Expression Recognition System: Read Me!

Eric DeFelice, Yasemin Ersoy, Kevin Zhou
Department of Electrical Engineering
Stanford University
{edefelic,yasersoy,zk2zhou}@stanford.edu

***Abstract*—Expression recognition is increasingly being used in the mobile and robotics industry. A robust application to recognize certain facial expressions in real time has many obstacles starting from correctly identifying a face and keeping track of key points on the face to then mapping these points to the right expression. Through the cascaded use of feature detection techniques, adaptive histogram equalization, morphological processing, and filtering this application recognizes 15 key points on the face and compares them to a calibrated neutral face model to extract the current expression. This application can be used to indicate an emotion to those that cannot comprehend expression or to react to the detected expression by playing music based on the users' current mode, detecting if the user is lying, and allowing robots to have human-like interaction with the user.**

Keywords—facial expression recognition; edge detection; face recognition; emotion recognition

I. INTRODUCTION

Real time facial expression recognition involves detection of a face in the current video frame, followed by the identification of key points, such as corners of the mouth, which are analyzed to best fit an emotion that is outputted by the application. Facial expression recognition has been studied since the 1980's and has been spreading to a variety of fields such as psychology, art, robotics, and computer vision. Though initially studies targeted those with impaired understanding of emotions and lie detection, with our current technology these valuable expressions can be utilized to create a more fluid user interface.

Cameras surround our world, and with the development of image processing even the resolution cameras on laptops or phones is enough to robustly extract the necessary key points for expression recognition.

Our application is an interactive prototype that can be run using MATLAB with a laptop camera to detect happiness, sadness, anger and surprise. First images are captured through the camera, and then the face is identified. Once the face is identified, the key points are detected and calibration is run for each individual to capture the neutral face and produce optimal results. After calibration the emotion that is detected along with lower possibility emotions are displayed to the user (Figure 1).

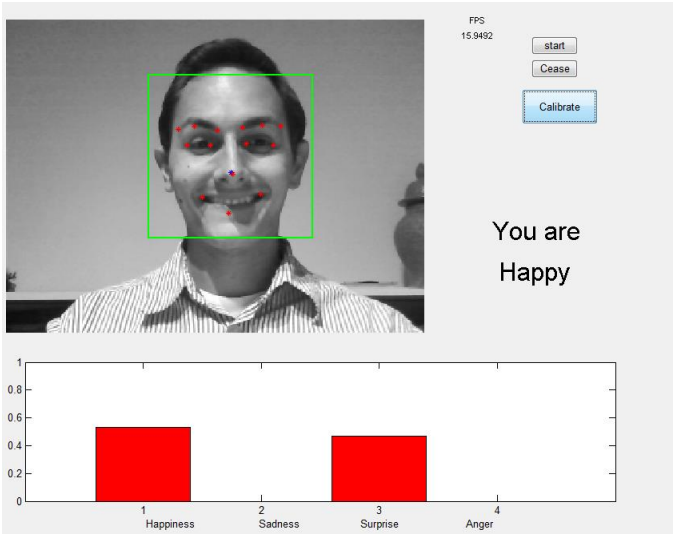

Figure 1: Read Me! User Interface

II. PROCESS FLOW

A. *Detecting Face*

Our application first detects the general frontal face area by putting a box around it. Subsequently, this region of interest gets pass into a more robust algorithm block that detects facial key points.

The face detection/recognition works by applying several simply classifiers to a region sequentially to reject all unmatched classifiers. These classifiers are combinations of simple contrast features such as edges, lines, and holes in different orientations (Figure 2). Each of the features are Haar-like features because they are computed similar to the coefficients in Haar wavelet transforms [1]. When the search window is slid across the entire image, the spatial relationship of these features are fed into a decision-tree with at least 2 leaves. The matched region will be considered as region of interest and gets passed into the next algorithm block. The number of chosen regions is limited to 1, and is indicated by a green box, as shown in Figure 1.

For the purpose of our project, we used haarcascade_frontalface_alt2.xml (part of the OpenCV facedetect package) as our decision-tree classifiers, and this was trained with 200+ positive examples of frontal faces.

