A Novel LDA and HMM-based technique for Emotion Recognition from Facial Expressions

Akhil Bansal, Santanu Chaudhary, Sumantra Dutta Roy

Indian Institute of Technology, Delhi, India

akhil.engg86@gmail.com, santanuc@ee.iitd.ac.in, sumantra@ee.iitd.ac.in

Abstract. Automatic human emotion recognition from facial expressions has grabbed the attention of computer scientists for long, as it is necessary to understand the underlying emotions for effective communication. Over the last decade, many researchers have done a lot of work on emotion recognition from facial expressions using the techniques of image processing and computer vision. In this paper we explore the application of Latent Dirichlet Allocation (LDA), a technique conventionally used in Natural text processing, used with Hidden Markov Model (HMM), for learning the dynamics of facial expression for different emotions, for the classification task. The technique proposed works with facial image sequence. Each frame of an image sequence is represented by a feature vector, which is mapped to one of the words from the dictionary generated using k-means. Latent Dirichlet Allocation then models each image sequence as a set of topics, where each topic is in turn probability distribution over words. LDA doesn't take into account the order of words which appear in an image sequence to find topics. However we have the information of the order in which the words and hence the topics appear in an image sequence. We leverage the information about the order of topics for classification in the next step. This is done using a separate Hidden Markov Model for each emotion, each of which is trained to recognize the dynamics of facial expressions for the emotion. The emotions dealt with are six basic emotions: happy, fear, sad, surprise, angry, disgust and contempt. We further compare our results with the technique in which sequence of words instead of topics is used by HMM for learning facial expression dynamics. The results have been presented on Extended Cohn-Kanade database. The accuracy obtained on the proposed technique is 80.77% .The use of word sequence in general performs better than use of topic sequence for learning facial expression dynamics.

Keywords: Emotion Recognition, Bag of Words (BoW), K-means, Latent Dirichlet Allocation (LDA), Hidden Markov Models(HMM), Topic Modeling

1 Introduction

For effective communication, spoken words and their comprehension only is not sufficient, rather understanding of facial expressions is very important as facial expressions communicate certain messages important for effective communication such as emotional state which the other person involved in conversation might not communicate intentionally or unintentionally. Hence a lot of research is going on to automate emotion recognition from facial expressions.

Most facial expression recognition methods attempt to recognize six prototypic expressions (namely joy, surprise, anger, disgust, sadness and fear) proposed by Ekman [2]. Over the last decade, many research works have done facial expression analysis from still images. Neural networks is often used [4,5,6]. In [4,6] it was applied directly on face images, while in [6,5] it was combined with methods such as PCA , ICA or Gabor wavelet filters. Psychological studies show that facial image sequences often produce more accurate and robust recognition compared to static images [3]. Therefore, recent attention has been moving to model the facial expression dynamics through integrating temporal information. Some prominent work includes using Dynamic Texture [7] and Dynamic Graphical Model [8]. Yang et al. [9] used dynamic Haar-like feature, while Zhao et al. [7] extended the well-known local binary feature (LBP) to the temporal domain and applied it to facial expression recognition. Recently, LDA, a tool from the statistical text community has also been widely used to solve computer vision problems, e.g. object discovery [11] and scene categorization [12].

In this paper, we propose a novel LDA and HMM based technique for emotion recognition from facial expressions. The technique proposed works on an image sequence. The emotions dealt with are: happy, fear, sad, surprise, angry, disgust and contempt. These emotions, except for contempt are referred to as basic emotions as proposed by Ekman [14].These emotions are universally displayed and recognized from facial expressions and are not culturally determined.

