

## Shading Surface Estimation using Piecewise Polynomials for Binarizing Unevenly Illuminated Document Images

Xiaohua Zhang<sup>1\*</sup>, Yuelan Xin<sup>2</sup>, Ning Xie<sup>3</sup>, Wenbo Jiang<sup>4</sup>, Yue Zhao<sup>1</sup>

<sup>1</sup> Hiroshima Institute of Technology, Hiroshima, 731-5193, Japan

<sup>2</sup> Qinghai Normal University, Xining, 810008, China

<sup>3</sup> Tokyo Institute of Technology, Tokyo, 152-8550, Japan

<sup>4</sup> Xihua University, Chengdu, 610039, China

\* Corresponding author's Email: [zhxh@cc.it-hiroshima.ac.jp](mailto:zhxh@cc.it-hiroshima.ac.jp)

---

**Abstract:** As a pre-processing step for document analysis, binarizing document image is often performed. The binarization is regarded as a necessary step to separate text and background. When document images are acquired under poor lighting circumstances, the images are soiled with uneven shading. This paper presents an improved approach for binarizing document images which are illuminated unevenly. We employ 2D polynomials with order 3 to fit very complicated uneven shading surface of document images. The document image is divided into several blocks overlapped partially with each other to reduce the discontinuity between polynomials. To improve the estimation accuracy, we normalize pixel coordinates to establish a stable equation system for computing the polynomial coefficients. After removing the uneven shading, most part of the background is filled with the maximum of surface and then a global thresholding is applied. In the experiment section, we demonstrate that the binarization results are satisfied, compared with other methods.

**Keywords:** Document image; Uneven shading; Binarization; Piecewise polynomial; Surface fitting.

---

### 1. Introduction

Although document image binarization has a long history for document image analysis such as OCR, it is still an interesting field in image processing and pattern recognition. Many document images are photoed or scanned from books, magazines, maps or engineering drawings etc. and are converted to gray level for analysis. To analyze this kind of images, the prerequisite called document image binarization is essential to extract texts or drawings. The binarization classifies all pixels into "dark" or "light" categories to divide document image into foreground layer containing texts or drawings and background layer containing no texts or drawings.

When digitizing document from media such as paper with a uniform gray level under well-conditioned illumination, the image probably contains two classes

of pixels and the histogram owns bi-modality with two peaks and a valley. The image can be easily binarized with a global thresholding such as Otsu's method [1] or an improved Otsu's method called Kittler's method [2]. However, when the media are illuminated under a bad condition, the acquired image contains uneven non-uniform shading. The image quality becomes poor with variable background intensity. In this case, the global thresholding approach with a fixed threshold value will fail, since the histogram does not contain bi-modality. Figure 1(a) shows an original document image with non-uniform shading; and (b) is a binarized result of Otsu's method. It is observed that most part of non-text is categorized into foreground, while text is incorrectly classified into background.

Since the global thresholding method cannot binarize non-uniformly shaded document image, many lo-

