International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Pre-Processing of Mobile Camera Captured Images for OCR

Pushpinder Singh[1], Dharam Veer Sharma[2]

Submitted: 19/10/2022 **Revised:** 22/12/2022 **Accepted**: 21/01/2023

Abstract-Optical Character Recognition (OCR) systems are nowadays capable of recognizing different printed scripts but the accuracy of any OCR system mainly depends upon the quality of text image. Mobile phones have become the most popular handheld device in this era of technology. A new way to digitize the image is using the mobile camera. Although it is very easy to capture the image with mobile camera but it also brings a lot of challenges. Various challenges in mobile camera captured images are discussed in this paper. Various pre-processing operations need to be performed on the camera captured input image to enhance its quality. This paper also presents the implementation of different pre-processing techniques to improve the quality of camera captured image which can be further used in text recognition.

Keywords: *Pre-processing, Mobile camera captured images, Skew detection and correction, Cropping, Perspective projection, Noise removal, Binarization.*

1. Introduction

Pre-processing of a document image is an important step in Optical Character Recognition (OCR). A paper document can be converted to a digital image into two ways. One way is to scan the object with a flatbed scanner and the other way is to capture the document with a mobile camera. In modern era of technology, mobile phone is not only a device for calling but it has become a part of our life. Due to its versatility and portability, a lot of attention has been put forward for processing of mobile camera captured images. As compared to scanner, there is no contact between the object and the mobile camera while capturing an image. This non-contact nature and portability become the most important advantage of a mobile phone and increased the usability of developed mobile applications. Because of portability of mobile, images are captured in different environment and quality of the captured images varies due to different lighting and weather conditions. There are many challenges in mobile based camera captured image processing which are discussed in this paper. The pre-processing operations including skew detection and correction, noise removal, correction of perspective projection, converting the color image to gray scale image and binarization needs to be performed on a document image captured by mobile camera to improve its quality. The captured input image needs to be improved as the accuracy of text recognition depends upon the quality of input image.

In this paper, various techniques of skew detection have been implemented and compared to correct the input image. Noise removal techniques have been used to smoothen the image and to remove salt and pepper noise which is the most prevalent noise in images. After cropping and correction perspective distortion, different local and global binarization algorithms have been implemented to covert the input image into binary image. These operations are implemented by using OpenCV (Open-Source Computer Vision Library) which is an open-source library for image processing.

2. Related Work

Extensive research work has been carried out regarding pre-processing operations on document image but every technique comes with some advantages as well as some limitations. Various local and global binarization techniques for gray scale images such as Fixed Threshold, Otsu, Kitter, Niblack, Adaptive, Sauvola and Bernsen have been discussed and compared by Puneet and Garg in **[1]**. Jyotsna et al. have reviewed different binarization methods and performance metrics such as PSNR, F-Measure, NRM and MPM in **[2]**. Many researchers have proposed different methods for detecting a skew angle at which an image is rotated. O'Gorman **[3]** has presented three different techniques for detecting skew in document image namely Projection Profile Analysis, Hough Transform and Nearest Neighbor. These methods have been evaluated and compared by A. Al-Khatatneh et al. **[4]**. Projection Profile method was first proposed by Postl et al. **[5]** and later some

[1]Research Scholar, Department of Computer Science, Punjabi University, Patiala, Punjab, India, pushpinder@pbi.ac.in

[2]Department of Computer Science, Punjabi University, Patiala, Punjab, India, dveer72@hotmail.com

other variations have been introduced in **[6-8]** to reduce the computational time. Hashizume et al. **[9]** found all the connected components and computed the direction that connects the text and thus introduced the nearest neighbor clustering algorithm. Various filters can be applied to remove different types of noise **[10]**. A novel technique based on kFill algorithm is implemented in single-pass rather than iterative processing to remove the salt and pepper noise by Chinnasarn et al. **[11].** The median filter is one of the popular non-linear filter which is used to remove salt and pepper noise by Maheshwari and Radha **[12].**

3. Challenges in Mobile Camera Captured Image Processing

There are many challenges in mobile based camera captured image processing which are discussed below:

3.1 Geometric Distortion

Geometric distortion is one of the most common problems in images captured by a mobile camera. Mainly, there are two types of geometric distortions: skew distortion and perspective distortion.

