PalArch's Journal of Archaeology of Egypt / Egyptology

A Hand Written Digit Recognition Based Learning Android Application

Swarna Prabha Jena1[1], Debraj Rana[2], Subrat Kumar Pradhan[3]

[1,2,3] Centurion University of Technology & Management, Odisha, India

Email: [1] swarnaprabha@cutma.c.in

Swarna Prabha Jena1, Debraj Rana, Subrat Kumar Pradhan: A Hand Written Digit Recognition Based Learning Android Application -- Palarch's Journal Of Archaeology Of Egypt/Egyptology 17(9). ISSN 1567-214x

Keywords: Convolutional Neural Network, Tensor Flow Lite, Android, Digit classifier.

ABSTRACT

The Hand Written Digit Recognition is an interesting topic which comes under classification model, in current date a lot of work has been done using this technique .This paper works on the same model and technique but directed towards a learning application for Early Childhood. Small kids age between (3-5 years) approximately use to start learning the digits and classify the difference in this period of time .So this work come up with an android application which runs a Hand Written Digit Recognition model to teach those small kids in an effective and interesting manner. The proposed work involved open-source software library Tensor flow to implement Neural Network for the classification model and the use of Android studio for development of Android application for kids use.

1. Introduction

In 21st century all age groups from toddlerhood to late adulthood have very much affection towards smartphones ,In a very practical example , It has seen kids(Early Childhood) crying out very-loudly, but the scene takes a sudden change when make them held with a smartphone with cartoon Movie or game(say Talking Tom) playing on it ,So it puts a clear conclusion that how much affection kids(Early Childhood) have towards the smartphones or mobile devices . During that age group it always attracts the kids toward graphical and interactive product. So the proposed work focused this age group (Early Childhood, Age=3-5 years) to make them

learn the very basics of Educational Learning using an android application. The very idea or learning mode will keep their energy and interest both high and they can catch the thing easily, which can lead them to their next learning phase much quicker with lots of curiosity.

Machine learning the two words itself says a lot of things in this digit world. It provide algorithm to make the machine learn for decision making. Basically it every machine learning technique has two phases: one is learning phase which deal with train the model using some training data and the second phase is the testing phase which predict the output based on the learning experience on a test known data set or unknown data set. So far there are numerous number of machines learning algorithm has been developed like Naïve Bayes, Decision Tree, etc. But for the proposed work convolutional neural network (CNN) provides the maximum accuracy for the hand written digit recognition.

Tenser flow is an open source platform used in machine learning developed by Google. It has its own tools and library and community resources that very much effective in building Machine learning real time application. It is very much suitable for GPU and CPU including mobile and embedded platform.

2. Literature Review

This portion deals with the work that has been done in the field of Hand Written Digit Recognition in the favor of mankind. Liu, C.L et.al.[1] presented the updated results of handwritten digit recognition on databases which includes images using feature extraction and classification techniques. They used CENPARMI, CEDAR & MNIST databases here. By testing dataset of each database, they got 56 recognition accuracies by combining 7 classifiers with 8 feature vectors. Ma, C. and Zhang, H used the multi-feature extraction and deep learning analysis which is not a listed in the traditional way of digit recognition and approached for the extraction, structuration, distribution and projection of images which are already processed. Images in different writing styles can also be recognized by their proposed system. MNIST database has been used for validation, effectiveness and superiority of their task [2]. In this paper [3] the main focus of the author is on various Neural Network (NN) approaches like deep belief network (DBN) the deep neural network (DNN), and convolutional neural network (CNN) which are also included in most used neural network approach. After applying all these neural network approaches it was concluded that the DNN is the best among all in terms of performance and accuracy. According to their paper, the accuracy of DNN is nearly about 98.09% and the error percentage in every system has a similar rate of error which is between 1-2% and it is because of the similarity in the shape of the digits. This time the author [4] increased the number of layers of the system and they found that it is directly proportional to the accuracy and the

computation time. Here they have used the MNIST database for training and testing purposes. They also suggested that the best possible method for handwritten digit recognition is NN(Neural Network) having 6-layers which have the least error rate of 0.35%. According to Kh- Tohidul Islam et.al [5] and their paper, they have achieved an accuracy of 99.6% by taking the help of digital image pixels as features-vector and Artificial Neural Network as their classification technique for handwritten digits recognition and implemented a multi-layer neural network connected with each other and with a hidden layer for digits recognition. The paper also includes the use of the MNIST database. From the database, they used 28,000 and 14,000 digital images for training and testing respectively. The most interesting fact about their paper is that it was combined of both ensemble classifier with feature extraction using the hybrid mechanism.

