ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

Educational Platform for Pre-Primary Students using Image Processing and Machine Learning

Aditi Jahagirdar[1], Abhishek Pawar[2], Hrushabh Kedari[2], Srushti Borade[2], Surabhi Khilari[2]

Asst. Professor, Dept. of Information Technology, MAEER's MIT College of Engineering, SavitribaiPhule Pune University, Pune, Maharashtra, India[1]

B.E Student, Dept. of Information Technology, MAEER's MIT College of Engineering, SavitribaiPhule Pune University, Pune, Maharashtra, India[2]

ABSTRACT: This paper proposes us a solution on how teaching can be more effective and interesting for the pre-primary students. The goal is to present a mixed methodology that combines machine learning, image processing and a flavour of augmented with traditional educational materials like books and course packs which will be effective in increasing learners' concentration and passion for learning. With the development of smartphone technology, an application is developed on Android platform with the help of TensorFlow technology and Unity 3D. The application consists of two modules- a) Teaching A to Z alphabets with a 3D pop up of respective objects b) Detecting those objects in real time.The average training accuracy of the model is 95% which shows that the project has strong availability and applicability, at the same time, it also has a good future scope. This new style of learning helps to attract more learners.

I. INTRODUCTION

All we sail through the 21st century, technology in the classroom or at home is becoming more and more predominant. The use of smartphones for any purpose is increasing day by day. Tablets are replacing our textbooks, and blackboards are replaced by digital blackboards. Now Education is more about visualization and practical knowledge rather than the theoretical one. This widespread adoption of technology has completely changed how teachers teach and students learn. Students are using advanced technology to shape how they learn. As mobile is handy we can use it anywhere anytime, so many business units in many industries have preferred to develop applications on the mobile platform.

Our brains make vision seem easy. It isn't difficult for a human being to differentiate between a tiger and a zebra or between different faces. But when it comes to the computer, it is really hard for a machine to differentiate between objects. Thefield of machine learning has made tremendous progress in solving or addressing such difficultproblems. A new model is introduced called as Convolutional Neural Networks (CNN) which achieves better performance. Augmented Reality[1] [10]is another new technology which can be implemented well in Educational fields for improving imagination and visualization of the students. AR technology has an ability to render objects that are hard to imagine and turn them into 3D models, thus making it easier to grasp the abstract and difficult content.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

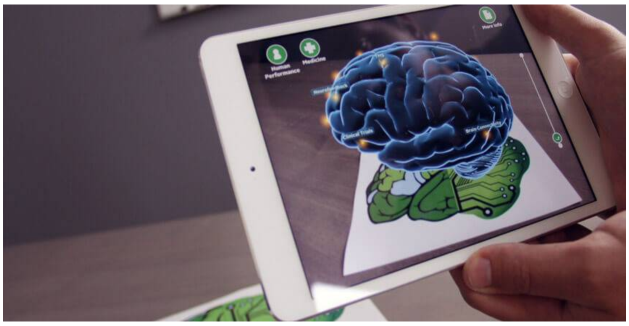

Figure 1: The user points a smartphone screen on the textbook and then augmentation is seen with more information on the screen. [6]

"Alphabetter" is an application we developed which will help the pre-primary students to learn in a more exciting way. We focused on developing a real-time recognition of objects and 3d pop of those objects up to help visualize the child more about the specific object.

In this paper, we will explain more details about our application, the users who can use this application, the system architecture and design, methodology, experimental results of software, conclusion and future scope of our application.

II. RELATED WORK

TensorFlow[2] [9] [11] provides an interface to express machine learning algorithms and an implementation to execute such algorithms. TensorFlow uses two types of models for image recognition which can be used 1) MobileNet[3] 2) Inception V3 [4], both models work on CNN [5] [10] [12]. The primary difference in both the models is that Inception V3 is optimized to achieve more accuracy, while the MobileNets are optimized to be small in size and efficient, at the cost of some accuracy. ACNN consists of one or more convolutional layers proceeded by one or more fully-connected layers such as in a standard multilayer neural network. The CNN architecture is designed to take advantage of the 2D structure of an input image. This can be achieved with local connections and tied weights followed by some type of pooling which results in translation invariant features. Another benefit of CNN is that it is easier to train with a few parameters than fully connected networks which have the same number of hidden units

An existing commercial augmented reality application such as Lay(AR) [7] is available for advertising purposes. It is a mobile application that applies augmented reality technology to make the product more attractive, interesting and valuable. It means that using augmented reality can make the product more valuable and increase the number of buyers more than the non-augmented reality product. The users can use Lay(AR) by pointing the mobile camera at images to see extra digital information such as text, videos and 3D models.

