



Text Separation from Digital Images: a Pair-Copula Based Approach and Performance Analysis

Anandarup Roy¹ and Oendriila Samanta²

¹Dept. of Computer Science, Sarojini Naidu College for Women, Kolkata-700108, India
¹roy.anandarup@sncwgs.ac.in

²Dept. of Computer Science and Engineering, Academy of Technology, Adisaptagram, Hooghly-712121, India

²oendriila.samanta@aot.edu.in

Abstract Automatic separation of text from digital images holds significant importance in various domains, including document processing and content-based image retrieval. This paper presents a statistical model-based approach for automatic text component extraction from digital images. The methodology comprises two primary tasks. The first step involves color image segmentation by means of a mixture model. This process helps to identify distinct components within the image, where some of the components contain text. In the second step, the task is to separate text components from the non-text components. This task requires a learned model for text features. In this context, we utilize ground truth text components provided by the “Born-Digital Images” dataset. From these text components, we extract text-representing features. Later, a D-vine multivariate distribution is fitted to these features, which serves as a model for text features. This trained model is used to discriminate between text and non-text components obtained after segmentation. For this purpose, a statistical hypothesis testing method is employed on the log-likelihood statistic. Experimental evaluations on the ICDAR 2011 “Born-Digital Images” dataset demonstrate the efficacy of the proposed method. Specifically, when coupled with D-vine Mixture Model (DVMM) segmentation, the proposed D-vine feature model achieves a Recall of 82.25% and Precision of 66.39%. The experimental performance of the D-vine-based model is compared to the multivariate Gaussian copula-based model, and the former generally outperforms the latter in terms of recall percentages. The novelty of this research lies in the utilization of D-vine modeling. A D-vine model is capable of capturing various feature distributions and associations, significantly enhancing the approximation of the joint distribution of features. This, in turn, boosts the method’s ability to discriminate between text and non-text features effectively.

Keywords: Text extraction from image, D-vine copula, Pair-copula construction

1 Introduction

The automatic separation of textual elements from digital images bears significant implications for document processing, content-based image retrieval, robotics, and intelligent transportation systems. With the increasing popularity of diverse image acquisition devices like digital cameras, mobile phones, and handheld devices, digital images have become readily accessible. However, the online transmission often necessitates their downscaling, posing a challenge for text extraction from such images. Extensive research has been conducted on text segmentation in recent years. Initial studies by Wu et al. [30] employed

a local threshold technique to partition text from grayscale image blocks containing textual content. Tsai and Lee [28] proposed a threshold-based approach utilizing intensity and saturation attributes for text segmentation in color document images. Color clustering has also been employed for text identification. Pioneering works by Lienhart and Stuber [18] and Sobottka et al. [26] employed analogous methodologies. In recent years, Jung et al. [16] harnessed a multi-layer perceptron classifier to distinguish text from non-text pixels. More recent advancements by Bhattacharya et al. [5] and Banik et al. [2] introduced methodologies based on the analysis of connected components, for extracting Devanagari and Bangla texts from camera-captured scene images. Anthimopoulos et al. [1] reported a study on the detection of scene text in both images and video frames. This investigation has used a Random Forest classifier to which a set of feature vectors generated using the local binary pattern were fed. In another recent study [29], convolutional neural network (CNN) based end-to-end, lexicon-driven method has been proposed. Yin et al. [32] proposed a method for multi-orientated scene text detection, where text candidates have been constructed through clustering of characters based on the adaptive hierarchical clustering technique.

Recent literature indicates a shift towards more semantically aware and efficient deep learning architectures for text extraction. Addressing the need for user-guided extraction, Du et al. [8] introduced instruction-guided scene text recognition (IGTR) paradigm. Their framework integrates a text encoder that processes instructions and a visual-text fusion module that combines image features with these instructions. This approach enables the model to interpret text more accurately and flexibly. In the domain of accurate localization, Su et al. [27] proposed ERRNet which reformulates text detection as a component tracking problem rather than simple pixel segmentation. This approach effectively eliminates the computational bottleneck of post-processing while improving accuracy on arbitrary-shaped text. Finally, focusing on computational efficiency, Xu et al. [31] presented OTE. They demonstrated that a single global visual token is sufficient to characterize an entire text image. OTE significantly reduces the computational overhead while maintaining high accuracy. This work introduces a new direction for lightweight text extraction from natural images.

In this article, we consider clustering as an important counterpart in our text separation scheme. Assuming color homogeneity within text components, we applied clustering to segment an image into several components. These components are further analyzed to identify possible text components. A number of recent research articles are available following a similar text separation scheme. For example, Roy et al. [20] applied a mixture model with von-Mises and Gaussian marginal distributions, for image segmentation. Further, text components were identified using a simple threshold on elongatedness ratio feature. Ghoshal et al. [13] followed a similar scheme and applied a semi-wrapped Gaussian mixture model [22, 24] for segmentation purposes. In order to characterize a text component, they considered five features. Later, the joint distribution of these features was explained using a multivariate Gaussian Copula [19]. Another recent paper by Ghoshal et al. [12] proposed a binarization method to extract components from an image. However, identification of text components followed a similar scheme as was in [13].

The above-discussed studies indicate that identifying text components from a set of components produced by image segmentation, is a crucial task. Text components possess certain characteristics (features) as pointed out by previous studies [13, 12]. However, obtaining a statistical model (i.e. joint distribution) with these features is not trivial and therefore opens an avenue of research. As pointed out by Ghoshal et al. [12], the individual features may follow different families of distributions, independently. A multivariate distribution (e.g. multivariate Gaussian) is unable to combine margins of different families. Ghoshal et al. [12] addressed this problem using a multivariate Gaussian Copula. However, we observe that the structure (type) of correlation between a pair of features may be different from another pair. Multivariate Gaussian Copula permits only one (linear) type of correlation between each pair of features. Therefore, a further improvement of the above scheme can be done by considering the specific correlation structure for each pair of features. To address this, we formulate the following research questions:

- **RQ1:** How can we effectively model the complex, non-linear dependencies and tail-dependence often exhibited by geometric features of text components, which standard multivariate Gaussian models fail to capture?
- **RQ2:** Does employing a D-Vine copula structure, which allows flexible pairwise dependency modeling, yield an improvement in text-background separation performance (specifically Recall) compared

to multivariate Gaussian Copula?

To answer these research questions, we take into consideration a specific method, namely, pair-copula construction, for forming the joint distribution. Considering clustering, pair-copula construction was previously employed by Roy et al. [23] for image segmentation. Further, such a construction was examined as a clustering method using a number of datasets, by Roy et al. [25]. We here do not utilize pair-copula construction as a clustering method. Rather, we find it as a suitable model for the features obtained from a set of text components. Most importantly, pair-copula construction allows different types of correlation among different pairs of features, unlike multivariate Gaussian Copula. In effect, our method improves the previous models provided by Ghoshal et al. [13, 12].

