A New 3-D Model Retrieval System Based on *Aspect-Transition Descriptor*

Soochahn Lee[1], Sehyuk Yoon[2], Il Dong Yun[3], Duck Hoon Kim[4], Kyoung Mu Lee[1], and Sang Uk Lee[1]

[1] School of Electrical Engineering and Computer Science, Seoul National University, Seoul, 151-742, Republic of Korea
redhouse@diehard.snu.ac.kr, kyoungmu@snu.ac.kr, sanguk@ipl.snu.ac.kr
[2] Service Planning Department, KT corporation, Seognam, Kyonggi-do, 463-711, Republic of Korea
uniqness@kt.co.kr
[3] School of Electronics and Information Engineering, Hankuk University of Foreign Studies, Yongin, 449-791, Republic of Korea
yun@hufs.ac.kr
[4] Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA 90089, USA
duckkim@usc.edu

Abstract. In this paper, we propose a new 3-D model retrieval system using the *Aspect-Transition Descriptor* which is based on the aspect graph representation [1, 2] approach. The proposed method differs from the conventional aspect graph representation in that we utilize transitions as well as aspects. The process of generating the *Aspect-Transition Descriptor* is as follows: First, uniformly sampled views of a 3-D model are separated into a stable and an unstable view sets according to the local variation of their 2-D shape. Next, adjacent stable views and unstable views are grouped into clusters and we select the characteristic aspects and transitions by finding the representative view from each cluster. The 2-D descriptors of the selected characteristic aspects and transitions are concatenated to form the 3-D descriptor. Matching the *Aspect-Transition Descriptor*s is done using a modified Hausdorff distance. To evaluate the proposed 3-D descriptor, we have evaluated the retrieval performance on the Princeton benchmark database [3] and found that our method outperforms other retrieval techniques.

1 Introduction

For years, 3-D model retrieval has been of interest for applications mostly in specialized areas such as mechanical CAD, molecular biology, and computer graphics. With the recent increase in the number and variety of 3-D models, however, new types of applications intended for the general public, such as personalized 3-D cyber-rooms and avatars, and tools for creating amateur 3-D animated

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part IV, LNCS 3954, pp. 543–554, 2006.

motion pictures have been gaining interest. In order to make this a reality, tools for easy authoring of new 3-D models, the construction of large databases of 3-D models, and the efficient and accurate retrieval of 3-D models from the database are problems that must be solved. Especially, for these kinds of applications that are not related to only a few classes of models, it is necessary to develop a 3-D model retrieval system that is not task-dependent and can accurately retrieve free-form objects.

The problem of retrieving the relevant 3-D models from the database is closely linked to how the model is described, *i.e.*, which characteristic the 3-D descriptor is based on. Whether it is based on the 3-D geometry or, as shown in Fig. 1, the projected 2-D views of the model divides 3-D descriptors into geometry-based [4, 5, 6, 7, 8] and view-based [9, 10] ones. Geometry-based descriptors utilize different geometrical attributes such as shape histograms [4] of vertices and the distribution of distances of two random points on the model [5]. Descriptors based on more complex geometrical attributes include extended Gaussian images [6], spherical extent functions [7], and spherical harmonic descriptors [8]. On the other hand, view-based descriptors are all based on 2-D views, but differ in how the viewpoints are organized or selected.

As 3-D models become free-form, it becomes almost impossible to extract high-order geometrical attributes. Since geometry-based descriptors are based on primitive geometrical attributes, they rely on extensive statistical information to characterize 3-D models. But it can be difficult to determine the amount of statistical information needed for accurate description of models. View-based descriptors, however, do not suffer from this problem since they intuitively encapsule the information needed to discriminate between inter-class differences and intra-class similarities for any class of 3-D models. Also, view-based descriptors have the advantage that 3-D model retrieval systems based on it is able to provide a more user friendly interface by enabling the user to utilize a 2-D image or sketch as a query input. It is worthy to note that the Light Field Descriptor [9], which belongs to view-based descriptors, is producing superior results compared to other geometry-based descriptors [3].

