Gabor filtering for feature extraction on complex images : application to defect detection on semiconductors

Pierrick Bourgeat[ab], Fabrice Meriaudeau[b&], Kenneth W. Tobin[a], Patrick Gorria[b]

[a]Oak Ridge National Laboratory, P.O.Box 2008, Oak Ridge, TN 37831-6010, USA
[b]Le2i Laboratory – Univ.of Burgundy, 12 rue de la fonderie, 71200 Le Creusot, France
& Corresponding author : Fabrice Meriaudeau, f.meriaudeau@iutlecreusot.u-bourgogne.Fr

ABSTRACT

This paper is an extension of 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 upon the comparison of the same area on two neighborhood dies. The dissimilarities between the images are a result of defects in this area of one of the die. The noise level can vary from one structure to the other, within the same image. Therefore, segmentation is needed 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. This paper shows a method to anticipate these variations with a limited number of training samples, and modify the classifier accordingly to improve the segmentation results.

Keywords: Wafer inspection, Gabor filters, segmentation, thresholding.

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 [1,2,3,4] .This paper introduces an improvement of our previous work [5,6,7] 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 upon the comparison of the same area on two neighborhood dies (Fig. 1), 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 die. The two images are subtracted, and a threshold level is selected to locate any abnormality. This threshold is established upon the noise level in the difference image, to improve the signal-to-noise ratio. The noise level can vary from one structure to the other, 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 needed to create a mask of these different regions (Fig. 2). This mask is then used to produce a measure of noise for each structure in the difference image, and apply an individual threshold in each region.

For this work where images are obtained by Direct Digital Holography[5], which enables recording the complex wave front of the surface wafer, segmentation is performed using a bank of Gabor filters[8,14] .
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. Due to its mathematical expression, Gabor filters can directly be applied to complex images (composed of a real and an imaginary part) obtained from digital holography[17]. In our previous work[5], a local correction was applied to remove the non-uniformities. This is sufficient in the case of small variations. However, in some cases where the variations become very important like large process variation or bad focus selection, the classifier needs to 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 processes when dealing with huge amounts 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 needed. Meanwhile, it is time consuming for an operator to go through the iterative cycle of training the classifier, and testing its performances until they become acceptable. This paper introduces an original method to anticipate the impact of the variation on each feature, and modify the classifier consequently to accommodate these variations. First, we present the method we used to extract the features using a bank of Gabor filters which in our case, leads to better results than wavelets decomposition[7]. Next, the stress polytopes classifier [9,10] is described, as well as the modifications introduced to correct the

variations. Then segmentation results are presented and discussed. The paper ends with a short conclusion which summarizes the obtained presents and presents some perspectives.

2. FEATURE EXTRACTION

The selection of appropriate features is one of the most important elements of segmentation work. The gabor transform produces good discrimination between the different structures, but is also extremely sensitive to the variations induced by the imaging system and process variations. Therefore, it can be necessary to correct the images before the feature extraction, to remove these variations as much as possible. [7]

2.1. Bank of Gabor Filters

The advantages of the phase image over the magnitude rely on the formation of those images. On one hand, the magnitude image is very sensitive to intensity variation on the wafer, to non-uniform illumination and more sensitive to noise in general. On the other hand, the phase image is not affected by color variation or non uniform illumination since it is a measure of the topography of the wafer that gives a greater immunity to noise. However, phase wraps constitute a major drawback that prevents the phase image from being directly used for segmentation. There have been several unwrapping algorithms developed [11], but they do not typically resolve abrupt height changes in the phase, and they are usually compute intensive.

With true holograms, since the complex information is contained in the original complex image (i.e., prior to phase unwrapping), it can be used directly to extract 3D information without dealing with phase unwrapping. With this type of information, features can be extracted using complex Gabor filtering [12,13] in Fourier space. The features are then a representation of both the amplitude and the phase of the signal, thus containing more information than processing just the amplitude of the signal, or just the phase. The Gabor filters are defined by their impulse response $h(x, y)$:

$$h(x,y) = g(x,y)\exp\left[j2\pi(Ux+Vy)\right]. \tag{1}$$

with

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left\{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2+\left(\frac{y}{\sigma_y}\right)^2\right]\right\}. \tag{2}$$

The $h(x, y)$ function is a complex sinusoid of frequency (U, V) with a gaussian envelop $g(x, y)$ of shape defined by (σ_x, σ_y). The Fourier transform of $h(x, y)$ is:

$$H(u,v) = G(u-U, v-V), \tag{3}$$

with

$$G(u,v) = \exp\left[-2\pi^2\sigma_x\sigma_y\left(u^2+v^2\right)\right], \tag{4}$$

the Fourier transform of $g(x, y)$.

The Fourier transform of the input image is multiplied by H(u, v), and transformed back into spatial coordinates. The amplitude of the resulting image is directly used as a feature for the

classifier. The bank of Gabor filters is oriented in the horizontal and vertical direction corresponding to the principal orientation of the structures on a semiconductor wafer. The filter bank layout is presented on Fig. 3.

Using complex images can be very helpful, especially when intensity or focus variations occur. Intensity variations will be visible in the magnitude image, and will affect the features produced by Gabor analysis of the magnitude image. Nonetheless it will not affect the phase image that only reflects the geometrical structure of the texture. Thus, features produced by a Gabor analysis on the complex image will not be as sensitive to intensity variation, and will provide a more uniform response. Focus changes are very obvious on the magnitude image, and dramatically change the features produced by Gabor filtering. The complex image is not as sensitive to focus changes if they stay within the wavelength depth, because part of the 3D information will still overlap between two different focus levels. Thus, the features produced by Gabor filtering on complex images are almost invariant to focus variations. For this study, we compared several algorithms to take better advantage of the complex information contained in the images.

