[21] A.P. Morgan and A.J. Sommese, "Coefficient-parameter polynomial continuation," *Applied Mathematics and Computation*, vol. 29, pp. 123–160, 1989.
[22] A.N. Netravali, T.S. Huang, A.S. Krishnakumar, and R.J. Holt, "Algebraic methods in 3D motion estimation from two-view point correspondences," *Int'l J. Imaging Systems and Technology*, vol. 1, pp. 78–99, 1989.
[23] I.R. Shafarevich, *Basic Algebraic Geometry*, Springer-Verlag, New York, 1974.
[24] M. Stillman, M. Stillman, and D. Bayer, *Macaulay User's Manual, Version 3.0*, 1989.
[25] B. Sturmfels, "On the Newton polytope of the resultant," *J. Algebraic Combinatorics*, vol. 3, pp. 207–236, 1994.
[26] R.Y. Tsai and T.S. Huang, "Estimating three-dimensional motion parameters of a rigid planar patch, II: Singular value decomposition," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 30, pp. 525–534, 1982.
[27] S. Ullman, *The Interpretation of Visual Motion*, MIT Press, Cambridge, Mass., 1979.
[28] J. Weng, T.S. Huang, and N. Ahuja, "Motion and structure from two perspective views: Algorithm, error analysis, and error estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 451–476, 1989.

A Mechanism of Automatic 3D Object Modeling

Xiaobu Yuan

***Abstract*—The symbolic representation of 3D objects is the fundamental knowledge for computer systems to understand the environment. This knowledge is usually assumed to exist in a computer but can also be acquired by accumulating spatial features extracted from sensory inputs at different viewing directions. This paper first investigates surface visibility and, then, after introducing mass vector chains (MVC), discusses the relationship between MVC and the spatial closure of object models. An automatic modeling mechanism is established with the observation that the boundary of an object is closed only if the MVC of its model is closed or, alternatively, the tail-to-head vector of an unclosed MVC estimates the visible direction of the missing surfaces. Experimental results and an algorithm are also given at the end.**

***Index Terms*—Spatial reasoning, automatic processing, Gaussian sphere, object reconstruction, computer vision, surface visibility, occlusions, view planning.**

I. Introduction

Three-dimensional (3D) computer vision studies techniques that enable a computer system to understand its environment from various sensory inputs [1]. While the cognitive abilities of a vision system are normally investigated in the field of object recognition, its fundamental visual capabilities of acquiring symbolic representations from sensory inputs come from the research of object reconstruction [2].

There has been considerable discussion on the theoretical, as well as the practical, aspects of reconstruction systems [3], [4]. Significant results have been achieved from investigations on optical devices [5], spatial feature extraction techniques [6], [7], and modeling methods [8]. However, few attempts have been devoted to creating an automatic modeling system [4], [9], i.e., a reconstruction system that builds up object models without human intervention. The major problem is how to construct a system that can decide for itself the number and direction of necessary views from which all the spatial features of objects can be obtained.

In principle, a modeling system includes both feature extraction to extract the symbolic representation of spatial surfaces from input data and feature integration to accumulate individual surfaces into consistent object descriptions or object models. For regular convex objects, their models can be easily reconstructed with spatial features from two opposite or at most three viewing directions in which all of their boundary surfaces are visible. Unfortunately, 3D objects usually have complicated shapes. It is impossible to preset two, three, or even more viewing directions and guarantee that every boundary surface of an object is extractable from those directions. Therefore, in addition to the primary functions of extracting and accumulating features, a reconstruction system expects a self-controlled modeling mechanism for the system to check the spatial closure of object models and to estimate the direction of unprocessed features.

Manuscript received May 18, 1994; revised September 28, 1994.
The author is with Dept. of Computer Science, Memorial University of Newfoundland, St. John's, Newfoundland, Canada A1B 3X5; e-mail yuan@cs.mun.ca.
IEEECS Log Number P95034.

II. Surface Visibility

In boundary representation, an object is defined by a group of surfaces. As the boundary surfaces of different objects have different spatial shapes and different topological relations, the number and direction of views necessary to extract from a sensory device all of the boundary surfaces of 3D objects depend on the visibility of each individual boundary surface.

