CUSTOMIZED FACE CLASSIFICATION TO REDUCE THE TRAINING TIME: A PROPOSAL

Raghurama*, Gururaja Rao P#, Rajesh Poojary*, Gajendra S Patagara*, B. Neelima $
*$ NMAM Institute of Technology, Nitte, Udupi, Karnataka, India, 574 110
#Senior Software Engineer, MulticoreWare India Pvt. Ltd., Block #3, Ground floor, DLF IT Park, manapakkam, Chennai, India - 600 089
*raghuramraop@gmail.com; #gururaja@multicorewareinc.com; * rajeshpoojary18@gmail.com; *gajendra.st@gmail.com
$ neelimareddy@nitte.edu.in;

***Abstract*-This paper proposes a new model for training using Convolution Neural Network (CNN) to reduce the training time. The intense computations involved in the CNN layers make it very computationally expensive job. The proposed model is customized for face classification. Face classification is the basic and first step of face recognition. However, face classification is not straightforward because it has to deal with different variation of images, such as face orientation, illuminating condition and facial expression. Further CNN has intensive computations and are data parallel in nature, the paper has used Graphics Processing Unit (GPU) for getting high performance. in the proposed work in this paper gives a new model for convolution neural network for face classification and compared it with the existing models such as Alexnet, VGGnet and overfeat in terms of training accuracy, validation accuracy and training time. The proposed model takes less time for training while maintaining the same training and validation accuracies.**

***Keywords*—**Face Classification, Convolution Neural Networks (CNN), Machine Learning, Face Recognition, Graphics Processing Unit (GPU).

I. INTRODUCTION

Machine learning is a science of providing computers with the ability to learn without being explicitly programmed. It deals with computer programs that can grow and change themselves when exposed to new data. Convolution neural networks (CNN) are inspired by the structure of visual system. CNN can be used in dealing with problems like face detection, face classification, emotion detection and many more. The close connection and spatial formation between layers of CNN make them suitable for image processing and understanding the images

.

The major layers used in CNN for object classification are convolution layer, pooling layer and fully connected layer as explained below:

- *Convolution layer*: This is the most important layer of CNN. It performs dot product of weights and small region of input volume [7].
- *Pooling layer*: It is used to reduce the spatial size and the amount of parameters of the network. Thus it helps in reducing the computation and also, helps avoiding overfitting [7].
- *Fully connected layer*: It has full connection to all activations of previous layer and are used to compute the class scores and hence classification [7].

II. CLASSIFICATION USING CONVOLUTION NEURAL NETWORKS

This section gives the background details of the different classifications using convolution neural networks.

A. Object classification using CNN

Object classification is one of the fundamental challenges in Computer Vision [1]. The technique of classifying objects detected in the image into different classes is known as object classification.

Even though object classification has been the area of interest for many researchers for decades, the accuracy obtained was not upto the mark. Recent approaches for classification of object using CNN improved the accuracy by a large extent. Also CNN can be used with the low quality images for training and still come up with good accuracy [2].

B. Face classification using Convolution Neural Network

Face classification is one of the popular problems studied by the researchers. But the existing face classification algorithms can only deal with identification of near frontal faces.

The CNN is inspired by the robustness of the human visual system in identifying different objects under different circumstances. However training a CNN involves multiple tasks such as pre-processing, segmentation, feature extraction for a good performance [3].

C. General classification algorithm of CNN

The general classification steps in CNN are as follows:

Step-1: The neural network (CNN model) consisting of various layers like convolution layers, pooling layers, fully connected layer is initialized with either pre-trained checkpoints or with random numbers.

Step 2: The CNN is trained using labeled images from the training data set. The network gets trained or learns using forward and backward propagation algorithms.

Step 3: For each epoch (epoch is one round of training the network using forward-backward propagation algorithm for all the input images in training data set), a forward propagation algorithm is used to validate the accuracy of the network for that epoch.

