

## Scanning Electron Microscope-Based Stereo Analysis

Ali Kayaalp, A. Ravishankar Rao, and Ramesh Jain

Artificial Intelligence Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan

**Abstract:** In this paper we present a novel technique to analyze stereo images generated from a SEM. The two main features of this technique are that it uses a binary linear programming approach to set up and solve the correspondence problem and that it uses constraints based on the physics of SEM image formation. Binary linear programming is a powerful tool with which to tackle constrained optimization problems, especially in cases that involve matching between one data set and another. We have also analyzed the process of SEM image formation, and present constraints that are useful in solving the stereo correspondence problem.

This technique has been tested on many images. Results for a few wafers are included here.

---

**Key Terms:** stereo, applications of computer vision

---

### 1 Introduction

In the semiconductor IC manufacturing facilities of the future, individual fabrication processes will be controlled by intelligent systems. Automatic inspection systems will allow control of process parameters for quality control and yield improvement. Such systems will eliminate human inspectors and interface with expert systems to provide them with appropriate information for decision making.

Figure 1 illustrates a typical problem faced by the process engineer in a semiconductor plant. Figure 1a shows the desired profile for a surface that is to be obtained after etching. Figures 1b, c, and d show profiles that could result from malfunctions in the photolithography process. Such profiles may arise due to the formation of “feet” at the bottom of the

lines, or due to the presence of photoresist “scum” that may be left behind as a residue. Furthermore, the slope of the sidewall is a critical factor in the lithography of submicron devices.

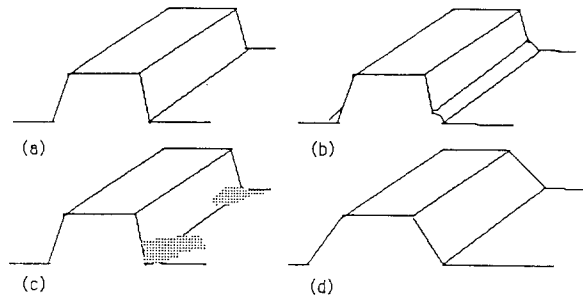
Thus, there are several parameters of the actual surface that are crucial to controlling the lithography process, such as height of the step, slope of the sidewall, and presence of undesired material. How does one measure these parameters? There are several methods available to do this, such as profilometry, analysis of the cross section of the sample after cleaving, stereo using optical microscopes, and SEM stereo. However, there are restrictions for the problem of semiconductor wafer inspection, which narrow the range of options. These are mainly that the method must be nondestructive and have a high (submicron) resolution. In the light of these restrictions, SEM stereo seems to be particularly attractive because of its nondestructive nature and the high resolution it offers.

Binocular stereo vision is the major mechanism used by humans for obtaining three-dimensional depth information over close distances. It is based on obtaining three-dimensional surface information from two images of a scene taken at two different viewing angles. On the SEM the stereo image pair can be obtained by tilting the specimen and taking its images at two different tilt angles.

Given the needs of automated semiconductor wafer manufacturing, the development of an automated SEM stereo algorithm is very crucial to fulfill these needs. However, SEM stereo algorithms that are used currently are not fully automated, in that the operator has to perform manually the correspondence between features in the two stereo images. The major computational effort in a stereo algorithm is solving the *correspondence problem*. Once correspondence is established, three-dimensional depth (height) information can easily be obtained using a set of three-dimensional reconstruc-

---

Address reprint requests to: Ramesh Jain, Artificial Intelligence Laboratory, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122, USA.



**Figure 1.** (a) Desired profile for a step. (b) Profile that results from the formation of feet. (c) Profile that results due to the deposit of “scum.” (d) Profile resulting from under etching. Note that the side wall slope is not so steep as desired.

tion equations, derived using the geometry of image formation and the specific camera setup used in getting the stereo image pair.

We address the problem of automating SEM stereo in this paper. Although stereo techniques have been widely researched in computer vision, they have been used mainly on optical images, and none of them has been applied to SEM images. A novel aspect of our approach to the correspondence problem is that we view it as an optimization problem, based on binary linear integer programming. This puts the problem on a sound mathematical footing, as several methods are available to solve such optimization problems. Furthermore, principles from the physics of SEM image formation have been used in order to solve the stereo problem. This combination of SEM imaging constraints, together with an optimization approach, results in a powerful technique for solving the SEM stereo problem. Our results show that the SEM stereo algorithm that we have developed works very well on real images. An outcome of our research is the view that not only stereo but other matching problems, such as image-to-model matching, can be cast in the framework of an optimization problem. What is important is the transformation of problem constraints to the form of a linear programming problem. The details of such a transformation are clearly developed in this paper.

## 2 Background

In this section we briefly consider methods for extracting surface topography information. Surface microtopography information has been used in the quality control of surfaces in metal finishing, optical polishing (Larrabee 1977) and in monitoring some

semiconductor processes by analyzing the wafer surface (Kato et al. 1977). Larrabee (1977) reviews various techniques that are used for extracting surface microtopography, their limitations and characteristics.

Based on the physical principle behind their operation, these techniques can be classified into mechanical techniques, optical techniques, and electron-beam techniques. We will consider the pros and cons of each technique within the context of the problem in hand, that is, the nondestructive extraction of surface topography from semiconductor wafers with *submicron* patterns.

The mechanical technique is based on measuring the vertical displacement of a stylus tip as it is scanned across the specimen surface. The principal limitations of this technique (Vorburger and Teague 1981) are the slowness of the device for on-line applications, the possibility of surface damage due to the sharp stylus, the fragility of the transducer and stylus tips, and the fact that the best horizontal resolution is only  $0.1 \mu\text{m}$ .

A comprehensive overview of optical techniques may be found in Vorburger (1981). The main shortcoming of optical techniques is the horizontal resolution, which is limited by the wavelength of light used. The horizontal resolution reported is typically in the  $0.4$  to  $0.5 \mu\text{m}$  range (Singer 1983). This value is unacceptable for making accurate measurements on submicron features. Conventional optical microscopes have poor vertical resolution due to their relatively large depth of focus of around  $1.0 \mu\text{m}$  (Larrabee 1977). Thus, instruments with higher resolution will be needed.

Currently, low-voltage scanning electron microscopes (SEMs) are the most commonly available, general purpose high-resolution inspection instruments that can be used (Rose 1982; Singer 1983). The horizontal resolution of a SEM for typical commercial equipment is in the  $20$  to  $25 \text{ nm}$  range (Nanometrics 1984). Another advantage of SEMs over some optical devices is its much larger depth of focus.

Viable nondestructive techniques for extracting specimen surface microtopography using the SEM can be grouped into the following:

1. *The stereo technique.* A SEM stereo pair is commonly obtained by tilting the sample. Several researchers have investigated SE image formation in the SEM and the use of stereo techniques [see Lane (1969), Hilliard (1972); Piazzesi (1973); Boyde (1974a, b); and Kato (1977)].

Currently, the extraction of three-dimensional data using SEM stereo techniques is mostly car-

ried out semiautomatically,<sup>1</sup> in that the correspondence problem is performed manually by positioning markers on the CRT screen. The rest, namely, obtaining the disparity and making the necessary calculations, are performed automatically. The reason for this is the difficulty of solving the stereo correspondence problem. Our aim in this work was to find a systematic solution to the correspondence problem, thereby fully automating the SEM stereo-based surface microtopography extraction process.

2. *The multiple detector technique.* The analogue of this technique in computer vision is the *photometric stereo* technique (Woodham 1979; Horn 1986). This technique is based on relating the orientation of a local surface patch on the specimen defined by the electron beam spot to the signal detected by the SEM secondary electron and backscattered electron (BSE) detectors. Details of the system using such a technique may be found in Lebedzik and White 1975).

### 3 Theory

The novelty of the technique presented in this paper is in setting up the stereo-matching problem as a binary linear integer programming problem. Although linear programming problems have been well researched and understood (Hillier and Lieberman 1980), there has been no application of these ideas to stereo vision thus far. Ullmann (1979) used a linear programming technique to solve the correspondence problem in motion analysis.

There are several advantages to using a linear programming framework for analyzing problems in computer vision.

1. These techniques have been well studied in the operations research discipline (Hillier and Lieberman 1980; Murty 1983). Given certain constraints on the problem, such as the nonnegativity of the cost vector one is *guaranteed* to find an optimal solution to the problem.
2. Mathematically sound algorithms for solving various types of optimization problems are widely available (Hillier and Lieberman 1980; Murty 1983). This enables one to concentrate on formalizing the specifications of a physical problem rather than getting involved in the implementation details of an algorithm to solve the problem.

