Person spotting: video shot retrieval for face sets

Josef Sivic, Mark Everingham, and Andrew Zisserman

Department of Engineering Science
University of Oxford
http://www.robots.ox.ac.uk/∼vgg

Abstract. Matching people based on their imaged face is hard because of the well known problems of illumination, pose, size and expression variation. Indeed these variations can exceed those due to identity. Fortunately, videos of people have the happy benefit of containing multiple exemplars of each person in a form that can easily be associated automatically using straightforward visual tracking. We describe progress in harnessing these multiple exemplars in order to retrieve humans automatically in videos, given a query face in a shot. There are three areas of interest: (i) the matching of sets of exemplars provided by "tubes" of the spatial-temporal volume; (ii) the description of the face using a spatial orientation field; and, (iii) the structuring of the problem so that retrieval is immediate at run time.
The result is a person retrieval system, able to retrieve a ranked list of shots containing a particular person in the manner of Google. The method has been implemented and tested on two feature length movies.

1 Introduction

The objective of this work is to retrieve shots containing particular people/actors in video material using an imaged face as the query. There are many applications of such a capability, for example: 'intelligent fast-forwards' – where the video jumps to the next scene containing that actor; and retrieval of all the shots containing a particular family member from the thousands of short video sequences captured using a typical modern digital camera.

In this paper we explore person retrieval using (near) frontal faces, though clearly other attributes such as hair or clothing could be added to the feature vector. Face matching is notoriously difficult [4, 5, 8, 18] – even under quite controlled conditions the variation in the imaged face due to lighting, pose, partial occlusion, and expression, can exceed that due to identity. The approach we take is to eschew matching single faces but instead match *sets of faces* for each person, with the representation for each person consisting of a distribution over face exemplars. This approach has been investigated in the literature, e.g. [1, 2, 11, 19]. However, we bring three areas of novelty: first, sets of face exemplars for each person are gathered automatically in shots using tracking (section 2); second, an individual face is represented as a collection of parts [9, 23], with the feature vector describing local spatial orientation fields (section 3.2); third, a face set is represented as a distribution over vector quantized exemplars (section 3.3).

Our aim is to build a description which is largely unaffected by scale, illumination, and pose variations around frontal. Expression variation is then represented by a distribution over exemplars, and this distribution (which in turn becomes a single feature



Fig. 1. Example face detections of the Julia Roberts' character in the movie 'Pretty Woman'. Note that detections are not always perfectly frontal. Note also successful detections despite varying lighting conditions, changing facial expressions and partial occlusions.

vector) is distinctive for each identity. This single feature vector for identity enables efficient retrieval.

We will illustrate the method on the feature length movie 'Pretty Woman' [Marshall, 1990], and use the 'opera' shot shown in figure 4 as our running example. Shots are detected by a standard method of comparing colour histograms in consecutive frames and motion compensated cross-correlation.

In terms of the challenge faced, we have uncontrolled situations with strong lighting changes, occlusions and self-occlusion. Also we can have multiple people in a frame/shot. The entire processing is automatic.

2 Obtaining sets of face exemplars by tracking

In this section we describe the method for associating detected faces within a shot in order to have multiple exemplars covering a person's range and changes of expressions.

Face detection: A frontal face detector [17] is run on every frame of the movie. To achieve a low false positive rate a rather conservative threshold on detection strength is used, at the cost of more false negatives. The face detector is based on AdaBoost with weak classifiers built from local orientation detectors. Example face detections are shown in figure 1. Alternatively, a face detector for video could be used instead [3].

2.1 Associating detected face exemplars temporally

The objective here is to use tracking to associate face detections into *face-tracks* corresponding to the same person within a shot. This is achieved by first running a general purpose region tracker and then associating face detections in different frames based on the region tracks connecting them.