Step 4: Step 2-3 are repeated till the network achieves desired or maximum accuracy.

Step 5: Once after training CNN, a forward propagation algorithm is used for entirely new set of test data to check the correctness or accuracy of the trained network.

III. REFERENCE MODELS USED FOR FACE CLASSIFICATION

This section gives the details of the CNN modles used as reference modle to compare performance of the proposed model. The reference model details are given in this section. Imagenet winner models are one of the best models used for object classification. Some of the most accurate models from imagenet are used for comparing the accuracy of proposed model that are explained as below:

A. Alex net

Alexnet consists of five convolution layers, three pooling layers and one fully connected layer. In fully connected layer, it has two non-linear rectification layers and logsoftmax loss layer for the score computation. It uses ReLU as activation function [4].

B. VGG Net

VGG net can have a stack of Convolution Layers followed by three fully connected layers. First two fully connected layers have 4096 channels each and the last fully connected layer consists of “n” channels where n is the number of classes that are to be classified. Activation function used in VGG net is ReLu.VGG net incorporates three non-linear rectification layers instead of single one, which makes the decision function more discriminative [5].

C. Overfeat

Overfeat uses multi-scale and sliding window concept efficiently inside the Convolution Neural Network. [6].

IV. PROPOSED MODEL FOR FACE CLASSIFICATION

This section details the propsed model in this paper. As mentioned previously, the proposed model is customized for face classification. It consists of three convolution layers, three pooling layers and a fully connected layer. The complete convolve layer comprises of three spatial convolution layers, three maximum pooling layers along with ReLU activation function. The proposed model uses ReLU since it runs faster on large data sets compared to traditional activation functions like ‘tanh’ or ‘sigmoid’. It also helps in reducing the chances of overfitting. Fully connected layer consists of non-linear layers, regularization layer and loss functions. There are two non-linear rectification layers in the proposed model which is used to get better scores. Also, dropout layer with probability of 0.5 is used for regularization, that helps in avoiding overfitting of the model. The last layer of fully connected layer consists of ‘logsoftmax’ loss function.

Torch-7 framework is used for all the layer implementations as their forward-backward algorithms can use GPUs for parallel computation of all those layers.

V. INPUT DATA SET FOR FACE CLASSIFICATION

Data set used in face detection consists of images randomly chosen from internet. Some of the images contain human faces and the rest do not. Images were classified into 3 categories Training, Testing and Validation. All of these categories were then sub-categorized as face and no-face. All the images used in the dataset are resized to 256x256 in order to achieve least

data loading and adjusting time. Bi-linear image scaling algorithm is used for resizing all these images. Finally the input image for the model is resized and cropped randomly to 224x224 to train the network effectively.

VI. RESULTS AND ANALYSIS

This Section gives the details of experimental set-up along with the results and analysis of the results.

A. Experimental setup

All the experiments are done in the same system having system configuration as follows: Ubuntu 16.04 Operating System, 4 GB RAM. The GPU used is Nvidia GeForce GTX Titan X with compute capability 5.2 and cuda 7.5. All our training, validation and testing are run on GPU. Torch 7 machine learning framework is used for the training and testing, as it has inbuilt functions that can run the computations of training and testing in GPUs using CUDA.

Each model is trained for 12 epochs and corresponding train and test accuracy are calculated.

B. Results and analysis

All the models achieve a training accuracy of ~95% and testing accuracy of ~90%. Figure 1 shows the accuracy of various models after training for 12 epochs.

Model	Train Accuracy (%)	Validation Accuracy (%)	Test accuracy (%)
Alex net	96.79	96.28	90.04
Overfeat	96.36	94.89	90.90
VGG	96.06	95.49	88.21
Proposed Model	95.23	95.04	89.55

Fig. 1. Train, validation and test accuracy after 12 epochs of training for face classification

The biggest challenge of training CNN is overfitting. Overfitting is a phenomenon where the model gets over tuned for the training dataset and thus resulting is very high training accuracy and low testing accuracy. Based on the accuracy values of all models from figure 1, it is clear that overfitting was not met in any of the models.