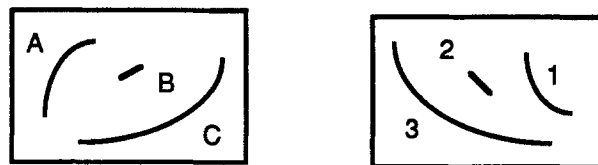


Figure 2. Two sets of two-dimensional image contours.

3. The linear programming framework provides a systematic and unified scheme to handle different constraints that may exist in a given domain. Modifying the specifications of a physical problem typically amount to adding or deleting some of the constraints in the equivalent optimization problem, without necessitating a change in the way the problem is solved.
4. The same linear programming technique can be used in a variety of domains, such as stereo matching and image-to-model matching. In fact, the biggest hurdle in the use of this technique is to transform physical constraints into the format of a linear programming problem. One of the goals of this paper is to analyze SEM-based stereo, and to demonstrate how the various constraints can be cast in the mold of a linear programming problem.

In this section we discuss how the stereo-matching problem can be formulated as a general contour matching that can be easily transformed into an equivalent binary linear programming problem.

#### 3.1 The Contour-Matching Problem in Computer Vision

Given two sets of two-dimensional contours, the contour-matching problem tries to find the overall best match between pairs of contours, with one member of each pair being an element of one set and the other, an element of the other set. Figure 2 shows two sets of two-dimensional contours. We seek to associate each contour in the left image of Figure 2 with some contour in the right image with the aim of determining a set of associations that will best meet the requirements of our specific problem.

Using the contour sets shown in Figure 2, a simple example will be the problem of finding the association that results in the smallest difference in the lengths of matching contours while ensuring that each left and right image contour has only one match. The solution of this problem would be *A* matching 1, *B* matching 2, and *C* matching 3.

Several problems in computer vision can be formulated as contour-matching problems. Examples

<sup>1</sup> See Breton et al. (1987) for recent work on automating this process. More on this is discussed later under shape from stereo using area matching.

include object recognition, shape-from-stereo and model-based shape inspection problems. The requirements of each matching problem are different, however.

### 3.2 Contour-Matching Problems Posed as Binary Programming Problems

The solution of a contour-matching problem has to meet the following criteria:

- Satisfy all the requirements of the problem.
- It is the *best* solution among all others that satisfy the problem requirements.

In the example problem presented previously we required that each contour have only one match, and we defined the best match as the one resulting in the minimum difference in the lengths of matching contours.

It turns out that problems with the preceding characteristics can be directly modeled as optimization problems. The requirements of the problem correspond to *constraints* in an optimization problem, and a candidate solution (referred to as a *feasible solution* (Murty 1983)) is evaluated based on the value of an *objective function*.

Now let  $\mathbf{X}$  represent a vector of variables where each variable denotes the match of a left image contour with a right image contour. This variable has two values, 0 or 1; 0 representing the rejection of that match and 1 representing its acceptance. Let us assume that the variables  $x_1$ ,  $x_2$ , and  $x_3$  represent the match of contour  $A$  in the left image with contours 1, 2, and 3 in the right image (see Figure 2). The requirement of  $A$  having only one match can now be represented as  $x_1 + x_2 + x_3 = 1$ . If  $x_1$  is 1, then its contribution to the objective function would be the difference in the lengths of contours 1 and  $A$ ; in general, the contribution of match  $i$  will be  $x_i \cdot |L(i) - R(i)|$ , where  $L(i)$  and  $R(i)$  are matching left and right image contours, respectively. Hence, the equivalent optimization problem will be:

$$\begin{aligned} \min \sum x_i \cdot |L(i) - R(i)| \\ \text{over all binary vectors } \mathbf{x} = (x_1, x_2, \dots) \quad (1) \\ \text{subject to} \quad x_1 + x_2 + x_3 = 1 \end{aligned}$$

where the binary variable  $x_i$  represents the match of left and right image contours  $L(i)$  and  $R(i)$ .

Such problems where all the variables have a 0 or 1 value are referred to as *binary programming problems* (Hillier and Lieberman 1980). Furthermore, if both the objective function and the constraints are linear functions, then the problem becomes a *binary linear programming problem*.

The contour-matching problems that are discussed in this paper are posed as binary linear programming problems that have the special form:

$$\begin{aligned} \max \mathbf{p} \cdot \mathbf{x}^t \quad \text{over all binary vectors } \mathbf{x} \quad (2) \\ \text{subject to} \quad \mathbf{C} \cdot \mathbf{x}^t \leq \mathbf{b}^t \end{aligned}$$

where  $\mathbf{p}$  is the *objective function coefficient vector*,  $\mathbf{x}$  is the *vector of variables* with binary elements,  $\mathbf{C}$  is the *constraint coefficient matrix* with binary elements and  $\mathbf{b}$  is the *constraint right-hand side vector*.

### 3.3 Solving Binary Programming Problems

The most popular techniques for solving reasonable size binary programming problems are based on implicit enumeration via branch and bound techniques. A branch and bound algorithm for solving binary programming problems was first proposed by Balas (1965). Since then many variations of such algorithms have been developed. In our work an algorithm of this type presented in Hillier and Lieberman (1980) was used.

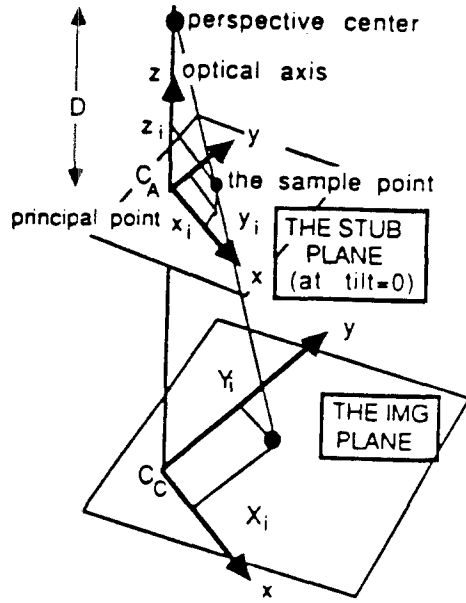
## 4 Methods

In this section we examine the problem of SEM stereo in more detail. This section is divided into two parts. The first part deals with SEM imaging geometry and presents the stereo reconstruction equations used in our experiments. The second part investigates the constraint that we use in stereo matching and shows how these constraints can be expressed in the form of a binary linear integer programming problem.

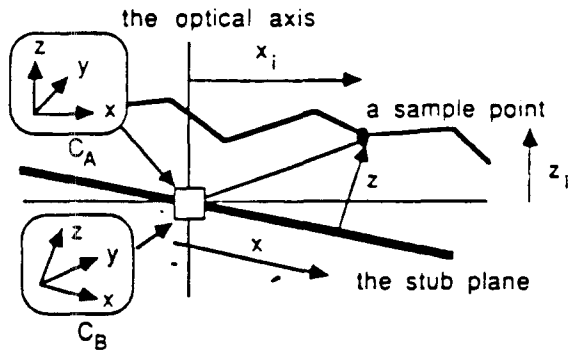
### 4.1 SEM Stereo Reconstruction Equations

Image formation in the SEM can be modeled as the perspective projection of specimen surface points onto an image plane. The image plane is perpendicular to the optical axis, with the perspective center placed at the final aperture in the objective lens of the SEM (see Figure 3a). To simplify the algebra that follows, three right-handed Cartesian coordinate frames are introduced. They are defined as follows (see Figure 3):

- Coordinate frame  $C_A$  is positioned at the point where the optical axis of the SEM crosses the stub plane, which is called the *principal point* (see Figure 3a). Its  $Z$ -axis is aligned with the optical axis and points toward the perspective center. Its  $Y$ -axis is parallel to the tilt axis of the stage and will point in such a direction as to cause the tilt made, in getting the stereo pair, to be equivalent to a rotation in the  $+Y$ -axis, using the right-hand



(a)



(b)

**Figure 3.** SEM stereo imaging geometry. (a) Sample of the stub plane at stage tilt position  $i$ , and illustration of the coordinate frames  $C_B$  and  $C_C$ . (b) Illustration of the image plane and the coordinate frames  $C_A$  and  $C_C$ .

rule convention. Coordinates of points expressed with respect to this frame are printed using lowercase  $x, y, z$ 's with a subscript [e.g.,  $(x_i, y_i, z_i)$ ] where the subscript indicates the two tilt positions of the stage.

