Hotspot Classification Based on Higher-order Local Autocorrelation

Bin Lin[*a], Zheng Shi[a], Ye Chen[b]
[a]Institute of VLSI Design, Zhejiang University, 38 Zheda Road, Hangzhou 310027, China
[b]Anchor Semiconductor Inc., 3235 Kifer Road, Ste 200, Santa Clara, CA 95051, USA

ABSTRACT

Hotspot classification is an important step of hotspot management. Under possible center-shifting condition, conventional hotspot classification by calculating pattern similarity through overlaying two hotspot patterns directly is not effective. This paper proposes a hotspot classification method based on higher-order local autocorrelation (HLAC). Firstly, we extract the features of the hotspot patterns using HLAC method. Secondly, the principal component analysis (PCA) is performed on the features for dimension reduction. Thirdly, the simplified low dimensional vector features of the hotspot patterns are used in the pre-clustering step. Finally, detailed clustering using pattern similarity calculated by discrete 2-d correlation is carried out. Because the HLAC based features are shifting-invariant, the center-shifting problem caused by the defect location inaccuracy can be overcome during the pre-clustering process. Experiment results show that the proposed method can classify hotspots under center-shifting condition effectively and speed up the classification process greatly.

Keywords: hotspot classification, HLAC, PCA, clustering, discrete 2-d correlation

1. INTRODUCTION

As the technology node advances below 45nm, many of the manufacturability problems are caused by specific layout configurations defined as hotspots which might become yield detractors. Fig.1 shows a typical mask manufacturing and hotspot management flow which starts from design rule clean layout followed by Optical Proximity Correction (OPC) and Model Based Verification (MBV). Although after several iterations of OPC and MBV, the post-OPC layout can pass MBV, it cannot guarantee that there is no defect on the wafer because the model is not perfect especially when the pattern complexity increases. As a result, after wafer process, wafer inspection becomes a necessary step which finds the defect locations that are usually not accurate enough. Subsequently, fake defects which do not affect electrical connection are filtered out in defect filtering step. If there is no defect left after defect filtering, the design becomes the final product design. Otherwise, small pieces of design layout centered on the positions corresponding to the locations found in defect inspection step are clipped from the design layout with a fixed size. These small layout snippets are defined as hotspots. After that, the hotspots are classified into different groups since usually there are too many hotspots to be reviewed one by one. For each group, some representatives are selected out and added into the hotspot library which will be fed back to design stage [1, 2]. Hotspot classification is an important step of such a hotspot management.
Ma and Ghan proposed an automatic hotspot classification method using pattern-based clustering in [3, 4]. It calculates pattern similarity through overlaying two hotspot patterns directly. It is workable for non-center-shifted hotspot patterns, but not suitable for hotspot classification under center-shifting condition.

For higher performance of hotspot classification under center-shifting condition, this paper proposes a hotspot classification method based on higher-order local autocorrelation (HLAC). Section 2 describes the metric which indicates the similarity between two hotspot patterns and discrete 2-d correlation used for pattern similarity calculation under center-shifting condition. Section 3 discusses the clustering algorithm. Section 4 presents the proposed hotspot classification method based on HLAC. Section 5 shows the experiments and results followed by conclusions in Section 6.

