



ELSEVIER

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Defect detection in patterned wafers using multichannel Scanning Electron Microscope

Maria Zontak, Israel Cohen *

The Department of Electrical Engineering, Technion – Israel Institute of Technology, Technion City, Haifa 32000, Israel

ARTICLE INFO

Article history:

Received 16 April 2008

Received in revised form

9 February 2009

Accepted 10 February 2009

Available online 27 February 2009

Keywords:

Semiconductor defect detection

Anomaly detection

Anisotropic kernels

Image reconstruction

Similarity measure

ABSTRACT

Recent computational methods of wafer defect detection often inspect Scanning Electron Microscope (SEM) images of the wafer. In this paper, we propose a kernel-based approach to multichannel defect detection, which relies on simultaneous acquisition of three different images for each sample in a SEM tool. The reconstruction of a source patch from reference patches in the three channels is constrained by a similarity criterion across the three SEM images. The improved performance of the proposed algorithm is demonstrated, compared to a single-channel kernel-based defect detection method.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Defect detection in wafers is a critical component of wafer manufacturing process. Various image processing techniques have been introduced for automatic defect detection in wafers [1–5]. Here we consider the problem of defect detection in patterned wafers using Scanning Electron Microscope (SEM) images. A wafer is irradiated with a focused beam of electrons directed to scan its surface. The analysis is carried out by moving the focused beam of electrons in a sweeping (raster) scan over the surface of the wafer. The energy exchange between the electron beam and the sample generates the emission of electrons and electromagnetic radiation which can be detected to produce an image. A SEM tool that is manufactured by Applied Materials can simultaneously produce three different images for a given sample, namely $External_1$, $External_2$ and $Internal$ images. The external images are acquired by detecting low energy secondary

electrons using external detectors placed by the two sides of the electrons beam, and the internal image is acquired by detecting high-energy backscattered electrons with a detector placed above the sample. The external images indicate the topography of the sample by light and shadows as if a “light source” is directed to a sample from top-left ($External_1$) or top-right ($External_2$). The internal image provides information about edges and material of the sample. Fig. 1 shows a SEM tool and examples of $External_1$, $External_2$ and $Internal$ images of a patterned wafer. Arrows in the images point to faults in the pattern associated with imperfect connections.

A semiconductor wafer typically contains many copies of the same electrical component (denoted as “dies”) laid out in a matrix pattern. A reference set of SEM images for one die is obtained by acquiring images of a random neighboring die, which is verified to be clear of defects. A common approach for defect detection utilizes the reference images for comparison with the inspected (source) images [6–10]. This method does not require a defect learning process, and identifies the defects according to the differences between the source images and their reference images. However, the reference images need to be aligned with the source images, and even if the

* Corresponding author. Tel.: +972 4 8294731; fax: +972 4 8295757.

E-mail addresses: zontakm@tx.technion.ac.il (M. Zontak), icohen@ee.technion.ac.il (I. Cohen).

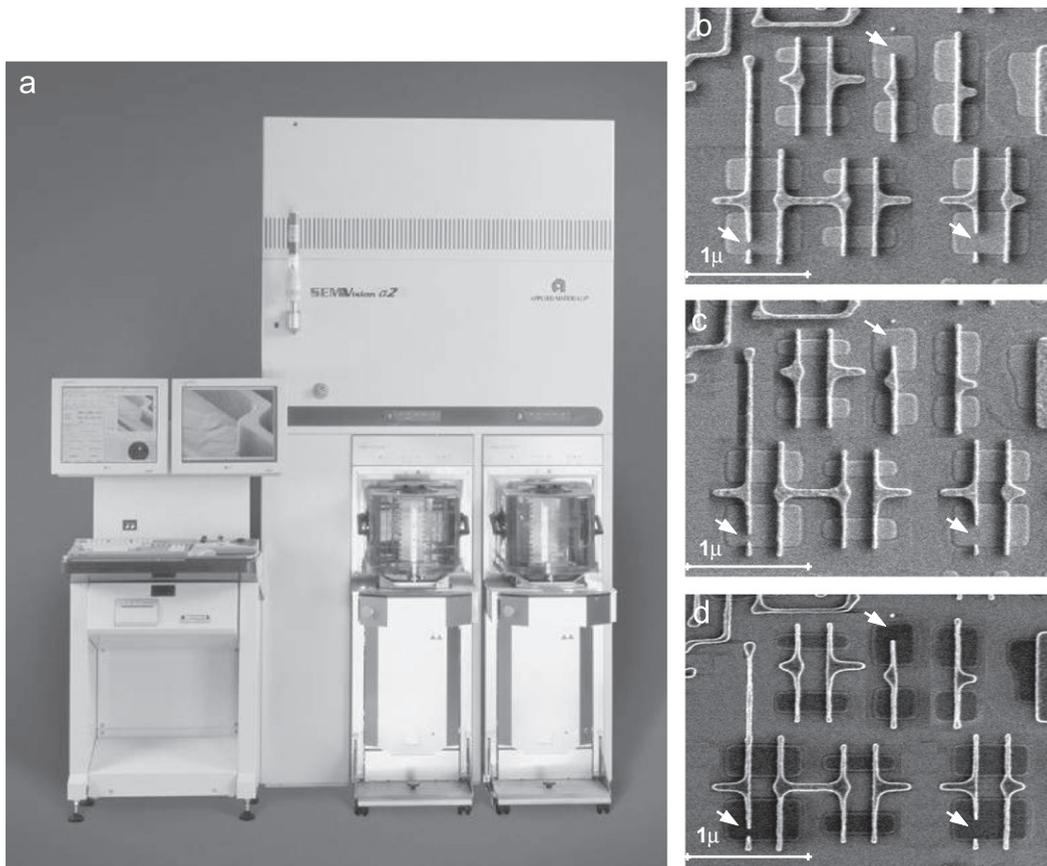


Fig. 1. (a) Scanning Electron Microscope; (b) $External_1$ image of a wafer, acquired by a SEM tool from top-right direction; (c) $External_2$ image of the same wafer, acquired from top-left direction; and (d) $Internal$ image of the same wafer, acquired from top direction. Arrows in the images point to defects.

registration technique (e.g., [11–13]) is perfect, pattern variations between source and reference images may yield large differences that obscure the differences associated with defects.

