17^{th} *Computer Vision Winter Workshop*
Matej Kristan, Rok Mandeljc, Luka Čehovin (eds.)
Mala Nedelja, Slovenia, February 1-3, 2012

Using Aspect Graphs for View Synthesis

Geetha Ramachandran[1] and Walter Kropatsch[2]

[1]Institute of Telecommunications
Vienna University of Technology
Gusshausstrasse 25/389, 1040 Vienna Austria
gramacha@nt.tuwien.ac.at
[2]Pattern Recognition and Image Processing Group
Vienna University of Technology
Favoritenstrasse 9/186-3, 1040 Vienna Austria
krw@prip.tuwien.ac.at

Abstract. *A simple method for the generation of aspect graphs from multiple views of a polyhedron is presented in this paper. Re-creation of these multiple views given an aspect graph is also discussed. This method is the first step in the use of aspect graphs to create and transmit a 3D scene captured from multiple viewpoints. The proposed method has potential applications in multiview autostereoscopic displays and free-viewpoint videos.*

1. Introduction

Technology in visual communication is moving fast in the direction of adding a third dimension to its every aspect. The goal of such a move is to provide viewers with a real world experience. The lines between the physical and virtual worlds are becoming blurred and 3D media is progressing towards creating a seamless immersive environment. This has far-reaching implications in the world of entertainment, education and medicine among other applications.

The last few years have seen a huge spike in the creation of 3D cinema as well as the market availability of 3D-capable displays for commercial purposes. One of the key factors in the evolution of 3D media is the effective display of 3D content. 3D displays can be broadly classified into being stereoscopic or autostereoscopic. Autostereoscopic displays are different from stereoscopic displays in that the viewer is not required to wear special glasses or other user-mounted devices to perceive a 3D view. Autostereoscopic displays can be further classified into being two-view, multiview with fixed viewing zones or head/pupil tracked and super multiview [14]. Two-view displays transmit a single stream of a pair of left and right views to the user. These are ideal for use in mobile applications or single user gaming systems. Multiview displays with fixed viewing zones present multiple and different stereo pairs to the viewers to provide the appearance of the desired motion parallax. The viewer's left and right eyes receive different perspectives from adjacent regions. Although the number of views presented is too small for a continuous motion parallax, many techniques exist to minimize obvious and large transitions between the views. Super multiview displays show multiple discrete images and allow directional rays to pass through each point in the 3D space into the viewer's eyes so that a parallax is obtained. However, these require complex hardware and software and the display panels need to have a high native resolution which is not available currently. As opposed to the other options for autostereoscopic displays, multiview displays with fixed viewing points have the capabilities to be used in commercial environments such as cinema or as table-top 3D displays.

Moving along the 3D video processing chain, content creation and transmission is another area that needs to be addressed. One of the blocks in the mainstream use of 3D displays is the lack of readily available 3D content. This becomes more of a challenge in case of multiview displays as multiple views of the same scene are required. The capture of these views presents a time and infrastructure bottleneck. Also, transmission of these streams of data simultaneously requires a large bandwidth. Hence, there is a need for the generation of multiple views on the fly.

Visual content comprises of images that are generated as a result of the movement of an observer through the environment generating a stream of data in his/her visual field. Sampling of this image data gives rise to a scene-by-scene view of the environment. Generation of content for 3D media requires accurate and efficient modelling and re-creation of 3D scenes. The concept of aspect graphs presents a topological approach to 3D content creation. In this approach, the environment is perceived to consist primarily of solid bodies bounded by smooth surfaces [5]. The changes in the aspect of a scene are identified by visual events. The aspects and the visual events of the scene provide enough information to model the scene from any given viewpoint and to trace the changes from one view to the next. Aspect graphs are described in more detail in the next section.

