

Image processing for detection of product tampering using a camera

Nguyen-Khang Vo, *Electrical and Computer Engineering Department, National University of Singapore*

Abstract—PIN entry device (PED) is one type of devices that require high security protection since it directly receives information about customer's card and PIN. Many methods has been applied to make the PEDs tamper-proof, including complicated sensing circuits, but PEDs still remains vulnerable to many attacks, typically in the type of object intrusion. This paper present another tamper-resistance method using computer vision technique to protect the product at the PCB assembly level. Every time the device is powered on, the image of the PCB is captured and compared with the reference image to check whether suspicious components have been inserted into the device. The approach used in this paper is background subtraction with image registration as the pre-processing step to enhance accuracy. With this approach, most of the test objects are detected successfully, even for small objects such as a $1.25\text{mm} \times 2\text{mm}$ SMD capacitor and even a 0.25mm diameter AWG30 wire.

I. INTRODUCTION

Product tampering involves the deliberate altering of a system, leading to unexpected results that may cause damage or loss to other people. Preventing product tampering may require actions in all phases of product production, distribution, logistics, sale, and use. This problem is most concerned in security critical devices such as the PIN entry devices (PEDs) because they directly receive information about customer's card and PIN. Moreover, since PEDs are available at almost every point-of-sale (POS), they become attractive targets for credit card thieves. In 2008, Drimer [1] demonstrated two simple attacks using a bent paperclip and a needle to attach to the data lines on the PCB and the tapped signals were decoded successfully. He suggested that detecting such a tap from within the PED is extremely difficult, since high input-impedance probes do not significantly distort signals, and proper termination suppresses reflections. Therefore, the system cannot detect the difference when the wires are attached to the data lines. In contrast, human's vision system can easily see the suspicious items and wires. That is the reason why such trivial attacks can defeat PED's anti-tampering system. To overcome this problem, this paper propose a new anti-tampering method based on computer vision and image processing techniques to protect the product at the PCB assembly level. Basically, a camera is mounted inside the product and the image of the internal components is stored as a reference before the product is release to the market. Every time the device is powered on, the new image is captured and compared with the reference data to check whether suspicious components have been inserted into the device.

There are 4 assumptions made in this paper:

- 1) The images obtained have good quality for identifying the objects.
- 2) The PCB can be rotated and translated up to 0.5mm in any 2D direction and the camera can be rotated up to 3 degrees.¹
- 3) The illumination intensity is sufficient, and the illumination condition does not change suddenly.
- 4) Computational power is sufficient to perform the tasks.

The approach used in this paper is background subtraction with image registration as the pre-processing step to enhance accuracy. The post-processing step consists of image filtering and morphological operation, so that a proper decision can be made from the result. With this approach, most of the test objects are detected successfully, even for small objects such as a $1.25\text{mm} \times 2\text{mm}$ SMD capacitor and even a 0.25mm diameter AWG30 wire.

The remaining of this paper is organized as follow: Section II reviews the works on related fields, including automated optical inspection, image registration, feature detection and background subtraction. Section III focuses on the image registration process using SIFT feature descriptor. Section IV discusses about the key parameters of Gaussian mixture model (GMM) background subtraction. Section V presents the flow of the whole system with details about each step. Section VI discusses about the results of the project and the drawbacks of the approach. Section VII concludes the project and suggests improvements for future work.

II. LITERATURE REVIEW

A. Automated optical inspection

Automatic optical inspection (AOI) is one of the earliest applications of computer vision in the industry. In 2003, Malamas et al. [2] described the components of a typical AOI system, as shown in Figure 1. Depending on the constraints on speed or accuracy, special hardware is used, such as DSP, ASIC, or FPGAs.

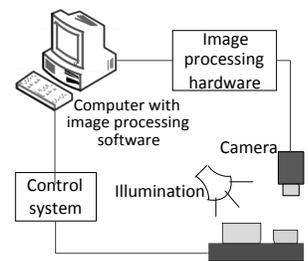


Fig. 1: A typical AOI system [2]

The operating sequence of the system are summarized consists of 4 steps: Image acquisition, Image processing, Feature extraction, and Decision-making. Firstly, the generated feature set is refined to match the size of the reference set. After that, the reduced feature set is compared with the

¹The amount of translation and rotation is specified by VeriFone

reference set through many stages to reach an application-specific decision. The decision may involve processing with thresholds, statistical or soft classification.

