View Context: A 3D Model Feature for Retrieval

Bo Li and Henry Johan

School of Computer Engineering, Nanyang Technological University
Block N4, Nanyang Avenue, 639798, Singapore
{libo0002,henryjohan}@ntu.edu.sg

Abstract. 3D model feature extraction is an important issue for 3D object retrieval. We propose a novel 3D model feature named view context and based on this we construct a view context shape descriptor for 3D model retrieval. The view context of a particular view captures the distribution of visual information differences between this view and a set of arranged views. We select a set of feature views, compute their view contexts and use them as the shape descriptor of a 3D model. In order to enhance the retrieval accuracy, we perform an approximate symmetric axis-based 3D model alignment and propose a combined shape distance between two models, by incorporating the dissimilarity between two view context shape descriptors and the Zernike moments feature difference between the feature views of two models. Experiment results show that the view context shape descriptor is comparable with the state-of-the-art descriptors in retrieval performance.

Keywords: 3D model feature, view context, 3D model retrieval, shape deviation, visual information.

1 Introduction

With the increase in the number of available 3D models, the ability to accurately and efficiently search for 3D models is crucial in many applications. As a result, 3D model retrieval has become an important research area. In recent years, several algorithms that extract different 3D model features such as shape histogram [1], shape distribution [13], moment [5], Light Field [3], 3D harmonics [10], have been proposed for 3D model retrieval. Funkhouser et al. [6] present a multimodal web-based search engine which supports queries based on 3D sketches, 2D sketches, 3D models and/or text keywords. Even though there is much work done on 3D model retrieval, it is still a challenging problem to find a good 3D model feature for retrieval.

In this paper, we propose a novel 3D model feature named "view context". When we look at a 3D model from a view V (i.e. the viewer is located at V), the visible features of the model are the important visual information of the model from this view V. We assume a 3D model is represented as a triangle mesh. We present two definitions of the visual information of a 3D model from a view V: (1) A hybrid point feature set composed of contours, suggestive contours [4], boundaries as well as silhouettes of the model as seen from the view V;

S. Boll et al. (Eds.): MMM 2010, LNCS 5916, pp. 185–195, 2010.

(2) Zernike moment features of the silhouette image rendered from the view V. The view context of a view V is then defined as the differences of the visual information between V and a set of arranged views.

We apply the view context for 3D model retrieval. We concentrate on 3D model retrieval using 3D models as queries. In order to apply view context in retrieval, we propose a view context shape descriptor of a 3D model. The proposed shape descriptor consists of the view contexts computed at several sampling views. To improve the retrieval accuracy, we first align the 3D model using a proposed simple approximate symmetric axis-based approach before computing the shape descriptor for a 3D model. To further improve the retrieval accuracy, we also add the difference between the feature views sets of two models. Our main contributions are as follows:

(1) We devise a new 3D model feature named view context for identifying models. Similar models in general have similar view contexts and the view contexts of models from different classes are often distinctively different.

(2) We propose a retrieval algorithm based on view context and through experiments verify that it can often achieve comparable retrieval performance with the state-of-the-art descriptors.

The remaining of this paper is organized as follows. In Section 2, we review the related work in 3D model retrieval. Section 3 describes in detail the idea of view context. In Section 4, we propose the view context shape descriptor. In Section 5, an algorithm for 3D model retrieval using the view context shape descriptor is explained. The results of retrieval experiments are demonstrated in Section 6. Section 7 contains the conclusions and lists some topics for future work.

2 Related Work

The existing 3D model retrieval techniques can be classified into two categories: geometry-based techniques and visual information-based techniques. Geometry-based techniques use the distribution of geometric elements, such as vertices and faces, or some intrinsic topological structures to characterize the features of 3D models while visual information-based techniques extract features based on the rendered view images.