Through this work, we aim to create a hierarchical aspect graph that encompasses information about a given 3D scene when viewed from multiple view points. An aspect graph of this nature will provide the capability to generate views dynamically. In this paper, we look at the creation of aspect graphs from of multiple views of a scene containing a single object and explore the possibility to re-create the scene using the information obtained through aspect graphs. The paper is organized as follows: a review of the state of the art in view synthesis and aspect graphs is provided in Section 2. Our methodology and results are presented in Section 3 and Section 4 describes our conclusions and the direction we plan to take this work forward.

2. Related Work

View synthesis for multiview systems presents a challenging and multi-faceted topic of research because of the widespread applications that can potentially use it. View synthesis involves the generation of new streams of data at new viewpoints from given streams. These given streams may be single left and right view pairs from stereo cameras or images from multiple cameras around the scene. In [8], a method is described for view synthesis from a stereo pair. The intermediate view is generated from a rectified stereo pair by creating multiple layers of the image based on disparity. These multiple layers are then merged together to create a single new image. In [12], the focus is to deal with depth discontinuities occurring in the stereo images. They are handled by separating the foreground and background boundary layers. These are then fused with a reliable layer to create the final image.

In [10], view synthesis using multiple cameras is explored. A volumetric model is created here from multiple input views to provide correspondences between the different components of the scene and these correspondences are then used to interpolate between two views to provide a new intermediate view. A Virtual Video Camera has been proposed to generate free-viewpoint video from unsynchronized, uncalibrated video streams in [7]. Here, dense pixel correspondences are used for image interpolation. A topological approach based on aspect graphs for multiple view synthesis of 3D scenes has not been explored before.

Aspect graphs were introduced to the field of object recognition in [5]. The underlying concept of aspect graphs is that human visual perception enables us to decide on the solid shape of an object from a few lines laid out on a plane surface. Infinite images can be captured from a scene through variation of the vantage point in the viewing space. These infinite images are grouped into a finite number of regions which are referred to as characteristic view domains [15]. An equivalency class is defined so that all the views belonging to one characteristic view domain are different from those belonging to another. Thus, a characteristic view domain can be represented by any one view in its domain and this is referred to as its aspect. An aspect is defined as a set of viewpoints of the object from which a set of 'singularities' are visible. Singularities depict the functional relation between the surface of objects and the visual field. These singularities are categorized into being either folds, cusps or T-junctions exhibited by the two-dimensional projection of the object on the image plane. Transition from one characteristic view domain to the next is marked by changes in topology of the scene. These changes are referred to as visual events. A graph depicting all the possible aspects of an object as nodes is referred to as its aspect graph. The edges connecting the nodes of the graph represent the visual events that result in the transition from one aspect to the next.

Since the introduction of aspect graphs, research has been performed into how they can be implemented for 3D object recognition. In [13], a theoretical perspective on the computing of aspect graphs for three-dimensional scenes is presented. If the vertex-edge pairs and triple-edge junctions of a scene can be identified, boundary conditions of the visual event regions are described here. The aspect regions are defined as the non-overlapping convex regions, that the scene is decomposed into by the visual event regions. In [2], a pragmatic discussion is presented on the utility of aspect graphs in computer vision. Some of the key issues highlighted here are as follows. Identifying the objects in a scene depends on various factors such as illumination and the sampling factor. Occlusions need to be dealt with in case of multiple objects in the scene. These vary based on the viewpoint chosen. The number of nodes and levels in the aspect graph increases and the structure of the aspect graph becomes more complex in direct relation to the structure of the object under consideration. The usage of aspect graphs as a method for object recognition presents a challenging indexing problem as this involves the search of a good match for one view of an unknown object among multiple aspects of multiple objects. In an attempt to solve some of these problems, aspect graphs have been defined for polyhedra [11], curved objects [6] and solids of revolution [1]. However, it has been seen that creating aspect graphs for general classes of objects is not a trivial task.

To deal with the complexity of aspect graphs of scenes, one of the solutions suggested in [2] is to look at the scene from a component perspective. This is the approach we adapt here to the deal with multiview synthesis of an object. Assuming we have multiple views of a scene captured from different angles, generation of the aspect graphs of components of the scene can be used as an approach to reduce the amount of data being transferred. In this paper, we present an algorithm to generate the aspect graph of a single object captured from various angles. We demonstrate our results using an image set obtained from the Amsterdam Library of Object Images [3]. Here, we use the '16' dataset which consists of a set of views of an opaque polyhedron with images captured by rotating the object at $5°$ resolution.

