

Automatic Cast Listing in Feature-Length Films with Anisotropic Manifold Space

Ognjen Arandjelović
University of Cambridge, UK
oa214@eng.cam.ac.uk

Roberto Cipolla
University of Cambridge, UK
cipolla@eng.cam.ac.uk

Abstract

Our goal is to automatically determine the cast of a feature-length film. This is challenging because the cast size is not known, with appearance changes of faces caused by extrinsic imaging factors (illumination, pose, expression) often greater than due to differing identities. The main contribution of this paper is an algorithm for clustering over face appearance manifolds. Specifically: (i) we develop a novel algorithm for exploiting coherence of dissimilarities between manifolds, (ii) we show how to estimate the optimal dataset-specific discriminant manifold starting from a generic one, and (iii) we describe a fully automatic, practical system based on the proposed algorithm. The performance of the system is evaluated on well-known feature-length films and situation comedies on which it is shown to produce good results.

1. Introduction

The problem that we address in this paper is that of automatically determining the cast of a feature-length film. Applications of this research include content-based retrieval, rapid browsing and organization of video collections, to name just a few. Our approach is based on the recognition of faces, faces being the most repeatable cue for person identification in this setting (although others, such as clothes [20], can be used for further performance improvement).

Problem challenges. The inherent difficulties of face recognition [1] and those specifically in the context of feature-length films [3, 11, 27] are well appreciated across the literature. Lighting conditions, and especially light angle, facial expressions and head pose drastically change the appearance of faces. Partial occlusions, be they objects in front of a face or resulting from hair style change also cause problems. To further complicate the problem, artefacts caused by motion blur and low spatial resolution are also common, see Fig. 1. In brief, the uncontrolled imaging

conditions in films provide an especially challenging working environment for automatic face recognition (AFR) algorithms.

The broad topic of AFR encompasses a multitude of different problem settings: classification (one-to-many matching)[16], verification (one-to-one matching) [15], retrieval by similarity [3, 27] etc. In this paper we consider the most difficult setting of all – fully automatic (i.e. without any dataset-specific training information) listing of the individuals present in a video.

1.1. Previous Work

Good general reviews of recent automatic face recognition (AFR) literature can be found in [4, 14, 35]. In this section, we focus specifically on methods that deal with AFR in a setting similar to ours.

Recent years have seen a development of algorithms that use AFR for the analysis of media content. Most of these deal with the *retrieval* problem in video [3, 11, 27]. Arandjelović and Zisserman [3] use signature images, obtained by a cascade of transformations of the detected faces. These are matched using a robust distance measure in an image-to-image or image-to-set fashion to retrieve film shots based on the presence of specific actors. Sivic *et al.* [27] match face sets, representing individual faces using SIFT descriptors corresponding to salient facial features. Everingham and Zisserman [11, 12] employ a quasi-3D model of the head to correct for varying pose and enrich the training corpus via shot tracking.

Visual clustering of individuals in movies was first attempted by Fitzgibbon and Zisserman [13]. Affine-invariant image-to-image matching was used to achieve robustness to pose and a simple band-pass filter to illumination changes. Berg *et al.* [6] consider the problem of clustering detected frontal faces extracted from web news pages. Faces are first affine registered and then classified in a Kernel PCA space using combined image and contextual text-based features. In this paper, we use only visual cues (i.e. no text).



Figure 1. The appearance of faces in films exhibits a great variability depending on the extrinsic imaging conditions. Shown are the most common sources of intra-personal appearance variations (all faces are from the same episode of the situation comedy “Yes, Minister”).

1.2. Method overview

The first idea of our work concerns the observation that some people are inherently more similar looking to each other than others. As an example from our data set, Sir Hacker (see Fig. 1) may be difficult to distinguish from his secretary, Sir Humphrey (see Fig. 8 in §2.3), but he is unlikely to be mistaken, say, for his wife (see Fig. 4 in §2.1). The problem is then of automatically extracting and representing the structure of these inter-personal similarities from unlabelled sets of video sequences. We show that this can be done by working in what we term the *manifold space* – a vector space in which each point is an appearance manifold.

The second major contribution of this paper is a method for unsupervised extraction of inter-class data for discriminative learning on an unlabelled set of video sequences. In spirit, this approach is similar to the work of Lee and Kriegman [21] in which a generic appearance manifold is progressively updated with new data to converge to a person-specific one. In contrast, we start from a generic *discriminative* manifold and converge to a data-specific one, *automatically* collecting within-class data.

