International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Handwritten Recognition with Neural Network

Chirag Dodiya[1], DR. Gayatri S Pandi[2]

[1] Post Graduate Scholar, Post Graduate Department LJ University, Ahmedabad
[2] Head of Department, Post Graduate Department, LJ University

ABSTRACT

The paper will describe the best approach to get more than 90% accuracy in the field of Handwritten Recognition (HCR). There has been plenty of research done in the field of HCR but still, it is an open problem as we are still lacking in getting the best accuracy. In this paper, the offline handwritten recognition will be done using a Convolutional neural network and TensorFlow. A method called Soft Max Regression is used for assigning the probabilities to handwritten characters being one of the several characters as it gives the values between 0 and 1 summing up to 1. The purpose is to develop the software with a very high accuracy rate and with minimal time and space complexity and also optimal.

Keywords: Handwriting,Recognition, Neural Network

Introduction

Understanding the handwritten characters or typed documents is simple to human beings as we have the ability to learn. The same ability can be induced to the Machines also by the use of Machine Learning and Artificial Intelligence. The field that deals with this problem are called the OCR or also known as Optical Character Recognition. It's the area of study in various fields such as recognizing patterns, also Image vision, and also AI. This is the system for changing electronic and image text into digital characters to be read by the machines. The time used in entering the data and also the storage space required by the documents can be highly reduced by the use of OCR in other words, it can be retrieved fast. By using the OCR in the banking field, legal scenarios, etc. many important and sensitive documents can be processed faster without human intervention. OCR in advance can be inferred in two ways based on the type of the text and document acquisition (Figure 1). Further, if we take into consideration the text type, then OCR is further of two types, HCR (Handwritten Character Recognition) which is intelligent recognition of the handwritten text, and PCR (Printed Character Recognition). We need a high recognition ability due to the varying handwriting of humans. Many times the writing style of the same individual is different at times. Further OCR is characterized by two forms Offline and Online recognition systems based on acquiring the documents. Offline System deals with recognizing the pre-written document acquired through various input methods. But in the Online recognizing system, the writing is recognized the moment it is written. The device used for the online system is an Electric pen which is used for writing the letters or words on the device called a digitizer and on the basis of the pen movement the input is recorded.

Related Works

Each person has a unique writing style. Written characters are varying with size, width, and thickness. The purpose of this project is to convert handwritten words into readable standard words.

Handwriting Recognition

Pattern recognition is extremely difficult to automate. Animals recognize various objects and make sense out of a large amount of visual information, apparently requiring very little effort. In order to simulate animal recognition capabilities, physical limitations must allow the system to be as realistic as possible. This necessitates the study and simulation of Artificial Neural networks. A Neural Network consists of nodes each performing simple computations and each connected with a signal from one node to another denoted by a number called "connection strength and weight" which indicates how much each node contributes