Region tracking: The affine covariant region tracker of [21] is used here. Figure 3(c) shows a typical set of tracked elliptical regions. This tracking algorithm can develop tracks on deforming objects (a face with changing expressions, see figure 2), where the between-frame region deformation can be modelled by an affine geometric transformation plus perturbations, e.g. a region covering an opening mouth. The outcome is that a person's face can be tracked (by the collection of regions on it) through significant pose variations and expression changes, allowing association of possibly distant face detections. The disadvantage of this tracker is the computational cost but this is not such an issue as the tracking is done offline. Note, the face detections themselves are not tracked directly because there may be drop outs lasting over many consecutive frames (e.g. as the person turns towards profile and back to frontal). However, the region tracker survives such changes.

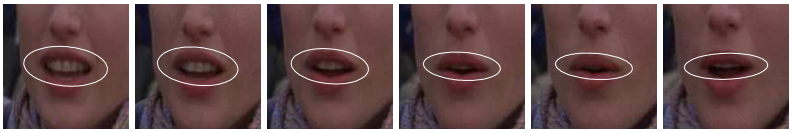


Fig. 2. Detail of a region track covering the deforming mouth whilst the actor speaks. This track extends over 28 frames. The figure shows alternate frames from a subset of the shot.

Connecting face detections using region tracks: A typical shot has tens to hundreds of frames with possibly one or more face detections in each frame. Face detections are usually connected by several region tracks as illustrated in figure 3 – think of this as magnetic flux linking the detected rectangular face regions. We use a single-link agglomerative grouping strategy which gradually merges face detections into larger groups starting from the closest (most connected) detections. We also utilize a temporal exclusion constraint in the clustering, not allowing face tracks arising from distinct face detections in a single frame to be grouped (cf [15]). The temporal exclusion is implemented as a 'cannot link' constraint [10] by setting connectivity to zero for all groups which share the same frame. The merging is run until no two groups can be merged, i.e. have connectivity above certain threshold (five region tracks in this work). This technique is very successful when region tracks between nearby face detections are available. An example of temporal associations of face detections is shown in figure 4(c).

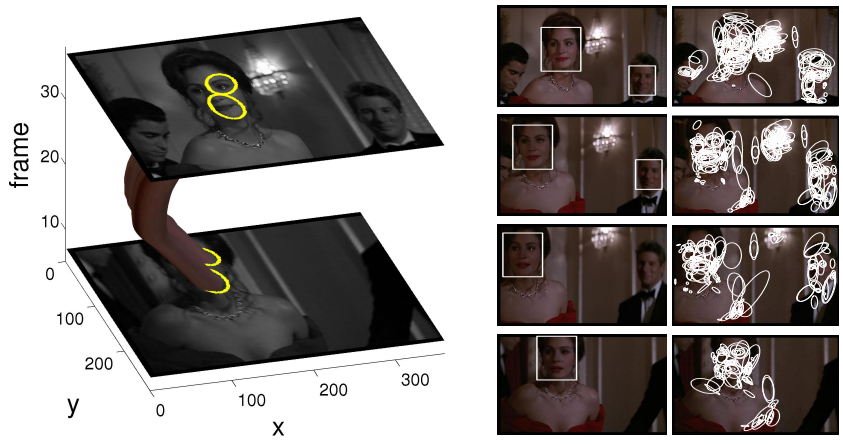
3 Representing and matching sets of face exemplars

In this section we describe our representation of face sets and the matching distance used to compare them. Each face in the (face-track) set is described by a collection of five affinely transformed local spatial orientation fields based around facial features. The entire set is represented as a single distribution over these local feature descriptors. This turns matching sets of exemplars into comparing probability distributions. The following sections describe each of these steps in more detail.

3.1 Facial feature location

The goal here is to localize facial features (left and right eyes, tip of the nose and centre of the mouth) within a face detection. This allows us to place the local face descriptors and affinely deform their support regions to normalize for pose variations. As shown in figure 1 the face feature positions within the face detections vary considerably. This is mainly due to varying head pose and noisy face detector output, e.g. over scale.

