2009 Fifth International Conference on Image and Graphics

A Solution to Efficient Viewpoint Space Partition in 3D Object Recognition

Xiao Yu Huimin Ma Shaodi You Ze Yuan
Dept. of Electronic Engineering
Tsinghua University
Beijing, 100084
x-yu05@mails.thu.edu.cn, mhmpub@tsinghua.edu.cn, {youshaodi, yuanze.2008}@gmail.com

***Abstract*—Viewpoint Space Partition based on Aspect Graph is one of the core techniques of 3D object recognition. Projection images obtained from critical viewpoint following this approach can efficiently provide topological information of an object. Computational complexity has been a huge challenge for obtaining the representation viewpoints used in 3D recognition. In this paper, we discuss inefficiency of calculation due to redundant nonexistent visual events; propose a systematic criterion for edge selection involved in EEE events. Pruning algorithm based on concave-convex property is demonstrated. We further introduce intersect relation into our pruning algorithm. These two methods not only enable the calculation of EEE events, but also can be implemented before viewpoint calculation, hence realizes view-independent pruning algorithm. Finally, analysis on simple representative models supports the effectiveness of our methods. Further investigations on Princeton Models, including airplane, automobile, etc, show a two orders of magnitude reduction in the number of EEE events on average.**

Keywords- viewpoint space partition, object recognition, concave and convex property

I. INTRODUCTION

Three-dimensional object viewpoint space partition is the core technique in 3D object recognition based on aspect graph [1], [2]. Viewpoint space partition refers to the process of dividing viewpoint sphere (Gauss sphere) into several connected subsets, based on invariance of topology for the line drawing of projection pictures [2]. When projecting on an object from one particular subset, all views are isomorphic in terms of shape similarity character and topology; otherwise, from different subsets, we will get rather different topology and shape similarity characters. Hence, we can represent a 3D object by several two-dimension projection pictures, aspect graph, according to the differences in topology.

The problem with current viewpoint space division method [3] based on critical event is that they require full-scale traversal of 3D grid, which inevitably increases the number of subsets. On the other hand, many subsets do not have actual meaning since corresponding critical visual events (changes of topology for the line drawing of projection pictures) are nonexistent.

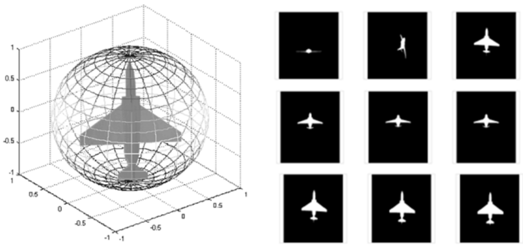

Figure 1. Viewpoint space partition method and corresponding aspect graphs of a model

For ideal 3D object constructed by triangle planes, pruning strategy mainly aims at Edge-Vertex (EV) event and Edge-Edge-Edge (EEE) event. Pruning methods for EV event has been illustrated in reference [4]; those for EEE events have not been introduced yet. This paper proposes one novel scheme independent with viewpoints, which can effectively reduce the number of potential EEE events, enable the calculation of EEE events.

II. PRUNING ALGORITHM FOR EEE EVENTS

Models used in this paper are all ideal models constructed by triangle planes, as shown in the Figure 2.

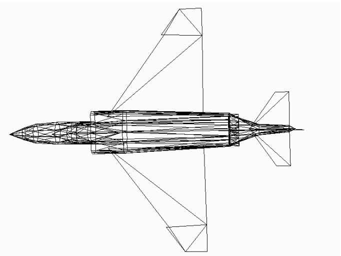

Figure 2. Ideal model of F4 constructed by triangle planeș

As for these models, what we care about are EV event and EEE event involving edge, plane, and points [5], as shown in the Figure 3. An EV-event occurs when an image vertex intersects an image edge. This happens when the corresponding object vertex and non-adjacent object edge are aligned along an extended sight line from the viewpoint. An EEE-event occurs when three image edges intersect at a point. Such an event happens when the three

978-0-7695-3883-9/09 $26.00 © 2009 IEEE
DOI 10.1109/ICIG.2009.65

corresponding pairwise non-adjacent object edges are aligned along an extended sight line from the viewpoint.

