Optik 125 (2014) 2671–2676

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

# Automatic defect recognition of TFT array process using gray level co-occurrence matrix

Shih-Wei Yang[a,*], Chern-Sheng Lin[b], Shir-Kuan Lin[a], Hsien-Te Chiang[b]

[a] Institute of Electrical and Control Engineering, National Chiao Tung University, Hsinchu, Taiwan
[b] Department of Automatic Control Engineering, Feng Chia University, Taichung, Taiwan

ARTICLE INFO

*Article history:*
Received 5 June 2013
Accepted 5 November 2013

*Keywords:*
Automatic optical inspection system
TFT array
Defect classifier

ABSTRACT

This study proposed an automatic optical inspection (AOI) system for detection of thin-film transistor (TFT) array defects. gray level co-occurrence matrix (GLCM) and MATLAB regionprops function were used to calculate 53 TFT array defect features, which were inputted into the neural network to train the defect classifier. The images to be inspected were compared with a standard image first, in order to judge whether the TFT array samples have defects. For defective images of a TFT array, the proposed defect classifier can successfully recognize five kinds of defects in the process. According to the experimental results, the defect recognition rate of proposed system is verified to be 83.3%, which can replace manual inspection and reduce the risks of false inspections due to long duration manual work. Moreover, the proposed AOI system can improve testing efficiency and reduce manufacturing costs.



## 1. Introduction

The TFT–LCD process can be divided into three stages: TFT array process, cell process, and module process [1–5]. These processes are very cumbersome and complex, and often require mass manual inspection to increase product yield. In order to avoid misjudgment caused by the human eye, automatic inspection techniques have been proposed for TFT–LCD cell process and module process [6–10]. Lin et al. proposed a genetic algorithm to determine optimal inspection parameters and recognize TFT–LCD dark points, bright points, and defective points [11]. A template matching technique was also developed to inspect various defects [12] in the TFT–LCD PI-coating process. Park et al. established a set of standards for various defects in TET–LCD in order to evaluate whether these defects can be perceived by the naked eye [13]. Fan et al. proposed a method to inspect TFT–LCD Mura defects. This method can separate Mura defects from the background through the prediction error sum of squares residuals and overcome non-uniform brightness or darkness [14].

However, the TFT array process is the first phase of the TFT–LCD. If the array process has any problems, future operations will be affected. Five common defects are exemplified. If fibers from a worker's clothes fall into the TFT array, fiber defect occurs. Any foreign matters in the exposure equipment that fall into the light resistance may cause particle defects after substrate coating and before exposure. If tightness between the pressure valve and oven of the equipment is not good, the chemical agent will drop into the TFT substrates and result in defects of pattern damage. Defects of pattern residual will occur if the developer concentration is not sufficient, and defects of pattern scratches are formed if the equipment roller and baffle have friction.

Thus, this study proposed an automatic defect recognition system for the TFT array process in order to solve the above problems. The image processing techniques [15–17] were applied to judge whether TFT array samples have defects. The features of the defective image samples were calculated by gray level co-occurrence matrix (GLCM), and then used to recognize the five common defects in the TFT array process through the defect classifier trained by the neural network, which serve as the basis for process correction.

## 2. Methods of automatic defect recognition of TFT array process

This study inspected five common defects in TFT array process: (1) fiber defects; (2) particle defects; (3) pattern damage; (4) pattern residual; and (5) pattern scratches. The actual images of the five defects are as shown in Fig. 1, and the defect recognition flow is illustrated in the following subsections.

### 2.1. Preliminary recognition of TFT array defect

In order to improve inspection speed, the proposed system first judges whether images to be inspected have defects, and then further recognizes images with possible defects. The images to be

* Corresponding author. Tel.: +886 3 5712121x54423.
*E-mail address:* swyang.nctu@msa.hinet.net (S.-W. Yang).