- Coordinate frame  $C_B$  will also be positioned at the principal point, but it is attached to the stub plane (see Figure 3b). Its orientation is such that at  $0^\circ$  tilt it becomes identical to coordinate frame  $C_A$ . Coordinates of a point with respect to this frame

is expressed using lowercase  $x, y, z$  with no subscripts [i.e., as  $(x, y, z)$ ]. Note that the coordinates of a sample point with respect to this frame will not change as the sample is tilted,

- Coordinate frame  $C_C$  is positioned at the intersection of the optical axis with the image plane (see Figure 3a). It is oriented such that its  $X$ - and  $Y$ -axes lie on the image plane and are aligned with the  $X$ - and  $Y$ -axes of the other two coordinate frames, respectively. Coordinates of a point on the image plane with respect to this frame is denoted by uppercase  $x, y$ 's with a subscript [i.e.,  $(X_i, Y_i)$ ] to indicate the two tilt positions of the stage. Also note that only the  $x$  and  $y$  coordinates of this frame are used.

Making use of these coordinate frames, the overall transformation relating image points to three-dimensional scene points (and vice versa) can be decomposed into two simpler transformations, each of which can be obtained by simple algebra and geometry, as follows:

- Using the illustrating in Figure 3b, it can be shown that the transformation  $T_A^B$ , which transforms the coordinates of surface points expressed with respect to  $C_B$ , that is,  $(x, y, z)$ , into those expressed with respect to  $C_A$ , that is,  $(x_i, y_i, z_i)$ , can be given implicitly through the following relations:

$$z = x_2(\sin \gamma + \cos \gamma / \tan \alpha) - x_1(\cos \gamma / \sin \alpha) \quad (3)$$

$$x = x_2(\cos \gamma + \sin \gamma / \tan \alpha) + x_1(\sin \gamma / \sin \alpha) \quad (4)$$

$$y = y_1 = y_2 \quad (5)$$

where  $\gamma$  is the tilt angle of the stage after being tilted and  $\alpha$  is the amount of stage tilt. Note also that the coordinates  $(x_1, y_1, z_1)$  of a specimen surface point at position I of the stage becomes  $(x_2, y_2, z_2)$  upon tilting the stage by an angle  $\alpha$ , as follows:

$$y_1 = y_2 \quad (6)$$

$$x_2 = x_1 \cos \alpha - z_1 \sin \alpha \quad (7)$$

$$z_2 = x_1 \sin \alpha + z_1 \cos \alpha \quad (8)$$

- Similarly, using the illustration in Figure 3a, it can be shown that the transformation  $T_C^A$ , that transforms the coordinates of specimen surface points expressed with respect to  $C_A$  [i.e.,  $(x_i, y_i, z_i)$ ] into their corresponding image coordinates expressed with respect to  $C_C$ , that is,  $(X_i, Y_i)$ , can be given implicitly through the following relations:

$$x_1 = (X_1/M_x)(1 - z_1/D) \quad (9)$$

$$y_1 = (Y_1/M_y)(1 - z_1/D) \quad (10)$$

$$x_2 = (X_2/M_x)(1 - z_2/D) \quad (11)$$

$$y_2 = (Y_2/M_y)(1 - z_2/D) \quad (12)$$

where,  $D$  is the working distance (i.e., the distance from the perspective center to the principal point) and  $M_j$  is the magnification from the stub plane at  $0^\circ$  tilt to the image plane, along axis- $j$ .

At high magnifications, that is, above  $\sim 500$  to  $1000\times$  depending on the working distance, the imaging geometry can be modeled as the orthographic projection of specimen surface points onto the image plane with negligible error (Howell 1978b). In this case the transformation  $T_C^A$  simplifies to the following:

$$x_i = X_i/M_x \quad \text{for } i = 1, 2 \quad (13)$$

$$y_i = Y_i/M_y \quad \text{for } i = 1, 2 \quad (14)$$

where  $M_x$  and  $M_y$  are the magnifications along the  $X$ - and  $Y$ -axes of the  $C_A$  and  $C_C$  coordinate frames.

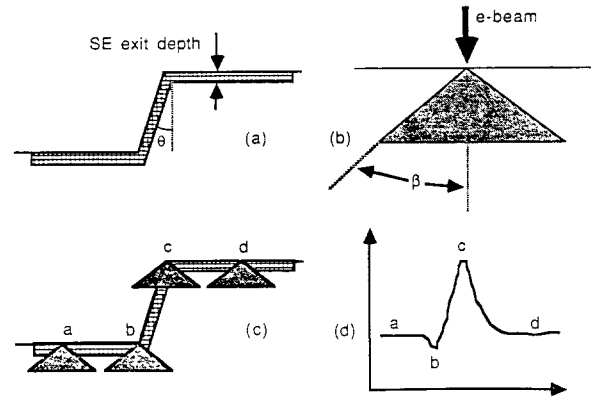
Due to the orthographic projection assumption, at a constant tilt angle, spatial dimensions on the projected image will be invariant to stage  $X$ - $Y$  translation. This is important because, unless the region of interest on the sample is positioned right on the tilt axis, upon tilting the specimen the stage will need to be translated in a direction perpendicular to the tilt axis so as to put the same region of the sample back into the field of view.

In summary, the relative three-dimensional position of one surface point with respect to another can be obtained using Eqs. (3) to (5) and (13) to (14), with the following modifications made in Eqs. (13) and (14):

$$X_i \leftarrow \Delta X_i, Y_i \leftarrow \Delta Y_i \quad (15)$$

where  $\Delta X_i$  is the difference in the  $X$  coordinates and  $\Delta Y_i$  is the difference in the  $Y$  coordinates (with respect to  $C_C$ ) of the pair of points whose three-dimensional separation we are after.

Once the image coordinate differences, that is,  $\Delta X_i$  and  $\Delta Y_i$ , are measured on the image, these values together with stage parameters, that is,  $\alpha$ ,  $\gamma$ , and  $M$ , can be substituted into the preceding equations to get relative three-dimensional spatial information between pairs of points on the sample. Thus, the problem now becomes one of locating projections of the same three-dimensional point in



**Figure 4.** Illustration of contrast due to the edge effect. (a) Surface profile, with the SE exit area crosshatched. (b) Model of the primary electron scatter area within the sample. (c) Computing the SE emission across the surface profile using the model in (b). (d) Illustration of the SE intensity corresponding to (c).

the two images, which is commonly known in computer vision as the *correspondence problem*.

#### 4.2 Image Formation in the SEM

We now present a brief description of the process of image formation in the SEM. It is important to consider this process because constraints to solve the stereo-matching problem can be generated from the knowledge of image formation. In fact, in section 4.3 we develop constraints for stereo matching on the SEM based on the knowledge of image formation.

On specimen rich in topographic detail, that is, one that has many high surface curvature points, the diffusion contrast (or the edge effect) is the primary contrast mechanism. Figure 4a shows a vertical cross section of a specimen, with the area over which secondary electrons can escape crosshatched. This exit area corresponds to the so-called *SE exit depth*, which is on the order of 0.5 to 1.5 nm for metals and 10 to 20 nm for insulators (Reimer 1985). Figure 4b shows a triangular approximation to the primary electron scatter area, similar to the model presented in Goldstein et al. (1981, p. 71). This model is based on analyzing the paths of primary beam electrons within the sample using Monte Carlo simulation. The apex angle of this triangle is equal to twice the average scattering angle  $\beta$ . We assume that the sidewall of the specimen surface (i.e., the slope of the sidewall profile in Figure 4a) is steep enough so that  $\theta$  is less than  $\beta$ . The height of this triangle is the maximum electron range that is on the order of  $0.3 \mu\text{m}$  for aluminum at an accelerat-

ing voltage of 5 keV and reaches to  $>0.8 \mu\text{m}$  at 10 keV.

Using this simple model, we can now estimate the shape of the SE intensity profile that would be obtained from the surface profile shown in Figure 4a. The relative magnitude of SE emission, when the primary beam hits a point on the surface, can be computed as the area of the crosshatched region that falls within the triangle of Figure 4b when the apex of the triangle is positioned at that surface point. This is illustrated in Figure 4c at several important points along the profile, and the corresponding intensity profile is illustrated in Figure 4d. As can be seen, high surface curvature points result in high curvature points in the corresponding image intensity profile. Furthermore, considering the surface and image intensity profiles as one-dimensional functions, locally convex high curvature regions on the corresponding image intensity profile, and similarly, locally concave high curvature regions of the surface profile result in locally concave high curvature regions in the image intensity profile.

