EURASIP Journal on Applied Signal Processing 2002:7, 704–713


# Content-Based Image Retrieval for Semiconductor Process Characterization

**Kenneth W. Tobin**
*Oak Ridge National Laboratory, P.O. Box 2008, Building 3500, MS-6010, Oak Ridge, TN 37831-6010, USA*
*Email: tobinkwjr@ornl.gov*

**Thomas P. Karnowski**
*Oak Ridge National Laboratory, P.O. Box 2008, Building 3500, MS-6010, Oak Ridge, TN 37831-6010, USA*
*Email: karnowskitp@ornl.gov*

**Lloyd F. Arrowood**
*Oak Ridge National Laboratory, P.O. Box 2008, Building 3500, MS-6010, Oak Ridge, TN 37831-6010, USA*
*Email: arrowoodlf@y12.doe.gov*

**Regina K. Ferrell**
*Oak Ridge National Laboratory, P.O. Box 2008, Building 3500, MS-6010, Oak Ridge, TN 37831-6010, USA*
*Email: ferrellrk@ornl.gov*

**James S. Goddard**
*Oak Ridge National Laboratory, P.O. Box 2008, Building 3500, MS-6010, Oak Ridge, TN 37831-6010, USA*
*Email: goddardjsjr@ornl.gov*

**Fred Lakhani**
*International SEMATECH, 2706 Montopolis Dr., Austin, TX 78741-6499, USA*
*Email: fred.lakhani@sematech.org*

*Received 14 August 2001 and in revised form 12 February 2002*

Image data management in the semiconductor manufacturing environment is becoming more problematic as the size of silicon wafers continues to increase, while the dimension of critical features continues to shrink. Fabricators rely on a growing host of image-generating inspection tools to monitor complex device manufacturing processes. These inspection tools include optical and laser scattering microscopy, confocal microscopy, scanning electron microscopy, and atomic force microscopy. The number of images that are being generated are on the order of 20,000 to 30,000 each week in some fabrication facilities today. Manufacturers currently maintain on the order of 500,000 images in their data management systems for extended periods of time. Gleaning the historical value from these large image repositories for yield improvement is difficult to accomplish using the standard database methods currently associated with these data sets (e.g., performing queries based on time and date, lot numbers, wafer identification numbers, etc.). Researchers at the Oak Ridge National Laboratory have developed and tested a content-based image retrieval technology that is specific to manufacturing environments. In this paper, we describe the feature representation of semiconductor defect images along with methods of indexing and retrieval, and results from initial field-testing in the semiconductor manufacturing environment.

**Keywords and phrases:** content-based image retrieval, semiconductor manufacturing, image indexing, automatic defect classification, approximate nearest neighbors.

## 1. INTRODUCTION

The ability to manage large image databases has been a topic of growing research. Imagery is being generated and maintained for a large variety of applications including remote sensing, architectural and engineering design, geographic information systems, and weather forecasting. Content-based image retrieval (CBIR) is a technology that is being developed to address these application areas [1]. CBIR refers to

techniques used to index and retrieve images from databases based on their pictorial content [2, 3]. Pictorial content is typically defined by a set of features extracted from an image that describe the color [4, 5], texture [6, 7], and/or shape [8, 9, 10] of the entire image or of specific objects [11]. This feature description is used in CBIR to index a database through various means such as distance-based techniques, rule-based decision-making, and fuzzy inferencing [12, 13, 14]. The manufacturing environment represents an application area where CBIR technologies have not been extensively studied. In this paper, we will describe the application of CBIR technologies to the semiconductor manufacturing environment.

CBIR addresses a problem created by the growing proliferation of automated microscopy inspection in semiconductor manufacturing applications, that is, the management and reuse of the large amounts of image data collected during defect inspection and review. In semiconductor integrated circuit (IC) manufacturing, the rapid identification of yield detracting mechanisms is a primary goal of defect sourcing and yield learning. At future IC manufacturing technology nodes, yield learning must proceed at an accelerated rate to maintain current defect sourcing cycle times despite the growth in circuit complexity and the amount of data acquired on a given silicon wafer or lot [15]. For semiconductor yield management applications, we have denoted CBIR technology as Automated Image Retrieval (AIR) [16, 17]. Digital imagery for failure analysis is generated between process steps from optical microscopy and laser scattering systems and from optical confocal microscopy, scanning electron microscopy (SEM), atomic force microscopy (AFM), and focused ion beam (FIB) imaging modalities. This data is maintained in a data management system (DMS) and used by fabrication engineers to diagnose and isolate manufacturing problems. The semiconductor industry currently has no direct means of searching the DMS using image-based queries, even though 20 000 images are collected on average at a typical fabrication (fab) facility every week [18]. Current abilities to query the fabrication process are based primarily on product ID, lot number, wafer ID, time/date, process layer, engineer classification, or automatic defect classification (ADC) [19], and so forth. Although this approach can be useful, it limits the user's ability to quickly locate historical examples of visually similar imagery, especially for data that was placed in the database over one or two weeks prior. Data much older than this is nearly irretrievable since retrieval is dependent on human memory and experience. Without the addition of datamining capabilities such as AIR, this large image repository will remain virtually untapped as a resource for rapidly resolving manufacturing problems.

The Oak Ridge National Laboratory (ORNL) AIR system represents a unique application of CBIR technologies to the manufacturing environment. In Section 2, we provide an overview of the AIR software system and the fundamental premise of operation in the manufacturing environment. In Section 3, we describe the method of image analysis, feature combination, indexing, and image retrieval. In Section 4, we compare the function of the AIR system to that of an ADC tool, a common automation technology used in semiconductor wafer inspection today. In Section 5, we present results obtained from field-testing of our image retrieval system at two semiconductor fabrication sites during the fall of 2000, including statistics on system performance, quantitative results on the efficacy of AIR as a defect sourcing tool, and qualitative visual results of various queries for SEM and optical imagery.

