

Online Handwritten Gujarati Character Recognition: Two-Layer Classification Approach

Vishal A. Naik and Apurva A. Desai

Abstract- The Gujarati language has large and complex character set and many characters have similar strokes, which makes OCR more challenging. Here we suggest a two-layer classification technique with SVM (RBF) and k-NN classifiers in order to propose a robust online handwritten character recognition for Gujarati language. In the first layer of classification, SVM classifier with the RBF kernel is used and in the second layer, k-NN classifier is used. The training data of second layer classifier is decided based on the outcome of first layer classifier. Training data of a group of characters which are similar to a character returned by first layer classifier, is supplied to k-NN classifier. A hybrid feature set consisting first and second order derivative of pixel values, zoning, and normalized chain code feature. The data set of around 12000 samples was generated from different writers. Around 2000 samples of data set is used for training and rest of the samples are used to test the system. The proposed system has obtained an average accuracy of 94.65% and an average processing time of 0.095 seconds per stroke.

Index Terms- Online Handwritten Character Recognition (OHCR), Handwritten Character Recognition (HCR), Optical Character Recognition (OCR), Support Vector Machine (SVM), Gujarati character Recognition

I. INTRODUCTION

The handwritten character recognition is fast-growing and composite area of machine learning and pattern recognition. The world is moving away from traditional pen and paper kind of communication tool and moving towards a digital world. As the world is becoming digital, more and more people are adopting digital technologies which result in increasing use of handheld devices. Digital world opens many opportunity and areas for researchers. There is a need of online handwritten character recognition system to provide an easy and efficient tool to communicate with digital gadgets in a traditional language and traditional way.

Manuscript revised June 16, 2019 and published on July 10, 2019
Vishal A. Naik, Department of Computer Science, Veer Narmad South Gujarat University, Surat, India
Apurva A. Desai, Department of Computer Science, Veer Narmad South Gujarat University, Surat, India

Indian languages have large and complex character sets which makes a communication with the digital devices more difficult using a simple keyboard. User interaction in handheld devices can be made easy and efficient using online character recognition.

There are 14 major official languages of India besides Hindi and it is required to take serious measures for the development of these languages [1]. Gujarati is a native and official language of Indian state Gujarat. The Gujarati language has a large and complex character set. There are some strokes which are used in multiple characters. For example, Fig. 1 shows character 'સ', 'સ' and 'ઠ'. Stroke of character 'સ' is used in other two characters. There are some characters which has very high similarities with some other stroke. For example, Fig. 2 shows character 'ઇ' and 'ઇ'. Fig. 3 shows character 'પ' and 'ર'. Fig. 4 shows a group of confusing characters which includes 'ન', 'ત', 'મ', 'બ', 'ભ', and 'ખ'.



Fig. 1 Character 'સ', 'સ' and 'ઠ'

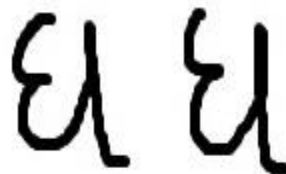


Fig. 2 Character 'ઇ' and 'ઇ'

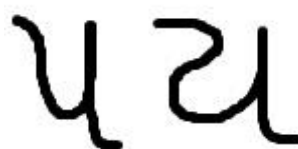


Fig. 3 Character 'પ' and 'ર'

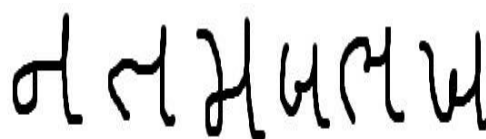


Fig. 4. Character 'ન', 'ત', 'મ', 'બ', 'ભ', and 'ખ'

The process of online handwritten character recognition is very simple. The system tracks the pen movement and an array of active pixels is recorded. These raw pixel coordinate values must be processed to fetch variety of features. The extracted features are passed to the classifier for classification.

Handwritten character recognition can be categorized into offline and online. Notable work is done by many researchers in offline category.