An overview of the entire system is shown in Alg. 1.

2. Method Details

In this section we describe each of the steps in the algorithmic cascade of the proposed method: (i) automatic data acquisition and preprocessing, (ii) unsupervised discriminative learning and (iii) clustering over appearance manifolds.

2.1. Automatic Data Acquisition

Our cast clustering algorithm is based on pair-wise comparisons of face *manifolds* [2, 22, 24] that correspond to sequences of moving faces. Hence, the first stage of the proposed method is automatic acquisition of face data from a continuous feature-length film. We (i) temporally segment the video into *shots*, (ii) detect faces in each and, finally, (iii) collect detections through time by tracking in the (X, Y, scale) space.

Shot transition detection. A number of reliable methods for shot transition detection have been proposed in the lit-

Algorithm 1 Method Overview

Input: film frames $\{f_t\}$,
generic discrimination subspace \mathbf{B}_G .

Output: cast classes \mathbb{C} .

- 1: **Acquisition: face manifolds**
 $\mathbf{T} \leftarrow \text{get_manifolds}(\{f_t\})$
- 2: **Synthetically repopulate manifolds**
 $\mathbf{T} \leftarrow \text{repopulate}(\mathbf{T})$
- 3: **Adaptive discriminative learning: distance matrix**
 $\mathbf{D}_S \leftarrow \text{distance}(\mathbf{T}, \mathbf{B}_G)$
- 4: **Manifold space**
 $\mathbb{M} \leftarrow \text{MDS}(\mathbf{T}, \mathbf{B}_G)$
- 5: **Get initial classes**
 $\mathbb{C} \leftarrow \text{classes}(\mathbb{D}_G)$
- 6: **Anisotropic boundaries in manifold space**
for $\mathbf{C}_i, \mathbf{C}_j \in \mathbb{C}$
- 7: **Get PPCA models**
 $(\mathcal{G}_i, \mathcal{G}_j) \leftarrow \text{PPCA}(\mathbf{C}_i, \mathbf{C}_j, \mathbb{M})$
- 8: **Merge clusters w/ Description Length**
 $\Delta\text{DL}(i, j) < \text{threshold} ? \text{merge}(i, j, \mathbb{C})$
- 9: **end loop**

erature [18, 26, 33, 34]. We used the Edge Change Ratio (ECR) [33] algorithm as it is able in a unified manner to detect all 3 standard types of shot transitions: (i) hard cuts, (ii) fades and (iii) dissolves. The ECR is defined as:

$$ECR_n = \max(X_n^{in}/\sigma_n, X_{n-1}^{out}/\sigma_{n-1}) \quad (1)$$

where σ_n is the number of edge pixels computed using the Canny edge detector [8], and X_n^{in} and X_n^{out} the number of entering and existing edge pixels in frame n . Shot changes are then recognized by considering local peaks of ECR_n , exceeding a threshold, see [23, 33] for details and Fig. 2 for an example.

Face tracking through shots. We detect faces in cluttered scenes on an independent, frame-by-frame basis with

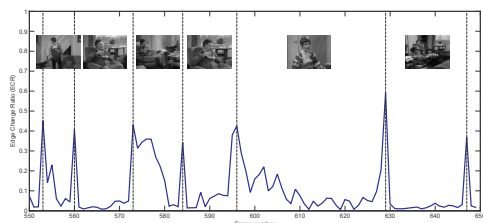


Figure 2. The unsmoothed Edge Change Ratio for a 20s segment from the situation comedy “Yes, Minister”.

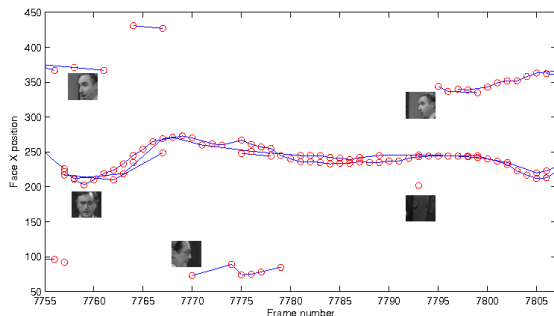


Figure 3. The X coordinate of detected faces (red dots) through time in a single shot and the resulting tracks connecting them (blue lines) as determined by our algorithm.

the Viola-Jones [31] cascaded algorithm¹. For each detected face, the detector provides a triplet of the X and Y locations of its centre and scale s . In the proposed method, face detections are connected by tracks using a simple tracking algorithm in the 3D space $\mathbf{x} = (X, Y, s)$. We employ a form of the Kalman filter in which observations are deemed completely reliable (i.e. noise-free) and the dynamic model is that of zero mean velocity $[\dot{\mathbf{x}}] = 0$ with a diagonal noise covariance matrix. A typical tracking result is illustrated in Fig. 3 with a single face track obtained shown in Fig. 4 (a).

