International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 8 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT183844

Zone-Wise Segmentation and Lexicon-Driven Recognition for Printed Myanmar Characters

Chit San Lwin[1], Xiangqian Wu[2]
[1,2]School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, P. R. China
[1]Department of Mathematics, Monywa University, Monywa City, Sagaing Region, Myanmar
Corresponding Author : chitsanlwin.maths.mm@gmail.com

ABSTRACT
This paper presents a new segmentation and recognition algorithms for Myanmar script inputted from offline printed images. Zone segmentation considers horizontal and vertical zones; it is applied to segment letters according to their roles such as primary or peripheral characters. In doing so, statistical and structural features of segmented characters are explored and exploited in recognition process. Hidden Markov model is used for recognition of primary characters while Kohonen self-organization map is used for peripheral characters. The recognized characters by each model are then combined, and finally are recognized by k-nearest neighbors algorithm with the help of lexicon is composed of all common Myanmar characters. Our OCR system for Myanmar characters tested on a dataset that approximately contains 7560 compounded characters. From the results, our system achieves higher significant results both segmentation and recognition compared to the other contemporary Myanmar OCR's approaches.

Keywords: Character Segmentation, Hidden Markov Model, Self-organization Map, k-nearest Neighbors, Lexicon

I. INTRODUCTION

Character recognition from the printed or handwritten documents have already been an intensive research area in recent years. It is a fundamental and an essential process in intelligent and machine learning systems for automatic recognition and translation of text images via a robot's eye, mobile phone or other electronic devices. Due to OCR's complex structure and high computational demand, it is still challenging to develop a robust language independent offline and online recognition systems. In spite of the relatively mature stage of OCR in most widely used languages like English, Indian, Chinese, Arabic, etc., Myanmar (known as Burmese) language is still struggling for a robust OCR system.

The Myanmar script is a cursive language like Arabic, Persian and Urdu scripts, it has a unique character set and several of these characters are similar with different meanings. One major challenge here is that Myanmar OCR system has been greatly under-researched. Another challenge is that the unique features of the Myanmar script stands as one of the main unresolved problems in the literature of the Myanmar OCR system. As such, there is a demand for a considerable and significant improvement in Myanmar OCR research, in order to keep pace with today's device-oriented technologies without needs of human assistances. This paper therefore considers these challenges of Myanmar OCR system and proposes a new algorithm for segmentation and recognition of printed Myanmar script.

Zone-wise Segmentation: It segments Myanmar script based on two groups, namely, primary (for consonants) and peripheral (for vowels) characters. It later discriminates each peripheral character depending on their features such as statistical and structural features and puts them into corresponding groups for later recognition process.

Lexicon-driven Recognition: It recognizes the characters with two different models. The first model called hidden Markov model (HMM) is used to recognize primary characters while self-organization map (SOM) is trained to classify different peripheral characters. It then combines the partial recognized results from each model and compares the resulted compound character par rapport the characters from the lexicon with the help of k-nearest neighbors (k-NN) algorithm. In comparison, k-NN clusters the most similar groups with input character in k-clusters and finds the most similar character in it for resulted compound word.

Specific Feature Extraction: Due to the existence of specific character patterns in Myanmar script, necessary features for each character are extracted such as number and size of dots, place of dots, open-loops or close-loops, number of strokes, end-points, etc. These extracted features are exploited in all stages of OCR processes; segmentation, feature extraction and recognition.

The remainder of this paper is structured as follows. Section 2 presents literature review regarding OCR. The characteristic and peculiarities of Myanmar script and our OCR system splitting is discussed in Section 3. A detailed explanation of our proposed system is given in Section 4. In addition, the development processes and experimental results are presented in Section 5, and the paper is finally concluded in Section 6 with description of its limitation and prospective future works.

II. LITERATURE WORK

Segmentation in OCR system is generally classified into line, word and character segmentations. Line segmentation has reached a quite standard level that can successfully be used by several types of languages. It segments the lines from paragraph text. In word and character segmentations, there are slightly or specifically different approaches that are proposed by a number of research works relative to the languages they focus on.

Sahare et al [1] proposed a character segmentation algorithm for Latin and Devanagari scripts. They mainly considered structural properties of characters in finding primary segmentation path. The other overlapped and joined characters are observed by using graph distance theory and individually split them as separated characters. Afterwards, they validated the segmentation results with support vector machine (SVM) for accurate segmented results. Regard to recognition, they tried to recognize the characters using their three new geometrical shape-based features together with k-NN classifier.

The Indic handwritten character segmentation was performed by a study [2] with three horizontal zones segmentations using HMM model and SVM model. Whilst HMM is used for middle zone segmentation, SVM is used for other two zones. Water reservoir feature and widow-based feature called pyramid histogram of oriented gradient (PHOG) features are used in middle zone segmentation. They then combined with the partial recognition results produced by each zone and finally performed word level recognition.

Chinese character recognition system was proposed by Tao et al [3] introduced a new manifold learning algorithm for characters based on subspace learning algorithm, discriminative locality alignment (DLA) to find similar character groups for recognition of

input characters. They afterwards proposed a kernel version of their DLA algorithm, KDLA, by conducting principal component analysis (PCA). Another Chinese character recognition system applied convolutional neural networks (CNN) for offline handwritten OCR [4]. They proposed a global supervised low-rank expansion method and an adaptive drop weight (ADW) for speed and storage capacity of their nine network layers for recognition.

Zarro et al [5] proposed an online Kurdish characters recognition system using HMM model and harmony search. Their system firstly split the characters into different sub-groups based on common directional feature vectors. Markov model was then used in classifying each group of characters. After getting the candidate characters with their associated features, they were classified by harmony search recognizer. They highlighted in their paper that working with smaller groups reduces the processing time in later recognition process.

In accordance with the most popular approach in OCR system, HMM model is properly used in Indic scripts online recognition system [6]. In this study, the researchers presented two main techniques using HMM: lexicon driven and lexicon free for two Indic scripts, namely Devanagari and Tamil. The difference of two techniques, lexicon driven and lexicon free are dependent or independent of handwritten writing orders but similarly consideration in symbol representation in the lexicon as the sequence of symbol HMM.

The lexicon-based text recognition was also proposed by a research [7]. They investigated scene text recognition (STR) system to recognize the text from signboard, or anything that describes the text. This type of recognition is quite challenging due to variability of font size, position of visible parts, minimal language context, and unexpected and uncontrolled conditions. To solve them, they presented a probabilistic model for STR system to organize similarity, language properties and lexical decision by using sparse belief propagation, a bottom-up method for shortening messages to decrease the dependency between weakly supported hypotheses.

Premaratne et al [8] used lexicon-based Sinhala script recognition system with HMM. They proposed segmentation-free recognition method by using orientation features and linear symmetry. They exploited the advantages of lexicon to verify and correct false rejections, missing character positions from the recognition stage, and also optimized the accuracy of missing words to an acceptable level.

III. CHARACTERISTICS OF MYANMAR LANGUAGE

3.1 Myanmar Cursive Script Language

Myanmar language, also known as Burmese language, is the national language of Myanmar. There are approximately a hundred spoken languages in Myanmar due to existence of 135 distinct ethnic groups who are speaking their own languages in their regions. Amongst all, Myanmar language is an official language spoken by almost 44 million, primarily by Burma (Burman) people and related ethnic groups in Myanmar and neighboring countries, [9].