Unlike the Light Field Descriptor [9] which samples the 3-D model at vertices of a regular dodecahedron, the aspect graph representation [1, 2] focuses on generating aspects and connecting them into a graph. Specifically, aspects that have small change in the projected shape of the model with change of viewpoint are connected by transitions in which the change in projected shape is dramatic. Note that, in this paper, aspects and transitions correspond to stable views and unstable views, respectively. Aspect graph representations have been defined for special classes of models such as polyhedra [11], solids of revolution [12, 13], piece-wise smooth objects [14], and algebraic surfaces [15]. However, one main issue arises, *i.e.*, how to reduce the number of aspects for complex shapes or free-form models. Ikeuchi and Kanade [16] proposed a method to group views using similar features extracted from the model, and Weinshall and Werman [17] formally analyzed the notion of view stability and view likelihood which can be used to determine characteristic views. Recently, Cyr and Kimia [10] proposed a

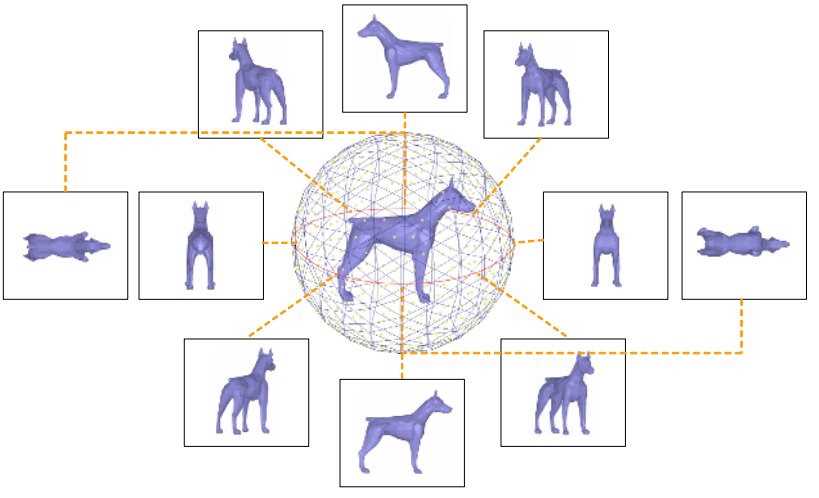

Fig. 1. Describing a 3-D model using its projected images at various viewpoints

new aspect graph representation which first groups views using a region-growing approach based on the similarity between adjacent views and then defines the aspect as the characteristic view for each group. Their approach has advantages that it is applicable to free-form objects and it reduces the number of aspects needed to describe a 3-D model.

The proposed method is similar to that of Cyr and Kimia [10] in that we also focus in grouping views and defining aspects using characteristic views. The main difference is that we utilize not only stable views but also unstable views, *i.e.*, transitions as well as aspects. This is based on the observation that transitions may contain more additional information compared to similar aspects. Specifically, we first classify views on the view sphere into stable views and unstable views using *local variation* which is a notion that will be made clearer in Section 2. Next, we separately group the stable views and unstable views into clusters based on similarity and define the aspect for stable view clusters and the transition for unstable view clusters by finding the characteristic views of each cluster. We call the 3-D descriptor constructed using the aspects and transitions the *Aspect-Transition Descriptor*. Consequently, the *Aspect-Transition Descriptor* utilizes only the most representative aspects, and as a result utilizes a small number of views. Note that, we avoid the need to align the 3-D models in the process of constructing descriptors.

The paper is organized as follows: In Section 2, the proposed view classification method and the process of generating the *Aspect-Transition Descriptor* are explained in detail. Then Section 3 illustrates the matching technique based on the Hausdorff distance for the proposed retrieval system. Section 4 presents and analyzes the performance of the proposed retrieval system from experiments conducted using the Princeton benchmark database [3]. Finally, Section 5 concludes this paper.

