Journal of Electronic Imaging 13(3), 428–435 (July 2004).

Content based segmentation of patterned wafers

Pierrick Bourgeat
Oak Ridge National Laboratory
P.O. Box 2008
Oak Ridge, Tennessee 37831-6011
and
University of Burgundy
Le2i Laboratory
12 rue de la fonderie
71200 Le Creusot, France

Fabrice Meriaudeau
University of Burgundy
Le2i Laboratory
12 rue de la fonderie
71200 Le Creusot, France

Kenneth W. Tobin
Oak Ridge National Laboratory
P.O. Box 2008
Oak Ridge, Tennessee 37831-6011

Patrick Gorria
University of Burgundy
Le2i Laboratory
12 rue de la fonderie
71200 Le Creusot, France

Abstract. *We extend our previous work on the image segmentation of electronic structures on patterned wafers to improve the defect detection process on optical inspection tools. Die-to-die wafer inspection is based on the comparison of the same area on two neighboring dies. The dissimilarities between the images are a result of defects in this area of one of the dies. The noise level can vary from one structure to the other, within the same image. Therefore, segmentation is required to create a mask and apply an optimal threshold in each region. Contrast variation on the texture can affect the response of the parameters used for the segmentation. We show a method to anticipate these variations with a limited number of training samples, and modify the classifier accordingly to improve the segmentation results.* © 2004 SPIE and IS&T.
[DOI: 10.1117/1.1762518]

1 Introduction

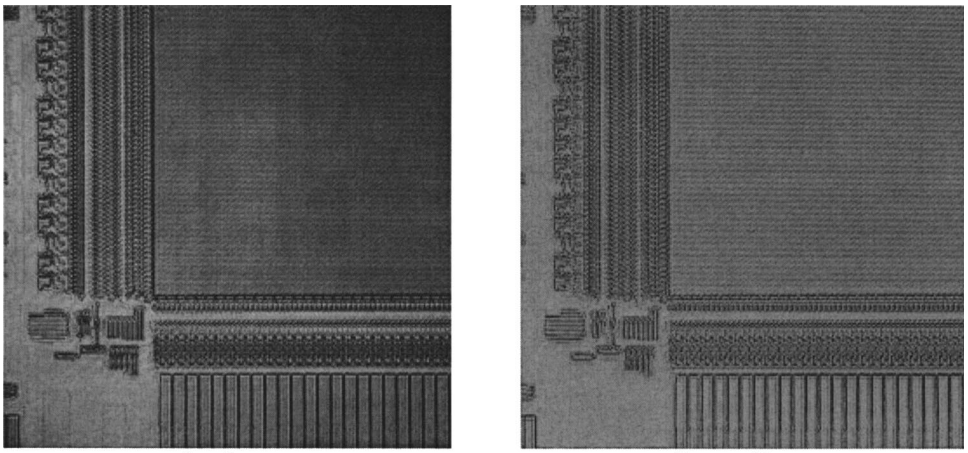
As semiconductor device density and wafer area continue to increase, faster and more sensitive automatic inspection tools are required. The size of the defects is becoming smaller, and harder to detect.[1–4] This paper introduces an improvement of our previous work[5,6] on the image segmentation of electronic structures on patterned wafers to improve the defect detection process on optical inspection tools.

Die-to-die wafer inspection is based on the comparison of the same area on two neighboring dies using the assumption that they are identical, except for the defects. The dissimilarities between the images are a result of defects in this area on one of the dies. The two images are subtracted, and a threshold level is selected to locate any abnormality. This threshold is established on the noise level in the difference image, to improve the SNR. The noise level can vary from one structure to another within the same image, since multiple structures coexist in the field of view. Therefore, the measure of noise within the whole image is not relevant for each individual type of structure. Segmentation is required to create a mask of these different regions. This mask is then used to produce a measure of noise for each structure in the difference image, and we apply an individual threshold in each region.

For this work, segmentation is performed using the discrete wavelet transform[7] and the ''à trous'' algorithm.[8–10] This algorithm is well adapted to discriminate local frequencies of the repetitive pattern, and it is restricted to principal directions that correspond to the geometric patterns found on integrated circuits. The weakness of this

Paper ORNL-015 received Aug. 6, 2003; accepted for publication Feb. 27, 2004.
1017-9909/2004/$15.00 © 2004 SPIE and IS&T.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms



Raw Image Corrected image

Fig. 1 Image correction with (a) the raw image and (b) the corrected image.

method is its sensitivity to contrast variation and small texture variation. In our previous work,[5] a local correction was applied to remove the nonuniformities. This is sufficient in the case of small variations. However, in some cases where the variations become very important, such as with large process variation or bad focus selection, the classifier must be trained with many different samples that cover all the variations contained within the die.