## 2. OVERVIEW OF THE AIR SYSTEM

Image retrieval technologies have been under development since the early 1990s, but very few applications have evolved for solving specific, real-world problems such as those in the manufacturing environment. Researchers at ORNL developed the capability for a flexible image retrieval technology for industrial applications that independently takes into account details regarding the product defectivity, substrate (i.e., the background structure on which the defect resides), and imaging modality characteristics [16]. *The fundamental premise of the ORNL AIR method and technology is that a similar process or phenomenon likely generates images that are visually similar*. This implies that statistical process information that is associated with retrieved images can be used to identify and isolate errant process tools and equipment. Therefore, in our AIR system, process data associated with the inspected product is included with the defect imagery in a relational database for subsequent statistical analysis to provide the yield engineer with defect sourcing information.

The basic component of the AIR system is the indexing and retrieval engine, a dynamic link library (DLL), which generates the defect and substrate image feature descriptions, and the indexing structure used for efficient storage and retrieval of images from the database. In addition to the core AIR DLL, the system includes an ORACLE™ database, a set of interface DLLs and executables, and graphical user interfaces such as shown in Figure 1. For our current semiconductor application, the fab DMS system generates an ASCII data file on a daily basis that provides process and image data in a format suitable for inclusion into the AIR database. A Windows NT service executable periodically checks for output from the DMS system, and when detected, the service adds the imagery and associated process data to the ORACLE™ database, and builds the indexing structure necessary for efficient image retrieval.

## 3. IMAGE INDEXING AND RETRIEVAL METHOD

We describe the methodology associated with AIR processing. In overview, this begins with the generation and/or use of the defect detection mask, which localizes the defect in the image. Next, a series of image features are extracted from the defect and substrate regions of the image. The features become the entire representation of the image and are indexed for rapid retrieval from the database. Finally, the image data is associated with the manufacturing process within a relational database for subsequent statistical process analysis.

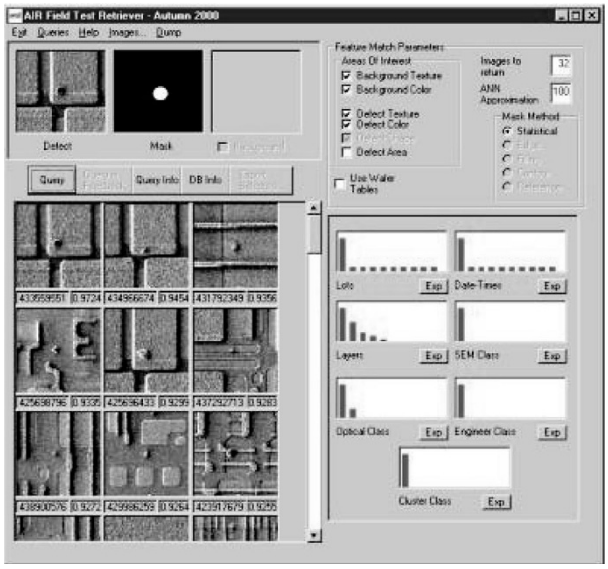
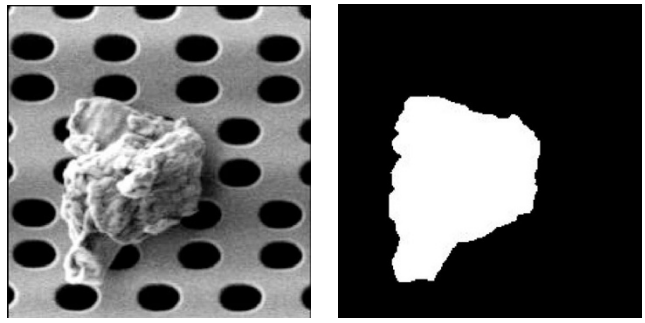



FIGURE 1: Example interface to the AIR system. Queries are entered in the upper left, results are displayed below in the lower left, and associated statistical process information is shown in the lower right. AIR control parameters are listed in the upper right.



(a) (b)

FIGURE 2: (a) SEM image of a particle defect. (b) Associated defect mask generated by the inspection tool during automatic redetection.

### 3.1. Defect masks and feature analysis

Figure 2 shows an SEM image containing a particle-type defect sitting on the top of a product substrate in (a) and the associated defect mask in (b). The defect mask is typically a binary representation that localizes the defect boundaries in the field of view and separates defect from substrate regions in the image. This mask can be generated in the form of a filled region, as shown in the figure, or as a perimeter composed of boundary pixels. Every inspection tool in the semiconductor industry today that performs automated defect detection or redetection generates a defect mask during the process. The defect mask is used to calculate descriptive features regarding the defect such as its size and location, or more extensive information useful for ADC, such as color, texture, and shape features.

The defect mask is used in AIR to generate an extensive description of the defect region and the substrate region. There are currently 60 numerical features measured for the substrate that describe the color and structure. The defect is decomposed into 51 numerical features that describe the color, texture, shape, and area. These feature types are detailed in Table 1. These features are organized into distinct sets of descriptors that are used to independently describe various attributes of the image according to a particular user's query specifications. For example, there are 24 features used to describe the color of the defect and 36 features used to describe the structure of the substrate. The feature descriptor lists are represented in the relational database (DB) as independent tables as will be described below. The user has the ability to select various sets of feature descriptors when formulating a query so that, for example, a search can be accomplished to locate one defect shape on another product substrate by ignoring color attributes, which are likely to be highly variable from one process layer or product to the next. The user could also enable or disable other descriptive groups such as texture or shape as required.