In [2], the segmentation of text line and words for Gujarati handwritten text was presented. Segmentation was performed using projection profile-based algorithm. The system obtained an accuracy of 89.24%. In [3], the Handwritten Gujarati numeral recognition system was presented. The bilinear interpolation method was used for normalization and erosion method was used for thinning. The feature set included four profile vectors. The FFBPNN was used for classification. The system obtained an accuracy of 82%. In [4], the Handwritten Gujarati numeral recognition system was presented. Global threshold, erosion, dilation, and skew correction pre-processing methods were used. The hybrid feature set includes local and aspect ratio features. The k-Nearest Neighbor classifier was used for classification. The system obtained an accuracy of 96.99%. In [5], the zone identification for Gujarati handwritten word was presented. Zone identification was performed using Euclidean transform method. The results found that this method is simple and fast. The result showed an accuracy of 75.2% for the upper zone, 75.2% for the middle zone and 83.6% for the lower zone. In [6], character segmentation method is presented for old typewritten documents. Digitization of the document was performed using a global threshold method. Noise removal was done using a median filter method. Skew correction and line segmentation were performed using Radon transform method. Character segmentation was performed using a vertical profile method. Line segmentation of both languages obtained 100% accuracy. Gujarati words segmentation obtained 94.79% accuracy and Gujarati characters segmentation obtained 65.75% accuracy. Hindi words segmentation obtained 96.18% accuracy. In [7], the handwritten character recognition for Gujarati was presented. The feature set included primary and secondary features. The tree structure was used to compute a subset of characters. Classification was performed using the k-NN classifier. The system obtained an accuracy of 63.1%. In [8], the handwritten Gujarati alphabets identification system was presented. The data set of forty handwritten characters was collected from 199 different writers. The extracted features were aspect ratio, the extent of the alphabet, and image subdivision approach. Classification was performed using the SVM classifier with a polynomial kernel. The result showed an accuracy of 86.66%. Similar work is proposed in [9]. The classification was performed using the Self-

Organizing Map (SOM) with k-NN. Training data was directly provided to the SOM without implementing any feature extraction method. The authors claimed that their algorithm to be faster than other feature-based algorithms. Real-time applications can be built using this algorithm. The system obtained an accuracy of 98.13%. In [10] the authors have compared and experiment with K-NN, SVM and Back Propagation ANN classifiers with different possible options. The feature set included spatial and transform domain features like freeman chain code, Fourier descriptors, discrete cosine transform coefficient. The k-NN, SVM, and BPNN classifiers were compared using 10-fold cross-validation and obtained an accuracy of 85.67%, 93.60%, and 93.00% respectively.

In [11], the SVM was used for classification with experiments and comparison between different kernels with control variables for Gujarati numerals. The hybrid features were implemented which includes zoning, directional chain code features. The data set of 2000 samples were generated from different type of writers. The authors have compared kernel's accuracy and processing time. They have obtained an average accuracy of 95% using polynomial kernel and average processing time per stroke of 0.13 seconds using linear kernel. In [12] the SVM, MLP and k-NN classifiers are tested and compared for Gujarati characters. They have used aspect ratio, zoning and directional chain code features for all classifiers. The system obtained the highest accuracy of 91.63% using SVM with RBF kernel. In [13] the authors have used multi-layer classification using support vector machine classifier. They have used different kernels in both layers. Polynomial kernel is used at first layer and linear kernel is used at second layer. Second classifier is used only if first classifier classifies any confusing character with training data of that confusing character's group. They have used zoning and DP based normalized chain code features. The system obtained an average accuracy of 94.13% and an average processing time per stroke of 0.103 seconds.

In [14], the Online Handwriting Recognition for Tamil was presented. Normalization, resampling using a Gaussian low-pass filter for removing the noise, and an equidistant resampling to remove variations in writing speed methods were used. The feature set includes local and directional features. The classification was performed using the SVM with a DDAG and discriminative classifier. The Viterbi decoding algorithm was used for post-processing. The result showed an accuracy of 95.78% for Malayalam and 95.12% for Telugu characters.

