ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

An Ensemble Method for Character Recognition Using Machine Learning Techniques

Nitisha Uday Govekar

M.E Student, Dept. of I.T., Padre Conceicao College of Engineering, Verna-Goa, India

ABSTRACT: There are different learning-based classification techniques that are used individually to successfully recognize characters. But each classifier when used individually has some drawbacks, such as, generating erroneous words; besides, different classification techniques make relatively independent mistakes. This project describes an ensemble method for character recognition using diverse machine learning techniques.

The proposed ensemble method takes the bitmap image as input and shows the recognized character as its output. The base classifiers used here are Multi-Layer Perceptron (MLP) and Support Vector Machines (SVM). Each classifier is trained and tested individually and the optimal result is obtained by combining the MLP and SVM outputs. Each training data subset is used to train different classifier of the same type. Their outputs are then analyzed to generate the rules that are used to combine their outputs to give the ensemble decision. For any given test sample, the class chosen by most classifiers is the ensemble decision.

KEYWORDS: Ensemble methods, Multilayer Perceptron, Support Vector Machines

I. INTRODUCTION

Character Recognition has been an active area of research in the field of image processing and pattern recognition and due to its diverse applicable environment; it continues to be a challenging research topic. Optical Character Recognition (OCR) Systems [7] are widely used to process scanned text into text usable by computers. We observe that current OCR systems have bad performance on domain-specific papers, even generating lots of incorrect words; besides, different OCR systems [19] make relatively independent mistakes. In most OCR systems the recognition proceeds as a two-pass process. In the first pass, an attempt is made to recognize each word in turn. Each word that is satisfactory is passed to an adaptive classifier as training data. The adaptive classifier then gets a chance to more accurately recognize text lower down the page.

There are different machine learning-based classification techniques that are used individually to successfully recognize characters. But each classifier when used individually has some drawbacks. Over time, different hybrid models [4, 5] comprising of a pair of diverse machine learning techniques are proposed to improve the recognition accuracy.

In this project, the recognition accuracy is aimed to improve by training an ensemble of classifiers. The base classifiers chosen in this project are Multi-Layer Perceptron (MLP) and Support Vector Machines (SVM). Each classifier is trained and tested individually and the optimal result of recognition is obtained by an ensemble method of Bagging. Diversity in bagging is obtained by using bootstrapped replicas of the training data: different training data subsets are randomly drawn—with replacement—from the entire training data. Each training data subset is used to train a different classifier of the same type. Individual classifiers are then combined by taking a majority vote of their decisions. For any given instance, the class chosen by most classifiers is the ensemble decision.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

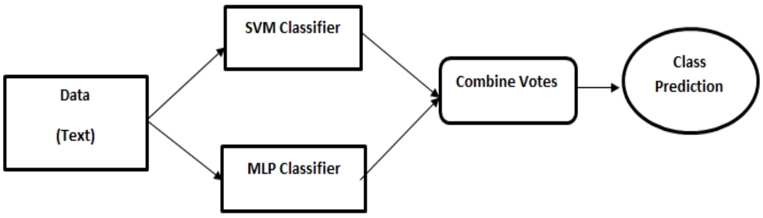

Fig.1 Architectural Diagram of Ensemble Classifier

The Fig.1 depicts the architectural diagram of the ensemble classifier which first generates the MLP output and SVM output and then combines these results to generate the ensemble output. So, the ensemble classifier generates the base classifiers in a way so that they complement each other and produce better accuracy than their base counterparts. This approach attempts to overcome the problem that those of the individual classifiers. The accuracy depends to a lot of extent on how the feature sets are combined in ensemble architecture.