0030-4026/$ – see front matter © 2014 Elsevier GmbH. All rights reserved.
http://dx.doi.org/10.1016/j.ijleo.2013.11.070

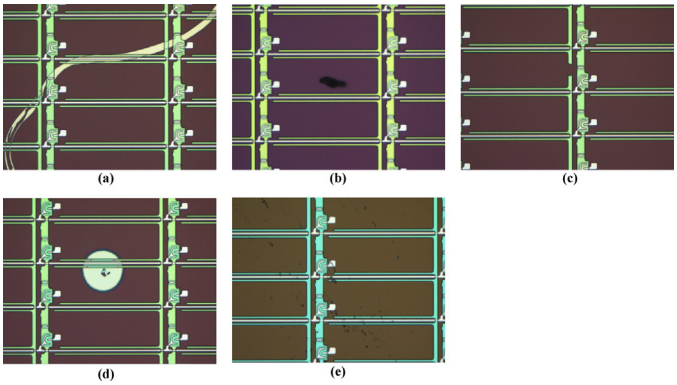


**Fig. 1.** Five common defects in TFT array process (a) fiber defect (b) particle defect (c) pattern damage (d) pattern residual (e) pattern scratch.

inspected are converted into gray scale images, and subtracted by a standard gray scale image of TFT array accepted products. If the TFT array to be inspected has no defect, the gray value of each pixel in the image subtraction results approaches to 0; otherwise, TFT array defects may cause higher gray values of the corresponding pixel in image subtraction results. The binary image subtraction results are analyzed. As shown in Fig. 2, the TFT array may have defects if the white block area exceeds the preset threshold. The defect will be recognized using the proposed defect classifier.

### 2.2. Feature extraction of TFT array defects using GLCM

TFT array images with defects are normalized to $320 \times 240$ images. The gray level co-occurrence matrix (GLCM), as proposed by Haralick et al. [18], was used to calculate the features of various defects, which can serve as basis for defect recognition. GLCM counts concurrent times of grey tone values of pixels on two relative positions in gray scale images under the preset conditions (such as: direction and distance). More specifically, $P(i, j|d, \theta)$ means concurrent times for pixel pair of $g(u, v)$ and $g(m, n)$ in an image, and their grey tone values are $i$ and $j$ respectively, when distance is $d$ and angle is $\theta$. In this paper, $\theta$ is set to 0°, 45°, 90° and 135° to calculate non-normalized GLCM, as shown in Fig. 3.

As shown in Fig. 3, the calculated GLCM can be further normalized to make each matrix element representing the occurrence probability of grey tone value of every pixel. Each element value in the matrix is divided by the sum of all element values in order to obtain a normalized GLCM:

$$P_{\text{nor}}(i, j|d, \theta) = \frac{P(i, j|d, \theta)}{N} \tag{1}$$

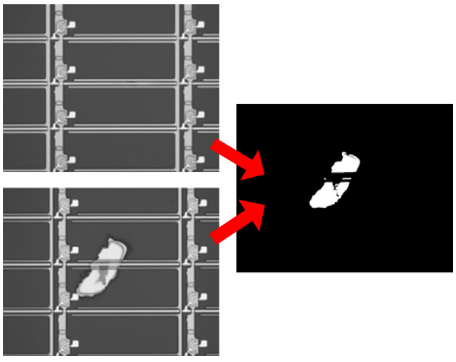

**Fig. 2.** TFT array defect after filtering background.

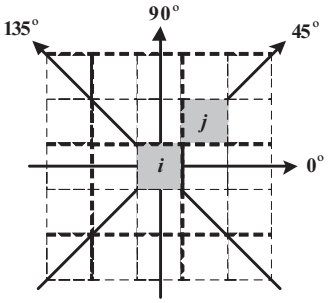


**Fig. 3.** Diagram of GLCM in four directions.

where $P_{\text{nor}}(i, j|d, \theta)$ is a normalized matrix element value. $N$ is total concurrence times of grey tone values $i$ and $j$, namely, the sum of all element values in the matrix.