*linbin@vlsi.zju.edu.cn

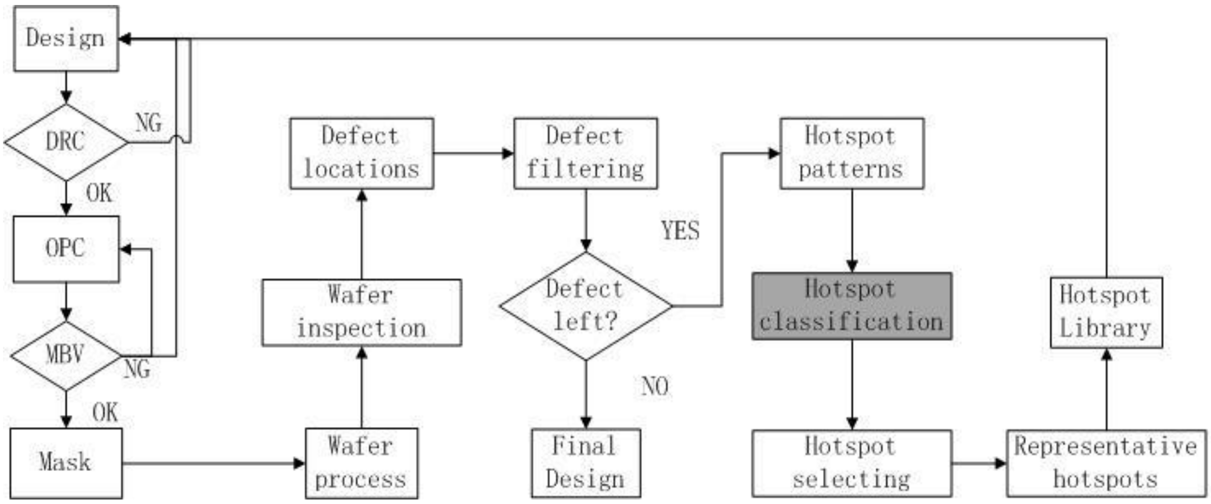

Fig. 1 Mask manufacturing flow

2. HOTSPOT PATTERN SIMILARITY AND DISCRETE 2-D CORRELATION

As described in Section 1, the hotspot patterns are clipped from the full-chip layout with a fixed size of $2R$ x $2R$, where R denotes the influencing distance of optical model kernels. An IC layout is binary, even with attenuated phase-shifting mask (PSM), therefore without losing generality we set the value of pixels covered by polygons to 1, and the value of pixels in the space to 0. If we overlay two binary images of hotspot pattern, the larger area of overlap indicates the higher similarity between them [3, 4]. Fig. 2 shows an example of pattern overlap.

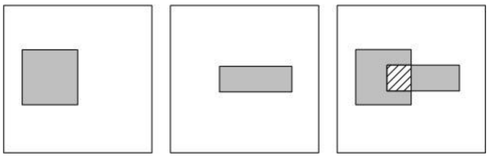

Fig. 2 An example of pattern overlap

We thus define the similarity between two patterns P_1 and P_2 as following,

$$similarity(P_1, P_2) = \min\left(\frac{overlap(P_1, P_2)}{area(P_1)}, \frac{overlap(P_1, P_2)}{area(P_2)}\right) \tag{1}$$

Overlap(P_1,P_2) denotes the overlapped polygon area of pattern P_1 and P_2, area(P_1) and area(P_2) indicate the total polygon areas of pattern P_1 and P_2 respectively.

In hotspot classification process, the rotation or reflection of a pattern should be considered to be identical to the original one, as Fig. 3 shows [3, 4].

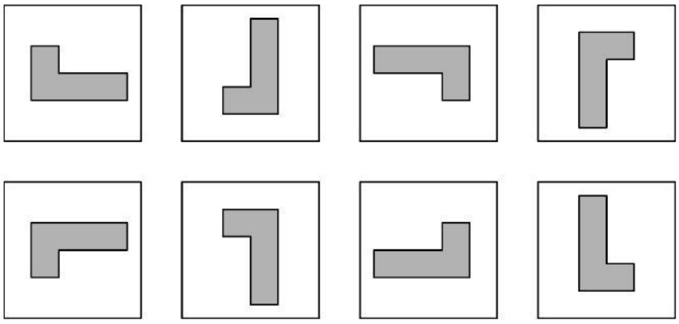

Fig. 3. The eight transformations of a pattern

A hotspot pattern clipped from the full-chip layout is centered on the defect location found in wafer inspection stage. Unfortunately, the defect location is not accurate enough usually. In other words, the hotspot pattern is very possibly center-shifted. As a result, the overlap metric cannot precisely indicate the similarity between the two patterns if we overlay such patterns. Fig. 4 shows an example of two similar yet center-shifted patterns. There is no overlap between the two patterns when we overlay them directly, although an almost exactly matching position can be found when we keep moving pattern B over pattern A, where the similarity between pattern A and B can approach 100%, actually.

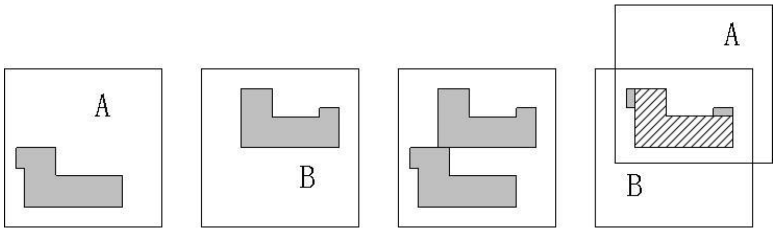

Fig. 4 An example of pattern overlay under center-shifting condition

We calculate the discrete 2-d correlation to get the maximal overlap between two patterns. Suppose f(x,y) and g(x,y) are binary images of size M_1xN_1 and M_2xN_2. The size of discrete 2-d correlation f(x,y)*g(x,y) would be (M_1+M_2-1)x(N_1+N_2-1). We can extend the images by padding with zeroes,

$$f_e(x,y)=\begin{cases} f(x,y) & 0\le x\le M_1-1,\ 0\le y\le N_1-1 \\ 0 & M_1\le x\le M_1+M_2-2 \ or\ N_1\le y\le N_1+N_2-2 \end{cases} \tag{2}$$

$$g_e(x,y)=\begin{cases} g(x,y) & 0\le x\le M_2-1,\ 0\le y\le N_2-1 \\ 0 & M_2\le x\le M_1+M_2-2 \ or\ N_2\le y\le N_1+N_2-2 \end{cases} \tag{3}$$

The correlation can be defined as

$$f(x,y)\bullet g(x,y)=\frac{1}{(M_1+M_2-1)(N_1+N_2-1)}\sum_{m=0}^{M_1+M_2-2}\sum_{n=0}^{N_1+N_2-2} f_e(m,n)g_e(x+m,y+n) \tag{4}$$

$$(x=0,1,...,M_1+M_2-2,\ y=0,1,...,N_1+N_2-2)$$

According to correlation theorem, it can be calculated by

$$f(x,y)\bullet g(x,y)=F^{-1}\{F[f_e(x,y)]\cdot F^*[g_e(x,y)]\} \tag{5}$$

where ‘*’ denotes the conjugate operation.

