Pattern Analysis and Applications (2019) 22:1361–1375
https://doi.org/10.1007/s10044-018-0687-2

THEORETICAL ADVANCES

A novel method for binarization of scene text images and its application in text identification

Ranjit Ghoshal[1] · Anandarup Roy[2] · Ayan Banerjee[3] · Bibhas Chandra Dhara[4] · Swapan K. Parui[5]

Received: 21 March 2017 / Accepted: 18 January 2018 / Published online: 14 February 2018

Abstract
The aim of this article is twofold. First, we propose an effective methodology for binarization of scene images. For our present study, we use the publicly available ICDAR 2011 Born Digital Data set. We introduce a new concept of variance map of a gray-level image for detection of text boundary in an image. Based on this boundary information, the image is binarized by means of adaptive thresholding. This binarization procedure produces a number of connected components. Next, these connected components are examined in order to identify possible text components. In this context, a number of shape-based features that distinguish between text and non-text components are proposed. We consider text component identification as an one-class classification problem, i.e., the ground truth information for only the text class is available for the ICDAR 2011 Born Digital Data set. Then, the ground truth text components are used to obtain a certain statistical distribution of the shape-based features. Here, we observe that all the features may not follow a single family of distributions. Therefore, we construct a joint distribution by using multivariate Gaussian copula which allows a coupling of different marginal distributions. As our experiments suggest, the copula-based method is superior to multivariate Gaussian distribution in describing the feature distribution. Finally, a text connected component of an unknown class is subjected to the trained statistical model, and by performing a hypothesis test we successfully identify a possible text component. For a comparative study, we consider a number of state-of-the-art methods. Our proposed approach significantly outperforms most of these methods in terms of recall, precision and F-measure in both the binarization and text identification tasks.

Keywords Scene image binarization · Scene text identification · Connected component · One-class classifier · Gaussian copula

1 Introduction

Mobile phones, digital cameras, various handheld devices and state-of-the-art intelligent robotic systems are becoming increasingly popular. They are equipped with various technologies that are useful in all spheres of life. Manufacturers are now enabling these devices with technologies like extraction and recognition of text from scene images, text-to-speech converter, content-based web image search, video information retrieval, etc. These text-based softwares work by extracting the text from scene image. Text image binarization is one of the important steps of any document analysis system. The performance of the subsequent steps like character segmentation and recognition are highly

✉ Ranjit Ghoshal
ranjit.ghoshal.stcet@gmail.com

Anandarup Roy
roy.anandarup@gmail.com

Ayan Banerjee
ayanbanerjee.stcet@gmail.com

Bibhas Chandra Dhara
bibhas@it.jusl.ac.in

Swapan K. Parui
swapan@isical.ac.in

[1] St. Thomas' College of Engineering and Technology, Kolkata 700023, India

[2] Usha Martin University, 12th Mile, Ranchi Khunti Road, Ranchi, Jharkhand 835221, India

[3] Lexmark Research and Development Corporation, Kolkata, India

[4] Department of Information Technology, Jadavpur University, Kolkata 700098, India

[5] CVPR Unit, Indian Statistical Institute, 203 B. T. Road, Kolkata 700108, India

dependant on the success of binarization. Text image binarization has been an active area of research for several years due to the emerging need for the recognition of text in video sequences, born-digital images, old historic manuscripts and natural scenes where the state-of-the-art recognition performance is really poor. So, designing a powerful binarization scheme can be considered as a major contribution toward robust text understanding. In case of born-digital images, text is superimposed by a software. Born-digital images are applied in web pages and e-mail as logo, name or ads. In Fig. 1, we have shown two sample images and their corresponding ground truth images from ICDAR 2011 Born Digital Data set [13]. The resolution of the text present in the image and anti-aliasing of text are the major dissimilarities between scene and born-digital images. Numerous methods exist for text image binarization in document images, but they cannot be directly applied on natural scene images as the size of the text can vary drastically from a few pixels to a large part of the image. Their orientation also changes from image to image. Scene images are much more complex than document images as the background is in most cases not uniform and simple as in the case of document images. Moreover, illumination variation caused by light reflection, shadows and other noise sources add significant complexity to the problem. Thus, to solve this problem, more effective approach is required. Conventional binarization techniques are classified into two categories (1) global thresholding and (2) local adaptive thresholding. Global thresholding is quite simple and effective for simple images with bimodal histogram. But in scene images, the histogram is much more complex to give any fruitful result of Otsu's method [23]. Binarization of such an image using a single threshold value often leads to loss of textual information against the background. On the other hand, local thresholding methods, Niblack [22], Sauvola [25] and Lu [20], apply window-based processing where the size of the window is crucial for text detection. In scene images, the size of the text can vary drastically; thus, to choose a window size is an impossible task. Among recent works, Gatos et al. [6] proposed a document image binarization model by combining multiple global and local adaptive methodologies and reinforcing it with edge information. On the other hand, Automatic recognition of text symbols in a natural scene image is useful to the blind and foreigners with language barrier. Such a recognition method should also employ an extraction of text portions from the scene images. Extraction and recognition of texts from outdoor images captured by such devices is a challenging problem nowadays due to variations in style, color, background complexity, etc. Earlier, Jung et al. [10] employed a multilayer perceptron classifier to discriminate between text and non-text pixels. A survey work of existing methods for detection, localization and extraction of texts embedded in natural scene images can be found in [17].

There have been several studies on text extraction in the last few years. Wu et al. [31] use a local threshold method to extract texts from gray image blocks containing texts. By considering that texts in images and videos [27] are always colorful, Tsai et al. [29] developed a threshold method using intensity and saturation features to extract texts in color document images. Lienhart et al. [18] and Sobottka et al. [28] use color clustering algorithm for text segmentation. A sliding window scans the whole image and serves as the input to a neural network. High-probability areas inside a probability map are considered as candidate text regions. Wavelet transform has also been applied for text identification [16, 26]. In this context, Gllavata et al. [8] considered wavelet transform and K-means-based texture analysis for text detection. More recently, Bhattacharya et al. [1] proposed a scheme based on analysis of connected components (CCs) for the extraction of Devanagari and Bangla texts from camera-captured natural scene images. Also a few criteria for robust filtering of text connected components have been studied.

This paper is organized as follows: Section 2 describes in detail the proposed binarization methodology, Sect. 3 describes the details of the features, Sect. 4 describes text identification methodology and results and discussion are given in Sect. 5. Finally, in Sect. 6 we discuss the summary and future scope.

Fig. 1 Sample images and their corresponding ground truth images from ICDAR 2011 Born Digital Data set. **a**, **b** Sample images and **c**, **d** ground truth images

2 Proposed binarization methodology

This section summarizes the details of our proposed binarization methodology. Figure 2 shows an overview of the overall binarization architecture. First, a color scene image containing text is converted into a grayscale image which is input to our algorithm. Our proposed methodology for text image binarization is composed of the following four sections.

2.1 Variance calculation

Text components have certain basic characteristics based on which they can be distinguished in an image—(a) they have a distinct boundary unlike non-text regions, (b) color of the text region is homogeneous and is different from the background non-text portion. These basic features of text regions are exploited to classify the image into text and non-text clusters. Generally, edge detection techniques are applied to find the boundary of a text. There are numerous methods to extract text based on its edge properties. But the basic problem with this method is that it is highly susceptible to noise. Edge detection works by computing the gradient of the image and then finding the high gradient line segments. Noise causes high gradient values and thus contributes to faulty edges. Here, we use another method to detect the boundary. A boundary pixel corresponding to a text region usually has a highly different gray value compared to the neighboring background pixels. In other words, within an image $n \times n$ window, a text boundary pixel is detectable by its higher variance of gray values compared to the non-boundary pixels. Even when the background is fairly uneven in terms of gray values, we observed that text boundary pixels are identifiable in most cases. Finally, we observed that noise could not make such a significant increase in variance compared to text boundary, in a neighborhood window. Thus, detecting text boundary based on variance is fairly robust with respect to noise. Thus, this method is immune to noise and also effectively detects boundary regions with high accuracy. Let us now take an $n \times n$ (n is an user choice) neighborhood window and compute the variance map (Fig. 3c) for the whole image. Afterward, we apply Canny edge detection principle [2] to obtain edges (hereafter boundary lines (Fig. 4a)) from the variance map, considered as a gray-level image, which is a new concept.