After normalization of GLCM, this study calculates 11 features [18,19] at different angles (0°, 45°, 90° and 135°) in order to recognize various TFT array defects:$P_1$: Angular second moment

$$P_1 = \sum_i \sum_j \left[P_{\text{nor}}(i, j|d, \theta)\right]^2 \tag{2}$$

$P_2$: Contrast

$$P_2 = \sum_{n=0}^{Ng-1} n^2 \left[\sum_{i=1}^{Ng} \sum_{j=1}^{Ng} P_{\text{nor}}(i, j|d, \theta)\right], \quad |i - j| = n \tag{3}$$

$P_3$: Correlation

$$P_3 = \frac{\sum_i \sum_j (i \times j) P_{\text{nor}}(i, j|d, \theta) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{4}$$

$P_4$: Sum of squares

$$P_4 = \sum_i \sum_j (i - \mu)^2 P_{\text{nor}}(i, j|d, \theta) \tag{5}$$

$P_5$: Inverse difference moment

$$P_5 = \sum_i \sum_j \frac{P_{\text{nor}}(i, j|d, \theta)}{1 + (i - j)^2} \tag{6}$$

$P_6$: Sum average

$$P_6 = \sum_{i=2}^{2Ng} i P_{x+y}(i) \tag{7}$$

$P_7$: Sum entropy

$$P_7 = -\sum_{i=2}^{2Ng} P_{x+y}(i) \log \left[P_{x+y}(i)\right] \tag{8}$$

$P_8$: Total variance

$$P_8 = \sum_{i=2}^{2Ng} (i - P_7)^2 P_{x+y}(i) \tag{9}$$

$P_9$: Entropy

$$P_9 = -\sum_i \sum_j P_{\text{nor}}(i, j|d, \theta) \log \left[P_{\text{nor}}(i, j|d, \theta)\right] \tag{10}$$

$P_{10}$: Difference variance

$$P_{10} = \sum_{t=0}^{Ng-1} i^2 P_{x-y}(i) \tag{11}$$

$P_{11}$: Difference entropy

$$P_{11} = -\sum_{i=0}^{Ng-1} P_{x-y}(i) \log\left[P_{x-y}(i)\right] \tag{12}$$

where $Ng$ = (maximum grey scale value in image – minimum grey scale value in image) + 1, and also called grey tone value.

$$\mu_x = \sum_i \sum_j i \times P_{\text{nor}}(i,j|d,\theta), \quad \sigma_x = \sum_i \sum_j (i-\mu_x)^2 \times P_{\text{nor}}(i,j|d,\theta),$$

$$\mu_y = \sum_i \sum_j j \times P_{\text{nor}}(i,j|d,\theta), \quad \sigma_y = \sum_i \sum_j (i-\mu_y)^2 \times P_{\text{nor}}(i,j|d,\theta),$$

$$P_{x+y}(k) = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} P_{\text{nor}}(i,j|d,\theta), \quad k = 2,3,\ldots,2Ng,\ i+j=k,$$

$$P_{x-y}(k) = \sum_{i=1}^{Ng} \sum_{j=1}^{Ng} P_{\text{nor}}(i,j|d,\theta), \quad k = 0,1,\cdots,(Ng-1),\ |i-j|=k.$$

The GLCM at four angles (0°, 45°, 90° and 135°) respectively correspond to 11 features, totaling 44 features; each of which can be regarded as the feature of each TFT array image.

### 2.3. Design of defect classifier using a backward propagation neural network

The determined defect features in Section 2.2 are input as a multi-layer feed forward neural network to train a TFT array defect classifier for defect recognition. The Backward Propagation Algorithm (BP Algorithm) [20,21] is used as the neural network learning method. The BP Algorithm sends back the difference between the desired output and the actual output in order to update the weight of each node, through which errors between the desired output and actual output can be reduced.

In general, the BP update rule can be expressed by [21]:

$$\begin{aligned} w_{ij}^{\text{new}} &= w_{ij}^{\text{old}} + \Delta w_{ij} \\ \Delta w_{ij} &= \eta \delta_i x_j \end{aligned} \tag{13}$$

where $w_{ij}^{\text{new}}$ is the node weight after learning, $w_{ij}^{\text{old}}$ is the node weight before learning, $\Delta w_{ij}$ is the weight change, $\eta$ is a positive number, and called learning constant, $\delta_i$ is the learning signal, and $x_j$ is an external input or input of one processing element of the hidden layer.

This study uses the bipolar sigmoid function $a(f) = [2/(1+e^{-f})] - 1$ as the activation function, and by corresponding to Eq. (13), the learning signal $\delta_{oi}$ of the $i$th node in the output layer can be expressed as:

$$\delta_{oi} = \frac{1}{2}(1-y_i^2)[d_i - y_i] \tag{14}$$

where $y_i$ is the actual output of the $i$th node in the output layer, and $d_i$ is the corresponding desired output.

