Automated Person Identification in Video

Mark Everingham and Andrew Zisserman

Visual Geometry Group, Department of Engineering Science, University of Oxford
{me|az}@robots.ox.ac.uk

Abstract. We describe progress in the automatic detection and identification of humans in video, given a minimal number of labelled faces as training data. This is an extremely challenging problem due to the many sources of variation in a person's imaged appearance: pose variation, scale, illumination, expression, partial occlusion, motion blur, etc.
The method we have developed combines approaches from computer vision, for detection and pose estimation, with those from machine learning for classification. We show that the identity of a target face can be determined by first proposing faces with similar pose, and then classifying the target face as one of the proposed faces or not. Faces at poses differing from those of the training data are rendered using a coarse 3-D model with multiple texture maps. Furthermore, the texture maps of the model can be automatically updated as new poses and expressions are detected. We demonstrate results of detecting three characters in a TV situation comedy.

1 Introduction

The objective of this paper is to annotate video with the identities, location within the frame, and pose, of specific people. This requires both detection and recognition of the individuals. Our motivation for this is two fold: firstly, we want to annotate video material, such as situation comedies and feature films, with the principal characters as a first step towards producing a visual description of shots suitable for blind people, e.g. "character A looks at character B and moves towards him". Secondly, we want to add index keys to each frame/shot so that the video is searchable. This enables new functionality such as "intelligent fast forwards", where the video can be chosen to play only shots containing a specific character; and character-based search, where shots containing a set of characters (or not containing certain characters) can easily be obtained.

The methods we are developing are suitable for any video material, including news footage and home videos, but here we present results on detecting characters in an episode of the BBC situation comedy 'Fawlty Towers'. Since some shots are close-ups or contain only face and upper body, we concentrate on detecting and recognizing the face rather than the whole body.

The task is a staggeringly difficult one. We must cope with large changes in scale: faces vary in size from 200 pixels to as little as 15 pixels (i.e. very low resolution), partial occlusion, varying lighting, poor image quality, and motion blur. In a typical episode the face of a principal character (Basil) appears frontal

in one third of the frames, in profile in one third, and from behind in the other third, so we have to deal with a much greater range of pose than is usual in face detection.

Previous approaches to character identification have concentrated on frontal faces [7, 9]. This is for two reasons: (i) face detection is now quite mature and successful for frontal faces [10, 15, 16] (both in terms of false positive/ false negative performance, and also in efficiency); and (ii) because most recognition methods are developed for frontal faces [17]. For example, image-based 'eigenface' or 'Fisherface' [2] approaches are successful for registered frontal faces with stable illumination. Detection of profile faces [15] or arbitrary pose [10, 12] has not yet reached the same level of performance. This is principally because in the case of frontal faces pattern matching methods can be used to classify an image region through a fixed mask as a face or non-face, since there are sufficient distinctive internal features visible (eyes, mouth, etc.). In the case of profiles there are fewer distinctive features, and the silhouette varies. Consequently, simple fixed regions of interest include background, and the resulting learning problem is then much more difficult.

The approach we have developed is closest in spirit to the pose and multiple view based approaches of [3, 5, 13]. Suppose that we have identified a face region in a target frame, and our task is now to decide if this is the face of one of the characters in our training data. This is a matching problem, and in the case of faces we must account for three principal 'dimensions' of variation: pose change, illumination change, and expression change. Conceptually we divide this problem into two parts:

1. Pose based rendering: a set of candidate faces is proposed by rendering faces from the training data at the same pose as the target face, see figure 4. The candidate faces will typically contain several examples of the correct face with a range of expressions, as well as examples of other characters. This largely eliminates the pose variation, and we have reduced the problem to matching over expression and illumination change.

2. Classification: a matching decision is made amongst the proposed faces. The outcome is a match with one of the faces, or a non-match (if the target face is not one of the learnt characters). This requires a matching measure which is tolerant to small changes of expression, and largely invariant to illumination conditions.

2 Approach

In this section we describe the two stages of the algorithm: learning face models, and recognition of faces in target frames. The overall recognition approach consists of three steps: (i) detecting candidate face regions in the target frame, (ii) determining the pose of the target face and proposing candidate faces at that pose, and (iii) classification.

a) Input image b) Skin probability c) Candidate face regions

Fig. 1. Candidate face region detection using skin colour model and multi-scale blob detector. Darker grey levels in (b) represent higher probability. Concentric circles in (c) show the scale uncertainty in the detections. Note there are several false positives due to non-face skin regions, and non-skin regions of similar colour. These false positives will be removed by subsequent verification.

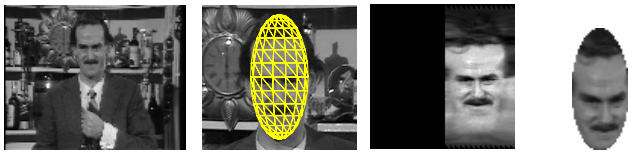
2.1 Candidate face region detection

The first step in detection is to propose candidate face regions in an image for further processing. Requirements are that the algorithm proposes all faces in the image as candidates across a wide range of scale and pose. We desire to have a relatively small number of false positive (non-face) responses from the algorithm, since processing false detections incurs computational cost, but we can cope with some false positives since candidate regions will be subsequently verified. This differs somewhat from the isolated problem of face detection [15, 16], where detections are not subject to additional verification.