2.1.a Normalized amplitude image

In this case, the phase of the signal is discarded, and the amplitude of the image $A(x,y)$ is normalized by dividing each pixel by the sum of the mean and the standard deviation values in an L×K sliding window (we usually set L=K=5). This case is used as a reference of the performances of the Gabor filtering on a real image.

$$A_N(x,y) = \frac{A(x,y)}{\mu_{L\times K}(x,y) + \sigma_{L\times K}(x,y)}. \tag{5}$$

2.1.b Phase image

The phase image $I_\varphi(x,y)$ is obtained by dividing the complex image $I(x,y)$ by the amplitude $A(x,y)$ of $I(x,y)$. In practice this is accomplished by dividing the real and imaginary part of the image by the amplitude.

$$I_\varphi(x,y) = \frac{I(x,y)}{A(x,y)} = e^{j\varphi(x,y)}. \tag{6}$$

We can evaluate the importance of the phase information on the image when no amplitude information is available and see its relevance to the segmentation process. Since we get a complex image, it can be directly transformed into Fourier space to be filtered by Gabor filters so that we do not have to deal with phase unwrapping.

2.1.c. Complex image

The complex image $I(x,y)$ is transformed in the Fourier space and directly used as input for the Gabor filters.

2.1.d . Complex image with normalized amplitude

In this case, we recreate a complex image $I_N(x,y)$ as a combination of the normalized amplitude image $A_N(x,y)$, and the phase image $I_\varphi(x,y)$.

$$I_N(x,y) = I_\varphi(x,y)\cdot A_N(x,y) = \frac{I(x,y)}{A(x,y)}\cdot A_N(x,y). \tag{7}$$

3. SEGMENTATION AND RESULTS

3.1. Classifier Training

Once the features are extracted, each training point is associated with a 6-dimentional vector (Fig . 4). These vectors form the input to the classifier for the training. The 6-dimentional parameter space is sliced into 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 $l(X, Y)$ between two vectors $X(x_1, \ldots, x_N)$ and $Y(y_1, \ldots, y_N)$ is measured using the L-infinite norm, also known as maximum distance:

$$l(X,Y) = \|X - Y\|_\infty = Max_{1 \le p \le N} |x_p - y_p| \tag{8}$$

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 dimensional space, $2N$ boundaries are enough to define a hypercube. To prevent any overlap between the hypercubes of different classes, these distances are multiplied by a coefficient R that must satisfy the condition :

$$0 < R < \frac{1}{2} \tag{9}$$

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 surrounding 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 parameter space. Using (8) and (9) 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 (Fig. 5).

Dealing with fixed boundaries in the classifier allows 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 application, unexpected variations in the parameter response can affect the performances. This is usually resolved by increasing the number of training samples, and go 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 (Fig. 8). It is technically difficult to train the classifier with a very large set of samples since there is no guaranty 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[7].

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 the manageable number of hypercubes (Fig. 6).

To include a maximum contrast variation, we scanned a column of 1728 images taken across an entire wafer, and covering 24 dies. Fifteen sets of training samples (Fig. 9) are selected within the column, to cover the whole range of fluctuations. For each training set, a single image is used to extract training samples of each type of texture. This way, we can evaluate the classifier performances when a single image that contains the three classes is used to train it. This operation is performed with each training set, and each set of features, to identify the best set of feature when using a single image to train the classifier.
Each image is 436 by 436 pixels, and the segmentation is performed pixelwise. The segmented images are placed in a mosaic of 24 columns, corresponding to the 24 dies.

3.2. Segmentation and Results

During the classification process, each point is tested with the hypercubes to verify the membership relation. The hypercubes do not cluster the whole space; therefore, some points do not belong to any hypercubes. In that case, a class prototype corresponding to the mean of the training points of the same class is used. The closest neighbor defines the class of the point (Fig. 7).

The segmentation tests were performed on a memory wafer, where 3 different areas where defined for segmentation

- the DRAM (Fig. 10) area which is a fine regular texture exhibiting significant process variations
- the logic area (Fig. 11) which is a composite of coarse textures
- the field area (Fig. 12) which does not contain any structure

Fig. 13 is an example of a pixelwise segmentation on one image, whereas Fig. 14 and 15 present some results obtained on the whole test basis.
Fig. 17 and Fig. 18 present some examples of the obtained misclassification rates versus the used training sample with or without correction of the classifier for different input images.

These results show that without the correction of the hypercubes and training sample extracted from a single image, the misclassification rate is highly dependant on the quality of training sample. When the training sample is less representative of the global standard deviation (higher training set number), the misclassification rate increases up to 10%. With the help of the hypercubes correction, the misclassification rate becomes less sensitive to the quality of the training set, and produces more uniform results, with a misclassification rate up to 6.5% for normalized amplitude images. Moreover, the correction is only applied when needed (only on certain resolution level of complex normalized amplitude images and normalized amplitude

images), and thus never significantly impair the classification. Using the complete training set to train the classifier leads to a misclassification rate of 3.8% for normalized amplitude images and 3.92% for complex normalized amplitude images, whereas respective averages of 4.5% for normalized amplitude images and 5.76% for complex normalized amplitude images were obtained using a single image for the training which confirms the usefulness of our approach.

However, the results show the importance of the phase information for the segmentation task, since the best results are achieved when the phase is used. Gabor filters on the normalized amplitude images give overall good results; however, the boundaries between two different textures are not very well resolved, with a smearing of the decode bars area over the DRAM area, and the features are correlated with the contrast variation, which causes the features of different classes to overlap. Therefore, few places of the DRAM area are classified as decode bars. Using Gabor on phase images gives good results, especially with a good segmentation at the boundary between different textures. There are only a few places where we can see some misclassification in the DRAM area. Using Gabor on complex images with normalized amplitude images gives the best results, with a very good segmentation at the boundaries. The combination of the different segmented images (phase, normalized amplitude…) should lead even better results which will be more independent of the training sample [14,15].

