International Journal of Innovative Computing, Information and Control
Volume **8**, Number **7(A)**, July **2012**
ICIC International ©2012 ISSN 1349-4198
pp. 4837–4850

LCD MURA DETECTION WITH MULTI-IMAGE ACCUMULATION AND MULTI-RESOLUTION BACKGROUND SUBTRACTION

Din-Chang Tseng*, You-Ching Lee and Cheng-En Shie
Institute of Computer Science and Information Engineering
National Central University
No. 300, Jungda Rd., Chungli 32001, Taiwan
*Corresponding author: tsengdc@ip.csie.ncu.edu.tw
{ fcrdxesz; pluto0327 }@hotmail.com

Received April 2011; revised November 2011

Abstract. *We integrated the techniques of multi-image accumulation and multi-resolution background subtraction to detect mura defects in low-contrast and high-noised TFT-LCD images. First, several images of an LCD on a moving product conveyer are contiguously captured and then a synthesized LCD image is used to calibrate the non-uniform illumination of these images. Second, the images are aligned in position to accumulate the gray levels of the pixels which all correspond to a point on the LCD. The multi-image accumulation process enhances the contrast between the mura defects and the background; moreover, visible mura problems due to the view angle and the uneven illumination are also mostly resolved. Third, the multi-resolution backgrounds of the accumulated image are progressively estimated based on the discrete wavelet transform (DWT) and the defect candidates are extracted coarse-to-fine and accumulated by subtracting the background from the accumulated image. The multi-resolution background subtraction strategy speeds the detection process and solves the problem of different sizes of mura defects without reducing the detection rate. Finally, a standard thresholding method is used to "threshold out" the mura defects. The stability and effect of the proposed method are demonstrated in experiments.*

Keywords: Automatic optical inspection, Mura detection, Multi-image accumulation, Multi-resolution background subtraction, TFT-LCD

1. **Introduction.** Currently, thin film transistor liquid crystal displays (*TFT-LCDs*) are the most popular flat panel display devices. A critical task for *TFT-LCD* manufacturers is to control their visual quality, that is, to quickly inspect the *TFT-LCD* for outward appearance and mura defects. The outward appearance defects are generally made from the existence of foreign bodies in flat panel units, non-uniform color filters, non-uniform gap in glass bases, ill-functioning polarizers, or poorly backlit units [1]. These kinds of defects generally appear microscopically, but can still be detected successfully [2-6]. In contrast, macro-scope mura defects are difficult to perceive with the human eye, due to their properties of low contrast and non-uniform brightness as shown in Figure 1. Moreover, non-uniform illumination makes the mura harder to detect and some mura defects can only be detected at a special angle, also shown in Figure 1.

There are many types of mura defects: lines, black spots, white spots, black regions, white regions, and rings [7]. Many methods have been proposed to detect mura defects, which can be categorized into three classes: spatial-domain [7-21], frequency-domain [22,23], and feature-based [24] methods. The spatial-domain methods directly process image pixels to detect mura defects. Those methods can be categorized into three sub-classes: thresholding, template matching, and background estimation. In the thresholding sub-class, Chen and Chiang [8] proposed a threshold method to detect mura defects; the method is very fast but is not accurate for non-uniform backgrounds. Nakano and Mori

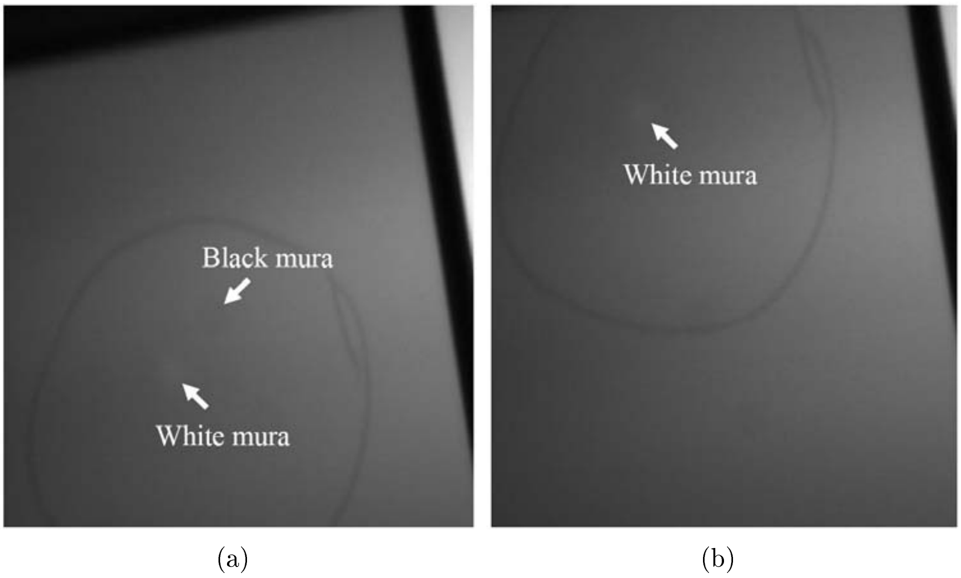


Figure 1. Mura defects can be perceived in some special angles: (a) both black and white mura defects, (b) only a white mura

[9], Oh et al. [10], and Ryu et al. [11] initially de-noised and enhanced mura defects in the wavelet domain and then thresholded the mura defects in the spatial domain. Their methods can only detect small-sized mura defects, but sizes of mura defects vary greatly. Kim et al. [12] modified the standard deviation of gray levels to reduce the non-uniform gray levels and then "thresholded out" mura defects. This method can detect most kinds of mura defects other than line mura defects. In the template matching sub-class, Chen et al. [13] proposed a Laplacian of Gaussian mask to detect region and line mura defects. The method is quick, but the mask is unsuitable for other types and sizes of mura defects. Oh et al. [14] proposed a directional filter bank to detect mura defects. Their method can detect horizontal and vertical line mura defects, but it fails to detect other shaped defects. Song et al. [15,16] proposed a blob mask to detect low-contrast blob mura effectively but the mask is not suitable for other shaped defects. In the background estimation sub-class, Lee et al. [17] used a B-spline surface to fit the image background and then subtracted the background from the source image to detect mura defects. The method could detect mura at a 90% detection rate, but normal panels have a 12% false positive rate (i.e., mistaking non-mura pixels as mura pixels). Baek et al. [18] used one-dimensional (1D) polynomial background estimation to detect region mura defects. Lee and Yoo [7] used two-dimensional (2D) polynomial curves to estimate image background in a moving window. These methods [7,18] can detect mura defects of any shape or orientation. Overlapped windows, however, take more time to process; and if the defect size is larger than the window, the defect cannot be detected. For time-consuming problems, Wang and Ling [19] used recursive least square estimation to reduce the computational complexity of polynomial background fitting. Chen and Chang [20] accelerated Lee and Yoo's [7] method by calculating Welsch's distance for the 2D curve fitting. However, Chen and Chang's method still takes 500 seconds to process a 256 × 256-pixel image. Chen and Jhou [21] accelerated background estimation by simplifying Chen and Chang's fitting method and proposing a one-search-time labeling method. Their method still takes 45 seconds to process an image, demonstrating that the background estimation method is still not efficient enough to use in common practice.