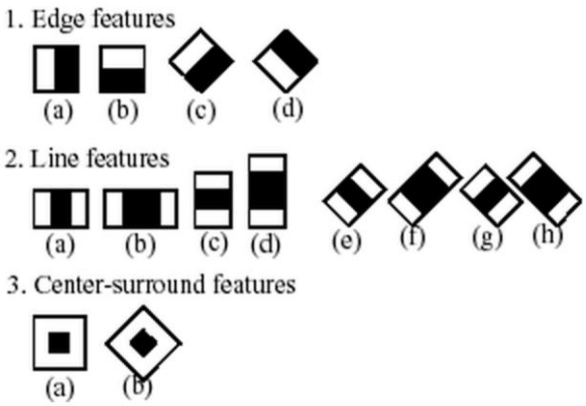

Figure 2: Contrast Features

To lower the load of the search, we have subsampled each frame by 2, and have converted the input from RGB to a grayscale image. To combat noise, non-uniformity, we have equalized the histogram of the image, used adaptive thresholding and histogram equalization on the detected bounding box.

Finally, we have limited the size of the frontal face bounding box by 40x40 and 200x200, which roughly translates to 30cm to 120cm in distance.

B. Collecting Facial Key Points

Once the main user face is detected, the key points can start being detected. This is a very crucial step in the processing chain, as the facial key points need to be detected quickly enough to allow for a real-time video application, but also need to be accurate enough to be used for expression recognition. For this project, we decided to utilize the flandmark facial landmark detector [2] for its performance and speed. The flandmark package detects eight key points (two for corners of each eye, one for tip of nose, two for corners of mouth, and one for center of face) by using Deformable Part Models (DPM) that has been trained by Stuctured Output Support Vector Machines (SO-SVM) [2]. The package also contains functionality to allow for supervised learning of the model parameters, but for our case, the model provided with the framework is sufficient, as we are assuming a front facing full face in acceptable lighting conditions.

Once these first eight key points are determined, the remaining seven necessary feature landmarks are collected by using a cascaded processing pipeline approach. The stages in the pipeline combine some of the region segmentation techniques in [3] and [4] with various filtering techniques to map the key points.

C. Additional Key Points and Optimization

The initial eight key points that are outputed from the flandmark package [2] are quite accurate, and provide a good starting point for detecting further key points on the current detected face. However, additional key points are needed to improve the accuracy of the expression recognition, as much of the expression model is contained in the information from the eyebrow location or the wrinkles between the eyes, for example. Since we already have eight key points, we use these points as *a priori* information when looking for further facial landmarks. For this project, we used 7 additional key points: three for each eyebrow (one for each corner and one for the center), and one for the center of the lower lip. A couple different facial key point algorithms were examined, and their performance was analyzed.

The first step after the initial eight key points are detected is to find the position of the eyebrows. To perform this task, the positions of the eyes, as well as the estimated size of the detected face are used as inputs to the processing pipeline. Using this input data, the forehead region is first cropped from the face bounding box. This new forehead image is used in the processing chain, as opposed to the full face image, to improve speed and accuracy. A representative example of one such image is shown in Figure 3.

Figure 3: Contrast Features

The eyebrows are always contained in this image, because it is adaptive with respect to the bounding box size and the eye locations.

Since a couple different algorithms were examined, they will be described here separately, followed by the performance of each algorithm set. For algorithm set 1, a region based approach was used to detect the eyebrows. The first stage of the pipeline takes the difference between all of the pixels in the y-direction. This results in an image where the horizontal edges have been detected. A threshold is then applied to this edge image, where any pixel value above the mean is kept. This process filters out many of the unwanted regions, but still requires some further processing. Finally the logical image is labeled and the regions are grouped and filtered by size, orientation, and solidity. After the region filtering, only the regions that have a similar shape and size of the eyebrows remain. Out of these regions, the largest ones that have a centroid above one of the eyes are determined to be the eyebrows.

This approach is fairly accurate on static faces, but does not perform quite as well when presented with lighting changes or noisy conditions. This approach is also slower than the approach used in algorithm set 2, because it relies on searching through many different image regions to filter.