When calculating features of the substrate, it has been determined to be advantageous to "fill" the known defect region with an estimate of the substrate obtained from areas adjacent to the defect as shown in Figure 3. This is accomplished by measuring the predominant orientation of the substrate structure using a vertical and horizontal gradient operator. The postprocessed gradient image energy is then determined as $G_h = \sum \partial f(x, y)/\partial x$ for the horizontal energy and $G_v = \sum \partial f(x, y)/\partial y$ for the vertical energy, where the summation is over all pixels in the gradient images. To predict the direction of fill, a horizontal and vertical distribution of energy is defined as $E_h = G_h/(G_h + G_v)$ and $E_v = G_v/(G_h + G_v)$, respectively, with $E_h + E_h = 1$. For example, in Figure 3, the predominant orientation is horizontal, therefore the defect region, defined by the known mask, is filled from left to right, row by row, by extracting a vector of pixels immediately adjacent to the defect vector along each row in the image. This procedure is well suited to semiconductor device imagery since device structure is printed predominantly in a vertical or horizontal direction.

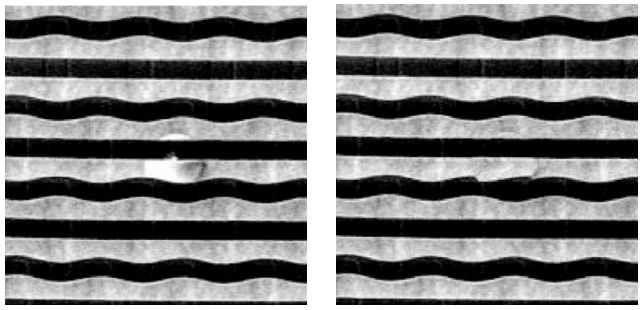
This independent description of defect and substrate facilitates a wide variety of queries such as *find this defect on a different substrate,* or *find this defect on any substrate*. This architecture and flexibility allows a single AIR system to be used by a broad population of users with widely varying needs while still providing focused and specific image retrieval searches.

### 3.2. Indexing and retrieval

The goal of indexing is to organize the image features in the database such that a ranked list of nearest neighbors can be retrieved without performing an exhaustive comparison with all the records in the database. For AIR this is achieved by generating a binary decision tree of the image features.

Table 1: Feature summary of defect and substrate regions in the imagery.

| Broad area | Specific area | Measurement | Number of values |
|---|---|---|---|
| Defect | Color | Histogram | 24 |
| Defect | Texture | Statistical-based values derived from intensity distributors | 6 |
| Defect | Shape | Fourier coefficients | 20 |
| Defect | Area | Number of pixels in defect | 1 |
| Substrate | Color | Histogram | 24 |
| Substrate | Structure | Statistical-based values derived from intensity distributions, measured on blocks of the background image | 36 |



(a) (b)

Figure 3: Example of an image with defect in (a) and the estimated "filled" version used to estimate substrate feature measurements in (b).

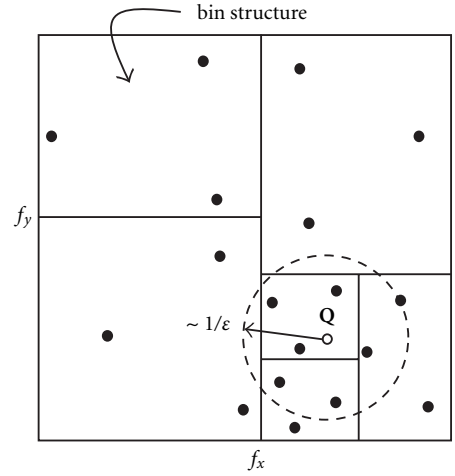



Figure 4: Example of kd-tree bin structure showing the ANN search region about a query point, **Q**.

A bin is defined as a bottom-level element in our tree structure, sometimes described as a "leaf" or terminal node, that contains a small list of images, for example, a bottom-level bin may contain a list of image vectors $\{\mathbf{v}_a, \mathbf{v}_b, \mathbf{v}_c, \ldots\}$. Under the AIR architecture, a query vector is compared at the top level to each of the two subnodes, and a decision is made as to which subtree to take. There are many ways to implement decision trees. For this work we have implemented an approximate nearest-neighbor (ANN) indexing and search method that builds on *kd-tree* methods [20]. Whereas an exhaustive nearest-neighbor search of the $n$ vectors (i.e., images) in the database would be of $O(n)$ computations, the *kd-tree* approach is of $O(\log(n))$.

Figure 4 shows a simple example of a two-dimensional feature space, $(f_x, f_y)$, containing 18 image vector points partitioned into a *kd-tree* structure where each bin contains 3 points (i.e., image vectors). The *kd-tree* method allows for the rapid retrieval of the closest bin to the query point, **Q**, but the data in this bin are not necessarily the closest points and the nearest-neighbor result can be in error by an amount $\varepsilon$.

The ANN method incorporates a search window that results in the collection of neighboring bins about the query point. As this window increases in radius, the nearest-neighbor error, $\varepsilon$, decreases, but the performance of the system also decreases to $O(n)$. The efficiency of the ANN method is proportional to $O((1/\varepsilon)^{N/2}\log(n))$, where $N$ is the dimension of the feature space, $n$ is the number of data points, and $\varepsilon$ is the nearest-neighbor error. The nearest-neighbor error is therefore inversely proportional to the size of the search window as shown in Figure 4. As the radius of the search window increases, neighboring bins containing additional image vectors are included in the final nearest-neighbor search. As the radius continues to grow, the system approaches the complexity of an exhaustive nearest-neighbor search. Therefore, the accuracy of the AIR system is selectable as a trade-off between nearest-neighbor performance and computational efficiency.

