Facial Expression Classification using Visual Cues and Language

Abhishek Kar
Advisor: Dr. Amitabha Mukerjee
{akar,amit}@iitk.ac.in
Department of Computer Science and Engineering, IIT Kanpur

Abstract

In this work, we attempt to tackle the problem of correlating language with facial expressions to learn in an unsupervised manner, the adjectives used to describe emotions. The problem is divided into two parts - i) a supervised method for classifying facial emotions using visual cues and ii) an unsupervised algorithm to extract keywords out of commentary on videos depicting facial expressions. We use a method based on Gabor filters and Support Vector Machines to classify emotions into 7 categories - Anger, Surprise, Sadness, Happiness, Disgust, Fear and Neutral. We explore various dimensionality reduction and feature selection methods like PCA and AdaBoost. The Extended Cohn-Kanade database [7] is used for testing the algorithms. We achieve an accuracy of 94.72% for a 7-way forced choice SVM classifier after feature selection using Adaboost which is a significant improvement over many previous successful approach based on PCA and LDA and Local Binary Patterns. In the next step, we obtain commentary on 40 videos depicting 4 emotions - Anger, Sadness, Happiness and Surprise and cluster keywords obtained using a maximum co-occurrence method to discover descriptors for these emotions.

1 Introduction

Facial expression analysis has been a long standing problem in computer vision with applications in Human Computer Interaction, video summarization, effective indexing of videos and finding lower dimensional embedding for facial actions. Consider the task of summarizing a video by the facial expressions of the subject (see Figure 1). Facial expression classification would enable us not only to achieve the above task but also efficiently answer queries such as 'Frames where Subject A is smiling'. Facial expression categorization systems are also used in personal robots like Sony's Aibo, ATR's RoboVie and CU animator. A major part of facial expression classification involves defining a robust vocabulary for facial actions. The Facial Action Coding System(FACS)[4] has been the state of the art in manual coding of facial expressions and have been widely used to train supervised classifiers to recognize emotions in humans. In this work we try to explore the use of gabor filters for feature extraction and subsequent classification by SVMs to classify images into 7 basic emotion categories - Neutral, Anger, Disgust, Fear, Happy, Sadness and Surprise. We investigate various dimensionality reduction and feature selection methods like PCA and AdaBoost to efficiently represent the data. We further extend our work to discovering keywords describing four different emotions - Anger, Sadness, Happiness and Surprise in an unsupervised manner from video commentary data.

2 Previous Work

There has been substantial effort devoted to automatic facial image analysis over the past decade. The pioneering work of Black and Yacoob [3] recognizes facial expressions by fitting local parametric motion models to regions of the face and then feeding the resulting parameters to a nearest neighbor classifier for expression recognition. Hoey [5] presented a multilevel Bayesian network to learn in a weakly supervised manner the dynamics of facial expression. De la Torre et al. [2] proposed a geometric-invariant clustering algorithm to decompose a stream of one person's facial behavior into facial gestures. Shan et al. [8] explore the use of local binary patterns (LBP) for the given task. Bartlett et al. [1] proposed a method based on gabor filters and AdaSVMs for the same purpose. Their approach has proved to be one of the

Figure 1: Video summarization using facial expression clustering

most successful ones and it has been adapted into a commercial facial expression categorization system called the Computer Expression Recognition Toolbox [6].

Our work is mainly inspired by the work of Bartlett et al. as it provides a simple and highly configurable method with a lot of opportunity for experimentation. Their reported accuracy is around 90% and it stands amongst the top methods for expression classification. Moreover it is possible to build a realtime system for emotion recognition using this approach. Correlation of language with facial expressions using computational methods has not been explored till date to the best of our knowledge.