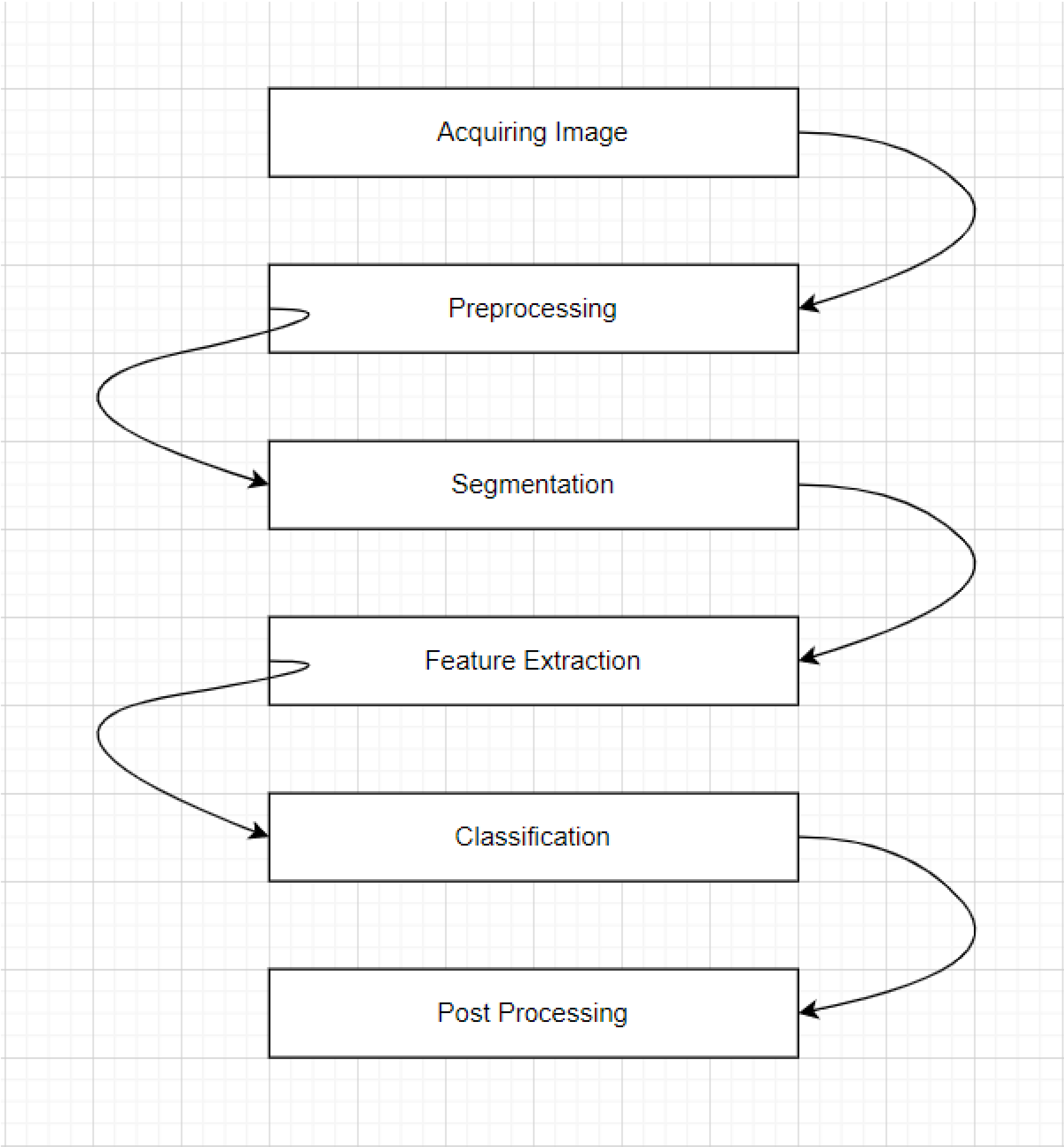

Figure 1: Proposed System

Acquiring Image

Digitized/Digital Image is initially taken as input. The most common of these devices is the electronic tablet or digitizer. These devices use a pen that is digital in nature. Input images for handwritten characters can also be taken by using other methods such as scanners, photographs, or by directly writing on the computer by using a stylus.

Pre-processing

Pre-processing is the basic phase of character recognition and it's crucial for a good recognition rate. The main objective of pre-processing steps is to normalize strokes and remove variations that would otherwise complicate recognition and reduce the recognition rate. These variations or distortions include the irregular size of text, missing points during pen movement collections, jitter present in the text, left or right bend in handwriting, and uneven distances of points from neighbouring positions. Pre-processing includes five common steps, namely, size normalization and centering, interpolating missing points, smoothing, slant correction, and resampling of points.

