Word Segmentation by Component Tracing and Association (CTA) Technique

DOI:10.36909/jer.15207

Enock Osoro Omayio*, Indu Sreedevi, Jeebananda Panda

Electronics and Communication Dept, Delhi Technological University, India. Shahbad Daulatpur, Bawana Road, Delhi 11004

* Corresponding Author: enockosoro_phd2k17@dtu.ac.in & omayio2008@gmail.com

ABSTRACT

Word-level segmentation is a very important step in many document analysis systems. This is because word is the most important unit in any language systems. Word segmentation of handwritten documents is a very challenging task due to cursive nature of handwriting, overlap, touching and crossing of adjacent words, non-straight baselines, and cluttering among many others. Of these challenges, crossing is the most difficult challenge to deal with. This paper proposes a novel offline word-level segmentation technique for handwritten documents that addresses the challenges of touching and crossing of words. The main contribution of the paper is junction branch association (JBA) method that specifically handles touching and crossing words where many other proposed methods fail. The proposed method has been evaluated with ICDAR2009 and ICDAR2013 benchmark datasets of handwritten scripts. Also, crossing words extracted from FireMaker dataset of handwritten documents have been used to specifically evaluate performance of JBA method in segmenting crossing words.

Keywords: Word segmentation; line segmentation; document analysis; connected component (CC); dynamic time warping (DTW)

1. INTRODUCTION

Handwritten documents (HWD) are common primary source of historical and current information. In HWDs, word is the most important unit that is made up of connected components (CC) of characters and strokes. Words are used by many document analysis

systems (DAS) in different domain tasks like word recognition, writer identification, writer verification, historical manuscript dating, and word spotting. During operation of aforementioned DAS, word segmentation is a necessary process to obtain segmented words from input HWDs for subsequent stages. Word segmentation is a challenging task due to factors like cursive nature of free-handwriting, overlapping, non-straight baselines, document degradation, accents, punctuation marks, diacritic symbols, cluttering, irregular inter-line distance, non-uniform intra-word, and inter-word distances (Fernández-Mota *et al*., 2014; Louloudis *et al*., 2009; Huang and Srihari, 2008). These factors cause segmentation errors in many instances with overlapping and cluttering being the most difficult to address. Cluttering occurs when words/lines are very close to each other which may be worsened by non-straight baselines. Overlapping occurs when part of a character of a word extends to a region of another nearby word (figure 1d) or when characters of adjacent words touch (figure 1b(ii&v)) or cross one another (figures 1a(i & iii) and 1b(i & ii)). Crossing words (figures 1a(i & iii) and 1b(i & ii)) are most challenging to segment without under/over-segmentation because at the cross point (where strokes from different words meet), strokes from different words share pixels and also extend into another's region as seen in figure 1e. Many of the existing word segmentation techniques perform well in HWD with well-spaced words/characters but their performances dive in cases of crossing words (Pal and Datta, 2003). Since most HWD contain crossing words, a technique to efficiently segment such crossing words without over-segmentation (figure 1c(i)) or under-segmentation (figure 1c(ii)) is necessary. It is for this reason that this paper proposes a novel offline word segmentation technique called component tracing and association (CTA) for segmenting words in HWDs. With CTA overlapping and crossing words are efficiently segmented.

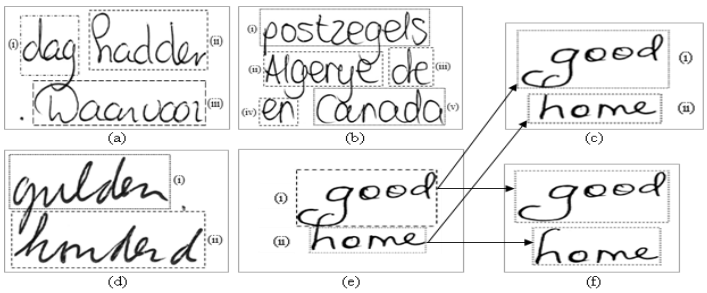

Fig 1 Core word segmentation of handwritten words

In CTA technique, HWD is first segmented to text lines by projection profile method. In each line segment, gaps between adjacent CCs are modelled to intra/inter-word gaps by modelling them using multi-variate Gaussian mixture model (GMM) using 3 metric distances: hull distance, bound box distance, and principal hull distance. This is followed by clustering adjacent CCs with intra-word gaps between them to get core word segments (CWS). CWS form large and essential portion of target words. Other portions extending outside are traced and joined with original CWS to form full words. Crossing strokes are separated using junction branch association (JBA) method (discussed in section 3.4). JBA is based on the principle that short sections on both side of a reference point (RP) on a continuous stroke are symmetric or almost symmetric w.r.t RP. Thus, crossing strokes can be efficiently separated.

In the remaining part of the paper, section 2 discusses related work in word-level segmentation, section 3 discusses the proposed technique for full word segmentation, section 4 discusses performance of the proposed word segmentation technique, and section 5 is conclusion.

