A Line Extraction Method for Automated SEM Inspection of VLSI Resist

D. B. SHU, C. C. LI, J. F. MANCUSO AND Y. N. SUN

Abstract—**This correspondence presents a precision digital edge line detection method. It is developed for extracting edge contours of resist lines of submicron width as imaged by scanning electron microscopy. The accurate determination of resist edge lines is crucial for automatic measurement, microregistration and automatic detection of defects on resist lines.**

Index Terms—**Digital line detection, Hough transform, SEM imaging, VLSI resist, wafer inspection.**

I. Introduction

Reliability control during integrated circuit fabrication is becoming more demanding as devices get smaller and denser. The fabrication of integrated circuits is a multistep process that includes developing photo resist patterns on wafers followed by etching, diffusion, oxidation, and implantation or metallization steps. Resist patterns used in device fabrication must have high integrity and conform to strict geometric specifications. Currently, wafers used in the manufacturing of IC's are electrically and visually inspected at various stages of fabrication, using optical microscopy, to ensure reliability of the circuits and to monitor the complex production process [1]–[6]. An optical-microscopy based automated reticle/mask inspection system has also become available [5]. Such inspection procedures and systems are feasible for patterns with linewidths greater than one micron. However, in-process inspection of submicron lines in VLSI (very large scale integration) wafers will require higher resolution than optical systems can deliver. Thus, scanning electron microscopy (SEM) and automatic image pattern analysis will be needed [7]–[11].

A SEM consists of an electron beam focusing and specimen scanning system with capability of synchronous specimen scanning, signal detection and display. Images obtained from a SEM are determined by the interaction of the scanning electron beam with the surface of the specimen and the stimulated emissions, either secondary emission or backscattered electron emission, received by the respective detectors. Contrast in a SEM image is normally due to specimen topology, chemical composition, as well as electric potentials. Relative to light microscopes the SEM has 10–100 times better resolution and 100–1000 times greater depth of field. Hence, it can resolve defects in submicron lines which cannot be resolved by light microscopes. Along with increased capabilities the SEM introduces new challenges for image processing because of the difficulties in contrast, noise, and edge properties relative to optical images.

In a VLSI resist pattern, straight line edges are the dominant features which are used for automatic measurement, microregistration and defect detection. Since the SEM imaging produces the blurring effect on edges of very fine resist lines, the accurate determination of resist edge lines is a critical problem. In this correspondence, we present a method for digital edge line extraction by a modification of the Hough transform.

Manuscript received December 15, 1986; revised May 15, 1987. This work was supported by the Ben Franklin Partnership Program of the Commonwealth of Pennsylvania and the Westinghouse Research and Development Center.

D. B. Shu was with the Department of Electrical Engineering, University of Pittsburgh, Pittsburgh, PA 15261. He is now with Hughes Research Laboratories, Malibu, CA 90265.

C. C. Li and Y. N. Sun are with the Department of Electrical Engineering, University of Pittsburgh, Pittsburgh, PA 15261.

J. F. Mancuso was with the Material Science Division, Westinghouse Research and Development Center, Pittsburgh, PA 15235. He is now with Electro-Scan Corporation, Danvers, MA 01923.

IEEE Log Number 8717229.

II. A Precision Digital Edge Line Detection Algorithm

Edge pixels in a field of view are first determined by an appropriate process. In our study, we considered SEM secondary emission images of resist on an aluminum film deposited on a silicon wafer. The incident electron beam energy caused some charging of the interior of the resist lines and blurring of the resist lines. We tried both Sobel gradient method and Laplacian–Gaussian method to determine edge pixels. Each field of view is digitized into 512 $\times$ 512 pixels of 256 gray levels. Sobel method gave either thick and noisy edges or broken edges, depending on the gradient threshold used. We chose to use the Laplacian–Gaussian operator which, under a proper choice of operator parameters, gave better results without shift in edge position. Fig. 1(a) illustrates a sample SEM image of resist lines. The spatial resolution of the digitization provides about 40 pixels per line width. Fig. 1(b) shows an experimental result of applying the Laplacian–Gaussian operator (with $w = 10$, where w is the diameter of the positive central region of the operator) to a sample SEM image as shown. Fig. 1(c) shows the resulting zero crossing points on the resist line contours as well as on small contours in the noisy background and those inside the resist due to the charging effect. The subsequent task is to extract the resist edge contours, which contain relatively long straight line segments, and remove the other unwanted zero crossing points.

