

Visual Hull from Imprecise Polyhedral Scene

Peng He

*Department of Computing
Imperial College London (ICL)
180 Queens Gate, London, SW7 2AZ, UK
Email: ph206@imperial.ac.uk*

Abbas Edalat

*Department of Computing
Imperial College London (ICL)
180 Queens Gate, London, SW7 2AZ, UK
Email: a.edalat@imperial.ac.uk*

Abstract—We present a framework to compute the visual hull of a polyhedral scene, in which the vertices of the polyhedra are given with some imprecision. Two kinds of visual event surfaces, namely VE and EEE surfaces are modelled under the geometric framework to derive their counterpart object, namely partial VE and partial EEE surfaces, which contain the exact information of all possible visual event surfaces given the imprecision in the input. Correspondingly, a new definition of visual number is proposed to label the cells of Euclidean space partitioned by partial VE and partial EEE surfaces. The overall algorithm maintains the same computational complexity as the classical method and generates a partial visual hull which converges to the classical visual hull as the input converges to an exact value.

Keywords-visual hull; shape from silhouettes; imprecise input; solid domain; quadratic surface

I. INTRODUCTION

The visual hull is the shape approximation of a geometric object from its silhouettes and captures all the geometric information that we can gain from the silhouettes of the target object(s). The visual hull concept was introduced by Laurentini [1] and is used in many image-based reconstruction methods [2], [3], [4], [5]. Since the visual hull contains the target object and it is a better approximation than the convex hull, it has great advantage in obstacle avoidance, robotic navigation [6], 3D model acquisition and human motion tracking [7]. Since it is straightforward to extract silhouettes of an object from its pictures, the visual hull reconstruction is widely used in applications.

We address here the issue of robustness of the visual hull in 3D, which is far more involved than the 2D case as treated in [8]. Imprecision in the silhouette capturing process may vary the shape and size of the visual hull. Understanding the geometric changes of the visual hull with respect to a polyhedral scene given the imprecision in its vertices is the motivation of our research. We formulate the visual hull and its generating method under the partial geometric scheme, which robustly captures the visual geometric information with given imprecision in the input but maintains the same computational complexity as the classical method. The overview of our method is given in Fig. 1.

The geometric (solid) domain is a framework designed to capture imprecision in the input of geometric methods. The

basic building blocks including partial points, lines, planes, polygons and polyhedra are properly defined [9], [8]. Also, several geometric methods are re-designed or extended to the geometric domain, including convex hull [10], Delaunay Triangulation [11], [12] and Voronoi Diagram [11]. In visual hull reconstruction or any multi-view image-based reconstruction, data are captured in digital images. When the imprecision of the images (*e.g.* imprecision from capturing a point with a square pixel) is projected back to 3D, a half-infinite-polyhedral visual cone is obtained. The intersection of visual cones from different viewing points is a polyhedron in the space which is the best approximation to a tiny piece of the 3D object. In the geometric domain, a point is represented as a sequence of nested convex polyhedra approximating the classical point at different precision levels, which makes it a suitable framework for analysing imprecision in the visual hull and other image-based computer vision methods.

II. PARTIAL GEOMETRIC OBJECTS

We review the concept of partial geometric objects including partial points, partial line segments, partial lines, partial faces, partial surfaces and partial polyhedron, which are the building blocks of the partial visual hull concept.

A partial geometric object is defined by 3 disjoint sets whose union is \mathbb{R}^3 : interior, exterior and partial thick boundary. The interior (respectively, exterior) is an open set that contains all the points definitely known to be inside (respectively, outside) the object. The partial thick boundary is a closed set that contains the non-determinate points. Partial objects, ordered with subset inclusion, form a partial order and provide a domain in which each object approximates the target object, which is a maximal element, at a certain level of precision. These notions are formalized in [9] and used in representing the geometric computational results from input with imprecision. When we talk about a partial geometric object in this paper, we usually refer to its partial thick boundary. It is quite similar to the classical cases: when we consider a polyhedron, we usually refer to its boundary rather than its interior or exterior.

Basic partial geometric objects like partial points as in Fig. 2 (b), partial line segments as in Fig. 2 (c), partial

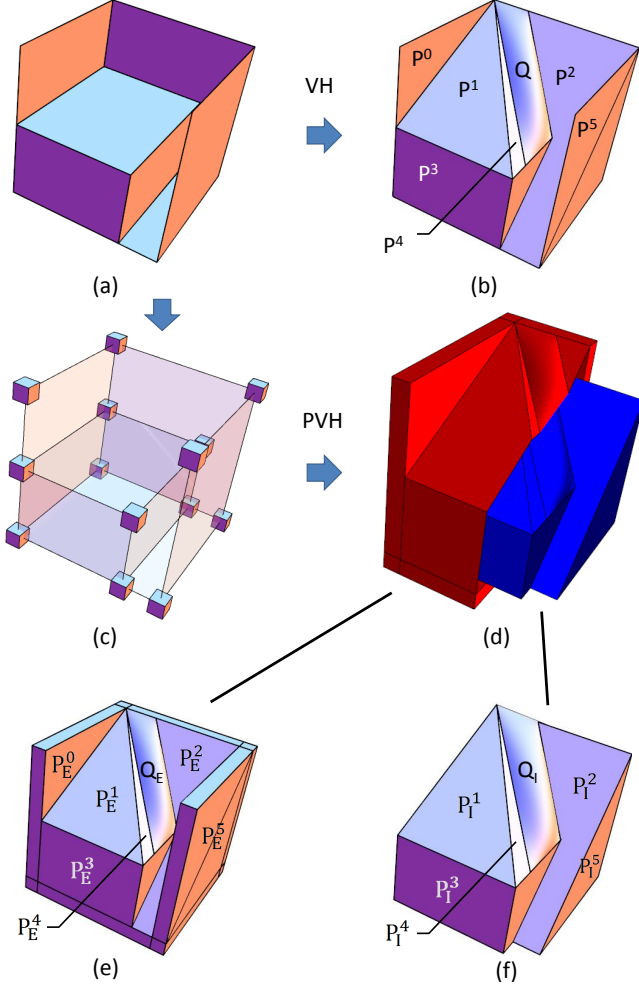


Figure 1. An overview of the partial visual hull generation. (a) A classical polyhedral scene. (b) With classical visual hull method, we can generate the classical visual hull. The face Q is a quadratic surface patch. (c) With imprecise input, we generate the partial points corresponding with all the vertices. (d) With the method presented in this paper, we give a partial visual hull including: the exterior (points definitely outside the visual hull) and the interior (points definitely inside the visual hull). (e) The boundary of the exterior. (f) The boundary of the interior. Note that each boundary face of the exterior or interior has a corresponding face on the classical visual hull.