Myanmar language is one of Sino-Tibetan language groups and its alphabets are derived from a Brahmic and Kadamba-Pallava scripts. It is a tonal and syllable-timed language composed of subject-object-verb order in sentence structure. Myanmar language is cursively written from left to right without concept of lower and upper-case letters. There are basic 33 consonants, 16 vowels, 10 special characters and only two punctuation marks that act like comma (,) and full stop (.) illustrated in Fig. 1(a-

c, e). There are additional glyphs of Myanmar characters called double-layers characters showing in Fig. 1(d). These double-layer characters can be formed by placing similar or different consonants by layers. Though not all consonants can be use in this form, it is applicable to almost half of consonants. However, they cannot stand alone to represent a meaning of a word without combination with other consonants.

Traditionally, a Myanmar word has one or more consonants with zero or more vowels are separately or jointly together. In this paper, we regard consonants as either primary letters or peripheral letters depending on its position, whereas vowels always regard as peripheral letters. All of them are interchangeably termed as ligatures or characters in this paper.

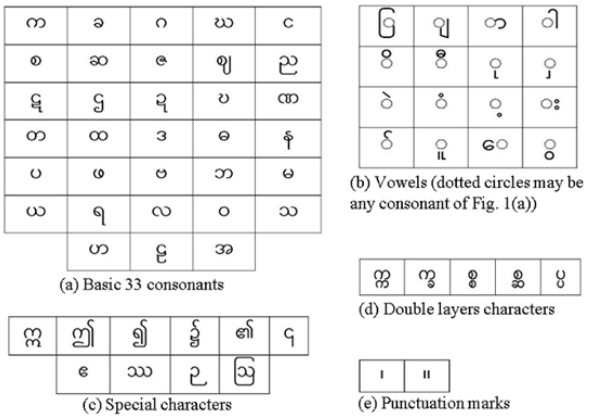

Figure 1. Basic characters in Myanmar script

3.2 Peculiarities of Myanmar Language

As mentioned earlier, Myanmar language is a cursive language like Urdu, Arabic, etc. Although character recognition process for those languages has already reached a mature level, otherwise, Myanmar OCR is still struggling to recognize all combined words due to large set of characters, complex combinations of consonants and vowels into one or more layers, other special characters and very similar characters in shape. This section discusses peculiarities of Myanmar script with pictorial representation in Fig. 2.

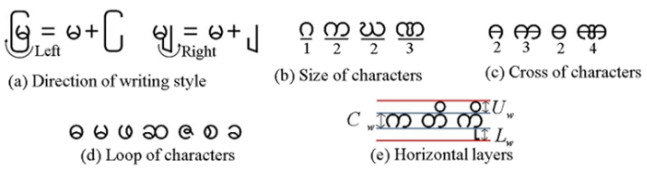

Figure 2. Characteristic of Myanmar ligatures

Cursive Style: Due to the delicate joint of letters in Myanmar script, it can be said that Myanmar language is cursive writing style. Myanmar ligatures are based on different sizes of close-circles, open-circles in different directions and combine them with straight lines, rounded corners, slope lines and dots. A word can be cursively organized with more than one ligature in different ways. A complete sample Myanmar sentence is demonstrated in Fig. 3.

Figure 3. A complete sentence in Myanmar script

Upper/Lower Case: There is no concept about letter casing in Myanmar language. Moreover, the sizes of letters, small or large have no meaning.

Space Usage: In English, space plays a key role in separating the words, but in Myanmar language, it is for separating different phrases. In formal writing such as news in newspaper, spaces are normally used between phrases. However, it is optionally except for official letters.

Number of Dots and Their Position: Dots play a key role in describing different meanings of a word in Myanmar script. They can be put into three places of a letter: upper, lower and right sides. They can be used alone in one place or together in some possible places as mentioned. Specifically, single dot is used at upper and lower places whereas double dots are used at the right side of a letter. Although they are generally used as dots, some Myanmar scripts use it as small circle without changing any meaning.

Direction of Writing: There are bi-directional writing styles in Myanmar language. Almost all words' directions lead from left to right in general. However, a few words are in opposite direction.

Circles in Different Sizes: Circles are very basic letters in Myanmar language. It can be used in all layers of horizontal and vertical places. Normal circles represent characters while small circles represent dots as above mentioned.

Loop: Myanmar script uses some loops inside a circle showing in Fig. 2(d).

Size and Cross: The size, the space taken by each primary letters varies depending on its structure. The size of a character can be known with its number of crosses as shown in Fig. 2(b, c).

Layers: There are three horizontal layers in Myanmar character. The outermost lines in Fig. 2(e) are for boundary of the characters. The central layer is to hold primary letters and other peripheral letters whereas the upper and lower layers are just for peripheral letters. According to the nature of Myanmar script, those three layers have same height such that height of C_w, L_w and U_w are equal. The concept of equal layers will be mainly used in segmentation process that will be discussed in next section. The combination of primary and peripheral characters of a word with their left to right sequences is demonstrated in Fig. 4.

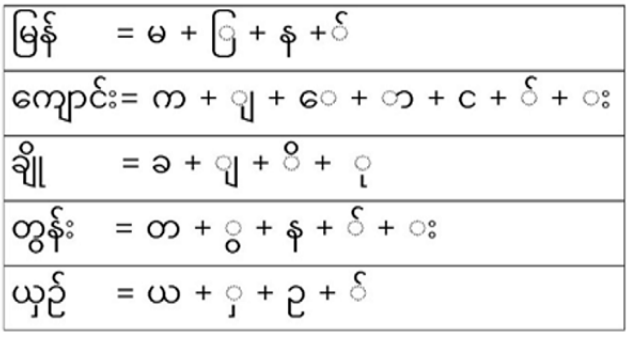

Figure 4. Writing sequence of Myanmar script (from left to right)

IV. OFFLINE PRINTED MYANMAR CHARACTERS RECOGNITION TECHNIQUE

This section elucidates the proposed system in details with its overall design and major components as schematically described in Fig. 5. The OCR system accepts input as scanned image files; high quality and high-speed scanners or other electronic devices like phone and cameras capture images. The preprocessing comes as the first phase of overall processes in order to smooth the images to be ready for segmentation and recognition process.

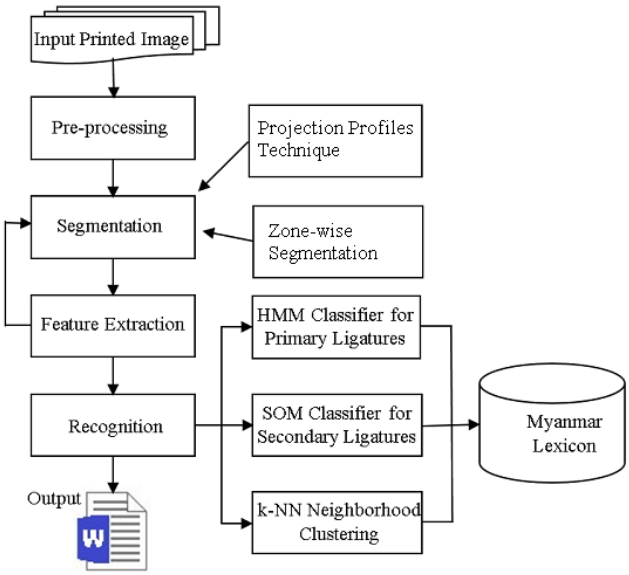

Figure 5. Overall design of proposed system

Unlike other OCR approaches, in our approach, the segmentation process performs zone-wise segmentation and separates the detected string of text into corresponding groups with the help of features obtained from feature extraction phase. There are two steps in recognition process for primary and secondary ligatures. HMM is used for primary ligatures recognition while SOM is exploited for recognition of peripheral ligatures that contain complex structures and features. As final step of our work, a set of accurately recognized characters is produced as text file.