Latent Dirichlet Allocation is a technique conventionally used with Bag of Words (BoW) model in text document processing, and information retrieval. In BoW model, each textual document contains some words, where each word is one of the words present in a dictionary already defined. The assumption in Bag of Words model is that order of words doesn't matter. For example, "a good book" and "book good a" are the same under this model. Thus each document can be said to contain a bag of words (because in a bag there is no order in which things may be present). Now after all documents have been represented by bags of words, we apply LDA. LDA is a generative three-level hierarchical Bayesian model, in which each item of a document is modeled as a finite mixture over an underlying set of topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. The topics are hidden topics learnt in an unsupervised manner. In the context of text modeling, the topic probabilities provide an explicit representation of a document. Thus after LDA has been performed, a document can be represented by a mixture of topics and also a topic can be associated with each word. This information can be used in the next step for document classification or categorization.

The technique proposed here starts with the application of BoW to an image sequence. An image sequence can be taken as analogous to a document in Natural text processing. Now, dictionary of words need to be defined. However, "word" in an image sequence is not off-the-shelf thing like the word in text documents. To achieve this, it includes two steps: Feature detection and Dictionary generation. In feature detection step, we find a feature vector for each frame of the sequence. The next step is to convert feature vectors to "codewords" (analogous to words in text documents), which also produces a "codebook" (analogous to a word dictionary). A codeword can be considered as a representative of several similar feature vectors. One simple method is performing k-means clustering over all the vectors.

In data mining, k-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. This results in a partitioning of the data space into Voronoi cells. K-means is an simplest unsupervised learning algorithms. The main idea is that the number of clusters is fixed a priori and a centroid is defined for each cluster. These centroids should be placed in a cunning way because different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as new centers of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid in a loop. As a result of this loop we may notice that the k centroids change their location step by step until centroids do not move any more. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Codewords are then defined as the centers of the learned clusters. The number of the clusters is the codebook size (analogy to the size of the word dictionary). It may be noted that each word is a point in a feature space, and hence has the same dimensions as the feature vector. However, each word can be assigned an index in the dictionary. Now once the dictionary is defined, then to each feature vector we can assign a word, which is nothing but the index of the word, in the dictionary, which is nearest to that feature vector.

After BoW model step, each image sequence document is associated with a set of words. In the next step, we apply Latent Dirichlet Allocation, which assigns a topic corresponding to each word in the image sequence document. And thus a document can now be associated with a set of topics. Now though LDA doesn't take into account the order in which words appear in the document, but since we find one word for each frame, we know the order of words in the image sequence document. This means that we also know the order of topics in each document. We leverage this information for classifier training in the final stage. The feature vector for the classifier is thus a set of topics, which appear in a specific order. It may be noted that classifiers like Neural Network and SVM can't be used as the number of frames in different sequences may vary, and hence the size of feature vector may vary for different sequences. Also since we know the order in which topics appear, we employ Hidden Markov Models as the classifier.

A hidden Markov model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (hidden) states, which control the mixture component to be selected for each observation. A HMM model is specified by:

- The set of states, S , which are hidden
- The prior probabilities $\pi_i = P(q_i = s_i)$ are the probabilities of s_i being the first state of a state sequence
- The transition probabilities $P(q_{n+1} = sj \mid q_n = s_i)$ are the probabilities to go from state i to state j
- The emission probabilities characterize the likelihood of a certain observation x, if the model is in state s_i

The operation of a HMM is characterized by

- The (hidden) state sequence $Q = [q_1 q_2 \ldots\ldots q_n]$ $q_n \in S$
- The observation sequence $X = [X_1 X_2 \ldots\ldots X_{n-1}]$

A separate HMM is learnt for each emotion category, each of which can learn facial expression dynamics for that emotion category only. To train a HMM for an emotion type, the sequence of topics for all the sequences in the training data set corresponding to that emotion category is shown to HMM to learn the dynamics of facial expression for that emotion.

We compare our results with another technique, in which the sequence of words, instead of topics, is used to learn the facial expression dynamics for each emotion.