The approach used in algorithm set 2 for the eyebrow detection is more efficient and slightly more accurate. This approach starts with the same cropped forehead image, but then performs adaptive histogram equalization followed by median filtering with a 4x4 pixel window. These two functions give the algorithm some resilience against adverse lighting conditions and provide some noise suppression. This improves the accuracy of eyebrow detection, especially when there is some motion blur and shadowing effects. The edges are now

detected using difference-of-box filtering, with 4x2 and 8x2 windows. These box sizes were chosen to mainly detect the horizontal edges, which is the main feature that differentiates the eyebrows. This resulting image is then adjusted to have a zero-mean, and thresholded. There are fewer regions in the logical image because of the pre-processing, and the filtering is much quicker to find the eyebrow regions. Figure 4 shows the resulting images for the intermediate steps of the processing pipeline.

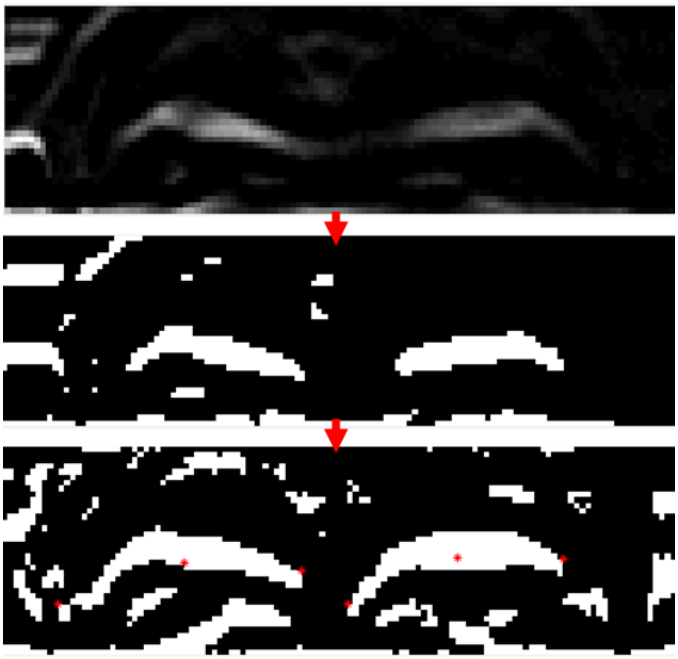

Figure 4: Eyebrow detection processing pipeline

The final key point that is needed is the center of the lower lip. Two separate approaches were also investigated to detect this feature. The two different algorithms were lumped in with the eyebrow detection filters to form 'algorithm set 1' and 'algorithm set 2'. The first approach for lip detection again uses a region based technique. A difference image is generated, and then a threshold is applied to create a logical image of the mouth area. The regions are then filtered and the largest region is chosen as the lower lip.

The second approach uses a more cascaded filtering technique. The first stage performs adaptive histogram equalization and median filtering to account for lighting variations and random or blurred noise. The second stage applies two different operators to the image, filtering it in the y-direction. The difference is taken between the two filtering results, and the difference image is then adjusted to have zero-mean. The maximum value after this operation is determined to be the lower lip.

The performance of the two algorithm sets was looked at, and the optimal technique was chosen for final implementation. The key performance parameters were the speed at which all of the key points were detected, and the accuracy of these key points to the actual features. The accuracy was measured as the variance of key point locations when mapped to a static face. This is done during the calibration process for our project. Figure 5 shows the average speed and accuracy for each algorithm set, over the span of 200 frames.