A. Modification of the Conventional Hough Transform Procedure

Straight lines may be detected by the Hough transform [12]–[14]. In practice, because of the noisy edge region and the nature of the digital lines, false lines and aberration lines were often obtained [15]. Although the standard Hough transform has been successfully used for some disk head inspection [16], we did not find it satisfactory for our application. There are two major shortcomings in the conventional procedure of applying the Hough transform for the detection of connected edge segments. First, it fails to check the edge pixel connectivity. Second, the count in an accumulator cell does not necessarily reflect the length of a line segment even if those edge pixels are connected.

Let the parameterization of a straight line in the (x, y)-plane be specified by its normal distance ρ from the origin of the plane to the line and angular position θ of the normal; $\rho = x \cos \theta + y \sin \theta$. Each pixel (x_i, y_i) is transformed into a sinusoidal curve in (ρ, θ)-parameter plane. Let the (ρ, θ)-plane be quantized into a quadruled grid, and the quantized region be considered as a two-dimensional accumulator array. The peak count M in a (ρ_k, θ_k)-cell indicates M collinear pixels on a straight line in the image with its parameters ρ_k and θ_k. Let a $q \times q$ window be used for the transform and a count threshold M_t be specified for detecting digital line segments with length equal to or greater than M_t spatial units (containing at least M_t connected and digitally collinear edge pixels) within the window, where a spatial unit is defined by the interpixel distance. Let θ be quantized by $\Delta\theta$, for example, $\Delta\theta = 9°$, and ρ be quantized in the same manner as x and y. As illustrated in Fig. 2, the digital line A consisting of 19 connected pixels will be detected by the conventional Hough transform procedure if the count threshold is set at $M_t = 11$. Here, the count at the accumulator cell $(18, 45°)$ is 19, but this number does not reflect the actual length of the line which is $11\sqrt{2}$. On the other hand, an aberrant line B will also be detected because the 11 pixels marked by x, which are not completely connected, will give a count equal to the count

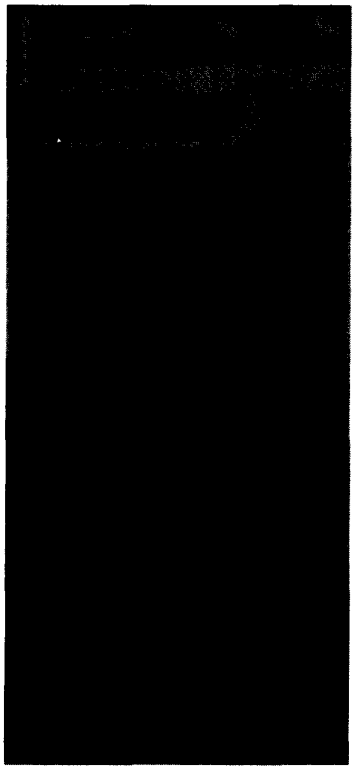

Fig. 1. An experimental result of applying the precision digital edge line detection algorithm to a sample SEM image of a resist pattern; (a) original SEM image (secondary electron image), (b) output of Laplacian–Gaussian filter ($w = 10$), (c) zero-crossings (slope map), and (d) detected edge contours.

threshold 11 at the accumulator cell (19, 36°). This is a false line consisting of three short line segments each of which has its length smaller than 11 units.

A new procedure has been developed to overcome these difficulties. First, it modifies the approach to increment the count in an accumulator cell by checking the content of an associated bit array of each cell, instead of increment unconditionally, such that the count indicates the actual length of the line segment if the pixels are all connected. Second, the threshold for the count in an accumulator cell is made adaptive to the orientation and location of the line segment under consideration (i.e., different threshold counts for different (ρ_k, θ_k)-cells). The procedure will be discussed in the following.

