Published by :
http://www.ijert.org

216_Interactive Learning Platform for Kids

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 03, March-2021

Interactive Learning Platform for Kids

Shardul Shetye
Dept. of Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, Mumbai

Ganesh Patil
Dept. of Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, Mumbai

Abhinav Mahajan
Dept. of Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, Mumbai

Prof. Anagha Patil
Dept. of Information Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, Mumbai

Abstract—Education is important for every individual and the country's growth. Our basic objective is to help a child to build a strong foundation for their future academic success by providing a comprehensive and engaging online curriculum to greatly assist early learners to succeed in kindergarten and in early elementary school programs. The current e-learning platforms lack the appropriate infrastructures efficacy. This platform will provide an optimum affordable price package to educational organizations in particular for trainers and learners. We need to combine various technologies to achieve this particular objective.

Index Terms—GoogLeNet, Mobilenet, Alexnet, VGG19 Conovolutional Neural Network.

I. INTRODUCTION

Today's children are spending a considerable amount of their leisure time on mobile or other devices. Very often, they play games on these gadgets. These developments have inevitably led to a new paradigm shift; as learning-via-play or the use of mobile technologies in education have changed the way how students think and process information. Very often, technologies, including visuals and audios provide an immersive, voluntary and enjoyable activity as challenging goals are pursued. Most of the best-selling paid apps in the education category are targeted towards children of age 12 and higher. This learning platform is for children from 2 to 8 years of age, through which they will learn Alphabets, Numbers, Shapes, Colours, etc in a fun, playful way. The most effective way for children to learn a new language is through visuals and interaction, that's why this project has been developed in early childhood so to offer children the best learning experience as they have a good time. It is designed for the youngest to offer a safe learning environment, free from distractions by restricting access to things on smart devices. It provides content that adapts to their age since our main approach is adaptive and interactive. Children will listen, recite, write and memorize letters and numbers effortlessly since day one. They will learn to pronounce and spell in English, from the ABC to 123 to colors and shapes. The student's learning ability and his/her performance will be recorded through regular tests, quizzes, and other activities. This data then will be analyzed with the use of deep learning algorithms. This analysis will be used to plan an efficient learning course for the candidate.

II. PROBLEM STATEMENT

Interactive Learning is Platform for taking tests on drawing numbers and alphabets so to validate drawn input we need to train a deep learning model where the system automatically understands the drawn input and produces the correspond-ing result. We have different architecture available such that Neural Net, Convolutional neural net, VGG 19, res net, etc. selecting best which will give the highest accuracy is hard. Therefore we need to all possible architecture and select one from it. We will also be trying to transfer learning to speed up training and improve performance.

III. LITERATURE SURVEY

The convolutional neural system (CNN) is a traditional kind of DL technique, and it is motivated by the idea of basic and complex cells in the visual cortex in mind. There are three significant layers in CNN: 1) convolutional layer; 2) pooling layer; 3) completely associated (FC) layer. After a few rotating convolutional and pooling layers, the FC layers are prepared for the last order. CNN has accomplished generally effective applications in picture acknowledgment errands. In ImageNet, there are a few effective CNN models, for example, LeNet-5, AlexNet, VGGNet, Google Inception V3, VGG19, MobileNet and so on.

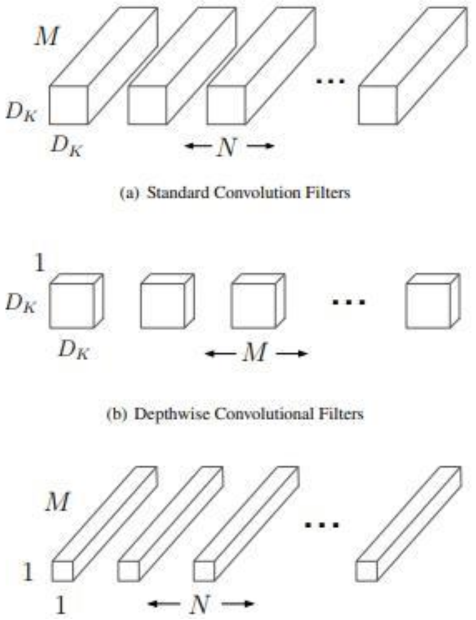
A. MobileNets

Mobile net[1] proposes a class of network architectures that permits a model designer to explicitly pick a small network that coordinates the asset limitations (latency, size) for their application.MobileNets principally center around optimizing the latency of the small networks. Many papers on small net-works center just around size yet don't think about speed. An alternate methodology for getting small networks is shrinking, factorizing or compressing pre-trained systems. Compression is based upon product quantization, pruning, and hashing.

