Aspect Graphs of Three-dimensional Scenes

Eduard Sojka

Department of Computer Science, Technical University of Ostrava,
tr.17.listopadu,708 33 Ostrava-Poruba, Czech Republic
e-mail: eduard.sojka@vsb.cz

Abstract

The paper presents a new approach to the theory of aspect graphs. This approach is based on the presumption that an equivalence relation can be introduced on the set of images of a given scene. The equivalence relation induces the decomposition of the set of images, which is infinite, into the finite number of classes. Similarly, also the space surrounding the scene is decomposed into the classes. From every point of certain class, equivalent images of the scene are perceived. The aspect graph is the graph that contains the information about these classes and the relations between them. Presented theory enables to put the approaches published earlier into the unifying framework. The paper also presents an example in which the general theory of aspect graphs is applied to the three-dimensional problem. Also an algorithm for computing the decomposition of space into the classes is proposed.

Keywords. Aspect graph, visual potential graph, visibility.

1 Introduction

The idea of aspect graphs is a relatively new one. From one point of view, it develops a traditional problem of computer graphics, the problem of visibility. On the other hand, it brings some completely new concepts. It seems, for example, that the aspect graph can be used for representing objects in three-dimensions. The idea is to represent object by a number of different two-dimensional views from different viewpoints. Views are organised in an aspect graph. Thus, the aspect graph can be used as an alternative or supplementary model of a scene, which may be useful when solving various problems such as, for example, the problem of visibility, problem of navigation in a scene, and the problem of three-dimensional object recognition.

The idea of aspect graphs is based on work done by Koenderink and van Doorn (Koenderink, 1975, 76, 79). They studied the optical field that occurs when observing a plane. They also introduced a graph structure which they called the visual potential. Each node in this graph structure represents a different view of the object. A number of algorithms for computing different versions of the aspect graph of practical scenes have been described since then. Some of them are based on the use of the orthographic projection (Gigus, 1991), (Plantinga, 1986), others use the perspective projection (Stewman 1991), (Eggert 1993). Many algorithms deal only with certain classes of objects (Stewman 1991),(Maripuri, 1990). In this paper we present a general theory of aspect graphs, which enables to put the approaches published earlier into the unifying framework.

2 General Theory of Aspect Graphs

Let us now consider a scene $\mathcal{S}$ which is a subset of the three-dimensional Euclidean space $\mathbf{E}^3$. The scene may consist of an arbitrary number of mutually disconnected objects. Mathematically, $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots \cup \mathcal{S}_n$. All the objects of the scene are opaque. Points that do not belong to the objects form the exterior $\mathcal{X}$. Obviously, $\mathcal{X}=\neg\mathcal{S}$. Let us observe the scene $\mathcal{S}$ from any particular point $p\in\mathcal{X}$ of the exterior. Thus we obtain the image I_p, which is a subset of the two-dimensional Euclidean space $\mathbf{E}^2$. Now, if the viewpoint p moves through all the possible positions in the exterior, we obtain a set of images $\mathcal{I} = \{I_p \mid p\in\mathcal{X}\}$. For every two distinct viewpoints p, q, the corresponding images I_p, I_q generally differ. Therefore, $\mathcal{I}$ is an infinite set. We suppose that the mapping π: $\mathcal{X}\rightarrow\mathcal{I}$, which maps every point $p\in\mathcal{X}$ of the exterior onto the corresponding image $I_p\in\mathcal{I}$, is given by a central projection of all those points of the scene that are visible from a chosen viewpoint p. The viewpoint p is the centre of the projection.

In many applications, however, it is not necessary nor feasible to consider the image I_p as the collection of all its points. Instead, it is often appropriate to extract only some set of relevant information. We use the term aspect for this set of information, and we use A to denote this set. The set of all possible aspects of the scene is $\mathcal{A} = \{A_1,A_2,\ldots,A_n\}$. We suppose that $\mathcal{A}$ is a finite set.

Let us suppose that the mapping $\alpha:\mathcal{I}\rightarrow\mathcal{A}$ of the set of images onto the set of aspects is known. Thus, the aspect A_i corresponding to a given image I_p can be determined. We let $\mathrm{Ker}\alpha$ denote the kernel of the mapping α. $\mathrm{Ker}\alpha$ is a binary equivalence relation on the set $\mathcal{I}$. Two members of the set $\mathcal{I}$ are connected by this relation if and only if they have the same destination in the set $\mathcal{A}$, that is, $I_p \,\mathrm{Ker}\alpha\, I_q \Leftrightarrow \alpha(I_p)=\alpha(I_q)$. The kernel $\mathrm{Ker}\alpha$ of the mapping α induces the decomposition of the set $\mathcal{I}$ of images into the classes. All the images corresponding to the same aspect are in the same class. We denote $\mathcal{I}/_{\mathrm{Ker}\alpha}$ the set of classes of this decomposition.