The architecture for AIR incorporates one table (represented by the kd-tree shown in Figure 4) for each set of descriptors that were defined above. Therefore, there are six descriptor sets that are used to independently characterize the query image and perform retrievals from the system. Figure 5 shows schematically how the results from these tables are combined. Once a query vector, **Q**, is submitted to the AIR engine, each table will return a prespecified number of

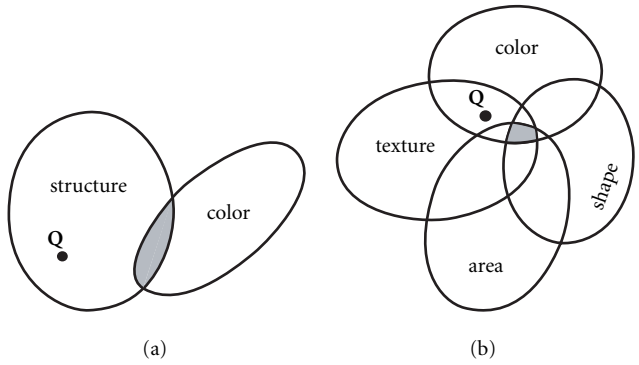



Figure 5: (a) Representation of feature sets describing the semiconductor defect substrate in (a) and defect in (b). The intersection of these descriptive sets defines a similar retrieval.

image matches that form sets. These sets are combined according to Boolean logic operations to determine the regions of most overlap. For example in Figure 5, the query point, **Q**, (shown in both the substrate set space and the defect set space) would have matches only for similar substrate structure, but similar defect texture and color. The list of feature vectors are then returned for similarity processing to determine a quantitative estimate of visual similarity. For AIR, similarity $s_i(\mathbf{Q}, \mathbf{v}_i)$ is determined for each (query, vector) pair in each set independently as a function of the $L$-norm distance, $s_i(\mathbf{Q}, \mathbf{v}_i) = 1 - d_i(\mathbf{Q}, \mathbf{v}_i)/\sqrt{N}$, where $N$ is the dimension of the feature space and $d_i(\mathbf{Q}, \mathbf{v}_i) = \|\mathbf{Q} - \mathbf{v}_i\|$. These similarity values from each set are then combined in a cross-set average and ranked for the display of results or subsequent analysis of statistical process information.

It should also be noted that the structure of the ANN method facilitates the inclusion of new image data into the data set without necessarily requiring a rebuild of the database, or more specifically the indexing structure. The database is updated on a periodic basis, for example, once daily. During use, the system will be gathering image data that will be incorporated into the indexing structure during these periodic maintenance cycles. While images are being collected, they can be placed within the bin structure and retrieved during subsequent queries. As the number of image vectors in these bins increases, the efficiency of the ANN process will begin to degrade. When the periodic build is actuated, the image vectors will be redistributed to result in the predefined minimum bin size required for optimal retrieval efficiency. The result is that this structure allows access to the latest image data by incorporating it into the database on-the-fly without immediate reindexing.

## 4. DIFFERENCES BETWEEN AIR AND ADC TECHNOLOGIES

AIR has been designed to allow the management of large repositories of defect image data through one system. Since its inception as a yield management tool, there have been many questions regarding the differences between AIR and the more common ADC systems that have been proliferating throughout the industry over the past decade. To respond to this, it is necessary to view the two systems through the concept of a simple feature space as shown in Figure 6. Each point shown in the graph in (a) and (b) corresponds to the feature description of an image. In the case of the classifier in (a) the goal is to classify the data vector, **C**, whereas for image retrieval in (b) the goal is to retrieve other data points that are similar to the query vector, **Q**.

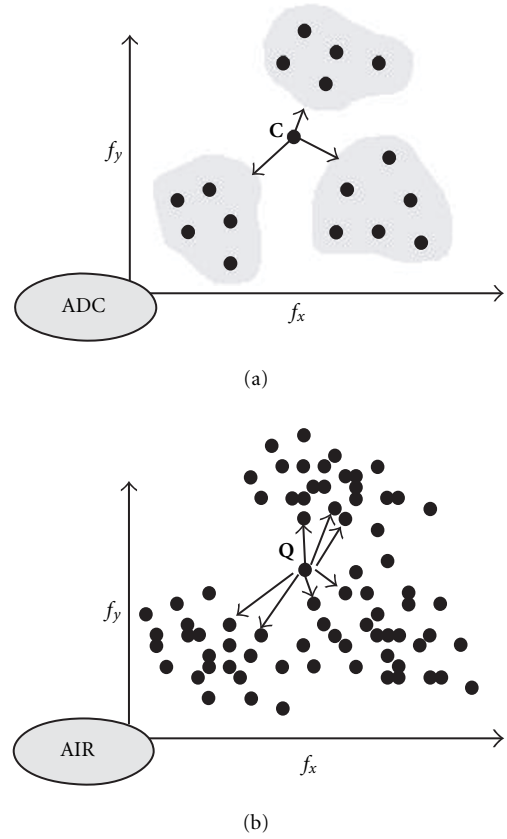



Figure 6: A feature-space comparison of (a) ADC versus (b) AIR. In (a) the query point **C** is a point to be classified by the ADC system whereas in (b) the data query **Q** is a query upon which to perform image retrieval.

In more detail, Figure 6a shows a representation of the ADC system whose function is to classify, or assign, an unknown data point, **C**, to a class that has been defined through a training procedure. The ADC system typically requires training with data that is specific to an inspection tool. Within that tool set, there is a requirement to train on specific layers or process steps, and for the various products that are being inspected. Training is a cumbersome and sometimes unwieldy process that has proved to be a limitation, especially in fabs that manufacture many different products with short cycle times [21]. The ADC system is typically trained with relatively few samples, for example, ten examples per

class, therefore resulting in a class representation that is limited to a small fraction of the universe of images that are generated by tools and inspection processes. This is represented in Figure 6a by the small number of points shown in each class region. The training set also defines the partition of the class region (e.g., the shaded areas in Figure 6a), which can vary greatly depending on the training data and classifier method used. The ADC system has evolved to perform the function of associating defects with labels (e.g., tungsten particle, missing pattern, poly flake, etc.) and therefore has the potential to be correct or incorrect, and the classification process is only an intermediary step towards associating the label with an errant manufacturing process, that is, defect sourcing. And finally, ADC is defect-centric in that training and execution of the classification procedure focuses primarily on the defect itself, and largely ignores the substrate as an identifying characteristic of the image or errant manufacturing process.

