Image Based Calculator

Md Ashiqul Amin
School of Informatics, Computing and Cyber Systems
Northern Arizona University
AZ, USA
ma3359@nau.edu

Mertay Dayanc
School of Informatics, Computing and Cyber Systems
Northern Arizona University
AZ, USA
md2289@nau.edu

***Abstract*—This paper represents a new technique to calculate the mathematical equations by analyzing the handwritten equations. Additionally , this image based technique recognizes the handwritten equations from the image and provides the solution of the equation as a result. Moreover, neural networks in tensor flow is used to classify the hand written digits and operators. A new data set is created and analyzed to recognize the hand written equations . To create a new data set, a newly created hand written mathematical operator data set was merged with the MNIST data set. The accuracy of the new data set is 99.7% where the accuracy of the MNIST data set is 98.3%. At the end, we introduced an algorithm to calculate the recognized equations.**

***Index Terms*—Hand Written Equation , Calculator, Image, Neural Networks, Data Set**

I. Introduction

Mathematics is the backbone of Science. In every level of our study and research we have to solve lots of mathematical problems. Sometimes, it is difficult to solve some mathematical equations by conventional calculators or it requires a long time to solve complex equations. In recent times many stat ups provide programmable calculator to solve complex equations, but its not easy to type the complex equations in the programmable calculators. Moreover, students are reluctant to type long and complex equations. Our main vision was to make this stuffs very easy for the students and the researchers. As a result, we introduced an image based math calculator which can solve mathematical equations from the images. To make a more reliable image based calculator , we proposed a calculator which can also recognize the hand written equations.

To implement project , a new data set will be created. The new data set is the combination of MNIST Data set and our own data set. To create data set we collect images from different sources and then converted the image into 28x28 format. Then we created the 1D matrix of the pixels for the images and put them into a CSV file. After creating CSV file for every type of image, the csv files are merged to create a new and big dataset.

OpenCV is used to read the raw images of equation [6]. After reading the image , the images converted into the Numpy array. The input was converted to the Numpy array to fit the input data with the model. If the pixel values of the array matches with the values of node, the node will be activated and pass the information into the next layer. At the end of the network it represents the classification results.

In Figure 1, the algorithm of our proposed system is represented with a flow chart. At the end of classification steps , the system provided the recognized digits and operators. Then this recognized digits and operators used as a input of the calculation algorithm. After receiving the input the algorithm first try to recognize the equation pattern [1] . At the end , the algorithm provides the solution of the equation as a result.

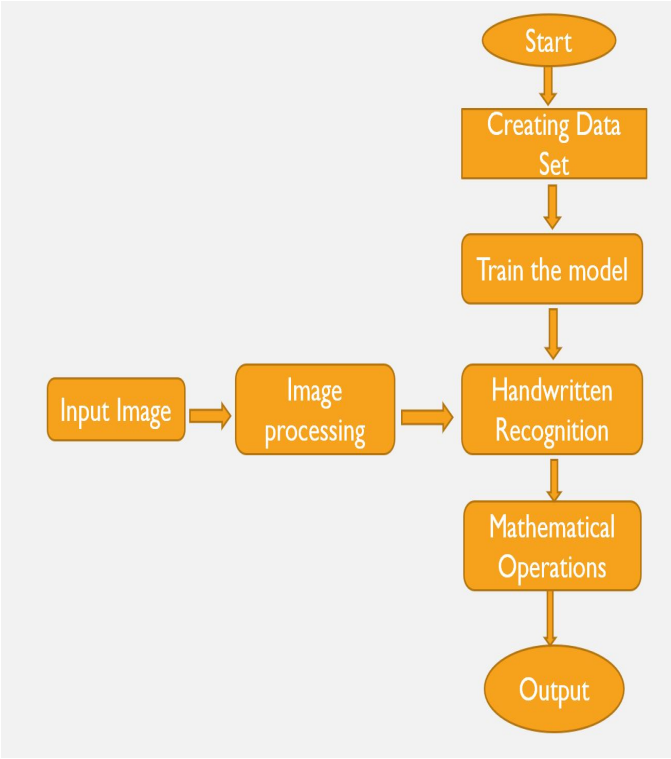

Figure 1. Algorithm

In this study, the performance of our networks and algorithm are evaluated. Moreover, the accuracy of our own data set is also measured. Overall, This application can help all types people to solve mathematical operations.