The rest of this paper is organized as follows. In Section 2.1, we describe the overall algorithm proposed in this paper. In Sec. 2.2, we discuss representation of face and features chosen. In Sec.2.3, we discuss the extraction details of Words and Topics respectively. Sec. 2.4 discusses the classifier used for classification. Sec. 3 discusses the algorithm for the compared approach. Sec. 4 is the results section. Sec. 5 discusses observations. Sec. 6 summarizes the paper followed by acknowledgements and references.

2 Emotion Classification

2.1 Algorithm

The approach taken to solve the problem is divided into following steps:

A. Representation of face and extraction of feature vectors
B. Representation of Image Sequence as Topics
C. Expression Recognition Using Emotion Specific HMMs

2.2 Representation of face and extraction of feature vectors

To extract information for classification from the face, face is first represented by a set of MPEG-4 feature points, and some extra points as shown in Fig.1. The points are selected around eyes, eye brows, nose and mouth, as facial expressions can be characterized by the motion-deformation information of these facial features. In all 37 feature points are used.

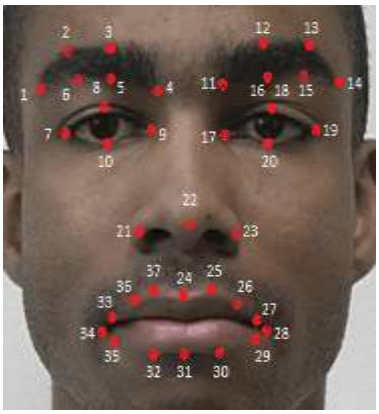

Fig. 1. Position of 37 facial points selected on face. The above pictures have been referenced from Extended Cohn-Kanade Database CK+)

The points P_9 and P_{17} on inner corners of the eyes and the nasal spine point $P_{2\,2}$ are called referential facial points because contractions of the facial muscles do not affect these points. These points can get displaced only due to rigid head motion. Thus displacement of a facial point can be measured relative to one of these points to cancel out any displacement of that facial point due to rigid head motion. The displacement then obtained is purely due to facial expressions changes.

Once the facial points have been selected in the first frame, these are tracked in the subsequent frames. For tracking any established point tracker may be used. We used Lucas Kanade tracker.

Feature Vectors

Since the facial expressions can be characterized by the motion-deformation information of facial features such as eyes, eye brows, and mouth, so we take as features the motion-deformation information of these facial features for our analysis. The motion-deformation information of various facial features is measured as the displacement of various facial points selected on the face. The set of features extracted and the corresponding facial points used for feature extraction is discussed in Table 1.

Feature	Information Contained	Set of feature points used for feature extraction	Direction of displacement measured
1	Mouth stretch	27,28,29 and 33,34,35	Horizontal
2	Mouth open	30,31,32 and 24,25,37	Vertical
3	Eye brow rise/lowering	2,3,5,6,12,13,15,16	Vertical
4	Eye brow stretch	2,3,5,6 and 12,13,15,16	Horizontal
5	Eye open	8,18 and 10,20	Vertical
6	Displacement of outer corners of eye brows along horizontal direction	1 and 14	Horizontal
7	Displacement of inner corners of eye brow along horizontal direction	4 and 11	Horizontal
8	Displacement of outer corners of eye brows along vertical direction	1 and 14	Vertical
9	Displacement of inner corners of eye brows along vertical direction	4 and 11	Vertical

Table 1: Information extracted in each feature ,the set of points used to measure corresponding feature and the direction in which displacement of feature points is measured.

Now the displacement of facial points may be measured with respect to neutral face. Also depending upon the approach it may be measured with respect to that in previous frame. In the technique proposed, both can be used, and hence analysis was done on 3 different feature vectors FV1, FV2 and FV3:

FV1: A 9-dimensional feature vector containing features as discussed in Table 1 , where the displacement is measured with respect to previous frame. Thus local temporal information is extracted.

FV2: A 9-dimensional feature vector containing features as discussed in Table 1 , where the displacement is measured with respect to neutral face (first frame in CK+ database). Thus local temporal information is extracted.

