SEIS 772 Final Project Report

Image Processing Techniques to Recognize Facial Expressions  
Team members: Princewill Eneh, Sona Maniyan and Alex Gelbutovskiy

**Table of Content**

1. Introduction
   1. Why do this
   2. Goal of the project
   3. What do we explore to achieve our goal. How do we solve the problem.
2. Project Overview
   1. Your project design
   2. Reason for the design
   3. Dataset
      1. what is the dataset
      2. why did we chose this dataset
   4. workflow
3. Preliminary Analysis
   1. Why we picked the mouth
4. Tools
5. Algorithms
   1. Viola Jones Algorithms
   2. SURF (Speeded up Robust Feature)
   3. Neural Networks
6. Workflow (include diagram)
7. Pre-processing
   1. Filtering
   2. Categorizing
   3. Isolate the face region
8. Heuristics: Polynomial Coefficients and Tortuosity
   1. Extract Relevant facial features
   2. Remove Noise
      1. bwareopen
      2. structural elements
      3. Spur
      4. dilate
   3. Acquire Feature points
      1. Coefficients for the line of best fit (Laplacian Algorithm)
      2. Tortuosity
   4. Resultant ANN Dataset
9. Test Predictive Algorithm
10. Challenges
    1. Issues
    2. Solutions
11. Lessons learned
12. Applications
13. Future Research
14. Conclusion
15. Annotated Bibliography

***Abstract:*** *The objective of our project is to provide a framework to identify what facial expression a person in an image is displaying, using image processing and analytic techniques.*

1. **Introduction**

Emotions are an incredibly important aspect of human life and recognizing emotions is important in many aspects of daily life when interacting with people. In our modern age, human computer interaction is increasingly common and adding the capability for computers to recognize emotions can add a new dimension to this interaction.

Our project has attempted the task of building a classifier that would predict which emotion is displayed on a still frontal image of a face. The system initially extracts the mouth region and then applies algorithms to measure tortuosity and extract coordinates indicating the curvature of the lip region. A simple neural network is used to classify facial expression to the proper emotional state. Due to the time limitations, this project currently only explores two emotions - ‘happy’ and ‘angry’.

1. **Project Overview**

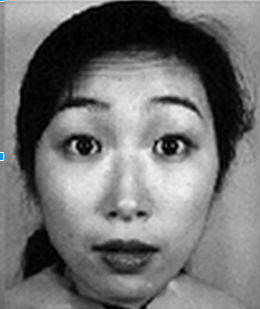
**Objective:** The objective of this project was to create a classifier that could predict facial emotion appearing in a frontal image of a subject based on features extracted from the image.

**Dataset:** The source for the raw database was the Japanese Female Facial Expression (JAFFE) Database   
<http://www.kasrl.org/jaffe.html>. The database contains 213 images that have 60 subjects expressing a range of emotions. There are six emotions expressed by the subjects and a neutral expression. The six categories are surprise, sad, disgust, fear, happy and angry.

1. **Preliminary Analysis**

For us to be able to differentiate a facial expression, we have to know what facial features change positions when an expression is shown. Identifying these facial features will help us narrow our test to those specific areas of the face.

When a person smiles or displays any kind of expression, the mouth and the eyes change significantly, while the nose remains fairly static in most cases (see image below). Knowing this, We will only focus our test and analysis on the mouth region.