Suppose, in Fig. 1(a), an object is observed from a viewing coordinate system **r-u-v** [8], and the size of the object is relatively small in comparison to its distance to the viewing point $\mathbf{p}_0$. A surface $\mathbf{S}(\mu, \upsilon)$ of the object intersects along a curve $\mathbf{L}(\mathbf{r})$ with an observation plane, which is parallel to the **r- u** plane. At a point $\mathbf{x}$ on $\mathbf{L}(\mathbf{r})$, its shifted surface normal $\mathbf{n}$ projects a vector $\mathbf{n}'$ onto the **r-u** plane that in turn forms an angle θ with $\mathbf{r}$ reflecting the visibility of the surface in the viewing direction.

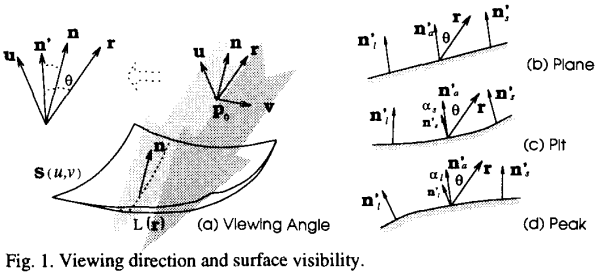


Fig. 1. Viewing direction and surface visibility.

The intersection curve $\mathbf{L}(\mathbf{r})$ may consist of small segments of flat, concave, and convex shapes. If $\mathbf{n}'_a$ is the average projected vector of $\mathbf{L}(\mathbf{r})$, between $\mathbf{n}'_a$ and the projected individual normal at a point, there is an angle α_s in Fig. 1(c) or α_l in Fig. 1(d), where subscripts s and l tell if the point is closer to or farther from $\mathbf{p}_0$. Since the intersection line $\mathbf{L}(\mathbf{r})$ is visible only if all of its surface points are visible, for a flat line in Fig. 1(b), the viewing vector $\mathbf{r}$ can move from $\mathbf{n}'_a$ at the 's' side, making θ less than $\pi/2$,and at the 'l' side, making θ bigger than $-\pi/2$. If α'_s is the largest angle among all α_s and α_l at the 's' side and α'_l is the largest at the 'l' side, the visible range of a general curve $\mathbf{L}(\mathbf{r})$ is within the limit between $\pi/2 - \alpha'_s$ and $-\pi/2 + \alpha'_l$.

Since $\mathbf{L}(\mathbf{r})$ moves along the $\mathbf{v}$ axis, to make all surface points on $\mathbf{V}(L)$ visible, the visible range $\mathbf{V}(\mathbf{r})$ of the surface takes the smallest conjunction of all $\mathbf{V}(L)$. Let $\theta(\mathbf{r})^+$ be the minimum of all the intersection lines at the 's' side and $\theta(\mathbf{r})^-$ the maximum at the 'l' side, the visible range $\mathbf{V}(\mathbf{r})$ of a general surface patch $\mathbf{S}(\mu, \upsilon)$ in the $\mathbf{r}$-$\mathbf{u}$ plane is then the integration of viewing directions within a range upper bounded by $\theta(\mathbf{r})^+$ and lower bounded by $\theta(\mathbf{r})^-$.

$$\mathbf{V}(\mathbf{r}) = \int_{\theta(\mathbf{r})^-}^{\theta(\mathbf{r})^+} \mathbf{r} d\theta \tag{1}$$

Nevertheless, $\mathbf{V}(\mathbf{r})$ represents only the latidudinal visibility of a surface. If the viewing direction $\mathbf{r}$ moves longitudinally around the surface in the $\mathbf{r}$-$\mathbf{v}$ plane of the viewing coordinate system, there is a new latidudinal visible range at each location. Let φ be the rotation angle around $\mathbf{n}_a'$ the visible range of a surface is integrated from all the latidudinal visible ranges when $\mathbf{r}$ makes a round sweep.

$$\mathbf{V}(S) = \int_0^{2\pi} \int_{\theta(\mathbf{r})^-}^{\theta(\mathbf{r})^+} \mathbf{r} d\theta d\varphi \tag{2}$$

The definition of $\mathbf{V}(S)$ shows that the visible range of an individual surface depends only on the shape of the surface. However, for objects exhibiting concave shapes, surface occlusion usually appears. In such case, the visible range of a surface patch becomes smaller because at certain φ there are surfaces blocking parts of its visible range (Fig. 2(a)). If the narrowed upper and lower limits are θ'^+ and θ'^-, the visible range of a surface has to be updated for occlusion.

$$\mathbf{V}'(S) = \int_0^{2\pi} \int_{\theta'(\mathbf{r})^-}^{\theta'(\mathbf{r})^+} \mathbf{r} d\theta d\varphi \tag{3}$$

(a) Changed Viewing Range (b) Changed Visibility

Fig. 2. Visibility under occlusion.