5. CONCLUSION

In wafer inspection, the performance of image segmentation is critical since the misclassification of an area can create a false detection, or increase the overall noise level in the area, that would result in a higher threshold with the risk of missing a critical defect. It appears to be a trivial problem since we deal with regular textures, but it is not since the textures are subject to big variations that can be difficult to correct. There are also limitations on the system flexibility to train the classifier since we have to work with data volumes and high data rates. To address these issues, we develop an original method to anticipate the variations with a limited number of training samples, thus increasing the training speed, and the segmentation performances. When applying this method to defect detection, an increase of 25% in the correct detections was observed and a significant reduction (greater than 300%) in the false detection rate was achieved.

REFERENCES

1. K. W. Tobin, “Inspection in Semiconductor Manufacturing”, Webster’s Encyclopedia of Electrical *and Electronic Engineering*, Wiley & Sons, New York, NY, 1999, **10**, 242-262.
2. K. W. Tobin, L. Neiberg, “Metrology Data Management and Information Systems”, Handbook of Silicon Semiconductor Metrology, Marcel Dekker, New York, NY, 2001, 679-703.
3. “The National Technology Roadmap for Semiconductors: Technology Requirement”, Semiconductor Industry Association , 2001.
4. J. Harrigan, M. Stoller, “Automated Wafer Inspection in the Manufacturing Line”, *Solid State Technology*, 1991, 34(**10**), 69-72.
5. P. Bourgeat, F. Meriaudeau, P. Gorria, K.W. Tobin, “Content-based segmentation of patterned wafer for automatic threshold determination”, Machine Vision Applications in Industrial Inspection XI 2003, Proc. SPIE Vol. 5011, 183-189.
6. P. Bourgeat, F. Meriaudeau, K.W. Tobin, P. Gorria, “Patterned wafer segmentation”, in QCAV’2003, Gatlinburg, USA, May 2003, Proc. SPIE Vol. 5132, 36-44.
7. P. Bourgeat, F. Meriaudeau, K.W. Tobin, P. Gorria , “Content based segmentation of patterned wafers”*, Journal of Electronic Imaging*, 2004, Vol 13, (**3**), 428-435.
8. S.E. Grigorescu, N. Petkov, P. Kruizinga, “Comparison of texture features based on Gabor filters*”, IEEE Trans. on Image Proc*., 2002, Vol. 11, **10**, 1160-1167.
9. J. Miteran, P. Gorria, M. Robert, "Classification géométrique par polytopes de contraintes. Performances et intégration", *Traitement du Signal*, 1994, 11(**5**), 393-408.
10. J. Miteran, Performances et integration d’un algorithme de classification géométrique par apprentissage. Applications en traitement d’image, PhD thesis, Universtity of Burgundy 1994.
11. H.A. Zebker, Y. Lu, “Phase unwrapping algorithms for radar interferometry: residue-cut, least-squares, and synthesis algorithms” *JOSA A*, 1998, Vol. 15, **3,** 586-590.
12. A.C. Bovik, M. Clark, W.S. Geisler, “Multichannel texture analysis using localized spatial filters”, *IEEE Trans. Pattern Anal. Machine Intell*., 1990, vol. 12, **1**, 55-73.
13. S.E. Grigorescu, N. Petkov, P. Kruizinga, “Comparison of texture features based on Gabor filters”, *IEEE Trans. on Image Proc*.,2002, Vol. 11, **10**, 1160-1167.
14. P. Bourgeat, F. Meriaudeau, K.W. Tobin, P. Gorria, “Features extraction on Complex images”, Proc. OSAV’2004, Int. Topical Meeting on Optical Sensing and Artificial Vision, Saint Petersburg, Russia, 18-21 October 2004, State University ITMO, pp 103-110.
15. P. Bourgeat, F. Meriaudeau, “Classifier combination for wafer segmentation”, Machine Vision Applications in Industrial Inspection XIII, San-Jose, USA, January 2005, SPIE Vol. 5679.
16. P. Bourgeat, F. Meriaudeau, K.W. Tobin, P. Gorria , “Classifier vote for wafer segmentation”, *submitted to Signal Processing.*
17. C. E. Thomas Jr., et Al. “Direct to digital holography for high aspect ratio inspection of semiconductor wafers.”, Austin, USA, March 2003 International Conference on Characterization and Metrology for ULSI Technology, Proc. AIP Vol. 683, 254-270.

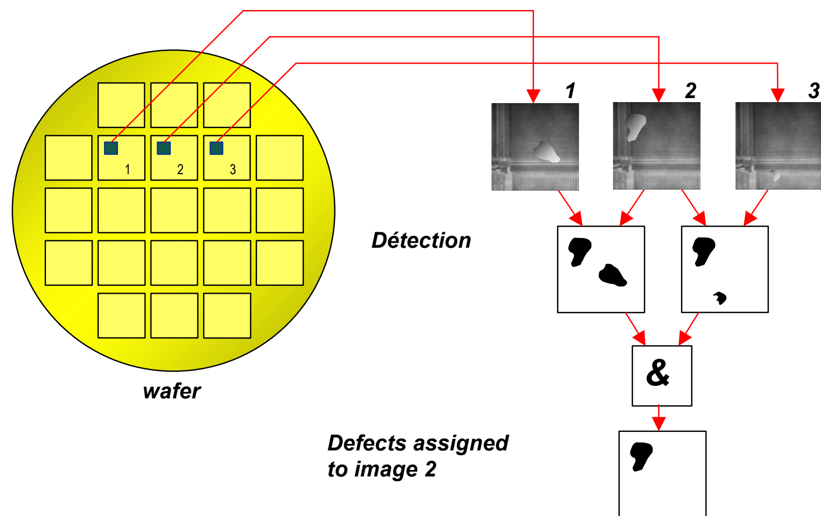

Fig. 1 : Principle of the die-to-die wafer inspection.

