Robotic Inference

Gassó Loncan, J.C.

Abstract—Thanks to the huge development technology during the last decade concerning to information, data storage and GPU Internet and so on, the Neural Networks has reemerge from the ashes with a new name, Deep Learning. In this project we make use of a tool for the development and implementation of deep neural network (DNNs) such as the NVIDIA's DIGITS. The robotic inference idea was to prototype a kind of security unlock system of electronic devices. It was selected the GoogleNet 2014 model architecture for both task. The required project specifications were achieved. A valuable experience has been gotten during this project regarding the use of NVIDIA DIGITS and data set generation.

Index Terms—Robot, Udacity, Inference, deep learning, Nvidia DIGITS.

1 Introduction

THANKS to the huge development technology during the last decade concerning to information, data storage and GPU Internet and so on, the Neural Networks has reemerge from the ashes with a new name, Deep Learning, and are now installed on our life in more aspects than we are really conscious. Deep Learning is indeed something like the prodigal son of artificial intelligence, and as such promises an incredible future in the union with robotics [1]–[3].

In this project we make use of a tool for the development and implementation of deep neural network (DNNs) such as the NVIDIA's DIGITS. This tool provide a complete framework that can be used to rapidly train the highly accurate DNNs for image classification, segmentation and object detection tasks. Furthermore DIGITS make easy to export any DNN model to a embedded platform such as the NVIDIA Jetson to deploy in close real time.

The following sections will cover aspects such as data acquisition and further management, training and model testing, and the inference time. For this purpose, the project is divided into two tasks, the first one focused on training a model on a database provided and obtaining 75% accuracy and an inference time of less than 10ms. The second is more focused on data acquisition and then training a model with meaningful results.

The robotic inference idea was to prototype a kind of security unlock system of electronic devices. So as to there were collected several captures of faces of five persons on different conditions (such hairstyle, clothes and accessories) in order to induce variability between samples. The acquired data were divide into three classes, *Correct Person*, *Incorrect Person*, *No Person*. Lastly, the purpose of the DNN will be recognise the correct person between other incorrect persons and the absence of it.

2 Formulation

It was selected the GoogleNet 2014 model architecture for both task. The main aspect that justify its selection is that their architecture is consisted of a 22 layer deep CNN but reduced the number of parameters from 60 million (AlexNet) to 4 million [4], [5]. Such reduction of the number of parameters has a direct impact to the inference time due to a reduction in the number of operations required for forward propagation. On the other hand, LeNet was ruled out because the image that will be used are RGB and LeNet input need single channel images.

The first model trained was with the data set provided. The choice of hyperparameters was based on previous experience with this type of network acquired during the course. As the results were successful and a priori the data set provided was considered more complex, it was decided to use the same configuration for the second model with the homemade data set. The hyperparameters are resumed on table below. While the first data set was split on train (75%) and validation (25%) as the test data set was also provided, the second data set was split on train (80%), validation (20%) and test (10%).

TABLE 1
Image data set main characteristics.

HYPERPARAMETERS	
Input	256x256 (Squash)
Epoch	10
Solver	Adam
Learning Rate	0.001
Policy	Step Down
Step size	10
Gamma	0.5

3 Data Acquisition

As the supplied data set for the first part of the project, the new data set was designed for a classification task between three class: *Correct Person*, *Incorrect Person*, *No Person*.

The data was acquired using a notebook web cam using the slightly modified version of Python Data Capture Script example provided in classroom. As it can be seen on Table 2, the data set it is not balanced between classes, having more examples for the Correct Person class witch is, for classifier objective the main class.

There were not implemented any formal protocol for the images acquisition. There were a total of five subject who

TABLE 2
Image data set main characteristics.

Class	Number	Size	Type
Correct Person	1051		
Incorrect Person	674	640x480px	RGB
No Person	645		
Total	**2730**		

kindly volunteered to be photographed to help carry out this project. All of them were part of the decent staff of the FI-UNER. Their identity will be kept anonymous, except for the subject labelled as correct who is the author of this work.

In order to enhance the variability of the data, the image were taken on two days from the same spot, so as to at least the clothes and people around would be different. As it can be seen on the Fig.1 the *Correct person* tagged samples has more variety, as it wear different clothes, hairstyle and accessories as cap or glasses. To have done this kind of variation with the other subjects was complicated because everyone was busy working. However, similar modifications were attempted where possible.

Finally but no least, there were a little routine asked to the volunteers. All the picture were taken manually by pushing a key of the notebook. Each subject was asked to slowly move their face from side to side and up and down as they took pictures, in order to make the classification a little more complicated. This process was done one time per *Incorrect Person* class subject.