4.1 Pre-processing

The preprocessing step is the first step of the OCR system. It includes the process of binarization, filtering, noise or outlier-removal, skew correction and baseline detection, etc. This step prepares an input image to be smooth for further recognition steps such as segmentation, feature extraction, etc. For a 2D gray scale input image, $im(x, y)$ that has a function of intensity value $f(x, y)$ for m to n pixel numbers for maximum row and column of image; binarization is executed to decrease the complexity for computations of OCR system. To remove irregular patterns such as disconnected parts between ligatures, morphological erosion is performed and thinning process is afterwards applied. After this steps, we get all input images have the proper orientation and are free of any skewness.

4.2 Segmentation

Offline printed or scanned document usually contains paragraphs composed of line-by-line sentences. Each sentence comprises a group of characters are partially or totally connected to each other. The segmentation technique is an essential step of OCR system. Its process is to divide the character strings into individual character in which ligatures may or may not be connected.

Before word segmentation, we perform line segmentation of the input scanned text files. Line segmentation is overcame with horizontal projection profile technique that separates paragraph into disjoint lines with upper or lower line. It finds the peaks and valleys between lines as the separators of the text lines. Therefore, we get individual lines from inputted paragraph text files. Not only horizontal but also vertical projection profile techniques accomplish perfectly for Myanmar script like doing English languages in word segmentation. Due to this processes, we also get word levels from individual lines. English language is also used vertical projection profile to get character strings by searching the space between characters. After that, they get completely character strings. However, vertical projection profile do not perfectly acquit to get character strings for Myanmar script because there is neither space usage between the characters nor definite ending characters of a word, that is, there may be different ending characters depending on combination of different ligatures as shown in Fig. 3 and Fig. 4. As a result, we get lacking character levels. This means that, some segmented characters are disjoint characters and joint ones. These facts, we must to divide the jointed characters until to achieve disjoint characters before character recognition step.

Separation of characters into primary or peripheral characters is a major task in cursive language segmentation unlike English language. Of the cursive languages OCR's such as Urdu [10] and Bangla [11], they used freeman chain codes (FCC), while other research [12] used trigram probabilities by normalizing over the number of ligatures and words in the sequence.

We observed a variety of segmentation methods in literature for different languages (especially cursive languages) due to their unique shapes and levels of structural complexity. In the light of this, we present a novel segmentation algorithm that fits the cursive Myanmar script as shown in Alg. 1.

Algorithm 1. H-zone segmentation

Input: $im(x, y)$: input image included a set of characters

f : statistical and structural features of a character

Output: $Hzone_i[] : i = 1, 2, 3$

$L = 3$ // for three horizontal layers

$mu_{x,y}[]$: middle-upper coordinates of a character existing in $Hzone_i[]$

$ml_{x,y}[]$: middle-lower coordinates of a character existing in $Hzone_i[]$

$lu_{x,y}[]$: left-upper coordinates of a character existing in $Hzone_i[]$

$ll_{x,y}[]$: left-lower coordinates of a character existing in $Hzone_i[]$

$ru_{x,y}[]$: right-upper coordinates of a character existing in $Hzone_i[]$

$rl_{x,y}[]$: right-lower coordinates of a character existing in $Hzone_i[]$

Algorithm:

1. $im[1]=$ partition $im(x,y)$ into three layers according to length $(im)/3$
2. for $i=1$ to L
3. $mu_{x,y}[i]=$ find uppermost mid-point of $im[i]$
4. $ml_{x,y}[i]=$ find lowermost mid-point of $im[i]$
5. $lu_{x,y}[i]=$ find leftmost uppermost point of $im[i]$
6. $ll_{x,y}[i]=$ find leftmost lower point of $im[i]$
7. $ru_{x,y}[i]=$ find rightmost uppermost point of $im[i]$
8. $rl_{x,y}[i]=$ find rightmost lower point of $im[i]$
9. $f=$ get structural and statistical information as straight line, corner detection (round or 90°), etc.
10. end for
11. for $i=1$ to $L-1$
12. if $ml_{x,y+1}[i]!=mu_{x,y}[i+1]$ && $im[i+1]$ does not contain features f then
13. $Hzone_{i+1}=$ partition lower zone from $im(x,y)$
14. $Hzone_i=$ partition central zone from $im(x,y)$
15. else
16. $im[i]=$ combine $im[i+1]$ to $im[i]$
17. end if
18. if $ll_{x,y+1}[i]!=lu_{x,y}[i+1]$ && $im[i+1]$ does not contain features f then
19. $Hzone_{i+1}=$ partition lower zone from $im(x,y)$
20. $Hzone_i=$ partition central zone from $im(x,y)$
21. else
22. $im[i]=$ combine $im[i+1]$ to $im[i]$
23. end if
24. if $rl_{x,y+1}[i]!=ru_{x,y}[i+1]$ && $im[i+1]$ does not contain features f then
25. $Hzone_{i+1}=$ partition lower zone from $im(x,y)$
26. $Hzone_i=$ partition central zone from $im(x,y)$
27. else
28. $im[i]=$ combine $im[i+1]$ to $im[i]$
29. end if
30. end for
31. produce $im[1]$, $im[2]$ and $im[3]$

According to the nature of Myanmar letters, there are three layers in equal height, viz. upper, central and lower layers as shown in Fig. 2(e). The letters are usually written within each layer depending on their roles. Every character in writing is started with a primary ligature that is always situated in central layer. Afterwards, the other ligatures are separately or jointly written in upper or lower layer by surrounding the primary ligatures.

In addition to traditional three horizontal layers in the most OCR works, we consider vertical layers

that play an important role to separate words into characters. In this case, we consider minimum one vertical layer and maximum four vertical layers that can probably have by a character. To give a clear representation of characters with their corresponding zone positions, a number of horizontal and vertical layers owned by each character is listed in Table 1.

Table 1. Different horizontal and vertical layers for different Myanmar characters

မြ	မြေ	မြော	မြောင်	မြောင်း
3HZ, 1VZ	3HZ, 2VZ	3HZ, 2VZ	3HZ, 3VZ	3HZ, 4VZ
မ	မိ	မု	မီ	မီး
1HZ, 1VZ	2HZ, 1VZ	2HZ, 1VZ	2HZ, 1VZ	2HZ, 2VZ
မှ	မှ	မူ	မူ	မ့ီး
2HZ, 1VZ	2HZ, 1VZ	2HZ, 1VZ	2HZ, 1VZ	3HZ, 2VZ
မျ	မျာ	များ	မျိ	မျိုး
2HZ, 1VZ	2HZ, 1VZ	2HZ, 2VZ	3HZ, 1VZ	3HZ, 2VZ

Horizontal Zone Segmentation **(H)**: Input of an algorithm is a scanned image released by preprocessing steps. The component $im(x, y)$; represents image array, the foreground of image is represented by $1s$ while $0s$ represent background pixels.

As aforementioned, Myanmar script falls into three horizontal layers with equal sizes. The primary letters generally fall into central zone by nature, whereas peripheral ligatures are in other zones. However, there are unusual primary and peripheral characters that take more than one zone. In Fig. 6(a, b), primary characters take both central and lower zones while characters in Fig. 6(c, d) are compound characters combined with primary and peripheral characters in more than one layer. Fig. 6(b) shows the primary characters, which is quite similar to primary characters in Fig. 6(c) but only difference in angle and direction features.

