

# Journal of Electronic Imaging

JElectronicImaging.org

## Correcting geometric and photometric distortion of document images on a smartphone

Christian Simon  
Williem  
In Kyu Park



# Correcting geometric and photometric distortion of document images on a smartphone

Christian Simon, Williem, and In Kyu Park\*

Inha University, Department of Information and Communication Engineering, 100 Inha-ro, Nam-gu, Incheon 402-751, Republic of Korea

**Abstract.** A set of document image processing algorithms for improving the optical character recognition (OCR) capability of smartphone applications is presented. The scope of the problem covers the geometric and photometric distortion correction of document images. The proposed framework was developed to satisfy industrial requirements. It is implemented on an off-the-shelf smartphone with limited resources in terms of speed and memory. Geometric distortions, i.e., skew and perspective distortion, are corrected by sending horizontal and vertical vanishing points toward infinity in a downsampled image. Photometric distortion includes image degradation from moiré pattern noise and specular highlights. Moiré pattern noise is removed using low-pass filters with different sizes independently applied to the background and text region. The contrast of the text in a specular highlighted area is enhanced by locally enlarging the intensity difference between the background and text while the noise is suppressed. Intensive experiments indicate that the proposed methods show a consistent and robust performance on a smartphone with a runtime of less than 1 s. © 2015 SPIE and IS&T [DOI: [10.1117/1.JEI.24.1.013038](https://doi.org/10.1117/1.JEI.24.1.013038)]

Keywords: document image; optical character recognition; smartphone; perspective distortion; photometric distortion; moiré pattern noise; specular highlight.

Paper 14356 received Jun. 20, 2014; accepted for publication Feb. 6, 2015; published online Feb. 26, 2015.

## 1 Introduction

The development of mobile devices has shown rapid advances in recent years. The latest smartphones are equipped with advanced hardware components, such as high-speed processors, high-resolution cameras, and high-resolution color displays. Such hardware has stimulated a variety of application developments, including image and video processing. The necessity to process the document image is invoked to accomplish more flexibility and mobility over a conventional scanning method. In this context, the optical character recognition (OCR) application has changed from scanning the document images on a flatbed scanner to capture them directly using smartphone. However, mobile OCR is a complex image processing application that requires a high-resolution image, a large memory space, and huge numbers of computations. In the early period of mobile phone development, Laine and Nevalainen<sup>1</sup> described the technical challenges and limitations of OCR on mobile phones and proposed a simple mobile OCR system on a Symbian operating system. Most of the problems of creating the algorithms in smartphone are related to the processing time and memory consumption.

An OCR application extracts text information from captured document images, allowing users to utilize it for information retrieval or translation into different languages. However, the OCR performance is highly affected by the quality of the input document images. Geometric and photometric distortion, which are the main issues addressed in this paper, often degrade the quality of the input images when users capture them with a smartphone. Therefore, several processing methods should be applied to improve the quality

of the input images, allowing a higher performance of an OCR application to be achieved.

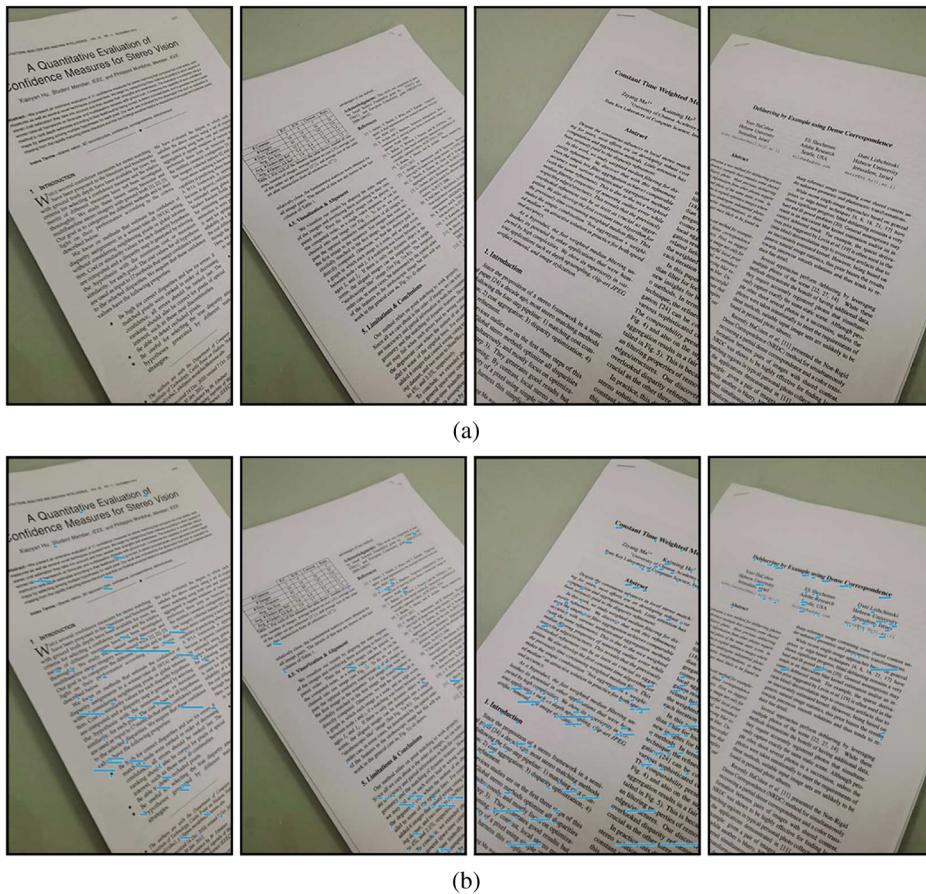
In this paper, we propose a set of preprocessing algorithms to handle geometric and photometric distortion on a smartphone. A strategy is developed mainly to meet the industrial needs, i.e., accuracy, speed, and ease of use. The proposed framework works robustly on various types of input document images and runs completely on a smartphone within a second. The proposed algorithms are adaptive and have a few parameters to be adjusted, thus they produce acceptable results and consistently improve the quality of the input images. The main contributions of this paper are summarized as follows.

- We propose an efficient preprocessing framework for mobile OCR on a smartphone that satisfies the industrial requirements.
- The framework includes robust geometric and photometric distortion correction algorithms that run completely on a smartphone with a fast running time.
- The proposed algorithms are adaptive and require a few parameter adjustments.

### 1.1 Geometric Distortion

Current mobile OCR algorithms usually extract the textual information by detecting text lines in the horizontal direction. However, it is difficult to capture an ideal input document image without geometric (i.e., perspective) distortion. Figure 1 shows a few examples of document images with perspective distortion and inaccurate OCR results.

