Pairwise Proximity-Based Features for Test Escape Screening

Fan Lin, Chun-Kai Hsu, Alberto Giovanni Busetto, and Kwang-Ting Cheng
Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106

***Abstract*—Test escapes are chips that pass the chip-level test program but fail system-level test or in the field. It is known that statistical analysis based on chip production test data could identify abnormalities for screening test escapes. It has also been shown that from the chip test data, we can generate revealing features for statistical analysis by comparing the measurement data to different references such as the measurement mean of a wafer, the spatial pattern of a wafer, and the measurements of neighboring chips. Given these existing features as the base features, this paper proposes a new class of transformations which could generate additional informative features based on pairwise proximities between chips on the same wafer.**

Specifically, we apply multiple distance functions in a feature space composed of the base features and calculate the corresponding pairwise proximities between each pair of chips. Each of the resulting proximities could potentially embed some unique information that reveals the abnormalities of some test escapes. Then we convert the proximities into Euclidean vector spaces using constant shift embedding (CSE), which preserves the cluster structure through the conversion, so that traditional outlier analysis algorithms such as local outlier factor (LOF) can be applied. The LOF value and the first dimension in each embedded space are used as additional features for each sample. These new features, jointly analyzed with the base features, provide more revealing information about test escapes and thus further improve the test escape detection rate in our experiment based on production test data.

I. Introduction

Modern test programs have been including more tests to accommodate the growing chip complexity and stringent quality requirements. Even with the large number of tests applied under multiple environment settings with different temperatures and supply voltages, there still are chips that pass the entire test program but fail at system-level applications. Such chips are referred to as *test escapes* and incur significant cost. Therefore, our goal is to apply statistical analysis on a huge amount of production test data for screening test escapes. The analysis is referred to as *statistical test* [1] and does not require any additional physical measurements.

For screening test escapes, a well-adopted method proposed by the Automotive Electronics Council is the part average test (PAT) [2]. For some specific test items, PAT compares a query chip's measurements with the distribution of a population of reference chips, and discard the chip if it is an outlier. While PAT examines each test item independently, other methods investigate the multivariate properties of the chips using principal component analysis (PCA) [3], [4] and Mahalanobis distance [5]. In this paper, we propose a set of transformations for generating informative features that can reveal the abnormalities of test escapes compared with the population of good chips.

A concept of *residual* was proposed in [6] for outlier screening, and its applications using the neighboring chips' measurements as the reference for deriving the residual, called *nearest neighbor residual* (NNR), have been studied in [7]–[10]. In [11], it was demonstrated that using *residual vectors*, which are vectors composed of the differences between a chip's measurement values and expected values for each test item, as the features for statistical analysis could result in effective test escape screening. Three expected values were suggested in [11]: the mean of the measurements of chips on the same wafer, a value based on the bilateral filtered [12] spatial pattern of the wafer, and the median of the query chip's eight nearest neighbors. For each test item, a chip's measurement value is compared with the three expected values to reveal different aspects of the abnormalities of test escapes. Based on the three expected values, three feature sets were generated. [13] and [11] showed that each of the feature sets consists of information that could uniquely reveal some of the test escapes.

In this paper, we propose new features that are based on the pairwise proximities calculated from the abovementioned three feature sets, which are referred to as the *base feature sets* in the rest of the paper. Given a test data with T test items and D chips, we calculate a $D \times D$ proximity matrix based on a selected proximity/distance function in the $3T$ feature space constructed from the base feature sets. Different distance functions could potentially provide unique information that reveals the abnormalities of some test escapes. In this paper we investigate six different distance functions including *cosine distance*, *correlation distance*, and *Minkowski distance* with $p = 1, 2, 3, \infty$ for deriving the proximities.

The *proximity representation* of a dataset (represented by a $D \times D$ proximity matrix), however, could not be analyzed with the traditional machine learning algorithms that are designed for a Euclidean space, or a *vector representation* (represented by a $D \times T$ matrix containing all the feature values for all samples). Therefore, we apply a technique named *constant shift embedding* (CSE) [14] to convert the proximity information back into a Euclidean space. The most prominent property of CSE is the complete preservation of cluster structure in the embedded Euclidean space. The six distance functions would lead to six unique proximity matrices, and therefore results in six unique embedded spaces. In addition, we further investigate a traditional kernel PCA (kPCA) embedding method [15] with radial basis function (RBF) kernel and generate a seventh Euclidean space to provide even more information that could potentially separate test escapes.

In each of the seven unique embedded spaces constructed based on the pairwise proximities calculated from the base feature sets, we apply a density-based outlier analysis called *local outlier factor* (LOF) [16]. LOF compares a sample's local density with its neighbors' densities in a feature space and produces a single value. A sample with a relatively higher LOF value than that of the majority of the samples is more likely to be an outlier. We also observed that in the first dimension of each embedded space, some test escapes are away from the good chip population, which makes them easily separable. Therefore, we use the LOF value and the first dimension of each of the embedded spaces jointly as the new proximity-based features. Given the seven proximity definitions, 14 new features are generated. Based on these new features plus the $3T$ features from the base feature sets, we then perform feature reduction using *canonical analysis* [13]. Canonical analysis is a linear transformation that maximizes the separation between the two populations of samples (good chips and test escapes) in the first few dimensions of the transformed feature space, called a canonical space. A classical classifier such as the support vector machine (SVM) [17], [18] can then be applied in the canonical space for more efficient and in some cases, more effective classification [13]. With these proximity-based features which provide additional revealing information about test escapes, the test escape detection rate based on an industrial production test dataset is improved to 31%, compared with 27% for similar analysis using the base feature sets only.

