ShapeLab: A Unified Framework for 2D & 3D Shape Retrieval

Jiantao Pu, Karthik Ramani

Purdue Research and Education Center for Information Systems in Engineering (PRECISE), Purdue University, West Lafayette IN 47907-2024, USA
pjiantao@stanford.edu, raman@purdue.edu

Abstract

2D or 3D shapes are the most important visual information that we use to recognize an object. We propose a unified framework “ShapeLab” to search similar 2D or 3D shapes from an existing database. Users can search 3D shapes with a 2D input, and vice versa. ShapeLab is composed of four key components: (1) pose determination for 3D models; (2) 2D orthogonal view generation based on multiple levels of detail; (3) similarity measurement between 2D shapes; and (4) freehand sketch-based user interface. Key algorithms supporting the above components are briefly described. Experiments show ShapeLab can provide a better performance such as high accuracy, flexibility and scalability compared to the available methods.

1. Introduction

In almost in every field, pictures, diagrams, legacy drawings and 3D models, have knowledge embedded in them. In designing people like to express their ideas in intuitive ways using 2D and 3D graphical tools as well as sketches. Reusing and sharing the knowledge embedded in these representations is becoming an important way to accelerate the design process, improve product quality, and reduce costs. Generally, 2d and 3d content can be simply described by their properties such as size, function, texture and topology. It is also natural to perform information retrieval queries using some traditional approaches such as keywords, encoding approach, and tree-like structure navigation. Although these approaches are simple to implement, they are not sufficient to describe a 2D or 3D shape. The content in 2D and 3D models that are unique to each one, can be depicted uniformly by using its 2D or 3D shape representation which contains the most important visual information. Therefore, the reuse of knowledge is transformed into the similarity computation between 2D or 3D shape descriptors. However, it is not easy to compute the similarity between 2D or 3D shapes: (1) Unlike other pattern recognition tasks such as face recognition that have predefined features such as eyes, nose and mouth, it is impossible to use a set of limited features to uniformly describe all kinds of 3D shapes, and

(2) It is difficult for a shape representation to achieve a balance between the global and local shape. The term ‘similar models’ is meant for objects that are similar on the whole, but still have some differences. If we only consider their global similarity, it is difficult to rank a group of similar models in an order according to their similarity. In addition, if we only consider their local similarity, it is hard to assure that the searched models are similar at the global level. Moreover, the bridging of the gap between 2D & 3D shapes is still a challenging problem.

In this paper, we describe a unified framework by which users can search 2D and 3D shapes using freehand sketches, 2D and 3D input. Since 2D shapes and 3D models are very popular in engineering, we use them as examples to explain our approach. As Figure 1 shows, the proposed framework is supported by four key components: (1) pose determination for 3D models: bridging the space gap between 2D and 3D shapes by representing 3D shape in 2D space in three intuitive and robust orthogonal orientations; (2) 2D orthogonal view generation based on multiple levels of detail: representing a 3D model by three orthogonal views along the pose orientations at different details; (3) similarity measurement between 2D shapes: finding 2D and 3D shapes similar to a user’s query; and (4) freehand sketch-based user interface: providing a natural way for a designer to express his shape intentions. This proposed framework has some distinct characteristics: (1) users can search 3D shapes with a 2D shape input, and vice versa; (2) it allows users to sketch queries at three levels of detail, thus supporting a coarse-to-fine search; and (3) the retrieval performance is robust to noise and independent of the size and space position of a shape.

