International Journal of Multidisciplinary and Current Research

Research Article

ISSN: 2321-3124
Available at: http://ijmcr.com

Document Image Binarization using GSA & TCM

Jyoti Rani and Davinder Parkash

Department of Electronics & Communication Engineering, HCTM Kaithal, Haryana, India

Accepted 14 Dec 2015, Available online 18 Dec 2015, **Vol.3 (Nov/Dec 2015 issue)**

Abstract

Document Image Binarization is performed in the preprocessing stage for document analysis and it aims to segment the foreground text from the document background. A fast and accurate document image binarization technique is important for the ensuing document image processing tasks such as optical character recognition (OCR) and Document Image Retrieval (DIR). This research area has been studied for decades; many techniques have been reported and applied on different commercial document analysis applications. However, there are still some unsolved problems need to be addressed due to the high inter/intra-variation between the text stroke and the document background across different document images. Image binarization is the method of separation of pixel values into dual collections, black as foreground and white as background. Thresholding has found to be a well-known technique used for binarization of document images. Thresholding is further divide into the global and local thresholding techniques.

Keywords: *Documents, Binarization, Gravitational Search Algorithm, Texton Co-occurance Matrix*

Nomenclature

OCR : Optical Character Recognition
DIR : Document Image Retrieval
MS : Multi Spectrum
PCA : Principle Component Analysis
LDA : Linear Discriminate Analysis
GSA : Gravitational Search Algorithm
TCM : Texton Co-occurance Matrix
DIBCO : Document Image Binarization Contest

Introduction

There is huge amount of textual information that is embedded within images. For example, more and more documents are digitalized everyday through scanner, camera and other equipment. Many digital images contain texts, and a large amount of textual information is embedded in web images. It will be very useful to convert the characters from image format to textual format by using optical character recognition (OCR). This converted text information is very important for documents images, document image retrieval and so on. However, in many cases, the document images cannot be directly fed to an OCR system due to the following reasons:

- The original document papers suffer from different kinds of degradation including smear, ink-bleeding through and intensity variation, especially for historical documents.
- The process of obtaining digital images from the real world is not perfect. There are many factors that may cause image distortion, such as incorrect focal length, camera shaking/object movement, low resolution, etc.

The web images in the internet are often susceptible to certain image degradation such as low resolution and small size, which is specially designed for faster network transmission rate, computer-generated-character artifacts, and special effects on images to attract visual attention.

Binarization

The main aim of image segmentation is to group image pixels according to constituent regions or objects. On document images this problem consists of two classes: foreground and background. The binarized image should be perceptually similar. For the binarization of documents global and adaptive binarization methods exist. The single threshold value is used for every pixel by global method, adaptive methods define local regions in which separate threshold values are calculated. Current binarization methods use gray value images as input. Color images can be converted with the standard conversion:

$$I(x, y) = 0.3R(x, y) + 0.59G(x, y) + 0.11B(x, y),$$

where R, G and B are the Red, Green and Blue channel of the color image. For a gray value image I(x, y) with intensity values between 0 and 1 and a threshold T(x, y) each image pixel is classified in foreground (labeled as 1) and background (labeled as 0) resulting in the thresholded image $I_{th}(x, y)$:

$I_{th}(x, y) = (1$ if $I(x, y) > T(x, y)$
0 if $I(x, y) \leq T(x, y)$

where T(x, y) = Tg = constant if a global threshold is applied. Adaptive methods have the characteristic that the value of T depends on the local gray value characteristics. Global thresholds are suitable for images with a bimodal gray value distribution, where adaptive methods can handle documents with e.g. non-uniform illumination. Recent developments (see DIBCO and H-DIBCO) show that binarization methods estimate the background or combine multiple binarization methods to achieve a better segmentation. The methods presented comprise Niblack, Sauvola and a color segmentation method. In the following, state of the art methods of image binarization are categorized in global and adaptive methods, methods based on background estimation and methods that use a combination of different binarization methods.

Gravitational Search Algorithm (GSA)

GSA was introduced by Rashedi *et al.* in 2009 and is intended to solve optimization problems. The population based heuristic algorithm is based on the law of gravity and mass interactions. The algorithm is comprised of collection of searcher agents that interact with each other through the gravity force. The agents are considered as objects and their performance is measured by their masses. The gravity force causes a global movement where all objects move towards other objects with heavier masses. The slow movement of heavier masses guarantees the exploitation step of the algorithm and corresponds to good solutions. The masses are actually obeying the law of gravity as shown in Equation (1) and the law of motion in Equation (2).

