This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

GiB: a *G*ame theory *I*nspired *B*inarization technique for degraded document images

Showmik Bhowmik, Ram Sarkar *Senior Member, IEEE*, Bishwadeep Das and David Doermann, *Fellow, IEEE*

***Abstract*—Document image binarization classifies each pixel in an input document image as either foreground or background under the assumption that the document is pseudo binary in nature. However, noise introduced during acquisition or due to aging or handling of the document can make binarization a challenging task. This paper presents a novel game theory inspired binarization technique for degraded document images. A two-player, non-zero-sum, non-cooperative game is designed at the pixel level to extract the local information, which is then fed to a K-means algorithm to classify a pixel as foreground or background. We also present a preprocessing step that is performed to eliminate the intensity variation that often appears in the background and a post-processing step to refine the results. The method is tested on seven publicly available datasets, namely, DIBCO 2009-14 and 2016. The experimental results show that GiB (Game theory Inspired Binarization) outperforms competing state-of-the-art methods in most cases.**

***Index Terms*—Binarization, Game theory, Two-player game, Document image, DIBCO, K-means.**

I. Introduction

Document image binarization is the process of converting gray-level or color document images into a binary representation, where each pixel is labeled as either foreground or background. Binarization forms an integral component of many document image processing systems (DIPSs) [1], [2] ultimately affecting the performance of high-level processing tasks, such as optical character recognition (OCR) [3], word recognition [4], [5], and document layout analysis (DLA) [6].

Notably, binarization can be a very challenging task depending on the quality of the image to be processed. Common quality issues include uneven illumination, faded ink, clutter and artifacts, such as dark patches, bleed through, and creases. Uneven illumination is where the image suffers from inconsistent lighting or shadow effects during acquisition. Faded ink makes it difficult to distinguish light text from the background. Dark patches appear as stains with arbitrary shapes and are often present in areas containing characters. These patches can be difficult to remove because of their varying sizes and intensities. Bleed through occurs when content from the back of a page leaks through or becomes visible during acquisition. Bleed through can result in pixels that are very similar in intensity to the text of the document, thereby making it difficult to distinguish them on that basis alone. A few samples of degraded document images taken from the datasets are shown in Fig 1.

Robustness to all these degradations is desired when designing a binarization algorithm. However, according to research published in 2014 [7], none of the proposed methods available for document image binarization can handle all types of degradation.

In this paper, a game theory inspired technique is proposed for the binarization of degraded document images to overcome the somewhat arbitrary nature of most algorithms. To the best of our knowledge this is the first time the concept of game theory has been employed to develop a binarization technique. In doing so, a *two-player non-zero-sum* and *non-cooperative* game is designed and modeled at the pixel level to extract local information. Based on the extracted information, the pixels of the input image are classified as foreground or background using the K-means clustering algorithm. Before binarization, background estimation and image normalization are performed using the *inpainting* method [8]. For the evaluation of *GiB*, seven standard document image binarization datasets are used: H-DIBCO 2016 [9], H-DIBCO 2014 [10], DIBCO 2013 [11], H-DIBCO 2012 [12], DIBCO 2011 [13], H-DIBCO 2010 [14] and DIBCO 2009 [15]. To compare the performance of *GiB*, methods that were submitted to the DIBCO contests are considered here, along with some other state-of-the-art methods. The results show that *GiB* outperforms most of these methods.

The remainder of this paper is organized as follows. In Section II, we describe related work. The required definitions and notations are given in Section III. Our approach is presented in Section IV. Section V elaborates the experimental results; finally, conclusions are given in Section VI.

II. Related Work

Many methods for document image binarization have been introduced in the literature, and these methods can be broadly categorized into three major groups: *threshold-based methods*, *optimization-based methods* and *classification-based methods*.

A. Threshold-based methods

The methods that belong to this category are the classic solutions to the binarization problem and were the first to appear in the literature. These methods generally compute a

Showmik Bhowmik is a PhD Scholar in the Department of Computer Science and Engineering, Jadavpur University, India (e-mail: showmik.cse@gmail.com).

Ram Sarkar is an Associate Professor in the Department of Computer Science and Engineering, Jadavpur University, India (e-mail: rsarkar@ieee.org).

Bishwadeep Das is a Master's student in the Faculty of Electrical Engineering, Mathematics and Computer Science, TU Delft, Netherlands (e-mail: bishwadeepdas@tudelft.nl)

David Doermann is a Professor in the Department of Computer Science and Engineering at the University at Buffalo (e-mail: doermann@buffalo.edu).

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

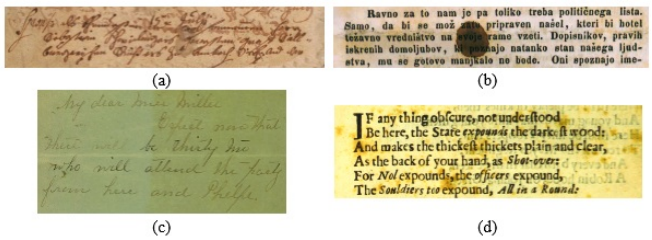

Fig. 1. Sample document images taken from (a) H-DIBCO 2016 [9] having bleed through with dark patches, (b) DIBCO 2013 [11] having dark patches with mild illuminations, (c) H-DIBCO 2012 [12] having text with faded ink, and (d) DIBCO 2011 [13] having multiple spots with bleed through

threshold to classify a pixel as foreground or background. Depending on the approach, we can further divide these methods into i) global methods [16], [17] and ii) local methods [18], [19], [20]. Global methods compute a single threshold for the entire image, typically based on global statistics. By contrast, local methods estimate the statistics of neighboring pixels to determine the threshold in a local area. Global methods are relatively fast and are useful when a stable intensity level difference is present between foreground and background pixels. However, they are less effective for documents with uneven illumination, patches and other degradations. In such situations, local methods have been shown to outperform global methods [21]. In [22], local methods are shown to work well compared to global methods in slowly changing backgrounds. Similarly, in [23], the authors consider 40 binarization methods, including both global and local methods, to evaluate their performance for noisy document images. The authors show that local methods perform better than global methods in such scenarios. However, a major issue with these local methods is that they require a set of parameters to be tuned, and the performance is often sensitive to the parameter values [24]. Hence, for documents with complex backgrounds, a set of generalized parameter values that can perform equally well for different kinds of images is difficult to find [21]. In some recent literature [1], [25], researchers have added a denoising or contrast-enhancement step before applying a local or adaptive technique. This additional preprocessing step helps in handling the uncertainty present in the adaptive thresholding methods with a generalized set of parameter values for extremely noisy document images.

B. Optimization-based methods

Researchers have recently introduced another class of methods in which binarization is formulated as an optimization problem. In most of these methods, a Markov random field (MRF) is applied to model the solution. One such method is proposed in [26], where the authors use an MRF to binarize camera-captured document images. In this work, a threshold-based method is employed to generate a binary image, which is then refined using a graph-cut algorithm. Similarly, in [27] the researchers segment the input image into text, near text and background zones and then apply a graph-cut algorithm to generate the final binarization result. In [28], simulated annealing (SA) is applied to minimize the cost function. Most of these methods use stroke width information to binarize document images and achieve good results for printed documents. However, for handwritten documents, these methods may not be as accurate in the estimation of the stroke width. Recently, in [29], researchers have proposed a specialized binarization method for handwritten document images, where a Laplacian of the image gray values is used in the energy term. In a later work [30], the authors automate the parameter selection. These optimization-based methods are relatively time consuming compared to the earlier category of methods.

C. Classification-based methods

Methods belonging to this category are comparably new in the domain of document image binarization. Both supervised and unsupervised machine learning methods have been applied to binarize document images. One such method is proposed in [31], where the authors have developed a neuro-fuzzy technique for binarization and gray value (or color) reduction of poorly illuminated documents. Some studies [22], [32] suggest that developing a common binarization technique for handling various types of noise would be difficult. Thus, the authors in [33], [34] combine several binarization techniques using a Kohonen self-organizing map (KSOM). The combined method outperforms other binarization methods in most cases. Recently, researchers have used a mixture of Gaussian distributions [35] to cluster the pixels of a document image based on local information.

In addition to these classification-based methods, a few deep learning based solutions have been introduced for degraded images. One such method is proposed in [36], where the authors use a hierarchical deep supervised network to predict foreground or text pixels. In [37], the researchers introduce a deep neural network architecture to perform binarization, where a fully convolutional neural network (FCNN) is combined with an unrolled prime dual network. The authors in [38] also use an FCNN for this purpose. In this work, a color image is taken as the input, and for each pixel, the probability that the pixel is a part of the foreground is calculated. Although deep neural networks have been successfully applied to various problems, the recent methods in binarization do not consistently outperform the state-of-the-art methods on public datasets [36]. Furthermore all these methods have large sets of parameters that need to be optimized during training, which requires a long training time.

Our binarization method is also a classification-based method that uses unsupervised learning for the final classification of pixels as foreground or background.

III. Definitions and Notations

In our work, the concepts of a *two-player non-zero-sum non-cooperative game* and the *Nash equilibrium* are employed to design a binarization technique called *GiB (Game theory Inspired Binarization)*. In this section, some definitions and notations of the newly designed game, called *Neighbors Collision (NC)*, are presented.

Game theory provides a mathematical framework for analyzing the decision-making processes and strategies of adversaries (or players) in different competitive situations [39].

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

These situations are generally conceptualized by designing different types of games and are dependent on the number of players, the symmetry of the game and co-operation among players [39].

In game theory, a game can formally be defined as follows:

Definition 1: *An n-player game can be represented by a two-tuple list* $G = \{S, U\}$ *where,* $S = \{S_1, S_2, ...S_n\}$ *is the finite non-empty set of strategies. Here, each* S_i *represents the set of available strategies for the* i^{th} *player.* $U = \{u_1, u_2, ...u_n\}$ *is the finite non-empty set of utility functions. Here, each* u_i *represents the utility function for the* i^{th} *player such that* $u_i : (S_1 \times S_2 \times .... \times S_n) \rightarrow R$.