When the occluded visible range becomes so small that the surface is partially invisible, as illustrated in Fig. 2(b), it is referred to as a changed visibility since not only the visible range but also the visibility of the surface are changed.

III. Mass Vector Chains

When a 3D object is defined with its boundary surfaces, the representation of the object becomes a set of symbolically represented surfaces. While surface visibility reflects the shape of the object, in the process of building a model with spatial features extracted at different viewing locations the orientation of unobtained surfaces can be estimated from a mass vector chain of processed surfaces.

A. The Definition of Mass Vectors

Suppose the boundary of an object is defined by a group of m surface patches S_i, $0 \le i \le m-1$. Consider a tiny piece δs on a surface patch S_i in Fig. 3. All of the surface points on δs point to the same unit surface normal $\mathbf{n}(s)$ when δs is small enough. The average normal $\mathbf{n}_i$ of S_i is the integration of all of these unit normals over the entire surface divided by the surface.

$$\mathbf{n}_i = \frac{1}{S_i} \iint_{S_i} \mathbf{n}(s) ds \tag{4}$$

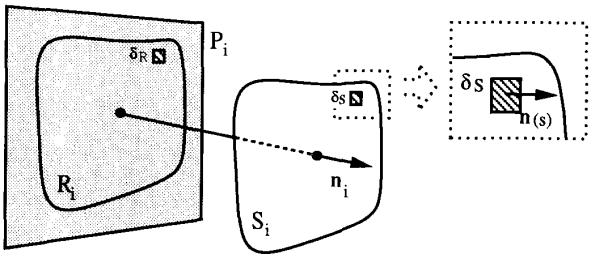


Fig. 3. The definition of mass vector.

Let plane P_i be perpendicular to $\mathbf{n}_i$. Surface patch S_i becomes a planar region R_i when projected onto P_i. As δs maps to δR by $(\mathbf{n}(s) \cdot \mathbf{n}_i)\, \delta s$, the projected region R_i of S_i is integrated from the projected tiny regions.

$$R_i = \iint_{S_i} (\mathbf{n}(s) \cdot \mathbf{n}_i) ds \tag{5}$$

By definition, a mass vector chain of an object is a series of weighted vectors. In this series, a vector $\vec{\mathbf{V}}_i$ is assigned to each individual surface patch S_i of the object. This vector points to the average normal direction $\mathbf{n}_i$ of the surface patch S_i; and its weight is the projected region R_i on a plane P_i perpendicular to $\mathbf{n}_i$.

$$\vec{\mathbf{V}}_i = \mathbf{n}_i R_i \tag{6}$$

For the surface patch, $\mathbf{n}_i$ is its average visible direction and R_i is the surface size when viewed in that direction.

B. MVC and Boundary Closure

It has been proved [10], [11] that, for an object of convex surfaces, the total Gaussian mass of the surfaces must be zero,

$$\iint_S G(\mathbf{n}(s))\mathbf{n}(s) ds = \oiint_S \mathbf{n}(s) ds = 0 \tag{7}$$

where $G(\mathbf{n}(s))$ is the Gaussian mass with a same $\mathbf{n}(s)$. This conclusion also applies to ordinary objects [12]. Since the total Gaussian mass of an object is the sum of subtotals of individual surface patches, (7) can be further grouped in terms of surfaces.

$$\oiint_S \mathbf{n}(s) ds = \sum_{j=0}^{m-1} \iint_{S_i} \mathbf{n}(s) ds = 0 \tag{8}$$

On the other hand, the total mass vector of an object is the sum of all its mass vectors, i.e. $\sum_{j=0}^{m-1} \vec{\mathbf{V}}_i$. Using (4) to (8), the total mass vector can be derived. The result points out that the boundary surfaces of an object compose a closed surface boundary only when their mass vectors form a closed chain.