(1)Depth-wise separable convolution The Base of mobile net architecture is depth-wise separable

convolutions. These depth wise convolutions are a form of factorized convolutions that further factorize a standard convolution into a depth-wise convolution and it is called as a point-wise convolution. In the mobile net depth wise convolution a single filter is applied to each input channel. Then 1 X 1 convolution is applied by the point-wise convolution to combine the outputs. A standard convolution has two jobs that are filtering and combining the inputs into a new set of outputs in one step. The depth-wise separable convolution parts this into two layers, a different layer for filtering and a different layer for combining.

.



(a) Standard Convolution Filters

(b) Depthwise Convolutional Filters

Figure 1:Mobile Net Model

(2) Width multiplier and thinner models The base mobile net architecture is a compact and low latency model. Many time there are requirements by a specific case or by an application that the model should be smaller and faster. In order to build these smaller and less expensive models. we present a straightforward parameter called width multiplier. The job of the width multiplier is to thin a network uniformly at each layer. For a given layer and width multiplier , the number of input channels M becomes M and the number of output channels N becomes N. The computational cost of the depth-wise separable convolution using width multiplier alpha () can be given by

formula $D_K \cdot D_K \cdot \alpha M \cdot D_F \cdot D_F + \alpha M \cdot \alpha N \cdot D_F \cdot D_F$

where (0, 1) with ordinary settings of 1, 0.75, 0.5 and 0.25. = 1 is the pattern MobileNet[1] and ¡ 1 are diminished MobileNets. Width multiplier has the impact of re-reducing computational expense and the number of parameters quadratically by generally 2. Width multiplier can be ap-handled to any model structure to characterize another littler model with a sensible precision, inactivity and size exchange off. It is utilized to characterize another decreased structure that should be prepared without any preparation.

B. VGG-19

The CNN models for ImageNet have many layers. So, it is difficult to prepare an exceptionally profound CNN model without the enormous sum of efficient datasets and layers like ImageNet. Hence to simplify it, the transfer learning is joined with CNN models. A few specialists moved the CNN models via preparing a profound CNN model on the ImageNet dataset and afterward applied the prepared CNN model as the component extractor to the little dataset, and they accomplished exceptional outcomes. Right now, new transfer learning dependent on VGG-19 (TranVGG-19)[2] is proposed for fault analysis and it takes the pre-prepared VGG-19 system as the component extractor. VGG-19 is a well known CNN model, and it has effectively applied in image classification, pattern recognition, and speech recognition. Since the RGB picture is the information kind of VGG-19, the information pre-handling strategy is basic to change over the time-space crude information to the RGB picture. a new data prepro-cessing method is introduced. Then, the high-level feature is extracted by using the VGG-19 network. Finally, a new fully-connected layer and softmax classifier are added and trained using the obtained features.

Right now, transfer learning is joined with VGG19 and another TranVGG-19[2] is proposed for shortcoming conclusion by reusing the pre-prepared VGG-19 on ImageNet dataset as the element extractor. The structure of VGG-19 has appeared in Fig. The indication of ’Conv3-64’ presents that the channel is 3*3, and the profundity is 64. Pool/2 implies that it is a maxed pool and the channel is 2*2.

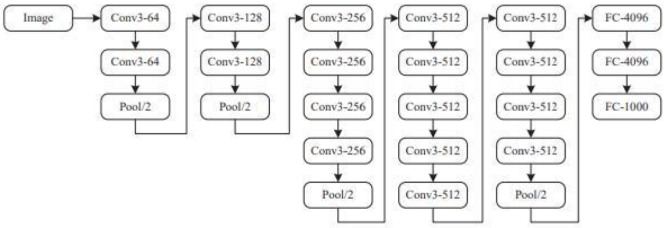


Figure 2 : VGG-19 Model

The principal commitments of this exploration are building up a signal to-RGB transformation technique, proposing TranVGG-19, and applying the TranVGG-19 model to the issue determination documented. The proposed TranVGG-19[2] is conducted on the famous motor bearing data provided by the Case Western Reserve University (CWRU), and achieves the prediction of 99.175 percents. The preparation time of TranVGG-19 is just close to 200 seconds, which is quick in the profound learning field. With the assistance of reusing the pre-prepared VGG-19 system, the TranVGG-19 acquired a noteworthy presentation, including the last prediction exactness and the preparation time, demonstrating the extraordinary potential in fault diagnosis.

C. Qualitative Analysis of GoogleNet and AlexNet for Fabric defect Detection(2019)

Google nets architecture[3] uses 3 different size filters (i.e., 1X1, 3X3, 5X5) for the same image and combines the features to get output. It consists of 22 layers. It has 4 million parameters. The 1x1 convolution is introduced for dimension reduction. This architecture finds the best weight while training and naturally selects the appropriate features. The input RGB images for GoogLeNet is of the size 223 X 223 X 3.

