Handwritten Character Recognition with Motion Detection

1st Rakib Raihan Fahim
dept. of CSE
East West University
email: 2017-1-60-143@std.ewubd.edu

2nd Halima Begum
dept. of CSE
East West University
email: 2017-1-60-169@std.ewubd.edu

3rd Hasnat Mahbub
dept. of CSE
East West University
email:2017-1-60-052@std.ewubd.edu

4th Md. Iftekhar Ahmed Nasim
dept. of CSE
East West University
email:2017-1-60-117@std.ewubd.edu

***Abstract*—Handwritten character recognition has been a very firm topic of interest for the researchers. Pattern recognition is a huge field and handwritten character recognition is one of the most challenging in the field of recognizing patterns. there has been plenty of research done on this field but still, it is an open problem. This paper presents a technique to recognize handwritten characters with high accuracy and robust solution. This technique is to receive data via motion and detecting the character by analyzing it from a large group of handwritten characters. The motion can be detected through a webcam. In this approach, data is trained so that a robust solution can be found. *Index Terms*—Handwritten characters, predictions, pattern, recognition, motion detection, handwritten characters recognizer, webcam.**

I. Introduction

Recognizing handwriting and handwritten characters is a simple ability of the human mind. Writing is an everyday part of our daily lives and it has been a significant part of humans for thousands of years. But now we live in a very tech-savvy world. This is the new era of technology. And this ability to recognize handwritten characters can be induced to the machines also by the use of Machine Learning and Artificial Intelligence. In recent years, remarkable progress has been made in the field of pattern recognition. Handwritten character recognizer is a part of a huge field of pattern recognition. It is an ongoing field of research encompassing artificial intelligence, computer vision, and pattern recognition. Though recognizing handwritten character is a completely solved problem but the researchers are still working on different algorithms and systems to have a more robust solutions and reliable systems. Recognition of handwritten characters has been a big field of research for years. [1] [2] [3]. Various kinds of algorithms and systems have been introduced and refined by researchers for years. These systems receive input of handwritten characters via images, paper documents, screens, and motions and predict the outcome. Several approaches have been made for predicting handwritten characters over the years. Our system is primarily focused on recognizing handwritten characters with motions via webcam. After receiving the character by motion, it analyzes the character with a set of handwritten characters and predicts the outcome.

II. Related Work

A lot of work has been done on pattern recognition as it is a huge field in the past decede. [1] [2] [3]

Batuhan Balci et al. proposed various architectures to recognize and identify the handwritten characters and then to convert it into the digital format. [4].A.L. Koerich et al. proposed a system to improve an existing state-of-art unconstrained offline handwritten word recognizing system from a big data set that is writer independent. [5] This handwritten recognizing system(HRS) includes segmentation and feature extractions with a pre-processing stage (eliminating some stages which are not required for recognition). The goal is to recognize a word with an HMM-based word recognizing system and verify it to give a result. I-Jong Lin et al. focused on recognizing patterns where the data has linear ordering with the help of directed acyclic graphs (DAG). [6]

While there are hundreds of paper about handwriting recognition system with feature implementation J Pradeep et al. tried to give something different where J Pradeep et al. proposed a work without feature implementation that can recognize alphabets with more accuracy [7]. Recognizing not just only hand writing but also gender of a person is a huge deal. For this purpose, with the Man-Wo-Man program and neural network with artificial intelligence an AFIRS was developed [8].

Megha Agarwal et al. tried to improve the existing offline handwritten character accuracy method using convolutional neural networks and tensor flow. They try to improve the accuracy by more than 90%. Their motive to develop a software that takes minimal time and give high accuracy result. [9] Kacalak et al. tried to develop a new method to improve the offline reorganization system. They try to combine neural networks with the character geometric features extraction. There motive for this to reduce the dimensionality of the neural network. [10] Mijwil et al. described that there are two types of group of Handwriting recognition, they are, Affective

(a) Erotion (b) morphologyEx (c) Dialation

Fig. 1: Figure showing morphological operation

(Online) and Ineffective (Offline) methods. [11] The effective method is like an automated system. It normally electronic tablets the coordinate of the pen movement by the handwriting automatically systems. The ineffective method consists of 5 processes like Pre-amplification, Segmentation, Attribute extraction, Recognition and post-processing. R. Sethi et al. narrated that optical Character Recognition (OCR) is used for identifying the text, then printing it into machine-encoded text (the form of ASCII or Unicode). [12] For segmentation Hidden Markov model (HMM) used. KNN Algorithm and Euclidean distance formula applied to calculate nearest data from the dataset.