Model of feature position and appearance: A probabilistic parts-based "constellation" model [6, 7] of faces is used to model the joint position (shape) and appearance of the facial features. To simplify the model, two assumptions are made: (i) the appearance of each feature is assumed independent of the appearance of other features, and (ii) the appearance of a feature is independent of its position. The position of the facial features is modelled as a single Gaussian with full covariance matrix. In contrast to other work [6, 7] the model does not need to be translation invariant as we expect the



(a) (b) (c)

Fig. 3. (a) Two region tracks as 'tubes' in the video volume between frames 7 and 37 of the 'opera shot' (shown in full in figure 4). The two tracked regions are superimposed in yellow. There are 27 region tracks on the actor's face between the two frames. These 'tubes' allow us to temporally associate face detections in different frames. The 'kink' in the tube arises when the actor moves first left and then right while standing up from a chair. At the same time the camera follows the actor's vertical motion. (b) Four frames from the video volume with face detections superimposed. (c) The same four frames with tracked regions superimposed. In (b) and (c) the frame numbers shown are 7, 17, 27, and 37 (from bottom).

face detector to have approximately normalized the position of the face. To model the appearance of each feature, a rectangular patch of pixels is extracted from the image around the feature and projected onto a subspace determined by principal component analysis (PCA) during the training stage; in this subspace, the appearance is modelled as a mixture of Gaussians, allowing the model to represent distinct appearances such as open and closed eyes. To model the appearance of background (image patches where a facial feature is not present), the same form of model is used as for the facial features, but the position of the patches is assumed uniform.

The parameters of the model are learnt from around 5,000 hand-labelled face images taken from the web. The face detections are scaled to 51×51 pixels and the patches around each feature are between 13×13 (eye) and 21×13 pixels (mouth) in size. A mixture of five Gaussians is used to model the appearance of each part, and the dimensionality of the subspace for each part is chosen by PCA to retain 80% of variance in the training patches.

Locating the facial features using the model: Given the learnt model, the facial features are located by searching for the joint position of the features which maximizes the posterior probability of the feature positions and appearance. To make this search tractable, a few (5) candidate positions are selected for each facial feature by finding local spatial maxima of the appearance term. An example of detected feature points is shown in figure 5(c).