Let matrices P_1 and P_2 indicate two MxN pixelated images of hotspot patterns. The discrete 2-d correlation of P_1 and P_2 is a (2M-1)x(2N-1) matrix. The maximal value of element in the discrete 2-d correlation matrix indicates the maximal overlap between P_1 and P_2 when we move one pattern all over another. With the maximal overlap value, the similarity can be calculated by formula (1).

3. CLUSTERING ALGORITHM

Now that pattern similarity has been defined in Section 2, we need a clustering algorithm to group the hotspot patterns with certain similarity. Partitioning methods and hierarchical methods are two major types of clustering algorithms. In partitioning methods, the number of clusters should be determined before clustering process, while hierarchical methods can divide the dataset into any number of clusters. Hierarchical methods are suitable for hotspot classification because the number of clusters is uncertain before clustering, and they usually require the computation of a pair-wise distance matrix indicating the distance correlations between any two objects in the dataset. When the dataset is large, they are time-consuming. In this paper, we propose a heuristic sequential clustering algorithm, in which hotspot patterns are read from the dataset sequentially and inserted into evolving clusters. The pseudo code of this algorithm is listed below.

{*CLUSTERS* represents the clustering result}
{*CLUSTER_METRIC* is the representative pattern of this cluster, it is the first member of this cluster here}
{*Threshold* (or *Threshold_dist*) is defined by user, larger *Threshold* (or smaller *Threshold_dist*) leads to tighter and more clusters}

for each hotspot pattern *f* in the dataset **do**
    if *CLUSTERS* is empty
        make a new cluster and take this pattern *f* as the *CLUSTER_METRIC*
    else
        for each cluster in *CLUSTERS* **do**
            calculate the pattern similarity (or distance) between *f* and *CLUSTER_METRIC*
            if the pattern similarity > *Threshold* (or distance < *Threshold_dist*)
                add the pattern *f* to this cluster
                break
            end if
        end for
        if the pattern *f* cannot be added to all of the existing clusters
            make a new cluster and take this pattern *f* as the *CLUSTER_METRIC*
        end if
    end if
end for

The time complexity of the proposed algorithm is O(mn), while the one of hierarchical method is $O(n^2)$, where m is the number of resulted clusters, n is the size of hotspot dataset.

4. HIGHER-ORDER LOCAL AUTOCORRELATION BASED HOTSPOT CLASSIFICATION

HLAC has been widely used in facial recognition [5,6], texture classification [7] and motion recognition [8], etc. With similarity defined by formula (1), discrete 2-d correlation and clustering algorithm described in Section 3, the classification is time-consuming when a hotspot database is large and shows diversity. For higher classification performance, we propose the HLAC based hotspot classification method, in which HLAC is used in hotspot patterns feature extraction and hotspot pre-clustering followed by detail clustering using discrete 2-d correlation.

4.1 HLAC based Feature Extraction

The *N*th-order autocorrelation functions, extensions of autocorrelation functions, are defined as

$$x(a_1, a_2, ..., a_N) = \int I(r)I(r + a_1)...I(r + a_N)dr \tag{6}$$

where I(r) denotes the intensity at the observing pixel r, and a_1, a_2, …, a_N are N displacements. For binary images, the N*th*-order autocorrelation function can be regarded as counting the number of pixels which satisfy formula (7)

$$I(r) \wedge I(r + a_1) \wedge ... \wedge I(r + a_N) = 1 \tag{7}$$

where "Λ" denotes logical operation "AND".

In other words, the N*th*-order autocorrelation function can be regarded as counting the patterns characterized by the above logical statement over the binary image [9].

The orders and displacements in Eq. (6) are arbitrary. However, high-order features with a large range of displacement region become extremely numerous. Hence, the original HLAC features are restricted up to the second order and within a displacement range of 3×3 window. Fig. 5 [10] shows 25 mask patterns with 0, 1 and 2 displacements respectively.

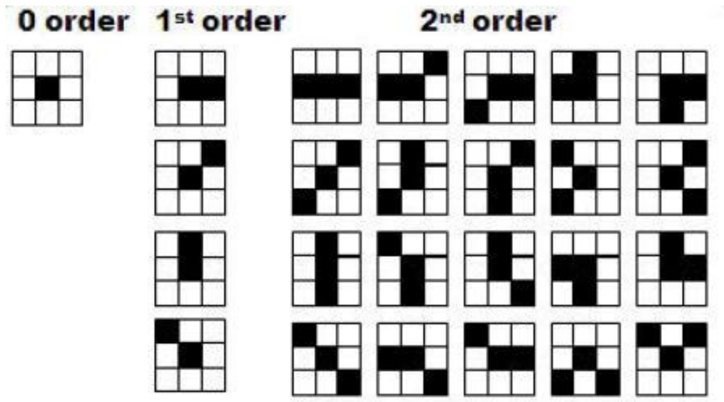

Fig. 5 25 mask patterns of the HLAC features (3x3)

The HLAC feature values of a image are calculated by scanning the image with the 25 mask patterns in Fig. 5 and computing the sums of products of the intensities of corresponding pixels. Each value represents the power spectrum of the mask pattern. Therefore, the 25 mask patterns are regarded as the basis functions of frequency analysis [11].

