# "Hello! My name is... Buffy" – Automatic Naming of Characters in TV Video

Mark Everingham, Josef Sivic and Andrew Zisserman
Department of Engineering Science, University of Oxford
{me,josef,az}@robots.ox.ac.uk

**Abstract**

We investigate the problem of automatically labelling appearances of characters in TV or film material. This is tremendously challenging due to the huge variation in imaged appearance of each character and the weakness and ambiguity of available annotation. However, we demonstrate that high precision can be achieved by combining multiple sources of information, both visual and textual. The principal novelties that we introduce are: (i) automatic generation of time stamped character annotation by aligning subtitles and transcripts; (ii) strengthening the supervisory information by identifying when characters are speaking; (iii) using complementary cues of face matching and clothing matching to propose common annotations for face tracks. Results are presented on episodes of the TV series "Buffy the Vampire Slayer".

## 1 Introduction

The objective of this work is to label television or movie footage with the identity of the people present in each frame of the video. As has been noted by previous authors [1, 5] such material is extremely challenging visually as characters exhibit significant variation in their imaged appearance due to changes in scale, pose, lighting, expressions, hair style etc. There are additional problems of poor image quality and motion blur.

We build on previous approaches which have matched frontal faces in order to "discover cast lists" in movies [7] (by clustering the faces) or retrieve shots in a video containing a particular character [1, 17] (starting from a query consisting of one or more images of the actor). The novelty we bring is to employ readily available textual annotation for TV and movie footage, in the form of subtitles and transcripts, to *automatically* assign the correct name to each face image.

Alone, neither the script nor the subtitles contain the required information to label the identity of the people in the video – the subtitles record *what* is said, but not by *whom*, whereas the script records *who* says *what*, but not *when*. However, by automatic alignment of the two sources, it is possible to extract *who* says *what* and *when*. Knowledge that a character is speaking then gives a very weak cue that the person may be visible in the video.

Assigning identities given a combination of faces and textual annotation has similarities to the "Faces in the News" labelling of [2, 4]. Here we are also faced with similar problems of ambiguity: arising from the face detection, e.g. there may be several characters in a frame but not all their faces are detected, or there may be false positive detections; and from the annotation, e.g. in a reaction shot the person speaking (and therefore generating a subtitle) may not be shown. Here, we also exploit other supervisory information



SES-1014
11,087,272

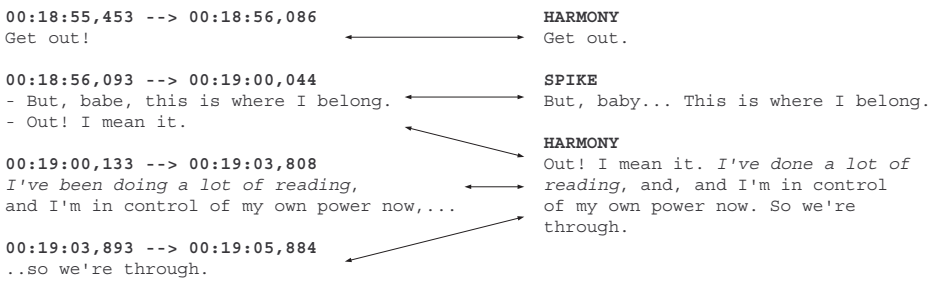



Figure 1: Alignment of the subtitles (left) and script (right). The subtitles contain spoken lines and exact timing information but no identity. The script contains spoken lines and speaker identity but no timing information. Alignment of the spoken text allows subtitles to be tagged with speaker identity. Note that single script lines may be split across subtitles, and lines spoken by several characters merged into a single subtitle. The transcribed text also differs considerably – note the example shown in italics.

that is present in videos (but not in still images) to reduce the ambiguity by identifying visually when a character is speaking.

### 1.1 Outline

As in previous work in this area [1, 7, 17] we adopt an exemplar-based representation of the appearance of each character. Robustness to pose, lighting and expression variation in the description of the facial appearance is obtained by using a parts-based descriptor extracted around detected facial features.

Our method comprises three threads: first, section 2 describes processing of subtitles and script to obtain proposals for the names of the characters in the video; second, section 3 describes the processing of the video to extract face tracks and accompanying descriptors, and to extract descriptors for clothing; and third, section 4 describes the combination of the textual and visual information to assign labels to detected faces in the video. Results of the method are reported in section 5, and conclusions presented in section 6.

