Emotion Recognition Using Machine Learning

Fahad Ur Rahaman[1]
University of Texas at Arlington[1]
fahadur.rahaman@mavs.uta.edu[1]

Abstract

Emotion recognition based on facial expression is an interesting research field, which has presented and applied in several areas such as safety, health and in human machine interfaces. Researchers in this field are interested in developing techniques to interpret, code facial expressions and extract these features to have a better prediction by computer. Face detection has been around for ages. Taking a step forward, human emotion displayed by face and felt by brain, captured in either video, electric signal (EEG) or image form can be approximated. Human emotion detection is the need of the hour so that modern artificial intelligent systems can emulate and gauge reactions from face. This can be helpful to make informed decisions be it regarding identification of intent, promotion of offers or security related threats. Recognizing emotions from images or video is a trivial task for human eye but proves to be very challenging for machines and requires many image processing techniques for feature extraction. Several machine learning algorithms are suitable for this job. Any detection or recognition by machine learning requires training algorithm and then testing them on a suitable dataset. This paper explores a couple of machine learning algorithms as well as feature extraction techniques which would help us in accurate identification of the human emotion.

1 Introduction

Emotion recognition is a large and important research area that addresses two different subjects, which are psychological human emotion recognition and artificial intelligence (AI). The emotional state of humans can obtain from verbal and non-verbal information captured by the various sensors, for example from facial changes, tone of voice and physiological signals. In 1967, Mehrabian showed that 55% of emotional information were visual, 38% vocal and 7% verbal. Face changes during a communication are the first signs that transmit the emotional state, which is why most researchers are very interested by this modality.

Human emotion recognition is implemented in many areas requiring additional security or information about the person. It can be a second step to face detection where we may be required to set up a second layer of security, where along with the face, the emotion is also detected. This can be useful to verify that the person standing in front of the camera is not just a 2-dimensional representation. Another important domain where we see the importance of emotion detection is for business promotions. Most of the businesses thrive on customer responses to all their products and offers. If an artificial intelligent system can capture and identify real time emotions based on user image or video, they can decide on whether the customer liked or disliked the product or offer.

We have seen that security is the main reason for identifying any person. It can be based on finger-print matching, voice recognition, passwords, retina detection etc. Identifying the intent of the person can also be important to avert threats. This can be helpful in vulnerable areas like airports, concerts and major public gatherings which have seen many breaches in recent years.

Human emotions can be classified as: fear, contempt, disgust, anger, surprise, sad, happy, and neutral. These emotions are very subtle. Facial muscle contortions are very minimal and detecting these differences can be very challenging as even a small difference results in different expressions. Also, expressions of different or even the same people might vary for the same emotion, as emotions are hugely context dependent. While we can focus on only those areas of the face which display a maximum of emotions like around the mouth and eyes, how we extract these gestures and categorize them is still an important question. Neural networks and machine learning have been used for these tasks and have obtained good results.

Machine learning algorithms have proven to be very useful in pattern recognition and classification. The most

[1]MSCS Masters Candidate, Machine Learning Programming Group

important aspects for any machine learning algorithm are the features. In this paper we will see how the features are extracted and modified for algorithms like Support Vector Machine. We will compare algorithms and the feature extraction techniques. The human emotion dataset can be a very good example to study the robustness and nature of classification algorithms and how they perform for different types of dataset.

Usually before extraction of features for emotion detection, face detection algorithms are applied on the image or the captured frame. We can generalize the emotion detection steps as follows:

1) Dataset preprocessing
2) Face detection
3) Feature extraction
4) Classification based on the features

In this work, we focus on the feature extraction technique and emotion detection based on the extracted features, focus on some important features related to the face. It also covers some important algorithms which can be used for emotion detection in human faces. In this paper, we will also discuss the details about the tools and libraries used in the implementation and explain the implementation of the feature extraction and emotion detection framework. In this paper, we will discuss about the dataset description, very brief description of the main references of your project, Difference in approach/ between our project and the main projects of our references, Difference in accuracy/performance between our project and the main projects of our references, List of your contributions in the project, analysis of what went well, what could have been better and what left for future work. Also, we will discuss about the conclusion and references.