This study necessitates a dataset with annotated ground truth text components, ideally specified at the pixel level rather than the OCR level. The ICDAR 2011 “Born-Digital Images” dataset (refer to Sect. 6) aligns with these prerequisites and is consequently adopted. It is pertinent to note that our proposed methodology is not contingent on any specific dataset and remains adaptable. Our proposed approach to text separation comprises the following phases:

- Leveraging the annotated ground truth text components within the training set, we compute an array of discriminative features facilitating the differentiation between text and non-text constituents. Subsequently, a D-vine multivariate distribution is fitted to these feature vectors. In this process, a selection of pair-copulas is applied, leading to the construction of a D-vine model characterizing the distribution of features.
- In the testing stage, each color image undergoes segmentation using statistical distribution-based methodologies. This yields a collection of spatially connected components, some of which correspond to textual content. As before, features are computed for these components. Each component (i.e., all its features) is then subjected to the previously learned D-vine model. Afterwards, employing hypothesis testing, we ascertain potential text components.

In summary, this work addresses the task of segregating textual content from digital images. By amalgamating segmentation techniques with probabilistic D-vine modeling, our method exhibits promise for enhancing text identification in diverse real-world scenarios. Let us start by introducing the concept of pair copula construction, one of whose subcategories is the D-vine representation.

2 Pair-Copula Construction for Multivariate Distribution

Let us introduce the concept of copula through the following theorem due to Sklar [19].

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 $x, y \in [-\infty, \infty]$,*

$$F(x, y) = C(F_1(x), F_2(y)). \quad (1)$$

Here $C(u, v)$ 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 . The advantage of the copula is that knowing only the margins, one can construct joint distributions having complex forms of dependence structure, using different types of copulas. This property of copulas makes them widely popular in financial mathematics [9] where often the joint distribution of two or more variables does not take any well-known parametric form. Other fields of application involve actuarial science [11] and hydrology [7]. For a theoretical study on copulas, Nelsen [19] provides a good introduction.

According to Nelsen [19], there are a large number of available copula families. The most popular are the elliptical and the Archimedean families. In this study, we use “Clayton”, “Gumbel” and “BB7” copulas from the Archimedean family. In addition, we use “Gaussian” copula from the elliptical family. Fig. 1 shows four copulas (including Gaussian copula) used to bind two normally distributed random variables.

Let us now concentrate on the pair-copula construction (PCC) of a multivariate distribution. We use “ f ” and “ F ” to denote probability density function and cumulative distribution function respectively.

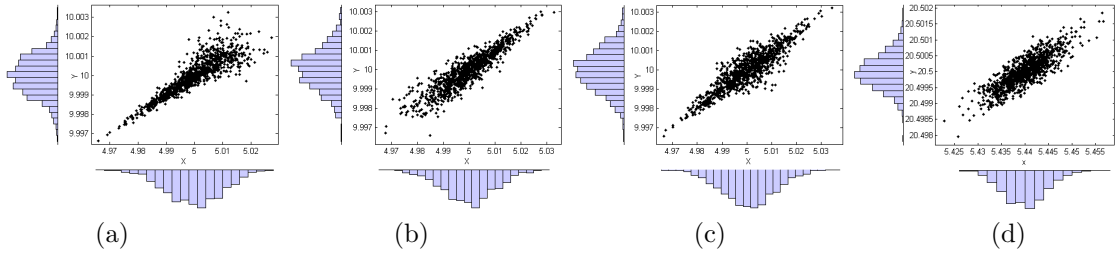


Figure 1: Visual comparison of dependence structures for two Gaussian margins coupled via (a) Clayton, (b) Gumbel, (c) BB7, and (d) Gaussian copulas. The dependence structures are clearly different.

Similarly, “ c ” and “ C ” denote respectively the probability density function and cumulative distribution function of a copula. Now, consider a d -dimensional random variable $\mathbf{X} = (X_1, \dots, X_d)$ with joint density $f(x_1, \dots, x_d)$. This density can be factorized as follows.

$$f(x_1, \dots, x_d) = f(x_d) \prod_{t=1}^{d-1} f(x_t | x_{t+1}, \dots, x_d). \tag{2}$$

The conditional distribution involved in Eq. 2 can be written as functions of the corresponding copula densities. Let u_1 and u_2 be standard uniform. Then we have the following “ h -function”

$$h(u_1, u_2, \Theta) = F(u_1 | u_2) = \frac{\delta C(u_1, u_2, \Theta)}{\delta u_2}, \tag{3}$$

where Θ is the set of parameters for the copula C of the joint distribution function of u_1 and u_2 . Now, consider the variable x_i and a set of variables \mathbf{v} that does not include x_i . Suppose v_j is the j^{th} element of \mathbf{v} . Let \mathbf{v}_{-j} denote the set \mathbf{v} that does not include v_j . It follows from Czado [6] that for any $v_j \in \mathbf{v}$,

$$F(x_i | \mathbf{v}) = h(F(x_i | \mathbf{v}_{-j}), F(v_j | \mathbf{v}_{-j}), \Theta_{i,j | \mathbf{v}_{-j}}). \tag{4}$$

Here $\Theta_{i,j | \mathbf{v}_{-j}}$ represents the parameters of the corresponding copula density $c_{i,j | \mathbf{v}_{-j}}(F(x_i | \mathbf{v}_{-j}), F(v_j | \mathbf{v}_{-j}))$. This shows that the conditional distributions with the conditioning set \mathbf{v} can be constructed recursively using the h -functions from the conditional distributions with a lower dimensional conditioning set. For the conditional density $f(x | \mathbf{v})$, it easily follows that

$$f(x_i | \mathbf{v}) = c_{i,j | \mathbf{v}_{-j}}(F(x_i | \mathbf{v}_{-j}), F(v_j | \mathbf{v}_{-j})) f(x_i | \mathbf{v}_{-j}). \tag{5}$$

It is to be noted that actually, the conditional copula $c_{i,j | \mathbf{v}_{-j}}(\cdot, \cdot)$ depends on the conditioning set \mathbf{v}_{-j} . In pair-copula model, $c_{i,j | \mathbf{v}_{-j}}(\cdot, \cdot)$ is simplified by dropping the dependence on \mathbf{v}_{-j} . Hobæk Haff et al. [15] observed that this simplification provides a good approximation to the multivariate distribution. Using Eq. 5, we could express the joint density $f(x_1, \dots, x_d)$ in terms of bivariate copulas. Such copulas are popularly known as pair-copulas and the method as pair-copula construction.

