2. Advanced Methods for the Analysis of Semiconductor Manufacturing Process Data

Andreas König[1] and Achim Gratz[2]

[1] Technische Universität Kaiserslautern, Kaiserslautern D-67663, Germany; email: koenig@eit.uni-kl.de

[2] Infineon Technologies Dresden GmbH & Co. OHG, D-01076 Dresden, Germany.

The analysis, control, and optimization of manufacturing processes in the semiconductor industry are applications with significant economic impact. Modern semiconductor manufacturing processes feature an increasing number of processing steps with an increasing complexity of the steps themselves to generate a flood of multivariate monitoring data. This exponentially increasing complexity and the associated information processing and productivity demand impose stringent requirements, which are hard to meet using state-of-the-art monitoring and analysis methods and tools. This chapter deals with the application of selected methods from soft computing to the analysis of deviations from allowed parameters or operation ranges, i.e., anomaly or novelty detection, and the discovery of nonobvious multivariate dependencies of the involved parameters and the structure in the data for improved process control. Methods for online observation and offline interactive analysis employing novelty classification, dimensionality reduction, and interactive data visualization techniques are investigated in this feasibility study, based on an actual application problem and data extracted from a CMOS submicron process. The viability and feasibility of the investigated methods are demonstrated. In particular, the results of the interactive data visualization and automatic feature selection methods are most promising. The chapter introduces to semiconductor manufacturing data acquisition, application problems, and the regarded soft-computing methods in a tutorial fashion. The results of the conducted data analysis and classification experiments are presented, and an outline of a system architecture based on this feasibility study and suited for industrial service is introduced.

2.1 Introduction

The exponential increase of available computational resources leads to an explosive growth in the size and complexity of application-specific databases. In fact, today's industrial sites can produce so much data per day that the evaluation of potentially beneficial information and even complete storage become close to impossible. The monitoring of complex processes, for instance, in industrial manufacturing, however, requires online monitoring and decision making as well as ensuing extraction of nonobvious information and

structure of the data. This procedure of knowledge discovery and the online decision making serve to control the respective complex processes, e.g., for quality assurance purposes, keeping the process in a multivariate window of allowed parameter tolerances.

One important instance of this general problem class with a significant commercial impact and stringent information processing demands is represented by the analysis, control, and optimization of manufacturing processes in the semiconductor industry. Typical aims are the centering of the process in a so-called process window and the assurance of an optimum yield based on functional and electrical tests. For instance, in [2.53] a good general introduction to the topic can be found. In this particular work, decision trees are applied to determine significant individual variables or groups of variables. A more focused example is given in [2.3], where data mining and various classification techniques are applied to a single processing step dealing with wafer cleaning. Leading-edge technology and the corresponding manufacturing lines have reached an unprecedented complexity in terms of both required machinery and the required process monitoring, control, and optimization demands. Thus, modern semiconductor manufacturing processes feature an increasing number of processing steps with an increasing complexity of the steps themselves from initial wafer preparation to final passivation. Due to the continued validity of Moore's exponential growth law (see, e.g., the SIA ITRS roadmap [2.2]) the complexity of the processes will continue to increase at a rapid pace. In Section 2.2.2, a brief introduction to this part of the presented work will be given. Consequently, a tremendous amount of monitoring data are generated by the manufacturing line. The generated data have to be analyzed with regard to the required process specification or qualification, i.e., whether the process remains in the process window (see Fig 2.1). In simple models, the process window can be described, e.g., by a multiparameter or multivariate bounding box with thresholds in each parametric dimension. Exceeding the threshold makes overt that the process is going out of specification for one or several of the involved parameters. This approach neglects multivariate dependencies and higher-order correlations of variable groups. Figure 2.2 depicts typical problems occurring, such as the process being off-centered or showing correlated parameters or multimodality. The same holds for the typ-

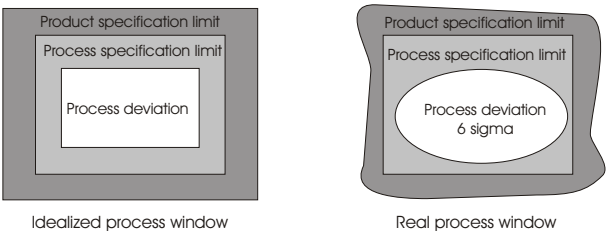

Fig. 2.1. Illustration of a process window.

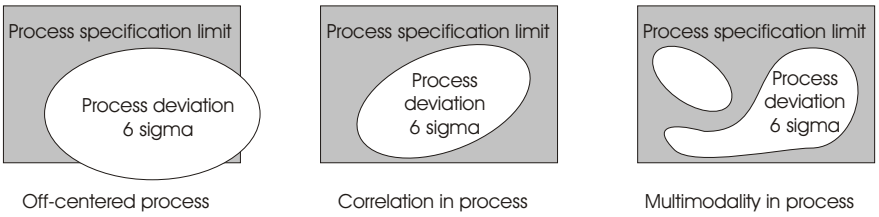

Fig. 2.2. Illustration of process window problems.

ical statistical analysis approach employed for the analysis and evaluation of process-related data. Individual parameters are checked for model consistency with regard to univariate, typically Gaussian assumptions. Further, methods like principal component analysis are used, which by its nature is a linear and parametric approach and, thus, is of limited applicability for nonlinear cases not obeying a multivariate Gaussian model. The significant economic potential of the data mining field in general and the field of semiconductor process data analysis in particular has triggered many activities. Numerous statistical tools with interactive visualization have recently become available. For instance, for the semiconductor industry, tools like dataPOWERsc [2.51], Knights' Yield Manager [2.52], or Q-Yield [2.7] are on the market. These tools dominantly apply parametric first order methods, i.e., methods based on the statistical information of a single variable or the correlation of two selected variables.

Thus, for the cases regarded earlier, advanced methods from soft computing originating from the fields of pattern recognition, neural networks, bio-inspired computing and statistics, and corresponding tool implementations provide improved leverage by multivariate, nonparametric, and nonlinear approaches. In Section 2.3.1, specific methods and their potential for advanced process window modeling and detection of deviation from the process window in (semi)automatic operation are briefly presented.

For the offline analysis of the multivariate process data as a baseline for ensuing process control and optimization, advanced methods for efficient multivariate data dimensionality reduction and interactive visualization can be salient. The benefit is given in terms of capturing multidimensional relations in the data, transparency as well as speed in the process of analysis, and knowledge extraction. In prior work of other groups, e.g., Goser's group in Dortmund [2.38] [2.14], Kohonen's self-organizing map (SOM) has been applied. In an enhancement of this work Rückert et al. [2.47] have developed the dedicated tool DANI for the analysis of semiconductor data of Robert Bosch GmbH. In this kind of application, the topology-preserving and dimensionality-reduction mapping properties of the SOM are exploited in conjunction with visualization enhancements, as, e.g., the U-Matrix of Ultsch [2.54]. The properties of the SOM and other neural networks have

also been employed in the *smart fabrication* project from 1995 to 2000 by a consortium including TEMIC, Siemens, and the University of Tübingen (Rosenstiel et al.). The detection of characteristic failure patterns [2.36] and yield prediction [2.37] were some of the pursued goals in this project.

In these and similar efforts, Kohonen's SOM has been employed with static visualization techniques. The advanced methods investigated here, however, differ in many ways and especially target on bringing improvements with regard to mapping speed, mapping error reduction, user convenience, and interactivity in the analysis process. The respective methods briefly browsed in Section 2.3.2 can serve to project data in a lower-dimensional space to make it amenable for interactive human perception–based analysis as well as automatic variable or variable group selection and pattern clustering. The objective of the current phase of the work and this chapter is to demonstrate the viability of the addressed methods for real process data extracted from a modern CMOS process. As a feasibility study, data with known but nonobvious information content prove that the methods can indeed help in rapidly detecting the desired information. In the second phase of the feasibility study, novel information and knowledge shall be extracted from additional process data by applying the proposed methods, e.g., interactive multivariate data visualization. In this regard, the chapter is as organized as follows. In the next section, the general data acquisition process and the chosen instance data for the conducted experiments are described. In the following section, the spectrum of applied methods and their tool implementations are covered. Then the conducted experiments and the achieved results are presented and discussed. Before concluding, the envisioned perspective of the work and the related information processing architecture for manufacturing process monitoring and optimization are introduced.

2.2 Semiconductor Manufacturing and Data Acquisition

2.2.1 Brief History of the IC

Semiconductor devices had a slow start as a curiosity that was not well understood. Still, they had important niche applications in radio communications, when vacuum tubes could not be used. As the understanding of their principles of operation grew, refinements to the manufacturing process first enabled military applications and then delivered the first commercially available devices in the form of single-pn-junction diodes and transistors in the early 1950s. The year 1958 marked the birth of the monolithic integrated circuit, now commonly just called IC. The invention of the IC is attributed to TI engineer Jack Kilby, but without the planar manufacturing process developed in the same year by Jean Hoerni and advanced by Robert Noyce and Gordon Moore at Fairchild,[3] it would likely have taken quite a bit longer for the idea

[3] R. Noyce and G. Moore left Fairchild to cofound Intel in 1968.

to take off. Meanwhile, also at Fairchild, a group of researchers[4] were getting a handle on manufacturing stable metal-oxide semiconductor (MOS) field effect transistors. They had actually been invented decades before the bipolar transistor, but irreproducible characteristics and fast degradation had prevented their application. The MOS transistor came back into focus because as a surface device it is a natural match to planar processing. In 1963 complementary MOS,[5] or CMOS, now the dominant technology for ICs, was invented. In the April 1965 issue of *Electronics* [2.40], Gordon Moore boldly predicted[6] that the number of components per IC would double each year at least through 1975. Depending on how you count components, the actual doubling interval turned out to be 18 months, but the general pattern of exponential growth has proven to be accurate for more than 40 years, with no end in sight. One of the important consequences is that the smallest feature F of an IC has to be halved about every three years. The diminishing of the feature size is commonly called technology scaling or shrinking, derived from the fact that at larger feature sizes it sufficed to simply draw the layout of an IC at a smaller scale to go from one technology generation to the next (provided the new technology was designed to be compatible with the old). As F becomes smaller, it becomes more difficult, if not impossible, to keep this strict compatibility between technology generations; however, there are design tools to "scale" IC layouts down to the new generation while making these differences transparent. Technologies with an F of 0.13 μm are in production right now, and the next technology generation with sub-100-nm structures is imminent. These ICs will integrate more than 100 million transistors.

2.2.2 IC Production Process

The prevalent technology for producing ICs today is CMOS on silicon. The silicon substrate (called the wafer) is sliced off of a single crystal of extremely pure silicon (the ingot) at a precise angle with respect to the cristallographic orientation. The wafers are then polished to achieve an atomically smooth surface and extreme flatness. Currently, wafer diameters of 200 mm are most common, while 300 mm wafers just being put into production.

The actual IC production process takes place in clean rooms, at the so-called fab floor. Clean rooms are classified by the number of particles larger than a certain size in a cubic meter of air. A laminar flow of air from the ceiling to the bottom is maintained to quickly remove any particles becoming airborne. The IC production process is roughly divided into the wafer or frontend processing, wafer test, and the back-end processing where the chips are singulated, packaged, and subjected to more tests. Commonly test and

[4] One of them was Andrew Grove, later to become Intel employee number 4.

[5] Thus far, MOS IC technology had employed only n-conducting devices, which led to the name NMOS technology.

[6] In various forms, this prediction is now known as Moore's law. Beyond that prediction, this article is an elucidating read even today, almost 40 years later.

packaging make up more than 50% of the production cost. Wafer processing takes places in a so-called wafer fab or manufacturing line and is often further divided into front-end-of-line (FEOL) and back-end-of-line (BEOL) processing. Simply speaking, the FEOL processing provides the active devices within the silicon, BEOL processing produces the connections between the devices, and the back-end processing provides the connections to the outside world as well as protective packaging. To simplify the fab logistics, wafers typically run in lots of 25, [7] although some tools demand batches (see Fig. 2.3) of up to six lots to be used effectively, while other tools can't process a complete lot, which will then be split into smaller batches or even single wafers.

All wafer processing, whether FEOL or BEOL, has the same general structure of producing so-called layers, one after another. The whole wafer is subjected to some processing, like producing a thin film of oxide or metal. Then a mask is transferred to the wafer, most commonly by optical lithography, to selectively protect parts of the wafer from the following process steps. Then the wafer is subjected to further processing, like etching or implantation of ionized dopants. Manual and automatic inspections are inserted at various stages (Fig. 2.4). Then the mask is removed and the next layer is processed. Layers vary widely in the number, complexity, and cost required to make them. This leads to a distinction between critical and uncritical layers. Modern technologies make use of 20 to 30 layers, and this number continues to go up. The number of layers that make up the actual devices stays relatively constant. However, as the minimum feature size F continues to shrink, the exponentially growing number of devices requires much more interconnect between them. For this reason, the number of interconnect or metal layers in the BEOL, another commonly cited characteristic of a technology, increases quite rapidly. In fact, the interconnect of the devices (the BEOL) is now more costly to produce then the devices themselves (the FEOL).

2.2.3 Data from the Fab – Inline Data

Historically, each lot was accompanied by a stack of paper, called the process record. Each sheet detailed one process step and the operator would set up the tool accordingly, run the process, sign off, note remarks and the result of any measurements taken, look up the next operation, and hand the lot over to the next operator. This is really where the term semiconductor manufacturing stems from. The process record has been replaced[8] by a database and the lots are moved to the next operation by automatic transport systems (Fig. 2.5) coupled to that database. The so-called process flow is defined by the layer sequence at the top level. This has to be broken down into individual process steps, often called moves. Each of the process steps is made up of a

[7] The lot size is somtimes reduced to 12 wafers for 300-mm wafers, as a lot of 25 wafers is too heavy to be handled manually.

[8] Some fabs still use printouts to accompany the lots.

Fig. 2.3. Batched 300-mm wafers ready to go into a vertical furnace (open furnace tube on the upper left).

sequence of operations (called a recipe) within the tool. It is now common to have so-called cluster tools comprising of multiple stations capable of running a variety of processes, so a recipe can be quite complex.

