How to Decide from the First View Where to Look Next

MS-CIS-90-39
GRASP LAB 219

Jasna Maver
Ruzena Bajcsy

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104-6389

July 1990

ACKNOWLEDGEMENTS:
This research was supported in part by AFOSR Grants 88-0244, AFOSR 88-0296; Army/DAAL 03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719, IRI 89-06770; DARPA Grant N0014-88-0630 and DuPont Corporation.

How to decide from the first view where to look next

Jasna Maver and **Ruzena Bajcsy**

GRASP Laboratory
Department of Computer and Information Science,
University of Pennsylvania, Philadelphia, PA 19104

Abstract

The task of constructing a volumetric description of a scene from a single image is an underdetermined problem, whether it is a range or an intensity image. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurements from several different views. We have limited ourselves to the range images obtained by a laser scanning system. It is an active binocular system which can encounter two types of occlusions. An occlusion arises either when the reflected laser light does not reach the camera or when the directed laser light does not reach the scene surface. The task of 3-D data acquisition is divided into two subproblems: to see what is illuminated and to properly direct the illuminating plane to illuminate the entire scene. The first kind of occlusions are easily detected and can be used in designing an efficient algorithm. We develop a strategy to determine the sequence of different views using the information in a narrow zone around the occluded regions. Occluded regions are approximated by polygons. Based on the height information of the border of the occluded regions and geometry of the edges of the polygonal approximation, the next views in the same scanning plane are determined. From the acquired information in the first scanning plane the directions of the next scanning planes are computed.

1 Introduction

The task of constructing a volumetric description of a scene from a single image is an underdetermined problem, whether it is a range or an intensity image. To resolve the ambiguities that are caused by occlusions in images, we need to take sensor measurments from several different views. The occlusions ariese because of geometrical complexsity of the objects, because the objects are opaque and, camera's visibility cone is typically less than 180°.

The problem of acquisition of multiple views, "where to look next", can be decomposed into several sub-problems. Developing the strategy for choosing the next viewing directions is perhaps the most challenging one. Another problem is: how to merge information from different views. This problem is closely related to object representation. We can merge the 3-D information obtained from each view into a common coordinate frame. When the complete spatial map is obtained it can be modeled by volumetric primitives such as superquadrics [Solina and Bajcsy, 1990]. Another approach is to find some topological features (vertices, edges, faces, etc.) which represent the basic elements for building the model of a scene. Based on incomplete information of different views the structure of relations between these topological features may be constructed. An important problem which arises in merging 3-D information from different views is how to deal with the errors which appear in measurements and transformation parameters of applied transformations [Krotkov and Kories, 1988].

We limit ourselves to planning the next view. Some work has already been done on the planning of visual sensors [Sakane *et al.*, 1987a, Sakane *et al.*, 1987b]. Here the plan generation is done on the bases of geometrical and physical knowledge of the environment and models. This approach is model-based. N. Ahuja and J. Veenstra [1989] construct a 3-D model from the orthographic projections of an object onto a plane perpendicular to a viewing direction. The thirteen views must be selected from any subset of directions corresponding to the three "face" views, six "edge" views, and four "corner" views of an upright cube.

In our work we use range data. The next view planning is data driven. The first range image can be obtained from any direction in the scanning plane of our sensor system. All the next views are defined from the incomplete information of the previous views, and we also take into account the properties of our sensor system. We made experiments on real data.

The paper is organized as follows. In section 2 we describe our sensor system. The task of 3-D data acquisition is divided into two subproblems: to see what is illuminated and how to direct the light to illuminate the entire scene. The problem definition of the first subproblem and a proposal to its solution is described in section 4. The second subproblem is represented in section 5. Experimental results are presented in section 6.

2 Sensing

The depth of the scene is measured by the range scanner system [Tsikos, 1989]. It is an active binocular system consisting of a laser, a CCD camera, and a turntable which supports the scanned object. The laser and the camera are coupled. The laser produces a beam which is spread into an illuminating plane. The illuminating plane intersects the object surface, forming a planar curve (laser—stripe). Each point on the curve is mapped onto a single point in the camera image plane.

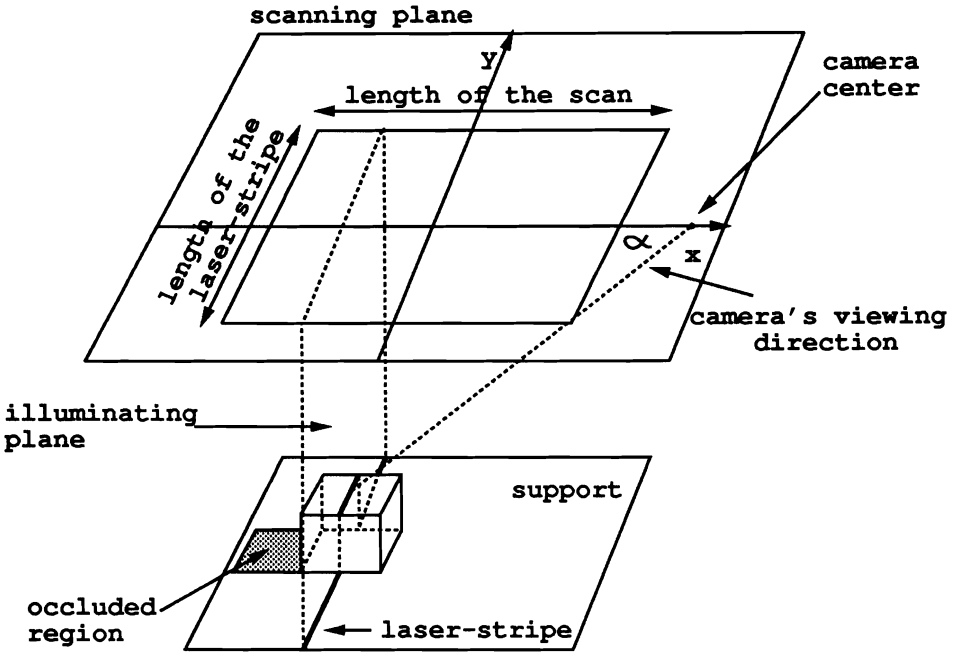

Figure 1: Sensing

The distance of the point on the curve to the camera center is determined by the intersection of the illuminating plane and the line which goes through the camera center and the point on the image plane. The range image is obtained by scanning the object in a series of parallel illuminating planes. This is achieved by moving the scene support with constant velocity perpendicular to the illuminating plane. The y-axis on the image corresponds to the laser–stripe and the x-axis corresponds to the shift of the support. During the scanning, the laser sweeps out a plane, called the *scanning plane* (see fig. 1). The distance measurements are transformed into the perpendicular distance between the scanning plane and the surface of the objects and are stored as intensity values in the final range image. The higher values correspond to smaller distances, the lower to greater distances.

3 Two step strategy

The described sensor system can encounter two types of occlusions. An occlusion arises either when the reflected laser light does not reach the camera or when the directed laser light does not reach the surface. The task of 3-D data acquisition is divided into two subproblems: to see what is illuminated and to properly direct the illuminating plane to illuminate the entire scene.

First step

The system is calibrated so that the depth of the support plane has the lowest value, which is greater than zero. This enables us to detect the first kind of *occluded regions* whose intensity value is zero. Due to the occluded regions the range image represents incomplete $2\frac{1}{2}$-D data of the scene. We can construct a complete $2\frac{1}{2}$-D data of the scene by scanning from different directions in the same scanning plane. For every pair (x, y) in the scanning plane, we get the distance of the closest point on the surface to the scanning plane (fig. 2),

$$h(x, y) = \max(h_{on\ surface}(x, y)). \tag{1}$$

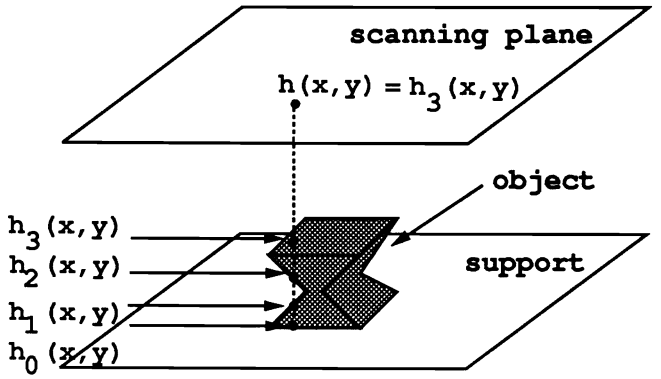

Figure 2: Height at the point (x,y) is defined by the closest point on the surface to the scanning plane

We generate complete $2\frac{1}{2}$-D data of the scene from a union of all those views that together eliminate the occluded regions. In each scan-view the illuminating plane illuminates the same part of the scene surface but camera sees its different parts.

Second step

From the acquired $2\frac{1}{2}$-D data in the first scanning plane we compute the directions of the next scanning planes for 3-D data acquisition.

4 Getting the complete $2\frac{1}{2}$-D data of the scene

In all occluded regions the information about the height of the surface with respect to the scanning plane is unknown. The question is how to define the next scanning direction to acquire the missing information.

Definition of the scanning direction. Let the laser–stripe and the camera move from left to the right where the camera is to the right of the laser–stripe. The *scanning direction* is the angle of counterclockwise rotation of the camera–laser configuration about the origin of the scanning plane. However, in our experiments we rotated the support in a clockwise direction instead.

4.1 Problem definition

For a given point $p_i(x, y)$ in the scanning plane, we want to find out all the directions from which $p_i(x, y)$ is visible.

In order to determine whether a given point $p_i(x, y)$ is visible from the direction φ_j, we must check if any of the points $p_k(x, y)$ in the scene along this direction (fig. 3) occludes the point $p_i(x, y)$. We consider the relation between the distance l of two points in the scanning plane and the length of occlusion l' produced by $p_k(x, y)$ in direction φ_j and defined as:

$$l' = \frac{h(p_k) - h(p_i)}{\tan(\alpha(h(p_k)))}, \tag{2}$$

where $h(p_i)$ and $h(p_k)$ represent the height of the scene at points $p_i(x, y)$ and $p_k(x, y)$, respectively.

If the distance between $p_i(x, y)$ and each $p_k(x, y)$ is greater than the computed length of occlusion l', then $p_i(x, y)$ is visible from direction φ_j. In the same scanning plane the point $p_i(x, y)$ can be seen from several different directions $\varphi_j, \varphi_{j+1}, \ldots$. To compute the set of all directions from which $p_i(x, y)$ is visible, we have to know the corresponding height for each point in the scanning plane.

