Handwritten Character Recognition using Morphological Operators with Competitive Neural Trees

Theingi Htike, Yadana Thein
University of Computer Studies
htike83@gmail.com

Abstract

In this paper an attempt is made to develop Myanmar handwritten character recognition system. Character recognition is an important area in image processing and pattern recognition fields. The aim of character recognition is to translate human readable characters to machine readable characters. The paper describes the process of character recognition using morphological operators with the competitive neural trees. The morphological operators are used to extract the edge of each character image. Then, competitive neural trees (CNeT) are used for classification. It is one of the fast supervised neural networks with high performance. The main advantage of the CNeT is its structured, self-organizing architecture that allows for short learning and recall times. High speed recognition rates can be gained by using CNeT.

Keywords – Myanmar handwritten characters, morphological edge extraction, CNeT

1. Introduction

Handwritten character recognition is a very interesting field for researchers. Many feasible techniques have been developed in this field. Character recognition is the ability of a computer to receive and interpret handwritten input from sources such as paper documents, photographs, touch-panels, light pen and other devices. The domain of hand written text recognition has two completely different problems of On-line and Off-line character recognition.

On-line character recognition [1] involves the automatic conversion of characters as it is written on a special digitizer, where a sensor picks up the pen-tip movements as well as pen-up/pen-down switching. The off-line character recognition is comparatively difficult, as different people have different handwriting styles and also the characters are extracted from documents of different intensity and background [2].

Feature extraction is an integral part of any optical character recognition system. Extremely features vectors can be extracting to get a high accuracy in classification these features vectors should be minimized by purification the set of non suitable features vectors, and then the rate of success identification can be improved.

The Myanmar language is the official language and is more than one thousand years old. Myanmar script is considered a complex script by software developers, as it originated from Indic scripts like Thai or Khmer. The Myanmar (formerly known as Burmese) script developed from the Mon script, which was adapted from a southern Indian script during the 8th century.

In this paper we proposed morphological operators with competitive neural trees. In Myanmar character recognition fields, competitive neural trees had not been applied for recognition. Therefore, we applied competitive neural trees for Myanmar handwritten recognition. It is one of the fast supervised neural networks. The generalization ability of the CNeT is guaranteed by forward pruning, which is an inherent part of the learning process.

The aim of our proposed system is to implement an effective approach which is able to recognize for Myanmar handwritten characters. The style of writing characters is different and they come in various sizes and shapes. The proposed system is easily to recognize handwritten characters of several different writing styles. Competitive neural trees are used to improve accuracy rate. Feature vectors of each character are extracted using morphological operators to reduce processing time.

The remainder of the paper is organized as follows: **Section 2** describes Myanmar language pattern. **Section 3** explains proposed system design and the various steps involved in the OCR System. **Section 4** presents competitive neural trees and **Section 5** explains the conclusion.

2. Myanmar Language Pattern

The interests in Myanmar handwritten characters recognition research have grown over the past few years but practical research is only a few works in research field. Because, the problem of Myanmar characters recognition is more difficult than English languages in respects including the similarity of characters, absence space between each word, etc. They are still in research field, not complete work. Myanmar language is the official language and widely used in many Myanmar states. Myanmar language consists of (33) consonants, (12) vowels and (4) medials.

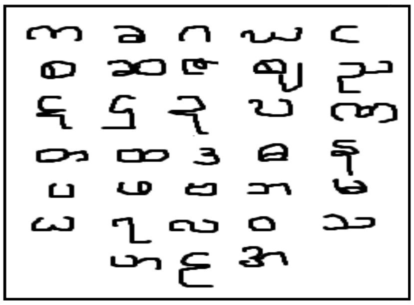


Figure 1. A Set of Myanmar Characters

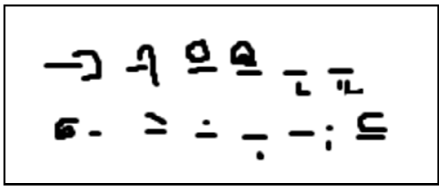


Figure 2. A Set of Myanmar Vowels

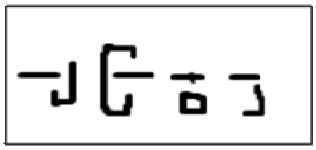
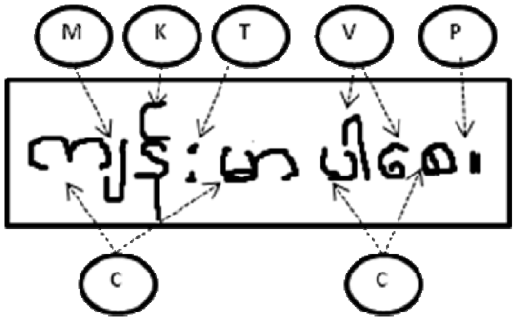


