



Automatic detection of Mura defect in TFT-LCD based on regression diagnostics

Shu-Kai S. Fan^{*}, Yu-Chiang Chuang

Department of Industrial Engineering and Management, Yuan Ze University, 135, Yuan-Tung Rd., Chung-Li City, Taoyuan County 320, Taiwan, ROC

ARTICLE INFO

Article history:

Received 11 December 2008

Available online 27 July 2010

Communicated by H. Sako

Keywords:

TFT-LCD manufacturing

Mura defect detection

Regression diagnostics

PRESS residuals

ABSTRACT

This paper proposes a computationally efficient Mura defect detection method that is constructed based on regression diagnostics using the prediction error sum of squares (PRESS) residuals and an image estimation procedure for automatic Mura inspection of thin film transistor liquid crystal display (TFT-LCD) devices. The gray-level data of the input image is estimated by a linear model and then the PRESS residuals are calculated to filter Mura regions out. After image dilation, the threshold value is determined for detecting the non-uniform brightness or darkness areas in TFT-LCD by means of examining every pixel in the image. The experimental results of several test images returned by using the proposed method and an existing method in the literature are used to evaluate the performance of effectiveness and efficiency for Mura detection. It has been found that the method proposed in this paper is very swift in processing time and also returns competitive Mura detection performance in comparison to the existing method.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction and background

Large-sized thin film transistor liquid crystal display (TFT-LCD) devices have been applied to many industrial applications in recent years. To assure the product quality and maintain the competitiveness in the market of the TFT-LCD display, one of the most important processes is to visually inspect potential defects which might occur during TFT-LCD manufacturing. The most common defect inspection is done by human visual inspection, but it cannot generate consistent inspection outputs due to restrictions of human vision and human subjectivity. In addition, some defects are not easy to be detected by human-eye inspection especially in large-sized TFT-LCD, such as Mura defects. Mura, a Japanese word for blemish, denotes a local lightness variation without a clear contour on a uniformly-produced surface, resulting in an unpleasant sensation to human vision. Intense Mura is usually called just a defect, and a Mura too weak to perceive is called a latent image so the contrast of Mura lies between them (Taniguchi et al., 2006). Mura appears in the form of a low-contrast region without having clear edges in the surface, so inspectors need to “repetitively” examine potential Mura areas from different viewing and light angles. For this token, automatic Mura defect detection becomes a critical process to maintain quality and reduce production cost in LCD manufacturing.

Typically, Mura defects appear as low-contrast, non-uniform brightness regions in TFT-LCD, and they are larger than one pixel in usual (Pratt et al., 1998). In Fig. 1, there are several types of Mura

defects in the TFT-LCD, and they are defined as spot-Mura, line-Mura and region-Mura, respectively. Those defects are not easy to be detected by human inspection or simple thresholding method due to the human subjectivity and non-uniform illumination conditions. In this paper, an automatic inspection method is proposed for detecting Mura defects in TFT-LCD manufacturing.

In the literature, a large number of algorithms have been proposed for defect detection on TFT-LCD. For example, Lu and Tsai (2005) proposed a global approach for automatic visual inspection of macro defects on TFT-LCD. The method was designed based on a global image reconstruction scheme using the singular value decomposition (SVD). Taking the image as a matrix of pixels, different degrees of the textured images would result in the different singular values on the decomposed diagonal matrix. By setting the proper singular values for reconstructing the image matrix without the selected patterns of the textured image, this method could preserve the anomalies in the restored image. Jiang et al. (2005) proposed a method which attempted to establish a more objective, automatic detection process for Mura in TFT-LCD. The first step was to perform the analysis of variance of luminance on different areas for detecting the existence of Mura defects. The second step involved plotting exponentially weighted moving average (EWMA) control charts for obtaining the location of the defects. Later, Lu and Tsai (2008) proposed a new image reconstruction procedure using independent component analysis (ICA) for defect detection in patterned liquid crystal display surfaces. ICA was employed to determine the de-mixing matrix and then reconstruct the TFT-LCD image under inspection. In the method, first ICA was applied to perform faultless image training for determining the de-mixing matrix and the corresponding independent

^{*} Corresponding author. Tel.: +886 3 4638800x2510; fax: +886 3 4638907.

E-mail address: simonfan@saturn.yzu.edu.tw (S.-K.S. Fan).

