Inference Project

Abstract:

This project implements the inference idea using NVDIA DIGITS. By designing an image classifier, an attempt was made to solve a real-world problem of automatic door unlocking. This report discusses how the data was collected and the various inference networks that were used. It is demonstrated how just about four hundred images per class result in pretty good accuracy. Various improvement techniques for this project to be commercially successful are discussed in the end.

Introduction:

Quite often, people tend to forget home keys or even loose them. In such circumstances, one has to wait for long time till someone else gets the keys. If the keys are lost, it is even more tedious, and even risky at times, to call the key maker to unlock the door. This leads one to ask the question - what if the door would automatically be unlocked for the right person? In today's world with such an advanced technology, this problem is solvable by automatic door unlocking using face recognition. To solve this problem, an image classifier is trained to recognize the images. In this project, an image classifier to classify an image in one of the three classes - two different faces or a No-face - was designed. Although much improvement is needed for security purposes, this project demonstrates satisfactory results with only a small amount of data.

Background/Formulation:

For this project, images of three classes named Mihir, Sadhana and No-face were collected using laptop's webcam. No-face class doesn't include any face and mainly includes the background. The other two classes contain images of respective person. The data set was uploaded to the Amazon S3 bucket and then curled into the workspace provided by Udacity. From this dataset, a classification dataset with 256x256 color images was created in NVDIA digits and 25% images were used for validation. For this dataset two image classifier networks available in the NVDIA DIGITS were tried – first was AlexNet and second was GoogLeNet. For both the networks that were tried, a default learning rate of 0.01 was used. After training the network, the inference models were also tested with some additional test images.

Initially, AlexNet network with five epochs was trained. This gave a validation accuracy of about 96%. Next, the same pretrained model was trained for 3 more epochs. However, the accuracy decreased to about 93%. Consequently, the prediction accuracy on the test images also decreased. After this, GoogLeNet network was trained with five epochs. This network performed poorly leading to an accuracy of only about 65%. Therefore, AlexNet network with five epochs was chosen as the final network for this project.

The third inbuilt network, LeNet, was not tried for two reasons. Firstly, original image being too large (1280x720) compared to the one required for this network (28x28), there would have been data loss. Secondly, the conversion from color to grayscale would lead to further reduction of data from original image.

Data Acquisition:

The project classifies the given image into one of the three types – either of the two persons or the background. Data for each class was collected using laptop's webcam. Each image size is 1280x720. Main reason to choose the webcam was that each image size was less than 250kB. The total images collected for classes named Mihir, Sadhana and No-face were 532, 415 and 362. Additionally, five images per class were collected for testing the inference model. As the camera would be fixed, all the images were taken with same background. For prototyping, outdoor images were not captured as it was easier to capture indoor images. Images for classes with face were captured with three variations and their mixtures – first, different face angles, second, different distance from the camera and third, partial faces at the edge of the image. For the class No-face, images were collected mostly with pure background, but also with some other materials or persons without their faces. Below are some sample images used.