II. RELATED WORK

The optical character recognition (OCR) systems are widely used in digitizing books and texts so that they can be used in electronically search, machine translation and text mining. Usually OCR will do pre-processing, character recognition and post- processing.In creating an Ensemble system [3, 13] the first step is to decide on the base classifiers to be used. There are different machine learning-based classification techniques [2, 6, 8, 9, 14] that are used individually to successfully recognize characters. But each classifier when used individually has some drawbacks as in [19]. Over time, different hybrid models [4, 5] comprising of a pair of diverse machine learning techniques are proposed to improve the recognition accuracy.The results show that the hybrid MLP-SVM recognizer improves significantly the performance in terms of recognition rate and error rate compared with one MLP network for one classification task of characters. Also, the KNN-SVM hybrid model showed significant improvements in terms of recognition rates. This proposed model is compared with the MLP technique and concludes that the results show that the hybrid model performs better. Another approach [19] is used to train an ensemble system from multiple open-source OCR systems, which chooses outputs candidates generated by each OCR, and train the system with machine learning techniques, it gives satisfactory results.

In this paper, the recognition accuracy is aimed to improve by training an ensemble of classifiers. Many papers [10, 11, 12, 16, 17, 18, 20] that compare different machine learning techniques exist. Using the analysis in [10, 11] the base classifiers decided upon are: Multi-Layer Perceptron MLP andSupport Vector Machines SVM. These two techniques are chosen in this project mainly because the recognition rate for recognizing the characters is more accurate than other existing machine learning techniques.Previous research [1] has shown that an ensemble is often more accurate than any of the single classifiers in the ensemble. Our system is like this, to combine the two results of individual classifiers together and make one decision.

III. RESEARCH METHOD

A. Data Preparation

The base classifiers: MLP and SVM have been trained on three different fonts: Latin Arial, Latin Tahoma and Latin Times New Roman. The MLP and SVM take bitmap images of the machine-printed text as input and perform training individually.

There are three kinds of characters considered for each font: uppercase, lowercase and digits. Once trained and tested individually the ensemble classifier recognizes the characters and displays the output as text.In case of MLP technique, the process of analyzing the image begins with detecting the character symbols by examining pixels that form the vital part of the input set preparation in both of the training and testing phases. The symbolic extensions are recognized based on the color value of individual pixels in an input image file, reconsidered as black RGB (255,0,0,0) or white RGB (255,255,255,255). The input images are restricted to be in the bitmap form of any resolution which can

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

be used to map the existing techniques in the Microsoft Visual Studio environment that is an internal bitmap object. The procedure also considers that the input bitmap image is composed of only characters.

In a training phase all such character symbols from the trainer set image file that is .cts files are mapped into their corresponding font types. The trainer set would also contain a file composed of character strings that directly map the input symbol images to their corresponding desired output of the training. A sample example of the trainer set is illustrated below:

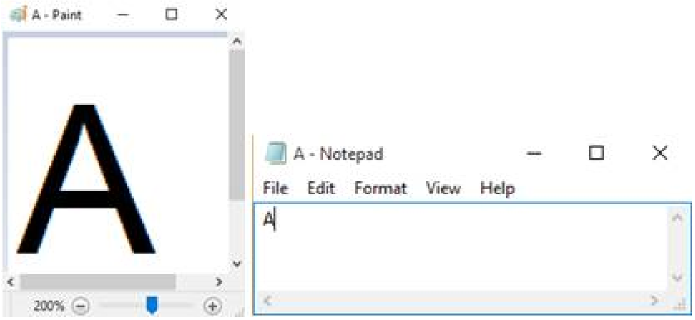

Fig.2 Input Image and its corresponding text file

In case of SVM technique, the data prepared produces the bitmap images for each piece of character and transform it into binary 0 for white pixel and 1 for black pixel. The image processing then consists of identifying the black or gray pixel. “Fig.3” is the matrix of 32x32 dimensions of a character.