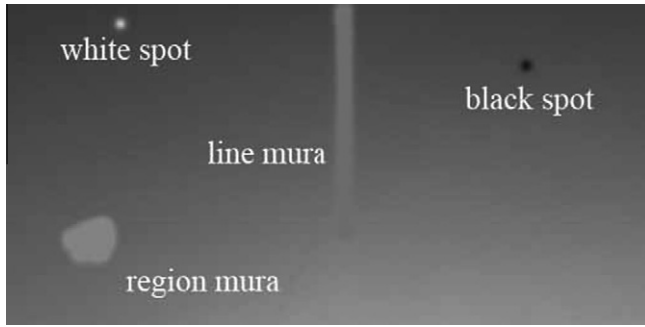


Fig. 1. Illustration of line, spot and region-Mura.

components (ICs), and then the reformed de-mixing matrix evaluated from ICs and the training image were used to reconstruct the TFT-LCD image under inspection. This method could effectively remove the global structural pattern and preserve only local anomalies. Chen and Chou (2008) proposed two Mura detection methods which were discrete cosine transform (DCT) and discrete wavelet transform (DWT). These two methods transformed gray scale spatial signals into the frequency domain and wavelet domain for preserving background signals, enhancing Mura features separately. Subsequently, these two methods transformed the inversely frequency and wavelet signals to spatial signals, and inspection images were processed by the Otsu's method (1979) for filtering out Mura defects. A very limited number of journal paper publications on Mura defect detection are available in the literature. We have noticed two important works: Lee and Yoo (2004) and Taniguchi et al. (2006). In essence, Lee and Yoo's work is more closely related to the proposed method, so their method will be implemented in this paper to calibrate the proposed method. The inspection procedure proposed by Lee and Yoo is briefly reviewed as follows. First, each input image which is the TFT-LCD panel under test is divided into non-overlapping windows of size $(W \times H)$ for local processing. The objective of local processing is to eliminate the non-uniformity in the input image, and then those windows are merged into a single binary image with their original positions. The merged image is then processed by median filtering (Gonzalez and Woods, 2002) for removing noise and improving the initial inspection results. The second step is to reconstruct the inspection image while the non-uniformity region is not taken into account. The inspection result is presented by setting a threshold, according to Niblack's method (1986), on the residual of the input image and the reconstruction image. The estimation function is defined by $f^{(d)}(x, y) = \sum_{m+n \leq d} a_{mn} x^m y^n$. The image data is approximated by a bivariate polynomial model with order d , and $f^{(d)}(x, y)$ is the estimated intensity value at position (x, y) . The estimation is accomplished by using the least-squares method; that is, the model parameters are estimated by minimizing the sum of the squared residuals. The background surface estimation algorithm is executed in the following steps as the size of region-Mura is upper-bounded by 20% of the window size:

Step 1: For each pixel p , remove the pixel from the inspection image and then determine the polynomial model with order 2 for computing the diagnostic measure:

$$J(p) = \frac{1}{WH-1} \sum_{0 \leq x < W} \sum_{0 \leq y < H} |z_{xy} - f^2(x, y)|. \quad (1)$$

Step 2: Earmark the 20% pixels which have smaller values of $J(p)$ as white and the others as black in constructing binary image.

Step 3: Apply median filtering to the binary image.

Step 4: Estimate the image without outlier pixels which correspond to the white pixels in Step 3 by using the polynomial model with order four.

Step 5: The resulting image is filtered by the Niblack's threshold method with $T = 2$.

For interested readers, additional references of Mura detection include Wang and Ma (2006), Kim et al. (2008) for further reading. The development of an "easy-to-implement" and "computationally efficient" method for detecting TFT-LCD Mura defects is the main goal of this paper. The gray-level data of the image under inspection is first fitted by a linear model, and then Mura areas are extracted by using regression diagnostics and setting a threshold value. Finally, the Mura pixels are sieved pertaining to the information of the Mura areas that were previously determined. Section 2 describes the proposed Mura detection method. In Section 3, the parameter setting issue of the proposed method is analyzed, and the experimental results between the Lee and Yoo's (2004) method and the proposed method are illustrated by means of different types of Mura defects. The conclusion is given in Section 4.

2. Mura defect detection method based on linear regression diagnostics

The Mura defect of gray-level data represents non-uniform brightness or darkness regions in the image. In Fig. 2, a test example is illustrated by a 3-D plot of the gray-level data where the region-Mura appears as a peak extruding from a smooth plane. In order to filter out the region-Mura area based on the illustration in Fig. 2(b), it appears logical to fit the gray-level data of the input image by a low-order regression model (Fig. 2(c)) such that the residuals, the difference between the actual and fitted gray-level data, can be computed as the threshold. The surface of the gray-level data should exhibit a statistical fit to a first- or second-order polynomial if the region-Mura, treated as outliers, can be effectively removed. In doing so, the Mura defect can be determined by image subtraction. The first step is to fit the gray-level data and then provide the regression diagnostic for screening out the outlier area. In Fig. 2(d), the outlier areas of the gray-level data are removed according to some regression diagnostic. Once the non-Mura gray-level image is retrieved, Mura pixels are filtered by using another proposed threshold for residuals.