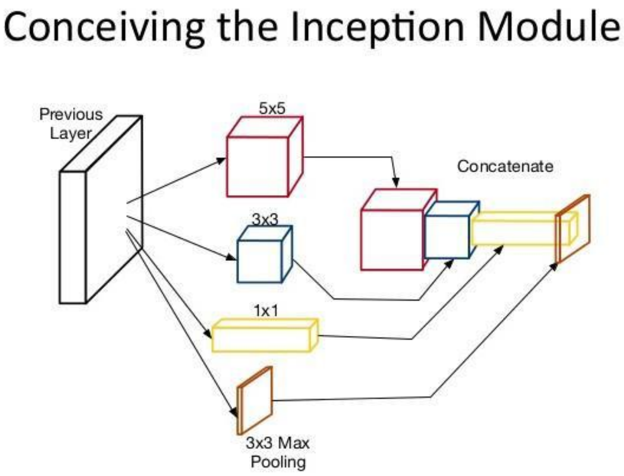


Figure 3:Basic Inception Module

The Fig. shows the basic module in the overall network architecture. Multiple inceptions modules combined forms deeper network, number of parameter increases.

Alexnet

It was proposed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton in 2012. leNwet is extremely small compared to AlexNet. AlexNet[3] comprises of eight layers: 5 convolu-tional layers and 3fully connected layers. 2 new concept layers

(1) Maxpooling (2) ReLU activation. AlexNet takes the input of the size 227 X 227 X 3. The 11 X 11 convolutional filters are with stride 4 and 3 X 3 max polling filters are with stride 2.

The Fig. 4 shows the overall architecture of AlexNet

Analysis of GoogleNet and Alexnet Training

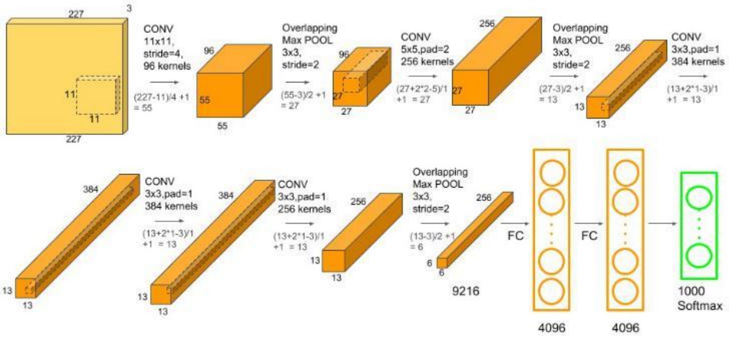
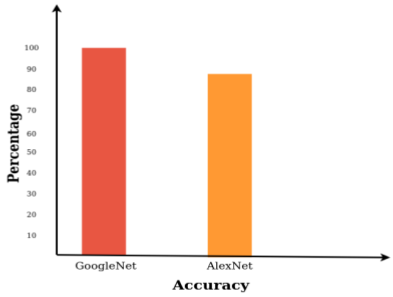


Figure 4:ALexNet Architecture

GoogLeNet and AlexNet[3] are some of the top deep learning methods employed in the defect detection process. While both do a great job in enabling this process, GoogLeNet's performance outweighs that of AlexNet concerning different parameters. The results of the experiments indicate that the former is better than the latter in terms of speed, accuracy, dropout, and the initial learning rate. As such, preference should be one GoogLeNet over AlexNet when faced with the choice of the best deep learning methods.

Comparitive plots for GoogLeNet Vs AlexNet (1)Training Time (2)Validation Accuracy (3)Number of layers (4)Initial learning rate



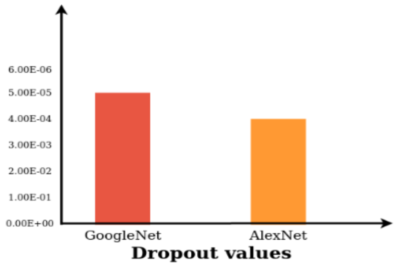
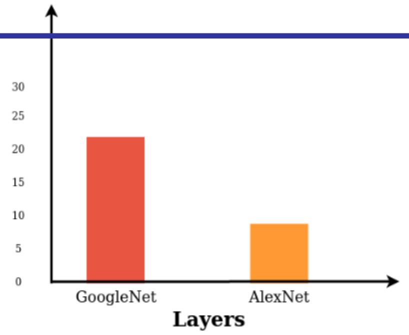
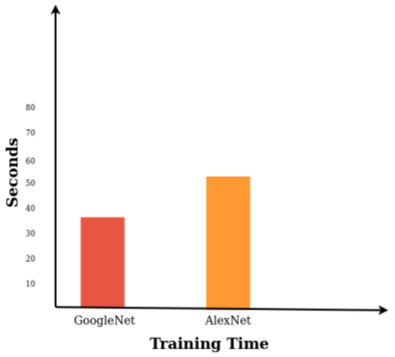