As indicated previously, after the first scan, the height of the points that belong to the occluded regions is unknown. To determine the next scanning direction we want to compute the visible directions for all the points in the occluded regions. It follows that we must make some assumptions about the height of these points. The question is: how restrictive assumption about the height of the point in the occluded region can be made. If the assumption is too restrictive then the solution may not be found at all, while if the assumption is too weak then it may give the direction from which the point is not visible.

4.2 Occluded region representation

We approximate the occluded regions by polygons. The subsequent views are computed from the analysis of these polygons and from the height values of their borders.

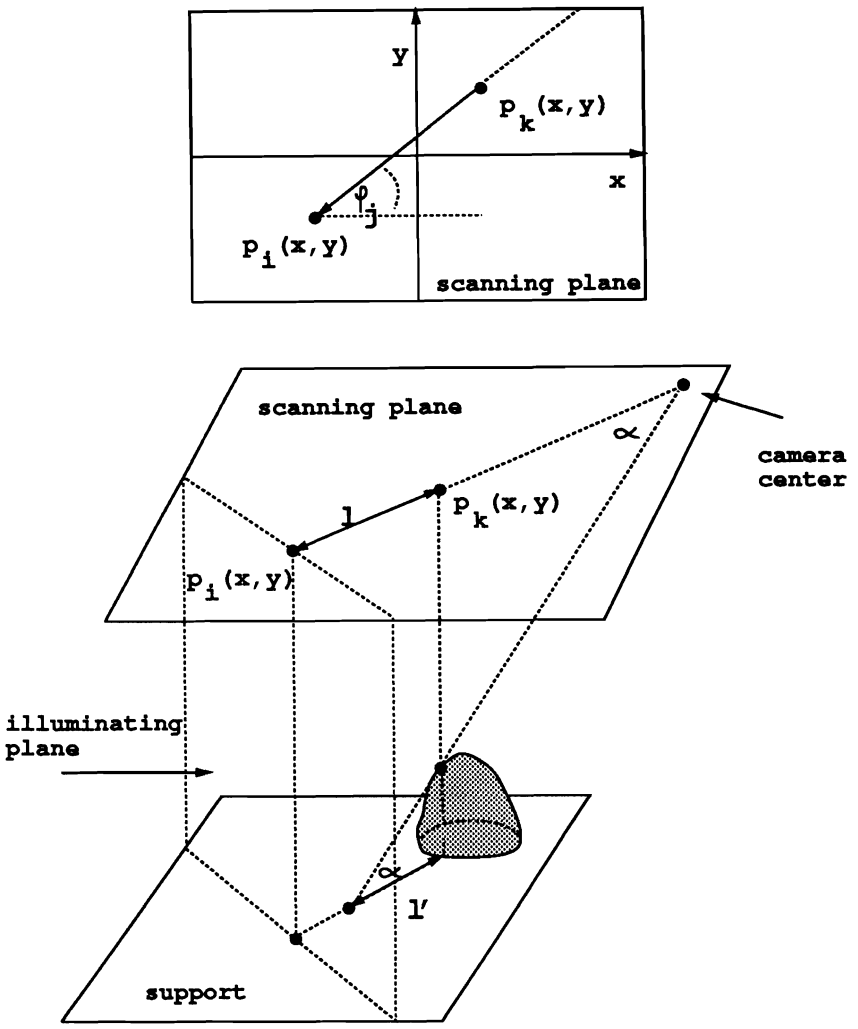

Figure 3: The relation of the distance l between points $p_i(x,y)$ and $p_k(x,y)$ and the length of occlusion l'

The contours of the occluded regions in the image are found by Pavlidis' algorithm TRACER [Pavlidis, 1982]. The contours of the occluded areas are represented as contour descriptors $x(s)$ and $y(s)$ for each area separately. Each contour must be segmented into a series of straight lines. The breakpoints on the contour are points of maximum curvature.

Points of maximum curvature

In order to find the points of maximum curvature, the contour descriptors $x(s)$ and $y(s)$ are first smoothed to filter out noise. The derivatives $\frac{dx(s)}{ds}$ and $\frac{dy(s)}{ds}$ are computed to get the tangent angle function as

$$\phi(s) = \tan^{-1}(\frac{dy/ds}{dx/ds}). \tag{3}$$

Special processing is required to handle phase wrapping. The breakpoints of the contour are

obtained from curvature function which is the derivative of the tangent angle function

$$\kappa(s) = \frac{d\phi(s)}{ds}. \tag{4}$$

The positive maxima and negative minima of the curvature function $\kappa(s)$ represent the convex and concave vertices of the polygon.

Border height definition

The height of the contour $h(s)$ is found by searching in a narrow zone around the occluded area. For each pixel on the contour the search is done for n pixels in three directions. The directions of the search are defined by the change in the x and y coordinates of the new pixel on the contour (fig. 4). The heighest value found in the search is chosen as the pixel height. The height of the border between two vertices is approximated by a constant. It is defined as the median of all height values on the contour between the two vertices.

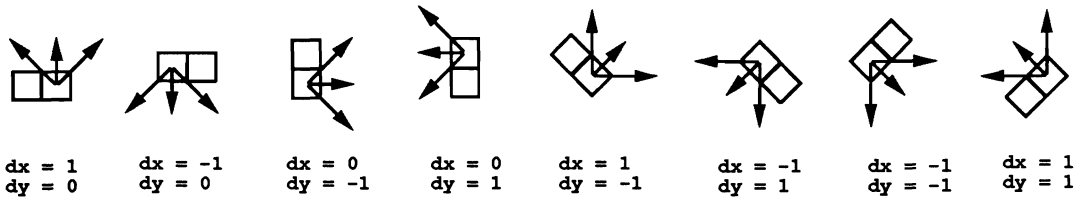

Figure 4: Searching directions for the pixel height.

Polygon representation

The points of maximum curvature $p_i(x, y)$ with $i = 1, \ldots, n$ are the vertices of a polygon $P(p_1, p_2, \ldots, p_n)$ which approximates the occluded region. The vertices are ordered clockwise so that when one moves in the direction of ordered vertices an occluded region is on the right side of an edge. For each edge $e_i(p_i, p_{i+1})$ in polygon P its angle (fig. 5) with respect to the common coordinate frame is defined by the equation

$$\theta(e_i) = \tan^{-1}\left(\frac{y_i - y_{i+1}}{x_i - x_{i+1}}\right) \tag{5}$$

where (x_i, y_i) and (x_{i+1}, y_{i+1}) are the coordinates of the vertices p_i and p_{i+1} respectively.

4.3 Analysis of occluded regions

Two properties are added to each edge according to its angle $\theta(e_i)$ in the common coordinate frame and to its height $h(e_i)$ (fig. 7).

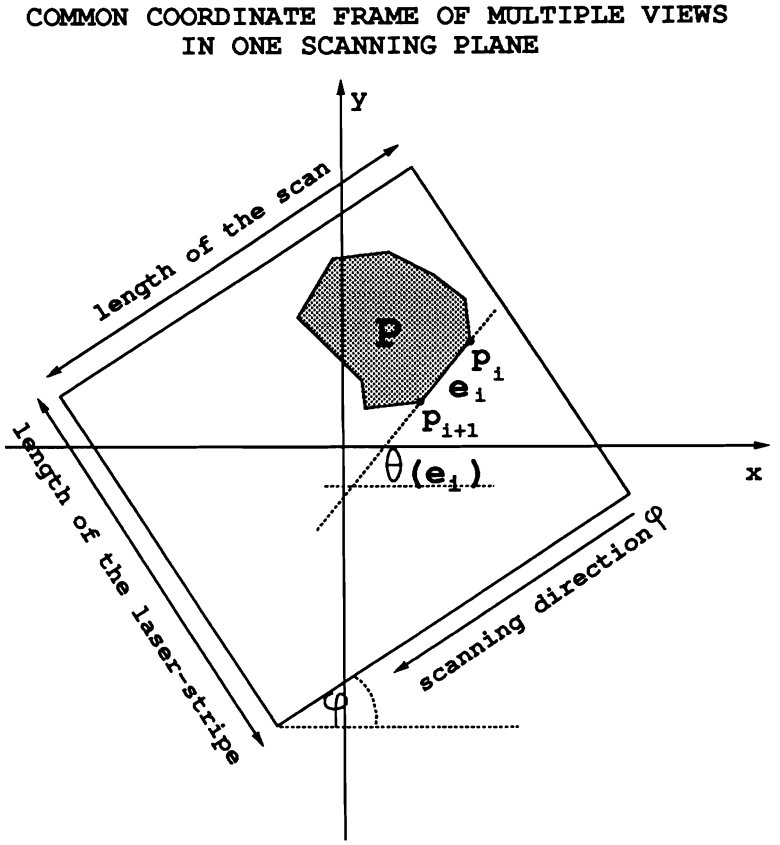

Figure 5: Angle θ of the edge e_i with respect to the common coordinate frame.

1. **Occlusion:** If φ is the scanning direction by which the image was obtained then an edge $e_i(p_i, p_{i+1})$ with angle $\theta(e_i) \in (\varphi, \varphi + 180°)_{counterclockwise}$ is called an *occluding edge* e_{occ}[i] and belongs to the set of occluding edges, O.

   If in polygon P we draw lines through the endpoints of the occluding edges in the direction φ, we cut the polygon P into areas A, as shown in figure 6. To each edge e_{occ}[i] in the polygon P belongs such an area A_i. Polygon P can then be defined in the following way

$$P(p_1, p_2, ..., p_n) = \bigcup_{\{i | e_{occ}[i] \in O\}} A_i, \tag{6}$$

   Note that only occluding edges create areas A.

2. **Activity:** The second property is derived from the height of the border. Let us say that $e_{occ}[i]$ is an occluding edge to which the area A_i belongs, and e_j is an edge which limits the same area A_i. We compare the length of occlusion l' on the range image caused by the occluding edge $e_{occ}[i]$ with the occlusion caused by the difference in heights $(h(e_{occ}[i]) - h(e_j))$ for $\alpha(h(e_{occ}[i]))$. For each area A_i, the edges which limit the area are tested as follows:

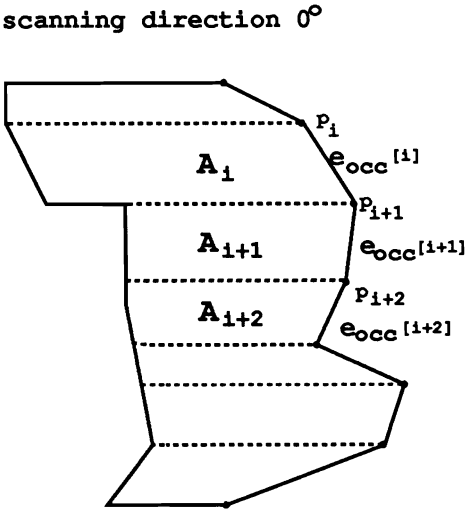

Figure 6: Areas A belonging to occluding edges.