2 RELATED WORK

Jindal and Jindal (2015) used mid points of white spaces between text lines to segment Gurmurkhi HWD to lines and words. 95% accuracy is reported. In the work of Jain et al. (2014), their word segmentation technique regarded text area as a large window, which is then divided into smaller windows of text lines. The text line windows are further divided to smaller windows of words. Karmakar et al. (2014) used inter-line/word spaces to segment HWD to lines and words. Louloudis et al. (2009) used Hough transform-based technique for line

segmentation and univariate Gaussian mixture model to cluster CCs in line segments of HWD to words. Yin and Liu (2009) used distance metric in word/line segmentation of Chinese HWD. Sharma and Dhaka (2020) used speeded up robust features (SURF) descriptors of connected components (CC) with support vector machine (SVM) for word segmentation of HWD. Fernández-Mota et al. (2014) modelled text document to having crests (text/foreground) and valleys (background or non-text empty spaces between words/lines). Lines are separated by optimal path going through valleys between crests (words). Rohini et al. (2012) applied threshold to run lengths to separate touching words from consecutive lines. In the work of Sanasam et al. (2020), words are segmented from local vertical projection profiles (VPP) of segmented lines. Text lines are segmented using local horizontal projection profile (HPP) of vertical strips. Patel and Desai (2010) also used projection profile-based approach for text segmentation. Mullick et al. (2015) in their work of segmentation of handwritten Bangla document images, separated touching words by using separation boundary obtained after thinning. Singh et al. (2016) has used Euclidean distance transform (EDT) for word segmentation of handwritten Bangla documents. The distance used is between foreground (text) pixel to nearest background pixel. Neche and Kacem-Echi (2019) used deep learning to segment Arabic scripts to lines and words. RU-Net was used for line segmentation. Text line segments were segmented to words by combining CNN and BLST (bi-directional long short-term memory) methods. Savitha et al. (2021) used text block characteristics to segment words in Tulu handwritings. Text-specific Text Refinement Network has been used by Xu et al. (2021) to segment words with unique shape and textural characteristics. In recent years, deep learning-based segmentation approaches have been used with very good performances those by Bonechi et al. (2020) for scene text segmentation, Fermanian et al. (2020) with Syriac document images, and Divya et al. (2020) used in Gujarat document images.

The segmentation methods discussed perform very well in HWD with well-spaced lines and words with minimal overlapping, but perform dismally in cases of crossing, overlapping, and

touching words/lines. In this paper, a novel CTA technique is proposed that addresses the mentioned challenges such as to efficiently segment full words as they are in the source documents including those parts crossing with strokes from adjacent words.

3 METHODOLOGY

In this section, the proposed word segmentation technique is discussed in detail. It consists of 3 main steps: (i) Line segmentation (section 3.1), (ii) Core word segmentation (section 3.2), and (iii) Full word segmentation (section 3.5). Figure 2 shows the framework of the proposed word segmentation technique. Component tracing and association (CTA) method is used for full word segmentation as will be explained in section 3.5.

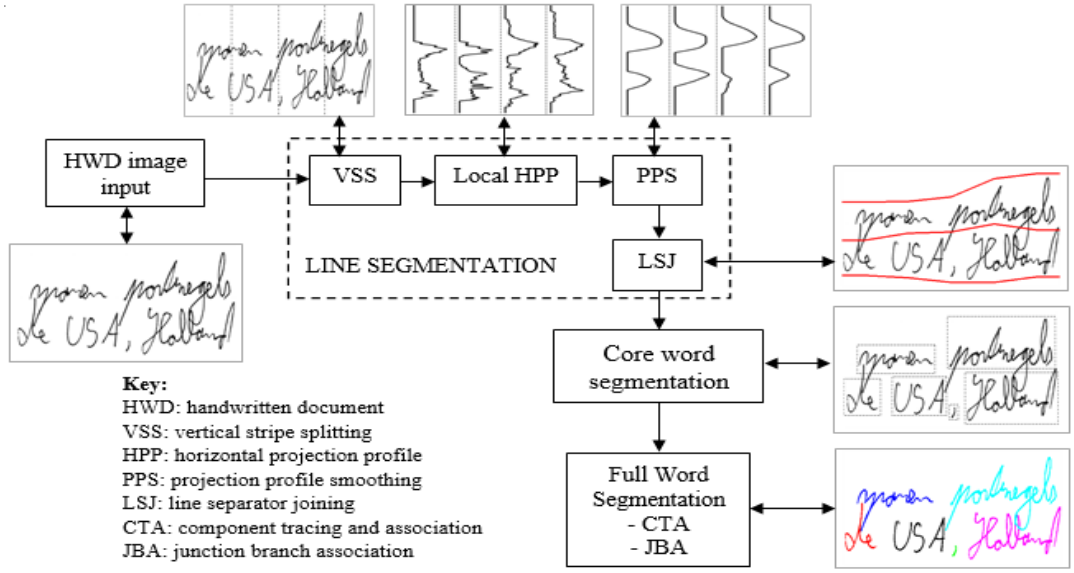

Fig 2 Framework of the proposed word segmentation technique

3.1 Line segmentation.

This step consists of 4 sub-steps by which document image is split into text line: (i) vertical stripe splitting (VSS), (ii) Local horizontal projection profile (HPP) computation, (iii) Projection profile smoothing (PPS), and (iv) Line separator joining (LSJ). The I_{bin} (input binary document image) is first split into vertical stripes in a process called VSS. Local HPP of the 1st stripe is then computed. In PPS step, local HPP is smoothed as weighted summation of N profiles in the neighborhood of a given raw profile (P_i) as shown in equation 1. PPS is improvement of method used by Papavassiliou et al. (2010) for better results.