The usual way to train a classifier on this type of data is an empirical approach. The classifier is trained with randomly selected samples, and then is tested over the whole set of data. The areas where the classifier performs poorly are used to extract new training samples. These new samples are added to the original set to retrain the classifier until the best performances are obtained. This method is not realistic in the in-line inspection process when dealing with a huge amount of data. It would require storing all the images of a die to process them off-line, and therefore a huge amount of memory would be necessary. Meanwhile, it is time consuming for an operator to go through the iterative cycle of training the classifier and testing its performances until it becomes acceptable. This paper introduces an original method to anticipate the impact of the variation on each feature and to modify the classifier consequently to accommodate these variations. First, we discuss the correction applied to the images to remove the nonuniformity induced by the imaging system and the semiconductor process variations, followed by feature selections using the wavelet transform and the ''à trous'' algorithm. Next, the stress polytopes classifier[11,12] is described as well as the modifications introduced to correct the variations.

2 Image Correction and Feature Extraction

The selection of good features is one of the most important parts of the segmentation work. The wavelet transform produces good discrimination between the different structures, but is also extremely sensitive to the variations induced by the imaging system and the process variations. Therefore, it is very important to correct the images before the feature extraction, to remove these variations as much as possible.

2.1 *Image Preprocessing*

Bright field illumination used in the imaging system produces a Gaussian-type illumination, where the image is brighter in its center and darker near the borders. This nonuniform illumination creates contrast variations over the image that can change the wavelet response. The texture of the electronic structures is also affected by focus changes and process variation over the wafer. When small enough, they produce a contrast variation that can be partially corrected.

The normalization is performed using the statistics in the neighborhood of each pixel. The mean value is used to correct the illumination, and the standard deviation is used to partially correct the contrast variation. Each pixel is divided by the sum of the mean and the standard deviation values in a 5×5 sliding window. The illumination variation as well as some of the contrast variation is removed (Fig. 1).

2.2 *Wavelet Transform*

Once the images are normalized, the features can be extracted. The wavelet transform[7] is based on horizontal and vertical filtering that is well designed for the geometric structures found on semiconductor devices. It enables an efficient discrimination of the horizontal and vertical local frequencies.

The ''à trous'' algorithm is used to process a fast wavelet transform that is translation invariant.[8–10] Unlike the classical wavelet decomposition, where the image is subsampled at each decomposition level, the ''à trous'' algorithm works with a constant image size, and the filters' kernel is up-sampled by adding zeros between the coefficients. It creates an overcomplete decomposition that leads to some redundancy in the information, but also an invariance in translation, which is a requirement in this application. The image size is kept constant during the whole decomposition, and there is a good spatial localization of low frequencies.

Two filters are necessary to perform the wavelet decomposition: a low-pass filter (LP) and a high-pass filter (HP).

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

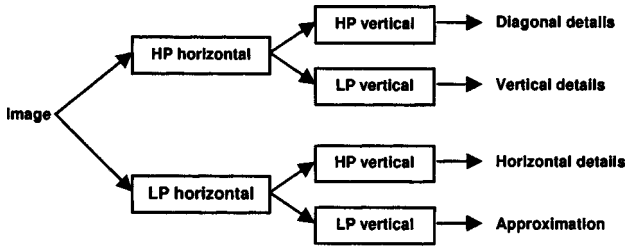


Fig. 2 Wavelet decomposition.

Coifman's orthonormal filters are used.[13] Figure 2 shows one level of the wavelet decomposition.

Between each decomposition level, the filters are upsampled to match different frequencies. Practically, for each level, $2^{\text{level}}-1$ zeros are inserted between each coefficient (Fig. 3). Thus, the number of meaningful coefficients required to convolve with the image is kept constant during the whole decomposition, so the processing time is almost the same for each level, enabling fast wavelet decomposition.

Semiconductor images are mostly a composite of geometric structures with horizontal or vertical orientation. For that reason, only horizontal and vertical details are kept for the segmentation. Diagonal details are ignored, because they are not good discriminating features, and they carry most of the noise. Three levels of decomposition are used to produce six different features.