If e_j limits A_i then

$$e_j = \begin{cases} inactive & l' \leq \frac{h(e_{occ}[i]) - h(e_j)}{\tan(\alpha(h(e_{occ}[i])))} \\ \\ active & l' > \frac{h(e_{occ}[i]) - h(e_j)}{\tan(\alpha(h(e_{occ}[i])))} \end{cases} \tag{7}$$

where j=1,...,n and l' is measured at the middle point of the edge e_j. An occluding edge is also an active edge. Active edges are these edges of polygon P which are able to occlude the polygon P.

From these properties, we say that the pixels in occluded regions can be visible only from inactive edge. This statement implicitly includes two assumptions:

1. Changes in surface heigth inside the occluded regions are so small that they cannot cause occlusions.

2. Changes in surface heigth outside the border of the occluded regions are so small that they cannot cause occlusions.

Computing visible directions for each black pixel in the image is computationally expensive. Instead of computing visible directions for each pixel, we compute the visible directions for areas A.

For each area A_i in polygon P, we shall define the *viewing angle* $\Phi(A_i)$ as the sector containing all directions from which the area A_i can be seen. The viewing angle is computed in two steps:

1. Initial viewing angles $\Phi(A_i)_{initial}$ are set up.

2. The viewing angles are narrowed by each active edge.

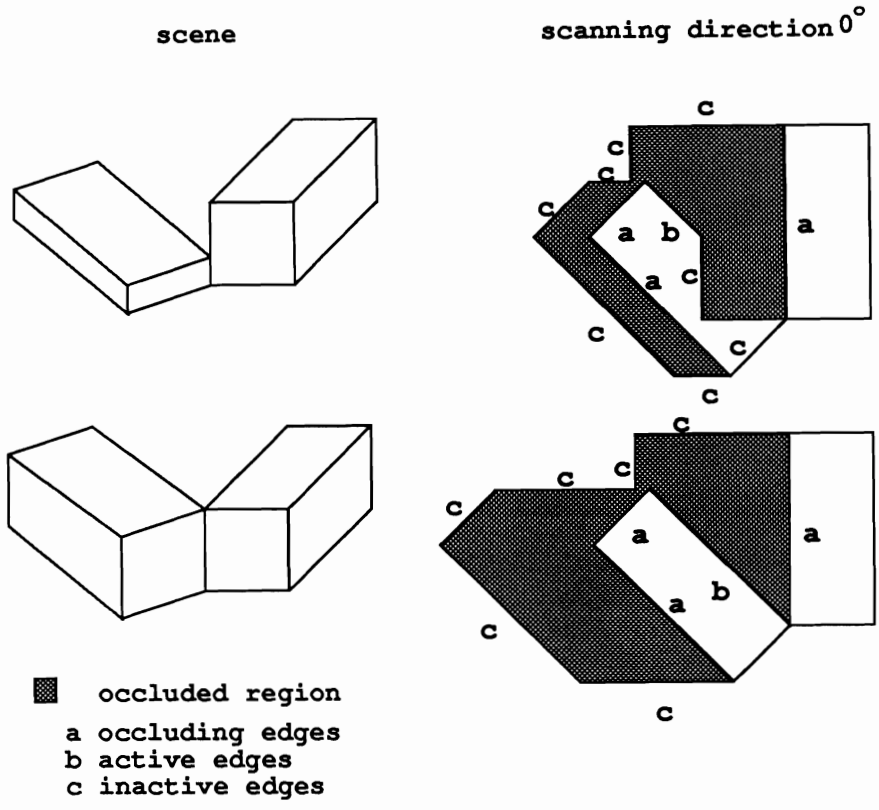

Figure 7: Definition of edges.

The initial viewing angles definition

The initial viewing angle $\Phi(A_i)_{initial}$ for each area A_i is defined by the angle of the occluding edge e_{occ}[i] the area A_i belongs to:

$$\Phi(A_i)_{initial} = [\varphi(e_{occ}[\mathrm{i}]), \varphi(e_{occ}[\mathrm{i}]) + 180°]_{counterclockwise} \tag{8}$$

Viewing angles modification.

To each initial viewing angle $\Phi(A_i)_{initial}$ belongs an area which is to the right of the line defined by the occluding edge e_{occ}[i] (fig. 8a). This area can be divided into three parts, an area above A_i, an area in which A_i lies and an area below A_i. Only the edges within these three areas modify the angle:

- Inactive edges do not modify the angle.
- Active edges can modify the initial viewing angle $\Phi(A_i)_{initial}$ in the following way.
  - If the edges $e_l, \ldots, e_r$ are active and limit the area A_i, the viewing angle (fig. 8c) is

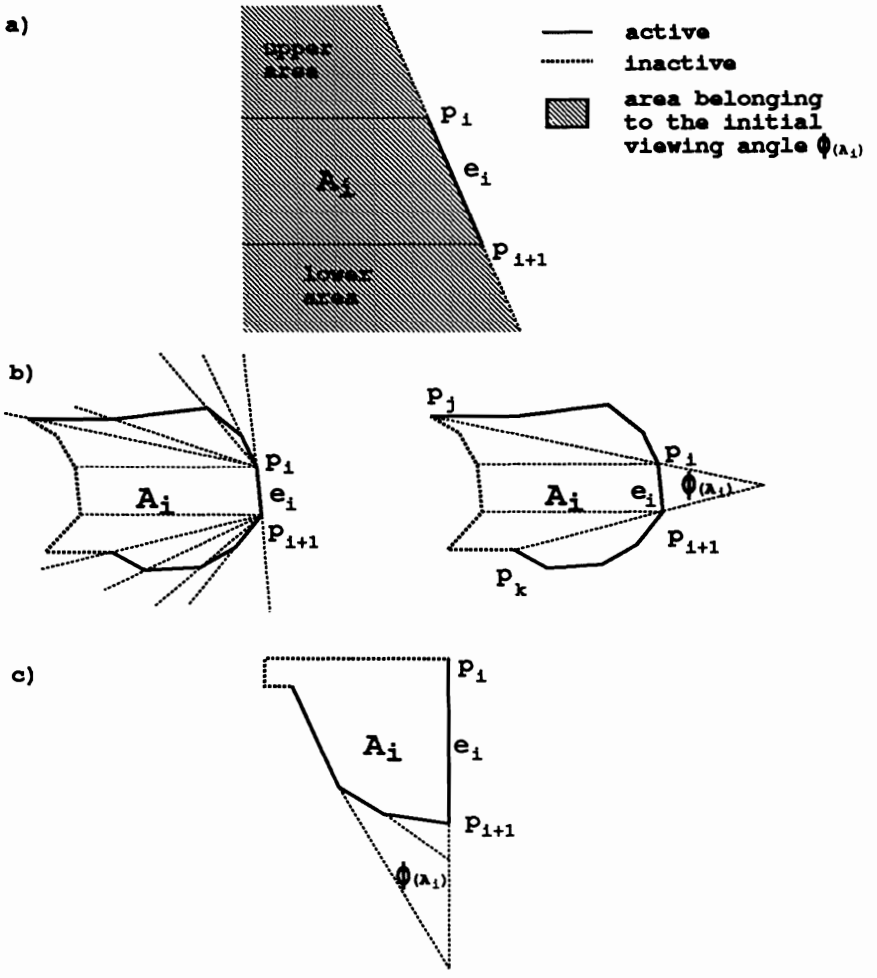

Figure 8: Narrowing the angle $\Phi(A_i)$

defined as

$$\Phi(A_i) = \Phi(A_i)_{initial} \cap [\theta(e_l), \theta(e_l) + 180°] \cap$$

$$\cap \ldots \cap [\theta(e_r), \theta(e_r) + 180°]. \tag{9}$$

- If an edge e_j is active and is not a part of the border of the area A_i then the angle $\Phi(A_i)$ is narrowed so that e_j is excluded from the area of the viewing angle. The narrowing is done through the endpoints of the active edges on both sides of the area A_i (fig. 8b). The new viewing angle for area A_i is then:

$$\Phi(A_i) = [\varphi_1, \varphi_2]_{counterclockwise}. \tag{10}$$

φ_1 and φ_2 are the lower and upper bound of the viewing angle, defined as

$$\varphi_1 = \tan^{-1}(\frac{y_j - y_i}{x_j - x_i}), \tag{11}$$

$$\varphi_2 = \tan^{-1}(\frac{y_k - y_{i+1}}{x_k - x_{i+1}}). \tag{12}$$

(x_j, y_j), (x_k, y_k) are the coordinates of the vertices $p_j(x,y)$ and $p_k(x,y)$ in the common coordinate frame. Vertex $p_j(x,y)$ is that vertex of the upper area which narrows the viewing angle of the area A_i the most from above. Similarly, the vertex $p_k(x,y)$ narrows the viewing angle the most from below.

Whenever the viewing angle $\Phi(A_i)$ becomes empty the area A_i can not be seen from one view. In this example the area A_i must be split on smaller parts and new viewing angles must be computed.

