Project : Robotic Inference

Goutam Gupta

Abstract—In this project, Inference model based on deep neural network is built on the supplied data for image classification. Standards networks like AlexNet and Googlenet are used on the Nvidia DIGITS server for training the network. Data is collected for classification of Indian currency notes and also for the classification of electronic components.

Index Terms—Robot, IEEEtran, Udacity, Inference, Image classification, Deep neural Networks

1 Introduction

THE most challenging task for robots is to infer objects in its surrounding. The task of robotic inference gives the Robot necessary intelligence required for perception and decision making. The task of image classification and semantic segmentation can be easily achieved with the use of deep neural network. Very deep neural networks can be built and trained on powerful GPUs and the same model can be deployed on embedded robotic platforms e.g Nvidia Jetson for real time robotic inference applications. The application of inference can range from industrial domain e.g. inventory management, home automation etc. to military domain e.g. in target detection applications. In this project, a network model is built upon the supplied data on Nvidia DIGITS and deployed on Nvidia Jetson TX2 kit to test real time robotic inference. Data is also collected for building two inference applications. One of the application can detect Indian currency notes and the other application helps in classifying objects used in electronic laboratories.

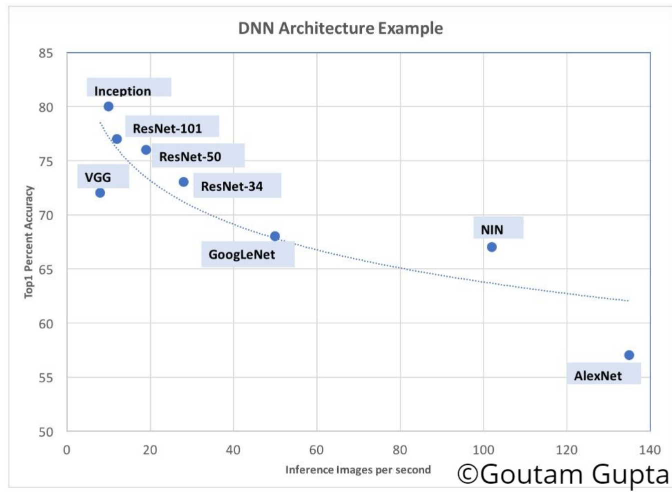

Fig. 1. DNN architecture Example.

2 Background

Nvidia DIGITS server is used for Network training and model building. DIGITS simplifies common deep learning tasks such as managing data, designing and training neural networks on multi-GPU systems, monitoring performance in real time with advanced visualizations, and selecting the best performing model from the results browser for deployment.The work-space provide three types of DNNs i.e LeNet, AlexNet and Google-Net. We need a color image for all the image classification tasks in the supplied data sets as well as in the collected data sets. Inference speed should be fast in all the three applications. Segregation of currency notes or the classification of lab components, both requires higher accuracy on real time use cases. A fps of more than 50 looks to be suitable for all these applications. As presented in the Canziani analysis above, two conclusions are helpful when considering the speed and accuracy trade-off for a robotic systems design that uses DNN inference on a deployed platform such as the Jetson.Accuracy and inference time are in a hyperbolic relationship.Experiments sows that performance of the GoogleNet architecture for the chosen data sets is best in terms of efficiency (accuracy per operations as well as inference speed). As evident from above analysis, GoogleNet gives a accuracy of 68% while maintaining an inference speed of 50fps. After a few iteration, Google-Net model is selected for modelling the network for supplied data as well as for captured data sets for both the applications. The parameters are tuned to achieve best accuracy. SGD for the optimiser, 0.002 for base learning rate and no of epochs as 60 (for electronic components data sets) and 40(for Indian currency classification), are some of the parameters which were fixed after running through few cycles.

3 Data Acquisition

3.1 Supplied Dataset

The supplied dataset consists of photos taken from a Jetson mounted over a conveyor belt.We are training pictures of candy boxes, bottles, and nothing (empty conveyor belt) for the purpose of real time sorting. This kind of design can be extrapolated to many things that require real time sorting. The data looks as shown in figure 2 :

3.2 Collected Datasets

Data sets were collected for building two robotic inference based applications. The first application is about sorting components in electronic laboratories. Data is collected with the USB camera connected to Nvidia Jetson Kit. Data is

Fig. 2. Supplied dataset by Udacity.

categorised into four categories : Black_Wire, Red_Wire, LED and Resistor. The data looks as shown in figure 3 :

Fig. 3. Captured dataset for lab components classification.

Another application is built for classification of Indian currency notes. Four categories were created : INR-2000, INR-500 INR-200 and INR-100. The data looks as shown in figure 4:

Fig. 4. Captured dataset for indian currency classification.