Figure 1. Uniform framework for 2D&3D shape retrieval

2. Related work

2D contour is widely used to recognize an object since it is a compact representation. Well known methods for 2D contour matching include Fourier descriptors [1], curvature scale space (CSS) [2], chain codes [3], Zernike moments [4] and the classical Hausdorff distance [5]. In MPEG-7, a region base shape descriptor named Angular Radial Transform (ART) [6] was adopted to differentiate 2D shapes. As a moment-based image description method, ART has many good properties, such as compact size, robustness to noise, and invariance to rotation. Although the contour approximation of a 2D shape is a possible solution for 2D shape similarity computation, it leads to a low recognition rates since the contour-based representation does not depict a 2D shape fully. Recently, some methods were proposed to compute the similarity of 2D shapes such as legacy drawings and diagrams which contain rich inside structure and can not be described enough by only a contour. A brief survey on online graphics recognition was presented in [7], and some representative approaches include [8-11]. For example, Park et al. [8] suggested an approach in which a complex drawing is decomposed into many adjacent closed loops or blocks. By recursively dividing these blocks into many primitive shapes, a graph structure is built to describe the geometry of a drawing. This method needs a robust graph-matching algorithm and an efficient correspondence searching scheme. Fonseca et al. [9] proposed an approach to retrieve vector drawings in electronic format through hand-sketched queries. First, a simplifying process is adopted to remove irrelevant elements contained in a drawing. Then the polygons, lines, and topological information are extracted from a drawing by using the Inclusion and Adjacency relationships. Finally, they used the graph spectra of the extracted topology to compute the shape descriptors. Tabbone et al. [11] put forward a method to match complex line-drawings. It is based on a notion named $\mathcal{F}$ -signature. As a special kind of histogram, $\mathcal{F}$ -signature has low time complexity and is invariant to fundamental geometrical transformations such as scaling, translation, and rotation.

With the development of 3D computer graphics, a large amount of 3D models have been accumulated in many fields. Up to this point, many methods have been proposed to retrieve 3D models. These methods can be classified into four categories: **(1) Feature-vector based method.** Since a 3D shape can be described by some global geometric properties such as moment [12], tensors [13], density [14], symmetry [15], and some high-level shape features such as geons [16], it is therefore natural to use these features or their combinations to measure the similarity between 3D models. However, when a large number of feature descriptors are used for the query, the

system can not respond quickly because a tremendous amount of computation is needed. Also it is not easy to describe all aspects with a single feature-vector that comprehensively describes the shape. **(2) Statistics-based method.** To overcome the limitation of the feature-based method, some statistics-based methods have been proposed to describe a 3D shape by computing the distribution of some specified feature, such as area or volume distribution along concentric spheres [14], moment or normal distribution [17, 18], and distance distribution [19]. Compared with other methods, the statistical method is not only fast and easy to implement, but also has some desired properties for practical applications such as robustness and invariance. However, these methods are not sufficient to distinguish similar objects with obvious local differences because local shapes are not depicted explicitly. **(3) Topology-based method.** Besides the geometric and physical properties, the topology structure is important information for a 3D shape because human shape perception is partly based on structural decomposition [20]. Topology-based methods have many desired properties, such as intuitiveness, invariance, generality, and robustness. Global features and local features are depicted in the form of a skeleton. It is convenient for users to specify the desired part that they would like to match. The representative methods include Hilaga et al.'s Multiresolutional Reeb Graphs (MRGs) [21], Siddiqi et al.'s shock graph [22] and Sundar et al.'s skeleton graph [23]. However, it is still difficult to compute the topology in a robust and uniform way. **(4) Image-based method.** To imitate human being's uncanny ability to recognize objects from single or multiple views, some methods have been proposed to recognize a 3D shape from multiple 2D views. For example, Cyr et al. [24] adopted a concept named aspect graph to represent a 3D object with a minimal set of 2D views. However, it has to determine the view transitions named "visual events" precisely; otherwise, a large number of views are needed. Funkhouser et al. [25] and Chen et al. [26] proposed algorithms in which the similarity between 3D models is converted to the similarity computation between 2D contours of the 3D models. The similarity between sketches and rendered contours obtained from 3D objects is measured by a 2D analog of spherical harmonics representation. However, 13 and 10 thumbnail images from different directions are needed respectively. Also only contours are considered in these methods. Therefore, significant useful information is lost.

