Group 26, Notes on issues identified in first review

We previously worked on and submitted an image classifier that was able to detect different animals based on a large dataset. Upon submission, the classifier was trained for facial recognition and was unable to provide any accuracy. This needed to be investigated further.

Upon further study, we found that the photos used for facial recognition were misclassified because images in the training set had features outside of the actual face and these were used for classification. Such irrelevant features made classification inaccurate.

It is general practice in facial recognition models to first detect the image area corresponding to the face and then use only this area for training and classification. We realised that we could improve the model by adding a face detection step. For this, we used OpenCV's face detection function which helped us to extract just the face from training images. These images are then used to train the new model.

The following code is used for face detection.

```
    image = cv2.imread(filePath, cv2.IMREAD_GRAYSCALE)
    #face_cas = cv2.CascadeClassifier ('haarcascade_frontalface_default.xml')
    #faces = face_cas.detectMultiScale (image,
        scaleFactor=1.1, minNeighbors=3)
    cnn_face_detector =
        dlib.cnn_face_detection_model_v1("./mmod_human_face_detector.dat")
    faces = cnn_face_detector(image, 1)
    #(x, y, w, h) = faces [0]
    rect =  faces[0].rect
    #image = image[y: y+w, x: x+h]
    image = image[rect.top():rect.bottom(), rect.left():rect.right()]
    image = cv2.resize(image, (180,180))
```

Let us take the above image to illustrate how the code works. Using OpenCV, we convert it into the image as given below. This is then resized into an image array of 180 X 180 pixels and is used to train the data. For this face detection, we use a Haar Cascade frontal face classifier in order to reduce computation and make it more viable to run on other devices as well.

With this model, we were able to train a dataset of 5 celebrities and gain an 80% accuracy even though the whole training set had only 14-22 images for each celebrity.

Furthermore, we have also worked to increase the accuracy of the face detection by using a CNN through dlib's face detection model. However, this is more computationally expensive taking longer to detect faces but gives more accurate results.

We encountered some misclassification even after we added face detection.

Due to the complex nature of facial recognition and the black-box nature of CNNs (Convolutional Neural Networks), it is very hard to debug and many of the features that are used cannot be discerned. We cannot easily observe the features that the CNN uses because it is hidden within the layers. By visual inspection, we found that even a hand placed on one's face may cause issues since training data images may not have the same feature.

For instance, this image is classified as Amit Shah and has a low probability for Nitin Gadkari. This is because in all training data samples for Gadkari, we do not have images where he places his hand near his face. However, for Amit Shah, we have such training images and the CNN uses that as a feature for classification. Upon removal of that hand, we get the following image that is used for detection.

Now, upon classification with the same model, we see that the classifier has reduced probability considerably for Amit Shah and now classifies it as Narendra Modi. This could probably be because of the lettering on top left or some other feature that is seen predominantly in the training samples for Narendra Modi – but we do not know for sure and further analysis is required.

An easy way to increase accuracy is to feed more accurate training images having only that person, with only frontal images. It would also help to add specific images for all classes where a certain posture or object is present (hand in front of face, namaste, microphone etc.).

Since the next phase of the project is object classification, and since optimizing the training set for facial recognition is not required for that phase, we are not planning to refine this further. We plan to work further on object classification using ImageAI's object detection as a next step.