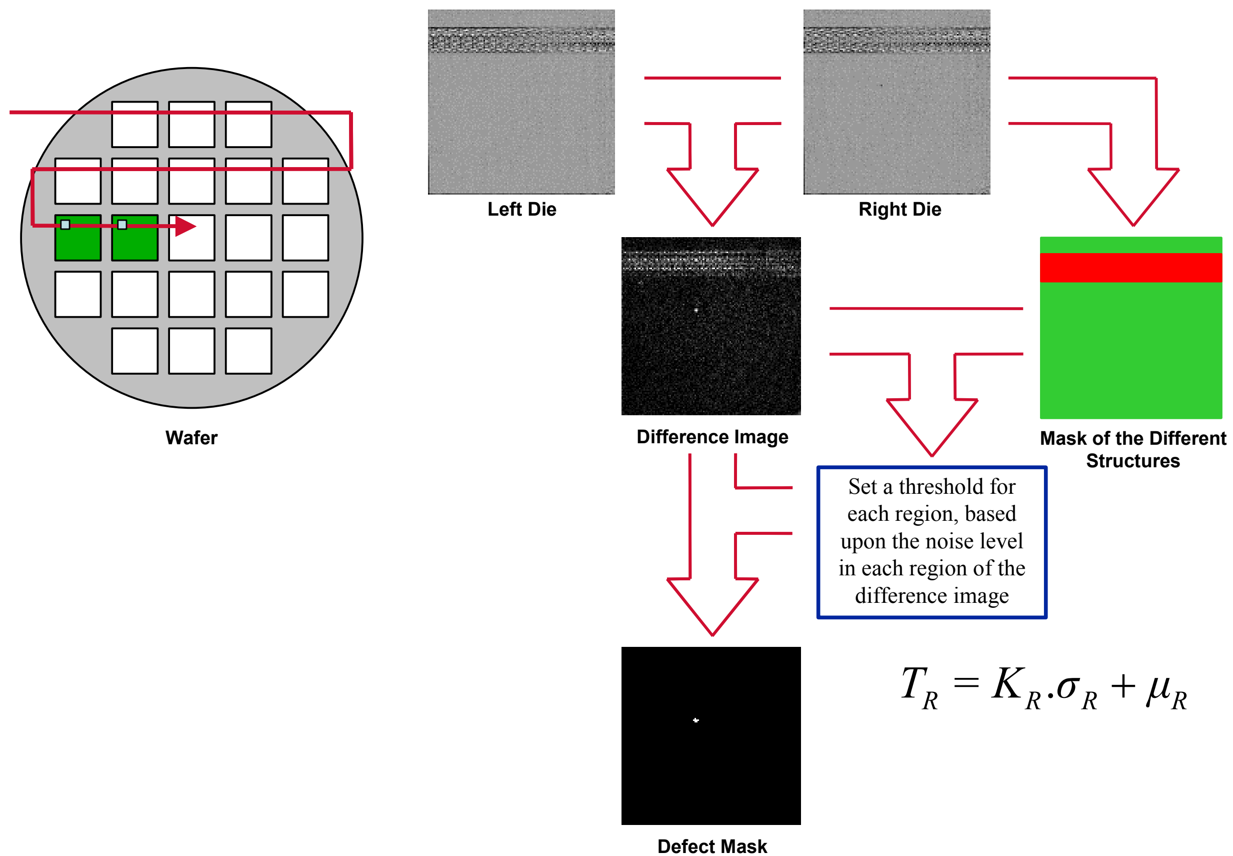

Fig. 2 : Principles of defect detection with the creation of a mask for local noise estimation and threshold estimation in a local area. μ and σ are respectively the average grey level and grey level standard deviation in the mask areas, whereas T is the threshold applied to the difference image to reveal the defects.