Likewise, by corresponding to Eq. (13), the learning signal $\delta_{hq}$ of the $q$th node in the hidden layer is expressed as:

$$\delta_{hq} = \frac{1}{2}(1-z_q^2)\sum_i \delta_{oi} w_{iq} \tag{15}$$

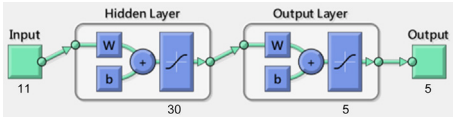


**Fig. 4.** Neural network architecture of using 11 features of defects.

where $z_q$ is the output of $q$th node in the hidden layer, and $w_{iq}$ is the weight from the $q$th node in the hidden layer to the $i$th node in the output layer.

Thus, the weight of each node in the network can be updated for learning according to Eqs. (13–15).

MATLAB was used to implement the above algorithm. Taking 11 defect features in GLCM at one angle as an example, the neural network is set to have 11 inputs, where one hidden layer has 30 nodes and 5 outputs (5 kinds of tested defects). The corresponding neural network architecture is as shown in Fig. 4. If the root mean square error of all actual outputs and desired outputs is smaller than the preset threshold, the TFT array defect classifier has completed learning and can be used for defect recognition.

## 3. Experimental results and discussion

### 3.1. Experimental framework of the proposed system

The proposed TFT array defect recognition system was divided into electric control and vision analysis. The inspection system framework is as shown in Fig. 5.

TFT array samples were placed on the linear motor, and positioned and moved using Programmable Logic Controller (PLC). All samples were moved under the camera for inspection. High speed cameras extracted the images and sent them to computers for comparison with the standard images. If they were defective samples, the system would use the defect classifier, as designed in Section 2, for defect recognition, and record the defect information. According to the TFT array defect recognition result, this system stored the classified defective images to the database. The continuously expanded database could enhance neural network training results, thus, system recognition could be more accurate.

This study also used MATLAB regionprops function to analyze images with five TFT array defects. This function can calculate the geometric features of selected images. The selected 9 geometric features are as shown in Table 1.

The geometric features were input of a neural network to train the classifier for recognition of the five defects. The recognition

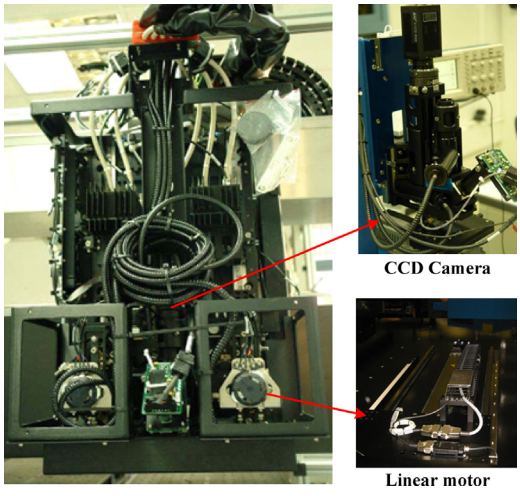


**Fig. 5.** Framework of the proposed system.

**Table 1**
The selected geometry features of MATLAB regionprops function.

| Geometry features | Description |
|---|---|
| Area | Total number of pixels of an image to be inspected |
| Perimeter | Total number of edge points of an image to be inspected |
| Density | Average gray value of all pixels |
| Centroid | Image central coordinates |
| Major radius | Maximum distance between edge and central point |
| Minor radius | Minimum distance between edge and central point |
| Radius ratio | The ratio of major radius and minor radius |
| Filled area | Area of hole of an image to be inspected |
| Roundness | $(\text{Perimeter})^2/(4\pi \times \text{area})$ |

results of regionprops function were compared with the GLCM recognition results. Finally, the features of the two methods were integrated in order to obtain a better defect recognition rate and enhance validity. The complete inspection process flow is as shown in Fig. 6.

