ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

Handwritten Devanagari Character Recognition using SVM and ANN

Arjun Singh[1], Kansham Angphun Maring[2]

M.Tech, Computer Science and Engineering, Dr. B.R. Ambekar National Institute of Technology, Jalandhar, India [1, 2]

Abstract: Character recognition is a challenging research area due to its diverse applicable environment. It is able to solve complex problem and make easy task for human. This paper proposed a system for recognizing offline Handwritten Devanagari Characters using Artificial Neural Network and Support Vector Machines as classifiers and the results are compared. We discuss various characteristics of the classification methods that are applied successfully to handwritten Devanagari characters. It involves binarization, noise removable, and normalization. Statistical and structured based features are used for extracting the feature of characters. The feature extraction techniques: Chain Code, Zone Based Centroid, Background Directional Distribution and Distance Profile features are applied to the pre-processed images. Experiment is carried out by varying the image sizes: 30x30, 40x40, and 50x50 using MATLAB on more than 20,000 samples. The overall recognition accuracy is 97.61 % by using SVM.

Keywords: Optical Character Recognition (OCR), Artificial Neural Network (ANN), Pre-processing, Binarization, Feature Extraction, Support Vector Machine (SVM).

I. INTRODUCTION

An earliest notable attempt in the area of character recognition research is by Grim dale et al (1959).Devanagari is the script for writing Hindi, Marathi, Nepali and Sanskrit languageswhich all are Indian languages. The alphabets of Devanagari script consists of 33 consonants and 14 vowels. It is written from left to right. Devnagari has no concept of lower or upper case in Hindi language [1]. Most of the characters have a horizontal line at upper part in this language. The concept of handwriting has existed for old time, for the purpose of expanding people's memory and facilitating communication together, much of culture may be attributed to the advent of handwriting, many researchers began to focus their attention on attempting to simulate intelligent behaviour, one such example was the attempt to imitate the human ability to read and recognize printed and handwritten characters [2].

Optical character recognition [3] is used for recognizing the machine printed and handwritten character/numeral. Due to this has been an active area of research. It has ability to receive and interpret intelligible handwritten/printed input from various sources such as photographs, scanned paper or documents and other devices. The main work of OCR is to digitizing the scanned documents.In this paper; we proposed a system for offline recognition of Devanagari handwritten characters using ANN and SVM classifiers. Accuracy of printed Devanagari character recognition has been achieved about 100 % but there is still problem for handwritten Devanagari character recognition. We used Chain Code Histogram (CCH), Zone based Centroid Features, Distance Profile and Background Directional Distribution (BDD) features to extract the features of the characters.

The rest of the paper in organized with Literature Survey in section II followed by brief description of the Proposed System Architecture in Section III. Then the experimental results are presented in section IV followed by Conclusions in section V.

II. LITERATURE SURVEY

Character recognition of Devnagari started in late 1970s.Sethi et al (1977) have presented a Devanagari numeral recognitionin which the presence or absence of 4 basic primitives such as horizontal,vertical, right slant, and left slant are used for recognition with the help ofdecision tree. Later they attempted to recognize constraint hand-printedDevanagari characters using the same method. Pal et al (1997) also haveattempted to recognize the Printed Devanagari characters.Sinha et al (1985) carried out Devanagari script recognition usingsyntactic method with an embedded picture language. It is prototype context based recognition. Sinha (1987) later suggested Role of context in Devanagari text recognition system.Features used by N. Sharma et al (2006) for handwritten Devanagari characters are obtained from the directional chain code information of the contour points of the characters. Based on the CH, they have used 64-D features for recognition. They proposed a quadratic classifier-based scheme for the recognition of handwritten characters and obtained 80.36 % accuracy with the 11,270 dataset size.P. S Deshpande et al (2008) used Gaussian filter to give feature vector of dimensionality 200 (5x5x8). Accuracy of 94% is obtained using Support vector Machines (SVM) as the classifier. S. Arora et al (2010) used shadow features, and CH features for classification of handwritten Devanagari non compound characters. Two MLPs and a minimum edit distance (MED) method are used for classification. In the first stage of classification, characters with distinct shapes are classified using two MLPs. Shadow features are used for one MLP and CH features

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

are used for the other MLP for classification. In the second stage of classification, confused characters having similar shapes are classified using a MED method.

