Handwritten Digit Recognition Using Machine Learning Algorithms

S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana and Abdullah Al Jobair

Department of Information and Communication Technology
Mawlana Bhashani Science and Technology University
Santosh, Tangail-1902

Abstract

Handwritten character recognition is one of the practically important issues in pattern recognition applications. The applications of digit recognition includes in postal mail sorting, bank check processing, form data entry, etc. The heart of the problem lies within the ability to develop an efficient algorithm that can recognize hand written digits and which is submitted by users by the way of a scanner, tablet, and other digital devices. This paper presents an approach to off-line handwritten digit recognition based on different machine learning technique. The main objective of this paper is to ensure effective and reliable approaches for recognition of handwritten digits. Several machines learning algorithm namely, Multilayer Perceptron, Support Vector Machine, Naïve Bayes, Bayes Net, Random Forest, J48 and Random Tree has been used for the recognition of digits using WEKA. The result of this paper shows that highest 90.37% accuracy has been obtained for Multilayer Perceptron.

Keywords:

Pattern recognition, Handwritten recognition, Digit recognition, Machine learning, WEKA, Off-line handwritten recognition, Machine learning algorithm, Neural Network, Classification algorithm.

1. Introduction

Intelligent image analysis is an appealing research area in Artificial Intelligence and also crucial for a variety of present open research difficulties. Handwritten digits recognition is a well-researched subarea within the field that is concerned with learning models to distinguish pre-segmented handwritten digits. It is one of the most important issues in data mining, machine learning, pattern recognition along with many other disciplines of artificial intelligence [1].The main application of machine learning methods over the last decade has determined efficacious in conforming decisive systems which are competing to human performance and which accomplish far improved than manually written classical artificial intelligence systems used in the beginnings of optical character recognition technology [2]. However, not all features of those specific models have been previously inspected.

A great attempt of research worker in machine learning and data mining has been contrived to achieve efficient approaches for approximation of recognition from data [3]. In twenty first Century handwritten digit communication has its own standard and most of the times in daily life are being used as means of conversation and recording the information to be shared with individuals. One of the challenges in handwritten characters recognition wholly lies in the variation and distortion of handwritten character set because distinct community may use diverse style of handwriting, and control to draw the similar pattern of the characters of their recognized script.

Identification of digit from where best discriminating features can be extracted is one of the major tasks in the area of digit recognition system. To locate such regions different kind of region sampling techniques are used in pattern recognition [4].The challenge in handwritten character recognition is mainly caused by the large variation of individual writing styles [5]. Hence, robust feature extraction is very important to improve the performance of a handwritten character recognition system. Nowadays handwritten digit recognition has obtained lot of concentration in the area of pattern recognition system

Manuscript received September 5, 2025
Manuscript revised September 20, 2025
https://doi.org/10.22937/IJCSNS.2025.25.9.12

sowing to its application in diverse fields. In next days, character recognition system might serve as a cornerstone to initiate paperless surroundings by digitizing and processing existing paper documents.

Handwritten digit dataset are vague in nature because there may not always be sharp and perfectly straight lines. The main goal in digit recognition is feature extraction is to remove the redundancy from the data and gain a more effective embodiment of the word image through a set of numerical attributes. It deals with extracting most of the essential information from image raw data [6]. In addition the curves are not necessarily smooth like the printed characters. Furthermore, characters dataset can be drawn in different sizes and the orientation which are always supposed to be written on a guideline in an upright or downright point. Accordingly, an efficient handwritten recognition system can be developed by considering these limitations. It is quiet exhausting that sometimes to identify hand written characters as it can be seen that most of the human beings can't even recognize their own written scripts. Hence, there exists constraint for a writer to write apparently for recognition of handwritten documents. Before revealing the method used in conducting this research, software engineering module is first presented. Pattern recognition along with Image processing plays compelling role in the area of handwritten character recognition. The study [7], describes numerous types of classification of feature extraction techniques like structural feature based methods, statistical feature based methods and global transformation techniques. Statistical approaches are established on planning of how data are selected. It utilizes the information of the statistical distribution of pixels in the image. The paper [8], provided SVM based offline handwritten digit recognition system. Authors claim that SVM outperforms in the experiment. Experiment is carried out on NIST SD19 standard dataset. The study [9] provide the conversion of handwritten data into electronic data, nature of handwritten characters and the neural network approach to form machine competent of recognizing hand written characters. The study [10] addresses a comprehensive criterion of handwritten digit recognition with various state of the art approaches, feature representations, and datasets. However, the relationship of training set size versus accuracy/error and the dataset-independence of the trained models are analyzed. The paper [11] presents convolution neural networks into the handwritten digit recognition research and describes a system which can still be considered state of the art.