Recently, we introduced a defect detection procedure, which avoids image registration and is robust to pattern variations [14]. The method is based on anisotropic kernel reconstruction of the source image using its reference image. The source and reference images are mapped into a feature space, where every feature from the source image is estimated by a weighted sum of neighboring features from the reference image. We used patches around pixels as features and showed that patches originating from defect regions are not reconstructible from the reference image, and hence can be identified.

In this paper, we extend the kernel-based approach to multichannel defect detection, which relies on the simultaneous acquisition of three different images for each sample in the SEM tool. The proposed method assumes that if a pattern-originated region in the source wafer is similar to certain regions in the reference wafer, then this similarity is maintained across the three SEM images. Accordingly, the reconstruction of a source patch from reference patches in the three channels is constrained by a consistency criterion that the locations of

reference patches, which are most similar to the source patch, are identical in the three channels. We show that the proposed defect detection under constrained multi-channel reconstruction is more advantageous than the single-channel defect detection method.

This paper is organized as follows. In Section 2, we briefly review the single-channel kernel-based defect detection method and discuss the motivation for a multichannel defect detection approach. In Section 3, we introduce an algorithm for multichannel defect detection and demonstrate its performance. In Section 4, we address some open issues and future research, and conclude in Section 5.

2. Background and problem formulation

In this section, we review the single-channel kernel-based defect detection algorithm [14], and discuss some of its drawbacks in the case of non-periodically patterned wafers.

Pattern to pattern comparison is the most suitable technique for a SEM-based inspection system. This comparison could be performed by using a reference image that is obtained from another random die of the

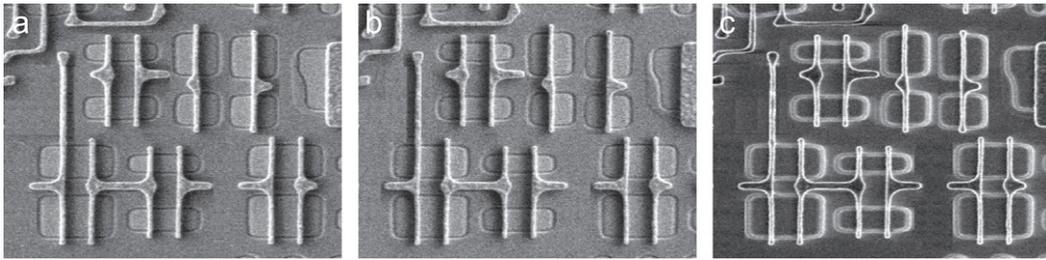


Fig. 2. Non-registered reference images of the wafer images presented in Figs. 1(b)–(d): (a) External₁ image (b) external₂ image and (c) Internal image.

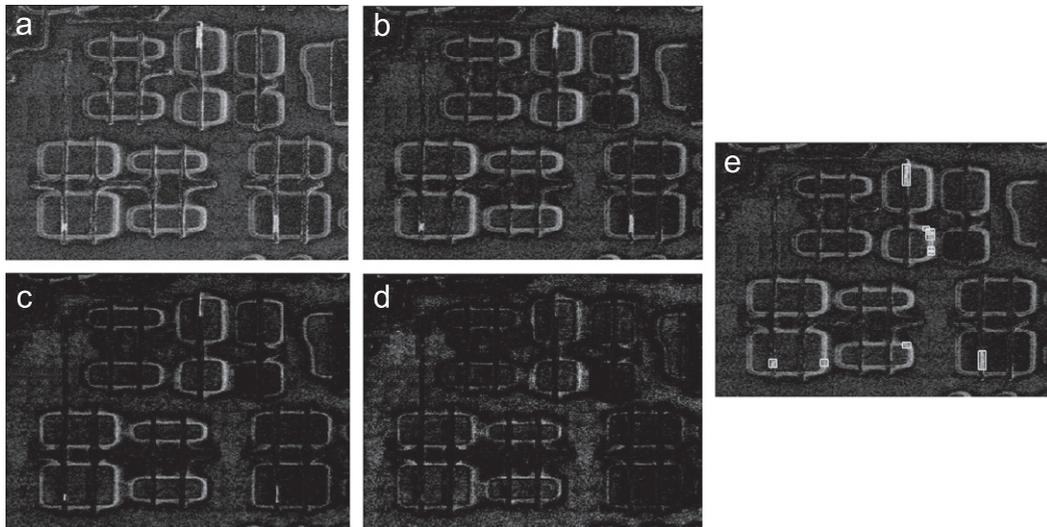


Fig. 3. Difference images of the internal source and reference images vs. different sizes of local registration neighborhood: (a) $[0, 0] \times [0, 0]$; (b) $[-1, 1] \times [-1, 1]$; (c) $[-3, 3] \times [-3, 3]$; (d) $[-5, 5] \times [-5, 5]$; and (e) $[-1, 1] \times [-1, 1]$ with overlaid true and false detections.

same wafer, which is verified to be clear of defects. Fig. 2 shows the examples of reference images to the images shown in Figs. 1(b)–(d). In many existing methods [6–10], the reference image is aligned with the source image and the defect detection procedure relies on their difference. However, the pattern in the reference image is generally not identical to the pattern in the source image and even if the alignment is perfect, pattern variation differences are still significant. These differences may be as intense as differences caused by defects and may cause false detections.

Figs. 3(a)–(d) demonstrate the difference images, created by global alignment of the inspection and reference images of the *Internal* channel (Figs. 1(d) and 2(c), respectively) and local alignment of every pixel in various-sized neighborhoods. The improved difference image is calculated for every pixel according to the minimal difference between the inspected pixel and the reference pixels within the chosen neighborhood. Fig. 3 demonstrates that the local registration cannot deal with the pattern variations problem. Differences caused by pattern variations may be reduced by increasing the size of the local registration neighborhood, but then the differences associated with defects are also suppressed.

Onishi et al. [5] presented a more robust algorithm, which tries to overcome the problem of slight distortion or rotation misalignment between the source and the reference patterns by using gray-scale morphological dilation of the reference and inspected images. The difference image is calculated according to the minimal distance between the reference and the inspected images in the dilation range. However, this technique can only manipulate slight mis-registration and pattern variations, and does not exploit the neighborhood replication of a periodic pattern. Fig. 4 shows the differences between the source images from Fig. 1 and their reference images after an alignment by the above algorithm. Clearly, defect detection by thresholding the difference images is characterized by high false and missed detection rates due to pattern variations.