\*Address all correspondence to: In Kyu Park, E-mail: [pik@inha.ac.kr](mailto:pik@inha.ac.kr)



**Fig. 1** Geometric (perspective) distortion of document images. (a) Captured document images using a smartphone. (b) The output images after running OCR application without perspective correction. Correctly identified words are underlined in blue. “Optical Reader” on Samsung’s Galaxy S4 is used.

Several works have been conducted to rectify the perspective<sup>2–5</sup> and geometric distortion in document images.<sup>6</sup> However, most of them are computationally too heavy to run on a mobile processor. In this paper, we propose a perspective rectification method for mobile device applications, while considering the limited resources of a smartphone.

## 1.2 Photometric Distortion

### 1.2.1 Moiré pattern noise

Moiré pattern noise usually appears when we capture the image of a digital display, i.e., an LCD monitor screen, directly using a digital camera. Muammar and Dragotti<sup>7</sup> stated that such noise occurs because a scene is insufficiently band-limited when captured by a digital camera. Note that users often capture text information from digital monitors when they need an instant capture or have no access to the source data. It is, therefore, important to filter out moiré pattern noise, allowing users to obtain clear document images that can be used as inputs for an OCR application.

Figure 2(a) shows a few examples of corrupted document images with moiré pattern noise. The key idea of suppressing moiré pattern noise is to filter the background and text region separately. This makes sense because a moiré pattern is usually more noticeable in the background than in a text area.

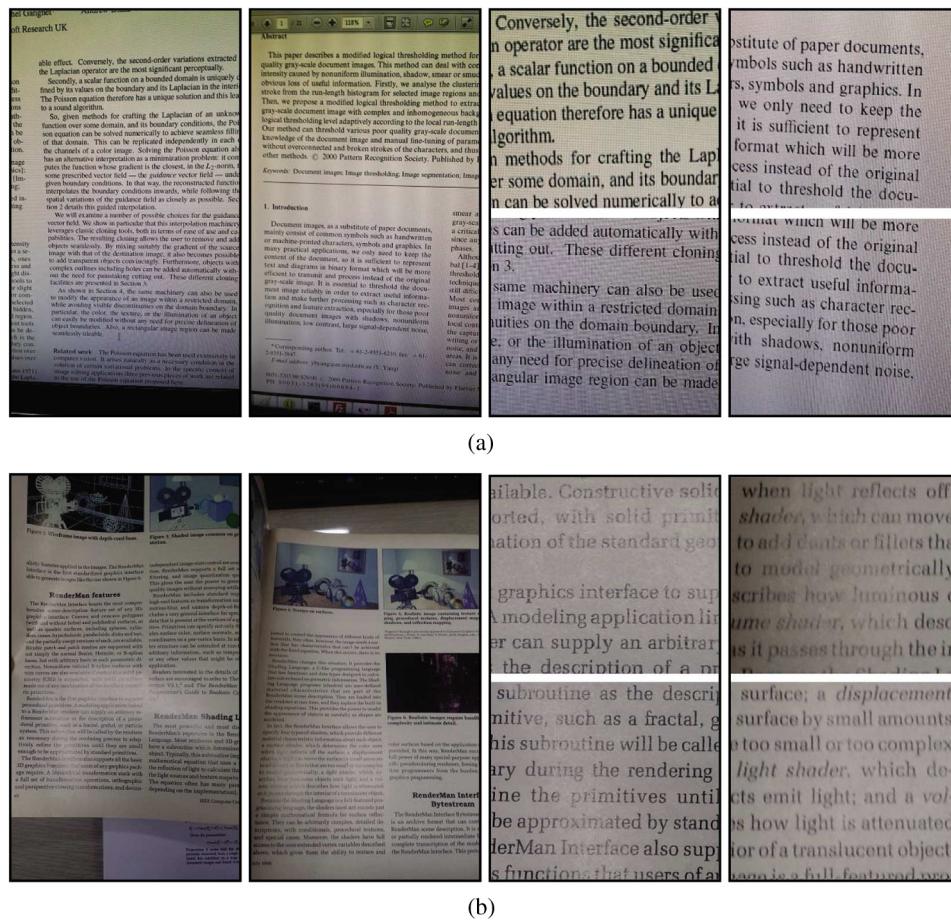
### 1.2.2 Specular highlight

Document images captured from glossy magazine pages usually suffer from specular highlights that yield a poor contrast between the text and background, as shown in Fig. 2(b). Therefore, OCR does not show its original performance when the input document images have strong specular highlights. To solve this problem, the contrast between the text and background should be adjusted locally to achieve a significant difference in intensity.

The remainder of this paper is organized as follows. Section 2 introduces the previous works. The perspective rectification method is presented in Sec. 3. Sections 4 and 5 describe the moiré pattern noise removal and contrast enhancement algorithms for specular highlighted areas, respectively. Section 6 provides the experimental results of the proposed algorithms. Finally, we provide concluding remarks in Sec. 7.

## 2 Previous Works

The approaches that deal with perspective distortion in document images can be typically categorized into two different types, i.e., three-dimensional (3-D) and two-dimensional (2-D) approaches, the first of which reconstructs a 3-D shape of the document images<sup>8–11</sup> and the second which utilizes 2-D image processing techniques.<sup>12,13</sup> Several works have utilized the boundary information of a document image regardless of



**Fig. 2** Photometric distortion of document images. (a) Distorted images with moiré pattern noise (better visible on the monitor). (b) Distorted images with specular highlighted area. In each row, right two images show the magnified areas.

the type (flat or curved) of document surface.<sup>14–17</sup> However, such algorithms may fail to rectify the input when there is no clear boundary information. Moreover, the methods developed by Lu et al.<sup>18,19</sup> restore the global distortion of document images including planar and curved documents. However, they depend heavily on the resolution of the document image. To recover the text structures, several researchers have developed perspective rectification algorithms based on the vanishing points.<sup>2–5</sup> They mostly achieve this by sending the horizontal and vertical vanishing points toward infinity. Hence, it is possible to reconstruct a frontoparallel view from a document image with perspective distortion. However, there has been a lack of discussion regarding perspective rectification algorithms applied to a smartphone when considering the limitations of such hardware.

These methods rely on vertical character strokes to detect the vertical vanishing points, and yet vertical lines can be detected reliably only if the size of the character stroke is sufficiently large. Note that typical input document images for an OCR application have small sized character strokes, as shown in Fig. 1.