B. Construction of a Bit Array for Accumulator Cell

For each edge pixel (x_i, y_i), compute the values of $\rho_{ik} = \rho_i(\theta_k)$ at different values of $\theta_k = k\Delta\theta$, $\rho_{ik} = x_i \cos\theta_k + y_i \sin\theta_k$, where k varies from 0 to $(K - 1)$ and $K = \pi/\Delta\theta$. When each ρ_{ik} is rounded off, as designated by $\tilde{\rho}_{ik}$, the ordered pairs, $\{(\tilde{\rho}_{ik}, \theta_k) \mid k = 0, 1, \cdots, K - 1\}$, denote a set of accumulator cells for the edge pixel $P_i = (x_i, y_i)$. The pair (ρ_{ik}, θ_k), however, defines a line L_{ik} within a one-unit wide band $(\Delta\rho = 1)$ centered at the digital line parametrized by $(\tilde{\rho}_{ik}, \theta_k)$, as shown in Fig. 3. In the following discussion, the digital line $(\tilde{\rho}_{ik}, \theta_k)$, is referred to as the connected pixels within the band.

Let the normal from the origin to the line L_{ik} intersects with L_{ik} at a point (x_I, y_I), this point is termed the "normal intercept" of L_{ik}. Compute the distance $d_{i\theta_k}$ from the normal intercept (x_I, y_I) to the edge pixel (x_i, y_i), and round it off to an integer n to give a quantized $\tilde{d}_{i\theta_k} = n$ of the distance along the digital line specified by

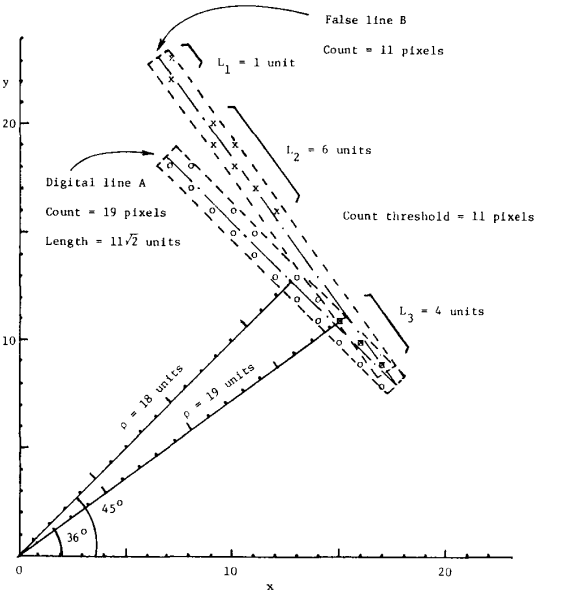

Fig. 2. Illustration of detection of digital line segments by the conventional Hough transform procedure.

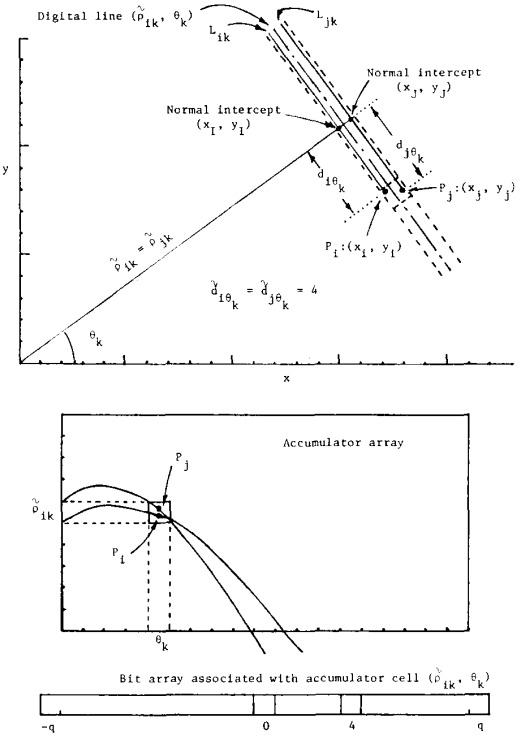

Fig. 3. Use of a bit array for controlling the count of accumulator cell to reflect the length of a digital line segment.