From basic critical event, we can get equation of the cutting plane. Cutting place determined by EV event has a general form of: [6]

$$ax + by + cz + d = 0 \tag{1}$$

The cutting surface determined by EEE event is a quadratic surface: [6]

$$\begin{aligned} &a_0x^2 + a_1y^2 + a_2z^2 + b_0xy + b_1xz + b_2yz \\ &+c_0x + c_1y + c_2z + d = 0 \end{aligned} \tag{2}$$

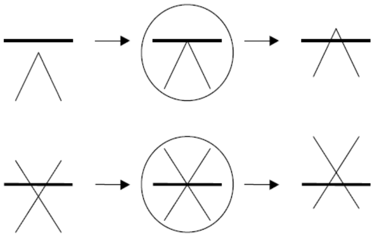

Figure 3. EV-event (top row) and EEE-event(bottom row). Circled one indicates the occurrence of critical events (From [5])

The intersection of spatial cutting surface and viewpoint space (gauss sphere) is called cutting edge. From the relationship between different cutting edges, we can find out all closed region divided by these cutting edges and further determine subsets of viewpoint space. The boundaries of these points set are a group of cutting boundary of viewpoint space. As we pick representative viewpoint from each subsets, results of viewpoint division can be presented.

Equations of cutting surfaces correspond to one and only one EV, EEE events. If you include more critical events, the Gauss Sphere will be divided into more subsets inevitably, and result of viewpoint space partition will be more delicate. The number of EV events for different models have been shown in reference. For EEE event, since the involvement of three non-coplanar lines, compared with EV events, which only involve one point and one edge, the interception relations and partial visibility, as well as determining condition needed, are much more complicated. Meanwhile, cutting surface of EEE event is quadratic surface [6], which has 17 different situations for different configurations of parameters. Intuitive pruning is more difficult compared with pruning for EV events [6]. Table 1 shows the number of EEE events calculated without any pruning algorithm for different models [7]. The huge burden induced by these numbers has surpassed normal computing capability. Among all potential EEE events, only a small portion happen in reality, while the rest of them do not have actual meaning [6]..

Therefore, in the process of viewpoint space partition, selection of EV and EEE events is vital. If nonexistent EV and EEE events get chosen, corresponding viewpoint space partition will be of little use, as it does not have any physical significance. In that sense, introduction of pruning algorithm not only reduces computational complexity but also guarantee the correctness of the partition. In this paper, two schemes to reduce EEE events in the problem of viewpoint space partition are proposed. The first scheme is to prune EEE events based on convex-concave property. This pruning algorithm bases on the principle that the concave polyhedrons only introduce nonexistent events. The second is to prune based on local interception. The basic idea is that if three edges can be masked by any other faces, the EEE event is not a valid EEE event.

TABLE I. NUMBER OF EEE EVENTS IN DIFFERENT 3D MODELS

Name of models	**F4**	**F16**	**Foot_bone**
number of EEE events	347800	483712	2287356

III. PRUNING ALGORITHM BASED ON CONVEX-CONCAVE PROPERTY

The target of pruning is to eliminate nonexistent visual events. These events, though have corresponding cutting surfaces, have no actuality, and only introduce the events number of an order of magnitude. For EEE event, described by quadratic surface, pruning process may not be intuitive. On the other hand, if we accomplish pruning via geometric observation, then the process depends on viewpoints, hence loses versatility. Therefore, beginning with massive analysis on real objects, this paper proposes condition for real EEE events, which is set as criteria for pruning.