Skewness Distortion

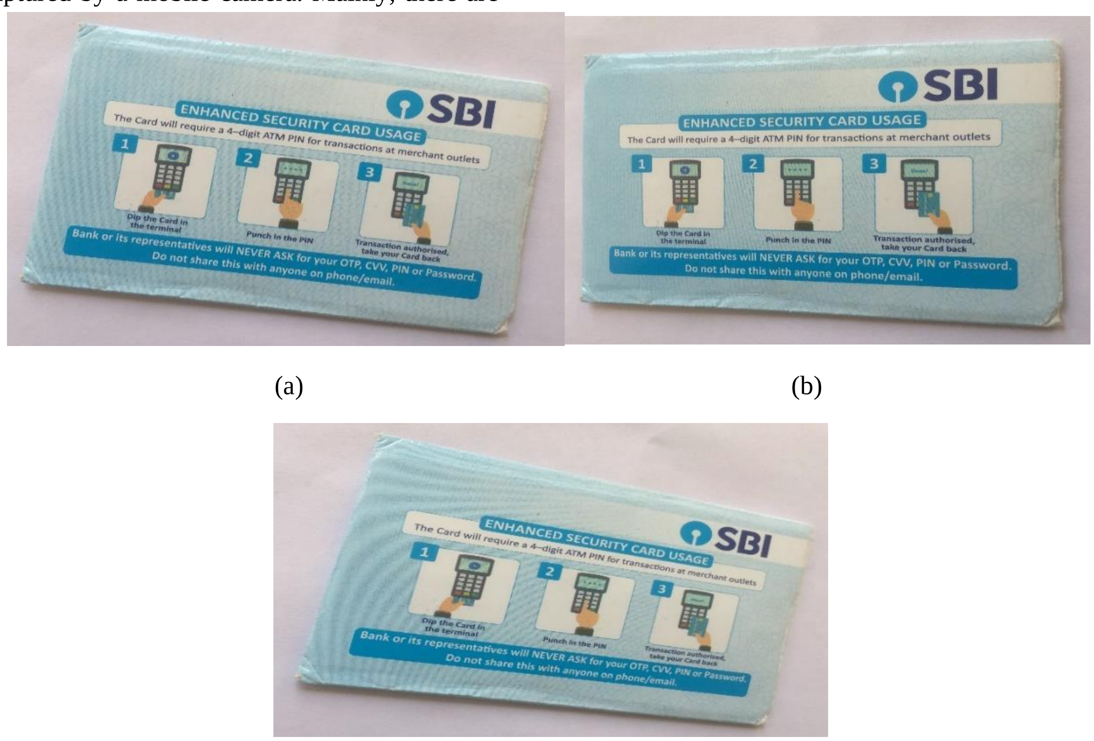
An image is said to be skewed if the image is tilted to a particular angle. It commonly happens, while capturing an image, when axes of the object and mobile camera are not parallel to each other. Figure 1(a) shows the skewed image captured with a mobile phone.

Perspective Distortion

Perspective distortion occurs when the plane of an object is not parallel to the capturing device. An image with perspective distortion is shown in Figure 1(b).

Skewness and Perspective Distortion

Every image captured with mobile camera is commonly less or more skewed and perspective distorted. Both skew and perspective distortion happen in image in most of the cases. An image with skew and perspective distortion is shown in Figure 1(c).



(a) (b)

(c)

Fig. 1 (a) Image with skewness distortion (b) Image with perspective distortion (c) Image having both skewness and perspective distortion.

3.2 Variable Lighting Conditions

There can be different lighting conditions which affect the quality of the captured image. Images captured inside a closed room called *Indoor images* which are illuminated with artificial light. Different frequencies of this light produce different images as shown in Figure 2(a).

Outdoor images captured in an open environment are illuminated with natural light. There can be different weather conditions which affect the quality of captured images. Sometimes, images are captured with both artificial light and natural light. Same images captured in different lighting conditions are shown in Figure 2(b).

The *uneven illumination and shadow* are the other parameters which affect the quality of the image besides lighting conditions as shown in Figure 2(c).

The *mobile camera flash* can be used to solve the illumination problem but it may cause other type of problem. If the mobile flash is used while capturing an image, the focus of the view on image become the brightest and text in center of the image blurred and difficult to recognize as shown in Figure 2(d).

(a) (b)

(c) (d)

Fig. 2 (a) Captured image inside a room (Indoor image) (b) Captured image in natural light (Outdoor image) (c) Image with uneven illumination and shadow (d) Image captured using camera flash.

3.3 Blurring Effect

Blurring effect distorts the captured image in such a way that the detail of information in image is lost, particularly, edge information. Blurring may occur due to improper functioning of mobile camera, poor lighting conditions or the mobile camera is not stationary while capturing an image. Figure 3 shows a blurred image.

