002). However, in order to re-

duce the computing complexity, researchers usually employ only a partial number of points which are randomly selected or sampled with fixed distances along the shape contour. This strategy could easily lose critical features since some vertexes or partial deformations might be overlooked.

Motivated by the above mentioned problems, in this paper, we propose a new shape descriptor which involves both global and partial shape features with low computation complexity. Moreover, the proposed shape descriptor can be easily fused with other meaningful descriptors (Belongie et al., 2002; Bai and Latecki, 2008; Ling and Jacobs, 2007; Chang and Kimia, 2009; Siddiqi et al., 1998) to enhance the shape matching and classification performance without increasing too much the computation complexity. The contribution of this paper addresses as well the challenges mentioned above. Firstly, we introduce a new shape descriptor in form of a 10-dimensional feature vector. This shape descriptor integrates geometrical and topological features with low computation complexity and is robust to shape deformation. Secondly, we introduce a matching algorithm in which feature weights are optimised by a supervised optimisation strategy. This strategy can efficiently adopt our feature vector to diverse datasets. Experimental results demonstrate that the optimised SVM classifier obtains much better accuracy than the non-optimised one.

2 RELATED WORK

In this section, some existing approaches relating to shape descriptors for object retrieval and classification are discussed. We categorise them into two groups: contour-based descriptors and skeleton-based descriptors. The relevant matching algorithms are also introduced in this section.

2.1 Contour-based Shape Descriptors

Nguyen et al. (Nguyen et al., 2013) propose a shape-based local binary descriptor for object detection that has been tested in the task of detecting humans from static images. In (Cao et al., 2011), an algorithm for partial shape matching with mildly non-rigid deformations using Markov chains and the Monte Carlo method is introduced. Both of these methods do not involve context information for object matching. Shotton et al. (Shotton et al., 2005) present a categorical object detection scheme that uses only local contour-based features and is realised in a partly supervised learning framework. However, in this method many parameters need to be configured manually. Yang et al. (Yang et al., 2012) formulate the contour-based object detection as a matching problem between model contour parts and image edge fragments. They treat this problem as the task of finding dominant sets in weighted graphs. Though insensitive to noise and outliers, the approach is not rotation invariant. Different from the previous contour-based descriptors, our proposed descriptor combines both global and local shape features. Moreover, all features in our descriptor have corresponding weights which can be easily adapted to different datasets. Lastly, parameters in our method are automatically optimised by a supervised optimisation strategy.

2.2 Skeleton-based Shape Descriptors

Compared to contour-based descriptors, skeleton-based shape descriptors feature lower sensitivity to occlusion, limb growth, and articulation (Goh, 2008). However, they are computationally more complex (Sebastian and Kimia, 2001) and still have not been fully successfully applied to real images. Baseski et al. (Baseski et al., 2009) present a tree-edit-based shape matching method that uses a recent coarse skeleton representation. Their dissimilarity measure obtains a better result within groups than between group separation which mimics the asymmetric nature of human similarity judgements. To the best of our knowledge, the best performing skeleton-based object matching algorithm has been proposed

by Bai et al. (Bai and Latecki, 2008). Their main idea is to match skeleton graphs by comparing the geodesic paths between skeleton endpoints. Unfortunately, the performance of this method is limited to the presence of large protrusions, since they require skipping a large number of skeleton endpoints. Moreover, skeleton pruning and correspondence estimation increase the computation complexity of the method. Based on skeleton, Shock Graphs (Siddiqi et al., 1998) and Medial Scaffolds (Chang and Kimia, 2009) are proposed for shape matching. In this paper, skeleton is also used for our proposed feature generation. However, we only use skeleton length as one feature for global shape description and do not perform any pruning or correspondence estimation.