the parameters $(\tilde{\rho}_{ik}, \theta_k)$. Let us construct a bit array for each accumulator cell; the array has a length of $(2q + 1)$ with an index running from $-q$ through $+q$. The index "0" corresponds to the location of (x_I, y_I), that is, zero distance. Initially, all the bits in the bit array are set to 0. If $x_i > x_I$, the bit corresponding to the

index "$+n$" is examined. If the bit is 1, nothing will be done. If the bit is 0, it will be set to 1 and the accumulator cell $(\tilde{\rho}_{ik}, \theta_k)$ will be incremented by one. If $x_i < x_I$, the negatively indexed ("$-n$") bit will be examined. If $x_i = x_I$, then y_i in relation to y_I will be considered using either positively or negatively indexed bit. If there are m consecutive "1"s in the bit array, it represents an edge line segment of m units in length on the digital line. That is, each bit in the array contributes to the corresponding edge line one unit of length at a location whose index value reflects the quantized distance from the normal intercept of the line $(\tilde{\rho}_{ik}, \theta_k)$.

When a neighboring edge pixel $P_j = (x_j, y_j)$ is considered (see Fig. 3), for the same θ_k, the line L_{jk} passing through P_j will be parallel to L_{ik}. If $\tilde{\rho}_{jk} = \tilde{\rho}_{ik}$, these two lines lie within the quantized band of the digital line parametrized by $(\tilde{\rho}_{ik}, \theta_k)$. The distance $d_{j\theta_k}$ is computed similarly from P_j to the normal intercept (x_J, y_J). If $\tilde{d}_{j\theta_k} = \tilde{d}_{i\theta_k} = n$, P_i and P_j are considered at the same location as far as the digital line $(\tilde{\rho}_{ik}, \theta_k)$ is concerned. Therefore, P_j should not increase the count in the accumulator cell $(\tilde{\rho}_{ik}, \theta_k)$, otherwise, P_i and P_j would inappropriately contribute two units of length. Since the corresponding bit in the bit array has already been set by P_i, that bit and the accumulator count will not be updated. In this way, the count in each accumulator cell will reflect the actual length of the digital line segment in the processing window if the pixels are all connected.

Each accumulator cell has two parts. The first part is the count field for length measurement mentioned above. The second part contains the field for accumulation of the edge strength of pixels on the corresponding digital line segment (if the edge pixels are determined by the Laplacian–Gaussian edge operator, the slope magnitude at each zero-crossing point is used as its edge strength [17]).

The procedure described thus far may still fail to detect an edge line segment where the determined edge pixels are distributed as shown in Fig. 4(a), if a certain threshold is specified (such as 15 in this example) for the count in the corresponding accumulator cell. The following modification is incorporated to cope with such a situation. For each pixel P_i, the count incrementing process is expanded such that, in addition to increment the accumulator cell $(\tilde{\rho}_{ik}, \theta_k)$, the neighboring accumulator cells $(\tilde{\rho}_{ik} + 1, \theta_k)$ and $(\tilde{\rho}_{ik} - 1, \theta_k)$ are similarly incremented. Although the resulting counts in two neighboring accumulator cells shown in Fig. 4(b) are all greater than or equal to the specified threshold, the digital line $(\tilde{\rho}_{ik}, \theta_k)$ is detected for its maximum count.

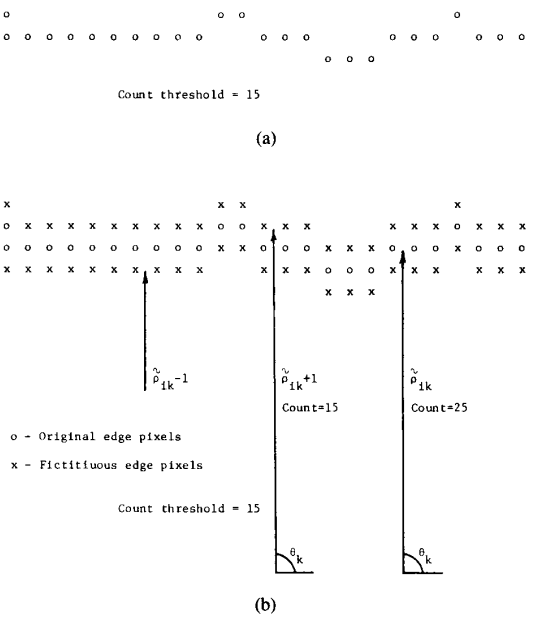

Fig. 4. A further modification of count increment process for neighboring accumulator cells.