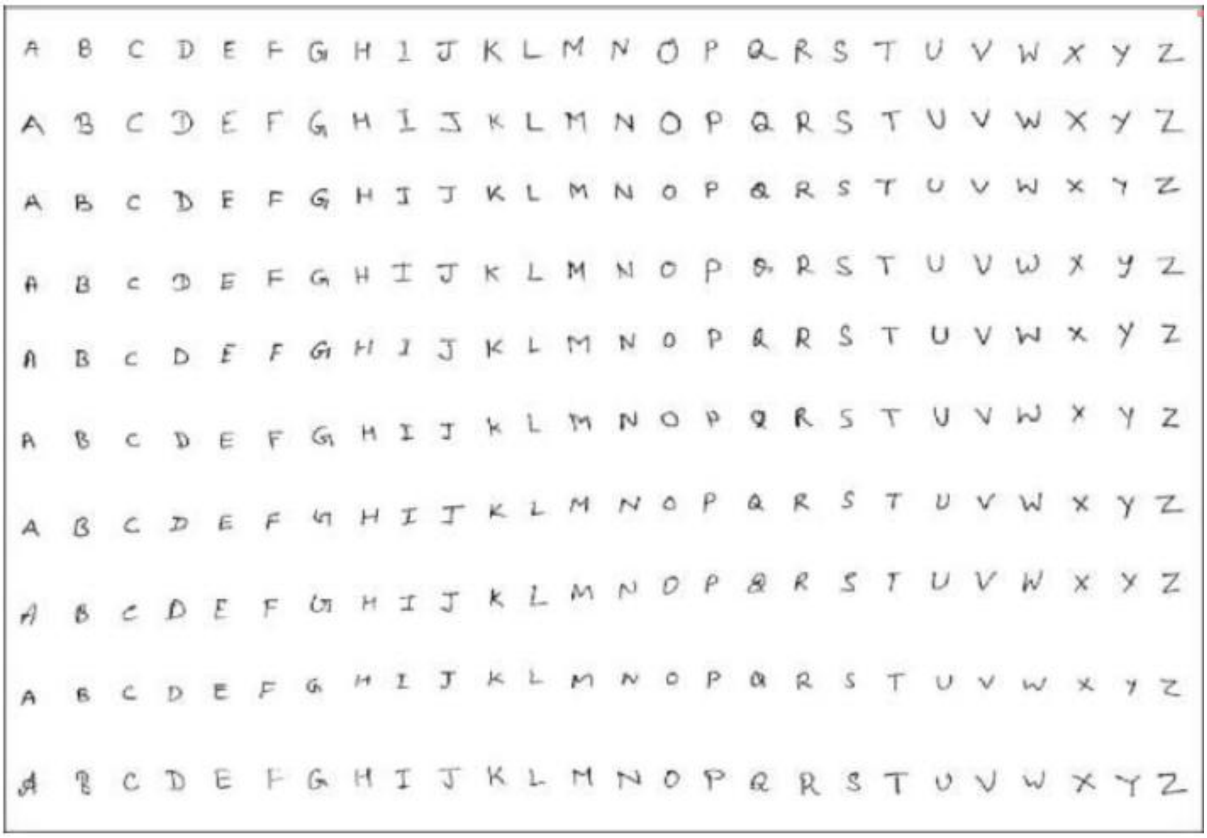

Figure 2: Character Set Of Handwritten Character

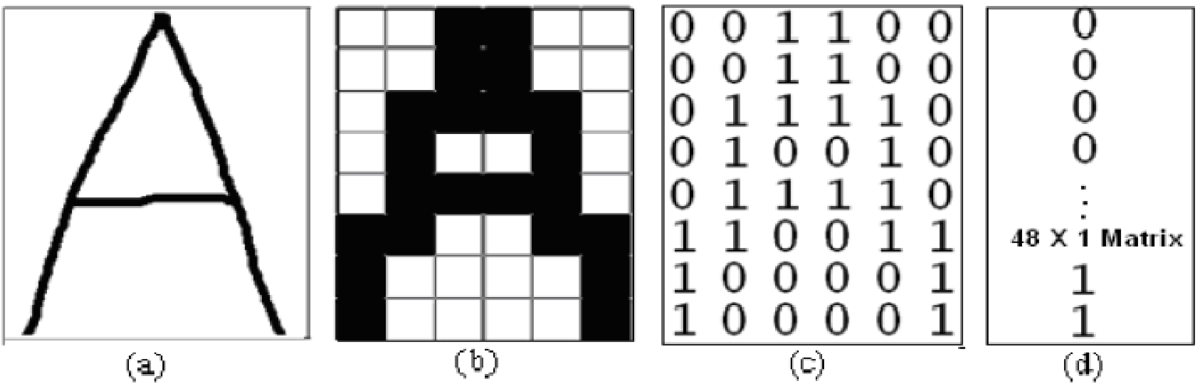

Figure 3. Digitization Process

Segmentation

In the segmentation stage, an image is decomposed into sub-images of individual characters. Segmentation includes:

1. line segmentation which is the separation of the line from paragraph,
2. Word segmentation which is the separation of words from the line.
3. Character segmentation which is the separation of character from words.

