

# Automated Detection and Localization of Counterfeit Chip Defects by Texture Analysis in Infrared (IR) Domain

Pallabi Ghosh<sup>1</sup>, Ulbert J Botero<sup>1</sup>, Fatemeh Ganji<sup>1</sup>, Damon Woodard<sup>1</sup>, Rajat Subhra Chakraborty<sup>2</sup>, Domenic Forte<sup>1</sup>  
{pallabighosh, jbot2016, fganji}@ufl.edu, dwoodard@ece.ufl.edu, rschakraborty@cse.iitkgp.ac.in, dforte@ece.ufl.edu

**Abstract**—Today’s globalized supply chain for electronics design, fabrication, and distribution has resulted in a proliferation of counterfeit chips. Recycled and remarked chips are the most common counterfeit types in the market, and prior work has shown that physical inspection is the best approach to detect them. However, it can be time-consuming, expensive, and destructive while relying on the use of subject matter experts. This paper proposes a low-cost, automated detection technique that examines surface variations within and between chips to identify defective chips. Further, it can estimate the location of the defects for additional analysis. The proposed method only requires a cheap IR camera-based setup to capture images of the chip package surface and is completely unsupervised and non-destructive. Experimental results on 25 chips in our lab demonstrate 100% detection accuracy.

## I. INTRODUCTION

Counterfeit integrated circuits (ICs) are a serious threat to critical systems. There are several categories of counterfeit ICs, among which recycled and remarked ICs are the most prevalent found in the market [1]. Recycled and remarked ICs are removed from scrapped printed circuit boards (PCBs) and sold as new, often after their packages are sanded, recoated, and/or remarked. These recycled ICs are prone to failure and thus decrease system reliability and lifetime. If they are incorporated in safety and mission-critical systems like aircrafts, submarines, etc. they may cause catastrophic hazards and loss of lives. Therefore, identification of such ICs before they are integrated into systems is dire.

Fortunately, various defects commonly found on the package surface of counterfeit ICs [2], [3], [4], can be determined, including scratches, differences in indent size and position, sanding/ grinding marks, ghost markings, burned markings, mold variation, package damage, corrosion/ contamination, and extraneous markings. Currently, most of these defects are detected manually by highly skilled subject matter experts, making the process both costly and time-consuming.

In this paper, we propose an automated, non-destructive counterfeit-IC detection approach that leverages the texture information from known authentic chips to achieve high accuracy in identifying such defects. A stand-out feature of our proposed technique compared to the state-of-the-art techniques is the use of infrared (IR) images in place of optical images. The main advantage of IR images is that they minimize the reflections which often result in false positives. Besides, it has the ability to highlight features generally invisible to the human eye, e.g., remarked surface, scratches, ghost marking, etc. Moreover, while prior work employed image texture analysis on optical images using Local Binary

Pattern (LBP)-type features to detect counterfeit ICs [5], [7], they depend heavily on the availability of texture information of known authentic (i.e., “golden”) ICs. Some of our approaches only require prior knowledge of a golden texture which can come from any chip. Further, another shortcoming of methods proposed before in the literature is that LBP is sensitive to noise and sometimes wrongly estimates near-flat regions.

To tackle this, in this paper, we propose a block-wise two-fold feature extraction method. For this, firstly, each image is divided into multiple non-overlapping blocks. Afterward, from each of these blocks, Local Gabor Directional Pattern (LGDIP) features are extracted. LGDiP is known to be much more effective than LBP as it extracts features in two phases. LBP extracts only the fine details while LGDiP focuses on the appearance information over a broader range of scale as well as minute details for each of that scale. It also uses Local Directional Pattern (LDP) in the second phase instead of LBP, which is more advanced and does not suffer from the shortcomings of LBP [6]. After feature extraction, another set of inter- and intra-chip distance features are computed using the extracted features. This step helps in examining whether the patterns are uniform among two different sets of chips. While the inter-distance helps in quantifying the non-uniformity among the same blocks between the ICs, intra-chip distance helps in quantifying that within the same chip. A combined distance vector is then used as a feature vector to observe the sudden spike in values. The position of the sudden spike is then referred to as the defective block, and the IC with that block is probably a counterfeit IC. We demonstrate the efficacy of our approaches using 25 chips of four different types, with two genuine classes and two defective classes. In a nutshell, the main contribution of our paper is that our approach is capable of automatically detecting and identifying the location of the defects with high accuracy.