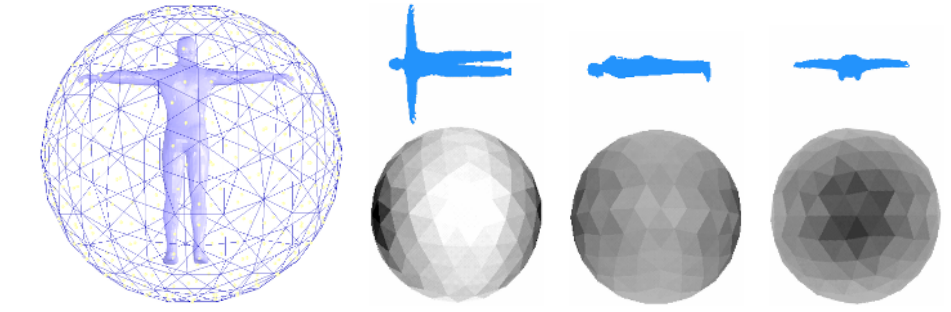

Fig. 2. Visualization of the view sampling and view classification process. (a) The wireframe is the twice subdivided icosahedron with 320 faces. The bright dots are the centers of each face which represents the viewpoints actually used. The top shapes in (b), (c), and (d) represent the projected images of the model in (a), and the bottom polyhedra represent the distribution of adjacent local variations for each image. Bright faces indicate low local variation, while dark faces indicate high local variation.

2 Generation of the *Aspect-Transition Descriptor*

2.1 View Sampling and Classification

The generation of the *Aspect-Transition Descriptor* begins with view sampling. View sampling is the process of sampling the view of the 3-D model from various viewpoints on the view sphere which is a sphere normalized relative to the size of the model. Here, the viewpoints in the sampling process must be uniformly distributed on the view sphere, so the geometry of regular polyhedra is used. Also, the number of viewpoints must be large enough to sufficiently sample the shape of the model. Therefore, we somewhat over-sample the model using the center point of the faces in a twice subdivided icosahedron as viewpoints [18]. Specifically, we create 320 sampled views of the 3-D model.

Next, we classify the sampled views into stable views and unstable views based on *local variation* which is the approximation of local shape variation extensively explored in [17]. Unfortunately, the notion of differentiation of local shape variation is practically inapplicable, so we approximate that notion with computing the dissimilarity of each view V_i with its neighboring views V_j on the assumption that the sampled views are sufficiently dense. This approximation denoted as $L(V_i)$ is called *local variation* and is defined by

$$L(V_i) = \sum_j \frac{d(V_i, V_j)}{g(V_i, V_j)}, \tag{1}$$

where $d(V_i, V_j)$ is the dissimilarity between V_i and V_j, and $g(V_i, V_j)$ is the geometric distance between the viewpoints of the two views. Here, the dissimilarity is computed using 2-D descriptors as image metrics for each view. We will discuss 2-D descriptors in Section 2.3.

Now, using local variation as the criteria, the stable view set **SV** and the unstable view set **UV** is defined as follows:

$$\mathbf{SV} = \left\{ V_{sv} \,\middle|\, L(V_{sv}) < \frac{1}{n}\sum_{i} L(V_i) - \theta_{sv} \right\}, \tag{2}$$

$$\mathbf{UV} = \left\{ V_{uv} \,\middle|\, L(V_{uv}) > \frac{1}{n}\sum_{i} L(V_i) - \theta_{uv} \right\}, \tag{3}$$

where n is the total number of sampled views, and θ_{sv} and θ_{uv} are thresholds based on the standard deviation of the *local variation* values that control the number of stable and unstable views. Fig. 3 shows an example of a stable view and an unstable view with its neighboring views.

2.2 Selecting Characteristic Views

After dividing stable views and unstable views, we separately select characteristic views from each set. Here, characteristic aspects and transitions are selected in a similar manner. Specifically, as characteristic aspects and transitions are selected by first grouping adjacent stable views and then finding the most representative view in each group.

Characteristic aspects are selected by the following steps: First, we assign each stable view to be in its own group. Next, we calculate the distance between every group and merge the two groups with the shortest distance where distance is the dissimilarity of the two views. After merging two groups, we assign a new