Because of their waveform, the wavelet coefficients cannot be used directly in a classifier. That is why the local estimate of the wavelet standard deviation is used in a 17×17 neighborhood as a texture feature. This is a good way to smooth the wavelet response and obtain a uniform response when a frequency is matched, which is more suitable for the classifier.

3 Classification

The classification is performed using the stress-polytopes classifier.[11,12] This is a statistical classifier designed for high-data-flow segmentation. The parameter space is clustered in a small set of hypercubes called stress-polytopes, where each hypercube contains samples of a single class. Thus, the classification of a new point is performed by comparing its features with the boundaries of the hypercubes to verify a membership relation. When the set of hypercubes is small, this is a very fast way to perform the classification.

3.1 *Training 1*

Once the features are extracted, each training point is associated with 6-D vector. These vectors form the input to the classifier for the training. The 6-D parameter space is sliced

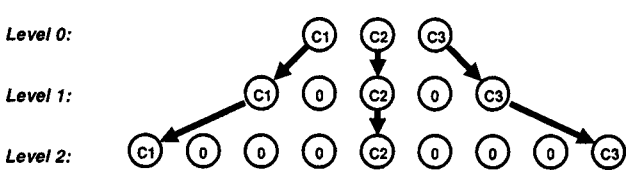


Fig. 3 Wavelets kernels for three-level decomposition.

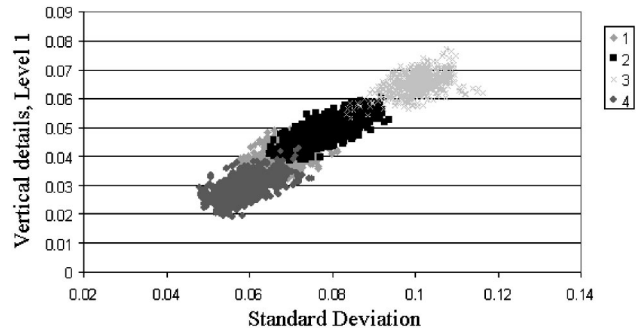


Fig. 4 Representation of one parameter versus the standard deviation corresponding to four samples of the same texture taken on four different images.

in hypercubes corresponding to the different classes. A hypercube is created around each training point, so that it only includes points of the same class. This is performed using the distance to the closest neighbor of dissimilar class for each side of the hypercube. The distance $l(\mathbf{X},\mathbf{Y})$ between two vectors $\mathbf{X}(x_1,\ldots,x_N)$ and $\mathbf{Y}(y_1,\ldots,y_N)$ is measured using the L-infinite norm, also known as maximum distance:

$$l(\mathbf{X},\mathbf{Y})=\|\mathbf{X}-\mathbf{Y}\|_\infty=\max_{1\leqslant p\leqslant N}|x_p-y_p| \tag{1}$$

For each training point, the distance l is measured with every other point of dissimilar class. In each direction, the shortest distances set the boundaries of the hypercube. In N-D space, $2N$ boundaries are enough to define a hypercube. To prevent any overlap between hypercubes of different classes, these distances are multiplied by a coefficient R that must satisfy the condition

$$0<R<\tfrac{1}{2}. \tag{2}$$

The hypercubes are fully described once there are two boundaries in each dimension defined by a neighbor in each direction. This means that with N parameters, there are $2N$ directions in which to look for a neighbor. If no neighbor is found in one direction but there is a closest neighbor in the opposite direction, then this distance is used in both directions. Otherwise, if there are neighbors in one dimension, then the shortest distance in every other dimension is used; as a result, limits cannot go to the infinite and saturate the

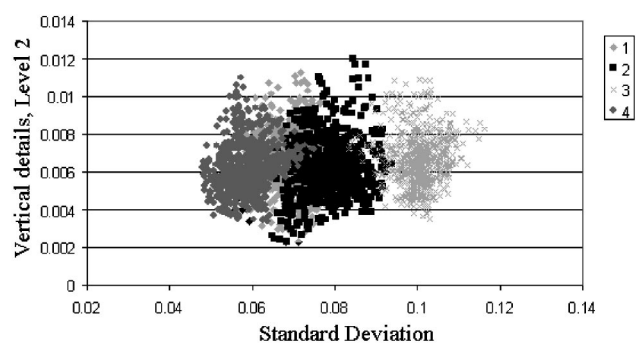


Fig. 5 Representation of one parameter versus the standard deviation corresponding to four samples of the same texture taken on four different images.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

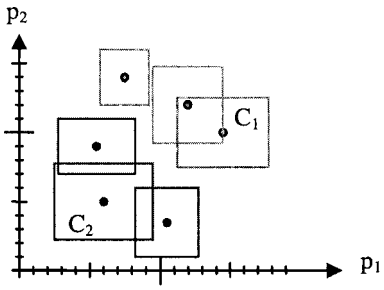


Fig. 6 Define the hypercubes using the distance to the neighbor of a different class.