$$SP_i = \sum_{j=-N}^{N} d_{i+j} w_j P_{i+j} \quad (1)$$

Where $d = 1$ if $P_i \geq P_{th}$ and 0 otherwise, w_j are weights computed using equation 2.

$$w_j = exp\left(-\frac{|j|^2}{2|j|+1}\right) \quad (2)$$

Where N is the number of profiles/steps in the neighbourhood of current profile (P_i), $j = \{-N, -N+1, \ldots\ldots, N\}$ and d determines if a given row participates (has text) or is ignored since it is marginal (i.e., are empty rows or have very few text pixels). Weights w_j (equation 2) have better and smoother exponential decay with distance away from current profile P_i, better than those used by Papavassiliou et al. (2010). This brings out well text lines and inter-line valleys hence easily identified as shown in figure 3a(ii) for blue/dashed plot. This is further refined by obtaining first derivative of the smoothed profiles (SP_i), estimated using equation 3.

$$\Delta SP_i(j) = \frac{1}{2h+1} \sum_{x=1}^{h} \left(SP_i(j+x) - SP_i(j-x) \right) \quad (3)$$

Where h is near odd integer to half of mean height of all CCs in the HWD. Values of $\Delta SP_i(j) < 0$ are replaced with 0 so that text boundaries and their attributes are well brought out. This helps to iron out problem of false local extremas which characterize a similar approach by Papavassiliou et al. (2010). In a plot of $\Delta SP_i(j)$ (figure 3a-iii), there are three main local turn points that make a recurring sequence, i.e., L_u, L_p, and L_d. L_u is 1st local minima denoting valley points (spaces between lines having few/no foreground pixels), L_p is local maxima denoting a point where headline (upper bound of middle zone of a word), and L_d is 2nd local minima denoting a point where principal text line passes, that is, a line passing through center of middle zone of a CC/word. Consecutive L_u points form text line separators as shown in figures 3a(iii-iv). The same is repeated for all stripes to obtain their respective line separators. In Line separator joining (LSJ) step, corresponding line separators of adjacent vertical stripes are joined to complete line segmentation as shown in figure 3b.

3.2 Core word segmentation

In this step adjacent CCs in a line segment are clustered based on inter-CC gaps to obtain core word segments (CWS). Inter-CC gaps are categorized to intra/inter-word gaps by modelling

them using bi-variate Gaussian mixture model (GMM) with expectation maximization method (Chen and Gupta, 2010) using 3 metric distances for gaps: hull distance (d_h), bound box distance (d_b), and principal hull distance (d_{hp}). A gap g_i is represented by the 3 gap distances, $g_i = \{d_h, d_b, d_{hp}\}$. Distances h_d and h_b are obtained as explained by Huang and Srihari (2008) and Mahadevan and Nagabushnam (1995). d_{hp} is a newly proposed gap metric which is distance between points where principal lines of 2 adjacent CCs (or words) intersect their respective convex hulls as shown in figure 3c. Principal line is one that passes through the middle of central zone of a CC/word. d_{hp} factors in non-uniform base lines of HWDs.

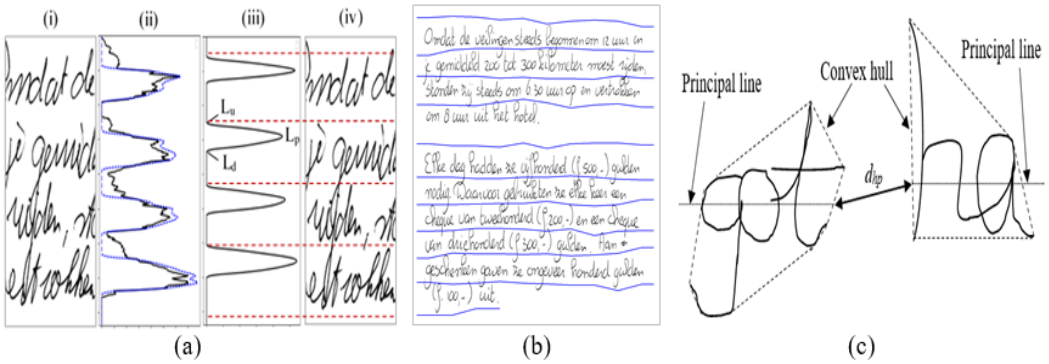

Fig 3 (a) Locating text line boundaries of HWD: a(i) is document stripe, a(ii) raw (black/full line) and smoothed (blue/dashed line) horizontal profiles of a(i), a(iii) first derivative of smoothed profile, and a(iv) stripe with line boundaries. (b) is HWD showing line segmentations. (c) principal hull distance(d_{hp}) metric.