Most of the previous work in 3D model retrieval belong to the geometry-based techniques. The key idea of shape distribution [13] is to represent the shape signature as a probability distribution sampled from a shape function measuring the global geometric properties of a 3D model. Shape histogram [1] is an extension of the shape matching techniques from 2D to 3D. It encodes the distance distribution of the surface points as a function of the distance from the center of mass and spherical angle. It includes three descriptors: SHELL (only use distance), SECTOR (only use spherical angle), and SHELL and SECTOR (use both). 3D harmonics [10] uses spherical harmonics to compute the similarity between 3D models without the need to align their orientations. It decomposes a spherical function into orthogonal components while preserving the norms.

In the visual information-based techniques, the visual similarities between view images of different models are compared with each other to measure the difference of the models. Multiple view descriptor [7], Light Field descriptor [3] and our view context shape descriptor belong to this category. Multiple view descriptor classifies models by comparing the views rendered from the primary, secondary and tertiary viewing directions of principle axes after an alignment with PCA. Chen et al. [3] proposed the Light Field descriptor to define the distance of two models as the minimum distance between their 10 corresponding silhouette views, rendered from the vertices of a dodecahedron using the orthographic projection. The features in each image are encoded using the Zernike moments and Fourier descriptor. Their Light Field descriptor compares the views of different models directly, while our view context shape descriptor compares and encodes the difference of views of the same model first and then we compare the view context features of different models to measure their difference.

3 View Context

3.1 Definition

The view context of a particular view of a 3D model encodes the visual information differences between this view and a set of arranged views. It captures the shape appearance deviation of a 3D model. In our work, a 3D model is represented as a triangle mesh.

Assume that the 3D model is centered in the origin of a 3D coordinate system. Its view context from a view V_0 is defined as follows. First, we rotate the 3D model such that view V_0 coincides with the z-axis of the coordinate system. Then, we orderly sample a set of views $\{(\varphi, \theta)\}$ based on the current initial pose. For example, view (φ, θ) can be generated by first rotating the model φ degrees about the x axis and then θ degrees about the y axis. The view context of a view V_0 is composed of a set of feature vectors:

$$\{(\varphi, \theta, d) | (\varphi, \theta) \in \mathbf{V}\} \ , \tag{1}$$

where d is the view appearance distance between view (φ, θ) and view V_0. $\mathbf{V}$ is a sequence of sampling views. The methods of view sequence sampling and view distance computation will be presented in Sections 3.2 and 3.3, respectively.

View context represents the relative appearance deviations with other views. Figure 1 gives an example of the view contexts of the initial poses of six models (Figures 1(a)$\sim$(f)) in Princeton Shape Benchmark Database [14]. Figures 1(g)$\sim$(l) show the matrix representation of the six view contexts. In these examples, the view sequence $\mathbf{V}$ consists of 12 views based on the uniform step sampling (Section 3.2) and it is organized into a 3×4 matrix. The top-left element represents the view V_0 and other elements indicate other relative views with respect to V_0. The gray scale values represent the view appearance distances and darker means smaller values. Figure 1(m) shows the plot of the six view context. The graphs are obtained by ordering the elements of the matrix based on the following rule: from top to bottom

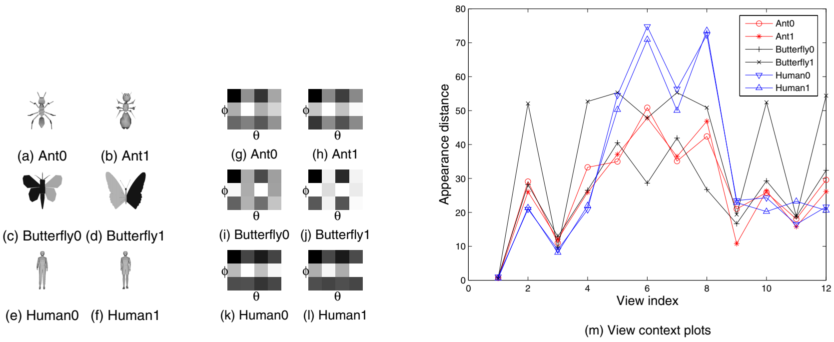


Fig. 1. View context of six models

and from left to right. We can see that similar models will have similar view contexts and the view contexts of different models are often distinctively different.