In the frequency-domain class, Fang et al. [22] used *Laplacian* of *Gaussian* filters on the Fourier coefficients of images to detect blob mura defects. The method is more efficient than the spatial-domain methods; however, different filters are required to detect mura defects with different shapes and orientations. Chen and Kuo [23] acquired discrete cosine transform (*DCT*) coefficients of a mura image to estimate image background and then subtracted the background from the source image to detect mura defects. Chen and Kuo's method is fast and suitable for most kinds of mura defects, but the method cannot detect low-contrast and line mura defects at the same time; moreover, it fails to estimate backgrounds of large mura defects because *DCT* does not retain the spatial information of the images.

The feature-based methods detect mura defects by the feature information of the images. Tsai and Tsai [24] used 1D optical flow to detect mura defects. This method is fast and processes 20 200 × 200-pixel images per second, but the non-uniform environmental illumination may influence the detection, and the shapes of the detected defects are fragmented.

Summarily, the above methods cannot simultaneously solve the eight problems previously mentioned: low contrast, non-uniform illumination, non-uniform background, different view angles, different mura direction, different mura shape, different mura size, and time consuming.

In this study, we propose a method of *LCD* mura detection based on multi-image accumulation and multi-resolution background subtraction. First, an *LCD* on a moving product conveyer is contiguously captured by several images with different locations and a synthesized *LCD* image is used to calibrate the non-uniform illumination of the images. Second, the images are aligned in position to accumulate the gray levels of pixels which all correspond to a point on the *LCD*. The accumulated images have the advantages of enhancing the contrast, depressing the random noise, and solving the view-angle problem where a point on the *LCD* has different view angles on different images. Third, the multi-resolution backgrounds of the accumulated image are progressively estimated based on the discrete wavelet transform (*DWT*). Here, we use a 2D plane polynomial to estimate the backgrounds. Several studies [7,17-21] have shown that the estimation methods can solve the problems of non-uniform background, different-shape, and different-direction mura; however, these methods are time consuming. Additionally, using different-size estimation windows to detect mura with different sizes will also take too much time. To solve the different-size problem and to accelerate the background estimation, we take the accumulated image into a multi-resolution and then refine the estimated background from coarse to fine. The accumulated image subtracted from the estimated background leaves the defect candidates. Finally, a standard thresholding method is used to "threshold out" the mura defects. Figure 2 briefly describes the steps of the proposed method.

The remains of this paper are organized as follows. Section 2 presents the multi-image accumulation method for non-uniform illumination and non-uniform displacement of *LCD*s on product conveyers. Section 3 provides the mura detection method using multi-resolution background subtraction. The experiments and their results are reported and discussed in Section 4, and the conclusions are presented in Section 5.

2. Multi-image Accumulation.

The non-uniformity of a background mainly results from the non-uniform material of *TFT-LCD* and a fault of the producing processes. Non-uniform material creates mura defects; however, the non-uniform environmental illumination also makes *TFT-LCD* non-uniform in appearance. Non-uniform illumination must therefore be calibrated to eliminate false detections. On the other hand, we can accumulate the gray levels of pixels which correspond to a point on the *LCD* to enhance the

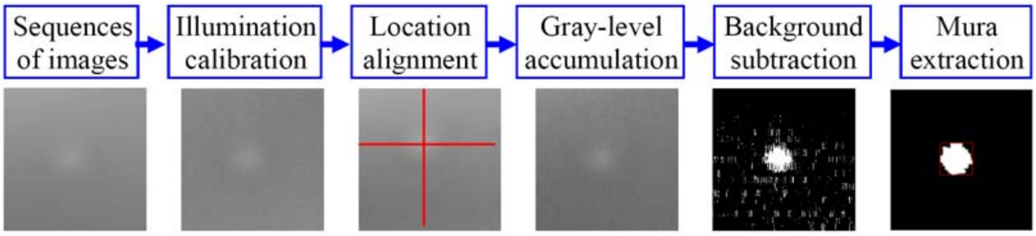


Figure 2. The steps of the proposed mura detection method

contrast between a mura and background. The accumulation of the pixels' gray levels needs to align the contiguous images on the product conveyer which move at unstable speed.

2.1. **Calibration of non-uniform illumination.** Initially, m non-defected *TFT-LCD* images were manually selected as referenced images for analysis. For each of the referenced images, gray level $f_r(x, y)$ of pixel (x, y) is composed of illumination $i(x, y)$ and material reflectance $r_r(x, y)$,

$$f_r(x, y) = i(x, y) r_r(x, y). \tag{1}$$

We take the average gray levels of the m referenced images for pixel (x, y) as

$$\bar{f}_r(x, y) = i(x, y) \bar{r}_r(x, y), \tag{2}$$

and then divide every un-calibrated image

$$f(x, y) = i(x, y) r(x, y) \tag{3}$$

by $f_r(x, y)$ to get

$$g(x, y) = \frac{f(x, y)}{\bar{f}_r(x, y)} = \frac{r(x, y)}{\bar{r}_r(x, y)}, \tag{4}$$

where $g(x, y)$ is invariant to the non-uniform illumination and will be used to detect mura defects. $g(x, y)$ has the same signal-to-noise ratio (*SNR*) to $f(x, y)$ as described in the Appendix; but the contrast of $g(x, y)$ is reduced. Here, we can enhance the contrast by multiplying $g(x, y)$ and constant c, where c can be set as the average gray level of $\bar{f}_r$ as one example shown in Figure 3.