The rest of this paper is organized as follows. In Section II, the necessary background on image texture analysis and the LGDiP algorithm is provided. In Section III, the proposed methodology for counterfeit IC detection is described in detail, along with the relevant automated analysis flow. The experimental setup, results, and discussions are presented in Section IV. This paper is concluded with directions for future research in Section V.

## II. BACKGROUND

A texture can be defined as the spatial arrangement of the intensity values in an image. State-of-the-art texture analysis techniques can be classified into three broad categories [6]:

- Convolutional Neural Network-based: In recent years, CNN based techniques for texture recognition have

<sup>1</sup> Electrical and Computer Engineering, University of Florida, Gainesville, 32611, USA

<sup>2</sup> Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur, West Bengal, 721302, India

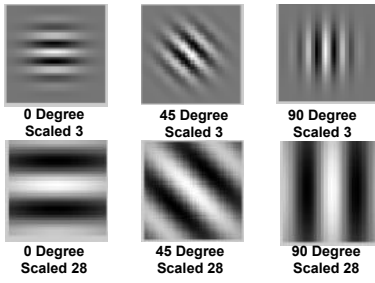


Fig. 1. Gabor Wavelet Example

been growing in popularity. However, they require a relatively large labeled dataset. Building such a dataset for counterfeit IC detection is challenging since it is difficult to obtain known counterfeit ICs, and there are many different types of defects.

- **Attribute-based:** The main aim of this relatively new technique is to provide a comprehensive understanding of textures for objects that are characterized by patterns. It matches the pattern present on an object, e.g., striped or banded, with a known pattern set. To this end, there is a need for universal texture attribute vocabulary, which contains all types of such pattern sets. Unfortunately, for ICs, this has not been accomplished yet.
- **Bag of Words:** In this technique, an image is converted into a texture image, where each location is referred to as a “texel”. Each texel contains a set of local features extracted by analyzing the local pixel neighborhood. Thus, the texture image formed through this has a pool of local features. These features are then combined into a global representation like histograms. We adopt this technique since it works well for smaller datasets and does not require any universal feature vocabulary.

More concretely, our feature extraction method, i.e., LGDiP, combines Gabor filter and LDP image processing techniques, as described below.

#### A. Gabor Filter

The Gabor filter is a linear filter used to detect the presence of certain frequencies in an image. A two-dimensional (2D) Gabor filter is a sinusoidal wave modulated by a Gaussian model [12]. When working with Gabor filters, it is common to work with the magnitude response of each filter, sometimes referred to as “Gabor Energy”. A bag of Gabor wavelets are generated by varying the orientation and scaling factors present in the Gabor function [13]:

$$\phi(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\left(\frac{x^2}{2\sigma_x^2} + \frac{y^2}{2\sigma_y^2}\right)\right] \exp(j2\pi\omega), \quad (1)$$

where  $\phi(x, y)$  is the Gabor function,  $\sigma_x$  and  $\sigma_y$  represent the standard deviations of the Gaussian model along  $x$  and  $y$  dimensions, respectively, and  $\omega$  is the radial center frequency of the filter in the frequency domain. A sample bag of filters computed using  $\phi(x, y)$  for different scales and orientations is shown in Figure 1.

#### B. Local Directional Pattern (LDP)

Local Directional Pattern (LDP) computes the edge response values in different directions and uses that to create

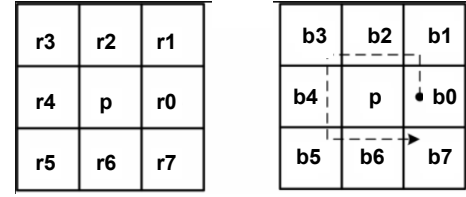


Fig. 2. Local Directional Pattern Examples.