In [15], the recognition of Online Handwriting for Malayalam and Telugu was presented. They have used normalization, resampling using Gaussian low-pass filter, and an equidistant resampling. They have used following features, x & y co-ordinates, moments, area, direction, curvature, length, and aspect ratio. They

have used SVM using a DDAG and discriminative classifier. They have used Viterbi decoding algorithm for post processing. They have achieved accuracy of 95.78% accuracy.

In [16], the handwritten character segmentation and recognition was presented for multiple languages. The proposed character segmentation method uses vertical projection, smoothing, thresholding, and over segment reduction. The k-NN and majority voting principle was used for classification. The geometric properties are used as features. The system obtained an accuracy of 98.79% for segmentation and 99.6% for classification.

In [17], the authors have presented work for handwritten Devanagari character recognition. The combination of quadratic and SVM is used for classification with gradient based directional features. The complete features set has 392-dimensional features. The system has obtained an accuracy of 95.81%.

In [18]–[20], the online Bangla character recognition was presented. The database of 10000 data samples was created which represents 50 different types of strokes. Classification was performed using SVM, SMO, MLP, Random Forest, Bayes net and Naïve Bayes classifier. The authors compared three types of features sets with different classifiers.

The first feature set had 136 feature values which included center of gravity-based global and local features. The result showed an accuracy of 98.26% using SVM, 94.23% using SMO, 91.50% using MLP, 90% using simple logistic, 89.59% using Naïve Bayes, and 88.24% using Bayes net classifier.

The second feature set included a Hausdorff Distance (HD) and Directed HD features. The result showed an accuracy of 95.57% using MLP, 94.85% using simple logistic, 94.43% using SVM, and 88.61% using Bayes net classifier.

The third feature set had 192 feature values which included area features, mass distribution, chord length features. The Krill-Herd algorithm was used to select optimum features from all features. The result showed an accuracy of 98.57% using SMO, 98.10% using MLP, 90.48% using Naïve Bayes, 96.67% using Bayes net, and 97.43% using Random forest classifier. The highest accuracy of 98.57% was obtained using SMO classifier with all features and computation time of 12.05 seconds.

In [21], the authors have presented work on online Bangla character recognition. The database of 15000 data samples was used here. The feature set included transition counts, a center of gravity-based features, and topological features. This features set is compared with different classifiers. The result showed an accuracy of 94.56% using SVM, 93.07% using MLP, 91.70% using simple logistic, and 86.45% using Bayes net classifier. After parameter tuning of SVM, an accuracy of 95.49% was obtained.

In [22], the authors have presented work on online handwritten English character recognition. The Hidden Markov Model was used for classification. The comparison between Gaussian mixture model-HMM and hybrid deep neural network-HMM was performed. The feature set included stroke velocity and raw points-based features. The velocity profile was created for every stroke which included sinusoidal based horizontal, vertical and zero-crossing velocities. The system was tested on the UNIPEN and the IRONOFF databases. For the UNIPEN database, the highest accuracy obtained was 99.04% for digits using GMM-HMM, 97.43% for upper case characters using DNN-HMM, and 95.42% for lower case characters using DNN-HMM. For the IRONOFF database, the highest accuracy obtained was 98.53% for digits using DNN-HMM, 95.59% for upper case characters using DNN-HMM, and 93.65% for lower case characters using DNN-HMM. For the IRONOFF word database, the highest accuracy obtained was 91.07% using DNN-HMM.

II. PRE-PROCESSING

Different pre-processing methods are used to make stroke level corrections before further processing. The simple pre-processing methods are used here. The normalization of stroke size and smoothing method are used in pre-processing. Each stroke should have same size so that similar features can be extracted. Bilinear interpolation method is used to make a stroke size normalized. Interpolation will be performed in X & Y directions. Strokes should be smoothed before feature extraction to remove additional noise. Nonlinear median filter is used as a smoothing method.

III. FEATURE EXTRACTION

Raw pixel coordinate represents variety of information which needs to be fetched out as meaningful unique information. The feature set includes various features to represent variety of information about the stroke. Various feature extraction methods are used to fetch variety of important values from raw coordinates. These variety features values are the input for the classification algorithms. Features is categorized into global and local features and different types of structural, statistical and hybrid methods to extract such features.