2.2 Boundary linking

The *boundary lines* originating from the *variance map* can partially define the boundaries of the text regions. In fact, there are discontinuities in the boundaries due to low-variance regions. Here, we take the following strategy to obtain a complete description of the boundaries. First, we intend to obtain insignificant edges in the image. Such edges are obtained from the gray image (Fig. 3b) by applying Canny edge detector with a small sensitivity threshold. Among these edges (Fig. 4b), we keep only the edges which are connected with the *boundary* lines. This process almost successfully recovers all the boundary lines. However, a boundary

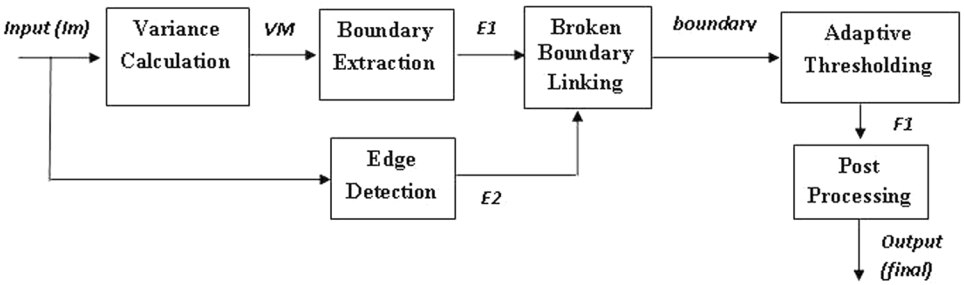


Fig. 2 Block diagram of the proposed edge and variance-based scene image binarization methodology

Fig. 3 **a** Input color image. **b** Gray image (Im), **c** variance map (VM) with $n = 5$

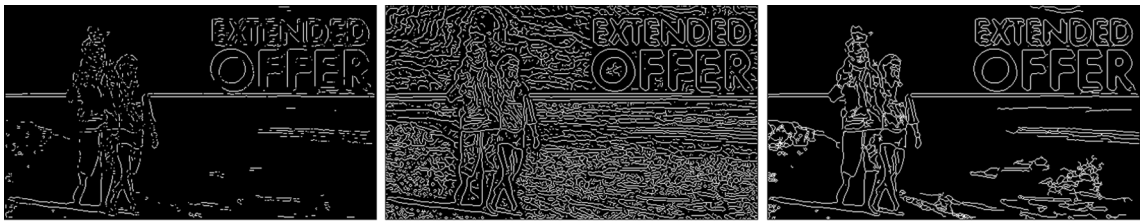


Fig. 4 **a** Boundary lines (E1), **b** Canny edge map of *Im (E2)*, **c** complete boundary map *(boundary)*

line may still be discontinuous by only a few pixels. Thus a morphological bridging is performed next to construct the final set of boundary lines (Fig. 4c). Morphological bridging operation works on binary images. It bridges unconnected pixels, that is, sets 0-valued pixels to 1 if they have two nonzero neighbors that are not connected (see Eq. 1).

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \text{ becomes } \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \tag{1}$$

2.3 Adaptive thresholding

Let us now take an adaptive thresholding policy in order to binarize the gray image. This adaptive thresholding is performed with respect to the boundary map (Fig. 4c). For each horizontal line, we run from left to right until we find the first white pixel (p_1) and then, go on while the next white pixel is found (p_k). Similarly, for the same horizontal line, we find other pixels if any exist, i.e., (p_2) and (p_m) and so on.

Let us consider 3×3 neighborhood pixels from the gray image, separately around each of p_1 and p_k. These two neighborhoods indicate abrupt changes in gray values. We next compute the average of these 18 values. Further we divide the pixels inside the horizontal run into two parts by taking the average value as a threshold. This procedure is continued for other pixels (if exist) on the same horizontal line. Further, the same process is performed for all the vertical lines. Finally, these two outcomes are merged to obtain the final binarized image (Fig. 5a).

2.4 Post-processing

After binarizing the image (applying the adaptive thresholding technique), we observe a problem in several cases that different segmented regions (text and non-text components) may become connected (Fig. 5a) to each other by small bridges which have to be removed. So, we apply the following post-processing scheme to separate out such connected regions.

Let us place a 5×5 window around each pixel on a *boundary* line. Within each such window, we compute the average pixel value, where pixels are collected from the gray-level image. Considering this average as a threshold, we divide the pixels inside each window into two parts. Finally, each window is *logically ANDed* with the binarized image. This process separates the pixels in the boundary region into two portions thus adding some pixels to the text regions and some to the non-text regions. This creates a boundary of zeroes over the ones or vice versa and disconnects all the small bridges (Fig. 5b). Here, in Fig. 6 for visual inspection we present outcomes from different steps of our binarization methodology.

Fig. 5 **a** After binarization, text and non-text are connected, **b** text separated from non-text

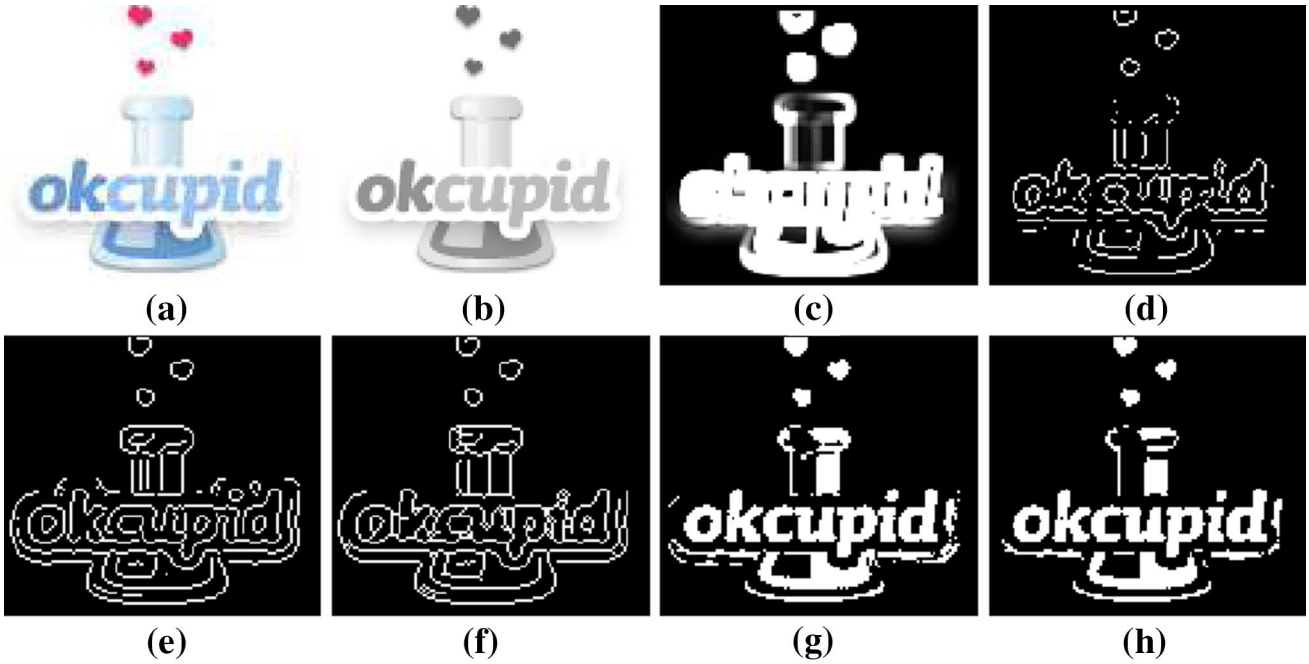


Fig. 6 Steps of the proposed binarization techniques: **a** input image, **b** gray image, **c** variance map (VM), **d** boundary lines (E1), **e** Canny edge map of *Im (E2)*, **f** complete boundary map *(boundary)*, **g** binarized image and **h** result after post-processing

3 Feature extraction from connected components

Image binarization produces a number of connected components (CCs) including possible text components. In order to separate non-text from text components, we extract the following features from each components. These features highlight certain shape-based characteristics of text components.