a texture image [14]. Each edge detector algorithm employs a set of masks for different directions. Using any such set of masks  $m_i$  the directional responses surrounding a particular pixel is computed. The responses are represented as  $r_i$ . Then based on these responses and Eqn. 2, a bit pattern is generated.

$$LDP_k = \sum_{i=0}^7 b_i(r_i - r_k) \cdot 2^i, \quad (2)$$

with  $b_i(\cdot)$  defined as follows.

$$b_i(a) = \begin{cases} 1 & a \geq 0, \\ 0 & a < 0 \end{cases} \quad (3)$$

For instance, suppose that there are eight masks in an edge detector for eight directions. Applying these masks on the neighborhood of the pixel  $p$ , eight different responses are obtained, as shown in Figure 2. Then, the top  $k$  values or any value greater than certain threshold  $a$  are assigned 1 and rest 0. These bits are represented as  $b_i$  in Figure 2.

Following Eqns. 2 and 3 and taking a definite direction for the entire image into account, an eight bit LDP pattern is obtained [14]. The decimal value for each of these bit patterns represents the texel value for the corresponding pixel location.

#### C. Local Gabor Directional Pattern (LGDiP)

Gabor filters encode appearance information over a broader range of scales while LDP captures small and fine details. Hence, the combination of these two gives a much more accurate representation of the texture [15], [16]. ICs with and without defects have similar texture, so the feature extraction algorithm should be accurate and should not be sensitive to noise. LGDiP addresses both requirements and unlike CNN-based methods, can be implemented on a small dataset.

For LGDiP, an image is first divided into multiple non-overlapping blocks. For each block, the following steps are performed:

- 1) Gabor filter at  $R$  orientations and  $S$  scales is applied. This results in  $S \times R$  images.
- 2) LDP is applied on each  $S \times R$  image to form histograms.
- 3) All histograms are then concatenated together to form a single histogram having  $d$  features (texel values).

In our work, for the entire image, all  $d$  feature histograms for each of the blocks are concatenated together in row-major order to obtain a single feature vector representing each IC image.

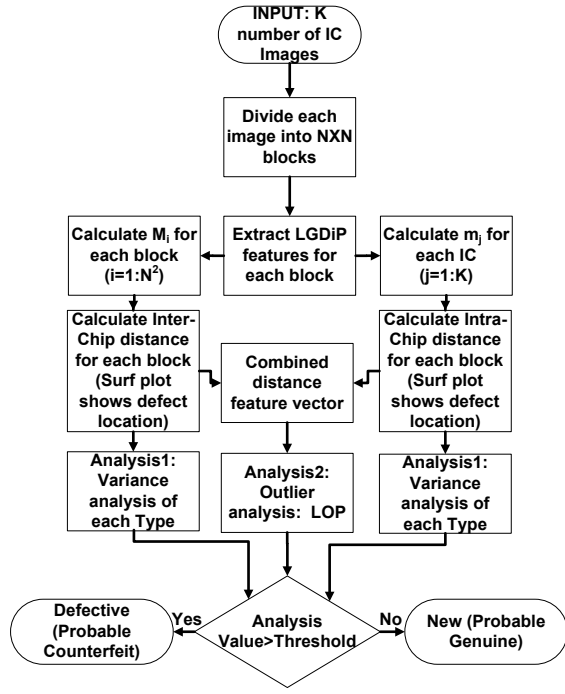


Fig. 3. Flowchart of defect detection methodology.

### III. METHODOLOGY

The overall methodology is shown in Figure 3. It performs a two-fold feature extraction. In the first step, it follows the LGDiP technique discussed in Section II. In our technique, each IC image is divided into  $N \times N$  blocks. For each block,  $d$  LGDiP features are extracted. Therefore, each image has  $d \times N \times N$  features. In our second step, it computes two more sets of features from the LGDiP features extracted in the first step. They are inter-chip and intra-chip distances.

Our approach is most effective under the following assumptions:

- Authentic ICs belonging to the same lot are similar and their texture is to a high extent uniform.
- Defects are not identical in terms of size, location, and/or degree on all counterfeit ICs.
- For inter-chip distance, the relative number of the golden IC samples should be equal or greater than the counterfeit ICs. Alternatively, test data should be compared with a fewer number of known golden IC data in batches, such that the ratio is maintained.