lines as in Fig. 3 (e), partial planar faces as in Fig. 2 (d) and partial planar surfaces as in Fig. 3 (f) have partial thick boundaries but empty interiors. The partial point is the approximation of a point from several given directions. To keep the presentation simple in this paper, we use partial points approximated by the x , y and z coordinates, *i.e.* hyper-rectangles with sides parallel to the axes. A hyper-rectangular partial point has 8 vertices of different corner types (*e.g.* top-left corner of the front face). The corner types used in this paper are enumerated in Fig. 2 (b'). All our results can be extended to more general partial points in the shape of convex polyhedra as defined in [9]. Partial line segments and partial planar faces are convex

hulls of their partial vertices (points). Partial lines and partial planer surfaces are the union of the possible lines and planar surfaces restricted by their partial vertices. A partial polyhedron as in Fig. 2 (e) is defined as the union of its partial planar faces. Its interior as in Fig. 2 (f) and exterior as in Fig. 2 (g) are derived thereafter. A more detailed description can be found in [8].

For a partial object P , we denote by $SEL(P)$ the set of object selections with respect to P . A selection is a possible classical object given the partial object. In the case of a partial polyhedron from partial points, a polyhedron selection is a classical polyhedron constructed by selecting exactly one point in each partial point and connecting them according to the given topology.

In this paper, we assume that the target polyhedral scene (*i.e.*, a finite set of polyhedra) is given as a set of vertices and their topological connection, *i.e.* the scene is formed by connecting a given set of vertices in a given topological order. Due to the limitation of measuring devices, the vertices are assumed to be imprecisely given as partial points (*i.e.*, hyper-rectangles). Therefore, we model the scene as a set of partial polyhedra that we call a partial scene (PSN). Our aim is to find a partial geometric object called a partial visual hull (PVH) with its interior (PVH_I), respectively exterior (PSN_E), containing all the points that are definitely inside, respectively outside, the visual hull of the scene.

III. COMPLEXITY OF THE CLASSICAL 3D VISUAL HULL

In this section, we briefly review the concepts in the visual hull concept: the visual number and visual active surfaces. We also describe the classical 3D visual hull construction method, which is the basis of the method in constructing the partial visual hull from input with imprecision.

A. Visual Number and Visual Active Surfaces

The 2D visual number is defined as the number of ways visual lines can penetrate through the objects. However, the 3D case is defined in a different way. The visual number of a point with respect to a set of polyhedra is defined as the number of edges (or faces) of the visual cones [1], which will give the 3D visual hull with a subtly different partitioning method compared to the 2D method in [1].

Imagine a local sphere (circle) centred at the view point. The visual lines are half lines emitted from the view point and each of them has one intersection point with the local sphere (circle). If the visual line is blocked by the polyhedral scene, we paint the intersection point black. The non-topological definition of visual number is the number of disconnected black regions. However, the topological visual number is defined as the number of edges of the black regions.

In the classical 3D visual hull configuration, the visual active surfaces are defined as planar or quadratic surfaces

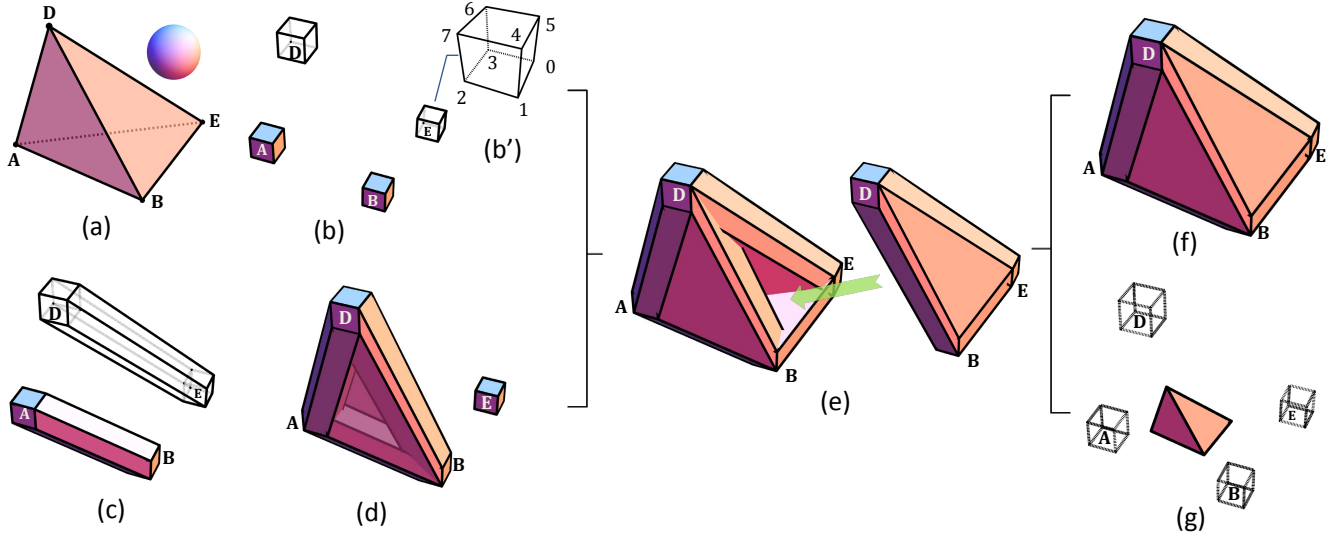


Figure 2. An overview of the basic partial geometric objects: (a) A classical geometric object (tetrahedron) with a normal-colour reference sphere. (b) Four partial points corresponding to the vertices in the classical tetrahedron. Two of them are presented in wireframes; (b') The partial point E is enlarged with 8 corner types. (c) Two partial edges respectively formed by partial points $\{A, B\}$ and $\{E, D\}$. The partial edge formed by $\{E, D\}$ is sketched in wireframe to show the structure at the back. (d) A partial face formed by partial points $\{A, B, D\}$. The internal structure is revealed with a transparent face. (e) A partial polyhedron (tetrahedron). The partial face $\{B, E, D\}$ is lifted so that the interior can be observed. (f) The boundaries of the exterior of the partial tetrahedron. (g) The boundaries of the interior of the partial tetrahedron. The skeletons of partial points are used to indicate the relative position of the interior.

