Capstone Proposal

Willian Ver Valen Paiva

1 Machine Learning Engineer Nanodegree

1.1 Domain Background

For a long time work in facial recognition has being done and one of the key points of this work is the face alignment as it poses its own challenge. And the main tool used for the job is OpenCV which is used with DLIB to recognize Facial landmarks.

The automatic recognition of landmarks is essential to be able to classify facial expressions, or face tracking, face animation, and even 3D face modeling.

For example to classify facial expressions it is necessary to classify Facial Action Units also known as FACS [1], which in turn needs a proper face alignment.

As one of my main projects today is to create a model capable to recognize facial expression of pain, this subject comes to be perfect as it cover a personal necessity and brings a good subject to work and learn.

1.2 Problem Statement

The problem of face alignment is among the most popular in the field of computer vision and today we have many different implementations to automatically recognize facial landmarks on images. The most known is the Active Appearance Model (AAM) [2, 3].

But today we also some good results using Deep learning to achieve the results for example the work done by Adrian Bulat on recognizing 3D facial landmarks [4] that shows remarkable results it is implemented in torch. a framework for **LUA**.

In resume to find an implementations on Tensorflow is not that easy. As most of the works done on the subject is heavily dependent of DLIB to recognize the landmarks. and as of today Tensorflow is a library that is on

the rise and having such a tool would be a plus and a entry point for more detailed facial expressions.

So for that reason I propose for this project to create a Deep Neural Network to tackle such subject and have a model with better of equivalent performance of the DLIB counter part on Tensorflow.

As the DLIB model has difficulty on recognizing points on faces by the side view.

The problem in question take a image as input (the format and details will be discussed on the dataset section), and by using a regression model to calculate the position of the face landmarks on the given image.

1.3 Dataset and Inputs

When looking for a data set for facial landmarks is possible to find many of them example:

- AFLW [5]
- Cohn-Kanade AU-Coded Expression Database [6]
- Affectiva-mit facial expression dataset (am-fed) [7]

But for this capstone project I propose to use the MUCT Face Database [8], this dataset consists of pictures (resolution 480x640) taken from 276 subjects using 5 cameras in different angles (totalizing) an light conditions like the image below:

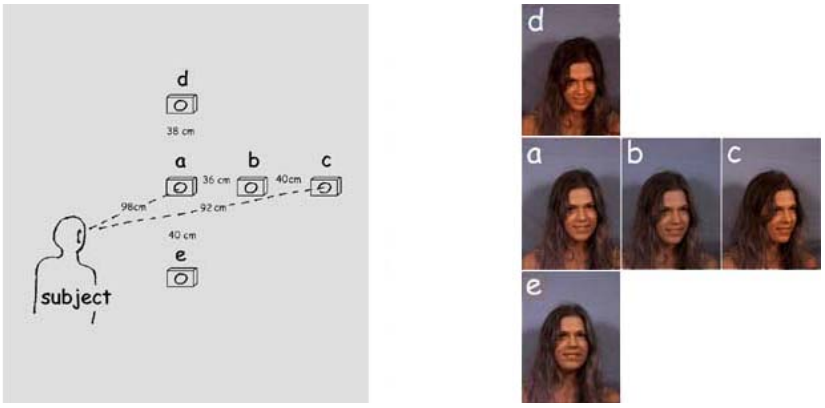
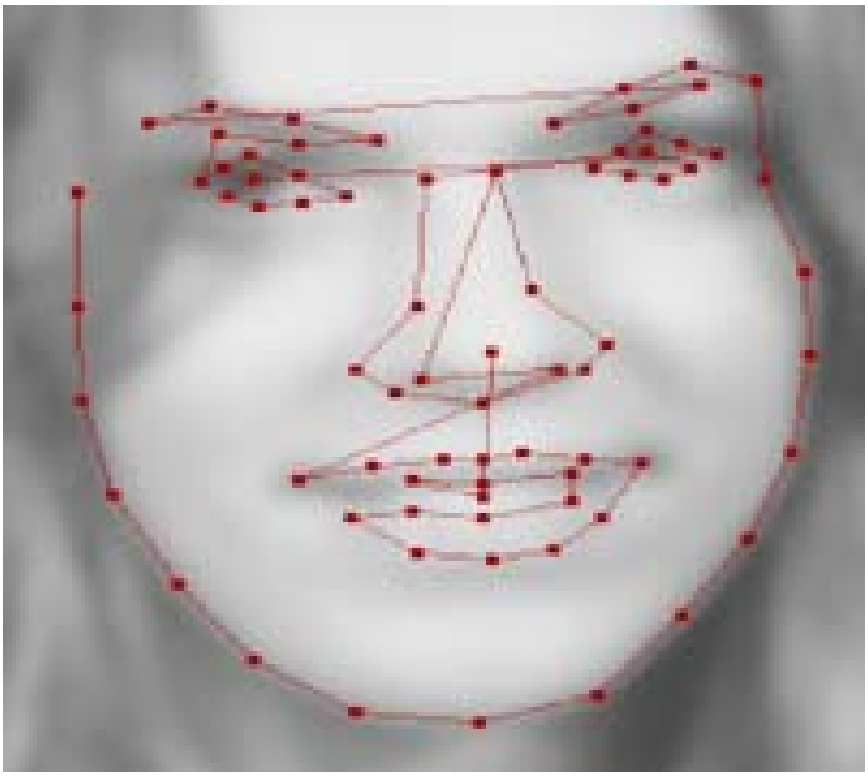


Figure 1: There is no images on the left but they cam be reproduced by mirroring the right side

Each picture is manually coded with 76 facial landmark like:



The landmarks are saved in to 4 different file formats

- muct76.shape shape file [9]
- muct76.rda R data file (www.r-project.org/
- muct76.csv comma separated values
- muct76-opencv.csv comma separated values in OpenCV coords (0,0 at top left).