The rest of the paper is organized as the following: Section II introduces the distance functions used for generating the pairwise proximities between samples. Section III illustrates the concepts and properties of constant shift embedding, followed by a discussion about the distribution of the chips in the embedded space. Section IV discusses data standardization, feature generation, outlying wafer detection, and feature transformation using canonical analysis. Additional experimental results are presented in Section V, and Section VI concludes the paper.

II. Pairwise Proximity

In this paper we extract more information to reveal the abnormalities of the test escapes by comparing each chip with all the other chips on the same wafer. The comparison is made in a feature space composed of the three base feature sets developed in [11], in which each chip with T test measurements is characterized by a $T \times 1$ residual vector $\mathbf{r}$:

$$\mathbf{r} = \mathbf{x_m} - \mathbf{x_e} \tag{1}$$

where $\mathbf{x_m}$ is a $T \times 1$ vector of the measured values and $\mathbf{x_e}$ is a $T \times 1$ vector of the expected values. Three expected values were used to produce three base feature sets: the mean of the measurements on the same wafer, a value based on the bilateral filtered [12] spatial pattern of the wafer, and the median of the eight nearest neighbors of the query chip [11]. The third feature set can be considered as a special case of NNR [6]. We denote the three generated feature sets as F_M, F_B, and F_N respectively. Throughout the analysis in this paper, we will be using the three feature sets jointly, denoted $F_M \cup F_B \cup F_N$, as our base feature space for deriving the proximities.

Given a wafer with D chips, the pairwise comparisons between each pair of chips result in a $D \times D$ symmetric proximity matrix, each of whose elements represents the pairwise proximity between two chips. Our strategy is to generate all potentially useful features, followed by a feature reduction technique such as canonical analysis [13] to automatically extract the most useful information out of the large set of generated features for classification. Therefore, we apply multiple different distance functions for calculating the pairwise proximity between each two chips to potentially reveal more aspects of the abnormalities of test escapes with the conjecture that each of these distance functions might uniquely separate some of the test escapes from the normal populations. Let $\mathbf{x_a}$ be a $T \times 1$ vector consisting of sample a's feature values $x_{a1}, x_{a2}, ..., x_{aT}$, the distance functions we investigated are:

- *Cosine distance*:

$$d_{ab} = 1 - \frac{\mathbf{x_a}'\mathbf{x_b}}{\sqrt{(\mathbf{x_a}'\mathbf{x_a})(\mathbf{x_b}'\mathbf{x_b})}} \tag{2}$$

- *Correlation distance*:

$$d_{ab} = 1 - \frac{(\mathbf{x_a} - \overline{\mathbf{x}}_\mathbf{a})'(\mathbf{x_b} - \overline{\mathbf{x}}_\mathbf{b})}{\sqrt{(\mathbf{x_a} - \overline{\mathbf{x}}_\mathbf{a})'(\mathbf{x_a} - \overline{\mathbf{x}}_\mathbf{a})(\mathbf{x_b} - \overline{\mathbf{x}}_\mathbf{b})'(\mathbf{x_b} - \overline{\mathbf{x}}_\mathbf{b})}} \tag{3}$$

where $\overline{\mathbf{x}}_\mathbf{a}$ is the mean of vector $\mathbf{x_a}$.

The following four distances are derived from *Minkowski distance* with different values for parameter p:

$$d_{ab} = \sqrt[p]{\sum_{j=1}^{T} |\mathbf{x_{aj}} - \mathbf{x_{bj}}|^p} \tag{4}$$

- $p = 1$ *(Manhattan distance)*:

$$d_{ab} = \sum_{j=1}^{T} |\mathbf{x_{aj}} - \mathbf{x_{bj}}| \tag{5}$$

- $p = 2$ *(Euclidean distance)*:

$$d_{ab} = \sqrt{(\mathbf{x_a} - \mathbf{x_b})'(\mathbf{x_a} - \mathbf{x_b})} \tag{6}$$

- $p = 3$:

$$d_{ab} = \sqrt[3]{\sum_{j=1}^{T} |\mathbf{x_{aj}} - \mathbf{x_{bj}}|^3} \tag{7}$$

- $p = \infty$ *(Chebyshev distance)*:

$$d_{ab} = \max_j |\mathbf{x_{aj}} - \mathbf{x_{bj}}| \tag{8}$$

In addition to the above six distance functions, we also include a traditional kernel PCA method [15] with a radial basis function (RBF) kernel:

- *RBF/Gaussian kernel*:

$$k_{ab} = \exp\left(-\frac{(\mathbf{x_a} - \mathbf{x_b})'(\mathbf{x_a} - \mathbf{x_b})}{2\sigma^2}\right) \tag{9}$$

Each of the first six distance functions would lead to a unique proximity matrix, which will be further converted to a Euclidean space by CSE. For the proximity matrix generated using the RBF kernel, we apply the traditional kernel PCA algorithm for producing an embedded space without CSE to validate if the existing kPCA technique could also provide additional information.