The authors of the paper [6] applied Cyclone IVE FPGA (Field Programmable Gate Array) to an MLP (Multi-Layer Perceptron) using the most commonly used database MNIST (Modified National Institute of Standards and Technology). They found that the hardware design based on FPGA used here offered a high-performance, low power alternative to traditional software methods for the learning network. They found that the performance in the 8-bit system and the 32-bit system is quite similar. Teow, M.Y [7] explained the layering approach using minimal convolutional neural networks in such a way that a beginner can also get the idea of how mathematical operation has been used to classify an image. Here MNIST dataset is taken for experiment. As a result, it has given a simplified and comprehensive explanation of the mathematical structure used in digit classification or recognition. At last, it also describes how CNN learns the features of an inputted image by detail analysis and discussion. Garg, N.K et.al [8] uses a small, simple, and new feature which is extraction using pixel counting technique and contour following techniques. Here binarization and thinning are applied on the data no pre-proccing of the image is required in this feature. It is based on two primary steps. Firstly, it explains to develop the general software using the help of features like slant invariant and size invariant, which does not include any pre-processing steps. In the second step, it proves that the performance of the pixel counting technique is much more useful when compared to the former contour following technique. Here classification is done using the Tree classifier and SVM method.

In this system the proposed by the paut, A et.al [9], a method to combine two different neural networks those which are used earlier individually for digit classification and prepare a whole new model has been implemented. At first, they choose to send every information to the first network, and then from their onwards they have only the local information to the second network as like the human brain structure. The first neural network uses images of various digits than in the second neural network uses digits as contours. They have developed a strategy to describe how to combine the

different neural networks. As a result, they got to know that using two neural at the same time gives a better performance than using a single neural network. The author in [10] whole new method for handwritten digit recognition. The method is like integrating the decision that is found when two Multi-Layer Perceptron (MLP) artificial neural network classifiers operate on two different feature types. Here the two features are defining pseudo-Zernike moments of an image and extraction from the shadow code of the newly defined projection mask. For combining the two task an MLP network is deployed. For the performance results, the system is tested with samples of nearly 15000 and the advantage was described. Here it was found that after combining the techniques the result is comparatively better than others. In this article [11] have proposed a system in which the first step is to the extraction of the local features from the pattern is done by using neural network, and then the fuzzy logic is used to the recognition of the digits from the feature maps. It was found that the purposed system is unaffected from distortion and shift variation. Here they also have future expansion in the field of VLSI that can also be done. K Labusch, et.al [12] have proposed for using a sparse-coding strategy and a local maximum operation as a method of feature extraction for digit recognition. For a highly competitive digit-recognition they also used a state-of-the-art classification technique. To learn the basis for representing patches of handwritten digit images they deployed the unsupervised sparse net algorithm then used its output to extract local coefficients. In a second step, to implement local shift invariance they applied a local maximum operation. Finally, they obtained a state-of-the-art classification performance in the digit recognition task defined by the MNIST trained by a support vector machine (SVM) on the resulting feature vectors. After comparing their work with other classification performances obtained using principal component analysis (PCA), Gabor wavelets and sparse coding, they came to the conclusion that combination of local maximum operation and sparce representation for feature extraction and learning purpose gives the optimized result in case of hand written digit recognition.