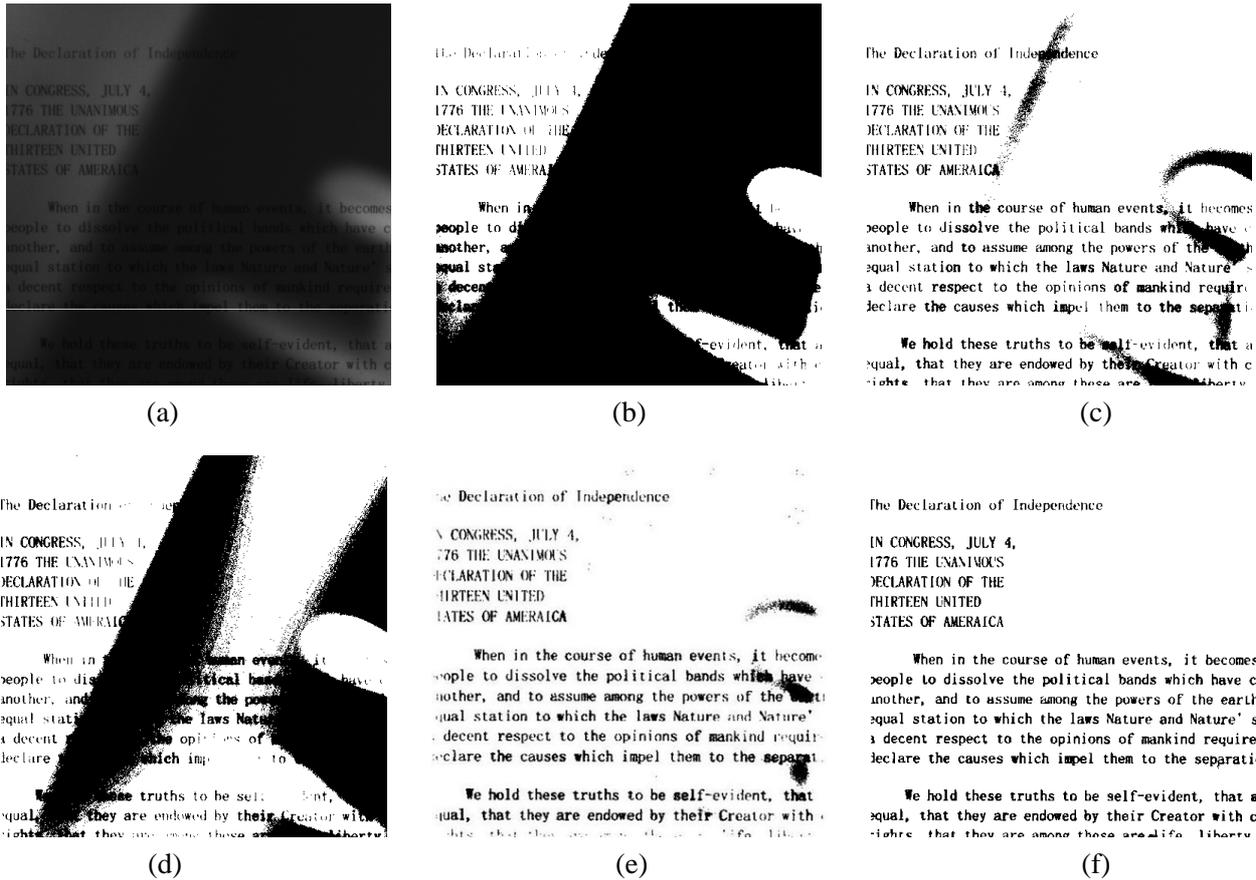


Figure 1 Experimental results: (a) an original image with complicated uneven shading; (b) result obtained by using Otsu's method; (c) result acquired by using Sauvola's method; (d) result by using Lu's method with a single polynomial; (e) result by using Wen's method with curvelet transform; (f) Our final thresholding result using piecewise polynomials.

cal thresholding techniques have been proposed. Several early reported local adaptive thresholding techniques [3, 4, 5] computed local threshold values by using mean and standard deviation of pixels within a neighborhood window centered at the current pixel. These window-based methods can be accelerated by using integral image [6, 7] while maintaining the binarized quality unchanged. However, the problem of window-based methods is that the binarized quality excessively depends on the neighborhood window size. Moreover, when the foreground contains texts or drawings with thick width or large area, the print information will be lost. Figure 1(c) is a result produced by using Sauvola's technique. Some of the background is segmented into foreground incorrectly.

In this work, we develop an approach for binarizing document image with complicated background caused by bad illumination. We focus on document images with uneven shading. For a document image with a uniform background, we simply consider the background as a plane when viewing an image in a 3D

space that is constructed with  $X, Y$  axes aligned to image's horizontal and vertical orientations and  $Z$  axis being the intensity axis. In an unevenly shaded document image such as shown in Figure 1(a), the intensities of the complicated background have variations and cannot be viewed as a simple plane. We use piecewise polynomial functions to estimate the uneven shading, then reconstruct the image almost without non-uniform shading. Finally, a global thresholding technique is applied for binarization.

## 2. Related work

Over past several years, some thresholding methods by applying estimating background surface have been reported. Krzysztof et al. [8] used a local polynomial surface to binarize fingerprint by fitting the surface to pixels within a small neighborhood window. They showed that their method acquired good results in the context of fingerprint recognition. Gatos et al.[9] estimated document background surface, but the esti-