C. Assured Connectivity of Pixels Through Adaptation of Count Threshold

In this study, we used a 31 × 31 processing window where the center pixel is referred to as the origin. It is intended to detect long digital line segments extended across the window. The number of connected pixels on each of these digital line segments and its length vary with its location and orientation. We consider those line segments in the window with length ranging from 29 up to 44. With $\Delta\theta = 9°$, θ_k indexed from θ_0 to θ_{19}, and $\tilde{\rho}_{ik}$ indexed from -15 to $+15$, there are 20 × 31 accumulator cells for the Hough transform. The full length of each of the corresponding 620 digital line segments can be computed, and the length threshold for detection of these line segments can be determined for individual accumulator cell. The length thresholds varying from 29 to 43 for longer digital line segments are entered into a table as shown in Table I, where the column and row positions of each entry are indexed by $\tilde{\rho}_{ik}$ and θ_k, respectively. Shorter line segments at other locations are not acceptable and their length thresholds are set at the highest level "124"; if any of them is a part of a longer line segment observable beyond the window, that segment will be detected when the window is moved 15 pixels to the right or downward. Thus the digital line segments of connected pixels, satisfying their individual length criteria, can be detected. For practical purposes, the adaptive threshold can be set at, say, 80 percent of the length values given in the table. It will allow at most 20 percent missing points in any detected line segment where pixel connectivity will be forced upon. The final counts in the accumulator array are examined against the adaptive threshold. Only those cells with counts equal to or greater than their respective thresholds retain their counts. Otherwise, counts will be ignored and reset to zero. In each processing window, the line detection process begins by scanning the accumulator array. Whenever a cell $(\tilde{\rho}_{ik}, \theta_k)$ with satisfactory count is met, a search is made for the maximum count in a subset of 3 × 3 cells with $(\tilde{\rho}_{ik}, \theta_k)$ at its top left corner. A digital line segment parameterized by $\tilde{\rho}_{rs}$ and θ_s is detected if the length count at cell $(\tilde{\rho}_{rs}, \theta_s)$ is the maximum. If equal maximum counts occur at two cells in the subset, the two corresponding line segments are compared for their respective edge strengths and the one with stronger edge strength will be chosen. Many aberration lines can thus be eliminated.

D. Extraction of Resist Line Contours

It has been observed that higher contrast in brightness always exists across a resist contour in comparison to the contrast across edge contours introduced by the charging effect within the resist region. An average contrast measure across an edge line segment is computed as the arithmetic average of the unnormalized contrast across each edge pixel on the line segment. This contrast is defined as the difference in the maximum gray level and the minimum gray level over an interval of ±5 pixels orthogonal to the edge line. A threshold value for the average contrast is chosen based upon the learning samples. The digital line segments with stronger contrast measures are finally detected as the edge line segments of a resist pattern. The processing window is moved by 15 pixels to the right in succession, and then is shifted downward by 15 pixels to repeat the same process, so that there are 50 percent overlap of every pair of processing windows. The detected line segments are thus linked together to form longer straight line edges of the resist image. An experimental result is shown in Fig. 1(d). The curved portions are traced out in the edge map (zero-crossing image in this case), by starting from the end points of each straight line edge, to form a complete edge contour of a resist line. Small protruding and convoluted portions on the contour are due to defects on the resist line.

TABLE I
ADAPTIVE COUNT THRESHOLD IN 20 × 31 ACCUMULATOR CELLS FOR LINE SEGMENTS OF 29-44 UNITS OF LENGTH WITHIN A 31 × 31 WINDOW