3. Proposed Methodology

Designing a hand written Digit Recognition model was a challenging job, where there are various algorithm options (Naïve Bayes, Decision Tree, ANN etc) are available but CNN gives an accuracy more accuracy as compared to other method. So, the proposed work based on CNN based digit classifier and recognizer. For the implementation of the CNN model the Python platform has been used The CNN model implemented using the open source library tool Tensor Flow. This library is very much accurate and faster as compared to other available library for Machine learning. The CNN implemented here uses input layer with 784 nodes with one hidden layer and 10 number of output class

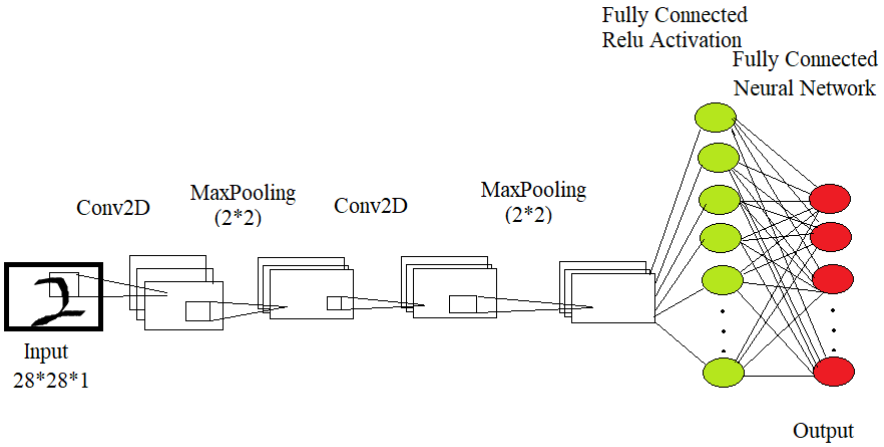

Figure. 1. *Block Diagram of CNN*

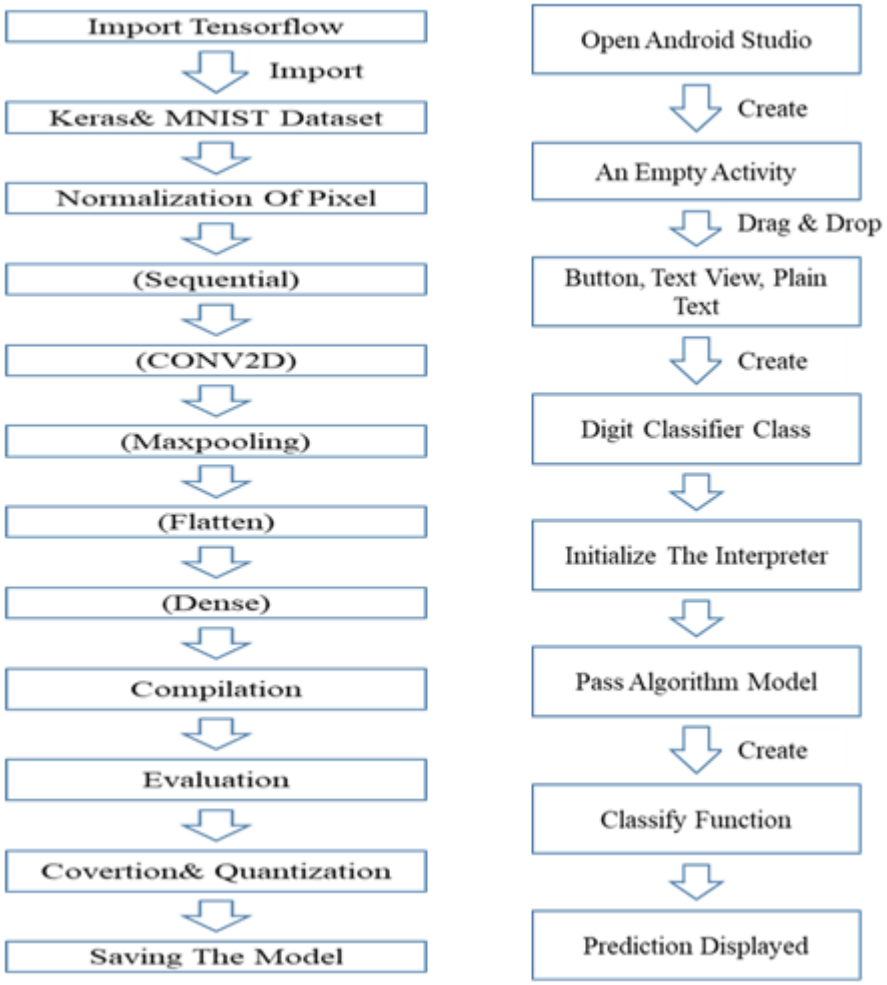

Figure. 2. *Flow diagram (a) Tensor Flow based CNN implementation (b) Android app development with Android Studio*

The implementation steps are given below to develop a CNN model using Tensor Flow.
Step 1: Import Tensor Flow library
Step 2: From Tensor Flow import keras dataset (MNIST) Dataset containing 70,000 images (60,000 training purpose and 10,000 testing purpose).
Step 3: Normalize the pixels into high/low (0/1) by dividing the pixel values by 255.
Step 4: Define the model architecture (SEQUENTIAL, CONV2D, MAXPOOLING, FLATTEN, DENSE).
Step 5: Compile the model.
Step 6: Evaluate the model.
Step 7: Convert the model in TF lite format, further apply quantization to further shrink the size.
Step 8: Now save the model with. tflite extension.
Tensor Flow provides a lite extension where the model is so simplified and shrinked in to a size that it can run in low computational preferences like microcontroller or mobile devices. This work implemented on android mobile phone which need to develop and android app of the working model of digit classifier. For generation of .apk file for the model, the lite version of TensorFlow as well as Android Studio has been used. The flowing steps provide the flow of the android app development process.