Recently, we introduced a different defect detection method [14], which is more robust to pattern variations. Pixels in the source and reference images are mapped into a feature space. This mapping is flexible and there are many interesting choices for the possible features, as discussed in [14,15]. In this application, every pixel is represented by an intensity gray level vector, constructed from a square neighborhood (a patch) of fixed size $[s_x \times s_y]$ around the pixel. Let \mathbf{s} denote a pixel from the source

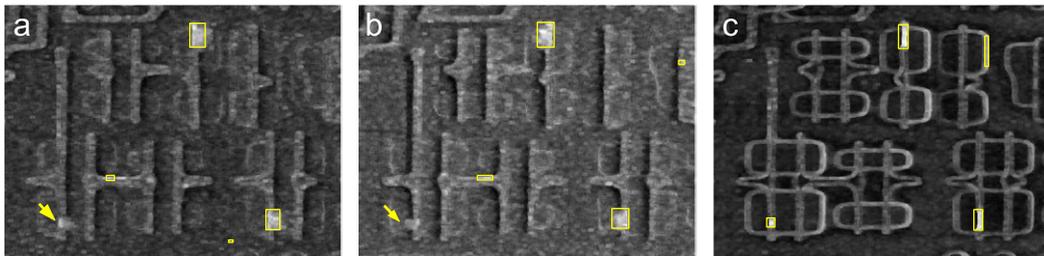


Fig. 4. Difference images with overlaid defect detection results (denoted by light rectangular). Pattern variations between the acquired wafer images (shown in Figs. 1(b)–(d)) and their reference images (shown in Fig. 2) generate non-negligible differences and high false detection rate: (a) External₁ image (b) external₂ image and (c) internal image.

image with coordinates (i, j) and let x denote a respective feature vector. Every source feature x is reconstructed using reference features, $\{x_i\}_{i=1}^m$, which represent all the m pixels from a search region in the reference image. The search region \mathcal{N}_s of the pixel s is given by $\mathcal{N}_s = \{s' | s' \in n_\kappa(s)\}$, where $n_\kappa(s)$ is the set of κ spatial nearest neighbors of s in the image domain. The reconstruction of the source feature x is obtained according to the similarity measure $k(x_i, x)$:

$$\hat{x} = \sum_{i=1}^m \frac{k(x_i, x)}{\sum_{j=1}^m k(x_j, x)} x_i, \quad (1)$$

and the total similarity of the source patch x to the pattern is defined as

$$\frac{1}{m} \sum_{i=1}^m k(x_i, x), \quad (2)$$

where $k(x, y) = \mathcal{G}_\varepsilon(x, y) \equiv e^{-1/2(\|x-y\|_2^2/\varepsilon)}$ is the Gaussian kernel [16]. If the total similarity is close to zero, which indicates that the source patch cannot be well reconstructed from the pattern, represented by patches of the reference image, then the presence of a defect is declared.

The above detection procedure, when applied to periodically patterned wafer images, does not require image registration and is robust to pattern variations. Registration of a reference image relative to the source image is not required, as long as reference patches are taken from a wide search region that covers at least one pattern period. Furthermore, a source patch does not have to be identical to one reference patch, but could be a combination of several patches to overcome the problem of pattern variations. The presented procedure not only compares the gray level in a single point but incorporates the information of the neighborhood, using patches as features. For periodic patterns, the search region is often more than one period of the pattern, in order to increase the number of potentially similar reference patches. It should be noticed that the proposed algorithm is invariant to rotation of the inspected wafer, because the source and reference images are acquired from neighboring dies on the same wafer and their patterns will be rotated similarly. However, if a reference pattern is rotated relatively to the source pattern, the performance may degrade, depending on the rotation angle and pattern characteristics.

Unfortunately for non-periodic patterns, the single-channel kernel-based detection method is insufficient, lacks robustness to pattern variations and is characterized by high false detection rate. Fig. 5 shows the reconstruction of the three images from Figs. 1(b)–(d) using their reference images from Fig. 2, where non-reconstructible regions are marked and identified as defects. In all the three channels, there are some false detections, which are associated with differences between the source and the reference images, rather than with defects. Our objective is to eliminate such false detections by exploiting relations between the three channels.

3. Multichannel detection

In this section, we discuss the statistical interpretation of the single-channel kernel-based method, which facilitates its extension to multichannel. Then we introduce the multichannel kernel-based algorithm for defect detection.

3.1. Constraint of consistent similarity between channels

Let $p(x)$ denote a probability density function of a random variable X , and let $\{x_i\}_{i=1}^m$ represent the samples of X . A non-parametric estimate of $p(x)$ can be obtained by Parzen method [17] and is given by

$$\hat{p}_\varepsilon(x) = \frac{1}{m} \sum_{i=1}^m b_\varepsilon(\|x - x_i\|), \quad (3)$$

where b_ε is a normal density with zero mean and variance ε . Ruiz and Lopez-de-Teruel [18] noted that a Parzen estimator can be related to the kernel similarity measure presented in (2) by

$$\hat{p}_\varepsilon(x) = x_x^T \cdot \vec{1}_m, \quad (4)$$

where $x_x = (k(x_1, x), k(x_2, x), \dots, k(x_m, x))^T$ is referred to as *empirical kernel map* [19]. We denote x_x as a *within-similarity map* between the sample of X and x . $\vec{1}_m$ denotes an m -dimensional column vector with all components equal to m^{-1} . Features originated from defects will have low similarity to the pattern, represented by reference features, and hence can be identified. We aim to perform multichannel detection using joint similarity between source features x , y and z and their reference features

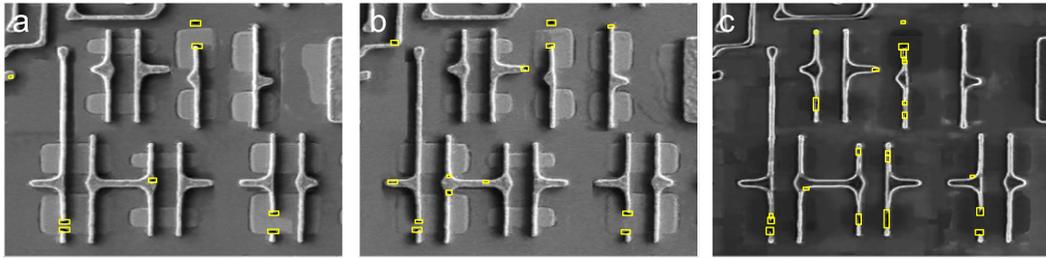


Fig. 5. Reconstructed wafer images with overlaid defect detection results obtained by using single-channel kernel-based algorithm (suspicious regions are denoted by light rectangulars). High false detection rate results from sensitivity to pattern variations: (a) External₁ image (b) external₂ image and (c) internal image.