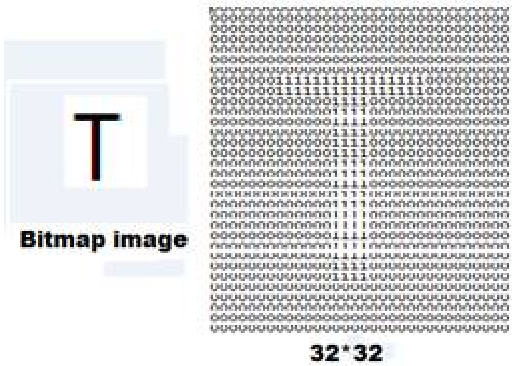

Fig.3 Sample binary data to represent a character for computing

B. Proposed Ensemble Classifier

Once the data is prepared to train the two base classifiers, we now have a multi-classification problem. We try two learning algorithms to achieve an ensemble system: Multilayer Perceptron and Multi-class Support Vector Machine. The ensemble system consists of two phases: training and testing. In the training phase, the two base classifiers are trained and tested individually and in the testing phase the ensemble classifier analyzes the two individual outputs of two base classifiers and gives the majority output as the ensemble output.

The ensemble system uses Breiman’s bagging, short for bootstrap aggregating,[1] the earliest ensemble based algorithms. Diversity in bagging is obtained by using bootstrapped replicas of the training data: different training data subsets are randomly drawn—with replacement—from the entire training data. Each training data subset is used to train a different classifier of the same type. Individual classifiers are then combined by using the hardcoded rules that are generated by analyzing the base classifiers outputs.

A. Training:

I. MLP Training and Testing

The Multi-Layer Perceptron Neural Networks [8] have been applied successfully to solve some difficult and diverse problems by training them in supervised manner with a highly popular algorithm known as error back-propagation

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

algorithm. The error back-propagation algorithm [22] is based on error-correction learning rule.For the purpose of this project the MLP network is composed of 3 layers, one input, one hidden and one output as in "Fig 4".

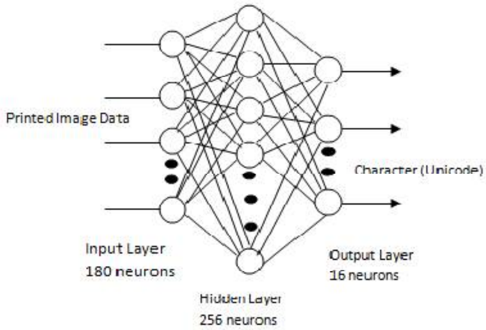

Fig.4 Formation of MLP network

The input layer of the network is composed of 150 neurons that receive pixel binary data as input from a 10x15 symbol pixel matrix. This matrix is formed corresponding to the average height and width of character images present in the sample text that can be considered without introducing any significant pixel noise. On the basis of optimal results on a trial and error basis, the hidden layer is set to 250 neurons. The corresponding output layer constitutes 16 neurons that are mapped from the 16-bits of character output.

Begin with initializing the weights of a random function to an initial random number which is in the range of two preset integers namely ±weight bias. The weight bias is decided upon from the trial and error analysis to correspond to the average weights.

The steps used to train the network are

Step 1: The input image that is scanned must be a bitmap image of any resolution or a binary image.

Step 2: That image is then digitized. By digitizing we mean that a rectangular matrix of 0s and 1s needs to be formed from the input image.

Step 3: When encountered with a 1-white and 0-black pixel, all RGB values must be converted to 0s and 1s.This formed matrix of dots represents two-dimensional array of bits. Digitization converts a bitmap image into a binary image using an adaptive thresholding. This process is sometimes called as binarization.

Step 4: Next step is to detect the line and boundary which consists of identifying points in a input image at which the character left, right, top and bottom are calculated.

Step 5: The feed-forward approach is used to train all the unique features, which are basically the inputs, one middle layer- hidden layer integrates and collaborates similar features and if need exists adjusts the inputs by changing or modifying the weight values, and lastly the output layer finds the overall score of the network.

Once the network is formed you can use the saved network files to test the data.

II. SVM Training and Testing

SVM classifier is trained using the Accord.Net library available as open source. It uses extracted features of 1024 blocks array which is the matrix of 32x32 dimensions obtained from extraction.

The training module is designed with character-based database as the input by dividing them into different vertical level. Here we use multi-class SVM with One-Again-One (OAO) strategy with kernels of non-linear SVM. The OAO strategy will eliminate the duplication option during classify by k = (n (n - 1)) / 2.