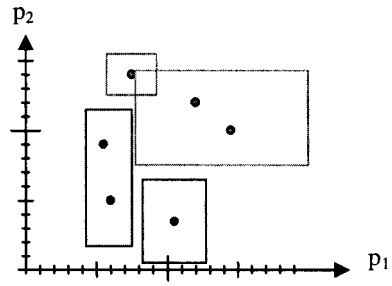


Fig. 8 Merge the hypercubes within each class.

parameter space. Using Eqs. (1) and (2) guarantees that there is only one class per hypercube, and there is no overlap between the hypercubes of different classes. This results in clustering the parameter space in hypercubes with fixed boundaries.

Dealing with fixed boundaries in the classifier enables processing the data very quickly, but it removes any flexibility. It is not a real problem when the parameter response is consistent for a given texture, but in most applications, unexpected variations in the parameter response can affect the performance. This is usually resolved by increasing the number of training samples and going through a cycle of retraining and testing the classifier until the expected performance is reached. In the particular application of image segmentation of electronic structures, variation of the parameters can be correlated with the residual contrast variations that have not been completely corrected. In other words, for a given texture, some parameters will follow the contrast variation, but not all of them. It is technically difficult to train the classifier with a very large set of samples since there is no guarantee that the whole range of contrast variation will be represented. Therefore, a technique was developed to modify the hypercube's boundary according to the anticipated variations of the parameter for each class.

Since the contrast variation is produced by the variations induced by the system, the measure of correlation between the contrast and a particular feature is a good way to determine if the feature will be sensitive to focus and process variations. This measure will then be used to modify the boundaries of the hypercubes. When a parameter has a strong correlation with the standard deviation for a given class, it will be more likely to be subject to variations itself (Fig. 4). That is why the corresponding hypercube needs to be enlarged in the direction corresponding to this parameter to allow more variation of the parameter. Similarly, if the correlation is weak, the parameter will be less likely to be subject to variations (Fig. 5). Therefore, the corresponding hypercube can be reduced in the corresponding direction. Figures 4 and 5 show the difference of correlation with the standard deviation for two parameters on the same set of four different samples.

This correction can be accomplished by making R a function of the correlation factors, the parameters, and the classes. This must be done carefully to ensure that there is only one class per hypercube, and that no overlap exists between the hypercubes of different classes. Actually, for each parameter of each class, the expansion/retraction must be done according to the correlation coefficients of the neighbors of other classes. For the same parameter, the correlation coefficient can be extremely different from one class to the other one.

During the hypercube creation step, the class of the closest neighbor is recorded. Thus, the class of each neighbor in each dimension is known. The boundary between two neighboring points in one dimension is defined using the centroid of mass between the respective correlation coefficient.

Considering a vector $\mathbf{X}$ of class C_X, and a vector $\mathbf{Y}$ of class C_Y, Eq. (2) can be rewritten to fulfill the requirement with the corrected coefficient $R_C(x_p, y_p)$:

$$0 < R_C(x_p, y_p) + R_C(y_p, x_p) < 1, \forall p, \tag{3}$$

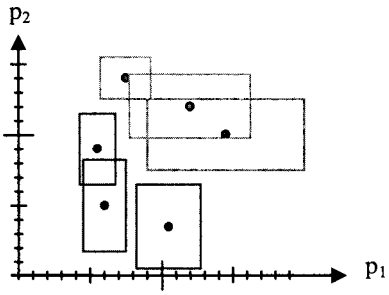


Fig. 7 Correct the hypercubes using the correlation information.