III. Methodology

In this section we present step by step approach of the handwritten character recognizer system. This section will also showcase our workflow and also will generally introduce to the model.

A. Data-set Description

There are several datasets of handwritten characters available. We use the MNIST dataset in the system. From the dataset we use the letters part only. Then we slit the remaining datasets in two parts for training and testing purpose. First 80% of the data for training and rest 20% for test

B. Defining model calculate the accuracy

Here we take the sequential model from Keras library to stack the layers linearly. In this model, data flow continues from one layer to another until it reaches the final layer. After that, we use the flatten function to convert the entire matrix into a single column and to the neural network to processing. Then the dense layer takes the output of the previous layer and passes them to the next layer. Using the rectified linear unit increases the non-linearity in the neural network. Input data can contain unwanted data that will remove by dropout and prevent the model from over-fitting. Many different values can be found in the input layer. Softmax function converts those values between 0 and 1 that interpret the result as the probability of each class. We need to configure the learning process before train the model, which is done by the compile method. In the compile method, we try to minimize the loss, optimize the network model using adam algorithm, and use metrics to choose the best accuracy model. After that, we calculate the accuracy before training and saw the accuracy is very low. We use Modelcheckpoint API to save our best model and use the fit function to train the best model to increase the reorganization accuracy. After training the model, we again calculate the accuracy and find out that a large margin increases the accuracy.

C. System Architecture

1) Video Capture: For motion detection, we defined the video camera through which our system will detect handwritten characters. With the python's OpenCV library, we captured motion. The motion or gesture can be captured through the internal or external camera. In our case, we used (0) index which is the web camera of our laptop. The video resolution was pretty good as well. For every frame, OpenCV will capture the video, read it, and will return the frame. The writing or drawing of characters will appear on the frame. We are converting the frame into HSV format because it is easy to work with. And to pass images or captured drawing, we used a blackboard which is initialized in zero vector and later converted into gray.

2) Object Detection: The major part of motion detection is defining the object with which motion will be created. We are using a blue color object. But to detect the object not only we had to define the color but also we had to detect the area of the object. There could be a lot of blue objects present in the background as noise. So to define the area we will use findcontours() function where we are using $RETR\_EXTERNAL$ to detect the outermost contour of the blue object. Also for findcontours() function, we are using cv2.$CHAIN\_APPROX\_SIMPLE$ to avoid detecting all the points and save some memory. But after finding the contours of the blue object we want our system to pick the highest one. Because as mentioned earlier there could be a lot of blue objects present in the background. We are taking the highest contour so that the camera can detect it properly. But even after that to reduce error if the background is clear from the color it will be nice. When any contour is found then a center is calculated from which writing will start. To calculate the center we are using the popular moment's calculation where all the pixels of the selected contour will be added [13]. The equation of the moments is:

$$m_{p,q} = \sum_{i=1}^{n} I(x,y) x^p y^q$$

Where I(x,y) is the value of image pixel at location (x,y) ,
p= x-order,
q=y-order.
From the above equation, we can find the center of the blue object by-

$$(X,Y) = (\frac{m10}{m00}, \frac{m01}{m00})$$

Where m01 and m10 are first-order moments and m00 is 0th order moment. The writing will start from the centroid and will be stored in deque as points. Later the points will connect and make a line which in conclusion will show us the complete

character. The drawing will be sharp and noise-free because we have used erosion(Fig a), dilation(Fig c), and morphology(Fig b) of morphological operation as well as we used the kernel for smoothing.

3) Blackboard Definition and Display: Then we pass the image to the models.Then the prediction instructions work where prediction takes the maximum argument of indexes. Then the display is occur using instruction imshow() of Opencv library where we pass the frame. After, all writing and prediction is done we destroy all the window by pressing ENTER which we declare by ASCII.