### 3.2. Defect recognition results and discussion

This study collected 20 sets of fiber defect samples (Class 1), 41 sets of particle defect samples (Class 2), 33 sets of pattern damage samples (Class 3), 44 sets of pattern residual samples (Class 4), and 30 sets of pattern scratch samples (Class 5), totaling 168 sets of TFT array samples for defect recognition. Seven types of classifiers, with different conditions, are used in the experiment to analyze case recognition results.

Case 1: Use the features of GLCM with $\theta = 0^\circ$ to recognize TFT array defects.
Case 2: Use the features of GLCM with $\theta = 45^\circ$ to recognize TFT array defects.
Case 3: Use the features of GLCM with $\theta = 90^\circ$ to recognize TFT array defects.
Case 4: Use the features of GLCM with $\theta = 135^\circ$ to recognize TFT array defects.
Case 5: Use the features of GLCM at four angles to recognize TFT array defects.
Case 6: Use the features of MATLAB regionprops function to recognize TFT array defects.
Case 7: Use the features of GLCM at four angles and MATLAB regionprops function to recognize TFT array defects.

In this study, recognition rate was used to evaluate recognition effect. Recognition rate $R$ can be defined as:

$$R = \frac{\text{amount of correctly detected samples}}{\text{amount of all samples}} \times 100\% \quad (16)$$

Fig. 7 shows the corresponding Confusion Matrix and Receiver Operating Characteristic (ROC) curve of each case. The recognition rate of Case 1 is 69.6%, Case 2 is 70.2%, Case 3 is 70.8%, Case 4 is 69.6%, Case 5 is 78.6%, Case 6 is 73.2%, and Case7 is 83.3%. Note that the recognition rate is better when the ROC curve is concentrated in the left upper.

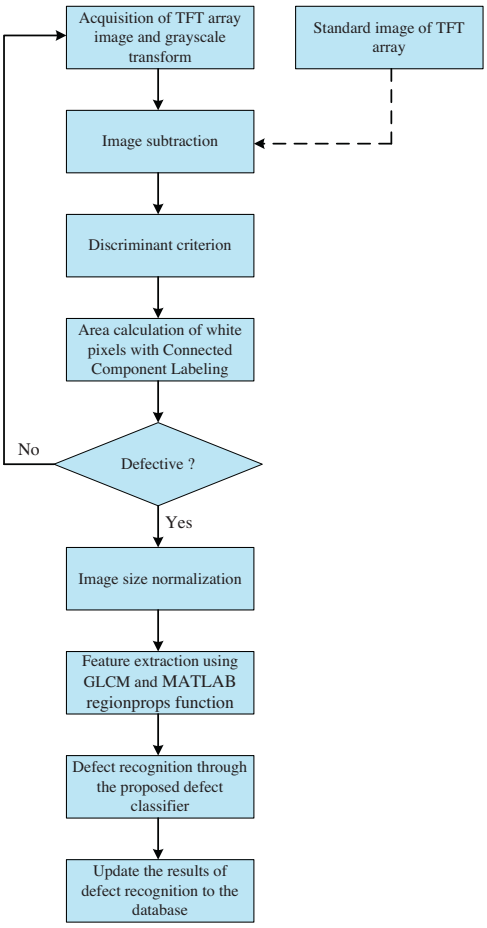


**Fig. 6.** Flowchart of the proposed defect recognition method.

The detailed experimental results are as shown in Table 2. As seen, the defect recognition rate of Case 1 and Case 4 is about 70%. In other words, the recognition effect using GLCM at four angels as features has no significant difference. However, the recognition rate of Case 5 is increased by 78.6%, which reveals that GLCM at four angles are all important references in the recognition of the five defects, and all should be considered in recognition.

In Case 6, MATLAB regionprops function are used to calculate the features for defect recognition. The recognition rate is 73.2%, which is better than GLCM at a single angle. In Case 7, the features of GLCM at four angles and MATLAB regionprops function are used to recognize defects. The recognition rate is 83.3%, which is the best among the cases. As more features have significant discriminations, the effect of the trained defect classifiers is better, and the recognition rate is better. However, as excessive features may affect