### 4.3 The SEM Stereo Correspondence Problem

Solving the stereo correspondence problem requires two steps—the detection of predefined features in each image, followed by the matching of these features. The predefined features are termed *matching primitives*. In order to match the detected features, several constraints can be employed, and these are discussed in detail in the following section.

**4.3.1 Choice of matching primitives.** To reconstruct accurately the actual three-dimensional shape of an object, one needs to be careful in selecting matching primitives since shape from stereo algorithms can return the depths of only those three-dimensional object points that correspond to feature points in the image. Some of the factors affecting the choice of features include:

- Feature points corresponding to high surface curvature points on the sample should be used. The idea here is that the surface interpolation/approximation step that follows feature-based stereo matching will use these sparse sets of three-dimensional points as control points to fit smooth, low-order polynomial surface patches, and such surface patches that fit in between high surface curvature points will result in a very good approximation of the actual three-dimensional surface of the object. It was this factor that led Mayhew and Frisby (1981) to suggest that, for certain images,

zero-crossings alone were not able to predict the perceived shape by humans.

- The likelihood of the presence of the match of a feature point in the other image should be high.
- It should be possible to locate spatially corresponding feature points in the images accurately.

As primitives we use points where the curvature of the image intensity function  $i(x, y)$  achieves a local maximum with a sufficiently large magnitude,<sup>2</sup> along with intensity edges. Considering the physics of SEM secondary electron image formation, we will show the importance of locating high image curvature points based on the criteria given earlier. Incidentally, high image curvature points are nearly identical to the peaks of the  $\nabla^2 G$  convolved image that were suggested by Mayhew and Frisby (1981) as being used by humans as features. This is due to the fact that the curvature of a function is proportional to its second (partial) derivatives.

### 4.4 Constraints for the Matching Problem

The features that we proposed for detection in the previous section now have to be matched, based on certain constraints. In a general situation these constraints are derived from the imaging geometry, the physics of image formation, and the geometry of the three-dimensional scene being viewed.

The stereo-matching constraints that we have investigated are discussed as follows.

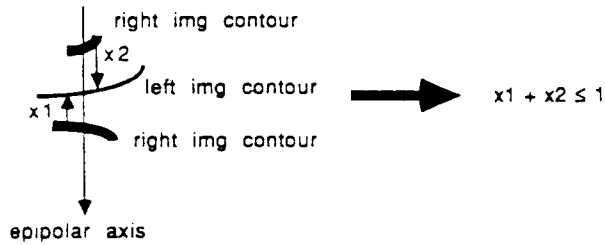
**4.4.1 The Epipolar Constraint.** The epipolar constraint states that the match of an image point on one epipolar line must lie somewhere along the corresponding epipolar line in the other image.

It is shown in Kayaalp (1988) that when a perspective projection imaging geometry is assumed, the match  $(X_2, Y_2)$  of a point  $(X_1, Y_1)$  in the first image, will lie in the other image on the line whose equation is given by

$$Y_2 = \left( \frac{Y_1(1 - \cos \alpha)/M_x}{D \cdot \sin \alpha - \frac{(1 - \cos \alpha)X_1}{M_x}} \right) \cdot X_2 + \left( \frac{D \cdot \sin \alpha \cdot Y_1}{D \cdot \sin \alpha - \frac{(1 - \cos \alpha)X_1}{M_x}} \right) \quad (16)$$

where  $\alpha$  is the amount of tilt,  $D$  is the SEM working distance, and  $M_x$  is the SEM magnification along the  $X$ -axis.

<sup>2</sup> Hereafter we will refer to these points as *high image curvature points*.



**Figure 5.** Implementation of the uniqueness of a left image contour's match constraint.

Note that, unlike the optical epipolar geometry where epipolar lines in one image correspond to epipolar lines in the other image, for SEM stereo pictures obtained by tilting the sample, the match of each image point is to be found on a different line in the other image.

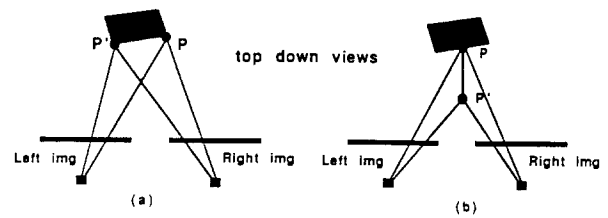
As discussed earlier, since we are operating at high magnification, we can make a parallel projection imaging geometry assumption with very little error. In this case the match of a point  $(X_1, Y_1)$  will lie on a line obtained by using Eq. (16) with  $D \rightarrow \infty$ , which gives

$$Y_1 = Y_2 \quad (17)$$

This is precisely the optical epipolar line constraint. Hence, imaging geometry dictates that for a tilt axis parallel to the  $X$ -axis the match of a point lying on the line  $Y_1$  is constant must lie on the line  $Y_2 = Y_1$  in the image obtained by tilting the SEM stage by any angle.

**4.4.2 The uniqueness constraint.** The *uniqueness* constraint states that for each feature point in one image there can be at most one matching feature point in the other image. This constraint was originally suggested by Marr and Poggio (1979) in the form of the uniqueness of the three-dimensional position of a surface point at any one time.

In our algorithm we enforce the uniqueness constraint in two ways—the uniqueness of the match of a left image contour and the uniqueness of the match of a right image contour. We say that there is a possible match between two contours if and only if there exists at least one point on one contour that can possibly match a point in the other when we search for its match along the epipolar line direction. Figure 5 illustrates how the uniqueness of a left image contour's match constraint is enforced and shows the resulting optimization problem constraint. This illustration shows a left image contour that can match two right image contours where each match is represented by binary variables  $x_1$  and  $x_2$ .



**Figure 6.** Illustration of the feature point-ordering constraint. (a) Preservation of feature point-ordering between left and right images. Projections of point  $P$  is to the right of those of point  $P'$  in both images. (b) The order of the left and right image components of overlapping contour matches  $x_1$  and  $x_2$  are reversed in the two images, hence violating the feature point-ordering constraint. This violation is expressed in terms of the optimization problem constraint shown above.

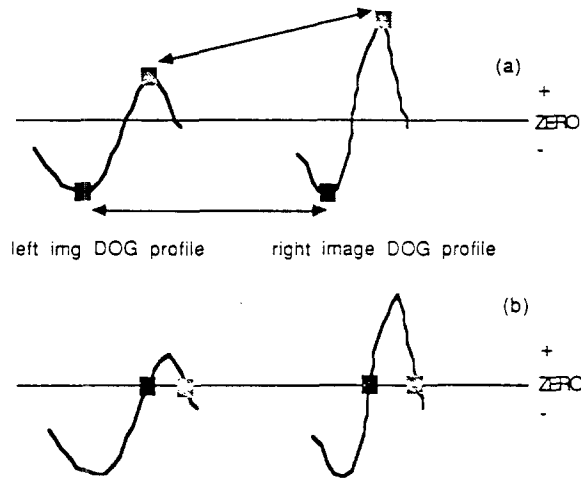
The constraint that is generated for the optimization problem states that at most one of these two matches can be accepted.

**4.4.3 The ordering constraint.** The preservation of *feature point ordering constraint* states that the ordering of feature points along an epipolar line in one image should be the same as the ordering of the corresponding match points along the epipolar line in the other image. That is, if an image point is to the right of another image point along an epipolar line in the left image, then its match should also be to the right of the match of the other point on the corresponding epipolar line in the right image. This is illustrated in Figure 6.

**4.4.4 The feature point similarity constraint.** The *feature point similarity constraint* states that the image characteristics should be similar in the vicinity of a matching pair of points in the two images. This constraint arises from the assumption that the difference between the viewing angles of the two cameras (for the SEM, the tilt angle difference) is small, and hence illumination conditions are nearly the same when the two images are obtained.

As mentioned earlier, we use two types of image features—high curvature points and edge points. It was shown that, based on the diffusion contrast mechanism, the brightness and contrast levels in the two images might be different, yet it is highly likely that intensity profiles along corresponding epipolar lines will have the same structure. Hence, it is highly likely to see corresponding high curvature points in the two images have the same type (sign) of curvature. That is, a locally convex high curvature section of the intensity profile will stay





**Figure 7.** Implementation of the feature point similarity constraint. (a) Feature point similarity for high curvature points: the type of curvature preserved. This is implemented by requiring matching feature points to have the same peak sign in the DOG filtered profiles. (b) Feature point similarity for edge points: the sign of the slope of the edge preserved. This is implemented by requiring matching zero-crossings to have the same slope sign in the DOG filtered image profiles.

locally convex in the corresponding intensity profile in the other image, and similarly for a locally concave section. This is illustrated in Figure 7a.

