A Novel Geometrical Scale and Rotation Independent Feature Extraction Technique for Multi-lingual Character Recognition

Narasimha Reddy Soora[1]
Associate Professor, Computer Science and Engineering, Kakatiya Institute of Technology and Science Warangal, Telangana, India

Ehsan Ur Rahman Mohammed[2]
Student, Computer Science and Engineering, Kakatiya Institute of Technology and Science Warangal, Telangana, India

Sharfuddin Waseem Mohammed[3]
Assistant Professor, Computer Science and Engineering, Kakatiya Institute of Technology and Science Warangal, Telangana, India

Abstract—This paper presents a novel geometrical scale and rotation independent feature extraction (FE) technique for multi-lingual character recognition (CR). The performance of any CR techniques mainly depends on the robustness of the proposed FE methods. Currently, there are very few scale and rotation independent FE techniques present in the literature which successfully extract the robust features from characters with noise such as distortion and breaks in the characters. Many FE methods from the literature failed to distinguish the characters which look similar in their appearance. So, in this paper, we have proposed a novel scale and rotation independent geometrical shape FE technique which successfully recognized distorted, broken, and similarly looking characters. Aside from the proposed FE technique, we've used crossing count (CC) features. Finally, we have combined the proposed features with CC features to make as Feature Vector (FV) of the character to be recognized. The proposed CR technique is evaluated using publicly available media-lab license plate (LP), ISI_Bengali, and Chars74K benchmark data sets and achieved encouraging results. To further assess the performance of the proposed FE method, we've used a proprietary data set containing nearly 168000 multi-lingual characters from English, Devanagari, and Marathi scripts and achieved encouraging results. We have observed better classification rates for the proposed FE method using publicly available benchmark data sets as compared to few of the CR FE methods from the literature.

Keywords—Feature extraction; character recognition; crossing count features; edit distance; scale and rotation independent feature extraction

I. INTRODUCTION

We the human beings have the beautiful ability to recognize the text present in all sorts of forms such as those printed in different font styles, handwritten, sloppy, and inclined, which are camouflaged with the background, possessing variations in illumination and brightness, of varying sizes, occluded ones, from various viewpoints, written upside down, having characters with missing parts, unwary decorations and marks, broken or even misspelled, having artistic and figurative designs, and many more. It comes as no surprise that the creative computer vision (CV) community despite six and more decades of intensive research could not achieve much in making computers represent images and perform well-defined white-box generic and low-level

processes (ANNs are algorithms belonging to the category of the black box) on images thereby making computers capable of detecting and recognizing texts robustly in a way much similar to that of the humans along with many other activities performed by human visual cortex such as classification, and alike.

The class of computer algorithms performing text recognition is known as Optical Character Recognition (OCR). The path tread for image formation and analysis started with mechanical means, followed by optical means, and now uses the concepts of digital representation and processing, has led many works to establish their base on those physical phenomena such as inertia and concepts such as moments which is descriptive of the physical body, etc. Working on the similar lines there is literature which takes heed from physical and biological concepts in order to describe the image using its features, different features being responsible to obtain robustness from different kinds of variations and thereby incorporating as many features as possible gives less space to the algorithms to make mistakes and makes the FV generic. Less computation cost and fastness are also key concerns in image processing (IP) as images are an inherently large collection of data, so care must be taken in making FV's less redundant and compact.

OCR is the most important to the real-world document analysis and storage tasks [10], and it has been used even in technologies that are dealing with automation as in the tasks of natural language processing (NLP) such as machine translation, data mining, along with tasks such as indexing, word spotting, LP recognition, signature verification in banks, sorting postcards based on address, reading aid for blind, etc. FE techniques reduce the dimensionality of the image to be recognized and thereby making the recognition process computationally efficient and mathematically feasible. These features are then checked for similarities with an abstract vector representing a character. General FE techniques from CV are all applicable to an OCR and such OCRs are commonly seen in "intelligent" handwriting recognition and indeed most modern OCR software utilizes such FE techniques which are more based on learning rather than hard-coded mathematics. In the deep learning era, were the internal formulation of ANNs has worked out ways to extract and

learn from features much in a biological way. But robustness could be enhanced to ultra-fine intricacies of data for proper categorization and decision making can be learned if traditional CV and deep learning could go hand-in-hand.

As per the paper [16], there are very few robust scale and rotation independent FE techniques present in the literature which can recognize normal and similarly looking characters from various multi-lingual languages and this inspired us in proposing a novel scale and rotation independent geometrical FE technique for multi-lingual CR and it takes inspiration from some more related, empirically well-established scale and rotation invariant FE techniques, which are briefly summarized in the next section. Most of the existing CR algorithms [16] used a combination of FE techniques from the literature which are elaborated in the paper [16].

The remaining sections of this paper are organized as follows. Section II describes related work from the literature; Section III describes the detailed proposed CR technique; Section IV describes experimental results and discussion and Section V is about conclusion.

II. RELATED WORK

Despite many years of advancement in CV and machine learning, CR is challenging till today [13] because of varying complexity of characters in the form of character graphemes of various languages worldwide, presence of broken, distorted, rotated, varying sized and similarly looking characters. We have observed there are very few scale and rotation independent FE techniques for multi-lingual CR. In this section we will discuss a few of the existing FE techniques (mostly scale and rotation independent) for CR from the literature.