FV3: A 18-dimensional feature vector obtained as FV2 concatenated to FV1. Thus both local temporal as well as global temporal information is contained in this feature vector.

The other set of features that can be extracted from the tracked feature points is the displacement of each feature point along x and y direction. We try these features as well for our analysis. The features are extracted in a 74-d feature vector FV4 and the displacement is measured with respect to previous frame.

Now to account for pose variation problems due to rigid head motion, the displacement of each feature point is calculated relative to nasal referential point (P_{23} in Fig. 1) for all feature vectors.

Further though two image sequences may show similar facial expressions, but sometimes the feature vector may vary largely due to inter-person variations in facial features or scale variations. To solve this problem, we normalize the feature vector as discussed below:

Let an image sequence I contains n frames and a p-dimensional feature vector is extracted for each frame starting from second frame.

Now let $f_{maximum}$ = maximum {absolute ($f_{i,j}$)} such that $1 \leq i \leq n$ and $2 \leq j \leq p$.

The normalized feature $f_{normalized}$ is then obtained as $f_{normalized\ i,j} = f / f_{max}$ for $1 \leq i \leq n-1$ and $2 \leq j \leq p$.

Let a p-dimensional feature vector be represented as

$$F = [\ f_1 \quad f_2 \quad f_3 \ldots\ldots \quad f_p\ ]$$

If there are 'n' frames in a video sequence then a video sequence can be represented as

$$V = [\ F_1 \quad F_2 \quad F_3 \ldots\ F_{n-1}]$$

Each feature is assumed to have been normalized.

2.3 Representation of Image Sequence as Topics

Before representing the image sequence as a set of topics, it is represented as a set of words. Before we do that we need to define a dictionary of words. So once we have extracted the feature vector for all the image sequences in the training dataset, we employ k-means clustering for dictionary generation.

Let the word defined by k-means clustering can be represented as

$$W = [\ w_1 \quad w_2 \quad w_3 \ldots\ w_p]$$

And a dictionary of size 100 be represented as

$$D = [\ W_1 \quad W_2 \quad W_3 \ \ldots\ \ W_{100}]$$ where each W is p-dimensional word.

Once we have defined the dictionary, each feature vector is assigned a word, such that the word assigned is nearest to that feature vector. The distance measure used is using Squared Euclidean Distance.

Let for a feature vector F_j, the word assigned to it be W_{Fj}

Now we assume that W_i is mapped to i^{th} index in dictionary index.

So now a video can be represented as a sequence of index numbers only. Let for feature vector F_j, the word assigned to it be W_{Fj}. Let W_{Fj} comes at index value I_{WFj} such that $1 \leq I_{WFj} \leq 100$ for all j.

Then now a video can be represented as

$$V = [\ I_{WF1} \ I_{WF2} \ I_{WF3\ \ldots.} \quad I_{WF\ n-1}].$$

So a video can be represented as a sequence of just n-1 index values each of which is an integer € [1,100].

After each document in the training dataset has been represented as a set of words, then topics are extracted from them using Gibbs Sampler LDA.

So if earlier an image sequence was represented as $V = [\ I_1 \quad I_2 \quad I_3 \ \ldots\ \ I_{n-1}]$, then it can now be represented as

$$V = [\ T_1 \quad T_2 T_3 \ \ldots\ \ T_{n-1}]$$

So after this step, we can think of an image sequence in terms of topics.

2.4 Expression Recognition Using Emotion Specific HMMs

LDA clusters co-occurring words into topics, and the topic probabilities provide an explicit representation of an image sequence. However for final classification of image sequence into one of the 7 emotion categories or neutral, we need a classifier. Here in this piece of research, HMM is used as classifier. A separate HMM is learnt for each emotion category, each of which can learn facial expression dynamics for that emotion category only. To train a HMM for an emotion type, the sequence of topics for all the sequences in the training data set corresponding to that emotion category is shown to HMM to learn the dynamics of facial expression for that emotion.