While the process record has been moved into an electronic database, it has also been expanded over time to contain more data. Measurement equipment will generally store results to a dedicated database before a result summary is attached to the process record. Additionally there are separate databases dedicated to certain tools or tool groups for recipe repositories and recording events and in situ measurements during processing. The trail of

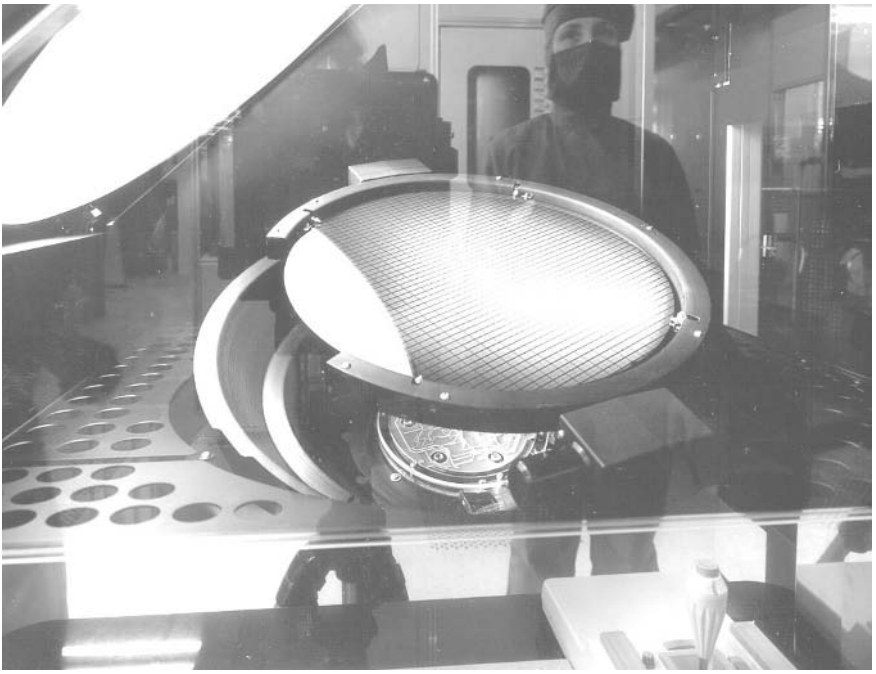

Fig. 2.4. A 300-mm wafer at so-called floodlight inspection to check for correct printing of the mask).

data collected about each lot is therefore scattered about various databases. Lately, single-wafer processing has become more important. Often the exact sequence of wafers through a single-wafer process or the position of wafers (respectively lots in batch tools) will be needed to pinpoint problems found with specific wafers. For so-called single-wafer tracking, this information needs to be fully recorded, which is only possible if all tools can read the wafer ID and lot information automatically and are connected to a database system. Additionally, a vast amount of (often temporary) data is produced and evaluated for inline process monitoring and closed-loop process control. It can be estimated that a typical semiconductor manufacturing line produces such data in excess of 1 TByte per day. It is therefore essential to evaluate, prune, and compact much of this data directly at the source. Routine reports are extracted for common purposes like maintenance, documentation, process control and optimization, and quality management. Process data that are actually stored, whether on the process tool itself or in a database, are usually kept only for a limited time or in a rolling log file to limit the storage requirements. This is far from an optimal solution as most of the data will be completely normal and therefore uninteresting, while crucial data needed to

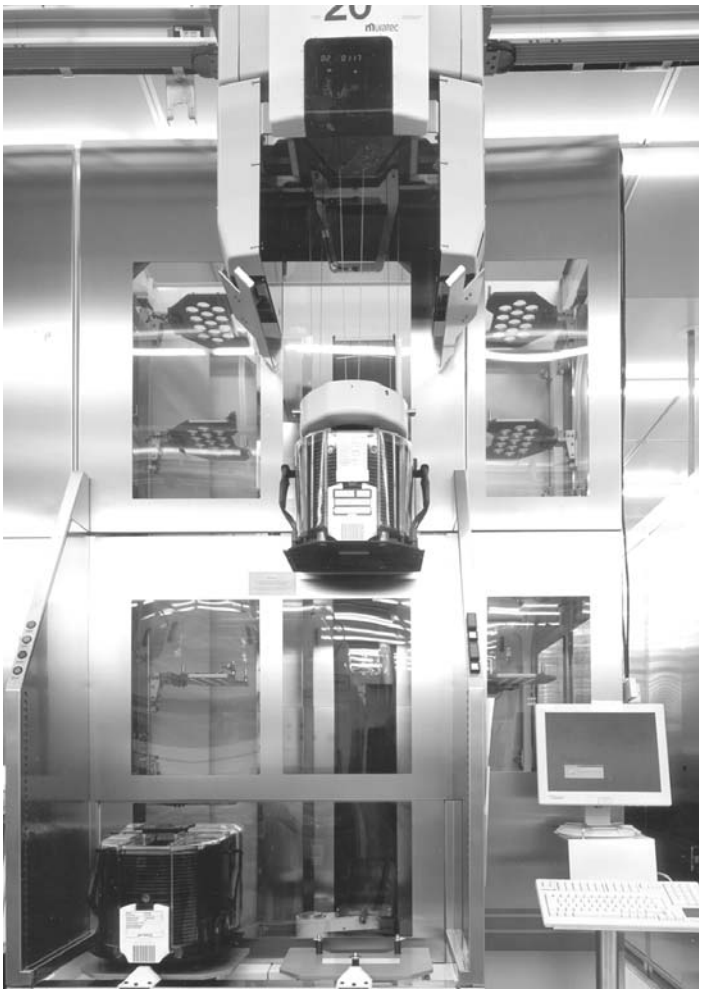

Fig. 2.5. Automatic transport system loading up a fully automatic wafer storage (Stocker).

analyze a process failure may already have been deleted before the anomaly is recognized and triggers a detailed investigation.

Collecting and evaluating all data for even a single lot are a formidable tasks. The resulting very large multivariate data set must therefore be analysed for deviations from allowed parameters or operation ranges, i.e., anomaly or novelty detection, and nonobvious multivariate dependencies of the involved parameters and the structure in the data must be disclosed for improved process control. Here, appropriate methods, e.g., from soft computing, for online observation and offline interactive analysis employing novelty clas-

sification, dimensionality reduction, and interactive data visualization techniques can be employed.

2.2.4 Electrical Test Data

After fabrication, electrical tests (ET) on the wafer level are carried out to assess that all single devices defined by the process are within their specified range. The devices tested are separate from the actual ICs on the wafer, often placed into the space between individual chips that is needed to singulate them later. These test structures are laid out carefully to isolate the layers needed to process them as much as possible from other layers. These tests are also called parametric tests as the results are actual measurement values for device parameters, like the threshold voltage or saturation currenrt of some specific transistor.

Later the actual ICs on the wafer are subjected to functional and parametric tests (FT and PT) on the wafer to decide which devices should be packaged after singulation. These tests are usually performed on a multitude of devices to save time. A sequence of tests is performed on each chip, and the first test that fails is recorded. The failed chips continue to be tested, but as the fail may have put it into an undefined state, the results of these tests cannot be relied on.

Both electrical and functional test data are stored in databases (s. Fig. 2.6) and is often preprocessed to facilitate analysis. Such preprocessing routinely includes the removal of spurious faults, calculation of derived values for parameteric data, and binning for functional data. Binning collects several individual tests that are associated with the same failure mechanism into a so-called fail bin.

Often the IC will again be tested after being fully packaged. When reliability is of utmost concern, a burn-in procedure may be performed to weed out early fails, necessitating further tests.

2.2.5 Data Analysis

Standard data analysis concentrates on keeping the process within specification limits, thus ensuring the quality of the final product. Typically a normal distribution of the measured parameter is assumed and parameters of the distribution like median and sigma are reported. In conjunction with the process specification limits, the so-called process capability *cp* and process centering *cpk* can be calculated. These methods and their application are widely accepted and mandated by various quality management methods and standards like ISO 9000.

However, their application to process specification, process trouble shooting, and process optimization often does not yield the desired results. Due to their univariate nature, complex interactions between parameters are not

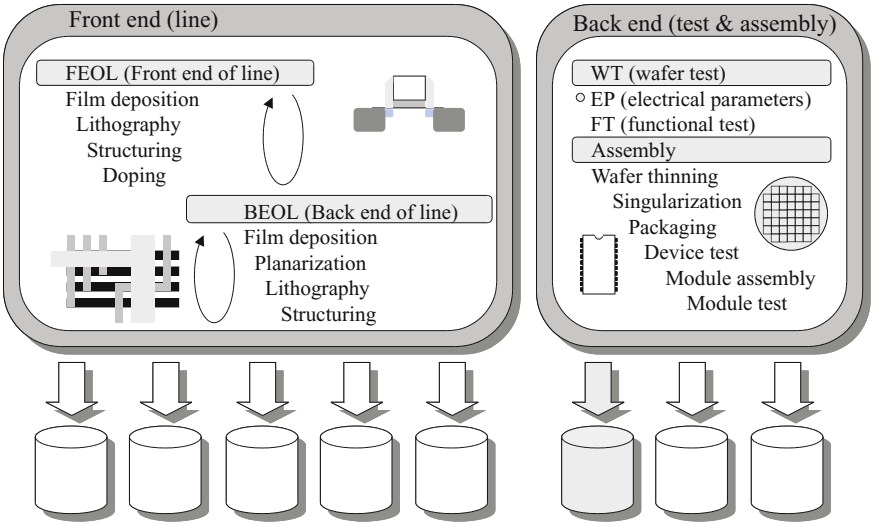

Fig. 2.6. Illustration of the overall process flow and the various origins of data.

taken into account. Also, while nonnormal distributions can in principle be accounted for properly, the procedure is cumbersome to implement and does not immediately address the failure mechanisms that change the shape of the distribution, for instance, to a multimodal distribution.

2.2.6 Process Experiment

Two lots of 25 wafers each were split identically into three groups at two process steps (s. Fig. 2.7) to vary the process parameters of these steps and in accordance the electrical parameters of certain devices. The intention of the split was to vary the threshold voltages of both n- and p-type logic transistors about the target voltage for each device. This is also called a performance split, indispensable for dynamic performance characterization, as it results in slow, nominal, and fast logic gates for the final product. As a side effect, some parameters related to the threshold voltage (most notably saturation current) and the so-called IO device coupled to the logic device will also follow the split.

This particular experiment was chosen because its effects are well known in advance and the analysis is reasonably tractable by conventional methods (Fig. 2.8). Thus there is an established baseline to compare the results of our newly developed data analysis methods against. We expect any successful method to reconstruct the split information in the two individual lots and to recognize that any residual differences between the two lots are not related to the splits, as the splitgroups are identical. Further, both the intended

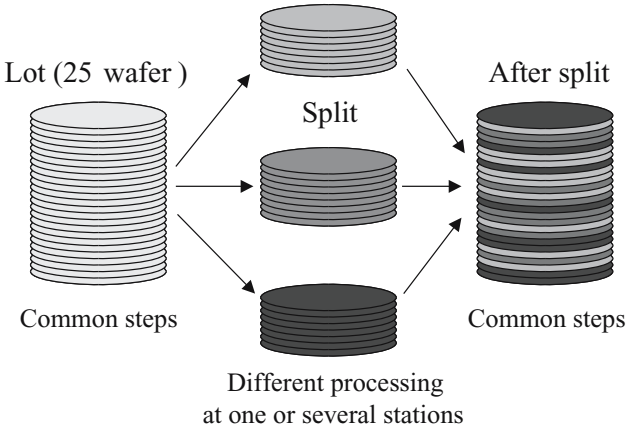

Fig. 2.7. Illustration of the split operation.

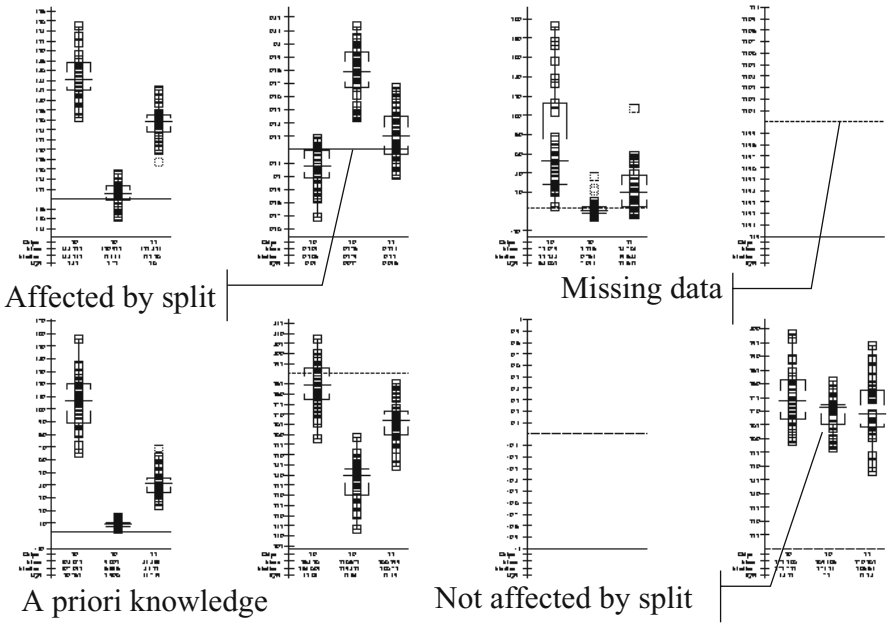

Fig. 2.8. Typical result from conventional statistical analysis. The three split groups have been separated by a priori knowledge and are shown in a single diagram to facilitate further evaluation of the experiment.

parameter changes and the side effects should be flagged as belonging to the split.

2.2.7 Experimental Data

From the proprietary software system and company database affiliated with the regarded manufacturing process, a subset of data generated for processing two wafer lots with five measurement positions for each wafer was extracted. A split of three, i.e., a partitioning of each wafer batch into three subgroups for individual processing of each partition, was carried out during production. Six wafers from the second lot were still staged in the fab for another experiment, so no data were available for these wafers. The electrical test data contain redundancies with regard to the particular split, as for each device specimen, different channel length and width are available. Also, as already explained, variation of the threshold voltage will influence further parameters belonging to that particular device. By means of conversion to an Excel spread-sheet and the application of a standard conversion tool, the database is converted to the QuickCog system requirements. The QuickCog system comprises all the methods discussed in this chapter, in particular the interactive data visualization methods and tools. A first database of 220 vectors with 205 dimensions will be regarded in the following experiments. It will be denoted by SPLIT in the following. The size of this database is given by the typical wafer batch size of 25 times the five measurement sites per wafer. However, the measurement values of six of the wafers from one set were not available, which reduces the data from the expected 250 to 220 samples. Larger databases could only be generated if larger wafer batches were made subject to identical split processing. With regard to the associated effort and cost, the aim of this work was to assess the applicability of the regarded methods also for rather sparse data of this application. No general limitation of the approach is implied by the choice of this practically relevant problem, as the regarded methods themselves scale well for large database sizes [2.23], [2.24].

Complementing the parameter data, class affiliations were generated in two files. A three-class file was generated, regarding split information only for the complete database. The data labeled by this class file will be denoted by SPLIT3 in the following. Additionally, a six-class file was generated according to lot and split affiliation of each wafer/measurement location. The labeled data will be denoted by SPLIT6 in the following. Additionally, according to the underlying lots the data have been separated into two databases denoted by SPLITTrain3 and SPLITTest3 with three classes each, corresponding to the underlying split of 3 of each lot. Finally, for the novelty classification purposes, a training set was extracted from the first lot containing data only from one split. This will be denoted by SPLITTrainOCC in the following.

2.3 Selected Soft-Computing Methods

In the following, we focus our investigations on two method groups. With the objective to achieve a (semi)automatic monitoring and control system, selected classification methods are regarded first. These shall serve the purpose of automatic assessment or classification of online generated process data with regard to its relevance and potential storage as well as the determination of the current process state within the process window. As the second group, methods for offline exploratory data analysis are regarded. We focus on relevant methods of dimensionality reduction and interactive visualization that allow us to extract nonobvious structure and underlying dependencies from the database. The results obtained using these methods also provide the baseline for the design of the effective (semi)automatic classification methods.

2.3.1 Novelty or Anomaly Detection

For the (semi)automatic classification task, powerful decision units are required that can deal with complex, nonlinear, separable, nonparametric, and potentially multimodal data. For instance k-nearest-neighbor classifiers (kNN), multi-layer perceptrons (MLP), radial-basis-function networks (RBF), or, more recently, support-vector machines (SVM) are attractive candidates for this task. In the context of the regarded application, dominantly decision trees, adaptive-resonance theory (ART) networks, and MLPs have been applied so far (see, e.g., [2.53] [2.36] [2.3]). However, especially RBF networks are intriguing for this application due to numerous salient features. In addition to being universal function approximators, RBF networks provide iterative topology learning, rapid training, fast convergence, and excellent predictable generalization capabilities [2.4], [2.43], [2.44]. In contrast to MLPs, the hidden layer of RBF networks comprises distance computation units equipped with a radially declining nonlinearity. The Euclidean distance and the Gaussian function are typical instances for RBF networks, which are closely related to the Parzen-Window technique [2.41]. However, storing all sample patterns is a significant burden with regard to storage and computation requirements. Thus, generalized RBF networks [2.4], i.e., networks with fewer hidden neurons N^* than training patterns N, are typically applied, which are given for the case of a one-dimensional function $s(\boldsymbol{x})$ by

$$s(\boldsymbol{x}) = \sum_{i=1}^{N^*} w_i \phi_i(\| \boldsymbol{x} - \boldsymbol{t}_i \|), \quad \boldsymbol{x} \epsilon \Re^M. \tag{2.1}$$

Here, t_i denotes the centroid vector of the basis function, ϕ_i denotes the radial basis function, w_i denotes the weight for the linear combination of the basis function outputs by the output neuron, $\boldsymbol{x}$ denotes an input vector, M denotes the dimension of the input vector, and N^* denotes the number of hidden neurons. Judicious and efficient choice of a sufficient but minimum

number N^* of hidden neurons is a major issue, especially for large scale problems. Several top-down and bottom-up strategies have been developed in the past [2.42] [2.15] [2.35] [2.30], employing and combining both supervised and unsupervised learning techniques. In a typical top-down strategy, a large number of centers will be determined by vector quantization techniques, e.g., Kohonen's self-organizing map. Fine-tuning of the network is achieved by a following supervised learning step, e.g., using gradient descent. Further network optimization and size reduction can be achieved by pruning techniques.