III. PROPOSED SOLUTION

The proposed mobile application is called Alphabetter. It is an application that is designed on Android platform with the help of TensorFlow technology and Unity 3D[8]. If you ask any parent of a 3-year-old or a teacher teaching in pre-primary as to how difficult it is to make small children sit at one place and make them study, you'll mostly get answers saying the task is particularly difficult. Also, who can you blame; kids are so interesting in themselves that studying from a 2D book while sitting still is just too boring for them. Here is where our AR book application for children will come in handy! The basic idea here is to design a book ourselves for starters, having the letters from the alphabet written out. Then using our application on their parent's cell phones, all the kids have to do is point the phone's camera towards any alphabet and voila, a pop-up image is seen on top of this letter corresponding to it. For example, if the

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

letter 'A' is pointed at by the camera then the image of an 'Apple' is popped up using AR technology through Unity 3D. And in the later part the student is asked to find an apple for the letter A in the real world, scanning it correctly will give him/her points and for wrong answers no points will be given. The scanned image is processed with the help of TensorFlow technology which performs correct classification of the image.

To provide the details of our application we like to adopt the following questions: "Who will use Alphabetter?", "How many modules are there in Alphabetter?", "WhyAlphabetter is needed? ", "How to access Alphabetter system?"

- Who will use Alphabetter? **-** Alphabetter is developed for pre-primary students to learn the alphabets in more interesting way. This application can be used by nursery schools and even at home.

- How many modules are there in Alphabetter? **-** Alphabetter has two modules i.e. two levels in first level the child will be taught A to Z alphabets with object popup and in the second level the child will be told to identify different objects by scanning them in real time.

- Why Alphabetter is needed? **-** Alphabetterwould be served for pre-primarystudents and they will now find it really interesting to study and learn much more thanbefore in a fun way.

- How to access Alphabetter system? **-** Alphabetter would be accessed through the installed Android application. The application needs good camera i.e. greater than 4 MP.

IV. SYSTEM ARCHITECTURE AND DESIGN

Alphabetter consists of two Levels: In level 1teaching A to Z alphabets with 3d pop of respective objects. In level 2 detecting those objects in real time. This recognition of objects in real time is done by TensorFlow technology. The mobile application is based on Android platform with JAVA as the programming language.

For the first level Unity 3d software is used with database in Vuforia with programming language as C#. For the second level TensorFlow software is used which uses Python as the programming language. Using the book designed by us, if the letter 'A' is pointed at by the camera then the 3D image of an 'Apple' is popped up using AR technology with Unity 3D software.. In level 2 TensorFlow software is used with MobileNet as the model wherein the student is asked to find an apple for the letter "A", then the graph matches the pattern and the respective label with the input image.Then we compare that the pattern of the object and the label the user selected for detection and the pattern of the live image is almost same. And if they are then we print on the screen "Correctly found" else we display "Wrong".