**Table 2**
The defect recognition rate of all cases.

| Defect type | Amount of samples | Amount of inspected samples with defects | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | Case 1 | Case 2 | Case 3 | Case 4 | Case 5 | Case6 | Case 7 |
| Fiber defect | 20 | 14 | 13 | 7 | 4 | 17 | 14 | 16 |
| Particle defect | 41 | 22 | 24 | 33 | 30 | 35 | 26 | 32 |
| Pattern damage | 33 | 26 | 25 | 27 | 24 | 28 | 30 | 29 |
| Pattern residual | 44 | 27 | 29 | 28 | 33 | 24 | 31 | 34 |
| Pattern scratch | 30 | 28 | 27 | 24 | 26 | 28 | 22 | 29 |
| Total | 168 | 117 | 118 | 119 | 117 | 132 | 123 | 140 |
| Recognition rate (%) | – | 69.6 | 70.2 | 70.8 | 69.6 | 78.6 | 73.2 | 83.3 |

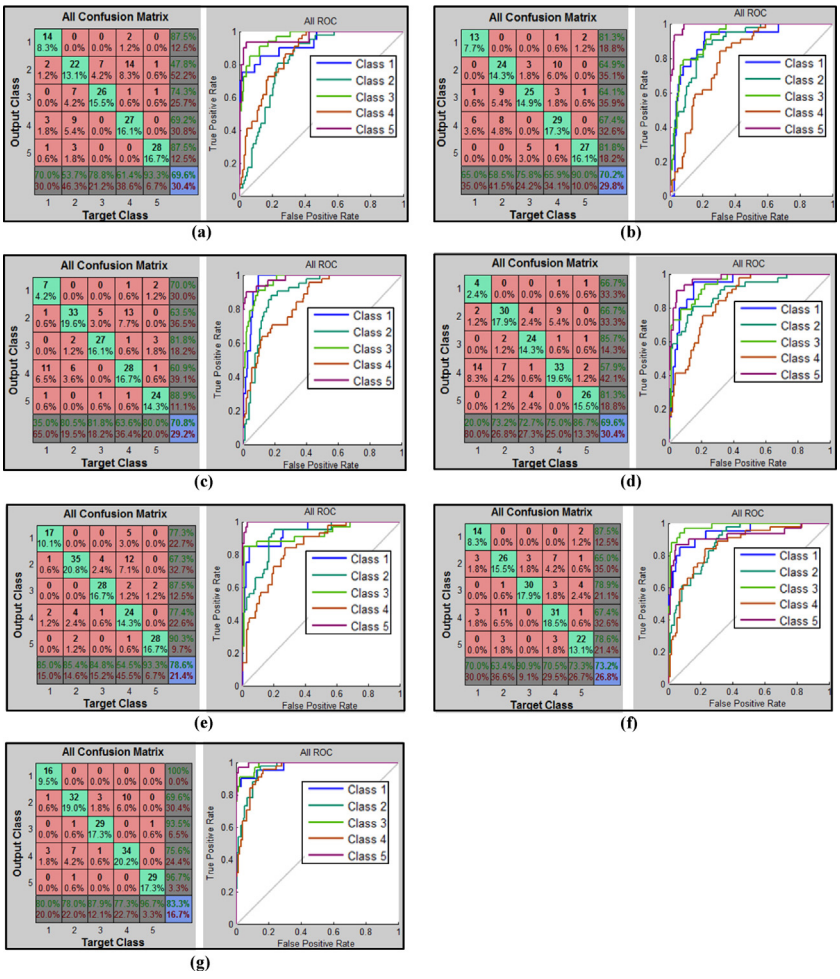


**Fig. 7.** Corresponding confusion matrix and ROC curve of 7 cases. (a) Case 1 (b) Case 2 (c) Case 3 (d) Case 4 (e) Case 5 (f) Case 6 (g) Case 7.

inspection speed, features with a significant discrimination should be carefully selected. It is noted that when the defect type changes or type of TFT array samples change, the proposed defect classifier must be retrained in order to adapt to new defects or samples. This is also the system inspection limitation.

## 4. Conclusions

This study proposed an automatic defect recognition system for the defect detection of TFT array process. GLCM and MATLAB regionprops function were used to calculate 53 TFT array defect features, which were input into a neural network to train the defect classifier. High speed cameras extracted the images, which were sent back to the computers for comparison with the standard images. For the defective TFT array samples, the defects were recognized by the proposed defect classifier. The experimental results suggested that the recognition rate of the proposed algorithm reached 83.3%, which meets panel industry inspection standards, and can replace current process equipment to reduce manual inspection.