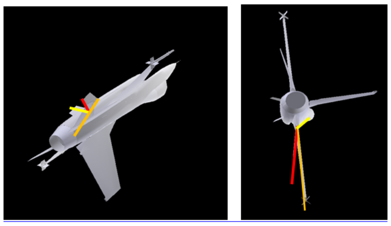

Figure 4. Real EEE events for F16 model (edges involved are highlighted in colored lines).

Models used in this paper come from Princeton Model, including airplane, racecar, etc. Real EEE events for F16 model are shown in Figure 4.

Thus, in 3D object model, EEE visual event could happen, and it contains information not included in EV events. This particular information deserves our consideration and calculation in viewpoint partition process, in order to get more accurate results.

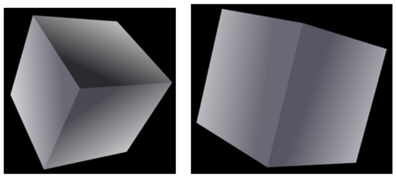

Figure 5. No EEE of actual meaning event for concrete cube.

In Figure 5, for the most fundamental model, concrete cube, we cannot find EEE event if we do not take extension of edges into account.

First, confirm that you have the correct template for your paper size. This template has been tailored for output on the US-letter paper size. If you are using A4-sized paper, please close this template and download the file for A4 paper format called “CPS_A4_format”.

The situation is much the same for convex octahedron, as shown in the graph as shown in Figure 6.

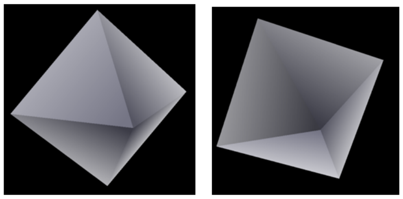

Figure 6. No EEE of actual meaning event for convex octahedron.

When constructing a model with potential EEE event, e.g. Figure 7, we can discover that it is possible only when plane related to edge involved in EEE event is in concave part of the model. In other words, we cannot construct a convex polyhedron with real EEE event.

Based on that, this paper puts forward a hypothesis involving of concave part of an object is necessitated in actual EEE events. Edge of other positions cannot induce EEE event. Hence, we can determine the actuality of EEE event based on local concave and convex property.

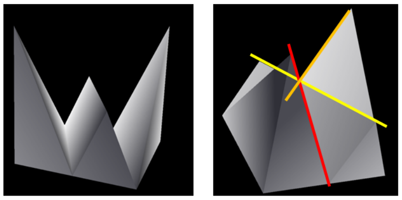

Figure 7. A potential EEE event for a model with concave part.

For this kind of property, though many algorithms related to convex polyhedron have been introduced; none of them has application in 3D recognition process. For the first time, we introduce concave and convex property into viewpoint space partition, proposes the idea of using local concave and convex property as determining criteria for EEE events. In the following section provides corresponding demonstration.

A. The principle of local convexity and its proof

In this section, the following theorem will be proved: For convex polyhedron, EEE event does not exist.

To prove the theory, we use the following three criterions.

Criterion 1: If any face of a convex polyhedron is expanded to a plane, the rest of the polyhedron should be located at the same side of the plane. This is one of the criterions of polyhedron. The outer norm and inner norm of each face could be decided from this criterion.

Criterion 2: All the faces on a polyhedron are convex polygons. If a polyhedron has vertexes, edges and face in the same plane, they are on the same convex polygon. This criterion can be deduced from criterion 1 easily.

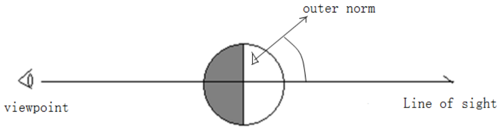

Figure 8. Spatial relation between viewpoint and object.

Criterion 3: for each face on an opaque polyhedron, if the angle between the line of sight and the outer norm of the face is a critical angle, the face is sheltered from other faces on the same polyhedron. This criterion is illustrated in Figure 8. Only those faces represented in grey can be seen.

