ISSN 2277-3061

EFFECT OF NON-IMAGE FEATURES ON RECOGNITION OF HANDWRITTEN ALPHA-NUMERIC CHARACTERS

Ibrahim A. Adeyanju, Olusayo D. Fenwa, Elijah O. Omidiora

Department of Computer Science and Engineering,

Ladoke Akintola University of Technology, Ogbomoso, Oyo state, NIGERIA

{iaadeyanju | odfenwa | eoomidiora}@lautech.edu.ng

ABSTRACT

Handwritten character recognition has applications in several industries such as Banking for reading of cheques and Libraries/ National archives for digital searchable storage of historic texts. The main feature typically used for the recognition task is the character image. However, there are other possible features such as the hand (left or right) used by author, number of strokes and other geometric features that can be captured when writing on digital devices. This paper investigates the effect of using some non-image features on the recognition rate of three classifiers: Instance Based Learner (IBk), Support Vector Machines (SVM) and the Multilayer Perceptron (MLP) Neural Network for singly-written alpha-numeric character recognition. Our experiments were conducted using the WEKA machine learning tool on offline and online handwritten acquired locally. A percentage split (66%-34% train-test) evaluation methodology was adopted with the classification accuracy measured. Results indicate that non-image additional features improved the accuracy across the three classifiers for the online and offline character datasets. However, this improvement was not statistically significant. SVM gave the best accuracy for the online dataset while IBk performed better than the other two classifiers for the offline dataset. We intend to investigate the effect of non-image features at other levels of text granularity such as words and sentences.

Indexing terms/Keywords

Handwritten character recognition; support vector machines; multilayer perceptron neural network; instance based learning; nearest neghbour algorithm

Academic Discipline And Sub-Disciplines

Computer Science, Computer Engineering, Information Technology, Information Systems

SUBJECT CLASSIFICATION

Pattern Recognition, Artificial Intelligence, Machine Learning

TYPE (METHOD/APPROACH)

Experimentation, Empirical studies, Computer Simulation

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATION JOURNAL OF COMPUTERS AND TECHNOLOGY

Vol. 13, No. 11

editorijctonline@gmail.com

www.ijctonline.com, www.cirworld.com

1. INTRODUCTION

Character recognition is the conversion of textual character images into computer readable fomat. Such character images can be classified as handwritten, typewritten or machine printed depending on the source. Handwritten characters can either be online or offline with respect to the means of data acquisition. Offline data is typically written using pen and paper before being scanned into electronic form as images. Online data are captured directly using modern devices such as a digitizer and an electric pen. A major advantage of online devices is that they capture the temporal or dynamic information about the writing such as the number of strokes, the order of the strokes, the direction of the writing for each stroke, and the speed of the writing within each stroke [16].

This work investigates the effect of non-image features on the recognition rate of three classifiers for alpha-numeric character recognition. Section 2 discusses related work while Section 3 provides theoretical background to the classifiers. Experimental setup is stated in Section 4 with a discussion of results inSection 5. Section 6 concludes with a summary of the work presented and our plans for future work.

2. RELATED WORK

Humans have been writing important information on paper (and other primitive materials such as woods, clay tablets etc.) even before the advent of computer systems. However, storing such information has typically been very cumbersome and expensive. The use of these handwritten data on paper is generally limited to those around where it is stored though several others across the globe might benefit from it. Digitising such information and converting them to recognisable text allows search engines such as Google, Yahoo and MSN to be able to index and make them available for user around the world once they are uploaded to the web.

Selection of a feature extraction method is undoubtedly the single most important factor in achieving high recognition performance in character recognition system [13]. No matter how sophisticated the classifiers and learning algorithms, poor feature extraction will always lead to poor system performance [2][7][10][15]. The main feature of any handwritten character image is the pixel values. However, online data acquisition of character images with electronic devices such as a pen digitizer allows extraction of other features such as number of stroke, pressure of the pen and contour pixels. This work attempts to quantify the importance of some non-image feature for character recognition.

3. CHARACTER RECOGNITION CLASSIFIERS

This section discusses the details of the three classifiers used for character recognition in this paper.

3.1 Instance Based Learner (IBk)

Instance-based learning, also known as memory-based learning, is a family of learning algorithms that, instead of performing explicit generalization, compare new problem instances with instances seen in training, which have been stored in memory. Instance-based learning is a kind of lazy learning because it constructs hypotheses directly from the training instances themselves [1][14]. This means that the hypothesis complexity can grow with the data. One advantage that instance-based learning has over other methods of machine learning is its ability to adapt its model to previously unseen data. Where other methods generally require the entire set of training data to be re-examined when one instance is changed, instance-based learners may simply store a new instance or throw away old instances.