K. Sampath et al. in the paper [1] proposed a feature extraction technique for character recognition using combination of existing features such as histogram oriented Gabor features, grid level features (local gradient), and gray level co-occurrence matrix and reported a success rate of 96% using Chars74K data set. The concept of calculating moments has been central to some IP tasks [2] and applications like pathological brain detection problems, etc. Many moments like Hue, Zernike, and pseudo-Zernike moments (orthogonal radial moments) are invariant to rotation and scale with the help of few geometric transforms. Zernike Moments have the least redundancy of information and hence are less susceptible to noise in the image and it also has better numerical stability. Paper [2] used the magnitude of Zernike moments as rotation invariant features for the classification of grayscale face images and binary character images and reported an accuracy of 99.7% using the Roman proprietary data set of 1560 characters.

Authors in the paper [3] uses a traditional approach wherein a covariance matrix is constructed from a set of rotated versions of each character and an Eigenspace is derived from the matrices obtained, a locus is constructed by projecting the respective rotated characters onto their Eigen sub-space a part of the actual Eigenspace obtained from all the categories. Recognizing a character is done by simply projecting it onto the Eigen sub-spaces and measuring the

distance between the projected points and the locus present in each Eigen sub-space. The problem which needs to be acknowledged is that some characters form similar types of locus and hence more rigorous testing needs to be done while recognizing such characters, sometimes this can make the computation cost high due to suggestive interpolation that needs to be applied during the formation of the loci. As this method is dependent on training samples, the samples need to be selected carefully and the number of training samples must also be high for more accurate and precise locus. A strange observation was that few symmetric characters were recognized correctly despite the angle of rotation being wrongly interpreted. The authors have tested the proposed method using a proprietary data set containing 2808 characters with different orientations and reported an accuracy of 99.89%.

T. Hayashi et al., in the paper [4] divided the input character into sub-level patterns which can be mapped to elementary components. The set of sub-patterns derived are used for the task of recognition and this division is based on cross points, angle point's and the system is free from scale and rotation variations as the classification is based on the combination of elementary components which these characters are composed of. The authors in the paper [4] reported 100% accuracy on the proprietary Arabic numerals data set of size 150. For texture images, rotation invariant representation is possible by using dominant orientation which is the orientation with highest total energy across different scale considered during image decompositions as in [6] and finally, rotation-invariance is obtained by circularly shifting the elements of FV within the same given set of scales so that the first elements found at each scale shall have maximum correspondence with the dominant orientations. Representation with the highest total energy across the different orientations (dominant scale) results in the scale-invariant one. The feature alignment process to classify texture is based on the assumption that the images should be rotated so that their dominant orientations/scales are the same. It has been proved that the image rotation in the spatial domain is equivalent to a circular shift of feature vector elements. This paper reported an average recognition accuracy of 98.89% using four image data sets from the Brodatz database.

K. U. Rehman et al. in the paper [7] proposed a feature extraction technique for character recognition using existing moment based features such as raw moments, central moments, hu moments, and Zernike moments and reported a success rate of 96.922% using Urdu proprietary data set. L. A. Torres-Méndez et al., in the paper [8] presented a translation, rotation, and scale invariant method for object recognition by extracting topological object characteristics with the help of novel coding of the normalized moment of inertia. They have tested the proposed method with 238 proprietary images and reported 98% accuracy. Paper [9] presents a rotation and scale invariant multi-oriented CR technique wherein a given character has divided multiple circular zones and each zone is divided into three centroids through the combination of the segments obtained from its constituent character into two clusters and one being the global centroid of the segmented

character. The ordering of centroids as per their distances from the global centroid makes the farther one to be included in one set and another one to be included in the next set and this is crucial to the construction of rotation invariant FV. Observations are that the relative positions and structure is unaltered after clustering is performed and reported the highest accuracy of 99.01% and 99.25% using Bangla and Devanagari data sets of 7874 and 7515 characters respectively.

Parul Sahare et al., in the paper [10] proposed a set of FE techniques for CR based on geometrical properties of characters. The first set of proposed features is based on adaptive center distance based on Euclidean distance from the centroid of each non-overlapping block to each foreground pixels. The second set of features is based on fixed center triangular cut based features for each non-overlapping block is computed. The last set of features is based on neighborhood count based features with the help of a window of size 3x3. The generated features are combined to form a feature vector of a character. The authors in the paper [10] reported an average recognition accuracy of 98.56% using the Chars74K data set containing alphabets and numerals. Rina D. Zarro et al., in the paper [11] proposed a hybrid algorithm using Hidden Markov Model and harmony search algorithm for online Kurdish CR. Authors have classified a group of characters into smaller subgroups based on directional features with the help of Markov model. The small number of group of characters is fed to a harmony search algorithm that uses a common movement pattern for recognition. The proposed system was tested using a proprietary data set having 4500 words structured with 21234 Kurdish characters, and reported an accuracy of 93.52%.

R. P. Kaur et al. in the paper [12] proposed a feature extraction technique for Gurumukhi characters recognition using existing techniques such as zoning features, diagonal features, and parabola curve fitting based features and reported an accuracy of 96.19% using 1605 Gurumukhi characters. J. Chaki et al. in the paper [19] proposed a framework to classify fragmented handwritten digits into three classes based on geometrical functions, grading scheme and fuzzy rules. Authors in the paper [13] have proposed robust geometrical FE techniques for the license plate (LP) CR. They have proposed geometrical shape FV generation using horizontal and vertical scan lines and angular width FV generation using horizontal, vertical, right diagonal, and left diagonal scan lines. They combined these two FVs along with the crossing count FV to form the FV of a character. Authors have tested the proposed FE method with the help of publicly available MediaLab benchmark LP database [5] containing 741 images with 6584 characters (English alphabets and numerals) extracted from these 741 images using LP detection method proposed in the paper [14] and reported an accuracy of 98.8% at the character level.