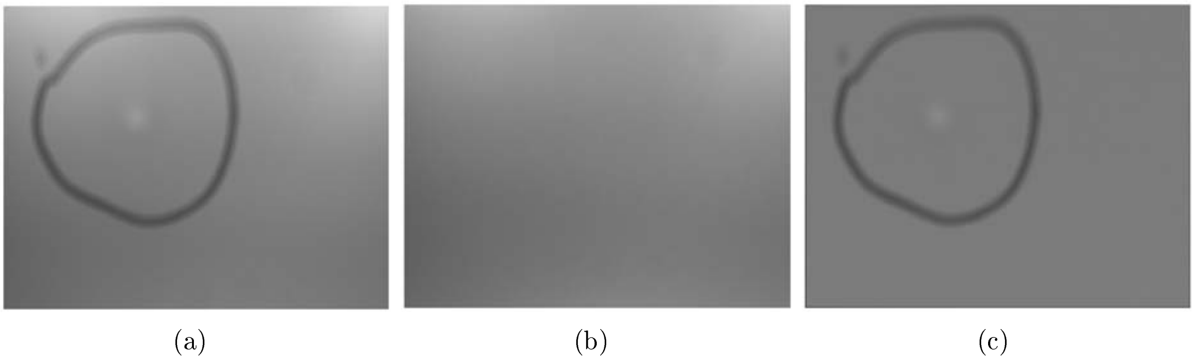

(a) (b) (c)

Figure 3. Before and after illumination calibration: (a) the un-calibrated image, (b) the background, (c) the calibrated image

2.2. **Alignment of multiple images.** The gray levels of all calibrated images of an *LCD* will be accumulated to enhance the contrast between mura and the background as one example shown in Figure 4. Initially, the positions of all images must be aligned. There is only a slight rotation, and it can be ignored. A 2D template matching is commonly used for the alignment; however, the 2D matching is very time consuming. Moreover, as the contrast of images is weak and features are not clear, the matching may not be as stable. We propose a 1D profile matching method to enhance the contrast for matching and to increase the speed of the matching process. First, the gray levels of an image are horizontally and vertically accumulated to generate two profiles. Then the profiles of each two adjacent images are matched to find the displacement based on the correlation measurement.

The accumulations of gray levels in the vertical and horizontal directions are used to enhance the features for the alignment. Let $h(y)$ and $v(x)$ be the horizontal and vertical accumulations of calibrated image $g(x, y)$,

$$\begin{cases} h(y) = \sum\limits_{x} g(x, y) \\ v(x) = \sum\limits_{y} g(x, y) \end{cases}, \tag{5}$$

respectively. Assuming the ranges of longitudinal and latitudinal displacements are $[-l_o, l_o]$ and $[-l_a, l_a]$, respectively, the displacement of two adjacent (ith and $i+1$th) images is calculated by maximizing the correlation

$$\begin{cases} d_x = \arg\left\{ \max\limits_{d_x^*=-l_a}^{l_a} \left\{ \dfrac{\sum\limits_{x} \left\{[v_i(x) - \bar{v}_i]\,[v_{i+1}(x - d_x^*) - \bar{v}_{i+1}]\right\}}{\left\{\sum\limits_{x} [v_i(x) - \bar{v}_i]^2 \sum\limits_{x} [v_{i+1}(x - d_x^*) - \bar{v}_{i+1}]^2\right\}^{1/2}} \right\}\right\} \\ d_y = \arg\left\{ \max\limits_{d_y^*=-l_o}^{l_o} \left\{ \dfrac{\sum\limits_{y} \left\{\left[h_i(y) - \bar{h}_i\right]\,\left[h_{i+1}(y - d_y^*) - \bar{h}_{i+1}\right]\right\}}{\left\{\sum\limits_{y} \left[h_i(y) - \bar{h}_i\right]^2 \sum\limits_{y} \left[h_{i+1}(y - d_y^*) - \bar{h}_{i+1}\right]^2\right\}^{1/2}} \right\}\right\} \end{cases}, \tag{6}$$

where d_x is the horizontal displacement, d_y is the vertical displacement, $\bar{v}$ is the average of $v(x)$, and $\bar{h}$ is the average of $h(y)$.

FIGURE 4. A sequence of calibrated images

The displacement calibration may be somewhat inaccurate because there is no obvious feature or contrast in mura images. Stable speed of a product conveyer is useful for estimating the longitudinal displacement and helpful for finding an accurate displacement. Fortunately, an inaccurate location does not influence the detection result if there is no mura defect; moreover, if there are some mura defects in an image sequence, a slightly inaccurate location only slightly enlarges or reduces the size of mura defects without the existence of defects disappearing. Therefore, the proposed displacement calibration is still usable.

After aligning every two adjacent images, we accumulate the same-position gray levels of all contiguous images to obtain an accumulated image.

3. Multi-resolution Background Subtraction.

A *TFT-LCD* image is composed of background, noise, or mura defects as shown in Figure 5. We detect defects by subtracting the background from the image and "thresholding out" noise from the subtracted image. In general, the sizes of the mura defects vary so that a fixed-sized mask is unsuitable to detect all possible mura defects. We here use a multi-resolution detection method to extract mura as follows:

Step 1. Decompose an image into a hierarchical multi-resolution image structure with three levels.
Step 2. Estimate initial backgrounds for the three different-resolution images.
Step 3. Determine suspected defect blocks in every resolution.
Step 4. Refine the background of the finest resolution image according to the coarse resolution images.
Step 5. Subtract the refined background of the finest-resolution image from the calibrated image and "threshold out" the mura defects.

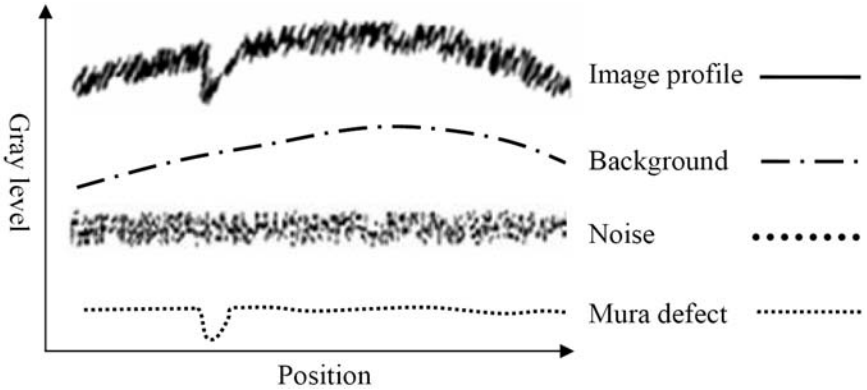


FIGURE 5. Components of an image profile

To construct the hierarchical multi-resolution image, we use the Haar function as the basis function to decompose the calibrated image into a wavelet hierarchy of three levels with ten subbands. Then the three different-scaled low-frequency subbands, named L_3, L_2, and L_1, are sequentially used to estimate the background from the coarsest-resolution subband L_3 to the finest-resolution subband L_1 as shown in Figure 6.