SR: The connected component size ratio (SR) is used to discard large components. It is defined as:

$$\text{SR} = \frac{\text{area of the connected component}}{\text{area of } P} \tag{2}$$

Here, P is the input scene image. We observe that text components occupy only a small region inside an image. Thus, large components usually contribute to the background and hence are removed.

AR: The aspect ratio $\text{AR} = min\{(height/width), (width/height)\}$ of a text component belongs within a compact range. Non-text components generally have irregular shape; hence, their aspect ratio falls outside the range.

ER: The text-like patterns are usually elongated. The elongatedness ratio (ER) is defined as follows:

$$\text{ER} = \frac{\text{total number of boundary pixels}}{\sqrt{(\text{total number of pixels in the CC})}} \tag{3}$$

OBR: The object-to-background pixels ratio (OBR) is computed by taking the bounding box. Due to the elongated nature of texts, only a few object pixels fall inside the bounding box. On the other hand, elongated non-texts are usually straight lines, and hence contribute enough object pixels.

AXR: Axial ratio (AXR) of any shape which is the ratio of the length (or magnitude) of the two axes to each other—the longer axis divided by the shorter. Here, we calculate the *MajorAxisLength* and *MinorAxisLength* of a connected component. The axial ratio is defined as:

$$\text{AXR} = \frac{\text{Major axis length}}{\text{Minor axis length}} \tag{4}$$

Major axis and minor axis of a connect component: Consider the straight line L passing through the center of gravity of a connected component (CC). Let $a(L)$ be the sum of squares of the perpendicular distances from all the pixels in the CC to L. The major axis of a CC is the line L that minimizes $a(L)$. On the other hand, the minor axis of a CC is the line L that maximizes $a(L)$.

TH: Thickness (TH) defined by Ghoshal et al. [7] of a connected component is calculated as: let h_i and v_i be the horizontal and the vertical run lengths of an object pixel p_i at the ith position of a component CC_j. We next compute the minimum of h_i and v_i and further constitute a set

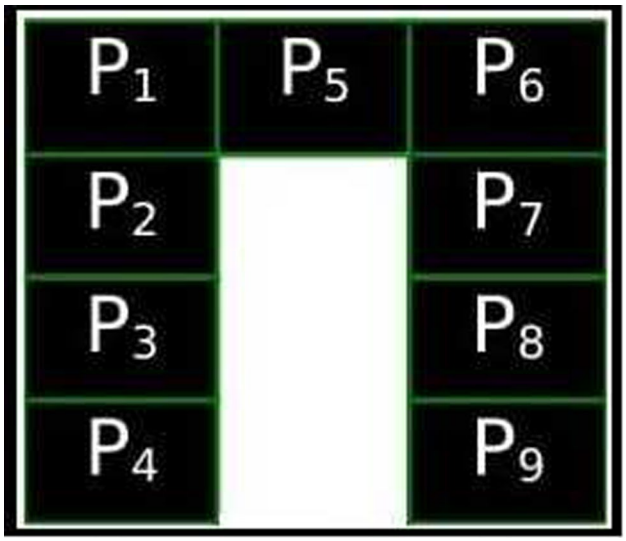


Fig. 7 A connected component containing 9 pixels

Table 1 Values of horizontal run length (h_i), vertical run length (v_i) and minimum (h_i and v_i) for a connected component

Pixel (p_i)	h_i	v_i	Min (h_i, v_i)
p_1	3	4	3
p_2	1	4	1
p_3	1	4	1
p_4	1	4	1
p_5	3	1	1
p_6	3	4	3
p_7	1	4	1
p_8	1	4	1
p_9	1	4	1

$\text{MIN}_j = \{m_i \text{ s.t. } m_i = \min\{h_i, v_i\}, \forall i\}$. Thus MIN_j denotes the set of all the minimum run lengths considering all the pixels of CC_j. The thickness TH_j of the component CC_j is defined to be the maximally frequent element of the set MIN_j. Let us consider an example (Fig. 7). Here, we consider a connected component that contains 9 pixels. Now, for each pixel p_i, i= 1 to 9, the horizontal run length (h_i), the vertical run length (v_i) and the minimum of h_i and v_i are calculated. These values are presented in Table 1. Now, a set $MIN_j = \{3, 1, 1, 1, 1, 3, 1, 1, 1\}$ is constructed. The thickness of this connected component is the most frequent element of the set (MIN_j), i.e., 1.

LR: Length ratio is calculated as:

$$\text{LR} = \frac{\text{length of a component}}{\text{length of the image}} \tag{5}$$

Here, (1) length of a component is the maximum of width and height of the CC and (2) length of the image means maximum of width of the image and height of the image.

WV: The width variation is defined as:

$$\text{WV} = \frac{\text{variance of width}}{\text{mean of width}} \tag{6}$$

where width of an object pixel p_i is the minimum of horizontal and vertical run lengths of the runs passing through p_i.

Combining these features, we construct the feature vector $\Upsilon = \{\text{SR, AR, ER, OBR, AXR, TH, LR, WV}\}$ for a connected component.

A few non-text and text components are presented in Fig. 8a, b, respectively. Sample feature files for text and non-text components are also presented in Tables 2 and 3, respectively.

4 Text identification methodology

Once we define the feature set for connected components, the task is to identify a component as text with respect to the aforementioned features. Given the training sets for text and non-text, this task is a two-class classification problem. However, the Born Digital data set provides ground truth [12] only for the text components. In fact, it is rather unrealistic to assume any prior shape of the non-text components. Thus, let us consider text identification as an one-class classification problem, such problems are also popularly known as anomaly detection [24]. The strategy for text identification is as follows.

Let us consider the training examples. The ground truth text components for these training images are already provided. Then, we obtain a statistical model to describe the corresponding feature distribution. This completes the training phase. Further we subject connected components obtained from a test image, to this trained model, and perform a hypothesis testing to identify text components. The following section elaborates these two steps.

4.1 Gaussian copula-based modeling of text component features

We intend to obtain a parametric statistical distribution that could approximate the distribution of the feature vectors. In this context, the most popular model is the multivariate Gaussian distribution [5]. However, by definition, the margins of a multivariate Gaussian distribution are themselves univariate Gaussian distribution. Note that individual features constitute the margins of the distribution. Now, it is