Moiré pattern noise removal can be achieved in two ways: by improving the quality of the display, as achieved by Saveljev et al.,<sup>20</sup> or by improving the quality of the captured image. We focus on the second method by performing some preprocessing methods on a smartphone. A few smoothing

techniques have been proposed to suppress moiré pattern noise. Han et al.<sup>21</sup> proposed a technique applied to scanned halftone images. Siddiqui and Bouman<sup>22</sup> also provided a training-based method for scanned halftone images. Furthermore, Ville et al.<sup>23</sup> described an approach to suppress moiré patterns using modified splines. However, these techniques focus on scanned halftone or printed images, which are not suitable for suppressing moiré pattern noise in an image captured from an LCD monitor. Cao and Kot<sup>24</sup> stated that moiré pattern noise can be suppressed by adjusting the camera settings. However, this method is impractical for mobile devices. Related to moiré pattern removal techniques for a mobile device, Pilu and Pollard<sup>25</sup> proposed demosaicing and text extraction methods implemented on a consumer PDA. However, they are not applicable for direct moiré pattern removal because an input document image recaptured from a monitor has a different pattern format that cannot be handled using only a small low-pass filter. The most related work to eliminating pattern noise was proposed by Liu and Srihari,<sup>26</sup> which presents a technique using a run-length histogram producing a binary image. To the best of our knowledge, there are no previous works focusing on removing moiré pattern noise in document images recaptured by a mobile camera.

A few existing methods<sup>27–29</sup> have focused on handling uneven illumination conditions of a document image. However, they have generally focused on converting a color

image into a binary form. In addition, they do not handle serious contrast distortion and noises residing in specular highlighted areas that degrade the readability of many characters. Shi and Govindaraju<sup>30</sup> proposed a method to enhance the characters in historic documents by normalizing the light intensity of the background. Furthermore, Sharma et al.<sup>31</sup> proposed a method for removing visible artifacts on the translucent pages of historic documents with improper opacity. However, the material and reflection properties of historic documents differ from those of glossy documents with specular highlights.

### 3 Perspective Distortion Correction

The proposed algorithm for rectifying perspective distortion in document images consists of preprocessing, skew rectification, and perspective rectification steps. The novel part of the proposed approach is to find the horizontal and vertical vanishing points in a fast and robust manner. The typical results of each intermediate step are shown in Fig. 3.

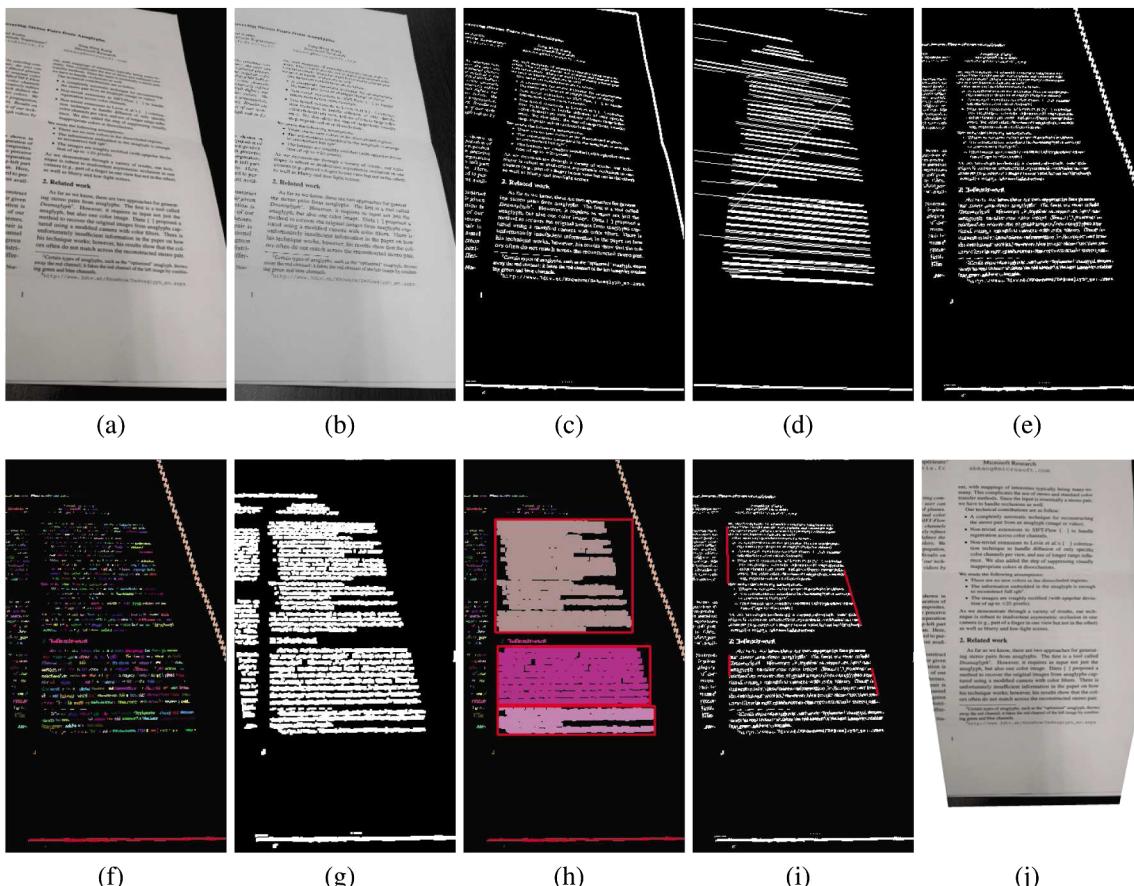
#### 3.1 Preprocessing

To achieve a fast computational speed, the input image is first downscaled to a normalized resolution and converted into a grayscale. The preprocessing includes binarization and text selection. The binarization technique is not complicated because we have no intention of handling ink-bleed or

historical documents. For this reason, the adaptive thresholding technique proposed by Bradley and Roth<sup>32</sup> is employed to obtain the background and text. The key point of adaptive thresholding is the ratio between the pixel intensity  $I(i, j)$  and average intensity  $\bar{I}(i, j)$  of  $36 \times 36$  neighborhood pixels in a window. Note that the average intensity is computed using an integral image. If the ratio is smaller than the threshold  $\tau_1 (=0.85)$ , the pixel is considered a part of the background. For text selection, it is necessary to select the text region despite the presence of a large number of unrelated surrounding objects. Therefore, we utilize Clark and Mirmehdi's algorithm,<sup>33</sup> which assumes that a text region usually has a high edge density. To extract an edge, Sobel filtering is applied. The local sum of the edge responses in a window  $S(i, j)$  is then computed to calculate the density in the window. If the ratio between the local sum  $S(i, j)$  and maximum sum  $S_{\max}$  is smaller than the threshold  $\tau_2 (=0.1)$ , then the pixel is considered as a background pixel. Then, the final binary image  $B(i, j)$  is generated by integrating both results.