Conversely, the AIR system, shown in Figure 6b, performs the function of image retrieval based on a query vector, **Q**. The AIR system organizes and maintains multiple sources of images in one system (e.g., optical and SEM, multiple layers, steps, tools, etc.) and an image-based query will retrieve a specified number of images from the database that are close to the query in the sense of visual similarity (e.g., based on an $L$-norm distance in feature space). Therefore, the AIR system does not perform classification and does not assign a query point to a predefined label. When the database of images is coupled with the manufacturing data that describes the fabrication process (e.g., layer, step, lot, date, inspection tooling, EDX spectra, multiple modes of imaging such as optical, SEM, and confocal) it becomes possible to associate the query vector, **Q**, with visually similar historical images from the database, therefore, linking the query image directly to the process and potentially the source of the problem. And, since the AIR system focuses on both an extensive defect and substrate description, the association of defects with products, substrates, process steps, and layers is inherent in the analysis. An AIR system does not require training and its ability to comprehend a large population of images from multiple inspection tools and processes over a long period of time means that the limitations of ADC associated with focused training scenarios and frequent modifications to accommodate new products and process drift do not apply to AIR as they do with ADC.

## 5. FIELD TESTING AND RESULTS

ORNL performed two field tests of the AIR software system during the fall of 2000 for the purpose of verifying the fundamental premise that *a similar manufacturing process or phenomenon likely generates images that are visually similar*. It was also desired that testing in a manufacturing environment be performed to determine information regarding system robustness, timing, capacity, usability, and what fab data was key to sourcing problems based on defect imagery. Additional defect information (defect position on the wafer, wafer ID, Lot No., etc.) had been incorporated into the ORNL AIR system through the use of additional database tables and established relationships (i.e., foreign keys). The system was deployed at two semiconductor manufacturing sites to demonstrate the utility of this approach in managing large databases of images and to show causal relationships between image appearance and wafer information such as layer, lot, dates, and so forth. This section summarizes the results of these field tests and demonstrates the utility of this approach through data analysis conducted on approximately one month of historical defect data at the two independent fabrication sites.

Table 2: Database statistics for Sites 1 and 2.

| Value | Site 1 | Site 2 |
|---|---|---|
| Number of defects | 59 593 | 76 653 |
| Number of wafers | 3 856 | 3 336 |
| Number of lots | 1 375 | 1 021 |
| Number of step/layers | 99 | 164 |
| Number of images | 62 594 | 78 953 |
| Oldest date | 10-7-2000 | 9-14-2000 |
| Latest date | 11-6-2000 | 11-1-2000 |

### 5.1. Architecture and implementation

The AIR system and architecture was described in detail in Section 3. For field-testing, a method for maintaining and associating process information with defect imagery was created. Although the AIR field test software was not designed to be a complete defect management system, it was necessary to include some DMS-type functionality to reach our project goals. Toward this end, we envisioned the submission of images to our system as the result of a defect detection during inspection. During our design process, we developed database tables containing several relevant entities. These entities included the *Image*, which stores the file name associated with the image along with the feature values that describe its content; the *Defect*, including the classification of the defect, its location on the wafer and die, and so forth; the *Inspection*, a single act of taking one wafer and running it through an inspection on a defect detection instrument; the *Wafer*, an entity containing a set of die and possibly one or more defects; and associated tables of defect classifications and inspection tool types. The tables were embodied in a software object coded in the AIR field test DLL.

### 5.2. Experimental results

Table 2 shows the database statistics for each of the manufacturing sites after approximately one month of data collection. The table shows the total number of images associated with the various defects (i.e., there can be more than one image of each defect generated by different inspection tools), and the number of wafers, lots, and process steps.

Regarding system performance there are generally two times of interest to the semiconductor fab user. First, the

Table 3: Timing statistics for image addition and retrieval.

| Value | Site 1 | Site 2 |
|---|---|---|
| Addition mean | 0.834s | 0.476s |
| Addition median | 0.765s | 0.328s |
| Addition maximum | 6.0s | 3.813s |
| Addition minimum | 0.312s | 0.016s |
| Theoretical daily rate (Max.) | 103 598 images | 181 590 images |
| Retrieval time (128 images) | 7.5s | 7.25s |
| Retrieval time per image | 0.12 ms | 0.09 ms |

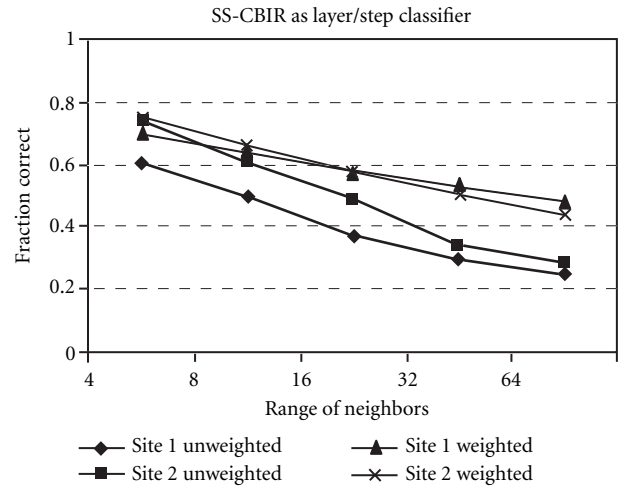
time to add images to the database is important because the AIR system should basically be invisible to the underlying defect detection and inspection activity. Second, retrieval time is important because of usability issues and engineering response time.