1. **Tools**

The main tools used for processing the data and creating the Artifical Neural Network (ANN) were -

**Matlab -** Computer Vision Package

**R -** nnet

**-** ROCR

1. **Algorithms**

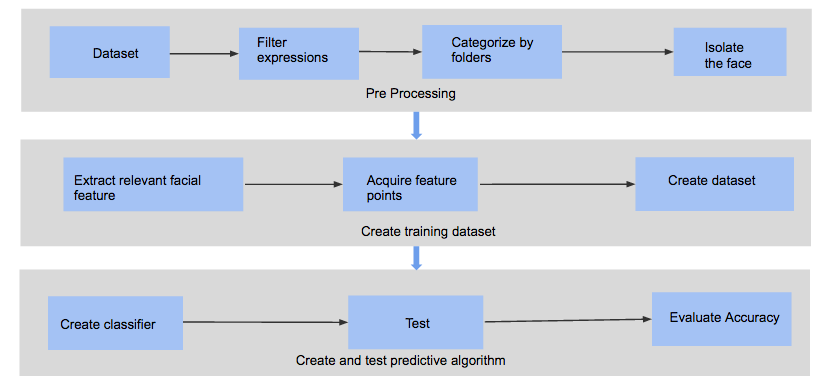
The main algorithms which supported the extraction of regions for further processing was the Viola Jones Algorithm. Using an implementation of this algorithm in Matlab it was possible to extract the face region in a frontal picture and also the mouth region.

Interesting feature points in the extracted mouth region were detected using the SURF (Speeded Up Robust Feature) Algorithm. Matlab’s implementation was used and the key points of interest were detected using the sign of Laplacian. The sign of the Laplacian distinguishes between bright blobs on

dark backgrounds and vice versa.

We used Artificial Neural Networks for pattern recognition and classification. A simple neural network was build using R’s nnet package.

1. **Workflow**



We plan to be systematic in our approach to this project. The approach we adopted was to divide the workflow into three different phases with each face dealing with a specific challenge to achieving our final goal of facial expression detection.

In the preprocessing phase we collect our dataset, filters the expression we are interested in analyze and categorize them for later analysis. Next we isolate the faces from the background noise.

The Isolated face is now sent to the next phase of our project which is the phase where we do our analysis and train our dataset. In this phase we also extract relevant facial features and acquire feature point which will be used to train our dataset.

1. **Pre-processing**

**Filtering**

Like we indicated above, the preprocessing phase is where we select two basic facial expressions happy and angry from seven expression in the original dataset. These are the expressions we are interested in analysing because of the scope and time available to us.

**Categorizing**

The images with two expressions we are looking for are categorized into two folders.

**Isolate the Face Region**

The face is isolated and all other features are treated as noise. 

1. **Heuristics: Polynomial Coefficient and Tortuosity**

As discussed earlier, our goal is to train ANN which would seamlessly distinguish a smile from a frown, i.e. happy and angry expressions. Thus, the appropriately chosen training dataset would be a determining factor on the overall performance and veracity of ANN. Input dataset uses two heuristics (metrics): 5 coefficients from the polynomial of degree 4 and tortuosity of the lip region.

**8.1 Extract Relevant facial features**

Viola-Jones algorithm identifies facial features. The pre-trained MATLAB’s cascade object detector helps to isolate bounding boxes associated with specific facial regions (**mouth**, nose, eyes). Indeed, we can crop out these regions for further analysis.

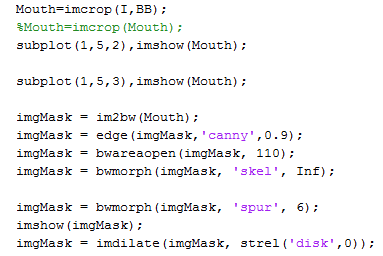


The edges for the mouth region were extracted by applying edge detection methods in conjunction with noise removal techniques. Experiments were carried out with both Canny and Sorbel edge detection functions but Canny edge detector gave the most prominent results and made the mouth features more visible.

**8.2 Remove Noise**

A close look at the images reveals the presence of noise. Any attempts to analyze the images were problematic. To alleviate this issue we employed several noise reduction techniques some of which are listed below.

* + 1. bwareopen
    2. structural elements
    3. Spur
    4. dilate

In use:

**8.3 Acquire Feature points**

To acquire the points and data that we need as inputs for our ANN, we employed several techniques for gathering data about the mouth region. Edge points, Coefficient of the line of best fit, and tortuosity were the three main properties we used.

