JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

HANDWRITTEN CHARACTER RECOGNITION FROM IMAGE USING AI

[1]Dr. Harish B G, [2]Mr.Chetan Kumar G S, [3]Deekshith H M, [4]Lavanya S L, [5]Manjunath C M, [6]Mohan N J

[1]Professor, [2]Assistant Professor, [3,4,5,6]Student
Department of MCA
University BDT College of Engineering, Davangere-577004,Karnataka, India

Abstract: Truly created digit attestation has beginning late been of very vitality among the analysts in light of the movement of different Machine Learning, Deep Learning and Computer Vision estimations. In this report, I consider the results of verifiably the most generally utilized Machine Learning Algorithms like SVM, KNN and RFC and with Deep Learning calculation like multilayer CNN utilizing Keras and Tensorflow. Utilizing these, I had the choice to get the accuracy of 99.70% utilizing CNN (Keras) when veered from 97.91% utilizing SVM, 98.00% utilizing KNN, 98.10% utilizing RFC.

Keywords **-** image processing, handwritten character Recognition, artificial intelligence, Tensorflow, Convolutional neural networks(CNN), MNIST Benchmark Database.

I. INTRODUCTION

In India, at present we are having In this era where computer technologies and artificial intelligence is booming widely all around the globe, the role of deep learning and machine learning in recognizing the hand written digits has become very important. This is nothing but the ability of a machine to understand a hand written digit and then to classify it. In order to develop such machine learning systems, it is important for the machine being able to understand and then classify the hand written digits as the 10 different digits starting from 0 to 9. This targets in recognizing the hand written digits from various sources such as papers, images, letters etc. There are various machine learning algorithms and models built in the past for this same purpose. The algorithms such as K-Nearest Neighbor, Support Vector Machines, CNN, Random Forest, etc. but these methods although having the accuracy of 97% are not enough for the real world applications.

In the past decade, deep learning has become the hot tool for image processing, object detection, handwritten digit and character recognition etc. A lot of machine learning tools have been developed like scikit-learn, scipy-image etc. and pybrains, Keras, Theano, Tensorflow by Google, TFLearn etc. for Deep Learning. These tools make the applications robust and therefore more accurate. The Artificial Neural Networks can almost mimic the human brain and are a key ingredient in image processing field. These technologies are used to develop such a model with good accuracy to recognize and classify the hand written digits.

II. LITERATURE SURVEY

"Handwritten Digit Recognition based on Output Independent Multi-Layer Perceptron"

With handwritten digit recognition being an established and significant problem that is facing computer vision and pattern recognition, there has been a great deal of research work that has been undertaken in this area. It is not a trivial task because of the big variation that exists in the writing styles that have been found in the available data. Therefore, both, the features and the classifier need to be efficient. The core contribution of this research is the development of a new classification technique that is based on the MLP, which can be identified in handwritten documents as the binary digits '0' and '1'. This technique maps the different sets of various input data onto the MLP output neurons. An experimental evaluation of the technique's performance is provided. This evaluation is based on the well-known 'Pen-Based Recognition of Handwritten Digits' dataset, which is comprised of a total of 250 handwriting samples that are taken from 44 writers. The results obtained are very promising for such an approach in accurate handwriting recognition.

"Bangla text document categorization mistreatment random Gradient Descent (SGD) classifier, Vol.71, 2007"

This paper describes the Bangla Document Categorization mistreatment random Gradient Descent (SGD) classifier. Here, document categorization is that the task within which text documents are classified into one or a lot of of predefined classes supported their contents. The projected system will be divided into 3 steps: one. feature extraction incorporating term frequency (TF) and inverse document frequency (IDF), 2. classifier style mistreatment the random Gradient Descent (SGD) algorithmic rule by learning the distinct options, and 3. performance live mistreatment F1-score. within the experiments on BDNews24

documents, it's discovered that our projected methodology provides higher accuracy compared with the strategies supported Support Vector Machine (SVM) and Naive Bayesian (NB) classifier.

"Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis".

Neural networks are a strong technology for classification of visual inputs arising from documents. However, there's a confusing superfluity of various neural network strategies that are utilized in the literature and in trade. This paper describes a group of concrete best practices that document analysis researchers will use to induce sensible results with neural networks. the foremost necessary observe is obtaining a coaching set as giant as possible: we tend to expand the training set by adding a replacement variety of distorted knowledge. the subsequent most vital observe is that convolutional neural networks are higher fitted to visual document tasks than totally connected networks. we tend to propose that a straightforward "do-it-yourself" implementation of convolution with a versatile design is appropriate for several visual document issues. this easy convolutional neural network doesn't need complicated strategies, like momentum, weight decay, structure dependent learning rates, averaging layers, tangent prop, or perhaps finely-tuning the design. the top result's a really straightforward however general design which might yield progressive performance for document analysis. we tend to illustrate our claims on the MNIST set of English digit pictures.