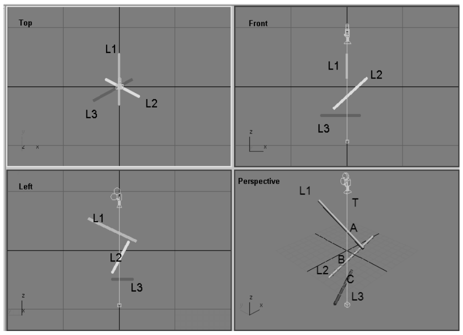

Figure 9. Different Spatial relation of three edges involved in EEE events.

We now prove EEE event does not exist on convex polyhedron. For the situation that three edges do not intercept each other, as shown in Figure 9, L1, L2, L3 are involved in an EEE event from the top view. We prove the three edges are not on the same polyhedron. Consider the face related to L2. The only situation the face does not violate Criterion 1 is: the face is in the plane formed by T (line of sight) and L2. We depict the T-L2 plane as Figure 10.

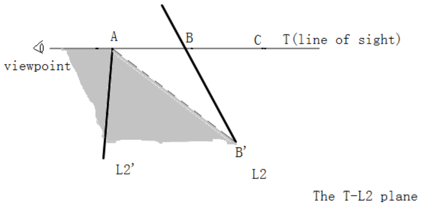

Figure 10. The interception of L2 by L2' in the T-L2 plane.

As shown in Figure 10, one of the vertexes of L2 which is not on line T is denoted as B’. From Criterion 2: A, B, B’ are in the same polygon. As A is a vertex of the polygon, one of those edges enclosed the polygon should be in the grey domain and pass through A. This edge is denoted as L2’. We mention that L2’ will occlude L2. That is to say, what can be seen from the viewpoint is another event of L1, L2’ and L3, but not event of L1, L2 and L3.

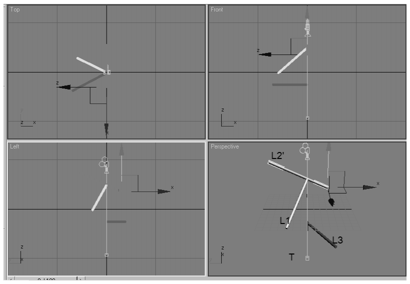

Figure 11、 The interception of two edges near the viewpoint.

We prove this kind of event does not exist on polyhedron, which the two edges near the viewpoint have an interception. As shown in Figure 11, we consider the face related to L3. If the outer norm of the face and the line of sight(T) has an obtuse angle, then L1 and L2’ are on the outer side of the polyhedron, this will violate criterion 1. If the outer norm of the face and the line of sight has a critical angle, from criterion 3, this face cannot be seen. Hence, the only situation left is the face lies in the T-L3 plane. The same as the first situation, we can prove that L3 will be occluded by another edge L3’ which pass through A. Hence, this kind of event is occluded by a normal three-edge interception event.

The last situation which two latter edges have an interception, can be proved non-exist on polyhedron in the same way.

To sum up, for local convex polyhedron, EEE event does not exist. From the procedure of our demonstration, we may focus on local convex property. If any other parts of polyhedron are on the same side of the face, then it is called partial convex and EEE event related to this face cannot happen.

B. Mathematical expression and realization of pruning algorithm based on Concave and Concave property

1) Elimination of edge-on-plane.

Edge that is not formed by two meeting surfaces is called edge-on-plane. From the demonstration, we understand edge-on-plane does not contribute to real EEE event, because if not, line of sight should be parallel to two planes constructing the edge, which means other edges and edge-on-plane will mask each other. Therefore, edge-on-plane can be eliminated. In order to determine whether an edge is on a plane, we can calculate the normal vector of two planes related to the edge. If two vectors are parallel to each other, then the edge should be on a plane, thus can be eliminated.

2) Determine Concave and Convex Property of polyhedron.