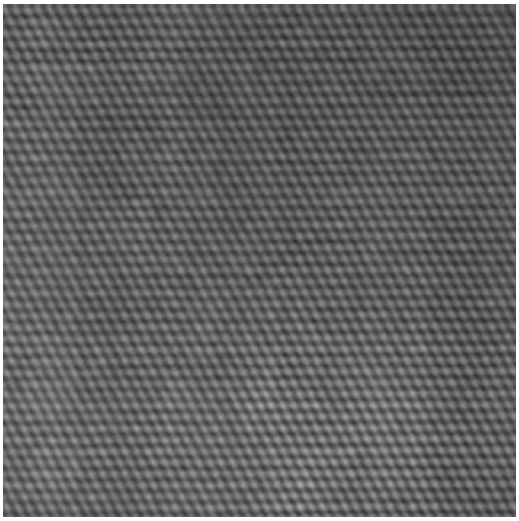


Fig. 9 DRAM area.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

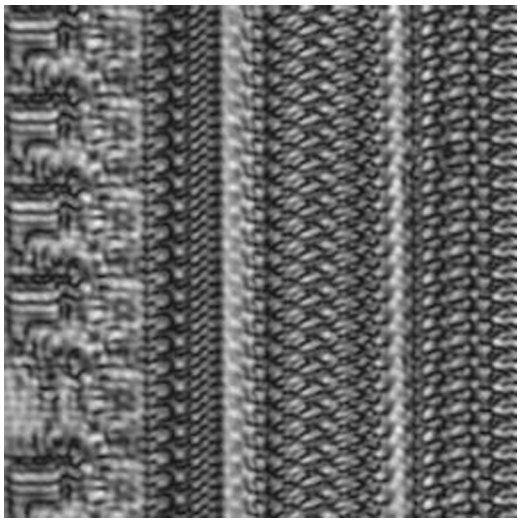


Fig. 10 Logic area.

where $R_C(x_p, y_p)$ is the coefficient for the parameter p of class C_X with its closest neighbor of class C_Y.

To perform the correction, a weight $W_C(w_{c_1},\ldots,w_{c_N})$ is assigned to each class. This weight is inversely proportional to the correlation with the standard deviation:

$$w_{c_P} = \frac{1}{|r_{c_P}|}, \tag{4}$$

where r_{c_P} is the correlation factor between the parameter p of class C and the standard deviation $(0<|r_{c_P}|<1)$.

Given a vector $\mathbf{X}$ of class C_X and a vector $\mathbf{Y}$ of class C_Y, for each parameter p, the distance ratio $d(x_p, y_p)$ between x_p and the centroid of x_p and y_p can be defined as

$$d(x_p, y_p) = \frac{w_{y_P}}{w_{x_P} + w_{y_P}}, \tag{5}$$

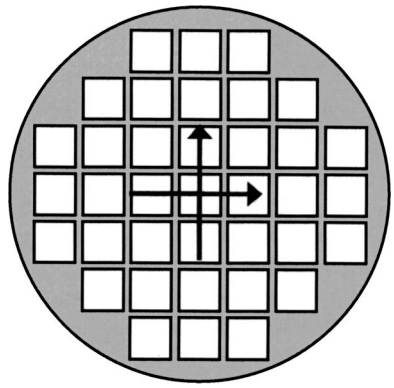


Fig. 12 Inspection layout.

$$\Rightarrow d(x_p, y_p) = \frac{|r_{x_P}|}{|r_{x_P}| + |r_{y_P}|}. \tag{6}$$

It is easy to show that

$$d(x_p, y_p) + d(y_p, x_p) = 1. \tag{7}$$

Requirement (3) can be fulfilled using Eq. (7) and R defined as in Eq. (2). The corrected coefficient $R_C(x_p, y_p)$ is described as

$$R_C(x_p, y_p) = 2Rd(x_p, y_p), \tag{8}$$

$$R_C(x_p, y_p) = 2R\frac{|r_{x_P}|}{|r_{x_P}| + |r_{y_P}|}. \tag{9}$$

The coefficient $R_C(x_p, y_p)$ provides a way to control the expansion/retraction of the hypercubes so that two hypercubes of different classes will not overlap. Any expansion in a given direction is followed by a proportional retraction in the same direction of the hypercubes corresponding to the neighboring points of another class. Since the corrections affect only the coefficient used to set the boundaries,

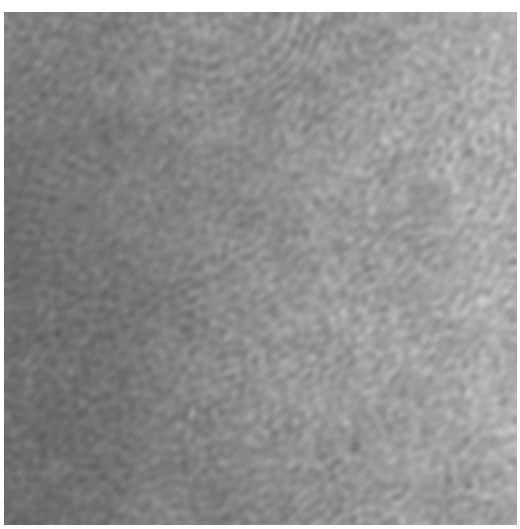


Fig. 11 Blank area.