#### A. Inter-Chip Distance Calculation

The inter-chip distance for each block captures the variability among blocks belonging to the same location in all the ICs. This variability measurement is not specific to any particular type of chip and thus helps in finding a common feature that can describe patterns in *any* type of IC. This follows from the assumption that all chips, especially those from the same lot, should have similar texture in the same areas of the chip package.

To compute the inter-chip distance, first, the mean of all the blocks at the same position is computed. That is, all the  $N \times N$  blocks are numbered in a row-major order such that for each chip, there are  $N^2$  block numbers as shown in Figure 4. The mean  $M_i$  of the features of each block  $i$  of all

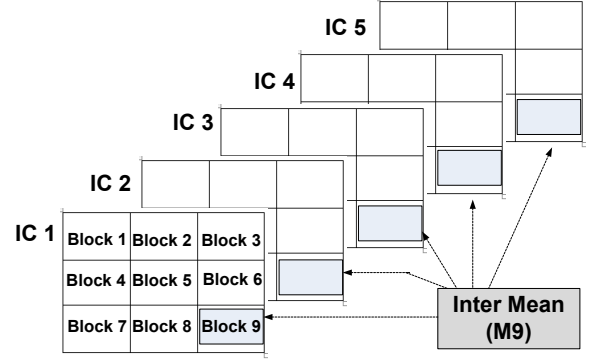


Fig. 4. Inter-chip mean vector calculation for block 9 ( $M_9$ ) for  $N = 3$  and  $K = 5$ . The Euclidean distance between this mean and block 9 of  $IC_k$  is the inter-chip distance of block 9 for  $IC_k$ .

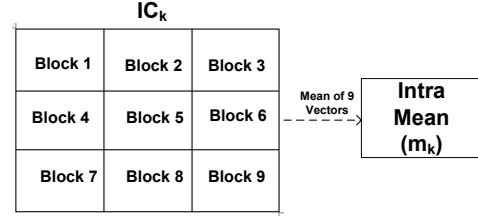


Fig. 5. Intra-chip mean vector ( $m_k$ ) calculation for  $IC_k$  where  $N = 3$ . For each IC, there is one intra-chip mean. Intra-chip distance of block  $n$  is the Euclidean distance between  $m_k$  and LGDiP feature vector of block  $n$ .

the ICs are computed. Following this technique,  $N^2$  numbers of mean vectors (one per block) are obtained. For each IC, the inter-chip distance for every block is computed as the Euclidean distance between the block and its corresponding mean vector. For instance, for  $K$  ICs, a distance matrix of size  $K \times N^2$  is obtained, where each row corresponds to inter-chip distance features for one IC.

#### B. Intra-Chip Distance Calculation

Unlike inter-chip distance, the intra-chip distance calculates the variability within a single chip. As stated in the list of our assumptions, it is assumed that the texture within an authentic chip is more uniform than in a counterfeit chip. This is often the case since sanding, recoating, and remarking are only performed on a portion of the package surface.

Here, the mean vector  $m_k$  is the average of the feature vectors of all the blocks belonging to the  $k$ th IC. Thus, in intra-chip distance calculation, the Euclidean distance of the LGDiP feature vectors of each block of an IC is calculated from the same mean for a particular IC. For  $K$  ICs, there are  $K$  mean vectors. The scheme to calculate the mean has been diagrammatically depicted in Figure 5.