Figure 5 : Graphs Showing Comparisons of Accuracy, Dropout Values, Layers,





D. Recognition of Handwritten digits using various Algorithms

CNN is assuming a significant job in numerous areas like image processing. It powerfully affects numerous fields. Indeed,in nano-innovations like fabricating semiconductors, CNN is utilized for flaw location and order. Manually written digit acknowledgment has become an issue of interest among scientists. There are a huge number of papers and articles are being distributed these days about this subject. In examination, it is indicated that DeepLearning calculation like multilayer CNN utilizing Keras withTheano and Tensorflow gives the most noteworthy exactness in examination with the most generally utilized AIcalculations like SVM, KNN, and RFC. In view of its most elevated exactness, Convolutional Neural Network (CNN)[5] is being utilized for an enormous scope in picture grouping, video examination, and so on. Numerous analysts are attempting to make a supposition acknowledgment in a sentence. CNN is being utilized in characteristic language preparation and assumption acknowledgment by differing var-ious parameters. To understand the manually written digits, a seven-layered a convolutional neural system with one info layer followed by five concealed layers and one yield layer is structured The input layer comprises of 28 by 28-pixel images which imply that the system contains 784 neurons as information. The information pixels are in grayscale with a worth 0 for a white pixel and 1 for a dark pixel. Here, this model of CNN has five shrouded layers. The primary concealed layer is the convolution layer 1 which is liable for include extraction from input information. This layer performs convolution activity to little limited regions by convolving a channel with the past layer. In addition, it comprises of various component maps with learnable bits and corrected direct units (ReLU). The bit size decides the territory of the channels. ReLU is utilized as an initiation work toward the finish of every convolution the layer just as a completely associated layer to upgrade the execution of the model. The following shrouded layer is pooling layer 1. It decreases the yield data from the convolution layer and decreases the number of parameters and computational multifaceted nature of the model. The various kinds of pooling are max pooling, min pooling, normal pooling, also, L2 pooling. Here, max pooling is utilized to subsample the measurement of each element map. Convolution layer 2 and pooling layer 2 which has a similar capacity as convolution layer 1 and pooling layer 1 and works similarly aside from their component maps and bit size shifts. A smooth layer is utilized after the pooling layer which changes over the 2D included guide framework to a 1D highlight vector and permits the yield to get taken care of by the completely associated layers. A completely associated layer is another concealed layer too known as the thick layer. It is like the shrouded layer of Artificial Neural Networks (ANNs)[5] however here it is completely associated and interfaces each neuron from the past layer to the following layer. So as to diminish overfitting, the dropout regularization technique is utilized at completely associated layer 1. It arbitrarily turns off certain neurons during preparing to improve the presentation of the system by making it more hearty.

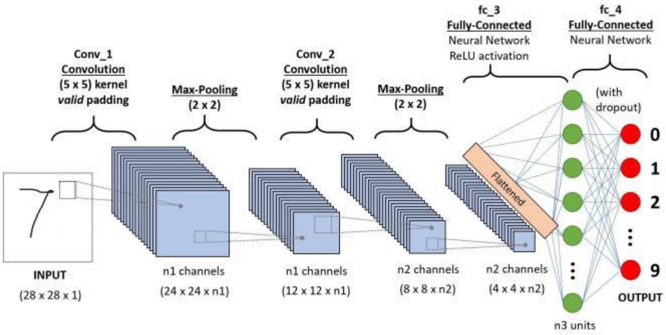


Figure 6 : CNN Architecture

Modified National Institute of Standards and Technology (MNIST) is a huge arrangement of PC vision dataset which is broadly utilized for preparing and testing various frameworks. It was made from the two exceptional datasets of the National Foundation of Standards and Technology (NIST) which holds parallel pictures of manually written digits. The preparation set contains written by hand digits from 250 individuals, among them, half preparing dataset was workers from the Census Authority and its remainder was from secondary school understudies. Be that as it may, it is regularly credited as the first datasets among different datasets to demonstrate the adequacy of the neural systems. The database contains 60,000 pictures utilized for preparing as well as not many of them can be utilized for cross-approval purposes and 10,000 pictures utilized for testing. All the digits are grayscale and situated in a fixed size where the power lies at the focal point of the picture with 28×28 pixels. Since all the pictures are 28×28 pixels, it frames an exhibit that can be leveled into a 28*28=784 dimensional vector. Every part of the vector is a twofold worth that portrays the power of the pixel.