4.4 Determination of the next scanning direction

After having defined the viewing angles, we must determine the next scanning direction. If there exists a direction from which the whole occluded region can be seen at once then this direction must appear in all viewing angles. We have to intersect the viewing angles and choose a scanning

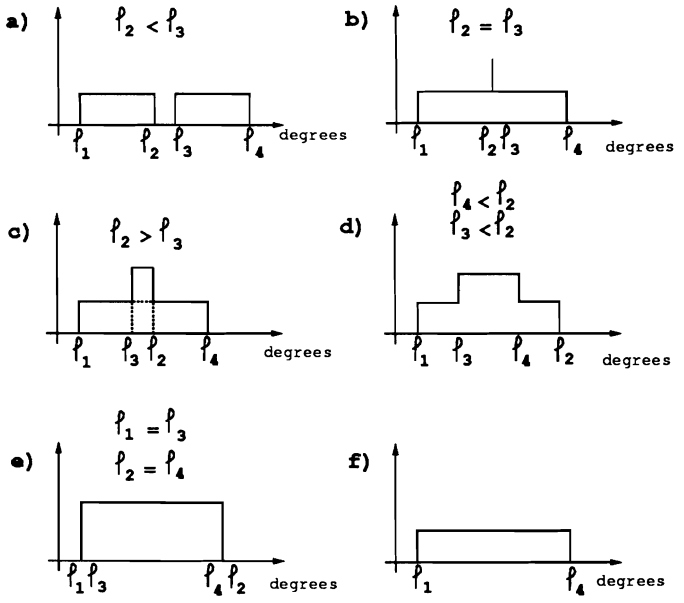

Figure 9: a)-e) possible histograms of two viewing angles, f) impossible histogram

direction from the global intersection if it is nonempty or from more partial intersections otherwise. To select the scanning directions from the right partial intersections we build a histogram which shows in how many viewing angles a certain direction appears. The histogram can be constructed by successionally adding viewing angles and incrementing the count of all directions in added viewing angle. Since the viewing angles are closed intervals, two viewing angles, $[\varphi_1, \varphi_2]_{ccw}$ and $[\varphi_3, \varphi_4]_{ccw}$ where $\varphi_1 \leq \varphi_3$, can form the histograms like in (fig. 9 case a,b,c,d,e). The case (fig. 9f)

is not possible. Whenever we include a viewing angle into the histogram, there exist the following possibilities:

1. The viewing angle forms a new maximum (Fig. 10a).
2. The viewing angle leaves the width of an old maximum unchanged from the left and from the right side (Fig. 10b).
3. The viewing angle modifies an old maximum from
   - the right side (Fig. 10c),
   - the left side (Fig. 10d),
   - both sides (Fig. 10e).

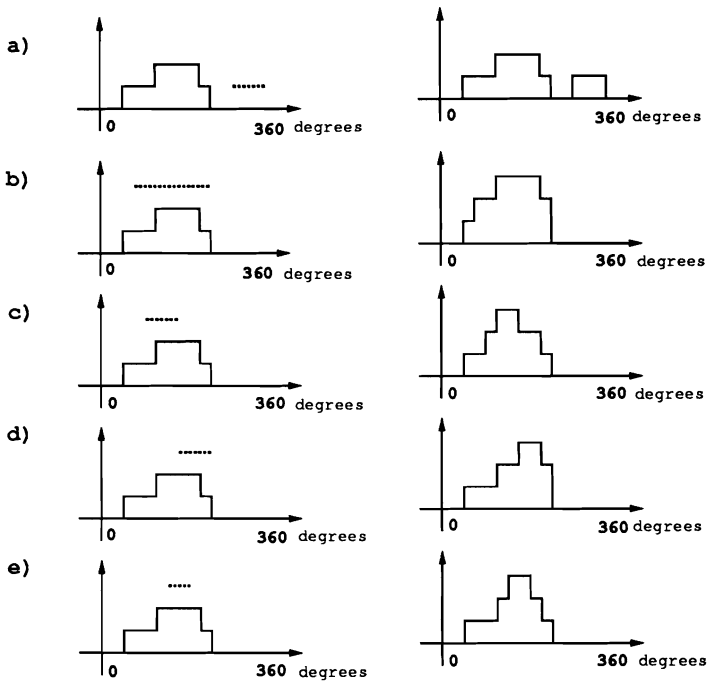

Figure 10: Including a new viewing angle into the histogram

In the third case above we decrease the maximum's width but increase its height by one which means that maximum remains within all previous viewing angles. In each viewing angle in the histogram there is at least one maximum. If there is only one maximum in the histogram, it represents the global nonempty intersection. From each direction in the global maximum the whole occluded region can be seen at once. If there is more than one maximum in the histogram and we select one scanning direction from each maximum, then we are able to see the whole occluded region.

Histogram decomposition

Some viewing angles can include more than one maximum which means that some areas A will be seen more than once if we take a scanning direction from every maximum. We must test if views from *all* maxima are really necessary. For each viewing angle we must find the number of maxima it includes. If in viewing angle $\Phi(A_i)$ there exists only one maximum then the area A_i can be seen only from this maximum, so the scanning direction from this maximum is necessary. We label all such maxima and remove from the histogram all viewing angles which include at least one labeled maximum. From the remaining viewing angles we construct a new histogram and repeat the procedure until all viewing angles are removed. Then the new scanning directions must be selected from all labeled maxima.

4.5 Combining images from different viewing positions into one coordinate frame

The images obtained from the previously selected scanning directions are combined into one coordinate frame where the largest pixel value is kept as the pixel height.

The coordinates (x_i, y_i) of the pixels in the range image obtained by the scanning direction φ are transformed into the coordinates (x_c, y_c) of the common coordinate frame by the following transformation:

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \tag{13}$$

Because of imprecise sensing and inaccuracy of the transformation parameters the final merging is achieved by correlation;

$$\begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} + \begin{bmatrix} \cos(\varphi + \Delta\varphi) & -\sin(\varphi + \Delta\varphi) \\ \sin(\varphi + \Delta\varphi) & \cos(\varphi + \Delta\varphi) \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \tag{14}$$

where Δx, Δy, and $\Delta\varphi$ are discrete values from the intervals $[-X, X]$, $[-Y, Y]$, and $[-\phi, \phi]$, respectively, which give the maximum correlation between the image of the common coordinate frame and the current transformed image.

The height at pixel (x_c, y_c) is the maximal height of all pixels which are transformed into it.

4.6 Completion of the analysis of one plane

To compute the new scanning directions from the first view we assume that the changes in surface height in the occluded regions are too small to cause occlusions. The same is assumed for the regions

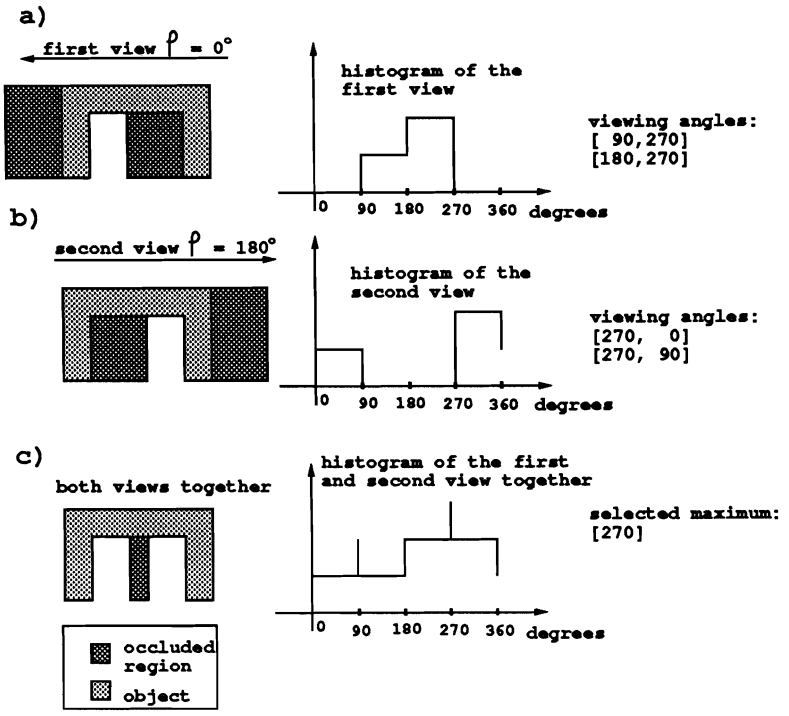

Figure 11: Example 1

outside the border of the occluded regions. If this is not the case, and after combining images from different viewing positions into one coordinate frame, there still exist occluded regions, they should be explored from other directions. Since we use only the height information of the border of occluded regions to compute the viewing angles, we may end up with too wide viewing angles. When we compute the next scanning direction it is important to use the information about the active borders obtained by the previous scans. In our algorithm, we incorporate this information by building a new histogram from the previous one. Let us illustrate this by two examples.

Example 1

The scene consists of an U shaped object with a constant height. In the image of the first scan we find two occluded regions (fig. 11). From their borders the viewing angles are computed. The viewing angles are $[90°, 270°]$ for the larger region and $[180°, 270°]$ for the smaller region. The viewing angle of the smaller region is too wide, because we do not check the visible region outside the border of the occluded regions. In the histogram we get one maximum at $[180°, 270°]$. If we select $180°$ as the second scanning direction we still have an invisible part after merging both images. The maximum of the histogram of the second view is again not restrictive enough. By combining both histograms, we get two maxima. Since all viewing angles include the maximum $270°$ that one is selected as the next scanning direction.

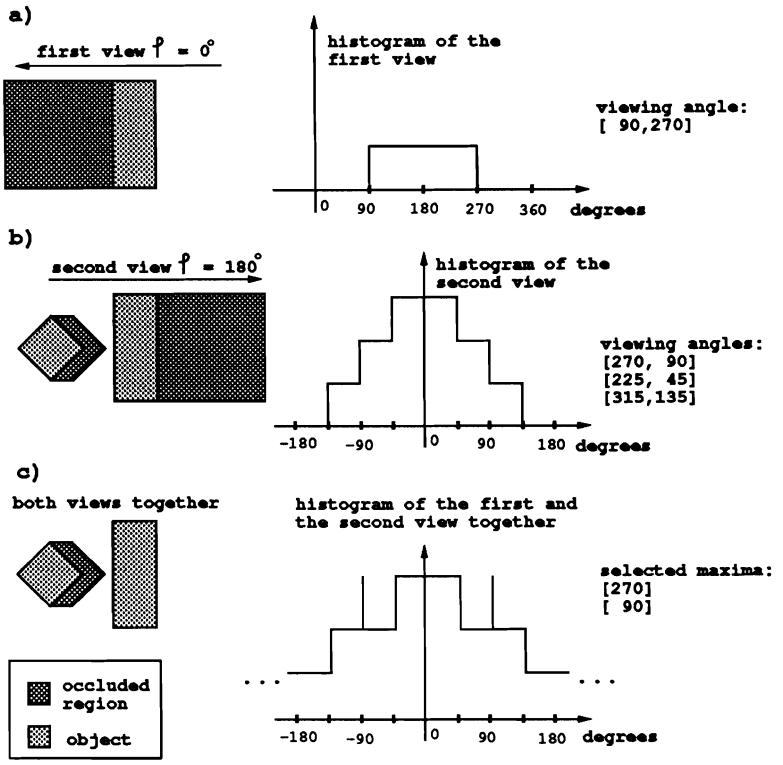

Figure 12: Example 2

Example 2

The second scene consists of two boxes different heights. In the image of the first view (fig. 12) we find one occluded region. The computed viewing angle is $[90°, 270°]$. After the second view we detect a low box inside the occlusion of the first scan. If in the second view the occluded region of that box does not reach the larger box, the histogram of the second view does not give the correct solution. By incorporating the first histogram into the second histogram, we get three maxima. We do not select a scanning direction from the middle maximum $[-45°, 45°]$ because it includes the scanning direction $\varphi = 0°$ of the first scan.