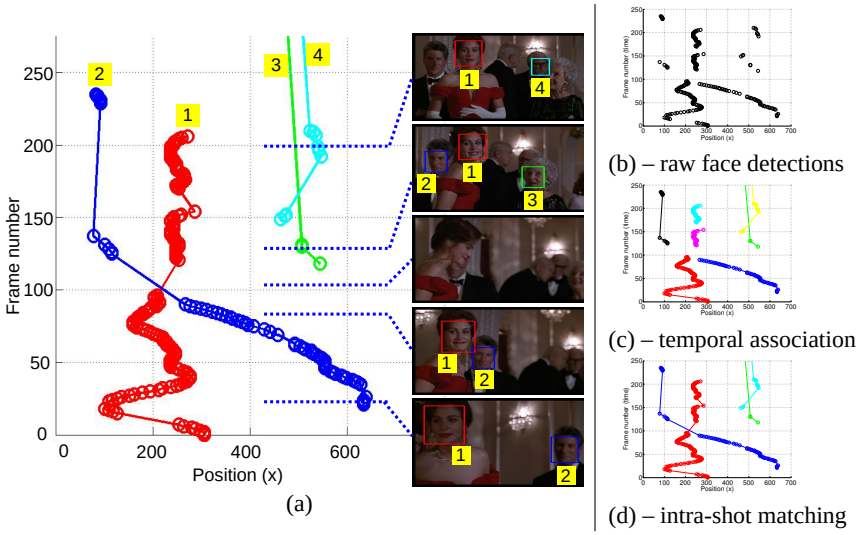


Fig. 4. Associating face detections within a shot. (a) Overview of the first 250 frames of a shot where actors 1 and 2 cross while the camera pans to follows actor 1. Around frame 100 actor 1 turns away from the camera while occluding actor 2. Actors 3 and 4 appear and are detected later in the shot. The circles show positions of face detections. Face detections of the same character are colour coded and connected by lines. The thumbnails on the right show face detections numbered and colour coded according to the actors identity. The raw face detections in the shot (shown in (b)) are connected temporally into face-tracks (shown in (c)). Note some face-tracks are still broken due to occlusion (actor 2) and self-occlusions (actor 1 turns away from the camera). These face-tracks are subsequently linked using intra-shot face-track matching (shown in (d)). The whole process is fully automatic. The temporal association and the intra-shot matching are described in sections 2 and 3.4 respectively.

3.2 Representation of single faces

Each face in the set is represented as a collection of local overlapping parts. Part based approaches to face recognition [23] have been shown [9, 20] to outperform global face description as they cope better with partial occlusions and pose variations. The disadvantage is that the process of facial feature detection is an additional source of possible errors. This becomes a significant factor for more extreme poses [9] where some of the salient components (eyes, mouth, nose) are not visible or extremely distorted. We exclude such cases by limiting ourselves to near frontal poses (by using a frontal face detector).

Our face representation consists of a collection of five overlapping local SIFT descriptors [14] placed at the detected feature locations (eyes, mouth, nose) and also at the mid point between the eyes. The intention is to measure local appearance (e.g. of an eye) independently and also, by the support region overlap, (e.g. of the two eyes) some *joint* feature appearance. Each local SIFT descriptor is an eight bin histogram of image gradient orientations at a spatial 3×3 grid. This gives a 72-dimensional descriptor for each local feature position, i.e. the joint feature for the five regions is a 360-vector. The circular support regions of SIFT descriptors are deformed into ellipses by (the inverse



(a) (b) (c) (d)

Fig. 5. (a) Original frame. (b) Close-up with face detection superimposed. (c) Detected facial features (eyes, nose, mouth). (d) Face is represented as a collection of local affinely deformed spatial orientation fields (SIFT descriptors). The green circles illustrate the location and support region for each of the five SIFT descriptors. Note how the position of the local regions adapts to the slightly rotated pose of the head in this case.

of) an affine geometric transformation which maps feature locations within the face detection into a common canonical frame. This compensates for head pose variation to a certain degree, as is illustrated in figure 5(d). The SIFT descriptor has been shown superior to other local descriptors [16] because it is designed to be invariant to a shift of a few pixels in the feature position, and this localization error often occurs in the facial feature detection process. The SIFT descriptor is also invariant to a linear transformation of image intensity within the (local) support region. This in turn makes the face description robust to more local lighting changes, such as shadows cast by the nose.

In some cases there is a gross error in the face or feature detection process, e.g. one of the features is detected outside of the face. We flag such cases by putting limits on the affine rectifying transformation and do not use those as exemplars.

3.3 Representation of face sets

The goal here is to compactly represent an entire face-track containing a set of (10 to 600) faces. Representing entire face tracks brings a significant data reduction which is very advantageous in the immediate retrieval scenario, i.e. a query face(-track) needs to be compared only to few hundred face-tracks in the entire movie (instead of tens of thousands of single face detections).

Each face is a point, $\mathbf{x}$, in the the 360-dimensional descriptor space (section 3.2) and we assume that faces of a particular character have certain probability density function $f(\mathbf{x})$ over this space. A face track of that person than provides a set of samples from $f(\mathbf{x})$. We use a non-parametric model of $f(\mathbf{x})$ and represent each face track as a histogram over precomputed (vector quantized) face-feature exemplars. A similar representation (over filter responses) has been used in representing texture [13] and recently has been also applied to face recognition [12]. An alternative would be to use a mixture of Gaussians [1]. The vector quantization is performed separately for each local face feature, and is carried out here by k-means clustering computed from about 30,000 faces from the movie 'Pretty woman'. The k-means algorithm is initialized using a greedy distance based clustering which determines the number of clusters K. The final number of face feature clusters is 537, 523, 402, 834 and 675 for the the left eye, the eyes middle, the right eye, the mouth and the nose respectively. Random samples from facial feature clusters are shown in figure 6.