On the other hand, in bottom-up approaches the network is generated from scratch, thus completing a network-size tailored to the training data. The RBF network proposed by Platt [2.42] and the *restricted-Coulomb-energy* (RCE) network [2.46], [2.5] are significant examples of this category, as they allow dynamic automatic topology construction tailored to the problem requirements. This and an additional advantage of RBF-type networks make them excellent candidates for the investigations in this work. They also allow the concept of background classification (BC) to be implemented, which can be generalized from multiclass to one-class classification (OCC). BC is implemented by assigning the whole feature space to the selected background class. Other class regions are established by placing kernel functions and appropriately adjusting their widths during the learning process. Clearly, the network loses the rejection capability associated with the appearance of data far from the training samples. However, in cases like visual inspection or semiconductor manufacturing, in contrast to the plethora of potential errors, the desired condition can be described by sufficient examples. Thus assigning the background to such an error class can be advantageous. Initial ideas can be found in the *Nestor-learning-system* (NLS) [2.5], [2.6], which comprises a special RBF model denoted by RCE network [2.46]. The concept has been generalized to RBF networks in [2.20]. The special case of OCC, also addressed in the literature as novelty filtering [2.19] or anomaly detection [2.17], [2.31], [2.50], [2.33], is attractive because the classifier structure can be generated just by presenting data from a normal process situation. This is fortunate, as typically a lot of data from normal operation conditions are available; however, the universe of potential deviations is hard to grasp in terms of representative data samples actually covering all relevant regions in the high-dimensional parameter space for appropriate class border definition.

Thus, in the following, a model for OCC will be briefly derived from RBF-type networks for the regarded application domain.

The RCE Algorithm. The RCE network [2.46] is a special case of the RBF network given earlier. Instead of smooth nonlinearities as, e.g., the Gaussian function, a hard limiter or step function with a variable threshold parameter is applied. Each RCE basis function is equivalent to a hypersphere, represented by a center $\boldsymbol{t}_j$ and the threshold parameter, which has the meaning of a radius R_j. Each hypersphere is affiliated to one of the classes of the

application and gets activated if $S(||\boldsymbol{x} - \boldsymbol{t}_j|| <= R_j)$, i.e., if pattern $\boldsymbol{x}$ is situated within the hypersphere. The RCE output layer is also modified from a linear combination to an OR-like logic operation combining the hypersphere responses to determine the overall classification.

The algorithm practically requires only two parameter settings, $R_{\max}$ and $R_{\min}$, for operation. The following situations can arise in classification:

- A pattern is uniquely classified by one or several hyperspheres of the same class.
- No hypersphere is activated by the presented pattern. This defines a rejection mechanism, which can be controlled by setting $R_{\max}$ in training. A decision can be forced by, e.g., the nearest-neighbor rule. The rejection mechanism is replaced if the background is affiliated to one of the problem classes in BC.
- Several hyperspheres of different classes are activated by the presented pattern. The pattern is identified as ambiguous. A decision can be made according to the affiliation of the majority of the activated hyperspheres or by the nearest-neighbor rule.

The iterative RCE training algorithm starts with an empty network and presents all patterns of the training set until no more changes take place in the following basic training steps:

- If no hypersphere is activated by the presented pattern k, it is stored as $\boldsymbol{t}_{J+1}$ with $R_{J+1} = R_{\max}$, where J denotes the current number of reference vectors.
- A pattern is uniquely classified by one or several hyperspheres of the same class. All radii are left unchanged, the pattern is not stored.
- Several hyperspheres of the same and different classes are activated by the presented pattern. Radii of hyperspheres affiliated to different classes will be reduced until the pattern is no more included, or $R_j = R_{\min}$ is reached for the regarded hypersphere j. The pattern is not stored.
- Only hyperspheres of different classes are activated by the presented pattern. Radii of activated hyperspheres will be reduced until the pattern is no more included or $R_j = R_{\min}$ is reached. In the first case, pattern k will be stored with $R_{J+1} = ||\boldsymbol{t}_l - \boldsymbol{t}_{J+1}||$, i.e., the radius will extend just to the center of the closest or nearest-neighbor hypersphere l. In the second case, pattern k will not be stored.

With the choice of $R_{\min}$, the storage of vectors close to class borders can be suppressed, thus influencing network resubstitution and generalization properties. Evidently, patterns once stored in the RCE network will never be removed. Just the pattern radii will be reduced until $R_{\min}$ is reached. This means that the size and quality of the achieved network are determined by the order of presentation of training vectors. A probabilistic presorting of sample data for RCE (ProRCE) based on local probability estimation and sorting of the training presentation order proportional to the probability has

proven to be one beneficial extension of the method [2.20]. However, for the regarded application, the focus will be on the extension of RCE to BC and OCC.

Extension of RCE for OCC. As already addressed, it is of practical interest to derive a system that is trained just by available examples of one class and that detects samples from the other class, e.g., production errors or system malfunctions, as deviations from the normal state. An instance of such a system has been introduced in prior work for image processing. The NOVelty detecting ASsociative memory (NOVAS) [2.31] stores a number of multidimensional pixel images and generates for each pixel an internal representation of hyperspheres with uniform radii, which is quite similar to an RCE classifier with BC. The difference is that RCE with BC assigns one problem class as the background class and trains the radii of the remaining classes' hyperspheres according to the correct classification of training patterns from all classes. In case of OCC, no patterns will be available for the background class. So the hypersphere radii must be determined by an additional rule or method. RCE can heuristically be adapted to that aim by storing all selected examples of the normal class from the training set based on a prior computation of a radius $R_{\max}$ for all hyperspheres according to the maximum distance of two nearest neighbors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the normal class [2.31]:

$$R_{\max} = \max_{j=1}^{N}(\min_{\substack{i=1 \\ i \neq j}}^{N} ||\boldsymbol{x}_i - \boldsymbol{x}_j||). \tag{2.2}$$

After $R_{\max}$ computation, the normal training data can be completely stored as classifier reference data of the novelty classifier (NOVCLASS). Data vectors $\boldsymbol{x}_l$ from the monitored process can be classified with regard to their novelty by the following steps:

1. Compute the nearest neighbor $\boldsymbol{t}_{NN}$ of $\boldsymbol{x}_l$ in the prototype set $\mathbf{T}$ with:

$$d_{t_{NN}} = \min_{j=1}^{N}(\sum_{i=1}^{M}(x_{li} - t_{ji})^2). \tag{2.3}$$

2. Classify the pattern $\boldsymbol{x}_l$ as:

$$\boldsymbol{x}_l \quad \text{is} \quad \begin{cases} \text{normal} & \text{for} \quad (\sum_{i=1}^{M}(x_{li} - t_{NN\ i})^2) < R_{\max} \\ \text{novel} & \text{for} \quad (\sum_{i=1}^{M}(x_{li} - t_{NN\ i})^2) \geq R_{\max}. \end{cases} \tag{2.4}$$

The resulting novelty detection can be employed to perceive process deviations and filter data out as representing an important event worth storing. Deviations or anomalies are detected as patterns on the background, outside of the normal domain, similar to the BC mode of RCE in multiclass problems. This is illustrated for the two-dimensional case in Fig. 2.9. Employing an iterative presentation of the training data, data reduction in terms of stored vectors, similar to the original RCE classifier, could be achieved, trading off alleviation of storage requirements and real-time classification with

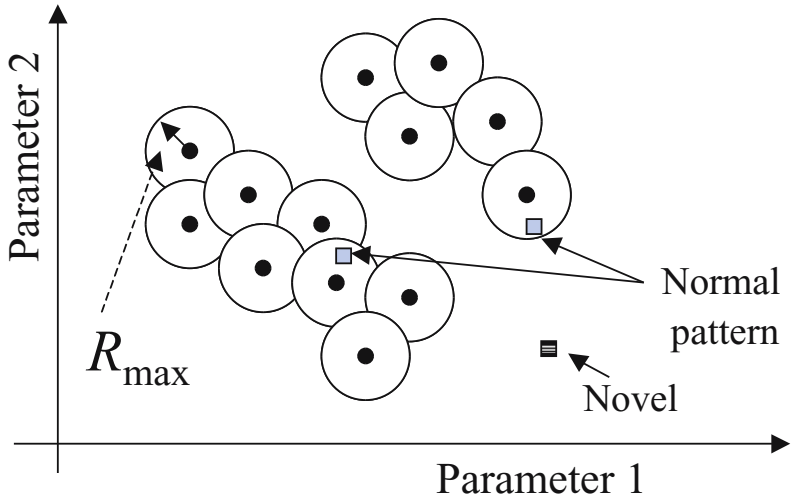

Fig. 2.9. Principle of OCC by NOVCLASS.

sufficient covering of the normal domain. In this iterative training case, a new hypersphere with center $\boldsymbol{t}_{J+1} = \boldsymbol{x}_l$ and radius $R_{\max}$ is added to the initially empty classifier iff a presented vector $\boldsymbol{x}_l$ from the training set is classified as novel by the already stored J reference vectors $\boldsymbol{t}_j$ according to the basic steps given earlier. The denseness of the NOVCLASS model potentially can be controlled by scaling the $R_{\max}$ parameter by a scale factor η to $\eta \times R_{\max}$ in the training process. A large-scale factor implies few stored vectors and potential coarse window modeling, whereas a small-scale factor $\eta < 1$ means fine window modeling at the cost of storing and processing a potentially large number of vectors. A functional nonparametric classifier is thus achieved with examples of just one class. Additionally, if at least a few examples for anomalies are available, these can be used to fine-tune the radii of the stored normal class hyperspheres by applying RCE-like adaptation for the conflicting hyperspheres. In this case, radii will no longer be uniform.

Currently, a prototype NOVCLASS version has been implemented and validated with modified Iris data, where all examples of class 3 were affiliated to class 2. Class 1 was chosen as the normal class. Resubstitution of the training set was perfect and in generalization just one vector slightly separated from the main cluster was misclassified.

Summarizing, the NOVCLASS algorithm allows both data reduction and arbitrary coverage of the parameter space. Thus, the concept of the process window is generalized to arbitrary shapes, including no convex boundaries. The current rather ad hoc uniform $R_{\max}$ computation approach could be improved by more sophisticated methods, e.g., locally adaptive radii computation, in future work. The present NOVCLASS implementation will be

applied to semiconductor application data for basic feasibility demonstration in Section 2.4.

2.3.2 Dimensionality Reduction and Interactive Visualization

Motivation. In addition to semiconductor manufacturing, a wide variety of other technical problems are characterized by typically large sets of high-dimensional data, obtained, e.g., from sensor registration, medical laboratory parameters, manufacturing process parameters, financial databases, measurements, or other generally observed features. With regard to the given application, significance, correlations, redundancy, and irrelevancy of the variables x_i are a priori unknown. The extraction of underlying knowledge or the reliable automatic classification requires reduction of the initial data set to the essential information and the corresponding variables. This especially holds, as the well-known curse of dimensionality (COD) [2.12] makes the compaction of the data a mandatory prerequisite for reliable decision making. Unsupervised and supervised methods can be employed for this reduction step for interactive and automatic processing of the data. The exploitation of the remarkable human perceptive and associative capabilities for the complex problem of identifying nonobvious correlations, structure, and hidden knowledge in the data can be a powerful complement of existing computational methods. Of course, an appropriate visual representation is required, which can be achieved by means of dimensionality reduction or multivariate projection methods combined with interactive visualization of the data [2.49]. Typical database representation, e.g., as an Excel spread-sheet is not easily amenable to human perception and understanding. This is illustrated in Fig. 2.10, together with the alternative human-adapted visual representation of the same database. Thus, dimensionality reduction is a ubiquitous problem and together with multivariate data visualization a topic of interest and interdisciplinary research for more than three decades. Applications of high economical interest, e.g., the one investigated in this work and other data mining and knowledge discovery applications, give renewed strong incentive to the field. Numerous methods were derived in the past for dimensionality reduction that considerably differ with regard to the methodology, computational complexity, transparence, and ease of use. In this work, effective methods promising the best productivity increase will be preferred. The following common definitions of two main groups of dimensionality reduction methods, briefly adapted from [2.16], shall clarify the pursued objectives. For a given sample set $\mathbf{X}$ with N M-dimensional feature vectors $\mathbf{x} = [x_1, x_2, \ldots, x_M]^T$ feature extraction is defined as a transformation

$$J(\mathbf{A}) = \mathbf{max}_{\mathcal{A}} \mathbf{J}(\mathcal{A}(\mathbf{v})) \tag{2.5}$$

and the special case of feature selection is defined as a transformation

$$J(\mathbf{A^S}) = \mathbf{max}_{\mathcal{A}^S} \mathbf{J}(\mathcal{A}^S). \tag{2.6}$$

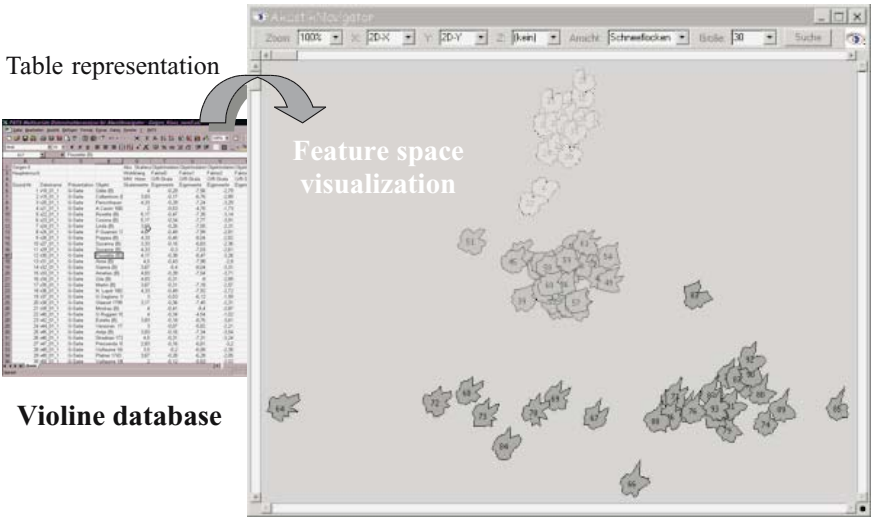

Fig. 2.10. Exploitation of human perceptive capabilities by appropriate presentation of multivariate data employing dimensionality reduction and interactive visualization.