Character segmentation is performed if the segmentation-based method is adopted for cursive word recognition, for the holistic method character segmentation is not performed

Feature extraction

The aim of feature extraction is to allow the extraction of the pattern which is most important for the classification. Some of the Feature extraction techniques like Principle Component Analysis (PCA), Scale Invariant Feature Extraction (SIFT), Linear Discriminant Analysis (LDA), Histogram, Chain Code (CC), zoning, and Gradient-based features can be applied to extract the features of individual characters. All of these features are used to train the given system.

Thereby flattening the array into a vector of 28*28 = 784 numbers. Thus, the image now converges to a minimal bunch of arrays in a 784-cell dimension of a highly efficient structure. The image now becomes a tensor of the n-dimensional array.

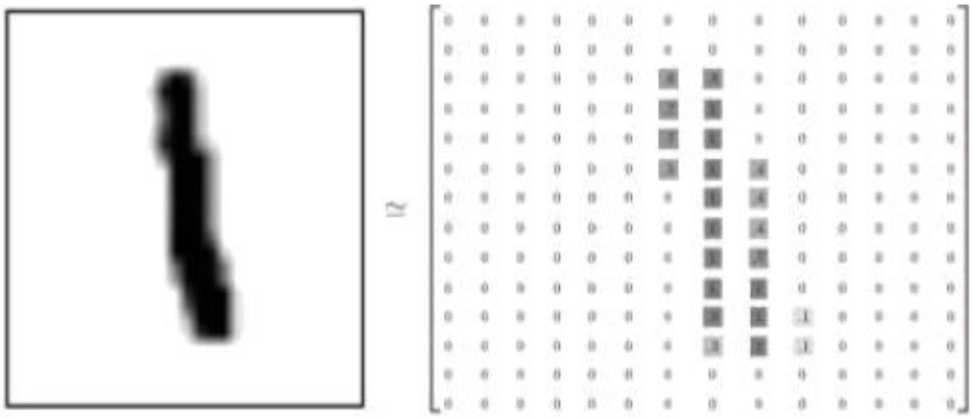

Figure 4. Feature extraction of data

Classification

The classification stage is the decision-making part of a recognition system and it uses the features extracted in the previous stage. The feature vector is denoted as X where X = (f1, f2,....., fd) where f denotes features and d is the no. features extracted from the character. Based on the comparison of feature vector characters are efficiently classified into appropriate classes and recognized. Classifiers are based on two types of learning methods.

1. Supervised-learning
   In supervised learning training data with correct detail of class is applied to train a model. This model is used to test data for proper classification. Training data includes both the input and the desired results. The model undergoes a learning process and based on this learning it classifies test data.
   For example SVM, HMM, etc.

2. Unsupervised-learning
   An unsupervised learning model is not provided with training data. It does not require learning. The model classifies test data based on statistical properties and by their spatial grouping and considering their nearest neighbor. For example Clustering, k means, etc.

u

Post Processing

The final and last phase of character recognition is Post-processing. It is the procedure for correcting the misclassified output by using natural language. It processes output by getting it after the shape has been recognized. If the shape is recognized purely then the accuracy can be improved in accordance with the knowledge of the language. Shape recognizers behave differently for different handwriting inputs. For the few, it results in an individual character of string while also including a few numbers of alternates in the second case, by including the measure of confidence in every alternative