3.2 View Sequence Sampling

To decide the view sequence $\mathbf{V}$, we need to determine the values of φ and θ in Equation 1. The ranges of their values are $\varphi \in [0, 180]$ and $\theta \in [0, 360)$. To consider different types of features extracted, we adopt two sampling methods: uniform step-based sampling and cube-based sampling. The former is used for contour-based features, and the latter is for region-based features.

Uniform Step-Based Sampling. Using sampling steps $\Delta\varphi$ and $\Delta\theta$, the (φ, θ) for a view sequence $\mathbf{V}$ can be defined as follows.

$$\{(i * \Delta\varphi, j * \Delta\theta) | 0 \leq i \leq \left\lfloor \frac{180}{\Delta\varphi} \right\rfloor, 0 \leq j < \left\lfloor \frac{360}{\Delta\theta} \right\rfloor\} \ . \tag{2}$$

The view sequence $\mathbf{V}$ is written as $\mathbf{V} = \{V_0, V_1, \cdots, V_{n-1}\}$, where $n = (\left\lfloor \frac{180}{\Delta\varphi} \right\rfloor + 1) * \left\lfloor \frac{360}{\Delta\theta} \right\rfloor$. In our experiments, we set $\Delta\varphi = 90$ and $\Delta\theta = 90$. Then, the view sequence $\mathbf{V}$ consists of the following 12 views.

$$\{(\varphi, \theta) | \varphi \in \{0, 90, 180\}, \theta \in \{0, 90, 180, 270\}\} \ . \tag{3}$$

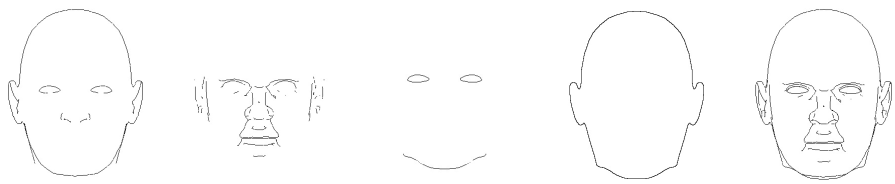
Cube-Based Sampling. By adopting region-based image descriptors, for every view we only need to generate a sequence of silhouette images rendered from a set of cameras using orthogonal projection. Considering the symmetry property of the projection, we render 13 views by setting the cameras on the surface of a cube. The camera locations are (1,1,1), (-1,1,1), (-1,-1,1), (1,-1,1), (1,0,0), (0,1,0), (0,0,1), (1,0,-1), (0,1,-1), (1,1,0), (0,1,1), (1,0,1), (1,-1,0). They comprise 4 top corner views, 3 adjacent face center views and 6 middle edge views. Based on these camera locations, we then compute the view sequence $\{(\varphi, \theta)\}$.

3.3 View Appearance Distance

To compute the view appearance distance d between two view images V_0 and V_i, we use hybrid feature matching cost if extracting contour-based features and use Zernike moment distance if extracting region-based features.

Hybrid Feature Distance. We first extract a hybrid feature point set of a view and then use a 2D shape matching algorithm to compute the view distance.

(1) **Feature extraction.** We define a hybrid feature set of a view image (Figure 2) by integrating contours, suggestive contours [4], boundaries as well as silhouettes. Contours are the point sets whose normal vectors are perpendicular to the view vector. Suggestive contours are contours in nearby views. A view image that includes the suggestive contours already integrates the contour features in the nearby views. Therefore, it facilitates a sparse view sampling and thus improves the efficiency of 3D model retrieval. Beside contours and suggestive contours, boundaries and silhouettes are also essential to accurately represent the features of 3D models. Boundary comprises edges that only belong to a single triangle face. Silhouette is an outline of the model from a certain view. Figure 2 shows one example of these features.