3.1. Initial background estimation.

We divide every L_k subband into several l^2-pixel blocks. Initial backgrounds of L_1 and L_2 are estimated block by block. In a block, all pixels (x, y) are used to fit a 2D plane as the background by the least square estimation method. To estimate the initial background of the L_3 subband, mean μ and standard deviation σ of L_3 pixels are calculated. Then the values of all L_3 pixels (x, y) whose

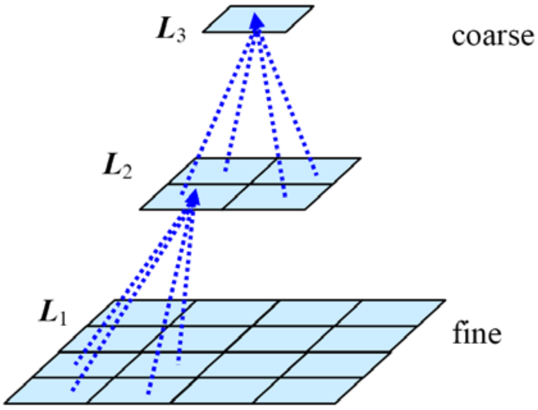


FIGURE 6. The L subbands in a dyadic wavelet tree

values satisfy $|L_3(x, y) - \mu| < \sigma$ are used fit 2D plane S_3 as the background by the least square estimation method. The initial backgrounds will be refined from the coarsest L_3 to the finest L_1.

3.2. Background refinement.

A block may contain defects; the background of a defect block should be refined; otherwise, the defect may be undetectable. The background refinement is not very sophisticated. Let $S_k^j(x, y)$ be a pixel on the background of the jth block in L_k subband, we use the criterion

$$\begin{cases} (x, y) \text{ is a suspected defect pixel,} & \text{if } \left|L_k^j(x, y) - S_k^j(x, y)\right| > 2\sigma; \\ (x, y) \text{ is not a suspected defect pixel,} & \text{otherwise.} \end{cases} \tag{7}$$

to determine whether a pixel is a suspected defect. If a block contains less than $p\%$ suspected defect pixels, the block's background is called pure; otherwise the background is non-pure and needs to be refined. In general, p can be set as 25 because a wavelet coefficient in a coarse subband is calculated from its four super coefficients in the fine subband.

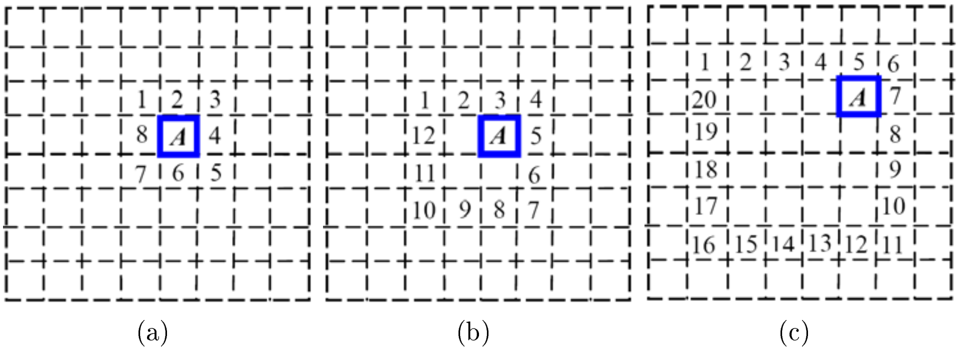
The coarse-to-fine refinement of a block uses its eight neighboring blocks to fit a new background replacing the block's initial background. Actually, only L_1 blocks need to be refined; the blocks in L_2 and L_3 need no refinement, but only need tags to indicate that they are pure or non-pure.

Step 1. Every $\boldsymbol{L}_3$ block is tested pure or non-pure.
Step 2. Every $\boldsymbol{L}_2$ block is tested pure or non-pure.
Step 3. Every $\boldsymbol{L}_1$ block is tested pure or non-pure. If the background of $\boldsymbol{L}_1$ block $\boldsymbol{A}$ is non-pure and needs to be refined, three cases are considered.

3.1 If block $\boldsymbol{A}$'s sub-block on L_2 is pure, the pixels of $\boldsymbol{A}$'s eight neighboring L_1 blocks (whose L_2 sub-blocks are pure) are used to fit a new background instead of $\boldsymbol{A}$'s background, as illustrated in Figure 7(a).

3.2 If block $\boldsymbol{A}$'s L_2 sub-block is non-pure and $\boldsymbol{A}$'s L_3 sub-block is pure, the pixels of $\boldsymbol{A}$'s 12 neighboring L_1 blocks (whose L_2 sub-blocks are pure) are used to fit a new background instead of $\boldsymbol{A}$'s background. There are four sub-cases based on the wavelet structure and one example is illustrated in Figure 7(b).

3.3 If block $\boldsymbol{A}$'s L_2 sub-block and L_3 sub-block are non-pure, the pixels of $\boldsymbol{A}$' 20 neighboring L_1 blocks (whose L_2 sub-blocks are pure) are used to fit a new background instead of $\boldsymbol{A}$'s background. There are sixteen sub-cases based on the wavelet structure and one example is illustrated in Figure 7(c).



(a) (b) (c)

FIGURE 7. The neighboring blocks for refinement: (a) sub-block is pure, (b) only sub-block on L_2 is non-pure, (c) both sub-blocks on L_2 and L_3 are non-pure

3.3. **Post-processing.** After the process of the background refinement, the final background image $b(x, y)$ is reconstructed from the background of L_1 by inverse discrete wavelet transform. Then, the mura defect pixels are extracted by the criterion

$$M(x,y) = \begin{cases} 1, & \text{if } |g(x,y) - b(x,y)| > 2\sigma; \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

where "1" indicates a mura pixel and "0" is a non-mura pixel. Finally, we use the opening and closing morphology operations to prune the mura defects. A time-consuming adaptive thresholding method seems to be more suitable for this extraction; actually, with our complete consideration on the process of images, the simple thresholding is fully sufficient for this extraction.