Equation 5 is probability density function used in modelling inter-CC gaps as bi-variate GMM.

$$P(g_i|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(g_i - \mu)^T \Sigma^{-1}(g_i - \mu)\right) \quad (5)$$

Where g_i is vector of i^{th} gap metrics as mentioned before, $D = 3$ is dimension of data, Σ is a $D \times D$ covariance matrix for g_i of gaps in entire text document, and $\mu = \{\mu_h, \mu_b, \mu_{hp}\}$ is vector with μ_h, μ_b, μ_{hp} being means of d_h, d_b, d_{hp} respectively. During modelling, parameters obtained are mixing coefficients $\Pi = \{\Pi_1, \Pi_2\}$, cluster means $\mu = \{\mu_1, \mu_2\}$, and cluster covariances $\Sigma = \{\Sigma_1, \Sigma_2\}$ which are associated with each Gaussian (for inter/intra gap). These parameters are used to compute posterior probabilities (assignment scores) (r_k) of a gap g_i belonging to both the clusters/Gaussians using equation 6.

$$r_k(g_i) = \frac{P(k)P(g_i|k)}{P(g_i)} = \frac{\Pi_k P(g_i|\mu_k,\Sigma_k)}{\sum_{j=1}^{K} \Pi_k P(g_i|\mu_j,\Sigma_j)} \quad (6)$$

A gap g_i is assigned to a cluster/Gaussian where it gets largest assignment score/posterior probability. Adjacent CCs with intra-word gaps between them are grouped together, otherwise they belong to different words. The clustered CCs are referred to as core word segment (CWS). CWS may or may not amount to a full target word. CWS can either be full CSW (FCWS) or partial CWS (PCWS). FCWS is one in which contains all parts of a target word only and has no parts from other words like in figures 1a(ii) and 1b(iii). PCWS is one has some parts of target word left-out as in figures 1a(i), 1b(ii), and 1d(i). PCWS are further processed for segmentation of full target word(s) by CTA process (discussed in section 3.5). CTA employs junction branch association (JBA) and multi-dimensional dynamic time warping with dependence (MD-DTW$_D$) to handle junctions and crossings between strokes of different words/CSW. Therefore, these dependencies (MD-DTW$_D$ and JBA) are first discussed.

3.3 Multi-dimensional dynamic time warping with dependence (MD-DTW$_D$)

This is a technique of comparing multi-dimensional sequences say, $A = \{a_{i,k}\}$ and $B = \{b_{j,k}\}$ where $i = 1,2,\ldots m$ and $j = 1,2,\ldots n$ are indices of elements in A and B respectively, $k = 1,2,\ldots L$ is index of dimension of either of sequences, where *m* & *n* need not be equal. *L* is number of dimensions in each of the sequences *A* and *B.* It is assumed that dimensions of each sequence are inter-dependent. In MD-DTW$_D$, a matrix of *L1* distances *D(i,j)* between datapoints in *A* and *B* is first computed using equation 7 (Shokoohi-yekta et al., 2017).

$$D(i,j) = d(a_i, b_j) + min[D(i-1,j-1), D(i-1,j), D(i,j-1)] \quad (7)$$

Where $d(a_i, b_j) = \sum_k^L |a_{i,k} - b_{j,k}|$, $i = 1,2,\ldots\ldots m$ is i^{th} datapoint in *A*, $j = 1,2,\ldots\ldots n$ is j^{th} datapoint in *B*, and $k = 1,2,\ldots\ldots L$ is k^{th} dimension. table of accumulated cost, *Cost(i,j),* is computed from distance table *D(i,j)* using equation 8 (Shokoohi-yekta *et al*., 2017).

$$\text{Cost}(i,j) = \text{Cost}(i,j) + min[\text{Cost}(i-1,j-1), \text{Cost}(i-1,j), Cost(i,j-1)] \quad (8)$$

Warping cost (*Wcost*) between sequences *A* and *B* is computed by equation 9.

$$Wcost = \frac{Cost(m,n)}{m*n} \quad (9)$$

Where *m* and *n* are respectively lengths of sequences *A* and *B*.

3.4 Junction branch association (JBA) method

This method is used to segment crossing strokes from different words/CCs as shown in figures 5(a&b) by identifying junction branches belonging to same strokes (e.g., AJ & JB in figure 5b).

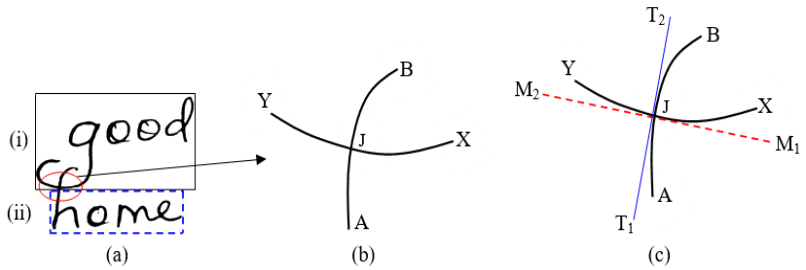

Fig 5 (a) Crossing words with a cross point circled (in red), (b) Cross point (J) shown for words in (a) constituted by 4 junction branches: AJ, JB, XJ, and JY, (c) Crossing with tangent (T_1T_2) at J and mirror line (M_1M_2) for branch AJ belonging to word a(ii)

Let (x_j,y_j) and (x_i,y_i) respectively be coordinates of cross point J and various points on junction (or candidate) branches (figures 5(b&c)). Using AJ as search branch, proceed as follows:

(i) Obtain tangent of AJ at *J* (T_1T_2 in figure 5c). Compute gradient (g_t) of the tangent T_1T_2.

