DYNAMIC PROCESSING OF FACIAL PARAMETERS FOR EFFICIENT EMOTION DETECTION

Somya Lohani | somyaloh@iitk.ac.in
DEPARTMENT OF COGNITIVE SCIENCE
Indian Institute of Technology Kanpur

Abstract – **Facial Expressions are one of the most prominent tools for expression of affective states by humans. Hence, emotion detection based on processing of facial parameters in static images has been studied extensively in the recent past. It has also been noted that facial expressions are dynamic in nature and vary rapidly based on the underlying emotion, which is what form the basis of this paper. The objective of this paper is to propose a technique to process the facial feature points and animation units of a person's face dynamically to achieve accurate emotion recognition. Different SVM models are trained to detect emotions in a static images, the results of which are then extended to the frames of dynamic videos. Based on the results of the classification on the video frames, a model is trained to detect the underlying emotion by means of a maximum confidence based algorithms and a discrete Hidden Markov Model classifier with the emotion detected per frame as the observation sequence.**

Keywords – **Emotion Detection, Facial Expressions, Hidden Markov Models, Maximum confidence, Support Vector Machines**

I. INTRODUCTION

Our facial features present a vivid display of our underlying emotional state. Hence a lot of our day to day interaction revolves greatly around processing of the facial expression of people. The way we process expressions defines our perception of the world around us and consequently the way we interact with it. Given the importance of facial expression processing in our lives, it has become a topic of considerable interest in the recent times. The several classifiers that have been built to detect emotions can be broadly divided into two types, static classifiers and dynamic classifiers.

Most current researches revolve around the processing of facial parameters of static images, i.e. static classification. However, studies have shown that processing of dynamic expressions more reliably mimics the working of neural networks of emotion processing in our brain in contrast to static ones. To that end we hope to leverage this information to develop an intelligent model that processes facial parameters dynamically to achieve accurate emotion recognition.

Our work focuses on the design of the static classifiers that use facial features for emotion detection, the results of which, are then used for performing the emotion recognition in a dynamic stream of images. Hence, our paper is broadly divided into two sections: one for static classification and the other for temporal classification. Post the two a conclusion has been presented and acknowledgments have been made.

II. STATIC EMOTION CLASSIFICATION

As the first step in the process, two efficient SVM models are to be trained to detect the facial expressions in a static image based on the Animation units (AUs) and Facial feature points (FPPs).

A. MODEL TRAINING AND OPTIMIZATION

The SVM model was trained to detect 7 emotional states: anger, disgust, surprise, happiness, and neutral emotion. In the present case, the model was trained on the basis of the facial landmark points on static images. To be able to generate the landmark points, a facial landmark point detector (from dlib library [1]) was used. A SVM model was trained with 48X48 matrices (representing people's images) and the emotional state potrayed in the image as the inputs. Post training, the model predicted the emotion represented in the image based on it's 48 x 48 matrix representation.

After initial training, the most effective gamma (gamma decides the amount of curvature we want in a decision boundary) and the most effective decision function for the SVM model's training were found by running over 15 evaluations. The model was then trained again with the optimized hyper parameters.

B. DATABASE GENERATION

The datasets [2] consisted of two columns, "emotion" and "pixels". The "emotion" column contained a numeric code ranging from 0 to 6 for the emotion that was present in the image. The "pixels" column contained a string surrounded in quotes for each image. The contents of this string were space-separated pixel values in row major order.

There were a total of 28,709 datapoints for training and a total of 3,589 examples for validation. For evaluation, there were 3,589 examples for which given the "pixels" column , the emotion column was predicted by the model to determine it's accuracy.

Since the training and testing dataset were taken from the same source, the validation couldn't be accurately processed. To that end, another dataset [3] consisting of 700 images was used. The images were processed to crop the face, followed by pixel dimensions resizing, matrix representation and emotion tagging. The database thus generated was tested on the model.

C. VALIDATION AND EVALUATION

The model was tested with a validation accuracy of 47.1 percent and a testing accuracy of 46.4 percent on the first testing dataset [2]. Post optimization of the hyper parameters, the testing accuracy rose to 48.0 percent for the same dataset. The model was cross-validated on the second database [3] with an validation accuracy of 48.2 percent and a testing accuracy of 47.8 percent.

D. FUTURE DEVELOPMENT

In the presence of kinect sensors and other laboratory equipment, database generation can be done for deriving animation units and facial feature points from images as per the refined method described in the paper by Qi-rong et al [4]. The facial landmarks derived from the database in the above model are accurate since they have been detected by dlib library [1] which is also used in Kinect sensors. However, the classification training would be better if a curated dataset of FPPs and AUs were used. In such a case, two robust SVM models would be trained.

III. TEMPORAL EMOTION CLASSIFICATION

In the second phase of the project, temporal emotion detection was looked into. At this stage, the emotional state per frame had already been determined by means of the the SVM classifier. To achieve accurate emotion detection, the following were implemented:

1. A Hidden Markov Model (HMM) classifier with the emotion detected per frame emotion as the observation sequence.
2. A maximum confidence based fusion algorithm.