As an example, consider $d = 3$ where $\mathbf{X} = (X_1, X_2, X_3)$. Then Eq. 2 becomes:

$$f(x_1, x_2, x_3) = f(x_3) f(x_1 | x_2, x_3) f(x_2 | x_3). \tag{6}$$

Following the previous construction, the full PCC expansion for Eq. 6 becomes

$$\begin{aligned} f(x_1, x_2, x_3) &= f(x_1) f(x_2) f(x_3) \times \\ & c_{1,2}(F(x_1), F(x_2)) \times \\ & c_{2,3}(F(x_2), F(x_3)) \times \\ & c_{1,3|2}(F(x_1 | x_2), F(x_3 | x_2)). \end{aligned} \tag{7}$$

There are six ways of permuting the variables X_1, X_2 and X_3 . But only three of them give different decompositions of $f(x_1, x_2, x_3)$. However, in high dimensions, a significantly large number (in fact, $d!/2$)

of possible pair-copula constructions exist. To facilitate organizing these constructions, Bedford and Cooke [3, 4] introduced a graphical representation termed as “regular vine”. Here we concentrate on a special case of regular vines: the “D-vine”. Eq. 7 is a D-vine representation of $f(x_1, x_2, x_3)$. Now, let $\mathbf{v} = \{x_{i+1}, \dots, x_{i+j-1}, x_{i+j}\}$. Then, in general, the D-vine representation for the density $f(x_1, \dots, x_d)$ can be expressed as

$$\prod_{k=1}^d f(x_k) \prod_{j=1}^{d-1} \prod_{i=1}^{d-j} c_{i,i+j|\mathbf{v}_{-(i+j)}} [F(x_i|\mathbf{v}_{-(i+j)}), F(x_{i+j}|\mathbf{v}_{-(i+j)})]. \tag{8}$$

This D-vine specification may be given in the form of a set of trees [3, 4]. Fig. 2 shows the tree representation of a D-vine, in five dimensions. It consists of four trees arranged in four levels. An edge of a tree corresponds to a pair-copula density denoted by the edge label. For example, the edge 13|2 denotes the pair-copula $c_{13|2}$. The whole structure has $d(d - 1)/2$ edges. This tree structure is easy to construct and is helpful in understanding Eq. 8.

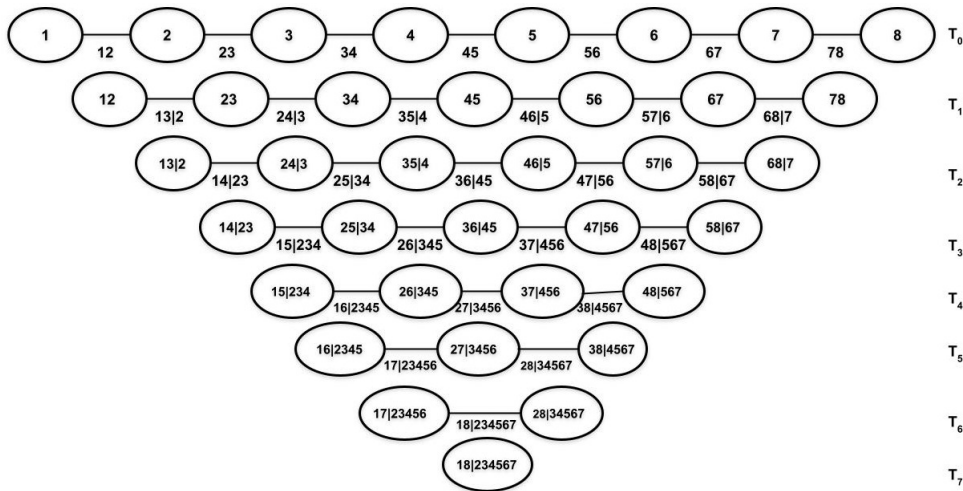


Figure 2: D-Vine Specification with 5 Variables, Consisting 4 Trees and 10 Edges. An Edge is Associated with a Pair-Copula.

To provide an intuitive understanding of this construction, consider the multivariate distribution of the feature vector $\mathbf{X} = (X_1, \dots, X_d)$ as a complex “group conversation” involving d participants. The D-Vine approach decomposes this complex group dynamic into a structured series of “private, one-on-one discussions” (pair-copulas).

In this analogy:

- The “full group conversation” represents the joint probability density $f(x_1, \dots, x_d)$.
- The “one-on-one discussions” represent the bivariate pair-copulas (e.g., c_{12}, c_{23}).
- The “flow of information” represents the conditional structure.

First, the D-Vine models the direct dialogue between adjacent neighbors (e.g., the dependence between X_1 and X_2). Then, in subsequent levels, it models the relationship between non-adjacent participants (e.g., X_1 and X_3)—but critically, this relationship is conditioned on the information passed through their common neighbor (X_2). By stacking these pairwise building blocks, we can reconstruct the full, complex dependency structure of the entire group using only simple bivariate models.

3 Shape Based Feature Extraction from Connected Components

We now direct our attention to the initial phase of our framework, specifically, the establishment of a model describing the features found within text elements in an image. As mentioned earlier, we have access to ground truth text components in the training images. With this in mind, we can proceed to calculate various features using these connected components. Below, we list the specific features we use to distinguish between components that contain text and those that do not.

ER: The text-like patterns are usually elongated. Thus elongation is an important measure to distinguish text and non-text. For the elongatedness ratio (*ER*) we use the following measure.

$$ER = \frac{\text{No. of boundary points}}{\sqrt{\text{Total no. of points}}}. \quad (9)$$

OR: The object pixels ratio (*OR*) is computed by taking the bounding box. Due to the elongated nature of texts only a few object pixels fall inside the text bounding box. On the other hand, elongated non-texts are usually straight lines, so, contribute enough object pixels. The *OR* is computed by counting the object pixels and normalizing the number by the area of the bounding box.

AR: The aspect ratio $AR = \min\{\text{height}/\text{width}, (\text{width}/\text{height})\}$ of a non-text component is either very small or very large compared to text components.

ICAR: This is the ratio of component area and the image area. A text symbol usually occupies only a small portion of the total image. *ICAR* is used to exclude components with excessively large areas.

TH: Thickness (*TH*) of a component is calculated as follows. Let h_i and v_i be the horizontal and the vertical run lengths of a pixel p_i at the i^{th} position of a component CC_j . We next compute the minimum of h_i and v_i and further constitute a set $MIN_j = \{m_i \text{ s.t. } m_i = \min(h_i, v_i), \forall i\}$. 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 as the most frequently occurring m_j in the set MIN_j .

AxR: Axial ratio (*AxR*), for any structure or shape with two or more axes, is the ratio of the length (or magnitude) of those axes to each other - the longer axis divided by the shorter. Here, we calculate the *MajorAxisLength* and *MinorAxisLength* of a CC. The axial ratio is defined as $AxR = (\text{MajorAxisLength}/\text{MinorAxisLength})$.

covTH: The *coefficient of variation* of the thickness values.

Perim: The perimeter of the connected component.

Combining these features, we construct the feature vector $\Upsilon = \{ER, OR, AR, ICAR, TH, AxR, covTH, Perim\}$ for a connected component. Such a feature vector obtained from ground truth text component is termed as $\Upsilon^{(train)}$.