The Gaussian kernel [21] is used in this project. The SVM classifier with the Gaussian kernel is simply a weighted linear combination of the kernel function computed between a data point and each of the support vectors. The RBF (Gaussian kernel) is given as:

$$k(x, y) = \exp\left(\frac{-||x-y||^2}{2\sigma^2}\right)$$

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

The steps used to train the network are:

Step 1: The database of scanned images forms the input to the datagridview in Microsoft Visual Project which is the bitmap images of characters.

Step 2: These bitmap images are then used to compute the feature set.

Step 3: Create the Multi-class Support Vector Machine using the selected Gaussian Kernel parameters, feature set and the total number of character labels.

Step 4: Create the learning algorithm, SMO using the machine and the training data.

Step 5: Configure the learning algorithm and run the machine

Once finished training, similar procedure is used to test the data.

B. Testing:

The testing phase for the ensemble system consists of segmentation, feature extraction, classification and character assembling as shown in “Fig. 5”.

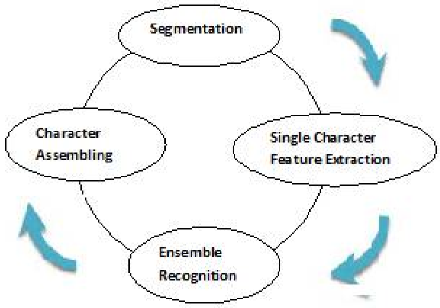

Fig.5 Ensemble System Features

- Segmentation

The segmentation module begins detecting the edges by scanning the image document pixel by pixel within a line separation under the main body block from top-to-bottom and from left-to-right to extract black block. When the system finds the black pixel, it will start to detect surrounding connected black blocks until no more black pixel founds [3, 13]. The segment starts first detect line and then by each line it segments each character into different pieces. The text- detection algorithms in the previous chapter explains the procedure.

- Feature Extraction

During the character segmentation, each character has been extracted and the system calculates the object feature. In case of SVM the character object is transformed into binary data and store as matrix of 32x32 dimensions and then it extracts 1024 (32*32) features from the single object and stores each binary character of 0 or 1 into each block of the vector array. In case of MLP, the symbol image is mapped into a corresponding two dimensional binary matrix. As in training phase, we employ a sampling strategy which will map the detected symbol image in the input into a 10×15 binary matrix with only 150 elements.

- Classification and Recognition

For each detected character, the MLP and the SVM classifier each assign a class label and gives the output individually as in the training phase. Once finished with all the characters we perform majority voting for that test instance using the rules that we have hard-coded and display the output of one of the base classifier as ensemble output for each of the detected character symbol. The output chosen is dependent on the accuracy of each classifier for that instance only.

The rules are generated by observing the results of the two classification techniques. For instance we observe that the for the Arial font, the MLP technique incorrectly classifies the character ‘A’ as ‘Q’ while the SVM correctly classifies it so we include this as a rule such that

If the output of MLP = = ‘Q’

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Then set the output of Ensemble = output of SVM.

Similarly the SVM classifier sometimes incorrectly classifies 'p' as either 't' or 'P' while the MLP classifier classifies it correctly, so we append the rule as follows

If the output of SVM = = 't' or 'P'

Then set the output of Ensemble = output of MLP.

Consider another example in case of the Times New Roman font, we observe that the SVM classifier sometimes classify 't' as 'I' which is incorrect so we append the rule such as the following to encounter that mistake:

If the output of SVM = = 'I'

Then set the output of Ensemble = output of MLP.

In this way we used the observations of MLP and SVM classifier outputs to hard-code the rules to give the ensemble decision for the input sample. So by increasing the number of input sample images which are not seen by the MLP and SVM we can observe their outputs and if there is any discrepancy in their outputs then we can then append new rules to the already existing ones.

Fig. 6 shows the interface to the ensemble system. We load the two classifiers, the input image and perform the SVM and MLP classification for the loaded image and using their results we perform the ensemble classification. We can also save the ensemble output in a corresponding text file.