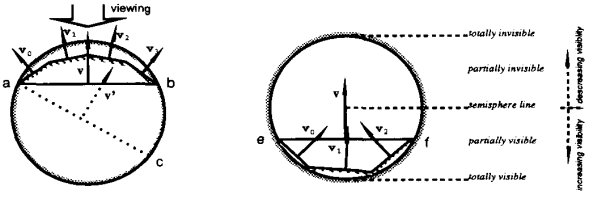
$$\begin{aligned}\sum_{j=0}^{m-1}\vec{\mathbf{V}}_i &= \sum_{j=0}^{m-1}\mathbf{n}_i R_i = \sum_{j=0}^{m-1}\mathbf{n}_i \iint_{S_i}(\mathbf{n}(s)\cdot\mathbf{n}_i)ds \\ &= \sum_{j=0}^{m-1}\mathbf{n}_i\left[\mathbf{n}_i\cdot\iint_{S_i}\mathbf{n}(s)ds\right] \\ &= \sum_{j=0}^{m-1}\left[\frac{1}{S_i}\iint_{S_i}\mathbf{n}(s)ds\right]\left[S_i(\mathbf{n}_i\cdot\mathbf{n}_i)\right] \\ &= \sum_{j=0}^{m-1}\left[\frac{S_i}{S_i}(\mathbf{n}_i\cdot\mathbf{n}_i)\iint_{S_i}\mathbf{n}(s)ds\right] \\ &= \sum_{j=0}^{m-1}\iint_{S_i}\mathbf{n}(s)ds = 0\end{aligned} \tag{9}$$

C. Estimation of unprocessed surfaces

The mass vector sum of a closed object model is a zero vector. If, during reconstruction, the total of all mass vectors of a building model is not zero but equals a vector $\vec{\mathbf{V}}_{gap}$, there must be some unprocessed surface patches whose mass vectors sum to be the negative of $\vec{\mathbf{V}}_{gap}$. Suppose the number of processed surface patches is m', then, from the relation $\sum_{j=0}^{m'-1}\vec{\mathbf{V}}_i + \sum_{j=m}^{m-1}\vec{\mathbf{V}}_i = 0$, it is easy to reach the conclusion.

$$\sum_{j=m'}^{m-1}\vec{\mathbf{V}}_i = -\sum_{j=0}^{m'-1}\vec{\mathbf{V}}_i = -\vec{\mathbf{V}}_{gap} \tag{10}$$

Defined to be the average normal $\mathbf{n}_i$, each mass vector $\vec{\mathbf{V}}_i$ is actually the average visible direction of that surface patch. Therefore, $\vec{\mathbf{V}}_{gap}$ provides an estimated direction from which these unprocessed surface patches could be observed.



(a) Convex Surfaces (b) Concave Surfaces

Fig. 4. View estimation.

This can be further explained by the two spheres in Fig. 4 for convex and concave surfaces, respectively. Since a convex object has a unique extended Gaussian image expression [10], a Gaussian sphere covering the object stands for its boundary condition, or a circle in a two-dimensional case. Suppose four surface patches are extracted (Fig. 4(a)); the total mass vector is $\mathbf{v} = \mathbf{v}_0 + \mathbf{v}_1 + \mathbf{v}_2 + \mathbf{v}_3$. It is obvious that the average visible direction of the unprocessed big arc from b to a is in the opposite direction of $\mathbf{v}$. Even if more surface patches are processed after several views, the negative of the updated total mass vector, which is $\mathbf{v}'$ in the figure, still points to the visible direction of the unprocessed portion.

A similar Gaussian sphere can also be used to examine the boundary of concave surfaces. In Fig. 4(b), a cave opens the circle at e and f. Concave surfaces usually do not form a closed mass vector chain unless a virtual surface patch is introduced to represent the cutting surface. Suppose there are m_h concave patches in a hole, the following relation exists,

$$\sum_{j=0}^{m_h-1}\vec{\mathbf{V}}_i + \vec{\mathbf{V}}_i(hole) = 0 \tag{11}$$

where $\vec{\mathbf{V}}_i(hole)$ is the mass vector of the virtual surface patch. As a result, an equation is obtained to estimate the visible direction of unprocessed concave surfaces.

$$\sum_{j=m_h'}^{m_h-1}\vec{\mathbf{V}}_i = -\left(\sum_{j=0}^{m_h'-1}\vec{\mathbf{V}}_i + \vec{\mathbf{V}}_i(hole)\right) = -\vec{\mathbf{V}}_{hgap} \tag{12}$$

This equation can also be applied to objects with more than one open hole where more virtual mass vectors, instead of one, are added to the equation to count in all cutting surfaces.