III. Constant Shift Embedding

A. Concepts and Properties

After generating the proximity matrices based on multiple distance functions, we need to convert the proximity representation back into a vector representation before applying traditional outlier detection algorithms that are designed for a Euclidean vector space. *Constant shift embedding* (CSE) [14] is a technique to embed pairwise proximity data into the equivalent Euclidean embedding with no distortions. Specifically, CSE finds a Euclidean space in which the cost function of a Euclidean distance-based clustering algorithm such as *k-means* could be equivalent to the cost function of pairwise clustering on the proximity matrix. Detailed computations of CSE are out of the scope of this paper and can be found in [14].

Fig. 1 shows the process of producing new embedded feature spaces based on proximity matrices. From the original space O_1, which is composed of the base feature sets $F_M \cup F_B \cup F_N$, we generate multiple proximity matrices based on different distance functions, followed by CSE for each proximity matrix to convert them into embedded Euclidean spaces. While CSE preserves the cluster structure through the conversion from a proximity representation to a Euclidean vector representation (e.g., the cluster structure is preserved between E_1 and P_1, between E_2 and P_2, and so on), the k-means cost function in the original feature space O_1 would also be identical to the cost function of pairwise clustering in the proximity matrix derived using Euclidean distance P_1 [19]. Therefore, applying k-means clustering in E_1 is equivalent to applying k-means clustering in O_1. In this special case, in fact, the original space is already Euclidean. The added value of CSE in our analysis comes from the ability to assimilate also other arbitrary measures of proximity into more informative Euclidean spaces.

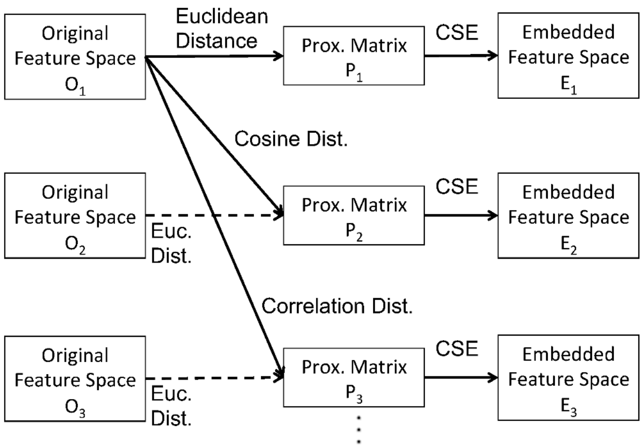


Fig. 1: The conversion between proximity matrices and Euclidean spaces. CSE preserves the cluster structure through the conversion from a proximity matrix P_i to an embedded Euclidean space E_i.

For a proximity matrix derived from O_1 with distance functions other than Euclidean distance, e.g. P_2 using cosine distance, we can also consider it is derived from a virtually equivalent original feature space O_2 using Euclidean distance. In such case, applying k-means in E_2 would be identical to applying k-means in O_2. However, with the use of nonlinear distance functions to derive the proximities, a direct transformation from O_1 to O_i, where $i \neq 1$, is often not feasible. Analyzing E_i through the calculation of P_i followed by CSE, achieves the same goal without the need of finding O_i.

CSE involves eigendecomposition of the proximity matrix [14]. That is, the embedded space is composed of the eigenvectors of the proximity matrix. In our application, we analyze only the first few dimensions, which have relatively significant eigenvalues, for feature reduction. Let u be the number of eigenvectors found from the eigendecomposition, and $ev_1, ev_2, ..., ev_u$ be the sorted eigenvalues such that $ev_1 \geq ev_2 \geq ... \geq ev_u$, we estimate the number of effective dimensions of the embedded space by:

$$D_{eff}(i) = \frac{\sum_{j=1}^{u} ev_j}{ev_1} \qquad (10)$$

For each embedded space E_i, we apply the outlier analysis algorithm in its corresponding dimensionality of $ceil(D_{eff}(i))$.

Note that for the seventh Euclidean space, we apply the traditional kernel PCA approach with an RBF kernel for generating the proximity and deriving an embedded space, without applying CSE. Our overall strategy is to generate as many potentially useful features as possible. Since kPCA is known to be one of the potentially useful transformations, it is worthwhile to include it to enrich our analysis. In the experimental results demonstrated later, we validate that features based on E_i's and the kPCA space are both useful for further improvement of classification accuracy.

B. Distribution in the Embedded Space

For one exemplar wafer with two test escapes, Fig. 2 shows the distributions of good chips (blue dots) and the test escapes (red crosses) in the embedded spaces constructed based on the six types of proximities, with the number of effective dimensions marked above each distribution. The distribution in the embedded space constructed using kPCA is shown in Fig. 3.