According to the definition of convex polyhedron, we expand a face to an infinite plane, if all other faces are located at one side of the plane, then the polyhedron is convex. We further promote this definition, for one particular face of the object, if other parts of the object are located at one side, then it is convex. Referring to the demonstration in section 3.1, faces of local convex property do not contribute EEE event. In order to determine local convex property, following method can be used: suppose the normal vector of the face is $\vec{n} = (n_x, n_y, n_z)$, purpose the coordinate of one random point on face is $P = (px, py, pz)$,coordinate of vertex $V_i = (x_i, y_i, z_i)$ $i = 1, 2..$ that are not on the face if inner product of $\vec{n}$ and PV_i are of the same sign, then the face is locally convex, edges on this face can be eliminated.

Table 2 lists the number of EEE events with and without pruning. After the introduction of pruning algorithm based on convex and concave, take F4 model for example, the number of EEE events reduced from 347800 to 21369, which indicates a drop in magnitude. Consequently, this particular pruning algorithm is reasonable and necessary.

TABLE II. NUMBER OF EEE EVENTS FOR DIFFERENT 3D MODELS WITH AND WITHOUT PRUNING.

Name of models	F4	F16	Foot_bone
before pruning	347800	483712	2287356
after pruning	21369	216562	1612170

IV. PRUNING ALGORITHM BASED ON LOCAL INTERCEPTION

From statistics in the form above, for relatively simple model F4, after the introduction of convex and concave property, the reduction in the number of EEE events is dramatic. For more complicated models, since condition for local convex cannot be easily satisfied, the effect is not superior enough. The number of edges eliminated is limited. But for sophisticated models, the interception relations are quite complicated as a result, which excludes the actuality of considerable amount of EEE events. Therefore, while we introduce pruning algorithm based on convex and concave property, pruning algorithm based on interception relation is necessary as well.

A. Principle of pruning algorithm based on local interception

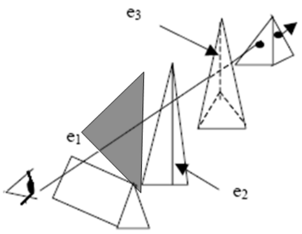

Figure 12. Non-existent EEE event due to interception of the gray plane.

As shown in Figure 12, situation shown in Figure 12 cannot be eliminated by convex and concave property, because in previous pruning algorithm only consider local circumstances, but the impact of interception of other faces is not taken into account. However, in reality, three edge e1, e2, e3 can be masked by any other faces, which eliminates the possibility of EEE event like as shown in Figure 12.

Considering situations like this, we adopt pruning algorithm based on interception relations.

Step 1: First, implement pruning algorithm based on concave and convex property.

Step 2: For every EEE event, determine whether there is a face blocking any pairs of edges among the three involved. If we can find such a face, then this EEE event should be eliminated. Otherwise, after traversal of all faces, if we cannot find such a face, then this EEE event can be reserved.

This kind of pruning process is implemented before calculation of viewpoints, hence independent with views.

In order to determine whether two edges are blocked by a triangle, all we need to do is to check whether vertex of one edge is blocked from vertex of the other edge. This converts the problem into pruning of interception relation between point and point. If vertexes of two edges are all blocked by one particular triangle, then these two edges is blocked from each other.

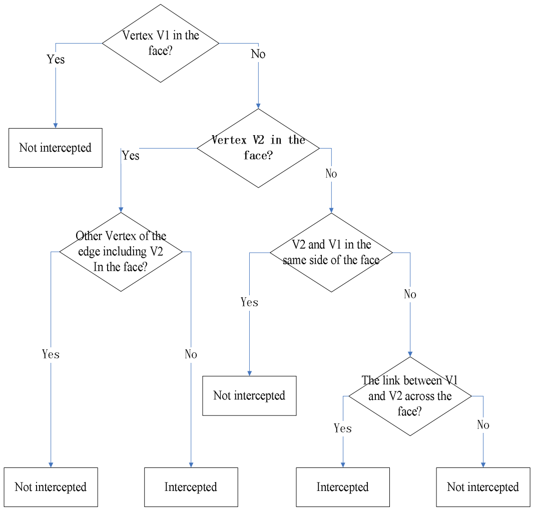

Figure 13. Flow chart of judgement of interception between two vertexes.