An overview of the proposed method for detecting region-Mura is graphically displayed in Fig. 3. The digital input image of TFT-LCD panels is loaded from automatic inspection camera in TFT-LCD manufacture. The captured image is transformed into the gray-level image with image size $W \times H$, denoting width and height, respectively. To accelerate computation time, the whole gray-level image is first partitioned into several sample region images with the size $W_{sr} \times H_{sr}$ and the average gray-level of each sample image is treated as the sample point instead of evaluating all pixels. Basically, there are 4 steps in the proposed method. In step 1, the gray-level data are estimated by the linear regression model (Fig. 3(b)). In step 2, the regression diagnostic of the prediction error sum of squares (PRESS) residuals is used to detect the outlier. The potential Mura area is screened by setting the first threshold to the PRESS residuals and then the outliers being detected are earmarked in the positions of the original image. A binary image to pinpoint the Mura area (Fig. 3(c)) is created. Note that the PRESS of regression diagnostic proposed by Allen (1971, 1974) is a quite useful measure for residual scaling. The PRESS residual uses each possible subset of observations minus one as the estimation data set, and every observation is used in turn to form the prediction data set. In step 3, the binary image is then processed by the morphological region dilation (Fig. 3(d)) for ensuring that the non-Mura region contains only defect-free image. In step 4, Mura pixels are extracted (Fig. 3(e)) by setting the second threshold to residuals between the original and estimated gray-level data. The

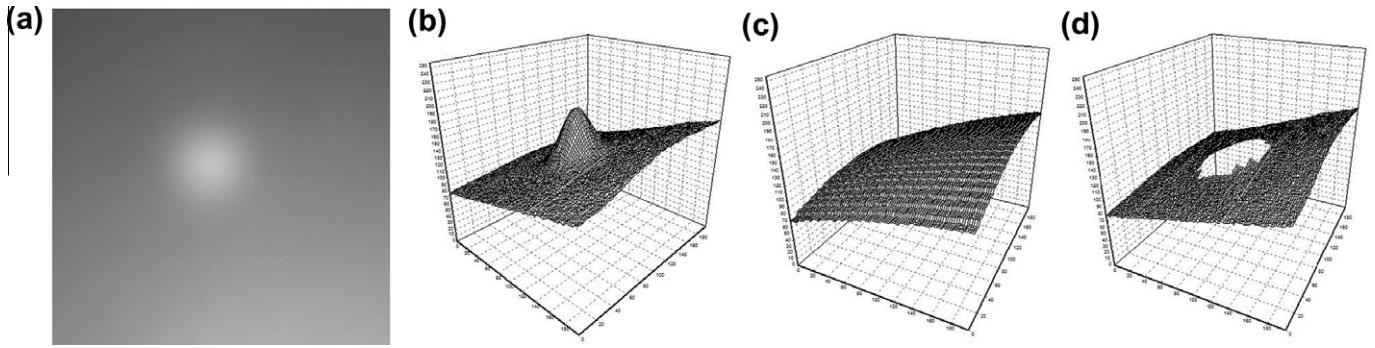


Fig. 2. Illustration of the Mura regions in the gray-level data in a 3-D representation: (a) the gray-level image under inspection, (b) 3-D plot of the gray-level data of the original image, (c) 3-D plot of the estimated gray-level data of the original image, and (d) 3-D plot with outlier removal.

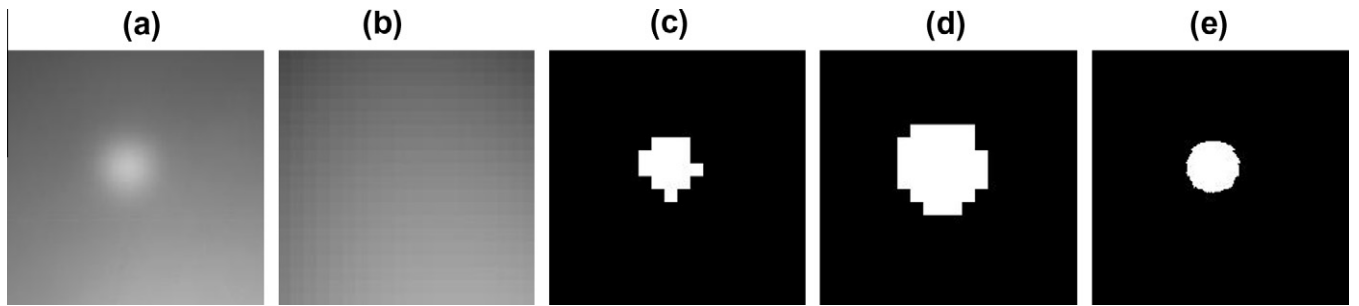


Fig. 3. Four steps in the proposed method: (a) original image, (b) estimated image, (c) outlier region image, (d) Mura region dilation, and (e) Mura image.

second threshold is computed based upon the mean and standard deviation of non-Mura residuals computed in step 3. For a detailed account, these four steps are elaborated in following sections.