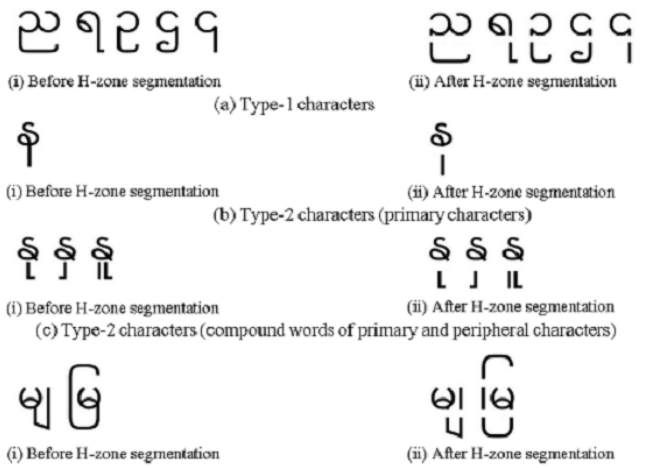

Figure 6. Unusual character in H zone-wise segmentation

The peripheral characters in Fig. 6(d) fall into all three zones although primary characters are in central zone. In this H-zone segmentation, we deal with all types of characters to segment correctly without unwisely separation of them. In order to do so, statistical and structural features are considered so that correct segmentation could be achieved depending on their total length, direction (left or right), angle detection (rounded or 90° etc.).

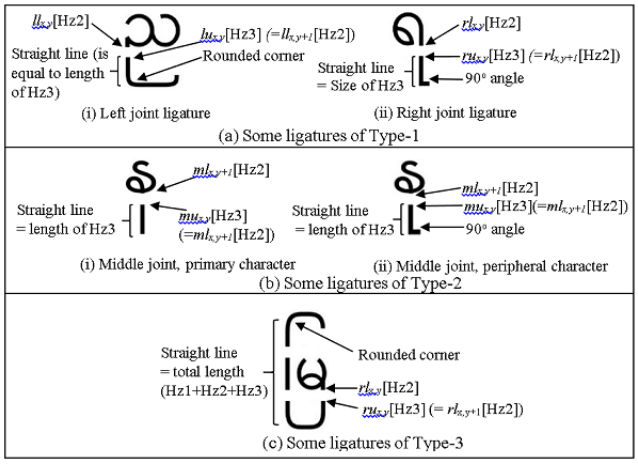

Figure 7. Horizontal segmentation on different ligatures

In H-zone segmentation, every ligature from input string sequence is initially considered to segment into equally separated zones as $1/3$. If there is no sign of extension beyond one zone for both primary and peripheral ligatures, they are separated according to their respective zones following the "if" condition of Alg. 1 (line 12-14, 18-20, 24-26).

When they have specific features, they are not separated but left as they are or merged together with primary characters as described in "else" condition of Alg. 1. For these kinds of unusual ligatures, we organized them depending on their types: Type-1, Type-2 and Type-3 respectively as shown in Fig. 6 and Fig. 7.

The first type of letters are those whose lower zone is connected with their central zone at the rightmost or leftmost position. Detailed illustration of these features is presented in Fig. 6(a) and Fig. 7(a) with their coordinate regions situated in corresponding three H-zones. Type-1 character's segmentation process is performed in line 18-23 of Alg. 1 for left joint and 24-29 for right joint.

The second type of letters are those whose lower part is connected at the middle position of that the letters existed in central zone. In the aspect of second characters shown in Fig. 6(b, c) and Fig. 7(b), there are some conflicts to determine whether it should be segmented or not due to very similar pattern between primary characters (ခ) of Fig. 6(b) and compound word of primary and peripheral characters (ခ, a combination of ခ +) of Fig. 6(c). If there is no angle and direction in lower zone, the whole character existed in central and lower zones is primary characters unlike peripheral characters that connect to central characters. The features are described in Fig. 7(b).

The middle point consideration is also considered in Type-3 but its additional considerations (such as total length of characters, rounded corners etc.) are a little different with Type-2. Due to similar consideration in middle point, we combined the checking both types condition in line 12 of Alg. 1. Those lines perform checking of Type-2 and Type-3 with their middle point of upper or lower zones' characters against types in middle position of central zone.

For all the above types of words, we used sliding windows/frames technique to learn the different structural and statistical information for offline recognition. Sliding is done from left to right starting from the first pixel of the word similar to [13]. Generally, the features such as structure and static features are extracted only after segmentation in OCR's work [14-19]. However, due to necessity in our present study, required features are extracted for segmentation. We therefore highlight that segmentation and feature extraction processes recursively work in Myanmar script as mentioned in Fig. 5. The segmented character after H-zone segmentation is sampled in Fig. 9(b).

Vertical Zone Segmentation **(V)**: After H-zone segmentation, V-zone segmentation is performed. In V-zone segmentation, there are disjoint or joint ligatures (V1, V2, V3 and V4) remaining in V-zone layer after H-zone segmentation. Subject to nature of Myanmar script, V1 and V4 own original disjoint ligatures, and V3 also occupies segmented ligatures due to H-zone segmentation. The only area that needs to be segmented for V-zone is V2, which have remaining connected parts after H-zone segmentation.

Input character strings from H-zone is segmented into vertical zones by finding disjoint portions of the characters. In segmentation, Nomura et al [20] used morphological thickening and thinning operations to separate touched and overlapped characters while Garain et al [21] adopted multi-factorial analysis for connected character's segmentation. In formulating our method, we considered projected density of pixels and an adaptive refinement rate parameter to find the disjoint ligatures of Myanmar script and overlapped portion of the characters consistent with the theory of Haji et al [22] as in line 2 of Alg. 2. We extensively utilized morphological heuristics to identify segments with more than one possible connection to other ligatures following Alg. 2 for

the remaining connected characters in V2 zone, i.e. မျ, မျာ, မာ, မြစ္ပြ, ကျာ, ကာ, etc. The sample results of V-zone segmentation are presented in Fig. 9(c).

Algorithm 2. V-zone segmentation

Input: $im_{x,y}$ in V2 zone, w_thres = threshold value for width of a compound character
Output: $Seg\_im_{x,y}$: Segmented or non-segmented original character
Algorithm:

1. if zone is V2
2. $im_{x,y}$ = disjoint the connected parts and extract G2 zone from V2
3. f = get statistical information such as number of pixels, cross, loop, open direction, etc.
4. if length of $im_{x,y}$ in G2 $> w\_thres$ then
5. if $im_{x,y}$ in G2 contains features f then
6. $Seg\_im_{x,y}$ =perform V-segment
7. else
8. $Seg\_im_{x,y}$ =no segmentation
9. end if
10. else
11. $Seg\_im_{x,y}$ =no segmentation
12. end if
13. end if

Alg. 2 captures G2 portion of an $im_{x,y}$ existed in V2 and w_thres threshold value for vertical segmentation. It first performs character thinning using Zhang Suen thinning algorithm [23] and study [24] and get disjoint parts such that resulted disjoint parts for မျာ = {ဓ ,ျ, ာ}. It then analyzes statistical information such as number of pixels, open-direction, etc. Depending on size of characters shown in Fig. 2(b, c) and statistical information, it decides whether the input character should vertically be segmented or not.

4.3 Feature Extraction

Feature extraction is a key step in the OCR system. It helps differentiate characters depending on their salient patterns. An efficient extraction process reduces the volume of data to be classified and that enhances the classification performance of the OCR system [25].

Comparatively, there are a common set of features usually extracted by OCR system of earlier studies [14, 15, 16, 19]. Whilst some research works [14, 15, 16] extract structural features based on geometrical or topological features like place of dots, number of strokes, number of cross, loops, end-points, aspect ratio of an image or slope, others [17, 18] relied on statistical distribution of pixels, normalized length features, number of holes and curvature features together with the consideration of computational facts such as rotation, translation and scaling invariant features among others. The study in [19] adopted a global transformation method to extract the features of the input images based on scale, location invariant features transform on different locations, various sizes, moments and orientations.