2. Methods and Materials

2.1. Multilayer Perceptions

A neural network based classifier, called Multi-Layer perception (MLP), is used to classify the handwritten digits. Multilayer perceptron consists of three different layers, input layer, hidden layer and output layer. Each of the layers can have certain number of nodes also called neurons and each node in a layer is connected to all other nodes to the next layer [12]. For this reason it is also known as feed forward network. The number of nodes in the input layer depends upon the number of attributes present in the dataset. The number of nodes in the output layer relies on the number of apparent classes exist in the dataset. The convenient number of hidden layers or the convenient number of nodes in a hidden layer for a specific problem is hard to determine. But in general, these numbers are selected experimentally. In multilayer perceptron, the connection between two nodes consists of a weight. During training process, it basically learns the accurate weight adjustment which is corresponds to each connection [13]. For the learning purpose, it uses a supervised learning technique named as Back propagation algorithm.

2.2. Support Vector Machine

SVM or Support Vector Machine is a specific type of supervised ML method that intents to classify the data points by maximizing the margin among classes in a high-dimensional space [14]. SVM is a representation of examples as points in space, mapped due to the examples of the separate classes are divided by a fair gap that is as extensive as possible. After that new examples are mapped into that same space and anticipated to reside to a category based on which side of the gap they fall on [15]. The optimum algorithm is developed through a "training" phase in which training data are adopted to develop an algorithm capable to discriminate between groups earlier defined by the operator (e.g. patients vs. controls), and the "testing" phase in which the algorithm is adopted to blind-predict the group to which a new perception belongs [16]. It also provides a very accurate classification performance over the training records and produces enough search space for the accurate classification of future data parameters. Hence it always ensures a series of parameter combinations no less than on a sensible subset of the data. In SVM it's better to scale the data always; because it will extremely improve the results. Therefore be cautious with big dataset, as it may leads to the increase in the training time.

2.3. J48

The J48 algorithm is developed for the MONK project along with WEKA [17]. The algorithm is an extension for C4.5 decision tree algorithm [18]. There are many options for tree pruning in case of J48 algorithm. The classification algorithms convenient in WEKA try to clarify the results or prune. This method will help us to produces more generic results and also can be used to correct potential over fitting issues. J48 helps to recursively classify until each of the leaf is pruned, that is to classify as close knit to the data. Hence this will helps to ensure the accuracy, although excessive rules will be produced. However pruning will cause to less accuracy of a model on training data. This is due to pruning employs various means to relax the specificity of the decision tree, hopefully improving its performance on the test data. The complete concept is to increasingly generalize a decision tree until it gains a balance of accuracy together with flexibility. The J48 applies two pruning methods. First one is known as subtree replacement. This concludes that nodes in the decision tree can be replaced with a leaf -- which reduces the number of tests along a particular path. This process begins from the leaves of the completely formed tree, and attempts backwards toward the root. Second category of pruning adopted in J48 is termed subtree rising. In this respect, a node can be moved upwards towards the root of the tree, replacing other nodes another way. Subtree rising repeatedly has a insignificant effect on decision tree models. There is generally no clear way to anticipate the utility of the option, though it may be desirable to try turning it off if the induction process is catching a long time. This is because of the fact that subtree rising may be somewhat computationally complicated. Error rates are needed to make actual conclusions about which parts of the tree to rise or replace. There exist multiple ways to perform this. The straight forward way is to reserve a portion of the training data in order to test on decision tree. Reserved portion may then be adopted as test data for the decision tree, aiding to reduce potential over fitting. This method is recognized as reduced-error pruning. Though the approach is straightforward, it also decreases the overall volume of data available for training the model. For specifically small datasets, it may be advisable to avert using reduced error pruning.

2.4. Random Forest Algorithm

Random forest as is an ensemble of un-pruned regression or classification trees, activated from bootstrap samples of the training data, adopting random feature selection in the tree imitation process. The prediction is made by accumulating the predictions of the ensemble by superiority voting for classification. It returns generalization error rate and is more potent to noise. Still, similar to most classifiers, RF may also suffer from the curse of learning from an intensely imbalanced training data set. Since it is constructed to mitigate the overall error rate, it will tend to focus more on the prediction efficiency of the majority class, which repeatedly results in poor accuracy for the minority class.