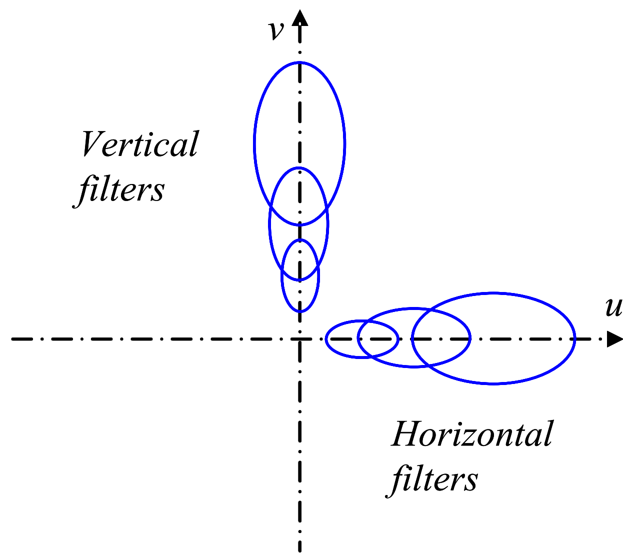

Fig. 3 : Bank of Gabor filters with 3 scales and 2 orientations

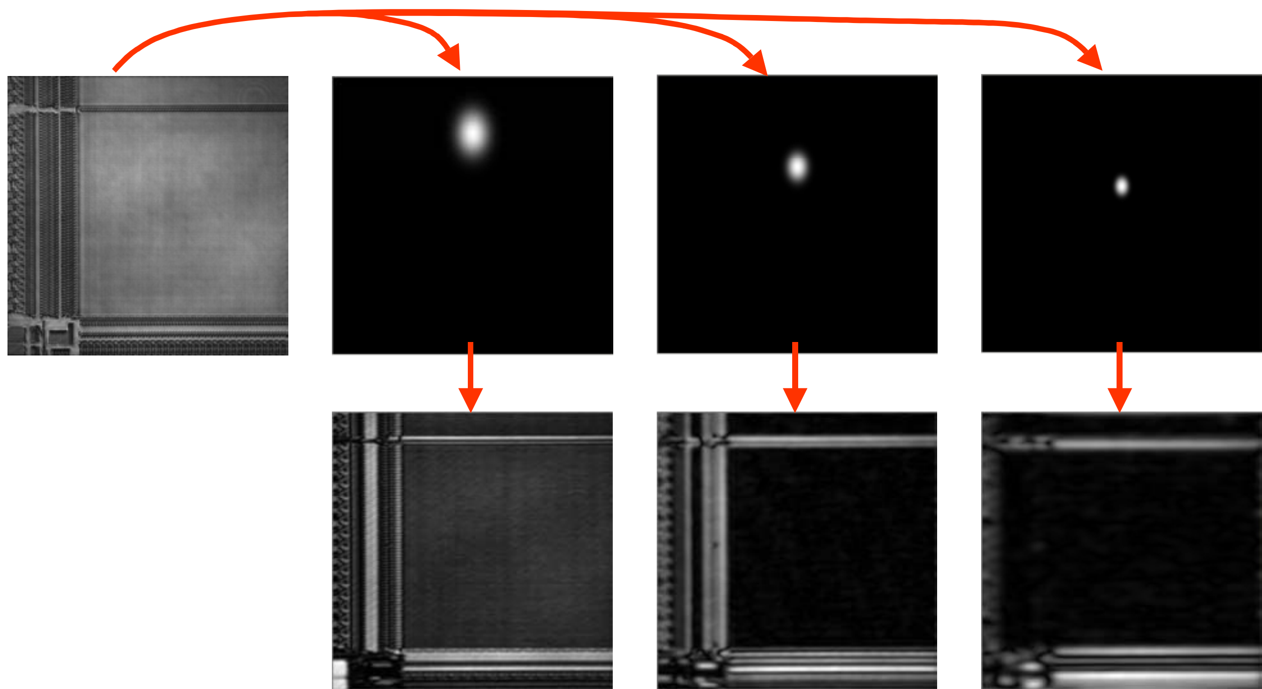

Fig. 4 : Original image and Horizontal details extraction on a three level decomposition. Each pixel is then represented by a 3-dimensional vector (1 value per image or multiresolution level), the same operation is carried out in the vertical direction leading finally to a 6-dimensional vector in the feature space for each pixel of the original image.

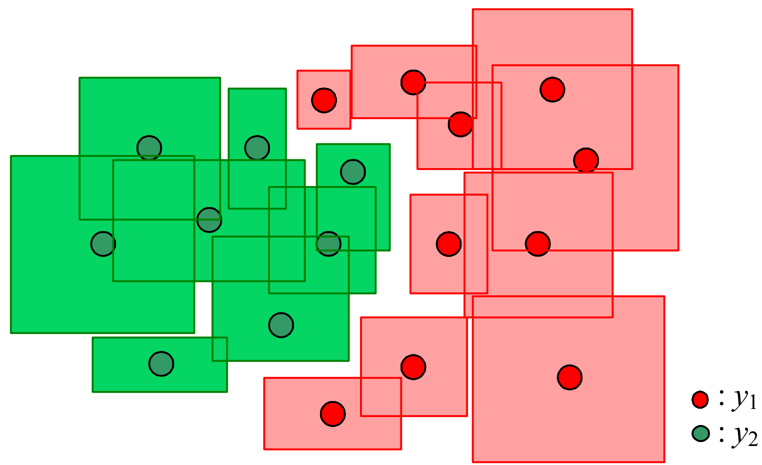

Fig. 5: Creation of a hypercube around each training point

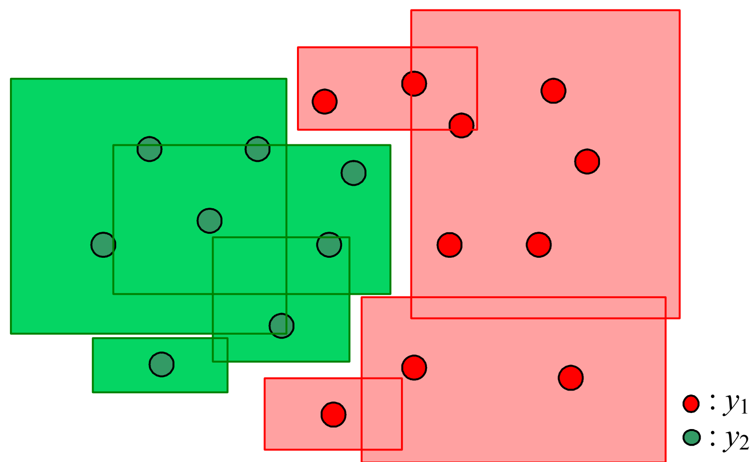

Fig. 6: Fusion of the hypercubes to reduce their number, and speed-up the decision stage

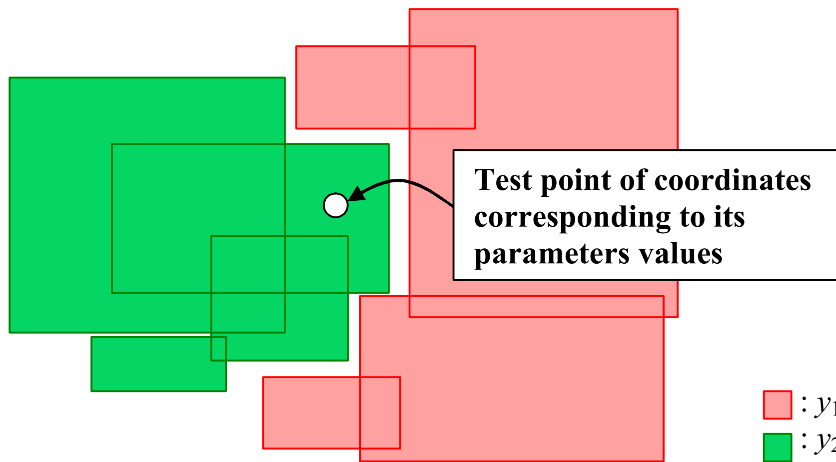

Fig.7 : Classification of a pixel according to its position in the hypercube space.