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

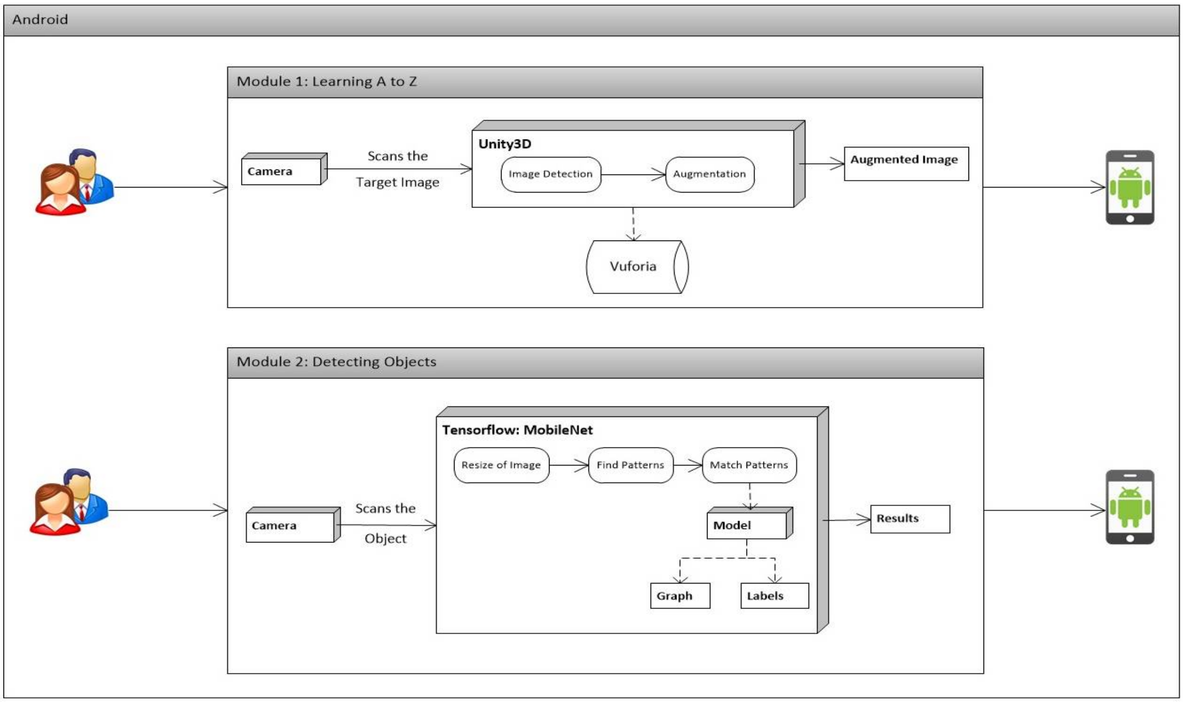

Figure 2: System architecture and design

V. METHODOLOGY

Our application applies the knowledge of image processing on received input from the mobile camera, match those images in the database and represent the popup objects for first level and recognition for the second level. To understand the methodology in more detail we have divided it into parts for each level: In first level A) Detection Method B) Augmentation Method. For second level C) Process D) Detection Method E) Pattern Matching.

A.Detection Method

Unity3D [5] is a powerful suite of tools (Project IDE, Code IDE, run-time) for AR. Firstly the target image for each alphabet is stored in the database called Vuforia[6]which submits it to Vuforia's Target Manager, making it recognizable by the system. The target image is nothing but a marker which differentiates each alphabet from other.In the background, Vuforia creates an arrangement of the image using its features, and then the algorithm can find such patterns and track the targets. Basically, a feature in an image is a sharpened angle, for example: a box corner or the tip of a star. We have created the AR book which have the target image for each alphabet. We have taken the alphabets in upper case and lower case as the target images. If you scan the target image the 3d object will be augmented on the screen. When the mobile camera scans the target image then the target image is searched in the Vuforia[6]database.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

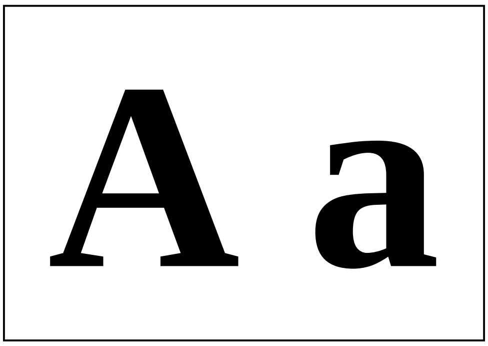

Figure 3: Target Image.

B. Augmentation Method

The second process is augmentation method. First of all, the application will capture the frames from video and scans the target image. This target image will be compared with the target images in the Vuforia database. Now if the target image matches then the respective 3d object is augmented on the mobile screen with the respective audio. E.g.: "A for Apple" will be played at certain intervals till the mobile camera is scanning the target image of alphabet "A".

C. Process

1. Firstly, a set of images are used to teach the network about the new classes to be recognized.
2. Then the first phase analyzes all the images on disk and calculates the bottleneck values for each of them.
3. Bottleneck is penultimate layer which has been trained to output a set of values that's good enough for the classifier to use to distinguish between all the classes it's been asked to recognize.
4. Once the bottlenecks are complete, the actual training of the top layer of the network begins
5. You'll see a series of step outputs, each one showing training accuracy, validation accuracy, and the cross entropy.
6. The training accuracy shows what percent of the images used in the current training batch were labelled with the correct class. The validation accuracy is the precision on a randomly-selected group of images from a different set
7. Cross entropy is a loss function which gives a glimpse into how well the learning process is progressing.
8. By default, this script will run 4,000 training steps.
9. An accuracy value of between 90% and 95%, though the exact value will vary from run to run since there's randomness in the training process.
10. This number is based on the percent of the images in the test set that are given the correct label after the model is fully trained.
11. The script consists of TensorBoard[7]Summaries that makes it easy to understand, debug and optimize the retraining.
12. Finally, the test data can be tested.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