N. R. Soora et al., in the paper [15] proposed two novel FE techniques namely shape geometry encoding of components of characters with the help of perpendicular distances and encoding of triangular areas computed using four scan lines namely horizontal, vertical, right diagonal and left diagonal scan lines. Authors have tested the proposed method with the help of the MediaLab LP benchmark database [5] containing 6584 characters (English alphabets and numerals) and reported an accuracy of 99.03%. The authors have tested the proposed method using proprietary data sets containing nearly 30000 characters (Devanagari, Marathi, and English alphabets) and reported an accuracy of 98.5%. A good list of FE techniques could be found at [16] and some key points from the authors are provided here. The features which do not have discriminating capabilities to classify an input character when considered alone are called non-shape-based features and are used to eliminate false hits or pooled with other features to recognize characters that look similar in shape. Statistical methods are not generic because they generally involve threshold values that have to be set by the programmers and are not learned or adjusted according to variations in data and the tasks which are to be performed on the given image (data). So, generally statistical methods are not used due to their inflexibility and a great deal of trial and error involved to reach sufficient or good performance.

Tian et al., in the paper [17] proposed two FE techniques for multi-lingual scene CR using co-occurrence of HOG. The first FE technique is based on co-occurrence HOG (Co-HOG) in which authors have encoded co-occurrence of oriented pairs of neighboring pixels. The next FE technique is based on the convolution of Co-HOG (ConvCo-HOG) in which authors have extracted Co-HOG features from all possible images. The performance of the method was tested using five charter data sets in which the ISI_Bengali data set is publicly available and they have reported an accuracy of 92.2% on the ISI_Bengali data set. U Pal et al., in the paper [18] proposed a FE technique for multi-oriented and multi-sized CR using a contour distance based approach. In this paper, the authors have extracted the features by finding the distances from the centroid of the character to the contour points of the characters. They rearranged the extracted features in such a way that the FE is size and rotation invariant. Authors have reported an accuracy of 97.8% and 98.1% using proprietary 2900 Bangla characters and proprietary 3100 Devanagari characters respectively. Authors of the paper [19] tested the proposed method using MNIST, Numta, and Devanagari numerical data bases and reported good recognition accuracies.

We have considered 30000 characters extracted from 280 aged multi-lingual Indian documents having English, Marathi, and Devanagari scripts to assess the performance of the proposed algorithm and achieved a success rate of 98.8% which is almost equivalent the success rate of 98.5% reported in [15]. The advantage of the proposed system as compared to [15] is that it is scale and rotation independent. To test the scale and rotation independent factor of the proposed method, we have generated 168000 characters from proprietary test set of 12000 characters that are different in orientation and sizes and achieved nearly 98% accuracy using the manually generated proprietary data set. As it is not justifiable to compare the performances using proprietary databases, we have considered publicly available MediaLab LP, Chars74K, and ISI_Bengali benchmark data sets to compare the performance of the proposed method with methods from literature and explained the same in detail in the experimental results section.

III. PROPOSED WORK

In this section we have explained in detail about the proposed novel geometrical scale and rotation independent (SRI) FE technique for multi-lingual CR. We have extracted crossing count (CC) features along with the proposed SRI FE techniques and we have combined both SRI and CC features to form the FV of the character.

A. Scale and Rotation Independent Feature Extraction Generation

SRI features are generated with the help of sweep lines which will pass through the centroid of the input character. At first, remove all the background pixels in all directions of the input character so that the input character will properly fit into a rectangular box as shown in Fig. 1. Find the centroid of the character and let it be C(X_c, Y_c). Next, find the boundary points of the input character by traversing in 8-directions from the centroid C. Let N (≤ 8) be the number of boundary points that we encounter while traversing from centroid C. Find the distance Di (for i = 1 to N) from centroid C to the N boundary points using Equation (1) where Pi(X_i, Y_i) be any general point on the boundary of the input character. Find a point Pi(X_i, Y_i) which is closest to centroid C as shown in Fig. 2. Find the slope m of the line joining the points C and Pi using Equation (2). Find the angle θ between the line joining CPi and x-axis using the Equation (3).

$$D_i = (X_c - X_i)^2 + (Y_c - Y_i)^2 \tag{1}$$

$$m = ((Y_c - Y_i) / (X_c - X_i)) \tag{2}$$

$$\theta = (180/\pi) * \tan^{-1}(m) \tag{3}$$

Extend the line CPi from both sides of the centroid in the direction of slope m up to the boundary of input character using the formula's (4) and (5) to find all points (Xnew, Ynew) on the line by varying the distance d value from centroid C (X_c, Y_c) as shown in the Fig. 2. (Xnew, Ynew) are points which are the combination of (Xnew1, Ynew1) and (Xnew2, Ynew2). (Xnew1, Ynew1) are the set of points of the line CPi which reside on one side of the centroid C and (Xnew2, Ynew2) are the set of points of the line CPi which reside on the second side of the centroid C.