Since it is necessary to provide users with a natural way to input their shape query, developing a natural user interface is also crucial for a searching task. Generally, there are four popular interactions: **(1) Keyword query:** As the most popular way for information retrieval, keywords are also used in the shape retrieval [27] since a

geometric shape can be described by some semantics information, such as name, color, texture, function, manufacturing, materials, file size or ID number. However, defining this semantics information depends on many factors, such as culture, habits and language. **(2) Hierarchy browser:** To reduce the search space, most 2D or 3D shape repositories provide a browsing interface for retrieval purposes. Users are required to traverse a hierarchy by selecting keywords. The data in the library are classified according to the predefined keywords. This paradigm can be implemented easily although heavy manual effort is needed to organize the models. And users need to have some knowledge about the semantics of shapes. **(3) Example query:** An interaction method adopted by most shape retrieval systems is to use an existing shape. Users are allowed to upload a 2D/3D shape or select a result from a previous search as an input. But if a query shape is not available, a user has to create it from scratch. **(4) Sketch query:** Users are allowed to sketch a 2D or 3D shape as query. Compared to the other interaction ways, the sketch-based user interface [25, 26] is more natural and efficient.

3. Bridging the gap between 2D & 3D shapes

In [28], a method was proposed to compute the pose of a 3D model by finding the orthogonal orientations with maximum virtual contact area (VCA). It can be used here to bridge the space gap between 2D & 3D shapes. By this method, a 3D model is represented by three views along the orthogonal axes.

The basic idea stems from the fact that the orthogonal directions that have the maximum VCA (Virtual Contact Areas) are the major principal axes of the 3D model and provide a stable pose. The key step in obtaining the principal axes is to determine the polygons of a 3D object that have the same normal and lie in the same plane. VCA is defined as the bounding area formed by polygons that have the same distance from a predefined point and have the same normal. Generally, a 3D shape $\mathcal{S}$ can be represented as a 3D mesh composed of a series of discrete 2D polygons p_i ($0 \leq i \leq n-1$) with uniform density. In order to efficiently figure out the VCA of a 3D object along a certain direction, we use a triple $\mathbf{S} = \{ p_i \mid (\mathbf{N}_i, A_i, D_i), 0 \leq i \leq n-1 \}$ to represent a 3D shape, in which $\mathbf{N}_i$ represents the normal of polygon p_i , A_i represents the area, D_i represents the signed distance between the mass center C and the polygon p_i . D_i can be the distance between any predefined origin in 3D space and the polygon p_i . In our method, we adopt the mass center as the origin for the sake of simplicity. Although our proposed representation is similar to the CEGI (Complex Extended Gaussian Image) and has the same valuable properties that Kang et al. pointed out in [29], their

purposes are different. Our proposed representation is designed to compute the VCA efficiently, while the EGI (Extended Gaussian Image) and CEGI methods [29] are used to compute the translation parameters and their orientation calculation sums the area of the polygons which have the same normal.

To obtain the direction along which the VCA is the maximum, we need to find all polygons that have the same normal direction and the same distance to the mass center. The direction that gives the maximum VCA is the first principal axis $\mathbf{b}^u$ of the 3D object orientation. To get the next principal axis $\mathbf{b}^v$ of an object orientation, we find the normal that satisfies two conditions: (a) has maximum area; and (b) is orthogonal to the first principal axis. The third axis can then be obtained by performing the cross product between $\mathbf{b}^u$ and $\mathbf{b}^v$:

$$\mathbf{b}^w = \mathbf{b}^u \times \mathbf{b}^v \quad (1)$$

For detailed comparison between our pose estimation method and PCA (Principal Component Analysis) [30] and EGI/CEGI [29] methods, please refer to [28]. In Table 1, a few more examples from the multimedia and engineering domain are presented to show the generality of the proposed method. Some of these objects are formed by assembling different 3D objects, (e.g. objects in (d)). The plant object in (c) shows the robustness of this method since the leaves can be regarded as noise.