(a) (b)

Figure 1. '16' dataset from Amsterdam Library of Object Images. (a) View of the object at $0°$ (b) View of the object at $45°$

3. Proposed Method

The proposed method is described in detail below. As the first step, the vertices, edges and faces of the given views are obtained (Section 3.1). The aspect graph is generated in the next step, using the information obtained in the first step (Section 3.2). The next section (Section 3.3) investigates the possibility of how an aspect graph can be used to generate a view at any given viewpoint.

3.1. Determining the vertices and edges

Given multiple views of an object (V_1, V_2,...,V_N), as shown in Figure 1, the first step is to obtain the line drawing of the object to identify the key vertices and edges. As discussed in Section 2, obtaining an accurate line drawing of an object from a real-world scene is not trivial owing to various factors such as the illumination conditions, noise in the image and the sampling factor used in creating the image. Multiple approaches have been taken to arrive at the line drawing for a given scene based on the a priori knowledge of the scene.

Here, we propose two alternative methods. Our first approach assumes that we have a priori knowledge of the geometrical structure of the object in the scene, namely that it is a convex polyhedron. Given this knowledge, we propose a method for automatically detecting vertices of the object. Our first step is to use the morphological 'opening' operation to ensure that the object in the scene is well-connected. A new image is obtained using the operation

$$V_n = ([V_n \circ se1] \circ se2), \quad (1)$$

where V_n represents a view from the set (V_1, V_2,...,V_N) and $\circ$ indicates the opening operation and the structural elements se1 and se2 are flat linear structural elements lying along the x and y axes respectively. For the creation of a line drawing, it is easier if elements of the scene such as illumination, texture and noise are removed. Using Otsu's thresholding method [9], from the zeroth and first order cumulative moments of the gray-level histogram, a threshold is chosen for the scene and a binary image is created. Here the points on the object are indicated with the value one and the points in the background have value zero.

In order to determine the vertices of the image, the minima and maxima of the points on the x and y axes are determined. These peaks and troughs give approximate values for four of the vertices. This is shown in Figure 2(a).

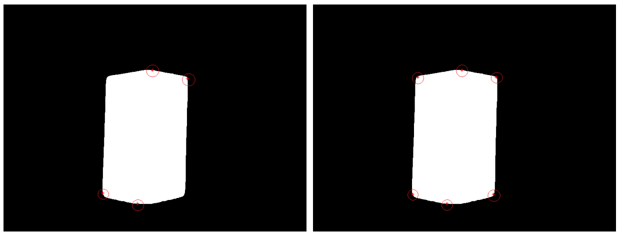
To identify the remaining vertices, the boundaries of the binary object are traced using the Moore-Neighbor tracing algorithm modified by Jacob's stopping criteria as presented in [4]. A window $W_{x,y}$ of size 5×5 is selected around each point on the boundary. A point is marked as a vertex subject to the condition

$$\Sigma W_{x,y} < \frac{N^2}{2} + 1, \quad (2)$$

where N indicates the size of the window around the point. Here, a summation of the binary values in the window is calculated. The condition shown in (2) is satisfied in the event that the point at the centre of the window has fewer boundary pixels than its neighbours, thereby indicating a peak or a trough and hence a potential vertex. The vertices obtained above (indicated in Figure 2(a)) are adjusted accordingly and the newly identified vertices are marked. Figure 2(b) illustrates the results.

As can be seen in Figure 2(b), the corners of the binary image provide information only about six vertices. From a priori knowledge, we know that the object under consideration is a polyhedron with eight vertices. From the elevation and viewpoint under consideration, only six or seven vertices are visible depending on the viewpoint chosen. Since the topological structure of the object is already known, we take advantage of the parallel edges and the other vertices are identified and marked.