Figure 3. A Set of Myanmar Medials

In Myanmar (Burmese) writing system: syllabic alphabet - each letter has an inherent vowel. Other vowels are indicated using separate letters or diacritics which appear above, below, in front of, after or around the consonant. The rounded appearance of letters is a result of the use of palm leaves as the traditional writing material. Straight lines would have torn the leaves. The Burmese name for the script is 'round script', is written from left to right, as shown in Figure 4.

In Myanmar syllable structure, syllables or compound words are formed by consonants combining with vowels or medials. However, some syllables can be formed by just consonants, without any vowel. In our proposed system, we consider to implement not only 33 consonants but also the compound words.



C – Consonants P – Punctuation K – Killer
T – Tone V – Vowels M – Medial

Figure.4. Terms of Character's Sample

3. Proposed System Design

A typical character recognition system is characterized by a number of steps, which include
(1) Image Acquisition
(2) Preprocessing
(3) Feature Extraction, and
(4) Recognition

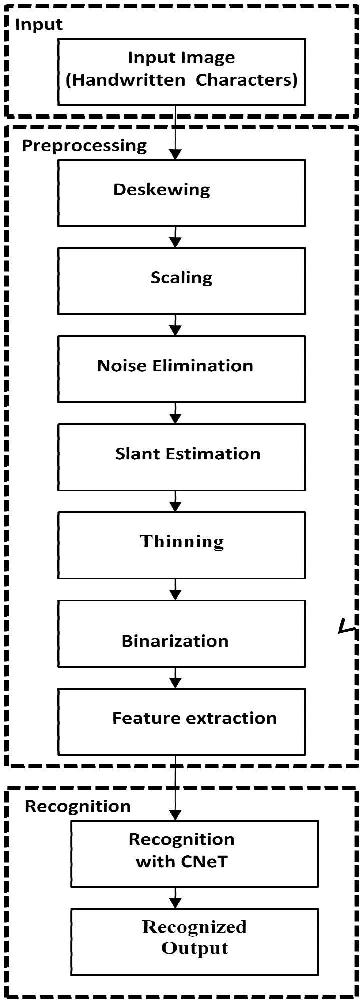


Figure.5. Proposed System Design

The steps required for character recognition are described here in detail:

3.1 Image acquisition

It performs getting image from other sources like digital cameras, video cameras, or scanned image, etc.

3.2 Preprocessing

Preprocessing aims at eliminating the variability that is inherent in cursive and hand-printed characters. The preprocessing techniques that have been employed in an attempt to increase the performance of the recognition process are as follows:

Deskewing: It is the process of first detecting whether the handwritten word has been written on a slope and then rotating the word if the slope's angle is too high so the baseline of the word is horizontal.

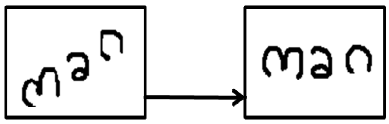


Figure.6. Deskewing

Scaling: It sometimes may be necessary to produce words of relative size.

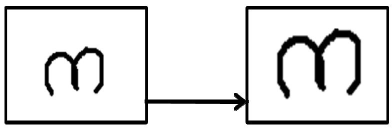


Figure.7. Scaling

Noise elimination: It in character images is important for further processing; therefore, these small foreground components are usually removed.

Slant estimation and correction: It is an integral part of any word image preprocessing. The slope can be estimated through analysis of the slanted vertical projections at various angles [4].

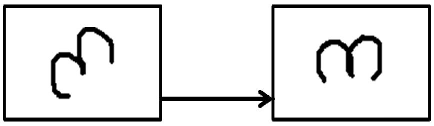


Figure.8. Slant estimation and correction

Thinning: It is a process in which the skeleton of the word image is used to normalize the stroke width.