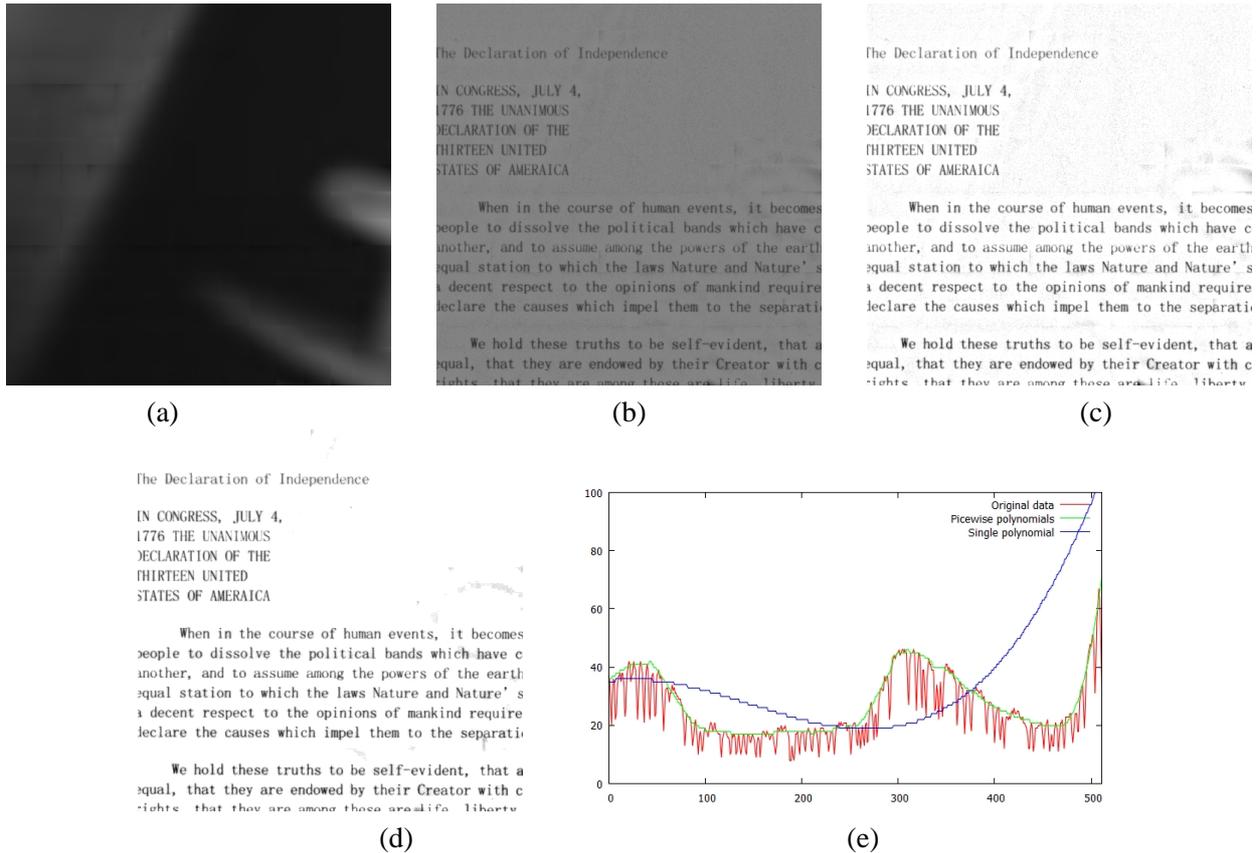


Figure 2 Experimental process: (a) estimated surface map of image in Figure 1(a); (b) reconstructed image in the first process; (c) normalized image in the second process; (d) after background noises removed; (e) curve fitting examples. More detailed explanation can be found in the text.

mation is based on the binarized result produced by Sauvola's method. Lu also reported two similar approaches for binarizing badly illuminated document images through photometric correction[10] and shading estimation[11]. Later, degraded document image binarization was proposed [12] by using a two-pass (one for rows and the other for columns) global polynomial surface fitting to the background. Recently Wen reported a new binarization method [13] for non-uniform illuminated document images. The document image is transformed into Curvelet coefficients, then adaptive and nonlinear functions are applied for histogram adjusting and denoising. Finally, the inverse Curvelet transform is used. One of the results is shown in Figure 1(e). There are also many other references which can be found in a survey [14].

Since Lu's method [10] is most related to ours, we briefly introduce it as follow. A two-round global polynomial surface fitting is applied to estimate document background. Suppose the background is brighter than the text or drawings, then the first round fits a poly-

mial surface to the intensities of pixels of the whole image. Pixels which are below the estimated surface for a variable threshold are removed. The removed pixels are probably as foreground as texts or drawings. Then, the second round fits another polynomial surface to the left pixels which are considered to contain most of background pixels. The estimated polynomial surface in the second round is expected to be more accurate to fit the non-uniformly shaded background. The document image is then normalized using the estimated shading variations which are roughly compensated. Finally, a global thresholding such as Otsu's method is used to binarize the compensated image. Figure 1(d) shows the binarized result using a single polynomial. As shown in the figure, a single polynomial surface is unable to fit complicated background such as the one in Figure 1(a). Most part of the background is classified into "dark" category incorrectly. Although lot of pixels belong to text, they are above the incorrectly estimated surface and are classified into "light" category.

Motivated by the above work, we propose a method by using piecewise polynomials to fit the complicated background surface. The image is divided into several blocks and a polynomial fitting is applied into each block. To alleviate the discontinuity between each polynomial surface, the blocks are partially overlapped with each other sharing common region. Moreover, after image reconstruction, we compute narrow Gaussian blocks [15] and fill them with the maximum of the estimated surface, since the estimation is not always a complete fitting. This process produces an image with almost uniform shading and finally a global thresholding such as Otsu's method is applied to binarize the final reconstructed image.

### 3. Piecewise polynomial surface fitting and binarization

To binarize unevenly illuminated document image such as Figure 1(a), the complicated background is estimated first. The background of document image with uneven shading is unable to be regarded as a surface varying gently. Therefore it cannot be fitted with a single global polynomial function. Since the background changes severely, we use piecewise polynomials to fit the background surface.

#### 3.1 Polynomial fitting using normalized coordinates

To estimate complicated shading background, we divide the document image into several blocks and fit each block to a polynomial. These blocks are overlapped partially to reduce the discontinuity between polynomials. The block size is decided by the number of blocks which depend on the complexity of the uneven shading. Figure 3 illustrates a layout of blocks for an image. The rightmost and bottommost blocks should contain much enough pixels to guarantee a stable polynomial surface estimation.