Background subtraction is the direct approach, in which the image of the PCB under inspection is compared with a reference perfect PCB. After that, the obtained difference is displayed and analyzed, such as using morphology operation [3]. This technique is simple and fast, but it suffers from several problems: large storage for the reference, precise alignment is needed, and it is sensitive to illumination. The solutions to these problems are using complex illuminating and alignment subsystems, or applying another pre-processing step to align the images by detecting the special marks on the PCB [4]. Generally, industrial vision systems used specialized subsystems to achieve high precision alignment and uniform illumination.

Besides background subtraction, feature matching is another widely-used technique in AOI systems. In this technique, the features are extracted from the PCB under inspection and compared with the model. The advantage of this technique is less data for storage, reduced sensitivity to the input data. However, the limitation is that the feature set must be built for every object under inspection. In 2008, Garcia [5] proposed an automated feature selection methodology using multivariate step-wise discriminant analysis methods to come up with a reduced set of features that gives the best discrimination.

B. Image registration

Image registration is the process of transforming different sets of data into one coordinate system. In 2006, Szeliski [6] reviewed the 2D and 3D motion models and the image alignment algorithms, including direct (pixel-based) alignment and feature-based registration. Projective, or perspective transform or homography, is the super-set of all other 2D transformations. Its transformation function can be defined as $\tilde{x}' \sim \tilde{H}\tilde{x}$, where \tilde{H} is an arbitrary 3×3 matrix. After having the 2D transform model, the parameters can be estimated by direct (pixel-based) alignment or feature-based registration. Szeliski also discussed about the two approaches, and he supported the feature-based method because of the following reasons:

- The direct method has high computational requirement, even with the image pyramid (coarse-to-fine) estimation.
- The feature-based method is remarkably robust: it can match images that differ in scale, orientation, and even foreshortening.

Because of the above reasons, in this paper, the feature-based approach will be used. Furthermore, to simplify the process of selecting features to track, special marks can be painted on the device and they can be used as the supplement tracking features.

The task of finding the correspondence points from two sets of features can be done by RANSAC (RANDOM SAMPLE Consensus) [7] algorithm. There are also many variations of RANSAC to improve accuracy (MLE-SAC, LO-RANSAC),

speed (R-RANSAC) or robustness (MAPSAC). The idea of RANSAC is to randomly choose a number of samples from the set of all measurements, try to fit a model to them, and check how many other points are in consensus with this model estimate. The process is repeated and the best fit, i.e. an estimate supported by the maximal number of measurements, is left as a solution. All other points are treated as outliers.

C. Feature detection

Many feature detection algorithms have been developed including SIFT (Scale Invariant Feature Transform) [8], SURF (Speeded Up Robust Feature) [9], GOH (Gradient Location and Orientation Histogram) [10]... Also in [10], those descriptors were compared in performance; SIFT and GLOH (an extension of SIFT) outperform other descriptors, but SURF is not included in that study. Later, in 2006, Bauer *et al.* [11] compared SIFT, SURF and their variants; and the results showed that SURF is several times faster than SIFT, with slightly lower accuracy. In 2009, Yu and Morel [12] introduced ASIFT (Affine-SIFT) which is fully invariant with respect to all six parameters, namely zoom, rotation, translation, and two parameters defining the camera axis orientation. However, ASIFT has twice the complexity of SIFT.

1) *Scale Invariant Feature Transform (SIFT)* : SIFT [8] is a well-known algorithm in computer vision to detect and describe local features in images. The SIFT algorithm consists of 5 stages: Scale-invariant feature detection, Feature matching and indexing, Cluster identification by Hough transform voting, Model verification by linear least squares, and Outlier detection.

In 2010, Hess [13] presented an open-source SIFT library², implemented in C using OpenCV with similar accuracy and performance as the original implementation. There are also other open source implementations available^{3,4}.

2) *Speeded Up Robust Feature (SURF)*: SURF [9] is partly inspired by the SIFT descriptor, which is claimed to approximate or even outperform previously proposed schemes with respect to repeatability, distinctiveness, and robustness. SURF relies on integral images for image convolutions to reduce computation time, builds on the strengths of the leading existing detectors and descriptors (using a fast Hessian matrix-based measure for the detector and a distribution-based descriptor). It describes a distribution of Haar wavelet responses within the interest point neighborhood. Integral images are used for speed and only 64 dimensions are used reducing the time for feature computation and matching. The indexing step is based on the sign of the Laplacian, which increases the matching speed and the robustness of the descriptor.