A game is said to be *non-cooperative* if the strategy taken by a player is entirely dependent upon self-interest (i.e., without any knowledge of the others' decision/indulgence). A game is said to be *non-zero-sum* when the aggregated gain or loss of the participants can be less than or greater than zero, i.e., there exists a strategy profile $(s_1, s_2, ...s_n)$ such that $s_i \epsilon S_i$, for which the value of $\sum_{i=1}^{n} = u_i\,(s_1, ..., s_n)$ is greater than or less than zero.

One of the most important issues in game theory is to identify the *Nash equilibrium*, which is essentially a solution model for non-cooperative games. The *Nash equilibrium* refers to a stable strategy profile in which no player can gain a reward by changing his/her strategy while the strategies of the other players remain unchanged, i.e., the strategy profile contains all the relatively best strategies applied by the players. The best strategy of a player can formally be defined as follows:

Definition 2: *If* $S^{'}$ *is the set of strategies played by other players such that* $S^{'} = \{s_1, .., s_{i-1}, s_{i+1}, ..., s_n\}$, *then for any player i, the best responding strategy* s_i^B *against* $S^{'}$ *can be defined as* $u_i\left(s_i^B, S^{'}\right) \geq u_i\left(s_k, S^{'}\right) \forall_{k,k\neq i} s_k \epsilon S_i$.

Now, the *Nash equilibrium* state of a game can formally be defined as follows:

Definition 3: *A strategy profile* $\left(s_1^N, ..., s_n^N\right)$ *is a Nash equilibrium iff* s_i^N *is the best responding strategy against* $S^{N'}$, *where* $S^{N'} = \left\{s_1^N, .., s_{i-1}^N, s_{i+1}^N, ..s_n^N\right\}$ *for each player i, i.e., for every* i^{th} *player* $u_i\left(s_i^N, S^{N'}\right) \geq u_i\left(s_k, S^{N'}\right) \forall_{k,k\neq i} s_k \epsilon S_i$.

A two-player game is the simplest type of game in game theory. Two-player games are generally described using a matrix called a *payoff matrix*, which illustrates the strategies adopted by the players and their corresponding outcomes. This matrix-based description of a game facilitates the computation of the *Nash equilibrium*.

Notably, although *NC* is a *two-player, non-zero sum, non-cooperative game*, here we have customized the concept of the conventional two-player games available in game theory to make it more applicable for our purpose. Specifically, the game is played under three possible circumstances, which result in three different payoff matrices. Each payoff matrix corresponds to an independent *non-cooperative game*. Thus, *NC* is a 3-in-1 *non-cooperative game*. As we have formulated three possible games, we obtain three possible *Nash equilibria*, one corresponding to each game.

Definition 4 (Neighbor's Collision) : *Suppose Player-1 has an argument with a neighbor (here, Player-2) over the issue 'who is stronger?'. As a result, the players fight with each other. Each player can use one of two strategies, 'Fight (F)' or 'Withdraw (W)', i.e.,* $\forall i S_i = \{F, W\}$.

The following situations can occur.

Situation-1: *Player-1 and Player-2 have different levels of strength. In this situation, two cases can occur.*

Case 1: *Player-1 is stronger than Player-2*

		Player-2	
		F	**W**
Player-1	**F**	$(+\boldsymbol{m/2}, -\boldsymbol{m/2})$	$(+\boldsymbol{m/2}, -\boldsymbol{m/4})$
	W	$(-\boldsymbol{m/4}, +\boldsymbol{m/2})$	$(-\boldsymbol{m/4}, -\boldsymbol{m/4})$

Case 2: *Player-2 is stronger than Player-1*

		Player-2	
		F	**W**
Player-1	**F**	$(-\boldsymbol{m/2}, +\boldsymbol{m/2})$	$(+\boldsymbol{m/2}, -\boldsymbol{m/4})$
	W	$(-\boldsymbol{m/4}, +\boldsymbol{m/2})$	$(-\boldsymbol{m/4}, -\boldsymbol{m/4})$

Situation-2: *Both players are equally strong.*

		Player-2	
		F	**W**
Player-1	**F**	$(+\boldsymbol{m/2}, +\boldsymbol{m/2})$	$(+\boldsymbol{m/2}, -\boldsymbol{m/4})$
	W	$(-\boldsymbol{m/4}, +\boldsymbol{m/2})$	$(-\boldsymbol{m/4}, -\boldsymbol{m/4})$

Each cell of the payoff matrices contains a payoff profile, and each payoff profile has two elements: first, the payoff value for Player-1; second, the payoff value for Player-2. Here, as payoff, we have assigned $(+\frac{m}{2})$ to the winner and $(-\frac{m}{2})$ to the runner-up. If any player decides to withdraw his/her participation, then s/he has to pay the penalty; thus, we have assigned the value $(-\frac{m}{4})$ as payoff to the player who withdraws. For example, in the payoff matrix of Case-1, where Player-1 is stronger than Player-2, if both decide to fight, then Player-1 is likely to win the game; thus, $(+\frac{m}{2})$ is assigned to Player-1 and $(-\frac{m}{2})$ is assigned to Player-2. If either player decides to withdraw, then the opponent wins automatically; therefore, the value $(-\frac{m}{4})$ is assigned as payoff to the player who withdraws and $(+\frac{m}{2})$ is assigned to the opponent. If both players decide to withdraw, then they must pay the penalty; thus, $(-\frac{m}{4})$ is assigned to both players.

However, when both players have almost equal strength (i.e., Situation-2), then it is hard to predict the winner. Logically, we can say that there is a high chance that the game would end in a tie. Therefore, in the last payoff matrix, the value $(+\frac{m}{2})$ is assigned as the payoff for both players if they both decide to fight.

According to *Definitions* 2 and 3, the *Nash equilibrium* profiles for Case-1 and Case-2 under Situation-1 and Situation-2, respectively, are $\left(+\frac{m}{2}, -\frac{m}{4}\right)$, $\left(-\frac{m}{4}, +\frac{m}{2}\right)$ and $\left(+\frac{m}{2}, +\frac{m}{2}\right)$.

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

IV. Our Approach

Our work has three main phases *i.e., background estimation and image normalization*, *binarization* and *post-processing*. First, the background of the input grayscale image is estimated and then normalized. Second, features from the normalized image are calculated based on the *NC* at the pixel level; then, all these pixels are classified as foreground or background by feeding the extracted features to a K-means clustering algorithm. Finally, the result of *GiB* is post-processed to improve the quality of the output image. A flowchart of *GiB* is given in Fig 2.

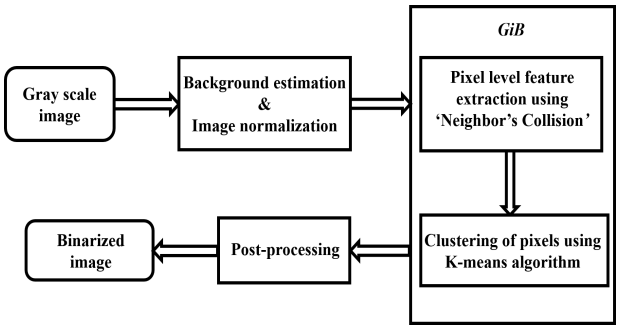

Fig. 2. Flowchart of our game theory based binarization method

A. Background estimation and image normalization

Researchers have adopted several background estimation techniques to make the subsequent processing easier. For example, *Lu et al.* in [40] and *Messaoud et al.* in [41] use a smoothing-based background estimation method. *Gatos et al.* in [21] use an interpolation-based technique, which uses Sauvola's binarization result as a mask. Another technique, called *inpainting*, has been adopted by many researchers for the same purpose. *Zhang et al.* in [8] use a complex and time-consuming version of *inpainting* for background estimation. *Ntirogiannis et al.* in [42] introduce a relatively simple and fast version of *inpainting* for binarization. In our work, we use the *inpainting* technique given in [42] with a minor modification of the mask generation.

1) Generation of an inpainting mask: Inpainting is the process of estimating the missing values of a specific region, called a hole, present in an input image. The idea is to generate a mask image and then replace each pixel in the input grayscale image that corresponds to a data pixel in the mask image based on the intensity values of the neighboring pixels to generate the background image. Finally, this image is used to generate the background separated image. Hence, the performance of the inpainting process depends heavily on the quality of the mask image. In [8], *Zhang et al.* generate a mask image using Canny edges [43] with the application of morphological dilation and closing operations. This process is very time consuming. In [42], *Ntirogiannis et al.* use the Niblack binarization method [18] as a mask because the method preserves almost all textual content in the mask image. However, the major issue is that Niblack binarization is performed with a fixed and relatively large window. For a local threshold-based binarization technique, the selection of the window size is a critical issue. If the size of the window is sufficiently large, then the binarization method may start behaving like a global threshold-based method, which can be highly sensitive to background noise. Thus, the quality of the final background separated image would be compromised (see Fig 3(a)). By contrast, if the window size is too small, then the sliding window may be positioned entirely within the potential foreground region for a document image with a large stroke width. In this case, some hollow regions may appear within the stroke in both the mask and the final background separated images (see Fig 3(b)). On the basis of this fact, we estimate the window size dynamically to generate the inpainting mask using Niblack binarization. Initially, we compute an edge image I_e from the input gray image I_g using the Sobel edge detection technique [40]. Then, we estimate the different stroke widths and their corresponding frequencies from the edge image I_e.