The hybrid feature set is used here which has the derivative of x & y, 16 zoning values, and normalized chain code features. The first and second order derivative of x & y at each pixel is calculated independently of each other. The first and second order derivative provides information about the change in the trajectory at the current pixel. The first and second order derivatives are calculated using the formulas 1 to 4 [23].

The second order derivative values for character “u” are, d^2x is 3,3,3,2,2,2,2,2,2,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,3,3,3,4,4,4,3, 3 and d^2y is -1,-1,0,0,0,1,1,2,2,2,3,3,3,3,3,3,3,3,3,3,3,2,2,2,2,2,2,2,2,1,1,1,0,0,-1,-2,-2,-3.

$$x'(j) = \frac{\sum_{i=1}^2 i(x(j+i) - x(j-i))}{2 \sum_{i=1}^2 i^2} \quad (1)$$

$$y'(j) = \frac{\sum_{i=1}^2 i(y(j+i) - y(j-i))}{2 \sum_{i=1}^2 i^2} \quad (2)$$

$$x''(j) = \frac{\sum_{i=1}^2 i(x'(j+i) - x'(j-i))}{2 \sum_{i=1}^2 i^2} \quad (3)$$

$$y''(j) = \frac{\sum_{i=1}^2 i(y'(j+i) - y'(j-i))}{2 \sum_{i=1}^2 i^2} \quad (4)$$

Key points are selected from an input stroke after every 10% of pixels which are ten key points in total. These key points are used to measure the curve direction between each pair of key points. Different curve direction values are shown in Fig. 5 based on which chain code can be generated[24]. Fig. 6 shows freeman chain code values for character “u” are 5, 12, 13, 3, 0, 1, 5, 11, and 13. Starting value of the chain code for the same character can be different based on writing style. The chain code can be starting point invariant, the starting point should be redefined, starting with the minimum value of chain code in the circular order.

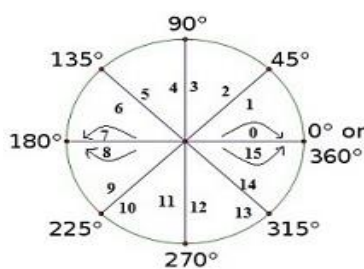


Fig. 5 Constant directional values

The chain code values for character “u” are 0, 1, 5, 11, 13, 5, 12, 13, and 3. If the user writes strokes with some skew or the stroke rotates then its chain code values will be different. The difference between the pair of chain code values are computed by counting the number of code values between pair of code values counter-clockwise. The chain code values for character “u” is 1, 4, 6, 2, 8, 7, 1, 6.

In zoning, each stroke is divided into 16 equal sized zones. The number of active pixels in each zone is computed and its percentage is considered as a feature. Fig. 7 shows stroke of character “u” which is divided into 16 equal sized zones.

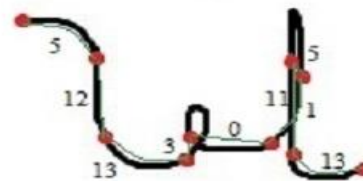


Fig. 6 Key points and chain code values of “u”

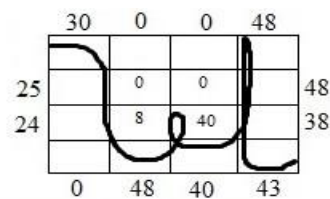


Fig. 7. Pixel distribution of “u” into 16 equal zones

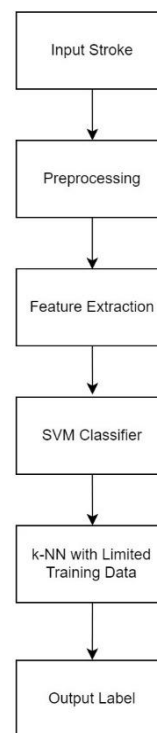


Fig. 8. Steps of the proposed system

IV. CLASSIFICATION

The two-layer classification approach is proposed using support Vector Machine classifier with Radial Basis Function kernel at the first layer and k- Nearest Neighbor at the second layer. The SVM classifier is selected based on the comparison between SVM, MLP, and k-NN[12]. The k-NN classifier is selected because it requires low resources. Fig. 8 shows different steps required in the proposed system.