IV. Experimental result

Our proposed system can recognize all handwritten characters with high accuracy with robust precision. This system is user-dependent and there is no restriction on how a character can be drawn or written. Handwritten characters are consistently recognizable by this system as it predicts a character by the maximum argument of the index. The sample of handwritten characters were collected from the MNIST dataset which is a large database of handwritten digits that is commonly used for training various image processing systems. [14]

V. Conclusion

We have tried to merge some work to build a character recognition system with detecting motion. With more than 90% accuracy of selected data after training and testing the system gave us an impressive outcome.

A. Challenges

Achieving the expected outcome was not easy. Training and testing data was the first important work to do because the accuracy of data depends on it. Getting data from MNIST and processing them in a particular method to attain the best accuracy was tough. But capturing the gesture of the object and exhibit them on the screen at the same time was strenuous. We had to build a frame and blackboard separately because the frame shows the continuously generating images where the drawing is done and the blackboard displays those images which in turn gives us our expected outcome. Defining the blackboard and frame and separating them to gain the outcome was difficult.

B. Limitations

We have defined a particular color of the object in the system. Though our instruction was to detect the selected color object in descending order which means if there are more of that color in the environment the system can detect, it will take the one which has the maximum contour area. But if the object's contour area is not big enough and there are other similar color objects, the system will detect with contour area and will pick the highest contour area.

C. Future Work

Our complete work shows how we can write handwritten characters on-air and the camera capture it and then predict what we have written. But there is a lot of outcomes this system can achieve. The handwritten character recognition system itself is a big deal but recognizing the gender and age of a person alongside this system will be a game-changer. Though there are already works for recognizing age but implement this project will be great as its accuracy rate is more than 90%.

References

[1] P. D. Gader, M. A. Mohamed, and J. M. Keller, "Fusion of handwritten word classifiers," *Pattern Recognition Letters*, vol. 17, no. 6, pp. 577–584, 1996.

[2] T. Steinherz, E. Rivlin, and N. Intrator, "Offline cursive script word recognition–a survey," *International Journal on Document Analysis and Recognition*, vol. 2, no. 2-3, pp. 90–110, 1999.

[3] M.-Y. Chen, A. Kundu, and J. Zhou, "Off-line handwritten word recognition using a hidden markov model type stochastic network," *IEEE transactions on Pattern analysis and Machine Intelligence*, vol. 16, no. 5, pp. 481–496, 1994.

[4] B. Balci, D. Saadati, and D. Shiferaw, "Handwritten text recognition using deep learning," *CS231n: Convolutional Neural Networks for Visual Recognition, Stanford University, Course Project Report, Spring*, pp. 752–759, 2017.

[5] A. L. Koerich, R. Sabourin, and C. Y. Suen, "Recognition and verification of unconstrained handwritten words," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1509–1522, 2005.

[6] I.-J. Lin and S.-Y. Kung, "Coding and comparison of dag's as a novel neural structure with applications to on-line handwriting recognition," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2701–2708, 1997.

[7] J. Pradeep, E. Srinivasan, and S. Himavathi, "Neural network based handwritten character recognition system without feature extraction," in *2011 international conference on computer, communication and electrical technology (ICCCET)*. IEEE, 2011, pp. 40–44.

[8] S. Kulik, "Neural network model of artificial intelligence for handwriting recognition." *Journal of Theoretical & Applied Information Technology*, vol. 73, no. 2, 2015.

[9] V. T. P. G. Megha Agarwal, Shalika, "Handwritten character recognition using neural network and tensor flow," https://www.ijitee.org/wp-content/uploads/papers/v8i6s4/F12940486S419.pdf, 2019.

[10] W. Kacalak, "Handwriting recognition methods using artificial neural networks," 11 2006.

[11] M. Mijwil, "Handwriting recognition methods," 09 2015.

[12] R. Sethi and I. Kaushik, "Hand written digit recognition using machine learning," in *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, 2020, pp. 49–54.

[13] S. Bhowmick, S. Kumar, and A. Kumar, "Hand gesture recognition of english alphabets using artificial neural network," in *2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*. IEEE, 2015, pp. 405–410.

[14] "Mnist dataset," https://en.wikipedia.org/wiki/MNIST$_database$.