#### 3.2 Skew Rectification

To rectify the skew distortion, the proposed framework extracts the horizontal lines using a Hough line detection algorithm,<sup>34</sup> as shown in Fig. 3(d). All intersection points from all lines are then computed and clustered using a K-means clustering. Each cluster is considered a candidate



**Fig. 3** Step-by-step result of perspective distortion correction (i.e., perspective rectification). (a) Input document image. (b) Grayscale image. (c) Binary image. (d) Hough lines image. (e) Skew rectified binary image. (f) Text blob image. (g) Dilated image. (h) Paragraph blob image. (i) Boundary lines image. (j) Perspectively rectified image.

of a horizontal vanishing point. Given an  $i$ 'th candidate, a skew projective histogram is computed to find the best horizontal vanishing point. The histogram  $H_i$  is generated by projecting angular slice from each point candidate. Each angular slice is regarded as a histogram bin. The histogram computation is similar to Clark and Mirmehdi's work,<sup>4</sup> but it has substantial differences in the image resolution, the usage of K-means clustering, and the method to accumulate the derivative histogram. Then, the number of peaks in a derivative histogram larger than the threshold  $\tau_3 (= 0.05)$  is counted as follows:

$$D_i(j) = \begin{cases} 1 & |H_i(j+1) - H_i(j)| > \tau_3 \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

$$V_i = \sum_{j=1}^{N-1} D_i(j),$$

where  $i$ ,  $j$ , and  $N$  are the index of the candidate horizontal vanishing point, the index of the histogram bin, and the total number of histogram bins, respectively. The binary voting value and its total are denoted as  $D_i$  and  $V_i$ , consecutively. The candidate with the highest voting value (i.e.,  $\text{argmax } V_i$ ) is selected as the horizontal vanishing point. Skew rectification is then conducted by warping the input image with the resolution  $W \times H$  using homography matrix  $\mathbf{H}$ , which is modeled as follows:

$$\theta = \arctan\left(\frac{V_y^h}{V_x^h}\right), \quad \mathbf{T}_1 = \begin{bmatrix} 1 & 0 & -\frac{W}{2} \\ 0 & 1 & -\frac{H}{2} \\ 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

$$\mathbf{R}_1 = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} V_x^{h'} \\ V_y^{h'} \\ 1 \end{bmatrix} = \mathbf{R}_1 \mathbf{T}_1 \begin{bmatrix} V_x^h \\ V_y^h \\ 1 \end{bmatrix}, \quad \mathbf{K}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{1}{V_x^{h'}} & 0 & 1 \end{bmatrix}, \quad (3)$$

$$\mathbf{H} = \mathbf{T}_1^{-1} \mathbf{K}_1 \mathbf{R}_1 \mathbf{T}_1, \quad (4)$$

where  $(V_x^h, V_y^h)$  is the position of the detected horizontal vanishing point and  $(V_x^{h'}, V_y^{h'})$  is the position of the horizontal vanishing point after being rotated toward the  $x$  axis. In addition,  $\mathbf{T}_1$  is a translation matrix, and the center of the image is moved to the  $(0,0)$  position. Moreover,  $\mathbf{R}_1$  is a rotation matrix used to rotate the horizontal vanishing point toward the  $x$  axis (i.e.,  $y = 0$ ), and  $\mathbf{K}_1$  is a matrix used to send a horizontal vanishing point toward infinity. Figure 3(e) shows the results of the skew rectification.

### 3.3 Perspective Rectification

The proposed perspective distortion method is similar to skew rectification, in which the vertical vanishing point is detected and sent to infinity. Conventional approaches utilize character text strokes to detect the vertical lines. However, this leads to unreliable lines, especially when the character strokes are not sufficiently large. Because we apply the

rectification method to a downsampled image, we cannot apply these approaches directly. Instead, we introduce a novel vertical line detection method that is robust and does not depend on the size of the character strokes. The idea of vertical vanishing point detection is to extract the reliable paragraphs and their left and right boundaries. To detect a paragraph, paragraph blob detection is conducted by employing morphological dilation and a connected component analysis. However, the size of the window for the dilation operation should be adaptive to the scale of the captured image. In our case, the size of the window is  $3 \times 3$  or  $5 \times 5$  depending on the decision value. In addition, it is necessary to determine whether perspective rectification should be computed. Note that if the vertical vanishing point is incorrect, the rectification result is significantly degraded. Using a skew-rectified image, a connected component analysis is conducted to obtain the text blobs as shown in Fig. 3(f). The average height  $\bar{h}$  of all text blobs is then computed. The average height is compared with two thresholds  $\tau_4 (= 1.5)$  and  $\tau_5 (= 6.0)$ . If the average height is larger than  $\tau_4$ , then there is no perspective rectification because the document was captured within a close distance to the camera. In this type of case, it will be extremely difficult to detect a paragraph or a paragraph may not even exist. In another case, the dilation window is set to a large size if the average height is between  $\tau_4$  and  $\tau_5$  and is set to be a small size if the average height is smaller than  $\tau_5$ . Figure 4 shows an example of a dilation operation for different window sizes.

Once the size of the dilation window is set, the skew-rectified image is dilated and paragraph blobs are reconstructed using a connected component analysis. Each paragraph blob is considered a candidate paragraph. However, there are many unreliable paragraph blobs that contain noise and small text regions. To find a reliable paragraph blob, the average height of all detected paragraph blobs is computed. If the paragraph blob size is smaller than the average size, it is considered an unreliable paragraph blob and is excluded in the subsequent procedures. The results of paragraph detection are shown in Fig. 3(h).

For each detected paragraph, the left- and right-vertical boundaries are detected by computing the first and last binary pixels on the top and bottom parts of the paragraph. Certain criteria are utilized to reject unreliable boundaries, such as short lines (shorter than 25 pixels) and lines near the document image boundary. Figure 3(i) shows the results of the boundary line detection. After the boundary detection, the intersection point of each boundary pair is computed such that it is considered as a candidate for a vertical vanishing point. For each candidate, the derivative of perspective projective histogram is calculated to evaluate the paragraph spacing in the vertical direction by counting the number of

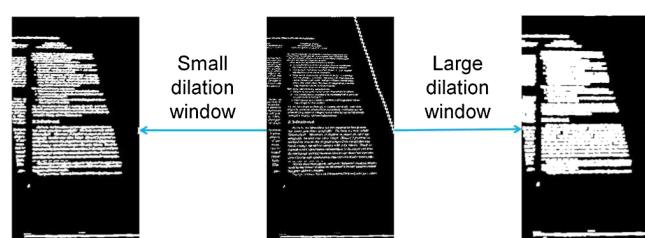


Fig. 4 An example of adaptive window size selection.