(ii) Find mirror line M_1M_2 (figure 5c) as line perpendicular to T_1T_2 and passing through *J*.

(iii) Flip *AJ* about M_1M_2 to get mirror image A'J' such that for every object point (x_i, yi) on *AJ*, its mirror point (x_m,y_m) about M_1M_2 on A'J' is computed using equations 9 and 10.

$$x_m = \frac{2y_i - x_i(g_m - g_t) - 2c_m}{g_m - g_t + \varepsilon} \quad (9)$$

$$y_m = y_i - g_t(x_m - x_i) \quad (10)$$

Where x_i, y_i, and g_t, assume initial meanings, g_m is gradient of M_1M_2, c_m is y(row) intercept for M_1M_2, and ε is regularization value (very small positive value) that prevents division by zero.

(iv) Sequence of coordinates of points on A'J' is compared with sequences of coordinates of points on all other junction branches (JB, XJ, and JY) using MD-DTW$_D$ (section 3.3) and warping cost (equations 7-9) obtained in each case. This warping cost is the associativity score (A-Score). The smaller the A-Score, the more similar the two sequences are. Lowest A-Score is 0. A candidate junction branch with minimum A-Score and that is less than threshold (1) is

deemed to be an associate of AJ. The stroke terminates at the cross point if no associate branch is found. Associates of other junction branches are obtained in the same way.

3.5 Full word segmentation by Component tracing and association (CTA)

In this step, full words are completely segmented without over/under-segmentation. Figure 6 shows CTA framework where inputs are thinned HWD binary image (I_{th}) and CWS_i of a given text line where $i = 1,2,\ldots,N$ is index of CWS in a segment text line. CTA consists of CSW_i classification, CC tracing, and JBA (figure 6). For CWS of a text-line, proceed as follows:

(i) CWS_i is categorized to either PCWS or FCWS (section 3.2) by CWS-classifier as follows: search for CCs in neighborhood of a region occupied CWS_i in I_{th} which are connected to CWS_i by 8N connectedness. Presence of such CCs means CWS_i is PCSW, otherwise it's FCSW.

(ii) If CWS_i is a FCWS, it is regarded as full segmented word. Go back to step (ii) with next CWS (i.e., CWS_{i+1}). If CWS_i is a PCWS, it means it has some portions belonging to it that are left out. Proceed to next step in order to search for the left-out portions.

(iii) Using 8N connectedness approach, foreground pixels connected to CWS_i (in I_{th}) are traced out, a process called CC tracing. If trace-path terminates with no junction/crossing encountered during CCT, the trace path (consisting of traced-out pixels of left-out portion belonging to CWS_i) is de-skeletonized and then joined to CWS_i to form a full segmented word. Go back to step (i) with CWS_{i+1}. If a junction/crossing is encountered, proceed to next step.

(iv) Apply JBA technique (section 3.4) to the crossings. If the trace path terminates at the cross point, it is de-skeletonized and then joined to CWS_i to form a full segmented word, then go back to step (i) with CWS_{i+1}. If trace path proceeds beyond cross point, associate junction branch to trace-path up to cross point is identified, and then go back to step (iii).

Steps (i-iv) are repeated for all CWS in all text lines to obtain full words from entire HWD.

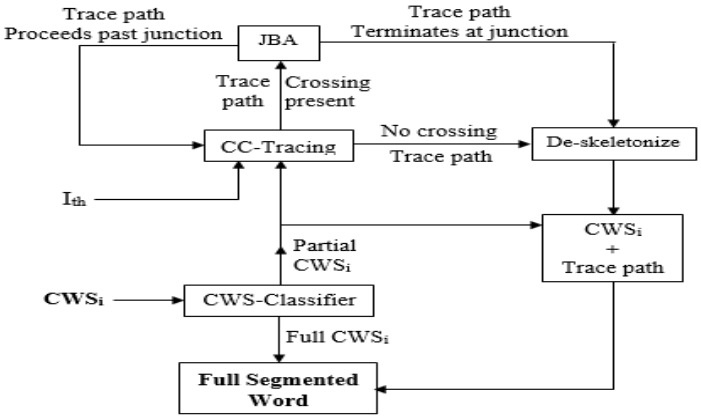

Fig 6 CTA framework for full word segmentation with inputs being thinned HWD image (I_{th}) and CSW_i outputting a full segmented word.