E. Handwritten Digits Recognition with Artificial Neural Net-work

It is very difficult to build a database, including all the normal examples of the unconstrained numerals. For feature extraction of character recognition, various approaches have been proposed. Handwritten digit recognition is an active area of research in optical character recognition applications also, design arrangements. Handwriting recognition is offline handwriting recognition whereas online touch pad writing is called online handwriting recognition.

This study[6] focuses on feature extraction and classification. Moreover, our study presents an efficient offline and online handwritten and digital touchpad character recognition system based on diagonal features transitions features using the KNNclassifier. Diagonals and advance highlights of a character have been processed dependent on the distribution of points on the bitmap image of the character. In this study, we have compared the performance of five different machine learning classifiers for the recognition of digits. The five classifiers are the Neural Network, K-Nearest Neighbor, Random Forest, Decision Tree, and Bagging with gradient boost.

The Multi feature extraction of Handwritten Images

A.Pre Processing: The accuracy in recognition of handwritten

digits can be improved by preprocessing the data. With a

brief study of the raw image data, the main issues found

are image noise and unrecognizable handwriting. Therefore,

preprocessing of the data is made to be necessary before

training them.

B.Normalization: This is done to apply distance calculations on it. It involves transforming the data to fall within a smaller or common range, such as [0, 1]. The raw image data is based on the standard 8-bit unsigned integer which has a high-value range of [0, 255] at each pixel (attribute). Expressing an attribute in smaller units will lead to a larger range for that attribute, thus tend to give such attributes greater effect or weight.

C.Noise Reduction: After Normalization, we can use the median filter to remove noise this is a nonlinear digital filtering technique to improve the image by removing especially Gaus-sian noise. Median Filter preserves the edge while removing the noise as the edge is an important aspect of an image.

D.Image Sharpening: We can sharpen the image by sharpening technique which uses a blurred, or ”unsharp”, negative image to create a mask of the original image.

Another approach[6] to recognize the numerals is by using CNN. Handwritten digits recognition has been widely used for implementing practical applications like computerized bank checks numbers reading. It is a complex task and uses MNIST data set of handwritten digits to train and test the model.This multi layer artificial neural network provides an accuracy of about 99.60%. This paper states an artificial neural network ANN to recognize handwritten digits (0 to 9). The MNIST data set has thousands of labeled images of handwritten digits written by numerous people. These images are low resolution, just 28-by-28 pixels in gray scale, and are segmented. Thus each image has 784 pixels and these pixels are used as features. The model uses the flattened representation of the image that is the image is converted from a 2D array to 1D array by unstacking the rows and lining them up. ANN is employed as a classifier to construct a classification model. Two parameters were provided to ANN. The first indicates the number of classes in the data set which is 10 in our data set, one for each digit. The second parameter informs the classier about the number of features that have been used. The basic ANN contains an input layer, a hidden layer, and output layer. In general, input neurons are exactly of the same size that of features vector i.e. 784. The neurons of the output layer are10 as the proposed system have 10 class (09). A systematic method and back propagation learning algorithm[6] are applied to train the ANN. Only one hidden layer with 100 neurons was taken, which found to give the best performance for the proposed application. Research shows that a neural network with one hidden layer can perform the approximation of any function.At the training stage, the data set is distributed in appropriate percentages (%) of training, validation, and testing data to avoid the network over fitting problem. While training, the ANN is adapted according to the error of the network by back propagation.After a satisfactory training result, testing is performed. A confusion matrix is used to summarize the performance of the model.

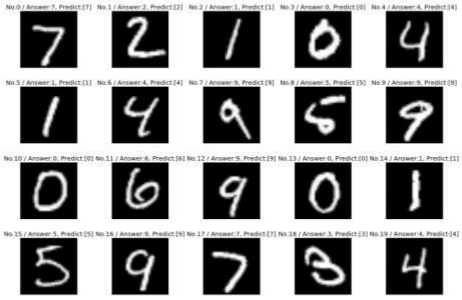

No.0 / Answer:7, Predict:[7]
No.1 / Answer:2, Predict:[2]
No.2 / Answer:1, Predict:[1]
No.3 / Answer:0, Predict:[0]
No.4 / Answer:4, Predict:[4]
No.5 / Answer:1, Predict:[1]
No.6 / Answer:4, Predict:[4]
No.7 / Answer:9, Predict:[9]
No.8 / Answer:5, Predict:[5]
No.9 / Answer:9, Predict:[9]
No.10 / Answer:0, Predict:[0]
No.11 / Answer:6, Predict:[6]
No.12 / Answer:9, Predict:[9]
No.13 / Answer:0, Predict:[0]
No.14 / Answer:1, Predict:[1]
No.15 / Answer:5, Predict:[5]
No.16 / Answer:9, Predict:[9]
No.17 / Answer:7, Predict:[7]
No.18 / Answer:3, Predict:[3]
No.19 / Answer:4, Predict:[4]