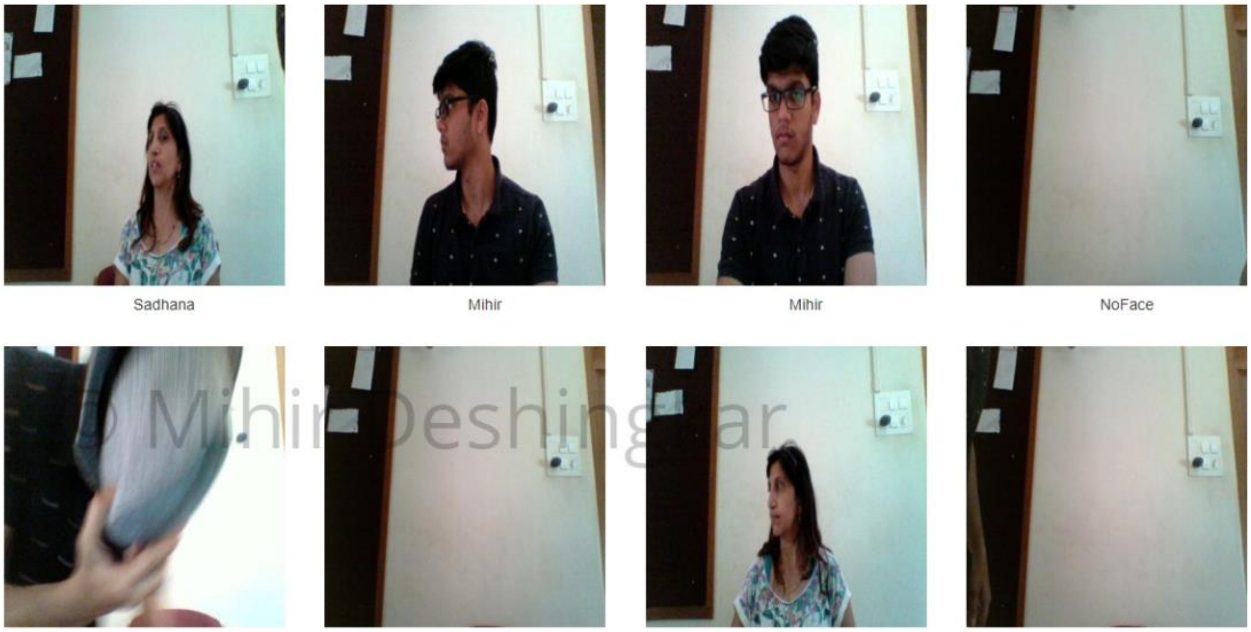


Fig. 1

Results:

Even with such a small dataset, good results were obtained. The validation accuracy was about 96%, as shown below –

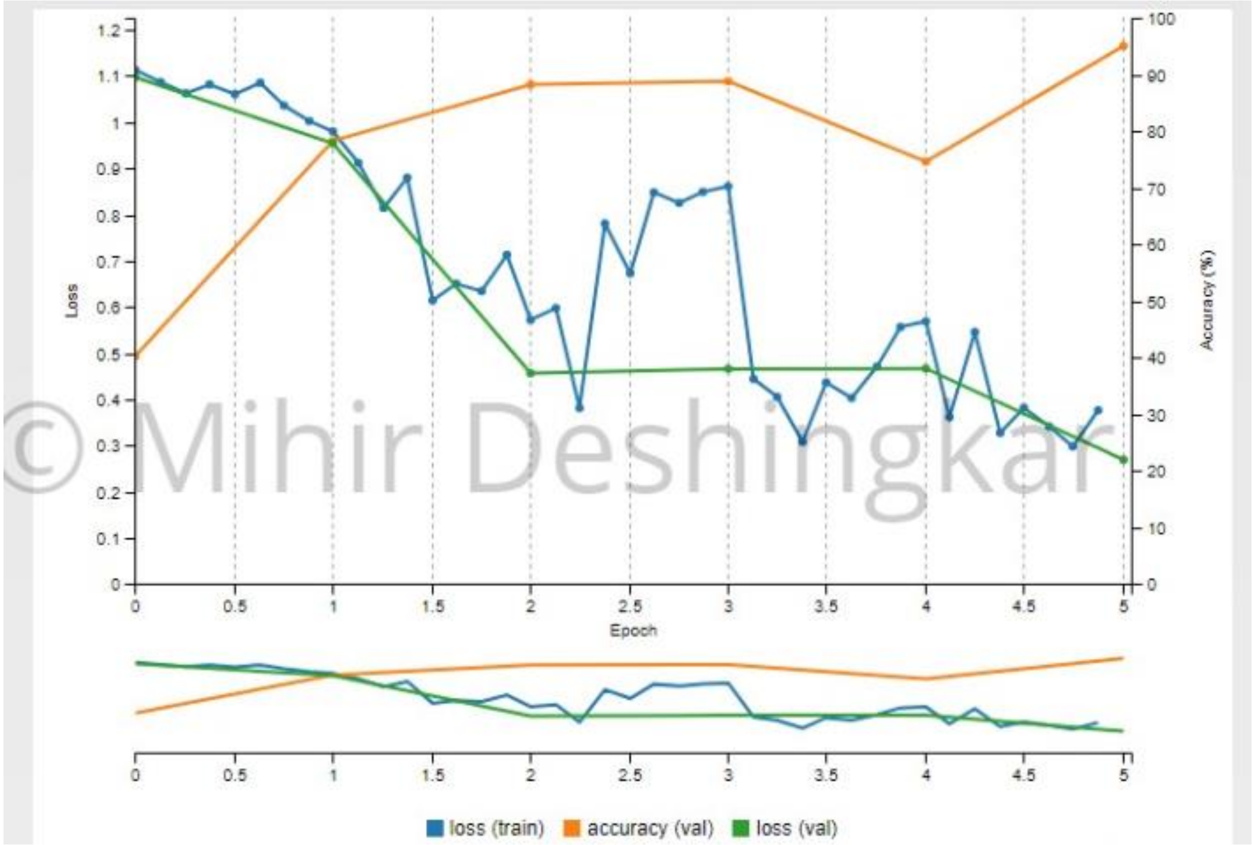
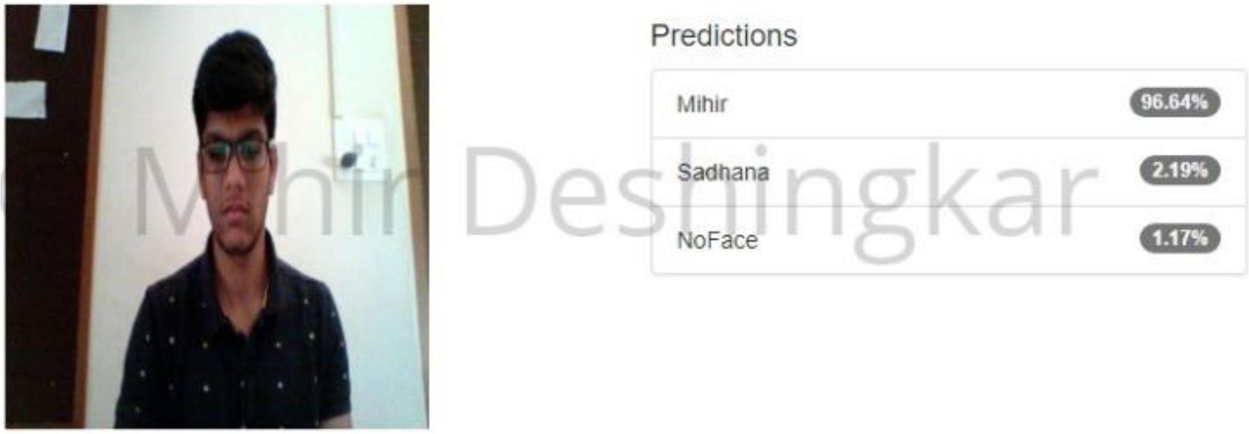