Nevertheless, for each group of connected concave surfaces, not all surfaces are visible in the estimated direction. Whether or not a particular concave surface is visible depends on the viewing direction and the shape of the hole. Shown in Fig. 4(b) are five levels of visibility. At the bottom is the line "totally visible" since no occlusion occurs in this case. When the cutting line moves up, surface visibility decreases. If the concave surfaces form a closed sphere, none of the surfaces are visible with normal sensory devices.

Between the two extremes there are two levels where concave surfaces can be either partially visible or partially invisible depending on the updated surface visible ranges given in (3). Targeting a surface under occlusion is a topic actively investigated in active vision [13], [14] and sensor planning [15], [16]. The difference in a modeling system is that the visible direction of a targeted surface is estimated, rather than known before hand. Whenever their occluded visibility is confirmed by (3) and visible direction is estimated with MVC, the surfaces can be made visible to the modeling system by using corresponding techniques.

D. Adjustment for concave surfaces

With the relation given in (10), the visible direction of unprocessed surfaces becomes predictable. Nevertheless, when concave surfaces are involved, a two-sphere method has to be used that combines (10) for convex surfaces and (12) for concave surfaces. As the two types of surfaces form the boundary of the object together, the virtual surface patch $\vec{\mathbf{V}}_i(hole)$ in (12) closes the boundary without the concave surfaces. After taking off the virtual surface from (10) and $\vec{\mathbf{V}}_i(hole)$ from (12), the total mass vector of the object becomes zero again.

$$\sum_{j=0}^{m-1}\vec{\mathbf{V}}_{vex} + \sum_{j=0}^{m_h-1}\vec{\mathbf{V}}_{cave} = 0 \tag{13}$$

Equation (13) predicates the unprocessed surfaces by checking the mass vector sum of the processed surfaces, the same function as provided by (10) and (12). Even though the estimated viewing direction reflects the group effect of both convex and concave surfaces, partially visible concave surfaces can be processed with sensor planning techniques when convex surfaces are finished. It is explained further in the following section where a simplified version of (12) is applied.

IV. View Estimation for Automatic Modeling

A 3D reconstruction system is able to automatically build up object models only when the system can estimate the direction of unprocessed surfaces by itself. With MVC, introduced in the previous sections, viewing directions of object surfaces can be predicated by simply checking the closure of an MVC. The following discussion uses experimental examples to describe how MVC is used to predict

the direction of unprocessed surfaces when the model of a sample object is under reconstruction. Details, including surface merging and partial model updating, are available in [12].

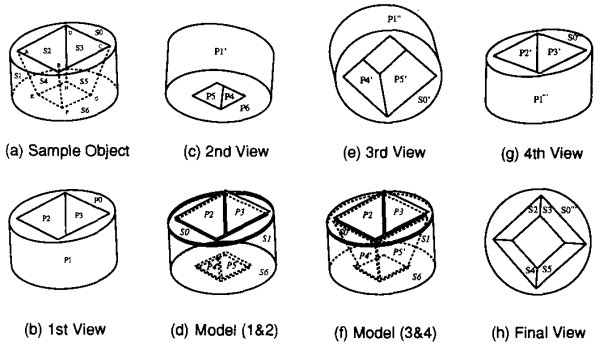


Fig. 5. Automatic modeling

The object in Fig. 5(a) is a cylinder cut by two planes and has a hole piercing through both its top and bottom surfaces. Its graphical surface model can be represented by seven boundary surfaces.

$$M_g = \{\text{S0, S1, S2, S3, S4, S5, S6}\} \tag{14}$$

The four polygons 'S2,' 'S3,' 'S4,' and 'S5' specify the hole, and they are sets of ordered vertices whose order indicates a polygon's orientation, e.g., S2 = $\langle V_A, V_E, V_H, V_D\rangle$. The eight vertices are given in coordinates.