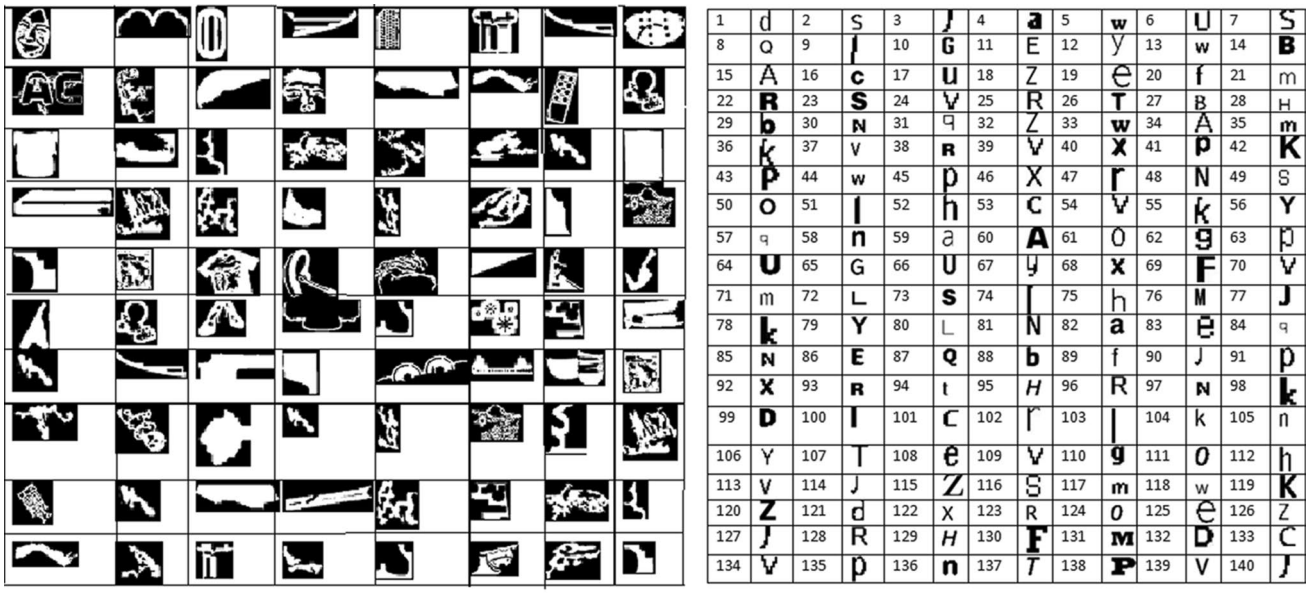


Fig. 8 **a** A few sample non-text components and **b** a few text components. Here, a black pixel implies an object pixel

Table 2 Feature file for a few text connected component

CC	SR	AR	ER	OBR	AXR	TH	LR	OBR
cc_1	8.73E−03	1.00E+00	4.00E+00	1.91E+00	1.10E+00	12	1.08E−01	1.42E+00
cc_2	1.09E−02	1.46E+00	6.86E+00	1.29E+00	1.50E+00	9	1.58E−01	5.91E−01
cc_3	6.45E−03	1.18E+00	5.63E+00	1.34E+00	1.45E+00	10	1.08E−01	8.69E−01
cc_4	9.59E−03	8.67E−01	6.52E+00	1.67E+00	1.13E+00	13	1.25E−01	1.00E+00
cc_5	1.16E−02	1.36E+00	4.96E+00	1.25E+00	1.59E+00	14	1.58E−01	1.79E+00
cc_6	4.87E−03	1.86E+00	4.56E+00	2.14E+00	2.64E+00	6	1.08E−01	2.78E−01
cc_7	1.34E−03	8.00E−01	2.97E+00	5.67E+00	1.26E+00	4	4.17E−02	1.49E−01
cc_8	1.20E−02	1.27E+00	5.11E+00	1.16E+00	1.48E+00	14	1.58E−01	1.57E+00
cc_9	5.08E−04	1.60E+00	6.10E+00	8.18E−01	1.87E+00	5	4.00E−02	5.00E−01
cc_{10}	5.65E−04	5.00E+00	4.47E+00	2.00E+04	5.00E+00	2	5.00E−02	1.10E+00

Table 3 Feature file for a few non-text connected component

CC	SR	AR	ER	OBR	AXR	TH	LR	OBR
cc_1	3.99E−02	3.93E−01	1.82E+01	1.16E−01	3.75E+00	7	9.29E−01	5.08E+00
cc_2	1.27E−03	3.70E+00	4.81E+00	9.88E+00	3.98E+00	1	6.99E−02	1.20E−01
cc_3	2.06E−04	1.48E−01	6.59E+00	2.27E+00	8.44E+00	4	3.71E−02	1.37E−01
cc_4	7.28E−04	3.25E+00	4.46E+00	7.00E+00	3.39E+00	8	5.20E−02	1.74E−01
cc_5	5.03E−03	2.89E−01	9.33E+00	9.18E−01	3.17E+00	2	1.35E−01	2.87E+00
cc_6	5.03E−05	8.75E−01	4.19E+00	1.24E+00	1.91E+00	5	6.49E−03	3.63E−01
cc_7	1.08E−04	2.00E+00	5.42E+00	7.30E−01	3.58E+00	6	1.60E−02	5.16E−01
cc_8	2.08E−04	1.50E+00	3.60E+00	3.55E+00	1.40E+00	1	1.33E−02	4.69E−01
cc_9	3.44E−04	8.40E−01	5.60E+00	6.77E−01	2.12E+00	2	2.03E−02	1.96E+00
cc_{10}	3.74E−04	7.50E−01	3.70E+00	2.25E+00	1.50E+00	1	2.25E−02	5.94E−01

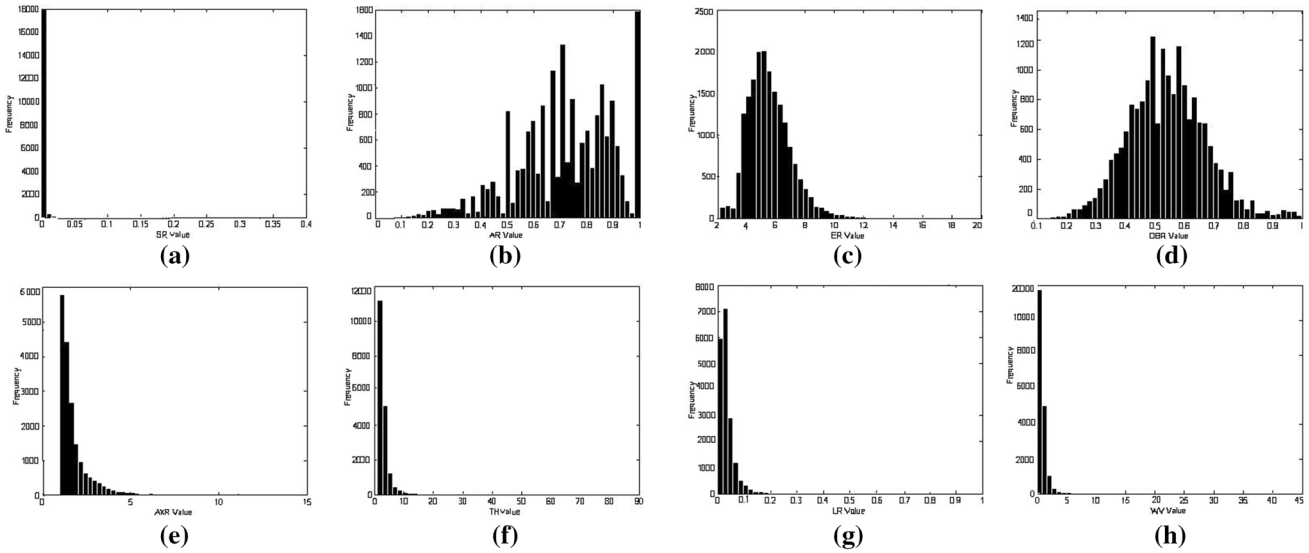


Fig. 9 Histogram of each text feature: **a** SR, **b** AR, **c** ER, **d** OBR, **e** AXR, **f** TH, **g** LR and **h** WV

clear from Fig. 9 that individual features may not necessarily follow Gaussian distributions. Hence, a non-Gaussian multivariate distribution may be a suitable choice to approximate the distribution of the present features. There is, however, another downside of using such a distribution. A multivariate distribution is usually composed of the margins belonging to the same family. In real situations, it is possible that the individual features may follow different families of distributions, independently. In fact from Fig. 9, clearly we cannot approximate individual features by the same family of distributions. A suitable multivariate distribution should allow the marginal distributions to follow different families of distributions. The known families of multivariate distributions (for example, Gaussian or Dirichlet distributions) do not have this characteristic. However, there is an alternative construction of a multivariate distribution. In multivariate statistics, the copula approach is often taken to model the dependence between two or more random variables. Concerning the bivariate case, the copula approach to dependence modeling was first stated in a theorem due to Sklar [21].