The method is illustrated on two 40 minute episodes of the TV serial "Buffy the Vampire Slayer". The episodes are "Real Me" (season 5, episode 2) and "No Place Like Home" (season 5, episode 5). In both cases there is a principal cast of around 11 characters and various others including vampires (who *are* detected by the face detector).

## 2 Subtitle and Script Processing

In order to associate names with characters detected in the video, we use two sources of textual annotation of the video which are easily obtained without further manual interaction: (i) subtitles associated with the video intended for deaf viewers; (ii) a transcript of the spoken lines in the video. Our aim here is to extract an initial prediction of *who* appears in the video, and *when*.

The source video used in the experiments reported here was obtained in DVD format, which includes subtitles stored as bitmaps. The subtitle text and time-stamps (figure 1) were extracted using the publicly available "SubRip" program which uses a simple OCR

(a) Face detections in original frames (b) Localized facial features

Figure 2: Face detection and facial feature localization. Note the low resolution, non-frontal pose and challenging lighting in the example on the right.

algorithm. Most errors in the extracted text were corrected using an off-the-shelf spelling correction algorithm without user intervention.

Scripts for the video were obtained from a fan web-site [18] in HTML format designed for human use. Straightforward text processing was used to extract each component of the script by identifying the HTML tags enclosing each script component. The script contains spoken lines and the identity of the speaker (figure 1), and partial natural text description of the action occurring in the video, but *no* timing information other than the sequence of spoken lines. The processed script gives us one of the pieces of information we require: *who* is speaking; the knowledge that someone is speaking will be used as a cue that they may be visible in the video. However, it lacks information of *when* they are speaking. By aligning the script and subtitles on the basis of the spoken lines, the two sources of information are fused. Figure 1 illustrates the alignment.

A "dynamic time warping" [13] algorithm was used to align the script and subtitles in the presence of inconsistencies such as those in figure 1. The two texts were converted into a string of fixed-case, un-punctuated words. Writing the subtitle text vertically, and the script text horizontally, the task is to find a path from top-left to bottom-right which moves only forward through either text (since sequence is preserved in the script), and makes as few moves as possible through unequal words. The solution is found efficiently using a dynamic programming algorithm. The word-level alignment is then mapped back onto the original subtitle units by a straightforward voting approach.

# 3 Video Processing

This section describes the video processing component of our method. The aim here is to find people in the video and extract descriptors of their appearance which can be used to match the same person across different shots of the video. The task of assigning *names* to each person found is described in section 4.

## 3.1 Face Detection and Tracking

The method proposed here uses face detection as the first stage of processing. A frontal face detector [19] is run on every frame of the video, and to achieve a low false positive rate, a conservative threshold on detection confidence is used. The use of a frontal face detector restricts the video content we can label to frontal faces, but typically gives much greater reliability of detection than is currently obtainable using multi-view face detection [10]. Methods for "person" detection have also been proposed [3, 12] but are typically poorly applicable to TV and movie footage since many shots contain only close-ups or "head and shoulders" views, whereas person detection has concentrated on views of the whole body, for example pedestrians.

Figure 3: Matching characters across shots using clothing appearance. In the two examples shown the face is difficult to match because of the variation in pose, facial expression and motion blur. The strongly coloured clothing allows correct matches to be established in these cases.

A typical episode of a TV series contains around 20,000 detected faces but these arise from just a few hundred "tracks" of a particular character each in a single shot. Discovering the correspondence between faces within each shot reduces the volume of data to be processed, and allows stronger appearance models to be built for each character, since a track provides multiple examples of the character's appearance. Consequently, face tracks are used from here on and define the granularity of the labelling problem.

Face tracks are obtained as follows: for each shot, the Kanade-Lucas-Tomasi tracker [16] is applied. The output is a set of point tracks starting at some frame in the shot and continuing until some later frame. The point tracks are used to establish correspondence between pairs of faces within the shot: for a given pair of faces in different frames, the number of point tracks which pass through both faces is counted, and if this number is large relative to the number of point tracks which are not in common to both faces, a match is declared. This simple tracking procedure is extremely robust and can establish matches between faces where the face has not been continuously detected due to pose variation or expression change. By tracking, the initial set of face detections is reduced to the order of 500 tracks, and short tracks which are most often due to false positive face detections are discarded.