Suppose, $L_{st} = \{l_1, l_2, ..., l_n\}$ denotes the set of n different stroke widths. Consider a frequency function f'' that takes a stroke width $l_i \epsilon L_{st}$, where $1 \leq i \leq n$, and returns its corresponding frequency, i.e., $f'' : L_{st} \rightarrow N$. In our work, we further refine L_{st} based on a specific criterion to generate $L_{st}^{'}$. Any stroke length $l_i \epsilon L_{st}$, where $1 \leq i \leq n$, is kept in $L_{st}^{'}$ iff it satisfies the following criterion,

$$(\mu_{L_{st}} - \alpha \times \sigma_{L_{st}}) \leq l_i \leq (\mu_{L_{st}} + \beta \times \sigma_{L_{st}}) \quad (1)$$

Here, $\mu_{L_{st}} = \frac{\sum_{i=1}^{n} l_i}{n}$, and $\sigma_{L_{st}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(l_i - \mu_{L_{st}})^2}$. The values of α and β are chosen empirically. After the generation of $L_{st}^{'}$, we measure the corresponding frequency list F_{st} such that $F_{st} = \left\{ f''(l) \mid l \epsilon L_{st}^{'} \right\}$. Then, F_{st} is sorted, and the first three frequency values are chosen to obtain the most frequent stroke widths from $L_{st}^{'}$. Next, the average of these three stroke widths is computed to generate S_{avg}. The final window size for the generation of the mask image using Niblack binarization is as follows:

$$W_{mask} = \gamma \times S_{avg} + \varepsilon_0 \quad (2)$$

where $\gamma > 0, \varepsilon_0 > 0$ and ε_0 is an odd number. Here, γ is set to 2 as it helps to consider the closest neighborhood, consisting of both background and foreground pixels. ε_0 ensures the window size is always odd, which is useful for obtaining a well-defined center. After mask generation, the background image I_{back} and the corresponding normalized image I_{norm} are generated using the method described in [42]. The I_{norm} is obtained using I_{back}, and the input gray image I_g is generated as follows:

$$I_{norm} = \left\lceil \left(I_g^{max} - I_g^{min}\right) \times \frac{\left(I_f - I_f^{min}\right)}{\left(I_f^{max} - I_f^{min}\right)} + I_g^{min} \right\rceil \quad (3)$$

Here, I_f is computed as $I_f = \frac{I_g + 1}{I_{back} + 1}$.

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

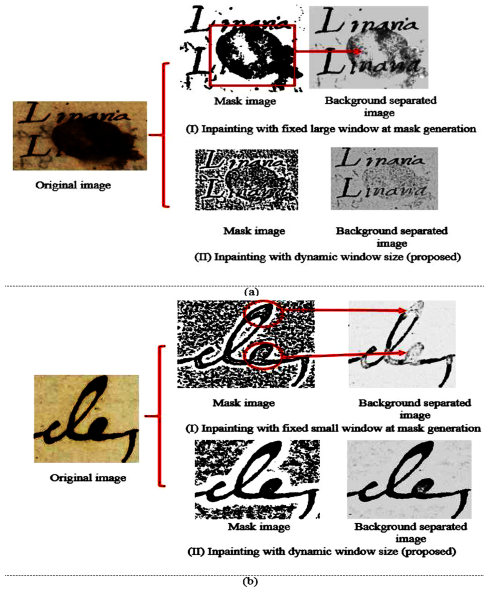

Fig. 3. Demonstration of the problems with using a fixed sized window during the *inpainting* mask generation process and the corresponding effects in the background separated image or normalized image. (a) The effect of using a large fixed-size window vs. a dynamically chosen window during mask generation. (b) The effect of using small fixed-size window vs. a dynamically chosen window during mask generation.

B. Binarization using GiB

Having obtained the output of phase 1, we now discuss the core part of the present binarization method. In this phase, the two-player game *NC* is initially modeled at the pixel level to estimate some useful features from the pixels of I_{norm}; then, based on these features, pixels are classified as either foreground or background using a K-means algorithm.

During the feature extraction process, I_{norm} is scanned with a $w \times w$ overlapping window (here, $w = 3$). The underlying region of each window, K, has a central pixel C_K surrounded by t immediate neighboring pixels $N_1^k, ..., N_t^k$ ($t = 8$ here) (see Fig 4). We collectively call these eight pixels the first-order neighborhood of C_K. We apply *NC* in each window to estimate some useful features for C_K.

We consider C_K to be Player-1, and its intensity is taken as its strength. The first-order neighborhood of C_K is considered to be Player-2. Here, the strength of Player-2 is estimated as the arithmetic mean of the intensity values. According to *Definition 4, NC* can have three possible Cases based on the similarity of the strengths of the participating players. Thus, the next important step is to identify the exact payoff matrix for window K under consideration. Here, the selection of the payoff matrix is done by comparing the strength of C_K with the strength of its first-order neighborhood. Once the payoff matrix is identified, the *Nash equilibrium* is computed.

1) Estimation of payoffs for NC: The estimation of the payoff values plays a crucial role when modeling the game at the pixel level. However, before estimating the values, we must understand their significance in terms of binarization.

During the design of the game NC, we consider the two most obvious and ideal *pixel-level contrast-based scenarios*: (i) *non-uniform regions*, such as the edge region, and

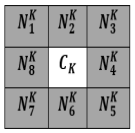

Fig. 4. The players participating in a game. Here, the white cell represents Player-1, i.e., the central pixel, and the gray cells are its first-order neighborhood, collectively known as Player-2

(ii) *uniform regions*, such as completely text or completely background regions. These scenarios are reflected when we consider different *cases* under the different *Situations* stated in the game. In *Situation 1*, we consider *Player 1 and Player 2 to have different strengths*. Here, our intention is to indicate a *non-uniform/high-contrast* region. *Situation 1*, has two *cases*: first, *the strength of Player 1 is greater than the strength of Player 2*; second, the reverse of the first case. By contrast, in *Situation 2*, we assume that *both players are equally strong*, which indicates a *uniform/low-contrast* region. In all three *Cases*, the corresponding payoff matrices have four payoff profiles according to the decisions of the players. We have carefully chosen theses payoff profiles to not only help us to represent the different cases of the game NC but also to capture those *pixel-level contrast-based scenarios* with maximum accuracy. Note that each payoff profile indicates the degree of dissimilarity between a pixel and its locality of reference in terms of intensity and thus has considerable importance in terms of binarization. Let us consider the four payoff profiles of *case 1* under *Situation 1*,

- $(+\frac{m}{2}, -\frac{m}{2})$ reflects a *high-contrast* region where a large intensity level difference exist between the pixel under consideration and its locality of reference. This case likely represents a 3×3 region with a noisy pixel at its center because only a noisy pixel can have such a large intensity level difference with its context.
- $(+\frac{m}{2}, -\frac{m}{4})$ and $(-\frac{m}{4}, +\frac{m}{2})$ both reflect a regular high-contrast scenario with a small intensity level difference compared to $(+\frac{m}{2}, -\frac{m}{2})$.
- $(-\frac{m}{4}, -\frac{m}{4})$ reflects a low-contrast scenario.

From the above discussion, it can be observed that these four profiles are simply four possibilities for a high-contrast scenario. Although the last profile is not highly relevant to binarization as per the assumption made during the consideration of this *Case*, this profile is included in the payoff matrix to make the game a general one.

We now have to choose an appropriate payoff profile that can efficiently represent a common high-contrast pixel-based scenario, where the central pixel has high intensity compared to its context. As we have tried to capture a high-contrast region, the consideration of the payoff profile $(-\frac{m}{4}, -\frac{m}{4})$ is irrational with respect to binarization. In this game, as we have considered two ideal pixel-level contrast-based scenarios, we have performed *background estimation and image normalization* to eliminate the intensity level variation present in the background of an image in the initial stage. This reduces the possibility of the presence of isolated noisy pixels in the image. Therefore, the payoff profile $(+\frac{m}{2}, -\frac{m}{2})$ is not highly

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

useful, as it represents mostly a region with a noisy pixel at its center. This reasoning, in turn, indicates that $(+\frac{m}{2}, -\frac{m}{4})$ and $(-\frac{m}{4}, +\frac{m}{2})$ are the two most relevant profiles with respect to binarization in this *case*. However, because we have assumed that Player 1 is stronger than Player 2, $(+\frac{m}{2}, -\frac{m}{4})$ arises as a more rational profile to more accurately represent the high-contrast relationship between the central pixel and its neighbors. Following the same logic, we see that the profile $(-\frac{m}{4}, +\frac{m}{2})$ becomes more relevant in *Case 2* of *Situation 1*. By contrast, in *Situation 2*, we assume that *both players are equally strong*, which indicates a *uniform/low-contrast* region. Thus, $(+\frac{m}{2}, -\frac{m}{4})$ and $(-\frac{m}{4}, +\frac{m}{2})$ are of no use. However, for the sake of generality of the game, we include these profiles. Now $(+\frac{m}{2}, +\frac{m}{2})$ and $(-\frac{m}{4}, -\frac{m}{4})$ are both relevant with respect to binarization. But according to the game, both players are equally strong; hence, $(+\frac{m}{2}, +\frac{m}{2})$ appears to be the more relevant profile.

The above discussion indicates that the profile $(-\frac{m}{4}, -\frac{m}{4})$ is not considered in any of the *Cases* because the payoff value $(-\frac{m}{4})$ is assigned to a player who wishes to withdraw his/her participation from the game. Therefore, the profile $(-\frac{m}{4}, -\frac{m}{4})$ indicates that no one is ready to participate, which leads to dismissal of the game. From the binarization perspective, dismissal of the game implies that the game is not conducted for a particular window, which is illogical because when we decline to conduct the game for a window, we cannot classify a pixel as foreground or background.

If we carefully consider the *Nash equilibrium* profiles for the payoff matrices of the cases discussed here, we see that these are also $(+\frac{m}{2}, -\frac{m}{4})$, $(-\frac{m}{4}, +\frac{m}{2})$ and $(+\frac{m}{2}, +\frac{m}{2})$.