While in selection, according to a chosen criterion J and the applied selection matrix A^S (see Eq. 2.13), the best features are retained and the remaining ones are discarded; in extraction all features are retained and subject to transformation A. In both cases a mapping $\Phi : R^M \rightarrow R^m$ optimizing a criterion J with $m \leq M$ and $\mathbf{y} = [y_1, y_2, \ldots, y_m]^T$ is determined. Here $\mathbf{y} = \mathcal{A}(\mathbf{v})$ can be a linear or nonlinear mapping and employ unsupervised as well as supervised information. The optimization criterion or cost function J can represent various objectives, e.g., signal preservation, distance preservation, topology preservation, or discrimination gain for the underlying L-class problem (see Fig. 2.12). For the latter case, selected instances of J will be given in the following. Figure 2.11 gives a taxonomy of state-of-the-art dimensionality reduction methods for multivariate data classification, analysis, and visualization in a unified presentation. This taxonomy has been elaborated on in the last few years and is continuously enhanced, including new methods. Most of the methods have been implemented in the QuickCog system [2.28] [2.29] and compared in previous survey publications [2.24] and tutorials [2.29] [2.22]. The taxonomy given in Fig. 2.11 covers methods as, e.g., the principal-component analysis (PCA) [2.12], scatter matrices (SCM) [2.12], Sammon's nonlinear mapping (NLM) [2.48], and accelerated heuristic variants, the nonlinear discrimination analysis method of Koontz and Fukunaga [2.32], or Kohonen's self-organizing map [2.19] (see also [2.24]). For visualization purposes, in this work distance-preserving nonlinear mappings, e.g., the one introduced by Sammon [2.48] have been applied. Interpoint distances

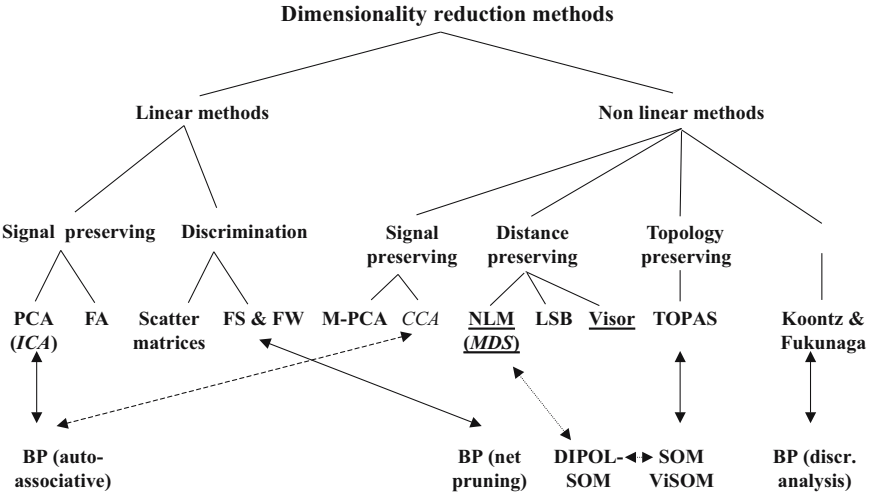

Fig. 2.11. Taxonomy of dimensionality reduction methods.

d_{Xij}, and, thus, implicitly the data structure, shall be preserved in the NLM according to the cost function $E(m)$:

$$E(m) = \frac{1}{c} \sum_{j=1}^{N} \sum_{i=1}^{j} \frac{(d_{Xij} - d_{Yij}(m))^2}{d_{Xij}}. \tag{2.7}$$

Here

$$d_{Yij}(m) = \sqrt{\sum_{q=1}^{d} (y_{iq}(m) - y_{jq}(m))^2} \tag{2.8}$$

denotes the distance of the respective data points in the visualization plane and

$$d_{Xij} = \sqrt{\sum_{q=1}^{M} (x_{iq} - x_{jq})^2} \tag{2.9}$$

in the original data space and

$$c = \sum_{j=1}^{N} \sum_{i=1}^{j} d_{Xij}. \tag{2.10}$$

Based on a gradient descent approach, the new coordinates of the N pivot vectors in the visualization plane $\mathbf{y}_i$ are determined by:

$$y_{iq}(m+1) = y_{iq}(m) - \mathrm{MF} * \Delta y_{iq}(m) \tag{2.11}$$

with

$$\Delta y_{iq}(m) = \frac{\partial E(m)}{\partial y_{iq}(m)} \Bigg/ \left| \frac{\partial^2 E(m)}{\partial y_{iq}(m)^2} \right| \quad and\ 0 < MF \leq 1. \tag{2.12}$$

In particular for large databases, due to the underlying computational complexity of the standard methods, e.g., the NLM with $O(N^2)$, mapping computation becomes infeasible. Therefore, particular interest was placed on heuristic and hierarchical methods of dimensionality reduction as mapping accelerators.

One of the first heuristic accelerating methods of the NLM was published by Lee, Slaggle, and Blum [2.34]. Rightly assuming that the gradient procedure does not always achieve an accurate projection (cf., e.g., [2.9]), they developed a fast distance-preserving mapping that focuses on the exact preservation of only a limited number of $2N - 3$ distances, neglecting all remaining ones. For this mapping, the minimum spanning tree (MST) of the data distance graph is computed. Points are mapped by common triangulation while traversing the MST, based on the previously mapped MST neighbors serving as pivot point. However, in spite of the appealing heuristic idea, MST computation and traversal itself still has $O(N^2)$ complexity. Thus, in own prior work, an even faster mapping algorithm was developed [2.26]. This alternative mapping, denoted as Visor mapping, also uses a triangulation mapping step, but with three fixed global pivot points that are heuristically chosen from the data set. The purpose of the pivot point determination is to find the three most extruded data points that meet the additional constraint of maximum mutual distance while enclosing the remaining data set. Based on centroid computation, these three data points are successively selected as pivot points from the data set. These points are placed first and the remaining N-3 data points are placed in the visualization plane employing triangulation.

This algorithm, denoted by Visor [2.26], has O(N) complexity and thus provides data projections with a very short response time and negligible sensitivity to the database size. As shown by prior investigations with a mapping quality measure, achievable mapping quality is similar to the NLM [2.26], [2.24]. Due to their salient properties with regard to speed, convenience, and transparence, distance-preserving mappings have been applied throughout this work to the regarded semiconductor manufacturing data. In addition, efficient hierarchical methods, offering a more delicate speed-accuracy tradeoff are available [2.23] and will be employed in the next stages of the work.

The unsupervised mapping methods discussed so far retain all features from the high-dimensional feature space and compute a more compact optimized feature space, e.g., for visualization and analysis purposes.

In contrast to this, feature selection actually helps to discard incoming variables that have no or little significance for the tackled problem. It must be remembered, that two very different aims can be pursued by the method of feature selection. For classification tasks, the selection of an as-small-as-possible group is desired, to allow generalization with a minimum classifica-

tion error. For data analysis, the discovery of all involved variables and the underlying knowledge are aspired. Feature selection can be understood as the computation of a constrained matrix A^S for a linear mapping with the following form

$$A^S = \begin{pmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & c_M \end{pmatrix}, \tag{2.13}$$

where only diagonal elements can have nonzero values and the $c_i \in \{0, 1\}$ are binary variables or switch variables determined by a preceding optimization process. Thus, a linear mapping $\mathbf{y} = A\mathbf{x}$ is constituted. However, due to the constrained matrix A and the fact that column vectors with $c_i = 0$ can be entirely omitted, computation can be simplified to $\mathbf{y} = [y_1, y_2, \ldots, y_m]^T$ with $y_i = x_j \quad \forall c_j \neq 0$, i.e., m corresponds to the number of $c_j \neq 0$ and the corresponding features x_j are just copied to the y_i. Feature selection performs a scaling of feature or coordinate axes by binary variables, i.e., switching off dimensions and thus defining a subspace that is salient with regard to the chosen criterion J. As no rotation of the basis vectors is carried out, explicit interpretability of the result is sustained. However, due to the binary nature of the selection process, the difference in importance or the impact of individual features is occluded. A straightforward extension of the binary matrix A^S given for feature selection is feasible, which allows continuous valued ranking of the features. The binary c_i are replaced by real variables $a_i \in [0, 1]$, which are determined by a preceding optimization process. The limitation or normalization to $[0, 1]$ is introduced for the sake of interpretability and comparison with corresponding feature selection results. This approach commonly denoted by feature weighting (FW) allows a continuous scaling of features or coordinate axes for $a_i \neq 0$. Those columns with $a_i = 0$ can be omitted, reducing the matrix from $M \times M$ to $M \times m$ with $m \leq M$. Thus, in addition to the aspired potentially higher achievable discrimination and better generalization properties, explicit salient information for data analysis purposes and rule weighting is extracted by this method. One particular method of finding appropriate a_i based on a certain cost function J and a gradient descent technique can be found in [2.21]. Numerous other options with regard to the chosen J and the optimization strategy, e.g., evolutionary computation, are feasible [2.45] and are currently being pursued in ongoing work. Various strategies and methods for feature selection will be discussed after presentation of relevant cost functions J.

Cost Functions. In the following, from a larger collection of potential cost or assessment functions summarized in Fig. 2.12, dedicated cost functions for feature space assessment introduced in prior work, e.g., [2.27] and [2.28] [2.22], will be briefly presented for the aim of a self-contained presentation. These serve for discrimination measuring in terms of class regions separability,

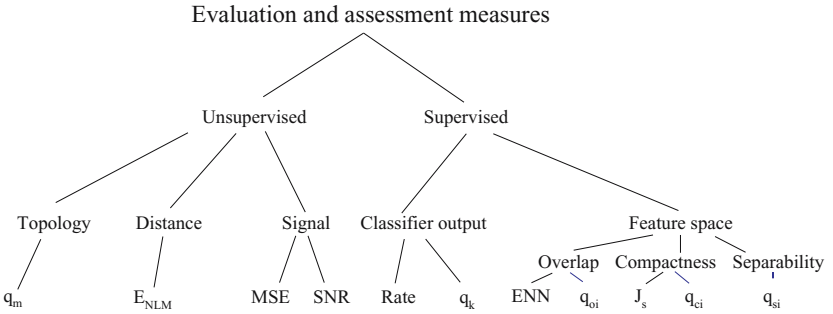

Fig. 2.12. Taxonomy of cost functions.

overlap, or compactness in the regarded feature space and ensuing systematic dimensionality reduction. Though the classification rate or a posteriori probabilities of any classifier could serve here (cf, e.g., [2.16] or [2.45]), for obvious practical reasons, robust measures nearly free of required parameters, model assumptions, and intricate training requirements are preferred in this work.

For instance, to measure separability, a nonparametric measure q_s exploiting nearest-neighbor techniques can be computed. For this class separability assessment, the RNN-classifier [2.13] is exploited, which iteratively selects a subset of relevant vectors as reference vectors from the training set, as the number of these selected reference vectors T_{RNN} is proportional to the feature space separability. This is illustrated in Fig. 2.13, where selected reference vectors T_{RNN} are emphasized in bold. In the case of linear separability of class regions, one vector per class region would be required. So the quality measure given by

$$q_s = \frac{N - (T_{RNN} - L)}{N} \tag{2.14}$$

has 1.0 as its optimum value indicating linear separability. An improved variant of q_s takes significantly different a priori probabilities in account:

$$q_{si} = \frac{1}{L} \sum_{i=1}^{L} \frac{N_i - (T_{RNN_i} - 1)}{N_i}. \tag{2.15}$$

Here N_i denotes the number of patterns affiliated to class ω_i and T_{RNN_i} the number of reference vectors selected for class ω_i. (It is assumed here that N_i corresponds to the actual a priori probability of class ω_i). The quality measures q_s and q_{si} have O(N) complexity and thus are very fast; however, the resolution is quite coarse, which can be detrimental for optimization schemes. Numerous feature space configurations can be mapped on the same assessment value.

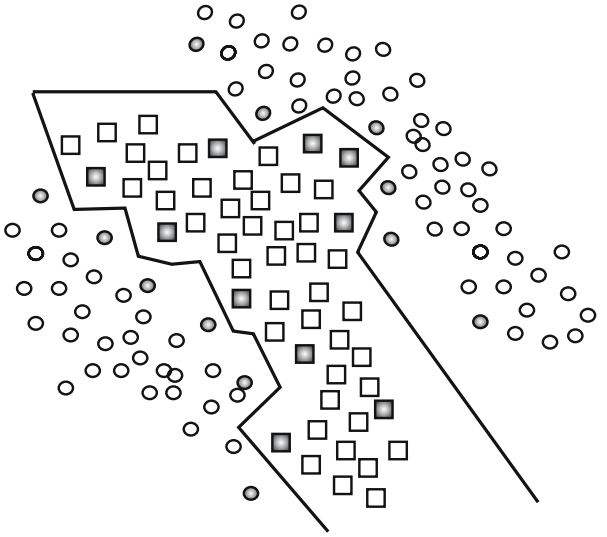

Fig. 2.13. Class separability assessment.

A very simple parametric measure for overlap computation was introduced in [2.49]. The class specific distributions are modeled by Gaussian functions and an overlap of two-class regions, denoted by ω_i und ω_j, can be computed from the respective mean values μ_i, μ_j and standard deviations σ_i, σ_j by

$$q_{x_{l_{ij}}} = \frac{|\mu_i - \mu_j|}{(N_i - 1)\sigma_i + (N_j - 1)\sigma_j}. \tag{2.16}$$

The merit of a feature for the separation of one class from all others is given by

$$q_{x_{l_i}} = \frac{1}{L-1} \sum_{j \neq i}^{L} q_{x_{l_{ij}}}. \tag{2.17}$$

Also, the merit of a single feature to distinguish all classes could be computed by

$$q_{x_l} = \frac{1}{L} \sum_{i=1}^{L} q_{x_{l_i}}. \tag{2.18}$$

However, practical experience has shown that the global summation can be misleading in some cases. A feature can, for instance, be excellent for certain class separations and meaningless for most others but have a summation value that outperforms other features that are good everywhere in feature space.

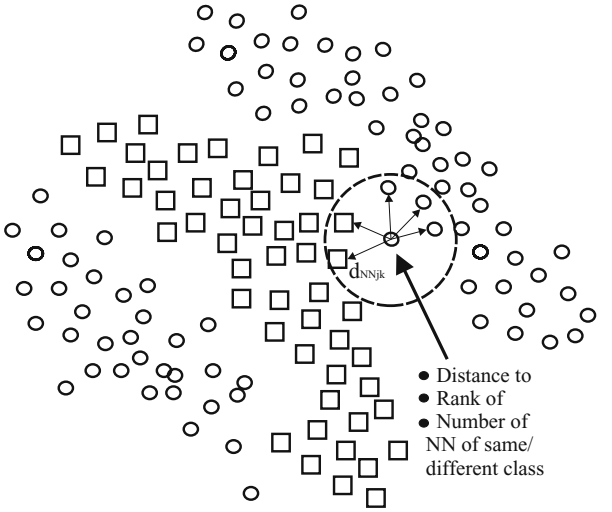

Fig. 2.14. Class overlap assessment.

Proposals for efficient application of these simple measures will be given in the following sections on feature selection strategies.