II. Data Set

To recognize the hand written equations, we need a data set of hand written digits, alphabet letters and mathematical operation. However, there are lots of hand written digit and alphabet letter data set in the online repositories, we could

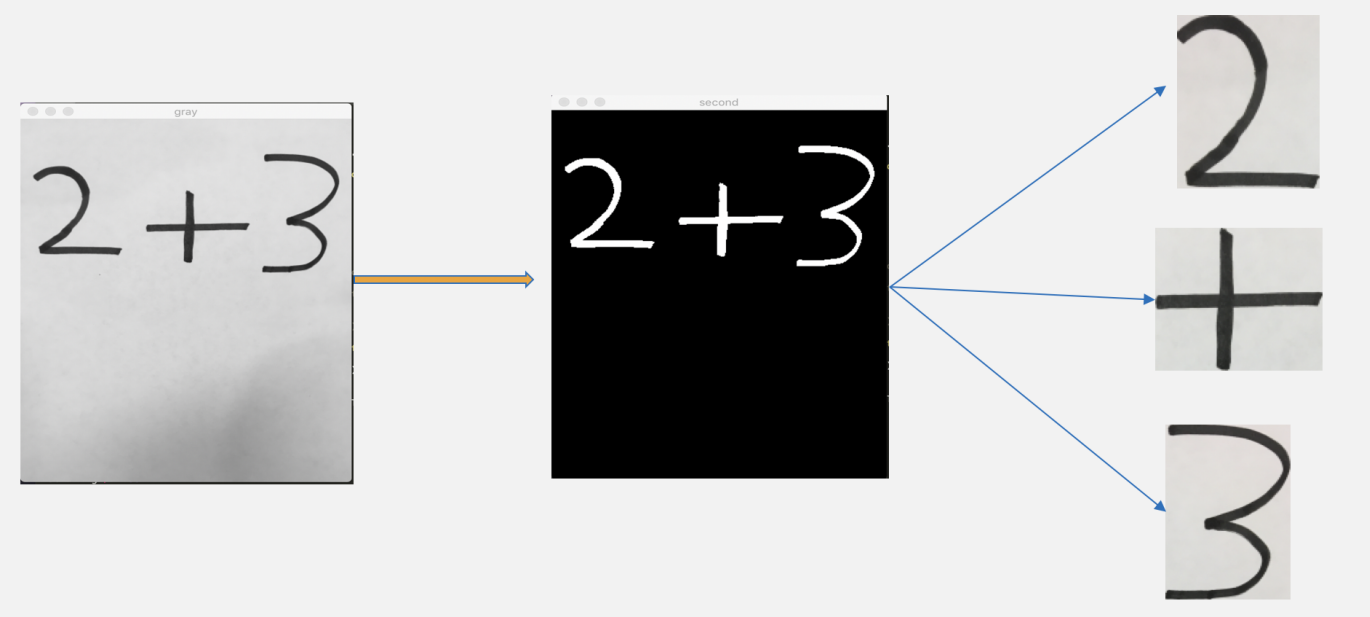

Figure 2. Image Processing

not find a unique data set which has the hand written digits,alphabet letters and operator data sets. MNIST [8] , NIST are very popular data sets in hand written recognition process. NIST database has the hand written data of digits and alphabet letters and MNIST is the subset of NIST set. As a result , these conventional data sets can not meet our requirement.

At the beginning of our project we created our own data set to meet all the requirements. To create our data set , we collect the image of all kind of mathematical operator ,digits and alphabets. The dimension of these images are 45x45. First we put the same images in a folder. After that, using openCV to read the images of that folder. Then resize the images in the dimension of 28x28 and translating the image so as to position this point at the center of the 28x28 field. After changing the dimensions , the image is converted into the gray scale. Then the images converted in to the pixel matrix. Pixel values are 0 to 255. Here 0 means background is white, 255 means foreground is black. This matrix converted into a single dimensional matrix and then put the value into a csv file. Thus we create the .csv file for all the images.

Moreover, the .csv files were labeled and merged manually. Thus , the new data set was created to analyze which fulfilled all the requirements of the application. The data set was divided into two portion, i.e, test data set, training data set. The number of data that the test data set carries is about 70000 where the test data set carries 17000 data.

III. The Image Prepossessing Stage

Image processing part of this project is written in python using openCV library.In Figure 2, the image processing technique is clearly illustrated. Raw data was taken image from user and image was processed such that it returns extracted digit matrices and operators on the picture in an array. Then extracted digits given as input to the neural network model that was trained. In order to extract the digits and operators, four steps were applied to the raw image; removing channels from raw image, applying a threshold to catch the contrast difference and binarize the image, applying dilation morphology to the binarized image with kerne value in order to enlarge the boundaries of the white foreground [5]. Binarized and dilated image was used to draw surrounding rectangles around them so that region of interests of each of them can be extracted separately, using the width, height and (x,y) coordinates of the rectangle. Later that , those region of interest(ROI) on the original image stored in an array. Then those extracted images sorted from left to right to create correct mathematical expression.