Table 1. Some examples of pose estimation results from engineering domains

4. Multiple levels of details (MLD) representation of a 3D model

To depict a 3D model using 2D views, we represent a 3D shape as multiple levels of detail as illustrated in Figure 2. It can be seen that the contour level reflects its global shape by which a user can “guess” the true object to some extent; the silhouette level conveys more shape details using a few more simple addition to the contour sketch. When the detailed shape information is not important, the silhouettes are enough to differentiate two similar objects more confidently compared to the

contours. The third level contains the complete information, including the visual appearance and the occluded structure, by which a user can determine its shape precisely. In practice, especially in engineering fields, there are a lot of models that look similar from visual appearance but still have a different inside structure. Therefore, we consider the complete details of the object. However, the projected shapes at the three levels are different from different view points. With the help of the pose determination method, we project a 3D shape onto the six faces along the principal axes to represent the MLD. At the contour level, there are three different views along the principal axes; at the silhouette level, there are six different views; and at the full level, we use the traditional drawing-like views to represent the drawing level, and there are three different views along the principal axes. As for the details about how to generate these views efficiently, please refer to [28].

Figure 2. Multiple levels of detail: (a) a 3D model; (b) contour level; (c) silhouette level; (d) drawing level.

5. 2D shape descriptors

In [31], two methods have been proposed to retrieve 2D shapes by measuring their shape similarity. The first approach represents a 2D shape as a spherical function by transforming it from 2D space into 3D space and then employs a fast spherical harmonics transformation to get a rotation invariant descriptor. The second method represents the 2D shape from a statistics perspective as a distance distribution between pairs of randomly sampled points. Both the representations have many valuable advantages: they are invariant to affine transformation, insensitive to noise or cracks, simple, and fast.

5.1. 2.5D spherical harmonics transformation

As a robust rotation invariant descriptor, spherical harmonics representation has been successfully applied to 3D shape matching [32]. A 2D analog of the spherical harmonics was used in [25] to extract a series of rotation invariant signatures by dividing a 2D silhouette shape into multiple circular regions. However, this method has two major limitations as mentioned in [31]: one-to-multiple correspondence and instability caused by shape perturbation. In order to overcome these limitations and thus obtain a set of robust rotation invariant signatures for a 2D shape, we propose a strategy called 2.5D spherical

harmonics representation, which can extract a series of rotation invariants by transforming a 2D shape from 2D space into 3D space uniquely. The name “2.5D” arises from the fact that a 2D shape is represented in a 3D space.

Figure 3. An example of 2.5D spherical harmonics representation: (a) is a 2D shape; and (b), (c), and (d) show the 3D representation of the drawing from different perspectives.

The proposed transformation is explained by the following steps:

(1) Given a 2D shape D , compute its axis-aligned bounding box B .

(2) Determine a sphere S that satisfies three conditions: Its center $\mathbf{c}$ is aligned with the center of bounding box B ; its radius r is equal to half the diagonal length of bounding box B ; and the 2D shape lies in the equator plane of sphere S .

(3) Generate a set of uniformly spaced rays, which start from the sphere center $\mathbf{c}$ and located in the plane where the 2D shape lies, and compute the intersections between these rays and 2D shape D . The resulting intersection point set $\{p_i\}$ can be regarded as an approximating representation of 2D shape D . Since the intersection points are distributed along a certain angle θ_i with respect to axis x , they can also be represented by θ_i and d_i , i.e., $p_i = f(\theta_i, d_i)$, where d_i is the distance between point p_i and the sphere center $\mathbf{c}$. However, along a single θ_i , there might be multiple intersection points. To make use of the valuable property of the spherical harmonics transformation, we transform all intersection points $\{p_i = f(\theta_i, d_i)\}$ into a spherical function form $\{p_i = f(\theta_i, \varphi_i, d_i)\}$ by introducing a new variable φ_i . To ensure that each intersection point p_i corresponds to a unique (θ_i, φ_i) , we use a simple transformation to determine φ_i as shown in Equation 2.