Channel	Pattern-Originated Source Patch		Defect-Originated Source Patch	
	Source Image	Reference Image	Source Image	Reference Image
External ₁				
External ₂				
Internal				

Fig. 6. Example of similarity consistency between channels. The figure shows four columns of *External*₁, *External*₂ and *Internal* images. In the left column, a source patch that is free of defects is delineated by a rectangular. The second column shows the corresponding reference images, and the patches that are most similar to the source patch. In the third column, a source patch that contains a defect is delineated by a rectangular. The fourth column shows the corresponding reference images, and the most similar patches. For defect-free source patches, the locations of the most similar reference patches are the same in all channels (cf. second column). For a defect source patch, the locations of the most similar reference patches may be different in each channel (cf. fourth column).

$\{x_i\}_{i=1}^n$, $\{y_i\}_{i=1}^n$ and $\{z_i\}_{i=1}^n$ (from *External*₁, *External*₂ and *Internal* images, respectively).

We assume that if a pattern-originated region in the source wafer is similar to certain regions in the reference wafer, then this similarity is maintained across the three SEM images. Accordingly, the similarity between a source patch and its reference patches in the three channels is constrained by a consistency criterion that the locations of reference patches, which are most similar to the source patch, are identical in the three channels. The consistent similarity concept for three channels is demonstrated in Fig. 6. The figure shows four columns of *External*₁, *External*₂ and *Internal* images. In the left column, a source patch that is free of defects is delineated by a rectangular. The second column shows the corresponding reference images, and the most similar patches (for each channel, the patches in the reference image that are most similar to

the source patch are identified by rectangulars). In the third column, a source patch that contains a defect is delineated by a rectangular. The fourth column shows the corresponding reference images, and the most similar patches. It turns out that the locations of the most similar reference patches are the same for all channels in case the source patch is free of defects (cf. second column). However, in case the source patch contains a defect, the locations of the most similar reference patches may be different for each channel (cf. fourth column). Consistent similarity is an additional characteristic that may be used to enhance multichannel distinction between defective and pattern-originated regions.

For clarity of presentation, first we consider two-channel consistent similarity, and then extend the formulation to three channels. Based on a data sample $X = \{x_1, \dots, x_m\}$, the general form of a kernel density

estimator is given as $\hat{p}_{e,\gamma}(x) = \sum_{i=1}^m \gamma_i b_e(\|x - x_i\|)$, where γ_i are non-uniform weighting coefficients. According to the relation from (4) between Parzen's estimator and the similarity measure, the weighting coefficients γ_i could be regarded as *a priori* estimation of a similarity between a reference patch x_i and a source patch x . Without any *a priori* assumptions about these similarity relations, a zero-order estimation of γ_i is $\gamma_i^{(0)} = 1/m, \forall i$.

Under the assumption of consistent similarity between the channels, we consider the following first-order refinement of γ_i :

$$\gamma_i^{(1)} = \frac{k(y_i, y)}{\sum_{i=1}^m k(y_i, y)}, \quad (5)$$

where x and y relate to the same pixel \mathbf{s} in two different channels and $\{y_i\}_{i=1}^m$ are the reference patches in the second channel. Hence, we can write the conditional estimated probability of a patch x to arise from the pattern statistics, given a *within-similarity map* of a respective patch y from the second channel, as follows:

$$\hat{p}(x|y) = \sum_{i=1}^m \gamma_i^{(1)} k(x_i, x). \quad (6)$$

Note that if $k(y_i, y) = K, \forall i$, then $\gamma_i^{(1)}$ reduces to $\gamma_i^{(0)}$ and (6) reduces to (4).

Hence, we define the joint similarity measure of two channels as

$$\begin{aligned} J_{xy} &= \hat{p}(x, y) = \hat{p}(x|y)\hat{p}(y) \\ &= \left(\sum_{i=1}^m k(x_i, x) \frac{k(y_i, y)}{\sum_{i=1}^m k(y_i, y)} \right) \left(\frac{1}{m} \sum_{i=1}^m k(y_i, y) \right) = \frac{1}{m} x_X^T \cdot y_Y. \end{aligned} \quad (7)$$

The single-channel similarity measure, given by (2), answers the question whether a source image has similar patches in its reference search region. The joint similarity measure answers whether the similar patches locations in the first channel are consistent with the similar patches locations in the second channel. This measure from (7) is symmetric, $\hat{p}(y, x) = (\sum_{i=1}^m k(y_i, y)k(x_i, x) / \sum_{i=1}^m k(x_i, x)) / ((1/m) \sum_{i=1}^m k(x_i, x)) = (1/m)y_Y^T \cdot x_X = (1/m)x_X^T \cdot y_Y = \hat{p}(x, y)$, and bounded between zero and one. However, it disobeys the triangle inequality (i.e. it is non-metric distance), which often happens in the case of distance functions that are robust to outliers and noise [20]. It can be shown that its negative natural logarithm is a probabilistic distance [21]. For the convenience of presentation we adopt its positive natural logarithm as a likelihood measure.

Finally, the extension of (7) for three channels is

$$\begin{aligned} J_{xyz} &= \hat{p}(x, y, z) = \hat{p}(x, y|z)\hat{p}(z) \\ &= \left(\sum_{i=1}^m k(x_i, x)k(y_i, y) \frac{k(z_i, z)}{\sum_{i=1}^m k(z_i, z)} \right) \frac{1}{m} \sum_{i=1}^m k(z_i, z) \\ &= \frac{1}{m} \sum_{i=1}^m k(x_i, x)k(y_i, y)k(z_i, z) \end{aligned} \quad (8)$$

and the corresponding joint likelihood to the pattern is measured by

$$L_{xyz} = \log(J_{xyz}). \quad (9)$$

3.2. Implementation of the algorithm

Algorithm 1.

