A Machine Learning Approach to 3D Model Retrieval

Kuan-Hsien Liu, Pei-Ying Chiang and C.-C. Jay Kuo
Ming Hsieh Department of Electrical Engineering and Signal and Image Processing Institute
University of Southern California, Los Angeles, CA 90089, USA
E-mail: liuk@usc.edu, peiyingc@usc.edu, cckuo@sipi.usc.edu

***Abstract*— A novel 3D model retrieval system based on the machine learning approach is proposed in this work. This approach can be integrated with any existing 3D model matching algorithm which includes 3D model feature extraction and distance computation. By calculating the variance of each component among all feature vectors and removing those components with a larger variance, we can reduce the feature dimension. Furthermore, we derive the SVM classifier based on features extracted from the training set. We conduct experiments using the McGill Articulated Shape Benchmark database [1] for 3D model classification and retrieval, and demonstrate a significant performance improvement in the precision-recall curves.**

I. Introduction

The number of 3D models has increased rapidly in the last decade. A large amount of research has been conducted on the development of an automatic 3D model retrieval system with a focus on retrieval accuracy characterized by precision and recall.

One of the key ingredients in a 3D model retrieval system is *feature extraction*. To be qualified as a good shape feature, it should possess high discriminant power and be invariant to various transformations. Generally speaking, shape features of 3D models can be categorized into several types; namely, statistics-based [2], visual-similarity-based [3][4], transform-based [5], and skeleton-based [6] methods.

The retrieval system computes the distance between any pair of shape features. If a pair of models is similar, the feature distance will be smaller, too. Hence, for a given 3D model, we can retrieve similar 3D models by computing the distance of its features and those of 3D models in the database. This is known as "content-based 3D model retrieval". We refer to [7]-[9] for detailed survey on this research topic.

There has been little work on feature dimension reduction in the context of content-based 3D model retrieval. One reason could be that the number of 3D models is large and it would be difficult to select a subset of features that are much more important than others a priori. In this work, we would like to address this problem from a new angel. That is, for a given set of models, we adopt a machine learning approach to learn the classifier in the training stage. Then, we use the obtained classifier to retrieve models in the test stage. There are two major advantages with the proposed approach: 1) lower complexity in the test stage and 2) better retrieval performance in terms of the precision-recall tradeoff.

The rest of this paper is organized as follows. We first review several well-known shape features for 3D model retrieval in Section II. The proposed class-dependent feature learning and selection approach is described in Section III. Experimental results are reported and performance evaluation is conducted in Section IV. Finally, concluding remarks and future research topics are given in Section V.

II. Shape Features Extraction and Distance Computation for 3D Model Matching

In this section, we review features and their distance computation used in 3D model matching and retrieval.

- D2 (Distance between 2 random points) [2]. It is a method that measures the histogram of Euclidean distances between pairs of randomly selected points on the surface of a 3D model. The number of histogram bins is chosen as 1024 so that the length of a D2 feature vector is 1024.
- LFD (Light Field Descriptor) [3]. The light field cameras are put on 20 different views uniformly distributed over a 3D model. Since the silhouettes projected from two opposite vertices are identical, 10 different silhouettes are produced for a 3D model. To be robust against rotations among 3D models, a set of 10 LFDs is applied to each 3D model. Therefore, it is a method that represents a 3D model by 100 silhouettes (10 views per group) rendered from uniformly distributed viewpoints over a hemisphere and the silhouette is encoded by a feature vector with 47 entries including 35 Zernike moments, 10 Fourier coefficients, 1 eccentricity and 1 compactness. The length of an LFD feature vector is 4700. For any 3D model, even a simple one, 10 descriptors are created, and 10 silhouettes are represented for 20 viewpoints in each descriptor. Therefore, a total of 100 silhouettes will be rendered and the length of an LFD feature vector for any 3D model is 4700.
- SHD (Spherical Harmonic Descriptor) [5]. It is a method that describes a 3D model as a feature vector consisting of spherical harmonic coefficients, which are extracted from three spherical functions giving the maximal distance from the center of mass as a function of a spherical angle. The length of a SHD feature vector is 544.

