



**The Google Challenge: Video Genre
Classification**

Meng Wang, Yuecheng Shao

May 3, 2010

Boston University

Department of Electrical and Computer Engineering

Technical report No. 2010-05

**BOSTON
UNIVERSITY**

**The Google Challenge: Video Genre
Classification**

Meng Wang, Yuecheng Shao



Boston University
Department of Electrical and Computer Engineering
8 Saint Mary's Street
Boston, MA 02215
www.bu.edu/ece

May 3, 2010

Technical Report No. ECE-2010-05

Summary

Video files are large and usually take long time to view. Manual browsing and organizing a big collection of video files is tedious and impractical. The goal of this project is to develop an automatic method for video classification based on their genre,. Take movie as examples, the program will classify movies into action movies, comedy movies, tragedy movies, etc. In other words, our algorithm can ‘understand’ the ‘plot’ of a movie based on its image content without using metadata or other manual annotations.

Conventional photometric features, e.g. color histogram and SIFT bear little semantic meaning. So we propose other ways that can capture this semantic meaning. We argue that facial expression of characters and interactions are very much related to the story line. A graph of such interactions will be a good visualization of movies.

To begin with, we use face detector to detect all the characters. Clustering on the detection result will reveal the main characters of the movies. A supervised clustering technique has been tested on face images from movie, and yield promising results.

Facial expressions of the main characters are recognized to further estimate the emotional variations. We have shown that concepts such as happy, angry and sad can be learnt using collective resources like Flickr and Google.

Inspired by social network visualization techniques, a novel graphical method is presented to extract and visualize genre of a movie.

Contents

1. Introduction.....	1
2. Method Description	1
2.1. Face Detection	2
2.2. Face Clustering.....	2
2.3. Emotion Analysis	4
2.4. Character Graph.....	6
3. Experimental results.....	7
3.1. Face Detection	7
3.2. Emotion Analysis	8
3.3. Face Clustering.....	10
3.4. Character Graph.....	11
4. Conclusions.....	12
5. Future work.....	12
6. References.....	13

List of Figures

Fig. 1	Character graph can help the user to classify the movies	1
Fig. 2	System flow chart	1
Fig. 3	2D Haar features	2
Fig. 4	A video window to calculate the connection matrix R.....	6
Fig. 5	Samples of detected faces from a video	8
Fig. 6	Samples of extracted face images with positive emotions.....	8
Fig. 7	Sample of smile detection result from test video	9
Fig. 8	Clustering result using traditional clustering method	10
Fig. 9	Clustering result using supervised clustering	10
Fig. 10	Character graph using the clustering result	11
Fig. 11	Character graph using the clustering result	11

Introduction

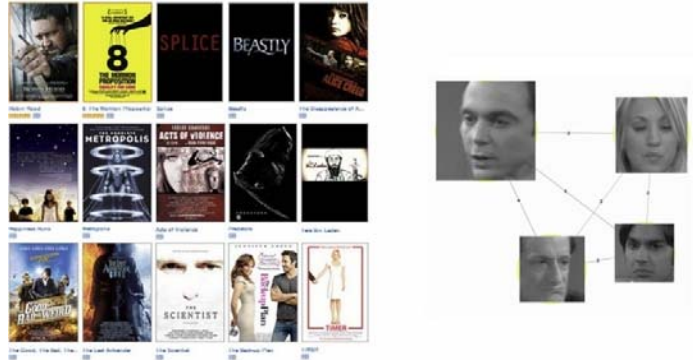


Fig.1 Character graph can help the user to classify the movies

The movie industry has become a very important part of our lives. Hundreds of movies are made every day. It is often hard for people to find the movie match their tastes. Our task is to classify these movies by their genre without using metadata or other manual annotations. We proposed a scheme that automatically extract character graph from movies and different kinds of character graph will represent different kinds of movies. Such graph can help the user to choose their desired movies, and can be used as a semantic feature in movie recommendation system.