2.2. Appearance Manifold Discrimination

Having collected face tracks from a film, we turn to the problem of clustering these (relatively) short sequences by identity. Due to the smoothness of faces, each track corresponds to an appearance manifold [2, 22, 24], as illustrated in Fig. 4. We want to compare these manifolds and use the structure of the variation of dissimilarity between them to deduce which ones describe the same person.

Data preprocessing. The first step in the comparison of two appearance manifolds is a simple preprocessing on a frame-by-frame basis that normalizes for the majority of illumination effects and suppresses the background. If \mathbf{X} is an image of a face, in the usual form of a raster-ordered

¹We used the freely available code, part of the Intel[®] OpenCV library.

Algorithm 2 Data-specific discrimination.

Input: manifolds $\mathbf{T} = \{T_i\}$.
generic discrimination subspace \mathbf{B}_G .

Output: distance matrix \mathbf{D}_S .

1: **Distance matrix w/ generic discrimination**

$$\mathbf{D}_G \leftarrow \text{distance}(\mathbf{T}, \mathbf{B}_G)$$

2: **Get provisional classes**

$$\mathcal{C}_T \leftarrow \text{classes}(\mathbf{D}_G)$$

3: **Data-specific discrimination space**

$$\mathbf{B}_S \leftarrow \text{constraint_space}(\mathcal{C}_T)$$

4: **Mixed discrimination space**

$$\mathbf{B}_C \leftarrow \text{combine_eigenspaces}(\mathbf{B}_S, \mathbf{B}_G)$$

5: **Distance matrix w/ data-specific discrimination**

$$\mathbf{D}_S \leftarrow \text{distance}(\mathbf{T}, \mathbf{B}_C)$$

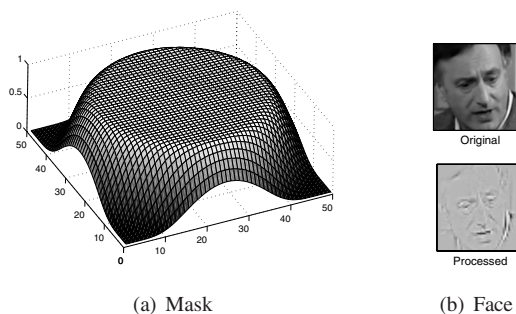


Figure 5. (a) The mask used to suppress cluttered background in images of automatically detected faces, and (b) an example of a detected, unprocessed face and the result of illumination normalization and background suppression.

pixel vector, we first normalize for the effects of illumination using a high-pass filter (previously used in [3, 13]) scaled by local image intensity:

$$\mathbf{X}_L = \mathbf{X} * \mathbf{G}_{\sigma=1.5} \quad (2)$$

$$\mathbf{X}_H = \mathbf{X} - \mathbf{X}_L \quad (3)$$

$$X_I(x, y) = X_H(x, y) / X_L(x, y). \quad (4)$$

This is similar to the Self-Quotient Image of Wang *et al.* [32]. The purpose of local scaling is to equalize edge strengths in shadowed (weak) and well-illuminated (strong) regions of the face.

Background is suppressed with a weighting mask \mathbf{M}_F , produced by feathering (similar to [3]) the mean face outline \mathbf{M} , as shown in Fig. 5:

$$\mathbf{M}_F = \mathbf{M} * \exp\left(-\left(\frac{r(x, y)}{4}\right)^2\right) \quad (5)$$

$$X_F(x, y) = X_I(x, y) \mathbf{M}_F(x, y). \quad (6)$$



Figure 4. A typical face track obtained using our algorithm. Shown are (a) the original images are detected by the face detector (rescaled to the uniform scale of 50×50 pixels) and (b) as points in the 3D principal component space with temporal connections.

Synthetic data augmentation. Many of the collected face tracks in films are short and contain little pose variation. For this reason, we automatically enrich the training data corpus by stochastically repopulating geodesic neighbourhoods of randomly drawn manifold samples.