SVM algorithm is using a supervised learning approach which implicitly transforms data using different mathematics-based kernels. By using different kernels, SVM can select appropriate threshold. SVM tries to find global optimum and unique solution. A constant value of control variables is $\gamma = 0.002$ and $C = 1$ which is selected after trial and error method.

k-NN is a distance-based classifier. It uses Euclidean distance or Hamming distance to calculate a distance between testing and training data. K nearest data samples define the class label. Different values of k results in different accuracy and processing time. The value of k is set to 3 after experimenting with different values of k.

In the two-layer classification approach, the first layer classifier uses a complete training data set. The second layer classifier uses a limited or selected training data based on the character returned by the first classifier. For second layer, training data is divided into eight groups based on the similarities between characters. Each group is consisting of characters which is similar to each other. Figure 4.8 shows a group of character consisting of six half characters and six full characters. If first layer classifier returns a character “૫” which belongs to the group shown in figure 9 then the training data of these 12 characters are used for training of second layer classifier.

No.	Stroke	No.	Stroke
1	૫	1	૫
2	૫	2	૫
3	૫	3	૫
4	૫	4	૫
5	૫	5	૫
6	૫	6	૫

Fig. 9 Training data group

V. RESULTS AND DISCUSSION

Training of the system will be performed in two layers. In the first layer, all training data is used to train SVM and limited training data of a single group is used to train k-NN. There are total 08 groups are created for second layer training. Selection of a group for second layer training is based on the result of first layer. 10000 samples are used to test the proposed system. The training and testing are performed on a touch screen personal computer using developed Graphical User Interface (GUI) system.

Table I describes performance comparison between different system approaches. The system has obtained an average accuracy of 95.65% using proposed two-layer classification approach and it took an average processing time of 0.095 seconds per stroke. The performance of the proposed system is increased

compared to the multilayer classification approach[13]. An average accuracy of 94.13% is obtained using a multilayer system and it took an average processing time of 0.103 seconds per stroke.

Table I. Performance Comparison

Method	Accuracy (%)	Avg. Processing time (Seconds)
Multilayer system	94.13	0.103
Proposed system	95.65	0.095