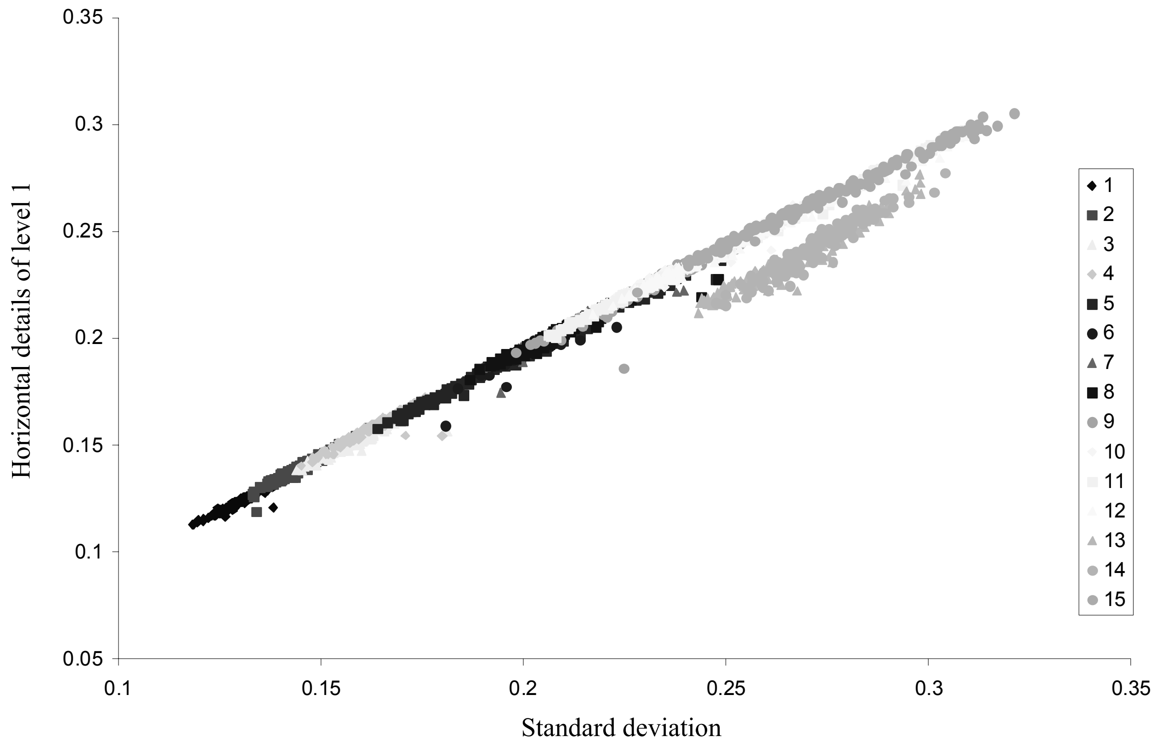

Fig. 8 : Representation of the horizontal details of level 1 against the standard deviation for 15 samples of the same texture captured at different locations of the wafer.

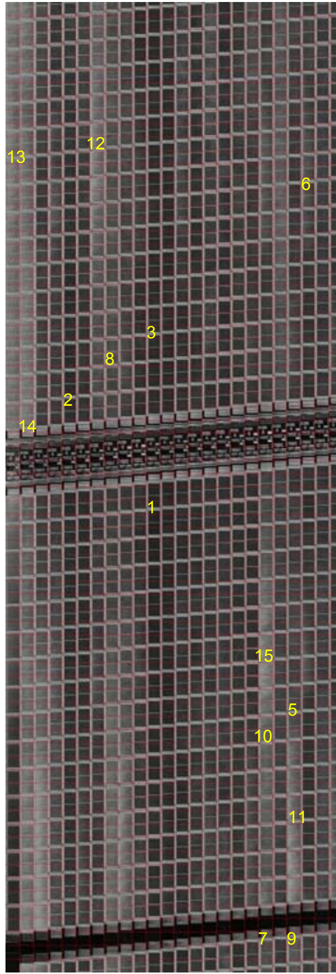

Fig. 9: Example of the 15 training samples and the test images.

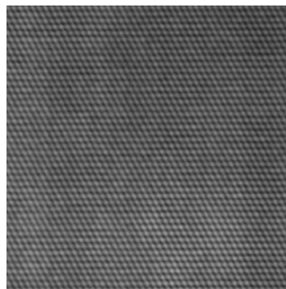

Fig. 10 : DRAM Area.

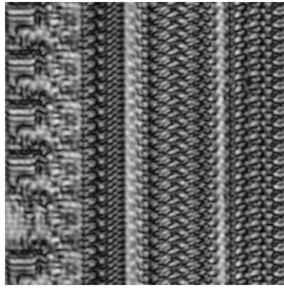

Fig. 11 : Logic Area

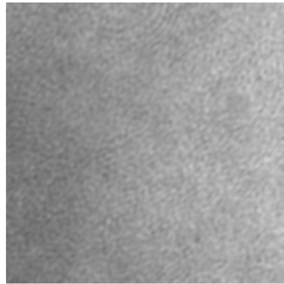

Fig. 12 : Field Area.