Theorem 1 (Sklar) Let *F be a joint distribution function with marginal distributions* F_1*and* F_2*. Then there exists a copula C such that for all* $x, y \in [-\infty, \infty]$,

$$F(x, y) = C(F_1(x), F_2(y)). \tag{If7}$$

F_1 *and* F_2 *are continuous, then C is unique; otherwise, C is uniquely determined on* $\text{Ran}(F_1) \times \text{Ran}(F_2)$*. Conversely, if C is a copula and* F_1 *and* F_2 *are distribution functions, then the function* $F(., .)$ *defined by Eq. 7 is a joint distribution function with margins* F_1 *and* F_2*.*

Here, $C(., .)$ is a mapping $[0, 1] \times [0, 1] \rightarrow [0, 1]$, termed as copula in the sense that it couples the random variables X and Y. Since $F(., .)$ is the distribution function, the joint density function $f(., .)$ can be obtained by differentiating $F(., .)$ with respect to X and Y. It is given by Eq. 8.

$$f(x, y) = c(F_1(x), F_2(y))f_1(x)f_2(y), \tag{8}$$

where, $f_1(.)$ and $f_2(.)$ are the density functions for X and Y, respectively. The function $c(., .)$ is the density of the copula $C(., .)$. The advantage of copula is that using the knowledge of the margins only, one can construct the joint distribution, accommodating, however, complex form of dependence structure, on the basis of different types of copulas. In addition, the margins can be freely adapted from different families of univariate distributions. The Sklar's theorem has an analogous multivariate version for continuous margins. On the basis of this theorem, we can construct the multivariate distribution incorporating different margin families.

Let us start by specifying the marginal distributions. The margins correspond to individual features. We here fit a series of parametric distributions on each feature. The best fitted distribution that maximizes the likelihood is selected as the marginal distribution for that feature. Here, we use an archive of distributions consisting of Gaussian, Gamma and log-normal distributions. Each of the different features has its own parametric distribution obtained by the above procedure. The individual features and their corresponding distributions are presented in Table 4. Note that most of the time we obtain the log-normal as a suitable distribution. The log-normal parameters are α and β that control the location and the shape of the distribution, respectively. In two occasions, we obtain Gaussian distribution as an appropriate

Table 4 Parametric distribution correspond to individual features

Feature	SR	AR	ER	OBR
Distribution	Log-normal	Gaussian	Log-normal	Gaussian
	$\alpha = -7.4877$	$\mu = 0.7155$	$\alpha = 1.6865$	$\mu = 0.5401$
	$\beta = 1.4379$	$\sigma^2 = 0.0335$	$\beta = 0.0670$	$\sigma^2 = 0.0176$
Feature	AxR	TH	LR	WV
Distribution	Log-normal	Log-normal	Log-normal	Log-normal
	$\alpha = 0.4862$	$\alpha = 0.9311$	$\alpha = -3.5520$	$\alpha = -0.5572$
	$\beta = 0.1583$	$\beta = 0.2358$	$\beta = 0.4439$	$\beta = 0.7811$

distribution. In Table 4, μ and σ^2 represent the mean and the variance of the corresponding Gaussian distribution.

Now, in order to combine these feature distributions we use the multivariate Gaussian Copula [21]. Let $\mathbf{u} = \{u_1, \ldots, u_d\}$ be a d-dimensional (here $d = 8$) vector consisting of the distribution functions of all the margins. Here u_i denotes the distribution function of the ith feature (i.e., ith element of $\boldsymbol{\Upsilon}$). Then $\mathbf{u} \in [0, 1]^d$ according to the probability integral transform. The d-dimensional Gaussian copula is defined over $[0, 1]^d$. It is constructed from a multivariate Gaussian distribution over $\Re^d$ by using the probability integral transform. Given the correlation matrix $\Sigma \in \Re^{d \times d}$, the density of a Gaussian copula can be written as:

$$c_\Sigma(\mathbf{u}) = \frac{1}{\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2} \begin{pmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_d) \end{pmatrix}^{\mathrm{T}} \left(\Sigma^{-1} - \mathbf{I} \right) \begin{pmatrix} \Phi^{-1}(u_1) \\ \vdots \\ \Phi^{-1}(u_d) \end{pmatrix} \right) \tag{9}$$

where Φ^{-1} is the inverse cumulative distribution function of a standard Gaussian distribution. $\mathbf{I}$ is the identity matrix. The Gaussian copula parameter, i.e., the correlation matrix Σ is estimated on the basis of training samples. In Table 5, we present the correlation matrix. Once having the correlation matrix in hand, we need to decide whether to consider the features independently (i.e., without being correlated). If so, we may use an independent copula to obtain a suitable model for the features. Then, by using a Naive–Bayes classifier, we may be able to separate text and non-text components. However, in Table 5, we can see that some pair of features are highly correlated. Therefore, the correlation information should be incorporated while modeling these features. A multivariate distribution is desirable to capture this correlation information, and hence, we prefer to use the multivariate Gaussian copula.

Once the density function of the Gaussian copula is available, we can construct the joint distribution by coupling all the eight margins through Eq. 9. We have the following multivariate distribution for a feature vector $\boldsymbol{\Upsilon}$.

$$f(\boldsymbol{\Upsilon}) = c_\Sigma(u_1, \ldots, u_8) \prod_{i=1}^{8} f_i(\Upsilon_i). \tag{10}$$

Here, $f_i(\Upsilon_i)$ is the density of the ith feature, i.e., Υ_i. Note that all $f_i (i = 1, \ldots, 8)$ may not belong to the same family (see Table 4).

4.2 Hypothesis testing for identifying text components

As we have pointed out earlier, we do not have training samples of the non-text components. Hence, we consider text identification as a problem anomaly detection. Let $\boldsymbol{\Upsilon}^{(\text{test})}$ be the feature vector corresponding to a connected component of a test sample. Since we have the learned model $f(.)$, the task is to subject $\boldsymbol{\Upsilon}^{(\text{test})}$ to $f(.)$ and test the hypothesis that $\boldsymbol{\Upsilon}^{(\text{test})}$ comes from $f(.)$. For this task, we take the following

Table 5 Correlation matrices of the Gaussian copula corresponding to training samples

1	0.09	0.27	− 0.04	− 0.13	0.57	0.87	0.35
0.09	1	− 0.09	− 0.05	− 0.84	0.11	− 0.14	0.30
0.27	− 0.09	1	− 0.46	0.06	− 0.04	0.34	0.18
− 0.04	− 0.05	− 0.46	1	− 0.08	0.33	− 0.20	− 0.05
− 0.13	− 0.84	0.06	− 0.08	1	− 0.09	0.11	− 0.32
0.57	0.11	− 0.04	0.33	− 0.09	1	0.38	0.42
0.87	− 0.14	0.34	− 0.20	0.11	0.38	1	0.21
0.35	0.30	0.18	− 0.05	− 0.32	0.42	0.21	1

strategy. We compute the log-likelihood statistic for the feature vectors $\Upsilon^{(\text{train})}$ belonging to the training text components. Now, we know the distribution of the log-likelihood statistic when the data actually comes from Eq. 10. If the log-likelihood statistic of $\Upsilon^{(\text{test})}$ lies at the tails of this distribution, we could reject the null hypothesis and hence that component is a non-text.

The distribution of the log-likelihood statistic is shown in Fig. 10. We use a nonparametric fitting to approximate this distribution. Let us denote it by g. This distribution is obtained empirically. Then the testing is designed as follows. The confidence interval considered here is 95%. Thus two values a and b are determined such that (1) $\int_a^b g(x)\mathrm{d}x = 0.95$ and (2) $(b-a)$ is minimized. Now, for any arbitrary CC we find its log-likelihood value, say, w. If w lies between a and b, then we accept the hypothesis that the CC is text. If $w < a$ or $w > b$, then the CC is non-text.