2 Dataset Description

We have used the COCO-Common Objects in Context database and facial expression set of images repository created by machine learning graduate students. COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features:

- Object segmentation
- Recognition in context
- Super pixel stuff segmentation
- 330K images (>200K labeled)
- 1.5 million object instances
- 80 object categories
- 91 stuff categories
- 5 captions per image
- 250,000 people with key points

COCO database has 2 files viz. “val2017” and “train2017” files from which we have downloaded 2693 and 1000 images respectively for category “person”. These images were not labeled, and we had to segregate the images for each emotion in a text file. The database covers all the basic human emotions displayed by the face. The emotions and are as follows: 0 - anger, 1 - happy, 2 - neutral, 3 - sad, 4 - surprise. The database is widely used for emotion detection research and analysis.

3 Project Description

In this project, we will aim to recognize the human emotions such as anger, happy, neutral, sad, surprise using the COCO database images. Here, the quality of the image and the clarity of the facial expression in the images matters to extract emotions from the image.

3.1 Description

A static approach using extracted features and emotion recognition using machine learning is used in this work. The focus is on extracting features using python and image processing libraries and using machine learning algorithms for prediction. Our implementation is divided into three parts. The first part is image pre-processing and face detection. For face detection, we have used OpenCV’s HAAR classifier. Once the face is detected, the region of interest and important facial features are extracted from it. There are various features which can be used for emotion detection. In this work, the focus is on facial points around the eyes, mouth, eyebrows etc.

We have a multi-class classification problem and not multi-label. There is a subtle difference as a set of features can belong to many labels but only one unique class. The extracted facial features are used to detect the multi-class emotions. Our database has a total of 5 classes to classify. In our project we have implemented and analyzed our results with Fisherface algorithm. The processing pipeline can be visualized as Figure 1.

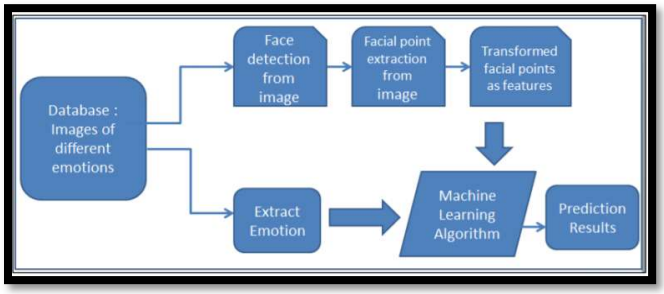

Figure 1: Implementation Pipeline.

Below is the list of steps performed in the project:

- Organizing the dataset
- Extracting the faces
- Creating training and classification set
- Testing and analyzing results

3.2 Fisherface Algorithm

One can perform dimensionality reduction using linear projection and still preserve linear separability. This is a strong argument in favor of using linear methods for dimensionality reduction in the face recognition problem, at least when one seeks insensitivity to lighting conditions.

Since the learning set is labeled, it makes sense to use this information to build a more reliable method for reducing the dimensionality of the feature space. Here we argue that using class specific linear methods for dimensionality reduction and simple classifiers in the reduced feature space, one may get better recognition rates than with either the Linear Subspace method or the Eigenface method. Fisher's Linear Discriminant (FLD) is an example of a class specific method, in the sense that it tries to "shape" the scatter in order to make it more reliable for classification. This method selects W in such a way that the ratio of the between-class scatter and the within-class scatter is maximized.

Let the between-class scatter matrix be defined as

$$S_B = \sum_{i=1}^{c} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$$

and the within-class scatter matrix be defined as

$$S_W = \sum_{i=1}^{c} \sum_{\mathbf{x}_k \in X_i} (\mathbf{x}_k - \boldsymbol{\mu}_i)(\mathbf{x}_k - \boldsymbol{\mu}_i)^T$$

where μ_i is the mean image of class X_i, and N_i is the number of samples in class X_i. If S_W is nonsingular, the optimal projection W_{opt} is chosen as the matrix with orthonormal columns which maximizes the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples, i.e.,

$$W_{opt} = \arg\max_W \frac{\left|W^T S_B W\right|}{\left|W^T S_W W\right|} = \left[\mathbf{w}_1 \;\; \mathbf{w}_2 \;\; \cdots \;\; \mathbf{w}_m\right]$$