A nonparametric overlap measure q_o, which was inspired by the edited-nearest-neighbor (ENN) algorithm [2.8], in contrast to q_s, provides a very fine-grained value range and thus is better suited for optimization schemes. However, the price tag is an increased complexity of $\mathrm{O}(N^2)$ with regard to q_s. The basic idea of q_o is illustrated in Fig. 2.14. The overlap measure q_o is computed by:

$$q_o = \frac{1}{N}\sum_{j=1}^{N}\frac{\sum_{i=1}^{k} q_{NN_{ji}} + \sum_{i=1}^{k} n_i}{2\sum_{i=1}^{k} n_i} \tag{2.19}$$

with

$$n_i = 1 - \frac{d_{NN_{ji}}}{d_{NN_{jk}}} \tag{2.20}$$

and

$$q_{NN_{ji}} = \begin{cases} n_i & : \quad \omega_j = \omega_i \\ -n_i & : \quad \omega_j \neq \omega_i. \end{cases} \tag{2.21}$$

Here, n_i denotes the weighting factor for the position of the ith nearest neighbor NN_{ji}, $d_{NN_{ji}}$ denotes the distance between $\mathbf{x}_j$ and NN_{ji}, $d_{NN_{jk}}$ denotes

the distance between $\mathbf{x}_j$ and most distant nearest neighbor NN_{jk}, $q_{NN_{ji}}$ denotes the measure contribution of $\mathbf{x}_j$ with regard to NN_{ji}, and ω_j and ω_i denote the class affiliation of $\mathbf{x}_j$ and NN_{ji}, respectively. The influence of a nearest neighbor in the quality measure decays with its rank position to $n_i = 0$ for NN_{jk}. The final measure q_o is fine-grained and sensitive to small changes in the feature space. Further q_o is also normalized in [0,1], where 1.0 indicates no overlap in the feature space. Typically, 5 to 10 nearest neighbors are well suited for computation of this quality measure. Simplification of the measure is feasible, trading off fine-grained resolution in overlap computation and, thus, sensitivity to small changes in the feature space against computational savings. An improved variant of q_o takes significantly different a priori probabilities into account

$$q_{oi} = \frac{1}{L}\sum_{c=1}^{L}\frac{1}{N_c}\sum_{j=1}^{N_c}\frac{\sum_{i=1}^{k} q_{NN_{ji}} + \sum_{i=1}^{k} n_i}{2\sum_{i=1}^{k} n_i}. \tag{2.22}$$

Finally, compactness q_c can be measured by explicitly computing the ratio of current intra- and interclass distances. Implicitly this criterion is also used in the computation of scatter matrices [2.12]. The compactness q_c previously introduced in [2.22] suffers from the flaw that the measure will be optimum, if the majority of intraclass distances will be made small, i.e., class regions with the majority of patterns will dominate the assessment and consequently any optimization process based on the measure q_c. An improved measure q_{ci} for different a priori probabilities and corresponding N_l in the L-class problem can be obtained by class-specific normalization during compactness computation

$$q_{ci} = \frac{\frac{1}{L}\sum_{l=1}^{L}\frac{2}{N_l(N_l-1)}\sum_{i=1}^{N}\sum_{j=i+1}^{N}\delta(\omega_i,\omega_j)\delta(\omega_i,l)d_{X_{ij}}}{\frac{1}{N^B}\sum_{i=1}^{N}\sum_{j=i+1}^{N}(1-\delta(\omega_i,\omega_j))d_{X_{ij}}} \tag{2.23}$$

with

$$d_{X_{ij}} = \sqrt{\sum_{q=1}^{M}(x_{iq}-x_{jq})^2} \tag{2.24}$$

and $\delta(\omega_i,\omega_j)$ is the Kronecker delta, which is $\delta(\omega_i,\omega_j) = 1$ for $\omega_i = \omega_j$, i.e., both patterns have the same class affiliation, and $\delta(\omega_i,\omega_j) = 0$ elsewhere. Also, $\delta(\omega_i, l)$ prescribes that only distances with $\omega_i = \omega_j = l$ are accumulated for the lth-class sum of intraclass distances. Further, the normalization factor N^B is given by

$$N^B = \sum_{i=1}^{N}\sum_{j=i+1}^{N}(1-\delta(\omega_i,\omega_j)). \tag{2.25}$$

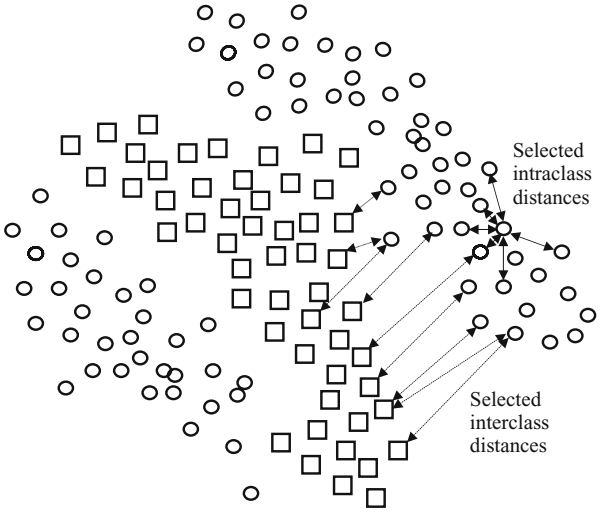

Fig. 2.15. Class compactness assessment.

The principal idea of intraclass and interclass distance computation for q_{ci} is illustrated in Fig. 2.15. The improved compactness q_{ci} has a complexity of $O(N^2)$, is a nonparametric measure, and requires no parameters to be set by the user. It shows a high sensitivity to changes in feature space, as these are immediately mirrored by changes in distance, and, thus, in changes in q_{ci}. In comparison to the existing overlap measure q_{oi} with equal sensitivity and computational complexity, q_{oi} is inferior, as it requires the parameter k to be set. But q_{oi} is superior with regard to normalization properties, returning a value in [0,1], whereas q_{ci} values depend on the distances in the data set and only allow the observation of relative changes. For FS, an additional normalization step for each selection or configuration is required for q_{ci}.

These measures will serve in the following as feature space assessment or ranking measures J. Information on the individual features' merit as well as the current feature combinations' merit can be obtained by employing the presented measures. Also, the results of different dimensionality reduction methods, e.g., FS or FW, can be quantitatively compared and assessed [2.24].

Feature Selection Methods. The process of finding the appropriate coefficients c_i in (Eq. 2.13) is an intricate optimization problem. Due to the combinatorial complexity inherent to the problem of FS, the computational effort of finding the best selection, i.e., feature combination, grows exponentially. Thus, the global optimum solution for the selection process cannot be found with polynomial complexity or effort, i.e., we have an NP-complete problem (cf., e.g., [2.1]). Therefore, a complete or exhaustive search of all

feature combinations in general is out of the question. Several alternative search strategies for FS, employing the cost functions from Section 2.3.2, will be summarized with regard to achievable performance and required computational effort.

First-Order Selection Techniques. One simple but often effective way of finding a suboptimum solution with minimum effort is to compute an individual figure of merit for each feature. This first-order approach neglects possible higher-order correlations between feature pairs or feature tuples. For assessment or figure of merit computation, for instance, one of the cost function given in the previous subsection has to be applied. However, the cost function in this simplified case will be computed separately for each feature. Three permutations are basically feasible:

- The figure of merit is computed for a selected feature and a selected combination of classes, i.e., the feature contribution to pairwise class discrimination is assessed. For instance, the measure $q_{x_{l_{ij}}}$ could be computed here. For each class pair, features are ranked according to their individual merit. Selection from these rank tables can be achieved, for instance, by choosing all features in first-rank position. Table 2.1 gives an example of this first-order selection scheme for the well-known Iris data. Obviously, for first-rank position **R**, features 3 and 4 will be selected. The method can be computed very quickly, but the rank table grows for given feature number M and class number L by $M * (L(L-1)/2)$.
- The figure of merit is computed for a selected feature and for the discrimination of one class versus all others. The corresponding rank table grows for given feature number M and class number L by $M * L$.
- Computing the figure of merit with regard to discriminating all classes for each feature returns a single column with M elements.

As shown in Table 2.1, the parametric overlap measure $q_{x_{l_{ij}}}$ and its variants can serve for the three approaches of fast first-order feature selection. If the parametric assumption is met, then this simple scheme can be very effective. However, in many practical cases, even for the one-dimensional distributions of the individual features, a nonparametric nature can be observed. An effective remedy for this situation is the application of, e.g., the overlap

Table 2.1. Rank table from first-order assessment for Iris data.

Feature	R	C 1-2	R	C 1-3	R	C 2-3
x_1	4	1,020	3	1,482	3	0,442
x_2	3	1,065	4	0,890	4	0,255
x_3	2	4,139	1	**5,451**	2	1,218
x_4	1	**4,387**	2	5,180	1	**1,660**

Table 2.2. Rank order for first-order feature selection computed for visual inspection feature data based on parametric (left column) and nonparametric (right column) assessment measure.

Feature	R	C 1-2	R	C 1-2
x_1	5	0,2917	5	0,6977
x_2	4	0,6489	3	0,8211
x_3	1	1,1558	4	0,7058
x_4	3	0,8547	2	0,8808
x_5	2	1,0047	1	0,9270

measure q_o (or q_{oi}) separately for each individual feature. This returns a corresponding nonparametric measure to the parametric one given earlier. For Iris data, the selection will be identical. In Table 2.2, however, a feature set computed from images of a practical visual inspection problem is subject to both the parametric and the nonparametric first-order feature selection scheme.

For the regarded nonparametric example data set only the nonparametric measure provides the a priori known correct solution. Summarizing, first-order selection schemes are a special case of heuristic approaches to find solutions to the otherwise NP-complete feature selection problems. Suboptimum solutions can be found at very low computational costs. Employment of the nonparametric measure provides more robustness due to the relaxed distribution assumption at moderate cost increase, which is dependent on the sample set size with $O(N^2)$. Further, the simple first-order selection could be employed to weed out variables, which already possess distinct meaning for themselves, and apply more complex search strategies on the residual variables.

Higher-Order Selection Techniques. Higher-order correlations or dependencies of features require the computation of the feature merit with regard to a tuple of other features. In the limit, the effect of a certain feature with regard to all other features has to be considered. As mentioned before, this is a problem of combinatorial optimization and the best possible solution, i.e., the global optimum can be found by exhaustive search. Due to the exponential increase of possible combinations, which grow by 2^M for the binary selection problem and the number M of features, and the underlying NP-completeness of the problem only for small to moderate M is an exhaustive search feasible.

Let us assume that computation of the assessment measure q_o, which depends on the sample set size N with $O(N^2)$, takes one second on a standard computer. Then an exhaustive search for $M = 12$ will consume $2^12 = 4096$ seconds, which amounts approximately to 1 hour and 8 minutes of computation time. For $M = 16$, more than 18 hours of computation time will be required. It is obvious that for larger databases, either for classification or for

data analysis, the employment of exhaustive search, and thus the guaranteed finding of the global optimum, will be infeasible.

In addition to first-order selection schemes, for more features, heuristic search strategies employing tree search schemes, e.g., Sequential Forward/Backward Selection (SFS/SBS) were devised [2.16]. These are also implemented in the method collection and corresponding toolbox within the QuickCog system [2.28]. These heuristic approaches systematically reduce the number of searched and assessed feature combinations. As many combinations are left out of consideration, the global optimum can be missed, and convergence to just a local optimum solution for the selection problem is guaranteed. In SFS, for instance, initially no features are selected. Now each of the N features is tentatively selected and its effect on the figure of merit, e.g., class regions overlap, is computed. The feature with the best assessment is permanently selected and frozen. The same procedure is iteratively repeated for the remaining $(N-1)$ features until only one feature can be altered. Now either the feature combination with the best assessment value can be selected, regardless of the number of selected features, or for a fixed maximum number of features the row with the best compromise of assessment value and required minimum number of features will be selected. In SBS, the same process starts from the initial condition that all features are selected and get rejected in the process. Figure 2.16 elucidates the SBS process and Table 2.3 shows an example of a selection process protocol for Iris train data using SBS [2.16] and the q_s quality measure [2.28]. As $M*(M-1)/2+M$ combinations have to be assessed in both cases, the computational complexity is given by $O(M^2)$. Thus, for $M = 16$, in this case, a local optimum solution will be found within approximately 4 seconds compared to more than 18 hours for an exhaustive search. Though the finding of a global optimum is not guaranteed, these robust methods provide good solutions quickly, and in practical work the global optimum was often found.[9] Comparing these heuristic methods with the simple first-order selection schemes, it can be stated with some

[9] These were cases where an exhaustive search for result comparison was still feasible.

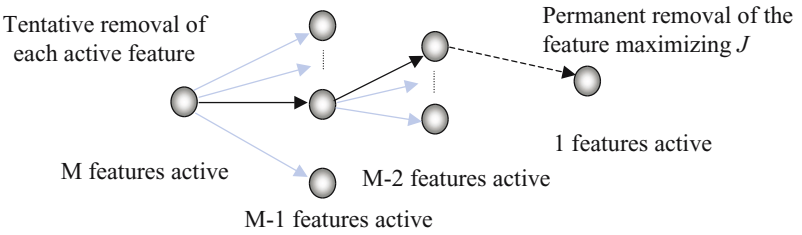

Fig. 2.16. Illustration of SBS feature selection.

Table 2.3. Feature selection protocol for Iris data.

```
Selection strategy:  SBS
Assessment measure:  Separability qs
1 2 3 4   0.90667
- 2 3 4   0.94667
- - 3 4   0.96000
- - - 4   0.00000
Optimum quality     :   0.96
Significant Features:   3 4
```

caution, that the higher-order methods are usually superior. But, of course, it is possible that the simple first-order scheme runs on a configuration that is neglected by the higher-order methods due to the search strategy and returns a better solution. Instead of strict top-down or bottom-up processing, as met in SBS or SFS, an alternation between feature rejection and selection during the search process can be found in other approaches, e.g., branch-and-bound approaches or floating search.

Further heuristic search strategies, employing stochastic methods, e.g., simulated annealing (SA) [2.1] or Boltzmann machines (BM) [2.1], as well as bio-inspired techniques for optimization, e.g., genetic algorithms (GA) in particular and evolutionary strategies (ES) in general, can be applied for FS [2.45], [2.11]. Also, multiobjective optimization can be merged with the GA/ES approach [2.11]. This subject is pursued in ongoing work.

The permanent elimination of redundant and irrelevant features from the sample set by FS provides an effective means of dimensionality reduction. However, the crispness of the selection process can lead to stronger sensitivity with regard to variances in the feature representation in generalization due to the loss of information contained in the discarded features. The issue of the stability of the FS solution and the underlying maximum of the cost function is raised here. It is especially painful for data analysis and knowledge acquisition, if for minor changes in the data entirely different features are selected. The methods discussed so far are specialized to classification problems and require revision and enhancement with regard to stability and data analysis.

Visualization Techniques and Dedicated Tools. In contrast to the state of the art, e.g., static scatter plots, in the methodology pursued in this research work, the achieved projections are the baseline for interactive human analysis. Interactive CAD-like visualization techniques, e.g., interactive navigation, diverse component plots, grid plots, and attribute plots, support human perception and analysis [2.24]. Figure. 2.17 gives a taxonomy of relevant visualization techniques for large high-dimensional data. For instance, at each projection point, the value of a selected variable can be plotted in a Hinton diagram style, i.e., the variable value is coded by the side length

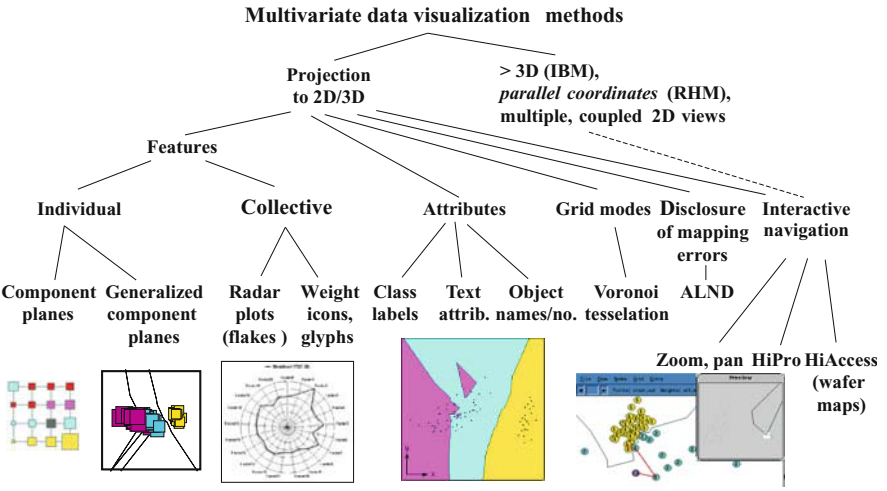

Fig. 2.17. Taxonomy of visualization techniques for high-dimensional data.

or the area of a rectangle. Alternatively, several variables can be plotted by iconified radar plots at each projection point (see Fig. 2.10).