Fig. 13 Example of the pixelwise segmentation on one image.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

Table 1 Misclassification rate on the segmentation on one column and one row.

	Column			Row		
Misclassification Rate	Training 1	Training 2	Training 1 and 2	Training 1	Training 2	Training 1 and 2
Without correction	2.45%	7.85%	2.29%	3.04%	2.97%	2.95%
With correction	2.32%	2.36%	2.49%	2.94%	2.69%	2.87%
Improvement	0.13%	5.49%	−0.20%	0.09%	0.29%	0.08%

the hypercubes will not be reduced to a point where they would exclude training points of the same class, or expanded to a point where they would include training points of another class.

Once all the hypercubes are created, they are merged inside each class. Two hypercubes are merged together if the newly created hypercube does not overlap with a hypercube of another class. This step reduces manageable number of hypercubes.

The full process is illustrated in Figs. 6, 7, and 8 for the two dimensional and two classes' case. First, the hypercubes are defined for both classes using Eq. (1) and Eq. (2), as shown in Fig. 6. Then in the case where the parameter p_1 of the class C_1 has a strong correlation with the contrast, but at the same time, the parameter p_1 of the class C_2 has a weak correlation with the contrast, hypercubes of class C_1 can be extended in the p_1 direction, while hypercubes of class C_2 can be retracted in the same direction (Fig. 7). Thus, the parameters of class C_1 can afford more variation in the p_1 direction. After this step, the hypercubes can be merged inside each class (Fig. 8).

3.2 *Segmentation*

During the classification process, each point is tested with the hypercubes to verify the membership relation. The hypercubes do not cluster in the whole space; therefore, some points do not belong to any hypercubes. In that case, an artificial point for each class, which corresponds to the mean of the training points of the same class, is used. The closest neighbor sets the class of the point.

4 Results

The segmentation tests were performed on a memory wafer, where three different areas must be segmented:

- the dynamic random access memory (DRAM) (Fig. 9) area, which is a fine regular texture that shows a lot of process variations
- the logic area (Fig. 10), which is a composite of coarse textures
- the blank area (Fig. 11), which does not contain any structure

Since the DRAM shows most of the variations, the test is

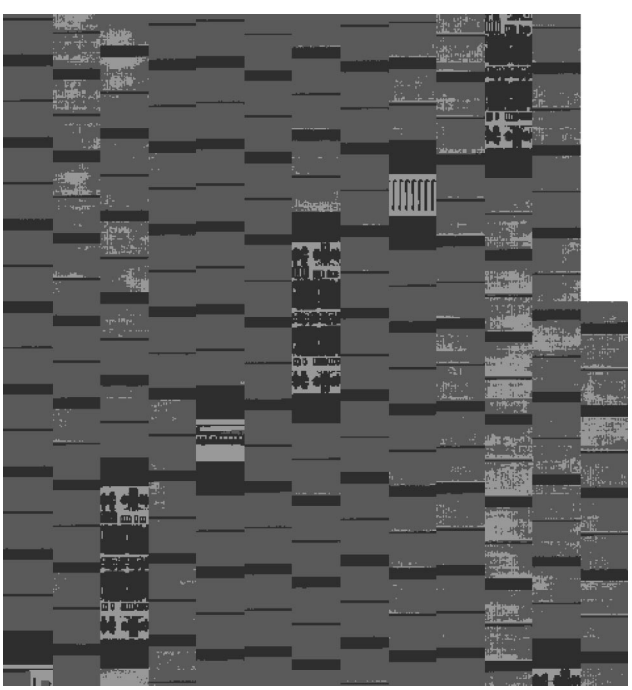


Fig. 14 Segmentation on one row without correction.