Let σ be the composite mapping $\sigma=\pi\alpha$, $\sigma:\mathcal{X}\rightarrow\mathcal{A}$. Then similarly $\mathrm{Ker}\sigma$ induces the decomposition of the exterior $\mathcal{X}$ into the classes. A certain class X_i of this decomposition contains all the viewpoints giving the images corresponding to the same aspect A_i, that is, $X_i = \{ x\in\mathcal{X} \mid \alpha(I_x) = A_i \}$. We use the term aspect region for a class of this decomposition, and denote $\mathcal{X}/_{\mathrm{Ker}\sigma}$ the set of all these classes. It is easy to see that $\mathcal{X} = X_1 \cup X_2 \cup \ldots \cup X_n$, and $X_i \cap X_j = \varnothing$ if $i\neq j$. Obviously, there is a mapping $\mathcal{I}/_{\mathrm{Ker}\alpha} \leftrightarrow \mathcal{X}/_{\mathrm{Ker}\sigma}$, which is one-to-one and onto.

Let us now consider a viewpoint $p\in\mathcal{X}$ and its open ball neighbourhood $B(p,r)=\{ x\in\mathbf{E}^3 \mid \mathrm{distance}(p,x) < r \}$, $r>0$. We suppose that this neighbourhood does not contain any points of the scene. Let us now construct the set $\mathcal{A}_{B(p,r)} = \{ \sigma(x) \mid x \in B(p,r) \}$ of aspects that belong to points of the neighbourhood B. The following cases can be distinguished:

i) For the viewpoint p, a radius r>0 can be found such that $\mathcal{A}_{B(p,r)} = \{A_i\}$, i.e., $|\mathcal{A}_{B(p,r)}| = 1$. In this case p is an inner point of the region X_i.

ii) For the viewpoint p and any radius r>0, the inequality $|\mathcal{A}_{B(p,r)}|>1$ holds. In this case p lies on the boundary between the aspect regions. Specifically, if for $r \to 0$ $\mathcal{A}_{B(p,r)} = \{A_i, A_j\}$, i.e., $|\mathcal{A}_{B(p,r)}| = 2$, then p lies inside the face that forms the boundary between the regions X_i, X_j of the aspects A_i, A_j. The regions X_i, X_j and aspects A_i, A_j are called adjacent regions and adjacent aspects, respectively. We use the notation X_i Adj X_j, A_i Adj A_j to express this relationship.

Now we assume that the viewpoint p occupies various positions in the exterior, for example, it moves along some curve. In the moment when the viewpoint moves across the boundary from one aspect region to another, a sudden change occurs in the image. We use the term visual event for this change.

Let $A_i, A_j \in \mathcal{A}$ be two adjacent aspects. We let $\mathcal{N}$ denote the set of all ordered pairs of adjacent aspects of a given scene. That is, $\mathcal{N} = \{ (A_i, A_j) \mid A_i, A_j \in \mathcal{A} \wedge A_i \text{ Adj } A_j \}$. Obviously, every element (A_i, A_j) of the set $\mathcal{N}$ represents the situation when the viewpoint moves from the region X_i to the region X_j through the boundary between them. The move yields the change of perceived picture. This event corresponds to the change of the aspect from A_i to A_j. Given a pair (A_i, A_j) of adjacent aspects, we assume that the difference between these aspects can be determined. More formally, a mapping $\delta: \mathcal{N} \to \mathcal{D}$ is considered. Every element D of the set $\mathcal{D}$ represents the difference of some pair of adjacent aspects. As D represents some visual event, $\mathcal{D}$ then is the set of all these possible events. The kernel Kerδ of the mapping δ induces the decomposition on the set $\mathcal{N}$. In one class of this decomposition, there are all the pairs (A_i, A_j) of adjacent aspects for which the move of the viewpoint from the region X_i to X_j yields the same visual event. $\mathcal{N}/_{\mathrm{Ker}\delta}$ denotes the set of classes of this decomposition.