In the light of these features, we consider structural and statistical information for feature extraction to discover loops (open or close), slopes detection (ေ), angle detection (ျ), number of strokes, cross and ends, etc. that need to be analyzed for Myanmar script. To generate all required characteristics that describe Myanmar script, we consider PHOG features [26] to extract shape and spatial layout of image. PHOG is a spatial shape descriptor as an extension of the histogram of gradient (HOG) with different pyramid levels. Our window or frame sliding technique for feature extraction is consistent with the forms proposed in [26, 27]. The cells in

each window are divided into different pyramid levels N, and then calculates histogram of gradient orientation from each cell are calculated before they are combined into L bin that indicates an octant of angular radian space for the input image. Practically, there are two optional orientation ranges, i.e. [0-180] and [0-360], but different studies used their own ranges. We use a [0-360] orientation range for $N = 3$ and pyramid level $L = 8$.

4.4 Recognition

OCR recognition system relates to the problem of heuristic logic of human being as they recognize the characters and their shapes with their natural intelligence and experiences [28]. In our recognition phase, we use HMM to recognize primary characters while SOM is exploited for peripheral characters. They are explained in details as follows.

A. HMM Classification for Primary Ligatures

HMM is a statistical Markov model being structured to search hidden (unobserved) states based on their transition probabilities. HMM modeling of character recognition generates a sequence of thin vertical images called segments. HMM for each character is separately built such as 'ന', 'ə', etc. The underlying model of HMM can be described with transition probability X, observation probability Y and initial state probability π as HMM $= (X, Y, \pi)$.

HMM is modeled as categorical distribution for the parameters of X and Y by analyzing sequence observations. Depending on these observations, HMM tries to find suitable patterns related with them. HMM calculates conditional probability distribution of the hidden random variable $X(t)$ at a hidden state at time t, $X(t) \in \{x_1, x_2, \ldots$ based on random variables $Y(t) \in \{y_1, y_2, \ldots$ They are captured at observation time T, $T \in \{t, t+1, \ldots$ The N, M and k are constant limited numbers of X, Y and T variables.

The recognition capability of HMM depends on how well the model is trained with a well-organized training dataset and the number of states adopted during training [5]. When training our model, we used HMM for primary characters which are situated in the zones of G2, G3 and GL1 of Fig. 8 with their corresponding feature vector sequences. To recognize characters, Viterbi algorithm is exploited to predict the most likely characters.

B. SOM Classifier for Peripheral Characters

Character recognition is an intensive computational problem in the nature of heuristically complex logics and adaptability of characters such as shapes, rotations, etc. To solve this problem, we use competitive learning in network architectures called Kohonen's self-organization map in the form of feedback networks. Self-organization feature map classifies the input feature vectors into corresponding character groups based on the distribution layers and topology of the input vectors they are trained by.

SOM is a map space composed of neurons or nodes are arranged in hexagonal or rectangular grid. Each node relates with a weight vector, which will be updated to proceed to input data by training phase without updating the topology induced from the map space [29]. Once trained, SOM can classify a vector from data space by searching corresponding node, which has less dissimilar distance metric of weight vector to that data space.

SOM maps the vectors with N input variables par rapport, the weights of neurons existing in the Kohonen layers composed of only input and output layers without consideration of hidden layers. SOM assists recognition process for the groups of similar input vectors in such a way that neurons in layers

are physically similar to each other against with those vectors. The similar neurons are determined using a common distance function called Euclidean distance in following equation. The weights of most similar neurons and its neighbors are updated depending on learning rate and neighborhood functions in each iteration

$$Distance_{euclidean}(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}.$$

In training phase, the input feature vectors of peripheral characters become as input neurons. There are 5 output neurons for upper zone, another 5 neurons for lower zone, etc. For input layers, we use 49 neurons for 7×7 matrix for input character image while the total number of output neurons is 14. The input vectors are normalized by multiplying the input with normalization factor, which is *1/vector* length. The most similar neurons called winning neurons are calculated from normalized input vector by multiplying it with each row of output matrix. The maximum value from output neuron is regarded as winning neurons. These steps are repeated after adapting the weights of output neurons until it finds winner neuron for each character. For testing or recognition step, similarly as training phase but at this step, winning neuron is declared as recognized character.

C. Post Recognition Process to Combine Two Recognition Results

The purpose of post recognition process is to form a character from segmented and recognized character strings from HMM and SOM. During segmentation, phrase is segmented without knowledge of any character in the collection of words. Therefore, post recognition process is not only to organize the partial parts recognized by previous two recognition processes into compound character, but also to produce correct character by matching resulted compound characters with similar characters from the lexicon. In this case, coordinates transformed into groups of character position plays a key role recognizing individually separated characters. We refer to this as spatial data, it is owned by every ligature.

To reorganize partially recognized characters, resulted characters are put into to their corresponding groups as shown in Fig. 8. The regions in black are not considered in grouping because the characters will not fall into those regions due to structure of Myanmar script. In accordance with systematic structure of Myanmar script, we can predict which part of characters will fall into which groups and also what kind of characters will be existent in those parts. The gray region holds primary characters while the other regions receive peripheral ligatures.

Figure 8. Horizontal (H) and vertical (V) zones of Myanmar script

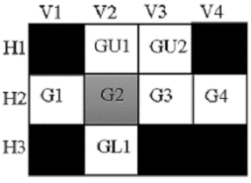

Figure 9. Sample characters partition according to

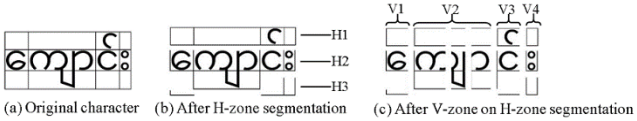

Fig. 8

The regions G2, G3 and GL1 can contain almost all 33 consonants except few unconventional characters. Primary characters will certainly fall in G2 region. The one and only character will be occupied by GU2 is "ၚ" for prior knowledge of Myanmar script nature. All possible characters that will be owned by each group are listed in Table 2. To sum up, each group holds specific characters, which is helpful to reorganize the recognized characters in correct position of compound characters.

Table 2. Groups of classified characters depending on zones

Upper peripheral zone	GU1	◌ိ, ◌ီ, ◌ဲ, ◌ံ
Upper peripheral zone	GU2	◌ိ်
Lower peripheral zone	GL1	◌ု, ◌ူ, ◌ျ, ◌ြ, ◌ွ
Left vertical peripheral zone	G1	◌ေ
Right vertical peripheral zone	G4	◌း
More than one zone	GU1, G2, GL1	◌ျ, ◌ြ

In differentiating the characters into groups; the spatial data, lexicon and probability value called propensity scores are used to determine the differences between specific character groups. Propensity scores are the conditional probability of receiving the treatment, given by their observed characteristics such as group, spatial data, etc. It then matches the current primary characters to all possibly combinatorial groups exclusively. It matches this character to one or more other similar character groups on propensity scores using k-NN algorithm. k-NN performs recognition by comparing recognized compound characters with the most similar complete ligatures from the lexicon. Here, the distance between pairs of compared characters are measured with a cosine distance function.

The possible combination for a character is predetermined and stored in lexicon. There will be different combination numbers for each specific character as presented in Table 3, and we denote this as *n*, representing number of ligatures that should be owned by each character.