4 D-vine-based Statistical Model for Feature Distribution

We now intend to obtain a parametric statistical distribution that effectively approximates the distribution inherent in the feature vectors. Although the multivariate Gaussian distribution is a plausible candidate, it's important to understand that each margin of such a distribution corresponds to a univariate Gaussian distribution. This intrinsic characteristic poses limitations in selecting appropriate distributions for each individual margin. To understand this, let us present histogram plots for each margin within the feature vector $\Upsilon^{(train)}$ in Fig. 3. The clear deviation from Gaussian distribution across most of the features highlights the unsuitability of the multivariate Gaussian for $\Upsilon^{(train)}$. On the other hand, our previous section discussed the benefits of the D-vine distribution. This approach not only lets us choose

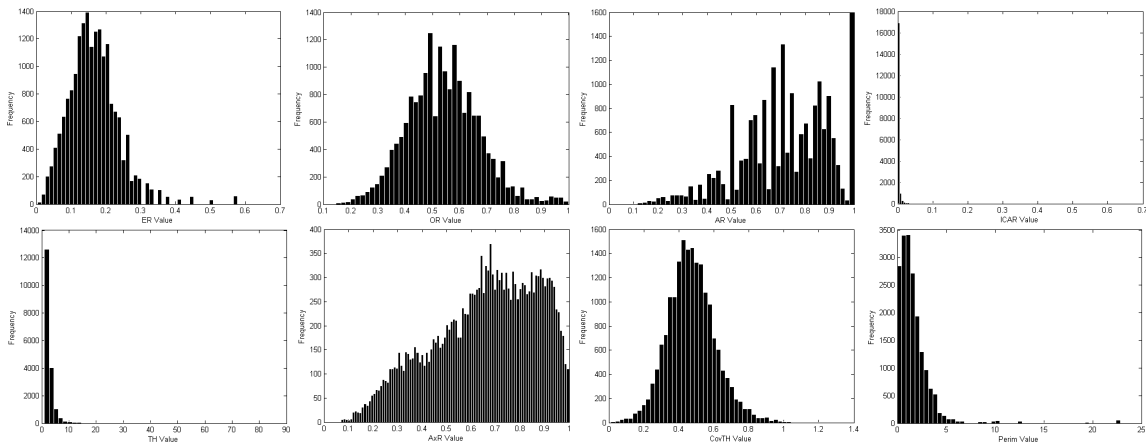


Figure 3: The marginal distributions of the feature vector $\Upsilon^{(train)}$

suitable margins but also helps us pick appropriate relationships between margins, from a predefined set. Thus, the D-vine is more suitable to model the feature distribution.

To find appropriate marginal distributions, we deploy a series of parametric distributions on each feature. The best fitted distribution, determined through maximizing likelihood, is selected as the marginal distribution for that specific feature. Our options include Gaussian, Gamma, and Log-Normal distributions. This procedure may yield distinct parametric distributions for each feature. The important aspect in this context is the arrangement of individual features. Based on different permutations of the features, different D-vine trees can be generated. Therefore, a specific feature permutation should be selected prior to pair copula estimation. To address this, we adopt pairwise Kendall’s Tau as a measure of association between a pair of features. This guides us in establishing a feature order, which is based on the decreasing order of Kendall’s Tau correlation. In Fig. 4, we present the pairwise Kendall’s Tau values. We observe that *ER* and *Perim* have maximum correlation. Therefore, we consider these two features as first two candidates of our feature vector. Afterwards, we observe that the pair (*ER*, *TH*) has more correlation compared to any other pair. Thus, our feature vector so far becomes $\{Perim, ER, TH\}$. Proceeding this way, we establish the ordered feature vector as $\Upsilon^{(train)} = \{Perim, ER, TH, covTH, OR, ICAR, AR, AxR\}$.

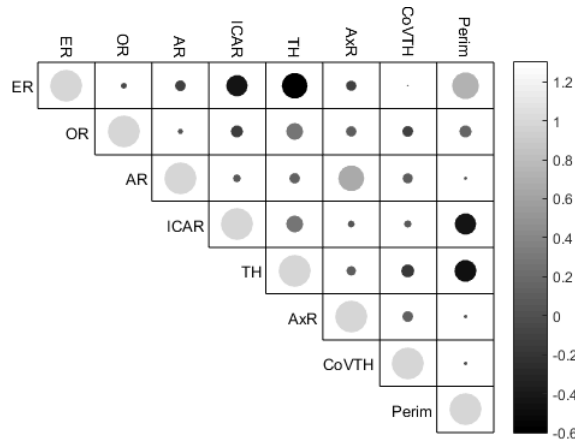


Figure 4: Kendall’s Tau for all pairs of features. The colors correspond to the values of Kendalls Tau.

Given the ordered feature vector $\Upsilon^{(train)}$, the marginal distributions are presented in Table 1. Notably, in most instances, the Log-Normal distribution was found to be suitable.. The Log-Normal parameters are α and β that control the location and the shape of the distribution respectively. On three occasions, we obtain Gaussian distribution as an appropriate distribution. In Table 1, μ and σ^2 represent the mean and the variance of the corresponding Gaussian distribution.

Table 1: Parametric distributions corresponding to individual features

Feature	Perim	ER	TH	covTH
Distribution	<u>Log-Normal</u> $\alpha = 3.6585$ $\beta = 0.6311$	<u>Log-Normal</u> $\alpha = 1.6865$ $\beta = 0.0670$	<u>Log-Normal</u> $\alpha = 0.9311$ $\beta = 0.2358$	<u>Gaussian</u> $\mu = 0.4581$ $\sigma^2 = 0.1362$
Feature	OR	ICAR	AR	AxR
Distribution	<u>Gaussian</u> $\mu = 0.5393$ $\sigma^2 = 0.1326$	<u>Log-Normal</u> $\alpha = -6.8422$ $\beta = 1.2346$	<u>Gaussian</u> $\mu = 0.7155$ $\sigma^2 = 0.0335$	<u>Log-Normal</u> $\alpha = 0.4862$ $\beta = 0.1583$

Table 2: Pair-copulas involved in trees at three lower levels (2) of the D-Vine model for text features.