2.3 Matching Algorithms

Although Hausdorff distance (Mmoli, 2007) is one of the classical shape matching methods, we cannot directly use it since it is a correspondence-based method and has often been used to locate objects in an image. Shape contexts (Belongie et al., 2002; Bai et al., 2010) is an improvement to traditional Hausdorff distance based methods (Mmoli, 2007; Siddiqi et al., 1998; Del Bimbo and Pala, 1997). The matching of two shapes is done by matching two context maps of the shapes, which is a matrix-based matching (Belongie et al., 2002). However, considering the trade-off between accuracy and efficiency, involving matrix operations is too expensive for our experiment. Bimbo (Del Bimbo and Pala, 1997) proposed the use of elastic matching. This approach is not practical for on-line image retrieval, mainly because of the computation and matching complexity. In contrast to previous methods, in this paper, we design our matching algorithm with low complexity and fully exert merits of each feature in our feature space by individual weighting and global optimisation. Moreover, by supervised optimisation strategy on matching algorithm, our proposed shape descriptor can be easily adapted to different databases by selecting weights for each features.

3 OBJECT REPRESENTATION

Prior to feature extraction, we adjust the orientation of each object by rotating it to the point, that the straight line connecting its two maximally distant contour points becomes vertical and the majority of contour points lie on the left side of this line (See Figure 1). If the number of contour points on both sides of the line PP' are the same, we will adjust the orientation to the point, that the straight line connecting its two maxi-

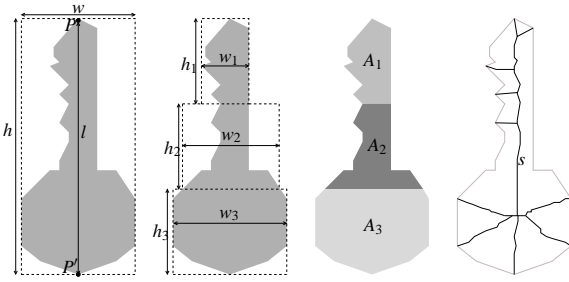


Figure 1: Shape bounding box and equally high sub-boxes ($h_1 = h_2 = h_3$) used for feature extraction; A_1 , A_2 , and A_3 are the areas of the top, middle, and bottom sub-objects, respectively.

mally distant contour points becomes vertical and the majority of contour points lies on the upper $h/2$ side. If the object is star-like or circle-like shape, we will select one straight line connecting its two maximally distant contour points and rotate the object so that the straight line becomes vertical.

An object shape is described by a 10-dimensional feature vector \mathbf{c}' . For this, we use the bounding box of the whole shape as well as the three equally high sub-boxes shown in Figure 1. Here we subdivide the bounding box into 3 equally high sub-boxes, this is based on the trade-off between configuration and fineness of subdivision. If we decompose the bounding box into more sub-boxes, each shape sub-component located in sub-boxes are tending to be similar. This could give rise to miss-corresponding during object matching process. Based on experiments, 3 sub-boxes selection achieves the best performance in terms of accuracy and robustness.

The first element c'_1 and the last element c'_{10} of the feature vector express the length of the object contour and the length of object skeleton s , respectively. The remaining elements are computed as follows:

$$c'_2 = \frac{h}{w}, \quad c'_3 = \frac{h_1}{w_1}, \quad c'_4 = \frac{h_2}{w_2}, \quad c'_5 = \frac{h_3}{w_3} \quad (1)$$

$$c'_6 = \frac{A_3}{A_1}, \quad c'_7 = \frac{A_2}{A_1}, \quad c'_8 = A_1 + A_2 + A_3, \quad c'_9 = l$$

Subsequently, we perform two feature normalisation steps. First, in order to ensure scale invariance, we divide the non-ratio elements of the feature vector by a half of the bounding box perimeter:

$$\mathbf{c}^* = \frac{\mathbf{c}'}{w+h} = (c_1^*, c_2^*, c_3^*, c_4^*, c_5^*, c_6^*, c_7^*, c_8^*, c_9^*, c_{10}^*)^T \quad (2)$$

Second, we linearly scale the feature values to the range $(0, 1]$:

$$\mathbf{c} = \frac{\mathbf{c}^* - \min\{c_1^*, \dots, c_{10}^*\} + 1}{\max\{c_1^*, \dots, c_{10}^*\} - \min\{c_1^*, \dots, c_{10}^*\} + 1} \quad (3)$$

In order to avoid the situation that $\mathbf{c} = 0$ and zero denominator, we add value 1 to both numerator and denominator. The scaling is needed for the Support Vector Machines applied in the classification step. The

main advantage of scaling is to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Another advantage is to avoid numerical difficulties during the calculation. Because kernel values usually depend on the inner products of feature vectors (e.g., the linear kernel and the polynomial kernel), large attribute values might cause numerical problems.