Let us use the term D-visual event for the event that is represented by a certain element D of the set $\mathcal{D}$. All the points of the exterior from where the D-visual event is perceived form the region of this visual event. We use U_D to denote this region . Let X_i and X_j be the regions of aspects A_i and A_j, and let $cl(X_i)$ and $cl(X_j)$ denote the closures of the regions X_i and X_j, respectively. The region of D-visual event then can be described as follows:

$$U_D = \{ x \in E^3 \mid \exists (A_i, A_j) \in \mathcal{N} : \delta(A_i, A_j) = D \wedge x \in cl(X_i) \wedge x \in cl(X_j) \}. \quad (1)$$

Every point of the boundary of the closure of any bounded aspect region lies in the region of some visual event or on the boundary of some object of the scene. Conversely, every point of any visual event region belongs to the closures of at least two aspect regions and cannot lie inside any aspect region. The regions of all possible visual events partition the exterior into the finite number of convex regions, which are aspect regions.

Consider a scene $\mathcal{S}$, exterior $\mathcal{X}$, set $\mathcal{I}$ of images, set $\mathcal{A}$ of aspects, and the mappings α, π, $\sigma=\pi\alpha$ as was explained earlier. We know that the kernels Kerα, Ker σ of the mappings α, σ induce the decomposition of the set of images, and the decomposition of the exterior into the finite number of classes. We also know that there is a one-to-one mapping $\mathcal{I}/_{\mathrm{Ker}\alpha} \leftrightarrow \mathcal{X}/_{\mathrm{Ker}\sigma}$. Let us now construct the directed graph $\mathcal{P}=(\mathcal{V},\mathcal{W})$ that corresponds to the decomposition of the set of images and thus also to the decomposition of the exterior. In this graph, $\mathcal{V}$ is the set of nodes and $\mathcal{W}\subseteq \mathcal{V}\mathrm{x}\mathcal{V}$ is the set of edges. The node $v_i\in\mathcal{V}$ corresponds just to one aspect A_i (thus it also corresponds just to one element of the set $\mathcal{I}/_{\mathrm{Ker}\alpha}$, and just to one element of the set $\mathcal{X}/_{\mathrm{Ker}\sigma}$). There is an edge v_iv_j in the graph $\mathcal{P}$ if and only if A_i and A_j are adjacent aspects, i.e., $(A_i, A_j)\in\mathcal{N}$. In the graph $\mathcal{P}$ we now introduce labels of nodes and labels of edges. We do it as follows: we use the pair (A_i, X_i) to label the node v_i, and the pair (U_{ij}, D) to label the edge v_iv_j. In the second pair, U_{ij} denotes the intersection $\mathrm{cl}(X_i) \cap \mathrm{cl}(X_j)$ (note that $U_{ij} = U_{ji}$), and $D\in\mathcal{D}$ is the difference $D=\delta(A_i,A_j)$ of corresponding aspects. For the graph $\mathcal{P}$, we will use the term aspect graph.

3 Aspect Graph for a Three-dimensional Scene

Now we will present an example of application of the general theory stated in the previous section. We will focus on three-dimensional scenes. First we will choose the form of aspect, which also determines the mapping α. Then we will study how the visual events manifest themselves, and what are the properties of the space decomposition. We will give special attention to determining visual event regions. Finally, we will state an algorithm for computing visual event regions.

Let us consider a scene in the three-dimensional Euclidean space $\mathbf{E}^3$. The scene contains objects bounded by planar faces. We suppose that we know the boundaries of all the objects of the scene. These boundaries consist of faces, edges and vertices. Let V be a set of symbols such that there is a one-to-one mapping between the set V and the set of all vertices of the scene. Similarly, let E be the set of symbols such that there is a one-to-one mapping between this set and the set of all edges of the scene.

3.1 Aspect and Aspect Region

In our example we let the aspect A be a subset of $(V\cup(E\mathrm{x}E))$, that is, $A\subset(V\cup(E\mathrm{x}E))$. One can easily see that any aspect contains two types of elements: symbols from the set V, and pairs of symbols from the set E. An aspect contains the symbol $v_i\in V$ if and only if the picture perceived from a given viewpoint contains the image of the vertex that corresponds to the symbol v_i. Similarly, an aspect contains the pair (e_i,e_j) $e_i,e_j\in E$ if and only if the picture perceived from a given viewpoint contains the intersection of the images of the edges that correspond to the symbols e_i, e_j. Obviously, the pairs (e_i, e_j) are unordered. Furthermore, we exclude the situation when $i=j$. Note that this definition of aspect seems to correspond with the idea that edges and vertices play the most significant role when decoding images by both human and artificial vision systems.