Table 3 lists the timing statistics for data from the two test sites. For the purposes of comparison on a common platform, the data sets from each test site for the initial month of testing (i.e., 62 594 images from Site 1 and 78 953 images from Site 2) were loaded on a common machine using data that had been collected and returned to ORNL. The machine used was a 750 MHz Pentium PC. The median, mean, maximum, and minimum time to add the images to the database are recorded in Table 2, along with image retrieval time. The image retrieval time was determined by requesting 128 returned images and measuring the system response for each database. The time to load images from a network and display them is not included in this total. Both these sets of times show a very acceptable rate of performance, allowing an overall daily sustained input of well over 100 000 images. The main difference between the timing for the sites is the image size; most images from Site 2 were JPEG, 320 × 240 images, while Site 1 images were JPEG, 640×480 or 480×480.

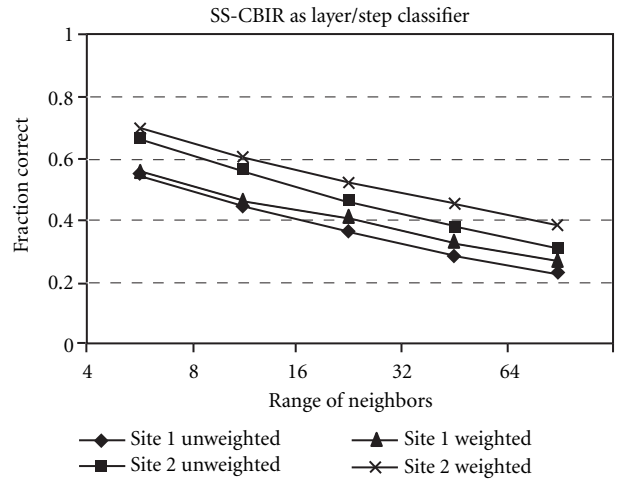
Then, we have modeled the retrieval system as a $k$-nearest-neighbor ($k$-NN) classifier for the step/layer and lot classification categories. The experiments were performed as follows. For each site, we sampled 1024 images from each database and submitted them as query images returning 64 results. We then counted how many times the most common occurrence in the results matched the selected parameter in the query image. For example, we determined the layer/step with the most common occurrence in the first 4, 8, 16, 32, and 64 returned images. If the most common occurrence matched our query image, the query was assigned a value of 1 (for success). Ties were assigned a value of 0.5 for unknown, and if no matches were returned, a value of 0 was assigned.

Figure 7 shows the results of this $k$-NN test. Each chart in the figure contains weighted and unweighted results for Sites 1 and 2, with classifiers using the first 4, 8, 16, 32, and 64 returned results. Unweighted results are computed by finding the number of correct classifications for a given layer/step or lot class, then averaging them.

This number considers all layer/steps and lots equally and does not depend on the number of occurrences of each class in the data set. Weighted results are computed by determining the number of correct answers and adding them, then dividing by the total number of queries. The performance drops as more neighbors are considered because images that are further down the list of retrieved images become visually dissimilar to the query and therefore are less likely to come

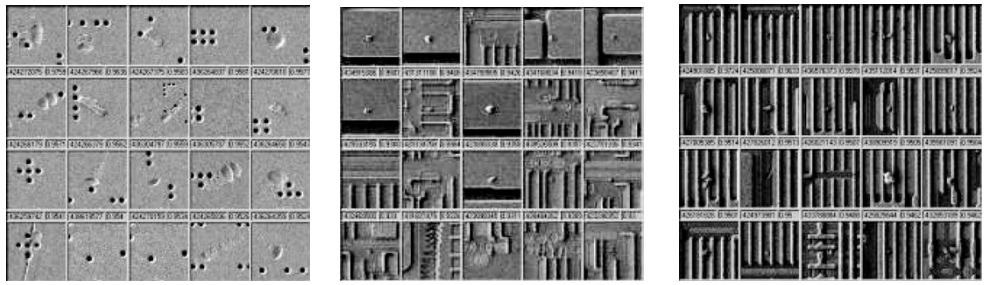



(a)



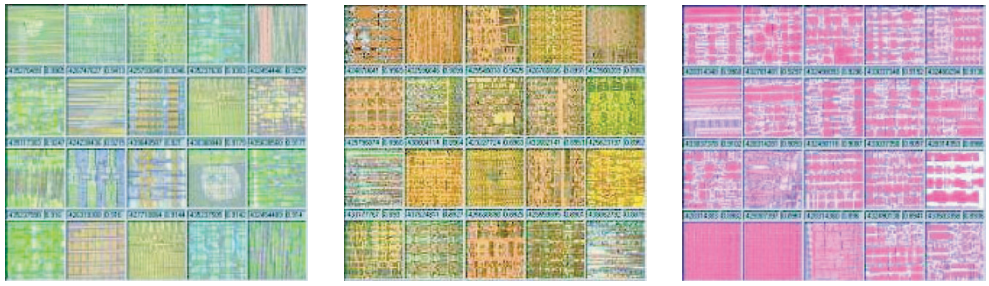

(b)

Figure 7: Results using the AIR field test system as a classifier. Plot (a) shows layer/step, and (b) lot $k$-NN classification results. Note that SS-CBIR refers to "semiconductor-specific CBIR."



(a) (b) (c)

Figure 8: Examples of SEM image query results for three different substrates on a semiconductor device. Note that the query image in (a), (b), and (c) are always the upper left image in the matrix.