2.5. Naive Bayes

The Naive Bayes classifier [19] contributes a simple method, representing and learning probabilistic knowledge with clear semantics. It is termed naive due to it relies on two important simplifying assumes that predictive attributes are conditionally self-reliant given the class, and it considers that no hidden attributes influence the prediction method. It is a probabilistic classifier which relies upon Bayes theorem with robust and naive independence assumptions. It is one of the best basic text classification approaches with numerous applications in personal email sorting, email spam detection, sexually explicit content detection, document categorization, sentiment detection, language detection [20]. Although the naïve design and oversimplified assumptions that this approach uses, Naive Bayes accomplishes well in many complicated real-world problems. All though it is often out performed by other approaches such as boosted trees, Max Entropy, Support Vector Machines, random forests etc, Naive Bayes classifier is very potent as it is less computationally intensive (in both memory and CPU) and it needs a small extent of training data. Moreover, the training time with Naive Bayes is considerably smaller as opposed to alternative approaches.

2.6. Bayes Net

Bayesian networks are a powerful probabilistic representation, and their use for classification has received considerable attention [21]. It reflects the states of some part of a world that is being modeled and it describes how those states are related by probabilities. Bayesian network is a graphical model that encodes probabilistic relationships among variables of interest. When used in conjunction with statistical techniques, the graphical model has several advantages for data analysis. One, because the model encodes dependencies among all variables, it readily handles situations where some data entries are missing. Two, a Bayesian network can be used to learn causal relationships, and hence can be used to gain understanding about a problem domain and to predict the consequences of intervention. This classifier learns from training data the conditional probability of each attribute given the class label [22, 23].

2.7. Random Tree

The algorithm may deal with both regression and classification problems. Random trees is a ensembleof tree predictors which is called forest .The classification performs as follows: random trees classifier takes the input feature vector, categories it with individual tree in the forest, outputs the class label which received the most of “votes”. In the event of a regression, the classifier response is the average of the responses over all the trees in the forest [24]. In random tree algorithm all the trees are trained with the same parameters but on different training sets. These sets are created from the original training set adopting the bootstrap procedure and for each training set, randomly choose the same number of vectors as in the

initial set. The vectors are chosen with replacement. That is, some vectors will occur more than once and some will be absent. In random trees there is no need for any accuracy estimation techniques, like cross-validation or bootstrap, or a separate test set to obtain an estimate of the training error. The error is estimated internally during the training.

2.8. Dataset Description

The handwritten digit recognition is a extensive research topic which gives a comprehensive survey of the area including major feature sets, learning datasets, and algorithms [25]. Contrary to optical character recognition which focuses on recognition of machine-printed output, where special fonts can be used and the variability between characters along with the same size, font, and font attributes is fairly small.

The feature extraction and the classification technique play an important role in offline character recognition system performance. Various feature extraction approaches have been proposed for character recognition system [26]. The problems faced in handwritten numeral recognition has been studied while using the techniques like Dynamic programming, HMM, neural network, Knowledge system and combinations of above techniques [27]. Wider ranging work has been carried out for digit recognition in so many languages like English, Chinese, Japanese, and Arabic. In Indian mainly worked in Devanagari, Tamil, Telugu and Bengali numeral recognition [28].

In our experiment we used digit dataset provided by Austrian Research Institute for Artificial Intelligence, Austria. This data set indicate that arbitrary scaling and a blur setting of 2.5 for the Mitchell down-sampling filter should perform well and used down-sample to 16x16 pixels.

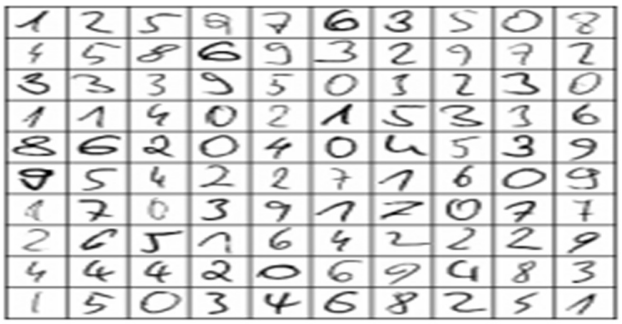

Figure-1: A small portion of handwritten dataset example

This dataset is divided in two parts training set and testing set. Training set has 1893 samples and test set has 1796 samples. The detail of the dataset is provided in [29].