3 Methodology

3.1 Face Detection

We use the Viola-Jones [9] face detector to find the face in the image. It is the state of the art face detection algorithm in use and performs exceptionally on a number of dataset. The detector is based on haar cascade classifiers. Each classifier uses rectangular haar features to classify the region of the image as a positive or negative match.

$$f_i = \begin{cases} +1 & v_i \geq t_i \\ -1 & v_i < t_i \end{cases} \tag{1}$$

The haar features used are shown in Figure 2.
The detector uses AdaBoost and constructs a rejection cascade of nodes. Each node in turn is a multitree AdaBoosted classifier with a low rejection rate so that a few number of regions are classified out at each stage. A rejection at any node terminates the whole cascade. An initial learning phase is used to learn the thresholds t_i. Boosting is then used to calculate the weights w_i. Finally the function below is used to classify whether the region is in an object of interest or not. Thus effectively a region is classified as a positive if it makes it through the whole cascade.

$$F = sign(w_1 f_1 + w_2 f_2 + ... + w_n f_n) \tag{2}$$

In this work we detect images in the dataset images using the OpenCV Viola Jones detector. The face area is cropped out and resized to 48x48 for subsequent processing.

3.2 Gabor filters

Gabor filters, in the spatial domain are gaussian kernels modulated by sinusoidal waves. They are mainly used in fingerprint recognition where they are used to enhance the features. Gabor representations of

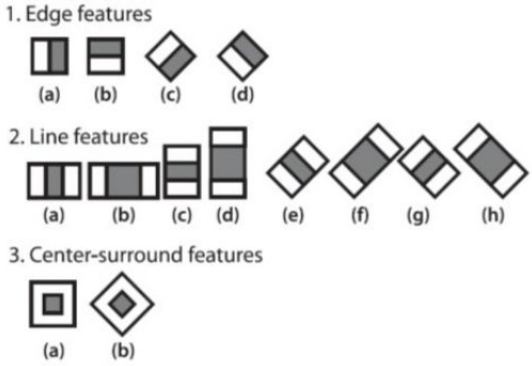

Figure 2: Haar-Features used in the Viola-Jones algorithm [9]

Figure 3: Face detection on a CK+ image

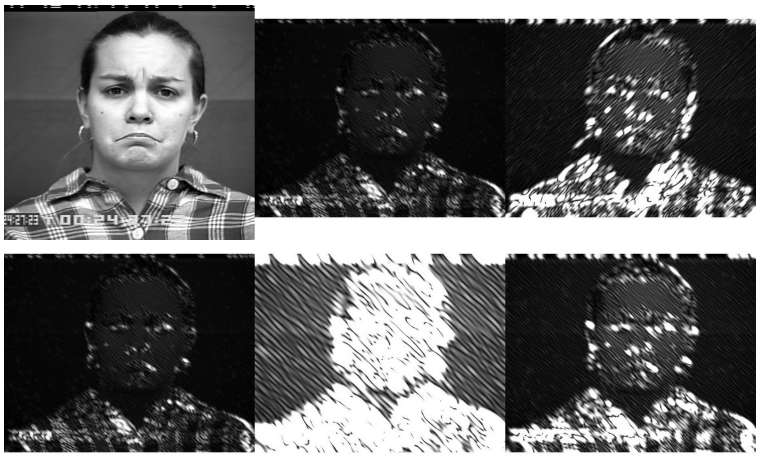

Figure 4: Gabor Filters applied on a CK+ image

images are generally complex and the magnitude is generally used for visualization. The convolution of Gabor filters is done by computing the Fourier transform of the image and multiplying with the Fourier transform of the Gabor filter in the frequency domain. This aids in reducing computation time.
In this work, we use a bank of 72 gabor filters (8 orientations and 9 spatial frequencies 2:32 pixels per cycle in half octave steps). The 48x48 face patch is convolved with all the 72 gabor filters to obtain a feature of size 48x48x72 = 165888 per image. Frequency and orientation representations of Gabor filters are similar to those of the human visual system, and they have been found to be particularly appropriate for texture representation and discrimination. In this problem, we need to find a feature set that appropriately models the orientation information of various facial units like the lips, eyebrows, eyes etc. The bank of local gabor filters used in our approach succeeds in achieving this with an added advantage of frequency information.

