Robotic Inference using NVIDIA Digits

Pritesh Gudge

Abstract—The Inference project is a part of the Robotics Nanodegree. The objective was to understand and implement a prototype of vision system for a robotics application. Two implementations are provided for evaluation. First, the provided dataset of bottles, boxes and other images is used to train a Google Lenet [1] based network and the results are used to classify relevant images provided in the test suite. Second, data collection, data conditioning and Google Lenet based model training is used to train a network for food classification.

Index Terms—Robot, Inference, IEEEtran, Udacity, Deep Learning.

1 Introduction

THE progress in the field of robotics is accelarating every day. Robots are being used in many applications like manufacturing, search and rescue, exploration etc. Many of the applications require identifying and detecting objects in multiple scenes and backgrounds. Deep learning is being leveraged into robotic vision and inference as it provides a framework for robots to be configured to operate in multiple environments both natural and man-made. Convolutional and reinforcement learning based techniques have been most effective in robotics applications. Previous approaches to robotic inference were based line follower and other approaches for structured operating environments [2]. Deep learning based approaches provide the possibility to provide identification and detection for robotic inference, given that adequate data is available for training the neural networks.

In the first part, the provided images of boxes and bottles are used to train an Google Lenet based network for classification. In the second part, a model is trained to classify between fruits and other foods. One can build a diet-tracking or junk food alert robotic system using such a classification model, possibly built into a refrigerator. It generates an alert if unhealthy food is taken out or stored in the refrigerator. A similar model can be used in a robot which sorts items purchased while grocery shopping. Eg. Bananas should not put into the refrigerator, while other fruits can be.

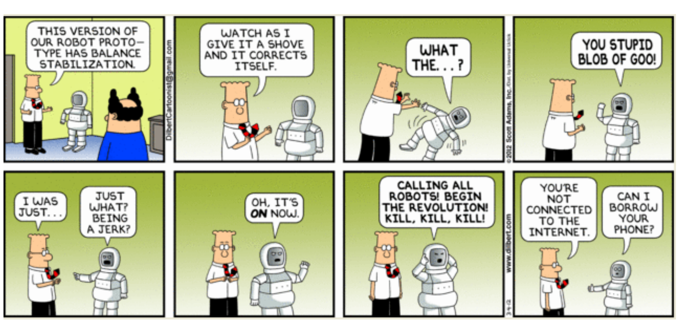

Fig. 1. Robot Revolution.

1.1 Subsection Heading Here

Subsection text here.

1.1.1 Subsubsection Heading Here

Subsubsection text here.

TABLE 1
Table

One	Two
Three	Four

2 Background / Formulation

The **NVIDIA Digits** environment - shell and GUI was used for structuring the databases and training, validation of the models.

2.1 Boxes vs Bottles

For the first part of the project, the images provided were color RGB images with size 256x256. For simplicity purposes, the networks provided in the digits workspace were used. Two pre-defined networks are available in the Digits workspace which take 256x256 RGB images as input. The training was done with 5 epochs first with Google LeNet with Adam Optimizer. Google LeNet was chosen as it offers state of the art results for ImageNet dataset and as it offers comparatively higher quality in classification results [1] at the cost of slightly higher compuational requirements . GPUs were available for training hence quality was chosen over lower computational requirements in the tradeoff. Adam Optimizer was chosen as it has been established to be more efficient compared to other optimizers [3]. The model with Adam Optimizer failed to converge.

In the next step the same Google LeNet network was used with standard SGD optimizer with 5 epochs and 0.01 learning rate at the start. 25% of the dataset was used for validation. The test dataset was stored separately before training. Once the model was found to be converging, the final model was set to train with an initial learning rate of 0.01 and 10 epochs of the network 22-layer Google LeNet.

[1]. The test images were used to determine performance of the trained model. Speed of each individual classification run was also evaluated. Once trained the Google LeNet network model provides quick classification on GPU ¡10ms.

2.2 Food Classification

6020 images were collected with approximately equal number of images for each of the classes:

- Banana
- Guava
- Other foods(Marshmallows)

Google Lenet [1] model was chosen for its effectiveness to classificiation and detection on ImageNet dataset as described above 2.1. The hyperparameters: 10 epochs and a starting learning rate of 0.01 was chosen for training. 25% of the dataset was used for validation. The test dataset was stored separately before training. The test images were used to determine performance of the trained model. Once trained the Google LeNet network model provides quick classification results on GPU and hence was chosen for this project. Eg. The model can be deployed on a TX2 processor and classification results can be obtained real-time for the robotic platform in its operating state.

- example

1) example

3 Data Acquisition

3.1 Bottles and Boxes

3.2 Food Classification

This section should discuss the data set. Items to include are the number of images, size of the images, the types of images (RGB, Grayscale, etc.), how these images were collected (including the method). Providing this information is critical if anyone would like to replicate your results. After all, the intent of reports such as these are to convey information and build upon ideas so you want to ensure others can validate your process. Justifying why you gathered data in this way is a helpful point, but sometimes this may be omitted here if the problem has been stated clearly in the introduction. It is a great idea here to have at least one or two images showing what your data looks like for the reader to visualize.

4 Results

This is typically the hardest part of the report for many. You want to convey your results in an unbiased fashion. If you results are good, you can objectively note this. Similarly, you may do this if they are bad as well. You do not want to justify your results here with discussion; this is a topic for the next session. Present the results of your robotics project model and the model you used for the supplied data with the appropriate accuracy and inference time For demonstrating your results, it is incredibly useful to have some charts, tables, and/or graphs for the reader to review. This makes ingesting the information quicker and easier.

5 Discussion

This is the only section of the report where you may include your opinion. However, make sure your opinion is based on facts. If your results are poor, make mention of what may be the underlying issues. If the results are good, why do you think this is the case? Again, avoid writing in the first person (i.e. Do not use words like I or me). If you really find yourself struggling to avoid the word I or me; sometimes, this can be avoid with the use of the word one. As an example: instead of : I think the accuracy on my dataset is low because the images are too small to show the necessary detail try: one may believe the accuracy on the dataset is low because the images are too small to show the necessary detail. They say the same thing, but the second avoids the first person. Reflect on which is more important, inference time or accuracy, in regards to your robotic inference project.

6 Conclusion / Future work

This section is intended to summarize your report. Your summary should include a recap of the results, did this project achieve what you attempted, and is this a commercially viable product? For Future work,address areas of work that you may not have addressed in your report as possible next steps. For future work, this could be due to time constraints, lack of currently developed methods / technology, and areas of application outside of your current implementation. Again, avoid the use of the first-person.

References

[1] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” *CoRR*, vol. abs/1409.4842, 2014.
[2] A. Harom, “Line follower robot,” *Universiti Malaysia Perlis*, 2008.
[3] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014.
[4] L. Lamport, *LATEX: a document preparation system: user's guide and reference manual*. Addison-wesley, 1994.