T_1	Parameters	T_2	Parameters	T_3	Parameters
C_{12} (Gaussian)	$\begin{bmatrix} 1 & -0.91 \\ -0.91 & 1 \end{bmatrix}$	$C_{13 2}$ (Gaussian)	$\begin{bmatrix} 1 & -0.51 \\ -0.51 & 1 \end{bmatrix}$	$C_{14 23}$ (Gaussian)	$\begin{bmatrix} 1 & -0.05 \\ -0.05 & 1 \end{bmatrix}$
C_{23} (Gumbel)	0.0574	$C_{24 3}$ (Gumbel)	0.01	$C_{25 34}$ (Gaussian)	$\begin{bmatrix} 1 & -0.06 \\ -0.06 & 1 \end{bmatrix}$
C_{34} (Gaussian)	$\begin{bmatrix} 1 & -0.43 \\ -0.43 & 1 \end{bmatrix}$	$C_{35 4}$ (BB7)	0.41 0.55	$C_{36 45}$ (Gaussian)	$\begin{bmatrix} 1 & -0.08 \\ -0.08 & 1 \end{bmatrix}$
C_{45} (BB7)	0.69 0.74	$C_{46 5}$ (Clayton)	0.08	$C_{47 56}$ (BB7)	0.21 0.43
C_{56} (Gaussian)	$\begin{bmatrix} 1 & -0.23 \\ -0.23 & 1 \end{bmatrix}$	$C_{57 6}$ (BB7)	0.67 0.69	$C_{58 67}$ (Clayton)	0.17
C_{67} (Gaussian)	$\begin{bmatrix} 1 & 0.09 \\ 0.09 & 1 \end{bmatrix}$	$C_{68 7}$ (Clayton)	0.01		
C_{78} (Gumbel)	0.84				

Once the marginal distributions are established, we can proceed to construct the D-vine tree. Regarding copula parameter estimation, Hobæk Haff [14] introduced a “stepwise semi-parametric estimator” (SSP) that separately estimates both the margins and the bivariate pair-copulas in a hierarchical manner within the D-vine tree (depicted in Fig. 2). Following this method, we estimate the bivariate copulas within the D-vine. This estimation process also entails determining the optimal pair-copula that best characterizes the dependence structure inherent in the data. To accomplish this, we leverage a collection of four bivariate copulas: Gaussian, Clayton, Gumbel, and BB7, for each bivariate dependence structure. The best-fit copula is identified through the maximization of log-likelihood. The fitted pair-copulas for the D-vine model are given in Table 2 (for three levels from the bottom of D-vine tree) and in Table 3 (for the remaining top levels). These two tables present the parameters for each pair-copula in our study. In the first level of copulas (T_1), we observe the utilization of Gaussian, BB7, and Gumbel copulas. Notably, it becomes apparent from C_{12} that the features *Perim* and *ER* exhibit a significant negative correlation. Interestingly, a BB7 copula is selected for the pair (*covTH*, *OR*), and an analysis of its parameters reveals significant tail-dependence at both tails. In contrast, the pair (*ER*, *TH*) exhibits upper tail dependence, as indicated by C_{23} , which corresponds to a Gumbel copula.

Recall that the study made by Ghoshal et al. [12] proposed a multivariate Gaussian copula to connect all the features. In our comparison, we adopt a similar model but retain the previously employed marginal

Table 3: Pair-copulas involved in four top levels (2) of the D-Vine model for text features.

T_4	Parameters	T_5	Parameters	T_6	Parameters	T_7	Parameters
$C_{15 234}$ (Gaussian)	$\begin{bmatrix} 1 & -0.18 \\ -0.18 & 1 \end{bmatrix}$	$C_{16 2345}$ (Gumbel)	0.02	$C_{17 23456}$ (Gumbel)	0.17	$C_{18 234567}$ (Gumbel)	0.01
$C_{26 345}$ (Gaussian)	$\begin{bmatrix} 1 & -0.70 \\ -0.70 & 1 \end{bmatrix}$	$C_{27 3456}$ (Gaussian)	$\begin{bmatrix} 1 & -0.12 \\ -0.12 & 1 \end{bmatrix}$	$C_{28 34567}$ (Gaussian)	$\begin{bmatrix} 1 & -0.09 \\ -0.09 & 1 \end{bmatrix}$		
$C_{37 456}$ (BB7)	0.23 0.15	$C_{38 4567}$ (Gaussian)	$\begin{bmatrix} 1 & -0.15 \\ -0.15 & 1 \end{bmatrix}$				
$C_{48 567}$ (Gumbel)	0.01						

distributions. Consequently, the Gaussian copula for these features yields the correlation matrix (Σ) detailed in Table 4. It is noteworthy that most of the first-level (T_1) copulas in the D-vine structure (Fig.

Table 4: Correlation matrix for multivariate Gaussian copula [12]

$$\Sigma = \begin{bmatrix} 1.00 & -0.90 & -0.15 & 0.07 & -0.17 & 0.66 & 0.11 & 0.11 \\ -0.90 & 1.00 & -0.05 & 0.02 & -0.04 & -0.63 & -0.16 & -0.17 \\ -0.15 & -0.05 & 1.00 & -0.43 & 0.64 & -0.17 & 0.33 & 0.27 \\ 0.07 & 0.02 & -0.43 & 1.00 & -0.29 & 0.08 & 0.24 & 0.25 \\ -0.17 & -0.04 & 0.64 & -0.29 & 1.00 & -0.23 & -0.02 & 0.12 \\ 0.66 & -0.63 & -0.17 & 0.08 & -0.23 & 1.00 & 0.11 & 0.09 \\ 0.11 & -0.16 & 0.33 & 0.24 & -0.02 & 0.11 & 1.00 & 0.85 \\ 0.11 & -0.17 & 0.27 & 0.25 & 0.12 & 0.09 & 0.85 & 1.00 \end{bmatrix}$$

2) are bivariate Gaussian copulas. The correlation coefficients presented at this level should align with those found in Σ . This alignment is indeed evident, such as the correlation coefficient of -0.91 for the pair ($Perim, ER$) in both Table 2 and Σ . However, at higher levels, the construction of the multivariate distribution takes diverse paths. An important aspect to consider is the presence of tail dependence, as represented by the BB7 copula in Table 2, which exhibits significant tail dependence characteristics. Now, this aspect cannot be identified by the multivariate Gaussian copula since, by definition, it does not possess tail dependence.

4.1 Testing of hypothesis to distinguish text and non-text

In the context of text identification, we encounter the challenging task of distinguishing between text and non-text components within an image. This necessitates a binary classification, where traditional binary classifiers such as Support Vector Machines (SVM) may seem like a natural choice. However, a significant hurdle arises in the form of a lack of examples for non-text components, unlike their text counterparts. Given the diverse and versatile patterns exhibited by non-text components, defining a comprehensive set of non-text examples becomes impractical. Consequently, conventional binary classifiers may not be well-suited for this scenario.

To tackle this issue, we approach the problem as a one-class classification problem, focusing on the identification of text components. The fundamental idea is to construct a confidence region within the D-vine multivariate distribution. Components falling within this region are identified as text components. However, two main challenges emerge in this context. Firstly, the distribution function of a D-vine multivariate distribution lacks an analytical form. Consequently, for each test component, we must numerically evaluate the D-vine distribution function to obtain its cumulative probability, which is a computationally intensive task given the typically large number of components generated from an image. Secondly, establishing a confidence region for a multivariate distribution typically requires non-analytical methods, such as bootstrapping, further increasing computational complexity.