2 Method Description

There are five main parts in our system: input movies, face detection, face clustering, emotion analysis and character graph formation.

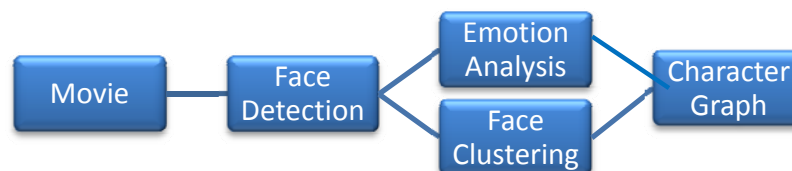


Fig.2 System flow chart

Face detection will extract face images from a video. These face images are further used to analyze facial expressions and to cluster to find out the main characters. At last, character labels are combined with frame labels (frame numbers) and emotion labels to

build a relationship graph of the main characters which can be the feature used for video genre classification. A discussion of the details of each function block will be presented in following sections.

2.1 Face Detection

Viola-Jones object detection method [5] is applied to detect faces in a video. The features employed by the detection framework universally involve the sums of image pixels within rectangular areas. The value of any given feature is always simply the sum of the pixels within clear rectangles subtracted from the sum of the pixels within shaded rectangles (Fig. 3). We use an image representation called the integral image,

$$I'(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y')$$

In which I' is integral image and I is the original image. By using integral image, rectangular features can be evaluated in constant time. Figure 3 illustrates the four different types of features used in the framework.

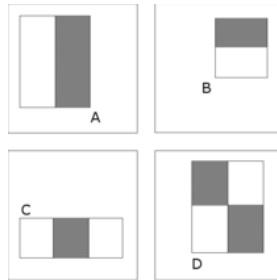


Fig.3 2D haar features

The speed with which features may be evaluated does not adequately compensate for their number. For example, in a standard 24x24 pixel sub-window, there are a total of 45,396 possible features, and it would be prohibitively expensive to evaluate them all. Thus, the object detection framework employs a variant of AdaBoost learning algorithm to both select the best features and to train classifiers that use them.

2.2 Face Clustering

In order to decide the number of characters using only the face images, assumptions are made such as the face images of the same person should be similar compared to those

of other people, so the face image corpus will form several clusters in an ideal feature space (invariant to pose, emotion, lighting, etc). Different clusters of face images represent different people. Here we tried out-of-the-box clustering method (e.g GMM, K-means, and spectrum clustering) to cluster the face images. Those methods fail badly because the images of the same person under various conditions (e.g. pose, lighting conditions and expressions) are quite different. Traditional clustering methods will falsely cluster different persons under the same condition into one cluster instead of clustering according to their identities. Because of the lack of clear criteria to cluster, those methods are bound to fail. To let those general-purposed clustering methods work, additional information is needed to break the ambiguity.

Although SIFT is known to be (partially) invariant to pose and illuminations, face images of the same person under different conditions can still form a separate cluster no matter what mainstream clustering method we try.

A specific clustering method draws our attention: supervised clustering, introduced by Thomas Finley[4]. In this approach, an auxiliary data is provided, including a series of sets of items and their complete corresponding clustering. This auxiliary data is used to guide the clustering such that future data is partitioned in the same fashion as training data. A similar approach, semi-supervised classification method uses both labeled and un-labeled data to classify the future data. The un-labeled data is used as a constraint (e.g manifold constraint) to adjust a classifier's fitting function. The future data is classified into existing labels. Supervised clustering, on the contrary, takes unlabeled data as future data, and since it is a clustering method, data will be assigned to the cluster index, rather than the existing label of labeled data.

In our system, the goal is to cluster the face images according to its owner's identity. We should guide the method to put faces belonging to the same person into one cluster, regardless of their conditions.

Inspired by supervised clustering methods, our auxiliary data is composed of images of several actors. We call those actors pivot actors, because those actors are used to span an identity space. For one actor, 25 images of him (her) are selected under different conditions (pose, lighting condition) to sample the 'condition space' of that person.