Figure 2.18 (a) shows the underlying multivariate data visualization architecture. Especially the features for accessing database contents from the top-level map should be pointed out here as unique characteristics of the approach. Two implementations have been conceived so far, the general-purpose tool WeightWatcher (WW) in QuickCog (Fig. 2.18 (b)) and the dedicated Acoustic Navigator [2.25] with enhanced interactive features (Fig. 2.18 (c)). Further interactive enhancements are on the way, e.g., interactive selection, labeling, and extraction of arbitrary data from the map. The outlined methods and tools have been compared, assessed [2.24], and employed in numerous scientific and industrial applications. Examples of applicability are given in

- rapid prototyping in the design of recognition systems [2.10];
- analysis of medical databases [2.18];
- analysis of psychoacoustic sound databases with the extension to synthesis in sound engineering [2.25]; and
- analysis and design of integrated circuits with regard to design centering and yield optimization.

For the case of rapid and transparent recognition system design a brief example will be given. A vision system was designed for a medical robot in an object recognition task [2.10]. Dimensionality reduction and interactive visualization approach helped to assess the current system's capability in terms of feature space discrimination and occurrence of pop-outs or outliers. This is illustrated in Fig. 2.19. Additionally, the backtracking capability from the resulting interactive map is illustrated by invoking the original image of a

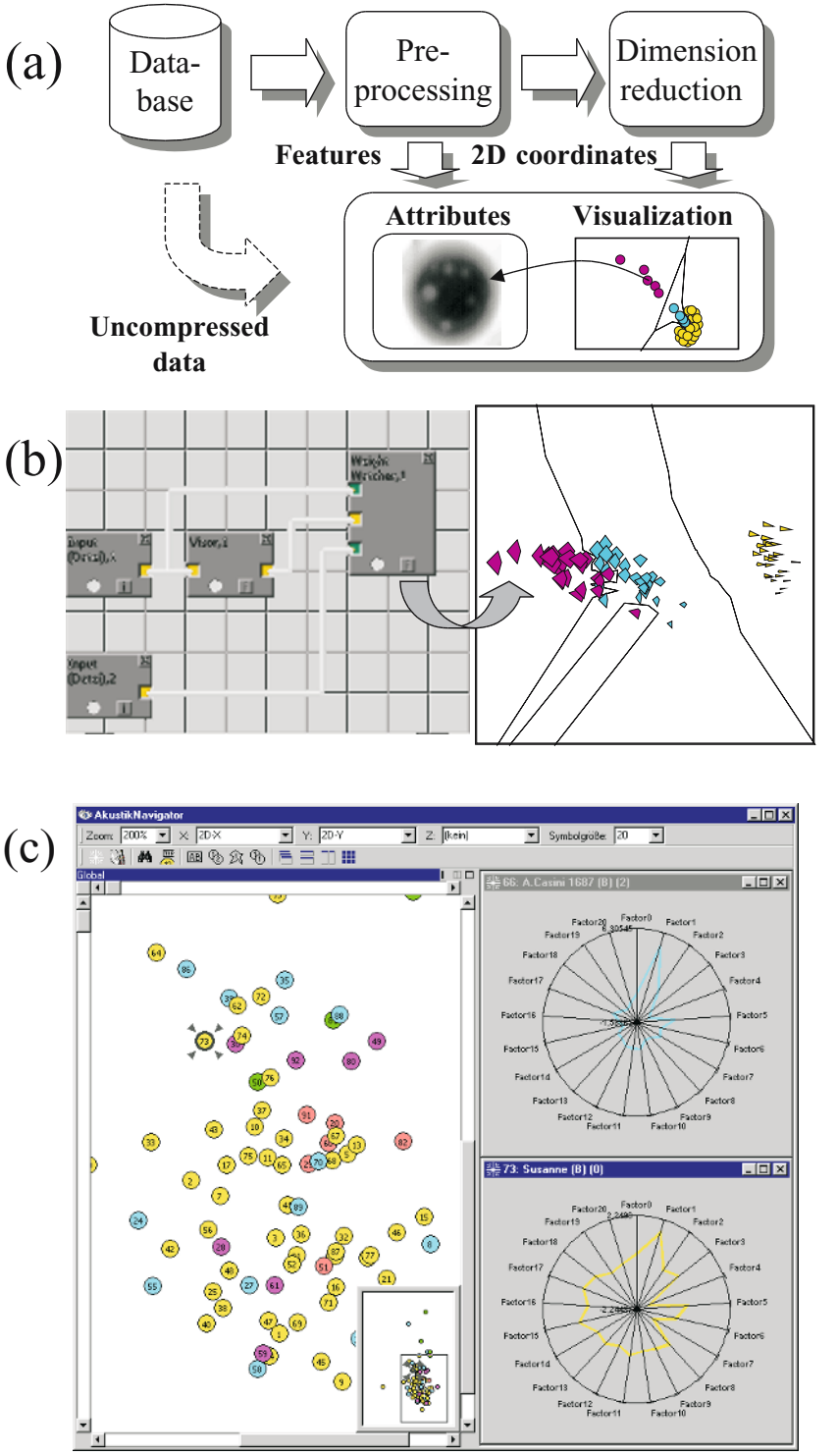

Fig. 2.18. Feature space reduction and interactive visualization: (a) Architecture and dedicated tools; (b)WeightWatcher; and (c) acoustic navigator.

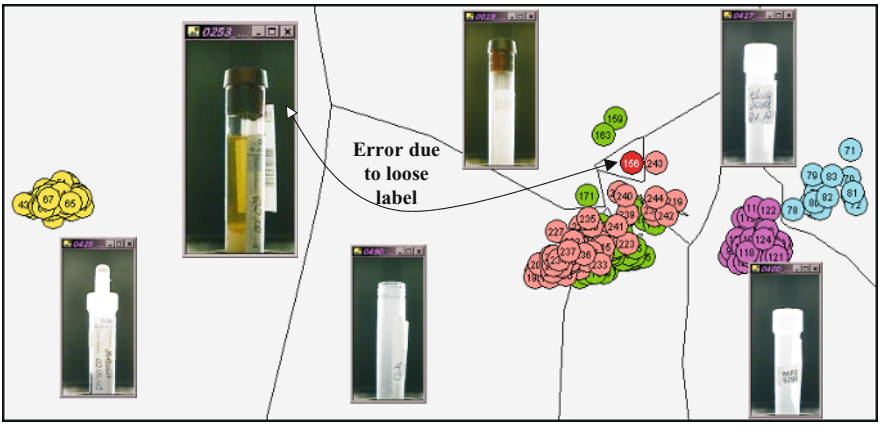

Fig. 2.19. Feature space for vision system of medical laboratory robot.

selected object for each class from the underlying database. Thus, occurring problems, e.g., misclassifications, and underlying causes, can be easily made overt. This alleviates troubleshooting in system design and increases design speed, reliability, and overall productivity. The work is extended to microelectronic manufacturing process data analysis and the features elaborated in prior research and application projects are adapted to this domain. For instance, data entries can be tracked back from the projection in the process database as illustrated in Fig. 2.19 for image data. Thus, the database can be browsed and analyzed according to the inherent clustering and structure in the data. The extension of the existing approach to semiconductor manufacturing will be presented in the following section and in Section 2.5, giving an outline of the envisioned domain-specific system.

2.4 Experiments and Results

The first step of the work in this feasibility study targets the validation and demonstration of the actual practical assistance of the dimensionality reduction and visualization approach to discover structure in and extract knowledge from the industrial high-dimensional database. Thus, it is expected from the visualization that the known split information can be effortlessly retrieved from the map. In this case, unknown clustering in the data, due to detrimental and unintended effects, could also be made overt to the process analyst at a glance.

The most simple and fast Visor projection method was applied to the data first [2.24]. Figure 2.20 shows that distinct yet overlapping clusters can be identified in the data. It is well known from physical and technological background knowledge, that the generated split affects only a fraction of

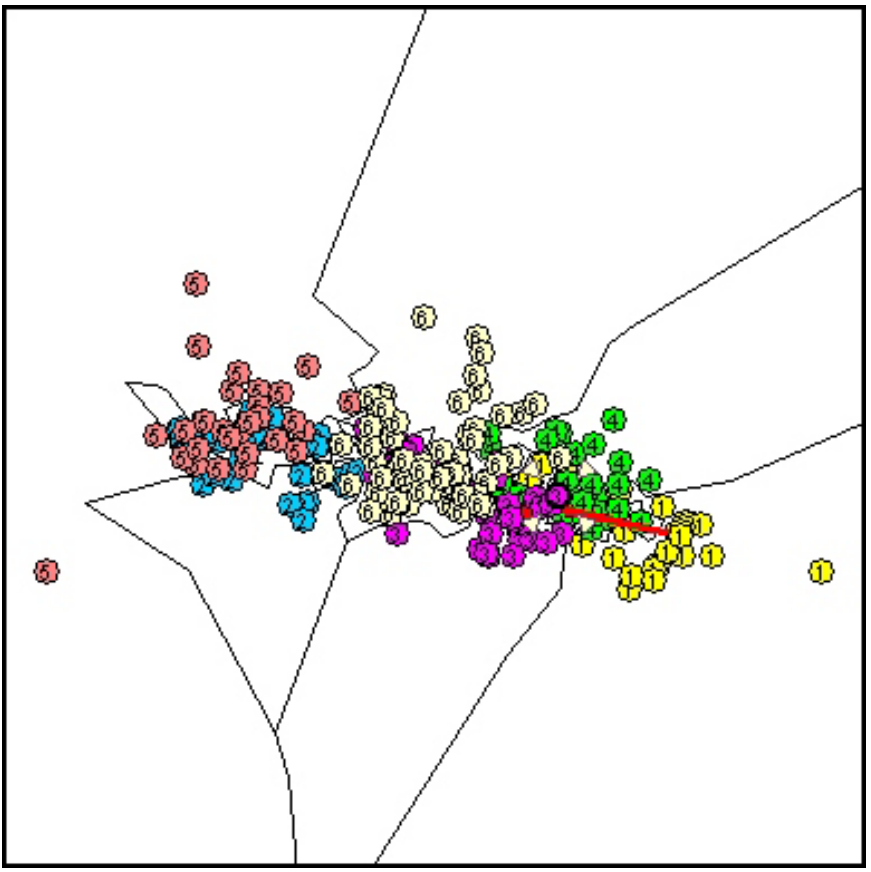

Fig. 2.20. Visualization of SPLIT6.

more than 200 parameters included in the database. Therefore, the observed cluster overlap in this unsupervised mapping approach is related to the quasi-noise of the large number of variables unrelated to the split. However, as in previous application projects, the feasibility of the dimensionality reduction and visualization approach could be shown for the regarded semiconductor manufacturing process.

Additionally, in Fig. 2.21 four selected variables are displayed by component plots. It can be perceived from this representation that the variables C118 and C119 are characteristic for the existing split, whereas C071 distinguishes the lots rather than the split, and finally C063, which is characteristic for neither the lots nor the split.

In addition to the overall visualization of the data, based on unsupervised dimensionality-reducing mapping and all variables, it is of importance to determine which parameters or groups of parameters are conforming with or

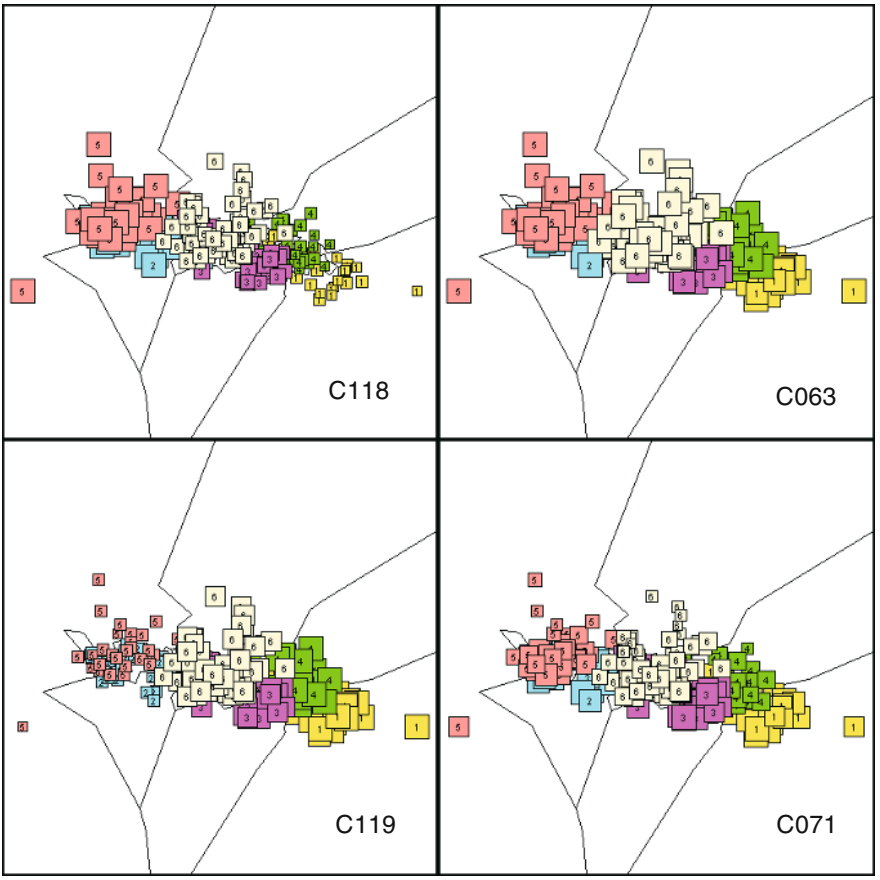

Fig. 2.21. Visualization of SPLIT6 by four selected component plots.

opposed to the existing split. Parameters also might be redundant with regard to this issue. From the available supervised methods, automatic selection of features has been employed to find an answer to this question for the regarded application data. The SBS selection method delivered the best results for the higher-order methods in the conducted experiments. In Table 2.4 the results for lot and split discrimination (SPLIT6) and only split discrimination (SPLIT3) are documented for the three regarded cost functions and the best obtained results.

For instance, application of SBS with q_{si} reduced the SPLIT6 database to just nine parameters. Figure 2.22 shows the resulting projection with nearly linear separability of the data. From the resulting projection in Fig. 2.22, as well as the later Fig. 2.23, the existing asymmetry of the split can clearly be observed, which is a very significant achievement of the regarded visualization method. The expectation, of course, is that the selected parameters are dom-

Table 2.4. FS results for SPLIT6 and SPLIT3.

Selection method	Cost function	Dim.	Chosen features
1rstOP	1. Rank	4	1, 32, 118, 141
SBS	$q_{si} = 0.99487$	9	32, 65, 79, 114, 119, 142, 191, 198, 199
SBS	$q_{oi} = 1.0$	8	32, 65, 86, 129, 131, 142, 191, 201
SBS	q_{ci}	15	78, 79, 114, 115, 118, 119, 120, 121, 126, 129, 140, 141, 142, 143, 144
1rstOP	1. Rank	2	118, 141
1rstOP	1.– 2. Rank	4	118, 120, 140, 141
1rstOP	1.– 3. Rank	6	118, 119, 120, 140, 141, 144
SBS	$q_{si} = 1.0$	1	126
SBS	$q_{oi} = 1.0$	2	118, 205
SBS	q_{ci}	15	78, 79, 114, 115, 118, 119, 120, 121, 126, 129, 140, 141, 142, 143, 144

inantly responsible for the observed split. However, it must be minded that weaker correlations of potential interest for the data analyst are removed by this method, which is tailored to the needs of classification. Only those variables of value for optimum separability or optimum overlap will be chosen. The measure q_{oi} saturated early in the selection process, i.e., the maximum cost function value 1.0 was reached very early, which means the measure lost capability to properly distinguish between the contribution of the remaining variables. Correlating the achieved result with the underlying physical meaning of the variables showed that only a fraction of the relevant variables were identified (see Table 2.5). For comparison purposes, the described first-order method (1rstOP) also has been applied, employing the first highest-ranking variables for pairwise class separation. Some of the relevant variables were found with a significant speed difference compared to the higher-order methods, i.e., seconds vs. several hours on a state-of-the-art PC. However, the method identifies an irrelevant variable, too, and regretfully leaves out of consideration numerous relevant ones.