²Rob Hess's SIFT library: <http://eecs.oregonstate.edu/~hess/sift.html>

³Fast SIFT image features library: <http://sourceforge.net/projects/lib sift/>

⁴VLFeat open source library: <http://www.vlfeat.org/>

TABLE I
COMPARISON IN DARKENING, NOISY NIGHT AND CAMOUFLAGE CHALLENGES

Method	Darkening	Noisy night	Camouflage
Stauffer1999	–		
Oliver2000	–		+
Li2003	+	+	
Zivkovic2006	+		+
Maddalena2008	+	+	+
Barnich2009	+	+	–

D. Background subtraction

Background subtraction (BS), or segmentation, is a widely used method to detect moving objects from a static scene. The works on BS is vast, but this paper will only review the commonly used techniques, especially on some BS method that are adaptable to varying illumination.

In 2004, Piccardi [14] classified the BS methods into 7 groups, namely Running Gaussian average, Temporal median filter, Mixture of Gaussians, Kernel density estimation (KDE), Sequential kernel density approximation (SKDA), Concurrence of image variations, and Eigen backgrounds. He also conducted a performance analysis based on speed, memory requirements and accuracy. Although the author only compared the accuracy qualitatively, he suggested that Running Gaussian average and Temporal median filter can guarantee adaptation to slow illumination changes, but cannot cope with multi-valued background distributions. Mixture of Gaussians and KDE can model well the background pdf in general cases, but MoG has high computational requirement and KDE has a high memory requirement which might not be suitable for low-memory devices. SKDA is not accurate as KDE, but it requires less memory and computational power. Co-occurrence of image variations and the eigenbackgrounds offer good accuracy against reasonable time and memory complexity. However, practical implementation of the co-occurrence method imposes a tradeoff with resolution. In 2009, Herrero and Bescos [15] performed further quantitative analysis and the result generally agreed with the qualitative analysis. Besides, the authors emphasized the importance of choosing the optimum parameters and the kind of approach (uni-modal/multimodal) according to the data set.

Recently, in 2011, Brutzer *et al.* [16] listed the main challenges of BS and compared the BS methods on the ability to meet those challenges. There are several challenges that directly related to this project, such as darkening, noisy night, camouflage. Table I compares the recent BS techniques in the related challenges.

III. IMAGE REGISTRATION USING SIFT

The image registration process consists of two stages: (a) estimate transformation matrix based on the correspondence points of the two images, and (b) transform the image using the matrix obtained in (a).

A. Extract SIFT features from image

a) *Scale-Space Extrema Detection:* The Gaussian scale-space pyramid is constructed as follow: for each level, or octave, a set of images of the same size is created, where the i^{th} image is the convolution of the 2D Gaussian function

with the $(i-1)^{th}$ image: $I(x, y)_i = g(x, y) \star I(x, y)_{i-1}$ where $g(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$. The value of σ is determined by the equation $\sigma_i = \sigma_0 k^i$. In Lowe's implementation, the base scale level $\sigma_0 = 1.6$, and $k = 2^{1/N}$ where $N = 3$ is the number of intervals per octave. When moving to the upper level of the image pyramid, the image is down-sampled to a quarter of its size (half in each dimension).

Next, the Difference of Gaussians (DoG) pyramid is formed by subtracting the adjacent scale images of the same octave. For example, if the base of the Gaussian pyramid has 5 images, then the base of the DoG pyramid has 4 images.

Keypoints are identified by comparing each pixel in the DoG images to its 26 neighbors at the same scale and in two neighboring scales. If the pixel value is the maximum or minimum among all compared pixels, it is selected as a candidate keypoint.

b) *Keypoint localization:* The candidate keypoints obtained in the first stage is first put through the interpolation step to increase the accuracy of the pixel position. Then these keypoints are checked to remove that the bad keypoints (including edges and low-contrast keypoints), and only the good keypoints are selected. By removing low-contrast keypoints, the spurious keypoints produced by noise are also eliminated. After this stage, the obtained keypoints are scale-invariant. There are several important parameters in this step:

1) Contrast threshold $|D(\hat{x})|$: keypoints with contrast lower than this value will be removed to improve stability and robustness of the SIFT algorithm. In Lowe's implementation, this contrast threshold is 0.03. However, in the experiment, even with the contrast threshold 0.1, more than 100 SIFT features can be extracted from the test image.