Step 1 - Open Android Studio.
Step 2 - create an empty activity.
Step 3 - Drag and drop, button, text view, plain text and a canvas window.
Step 4 - Create a Digit Classifier Class.
Step 5 - Initialize the Interpreter and pass the .tflite algorithm model.
Step 6 - Create a classify function which runs, when we draw anything on the canvas window.
Step 7 - The Prediction is displayed in plain text area.

4. Simulation Result and Discussion

Before The Tensor Flow library has been installed and the model has been developed in python 3. The developed model has under gone training and testing phase with MNIST dataset. The MNIST database was constructed from NIST's Special Database 3 and Special Database 1 which contain binary images of handwritten digits. The MNIST database consists of 60000 examples of training sets and 10000 examples of test set. The digits have been normalized to a fixed size. The models has been run and simulated with the training data followed by the testing phase which give a result accuracy of 98%. The Tensor Flow Light is a set of tools which helps the model to run on mobile and embedded devices. The developed model simulated using mobile phone and it has converted to an android app using the Android Studio. Basically, Tensor Flow light become an interface to run

the model in mobile and converted to a format suitable for android app development. Then by using the android app and following the above-mentioned step the app has been developed.
The android app (.apk file) has been generated and installed in an android mobile phone. As this app is not available in the Google play store and it has been generated through third party, so the android mobile has to be enabled for install from Unknown Sources method for the installation of third party app which is shown in Figure 3.

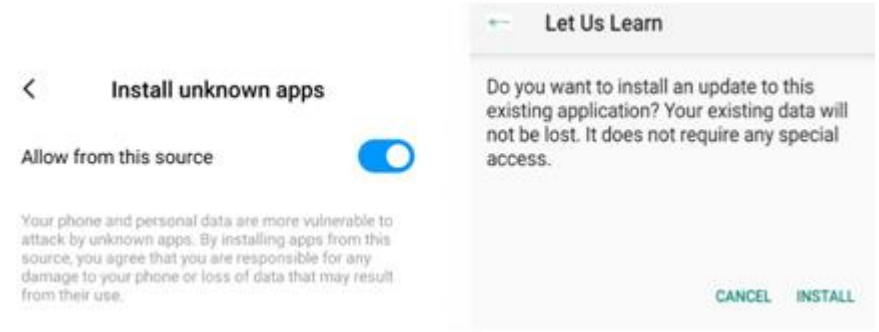

Figure. 3. *App permission for 3rd party installation*
The figure 4 shown below the installation and home page of the developed app known as Let Us Learn.