θ_k \ $\tilde{\rho}_{1k}$	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0°	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
9°	124	124	124	30	31	31	30	30	29	29	30	30	29	29	31	31	31	29	29	30	30	29	29	30	30	31	31	30	124	124	124
18°	124	124	124	124	124	124	33	34	33	33	34	33	33	34	33	33	33	34	33	33	34	33	33	34	33	124	124	124	124	124	124
27°	124	124	124	124	124	124	124	124	124	35	35	36	35	36	36	36	35	36	36	36	35	36	35	35	124	124	124	124	124	124	124
36°	124	124	124	124	124	124	124	124	124	35	37	38	39	40	40	40	39	40	40	39	38	37	35	124	124	124	124	124	124	124	124
45°	124	124	124	124	124	124	124	124	124	124	37	37	39	43	43	43	43	43	39	37	37	124	124	124	124	124	124	124	124	124	124
54°	124	124	124	124	124	124	124	124	124	35	37	38	39	40	40	40	39	40	40	39	38	37	35	124	124	124	124	124	124	124	124
63°	124	124	124	124	124	124	124	124	35	35	36	36	36	36	36	35	36	36	36	35	36	35	35	124	124	124	124	124	124	124	124
72°	124	124	124	124	124	124	33	34	33	33	34	33	33	34	33	33	33	34	33	33	34	33	33	34	124	124	124	124	124	124	124
81°	124	124	124	31	32	32	32	32	31	31	32	32	32	32	31	31	31	32	32	32	32	31	31	32	32	32	31	124	124	124	124
90°	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
99°	124	124	124	31	32	32	32	32	31	31	32	32	32	32	31	31	31	32	32	32	32	31	31	32	32	32	32	31	124	124	124
108°	124	124	124	124	124	124	33	34	33	33	34	33	33	34	33	33	33	34	33	33	34	33	33	34	33	124	124	124	124	124	124
117°	124	124	124	124	124	124	124	124	35	35	36	35	36	36	36	35	36	36	36	35	36	35	35	124	124	124	124	124	124	124	124
126°	124	124	124	124	124	124	124	124	124	35	37	38	39	40	40	39	40	40	39	38	37	35	124	124	124	124	124	124	124	124	124
135°	124	124	124	124	124	124	124	124	124	124	37	37	39	43	43	43	43	43	39	37	37	124	124	124	124	124	124	124	124	124	124
144°	124	124	124	124	124	124	124	124	124	35	37	38	39	40	40	39	40	40	39	38	37	35	124	124	124	124	124	124	124	124	124
153°	124	124	124	124	124	124	124	124	124	35	35	36	36	36	36	35	36	36	36	35	36	35	35	124	124	124	124	124	124	124	124
162°	124	124	124	124	124	124	33	34	33	33	34	33	33	34	33	33	33	34	33	33	34	33	33	34	33	124	124	124	124	124	124
171°	124	124	124	30	31	31	30	30	29	29	30	30	29	29	31	31	31	29	29	30	30	29	29	30	30	31	31	30	124	124	124

Row index refers to $\theta_k = k9^\circ$, (k = 0, 1, 2, ..., 19).

Column index refers to $\tilde{\rho}_{1k}$, varying from -15 to 15.

Count reflects the length of a digital line segment across a 31x31 window.

Count "124" refers to a short line segment with length less than 29.

IV. DISCUSSION

Our experiments have shown success of the above-described method of accurate detection of edge lines in SEM images of VLSI resist patterns. The determined edge lines in a test image can be used for measurement of resist line width and for microregistration with the corresponding lines of a golden image (a defect-free reference image) for defect detection. The microregistration can be performed by applying the unnormalized cross correlation of contrast method [4] to locate processing blocks which are to be inspected by the defect detection algorithms. There are three types of commonly observed defects: resist edge distortions (local swelling or local narrow-down of resist lines), bubbles or holes on resist lines, and stringers (residual resist material) on the sides of a resist line. The first two types are illustrated in Fig. 1(a); the third can be examined when the specimen is imaged in a tilted position. Some preliminary results on defect detection have been reported elsewhere [18]. With the very fine spatial resolution required for inspecting VLSI resist and with the practical constraints on data acquisition and image processing, one will have to sample only a few, say ten, positions on a wafer; this is also the practice being adopted in the current semiautomatic VLSI metrology. Within a sampled field of view, for example, a 10-micron square, the line detection process requires a large amount of computation and its processing time is a subject of serious concern. Toward this end, we have developed a systolic array processor architecture for this line detection method [19] so as to show its promise to be applicable for in-process inspection of VLSI resist.