For SPLIT3, for q_{si} only one and for q_{oi} only two variables were selected. The methods both saturated early in the selection process. In both cases the class regions are not compact and show considerable scatter. Though a lean classification system could be devised from this result for the information gathering and knowledge discovery this results is far from desirable. The application of the 1rstOP delivered similar results for first-rank variables. Only a few of the relevant variables were identified. Increasing the included rank positions, more relevant variables were included (see Table 2.4). However, it is difficult for the user to judge, which parameter value for the rank position should be set to include all relevant variables and avoid irrelevant ones. Also, redundant variables could still be present in the selection. Due to its speed,

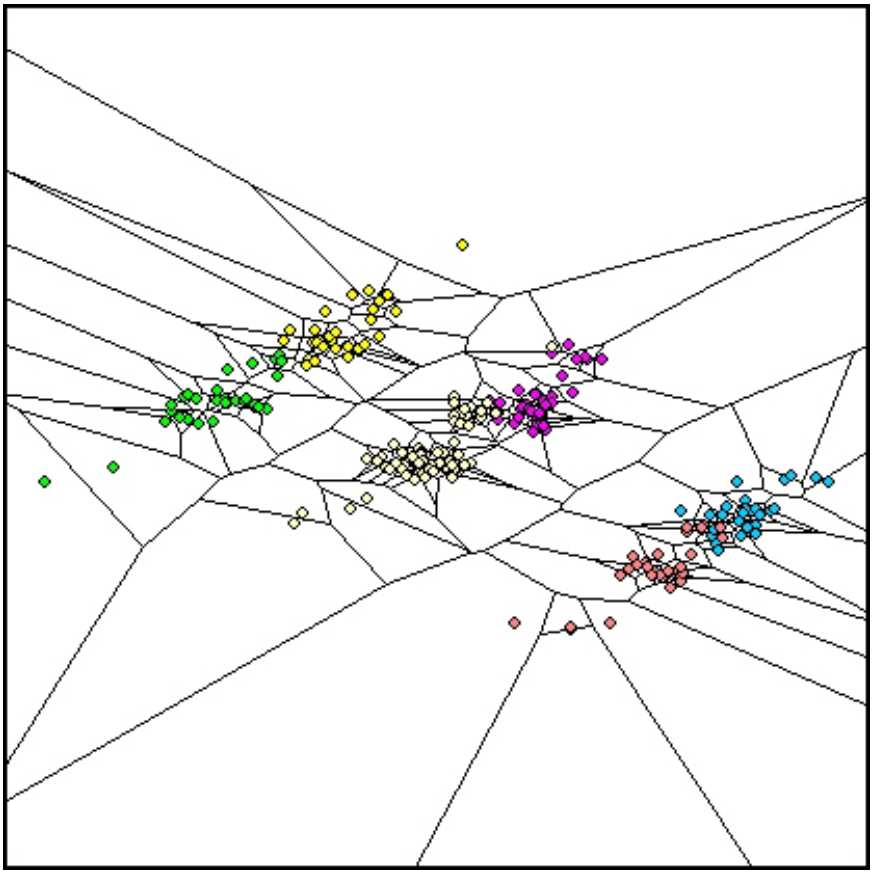

Fig. 2.22. Visualization of selected SPLIT6.

the method could be applied to create a starting solution for a higher-order method in a hierarchical approach. Such a hierarchical approach is considered very promising for future work.

The most meaningful result with regard to identified underlying physical and technological evidence was achieved by the most recent FS variant, employing SBS and q_{ci} for SPLIT6 as well as SPLIT3. Fifteen variables have been selected (see Table 2.4), and a feature space with compact and well-separated class regions is obtained by this selection. Figure 2.23 shows the resulting projection of the 15-dimensional data of SPLIT3, which is definitely superior to the result obtained for q_{si} application. Regarding the underlying physical meaning of the variables, the validity and significance of this selection is underpinned. Table 2.5 explains the meaning of the selected variables for the regarded submicron CMOS process.

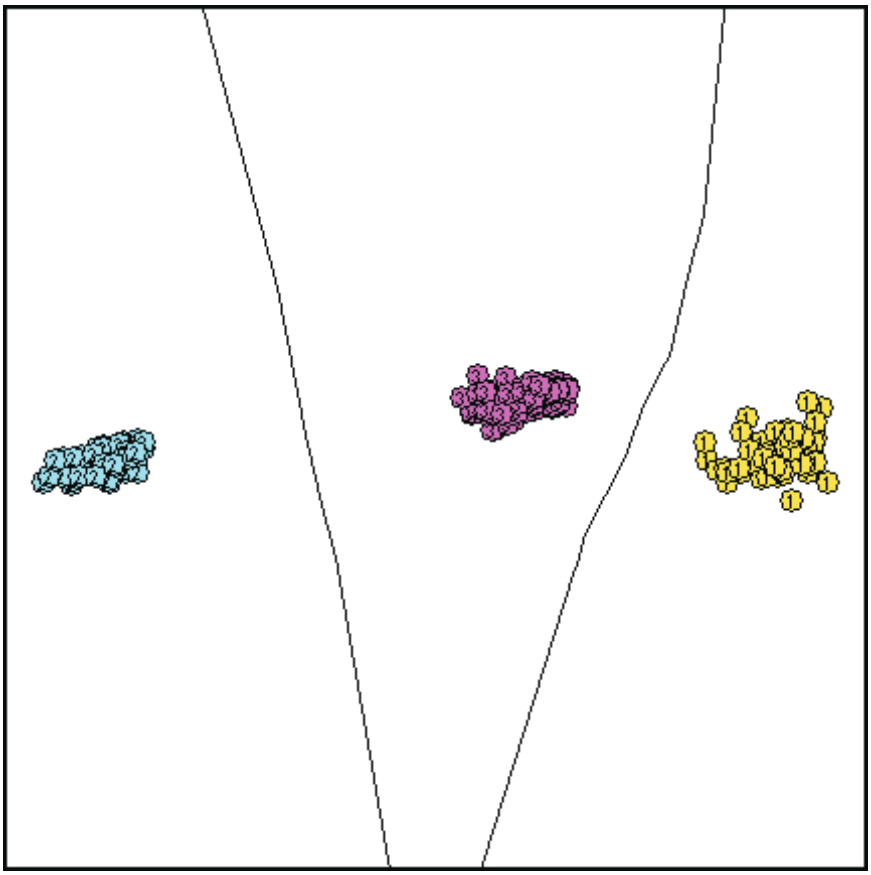

Fig. 2.23. Visualization of selected SPLIT3 according to compactness q_{ci}.

After the regarded steps of interactive visualization, analysis, and automatic determination of relevant variables, the monitoring of the process state by classification methods is investigated. According to the underlying lots, SPLIT6 was separated after feature selection (SBS, q_{si}, 9 features) into a training set, SPLITTrain3, and a test set, SPLITTest3. The six different classes in SPLIT6 were due to the distinguishing of the lots. Splitting SPLIT6 into a training and a test set reduces the classification task to an $L = 3$ class problem. In the first step of this part of the work, the training set was used to train a reduced nearest neighbor classifier (RNN) [2.13]. As can be seen from Fig. 2.24, generalization was perfect and data from the second lot can perfectly be classified according to the three split classes and the features chosen for optimum separability. However, in this approach numerous samples of the novel or abnormal cases were available. In the second step of this part of the work, OCC was applied to the same data. It must be kept in mind

Table 2.5. Physical and technological meaning of selected variables.

Parameter number	Explanation
IO device	
78, 79	Threshold voltages
Logic NMOS device	
114, 118, 120, 129	Threshold voltages
115, 119, 121	Saturation currents
131	Punchthrough current
Logic PMOS device	
140, 143	Threshold voltages
141, 144	Saturation currents
142	Channel leakage
Parameters unrelated to split	
1	Breakdown voltage
32	Saturation current MV device
65	Sheet resistance well
71, 73	Threshold voltage HV devices
191	Gate oxide thickness
198, 199, 201, 205	Sheet resistance poly
Derived parameter	
126	Universal curve FOM

that NOVCLASS only uses the samples affiliated to class 1 of the training set during learning. Thus, samples affiliated to classes 2 and 3 were not involved in the training of NOVCLASS and were unknown to the OCC classifier. In the following, classes 2 and 3 will be merged to class 2, denoting abnormal or novel measurements and respective process states. The aim was to assess the feasibility of NOVCLASS for (semi)automatic significance and novelty data

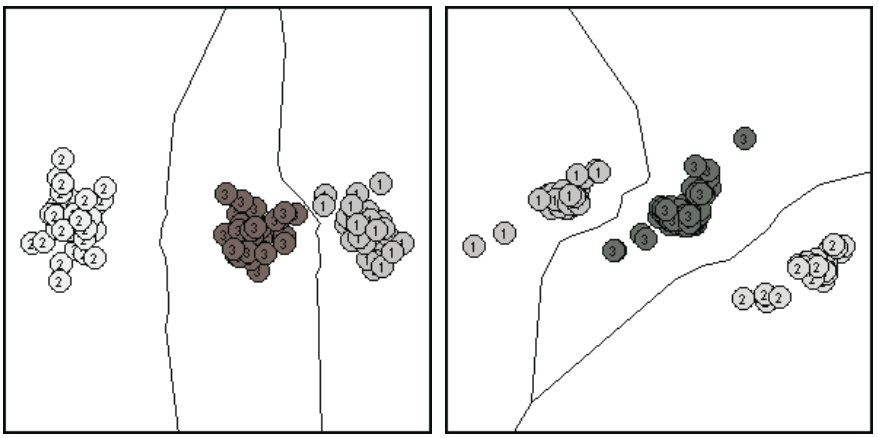

Fig. 2.24. Visualization of selected SPLIT6 classification.

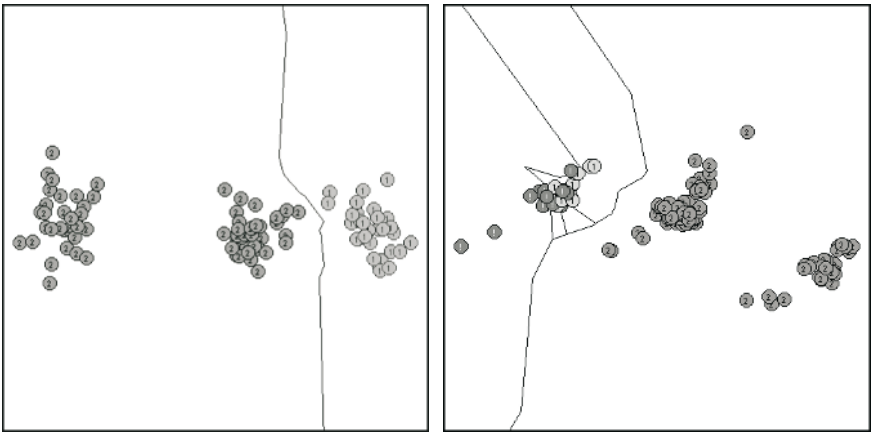

Fig. 2.25. Visualization of selected SPLIT6 novelty classification.

filtering within an information-processing hierarchy for process analysis, control, and optimization. The achieved results are illustrated in Fig. 2.25. The complete training set itself was correctly classified with regard to the bifurcation normal (class 1) or novel (class 2). For the test set, the vectors of classes 2 and 3 were also correctly identified as novel. However, numerous vectors of the normal test data were also classified as novel, as they occur a significant distance from the normal training data. Thus, a recognition rate of only 87.2% was achieved for the test set. It must be minded that the superior result of the RNN classifier required training by 95 vectors. The majority of these samples were counterexamples from the abnormal or novel range. In contrast, OCC was trained with only 30 vectors. The presented training data are rather sparse, so improvements of the OCC performance can be expected by providing larger data sets of normal process data as well as by a more sophisticated $R_{\max}$ computation and resulting normal range coverage in parameter space. However, though numerous practical improvements are possible, the feasibility of the described method to filter out significant novel data and perform as a data-reduction module also has been demonstrated.

The objectives of this feasibility study for the chosen problem and data have all been achieved. The feasibility of the selected soft-computing methods could be confirmed and relevant approaches for method improvement could be identified.

2.5 Proposed System Architecture

In the presented feasibility study, several selected methods were investigated with actual problem data with regard to their applicability for semiconductor manufacturing. As encouraging results have been obtained, a more so-

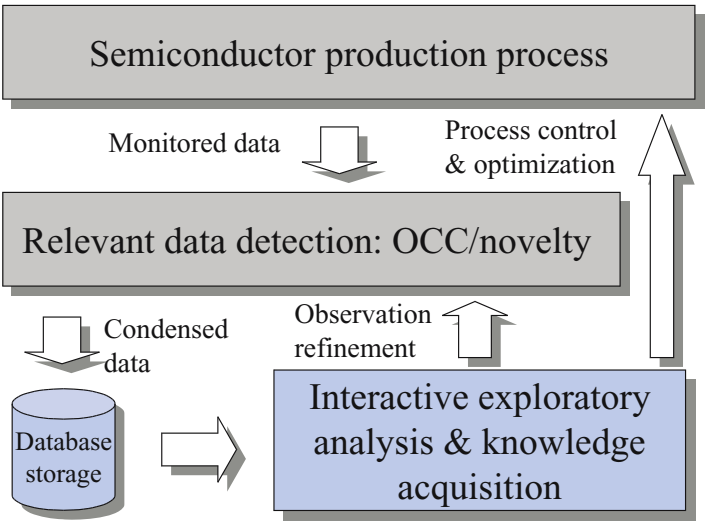

Fig. 2.26. Proposed system architecture for semiconductor manufacturing process analysis.

phisticated approach of employing and combining the regarded methods will be pursued next. A rough sketch of the envisioned information-processing architecture is given in Fig. 2.26. Similar to other applications, e.g., event classification in high-energy physics [2.39], a real-time classification stage is included in the proposed architecture. This module shall assess locally and in realtime whether interesting and relevant, i.e., novel, data occurred that should be stored for ensuing interactive analysis by human experts. OCC and the NOVCLASS model are first-choice candidates for this module. After storing in the database, dimensionality-reduction methods and interactive visualization will be undertaken for the analysis of the novel or abnormal data. Resulting understanding and knowledge extraction provide the baseline for potential actions as, e.g., classifier stage refinement or process control and optimization activities. Especially the interactive data visualization module can be significantly improved to the benefit of the regarded application. This has already been demonstrated for a different application domain in psychoacoustics, where an enhanced tool, denoted Acoustic Navigator (AN), was devised [2.25]. AN has been equipped with improved display features, such as multiple- and single-radar plots and practical search functions, which effortlessly direct the analyst to data entries of interest in the map visualization. These and numerous other convenience functions will allow transparent, fast, consistent, and thus, productive work on large, high-dimensional, and abstract databases. Figure 2.27 shows a first adaptation of the AN to the regarded application. Radar plots and the search function are illustrated. The focus of the follow-up research shall be put on this crucial system compo-

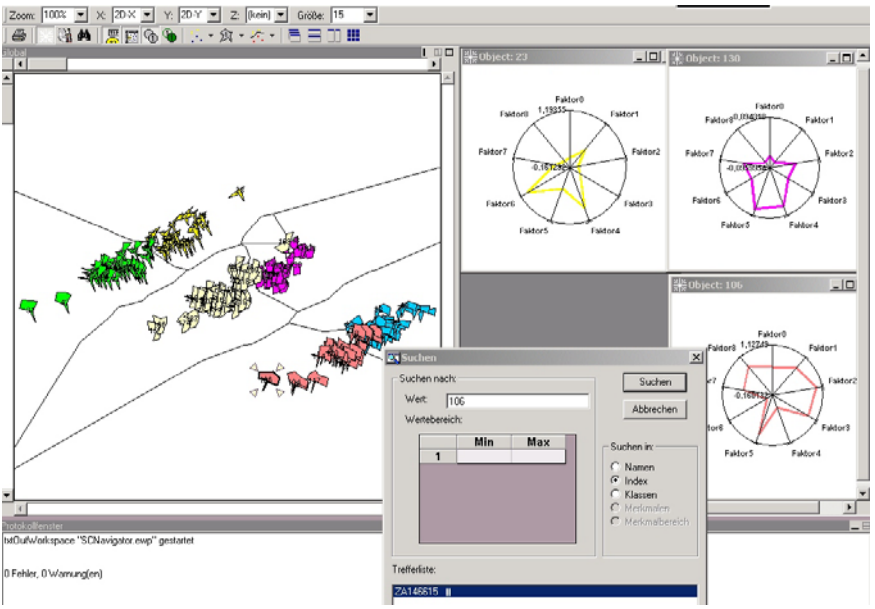

Fig. 2.27. Illustration of enhanced visualization features by the adapted AN.

nent and the related dimensionality-reduction methods, which also can be of assistance to cluster, select, and rank features or measurement parameters.