Shot changes are automatically detected using a simple method based on colour histogram difference between consecutive frames. The accuracy of shot detection is not crucial since false positive shot changes merely cause splitting of face tracks, and false negatives are resolved by the tracker.

## 3.2 Facial Feature Localization

The output of the face detector gives an approximate location and scale of the face. In the next stage, the facial features are located in the detected face region. Nine facial features are located: the left and right corners of each eye, the two nostrils and the tip of the nose, and the left and right corners of the mouth. Additional features corresponding to the centres of the eyes, a point between the eyes, and the centre of the mouth, are defined relative to the located features.

To locate the features, a generative model of the feature positions combined with a discriminative model of the feature appearance is applied. The probability distribution over the joint position of the features is modelled using a mixture of Gaussian trees, a Gaussian mixture model in which the covariance of each component is restricted to form a tree structure with each variable dependent on a single "parent" variable. This model is an extension of the single tree proposed in [6] and improves the ability of the model to capture pose variation, with mixture components corresponding approximately to frontal views and views facing somewhat to the left or right. Using tree-structured covariance enables efficient search for the feature positions using distance transform methods [6].

(a) (b) (c)

Figure 4: Examples of speaker ambiguity. In all the cases shown the aligned script proposes a single name, shown above the face detections. (a) Two faces are detected but only one person is speaking. (b) A single face is detected but the speaker is actually missed by the frontal face detector. (c) A 'reaction shot' – the speaker is not visible in the frame. The (correct) output of the speaker detection algorithm is shown below each face detection.

The appearance of each facial feature is assumed independent of the other features and is modelled discriminatively by a feature/non-feature classifier trained using a variation of the AdaBoost algorithm and using the "Haar-like" image features proposed in [19]. A collection of labelled consumer photographs was used to fit the parameters of the model and train the feature classifiers.

Figure 2 shows examples of the face detection and feature localization. The facial features can be located with high reliability in the faces detected by the face detector despite variation in pose, lighting, and facial expression.

### 3.3 Representing Face Appearance

A representation of the face appearance is extracted by computing descriptors of the local appearance of the face around each of the located facial features. Extracting descriptors based on the feature locations [1, 17] gives robustness to pose variation, lighting, and partial occlusion compared to a global face descriptor [8, 15]. Errors may be introduced by incorrect localization of the features, which become more difficult to localize in extremely non-frontal poses, but using a frontal face detector restricts this possibility.

Before extracting descriptors, the face region proposed by the face detector is further geometrically normalized to reduce the scale uncertainty in the detector output and the effect of pose variation, e.g. in-plane rotation. An affine transformation is estimated which transforms the located facial feature points to a canonical set of feature positions. The affine transformation defines an ellipse which is used to geometrically normalize the circular region around each feature point from which local appearance descriptors are extracted. Two descriptors were investigated: (i) the SIFT descriptor [11] computes a histogram of gradient orientation on a coarse spatial grid, aiming to emphasize strong edge features and give some robustness to image deformation. This descriptor has successfully been applied to a face matching task [17]; (ii) a simple pixel-wised descriptor formed by taking the vector of pixels in the elliptical region and normalizing to obtain local photometric invariance. In both cases the descriptor for the face was formed by concatenating the descriptors for each facial feature. The distance between a pair of face descriptors was computed using Euclidean distance. Slightly better results on the naming task were obtained using the simple pixel-based descriptor, which might be attributed to the SIFT descriptor incorporating too much invariance to slight appearance changes relevant for discriminating faces.