After sorting, each element in the ROI array given to the model script as an input. Before ROI's given to predicting part, they were processed more to format them to look like the data set inputs. Which has matrices with size of 28 by 28 and all the elements of the matrices was normalized. So, ROI array image matrices were resized to 28 by 28 then they were normalized by dividing by 255. Later that, since ROIs were extracted from the original image, it is on grayscale so in order to give those to prediction function, those images has to be binarized again. Then input matrices were flatten our to look like (1,784) matrices. Finally model was used to predict 1-D normalized array input. The most challenging part of the image processing was to figure out how to increase the white foreground which are the numbers and operator. When we tried to give the image without mathematical morphology, since the pixels being used is much less than the pixels being used in data set images so our model mapped all the input images to the same label, which is '+'. It took a while for us to understand the problem and we have solved the issue by using

dilation as mathematical morphology which increased white foreground. Then we ran into another issue where our model predict images wrong, but the problem was not in out model because it was predicting 99.8% of test data set correct, so we decided that the problem might in the image processing. Then we compared the data set images and the images extracted from user input. Then, I have noticed that the images extracted from user input, was touching the borders of the surrounding box of it. On the other hand, data set images are centered and has padding from the surrounding box of it. So, we put margin to the images extracted from user input and it fixed our problem. We pick openCV library for python because we think that we can find more resources than other libraries, also we didn't want to use frameworks because we couldn't understand image processing aspect of out project this deeply.

IV. Design of the Network

Neural Network is a very popular algorithm in hand writing recognition problems. In previous works, many researcher used multi-layer neural network in LeNet-5 [2]. More over, in some cases, the popular Backpropagation algorithm was used for training the network [3]. However, in most of the cases they suffered from the ineffectiveness of computing power at that time. To solve that issues some researchers proposed Alexnet which is a classic convolutional neural network [4]. This method improves the difficulties of previous classification tasks .

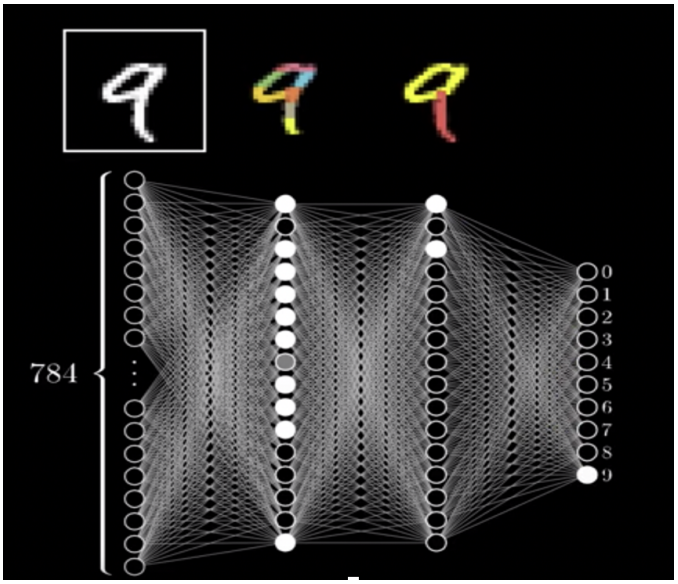

Figure 3. Neural Network

In this project, neural network with single hidden layer is used for classifying the image. We built a data set that contains 70000 images of 28X28 pixels. We trained the neural network with our own data set. In the input and hidden layer Relu activation function is used and in the output layer softmax function is used in the neural network. We used Tensorflow for the implementation of the network [7]. In Figure 3, digit 9 is input in the input layer of the neural network. Then it is processed in the hidden layers of the network. Finally, in the output layer it can recognize the digit 9. The hidden layers are trained in the training period. In our model , we input the Numpy array of the image instead of the raw image like Figure 3. Then this array values tried to match with the values of the nodes. If they matches the nodes are activated and pass the information into the next layer.

V. Calculation Algorithm

After model predicted the user input, predictions was stored in the queue so that, expressions created in the correct order as they were written in the input image. Elements in the queue was taken until it' reaches an operator then after operator detected, again elements taken until it reaches the end. Then operator and the digits combined to get the result.