It is well known that HLAC features are shift invariant. Hence, it is appropriate for the center-shifted hotspot pattern feature extraction. The features of a 3×3 displacement region mainly extract the local detailed information. Hotspot pattern image is binary, and most of the local subtlety information distributes on the polygon boundary. Therefore, after pixelate, we extract the polygon boundary of the hotspot pattern image to form the boundary image. Then each mask pattern in Fig.5 is scanned over the entire boundary image to calculate HLAC feature value *f*. For each hotspot pattern image, the operation is performed using 25 different mask patterns to create the feature vector ($f_1, f_2, ..., f_{25}$) in which f_1 denotes the total length of polygon boundary. With the total polygon area of a hotspot pattern as another feature f_0, combining with the 25 dimensional HLAC based feature vector ($f_1, f_2, ..., f_{25}$), we create a 26 dimensional feature vector ($f_0, f_1, f_2, ..., f_{25}$) for each hotspot pattern.

4.2 Dimension Reduction with Principal Component Analysis

Principal component analysis (PCA) is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components [12]. The 26 features of the feature vector described in Section 4.1 are correlated. Hence, we adopt PCA to reduce the vector dimension for higher pattern clustering efficiency.

Fig. 6 shows the dimension reduction flow. After feature extraction, with f_0 as a metric for convenience, a wide range subset of hotspot pattern feature vectors picked out of the hotspot pattern database is fed to PCA to find out the principal components. A low dimensional vector space is constructed with the several most important principal components with highest weights as bases. All of the hotspot pattern feature vectors are projected to the new low dimensional vector space. Finally, the simplified low dimensional feature vectors of the hotspot patterns are used in the clustering step.

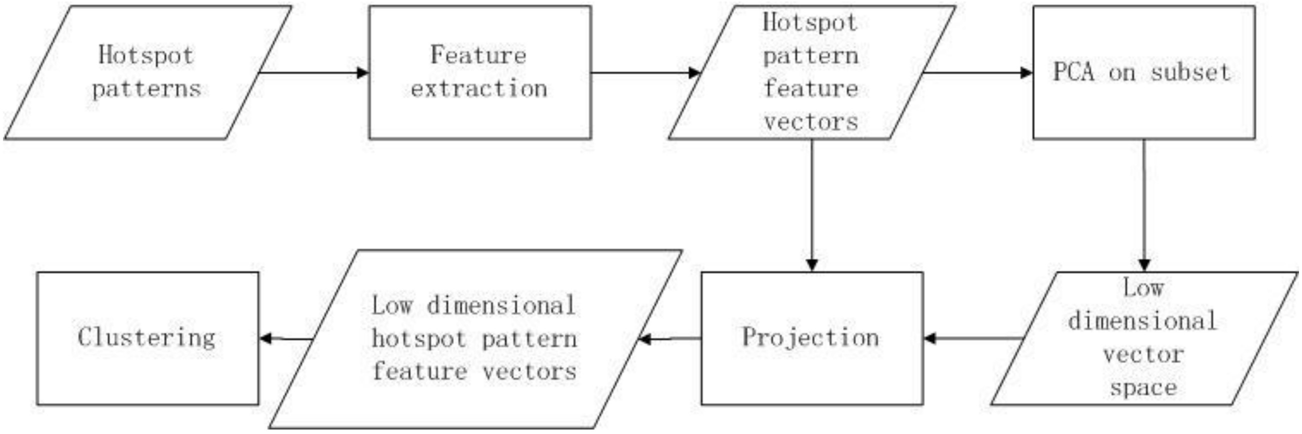

Fig. 6 Dimension reduction flow

4.3 Pattern distance defined in the low dimensional vector space

PCA is defined in such a way that the first principal component has the largest possible variance, and each succeeding component in turn has the highest variance possible under the constraint that it be orthogonal to the preceding components. Let $(x_1,x_2,\ldots,x_n)$ and $(y_1,y_2,\ldots,y_n)$ denote the coordinates in the low dimensional feature vector space of two hotspot patterns P_1 and P_2. The distance between P_1 and P_2 can be defined as

$$\text{distance} = \sum_{i=1}^{n} \lambda_i \left| x_i - y_i \right| \tag{8}$$

here λ_i denotes the weight of *i*th most important principal component, and $\lambda_1 > \lambda_2 >, \ldots > \lambda_n$. $n \le 26$, usually $n = 4$ is accurate enough. The shorter distance means the higher similarity.