Fig. 3 Image with blurring effect.

4. Pre-Processing of Mobile Camera Captured Images

The steps in pre-processing of mobile camera captured image including skew detection and correction, noise removal, cropping and correction of perspective projection, converting the color image into gray scale image and binarization are shown in Figure 4.

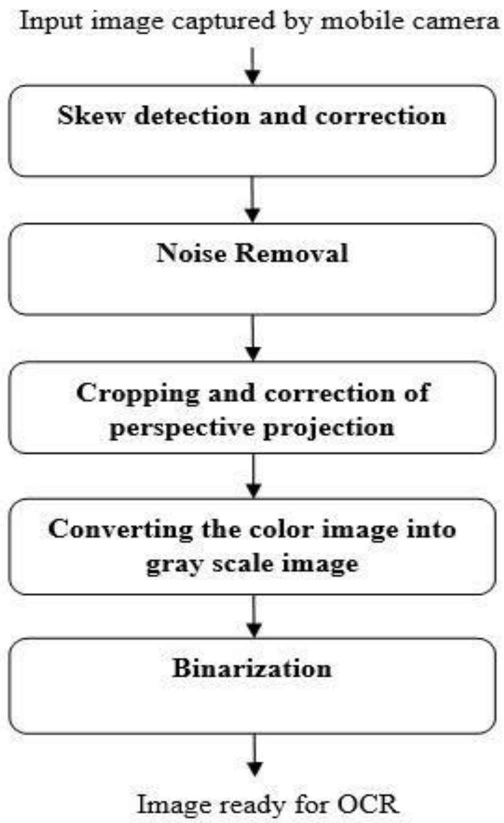


Fig. 4 Steps in pre-processing of Mobile camera captured image

4.1 Skew Detection and Correction

Skew detection and correction of camera captured image is an important and problematic step which is further used in segmentation and classification steps of OCR. The skew in image can be of three types: *Global skew*: When a document image has a common skew angle, *Multiple skew*: When a document image has different skew angle and *Non-uniform text line skew*: When document has different angles in single line. Most frequently used skew detection and correction algorithms are implemented as follows:

4.1.1 Projection Profile Analysis

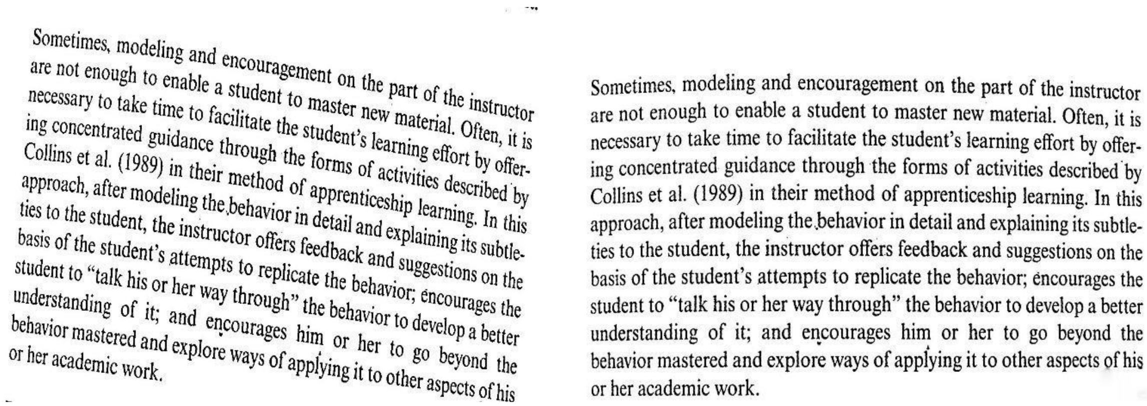
Projection profile analysis is one of the popular and straightforward skew detection techniques which uses horizontal and vertical projection profiles to detect the skew angle. Firstly, the input binary image is rotated from 0 degree to 180 degrees at an interval of 5 degrees. At each rotation, vertical histogram is calculated i.e. alongside the columns of the image. After that variance of histogram is computed in each iteration and the skew angle is selected where the variance is minimum.

4.1.2 Hough Transform

Hough Transform is widely used technique in computer vision and image analysis which is first introduced by Richard Duda and Peter Hart in 1972. In this implementation, first the edges of the input image are detected using Canny edge detector algorithm. Then Hough lines are calculated using *HoughLines()* method of *OpenCV*. The angle is calculated of each line, computed the average of each angle and the image is rotated at that particular skew angle.