2.1. Construction of estimated image

The first step is to construct the estimated (or fitted) image by a linear regression model. In the linear model, x_1 represents the position of the pixel in the x axis, x_2 represents the position of the pixel in the y axis, and y is the gray-level value of the pixel on position (x_1, x_2) , as defined by

$$y = f(x_1, x_2), \quad (2)$$

$$x_1 = x_{\text{pos}}, \quad x_2 = y_{\text{pos}}. \quad (3)$$

Thus, the equation of the linear model is defined by

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \varepsilon, \quad (4)$$

where ε is the error term; β_0 is the intercept; β_1 and β_2 are the parameters for the linear terms x_1 and x_2 ; β_{11} and β_{22} are the parameters for the pure quadratic terms x_1^2 and x_2^2 ; β_{12} is the parameter for the mixed quadratic term $x_1 x_2$. To employ the linear regression model for calculating the residuals of the image's gray-level values, the hat matrix \mathbf{H} can be first computed, as given by

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T, \quad (5)$$

where \mathbf{X} is the design matrix $[\mathbf{1}, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_1^2, \mathbf{x}_2^2, \mathbf{x}_1 \mathbf{x}_2]_{n \times 6}$ in the linear model where n is the number of sample points. In here, $\mathbf{1}$ is a vector of 1, \mathbf{x}_1 is a vector of x_1 , \mathbf{x}_2 is a vector of x_2 , \mathbf{x}_1^2 is a vector of x_1^2 , \mathbf{x}_2^2 is a vector of x_2^2 , and $\mathbf{x}_1 \mathbf{x}_2$ is a vector of $x_1 x_2$. The hat matrix has several useful properties. It is symmetric ($\mathbf{H}^T = \mathbf{H}$) and idempotent ($\mathbf{H}\mathbf{H} = \mathbf{H}$). Similarly, the matrix $\mathbf{I} - \mathbf{H}$ is also symmetric and idempotent. The estimated image is $\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}$. The hat matrix \mathbf{H} can be directly computed without having to explicitly construct the entire regression model, thus making the image estimation procedure extremely effi-

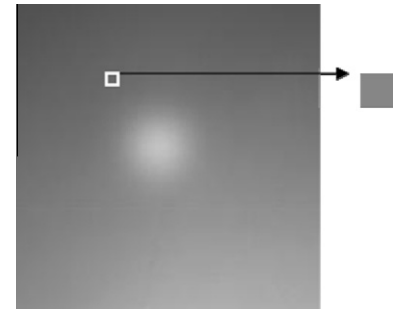


Fig. 4. Illustration of sample region.

cient. The hat matrix \mathbf{H} is a constant matrix if the image to be estimated has a fixed size and is split using the same window size since \mathbf{H} is only concerned with position (x_1, x_2) .

The original image is treated as a single window and the local gray-level approximation is accomplished according to the sample region size $W_{sr} \times H_{sr}$. For the sake of computation efficiency and noise reduction, the original image is estimated by using sample information (i.e., average gray-level) instead of going over all the pixels. The sample is a square region without overlapping as illustrated in Fig. 4. The local coordinate system \mathbf{X} is defined as follows:

$$(1, x_1, x_2, x_1^2, x_2^2, x_1 x_2) = (1, i, j, i^2, j^2, i \times j), \quad (6)$$

$$i \in [0, 1, 2, \dots, W/W_{sr}], \quad j \in [0, 1, 2, \dots, H/H_{sr}], \quad (7)$$

where W, H, W_{sr} and H_{sr} are the heights and widths of the image size for Mura inspection and sample regions for image estimation. Note again that each average gray-level value of the sample region is treated as a single sample point for the estimation of the image under inspection. Namely, every pixel in the same sample region shares the same estimated gray-level value.

2.2. Outlier area detection

To filter out the Mura defects in the estimated image, the second step is to conduct regression diagnostics. The PRESS residuals are used for detecting the outlier area. The vector of residuals \mathbf{e} is equal to \mathbf{y} minus $\hat{\mathbf{y}}$, which is defined by $\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}$, $\mathbf{e} = (\mathbf{I} - \mathbf{H})\mathbf{y}$. The i th PRESS residuals $e_{(i)}$ is expressed by

$$e_{(i)} = e_i / (1 - h_{ii}), \quad i = 1, 2, \dots, \text{num_sample}, \quad (8)$$

where e_i is the i th element in the vector of residuals, and h_{ii} is the i th diagonal element of the hat matrix in (5) (Montgomery and Peck, 1992). The decision rule is that if the absolute value of PRESS residuals is greater than the threshold, then that sample region is earmarked as an outlier area in the binary image (see Fig. 3(c)). The proposed threshold is a multiple of the mean square error (MSE) of the PRESS residuals as follows:

MSE = sum of square error/degrees of freedom

$$= \sum_{i=1}^n e_i^2 / (n - n_p), \quad (9)$$

$$\text{binary}(x_1, x_2) = \begin{cases} 1, & |e_{(i)}| > (c_1 \times \sqrt{\text{MSE}}), \\ 0, & |e_{(i)}| \leq (c_1 \times \sqrt{\text{MSE}}), \end{cases} \quad (10)$$

where n and n_p are the numbers of the sample regions and unknown parameters in the linear model in (4); c_1 is the proposed constant value that needs to be determined by the user. The outlier areas are set to be white with gray-level value 1 and the remaining areas are treated as background in black with gray-level value 0 that makes the binary image in the original positions.

2.3. Mura area dilation and Mura pixel detection

An extra manipulation required for the outlier areas is the image dilation where the method adopted here is the eight-neighbor region dilation. This idea is inspired by dilation in mathematical morphology (Gonzalez and Woods, 2002). When an area is considered an outlier, its eight-neighbor regions are set to be outliers as well. This manipulation tries to cover the Mura areas as thoroughly as possible. As such, the remaining areas contain almost purely the non-Mura pixels, thus making MSE a more accurate estimate of the variance of the defect-free gray-level values.

The last step is to decide on the second threshold for filtering Mura pixels. The idea of the threshold is inspired from statistical process control (SPC) in the field of quality control (QC), and it is set equal to $\mu + c_2 \times \sigma$, where c_2 is the constant to be pre-specified by the user. The mean μ and the standard deviation σ are calculated over all the differences of gray-level values (i.e., residuals) of “non-Mura pixels” between the original and estimated images. There is only the upper bound opted to filter the region-Mura pixel since the minimum of the difference expects to be zero. Next, the differences of all the gray-level values (including Mura and non-Mura pixels) between original and estimated images are filtered by using the proposed threshold. If the difference is greater than the threshold, then the pixel is classified as a Mura pixel.

3. Experimental study of the proposed Mura inspection method

In this section, an experimental study for evaluating the proposed Mura inspection method is conducted. First of all, the experimental results of several Mura images under different parameter settings are discussed. Then, the performance of the proposed method is compared to Lee and Yoo's (2004) method in terms of normal and Mura images. The original image under inspection is of size 200×200 and the sample region of size 10×10 without

overlapping. Note that the sample mean of gray-level values of all pixels in the 10×10 sample image is treated as a single sample point for image estimation. Regarding parameter setting, two scenarios of the experimental results are examined. The first one is to test various values of the parameter c_1 in the first threshold on the PRESS residual. The second one is to test various values of the parameter c_2 in the second threshold on the Mura pixel detection. Normal images indicate the input images without Mura defects. However, the images are sometimes examined under non-uniform illumination conditions, which will much complicate the inspection of Mura defects and then mislead the inspection method to yield incorrect decisions. The two above-mentioned parameter settings in these two scenarios are used to investigate the sensitivity of the proposed method under different illumination conditions.

3.1. Different threshold values for Mura area detection

To detect Mura areas between the estimated and original images, the PRESS residuals are calculated for each sample region. If the absolute value of i th PRESS residual ($e_{(i)}$) is greater than the threshold, the sample region corresponding to that PRESS residual will be treated as an outlier area as shown in Fig. 5. In the first scenario, the threshold parameter c_1 in (10) is set to 0.5, 1.0 and 1.5, respectively, for evaluating the sensitivity of the proposed method in Mura area detection. The detection results of different settings on c_1 are exhibited in Table 1. As the threshold parameter is set to 0.5, the outlier areas that result contain many misclassified regions. When the threshold parameter is set to 1.5, the outlier area is somewhat smaller than the actual area of the Mura defect. For the non-Mura image reconstruction, it is suggested to filter all potential Mura areas out even if there are some misclassified regions. Therefore, setting the threshold parameter to 1.0 in the first scenario is recommended.

Upon filtering out the outlier areas, the outlier area image is extended by processing the eight-neighbor region dilation. As illustrated in Fig. 6, the resulting image represents the outlier area image after applying eight-neighbor region dilation. This process attempts to ensure that all potential region-Mura pixels are covered by the Mura areas (i.e., white areas in Fig. 6).