Every visual event can be described by four types of basic actions. $Ins(v_i)$ and $Del(v_i)$ denote the actions when the aspect is modified by adding and deleting the symbol v_i, respectively. This corresponds to the situation when the image of some vertex appears or disappears. Similarly, $Ins(e_i, e_j)$ and $Del(e_i, e_j)$ denote the actions when the aspect is modified by adding and deleting the pair (e_i, e_j) of symbols. This corresponds to the situation when the intersection of the images of the edges e_i, e_j appears or disappears in the picture. Let Q be the set

$$Q = \{ Ins(v_i) \mid v_i \in V \} \cup \{ Del(v_i) \mid v_i \in V \} \cup \{ Ins(e_i,e_j) \mid e_i,e_j \in E \} \cup \{ Del(e_i,e_j) \mid e_i,e_j \in E \} \quad (2)$$

and let 2^Q denote the set of all subsets of Q. Then for any visual event must be

$$D \subseteq Q. \quad (3)$$

The set of visual events then is $\mathcal{D} \subseteq 2^Q$. (4)

In principle, the algorithm presented later is based on determining the members D of $\mathcal{D}$ with subsequent computation of U_D. For that reason we focus on the issue what sorts of visual events $\mathcal{D}$ can consist of and what are the mechanisms that cause these events.

3.2 Visual Event

Let us first consider the event when the image of a certain vertex v_i appears or disappears. This event will be referred to as a v-visual event. Since all the objects of the scene are supposed to be bounded by planar faces, the only way in which a v-visual event may happen is that the image of the vertex v_i seems to appear or disappear behind the image of some edge e_k of the scene. Thus, a v-visual event arises from the interaction between some vertex v_i and some edge e_k. Let us now suppose for a while that the scene contains a separated vertex v_i as depicted in Figure 1a. In this case a v-visual event consists in transition of the image of the vertex v_i across the image of the edge e_k. Since the vertex v_i appears or disappears during this event, the difference D is $D = \{Ins(v_i)\}$ or $D = \{Del(v_i)\}$. The region U_{ve} of this visual event is depicted in Figure 1a. When observing our scene from any point of this region, the vertex v_i seems to lie on the edge e_k.

However, if we deal with real solids, then the mere $D=\{Ins(v_i)\}$ or $D=\{Del(v_i)\}$ do not describe admissible visual events. Since some edges always originate from every vertex, the appearance or disappearance of a vertex is always accompanied by the appearance or disappearance of intersections of some edges. Considering possible cases how the edge e_k and the edges originating from the vertex v_i can be situated, several cases of v-visual event can be distinguished. Figure 2 gives more detailed information about this problem. For each case of v-visual event, the figure shows a pair of pictures that are perceived by an observer before and after that particular case of v-visual event. For each case, it also provides the corresponding value of the difference D.

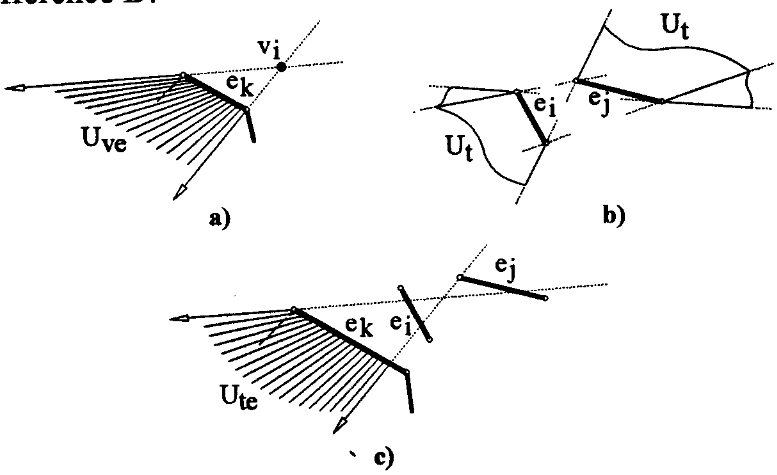

Figure 1. Two types of visual events can be distinguished in an image of a given scene. a) The v-visual event occurs when the vertex v_i seems to appear or disappear behind the edge e_k. This phenomenon can be perceived from the region U_{ve}. b) From the region U_t the edges e_i, e_j seem to intersect each other thus producing the illusion of a t-node. c) The t-visual event occurs when the t-node seems to appear or disappear behind the edge e_k. This phenomenon can be perceived from the region U_{te}.

Now we consider the event during which the intersection of two edges appears or disappears. First we suppose that the scene contains just two edges e_i,e_j. In this case all the viewpoints from where both edges seem to intersect each other form a region U_t, which is a subset of $\mathbf{E}^3$ (Figure 1b). The boundary of this subset forms the region of the events {Ins(e_i,e_j)} and {Del(e_i,e_j)}. This region consists of planar faces, each of which being determined by one edge and a vertex of the other edge. Not surprisingly, in this case the visual event region can be determined as a set of v-visual event regions.