In light of these challenges, we adopt an alternative strategy. We compute the log-likelihood statistic for feature vectors belonging to the training text components, and the resulting distribution is depicted in Fig. 5. Subsequently, we aim to develop a parametric model to approximate the distribution of this log-likelihood statistic, simplifying the testing process. When data indeed follows the D-Vine distribution (as illustrated in Fig. 2), we have knowledge of the log-likelihood statistic's distribution. During testing, if a feature vector's log-likelihood statistic lies at the tails of this distribution, we can reject the D-Vine distribution, indicating a non-text component. This strategy eliminates the need for numerical evaluation of the distribution function and replaces the multidimensional confidence region with confidence bounds for a univariate distribution, which is typically analytically tractable.

We observe that the log-likelihood statistic's distribution exhibits a significant right-skew. Consequently, we choose to fit a Beta distribution to approximate this distribution. We first normalize (using the min-max scaling) the log-likelihood values to the (0,1) interval to align with the Beta distribution's support. Subsequently, we employ maximum likelihood estimation to determine the Beta distribution parameters. For our dataset, these parameters are estimated as $\alpha = 16.55$ and $\beta = 1.67$ respectively.

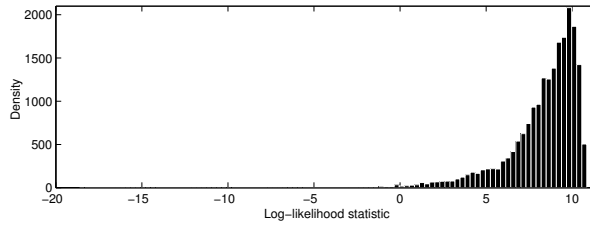


Figure 5: The Distribution of the log-likelihood statistic for $\Upsilon^{(train)}$.

Evidently, $\alpha > \beta$ indicates a left-skewed distribution. Therefore, the log-likelihood statistic of a potential non-text component lies at the left tail of this distribution.

In order to detect a potential non-text component, yielded from a test image, we proceed as follows. We adopt a one-sided hypothesis test with a significance level of 5%. A component from a test image is classified as non-text if its normalized log-likelihood statistic falls below the critical value corresponding to the 5th percentile of the fitted Beta distribution. Conversely, components with log-likelihoods residing within the upper 95% of the probability mass are retained as text candidates.

5 Text Identification Methodology

To obtain spatially connected components inside an image, we employ a segmentation for the images of the given dataset. It is important to note that the training image samples are exempt from the segmentation process, as connected components can be directly derived from the provided ground truths. Therefore, segmentation is exclusively applied to the test images. Upon segmentation, the result comprises multiple connected components distributed across various clusters, some of which are potential text components. Our assumption here is that a single text component exhibits homogeneity in terms of color and brightness, i.e., it remains unbroken following clustering. Post-segmentation, we first eliminate sufficiently small and large components, as they typically do not contribute significantly to text identification. Small components tend to represent noise, while large ones often constitute the image background. Subsequently, we compute a set of features as detailed in Sect. 3 for each retained component. Given the trained model, we proceed to apply the hypothesis test outlined in Sect. 4.1 to identify potential text components. The complete framework for text identification system is presented in Fig. 6.

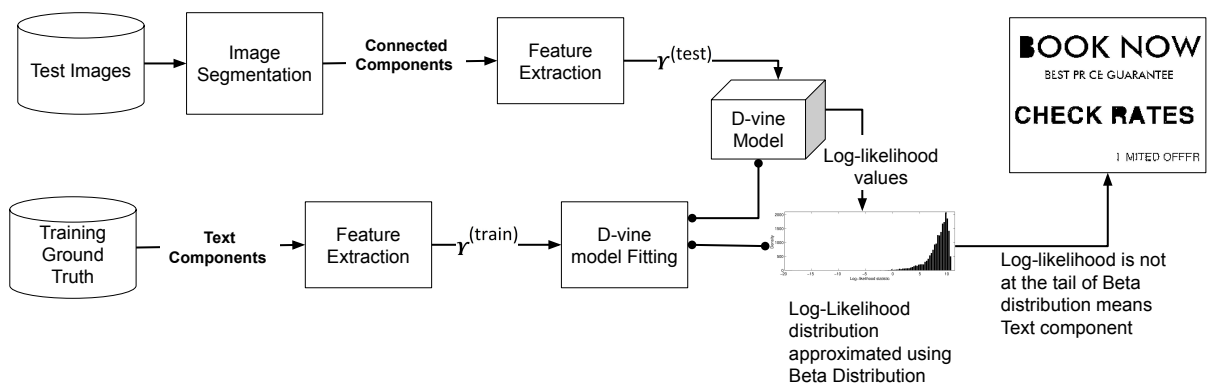


Figure 6: The text identification system framework.

6 Experimental Protocol

The experimental results are obtained on ICDAR 2011 Born Digital dataset [17]. This dataset contains 420 training images are used to extract the shape-based features and 102 test images. Born-digital images

are inherently of low resolution in order to be transmitted online efficiently. Here, we have two different models for the feature distribution. The first one is our proposed D-vine-based model and the second one is the model proposed by Ghoshal et al. [12] that used multivariate Gaussian copula. As pointed out in Sect. 5, every test image should undergo a segmentation method to identify various components. For this purpose, we employed model based segmentation methods encompassing Gaussian Mixture Model (GMM) segmentation [10], Semi-wrapped Gaussian Mixture Model (SWGMM) segmentation [22, 24] and D-vine Mixture Model (DVMM) segmentation [25]. Along with these, we also employ Topology Adaptive Self-Organizing Circular Neural Network (TASOCNN) [21] which is a neural model used for color image segmentation. In order to measure the performance of various models, we use the Recall and Precision metric. In Sect. 7 we describe the relative importance of these two measures. All the experiments were performed on a computer with *intel*[®] core i5 processor and 16 GB memory. Codes were written using *MATLAB*[®] R2015a with statistical toolbox.

7 Results and Discussion

Let us now apply the text extraction methodology for extracting possible text components. We first present some images and the corresponding detected text components, in Table 5. Table 5 is organized to compare two different models, namely, our D-vine-based model and multivariate Gaussian copula-based model of Ghoshal et al. [12]. Each pair of rows (except the first row) represent results obtained by these two models. Studying Table 5 in this way, we may observe that in most cases the D-vine-based model outperforms the multivariate Gaussian copula-based modeling of feature distribution. We already discussed the advantages of a D-vine-based model over the multivariate Gaussian copula. Nonetheless, it is crucial to highlight a particular consideration in this context. The D-vine model (as depicted in Fig. 2) incorporates Gaussian copulas at every hierarchical level. Consequently, the results yielded by the D-vine-based model may not significantly deviate from those obtained using the multivariate Gaussian copula-based model. Indeed, as evident from Table 5, there are cases where both models yield similar results, with only minor distinctions. To assess the overall performance of the two models and various segmentation methods, we have compiled a table summarizing the Recall and Precision percentages at the pixel level, as presented in Table 6. Before comparisons, it is crucial to grasp the relative importance of Recall and Precision scores. Recall gauges the accuracy of text component retrieval, while Precision quantifies the level of noise (i.e., non-text pixels) within the result. Although both measures might appear equally important at first glance, when considering the primary goal of text extraction followed by recognizing the extracted components, certain differences become evident.