Notably, the payoff values present in these profiles are a function of the metric m. To estimate the value of m for each window K, we consider the local contrast information, which is a widely used concept for the purpose of binarization. *Bernsen et al.*, in [45], measure the local contrast by computing the difference between the minimum and maximum intensities present within a local neighborhood as follows:

$$C_K^{contrast} = K_{max} - K_{min} \tag{4}$$

where $C_K^{contrast}$ denotes the contrast for the center pixel C_k. K_{max} and K_{min} indicate the maximum and minimum intensity values, respectively, in the local neighborhood K. *Su et al.*, in [46], estimate the local contrast for binarization using the following formula [47]:

$$C_K^{contrast} = \frac{K_{max} - K_{min}}{K_{max} + k_{min} + \varepsilon} \tag{5}$$

where ε is a positive but infinitely small number. The same authors, in [48], use another measure of local contrast, where the difference between the maximum intensity value and the intensity of the central pixel of a local neighborhood is estimated as the local contrast. This process can be expressed as follows:

$$C_K^{contrast} = \frac{K_{max} - C_K}{K_{max} + \varepsilon} \tag{6}$$

In our work, we define a piecewise function to estimate m that also incorporates the concept of local contrast but in a different way.

As a *uniform* region under a 3×3 window can be an entirely text or background region, the contrast measures given in [45], [46] and [47] cannot discriminate these two different variants of uniformity, as they produce a zero value for both cases. On the other hand, a *non-uniform* region contains both text and background pixels. In the adaptive binarization process, the characteristics of the central pixel of a window are generally decided based on the local neighborhood or context. As the pixels around the central pixel vary depending on writing style, ink/paper quality, content etc., classifying pixels with lower intensity values that reside around the edges as foreground or background pixels may not always be straightforward. Thus, the local contrast measures applied in [45], [46] and [48] cannot be directly used for these scenarios. Considering these facts, we define a piecewise function to distinguish these situations as follows:

$$m = \begin{cases} \frac{\sigma^K}{\mu^K}, & if\,(K_{max} - K_{min}) > \varphi \\ \frac{\mu^K}{I_{norm}^{max}}, & otherwise \end{cases} \tag{7}$$

where $\sigma^K = \frac{\sum_{i=1}^{t}\left(N_i^K - \mu^K\right)^2}{t}$, $\mu^K = \frac{\sum_{i=1}^{t} N_i^K}{t}$ and φ is a very small number.

In equation (7), the uniformity and non-uniformity are characterized by $(K_{max} - K_{min})$. All the regions having a value of this part that is less than the threshold φ are considered to be *non-uniform* regions, and the first part of the piecewise function is used to compute the m values for these regions. Otherwise, the second part is used to compute the m values, as the region under consideration is identified as a *uniform* region.

In our work, each pixel in the input image is represented by three features, *i) the intensity value of the central pixel C_K, ii) the payoff for C_K in the equilibrium state, and iii) the intensity difference between C_K and the pixel with maximum intensity among its first-order neighbors*. These three features are used as the input of the K-means algorithm to classify the pixels as foreground or background.

2) Clustering using K-means : Now that we have determined the feature values for each pixel, the next step is to label the pixels as background or foreground. This labeling is performed using the K-means algorithm [49]. K-means is a simple and well-known clustering algorithm that can be extremely useful when the number of clusters is known. In our work, the feature space is initially grouped into three clusters: foreground, background and fuzzy or intermediate. Since the performance of the K-means algorithm changes depending on its initialization [50], random initialization may lead to different outcomes. To overcome this problem, we initialize the K-means algorithm for each image with the help of the *instance-ranking policy* described below.

Let $F = \left\{\vec{f_1}, \vec{f_2}, ..., \vec{f_x}\right\}$ be a set of x column vectors, where each column vector $\vec{f_i} = [a_{i1}, ..., a_{iL}]^T$. For F, we generate a set of rank vectors $R = \{\vec{r_1}, \vec{r_2}, ..., \vec{r_x}\}$, and each element of a rank vector $\vec{r_i} = [p_{i1}, , p_{iL}]^T$ represents the rank of the corresponding element in the column vector $\vec{f_i}$, i.e., the k_{th} element of $\vec{r_i}$ represents the rank of the k_{th} element of $\vec{f_i}$. In doing so, we sort the corresponding vector $\vec{f_i}$ to obtain

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

a temporary vector $\vec{T_i}$ that contains all the elements of $\vec{f_i}$ in ascending order. Then, the position of each element of $\vec{f_i}$ in $\vec{T_i}$ is identified and stored as the rank of this element in $\vec{r_i}$. We then compute a score vector $\vec{G} = [g_1, ..., g_L]^T$ of each of the L components from R. Each g_i in $\vec{G}$ represents the score of the i^{th} instance or row vector in F, i.e, $[f_{1i}, , f_{xi}]$. The value of each element g_i in $\vec{G}$ is computed as follows:

$$g_i = \sum_{u=1}^{x} \vec{r}_{ui} \tag{8}$$

From the score vector, we initialize the first and third cluster centers C_1 and C_3, respectively, as follows:

$$C_1 = \left\{ (f_{i1}, f_{i2}, ..., f_{ix}) \mid g_i = \vec{G}_{min} \right\} \tag{9}$$

$$C_3 = \left\{ (f_{i1}, f_{i2}, ..., f_{ix}) \mid g_i = \vec{G}_{max} \right\} \tag{10}$$

To initialize the second initial cluster center C_2, we generate a difference vector $\vec{D}$ as

$$\vec{D} = \left| \vec{G} - \left(\left\lceil \frac{\vec{G}_{max} + \vec{G}_{min}}{2} \right\rceil \times \vec{I} \right) \right| \tag{11}$$

Here, $\vec{I}$ represents an identity vector of size $L \times 1$.

With the help of $\vec{D}$, C_2 is initialized as:

$$C_2 = \{ (f_{i1}, f_{i2}, ..., f_{ix}) \mid d_i = \vec{D}_{min} \wedge d_i \, is \, the \, i^{th} \, element \, of \, \vec{D} \} \tag{12}$$

According to recent research on clustering-based binarization [35], the desired cluster centers are usually situated along the main diagonal of the feature space. The centers of the text cluster and background cluster should reside at the beginning and end of this diagonal, respectively. Therefore, in our initialization method, we start with two extreme data points as the initial text and background cluster centers. Then, a data point equidistant to these extreme seeds, which represents the third cluster, is considered to be the initial intermediate cluster center for each image.

K-means is performed with these initial cluster centers, and the variance of the elements in each cluster is estimated. The experimental outcome indicates that the cluster corresponding to the background has the lowest variance, and the remaining two clusters have higher variance. Once we have identified the background cluster, we are left with two unlabeled clusters. We can either merge them to form a single unified cluster representing text or we can keep the cluster with the maximum variance as text and the other as background. The decision is based on the ratio between the variances of the two clusters. A ratio less than a threshold $T_{cluster}$ indicates similarity between the cluster elements. In this case, the clusters are combined. Otherwise, the cluster with lower variance is merged with the previously identified background cluster. Here, the value of $T_{cluster}$ is set to 1.45.

TABLE I
INFORMATION ABOUT THE STANDARD DOCUMENT IMAGE BINARIZATION DATABASES CONSIDERED HERE FOR THE EVALUATION OF THE *GiB* METHOD

Database	Number of printed pages	Number of handwritten pages
DIBCO 2009	5	5
H-DIBCO 2010	-	10
DIBCO 2011	8	8
H-DIBCO 2012	-	14
DIBCO 2013	8	8
H-DIBCO 2014	-	10
H-DIBCO 2016	-	10

C. Post-processing

During the post-processing stage, the output of *GiB* is further examined to remove remaining artifacts and to improve the text quality by preserving the stroke connectivity and eliminating isolated pixels. We initially apply a combination of shrink and swell filters [21] on the output of *GiB*. Then, the connected components present in the filtered image are analyzed to eliminate straight lines, such as noise that appears due to dark creases or dark stains.

Let CC^i represents the i^{th} connected component present in the image. H^i_{cc}, W^i_{cc} and A^i_{cc} represent the height, width and area of the bounding box of the i^{th} component, respectively. In the final stage, only those CCs that satisfy the following two conditions are considered to be part of the image:

$$\frac{min\left\{H^i_{cc}, W^i_{cc}\right\}}{max\left\{H^i_{cc}, W^i_{cc}\right\}} > Th_{cc} \tag{13}$$

$$A^i_{cc} > Th_{ar} \tag{14}$$

The values of Th_{cc} and Th_{ar} are computed empirically. Our experiments show that Th_{cc} produces an optimal result when its value is set in the range of 0.1 to 0.3, whereas the optimal value ofTh_{ar} is between 20 and 60.

V. EXPERIMENTAL RESULTS AND DISCUSSION

GiB is evaluated on seven standard databases made publicly available through document image binarization competitions. The images in these databases contain several types of noise and degradation. The details of these databases are given in Table 1. To assess the performance of *GiB*, six evaluation metrics are considered: *precision, recall, F-measure (FM), pseudo F-measure (Fps), peak signal-to-noise ratio (PSNR), and distance reciprocal distortion metric (DRD).*

The evaluation tool used in the HDIBCO 2016 competition [9] and available at the DIBCO competition website [51] is applied to estimate these metrics. A detailed assessment of the performance of *GiB* is given in Table II.

A. Comparison with the methods submitted to the DIBCO contests

In this subsection, the performance of *GiB* is compared with the performances of the methods that appeared in the binarization contests. This comparison is based on four evaluation metrics, *namely*, FM, Fps, $PSNR$ and DRD. Detailed results are provided in Tables III-IX.