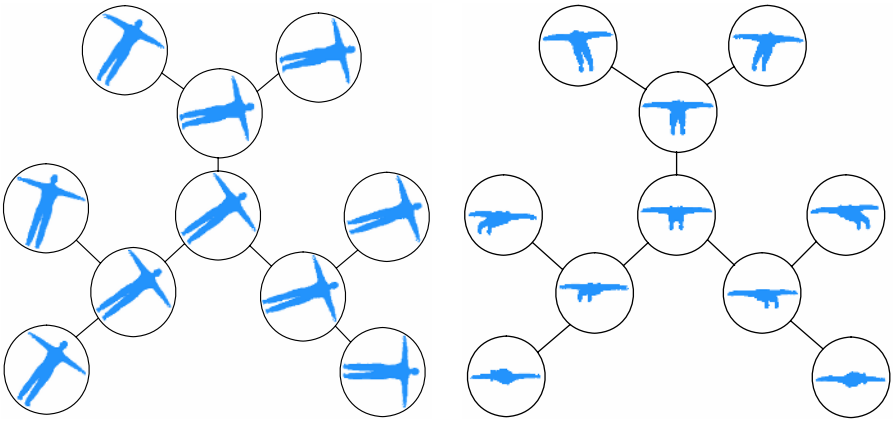

Fig. 3. Illustration of neighboring views for (a) a stable view and (b) an unstable view where the center image is the corresponding view. The connected neighboring views are from viewpoints connected on the twice subdivided icosahedron of Fig. 2(a).

representative view such that the sum of dissimilarity between other views and that cluster is the smallest. The above process is iterated until the sum of the maximum dissimilarity between the representative view and other views for each group grows larger than a certain threshold. Finally, the representative views for the remaining groups of stable views are defined as the characteristic aspects. We select characteristic transitions by the same process using the set of unstable views instead of stable views.

The characteristic aspects and transitions for several models are presented in Fig. 4. We can see that viewpoints of the characteristic aspects and transitions based on the proposed method are unevenly spaced which enables the appropriate information corresponding to aspects and transitions to be extracted.

2.3 Extracting the *Aspect-Transition Descriptor*

Now that the characteristic aspects and transitions have been selected, the remaining process is to construct the *Aspect-Transition Descriptor*. Since the *Aspect-Transition Descriptor* is constructed by concatenating 2-D descriptors of the selected characteristic aspects and views, we focus mainly on which 2-D descriptor to be utilized.

The method for describing 2-D images can be classified into two main classes, which are contour-based descriptor and region-based descriptor. Specifically, the contour-based descriptor utilizes only the information of the boundary of the shape, while the region-based descriptor utilizes all the pixels that constitute the shape region. Both the descriptors can be viewed as only partial information

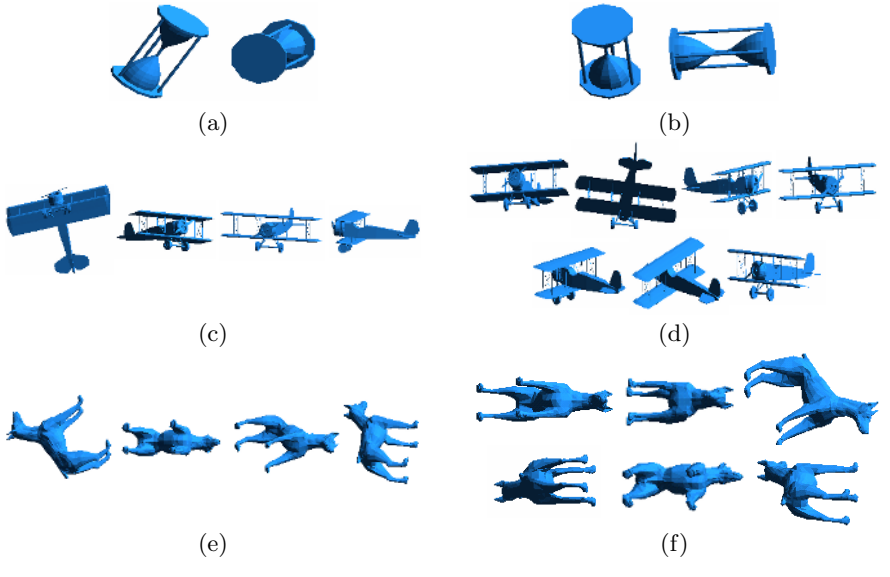

Fig. 4. Characteristic aspects (a, c, e) and characteristic transitions (b, d, f) for an hourglass (a, b) a biplane (c, d), and a dog (e, f)

of the image since the contour-based descriptor loses the interior information of the shape and cannot describe unconnected regions, and the region-based descriptor loses the detailed boundary information. Therefore, we utilize the Angular Radial Transform (ART) [19] and the Curvature Scale Space (CSS) [20] descriptors as the region based and contour based descriptors, respectively. Here, the actual extraction of the 2-D descriptors is performed at the time of view sampling for every view in order to use the descriptors as the basis of both dissimilarity and local variation in the view classification process described in Section 2.1.