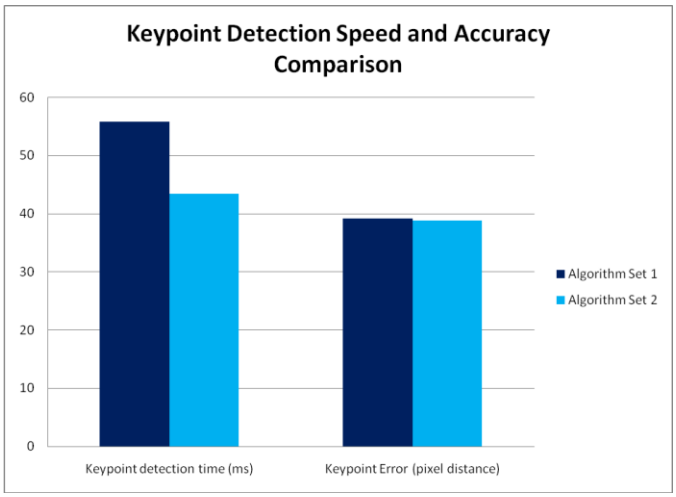

Figure 5: Performance of key point detection algorithms

It is seen in the figure above that 'algorithm set 2' decreases the detection time by about 15 ms (~28% improvement) and has very similar accuracy.

D. Expression Recognition

The expression recognition part of the algorithm uses the 15 key points and bounding box of the face for detection. These points are used to calculate the angles of the face as shown in Figure 6 and Figure 7.

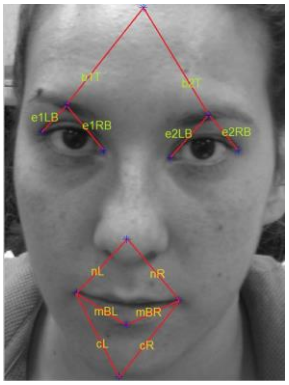

Figure 6: Neutral Face Angles

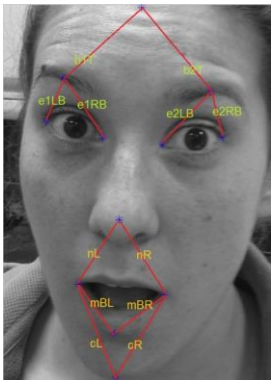

Figure 7: Surprised Face Angles

The lower face angles surrounding the mouth are more relevant indicators for happiness and sadness and the upper face angles regarding the eyebrows are more prominent in detecting anger and surprise [5].

Using angles rather than distances allows different sized faces to be treated the same and is a novel and speed optimized method based on past expression detection methods which look at the direction of face flow, distance of features, and relative facial key point location training [5, 6, 7]. Knowing the neutral angles is the base for using this method of expression recognition. However, each person has a different neutral face and it is best to measure the difference of the expression angles relative to the correct subject's neutral expression angles. The variation of expression angles for a given expression varies far less when dealing with a single person. Below is an example of expression angles of a single subject (Figure 8) and variation of 5 different subjects (Figure 9).