Figure 2 shows the improvement in the accuracy of various models over the epochs during training. It is obtained by plotting epochs along X-axis and train accuracy for each epoch along Y-axis.

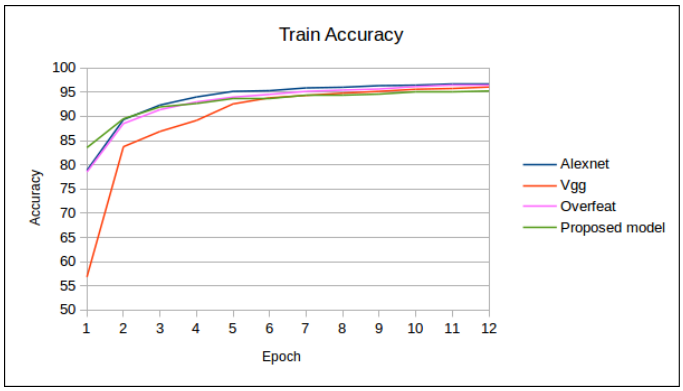

Fig. 2. Training accuracy of various models for first 12 epochs

The proposed model was mainly used to reduce the training and testing time while maintaining the same accuracy. Figure 3 shows the average time taken for training, validation and testing per epoch by each model. Clearly the proposed model takes the least time while maintaining the accuracy same as standard reference models.

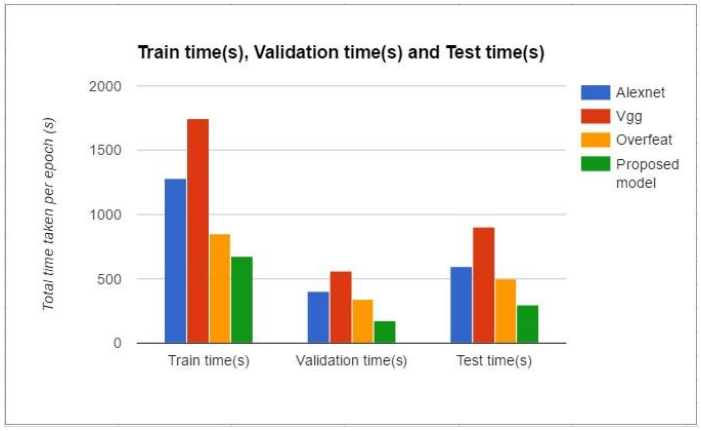

Fig. 3. Total time taken for training, validation and testing per epoch by various models.

The input image for the forward and backward propagation algorithm of all the models is of size 224x224. The average time taken by each image for both forward and backward propagation is shown in Figure 4. The proposed model takes least time of all the models tested.

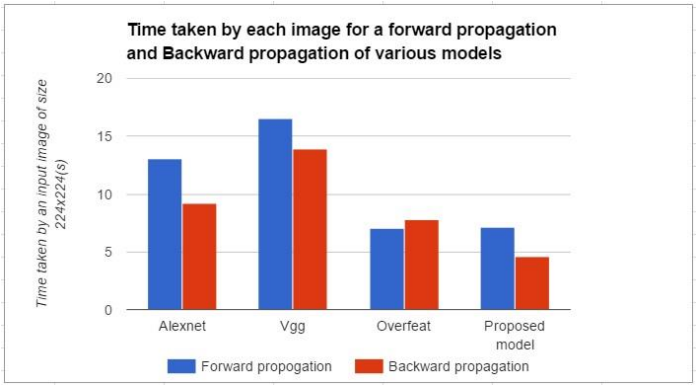

Fig. 4. Average time taken by an input image of size 224x224 for forward and backward propagation for various models discussed.