Fig. 6. Quantized facial features. Each row shows ten random samples from one cluster of a vector quantized facial feature (upper: left eye; lower: nose). Each sample is shown as an affinely transformed elliptical region superimposed on an image. The size of the features shown is 3/5 of the actual scale. This reflects the Gaussian weighting of the region support (with decreasing weight towards the boundary) which occurs in the SIFT descriptor computation. Note, there is generalization over pose and illumination.

For each detected face each facial feature is assigned to the nearest cluster centre (e.g. the left eye is coded as one of 537 possibilities). The final representation of a face then is similar to a face identikit where the appearance is composed from the nearest cluster centre for eyes, nose, mouth etc. Each *set* of faces is represented as a (2971 bin) histogram, p, over the cluster centres, where an element p_i of p is the frequency of occurrence of the ith vector quantized face feature cluster. Note that this representation ignores any image ordering or temporal information. The histogram is normalized to sum to one so that it is a probability distribution.

3.4 Matching face sets

The distribution, p, covers expression changes naturally, for example closed and open eyes, or neutral and smiling faces. It is here that we benefit from matching sets of faces: for example with the correct matching measure a shot containing a smiling person can match a shot containing the same person smiling and neutral.

Two histograms, p, q, are compared using the χ^2 statistic as

$$\chi^2(p,q) = \sum_{k=1}^{S} \frac{(p_k - q_k)^2}{(p_k + q_k)}, \tag{1}$$

where S is the number of histogram bins (2971 in our case). $\chi^2(p,q)$ takes value between 0 and 2, being zero when $p = q$.

Matching sets of faces within a shot: The face-tracks developed in section 2 can be broken due to e.g. occlusion by another person or object, or self-occlusion when the actor turns away from the camera. The goal here is to connect such face-tracks. This is beneficial as it gives larger and more representative sets of faces. It is also an easier task than inter-shot matching as the imaging conditions usually do not change dramatically within a shot. The intra-shot matching is achieved by grouping face-tracks with similar distributions, where the distance between distributions is measured by χ^2 as in (1). The grouping is again carried out by the single link clustering algorithm used in section 2. Note that the temporal exclusion constraint is used here again. An example of connecting several face-tracks within a shot is shown in figure 4. The intra-shot matching performance on ground truth data is given in section 4.