local peaks. To reduce the influence of other paragraph blobs, only the local text region inside the paragraph blob is projected rather than projecting all the text in the document image. Finally, the candidate point with the maximum number of peaks (usually 2) in the derivative of the perspective projective histogram is selected as the best vertical vanishing point. Using the detected horizontal and vertical vanishing points, perspective rectification is conducted to restore the frontoparallel view of a distorted image. An extended homography matrix for perspective rectification is described as follows:

$$\mathbf{T}_2 = \begin{bmatrix} 1 & 0 & -V_x^v \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{K}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{1}{V_y^v} & 1 \end{bmatrix}, \quad (5)$$

$$\mathbf{H} = \mathbf{T}_1^{-1} \mathbf{T}_2^{-1} \mathbf{K}_2 \mathbf{T}_2 \mathbf{K}_1 \mathbf{R}_1 \mathbf{T}_1, \quad (6)$$

where  $(V_x^v, V_y^v)$  is the position of the detected vertical vanishing point. In addition,  $\mathbf{T}_2$  is the translation matrix such that the position of the vertical vanishing point is moved to the  $y$  axis (i.e.,  $x = 0$ ), and  $\mathbf{K}_2$  is a matrix used to send the vertical vanishing point toward infinity. Note that the

proposed rectification method has a fixed parameter setup for various input document images.

## 4 Moiré Pattern Noise Removal

The proposed algorithm for moiré pattern noise removal is based on selective image filtering for the missing parts of the characters. The filtering process should be sufficiently computationally inexpensive for implementation on a smartphone. The proposed method consists of several consecutive steps, which are described in the following subsections.

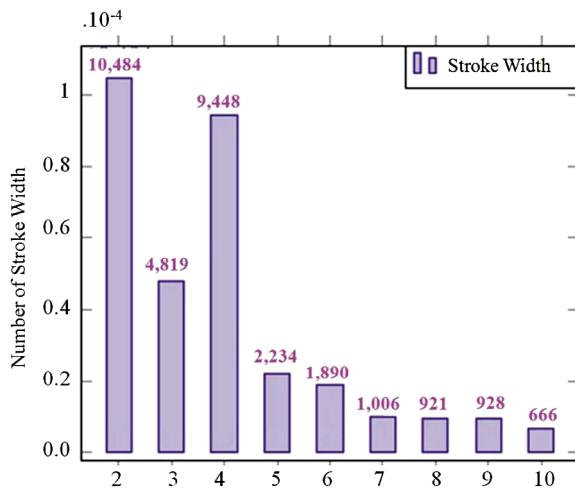
### 4.1 Foreground Extraction

The grayscale input image  $I$  is smoothed using a  $3 \times 3$  Gaussian filter. Then, for fast computation, we downsample the filtered image to half its original size. To extract the foreground, an adaptive thresholding algorithm<sup>32</sup> is employed to obtain a binary image  $B_I$ , as shown in Fig. 5(c) with pattern noise. Note that we set the same parameters for adaptive thresholding with perspective distortion correction.

To remove the pattern, we use a run-length histogram computed using the stroke width information. The proposed approach eliminates all strokes with a width of less than or equal to the highest peak in the run-length histogram, as shown in Fig. 5(d). Figure 6 shows an example of the



**Fig. 5** Step-by-step result of moiré pattern noise removal (better visible on the monitor). (a) Input image. (b) Grayscale image. (c) Binary image after adaptive thresholding. (d) Binary image after run-length filtering. (e) Background image. (f) Binary of the background image. (g) Final background image. (h) Result with moiré pattern noise removed.

**Fig. 6** Histogram of stroke width.

run-length histogram and Fig. 5(d) shows a binary image after run-length filtering which is denoted by  $B_L$ . Note that most of the moiré pattern in a binary image  $B_I$  is eliminated using run-length filtering with the text of the document image remaining.

However, in some cases, the run-length filtering erodes a large number of characters in the text region which can degrade the quality. Hence, it is necessary to verify whether the run-length filtering leads to erroneous results. To this end, the text area  $B_E$  is first obtained using morphological erosion on  $B_L$ . The difference  $B_P$  of two binary images  $B_I$  and  $B_L$  is also calculated to obtain  $E_0$  and  $E_1$  which are the numbers of eliminated pixels inside the text and non-text regions, respectively, as follows:

$$E_0 = \sum_{x,y} [B_E(x,y) == B_P(x,y)], \quad (7)$$

$$E_1 = \sum_{x,y} [B_E(x,y) \neq B_P(x,y)]. \quad (8)$$

The ratio of  $E_0$  to  $E_1$  is then measured to determine which binary image is set as the foreground mask  $M(x,y)$ , as formulated below:

$$M(x,y) = \begin{cases} B_L & \text{if } \frac{E_1}{E_0} > \tau, \\ B_I & \text{otherwise,} \end{cases} \quad (9)$$

where  $\tau (=0.1)$  is the ratio threshold. As a consequence, the foreground mask is resized to the original image resolution. The final foreground  $I_{FF}$  is obtained by masking the input image using  $M(x,y)$  along with its eight neighboring pixels.

## 4.2 Background Extraction

Moiré pattern noise in the background image typically appears with a different size compared to the noise around the text area. Therefore, we apply a smoothing operation to the background area using a different filter. The foreground mask  $M(x,y)$  is utilized to create a background image (without a text area).

In the input image, pixels inside the foreground mask  $M(x,y)$  are replaced by the average value from their

neighboring pixels (outside the mask). The background image  $I_B$  is constructed as follows:

$$I_B(x,y) = \begin{cases} \frac{1}{N} \sum_{(x',y') \in E} I(x',y') & \text{if } M(x,y) = 0 \\ I(x,y) & \text{otherwise} \end{cases}, \quad (10)$$

where  $E$  and  $N$  denote the set of pixels in the edge boundary and the cardinality of  $E$ , respectively. Figure 5(e) shows the intermediate results of the background image. Furthermore, the noise in the background image is smoothed using an adaptive mean filter. To indicate the location of noise, a noisy binary image  $B_B$ , shown in Fig. 5(f), is formed by applying the adaptive thresholding method<sup>32</sup> on  $I_B$ . A noiseless background image  $I_{FB}$  is then generated as follows:

$$I_{FB}(x,y) = \begin{cases} \frac{1}{N} \sum_{(x',y') \in W} I_B(x',y') & \text{if } B_B(x,y) = 0 \\ I_B(x,y) & \text{otherwise} \end{cases}, \quad (11)$$

where  $W$  denotes a large window centered at  $(x,y)$ . We utilize the integral image<sup>32</sup> to apply the mean filtering method for a fast computation. The final smoothed background image is shown in Fig. 5(g).

