

HANDWRITTEN DIGIT RECOGNITION USING NEURAL NETWORKS AND MACHINE LEARNING

Sai Spoorthi Jupudi¹, Lakshita Sehgal², Anshal Aggarwal³, Sri Sahith Reddy⁴, Anirudh J S⁵

Abstract - We aim at constructing a machine that could deal with confine handwritings and deal with every digit personally. We also plan initially to train the machine using neural matrix after which make it discover every and each digit and even the cuts for the duration of the text meaning if someone strikes to take a look at it should be not noted. We may even attempt to educate with extraordinary neighborhood languages and educate it with neighborhood local languages.

1. INTRODUCTION

Motivation

Through this venture we purpose at breaking the limitations of flawed Handwritten patterns of people. The way they write all through version the use of Neural networks and training matrix.

We plan to broaden a clever digit popularity system the use of Neural Networks aside from the conventional strategies in which the exclusive varieties of digits aren't recognized in distinct handwritings we are able to attempt to get over this and try examine such numerous structures right from the supply code level.

Significance

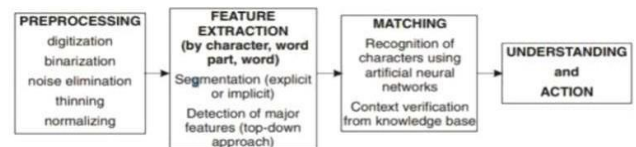
The significance of the hassle is very excessive as in recent times in the course lot of troubles are faced in recognizing words which can be strike off in another way with the usage of specific patterns. It could be very hard in the existing systems to understand that which phrases are strike off and which of them are not We plan to design the system in this type of manner that understand all the characters and the abnormalities if any have to be skipped .We may also attempt to educate the neural networks in this sort of manner that it can recognize at least some percent of the nearby languages for person convenience.

This will help to reduce the computation time of individual reputation and will also be useful for the society as can be used by people of numerous exclusive sections of the society to experiment written files mechanically using neural networks and gadget mastering.

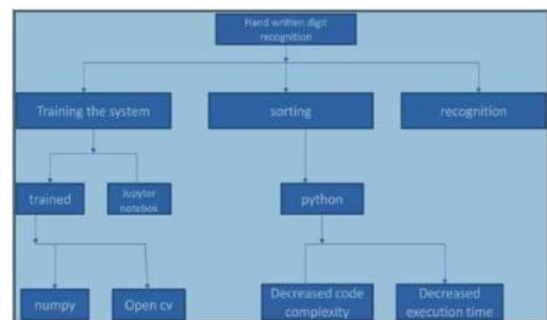
1.1 Scope and Applications

Use equipment to examine the instructiveness from a source code stage. Vulnerabilities may be identified without problems using the system. Manipulate the pixel length and different parameters to lower the computation time. Use in Cyber forensics for case investigation functions and for exam

correction purposes. There are additionally different areas in which it's far of massive importance like signature verification, conversion of languages, etc.



1.2 Implementation



As shown in above figure, In this the primary and principal access and physical element is the schooling matrix and Jupiter notebook that have been used of as the base for our neural schooling within the algorithm. We are storing the matrix the use of python so that the gadget can discover every and each precise digit with the aid of its pixel range. The python code is executed in this type of manner that it presents the time as well as the accuracy each set of rules will deliver.

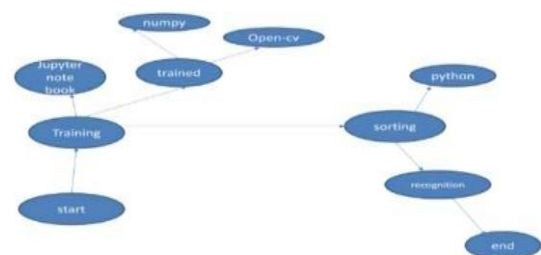


fig4: Functional Architecture of the proposed system

As visible in figure - 4, we clearly can display that how we can supply the input of the digits to the device and via the training code snippet that we put inside the neural network of the system it's going to help us to understand the digits. It first trains itself thru the schooling matrix after which it types out the digits at the paper and for that reason supply us the output n pinnacle every digit what's written at the sheet. All that is finished via most important algorithms of SVM and KNN.

In k-NN regression, the output is the belongings price for the item. This value is the average of the values of its okay nearest pals.

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-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, a useful technique can be used to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of $1/d$, where d is the distance to the neighbor. The neighbors are taken from a set of objects for which the class (for k- NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

2.1 Drawbacks

A peculiarity of the k-NN algorithm is that it is sensitive to the local structure of the data. The algorithm is not to be confused with k-means, another popular machine learning technique.

2.2 COMPLEXITY ANALYSIS

Complexity is taken into consideration. Increased instructiveness by changing the dimensions. More accurate and precise results. Improvement in security of the system (Reduction in hackable systems). Also reduction in processing time is the innovation in our system. Our System also has the added feature of choosing the best algorithm according to the resources and time we have. We have created a graph that depicts that how execution time of KNN is lesser than that of SVM and how the accuracy is higher as compared to SVM for hand digit recognition using neural networks.

```
In [1]: TRAIN_PATH = "digits.png"
TRAIN_DATA = "mnist_train_digits.pkl"
TEST_DATA = "mnist_test_digits.pkl"

digits, labels = load_digits(TRAIN_PATH)
train_data_shape = digits.shape
test_data_shape = labels.shape

digits, labels = shuffle(digits, labels, random_state=20)
train_data_shape = digits.shape
test_data_shape = labels.shape

# Create and fit the K-Nearest Neighbors classifier
knn = KNeighborsClassifier(n_neighbors=1)
knn.fit(train_data_shape, labels)

# Predict the labels for the test data
predicted_labels = knn.predict(test_data_shape)

# Calculate the accuracy
accuracy = np.mean(predicted_labels == labels)

print("Accuracy: %f" % accuracy)
```