Figs. 5 and 6 show the characteristic aspects and transitions of a cocktail glass selected using the CSS [20] descriptor and those of an hourglass using the ART [19] descriptor, respectively. Fig. 5(a) shows that the aspects represent a circular shape since the circular shape will have the smallest variation to change of viewpoint, and the median of the frontal and top view since at this point the shape variation exerts a local minima. On the other hand, Fig. 5(b) shows that transitions are selected from viewpoints where a viewpoint change brings a change in the topology of the contour shape, *i.e.*, from two overlapped circles to two connected circles. For the case of the hourglass, it can be seen in Fig. 6(a) that viewpoints for the aspects are located straight in front and at the top of the model, while the left transition in Fig. 6(b) is located at a region diagonal to the front or top of the model. Here, the right part of Fig 6(b) is selected as a transition since one of the pillars in occluding the two cones which could be in non-occlusion by changing the viewpoint a little bit.

Finally, the *Aspect-Transition Descriptor* is constructed by concatenating the 2-D descriptors for the characteristic aspects and transitions.

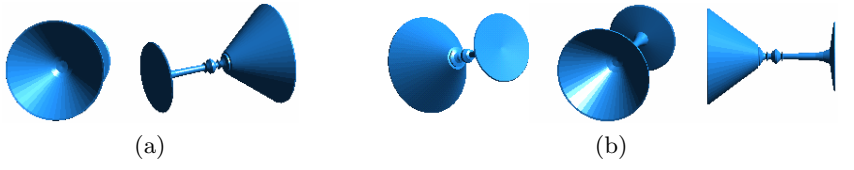

Fig. 5. (a) Characteristic aspects and (b) characteristic transitions for a cocktail glass model using the CSS [20] descriptor as the dissimilarity measure

Fig. 6. (a) Characteristic aspects and (b) characteristic transitions for an hourglass model using the ART [19] descriptor as the dissimilarity measure

3 Computing the Distance Between Two *Aspect-Transition Descriptor*s

Now, the remaining problem is how to compute the distance between *Aspect-Transition Descriptor*s. Generally, the number of both the characteristic aspects and transitions varies according to the complexity of a given 3-D model. Therefore, a specific distance measure is required for computing the distance between *Aspect-Transition Descriptor*s. Here, we convert with this problem into the problem of matching two point sets of which the numbers of elements are different.

The Hausdorff distance is widely used as a metric defining the distance between two point sets. Formally, given two point sets A and B, the Hausdorff distance is defined as follows:

$$f(A,B) = \max\left\{\overrightarrow{f}(A,B), \overrightarrow{f}(B,A)\right\}, \tag{4}$$

where $\overrightarrow{f}(A,B)$ and $\overrightarrow{f}(B,A)$ imply the directed Hausdorff distance. Here, the directed Hausdorff distance $\overrightarrow{f}(A,B)$ is represented in the following equation:

$$\overrightarrow{f}(A,B) = \max_{a\in A}\min_{b\in B} d(a,b). \tag{5}$$

Here, a and b imply a point in A and B, respectively, and $d(a,b)$ is the distance between a and b. In general, $\overrightarrow{f}(A,B)$ is not equal to $\overrightarrow{f}(B,A)$ since the directed Hausdorff distance is not commutative.

Although the Hausdorff distance is suitable for computing the distance between two point sets of which the numbers of elements are different, some modifications must be done in order to be utilized in our retrieval system. Considering that the Hausdorff distance ultimately measures only the maximum of minimum distances between two point sets and discards all other minimum distances, we need to modify this to take the similarities between all of the characteristic aspects and transitions into account. Specifically, the modified Hausdorff distance uses the average instead of the maximum as follows:

$$f_{\,\mathrm{mod}}(A,B) = \max\left\{\overrightarrow{f_{\,\mathrm{mod}}}(A,B), \overrightarrow{f_{\,\mathrm{mod}}}(B,A)\right\}, \tag{6}$$

where

$$\overrightarrow{f_{\,\mathrm{mod}}}(A,B) = \frac{1}{n}\sum_{a\in A}\min_{b\in B} d(a,b). \tag{7}$$

Here, n is the number of views in A.