$$\varphi_i = \arctan\left(\frac{d_i}{r}\right) \quad (2)$$

where r is the radius of sphere S . For a given 2D shape, the radius r is determined uniquely, while for an intersection point p_i , d_i is also uniquely determined. Therefore, for an intersection point p_i , the corresponding φ_i obtained by Equation 2 is also unique. Consequently, a 2D shape is uniquely transformed into a 3D spherical representation, i.e., the correspondence between a 2D shape and its spherical function is one to one mapping. We name this process a 2.5D transformation. Figure 3 shows an example of this transformation for a 2D shape example. From this example, we notice that the geometric information is represented clearly in 3D space along the surface of a cylinder.

Finally, to obtain the rotation-invariant, we use the fast spherical harmonics transformation method in which a spherical function of bandwidth B is sampled on the $2B$ -many Chebyshev points and not the B -many Gaussian points. These sampled points form a $2B \times 2B$ equiangular grid along the longitude and latitude of a sphere.

5.2. 2D shape histogram

To measure the similarity between 2D shapes, we derive a 2D analog called 2D shape histogram on the basis of [19]. Here we summarize it as the following steps:

(1) Discretize all entities contained in a drawing into a set of line segments.

(2) Sample enough points on the line segments uniformly.

(3) Select two points randomly from the point set and compute their Euclidean distance.

(4) Calculate the distance distribution of all sampled point pairs, and build a histogram.

For two histograms, i.e., $\mathbf{H}_1$ and $\mathbf{H}_2$, their similarity W is given by:

$$W(H_1, H_2) = L_n(H_1, H_2) = \sqrt[n]{\sum_{i=0}^h (H_1(i) - H_2(i))^{1/n}} \quad (3)$$

6. Similarity computation

For 2D shapes, we can use Equation (4) to compute their similarity directly. Since a 3D model is represented by three orthogonal 2D views, a procedure is intuitively needed to find the correspondence between the views from different models. To simplify this process, we use the minimum values between all possible view-pairs to represent their similarity:

$$W(M_1, M_2) = \sum_{k=0}^2 \min\{W(m_{1i}, m_{2j}) \mid 0 \leq i \leq 2, 0 \leq j \leq 2\} \quad (4)$$

Where m_{1i} represents view i of model M_1 , m_{2j} represents view j of model M_2 , (m_{1i}, m_{2j}) represents a view-pair between 3D models. It is worth pointing out that a view-pair can not be computed twice.

When a user tries to find a 3D shape using a 2D shape query or vice versa, we compute the similarities between the 2D shape and each view of the 3D shape. The highest similarity will be regarded as the similarity between a 2D shape and a 3D shape.

To make sure that the three levels can be applied to the whole 3D model space, we propose applying a weight value to each level, and using their combined confidence as the overall similarity. The similarity confidence C using the three levels is expressed as

$$C = w_{L1}S_{L1} + w_{L2}S_{L2} + w_{L3}S_{L3} \quad (5)$$

where S_{L1} , S_{L2} , and S_{L3} are the similarities of the three levels, and w_{L1} , w_{L2} , and w_{L3} are the weights values of the respective levels. Higher weight value means that the corresponding level plays a more important role in describing the 3D model. Users can adjust these weights to find the most similar shapes according to their needs.

Given the two shape similarity methods, it is natural to try combining them together to achieve a better performance. We propose applying a weight value to each of the two shape similarity methods and using their combined confidence to measure overall similarity. The similarity confidence T using the two approaches is represented as

$$T = w_s C_s + w_d C_d \quad (6)$$

where C_s is the similarity obtained by 2.5D spherical harmonics method, C_d is the similarity obtained by 2D distance histogram method, and w_s and w_d are the weight values of the respective methods. Higher weight value means that the corresponding method plays a more important role in differentiating a 3D model.