(a) Contours (b) Suggestive contours (c) Boundaries (d) Silhouettes (e) Hybrid features

Fig. 2. One example of the features of a view image

(2) **Feature sampling.** To reduce the time for feature matching, we need to first resample the feature points. In our experiments, we sample 100 points for every view image by adopting the following feature points sampling steps: curve extraction, cubic B-Spline curve interpolation and uniform sampling. During the uniform sampling, we set the number of sampling points for each curve to be proportional to its original length (in pixel).

(3) **View distance computation.** After sampling two feature point sets from the view images V_0 and V_i, we use the shape context matching algorithm [2] to compute the view distance of these two views.

Zernike Moments Distance. We use moments to extract the feature of the silhouette views V_0 and V_i. To achieve rotation invariance when measuring the difference of two images, we adopt the Zernike moments [11] and compute up to 10^{th} order moments (36 moments in total) for the feature of every view image. To compare the difference between two moments feature vectors, we adopt the Canberra distance [12]: $\sum_{i=1}^{36} |x_i - y_i|/|x_i + y_i|$, where $(x_1, x_2, \ldots, x_{36})$ and $(y_1, y_2, \ldots, y_{36})$ are the Zernike moment feature vectors of two views V_0 and V_i.

4 View Context Shape Descriptor

4.1 Definition

To build the view context shape descriptor for a 3D model, we can first align the model (Section 4.2) or use the original database directly, then select a set of feature views and compute the view contexts of these views, and finally concatenate them into a matrix as the view context shape descriptor.

For the feature views set selection, generally, the front, right, back and left views are the standard views for viewing a 3D model. Since we can perform an approximate alignment (Section 4.2) on the 3D model beforehand, we can only select these 4 views. However, if we do not perform the alignment, we select the 13 cube-based sampling views (Section 3.2) as our feature views.

4.2 Symmetric Axis-Based Alignment

Symmetry is an important factor of human perception to align similar models in similar poses. Considering this, we propose a simple method to align a 3D model based on the approximate symmetric-axis. The algorithm is as follows:

(1) **Align the 3D model using the Principal Component Analysis method [8].**

(2) **Find the best approximate symmetric axis.** We only consider the x, y and z axes after the alignment in step (1). We assume the vertices of the model are $\{v_i =< v_{i,x}, v_{i,y}, v_{i,z} > |i = 1, \cdots, t\}$, where t is the number of vertices. First, we find the best symmetric axis A by summing up x, y and z coordinates of all the vertices respectively and then find the axis that has the minimum absolute value,

$$A = argmin\{|\sum_{i=1}^{t} v_{i,x}|, |\sum_{i=1}^{t} v_{i,y}|, |\sum_{i=1}^{t} v_{i,z}|\} \ . \tag{4}$$

If A=$x/y/z$ (/ means or), it means that the symmetric plane is roughly yoz /zox /xoy. Then, we need to find the axis (y/z, z/x, x/y) that has bigger variance (larger eigenvalue) and regard it as the best symmetric axis.

(3) **Rotate the model such that the best symmetric axis coincide with the y axis.** If the three eigenvalues are all negative, we flip the model about the x axis.

We tested this simple and approximate algorithm on the 3D models in the Princeton Shape Benchmark Database [14] and found that in most cases it can align similar models in similar way except for some reflection errors about x axis. These errors may be due to the ambiguity of the principle axes, which is an intrinsic shortcoming of PCA. Some alignment results are shown in Figure 3.

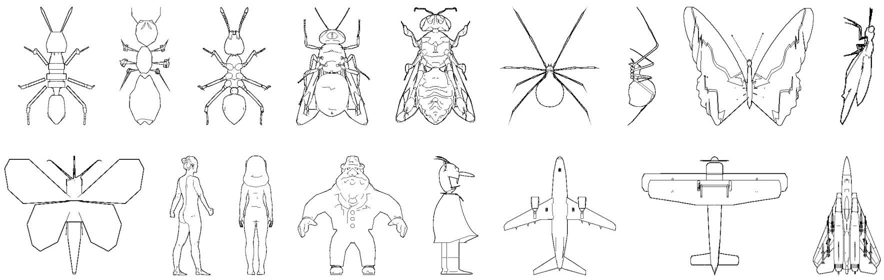


Fig. 3. Some alignment results

5 3D Model Retrieval with View Context

5.1 Retrieval Algorithm

We focus on retrieval using 3D models as queries. Given a query model, we propose a 3D model retrieval algorithm as follows.