2.5 Classification of a new image sequence

So given a new image sequence, first the facial points are selected in the first frame. Then facial points are then tracked in the subsequent frames using KL tracker. Then the feature vector is extracted for each frame of image sequence. Then each frame is represented as one of the words learnt during the training phase. Then for each word, one of the learnt topics can be extracted. Finally the sequence of topics is fed to each HMM trained, and the output of each HMM is compared to get the classification label.

3 Algorithm for Word sequence based learning of facial expressions dynamics by HMM

The recognition results obtained from the proposed technique are compared with another technique in which for HMM training, we use the sequence of words obtained in the previous approach instead of topics. The rest of the steps remain the same as for the proposed approach. Thus approach taken to solve the problem is divided into following steps:

A. Representation of face and extraction of feature vectors
B. Representation of Image Sequence as words
C. Expression Recognition Using Emotion Specific HMMs

4 Results

The k-means algorithm is significantly sensitive to the value of k which needs to be fixed apriori. So, to get optimum value of k, we ran the algorithm using different values of k. Further corresponding to a given dictionary size, the algorithm was tried for of different values of topic counts. The different variations tried were :

Dictionary size =25 Topic count: 10 and 25.
Dictionary size =50 Topic count: 10, 25 and 50.
Dictionary size =100 Topic count: 25, 50, 75 and 200.
Dictionary size =200 Topic count: 50, 100, 150 and 200.

To assess how well the results of this approach will generalize to an independent data set, we used 10-fold cross validation on entire labeled CK+ database.

The results shown in table below can be interpreted as:

- Positive refers to number of samples correctly classified.
- Negative refers to number of samples wrongly classified.
- No class refers to number of samples which were not recognized by any of the HMM classifiers learnt. This happens when there are not enough variations in training data to learn all the probabilities by HMM. The problem of 'No Class' can be solved by increasing the database size and including as much variations in database as possible.
- Accuracy Excluding No Case: It is calculated as Positive*100 / (Positive + Negative).
- Accuracy Including No Case: It is calculated as Positive*100 / (Positive + Negative + No Class). Thus cases not recognized by HMM are considered as falsely recognized cases.

Table 2 presents best results corresponding to each feature vector for proposed approach.

Table 3 presents best results for 2 accuracies (one including No Class cases and other Excluding No Class Cases) corresponding to each feature vector.

Feature Vector	Word Count	Topic Count	Negative	No Class	Positive	Accuracy excluding No Class cases	Accuracy including No Class cases
1	25	25	101	0	289	74.10	74.10
2	25	25	77	1	312	80.21	80.00
3	100	25	75	0	315	80.77	80.77
4	100	25	84	0	306	78.46	78.46

Table 2. Results for Proposed Approach

Feature Vector	Word Count	Negative	No Class	Positive	Accuracy excluding No Class cases	Accuracy including No Class cases
1	200	41	105	244	85.61	62.56
1	50	72	17	301	80.70	77.18
2	200	34	63	293	89.60	75.13
2	50	61	18	311	83.60	79.74
3	200	33	90	267	89.00	68.46
3	25	68	4	318	82.38	81.54
4	200	33	95	272	89.18	68.00
4	50	52	15	323	86.13	82.82

Table 3. Results for Compared Approach

The results obtained are near the state of art results. The best accuracy for the proposed approach is 80.77 % using feature vector 3, word count = 100 and topic count = 25 for both type of accuracies. For the compared approach, if accuracy is calculated excluding cases which couldn't be recognized by any of the HMM, then the best accuracy obtained is 89.18% for feature vector 4 and dictionary size =200. If accuracy is calculated considering cases which couldn't be recognized by any HMM as falsely recognized cases, then the best accuracy obtained is 82.82% for feature vector 4 and dictionary size=50. The results reflect using the sequence of words for classification gives better results in general.