In general the computation of the next view is completed in the following way. For each scanning direction that was taken we compute corresponding viewing angles. A single histogram is then constructed from both the old one and new viewing angles. New scanning directions are computed in a similar way as in subsection 3.1.4. During the histogram decomposition we do not remove the viewing angles of the old histogram. We do not select the scanning direction from those maxima which include the scanning directions already taken. A possibility exists where there is no viewing angle which includes only one maximum. In this case, a histogram decomposition must be done for each local maximum. The solution with the minimum number of necessary scanning directions is selected. The procedure stops either when all occluded regions are removed or when

from all maxima scanning direction have been taken.

5 Searching for the next scanning plane.

In the previous section we described how to rotate the sensor system in one plane to get the complete $2\frac{1}{2}$-D data of the scene. In each scan the illuminating plane illuminates the same part

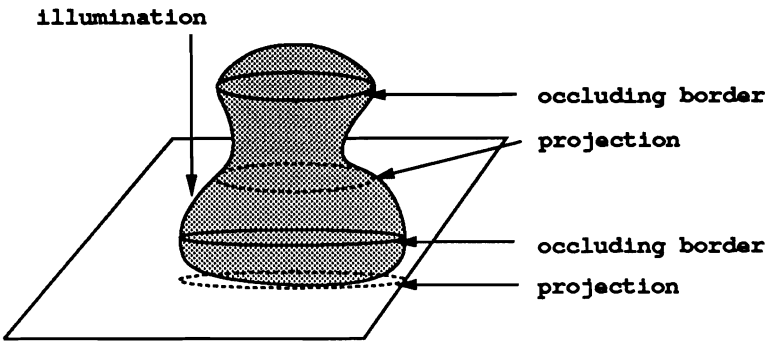

Figure 13: Occluding borders, and their projections

of the scene surface but camera sees its different parts. The next problem is how to orient the illuminating plane to illuminate the entire surface. We would like to solve this problem on the base of the $2\frac{1}{2}$-D information we already have.

The points on the surface to which the illumination is tangent (fig. 13) form borders which separate illuminated from non-illuminated surfaces. These borders are *occluding borders.* An occluding border makes a projection on the other parts of the surface in direction of illumination. The surfaces, which are spread between occluding borders and their projections, must be illuminated.

Assume that the scanning plane in which we constructed the $2\frac{1}{2}$-D data is the upper basic face of a cylinder (fig. 14). If the new scanning plane can only be perpendicular to the first one, it can be defined by an angle δ, which is the angle of rotation of the illuminating plane about the origin of the upper basic face of the cylinder. From the occluding borders we would like to compute the rotation angles δ of the next scanning planes.

5.1 Occluding borders

Occluding borders are defined by the jumps in height in the $2\frac{1}{2}$-D image of the first scanning plane. With an edge operator which is sensitive to C0 discontinuities, we can locate occluding borders. Occluding borders can be approximated by piecewise linear segments l_i with $i = 1, \ldots, m$. In our experiments we use the algorithm proposed by R. Nevatia and K. Ramesh Babu [1980]. In their algorithm, line finding consists of the following steps: determining edge magnitude and direction

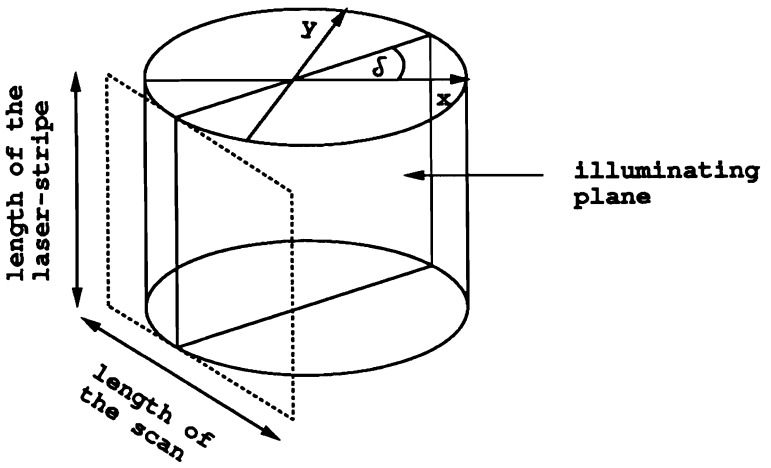

Figure 14: Orientation of the illuminating plane defined by angle δ

by convolution of an image with a number of edge masks, thining and thresholding these edge magnitudes, linking the edge elements based on proximity and orientation, and approximating the linked elements by piecewise linear segments.

Each line segment l_i represents, at the same time, the part of occluding border and its projection. To compute the next scanning plane we will assume that the surface which is spread between the occluding line segment l_i and its projection, is a flat face F_i. In a similar way, as we did in the previous section, we would like to define for each line segment l_i the illuminating angle $\Psi(F_i)$ as a set of all directions from which the face F_i can be illuminated.

5.1.1 The height of occluding border and the height of its projection

The height of occluding border $h_b(l_i)$ and the height of its projection $h_p(l_i)$ are found by searching in a narrow zone left and right of the linked edge elements. The direction of the search is defined by the change in x and y coordinates of the new edge element in the link. The directions of the search on the left side are the same as those in figure 4 where the searching directions on the right side are just opposite. For each edge element, we calculate a left height taking the median of the height values found in the search on the left. Similarly, we do the same on the right for each edge element. The left and right height of the line segment l_i are approximated by constants which are defined as medians of all height values, between break points p_i and p_{i+1}, which are found on the left and on the right side from the edge elements in the link, respectively. The higher value corresponds to the height of occluding border $h_b(l_i)$ and the lower to the height of its projection $h_p(l_i)$.

5.1.2 An angle of a line segment

The endpoints p_i, p_{i+1} of the line segment l_i are ordered such that the higher surface is always on the right side going in direction from p_i to p_{i+1}. The angle of the line segment l_i in the common coordinate frame of the first scanning plane is

$$\psi(l_i) = \tan^{-1}(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}) \tag{15}$$

Since the higher surface is on the right side of the line segment l_i the face F_i can be illuminated only from directions in the range $[\psi(l_i), \psi(l_i) + 180°]_{ccw}$.

5.2 Illuminating angles computation

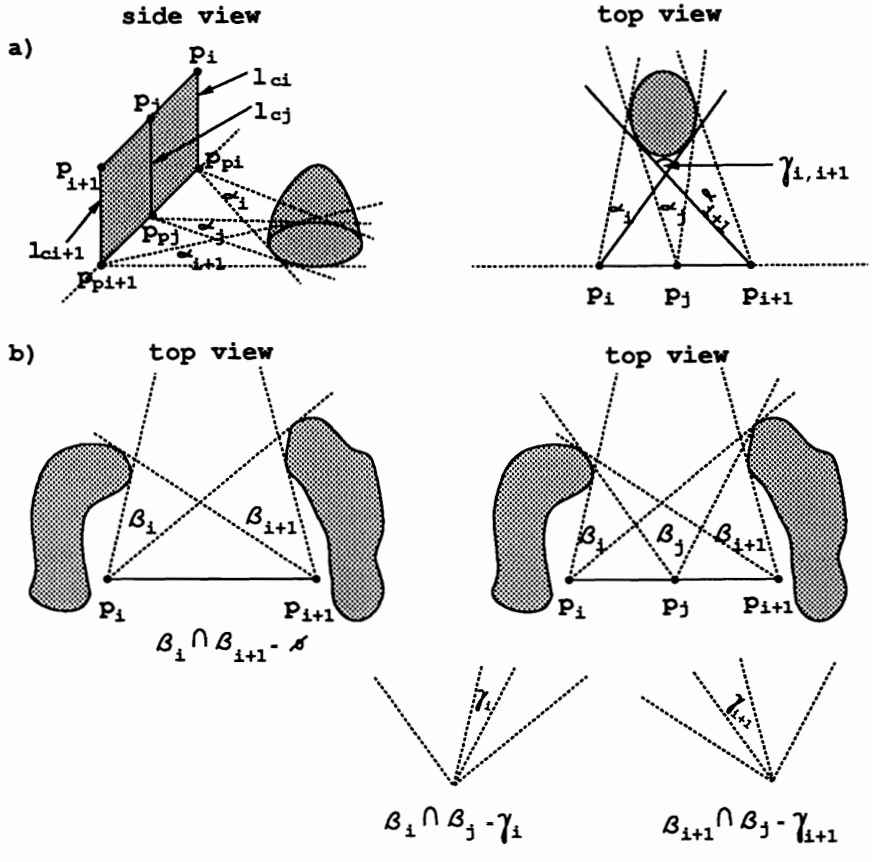

Figure 15: Illuminating angle computation

To compute the illuminating angle $\Psi(F_i)$ for each face F_i we must locate the areas (islands) I in the $2\frac{1}{2}$-D image of the first scanning plane which are higher than the height of projection $h_p(l_i)$ of the occluding line segment l_i. We find the contours for each l_i of such islands I with the same TRACER algorithm as we found the contours of occluded regions in the previous section. These islands represent the obstacles for the light. We must find all these directions from which the light

reaches the face F_i without collisions with the islands. To find a solution to this problem we use the work of G.T.Toussaint and J.R.Sack [1983] which has solved the problem of moving polygons in the plane.[1] The line, on which lies the line segment l_i, separates the first scanning plane into two parts. If on the left side of the line there is no such island I which is higher than $h_p(l_i)$, then the face F_i can be illuminated from any direction in $[\psi(l_i), \psi(l_i) + 180°]_{ccw}$. If there is an obstacle island, then each line lc_j, on face F_i connecting the point p_j on the occluding line segment l_i and its projection p_{p_j}, (fig. 15a) can be seen from a different subset of directions from the range $[\psi(l_i), \psi(l_i)+180°]_{ccw}$. The illuminating angle of line lc_j for K obstacle islands is:

$$\Psi(lc_j) = [\psi(l_i), \psi(l_i) + 180°]_{ccw} - (\cup_{k=1}^{K} \alpha_j(I_k)). \tag{16}$$

The angle $\alpha_j(I_k)$ includes all directions from which the island I_k occludes the line lc_j. If we insist upon illuminating the whole face F_i at the same time, then the face can be illuminated, if the rotation angle δ is selected from the illuminating angle:

$$\Psi(F_i) = [\psi(l_i), \psi(l_i) + 180°]_{ccw} - (\cup_{k=1}^{K} \gamma_{i,i+1}(I_k)). \tag{17}$$