Each block is fitted with a least square polynomial surface with order 3 as shown in equation (1):

$$P(x, y) = \sum_{k+l=0}^d a_{kl} x^k y^l \quad (d = 3). \quad (1)$$

Suppose pixel value in the given document image is represented as  $f(x, y)$  which is scaled to range  $[0, 1]$ , we note each pixel coordinate as  $(x_i, y_i)$  for convenience where the  $i$  is an index of pixels. We hope computing the coefficients in equation (1) to fit the polynomial to the intensity surface. This is a typical

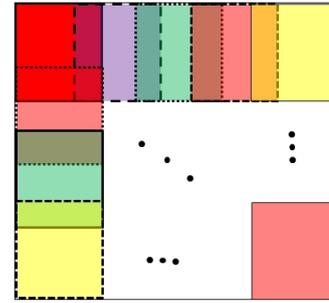


Figure 3 An illustration shows how to divide image into partially overlapped blocks.

linear regression, and the fitting becomes a minimization problem with a cost function as equation (2):

$$\begin{aligned} E(\mathbf{a}) &= \sum_i (P(x_i, y_i) - f(x_i, y_i))^2 \\ &= \sum_i \left( \sum_{k+l=0}^d a_{kl} x_i^k y_i^l - f(x_i, y_i) \right)^2. \end{aligned} \quad (2)$$

The sum is computed over all pixels in the block. We make the partial derivatives of the cost function respect to the coefficients be 0:

$$\nabla_{\mathbf{a}} E(\mathbf{a}) = \mathbf{0} \quad (3)$$

There are 10 coefficients in the polynomial as shown in equation (1) which can be written as a column vector  $\mathbf{a}^T = (a_{00}, a_{10}, a_{01}, a_{20}, a_{11}, a_{02}, a_{30}, a_{21}, a_{12}, a_{03})$ . We rewrite this vector as  $\mathbf{a}^T = (a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9)$  in a single index form with continuous index number for convenience.

We then have 10 equations and stack them together to form a linear system in matrix form:

$$\sum_i \begin{pmatrix} 1 & x_i & \dots & x_i y_i^2 & y_i^3 \\ x_i & x_i^2 & \dots & x_i^2 y_i^2 & x_i y_i^3 \\ \dots & \dots & \dots & \dots & \dots \\ x_i y_i^2 & x_i^2 y_i^2 & \dots & x_i^2 y_i^4 & x_i y_i^5 \\ y_i^3 & x_i y_i^3 & \dots & x_i y_i^5 & y_i^6 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_8 \\ a_9 \end{pmatrix} = \sum_i \begin{pmatrix} f(x_i, y_i) \\ x_i f(x_i, y_i) \\ \vdots \\ x_i y_i^2 f(x_i, y_i) \\ y_i^3 f(x_i, y_i) \end{pmatrix}. \quad (4)$$

The equation can be written in a concise form as  $\mathbf{X}\mathbf{a} = \mathbf{b}$ , where  $\mathbf{X}$  represents a  $10 \times 10$  matrix at the left of

equation,  $\mathbf{a}$  is a vector consisting of the coefficients and  $\mathbf{b}$  is the right part of equation (4). Our matrix is only  $10 \times 10$  and only consumes little memory. By solving the above linear system of equation, we can obtain the coefficient vector  $\mathbf{a}$ .

By observing the equation (4), the matrix  $\mathbf{X}$  is poor conditioned and is lack of homogeneity in the image coordinates. For example, the left-up most element is 1, while the bottom-right most element is  $y^6$ . With a moderate size, such as  $500 \times 500$  pixels, for a pixel located at the bottom, the element will become about  $1.56 \times 10^{+16}$ . Summing over all pixel locations will result in a matrix for which the differences between elements are extremely large. As pointed out by Hartley [16], this kind imbalance of the matrix will cause linear system being unstable, thus results in inaccurate solution.

To improve the condition of matrix  $\mathbf{X}$ , all pixel coordinates are first translated so as to bring the centroid of coordinates to the origin. Then we use an isotropic scaling to simply make all pixels locate in the range  $[-\sqrt{2}, \sqrt{2}]$ . The transformation is represented as  $T$ :

$$T = \begin{pmatrix} \frac{2\sqrt{2}}{w} & 0 & -\sqrt{2} \\ 0 & \frac{2\sqrt{2}}{h} & -\sqrt{2} \\ 0 & 0 & 1 \end{pmatrix} \quad (5)$$

where  $w$  is the width and  $h$  is the height of block. Before the homogeneous coordinate  $(x, y, 1)$  is substituted into the matrix, it is transformed as  $(x', y', 1)^\top = T(x, y, 1)^\top$ .

To show what is different by using single polynomial and piecewise polynomial function, we take one scanline data from the original image and the fitting result as illustrated in Figure 2(e). The red curve means raw data as shown in Figure 1(a) with gray line; the blue one is a fitting result by using single polynomial, while the green one shows the result by using our proposed piecewise polynomials. It is obvious that our method fits the raw data more accurately.

### 3.2 Surface map computation

After piecewise polynomial surface fitting, we compute the background surface map  $M_s(x, y)$ . Again, the coordinates are transformed using equation (5) before computing surface map. We simply compute the polynomials as the map values. On the overlapped regions between two blocks, an average is calculated as map value. A smoothing method is applied to the surface map to further reduce discontinuity.

Since the surface map is computed from the piecewise polynomials which are estimated by using both “dark” and “light” pixels, the dark pixels in the foreground have negative effect on the estimation. To estimate the polynomials, the potential foreground (text) pixels should be rejected. The potential foreground consists of pixels which are far below the estimated surface. These pixels are represented as a pixel location set with a threshold  $2m/3$ :

$$F_p = \{(x, y) : M_s(x, y) - f(x, y) > \frac{2m}{3}\} \quad (6)$$

where  $m$  is the mean of the differences between estimated surface value and pixel intensity whose value is smaller than the value of the surface. This process makes sure that the inked texts or drawings are left as foreground. After removing the potential foreground pixels in set  $F_p$ , the background is almost constituted of non-inked pixels. Then a single polynomial surface is further fitted to the non-inked pixels to find more accurate background. Finally, a surface map is recomputed using the estimated polynomial as shown in Figure 2(a).

