Reading Faces with Conditional Random Fields

Javier Hernandez Rivera  
javierhe@andrew.cmu.edu

Tomas Simon Kreuz  
tsimon@andrew.cmu.edu

Abstract

The purpose of this work is to explore the use of Conditional Random Fields (CRFs) in the automatic recognition of the basic units of facial expression, described by the Facial Action Coding System (FACS) as Action Units (AU). Facial expression recognition is a complex problem aggravated by (1) variability between the subjects and (2) variability in the dynamics of expression. The discriminative nature of CRFs and their ability to capture temporal dependencies makes them well suited to tackle these two difficulties. In this work, CRFs are compared with SVMs, which are considered state of the art for expression recognition. To evaluate the contribution of each of the purported theoretical benefits of CRFs, performance is also compared with logistic regression (their static analogue), and HMMs (their generative pair). To recognize subtle expressions, a combination of shape and appearance features is used. Results are evaluated on a subset of the Ru-FACS database, a large video corpus of diverse subjects undergoing a variety of spontaneous expressions.

1 Introduction

The human face is highly expressive, with the potential to reveal (and conceal) emotions, intentions, and other mental states. Playing a key role in nonverbal communication, facial expression provides a communication channel with a language of its own. The purpose of this work is to explore the use of Conditional Random Fields [10] in the automatic recognition of the basic units of this language. These units will be assumed to be those proposed in the Facial Action Coding System (FACS) [7], which decomposes facial expression into a set of basic discernible movements (Action Units or AU), such as AU12, the lip-corner pulling motion in smiles.

Most approaches to automatic facial expression recognition simplify the problem into one of binary classification, where presence or absence of a certain expression or AU is evaluated on a frame-by-frame basis. Usually, a classifier (eg. Support Vector Machines and boosting methods) is trained on frames where a certain facial expression was present, using as the negative set a sampling of frames where it was not present. A problem with this approach is that it fails to consider the temporal dynamics inherent to the execution of facial expressions.

From a theoretical point of view, it would be more attractive to use a model which includes temporal dynamics in its formulation. Hidden Markov models (HMM) are one such option, but the large negative class makes the use of discriminative models such as conditional random fields more appropriate. Assessing the practical use of CRFs in this context will be the object of this study.

2 Related Work

There has been a substantial body of research in automatic facial expression analysis; see [8] for a survey. Many systems consider static evaluation on each frame, and are differentiated by the features and classifiers used. In [12], an SVM classifier is trained on AAM-derived shape-normalized features and appearance features combined using Principal Component Analysis (PCA). [2] obtained
best recognition performance using an SVM when using a set of Gabor wavelet coefficients selected by AdaBoost.

Temporal information is often included in a probabilistic framework. In [17], a Dynamic Bayesian Network captures the co-occurrence statistics of different AU as well as the dynamic succession of events. Hidden Markov models are used in [19]. The most relevant work is [9], where CRFs are used to discriminate between the six universal facial expressions (ie. happy, sad, etc.) in a posed expression database. Although high performance has been achieved on posed facial action data, only a small number of studies are conducted on spontaneous data [2, 12], which are more representative of real world situations including confounds such as non-frontal head orientation, motion and speech.

3 Methods

This section explains the algorithms and methods used for the recognition of different AUs. It is divided in two parts: Section 3.1 describes the features chosen, and Section 3.2 explores the theory behind CRFs and their applicability to this problem.

3.1 Feature extraction

Our data-set consists of video sequences that were tracked using a person-specific AAM tracker which yields, on a frame by frame basis, the position of several points of interest on the face of the subject. It is from this data that we will extract features.

The features for this classification task must satisfy two criteria. On one hand, they must capture sufficient information to discriminate the changes that characterize facial expression, and on the other, they must prove somewhat robust to confounding factors such as head pose, inter-subject variability (eg. skin tone, facial structure), and other irrelevant peculiarities such as illumination changes [2].

To achieve this, two complementary types of features were extracted: shape features and appearance features. Previous work has shown that using a combination of shape and appearance can yield improved classification performance [12]. Shape features are obtained from the AAM tracker and are meant to be robust to many of the aspects that affect pixel intensity and color (eg. illumination changes, skin tone). Appearance features can provide robustness to inaccuracies in the tracked points and, if the facial region is adequately registered, to certain changes in pose.

3.1.1 Shape features

Shape features are extracted from the tracked AAM mesh points or landmarks. There are 66 of these landmarks, distributed along interesting regions on the face (see Fig. 1), in particular, along the top of the eyebrows, the inner and outer lip outlines, the outline of the eye, the jaw, and rigid points along the nose.