Multichannel defect detection

- 1: **for all** $\mathbf{s} \in \Omega$ **do** $\{\mathbf{s}$ —pixel index in the source images, Ω —image support
- 2: **for all** $\text{ch} \in \text{CHANNELS}$ **do** $\{\text{CHANNELS} = [\text{External}_1, \text{External}_2 \text{ and } \text{Internal}]\}$
- 3: $p_{\text{ch}}^{\mathbf{s}} \leftarrow$ a raster scan of the neighborhood window $[s_x \times s_y]$ around pixel \mathbf{s} in the respective channel ch
 $\{p_{\text{ch}}^{\mathbf{s}}$ represents x, y and z feature vectors in the *External*₁, *External*₂ and *Internal* channels}
- 4: **for all** $\mathbf{s}' \in \mathcal{N}_{\mathbf{s}}$ **do** $\{\mathbf{s}'$ —pixel index in the reference image, $\mathcal{N}_{\mathbf{s}}$ —search region neighborhood of \mathbf{s}
- 5: $p_{\text{ch}}^{\mathbf{s}'} \leftarrow$ a raster scan of the neighborhood window $[s_x \times s_y]$ around pixel \mathbf{s}' in the respective channel ch
 $\{p_{\text{ch}}^{\mathbf{s}'}$ represents x_i, y_i and z_i feature vectors in the *External*₁, *External*₂ and *Internal* channels}
- 6: $\mathcal{W}_{\text{ch}}^{\mathbf{s}, \mathbf{s}'} \leftarrow \exp(-\rho(p_{\text{ch}}^{\mathbf{s}}, p_{\text{ch}}^{\mathbf{s}'})^2 / 2\sigma_{\text{ch}}^2)$ (ρ —Euclidean metric; $\mathcal{W}_{\text{ch}}^{\mathbf{s}, \mathbf{s}'}$ refers to $k(x, x_i), k(y, y_i)$ and $k(z, z_i)$)
- 7: $J_{\mathbf{s}} = \sum_{\text{vs}} \prod_{\text{vch}} \mathcal{W}_{\text{ch}}^{\mathbf{s}, \mathbf{s}'}$
- 8: **if** $\log(J_{\mathbf{s}}) < \tau$ **then**
- 9: $\mathbf{s} \in \mathcal{A}$ $\{\mathcal{A}$ is a set of defect

Algorithm 1 summarizes the reconstruction and decision procedures for multichannel defect detection. To verify whether a pixel from a source image belongs to a defect area, we execute the following steps. Patches around each pixel in a source image are column-stacked into vectors, which represent features for the reconstruction process (step 3). The same construction of features is performed for each pixel in the reference search regions of all the channels (step 5). The respective *within-similarity maps* are calculated in every channel for the determined search region (step 6), and the joint similarity measure is calculated using (8) (step 7). Finally, the detection is performed by thresholding the log likelihood values (steps 8 and 9). In our experiments the threshold was set to detect the outliers of the log-likelihood values. It should be noticed that the parameter that controls the trade-off between miss-detection and false detection rates is not the threshold, but the choice of similarity parameter, as will be discussed in Section 4.

3.3. Experimental results

Fig. 8 shows the detection result for images from Figs. 1(b)–(d) using joint likelihood measure. Thresholding low likelihood values reveals the defect edges without false detections, as presented on the *Internal* image. This joint likelihood could be compared to the single channel likelihoods from Figs. 7(a)–(c), which result in high false detection rates (see Fig. 5). Additionally, the joint likelihood of all the three channels could be compared to the likelihood measures based on the combination of only two channels, which are presented in Figs. 7(d)–(f). Although these similarity measures succeed to provide the same detection like the joint similarity measure, their separation tolerance is lower (for Figs. 7(d) and (f)) or similar (for Fig. 7(e)). The separation tolerance is calculated according to the ratio between the range of threshold values that allow the exact detection without false alarms and the total range of the log-likelihood values.

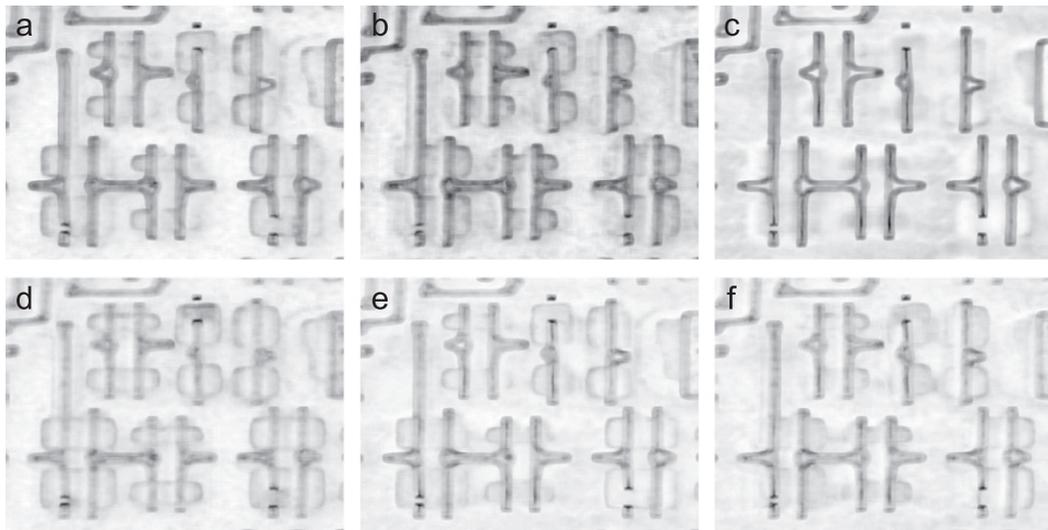


Fig. 7. (a)–(c) Single channel similarity measures, related to the reconstruction results from Fig. 5; (d)–(f) similarity measures based on the combination of two channels: (a) External₁ image (b) external₂ image and (c) internal image (d) external₁–external₂ (e) external₁–internal and (f) external₂–internal.

Table 1

Detection and false detection results obtained by using the multichannel and single-channel algorithms.