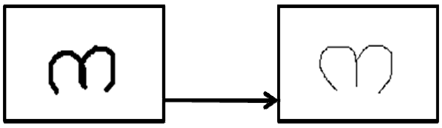


Figure.9. Thinning

Binarization: All hand printed characters are scanned into grey scale images. Each character image is traced vertically after converting the gray scale image into binary matrix [3].
Feature extraction: Feature extraction is a process of studying and deriving useful information from the filtered input patterns. The derived information can be general features, which are evaluated to ease further processing. Feature extraction method is explained in detail section 3.3.

3.3 Morphological Feature Extraction

In this paper, feature extraction is done by an exhaustive search process using the morphological operators. The shape of the characters is extracted using morphological operations such as erosion, dilation, top hat transform, bottom hat transform.

Morphological operations can be defined as a combination of two basic operations, dilation and erosion. In dilation the value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood and in erosion value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood.

Morphological operations also make use of a structuring element *M*; which can be either a set or a function that correspond to a neighborhood-function related to the image function $g(x)$ [5]. In general, a dilation (denoted by $\oplus$) is every operator that commutes with the supremum operation. On the other hand, erosion (denoted by $\ominus$) is every operator that commutes with the infimum operation. There is a homomorphism between the image function g and the set B of all pixels with image function value 1. The structuring element $M(x)$ is a function that assigns a subset of $N \times N$ to every pixel of the image function. Then dilation, an increasing transformation is defined as

(1)

and erosion, a decreasing transformation is defined as

(2)

Similarly, opening of set B by structuring element M is defined as

(3)

and closing of set B by structuring element M is defined as

(4)

The residual of the opening compared to the original signal, i.e., represents the top-hat transform. Thus, when the opened signal is subtracted from the original, the desired detail is obtained. Its dual, the bottom-hat transform, is defined as the residual of a closing compared to the original image f, i.e.

.

3.4 Character Recognition with CNeT

This is the stage where an automated system declares that the inputted character belongs to a particular category. The recognition techniques here we have used is competitive neural trees. CNeT's are self-organizing neural architectures that combine the advantages of competitive neural networks and decision trees.

4. Competitive Neural Trees (CNeT)

The CNeT contains m-ary nodes and grows during learning by using inheritance to initialize new nodes. At the node level, the CNeT employs unsupervised competitive learning. The CNeT performs hierarchical clustering of the feature vectors presented to it as examples, while its growth is controlled by forward pruning. Because of the tree structure, the prototype in the CNeT close to any example can be determined by searching only a fraction of the tree.

4.1. CNeT Architecture

The CNeT has a structured architecture. A hierarchy of identical nodes forms a m-ary tree. Each node contains m slots $S_1,S_2,\ldots,S_m$ and a counter age that is incremented each time an example is presented to that node. The behavior of the node changes as the counter age increases. Each slot S_i stores a prototype $V_i \in V \subset R^n$, a counter count, and a pointer to a node. The prototypes are updated to represent clusters of examples. The slot counter count is incremented each time the prototype of that slot is updated to match an example. Finally, the pointer contained in each slot may point to a child node assigned to that slot. A NULL pointer indicates that no node was created as a child so far. In this case, the slot is called terminal slot or leaf. Internal slots are slots with an assigned child-node. The slots shown in Fig.10 also have class counters.

An alternative approach to the development of neural trees was motivated by competitive learning. The connection weights of the winner are then updated using gradient descent learning. New nodes are added to the tree when the error rate exceeds a certain threshold and some nodes are deleted if they remain inactive for a long period.

Neural trees are grown and pruned. Some algorithms grow a perfect tree that classifies all examples correctly [6]. Then a set of pruned sub trees is checked for performance on an independent testing set of examples and the best performing sub tree is selected. This method is called backward pruning. While the tree is growing, its performance on an independent testing set is checked. Usually the growing of the tree is terminated when its performance on the testing set begins degrading.