Table 3. Possible combination of a primary character

Primary characters	Pc 1	Pc 2	Pc 3	Pc 4	Pc 5	Pc 6	Pc 7	Pc 8	… …	Pc n
		--	--	--						
က (G2)	ကျ (G2+ G2GL1)	ကြီ (G2+ G2GL1+GU1)	ကြီး (G2+ G2GL1 +GU1 +G4)	ကေျ (G1+ G2+ G2GL1)	ကေျး (G1+ G2+ G2GL1 +G4)	ကေျာ (G1+ G2+ G2GL1)	ကေျာ့ (G1+ G2+ G2GL1 +GL1)	ကေျာ် (G1+ G2+ G2GL1 +GU2)	…	ကေျာင်း (G1+G2+ G2GL1) +GU2+G 3+G4)
ဝ (G2)	ဝံ (G2+ GU1)	ဝဲ (G2+ GU1)	ဝန် (G2+ G3+ GU2)	ဝန်း (G2+ G3+GU 2+G4)	ဝက် (G2+ G3+ GU2)					
တ (G2)	တဲ (G2+ GU1)	တံ (G2+ GU1)	တွဲ (G2+ GU1+ GL1)	တွင် (G2+ GL1+ G3+ GU2)	တွင်း (G2+ GL1+ G3+ GU2+ G4)	တီ (G2+ GU1)	တာ (G2)	တေ (G1+ G2)	..	တောင်း (G1+ G2+ G3+ GU2+ G4)

Pc = Possible combination

D. Recognition Algorithm

Our complete recognition algorithm is described in Alg. 3. It takes a set of input character strings and their feature vectors for processing. The output of recognition algorithm is arranged into *Reg _ X*[], where each recognized character is placed as individual element of an array. All array elements of *Reg _ X*[] are recognized output of input phrase.

Algorithm 3. Recognition algorithm

Input: *MX*[], a set of primary ligatures *X*, contained in a phrase

$SX_{gk}[]$, a set of peripheral ligatures X of groups K, $K \in \{\text{GU2, GU3, G1, G3, G4, GL2}\}$

$MX\_F[]$, a set of statistical and structure features of primary ligatures X

$SX_{gk}\_F[]$, a set of statistical and structural features of peripheral ligatures X of peripheral groups K

$Reg\_X_{SX}[\phi]$, a set of recognized peripheral ligatures of each peripheral group $SX_{gk}[]$

$X_{lexi}[]$, a set of characters trained in lexicon with their corresponding features SX_{lexi}_F

Output: $Reg\_X[]$, a set of recognized letter in the form of phrase

Algorithm:

1. for $i = 1$ to size $(MX[])$
2. Reg_X_{MX} = recognize $(MX[i])$ by HMM model
3. for $j = 1$ to k // number of peripheral groups
4. $Reg\_X_{SX}[j]$ = recognize $(SX_{gj}[1])$ by SOM // gets first peripheral characters of each different groups
5. end for
6. // combines primary characters and all its possible secondary characters
7. SD_MX = get spatial data of $MX[i]$
8. $Reg\_char = MX[i]$
9. for $j = 1$ to k
10. $SD\_SX[j]$ = get spatial data of $SX_{gj}[1]$
11. if SD_MX and $SD\_SX[j]$ are within same $X\_$ Coordinate then
12. //checks for G1
13. if ($X\_$ Coordinate $(SD\_SX[j]) < X\_$ Coordinate (SD_MX)) then
14. Reg_char = concatenate $SX_{gi}[1]$ to the leftmost of Reg_char
15. update array (j)
16. // checks for GU2
17. else
18. Reg_char = concatenate $SX_{gi}[1]$ to the rightmost of Reg_char
19. update array (j)
20. end if
21. else if SD_MX and $SD\_SX[j]$ are within the same $Y\_$ Coordinate then
22. // checks for GU2
23. if ($Y\_$ Coordinate $(SD\_SX[j]) < Y\_$ Coordinate (SD_MX)) then
24. Reg_char = concatenate $SX_{gi}[1]$ to the topmost of Reg_char
25. update array (j)
26. // checks for GL2
27. else
28. Reg_char = concatenate $SX_{gi}[1]$ to the bottom of Reg_char
29. update array (j)
30. end if
31. end if
32. end for
33. // k clusters run from lexicon with Reg_char
34. for $g = 1$ to k // k clusters
35. Reg_char = find most similar character from a set most similar character form lexicon $X_{lexi}[]$ compares against with Reg_char upon their features SX_{gk}_F and SX_{lexi}_F
36. end for
37. $Reg\_X[i] = Reg\_char$
38. end for
39. produce a set of recognized characters $Reg\_X[]$

The input character strings are separately put into two arrays, $MX[]$ for primary ligatures and $SX_{gk}[]$ for the other ligatures. Their sample representation is described in Table 4 and Table 5 respectively. There is only one array for primary ligatures while there are multiple arrays for peripheral ligatures due to different groups of peripheral ligatures. We organize similar ligatures into the same groups as shown in Table 5. HMM model is executed to recognize list of primary ligatures as line number 2 in Alg. 3 whereas all lists of different peripheral groups are represented as K and repeatedly recognized by SOM in 3-5 of Alg. 3.

Table 4. Groups and spatial data for a character string

Phrase with four characters	ကျောင်းဝန်းထဲတွင်															
Segmented strings	ေ	ာ	ျ	ာ	င	်	း	ဝ	န	်	း	ထ	ဲ	တ	ွ	င	်
Groups	G1	G2	G2	G2	G3	GU2	G4	G2	G3	GU2	G4	G2	GU1	G2	GL1	G3	GU2
Types	Pp	Pr	Pp	Pp	Pp	Pp	Pp	Pr	Pp	Pp	Pp	Pr	Pp	Pr	Pp	Pp	Pp
Spatial Data	SD 1	SD 2	SD 3	SD 4	SD 5	SD 6	SD 7	SD 8	SD 9	SD 10	SD 11	SD 12	SD 13	SD 14	SD 15	SD 16	SD 17

Pr = primary character, Pp = peripheral character

The partially recognized characters are combined in accordance with characters' spatial coordinate's values. The specific condition is performed for each group such as G1, GU2, etc. The first character from each peripheral group that are already recognized from SOM are extracted (line 10) and checked where to place those ligatures to primary ones (line 11-31). After putting them together with primary, the character element of each peripheral groups are removed and the array size and index in Alg. 3.1 is updated.

Index For each group SX_{ij}, i = groups, j = number of ligature	1	2	3	4	5
List of primary characters of G2	ာ	ဝ	ထ	တ	ာ
List of characters of G2 and GL1 named as G2GL1	ျ				
List of peripheral characters of G1	ေ				
List of peripheral characters of G3	င	န	တ		
List of peripheral characters of GU2	်	်	်		
List of peripheral characters of G4	း				
List of peripheral characters of GL1	ွ				

Table 5. Characters contained in each group

Algorithm 3.1 Update array (array $SX_{gk}[]$, index i)

1. remove $SX_{gj}[1]$ from $SX_{gj}[]$
2. update size and index so as to reduce size and start the array with index 1
3. return updated array $SX_{gk}[]$

For recognition of compound characters, k-NN process is performed from line 34-36. In line 37, each recognized character is put into array, indexed according to the element of primary ligatures array. The recognized character string for input image string is produced in line 39 as collection of $Reg\_X[]$.

V. RESULTS AND DISCUSSION

In this section, experiments are conducted with a set of training and testing scanned images for offline scanned text recognition. Each output is a

subject to all stages of proposed system. Prior to experimental works, the preparation of lexicon and training datasets are firstly discussed as follows.