consisting of patches serve as the boundaries between regions of different visual numbers. There are two kinds of visual active surfaces. A vertex-edge (VE) surface is the plane formed by a vertex and an edge of the polyhedral scene. A triple edge (EEE) surface is the quadratic surface formed by 3 straight lines, each of which coincides with an edge of the scene.

B. Construction of the Classical 3D Visual Hull

In [1], Laurentini presents a method that generates the exact visual hull of a polyhedral scene mainly in two steps: partitioning and labelling.

In the partitioning phase, a set of visual active surfaces are constructed, which are potential boundaries of regions with different visual numbers. They interweave with each other and the \mathbb{R}^3 space is divided into a number of cells. Due to the property of visual active surfaces, the space is over-partitioned, *i.e.* a cell may have the same visual number with its neighbouring cells. However all points inside a cell are of the same visual number. The visual active surfaces can be planes, hyperboloids of one sheet and, in some unlikely cases, hyperbolic paraboloids.

By definition, the visual hull is the set of points with visual number 0. In the labelling phase, each cell is given a visual number. Since the visual number is constant in each cell, we randomly pick a point inside a given cell and find its visual number which will be the visual number of the whole cell. The detailed algorithm can be found in the original paper [1].

Finally, all cells with visual number 0 are merged together to give the visual hull.

IV. PARTIAL VISUAL EVENT SURFACES

A classical 3D surface S (e.g. a plane) partitions the \mathbb{R}^3 space into 2 disjoint open regions. If the surface has a canonical form, we can further distinguish the 2 regions as a positive region \hat{S} and a negative region \check{S} . For example, assume the plane pln is given by:

$$pln(a, b, c, d) = \{(x, y, z) | ax + by + cz + d = 0\}.$$

Then, the induced positive and negative regions are:

$$\widehat{pln}(a, b, c, d) = \{(x, y, z) | ax + by + cz + d > 0\},$$

$$\widetilde{pln}(a, b, c, d) = \{(x, y, z) | ax + by + cz + d < 0\}.$$

Note that the positions of $\widehat{pln}(a, b, c, d)$ and $\widetilde{pln}(a, b, c, d)$ in \mathbb{R}^3 , given by the normal (in two different directions) of $pln(a, b, c, d)$ and the distance from the origin to pln , depend continuously on (a, b, c, d) . As explained in [13], there is also a continuously parameter-dependant scheme in generating a quadratic surface with an ordered triple of skewed lines (see [14] for the scheme). The positive region \hat{Q} and the negative region \check{Q} with respect to a quadratic surface Q can be defined in the same way.

A partial surface PS is a partial object partitioning the \mathbb{R}^3 space into 3 disjoint parts: a positive region \widehat{PS} , a negative region \widetilde{PS} and a partial thick boundary $\partial_P PS$. The 3 parts are separated by 2 classical surfaces: the positive surface s^+

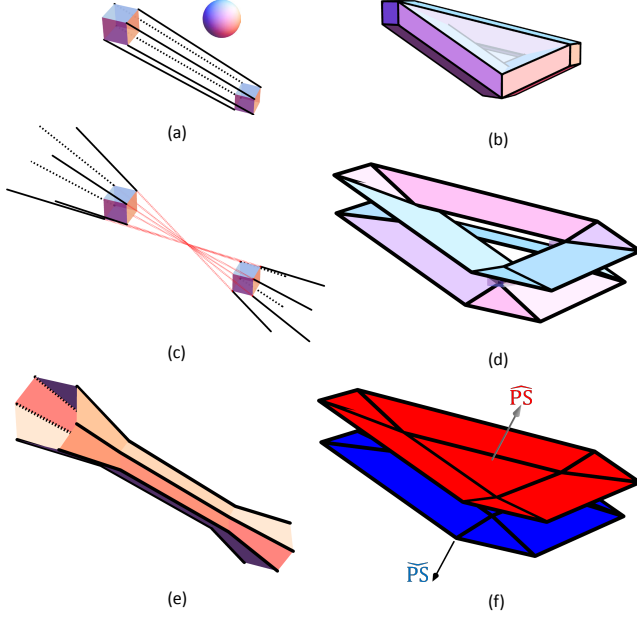


Figure 3. Generating a partial line and a partial planar surface. **(a)** The skeleton of a partial line segment and a normal-colour reference sphere. **(b)** A partial face. **(c)** The skeleton of the rest of the partial line. **(d)** The skeleton of the rest of the partial planar surface. **(e)** By combining (a) and (c), we generate a partial line. **(f)** By combining (b) and (d), we generate a partial planar surface. The red and blue surfaces are respectively positive (s^+) and negative (s^-) surfaces. The silver and black arrows respectively represent the positive (\overline{PS}) and negative (\underline{PS}) regions.

and the negative surface s^- . We have *i)* $\widehat{PS} = \widehat{s^+}$; *ii)* $\widetilde{PS} = \widetilde{s^-}$; *iii)* $\partial_P PS = \widehat{s^+} \cap \widetilde{s^-}$. The following relationship is implied when using positive and negative surfaces ¹:

$$s^+ \subseteq (\widetilde{s^-})^c \text{ and } s^- \subseteq (\widehat{s^+})^c$$

The partial surface PS is uniquely defined if we derive any of the following 3 sets of geometrical shapes, (i) $\partial_P PS$, (ii) \widehat{PS} and \widetilde{PS} , and (iii) s^+ and s^- .