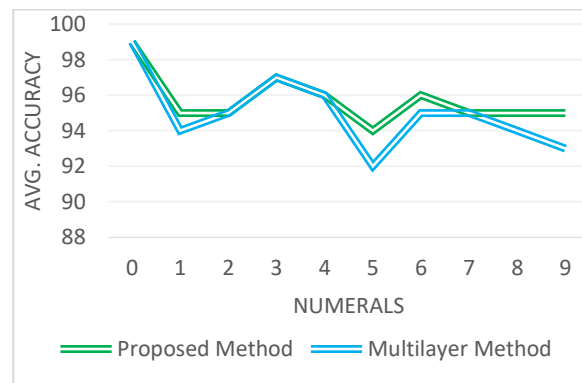


Fig. 10. Accuracy of Gujarati Numerals

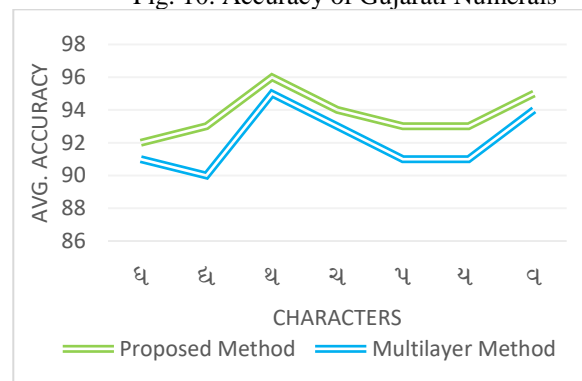


Fig. 11. Accuracy of Gujarati characters

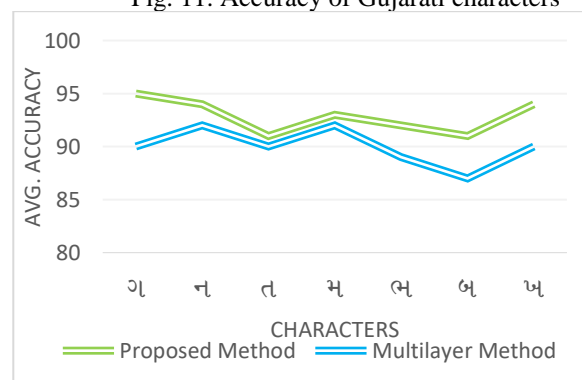


Fig. 12. Accuracy of Gujarati characters

Fig. 10 shows an average accuracy of numerals. The two-layer method obtained 95.7% and multilayer method obtained 95% accuracy for numerals.

Fig. 11 shows an average accuracy of a group 1 of similar characters. The two-layer method obtained 93.71% and multilayer method obtained 89.83% accuracy for a group of similar characters. Fig. 12 shows an average accuracy of a group 2 of similar characters. The two-layer method obtained 93.83% and multilayer method obtained 82.14% accuracy for another group of similar characters.

Fig. 13 shows an average accuracy of a group 1 of multi-stroke characters. The two-layer method obtained 96.75% and multilayer method obtained 95% accuracy for a group of multi-stroke characters.

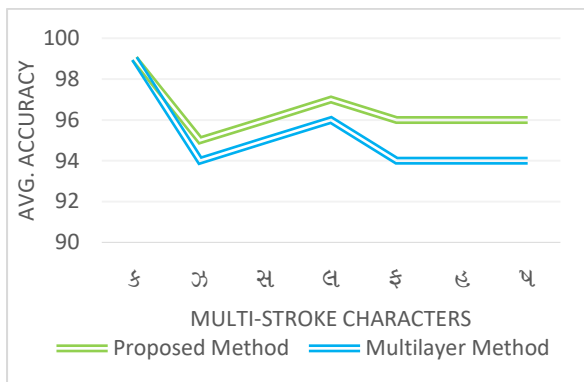


Fig. 13. Accuracy of multi-stroke characters

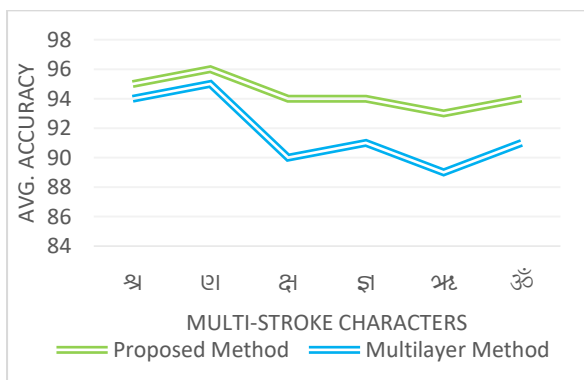


Fig. 14. Accuracy of multi-stroke characters

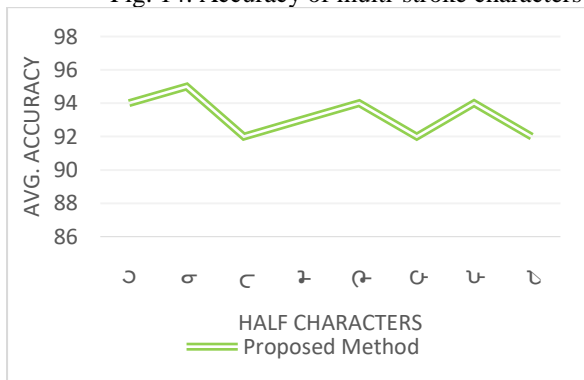


Fig. 15. Accuracy of half characters

Fig. 14 shows an average accuracy of a group 2 of multi-stroke characters. The two-layer method obtained 94.4% and multilayer method obtained 91.8% accuracy for another group of multi-stroke characters.