Case	Detection results									Separation tolerance			
	Single channel detection						Multichannel detection			Single channel detection			Multichannel detection
	External ₁		External ₂		Internal		D	FD	D	FD	D	FD	
	D	FD	D	FD	D	FD							External ₁
1	1 (partial)	0	1 (partial)	0	0	1	1	0	0.3	0.3	0	0.5	
2	0	2	0	1	1	0	1	0	0	0	0.4	0.3	
3	0	2	1	0	0	2	1	0	0	0.35	0	0.3	
4	1	0	1	4	0	1	1	0	0.1	0	0	0.2	
5	0	1	1	0	0	1	1	0	0	0.1	0	0.1	
6	2 of 2	>5	1 (partial) of 2	1	1 (partial) of 2	5	2 of 2	0	0	0	0	0.1	
7	8 of 9	0	9 of 9	>5	2 of 9	0	9 of 9	0	0	0	0	0.1	

Note that in this example only the edges of the defects are detected, because the patch window is relatively small (11×11 pixels in our example) compared to the whole defect (which reaches up to 60 pixels in one of the two dimensions in the presented example). The choice of the patch sized is discussed in Section 4

The proposed kernel-based algorithm was successfully applied to images with pattern variations, wherever the algorithm that was based on the difference image failed. The multichannel algorithm was also compared to a single-channel algorithm that was based on the calculation of log-likelihood in every channel separately, without similarity consistency constraint. The joint multichannel detection outperformed separate detections, in the case of non-periodic patterns and defects that were not evident in all the channels. Table 1 shows several examples of the comparison, which involve different patterns. The table presents detection results, which include the number of exact detections (D), the number of false detections (FD), and the separation tolerance. In cases 1–5 the source

image contains only one defect, while in cases 6–7 the source image contains several defects (two and nine, respectively) of different sizes and shapes. In all the tested cases, the threshold was set automatically for optimal detection.¹ The search region and the patch sizes were identical for all cases. Similarity parameters were chosen according to what is described in Section 4. In cases where the detection was partial, i.e. only part of a defect was detected, this was stated. The results show that the performance of the single channel algorithm is not

¹ Table 1 demonstrates that given the optimal thresholds for both multichannel and single-channel algorithms, the multichannel algorithm succeeds to detect all the defects, while the single-channel algorithm can detect all the defects only with additional false alarms. Therefore, the optimal threshold was set under an assumption that there are defects in the image. In our experiments we used the following procedure: the likelihood image is normalized in the range of [0,1] and pixels with likelihood values above 0.95 are considered to be defects (this value was empirically chosen to achieve the best PD-FAR ratio for all the algorithms).

sufficient. The application of the single channel algorithm to the *External*₁, *External*₂ or *Internal* channel yields either low detection rate or high false detection rate. Furthermore, combing the single channel detection results by “AND” or “OR” operations over the three channels may either increase the false detection rate or decrease the detection rate. Compared to the single channel algorithm, the multichannel algorithm enables to improve the detection rate while decreasing the false detection rate.

4. Discussion

4.1. Choice of search region and features sizes

The performance of the proposed algorithm depends on several parameters: size of a search region, patch size and similarity parameter ε . The minimal search region should allow compensation for mis-registration; however, in order to achieve robustness to pattern variations, it should cover several periods of the pattern in the case of a periodic pattern. Generally, for periodic and non-periodic patterns, enlarging the search region increases the number of potentially similar reference features to a pattern originated source feature. This allows to enforce higher similarity, by choosing a smaller ε , which increases the detection rate of defects. Hence, to achieve higher detection and lower false detection rates, it is advantageous to use a search region as wide as possible. However, this is computationally disadvantageous and the choice depends on the trade-off between the computational load and the detection performance. For a detailed discussion the reader can refer to [14].

In general, the predicted size of defects determines the choice of a patch size. According to the ROC analysis presented in [14], the optimum is achieved when a patch captures not only the defect but also the surroundings as well. Although surroundings incorporation in the patch is important to achieve a contrast between normal and abnormal areas, it is important to preserve dominance of the defect over the surroundings. Hence, the patch size should not be too small or too large, according to the defect sizes. Relatively small sized patches are suitable for capturing small defects, but also enable detection of edges of large defects. Large patches, while suitable for large defects, yield poor detection results for small defects. Furthermore, the computational complexity increases as the patch size increases. Hence, smaller patches are generally preferable.

4.2. Adjustment of the similarity parameter

The similarity parameter ε determines the nearest neighbors that take part in feature reconstruction and pixel estimation. It controls the relation between the distances in feature space and the corresponding weighting factors. It is important to choose a sufficiently large ε to enable reconstruction of the source features from the reference features, according to (1), even in case of pattern variations (to prevent false detections). However, ε should be sufficiently small (high similarity constraint) to prevent

reconstruction of features related to defects and thereby facilitate the distinction between pattern variations and defects. In previous experiments [14], we adjusted this parameter in a single channel, so that any reference feature could be reconstructed from a large representative set of reference features (excluding itself). We chose the minimal ε that provided good reconstruction results.

Since ε represents the local scale of similarity, there is no one scale value that is optimal for every point. For example, we would like to have higher similarity requirement (smaller ε) for smooth areas and lower similarity requirement (larger ε) for edges. In order to perform local scaling, it is possible to apply an approach that is given in [22], where given a number of neighbors, m , the distances at each point are scaled so that the m th nearest neighbor has a distance of 1; that is, we let $\rho_x(a, b) = \rho(a, b) / \rho(x, x_m)$ ($\rho_x(a, b)$ is the Euclidean metric), where x_m is the m th nearest neighbor to x . Since ρ_x varies over the data set, to make the weight matrix symmetric, they use the geometric mean of ρ_x and ρ_y in the argument of the exponential, i.e.

$$k(x_i, x_j) = e^{-\rho_{x_i}(x_i, x_j) \rho_{x_j}(x_i, x_j) / \varepsilon}. \quad (10)$$

This is called the self-tuning similarity weight. There is still a similarity parameter in the weight, but a global ε in the self-tuning weights corresponds to some location dependent choice of ε in the standard exponential weights.

In our multichannel experiments, the similarity parameter is adjusted relatively in different channels. Pattern variations between the source and the reference images reduce the similarity of the source feature to its reference features. Too small ε applies high similarity requirement, which may cause false detections due to pattern variations between the source and the reference images.

Often pattern variations are more disturbing in the *Internal* channel, what can be, for example, observed by comparing Figs. 1(b)–(d) and 2. It is important to eliminate the influence of the channel with high pattern variations on the detection procedure, by adjusting the corresponding ε to be relatively large. In the presented examples we adjusted the similarity parameter according to the global similarity of the source and reference images in every channel, which is determined according to the Bhattacharyya distance between the gray level histograms of the images.

