2nd International Symposium on Computer, Communication, Control and Automation (3CA 2013)

EmpaTech: Emotion Recognition Using the Perceptual Computing SDK

Răzvan Rughiniș
Faculty of Automatic Control and Computers
Univ. POLITEHNICA of Bucharest
Bucharest, Romania
razvan.rughinis@cs.pub.ro

Silviu Petria
Faculty of Automatic Control and Computers
Univ. POLITEHNICA of Bucharest
Bucharest, Romania
silviu.petria@gmail.com

George Milescu
Faculty of Automatic Control and Computers
Univ. POLITEHNICA of Bucharest
Bucharest, Romania
george.milescu@cs.pub.ro

***Abstract*—The ability to identify users' emotions represents a valuable asset for improving human-computer interaction. Considering that emotions are conveyed mostly through facial expressions, we have devised EmpaTech, a system that allows the recognition of expressions from a live video feed. Features are extracted using the Intel Perceptual Computing SDK and then are classified using learning algorithms. Classification is based on the Facial Action Coding System introduced by Ekman and Friesen. Promising results were obtained from experimental testing.**

Keywords-face analysis, emotion recognition, Perceptual Computing SDK

I. INTRODUCTION

Emotions and emotional states form an essential part of our lives. Through emotions our mind and body react positively or negatively to external events or stimuli. Our actions are often motivated by emotions. Our experiences are enhanced by them. These aspects remain true when talking about interaction with computers. Anger when dealing with errors or unexpected behavior by an application, joy when achieving a goal in a video game, excitement when making an online purchase, they are all expressions of emotional states that one could tap into and use in order to improve the interaction between a human and a computer.

This paper presents EmpaTech, an application that is able to perform facial recognition and detect one of six basic emotions (happiness, sadness, fear, surprise, anger, disgust) that the user is experiencing. Emotion recognition is performed using facial expressions extracted from three regions of interest (ROI): the eyes, the mouth and the nose. After relevant data is extracted, expressions are classified using a theoretical model and the decision is taken through probabilistic methods.

II. RELATED WORK

Emotion and emotion recognition have been popular subjects of research in the past decades in fields such as psychology, neuroscience, or sociology. After studying preliterate and completely isolated tribes in Papua New Guinea, Ekman famously proposed the theory that emotions are universal and innate and has identified six basic ones: happiness, sadness, anger, disgust, fear, and surprise [1]. He later expanded this list, adding more emotions, some of which are not encoded in facial muscles such as guilt, pride, satisfaction or shame [2]. Another famous Ekman experiment showed that reproducing the expressions associated with a certain emotion caused the subject to actually enter the respective emotional state, providing further evidence of the close link between expressions and emotions [1].

The past two decades have seen a growing trend of research done in the domain of emotion recognition from facial expressions. Most of the work follows a similar model, with focus on a probabilistic algorithm that would achieve a high accuracy rate on a reasonably large data set. Some proposed algorithms can achieve a very high rate of recognition, but the execution time is too long for them to be applied to a live system [3].

The most widely used data set is the Cohn-Kanade database [4], a collection of images depicting various facial expressions performed by 97 subjects. Since the images come in a standard format and size, there is often no need for a face tracking and detection method. This makes the method unsuitable for expression recognition from a video feed.

Cohn, Kanade and Lian [5] have developed a system for recognizing lower and upper face expressions from video sequences using neural networks. This system makes use of both permanent and transient facial features and bases the expressions on the Facial Action Coding System. Facial feature extraction is done using a multi-state facial component model, with focus on the mouth and eyes as permanent features and cheeks and furrows as transient. Still, their face detection method is not 100% automated, with tracking points needed to be set in the first frame. Their neural network system can be adapted to recognize individual unitary expressions or expressions appearing in combination, using the aforementioned Cohn-Kanade database and a separate one, the Ekman-Hager database.

Seyedarabi et al. [6] have designed a system to classify facial movements using contour tracking and cross-correlation tracking of Facial Feature Points (FFP). They are able to detect 25 FFPs automatically in the first frame and then track them in the following frames. Based on the FFP positions in the first and last frames, geometric features are extracted into a feature vector used to classify 16 expressions through a Probabilistic Neural Network (PNN). For detection of the FFPs various image processing algorithms are used, including Canny edge detection. Emotions are determined from expressions using a rule-based system.