An alternative to this method of vertex detection is to use a manual method of marking vertices. This is, however, subject to human error. However, the manual marking of vertices is advantageous under two conditions. First, in cases where the vertices cannot be clearly identified owing to illumination conditions or noise in the image, the manual method provides better results. Secondly, in views where the vertices cannot be marked owing to overlap of edges, this proves to be a better method. For instance, for the view of the object at 0°, only four vertices can be identified using morphological methods since the other two visible vertices are overlapped by edges. In this case, knowledge of the topology does not provide information about the potential positions of the other vertices.



(a) (b)

Figure 2. Results from Section 3.1. for the view at 45°. The vertices are encircled. (a) Initial vertex identification (b) Final vertices identified

Once all the vertices are identified, the edges are marked by joining the vertices and the faces are subsequently marked by joining the edges. The varying aspects of the objects can then be distinguished.

3.2. Creating the aspect graph

The creation of the aspect graph involves taking into consideration the infinite views of an object and grouping them into finite classes and connecting these classes by identifying the visual events leading to the transition from one aspect to the next. The aspect graph can contain multiple levels depending on the details taken into consideration. The dataset we are considering here contains images of an object obtained by revolving around one axis of view. Hence, the elevation of the object remains the same throughout the dataset. From a purely topological perspective, thus, the dataset under consideration has two main classes of views. Figure 3 shows these classes as seen by sweeping through the scene.

As shown in Figure 3, at the top level, there are two aspects to this dataset. The first aspect contains six identifiable vertices, while the second contains seven vertices. As the elevation remains the same, the transition from one aspect to the next is marked by the change in the number of vertices identified. Hence, the variation in the number of vertices is identified to be the top level of visual events. At the next level, the different vertices and edges constituting the faces seen in the aspect are taken into consideration.

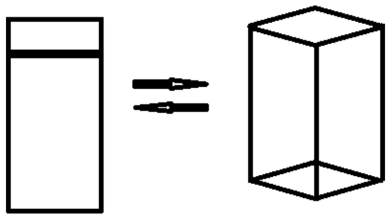


Figure 3. High level aspects obtained by sweeping through views from 0° to 360°

Figure 4 shows a more detailed aspect graph of the object. Here, the visibility and occlusion of specific vertices gives rise to the visual events. As we move from one aspect to another, from 0° to 360°, the faces that can be viewed change.

From the previous step, the vertices $(A - H)$ have been identified. The top face $EFGH$ is always visible as the camera moves around a horizontal axis around the scene. From each view, using the first step, the vertices and edges forming the faces are obtained. The aspects are created by aggregating the faces as shown below.

$$\begin{aligned} A_1 &= EFGH \cup ABFE \\ A_2 &= EFGH \cup ABFE \cup DAEH \\ A_3 &= EFGH \cup DAEH \\ A_4 &= EFGH \cup DAEH \cup CDHG \\ A_5 &= EFGH \cup CDHG \\ A_6 &= EFGH \cup CDHG \cup BCGF \\ A_7 &= EFGH \cup BCGF \\ A_8 &= EFGH \cup BCGF \cup ABFE \end{aligned} \tag{3}$$

In order to generate views using aspect graphs, a more detailed description of the object and the scene is required. In this paper, we only consider intensity information. However, to reproduce views for 3D video, information about the texture, illumination conditions and occlusions also need to be incorporated into the aspect graph. Hence, at the next level of detail, each aspect will contain this data as well. The advantage of such a multi-level approach lies in the fact that this becomes a scalable solution to view synthesis. Depending on the application at hand, the amount of detail and levels incorporated into the aspect graph can be varied.

3.3. Regenerating the views from the aspect graph

Given the aspect graph of the object and the vantage point of the observer, it is possible to create a new view. As mentioned above, the aspect graph used here provides only topological information about the object in the scene. It is possible to recreate a model view of the object. The intensity values are obtained from the first step. Given a viewpoint angle, the top level of the aspect graph (shown in Figure 3) is read. This gives information about how many vertices, edges and faces are visible in the view. Thus a preliminary view of the object is obtained. At the next level, depending on the angle provided, the specific vertices, edges and faces that are visible are chosen. Figure 5 illustrates the model obtained using this information.