4 OBJECT RETRIEVAL AND CLASSIFICATION

In this section, we propose a similarity function on our feature vector for object retrieval. For object classification, we introduce the way for classifier building and kernel function selection. The supervised optimisation method will be described in the last part of this section.

4.1 Object Retrieval

In order to solve the object retrieval problem, we introduce a similarity measure between contours. Assume \mathbf{C}^* and \mathbf{C}° are two objects represented by our proposed feature vectors:

$$\begin{aligned} \mathbf{C}^* &= (c_1^*, c_2^*, \dots, c_n^*, \dots, c_{10}^*) \\ \mathbf{C}^\circ &= (c_1^\circ, c_2^\circ, \dots, c_k^\circ, \dots, c_{10}^\circ) \end{aligned} \quad (4)$$

Now, we introduce a dissimilarity measure for feature vectors belonging to different objects \mathbf{C}^* and \mathbf{C}° :

$$d(\mathbf{C}^*, \mathbf{C}^\circ) = \frac{1}{10} \sum_{m=1}^{10} \frac{\sigma_m |c_m^* - c_m^\circ|}{|c_m^* + c_m^\circ|}, \quad (5)$$

where σ_m is the weight for each feature achieved in an optimisation process explained later in this section. σ_m can be optimised to adapt the proposed feature vector to different datasets. Moreover, it helps the proposed feature to avoid the overfitting problem by applying a proper σ_m to different features. Our dissimilarity measure has been inspired by Chi-Square kernel (Hazewinkel, 2001), which comes from the Chi-Square distribution. Since our shape descriptor contains a bag of features that are discretely distributed and Chi-Square kernel can effectively model the overlap among them. The values of the dissimilarity function (5) belong to the range $d(\mathbf{C}^*, \mathbf{C}^\circ) \in [0, 1]$ which enables their easy conversion to similarity values:

$$s(\mathbf{C}^*, \mathbf{C}^\circ) = 1 - d(\mathbf{C}^*, \mathbf{C}^\circ) \quad (6)$$

4.2 Object Classification

In this experiment, we selected an SVM which extracts a decision boundary between shapes of differ-

ent classes based on the margin maximisation principle. Due to this principle, the generalisation error of the SVM is independent of the number of feature dimensions. Furthermore, a complex (non-linear) decision boundary can be extracted using a non-linear SVM. In this process, images in a high-dimensional feature space are mapped into a higher-dimensional feature space using a kernel trick. In this experiment, we choose Radial Basis Function (RBF) as kernel function for three reasons. Firstly, RBF kernel non-linearly maps samples into a higher dimensional space so, unlike the linear kernel, it can handle the case in which the relation between class labels and attributes is nonlinear. Secondly, The RBF kernel has less hyper parameters than the polynomial kernel which reduces the complexity of model selection. In our case, there are only two parameters (C, γ) that need to be determined and we optimise them by our proposed optimisation method. Thirdly, as the number of instances is much larger than the number of features, the RBF kernel has fewer numerical difficulties and leads to shorter training time.

In our work, we apply a multi-class Support Vector Machine (mSVM) using its one-against-one (1vs1) version which works with a voting strategy. It uses a two-class SVM for each pair from a set of all considered classes $\{\omega_1, \omega_2, \dots, \omega_K\}$. Thus, if there are K classes in total, $K(K-1)/2$ two-class classifiers have to be used. First, a sample pattern (query pattern) is classified using all these two-class SVMs. The final classification result is determined by counting to which class the sample pattern has been assigned most frequently.