3. Experimental Tools

Waikato Environment for Knowledge Analysis (WEKA) is a prominent suite of machine learning which is written in Java and developed at the University of Waikato. It is free software accessible under the GNU General Public License. It contains a collection of algorithms and visualization tools for predictive modelling, data analysis, along with graphical user interfaces for smooth access to this functionality [30]. All of Weka's approaches are predicated on the assumption that the data is convenient as a single flat file or relation, where each data point is characterized through a fixed number of attributes [31].

WEKA has numerous user interfaces. Its main user interface is the Explorer, however essentially the same functionality can be accessed by the component-based Knowledge Flow interface and from the command line. The Experimenter allows the systematic comparison of the predictive performance of the Weka's machine learning algorithms on an accumulation of datasets.

4. Experimental Result and Discussion

WEKA has several graphical user interfaces that enable easy access to the underlying functionality. To gauge and investigate the performance on the selected methods or algorithms namely Support Vector Machine, Multilayer Perceptron, Random Forest Algorithm, Random Tree, Naïve Bayes, Bayes Net and j48 Decision tree algorithms are used. We use the same experiment procedure as suggested by WEKA.

In WEKA, all dataset is considered as instances and features in the data are also known as attributes. The experiment results are partitioned into several sub division for easier analysis and evaluation. In the first part, correctly and incorrectly classified instances will be divided in numeric and percentage value and subsequently Kappa statistic, mean absolute error and root mean squared error will be in numeric value . Experiment shows the relative absolute error and root relative squared error in percentage (%) for references and in evaluation process. Our simulation results are shown in below tables-1 and tables-2. In table-1 mainly summarizes the result based on accuracy and time taken for each simulation in our experiment. Moreover, table-2 shows the result based on error during the simulation in WEKA.

Table-1: Simulation result based on accuracy and time consumption

Name of Algorithms	Correctly Classified Instances % (value)	Incorrectly Classified Instances % (Value)	Time Taken (seconds)	Kappa Statistic
Multilayer Perceptron	90.37	9.63	3.15	0.893
Support Vector Machine	87.97	12.03	0.56	0.8664
Random Forest	85.75	14.25	0.44	0.8416
Bayes Net	84.35	15.65	0.86	0.8262
Naïve Bayes	81.85	18.15	3.45	0.7983
J48	79.51	20.49	0.53	0.7722
Random Tree	85.6	24.94	0.55	0.7228

Table-2: Simulation result based on different error

Name of Algorithms	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error (%)	Root Relative Squared Error (%)
Multilayer Perceptron	0.023	0.1231	12.78	41.04
Support Vector Machine	0.1611	0.2734	89.49	91.15
Random Forest	0.0593	0.1532	32.97	51.06
Bayes Net	0.0312	0.1745	17.36	58.15
Naïve Bayes	0.0361	0.1879	20.06	62.65
J48	0.0444	0.1957	24.66	65.25
Random Tree	0.0499	0.2234	27.72	74.45

Based on the above table-1, the highest accuracy is 90.37 % and the lowest is 75.06%. The other algorithm yields an average accuracy of around 83.89%. In fact, the highest accuracy belongs to the Multilayer Perceptron classifier, followed by Support Vector Machine with a percentage of 87.97% and subsequently Random Forest Algorithm 85.75%, Bayes Net 84.35%, Naïve Bayes 81.85%, j48 79.51% and Random Tree 75.06%. Kappa statistics value ranges from 0 to 1. Value 0 means totally disagreement and 1 means full agreement. It checks the reliability of Classifying algorithm on dataset. The total time, mean absolute error, root mean absolute error, relative absolute error and root relative absolute error is also a crucial parameter to build the model in comparing the different classification algorithm. Mean absolute error is the mean of overall error made by classification algorithm and least the error will be best classifier. In the table-2 Multilayer Perceptron has least 0.023 mean absolute errors among all seven algorithms.

In [32] experimental results reveal that it is possible to train a face detector without having to label images as containing a face or not. Their experiment is only sensitive to high-level concepts such as cat faces and human bodies. Multi-column deep neural networks for image classification have been presented in [33]. They only improve the state-of-the-art on a plethora of common image classification benchmarks. Supervised learning unsupervised learning, reinforcement learning & evolutionary computation, and indirect search for short programs encoding deep and large networks has been presented in [34]. They only proposed how different technique can be used for pattern recognition. In [35] recognition of handwritten bangla basic characters and digits using convex hull based feature set has been proposed. Their experiment result shows that with a database of 10000 samples, the maximum recognition rate of 76.86% is observed for handwritten Bangla characters. Online and offline handwritten Chinese character recognition has been proposed in [36]. Their experiment result reported that the highest test accuracies 89.55% for offline. In our experiment different machine learning algorithm has been used for handwrite digit recognition and obtained highest 90.37% accuracy obtained for Multilayer Perceptron.