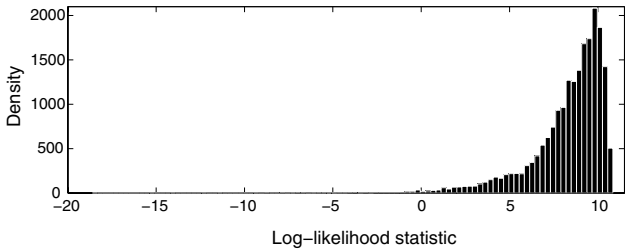


Fig. 10 Distribution of log-likelihood statistic for $\Upsilon^{(\text{train})}$

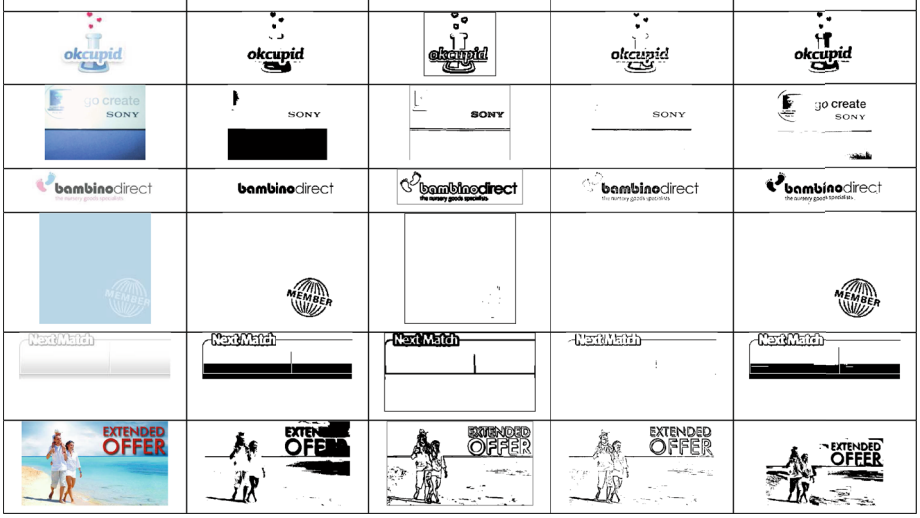
5 Results and discussion

The experimental results are obtained on ICDAR 2011 Born Digital Data set [13]. This data set contains 420 training images and for 102 test images. Born-digital images are inherently of low resolution in order to be transmitted online efficiently. Automatically extracting text from born-digital images is therefore an interesting research topic. Our experiments are divided into two parts based on the aim of this study. First, the binarization results are presented in Sect. 5.1. Afterward, we present and evaluate text identification results, in Sect. 5.2.

5.1 Results and evaluation of binarization

Let us first pictorially observe some binarization results form ICDAR 2011 Born Digital Data set. For the sake of comparison, we apply the well-known Otsu, Niblack and Sauvola binarization algorithms. Some example results are presented in Table 6. Clearly, our method outperforms the other three methods, for these images. For an in-depth study, consider the input image at the third row of Table 6. This image contains the text: "go create SONY." After binarization, we observe that all the methods (Otsu, Niblack and Sauvola) except our proposed binarization method fail to preserve all the text components. Next consider the input image at the forth row, which contains "bambinodirect" and under that "the nursery goods specialists" in very small fonts. We notice that Otsu's method fails to preserve all the text components in small fonts. Comparing to Niblack and Sauvola, our method pictorially looks better in terms of text

Table 6 Result of various binarization techniques

Input image	Otsu	Niblack	Sauvola	Proposed method
				

components. Finally, consider the input image at the fifth row, which contains the text "MEMBER." Here, we observe that Niblack and Sauvola methods fail to preserve text components. On the other hand, both Otsu's and our method preserve the text components equally well when judged from a viewer's end. The objective of our proposed binarization methodology is to binarize the text-based scene image efficiently. Thus, to compare our binarization technique we consider only the text region of the binarized image with the ground truth to compute the precision, recall and F-measure. The performance evaluation is based on a well-established technique by Dance et al. [4] that counts true positive (TP), false positive (FP) and false negative (FN) pixels in order to calculate recall and precision metrics.

- A pixel is classified as TP if it is ON in both ground truth (GT) and binarization result images.
- A pixel is classified as FP if it is ON only in the binarization result image.
- A pixel is classified as FN if it is ON only in the GT image.

The recall metric shows the ratio of the number of pixels, which our method truly classified as foreground, to the number of all pixels classified as foreground from the ground truth image. Precision metric is the ratio of the number of pixels, which our method truly classifies as foreground, to the number of all pixels which are classified as foreground. Setting C_{TP} as the number of TP pixels, C_{FP} as the number of FP pixels and C_{FN} as the number of FN pixels, recall (RC) and precision (PR) metrics are given as follows:

$$\mathrm{RC} = \frac{C_{\mathrm{TP}}}{(C_{\mathrm{FN}} + C_{\mathrm{TP}})} \tag{11}$$

$$\mathrm{PR} = \frac{C_{\mathrm{TP}}}{(C_{\mathrm{FP}} + C_{\mathrm{TP}})} \tag{12}$$

recall and precision metrics have values between zero and one. As these metrics approach one, the results get better.

The overall metric that is used for evaluation is the F-measure (FM) which is calculated as follows:

$$\mathrm{FM} = (2 \times \mathrm{RC} \times \mathrm{PR}/(\mathrm{RC} + \mathrm{PR})) \times 100\% \tag{13}$$

We have compared the proposed binarization method with a few well-known image binarization techniques proposed by Otsu, Niblack and Sauvola in terms of recall, precision and F-measure (FM) on the dataset of ICDAR 2011 Born Digital Dataset. As can be seen from the results shown in Table. 7, our proposed binarization method significantly outperforms the existing methods in terms of recall, precision and F-measure.

Table 7 Average value of F-measure, recall and precision for each binarization technique

	Otsu	Niblack	Sauvola	Proposed
Recall	0.92	0.87	0.91	**0.93**
Precision	0.67	0.36	0.14	**0.87**
FM	73.13	38.17	20.4	**84.37**