A simple example of an instance-based learning algorithm is the k-nearest neighbour algorithm [9]. The k-nearest neighbour algorithm (k-NN) is a non-parametric method for classification and regression that predicts objects' "values" or class memberships based on the k closest training examples in the feature space. k-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The k-nearest neighbour algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbours, with the object being assigned to the class most common amongst its k nearest neighbours (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbour. The same method can be used for regression, by simply assigning the property value for the object to be the average of the values of its k nearest neighbours. It can be useful to weight the contributions of the neighbours, so that the nearer neighbours contribute more to the average than the more distant ones. A common weighting scheme is to give each neighbour a weight of 1/d, where d is the distance to the neighbour. The neighbours are taken from a set of objects for which the correct classification (or, in the case of regression, the value of the property) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. The k-nearest neighbour (k-NN) algorithm is sensitive to the local structure of the data.

The training examples in k-NN are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the classification phase, k is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point. A commonly used distance metric for continuous variables is Euclidean distance. The Euclidean distance between two vectors is as defined below, where *a* and *a'* are vectors of same length.

$$d(a,a') = \sqrt{(a_1 - a_1')^2 + ... + (a_n - a_n')^2}$$

The accuracy of the k-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the feature scales are not consistent with their importance. Much research effort has been put into selecting or scaling features to improve classification. The k-NN instance based learner is implemented as IBk in Weka tool [4].

3.2 Support Vector Machines (SVM)

SVM is a binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories; SVM training algorithm builds a model that assigns new example into one category or the other by constructing a hyperplane or set of hyperplanes in a high- or infinite-dimensional space. SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible [5]. A good separation is achieved by the hyperplane that has the largest distance to the nearest training data point of any class (functional margin, δ) as shown in Figure 1. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

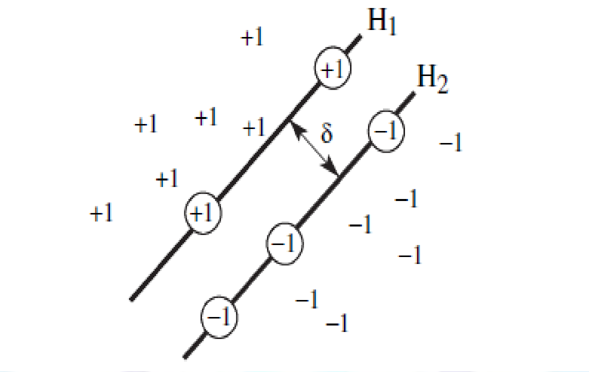

Figure 1: Maximum separation hyperplane

For any particular set of two-class objects, SVM finds the unique hyperplane having the maximum margin (Figure 1). The hyperplane H_1 defines the border with class +1 objects, whereas the hyperplane H_2 defines the border with class -1 objects. Two objects from class +1 define the hyperplane H_1, and three objects from class -1 define the hyperplane H_2. These objects, represented inside circles in Figure 2, are called support vectors. A special characteristic of SVM is that the solution to a classification problem is represented by the support vectors that determine the maximum margin hyperplane.

SVM can also be used to separate classes that cannot be separated with a linear classifier knowing well that character recognition vectors are non-linear in nature [12]. In such cases, the coordinates of the objects are mapped from the input space into a feature space using nonlinear functions called feature functions *Ø(x)* e.g. two-dimension to three-dimension. The feature space (Figure 2) is a high-dimensional space in which the two classes can be separated with a linear classifier (kernel trick). The nonlinear feature functions *Ø(x)* used are called Kernels $K(x_i,x_j)$.