$$X_{new} = X_c \pm \left(d / \sqrt{(1 + m^2)} \right) \tag{4}$$

$$Y_{new} = Y_c \pm \left((m * d) / \sqrt{(1 + m^2)} \right) \tag{5}$$

Collect all distinct points P_{new}(X_{new}, Y_{new}) generated using the Equations (4) and (5) which lie on the line CP_i with slope m as shown in the Fig. 3. At this stage, we have all the points lying on the line CP_i with slope m. Now, find all the starting and ending cut points (from centroid C) from the list of the points computed, which are foreground of the input character that CP_i intersects with each connected component. Let the points of intersection or the cut points of the line CP_i with foreground boundary points of the input character be P_k(X_k, Y_k) as shown in Fig. 4. Find the distances from centroid C to all P_k's using the Equation (1). Preserve all the computed

distances in a separate list called DISTANCES. Now change the angle θ that sweep line intersects with x-axis by 2 degrees and find the slope of the new sweep line using the Equation (3) with new θ computed. Generate all the points of the new sweep line passing through centroid C with slope m using Equations (4) and (5) as explained previously and find all the cut points of that the new sweep line that intersects with the foreground boundary points of the input character. Compute the distance from centroid C to the new cut-points of the sweep line with each of the connected components at the boundary points of the input character. Again, preserve the computed distances in the DISTANCES list. Repeat the steps of finding the distances from the centroid to the cut points for each new sweep line formed by varying θ as shown in Fig. 4.
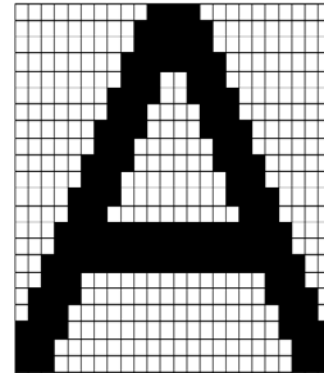

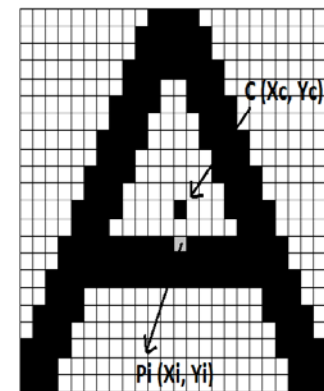Fig. 1. Original Image Fitted in a Rectangular Box by Removing Background Pixels from All Directions.

Fig. 2. Example Figure Showing Centroid C (X_c, Y_c) and Nearest Point P_i (X_i, Y_i) on the Boundary.
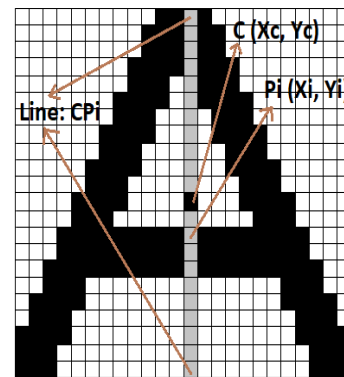

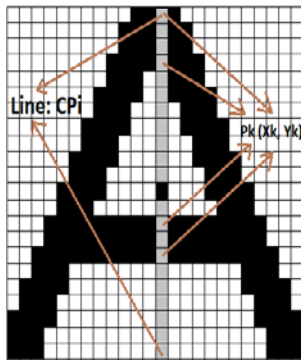Fig. 3. Example Figure Showing Generation of the Sweep Line CPi.

Fig. 4. Example Figure Showing Generation of Foreground Boundary Points $P_k (X_k, Y_k)$ of Input Character Intersection with Sweep Line CP_i.

This process has to be repeated for 90 times (as we are increasing the θ value each time by 2 degrees) so that one end of the initial sweep lines reaches to its opposite position. At this stage, we have all the distances (preserved in Distance list) from centroid C to the cut points of all sweep lines with all the probable connected components of the input character at the boundary points. Normalize the distances present in the DISTANCE list by finding the maximum distance of all the distances in the DISTANCE list and divide all the distances with maximum distance using the Equation (6). The resultant DISTANCE list will contain the values in the range from 0 to 1. Encode DISTANCE list values ranging from 0 to 1 using the Equation (7) which results in SRI features of the input character.

The described FE method extracts the shape of any input character in the form of shape symbols built using Equation (7) with the help of normalized distances from centroid to the boundary points of the input character where each sweep line intersected. The proposed FE method does not use any existing boundary extraction algorithms. To find the next boundary point, we have not traversed through the next available boundary point. The proposed SRI FE method extracts the complete set or subset of the boundary points with the help of sweep lines which retains the shape of the input character. This capability of the proposed FE technique extracts the robust features from the input character even in the presence of the noise and is independent of size, rotation, distortion, and breaks of the input character. It has the ability to distinguish similarly looking characters as mentioned in [16]. The detailed description of the algorithm is shown in Table I.

$$DISTANCE = DISTANCE / \max(DISTANCE) \tag{6}$$

$$SRIF = \begin{cases} A, & D_i \geq 0.0 \ and \ D_i < 0.1 \\ B, & D_i \geq 0.1 \ and \ D_i < 0.2 \\ C, & D_i \geq 0.2 \ and \ D_i < 0.3 \\ D, & D_i \geq 0.3 \ and \ D_i < 0.4 \\ E, & D_i \geq 0.4 \ and \ D_i < 0.5 \\ F, & D_i \geq 0.5 \ and \ D_i < 0.6 \\ G, & D_i \geq 0.6 \ and \ D_i < 0.7 \\ H, & D_i \geq 0.7 \ and \ D_i < 0.8 \\ I, & D_i \geq 0.8 \ and \ D_i < 0.9 \\ J, & D_i \geq 0.9 \ and \ D_i \leq 1.0 \end{cases} \tag{7}$$