We take advantage of working with colour video and use a skin colour detector to propose probable face regions. The probability distribution over the colour of skin pixels in RGB space is modelled as a single Gaussian with full covariance. A corresponding Gaussian distribution with large variance is estimated for 'background' pixels, and Bayes theorem is applied to obtain an image of the posterior probability that each pixel is skin. Skin blob detection is performed over an image pyramid by applying a Difference of Gaussians (DOG) operator [14] to the skin probability image at each level. A face region is declared at local maxima in the DOG response with positive response above threshold, and corresponding high skin probability. The approximate scale of the face is obtained from the pyramid level. Figure 1 shows an example image, skin probability, and detected candidate face regions.

2.2 Pose based face rendering

We require a method of rendering faces at poses different from those in the training material. The approach used here is to combine coarse 3-D geometry with multiple texture maps. The model has two parts: a global 3-D geometric model of the head, and a set of visual 'aspects' which define appearance over local regions of pose space. The shape of the head is modelled simply as an ellipsoid, the parameters of which are fitted to a single training image of the person. Figure 2a shows a training image for the 'Basil' model, and Figure 2b the ellipsoid model overlaid. The aims of using a 3-D model for the head are two fold:



a) Training image b) Ellipsoid c) Texture d) Novel view

Fig. 2. Ellipsoid head model. The triangulation shown (a) is coarser than that used, to aid visibility. The blank area of the texture map is the back of the head, which has not yet been observed.

1. Extrapolation: The 3-D model allows us to extrapolate some way from a single view of the person and propose how the person looks in nearby poses. The single training image is back-projected onto the ellipsoid to give a texture map (Figure 2c), then a new view of the head in a different pose can then be rendered by transforming the ellipsoid and projecting the texture map back into the image. Figure 2d shows an example: for poses near to the one from which the texture map was obtained, fairly accurate images can be rendered. Because the ellipsoid geometry only approximates the head shape, the realism of the rendered views degrades as the pose change increases, principally because the ellipsoid does not predict self occlusions (such as the eye being occluded as the face looks down). However, it will be seen that combining a simple shape model with *multiple* texture maps enables accurate rendering of many poses. By contrast, an accurate 3-D model could extrapolate further from a single view, but it is difficult to obtain such an accurate model, and an inaccurate but non-smooth model can introduce many artifacts that we wish to avoid. Ellipsoids [1] and close relatives (superquadrics [11], tapered ellipsoids [13]) have been applied successfully to head tracking by several authors.

2. Pose space: The second reason for the 3-D model is that it provides a global reference frame against which any image of the face can be aligned. Initially, having seen just a single image of the face, we have a good idea of the appearance in only a narrow range of poses, and with fixed facial expression. Estimating the pose of a new image and verifying the identity of the person allows a new image to be classified as: (i) close in pose and appearance to an already seen image, (ii) in a pose far from one observed up to this point, or (iii) in a known pose but with differing appearance (facial expression). In the latter two cases the algorithm considers expanding the model by adding additional texture maps, positioning them appropriately in pose space. This allows the model to be improved without manual supervision.

2.3 Pose estimation

Given a candidate face region in the image, the pose of the face is recovered by search in the joint pose/appearance space, proposing the appearance of the face and comparing against the target image. The pose is parameterized as a 6-D vector $\boldsymbol{p} = \langle \theta, \phi, \psi, \sigma, \tau_x, \tau_y \rangle$ corresponding to rotation, scale, and 2-D translation

Fig. 3. Pose estimation (best viewed in colour). Top rows show original image, middle rows show ellipsoid overlaid at the estimated pose, bottom rows show overlaid model rendered at the estimated pose.

in the image. Rotation is specified by azimuth θ, elevation ϕ, and in-plane rotation ψ. This parameterization allows reasonable bounds to be specified easily. A candidate face region provides an initial estimate of scale $\tilde{\sigma}$, up to the scale step between pyramid levels, and translation $\langle \tilde{\tau}_x, \tilde{\tau}_y \rangle$ (the centre of the candidate region). The task is to find the pose parameters $\hat{\boldsymbol{p}}$ which maximize the similarity between the rendered view $R(\boldsymbol{p}, \mu)$ and the target image I. Normalized cross-correlation (NCC), masked by the silhouette of the rendered view, is used as the similarity measure:

$$\hat{\boldsymbol{p}} = \arg\max_{\boldsymbol{p}} \left[\max_{\mu \in \{\mu_{\boldsymbol{p}}\}} \mathrm{NCC}\left(I, R(\boldsymbol{p}, \mu)\right) \right] \tag{1}$$