TABLE I

SUMMARY OF FEATURE VECTOR DIMENSIONS

Feature	D2	LFD	SHD	PS	AAD	mSPRH
Length (L)	1024	4700	544	567	256	625

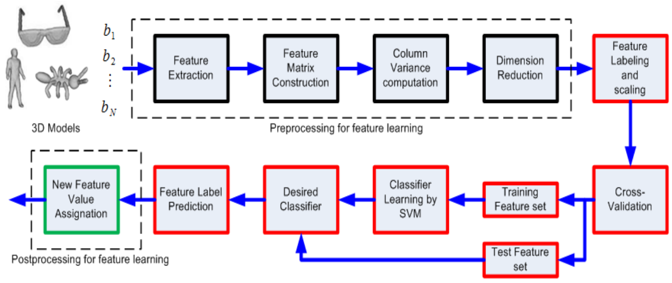

Fig. 1 The block-diagram of the proposed 3D model retrieval system.

- PS (linearly Parameterized Statistics) [10]. It is a method that uses a combination of three vectors (*i.e.,* the moment of inertia, the average distance of surfaces from the axis, and the variance of distances of surfaces from the axis.) Values in each vector are discretely parameterized along each of the three principal axes of inertia of the 3D model. The length of a PS feature vector is 567.
- AAD (Absolute Angle Distance) [11]. It is a method that computes the features by first converting a surface-based input model into an oriented point-set model and then computing joint 2D histogram of distance and orientation of pairs of points. The length of an AAD feature vector is 256.
- SPRH (Surflet-Pair-Relation Histograms) [12]. It uses the modified SPRH [13] to extract features of a 3D model. The length of a modified SPRH (mSPRH) feature vector is 625.

The dimensions of feature vectors discussed in above are summarized in Table I. After obtaining shape features of 3D models, we will analyze these features and select the subset of most discriminant features for a given class of models automatically using a machine learning approach as described in the following section.

III. FEATURE DIMENSION REDUCTION AND SVM CLASSIFIER TRAINING

The block-diagram of the proposed 3D model retrieval system is shown in Fig. 1. It consists of the following three main modules:

1. Pre-processing for feature dimension reduction;
2. Feature learning via support vector machine (SVM);
3. Post-processing of learned features.

They will be detailed in the following sub-sections.

A. *Feature Dimension Reduction*

We can adopt any feature extraction method as described in Section 2. Once the features of 3D models are extracted, we conduct the preprocessing task to reduce the complexity of task performed in the later stages. It consists of the following 4 steps.

1. Feature Extraction. Use any feature selection method to obtain a feature vector for a 3D model.
2. Feature Matrix Construction. Build a feature matrix that has dimension M x L, where M is the number of 3D models of consideration and L is the length of feature vectors. We have M=255 and L equal to a value in Table 1 (depending on the method to be selected) in our experiment. Each row provides a feature vector of a 3D model.
3. Column Variance Computation. Calculate the variance for each column in the feature matrix.
4. Feature Dimension Reduction. Discard columns with larger variances. The number of columns to be eliminated depends on the feature selection method. We observe that 5-25% of columns having larger variances can be discarded without loss of accuracy and even with an increase in accuracy.

The rationale of Step 4 in above can be intuitively explained as follows. Each column provides some information to differentiate 3D models, and we can treat each column as an estimator. Based on the following Cramer-Rao inequality, we have

$$var \ge 1/J \ , \qquad (1)$$

where J is the Fisher information and var denotes the estimator-variance. For those columns of a larger variance value, they do not provide useful information for estimation. They may even have a negative impact on the estimation result. Thus, we can discard columns with a larger variance to lower the complexity and improve the estimation accuracy.

B. *Training of SVM Classifier*

In this module, we would like to explain how to get the SVM classifier from the training data. The training, testing, and cross-validation steps are described as follows.