Similarly, based on the expectation that the structure of intensity profiles along corresponding epipolar lines will be the same, around corresponding edge points it is highly likely that the left and right image intensity profiles will have the same slope sign (see Figure 7b).

To summarize, if  $f_L(x)$  and  $f_R(x)$  are continuous functions approximating image intensity profiles along corresponding left and right epipolar lines, and  $x_e$  and  $x'_e$  ( $x_h$  and  $x'_h$ ) are corresponding edge (high curvature) points along these profiles, then we require that

$$\frac{df_L}{dx}(x_e) \cdot \frac{df_R}{dx}(x'_e) > 0 \quad (18)$$

$$\frac{d^2f_L}{dx^2}(x_h) \cdot \frac{d^2f_R}{dx^2}(x'_h) > 0 \quad (19)$$

Note that unlike some other stereo algorithms (Baker and Binford 1981; Ohta and Kanade 1985), we are not using  $f_L$  and  $f_R$ , that is, image intensity values, directly in the feature point similarity criterion since there can be differences in brightness and contrast in the left and right stereo images.

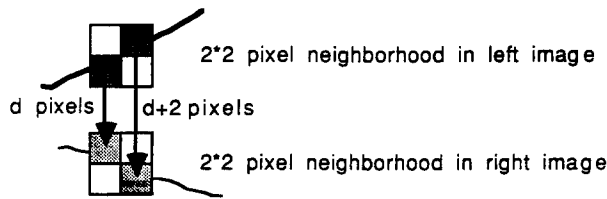
**4.4.5 The depth/height boundedness constraint.** The *depth/height boundedness constraint* states that computed depth/height (or disparity) values should be bounded. This constraint is based on the assumption that objects in the scene have finite depth. In cases where zero-crossings of  $\nabla^2 G$  filtered images are used in multiresolution stereo matching, after an initial coarse disparity map is used to align corresponding points in the two images, the size of the central excitatory region of the filter ( $w$ ) determines the disparity range over which matches will be sought. In our algorithm this constraint is enforced by limiting the search for the match of a point in the other image to only a section of the corresponding epipolar line.

**4.4.6 The surface smoothness and figural continuity constraints.** The *surface smoothness constraint* states that the computed three-dimensional surface points should not depict a surface that has abrupt jumps. This constraint comes from the observation that most physical objects have smooth surfaces when viewed at the proper resolution. In analyzing the surface of an integrated circuit pattern at very high magnifications, we do not expect to see a jump in the height values of two surface points that correspond to two neighboring pixel positions in the images. This constraint is violated at occlusion boundaries, but for the type of images we are dealing with we do not expect to run into this situation very often. This constraint was originally suggested by Marr and Poggio (1979).

The *figural continuity constraint* enforces smoothness of disparity along feature contours in the image (Mayhew and Frisby 1981). The assumption made here is that feature contours in the image are projections of a collection of points that are all on one smooth surface of an object; hence, disparity (or depth) should vary smoothly along the image contour.

In our algorithm the figural continuity and smoothness constraints were enforced as follows:

- *Figural continuity.* Our algorithm matches contours in one image to those in the other. Within the overlapping region of two matched contours the disparity will vary smoothly as we move along any of these two contours. Hence, due to the concept of *contour* matching that is employed, within such a region the figural continuity constraint is implicitly enforced. To be more specific, since neighboring points on our contours are eight-connected, two neighboring points on a left image contour that have been matched to two neighboring points on a right image contour cannot have



**Figure 8.** The disparity function will vary smoothly along two matched contours in the region where they overlap.

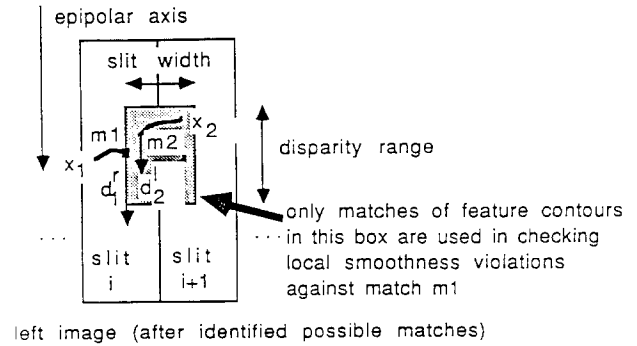
more than a 2 pixel difference in their disparities (see Figure 8),

- *Local smoothness.* Each left image contour's match is compared for "smoothness of disparity" violations against matches of other left image contours that are located within a box positioned immediately to the right of this left image contour and has dimensions equal to the disparity search range and the width of a slit (see Figure 9). Local smoothness violations are detected by checking if the disparity gradient between matches of the two left image contours exceeds a value of 2. In making this check, the two contour matches are represented by their end disparity vectors that face each other (i.e., that are closest to each other). In Figure 9, to determine if contour matches  $x_1$  and  $x_2$  violate the local smoothness constraint, the disparity vectors  $\mathbf{d}'_1$  and  $\mathbf{d}'_2$  (shown as arrows positioned at the proper left image contour point) are checked for a disparity gradient limit violation.

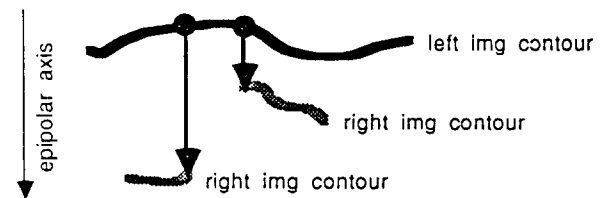
The major objective of the local smoothness constraint is to avoid situations where along a contour there is a sudden jump in disparity value due to two spatially well-separated contours along the epipolar line direction matching this contour over points that are very close to each other (see Figure 10). Note that aside from this check for local smoothness violations, no other (stricter) smoothness constraint was implemented. It is precisely due to strong smoothness assumptions that some of the other stereo algorithms have trouble around points where there is a disparity step such as occlusion boundaries.

## 5 Description of Algorithm

Assuming that the two stereo images are aligned in a direction orthogonal to the epipolar axis and are taken at the same magnification, our algorithm takes image profiles along each epipolar line in the left image and attempts to match its feature points to those found along the corresponding epipolar line in the right image. Note that each feature point detected along an epipolar line represents a single



**Figure 9.** Implementation of the local smoothness constraint. The two images are broken up into slits that extend along the epipolar line direction. Local smoothness is enforced by checking for disparity smoothness violations between each contour match  $x_1$  (with left image component  $m_1$ ) and contour matches with left image components falling within the box shown above.



**Figure 10.** Illustration of the need for the local smoothness constraint. A case where a long contour in the left image matches two contours in the right image that are spatially well separated from each other along the epipolar axis, over a close lateral distance.

point on an image feature contour. Figure 11 shows the block diagram of the stereo-matching algorithm. Our approach makes it possible to integrate constraint generators that extract scene information using other visual cues (such as shading, texture, etc.) into the system. These constraint generators can identify incompatible contour matches based on the physical principles that they are based on and generate appropriate optimization problem constraints. The constraint generators that are due to SEM imaging geometry, physics of SEM image formation, and the shape of the objects that are viewed have already been discussed in the previous section. In this section we discuss how the stereo-matching problem defined by these specifications can be transformed into an equivalent binary programming problem that can then be solved.