In practice, however, we do not deal with scenes that contain only two edges. Moreover, every edge belongs to the boundary of some solid. In a picture of a real scene, intersection of the edges e_i, e_j is perceived as a t-node where the visibility of further edge changes. In the previous paragraph we have clarified one mechanism how the t-node e_i, e_j can occur. In real scenes, however, the situation is more complicated. In fact, the t-node need not be visible from every point of the region U_t, instead, it can be hidden behind other objects of the scene. Since all the objects of the scene are supposed to be bounded by planar faces, the t-node e_i, e_j must always seem to appear or disappear behind some edge e_k of the scene. Therefore, this event will be referred to as a t-visual event. The region U_{te} of the t-visual event is depicted in Figure 1c. When observing the scene from any point of this region, the edges e_i, e_j, e_k seem to intersect in one point. If we deal with real solids, then again the mere D={Ins(e_i,e_j)} or D={Del(e_i,e_j)} do not describe admissible visual events. For scenes containing real solids, the admissible values of the difference D are stated in Figure 2a,b.

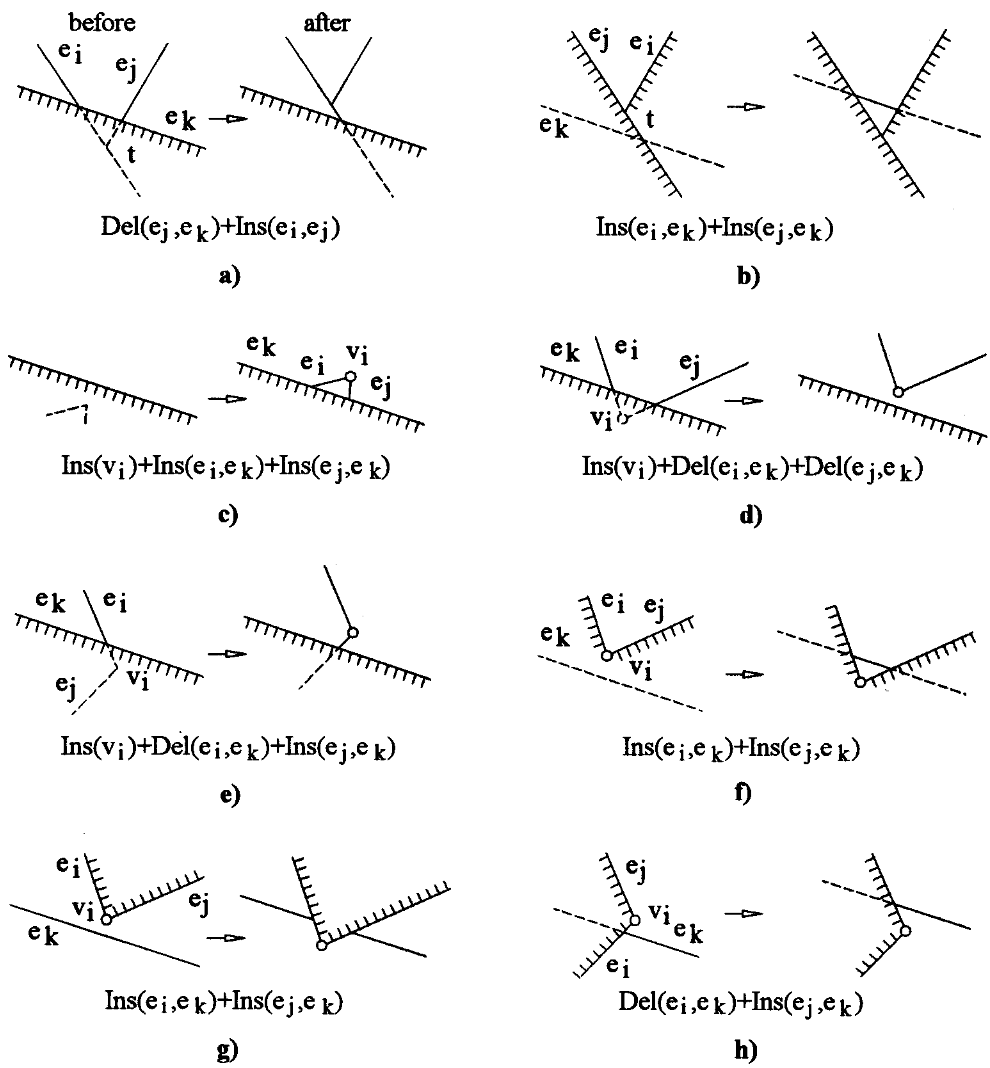

Figure 2. *In a picture of a real scene, the visual events manifest themselves in different ways. For t-visual event, following situations can be distinguished:* ***a)*** *the t-node appears (or disappears),* ***b)*** *the edge appears. Similarly for v-visual event:* ***c),d),e)*** *the vertex appears,* ***f)*** *the edge appears,* ***g)*** *splitting of an edge,* ***h)*** *transition of an edge.*