(1) **Coordinates normalization.** To achieve translation and scale invariance, we translate the centers of all the models to the origin of the coordinate system and then scale them such that their bounding spheres have the same radius.

(2) **Pose normalization.** To enhance the retrieval accuracy, we perform the approximate symmetric axis-based 3D model alignment (Section 4.2).

(3) **Compute view context shape descriptor.** First, we select a set of feature views (e.g. 4 or 13), then compute the view context for each feature view and finally construct a view context shape descriptor by concatenating them.

(4) **Compute the shape distance matrix and ranking.** We design two shape distance metrics to measure the difference between two types of view context shape descriptors. We also propose a combined shape distance by combining the dissimilarity between two view context shape descriptors and the Zernike moments feature difference between two models' feature views sets. We describe these three distances in Section 5.2.

5.2 Shape Distance Metrics

Two candidate metrics that can be used to measure the distance between two view contexts are correlation and χ^2 distance. After comparing their differentiation capabilities through experiments, we found that correlation performs better. Therefore, we use correlation to measure the difference of two view contexts. As depicted in Section 4.1, we may choose 4 standard views (front/right/back/left) or 13 cube-based sampling views as feature views set. Accordingly, we design one shape distance metric for each, described as follows.

Ordered Correlation Distance for 4 Feature Views. The initial pose of a model can be in any view in the feature views set (front/right/back/left).

Therefore, to achieve rotation invariance among these 4 views, we need to design a metric that can encompass these differences. We assume that the feature views of two models are $\mathbf{V}$={V_0, V_1, V_2, V_3} and $\widetilde{\mathbf{V}}$={$\widetilde{V}_0$, $\widetilde{V}_1$, $\widetilde{V}_2$, $\widetilde{V}_3$}. We design an ordered correlation distance d_v based on the best correspondence between the feature views of two models,

$$d_v = \min_i \{\sum_{j=0}^{3} ViewCost[j][(i+j) \bmod 4]\}, \ \ i \in \{0,1,2,3\} \ . \tag{5}$$

$ViewCost[i][j] = 1 - C(V_i, \widetilde{V}_j)$ and $C(V_i, \widetilde{V}_j)$ is the correlation between the view contexts of V_i and $\widetilde{V}_j$.

LAP Correlation Distance for 13 Feature Views. For 13 feature views, we get a 13 × 13 view cost matrix $ViewCost$ and we use the Jonker's Linear Assignment Problem (LAP) algorithm [9] to correspond these two sets of feature views and use the minimal matching cost as the distance between them.

Combined Shape Distance. To improve the retrieval performance, we can also consider the Zernike moments feature difference between two feature views sets. We propose a combined shape distance which combines the dissimilarity between two view context shape descriptors, depicted by d_v, and the Zernike moments feature difference between two sets of feature views, depict by d_m. d_m is computed in the same way as view context shape descriptor except that we use the Canberra distance [12] rather than the correlation distance. We combine the two distances based on an automatic weighting method according to the differentiation ability of each type of distance. First, we normalize d_v and d_m into $\widetilde{d_v}$ and $\widetilde{d_m}$ by their respective maximum distances. Then, we compute the weights ω_v and ω_m for the view context feature distance and moment feature distance,

$$\omega_v = \frac{s_1}{s_1+s_2}, \ \ \omega_m = \frac{s_2}{s_1+s_2} \ . \tag{6}$$

s_1 and s_2 are the standard deviations of d_v and d_m over all the models in the database. Finally, we combine these two normalized features by their corresponding weights,