Under the assumption that the face to image space embedding function is smooth, geodesically close images correspond to small changes in imaging parameters (e.g. yaw or pitch). Hence, using the first-order Taylor approximation of the effects of a projective camera, the face motion manifold is locally topologically similar to the *affine warp* manifold. The proposed algorithm then consists of random draws of a face image \mathbf{x} from the data, stochastic perturbation of \mathbf{x} by a set of affine warps $\{\mathbf{A}_j\}$ and finally, the augmentation of data by the warped images.

2.2.1 Comparing Normalized Appearance Manifolds

For pair-wise comparisons of manifolds we employ the Constraint Mutual Subspace method (CMSM) [15], based on principal angles between subspaces [19, 25]. This choice is motivated by: (i) CMSM's good performance reports in the AFR literature [2, 15], (ii) its computational efficiency [7] and compact data representation, and (iii) its ability to extract the *most similar* modes of variation between two subspaces.

As in [15], we represent each appearance manifold by a minimal linear subspace it is embedded in – estimated using Probabilistic PCA [29]. The similarity of two such subspaces is then computed as the mean of their first 3 canonical correlations, after the projection onto the *constraint subspace* – a linear manifold that attempts to maximize the separation (in terms of canonical correlations) between different class subspaces, see Fig. 6.

Computing the constraint subspace. Let $\{\mathbf{B}_i\} = \mathbf{B}_1, \dots, \mathbf{B}_N$ be orthonormal basis matrices representing subspaces corresponding to N different classes (cast mem-

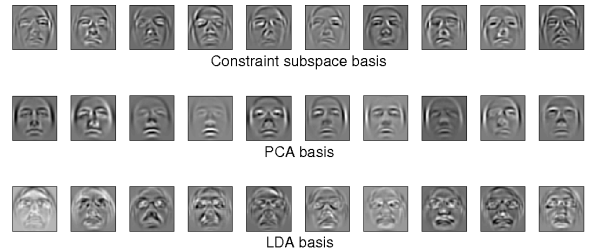


Figure 6. A visualization of the basis of the linear constraint subspace, the most descriptive linear subspace (eigenspace using PCA [30]) and the most discriminative linear subspace in terms of within and between class scatter (LDA [5]).

bers, in our case). Fukui and Yamaguchi [15] compute the orthonormal basis matrix \mathbf{B}_C corresponding to the constraint subspace using PCA from:

$$(\mathbf{B}_R \mathbf{B}_C) \begin{pmatrix} \Lambda_L & \mathbf{0} \\ \mathbf{0} & \Lambda_S \end{pmatrix} \begin{pmatrix} \mathbf{B}_R^T \\ \mathbf{B}_C^T \end{pmatrix} = \sum_{i=1}^N \mathbf{B}_i \mathbf{B}_i^T, \mathbf{B}_R^T \mathbf{B}_C = \mathbf{0} \quad (7)$$

where Λ_L and Λ_S are diagonal matrices with diagonal entries, respectively, greater or equal than 1 and less than 1. We modify this approach by weighting the contribution of the projection matrix \mathbf{B}_i by the number of samples used to compute it. This way, a more robust estimate is obtained as subspaces computed from smaller amounts of data (i.e. with lower Signal-to-Noise Ratio) are de-emphasized:

$$(\mathbf{B}_R \mathbf{B}_C) \begin{pmatrix} \Lambda_L & \mathbf{0} \\ \mathbf{0} & \Lambda_S \end{pmatrix} \begin{pmatrix} \mathbf{B}_R^T \\ \mathbf{B}_C^T \end{pmatrix} = N \sum_{i=1}^N N_i \mathbf{B}_i \mathbf{B}_i^T / \sum_{i=1}^N N_i \quad (8)$$

From generic to data-specific discrimination. The problem of estimating \mathbf{B}_C lies in the fact that we do not know which appearance manifolds belong to the same class and which to different classes i.e. $\{\mathbf{B}_i\}$ are unknown. We therefore start from a *generic* constraint subspace \mathbf{B}_C^g , computed offline from a large data corpus. For example, for the

evaluation reported in §3 we estimated \mathbf{B}_i , $i = 1, \dots, 100$ from 700 sequences (7 for each of the 100 people in the database) acquired in our laboratory.