4. **Experiments.**

In the experiments, we detected several different-typed mura defects with different numbers of accumulation images. The detection power and efficiency of the proposed method were evaluated and compared with five other methods.

We used the real contiguous images to demonstrate the proposed mura detection method; however, not all types of mura images were available; thus, several artificial mura images were generated for the test and comparison. Non-uniform background and noise were also generated in the artificial images. The test images include region, line, ring, and cloud muras as shown in Figure 8, where (a) and (b) are real images and (c), (d), (e), and (f) are artificial images. Contrasts between mura defects and backgrounds of the images are ranges of: (a) $5 \sim 8$, (b) $2 \sim 4$, (c) $3 \sim 6$, (d) $3 \sim 6$, (e) $2 \sim 5$, and (f) $4 \sim 6$ gray levels. The size of all images is 256×256 pixels.

Without pruning the defect regions, the detection results with different numbers of accumulation images were compared as shown in Figure 9. The higher an accumulation number, the less detected noise and the more compact structure of mura defects. A greater number of accumulation images, however, will take more computation time. Thus, we accumulated eight images accompanied with morphology operations to quickly solve the view-angle problem.

The mura defects shown in Figure 8 have different types, sizes, shapes, and directions; they were detected as shown in Figure 10. All types and shapes of mura defects were detectable. Large-size mura defects are shown in Figure 10(a) and (b), and small-size mura defects are shown in Figure 10(d). The minimum detectable contrast between defect and background by the proposed method was 3 gray levels.

We took twelve image sequences with four kinds of mura defects to statistically analyze the precision and accuracy of the proposed methods, as shown in Tables 1 and 2. The

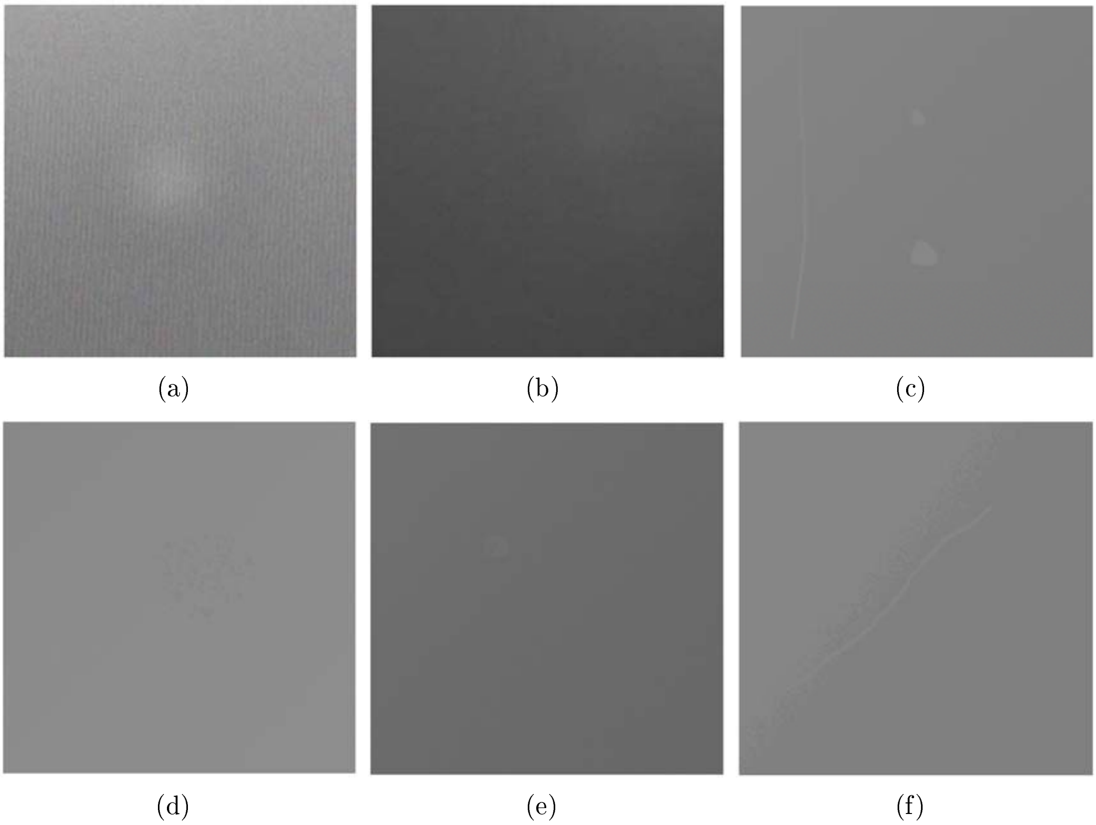


Figure 8. Test images: (a) real region mura, (b) real region mura, (c) artificial line and region muras, (d) artificial cloud mura, (e) artificial ring mura, (f) artificial line mura

precision and accuracy were defined from the testing of the hypothesis [25]. The null hypothesis was defined as the tested pixel is assumed to be a non-mura pixel. Then four values are defined for evaluation:

i. True positive: number of mura pixels that were detected as mura pixels;
ii. False positive: number of non-mura pixels that were mistaken as mura pixels, the number is also called type I error;
iii. True negative: number of non-mura pixels that were detected as non-mura pixels; and
iv. False negative: number of mura pixels that were mistaken as non-mura pixels. This number is also called type II error.

The precision is defined as

$$Precision = \frac{\text{True positive}}{\text{True positive + False position}}. \tag{9}$$

This is the ratio of "number of mura pixels which were correctly detected as mura pixels" to "number of pixels which were detected as mura pixels". The accuracy is defined as

$$Accuracy = \frac{\text{True positive + True negative}}{\text{True positive + False positive + True negative + False negative}}. \tag{10}$$

This is the ratio of "all correctly detected pixels" to "all pixels." From Table 2, we can see that the precision and accuracy of the proposed methods were 0.9818 and 0.9876, respectively. If we count the mura regions, all mura regions were detected; that is, the detection rate for mura regions is 100%.