Bold values signify the best result

5.2 Text identification results

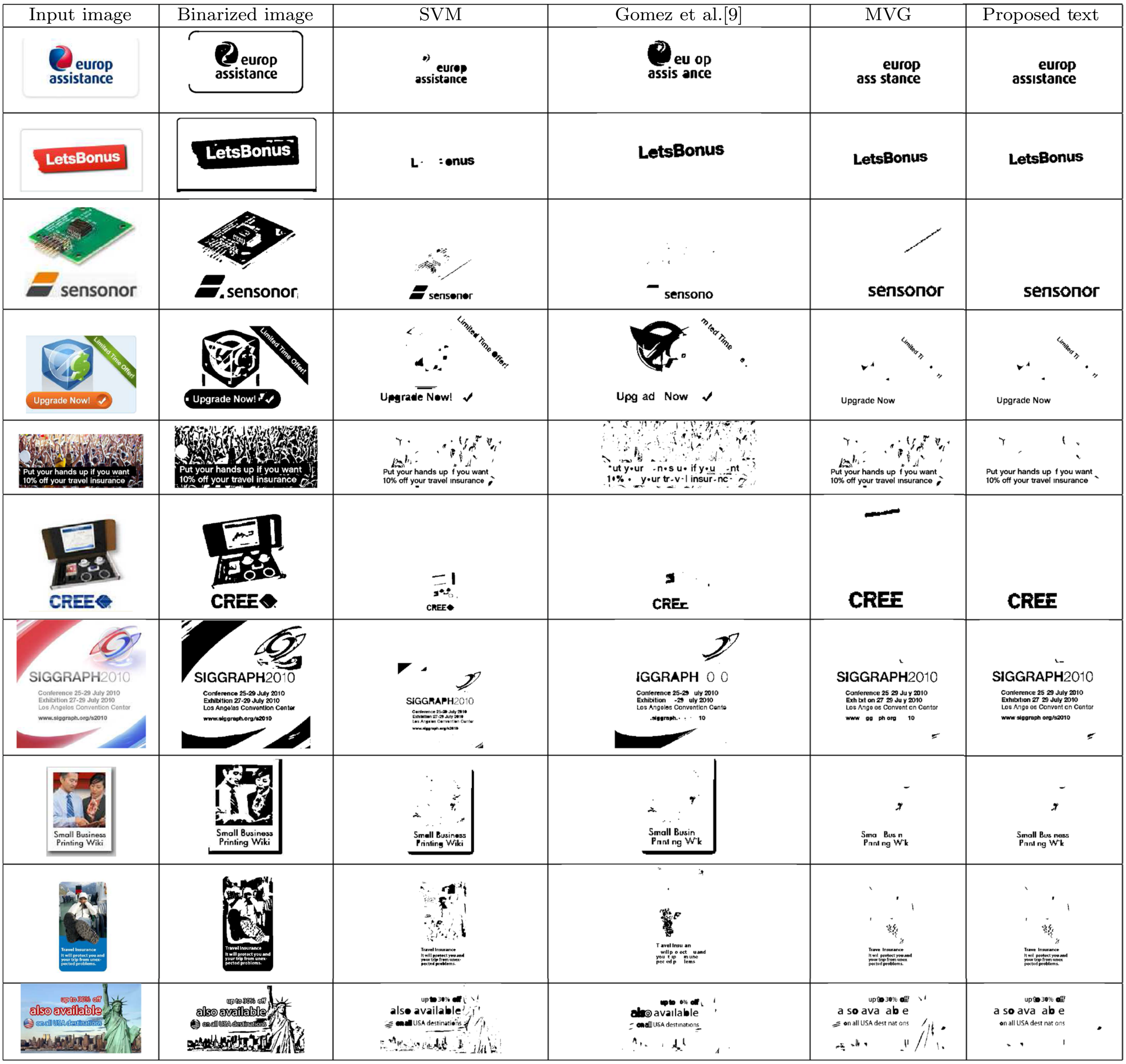
After obtaining the binarized images, we may now proceed toward the identification of possible text components. For this purpose, the multivariate Gaussian copula parameters are presented in Table 5. The issue arises here is how well the multivariate Gaussian copula-based method perform compare to other copula-free methods. As an example, the multivariate Gaussian (MVG) distribution can be considered. Unlike Gaussian copula, multivariate Gaussian distribution does not allow margins from different family and hence provides a less flexible model than the former. In this section, an experimental comparison between these two mentioned methods is provided.

Let us consider a few well-known simple classifiers i.e., SVM and Naive–Bayes NN [9] classifiers. We apply these two classifiers based on our features. The results are presented in Table 8. Visually our proposed method performs well with respect to these classifiers. Further, the recall, precision and FM values are calculated (Table 11), and our scheme significantly outperforms than these two methods.

Since binarization is one of our prime aim, let us first addresses the comparison among various binarization methods in the light of text separation. Table 8 presents some text separation results obtained after applying our proposed binarization. In contrast, Table 9 presents results on the same images, now binarized by different algorithms. Visually, it is clear that our approach outperforms all the three other approaches. In fact, apart from Sauvola's binarization, none of the other two can produce result visually close to our method. Moreover, both Otsu's and Niblack's methods perform very poorly in context of text extraction.

A visual comparison is heavily user dependent. A more robust comparison analysis can be performed by means of evaluation criteria. In this regards, Clavelli et al. [3] proposed a number of measures to assess the segmentation quality of each text-parts defined in the ground truth. According to Clavelli et al., the text-parts are classified as *Well segmented*, *Merged* and *Lost*. A text-part is Well segmented if it comes as a whole, i.e., it overlaps with a single text-part skeleton in the ground truth. Merged text-parts, on the other hand, contain more than one skeleton. A text-part is Lost, if it is not Well segmented or Merged and it is not overlapped

Table 8 Text identification results of multivariate Gaussian (MVG) distribution and proposed techniques



even with a fragment of one or more text-part skeleton. In addition to recall and precision, these measures are also included in our study. Now, in context of text extraction, let us first compare different binarization methods by means of these quality measures. Table 10 presents the outcomes. It is clearly observed that our proposed binarization method yields the best text extraction results compare to the other three methods, whereas Niblack method produces the worst. Our proposed method tends to produce *Merged* text-parts more than Otsu's method. However, the *Lost* text-parts for Otsu's method are much higher than our method. The closest competitor of our method is the Sauvola binarization. However, even if the later is close to our method with respect to *Merged* text-parts, it tends to retrieve less number of text-parts since the *Lost* measure is higher than our method.

The next evaluation of our method is performed by comparing with other known methods including the multivariate Gaussian distribution-based method we discussed earlier. The ICDAR 2011 and 2013 Robust Reading Competition [13, 14] presented evaluation results of a number of methods

Table 9 Text identification results from Otsu's, Niblack's and Sauvola's binarization techniques

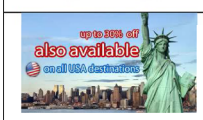
Input image	Text from Otsu	Text from Niblack	Text from Sauvola
			
			
			
			
			
			
			
			
			
			

Table 10 Average value of recall, precision and F-measure (FM) for each text identification technique

	Otsu	Niblack	Sauvola	Proposed
Recall	31.47	20.84	57.3	**76.57**
Precision	42.06	27.65	53.83	**65.03**
FM	36.00	23.76	55.51	**70.30**
Well seg	26.96	9.33	49	**69.72**
Merged	**7.50**	10.90	8.72	8.39
Lost	65.55	79.77	42.20	**21.90**

Bold values signify the best result

from different participants. In Table 11, some of these methods are compared with our proposed method. Our method has achieved a recall of 85.73, which is higher than the current performance. Our method outperforms other six methods considering the recall, Lost and Well segment.

Finally, we evaluate our scheme with respect to ICDAR 2015 Robust Reading Competition [11]. During this evaluation, we compare our method with several other methods reported in the competition. Table 12 presents some of these techniques compared with our proposed scheme. Our

Table 11 Performance comparison of different algorithms for text identification

Method	Well seg.	Merged	Lost	RC	PR	FM
Our proposed	74.34	11.23	**14.43**	**86.73**	**90.33**	**88.49**
Our MVG	72.64	10.49	16.88	84.07	78.68	81.28
Yin et al. [33]	-	-	-	84.21	93.49	88.61
Kumer et al. [15]	64.14	15.69	20.15	80.62	72.06	76.10
Adaptive EdgeDetection	66.55	9.24	24.20	78.24	70.97	74.43
Textorter	58.12	9.50	32.37	65.23	63.63	64.42
SASA	41.58	10.97	47.43	71.68	55.44	62.52
Naive–Bayes NN [9]	37.18	11.37	31.35	63.49	52.63	52.85
SVM	49.16	13.22	41.19	60.80	50.89	51.07

Bold values signify the best result

Table 12 A comparative study on our proposed text identification schemes with ICDAR 2015 competition on robust reading

Text identification methods	Strong			Weak			Generic		
	P (%)	*R* (%)	*F*	*P* (%)	*R* (%)	*F*	*P* (%)	*R*(%)	*F*
Proposed	80.63	75.31	77.89	80.16	74.92	77.45	80.09	75.11	77.52
Stradvision-2	83.93	73.02	78.1	77.61	70.86	74.08	57.35	56.68	57.01
Deep2Text-II [32, 33]	80.97	73.37	76.98	80.97	73.37	76.98	80.97	73.37	76.98
Stradvision-1	84.72	70.17	76.76	78.9	67.87	72.97	58.2	54.31	56.19
Deep2Text-I [32, 33]	83.46	61.4	70.75	83.46	61.4	70.75	83.46	61.4	70.75
PAL [19, 30]	65.22	61.54	63.33	–	–	–	–	–	–
NJU	60.12	41.31	48.97	–	–	–	–	–	–
Baseline [Gomez14]	46.48	37.13	41.28	47.2	32.82	38.72	30.29	24.2	26.9
TextCatcher-2	32.11	40.26	35.73	–	–	–	32.11	40.26	35.73
TextCatcher-1	11.58	5.01	6.99	–	–	–	11.58	5.01	6.99

scheme has achieved a recall of 75.31 which is higher than the other methods. The other measures are also comparable.