For positive and negative regions, we obtain:

Proposition 4.1: For any classical surface T ,

$$((\widehat{T})^c)^\circ = \widetilde{T} \text{ and } ((\widetilde{T})^c)^\circ = \widehat{T} \quad (1)$$

For a set of classical surfaces, $\{s_0, s_1, \dots, s_t\}$,

$$\begin{aligned} ((\bigcap_{k=0}^t \widetilde{s}_k)^c)^\circ &= \bigcup_{k=0}^t \widehat{s}_k; & ((\bigcup_{k=0}^t \widetilde{s}_k)^c)^\circ &= \bigcap_{k=0}^t \widehat{s}_k; \\ ((\bigcap_{k=0}^t \widehat{s}_k)^c)^\circ &= \bigcup_{k=0}^t \widetilde{s}_k; & ((\bigcup_{k=0}^t \widehat{s}_k)^c)^\circ &= \bigcap_{k=0}^t \widetilde{s}_k. \end{aligned}$$

For convenience, in the rest of the paper, the index i takes values $i = 0, 1, 2$, and addition of indices in subscripts is $\text{mod } 3$, i.e. l_{i+k} means $l_{i+k(\text{mod } 3)}$. Another convenient symbol we often use is $[pe]$ which denotes the classical straight line generated by extending an edge (line segment or half line) pe .

¹ A^c and A° denote the complement and interior of a set A .

A. Partial VE Visual Event

A partial VE visual event contains the information of all classical visual events it represents. Formally, we give a definition as follows.

We denote by $pln(p, e)$ the plane containing the point p and the line segment (edge) e with $p \notin e$. Similarly $pln(p_0, p_1, p_2)$ denote the plane that contains the three non-collinear points p_0, p_1 and p_2 .

Definition 4.2: Suppose PP is a partial point and PE a partial edge with $PP \cap PE = \emptyset$. The partial VE visual event surface, partial VE for short, $PVE(PP, PE)$ is:

$$PVE(PP, PE) := \bigcup pln(pp, pe)$$

where $pp \in SEL(PP)$ and $pe \in SEL(PE)$.

Essentially, a partial VE visual event surface is a partial surface constructed from a partial point and a partial edge. Assume we have a partial point PP with vertices $\{p^0, p^1, \dots, p^7\}$ and a partial edge PE constructed by 2 partial vertices (points) PV_0 with vertices $\{pv_0^0, pv_0^1, \dots, pv_0^7\}$ and PV_1 with vertices $\{pv_1^0, pv_1^1, \dots, pv_1^7\}$. Consider the 16 extremal planes $s^j = pln(p^j, pv_0^j, pv_1^j)$ and $t^j = pln(p^j, pv_0^{7-j}, pv_1^{7-j})$ where $j = \{0, 1, \dots, 7\}$.

Proposition 4.3: In the non-degenerate case, there are four extremal planes, s_0, s_1, t_0 and t_1 that bound the partial VE surface generated by PP and PE and satisfy:

$$\begin{aligned} \partial_P PP \subseteq \widehat{s}_0 \text{ \& } \partial_P PE \subseteq \widehat{s}_0 & \quad \partial_P PP \subseteq \widetilde{s}_1 \text{ \& } \partial_P PE \subseteq \widetilde{s}_1 \\ \partial_P PP \subseteq \widehat{t}_0 \text{ \& } \partial_P PE \subseteq \widehat{t}_0 & \quad \partial_P PP \subseteq \widetilde{t}_1 \text{ \& } \partial_P PE \subseteq \widetilde{t}_1. \end{aligned}$$

These planes are the *characteristic planes* of the partial VE visual event surface, which can be constructed as:

$$\begin{aligned} \widehat{PVE}(PP, PE) &= \widehat{s}_0 \cap \widehat{t}_0 \cap \widehat{t}_1 \\ \widetilde{PVE}(PP, PE) &= \widetilde{s}_0 \cap \widetilde{t}_0 \cap \widetilde{t}_1 \end{aligned}$$

B. Partial Quadratic Surface

Recall the classical situation where an EEE visual event surface is a quadratic surface $Q(l_0, l_1, l_2)$ generated from 3 skewed lines l_0, l_1, l_2 as the union of all straight lines, each of which intersects the three skewed lines (see [15], [14]). A partial quadratic surface is a partial object generated from 3 partial lines. Here we assume those 3 partial lines are pairwise disjoint. Note that $Q(l_0, l_1, l_2) = \emptyset$ if exactly two of l_i 's are coplanar.

Definition 4.4: Given 3 partial lines PL_0, PL_1 and PL_2 , the partial quadratic surface $PQ(PL_0, PL_1, PL_2)$ is defined by its positive and negative regions as:

$$\begin{aligned} \widehat{PQ}(PL_0, PL_1, PL_2) &= \bigcap \widehat{Q}(pl_0, pl_1, pl_2); \\ \widetilde{PQ}(PL_0, PL_1, PL_2) &= \bigcap \widetilde{Q}(pl_0, pl_1, pl_2); \end{aligned}$$

where $pl_i \in SEL(PL_i)$, $i = 0, 1, 2$. (See the remark before Prop. 4.1).