4.4 Hotspot clustering

When the hotspot patterns to be classified are diverse and of large scale, clustering using similarity defined in Section 2 is time-consuming. On the other hand, clustering with HLAC features only is not effective enough, because the subset used in PCA cannot cover the whole database and the low dimensional feature vectors of two different hotspot patterns can become similar, since the feature extraction method based on HLAC described in Section 4.1 cannot guarantee that the relationship between hotspot pattern and HLAC based feature vector is a one-to-one mapping, owing to the fact that the local detailed information of binary hotspot pattern image is limited. Therefore, in the clustering step, we combine the two methods together to get higher clustering performance and meaningful clustering result.

Fig. 7 shows the clustering flow. Firstly, we pre-cluster the hotspot patterns using HLAC features and pattern distance defined by formula (8). Then for each subset, we group them into sub-clusters using discrete 2-d correlation and similarity defined by formula (1). Finally, we pick out a representative for each sub-cluster, classify the representatives and put each sub-cluster into the final cluster to which its representative belongs.

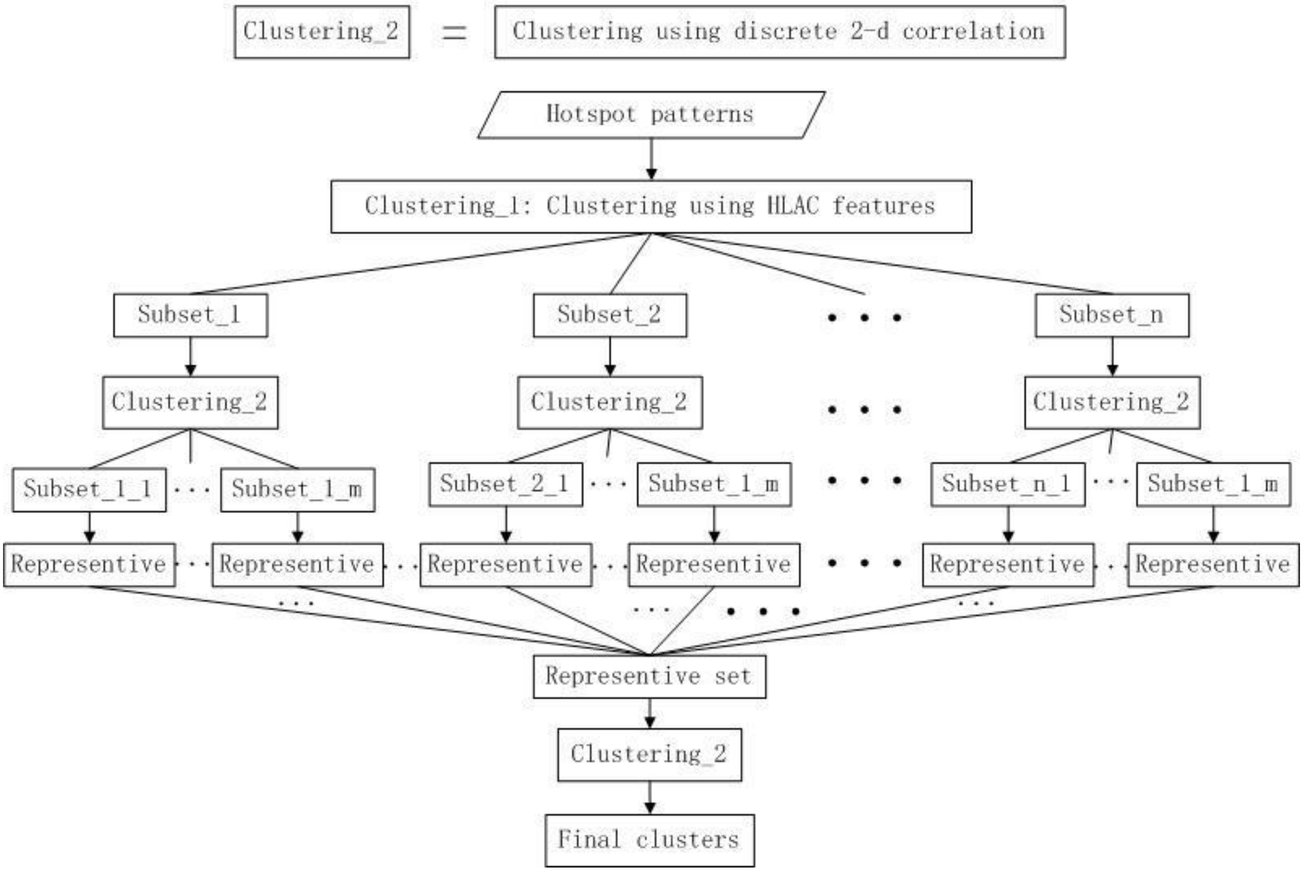

Fig. 7 HLAC based clustering

5. EXPERIMENTS AND RESULTS

Two datasets containing 955 and 3885 center-shifted hotspot patterns with maximal center-shifting distance up to 300nm respectively were used in the experiments. The hotspot patterns with size of 1um x 1um were extracted from an industrial 45nm M1 layer layout with an area of 3mm x 3mm. The grid used in pixelate is 20nm. The similarity thresholds used in clustering are 75% and 60%. All computations were performed on a Dell PowerEdge R610 workstation (Xeon 2.8 GHzx24 CPUs and 32GB memory).