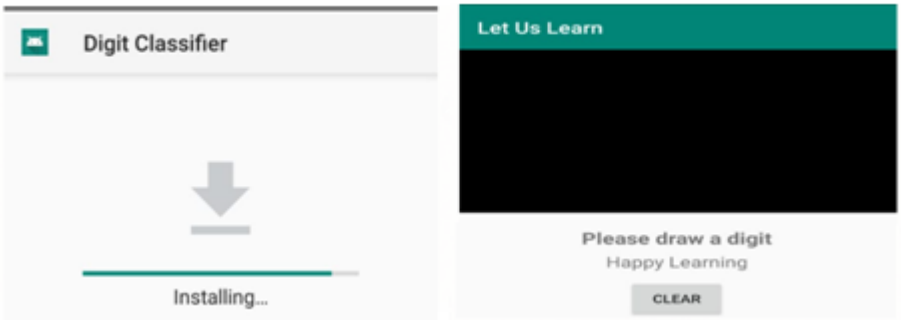

Figure. 4. *Installation and Home screen*

After proper installation with specific permission, automatically an icon will be shown in the home page. The app has been open and tested with hand written digits starting from 0 to 9. The entire handwritten pattern has been classified properly with their respective labels. The figure below shows simulation results of the handwritten digit classification and recognition for the children. It's a very simple to use App for the children.