## 4.3 Image Blending

The final output image is obtained by merging the extracted background  $I_{FB}$  and foreground  $I_{FF}$  images, as shown in Fig. 5(h).

## 5 Contrast Enhancement of Specular Highlighted Area

In the case of glossy paper, characters often have poor contrast between the background and text from highlight reflections. To increase the readability of an OCR application in the presence of specular highlights, the proposed method enhances the contrast of the text against the background, as described in the following subsections.

### 5.1 Background Extraction and Candidate Image Generation

The background image is generated by applying a morphological dilation operation. After dilation, the pixels in the text area are replaced by pixels in the background area. A small dilation area ( $5 \times 5$ ) is utilized in this method for computational reason. The dilated image  $D$  is shown in Fig. 7(c). The dilation operation takes the maximum intensity value within the window, and thus the intensity value in  $D$  is certainly larger than or equal to the value in  $I$ . Therefore, a candidate image  $I_C$ , utilized for a contrast enhancement algorithm, is calculated using  $I_C(x,y) = D(x,y) - I(x,y)$ . An example of a candidate image,  $I_C$ , is shown in Fig. 7(d).

### 5.2 Hole-Filling with Noise Removal

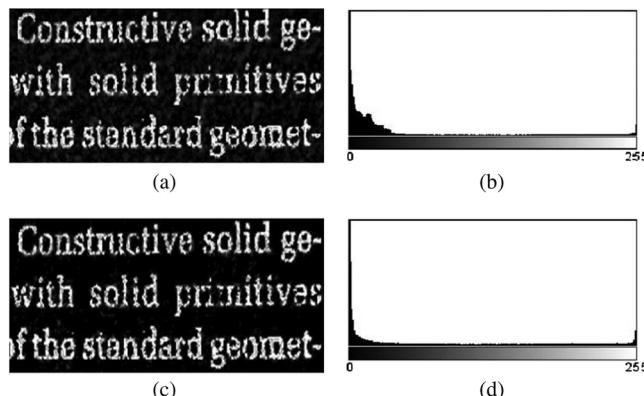
Hole-filling is a process used to fill in a text area that goes undetected by the dilation process. Note that such holes are usually found when the text has a large font size. First, a binary image  $I_B$  is obtained by applying the adaptive thresholding method,<sup>32</sup> as shown in Fig. 7(e). Then, the pixel value of  $I_C$  in the text area is set to the maximum (255), which results in a hole-filled image  $I_H$ , as shown in Fig. 7(f).



**Fig. 7** Step-by-step result of contrast enhancement on specular highlighted area. (a) Input image. (b) Grayscale image. (c) Image after dilation. (d) Candidate image. (e) Image after adaptive thresholding. (f) Hole-filled image. (g) Local contrast image. (h) Result with contrast enhanced on specular highlighted area.

The candidate image  $I_C$  may contain some noise, particularly in the background area, because of the subtraction process. A hole-filled image with noise around the text area is shown in Fig. 8(a). To remove such noise, the local contrast value LC is calculated within  $3 \times 3$ , the image patch as follows:

$$LC = \frac{I_{\max} - I_{\min}}{I_{\max} + I_{\min}}, \quad (12)$$



**Fig. 8** Noise removal in the hole-filled image. (a) Noise exists in background. (b) Histogram of image with noise. (c) Noise is suppressed using the local contrast. (d) Histogram of image with noise suppressed.

where  $I_{\max}$  and  $I_{\min}$  are the maximum and minimum intensity value in a window, respectively. The mean value  $\bar{\mu}$  of all pixels in LC is computed for use as a threshold value. Low intensity values in LC are then eliminated as follows:

$$LC(x, y) = \begin{cases} LC(x, y) & \text{if } LC(x, y) \geq \alpha \times \bar{\mu}, \\ 0 & \text{otherwise} \end{cases}, \quad (13)$$

where  $\alpha$  is the scale factor whose value is  $1/3$ . The local contrast image LC is shown in Fig. 7(g). Afterward, the noise-removed hole-filled image  $I_{NR}$  is computed as follows:

$$I_{NR}(x, y) = \begin{cases} I_H(x, y) & \text{if } LC(x, y) > 0 \\ 0 & \text{otherwise} \end{cases}. \quad (14)$$

A magnified image after the noise-removal process that has been applied in Fig. 8(c).

### 5.3 Reconstruction

The calculated results from the previous steps are utilized to increase the contrast between the background and text. Intermediate results, as shown in Figs. 7(d)–7(g), are not suitable to be processed in the OCR application because such an application has its own binarization algorithm. In addition, the target document image to be passed to the OCR application should be in the form of an enhanced

document image. Moreover, passing a binary document image can result in many incorrectly recognized words owing to the presence of noise. In this sense, the input image  $I$  is subtracted by the noise removed hole-filled image  $I_{NR}$  to produce a contrast-enhanced image with the specular highlights suppressed. Figure 7(h) shows the final results of the proposed method.

## 6 Implementation and Results

### 6.1 Experimental Settings

The proposed methods are implemented on a smartphone, and intensive experiments with real data are carried out to show the performance of each algorithm. We use an off-the-shelf smartphone (Samsung Galaxy S4) equipped with



**Fig. 9** Results of perspective rectification. (a) Perspectively distorted input document images. (b) Perspectively rectified images.

a 13 mega-pixel camera, a 1.6 GHz quad-core processor (Qualcomm Snapdragon 800), and 2 GB of RAM. The employed OCR application (Optical Reader) is operated on Android 4.2.x Jelly Bean.

For the test data, we utilize only real world data. Common documents with black text and white background are used for perspective correction. Document images containing moiré pattern noise were captured from LCD monitors. Furthermore,

magazines with glossy papers are utilized to acquire document images with specular highlights. We assume that the input document images do not have serious blur, which is out of the scope of this paper. In particular, for specular highlighted document images, it is assumed that the text has not fully disappeared and has a lower intensity than the background.

The resolution of the input image for the OCR application is  $1152 \times 2048$  pixels, which is sufficient to provide clear



**Fig. 10** Results of moiré noise removal. Moiré noise is better visible on the monitor. (a) Distorted input images with moiré pattern noise and their magnified images. (b) Output images with moiré pattern noise removed and their magnified images.

representation of the text structures. However, to obtain a fast computational speed, the image is downsampled to  $288 \times 512$  pixels for a perspective rectification. On the other hand, the moiré pattern removal and contrast enhancement of the specular highlighted area are processed at resolutions of  $576 \times 1024$  and  $288 \times 512$ , respectively. Note that the lower resolution image is utilized in the foreground and background extraction for moiré pattern removal, and only in the dilation operation of background extraction for specular highlight removal. In the implementation, memory usage

should be careful since the implemented program crashes if the image resolution is not properly constrained or the consumed memory space is not released properly after usage. It happens when the smartphone is continuously used to capture a series of document images.