4.3 Supervised Optimisation

The performance of matching and kernel functions in retrieval and classification systems is heavily dependent on the choice of appropriate parameters. These parameters are mutually dependent and therefore need to be optimised simultaneously. In practice, parameters are selected and optimised manually, based on the knowledge of experts. Obviously, this is an exhaustive and tedious process. In this section we propose the use of an effective, supervised optimisation strategy to automatically improve the quality of retrieval and classification systems.

Traditional optimisation methods use iterative strategies, which do not produce satisfactory results when applied to high dimensional problems. However, heuristic methods are well suited for such optimisation problems where multiple parameters have to be optimised simultaneously. In this paper we employ a combination of two heuristic optimisation meth-

ods: Gradient Hill Climbing (Russell and Norvig, 2009) integrated with Simulated Annealing (Kirkpatrick et al., 1983).

The Gradient Hill Climbing method starts with randomly selected parameters. Then it changes single parameters iteratively to find a better set of parameters. A fitness function then evaluates whether the new set of parameters performs better or worse. The Simulated Annealing strategy impacts the degree of the changes. In later iterations, the changes to the parameters are getting smaller. This strategy can efficiently reduce the computational complexity of our optimisation method.

By in- and decreasing all parameters separately with a specified magnitude that describes a convergent zero series, the gradient for maximum enhancement is computed. Adding this gradient to the previous parameters results in the parameters for the next iteration.

To use this heuristic strategy we have to define fitnessfunctions. A fitnessfunction evaluates the quality of a result for a set of given parameters. In case of classification systems the results are good when high classification rates are achieved. Hence, the fitnessfunction for our classification system simply computes the classification rate. The case is a bit more difficult for retrieval systems.

In our work we evaluate a retrieval result in two ways. One way is to calculate the bull's eye retrieval rate (see 5.1). The other way is to analyse the resulting similarity table for all possible query objects, and count how often the query class appears on every position. The result of this computation is then condensed to a single value s by application of the following methods:

The function f transforms the position of an object o in the similarity ranking of an object j to the index of object o . The indices are grouped by category, objects of the same category have following indices.

$$f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N} \quad (7)$$

Using f we can evaluate single lines of our ranking tables using the function $p(i, j, n)$ by taking i as index of query image, j as rank in ascending similarity tables and n as number of images per category. We count points for every object with the same class as the query object. To put a higher emphasis on the first positions of the ranking in the resulting score we give quadratic points for every right object.

$$p(i, j, n) = \begin{cases} j^2 & [i/n] = [f(i, j)/n] \\ 0 & [i/n] \neq [f(i, j)/n] \end{cases} \quad (8)$$

To calculate the final score an addition of all points in the last $2n$ lines is needed because the best $2n$ im-

Table 1: Retrieval results on MPEG-7 dataset. Results are summarised as the number of shapes from the same class among the first top 1-10 shapes. No Opt shows results from our matching algorithm without optimisation. CS Opt shows results using our matching algorithm with Graph Transduction optimisation. Our Opt shows results using our matching algorithm with supervised optimisation.

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
No Opt	700	647	600	567	521	488	447	426	405	342
GT Opt (Bai et al., 2010)	640	584	552	501	463	424	398	381	303	145
Our Method	700	657	615	591	553	518	475	467	420	363

ages are taken into consideration for bull's eye retrieval rate. Following is the corresponding formula:

$$s = \sum_{i=1}^n \sum_{j=nc-2n}^{nc} p(i, j, n) \quad (9)$$

where c denotes the number of classes.

5 EXPERIMENTS AND RESULTS

To evaluate the performance of the proposed method, we have performed experiments in an object retrieval and a classification scenario using four different datasets. Moreover, we also conducted an experiment with a fused descriptor to evaluate the improvement of existing shape descriptors that are fused with our proposed method.