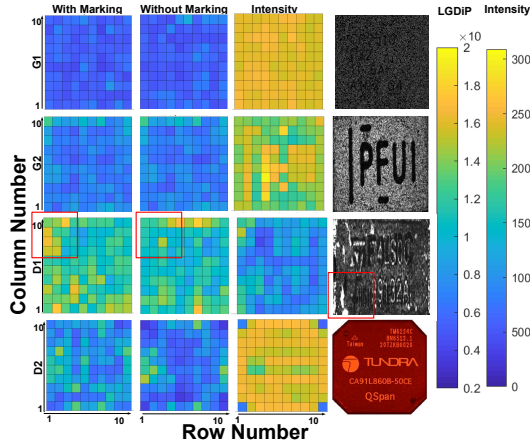
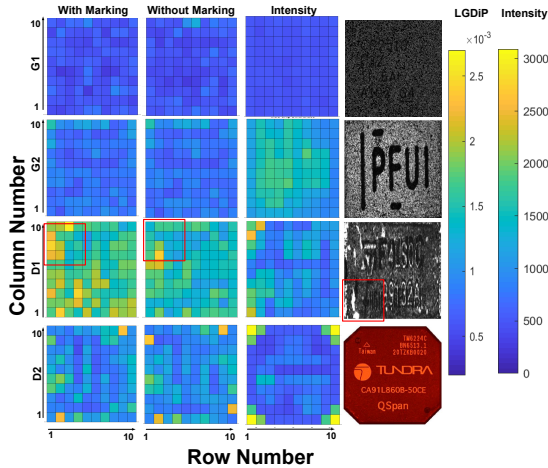
#### C. Analysis of Distances to Localize Defects

1) *Variance*: For golden ICs, it is assumed that variation among all the inter- and intra-distance values should be lower than counterfeits. This is because, for a golden IC, the texture pattern should be virtually similar over the entire surface of the IC. Therefore, the intra- and inter differences with the mean should be almost the same. So the maximum variance observed among any given golden sample set can be considered as the threshold.

TABLE I

SAMPLE OPTICAL AND IR IMAGES OF EACH CHIP TYPE IN OUR DATASET.

| IC Type | Optical Image | IR Image | Dimension (Approx)   |
|---------|---------------|----------|--|
| G1      |               |          | Real Size: 10.11 mmx10.11 mm<br>Actual IR image Pixel Dimension: 233X233 |
| G2      |               |          | Real Size: 5.32 mmx2.4 mm<br>Actual IR image Pixel Dimension: 509X230    |
| D1      |               |          | Real Size: 32 mmx8.9 mm<br>Actual IR image Pixel Dimension: 1800X500     |
| D2      |               |          | Real Size: 26 mmx26 mm<br>Actual IR image Pixel Dimension: 1580X1580     |

Fig. 6. Sample surf plot showing inter-chip distance variability with and without marking for ICs of type G1, G2, D1 and D2 ( $N = 10$ ). A sample red bounding box, indicating location of defect, is also shown.Fig. 7. Sample surface plot showing intra-chip distance variability with and without marking for ICs of type G1, G2, D1 and D2 ( $N = 10$ ). A sample red bounding box, indicating location of defect, is also shown.

2) *Dimensionality Reduction and Outlier Detection:* For golden ICs, the feature vectors should be densely positioned, whereas ICs with defects should behave like outliers. In our method, the inter- and intra-distance vectors of each chip are concatenated to form a feature vector of size  $2N^2$ . This is

TABLE II

VARIANCE COMPUTED AMONG THE INTER-CHIP DISTANCES FOR EACH TYPE OF ICs ( $10^{-8}$ )

| IC Type | w/ Marking<br>N=5     | w/out Marking<br>N=5 | w/ Marking<br>N=10    |
|---------|-----------------------|----------------------|-----------------------|
| G1      | 2.54                  | 3.90                 | 1.06                  |
| G2      | 3.90                  | 4.34                 | 2.77                  |
| D1      | 11.6                  | 15.6                 | 4.63                  |
| D2      | 20.5                  | 13.3                 | 12.13                 |
| IC Type | w/out Marking<br>N=10 | w/ Marking<br>N=23   | w/out Marking<br>N=23 |
| G1      | 1.26                  | 0.10                 | 0.10                  |
| G2      | 2.67                  | 0.13                 | 0.13                  |
| D1      | 5.32                  | 0.24                 | 0.30                  |
| D2      | 13.15                 | 0.55                 | 0.45                  |

done to find the outliers based on both within and between chip variability. Since the number of chips available to test is often small, it is difficult to detect outliers directly from these  $2N^2$  dimension feature vector. For this reason, Principal Component Analysis (PCA) is applied on the combined dataset, and the principal components having a total of 90% variability, which gave the best result, is taken for outlier detection. The measure that is used for outlier detection is the local outlier factor (LOF) [17]. This factor is a measure that computes the local deviation of a given data point with respect to its neighbors. The greater the LOF, the more likely the IC under test is an outlier. So a threshold is chosen by selecting the maximum LOF observed among a given set of golden ICs.