7. Sketch-based user interface

With the help of the proposed framework, we implement a sketch-based user interface (as Figure 4 shows) supporting 2D & 3D shape retrieval. Figure 4(a) is an example about 3D model retrieval, while Figure 4(b) is an example about 2D legacy drawing retrieval.

Figure 4. 2D&3D shape retrieval using freehand sketches: (a) is a 3D model retrieval example using freehand sketches; and (b) is a 2D drawing retrieval example using freehand sketches.

The query process is similar to the process that engineers use to express their shape concepts on a piece of paper. Users can not only emphasize certain shape features by specifying weights for corresponding views, but also do several local editing operations (such as draw and erase) on the views obtained from the retrieved models. The whole retrieval process forms a loop by which users can repeatedly refine the search results until satisfactory results are obtained. Experiments have been conducted in [31] to demonstrate the validity and

efficiency of this sketch-based user interface by 3D model retrieval.

8. Experiments

In order to test the performance of ShapeLab, we have built a benchmark, namely 2D&3D Shape Benchmark, which includes 2,000 2D drawings and 801 CAD models from industrial fields. All the data is available publicly from our website: <http://www.purdue.edu/shapelab>. The 2D drawing benchmark consists of 85 clusters from simple to complex according to their geometric shape, while the 3D model benchmark consists of 42 clusters of similar parts. The Precision-Recall (PR) curve is widely employed to evaluate the performance of a retrieval system. Recall measures the ability of a system to retrieve the desired objects, while precision measures the ability of a system to weed out what users do not want. A perfect retrieval retrieves all relevant models consistently at each recall level, producing a horizontal line at precision = 1.0. However in practice, precision decreases with increasing recall. The closer a PR curve tends to the horizontal line at precision = 1.0, the better the information retrieval method.

Extensive experiments have been conducted in [31] and [33] to evaluate the performances of 2D and 3D shape retrieval using the above mentioned benchmarks. Figure 5 shows the precise-recall curves about the 2D shape retrieval evaluation. For detailed descriptions and original references of these shape representations please refer to [31].

Figure 5. Precise-Recall curves for 2D shape retrieval evaluation

From this precise-recall curve, it is obvious that the four contour-based retrieval methods have the lowest performance. Therefore, it is safe to conclude that the contour alone is not a good way to describe the shape of a 2D drawing. Our two proposed methods have almost the same performance on the whole. In a strict sense, the 2.5D spherical harmonics transformation method is better than 2D shape histogram method. In practice, we have

found that the 2.5D spherical harmonics transformation method is good at differentiating drawings with obvious structural shape differences. The 2D shape histogram method is good at differentiating 2D drawings with similar contour but different inside structure. Therefore, in practice, we combine the two methods together so that higher retrieval accuracy is achieved.

The computed precision recall curves for the 3D model retrieval were compared against existing shape descriptors, including 3D Shape Distributions, Spherical Harmonics, and Solid Angle Shape Histograms. For detailed descriptions and original references of these shape representations please refer to [33, 34]. In their recent work Iyer et al. [33] computed a precision recall curves which compared our proposed methods and other available shape descriptors. Both our shape descriptors demonstrated consistently higher precision recall curves as compared to the other shape representations. Further, the 2.5D Spherical Harmonics method performed better than the 2D Spherical Harmonics method. When the two shape descriptors and the multiple levels of details are combined together (i.e., MLD), the experiments suggest that the performance is considerably better than most existing methods including the original 2.5D Spherical Harmonics and 2D Spherical Harmonics methods. As shown in Figure 6, the precision recall curve for the MLD method is consistently higher than the other methods and especially higher at higher recall values. As for the details of the experiments, please refers to [33].