Note that we separate the characteristic aspects and transitions of a 3-D model and consider them as two separate sets instead of considering them as one view set. Therefore, computing the distance between two *Aspect-Transition Descriptors* can be considered as combining the computed similarities of the aspects and transitions for the two models.

4 Experimental Results

We evaluated the performance of our retrieval system on the Princeton benchmark database [3]. The database contains 1814 models divided into two sets of 907 models for training and testing. Since training is not needed here, we used only the testing set which is classified into 92 classes. Generally, the performance of a retrieval system heavily depends on the database and categorization of 3-D models. The Princeton benchmark [3] is well organized and is recently being established as the standard benchmark for comparing the performance of 3-D model retrieval algorithms.

To compare the performance of several descriptors, we use three well-known measures: The first-tier, second-tier, and precision-recall plot [3]. The first-tier and second-tier are the percentage of models in the query's class that is retrieved in the top $|C| - 1$ matches and in the top $2 \times (|C| - 1)$ matches, respectively, where $|C|$ implies the number of models in a class C. The precision-recall plot describes the relationship between the precision and the recall of the retrieval system. Specifically, for a query model in a class C with $|C|$ models, the precision is the ratio of the retrieved members of class C in the top K matches to K, and the recall is the ratio of the retrieved members of class C in the top K matches to $|C|$. Note that the recall and precision are inversely proportional since increasing K raises the recall but brings a decrease in precision.

As described in Section 2.2, the classification of stable and unstable views and the number of selected characteristic views in the view selection process are determined by threshold values. In our implementation, the view classification thresholds θ_{sv} and θ_{uv} are both set to 0.5. Fig. 7 shows the the precision-recall plots for the following three cases: the first is when the threshold for the ART descriptor is 0.3 for selecting characteristic aspects and 0.2 for selecting characteristic transitions. Next, the second is the precision-recall plots for the cases using both aspects and transitions with both thresholds 0.8 for the ART descriptor and both 0.3 for the CSS descriptor. Here, the computed distances of aspects and transitions are combined by simple averaging. Finally, the third, which yielded the best performance, is when using the ART and CSS descriptors combined with thresholds 0.8 and 0.3 for ART and CSS, respectively. The distances computed using ART and CSS were combined by weighted averaging as $d_{combined} = 0.3 \times d_{ART} + 0.7 \times d_{CSS}$. Also, we note that the retrieval performance in the case of combined aspects and transitions with threshold 0.8 (18.7 views per model) is substantially better than the case using only aspects with a lower threshold 0.3 which has more views (19.5 views per model).

Next, the results of the first-tier and second-tier for our descriptor and several others evaluated on the Princeton benchmark database [3] are given in Table 1. Specifically, the results of D2 Shape Distribution(D2), Extended Gaussian Image (EGI), Complex EGI (CEGI), Shape Histograms (SHELLS, SECTORS, SECSHEL), Spherical Extent Function (EXT), Radialized EXT (REXT), Gaussian Euclidian Distance Transform (GEDT), Spherical Harmonic Descriptor (SHD),