**8.3.1 Edge Points**

We applied the “detectSURFFeatures()” function in Matlab to detect sign of Laplacian on the bounding box for the mouth region of the images. The ensuing images had points indicating the edges detected by the function and this formed the basis for curve fitting.

**8.3.2 Coefficients for the line of best fit**

A 4 degree polynomial was fitted to the image mentioned above. The resultant coefficients were inputs to the ANN.

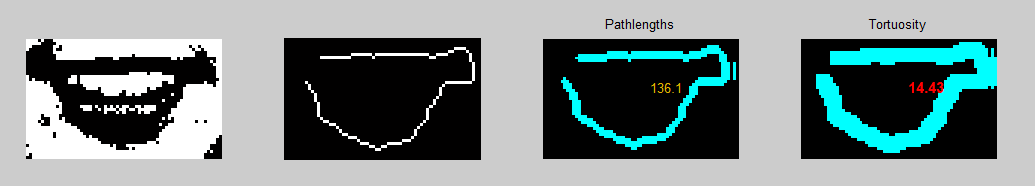
**8.3.3 Tortuosity**

We recognize that the human mouth exhibits different curve characteristics depending on the emotion a person is displaying. Smiling mouths display curvier lower lip regions while sad or angry lips are usually perched and therefore less curvy.



Knowing this we can determine if a lip is likely to be displaying a smile or anger by calculating the tortuosity of the lower lip region.

Happy/Smile



Angry



**8.4 Resultant Input ANN Dataset**

In the first stage of training ANN, we fed coefficients of the polynomial (degree 5) for the line of best fit and tortuosity into the (training) vector matrix. The input matrix was of this form:

*ANN\_mtrx = [c\_0 c\_1 c\_2 c\_3 c\_4 t\_const] (1x6 array),*

where each c\_{0..4} represents a coefficient from the polynomial of the fitted line between points, and *t\_const* stands for tortuosity of the disconnected lip segment.

Then, we store this information in a MAT file. We repeated this process for all available images (~32 per emotion) in our initial dataset. In order to train our ANN, we had to combine each MAT file, with derived heuristics, into a single dataset or comprehensive MAT file. Eventually, we had to obtain 2 MAT files: ANN\_HA.MAT and ANN\_SA.MAT, ie. dataset vectors for ‘Happy’ (32x6 array) and ‘Angry’ (32x6 array) emotions.

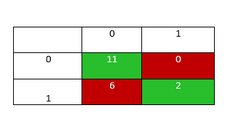
In the second stage of training, for each emotion, we have estimated respective ranges/intervals for tortuosity and coefficients (~95 confidence interval). In Excel, we used these intervals to generate mock datasets to train a more robust version of ANN which would better distinguish between happy and angry faces. Thus, we had to include additional output node (‘Other’) to existing two -- ‘Happy’ and ‘Angry’.

1. **Test Predictive Algorithm**

The input values for the ANN included polynomial coefficients and the tortuosity values for ‘Happy’ and ‘Angry’ images. The happy images had were identified with the status 1 and angry images had the status 0. A neural network was trained by feeding these values to a neural network implemented using R’s nnet package. The ANN was trained with 20 input nodes and set to a maximum of 10,000 iterations. A decay parameter (.001) was set to ensure that the model did not overtrain.

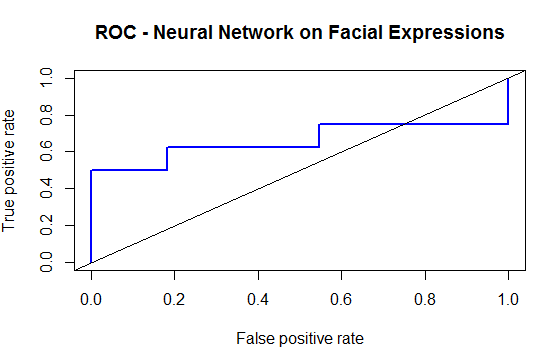
We validated the results using two tools - a confusion matrix and an ROC

The output of the classifier was checked using a confusion matrix. This matrix compares the predicted class to the actual class. The confusion matrix is created using the data set with real values from the image which was not used for training the model. The test data contained attribute values for 19 images - 8 Happy and 11 Angry.