4 Results

4.1 Supplied Dataset

Model is trained on Digits Server using the supplied data sets. The accuracy of the network is close to 100% as shown in the figure 5. GoogleNet model available on the DIGITS server is used and the no of epochs is set to 8. An "evaluation" command provided by Udacity was used for testing the Network. The maximum inference time was 5.0121ms and the accuracy was 75.40%, which met the requirement of 10ms and 75% respectively. The result is shown in the figure 6.

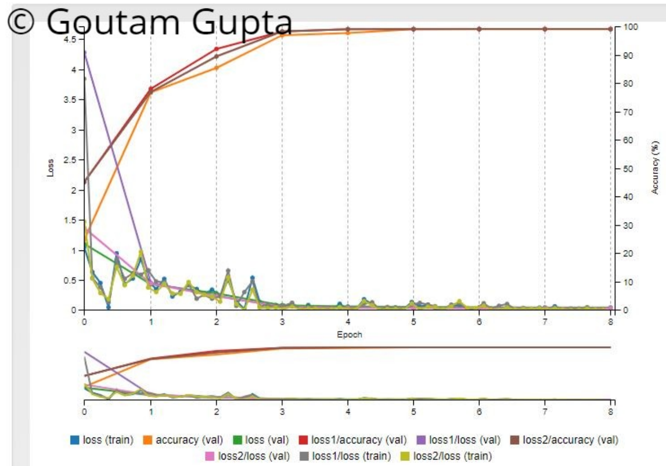

Fig. 5. Network Model on Udacity supplied dataset.

```
Average over 10 runs is 5.56931 ms.
Average over 10 runs is 5.56771 ms.
Average over 10 runs is 5.57014 ms.
Average over 10 runs is 5.02901 ms.
Average over 10 runs is 5.01231 ms.

Calculating model accuacy...

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 14667  100 12351  100  2316    211     39  0:00:59  0:00:58  0:00:01  2325

Your model accuacy is 75.4098360656 %
root@20414904bcb1:/home/workspace# 
```


Fig. 6. Evaluation of Network model on Udacity supplied dataset.

Model is tested on various images from Google. One of the test condition is shown in the figure 7. It is able to detect a bottle with 98.92% accuracy.

Fig. 7. Test : Model of Supplied dataset.

4.2 Captured Dataset

4.2.1 *Classification of Electronic components*

Data is captured for various electronic components in the lab. Four major classes were created : Black_Wire, Red_Wire,LED and Resistor. Network is trained on DIGITS server using GoogleNet model. No of of Epochs are set to 60. The network achieves an accuracy of as shown in the figure 8.

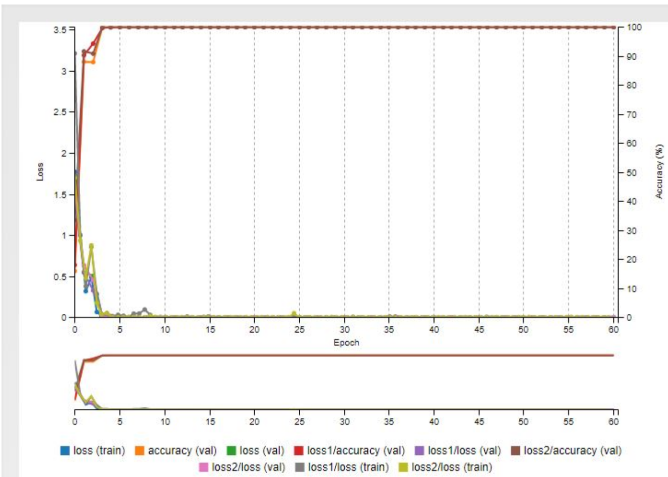

Fig. 8. Network Model on captured datasets of lab components.

Model is tested on various images from Google. One of test condition is shown in the figure 9. It is able to detect a Resistor with 98.92% accuracy. The same model is deployed

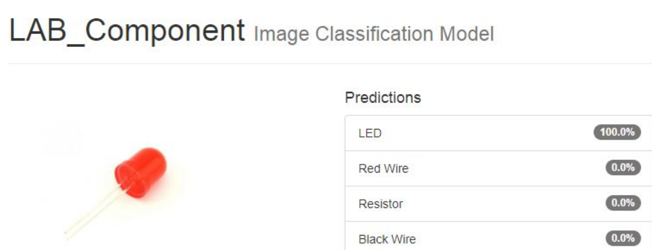

Fig. 9. Test : Model of Captured dataset : Lab components.

on Jetson EV kit and real time use case was tested with a black cable as shown in the figure 10.

Fig. 10. Test : Model of Supplied dataset.