The Bhattacharyya measure can be used to compare the similarity between two histograms as follows. Let R_i be the frequency coded quantity in bin i for the reference image histogram and S_i a similar quantity for the source image histogram. The Bhattacharyya distance $-\log(\sum_i \sqrt{R_i S_i})$ provides a measure of similarity between the two histograms and hence between the source and reference images. A successful utilization of the Bhattacharyya measure for histogram matching and similarity testing can be found in several applications [23,24]. Relying on our experiments, we propose to adjust the similarity parameter between the channels according to

$$\varepsilon_k = -\alpha \log \left(\sum_i \sqrt{R_i^k S_i^k} \right). \quad (11)$$

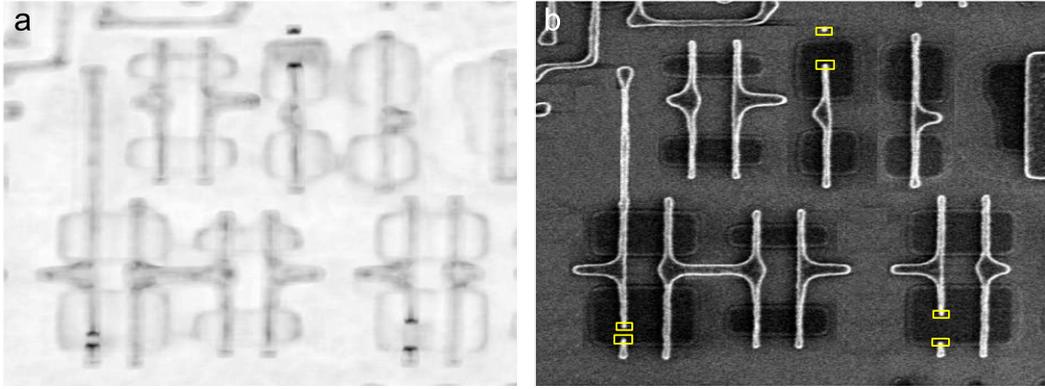


Fig. 8. Detection results using a joint similarity measure: (a) joint similarity L_{xyz} (the gray level represents the likelihood that a patch is similar to the pattern) and (b) *Internal* image with overlaid detection results (denoted by light rectangular).

The above approach allows the relative adjustment of ε between the channels. The parameter α can be determined using one of the procedures described above.

4.3. Reduction of the feature space dimension

We notice that using a Euclidian distance, the joint similarity measure given in (8) could be viewed as a single similarity measure with a combined feature:

$$\begin{aligned} J_{xyz} &= \frac{1}{m} \sum_{i=1}^m k_{\varepsilon_x}(x_i, x) \cdot k_{\varepsilon_y}(y_i, y) \cdot k_{\varepsilon_z}(z_i, z) \\ &= \frac{1}{m} \sum_{i=1}^m \text{Break} \end{aligned}$$

where the combined feature is

$$\mathbf{s} \rightarrow \mathbf{v} = \begin{bmatrix} \sqrt{\varepsilon_y \varepsilon_z} x \\ \sqrt{\varepsilon_x \varepsilon_z} y \\ \sqrt{\varepsilon_x \varepsilon_y} z \end{bmatrix} \quad (12)$$

and the combined similarity parameter is $\varepsilon = \varepsilon_x \varepsilon_y \varepsilon_z$. The dimension of the joint feature space is tripled compared to that of the single feature space, which is disadvantageous from the computational point of view. This dimension could be reduced by observing that both *External* channels incorporate depth information. Hence, creating one depth map image from two *External* images will reduce the dimension, while preserving the existing information.

4.4. Computational complexity

The computational complexity of the proposed algorithm is $O(n \cdot m \cdot d)$, where n denotes the number of pixels in the image, m denotes the number of reference features and d is a feature dimension. Although the implementation of the algorithm, as presented in Algorithm 1 on typical home computer results in high computational load, a reduction in complexity can be achieved by some modifications. For example, a multi-scale implementation, similar to that proposed for image denoising applications [25], may be advantageous in our framework. The main idea is first to perform a search in the coarsest scale and to

continue the search in finer scales only in regions that were found similar in coarser scales. The proposed algorithm can also be combined with the standard state-of-the-art wafer defect detection algorithms, to reduce the false alarm rate without increasing the missed detection rate. Suspicious regions are first detected by a conventional defect detection algorithms. Subsequently, the reconstruction procedure is applied only to patches around the suspicious pixels using the proposed algorithm, and regions that are not reconstructible are identified as defects.

Additionally, the implementation of the proposed algorithm can be accelerated by parallel calculation of the log-likelihood from (9). Consider a problem in three channels, where a search region contains m reference features of size d . The number of operations for each pixel is $m((9d + 2) + (4 + 1))$.² The example presented in Fig. 8 was processed using a search region of 49×49 pixels and patch size 11×11 pixels, which requires 0.0026 Giga-FLOPs (FLOP = operation) per pixel. The standard Graphical Processing Unit model GeForce 8800 GT [26] has 336 GigaFLOPs/s theoretical performance.³ Hence, the theoretical limit for computational rate is 128 KPPS (kilo pixels per second), not considering memory bandwidth that is relatively negligible in the given algorithm. The computation rate of a preliminary implementation⁴ of the joint detection procedure, using the parameters stated above ($d = 11 \times 11$ and $m = 49 \times 49$) and GeForce 8800 GT Graphical Processing Unit, has reached 31 KPPS. Hence, the run time of the algorithm for images presented in Figs. 1(b)–(d) (each image has 530×460 pixels) was 8 s. This result can be improved by implementation optimization and by using state-of-the-art hardware.

² The term $9d$ is due to the calculation of the joint exponent weight that involves d subtractions, d squares, $d - 1$ sums and one multiplication, and is performed three times (three channels), which are summarized twice; besides there are m exponents (each requires four operations), which are summarized $m - 1 \approx m$ times.

³ This is according to the fact that there are 112 stream processors (SP) at 1.5 GHz with each SP being able to run at least two operations (FLOPs) per clock.

⁴ The authors thank Mr. Yuri Pekelny for providing a CUDA implementation of the algorithm.