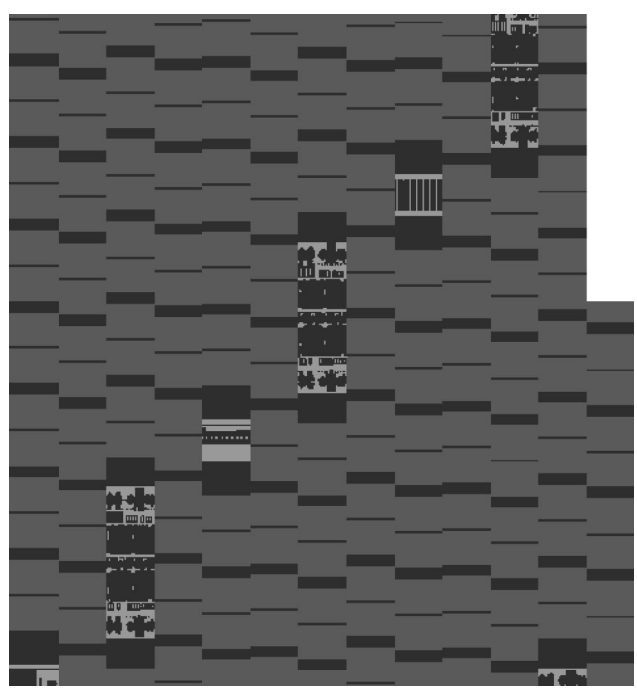


Fig. 15 Segmentation on one row with correction.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

Fig. 16 Segmentation on one column without correction.

done by comparing the results obtained with the same training sample for the logic and the flat area, but different samples for the DRAM in locations where the contrast is different. In each case, a single sample on the DRAM area is used. Two samples of the logic area are required to include two different types of structures and one sample of the blank area. Images for the segmentation come from a column of 170 images and a row of 170 images taken across three dies (Fig. 12) to include the maximum contrast variation. Each image is 472×472 pixels, and the segmentation is performed pixelwise (Fig. 13). The results for the two training sets with and without correction are presented in Table 1. Training sets 1 and 2 are also combined to provide a comparison.

These results show that without the hypercube correction and only one training sample in the DRAM area, the misclassification rate can vary depending on the training sample. The segmentation on the column is really dependent on the selection of the training set, whereas the segmentation on the row is more invariant because the contrast variations are less important. In both cases, the best results are achieved when combining the two training sets.

The correction on the first or the second training set improves the results in both cases, and the segmentation is more invariant with the selection of the training samples. With this segmentation, very good results are achieved (less than 3% of misclassification) with a few training samples. The results with one training set with the correction are as good as using a combination of two training sets without correction. Figures 14 and 15 show the improvement brought about by the correction on the column images using the second training set. Figures 16 and 17 show the slight improvement brought about by the correction on the row images using the second training set.

There are still limitations to the correction technique, especially when the variations are large or when the interclass distance is small. Furthermore, the correlation measure is accurate on uniform fine textures, but less accurate on nonuniform texture, where the correlation value is averaged. We also notice that on the column, the correction used on the combination of the first and second training sets slightly increases the misclassification rate. This is caused by the large difference between the two samples on the DRAM area. For some parameters, the two samples are so different that they do not overlap. Thus, the correlation value is dramatically increased, and the corresponding hypercubes can be overstretched. Nonetheless, misclassification is below 3%, which is adequate for this inspection process.

Fig. 17 Segmentation on one column with correction.

5 Conclusion

In wafer inspection, the performance of the segmentation is critical since the misclassification of an area can create a false detection or increase the overall noise level in the area, which would result in a higher threshold with the risk of missing a critical defect. It looks like a trivial problem since we deal with regular textures, but it is not since textures are subject to large variations that can be complex to correct. There are also limitations on the system flexibility to train the classifier since we have to work with huge data flow. We developed an original method to anticipate the variations with a limited number of training samples, thus increasing the training speed and the segmentation performance.

References

1. K. W. Tobin, ''Inspection in semiconductor manufacturing,'' in *Webster's Encyclopedia of Electrical and Electronic Engineering*, Vol. 10, pp. 242–262, Wiley & Sons, New York (1999).
2. K. W. Tobin and L. Neiberg, ''Metrology data management and information systems,'' in *Handbook of Silicon Semiconductor Metrology*, pp. 679–703, Marcel Dekker, New York (2001).
3. ''The national technology roadmap for semiconductors: technology requirement,'' Semiconductor Industry Association (2001).
4. J. Harrigan and M. Stoller, ''Automated wafer inspection in the manufacturing line,'' *Solid State Technol.* **34**(10), 69–72 (1991).