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

TABLE II
DETAILED PERFORMANCE ASSESSMENT OF OUR BINARIZATION METHOD ON THE DIBCO DATASETS

Database	FM	Fps	PSNR	DRD	Recall	Precision
DIBCO 2016	91.15± 2.79	93.03± 4.18	19.18± 3.01	3.20± 1.53	89.04± 2.85	93.51± 4.55
DIBCO 2014	94.00± 2.15	96.48± 2.57	19.93± 2.50	1.79± 0.54	92.63± 2.18	95.46± 2.98
DIBCO 2013	91.14± 4.46	94.75± 2.86	19.58± 2.35	2.77± 1.22	89.05± 6.83	93.54± 2.92
DIBCO 2012	90.99± 1.88	92.75± 2.31	19.34± 1.34	3.09± 1.11	89.90± 2.90	91.96± 2.59
DIBCO 2011	90.33± 3.54	93.82± 4.00	18.29± 2.69	2.99± 1.03	87.82± 5.06	93.21± 4.33
DIBCO 2010	90.00± 2.14	92.88± 2.72	19.14± 1.25	2.75± 0.76	88.84± 4.61	92.33± 2.86
DIBCO 2009	92.50± 3.03	94.40± 2.90	19.26± 2.60	2.41± 0.76	91.93± 4.09	93.13± 2.96

TABLE III
PERFORMANCE COMPARISON ON THE H-DIBCO 2016 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	87.61± 6.99	91.28± 8.36	18.11± 4.27	5.21± 5.28
Ranked 2^{nd}	88.72± 4.68	91.84± 4.24	18.45± 3.41	3.86± 1.57
Ranked 3^{rd}	88.47± 4.45	91.71± 4.38	18.29± 3.35	3.93± 1.37
Best results	88.72± 4.68	91.84± 4.24	18.45± 3.41	3.86± 1.57
GiB	**91.15± 2.79**	**93.03± 4.18**	**19.18± 3.01**	**3.20± 1.53**

TABLE IV
PERFORMANCE COMPARISON ON THE H-DIBCO 2014 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	96.88	97.65	22.66	0.902
Ranked 2^{nd}	96.63	97.46	22.40	1.001
Ranked 3^{rd}	93.35	96.05	19.45	2.194
Best results	**96.88**	**97.65**	**22.66**	**0.900**
GiB	94.00	96.48	19.93	1.790

TABLE VI
PERFORMANCE COMPARISON ON THE H-DIBCO 2012 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	89.47	90.18	21.80	3.44
Ranked 2^{nd}	92.86	93.34	20.57	2.66
Ranked 3^{rd}	91.54	93.30	20.14	3.04
Best results	**92.86**	**95.09**	**21.80**	**2.66**
GiB	90.99	92.75	19.34	3.09

TABLE VII
PERFORMANCE COMPARISON ON THE DIBCO 2011 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	80.86	-	16.13	104.43
Ranked 2^{nd}	85.20	-	17.15	15.66
Ranked 3^{rd}	88.74	-	17.84	5.36
Best results	88.72	-	17.84	5.36
GiB	**90.33**	93.82	**18.29**	**2.99**

Tables III, VII IX indicate that *GiB* outperforms all the binarization methods submitted to the H-DIBCO 2016 and DIBCO 2011 and 2009 contests in terms of all the evaluation metrics. Additionally, the standard deviations of the numeric results achieved by *GiB* for each metric in H-DIBCO 2016 are also the smallest, which reflects the consistency of *GiB* compared to the other methods submitted to the contests.

Tables IV and VIII show that although the current method does not perform the best on the H-DIBCO 2014 and 2010 datasets, it outperforms the method that secured 3^{rd} place in these contests in terms of all the considered evaluation metrics. Similarly, Table V shows that in DIBCO 2013, GiB achieves the highest score in terms of Fps and DRD, while it achieves comparable results in terms of FM and $PSNR$. For the H-DIBCO 2012 dataset, the method achieves comparable results for all four metrics (see Table VI).

B. Comparison with state-of-the-art methods

In this subsection, the performance of *GiB* is compared with that of 10 state-of-the-art methods in terms of $PSNR$, $precision$, $recall$ and FM. For this comparison, the DIBCO 2009, 2011 2013 and H-DIBCO 2010 2012 datasets are considered. The comparisons are performed separately for printed and handwritten documents.

Tables X, XI and XII show the performance comparison for the printed document images for the DIBCO 2009, 2011 and 2013 datasets, respectively. Table X indicates that among all 10 methods considered here, *GiB* performs the best in terms of $\{PSNR, recall, and FM\}$ and 3^{rd} best in terms of $precision$. Table XI shows that among the 11 state-of-the-art methods, *GiB* obtains 2^{nd} and 4^{th} place

TABLE VIII
PERFORMANCE COMPARISON ON THE H-DIBCO 2010 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked $1^{st}(a)$	91.50	93.58	19.78	-
Ranked $1^{st}(b)$	89.70	95.15	19.15	-
Ranked 2^{nd}	91.78	94.43	19.67	-
Ranked 3^{rd}	89.73	90.11	18.90	-
Best results	**91.78**	**95.15**	**19.78**	-
GiB	90.00	92.88	19.14	2.75

TABLE V
PERFORMANCE COMPARISON ON THE DIBCO 2013 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	92.12	94.19	20.68	3.10
Ranked 2^{nd}	92.70	93.19	21.29	3.18
Ranked 3^{rd}	91.81	92.67	20.68	4.02
Best results	**92.70**	94.19	**21.29**	3.10
GiB	91.14	**94.75**	19.58	**2.77**

TABLE IX
PERFORMANCE COMPARISON ON THE DIBCO 2009 DATABASE

Method	FM	Fps	PSNR	DRD
Ranked 1^{st}	91.23	-	18.65	-
Ranked 2^{nd}	90.06	-	18.23	-
Ranked 3^{rd}	89.34	-	17.79	-
Best results	91.23	-	18.66	-
GiB	**92.50**	94.4	**19.26**	2.41

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

TABLE X
PERFORMANCE COMPARISON ON PRINTED DOCUMENT IMAGES OF DIBCO 2009

Method	PSNR	Recall	Precision	FM
Su[46]	17.80	0.9220	**0.9591**	0.9393
Ramir[52]	18.03	0.9387	0.9348	0.9366
GPP[21]	16.93	0.9004	0.9539	0.9261
Otsu[16]	16.18	0.9440	0.8737	0.9044
Sau[19]	13.51	0.8161	0.8499	0.8290
ALLT[33]	13.63	0.7258	0.9263	0.8036
IIFA[33]	13.42	0.7276	0.9172	0.7976
Bern[20]	15.18	0.8371	0.9175	0.8723
LCM[35]	16.73	0.9202	0.9295	0.9243
GiB	**18.26**	**0.9497**	0.9384 (rank 3^{rd})	**0.9438**

TABLE XI
PERFORMANCE COMPARISON ON PRINTED DOCUMENT IMAGES OF DIBCO 2011

Method	PSNR	Recall	Precision	FM
Su[46]	15.55	0.8835	0.8357	0.8160
Ramir[52]	0.0190	0.9138	0.9226	**0.9162**
Howe[30]	**17.76**	0.9300	0.8855	0.9007
GPP[21]	15.15	0.8400	0.8822	0.8486
Otsu[16]	14.84	**0.9305**	0.8355	0.8708
Sau[19]	12.92	0.8279	0.8131	0.8177
ALLT[33]	13.61	0.764	0.910	0.828
IIFA[33]	12.88	0.7140	0.8915	0.7819
Bern[20]	14.39	0.8448	0.8733	0.8556
LCM[35]	17.72	0.8806	**0.9363**	0.9068
GiB	17.73 (rank 2^{nd})	0.8884 (rank 4^{th})	0.9319 (rank 2^{nd})	0.9081 (rank 2^{nd})

TABLE XII
PERFORMANCE COMPARISON ON PRINTED DOCUMENT IMAGES OF DIBCO 2013

Method	PSNR	Recall	Precision	FM
Su[46]	**18.89**	**0.9404**	0.9183	**0.9243**
Ramir[52]	17.04	0.9107	0.8848	0.8902
Howe[30]	18.42	0.9368	0.8966	0.9077
GPP[21]	16.35	0.8445	0.9244	0.8778
Otsu[16]	14.72	0.9200	0.7758	0.8186
Sau[19]	14.70	0.8353	0.8396	0.8284
ALLT[33]	14.50	0.7107	0.9174	0.7745
IIFA[33]	13.94	0.6988	0.8896	0.7681
Bern[20]	14.42	0.8742	0.7875	0.8063
LCM[35]	17.59	0.8979	0.9277	0.9107
GiB	18.37 (rank 3^{rd})	0.8940 (rank 6^{th})	**0.9501**	0.9203 (rank 2^{nd})

TABLE XIII
PERFORMANCE COMPARISON ON HANDWRITTEN DOCUMENT IMAGES OF DIBCO 2009

Method	PSNR	Recall	Precision	FM
Su[46]	**22.00**	0.9314	**0.9427**	**0.9369**
Ramir[52]	20.18	0.9332	0.8870	0.9092
GPP[21]	17.74	0.8508	0.8465	0.8365
Otsu[16]	13.92	**0.9450**	0.5806	0.6594
Sau[19]	16.77	0.8516	0.7763	0.7859
ALLT[33]	16.06	0.6378	0.8765	0.6922
IIFA[33]	17.31	0.8088	0.8324	0.7990
Bern[20]	14.02	0.8989	0.5815	0.6531
LCM[35]	19.57	0.8826	0.8879	0.8836
GiB	20.27 (rank 2^{nd})	0.8890 (rank 5^{th})	0.9242 (rank 2^{nd})	0.9062 (rank 3^{rd})

TABLE XIV
PERFORMANCE COMPARISON ON HANDWRITTEN DOCUMENT IMAGES OF H-DIBCO 2010

Method	PSNR	Recall	Precision	FM
Su[46]	**20.12**	0.8816	**0.9645**	**0.9207**
Ramir[52]	19.27	**0.8996**	0.9148	0.9059
GPP[21]	15.96	0.6575	0.9434	0.7494
Otsu[16]	17.44	0.8184	0.9016	0.8527
Sau[19]	16.03	0.7398	0.8833	0.7874
ALLT[33]	14.51	0.4870	0.9525	0.6051
IIFA[33]	16.83	0.6992	0.9263	0.7543
Bern[20]	16.86	0.7586	0.9211	0.8196
LCM[35]	18.08	0.8280	0.9307	0.8758
GiB	19.14 (rank 3^{rd})	0.8884 (rank 2^{nd})	0.9233 (rank 6^{th})	0.9000 (rank 3^{rd})

for $\{PSNR,\, precision,\, and\, FM\}$ and for $\{recall\}$, respectively. Similarly, Table XII reveals that among the 11 methods, *GiB* places 1^{st}, 2^{nd}, 3^{rd} and 6^{th} for $precision$, FM, $PSNR$ and $recall$, respectively. An image-level comparison for printed document images is given in Figs 5 and 6.