Each possible match of compatible left and right image contours is represented by a binary variable. The acceptance of a match is represented by having

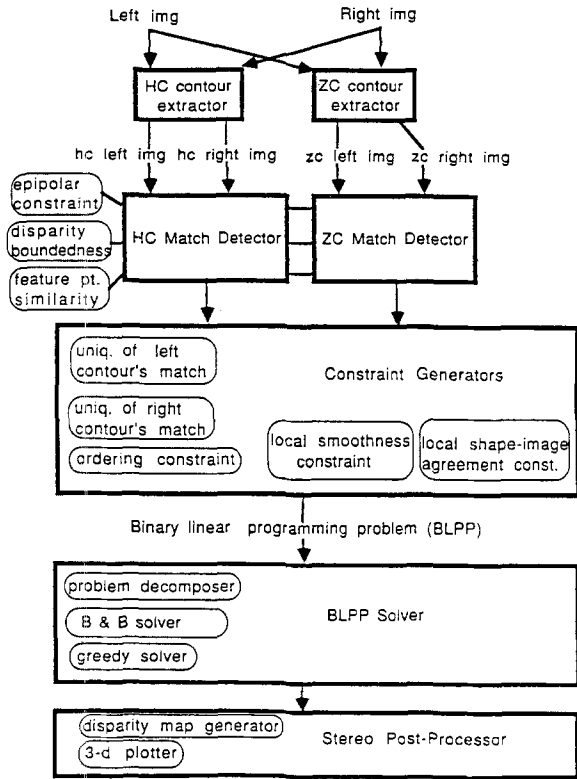


Figure 11. Block diagram of the SEM stereo system.

the binary variable corresponding to that match take on a value of one, whereas its rejection is represented by the variable taking on a value of zero. In considering the match of a left image contour to a right image contour, the number of points that are matched is used as the cost associated with that match. Figure 12 illustrates this idea. The objective function that is maximized is  $\sum p_i \cdot x_i$ , where  $p_i$  is the cost associated with accepting the contour match represented by  $x_i$ . This constraint is based on the expectation that, in general, a contour present in one image will also be present in the other will similar length, although it may have been fragmented, translated, and slightly rotated in going from one image to the other. Note that this condition will not be valid when a section of the scene appears in one image but not in the other due to occlusion. However, in the types of images that are encountered in our application we expect to see very few, if any, cases of occlusion. Since we view the stereo-matching problem as a global problem, the objective function used in our algorithm will be able to solve matching problems correctly in which local match ambiguities exist. Figure 13 illustrates such a case.

The constraint generators discussed previously result in a set of constraints that are in the form of

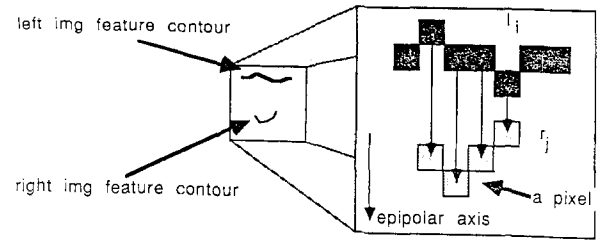


Figure 12. Cost associated with a contour match  $l_i - r_j$ . In this example if we were to accept this match 4 points on contour,  $l_i$  would be matched. Hence, the cost associated with this match is 4.

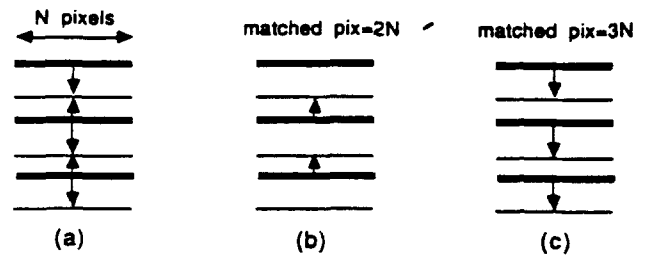


Figure 13. Illustration of how the objective function used resolves local match ambiguities. (a) Possible matches are displayed using arrows, where bold, solid lines denote left image contours and broken lines denote right image contours. Note each line is  $N$  pixels long. (b) One possible match, with an overall cost of  $2N$ . (c) The correct match picked by our algorithm with an overall cost of  $3N$ .

inequalities where the variables have a coefficient of 1 and the right-hand side is a 1 or 2 (i.e., they look like  $x_i + x_j + \dots \leq 1$  or 2).

We now present the steps of the algorithm:

- *Feature detection.* Each epipolar line in each image is convolved with one-dimensional Gaussian filters having  $\sigma = 4.0$  and  $6.4$ . The difference of Gaussian (DOG) profiles are obtained. High curvature points (HC features) and/or zero-crossing points (ZC features) are detected on each DOG profile in both images.
- *Contour detection.* A connected components algorithm is used to get HC and ZC feature contours.
- *Variable and cost of variable identification.* Based on the “similarity of feature points” and the “boundedness of disparity” constraints, possible matches between feature contours and their associated costs are identified. A binary variable is assigned to each such contour match.
- *Constraint generation.* Each of the constraint generators independently generates constraints between matches as discussed previously.

- *Solving the optimization problem.* The resulting binary linear programming problem is solved by decomposing it into subproblems (if possible) and solving each using a branch-and-bound algorithm presented in Hillier and Lieberman (1980).
- *Disparity to depth conversion.* Using Eqs. (3) to (5) and (13) to (15), the stereo-matching results are used to create a sparse depth map.

## 6 Experimental Results

In this section we present the results of running our algorithm on real data. These results are for two representative images. Several other test cases are presented in Kayaalp (1988).

The following experimental setup was used. Images were taken of silicon wafers provided by Hewlett Packard. The images were obtained through a JEOL-JSM-840 SEM. Typical settings used were an accelerating voltage of 10 kV, and a magnification of 10,000. Hard copies of the images were then digitized using a CCD camera into  $480 \times 480$  pixels. All the stereo experiments were carried out on the digitized images.

Figure 14a shows an image of a contact hole, obtained at a magnification of 10,000. The bar to the lower left, marked  $1 \mu\text{m}$  indicates that the bar measures one micron in length. Figures 14a and b show the images obtained by tilting the sample about a horizontal axis passing through the plane of the paper. The tilt angles used were  $15^\circ$  and  $30^\circ$  with respect to the original image. These will be termed the left and right stereo image pairs. Note that the apparent width of the edges changes as the tilt angle is increased. Figure 14c shows the result of running the stereo algorithm up to the stage of detecting matching contours. Both the left and right stereo images have been compressed laterally in order to fit into one image, as displayed. This explains the foreshortening seen along the horizontal axis, and has been done purely for display purposes. The vertical red bar indicates the disparity range within which searches for matches was constrained. High curvature and zero-crossing contours have been overlaid on the original image from which they were obtained. The epipolar axis is vertically oriented. Seven scan lines were used to generate the contours. The important point to note is that high curvature and zero-crossing contours in the left and right stereo images have been color coded. The color red is used to mark contours for which no match in the other image was found. Contours that were indeed matched are coded with the same color. Thus, a green contour in the left image is considered to be matched with a green contour in

the right image, provided both the contours lie within the disparity range indicated by the solid red bar.

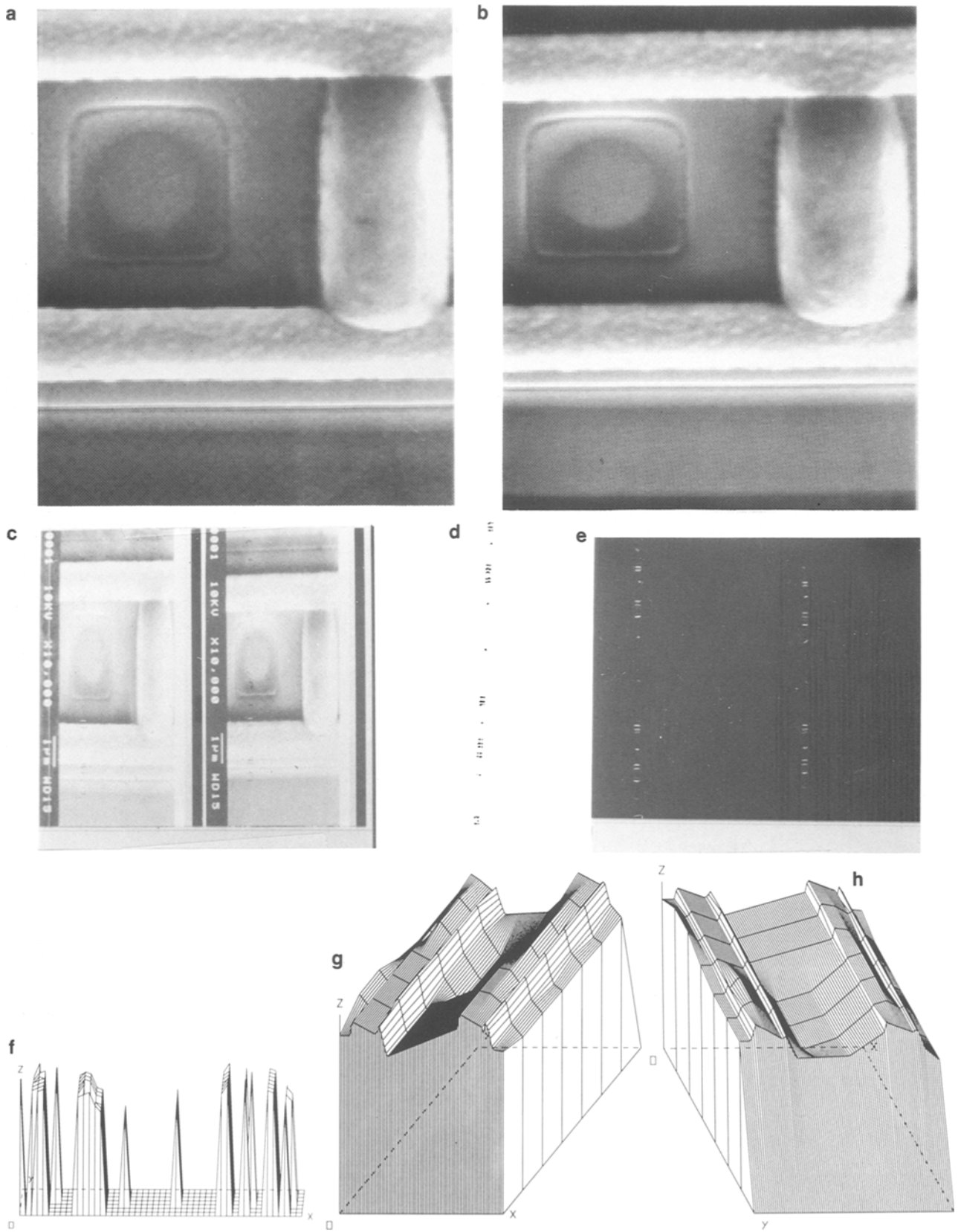
Since edge-based features are inherently fine, we have redisplayed them for the sake of clarity. Figure 14d shows only those contours in the left image that have been matched to contours in the right image. Figure 14e shows the color-coded contours separately, without overlaying them on the original image.