6 Summary and future scope

This article provides a new, effective image binarization methodology and its application in text identification from scene images. We propose several key improvements over traditional methods. The Binarization technique detects text boundary by applying edge detection on the variance map of a gray image. The proposed binarization method is not very sensitive to image color, text font, skew and perspective variation. It is effective in terms of low resolution, low contrast, non-uniform illumination and noisy text-based scene images. This method can be extended to document image also. Text identification is based on this image binarization followed by extraction of several shape-based features that lead toward identification of text components. Here, we consider text component identification as an one-class problem. The ground truth text components are used to obtain a statistical model to describe the corresponding feature distribution. Since, all the features may not follow a single family of distribution we construct the joint distribution by using multivariate Gaussian copula which allows different margins. Finally, a test connected component of an unknown class is subjected to the trained statistical model and by performing a hypothesis test we successfully identify a possible text component. By integrating the above methodologies, we built an effective, robust text identification system that exhibited superior performance over state-of-the-art algorithms on ICDAR 2011 Born Digital Data set. Our later work will explore the use of other copula models to improve our performance.

References

1. Bhattacharya U, Parui SK, Mondal S (2009) Devanagari and bangla text extraction from natural scene images. In: Proc. of the int. conf. on document analysis and recognition, pp 171–175
2. Canny J (1986) A computational approach to edge detection. IEEE Trans Pattern Anal Mach Intell 8(6):679–698
3. Clavelli A, Karatzas D, Lladós J (2010) A framework for the assessment of text extraction algorithms on complex colour images. In: Proceedings of the 9th IAPR international workshop on document analysis systems, DAS '10. ACM, pp 19–26

4. Dance CR, Seegar M (1999) On the evaluation of document analysis components by recall, precision, and accuracy. In: Proc. of the fifth int. conf. on document analysis and recognition, pp 713–716
5. Figueiredo M, Jain AK (2002) Unsupervised learning of finite mixture models. IEEE Trans Pattern Anal Mach Intell 24(3):381–396
6. Gatos B, Pratikakis I, Perantonis SJ (2008) Improved document image binarization by using a combination of multiple binarization techniques and adapted edge information. In: International conference on pattern recognition, ICPR '08. IEEE, pp 1–4
7. Ghoshal R, Roy A, Bhowmik TK, Parui SK (2011) Decision tree based recognition of bangla text from outdoor scene images. In: Proc. of the 18th international conference on neural information processing, pp 538–546
8. Gllavata J, Ewerth R, Freisleben B (2004) Text detection in images based on unsupervised classification of high frequency wavelet coefficients. In: Proceedings of the international conference on pattern recognition, vol. 1, pp 425–428
9. Gomez L, Karatzas D (2016) A fine-grained approach to scene text script identification. In: Proceedings of the 12th IAPR international workshop on document analysis systems, DAS '16, pp 192–197
10. Jung K, Kim IK, Kurata T, Kourogi M, Han HJ (2002) Text scanner with text detection technology on image sequences. In: Proceedings of the international conference on pattern recognition, vol. 3, pp 473–476
11. Karatzas D, Gomez-Bigorda L, Nicolaou A, Ghosh S, Bagdanov A, Iwamura M, Matas J, Neumann L, Chandrasekhar V (2015) Icdar 2015 robust reading competition-challenge 1: reading text in born-digital images (web and email). In: Proceedings of the 13th international conference of document analysis and recognition, ICDAR '15. IEEE, pp 1156–1160
12. Karatzas D, Robles S, Gomez L (2014) An on-line platform for ground truthing and performance evaluation of text extraction systems. In: Proceedings of the 11th IAPR international workshop on document analysis systems, pp 242–246
13. Karatzas D, Robles Mestre S, Mas J, Nourbakhsh F, Roy PP (2011) Icdar 2011 robust reading competition-challenge 1: reading text in born-digital images (web and email). In: In Proc. 11th international conference of document analysis and recognition, ICDAR '11. IEEE, pp 1485–1490
14. Karatzas D, Shafait F, Uchida S, Iwamura M, Bigorda, LGi, Mestre SR, Mas J, Mota DF, Almazn JA, Heras LPdl (2013) Icdar 2013 robust reading competition. In: 12th international conference on document analysis and recognition, ICDAR '13, pp 1484–1493
15. Kumar D, Ramakrishnan AG (2012) Octymist:otsu-canny minimal spanning tree for born-digital images. In: Proceedings of the 10th IAPR international workshop on document analysis systems, DAS '12, pp 389–393
16. Li H, Doermann D (1998) Automatic identification of text in digital video key frames. In: Fourteenth international conference on pattern recognition, ICPR '98. IEEE pp 129–132
17. Liang J, Doermann D, Li H (2005) Camera based analysis of text and documents : a survey. Int J Doc Anal Recognit 7:84–104
18. Lienhart R, Stuber F (1996) Automatic text recognition in digital videos. In: Image and video processing IV, proc. SPIE 2666, pp 180–188
19. Liu CL, Koga M, Fujisawa H (2002) Lexicon-driven segmentation and recognition of handwritten character strings for japanese address reading. IEEE Trans Pattern Anal Mach Intell 24(11):1425–1437
20. Lu S, Su B, Tan CL (2010) Document image binarization using background estimation and stroke edge. Int J Doc Anal Recogn 13(4):303–314
21. Nelsen RB (2006) An introduction to copulas. Springer, Berlin
22. Niblack W (1986) An introduction to digital image processing. Prentice Hall, Englewood Cliffs
23. Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):377–393
24. Roy A, Pal A, Garain U (2017) Jclmm: a finite mixture model for clustering of circular-linear data and its application to psoriatic plaque segmentation. Pattern Recognit 66:160–173
25. Sauvola J, Pietikinen M (2000) Adaptive document image binarization. Pattern Recognit 2:225–236
26. Shi B, Yao C, Zhang C, Guo X, Huang F, Bai X (2015) Automatic script identification in the wild. In: 13th international conference on document analysis and recognition, ICDAR '15, pp 531 – 535
27. Shivakumara P, Phan TQ, Tan CL (2011) A laplacian approach to multi-oriented text detection in video. IEEE Trans Pattern Anal Mach Intell 33(2):412–419
28. Sobottka K, Bunke H, Kronenberg H (1999) Identification of text on colored book and journal covers. In: Proceedings of the international conference on document analysis and recognition, pp 57–63
29. Tsai C, Lee H (2002) Binarization of color document images via luminance and saturation color features. IEEE Trans Image Process 11(4):434–451
30. Wang QF, Yin F, Liu CL (2012) Handwritten chinese text recognition by integrating multiple contexts. IEEE Trans Pattern Anal Mach Intell 34(8):1469–1481
31. Wu V, Manmatha R, Riseman EM (1999) Textfinder: an automatic system to detect and recognize text in images. IEEE Trans Pattern Anal Mach Intell 21(11):1224–1229
32. Yin XC, Pei WY, Zhang J, Hao HW (2015) Multi-orientation scene text detection with adaptive clustering. IEEE Trans Pattern Anal Mach Intell 37(9):1930–1937
33. Yin XC, Yin X, Huang K, Hao HW (2014) Robust text detection in natural scene images. IEEE Trans Pattern Anal Mach Intell 36(5):970–983