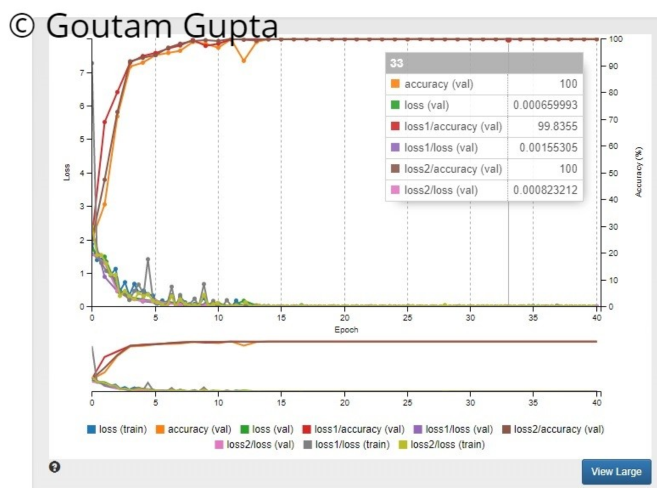

Fig. 11. Network Model on captured datasets of Indian Currency.

4.2.2 *Classification of Indian Currency*

Data is captured for different Indian Currency notes. Four major classes were created : INR_2000, INR_500,INR_200 and INR_100. Network is trained on DIGITS server using GoogleNet model. No. of Epochs are set to 40. The network achieves an accuracy of as shown in the figure 11.

Model is tested on various images from Google. One of test condition is shown in the figure 12. It is able to detect a 2000 Rupee note with 100% accuracy. The same model is deployed on Jetson EV kit and real time use case was tested with a black cable as shown in the figure 13.

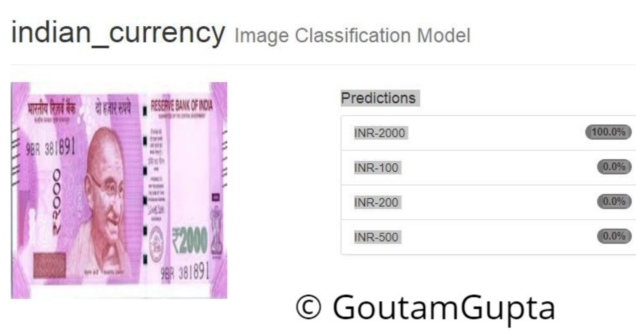

Fig. 12. Test : Model of Captured dataset : Indian Currency.

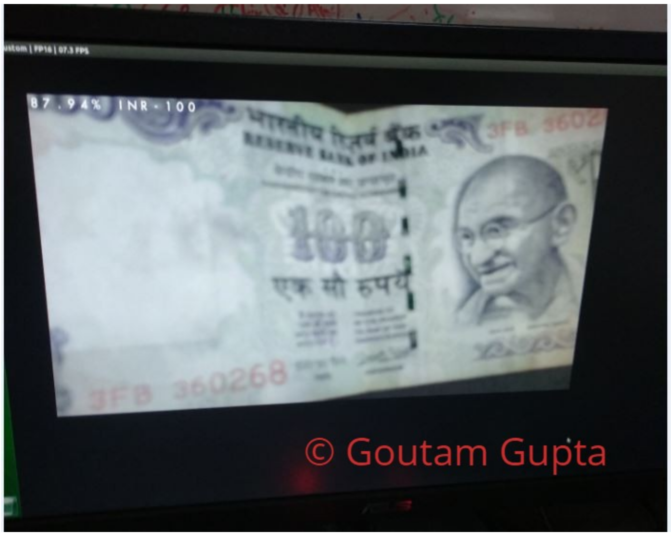

Fig. 13. Test : Model of Supplied dataset.

5 Discussion

As evident from above results, the accuracy of the network model on collected data set is not great particularly for the classification of Indian currencies. This is because of various reasons as listed below:

- Less number of collected data for each image class
- Less variation in the collected data sets.
- Images with lower resolution

The network can be further tuned to increase the accuracy of models on captured data set. This will also increase the accuracy of inference on real time use cases for unseen objects.

There was a big improvement in obtained accuracy when comparing GoogleNet architecture to AlexNet, however the required training time and hence the inference time is longer in GoogleNet as compared to AlexNet network. For our application, particularly the one where we need to classify various currency, accuracy is more important than the inference time and hence the selection of GoogleNet architecture justifies the case. Counting and evaluation of currency in an automated counting system should not go wrong in real time use cases. Moreover, Inference time can be further improved using optimised hardware. Accuracy can be further improved by adding more data per class but it will come at the cost of increased inference time as well as larger storage space.

6 Conclusion / Future work

Inference is performed on Udacity supplied data sets as well as on captured data sets. The network is built on DIGITS server using GoogleNet which works better for supplied data set. However, the network's accuracy can be further enhanced on the captured data set for both the robotic inference application : Classification of lab components and the classification of Indian currencies.Capturing large number of data sets with different features would help in increasing the accuracy of the network. The network can be also optimised by tuning various parameters like no of epochs, learning rate and by customising the deep neural network architectures.For example, few additional pooling layer or Convolutional layer can be introduced in GoogleNet architecture for better accuracy.