### 3.3 Image reconstruction and normalization

When the surface map is constructed, pixels will be divided into two clusters  $S_{upper}$  and  $S_{lower}$  according to their intensities by comparing values in the surface map,

$$\begin{aligned} S_{upper} &= \{(x, y) : f(x, y) \geq M_s(x, y)\} \\ S_{lower} &= \{(x, y) : f(x, y) < M_s(x, y)\} \end{aligned} \quad (7)$$

We use a piecewise linear transform function to remove the uneven shading and reconstruct the given document image. Suppose the maximum in the surface map is  $Max$ , the image  $R(x, y)$  is reconstructed by using following function. Note that  $f$  and  $M_s$  represent  $f(x, y)$  and  $M_s(x, y)$  individually for conciseness.

$$R(x, y) = \begin{cases} Max + \frac{(1-Max)(f-M_s)}{1-M_s} & f \geq M_s \\ f \frac{Max}{M_s} & f < M_s \end{cases} \quad (8)$$

As an example, for the original image in Figure 1(a), the reconstructed document image is shown in Figure 2(b). We can obviously observe that the uneven shading is compensated without too bright or too dark background and is well balanced. Since the reconstructed image contains almost flatten background, a

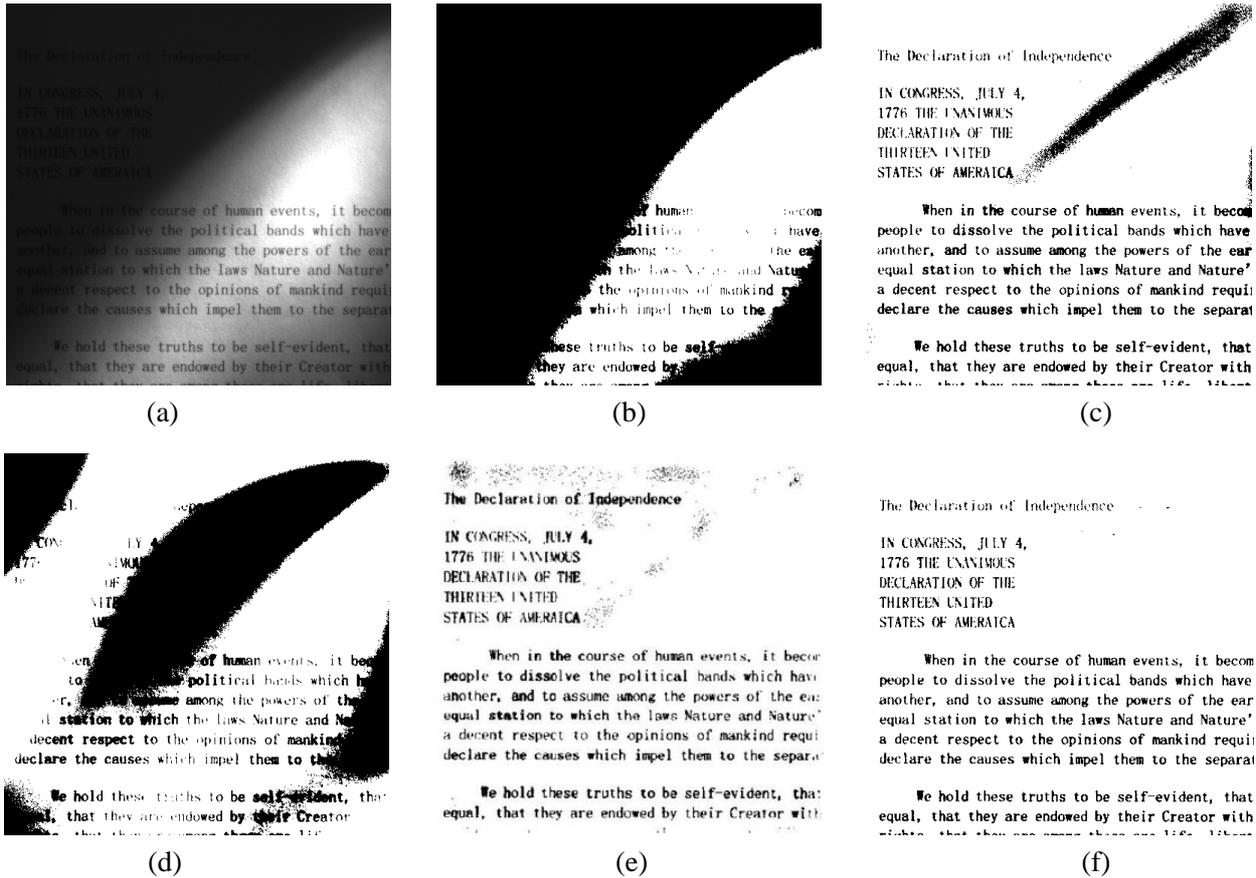


Figure 4 Experimental results: (a) an original image containing very bright and very dark shading; (b) obtained by using Otsu's method; (c) acquired by using Sauvola's method; (d) result by using Lu's method with a single polynomial; (e) result by using Wen's method with curvelet transform; (f) Our thresholding result with piecewise polynomials.

single polynomial surface fitting, that is, only one block is considered, can be applied to the estimated flat background as  $M_s^1(x, y)$ , then the document image is further reconstructed as  $Q(x, y)$  using following equation:

$$Q(x, y) = \begin{cases} -1.0 & M_s^1(x, y) \leq 0 \\ \frac{R(x, y)}{M_s^1(x, y)} & \text{Otherwise} \end{cases} \quad (9)$$

The reconstructed  $Q(x, y)$  is in the range  $[-1, 1]$  and is normalized to  $[0, 255]$ . This makes the background become bright like a "white" paper and foreground text become dark as inked. The reconstructed image is shown in Figure 2(c).