Azcarate et al [7] have come up with a system for identifying expressions from live video feeds. Instead of a set of static features of the face, they use feature motions called Motion Units (MU). The MUs describe movement of the lips, mouth corners, eyes, cheeks and eyelids. For classification they employ the Naive Bayes and Tree-Augmented-Naive (TAN) Bayes classifiers, because of their simplicity and good results. The MUs on the face are detected automatically as well using a system based on the famous Viola-Jones method [8]. For tracking they employ a Piecewise Bezier Volume Deformation tracker.

Naive Bayes classifiers were also used with a good degree of success by Sebe et al [9] together with Gaussian TAN classifiers as part of a static approach in detecting facial expressions. They also proposed a dynamic model that used temporal information to discriminate between expressions for more accurate classification results.

III. Facial Action Coding System

FACS is a widely known system for classifying facial muscle movements. It was designed and popularized by Ekman and Friesen in 1978 and updated in 2002 in collaboration with Hager [10]. Based on a system initially used in anatomy, FACS attempts to comprehensively separate expressions based on the muscle or group of muscles that trigger them and their impact on the appearance. It describes each of the instant changes to the human face as an Action Unit (AU). AUs are independent of interpretation, can occur in isolation or in combination and some applications consider the level of intensity for each AU. Typically, each AU is triggered by the contraction or relaxation of a muscle. The full set of AUs includes 46 movements [11] (see illustrations in Table 2).

After closely studying the connection between emotions and facial expressions, Ekman decided to assign each of the basic emotions he identified (happiness, sadness, anger, disgust, fear, surprise) a set of AUs. Based on the expressions produced by each emotion, we conclude that a number of eight AUs are necessary and sufficient in order to identify the basic emotions minus happiness (which will be classified by detecting smiles).

TABLE I. Action Units and Basic Emotions

Emotion	Mouth AUs	Eye AUs	Nose AUs
Sadness	AU15	AU1	-
Anger	AU23	AU4,5	-
Disgust	AU15	-	AU9
Surprise/Fear	AU26	Au2	-
Neutral	AU0	AU0	AU0

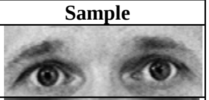
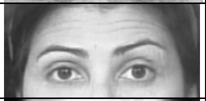
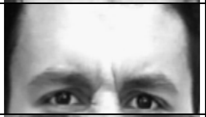
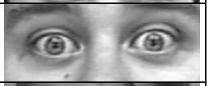
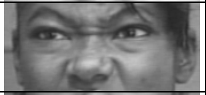
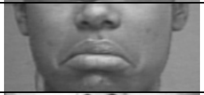
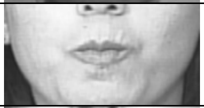
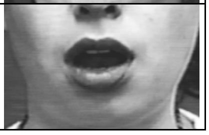
We note that surprise and fear are very similar to the point of being virtually indistinguishable by the naked eye and therefore we will consider them the same emotion for the purpose of classification. A differentiation between them can be done using duration (a surprised expression tends to be very brief). The neutral state of the face (AU 0) will be used for comparison and therefore is considered one of the different emotions we need to be able to identify.

IV. The Perceptual Computing SDK

The Perceptual Computing SDK was launched by Intel as a beta in October 2012 and got its full release in March 2013. The two main components of the SDK are the gesture analysis module and the face analysis module. The application makes exclusive use of the face analysis module. The high level face analysis API provided by the SDK offers four major functionalities: face detection; face recognition; landmark detection (corners of the mouth, corners of the eyes, nose tip); face attributes (gender and age group, smiles, eyes open / closed).

The main hardware component of the system is the Creative Interactive Gesture Camera. It was released to the public in June 2013 as the Creative Senz3D Peripheral Camera. It is a small portable camera, powered through the USB cable, and it is the recommended camera to use with the SDK.

TABLE II. Action Unit Samples

Action Unit ID.	Name	Sample
1	Inner Brow Raiser	
2	Outer Brow Raiser	
4	Brow Lowerer	
5	Upper Lid Raiser	
9	Nose Wrinkler	
15	Lip Corner Depressor	
23	Lip Tightener	
26	Jaw Drop	

Source: P. Ekman, W. Friesen, “Facial Action Coding System”. Retrieved from www.cs.cmu.edu/~face/facs.htm

V. EmpaTech Architecture

In order to perform facial expression recognition, we acquire data from the live video feed provided by the camera, we run it through image processing filters and algorithms and we input it into a classifier so that a decision can be made. Therefore, at the simplest level, we can distinguish three main stages of the application workflow:

- Data acquisition: 1) acquire raw RGB data from the camera input stream ; 2) store the data for the next step;
- Data processing: 1) identify the face region; 2) crop the areas of interest (eyes, mouth, nose); 3) convert images to grayscale; 4) apply image filters;
- Classification; 1) calculate projection profile for the ROI samples; 2) offline stage: train the classifier; 3) online stage: load a model obtained after training and use it in order to identify the AUs present in the live video feed, as well as the smiles (using the face attribute detection of the Perceptual Computing SDK).