2) Ratio of principal curvatures threshold: keypoints with the ratio of principal curvatures greater than this value will be removed. This threshold is used to remove edges. In Lowe's implementation, this threshold is 10.

c) *Orientation assignment:* The keypoints are assigned their dominated gradient directions, so that they are invariance to rotation as the keypoint descriptor can be represented relative to this orientation. The image is Gaussian-blurred, and the gradient of the neighborhood of the keypoint are calculated and allocated to a 36-bin orientation histogram, each bin is 10-degree wide. From the histogram, any bin within the top 80% is defined as a keypoint, and the corresponding orientation is assigned to that keypoint.

d) *Keypoint descriptor:* In this final step, the gradient data of the neighboring pixels in the 16×16 window around the keypoint is used to compute the descriptor. This 16×16 window is divided into 16 smaller windows, each of size of 4×4 . The gradients in each of the 4×4 window are put in 8 bins, each bin 45-degree wide. Next, the Gaussian weighing operator is applied on the 4×4 window so that the further the gradients from the keypoint, the lesser is contribute to the bin. After this step, each 4×4 window is represented by 8 values (of the 8 bins). With 16 of these 4×4 windows, each keypoint will be represented by a total of $16 \times 8 = 128$ values. This 128 dimensional vector is one SIFT feature.

B. Estimate the motion model's parameters

From Assumption 2, it can be concluded that the transformation model is *affine*, with 6 degrees of freedom and the dimension of the transformation matrix is 2×3 . However, in this project, it is better to assume the transformation is *perspective* (8 d.o.f, 3×3 transformation matrix) because of the following reasons:

- It is difficult to guarantee the conditions stated in assumption 3 in reality (including when obtaining the test images, and even when dealing with real device)
- Perspective transformation is the super-set of affine transformation. In other words, an affine transformation can be represented by a perspective transformation, but the other way around is not correct in general.

Therefore, the motion model of the object will be considered as perspective transformation, or homography.

From the SIFT features extracted, the RANSAC algorithm is used to find the corresponding points, and the 3×3 transformation matrix can be estimated from these corresponding points using least square method. In homogeneous form, the relation between the correspondence points $(x, y, 1)$, $(x', y', 1)$ and the transformation matrix is:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \text{ To enforce 8 d.o.f, set } h_{33} = 1.$$

In equation form: $x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + 1}$, $y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + 1}$

Rearrange: $h_{11}x + h_{12}y + h_{13} - h_{31}xx' - h_{32}yy' = x'$,
 $h_{21}x + h_{22}y + h_{23} - h_{31}xy' - h_{32}yy' = y'$

With n pairs of correspondence points, we have to solve the linear least square system $A\hat{x} = B$:

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1y'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1x'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2y'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2x'_2 \\ \dots & \dots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_nx'_n & -y_ny'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_ny'_n & -y_nx'_n \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ \dots \\ x'_n \\ y'_n \end{bmatrix}$$

SVD (singular value decomposition) is a powerful method to solve this type of over-determined problem.

C. Transform image

After having the transformation matrix, the new image is warped back so that the new image is aligned with the reference image. The interpolation method used is bicubic interpolation. From experimental results, the absolute difference between the two images without alignment is much greater compared with the case when the image is aligned.

IV. BACKGROUND SUBTRACTION USING GAUSSIAN MIXTURE MODEL

The GMM BS algorithm is similar to the standard Stauffer & Grimson algorithm with additional selection of the number of the Gaussian components. In OpenCV library, the GMM algorithm is implemented in the BackgroundSubtractor class. There are several important parameters:

CV_BGFG_MOG2_NGAUSSIANS This is the number of Gaussians K in mixture (default: 5). K determines the accuracy of GMM. High value of K will increase the accuracy,

but the memory usage will also increase. However, with low K , GMM cannot detect small changes.

CV_BGFG_MOG2_SHADOW_VALUE Set to 0 will disable shadow detection, and shadows will be treated as foreground object. In this application, shadow detection is not needed because a) with small motion, the object's shadow does not change much, and b) it is desired to detect anything different from the reference image.