Figure 7 : Real Dataset Image Showing Different Handwriting

F. An Efficient Handwritten Devnagari Character Recognition System Using Neural Network

Devnagari[12] character recognition using Artificial Neu-ral Network[5] is a greater challenge. Devnagari characters are converted to image files and after that, these images are digitized into binary form and preprocessed further for isolation with the help of a bounding box to generate input for training a neural network. In Feature extraction, each of these characters is defined by the presence or absence of key features like height, width, density, loops, lines, stems, and other characteristic traits. Using this each character is judged and data is minimized by discarding redundant and unnecessary information. This results in a vector with scalar values. The following parameters are being used for creating the network for training: No. of neurons in Input Layer: 35 No. of Hidden Layer: 2

No. of neurons in each Hidden Layer: 33

Network training parameter epochs: 5000

No. of epochs: 244

Network training parameter goal: 0.1

Transfer Function Used: “Logsig”, “Logsig”

Adaption Learning Function: “Traingdx”

Performance Function: “SSE”

Devnagari Script comprises of characters which are composed of more than one component, this leads to some difficulties in segmentation. This complexity and similarity of characters with each other make it a bit difficult to analyze the per-formance. With all the characters included, the accuracy of the model was about 60% whereas it was about 75.6% when excluded some of the complex characters.

G. Classification of Noisy English Alphabets Using Neural Network

Character recognition is one of the most fascinating areas of pattern recognition. The scope of character recognition systems has emerged as a very wide area of research. Its application ranges from Data entry for business documents, e.g. cheque, passport, invoice, bank statement, and receipt. There are many algorithms for this but noise suppression and correct identification of the characters up to a reasonable level of accuracy is still a challenging task. The presented algorithm requires a lesser number of input neurons for training as compared to other algorithms. A network is to be trained for the learning pattern of each of the alphabets. To minimize the inputs, only upper case letters are considered and represented by a 5×3 array of dots. For the modeling of each letter, these dots are coded as 1 and -1 for blank spaces.

IV. PROPOSED SYSTEM

The Flowchart depicted below emphasizes the procedural flow the constructed model is expected to follow

Steps for Image Classification using GoogLeNet:

Step 1: Load Images This is the first step in classification. It involves stacking a little arrangement of preprocessed picture information of around 1000 pictures. The dataset is split into two classifications. The first is training that makes up 70% of the pictures while the subsequent one is the test set comprising of the staying 30% of the pictures.

Step 2: Load Pretrained GoogLeNet Network In this step, We first load the pre-trained GoogLeNet network. We fine-tune some layers of the network. Even we freeze the learning of layers for few iterations which leads to increases in training speed. There is an extraction of features from the input images via the convolutional layer which are then analyzed by the classification layer.

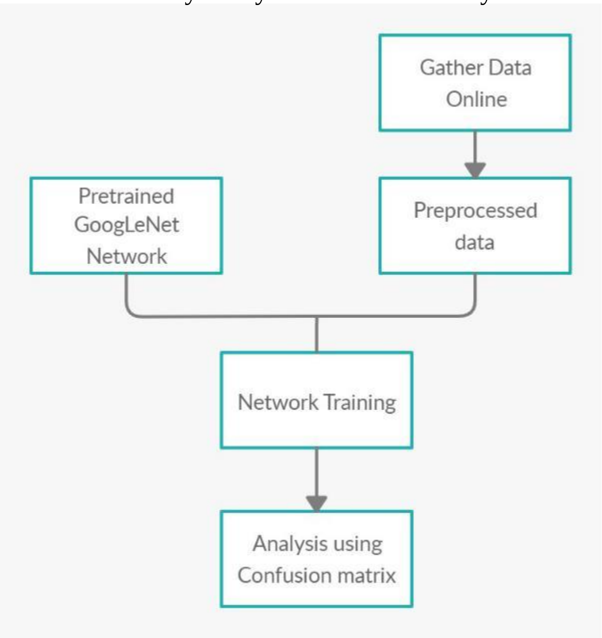


Figure 8: Proposed Model

Step 3: Network Training The preprocessed pictures for a system that will be prepared must be 223x223x3 in size. However, the images in the data set vary in terms of size. Image size is restored through the use of augmented image datastore. Augmentation of data when it comes to the training of the network restricts overfitting while at the same time keeping a memory of the characteristics of the training images. Further, a trained network is used for classifying the authorized images using predicted labels and forecasting the probabilities of the images that hold these labels.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 9 : Confusion Matrix

Step 4: Analysis using confusion matrix A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows how your classification model is confused when it makes predictions. It gives us knowledge not just into the mistakes being made by a classifier however more significantly the sorts of blunders that are being made.