If an IC fails in both of these analyses (i.e., variance and outlier detection) then there is high probability that, it is a counterfeit IC.

#### IV. EXPERIMENTAL RESULTS AND DISCUSSION

All the aforementioned algorithms are implemented using MATLAB (v. 2018a). The code for LGDiP feature extraction was obtained from [19]. We took images of four different types of IC sets, two genuine and two defective. We label the two genuine IC sets as G1 and G2 while the two defective IC sets are D1 and D2. IR images are acquired using PHAMOS-1000 Emission microscope (C1122-16) for G1, G2, and D1 while Nikon D850 with external IR/NIR lens (62 mm diameter, pass spectrum above 720 nm) was used for D2 due to better field of view. The laser setting used for PHAMOS is high power IR-LD module with  $1.3\mu\text{m}$  wavelength and lens in OBRICH mode with 5X resolution. Optical images were acquired using Leica DVM6 digital optical microscope. Experiments are done for  $N = 5, 10, 23$  and direct intensity values of images. For LGDiP feature extraction, the parameter values used are  $R = 8$  and  $S = 5$ . Hence, the number of features obtained for each block ( $d$ ) is 280.

##### A. Dataset

Our dataset contains four types of ICs, as shown in Figure I. The golden ICs, referred to as G1 and G2 type, are ICs bought directly from the manufacturers and are therefore authentic. Among the defective ICs, D1 has the most defects. Since many of the D1 category ICs contain ghost markings, they can be declared as counterfeit ICs. D2 is another category of defective ICs having less surface texture defects than D1. Our dataset contains 5 ICs of each



TABLE III  
LOCAL OUTLIER FACTOR (LOF) SCORES

| IC Type   | N=5           | N=10          | N=23          | Image Intensity |
|-----------|---------------|---------------|---------------|-----------------|
| G1        | 0.9885        | 0.9976        | 0.9977        | 1.0110          |
|           | 0.9898        | 0.9982        | 0.9991        | 0.9693          |
|           | 1.0016        | 0.9945        | 0.9968        | 0.9867          |
|           | 0.9872        | 0.9892        | 0.9947        | 0.9898          |
|           | 0.9866        | 0.9915        | 0.9993        | 0.9693          |
| G2        | 1.0011        | 1.0072        | 1.0000        | 1.4107          |
|           | 1.0237        | 1.0087        | 0.9997        | 1.5020          |
|           | 0.9964        | 1.0067        | 0.9972        | 1.4782          |
|           | 0.9919        | 0.9892        | 0.9995        | 1.4399          |
|           | <b>1.0651</b> | <b>1.0178</b> | <b>1.0050</b> | <b>1.5871</b>   |
| D1        | 0.9959        | 1.4217        | 1.0063        | 1.6700          |
|           | 1.6377        | 1.5644        | 0.9947        | 1.6743          |
|           | 1.0184        | 1.3187        | 1.0020        | 0.9980          |
|           | 1.0047        | 1.3512        | 1.0035        | 1.2630          |
|           | 0.9965        | 1.4466        | 1.0048        | 1.0393          |
| D2        | 1.6626        | 1.2948        | 1.1456        | 1.1823          |
|           | 1.7297        | 1.1842        | 1.2093        | 1.1305          |
|           | 1.6213        | 1.2722        | 1.1729        | 1.1878          |
|           | 1.6501        | 1.2329        | 1.2199        | 1.1733          |
|           | 1.6561        | 1.2743        | 1.1235        | 1.1853          |
| Threshold | <b>1.0651</b> | <b>1.0178</b> | <b>1.0050</b> | <b>1.5871</b>   |
| Accuracy  | <b>80%</b>    | <b>100%</b>   | <b>80%</b>    | <b>40%</b>      |

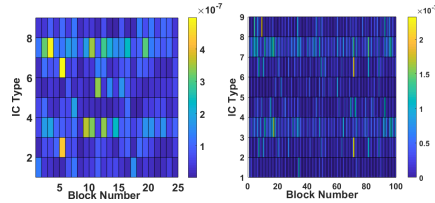


Fig. 8. Block-wise Intra-chip Distance Variability.

for G1, G2, and D2 type and 10 ICs for D1 type. All the ICs are reduced to a similar dimension of  $230 \times 230$  pixels. This resulted in some information loss, but the effect of scratches and defects is still noticeable. Pre-processing is done to remove markings<sup>1</sup> from the chips. Results are shown for both with and without markings.