F = G (M1M2 / R2) (1)
a = F/M (2)

Based on Equation (1), F represents the magnitude of the gravitational force, G is gravitational constant, M1 and M2 are the mass of the first and second objects and R is the distance between the two objects. Equation (1) shows that in the Newton law of gravity, the gravitational force between two objects is directly proportional to the product of their masses and inversely proportional to the square of the distance between the objects. While for Equation (2), Newton's second law shows that when a force, F, is applied to an object, its acceleration, a, depends on the force and its mass, M.

In GSA, the agent has four parameters which are position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass represents the solution of the problem, where the gravitational and inertial masses are determined using a fitness function. The algorithm is navigated by adjusting the gravitational and inertia masses, whereas each mass presents a solution. Masses are attracted by the heaviest mass. Hence, the heaviest mass presents an optimum solution in the search space. The steps of GSA are as follows:

Step 1: Agents initialization

The positions of the N number of agents are initialized randomly.

$$X_i = x_i^1, x_i^d, \ldots\ldots x_i^n \;\; for\; i = 1,2 \ldots\ldots, N \tag{3}$$

x_i^d represents the positions of the ith agent in the dth dimension, while n is the space dimension.

Step 2: Fitness evolution and best fitness computation

For minimization or maximization problems, the fitness evolution is performed by evaluating the best and worst fitness for all agents at each iteration.

Minimization problems:

$$best(t) = \min_{j\epsilon\{1,..N\}} fit\, j(t) \tag{4}$$
$$worst(t) = \max_{j\epsilon\{1,..N\}} fit\, j(t) \tag{5}$$

Maximization problems:

$$best(t) = \max_{j\epsilon\{1,..N\}} fit\, j(t) \tag{6}$$
$$worst(t) = \min_{j\epsilon\{1,..N\}} fit\, j(t) \tag{7}$$

fit j(t) represents the fitness value of the *jth* agent at iteration *t*, *best(t)* and *worst(t)* represents the best and worst fitness at iteration *t*.

Step 3: Gravitational constant (*G*) computation

Gravitational constant *G* is computed at iteration *t*

$$G(t) = G_0 e^{-\alpha t/T} \tag{8}$$

G0 and α are initialized at the beginning and will be reduced with time to control the search accuracy. *T* is the total number of iterations.

Step 4: Masses of the agents' calculation

Gravitational and inertia masses for each agent are calculated at iteration *t*.

Mai = Mpi = Mii = Mi, i = l, 2, ... ,N. (9)

$$m_i(t) = \frac{fit(t) - worst(t)}{best(t) - worst(t)} \quad (10)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \quad (11)$$

Mai and *Mpi* are the active and passive gravitational masses respectively, while *Mii* is the inertia mass of the ith agent.

Step 5: Accelerations of agents' calculation

Acceleration of the ith agents at iteration *t* is computed.

$$a_i^d(t) = F_i^d(t)/M_{ii}(t) \quad (12)$$

$F_i^d(t)$ is the total force acting on ith agent calculated as:

$$F_i^d(t) = \sum_{j \in Kbest,\, j \neq i} rand_j F_{ij}^d(t)$$

$$F_i^d(t) = \sum_{j \in kbest} rand_j F_{ij}^d(t) \quad (13)$$

Kbest is the set of first *K* agents with the best fitness value and biggest mass. *Kbest* will decrease linearly with time and at the end there will be only one agent applying force to the others.

$F_{ij}^d(t)$ can be computed as

$$F_{ij}^d(t) = G(t).\left(M_{pi}(t) \times \frac{M_{ai}(t)}{R_{ij}(t)} + \varepsilon\right).(x_j^d(t) - x_i^d(t))$$

$F_{ij}^d(t)$ is the force acting on agent *i* from agent *j* at *dth* dimension and *tth* iteration. *Rij(t)* is the Euclidian distance between two agents *i* and *j* at iteration *t. G(t)* is the computed gravitational constant at the same iteration while ε is a small constant.