III.PROPOSED SYSTEM ARCHITECTURE

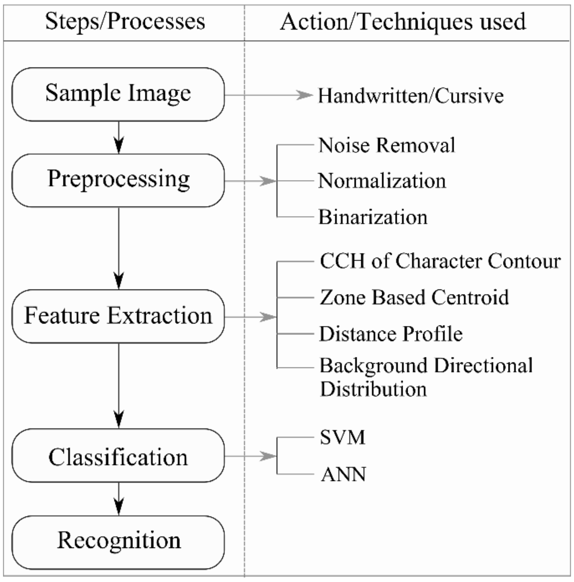

Fig.1. Proposed OCR System Architecture

A. Pre-processing
Pre-processing is a series of operations which are performed to enhance the image quality. It is essential to enhance the image feature for better results. The techniques used to enhance the image are described below:

A.1 Noise removal
During writing noise may be introduceddue to any writing mistakes or disturbance. When images is scanned, some scanning devices also introduced noises like filled loops, disconnected lines, bumps and gaps in line of characters [4]. Removing the noise is necessary to reorganization purpose.

A.2 Normalization
This is essential part of character recognition in pre-processing phase which attempt to remove variations in images so that image will not be able to change identity of character [5]. Normalization gives the proper shapes to the images so that feature of the character images can be compared. Basically it deals with sizes of the images. In this paper three different image sizes viz., 30x30, 40x40, and 50x50 are used and accuracies are compared.

A.3 Binarization
After Normalization of image thegray level image is converted into binary form so as toease analyze the behavior character [6]. Binarization converted gray scaled image into binary image where 0 shows black pixels and 1 shows white pixels. Global thresholding picks one threshold value for the entire document image based on an estimation of the background level from the intensity histogram of the image. In this paper global thresholding technique, Otsu's method is applied to binarize the images.

B. Feature Extraction
In this section, we discuss the feature extraction method used by the classifier. This techniques area applied after image pre-processing and the recognition accuracy of the system depends upon the feature extraction techniques. The four feature extraction techniques used in this paper are described below:

B.1 Chain Code Histogram of Character contour
This coding approach shows the direction of the next pixel in the image. Given a scaled image of handwritten character, we applied contour points method to the image and got contour of the scaled image. We have taken a 3×3 window which surrounded by the all object points of the image. If any of the 4-connected neighbour points is a background point then the object point (P), as shown in Fig. 2 is considered as contour point.

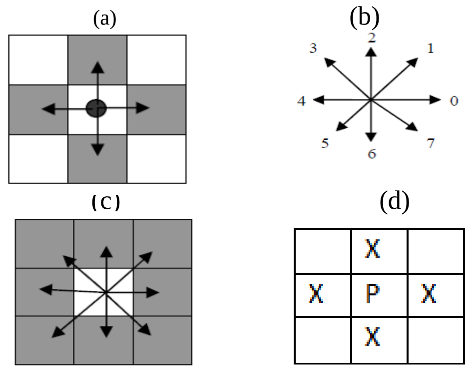

Fig.2. Chain Coding: (a) 4-connectivity, (b) 8-connectivity. (c) Direction of connectivity, generate the chain code by detecting the direction of the next-in-line pixel, (d) P is the processed element in Chain coding for feature.

The contour following procedure is used to trace the contour and a contour representation called "chain coding" as proposed by Freeman [7], shown in Fig. 3(b). Each pixel of the contour is assigned a different code that indicates the direction of the next pixel that belongs to the contour in some given direction.