It is clear that, in all the distributions, the good chip population exhibits a bimodal distribution - the good chips on a wafer are separated into two clusters through the proximity calculation and the CSE transformation. Fig. 4 shows the mapping of the distribution of good chips on the exemplar wafer in the embedded space constructed based on cosine distance to a wafer map. In Fig. 4a, chips are colored based on their locations, and the same colors are marked on the wafer map in Fig. 4b to indicate the corresponding locations of the chips on wafer. From the high-dimensional base feature space $F_M \cup F_B \cup F_N$, the proximities are able to reveal the underlying horizontal stripe pattern on the wafer even though in most test items this pattern are not directly observable and shadowed by some other more dominant types of spatial patterns. In other words, such stripe spatial variation may be subtle but consistently exists in most of the test items. Finding such hidden spatial patterns, which is feasible using the proposed analysis with proximities based on different nonlinear distance functions, could help the diagnosis of manufacturing/testing process and equipment such as multi-site probing.

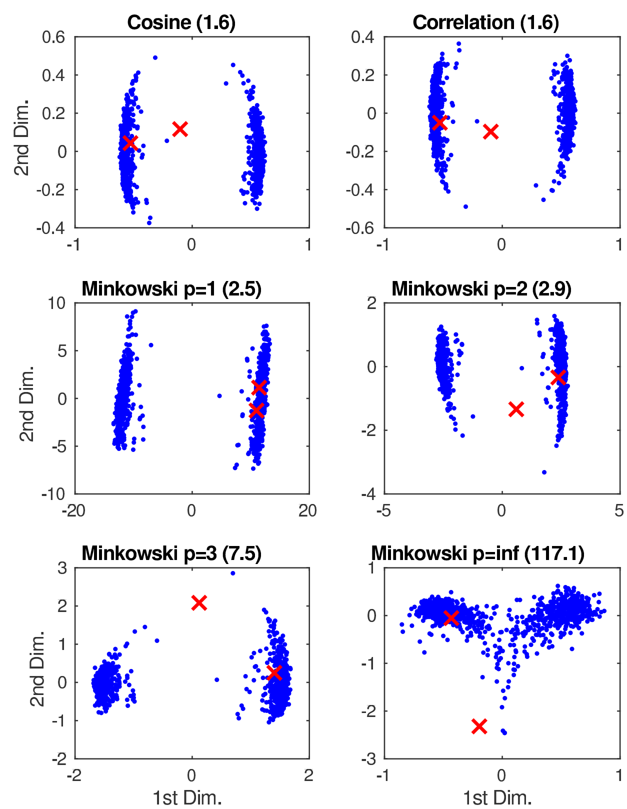


Fig. 2: The distributions of the chips on a wafer in the first two dimensions of the CSE embedded spaces based on six different proximity/distance functions. Blue dots represent the good chips and red crosses mark the positions of test escapes. The numbers of effective dimensions are shown above each figure.

Defined in (10), the number of effective dimensions in embedded spaces based on cosine, correlation, Manhattan (Minkowski with $p =$

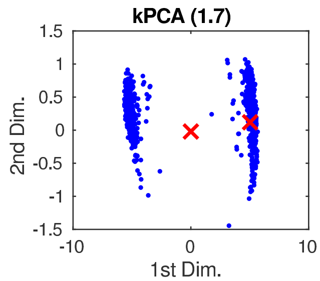
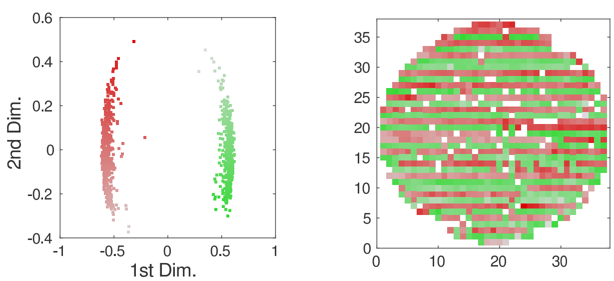


Fig. 3: The distribution in the first two dimensions of the embedded space constructed based on kPCA with RBF kernel.

1), and Euclidean (Minkowski with $p = 2$) distances are typically no greater than 3. In general, Minkowski distance with greater p leads to a greater number of effective dimensions, and Minkowski distance with a very small p, say 1, generates little information and is insufficient to expose the test escapes as outliers. Details of how to analyze the distributions for screening test escapes will be discussed in Section IV-B.



(a) Distribution of good chips in an embedded space

(b) Corresponding positions on wafer

Fig. 4: Color-coded distributions of good chips showing the correspondence of chips in the embedded space and on the wafer. Chips are colored to show their corresponding positions.

IV. Data Preparation and Feature Processing

In this section we discuss how we preprocess the production test data, generate new features from the embedded spaces, and transform the features for feature reduction. We also demonstrate a process to identify and remove some abnormal wafers from our analysis.