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

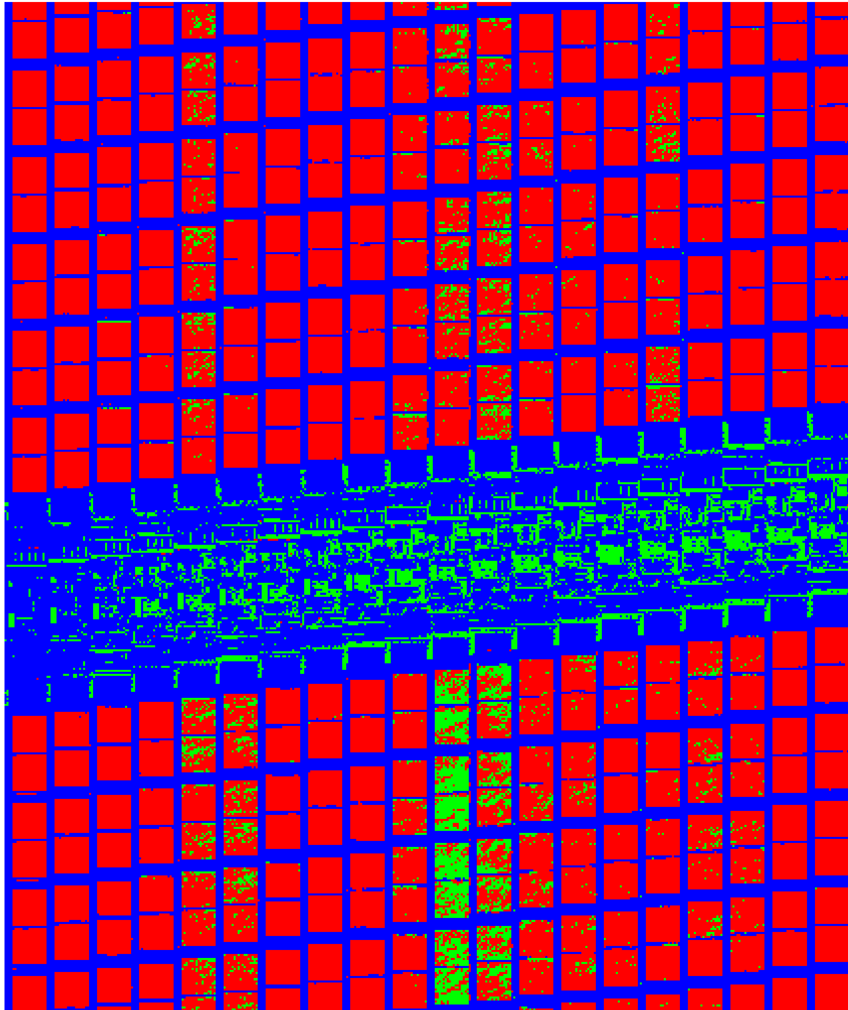

Fig 14 : Segmentation results based on Gabor filter bank applied to Normalized Amplitude Images with a single training sample (n°15 on figure 9).

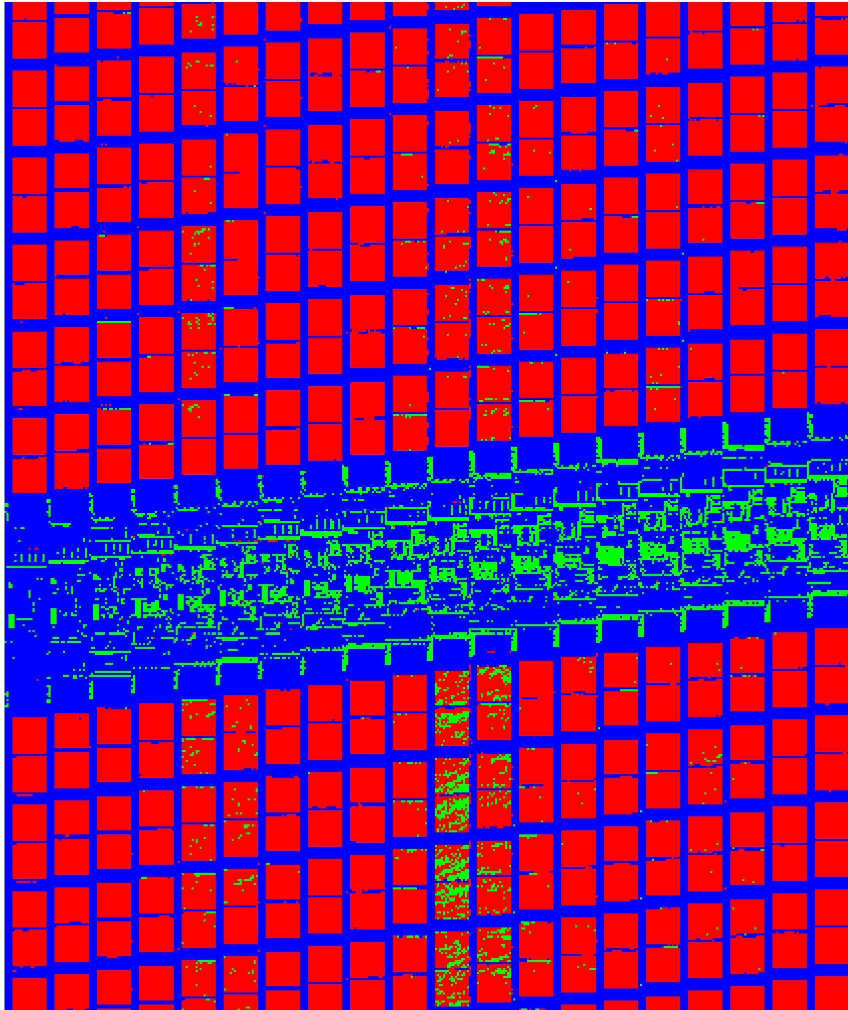

Fig 15 : Same as figure 15 with classifier corrections.