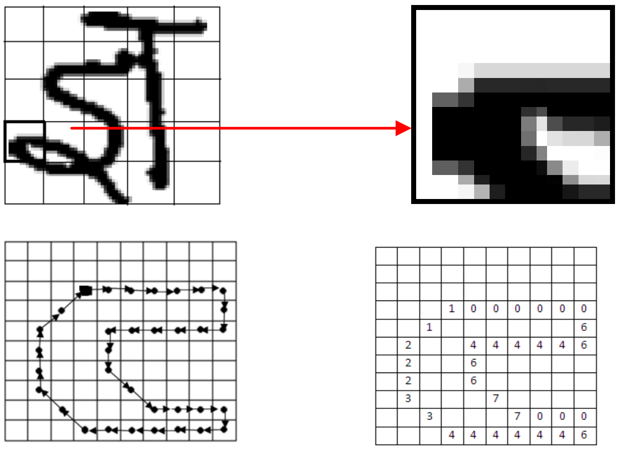

Fig. 3. (a) Character image is divided into 5x5 image block. (b) Block of image corresponding to character. (c) Chain Code process and (d) Chain Code corresponding to block.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

It provides the points which are related to one another position , these points are independent of the coordinate system.

This method is used to connecting all neighboring contour pixels; points and outline coding are captured in this process. Contour following procedure may proceed in clockwise or in counter clockwise direction. We have chosen to proceed in a clockwise direction. The contour image is divided into 5 × 5 blocks as shown in Fig. 3(a). In each of these blocks, each block is 10x10 sizes and applying the chain coding process and darkest spot is show the starting and ending point of chain code as shown in Fig. 3(b) and (c). Finally we get the chain code of the image as shown in Fig. 3(d). The frequency of the direction code is computed and a histogram of chain code is prepared for each block. Thus for 5 × 5 blocks we get 5 × 5 × 8 = 200 features for recognition.

B.2 Zone Based Centroid Feature
This method gives better result even certain preprocessing processes like smoothing, filtering, and skew detection are not considered. This is easy method for implementation and major advantage of this approach for its robustness for small variation. The character image centroid is computed and character image (50x50) is divided into 25 equal zones (10x10). Zone centroid is computed in each zone. This process is repeated for all 25zones ofimage.Figure 3.4 images is divided into 5x5 block and centroid of each is calculated which is the feature of block in this way f1, f2, f3….f25 features are computed.It could be possible that some zone have empty foreground pixels. Hence that zone is assigned by zero. Distance of each zone centroids with image centroid is computed.

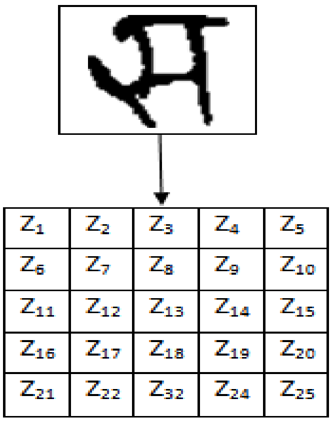

Fig. 4.Character image is divied into 5x5 zones

B.3 Distance Profile Feature
In this technique, profile computes the distance (pixels) from bounding box of character image to outer image of character. The distance is traced horizontal and vertical. We used four profiles of the character which are left, right, bottom and top. Left profiles are computed by traversing horizontally in forward direction and right profiles are computed by traversing backward direction. Similarly, we computed top and bottom profiles by traversing downward and upward direction from top and bottom bounding box. All these profiles show pixels in background and foreground of the image. We normalized dataset to 50x50, distance profiles consists 200 features.

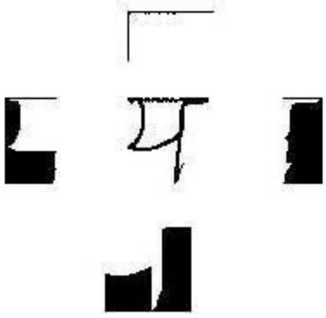

Fig.5. Shows the distance profile features

B.4 Background Directional Distribution Feature
This approach maximizes the discriminative power of the distance features and investigates the rich description of patterns. Directional distances in 8 directions are computed by pixels and both white and black pixels are computed.