TABLE I. ALGORITHM TO GENERATE SRI FEATURES FROM AN INPUT CHARACTER

1. Remove background pixels from all directions of the input character and the resultant input character looks as shown in the Fig. 1.
2. Find the centroid C (X_c, Y_c) of the input character.
3. Move from centroid C in 8-directions to find boundary pixels of the input character and let N (≤ 8) be the number of boundary pixels.
4. Compute the distance from C to the N boundary points of the input character using Equation (1).
5. Find a boundary point Pi (Xi, Yi) which is closest to the centroid C and let it be P_i (X_i, Y_i) as shown in Fig. 2.
6. Find the slope m of the sweep line joining C and P_i using Equation (2).
7. Find the angle θ between the sweep line CP_i and x-axis using Equation (3).
8. Find all the distinct points $P_{new1}(X_{new1}, Y_{new1})$, $P_{new2}(X_{new2}, Y_{new2})$ lying on both side of the centroid C of the sweep line CPi as shown in Fig. 3 till the borders of the input image using Equations (4) and (5) by changing the d value. The value of d indicating how far we are moving from centroid C of the input character. P_{new1} are the set of points lying on one side of the centroid C of the sweep line CP_i and P_{new2} are the set of points lying on second side of the centroid of the sweep line CP_i. The set of points P_{new1} and P_{new2} are useful in generating crossing count features.
9. Find the cut points or intersection points that the P_{new1} and P_{new2} making with each connected components of the input character at the boundary points as shown in Fig. 4. Let the intersection or cut points be $P_k(X_k, Y_k)$.
10. Find the distances from centroid C to all P_k's and store them in a separate list called DISTANCES.
11. At this stage we have to find new sweep line. To find new sweep line, increase the value of θ by 2 degrees. Find the new slope m of new sweep line using updated θ with the help of the Equation (3).
12. Repeat steps from 8 to 12 for 90 iterations (which results into moving one side of a sweep line to its opposite side) to compute the distances from centroid C of the input character to all the cut points generated by all the sweep lines.
13. At this stage, we have the shape of the input character in the form of distances from centroid C to the boundary pixels of the input character.
14. Find maximum distance from the set of distances present in DISTANCE list.
15. Divide all the distances present in DISTANCE list with maximum distance. After this step, DISTANCE list contains the values in the range from 0 to 1.
16. Encode the normalized distances present in DISTANCE list using Equation (7).
17. Now, the DISTANCE list contains shape of the input character in the form of shape symbols present in the Equation (7). This DISTANCE list is SRI features.

Let Nnew1i, Nnew2i be the number of connected components for each of the set of points Pnew1i, Pnew2i respectively for sweep line i. The total number of SRI features generated is given by the Equation (8).

$$No.of \ SRI \ features = \sum_{i=1}^{90} \left(2*(N_{new1i} + N_{new2i})\right) \tag{8}$$

B. Crossing Count Features Generation

CC features are generated during the SRI features generation. In Table I step 8, we have set of pixels Pnew1 and Pnew2 on either side of the centroid C of any sweep line CPi.

Using Pnew1 and Pnew2 information, find the count of continuous foreground pixels which indicates the number of connected components on both sides of the centroid C. If there are no foreground pixels, take count as '0'. Store the connected components computed from Pnew1 and Pnew2 into a new list called CCFE. Repeat the process of finding the connected components for each sweep line. Let CCnew1i, CCnew2i be the CC features generated using Pnew1, Pnew2 set of pixels respectively which lie on either side of the centroid C of the sweep line CPi. CC features are generated using the Equation (9) shown below. The steps described above generate CCF of size 180.

$$CCF = \left\{ CC_{new1i} , CC_{new2i} \right\}_{i = 1 \ to \ 90} \qquad (9)$$

Combine the proposed SRI features and CC features to form SRIF vectors. SRIF = {SRI, CCF}. The total size of SRIF can be found by summing up the values from Equations (8) and (9). The size of SRIF differs from one character to other character depending on the complexity of the input character.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

We have implemented the SRI FE technique discussed in the previous section using MATLAB 18a on the Intel Core i5 processor machine with 8 GB RAM. We have used the Edit Distance metric for classification because the FV we have generated is the collection of shape symbols from Geometrical SRI features and Statistical CC features which are generated using sweep lines. At first, we have a proprietary data set containing 30000 characters extracted from 280 aged printed multi-lingual Indian documents having Marathi, Devanagari, and English scripts [15]. Out of 30000 characters we have selected 18000 characters for training and 12000 characters for testing. We have stored all 18000 FV's generated using the proposed FE method in a flat file. For each test character, we have generated the proposed geometrical SRI FV and compared the test FV with all the FVs stored in the file. The character which generated minimum edit distance is considered as the classified input character.

With 12000 proprietary data set test characters, we have achieved 98.8% success rate which is almost equivalent to the performance of the method from [15]. To further assess the performance of the proposed system, we have used MediaLab LP benchmark data set having 6584 characters containing English alphabets and numerals and achieved 99.2% success rate which outperforms the performance of the methods from [13], [15], [18] and using ISI_Bengali character data set containing 19530 Bengali characters, achieved 97.72% success rate which outperformed the method proposed in [17].

Table II shows the performance comparison of the proposed method with a few of the methods from the literature that used similar publicly available benchmark data sets. As per the MediaLab LP benchmark data set, we have achieved 99.20% accuracy which outperformance the methods [13] (accuracy: 98.8%), [15] (accuracy: 99.03%), and [18] (accuracy: 95.4%). The disadvantage of the methods from [13], [15] is that they are not scale and rotation independent.