3.3 Dimensionality Reduction/Feature Selection

3.3.1 Adaptive Boosting

The AdaBoost algorithm iteratively combines many weak classifiers to approximate the Bayes classifier. Starting with the unweighted training sample, the AdaBoost builds a classifier, for example a decision stump or a classification tree that produces class labels. If a training data point is misclassified, the influence(weight) of that training data point is adjusted(boosted). In next iteration, a second classifier is built using the new weights, which are no longer equal. Again, misclassified training data have their

weights adjusted and the procedure is repeated. A score is assigned to each weak classifier, and the final classifier is defined as the linear combination of the classifiers from each stage.

In our problem, we select the best features (best weak learners) obtained by applying adaboost for every class using one-vs-rest strategy for every class. We do this by binning the best weak-learners obtained in all the iterations for every one versus rest sub-problem and picking the top 'k' features. The final feature set is taken to be the union of best features obtained for every sub-problem.

3.3.2 Principal Component Analysis

Principal Component analysis of data is a dimensionality reduction technique wherein the data dimension is reduced by mapping it into its eigen-space. In the process the top-k eigenvectors are chosen to reflect the directions of maximum variability of the data. In practice, the lower dimensional representation is calculated by a singular value decomposition of the data followed by extraction of its top-k eigenvectors (where k is chosen by the amount of energy to be retained) and mapping to the top-k eigenvector space.

In this problem, we tried various values of k ranging from 10 to 359 and optimized for the maximum classification accuracy. The final best dimensionality was found to be 60. It is interesting to note that the Facial Action Coding System[4] has 64 action units that are used to code various emotions. We can perhaps conclude that the intrinsic dimensionality of the data is 60 and PCA succeeds in mapping the data to this dimension.

3.4 Classification

3.4.1 Support Vector Machines

Support Vector Machines (SVMs) are primarily binary classifiers that find the best separating hyper-plane by maximizing the margins from the support vectors. Support vectors are defined as the data points that lie closest to the decision boundary. SVMs can be extended to multiclass problems by two

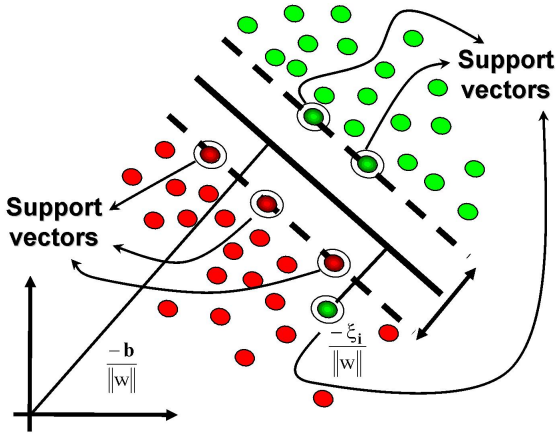

Figure 5: Classification using SVM

methods - One versus All and One vs One classifiers. In the latter method, $\binom{n}{2}$ SVMs are trained for each combination of classes and the final class label is decided using majority voting. A similar approach can also be used for the first case with all n classifiers.

We use both methods for classification. We use a novel way of choosing the final class in the One versus All case. If a feature vector is classified in multiple classes, we choose the class with the maximum margin from the decision boundary. If a feature vector is classified no single class, we take the class that has the minimum margin from the decision boundary. This results in a significant improvement over the

One vs One case. We also use stacked 1 vs rest and 1 vs 1 classifiers and a novel way of combining the Adaboost features for use with SVMs and present the results in the following section.

3.4.2 Bayesian Classification

We use baseline classifier based on data reduction and classification using a Bayesian classifier. We fit multivariate normal distributions to all the classes using Maximum Likelihood Estimation and classify based on posteriors.