4 RESULTS AND DISCUSSION

Performance of the CTA technique has been qualitatively and objectively evaluated with the publicly available ICDAR2009 (Gatos *et al*., 2011; Gatos *et al*., 2009) and ICDAR2013 (Stamatopoulos *et al*., 2013) handwriting datasets. The datasets consist of scanned images of handwritten scripts in different language scripts like English, French, Germany, Indic (of Indian origin like Telugu, Bengali etc). Objective evaluation is based on detection rate (DR), recognition accuracy (RA), and performance metric (PM) (Gatos, *et al*., 2011; Gatos, *et al*., 2009). The larger the DR, RA and PM values, the better the technique. Figure 8a shows segmentation results of the proposed CTA technique for Latin-like and non-Latin-like text blocks obtained from ICDAR2009 dataset. Efficient segmentation of overlapping, touching, and crossing words seen is due to the JBA method that capable of identifying a word's stroke at crossing/touching points, which is a challenge in other techniques.

The proposed CTA method attained 98.56%, 97.89%, and 98.22% respectively for DR, RA, and PM metrics with ICDAR2009 dataset. For ICDAR2013 dataset, the proposed method attained overall scores of 99.14%, 98.02%, 98.58% for DR, RA, and PM metrics respectively. The good performance is attributed to 2 main reasons: (i) GMM-based modelling of inter-CC gaps (section 3.2) to inter/intra word gaps that leads to efficient clustering of CCs to words especially for HWDs with well-spaced words and text lines, (ii) JBA method (section 3.4)

helps to efficiently segment crossing and touching words by tracing and identifying strokes of a words that cross with strokes of adjacent words. By CC tracing, a word's strokes extending to 'territories' of other words are obtained and associated with the word they duly belong.

The proposed method was also evaluated with the two categories of HWDs obtained from ICDAR2009 and ICDAR2013 datasets based on language writing system: Latin/Latin-like texts (LLT) and non-Latin/Latin-like texts (non-LLT). LLT include English and French HWDs and non-LLT include Indic, Chinese, Arabic HWDs. Figure 7 shows performance of CTA method for the 2 categories where it is seen that performance in non-LLT is less than that of LLT. This is due to diacritic marks present in non-LLT as compared to LLT, causing decrease in performance especially when diacritic marks are near or connected to CWS of a target word.

Performance of CTA method has been compared with state-of-art techniques using DR, RA, and PM metrics as shown in table 1. Performance metrics for methods by Dahake et al. (2017), Sharma and Dhaka (2016), Jain and Singh (2014), Karmakar et al. (2014), and Chaudhuri and Pal (1997) for ICDAR2009 and ICDAR2013 datasets have been obtained from Sharma and Dhaka (2020). Performance of run length smoothing algorithm (RLSA) (Konidaris *et al*., 2007) has been obtained from its implementation by Gatos et al. (2011). From table 1 results, CTA method outperformed other methods showing that it is efficient in word segmentation.

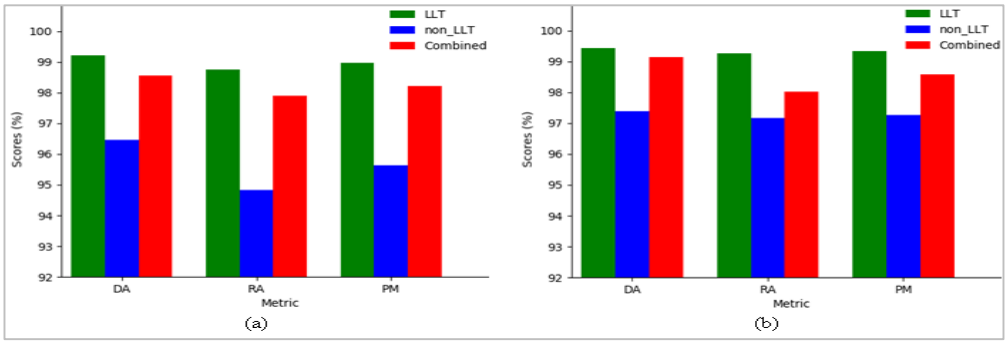

Fig 7 Performance of CTA method for LLT and non-LLT categories for (a) ICDAR200 (Gatos *et al*., 2009; Gatos *et al*., 2011), and (b) ICDAR2013 (Stamatopoulos *et al*., 2013) datasets.

Table 1 Comparison of segmentation performances with ICDAR2009 and ICDAR2013

Method	ICDAR2009			ICDAR2013		
	DR	RA	PM	DR	RA	PM
Jindal & Jindal (2015)	88.31	90.98	89.62	93.67	93.98	93.78
Sharma and Dhaka (2016)	87.93	88.37	88.15	94.37	94.38	94.77

Jain and Singh (2014)	90.50	91.55	91.03	94.25	94.98	94.61
Karmakar et al. (2014)	87.86	86.91	89.62	89.62	84.45	86.96
Dahake et al. (2017)	87.82	91.85	83.16	92.77	92.99	91.68
Chaudhuri and Pal, 1997	83.55	89.29	90.71	90.62	89.45	89.96
RLSA (Konidaris et al. 2007)	80.78	77.68	79.20	-	-	-
ILSP-LWSeg-09 (Gatos et l.,2011)	95.16	94.38	94.77	-	-	-
Sharma and Dhaka (2020)	96.32	95.74	95.72	98.32	96.74	95.99
Proposed	**98.56**	**97.89**	**98.22**	**99.14**	**98.02**	**98.58**