V. RESULT AND CONCLUSION

After looking at the related work and doing the combined study of the papers and planning the project implementation we came to this conclusion, we have selected some algorithms and flows which can provide us with better accuracy in our project and also help inefficiency of architecture without affecting the accuracy. There might come changes in the algorithms as the project progresses and also the flow might change as we go for more research on the flow and algorithms for time being we will be using the Google Net algorithm which until now provides the highest accuracy according to the author of one of the paper still it is to be tested on our level but the results might not differ much in case of accuracy.

The model we designed is tensorflow model based on python language. And the system we designed is node.js based platform. Hence we converted our model to javascript version using tensorflow.js library and integrated the model in our system. Tfjs library allows us to load the model in node.js backend.

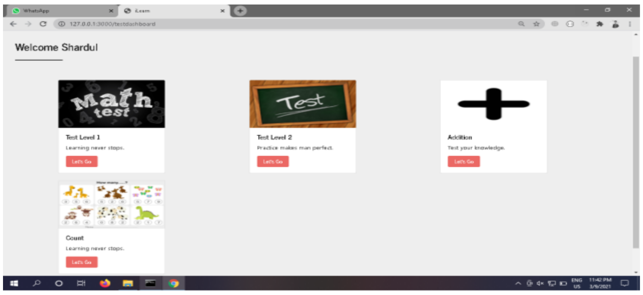
The first step of the model is to register in the application. After successful registration, the parental login system comes into the picture. This authentication is done with passport.js library and mongodb database. After logging in to the system the user will get a preference to select one of the phases i.e. Learning, Practise, Test. If Learning is selected the child will be getting knowledge of the Alphabet, Numbers without audio sound. In the Practice Section, the child will be improving handwriting. One canvas will be provided on which Alphabets will be displayed by writing them again by using a stylus. Finally in the test section, test will be provided and the student will be instructed to write a number or alphabet on the screen on the canvas, which will be saved and checked by model and the result will be displayed. The working of test phase is as followsWhen test gets on the interface, node.js loads the ml model in the backend.

The system logic will generate random digit to be asked as question in the test. User will draw the answer digit on the canvas provided and submit it. The image of canvas will be captured by the system and will be converted into image data uri i.e. base64 format. This image data will be then passed to pre processing function that will bring image in suitable format and pass it to our ml model. Then the model will classify the drawn digit in an image and pass it to the system where it will be compared with the initially generated question digit. If it matches that means user is correct and the interface will turn green to indicate the same else it will be red.

After we created the ml model for classifying handwritten digits with enough accuracy, we started working on integrating it in a proper system. The system we planned is a web based platform. Hence we decided to use Node.js as the web framework and mongodb as a database to store the user information. We first created the simple user authentication system using passport.js library.

Then the task was to convert the current python based ml model into javascript as we needed to integrate it in nodejs. For this we used the tensorflow.js library that lets us convert the model as well ass load it in our nodejs. By doing this we were almost achieved objective of this project that is test sessions. And since then we worked on other modules of project like learning and practicing phase. Following are some of our implementation results.

Figure 10 : Dashboard



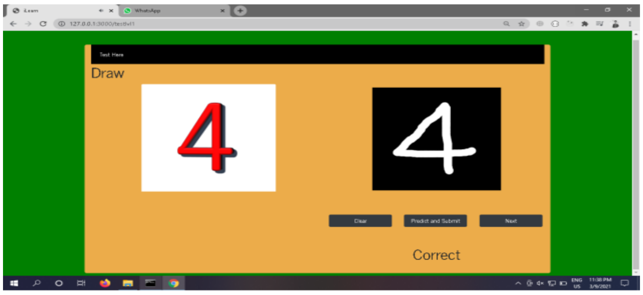


Figure 11 : Model predicting answer is correct.