The extracted components typically undergo recognition via Optical Character Recognition (OCR) systems, which often include modules for discarding possible non-text entities. A high Recall indicates a higher likelihood of all text components being present in the final outcome, even if some non-text components are included. The OCR system can usually handle the exclusion of non-text components. Conversely, a high Precision suggests fewer non-text components in the result, but it does not guarantee the inclusion of all text components. This scenario is more problematic, especially for non-dictionary words, as the OCR system cannot predict missing text components. Hence, in a text extraction system, a high Recall is generally more desirable than a high Precision.

Regarding different segmentation algorithms, we note that DVMM yields the highest Recall percentage, with TASOCNN closely approaching DVMM. SWGMM, on the other hand, produces the lowest Recall percentage but excels in Precision. This implies that SWGMM typically contains fewer non-text pixels in the result compared to other algorithms.

Comparing the scores based on the two feature distribution models, we find that except for SWGMM and GMM, the D-vine-based model exhibits better Recall values than the Multivariate Gaussian copula-based model. Further examination reveals that the D-vine-based model provides a more accurate fit for the feature distribution, enhancing its ability to correctly classify all text components. However, this increased accuracy may also classify more text-like non-text components as text, resulting in a lower Precision percentage for the D-vine-based model. Given the preference for good Recall, we conclude that the D-vine-based model outperforms the Gaussian copula-based model.

Finally, let us compare our method with the methods reported in Table II of [17]. The results (Recall and Precision) are presented in Table 7. We observe that the DVMM segmentation coupled with D-vine

Table 5: Text extraction results on born-digital images dataset. First row presents the original images. Each pair of the next rows present text components produced by different algorithms by applying D-Vine-based and multivariate Gaussian copula based models, respectively.

Images	
TASOCNN+ Dvine	
TASOCNN+ Gauss	
SWGMM+ Dvine	
SWGMM+ Gauss	
DVMM+ Dvine	
DVMM+ Gauss	
GMM+ Dvine	
GMM+ Gauss	

Table 6: Recall and Precision percentage for text extraction from Born-Digital Images dataset by different algorithms.

Methods	D-vine-based model		Multivariate Gaussian copula	
	Recall	Precision	Recall	Precision
TASOCNN	80.62%	67.38%	78.24%	63.85%
SWGMM	66.23%	83.63%	69.08%	85.54%
DVMM	82.25%	66.39%	80.13%	67.20%
GMM	67.66%	72.06%	70.39%	77.97%

modeling is comparable to the best performer i.e., OTCYMIST in terms of Recall. However, the Precision produced by OTCYMIST significantly exceeds the Precision given by our method. This implies that the results of OTCYMIST include less amount of non-text compare to our method.

OTCYMIST [17] uses a structural, rule-based approach that extracts connected components from multiple color channels and filters them using strict topological constraints (e.g., Euler number, aspect ratio). It further checks for group consistency (using Minimum Spanning Trees to enforce word alignment and height similarity). This structural verification filters background noise that does not form “lines” or “words,” resulting in superior Precision.

In contrast, our D-Vine method is a probabilistic approach that models the dependency structure of geometric features. While it matches OTCYMIST in Recall by effectively identifying text components with complex feature dependencies, it operates primarily on component-level statistics. It does not explicitly enforce the high-level structural constraints (like MST-based word grouping) used by OTCYMIST. Consequently, our method may accept isolated background elements that statistically resemble text features but do not form coherent text lines, leading to lower Precision compared to the structural rigidity of OTCYMIST.

Table 7: Text extraction results provided by our methodology (first row) and reported in [17].

Methods	Recall	Precision
DVMM+Dvine	82.25%	66.39%
OTCYMIST	80.99%	71.13%
SASA	71.93%	54.78%
Textorter	65.20%	62.50%

8 Conclusion And Future Scope

This paper presents a comprehensive methodology for the extraction of text entities from digital images. The core motivation behind this research is the critical role of text extraction in numerous applications, including document analysis and optical character recognition (OCR) systems. Recognizing the challenges associated with accurate text extraction from images, the paper outlines a two-step approach. The methodology comprises two fundamental components: color image segmentation and copula-based modeling of features. Color image segmentation is employed to identify distinct components within the image. On the other hand, the features are extracted from ground-truth text components, ensuring their relevance and effectiveness in distinguishing text from non-text areas. Subsequently, the copula-based modeling of features is introduced to characterize the distribution of these features. Finally, for an unknown (test) image, individual components are extracted using color image segmentation. Afterwards, the copula-based model is employed to identify candidate text components.

The proposed D-vine model (with DVMM) achieved a Recall of 82.25% and a Precision of 66.39% on the ICDAR 2011 “Born-Digital Images” dataset. It significantly outperforms the multivariate Gaussian copula baseline. These findings hold important implications for real-world applications. High recall is critical in forensic text analysis and historical document digitization. In these scenarios, missing a text segment is a severe failure. Our method minimizes missed detections. It captures text even when the geometry is complex. Therefore, it serves as a robust tool for systems where information retrieval is the priority.

Regarding text extraction, our method represents a promising starting point for characterizing feature distributions. However, there remains ample scope for future research in this domain. The following future directions may be considered.

- One avenue for exploration involves the incorporation of more discriminating features to enhance the differentiation between text and non-text regions. This could potentially lead to even more robust and accurate text extraction methodologies.
- The D-vine offers enough flexibility in feature modeling. We may customize the tree structure primarily by assuming independence between a pair of nodes in Fig. 2. For instance, independence

can be assumed at T_7 , as the correlation at this node is found to be insignificant (see Table 3). This tailored customization of the D-vine structure facilitates faster estimation of parameters. Future research avenues may explore the potential of such customized models.

- The ordering of features plays a crucial role in modeling the joint distribution of features. The feature order determines the extent to which association between pair of features is captured. However, it is essential to note that with the D-vine structure, not all significant associations may be directly captured. For example, in Fig. 4 we observe that ER has significant correlation with $ICAR$, TH and $Perim$. Nevertheless, we could only consider the pair $(ER, Perim)$ directly in our D-vine model. This is obvious due to the underlying construction of D-vine tree structure. To address this limitation, a prospective study could investigate alternative pair-copula construction methods. For instance, as indicated in Fig. 4, where ER exhibits substantial correlations with most other features, a Canonical vine (C-vine) construction could be employed. Such models are therefore worth exploring in future research endeavors.