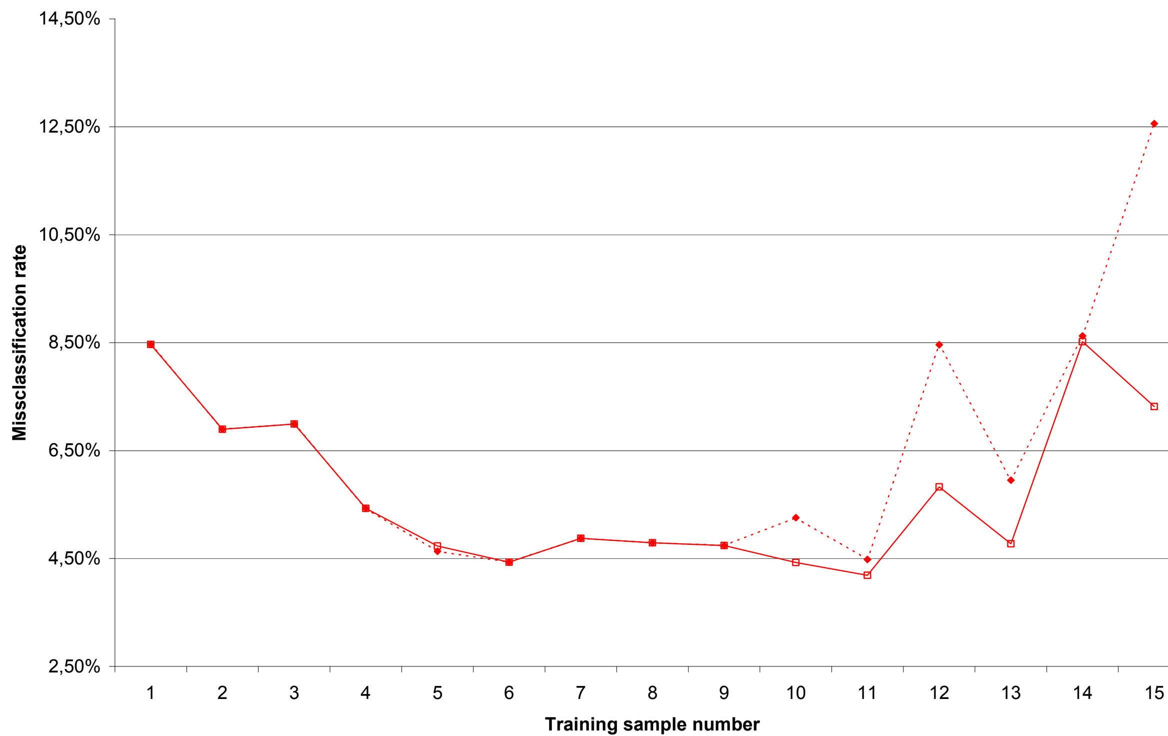

Fig. 16 : Misclassification rates obtained on Complex Normalized Amplitude images for the 15 different training bases without correction (dash line) and with correction (continuous line)

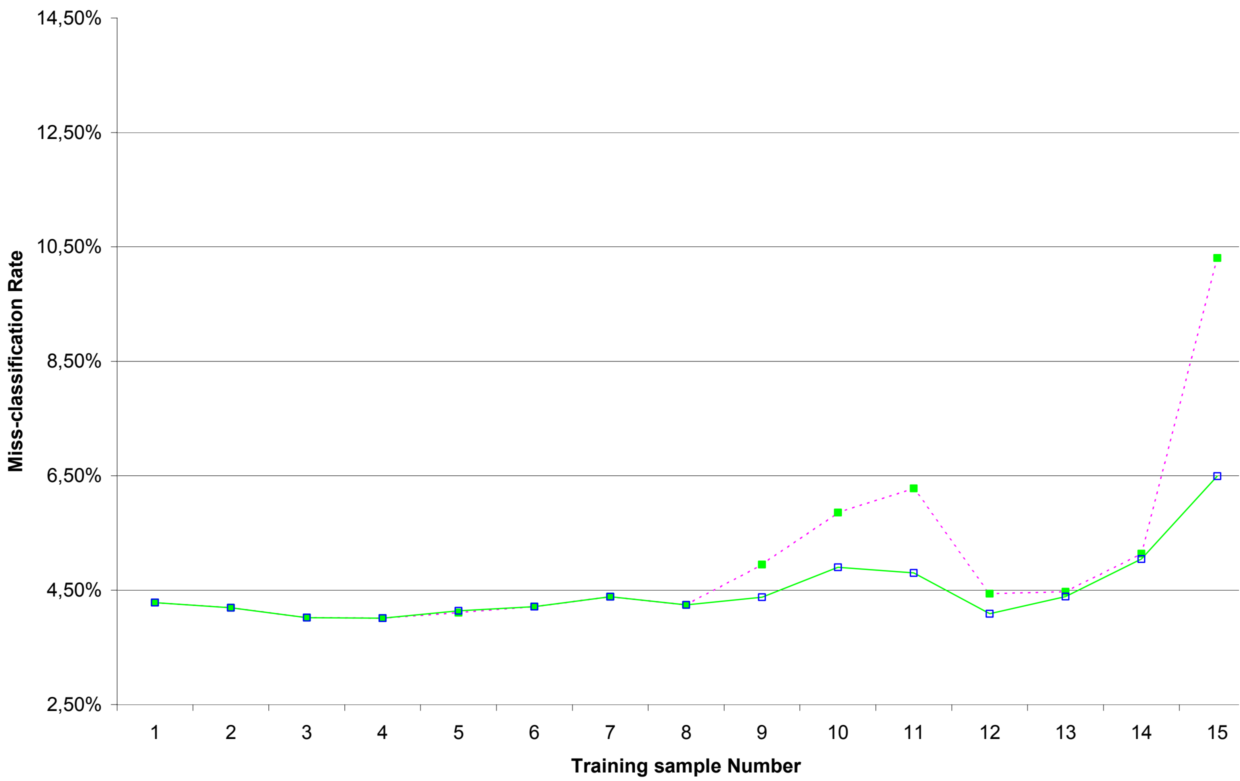

Fig. 17 : Misclassification rates obtained on Normalized Amplitude images for the 15 different training bases without correction (dash line) and with correction (continuous line)