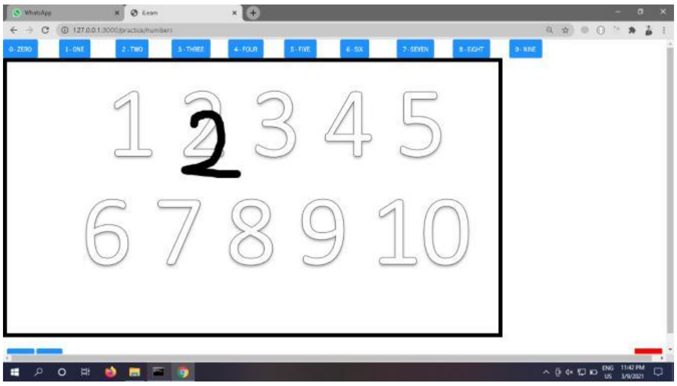
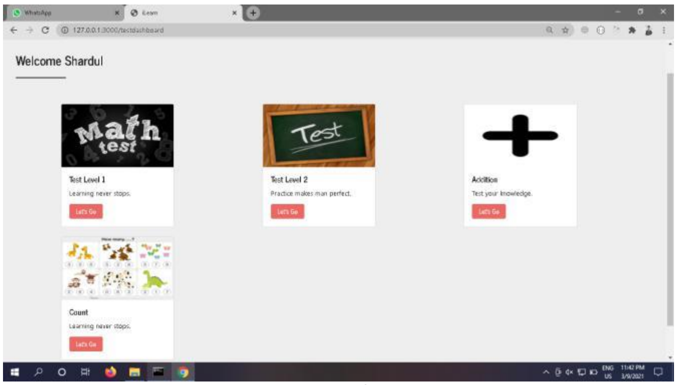


Figure 12 :Canvas for practising numbers.



Welcome Shardul

Test Level 1

Learning never stops.

Test Level 2

Practice makes man perfect.

Addition

Test your knowledge.

Count

Learning never stops.

Figure 13 :Features of the test section.

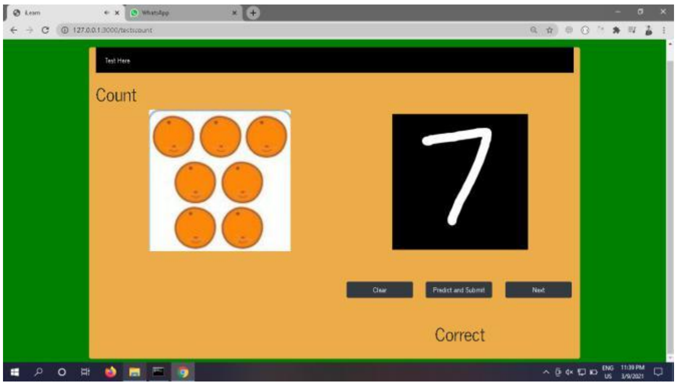


Figure 14 :Features of the count section.

REFERENCES

[1] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, ”MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications”, April 2017.

[2] Long Wen, X. Li, Xinyu Li and Liang Gao, ”A New Transfer Learning Based on VGG-19 Network for Fault Diagnosis” , IEEE 23rd International Conference on Computer Supported Cooperative Work in Design, April 2019.

[3] K. K. Sudha, P. Sujatha, ”Qualitative Analysis of GoogleNet and AlexNet for Fabric defect Detection”, International Journal of Recent Technology and Engineering, May 2019.

[4] Chuanqi Tan, Fuchun Sun, Tao Kong, Chao Yang, Wenchang Zhang, Chunfang Liu, ”A Survey on Deep Transfer Learning”, 6 Aug 2018.

[5] Kh Tohidul Islam, Ghulam Mujtaba, Dr. Ram Gopal Raj, Henry Friday Nweke ”Handwritten Digits Recognition with Artificial Neural Network”,20 September 2017.

[6] Shengfeng Chen, Rabia Almamlook, Yuwen Gu, Dr. Lee wells ”Offline Handwritten Digits Recognition Using Machine learning”,27 September 2018.

[7] Caiyun Ma, Hong Zhang ”Effective Handwritten Digit Recognition Based on Multi-feature Extraction and Deep Analysis”, 2015.

[8] S M Shamim, Mohammad Badrul Alam Miah,Angona Sarker,Masud Rana, Abdullah Al Jobair ”Handwritten Digit Recognition using Ma-chine Learning Algorithms”, 2018.

[9] Yusuf Perwej,Ashish Chaturvedi ”Neural Networks for Handwritten English Alphabet Recognition”, 7 April 2011.

[10] Jin Ho Kim, Kye Kyung Kim, Sung II Chien, ”A Survey on Deep Transfer Learning”, 1995.

[11] Chanda Thapliyal Nautiyal,Sunita, U.S. Rana,Rahul Kumar ”Classification of Noisy English Alphabets Using Neural Network”,16 October 2016.

[12] Ms. Neha Sahu, Mr. Nitin Kali Raman ”An Efficient Handwritten Devnagari Character Recognition System Using Neural Network”,2013.