---

1 The tracker is due to Iain Matthews & Simon Baker [13]. Landmark labeling and tracking was done by the Affect Analysis Group at the University of Pittsburgh.
As in [12], previous to feature extraction, a shape normalization step is carried out. The purpose of this normalization is to transform the shape points in each frame so as to remove as much rigid motion (translation, rotation) and scale changes as possible, while maintaining the non-rigid deformations (eg. the shape of the mouth, the relative position of the eyebrows) intact.

The normalization step consists of an iterative Procrustes alignment [5]. In this process, all shapes in the training set are transformed by a similarity transform to warp them to a common reference frame given by the mean or centroid of all shapes. This mean shape is recalculated as the iterations progress, yielding the shape that is closest fit to the full set of shapes. The result of this process can be seen in the left of Fig. 1.

Once the faces have been aligned and normalized, the features extracted from these point clouds are distances between a number of points in the region of interest for a particular AU (eg. outer mouth corners, eyebrow to rigid points, between eyebrows), angles (eg. angle in the outer mouth corners), and coordinates for some of the landmarks.

3.1.2 Appearance features

Due to their success in the object detection community, the appearance features were chosen to be based on the histogram of oriented gradients [6]. Intuitively, being gradient based, these features have the potential to capture much of the information that is described in the FACS manual [7] for the recognition of AUs (eg. the markedness of the naso-labial furrows, the direction and distribution of wrinkles, the slope of the eyebrows).

Feature extraction is similar to the well-known SIFT descriptor [11], with a few key differences. Instead of identifying scale-invariant and salient points, points of interest are given by the location of some of the landmarks of interest (eg. outer mouth corners, eyebrow corners, nose corners). The size of the texture window is fixed to a proportion of the scale of the face relative to the mean of all faces (determined in the shape normalization step) to account for changes in scale and size of the subject’s face. The image is normalized using standard histogram equalization to lessen the effect of illumination and skin tone, and the gradient of the grayscale image is then calculated in the texture window. This window is broken up into four regions. An 8-bin weighted histogram of oriented gradients is constructed in each region. These values are concatenated to become the appearance descriptor.

3.1.3 Dimensionality Reduction

The combination of shape and appearance features yields a large feature vector with 294 features (for lower face AUs) or 196 features (for upper face AU). Given the size of the data-set and the computation time required for training the classifiers, we chose to reduce the dimensionality of these feature vectors. For this purpose, Principal Component Analysis was applied, retaining 95% of the energy. These projected feature vectors are the input to the classifier.

3.2 Linear-Chain CRFs

Conditional random fields are state-of-the-art undirected graphical models commonly used for segmentation and sequential labeling tasks [10]. Nodes in the graph represent labels (unobserved states) and observations, while edges capture probabilistic dependencies between them. The graph models the conditional distribution of the data, which is sufficient for classification. Most suited to the sequence labeling task are linear-chain CRFs [16], and we will restrict our discussion to this special case structure illustrated in Figure 2. The sequence of blue nodes represent the hidden states or labels (indicating presence or absence of a particular AU in each frame), while green squares represent observations (shape and appearance features). The conditional probability given by this structure is:

\[
p(y | X) = \frac{\exp \left( \sum_k \left( \sum_i \alpha_i s_i(X, y_k, k) + \sum_j \mu_j t_j(X, y_k, y_{k-1}, k) \right) \right)}{Z(X)}
\]

Notation: Bold lower case x denote column vectors, bold upper case X denote matrices. Lower case letters y denote scalars.
Two types of cliques are present in this graph [18], corresponding to edges between labels, and edges between a label and an observation. In Eq. (1), these two types of potential functions are created by \( s_i(x, y_k, k) \) and \( t_j(x, y_k, y_{k-1}, k) \), where \( k \) iterates over all frames, and \( i \) and \( j \) iterate over all features. Conceptually, \( s_i \) are state functions which help determine the identity of the label (relating state and the observations), while \( t_j \) are transition functions, which determine the associations between consecutive states. \( \alpha_i \) and \( \mu_j \) are the weights corresponding to each of the feature functions, and \( Z(X) \) is the normalizing factor or partition function which ensures that the probability adds to 1.

Inference in CRFs can be performed efficiently and exactly by variants of the standard dynamic-programming algorithms (i.e. forward, backward) used for HMMs [16]. Parameter estimation is performed by maximizing the conditional log likelihood (Eq. (2)) with a regularization factor (e.g. L2 norm) which avoids over-fitting. The convexity of the function ensures convergence to a global optimum, and a commonly used method is the Scaled Conjugate Gradient [10, 16]. As a basis for our implementation we used the Conditional Random Field Toolbox for MATLAB developed by Kevin Murphy [14].

