

Praktikum Informatik 1 WS2017

Final Report

Project Name: Face Album

Zheng Li

0906101

## Project description

The aim of this project is to find a suitable approach to manage photos based on facial recognition and facial expression using different algorithms in digital image processing and machine learning technology. The final software product should allow user to have a collection of photo where he or she can sort the photo after different people and different emotions of the facial expressions of those people based on the prediction of the software.

## Project Approaches

The three main key points for the project are face detection, face recognition and facial expression classification. For each key point, the solution are described as follow. For detailed design decision, please refers to the design decision documents.

### Face detection

For this assignment, I used a frontal face Haar-like classifier in Open CV library. The algorithm is pre-trained and included in the Open CV library. The advantage of the Haar-like classifier is that, it can achieve a good accuracy with very high efficacy. In my observation, frontal faces in a picture are generally positive detected. It is robust against different light conditions and different size of faces.

### Face recognition

For face recognition, I tested two different solutions. They are eigenface and local binary patterns histograms. The eigenface approach uses a set of training data (face pictures) and project each face into a high dimensional space. Each face are represented by all the eigenface with a vector. By comparing the distance of a test data to all the eigenface vector, the algorithm can assign a prediction within the existing faces. The local binary patterns histogram avoid project the data set into a high dimensional space, it summarize the local structure in an image by comparing each pixel with its neighborhood. In my practice, I chose the local binary patterns histogram because it is not only less sensitive to lighting conditions than the eigenface approach but also retainable. Once a training is done with eigenface, the model is fixed, if we want to add new face to the model we will have to retain the model again.

### Emotion recognition

The the assignment of emotion recognition, I used a high level API from tensor flow, the tensor flow for poet. It uses a Nero-network that is pre-trained over millions of object image from the inception image database. The model itself is able to classify different objects common life, but it can be re-trained for classify certain types of objects in image by taking away the last few layers in the network and re-train with different data set. For this project, I used the Cohn-kanade facial expression dataset to train a model that can classify three types of emotions, happy, sad and surprise. This model is then embedded into the application for facial expression recognition.

## Application Prototype

To combine all the key points, I developed a prototype of the application in python. Using this application, users can import their own photos into the library of the application. After importing, the faces in the photo is cropped out and stored on the disk for later training, the cropped-out face picture is then passed to the facial expression model to label the emotion prediction. In the interface of the application, users can manage the detected face, remove the false positive detections, name a face. After naming a face in a picture, the information is stored in databank. When the criteria meet the need(at least 2 names and 10 face pictures in databank), that is there are enough data to train the facial recognition model, the data will be passed to the facial recognition model to train the model with their labels. The application will try to predict the names of the faces in all the picture once the model is trained, that include the newly imported photos and the photos that already exist, if the value of the prediction is over a threshold, the face is labeled as unknown. If user confirms a face that is correctly predicted, that face data is then added to the training set to improve the accuracy of the model. The model will be feed with faces that are either user named or user confirmed. While using the application, the facial recognition will get better and better if user give it more confirms.

After importing the photos, user can browse the library, go to each single photo detail view to check out information that has been extracted. The application can mark all the detected faces on the picture and show their emotion predicted. Users are also able to filter the library with name of the face and emotion of the face.

The application also provides the capability to manage all the recognized faces. User can have a collection view of all people that is recognized in the library by the application and manage them as he or she wants.

## Result

I tested the collection with a set of 80 photos, the result are shown in the following table.

Total faces count	Detected	False positive	Missed
163	159	12	4

Emotion	Manually classified count	Predicted count
Happy	128	109
Sad	12	39
Surprise	12	14

For face recognition, the application starts to deliver correct prediction after 4 to 5 user confirm for each person, the model achieved the best result after 10 user confirms.

## Limitation and Future work

During the test of the prototype, some issues are discovered.

In face detection, the application gives best result for frontal faces that are horizontal aligned. For side faces and faces that are tilted, the model can not recognize well. This is mostly because the Haar-like classifier is trained with very standard data set in which all the subjects are facing directly towards camera. To allow the model to detect side faces, one improve method is to train an extra classifier with side face data set. In order to deal with faces that are tilted in the picture, a alignment pre-process can be done by extract some key points of the face features like nose, eyes and transform the face image based on the direction of the key points.

The face recognition model can classify people that looks not very the same very well, especially when gender difference presents. But it can't give satisfying result for family members that have a lot of same appearance features, especially for sibling with the same gender. This is also sometimes a hard job for humans too.

The emotion classification can also be improved by training the model with a much bigger and much detailed data set, but unfortunately it will be very hard to achieve a very high accuracy by using machine learning, as in real life, people express their emotion differently, sometime it is for us human hard to read the emotion of some facial expression without certain situation context like conversation.