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms

5. P. Bourgeat, F. Meriaudeau, P. Gorria, and K. W. Tobin, ''Content-based segmentation of patterned wafer for automatic threshold determination,'' in *Machine Vision Applications in Industrial Inspection XI, Proc. SPIE* **5011**, 183–189 (2003).
6. P. Bourgeat, F. Meriaudeau, K. W. Tobin, and P. Gorria, ''Patterned wafer segmentation,'' in *Quality Control by Artificial Vision VI, Proc. SPIE* **5132**, 36–44 (2003).
7. M. Unser, ''Texture classification and segmentation using wavelet frames,'' *IEEE Trans. Image Process.* **4**(11), 1549–1560 (1995).
8. M. J. Shensa, ''The discrete wavelet transform: wedding the a trous and Mallat algorithms,'' *IEEE Trans. Signal Process.* **40**(10), 2464–2482 (1992).
9. P. Dutilleux ''An implementation of the algorithm à trous to compute the wavelet transform,'' in *Wavelets: Time Frequency Methods and Phase Space*, pp. 298–304, Springer, Berlin (1989).
10. M. Feil and A. Uhl, ''Real-time image analysis using MIMD parallel à trous wavelet algorithms,'' *Real-Time Imag.* **7**(6), 483–493 (2001).
11. J. Miteran, P. Gorria, and M. Robert, ''Classification géométrique par polytopes de contraintes. Performances et intégration,'' *Trait. Signal* **11**(5), 393–408 (1994).
12. J. Miteran, ''Performances et integration d'un algorithme de classification géométrique par apprentissage. Applications en traitement d'image,'' PhD Thesis, University of Burgundy (1994).
13. J. Tian, ''The mathematical theory and applications of biorthogonal Coifman wavelet systems,'' PhD Thesis, Rice University (1996).

Pierrick Bourgeat received his MS degree in 2001 from the University of Franche Comté, France, and is currently pursuing a PhD degree in image processing in collaboration with the Le2i Laboratory at the University of Burgundy, France, and the Image Science and Machine Vision Group at the Oak Ridge National Laboratory, U.S. He is interested in image segmentation and industrial inspection.

Fabrice Meriaudeau received his engineering degree in material science, his MS degree in physics in 1994, and his PhD degree in image processing in 1997 from the University of Burgundy. From June 1997 to September 1998, he was a postdoctoral fellow at the Oak Ridge National Laboratory working in the field of optical sensing and near-field optics. He is an associate professor and heads the research group (Le2i laboratory, UMR CNRS 5158) at the Technical Institute of Le Creusot, University of Burgundy, and his research interests include image segmentation, pattern recognition, multispectral, and IR imaging. He has authored or coauthored more than 90 papers on optical sensing and image processing. He chairs various conferences dedicated to machine vision.

Kenneth W. Tobin is a corporate research fellow and group leader of the Image Science and Machine Vision Group at the Oak Ridge National Laboratory, Oak Ridge, Tennessee. The group performs applied computer vision research and development in industrial inspection and metrology, biomedical imaging, and national security. He performs research in nondestructive test and analysis, image processing, and image-based metrology for automation and process characterization. He has authored and co-authored over 120 publications and he currently holds six U.S. Patents with five additional patents pending in the areas of computer vision, photonics, radiography, and microscopy. Dr. Tobin is a Fellow of SPIE—The International Society for Optical Engineering and a member of the Institute of Electrical and Electronics Engineers. He has a PhD in nuclear engineering from the University of Virginia, Charlottesville, Virginia, and an MS in nuclear engineering and a BS in physics from Virginia Tech., Blacksburg, Virginia.

Patrick Gorria received his engineering degree from the ENSAM, France, in 1980. From 1980 to 1982 he was in Mexico and received his PhD degree from the University of Burgundy, France, in 1984. Since 1993 he has been a professor at the Laboratory Le2i of Image Processing in the Institut Universitaire de Technologie (IUT) le Creusot, where he is engaged in research on and teaching of quality control by artificial vision. He has contributed to several areas of classification, sensors, and other systems and applications using artificial vision systems for quality control. He is currently a principal researcher at Le2i, where he conducts research and development of defects detection for quality control by artificial vision. Gorria holds several patents in the domain of image processing.

Downloaded from SPIE Digital Library on 24 Feb 2010 to 128.219.49.9. Terms of Use: http://spiedl.org/terms