**Handwritten Character Recognition Using Convolutional Neural Network(CNN)**

Gnana Mohan Reddy Narapareddy, Guru Sai Madupuri, Mohammed Johny Shaik

**Abstract**

In this project, we designed OCR based model to detect and count the number of alphabets and numerical values to choose the winner of the game called Play Dots and Boxes. We used the modern deep learning and image processing (OpenCV) techniques to get our required result. Firstly, We trained the model to using Keras and TensorFlow to recognize alphanumeric characters (i.e., the digits 0-9 and the letters A-Z). Then we give the pre-processed image of game as input to the pre trained model, which outputs the winner of the game. Our main purpose was understanding of how deep learning is applied to the classification of handwriting, documents and more specifically automating the decision of deciding the game winner. This can be very useful for future, where our model can be used to decide the winner for Play Dots and Boxes game.

**Keywords:** OpenCV, OCR, Keras

**Introduction**

In this project, we delve into the depth of Optical Character Recognition and its application areas. We made a application in Python that will help us detect characters from images of game called Dots and Boxes. Then Using this result, we count the number of characters to decide the winner of that game. Here are the main points of our project:

* We made Computers understand the contents of image simply by looking. Our main focus is Play Dots and Boxes game, where the computer will decide from the end resultant image that which result has won by counting the characters.
* We hope to make a system which can be applied on Play Dots and Boxes game to get the winner of the game simply by passing the image to computer
* No need for human to count, its all by automation and all by itself
* There are many OCR models for hand writing or vehicle player recognition but none of them are made for this game.
* Previous results have not been so accurate like our model

The key contributions of this work are:

*Names1: Gathering dataset from online sources (EDIT THE NAME)*

*Names2: Training the model*

*Names3: OpenCV operations and output model*

**Background and Literature:**

Humans can understand the contents of an image simply by looking. We perceive the text on the image as text and can read it. Computers don't work the same way. They need something more concrete, organized in a way, they can understand.

This is where Optical Character Recognition (OCR) kicks in. Whether it's recognition of car plates from a camera, or hand-written documents or game results that should be converted into a digital copy, this technique is very useful. While it's not always perfect, it's very convenient and makes it a lot easier and faster for some people to do their jobs.

There are many OCR models created till date. Examples include vehicle plate reading, documents scanning and many more. But there are none created that will help in deciding the winner of the game such as Dots and Boxes. So, for this purpose, we proposed a solution and implemented it and our findings are good and accurate. The famous data used to train the model is MNIST 0-9 and Kaggle A-Z dataset. The data contains thousands of examples of characters and integers. The Kaggle A-Z dataset was made by Sachin Patel which is based on NIST Special Database 19 .



The framework we used for our model Include Tensforflow, Keras and famous OpenCV libraries from python.

**Approach / Methods:**

The goal of our project to obtain a deeper understanding of how deep learning is applied to the classification of handwriting, and more specifically, our goal is to:

* Trying out the best readily available handwriting datasets for both digits and characters
* Understanding how to train deep learning model to recognize digits and characters for Dots and Boxes Game
* Understand some of the challenges with real-world noisy data, game data and how we can use OpenCV operations to improve our results

**Project Workflow:**

* Decided the Steps required to implement and train a custom OCR model with Keras and TensorFlow.
* Examined the handwriting datasets that we used to train our model
* Implemented a couple of helper/utility functions that aided us in loading our handwriting datasets from disk and then preprocessing them.
* After training, the results of our OCR work.

**Datasets:**

We used two datasets for our OCR training with Keras and TensorFlow.

* The standard MNIST 0-9 dataset.
* Kaggle A-Z dataset from Sachin Patel

**Implementation**

We implemented the following methods and utilities:

* Loaded both the datasets for MNIST 0-9 digits and Kaggle A-Z letters from disk
* Combined these datasets together into a single, unified character dataset
* Handled class label skew/imbalance from having a different number of samples per character
* Successfully trained a Keras and TensorFlow model on the combined dataset
* Plot the results of the training and visualize the output of the validation data

**Libraries Used:**

* ResNet Model (Main model used for training)
* Keras (Tensorflow backend)
* Sklearn (for evaluation purposes)
* Matplotlib (Graphs)
* Numpy (Numerical Operations)
* OpenCV (Image Processing Operations)

We used ResNet model along with SGD Optimizer to train our model with the above mentioned dataset because this model works well with these datasets. After we get the pre-trained model. We pass the input image to model for deciding the winner of the game. Before passing the image, to make it visible good we perfrom OpenCV image processing operations:

* RGB TO Grayscale
* Sharpness
* Thresholding
* Applying morophology operations
* Contours and Edge detection

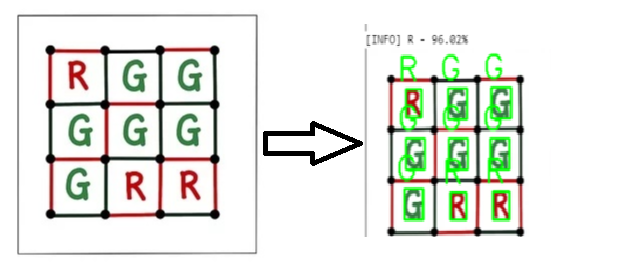
**Results**

**Training and Validation Loss graph:**

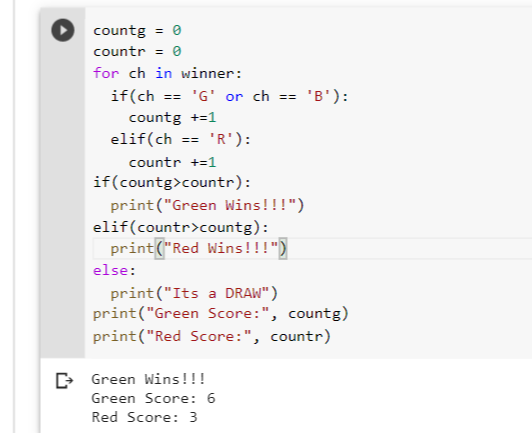


The plot shows our training history. It shows very little signs of overfitting, implying that our Keras and TensorFlow model is performing well on our OCR task.

**Game Output:**



**Deciding the Winner:**



**Conclusion**

We have successfully managed to accomplish our task, and decide the winner of the Dots and Boxes game. The Results can be further improved with making a generalized type for all types of interfaces which are used for this game.

Acknowledgments

Acknowledge everyone who has given you help and any relevant grants…

**References**

<http://yann.lecun.com/exdb/mnist/>

[**https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format**](https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format)

https://en.wikipedia.org/wiki/Optical\_character\_recognition