Implementation

```
import matplotlib.pyplot as plt
import cv2
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Flatten, Conv2D, MaxPool2D, Dropout
from keras.optimizers import SGD, Adam
from keras.callbacks import ReduceLROnPlateau, EarlyStopping
from keras.utils import to_categorical
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.utils import shuffle

data = pd.read_csv(r".//A_Z Handwritten Data.csv").astype('float32')
```

```
print(data.head(10))

X = data.drop('0',axis = 1)
y = data['0']

train_x, test_x, train_y, test_y = train_test_split(X, y, test_size = 0.2)

train_x = np.reshape(train_x.values, (train_x.shape[0], 28,28))
test_x = np.reshape(test_x.values, (test_x.shape[0], 28,28))

print("Train data shape: ", train_x.shape)
print("Test data shape: ", test_x.shape)

word_dict =
{0:'A',1:'B',2:'C',3:'D',4:'E',5:'F',6:'G',7:'H',8:'I',9:'J',10:'K',11:'L',12:'M',13:'N',14:'O',15:'P',16:'Q',17:'R',18:'S',19:'T',20:'U',21:'V',22:'W',2
3:'X', 24:'Y',25:'Z'}

y_int = np.int0(y)
count = np.zeros(26, dtype='int')
for i in y_int:
    count[i] +=1

alphabets = []
for i in word_dict.values():
alphabets.append(i)

fig, ax = plt.subplots(1,1, figsize=(10,10))
ax.barh(alphabets, count)

plt.xlabel("Number of elements ")
plt.ylabel("Alphabets")
plt.grid()
plt.show()

shuff = shuffle(train_x[:100])

fig, ax = plt.subplots(3,3, figsize = (10,10))
axes = ax.flatten()

for i in range(9):
    _, shu = cv2.threshold(shuff[i], 30, 200, cv2.THRESH_BINARY)
    axes[i].imshow(np.reshape(shuff[i], (28,28)), cmap="Greys")
plt.show()

train_X = train_x.reshape(train_x.shape[0],train_x.shape[1],train_x.shape[2],1)
print("New shape of train data: ", train_X.shape)

test_X = test_x.reshape(test_x.shape[0], test_x.shape[1], test_x.shape[2],1)
print("New shape of train data: ", test_X.shape)



train_yOHE = to_categorical(train_y, num_classes = 26, dtype='int')
print("New shape of train labels: ", train_yOHE.shape)

test_yOHE = to_categorical(test_y, num_classes = 26, dtype='int')
print("New shape of test labels: ", test_yOHE.shape)

model = Sequential()
```

```
model.add(Conv2D(filters=32, kernel_size=(3, 3), activation='relu', input_shape=(28,28,1)))
model.add(MaxPool2D(pool_size=(2, 2), strides=2))

model.add(Conv2D(filters=64, kecrnel_size=(3, 3), activation='relu', padding = 'same'))
model.add(MaxPool2D(pool_size=(2, 2), strides=2))

model.add(Conv2D(filters=128, kernel_size=(3, 3), activation='relu', padding = 'valid'))
model.add(MaxPool2D(pool_size=(2, 2), strides=2))

model.add(Flatten())

model.add(Dense(64,activation ="relu"))
model.add(Dense(128,activation ="relu"))

model.add(Dense(26,activation ="softmax"))
model.compile(optimizer = Adam(learning_rate=0.001), loss='categorical_crossentropy', metrics=['accuracy'])

history = model.fit(train_X, train_yOHE, epochs=1,  validation_data = (test_X,test_yOHE))

model.summary()
model.save(r'model_hand.h5')

print("The validation accuracy is :", history.history['val_accuracy'])
print("The training accuracy is :", history.history['accuracy'])
print("The validation loss is :", history.history['val_loss'])
print("The training loss is :", history.history['loss'])

fig, axes = plt.subplots(3,3, figsize=(8,9))
axes = axes.flatten()

for i,ax in enumerate(axes):
   img = np.reshape(test_X[i], (28,28))
ax.imshow(img, cmap="Greys")

   pred = word_dict[np.argmax(test_yOHE[i])]
   ax.set_title("Prediction: "+pred)
ax.grid()

   img = cv2.imread(r'.//hw1.jpeg’)
img_copy = img.copy()

img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
img = cv2.resize(img, (400,440))

img_copy = cv2.GaussianBlur(img_copy, (7,7), 0)
img_gray = cv2.cvtColor(img_copy, cv2.COLOR_BGR2GRAY)
_, img_thresh = cv2.threshold(img_gray, 100, 255, cv2.THRESH_BINARY_INV)

img_final = cv2.resize(img_thresh, (28,28))
img_final =np.reshape(img_final, (1,28,28,1))

img_pred = word_dict[np.argmax(model.predict(img_final))]

cv2.putText(img, "Dataflair _ _ _ ", (20,25), cv2.FONT_HERSHEY_TRIPLEX, 0.7, color = (0,0,230))
cv2.putText(img, "Prediction: " + img_pred, (20,410), cv2.FONT_HERSHEY_DUPLEX, 1.3, color = (255,0,30))
cv2.imshow('Dataflair handwritten character recognition _ _ _ ', img)

while (1):
   k = cv2.waitKey(1) & 0xFF
```

```
  if k == 27:
     break
cv2.destroyAllWindows()
```