#### B. Inter-Chip and Intra-Chip Distance Measurement Results

The results for inter- and intra-chip distances are represented using surface (surf) plots. For each IC, two  $N \times N$  grid surf plots are obtained in which each of the blocks of the grid corresponds to the same block in the original image. Samples of inter and intra surf plots, with and without markings, for each of G1, G2, D1 and D2 are shown in the first two columns of Figure 6 and Figure 7 respectively. The same distances calculated using the intensity values as features, instead of LGDiP, are also given in the intensity columns of the figures for comparison. The result shows, for G1 and G2 type ICs all the surf plots are mostly of the same color (blue and light blue). However, the surf plots of D1 and D2 types show more variation. They have different color shades, representing variability in the inter and intra chip distances among the blocks. Further, the green and yellow colors indicate high variance in associated blocks. The defects found on D1 type ICs are mainly texture inconsistency while the defects found on D2 are mainly scratches and resurfacing observed along the chip periphery.

<sup>1</sup>Markings are writings found on the surface of the chips containing company name, batch number, etc.

#### C. Variance

Table II and Figure 8 shows the variances calculated among all the inter-chip distances and block-wise intra-chip distances respectively. Results are tabulated for inter-chip variability with 100 blocks and the intra-chip variability is shown for 25 and 100 blocks in Figure 8, where first four types represents ICs with markings and next four represents same ICs without markings. Since similar location blocks are compared, no effect of markings is observed on the results. In fact, automated removal of markings may result in some information loss because of pre-processing. The result shows that variances in D1 and D2 are higher than the genuine ICs. This is because the texture pattern between the same blocks varies more with the presence of defects.

#### D. Outlier Detection

LOF values for each of the ICs are shown in Table III for  $N=5, 10$ , and  $23$ . We have also performed outlier detection by directly taking pixel intensity of the images as input to show that our proposed use of IR gave superior results. Unlike SMEs, machines are not as capable of detecting counterfeit ICs just by looking at the images. The major drawbacks of directly working with the pixel intensity values are the presence of unwanted noise and reflection and the difficulty in removing them. The threshold boundary, for our technique, is chosen by taking the maximum LOF score observed among the golden ICs only, that is, type G1 and G2. It is observed that our approach gave 100% accuracy for  $N=10$  and less accuracy for  $N=5$  and  $23$ . This is because the texture is best described as a local feature. If  $N$  is large then the local feature information will be lost, and it will be more of a global feature. Also, if  $N$  is substantially small, the texture features will be dominated by noise, which for our dataset, where the difference between two textures is very low, may increase the number of false positives. Hence, the results show it is important to choose the optimal number of blocks as well as threshold.

#### V. CONCLUSION

We have developed a two-fold feature extraction technique capable of detecting multiple defects present on the package of ICs. To summarize, the benefits for our approach are:

- The proposed technique is capable of detecting a wide variety of surface defects like scratches, texture differences, difference in indent position, sanding/grinding marks, ghost markings, burned markings, mold variation, package damage, corrosion/contamination, and extraneous markings.
- It is fully automated, non-destructive, and does not require any complex image acquisition infrastructure.
- It does not require prior knowledge of the golden IC, only golden textures which can be obtained from any authentic IC.
- It is capable of estimating the location of one or more defects at once.
- To the best of our knowledge, this paper, for the first time, uses IR imaging for counterfeit IC detection.

If the number of blocks  $N$  and the threshold are chosen appropriately, it is capable of producing 100% accuracy. This selection can be easily made at the initial tuning phase of the algorithm.