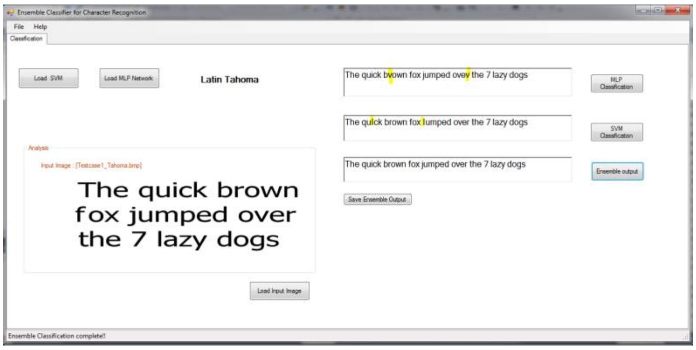

Fig.6 Ensemble System Interface

IV. RESULTS AND DISCUSSIONS

The training module consists of two main parts. The first part consists of creating a MLP classifier and the second part consists of creating a SVM classifier. The parameters used for MLP classifier are: Learning rate = 150, Sigmoid Slope = 0.014, Weight bias = 30 (determined by trial and error), Number of Epochs = 300-600 (depending on the complexity of the font types). Similarly the parameters used for SVM classifier are:Sigma value: 5.22 (since kernel function used is a Gaussian kernel), Epsilon: 0.001, Complexity: 1.00, Tolerance: 0.200.

The testing module basically consists of analyzing the outputs of the two base classifiers and then forming the rules. These rules will in turn be used to take the ensemble decision. This means that for some characters in the sample text, MLP decision may be accurate while for some, SVM decision may be accurate, so the ensemble decision for the text should be a combination of both.

The recognition rate, also known as the accuracy of the classifier is used to measure the percentage of characters that are correctly classified.The typical formula for calculating the accuracy is $\frac{(\mathrm{TP}+\mathrm{TN})}{Total}$ where TP is the number of true positive and TN is the number of true negative.An accuracy of 100% means that the detected characters are exactly the same as those that are present in the input image. We have tabulated the MLP, SVM and the ensemble classifier recognition rate for each of the three fonts. Table1 showsthe MLP, SVM and the ensemble classifier recognition rate for the Latin Arial font.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Table1: Recognition rate for the Latin Arial Font

Test Set	Recognition Rate (in %)		
	MLP Output	**SVM Output**	**Ensemble Output**
TestCase1: The quick brown fox jumped over the 3 lazy dogs	100	97.87	100
TestCase2: Vinya bought a new BMW yesterday She owns 5 cars already	100	95.83	100
TestCase3: Confusion Characters wvrmnQOqpbdyguvS2ZCG	100	95	100
TestCase4: Nia and Mia use the same perfume	100	90.62	100
TestCase5: The Taj Mahal was built by the Emperor Shah Jahan	95.83	93.75	95.83
TestCase6: Failure is a stepping stone to success	100	89.47	100
TestCase7: AV affa LTaa sertLV	84.21	89.47	89.47
TestCase8: Reason is Orthogonal inseparability	100	97.14	100
TestCase9: My name is Nitisha Govekar	100	96.15	100
TestCase10: My favorite cartoon show is Oggy and the Cockroaches	100	92.30	100
Total Recognition rate	98.00	84.81	98.53

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Table2 showsthe MLP, SVM and the ensemble classifier recognition rate for the Latin Tahoma font.

Table2: Recognition rate for the Latin Tahoma Font

Test Set	**Recognition Rate (in %)**		
	MLP Output	**SVM Output**	**Ensemble Output**
TestCase1: The quick brown fox jumped over the 3 lazy dogs	95.74	95.74	100
TestCase2: Vinya bought a new BMW yesterday She owns 5 cars already	95.45	100	100
TestCase3: Confusion Characters wvrmnQOqpbdyguvS2ZCG	92.5	100	100
TestCase4: Nia and Mia use the same perfume	96.87	100	100
TestCase5: The Taj Mahal was built by the Emperor Shah Jahan	97.96	100	100
TestCase6: Failure is a stepping stone to success	97.36	100	100
TestCase7: AV affa LTaa sertLV	47.36	52.63	52.63
TestCase8: Reason is Orthogonal inseparability	91.43	94.28	94.28
TestCase9: My name is Nitisha Govekar	96.15	100	100
TestCase10: My favorite cartoon show is Oggy and the Cockroaches	94.23	96.15	96.15
Total Recognition rate	90.5	93.88	94.3

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Table3 showsthe MLP, SVM and the ensemble classifier recognition rate for the Latin Times New Roman font.