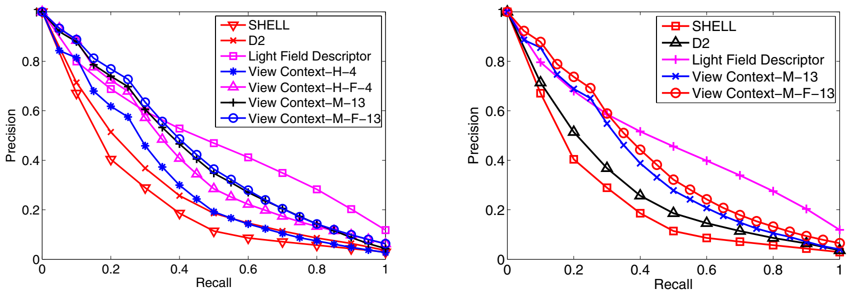
$$d = \omega_v * \widetilde{d_v} + \omega_m * \widetilde{d_m} \ . \tag{7}$$

6 Experiments and Discussion

To investigate the retrieval performance and the characteristics of our view context shape descriptor, we tested our view context descriptor on the Princeton Shape Benchmark Database (PSB) [14] and the National Taiwan University Benchmark (NTU) [3] and compared it with other state-of-the-art or related descriptors. We mainly use the precision-recall plots [14] to measure the retrieval performance. Recall means how much percentage of a class has been retrieved among the top K list while precision indicates how much percentage of the top K models belongs to the same class as the query model.

6.1 Princeton Shape Benchmark Database (PSB)

We used the test dataset of the PSB database [14]. It contains 907 models which are classified into 131 classes. Figure 4 gives the precision-recall plots of several variations of our proposed view context based retrieval algorithm as well as other three shape descriptors. For the precision-recall plots of D2 [13] and SHELL [1], we referred to the experiment results in [14]. For Light Field descriptor [3], we generated the plots using their provided execution file. Figure 4(a) shows the results on the approximately aligned PSB database, while Figure 4(b) shows the results on the original PSB database. We can see for this case, compared with the Light Field descriptor, view context shape descriptor can achieve better or comparable performance when retrieving a certain percentage of models (for example around 20 percent). Compared to the shape distribution (D2) and SHELL descriptor (one type of shape histogram descriptor), view context descriptor performs apparently better. We also found that for the same method (e.g. View Context-M-13), the approximate alignment process can improve the performance.



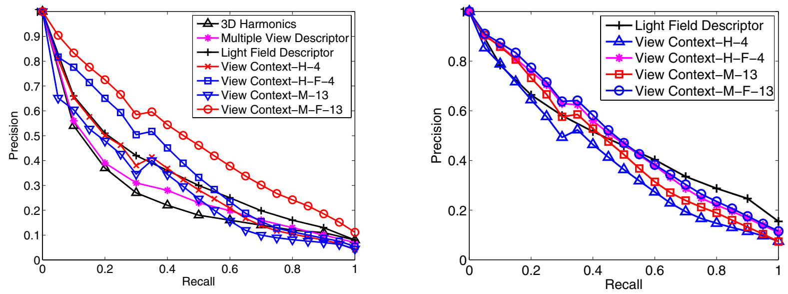
(a) Approximately aligned PSB database

(b) Original PSB database

Fig. 4. Precision-recall plots of our view context and other three descriptors: (a) Approximately aligned PSB database; (b) Original PSB database. "H" means the hybrid feature based view context and "M" means the Zernike moments based view context. "4": use the uniform step-based sampling and the 4 standard views as feature views set. "13": use the 13 cube-based sampling views as feature views set. "F": combined shape distance by integrating the differences in the view context shape descriptors and the feature views' Zernike moments.

6.2 National Taiwan University Database (NTU)

This database [3] contains 1833 3D models and only 549 3D models are clustered into 47 classes and the rest 1284 models are classified as the "miscellaneous". Therefore, we do the approximate alignment only for the 549 models. Since there are no PCA information in the database, we complete the approximate alignment process manually.



(a) Complete NTU database (b) Only classified NTU models

Fig. 5. Precision-recall plots of our view context and other three descriptors: (a) 1833 approximately aligned NTU database; (b) 549 approximately aligned NTU database (only classified models). The denotations of "H" ,"M" ,"F", "4" and "13" are described in Figure 4.