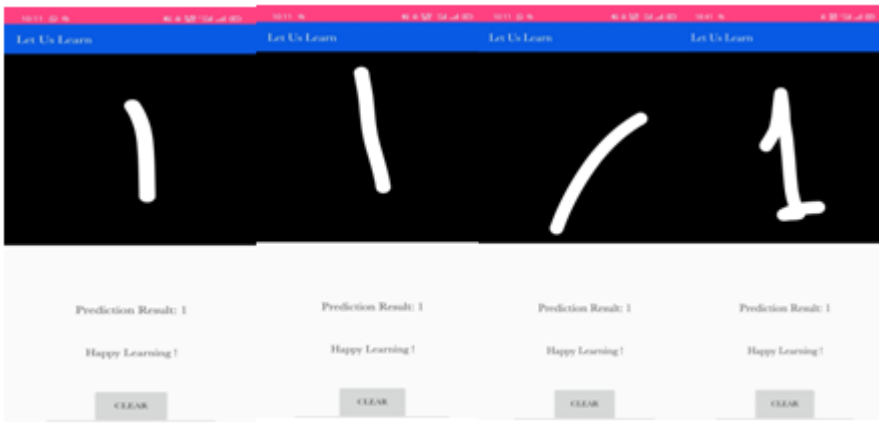

Figure. 5. *Different patterns of 1 drawn in the App*

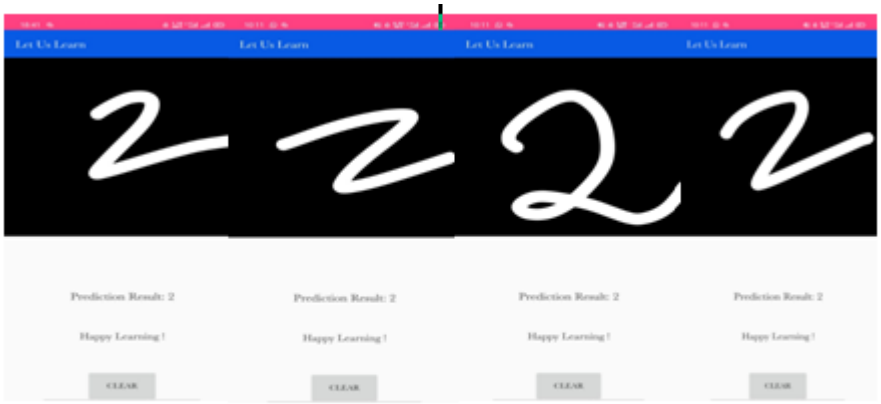

Figure. 6. *Different pattern of 2 drawn in the App*

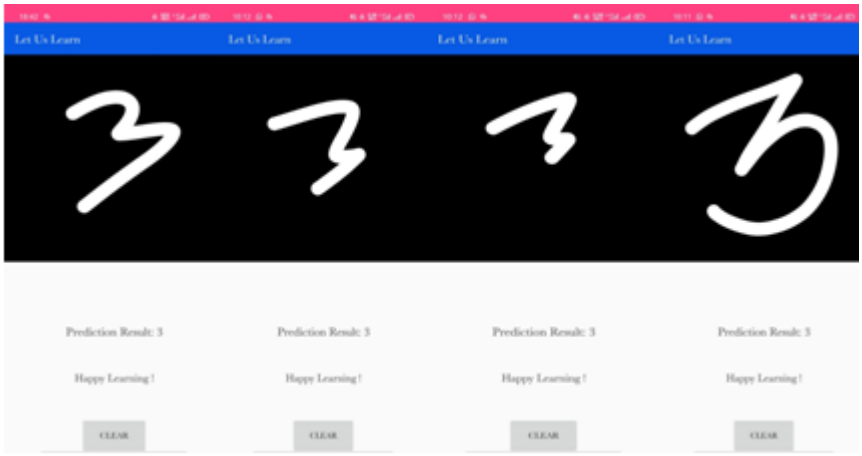

Figure. 7. *Different patterns of 3 drawn in the*

Figure. 8. *Different patterns of 4 drawn on the App*

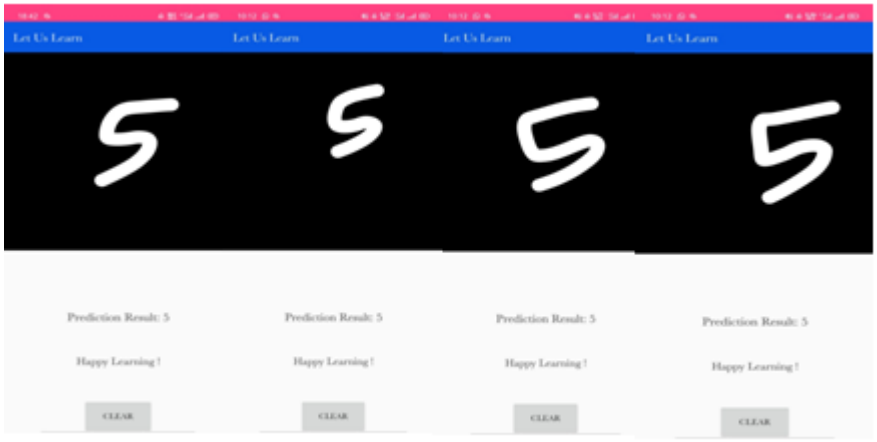

Figure. 9. *Different patterns of 5 drawn on the App*

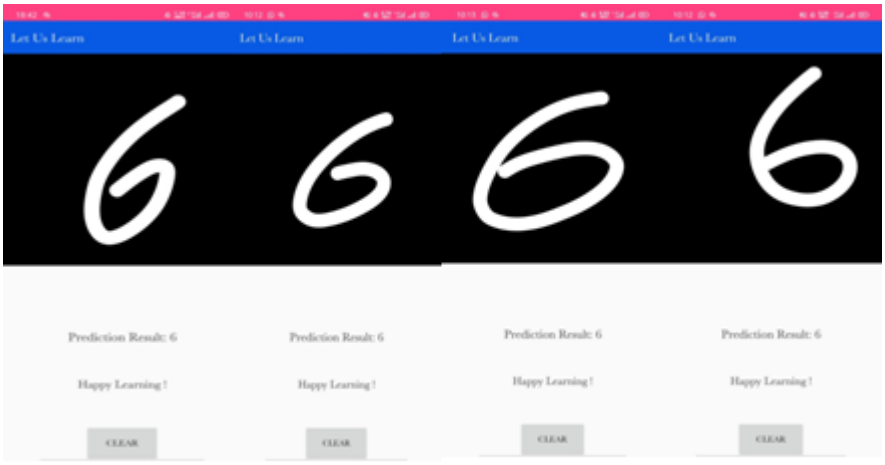

Figure. 10. *Different patterns of 6 drawn on the App*

Figure 11. *Different patterns of 7 drawn on the App*

Figure. 12. *Different patterns of 8 drawn on the App*

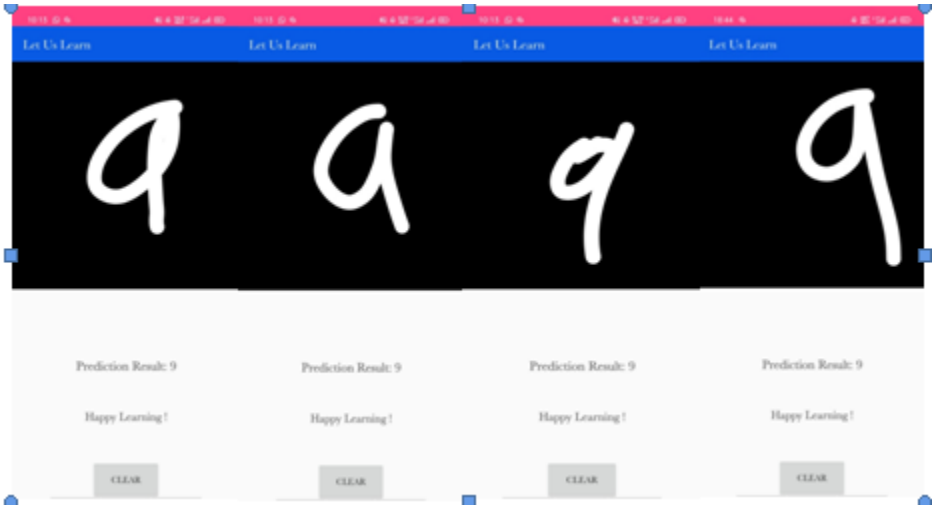

Figure. 13. *Different patterns of 9 drawn on the App*

Every time new pattern can be drawn after clearing the screen, so for that a clear button has been provided to clear the screen area. The result will be displayed below the black screen area. The prediction result based on the learning of the developed system and the learning process done with the training the model with some data set of pattern. It means classification is done based on the pattern. In the digit from 0-9 there are some part or pattern which can be similar with some other digit like 4 and 9, 1 and 7 etc. So in some cases it may give the false result. The sample image is shown in the below figure No. 13.