Apart from the MediaLab LP benchmark data set and ISI_Bengali data set, we have assessed the performance of the proposed method using the Chars74K benchmark data set containing Kannada and English alphabets and achieved 98.64% which is almost equivalent to 98.56% reported by the method from [10] and better than 96% reported by [1]. The disadvantage of the methods [10] and [1] is that they are not scale and rotation independent.

TABLE II. PERFORMANCE COMPARISON OF THE PROPOSED METHOD WITH FEW OF THE METHODS FROM THE LITERATURE THAT USED PUBLICLY AVAILABLE BENCHMARK DATA SETS

Reference	Name of the data set	Number of characters of data set	Recognition accuracy (%)
[13]	MediaLab LP benchmark data set	6584 characters	98.80
[15]	MediaLab LP benchmark data set	6584 characters	99.03
[18]	MediaLab LP benchmark data set	6584 characters	95.40
Proposed method	MediaLab LP benchmark data set	6584 characters	99.20
[13]	Chars74K benchmark data set	74107 characters	97.24
[15]	Chars74K benchmark data set	74107 characters	97.89
[10]	Chars74K benchmark data set	74107 characters	98.56
[18]	Chars74K benchmark data set	74107 characters	94.20
[1]	Chars74K benchmark data set	74107 characters	96.00
Proposed method	Chars74K benchmark data set	74107 characters	98.64
[13]	ISI_Bengali benchmark data set	19530 characters	94.45
[15]	ISI_Bengali benchmark data set	19530 characters	95.20
[17]	ISI_Bengali benchmark data set	19530 characters	92.20
[18]	ISI_Bengali benchmark data set	19530 characters	91.14
Proposed method	ISI_Bengali benchmark data set	19530 characters	97.72

Table III shows the performance comparison of the method [18] from the literature with the proposed method using publicly available benchmark data sets. We have implemented the method proposed in [18] to test its performance with a few of the benchmark data sets [7]. Even though we have implemented the method proposed in [18], the implementation may not meet the optimizations as per the expectations of the original author [16]. From Tables II and III, it is very clear that the proposed method outperformed many methods from the literature which used publicly available benchmark data sets and proprietary data sets.

A recognized correctly as A | B recognized correctly as B | H recognized correctly as H | M recognized correctly as M

M recognized correctly as M | M recognized correctly as M | N recognized incorrectly as U | Z recognized correctly as Z

0 recognized correctly as 0 | 2 recognized correctly as 2 | 3 recognized correctly as 3 | 5 recognized correctly as 5

5 recognized incorrectly as W | 6 recognized correctly as 6 | 8 recognized correctly as 8 | 9 recognized correctly as 9

Marathi character recognized correctly as 'la' | Marathi character recognized correctly as 'bhuu' | Marathi character recognized correctly as 'a' | Marathi character recognized correctly as 'pa'

Kannada character recognized correctly as 'a' | Kannada character recognized correctly as 'aa' | Kannada character rotated by 180° recognized correctly as 'ka' | Kannada character recognized correctly as 'kha'

Fig. 5. Recognition Results of Few of the Characters from Various Languages by the Proposed Method.

Fig. 5 shows a few of the example CR results by the proposed method. To test the scale and rotation invariance of the proposed method, we have generated manually the characters from MediaLab LP benchmark data set and

proprietary data sets using software to rotate the characters with 45°, 90°, 135°, and 180° and resized the set generated with two different sizes. As the FE method in [18] is invariant to scale and rotation, we have compared the performance of the method from [18] with the proposed method using the manually generated characters. We have generated 168000 characters from 12000 proprietary characters containing Marathi, Devanagari, and English scripts. Total we have 180000 characters as part of the proprietary data set. Similarly, we have generated a total of 92176 characters from 6584 characters of MediaLab benchmark LP data set. So, we have a total of 98760 characters as part of the MediaLab benchmark data set.

Table III shows the performance comparison of the proposed method with the method from [18] using newly generated data sets. It is very clear from Table III, that the scale and rotation invariance of the proposed method outperformed the method from [18]. The disadvantage of the method from [18] is that, it is sensitive to the breaks present in the characters. The disadvantage of both the methods (proposed and [18]) is that, if the input character distorted in shape completely, both methods fail to classify the input character. Examples of such characters '5' and 'N' are shown in Fig. 5. Such distorted characters are recognized properly by the methods proposed in [13] and [15]. The advantage of the proposed method is that, it is invariant to scale and rotation. Another advantage of the proposed method is that it was able to classify the similarly looking characters such as {"0", "o", "D", "O"}, {"Z", "z", "2"} and {"8", "B"} from English alphabets and numerals and other scripts as well. Table IV shows the performance comparison of the proposed method with few of the methods from the literature. Table IV gives complete details of the feature extraction methods from the literature such as the number of characters used in the data sets, the language of the data sets, proposed features, whether the data set is benchmark data set or proprietary data set, whether the feature extraction is scale and rotation invariant or not, and recognition rate. Authors of the papers [4], [3] reported 100%, 99.89% accuracies respectively but the authors have not used publicly available benchmark data sets for evaluation purpose. Table IV clearly shows that the proposed method outperformed the methods from literature which used publicly available benchmark data sets for performance evaluation.