Now, consider the Receiver-Operator Characteristic (ROC) curve of CMSM in Fig. 7, also estimated offline. The inherent tradeoff between recall and precision is clear, making it impossible to immediately draw class boundaries using the inter-manifold distance only. Instead, we propose to exploit the two marked salient points of the curve merely to collect data for the construction of the constraint subspace. Starting from an arbitrary manifold, the “high recall” point allows to confidently partition a *part* of the data into different classes. Then, using manifolds in each of the classes we can gather intra-class data using the “high precision” point. The collected class information can then be used to compute the basis \mathbf{B}_C^s of the *data-specific* constraint subspace.

The problem in using the above defined data-specific constraint subspace \mathbf{B}_C^s is that it is constructed using only the easiest to classify data. Hence, it cannot be expected to discriminate well in difficult cases, corresponding to the points on the ROC curve between “high precision” and “high recall”. To solve this problem, we do not *substitute* the data-specific for the generic constraint subspace, but iteratively *combine* the two based on our confidence $0.0 \leq \alpha \leq 1.0$ in the former:

$$\mathbf{B}_C = \text{mix}(\alpha, 1 - \alpha, \mathbf{B}_C^s, \mathbf{B}_C^g) \quad (9)$$

where α and $(1 - \alpha)$ are mixing weights. We used an eigenspace mixing algorithm similar to Hall *et al.* [17]. The mixing confidence parameter α is determined as follows. Consider clustering appearance manifolds using each of the two salient points. The “high precision” point will give an overestimate $N_h \geq N$ of the number of classes N , while the “high recall” one an underestimate $N_l \leq N$. The closer N_h and N_l are, the more confident we can be that the constraint subspace estimate is good. Hence, we compute α as their normalized difference (which ensures that the condition $0.0 \leq \alpha \leq 1.0$ is satisfied):

$$\alpha = 1 - \frac{N_h - N_l}{M - 1} \quad (10)$$

where M is the number of appearance manifolds.

2.3. The Manifold Space

In §2.2.1 we described how to preprocess and pair-wise compare appearance manifolds, optimally exploiting generic information for discriminating between human faces and automatically extracted data-specific information. One of the main premises of the proposed clustering method is that there is a structure to inter- and intra-personal distances between appearance manifolds. To discover and exploit this structure, we consider a *manifold space* – a vector space in which each *point* represents an appearance manifold. In the proposed method, manifold representations in this space are constructed implicitly.

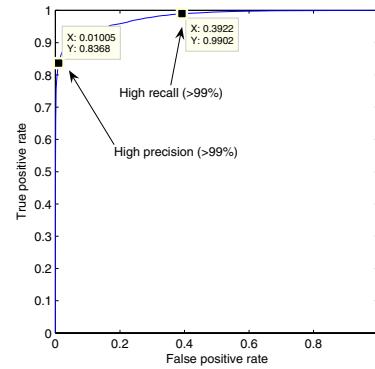


Figure 7. The ROC curve of the Constraint Mutual Subspace Method, estimated offline. Shown are two salient points of the curve, corresponding to high precision and high recall.

We start by computing a symmetric $N \times N$ distance matrix \mathbf{D} between all pairs of appearance manifolds using the method described in the previous section:

$$D(i, j) = \text{CMSM_dist}(i, j). \quad (11)$$

Note that the entries of \mathbf{D} do not obey the triangle inequality, i.e. in general: $D(i, j) \not\leq D(i, k) + D(k, j)$. For this reason, we next compute the normalized distance matrix $\hat{\mathbf{D}}$ using Floyd’s algorithm [9]:

$$\forall k. \hat{D}(i, j) = \min[D(i, j), \hat{D}(i, k) + \hat{D}(k, j)]. \quad (12)$$

Finally, we employ a Multi-Dimensional Scaling (MDS) algorithm (similarly as Tenenbaum *et al.* [28]) on $\hat{\mathbf{D}}$ to compute the natural embedding of appearance manifolds under the derived metric. A typical result of embedding is shown in Fig. 8.

Anisotropically evolving class boundaries. Consider previously mentioned clustering of appearance manifolds using a particular point on the ROC curve, corresponding to a distance threshold d_t . It is now easy to see that in the constructed manifold space this corresponds to hyper-spherical class boundaries of radius d_t centred at each manifold, see Fig. 9. We now show how to construct anisotropic class boundaries by considering the distributions of manifolds. First, (i) simple, isotropic clustering in the manifold space is performed using the “high precision” point on the ROC curve, then (ii) a single parametric, Gaussian model is fit to each provisional same-class cluster of manifolds, and finally (iii) Gaussian models corresponding to the provisional classes are merged in a pair-wise manner, using a criterion based on the model+data Description Length [10]. The criterion for class-cluster merging is explained in detail next.