## Acknowledgement

This work was supported by the National Science Council, under grant No. NSC 102-2221-E-009-064 and grant No. NSC 101-2221-E-035-039-MY2.

## References

[1] C.S. Lin, S.J. Shih, A.T. Lu, S.S. Hung, C.C. Chiu, The quality improvement of PI coating process of TFT–LCD panels with Taguchi methods, Optik 123 (8) (2012) 703–710.
[2] B. Jeong, S.W. Kim, Y.J. Lee, An assembly scheduler for TFT–LCD manufacturing, Comput. Ind. Eng. 41 (1) (2001) 37–58.
[3] C.S. Lin, S.C. Chang, Y.L. Lay, M.S. Yeh, C.C. Lin, A novel positioning method for optical automatic inspection of an LCD assembly process, Optik 121 (12) (2010) 1089–1095.
[4] E. Kaneko, A new fabrication technology for very-large-area TFT–LCDs, Displays 14 (2) (1993) 125–130.
[5] S. Lin, S.C. Lin, J.M. Yu, Modeling and characterization of TFT–LCD display module for LED projector, Optik 122 (17) (2011) 1538–1543.
[6] K.L. Hsieh, The application of clustering analysis for the critical areas on TFT–LCD panel, Expert Syst. Appl. 34 (2) (2008) 952–957.
[7] C.J. Lu, D.M. Tsai, Independent component analysis-based defect detection in patterned liquid crystal display surfaces, Image Vision Comput. 26 (7) (2008) 955–970.
[8] K. Amano, S. Yoshimoto, M. Miyatake, T. Hirayama, Basic investigation of noncontact transportation system for large TFT–LCD glass sheet used in CCD inspection section, Precis. Eng. 35 (1) (2011) 58–64.
[9] D.M. Tsai, S.C. Lai, Defect detection in periodically patterned surfaces using independent component analysis, Pattern Recogn. 41 (9) (2008) 2812–2832.
[10] Y.G. Yoon, S.L. Lee, C.W. Chung, S.H. Kim, An effective defect inspection system for polarized film images using image segmentation and template matching techniques, Comput. Ind. Eng. 55 (3) (2008) 567–583.
[11] C.S. Lin, Y.C. Liao, Y.L. Lay, K.C. Lee, M.S. Yeh, High-speed TFT–LCD defect detection system with genetic algorithm, Assembly Autom. 28 (1) (2008) 69–76.
[12] C.S. Lin, J. Kuo, C.C. Lin, Y.L. Lay, H.J. Shei, Automatic inspection and strategy for surface defects in PI coating process of TFT–LCD panels, Assembly Autom. 31 (3) (2011) 244–250.
[13] N.K. Park, S.I. Yoo, Evaluation of TFT–LCD defects based on human visual perception, Displays 30 (1) (2009) 1–16.

[14] S.K.S. Fan, Y.C. Chuang, Automatic detection of Mura defect in TFT–LCD based on regression diagnostics, Pattern Recogn. Lett. 31 (15) (2010) 2397–2404.
[15] C.S. Lin, J. Cho, Y.L. Lay, C.H. Lin, N. Lin, The auto-measurement of the gap of LCD glass plates using sub-pixel accuracy estimation, Optik 117 (8) (2006) 349–354.
[16] S.W. Yang, C.S. Lin, S.K. Lin, 3D surface profile measurement of unsymmetrical microstructure using Fizeau interferometric microscope, Opt. Laser Eng. 51 (4) (2013) 348–357.
[17] C.S. Lin, C.H. Lin, C.C. Lin, M.S. Yeh, Three-dimensional profile measurement of small lens using subpixel localization with color grating, Optik 121 (23) (2010) 2122–2127.
[18] R.M. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, IEEE Trans. Syst. Man Cyb. 3 (1973) 610–621.
[19] R.N. Sutton, E.L. Hall, Texture measures for automatic classification of pulmonary disease, IEEE Trans. Comput. 21 (1972) 667–676.
[20] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (1986) 533–536.
[21] C.T. Lin, C.S.G. Lee, Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems, Prentice Hall, New Jersey, 1996.