We used three methods for classification here. Method-1 and Method-2 cluster hotspots using similarity defined by formula (1) in Section 2. To calculate overlap, Method-1 overlays two patterns directly, while Method-2 uses discrete 2-d correlation. Method-3 is the proposed HLAC based method described in Section 4.4. In Method-3, we performed PCA on 100 hotspot patterns to construct the low dimensional vector space, which takes only 1.2 seconds. We set the weights of four most important principal components according to the eigenvalues generated by PCA, that is, $\lambda_1 = 0.47, \lambda_2 = 0.2, \lambda_3 = 0.15, \lambda_4 = 0.08, Threshold\_dist = 0.565$.

Table 1 compares the run time, the result cluster number and accuracies of the three methods. Min-SIG denotes the minimum value of similarity between two hotspot patterns in the same groups of classification result. Max-SBG indicates the maximal value of similarity between two hotspot patterns in different groups of classification result. Fig. 8 and Fig. 9 show the comparison of sizes of final clusters when Threshold = 0.75. Fig. 10 and Fig. 11 show the comparison of sizes of final clusters when Threshold = 0.6. Larger *Threshold* leads to tighter and more clusters.

From the results, we can see that the final cluster number and Max-SBG of Method-1 are much larger than those of Method-2 and Method-3, because Method-1 cannot cluster the similar hotspot patterns into the same cluster under center-shifting condition while Method-2 and Method-3 using discrete 2-d correlation can handle this problem effectively. Method-3 is much faster than Method-2.

Table 1. Run time and classification accuracy

Threshold = 0.75						
	Dataset 1 (955 hotspot patterns)			Dataset 2 (3885 hotspot patterns)		
	Method-1	Method-2	Method-3	Method-1	Method-2	Method-3
Run time	181s	733s	193s	721s	1733s	478
Cluster number	354	82	76	545	90	85
Min-SIG	73.5%	72.7%	73.1%	74%	73.5%	73.2%
Max-SBG	95.6%	68.9%	70.2%	94.7%	66.7	65.9%
Threshold = 0.6						
	Dataset 1 (955 hotspot patterns)			Dataset 2 (3885 hotspot patterns)		
	Method-1	Method-2	Method-3	Method-1	Method-2	Method-3
Run time	82s	244s	117s	296	845s	309s
Cluster number	173	44	44	231	46	44
Min-SIG	58.1%	57.6%	57.5%	58.5%	58%	57.8%
Max-SBG	95.6%	54.3%	55.1%	94.7%	53.9%	54%

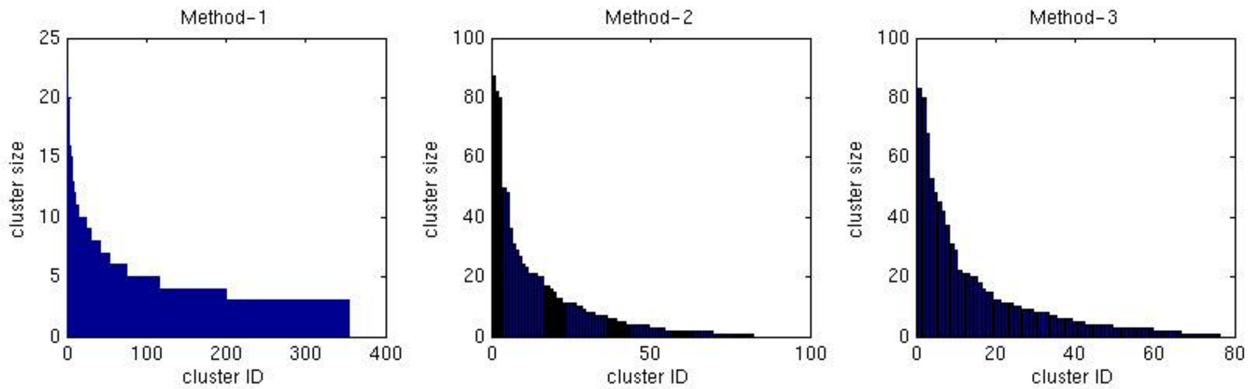

Fig. 8 The comparison of sizes of final clusters for dataset 1 when Threshold = 0.75