Figure 6. Precise-Recall curves for 3D model retrieval evaluation

9. Conclusion

In this paper, we introduce a 2D & 3D shape retrieval system called ShapeLab and describe the involved key algorithms. With the help of this system, users can retrieve 2D & 3D shapes using freehand sketches. All similarity computations are performed uniformly in 2D space. Our experiments show that the proposed system can achieve a better performance as compared to other

available approaches. In engineering practice, besides geometric information, a 2D or 3D shape also contain some other important information, such as dimension, material descriptions or some requirements related to production. To make use of the engineering information efficiently and provide designers with a more accurate retrieval system, we will focus on two related problems in the future: (1) how to provide designers with a parameterized query to retrieve 2D shapes or 3D models; and (2) how to organize the non-shape information to aid the retrieval process.

10. Acknowledgement

We would like to acknowledge the financial support provided by the 21st Century Research and Technology Funds provided by the State of Indiana.

11. References

- [1] H. Kauppinen, T. Seppanen, and M. Pietikainen, "An Experimental Comparison of Autoregressive and Fourier-based Descriptors in 2D Shape Classification", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2): 201–207, 1995.
- [2] F. Mokhtarian, S. Abbasi, and J. Kittler, "Robust and Efficient Shape Indexing through Curvature Scale Space", *Proc. British Machine Vision Conference*, Edinburgh (UK), pp. 53-62, 1996.
- [3] H. Freeman, "Computer Processing of Line-drawing Images", *Computing Surveys*, 6(1):57-97, 1974.
- [4] A. Khotanzad, and Y.H. Hong, "Invariant Image Recognition by Zernike Moments", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):489–497, 1990.
- [5] H. Alt, B. Behrends, and J. Blömer, "Approximate Matching of Polygonal Shapes," *Proc. the Seventh Annual Symposium on Computational Geometry*, pp. 186-193, North Conway (USA), 1991.
- [6] S. Jeannin, "Mpeg-7 Visual Prt of eXperimentation Model Version 9.0", In *ISO/IEC JTC1/SC29/WG11/N3914, 55th Mpeg Meeting*, Pisa (Italia), 2001.
- [7] W.Y. Liu, "On-Line Graphics Recognition: State-of-the-Art", *Proc. the 5th International Workshop on Graphics Recognition Recent Advances and Perspectives*, pp. 289-302, Barcelona (Spain), 2003.
- [8] J. Park, and B. Um, "A New Approach to Similarity Retrieval of 2D Graphic Objects Based on Dominant Shapes", *Pattern Recognition Letters*, 20: 591–616, 1999.
- [9] M.J. Fonseca, A. Ferreira, and J.A. Jorge, "Towards 3D Modeling using Sketches and Retrieval," *Proc. Eurographics Workshop on Sketch-Based Interfaces and Modeling (SBM'04)*, Lisboa (Portugal), pp. 127-136, 2004.
- [10] P. Fränti, A. Mednionogov, V. Kyrki, and H. Kälviäinen, "Content-based Matching of Line-drawing Images Using the Hough Transform", *International Journal on Document Analysis and Recognition*, 3 (3): 117-124, 2000.
- [11] S. Tabbone, L. Wendling, and K. Tombre, "Matching of Graphical Symbols in Line-drawing Images Using Angular Signature Information", *International Journal on Document Analysis and Recognition*, 6(2):115-125, 2003.
- [12] M. Elad, A. Tal, A., and S. Ar, "Content Based Retrieval of VRML Objects: An Iterative and Interactive Approach", *Proc. 6th Eurographics Workshop on Multimedia 2001*, pp. 107-118, Manchester, UK, 2001.
- [13] M.T. Suzuki, "A Web-based Retrieval System for 3D Polygonal Models", *Proc. Joint 9th IFSA World Congress and 20th NAFIPS International Conference (IFSA/NAFIPS2001)*, pp.2271-2276, Vancouver, 2001.
- [14] C. Zhang, and T. Chen, "Indexing and Retrieval of 3D Models Aided by Active Learning", *Proc. ACM Multimedia 2001*, pp. 615-616 Ottawa, Ontario, Canada, 2001.
- [15] M. Kazhdan, B. Chazelle, D. Dobkin, T. Funkhouser, and S. Rusinkiewicz, "A Reflective Symmetry Descriptor for 3D Models", *Algorithmica*, 38(2): 201-225, 2003.
- [16] K. Wu, and M. Levine, "Recovering Parametric Geons from Multiview Range Data", *Proc. CVPR*, pp.159–166, 1994.
- [17] B. Horn, "Extended Gaussian Images, *Proc. IEEE* 72, 12(12), pp.1671-1686. New Orleans, USA, 1984.
- [18] E. Paquet, and M. Rioux, "Nefertiti: A Tool for 3-D Shape Databases Management", *SAE Transactions: Journal of Aerospace* 108, pp.387–393, 2000.
- [19] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape Distribution", *ACM Transactions on Graphics*, 21(4): 807-832, 2002.
- [20] I. Biederman, "Recognition-by Components: A Theory of Human Image Understanding", *Psychological Review*, 94(2):115-147, 1987.
- [21] M. Hilaga, Y. Shinaagagawa, T. Kohmura, and T.L. Kunii, "Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes", *Proc. SIGGRAPH 2001, Computer Graphics Proceedings, Annual*