5. Summary

We have proposed an algorithm for automatic defect detection in wafers using three channels of SEM images. The kernel-based detection algorithm exploits the periodic nature of the wafer pattern and compensates for pattern variations and miss-registration. If the inspected pattern is not periodic, the proposed method exploits the multichannel information to compensate for pattern variations. We have introduced a kernel-based similarity measure that quantifies similarity relations between the inspected patch and its reference patches under the assumption that the locations of similar patches in the search regions are invariant for all the channels. We have demonstrated improved performances of the constrained multichannel detection compared to the single channel detection in the case of non-periodic pattern. The detection procedure based on a consistent similarity constrain is advantageous over simple integration of the detection results in different channels, because it allows compensation for pattern variations. The proposed approach is also appealing for defect detection in periodic patterned wafers, even when defects are observable in distinct channels. Future research directions may include developing a depth map from the *External* channel images, which will allow the reduction of the feature space dimension without the loss of the existing information. Additionally, local spatial adjustment of the similarity parameter ε will enable improving the detection of weakly noticed defects in smooth regions and robustness to pattern variations nearby edges.

Acknowledgments

The authors thank Prof. Ronald R. Coifman for helpful discussions. They also thank the anonymous reviewers for their useful comments that helped to improve the quality of this work.

References

- [1] C.Y. Chang, S.Y. Lin, M. Jeng, Using a two-layer competitive Hopfield neural network for semiconductors wafer defect detection, in: Proceedings of IEEE International Conference on Automation Science and Engineering, vol. 5, Edmonton, Canada, August 2005, pp. 301–306.
- [2] S. Gleason, R. Ferrell, T. Karnowski, K. Tobin, Detection of semiconductor defects using a novel fractal encoding algorithm, in: Proceedings of SPIE, Process Integration, and Diagnostics in IC Manufacturing, vol. 4692, March 2002, pp. 61–71.
- [3] P. Xie, S. Guan, A golden-template self-generating method for patterned wafer inspection, *Machine Vision and Applications* 12 (3) (October 2000) 149–156.
- [4] S.-U. Guan, P. Xie, H. Li, A golden-block-based self-refining scheme for repetitive patterned wafer inspections, *Machine Vision and Applications* 13 (5–6) (March 2003) 314–321.
- [5] H. Onishi, Y. Sasa, K. Nagai, S. Tatsumi, A pattern defect inspection method by parallel grayscale image comparison without precise image alignment, in: Proceedings of 28th IEEE Annual Conference of the Industrial Electronics Society, vol. 3, Santa Clara, CA, November 2002, pp. 2208–2213.
- [6] N. Shankar, Z. Zhong, Defect detection on semiconductor wafer surfaces, *Microelectronic Engineering* 77 (April 2005) 337–346.
- [7] D.M. Tsai, C.H. Yang, A quantile–quantile plot based pattern matching for defect detection, *Pattern Recognition Letters* 26 (13) (October 2005) 1948–1962.
- [8] D.-M. Tsai, C.-H. Yang, An eigenvalue-based similarity measure and its application in defect detection, *Image and Vision Computing* 23 (12) (November 2005) 1094–1101.
- [9] B. Dom, V. Brecher, Recent advances in the automatic inspection of integrated circuits for pattern defects, *Machine Vision and Applications* 8 (1) (January 1995) 5–19.
- [10] T. Hiroi, S. Maeda, H. Kubota, K. Watanabe, Y. Nakagawa, Precise visual inspection for LSI wafer patterns using subpixel image alignment, in: Proceedings of 2nd IEEE Workshop on Applications of Computer Vision, Sarasota, Florida, USA, December 1994, pp. 26–34.
- [11] X. Dai, M. Hunt, M. Schulze, Automated image registration in the semiconductor industry: a case study in the direct to digital holography inspection system, in: Proceedings of SPIE, Machine Vision Applications in Industrial Inspection XI, vol. 5011, Santa Clara, CA, January 2003.
- [12] T. Hiroi, C. Shishido, M. Watanabe, Pattern alignment method based on consistency among local registration candidates for LSI wafer pattern inspection, in: Proceedings of 6th IEEE Workshop on Applications of Computer Vision, Orlando, Florida, USA, 2002, pp. 257–263.
- [13] C. Costa, M. Petrou, Automatic registration of ceramic tiles for the purpose of fault detection, *Machine Vision and Applications* 11 (5) (February 2000) 225–230.
- [14] M. Zontak, I. Cohen, Defect detection in patterned wafers using anisotropic kernels, *Machine Vision and Applications*, to be published (Online). Available from: (<http://springerlink.com/content/c8p372452221277>).
- [15] A. Szlam, Non-stationary analysis on datasets and applications, Ph.D. Dissertation, Yale University, New Haven, Connecticut, USA, May 2006.
- [16] V.N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, Berlin, 1995.
- [17] E. Parzen, On estimation of a probability density function and mode, *The Annals of Mathematical Statistics* 33 (3) (September 1962) 1065–1076.
- [18] A. Ruiz, P.E. Lopez-de-Teruel, Nonlinear kernel-based statistical pattern analysis, *IEEE Transactions on Neural Networks* 12 (1) (January 2001) 16–32.
- [19] B. Schölkopf, S. Mika, C. Burges, P. Knirsch, K.-R. Müller, G. Ratsch, A. Smola, Input space vs. feature space in kernel-based methods, *IEEE Transactions on Neural Networks* 10 (September 1999) 1000–1017.
- [20] D. Jacobs, D. Weinshall, Y. Gdalyahu, Class representation and image retrieval with non-metric distances, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (6) (June 2000) 583–600.
- [21] P. Devijver, J. Kittler, *Pattern Recognition: A Statistical Approach*, Prentice-Hall International, London, 1982.
- [22] L. Zelnik-Manor, P. Perona, Self-tuning spectral clustering, in: 18th Annual Conference on Neural Information Processing Systems (NIPS), 2004.
- [23] F. Aherne, N. Thacker, P.I. Rockett, Optimal pairwise geometric histograms, in: Proceedings of 8th British Machine Vision Conference, Essex, UK, 1997, pp. 480–490.
- [24] S. Liapis, G. Tziritas, Color and texture image retrieval using chromaticity histograms and wavelet frames, *IEEE Transactions on Multimedia* 6 (5) (October 2004) 676–686.
- [25] A. Buades, B. Coll, J.M. Morel, A review of image denoising algorithms, with a new one, *Multiscale Modeling and Simulation* 4 (2) (2005) 490–530.
- [26] NVIDIA (Online). Available from: (http://www.nvidia.com/object/force_8800gt.html).