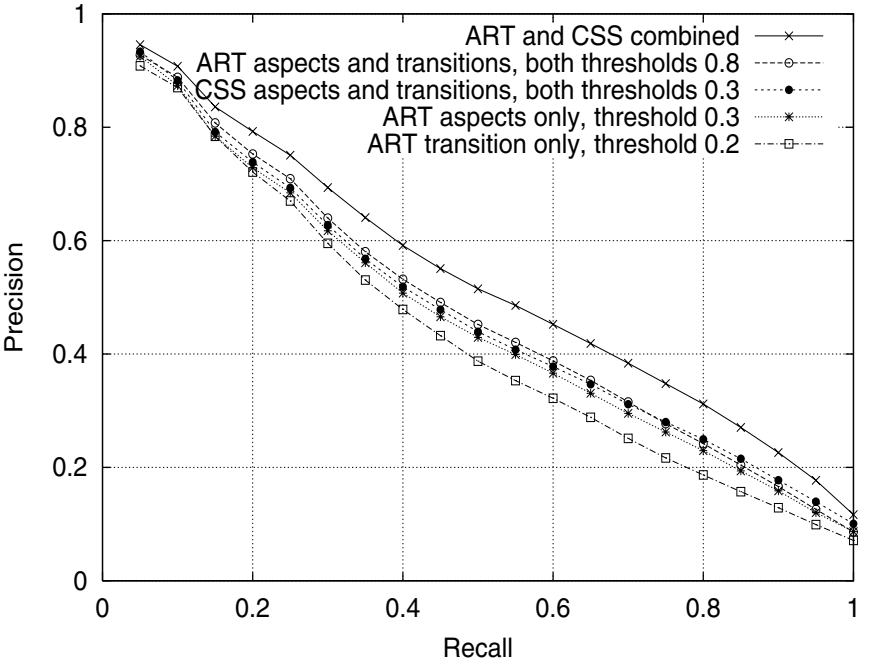

Fig. 7. The precision-recall plots using various parameters tested on the Princeton benchmark database [3]

and Light Field Descriptor (LFD) are compared to our method. It can be seen that the results of the *Aspect-Transition Descriptor* gives the best results. Specifically, the *Aspect-Transition Descriptor* outperforms the Light Field Descriptor [9] by 1.6% for the first tier and 1.4% for the second tier.

Finally, the precision-recall plots of the proposed descriptor and other descriptors discussed above are given in Fig. 8. It can be seen that the proposed method provide the best retrieval performance. Also, the proposed descriptor has the advantage that a smaller number of sampled views can be utilized to construct the proposed descriptor compared to the Light Field Descriptor.

5 Conclusion

We proposed a new 3-D descriptor called the *Aspect-Transition Descriptor* and described a novel retrieval system based on this descriptor. The *Aspect-Transition Descriptor* is a view-based descriptor based on the aspect graph representation, which particularly utilizes transitions as well as aspects to describe a 3-D model. This is based on the intuitive observation that transitions encapsule as much information of the model as aspects. From this, the proposed descriptor is able to achieve the information that was overlooked by traditional aspect graph representations. We also adopted a method to compare the

Table 1. First and second tier results using various 3-D descriptors on the Princeton benchmark database [3]

Shape Descriptor	First Tier	Second Tier
Proposed(ATD)	39.6%	50.1%
LFD	38.0%	48.7%
REXT	32.7%	43.2%
SHD	30.9%	41.1%
GEDT	31.3%	40.7%
EXT	28.6%	37.9%
SECSHEL	26.7%	35.3%
VOXEL	26.7%	35.0%
SECTORS	24.9%	33.4%
CEGI	21.1%	28.7%
EGI	19.7%	27.7%
D2	15.8%	23.5%
SHELLS	11.1%	17.3%