(a) (b) (c)

Figure 9: Examples of optical image query results for three different substrates on a semiconductor device. Note that the query image in (a), (b), and (c) are always the upper left image in the matrix.

from the same source. Note that in Figure 7a there are 99 individual process steps represented in the Site 1 data and 164 in the Site 2 data. In Figure 7b, there are 1 375 lots and 1 021 lots representing Sites 1 and 2 , respectively. In all three of these k-NN comparisons, the number of times the top four returned images matches the query for the indicated parameters averages around 70%—which supports our hypothesis regarding visual similarity and manufacturing processes. This also reveals much about the AIR system's ability to perform classification in a very complex data environment. For example, in Figure 6b, the system correctly classifies about 62% of the lots tested for 4-NN. This represents a single classifier system that can differentiate over 1 000 individual classes.

Finally, Figures 8 and 9 show query results from both SEM and optical imaging review tools, respectively. Note the difference in substrate specificity apparent across the SEM images in Figures 8a, 8b, and 8c. Figure 8a contains a substrate containing vias to a sublayer on the device, while Figures 8b and 8c contain increasingly dense structure. The ability of AIR to group these data together rapidly out of a large image repository has proved to be the only means available today to accomplish this task. Figures 9a, 9b, and 9c. show results of images in the same database as Figure 8 but from color optical microscopy review systems (i.e., these images are 24-bit, RGB color images). The detail apparent in the thumbnails that are shown are less apparent due to the change of magnification (i.e., field of view) in going from SEM to optical microscopy. Optical images reveal color as a function of the thickness of the semitransparent films being printed on the device whereas SEM imaging is a near-surface effect due to the small penetration distance of the primary electron beam. The notable quality of the AIR system is that it can manage image data from such diverse imaging modalities within one system.

## 6. CONCLUSIONS

In this paper we have described a novel content-based image retrieval and management system that has been designed specifically for manufacturing environments. The manufacturing focus of the ORNL CBIR application takes advantage of the way in which defects are detected with standard industry inspection equipment by uniquely describing the defect and the substrate areas of the image independently in terms of color, texture, structure, and shape. Current image retrieval systems for semiconductor manufacturing depend on additional alphanumeric data to perform retrieval functions (e.g., lot number, time/date, wafer ID, etc.), which produces an inherent limitation to the process of locating historic imagery that may have been caused by a similar manufacturing process. AIR overcomes these limitations. The AIR system has been installed in two semiconductor manufacturing sites to determine system performance and retrieval

characteristics. The system was shown to perform exceptionally well in terms of storage capacity and the time required to add and retrieve images and process data to the system. Finally, we were able to demonstrate our fundamental premise that *a similar process or phenomenon likely generates images that are visually similar* by performing a series of $k$-NN classification tests to associate queries with process parameters such as process step and lot number. Without the addition of content-based image retrieval, this large image repository of semiconductor images will remain virtually untapped as a resource for rapidly resolving manufacturing problems. The application of the ORNL AIR technology to other manufacturing environments that generate large amounts of product imagery during defect inspection and quality control is inherent.

## ACKNOWLEDGMENTS

The authors wish to acknowledge the Member Companies of International SEMATECH who participate in the Yield Management Tools Program in Austin, Tex, for their advice, fabrication data, direction, and field test support for our research in automated defect data reduction technologies and new strategies for yield learning. This paper was prepared by the Oak Ridge National Laboratory, Oak Ridge, TN 37831-6285, operated by UT-Battelle, LLC for the U.S. Department of Energy under contract DE-AC05-00OR22725.

## REFERENCES

[1] V. N. Gudivada and V. V. Raghavan, "Content-based image retrieval systems," *IEEE Computer*, vol. 28, no. 9, pp. 18–22, 1995.

[2] S. Santini, *Exploratory Image Databases, Content-Based Retrieval*, Academic Press, San Fransisco, Calif, USA, 2001.

[3] E. Vicario, Ed., *Image Description and Retrieval*, Plenum Press, New York, NY, USA, 1998.

[4] Y. Gong, C. H. Chuan, and G. Xiaoyi, "Image indexing and retrieval based on color histograms," *Multimedia Tools and Applications*, vol. 2, no. 2, pp. 133–156, 1996.

[5] V. E. Ogle and M. Stonebraker, "Chabot: retrieval from a relational database of images," *IEEE Computer*, vol. 28, no. 9, pp. 40–48, 1995.

[6] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[7] J. You, H. Shen, and H. A. Cohen, "An efficient parallel texture classification for image retrieval," *Journal of Visual Languages and Computing*, vol. 8, no. 3, pp. 359–372, 1997.

[8] B. M. Mehtre, M. S. Kankanhalli, and W. F. Lee, "Shape measures for content based image retrieval: a comparison," *Information Processing and Management*, vol. 33, no. 3, pp. 319–337, 1997.

[9] J. E. Gary and R. Mehrotra, "Similar shape retrieval using a structural feature index," *Information Systems*, vol. 18, no. 7, pp. 525–537, 1993.

[10] R. Mehrotra and J. E. Gary, "Similar-shape retrieval in shape data management," *IEEE Computer*, vol. 18, no. 9, pp. 57–62, 1995.

[11] Y. Rubner and C. Tomasi, *Perceptual Metrics for Image Database Navigation*, Kluwer Academic, Boston, Mass, USA, 2001.

[12] C.-C. Hsu, W. W. Chu, and R. K. Taira, "A knowledge-based approach for retrieving images by content," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 4, pp. 522–532, 1996.

[13] P. W. Huang and Y. R. Jean, "Design of large intelligent image database systems," *International Journal of Intelligent Systems*, vol. 11, no. 6, pp. 347–365, 1996.

[14] B. Tao and B. Dickinson, "Image retrieval and pattern recognition," in *Multimedia Storage and Archiving Systems*, vol. 2916 of *SPIE Proceedings*, pp. 130–139, Boston, Mass, USA, 1996.

[15] Semiconductor Industry Association, "International technology roadmap for semiconductors," 1999.