A close examination of these figures indicates that several features were matched between the two images, and these matches agree with our intuition. This verifies that the stereo algorithm does indeed pick out the correct matches.

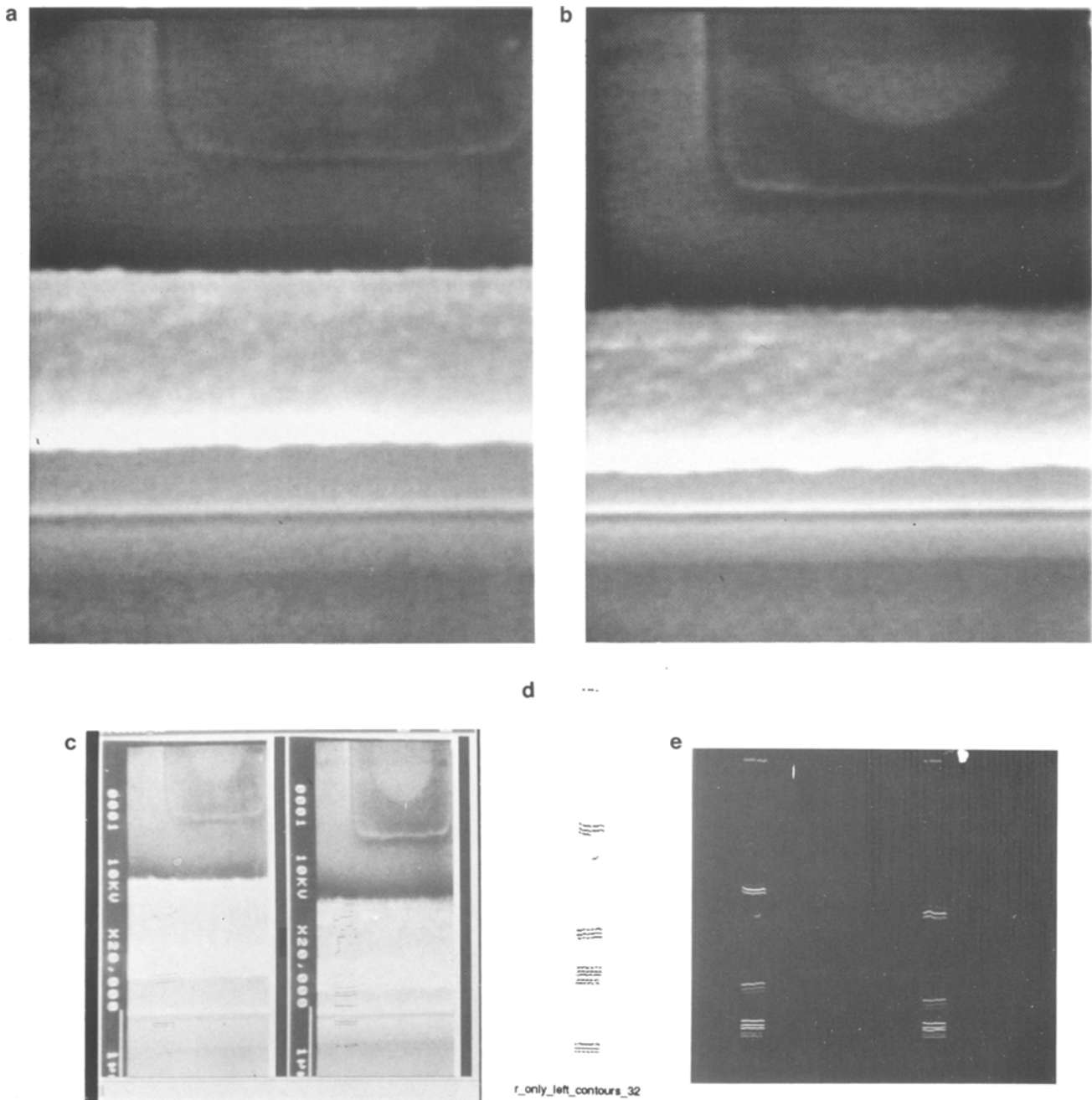
Now the next stage is to look at the reconstructed depth values. The algorithm is able to compute the depth values *only* along the contours that were matched. This results in a sparse depth map and is shown in Figure 14f. This is the final output obtained by the algorithm. In order to reconstruct the actual surface from these depth values, one needs to employ techniques from surface interpolation or surface approximation. However, this constitutes another different research topic altogether, which we have yet to address. Nevertheless, as a quick aid to view the reconstructed surface, one can simply perform a linear interpolation between the depth values actually determined by the algorithm. Of course, this is not entirely correct but is being done as an aid to visualization, and as a means of looking at the surface qualitatively. Figures 14g and h show the result of performing a linear interpolation between depth values *along* each scan line and the three-dimensional surface that results.

Figure 15a shows an image of a step, obtained at a magnification of 20,000. Figures 15a and b show the images obtained by tilting the sample about a horizontal axis passing through the plane of the paper. The tilt angles used were  $15^\circ$  and  $30^\circ$  with respect to the original image. Figure 15c shows the result of running the stereo algorithm up to the stage of detecting matching contours. The epipolar axis is vertically oriented. Thirty-two scan lines were used to generate the contours. Figure 15d shows only those contours in the left image that have been matched to contours in the right image. Figure 15e shows the color-coded contours separately, without overlaying them on the original image.

Again, a subjective examination of these figures indicates that several features were matched between the two images, and these matches agree with our intuition. This verifies that the stereo algorithm does indeed pick out the correct matches. The resulting sparse depth map is shown in Figure 15f.



**Figure 14.** (a) Contact hole viewed at a tilt angle of  $15^\circ$ . (b) Contact hole viewed at a tilt angle of  $30^\circ$ . The tilt axis is horizontal. (c) Result of the SEM stereo algorithm, up to the matching stage. (d) Contours in the left image that match contours in the right image. (e) Color-coded contours from both the left and right image. Contours coded in red indicate those contours for which no match was found. (f) Sparse depth map generated using the disparity values generated from the matched contours. (The scan lines are along the  $x$ -axis) (g) Three-dimensional relief map of the contact hole, obtained from a linear interpolation of the depth values along the scan lines (parallel to the  $X$ -axis). (h) The same reconstructed surface observed from a different viewing direction.



**Figure 15.** (a) The step viewed at a tilt angle of  $15^\circ$ . (b) The step viewed at a tilt angle of  $30^\circ$ . (c) Result of the SEM stereo algorithm, up to the matching stage. (d) Contours in the left image that match contours in the right image. (e) Color-coded contours from both the left and right image. Contours coded in red indicate those contours for which no match was found. (f) Sparse depth map generated using the disparity values generated from the matched contours. (g) Three-dimensional relief map of the step, obtained from a linear interpolation of the depth values along the scan lines. (h) The same reconstructed surface observed from a different viewing direction.