Observation shows that

$$P(h|I) = \int P(h, c|I) d c$$

Here h stands for character's identity, c for condition and I for image. The probability of identity given an image can be calculated and this quantity is invariant to condition of that image. All human faces are implicitly assumed to share a common condition space. By integrating $P(h, c|I)$ along the manifolds spanned by all the possible conditions, We obtain $P(h|I)$ which is invariant to any condition. In our experiment 25 images are used to approximate the condition space for each person. For one pivot actor A , calculating the probability of being A is achievable by

$$P(h = A|I) = \sum_{c=1}^{25} P(h = A, c|I)$$

A SIFT-based similarity measure is chosen to approximate $P(h = A, c|I)$. In fact, other similarity measures can be used here, e.g. SSD or covariance-based methods. By calculating the probability of being each pivot, $P(h = H_i|I), i \in \{pivot\ set\}$ is obtained. For current experiment, 6 actors are used. Each query image will have 6 real numbers which indicate how likely is each pivot actors. In this way, the dimensionality of each face image is reduced to 6.

A regularized Gaussian mixture model (GMM) clustering method is used on this 6 dimensional data. This method is adopted since the images of the same character should form a rather tight cluster already under our previous assumption on face conditions. Moreover, because the exact number of acts is unknown in a movie in advance, a heuristic prior is added: bias toward solutions with fewer mixtures. Specifically, by introducing a penalty term in EM framework of GMM, this is easily done. A parameter λ is used to tweak the strength of this bias: larger λ will lead to fewer clusters.

2.3 Emotion Analysis

For emotion analysis, we tentatively tried a new learning methodology: learning using collective data. With the fast growing number of images online, a vast amount of tagged images can be easily obtained via popular image repositories like Flickr or Google

image search. The following words are sample keywords used to get desired images containing positive emotion:

happy girl
smiling girl
happy woman
happy man

The following words are sample keywords used to get desired images containing negative emotion:

sad girl
crying girl
unhappy woman
unhappy man

For each keyword, we download around 2000 images. These raw images are not good for training a Viola-Jones classifier since they contain body parts and background. A pre-trained face detector is used to extract the face from those raw images to make these images more homogenous. The resulting images are faces with positive expression. In this way, more than 6000 happy faces are gathered.

An adaptive boost algorithm is implemented to train our classifiers. Our immediate experiment will focus on distinguishing happy and unhappy facial expressions, while keeping the ability to extend to other emotions. Viola-Jones (VJ) object detection method is used here. In the training step of the VJ method a subset of the image pool is chosen to build the positive training samples.

After training, the final classifier $H(x)$ is defined as a collection of Haar-like weak classifiers $h(x)$

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$$

$$h_t = \arg \min_{h_j \in \mathcal{H}} \epsilon_j, \text{ where } \epsilon_j = \sum_{i=1}^m D_t(i) [y_i \neq h_j(x_i)]$$

Here α_t are weights for weak classifiers. With such classifiers, happy facial expressions are able to be detected. The input x here is our input image, and H is one when the input has happy a facial expression.

2.4 Character Graph

For now, all the information needed to build our character graph is provided: character labels, frame labels and emotion labels. And those labels are expressed as column vectors in a label matrix L :

$$L = [C \quad F \quad E]$$

Here C , F and E stand for Character, frame and emotion labels.

Character labels are integer numbers from 1 to N . N is the number of characters.

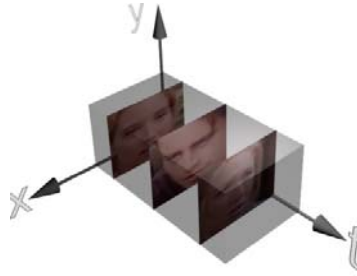


Fig.4 A video window to calculate the connection matrix R

The frame labels are frame numbers which indicate the time when characters appear. There are only 0 or 1 values in one emotion vector, in which 1 indicates smile face and 0 indicates other emotions, for the reason that only smile faces are detected for now. The future improvements will allow the algorithm to label different expressions like smile, cry and angry with different numbers. We define a connection matrix R as an N -by- N matrix in which N is the number of characters. In addition, it's a symmetric matrix with zeros on the diagonal.