### 3.4 Background noise removing

Although most part of uneven shading has been removed, there still exist some salt-and-pepper noises in the background as shown in Figure 2(c). This may cause a negative impact on the quality of binarization. We compute narrow Gaussian blocks [15] to find

background as could as possible and fill them with the maximum value of surface map.

To remove background noises, we divide the normalized image into small non-overlapped blocks with size such as  $5 \times 5$ . If a block contains only background, the value of pixels will distribute narrowly in a small range. Meanwhile, if a block contains both inked text and non-inked background, the distribution will be in a broad range, since the final reconstructed image has high contrast, we can identify a block belonging to the document background if its histogram close to a narrow gaussian form statistically.

A histogram is computed for each block. It is easy to find the *mode* with the highest peak in the histogram. Meanwhile, the minimum and maximum intensities *min*, *max* are computed in the block. If the histogram has narrow gaussian shape, the intensities are concentrated near the *mode*, that is, in a range  $[mode - \Delta\sigma, mode + \Delta\sigma]$ . In the contrast enhanced normalized image, we empirically found that if the amount of pixels in range  $[mode - 6, mode + 6]$  is

more than 60% in a block, this block belongs to the background if the distribution range is narrow.

In summary, to make a block belongs to background, it should satisfy two conditions:

- The amount of pixels in  $[mode - 6, mode + 6]$  is over more than 60% in the block.
- The distribution range is narrow, that is,  $||mode - min| - |max - mode|| \leq 60$ .

Figure 2(d) is an image after noise removing. One can observe that there are lots of noises in the background shown in Figure 2(c), however, the background becomes cleaner after noise removing.

### 3.5 Binarization and small spur removing

We simply use Otsu's method as a global thresholding for the final binarization. However, the thresholding may produce concavities and convexities along the inked text stroke.

There are many cases of concavity and convexity. Some single-pixel concavities and convexities in vertical and horizontal directions are illustrated in Figure 5. We use "○" to represent white pixel (background) and "●" to represent black pixel on the text stroke. If a pixel is white and its neighbors are in the cases just like the top row of the figure, this central pixel should be changed into black, while a black pixel surrounded by its neighbors like the cases in the bottom row should be changed into white pixel.

Figure 5 only shows cases in vertical and horizontal directions. As options, there are also other cases in oblique directions such as northwest, northeast, southeast and southwest directions.

After the above processing, there may still exist some falsely detected isolated points or relatively large components in the result. A post-processing as an option is to delete some useless components. All components of detected foreground are labeled and their area are

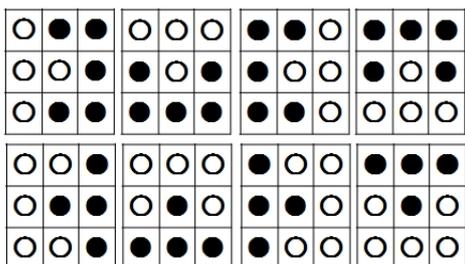


Figure 5 Cases of convex and concave on text stroke.

computed. If the area of a component is out of a designated range, it should be deleted. For image shown in Figure 1(a), we use a range  $[3, 200]$ . The final processed result after smaller and larger component deletion is illustrated in Figure 1(f).

## 4. Experimental results and discussion

To verify the effectiveness of the proposed method, we have implemented our algorithm using the standard ANSI C language on windows 7. The code without optimization runs on a desktop with CPU 2.93 GHz, 4.00GB memory.

We only designated the number of blocks and the pixels of overlaps as parameters. We have experimented on several images to demonstrate the binarization effects. One of the results is shown in Figure 1. Figure 1(a) is an  $512 \times 512$  original image with very complicated non-uniform uneven shading. The histogram of this image does not own bi-modality, the thresholding result is very bad as shown in (b) when using Otsu's method. We also implemented Sauvola's method as described in his paper[5], the result is (c). When Lu's approach is applied to the image (a), the produced result is (d). Obviously, the background cannot be estimated using a single polynomial. Figure 1(e) is a result from [13] recently published. We can know visually our result is cleaner and better. Figure 2(a) shows the constructed surface map; Figure 2(b) and (c) show the reconstructed and normalized images. Since the uneven shading is removed, the visibility is improved in a steep rise. Figure 2(d) shows normalized image with background noises removed and (e) demonstrate single polynomial and piecewise polynomial curve fittings as an comparing example.