Since relative spatial relations between three non-coplanar lines cannot be efficiently determined estimated, we have to consider the interception relations between any two non-coplanar lines. To be more specific, that is three pairs of interception problems between two edges. For two vertexes (V1 and V2), an effective way to determine whether they are blocked by a face (constituted by three vertexes) is illustrated in Figure 13[4].

B. *Results given by pruning algorithm based on Interception Relationship*

Corresponding numbers with multiple pruning algorithms are listed in Table 3. After introduction of pruning algorithm based on local convex and further inclusion of pruning based on interception, again we take the F4 model as an example, the number of EEE event reduces from 21369 to 5293, and another drop in magnitude. As a conclusion, with pruning algorithms, the number of EEE events is reduced significantly, indicating only a few EEE events have actual meanings after pruning process. Naturally, partition given by this scheme should be quite different from the original version.

TABLE III. NUMBER OF EEE EVENTS FOR DIFFERENT 3D MODELS WITH AND WITHOUT PRUNING

Name of Models	F4	F16	Foot_bone
before pruning	347800	483712	2287356
after pruning based on Local Convexity	21369	216562	1612170
further after pruning based on Interception	5393	37556	31585

V. RESULTS

We use the same model for program with and without pruning algorithms. The comparison proves the correctness and effectiveness of the pruning process.

A. *Correctness of pruning algorithm applied to simple models*

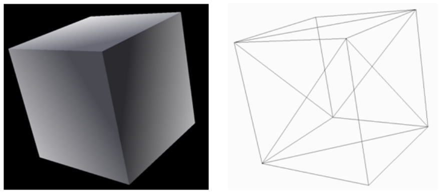

Figure 14. Sketch graph and corresponding model for a cube.

Take the simple cube as example (shown in Figure 14). The results with and without the application of pruning algorithms, EV and EEE events are taken into account simultaneously, are illustrated below. Sphere in the graph is Gauss sphere, regions with different color indicate separate partitions, and points stand for representative view points of corresponding subsets. Gauss sphere is expanded to rectangle.

As shown in Figure 15, before introduction of pruning algorithm for EEE event, equations of nonexistent EEE events have been calculated, while they do not have actual meanings. Result of partitions is incorrect either; redundant viewpoint space can be integrated. With the application of pruning algorithm, the result is significantly simplified. Partition is the same as that given by EV event alone.

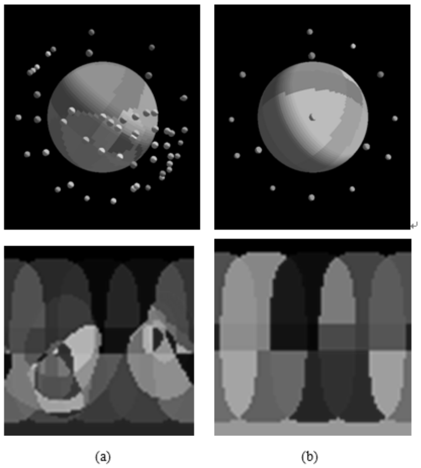

Figure 15. Critical viewpoint for cube model (a) result of space partition without pruning and (b) with pruning.

B. Effectiveness of pruning algorithm

We apply our algorithm to five models of five categories, those are airplane, F4 and F16; car; footbone; generic [8], illustrated in Figure 16.

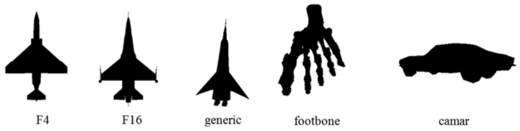

Figure 16. Sketch graphs for the objects.