Fig. 15 shows an average accuracy of a group 1 of half characters. The two-layer method obtained 95% accuracy for a group of half characters. Fig. 16 shows an average accuracy of a group 2 of half characters. The two-layer method obtained 90.33% accuracy for another group of half characters.

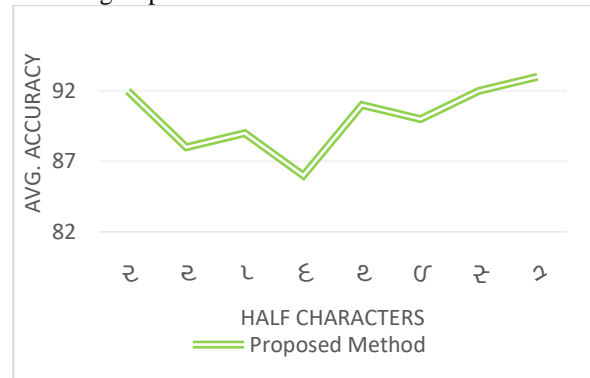


Fig. 16. Accuracy of half characters

VI. CONCLUSION

The two-layer classification system has used SVM with RBF kernel and k-NN classifiers with a hybrid feature set. The two-layer classification system has achieved an average accuracy of 95.64% which is better than multilayer system and 0.095 seconds of average execution time per stroke which is lesser than multilayer system. The two-layer system's performance is directly based on the result of first layer classifier. If the first layer classifier fails then second layer classifier leads to the wrong classification.

REFERENCES

- [1] "Official languages of India." [Online]. Available: <http://rajbhasha.nic.in/en/official-language-resolution-1968>. [Accessed: 27-June-2019].
- [2] C. Patel and A. Desai, "Segmentation of text lines into words for Gujarati handwritten text," *Proc. 2010 Int. Conf. Signal Image Process. ICSIP 2010*, pp. 130–134, 2010.
- [3] A. A. Desai, "Gujarati handwritten numeral optical character reorganization through neural network," *J. Pattern Recognit.*, vol. 43, no. 7, pp. 2582–2589, 2010.
- [4] A. A. Desai, "Handwritten Gujarati Numeral Optical Character Recognition using Hybrid Feature Extraction Technique," *Int. Conf. Image Process. Comput. Vision, Pattern Recognition, IPCV*, 2010.
- [5] C. Patel and A. Desai, "Zone identification for Gujarati handwritten word," *Proc. - 2nd Int. Conf. Emerg. Appl. Inf. Technol. EAIT 2011*, pp. 194–197, 2011.
- [6] A. Desai, "Segmentation of Characters from old Typewritten Documents using Radon Transform," *Int. J. Comput. Appl.*, vol. 37, no. 9, pp. 10–15, 2012.
- [7] C. Patel and A. Desai, "Gujarati Handwritten Character Recognition Using Hybrid Method Based on Binary Tree-Classifer And K-Nearest Neighbour," *Int. J. Eng. Res. Technol.*, vol. 2, no. 6, pp. 2337–2345, 2013.
- [8] A. A. Desai, "Support vector machine for identification of handwritten Gujarati alphabets using hybrid feature space," *CSI Trans. ICT*, vol. 2, no. January, pp. 235–241, 2015.
- [9] M. M. Goswami and S. K. Mitra, "Classification of Printed Gujarati Characters Using Low-Level," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 15, 4, Artic. 25 (April 2016),

26 pages. DOI <http://dx.doi.org/10.1145/2856105>, vol. 15, no. 4, 2016.