3.2. Different threshold values for Mura pixel detection

The second scenario is to set various values of the threshold parameter c_2 for Mura pixel detection. The parameter c_2 is set to 2.0, 2.5 and 3.0 for the comparison purpose. Tables 2 and 3 list the resulting images obtained by using different values of c_2 on normal and Mura images, respectively. First, the normal image (without Mura defects) is examined, and the ideal result is that there is no Mura pixel found. The best result is obtained as c_2 is set to 3.0. If c_2 is set to 2.0 or 2.5, some of the outputs are affected by non-uniform illumination and then generate unnecessary noises. In Table 3, while detecting the Mura pixels, the threshold parameter set to 3.0 could produce better results with less misclassified pixels than the other two settings. Under such a setting, the proposed method could detect the spot and line Mura effectively, but the result of the gravity-Mura image using the threshold parameter c_2 of 3.0 is not as satisfactory as expected. The threshold parameter c_2 of 2.0 or even smaller is strongly suggested to increase the detection sensitivity for the gravity Mura case. The proposed method is coded by Borland C++ Builder 6.0. On average, it takes about 0.35 second for image estimation and outlier area detection, and 0.43 second for Mura pixel detection. Test images are tested under Intel Core 2 Duo-2.66 GHz platform with 2 GB DDR SDRAM on the WinXP system.

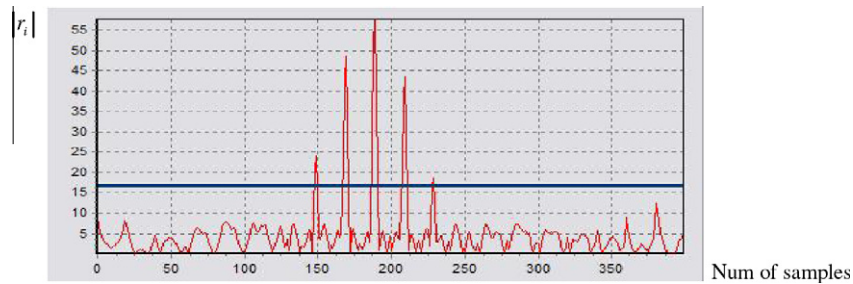


Fig. 5. Absolute PRESS residuals and the threshold.

Table 1

Outlier area images by using different values of c_1 .

Original image	$c_1 = 0.5$	$c_1 = 1.0$	$c_1 = 1.5$

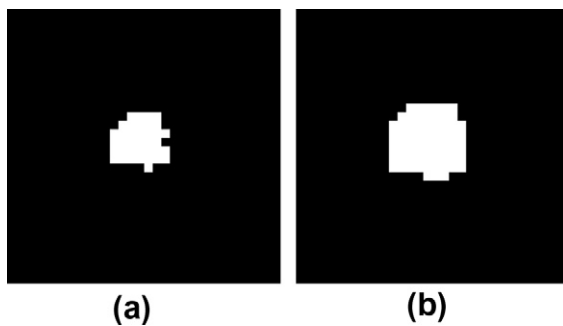


Fig. 6. Illustration of eight-neighbor region dilation: (a) outlier area image, (b) resulting image after eight-neighbor region dilation.

3.3. Performance comparison for the proposed method and Lee and Yoo's method

For the performance comparison between the proposed method and Lee and Yoo's (2004) method, 8 selected test images are used as the test-bed. In the implementation of Lee and Yoo's method, the parameter setting of $l = 2$, $h = 4$, $\alpha = 20$ and $T = 2$ is used accord-

ing to the original paper, and the parameter c_1 and c_2 are set to 1.0 and 3.0 in our method. The results of both methods are compared under the same platform of the machinery. It has been discovered that, in the phase 1, Lee and Yoo's method took about 4375 seconds in the computation of the diagnostic measure for all pixels. In the phase 2, the computation time for Mura pixel detection is about 0.85 second. By contrast, the overall computation time of the proposed method is just about the time in the phase 2 required by Lee and Yoo's method. The efficiency delivered by the proposed method envisions a potential application for on-line Mura defect inspection in TFT-LCD manufacturing.

The computational results for performance comparison are demonstrated in Table 4. The true Mura areas in all test images are circumscribed. The Mura areas in test images (c) and (d) are below the line, and the last two test images are the normal image without Mura that are intentionally included to evaluate the robustness of the two Mura detection methods. The detection rate (DR) is defined as the sum of the correct detected pixels divided by the sum of the real Mura pixels in each test image, and the false alarm rate (FAR) is defined as the ratio of false detected pixels in total detected pixels for each method. For the spot-Mura image in (a), both methods detect the correct position of the Mura with 0% of FAR. Lee and Yoo's method detects a larger proportion of

Table 2
Mura detection images by using different values of c_2 for the normal images.