The happy images are referred to as 1 and angry images as 0. As seen in the above confusion matrix Angry faces were predicted accurately but the prediction for the Happy faces is not as robust.

A ROC (Reciever Operating Characteristics) Curve was generated using R’s ROCR package. True positive rate is plotted against the false positive rate. The curve gives a visual illustration of how well the model is performing.



The accuracy depends on how well the test separates the group being tested into happy and angry. Accuracy is measured by the area under the ROC curve. Area under the curve for the model is 0.6590909. This indicates a fair performance. The performance of the model can be improved by adding other features and information points.

1. **Challenges**

Throughout the time of our project we encountered several challenges some of which we were able to tackle and others we had to find creative ways to tackle.

**Issues**

* **Coming up with a uniform Dataset**  
  One challenge we faced was coming up with a large dataset to be evaluated on our training data model.
* **Uncategorized image dataset**The images in our dataset were uncategorized meaning that all expressions were all in a single folder.
* **Variable thresholds**  
  Because of the diversity in human faces we discovered the different thresholds were require for facial feature identification.
* **Noise**  
  The images we had to work with had noise from the background and on the faces themselves.
* **Coming up with a MAT file**  
  This was one of the main challenges we had a hard time coming up with a mat file which will best describe our training input metrics (heuristics).
* **Our knowledge of Matlab was not robust (syntax)**

**Solutions**

* **Jaffe Database with images**  
  The Jaffe Database provided us with a fairly uniform set of face images with different expressions
* **Manually cluster the images**  
  Since the images were not categorized, we had to manually cluster the images into groups.
* **Try-catch statements for debugging purposes**  
  We implemented try catch statement as a way of trying various thresholds.
* **Use techniques from class lectures to remove noise**  
  The various techniques we learnt in class were helpful in us removing noise as much as possible
* **Using R’s neural network package**  
  A different tool like R that has a simplified ANN package helped us avoid the troubles of the more robust Matlab Neural network.

1. **Lessons Learned**Some of the Lessons we learned from our work on this project include the following:
   * Noise removal is an important aspect of image processing
   * Start with smaller number of images and work your way up
   * Each process during image processing is important. A poor input can result in poor results in later stages.
   * Facial features are hard to extract.
   * Find creative solutions to tackle ambiguity
   * Be open to using multiple tools
2. **Applications**

Facial expression recognition has many applications has the potential to improve human computer interaction and could be used in areas like -

* **Medical:** Monitor a patient’s face and report expressions to physicians.
* **Entertainment/Amusement parks:** Analyze customer experience before and after event
* **Shopping/Testing products:** Before and after experience
* **Security:** Monitor the expressions of suspects or persons of interest.

1. **Future Research**
   * Different Facial Features
   * Different genders
   * Different Races
   * Facial expression extraction from video
   * Additional predictive capabilities (decision trees etc.)
   * Calculate porosity
   * Other Metrics we can employ
   * Convex Hull
   * Histograms
   * Torsion/geodesic paths
2. **Conclusion**

During the span of our work on this project we were able to create a framework that pulled in image data, reduced noise in mouth region, got the tortuosity and coefficient of degree 4 in the Lip Region, trained our neural network and able to predict fairly what expression the person in the image had.

1. **References**

<http://cs.au.dk/~jtp/SURF/report.pdf>

http://alwaysinfo.co.uk/images/i/face/5

http://www.kasrl.org/jaffe.html

http://scg.sdsu.edu/ann\_r/

http://arxiv.org/ftp/arxiv/papers/1204/1204.2073.pdf