SOLVING SIMPLE EXPRESSIONS WITH APPLIED IMAGE PROCESSING

Praphull Dass

B.Tech. in Computer Science Engineering, Indraprastha Institute of Information Technology, Delhi

ABSTRACT

Mathematical expressions can be tiring to solve, especially when you are a beginner and are not very good with the subject. In this age of digitalization, where everyday work is also being done with the help of computers, we offer to complete this task too with the help of computers. Our project can be used to solve simple mathematical expression. Users can use a reasonably clear image of the expression on white background to obtain the answers. They can also view the image at every stage of the process using an image gallery. The solution is obtained by extracting the required region, segmenting each individual entity and predicting each element before finally evaluating the complete expression.

Key Terms— image transformation, *o*ptical character recognition, convolutional neural network, thresholding, tkinter

1. INTRODUCTION

Children learn to form mathematical expressions at a very young age and continue to use them for a long time. In the beginning children do not wish to put in much efforts and want to do work fast. Parents have to oversee young ones to see if they are obtaining the correct answers. Our program can help these children to check themselves without any overseer and have a speedy study sessions.

This kind of work lies under the optical character recognition category in the field. This field revolves around identifying different type of text from images of varied background. The work we do is based on focused text recognition in a specific scenario. Convolutional neural networks are known to be the state-on-the-art models for image classifications and are widely used in optical character recognition models. We implement our own convolutional neural network and try to build a reliable character recognition model along with good preprocessing of the data..

2. METHODOLOGY

The complete process for obtaining the correct answer for the expression can be divided into three parts: image processing, text detection and evaluation of expression. But, before the process initiates, we have to clear a certain prerequisites for the process. The expression must be written in dark and have a relatively light background. We have assumed that the expression is single lined and does not contain any other text in the image. A total of sixteen characters [0-9][+,-,*,/,),(] are only acceptable in the expression. The expression assumes each syllable to be written clearly without any overlapping (no cursive handwriting is allowed). If the image is visually clear and has cleared the pre-requisites, then it can be used to obtain the results.

2.1. Image Processing

The image entered by the user is discarded and instead the negative of the image is used (see Fig 2). It is done to have a background of smaller numbers and the main text having the major values, which is not the case with a white background and dark text because white is represented by 255 and black is represented as 0.

Thresholding is performed on the obtained negative image to clear out some noise and make the region of interest clearly distinguishable. The optimal value for thresholding is obtained with the help of histogram of the image. It is chosen such that it divides the histogram in a way in which one side of histogram has a hundred times more points that the other side. Since we are using the negative of the image, we obtain values which are closer to 255 compared to 0 (see Fig 3).

Since thresholding has removed most of our noise, the obtained image can now be used to find region of interest. We use an iterative procedure to find the region such that it has some useful values and cannot be discarded. This allows us to crop out the expression and we pad it make clear boundaries (see Fig 4).

Now a combination of sliding window technique and largest connected component is used to segment [2] out each

syllable from the expression. We iteratively move from left to right in an image and find the largest connected component and its boundaries. These boundaries are use to crop out these syllables (see Fig 5).

Lastly these cropped out syllables are resized and reshaped to match the requirements of our convolutional neural network. This steps marks the end to image processing in this complete process.

2.2. Text Detection

This process will allow us to find what the syllable in the cropped image means and return it in string format. Text detection is done using a three layered convolutional neural network [1][3]. From over a 1,00,000 images [5] of selected syllables, only 800 images of each syllable are used to training the neural network.

Each image in the dataset is turned to negative and are resized to desired size. These resized images are dilated with a 3x3 kernel to make the text on image spread out. Thresholding is performed on these images to make them match the images obtained from image processing of the expression image. The label attribute is turned categorical.

A sequential model with 3 convolutional layers and 2 dense layers is build using keras. Each convolutional layer is followed by relu activation, max pooling. A dropout layer is inserted between all major layers i.e. the convolutional layers and the dense layers.

The training data is split into training and validation data to test the accuracy of our model. The model is trained with adam optimizer and accuracy as metric. The model is run with a batch size of 100 for 20 epochs. We obtain a maximum accuracy of 98.44% and an end accuracy of 97.77% from our model. This model will help us predict the expression.

2.3. Evaluation of Expression

After using the convolutional neural network model with the segmented syllables and predicting what each image mean, we obtain a string expression which represents the expression enteres by the user in form of image. This string form expression needs to be segmented and an implementation of stack is required to decode this expression and obtain the final answer. However, we need not do it because there exist a very efficient algorithm implemented as “eval()” in python for this sole purpose. The images at each stage is also shown in form of an image gallery [4] to check what the program is doing.