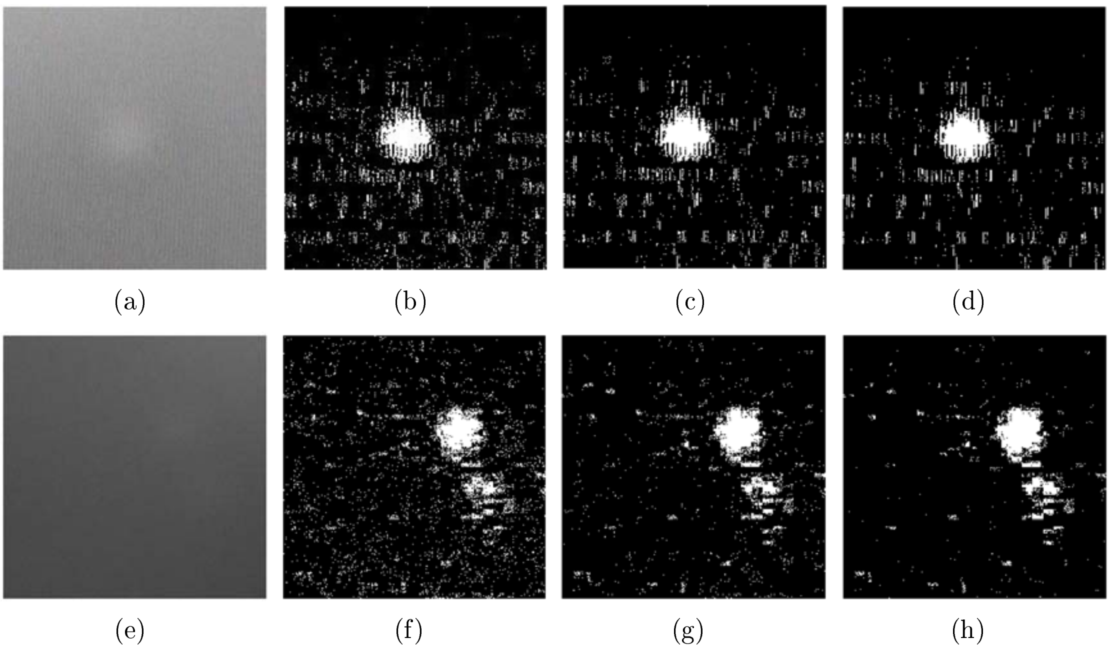
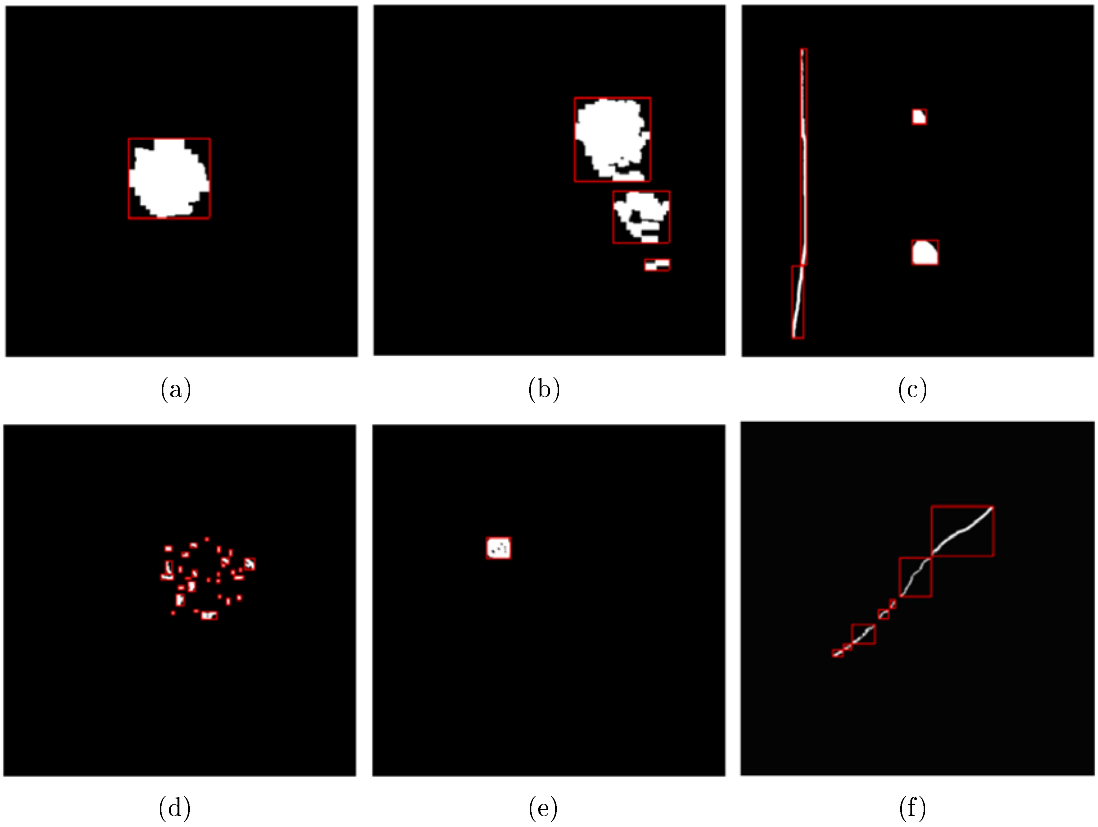


FIGURE 9. The comparisons of accumulation times: (a) a region mura, the detection results of (a) with accumulating (b) 4 images, (c) 8 images, and (d) 16 images; (e) two unapparent region muras, the detection results of (e) with accumulating (f) 4 images, (g) 8 images, and (h) 16 images

Ten image sequences of 256 × 256 pixels and ten image sequences of 1024 × 1024 pixels were used to evaluate the efficiency of the proposed detection method. Each image sequence was accumulated eight times. The average processing time of every step of the proposed method is listed in Table 3. The average processing time of a 256 × 256 image sequence and a 1024 × 1024 image sequence are 5.576 seconds and 23.268 seconds, respectively. Most of the total processing time was spent on the multi-resolution mura detection, which spent 77% and 67% processing time for the two-sized images, respectively. This is a hierarchically non-overlapped and block-by-block process, which means that we can easily accelerate processing time by using more processors.

The performance of the proposed method and the other five methods were compared and are listed in Table 4. The proposed method has the minimum detectable contrast over other methods and has no undetectable mura types. Furthermore, the proposed method accumulates contiguous images to solve the problem of inspecting angles. Although the processing speed of the proposed method is not the fastest, the processing time of the proposed method is easy to improve by adding hardware processors.