The steps of calculating the connection matrix R are shown below:

1. First, a circular shift L by $n \in [1, M]$. By choosing different M , it is equivalent to choose an interval in which characters are considered to have connection. We denote this circular shifted matrix as L_n .
2. For every n , calculate the difference between L and L_n :

$$L(i,1) - L_n(i,1)$$

3. We only care about those i 's with non-zero differences which indicate the existence of different characters.
4. With those i 's from the last step, it is able to compute our R_n which denotes R computed from L and L_n .

$$R_n(l, k) = R_n(k, l) = \alpha e^{-(|L(i,2) - L_n(i,2)|)} + \beta [L(i, 3) + L_n(i, 3)]$$

in which $k = L(i,1)$, $l = L_n(i, 1)$, α and β are scale factors that control contributions of time interval and emotion.

5. Final step is to add R_n up to get R :

$$R = \sum_{n=1}^M R_n$$

To illustrate the connection matrix R , eigenvector centrality is used to measure the importance of characters and character graph is displayed in 2D space.

$$Rx = \lambda x$$

There will be many different eigenvalues λ for which an eigenvector solution exists, and two eigenvectors are selected with the largest eigenvalues as x and y coordinates of characters represented in R . Thus characters can be shown in 2D space where their relative positions (x and y for each character), indicate connection strength.

3 Experimental results

3.1 Face Detection

Using VJ face detection method, face images can be extracted successfully from videos.

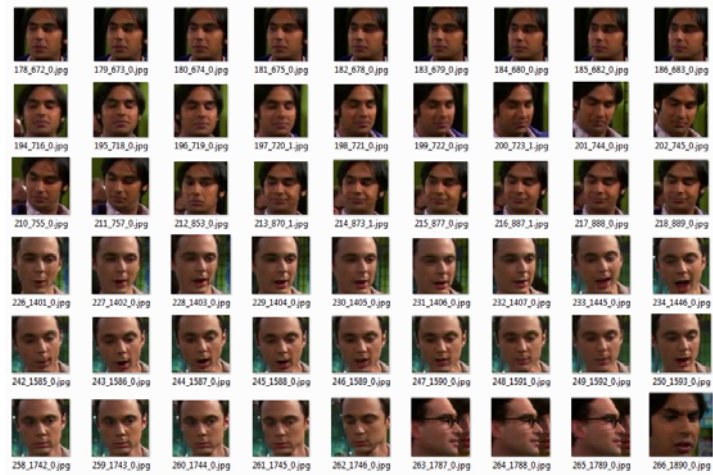


Fig.5 Samples of detected faces from a video

Figure 5 shows samples of face detection result from test video ‘The Big Bang Theory’.

3.2 Emotion Analysis

Using the collective images available online, more than 6000 face images have been gathered with positive emotions. The mosaic below shows a sample of these images.



Fig.6 Samples of extracted face images with positive emotions

So far, we have only achieved smile detection for emotion analysis; more emotions can be added. For example, using the keyword such as surprise or angry, the system

should be able to gather typical face images for each emotion, which can be used as training data for the VJ algorithm.

After training, our classifier was tested on movie clip ‘Friends’ and ‘The big bang theory’. This classifier could detect smiles in face images which have been extracted in advance using the VJ face detector.

