­Name: Ayush Tabhane Email ID: tabhane.ayush90@gmail.com

ID Number: ID-02

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No | Date | Work Done Today | Next Day Plan | Important things to Remember |
| 1 | 08/05 | Installing opencv python and requirements, plotting histograms of images. |  |  |
| 2 | 09/05 | Detection of human frontal and profile faces, eyes, mouth using different haar cascades. Experimenting with haar cascades and lbp cascades to predict person looking to his left. |  | Haar and lbp cascades do not predict profile faces in which person is looking to his left. |
| 3 | 10/05 | Learned github, crawled images from image-net and learned about fischer faces. |  |  |
| 4 | 11/05 | Crawling of images via link and via text files. Implementation of fischer face detection on CK+ dataset. | Improve accuracy for face detection. | The urllib package throws unicode error sometimes when crawling directly via a link, this error won't occur in case of text file url extraction. |
| 5 | 12/05 | Increased the accuracy of fischer face detection to 87% by increasing the dataset and combining some files for emotions. |  | contempt -> sadness  disgust -> anger |
| 6 | 13/05 | Tested different xml files generated for real test subject. Took the real camera feed of my emotions to evaluate model accuracy. |  |  |
| 7 | 14/05 | Holiday |  |  |
| 8 | 15/05 | Read about deep cnn, wrote a deep cnn based code and trained for 25 epochs. Got output training and test accuracies.  Accuracy – 72.81% for 25 epochs. | To train deep cnn for larger no. of epochs | Download emofbvp database,  digital ocean server training |
| 9 | 16/05 | Trained the deep cnn for 100 epochs. Deep cnn is found very good in predicting happiness emotion. Generated database of my own emotional facial expressions for proper evaluration. | To simultaneous-ly use fischer and deep cnn for predicting human emotions and evaluating accuracy. |  |
| 10 | 17/05 | Wrote 2 codes, 1 for reading images from a folder and other for reading input from a camera and predicting emotion simultaneously using fischer face prediction and deep learning convolutional neural network.  Tried to introduce some modifications to improve accuracy for code 1. | To test accuracy of input read from camera i.e. code 2 and improving the accuracy. |  |
| 11 | 18/05 | Compromised with emotion between anger and sadness and tested the code2 (getting camera feed) on other subjects directly with live feed from camera. Removed the bugs from code 2 to get it working correctly. | To test efficiency on demo videos generated from cctv cameras. |  |
| 12 | 19/05 | Tested the efficiency on two movie clips. The efficiency is found good, the predicted emotion even for profile view is good up to a large extent. However currently, we are relying heavily on haarcascades to detect faces which is the reason for major errors. We need to improve the face detection for target subjects. | To improve face detection accuracy for shortlisting target subjects in a test image. |  |
| 13 | 20/05 | Read about how to improve facial recognition accuracy. Standard PEP8 formatting of all codes developed till now, pushing into github and documentation about work done. |  |  |
| 14 | 21/05 | Holiday |  |  |
| 15 | 22/05 | Tried to improve the face recognition accuracy using upper body and full body detection haar cascades. But the improvement in accuracy was negligible. Increased the minimum face size size to be detected to 50x50. This excluded some minute errors. Increased the minNeighbours parameter in detectMultiScale to 10 for detecting better quality faces. Tested the model on another indian school video, the emotion detection accuracy was still good only there are problems with face recognition. | To test model on foreign school videos. | Haarcascades might be trained on western world people which may reduce its accuracy in predicting eastern world people. If face is not bright enough, then it will be hardly detected no matter the size or clarity or closeness to the camera. |
| 16 | 23/05 | Removed minor bugs from the code. Face detection working fine, multiple faces are being detected and emotion recognition can be easily applied. Errors reduced to minimal. Only one haar cascade found enough for face detection in many videos. Calculated frames per second for test videos, found around 5 to 6 fps on intel i5 processor with 4gb ram. Would work faster on server. | To evaluate the final efficiency by manually evaluating emotion for every face. |  |
| 17 | 24/05 | Final Documentation report for work done in emotion detection.  Installed Qt, started working on displaying images and video using Qt features. | To create a UI displaying images and taking user input for evaluated emotion and storing in txt file. |  |
| 18 | 25/05 | More work on Qt UI development.  Video display working fine. Image display working fine.  Made a demo video showing the working of emotion detection. | Need to integrate and run python script into Qt and get the output generated from it. |  |
| 19 | 26/05 | Python code integrated in Qt UI window. Any video can be browsed from pc and given as input for emotion detection. Output images in a newly generated folder. Images will have original frame and detected faces with predicted emotion. | To generate a UI window with specific size and display frame and faces with emotion inside the UI window. |  |
| 20 | 27/05 | Code debugging for Qt GUI. Unresolved errors occurred while running python code for GUI using pyqt5 in ubuntu OS. Need to implement GUI without Qt media player object | To display proper images in the final designed UI. | Ubuntu dont have developed media plugins needed by QmediaPlayer object of Qt. |
| 21 | 28/05 | Holiday |  |  |
| 22 | 29/05 | GUI implemented. All function working fine. Can properly select video.  Issues:-  1. PyQt5 install in python3. Emotion detection dependencies all installed in python2.  2. Both can be ran simultaneously but it will take more time.  3. Integrating into one code will make it faster.  4. Should be integrated all into python2. | PyQt5 needs to be installed for python2. Cannot be installed via pip. Resolving manual installation errors. |  |
| 23 | 30/05 | Resolved all GUI issues. Any video file can be browsed now and can be used for emotion detection. |  |  |
| 24 | 31/05 | Tried to implement emotion detection GUI with threading to make it faster. |  | If multiple threads are called to predict emotion on different faces, there is no guessing when shall we move to next frame. By threading, the optimized algorithm will start running frames faster than emotion prediction. These will in turn ruin the structured GUI display. |
| 25 | 01/06/17 | More stable implementation of GUI with multi-threading. Found a way around of making code faster by using multi-threading. For n number of faces, n threads are initialized simultaneously to predict emotion making the code faster. GUI is not updated till emotion detection is done. |  |  |
| 26 | 02/06 | Started learning Django. Will be implementing Emotion Detection using Django for Web Development. |  |  |
| 27 | 03/06 | Tested working of libraries and their compatibilities with django. Created a new app for Emotion detection in the project. Debugging some errors generated. |  |  |
| 27 | 04/06 | Holiday |  |  |
| 28 | 05/06 | Completed Emotion Detection integration with Django. Any template format or video can be generated as output as required by front end website development. |  |  |
| 29 | 06/06 | Wrote a code which can imitate any pattern from one image into another. The desired pattern was separated from original image by binarizing the image. |  |  |
| 30 | 07/06 | Wrote a code which can generate a zig zag pattern or any polygonal structure. The code is automated and generates a continuous pattern of desired parameters like separation between zig zag lines and distance from the border of polygon. |  |  |
| 31 | 08/06 | Working on a code which can align any object with reference to another one. Scaling and angle of both objects should be made same so that object will be properly aligned. |  |  |
| 32 |  |  |  |  |
| 33 |  |  |  |  |
| 34 |  |  |  |  |