Table3: Recognition rate for the Latin Times New Roman Font

Test Set	**Recognition Rate (in %)**		
	MLP Output	**SVM Output**	**Ensemble Output**
TestCase1: The quick brown fox jumped over the 3 lazy dogs	97.87	97.87	100
TestCase2: Vinya bought a new BMW yesterday She owns 5 cars already	100	95.83	100
TestCase3: Confusion Characters wvrmnQOqpbdyguvS2ZCG	89.47	86.84	89.47
TestCase4: Nia and Mia use the same perfume	90.62	90.62	93.75
TestCase5: The Taj Mahal was built by the Emperor Shah Jahan	95.91	93.87	95.91
TestCase6: Failure is a stepping stone to success	100	92.1	100
TestCase7: AV affa LTaa sertLV	78.94	73.68	78.94
TestCase8: Reason is Orthogonal inseparability	91.17	88.57	91.17
TestCase9: My name is Nitisha Govekar	96.15	100	100
TestCase10: My favorite cartoon show is Oggy and the Cockroaches	92.30	90.38	92.30
Total Recognition rate	93.24	90.97	94.15

We observe that some symbol sequences are orthogonally inseparable. By orthogonally inseparable, we mean that there is no vertical line that passes between the two symbols without crossing bitmap areas of either. Such images could not be processed for individual symbols within the limits of this method since it requires complex image processing. Some cases are such as:

AV - upper case A followed by some uppercase
ffa orfu or fa - lower case f followed by some short characters.
LT or LV – upper case L followed by characters that have side extensions
aj – lower case a followed by characters that have their bottom extensions
ty or tg – lower case t followed by characters that have their bottom extensions

V. CONCLUSIONS

In this paper, we have tried to improve the accuracy for recognition of characters by ensemble method. We trained MLP and SVM classifiers by varying the learning parameters used in each of the techniques and tested both of them separately. We analyzed the outputs of 10 different test cases of each type of font used to observe the behavior of each of the base classifier and used the results of analysis to generate the rules used by the ensemble system to vote its

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

decision on. We conclude that in case of the Latin Arial font and the Latin Times New Roman font, the MLP output is better than SVM output and the ensemble output tries to combine both and give a more accurate output. While in case of the Latin Tahoma font, the SVM output is better than MLP output and the ensemble output gives a more accurate output then their two base counterparts.

We conclude that the ensemble output in all the three types of fonts considered is more accurate than the base classifier outputs.

VI. FUTURE WORK

This method can be further improved by using a more complex image analysis algorithm to recognize the combination of characters accurately which were not recognized properly. Also one could automate the process of generating rules which are actually hard-coded in this method. This method can be used to train and test more complex type of fonts. Also one can use this system procedure to train and test handwritten characters.