CV_BGFG_MOG2_MINAREA This parameter is used for post filtering. Foreground objects smaller than this threshold will be deleted and treated as background object.

It is important to note that even when the image is aligned before subtraction, there is still a lot of noise in the raw result.

V. SYSTEM INTEGRATION

A. Capture new image

OpenCV provides some basic functions to interface with the camera, such as to set the width and height of the camera frame, and to capture new image from the selected camera. However, in the experiment, the test images is captured separately, so that it is easier to control the illumination and motion of the PCB.

B. Align new image

The images are only obtained from one camera, but at different times, and they can be from slightly different viewpoints due to physical impacts on the image acquisition system. Therefore, image registration is used as a pre-processing step to enhance the accuracy of the background subtraction process.

This is one of the steps that requires much processing power to extract the SIFT features and estimate the transformation matrix from the correspondence points. To reduce the processing time, it is preferred to have fewer features. However, with fewer features, there will be fewer correspondence points, and the estimated transformation matrix will not be accurate enough.

If the images are not aligned before subtractions without alignment, the background image becomes very blurry. Consequently, the result of GMM greatly differs from the ground truth image. Therefore, it is clearly that the alignment step is important to achieve accurate results.

C. Subtract background

In this paper, GMM BS is chosen because of its speed and its ability to adapt to illumination changes. For the GMM background subtraction to work properly, several images are needed to build the reference background model. With many images in different illuminations, the background model will be able to detect more accurately.

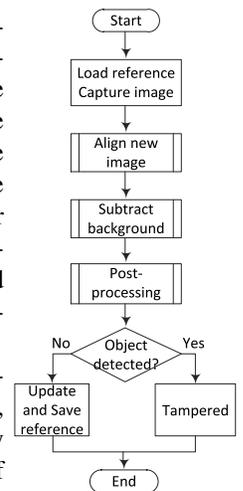


Fig. 2: Flowchart

D. Save and load reference data

After processing, the background model data is saved to non-volatile memory, so that subsequent runs can load this data and continue from the data of all the previous runs.

E. Post-process

Although the image has been aligned before subtraction, there are still a lot of noisy pixels in the raw result (Figure IV). These discrete noise pixels arise from the difference in illumination that the GMM BS cannot handle, and also from the interpolation step when the image is aligned. Besides, the detected object in the raw result of GMM BS is a group of closely clustered points. Therefore, the post-processing step is needed to remove discrete noise pixels and group the cluster of pixels together as a solid object. To achieve that result, the raw BS image is first convolved with a 3×3 median filter, and then the closing morphology function with a 5×5 square kernel is applied on the image twice to fill small holes in the cluster of points.

F. Make decision

Although the post-processing step can handle most of the noise and fill the object, there are still some remaining noise that cannot be removed due to great changes in lighting intensity or in the position. For example, the borders of the processed image have large incorrectly identified foreground objects because of the alignment step.

A simple way to deal with this problem is to set the region of interest to be a small rectangle inside the image, and count for non-zero values in that region. If the number of non-zero pixels exceeds a certain threshold, it is considered an object. Otherwise, it is considered noise. With the setup as in the previous section, the object threshold is determined as 100 pixels.

VI. EXPERIMENT AND RESULT

To obtain the test images, the PCB is put under slightly different lighting intensities and positions. The test object is placed on the PCB at several different locations. All the test images have the same size 800×600 pixels. The SIFT contrast threshold is set to 0.5. The object's size threshold is 100 pixels. All other parameters are left default. The first 5 test images are "clean" images with no test objects to build the background model. After that, the images with test objects are loaded and checked. Figure 3 shows the results of 6 test cases. The left most column is the raw output of GMM background subtraction, the middle column is the image after the post-processing step, and the right most column is the ground truth. The rectangle boxes in the middle column images are the region of interests.

To evaluate the accuracy of the method, the error percentage is calculated as $\%error = \frac{fg_pixels - real_size}{real_size}$. Generally, we want the detected foreground object to be equal or bigger than the real object. If the number of detected foreground pixels is smaller than the real object size, this error percentage will be negative, which is not desired.