$$\begin{aligned} V_{A-H} = \{&(-1.7,\ 0.879,\ 0.0),\ (0.0,\ 1.0\ 1.75),\\ &(1.798,\ 1.127,\ 0.0),\ (0.0,\ 1.0,\ -1.75),\\ &(-1.0,\ -1.0,\ 0.0),\ (0.0,\ -1.0,\ 1.0),\\ &(1.0,\ -1.0,\ 0.0),\ (0.0,\ -1.0,\ -1.0)\} \end{aligned}$$

The other three curve surfaces are specified below.

$$\text{S0} = \left\{(x, y, z)\middle| x^2 + z^2 = a^2, y = 1.0 + 0.071x\right\} - \text{S0}'$$

$$\text{S1} = \left\{(x, y, z)\middle| x^2 + z^2 = a^2, \text{S6} \le y \le \text{S0}\right\}$$

$$\text{S6} = \left\{(x, y, z)\middle| x^2 + z^2 = a^2, y = -1.0)\right\} - \text{S6}'$$

A. Closure check and view estimation

At a viewing direction not all of the seven boundary surfaces are visible. When a sensory device observes the object along a vector (–0.05, –0.5, –0.86), only four surface patches can be extracted to build up a partial model of the object (Fig. 5(b)). After repositioning the sensory device to the opposite direction, four more surface patches are extracted (Fig. 5(c)) from the second view. Since the newly obtained surface patch 'P1'' can be merged with 'P1' into the cylinder side 'S1,' the updated partial model M_{ptl2} has seven items as shown in Fig. 5(d).

$$M_{ptl2} = \{\text{S0, S1, P2, P3, P4, P5, S6}\} \tag{15}$$

From the two opposite views, all of the convex surfaces of the object are obtained. If the sample object has no hole, its three convex surfaces 'S0,' 'S1,' and 'S6' are all in 'M_{ptl2}.' Checking the mass vector chain, whose weights are the areas of one circle and two ellipses on the projection planes, the sum of mass vectors becomes zero.

$$\begin{aligned} \vec{V}_{chain} &= R_0\mathbf{n}_0 + R_1\mathbf{n}_1 + R_6\mathbf{n}_6 \\ &= 2.0\times 2.005\pi\mathbf{n}_0 + 2.0\times 0.28\pi\mathbf{n}_1 + 2.0^2\pi\mathbf{n}_6 \\ &= 0 \end{aligned} \tag{16}$$

However, the mass vector chain of 'M_{ptl2}' is not zero. Because of the hole, two cutting planes need to be deducted from the top and bottom surfaces respectively, and four more polygons need to be considered. The real mass vector chain of 'M_{ptl2}' is given below where all weights are polygon sizes after taking out $\vec{V}_{chain}$.

$$\begin{aligned} \vec{V}_{Mptl2} &= (|S0| - |S0'|)\mathbf{n}_0 + |S1|\mathbf{n}_1 + (|S6| - S6')\mathbf{n}_6 \\ &\quad + |P2|\mathbf{n}_2 + |P3|\mathbf{n}_3 + |P4|\mathbf{n}_4 + |P5|\mathbf{n}_5 \\ &= -|S0'|\mathbf{n}_0 + |P2|\mathbf{n}_2 + |P3|\mathbf{n}_3 + |P4|\mathbf{n}_4 \\ &\quad + |P5|\mathbf{n}_5 - |S6'|\mathbf{n}_6 \\ &= (0.414,\ -2.878,\ 0.0) \end{aligned}$$

Since the nonzero vector $\vec{V}_{Mptl2}$ predicates the orientation of the unprocessed surfaces, the third viewing direction should be opposite to $\vec{V}_{Mptl2}$, i.e. $\vec{V}_{3rd} = (-0.142, 0.989, 0.0)$ as a unit vector. In this direction, four surface patches 'S0',' 'P1'',' 'P4',' and 'P5'' are extracted in Fig. 5(e).