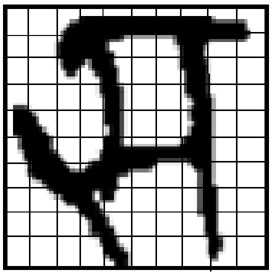

Fig. 6. Chracter image is divided into 10x10 zones for computing the directional feature

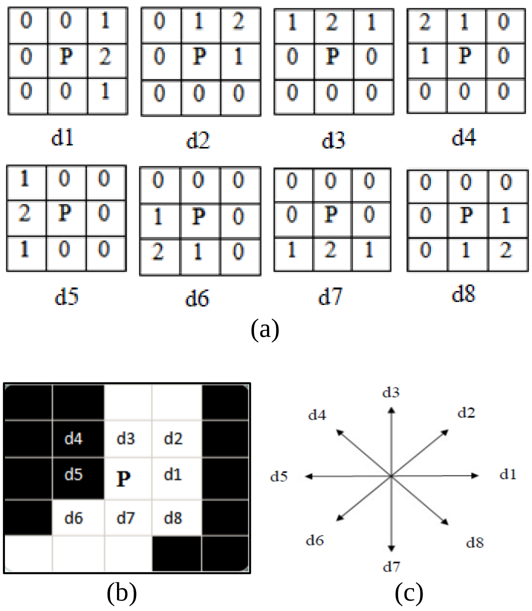

Fig. 7 (a) 8 directions used to compute directional distribution, (b) Masks used to compute directional distribution in different directions. (c) An example of sample

DOI 10.17148/IJARCCE.2015.4825

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

We used mask to compute the directional distribution values as shown in Fig. 7(c). Pixel 'P' is considered as foreground pixels for computing the directional distribution in each direction.To explain the working of BDD features, we compute directional distribution value for foreground pixel 'P' in direction d5 as shown in Fig. 7(c). Mask of d5 is superimposed on the sample image which is coinciding at centered pixels 'P'. d5 mask values is coinciding to background pixel neighbor 'P' as shown in Fig. 7(c) i.e., d5 and d4 (i.e. 1+2) will be feature value in direction d5. In this way we computed the mask value in each direction and then sum all these values in each zone. Each zone comprises total 200 features values for sample image.

C. Classification

Characters are classified by classifier. It works as decision making to classify from one category to another category of classes of characters. Performance of a classifier depends upon suitable features. There are many classification techniques are available but in our approach we used SVM and ANN classifier to achieve the best result possible.

C.1 Support Vector Machines (SVM)

SVM is a classifier separating classes in feature space.The idea of SVM was first shared by Vapnik [8].SVM used to identify a set of linearly separable hyperplanes which are linear functions of the feature space. Among the separable hyperplanes only one hyperplane is chosen and placed such that such that the distance between the classes is maximum [9]. SVM has very high accuracy rate for two class problem but it can also be modified to classify multiclass problem. If a classifier work with a large number of adjustable parameter and therefore large capacity probably learn training set without error. The effective number of parameters is adjusted automatically to match the complexity of the problem [10].

Finding decision hyperplane for simple two class problem is shown in Fig. 8. SVM algorithm for linearly non-separable classes is discussed here.$w^t.x + b = 0$ is the hperplane separating the two classes. Let us consider {Xi, Yi} for i=1,2,3……..,N denoting the training dataset where Yi is the target output for training data Xi. There may be numerous hyperplanes which can separate the two classes but the aim of SVM is to find the one that gives equal and maximum margin from both the classes. Mathematically,the aim of SVM is to maximize the objective function $L(\alpha)$ given by:

$$L(\alpha) = \sum \alpha i - \frac{1}{2}\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha i \alpha j Nj=1Ni=1 YiYj\Phi(Xi)^{T\Phi}(Xj) \quad (1)$$

Subject to constraints $\sum_{j=1}^{N} \alpha iYj=0,\ 0 \le \alpha i \le C;\ \forall i$

Where C is the cost parameter that determines the cost caused by constraint violation, αi is the hyper parameter and $\Phi(\cdot)$ is the feature mapping function. Asking for the maximum-margin linear separator in equation (1) leads to standard Quadratic Programming (QP) problems.With the mentioned constraints, the QP solution leads to the following classification function for SVMs [9].