Take F4 for example, after introduction of pruning algorithm, the number of viewpoint space partitions reduce from 112 to 96. Result is illustrated below:

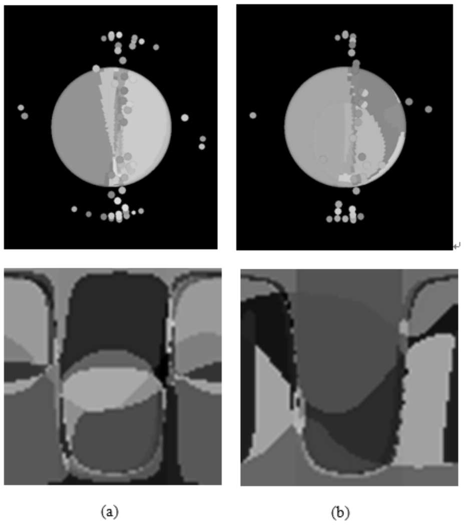

Figure 17. Critical viewpoint for F4 model (a) result of space partition without pruning and (b) with pruning.

In Figure 17, result given is unlike from that without pruning algorithm for EEE event because different EEE events are selected, and the equations for partition are different. As a matter of fact, EEE events selected without pruning algorithm are nonexistent, and hence, partition of viewpoint is not rooted in actual critical visual events. Therefore, introduction of pruning algorithm can guarantee the effectiveness of pruning algorithm.

VI. Conclusion

This paper implements viewpoint space partition for 3D object with triangle grid representation. Regarding redundancy of potential visual events, it systematically discusses the selection of edges involved in EEE events. For first time, we propose pruning algorithm based on concave and convex property, and provide theoretical demonstration. We further introduce pruning algorithm based on interception, reducing the number of EEE events significantly, much closer to its actual number, enable the calculation of EEE events. Since these two pruning processes are accomplished before calculation of viewpoints, we realize view-independent pruning processes. Experiments prove pruning algorithm realize effective viewpoint space partition. In 3D object recognition, it can effectively reduce time needed and system's real-time performance.

References

[1] Plantinga.H, and Dyer C.R, "Visibility, Occlusion, and the Aspect Graph, " International Journal of Computer Vision, vol. 5, no. 2, pp. 137-160, Nov. 2004.

[2] J.X. Zeng, Y.M. Lu, M. Li, and J. Chu, "A Representation Method Based on Aspect Graph for 3D Objects," Journal of Image and Graphic, vol. 7(A), no.9, pp. 906-910, Sep.2002.

[3] Yuan Luo, Huimin Ma, and Fengting Li, "An Improved Algorithm of Viewpoint Space Partition in 3D Object Recognition Application," Proceedings of the Seventh IASTED International Conference on Signal and Image Processing (SIP2006), pp. 34-39, Aug. 2006.

[4] WuChenye, and MA Huimin, "Efficiently Pruning for Visual Events in 3D Object Recognition Application," Proceedings of the 14th National Conference on Image and Graphics (NCIG2008), pp. 259-264, May. 2008.

[5] R.D. Schiffenbauer, "A survey of aspect graphs," Technical Report TR-CIS-2001-01, Department of Computer and Information Science, Polytechnic University, Brooklyn, Long Island, New York, Feb. 2001.

[6] WuChenye, and MA Huimin, "Computing EEE Events Based on Conicoid Theory," Tsinghua Science and Tecnology, unpublished

[7] O. Faugeras, J. Mundy, N. Ahuja, C. Dyer, A. Pentland, R. Jain, K. Ikeuchi, and Bowyer, K. "Panel Theme: Why Aspect Graphs are not (yet) Practical for Computer Vision," Proc. IEEE Workshop Directions in Automated CAD-Based Vsion, pp. 97-104, Jun. 1991, doi: 10.1109/CADVIS.1991.148762.

[8] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser., "The Princeton Shape Benchmark," Proc. Shape Modeling Applications, pp. 167-178, Jun. 2004, doi: 10.1109/SMI.2004.1314504