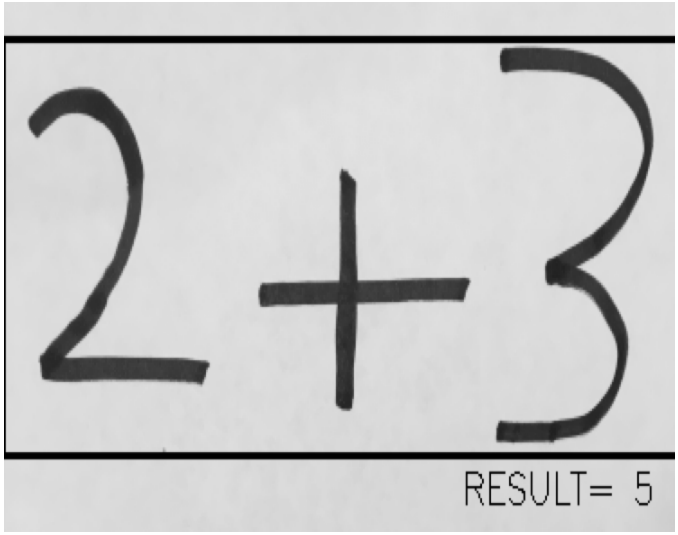

Figure 4. Output

After result obtained, original image was processed such that, surrounding box was drawn in the region of interest which is the expression not the digits separately and below the surrounding box, result was displayed. In order to draw a surrounding box over the expressions, kernel value increased on the x axis then original image was dilated again with the new kernel and expressions on image start intersecting each other so that it becomes like one character on the image. This helped to draw surrounding box over the expression. In Figure 4 , shows the out put of our application. Here the equation represented the addition of two and three. We all know that the sum of two and three is five. As a result , our application should provide the result five to prove its reliability. In the Figure 4, the result is five. It ensures that our application works properly.

VI. Challenges and Future work

The application faced some challenges in calculating equations. At the initial condition , now it can only solve single

digit equation. When we put an equation having double digit term , the system can not recognize the digits. Now this application can sole only addition and subtraction operations. Another challenge is our system can not recognize overlap digits. Moreover, the present data set contains only twelve types of data such as 0-9 and '+', '-' .

In future , we will introduce a complete data set. Add more features in our application. Our future application can solve integration and differentiation problems. Our image processing technique will be improved. The system can recognize any overlapped digits. Overall , this application can solve most of the mathematical problems.

VII. Conclusion

Image based calculator will provide a privilege to solve any kind of mathematical equation easily. This application bring a change in the field of modern day calculator. Moreover, it reduces the calculation time of equations and creates interest in students mind to mathematical equations. Beside, the accuracy of our new data set is very good. Our study also represents how a neural network works for the classifications. Overall , this study represents a novel technique to recognize the hand written equations and provides the result of the equation as a solution.

Acknowledgment

The Authors want to thank Dr. Razi for his valuable instruction and motivation for the projects. Dr. Razi instructed the authors that how they can solve the problems and select methods to complete the project.

References

[1] Matei O., Pop P.C., Vălean H. "A Robust Approach to Digit Recognition in Noisy Environments." In: Jiang H., Ding W., Ali M., Wu X. (eds) Advanced Research in Applied Artificial Intelligence. IEA/AIE 2012. Lecture Notes in Computer Science, vol 7345. Springer, Berlin, Heidelberg.

[2] Cun Y L, Boser B, Denker J S, et al. "Handwritten digit recognition with a back-propagation network[C]// Advances in Neural Information Processing Systems". Morgan Kaufmann Publishers Inc. 1990:465.

[3] Hecht-Nielsen R." Theory of the backpropagation neural network"Neural networks for perception (Vol. 2).

[4] Krizhevsky A, Sutskever I, Hinton G E. "ImageNet Classification with Deep Convolutional Neural Networks[J]". Advances in Neural Information Processing Systems, 2012, 25(2):2012.

[5] Morphology. https://homepages.inf.ed.ac.uk/rbf/HIPR2/dilate.htm. December,2018

[6] CVisionDemy. https://cvisiondemy.com/extract-roi-from-image-with-python-and-opencv/. December, 2018

[7] TensorFlow.$https://www.tensorflow.org/tutorials/keras/basic_classification.$

[8] Yann LeCun, Corinna Cortes, Christopher J.C. Burges. "THE MNIST DATABASE of handwritten digits," http://yann.lecun.com/exdb/mnist/ .

[9] Handwritten Multi-digit String Segmentation and Recognition using Deep Learning, http://www.emaraic.com/blog/multi-digit-segmentation-and-recognition

[10] Md Jubaer Hossain Pantho , Festus Hategekimana , Christophe Bobda "A System on FPGA for Fast Handwritten Digit Recognition in Embedded Smart Cameras" 11th International Conference on Distributed Smart Cameras Stanford, CA, USA — September 05 - 07, 2017 Pages 35-40