3.3 Visual Event Region

Now we will discuss the problem of determining visual event regions. Recall that a visual event region is defined as the set of points from where a certain visual event is perceived. Visual event region is formed by a surface in space. We will first examine the case when the visual event region does not contain any internal points of any solid of a given scene, that is, visual event region does not intersect any solid. We let U_{D0} denote the region determined using this assumption. Later on we will introduce additional correction to solve also more complicated cases.

For t-visual event, the region U_{D0} is defined by three different edges v_iv_j, v_kv_l, $v_mv_n \in E$. To obtain the most general case, we suppose that no two of these edges are collinear and no two intersect each other (Figure 3a). Since from all the points of the region U_{D0} the edges must seem to intersect in one point, U_{D0} consists of those lines of sight that simultaneously intersect all three edges. Thus, U_{D0} is a part of a ruled surface. The equation of this surface is given by

$$\mathbf{x}(u,w) = \mathbf{x}_m(u) + w\,\mathbf{r}(u) \tag{5}$$

where u,w are parameters, $\mathbf{x}_m(u)$ is a point on the edge $v_m v_n$, and $\mathbf{r}(u)$ is the direction of a line of sight (Figure 3a). Let $\mathbf{v}_m$ and $\mathbf{v}_n$ be the position vectors of the points v_m and v_n, respectively. Furthermore, let $\mathbf{q}_{mn}$ denote the difference $\mathbf{v}_n - \mathbf{v}_m$, and $\mathbf{x}_i(u)$ let be the intersection of the edge $v_i v_j$ with the line of sight. Then we have

$$\mathbf{x}_m(u) = \mathbf{v}_m + u\,\mathbf{q}_{mn}, \qquad \mathbf{q}_{mn} = \mathbf{v}_n - \mathbf{v}_m, \tag{6}$$
$$\mathbf{r}(u) = \mathbf{x}_m(u) - \mathbf{x}_i(u).$$

Finally, if we use $\mathbf{x}_k(u)$ to denote the intersection of the edge $v_k v_l$ with the line of sight, then the condition that the lines $x_i x_k$ and $x_m x_i$ are collinear can be expressed as

$$(\mathbf{x}_k - \mathbf{x}_i) = \lambda\,(\mathbf{x}_m - \mathbf{x}_i) \qquad \text{where } \lambda \in R. \tag{7}$$

By making use of (7) we obtain

$$t = \frac{(\mathbf{q}_{ij}\mathbf{q}_{ki}\mathbf{q}_{mi}) - u(\mathbf{q}_{ij}\mathbf{q}_{ki}\mathbf{q}_{mn})}{(\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{mi}) - u(\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{mn})}, \tag{8}$$

$$s = \frac{(\mathbf{q}_{ki}\mathbf{q}_{kl}\mathbf{q}_{mi}) - u(\mathbf{q}_{ki}\mathbf{q}_{kl}\mathbf{q}_{mn})}{(\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{ki}) - (\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{mi}) + u(\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{mn})}, \tag{9}$$

$$\mathbf{r}(u) = \mathbf{q}_{im} + u\mathbf{q}_{mn} - s(u)\mathbf{q}_{ij} \tag{10}$$

where t and s are parameters (see Figure 3a), $\mathbf{q}_{ij} = \mathbf{v}_j - \mathbf{v}_i$, and $(\mathbf{q}_{ij}\mathbf{q}_{kl}\mathbf{q}_{mn}) = \mathbf{q}_{ij}\cdot(\mathbf{q}_{kl} \times \mathbf{q}_{mn})$ (note that further similar expressions can be written when interchanging the indices). Substituting (5) and (9) in (4), we obtain

$$\mathbf{x}(u,w) = \mathbf{v}_m + u\mathbf{q}_{mn} + w\mathbf{q}_{im} + uw\mathbf{q}_{mn} - s(u)w\mathbf{q}_{ij} \tag{11}$$

In order to define the region U_{D0}, the parameters u, t, s, w must satisfy the following conditions

$$0 \le u \le 1, \qquad 0 \le t \le 1, \qquad 0 \le s \le 1, \qquad (w \ge 0) \vee (w \le -1). \tag{12}$$

It is worth pointing out that for real scenes, the condition for w produces in fact two different regions. One region for $w \ge 0$ and another for $w \le -1$. These regions differ in the difference D (see again Figure 2a,b).