Fig. 2

The network took only 35s to train with 5 epochs.

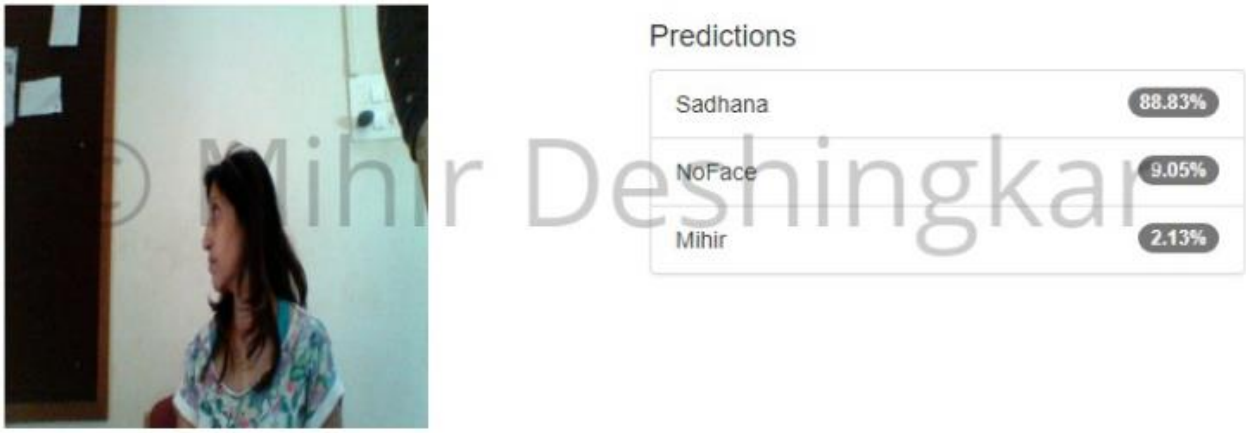
Also, the prediction with some test images was as high as 96%, and about 88% even with a side face, as shown below –