CTA method was also evaluated with 400 crossing words cropped from HWDs from FireMaker (Schomaker and Vuurpijl, 2000), ICDAR2009, and ICDAR2013 datasets. The test words have crossings of various kinds as shown in figure 8b. To the best knowledge of authors so far, there is no publicly available dataset of handwritten crossing and overlapping words. Figure 8b rows (i-iii) shows some of the crossing words that are efficiently segmented by the proposed CTA technique. This is because JBA method (section 3.4) is efficiently traces & identifies strokes of a word that crosses with those of other words. As can be seen from table 2 that JBA achieves 97% accuracy in segmenting crossing words. JBA method performs better in segmenting non-Latin crossing words (98% accuracy) as compared to Latin words (96.7% accuracy). This is because crossing of non-LLT texts is less complex as compared to that of LLT. JBA method couldn't do well in few cases where strokes of neighbouring words overlap such that a stroke segment of a word is erroneously assigned to another (figure 8b row iv). The proposed method is robust because of its CC-tracing and JBA approaches which are less computational compared to methods by Sharma and Dhaka (2020), Fernández-Mota et al. (2014), and Papavassiliou et al. (2010) that use computational approaches.

Table 2 Performance of JBA method in segmenting crossing words

	Word-pairs	Correctly segmented	Segmentation Accuracy (%)
LLT	300	290	96.7
Non-LLT	100	98	98.0
Combined	400	388	97.0

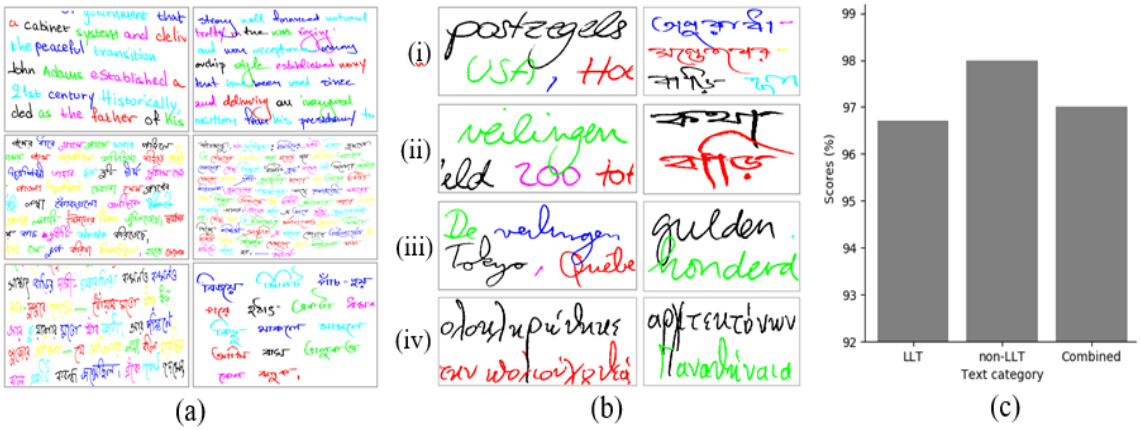

Fig 8: (a) HWD segmentation by the proposed technique, (b) segmentation of crossing words by JBA method, and (c) graphical representation of segmentation performance of JBA method.

5. CONCLUSION

A novel CTA method for word segmentation has been proposed. The method efficiently segments words with strokes overlapping, touching, and crossing with those of adjacent words. The method uses a novel JBA approach to separate crossing strokes from different words. The proposed word segmentation technique has been evaluated with ICDAR2009 and ICDAR2013 data sets of handwritten scripts posting detection rates of 98.56% and 99.14% respectively. JBA technique has been evaluated specifically on crossing words only from FireMaker dataset of handwritten documents attaining 97% segmentation accuracy.

REFERENCES

Bonechi, S., Bianchini, M., Scarselli, F., and Andreini, P. 2020. Weak supervision for generating pixel–level annotations in scene text segmentation. Patt. Recog. Letters 138:1–7.

Chaudhuri, B.B. & Pal, U. 1997. An OCR system to read two Indian language scripts: Bangla and Devnagari (Hindi). Proc. of 4th ICDAR, pp. 1011–1015.

Chen. Y. & Gupta. M.R. 2010. EM demystified: An Expectation-Maximization Tutorial. Technical Report, Department of Electrical Engineering, University of Washington.

Dahake, D., Sharma, R.K. & Singh, H. 2017. On segmentation of words from online handwritten Gurmukhi sentences. Proceedings of 2017 2nd Int. conf. on man and machine interfacing (MAMI).

Divya, B., Goswami, M.M., and Mitra, S. (2020). DNN based approaches for Segmentation of Handwritten Gujarati Text. IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC). Doi: 10.1109/iSSSC50941.2020.9358904

Fermanian, R., Yaacoub, C., Akl, A., & Petra Bilane, P. 2020. Deep Recognition-based Character Segmentation in Handwritten Syriac Manuscripts. Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA), IEEE.