Class-cluster merging. In the proposed method, classes are represented by Gaussian clusters in the implicitly computed manifold space. Initially, the number of clusters is

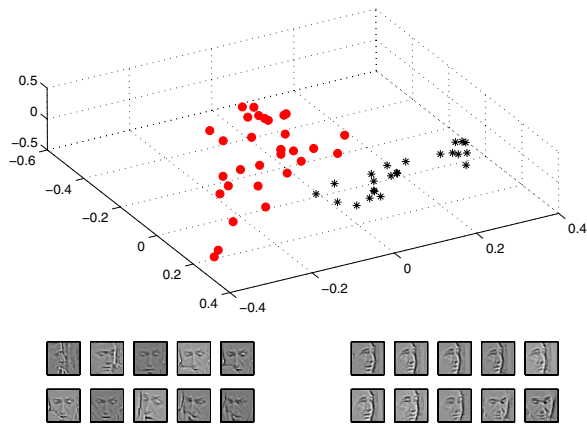


Figure 8. Manifolds in the manifold space (shown are its first 3 principal components), corresponding to preprocessed tracks of faces of the two main characters in the situation comedy “Yes, Minister”. Each red dot corresponds to a single appearance manifold of Jim Hacker and black star to a manifold of Sir Humphrey (samples from two typical manifolds are shown below the plot). The distribution of manifolds in the space shows a clear structure. In particular, note that intra-class manifold distances are often greater than inter-manifold ones. Learning distributions of manifolds provides a much more accurate way of classification.

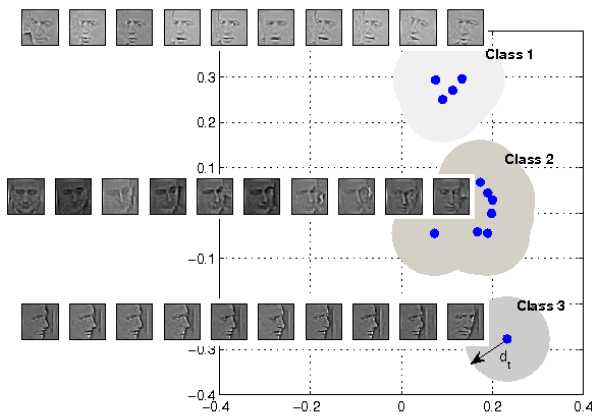


Figure 9. In the manifold space, the usual form of clustering – where manifolds within a certain distance (chosen from the ROC curve) from each other are grouped under the same class – corresponds to placing a hyper-spherical kernel at each manifold.

overestimated, each including only those appearance manifolds for which the same-class confidence is very high, using the manifold distance corresponding to the “high precision” point on the CMSM’s ROC curve. Then, clusters are pair-wise merged. Intuitively, if two Gaussian components are quite distant and have little overlap, not much evidence for each is needed to decide they represent different classes. The closer they get and the more they overlap, more supporting manifolds are needed to prevent merging. We quan-

tify this using what we call the *weighted Description Length* DL_w and merge tentative classes if $\Delta DL_w < \text{threshold}$ (we used $\text{threshold} = -20$).

Let j -th of C appearance manifolds be \mathbf{m}_j and let it consist of $n(j)$ face images. Then we compute the log-likelihood of \mathbf{m}_j given the Gaussian model $\mathcal{G}(\mathbf{m}; \Theta)$ in the manifold space, weighted by the number of supporting-samples $n(j)$:

$$C \sum_{j=1}^C n(j) \log P(\mathbf{m}_j | \Theta) / \sum_{j=1}^C n(j) \quad (13)$$

The weighted Description Length of class data under the same model then becomes:

$$DL_w(\Theta, \{\mathbf{m}_j\}) = \frac{1}{2} N_E \log_2(n(j)) - \left[\prod_{j=1}^C P(\mathbf{m}_i | \Theta)^{n(j)} \right]^{C / \sum n(j)} \quad (14)$$

3. Evaluation and Results

In this section we report the empirical results of evaluating the proposed algorithm on the “Open Government” episode of the situation comedy “Yes, Minister”². Face detection was performed on every 5th out of 42,800 frames, producing 7,965 detections, see Fig. 10 (a). A large number of non-face images is included in this number, see Fig. 10 (b). Using the method for collecting face motion sequences described in §2.1 and discarding all tracks that contain less than 10 samples removes most of these. We end up with approximately 300 appearance manifolds to cluster. The primary and secondary cast consisted of 7 characters: Sir Hacker, Miss Hacker, Frank, Sir Humphrey, Bernard, a BBC employee and the PM’s secretary.