## 6.2 Quantitative and Qualitative Results

The proposed methods are implemented in native C++ using an Android API. We achieved fast processing speeds of



**Fig. 11** Results of contrast enhancement on specular highlighted area. (a) Distorted input images with specular highlight distortion. (b) Output images with characters on specular highlighted area enhanced.

about 0.8, 0.7, and 1.0 s for perspective correction, moiré pattern noise removal, and contrast enhancement in specular highlighted areas, respectively.

The processing speed depends highly on the processed image resolution. Therefore, we measure the runtime for several image resolutions. For perspective correction, the runtimes are 0.2 s for  $144 \times 256$ , 0.8 s for  $288 \times 512$ , and 1.5 s for  $576 \times 1024$ . Note that we utilized the original parameter setup for these comparisons. A small image operation is fast but can lead to inaccurate horizontal and vanishing point detections. Therefore, we selected a  $288 \times 512$  resolution based on repetitive experiments. Typical results from applying the proposed techniques are shown in Figs. 9–11. It is clearly shown that the proposed methods work stably for many different cases.

The quantitative performance is measured in terms of OCR application readability. The performance of the geometric and photometric correction using 100 document images is summarized in Table 1. The test images are categorized into five groups (20 images in each group) based on each problem setting. Moiré pattern removal and contrast enhancement in specular highlighted area are compared to a bilateral denoising method<sup>35</sup> and adaptive thresholding,<sup>32</sup> respectively. The parameters configuration for bilateral filtering is  $5 \times 5$  window size, 3.0 for spatial-domain standard deviation, and 0.1 for intensity-domain standard deviation. In adaptive thresholding, the window size is 72 and the threshold value is 8.

The results show that the OCR application readability increases remarkably. Significantly improved results are achieved for perspective rectification because the OCR algorithm in Optical Reader fails in most cases when dealing with perspectively distorted document images. Note that moiré pattern and specular highlight appear locally on a document image. Therefore, a few percentage improvement still indicates that most characters in the distorted region are correctly restored.

In addition, we also conducted an application test to evaluate the proposed perspective correction framework. We captured 384 various document images with different angles and calculated the accuracy rate of the application.

We utilized various printed research papers as the input document images. For each paper, we captured the image in such a way that the information contained could be extracted by the OCR application. This scenario reflects the real use of the OCR application in making a digital version of printed papers. The output images are divided into four categories: skew and perspective rectified images, only skew rectified images, similar input images, and incorrectly rectified images (i.e., rectified images which have distortion). We then measured the percentage of each category. Note that the test images used are more challenging in terms of various skew and perspective angles and smaller texts, and conventional approaches may fail because of their dependency on the size of the character stroke. The percentage values of the proposed method are 95.6% for the skew and perspective rectified images, 3.1% for the only skew/perspective rectified images, 0% for the similar input images, and 1.3% for the incorrectly rectified images. We compared the results with those of Yin's algorithm.<sup>5</sup> The results of their algorithm are 53.1% for the skew and perspective rectified images, 18.9% for the only skew/perspective rectified images, 26.3% for the input similar images, and 1.7% for the wrong rectified images. The results of this experiment show that the proposed method obtains significantly better skew and perspective rectification results than the previous method.

The proposed rectification method requires left- and right-justified assumptions for the captured document images to compute the vertical vanishing point. If the captured document images have no left- and/or right-justification, the proposed rectification method conducts only skew rectification. We classified the results of 65 images without left- and/or right-justified paragraphs. The percentage values are 0% for the skew and perspective rectified images, 98.5% for the only skew/perspective rectified images, 0% for the input similar images, and 1.5% for the wrong rectified images.

The proposed contrast enhancement in a specular highlighted area cannot handle difficult cases in which the characters have higher intensity values than the background. This sometimes occurs because the intensity value of a specular highlighted area increases enormously. However,

**Table 1** Accuracy on the OCR application (“Optical Reader” in Galaxy S4).

	Method	OCR result (%)		
		Input	Processed	Processing time (s)
Proposed	Perspective correction	$1.19 \pm 1.36$	$97.63 \pm 3.97$	0.8
	Moiré pattern removal	$56.78 \pm 16.5$	$69.30 \pm 7.70$	0.7
	Specular highlight removal	$76.90 \pm 7.78$	$87.92 \pm 4.47$	1.0
	Perspective correction and moiré pattern removal	$39.52 \pm 35.29$	$86.78 \pm 9.16$	1.6
	Perspective correction and specular highlight removal	$41.37 \pm 38.13$	$84.13 \pm 11.35$	1.7
Conventional	Adaptive thresholding <sup>32</sup>	$76.90 \pm 7.78$	$80.16 \pm 12.72$	0.3
	Bilateral filter <sup>35</sup>	$56.78 \pm 16.5$	$22.36 \pm 15.51$	60

the intensity values in the background can be inverted by assuming that the background has a larger number of pixels for handling certain recoverable cases. Another possible improvement can be achieved by applying an improved method for converting color into a grayscale based on a principal component analysis, when a colored background and text need to be handled in a more serious manner.

## 7 Conclusion and Future Work

In this paper, we proposed fast and robust preprocessing methods for improving the readability of an OCR application on a smartphone. The preprocessing methods consist of perspective rectification, moiré pattern removal, and text enhancement of specular highlighted areas. For perspective rectification, Hough line detection and skew projective histogram were utilized to rectify the skew distortion. An adaptive paragraph boundary detection and perspective projective histogram were then used to rectify the perspective distortion. For moiré pattern noise removal, we effectively utilized different sized filters. Furthermore, the proposed specular highlight suppression algorithm significantly enhances the text visibility around specularly highlighted areas through subtraction. The experimental results show that the proposed methods are robust and fast for mobile implementation. The proposed framework not only improves the quality of the input document images but also increases the readability of the OCR application remarkably.

Despite a current runtime of <1 s, it is still desirable to reduce this time using several mathematical techniques such as a matrix precalculation. Further improvements can be made by utilizing the smartphone's graphics processing unit for parallel computations. We can also improve the fidelity of each preprocessing method for the OCR application by using a classification for dealing with the problem of document image degradation.

## Acknowledgments

This work was supported by Samsung Electronics. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2013R1A2A2A01069181). The authors would like to thank the associate editor and reviewers for their valuable comments.