As the amount of information carried in the aspect graph expands to include features such as texture and illumination, consequently re-creation of the object can be made more realistic. This can be extended to non-convex objects by taking into consideration occlusions and the visibility of the vertices from the vantage point.

4. Conclusion

The paper presents a simple method to create an aspect graph given multiple views of a 3D scene. Also, the possibility to re-create the views given the aspect graph is explored. Our results show that the application of aspect graphs for the generation of multiple views of an object is the first step to provide a scalable solution for the view synthesis of 3D scenes.

As a next step, the texture and illumination information needs to be captured and coded along with the intensity values and the aspect graph. This leads to a drastic improvement in the generated scene and enhances the objective quality measures. Also, investigation needs to be done into the creation of aspect graphs when multiple objects occur in the scene. Here, the challenges lie in capturing the topology of the objects given that there will be shadows and occlusions coming into the picture. Using this technique, aspect graphs can be adapted for coding streams of data where the same scene is captured from multiple viewpoints. With this, the efficient transmission of multiple streams of video data to multiview autostereoscopic displays will become a possibility in the future.

References

[1] D. Eggert and K. Bowyer. Computing the Perspective Projection Aspect Graph of Solids of Revolution. *IEEE Transactions on Pattern Analysis and Mechanics Intelligence*, 15:109–128, February 1993. 3

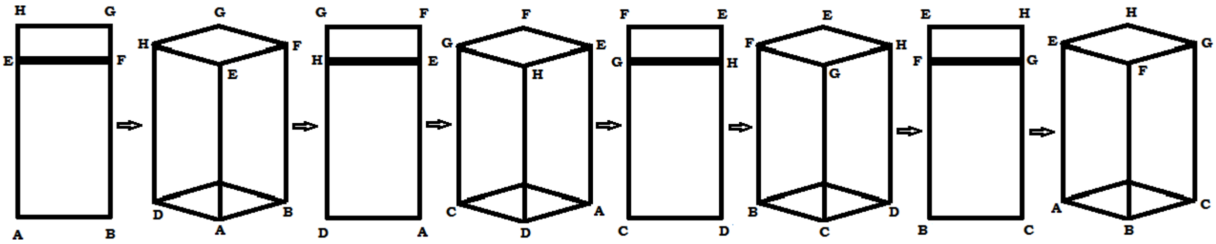


Figure 4. Detailed topological aspects obtained by sweeping through views from 0° to 360°