4.1.3 Nearest Neighbor

Another commonly used technique is Nearest Neighbor based on finding the connected component in document image. In this method, the input image is first inverted using bitwise not operation. A rectangle is calculated over the object of connected components using the *Imgproc.minAreaRect()* function of OpenCV. This function actually creates a box or rectangle around the pixels which are close to each other and form a cluster. It basically checks for the neighbors from a point to another and wraps all the neighbors into a box. After that the angle of rectangle is calculated and the image is rotated accordingly. The results of skew detection and correction technique are shown in Figure 5.


Sometimes, modeling and encouragement on the part of the instructor are not enough to enable a student to master new material. Often, it is necessary to take time to facilitate the student's learning effort by offering concentrated guidance through the forms of activities described by Collins et al. (1989) in their method of apprenticeship learning. In this approach, after modeling the behavior in detail and explaining its subtleties to the student, the instructor offers feedback and suggestions on the basis of the student's attempts to replicate the behavior; encourages the student to "talk his or her way through" the behavior to develop a better understanding of it; and encourages him or her to go beyond the behavior mastered and explore ways of applying it to other aspects of his or her academic work.

Sometimes, modeling and encouragement on the part of the instructor are not enough to enable a student to master new material. Often, it is necessary to take time to facilitate the student's learning effort by offering concentrated guidance through the forms of activities described by Collins et al. (1989) in their method of apprenticeship learning. In this approach, after modeling the behavior in detail and explaining its subtleties to the student, the instructor offers feedback and suggestions on the basis of the student's attempts to replicate the behavior; encourages the student to "talk his or her way through" the behavior to develop a better understanding of it; and encourages him or her to go beyond the behavior mastered and explore ways of applying it to other aspects of his or her academic work.

(a) (b)

Fig. 5 Figure 5(a) and 5(b) shows the input image with skew and rotated output image respectively.

4.2 Noise Removal

The main source of noise in digital images arises during image acquisition. Gaussian and Salt-and-pepper noise are the most common types of noise with which the captured image is contaminated. Here Gaussian filter and Median filter are implemented in order to reduce the noise and to smoothen the image.

4.2.1 Gaussian Filter

Gaussian filter is a linear filter used to reduce the noise in image. It is commonly used to blur the edges and reduce the contrast of an image. The following *OpenCV function* is used to implement Gaussian filter:

Imgproc.GaussianBlur(src,des, ***new*** *Size(5,5),3);*

In the above function, *src* represents an input image, *des* represents output image, ***new*** *Size(5,5)* is the size of kernel and *3* is the Gaussian kernel standard deviation in X direction. The results are shown in Figure 6.

(a)

(b)

Fig. 6 Figure 6(a) and 6(b) shows the input image and output image respectively.

4.2.2 Median Filter

Median filter is one of the most useful and effective order-statistic nonlinear filter to reduce the impulse noise. The impulse noise is also called salt-and–pepper noise which looks like black and white dots superimposed on image. As its name implies, median filter replaces the value of the pixel by the median of gray levels in the neighborhood of that pixel. The following *OpenCV function* is used to apply median filter:

Imgproc.medianBlur(src, des, 3);

In the above function, *src* represents an input image, *des* represents output image and *3* is the size of kernel. The results are shown in Figure 7.

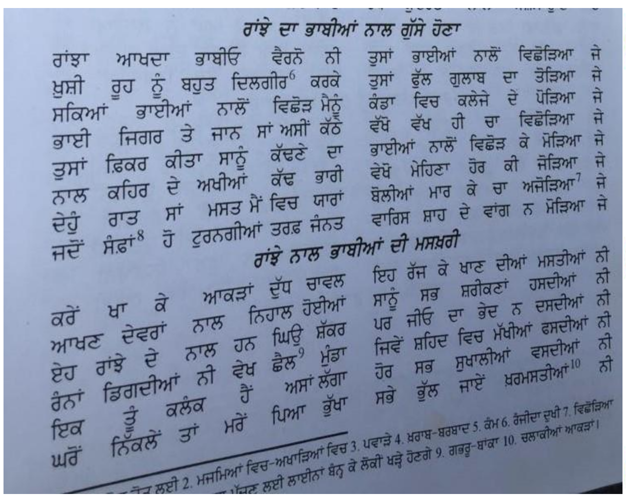
(a)

(b)