When sequence of words is used, then number of samples which couldn't be recognized by any of the HMM is normally high. The count increases as the size of dictionary increases, which however can be solved by using larger database. This problem gets solved by LDA based approach as can be inferred from results. Thus LDA based approach tends to get better when size of database is small.

5 Discussion

In this paper, we explore the application of Latent Dirichlet Allocation, a technique conventionally used in Natural text processing, used with Hidden Markov Model (HMM), for learning the dynamics of facial expression for different emotions, for the classification task. The proposed technique represents each frame of an image sequence as a word. Then considering the image sequence as a document, we find the associated topic for each word. The sequence information of topics is then used to get the emotion label. The results have been compared with the approach, when sequence information of words is used to get the emotion label. The results reflect using the sequence of words for classification gives better results in general. However, LDA based approach tends to get better when size of database is small.

In this model, given a typical image sequence, words are more likely to be drawn from the same topic rather than different ones. Thus application of LDA helps in reducing noise which may appear till each frame is represented as a word. Also the technique proposed works with an image sequence and uses spatio-temporal infor-

mation which is believed to lead to improved accuracy. Further since HMM is used in the last step, hence the proposed technique can be used with image sequences of varying length, which is important as the dynamics of facial expressions may vary from person to person and also with recording device used. The simple displacement features have been used. However the beauty of the algorithm is that some other features can also be used in the first step.

The proposed technique can also be used for other type of facial expression analysis, such as AU recognition only with little modification. It can also be used for classifying different temporal phases of facial expression, such as neutral to peak, peak to neutral etc.

The application of HMM along with LDA can be used in text classification, which takes it beyond just Bag of Words approach and helps in reducing the noise.

Acknowledgements

The authors gratefully acknowledge the support of the DIPR-sponsored project, "Human Emotion Recognition using Computer Vision".

References

1. P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression"
2. P. Ekman, W.V. Friesen, "Facial Action Coding System (FACS): Manual",Consulting Psychologists Press, Palo Alto (1978)
3. J. Bassili, "Emotion recognition: the role of facial movement and the relative importance of upper and lower areas of the face", Personality and Social Psychology, 2049–2059 (1979)
4. W. Fellenz, J. Taylor, N. Tsapatsoulis, S. Kollias,"Comparing template-based, feature-based and supervised classification of facial expressions from static images", Proceedings of Circuits, Systems, Communications and Computers (CSCC'99), Nugata, Japan, 1999, pp. 5331–5336.
5. M. Dailey, G. Cottrell, "PCA Gabor for expression recognition" , Institution UCSD, Number CS-629, 1999.
6. C. Lisetti, D. Rumelhart, "Facial expression recognition using a neural network", Proceedings ofthe 11th International Flairs Conference, AAAI Press, New York, 1998.
7. G. Zhao, M. Pietik¨ainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions", IEEE Trans. on PAMI 29(6), 915–928
8. Y. Zhang, Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequences", IEEE Trans. on PAMI 27(5), 699–714 (2005)
9. P. Yang, Q. Liu, D.N. Metaxas," Boosting coded dynamic features for facial action units and facial expression recognition", CVPR, pp. 1–6 (2007)
10. D. Blei, A. Ng, M. Jordan, "Latent Dirichlet allocation", JMLR 3(2-3), 993–1022 (2003)
11. X. Wang, , J. Grimson," Spatial latent Dirichlet allocation", NIPS (2007)
12. L. Fei-Fei., P. Perona, "A Bayesian hierarchical model for learning natural scene categories" CVPR, pp. 524–531 (2005)
13. B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proceedings of the 1981 DARPA Imaging Understanding Workshop* (pp. 121–130), 1981.
14. P. Ekman, W.V. Friesen, Constants across cultures in the face and emotion, J. Personality Social Psychol. 17 (2) (1971) 124–129.