In fact, the quadratic surface $Q(pl_0, pl_1, pl_2)$ is a selection of $PQ(PL_0, PL_1, PL_2)$, i.e., $Q(pl_0, pl_1, pl_2) \in$

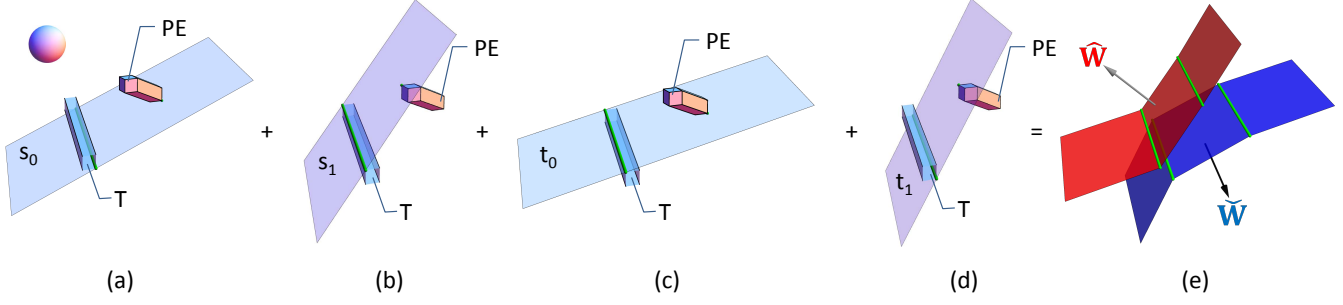


Figure 4. The four characteristic planes and the partial wedge (\widehat{W}) generated from a partial edge (PE) and an infinite tube (T). (a) A normal-colour reference sphere and an s_0 type plane: PE and T are both in \widehat{s}_0 , i.e. PE and T are both above s_0 . (b) An s_1 type plane: PE and T are both in \widehat{s}_1 , i.e. PE and T are both beneath s_1 . (c) A t_0 type plane: PE is in \widehat{t}_0 and T is in \widehat{t}_0 , i.e. PE is above t_0 and T is beneath t_0 . (d) A t_1 type plane: PE is in \widehat{t}_1 and T is in \widehat{t}_1 , i.e. PE is beneath t_1 and T is above t_1 . (e) The partial wedge generated from these 4 types of planes. The red surface is the positive surface (w^+) while the blue one is the negative surface (w^-). The silver and black arrows respectively represent the positive (\widehat{W}) and negative (\widehat{W}) regions.

$SEL(PQ(PL_0, PL_1, PL_2))$. However, there are infinitely many classical quadratic surface selections. The following theorem, whose proof is given in [14], reduces the problem to a finite number.

Theorem 4.5: Given 3 partial lines PL_0, PL_1 and PL_2 :

$$\widehat{PQ}(PL_0, PL_1, PL_2) = \bigcap \widehat{Q}([pe_0], [pe_1], [pe_2]); \quad (2)$$

$$\widetilde{PQ}(PL_0, PL_1, PL_2) = \bigcap \widetilde{Q}([pe_0], [pe_1], [pe_2]); \quad (3)$$

where the two intersections above are taken over all edges pe_i of $PL_i, i = 0, 1, 2$.

V. CONSERVATIVE APPROXIMATION

The partial quadratic surface generation method presented in the previous section is an exact derivation of the visual event surfaces with the given imprecision, which means that the generated partial quadratic surface represents all the possible classical visual event surfaces given by the 3 skewed partial lines. However, it requires the consideration of all possible quadratic surfaces generated by the edges of 3 partial lines, one from each. A partial line generated by 2 partial points in the shape of hyper-rectangles has at least 12 non-collinear edges. The number of quadric surfaces we would need to consider is therefore $12^3 = 1728$. The number of intersections between these surfaces is $C_{1728}^2 = 1492128$. This is quite expensive since we need to apply the same algorithm to each of the EEE event.

To balance the computational burden and exactness of our algorithm, we propose a conservative approximation of an EEE event, which only considers 6 quadratic surfaces whose spatial relationship is known.

Given three skewed edges, a classical EEE visual event surface (EEE surface for short) is defined as the quadratic surface swept by straight lines each of which intersects all the three edges or their extensions [1]. The classical EEE visual active patch (EEE patch for short) is defined as a patch of the quadratic surface, which is the union of all line segments, each intersecting all three edges. It

is called the active patch since in the visual hull, only an EEE patch or part of it can actively participate in forming the boundary of the visual hull. Other parts of the EEE surface do not separate \mathbb{R}^3 regions with different visual numbers, and therefore do not contribute in forming the visual hull boundary. However, since the 3D visual hull algorithm requires the partition of the 3d space, we need to keep the whole quadratic surface.

Definition 5.1: A partial EEE surface of three finite partial edges is a partial surface whose partial (thick) boundary contains all straight lines that intersect all 3 partial edges. However, there are infinitely many partial EEE surfaces according to the definition, including the partial quadratic surface given in Section IV-B. Our target in this section is to find a partial EEE surface that is easy to compute, reasonably approximates the given scene with known imprecision and converges to the classical scene with exact input.

Given a partial edge $PL(A, B)$ with partial end points A and B , we can generate an infinite polygonal tube $T(A, B)$ with the following method:

- extend the 2 partial end points A and B to the smallest partial points A' and B' , such that A' and B' are of the same size with $A \subseteq A'$ and $B \subseteq B'$;
- connect the corresponding corner types to form 8 mutually parallel corner lines;
- construct the infinite tube as the convex hull of the corner lines generated.

Below, we extend the definition of the function “ \square ” so that $[PE]$ is the infinite tube extended from a partial edge PE .

A quadratic surface $Q(l_0, l_1, l_2)$ is generated by three straight lines l_0, l_1 and l_2 . It can be partitioned into 3 disjoint parts $Q(l_0, l_1, l_2)|_{l_{i+1}}^{l_i}$. Each is a surface patch on $Q(l_0, l_1, l_2)$ that lies in between $i) l_i$ and l_{i+1} , and $ii)$ does not contain l_{i+2} .

Here, we need a method that generates a partial plane from a partial point and an infinite tube. The partial point PP is a hyper-rectangle with 8 vertices namely p^0, p^1, \dots, p^7

and the infinite tube T has 8 mutually parallel straight lines namely l^0, l^1, \dots, l^7 (some of which are in the interior of the tube and thus are not boundary edges), each corresponding to a corner type. There are 16 extremal planes namely the planes s^j through vertex p^j and line l^j and the planes t^j through vertex p^{7-j} and line l^j ($j = 0, 1, \dots, 7$). Note that p^j and p^{7-j} are two corners of opposite types, e.g. if p^j is the top-left corner on the front face, p^{7-j} is the bottom-left corner on the back face. See [8] for the definition of the corner types for general partial points.