Step 6: Velocity and positions of agents

Velocity and the position of the agents at next iteration (t+1) are computed based on the following equations:

$$v_i^d(t+1) = rand_i x v_i^d(t) + a_i^d(t)$$

$$x_i^d(t+1) = v_i^d(t+1) + x_i^d(t)$$

Step 7: Repeat steps 2 to 6

Steps 2 to 6 are repeated until the iterations reach their maximum limit. The best fitness value at the final iteration is computed as the global fitness while the position of the corresponding agent at specified dimensions is computed as the global solution of that particular problem.

Proposed Work

In this section proposed document binarization technique is explained. For a degraded image adaptive contrast map is constructed and then threshold is calculated from that which converts the document into binary. Here is the explanation.

The human visual system is more sensitive to contrast than absolute luminance, we can perceive the world similarly regardless of the huge changes in illumination over the day or from place to place. Contrast is the difference in luminance or color that makes an object (or its representation in an image or display) distinguishable. In visual perception of the real world, contrast is determined by the difference in the color and brightness of the object and other objects within the same field of view. The contrast of image can be categorized as: global contrast and local contrast. Global contrast measures the brightness difference between the darkest and brightest element in the entire image. Tools like Curves and Levels only change global contrast as they treat all pixels with the same brightness levels identical.

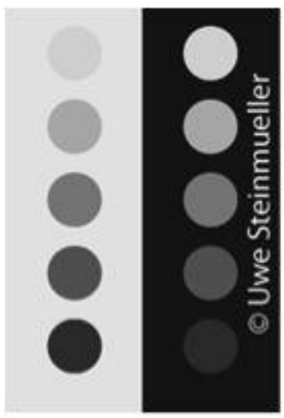
The global contrast has three main regions

- Mid-tones
- Highlights
- Shadows

The sum of the contrast amounts of these three regions defines the global contrast. This means if you spend more global contrast on the mid-tones (very commonly needed) you can spend less global contrast on highlights and shadows at any given global contrast level.

The mid-tones normally show the main subject. If the mid-tones show low contrast the image lacks snap. Adding more global contrast to the mid-tones (snap) often results in compressed shadows and highlights. Adding some local contrast can help to improve the overall image presentation.

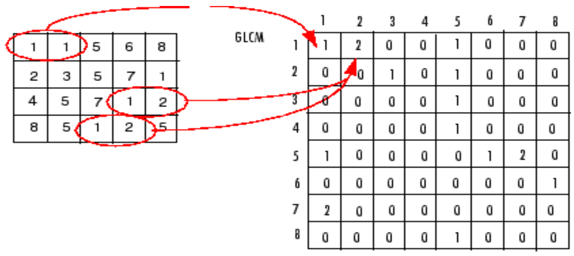
The local contrast is based on the retinas theory according to which our eyes sees the difference in respect to surroundings, a color map below can prove this point.



Contrast map

The circles in each row have exactly the identical brightness levels. Yet the top right circle looks a lot brighter than the one on the left. This is because our eyes see the difference to the local surrounding. The right circle looks much brighter with the dark gray background compared to a brighter background on the left. Just the opposite is true for the two circles on the bottom. For our eyes the absolute brightness is of less importance than the relative relation to other close areas. So, local contrast is very important for processing or enhancement of any image.

In our work because of this human visual system local contrast map is extracted from an image and then on the basis of that a local thresholding approach will be used to convert the image into binary format. Previously image gradient and normalize image gradient were used to extract local contrast of image, these methods are quite good, although the variation of bright to weak contrast can be compensated by these methods yet these don't perform well in case of document which have bright text. This is because a weak contrast will be calculated for stroke edges of the bright text. Calculation of local contrast and then local thresholding algorithm like Otsu is used and then local image edge detection algorithm is used in paper published by Sayali Shukla, Ashwini Sonawane, Pooja Tiwari & Vrushali Topale (2014). We have followed the same line of action we also use local thresholding, but it removes the need of using again local edge detection algorithm like canny edge detection. Gray level co-occurrence matrix (GLCM) also called texton co-occurrence matrix (TCM) fulfills our purpose. It is a local contrast mapping method. Here basically TCM serves two purposes: make image's local contrast map, unaffected by the illumination variation of image and local edge detection. Further, the GLCM functions characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix. A gray-level co-occurrence matrix (GLCM) is generated by calculating how often a pixel with the intensity (gray-level) value i occurs in a specific spatial relationship to a pixel with the value *j*. By default, the spatial relationship is defined as the pixel of interest and the pixel to its immediate right (horizontally adjacent), but you can specify other spatial relationships between the two pixels. Each element (i,j) in the resultant GLCM is simply the sum of the number of times that the pixel with value i occurred in the specified spatial relationship to a pixel with value *j* in the input image. The number of gray levels in the image determines the size of the GLCM. GLCM of an image is computed using a displacement vector d, defined by its radius δ and orientation θ. To illustrate, the following figure shows how gray co-matrix calculates the first three values in a GLCM. In the output GLCM, element (1,1) contains the value 1 because there is only one instance in the input image where two horizontally adjacent pixels have the values 1 and 1, respectively. GLCM(1,2) contains the value 2 because there are two instances where two horizontally adjacent pixels have the values 1 and 2. Element (1,3) in the GLCM has the value 0 because there are no instances of two horizontally adjacent pixels with the values 1 and 3. Gray co matrix continues processing the input image, scanning the image for other pixel pairs (*i,j*) and recording the sums in the corresponding elements of the GLCM. Figure shows this concept.