The angle $\gamma_{i,i+1}(I_k)$ is defined as interval limited by the first (minimal) direction (d_i) which belongs to the angle $\alpha_i(I_k)$ and the last (maximal) direction (d_{i+1}) of the angle $\alpha_{i+1}(I_k)$, if one moves counterclockwise.

$$\gamma_{i,i+1}(I_k) = [d_i(I_k), d_{i+1}(I_k)] = \bigcup_j \alpha_j(I_k). \tag{18}$$

$$d_i(I_k) = \min_d \{d | d \in \alpha_i(I_k)\}$$

$$d_{i+1}(I_k) = \max_d \{d | d \in \alpha_{i+1}(I_k)\}$$

If the illuminating angle $\Psi(F_i)$ is empty, then the face F_i cannot be illuminated at once. The solution can be found by splitting the line segment l_i and respectively the face F_i into two parts and computing the illuminating angle for each part separately (fig. 15b). In the drawing on figure 15a, each line lc_j on the face F_i is occluded from all directions inside the angle α_j. The whole face F_i can be illuminate from any direction in the range $[\psi(l_i), \psi(l_i) + 180°]_{ccw} - \gamma_{i,i+1}$. In the illustration on figure 15b the line defined by the first endpoint p_i and its projection p_{p_i} can be illuminated from any direction in the angle β_i. Also, the line defined by the second endpoint p_{i+1} and its projection $p_{p_{i+1}}$ can be illuminated from any direction in the angle β_{i+1}. Since the intersection $\beta_i \bigcap \beta_{i+1}$ is empty, the face F_i belonging to the line segment l_i cannot be illuminated from only one scanning plane. By splitting the line segment l_i into two parts, the solution is found. The face F_i must be

[1] The authors answered the following question: Given a direction d can the polygon P be translated in an arbitrary distance in direction d without colliding with polygon Q.

illuminated from the two planes. The orientation δ of the first scanning plane must be selected from the angle γ_i, and the orientation δ of the second scanning plane from the angle γ_{i+1}.

5.3 Computation of the next scanning plane

After computing the illuminating angles $\Psi(F_i)$ for all faces F_i, the angles δ of the next scanning planes are determined from the histogram in the same way as we compute the new scanning directions in the previous section. The directions in the necessary histogram maxima are the initial candidates for rotation angles δ for the next scanning planes.

6 Experimental Results

We tested our algorithm on a number of different scenes. Results are shown for two different scenes. The range images were scanned using a structured lighting laser-scanner with approximately 1mm/pixel spatial resolution and 1.2mm depth resolution.

Scene 1: The scene consists of four boxes of different heights and different sizes. The first view is done in direction 0°. In the image of the first view (fig. 16 left) the highest box partially occludes the lowest box. The other three boxes are arranged so that they occlude only the support. The contour of the occluded region whose intensity value is 0 (black area) is represented in the image of figure 16 right. From the curvature function of the contour of the occluded region (fig. 17), the polygonal approximation is computed. By searching around the contour in the belt of 7 pixels wide, the border height is obtained (fig. 18 left). The height of the border between two vertices is approximated by a constant which is the median of all height values between the vertices (fig. 18 right). From the angle of the edge and from its height, the type of the edge is defined (fig. 19). After computing the viewing angles, the histogram is built (fig. 20 left). It has two maxima $[129°, 144°]$ and $[157°, 188°]$. We need two additional views to get the complete $2\frac{1}{2}$-D representation of the scene. We can select as a next scanning direction any direction from the range of both maxima. Since the maximum $[157°, 188°]$ is wider and so more stable, the next view is selected from it. The selected direction is 180°. The third direction can be selected from the lower maximum. If we remove all viewing angles which include the direction 180° from the histogram, we can build a new histogram (fig. 20 right) where we get wider range $[90°, 144°]$ for the third view. The third selected scanning direction is 120°. In figure 21, there are the images of both selected directions. By merging the images of the first and the second view we get the image in figure 22 left. The image of all three views in the common coordinate plane is in figure 22 right. To compute the rotation angles δ for the next scanning planes from the side, we first locate edges in the image of all three views (fig. 23 left). The edges represent the occluding borders which are approximated

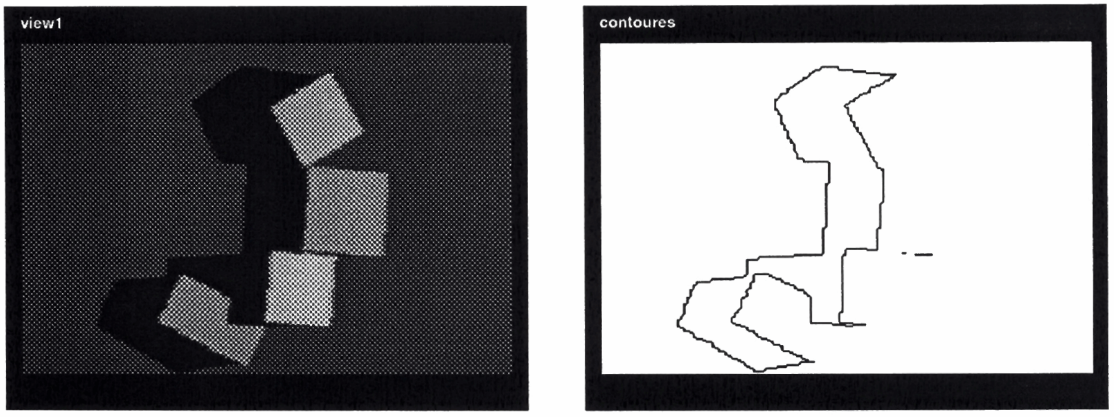

Figure 16: The first view and the contour of the occluded region

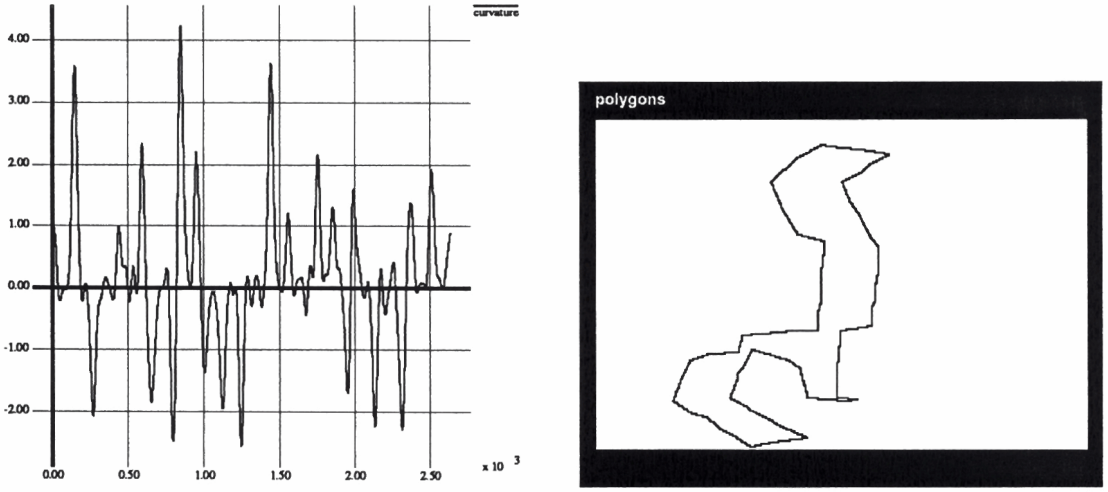

Figure 17: The curvature of the contour of the occluded region and the polygonal approximation of the occluded region