Tables XIII, XIV, XV, XVI and XVII present the performance comparison for the handwritten document images for the DIBCO 2009, H-DIBCO 2010, DIBCO 2011, H-DIBCO 2012 and DIBCO 2013 datasets. Table XIII indicates that our method achieves 2^{nd}, 3^{rd} and 5^{th} place for $\{PSNR\, and\, precision\}$, FM and $recall$, respectively, whereas Table XIV shows that the proposed method achieves 2^{nd}, 3^{rd} and 6^{th} place for $recall$, $\{PSNR\, and\, FM\}$ and $precision$, respectively. Table XV reports that out of the 11 methods, our method places 1^{st}, 2^{nd} and 5^{th} for $\{precision\, and\, FM\}$, $PSNR$ and $recall$, respectively. According to Table XVI, our method places 3^{rd}, 4^{th} and 7^{th} in terms of FM, $\{PSNR\, and\, recall\}$ and $precision$, out of 11 methods. Finally, Table XVII shows that our method ranks 3^{rd}, 4^{th} and 5^{th} for $recall$, FM and $\{PSNR\, and\, precision\}$, respectively. An image-level comparison for handwritten document images is given in Figs. 7 and 8.

In addition to the above comparisons, we have also compared the performance of *GiB* with some recent deep learning methods. For this comparison, all the DIBCO databases are used. Detailed results are given in Table XVIII. For DIBCO 2009 and H-DIBCO 2016, *GiB* outperforms the deep learning methods, whereas for the rest of the databases, *GiB* achieves comparable performance.

Figs 5-10 show some sample images from the datasets, along with the corresponding binarization results generated by our method and the state-of-the-art methods. Figs 5, 6 and 7 show that *GiB* not only eliminates the background noise but also preserves the character strokes. In Fig 11, we present an image-level comparison of *GiB* and some recently developed deep learning methods.

VI. CONCLUSION

In this paper, we have designed a two-player game called *NC* for document image binarization. The information extracted from each pixel by conducting this game is fed to a K-means clustering algorithm to label the pixels as foreground or

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

TABLE XV
PERFORMANCE COMPARISON ON HANDWRITTEN DOCUMENT IMAGES OF DIBCO 2011

Method	PSNR	Recall	Precision	FM
Su[46]	15.55	0.8835	0.8357	0.8160
Ramir[52]	18.08	0.9139	0.8659	0.8838
Howe[30]	**19.32**	0.8753	0.9101	0.8721
GPP[21]	16.74	0.8091	0.8719	0.8318
Otsu[16]	15.03	0.8461	0.7471	0.7671
Sau[19]	14.55	0.8400	0.7154	0.7556
ALLT[33]	14.95	0.6158	0.8690	0.7133
IIFA[33]	16.25	0.7672	0.8804	0.8098
Bern[20]	14.97	0.8101	0.7606	0.7658
LCM[35]	18.31	**0.9173**	0.8673	0.8897
GiB	18.86 (rank 2^{nd})	0.8679 (rank 5^{th})	**0.9324**	**0.8985**

TABLE XVI
PERFORMANCE COMPARISON ON HANDWRITTEN DOCUMENT IMAGES OF H-DIBCO 2012

Method	PSNR	Recall	Precision	FM
Su[46]	19.62	0.8692	0.9355	0.8887
Ramir[52]	20.28	0.9218	0.9314	0.9258
Howe[30]	**22.27**	**0.9485**	**0.9560**	**0.9521**
GPP[21]	17.04	0.7481	0.9399	0.8178
Otsu[16]	15.57	0.8647	0.7775	0.7748
Sau[19]	16.29	0.7777	0.8535	0.8008
ALLT[33]	14.91	0.4766	0.9374	0.5956
IIFA[33]	16.49	0.6745	0.9235	0.7456
Bern[20]	15.61	0.8383	0.7789	0.7666
LCM[35]	18.72	0.9077	0.8922	0.8971
GiB	19.34 (rank 4^{th})	0.8990 (rank 4^{th})	0.9196 (rank 7^{th})	0.9099 (rank 3^{rd})

background. Before binarization, an inpainting-based preprocessing method is applied to the grayscale images to eliminate the intensity variation present in the background of the input image. Finally, a post-processing step is performed to improve the output of the binarization. The developed method has been tested on seven publicly available datasets consisting of document images, namely, DIBCO 2009, 2011, and 2013 and H-DIBCO 2010, 2012, 2014 and 2016. The experimental outcomes show that the proposed method achieves promising results on these datasets. A performance comparison of the proposed method with state-of-the-art methods shows that the proposed method not only obtained reasonably stable performance on the datasets but also outperformed the state-of-the-art methods in many cases. As future research, we plan to extend the concept of the *GiB* approach by considering Bayesian and multi-player games to capture more relevant information from the pixels of the input images.

TABLE XVII
PERFORMANCE COMPARISON ON HANDWRITTEN DOCUMENT IMAGES OF DIBCO 2013

Method	PSNR	Recall	Precision	FM
Su[46]	22.84	0.8724	**0.9833**	0.9207
Ramir[52]	21.58	**0.9123**	0.9289	0.9184
Howe[30]	**23.80**	0.8685	0.9738	0.8945
GPP[21]	18.77	0.7744	0.9012	0.7927
Otsu[16]	18.43	0.7714	0.8844	0.7769
Sau[19]	16.30	0.8030	0.7629	0.7413
ALLT[33]	17.23	0.6470	0.9035	0.7281
IIFA[33]	17.81	0.7725	0.8694	0.7901
Bern[20]	17.32	0.7502	0.8560	0.7456
LCM[35]	21.57	0.8948	0.9492	**0.9209**
GiB	20.79 (rank 5^{th})	0.8870 (rank 3^{rd})	0.9208 (rank 5^{th})	0.9025 (rank 4^{th})

TABLE XVIII
PERFORMANCE COMPARISON WITH SOME RECENT DEEP LEARNING BINARIZATION METHODS

Database	Methods	FM	PSNR	DRD
H-DIBCO 2016	PDNet [37]	90.18	18.99	3.61
	TM [38]	89.52	18.67	3.76
	HDSN [36]	90.10	19.01	3.58
	GiB	**91.15**	**19.18**	**3.20**
H-DIBCO 2014	PDNet [37]	89.99	20.52	7.42
	TM [38]	91.96	20.76	2.72
	HDSN [36]	**96.66**	**23.23**	**0.79**
	GiB	94.00	19.93	1.79
DIBCO 2013	PDNet [37]	93.97	21.30	1.83
	TM [38]	93.17	20.71	2.21
	HDSN [36]	**94.4**	**21.4**	**1.80**
	GiB	91.14	19.58	2.77
H-DIBCO 2012	PDNet [37]	**93.04**	20.50	2.92
	TM [38]	92.53	**20.60**	**2.48**
	GiB	90.99	19.34	3.09
DIBCO 2011	PDNet [37]	91.87	19.07	2.57
	TM [38]	**93.60**	**20.11**	**1.85**
	HDSN [36]	93.30	20.10	2.00
	GiB	90.33	18.29	2.99
H-DIBCO 2010	PDNet [37]	92.91	20.40	1.85
	TM [38]	**94.89**	**21.84**	**1.26**
	GiB	90.00	19.14	2.75
DIBCO 2009	PDNet [37]	91.50	19.25	3.06
	TM [38]	89.76	18.43	4.89
	GiB	**92.50**	**19.26**	**2.41**

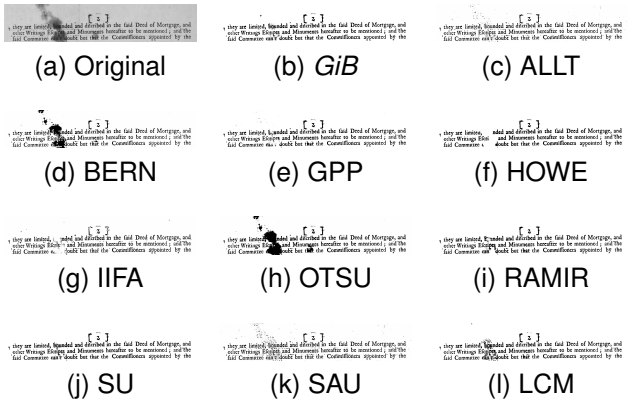

Fig. 5. Binarization results for a printed document image taken from the DIBCO 2009 database

ACKNOWLEDGMENT

We would like to thank CMATER research laboratory of the Computer Science and Engineering Department, Jadavpur University, India, for providing us the infrastructural support. This work is partially supported by the PURSE-II and UPE-II, Jadavpur University projects. Showmik Bhowmik is thankful to the Ministry of Electronics and Information Technology (MeitY), Govt. of India, for providing him a PhD-Fellowship under the Visvesvaraya PhD scheme. Ram Sarkar is partially funded by DST grant (EMR/2016/007213).

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

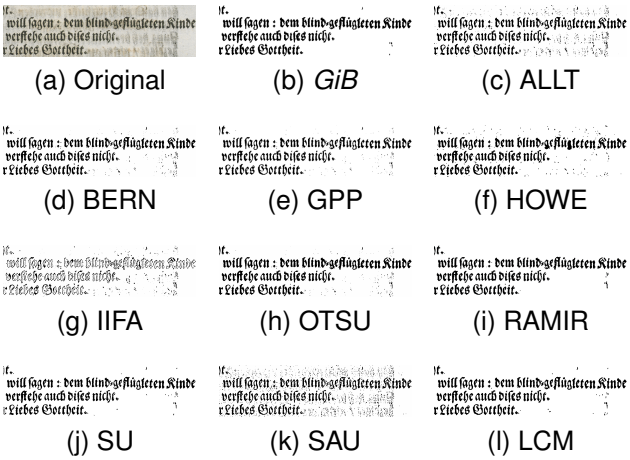

Fig. 6. Binarization results for a printed document image taken from the DIBCO 2011 database.