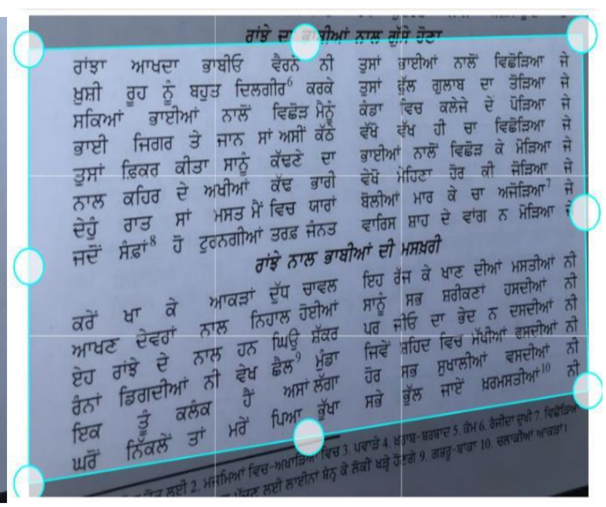
Fig. 7 Figure 7(a) and 7(b) shows the input image with noise and output image respectively.

4.3 Cropping and Correction of Perspective Projection

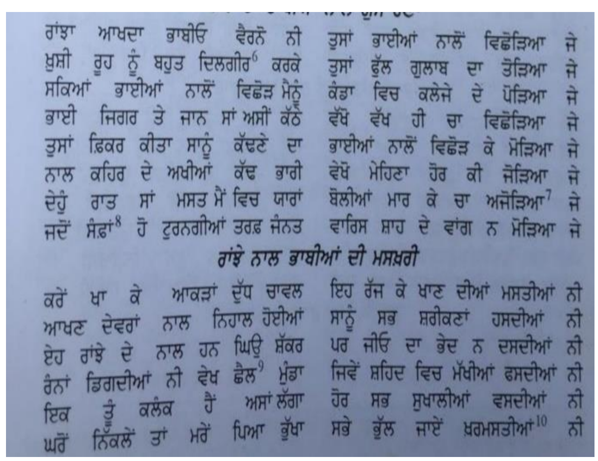
SmartCropper library is used to crop and to correct the perspective distortion. The library uses the canny edge detection algorithm of OpenCV library which identifies the border and crop the image. The algorithm uses the perspective transform to crop and correct the selection to restore the front image as shown in Figure 8.



(a)



(b)



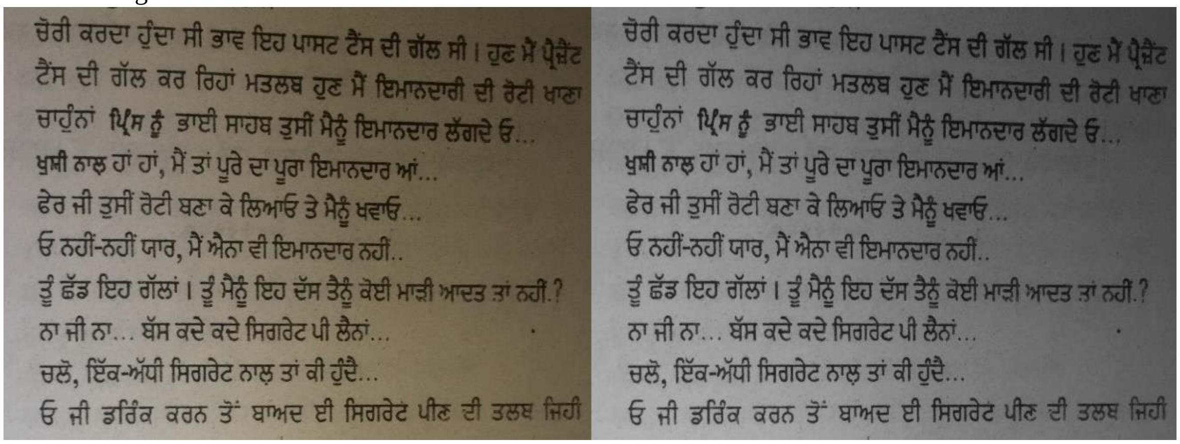
(c)

Fig. 8 Figure 8(a), 8(b) and 8(c) shows the input image with perspective projection, cropping the image and output image respectively.