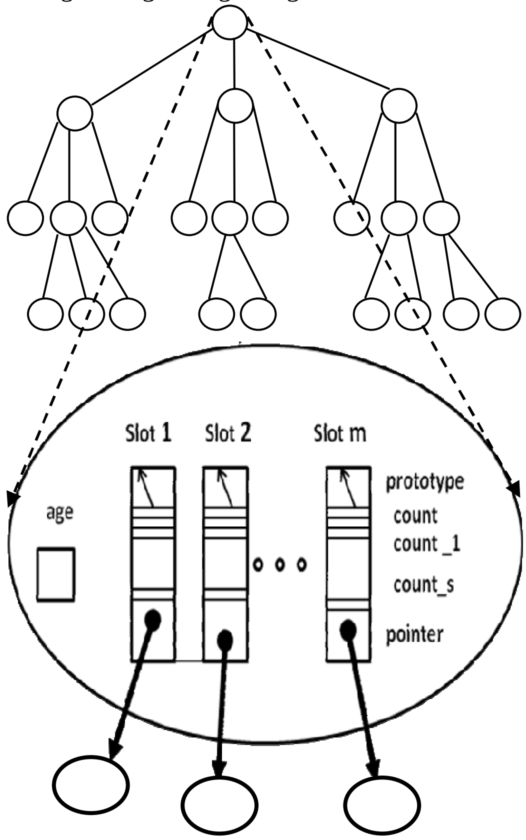


Figure.10. Tree structure of CNeT and node detail

4.2 CNeT Learning

Life Cycle: Each node goes through a life cycle. The node is created and ages with the exposure to examples. When a node is mature, new nodes can be assigned as children to it. A child-node is created by copying properties of the slot that is split to the slots of the new node. Right after the creation of a node, all its slots are identical. They will differentiate with the exposure to examples. Its prototypes are no longer updated in order to keep the partition of the input space for the child-nodes constant. The life cycle of a node may be partitioned into the following phases.

Step 1: Creation (at age 0): the node is initialized; the node inherits properties from the parent slot such as the prototype and a fraction of the class counters.

S**tep 2: Youth** (before the maturity age is reached): the prototypes compete to respond to the examples; the winning prototype is updated; the prototypes split the region of the input space that the node sees into sub regions.

Step 3: Maturity (after the maturity age has been reached): the prototypes still compete for the examples and they are updated; if a splitting criterion is TRUE, then a new child is created and is assigned to a slot.

Step 4: Frozen (as soon as a child is assigned): the prototypes compete for the inputs but they are not updated; if the winner has a child-node assigned, then it sends the example to the child.

Step 5: Destruction (after all children have been destroyed).

Training Procedure: If the CNeT is used for character recognition, its goal is to partition the input space into regions that are pure or almost pure. The general learning scheme works as follows. Do while stopping criterion is FALSE:

Step 1: Select randomly an example x. Let C_j be class that belongs to.

Step 2: Traverse the tree starting from the root to find a terminal prototype v_k that is close to x. Let n_ℓ and s_k be the node and the slot that v_k belongs to, respectively.

Step 3: If the node n_ℓ is not frozen, then update the prototype v_k

Step 4: If a splitting criterion for the slot s_k is TRUE, then assign a new node as child to s_k and freeze the node n_ℓ.

Step 5: Increment the counter $count_j$ for class C_j, the counter count in slot s_k, and the counter age in node n_ℓ.

5. Conclusion

This paper describes how the performance of character recognition can be improved. Morphological operators are used for feature extraction. Then CNeT is applied for classification. The CNeT performs hierarchical clustering by employing competitive unsupervised learning at the node level. The generalization ability of the CNeT is guaranteed by forward pruning, which is an inherent part of the learning process. The reliability and classification accuracy of the trained CNeT can be improved by a recall strategy that allows the rejection of some ambiguous examples. The CNeT can be trained in a tiny function to perform classification task.

REFERENCES

[1] A. Bharath and S. Madhvanath,” FreePad: a novel handwriting-based text input for pen and touch interfaces”, Proceedings of the 13th international Conference on Intelligent User Interfaces, pp. 297-300, 2008.

[2] Bhardwaj, F. Farooq, H. Cao and V. Govindaraju,” Topic based language models for OCR correction”, Proceedings of the Second Workshop on Analytics For Noisy Unstructured Text Data, pp. 107-112, 2008.

[3] A. Choudhary, R. Rishi, S. Ahlawat, V. S. Dhaka, ”Optimal feed forward MLParchitecture for off-line cursive numeral recognition,”International Journal on Computer Science and Engineering, vol. 2, no.1s, pp.1-7, 2010.

[4]D. Guillevic and C. Y. Suen,” Cursive script recognition: A sentence level recognition scheme”, Proceedings of the 4th International Workshop on the Frontiers of Handwriting Recognition, pp. 216–223, 1994.

[5].Soille: Morphological Image Analysis: Principles and Applications. Springer, Berlin (1999).

[6] A. Sankar and R. J. Mammone, “Optimal pruning of neural tree networks for improved generalization,” in *Proc. Int. Joint Conf. Neural Networks*, Seattle, WA, July 8–12, 1991, pp. 219–224.