5.1 Shape-based Object Retrieval

First, we illustrate our proposed algorithm for object retrieval on the MPEG-7 dataset, which has $70 \times 20 = 1,400$ shapes. In order to optimise the parameters involved in our matching algorithm, we randomly select 10 objects from each category, there are in total $10 \times 70 = 700$ objects used for supervised optimisation. After optimising all parameters, we employ the remaining 700 objects for testing. Table 1 shows the retrieval results with and without supervised optimisation process.

We employ Retrieval Rate for results comparison. The retrieval rate is measured by the so-called bulls-eye score. Every shape in the database is compared to all other shapes, and the number of shapes from the same class among the 20 most similar shapes is reported. The bulls-eye scores from other references are on all 1400 images, while here we are on 700 images. Therefore, in order to ensure the correctness of our result, we iteratively do the train/test split multiple times and an average value is reported. The proposed supervised optimisation with our matching algorithm achieves 94.0% retrieval rate on MPEG-7 dataset.

Although Michael (Donoser and Bischof, 2013) has already achieved a 100% bullseys score on

MPEG-7 dataset, the purpose of their method is different from ours. In this paper, we propose a simple and effective shape descriptor that can be fused with other shape descriptors. In order to make our descriptor more flexible and equip it with a higher adaptiveness, a supervised optimisation strategy is introduced. We do not involve any manifold structure defined by pairwise affinity matrices. On the contrary, Graph Transduction (Bai et al., 2010) and diffusion (Donoser and Bischof, 2013) can be used after our method to improve subsequent applications like retrieval.

Our algorithm achieves 94.0% retrieval rate on MPEG-7 dataset. This suggests that the MPEG-7 might not be sufficient to judge the quality of our proposed method. Therefore, we conducted another set of evaluation on dataset MPEG-400 which is a subset of the MPEG-7 collection, consisting of 400 objects categorised in 20 classes (second row in Figure 2). These shapes have much larger intra-class variations and inter-class similarities than the MPEG-7 dataset. We performed a comparison to the algorithm using contour segments corresponding. Since this database is the subset of MPEG-7, in this experiment, we employ the same parameter values from the first experiment, and implement the object retrieval process on MPEG-400. To evaluate the behaviour of our proposed feature space and matching algorithm, we apply the experiment in both with and without any optimisation. Experiment results show that our proposed method performs better than contour segment. Moreover, from Table 2, we can clearly observe that our proposed matching algorithm with supervised optimisation made a significant progress on MPEG-400.

In order to proof applicability of our approach to enhance the performance of existing shape descriptors, we conducted a set of evaluations on Kimia-216 and MPEG-400 dataset. We first perform the experiment only with Shape Context and then the fused descriptor between Shape Context (fused weight: 0.7) and proposed feature vector (fused weight: 0.3). The fused weights are also learned from our supervised optimisation method. Table 4 shows that the fused descriptor achieves significant progress on both databases.