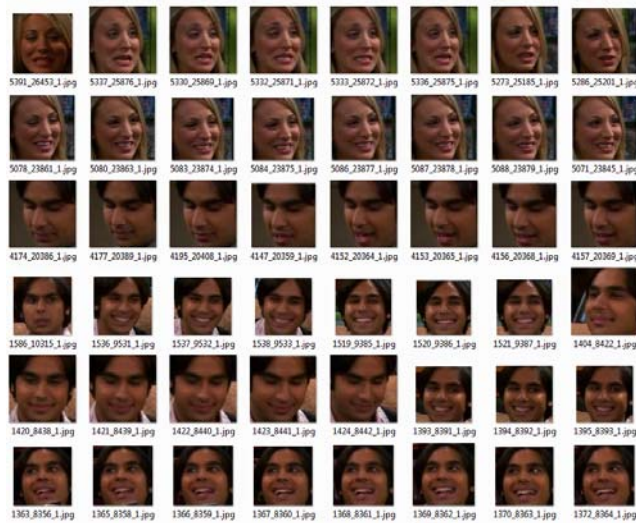


Fig.7 Sample of smile detection result from test video

From the result it is can be seen that our classifier is able to detect happy facial expressions. Although false positives may exist, a better training set will make our happy detector more accurate. Here, only quality result is exhibited. A more rigorous and quantitative result can be obtained in a form of ROC vs. size of the training dataset, which will indicate how useful this collective database can be. This test will require manual labeling all of the query images as smile/non-smile. In addition, it will test how many of them are correctly detected by this classifier. We will explore such performance in future work.

3.3 Face Clustering

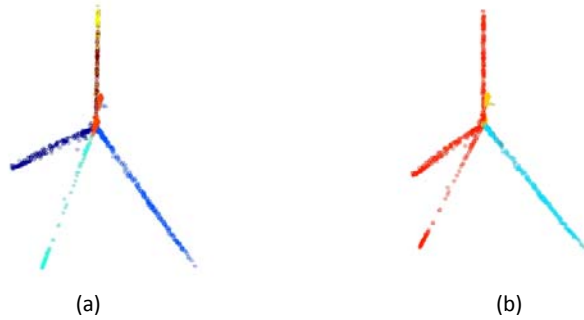


Fig.8 Clustering result using traditional clustering method

At first, a traditional clustering method was implemented, but the result is not reliable because face images of a single actor do not form a single tight cluster, preventing satisfying results. In the Figure 8(a), it partitions all the face images in one movie clip into 6 clusters (color coded in this figure, each color representing one cluster index). In fact, three of the clusters belong to one actor figure 8(b) (color coded in this figure, each color representing one actor).

Next, the supervised clustering method is carried out. More specifically, we label each of the resulting clusters with the majority identity in each cluster. Figure 9 shows a clustering result, using the first dimensions (out of 6) to plot this graph, color coded using the cluster index. The outcome shows that 76.8% of face images are correctly clustered.