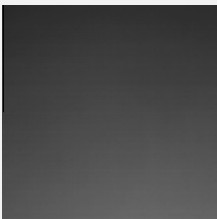

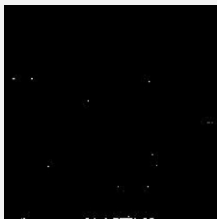
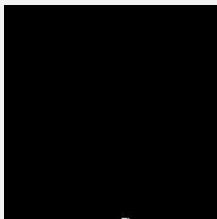
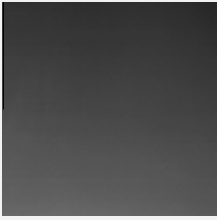


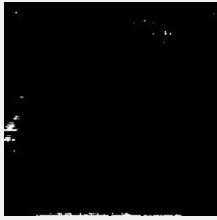

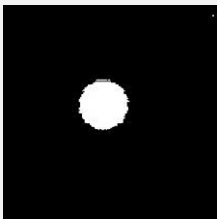
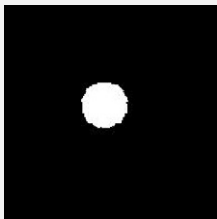
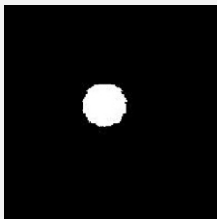

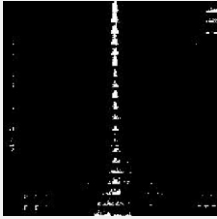
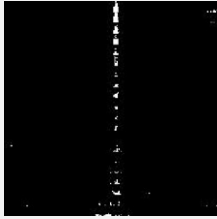
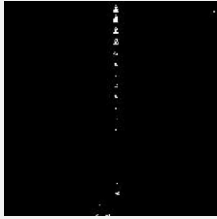
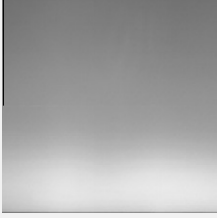
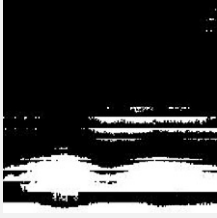
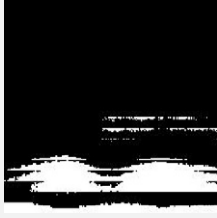
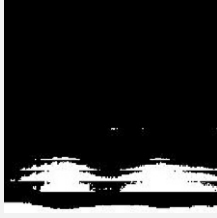
Original image	$c_2 = 2.0$	$c_2 = 2.5$	$c_2 = 3.0$
			
			

Table 3
Mura detection images by using different values of c_2 for Mura images.



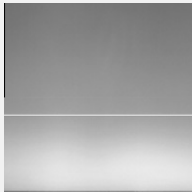

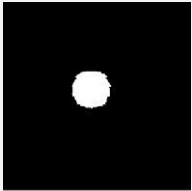

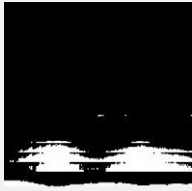



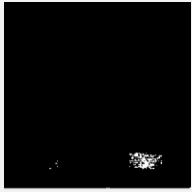




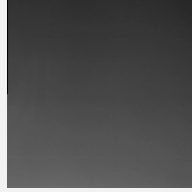
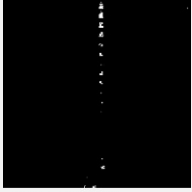

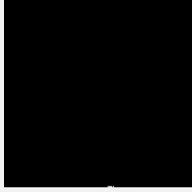

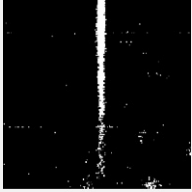



Original image	$c_2 = 2.0$	$c_2 = 2.5$	$c_2 = 3.0$
			
			
			

the Mura area with 55.99% of DR than 39.47% of DR from the proposed method. In the ring Mura image in (b), these two methods produce similar performance and can only detect partially the ring Mura region since the ring Mura area occupies most of the image, thus leading to high FARs. The proposed method still performs a bit better than Lee and Yoo's method in sketching the shape of the ring Mura. The results on gravity Mura images (c) and (d) show that Lee and Yoo's method could not detect the gravity Mura well, and it only detects very few Mura pixels in the resulting images with DR of 2.71% and 5.13%, respectively. By contrast, the proposed method presents better detection ability with DRs of 25.87% and 19.46% and lower FARs of 0% and 0% than Lee and Yoo's method on the obvious and latent gravity Mura images, respectively. In

the presence of the line Mura in (e) and (f), the proposed method does not perform as sensitively as Lee and Yoo's method according to DR, but Lee and Yoo's method also yields more misclassified pixels in the resulting images with FARs. In the normal images (g) and (h), the proposed method produces much less misclassified Mura pixels (6 pixels and 264 pixels) than Lee and Yoo's method (1687 pixels and 2030 pixels) under non-uniform illumination conditions. As evidenced by the comparison of the aforementioned results, the proposed method demonstrates an extremely competitive performance as compared to Lee and Yoo's method with comparable DRs and better FARs. Of course, the fast processing speed owned by the proposed method is of more practical relevance in on-line Mura inspection.

Table 4

Comparison results of the proposed method and Lee and Yoo's Method with detection rate (DR) and false alarm rate (FAR).