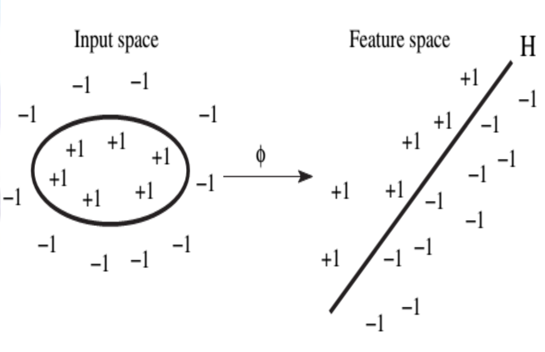

Figure 2: Linear separation in feature space

Kernels have the advantage of operating in the input space, where the solution of the classification problem is a weighted sum of kernel functions evaluated at the support vectors. Commonly used kernels are Linear or Dot kernel, Polynomial, Radial Basis Function, Gaussian Radial basis function and Sigmoid kernel. The question of “which SVM kernel gives the best recognition accuracy amidst numerous kernels?” can only be solved by experimenting the various available kernels to ascertain the one which has the best recognition accuracy for a given problem [12]. Hence the use of linear kernel, sigmoid kernel and radial basis function kernel of the SVM kernels in this paper. Character recognition classifications usually need more than two classes for classification (multiclass). Two common methods to build such binary classifiers are those where each classifier is trained to distinguish: (i) one of the labels against to all the rest of labels (known as one-versus-all) or (ii) every pair of classes (known as one-versus-one). Classification of new instances for one-versus-all case is done by a winner-takes-all strategy, in which the classifier with the highest output function assigns the class [17]. The classification of one-versus-one case is done by a max-wins voting strategy, in which every classifier assigns the instance to one of the two classes, then the vote for the assigned class is increased by one vote, and finally, the class with more votes determines the instance classification.

3.3 MultiLayer Perceptron (MLP)

MLP is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs [8]. It consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. Figure 3 shows the architecture for a typical MLP network. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. If a multilayer perceptron has a linear activation function in all neurons, that is, a simple on/off mechanism to determine whether or not a neuron fires, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two-layer input-output model [11]. What makes a multilayer perceptron different is that each neuron uses a nonlinear activation function which was developed to model the frequency of action potentials, or firing, of biological neurons in the brain. This function is modelled in several ways, but must always be normalizable and differentiable.

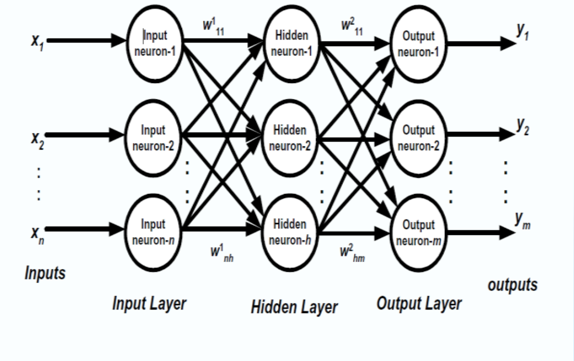

Figure 3: A typical Multilayer Perceptron

Although the variety of proposed neural network structures has grown, the multilayer perceptron remains the prevailing one and also the most widespread network structure. The inherent capability of the three-layer network structure to carry out any arbitrary input-output mapping qualifies the multilayer perceptron networks for recognition of handwritten digits. When trained on examples of observation data, the networks can learn the characteristic features “hidden” in the examples of the collected data and even generalize the knowledge learnt. The multilayer perceptron, because of its cascaded structure, performs the input-output mapping of nonlinearities [6]. To train a neural network to perform some task, we must adjust the weights of each unit in such a way that the error between the desired output and the actual output is reduced. This process requires that the neural network compute the error derivative of the weights (EW). In other words, it must calculate how the error changes as each weight is increased or decreased slightly. The back-propagation algorithm is the most widely used method for determining EW.

Neural networks have since the very beginning of their practical application proved to be a powerful tool for signal analysis, features extraction, data classification, pattern recognition, *etc*. Owing to their capabilities of learning and generalization from observed data, the networks have been widely accepted by engineers and researchers as a tool for processing of experimental data [3]. This is mainly because neural networks reduce enormously the computational efforts needed for problem solving. Generally speaking, the practical use of neural networks has been recognized mainly because of such distinguished features as: general nonlinear mapping between a subset of the past time series values and the future time series values [6][8]. The capability of capturing essential functional relationships among the data, which is valuable when such relationships are not *a priori* known or are very difficult to describe mathematically and/or when the collected observation data are corrupted by noise universal function approximation capability that enables modeling of arbitrary nonlinear continuous functions to any degree of accuracy.

4. EXPERIMENTAL SETUP

Experiments were designed to determine the effect of non-image features on the accuracy of character recognition systems. The Waikato Environment for Knowledge Analysis (Weka) [4][18] tool (version 3.6.6) was used in experimentation as it has the three classifiers implemented. The experiment was performed on 7,935 offline and 7,985 online character images datasets collected locally from 62 users. A percentage split evaluation methodology was employed with 66% of each data used for training and 34% for testing. These splits were chosen randomly and the process was repeated ten times. The specific configurations used for each classifier were as follows:

1. Instance Based Learner: The IBk algorithm was configured with k=3 using the 1/distance weighting function while the Euclidean distance was used as the similarity metric. Other parameters were left at default.