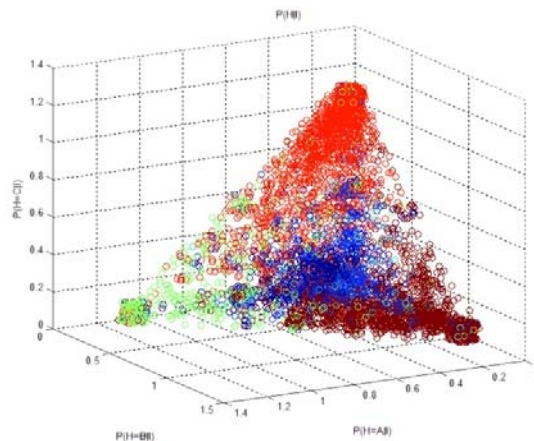


Fig.9 Clustering result using supervised clustering

3.4 Character Graph

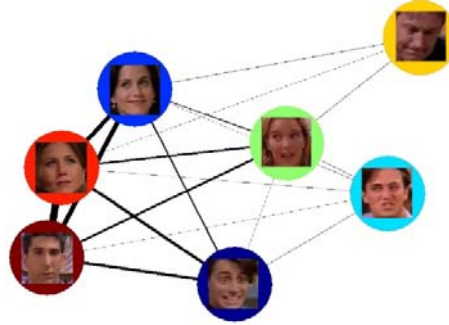


Fig.10 Character graph using the ground truth labels

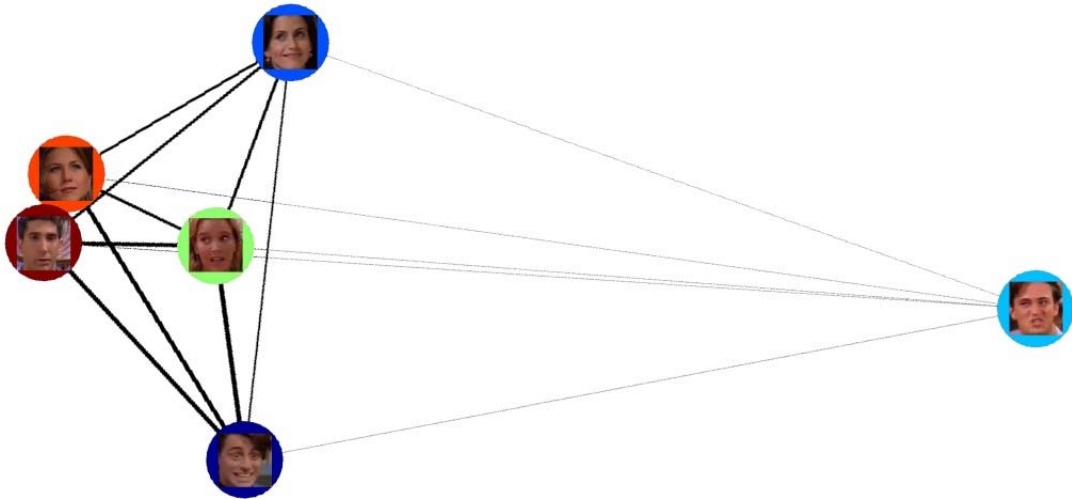


Fig.11 Character graph using the clustering result

In our test video, 6 characters are found by adopting the clustering methods to form a character graph. In each cluster, the image closest to the center is used as the representative image. The thickness of lines between characters is related to entries in the connection matrix, and their relative positions are related to 2D coordinates calculated by eigenvector centrality.

Compared to the ground truth character graph (we use ground truth character labels to compute this character graph), the relative positions of those characters are very similar.

4 Conclusions

Inspired by social network visualization techniques, a novel graphical method is presented to extract and visualize genre of a movie. In this work, visualizing the connection between characters is our main focus, i.e. how long they appear together in a movie and what's their facial expression.

In the experiment of emotion detection, the concepts such as facial expression can be learnt using collective resources like Flickr.

A supervised clustering technique has been tested on face clustering, and yields promising results.

5 Future work

Future improvement should exploit more robust facial features to relate frontal face to profile of the same person with the help of 3D face model. From different views, it is able to reconstruct 3D face of the same person, and then use this model as a template to compare the similarity with query faces.

More emotions can be added to our character graph. With reliable emotion results, instead of using this as a weight it will be used as an extra part of our character graph which will show emotion variation in a movie.

Moreover, character graph is going to be used as a feature to classify movies automatically. Action recognition and scene understanding can further enrich the content of our graph, leading to a more vivid representation of a movie.

User study is needed to get a quantitative evaluation of performance of this technique.

References

- [1] Josef Sivic and Andrew Zisserman, “Video Google: A Text Retrieval Approach to Object Matching in Videos”, SIVIC 2003,
- [2] Yushi Jing, and Shumeet Baluja, “VisualRank: Applying PageRank to Large-Scale Image Search”, PAMI 2008.
- [3] Alex Rav-Acha Yael Pritch Shmuel Peleg, “Making a Long Video Short: Dynamic Video Synopsis”, CVPR 2006.
- [4] Thomas Finley, “Supervised Clustering with Support Vector Machines”, ICML 2005
- [5] Viola, Jones, “Robust Real-time Object Detection”, IJCV 2001