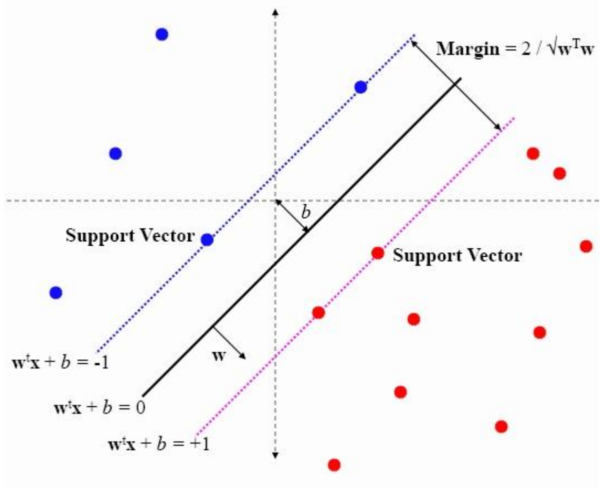

Fig.8. SVM classifier with linear kernel

$$Y=\mathrm{sgn}(W.\ \Phi(Z)+b)$$
$$Y=\mathrm{sgn}\left(\sum_{i=1}^{q} \alpha iYi\,(XiZ+b)\right) \quad (2)$$

Whereαi is the Lagrange multiplier assigned to each training data whose value depend on the role of training the data in the classifier system.The non-zero values of αi correspond to the support vectors that are used to construct the classifier in equation (2), 'q' denotes the number of support vectors. If the feature functions $\Phi(\cdot)$ are chosen with care one can calculate the scalarproducts without actually computing all features, therefore greatly reducingthe computationalcomplexity. In SVM the learning algorithms that only require dot products between the vectors in the originalinput space, and chooses the mapping such that these high-dimensional dot productscan be computed within the original space, by means of a kernel function is called "kernel trick".

$$K(x, xi) = \varphi(x).\ (xi) \quad (3)$$

C.2 Artificial Neural Network (ANN)

We used Multilayer perceptron (MLP) which is a feed-forward artificial neural network that maps sets of inputs to desired outputs [11][12]. This network is used with three layers including hidden layers for four different types of features sets which consisting 200 features for chain code, 25 features for zone centroid, 200 for distance feature and 200 for background directional distribution. In ANN each nodes in network is neuron with a nonlinear activation function except input nodes. In this process supervised learning is used which is called back-propagation to train a network. Results of this experiment are obtained using these four feature extraction method for recognition of Devanagari characters. We are used isolated characters so there is no need of segmenting the characters. Back-propagation approach is used for training to this classifier. We used activation sigmoid function. Total number of inputs is 625 for which is the total feature set of four feature extraction methods. Hidden layer is not fixed we worked on the values 10-60 to get optimal results. Each node of output layer is belongs to a class that's by number of outputs is 44.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 8, August 2015

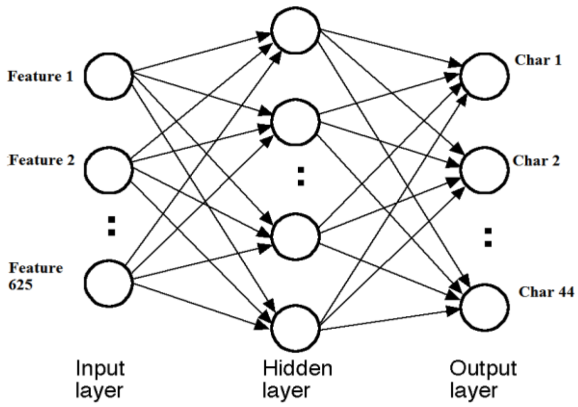

Fig.9. Neural Network diagram for proposed system

IV. EXPERIMENTAL RESULTS

To obtain the recognition results, ANN and SVM classifier has been used to recognize the characters. Results have obtained for Devanagari Recognition on datasets of more than 20,000 sample images respectively. In this approach 80% charactershave been taken for training and 20% characters for testing purpose, while using SVM technique to recognize the characters. In case of ANN we used 80% for training, 5% for validation and 15% testing purpose. System has been trained by neural network and retrain network again to get the best results. With help of this approach we get the following result which shown in table. This technique has been worked also on different sizes of images such as 30x03, 40x40 and 50x50 while corresponding feature vectorshas been taken 413,517 and 625 respectively, to get the best result. We got 97.61 %result using SVM. Table I shows the accuracy/recognition rate obtained for different images sizes by ANN and SVM.When we reduce the size of the images, number of features also reduced.