TABLE III. PERFORMANCE COMPARISON OF THE PROPOSED METHOD WITH THE METHOD FROM [18] USING MANUALLY GENERATED CHARACTERS FROM VARIOUS DATA SETS

Reference	Name of the data set	Number of characters generated manually from actual data set	Recognition accuracy (%)
[18]	proprietary data set	180000 characters	93.20
Proposed method	proprietary data set	180000 characters	98.00
[18]	MediaLab LP benchmark data set	98760 characters	92.55
Proposed method	MediaLab LP benchmark data set	98760 characters	98.25

TABLE IV.   PERFORMANCE COMPARISON OF THE PROPOSED METHOD WITH FEW OF THE METHODS FROM LITERATURE

Reference	Number of characters/images/videos	Language & Hand written / Printed characters	Proposed features	Benchmark data set / proprietary	Independency of FE (Scale, Rotation)	Recognition Rate (%)
[4]	150 images	Arabic Numeral & Printed characters	Shaped based local features (Elementary Components of the input character)	Proprietary	Scale and Rotation Invariant	100
[9]	7874 characters	Bangla & Printed characters	Shape based local features (circular zones)	Proprietary	Scale and Rotation Invariant	99.01
[9]	7515 characters	Devanagari & Printed characters	Shape based local features (circular zones)	Proprietary	Scale and Rotation Invariant	99.25
[6]	Non-character (Texture) Data base	NA	-	Benchmark data set (Non characters)	Scale and Rotation Invariant	98.89
[2]	1560 characters	Roman & Printed characters	Shape based (Moment Based Features)	Proprietary data set	Rotation Invariant	99.70
[8]	238 images	English Alphabets and Numerals & Printed characters	Shape based (Moment Based Features)	Proprietary data set	Rotation and Scale Invariant	98.00
[13]	6584 characters	English Alphabets and Numerals & LP characters (Printed)	Shape based (Geometrical shape encoding and Angular Width features)	MediaLab LP benchmark data set	None	98.80
[15]	6584 characters	English Alphabets and Numerals & LP characters (Printed)	Shape based (Geometrical shape encoding of perpendicular distances and triangular area calculations)	MediaLab LP benchmark data set	None	99.03
[15]	30000 characters	Devanagari, Marathi and English Alphabets & Printed characters	Shape based (Geometrical shape encoding of perpendicular distances and triangular area calculations)	Proprietary data set	None	98.50
[3]	2808 characters	English Alphabets & Printed characters	Shape based (Eigen space)	Proprietary data set	Rotation Invariant	99.89
[11]	21234 characters	Kurdish Alphabets & Hand written	Shape based (Movement patterns of characters)	Proprietary data set	None	93.52
[10]	74107 characters	Kannada and English Alphabets & Printed and Hand written characters	Shape based (Adaptive center distance based features, Fixed center Triangular Cut-based Features, Neighborhood Counts-based Features)	Chars74K benchmark data set	None	98.56
[17]	19530 characters	Bengali Alphabets & Printed characters	Shape based (Various types of HOG features)	ISI_Bengali benchmark data set	None	92.20
[18]	2900 characters	Bangla Alphabets & Printed characters	Shape based (Contour distance based features which are invariant to scale and rotation but sensitive to distortion and breaks)	Proprietary data set	Scale and Rotation Invariant	97.80
[18]	3100 characters	Devanagari Alphabets & Printed characters	Shape based (Contour distance based features which are invariant to scale and rotation but sensitive to distortion and breaks)	Proprietary data set	Scale and Rotation Invariant	98.10
[18]	6584 characters	English Alphabets and Numerals & LP characters (Printed)	Shape based (Contour distance based features which are invariant to scale and rotation but sensitive to distortion and breaks)	MediaLab LP benchmark data set	Scale and Rotation Invariant	95.40
[1]	74107 characters	Kannada and English Alphabets & Printed and Hand written characters	histogram oriented Gabor features, grid level features (local gradient), and gray level co-occurrence matrix	Chars74K benchmark data set	None	96.00
[7]	37440 characters	Urdu ligatures	raw moments, central moments, hu moments, and Zernike moments	Proprietary data set	Scale and Rotation invariant	96.92
[12]	1605 characters	Gurumukhi characters	Zoning features, diagonal features, and parabola curve fitting features	Proprietary data set	None	96.19

Proposed method	30000 characters	Devanagari, Marathi, and English & Printed characters	Shape based (Geometrical distance based features which are invariant to scale, rotation, distortion, and breaks)	Proprietary data set	Scale and Rotation Invariant	98.80
Proposed method	6584 characters	English Alphabets and Numerals & LP characters (Printed)	Shape based (Geometrical distance based features which are invariant to scale, rotation, distortion, and breaks)	MediaLab LP benchmark data set	Scale and Rotation Invariant	99.20
Proposed method	19530 characters	Bengali characters & Printed characters	Shape based (Geometrical distance based features which are invariant to scale, rotation, distortion, and breaks)	ISI_Bengali benchmark data set	Scale and Rotation Invariant	97.72
Proposed method	74107 characters	Kannada and English Alphabets & Printed and Hand written characters	Shape based (Geometrical distance based features which are invariant to scale, rotation, distortion, and breaks)	Chars74K benchmark data set	Scale and Rotation Invariant	98.64

V. CONCLUSION

In this paper, we have proposed a novel geometrical scale and rotation independent FE technique for multi-lingual CR with the help of various sweep lines. Along with the proposed method we have generated CC features with the help of sweep lines. We have combined both the FE techniques to form as FV of the character to be recognized. The proposed FE technique recognized the multi-lingual characters with noise such as distortion and breaks as shown in Fig. 5. The proposed FE technique has the ability to recognize characters accurately which are similar in shape from various languages. It is evidence from the results of the Tables II, III, and IV that the proposed FE technique outperformed many methods from literature using proprietary data set and various publicly available benchmark data sets. The FE technique proposed in this paper works on any kind of script. The limitation of the proposed method is that, it failed to extract the SRI features properly if the image is too small to distinguish with naked eye.