REFERENCES

[1] M. L. Baird, "SIGHT-I: A computer vision system for automated IC chip manufacture," *IEEE Trans. Syst., Man., Cybern.*, vol. SMC-8: pp. 133-139, 1978.

[2] N. Goto, T. Kondo, K. Ichikawa, and M. Kanemoto, "An automated inspection system for mask patterns," in *Proc. 4th Int. Joint Conf. Pattern Recogn.*, Kyoto, Japan, 1978, pp. 970-974.

[3] Y. Hara, K. Okamoto, T. Hamada, and N. Akiyama, "Automatic visual inspection of LSI photomasks," in *Proc. 5th Int. Conf. Pattern Recogn.*, Miami Beach, FL, 1980, pp. 273-279.

[4] Y. Y. Hsieh and K. S. Fu, "An automated visual inspection system for integrated circuit chips," *Comput. Graphics Image Processing*, vol. 14, pp. 293-343, 1980.

[5] H. Harris, P. Sandland, and R. Singleton, "Wafer inspection automation: Current and future needs," *Solid State Technol.*, vol. 26, Aug. 1983.

[6] K. Okamoto, K. Nakahata, S. Aiuchi, M. Nomoto, Y. Hara and T. Hamada, "An automatic visual inspection system for LSI photomasks," in *Proc. 7th Int. Conf. Pattern Recog.*, Montreal, P.Q., Canada, 1984, pp. 1361-1364.

[7] D. Nyyssonen, "Calibration of optical systems for linewidth measurement on wafers," *Proc. SPIE*, vol. 221, pp. 119-126, 1980.

[8] S. Jensen, G. Hembree, J. Marchiando, and D. Swyt, "Quantitative submicrometer linewidth determination using electron microscopy," *Proc. SPIE*, vol. 275, pp. 100-108, 1981.

[9] R. A. Simpson and D. E. Davis, "Detecting submicron pattern defects on optical photomasks using an enhanced EL-3 E-Beam lithography tool," *Proc. SPIE*, vol. 334, pp. 230-238, 1982.

[10] S. Arunkumar and S. V. Reddy, "Automatic inspection of VLSI masks," in *Proc. 7th Int. Conf. Pattern Recogn.*, Montreal, P.Q., Canada, 1984, pp. 1352-1354.

[11] H. Yamaji, M. Miyoshi, M. Kano, and K. Okumura, "High accuracy and automatic measurement of the pattern linewidth on very large scale integrated circuits," *SEM*, vol. 1985/I, pp. 97-102, 1985.

[12] R. D. Duda and P. E. Hart, "Use of the Hough transform to detect lines and curves in pictures," *Commun. ACM*, vol. 15, pp. 11-15, 1975.

[13] T. M. Van Veen and F. C. A. Groen, "Discretization errors in the Hough transform," *Pattern Recognition*, vol. 14, pp. 137-148, 1981.

[14] C. M. Brown, "Inherent bias and noise in the Hough transform," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, pp. 493-505, 1983.

[15] S. Tamura, M. Uga, and T. Ono, "Straight line extraction for wafer alignment," in *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, Washington, 1983, pp. 285-290.

[16] D. Petkovic, J. Sanz, K. Mohiuddin, E. Hinkle, M. Flickner, C. Cox, and K. Wong, "An experimental system for disk head inspection," in *Proc. 8th Int. Conf. Pattern Recog.*, Paris, France, 1986, pp. 9-13.

[17] E. C. Hildreth, "The detection of intensity change by computer and biological vision systems," *Comput. Vision, Graphics, Image Processing*, vol. 22, pp. 1-27, 1983.

[18] C. C. Li, J. F. Mancuso, D. B. Shu, Y. N. Sun, and L. D. Roth, "A preliminary study of automated inspection of VLSI resist patterns," in *Proc. IEEE Int. Conf. Robotics and Automation*, St. Louis, MO, 1985, pp. 474-480.

[19] H. Y. H. Chuang and C. C. Li, "A systolic array processor for straight line detection by modified Hough transform," in *Proc. IEEE Comput. Soc. Workshop Comput. Architecture for Pattern Anal. and Image Database Management*, Miami Beach, FL, 1985, pp. 300-304.