TABLE II
ERROR PERCENTAGE OF 6 TEST CASES

Case	# foreground pixels	Real size (pixels)	%error
a	2440	1981	23
b	804	901	-11
c	5053	4869	4
d	396	372	6
e	1446	2158	-33
f	6220	4939	26

Although the system can detect all objects, there are still cases when the error percentage is negative, especially for small objects such as case (b) and (e). With the AWG30 wire as the test object, the detected foreground object is much smaller than the real object (-33%). The problem comes from the background subtraction step, not in the post-processing step, because the wire is very thin, and its color does not differ much from the background objects, only the parts of the wire where there is strong reflection can be detected.

Another drawback of this approach is that the large amount of data needs to be saved and loaded. This problem comes from the nature of GMM BS because a large number of Gaussians are used to model the background. This problem can be compensated if the number of Gaussians in mixture is reduced; for example, when K decreases from 5 to 4, the reference file size reduces from 76MB to 68MB. However, the accuracy of the GMM BS decreases as well. Figure 4 shows the error percentage at different values of K . With $K \geq 5$, the error percentage remains constant regardless of K . With $K < 4$, the error percentage quickly deviates from the best achievable values. Therefore, the $K = 4$ is the most suitable value to balance between accuracy and complexity.

VII. CONCLUSION AND FUTURE WORK

In this paper, a new anti-tampering method based on computer vision and image processing techniques is proposed. This new layer of protection can protect the product at the PCB assembly level. The main idea is to capture the image of the PCB and compare it with the reference image to check whether any suspicious component (wires, bugs, etc.) has been inserted. The approach used in this project consists of three main steps: the pre-processing step aligns the image by using SIFT features, so that small motions are compensated; the main step performs Gaussian mixture model background subtraction which can adapt to small changes in illumination; the post-processing step cleans up noise and makes the objects easier to detect. The result of this approach, small objects such as a $1.25mm \times 2mm$ SMD capacitor and even a $0.25mm$ diameter AWG30 wire can be detected.

Although the idea of using computer vision for anti-tampering application is examined and verified to have good results even with small object, there is still much work to be done before this idea can be applied on real devices. Further study on optimizing the approach for low power processor is needed. In addition, the functions in the OpenCV library needs to be ported to run on the embedded operating system. Furthermore, some computational intensive tasks such as image warping and loading/saving data can be implemented in hardware description language (VHDL/Verilog) and run on dedicated hardware to improve performance.