4.4 Converting Color image into gray scale image

The next step in pre-processing operations is to convert the captured image into grayscale image before further processing. Input image is converted to grayscale image by using *cvtColor()* function of *Imgproc* class of *OpenCV* library as shown in Figure 9.

```
Imgproc.cvtColor(source,
destination, Imgproc.COLOR_RGB2GRAY);
```



ਚੋਰੀ ਕਰਦਾ ਹੁੰਦਾ ਸੀ ਭਾਵ ਇਹ ਪਾਸਟ ਟੈਂਸ ਦੀ ਗੱਲ ਸੀ। ਹੁਣ ਮੈਂ ਪ੍ਰੈਜ਼ੈਂਟ
ਟੈਂਸ ਦੀ ਗੱਲ ਕਰ ਰਿਹਾਂ ਮਤਲਬ ਹੁਣ ਮੈਂ ਇਮਾਨਦਾਰੀ ਦੀ ਰੋਟੀ ਖਾਣਾ
ਚਾਹੁੰਨਾਂ ਪ੍ਰਿੰਸ ਨੂੰ ਭਾਈ ਸਾਹਬ ਤੁਸੀਂ ਮੈਨੂੰ ਇਮਾਨਦਾਰ ਲੱਗਦੇ ਓ...
ਖ਼ੁਸ਼ੀ ਨਾਲ਼ ਹਾਂ ਹਾਂ, ਮੈਂ ਤਾਂ ਪੂਰੇ ਦਾ ਪੂਰਾ ਇਮਾਨਦਾਰ ਆਂ...
ਫੇਰ ਜੀ ਤੁਸੀਂ ਰੋਟੀ ਬਣਾ ਕੇ ਲਿਆਓ ਤੇ ਮੈਨੂੰ ਖਵਾਓ...
ਓ ਨਹੀਂ-ਨਹੀਂ ਯਾਰ, ਮੈਂ ਐਨਾ ਵੀ ਇਮਾਨਦਾਰ ਨਹੀਂ..
ਤੂੰ ਛੱਡ ਇਹ ਗੱਲਾਂ। ਤੂੰ ਮੈਨੂੰ ਇਹ ਦੱਸ ਤੈਨੂੰ ਕੋਈ ਮਾੜੀ ਆਦਤ ਤਾਂ ਨਹੀਂ.?
ਨਾ ਜੀ ਨਾ... ਬੱਸ ਕਦੇ ਕਦੇ ਸਿਗਰੇਟ ਪੀ ਲੈਨਾਂ...
ਚਲੋ, ਇੱਕ-ਅੱਧੀ ਸਿਗਰੇਟ ਨਾਲ਼ ਤਾਂ ਕੀ ਹੁੰਦੈ...
ਓ ਜੀ ਡਰਿੰਕ ਕਰਨ ਤੋਂ ਬਾਅਦ ਈ ਸਿਗਰੇਟ ਪੀਣ ਦੀ ਤਲਬ ਜਿਹੀ

(a) (b)

Fig. 9 (a) Input color image (b) Gray scale image

4.5 Binarization

Binarization is the process of converting a document image into bi-level image and image pixels are separated and grouped into two parts (e.g. black and white). Foreground text and background of an image can easily be identified after binarization. In Global binarization, a single threshold value is applied to the whole image while in Local binarization, a different threshold value is calculated for every pixel in the image. Binarization is the basis of segmentation and classification as the quality of the binarized image affects the accuracy of text recognition. Fixed Thresholding, Otsu, Niblack and Sauvola methods are implemented for binarization.

4.5.1 Fixed Thresholding Method

A single threshold value is used in fixed thresholding method for the whole binary image. A pixel value of binary image is set to 1 if it is greater than the threshold value otherwise it is set to 0. This method is implemented using the following function of OpenCV:

Imgproc.threshold(src, src,50,255,0);

The output image using fixed threshold value of 50 is shown in Figure 10(b).

4.5.2 Otsu Method

Otsu method is also a local binarization method which is better than fixed threshold method because it has the ability to choose an optimal threshold value. Threshold value divide the image into two different classes. The threshold value is chosen in a way that between class variance is maximized and within class variance is minimized. The weighted within class variance of two classes is shown in Eq. 1.

$$\sigma^2 p(t) = p_1(t)\sigma_1^2 + p_2(t)\,\sigma_2^2(t) \quad (1)$$

The following function is used to implement this method and the output image after applying Otsu method is shown in Figure 10(c).