4 Results

The required project specifications were achieved. Figure 3 shows the training evolution with the provided data set. It can be seen that the model reach a 100% of accuracy after the second epochs. On Figure 2 are shown the results of the inference evaluation of the model reaching more than 75% of accuracy and an inference time below 10ms, indeed a mean of less than 6ms. Figure 4 shows the training evolution with the generated data set. It can be seen that the model reach a 100% of accuracy after the firt epochs. On Figure 5 are shown the results of the test with a accuracy near 100% of accuracy.

5 Discussion

Despite having achieved the numerical objectives requested, the results can be clearly discussed. Compared to the first model, a test accuracy of 75% could be considered rather low. As can be seen in Figure 2, the model reached an accuracy of 100% after only 2 epochs, leaving 8 more epochs, this can lead to a case of overfitting. Some kind of early stopping could have been implemented by monitoring the validation or training with a more aggressive decay policy. The same reasoning can be made with respect to the results of the second model, but in this case it should also be noted that the data set is somewhat reduced or simply not sufficiently representative of the problem in question. It is clear that with these results it is not possible to conclude that the model can be taken to an unlock system by face recognition and it is evident that it is necessary to generate

Fig. 1. Data samples

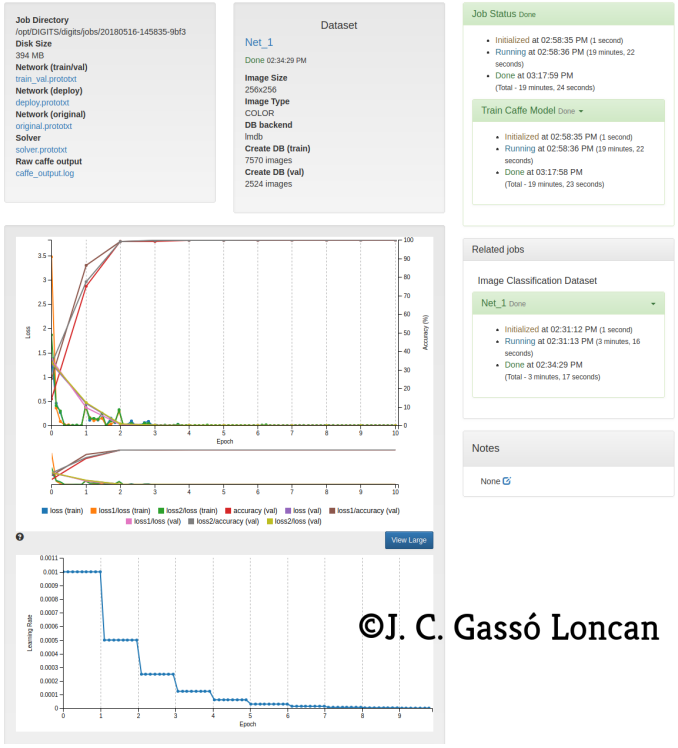

Fig. 2. Train curves of the model with the provided data set.

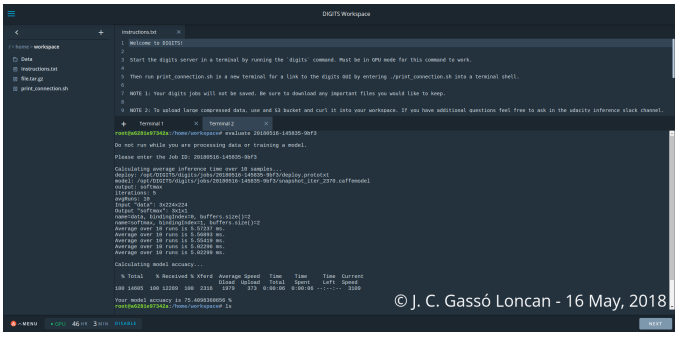

Fig. 3. Test inference result of the model with the provided data set.

a more complete and representative data set to achieve a greater generalisation.

6 Conclusion & Future work

The required project specifications were achieved. A valuable experience has been gotten during this project regarding the use such a powerful tool as it is NVIDIA DIGITS. Also was important the analysis of the models architecture provided and how important could be on a real time application. The other thing that can be highlighted is the experience and concepts gained with regard to generating a data set, which is much more complicated than simply taking pictures. It is very important to keep in mind that the generalisation and performance of the model during deployment depends directly on the quality of the data set on which it was trained.