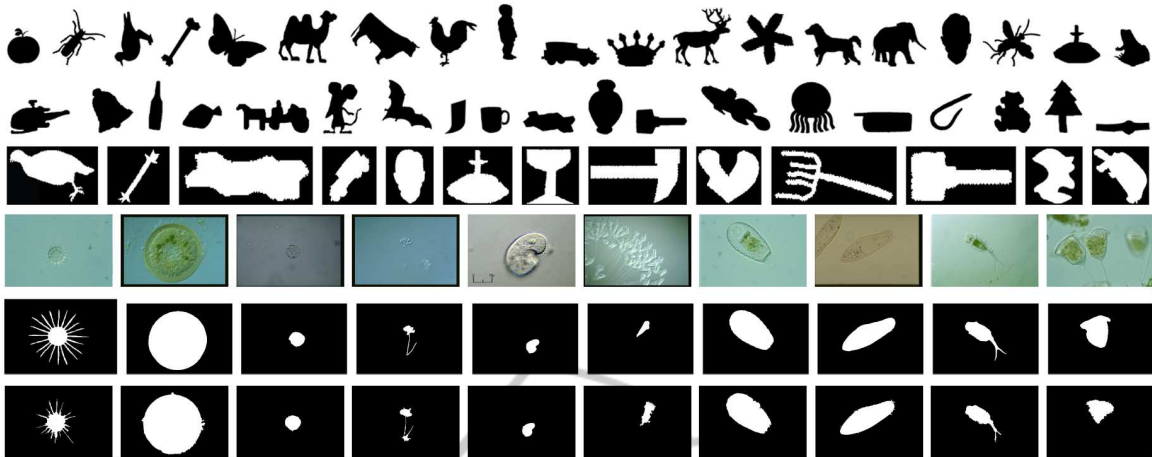


Figure 2: Example shapes from the experimental datasets: MPEG-7 (first row), MPEG-400 (second row), Kimia-216 (third row) and EM-200 (fourth row with original, fifth row with manually- and sixth row with semi-automatically segmented).

Table 2: Object retrieval on MPEG400 dataset. Our Method (No-Opt) represents the retrieval results using our matching algorithm with default parameters; Our Method (Opt) shows the retrieval results using our matching algorithm with supervised optimised parameters.

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Contour Segments (Yang et al., 2014b)	375	348	333	325	317	311	300	295	276	275
Our Method (No-Opt)	381	355	341	320	322	316	304	295	269	260
Our Method (Opt)	381	370	365	354	337	342	328	315	300	301

5.2 Shape-based Object Classification

In this part, we implement the experiment of classification on MPEG-7. As already discussed in Section 4, here we use Radial Basis Function (RBF) as the kernel function for support vector machine. Like in the previous experiment, we randomly split the shapes into half for training and half for testing. During the training phase, we are not only able to build classifiers with training data, but also seeking for best parameters with optimisation strategy. All of these tasks are done by 700 training images. The training set includes all the object classes. Another 700 testing objects are only used for testing. In this experiment, there are two parameters that need to be optimised, C for cost of SVM and γ in RBF kernel function. The default values are $C = 1$ and $\gamma = 1/n$, where n is the number of features. In our case, the default values are $n = 10$ and $\gamma = 0.1$. As shown in Table 3, in order to increase statistical relevance, we repeated the selection process 10 times which led to 10 different training datasets and corresponding testing datasets. Each column refers to an experiment with different training datasets. Experiments are performed for all these datasets and mean classification rates are reported. SSDP shows the results of scaled datasets with default parameters. Results shows that

our method achieves significant progress in this experiment. Actually, SVM and RBF kernel are highly related to parameters selection and our method can improve its performance sufficiently.

We also compare our optimised classification rate to existing methods with the MPEG-7 dataset. As shown in Table 5, our method yields a promising score compared to existing methods. The Skeleton Path achieves similar results as ours. However, the computation complexity of our feature vector is much lower than that of the skeleton path which needs to involve skeleton pruning for feature generation. Moreover, in (Bai et al., 2009), fused contour segment and skeleton path achieved 96.6% classification rate, but it is meaningless to compare it to our result since our descriptor is isolated.

5.3 EM Classification

To validate the idea of our proposed method for existing application, we proved the applicability of our new method to a real-world problem, namely the automatic classification of Environmental Microorganisms (EMs). EMs and their species are very important indicators to evaluate environmental quality, but their manual classification is very time-consuming (Li et al., 2013). Thus, automatic analysis techniques

Table 3: Object classification for the whole MPEG-7 dataset. SSDP represents the results of scaled datasets with default parameters. Our Method achieves significant progress in this experiment.

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	Average
SSDP(%)	47.6	45.6	50.7	47.6	46	44.4	46.4	46.0	45.9	44.9	46.5
Our Method(%)	85.7	86.0	85.9	85.7	84.1	86.6	85.1	87.7	87.9	88.0	86.3

Table 4: Experimental comparison of Shape Context descriptor to its fused descriptor with our method using the Kimia-216 and the MPEG-400 datasets as well as the proof of applicability of our approach to enhance the performance of existing shape descriptors. Results are summarised as the number of shapes from the same class among the first top 1-10 shapes.