Baseline clustering performance was established using the CMSM-based isotropic method with thresholds corresponding to the “high recall” and “high precision” points on the ROC curve. Formally, two manifolds are classified to the same class if the distance $D(i, j)$ between them is less than the chosen threshold, see (11) and Fig. 9. Note that the converse is not true due to the transitivity of the in-class relation.

3.1. Results and Discussion

The cast listing results using the two baseline isotropic algorithms are shown in Fig. 11 (a) and 11 (b) – for each class we displayed a 10 image sample from its most likely manifold (under the assumption of normal distribution, see §2.2.1). As expected, the “high precision” method produced

²Available at <http://mi.eng.cam.ac.uk/~oa214/academic/>



(a)



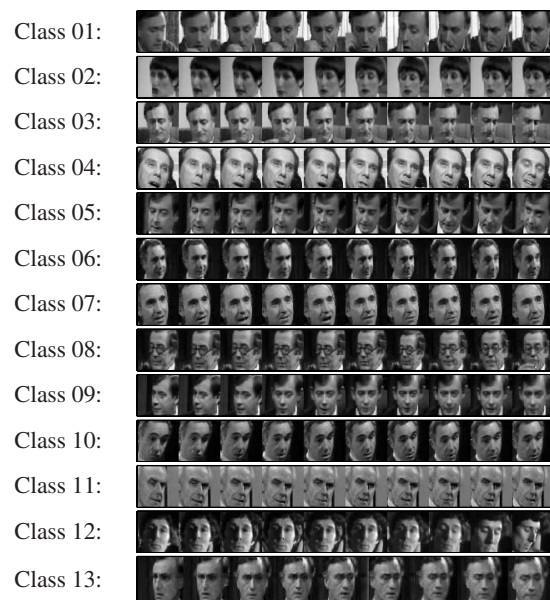
(b)

Figure 10. (a) The “Yes, Minister” data set – every 70th detection is shown for compactness. A large number of non-faces is present, typical of which are shown in (b).

a gross overestimate of the number of different individuals e.g. suggesting three classes both for Sir Hacker and Sir Humphrey, and two for Bernard. Conversely, the “high recall” method underestimates the true number of classes. However, rather more interestingly, while grouping different individuals under the same class, this result still contains two classes for Sir Hacker. This is a good illustration of the main premise of this paper, showing that the in-class distance threshold has to be chosen *locally* in the manifold space, if high clustering accuracy is to be achieved. That is what the proposed method implicitly does.

The cast listing obtained with anisotropic clustering is shown in Fig. 12. For each class we displayed 10 images from the highest likelihood sequence. It can be seen that the our method correctly identified the main cast of the film. No characters are ‘repeated’, unlike in both Fig. 11 (a) and Fig. 11 (b). This shows that the proposed algorithm for growing class boundaries in the manifold space has implicitly learnt to distinguish between intrinsic and extrinsic variations *between appearance manifolds*. Fig. 13 corroborates this conclusion.

An inspection of the results revealed a particular failure mode of the algorithm, also predicted from the theory presented in previous sections. Appearance manifolds corresponding to the “BBC employee” were classified to the class dominated by Sir Humphrey, see Fig. 13. The rea-



(a)



(b)

Figure 11. (a) “High precision” and (b) “high recall” point isotropic clustering results. The former vastly overestimates the number of cast members (e.g. classes 01, 03 and 13 correspond to the same individual), while the latter underestimates it. Both methods fail to distinguish between inter- and intra-personal changes of appearance manifolds.



Figure 12. Anisotropic clustering results – shown are 10 frame sequences from appearance manifolds most “representative” of the obtained classes (i.e. the highest likelihood ones in the manifold space). Our method has correctly identified 6 out of 7 primary and secondary cast members, without suffering from the problems of the two isotropic algorithms see Fig. 11 and Fig. 13.

son for this is a relatively short appearance of this character, producing a small number of corresponding face tracks.