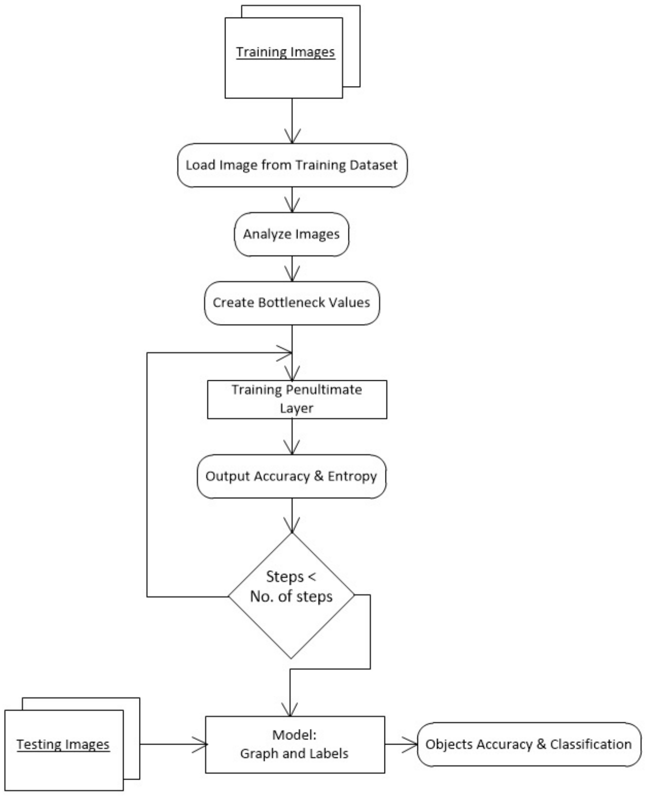

Figure 4: Overview of the process

D.Finding Pattern

We start by creating a Graph Builder, we use to specify a model to run or load. We then start creating nodes for the small model we want to run to load, resize, and scale the pixel values to get the result the main model expects as its input. The first node we create is just a Constant that holds a tensor with the file name of the image we want to load. Now the system tries to figure out what kind of file it is and decode it. Then the system will keep adding more nodes to decode the file as an image, tocast the integers into floating point values, to resize the image and finally run operations on the pixel value. Then the system keeps adding more nodes, to decode the file data as an image, to cast the integers into floating point values, to resize it, and then finally to run the operations on the pixel values. At the end of this we have a model definition stored, which we turn into a full graph definition. Then we create an interface to actually running the graph, and run it, specifying which node we want to get the output from, and where to put the output data.

This gives us a vector of Tensor objects. You can think of a Tensor as a multi-dimensional array in this context, and it holds a 224-pixel high, 224 pixels wide, and 3 channel image as float values, image size should be according to the model that we have chosen for our model we have chosen MobileNet.

E. Pattern Matching.

As the TensorFlow model uses graph and a label file as its database. The patterns are stored in this graph file which the application use for pattern matching and the label file contains the names of the object which the model is trained for. The graph file is stored in .pb extension i.e. it is a binary file and the label file is stored in the text format. The graph is

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

stored in the binary file so that no changes can be made to it once the graph is created. The first graph which is created requires more memory and is not compatible for mobile platform. Hence, we need to optimize and quantize the graph created. The graph is then optimized first and then it is quantized. Then for pattern matching the main graph i.e. the quantized graph is loaded with the labels and the apk is built. Then the image from disk as a float array of numbers, resized and normalized to the specifications the main graph are expected. A sorted list of the highest-scoring labels is created. And then runs the short graph to get a pair of output tensors. In this case they represent the sorted scores and index positions of the highest results. The Print function takes those sorted results, and prints them out in a friendly way. As the model which accepts the images of size 224 x 224 then we Load, resize, and process i.e. actually run the image through the model. As we detect the objects at runtime streams of frames are given to the model. The video is converted to images/frames per sec and then the images are given to the model.Then the graph matches the pattern with the input image. Then we compare that the pattern of the object the user selected for detection and the pattern of the live image is almost same. And if they are then we print on the screen “Correctly found” else we display “Wrong”.