A. Data Standardization

As proposed in [11], to minimize the wafer-to-wafer variation in production test data, we first standardize the measurement values of each wafer before further analysis. For each test item in each wafer, we identify outlying measurements using the general Extreme Studentized Deviate (ESD) test [20], and calculate the *robust mean* μ and *robust standard deviation* σ excluding the outlying measurements. We then standardize the measurements x in each wafer individually to *z-score* by:

$$z = \frac{x - \mu}{\sigma} \tag{11}$$

Given an upper bound for the number of outliers h, the general ESD test performs h hypothesis tests: a test for one outlier, a test for two outliers, and so on up to h outliers to conclude the number of outliers and identify them. Detailed implementation of the general ESD test can be found in [21].

B. Feature Generation

As observed in Fig. 2, in all embedded spaces except one constructed based on Manhattan distance, which takes into account only the first order difference between chips, one of the two test escapes is exposed as abnormal and far from the bimodal distribution of the good chips, while the other test escape is indistinguishable from the good chips. Since our goal is to maximize the test escape detection rate while minimizing the amount of induced yield loss (good chips misclassified as test escapes), the classification accuracy would be higher if test escapes could be outlying in as many embedded spaces as possible, and the good chips that happen to be outlying in one embedded space to be closer to the normal population in other embedded spaces. Therefore, although one embedded space seems sufficient to expose the test escape as an outlier in Fig. 2, it improves the robustness of the method to include the distribution information in all embedded spaces for further analysis.

To analyze the distributions in multiple embedded spaces jointly, we convert the outlying level of each chip in each embedded space to a score, defined by *local outlier factor* (LOF) [16]. LOF is an outlier analysis algorithm that compares the local density of the sample with the densities of its neighbors. Let *k-distance*(p) be the distance between sample p and its k-th nearest neighbor, a *reachability distance* is defined by:

$$\textit{reach-dist}_k(p, q) = max\{\textit{k-distance}(q), d(p, q)\} \tag{12}$$

where $d(p, q)$ denotes the distance from p to q. Including *k-distance(p)* in the reachability distance could produce a more stable result than using $d(p, q)$ directly.

Using a parameter $MinPts$ for k, the *local reachability density* of p is defined as:

$$lrd_{MinPts}(p) = 1/\left(\frac{\sum_{q \in N_{MinPts}(p)} \textit{reach-dist}_{MinPts}(p, q)}{|N_{MinPts}(p)|}\right) \tag{13}$$

where $N_{MinPts}(p)$ is the set of $MinPts$ nearest samples of p. Discussions about choosing the upper and lower bounds for $MinPts$ can be found in [16]. In our analysis, the range is set to $5 \leq MinPts \leq 10$.

The local outlier factor is then defined as:

$$LOF_{MinPts}(p) = \frac{\sum_{q \in N_{MinPts}(p)} \frac{lrd_{MinPts(q)}}{lrd_{MinPts(p)}}}{|N_{MinPts}(p)|} \tag{14}$$

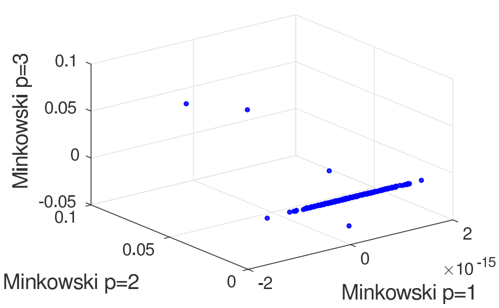
The LOF value is a relative value indicating the outlying level of a sample compared with its neighbors. Typically, an LOF value close to (greater than) 1 tends to indicate an inlier (outlier), but the actual threshold is data dependent. With the local density approach, a sample with some distance to a dense cluster could have a much greater LOF value than another sample with the same distance to a sparse cluster, and thus be exposed as an outlier.

Now that we can express the outlying level of each chip by a single LOF value, we use these LOF values as our new pairwise proximity-based features. Instead of setting a threshold directly on the LOF values, we use the LOF values jointly with other base features for machine learning algorithms such as SVM for classification. Another simple observation from the distributions is that the detectable test escapes, away from the bimodal distribution, are typically closer to the origin in the first dimension of the embedded space. Therefore, we also include the first dimension of the embedded spaces as input features for further analysis. In total, 14 new features are generated from the 7 embedded spaces based on pairwise proximities.

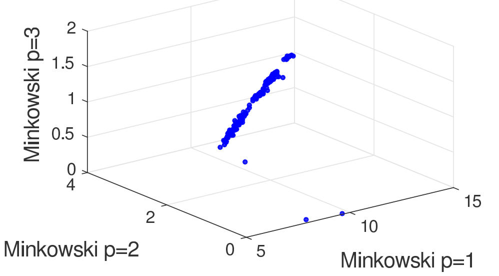
C. Feature Standardization and Outlying Wafer Detection

There exist wafer-to-wafer variations in production test data, and we standardize each wafer individually with respect to the robust mean and standard deviation before any analysis to remove the shifting and scaling variations. However, although all the wafers we analyzed exhibit the bimodal distributions as in Fig. 2, we have observed noticeable variations in the 14 new features, especially the LOF values since they are relative values depending on the local distribution. Thus, we further standardize the new features generated from each wafer to z-scores using the robust mean and standard deviation calculated from each wafer, as mentioned in Section IV-A, to remove some higher order wafer-to-wafer variations that were not eliminated in the first standardization.