3.3 Advantages and applicability of CRFs

One of the main advantages of CRFs is its ability to exploit contextual information by considering temporal dependencies in the data. To analyze the possible benefits of this capability, we establish a comparison with logistic regression. This well-established discriminative classification algorithm is commonly used in practice because of its ease of use and robustness. As with CRFs, the conditional likelihood is maximized, according to the following expression:

\[
\ell(\theta) = \sum_k \left( \sum_i \alpha_i s_i(x, y_k, k) + \sum_j \mu_j t_j(x, y_k, y_{k-1}, k) \right) - \log Z(X) - \sum_{k=1}^{K} \frac{\lambda_k^2}{2\sigma^2}
\]

This formula shows strong similarities with CRFs. The main difference is that logistic regression does not encode contextual information or consider temporal dependencies and, therefore, treats each sample independently of the others. Figure 3 shows the output confidence of logistic regression and CRFs for the video sequence of one subject. When using logistic regression, confidence values vary from one frame to the next, while CRF confidences are smoother over time. This temporal smoothness is a very desirable property for this particular application, ensuring temporal consistency in the final predictions.

Another commonly cited advantage of CRFs is their discriminative nature. The relevant comparison here is with hidden Markov models, the generative pair of CRFs. HMMs have been widely used to statistically model time-varying processes. The graphical model for an HMM is the same as for linear-chain CRFs (Fig. 2) but with directed edges. These directed edges reflect the underlying Markov assumptions and result in the following expression for the joint probability of the data:

\[
p(y, X) = p(y_1) \prod_{k=2}^{n} p(x_k | y_k)p(y_k | y_{k-1})
\]

The Markov assumptions impose limits on the features used as observations (they should depend only on the current state) and on the features used to model transition probabilities. Additionally, the
Figure 3: Prediction confidence $p(y = AU12 | X)$ for logistic regression (top) and CRFs (bottom). The color green denotes presence of AU12 in the ground truth labeling, while red denotes absence of this AU. Time (frame number) is represented along the x-axis. The graphs on the right show close-ups of a particular frame range.

Figure 4: PCA projection of the features for two sequences. Red points correspond to frames where no AU12 was observed and green points correspond to frames where AU12 was present.

The generative nature of HMMs requires a model of $p(x | y)$, which is often difficult due to feature interdependencies. By comparison, discriminative approaches such as CRFs model only the decision boundary and do not require a generative distribution over $x$, which is not needed for classification. This decision boundary between classes is often easier to learn than the underlying distribution of the data.

Figure 4 shows the projection of the data into the first two principal components for sequences corresponding to two subjects. The frames of each of the subjects appear projected onto semi-circles, where intense smiles appear at the bottom-right tips, and subtler smiles are clustered at the origin (where neutral expressions dominate and a large amount of overlap between the classes
exists.

Ideally, these hard-to-discriminate frames that are close to neutral would be disambiguated by the temporal component in CRFs and HMMs. For this example, it is easy to imagine how finding a boundary to split the classes might be simpler than fitting a distribution to such complex data, especially when considering the high-dimensionality of the complete feature space.

This two dimensional embedding also illustrates some of the difficulties of the classification problem. Subject variability creates clusters (the separate arcs or branches for each subject in Fig. 4), when we would ideally want our features to reflect smile intensity independently of the subject. Variability due to pose and lighting can be seen in the smaller red cluster at the top of the graph, where the subject was looking downwards in that particular group of frames.

4 Experiments

Evaluations of performance were carried out on a relatively large corpus of FACS coded video, the RU-FACS-1 [[13]] data-set. Recorded at Rutgers University, subjects were asked to either lie or tell the truth under a false opinion paradigm in interviews conducted by police and FBI members who posed around 13 questions to the subjects. These interviews resulted in 2.5 minute long continuous 30-fps video sequences containing spontaneous AUs of people of varying ethnicity and sex. Ground truth FACS coding was provided by expert coders. Data from 28 of the subjects was available for our experiments. In particular, we divided this dataset into 17 subjects for training and 11 subjects for testing.

As a performance measurement, we chose Receiver Operating Characteristic (ROC) curves and the area under their curves (AUC). This method shows the relative tradeoffs between false positive rate (1 - specificity) and true positive rate (sensitivity) for different thresholds on the output confidence of binary classifiers. This allows intuitively measuring and comparing performance between different classifiers. It is worth to mention that, since we have a larger set of negative samples (frames without AU) than positive samples (frames with AU), having False Negatives (FN) penalizes much more than having False Positives (FP). In other words, the effect of having FP is attenuated by the large amount of negative samples while the effect of having FN is magnified by the small set of positive samples.