"At liberty written numeral recognition mistreatment majority balloting classifier. In Parallel Distributed and Grid Computing"

Unconstrained offline written numeral recognition may be a difficult drawback. it's terribly tough to search out high recognition results employing a single classifier. This paper presents a straightforward profile, combined native & international options and majority balloting theme classifier for at liberty written numeral recognition. the easy profile feature is computed by mistreatment the left, right, prime and bottom profile of a picture. A feature vector of length 112 is created by combining all the profiles. The native feature vector is extracted by applying Daubechies moving ridge remodel on the four pictures that were obtained by applying the brandy operator, and therefore the international options that are obtained by applying the identical Daubechies moving ridge remodel on the initial image. A feature vector of length eighty is created by combining the sixty-four native and sixteen international options. The feature vectors are the intensity of a peel within the third level approximation component of a picture. during this experiment four neural network classifiers: Multilayer feed forward, Pattern recognition, Cascade forward, perform fitting neural network classifiers & 2 applied math classifiers: Linear discriminant analysis and KNN classifiers are used for classifying these options. A majority balloting theme has been performed with 3 neural network classifier and KNN classifier. The performance is tested on MNIST dataset. The network was trained on sixty,000 and tested on ten,000 numeral samples of that ninety-eight.05% take a look at samples are properly recognized.

III. EXISTING SYSTEM

The MNIST dataset, a subset of a larger set NIST, is a database of 70,000 handwritten digits, divided into 60,000 training examples and 10,000[2] testing samples. The images in the MNIST dataset are present in form of an array consisting of 28x28 values representing an image along with their labels.

IV. PROPOSED SYSTEM

For the purpose of hand written digits' recognition there exists various systems using various algorithms. There are systems which make use of SVM algorithm, Random Forest Algorithm, KNN algorithm, CNN etc. to recognize the hand written digits. These algorithms have shown accuracies between 85-95% which is not really sufficient for the purpose.

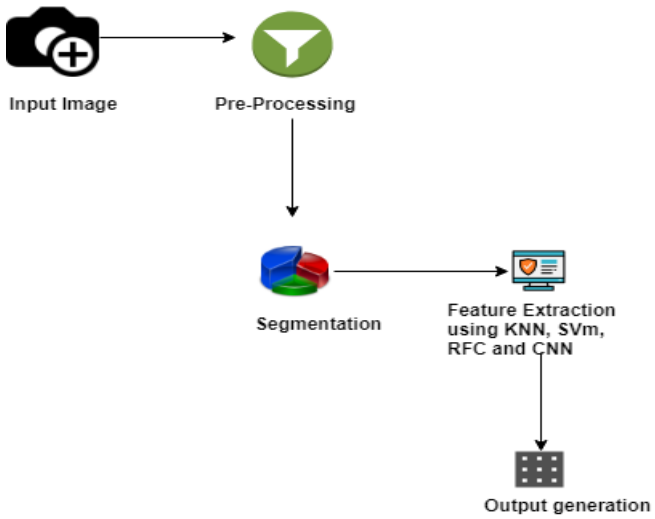

Figure 1: Block diagram

4.1TERMINOLOGIES

4.1.1 MNIST Dataset:

The MNIST dataset is a subset of a larger set NIST. It is a database consisting of around 70,000 handwritten digits, divided into 60,000 training and 10,000 testing samples. The images in the MNIST dataset are in form of an array consisting of 28x28 values representing an image along with their labels. This is the same in case of the testing images also.The pixels exist as an array of 784-d pixels and the values range from 0 to 255 i.e. 0 means background Black and 255 means it is White.

MNIST Dataset Format Analysis:

The MNIST data is provided in a specific format. In order to read the dataset, it is important to understand the format of the data present. Both the Training and Testing images and labels have the first two columns consisting of the "Magic Number" and the number of items in the file. The magic number has its first two bytes equal to zero. This magic number is read as MSB first.

4.1.2 Algorithm

Step 1: Load MNIST the dataset.
Step2: Label the information as preparing and testing set.
Step3: Train the classifier utilizing Random Forest calculation.
Step4: Then fit the Random Forest model with the information.
Step5: Total check number (digit) recognition is finished.
Step6: Confusion matrix displayed.
Step 7: Finally predict output.