VI. EXPERIMENTAL RESULTS

A. Matching Performance and Time Delay

We experimented on various combinations in which the model can be trained. By default, there are 4000 steps which the MobileNet model uses, but we can provide the number of steps by ourselves. We have also tried various combination in training images. And depending on this the accuracy and time required to build the model is being plotted on the graphs.

steps	Time required to train model	Average Accuracy	Recognition time (sec)	Test accuracy
500	14 min: 05 secs	95.2%	0.609	0.9998869
800	13 min: 15 secs	95%	0.609	0.99995613
1000	15 min: 58 secs	94.7%	0.625	0.99996936
2000	21 min: 43 secs	94.9%	0.609	0.99999046
4000	28 min: 30 secs	95%	0.609	0.99999714

Table 1: Combinations for 250 images

steps	Time required to train model	Average Accuracy	Recognition time (sec)	Test accuracy
500	10 min: 58secs	92.9%	0.641	0.9998207
800	9 min: 16secs	93.6%	0.625	0.9999273
1000	12 min: 57secs	93.8%	0.609	0.9999521
2000	16min: 40secs	93.1%	0.625	0.9999834
4000	15 min: 48secs	93.6%	0.641	0.9999941

Table 2: Combinations for 150 images

Steps	Time required to train model	Average Accuracy	Recognition time (sec)	Test accuracy
500	6 min: 48 secs	94.6%	1.355	0.9953011
800	6 min: 43 secs	94.7%	0.936	0.9972005
1000	7 min: 21 secs	94.9%	1.410	0.9985511
2000	8 min: 51 secs	95.2%	0.999	0.9998901
4000	10 min: 57 secs	95.4%	0.890	0.9999098

Table 3: Combinations for 100 images

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

steps	Time required to train model	Average Accuracy	Recognition time (sec)	Test accuracy
500	3 min: 18secs	94.7%	0.766	0.9993174
800	3 min: 59secs	93.4%	0.625	0.9998544
1000	4 min: 17secs	94.1%	0.625	0.9998930
2000	5 min: 52 secs	94.1%	0.625	0.9999484
4000	8 min: 53 secs	94.1%	0.625	0.9999844

Table 4: Combinations for 50 images

From the above experimental results, we came to the conclusion that the model will be either of 250 images or 100 images.