2. Support Vector Machine: The linear kernel function on WEKA LIBSVM was used with all parameters left at default.

3. Multilayer Perceptron (MLP) Neural Network: The Multilayer Perceptron algorithm was used with the number of hidden layers chosen as the number of possible classes (i.e. a-z,A-Z,0-9; 62 classes in our work) while the maximum iteration was set at 200 epoch. All other parameters were left at the default values.

5. RESULTS AND DISCUSSION

Comparison of percentage accuracy of the three classifiers in character recognition using a 7x7 down sampled sizes of the online and offline character images, test of significance was done at 95% confidence (Two tail). The online dataset had four variations of the original datasets while the offline had two variations to test the effect of different non-image features. An asterisk (*) beside a value in a table shows that it is significantly worse (statistically) than others in the same row. Tables 1 and 2 shows the classification accuracy across the three classifiers for datasets with the non-image features varied. The non-image features used were the hand (right or left denoted as R/L) used by the author either while writing characters online or offline, and the number of strokes which applies only to online writing of characters.

Table 1: Accuracy results for online datasets

Features	IBk	SVM	MLP
Pixels only	70.30	70.09	65.61*
Pixels + R/L Handed	70.27	70.04	66.24*
Pixels + NumStrokes	70.20	71.56	65.89*
Pixels, R/L Handed + NumStrokes	71.13	71.63	65.36*

Table 2: Accuracy results for offline datasets

Features	IBk	SVM	MLP
Pixels only	68.49	67.24*	61.51*
Pixels + R/L Handed	68.54	67.39*	61.38*

Tables 1 & 2 show that the non-images features generally improved the accuracy of the recognition system across all classifiers for both the online and offline datasets. Particularly, a combination of using the right/left hand and number of strokes features gave the best accuracy of 71.63% for the online dataset and was obtained by using SVM. The results obtained from using the IBk and SVM classifiers were significantly better than using MLP for the online dataset. This might be an indication that Neural networks are not good at generalising nominal values since the only numeric feature used in the number of stroke. This can be understood in the light of the fact that the functions used for generalisation in neural networks assumes that all input are numeric and has no specific way of handling nominal values unlike IBk and SVM. There is no significant difference between the results obtained from IBk and SVM though IBk performed slightly better when the number of stroke feature is not used. A different trend is observed when the number of stroke feature is used with SVM outperforming IBk, albeit slightly as well. A reasonable intuition from this trend is that IBk works better with nominal values that numeric values.

Results from the offline dataset indicated that IBk was significantly better than both SVM and MLP though only one non-image feature (right or left handed) was tested. This reinforces our intuition that IBk performs better with nominal values since there was numeric value tested in the offline recognition experiments. Also, the average result values from SVM were better than those from MLP. On the overall, comparison across classifiers shows that MLP gave the worse performance for character recognition while SVM was best for online dataset and IBK for the offline dataset. Expectedly, the accuracy values obtained from online character recognition experiments were better than those from their offline counterparts.

6. CONCLUSION

Three classifiers, Instance Based Learner (IBK), Support Vector Machines (SVM) and MultiLayer Perceptron (MLP) have been used to investigate the effect of non-image features for handwritten character recognition using online and offline datsets. Experimental results indicate that non-image additional features improved the accuracy across the three classifiers for the online and offline character datasets. However, this improvement was not statistically significant. We intend to investigate the effect of non-image features at other levels of text granularity such as words and sentences.

ACKNOWLEDGMENTS

This work was done as part of a research project funded through Ladoke Akintola University of Technology (LAUTECH) Ogbomoso Nigeria Research and Consultancy (LAURESCON) by the University Senate Research Grant (LAU/SRG/13/025) in the 2012/2013 session.

REFERENCES

[1] Aha, D., Kibler, D., & Albert, M. K. 1991. Instance-based learning algorithms, Machine Learning, 6, 37-66

[2] Ahmad, T. 2006. An Efficient Feature Extraction Algorithm for the Recognition of Handwritten Arabic Digits, International Journal of Computational Intelligence, 2 (2), 107-112

[3] Anders, U. and Korn, O. 1999. Model selection in neural networks, Neural Networks, 12: 309–323

[4] Bouckaert, R. R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A., and Scuse, D. 2011. WEKA Manual for Version 3-6-6, The University of Waikato

[5] Christopher, J. and Burges, C. 1998. A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 2, 121-167

[6] Cybenko, G. (1989), Approximation by superpositions of a sigmoidal function, Mathematical Control Signals Systems, 2,303–314