2.6 Conclusions

The presented work contributes to the industrial application of advanced soft-computing methods in the field of semiconductor manufacturing process data analysis. In particular, fast and efficient methods for multivariate data-dimensionality reduction, including automatic methods for parameter or parameter group saliency detection, and interactive visualization have been investigated in this first feasibility study.

Already the least complex and therefore most computationally inexpensive visualization methods allow significant insight into the structure of the data. Complemented by an interactive feature-selection tool, these visualization methods represent a powerful addition to the standard statistical analysis that is usually performed. Online visualization of the process trajectory in the multivariate space is also feasible by available fast methods for adding new data vectors in an existing mapping [2.23].

Furthermore, the investigation of automatic feature-selection methods has yielded very promising results. For instance, from the resulting projection, the asymmetry of the split can clearly be observed, which is a very significant achievement. Additionally, even in those cases where variables selected

were not pertinent to the split, the selection is soundly based. The bases are differences between the two lots, between single wafers in each lot, and even variations with regards to the position on the wafer. Again this is properly accounted for in the projections, further validating our approach.

In addition to these offline analysis and knowledge-extraction methods, dedicated classification techniques for online observation and potential control of the underlying process have been investigated. The feasibility of OCC and the proposed NOVCLASS method for selective data storage could be confirmed.

In this early stage of the work, the proposed methods were confronted with actual high-dimensional process data from a practical but, in terms of available samples N, small-scale problem. Most of the presented methods are more sensitive to the increase in the number of dimensions M than in the sample count N. Thus, it can be rightfully assumed that the methods will scale well with larger databases.

Future work will emphasize the improvement of the visualization tool and the integration of the algorithms and tools into the existing industrial environment for meaningful large-scale method application, assessment, and improvement based on more comprehensive data and data containing heretofore unknown information on the process.

Acknowledgments

The contributions of Michael Eberhardt and Robert Wenzel to the QuickCog System and Acoustic Navigator are gratefully acknowledged. Michael Eberhardt made part of this work feasible by contributing a data-converting tool and adapting Acoustic Navigator to the task presented. Thanks go to Bernd Vollmer and Christian Esser for friendly support and encouragement and to Klaus Franke for providing the photographs in Section 2.2.

References

2.1 Aarts, E., and Korst, J., *Simulated Annealing and Boltzmann Machines*, Addison Wesley, 1988.

2.2 Semiconductor Industry Association, *International Technology Roadmap for Semiconductors*, Semiconductor Industry Association, San Jose, CA, http://notes.sematech.org/ntrs/Rdmpmem.nsf, 1999.

2.3 Braha, D., and Shmilovici, A., Data mining for improving a cleaning process in the semiconductor industry, *IEEE Transactions on Semiconductor Manufacturing*, 15(1):91–101, Feb. 2002.

2.4 Broomhead, D. S., and Lowe, D., Multivariable functional interpolation and adaptive networks, in *Complex Systems 2*, pp. 321–55, 1988.

2.5 Collins, E., Glosh, S., and Scofield, C., Neural network decision learning system applied to risk analysis: Mortgage underwriting and delinquency risk assessment, in *DARPA Neural Network Study Final Report – Appendix E, Technical Report 840*, pp. 65–79. Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, MA, March 1989.
2.6 Cooper, L. N., Elbaum, C., Reilly, D. L., and Scofield, C. L., Parallel, multiunit, adaptive, nonlinear pattern class separator and identifier, in *United States Patents, Patent Number: 4,760,604*, July 1988.
2.7 Quadrillion Corporation, Q-Yield, http://www.quadrillion.com, 2002.
2.8 Devijver, P. A., and Kittler, J., On the edited nearest neighbor rule, in *Proc. 5th International Conference on Pattern Recognition*, vol. 1, pp. 72–80, Dec. 1980.
2.9 Dzwinel, W., How to make Sammon's mapping useful for multidimensional data structure analysis, in *Pattern Recognition*, 27(7), Elsevier Science Ltd, pp. 949–59, 1994.
2.10 Eberhardt, M., Hecht, R., and König, A., Einsatz des Konzepts *Machine-in-the-Loop-Learning* zum individuellen, robusten Anlernen von Laborrobotersystemen, in *KI-Zeitschrift*, No. 2, pages 44–7, 2002.
2.11 Eberhardt, M., Kossebau, F. K. H., and König, A., Automatic feature selection by genetic algorithms, in *Proc. Int Conf. on Artificial Neural Networks and Genetic Algorithms, ICANNGA01*, pages 256–9, Prague, April 2001.
2.12 Fukunaga, K., *Introduction to Statistical Pattern Recognition.* Academic Press, Harcourt Brace Jovanovich, Publishers, Boston, San Diego, New York, London, Sydney, Tokyo, Toronto, 1990.
2.13 Gates, G. W., The reduced nearest neighbour rule, in *IEEE Transactions on Information Theory*, vol. IT-18, pp. 431–3, 1972.
2.14 Goser, K., Marks, K. M., Rückert, U., and Tryba, V., Selbstorganisierende Karten zur Prozessüberwachung und -voraussage, in *Proc. 3. Int. GI-Kongress über wissensbasierte Systeme, München (16.-17. Okt.)*, pp. 225–37. Informatik Fachberichte Nr. 227, Berlin: Springer Verlag, 1989.
2.15 Katayama, R., Watanabe, M., Kuwata, K., Kajitani, Y., and Nishida, Y., Performance of self-generating radial basis function for function approximation, in *Proc. International Joint Conference on Neural Networks IJCNN'93*, Nagoya, Japan, Vol.I, pp. 471–4, IEEE, 1993.
2.16 Kittler, J., *Feature Selection and Extraction*, Academic Press, Inc., Tzai. Y. Young, King Sun-Fu, Publishers, Orlando, San Diego, New York, Austin, London, Montreal, Sydney, Tokyo, Toronto, 1986.
2.17 Kober, R., Howard, C., and Bock, P., Anomaly detection in video images, in *Proc. 5th International Conference on Neural Networks and Their Applications NEURO-NIMES'92*, 1992.
2.18 Köhler, C., König, A., Temelkova-Kurktschiev, T., and Hanefeld, M., Application of interactive multivariate data visualisation to the analysis of patients findings in metabolic research, in *Proc. 3rd Int. Conf. on Knowledge-Based Intelligent Information Engineering Systems KES'99*, pp. 397–402, Adelaide, Australia, Aug. 1999.
2.19 Kohonen, T., *Self-Organization and Associative Memory*, Springer-Verlag, Berlin, Heidelberg, London, Paris, Tokyo, Hong Kong, 1989.
2.20 König, A., Neuronale Strukturen zur sichtgestützten Oberflächeninspektion von Objekten in industrieller Umgebung, Darmstädter Dissertation D 17 (available from http://www.iee.et.tu-dresden.de/~koeniga), Sept. 1995.
2.21 König, A., A novel supervised dimensionality reduction technique by feature weighting for improved neural network classifier learning and generalization, in

Proc. 6th Int. Conf. on Soft Computing and Information/Intelligent Systems IIZUKA'2000, pp. 746–53, Iizuka, Fukuoka, Japan, Oct. 2000.

2.22 König, A., Dimensionality reduction techniques for multivariate data classification, interactive visualization, and Analysis – Systematic feature selection vs. extraction, in *Proc. 4th Int. Conf. on Knowledge-Based Intelligent Engineering Systems & Allied Technologies KES'2000*, pp. 44–56, University of Brighton, UK, Aug. 2000.

2.23 König, A., Interactive visualization and analysis of hierarchical neural projections for data mining. in *IEEE Trans. on Neural Networks, Special Issue for Data Mining and Knowledge Discovery*, pp. 615–24, May 2000.

2.24 König, A., Dimensionality reduction techniques for interactive visualisation, exploratory data analysis, and classification, in Pal, N. R. (ed.), *Pattern Recognition in Soft Computing Paradigm*, vol. 2, chap. 1, pp. 1–37, World Scientific, FLSI Soft Computing Series, Singapore, Jan. 2001.

2.25 König, A., Blutner, F. E., Eberhardt, M., and Wenzel, R., Design and application of an acoustic database navigator for the interactive analysis of psychoacoustic sound archives and sound engineering, in Hsu, C. (ed.), *Advanced Signal Processing Technology by Soft Computing*, vol. 1, chap. 3, pp. 36–65, World Scientific, FLSI Soft Computing Series, Singapore, Nov. 2000.

2.26 König, A., Bulmahn, O., and Glesner, M., Systematic methods for multivariate data visualization and numerical assessment of class separability and overlap in automated visual industrial quality control, in *Proc. 5th British Machine Vision Conf. BMVC'94*, pp. 195–204, Sept. 1994.

2.27 König, A., Eberhardt, M., and Wenzel, R., A transparent and flexible development environment for rapid design of cognitive systems, in *Proc. EUROMICRO'98 conference, Workshop Computational Intelligence*, Publisher IEEE CS, pp. 655–62, Västeraas, Sweden, Aug. 25–27 1998.

2.28 König, A., Eberhardt, M., and Wenzel, R., QuickCog self-learning recognition system – Exploiting machine learning techniques for transparent and fast industrial recognition system design, in *Image Processing Europe*, pp. 10–9, PennWell, Sept./Oct. 1999.

2.29 König, A., Eberhardt, M., and Wenzel, R., QuickCog – HomePage, in http://www.iee.et.tu-dresden.de/~ koeniga/QuickCog.html, 2000.

2.30 König, A., Raschhofer, R., and Glesner, M., A novel method for the design of radial-basis-function networks and its implication for knowledge extraction, in *IEEE International Conference on Neural Networks*, vol. III, Orlando, pp. 1804–9, Piscataway, NJ, June/July 1994.

2.31 König, A., Windirsch, P. and Glesner, M., Massively parallel VLSI-implementation of a dedicated neural network for anomaly detection in automated visual quality control, in *Proc. 4th Int. Conf. on Microelectronics for Neural Networks and Fuzzy Systems*, pp. 354–63, Sept. 1994.

2.32 Koontz, W. L. G., and Fukunaga, K., A nonlinear feature extraction algorithm using distance transformation, in *IEEE Transactions on Computers C-21*, no. 1, pp. 56–63, 1972.

2.33 Kozma, R., Kitamura, M., Sakuma, M., and Yokoyama, Y., Anomaly detection by neural network models and statistical time series analysis, in *Proc. International Conference on Neural Networks ICNN'94, Orlando, Vol.V*, pp. 3207–10, IEEE, 1994.

2.34 Lee, R. C. T., Slaggle, J. R., and Blum, H., A triangulation method for the sequential mapping of points from N-space to two-space, in *IEEE Transactions on Computers C-26*, pp. 288–92, 1977.

2.35 Lemarie, B., Size reduction of a radial basis function network, in *Proc. International Joint Conference on Neural Networks IJCNN'93, Nagoya, Japan, Vol.I*, pp. 331–4, IEEE, 1993.
2.36 Ludwig, L., Epperlein, U., Kuge, H.-H., Federl, P., Koppenhoefer, B., and Rosenstiel, W., Classification of fingerprints of process control monitoring-data with self-organizing maps, in *Proc. of EANN97, Stockholm, June 16*, pp. 107–12, 1997.
2.37 Ludwig, L., Pelz, E., Kessler, M., Sinderhauf, W., Koppenhoefer, B., and Rosenstiel, W., Prediction of functional yield of chips in semiconductor industry applications, in *Proc. of EANN98, Gibraltar, June 12-14*, pp. 157–161, 1998.
2.38 Marks, K. M., and Goser, K., Analysis of VLSI process data based on self-organizing feature maps, in *Proc. of Neuro-Nimes, Nimes (15.-17. Nov.)*, pp. 337–48, 1988.
2.39 Masa, P., Hoen, K., and Wallinga, H., A high-speed analog neural processor, in *IEEE Micro*, pp. 40–50, IEEE Computer Society, June 1994.
2.40 Moore, G .E., Cramming more components onto integrated circuits, *Electronics Magazine*, 38:114–7, 1965.
2.41 Parzen, E., On estimation of a probability density function and mode, in *Ann. Math. Stat., No.33*, p. 1065, 1962.
2.42 Platt, J., A resource-allocating network for function interpolation, in *Neural Computation, Vol.3*, pp. 213–25, 1991.
2.43 Poggio, T., and Girosi, F., Networks for approximation and learning, in *Proc. IEEE, Vol.78, No.9*, pp. 1481–97, 1990.
2.44 Powell, M. J. D., *Radial Basis Functions for Multivariable Interpolation*, Clarendon Press, Oxford, 1987.
2.45 Raymer, M. L., Punch, W. F., Goodman, E. D., Kuhn, L. A., and Jain, A. K., Dimensionality reduction using genetic algorithms, *IEEE Transactions on Evolutionary Computation*, 4(2):164–71, July 2000.
2.46 Reilly, D. L., Cooper, L. N., and Elbaum, C., A neural model for category learning, in *Biological Cybernetics, 45*, pp. 35–41, 1982.
2.47 Rückert, U., Softwareumgebung DANI zur schnellen explorativen Analyse sehr grosser Datenbestände, http://wwwhni.uni-paderborn.de/sct/cognitronics/#, 2002.
2.48 Sammon, J. W., A nonlinear mapping for data structure analysis, in *IEEE Transactions on Computers C-18, No.5*, pp. 401–9, 1969.
2.49 Sammon, J. W., Interactive pattern analysis and classification, in *IEEE Transactions on Computers C-19, No.7*, pp. 594–616, 1970.
2.50 Smith, S. .D. G., and Escobedo, R. A., Engineering and manufacturing applications of ART-1 neural networks, in *Proc. International Conference on Neural Networks ICNN'94, Orlando, Vol. VI*, pp. 3780–5, IEEE, 1994.
2.51 IDS Software Systems, dataPOWERSsc – Software for Semiconductor Analysis and Data Management, http://www.idsusa.com/site/products/datapower.html, 2002.
2.52 Knights Technology, KnightsYield Management Products (Data Explorer, Yield Manager), http://www.eletroglas.com/products/knights_datasheets/, 2002.
2.53 Turney, P., Data engineering for the analysis of semiconductor manufacturing data, in *Proc. of IJCAI Workshop on Data Engineering for Inductive Learning*, pp. 1–10, 1995.
2.54 Ultsch, A., and Siemon, H. P., Exploratory data analysis: Using Kohonen networks on transputers, in *Interner Bericht Nr. 329 Universität Dortmund, Dezember 1989*, 1989.