Fernández-Mota, D., Lladós, J. & Fornés, A. 2014. A graph-based approach for segmenting touching lines in historical handwritten documents. IJDAR, 17:293–312.

Gatos, B., Stamatopoulos, N. & Louloudis, G. 2011. ICDAR2009 handwriting segmentation contest. Int J. Doc. Anal Recognit (IJDAR) 14(1):25–33.

Gatos, B., Stamatopoulos. N. & Louloudis, G. 2009. ICDAR2009 handwriting segmentation contest. Proceedings of ICDAR pp 1393-1397.

Huang, C. & Srihari, S.N. 2008. Word Segmentation of Off-line Handwritten Documents. Proceedings of SPIE - The International Society for Optical Engineering 6815:68150.

Jain, S. & Singh, H. 2014. A novel approach for word segmentation in correlation based OCR system. Int J Comput Appl. 99(18):12–20.

Jindal, P. & Jindal, B. 2015. Line and word segmentation of handwritten text documents written in Gurmukhi script using mid-point detection technique. International Journal of Advance Research in Science and Engineering 4(1):11-19.

Karmakar, P., Nayak, B. & Bhoi, N. 2014. Line and Word segmentation of a printed text document. Int J. Comput. Sci. Inf. Technol. 5(1):157–160.

Konidaris, T., et al., 2007. Keyword-guided word spotting in historical printed documents using synthetic data and user feedback. Int. J. Document Anal. Recognit. **9**(2-4):167-177.

Louloudis, G., Gatos, B., Pratikakis, I. & Halatsis, C. 2009. Text line And Word Segmentation of Handwritten Documents. Pattern Recognition 42(12):3169-3183.

Mahadevan, U. & Nagabushnam, R.C. 1995. Gap Metrics for Word Separation in Handwritten Lines. Proceedings of 3rd ICDAR.

Mullick, K., Banerjee, S. & Bhattacharya, U. 2015. An efficient line segmentation approach for handwritten Bangla document image. Proceedings of 8th ICAPR, pp. 1-6.

Neche, C., Belaïd, A. & Kacem-Echi, A. 2019. Arabic Handwritten Documents Segmentation into Text-lines and Words using Deep Learning. Proc. of ICDAR Workshops, pp.19-24.

Pal U. & Datta, S. 2003. Segmentation of Bangla unconstrained handwritten text. Proceedings of *7th ICDAR* pp. 1128-1132.

Papavassiliou, V., Stafylakis, T., Katsouros, V. & Carayannis, G. 2010. Handwritten document image segmentation into text lines and words. Pattern Recognition 43:369-377.

Patel, C. & Desai, A. 2010. Segmentation of Text Lines into Words for Gujarati Handwritten Text. Proceedings of International Conference on Signal and Image Processing, pp. 130-134.

Rohini, S., Uma, D.R.S. & Mohanavel, S. 2012. Segmentation of Touching, Overlapping, Skewed and Short Handwritten Text Lines. Int. J. of Computer Applications 49(19):24-27

Sanasam, I., Choudhary, P. & Singh, K.M. 2020. Line and word segmentation of handwritten text document by mid-point detection and gap trailing. Multimedia Tools and Applications, 79:30135–30150.

Savitha, C. K., Ujwal, U. J., & Smitha, M. L. 2021. Detection of Single and Multi-character Tulu Text Blocks. IEEE Int. Conf. on Mobile Networks and Wireless Comm. (ICMNWC).

Schomaker, L. & Vuurpijl, L. 2000. Forensic Writer Identification: A Benchmark Data Set and a Comparison of Two Systems, Technical Report, NICI, Nijmegen.

Sharma, M.K. & Dhaka, V.P. 2016. Pixel plot and trace based segmentation method for bilingual handwritten scripts using feedforward neural network. Neural Comput. Appl., 27(7):1817-1829.

Sharma, M.N. & Dhaka, V.S. 2020. Segmentation of handwritten words using structured support vector machine. Pattern Analysis and Applications. 23:1355-1367.

Shokoohi-Yekta, M., Hu, B., Jin, H., Wang, J. & Keogh, E. 2017. Generalizing Dynamic Time Warping to the Multi-Dimensional Case Requires an Adaptive Approach. Data Min Knowl Discov. 31(1):1–31.

Singh, P.K., Sinha, S., Chowdhury, S.P., Sarkar, R. & Nasipuri, M. 2016. Word Segmentation from Unconstrained Handwritten Bangla Document Images using Distance Transform. Comput. Commun. Technol. pp. 473–484.

Stamatopoulos, N., Gatos, B., Louloudis, G., Pal, U. & Alaei, A. 2013. ICDAR 2013 handwriting segmentation contest. Proceedings of 12th ICDAR, pp 1402–1406.

Xu, X., Zhang, Z., Wang, Z., et al. 2021. Rethinking Text Segmentation: A Novel Dataset and A Text-Specific Refinement Approach. IEEE CVPR. Pp. 12040-12050.

Yin, F. & Liu, C.L. 2009. Handwritten Chinese text line segmentation by clustering with distance metric learning. Pattern Recogn. 42(12):3146-3157.