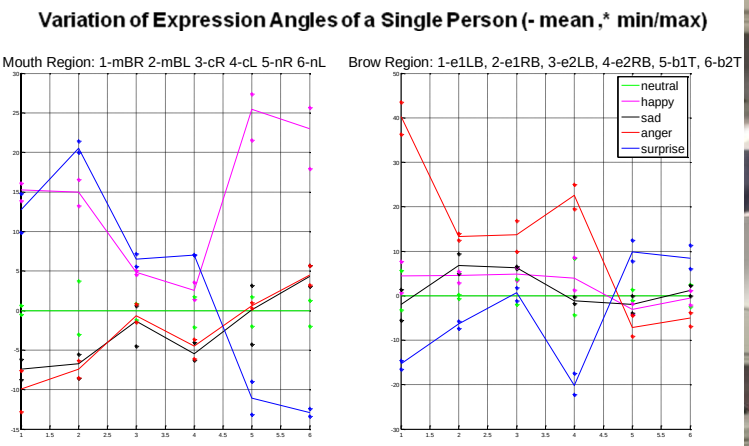

Figure 8: Single Subject Variation

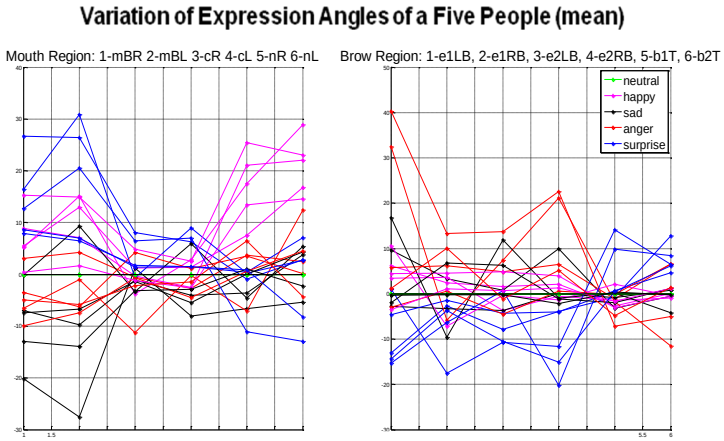

Figure 9: 5 Subject Variation

The thresholds for each expression were chosen by training using a collected database of faces (examples in Figure 10). The most prominent angles for each expression are given the heaviest weighting for the expression where prominent is defined as the least overlapping points with another expression and greatest change from neutral. The collected data showed that combinations of angles are also useful identifiers such as the first 4 angles in the lower face of surprise being positively different with respect to neutral while the last two are negatively related (Figure 8).

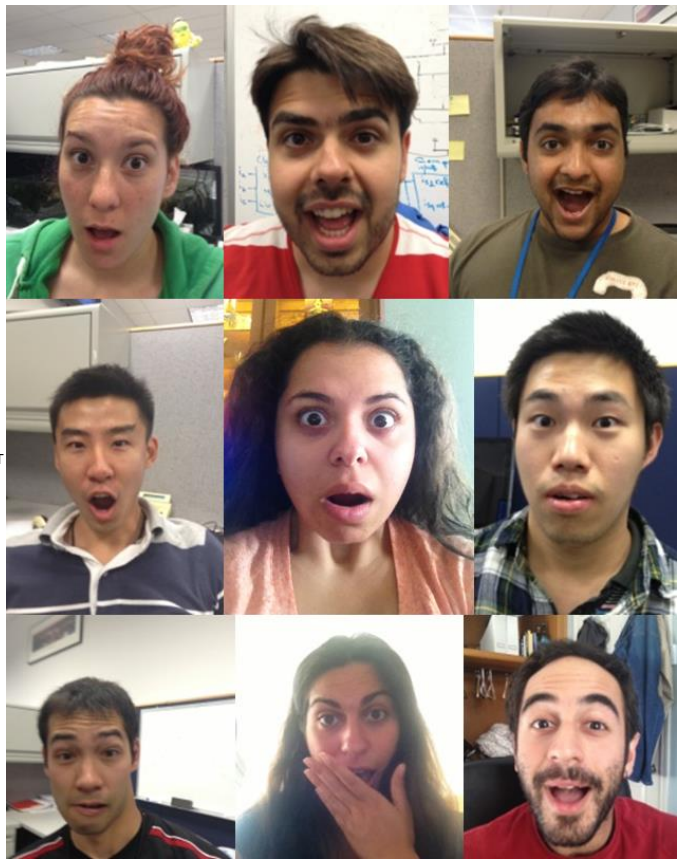

Figure 10: Sample Surprised Subjects

Given the face angles and using the calculated thresholds for each expression, facial expression at each instance is calculated as follows:

1) Add chin and center top of forehead to key points using face bounding box, eyebrow center and lip corner keypoints.

2) Calculate facial expresion angle using all key points

3) Check if past expression was inputed

- Yes: If past angles are within a threshold to equal new input, expression has not changed - output past expression
- No: Continue to face detection

4) Check if calibrating

- Yes: Record current expression angles as neutral (Figure 6) and check if previous average neutral angles were imported: if so average the two neutral angle sets and output as new neutral expression angle set
- No: Continue to face detection

5) Detect expression

a) Upper Face

a. Compare upper face thresholds to calculated emotion thresholds of eyebrows

b. If anger and surprise are detected in upper face give a higher weight

c. Check if movement is in all negative or all positive direction

b) Lower Face
   a. Weigh mouth corners to nose heavily for happiness
   b. Weigh mouth corners to bottom lip middle heavily for sadness
   c. Check for angle thresholds for surprise and anger and general direction of mouth
c) If the maximum weighted expression is not strong enough label face neutral
d) If contrasting emotions are detected (ie angry eyebrows with smiling face) label as confused
e) If two emotions are equally rated, which happens in anger and sadness quite often, further differentiate the two expressions
   a. Shape of eyebrow movement relative to eyes and top of forehead to differentiate sadness and anger
   b. General direction of mouth relative to nose and chin to differentiate happiness from surprise and sadness from anger

6) Output detected expression along with the probability of detecting each expression.