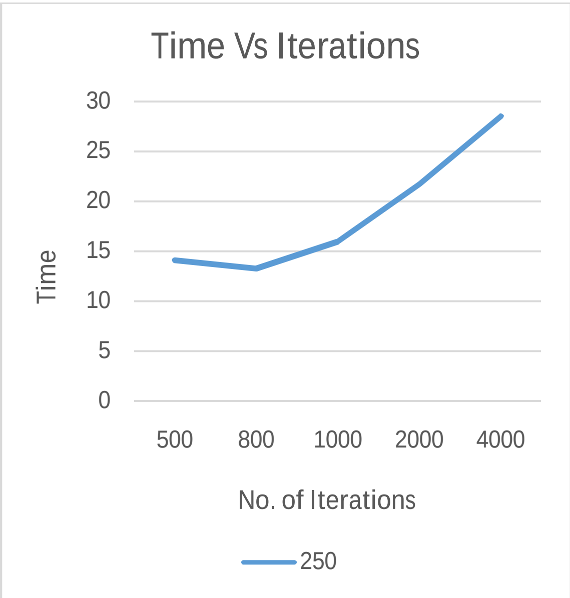

Figure 7: Graph of time vs iterations for 250 Images Iterations

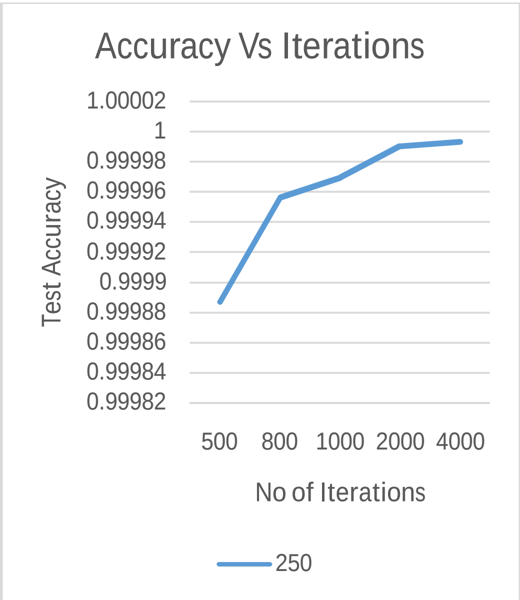

Figure 8: Graph of accuracy vs for 250 images

 DOI: 10.15680/IJIRCCE.2018.0602017

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

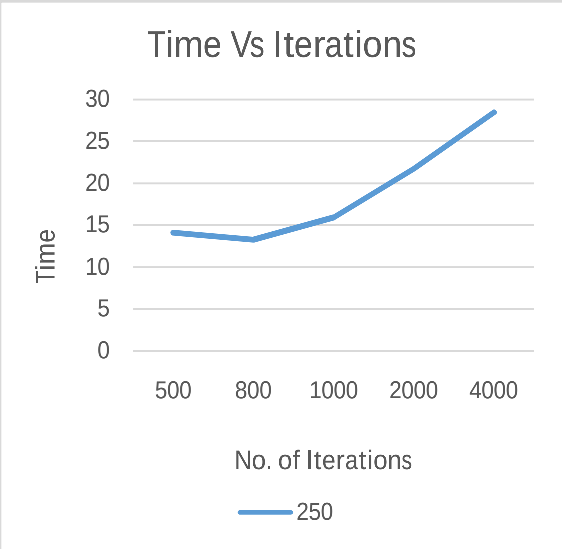

Figure 9: Graph of time vs iterations for 100 Images

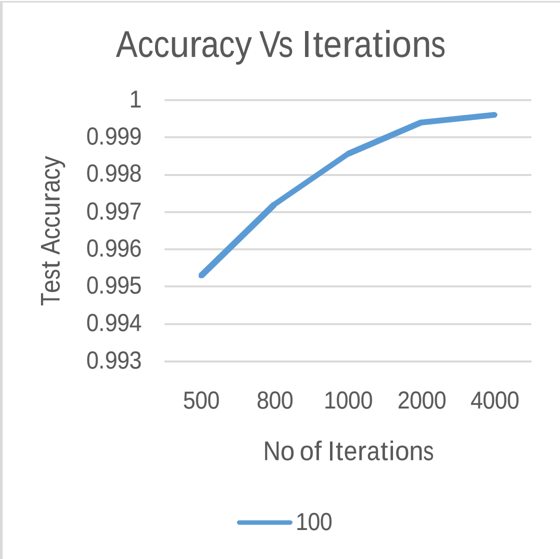

Figure 10: Graph of accuracy vs Iterations for 100 images

The following graphs will give the idea about the comparison of these two models

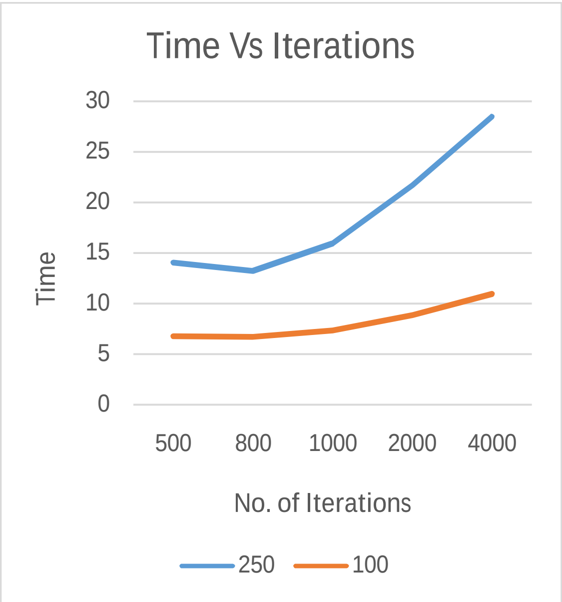

Figure 11: Graph of time vs iterations for 250 and 100 Images

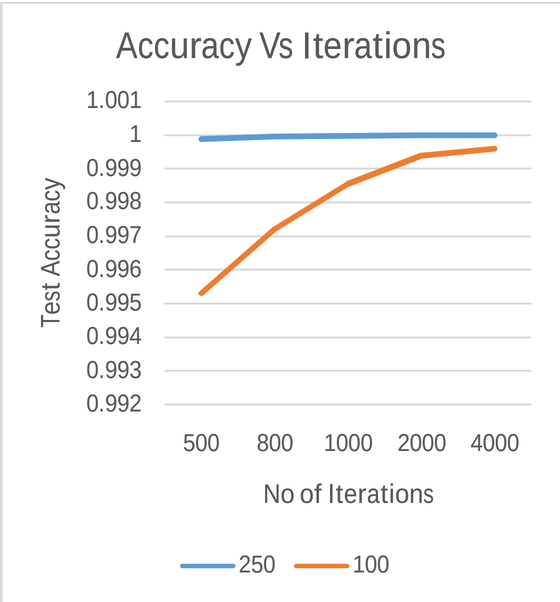

Figure 12: Graph of accuracy vs Iterations for 250 and 100 images

As from the graphs we can say that 250 images show good accuracy than 100 images. For the number of iterations/steps the accuracy remains almost same. But the time required is increasing. As in our application time is

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

more important and some compromise in accuracy is okay. Hence, we chose a model of 250 images 1000 steps with 95 % accuracy