GLCM output of a test matrix

Choice of Angle θ

In actual every pixel has eight neighboring pixels allowing eight choices for θ, which are 0°, 45°, 90°, 135°, 180°, 225°, 270° or 315°. However, taking into consideration the definition of GLCM, the co-occurring pairs obtained by choosing θ equal to 0° would be similar to those obtained by choosing θ equal to 180°. This concept extends to 45°, 90° and 135° as well. Hence, one has four choices to select the value of θ.

Choice of Radius δ

Above we have mentioned a term radius about GLCM. In the last example matching pairs have been taken upto one distance, this constitutes the radius of GLCM. Various research studies show δ values ranging from 1, 2 to 10. Applying large displacement value to a fine texture would yield a GLCM that does not capture detailed textural information. From the previous studies, it has been concluded that overall classification accuracies with δ = 1, 2, 4, 8 are acceptable with the best results for δ = 1 and 2. This conclusion is justified, as a pixel is more likely to be correlated to other closely located pixel than the one located far away. Also, displacement value equal to the size of the texture element improves classification.

The dimension of a GLCM is determined by the maximum gray value of the pixel. Number of gray levels is an important factor in GLCM computation. More levels would mean more accurate extracted textural information, with increased computational costs. The computational complexity of GLCM method is highly sensitive to the number of gray levels. As in above example in figure 4.1, the size of GLCM is 8 by 8 matrix as 8 gray levels have been considered. Thus for a predetermined value of G, a GLCM is required for each unique pair of δ and θ. GLCM is a second-order texture measure. The GLCM's lower left triangular matrix is always a reflection of the upper right triangular matrix and the diagonal always contains even numbers. Another test matrix and its various GLCM parameters are related to specific first-order statistical concepts. For instance, contrast would mean pixel pair repetition rate, variance would mean spatial frequency detection etc. Association of a textural meaning to each of these parameters is very critical. Traditionally, GLCM is dimensioned to the number of gray levels G and stores the co-occurrence probabilities g_{ij}. To determine the texture features, selected statistics are applied to each GLCM by iterating through the entire matrix. The textural features are based on statistics which summarize the relative frequency distribution which describes how often one gray tone will appear in a specified spatial relationship to another gray tone on the image.

Following notations are used to explain the various textural features:

g_{ij} = $(i, j)^{th}$ entry in GLCM
$g_x(i)$ = i^{th} entry in marginal probability matrix obtained by summing rows of

$$gij = \sum_{j=1}^{Ng} g(i,j)$$

N_g = Number of distinct gray levels in the image
Σi = sum of i's values from 1 to Ng
Σj = sum of j's values from 1 to Ng
$g_y(i) = \sum_{i=1}^{Ng} g(i,j)$
$g_{x+y}(k) = \sum_{i=1}^{Ng}.\sum_{j=1}^{Ng} g(i,j)$ where i+j = k = 2, 3, …, 2 N_g
gx-y(k) $=\sum_{i=1}^{Ng}.\sum_{j=1}^{Ng} g(i,j)$ where |i-j| = k = 0, 1, …, Ng-1
Contrast (con) = ΣΣ(i-j)^2 gij

This statistic measures the spatial frequency of an image and is difference moment of GLCM. It is the difference between the highest and the lowest values of a contiguous set of pixels. It measures the amount of local variations present in the image. A low contrast image presents GLCM concentration term around the principal diagonal and features low spatial frequencies.