The Gaussian Blur and Sobel edge detection [3] filters are applied to the collected samples. Figure 1 illustrates the resulting images.

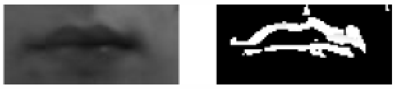


Figure 1. Mouth area sample pre- and post-filtering

One of the biggest issues that affect this method is uneven lighting. Under various lighting conditions the same image will produce different results. Light being more powerful on one side of the picture is also a problem, since it produces asymmetrical results. To solve this, we have experimented with different thresholds and we have settled on two separate values, one for well sunlit environments and one for lower intensity artificial light. Values of 70 for the first case and 25 for the second have produced virtually identical results. In the context of the current prototype, the user is prompted to select the light conditions at the start of the application. In the future we plan to implement a method that does this automatically.

VI. Image Processing

OpenCV is a free for use cross-platform library mainly aimed at real time computer vision. It offers a variety of image and video processing algorithms together with a powerful machine learning module. We use OpenCV functions for the image processing stage of the application.

VII. Classification

After the final results of the image processing stage are ready, we can apply a feature extraction method. In this case, the chosen method is projection profile [12]. It is based on the row sum of white pixels in the image. This pattern will constitute the feature of the area and it will be used for classification. If S (n, m) is the edge detected image of m columns and n rows, the row profile is given by a vector P of size n:

$$p_j = \sum_{i=0}^{m-1} s\,(j,i), j = 1 \ldots n$$

Projection profile extraction therefore produces an array of figures which correspond to the number of white pixels on each row of the image. This array can be considered a fingerprint of the image. Projection profile has been often used with success in similar problems such as handwriting or letter recognition from black and white images.

EmpaTech relies on a Naïve Bayes Classifier. OpenCV's machine learning module provides an implementation called NormalBayesClassifier. It offers the following important functions:

- bool CvNormalBayesClassifier::train(const Mat& trainData, const Mat& responses, const Mat& varIdx = Mat(), const Mat &sampleIdx = Mat(), bool update=false)
- float CvNormalBayesClassifier::predict(const Mat& samples, Mat* results=0)

The train function is used to determine the parameters of the classifier. Each image sample is described by an array of integers representing the row projection profile of that image. In the training phase, we store the arrays in a comma separated values (CSV) file together. We also add the correct class for each sample (the Action Unit it portrays). The arrays are loaded into a matrix and passed to the train function. The function will proceed to train the model, which can then be saved on the disk and loaded. The predict function is used in the testing phase. Once the classifier is trained (or loaded with a previously saved model), we feed new samples and check the results. The training of the model is done offline. After collecting a data set of samples, a classifier is trained. The model is then saved and loaded in the live application where the prediction is made. Figure 8 shows a diagram of the workflow.

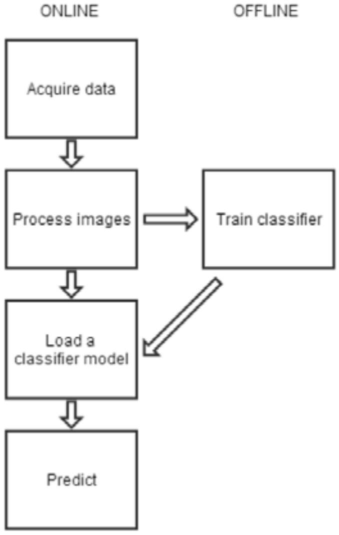


Figure 2. EmpaTech Classification Workflow

VIII. Experimental Results

The data set consists of ten filtered image samples for each AU necessary in determining the emotions we wish to classify and ten image samples for each neutral state (eyes, mouth, and nose). There are in total 50 images for the eye region, 40 for the mouth region and 20 for the nose region. We have conducted classifications using two different splits of a person-dependent data set (the samples used for testing come from the same person as those used for training). In the first experiment, we have used 60% of the samples for each AU in the training stage and 40% for testing. The split for the second experiment was 80-20.