VII. SCREENSHOTS OF THE APPLICATION

1] Level 1: Learning A to Z

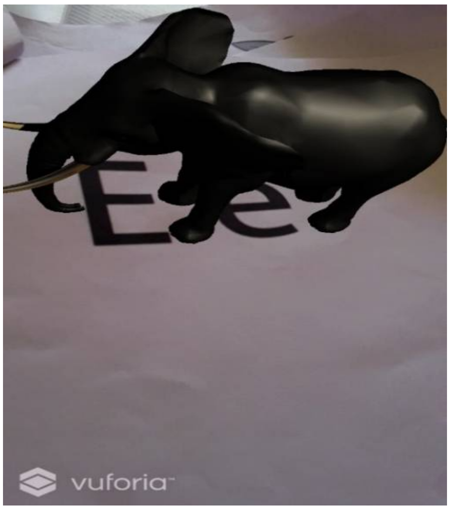

Figure 13: Augmentation on screen for alphabet E

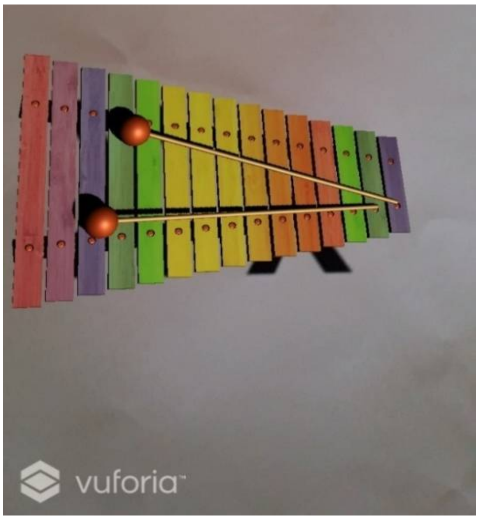

Figure 14: Augmentation on screen for alphabet X

2] Level 2: Detecting objects

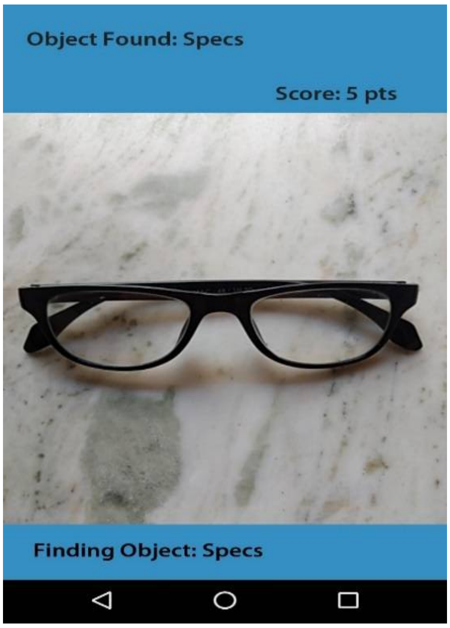

Figure 15: Object detection for Specs

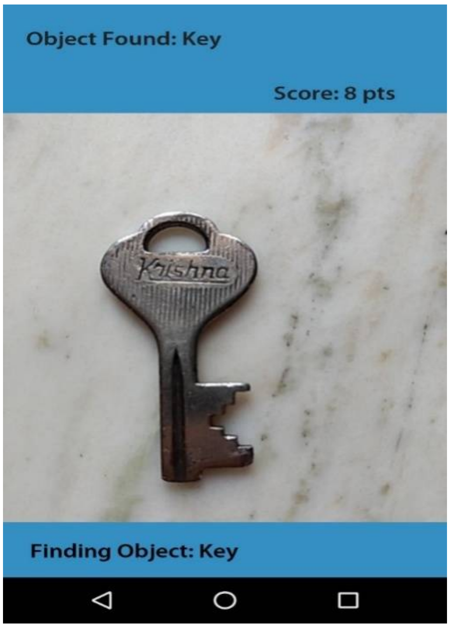

Figure 16: Object detection for key

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 2, February 2018

VIII. CONCLUSION

Alphabetterapplication applies image detection and augmentation techniques which makes learning interesting and the student can learn it by himself. The proposed application is separated into two modules which are first level built using Unity 3D in Csharp using Vuforia as the database which augment 3d object on the mobile screen and 2ndlevel in Python language using TensorFlow software which helps the student to detect the object in real time. The performance of the application is dependent upon lighting condition of the environment where the application is being used. The real time environment that the learner use to detect an image causes the accuracy rate of image detection. One more factor on which accuracy depends is the way in which the learner is scanning the object/image. If the image/object is scanned half or in a different angle the object/image might look different to the model and the results will be different. For instance, when the intensity of light is too low, the live image might not be matched with the image in the database or else if the object/image is scanned partially only will create errors in the results. For both the modules the light intensity should be good and the images/object should be scanned fully for correct detection. The current limitation of this implementation is mainly on scalability. This is because the application has to query all images from the database for checking the pattern similarity with the live image from mobile camera. Alphabetter will take time depending on the number of images in the database. The future work of Alphabetter application, therefore, includes faster detection method, better matching algorithms that can improve the overall searching performance. The scope can be further increased by using such system for higher educational levels. Like in 10th standard for studying the structure of organs as they can't be understood well through the 2d images. Even this system can be used at engineering level for studying graphics.