Note that the coordinate system in these files is the one used by Stasm (i.e. the origin 0,0 is the center of the image, x increases as you move right, y increases as you move up). The exception is muct76-opencv.csv, where the format is the "OpenCV format" (i.e. the origin 0,0 is at the top left, x increases as you move left, y increases as you move down).

Unavailable points are marked with coordinates 0,0 (regardless of the coordinate system mentioned above). "Unavailable points" are points that

are obscured by other facial features. (This refers to landmarks behind the nose or side of the face – the position of such landmarks cannot easily be estimated by human landmarkers – in contrast, the position of landmarks behind hair or glasses was estimated by the landmarkers).

So any points with the coordinates 0,0 should be ignored. Unavailable points appear only in camera views b and c. Unless your software knows how to deal with unavailable points, you should use only camera views a, d, and e.

Note that subjects 247 and 248 are identical twins.

When talking about the train and testing split I will split the data into train 70% and test 30% but it cannot be a random split as we have many images of a same subject. So to be sure the test set is well done the split has to be node at a subject level so the same person cannot be found on test and train.

The choice of this dataset is made because it has a reasonable size to train on personal computers and moreover it has large room for data augmentation if necessary The focus on using this data set is that it provides data to have a better result when the face is in the side view.

The dataset is public available via github on the following link `https://github.com/StephenMilborrow/muct`

1.4 Solution Statement

What I am hoping to achieve from this project is to have a pre trained model that is capable of marking images properly with landmarks using a Tensorflow backend, obtaining achieve results at least as good as the DLIB model, for that I will be using our own Convolutional Neural Networks and pre-trained networks to find the best result for the task. as benchmark I will aim high for this project to to start I will create a model based on the inceptionV3 and just change the final layer to a regression and use it as the first result to compare the progress. The main idea here is to create a model with 152 regression outputs giving the respective X and Y of each point.

1.5 Evaluation and Metrics

As the problem consists on a regression model I believe that for the evaluation of the results I could use the accuracy calculated by using the regression functions Root Mean Squared Error.

1.6 Project Design

To solve such a problem I will begin from preparing the data by doing a proper split and assuring that the same subject cannot be fount on the train and testing set, also some statistical on the data to be sure to have a fair distribution of people of different sex, race, with/without glasses on both sets.

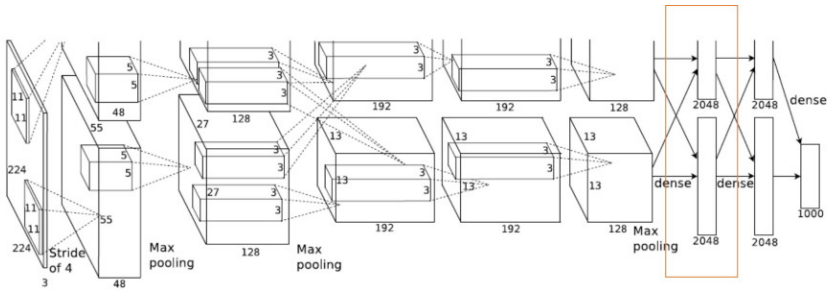
Also gaining some insight on the data.

From that I will use the pre-trained inceptionV3 and create the benchmark model for the project.

Once the environment is prepared the aim is to create many models until it reaches a good result. As planned:

- create a CNN from scratch
- use transfer leaning (inceptionV3, Resnet50, VGG16,...)

When creating my own CNN model i will be starting from the basic 5 Conv layers followed by 3 dense layers (a bit like the AlexNet architecture) and from that work up the architecture test and error.



In case the transfer learning don't give good results another approach would be go up on the pre-trained model and get more fine tuning. By using the option $\text{include}_{\text{top}}$ from keras and augment the number of layers that will be trained. What would increase the time of training but give better results.

So for this capstone will be pushing the max i can to reach a best model at the limit of computational power to train such models

References

[1] Paul Ekman and Wallace V Friesen. Facial action coding system. 1977.

[2] Gareth J Edwards, Timothy F Cootes, and Christopher J Taylor. Face recognition using active appearance models. In *European conference on computer vision*, pages 581–595. Springer, 1998.

[3] Iain Matthews and Simon Baker. Active appearance models revisited. *International journal of computer vision*, 60(2):135–164, 2004.

[4] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *International Conference on Computer Vision*, 2017.

[5] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 397–403, 2013.

[6] JF Cohn et al. Cohn-kanade au-coded facial expression database. *Pittsburgh University*, 1999.

[7] Daniel McDuff, Rana Kaliouby, Thibaud Senechal, May Amr, Jeffrey Cohn, and Rosalind Picard. Affectiva-mit facial expression dataset (am-fed): Naturalistic and spontaneous facial expressions collected. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 881–888, 2013.

[8] S. Milborrow, J. Morkel, and F. Nicolls. The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa*, 2010. `http://www.milbo.org/muct`.

[9] Stephen Milborrow. Active shape models with stasm. *Stasm Version*, 3, 2009.