1. Feature Vector Labeling. Label each feature vector, which is a row of the feature matrix, with value *i* if the model belongs to class *i*.
2. Linear Scaling. Linearly scale training and testing data. Every entry in a feature vector is a sub-feature. We scale each column linearly to range [0, 1]. This is conducted to avoid the dominance of attributes with a large dynamic range over those with a smaller dynamic range.
3. N-Fold Cross-Validation. We divide the entire database of 3D models N subsets of equal size (or nearly equal size) where each subset consists of about the same number of 3D models from each class. Then, we choose 1 subset as the testing set while using the other N-1 subsets as the training set. This process is repeated for N times where each subset is used as the testing set once. The technique, called the N-fold cross-validation, is

employed to average the testing results and increase the confidence level.

4. Kernel Selection. Given a training set of feature-label pairs ($\mathbf{x}_i$, y_i), i = 1, …, l, where $\mathbf{x}_i \in R^n$ and $y_i \in \{1, -1\}^l$, the SVM requires the solution of the following optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i$$
$$\text{subject to } \begin{cases} y_i(w^T \varphi(\mathbf{x}_i) + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases}, \quad (2)$$

Here the training feature vectors $\mathbf{x}_i$ are mapped into a higher dimensional space by function φ. Furthermore, $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)$ is called the kernel function. Two commonly used kernel functions are

- Linear:
$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i{}^T \mathbf{x}_j$$
- Radial basis function:
$$K(\mathbf{x}_i, \mathbf{x}_j) = exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right), \gamma > 0$$

where γ is the kernel parameter.

The radial basis function (RBF) kernel is often used as the kernel when the dimension of the feature vector is low. On the other hand, if the dimension of the feature vector is high, which is our current case, the nonlinear mapping does not improve the performance much. Thus, we choose the linear kernel for the SVM algorithm in our experiment.

C. 3D Model Classification

Every 3D model needs an index value to represent itself. Here, we assign a 3D model with the following new index:

$$F = i^* \cdot 10 + r, \quad (3)$$

where i^* is the new class index number and r is a random number in the unit interval (0,1). The reason to multiply the new class index i^* by 10 is to separate the index value of 3D models in different classes. We can view F as one additional feature of the K-SVM method, where K could be any of D2, LFD, SHD, PS, AAD and mSPRH in our experiments.

When each 3D model is predicted with a new class index i^*, the classification accuracy can be determined by comparing the predicted new class index and its ground-truth. If $i = i^*$, it means that this is a correct classification result. Otherwise, it is a wrong classification result. The distance matrix can then be constructed by calculating the distance between every pair of 3D models' new feature F. Thus, we can plot performance curves such as the precision-recall (P-R) curves accordingly.

IV. Experimental Results

We choose the McGill Articulated 3D model database, which contains 255 models with 10 classes in our experiments. The feature matrix has a size of 255 x L, where L is the length of the feature vector. In the training of SVM classifiers, we use the LIBSVM library [14]. In the cross-validation step, we divide the entire set of 3D models into N = 5 subsets. One subset is sequentially tested using the classifier trained based on the remaining 4 subsets. We will report the 3D model classification and retrieval performance in Sec. IV.A and Sec. IV.B, respectively.

TABLE II
ACCURACY AND COMPLEXITY OF 3D MODEL RETRIEVAL

feature type	Original feature matrix		Dimension-Reduced feature matrix		
	no. of columns	accuracy	no. of columns	accuracy	saved training time
D2	1024	79.6078% (203/255)	768	82.3529% (210/255)	25%
LFD	4700	89.0196% (227/255)	3760	89.4118% (228/255)	20%
SHD	544	89.4118% (228/255)	462	90.5882% (231/255)	15%
PS	567	62.3529% (159/255)	482	62.7451% (160/255)	15%
AAD	256	92.9412% (237/255)	233	93.3333% (238/255)	9%
mSPRH	625	7.84314% (20/255)	562	93.3333% (238/255)	10%

A. Classification Performance

First, we study the performance of 3D model classification and compare the accuracy and complexity. We implement the 6 methods as described in Section II. Furthermore, we implement the proposed dimension-reduced features as well as the SVM-based feature training and testing process in association with each method. In the feature dimension reduction process, we discard columns having the largest variance values gradually and find the best classification accuracy with respect to the number of columns. Thus, it is a result obtained from exhaustive search. The results are showed in Table II.