Predictions

Mihir	96.64%
Sadhana	2.19%
NoFace	1.17%

Fig. 3

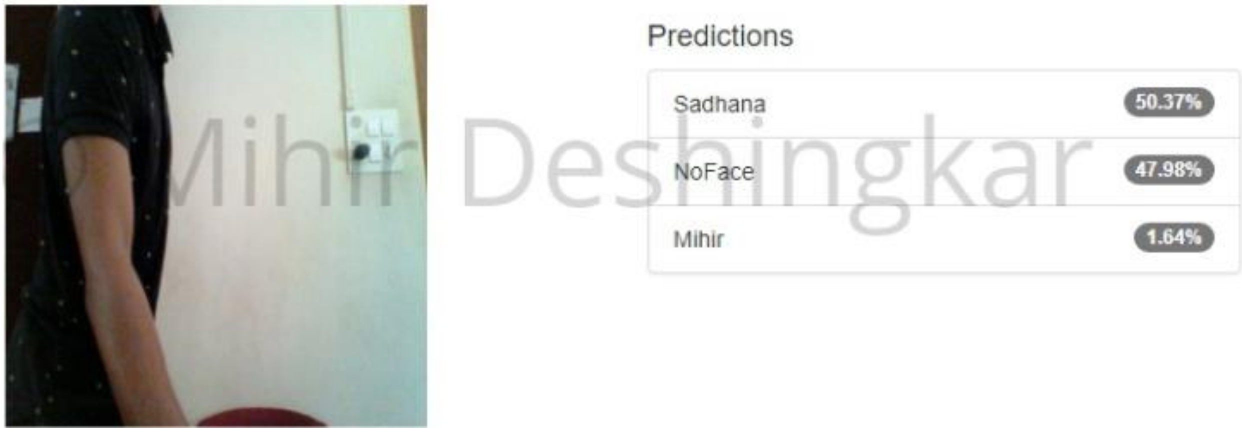


Predictions

Sadhana	88.83%
NoFace	9.05%
Mihir	2.13%

Fig. 4

However, the accuracy for No-face class without pure background was less. Below is an example where actually a No-face image is misidentified as Sadhana.

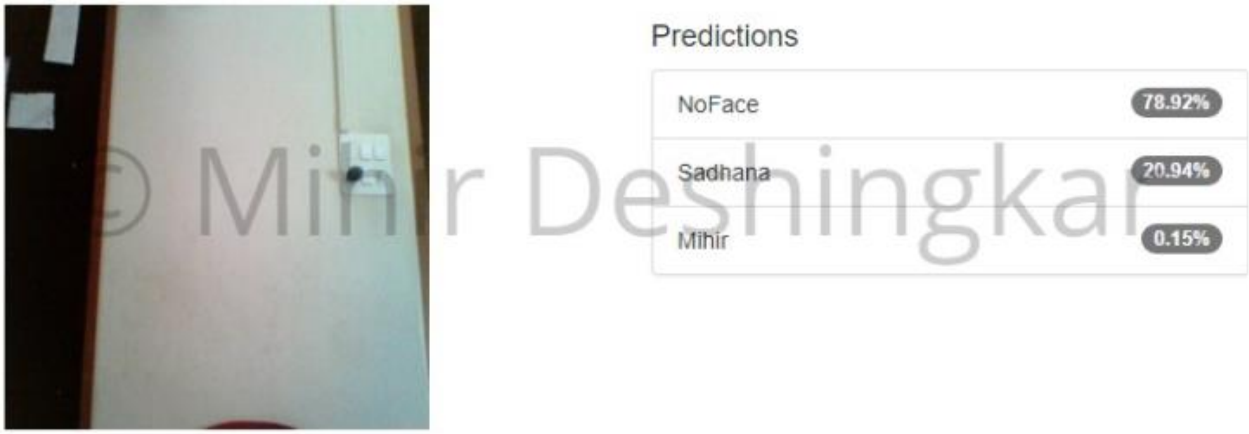


Predictions

Sadhana	50.37%
NoFace	47.98%
Mihir	1.64%

Fig. 5

Also, overall No-face accuracy is comparatively low compared to other two classes, as shown below –



Predictions

NoFace	78.92%
Sadhana	20.94%
Mihir	0.15%

Fig. 6

Discussion:

A good accuracy was achieved with image classifier using comparatively low number of images per class. The network trained faster and resulted in the achieved accuracy in only five epochs in 35 seconds. As the network was trained with side face images, the model could also predict the side face test images with high accuracy. Overall, quite satisfactory results were obtained with minimal data and computation time. It was observed that prediction for each class was proportional to the number of training images used for each class. Consequently, the class named Mihir had the highest prediction accuracy and No-face class prediction had the least accuracy. The No-face class is yet misidentified as some other class when the image does not have pure background, as in figure 5. This can be improved by training the network with more such images of No-face class.

Conclusion / Future Work:

This project has successfully demonstrated the use of an image classifier with NVDIA DIGITS for identifying specific persons. It is clear that such an image classifier can be used for authorizing of automatic door unlocking. The network reached maximum accuracy with only a few epochs and was trained in small amount of time.

For this project to be commercially viable, network accuracy is vital than the inference time. The network accuracy can be further increased by collecting more data and then tweaking with number of epochs. Also, for the prediction to be fail proof, the right type of data should be collected. For example, a No-face class should not identify a wrong face as right one. Hence, the No-face class should contain images of other people also, in order to not misidentify them as one of the other classes. Furthermore, fail-proofing can be added by allowing the image to pass only when the prediction is above a certain value. With such techniques and improved accuracy, it is possible to make the product commercially viable. The next step would be to implement this project on Jetson TX2.