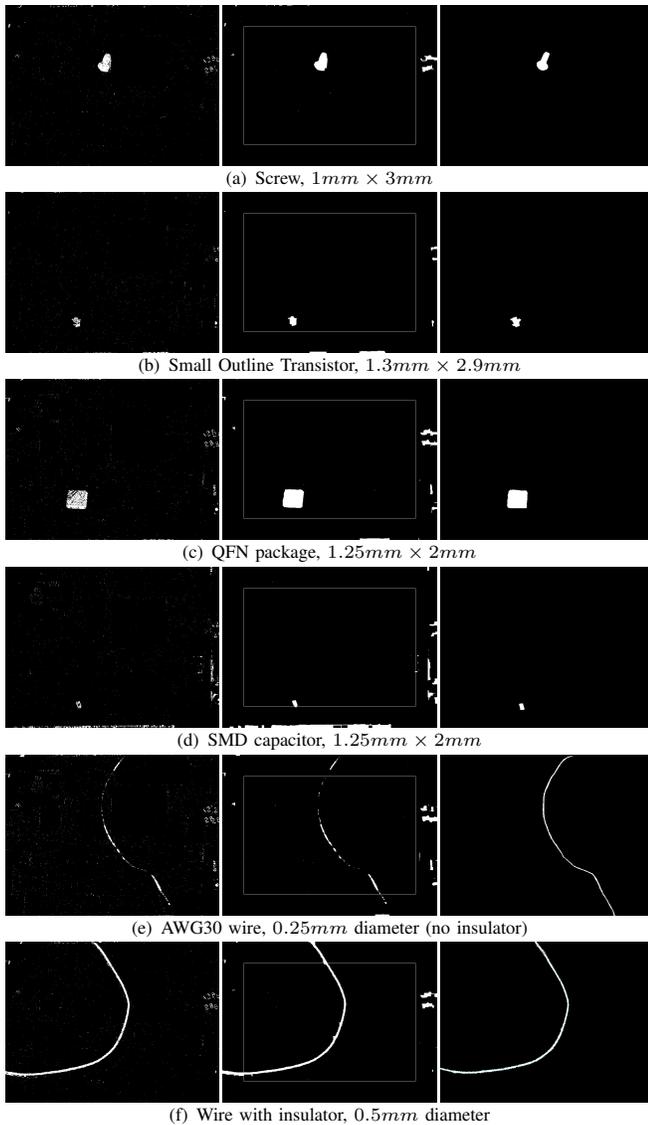


Fig. 3: Test results

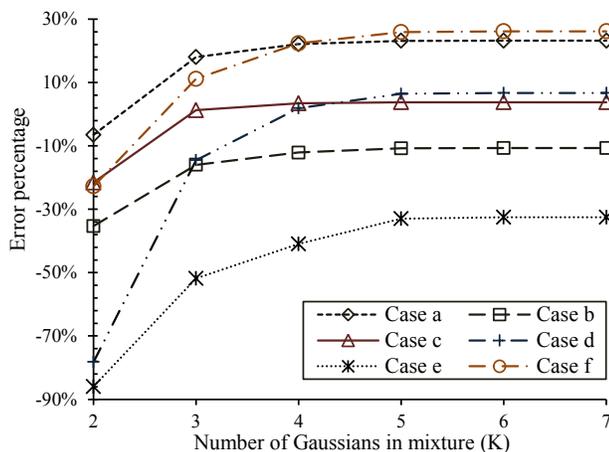


Fig. 4: Error percentage vs. Number of Gaussians in mixture

ACKNOWLEDGMENT

The author would like to thank Dr. Cheong Loong Fah and Mr. Leow Siew Kiat for their guidance and support throughout the project. This project is initiated by VeriFone Singapore Pte Ltd under the industry collaboration scheme.

REFERENCES

- [1] S. Drimer, S. J. Murdoch, and R. Anderson, "Thinking inside the box: System-level failures of tamper proofing," in *Proc. IEEE Symp. Security and Privacy SP 2008*, 2008, pp. 281–295.
- [2] E. N. Malamas, E. G. M. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image and Vision Computing*, vol. 21, no. 2, pp. 171–188, 2003.
- [3] L. zong Lin, L. shan Zhou, J. ding Wan, and Z. qin Qian, "Study of PCB automatic optical inspection system based on mathematical morphology," in *Proc. Int. Conf. Computer Technology and Development ICCTD '09*, vol. 2, 2009, pp. 405–408.
- [4] F. Lei, J. Cheng, S. Guo, and F. Zhang, "A locating algorithm based on OGHT for PCB mark orientation," in *Proc. Int Information Networking and Automation (ICINA) Conf*, vol. 1, 2010.
- [5] H. C. Garcia and J. R. Villalobos, "A feature selection method for automated visual inspection systems," in *Proc. 6th IEEE Int. Conf. Industrial Informatics INDIN 2008*, 2008, pp. 1371–1376.
- [6] R. Szeliski, "Image alignment and stitching: a tutorial," *Found. Trends. Comput. Graph. Vis.*, vol. 2, pp. 1–104, January 2006.
- [7] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
- [8] D. G. Lowe, "Object recognition from local scale-invariant features," *Computer Vision, IEEE International Conference on*, vol. 2, pp. 1150–1157, August 1999.
- [9] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, 2008.
- [10] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.
- [11] J. Bauer, N. Sünderhauf, and P. Protzel, "Comparing several implementations of two recently published feature detectors," *University of Pennsylvania Law Review*, vol. 154, no. 3, pp. 477+, Jan. 2006.
- [12] G. Yu and J.-M. Morel, "A fully affine invariant image comparison method," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, april 2009, pp. 1597 –1600.
- [13] R. Hess, "An open-source SIFT library," in *Proceedings of the international conference on Multimedia*, ser. MM '10. New York, NY, USA: ACM, 2010, pp. 1493–1496.
- [14] M. Piccardi, "Background subtraction techniques: a review," pp. 3099–3104, 2004, systems, Man and Cybernetics, 2004 IEEE International Conference on.
- [15] S. Herrero and J. Bescos, "Background subtraction techniques: Systematic evaluation and comparative analysis," in *Advanced Concepts for Intelligent Vision Systems*, ser. Lecture Notes in Computer Science, J. Blanc-Talon, W. Philips, D. Popescu, and P. Scheunders, Eds. Springer Berlin / Heidelberg, 2009, vol. 5807, pp. 33–42.
- [16] S. Brutzer, B. Hoferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, june 2011, pp. 1937–1944.