From this GLCM process local contrast of image is obtained. From this we have developed the equation to calculate the threshold value. Since it will be a local threshold value so, the size of threshold matrix will be same as test image. For this formula we were inspired by Abdenour Sehad's work in 2013. We have done changes in that and final thresholding formula includes local mean of image and a gain factor which will act as a bias factor. This factor has the range (0-1) always. Its value will be determined experimentally. This formula is used in a window size of image and later on combined. Mathematical expression is shown below.

$$Threshold(i,j) = k\left(I_{mean}(i,j) + \sqrt{contrast}\right)$$

Images in the database are of different kinds and different contrast with noise too. If same bias constant (k) and window size is used for all types of images then results are erroneous. For every image we have to test the 'k' for different block sizes. To avoid this work load gravitational search algorithm (GSA) is used, which will tune the gain value along with block size for every image it's a kind of unsupervised learning which will automatically set the optimal value of gain and window size to fit for every image. The pseudo code for proposed algorithm is below:

```
Begin
{
Select the input historical document image
Apply wiener filter
Initialize the GSA parameters and values of 'k' and block size
Call objective function (label B starts)
{
Calculate gray co occurrence matrix and take out contrast of local image
Calculate local mean
Use threshold formula mentioned above
Compare the intensity pixels with threshold value
    {
    If  Image(i,j)>threshold(i,j)
        Image (i,j)=1
Else
Image (i,j)=0
        }
Calculate fitness value (label B ends)
Update the value of gain and block size.
Repeat the label B
Compare the fitness value  fitness(t+1) > fitness(t)
    {
    If yes
        Terminate
    End
    }
Finalize the gain and block size value and again convert the image into binary using them
}
End
```

Since the above method discussed calculates the local threshold, the operations are done in image blocks like if outcome of the algorithm is block size =55, then image matrix will be segmented into 55 by 55 blocks and thresholded. The fitness function value thus calculated will be the inverse of sum of peak signal to noise ratio (PSNR) and F-measure. Mathematically it can be shown as:

$$Objf = \frac{1}{\mathrm{F-measure + PSNR + 0.001}}$$

Here 0.001 is used to avoid the fall of fitness function into infinity.

PSNR and F-measure are Parameters used for result analysis. This is reflected in the F-measure that is used to score boundary detectors, defined as the geometric mean of the classifier's precision, p and recall, r:

$$\text{F}-\text{measure} = \frac{2 \times \text{Precision} \times \text{recall}}{\text{Precision} + \text{recall}}$$

Where recall $= \frac{\text{TP}}{\text{TP+FN}}$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$

Where Tp = True Positive
Fp = False Positive
Fn = False Negative

$$PSNR = 10\log(\frac{C^2}{MSE})$$

Here PSNR is the peak signal to noise ratio. MSE should be minimum so that the PSNR should be high. Due to the high value of the PSNR the objective function value will be decreased which is our main objective.

Results & Discussion

Image binarization technique requires the calculation of adaptive threshold value which can be tuned to image type and its contrast. No a single method can work for all type of images. For that purpose we used self-tuning optimization which will extract the threshold value depending upon image as described in previous chapter. In our work we have used data sets of images. This data set consists of a set of 8 degraded document images of various contrast level. The extraction of binary image from a document image with less contrast is very easy and gives high values of out parameters. The example images are shown in table 5.2 below. By optimization gain value and block size for every image is set, which have been tabulated in table 5.3 below. Before processing the image pre processing of image is done by using wiener filter. The Wiener filter minimizes the mean square error between the estimated random process and the desired process. The goal of the Wiener filter is to compute a statistical estimate of an unknown signal using a related signal as an input and filtering that known signal to produce the estimate as an output. For example, the known signal might consist of an unknown signal of interest that has been corrupted by additive noise. The Wiener filter can be used to filter out the noise from the corrupted signal to provide an estimate of the underlying signal of interest. Thus wiener filter removes the noise in the image and make it more approachable to desired results. The output of wiener filter and test image is shown in figure 5.1 and 5.2 and notice the difference. Further gravitational search algorithm (GSA) is used and run for 30 iterations. In each iteration a new value of gain factor and window size is calculated and PSNR and F-measure output is observed at each iteration.