- [10] A. N. Vyas and M. M. Goswami, "Classification of handwritten Gujarati numerals," *2015 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2015*, pp. 1231–1237, 2015.
- [11] V. A. Naik and A. A. Desai, "Online Handwritten Gujarati Numeral Recognition Using Support Vector Machine," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 9, pp. 416–421, 2018.
- [12] V. A. Naik and A. A. Desai, "Online Handwritten Gujarati Character Recognition Using SVM, MLP, And k-NN," in *8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Delhi, 2017*, pp. 1–6.
- [13] V. A. Naik and A. A. Desai, "Multi-Layer Classification Approach for Online Handwritten Gujarati Character Recognition," *Comput. Intell. Theor. Appl. Futur. Dir. - Vol. II. Adv. Intell. Syst. Comput.*, vol. 799, pp. 595–606, 2018.
- [14] K. H. Aparna, V. Subramanian, M. Kasirajan, G. V. Prakash, and V. S. Chakravarthy, "Online Handwriting Recognition for Tamil," *Ninth Int. Work. Front. Handwrit. Recognit.*, pp. 438–443, 2004.
- [15] A. Arora and A. M. Namboodiri, "A hybrid model for recognition of online handwriting in Indian scripts," in *International Conference on Frontiers in Handwriting Recognition, ICFHR 2010*, 2010, pp. 433–438.
- [16] P. Sahare and S. B. Dhok, "Robust Character Segmentation and Recognition Schemes for Multilingual Indian Document Images," *IETE Tech. Rev. (Institution Electron. Telecommun. Eng. India)*, vol. 4602, pp. 1–14, 2018.
- [17] M. Bhalerao, S. Bonde, A. Nandedkar, and S. Pilawan, "Combined Classifier Approach for Offline Handwritten Devanagari Character Recognition Using Multiple Features," *Comput. Vis. Bio Inspired Comput.*, pp. 45–54, 2018.
- [18] S. Sen, A. Bhattacharyya, A. Das, R. Sarkar, and K. Roy, "Design of Novel Feature Vector for Recognition of Online Handwritten Bangla Basic Characters," in *the First International Conference on Intelligent Computing and Communication. Advances in Intelligent Systems and Computing*, 2016, vol. 458, pp. 485–494.
- [19] S. Sen, R. Sarkar, K. Roy, and N. Hori, "Recognize Online Handwritten Bangla Characters Using Hausdorff Distance-Based Feature," in *the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications. Advances in Intelligent Systems and Computing*, 2017, vol. 515, pp. 541–549.
- [20] S. Sen, M. Mitra, A. Bhattacharyya, R. Sarkar, F. Schwenker, and K. Roy, "Feature Selection for Recognition of Online Handwritten Bangla Characters," *Neural Process. Lett.*, 2019.
- [21] S. Sen, A. Bhattacharyya, P. K. Singh, R. A. M. Sarkar, K. Roy, and D. Doermann, "Application of Structural and Topological Features to Recognize Online Handwritten Bangla Characters," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 17, no. 3, 2018.
- [22] H. Choudhury and S. R. Mahadeva Prasanna, "Representation of Online Handwriting using Multi-component Sinusoidal Model," *Pattern Recognit.*, 2019.
- [23] G. S. Reddy, B. Sarma, R. K. Naik, S. R. M. Prasanna, and C. Mahanta, "Assamese online handwritten digit recognition system using hidden Markov models," *Proceeding Work. Doc. Anal. Recognit. - DAR '12*, p. 108, 2012.
- [24] X. Q. Bo YU, Lei Guo, Tianyun Zhao, "A curve matching algorithm based on Freeman chain code," *Int. Conf. Intell. Comput. Intell. Syst.*, pp. 669–672, 2010.

AUTHORS PROFILE



Mr. V. A. Naik received Master of Science (I.T.) from Veer Narmad South Gujarat University, Surat, Gujarat, India in 2010. He is currently pursuing Ph.D. from Veer Narmad South Gujarat University, Surat, Gujarat, India. His main research work focuses on handwritten character recognition, pattern matching, and image processing. He has 8 years of teaching experience.



Mr. A. A. Desai received a Ph.D. degree from the South Gujarat University, Gujarat, India in 1997. He is a professor with the Department of Computer Science, Veer Narmad South Gujarat University (VNSGU) India since 2004. He is also Dean of Faculty of Computer Science and Information Technology since 2012. He is involved in many projects during his long academic and research experience of over 25 years. His research activities are in the area of Digital Image Processing, Pattern Recognition, Natural Language Processing, and Data Mining.