Fig. 5 demonstrates the robust mean and standard deviation of three of the new features: the first dimension in the embedded spaces constructed using Minkowski distance with $p = 1, 2, 3$ as the proximity measure. Each dot in the figure represents the statistics of one wafer. In Fig. 5a, most of the wafers have their robust means very close to zero in all three features, and in Fig. 5b, the robust standard deviation in the first dimension of the embedded space from Minkowski distance with $p = 1$ and that with $p = 3$ are highly correlated, while the variation in the dimension of Minkowski $p = 2$ is relatively negligible. More importantly, both Figs. 5a and 5b show some outliers away from the normal distribution. The wafers with these outlying values have very different characteristics in the new features from the majority of the wafers and should be excluded from statistical analysis.



(a) Robust mean



(b) Robust standard deviation

Fig. 5: The robust mean and standard deviation of each wafer in the feature space of three proximity-based features.

While Fig. 5 provides an example to visualize these outliers in three selected features, we can also apply LOF or some simpler outlier analyses such as Mahalanobis distance [22] to quantitatively expose these outlying wafers. Mahalanobis distance is defined as:

$$d_{ab} = \sqrt{(\mathbf{x_a} - \mathbf{x_b})' \mathbf{C}^{-1} (\mathbf{x_a} - \mathbf{x_b})} \tag{15}$$

where C is the covariance matrix of the dataset. Intuitively, equation (15) computes the distance between two samples in a Euclidean space that is normalized with respect to the covariance matrix of the original Euclidean space, and therefore reveals outliers that has a smaller Euclidean distance to the major population but lies out of the shape of the major population's distribution.

D. Feature Transformation and Classification

After the generation and standardization of the proximity-based features, we analyze them jointly with the base features for test escape screening. Our objective has been generating *potentially revealing* features without custom investigation for each dataset of which features are really more informative for test escape screening. Our framework creates a general collection of potentially useful features that can be applied to any dataset/product, which are suitable for known feature reduction and classification algorithms to automatically extract the most useful information out of them for high accuracy classification. In our experiments, we employ *canonical analysis* [13] to the joint feature sets, consisting of the base features and the proximity-based features, for feature reduction. Canonical analysis is a linear transformation which compacts the multi-dimensional separation between classes of samples into the first few dimensions in a transformed canonical space. In our analysis for test escape screening, there are two classes of samples: test escapes (*positive* samples) and good chips (*negative* samples), and compacting the separation in the high-dimensional feature space into a small number of features has been demonstrated to achieve significant runtime reduction and in some cases, greater classification accuracy, based on a conventional classifier such as SVM [13]. Specifically, we apply *C-support vector classification* (C-SVC) provided by LIBSVM [18] as the final classifier. The complete flow of generating the proximity-based features for statistical analysis is illustrated in Fig. 6.

V. Experimental Results

In this section we present the results of analyzing the proposed proximity-based features jointly with the base features derived in [11] on a continue-on-fail production test data of an industrial product. We preprocessed the test data to remove confidential information while preserving all information that is relevant to the analysis. The dataset includes more than 700 wafers with 1000+ chips per wafer. We use 200+ wafers as the training set, 200+ wafers as the validation set for selecting SVM parameters, and the rest 200+ wafers as the testing set. The test program contains more than 200 parametric test items.

A. Test Escape Emulation

As this dataset does not include actual test escape information, we need to *emulate* test escapes for our analysis. The concept is to emulate test escapes using intrinsically defective chips with subtle syndromes in their test measurements. We identified faulty chips with minor failures, i.e. failed only one test item and passed all the other test items in the continue-on-fail test program, and replaced their failing measurements with the median value of their corresponding test items among the population of good chips. Through such process, we *hid* the direct failing evidences of the faulty chips but kept their subtle abnormal syndromes, if any, in all other passing test items intact. After this process, emulated test escapes would have measurement data that pass the entire test program, while their

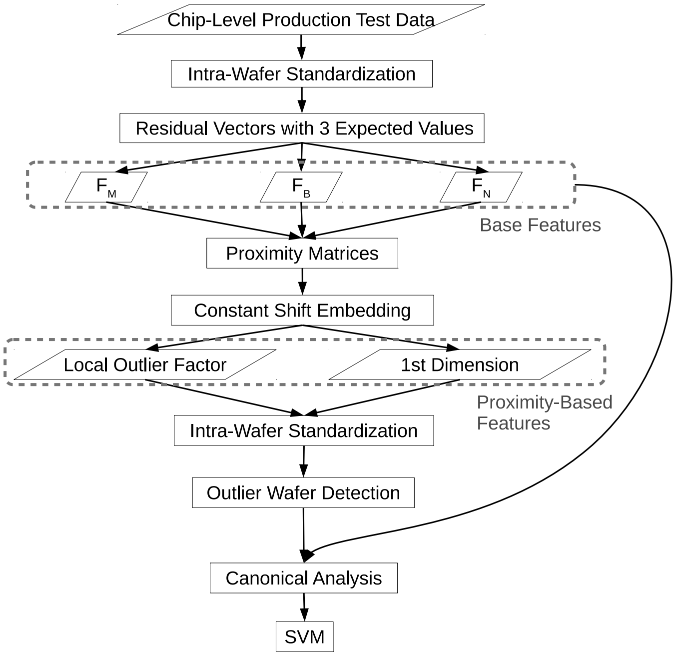


Fig. 6: The complete flow of generating the proximity-based features for statistical analysis.

intrinsic defects could still be reflected in the original non-failing test items.