Proposition 5.2: In the non-degenerate cases there are 4 extremal planes, say, s_0, s_1, t_0 and t_1 that bound the partial surface (plane) $PS(PP, T)$ generated by PP and T , and satisfy:

$$\begin{aligned} PP \subseteq \hat{s}_0 \ \& \ T \subseteq \hat{s}_0; & PP \subseteq \check{s}_1 \ \& \ T \subseteq \check{s}_1; \\ PP \subseteq \hat{t}_0 \ \& \ T \subseteq \hat{t}_0; & PP \subseteq \check{t}_1 \ \& \ T \subseteq \check{t}_1. \end{aligned}$$

These four planes are the *characteristic planes* of PP and T , which defined the the partial plane as follows:

$$\begin{aligned} \widehat{PS}(PP, T) &= \hat{t}_0 \cap \hat{t}_1 \cap \hat{s}_1; \\ \widetilde{PS}(PP, T) &= \check{s}_0 \cap \check{t}_0 \cap \check{t}_1. \end{aligned}$$

Actually, in the above proposition, if we replace the partial point PP with a partial edge PE , we can still find 4 classical planes, namely s_0, s_1, t_0 and t_1 , that bound a partial object called a *partial wedge*; see also Fig. 4.

Definition 5.3: A partial wedge $W(PE, T)$ generated by an infinite tube $T(P_0, P_1)$ and a partial edge $PE(Q_0, Q_1)$, where P_k and Q_k are partial points ($k = 0, 1$), is given by

$$\partial_P W(PE, T) := \bigcup \{pln(l, t) \mid l \in SEL(T), t \in \partial_P PE\}.$$

Note that any selection of an infinite tube is a straight line that goes through the two partial points and is parallel to the edges of the infinite tube. Note also that a selection of the partial wedge is thus given by $pln(l, t)$ with $l \in SEL(T)$ and $t \in \partial_P PE$.

By Prop. 5.2, we assume s_m^k and t_m^k are the characteristic planes corresponding to T and Q_k , with $k, m = 0, 1$.

$$\widehat{W}(PE, T) = \bigcap_{k=0,1} \hat{t}_0^k \cap \hat{t}_1^k \cap \hat{s}_1^k \quad (4)$$

$$\widetilde{W}(PE, T) = \bigcap_{k=0,1} \check{s}_0^k \cap \check{t}_0^k \cap \check{t}_1^k \quad (5)$$

Using the three partial edges PE_0, PE_1 and PE_2 we can construct 2 partial wedges $W(PE_{i+1}, [PE_i])$ and $W(PE_{i+2}, [PE_i])$ for each PE_i . For a classical straight line l intersecting all three partial edges, we have:

$$l \subseteq \partial_P W(PE_{i+1}, [PE_i]) \cap \partial_P W(PE_{i+2}, [PE_i]).$$

Let $s_{m,(i+n,i)}^k$ and $t_{m,(i+n,i)}^k$ with $k, m = 0, 1$ be the characteristic planes defined in Def. 5.3 of $W(PE_{i+n}, [PE_i])$

with $n = 1, 2$. Their intersection with each other are straight lines which are mutually parallel.

Definition 5.4: The positive line l_i^p and negative line l_i^n are given by:

$$\begin{aligned} l_i^p &:= w^+(PE_{i+1}, [PE_i]) \cap w^+(PE_{i+2}, [PE_i]); \\ l_i^n &:= w^-(PE_{i+1}, [PE_i]) \cap w^-(PE_{i+2}, [PE_i]); \end{aligned}$$

where the surfaces $w^+(PE_{i+r}, [PE_i])$ and $w^-(PE_{i+r}, [PE_i])$ are respectively the positive and negative surfaces of the partial wedge $W(PE_{i+r}, [PE_i])$, $r = 1, 2$.

Proposition 5.5: For $i = 0, 1, 2$, the lines l_i^p and l_i^n satisfy the following properties:

$$\begin{aligned} \exists u, v \in \{s_{1,(i+r,i)}^k, t_{m,(i+r,i)}^k \mid k, m = 0, 1; r = 1, 2\}. \\ l_i^p = u \cap v \ \& \ l_i^p \subseteq (\check{u})^c \ \& \ l_i^p \subseteq (\check{v})^c; \\ \exists u, v \in \{s_{0,(i+r,i)}^k, t_{m,(i+r,i)}^k \mid k, m = 0, 1; r = 1, 2\}. \\ l_i^n = u \cap v \ \& \ l_i^n \subseteq (\hat{u})^c \ \& \ l_i^n \subseteq (\hat{v})^c. \end{aligned}$$

For any plane pl that is a selection of $W(PE_{i+1}, [PE_i])$ and $W(PE_{i+2}, [PE_i])$, we have:

$$l_i^p \subseteq \hat{pl} \ \text{and} \ l_i^n \subseteq \check{pl}.$$

Theorem 5.6: A conservative partial EEE surface $PEEE_c$ from 3 partial edges PE_i can be constructed as follows:

$$\begin{aligned} PEEE_c^+ \Big|_{l_{i+1}^p} &= Q(l_i^p, l_{i+1}^p, l_{i+2}^p) \Big|_{l_{i+1}^p}; \\ PEEE_c^- \Big|_{l_{i+1}^n} &= Q(l_i^n, l_{i+1}^n, l_{i+2}^n) \Big|_{l_{i+1}^n}. \end{aligned}$$

See [14] for the proof; the generation procedure is visualized in Fig. 5.

VI. 3D VISUAL NUMBER LABELLING

In the classical case, after establishing the partition of the space, we label each cell with an appropriate visual number. Given a polygonal scene S and a view point p , the classical 3D visual number $VN3D(p, S)$ was defined in [1]:

Definition 6.1: The 3D visual number $VN3D(p, S)$ is the number of edges or faces of the visual cone of p relative to S .

This definition is not a natural extension from the 2D visual number. The definition of $VN3D$ depends too rigidly on the geometry of the scene [16]. Therefore, $VN3D$ is not suitable for our framework since the interior and exterior of an object contain only limited information about the scene geometry.

