Week 1 Analysis

Ahmad Sarmad Ali

Face Detection Models:

- I have researched upon different libraries that we can use for face detection.
- Found different models that we can use for face detection
- Harr Cascade
- YOLO
- Retinaface by Insightface. Link
- From all the above models I have found Retinaface very useful due to following resons

Retinaface Features:

- As we can detect faces by using any face detector but what makes the retinaface stand out from others is it's output format.
- Retinaface returns the output in the following format

```
{
    "face_1": {
        "score": 0.9993440508842468,
        "facial_area": [155, 81, 434, 443],
        "landmarks": {
            "right_eye": [257.82974, 209.64787],
            "left_eye": [374.93427, 251.78687],
            "nose": [303.4773, 299.91144],
            "mouth_right": [228.37329, 338.73193],
            "mouth_left": [320.21982, 374.58798]
```

- As you can see it returns a dictionary having score and it also returns coordinates for different facial landmarks which is very helpful in applying filters.
- We can apply even **emotion based filters** by using these coordinates.

Why Chose retinaface:

- We can of course train our custom model for that task but we need a data set for that I have also find a dataset for that as well. Link

- But training a custom model will not give us that good performance that we are getting from retina face.

Emotion Based filter:

- I have implemented a emotion based filter application using retinaface.
- It applies a mostache filter on smiling.
- The coordinates of facial land marks that we get from retinaface makes it much easier (not that much easy).
- We can use Nose, eyes and mouth coordinates (upper and lower lip) to detect change in emotions and apply filters respectively.
- Although this is not best way I will discuss another method that I have found and we will use that which is much more efficient.

Landmark detection:

- As I have discussed above with the use of retinaface we will be able to detect several facial landmarks which will be helpful in further processing.
- We will be using another method for emotion based filter application.
- I have already shared the format of landmarks above.

Face Recognation:

- Face detection and recognition are two different things but don't worry along with retinaface Insightface also provide a face recognition model.
- This model is Arface and it is also pretrained model with best accuracy nearly surpassing human accuracy. (according to creators)
- It gives 99% whiles humans give only 97%.
- We also have facility to use both retinaface and arcface together and they are wrapped in a single model Known as Deepface.
- Cool, isn't it?
- So know we have best models for face detection and recognition.
- Although we can train a custom model for recognition as well

Why pertained:

- I have already discussed above if we train a custom model we will not be able to get good performance as these pretrained models as they are trained on huge datasets giving good performance.

Nest work to do:

- Another more efficient method that I was talking about for applying emotion based filter application is using a Neural network to detect emotions in each frame and they apply filters based on the output from the emotion predictor model.
- This method is way more efficient then the currently implemented method.
- One reason is that currently we are applying emotion based filters by guessing the change in coordinates of face which doesn't work always as it have flaws.
- Now I am working to train a custom emotion detector model that can detect the emotions and then we can apply filters based on that.
- I have also found a useful resource for that as well. Link

Conclusion:

So, till now I have finalized best models for all three tasks face detection, recognition and landmark detection. I have also implemented a emotion based filter application but it is not that much efficient. Also found another efficient method for doing that. So, I will try to implement it in next week (IA).