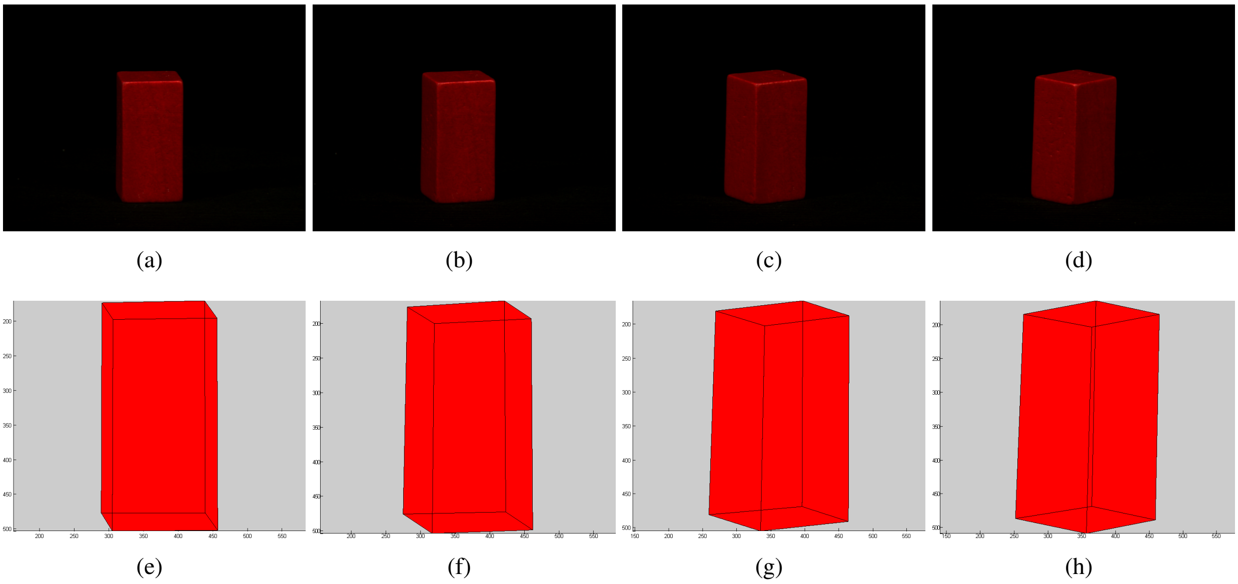


Figure 5. Models generated for the object at various viewpoints. (a) - (d) Object viewed at $5°, 15°, 30°$ and $45°$ (e) - (h) Models generated for the views at $5°, 15°, 30°$ and $45°$

[2] O. Faugeras, J. Mundy, N. Ahuja, C. Dyer, A. Pentland, R. Jain, K. Ikeuchi, and K. Bowyer. Why Aspect Graphs Are Not (Yet) Practical for Computer Vision. 3

[3] J. M. Geusebroek, G. J. Burghouts, and A. W. M. Smeulders. The Amsterdam Library of Object Images. *International Journal of Computer Vision*, 61(1):103–112, 2005. 3

[4] R. Gonzalez, R. Woods, and S. Eddins. *Digital Image Processing Using MATLAB*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2003. 4

[5] J. J. Koenderink and A. J. van Doorn. The Singularities of the Visual Mapping. *Biological Cybernetics*, 24:51–59, 1976. 2

[6] D. Kriegman and J. Ponce. Computing Exact Aspect Graphs of Curved Objects: Solid of Revolution. *International Journal of Computer Vision*, 5:119–135, November 1990. 3

[7] C. Lipski, F. Klose, K. Ruhl, and M. Magnor. The Virtual Video Camera: Simplified 3DTV Acquisition and Processing. In *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4, Antalya, Turkey, May 2011. 2

[8] N. Manap and J. Soraghan. Novel View Synthesis based on Depth Map Layers Representation. In *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4, Antalya, Turkey, May 2011. 2

[9] N. Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, January 1979. 4

[10] H. Saito, S. Baba, M. Kimura, S. Vedula, and T. Kanade. Appearance-based Virtual View Generation of Temporally-varying Events from Multi-camera Images in the 3D Room. In *Proceedings of the Second International Conference on 3-D Digital Imaging and Modeling*, pages 516–525, Ottawa, Ont., Canada, 1999. 2

[11] I. Shimshoni and P. J. Finite-Resolution Aspect Graphs of Polyhedral Objects. *IEEE Transactions on Pattern Analysis and Mechanics Intelligence*, 19:315–327, 1993. 3

[12] A. Smolic, K. Muller, K. Dix, P. Merkle, P. Kauff, and T. Wiegand. Intermediate View Interpolation based on Multiview Video plus Depth for Advanced 3D Video Systems. In *15th IEEE International Conference on Image Processing (ICIP)*, pages 2448–2451, San Diego, CA, USA, October 2008. 2

[13] E. Sojka. Aspect Graphs of Three-Dimensional Scenes. *Proceedings of Winter School of Computer Graphics*, page 100, February 1995. 3

[14] H. Urey, K. Chellappan, E. Erden, and P. Surman. State of the Art in Stereoscopic and Autostereoscopic Displays. *Proceedings of the IEEE*, 99(4):540–555, April 2011. 1

[15] R. Wang and H. Freeman. Object Recognition based on Characteristic View Classes. In *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, volume 1, pages 8–12, June 1990. 2