where {wi, i = 1 2,….,m}, is the set of generalized eigenvectors of SB and SW corresponding to the m largest generalized eigenvalues {λi, i = 1,2,…,m}, i.e.,

$$S_B \mathbf{w}_i = \lambda_i S_W \mathbf{w}_i, \quad i = 1, 2, \ldots, m$$

In order to overcome the complication of a singular S_W, we propose an alternative to the criterion. This method, which we call Fisherfaces, avoids this problem by projecting the image set to a lower dimensional space so that the resulting within-class scatter matrix S_W is nonsingular. This is achieved by using PCA to reduce the dimension of the feature space to N - c, and then applying the standard FLD defined by to reduce the dimension to c - 1. More formally, W_{opt} is given by

$$W_{opt}^T = W_{fld}^T W_{pca}^T$$

where

$$W_{pca} = \arg\max_W \left|W^T S_T W\right|$$

$$W_{fld} = \arg\max_W \frac{\left|W^T W_{pca}^T S_B W_{pca} W\right|}{\left|W^T W_{pca}^T S_W W_{pca} W\right|}$$

3.3 Main References Used for The Project

For our project, the main references that we have used is from van Gent, P. (2016). Emotion Recognition with Python, OpenCV and a Face Dataset. *A tech blog about fun things with Python and embedded electronics.* Retrieved from:
http://www.paulvangent.com/2016/04/01/emotion-recognition-with-python-opencv-and-a-face-dataset/

In this reference they have used the labelled data that has been organized first and then the labelled dataset is stored into the directory and then for each emotion the data was retrieved, split into training and classification set and stored into the list "training_data", "training_labels", "prediction_data", "prediction_labels". For each emotion, the data was trained, and prediction was made, and accuracy calculated for each emotion and then the mean accuracy of our emotion recognition was calculated.

3.4 Difference in Approach Between Our Project and Our Reference's Main Project

In the reference model, the training and classification split was different and it was using the pre-trained model and in our project we had to train the model from scratch and predict the images and find the accuracy of each emotion which was different. Also, we had to perform many enhancements such as applying filters, resizing of the images, rotation and flipping of the images, thresholding which increased the overall accuracy of the model's prediction.

3.5 Difference in Accuracy Between Our Project and Our Reference's Main Project

Our reference's main project accuracy was 32% and the best accuracy of our project was ~46%. In Our project we had performed many different approaches such as we split the dataset for each emotion with 90-10 split to increase the accuracy of our project in comparison to our reference's main project who performed 80-20 split. We then used techniques such as resizing of the images, rotating the images with angles of 15 and 125 degrees, flipping the images left to right and top to bottom, applying convolution filters, increase image sharpness, adaptive threshold-

ing, Otsu thresholding techniques which helped us in enhancing the accuracy of our model from 30% to 46%.

3.6 List of Contributions in the Project

The entire project was performed by "Fahad Ur Rahaman", MSCS University of Texas at Arlington graduate candidate. Below is the list of contribution made toward the project's success:

- Research on the emotion recognition topic
- Setting up the environment
- Downloading the COCO database images for "Person" category
- Segregating the images for each emotion class viz. anger, happy, neutral, sad, surprise
- Using the OpenCV's HAAR classifier to extract the faces and store the images as per the input face id's
- Defining the functions from scratch to retrieve the gray scaled images for each emotion, split the data into training and prediction set for each emotion
- Defining function to store the perform testing and prediction to recognize emotions
- Defining function to calculate the accuracy for each emotion, mean accuracy for the entire model
- Enhancing the accuracy of the model by resizing the images, rotating the images with angles of 15 and 125 degrees, flipping the images from left to right and from top to bottom, applying convolution filters, increasing image sharpness, applying adaptive thresholding, Otsu thresholding techniques

4. Analysis

4.1 What Went Well in the Project

The project was to recognize the emotion recognition of the human from COCO dataset images. We did well in extracting the emotions from images with quite good accuracy by training ~ 400 images. The COCO dataset images were not of good quality and the clarity of the human faces were not clear in the COCO database's images for person category. Yet, our model did the pretty good work of extracting the emotions, training, and predicting them accurately with ~46%.