Definition 6.2: The *reduced visual number* $\overline{VN3D}(p, S)$ is the number of families of visual numbers viewed from p with respect to S , i.e., if we consider a unit sphere centre at p and project S onto the sphere as silhouettes, then the visual number $\overline{VN3D}(p, S)$ is the number of disconnected regions of the non-shadowed area.

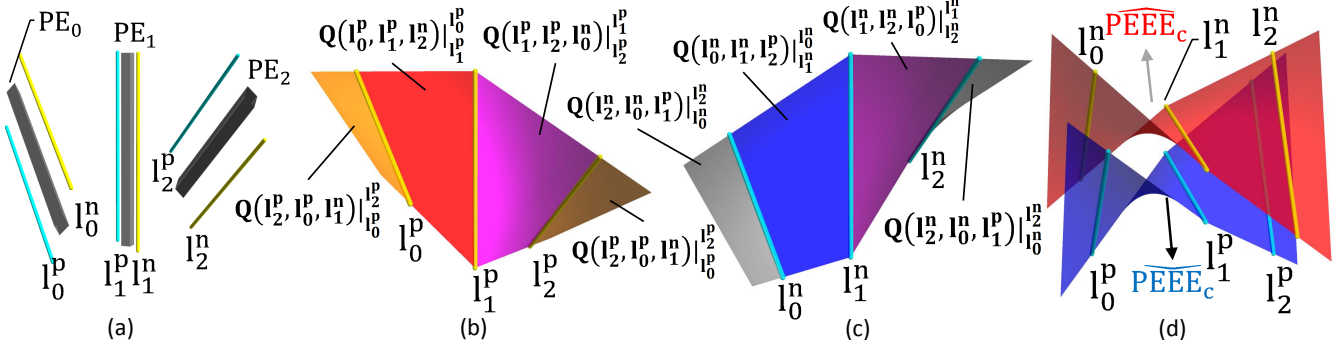


Figure 5. An overview of generating a conservative partial EEE surface. (a) Three partial edges (PE_i) and their positive (l_i^p) and negative (l_i^n) lines, respectively coloured in yellow and cyan. (b) The positive surface ($PEEE_c^+$) consists of 3 regions coloured in red, magenta and orange and labeled as in Theorem 5.6. (c) Similarly the negative surface ($PEEE_c^-$) consists of 3 regions, which are respectively coloured in blue, purple and gray. (d) The side view of the partial EEE visual surface. The red surface is the positive surface and the blue one is the negative surface. The silver and black arrows respectively represent the positive ($PEEE_c^+$) and negative ($PEEE_c^-$) regions.

The number $\overline{VN3D}(p, S)$ does not rigidly depend on the topology of the scene. The relationship between $VN3D(p, S)$ and $\overline{VN3D}(p, S)$ is as follows.

Proposition 6.3: Given a polygonal scene S and a view point p ,

- (i) $VN3D(p, S) = 0$ iff $\overline{VN3D}(p, S) = 0$.
- (ii) $VN3D(p, S) \geq \overline{VN3D}(p, S)$.

We now define the partial visual hull $PVH(PSN)$ for a given partial scene PSN . For convenience, in the rest of this section we put $\mathcal{S} := SEL(PSN)$.

$$PVH_I(PSN) = \bigcup \{p | VN3D(p, psn) = 0, psn \in \mathcal{S}\}$$

$$PVH_E(PSN) = \bigcup \{p | VN3D(p, psn) \neq 0, psn \in \mathcal{S}\}.$$

According to Prop. 6.3 (i), we have

$$PVH_I(PSN) = \bigcup \{p | \overline{VN3D}(p, psn) = 0, psn \in \mathcal{S}\}$$

$$PVH_E(PSN) = \bigcup \{p | \overline{VN3D}(p, psn) \neq 0, psn \in \mathcal{S}\}.$$

For the partial polygonal scene PSN , we now obtain two different notions, (i) the *partial visual number set* $PVN3D(p, PSN)$, and, (ii) the *reduced partial visual number set* $\overline{PVN3D}(p, PSN)$:

Definition 6.4:

$$PVN3D(p, PSN) := \bigcup \{VN3D(p, psn) | psn \in \mathcal{S}\}$$

$$\overline{PVN3D}(p, PSN) := \bigcup \{\overline{VN3D}(p, psn) | psn \in \mathcal{S}\}.$$

Therefore, we have:

$$PVH_I(PSN) = \bigcup \{p | \overline{PVN3D}(p, PSN) = \{0\}\};$$

$$PVH_E(PSN) = \bigcup \{p | 0 \notin \overline{PVN3D}(p, PSN)\}.$$

For polyhedral scenes $psn, psn' \in \mathcal{S}$, if every polyhedron in psn is contained in some polyhedron in psn' , then:

$$\overline{VN3D}(p, psn) \geq \overline{VN3D}(p, psn').$$

From this, by considering the classical boundaries ∂PSN_I and ∂PSN_E as polyhedral scenes, we obtain:

Proposition 6.5: For any $psn \in \mathcal{S}$,

$$\overline{VN3D}(p, \partial PSN_I) \geq \overline{VN3D}(p, psn);$$

$$\overline{VN3D}(p, \partial PSN_E) \leq \overline{VN3D}(p, psn).$$

Therefore, we can use $VN3D(p, \partial PSN_I)$ and $VN3D(p, \partial PSN_E)$ to obtain an upper and a lower bound for $PVN3D(p, PSN)$ and thus generate a conservative partial visual hull $CPVH(PSN)$ with respect to the partial scene PSN :

Definition 6.6:

$$CPVH_I(PSN) := \bigcup \{p | VN3D(p, \partial PSN_I) = 0\};$$

$$CPVH_E(PSN) := \bigcup \{p | VN3D(p, \partial PSN_E) \neq 0\}.$$

We now deduce the following result.