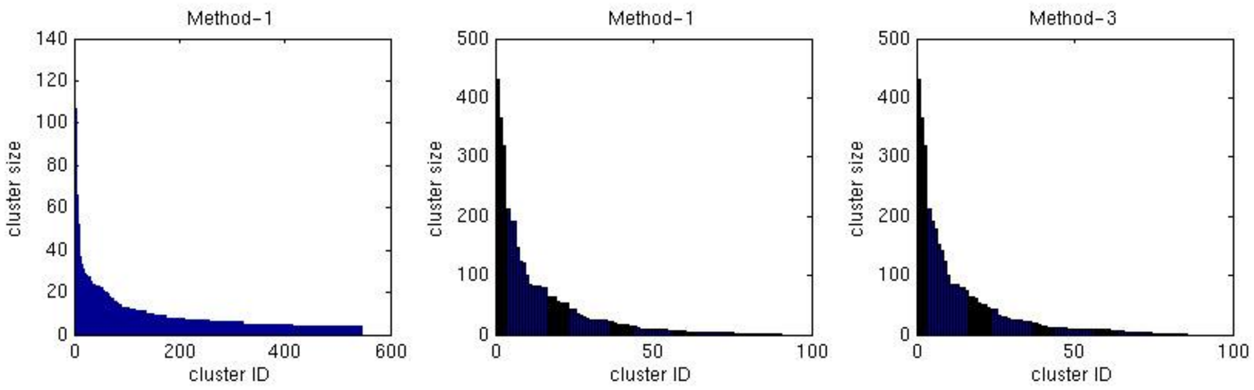

Fig. 9 The comparison of sizes of final clusters for dataset 2 when Threshold = 0.75

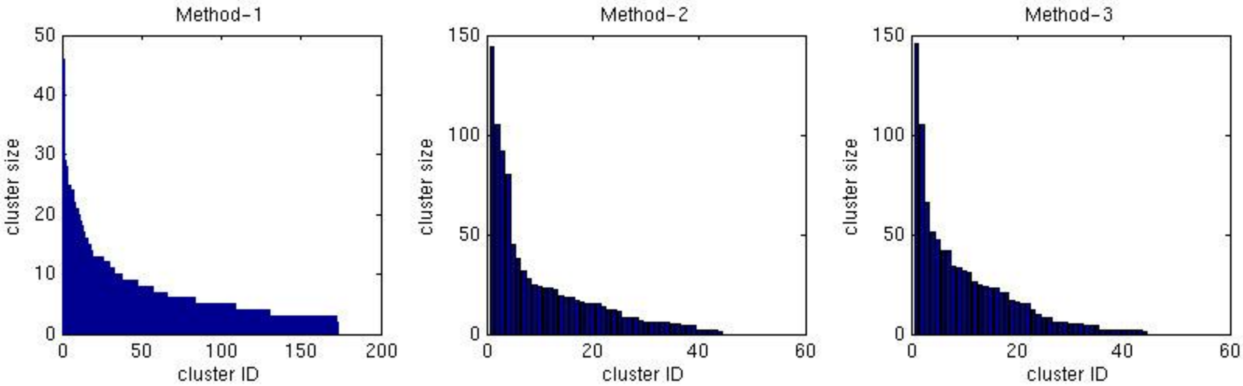

Fig. 10 The comparison of sizes of final clusters for dataset 1 when Threshold = 0.6

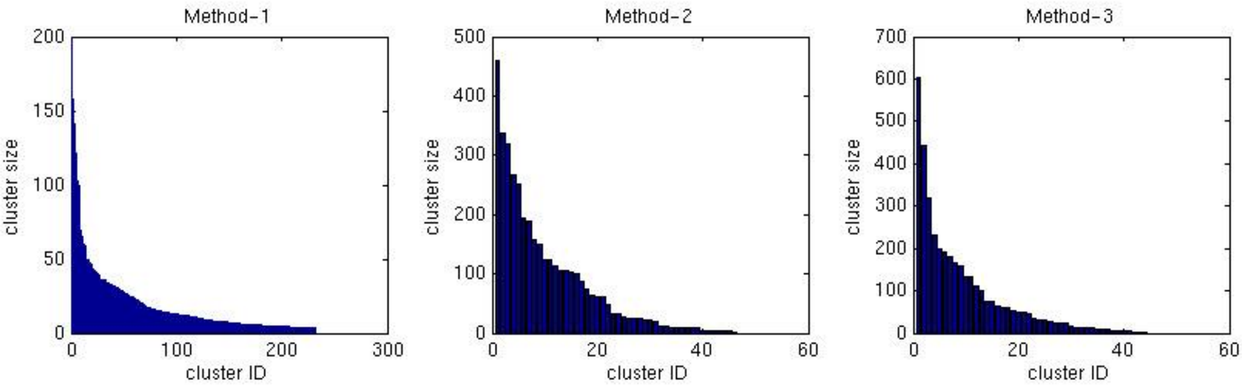

Fig. 11 The comparison of sizes of final clusters for dataset 2 when Threshold = 0.6

Fig. 12 and Fig. 13 show some sample patterns of clustered hotspots of dataset1 and dataset2 using the proposed HLAC based method. Through observing these samples, we can sense that the similar center-shifted hotspots have been clustered into the same group.