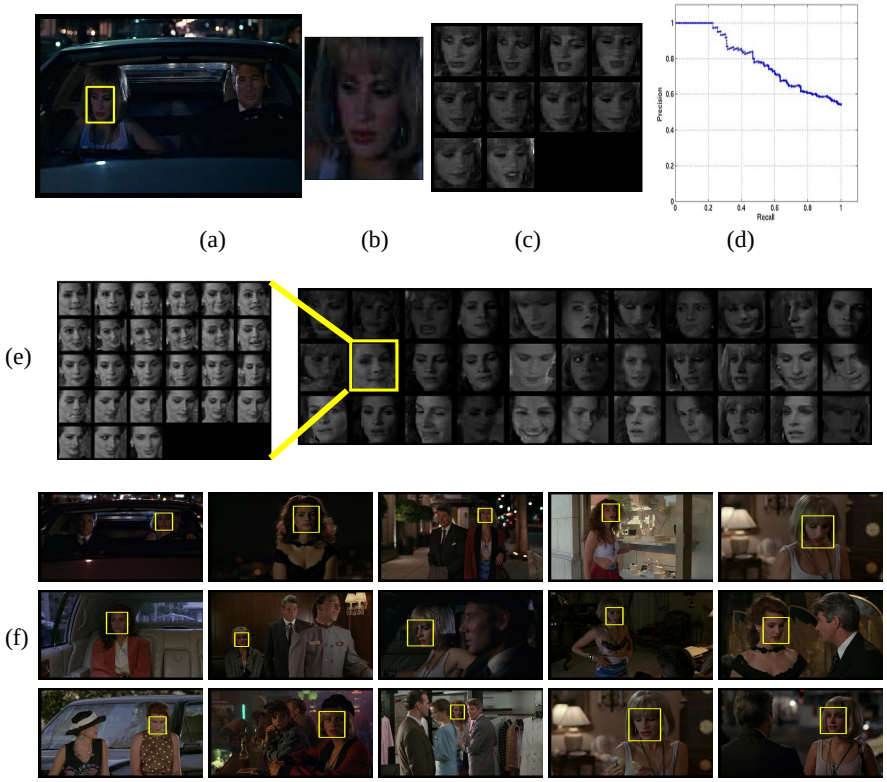


Fig. 7. Example retrieval of the main character from the movie 'Pretty woman'. (a) The query frame with the query face outlined in yellow. (b) close-up of the face. (c) The associated set of 10 face detections in this shot. (d) Precision-Recall curve. (e) Right: the first 33 retrieved face sets shown by the first face detection in each set. Left: example of a retrieved face set. (f) Example face detections from the first 15 retrieved face sets superimposed on the original frames. Note the extent of pose, lighting and expression variation among the retrieved faces. For this character, the number of relevant face-tracks in the ground truth set is 145.

Retrieving sets of faces across shots: At run time a user outlines a face in a frame of the video, and the outlined region tracks are used to 'jump' onto the closest face-track – a set of face detections. The face-tracks within the movie are then ranked according to the χ^2 distance to the query face-track.

4 Results

We have built a person retrieval system for two feature length movies: 'Groundhog Day' and 'Pretty Woman'. Performance of the proposed method is assessed on 337 shots from 'Pretty Woman'. Ground truth on the identity of the detected faces for the seven main characters of the movie is obtained manually for these shots. The entire movie has 1151 shots and 170,000 frames. The 337 ground truth shots contain 38,846 face detections of which 31,846 have successful facial features detection. The temporal

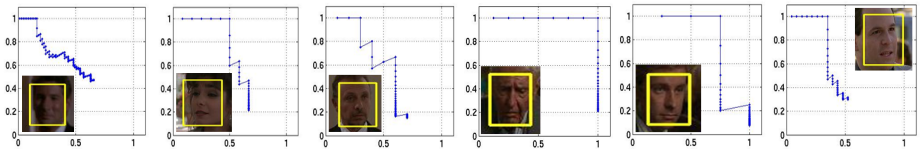


Fig. 8. Retrieval examples of the other six main characters from the movie 'Pretty woman'. The graphs show precision (y-axis) vs. recall (x-axis). Thumbnails show close-ups of the first face detection from each query set. The number of relevant face-tracks in the ground truth set is 67, 12, 10, 8, 4 and 14 for each character respectively (from left).

grouping algorithm of section 2 groups these into 776 face tracks of which 431 have more than 10 face detections.

The main parameters of the overall system are the face detection threshold (which controls the number of false positives and negatives); the size of the support regions for the SIFT descriptors; the distance threshold on SIFT responses determining the number of cluster centres for each face feature region; and the threshold on the χ^2 distance used in face-track intra-shot matching.

Intra-shot matching: The intra shot matching algorithm is applied to the 66 (out of the 337 ground truth) shots that contain more than two face-tracks. The 143 original face tracks from these shots are grouped into 90 face-tracks. The precision is 98.1% (1.9% incorrect merges, i.e. one incorrect merge) and recall is 90.7%, i.e. 9.3% possible merges were missed. Examples of several successful within shot matches on the 'opera' shot are shown in figure 4.