TABLE I THE RECOGNITION RATE OF DEVANAGARI CHARACTERS (%)

Size of Image	No. of features	ANN	SVM
30x30	413	95.69	97.54
40x40	517	95.13	97.47
50x50	625	94.78	97.61

V. CONCLUSION

The result obtained in this experiment for Devanagari character recognition shows that SVM gives reliable classification accuracy. In this experiment we used four different feature extraction techniques, Chain Code, Zone Based Centroid, Distance Profile, and Background Directional Distribution. These features are applied to SVM classifier. We will work on other feature extraction method to improve the accuracy for our System in future.

REFERENCES

[1] Sharma, Nabin, et al. "Recognition of off-line handwritten devnagari characters using quadratic classifier." Computer Vision, Graphics and Image Processing. Springer Berlin Heidelberg, pp 805-816, 2006.
[2] Pal, Umapada, et al. "Off-line handwritten character recognition of devnagari script." Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on. Vol. 1. IEEE, 2007.
[3] Mori, Shunji, Ching Y. Suen, and Kazuhiko Yamamoto. "Historical review of OCR research and development." Proceedings of the IEEE 80.7 pp 1029-1058, 1992.
[4] Fan, Kuo-Chin, Yuan-Kai Wang, and Tsann-Ran Lay. "Marginal noise removal of document images." Pattern Recognition 35.11 pp 2593-2611, 2002.
[5] Liu, Cheng-Lin, and Katsumi Marukawa. "Pseudo two-dimensional shape normalization methods for handwritten Chinese character recognition." Pattern Recognition 38.12 pp 2242-2255, 2005.
[6] Yang, Feng, Zheng Ma, and Mei Xie. "A novel binarization approach for license plate." In Industrial Electronics and Applications, 2006 1ST IEEE Conference on, pp. 1-4. IEEE, 2006.
[7] Freeman, Herbert, and Ruth Shapira. "Determining the minimum-area encasing rectangle for an arbitrary closed curve." Communications of the ACM 18.7, pp 409-413, 1975.
[8] Support Vector Machine [Online]. Available: {http://en.wikipedia.org/wiki/Support_vector_machine#History/Support_Vector_Machine}.
[9] Kansham Angphun Maring, RenuDhir, "Recognition of Cheising Iyek/Eeyek-Manipuri digits using Support Vector Machines." International Journal of Computer Science & Information Technology, Vol. 1 No. 2, pp 1-6, 2014.
[10] Swethalakshmi, Hariharan, AnithaJayaraman, V. SrinivasaChakravarthy, and C. Chandra Sekhar. "Online handwritten character recognition of Devanagari and Telugu Characters using support vector machines." In Tenth International Workshop on Frontiers in Handwriting Recognition. 2006.
[11] Singh, Raghuraj, C. S. Yadav, PrabhatVerma, and Vibhash Yadav. "Optical Character Recognition (OCR) for Printed Devnagari Script Using Artificial Neural Network." International Journal of Computer Science & Communication 1, No. 1pp 91-95, 2010.
[12] Saba, Tanzila, and AmjadRehman. "Effects of artificially intelligent tools on pattern recognition." International Journal of machine learning and cybernetics 4.2 pp 155-162, 2013.

BIOGRAPHIES

Arjun Singh did his B. Tech in Computer Science and Engineering from Hindustan Collage of Science and Technology, Mathura. He has completed his Masters of Technology in Computer Science and Engineering from Dr. B.R Ambekar National Institute of Technology Jalandhar, Punjab in May 2014. His area of interest are Digital Image Processing, Software Engineering, and Data Structure.

Kansham Angphun Maring did his B.Tech in Computer Science and Engineering from NERIST, Nirjuli, Arunachal Pradesh. He has completed his M.Tech in Computer Science and Engineering from Dr. B.R Ambedkar National Institute of Technology Jalandhar, Punjab, India in May 2014. His area of interest are Digital Image Processing, Pattern Recognition, Natural Language Processing, and Wireless Communication.