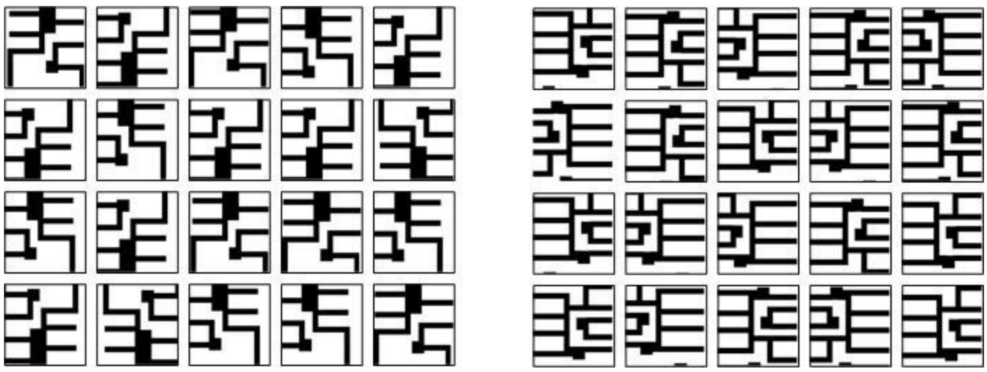

Fig. 12 Two examples of clustered hotspots of dataset1

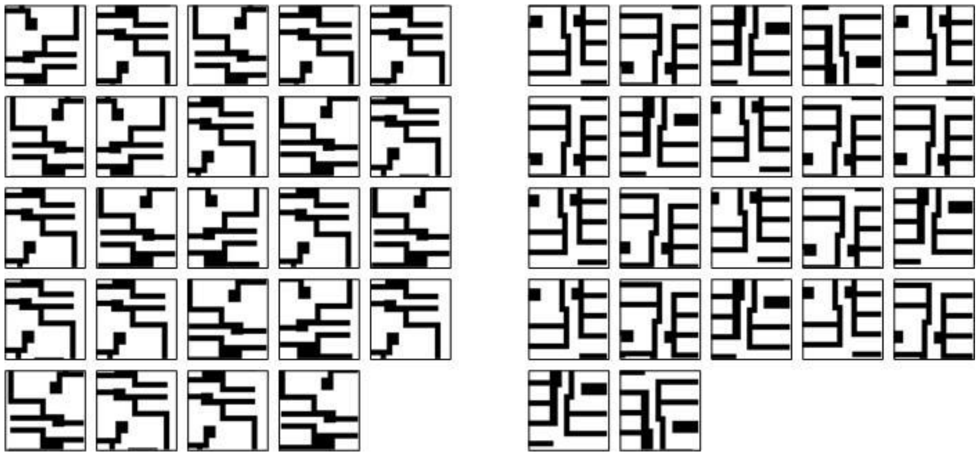

Fig. 13 Two examples of clustered hotspots of dataset2

6. CONCLUSIONS

This paper proposes a hotspot classification method based on HLAC. The key innovative part of this method is the pre-clustering step with feature extraction based on HLAC before detailed clustering using pattern similarity. Experiments show that the proposed method speeds up the classification process greatly.

REFERENCES

[1] Lee, T. et al., "Hot Spot Management through Design Based Metrology-Measurement and Filtering-", Proc. of SPIE vol. 7520, 75201U, (2009)

[2] Yoo, G. et al., "OPC Verification and Hot Spot Management for Yield Enhancement through Layout Analysis", Proc. of SPIE vol. 7971, 79710H, (2011)

[3] Ma, N. et al., "Automatic hotspot classification using pattern-based clustering", Proc. of SPIE vol. 6925, 692505, (2008)

[4] Ghan, J. et al., "Clustering and pattern matching for an automatic hotspot classification and detection system", Proc. of SPIE vol. 7275, 727516, (2009)

[5] Lajevardi, S.M., and Hussain, Z.M., "Novel higher-order local autocorrelation-like feature extraction methodology for facial expression recognition", IET Image Process, Vol. 4, Iss. 2, pp. 114-119, (2010)

[6] Nomoto, N. et al., "A New Scheme for Image Recognition Using Higher-Order Local Autocorrelation and Factor Analysis", Proc. of MVA2005, pp. 265-268, (2005)

[7] Toyoda, T. and Hasegawa, O., "Texture Classification Using Extended Higher Order Local Autocorrelation Features", Proc. of the 4th International Workshop on Texture Analysis and synthesis, pp. 131-136, (2005)

[8] Watanabe, K. and Kurita, T., "Motion Recognition by Higher Order Local Auto Correlation Features of Motion History Images", Proc. of BLISS `08, pp. 51-55, (2008)

[9] Otsu, N. and Kurita, T., "A new scheme for practical flexible and intelligent vision systems", Proc. of IAPR Workshop on computer Vision, pp.431-435, (1988)

[10] Maeda, S. et al., "Hotspot detection using image pattern recognition based on higher-order local auto-correlation", Proc. of SPIE vol. 7974, 79740X, (2011)

[11] Takahiro, T. et al., "Extension of higher order local autocorrelation features", Journal of the Institute of Image Electronics Engineers of Japan, Vol. 34, No. 4, pp. 390-397, (2005)

[12] http://www.wikipedia.org/