5. **Conclusions.** In this paper, an online *TFT-LCD* mura defect detection method is proposed. The method consists of illumination calibration, multi-image accumulation, and multi-resolution background subtraction. The multi-image accumulation of gray levels and the multi-resolution background subtraction are two special techniques in this study. The multi-image accumulation technique is the first trial in the optical inspection of objects on product conveyers. This technique enhances the contrast between objects and their backgrounds to make the mura detection more stable. The multi-resolution background subtraction reduces the processing time without influencing the detection rate. Experiments show that the proposed method is powerful and is able to simultaneously solve problems of low contrast, non-uniform illumination, non-uniform background,


(a) (b) (c)

(d) (e) (f)

FIGURE 10. The detected results of images in Figure 8: (a) real region mura, (b) real region mura, (c) artificial line and region muras, (d) artificial cloud mura, (e) artificial ring mura, (f) artificial line mura

TABLE 1. Numbers of mura and non-mura pixels in the experimental images

	Pixels	Mura pixels	Non-mura pixels	Detected mura pixels	Detected non-mura pixels
Region mura *i*	65536	3876	61660	3505	61660
Region mura *ii*	65536	3500	62036	3102	62036
Region mura *iii*	65536	2952	62584	2756	62584
Region mura *iv*	65536	2000	63536	1856	63536
Region mura *v*	65536	985	64551	570	64551
Line mura *i*	65536	4560	60976	3586	60976
Line mura *ii*	65536	4801	60735	2504	60735
Line mura *iii*	65536	4756	60780	2895	60780
Line mura *iv*	65536	4689	60847	3952	60847
Line mura *v*	65536	4523	61013	4358	61013
Ring mura	65536	2065	63471	2680	62856
Cloud mura	65536	3678	61858	2098	61858
Total	786432	42385	744047	33862	743432

different view angles, different mura direction, different mura shape, and different mura size. The performance of the proposed method is dependent on the stability of product conveyers. The more stable the conveyer, the more regular the captured images. A more stable carrying speed and less vibration of the conveyers make the proposed techniques

Table 2. Hypothesis testing for mura and non-mura pixels

	True positive	False positive	True negative	False negative
Region mura *i*	3505	0	61660	371
Region mura *ii*	3102	0	62036	398
Region mura *iii*	2756	0	62584	196
Region mura *iv*	1856	0	63536	144
Region mura *v*	570	0	64551	415
Line mura *i*	3586	0	60976	974
Line mura *ii*	2504	0	60735	2297
Line mura *iii*	2895	0	60780	1861
Line mura *iv*	3952	0	60847	737
Line mura *v*	4358	0	61013	165
Ring mura	2065	615	62856	0
Cloud mura	2098	0	61858	1580
Total	33247	615	743432	9138

Table 3. The average processing time for each step (seconds)

Steps\Image size	256×256 pixels	1024×1024 pixels
illumination calibration	0.063	1.133
Displacement calibration	1.200	6.463
Multi-resolution mura detection	4.313	15.672
Total time	5.576	23.268

Table 4. The detection comparison among six methods

Methods\Effect	Limitation in mura types	Minimum detectable contrast	View-angle problem	Processing time (256×256 pixels)
Kim et al. [12]	Line mura	4	*yes*	< 1 sec.
Chen-Kuo [23]	Line mura	6	*yes*	< 1 sec.
Lee-Yoo [7]	*No limitation*	8	*yes*	*Not available*
Chen-Chang [20]	*No limitation*	5	*yes*	> 500 sec.
Chen-Jhou [21]	*No limitation*	*Not available*	*yes*	45 sec.
The proposed	*No limitation*	3	*no*	5.576 sec.

more practicable. These techniques have been tested for mura detection on the *LCD* conveyers at *Chunghwa Picture Tubes*, *LTD* in Yangmei City, Taiwan. The performance of the conveyer was acceptable, and the detected regions were enlarged without decreasing the detection rate.

Acknowledgment. This work is partially funded by *Delta Electronics, INC.*, Taiwan under the research project grant for "Development of Image-based Inspection Techniques for Optical-electronic Components (III)".

REFERENCES

[1] Y. Mori, K. Tanahashi and S. Tsuji, Quantitative evaluation of mura in liquid crystal displays, *Optical Engineering*, vol.43, no.11, pp.2696-2700, 2004.

[2] K. B. Lee, M. S. Ko, J. J. Lee, T. M. Koo and K. H. Park, Defect detection method for TFT-LCD panel based on saliency map model, *Proc. of IEEE TENCON Conf.*, Chiang Mai, Thailand, pp.223-226, 2004.

[3] C. J. Lu and D. M. Tsai, Automatic defect inspection for LCDs using singular value decomposition, *The Int. Journal of Advanced Manufacturing Technology*, vol.25, no.1-2, pp.53-61, 2005.

[4] D. M. Tsai and C. Y. Hung, Automatic defect inspection of patterned thin film transistor-liquid crystal (TFT-LCD) panels using one-dimensional Fourier reconstruction and wavelet decomposition, *Int. Journal of Production Research*, vol.43, no.21, pp.4589-4607, 2005.

[5] Z. Zhang, I. Nakamura, C. J. Li, T. Imamura, T. Miyake and H. Fujiwara, Metal plating surface defect detection by template matching using morphological processing, *ICIC Express Letters, Part B: Applications*, vol.2, no.3, pp.615-620, 2011.

[6] N. Araki, K. Nishiuchi, T. Sato, Y. Konishi, E. Fujiwara and H. Ishigaki, Defect detection for mirror polished metal surface using independent component analysis, *ICIC Express Letters*, vol.5, no.9(B), pp.3291-3296, 2011.

[7] J. Y. Lee and S. I. Yoo, Automatic detection of region-mura defect in TFT-LCD, *IEICE Trans. Information and Systems*, vol.E87-D, no.10, pp.2371-2378, 2004.

[8] S.-H. Chen and Y.-M. Chiang, Automatic inspection of mura defect in color filter, *Proc. of INFORMS Int. Conf.*, Hong Kong, China, 2006.

[9] H. Nakano and Y. Mori, Measurement method for low-contrast nonuniformity in liquid crystal displays by using multi-wavelet analysis, *Proc. of SPIE Conf. on Optical Diagnostics*, Bellingham, WA, pp.588013-1-588013-6, 2005.