Experiment result

Conclusion

Immense work and research have been done on handwritten separate character recognition. But so far 100% accuracy is not achieved which gives scope for further work in this direction. Separate characters give good accuracy but word recognition is affected by the different writing styles. The holistic method eliminates the complicated segmentation but they use a limited vocabulary. Segmentation-based method due to its complexity acquire less accuracy. Good accuracy is observed in the classifier where the scope of words is limited to fixing numbers as it has to deal with a limited number of variations.

Advantages

A handwriting recognition system makes it easier to draft and produce documents as well as create useful applications such as keyboard-free input. A pen can provide powerful editing, marking, drawing, etc. functions when used with desktop computers. With the advancement of handwriting recognition technology, applications such as longhand note-taking in the classroom will become more common. They could write their documents and have them converted to text instantly without having to go through the iterative process of having their secretary type them.

Disadvantages

The disadvantage is that it is not done in real-time as a person writes and therefore not appropriate for immediate text input Sometimes, characters look very similar, making it hard for a computer to recognize accurately. Finding a proper spelling mistake is not work always.

Future Work

- As mentioned in Section VII, research in OCR domains is usually done in some of the most widely spoken languages. This is partially due to the non-availability of datasets in other languages. One of the future research directions is to conduct research on languages other than widely spoken languages, i.e., regional languages and endangered languages. This can help preserve the cultural heritage of vulnerable communities and will also create a positive impact on strengthening global synergy.

- Another research problem that needs the attention of the research community is to build systems that can recognize on-screen characters and text in different conditions in daily life scenarios, e.g. text in captions or news tickers, text on signboards, text on billboards, etc. This is the domain of ``recognition / classification / text in the wild". This is a complex problem to solve as a system for such a scenario needs to deal with background clutters, variable illumination conditions, variable camera angles, distorted characters, and variable writing styles.

- To build a robust system for ``text in the wild", researchers need to come up with challenging datasets that are comprehensive enough to incorporate all possible variations in characters. One such effort is [218]. In another attempt, the research community has launched ``ICDAR 2019: Robust reading challenge on multilingual scene text detection and recognition"[219]. The aim of this challenge invites research studies that propose a robust system for multi-lingual text recognition in daily life or the ``in the wild" scenario. Recently report for this challenge has been published and winner methods for different tasks in the challenge

References

[1] Chirag Patel, Ripal Patel, International Journal of Scientific and Engineering Research · April 2011

[2]Homayoon Beigi, An Overview of Handwriting Recognition, Research gate, 25 June 2013.

[3]T.L. Dimond, \Devices for Reading Handwritten Characters," Proc. of Eastern Joint ComputerConference,December1957,pp.232-237.

[4]CharlesC.Tappert,ChingY.Suen,andToruWakahara,\TheStateoftheArtinOn-LineHandwriting Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.12,No.8,Aug.1990,pp.787-808.

[5] Wacef Guerfali and R ejean Plamondon, \Normalizing and Restoring On-line Handwriting," Pat-ternRecognition,Vol.26,No.3,1993,pp.419-431.

[6]T. Fujisaki, T. E. Chefalas, J. Kim and C. C. Tappert, \Online Recognizer for Runon HandprintedCharacters," Proc. of the Tenth International Conference on Pattern Recognition, Atlantic City,NewJersey,June16-21,1990,pp.450-454.