The purpose of these experiments is to answer two main questions:

1. Can CRFs be successfully applied to this problem? To answer this, we compare performance of CRFs with SVMs, which is considered a state of the art method for expression recognition.

2. Do the theoretical advantages of CRFs prove useful in practice for this problem? The ability to capture temporal dependencies is evaluated by comparing performance to a logistic regression (the static analogue of CRFs), and the advantages of using a discriminative model are evaluated by comparing performance to HMMs (its generative pair).

In particular, we use LibSVM, a publicly available SVM implementation [[4]]. Matlab’s native implementation is used for logistic regression, and the Bayes Net Toolbox [[15]] is used for inference in HMMs. Learning for HMMs is done in a fully supervised manner; transition probabilities are estimated directly from labeled data and emission probabilities are modeled as Gaussian mixtures with a full covariance matrix on the feature data for each frame. All hyper-parameters for these algorithms, such as the regularization factor for CRFs, kernel type (e.g. Linear, RBF and Polynomial) for SVMs, and the number of Gaussian mixtures for HMMs, are optimized by performing 5-fold cross validation on the training set.

Figure 5 shows ROC curves for all the algorithms on several AUs. As can be seen, the performance of each algorithm varies for different AUs.[[4]] We can conclude that CRFs perform comparably to algorithms considered state of the art for this application, such as SVMs, and that they can therefore successfully be applied to this problem. On the other hand, given that for many of the AUs performance is not substantially better, it might be preferable to use the simpler classifiers. It is worth to

---

3It is also interesting to note how non-smiling expressions due to speech become increasingly similar to intense and wide smiles as the mouth becomes increasingly open (bottom-left tip of the semi-circles).

4The authors like to think that this agrees with the “no free lunch” theorem—given many different classification problems, no one classifier can be optimal in all cases.
mention that for the majority of the SVM classifiers, Radial Basis Function kernels obtained better performance than linear or polynomial kernels, which suggests that the kernel trick has been fundamental for the classification performance of SVMs.

As for our second question, we can surmise that the theoretical advantages of CRFs are exploited for some AUs, such as 6+12 (combination of AU6 and AU12) or 15+17. Nevertheless, for some cases (most of them upper face AUs such as AU1+2), the generative model of HMMs seems to outperform CRFs. This suggests that there is a strong overlap between positive and negative samples, for which it may be more difficult to determine a boundary rather than approximate a distribution. An additional consideration is that smoothing the output predictions may not always be helpful for the classification problem. The reasoning behind this is that there is a large amount of expression variability, resulting in no usable temporal consistency across subjects.

Figure 5: ROC curve results for a number of AUs.

5 Conclusions

We evaluated and studied the use of conditional random fields in the context of expression recognition, and established direct comparisons to other classifier algorithms that share their theoretical foundations (HMMs and logistic regression), obtaining mixed but comparable performance across several AUs of interest.

Spontaneous expressions in a realistic setting, such as those found in the Ru-FACS database, present serious difficulties from the standpoint of classification: variability between individuals (e.g. different skin colors, facial features), occlusion (e.g. glasses, hands on the face), and variability in pose and scale (e.g. constant movement of the head). Additionally, a major difficulty of this problem is the great variability in the execution of expressions across different people. For example, some people might sustain a smile for a long period of time, while others might exhibit micro-expressions lasting less than a second.
We believe the features to be a fundamental component to overcoming these problems. For instance, shape normalization removes a large amount of subject and pose variability. Nonetheless, this procedure introduces errors of its own, such as distortion due to errors in the affine transformations. A key issue is integrating information across time, which will require appropriate features as well as a suitable classifier.

Although linear-chain CRFs were not found to present a clear improvement over other algorithms, their discriminative nature and ability to capture temporal dependencies seem to adapt better with lower face AUs (e.g. smile). However both characteristics are not always beneficial, and special-case considerations are necessary for each AU.

In conclusion, CRFs provide a beautiful statistical framework that combines very desirable properties for temporal classification problems. Although the results are at the level of the state of the art, the strengths of CRFs have been constrained in part by the inherent difficulties of the data and the problem. Several improvements may enhance the capabilities of CRFs in this problem. An interesting extension to our CRF system would be the use of kernels to make the problem more linearly separable. Future research directions include learning the temporal patterns for each class, perhaps using extensions which consider an additional hidden state layer, such as Hidden-CRF and Latent-Dynamic-CRF.

References

[1] Rufacs1 dataset: http://mplab.ucsd.edu/?page_id=80