TABLE III. ACCURACY RATES IN EMPATECH EXPERIMENTAL TESTING

Training / testing ration	Experiment	Accuracy rate (%)
60 / 40	Eye region	75.0
	Mouth region	62.5
	Nose region	100.0
80 / 20	Eye region	80.0
	Mouth region	87.5
	Nose region	100.0

We noticed that the most difficult AUs to classify are those that represent sadness (AU 1 and AU 15) since they are the most difficult to willingly reproduce. AU 1 (inner brow raiser) is often confused with AU 5 (upper lid raiser) while AUs 15 and 23 (lip corner depressor and lip tightener) are also very similar. AUs 2, 4, 5 and 26 are usually well differentiated. The accuracy for the nose region is 100% in both cases, which is to be expected given the low number of classes. The accuracy, as expected, is higher for the 80-20 split.

The detection of the correspondent AUs determines whether or not the emotion is present. Since the mouth has been found to be more emotive than the eyes [13], in case of conflict the mouth AU takes precedence.

IX. CONCLUSIONS

We have designed EmpaTech, a system that allows us to recognize a set of emotions from the facial expressions of a person using facial landmark detection, feature extraction, and classification. The face detection and tracking phase is fully automated thanks to the facilities offered by the face analysis module of the Perceptual Computing SDK. Face information is extracted from RGB frames after face landmarks are detected, with the ROIs being the eyes, the mouth, and the nose. The classification is based on the Facial Action Coding System, a comprehensive model that describes unitary facial muscle movements named Action Units by their appearance on the face. In the offline stage, samples of Action Units are used to train a Naive Bayes Classifier and a probabilistic model is saved. The same model is later used in live prediction of emotions based on Action Units.

We believe further improvements to the system are possible. The naive Bayes classifier could be replaced by a tree augmented naive Bayes classifier to take into account dependencies amongst features. Further work could be done to reduce to impact of the lighting conditions and make the system more robust. The depth sensor of the camera offers information that could be used in improving the system by taking into account elements of pose, orientation (present in the SDK but not yet implemented); it could help eliminate the influence of lighting conditions as well. Also, extensive training could make the system more person-independent and able to adapt to more types of facial appearance.

It is worth noting that the Perceptual Computing SDK is still work in progress, with more libraries and features to be added. With these improvements, the EmpaTech application could be used for various forms of human-computer interaction, such as games, chat tools or other forms of interactive applications.

REFERENCES

[1] P. Ekman, “Emotions Revealed: Recognizing Faces and Feelings to Improve Communication and Emotional Life”, Phoenix, USA, 2004.

[2] P. Ekman, “Basic Emotions”, San Francisco, USA, 1999.

[3] M.A. Oskuyee, “Evaluation of Optimization Methods Ant Colony and Imperialist Competitive Algorithm in Face Emotion Recognition”, International Journal of Advanced Research in Computer Science, Volume 3, No. 1, Jan-Feb 2012.

[4] T. Kanade, J.F. Cohn, “Cohn-Kanade AU-Coded Facial Expression Database”, Carnegie-Mellon University.

[5] Y. Tian, T. Kanade, J.F. Cohn, “Recognizing Action Units for Facial Expression Analysis”, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (2), pp. 97-115, 2001.

[6] H. Seyedarabi, W.S. Lee, A. Aghagolzadeh, S. Khanmohammadi, “Classification of Upper and Lower Face Action Units and Facial Expressions using Hybrid Tracking System and Probabilistic Neural Networks”, 5th WSEAS International Conference on Signal Processing, Istanbul, Turkey, 2006.

[7] A. Azcarate, F. Hageloh, K. van de Sande, R. Valenti. “Automatic facial emotion recognition”, Amsterdam, Netherlands, 2005.

[8] P. Viola, M. Jones, “Rapid Object Detection Using a Boosted Cascade of Simple Features”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511-518, Hawaii, USA, 2001.

[9] N. Sebe, I. Cohen, A. Garg, M.S. Lew, T.S. Huang, “Emotion Recognition Using a Cauchy Naive Bayes Classifier”, International Conference on Pattern Recognition (ICPR02), vol. I, pp. 17-20, Quebec, Canada, 2002.

[10] P. Ekman, W. Friesen, “Facial Action Coding System: A Technique for the Measurement of Facial Movement”, Consulting Psychologists Press, Palo Alto, USA, 1978

[11] P. Ekman, W. Friesen, “Facial Action Coding System”. Retrieved from http://www.cs.cmu.edu/~face/facs.htm. Last visited: June 30th, 2013.

[12] M. Karthigayan, M. Rizon, R. Nagarajan, S. Yaacob, “Genetic Algorithm and Neural Network for Face Emotion Recognition”, Affective Computing, Jimmy Or (Ed.). 2008.

[13] M. Lucia, J. Pandya, T. Zajdel, “Emotion Detection”, CSE: 634, Computer Vision, 2010.