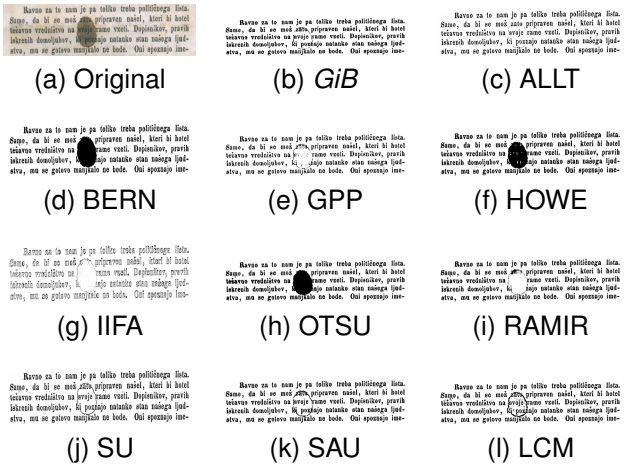

Fig. 7. Binarization results for a printed document image taken from the DIBCO 2013 database

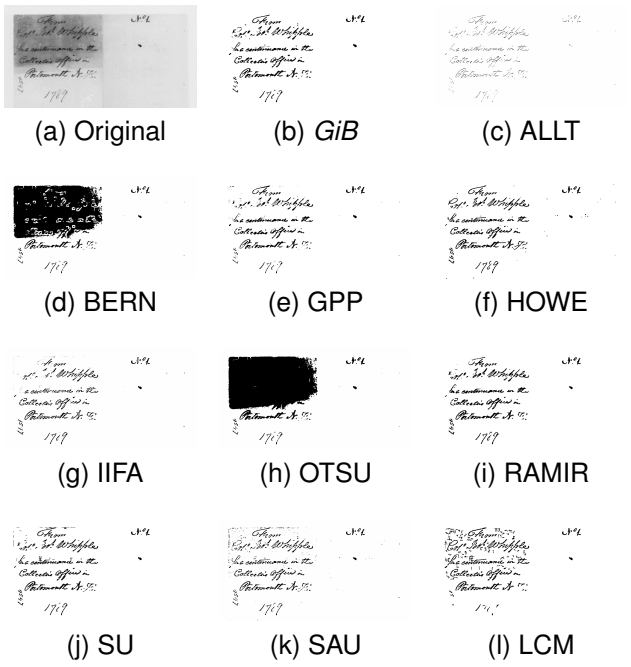

Fig. 8. Binarization results for a handwritten document image taken from the DIBCO 2009 database

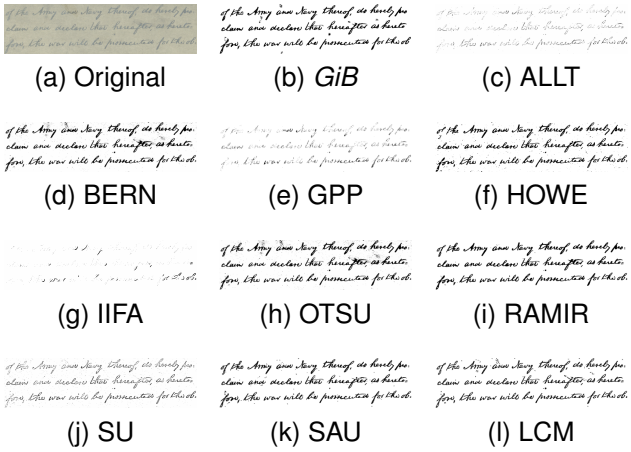

Fig. 9. Binarization results for a handwritten document image taken from the DIBCO 2010 database

REFERENCES

[1] Y. Chen and L. Wang, "Broken and Degraded Document Images Binarization," Neurocomputing, vol. 237, pp. 272-280, 2017.

[2] F. Jia, C. Shi, K. He, C. Wang, and B. Xiao, "Document image binarization using structural symmetry of strokes," in *Frontiers in Handwriting Recognition (ICFHR),* 15^{th} *International Conference on*, 2016, pp. 411-416.

[3] S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, and D. K. Basu, "A hierarchical approach to recognition of handwritten Bangla characters," Pattern Recognition., vol. 42, no. 7, pp. 1467-1484, 2009.

[4] S. Bhowmik, S. Polley, M. G. Roushan, S. Malakar, R. Sarkar, and M. Nasipuri, "A holistic word recognition technique for handwritten Bangla words," International Journal of Applied Pattern Recognition, vol. 2, no. 2, pp. 142-159, 2015.

[5] S. Bhowmik, S. Malakar, R. Sarkar, and M. Nasipuri, "Handwritten Bangla word recognition using elliptical features," in *Proceedings* - 6^{th} *International Conference on Computational Intelligence and Communication Networks, CICN* pp. 257-261, 2014.

[6] T. A. Tran, K. Oh, I. S. Na, G. S. Lee, H. J. Yang, and S. H. Kim, "A Robust System for Document Layout Analysis Using Multilevel Homogeneity Structure," Expert Systems with Application, vol. 85, pp. 99-113, 2017.

[7] H. Z. Nafchi, R. F. Moghaddam, and M. Cheriet, "Phase-based binarization of ancient document images: Model and applications," IEEE Transactions on Image Processing, vol. 23, no. 7, pp. 2916-2930, 2014.

[8] L. Zhang, A. M. Yip, M. S. Brown, and C. L. Tan, "A unified framework for document restoration using inpainting and shape-from-shading," Pattern Recognition, vol. 42, no. 11, pp. 2961-2978, 2009.

[9] I. Pratikakis, K. Zagoris, G. Barlas, and B. Gatos, "ICFHR2016 Handwritten Document Image Binarization Contest (H-DIBCO 2016)," in *Frontiers in Handwriting Recognition (ICFHR),* 15^{th} *International Conference on*, 2016, pp. 619-623.

[10] K. Ntirogiannis, B. Gatos, and I. Pratikakis, "ICFHR2014 competition on handwritten document image binarization (H-DIBCO 2014)," in *Frontiers in Handwriting Recognition (ICFHR),* 14^{th} *International Conference on*, 2014, pp. 809-813.

[11] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "ICDAR 2013 document image binarization contest (DIBCO 2013)," in *Document Analysis and Recognition (ICDAR),* 12^{th} *International Conference on*, 2013, pp. 1471-1476.

[12] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "ICFHR 2012 competition on handwritten document image binarization (H-DIBCO 2012)," in *Frontiers in Handwriting Recognition (ICFHR), International Conference on*, 2012, pp. 817-822.

[13] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "ICDAR 2011 Document

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

Fig. 10. Binarization results for a handwritten document image taken from the DIBCO 2012 database

Image Binarization Contest (DIBCO 2011)," in *Document Analysis and Recognition (ICDAR), International Conference on*, 2011, pp. 1506-1510.

[14] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "H-DIBCO 2010-handwritten document image binarization competition," in *Frontiers in Handwriting Recognition (ICFHR), International Conference on*, 2010, pp. 727-732.

[15] B. Gatos, K. Ntirogiannis, and I. Pratikakis, "ICDAR 2009 document image binarization contest (DIBCO 2009)," in *Document Analysis and Recognition, ICDAR09.* 10^{th} *International Conference on*, 2009, pp. 1375-1382.

[16] Otsu, N. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9(1), pp. 62-66, 1979.

[17] J. Kittler, J. Illingworth, and J. Fglein, "Threshold selection based on a simple image statistic," Computer Vision, Graphics, Image Processing, vol. 30, no. 2, pp. 125-147, 1985.

[18] W. Niblack, "An introduction to digital image processing," vol. 34. Englewood Cliffs: Prentice-Hall, 1986.

[19] J. Sauvola and M. Pietikinen, "Adaptive document image binarization," Pattern Recognition, vol. 33, no. 2, pp. 225-236, 2000.

[20] J. Bernsen, "Dynamic thresholding of grey-level images," in *International conference on pattern recognition*, 1986, vol. 2, pp. 1251-1255.

[21] B. Gatos, I. Pratikakis, and S. J. Perantonis, "Adaptive degraded document image binarization," Pattern Recognition, vol. 39, no. 3, pp. 317-327, 2006.

[22] O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 12, pp. 1191-1201, 1995.

[23] M. Sezgin, "Survey over image thresholding techniques and quantitative performance evaluation," Journal of Electronic imaging, vol. 13, no. 1, pp. 146-168, 2004.

[24] A. Mishra, K. Alahari, and C. V Jawahar, "Unsupervised refinement of color and stroke features for text binarization," International Journal on Document Analysis and Recognition, pp. 1-17, 2017.

[25] D. Lu, X. Huang, C. Liu, X. Lin, H. Zhang, and J. Yan, "Binarization of degraded document image based on contrast enhancement," in *Control Conference (CCC),* 35^{th} *Chinese*, 2016, pp. 4894-4899.

[26] X. Peng, S. Setlur, V. Govindaraju, and R. Sitaram, "Markov random field based binarization for hand-held devices captured document images," in *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing*, 2010, pp. 71-76.

[27] J. G. Kuk and N. I. Cho, "Feature based binarization of document images degraded by uneven light condition," in *Document Analysis and Recognition, ICDAR09.* 10^{th} *International Conference on*, 2009, pp. 748-752

[28] C. Wolf and D. Doermann, "Binarization of low quality text using a Markov random field model," in *Pattern Recognition, 2002. Proceedings.* 16^{th} *International Conference on*, 2002, vol. 3, pp. 160-163.

[29] N. R. Howe, "A Laplacian energy for document binarization," in *Document Analysis and Recognition (ICDAR), International Conference on*, 2011, pp. 6-10.