Another result is illustrated in Figure 4. (a) is an  $512 \times 512$  original image with very bright and very dark shading. Some shading contains pseudo sharp edges because of too strong lighting. As one can expect, the Otsu's method failed as shown in (b). (c) is a result by using Sauvola's method which is sensitive to the large shading variations. Background estimation by using a single polynomial also failed (d). Since the shading changes abruptly, a single polynomial function cannot fit the background shading well. Figure 4 (e) is a result from [13] and (f) is ours by using piecewise polynomials to fit background then thresholding by using Otsu's method. Comparing with all other results, our result (f) is cleaner and better even without background noise removing.

Figure 6 demonstrates another experimental results. (a) is an original  $400 \times 584$  image from a Tibet text-

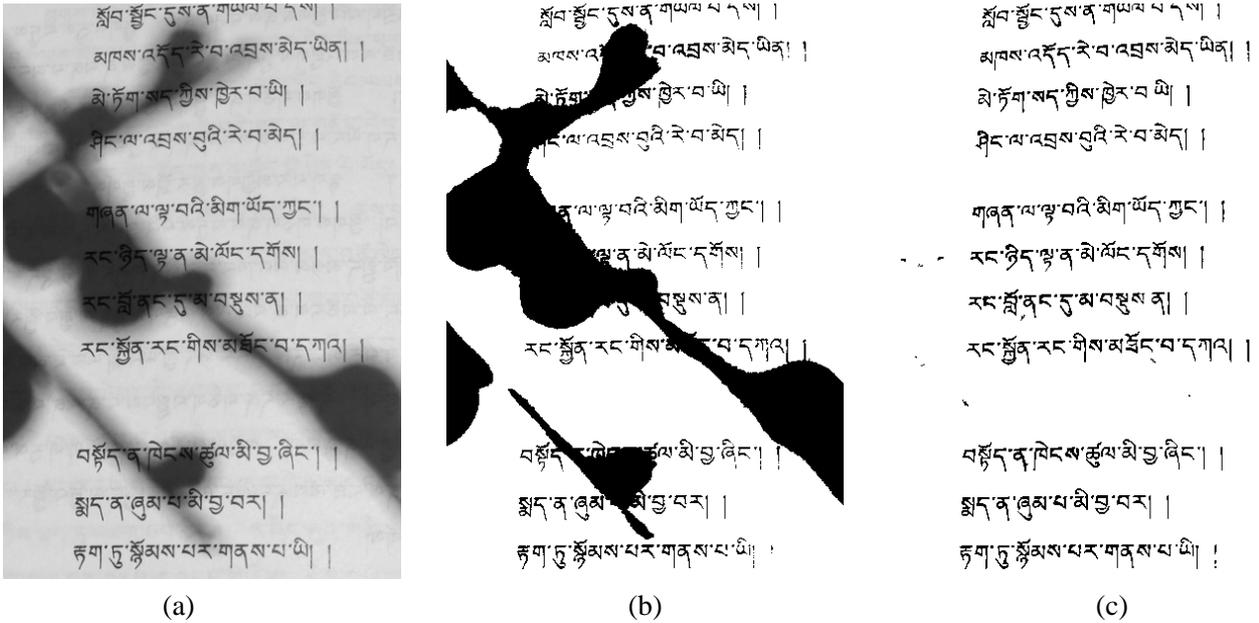


Figure 6 Experimental results: (a) an original image containing very sharp shading variation; (b) result by using Lu's method with a single polynomial; (c) Our thresholding result with piecewise polynomials.

book. The image is illuminated unevenly and has very sharp shading variations. It is needless to say that Otsu's method cannot threshold this image. Sauvola's method and Lu's method also cannot binarize it well. For example, (b) shows that a single polynomial cannot estimate this kind of sharp changing shading. Most of background is grouped into inked components. However, our piecewise polynomial approach binarized all pixels into two clusters very well, although a few background components are clustered as inked falsely as demonstrated in (c).

Figure 7 is another experimental result. (a) is an original image with size  $320 \times 240$ . The varying shading changes from dark left to bright right smoothly. It seems that a single polynomial can fit this kind of background well, however, unexpectedly, it failed as shown in (c). Since this original image used in [6], we also show the result created by using the window-based Bradley's method in (b). The left part seems dirty. (d) is our result by using piecewise polynomials background surface fitting. Compared with (c), it has a great improvement.

The running time in seconds of four binarization methods are shown in table 1. Note that the Sauvola's method has been accelerated by using integral image. According to [13], Wen's method seems taking longer time. Compared with single polynomial method, ours costs a little more time but the binarization qualities are improved greatly.

Table 1 Running time of four binarization methods.

	Otsu's	Sauvola's	Lu's	Ours
Fig.1	0.016	0.046	0.171	0.203
Fig.4	0.031	0.047	0.156	0.203
Fig.6	0.016	0.047	0.140	0.156

To further show the validity of the proposed approach, we use an OCR software of ABBYY FineReader [17] to examine binarization performance. We use Figure 4 as an example. By visual observation it is known that Figure 4(a), (b) and (d) have very low recognition rate. We mainly compare the quality of (c), (e) and (f). The original image contains 91 words. 49 words in (c), 71 words in (e) and 87 in (f) are recognized correctly with recognition rate of 53.8%, 78.0% and 95.6%.