[7]Jibu Mathew C, Ravi C. Shinde, Prof. C. Y. Patil, Segmentation Techniques for Handwritten script Recognition System, International Conference on Circuit, Power and Computing Technologies [ICCPCT], 2015

[8] https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be

[9] Sara Aqab, Muhammad Usman Tariq, International Journal of Advanced Computer Science and Applications, Vol. 11, No. 7, 2020

[10] Dr. Yusuf Perwej, Neural Networks for Handwritten English Alphabet Recognition, 2011

[11] Handwritten Character Recognition from Images using CNN-ECOC for Mayur Bhargab Bora, Dinthisrang Daimary, Khwairakpam Amitab, Debdatta Kandar, 2019

[12] Megha Agarwal, Shalika, Vinam Tomar, Priyanka Gupta for Handwritten Character Recognition using Neural Network and Tensor Flow, 2019

[13] Shital Thakkar for Handwritten Character Recognition in English: A Survey, 2015

[14] Ayush Purohit , Shardul Singh Chauhan for A Literature Survey on Handwritten Character Recognition 2016

[15] Radmilo M.Bozinovic and Sargur N. Srihari, “Off-line Cursive Script word recognition”, IEEE Transactions On Pattern Analysis and Machine Intelligence, Vol.11. No. 1, January 1989.

[16] H. Bunke, M. Roth and E.G.Schukat-Talamazzini, “Offline Cursive Handwriting Recognition Using Hidden Markov Models”, Pattern Recognition, Vol. 28, No. 9, 1995 Elsevier Science Ltd.

[17] Nafiz Arica, “An Off-line Character Recognition System for free style Handwriting”, a thesis submitted to the graduate school of natural and applied sciences of the middle east technical university,1998.

[18] Yong Haw Tay, Pierre-Michel Lallican, Marzuki Khalid, Christian Viard-Gaudin, Stefan Knerr,” An Offline Cursive Handwritten Word Recognition System”, IEEE Catalogue No. 01 CH37239 2001.

[19] Anshul Gupta, Manisha Srivastava and Chitralekha Mahanta,” Offline Handwritten Character Recognition International Conference on Computer Applications and Industrial Electronics (ICCAIE), 2011.

[20] J. Pradeep, E. Srinivasan and S. Himavathi,”Neural Network Based Recognition System Integrating Feature Extraction and Classification for English Handwritten”, International Journal of Engineering (IJE) Transactions B: Applications Vol. 25, No.

2, (May 2012) 99-106.

[21] D. K. Patel, T. Som and M. K Singh," Improving the Recognition of Handwritten Characters using Neural Network through Multiresolution Technique And Euclidean Distance Metric", International Journal of Computer Applications (0975 – 8887) Volume 45– No.6 May 2012.

[22] M. Blumenstein, B. Verma and H. Basli," A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Characters", Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03) 0-7695-1960-1/03 $17.00 © 2003 IEEE

[23] Sumedha B. Hallale, Geeta D. Salunke, " Twelve Directional Feature Extraction for Handwritten English Character Recognition", International Journal of Recent Technology and Engineering (IJRTE)ISSN:2277-3878, Volume-2, Issue-2, May 2013.

[24] Amit Choudhary, Rahul Rishi and Savita Ahlawat, "Off-Line Handwritten Character Recognition using Features Extracted from Binarization Technique", 2212-6716 © 2013 American Applied Science Research Institute doi:10.1016/j.aasri.2013.10.045

[25] Rafael M. O. Cruz, George D. C. Cavalcanti and Tsang Ing Ren," An Ensemble classifier for offline Cursive character recognition using multiple features Extraction technique", 978-1-4244-8126- 2/10/$26.00 ©2010 IEEE

[26] Singh, Sameer, Mark Hewitt,"Cursive Digit And Character Recognition on Cedar Database", Pattern Recognition, 2000.Proceedings. 15th international conference on. Vol. 2. IEEE 2000.

[27] K. Gaurav and Bhatia P. K., "Analytical Review of Preprocessing Techniques for Offline Handwritten Character Recognition", 2nd International Conference on Emerging Trends in Engineering & Management, ICETEM, 2013.

[28] Salvador España-Boquera, Maria J. C. B., Jorge G. M. and Francisco Z. M., "Improving Offline Handwritten Text Recognition with Hybrid HMM/ANN Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 4, April 2011.