3. RESULT

We were able o build a convolutional neural network which can predict hand written numbers with an accuracy of 97.77%. We were successful in building a program which can give the result of a handwritten mathematical expression from an image. This can be tested with the below example.

Below are the images shown as each stage of the process for an input image.

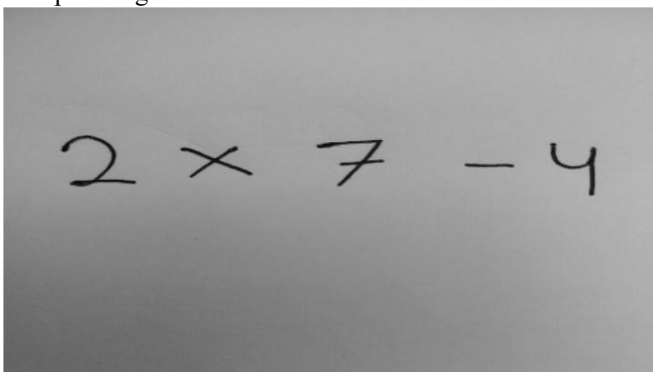

Fig 1 : Original Image

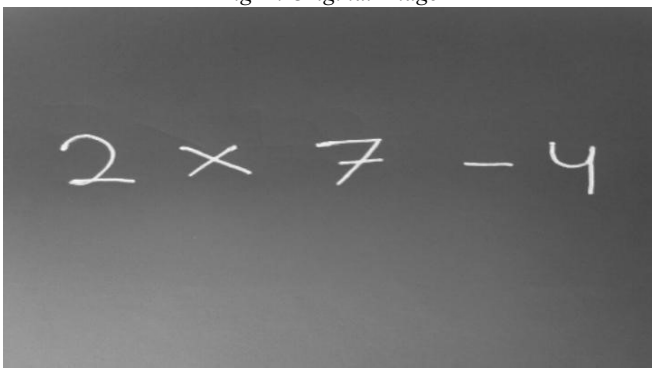

Fig 2 : Negative of original image

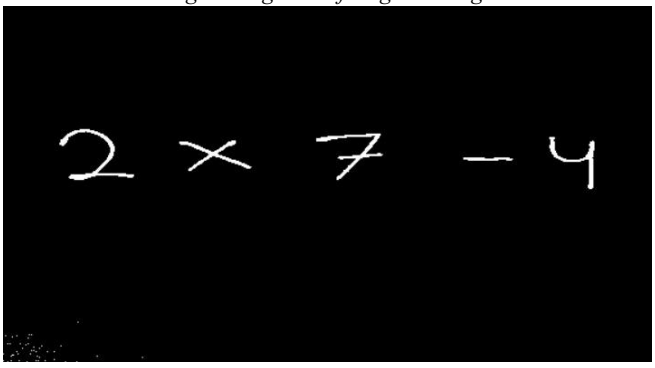

Fig 3 : Image after performing thresholding

Fig 4 : Cropped region of interest

Fig 5 : An example of segmented syllable from the complete region of interest

```
Your expression is : 2*7-4
Your answer is : 10
```

Fig 6 : Snipshot of printed final answer

4. CONCLUSION

We were able to successfully build a useful model. This model can be considered good but not the best. The convolutional neural network used in this project has an accuracy of 97.77% which can be increased if we obtain a more varied hand written dataset and finely tuning the hyper parameters of the convolutional neural network model. We can also try to identify the characters using and overlapping strategy. We can overlap the segmented syllable with all possible syllable images and chose the one which gives the least mean squared error.

5. REFERENCES

[1] T. Kobchaisawat and T. H. Chalidabhongse, "Thai text localization in natural scene images using Convolutional Neural Network," Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific, Siem Reap, 2014, pp. 1-7, doi: 10.1109/APSIPA.2014.7041775.

[2] S. Tingre and D. Mukhopadhyay, "An approach for segmentation of characters in CAPTCHA," International Conference on Computing, Communication & Automation, Noida, 2015, pp. 1054-1059, doi: 10.1109/CCAA.2015.7148562.

[3] S. Shreya, Y. Upadhyay, M. Manchanda, R. Vohra and G. D. Singh, "Optical Character Recognition using Convolutional Neural Network," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2019, pp. 5559.

[4] https://www.youtube.com/watch?v=zg4c92pNFeo&list=PLCC34OHNcOtoC6GglhF3ncJ5rLwQrLGnV&index=9 [5] https://www.kaggle.com/xainano/handwrittenmathsymb ols