[10] J. H. Oh, W. S. Kim, C. H. Han and K. H. Park, Defect detection of TFT-LCD image using adapted contrast sensitivity function and wavelet transform, *IEICE Trans. Electronics*, vol.E90-C, no.11, pp.2131-2135, 2007.

[11] J. S. Ryu, J. H. Oh, J. G. Kim, T. M. Koo and K. H. Park, TFT-LCD panel blob-mura inspection using the correlation of wavelet coefficients, *Proc. of IEEE TENCON Conf.*, Chiang Mai, Thailand, pp.219-222, 2004.

[12] W. S. Kim, D. M. Kawak, Y. C. Song, D. H. Choi and K. H. Park, Detection of spot-type defects on liquid crystal display modules, *Key Engineering Materials*, vol.270-273, pp.808-813, 2004.

[13] H.-C. Chen, L.-T. Fang, L. Lee, C.-H. Wen, S.-Y. Cheng and S.-J. Wang, LOG-filter based inspection of cluster mura and vertical-band mura on liquid crystal displays, *Proc. of Conf. on Machine Vision Applications in Industrial Inspection XIII*, San Jose, CA, pp.257-265, 2005.

[14] J. H. Oh, D. M. Kwak, K. B. Lee, Y. C. Song, D. H. Choi and K. H. Park, Line defect detection in TFT-LCD using directional filter bank and adaptive multilevel thresholding, *Key Engineering Materials*, vol.270-273, pp.223-238, 2004.

[15] Y. C. Song, D. H. Choi and K. H. Park, Morphological blob-mura defect detection method for TFT-LCD panel inpection, *Lecture Notes in Computer Science*, vol.3215, pp.862-868, 2004.

[16] Y. C. Song, D. H. Choi and K. H. Park, Multiscale detection of defect in thin film transistor liquid crystal display panel, *Japanese Journal of Applied Physics*, vol.43, no.8(A), pp.5465-5468, 2004.

[17] G. B. Lee, W. S. Kim, Y. S. Chung, J. J. Lee and K. H. Park, Adaptive surface fitting for inspection of FPD devices using multilevel B-spline approximation, *Proc. of IEEE TENCON Conf.*, Melbourne, Australia, pp.1445-1448, 2005.

[18] S. I. Baek, W. S. Kim, T. M. Koo, I. Choi and K. H. Park, Inspection of defect on LCD panel using polynomial approximation, *Proc. of IEEE TENCON Conf.*, Chiang Mai, Thailand, pp.235-238, 2004.

[19] Z. Wang and M. Ling, Implementation of region-mura detection based on recursive polynomial-surface fitting algorithm, *Proc. of Asia Int. Symp. on Mechatronics*, Hong Kong, China, pp.1-6, 2006.

[20] S.-L. Chen and J. H. Chang, TFT-LCD mura defects automatic inspection system using linear regression diagnostics model, *Proc. of the Institution of Mechanical Engineers – Part B: Journal of Engineering Manufacture*, vol.222, no.11, pp.1489-1501, 2008.

[21] S.-L. Chen and J. W. Jhou, Automatic optical inspection on mura defect of TFT-LCD, *Proc. of the 35th Int. Conf. on Machine Tool Design and Research*, Taipei, Taiwan, pp.233-236, 2007.

[22] L. T. Fang, H. C. Chen, I. C. Yin, S. J. Wang, C. H. Wen and C. H. Kuo, Automatic mura detection system for liquid crystal display panels, *Proc. of SPIE-IS&T Conf. on Electronic Imaging*, San Jose, CA, pp.143-152, 2006.

[23] L.-C. Chen and C. C. Kuo, TFT-LCD mura defect detection based on discrete cosine transform and filtering strategy, *Proc. of Int. Manufacturing Leaders Forum*, Taipei, Taiwan, 2006.

[24] D. M. Tsai and H. Y. Tsai, Low-contrast surface inspection of mura defects in liquid crystal displays using optical flow-based motion analysis, *Machine Vision and Applications*, vol.22, no.4, pp.629-649, 2011.

[25] D. L. Olson and D. Delen, *Advanced Data Mining Techniques*, Springer-Verlag, Heidelberg, German, 2008.
[26] J. T. Bushberg, J. A. Seibert, E. M. Leidholdt Jr. and J. M. Boone, *The Essential Physics of Medical Imaging*, 2nd Edition, Lippincott Williams & Wilkins, Pennsylvania, 2006.
[27] D. L. Donoho and I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, *Biometrika*, vol.81, pp.425-455, 1994.

Appendix. To see if a mura has disappeared or if noise is enhanced after the illumination calibration, the signal to noise ratio (*SNR*) is analyzed.

We use the *SNR* defined by Bushberg et al. [26] to define the *SNR* of un-calibrated image f. The *SNR* was defined as the ratio of the mean gray level to the standard deviation of the image noise. That is, the *SNR* of f is

$$SNR_f = \bar{f}/\sigma_f, \tag{11}$$

where $\bar{f}$ is the mean of f and σ_f is the standard deviation of the noise of f. The SNR of calibrated image g is

$$SNR_g = \bar{g}/\sigma_g, \tag{12}$$

where $\bar{g}$ is the mean of g and σ_g is the standard deviation of noise of g. Donoho and Johnstone [27] have defined the standard deviation of the image noise as media$(|w|)/0.6745$, where median$(\cdot)$ is a median filter and w is the set of wavelet coefficients in the HH_1 subband. A wavelet transform can convolute an image with a set of wavelet functions. Thus, Equation (11) can be rewritten as

$$SNR_f = 0.6745\bar{f}/\text{median}(|W * f|), \tag{13}$$

where W is a wavelet convolution function. Equation (12) can be rewritten as

$$SNR_g = 0.6745\bar{g}/\text{median}(|W * g|) = 0.6745(\bar{f}/\bar{f}_r)/\text{median}(|W * (f/\bar{f}_r)|). \tag{14}$$

Since $\bar{f}_r$ is a constant,

$$SNR_g = 0.6745(\bar{f}/\bar{f}_r)/\text{median}(|W * f|/\bar{f}_r) = 0.6745\bar{f}/\text{median}(|W * f|) = SNR_f. \tag{15}$$

This means that the non-uniform illumination calibration does not change the signal-to-noise ratio of the un-calibrated images.