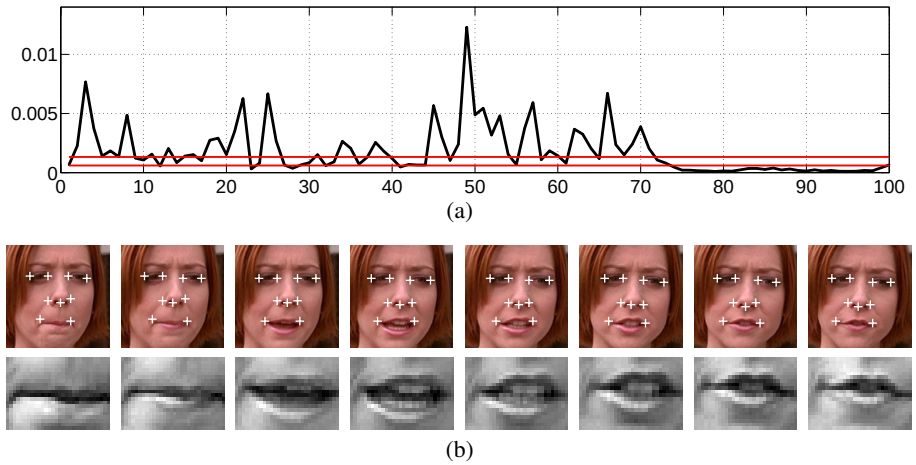



Figure 5: Speaker identification by detecting lip movement. (a) Inter-frame differences for a face track of 101 face detections. The character is speaking between frames 1–70 and remains silent for the rest of the track. The two horizontal lines indicate the 'speaking' (top) and 'non-speaking' (bottom) thresholds respectively. (b) Top row: Extracted face detections with facial feature points overlaid for frames 47–54. Bottom row: Corresponding extracted mouth regions.

### 3.4 Representing Clothing Appearance

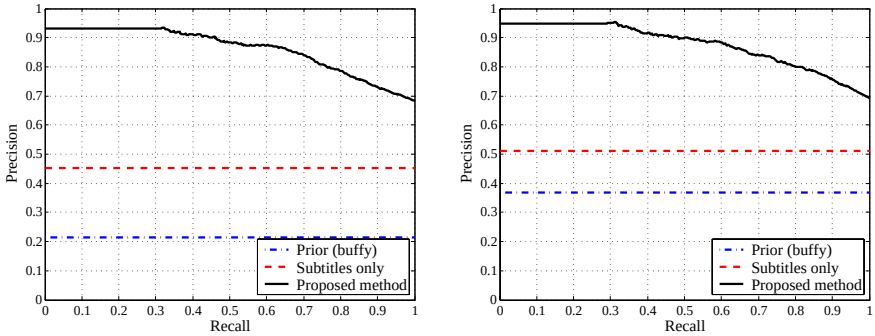
In some cases, matching the appearance of the face is extremely challenging because of different expression, pose, lighting or motion blur. Additional cues to matching identity can be derived by representing the appearance of the clothing [20, 9].

As shown in figure 3, for each face detection a bounding box which is expected to contain the clothing of the corresponding character is predicted relative to the position and scale of the face detection. Within the predicted clothing box a colour histogram is computed as a descriptor of the clothing. We used the YCbCr colour space which has some advantage over RGB in de-correlating the colour components. The distance between a pair of clothing descriptors was computed using the chi-squared measure. Figure 3 shows examples which are challenging to match based on face appearance alone, but which can be matched correctly using clothing.

Of course, while the face of a character can be considered something unique to that character and in some sense constant (though note that characters in this TV series who are vampires change their facial appearance considerably), a character may, and does, change their clothing within an episode. This means that while similar clothing appearance suggests the same character, observing different clothing does not necessarily imply a different character. As described in section 5, we found that a straightforward weighting of the clothing appearance relative to the face appearance proved effective.

### 3.5 Speaker Detection

The combined subtitle and script annotation (section 2) proposes one or more possible speaker names for each frame of the video containing some speech. This annotation is




(a) Episode 05-02 (b) Episode 05-05

Figure 6: Precision/recall curves for two episodes. Recall is the proportion of face tracks which are assigned labels by the proposed method at a given confidence level, and precision the proportion of correctly labelled tracks. The graphs show the performance of the proposed method and two baseline methods using the subtitles to propose names for each face track (see text for details).

| | **Episode 05-02** | | | | **Episode 05-05** | | | |
|---|---|---|---|---|---|---|---|---|
| Recall: | 60% | 80% | 90% | 100% | 60% | 80% | 90% | 100% |
| **Proposed method** | 87.5 | 78.6 | 72.9 | 68.2 | 88.5 | 80.1 | 75.6 | 69.2 |
| **Subtitles only** | | | | 45.2 | | | | 45.5 |
| **Prior (Buffy)** | | | | 21.3 | | | | 36.9 |

Table 1: Quantitative precision results at different levels of recall. The baseline methods do not provide a means for ranking, so only the overall accuracy is reported.

still extremely ambiguous: (i) there might be several detected faces present in the frame and we do not know which one is speaking; (ii) even in the case of a single face detection in the frame the actual speaking person might be undetected by the frontal face detector or the frame might be part of a 'reaction shot' where the speaker is not present in the frame at all. These ambiguities are illustrated in figure 4.