After producing the initial emulated test escape pool, we further remove a fraction of them to create a scenario with greater test escape diversity and at a more realistic test escape rate. The idea is to generate an emulated test escape pool such that the corresponding test items of those hidden failing measurements are widely spread among a large number of test items and no single test item can directly detect a large number of emulated test escapes. Therefore, we identified test items that hiding each of them would cause an increase of test escape rate by more than 50PPM to test escapes, and removed the test escapes created by hiding these test items from the emulated test escape pool. The resulting test escapes then correspond to approximately 560PPM for the testing set.

B. Classification Accuracy

Fig. 7 demonstrates the relative operating characteristics (ROC) curves, i.e. the test escape detection (true positive) rate vs. the yield loss (false positive) rate, of the classification based on the base features with and without the new proximity-based features. The two ROC curves exhibit different trends and cross each other at a yield loss rate of approximately 0.01%. This indicates that including the proximity-based features does provide more information, otherwise the classification accuracy would not be affected. The additional information provided, however, does not generalize from the training set to the testing set and becomes counter-productive at a very low yield loss rate. Given sufficient yield loss rate ($> 0.01\%$), the additional information from the proximity-based features starts to help classify more test escapes and improves the test escape detection rate to 31%, compared with 27% for using the base features alone at a yield loss rate of 0.027%. Therefore, even after the standardizations on the production test data and on the proximity-based features for each wafer, there still exist some significant discrepancies between the training set and the testing set. The cause of such discrepancies requires further investigation and should be removed to improve the consistency between the training set and the testing set.

Fig. 8 shows the ROC curves of classification based on the base features plus different subsets of the proximity-based features. We investigated the results using the features based on the two embedding methods (CSE and kPCA) individually. Similar to Fig. 7, the test escape detection rates for using the base features plus the features based on each of the two embedding methods are lower than that using only the base features at a lower yield loss rate. In fact, the classification accuracy based on the base features plus the two kPCA-based features (LOF value and the first dimension of the embedded space) never surpasses the classification accuracy based on the base features only, in the range we searched for an optimal pair of SVM parameters [18]. However, including both subsets of the proximity-based features for classification could lead to a significantly greater test escape detection rate than including either of the subsets alone. In this case, incorporating both subsets of the proximity-based features allows the classification to focus on the additional information that can be effectively generalized to the testing set and be free from the discrepancies between datasets.

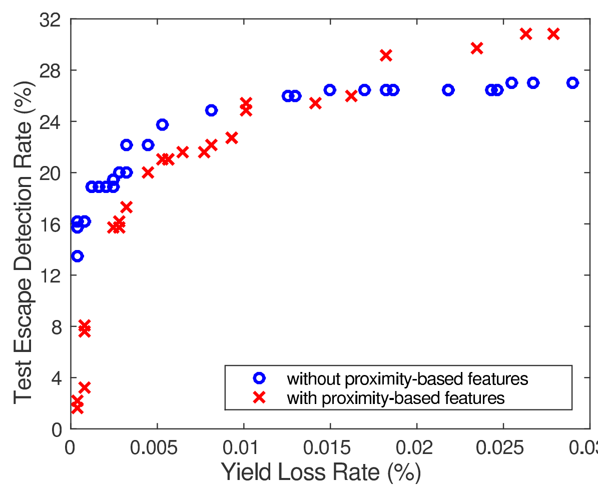


Fig. 7: The ROC curves of classification based on the base features with and without the proximity-based features.

C. Performance Overhead

On average, for one wafer with $1000+$ chips and $700+$ base features, deriving the pairwise proximity and applying CSE takes 2.4 seconds, while applying LOF takes another 2.2 seconds on an Intel Xeon Quad-core 3.6GHz system. Compared with the runtime for the canonical transform followed by SVM classification, which involves simple linear operations and takes 0.02 second per wafer, the runtime for the nonlinear proximity/distance functions and the LOF algorithm is relatively significant. Moreover, the memory usage and runtime for processing the pairwise proximity grows quadratically with respect to the number of chips per wafer. Therefore, a future direction would be to optimize the algorithms and the flow for generating the proposed features for better efficiency.