[7] Fenwa, O. D., Omidiora, E. O., & Fakolujo, O. A. 2012. Development of a Feature Extraction Technique for Online Character Recognition System, Journal of Innovative Systems Design and Engineering, 3 (3), 10-23

[8] Fogel, D. B. 1991. An Information Criterion for Optimal Neural Network Selection, IEEE Transactions on Neural Networks, 2, 490–497

[9] Hall, P., Park, B., & Samworth, R. 2008. Choice of neighbor order in nearest-neighbor classification, Annals of Statistics, 36, 2135–2152

[10] Marc, P., Alexandre, L., & Christian, G. 2001. Character Recognition Experiment using Unipen Data, Proceeding of International Conference on Document Analysis and Recognition (ICDAR), 10-13

[11] Omidiora, E. O., Adeyanju, I. A., Fenwa, O. D. 2013. Comparison of Machine Learning Classifiers for Recognition of Online and Offline Handwritten Digits, Computer Engineering and Intelligent Systems, 4(13), 39-47

[12] Ovidiu, I. (2007), Applications of Support Vector Machines in Chemistry, Reviews in Computational Chemistry, 23,291-400

[13] Pradeep, J., Srinivasan, E., & Himavathi, S. 2011. Diagonal Based Feature Extraction for Handwritten Alphabets Recognition using Neural Network, International Journal of Computer Science and Information Technology (IJCSIT), 3 (1), 27-37

[14] Russell, S., & Norvig, P. 2003. Artificial Intelligence, A Modern Approach (second edition), Prentice Hall

[15] Simon, G., & Horst, B. 2004. Feature Selection Algorithms for the Generalization of Multiple Classifier Systems and their Application to Handwritten Word Recognition, Pattern Recognition Letters, 25 (11), 1323-1336

[16] Tappert, C., Ching, Y., Suen, Toru W. 1990. The State Of The Art In Online Handwriting Recognition, IEEE Transactions On Pattern Analysis And Machine Intelligence, 12(8),777-808

[17] Thomé, A. C. G. 2012. SVM Classifiers – Concepts and Applications to Character Recognition, Advances in Character Recognition, (2), 25-50

[18] Witten, I., & Frank, E. 2005. Data Mining, Practical machine learning tools and techniques (2nd ed.), Morgan Kaufmann

AUTHORS' BIOGRAPHIES

Ibrahim Adepoju ADEYANJU is a Lecturer at the Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria where he teaches undergraduate and postgraduate students, supervises students' project/thesis and does research in the area of Machine Learning, Pattern Recognition and Artificial Intelligence. Ibrahim obtained a Bachelor degree (B.Tech. with Honours) in Computer Engineering from Ladoke Akintola University of Technology, Ogbomoso, Oyo state, Nigeria and his Masters (MSc.) degree in Computing Information Engineering and Doctorate (PhD) degree from the Robert Gordon University, Aberdeen, United Kingdom. Dr. Adeyanju currently has several peer reviewed journal articles and conference papers across different research areas of Computer Science and Engineering including Pattern Recognition, Distributed Constraints Satisfaction, Artificial Intelligence, Information Retrieval, Information Systems and Machine Learning. Ibrahim is happily married with kids and enjoys reading during his leisure times.

Olusayo Deborah FENWA is a Senior Lecturer at the Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria where she teaches undergraduate and postgraduate students, supervises students' project/thesis and does research. Deborah obtained a Bachelor degree (B.Tech. with Honours) in Computer Science from Ladoke Akintola University of Technology, Ogbomoso, Oyo state, Nigeria and her Masters (MSc.) and Doctorate (PhD) degrees in Computer Science from the same university. Dr. (Mrs) Fenwa currently has several peer reviewed journal articles and conference papers across different research areas of Computer Science and Engineering including Pattern Recognition, Artificial Intelligence and Information Systems. Deborah is happily married with kids.

Elijah Olusayo OMIDIORA is a Professor of Computer Engineering at the Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria where he teaches undergraduate and postgraduate students, supervises students' project/thesis and does research in the area of Biometrics, Hardware design, Information Systems and Artificial Intelligence. Elijah obtained a Bachelor degree (B.Eng. with Honours) in Computer Engineering from Obafemi Awolowo University, Ile-Ife, Osun state, Nigeria, his Masters (MSc.) degree from University of Lagos, Nigeria and Doctorate (PhD) degree from Ladoke Akintola University of Technology, Ogbomoso, Nigeria. Prof. Omidiora has several peer reviewed journal articles and conference papers across different research areas of Computer Science and Engineering including Pattern Recognition, Information Systems, Artificial Intelligence and Control Systems. Elijah is happily married with kids.