Mura Type	(a) Spot Mura	(b) Ring Mura	(c) Gravity Mura	(d) Gravity Mura
Original image				
The proposed method	 DR: 39.47% FAR: 0%	 DR: 7.30% FAR: 35.20%	 DR: 25.87% FAR: 0%	 DR: 19.46% FAR: 0%
Lee and Yoo's method	 DR: 55.99% FAR: 0%	 DR: 9.78% FAR: 23.25%	 DR: 2.71% FAR: 0%	 DR: 5.13% FAR: 13.57%
Mura Type	e. Line Mura	f. Line Mura	g. Normal Image	h. Normal Image
Original image				
The proposed method	 DR: 5.13% FAR: 5.75%	 DR: 11.42% FAR: 26.82%	 Detected pixel: 6 pixels	 Detected pixel: 264 pixels
Lee and Yoo's method	 DR: 51.94% FAR: 32.22%	 DR: 72.68% FAR: 33.28%	 Detected pixel: 1687 pixels	 Detected pixel: 2030 pixels

4. Conclusion

This paper presents a Mura defect detection method that is based on regression diagnostics using the PRESS residuals. The experimental results of the test images of several Mura types show that the proposed method produces competitive performances in effectiveness and efficiency while detecting Mura defects as compared to Lee and Yoo's (2004) method. Particularly, the

proposed method performs with an excellent computation speed about 0.8 second for handling a 200×200 image. The merit of computational efficiency makes the proposed method a viable on-line Mura inspection tool in TFT-LCD manufacturing. Furthermore, the proposed Mura detection method always returns lower false alarm rates which can greatly reduce the inspection cost. In the proposed method, the test image is estimated by the linear regression model for calculating the PRESS residuals, and then Mura regions and Mura

pixels are extracted by two proposed threshold methods. As a practical guidance, the suggested threshold parameter values for the PRESS residuals are 1.0, and 3.0 for the Mura pixel detection as to achieve robust inspection outputs. For further research, large-scale industrial images and other Mura types should be included to further validate the effectiveness of the proposed method. Alternative regression diagnostics particularly designed for ring- and gravity-Mura defects deserve a future study.

References

- Allen, D.M., 1971. The Prediction Sum of Squares as a Criterion for Selecting Variables, Technical Report No. 23, Department of Statistics, University of Kentucky.
- Allen, D.M., 1974. The relationship between variable selection and data augmentation and a method for prediction. *Technometrics* 16, 125–127.
- Chen, S.L., Chou, S.T., 2008. TFT-LCD Mura defect detection using wavelet and cosine transforms. *J. Adv. Mech. Des. Syst. Manuf.* 2, 441–453.
- Gonzalez, R.C., Woods, R.E., 2002. *Digital Image Processing*, second ed. Prentice-Hall.
- Jiang, B.C., Wang, C.C., Liu, H.C., 2005. Liquid crystal display surface uniformity defect inspection using analysis of variance and exponentially weighted moving average techniques. *Int. J. Prod. Res.* 43 (1), 67–80.
- Kim, S.H., Kang, T.G., Jeong, D.H., 2008. Region mura detection using efficient high pass filtering based on fast average operation. In: *Proc. 17th World Congress the International Federation of Automatic Control*, Seoul, Korea, pp. 8190–8195.
- Lee, J.Y., Yoo, S.I., 2004. Automatic detection of region-mura defect in TFT-LCD. *IEICE Trans. Inform. Syst.* E87-D (10), 2371–2378.
- Lu, C.J., Tsai, D.M., 2005. Automatic defect inspection for LCDs using singular value decomposition. *Internat. J. Adv. Manuf. Technol.* 25, 53–61.
- Lu, C.J., Tsai, D.M., 2008. Independent component analysis-based defect detection in patterned liquid crystal display surfaces. *Image Vis. Comput.* 26, 955–970.
- Montgomery, D.C., Peck, E.A., 1992. *Introduction to Linear Regression Analysis*, second ed. Wiley-Interscience.
- Niblack, W., 1986. *An Introduction to Image Processing*. Prentice-Hall. pp. 115–116.
- Otsu, N., 1979. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cynern. SMC-9*, 62–66.
- Pratt, W.K., Sawkar S.S., O'reilly, K., 1998. Automatic blemish detection in liquid crystal flat panel displays. In: *IS&T/SPIE Symp. on Electronic Imaging: Science and Technology*, San Jose, California, vol. 3306.
- Taniguchi, K., Ueta, K., Tatsumi, S., 2006. A Mura detection method. *Pattern Recognition* 39, 1044–1052.
- Wang, Z.Y., Ma, L., 2006. Implementation of region-mura detection based on recursive polynomial-surface fitting algorithm. In: *Proc. Second Asia Internat. Symp. on Mechatronics*, Hong Kong, China, pp. 1–6.