*Imgproc.threshold(src, src, 0, 255,
Imgproc.THRESH_OTSU);*

4.5.3 Niblack Method

Niblack method is a local thresholding method which calculates the threshold value for every pixel by sliding a rectangular window over the entire image. The threshold value is based on mean and standard deviation of all pixels in window. The threshold $T(i,\ j)$ value is calculated as shown in Eq. 2.

$$T(i,j) = \mu + k * \sigma \quad (2)$$

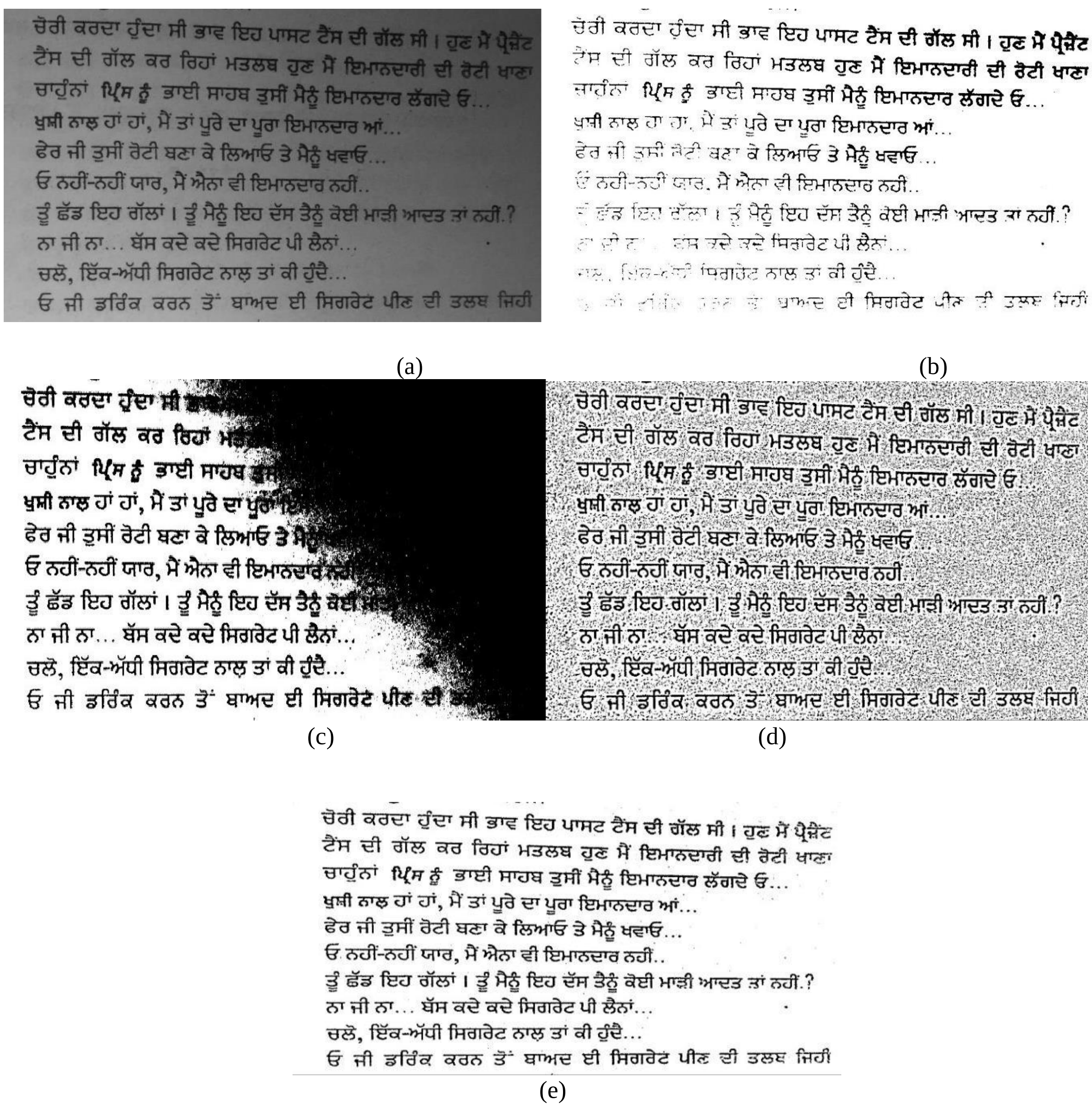
In Eq. 2, μ and σ represents mean and standard deviation of the window respectively and k is a constant. *Ximgproc.niBlackThreshold()* function of OpenCV with binarization method *BINARIZATION_NIBLACK* is used to implement this algorithm and the output image is shown in Figure 10(d).

4.5.4 Sauvola Method

Sauvola method is a modified and improved version of Niblack method. The threshold $T(i, j)$ value is calculated as shown in the Eq. 3.

$$T(i,j) = \mu * (1 - k * (\frac{\sigma}{R})) \quad (3)$$

In Eq. 3, μ is the mean and σ is the standard deviation of the window. The value of k and R is 0.5 and 128 respectively as it affects the quality of image. *Ximgproc.niBlackThreshold()* function of OpenCV with binarizatin method *BINARIZATION_SAUVOLA* and the output image after applying Sauvola method is shown in Figure 10(e).