[16] K. W. Tobin, T. P. Karnowski, and R. K. Ferrell, "Image retrieval in the industrial environment," in *Machine Vision Applications in Industrial Inspection VII*, vol. 3652 of *SPIE Proceedings*, pp. 184–192, January 1999.

[17] T. P. Karnowski, K. W. Tobin, R. K. Ferrell, and F. Lakhani, "Content-based image retrieval for semiconductor manufacturing," in *Machine Vision Applications in Industrial Inspection VIII*, vol. 3966 of *SPIE Proceedings*, pp. 162–172, March 2000.

[18] K. W. Tobin, T. P. Karnowski, L. F. Arrowood, and F. Lakhani, "Field test results of an automated image retrieval system," in *12th Annual IEEE/SEMI Advanced Semiconductor Manufacturing Conference and Workshop*, Munich, Germany, April 2001.

[19] K. W. Tobin, T. P. Karnowski, and F. Lakhani, "Integrated applications of inspection data in the semiconductor manufacturing environment," in *Metrology-based Control for Micro-Manufacturing*, vol. 4275 of *SPIE Proceedings*, pp. 31–40, 2001.

[20] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions," in *Proc. 5th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 573–582, Arlington, Va, USA, 1994.

[21] K. W. Tobin, T. P. Karnowski, S. S. Gleason, D. Jensen, and F. Lakhani, "Using historical wafermap data for automated yield analysis," *Journal of Vacuum Science Technology*, vol. 17, no. 4, pp. 1369–1376, 1999.

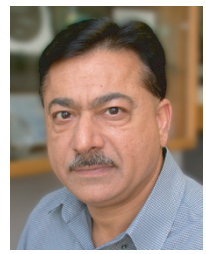
**Kenneth W. Tobin** leads the Image Science and Machine Vision Group at the Oak Ridge National Laboratory, Oak Ridge, Tenn. He has authored and coauthored over ninety technical publications and he currently holds four U.S. patents with three additional patents pending in the areas of computer vision and photonics. He has been performing research with the semiconductor industry since 1991 for rapid yield learning that includes image defect classification, wafermap spatial signature analysis, and content-based image retrieval. Dr. Tobin is a fellow of the International Society for Optical Engineering (SPIE) where he is currently serving as the Conference Chairman for Process Control and Diagnostics in IC Manufacturing, part of SPIE's Microlithography Symposium, Santa Clara, Calif. He is a contributing member of the Semiconductor Industry Association's International Technology Roadmap for Semiconductors, supporting the Yield Learning section of the Yield Enhancement roadmap. Dr. Tobin has a B.S. in Physics from Virginia Tech., Blacksburg Virginia, an M.S. in nuclear engineering and a Ph.D. in nuclear engineering from the University of Virginia, Charlottesville, Virginia.

**Thomas P. Karnowski** is a Development Associate and Research Scientist with the Image Science and Machine Vision group in the Engineering Science and Technology Division of Oak Ridge National Laboratory. Mr. Karnowski has over 11 years experience solving problems in systems integration, software design and development, and hardware development. His work has primarily focused on imaging and signal processing problems in industrial and military environments. Although he is an innovative thinker who always seeks to broaden his technical expertise, he is also very results oriented in his problem solutions. He has authored or coauthored over 15 publications in the areas of digital signal and image processing and pattern recognition applied to industrial environments. He is a coinventor on three patents in the areas of industrial image processing and pattern recognition for solving semiconductor yield problems. Mr. Karnowski has a B.S. in electrical engineering from the University of Tennessee and an M.S. in electrical engineering from North Carolina State University.

**Lloyd F. Arrowood** holds B.A. and M.S. degrees and is completing a Ph.D. in Computer Science from The University of Tennessee, Knoxville. He is currently employed at BWXT Y-12 in Oak Ridge, Tennessee.

**Regina K. Ferrell** is a research and development staff member of the Image Science and Machine Vision Group at Oak Ridge National Laboratory. Her most recent work is in the field of content-based image retrieval for x-ray images. She has also been working in the area of nonreferential defect detection for semiconductor imagery. She is a coinventor on two pending patents on the retrieval of industrial images. Ms. Ferrell is a member of the Institute of Electrical and Electronics Engineers (IEEE). She has a B.S. and an M.S. in electrical and computer engineering from the University of Tennessee, Knoxville.

**James S. Goddard Jr.** received a B.S. degree from Georgia Tech, an M.S. degree from the University of Maryland, and a Ph.D. from The University of Tennessee, Knoxville in electrical engineering. He is presently a development engineer with the Image Science and Machine Vision Group at the Oak Ridge National Laboratory. Primary research interests include machine vision inspection, classification, and 3D vision. His recent vision-based research at ORNL includes semiconductor inspection and classification, industrial process characterization, and high precision 3D measurement applications.



**Fred Lakhani** is a Senior Member of technical staff (SMTS) and a program mentor for the yield management tools (YMT) organization at International SEMATECH (ISMT). He currently leads projects in computer-aided fault to defect mapping (CAFDM) and automated image retrieval (AIR). Completed projects, led by Mr. Lakhani, include SEM/ADC, Spatial Signature Analysis (SSA), and Yield Modeling. Prior to joining ISMT, Mr. Lakhani held a variety of engineering and management positions in device characterization and yield management. Over the last 20 years, Mr. Lakhani has published numerous articles in his field and has served on various technical advisory boards. He presently serves as the chair of the Yield Learning section of the Yield Enhancement (YE) roadmap for the 2001 International Technology Roadmap for Semiconductors (ITRS). He also serves as a cochair for the International SEMATECH Yield Council—a member company forum for sharing Best Known Methods in Yield Management. Mr. Lakhani has a B.S. in electrical engineering from the University of Texas at Arlington, an M.S. in electrical engineering from the Southern Methodist University and he is a graduate of the Institute of Managerial Leadership at University of Texas at Austin.