## 5. Conclusion

In this paper, we proposed a new approach for binarizing unevenly shaded document image. The method estimates complicated background surface by employing piecewise polynomials to fit the sharply changed variation surface. The document image is reconstructed containing almost uniform shading. After normalization, the reconstructed image is improved by remov-

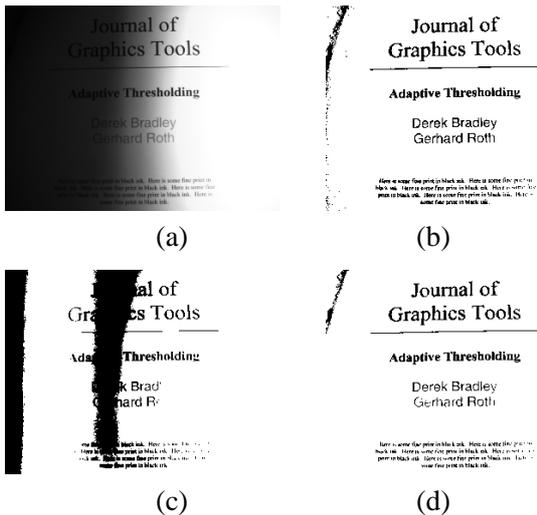


Figure 7 Experimental results: (a) an original image with complicated non-uniform shading; (b) obtained by using Bradley' method; (c) result by using a single polynomial fitting; (d) result by using piecewise polynomials fitting.

ing background noises, which is easier for binarization through global thresholding to segment the background surface and foreground text. The experimental results show that the proposed approach can binarize document image with very complicated shading. The runtime of the proposed method is a little expensive since we use piecewise polynomials, however, the binarization quality and OCR quality are improved. As shown in Figure 6(c) and 7(d), some pixels are still segmented incorrectly as inked foreground. Our next step is to improve the shading estimation quality by applying Gaussian kernel functions to estimate more unevenly shaded surface.

## Acknowledgment

The authors would like to thank JiangTao Wen from Yanshan university, who kindly permitted us to use his original images in Figure 1(a) and Figure 4(a), Heming Huang from Qinghai Normal University, who took image in Figure 6(a) for us. This work is a part of researches supported by Chunhui project of the Education Ministry of China under Grant No.Z2011092, No.Z2012100 and National Nature Science Foundation of China under Grant No.61307063.

## References

[1] N. Otsu, A Thresholding Selection Method from Gray-level Histograms, *IEEE Trans. Syst. Man Cybern.*, Vol.9, No.1, pp.62-66, 1979.

[2] J. Kittler and J. Illingworth, Minimum Error Thresholding, *Pattern Recognition*, Vol.19, No.1, pp.41-47, 1986.

[3] W. Niblack, *An Introduction to Digital Image Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1986.

[4] Wellner, P.D., Adaptive Thresholding for the Digitaldesk, *Technical Report EPC-93-110*, EuroPARC, 1993.

[5] J. Sauvola and M. Pietikainen, Adaptive document image Binarization, *Pattern Recognition*, Vol.33, No.2, pp.225-236, 2000.

[6] Bradley, G., Roth, G., Adaptive Thresholding Using Integral Image, *Journal of Graphics Tools*, Vol.12, No.2 p.13-21, 2007.

[7] F. Shafait, D. Keysers and T. M. Breuel, Efficient Implementation of Local Adaptive Thresholding Techniques Using Integral Images, *SPIE Document Recognition and Retrieval XV, DRR '08*, San Jose, CA, USA. Jan. 2008.

[8] M. Krzysztof, M. Preda and M. Axel, Dynamic Threshold Using Polynomial Surface Regression with Application to the Binarization of Fingerprints, *Proc. of the SPIE*, 5779, pp.94-104, 2005.

[9] Gatos, B., Pratikakis, I. and Perantonis, S., Adaptive degraded document image binarization, *Pattern Recognition*, Vol.39, pp.317-327, 2006.

[10] S. Lu and C. L. Tan, Thresholding of Badly Illuminated Document Images through Photometric Correction, *Proc. of the 2007 ACM symposium on Document Engineering, DocEng 2007*, pp.3-8, 2007.

[11] S. Lu and C. T. Tan, Binarization of Badly Illuminated Document Images through Shading Estimation and Compensation, *International Conference on Document Analysis and Recognition, ICDAR 2007*, pp.23-26, September 2007, Curitiba, Brazil.

[12] S. Lu, B. Su and C. L. Tan, Document image binarization using background estimation and stroke edges, *International Journal on Document Analysis and Recognition*, Vol.13, no.4, pp.303-314, December 2010.

[13] J. Wen, S. Li and J. Sun, A New Binarization Method for Non-uniform Illuminated Document Images, *Pattern Recognition*, Vol.46, pp.1670-1690, 2013.

[14] M. Sezgin and B. Sankur, Survey over Image Thresholding Techniques and Quantitative Performance Evaluation, *Journal of Electronic Imaging*, Vol.13, No.1, pp.146-168, 2004.

- [15] D. M. Oliveira and R. D. Lins, A New Method for Shading Removal and Binarization of Documents Acquired with Portable Digital Cameras, In: *Camera-Based Document Analysis and Recognition*, 2009, Barcelona (Spain).
- [16] R. I. Hartley, In Defence of the Eight-Point Algorithm, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.19, No.6, 1997.
- [17] ABBYY: <http://finereader.add-soft.jp/>