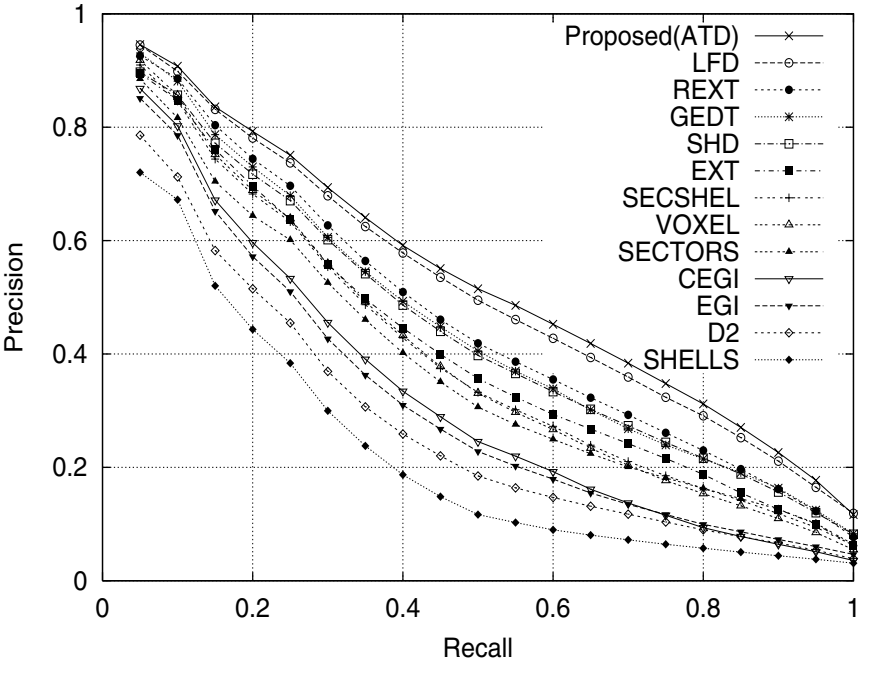

Fig. 8. The precision-recall plot of our retrieval system tested on the Princeton benchmark database [3] compared with other 3-D descriptors tested on the same database

distance of *Aspect-Transition Descriptor*s based on the Hausdorff distance. We have evaluated our retrieval system on the Princeton benchmark [3] and found that our system gives the best overall results.

References

1. Koenderink, J.J., van Doorn, A.J.: The singularities of the visual mapping. Biological Cybernetics **24** (1976) 51–59
2. Koenderink, J.J., van Doorn, A.J.: The internal representation of solid shape with respect to vision. Biological Cybernetics **32** (1979) 211–216
3. Shilane, P., Min, P., Kazhdan, M.M., Funkhouser, T.A.: The princeton shape benchmark. In: Proceedings of Shape Modeling International. (2004) 167–178
4. Ankerst, M., Kastenmüller, G., Kreigel, H.P., Seidl, T.: Nearest neighbor classification in 3d protein databases. In: Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology. (1999) 34–43
5. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D models with shape distributions. In: Proceedings of Shape Modeling International. (2001) 154–166
6. Horn, B.K.P.: Extended gaussian images. Proceedings of the IEEE **72** (1984) 1656–1678
7. Vranic, D., Saupe, D.: 3D model retrieval with spherical harmonics and moments. In: Proceedings of the DAGM:The German Association for Pattern Recognition. (2001) 392–397
8. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3d shape descriptors. In: Proceedings of on Geometry Processing. (2003)
9. Chen, D.Y., Tian, X.P., Shen, Y.T., Ouhyoung, M.: On visual similarity based 3d model retrieval. Computer Graphics Forum **22** (2003) 223–232
10. Cyr, C.M., Kimia, B.B.: A similarity-based aspect-graph approach to 3d object recognition. International Journal of Computer Vision **57** (2004) 5–22
11. Shimshoni, I., Ponce, J.: Finite-resolution aspect graphs of polyhedral objects. IEEE Transcations on Pattern Analsis and Machine Intelligence **19** (1997) 315–327
12. Kriegman, D.J., Ponce, J.: Computing exact aspect graphs of curved objects: Solids of revolution. International Journal of Computer Vision **5** (1990) 119–135
13. Eggert, D.W., Bowyer, K.W.: Computing the perspective projection aspect graph of solids of revolution. IEEE Transcations on Pattern Analsis and Machine Intelligence **15** (1993) 109–128
14. Shokoufandeh, A., Marsic, I., Dickinson, S.: View-based object recognition using saliency maps. Image and Vision Computing **17** (1999) 445–460
15. Petitjean, S., Ponce, J., Kriegman, D.: Computing exact aspect graphs of curved objects:algebraic surfaces. International Journal of Computer Vision **9** (1992) 231–255
16. Ikeuchi, K., Kanade, T.: Automatic generation of object recognition programs. Proceedings of the IEEE **76** (1988) 1016–1035
17. Weinshall, D., Werman, M.: On view likelihood and stability. IEEE Transcations on Pattern Analsis and Machine Intelligence **19** (1997) 97–108
18. Kim, D.H., Yun, I.D., Lee, S.U.: Regular polyhedral descriptor for 3-d object retrieval system. In: Proceedings of International Conference on Image Processing (3). (2003) 592–532
19. Manjunath, B., Salembier, P., Sikora, T.: Introduction to MPEG-7. Wiley (2002)
20. Mokhtarian, F., Mackworth, A.K.: A theory of multiscale, curvature-based shape representation for planar curves. IEEE Transcations on Pattern Analsis and Machine Intelligence **14** (1992) 789–805