Retrieval Results for Kimia-216	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Shape Context (Belongie et al., 2002)	216	212	201	187	186	175	171	164	162	146
Fused Method	216	214	207	204	201	204	191	188	192	185

Retrieval Results for MPEG-400	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Shape Context (Belongie et al., 2002)	400	370	343	310	302	277	272	265	264	239
Fused Method	400	389	374	368	368	358	347	344	339	346

for microscopic images of EMs would be very appreciated by environmental scientists. We have tested our methodology for the application using the EM-200 dataset. Since some EM-200 objects can hardly be skeletonised (e.g., the first two objects in the third row of Figure 2), Chen (Li et al., 2013) and Cong (Yang et al., 2014a) proposed some methods using the whole microorganism contours for object description. Here we employ the same feature vector proposed by Cong (Yang et al., 2014a), but classifiers will be built with our optimised C and γ . The impressive results for the EM-200 dataset (see Table 6) confirm the power of our supervised optimisation method to real-world applications.

5.4 Feature Analysis

In this part, we will employ Discriminate Analysis (DA) to evaluate the discriminating properties of the feature space. Especially, we applied Fisher Linear Discriminant Analysis (Viola and Jones, 2004) on MPEG-7 dataset and achieved the following results: $\lambda(c'_1) = 0.250$, $\lambda(c'_2) = 0.147$, $\lambda(c'_3) = 0.237$, $\lambda(c'_4) = 0.173$, $\lambda(c'_5) = 0.232$, $\lambda(c'_6) = 0.280$, $\lambda(c'_7) = 0.360$, $\lambda(c'_8) = 0.282$, $\lambda(c'_9) = 0.102$, $\lambda(c'_{10}) = 0.413$ for the different dimensions of the feature space, respectively. The value $\lambda(c'_i)$ expresses the overall discrimination power for the feature space dimension c'_i and is calculated as the ratio of the determinant of intra-class covariance matrix to the determinant of the total covariance matrix. The value of $\lambda(c'_i)$ belongs to the range $[0,1]$, whereas 0 corresponds to perfect discriminative properties and 1 denotes no discrimination. According to the analysis of the feature

Table 5: Experimental comparison of our methodology to the related algorithm for classification on MPEG-7.

Method	Score
Skeleton Path (Bai et al., 2009)	86.7%
CS (Sun and Super, 2005)	75.4%
IDSC (Ling and Jacobs, 2007)	76.5%
Our Method	86.3%

Table 6: Experimental comparison of our method to related algorithms for classification of Microorganisms as the proof of applicability of our optimisation approach to real world problems using the EM-200 dataset. (MS: Manually Segmented, SAS: Semi-Automatically Segmented).

Method	MS	SAS
SFSVM (Li et al., 2013)	89.7%	66.0%
NSFSVM (Yang et al., 2014a)	92.5%	79.5%
Our Method	95.0%	83.5%

space described above, the last feature c'_9 possesses the strongest discriminative power.

6 CONCLUSION

In this paper, we propose a simple and effective shape descriptor that can be easily fused with other descriptors. In order to make our descriptor more flexible and equip it with a higher adaptiveness, a supervised optimisation strategy is introduced. The proposed method can easily adapt to a concrete appli-

cation domain by optimising parameters assigned to different dimensions of the feature space and kernel function. Its promising performance has been proven in a meaningful experimental set-up. In the future, we will investigate possibilities of fusing more meaningful shape features into our feature space.

ACKNOWLEDGEMENTS

Research activities leading to this work have been supported by the China Scholarship Council (CSC) and the German Research Foundation within the Research Training Group 1564 (GRK 1564). We greatly thank M.Sc. Chen Li from University of Siegen and Prof. Dr. Beihai Zhou and M.Sc. Fangshu Ma from the University of Science and Technology Beijing for providing us with Environmental Microorganism image dataset for experiments.