5.1 Lexicon

Although there are availably commercial and academic lexicons for popular languages such as English, Sinhala [8], and Chinese [30], there is yet no standard lexicon for Myanmar script for OCR purposes. Therefore, in this work, we prepare a lexicon containing 45 compound words for each primary consonant. It has therefore almost $45 \times 28 = 1260$ ligatures (excluding special characters ဥ, ဍ, ဋ, ဝ and ဃ from Fig. 1(a)). Our system did not only store possible combinations of a character (as shown in Table 3) but also grouped similar ligatures with k-NN as described in the training stage of this system. The stored and grouped characters are shown in Table 6.

Tested groups	Ligature groups	k-similar members of a cluster	Total number of testing characters in different fonts and sizes	No. of characters Accurately segmented	No. of characters Accurately recognized	Segmentation Accuracy (%)	Recognition Accuracy (%)
T1	G2	ဓ, ဟ	200	197	196	98.5	98
T2	G2, GU1	ဒိ, ဒံ, ဒီ, ဓဲ	187	183	181	97.86	96.79
T3	G2, GL1	ဒျ, ဒူ, ဒူ, ဒူ, ဒှ	210	196	193	93.33	91.90
T4	G2, GU1, G2GL1, G3	ဒွဲ, ဒွီ, ဒို, ဒြ, ဒြာ, ဒြာ, ဒြာ:, ဒြဲ, ဒြိ, ဒြီ, ဒျိုု	320	271	262	84.67	81.86
T5	G2,G3, GU2	ဓန်, ဟန်, ဓက်	159	146	142	91.82	89.30
T6	G2, G2GL1, GU2, G3	ဒြန်, ဒြက်, ဒြင့်, ဗျက်	170	152	147	89.41	87.06
T7	G1, G2, G3, GU2, G2GL1	ဓျောက်, ဓြောက်	137	121	117	88.32	85.40
T8	G1, GU2, G2GL1, G2	ဓျော်	128	111	108	86.72	84.38
T9	G1, G2, G2GL1	ဓျော, ဓျော့	135	116	113	85.93	83.70
Total			1646	1493	1459	90.73	88.71

Table 6. Accuracy measurement of our system for each similar group of a sample character 'ဓ' suggested by k-NN

In this case, the minimum and maximum numbers of k-cluster are assigned to 4-9, which are very realistic with similar character patterns of Myanmar script. The purpose of using lexicon in this study is to check the spellings of the characters that are outputted at the final recognition process. The k-neighbors clustering algorithm is used to determine groups of similar characters in each phrase of a text line, which are afterwards put into corresponding groups of lexicon that is trained well with similar clusters for correct recognizing of new input characters.

5.2 Dataset Preparation for Training and Testing Process

Due to lack of standard dataset for Myanmar OCR system, we collect the scanned images for each consonant and its compound words in three different font sizes; 10, 12 and 14 for two different font styles, namely Myanmar3 Unicode font and Zawgyi-One font which is the most widely used

in Myanmar ($1260 \times 3 \times 2 = 7560$ sample images in total for character-level dataset).

5.3 Experimental Results and Analysis

This section presents all experimental works and results by comparing our work with contemporary Myanmar OCR systems, namely OCRMPD [31] named by themselves and OCRSVM [32] shorten by us for convenient reference. The former compared of work called OCRMPD stands for OCR for Myanmar printed documents to get machine understandable texts from printed images. They mainly used hierarchical SVM classifier for recognition of images in character level without giving a clear explanation of any single or compound character recognition. The later one proposed online handwritten Myanmar compound words based on solely focusing on statistical and semantic approach in recognition without segmentation process in compound words.

As traditional classification or recognition problem, recognized rate is accurately evaluated and symbolized as $accuracy = Ac / Tc$, where Tc is the number of testing examples and Ac is the number of correctly classified examples. It measures and analyzes under different evaluation metrics.

The accuracy values of each ligature group executed by our proposed system are listed in Table 6. From the results, the primary ligatures group G2 of tested group T1 results had the highest accuracy rate, while group T4 produced the least accurate results amongst all groups. To conclude the results, the overall accuracy 88.71% is achieved for all ligatures groups.

In order to shed light on zone segmentation, we compare our zone-segmentation with other non-zone segmentation works: OCRMPD and OCRSVM. The results are presented in Table 7 and it shows that our system significantly surpasses other two approaches in both zone and non-zone segmentation results. Due to effectiveness of zone segmentation, the other two approaches improve their recognition results but their results are relatively lower than our system.

Table 7. Recognition results upon different segmentation schemes

OCR Approaches	Number of test samples	No. of recognition without zone segmentation	Number of recognition with zone segmentation	Accuracy without segmentation (%)	Accuracy with segmentation (%)
OCRMPD	200	107	148	53.5	74.00
OCRSVM	200	139	114	69.5	57.00
Our system	200	141	187	**70.5**	**93.5**

Apart from segmentation, we performed accuracy analysis of all three approaches for the different ligatures forms, such as disjointed single word, compound words with joint or disjoint ligatures, and words that combine these two into two layers as shown in Table 8. Whilst the results for two-layer ligatures recognition are the lowest for all approaches, the results of single words recognition hits the highest rates. Overall, our system significantly outperforms in all types of ligatures.

Table 8. Recognition results from the aspect of different ligatures types

OCR Approaches	T1 Acc: (%)	T2 Acc: (%)	T3 Acc: (%)	T4 Acc: (%)	T5 Acc: (%)	T6 Acc: (%)	T7 Acc: (%)	T8 Acc: (%)	T9 Acc: (%)	Total Acc: (%)
OCRMPD	91.50	89.10	70.50	53.58	63.47	60.54	58.71	57.30	55.96	66.74
OCRSVM	94.33	90.52	66.15	46.26	54.14	53.22	51.76	49.32	48.65	61.59
Our system	98	96.79	91.90	81.86	89.30	87.06	85.40	84.38	83.70	88.71

To elucidate the efficiency of using different classifiers in each different ligature forms, the results are scrutinized in their classification level as shown in Table 9. According to the results shown in the tables, lexicon-driven recognition gives better accuracy rates than traditional recognition results.

Approaches	Total tested characters for all types of ligatures	Accuracy of HMM classifier (%)	Accuracy of SOM classifier (%)	Combined classification without k-NN	Combined classification without k-NN
Our system	2380	94.32	86.43	72.43	89.28

Table 9. Accuracy measurement for different classifiers

VI. CONCLUSION AND FUTURE WORK

This paper has presented a new segmentation and recognition algorithm for Myanmar script that is cursive and complex structure. Specific segmentation processes are performed depending on nature of ligatures and different recognition classifiers are executed in accordance with ligatures groups. This paper mainly used HMM model for primary ligatures and SOM model for peripheral ligatures. It combined with the recognition results obtained from this two models and performed final recognition process with k-NN algorithm to find the most accurately recognized results with a lexicon. We used a lexicon that preserved the Myanmar characters features. To conclude, our technique for Myanmar script recognition has significantly achieved higher recognition rates than other studies of Myanmar OCR. As shown in Fig. 1(c), special letters are not considered in the recognition algorithm, this will be a future research because they are rarely used in general writing and reading except Buddhism related to language called Pali in Buddhism Bible. In addition, we plan to extend this research for handwritten OCR system for both online and offline recognition with a more advanced recognition algorithm and techniques.

Conflict of interest statement: None declared.

VII. REFERENCES

[1]. P Sahare and S. B. Dhok, "Multilingual character segmentation and recognition schemes for Indian document images," Digital Object Identifier, Vol. 6, IEEE Access, 2018, pp. 10603-10617.