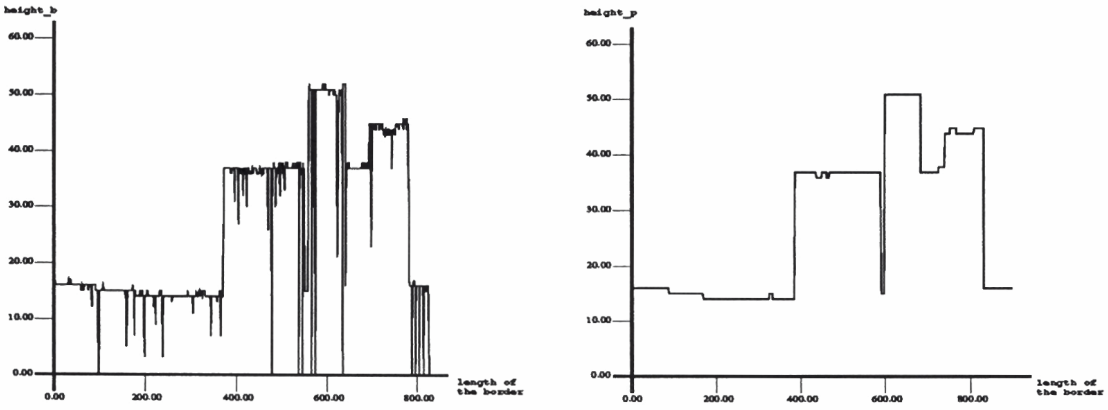

Figure 18: The height of the border of the occluded region and its approximation

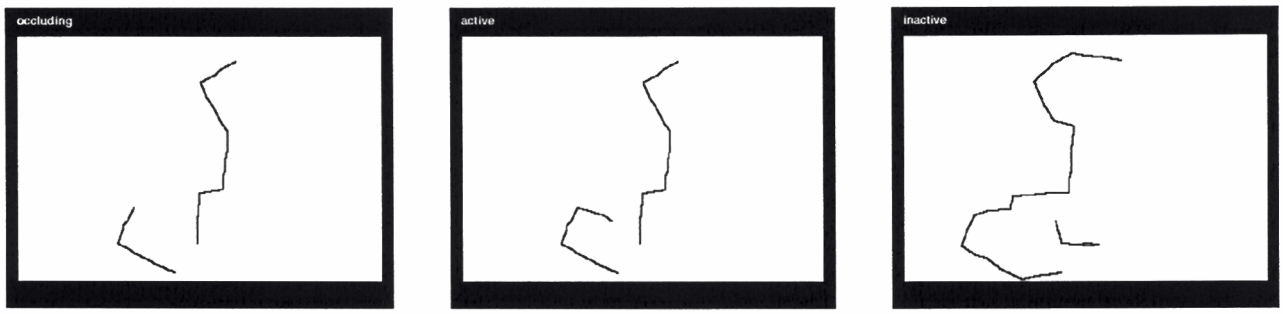

Figure 19: Occluding, active, and inactive edges

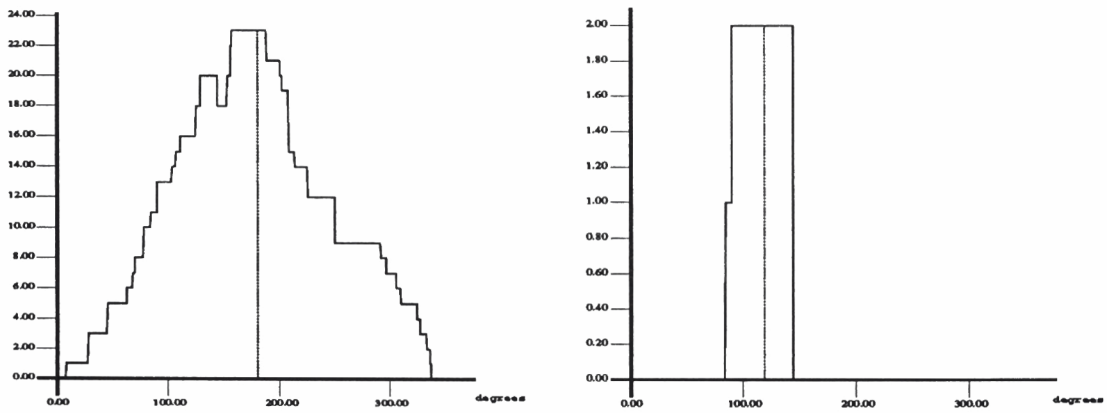

Figure 20: Histograms of viewing angles

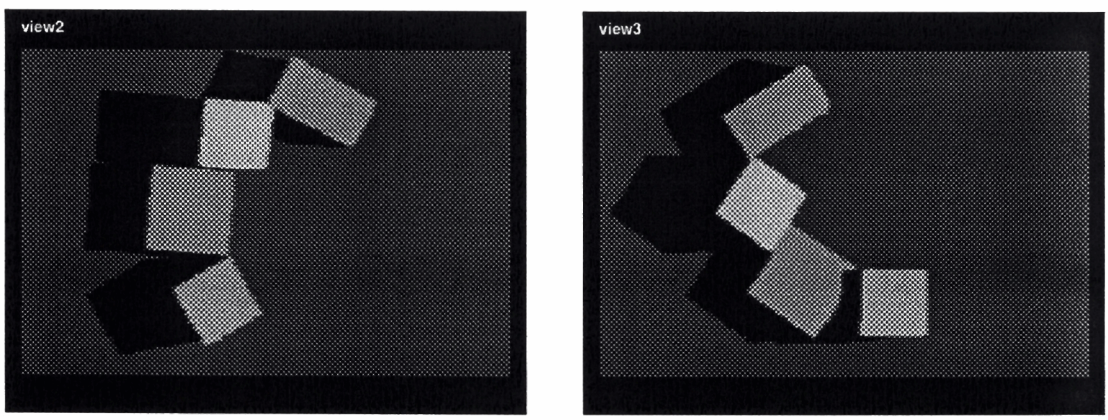

Figure 21: The second and the third view

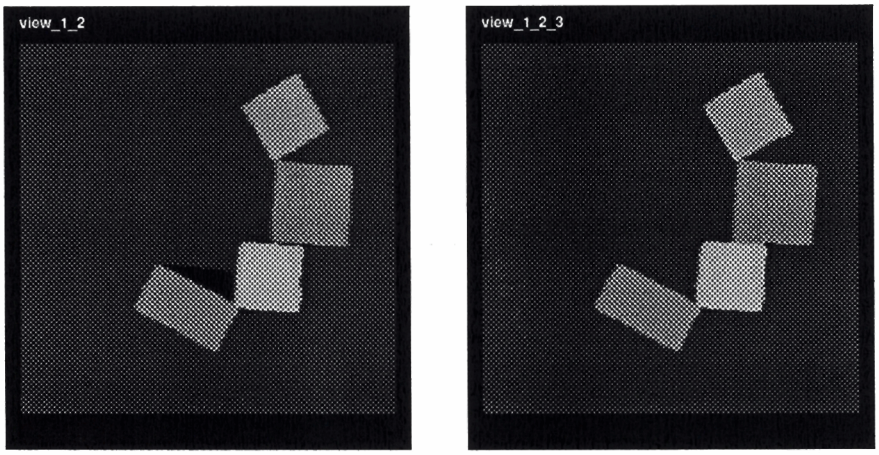

Figure 22: Merging the images of all three views in the common coordinate frame

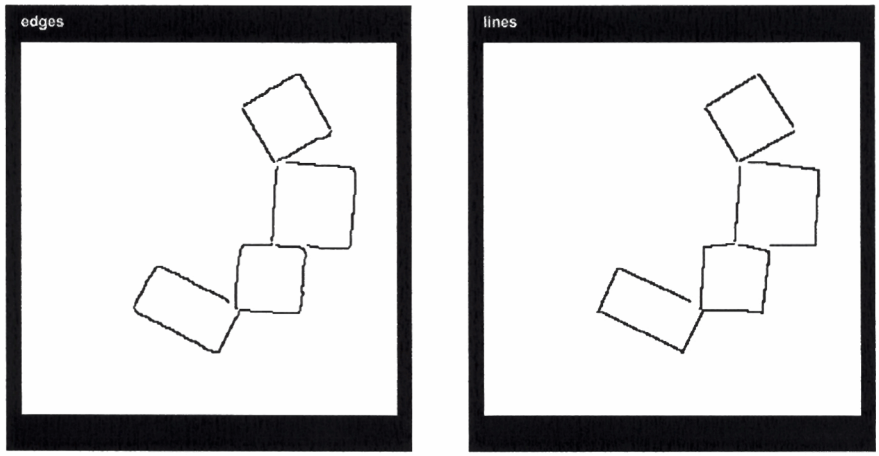

Figure 23: Edges and their linear approximation

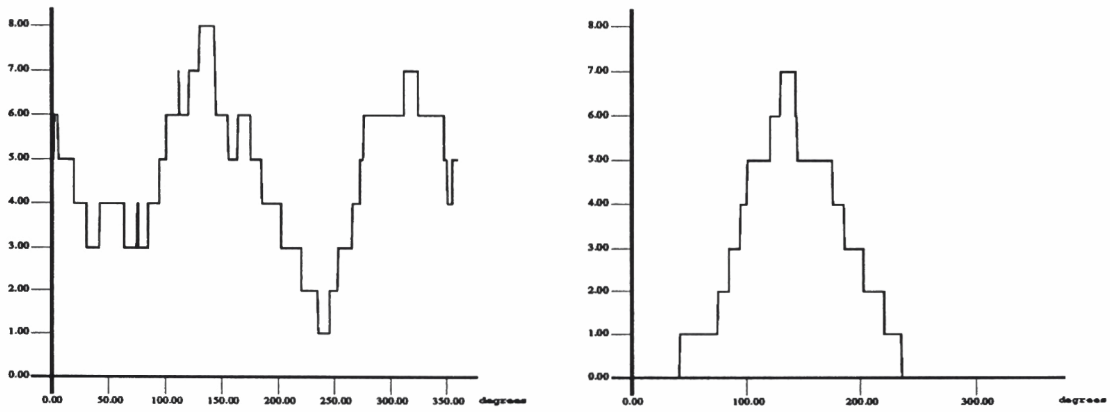

Figure 24: Histograms of illuminating angles

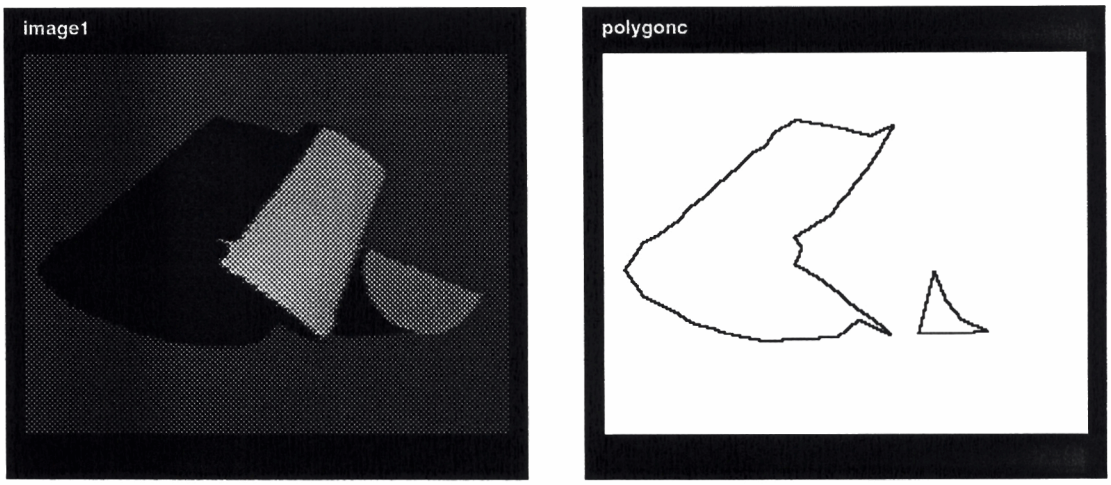

Figure 25: The first view of scene 2 and the polygonal approximation of the occluded regions