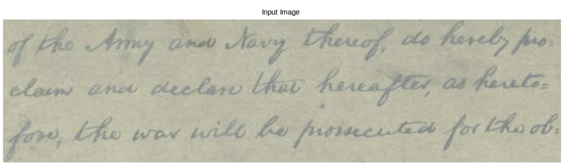


Figure: Input test image

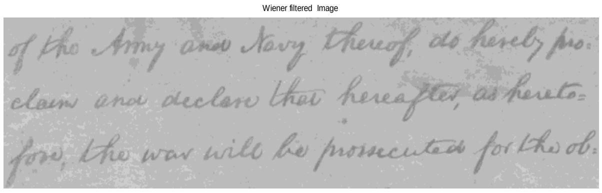


Figure: Wiener filter processed image

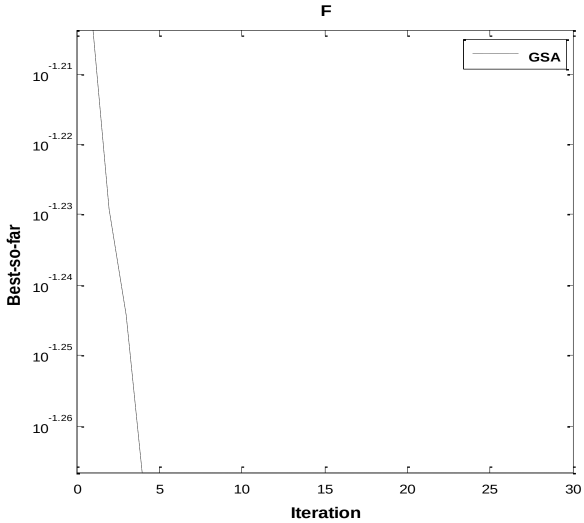


Figure: Cost function value after optimization of test image

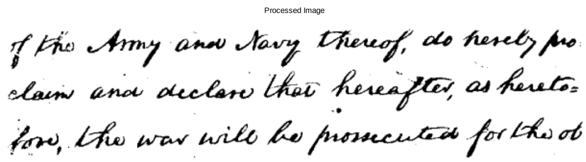


Figure: Final binarised image of input in figure 5.1

We have compared our results with Otsu's method (OTSU), Sauvola's method (SAUV), Niblack's method (NIBL), Bernsen's method (BERN), Gatos et al.'s method (GATO), and our previous methods (LMM, BE). The datasets are composed of the same series of document images that suffer from several common document degradations such as smear, smudge, bleed-through and low contrast. Results are shown in table. Our method achieves the highest score in PSNR and F-measure.

Methods	F-Measure (%)	PSNR
OTSU	85.27	17.51
SAUV	75.3	15.96
NIBL	74.1	15.73
BERN	41.3	8.57
GATO	71.99	15.1
LMM	85.49	17.83
BE	86.41	18.14
Proposed Method	91.16	20.64

Conclusion

In this paper, a thresholding technique has been proposed based on the gray level co-occurrence matrix. The technique extracts the edge information and the gray level transition frequency from the GLCM to compute the threshold value. The algorithm is also designed to have the flexibility over the edge definition. Thus, it can handle image with fuzzy boundaries between the image's object and background. The proposed technique was tested with star fruit defect image and result good segmentation in order to identify the area of the defect on the star fruit skin. The results were compared with three other techniques. It showed that segmentation using the proposed method gives the best result compared to the other method.

References

[1]. Sayali Shukla, Ashwini Sonawane, Vrushali Topale, Pooja Tiwari(May,2014) Improving Degraded Document Images Using Binarization Technique, International Journal Of Scientific & Technology Research Volume 3, Issue 5.

[2]. Jagroop Kaur, Dr.Rajiv Mahajan (May 2014) A Review of Degraded Document Image Binarization Techniques, International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 5.

[3]. Aroop Mukherjee, Soumen Kanrar (November 2010) Enhancement of Image Resolution by Binarization IJCA, Volume 10.

[4]. P.Subashini, N.Sridevi(September 2011) An Optimal Binarization Algorithm Based on Particle Swarm Optimization International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-4.

[5]. Yudong ZHANG , Lenan WU, (2011). Fast Document Image Binarization Based on an Improved Adaptive Otsu's Method and Destination Word Accumulation Journal of Computational Information Systems.

[6]. Chitrakala Gopalan, D.Manjula (Nov. 02, 2010) Sliding window approach based Text Binarisation from Complex Textual images (IJCSE) International Journal on Computer Science and Engineering Vol. 02, 309-313.