The region U_{D0} of v-visual event can be considered as a special case of the region U_{D0} of t-visual event. To achieve this specialisation, we let the edges $v_i v_j$ and $v_k v_l$ intersect each other in a common vertex, denoted by v_i. All the lines of sight pass through that vertex, therefore, U_{D0} is a planar region (see Figure 3b). The equation of this region can be expressed by

$$x(u,w) = v_i + w\, q_{im} + u\, w\, q_{mn} \quad (13)$$

where $q_{im} = v_m - v_i$, $q_{mn} = v_n - v_m$.

In this case the conditions for parameters u,w are as follows:

$$0 \le u \le 1, \qquad (w \ge 1) \vee (w \le 0). \quad (14)$$

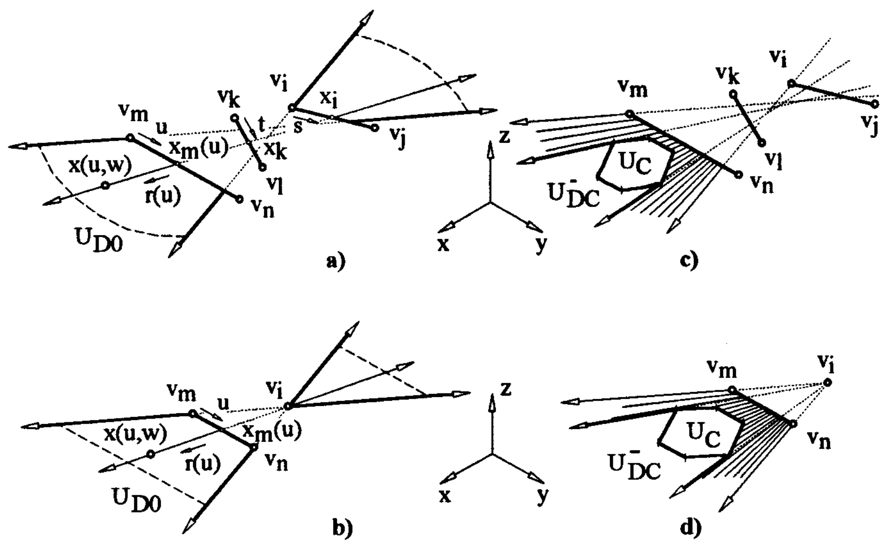

*Figure 3. Determining the aspect regions: **a)** The region U_{D0} for t-visual event, **b)** the same for v-visual event, **c)** the region U^-_{DC} for t-visual event, **d)** the same for v-visual event.*

Now we will consider the situation when the region U_{D0} of visual event can also contain internal points of some solids of the scene, which was excluded up to now. If we let U_{D0} cut the solids of the scene, then we obtain the cross-section region U_C. The region U_{D0} can now be decomposed into two subregions, which are denoted by U^+_{DC} and U^-_{DC}. If the trajectory of the viewpoint intersects the region U^+_{DC}, then the visual event is perceived. On the other hand, if the trajectory intersects the region U^-_{DC}, then the entities taking part in this particular visual event are hidden behind some solids of the scene. Since the visual event cannot be seen from the points of the region U^-_{DC}, U^-_{DC} must be excluded from the visual event region. Thus, the resulting visual event region U_D is given by

$$U_D = U^+_{DC} = U_{D0} - U^-_{DC}. \quad (15)$$

Figure 3c shows an example of the shape of the region U^-_{DC} for t-visual event. Similarly, Figure 3d shows the same for v-visual event.

The expression 15 represents a two-dimensional problem in the space u,w and suggests that the visual event region U_D can be found. In practice, however, the computation of visual event region can be organised in such a way that neither cross-section region nor the difference in the expression 15 need to be done explicitly. Instead, knowing the equation of the region U_{D0}, the segments of the boundary of visual event region can be determined directly. Subsequently, knowing these segments, the boundary of visual event region can be assembled. Note that the boundary contains certain parts of the boundaries of the regions U_{D0} and U_C, and certain parts of lines of sight that pass through the vertices of the cross-section region U_C.