Again, the partial model is updated with the new information from the third view. Surface patches 'S0'' and 'P1''' are no longer in concern as they are either the same as '*S0*' or a part of '*S1*'. However, polygons 'P4'' and 'P5'' are merged with 'P4' and 'P5', respectively, since they overlap. The updated partial model of the third view is shown in Fig. 5(f) and described as M_{ptl3}.

$$M_{ptl3} = \{\text{S0, S1, P2, P3, P4', P5', S6}\} \tag{17}$$

B. View adjustment

Every time after a new view is processed, the reconstruction system checks the mass vector chain. The vector connecting the tail to the head of an unclosed mass vector chain is the next viewing direction. Unfortunately, this is not always true, especially when the direct viewing direction of the unprocessed surfaces is blocked by other surfaces. In this case, the viewing direction needs to be adjusted.

For the sample object, the direction of the fourth view is obtained from the mass vector chain of 'M_{ptl3}' as $\vec{V}_{4th}$ = (–0.319, – 0.218, 2.565). The extracted surface patches are shown in Fig. 5(g). Surfaces 'S0'' and 'P1''' are immediately discarded for the same reason explained before. Since newly extracted polygons 'P2'' and 'P3'' do not provide any new information because they are just inside '*P2*' and '*P3*,' the partial model after the fourth view is the same as 'M_{ptl3}.' The unchanged model leads to the unchanged fifth viewing direction, i.e., $\vec{V}_{5th} = \vec{V}_{4th}$. The succeeding viewpoints will be the same in the reconstruction procedure.

At this time, surface visibility should be examined to check surface relationship. By calculating the visible range $\mathbf{V}(S)$ from (2) and then checking with the updated visible range $\mathbf{V}'(S)$ out of (3) for the three convex boundary surfaces S0, S1, and S6, their identical ranges $\mathbf{V}(S)$ and $\mathbf{V}'(S)$ indicate their occlusion-free configuration. As was shown in (16) they form a closed chain when added with the top and bottom virtual cutting planes. Left in the mass vector chain are the four hole polygons P2, P3, P4', and P5', plus two virtual planes S0' and S6'.

When the same procedure is applied to the four polygons P2, P3, P4', and P5', their blocked visible ranges tell the system that the four polygons are the occluded surfaces that make the mass vector chain

unclosed. Further investigation on their connectivity with shared vertices confirms that they form a connected concave surface. Therefore, (12) can be applied to adjust the fifth viewing direction. The adjusted viewing direction is obtained as the group surface normal.

$$\vec{V}_{5th} = (-1.0)\cdot(n_2 + n_3 + n'_4 + n'_5)/4.0$$
$$= (-0.044,\ 0.866,\ 0.498)$$

Visible in the viewing direction are five surfaces 'S0''',' 'S2,' 'S3,' 'S4,' and 'S5' (Fig. 5(h)). After substituting '*P*2,' '*P*3,' '*P*4',' and '*P*5'' with their overlapped surfaces '*S*2,' '*S*3,' '*S*4,' and '*S*5,' the final model is acquired. Its zero mass vector chain indicates the closure of the model.

$$M_{fnl} = M_g = \{S0,\ S1,\ S2,\ S3,\ S4,\ S5,\ S6\} \qquad (18)$$

C. The Algorithm

An operation algorithm is shown in Fig. 6. As a boundary model, an object is defined by its surfaces. When an image is obtained from the input device, it is segmented to extract surface patches and obtain surface descriptions. These surface patches are then used to build up a partial model for the first view or, alternatively, they are merged with the previously processed surfaces to update the partial model.

Meanwhile, the mass vectors of the obtained surface patches of the partial model are added together to check the boundary closure of the model after the first two views. A connected MVC indicates the closure of the current object model, and the system continues to process other objects. Otherwise, the sensory device is repositioned to obtain the unprocessed surfaces whose orientation is suggested by the tail-to-head vector of the mass vector chain.

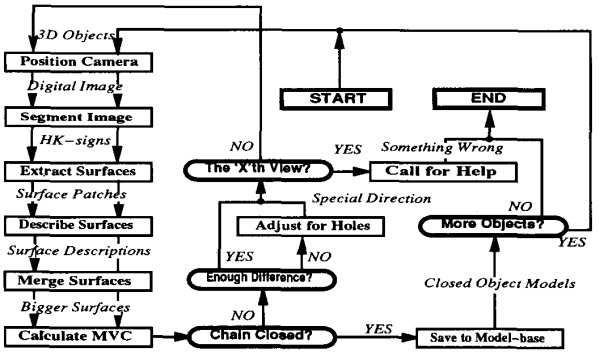


Fig. 6. The algorithm.