III. Experimental Results

A. Key Point Optimization

The speed and accuracy of the key point detection stage of the application was analyzed so that the algorithm providing the best performance could be used in the final implementation. Two separate key point detection algorithms were looked at, each consisting of an eyebrow detector and a lower lip detector. The detection time for the key points was compared across 200 sample frames. Figure 11 shows the detection time for the two different algorithm sets.

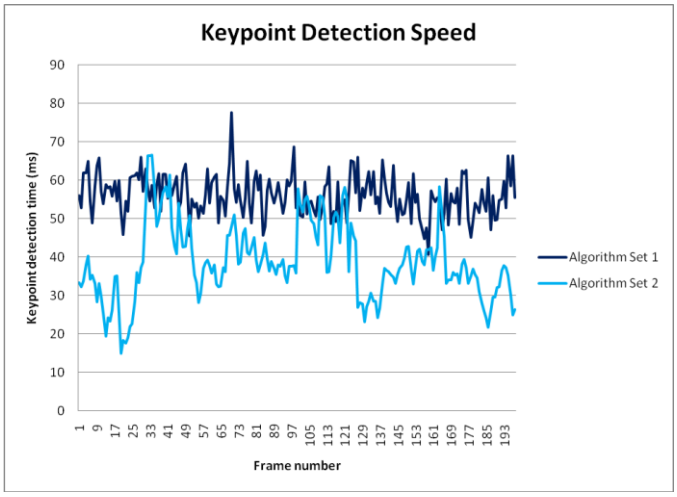

Figure 11: Key point detection time for each algorithm set across 200 sample frames

The accuracy of the two different algorithm sets was also compared using a static neutral face over 200 frames. The accuracy is measured as the sum of all of the distances for the key points from their neutral face average, taking during a calibration cycle. In the ideal case, these key points would not move during that time, because the features of the detected face are not moving. The relative accuracy for each of the algorithm sets is shown in Figure 12.

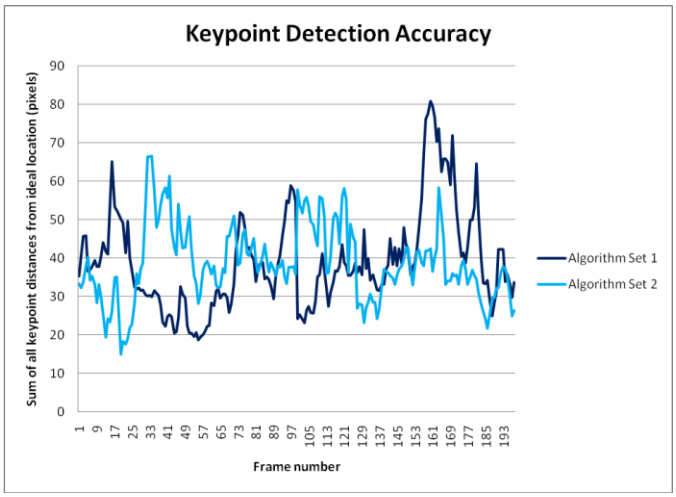

Figure 12: Relative accuracy for each of the algorithm sets across 200 sample frames, taken during a calibration cycle.

The two algorithm sets have similar error rates when looking at a static neutral face, but the noise suppression in 'algorithm set 2' does give it an advantage when motion blur is present. There is also a ~28% speed improvement when the key points are detected using 'algorithm set 2'.

B. Emotion Recognition Accuracy

The goal of this expression recognition application was to be achieved with plenty of lighting and a frontal face since the algorithm is running in real time, so the training sets used were close to these standards. The accuracy of the emotion detection was tested on all emotions of a diverse collection of more than 13 faces with manual key point detection to insure that errors were not caused by shifted key points (Figure 13).