Figure 5 shows the speedup of forward and backward propagation per image of proposed model with respect to various reference models.

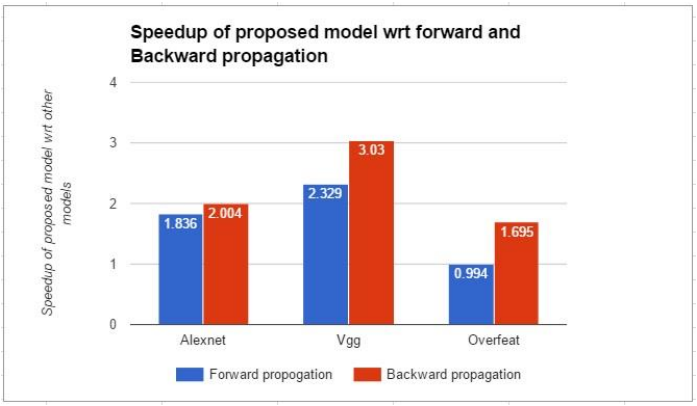

Fig. 5. Speedup of forward and backward propagation phases of proposed model with respect to reference models

The proposed model gives speedup of 1.94x, 2.67x and 1.17x for complete training phase with respect to reference models alexnet, vgg and overfeat respectively.

VII. CONCLUSION

In this paper, a simple CNN model for face classification is provided and it's accuracy is compared with various CNN classification winner models. Also, the model is designed to achieve higher speedup compared to various other standard models. All the models are tested for accuracy using same set of train, validation and test data set in same test platform. In every model, the input image of 256x256 is used as input to the application and cropped image of 224x224 is provided to the forward and backward propagation of the model. Torch 7 framework is used for the design and training of all the models.

The proposed model achieves almost same accuracy compared to various imagenet winner models for face classification. Also, the proposed model gives 1x to 2.4x speedup in forward propagation and 1.7x to 3x in backward propagation with respect to reference models.

The following future work is planned for the proposed work as follows:

- The proposed model can be tested and compared for the accuracy after training for 50-100 epochs. A better accuracy is expected by the models when trained for more epochs.
- The proposed model can be tuned further to achieve better accuracy with less training and testing time.
- The proposed model can be modified to achieve multi-GPU usage and thus enhancing the speedup.
- The CNN layers used in the proposed model can be optimized further to achieve higher speedup.
- The proposed model can be tuned to detect multiple objects in single image.

VIII. ACKNOWLEDGEMENT

The basic classification code for various imagenet winning models is taken from Soumith's "Training and object classifier in torch-7 on multiple GPUs over imagenet"[8]. This application is used for training the classifier over imagenet data consisting of millions of data from thousands of classes, using multiple GPUs in torch-7 framework.

REFERENCES

[1]Ross Girshik, Jeff Donahue, Trevor Darrell and Jitendra Malik, "Region-based Convolutional networks for Accurate Object Detection and Segmentation"

[2]Hao Jiang and Shiquan Wang, "Object Detection and Counting with Low Quality Videos"

[3]A.R.Syafeeza, M.Khalil-Hani, S.S.Liew, R.Bakhteri, "Convolutional Neural Network for Face Recognition with Pose and Illumination Variation"

[4]Alex Krizhevsky, Ilay Sutskever and Geoffrey E.Hinton, "ImageNet Classification with Deep Convolutional Neural Networks"

[5]Karen Simonyan and Andrew Zisserman , “Very Deep Convolutional Network for Large-Scale Image Recognition”
[6]Pierre Sermanet, David Eigen, Xiang Zhang, Michel Mathieu, Rob Fergus and Yann LeCun, “Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Network”
[7]http://cs231n.github.io/convolutional-networks
[8]https://github.com/soumith/imagenet-multiGPU.torch
[9]Subrath Kumar Rath, Siddharth Swarup Rautaray, “A Survey on Face Detection and Recognition Techniques in Different Application Domain”