Inter-shot matching: Example retrievals on a ground truth set of 269 face-tracks (after intra-shot matching) of the seven main characters are shown in figures 7 and 8. The query time on this data is about 0.1 second on a 2GHz PC using matlab implementation. Note that the 269 face-tracks contain 25,366 face detections. In some cases the precision recall curve does not reach 100% recall. This is because face tracks with non-overlapping histograms ($\chi^2(p,q) = 2$) are not shown.

5 Conclusions and extensions

We have developed a representation for sets of faces which has the dual advantage that it is distinctive (in terms of inter-person vs. intra-person matching), and also is in a vector form suitable for efficient matching using nearest neighbour or inverted file methods. Using this representation for sets of faces of each person in a shot reduces the matching problem from $O(10^4)$ faces detections over the entire movie, to that of matching a few hundreds probability distributions. This enables immediate retrieval at run time – an extension of the Video Google system [22] to faces.

This work may be improved in several ways, for example: (i) extending the intra-shot matching to clustering over the entire movie (with constraints provided by the exclusion principle); (ii) using the exclusion principle to provide negative exemplars for retrieval at run time.

Acknowledgements

This work was supported by the Mathematical and Physical Sciences Division of the University of Oxford, and the EC PASCAL Network of Excellence, IST-2002-506778.

References

1. O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density divergence. In *Proc. CVPR*, 2005.
2. E. Bart, E. Byvatov, and S. Ullman. View-invariant recognition using corresponding object fragments. In *Proc. ECCV*, pages 152–165, 2004.
3. R. Choudhury, C. Schmid, and K. Mikolajczyk. Face detection and tracking in a video by propagating detection probabilities. *IEEE PAMI*, 25(10):1215–1228, 2003.
4. P. Duygulu and A. Hauptman. What's news, what's not? associating news videos with words. In *Proc. CIVR*, 2004.
5. S. Eickeler, F. Wallhoff, U. Iurgel, and G. Rigoll. Content-Based Indexing of Images and Video Using Face Detection and Recognition Methods. In *ICASSP*, 2001.
6. P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1), 2005.
7. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. CVPR*, 2003.
8. A. Fitzgibbon and A. Zisserman. Joint manifold distance: a new approach to appearance based clustering. In *Proc. CVPR*, Jun 2003.
9. B. Heisele, P. Ho, J. Wu, and T. Poggio. Face recognition: component–based versus global approaches. *CVIU*, 91(1–2):6–21, 2003.
10. D. Klein, S. Kamvar, and C. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *In Proc. Int. Conf. on Machine Learning*, pages 307–314, 2002.
11. V. Krueger and S. Zhou. Examplar-based face recognition from video. In *Proc. ECCV*, 2002.
12. T. Leung. Texton correlation for recognition. In *Proc. ECCV*, 2004.
13. T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43(1):29–44, Jun 2001.
14. D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
15. J.P. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *Proc. ICCV*, 1999.
16. K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. In *CVPR*, 2003.
17. K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *Proc. ECCV*. Springer-Verlag, May 2004.
18. S. Satoh, Y. Nakamura, and T. Kanade. Name-It: Naming and detecting faces in news videos. *IEEE Multimedia*, 6(1):22–35, 1999.
19. G. Shakhnarovich, J. Fisher, and T. Darrell. Face recognition from long-term observations. In *Proc. ECCV*, 2002.
20. G. Shakhnarovich and B. Moghaddam. Face recognition in subspaces. In S.Z. Li and A.K. Jain, editors, *Handbook of face recognition*. Springer, 2004.
21. J. Sivic, F. Schaffalitzky, and A. Zisserman. Object level grouping for video shots. In *Proc. ECCV*. Springer-Verlag, May 2004.
22. J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, Oct 2003.
23. L. Wiskott, J. Fellous, N. Krueger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE PAMI*, 19(7):775–779, 1997.