5. Conclusion

The main objective of this investigation is to find a representation of isolated handwritten digits that allow their effective recognition. In this paper used different machine learning algorithm for recognition of handwritten numerals. In any recognition process, the important problem is to address the feature extraction and correct classification approaches. The proposed algorithm tries to address both the factors and well in terms of accuracy and time complexity. The overall highest accuracy 90.37% is achieved in the recognition process by Multilayer Perceptron. This work is carried out as an initial attempt, and the aim of the paper is to facilitate for recognition of handwritten numeral without using any standard classification techniques.

References

[1] Watada, J., & Pedrycz, W. A fuzzy regression approach to acquisition of linguistic rules. Handbook of Granular Computing, 719-732, 2008.

[2] Seewald, A. K. (2011). On the brittleness of handwritten digit recognition models. ISRN Machine Vision, 2012.

[3] Kloesgen, W., & Zytkow, J. Handbook of Knowledge Discovery and Data Mining.

[4] Das, N., Sarkar, R., Basu, S., Kundu, M., Nasipuri, M., & Basu, D. K. A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application. Applied Soft Computing, 12(5), 1592-1606, 2012.

[5] Plamondon, R., & Srihari, S. N. Online and off-line handwriting recognition: a comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), 63-84.

[6] AlKhateeb, J. H., Pauplin, O., Ren, J., & Jiang, J.. Performance of hidden Markov model and dynamic Bayesian network classifiers on handwritten Arabic word recognition. knowledge-based systems, 24(5), 680-688, 2011.

[7] Tokas, R., & Bhadu, A. A comparative analysis of feature extraction techniques for handwritten character recognition. International Journal of Advanced Technology & Engineering Research, 2(4), 215-219, 2012

[8] Neves, R. F., Alberto Filho, N. G., Mello, C. A., & Zanchettin, C. A SVM based off-line handwritten digit recognizer. In System, IEEE International Conference on Man, and Cybernetics (SMC) (pp. 510-515), 2011

[9] Perwej, Y., & Chaturvedi, A. Machine recognition of hand written characters using neural networks. arXiv preprint arXiv:1205.3964, 2012.

[10] Liu, C. L., Nakashima, K., Sako, H., & Fujisawa, H. Handwritten digit recognition: benchmarking of state-of-the-art techniques. Pattern Recognition,36(10), 2271-2285.

[11] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[12] Bhowmik, S., Roushan, M. G., Sarkar, R., Nasipuri, M., Polley, S., & Malakar, S. Handwritten Bangla Word Recognition Using HOG Descriptor. In 2014 Fourth International Conference of Emerging Applications of Information Technology (EAIT), (pp. 193-197). IEEE, 2014.

[13] Kruse, R., Borgelt, C., Klawonn, F., Moewes, C., Steinbrecher, M., & Held, P. Multi-Layer Perceptrons. In Computational Intelligence (pp. 47-81). Springer London, 2013.

[14] Pereira, F., Mitchell, T., & Botvinick, M. Machine learning classifiers and fMRI: a tutorial overview. Neuroimage, 45(1), S199-S209, 2009.

[15] Support Vector Machine, Online: https://en.wikipedia.org/wiki/Support_vector_machine, Accessed, August 2015.

[16] Orrù, G., Pettersson-Yeo, W., Marquand, A. F., Sartori, G., & Mechelli, A. Using support vector machine to identify imaging biomarkers of neurological and psychiatric disease: a critical review. Neuroscience & Biobehavioral Reviews, 36(4), 1140-1152, 2012.

[17] Cooper, G. F., & Herskovits, E. A Bayesian method for constructing Bayesian belief networks from databases. *In Proceedings of the Seventh conference on Uncertainty in Artificial* Intelligence (pp. 86-94). Morgan Kaufmann Publishers Inc..

[18] Salzberg, S. L. C4. 5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning,* 16(3), 235-240.

[19] John, G. H., & Langley, P. Estimating continuous distributions in Bayesian classifiers. In Proceedings of the Eleventh conference on Uncertainty in artificial intelligence (pp. 338-345). *Morgan Kaufmann Publishers Inc.*.

[20] Nguyen, H. *A., & Choi, D. Application of data mining to network intrusion detection: classifier selection model. In Challenges for Next Generation Network Operations and Service Management (pp. 399-408). Springer Berlin Heidelberg, 2008.*