To avoid the dead loop caused by occlusion, every updated partial model is compared with the previous partial model to see how much new information is obtained. View adjustment for occluded surfaces is necessary when the two partial models have little difference. However, in practice, with complicated objects, an adjusted view may not work at all, such as for partially invisible surfaces. Therefore, a maximum number of views is set up as 'X' in Fig. 6. Help from the operator becomes inevitable when the limit is reached.

In addition, as with the total Gaussian mass, the MVC of an object is a necessary, but not a sufficient, condition to check the closure of object boundary. The mass vectors of the surfaces extracted in two opposite directions may happen to be closed even if the boundary is not yet complete. View adjustment is also necessary when an MVC is closed in one or two dimensions.

V. Conclusion

In this paper, an automatic modeling mechanism is introduced. Using the relationship of mass vector chains with the visibility and spatial closure of boundary surfaces, a reconstruction system is able to predicate the viewing direction of unprocessed surfaces and to make adjustment for occluded surfaces. A simple object is used to explain the fundamental operations of automatic modeling. More objects have been tested in [12]. Further research is being conducted to investigate its usage in practical applications such as automatic mechanical assembly.

Acknowledgment

This work is supported by NSERC of Canada under grant OGP0155411.

References

[1] V.S. Nalwa, *A Guided Tour of Computer Vision*, Addison Wesley, 1993.
[2] J. Aloimonos and C.M. Brown, "Robust computation of intrinsic images from multiple cues," *Advances in Computer Vision*, vol. 1, chap. 1.2, pp. 115–164, Lawrence Erlbaum Associates, Hillside, N.J., 1988.
[3] B. Vemuri and R. Malladi, "Constructing intrinsic parameters with active models for invariant surface reconstruction," *IEEE Trans. Pattern and Machine Intelligence*, vol. 15, no. 7, pp. 668–681, 1993.
[4] S. Zhang, G.D. Sullivan, and K.D. Baker, "The automatic construction of a view-independent relational model for 3D object recognition," *IEEE Trans. Pattern and Machine Intelligence*, vol. 15, no. 6, pp. 531–544, 1993.
[5] W.B. Green, *Digital Image Processing: A Systems Approach*, second edition, Van Nostrand Reinhold, New York, 1989.
[6] R. Jain and A.K. Jain, "Report: 1988 NSF range image understanding workshop," *Analysis and Interpretation of Range Images*, chap. 1, pp. 1–32. Springer-Verlag, New York, 1990.
[7] P.J. Besl and N.D. Mckay, "A method for registration of 3D shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
[8] J. Foley, A. van Dam, S. Feiner, and J. Hughes, *Computer Graphics: Principles and Practice*, second edition, Addison Wesley, 1992.
[9] H. Nishida and S. Mori, "An algebraic approach to automatic construction of structural models," *IEEE Trans. Pattern and Machine Intelligence*, vol. 15, no. 12, pp. 1298–1311, 1993.
[10] B.K. Horn, "Extended Gaussian images," *Proc. IEEE*, vol. 72, pp. 1656–1678, 1984.
[11] J.R. Kender, "The Gaussian sphere: A unifying representation of surface orientation," *Proc. DARPA Image Understanding Workshop*, pp. 157–160, Springer-Verlag, 1980.
[12] X. Yuan, *3D Reconstruction as an Automatic Modeling System*, Ph.D thesis, Dept. of Computing Science, Univ. of Alberta, Canada, 1992.
[13] D. Wilkes and J.K. Tsotsos, "Active object recognition," *IEEE CS Conf. CVPR*, pp. 136–141, 1992.
[14] "Special issue-active robot vision: Camera heads, model based navigation and reactive control," H.I. Christensen, K.W. Bowyer, and H. Bunke (guest eds.), *Int'l J. PRAI*, vol. 7, no. 1, Feb. 1993.
[15] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Trans. Pattern and Machine Intelligence*, vol. 15, no. 5, pp. 417–433, May 1993.
[16] K.N. Kutulakos and C.R. Dyer, "Recovering shape by purposive viewpoint adjustment," *Int'l J. Computer Vision*, vol. 12, nos. 2-3, pp. 331–338, 1994.