For the full approximately aligned database (1833 models), we compared our view context with the three descriptors mentioned in [3]: Light Field [3], 3D harmonics [10], and multiple view descriptor [7]. Figure 5(a) shows the comparison of the performance. For Light Field descriptor, we generated the plots using their provided execution file, while for 3D harmonics and multiple view descriptor, we referred to the experiment results in [3]. We can see that our view context shape descriptor can achieve a similar and comparable performance as the Light Field descriptor. It outperforms the 3D harmonics and the multiple view descriptor on average.

Figure 5(b) gives the performance comparison between our four methods and Light Field descriptor over the approximately aligned 549 classified models database. We can see that adding more feature views or integrating the Zernike moments difference of feature views sets can apparently improve the accuracy. Overall, we can achieve better results than the Light Field.

7 Conclusions and Future Work

We have presented a new 3D model feature: view context, which captures the shape deviation distribution feature of a 3D model. It can differentiate models effectively because similar models have similar view contexts and different models in general have apparently different view contexts. To improve the retrieval accuracy, we perform an approximate alignment and propose a combined shape distance. Experiment results show that the view context shape descriptor is comparable with the state-of-the-art descriptors in retrieval performance.

There are still many facets about the view context to be explored. For instance, we can adopt a different view sampling method by setting the cameras on the 20 vertices of a regular dodecahedron. We can also organize a view context scale-space by dividing the view space uniformly at a series of scales, arranged from coarse to fine. For example, we set the uniform sampling steps $\Delta\varphi=\Delta\theta=60°, 30°, 15°$ and then define three view context features for a model at different scales, which are 3×6, 6×12 and 12×24 matrices.

References

1. Ankerst, M., Kastenmuller, G., Kriegel, H.P., Seidl, T.: 3D shape histograms for similarity search and classification. In: Güting, R.H., Papadias, D., Lochovsky, F.H. (eds.) SSD 1999. LNCS, vol. 1651, pp. 207–226. Springer, Heidelberg (1999)
2. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
3. Chen, D.Y., Tian, X.P., Shen, Y.T., Ming, O.: On visual similarity based 3D model retrieval. In: Eurographics, Computer Graphics Forum, pp. 223–232 (2003)
4. Doug, D., Adam, F., Szymon, R., Anthony, S.: Suggestive contours for conveying shape. ACM Transactions on Graphics 22(3), 848–855 (2003)
5. Elad, M., Tal, A., Ar, S.: Content based retrieval of VRML objects - an iterative and interactive approach. In: Proc. of 6th Eurographics Workshop on Multimedia, pp. 97–108 (2001)
6. Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., Jacobs, D.: A search engine for 3D models. ACM Transactions on Graphics 22(1), 83–105 (2003)
7. Jeannin, S., Cieplinski, L., Ohm, J.R., Kim, M.: MPEG-7 Visual part of eXperimentation Model Version 7.0. ISO/IEC JTC1/SC29/WG11/N3521, Beijing (2000)
8. Jolliffe, I.: Principal Component Analysis, 2nd edn. Springer, Heidelberg (2002)
9. Jonkerand, R., Volgenant, A.: A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing 38(4), 325–340 (1987)
10. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Proc. of the Symposium on Geometry Processing 2003, pp. 156–164 (2003)
11. Khotanzad, A., Hong, Y.: Invariant image recognition by Zernike moments. IEEE Transactions on Pattern Analysis and Machine Intelligence 12(5), 489–497 (1990)
12. Laga, H., Nakajima, M.: Supervised learning of similarity measures for content-based 3D model retrieval. In: Tokunaga, T., Ortega, A. (eds.) LKR 2008. LNCS (LNAI), vol. 4938, pp. 210–225. Springer, Heidelberg (2008)
13. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D models with shape distributions. In: Proc. of Shape Modeling and Applications 2001, pp. 154–166 (2001)
14. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The princeton shape benchmark. In: Proc. of Shape Modeling and Applications 2004, pp. 167–178 (2004)