SVM Algorithm Used for digit recognition which provides lesser accuracy because it calculates keeping all the planes in mind and this may predict a high amount of wrong results when considering striking out of digits for exam evaluation system thus we prefer KNN Over SVM in the given application using both standard and non- standard datasets the same comparative study comes up

NOW A COMPARITIVE ANALYSIS BETWEEN SVM (Support Vector Machine) and KNN (K-Nearest Neighbor) Algorithm TO PREDICT WHICH ONE IS BETTER ON THE GIVEN EXAM EVALUATION SYSTEM.

Table 1 – Time Accuracy

Serial Number	Total Time Accuracy
KNN	0.48 Sec 82%
SVM	0.12 Sec 30%

3. CONCLUSION

We have sincerely visible that KNN gives us greater accuracy in comparison to SVM inside the given examination assessment device. Thus through the above shown output we absolutely see that after we strike out digits accuracy of SVM decreases notably in comparison to KNN. The pattern consequences have been examined each for popular and non-trendy facts units. We can honestly finish that KNN is a better model for the given utility. My version is better than the rest as it does a comparative examine between the various fashions and concludes that is the nice one for our unique application which ha no longer been blanketed on any of the research papers until date.

3.1 FUTURE WORK

Complexity is taken into consideration. Increased instructiveness by converting the scale. More accurate and specific outcomes. Improvement in security of the machine (Reduction in hackable systems). Also discount in processing time is the innovation in our device. Our System additionally has the delivered function of selecting the high-quality set of rules in step with the assets and time we've. We have created a graph that depicts that how execution time of KNN is lesser than that of SVM and the way the accuracy is higher in comparison to SVM for hand digit popularity the usage of neural networks. It demonstrates that customers—even users who are not domain professionals can frequently assemble exact classifiers, without any assist from a getting to know algorithm, using a simple two- dimensional visible interface. Moreover, using human psychology within the learning algorithm and its consequent stages are the brand new innovation this is but to be applied and proven.

KNN algorithm Applied to recognize the digits using k-nearest points which provides us with a high amount of accuracy which is beneficial for us.

3.2 REFERENCES

[1] Handwritten digit recognition using neural networks and dynamic zoning with stroke- based descriptors David Álvarez-León Ramón-Ángel Fernández-Díaz Lidia Sánchez-Gonzalez José-Manuel Alija-Pérez(OXFORD JOURNALS) Logic Journal of the IGPL, Volume 25, Issue 6, 1 December 2017, Pages 979-990.

[2] 'Interactive machine learning: letting users build classifiers' International Journal of Human-Computer Studies Volume 55, Issue 3, September 2001, Pages 281-292

[3]Saeed AL-Mansoori Int. Journal of Engineering Research and Applications www.ijera.com ISSN : 2248-9622, Vol. 5, Issue 5, (Part -3) May 2015, pp.46-51

[4] D. Álvarez; R. Fernández ; L. Sánchez 'Stroke-based intelligent character recognition using a deterministic finite automaton' Logic Journal of the IGPL (Volume: 23, Issue: 3, June 2015)

[5] Nilsson Nils J., "Principles of Artificial Intelligence". Palo Alto, CA: Tioga Pub. Co., 1980, xv, 476 p.

[6] Y. Nakajima S. Mori S. Takegami S. Sato 'Global methods for stroke segmentation' International Journal on Document Analysis and Recognition July 1999, Volume 2, Issue 1, pp 19-23

[7] Toru Wakaharaa Yukihiro Yamashita 'k-NN classification of handwritten characters via accelerated GAT correlation'

[8] Nilsson Nils J., "The Mathematical Foundations of Learning Machines". 2nd Edition (Nilsson N. Learning machines. Mir, Moscow, 1967), San Mateo, CA: Morgan Kaufmann Publishers Inc., 1990, xxvii, 138 p.

[9] K. Nakamura, S. Iwai, "Topological fuzzy sets as a Quantitative Description of analogical inference and its application to Question – Answering systems for information retrieval", SMC-12, No2 (March), 1982, pp. 193-204.

[10] A. S. Vairadian, M. N. Petuxov, M. M. Chelysheff, V. I. Chuchkin, "Application theory and methods of pattern recognition in ACS". Moscow: MEPhI, 1978, 68 p.

[11] Oivind Due Trier, Anil K. Jain & Torfinn Taxt 'Feature extraction methods for character recognition-A survey'

[12] Hamed Shah-Hosseini 'Binary tree time adaptive self-organizing map Neurocomputing' Volume 74, Issue 11, May 2011, Pages 1823-1839

[13] Recognition of handwritten sentences using a restricted lexicon Thierry Paquet Yves Lecourtier Pattern Recognition Volume 26, Issue 3, March 1993, Pages 391-407.

[14] Il-Seok Oh Ching Y. Suen A class-modular feedforward neural network for handwriting recognition Pattern Recognition Volume 35, Issue 1, January 2002, Pages 229-244

[15] Amlan Kundu, Yang He, Paramvir Bahl 'Recognition of handwritten word: First and second order hidden Markov model based approach' Pattern Recognition Volume 22, Issue 3, 1989, Pages 283-297