Theorem 6.7:

$$CPVH_I(PSN) \subseteq PVH_I(PSN);$$

$$CPVH_E(PSN) \subseteq PVH_E(PSN).$$

In fact, if $VN3D(p, \partial PSN_I) = 0$, then by Prop. 6.3, we have $\overline{VN3D}(p, \partial PSN_I) = 0$. By Prop. 6.5, $\overline{VN3D}(p, psn) = 0$ for all $psn \in \mathcal{S}$. Therefore, $p \in PVH_I(PSN)$. The exterior case follows in a similar way.

VII. OVERALL ALGORITHM

The overall algorithm for generating the 3D conservative partial visual hull resembles the classical algorithm. From a given polyhedral scene with imprecision information, we can construct its conservative partial visual hull as follows:

- 1) Construct a partial polyhedral scene PSN , *i.e.* a set of partial polyhedra, according to the given information and imprecision. This provides the interior PSN_I and exterior PSN_E of the scene;
- 2) Compute all potential visual event surfaces, including partial VE surfaces and conservative partial EEE surfaces;

- 3) Construct the partition of the whole \mathbb{R}^3 space interwoven by positive and negative surfaces of partial visual event surfaces into disjoint cells;
- 4) Calculate the upper and lower bounds of the partial visual number set for each cell;
 - a) Pick a point p inside a cell c .
 - b) Calculate the two bounds for the partial visual number set of c given by:

$$\begin{aligned} VN3D(c, \partial PSN_I) &:= VN3D(p, \partial PSN_I) \\ VN3D(c, \partial PSN_E) &:= VN3D(p, \partial PSN_E). \end{aligned}$$

- c) Merge all cells c with $VN3D(c, \partial PSN_I) = 0$ to obtain the interior $CPVH_I(PSN)$. Similarly, merge all cells c with $VN3D(c, \partial PSN_E) \neq 0$, to obtain the exterior $CPVH_E(PSN)$.

Since the partial visual number set, and therefore its two upper and lower bounds, are constant in each cell, the above algorithm correctly computes $CPVH(PSN)$ by the results of Section VI.

In the above algorithm, the computational complexity of each step is the same as the corresponding classical algorithm. For example, to generate an classical EEE surface, we consider one quadratic surface going through 3 lines. Similarly, to generate a conservative partial EEE surface, we consider 6 (constant) quadratic surfaces. Therefore, the overall computational complexity, which is $O(n^{12})$, remains unchanged. We finally note that the algorithm generates a partial visual hull which can be shown to converge to the classical visual hull as the input converges to an exact value.

VIII. CONCLUSION

In this paper, we provided an algorithm to construct a partial visual hull for a polyhedral scene with imprecise input. It has the same computational complexity as the classical visual hull construction algorithm. Many of the visibility related problems (e.g. antipenumbra [17]) require the generation of quadratic surfaces from skewed lines. As part of the solution, we were able to provide a method to approximate the partial quadratic surface generated from 3 edges with imprecision, whose application is not limited to the visual hull problem. See [14] for further work on the partial visual hull using the notion of the visibility complex.

REFERENCES

- [1] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, 1994.
- [2] G. K. Cheung, S. Baker, and T. Kanade, "Visual hull alignment and refinement across time: A 3d reconstruction algorithm combining shape-from-silhouette with stereo," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 2, p. 375, 2003.
- [3] K. Grauman, G. Shakhnarovich, and T. Darrell, "A bayesian approach to image-based visual hull reconstruction," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 1, p. 187, 2003.
- [4] S. Lazebnik, E. Boyer, and J. Ponce, "On computing exact visual hulls of solids bounded by smooth surfaces," *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 1, p. 156, 2001.
- [5] H. Kim, R. Sakamoto, I. Kitahara, T. Toriyama, and K. Kogure, "Reliability-based 3d reconstruction in real environment," in *Proceedings of the 15th international conference on Multimedia*, ser. MULTIMEDIA '07. New York, NY, USA: ACM, 2007, pp. 257–260.
- [6] D. Pizarro, M. Marron, D. Peon, M. Mazo, J. Garcia, M. Sotelo, and E. Santiso, "Robot and obstacles localization and tracking with an external camera ring," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, pp. 516–521.
- [7] J. Yan, J. Song, and Y. Liu, "Simultaneous 3-d human-motion tracking and voxel reconstruction," *Optical Engineering*, vol. 49, no. 9, p. 097201, 2010.
- [8] P. He, "Visual hull with imprecise input," in *International Conference on Computational Geometry and Computer Vision*, 2010, pp. 639–646.
- [9] A. Edalat and A. Lieutier, "Foundation of a computable solid modelling," *Theoretical Computer Science*, vol. 284, no. 2, pp. 319–345, June 2002.
- [10] A. Edalat, A. Lieutier, and E. Kashefi, "The convex hull in a new model of computation," in *Proceedings of the 13th Canadian Conference on Computational Geometry*, University of Waterloo, August 2001, pp. 93–96.
- [11] A. Edalat, A. A. Khanban, and A. Lieutier, "Delaunay triangulation and Voronoi diagram with imprecise input data," *Electronic Notes in Theoretical Computer Science*, vol. 66, no. 1, July 2002, proceedings of the 5th CCA Workshop.
- [12] A. A. Khanban and A. Edalat, "Computing Delaunay triangulation with imprecise input data," in *Proceedings of the 15th Canadian Computational Geometry Conference*, 2003, pp. 94–97.
- [13] J. F. Blinn, "Lines in space—part 6: Our friend the hyperbolic paraboloid," *IEEE Comput. Graph. Appl.*, vol. 24, pp. 92–100, May 2004.
- [14] "Visual hull from imprecise polyhedral scene - supporting material," Website, www.doc.ic.ac.uk/~ae/papers/visual-hull-supporting-information.pdf.
- [15] D. Hilbert and S. Cohn-Vossen, *Geometry and the Imagination*, 2nd ed. New York: Chelsea, 1999.
- [16] S. Petitjean, "A computational geometric approach to visual hulls," *International Journal of Computational Geometry and Applications (IJCGA)*, vol. 8, no. 4, pp. 407–436, 1998.
- [17] S. J. Teller, "Computing the antipenumbra of an area light source," *SIGGRAPH Comput. Graph.*, vol. 26, pp. 139–148, 1992.