FA

FACE LANDMARKS DETECTION

DEEP LEARNING COURSE PROJECT REPORT

NIRDOSH RAWAL

INTRODUCTION

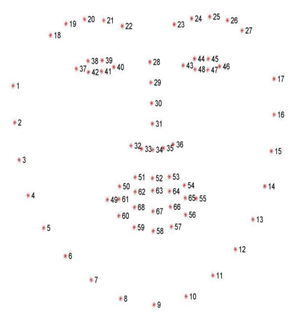
One of the most important aspects of the face processing pipeline is the recognition of dense facial landmarks. Virtual facial animation, emotion recognition and driver status tracking are just a few examples of applications. Have you ever wondered how Snapchat is able to add such stunning effects to our face? It was developed to recognise certain markings on our face and display a filter based on those marks. Face Landmarks are the names given to these markings in Machine Learning. We have used PyTorch in this project to face landmark detection with deep learning.

DATASET DESCRIPTION

There are 300 indoor and 300 outdoor in-the-wild photos in the collection. It encompasses a wide range of identity, expression, lighting, stance, occlusion, and facial size. The 300-W database has a higher percentage of partially-occluded photos and covers more expressions than the standard "neutral" or "smile," such as "surprise" or "scream," when compared to other in-the-wild datasets. A semi-automatic approach was used to annotate images using the 68-point mark-up. The database's photos were deliberately chosen to reflect a representative sample of demanding but natural face occurrences under completely unrestricted settings. Thus, approaches that achieve accurate performance on the 300-W database can show the same accuracy in the majority of real-world scenarios.

METHODOLOGY

Identifying faces in photographs or videos is interesting, but it's not enough information to build effective applications, we need additional information about the person's face, such as location, whether the mouth is open or closed, whether the eyes are open or closed, if the person is looking up, and so on.



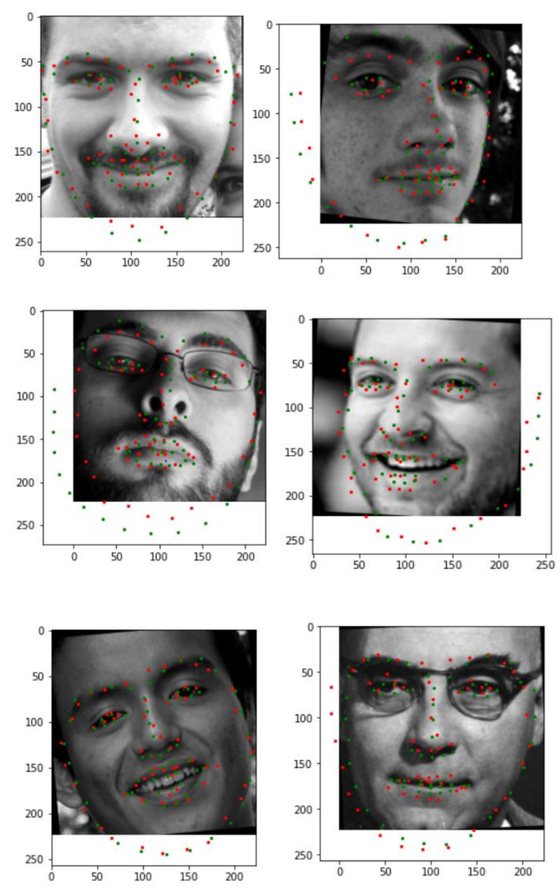
WE followed following steps for project:

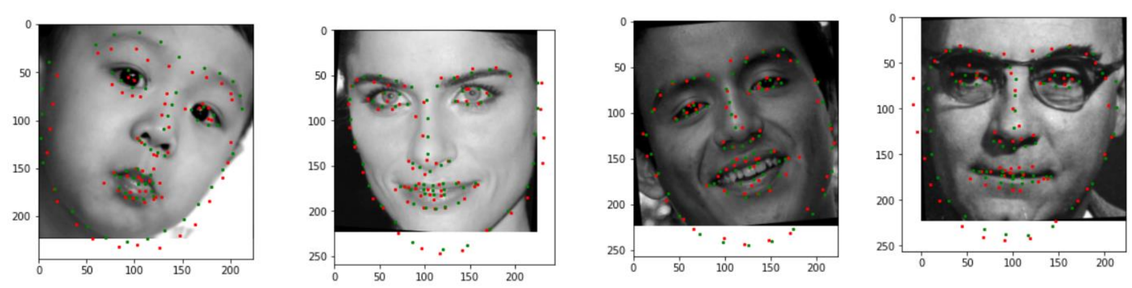
1. We have done this project using PyTorch. So first of all we started with importing the necessary libraries.
2. Then we downloaded the dataset, the dataset used is official DLIB dataset consist of 6666 images of different dimensions.
3. In the images of dataset, the faces contain very less space. If we pass the whole image in the neural network it will take the background as well. So like we prepare text data, we have prepared image data for further exploration.
4. Then we split the dataset into a train and a valid dataset.
5. Then we have used RESNET18 as our fundamental framework. We modified the first and last layers of the network so that layer will fit easily for the purpose.
6. Then we trained our neural network for face landmark detection. We have used the mean squared error between the true and the predicted face landmark.

RESULTS

Train Loss and Valid Loss for 10 epochs:

Epoch: 1 Train Loss: 0.0208 Valid Loss: 0.0060
Epoch: 2 Train Loss: 0.0053 Valid Loss: 0.0051
Epoch: 3 Train Loss: 0.0044 Valid Loss: 0.0041
Epoch: 4 Train Loss: 0.0032 Valid Loss: 0.0030
Epoch: 5 Train Loss: 0.0025 Valid Loss: 0.0025
Epoch: 6 Train Loss: 0.0020 Valid Loss: 0.0021
Epoch: 7 Train Loss: 0.0018 Valid Loss: 0.0022
Epoch: 8 Train Loss: 0.0016 Valid Loss: 0.0019
Epoch: 9 Train Loss: 0.0015 Valid Loss: 0.0017
Epoch: 10 Train Loss: 0.0013 Valid Loss: 0.0015





REFERENCES

1. AnchorFace : An Anchor-based Facial Landmark Detector Across Large Poses
2. Fast Facial Landmark Detection and Applications: A Survey
3. github.com