REFERENCES

- Bai, X. and Latecki, L. (2008). Path similarity skeleton graph matching. *PAMI*, 30(7):1282–1292.
- Bai, X., Latecki, L., and Yu Liu, W. (2007). Skeleton pruning by contour partitioning with discrete curve evolution. *PAMI*, 29(3):449–462.
- Bai, X., Liu, W., and Tu, Z. (2009). Integrating contour and skeleton for shape classification. In *ICCV Workshops*, pages 360–367.
- Bai, X., Yang, X., Latecki, L., Liu, W., and Tu, Z. (2010). Learning context-sensitive shape similarity by graph transduction. *PAMI*, 32(5):861–874.
- Baseski, E., Erdem, A., and Tari, S. (2009). Dissimilarity between two skeletal trees in a context. *Pattern Recognition*, 42(3):370–385.
- Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522.
- Cao, Y., Zhang, Z., Czogiel, I., Dryden, I., and Wang, S. (2011). 2d nonrigid partial shape matching using mcmc and contour subdivision. In *CVPR*, pages 2345–2352.
- Chang, M.-C. and Kimia, B. (2009). Measuring 3d shape similarity by matching the medial scaffolds. In *ICCV*, pages 1473–1480.
- Del Bimbo, A. and Pala, P. (1997). Visual image retrieval by elastic matching of user sketches. *PAMI*, 19(2):121–132.
- Donoser, M. and Bischof, H. (2013). Diffusion processes for retrieval revisited. In *CVPR*, pages 1320–1327.
- Goh, W.-B. (2008). Strategies for shape matching using skeletons. *CVIU*, 110(3):326–345.
- Hazewinkel, M. (2001). *Chi-squared Distribution*. Encyclopedia of Mathematics, Springer.
- Hedrich, J., Yang, C., Feinen, C., Schaefer, S., Paulus, D., and Grzegorzec, M. (2013). Extended investigations on skeleton graph matching for object recognition. In *ICCRS*, pages 371–381. Springer LNCS.
- Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, pages 671–680.
- Li, C., Shirahama, K., Grzegorzec, M., Ma, F., and Zhou, B. (2013). Classification of environmental microorganisms in microscopic images using shape features and support vector machines. In *ICIP*, pages 2435–2439. IEEE Computer Society.
- Ling, H. and Jacobs, D. (2007). Shape classification using the inner-distance. *PAMI*, 29(2):286–299.
- Mmoli, F. (2007). On the use of gromov-hausdorff distances for shape comparison. In *SPBG*, pages 81–90.
- Nguyen, D. T., Ogunbona, P. O., and Li, W. (2013). A novel shape-based non-redundant local binary pattern descriptor for object detection. *Pattern Recognition*, 46(5):1485–1500.
- Russell, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, 3rd edition.
- Sebastian, T. and Kimia, B. (2001). Curves vs skeletons in object recognition. In *ICIP*, volume 3, pages 22–25.
- Shotton, J., Blake, A., and Cipolla, R. (2005). Contour-based learning for object detection. In *ICCV*, volume 1, pages 503–510.
- Siddiqi, K., Shokoufandeh, A., Dickenson, S., and Zucker, S. (1998). Shock graphs and shape matching. In *ICCV*, pages 222–229.
- Sun, K. and Super, B. (2005). Classification of contour shapes using class segment sets. In *CVPR 2005*, volume 2, pages 727–733.
- Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154.
- Yang, C., Li, C., Tiebe, O., Shirahama, K., and Grzegorzec, M. (2014a). Shape-based classification of environmental microorganisms. In *ICPR*, pages 3374–3379.
- Yang, C., Tiebe, O., Pietsch, P., Feinen, C., Kelter, U., and Grzegorzec, M. (2014b). Shape-based object retrieval by contour segment matching. In *ICIP*, pages 2202–2206.
- Yang, X., Liu, H., and Latecki, L. J. (2012). Contour-based object detection as dominant set computation. *Pattern Recognition*, 45(5):1927–1936.
- Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition*, 37:1–19.