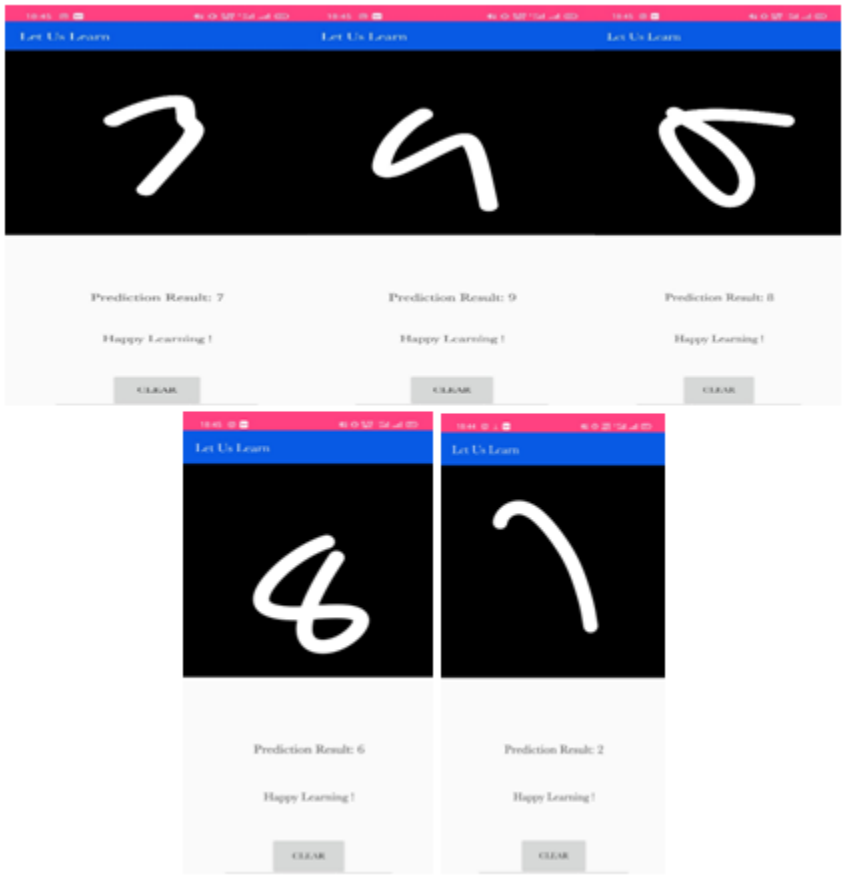

Figure. 14. *Wrong Recognition of digit from 0-9*

5. Conclusion

Hand-written digit recognition topic can have lot many implementations that can change the view/purpose of using mobile devices. Lot many implementations are still waiting for the innovation and modification. So, this work developed an android app based hand written digit classifier for small kid learning purpose. The algorithm used behind is the Tensor Flow based machine learning which gives around 98% of accuracy during the testing phase. Due to similar pattern in some digit some time it may give the false result at rate of two percent if the pattern of digit not drawn properly. In future the app will be included with the alphabet recognition as well as voice feedback system with some tutorial help with in the app.

References

Liu, C.L., Nakashima, K., Sako, H. and Fujisawa, H., 2002, August. Handwritten digit recognition using state-of-the-art techniques. In Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition (pp. 320-325). IEEE.

Ma, C. and Zhang, H., 2015, August. Effective handwritten digit recognition based on multi-feature extraction and deep analysis. In 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD) (pp. 297-301). IEEE.

Ghosh, M.M.A. and Maghari, A.Y., 2017, October. A comparative study on handwriting digit recognition using neural networks. In 2017 International Conference on Promising Electronic Technologies (ICPET) (pp. 77-81). IEEE.

Shrivastava, A., Jaggi, I., Gupta, S. and Gupta, D., 2019, October. Handwritten Digit Recognition Using Machine Learning: A Review. In 2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC) (pp. 322-326). IEEE.

Islam, K.T., Mujtaba, G., Raj, R.G. and Nweke, H.F., 2017, September. Handwritten digits recognition with artificial neural network. In 2017 International Conference on Engineering Technology and Technopreneurship (ICE2T) (pp. 1-4). IEEE.

Si, J. and Harris, S.L., 2018, January. Handwritten digit recognition system on an FPGA. In 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 402-407). IEEE.

Teow, M.Y., 2017, October. A minimal convolutional neural network for handwritten digit recognition. In 2017 7th IEEE International Conference on System Engineering and Technology (ICSET) (pp. 171-176). IEEE.

Garg, N.K. and Jindal, S., 2007, December. An efficient feature set for handwritten digit recognition. In 15th International Conference on Advanced Computing and Communications (ADCOM 2007) (pp. 540-545). IEEE.

Thepaut, A. and Autret, Y., 1993, October. Handwritten digit recognition using combined neural networks. In Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan) (Vol. 2, pp. 1365-1368). IEEE.

Cheng-I (Chris) Chung, 2000. Hand Written Digit Recognition Using Combination of Neural Network Classifiers (Doctoral dissertation, Southern Methodist University).

Lu, W., Li, Z. and Shi, B., 1995. Handwritten digits recognition with neural networks and fuzzy logic. In Proceedings of ICNN'95-International Conference on Neural Networks (Vol. 3, pp. 1389-1392). IEEE.

Labusch, K., Barth, E. and Martinetz, T., 2008. Simple method for high-performance digit recognition based on sparse coding. IEEE transactions on neural networks, 19(11), pp.1985-1989.