## References

- M. Laine and O. S. Nevalainen, "A standalone OCR system for mobile cameraphones," in *Proc. of IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, IEEE, Helsinki, Finland (2006).
- M. Pilu, "Extraction of illusory linear clues in perspectively skewed documents," in *Proc. of IEEE Conf. Computer Vision and Pattern Recognition*, pp. I-363–I-368, IEEE, Kauai (2001).
- C. Monnier et al., "Sequential correction of perspective warp in camera-based documents," in *Proc. of Int. Conf. Document Analysis and Recognition*, pp. 394–398, IEEE, Seoul, Korea (2005).
- P. Clark and M. Mirmehdi, "Rectifying perspective views of text in 3D scenes using vanishing points," *Pattern Recognit.* **36**(11), 2673–2686 (2003).
- X. C. Yin et al., "Robust vanishing point detection for mobilecam-based documents," in *Proc. of Int. Conf. Document Analysis and Recognition*, pp. 136–140, IEEE, Beijing, China (2011).
- G. Meng et al., "Metric rectification of curved document images," *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(4), 707–722 (2012).
- H. Muammar and P. Dragotti, "An investigation into aliasing in images recaptured from an LCD monitor using a digital camera," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pp. 2242–2246, IEEE, Vancouver, Canada (2013).
- Y. Tian and S. G. Narasimhan, "Rectification and 3D reconstruction of curved document images," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 377–384, IEEE, Colorado Springs (2011).
- L. Zhang, Y. Zhang, and C. L. Tan, "An improved physically-based method for geometric restoration of distorted document images," *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(4), 728–734 (2008).
- C. L. Tan, L. Xhang, and T. Xia, "Restoring warped document images through 3-D shape modeling," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(2), 195–208 (2006).
- J. Liang, D. DeMenthon, and D. Doermann, "Geometric rectification of camera-captured document images," *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(4), 728–734 (2008).
- Y. Cao, S. Wang, and H. Li, "Skew detection and correction in document images based on straight-line fitting," *Pattern Recognit.* **24**(12), 1871–1879 (2003).
- H. K. Kwag et al., "Efficient skew estimation and correction algorithm for document images," *Image Vision Comput.* **20**(1), 25–35 (2002).
- M. S. Brown and Y. C. Tsui, "Geometric and shading correction for images of printed materials using boundary," *IEEE Trans. Image Process.* **15**(6), 1544–1554 (2006).
- M. S. Brown et al., "Restoring 2D content from distorted documents," *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(11), 1094–1016 (2007).
- Y. C. Tsui and M. S. Brown, "Multi-view document rectification using boundary," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, Minneapolis, Minnesota (2007).
- S. Banerjee et al., "Real-time embedded skew detection and frame removal," in *Proc. IEEE Int. Conf. Image Processing*, pp. 2181–2184, IEEE, Hong Kong (2010).
- S. Lu, B. M. Chen, and C. C. Ko, "Perspective rectification of document images using fuzzy set and morphological operations," *Image Vision Comput.* **23**(5), 541–553 (2005).
- S. Lu, B. M. Chen, and C. C. Ko, "A partition approach for the restoration of camera images of planar and curled document," *Image Vision Comput.* **24**(8), 837–848 (2006).
- V. V. Saveljev et al., "About a moiré-less condition for non-square grids," *J. Disp. Technol.* **4**(3), 332–339 (2008).
- Y. Han, J. Kim, and M. Kim, "A new moiré smoothing method for color inverse halftoning," in *Proc. IEEE Int. Conf. Image Processing*, pp. I-820–I-823, IEEE, Rochester (2002).
- H. Siddiqui and C. A. Bouman, "Training-based algorithm for moiré suppression in scanned halftone images," *Proc. of SPIE* **6498**, 64981D (2007).
- D. V. D. Ville et al., "Suppression of sampling moiré in color printing by spline-based least-squares prefiltering," *Pattern Recognit.* **24**(11), 1787–1794 (2003).
- H. Cao and A. C. Kot, "Identification of recaptured photographs on LCD screens," in *Proc. IEEE Int. Conf. Acoustics Speech and Signal Processing*, pp. 1790–1793, IEEE, Dallas (2010).
- M. Pilu and S. Pollard, "A light-weight text image processing method for handheld embedded cameras," in *Proc. British Machine Computer Vision*, pp. 547–556, BMVA Press, Cardiff, UK (2002).
- Y. Liu and S. Srihari, "Document image binarization based on texture features," *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(5), 540–544 (1997).
- S. C. Pei, M. Tzeng, and Y. Hsiao, "Enhancement of uneven lighting text image using linebased empirical mode decomposition," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, pp. 1249–1252, IEEE, Prague, Czech Republic (2011).
- J. G. Kuk and N. I. Cho, "Feature based binarization of document images degraded by uneven light condition," in *Proc. Int. Conf. Document Analysis and Recognition*, pp. 748–752, IEEE, Barcelona, Spain (2009).
- S. A. Tabataei and M. Bohlool, "A novel method for binarization of badly illuminated document images," in *Proc. Int. Conf. Image Processing*, pp. 3573–3576, IEEE, Hong Kong (2010).
- Z. Shi and V. Govindaraju, "Historical document image enhancement using background light intensity normalization," in *Proc. Int. Conf. Pattern Recognition*, pp. 473–476, IEEE, Cambridge, UK (2004).
- A. Sharma, S. Mahaldar, and S. Banerjee, "Enhanced bleed through removal for scanned document images," in *IS&T/SPIE Electronic Imaging*, p. 787019, SPIE, San Francisco (2011).
- D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *J. Graph. GPU Game Tools* **12**(2), 13–21 (2007).
- P. Clark and M. Mirmehdi, "Finding text regions using localised statistical measures," in *Proc. of British Machine Vision Conf.*, pp. 675–684, BMVA Press, Bristol, UK (2000).
- R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Commun. ACM* **15**(1), 11–15 (1972).
- C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Computer Vision*, pp. 839–846, IEEE, Bombay, India (1998).

**Christian Simon** received his MS degree from Inha University, Republic of Korea, in February 2015. He also received his BS degree in computer science from Bina Nusantara University, Indonesia, in

2012. His research interests include computer vision and computational photography.

**Williem** received his BS degree in computer science from Bina Nusantara University, Indonesia, in 2011. He is currently working toward his PhD degree at Inha University, Republic of Korea. His research interests include 3-D reconstruction, computational photography, and GPGPU.

**In Kyu Park** received his BS, MS, and PhD degrees from Seoul National University in 1995, 1997, and 2001, respectively. From

September 2001 to March 2004, he was with Samsung Advanced Institute of Technology. Since March 2004, he has been with the Department of Information and Communication Engineering, Inha University, as an associate professor. From January 2007 to February 2008, he was an exchange scholar at Mitsubishi Electric Research Laboratories. Since September 2014, he has been a visiting associate professor at MIT Media Lab. His research interests include computer vision, computational photography, and GPGPU. He is a Senior Member of IEEE and a member of ACM.