4.1.3 CNN

For digit recognition using Deep Neural Networks, I am using multi-layer Deep Convolutional Neural Network [CNN]. I have also used two different tools, Keras and TensorFlow by Google to show the working accuracy of Deep Neural Network. A. Convolutional Neural Network A Convolutional Neural Network (CNN) is a type of feed forward Artificial Neural Network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. Convolutional Neural Networks consist of neurons that have learnable weights and biases. Each neuron receives some input, performs a dot product and optionally follows it with a non-linearity.

Accuracy comparison of KNN,SVM,RFC and CNN algorithms:

Algorithm	KNN	SVM	RFC	CNN
Percentage Accuracy	98.22%	98.00%	97.77	100%

V. RESULT AND ANALYSIS

In this paper we using the above techniques.

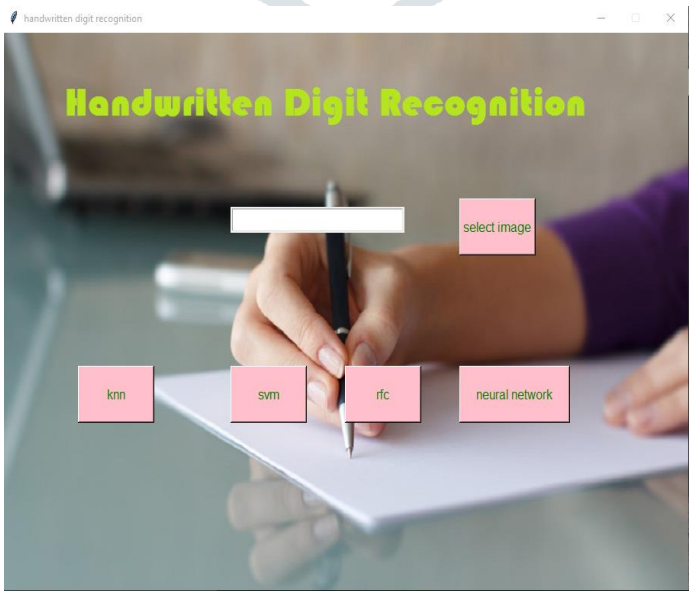

Fig 1: Home page.

```
training data points: 1212
validation data points: 135
testing data points: 450


Confusion matrix:
 [[43  0  0  0  0  0  0  0  0  0]
 [ 0 37  0  0  0  0  0  0  0  0]
 [ 0  0 38  0  0  0  0  0  0  0]
 [ 0  0  0 45  0  0  0  0  1  0]
 [ 0  1  0  0 54  0  0  0  0  0]
 [ 0  0  0  0  0 59  0  0  0  0]
 [ 0  0  0  0  0  0 45  0  0  0]
 [ 0  0  0  0  0  0  0 40  0  1]
 [ 0  1  0  0  0  0  0  0 36  1]
 [ 0  0  0  1  1  1  0  0  0 45]]

Accuracy: 0.9822222222222222

Percentage Accuracy: 98.22222222222223

Error Rate: 1.7777777777777715


Digit is: 5
```

Fig 2: KNN results.

```
training data points: 1212
validation data points: 135
testing data points: 450


Confusion matrix:
 [[42  0  0  0  1  0  0  0  0  0]
 [ 0 37  0  0  0  0  0  0  0  0]
 [ 0  0 38  0  0  0  0  0  0  0]
 [ 0  0  1 43  0  1  0  0  1  0]
 [ 0  0  0  0 55  0  0  0  0  0]
 [ 0  0  0  0  0 57  1  0  0  1]
 [ 0  0  0  0  0  1 44  0  0  0]
 [ 0  0  0  0  0  0  0 40  0  1]
 [ 0  2  0  0  0  0  0  0 36  0]
 [ 0  0  0  0  0  1  0  2  0 45]]

Accuracy: 0.9711111111111111

Percentage Accuracy: 97.11111111111111

Error Rate: 2.8888888888888857


Digit is: 6
```

Fig 3: RFC result.

```
training data points: 1212
validation data points: 135
testing data points: 450


Confusion matrix:
 [[43  0  0  0  0  0  0  0  0  0]
 [ 0 37  0  0  0  0  0  0  0  0]
 [ 0  0 38  0  0  0  0  0  0  0]
 [ 0  0  1 44  0  1  0  0  0  0]
 [ 0  1  0  0 54  0  0  0  0  0]
 [ 0  0  0  0  0 58  1  0  0  0]
 [ 0  0  0  0  0  0 45  0  0  0]
 [ 0  0  0  0  0  0  0 40  0  1]
 [ 0  1  0  0  0  1  0  0 36  0]
 [ 0  0  0  1  1  0  0  0  0 46]]

Accuracy: 0.98

Percentage Accuracy: 98.0

Error Rate: 2.0


Digit is: 4
```