REFERENCES

[1] ZhongyangZheng, Bo Wang, Yakun Wang, Shuang Yang, Zhongqian Dong Tianyang Yi, Cyrus Choi, Emily J. Chang, Edward Y. Chang,"Aristo: An Augmented Reality Platform for Immersion and Interactivity",MM'17 Proceedingof the 2017 ACM on Multimedia Conference, pp. 690-698, 2017.

[2] Martin Abadi, Paul Barham, Jianmin Chen, Sanjay Ghemawat,"TensorFlow: A system for large-scale machine learning",OSDI'16 Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation,pp.265-283, 2016.

[3] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", Preliminary White Paper byGoogle Inc, 2017.

[4] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens,ZbigniewWojna,"Rethinking the Inception Architecture for Computer Vision", IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818-2826, 2016.

[5] Matthew Browne and SaeedShiryGhidary,"Convolutional Neural Networks for Image Processing: An Application in Robot Vision", AI 2003: Advances in Artificial Intelligence, Vol 2903, pp. 641–652, 2003.

[6] https://www.vi-mm.eu/2017/08/23/usage-of-augmented-reality-in-education-process/[Accessed on January 2018].

[7] https://www.layar.com/[Accessed on January 2018].

[8] Sung Lae Kim, Hae Jung Suk, JeongHwa Kang, Jun Mo Jung, Teemu H. Laine, JoonasWestli, "Using Unity 3D to Facilitate Mobile Augmented Reality Game Development", 2014 IEEE World Forum on Internet of Things, pp. 21-26, 2014.

[9] Martin Abadi, AshishAgarwal, Paul Barham, Eugene Brevdo, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems",Preliminary White Paper byGoogle Inc, pp. 1-19, Nov 2015.

[10] SaimaJawad, AzharHabibt and Babar Ali, "Enhanced Interactive Learning using Augmented Reality", 2014 IEEE17th International Multi-Topic Conference,pp. 272-276, 2014.

[11] Young Jong Mo, Joongheon Kim, Jong-Kook Kim, Aziz Mohaisen, and Woojoo Lee, "Performance of Deep Learning Computation with TensorFlow Software Library in GPU-Capable Multi-Core Computing Platforms", 9th International Conference on Ubiquitous and Future Networks,pp. 240-242,2017.

[12] DeepikaJaswal,Sowmya V, K.P Soman, "Image Classification Using Convolutional Neural Networks",International Journal of Advancements in Research & Technology, Volume 5, Issue 6, pp. 1661-1668, June-2014.