A. DATABASE GENERATION

In an ideal case, the per frame classification would be performed using the SVM models and these models would be trained on the AUs and FPPs obtained per frame from kinect sensors. However in the absence of laboratory equipment, a modification of an open source detector [5] was used. It takes the video stream as input and returns a dictionary consisting of the breakup of the amount of each emotion type present in the face besides other important facial parameters. Given the breakup of the each emotion type present, the implementation of the above mentioned algorithms became possible and all the databasing was by done by using the detector.

B. MAXIMUM CONFIDENCE ALGORITHM

1) Implementation: Using the open source detector, an independent sub-classifier per emotion consisting of the six basic emotions and the neutral state was setup. Seven output labels representing membership in the class (denoted by adding 1) or out of the class (denoted by subtracting 1) were created to weigh the confidence of each emotion. As given in the paper by by Qi-rong et al [4], the cumulative weight of each class was recorded over the past 30 frames for every new frame and the class with the maximum weight was treated as the dominant class. Accordingly, the dominant emotion was predicted.

2) Future development: In an ideal case, we would have two SVM models as our sub-classifiers, for both of which the cumulative weights would be calculated independently. The final dominant emotion would be the one which would have the maximum cumulative weight from among the maximums obtained for the two models. With appropriate laboratory equipment, the data sets for the model could be generated and testing could be successfully performed.

C. HIDDEN MARKOV MODELS

Having gone through multiple papers, an implementation of Hidden Markov models was performed based on papers by Juan R. Terven et al [6] and Haolin Wei et al [7] These papers were implemented to develop a base model which could leverage the concept of sequence recognition and could then be easily extended to serve as an emotional state detector. In this particular instance, a model was developed to detect head nods based on different facial parameters.

1) Database generation: First, the data dictionary returned by the open source detector was scanned to identify viable parameters. Yaw, pitch and roll were identified as the parameters which were correlated with each other as well as to the action of nodding. Thereafter, exploratory data analysis was performed on these parameters to find the time duration for an average headnod and consequently the length of the sequence over which the Markov model would be trained. Based on the variation of the above mentioned parameters and the average head nod time, an algorithm for generating sequence of length 15 was developed. This algorithm would take video inputs from the open source detector and give data points for the model as outputs. The research papers [6] [7] set the number of training datapoints required to 50 for personal purposes. Hence, 44 datapoints were generated for the purpose of training and 24 datapoints for the purpose of testing.

2) Implementation: The core of the HMM model was based on the Baum-Welch Algorithm which was used to tune the parameters of the HMM model, namely the state transition matrix A, the emission matrix B, and the initial state distribution, such that the model was maximally like the observed sequence of data. The standard implementation of Baum Welch algorithm [8] was updated to take multiple such nod sequences and make the model learn better [9] The model was trained using the above generated dataset and then tested with an pattern recognition accuracy of 71.4 percent.

3) Future development: In the absence of good acting skills for facial-expression dataset generation, the model was developed for head nod detection. However, the availability of dataset would allow for training of models for sequence recognition of the seven different emotions. The advantage of the HMM model is that it predicts the probability, and not a binary 0 or 1, which would allow us to pick the emotional state to which the sequences matches with the most probability.

IV. CONCLUSION

The objective of this paper was to propose a technique to process the facial feature points and animation units of a person's face dynamically to achieve accurate emotion recognition. To that end, it has developed all the models which would be required to achieve the efficient emotion recognition. The pipeline of the detection follows as:

LIVE VIDEO STREAM ->
PROCESSING VIA KINECT SENSORS ->
SVM CLASSIFICATION BASED ON AUs AND FPPs ->
DOMINANT EMOTION DETECTION ->

DYNAMIC PROCESSING BY MORE-ACCURATE(HMM MODEL, MAXIMUM CONFIDENCE ALGORITHM)

V. ACKNOWLEDGEMENTS

The author is extremely grateful to their course professor, Dr. Devpriya Kumar, for his guidance throughout the timeline of the project. The development process would not have been possible without his passionate teaching and guidance. She also expresses her sincere gratitude to IIM Indore for sharing their facial expression database.

REFERENCES

[1] Dlib library at http://dlib.net

[2] Kaggle dataset at https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data

[3] Dataset consisting of 700 datapoints provided by IIM Indore Wesley, Massachusetts, 2nd ed.

[4] Paper titled 'Using Kinect for real-time emotion recognition via facial expressions' by Qi-rong MAO et al

[5] Open source detector given at https://gist.github.com/ahamino/638e59b587c1c9b68b87fe552dd19654

[6] Paper titled 'Robust Head Gestures Recognition for Assistive Technology' by Juan R. Terven et al

[7] Paper titled 'Real time head nod and shake detection for continuous human affect recognition' by Haolin Wei

[8] Baum Welch implementation at https://github.com/adeveloperdiary/HiddenMarkovModel/tree/master/part3

[9] Theory for updating scheme in the "Multiple sequences" section at https://en.wikipedia.org/wiki/Baum