The goal here is to resolve these ambiguities by identifying the speaker using visual information [14]. This is achieved by finding face detections with significant lip motion. A rectangular mouth region within each face detection is identified using the located mouth corners (section 3.2) and mean squared difference of the pixel values within the region is computed between the current and previous frame. To achieve translation invariance the difference is computed over a search region around the mouth region in the current frame and the minimum taken. Two thresholds on the difference are set to classify face detections into 'speaking' (difference above a high threshold), 'non-speaking' (difference below a low threshold) and 'refuse to predict' (difference between the thresholds). This simple lip motion detection algorithm works well in practice as illustrated in figure 5.

Proposed identities for face detections which are classified as speaking are accumulated into a single set of identities for the entire face track. In many cases this set contains just a single identity, but there are also cases with multiple identities, due to merging of script lines into a single subtitle and imprecise timing of the subtitles relative to the video.

# 4 Classification by Exemplar Sets

The combination of subtitle/script alignment and speaker detection gives a number of face tracks for which the proposed identity is correct with high probability. Tracks for which a single identity is proposed are treated as exemplars with which to label the other tracks which have no, or uncertain, proposed identity.

Each unlabelled face track $F$ is represented as a set of face descriptors and clothing descriptors $\{\mathbf{f},\mathbf{c}\}$. Exemplar sets $\lambda_i$ have the same representation but are associated with a particular name. For a given track $F$, the quasi-likelihood that the face corresponds to a particular name $\lambda_i$ is defined thus:

$$p(F|\lambda_i) = \frac{1}{Z}\exp\left\{-\frac{d_f(F,\lambda_i)^2}{2\sigma_f^2}\right\}\exp\left\{-\frac{d_c(F,\lambda_i)^2}{2\sigma_c^2}\right\} \tag{1}$$

where the face distance $d_f(F,\lambda_i)$ is defined as the minimum distance between the descriptors in $F$ and in the exemplar tracks $\lambda_i$:

$$d_f(F,\lambda_i) = \min_{\mathbf{f}_j\in F}\min_{\mathbf{f}_k\in\lambda_i}||\mathbf{f}_j-\mathbf{f}_k|| \tag{2}$$

and the clothing distance $d_c(F,\lambda_i)$ is similarly defined. The quasi-likelihoods for each name $\lambda_i$ are combined to obtain a posterior probability of the name by assuming equal priors on the names and applying Bayes' rule:

$$P(\lambda_i|F) = \frac{p(F|\lambda_i)}{\sum_j p(F|\lambda_j)} \tag{3}$$

Taking $\lambda_i$ for which the posterior $P(\lambda_i|F)$ is maximal assigns a name to the face. By *thresholding* the posterior, a "refusal to predict" mechanism is implemented – faces for which the certainty of naming does not reach some threshold will be left unlabelled; this decreases the recall of the method but improves the accuracy of the labelled tracks. In section 5 the resulting precision/recall tradeoff is reported.

# 5 Experimental Results

The proposed method was applied to two episodes of "Buffy the Vampire Slayer". Episode 05-02 contains 62,157 frames in which 25,277 faces were detected, forming 516 face tracks. Episode 05-05 contains 64,083 frames, 24,170 faces, and 477 face tracks. The parameters of the speaking detection and weighting terms in the quasi-likelihood (equation 1) were coarsely tuned on episode 05-02 and all parameters were left unchanged for episode 05-05. The speaking detection labels around 25% of face tracks with around 90% accuracy. *No* manual annotation of any data was performed other than to evaluate the method (ground truth label for each face track).

Figure 6 shows precision/recall curves for the proposed method, and quantitative results at several levels of recall are shown in table 1. The term "recall" is used here to mean the proportion of tracks which are assigned a name after applying the "refusal to predict" mechanism (section 4), and precision is the proportion of correctly labelled tracks. Two baseline methods were compared to the proposed method: (i) "Prior" – label all tracks with the name which occurs most often in the script (Buffy); (ii) "Subtitles only" – label any tracks with proposed names from the script (not using speaker identification) as one of the proposed names, breaking ties by the prior probability of the name occurring in

Figure 7: Examples of correct detection and naming throughout episode 05-02.

the script; label tracks with no proposed names as the most frequently occurring name (Buffy).