## REFERENCES

- [1] : U.S. Department of Commerce, Office of Technology Evaluation, Counterfeit Electronics Survey, November 2009.
- [2] Yiorgos Makris, John M. Carulli, Ujjwal Guin, Ke Huang, Mohammad Tehranipoor (2014) Counterfeit Integrated Circuits: A Rising Threat in the Global Semiconductor Supply Chain.
- [3] Tehranipoor, M.M., Guin, U. and Forte, D., 2015. Counterfeit integrated circuits. In *Counterfeit Integrated Circuits* (pp. 15-36). Springer, Cham.
- [4] Guin, U., Forte, D. and Tehranipoor, M., 2013, December. Anti-counterfeit techniques: from design to resign. In *2013 14th International workshop on microprocessor test and verification* (pp. 89-94). IEEE.
- [5] P. Ghosh and R. S. Chakraborty, "Recycled and Remarked Counterfeit Integrated Circuit Detection by Image-Processing-Based Package Texture and Indent Analysis," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 1966-1974, April 2019.
- [6] Liu, L., Chen, J., Fieguth, P. et al. From BoW to CNN: Two Decades of Texture Representation for Texture Classification. *Int J Comput Vis* 127, 74–109 (2019). <https://doi.org/10.1007/s11263-018-1125-z>
- [7] P. Ghosh and R. S. Chakraborty, "Counterfeit IC Detection By Image Texture Analysis," 2017 Euromicro Conference on Digital System Design (DSD), Vienna, 2017, pp. 283-286.
- [8] Ghosh, P., Bhattacharya, A., Forte, D. et al. Automated Defective Pin Detection for Recycled Microelectronics Identification. *J Hardw Syst Secur* 3, 250–260 (2019). <https://doi.org/10.1007/s41635-019-00069-7>
- [9] K. Huang, J. M. Carulli, Y. Makris, "Parametric counterfeit ic detection via support vector machines", *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Nanotechnol. Syst.*, pp. 7-12, 2012.
- [10] B. Gassend, D. Lim, D. Clarke, M. Van Dijk, S. Devadas, "Identification and authentication of integrated circuits", *Concurrency Comput.: Practice Exper.*, vol. 16, no. 11, pp. 1077-1098, 2004.
- [11] U. Guin, M. Tehranipoor, D. DiMase, M. Megrđichian, "Counterfeit IC detection and challenges ahead", *ACM SIGDA*, vol. 43, no. 3, pp. 1-5, 2013.
- [12] Birgé, L. and Massart, P., 2001. Gaussian model selection. *Journal of the European Mathematical Society*, 3(3), pp.203-268.
- [13] Manjunath, B. S., Ma, W.-Y. (1996). Texture features for browsing and retrieval of image data. *IEEE TPAMI*, 18(8), 837–842.
- [14] T. Jabid, M.H. Kabir, and O. Chae, Local directional pattern (LDP)—A robust image descriptor for object recognition, *Advanced Video and Signal Based Surveillance (AVSS)*, in: 2010 Seventh IEEE International Conference on, IEEE, 2010, pp. 482–487.
- [15] S.Z. Ishraque, A.H. Banna, and O. Chae, Local Gabor directional pattern for facial expression recognition, in: *Computer and Information Technology (ICCIT)*, 2012 15th International Conference on, IEEE, 2012, pp. 164–167.
- [16] Turan, C. and Lam, K.M., 2018. Histogram-based local descriptors for facial expression recognition (FER): A comprehensive study. *Journal of visual communication and image representation*, 55, pp.331-341.
- [17] Kriegel, H.P., Kröger, P., Schubert, E. and Zimek, A., 2009, November. LoOP: local outlier probabilities. In *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 1649-1652).
- [18] Huang, J., Zhu, Q., Yang, L. and Feng, J., 2016. A non-parameter outlier detection algorithm based on Natural Neighbor. *Knowledge-Based Systems*, 92, pp.71-77.
- [19] Cigdem Turan (2020). *ToolboxDESC* (<https://www.github.com/cigdemturanc/ToolboxDESC>), GitHub. Retrieved March 13, 2020.