In principle, whether it makes sense or not to apply the proximity-based features for statistical tests in addition to the existing base features depends on the cost and quality requirement of the products. For example, including proximity-based features in the analysis may not be cost effective for a high-volume product that requires real-time application of the analysis, e.g. chips for mobile devices. On

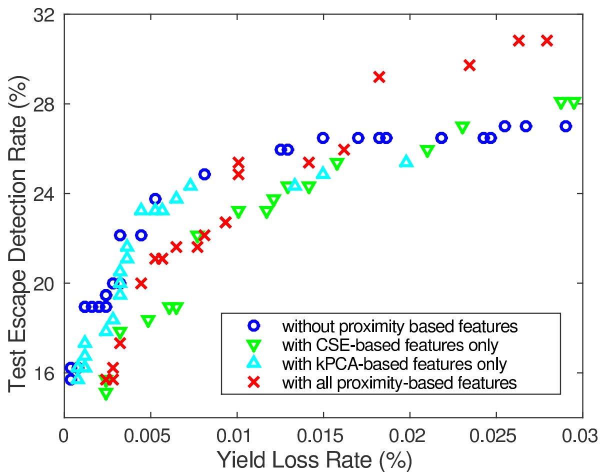


Fig. 8: The ROC curves of classification based on the base features plus different subsets of the proximity-based features.

the other hand, for an extremely quality demanding product that does not require real-time analysis, e.g. processors for centralized servers and chips for safety critical systems, applying the proximity-based features for offline statistical tests could help screen more test escapes without incurring unacceptable extra cost.

VI. Conclusions

This paper proposes a new set of proximity-based features based on a collection of base features: residual vectors with respect to three different expected values of test measurements. We demonstrate a complete flow of generating additional informative features and the reasoning for each step. To expose the abnormalities of test escapes, the proposed method first compares each chip with all other chips on the same wafer in the feature space composed of the base features, followed by constant shift embedding to embed the proximity matrix into an equivalent Euclidean embedding with no distortions. The outlying level of each chip in the embedded space is then converted into a single score using local outlier factor, and the LOF values, jointly with the first dimension of each embedded space, are used as the new features for test escape screening. The experimental results based on an industrial production test dataset demonstrate that the proximity-based features provide additional information revealing the abnormalities of some test escapes, which further improves the test escape detection rate beyond the state-of-the-art methods that are already comprehensive for test escape detection.

Acknowledgments

We would like to acknowledge the Semiconductor Research Corporation (SRC) and the Industrial Technology Research Institute (ITRI), Hsinchu, Taiwan for their support on this work, and the input by Dr. Chun-Lung Hsu of ITRI.

References

[1] P. M. O'Neill, "Statistical test: A new paradigm to improve test effectiveness & efficiency," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2007.

[2] Automotive Electronics Council, "Guidelines for testing," Dec. 2011.

[3] P. M. O'Neill, "Production multivariate outlier detection using principal components," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2008.

[4] N. Sumikawa, J. Tikkanen, L.-C. Wang, L. Winemberg, and M. S. Abadir, "Screening customer returns with multivariate test analysis," in *Proc. Int'l Test Conf. (ITC)*, Nov. 2012.

[5] S. Krishnan and H. G. Kerkhoff, "Exploiting multiple Mahalanobis distance metrics to screen outliers from analog product manufacturing test responses," *IEEE Design & Test*, vol. 30, no. 3, pp. 18–24, 2013.

[6] W. R. Daasch, J. McNames, D. Bockelman, and K. Cota, "Variance reduction using wafer patterns in IddQ data," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2000.

[7] W. R. Daasch, K. Cota, and J. McNames, "Neighbor selection for variance reduction in IDDQ and other parametric data," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2001.

[8] R. Madge, B. H. Goh, V. Rajagopalan, C. Macchietto, W. R. Daasch, C. Schuermyer, C. Taylor, and D. Turner, "Screening minVDD outliers using feed-forward voltage testing," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2002.

[9] R. Madge, M. Rehani, K. Cota, and W. R. Daasch, "Statistical post-processing at wafersort-an alternative to burn-in and a manufacturable solution to test limit setting for sub-micron technologies," in *Proc. IEEE VLSI Test Symp. (VTS)*, May 2002.

[10] W. R. Daasch and R. Madge, "Variance reduction and outliers: Statistical analysis of semiconductor test data," in *Proc. Int'l Test Conf. (ITC)*, Nov. 2005.

[11] F. Lin, C.-K. Hsu, and K.-T. Cheng, "Adatest: An efficient statistical test framework for test escape screening," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2015.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int'l Conf. Computer Vision*, Jan. 1998.

[13] F. Lin, C.-K. Hsu, and K.-T. Cheng, "Feature engineering with canonical analysis for effective statistical tests screening test escapes," in *Proc. Int'l Test Conf. (ITC)*, Oct. 2014.

[14] V. Roth, J. Laub, M. Kawanabe, and J. Buhmann, "Optimal cluster preserving embedding of nonmetric proximity data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1540–1551, Dec 2003.

[15] B. Schölkopf, A. J. Smola, and K.-R. Müller, "Advances in kernel methods." MIT Press, 1999, pp. 327–352.

[16] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int'l Conf. on Management of Data*, 2000, pp. 93–104.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[18] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. on Intelligent System and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[19] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[20] B. Rosner, "Percentage points for a generalized ESD many-outlier procedure," *Technometrics*, vol. 25, no. 2, pp. 165–172, May 1983.

[21] NIST/SEMATECH, "e-handbook of statistical methods," *http://www.itl.nist.gov/div898/handbook/*, 2003.

[22] P. C. Mahalanobis, "On the generalised distance in statistics," in *Proc. Nat. Inst. Sci. India*, 1936, pp. 49–55.