4 Dataset and Results

The primary database used for testing the results of the algorithms was the Cohn-Kanade+ database. It consists of 593 posed sequences from 123 subjects. All images are of dimensions 640x490 and have annotated tracked AAM feature points. Each sequence starts with a neutral expression and terminates with the peak expression. All the peak expressions in the database are FACS coded and 327 of the 593 sequences are emotion labeled. There are 8 expressions present in the database: Angry, Disgust, Fear, Happy, Sadness, Surprise, Contempt, Neutral. We ignored the contempt emotion as it has been to be confusing in literature and ends up hampering the accuracy. We took an average of 50 neutral expressions in the database to keep the number of samples of all emotions roughly equal.

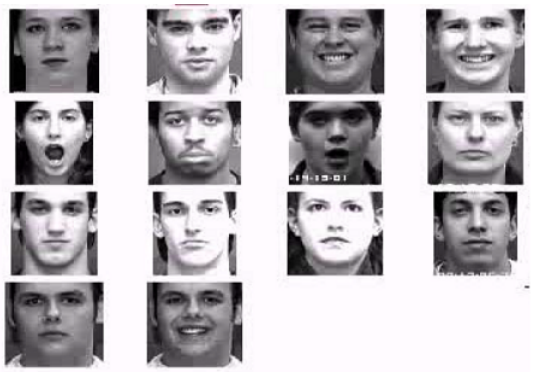

Figure 6: Images from the CK+ dataset

Emotion	Frequency
Neutral	50
Angry	45
Disgust	59
Fear	25
Happy	69
Sadness	28
Surprise	83

Figure 7: Frequency of emotions in CK+ dataset

We detected faces in the CK+ dataset and resizing them. This was done in OpenCV because of its speed and the inbuilt Viola Jones face detector. The resized images were then read in Matlab and only the peak expressions of image sequences containing emotion labels were used to compute the gabor magnitude representations. Finally, a feature set of 359x165888 elements was obtained. We reduced it to 60 dimensions using PCA. A linear SVM with a was trained on the PCA reduced data. We used both one-versus-one and one-versus-all SVMs from the libsvm library and MATLAB. We also had to modify the MATLAB SVM implementation to obtain the margin for the one-versus-one case. The results are compiled in table 1. All reported accuracies are obtained using 10 fold crossvalidation. The discrepancy between accuracies of 1 vs 1 and 1 vs rest may be attributed to the better technique used to decide the final class in the latter.

In the Adaboost approach, we found the best features corresponding to every one vs. all combination (total 7) and took the union of all the features to form the dimension reduced feature vector. In our implementation, we used 300 iterations of Adaboost to find the top features and obtained a final set of 175 features after union. These were further classified using SVMs and the baseline Bayesian approach. In another approach that we call *Adaboost-SVM*, we used only those features to train and test the SVM for class x that were found to be the best by the Adaboost iteration for class x vs. rest. We also used stacked classifiers wherein if a test image was classified into more than one classes in the 1 vs. rest approach, we used a 1 vs. 1 SVM on top of it to disambiguate between the classes. We present the results for all these approaches in table 1. It is worth noting that the *Adaboost-SVM* method gives almost the same accuracy as SVM followed by feature selection using SVM. The major difference is that while the latter takes the union of all features and reduces the feature set to 175, the former uses

Classifier Features	SVM (1 vs 1)	SVM (1 vs rest)	Adaboost-SVM	Stacked SVM	Bayes
PCA	71.08%	72.19%	–	69.66%	80.45%
None	75.39%	88.87%	–	–	–
Adaboost	80.43%	**94.72**%	94.61%	92.91%	86.64%

Table 1: Accuracy of classifiers using 10 fold cross validation

Emotion	None	Adaboost
Neutral	97.5%	98.05%
Angry	91.65%	95.26%
Disgust	98.04%	99.72%
Fear	96.1%	98.04%
Happy	98.6%	98.89%
Sadness	94.16%	94.99%
Surprise	97.78%	99.17%

Table 2: 10 fold cross-validation accuracies of single emotions with and without feature selection

different set of features for training different SVMs and uses feature vector sizes between 20-30. Thus the Adaboost-SVM method might be preferred over the traditional method if speed is a requirement.