(a) (b) (c) (d) (e)

Fig. 10 Figure 10(a), 10(b), 10(c), 10(d) and 10(e) shows the input text image, binarized image using Fixed Thresholding (Theshold Value=50), Otsu method, Niblack method and Sauvola methods respectively.

5. Results And Discussions

The three most commonly used techniques namely Projection profile analysis, Hough transform and Nearest neighbor have been implemented to detect the skew angle of the captured image. Although the projection profile technique is simple and easy to implement but it is very time consuming and the computation cost is very high. Hough transform technique has a very high accuracy in detecting the skew angle and takes less time to execute as compared to projection profile analysis. Nearest neighbor method is used to detect any range of skew angles and to detect the skew angle of the document image containing different types of font size and layouts. The limitation of this method is that the accuracy depends upon the quality of the binary image. All three methods detected the skew angle and the input image is rotated at the calculated angle.

Two filters namely Median filter and Gaussian filters have been implemented to reduce the unwanted noise and to smoothen the binary image. Median filter reduces the noise while keeping the edges relatively sharp as compared to Gaussian filter. An open-source library based on canny edge detection algorithm is used to crop the image and to correct the perspective projection of image.

The captured input color image has been converted into gray scale image which is required for further processing in text recognition. In binarization, four local and global binarization methods including Fixed thresholding, Otsu, Niblack and Sauvola have been implemented. In Fixed thresholding method, it is very difficult to choose an optimal threshold value. Although Otsu method gives good results for the images of good quality but it does not perform well for images having uneven illumination and shadow. Niblack method performs well and can recognize the text in the image as foreground but it also generates large quantity of binarization noise in non-text region of image. Sauvola method gives very good results for the text binary images even with shadow as compared to other methods.

All of these pre-processing operations have been implemented using the optimized functions and algorithms of OpenCV library which takes less time to execute as compared to other traditional approaches.

6. Conclusion

Various challenges in mobile camera captured image processing have been discussed in this paper. Different pre-processing operations are applied to improve the quality of the captured text image which will be further used in OCR. Different techniques of skew detection and correction, noise removal, cropping and correction of perspective projection and binarization have been implemented and compared.

References

[1] Puneet & Garg, Naresh. (2013). Binarization Techniques used for Grey Scale Images. International Journal of Computer Applications. 71. 8-11, 2013.

[2] Jyotsna, S. Chauhan, E. Sharma and A. Doegar, "Binarization techniques for degraded document images — A review," 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2016, pp. 163-166.

[3] O'Gorman, L.: "The document spectrum for page layout analysis,"IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 15, no. 11,1993, pp. 1162–1173.

[4] Al-Khatatneh, S. A. Pitchay and M. Al-qudah, "A Review of Skew Detection Techniques for Document," 2015 17th UKSim-AMSS International Conference on Modelling and Simulation (UKSim), Cambridge, UK, 2015, pp. 316-321.

[5] W. Postl, "Detection of linear oblique structures and skew scan in digitized documents," In Proceedings of the 8th International Conference on Pattern Recognition, pp. 687-689. 1986.

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in Advances in neural information processing systems, pp. 91–99, 2015.

[7] Lei Zhang, Fan Yang, Yimin Daniel Zhang, and Ying Julie Zhu, "Road crack detection using deep convolutional neural network," in 2016 IEEE international conference on image processing (ICIP). IEEE, pp. 3708–3712, 2016.

[8] Tianmei Guo, Jiwen Dong, Henjian Li, and Yunxing Gao, "Simple convolutional neural network on image classification," in 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), IEEE, pp. 721–724, 2017.

[9] Wolfgang Postl, "Detection of linear oblique structures and skew scan in digitized documents," in Proc. Int. Conf. on Pattern Recognition, pp. 687–689, 1986.

[10] R.C. Gonzalez and R.E. Woods, Digital Image Processing 26, Sept. 1992., Addison- Wesley, 1992.

[11] K. Chinnasarn, Y. Rangsanseri and P. Thitimajshima, "Removing salt-and-pepper noise in text/graphics images," IEEE. APCCAS 1998. 1998 IEEE Asia-Pacific Conference on Circuits and Systems. Microelectronics and Integrating Systems. Proceedings (Cat. No.98EX242), Chiang Mai, Thailand, 1998, pp. 459-462.

[12] D. Maheshwari, V. Radha," Noise removal in compound image using median filter"," International Journal of Advanced Trends in Computer Science and Engineering", January 2010, pp. 1359-1362.