[2]. P P. Roy, A. K. Bhunia, A. Das, P. Dey and U. Pal, "HMM-based Indic handwritten word recognition using zone segmentation," Pattern Recognition, Vol. 60, 2016, pp. 1057-1075.

[3]. D Tao, L. Liang, L. Jin and Y. Gao, "Similar handwritten Chinese character recognition by kernel discriminative locality alignment," Pattern Recognition Letters, Vol. 35, 2014, pp.186-194.

[4]. X Xiao, L. Jin, Y. Yang, W. Yang, J. Sun and T. Chang, "Building fast and compact convolutional neural networks for offline handwritten Chinese character recognition," Pattern Recognition, Vol. 72, 2017, pp. 72-81.

[5]. R D. Zarro and M. A. Anwer, "Recognition-based online Kurdish character recognition using hidden Markov model and harmony search," Engineering Science and Technology, an International Journal, Vol. 20, 2017, pp. 783-794.

[6]. A Bharath and S. Madhvanath, "HMM-based lexicon-driven and lexicon-free word recognition for online handwritten Indic script," IEEE Transactions on Pattern Analysis

and Machine Intelligence, Vol. 34, No. 4, 2012, pp. 670-682.

[7]. J J. Weinman, E. Learned-Miller and A. R. Hanson, "Scene text recognition using similarity and a lexicon with sparse belief propagation," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 31, No. 10, 2009, pp. 1733-1746.

[8]. H L. Premaratne, E. Jarpe and J. Bigun, "Lexicon and hidden Markov model-based optimization of the recognized Sinhala script," Pattern Recognition Letters, Vol. 27, 2006, pp. 696-705.

[9]. "Burmese language" or "Myanmar language," https://en.wikipedia.org/wiki/Burmee_language , January 2018.

[10]. H. Malik and M. A. Fahiem, "Segmentation of printed Urdu scripts using structural features," Second International Conference in Visualization, IEEE, 2009, pp. 191-195.

[11]. R. Pramanik and S. Bag, "Shape decomposition-based handwritten compound character recognition for Bangla OCR," Journal of Visual Communication and Image Representation, Vol. 50, 2018, pp. 123-134.

[12]. M. Akram and S. Hussain, "Word segmentation for Urdu OCR system," Proceedings of the 8th Workshop on Asian Language Resources, Asian Federation for Natural Language Processing, Beijing, China, 2010, pp. 87-93.

[13]. J. H. AIKhateeb, J. Ren, J. Jiang and H. AI-Muhtaseb, "Of?ine handwritten Arabic cursive text recognition using hidden Markov models and re-ranking," Pattern Recognition Letters, Vol. 32, 2011, pp. 1081-1088.

[14]. D. B. Megherbi, S. M. Lodhi and A. J. Boulenouar, "Fuzzy logic model-based technique with application to Urdu characters recognition," Proceedings of SPIE, Vol. 3962, 2000, pp. 13-24.

[15]. S. A. Sattar, S. Haque, M. K. Pathan and Q. Gee, "Implementation challenges for Nastaliq character recognition," Wireless Networks, Information Processing and Systems, Communications in Computer and Information Science (CCIS), Vol. 20, Springer-Verlag Berlin Heidelberg, 2008, pp. 279-285.

[16]. S. A. Sattar, S-ul Haque and M. K. Pathan, "A finite state model for Urdu Nastalique optical character recognition," International Journal of Computer Science and Network Security, Vol. 9, No. 9, 2009, pp.116-122.

[17]. U. Pal and A. Sarkar, "Recognition of printed Urdu script," 7th International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2003, pp. 1183-1187.

[18]. S. M. Lodhi and M. A. Matin, "Urdu character recognition using fourier descriptors for optical networks," Proc. SPIE, Photonic Devices and Algorithms for Computing VII, Vol. 5907, 2005, pp. 59070O-1-59070O13.

[19]. S. Zaman, W. Slany and F. Sahito, "Recognition of segmented Arabic/Urdu characters using pixel values as their features," 1st International Conference on Computer and Information Technology (ICCIT), 2012, pp. 507-512.

[20]. S. Nomura, K. Yamanaka, O. Katai, H. Kawakami and T. Shiose, "A novel adaptive morphological approach for degraded character image segmentation," Pattern Recognition, Vol. 38, 2005, pp. 1961-1975.

[21]. U. Garain and B. B. Chaudhuri, "Segmentation of touching characters in printed Devanagari and Bangla scrips using fuzzy multifactorial analysis," IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, Vol. 32, No. 4, 2002, pp. 449-459.

[22]. S. A. B. Haji, A. James and Dr. S. Chandran, "A novel segmentation and skew correction approach for handwritten Malayalam documents," Procedia Technology, Vol. 24, 2016, pp. 1341-1348.

[23]. Z. Sune, "Zhang-Suen thinning algorithm," https://rosettacode.org/wiki/Zhang-Suen_thinning_algorithm, January 2018.

[24]. "The Thinning Algorithm," University of Oxford, https://users.fmrib.ox.ac.uk/~steve/susan/thinning/node2.html, January 2018.

[25]. S. Naz, K. Hayat, M. I. Razzak, M. W. Anwar, S. A. Madani and S. U. Khan, "The optical character recognition of Urdu-like cursive scripts," Pattern Recognition, Vol. 47, 2014, pp. 1229-1248.

[26]. Y. Bai, L. Guo, L. Jin and Q. Huang, "A novel feature extraction method using Pyramid histogram of orientation gradients for smile recognition," 16th International Conference on Image Processing (ICIP), IEEE, 2009, pp. 3305-3308.

[27]. H. A. AI-Muhtaseb, S. A. Mahmoud and R. S. Qahwaji, "Recognition of off-line printed Arabic text using hidden Markov models," Signal Processing, Vol. 88, 2008, pp. 2902-2912.

[28]. Prof. S. A. Nirve and Dr. U. B. Shinde, "Hindi character recognition using Kohonen network," International Journal of Scientific & Engineering Research (IJSER), Vol. 4, Issue 5, 2013, pp. 1786-1790.

[29]. A. Goyal, A. Lakhanpal and S. Goyal, "Learning of alphabets using Kohonen's self-organized featured map," International Journal of Application or Innovation in Engineering & Management (IJAIEM), Vol. 2, Issue 12, 2013, pp. 283-287.

[30]. C. L. Liu, F. Yin, D. H. Wang and Q. F. Wang, "CASIA online and offline Chinese handwriting databases," International Conference on Document Analysis and Recognition, IEEE, 2011, pp. 37-41.

[31]. H. P. P. Win, P. T. T. Khine and K. N. Tun, "OCRMPD: OCR System for Myanmar printed document image with a novel segmentation method and hierarchical classification Scheme," 7th International Conference on Intelligent Computer Communication and Processing, IEEE, 2011, pp. 285-291.

[32]. H. P. P. Win, P. T. T. Khine and K. N. N. Tun, "Converting Myanmar printed document image into machine understandable text format," 6th International Conference on Digital Information Management, IEEE, 2011, pp. 96-101.

Authors:

Chit San Lwin holds a BSc degree from Monywa University, Myanmar, from 2006. Received MSc from Moscow State University (MSU), Russia, in 2011. Graduated with a master of research (MRes) from Monywa University, Myanmar, in 2015. Currently pursuing a PhD degree at Harbin Institute of Technology (HIT), Harbin, China.

Xiangqian Wu has been a professor at School of Computer Science and Technology, Harbin Institute of Technology, China, since 2009. He has authored one book and over 100 papers in international journals and conferences. Current research interests include computer vision, pattern recognition, and biometrics and medical image analysis. He is also my supervisor.