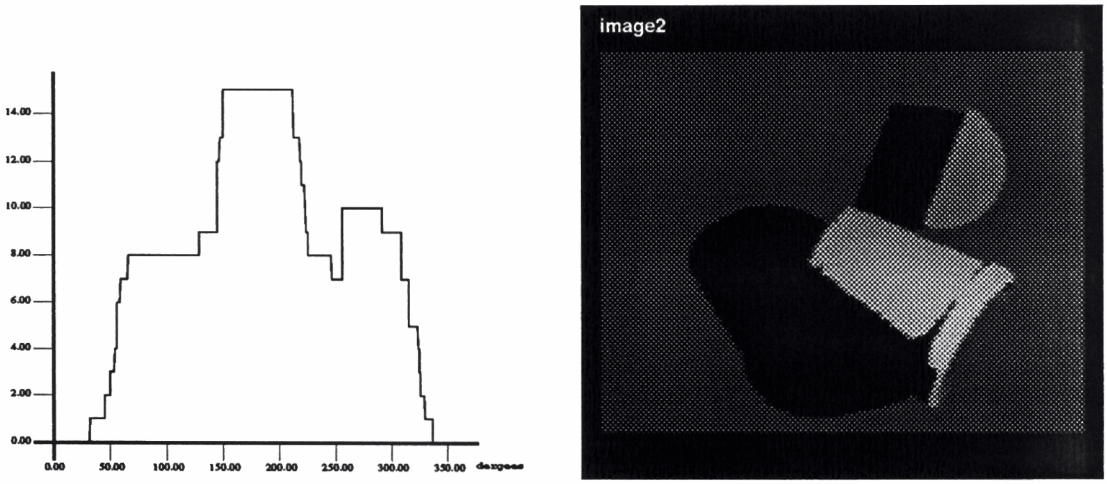

Figure 26: Histogram of viewing angles and the second view

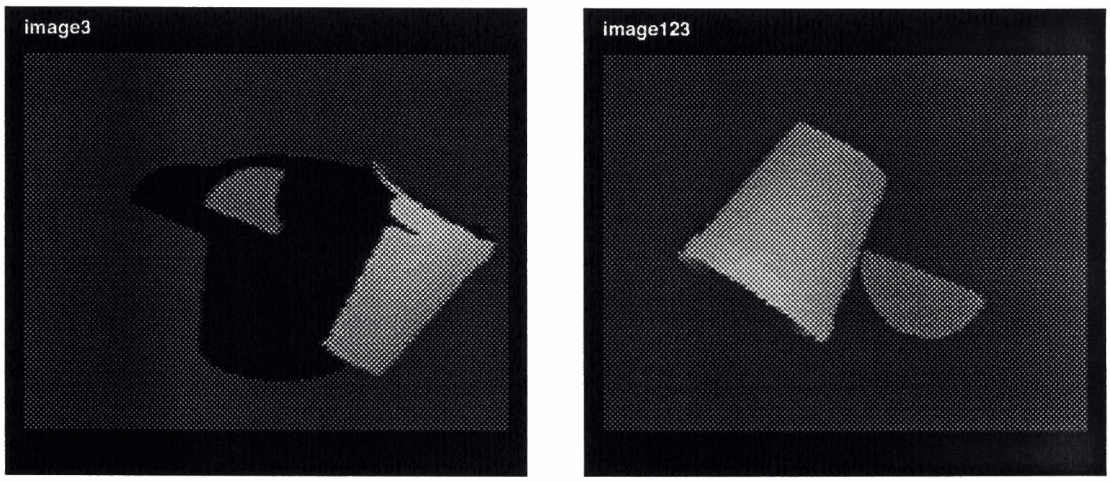

Figure 27: The third view and all three views in the common coordinate frame

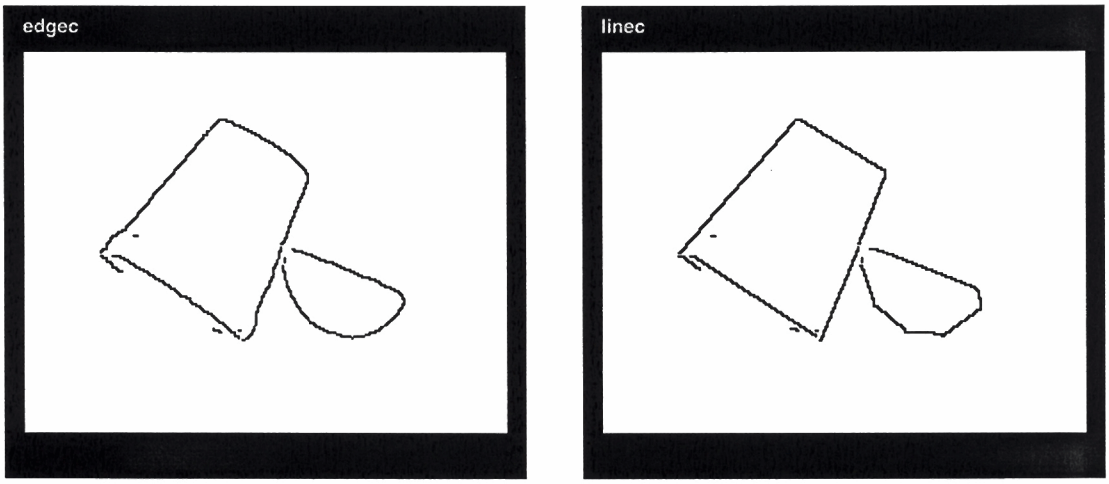

Figure 28: Edges and their linear approximation

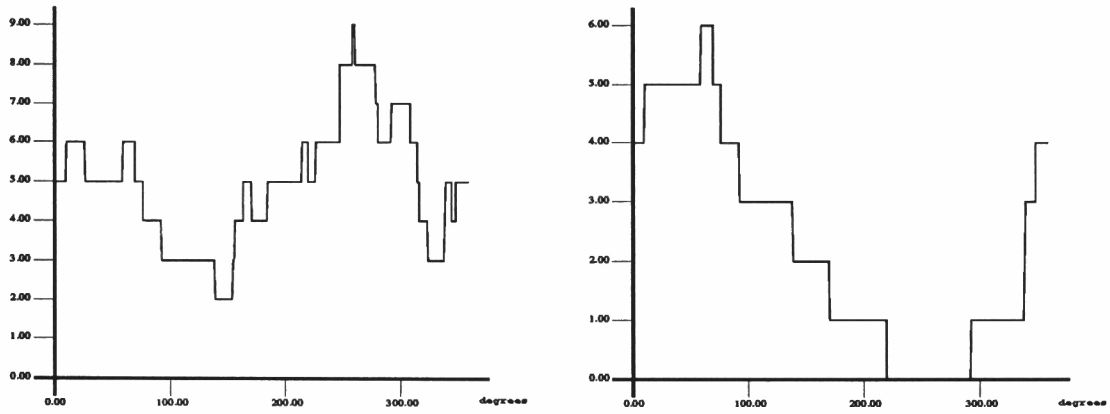

Figure 29: Histograms of illuminating angles

by piecewise linear segments (fig. 23 right). For each line, the illuminating angle is computed and the histogram of illuminating angles is constructed (fig 24 left). In the histogram there are seven maxima. During the histogram decomposition, two necessary maxima are found $[2°, 5°]$ and $[312°, 324°]$. From the illuminating angles which do not include the necessary maxima, the new histogram is built. In the second histogram (fig. 24 right) we have only one maximum $[130°, 144°]$. From all three maxima the rotation angle δ for the next scanning planes must be selected.

Scene 2: The scene consists of a cup and a low geometrical object which represents a half of a cylinder (fig. 25 left). The first view is done in direction 0°. From the polygonal approximation of occluded regions (fig. 25 right) the histogram of viewing angles is constructed (fig. 26 left). The histogram has two maxima: $[150°, 212°]$ and $[256°, 291°]$. Two additional views are done from directions 270° (fig. 26 right) and 180° (fig. 27 left). All three views are then merged in the common coordinate frame (fig. 27 right). The complete $2\frac{1}{2}$-D image is obtained. From the edges (fig. 28 left) their linear approximation are computed (fig. 28 right). For each linear segment the areas which are higher than the height of its projection are located and the illuminating angle is computed. From the illuminating angles, a histogram is constructed (fig 29 left). During the histogram decomposition one necessary maximum is found $[258°, 260°]$. From the illuminating angles which do not include the maximum $[258°, 260°]$ the second histogram is built (fig. 29 right) which has only one maximum $[59°, 69°]$. The scene must be illuminated from the side from two scanning planes. The angles δ must be selected from the intervals $[258°, 260°]$ and $[59°, 69°]$.

7 Conclusions

We have developed an algorithm for selecting the proper views to produce the complete $2\frac{1}{2}$-D data of the scene then from this complete $2\frac{1}{2}$-D data of the first scanning plane we compute the next scanning planes for 3-D data acquisition. Both planar and curved surfaces are treated in a uniform manner. For the construction of $2\frac{1}{2}$-D data, only the information in a narrow zone around the occluded regions are used. The holes in a surface which are occluded to the camera from all directions in the first scanning plane cannot be seen because of geometrical properties of the sensor system. To compute the next scanning planes we exploit the occluding borders which are located by an edge operator in the image of first scanning plane. In planning the next scanning planes, we have limited ourselves to the planes which are perpendicular to the first one.

Research is continuing on solving the problem of planning the next scanning planes without limitation.

Acknowledgements

The authors wish to thank Ulf Cahn von Seelen and Howie Choset for comments on an earlier draft of the paper.

This research was supported in part by AFOSR Grants 88 0244, AFOSR 88-0296; Army/DAAL 03-89-C-0031PRI; NSF Grants CISE/CDA 88-22719, IRI 89-06770; ARPA Grant N0014-88-K-0630 and Dupont Corporation.

References

[Ahuja and Veenstra, 1989] N.Ahuja and J.Veenstra, "Generating Octrees from Object Silhouettes in Orthographic Views, " *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, pp. 137-149, Feb. 1989.

[Bajcsy, 1988] R.Bajcsy, "Active Perception," *Proceedings of the IEEE*, Avgust 1988

[Krotkov and Kories, 1988] E.Krotkov, R.Kories, "Integrating Multiple Uncertain Views of a Static Scene Acquired by an Agile Camera System, " University of Pennsylvania, Tech. report MS-CIS-88-11, GRASP LAB 135, 1988.

[Nevatia and Babu, 1980] R.Nevatia and K.R.Babu, "Linear Feature Extraction and Description," *Computer Graphics and Image Processing,* Vol.13, 1980, pp. 257-269.

[Pavlidis, 1982] T.Pavlidis *Algorithms for Graphics and image processing*, Computer science press, 1982

[Sakane *et al.*, 1987a] S.Sakane, M.Ishii, and M.Kakikura, "Occlusion avoidance of visual sensors based on a hand-eye action simulator system:HAVEN," *Advanced Robotics*, Vol.2, No.2, pp. 149-165, 1987.

[Sakane *et al.*, 1987b] S.Sakane, T.Sato, and M.Kakikura, "Model-based planning of visual sensors using a hand-eye action simulator system:HAVEN," *Proceedings of the 3rd International Conference on Advanced Robotics ICAR'87*, pp. 163-174, 1987.

[Solina and Bajcsy, 1990] F.Solina and R.Bajcsy, "Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, pp. 131-147, Feb. 1990.

[Toussaint and Sack, 1983] G.T.Toussaint and J.R.Sack, "Some New Results on Moving Polygons in the Plane," in *Proc. Robotics Intelligence and Productivity Conference*, Detroit, Michigan, November 18-19, 1983.

[Tsikos, 1989] C.J.Tsikos *Laser Range Imaging System User's Guide*, 1989.