As shown in Table II, we see that the use of the dimension-reduced feature to learn a 3D model classifier improves the classification accuracy as well as reduces the training and testing complexity.

B. Retrieval Performance

Next, we examine the performance of content-based 3D model retrieval. The precision-recall plot is a common tool in evaluating the retrieval performance. For each query model in class i and any number N of top matches, "recall" and "precision" are defined as [7]:

$$recall = \frac{models\ in\ class\ i\ returned\ within\ the\ top\ N\ matches}{number\ of\ models\ in\ class\ i}$$

$$precision = \frac{the\ top\ N\ matches\ that\ are\ members\ of\ class\ i}{the\ top\ N\ matches}$$

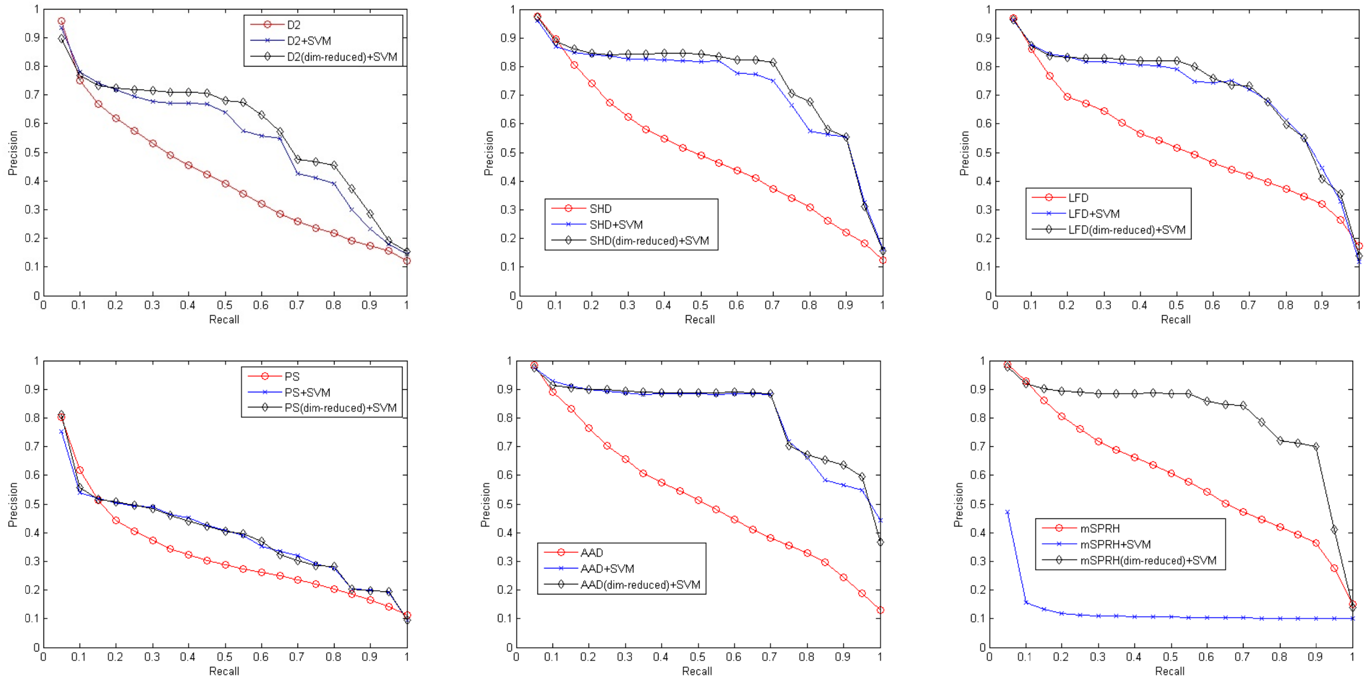

Fig. 2 Comparison of the precision-recall curves for three methods (original feature sets, original feature sets with SVM training, and original feature set with feature dimention reduction and SVM training, where the 6 different feature sets are D2, SHD, LFD (from left to right of the top row) and PS, ADD and mSPRH (from left to right of the bottom row).