3.4 Computing Aspect Regions

We now turn our attention to the problem of computing the aspect graph. The computation of aspect regions seems to play the key role in this problem. If we use the notation introduced in the previous sections, then the algorithm for computing the aspect regions of a given scene can be outlined as follows:

i) Generate systematically all the pairs vertex-edge and all the triplets of edges of a given scene. These pairs and triplets may cause v-visual events and t-visual events, respectively.

ii) For each such pair or triplet, determine the equation (expressions 11, 13) and the boundary (conditions 12, 14) of the region U_{D0}. Check whether U_{D0} intersects some solids of the scene. If that is the case, perform the correction expressed by the equation 15, which provides the resulting visual event region U_D.

iii) Determine the aspect regions as the non-overlapping convex regions that the exterior is decomposed into by visual event regions.

If the scene contains only one convex solid, then the algorithm can be considerably simplified. In this case t-visual events do not occur at all and v-visual events can only be produced by vertex-edge pairs in which both the vertex and the edge belong to one face of the solid. In all other cases $U_{D0} = U^{-}_{CD}$ and consequently $U_D = 0$. We also note that $U_C = 0$ and $U_D = U_{D0}$ if both the vertex and the edge belong to one face of a convex solid.

Another substantial simplification is naturally achieved for two-dimensional problems. Recall that in this case the scene is represented by a set of simple polygons. Note, too, that the only visual event that can occur in this case is the event when the vertex v_i of a polygon seems to appear or hide behind the vertex v_j. The region U_{D0}, which is a segment of the line v_i,v_j, begins at v_j, does not contain v_i, and continues to infinity. The algorithm for computing aspect regions must be therefore adapted to generate all the pairs v_i,v_j of the scene. If the line determined by the pair v_i,v_j does not intersect any polygon of the scene, then the visual event region U_D equals U_{D0}, otherwise an additional correction is needed. The correction does fully conform with the expression 15. In two-dimensions, however, it results in simple rules that examine the intersections of the line with the polygons representing the scene. These rules can be stated as follows: i) if there is an intersection that lies between the points v_i and v_j, then the visual event cannot occur at all, ii) if there are intersections within U_{D0}, then U_D begins at v_j and continues until it reaches the first of these intersections, and iii) in all other cases U_D equals to U_{D0}.

Note that some additional steps, rules and other details aimed at reducing the time complexity of the algorithm might be mentioned in this section (one idea is to recognise and exclude the pairs and triplets that cannot produce any visual event). Nevertheless, since this paper aims at more conceptual view, they are not discussed here.

4 Conclusion

In this paper, we have presented a new approach to the theory of aspect graphs. This approach is based on the presumption that an equivalence relation can be introduced on the set of images of a given scene. Presented theory enables to put the approaches published earlier into the unifying framework. We have presented an example of application of this general theory. In this example the theory has been applied to the three-dimensional problem and we have also presented some further theoretical results there. Namely, we have proposed certain form of aspect and studied how the visual events manifest themselves. We have also derived the equations of visual event regions, which partition exterior into the cells from which equivalent images of the scene are perceived. Finally, we have sketched an algorithm for computing these cells.

References

1. Eggert D. - Bowyer K.,1993: *Computing the Perspective Projection Aspect Graph of Solids of Revolution*, IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol.15, No 2, 109-128.
2. Gigus Z. - Canny J. - Seidel R.,1991: *Efficiently Computing and Representing Aspect Graphs of Polyhedral Objects*, IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol.13, No 6, 542-551.
3. Koenderink,J.J. - van Doorn A.J.,1975: *Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer*, Optica Acta, Vol.22, No 9, 773-791.
4. Koenderink J.J. - van Doorn A.J.,1976: *Local structure of movement parallax of the plane,* J.Opt.Soc.Am., Vol.66, No 7, 717-723.
5. Koenderink J.J. - van Doorn A.J.,1976: *The singularities of the visual mapping,* Biological Cybernetics, Vol.24, 51-59.
6. Koenderink J.J. - van Doorn A.J.,1979: *The Internal Representation of Solid Shape with Respect to Vision,* Biological Cybernetics, Vol.32, 211-216.
7. Maripuri S.R. - Zeid I.,1990: *Generating aspect graphs for nonconvex polyhedra,* Computer-Aided Design, Vol.22, No 5, 258-264.
8. Plantinga W.H. - Dyer C.R.,1986: An algorithm for constructing the aspect graph, Proc. IEEE 27th Symp. on Foundations of Comput. Sci, pp.123-131.
9. Stewman J.H. - Bowyer K.W.,1991: *Direct Construction of the Perspective Projection Aspect Graph of Convex Polyhedra,* Computer Vision, Graphics, and Image Processing, Vol.51, 20-37.