4.2 What Could Have Been Done Better

We could have used a greater number of images for training purpose that could have increased the accuracy of the model. We could have used multiple HAAR filters or other techniques for extracting faces. We could have used different machine learning algorithms or neural networks that could have helped in better training of the images and hence the model's accuracy would have been enhanced much more.

4.3 What is Left for Future Work

For future work, a more robust face detection algorithm coupled with some good features can be researched to improve the results. We focused on only some distances and areas, there can be many more such interesting features on the face which can be statistically calculated and used for training the algorithm. Also, not all the features help to improve the accuracy, some maybe not helpful with the other features. Feature selection and reduction technique can be implemented on the created feature to improve the accuracy of the dataset. We can experiment with facial action coding system or feature descriptors as features or a combination of both of them. Also, we can experiment with different datasets amongst different races. This will give us an idea if the approach is similar for all kinds of faces or if some other features should be extracted to identify the emotion. Applications such as drowsiness detection amongst drivers can be developed using feature selection and cascading different algorithms together.

4.4 Analysis of Results

	Anger	Happy	Neutral	Sad	Surprise	
2	39.66942149	37.19008264	38.84297521	41.32231405	36.36363636	
3	46.28099174	28.92561983	31.40495868	33.05785124	36.36363636	
4	32.73809524	29.16666667	33.92857143	33.33333333	27.97619048	
5	36.9047619	27.38095238	36.30952381	32.14285714	38.69047619	
6	41.88034188	41.02564103	37.60683761	39.31623932	40.17094017	
7	41.88034188	35.8974359	45.2991453	37.60683761	36.75213675	
8	47.00854701	42.73504274	41.88034188	44.44444444	43.58974359	
9	32.47863248	44.44444444	44.44444444	42.73504274	34.18803419	
10	42.10526316	42.10526316	38.59649123	24.56140351	45.61403509	
11	40.35087719	42.10526316	45.61403509	42.10526316	43.85964912	
12	36.84210526	42.10526316	43.85964912	36.84210526	43.85964912	
13	29.8245614	40.35087719	38.59649123	36.84210526	54.38596491	
14	47.36842105	35.0877193	47.36842105	50.87719298	29.8245614	
15	40.35087719	36.84210526	47.36842105	52.63157895	47.36842105	
16	47.36842105	36.84210526	40.35087719	42.10526316	45.61403509	
17	52.63157895	43.85964912	50.87719298	42.10526316	42.10526316	
18	40.35087719	38.59649123	31.57894737	40.35087719	28.07017544	
19						
20	40.9431833	37.92121309	40.81925439	39.55176309	39.69391462	Mean Accuracy for each emotion
21						

Figure 2: Comparison of mean accuracy for each emotion

5. Conclusion

Our implementation can roughly be divided into 3 parts:

1. Face detection
2. Feature extraction
3. Classification using machine learning algorithms

Feature extraction was very important part of the experiment. We tried various techniques in increasing the accuracy of the model from 30% to 46%. However, we can work with a greater number of images in future that could increase the accuracy of the model. Also, use K-fold CV that could help us in better train-test split. There are different types of algorithms which can be used for Face Recognition that are PCA (Principal Component Analysis), LDA (Linear Discriminant Analysis), ICA (Independent Component Analysis), EBGM (Elastic Bunch Graph Matching), Fisher faces.

Our main focus was on feature extraction and analysis of the machine algorithm on the dataset. But accurate face-detection algorithm becomes very important if there are multiple people in the image. If we are determining the emotion of a particular person from a webcam, the webcam should be able to detect all the faces accurately.

Algorithms like logistic regression, linear discriminant analysis and random forest classifier can be fine-tuned to achieve good accuracy and results. Also, metrics such as cross validation score, recall and f1 score can be used to define the correctness of model and the model can be improved based on these metric results.

6. References

Raut, Nitisha, (2018). *"Facial Emotion Recognition Using Machine Learning*" Master's Projects. 632.

Elsevier B.V 1877-0509. "*Facial Emotion Recognition Using Machine Learning*"

Peter N. Belhumeur, Joao~ P. Hespanha, and David J. Kriegman, Vol. 19, No. 7, July 1997. "*Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection*" IEEE Transactions on pattern analysis and machine intelligence

van Gent, P. (2016). *Emotion Recognition with Python, OpenCV and a Face Dataset.* A tech blog about fun things with Python and embedded electronics.