References

- [1] Marios Anthimopoulos, Basilis Gatos, and Ioannis Pratikakis. Detection of artificial and scene text in images and video frames. *Pattern Anal. Appl.*, 16(3):431–446, 2013.
- [2] Prakriti Banik, Ujjwal Bhattacharya, and Swapan K. Parui. Segmentation of bangla words in scene images. In *Proc. of Indian Conf. on Computer Vision, Graphics and Image Processing*, pages 70:1–70:7. ACM, 2012.
- [3] T. Bedford and R. M. Cooke. Probability density decomposition for conditionally dependent random variables modeled by vines. *Annals of Mathematics and Artificial Intelligence*, 3(1-4):245–268, 2001.
- [4] T. Bedford and R. M. Cooke. Vines—a new graphical model for dependent random variables. *Annals of Statistics*, 30(4):1031–1068, 2002.
- [5] U. Bhattacharya, S. K. Parui, and S. Mondal. Devanagari and bangla text extraction from natural scene images. In *Proc. of Int. Conf. on Doc. Anal. and Recog.*, pages 171–175, 2009.
- [6] C. Czado. Pair-copula constructions of multivariate copulas. In P. Jaworski, F. Durante, W. Härdle, and T. Rychlik, editors, *Copula Theory and Its Applications*, volume 198, pages 93–109. Springer Berlin Heidelberg, 2010.
- [7] C. De Michele, G. Salvadori, M. Canossi, A. Petaccia, and R. Rosso. Bivariate statistical approach to check adequacy of dam spillway. *J. Hydrologic Engineering*, 10(1):50–57, 2005.
- [8] Yongkun Du, Zhineng Chen, Yuchen Su, Caiyan Jia, and Yu-Gang Jiang. Instruction-guided scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(4):2723–2738, 2025.
- [9] P. Embrechts, F. Lindskog, and A. McNeil. Modelling dependence with copulas and applications to risk management. In S. Rachev, editor, *Handbook of Heavy Tailed Distributions in Finance*, pages 329–384. Elsevier, 2003.
- [10] M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):381–396, 2002.
- [11] E. W. Frees and E. A. Valdez. Understanding relationships using copulas. *North American Actuarial J.*, 2(1):1–25, 1998.
- [12] Ranjit Ghoshal, Anandarup Roy, Ayan Banerjee, Bibhas Chandra Dhara, and Swapan K. Parui. A novel method for binarization of scene text images and its application in text identification. *Pattern Anal. Appl.*, 22(4):1361–1375, 2019.

- [13] Ranjit Ghoshal, Anandarup Roy, and Swapan K. Parui. A copula based statistical model for text extraction from scene images. In Pradipta Maji, Ashish Ghosh, M. Narasimha Murty, Kuntal Ghosh, and Sankar K. Pal, editors, *Pattern Recognition and Machine Intelligence - 5th International Conference, PReMI 2013, Kolkata, India, December 10-14, 2013. Proceedings*, volume 8251 of *Lecture Notes in Computer Science*, pages 489–494. Springer, 2013.
- [14] I. Hobæk Haff. Parameter estimation for pair-copula constructions. *Bernoulli*, 19(2):462–491, 2013.
- [15] I. Hobæk Haff, K. Aas, and A. Frigessi. On the simplified pair-copula construction - simply useful or too simplistic? *J. Multivar. Anal.*, 101(5):1296–1310, 2010.
- [16] K. Jung, I. K. Kim, T. Kurata, M. Kouroggi, and H. J. Han. Text scanner with text detection technology on image sequences. In *Proc. of Int. Conf. on Pattern Recognition*, volume 3, pages 473–476, 2002.
- [17] D. Karatzas, S. Robles Mestre, J. Mas, F. Nourbakhsh, and P. Pratim Roy. Icdar 2011 robust reading competition - challenge 1: Reading text in born-digital images (web and email). In *Proceedings of the 2011 International Conference on Document Analysis and Recognition*, pages 1485–1490, USA, 2011. IEEE Computer Society.
- [18] R. Lienhart and F. Stuber. Automatic text recognition in digital videos. In *Image and Video Processing IV, Proc. SPIE 2666*, pages 180–188, 1996.
- [19] R. B. Nelsen. *An Introduction to Copulas*. Springer, 2006.
- [20] A. Roy, S. K. Parui, and U. Roy. A color based image segmentation and its application to text segmentation. In *Proc. of Indian Conf. on Computer Vision, Graphics and Image Processing*, pages 313–319. IEEE Computer Society, 2008.
- [21] A. Roy, S. K. Parui, and U. Roy. TASOCNN: a topology adaptive self-organizing circular neural network and its application to color segmentation. In *Proc. of Indian Conf. on Computer Vision, Graphics and Image Processing*, pages 427–434. ACM, 2010.
- [22] A. Roy, S. K. Parui, and U. Roy. Color image segmentation using a semi-wrapped gaussian mixture model. In *Proc. of Int. Conf. on Pattern Recognition and Machine Intelligence*, pages 148–153. Springer-Verlag, 2011.
- [23] A. Roy, S. K. Parui, and U. Roy. A finite mixture model based on pair-copula construction of multivariate distributions and its application to color image segmentation. In *Proc. of Indian Conf. on Computer Vision, Graphics and Image Processing*, page 10. ACM, 2012.
- [24] A. Roy, S. K. Parui, and U. Roy. SWGMM: A semi-wrapped gaussian mixture model for clustering of circular-linear data. *Pattern Anal. Appl.*, 19(3):631–645, 2016.
- [25] Anandarup Roy and Swapan K. Parui. Pair-copula based mixture models and their application in clustering. *Pattern Recognition*, 47(4):1689–1697, 2014.
- [26] K. Sobottka, H. Bunke, and H. Kronenberg. Identification of text on colored book and journal covers. In *Proc. of Int. Conf. on Doc. Anal. and Recog.*, pages 57–63, 1999.
- [27] Yuchen Su, Zhineng Chen, Yongkun Du, Zhilong Ji, Kai Hu, Jinfeng Bai, and Xieping Gao. Explicit relational reasoning network for scene text detection. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence*, pages 7069–7077. AAAI Press, 2025.
- [28] C.M. Tsai and H.J. Lee. Binarization of color document images via luminance and saturation color features. *IEEE Trans. on Image Processing*, 11(4):434–451, 2002.
- [29] Tao Wang, David J. Wu, Adam Coates, and Andrew Y. Ng. End-to-end text recognition with convolutional neural networks. In *Proceedings of the 21st International Conference on Pattern Recognition, ICPR 2012, Tsukuba, Japan, November 11-15, 2012*, pages 3304–3308. IEEE Computer Society, 2012.

-
- [30] V. Wu, R. Manmatha, and E. M. Riseman. Textfinder: An automatic system to detect and recognize text in images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(11):1224–1229, 1999.
- [31] Jianjun Xu, Yuxin Wang, Hongtao Xie, and Yongdong Zhang. Ote: Exploring accurate scene text recognition using one token. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28327–28336, 2024.
- [32] Xu-Cheng Yin, Wei-Yi Pei, Jun Zhang, and Hong-Wei Hao. Multi-orientation scene text detection with adaptive clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9):1930–1937, 2015.