Conference Series, pp. 203–212, Los Angeles, USA, 2001.

[22] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. Zucker, “Shock Graphs and Shape Matching”, *Computer Vision*, 35(1): 13-20, 1999.

[23] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson, “Skeleton Based Shape Matching and Retrieval”, *Proc. Shape Modeling International 2003*, pp.130-142, Seoul , Korea, 2003.

[24] C.M. Cyr, and B.B., Kimia, “3D Object Recognition Using Shape Similarity-Based Aspect Graph”, *Proc. 8th International Conference On Computer Vision*, pp.254-261, Vancouver, Canada, 2001.

[25] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, “A Search Engine for 3D Models”, *ACM Transactions on Graphics*, 22(1): 83-105, 2003.

[26] D.Y. Chen, X.P. Tian, Y.T. Shen, and M. Ouhyoung, “On Visual Similarity Based 3D Model Retrieval”, *Computer Graphics Forum (Eurographics’2003)*, 22(3):223--232, 2003.

[27] National Design Repository,
<http://www.designrepository.org>

[28] J.T. Pu, and R. Karthik, “An Automatic Drawing-like View Generation Method from 3D Models”, *Proc. ASME IDETC/CIE 2005, 25th Computers and*

Information in Engineering (CIE) Conference, Long Beach (USA), pp. 301-320, 2005.

[29] S.B. Kang, and K. Ikeuchi, “The Complex EGI: A New Representation for 3D Pose Estimation”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(7): 1993.

[30] M. Petrou, and P. Bosdogianni, *Image Processing: The Fundamentals*, John Wiley, 1999.

[31] J.T. Pu, and R. Karthik, “On Visual Similarity based 2D Drawing Retrieval”, *Computer Aided Design*, Vol. 38(3): 249-259, 2006.

[32] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, “Rotation Invariant Spherical Harmonic Representation of 3D Shape Descriptors”, *Proc. the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, Aachen (Germany), pp.156-164, 2003.

[33] N. Iyer, S. Jayanti, and K. Ramani, “An Engineering Shape Benchmark for 3D models”, *Proc. ASME IDETC/CIE 2005, 25th Computers and Information in Engineering (CIE) Conference*, Long Beach (USA), pp. 293-300, 2005.

[34] N. Iyer, S. Jayanti, S., K. Lou, Y. Kalyanaraman, and K. Ramani, “Three Dimensional shape searching: state-of-the-art review and future trends,” *Computer-Aided Design*, 37(5): 509-530, 2005.