REFERENCES

1. R. Polikar, "Ensemble based systems in decision making",Circuits and Systems Magazine, IEEE, Vol. 6,(3), pp. 21–45, 2006.
2. S. K. Shrivastava and S. S. Gharde, "Support Vector Machine for Handwritten Devanagari Numeral Recognition", Int. Journal of Computer Applications Vol.7, Issue11, pp.0975 – 8887 October 2010.
3. A. Rahman and B. Verma, "Ensemble Classifier Composition: Impact on Feature Based Offline Cursive Character Recognition", International Joint Conference on Neural Networks,San Jose, California, USA, July 31-Aug 5,2011.
4. W. W. Azevedo and C. Zanchettin, "A MLP-SVM Hybrid Model for Cursive Handwriting Recognition" , International Joint Conference on Neural Networks,San Jose, California, USA, July 31-Aug 5,2011.
5. W. W. Azevedo, C. Zanchettin and B. L. D. Bezerra, "A KNN-SVM Hybrid Model for Cursive Handwriting Recognition", WCCI 2012 IEEE World Congress on Computational Intelligence, Brisbane, Australia, June, 10-15, 2012.
6. H. W. Hao, C.-X. Mu, X.-C. Yin*, and Z.-B. Wang, "Handwritten Chinese Character Identification with Bagged One-Class Support Vector Machines", International Joint Conference on Neural Networks,San Jose, California, USA, July 31-Aug 5,2011.
7. F. Mohammad, J. Anarase, M. Shingote, and P. Ghanwat, "Optical Character Recognition Implementation Using Pattern Matching", International Journal of Computer Science and Information Technologies, Vol. 5 Issue 2, pp. 2088-2090, 2014.
8. Hirwani, S. Gonnade, "Character Recognition Using Multilayer Perceptron", International Journal of Computer Science and Information Technologies, Vol. 5 Issue 1, pp.558-661, 2014.
9. M. I. Jubairand P. Banik, "A Simplified Method for Handwritten Character Recognition from Document Image", International Journal of Computer Science and Information Technologies, Vol. 51 Issue 14, pp. 0975-8887, 2012.
10. B.E.Kessab, C. Daoui, R. Salouan and B. Boukhalene,"A Comparative Study between the K-Nearest Neighbors and the Multi-Layer Perceptron for Cursive Handwritten Arabic Numerals Recognition", International Journal of Computer Applications Vol. 107 Issue 21, pp. 0975- 8887, 2014.
11. B. El Kessab, C. Daoui, B. Bouikhalene and R. Salouan"A Comparative Study between the Support Vectors Machines and the K-Nearest Neighbors in the Handwritten Latin Numerals Recognition", International Journal of Signal Processing, Image Processing and Pattern Recognition Vol. 8 Issue 2, pp. 325-336, 2015.
12. P. Kumar, N. Sharma and A. Rana," Handwritten Character Recognition using Different Kernel based SVM Classifier and MLP Neural Network (A COMPARISON)", International Journal of Computer Applications Vol. 53 Issue 11, pp. 0975- 8887, 2012.
13. R. M. O. Cruz, G. D. C. Cavalcanti and Tsang IngRen"An Ensemble Classifier For Offline Cursive Character Recognition Using Multiple Feature Extraction Techniques",IEEE Transactions, 2010.
14. D. Nasien, H. Haron and S. S. Yuhaniz "Support Vector Machine (SVM) For English Handwritten Character Recognition", Second International Conference on Computer Engineering and Applications, IEEE Transactions, 2010.
15. N. Singh and S. Sachan "A Survey on Handwritten Devnagari Character Recognition Methods", International Journal of Advanced Research in Computer Science and Software Engineering [Online]. Vol.4, pp. 560-565, 2014, June. Available: http://www.ijarcsse.com
16. S. Mohanty and H. N. D. Bebartta, "Performance Comparison of SVM and K-NN for Oriya Character Recognition," (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Image Processing and Analysis.
17. J. Kim, B.-S. Kim and S. SAVARESE, "Comparing Image Classification Methods: K-Nearest-Neighbor and Support-Vector-Machines" unpublished.
18. S. Arora. D. Bhattacharjee, M. Nasipuri, L. Malik , M. Kundu and D. K. Basu, "Performance Comparison of SVM and ANN for Handwritten Devnagari Character Recognition" unpublished.
19. Z. Shan and H. Cao "Ensemble Optical Character Recognition Systems via Machine Learning" unpublished.
20. C.-L. Liu and H. Fujisawa, "Classification and Learning for Character Recognition: Comparison of Methods and Remaining Problems" unpublished
21. https://www.quora.com/What is the intuition behind Guassian kernel in SVM.
22. http://www.cse.unsw.edu.au/~cs9417ml/MLP2/BackPropagation.html.