This is the final output obtained by the algorithm. Figures 15g and h show the result of performing a linear interpolation between depth values *along* each scan line and the three-dimensional surface that results.

## 7 Conclusions

In this paper we presented a novel technique to analyze stereo images generated from a SEM. The two main features of this technique are that it uses a

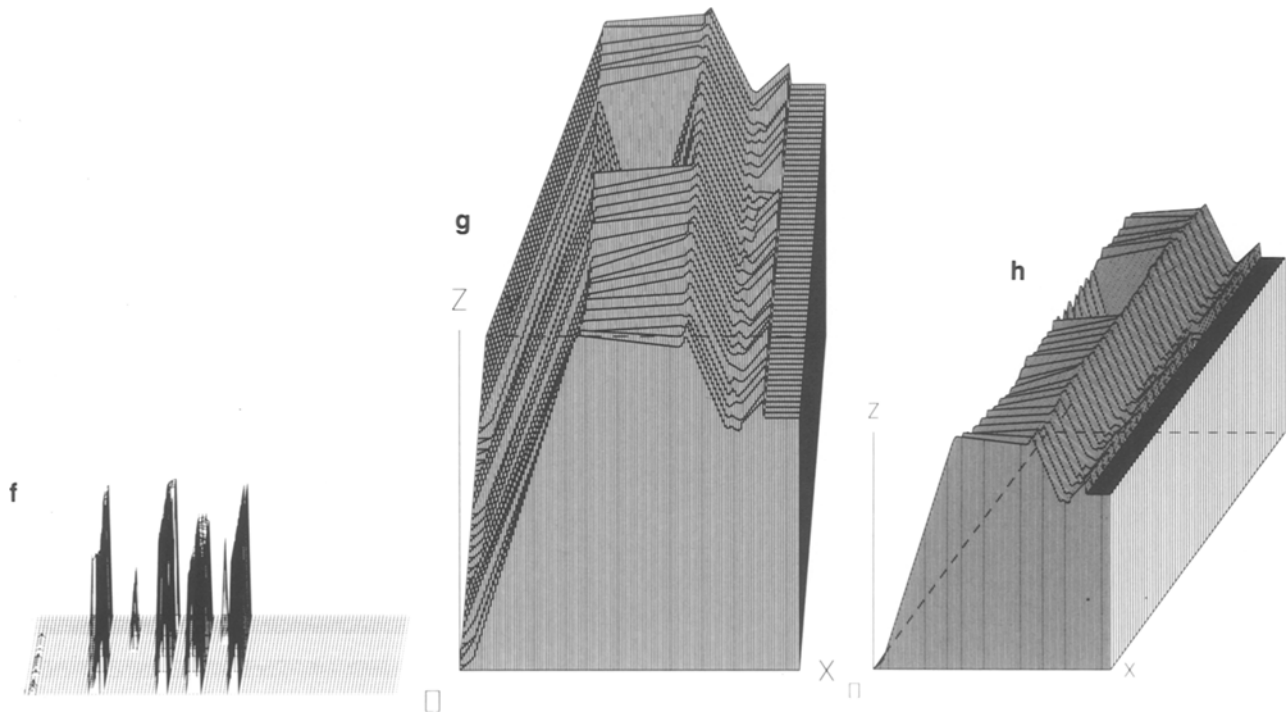


Figure 15. Continued.

binary linear programming approach to set up and solve the correspondence problem and that it uses constraints based on the physics of SEM image formation. Both these features are new in the field of computer vision.

We have applied the technique to a variety of images, some of which have been presented in this paper. The algorithm performs very well at detecting features, and also in finding matches for these features. The way we envision the use of such an algorithm is that it will act as a first step in the reconstruction of the actual physical surface under observation. The main limitation of the algorithm is that it generates only a sparse depth map. Additional points can be generated by making use of the knowledge of the types of surfaces that one is inspecting or by incorporating shape from shading techniques with these initial depth values as starting points. These are research directions that we will be pursuing in the future.

To sum up, we have investigated the feasibility of using SEM stereo in order to perform inspection of wafers for process control. We have determined that edge-based features are useful in the extraction of a sparse depth map in the case of SEM images. However, this data *alone* is not enough to obtain a complete description of the three-dimensional surface being inspected.

*Acknowledgments.* Support provided by the Semiconductor Research Corporation under contract number 86-07-085. We wish to thank the Semiconductor Research Corporation for funding this research, and Hewlett Packard for providing the wafers from which test images were obtained. We are also grateful to Professors Ken Wise and Mike Barnes for taking the SEM pictures and providing useful suggestions. We would also like to thank Pradeep Seneviratne for his comments on the drafts of this paper.

## References

- Baker HH, Binford TO (1981) Depth from edge and intensity based stereo. In: Proceedings of International Joint Conference on Artificial Intelligence-81, p. 631
- Balas E (1965) An additive algorithm for solving linear programs with zero-one variables. *Operations Research* 13:517–546
- Boyd A (1974a) Photogrammetry of stereo pair SEM images using separate measurements from the two images. SEM/I, Illinois Institute of Technology Research, Chicago, IL, p. 101
- Boyd A (1974b) Some practical applications of real time TV speed stereo SEM in hard tissue research. SEM/III, Illinois Institute of Technology Research, Chicago, IL, p. 109
- Breton BC, Thong JTL, Nixon WC (1987) A contactless 3-D measuring technique for IC inspection. *SPIE Inte-*

- grated Circuit Metrology, Inspection, and Process Control 775:109–117
- Goldstein JI et al. (1981) Scanning Electron microscopy and X-ray microanalysis. Plenum Press, New York
- Hilliard JE (1972) Quantitative analysis of scanning electron micrographs. *Journal of Microscopy*, February, p. 45
- Hillier FS, Lieberman GJ (1980) Introduction to operations research. Holden-Day Inc., Oakland, CA, pp. 725–729
- Horn BKP (1986) Robot vision. MIT Press/McGraw-Hill Book. Co., Cambridge, MA
- Howell PGT (1978) A theoretical approach to the errors in SEM photogrammetry. *Scanning* 1:118–124
- Kato Y et al. (1977) Stereoscopic observation and three dimensional measurement for scanning electron microscopy. SEM/I, Illinois Institute of Technology Research, Chicago, IL, p. 41
- Kayaalp, AE (1988) Automated visual inspection of integrated circuits using the SEM. PhD thesis, Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, MI
- Lane GS (1969) The application of stereographic techniques to the scanning electron microscope. *Journal of Physics E. Scientific Instruments* 2:565
- Larrabee GB (1977) The characterization of solid surfaces. SEM/I, Illinois Institute of Technology Research, Chicago, IL, p. 639.
- Lebiedzki J, White EW (1975) Multiple detector method for quantitative determination of microtopology in the SEM. SEM/I, Illinois Institute of Technology Research, Chicago, IL, p. 181
- Marr D, Poggio T (1979) A theory of human stereo vision. In: Proceedings of Royal Society of London B204:301–328
- Mayhew JEW, Frisby JP (1981) Psychophysical and computations studies towards a theory of human stereopsis. *Artificial Intelligence* 16:349–385
- Murty KG (1983) Linear programming. John Wiley, New York
- Nanometrics Inc. (1984) Cwicksan III specifications, Sunnyvale, CA
- Ohta Y, Kanade T (1985) Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 7:139–154, March
- Piazzesi G (1973) Photogrammetry with the scanning electron microscope. *Journal of Physics E. Scientific Instruments* 6:392
- Reimer L (1985) Scanning electron microscopy. Springer-Verlag, Berlin
- Rose M (1982) Masks and wafers: Linewidth measurements in a submicron industry. *Test & Measurement World* September, p. 30
- Singer PH (1983) Linewidth measurement approaching the submicron dimension. *Semiconductor International* March, p. 48
- Ullmann (1979) The visual interpretation of motion, MIT Press, Cambridge, MA
- The visual interpretation of motion (1979) MIT Press, Cambridge, MA
- Vorburger TV, Teague EC (1981) Optical techniques for on-line measurement of surface topography. *Precision Engineering* 3(2):61
- Woodham RJ (1978) Photometric stereo: A reflectance map technique for determining surface orientation from image intensity. In: Proceedings of 22nd International Symposium of SPIE, San Diego, CA, August, pp. 136–143