Robust Printing Quality Inspection on SHIBO Surfaces by Multiple Paired Pixel Consistency with modification

〇項 圣[†]，金子 俊一[†]，梁 棟[‡]

〇Sheng XIANG [†]，Shun'ichi KANEKO [†], Dong Liang [‡]

†: 北海道大学

‡： Nanjing University of Aeronautics and Astronautics

xiangsheng@hce.ist.hokudai.ac.jp

1 Abstract

A new method for printing quality inspection in SHIBO surfaces is proposed, called multiple paired pixel consistency (MPPC). In this method, we utilize orientation codes as basis. Since orientation codes can against the illumination changes. The proposed method consists of two major components, as illustrated in Fig. 1: training stage and detecting stage. Training stage is for making defect-free model based on multiple paired pixel consistency and orientation code difference. Detecting stage is to identify whether the target pixel match its model.

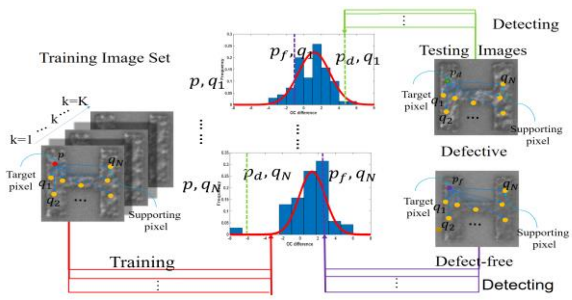

Fig.1 The framework of proposed method

2 MPPC defect-free model

MPPC is inspired by the previous work in [1], and it utilize orientation codes [2] as basis. The training stage can be divided to two steps. Select supporting pixels which have high consistency with target pixel. In other words, the selected supporting pixels have similar trend of change with target pixel. And make the model of each pixel pair by using a single Gaussian model to fit orientation code difference histogram.

2.1 Selecting the supporting pixel

Fig.1 shows the fundamental definitions of the image data. In this algorithm, we select the supporting pixel q by utilizing the Pearson's product moment correlation coefficient $\gamma(p,c)$. We select the pixels which are the highest N components of $\gamma(p,c)$ as the supporting pixels q_n for each target pixel p and record the position (u_n', v_n').

2.2 Modeling

We build the model for each pixel pair (p, q_n) by a single Gaussian model:

$$\Delta(\mathrm{p}, q_n) \sim N(\hat{\mu}_n, \hat{\sigma}_n^2) \quad (2)$$

where $\Delta(\mathrm{p}, q_n)$ is the OC difference between the target pixel and supporting pixel. Through the training, the $\hat{\mu}_n$ and $\hat{\sigma}_n^2$ can be determined. The defect-free model is a list consist of $[u_n', v_n', \hat{\mu}_n, \hat{\sigma}_n]$ for each target pixel p.

3 Defect detection

In the defect detection stage consists of two procedures: (1) to identify the normal/abnormal state of the pixel pair; (2) to determine target pixel is defective or defect-free pixel.

3.1 Pixel pair state

For each pixel pair (p, q_n), we utilize a binary function $\beta(q_n)$ for identifying the normal/abnormal state which can be estimated by following expression.

$$\beta(q_n)_{n=1,2,\ldots,N} = \begin{cases} 1 & \|\Delta(p,q_n) - \hat{\mu}_n\| \geq C * \hat{\sigma}_n \\ 0 & otherwise \end{cases} \quad (3)$$

where C is a constant. The constant C can be set from 1.0 to 3.0 to contain approximately an area of 68%-99% of its probability density function.

3.2 Decision function

In order to identify the defective/defect-free state of

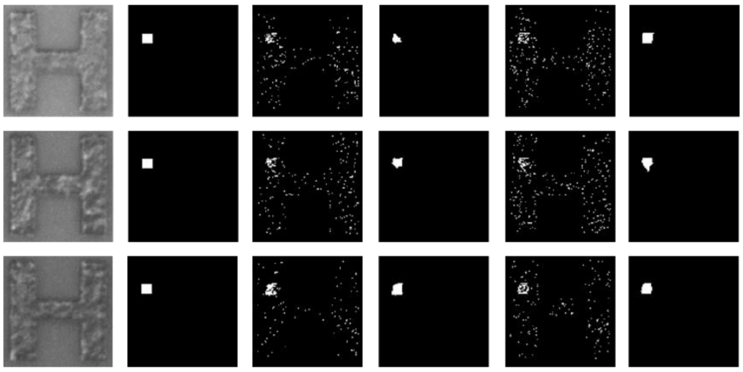

Fig.2 Some examples of detection result under different illumination conditions. From the left column, the test images, their ground truth images, the detection results by MPPC, filtered results, the detection results by MPPC with modification and filtered results. The first rows, from bright to dark illumination conditions images.

target pixel p, we define a decision function. The expression is shown in following.

$$\xi(p) = \frac{1}{N}\sum_{n=1}^{N}\beta(q_n) \quad (4)$$

where N is the total number of supporting pixels. if $\xi > T$, the target pixel can be considered as defective. And T is a global threshold that can be adjusted to achieve desired result.

4 Modification of MPPC

From the experimental result, we can see that it contains some holes in the defective squared area. We randomly select a pixel in the hole and analyze the statistical relationship between the target and its supporting pixels. From this, we can find a big gap between the fitted data and the actual measured data. Therefore, we proposed a modification method for MPPC modeling.

Here, we introduce Pearson's chi-square test to measure the goodness of fitting. If $\chi^2 \geq critical\ value$, we think the elemental model has no enough fitness. And then we must use the next candidate supporting pixels to replace it and compute χ^2 again in the new elemental model. This operation will be reused until the $\chi^2 < critical\ value$. We think this process can help us to get more perfect MPPC models which are closer to actual data. Therefore, we introduce this process to the training stage, making the trained MPPC models more effective and representative.

5 Experimental evaluation

We utilize the real production image from the real factory. Due to the difficulty to collect any real defects, in this we utilize some artificial defects. We cut out a sub image from the background, then add this sub image to the printed part. It is a simulation of missing print.

Fig.2 shows the inspection results. In these images, the white pixels are detected as defective pixels. Although there is a kind of severe fluctuation in illumination, we found the proposed method could detect the defect very similarly in size and shape. It shows the strong robustness of the orientation codes in the MPPC models. We can see that under the different illumination conditions, the performance of defect detection is very high and similar. And comparing original MPPC and MPPC with modification, we can see that MPPC with modification has obvious improvement.

6 Conclusion

MPPC is a robust printing quality inspection on SHIBO surface under dynamic illumination change. And comparing original MPPC and MPPC with modification, we can see that MPPC with modification has obvious improvement. For future works, we want to design an effective way to measure defect sizes and shapes.

Reference

[1] Liang, Dong, et al. "Co-occurrence probability-based pixel pairs background model for robust object detection in dynamic scenes." Pattern Recognition 48.4 (2015): 1374-1390.

[2] Farhan Ullah, and Shun'ichi Kaneko. "Using orientation codes for rotation-invariant template matching." Pattern recognition 37.2 (2004): 201-209.