Despite the fact that any of the model was deploy on the Jetson, some other test were done using the MNIST data

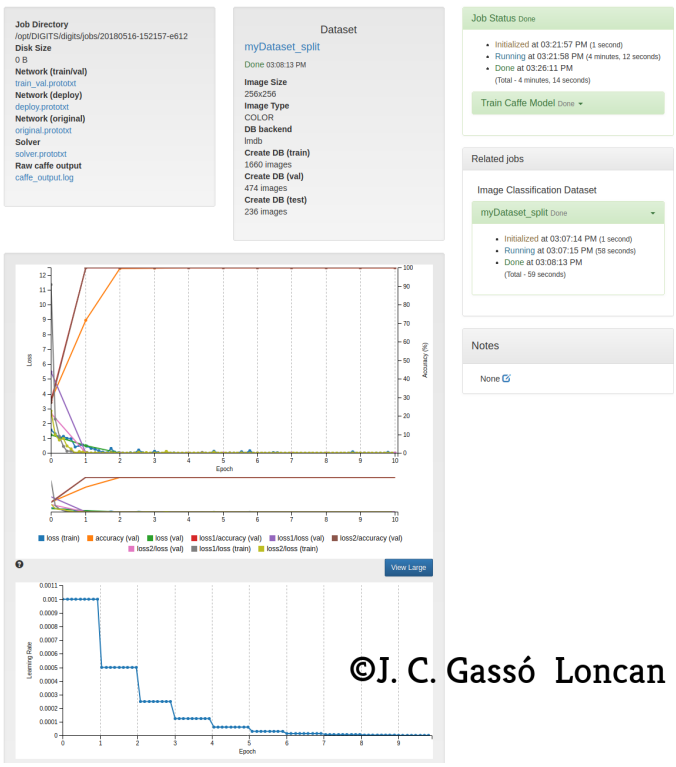

Fig. 4. Train curves of the model with the generated data set.

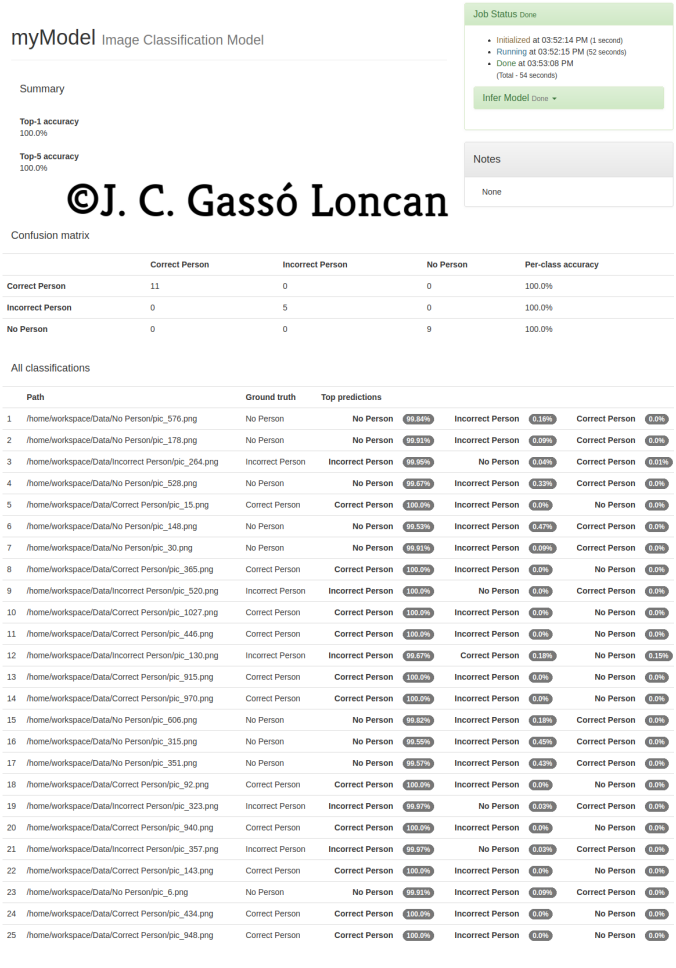

Fig. 5. Test result with the generated data set.

set training a LeNet model on DIGITS and then transferring the trained model to the Jetson using the Jetson-Inference repository.

For future work, a handwriting recognition model will be deployed on a Jetson TX2 using everything learned here. The idea is that a robotic manipulator can recognise the writing of individuals from the public and write them on glass or paper as a demo sample at a technology fair in my country as university publicity.

REFERENCES

[1] "Deep learning - with massive amounts of computational power, machines can now recognize objects and translate speech in real time. artificial intelligence is finally getting smart.." . Last update: Dic 4, 2017.

[2] "The rise of neural networks and deep learning in our everyday lives – a conversation with yoshua bengio." . Last update: Dic 4, 2017.

[3] J. D. Bosavage, "Three ways ai will impact your everyday life." . Last update: Jan 16, 2018.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[5] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, *et al.*, "Going deeper with convolutions," Cvpr, 2015.