5 Correlation with Language

As the next step to our work, we obtained commentary on 40 videos made from image sequences of people depicting various emotions. The methodology used for obtaining the commentaries was the following: Subjects were shown a demo video of a posed expression. They were asked to comment on the next video. No specific directions were provided as to what to comment on in the video. 60 such responses were recorded in English and subsequently transcribed. The 40 videos on which commentary was obtained was taken from the test set from the previous step and the labels assigned to them were labels predicted by our algorithm and not the truth labels. Extraneous words like articles and prepositions were stripped from the transcribed responses and just the keywords were retained. The keywords had derivatives like -ed, -es, -ing etc. Thus. Levenshtein edit distance was used to match keywords. The co-occurrence values in the same emotion for different pairs of keywords was obtained and they were grouped by emotion. Table 4 shows some keywords discovered for the four emotion categories.

6 Conclusions

In this work, we have developed a facial expression recognition algorithm that classifies into 7 emotion categories. Our method performs better than a number of previous approaches and is a significant improvement over the classic approach of using PCA + LDA. Our method uses Adaboost which is a very

Approaches	Best Accuracy
Our method	**94.72**%
Gabor filter + AdaSVM[1]	93.3%
Boosted LBP + SVM (linear)[8]	91.1%
Gabor filter + SVM[6]	90.1%
PCA + LDA[1]	80.7%

Table 3: Accuracies of various methods on the CK+ dataset

Emotion	Keywords
Happiness	Happy, Smile, Delight, Joy
Sadness	Distress, Unhappy, Sad, Gloomy, Sleepy, Grief, Sorrow
Anger	Anger, Curious, Frown, Furious, Irritate, Ill temper
Surprise	Amaze, Surprise, Shock, Astonishment, Stupefy, Awe, Bewilderment

Table 4: Keywords discovered for various emotion categories

slow algorithm and takes a long time to converge. Our feature set is also very large (165888 features per image) and thus required considerable computational power. On the other hand, this method a very good recognition accuracy on the CK+ dataset. We have also developed a method to discover adjectives describing basic emotion categories in an unsupervised manner. We can hypothesize that this is how we learn to interpret various emotions. We are exposed to many expressions as we grow up and we pick up words describing each category and form associations. This gives more importance to the visual system in the task of expression recognition. We may also hypothesize that it is language that helps us to form subcategories in each emotion category depicting various levels of the same emotion. Intensities associated with different adjectives can lead us to associate intensities with different depictions of the same basic emotion.
We would like to extend this work for more robust recognition keywords from the commentaries. In the current work, almost all the test images were classified correctly and the subjects recognized the emotions correctly. Thus a simple maximum co-occurence method gave sufficiently good results. In case of misinterpretation of emotions by subjects, we would like to use a graph min-cut based method to clusters the keywords.

References

[1] Marian Stewart Bartlett, Gwen Littlewort, Mark Frank, Claudia Lainscsek, Ian Fasel, and Javier Movellan. Recognizing facial expression: Machine learning and application to spontaneous behavior. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:568–573, 2005.

[2] F. De la Torre, J. Campoy, Z. Ambadar, and J.F. Conn. Temporal segmentation of facial behavior. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1 –8, 2007.

[3] F. De la Torre, Y. Yacoob, and L. Davis. A probabilistic framework for rigid and non-rigid appearance based tracking and recognition. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 491 –498, 2000.

[4] P. Ekman and W. V. Friesen. *Facial Action Coding System*. Consulting Psychologists Press, Stanford University, Palo Alto, 1977.

[5] J. Hoey. Hierarchical unsupervised learning of facial expression categories. In *Detection and Recognition of Events in Video, 2001. Proceedings. IEEE Workshop on*, pages 99 –106, 2001.

[6] G. Littlewort, J. Whitehill, Tingfan Wu, I. Fasel, M. Frank, J. Movellan, and M. Bartlett. The computer expression recognition toolbox (cert). In *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 298 –305, march 2011.

[7] P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94 –101, june 2010.

[8] Caifeng Shan, Shaogang Gong, and Peter W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803 – 816, 2009.

[9] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:511, 2001.