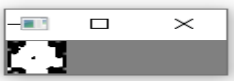

Fig 4: SVM result.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 24, 24, 16)        1216
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 12, 12, 16)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 8, 8, 32)          12832
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 4, 4, 32)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1000)              513000
_________________________________________________________________
dense_2 (Dense)              (None, 10)                10010
=================================================================
Total params: 537,058
Trainable params: 537,058
Non-trainable params: 0
_________________________________________________________________
WARNING:tensorflow:From C:\Users\knowx\Anaconda3\envs\tf16\lib\site-pac
ated. Please use tf.compat.v1.global_variables instead.

Digit is a 7 with 100.0% accuracy.
```

Fig 5: CNN result.

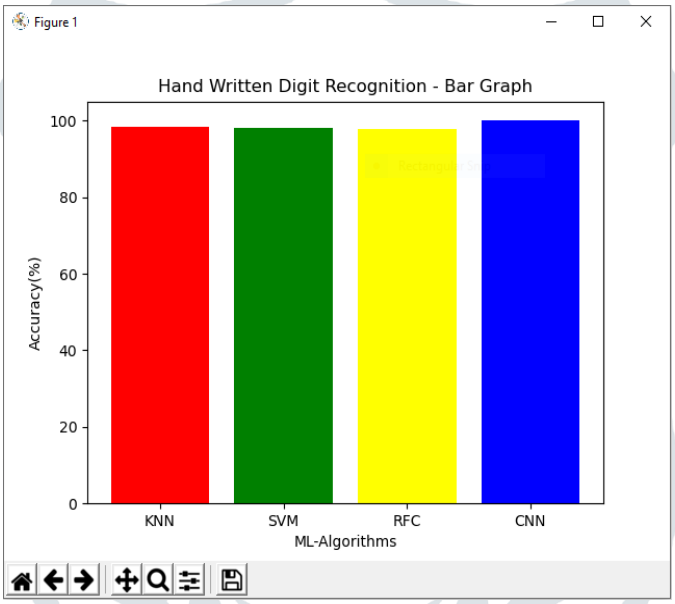

Fig 6: overall graph.

VI. CONCLUSION

The Convolutional Neural Network model was differentiated and models, for instance, SVM, KNN and RFC. In this way, we can to an end that the neural frameworks gave better result precision when appeared differently in relation to various estimations. The accuracy of CNN is gotten the opportunity to be 100%. Also, utilization of CNN using tensor stream gives a far prevalent outcome of 99.70%. Each instrument has its own complexity and accuracy. Notwithstanding the way that, we see that the multifaceted idea of the code and the method is bit more when stood out from normal Machine Learning counts yet looking at the accuracy achieved, it will in general be said that it is legitimized, regardless of all the difficulty. Moreover, the current execution is done simply using the CPU, it is depended upon to get essentially more parallelism and accomplish much better results.

REFERENCES

[1] Fatahi, M., 2014. MNIST handwritten digits.
[2] http://cvisioncentral.com/resources-wall/?resource=135
[3] https://en.wikipedia.org/wiki/Support_vector_machine
[4] http://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.
[5] http://scikitlearn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClas sifier.html.
[6] http://scikitlearn.org/stable/modules/generated/sklearn.svm.SVC.html
[7] Kumar, R., Goyal, M.K., Ahmed, P. and Kumar, A., 2012, December. Unconstrained handwritten numeral recognition using majority voting classifier. In Parallel Distributed and Grid Computing (PDGC), 2012 2nd IEEE International Conference on (pp. 284-289). IEEE.
[8] https://en.wikipedia.org/wiki/Support_vector_machine

[9] Kabir, F., Siddique, S., Kotwal, M.R.A. and Huda, M.N., 2015, March. Bangla text document categorization using Stochastic Gradient Descent (SGD) classifier.In Cognitive Computing and Information Processing (CCIP), 2015 International Conference on (pp. 1-4).IEEE.

[10] http://cs231n.github.io/convolutional-networks/

[11] Simard, P.Y., Steinkraus, D. and Platt, J.C., 2003, August. Best practices for convolutional neural networks applied to visual document analysis. In ICDAR(Vol. 3, pp. 958-962).