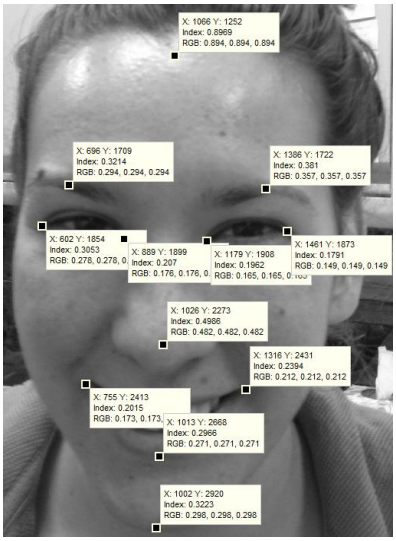

Figure 13: Manual key point emotion testing

Happiness and surprise were always recognized correctly after neutral calibration while sad and anger did output incorrect detections at times (Table I). Since sadness and

anger overlap for some faces and sadness greatly resembles neutral if the frowning mouth is not accentuated this was expected. To improve our implementation more research on characterization can be done on the entire mouth region because subjects have different upper face angles when expressing levels of sadness. Full mouth characterization along with full eye characterization can dramatically improve sadness and anger detection but might be too slow for our implementation. Results are very promising considering the speed in which they are generated.

TABLE I. EXPRESSION RECOGNITION ACCURACY

Detected Emotion	**Actual Emotion**			
	Happy	***Sad***	***Anger***	***Surprise***
Happy	**100%**	0	0	0
Sad	0	**65%**	12.5%	0
Anger	0	20%	**75%**	0
Surprise	0	0	0	**100%**
Confused or Neutral	0	15%	12.5%	0

IV. CONCLUSION

We were able to overcome many of the challenges of detecting facial expressions in real time by correctly identifying faces through a laptop camera input, capturing and following important key points of the face and dynamically mapping these points to the correct expression. Our application has been tested in different lighting environments and on different faces and it fairs well even in dimmer rooms. The level of processing that goes into creating distinct key points allows for accurate expression detection.

Our application can be further improved by more robust eyebrow and lower lip key point detection along with calibrating for the variations of the face bounding box; plus, the emotion detection of sadness and anger can greatly improve by further training for more precise thresholds and full feature characterization. Though there is a margin of error it is still pleasant to see this prototype functioning in real time. These expressions detected in real-time can be bifacial to many applications such as robot human interaction and in aiding those that cannot read faces.

V. WORK BREAKDOWN

Eric DeFelice: Key point detection algorithm development and analysis, initial expression detection, poster, and final report writing.

Kevin Zhou: Face detection implementation, Matlab GUI development, poster, and final report writing.

Yasemin Ersoy: Expression detection algorithm design and analysis, poster, and final report writing.

ACKNOWLEDGMENT

Thanks to Huizhong Chen for algorithm support and Pamela Davis for expression modeling support.

REFERENCES

[1] Rainer Lienhart, et al. (2002). "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection," Microprocessor Research Lab, Intel Labs

[2] M. Uricar, V. Franc and V. Hlavac, Detector of Facial Landmarks Learned by the Structured Output SVM, *VISAPP '12: Proceedings of the 7th International Conference on Computer Vision Theory and Applications*, 2012

[3] Tianxiang Yao; Hongdong Li; Guangyao Liu; Xiuqing Ye; Weikang Gu; Yiqing Jin, "A fast and robust face location and feature extraction system," *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol.1, no., pp.I-157,I-160 vol.1, 2002

[4] Li Ke; Jingjing Kang, "Eye location method based on Haar features," *Image and Signal Processing (CISP), 2010 3rd International Congress on* , vol.2, no., pp.925,929, 16-18 Oct. 2010

[5] Neeta Sarode, Shalini Bhatia. Facial Expression Recognition. *International Journal on Computer Science and Engineering 2*, 2010, p. 1552-1557.

[6] Wu, Yuwen, Hong Liu, and Hongbin Zha. "Modeling facial expression space for recognition." *Intelligent Robots and Systems*, 2005.(IROS 2005).

[7] Yang, Y., S. Ge, et al. (2008). "Facial expression recognition and tracking for intelligent human-robot interaction." *Intelligent Service Robotics 1*(2): 143-157.