[30] N. R. Howe, "Document binarization with automatic parameter tuning," International Journal on Document Analysis and Recognition, vol. 16, no. 3, pp. 247-258, 2013.

[31] N. Papamarkos, "A neuro-fuzzy technique for document binarisation," Neural Computing and Applications, vol. 12, no. 34, pp. 190-199, 2003.

[32] O. D. Trier and T. Taxt, Evaluation of binarization methods for document images, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 3, pp. 312-315, 1995.

[33] E. Badekas and N. Papamarkos, "Document binarisation using Kohonen SOM," IET Image Processing, vol. 1, no. 1, pp. 67-84, 2007.

[34] E. Badekas and N. Papamarkos, "Optimal combination of document binarization techniques using a self-organizing map neural network," Engineering Applications of Artificial Intelligence, vol. 20, no. 1, pp. 11-24, 2007.

[35] N. Mitianoudis and N. Papamarkos, "Document image binarization using local features and Gaussian mixture modeling," Image and Vision Computing, vol. 38, pp. 33-51, 2015.

[36] Q. N. Vo, S. H. Kim, H. J. Yang, and G. Lee, Binarization of degraded document images based on hierarchical deep supervised network, Pattern Recognition., vol. 74, pp. 568-586, 2018.

[37] K. R. Ayyalasomayajula, F. Malmberg, and A. Brun, PDNet: Semantic Segmentation integrated with a Primal-Dual Network for Document binarization, Pattern Recognition Letters., 2018.

[38] C. Tensmeyer and T. Martinez, Document Image Binarization with Fully Convolutional Neural Networks, arXiv Prepr. arXiv1708.03276, 2017.

[39] M. J. Osborne and A. Rubinstein, A course in game theory. MIT press, 1994.

[40] 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, 2010.

[41] I. Ben Messaoud, H. Amiri, H. El Abed, and V. Margner, "New binarization approach based on text block extraction," in *Document Analysis and Recognition (ICDAR), International Conference on*, 2011, pp. 1205-1209.

[42] K. Ntirogiannis, B. Gatos, and I. Pratikakis, "A combined approach for the binarization of handwritten document images," Pattern Recognition Letters, vol. 35, pp. 3-15, 2014.

[43] P. Bao, L. Zhang, and X. Wu, "Canny edge detection enhancement by scale multiplication," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 9, pp. 1485-1490, 2005.

[44] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the Sobel operator," IEEE Journal of Solid-State Circuits, vol. 23, no. 2, pp. 358-367, 1988.

[45] J. Bresen, "Dynamic thresholding of gray-level images", in 8^{th} *International Conference on Pattern Recognition, IEEE Computer Society Press*, pp. 1251-1255, 1986.

[46] B. Su, S. Lu, and C. L. Tan, "Binarization of historical document images using the local maximum and minimum," in 9^{th} *IAPR International Workshop on Document Analysis Systems*, 2010, pp. 159-166.

[47] M. Van Herk, "A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels," Pattern Recognition Letters, vol. 13, no. 7, pp. 517-521, 1992.

[48] B. Su, S. Lu, and C. L. Tan, "Combination of document image binarization techniques," in *Document Analysis and Recognition (ICDAR), International Conference on*, 2011, pp. 22-26.

[49] S. Lloyd, "Least squares quantization in PCM," IEEE Transactions on Information Theory, vol. 28, no. 2, pp. 129137, 1982.

[50] G. Tzortzis and A. Likas, "The MinMax k-Means clustering algorithm," Pattern Recognition, vol. 47, no. 7, pp. 2505-2516, 2014.

[51] H-DIBCO2016: Handwritten Document Image Binarization Contest, 2016. [Online], Accessed: 20-Jan-2017, Available: http://vc.ee.duth.gr/h-dibco2016/.

[52] M. A. Ramrez-Ortegn, L. L. Ramrez-Ramrez, I. Ben Messaoud, V. Mrgner, E. Cuevas, and R. Rojas, "A model for the gray-intensity distribution of historical handwritten documents and its application for binarization," International Journal on Document Analysis and Recognition, vol. 17, no. 2, pp. 139-160, 2014.

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2018.2878959, IEEE Transactions on Image Processing

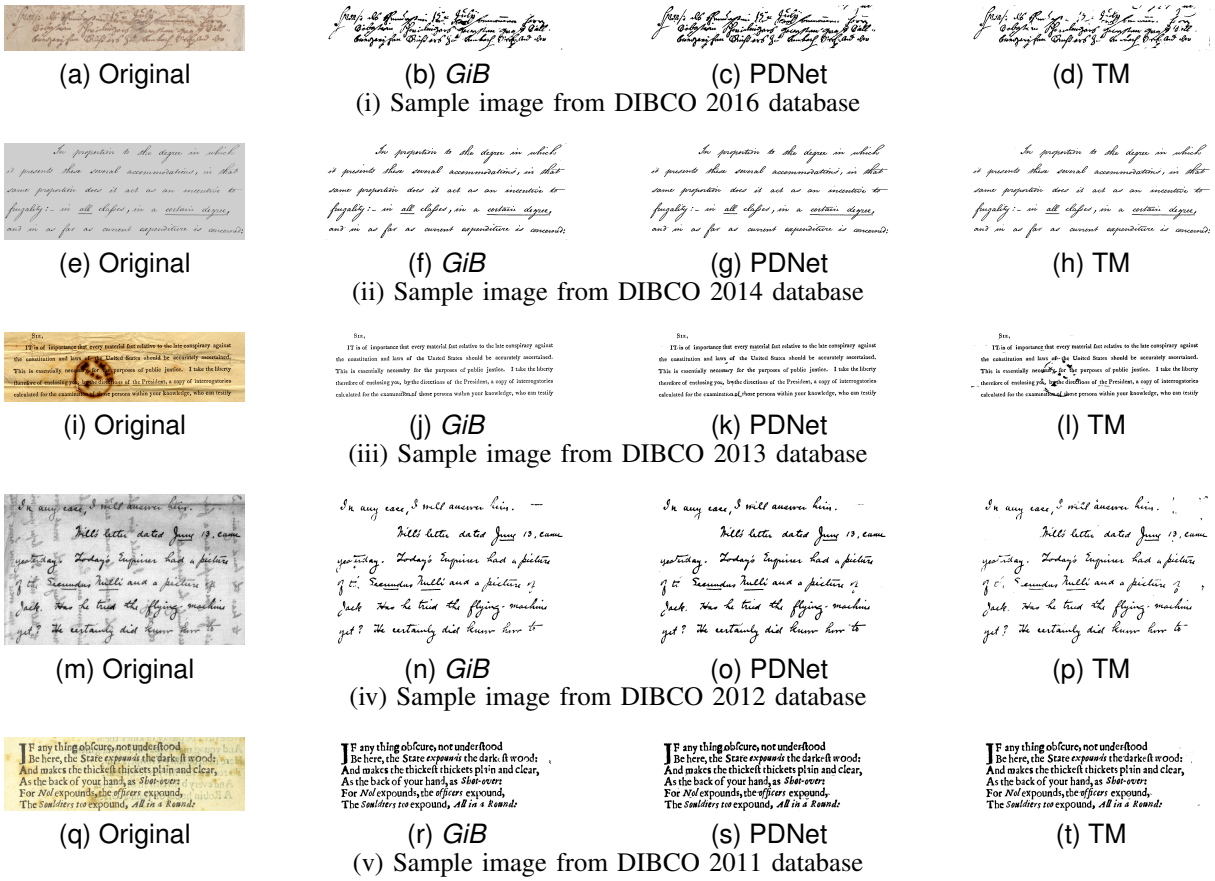

Fig. 11. Image-level performance comparison with some recently developed deep learning binarization methods

Showmik Bhowmik received his B. Tech degree in Computer Science and Engineering from West Bengal University of Technology, India, in 2008. He received his M.C.S.E degree from Jadavpur University, India, in 2014. Currently, he is pursuing a PhD in Engineering from Jadavpur University. His areas of current research interest are document image processing, offline handwriting recognition, machine learning, and soft computing.

Bishwadeep Das received his B. Tech degree in Electronics and Communication Engineering from NIT Allahabad, India in 2016. He is currently a second year Masters Student in Electrical Engineering at TU Delft, Netherlands. His areas of current research interest are statistical signal processing, convex optimization, pattern recognition, and estimation theory.

Ram Sarkar received his B. Tech degree in Computer Science and Engineering from University of Calcutta, India in 2003. He received his M.E. degree in Computer Science and Engineering and PhD (Engineering) degree from Jadavpur University, India, in 2005 and 2012, respectively. He joined Department of Computer Science and Engineering, Jadavpur University as an Assistant Professor in 2008, where he is currently working as an Associate professor. He received the Fulbright-Nehru Fellowship (by USIEF) for post-doctoral research at the University of Maryland, College Park, USA in 2014-15. His areas of current research interest are image processing, pattern recognition, machine learning, and bioinformatics. He is a senior member of the IEEE, U.S.A.

David Doermann received a B.Sc. degree in Computer Science and Mathematics from Bloomsburg University, Bloomsburg, PA, USA, in 1987, M.Sc. and PhD degrees from the University of Maryland, College Park, in 1989 and 1993, respectively, and an honorary degree in technology sciences from the University of Oulu, Oulu, Finland, in 2002 for his contributions to digital media processing and document analysis research. He was a Research Scientist at the University of Maryland from 1993-2018 and a Program Manager at the Defense Advanced Research Projects Agency (DARPA) from 2014-2018. He is currently a Professor in the Department of Computer Science and Engineering at the University at Buffalo and the inaugural Director of the University at Buffalo Artificial Intelligence Institute. He is a Founding Co-Editor of the International Journal on Document Analysis and Recognition, was the General Chair or Co-Chair of over a half dozen international conferences and workshops, and was the General Chair of the International Conference on Document Analysis and Recognition in 2013. He is a fellow of the IAPR and the IEEE.

1057-7149 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.