[7]. Bolan Su, Shijian Lu, and Chew Lim Tan, Senior Member, IEEE(April 2013) Robust Document Image Binarization Technique for Degraded Document Images IEEE Transactions On Image Processing, Vol. 22, No. 4.

[8]. Qiang Chena, Quan-sen Suna, Pheng Ann Hengb, De-shen Xia(2007) A double-threshold image binarization method based on edge detector Pattern Recognition Society. Published by Elsevier Science Ltd.

[9]. Tao Chen, Mikio Takagi Dept. of electronics engg. Instiute of Industrial Science 'Image Binarization by Back Propagation Algorithm.

[10]. Bolan Su, Shijian Lu, and Chew Lim Tan (April 2013) Document Image Binarization using Background Estimation and Stroke Edges IEEE Transactions On Image Processing, Vol. 22, No. 4.

[11]. Abdenour Sehad, Youcef Chibani, Mohamed Cheriet and Yacine Yaddaden (April 2014) Co-occurrence matrix for ancient degraded document image binarization IEEE Transactions on Image Processing, Vol. 23, No. 4.

[12]. M. M. Mokji, S.A.R. Abu Bakar (October 2007) Adaptive Thresholding Based On Co-Occurrence Matrix Edge Information Journal Of Computers, Vol. 2, No. 8.

[13]. O. Imocha Singh, Tejmani Sinam(August 2012) Local Contrast and Mean based Thresholding Technique in Image Binarization International Journal of Computer Applications, Volume 51– No.6.

[14]. J. Sauvola and M. Pietikainen (2000) Adaptive document image binarization, Pattern Recognit., vol. 33, no. 2, pp.225 -236.

[15]. Om Prakash Vermaa, Rishabh Sharma, Deepak Kumar([ICCCS-2012]) Binarization Based Image Edge Detection Using Bacterial Foraging Algorithm 2nd International Conference on Communication, Computing & Security.

[16]. K. Ntirogiannis, B. Gatos, I. Pratikakis (2012) combined approach for the Binarization of handwritten document images Pattern Recognition Letters, Elsevier.

[17]. Ioannis Pratikakis , Basilis Gatos and Konstantinos Ntirogiannis(2013) ICDAR 2013 Document Image Binarization Contest (DIBCO 2013), 12th International Conference on Document Analysis and Recognition.

[18]. T.Romen Singh , Sudipta Roy, O.Imocha Singh, Tejmani Sinam,
(November 2011)A New Local Adaptive Thresholding Technique in Binarization IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 6, No 2.

[19]. Bolan Su, Shijian Lu(April 2013) Binarization of Historical Document Images Using the Local Maximum and Minimum. IEEE Transactions On Image Processing, Vol. 22, No. 4.

[20]. Prashali Chaudhary, B.S. Saini(Jun-2014) An Effective And Robust Technique For The Binarization Of Degraded Document Images IJRET, Volume: 03 Issue: 06 .

[21]. Prof. S. P. Godse, Samadhan Nimbhore, Sujit Shitole, Dinesh Katke, Pradeep Kasar (May 2014) Recovery of badly degraded Document images using Binarization Technique International Journal of Scientific and Research Publications, Volume 4, Issue 5.

[22]. B. Gatos, I. Pratikakis, and I.J. Perantonis (2006) Adaptive degraded document image binarization Pattern Recognition, 39(3):317–327.

[23]. I. Blayvas, A. Bruckstein, and R. Kimmel (2006). Efficient computation of adaptive threshold surfaces for image binarization. Pattern Recognition, 39(1):89–101.

[24]. C.Wolf, J.M. Jolion, and F. Chassaing (2002) Text Localization, Enhancement and Binarization in Multimedia Documents International Conference on Pattern Recognition, (ICPR), 2:1037–1040.

[25]. B. Gatos, K. Ntirogiannis, and I. Pratikakis (Jul. 2009). ICDAR 2009 Document Image Binarization Contest (DIBCO 2009). In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), pages 1375 – 1382.

[26]. Esmat Rashedi, Hossein Nezamabadi-pour, Saeid Saryazdi(2009). A Gravitational Search Algorithm. In Proceedings of the Information Science 179 (2009) 2232-2248. Published by Elsevier Science Ltd.

[27]. Document Image Binarization with automatic Parameter Tuning by Nicholas R. Howe.