We have observed that there is still lot of scope in proposing novel robust scale and rotation independent FE techniques for Omni-font character recognition [16] which can recognized similarly looking characters from various multi-lingual languages and can be trained using various types of neural networks which combines the traditional way of extracting the features and new way of training and testing the proposed methods.

ACKNOWLEDGMENT

REFERENCES

[1] A. K. Sampath, and N. Gothami, "Decision tree and deep learning based probabilistic model for character recognition," J. of Cent. South Univ., Vol. 24 , No. 12, pp. 2862−2876, Dec. 2017.

[2] C. Singh, E. Walia, and N. Mittal, "Rotation invariant complex Zernike moments features and their applications to human face and character recognition," IET Comput. Vis., Vol. 5, No. 5, pp. 255-265, Oct. 2011.

[3] H. Hase, T. Shinokawa, M. Yoneda, and C. Y. Suen, "Recognition of rotated characters by Eigen-space," Proc. of 7th Int. Conf. on Doc. Anal. and Rec., Edinburgh, UK, pp. 731-735, 2003.

[4] H. Takayuki, and T. Noboru, "A Consideration on rotation invariant character recognition," 2006 World Automation Congress (WAC), Budapest, Hungary, pp. 1-6, July 2006.

[5] I. E. Anagnostopoulos, "MediaLab license plate database," Multimedia Technology Laboratory, National Technical University of Athens, http: //www.medialab.ntua.gr/ research/ LPRdatabase.htm.

[6] J. A. Montoya-Zegarra, J. P. Papa, N. J. Leite, R. D. S. Torres, and A. X. Falc ˜ao, "Learning how to extract rotation-invariant and scale-invariant features from texture images," EURASIP J. on Adv. Signal Process, Vol. 2008, no. 1, pp. 1-15, Jan. 2008.

[7] K. U. Rehman, and Y. D. Khan, "A scale and rotation invariant Urdu nastalique ligature recognition using cascade forward backpropagation Neural Network," IEEE Access, No. 7, pp. 120648-120669, Aug. 2019.

[8] L. A. Torres-Méndez, J. C. Ruiz-Suárez, Luis E. Sucar, and G. Gómez, "Translation, rotation, and scale-invariant object recognition," IEEE Trans. on Syst., Man, and Cybern.—Part C: Appl. and Reviews, Vol. 30, No. 1, pp. 125-130, Feb. 2000.

[9] N. Tripathy, T. Chakraborti, M. Nasipuri, and U. Pal, "A scale and rotation invariant scheme for multi-oriented character recognition," Proc. Of 23rd Int. Conf. on Pat. Rec. (ICPR), Cancún Center, Cancún, México, pp. 4041-4046, Dec. 2016.

[10] P. Sahare, and S. B. Dhok, "Robust character segmentation and recognition schemes for multilingual Indian document images," IETE Tech. Review, Vol. 36, No. 2, pp. 209-222, Apr. 2018.

[11] R. D. Zarro, and M. A. Anwer, "Recognition based on online Kurdish character recognition using hidden Markov model and harmony search," Eng. Sci. and Technol. an Int. J., Vol. 20, No. 2, pp. 783-794, Apr. 2017.

[12] R. P. Kaur, M. Kumar, and M. K. Jindal, "Recognition of newspaper printed in Gurumukhi script," J. Cent. South Univ., Vol. 26, No. 9, pp. 2495-2503, Oct. 2019.

[13] S. Narasimha Reddy, and P. S. Deshpande, "Robust feature extraction technique for license plate characters recognition," IETE J. of Res., Vol. 61, No. 1, pp. 72-79, Dec. 2014.

[14] S. Narasimha Reddy, and P. S. Deshpande, "Color, scale, and rotation independent multiple license plates detection in videos and still images," Math. Prob. In Eng., pp. 1-14, June 2016.

[15] S. Narasimha Reddy, and P. S. Deshpande, "Novel geometrical shape feature extraction techniques for multilingual character recognition," IETE Tech. Review, Vol. 34, No. 6, pp. 612-621, Oct. 2016.

[16] S. Narasimha Reddy, and P. S. Deshpande, "Review of feature extraction techniques for character recognition," IETE J. of Res., Vol. 64, No. 2, pp. 280-295, Jul. 2017.

[17] S. Tian, U. Bhattacharya, S. Lu, B. Su, Q. Wang, X. Wei, Y. Lu, and C. L. Tan, "Multilingual scene character recognition with co-occurrence of histogram of oriented gradients," Patt. Recogn., Vol. 51, No. 3, pp. 125–34, June 2015.

[18] U. Pal, and N. Tripathy, "A contour distance-based approach for multi-oriented and multi-sized character recognition," Sadhana, Vol. 34, No. 5, pp. 755–765, Oct. 2009.

[19] J. Chakey, and N. Day, "Fragmented handwritten digit recognition using grading scheme and fuzzy rules," Sadhana, Vol. 45, No. 1, pp. 1-23, Aug. 2020.