A perfect retrieval result will give a horizontal line across the top of the plot (with precision = 1). Thus, a curve that lies more towards the upper right position indicates a better retrieval performance. We compare the retrieval performance in terms of the precision-recall curves for each of the 6 methods described in Section II with three variants:

1) the original method (K);
2) its improved version by incorporating SVM (K+SVM);
3) its improved version by incorporating feature dimension reduction and SVM (K+FDR+SVM).

The results are shown in Fig. 2, where the 6 original methods are D2, SHD, and LFD (from left to right of the top row) and PS, ADD and mSPRH (from left to right of the bottom row).

As compared with the original method, we observe a remarkable improvement in the retrieval performance by incorporating SVM and/or joint FDR/SVM. It is also interesting to point out that, although the use of either SVM alone or joint FDR/SVM offers similar improvement for most methods, it demands joint FDR/SVM to achieve performance improvement for mSPRH. Since some of the mSPRH features for 255 models from different classes are similar, it affects the training of the SVM classifier. This could be the reason why the performance of mSPRH+SVM is much worse than that of mSPRH. With the FDR in mSPRH, the similarities between those similar features will be greatly reduced, and hence the mSPRH+FDR+SVM performs much better than mSPRH.

V. CONCLUSION

An SVM-based 3D model retrieval system was proposed in this work. This system can analyze the feature set provided by other methods and reduce its dimension based on the idea of increasing the Fisher information. The SVM algorithm is used to train a classifier and a cross-validation technique is employed to increase prediction reliability. Experimental results were given to demonstrate the superior performance of the proposed approach. Since the classification technique is built upon a well-trained classifier, it can possess much better discrimination ability, which is verified in the precision-recall plots.

REFERENCES

[1] J. Zhang, R. Kaplow, R. Chen, and K. Siddiqi, "McGill 3D Shape Benchmark," http://www.cim.mcgill.ca/shape/benchMark/, 2005.

[2] R. Osada, T. Funkhouser, B Chazelle, and D Dobkin, "Shape distributions," ACM TOG, vol. 21, no. 4, pp. 807– 832, 2002.

[3] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," in Proc. Eurographics 2003, 2003, vol. 22, pp. 223–232.

[4] Z. Lian, A. Godil, and X. Sun, "Visual Similarity based 3D Shape Retrieval Using Bag-of-Features," in Proc. SMI'10, 2010, pp. 25-36.

[5] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," in Proc. SGP'03, 2003, vol. 43, pp 156–164.

[6] H. Sundar, D. Silver, N. Gavani, and S. Dickinson, “Skeleton based shape matching and retrieval,” in Proc. SMI’03, 2003, pp. 130–139.

[7] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, “The Princeton Shape Benchmark,” in Proc. SMI’04, 2004, pp. 167–178.

[8] Y. Yang, H. Lin, and Y. Zhang, “Content-based 3-D model retrieval: A survey,” in IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, Vol.37, No. 6, Nov. 2007.

[9] J. W. Tangelder and R. C. Veltkamp, “A survey of content based 3D shape retrieval methods,” Multimedia Tools and Applications, vol. 39, no. 3, pp. 441–471, 2008.

[10] R. Ohbuchi, T. Otagiri, M. Ibato, and T. Takei, “Shape-similarity search of three-dimensional models using parameterized statistics,” in the proceedings of the Pacific Graphics 2002.

[11] R. Ohbuchi, T. Minamitani, and T. Takei, “Shape-similarity search of 3D models by using enhanced shape functions,” International Journal of Computer Applications in Technology (IJCAT), pp.70-85, Vol.23, No. 2/3/4, 2005.

[12] E. Wahl, U. Hillenbrand, and G. Hirzinger, “Surflet-Pair-Relation Histograms: A Statistical 3D-Shape Representation for Rapid Classification,” in proceedings Forth International Conference on 3-D Digital Imaging and Modeling (3DIM 2003), pp.474-481, 2003.

[13] R. Ohbuchi and Y. Hata, “Combining Multiresolution Shape Descriptors for 3D Model Retrieval,” in Proc. WSCG’2006, 2006.

[14] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” 2001. http://www.csie.ntu.edu.tw/~cjlin/libsvm .