Figure 13. Examples from the “Sir Humphrey” cluster – each horizontal strip is a 10 frame sample from a single face track. Notice a wide range of appearance changes: extreme illumination conditions, pose and facial expression variation. The bottom-most strip corresponds to an incorrectly clustered track of “BBC employee”.

Consequently, with reference to (13) and (14), not enough evidence was present to maintain them as a separate class. It is important to note, however, that qualitatively speaking this is a tradeoff inherent to the problem in question. Under an assumption of isotropic noise in image space, any class in the film’s cast can generate any possible appearance manifold – it is enough evidence for each class that makes good clustering possible.

Similar results to those shown were obtained on the film “Groundhog Day”.

4. Summary and Conclusions

The proposed method of extracting face appearance manifolds and anisotropically growing their class boundaries in the corresponding manifold space has been demonstrated to achieve good automatic cast listings in films.

In the future, we would like to employ a more sophisticated way of comparing appearance manifolds, which we believe will further increase the clustering robustness.

References

[1] Y. Adini, Y. Moses, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. *PAMI*, 19(7), 1997.

[2] O. Arandjelović, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density divergence. *CVPR*, 2005.

[3] O. Arandjelović and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. *CVPR*, 2005.

[4] W. A. Barrett. A survey of face recognition algorithms and testing results. *Systems and Computers*, 1, 1998.

[5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *PAMI*, 19(7), July 1997.

[6] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, and D. A. F. Yee Whye Teh, Erik Learned-Miller. Names and faces in the news. *CVPR*, 2004.

[7] Å. Björck and G. H. Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27(123), 1973.

[8] J. Canny. A computational approach to edge detection. *PAMI*, 8(6), 1986.

[9] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[10] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, Inc., New York, 2nd edition, 2000.

[11] M. Everingham and A. Zisserman. Automated person identification in video. *CIVR*, 2004.

[12] M. Everingham and A. Zisserman. Identifying individuals in video by combining generative and discriminative head models. *ICCV*, 2005.

[13] A. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. *ECCV*, 2002.

[14] T. Fromherz, P. Stucki, and M. Bichsel. A survey of face recognition. *MML Technical Report.*, (97.01), 1997.

[15] K. Fukui and O. Yamaguchi. Face recognition using multi-viewpoint patterns for robot vision. *Symp. of Robotics Research*, 2003.

[16] A. S. Georghiades, D. J. Kriegman, and P. N. Belhumeur. Illumination cones for recognition under variable lighting: Faces. *CVPR*, 1998.

[17] P. Hall, D. Marshall, and R. Martin. Merging and splitting eigenspace models. *PAMI*, 2000.

[18] A. Hampapur, R. C. Jain, and T. Weymouth. Production model based digital video segmentation. *Multimedia Tools and Applications*, 1(1), 1995.

[19] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28, 1936.

[20] G. Jaffré and P. Joly. Improvement of a person labelling method using extracted knowledge on costume. *CAIP*, 2005.

[21] K. Lee and D. Kriegman. Online learning of probabilistic appearance manifolds for video-based recognition and tracking. *CVPR*, 2005.

[22] K. Lee, M. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. *CVPR*, 2003.

[23] R. Lienhart. Comparison of automatic shot boundary detection algorithms. *SPIE*, 3656, 1998.

[24] B. Moghaddam and A. Pentland. Principal manifolds and probabilistic subspaces for visual recognition. *PAMI*, 24(6), 2002.

[25] E. Oja. *Subspace Methods of Pattern Recognition*. Research Studies Press and J. Wiley, 1983.

[26] O. Otsuji and Y. Tonomura. Projection detecting filter for video cut detection. *ACM International Conference on Multimedia*, 1993.

[27] J. Sivic, M. Everingham, and A. Zisserman. Person spotting: video shot retrieval for face sets. *CIVR*, 2005.

[28] J. B. Tenenbaum, V. d. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500), 2000.

[29] M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society*, 3(61), 1999.

[30] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1), 1991.

[31] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 57(2), 2004.

[32] H. Wang, S. Z. Li, and Y. Wang. Face recognition under varying lighting conditions using self quotient image. *AFG*, 2004.

[33] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying scene breaks. *ACM Multimedia*, 1995.

[34] H. J. Zhang, A. Kankanhalli, and S. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1(1), 1993.

[35] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4), 2004.