As expected, the distribution over the people appearing in the video is far from uniform – labelling all face tracks "Buffy" gives correct results 21.9% of the time in episode 05-02 and 36.9% of the time in episode 05-05. The cues from the subtitles increase this accuracy to around 45% in each episode, revealing the relative weakness of this cue to identity. Using our proposed method, if we are forced to assign a name to *all* face tracks, the accuracy obtained is around 69% in both episodes. Requiring only 80% of tracks to be labelled increases the accuracy to around 80%. We consider these results extremely promising given the challenging nature of this data. Figure 7 shows some examples of correctly detected and named faces.

# 6 Conclusions

We have proposed methods for incorporating textual and visual information to automatically name characters in TV or movies and demonstrated promising results obtained without any supervision beyond the readily available annotation.

The detection method and appearance models used here could be improved, for example by bootstrapping person-specific detectors [5] from the automatically-obtained exemplars in order to deal with significantly non-frontal poses, and including other weak cues such as hair or eye colour. Further use of tracking, for example using a specific body tracker rather than a generic point tracker, could propagate detections to frames in which detection based on the face is difficult.

In the current approach there is no mechanism for error correction, for example it might be possible to overrule errors in the annotation of the exemplars by strong similarities between a set of face tracks or clothing. A promising approach to this problem which we are pursuing is to cast the labelling problem as one of solving an MRF over the graph of connections generated by track and clothing similarities. However, this requires more "long-range" interactions between the tracks to be generated in order to build a richer, more connected graph structure.

**Acknowledgements.** This work was supported by EC project CLASS and an EPSRC Platform grant. This publication only reflects the authors' views.

# References

[1] O. Arandjelovic and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *Proc. CVPR*, pages 860–867, 2005.

[2] T. Berg, A. Berg, J. Edwards, M. Maire, R. White, Y. W. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *Proc. CVPR*, pages 848–854, 2004.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, pages 886–893, 2005.

[4] P. Duygulu and A. Hauptmann. What's news, what's not? associating news videos with words. In *Proc. CIVR*, pages 132–140, 2004.

[5] M. Everingham and A. Zisserman. Identifying individuals in video by combining generative and discriminative head models. In *Proc. ICCV*, pages 1103–1110, 2005.

[6] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1), 2005.

[7] A. W. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *Proc. ECCV*, volume 3, pages 304–320, 2002.

[8] B. Heisele, P. Ho, J. Wu, and T. Poggio. Face recognition: component–based versus global approaches. *CVIU*, 91(1–2):6–21, 2003.

[9] G. Jaffre and P. Joly. Costume: A new feature for automatic video content indexing. In *Proc. RIAO*, 2004.

[10] S. Z. Li and Z. Q. Zhang. Floatboost learning and statistical face detection. *IEEE PAMI*, 26(9), 2004.

[11] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, pages 1150–1157, 1999.

[12] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *Proc. ECCV*, volume 1, pages 69–82, 2004.

[13] C. S. Myers and L. R. Rabiner. A comparative study of several dynamic time-warping algorithms for connected word recognition. *The Bell System Technical Journal*, 60(7):1389–1409, 1981.

[14] K. Saenko, K. Livescu, M. Siracusa, K. Wilson, J. Glass, and T. Darrell. Visual speech recognition with loosely synchronized feature streams. In *Proc. ICCV*, pages II: 1424–1431, 2005.

[15] G. Shakhnarovich and B. Moghaddam. Face recognition in subspaces. In S.Z. Li and A.K. Jain, editors, *Handbook of face recognition*. Springer, 2004.

[16] J. Shi and C. Tomasi. Good features to track. In *Proc. CVPR*, pages 593–600, 1994.

[17] J. Sivic, M. Everingham, and A. Zisserman. Person spotting: video shot retrieval for face sets. In *Proc. CIVR*, pages 226–236, 2005.

[18] `http://uk.geocities.com/slayermagic/`.

[19] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR*, pages 511–518, 2001.

[20] L. Zhang, L. Chen, M. Li, and H. Zhang. Automated annotation of human faces in family albums. In *Proc. ACM MULTIMEDIA*, pages 355–358, 2003.