For a given pose, *multiple* appearances $R(\boldsymbol{p}, \mu)$ are proposed by selecting a subset of the texture maps $\{\mu_{\boldsymbol{p}}\}$ which are (i) close to the current pose, and (ii) varying in expression. This is done by first finding the texture map which has pose $\boldsymbol{q}$ closest to the current estimate $\boldsymbol{p}$, then selecting all texture maps with pose close to $\boldsymbol{q}$ (which represent different facial expressions). Distance between poses is computed by the dot product between a front-facing vector normal to

Fig. 4. Face classification based on multiple appearance proposals. The leftmost image is the target, with rows showing proposals rendered from the Manuel, Basil and Sybil models. The task is to decide which proposal to accept, or to reject all.

the ellipsoid, so that in-plane rotation about the frontal view does not influence the distance. Using this 'nearest neighbour plus siblings' approach to selecting texture maps allows the algorithm to consider texture maps corresponding both to close poses and varying facial expression. Numerical optimization is carried out using the coordinate descent algorithm of [8]. Figure 3 shows examples of pose estimation. Additional examples can be seen in Figure 4, discussed below.

2.4 Classification

Given an estimated pose, a set of images is proposed by the models of each person. Figure 4 shows an example, with each person model attempting to reproduce the leftmost image, of Basil. Note that the proposals here have the same pose but vary in facial expression. The aim now is to obtain a representation of the face image suitable for person classification, capturing the essential structure of the facial appearance but allowing for small local misalignments between the original and rendered images due to factors such as the approximation of the face shape as ellipsoidal. Using this representation, one of the proposed images may be accepted as a match, yielding classification of the person, or all may be discounted, in the case of a non-face region, or person other than those modelled.

Use of 'edges' rather than raw grey levels for emphasizing salient image structure has been proposed in many contexts [14] and an edge-based descriptor is used here, proposed most recently for comparing optical flow fields [6]. For an image I, the image gradients I_x, I_y are computed, and half-wave rectified to form four non-negative channels $I_x^+, I_x^-, I_y^+, I_y^-$. Each channel is then blurred with a Gaussian to give some robustness to local image deformations, and the descriptor for the image $D(I)$ is formed by normalizing and concatenating the four channels. The non-negativity and relative sparseness of signal in each channel allows the channels to be blurred without destroying orientation information or edges by cancelling positive and negative gradients. The width of the Gaussian is set proportional to the scale of the face in the image.

When comparing descriptors for a target image I and a rendered view of the ellipsoid $R(\boldsymbol{p}, \mu)$, the rendered view is overlaid on the target image (in the man-

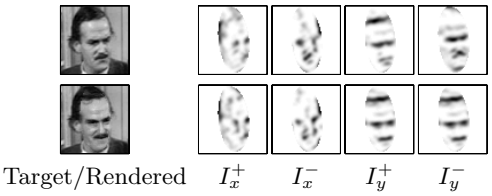


Fig. 5. Gradient descriptor for target (top) and rendered (bottom) images. Darker grey levels represent larger values. The similarity measure here is 0.98, a close match.

Fig. 6. Model update by tracking. A colour tracker successfully tracks the face over large pose variation and is used to validate proposed updates to the model.

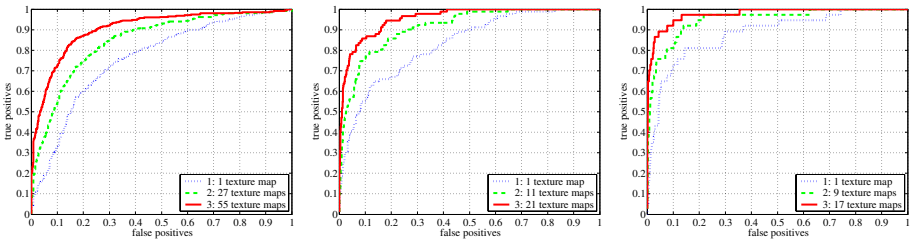
ner of Figure 4) before computing image gradients in order to avoid introducing spurious edges due to the ellipsoid boundary. Similarity between the corresponding descriptors $D(I)$ and $D(R(\boldsymbol{p}, \mu))$ is obtained by correlation, considering only pixels within the ellipsoid mask. Figure 5 shows example descriptors for target and rendered images.

2.5 Model learning

The supervision required for learning the face model is minimal: a face for each character is identified in one frame, and the ellipsoid model fitted. Additional training is automatic, as will now be described.

Having computed the similarity (section 2.4) between a set of face candidates and a particular person, a decision is made as to which detections to add to the model as new texture maps, enabling the model to cope with wider variations in expression and pose.

A low threshold on similarity t_l is defined, above which we are confident that a detection matches a particular person. Three cases then follow: (i) if the similarity of a match is above a second higher threshold ($t > t_h > t_l$) and the pose is close to one already seen, then the image need not be added to the model. (ii) If however the match is certain ($t > t_h$) and the pose is far from one already seen, the image is added to the model so that the range of pose covered is expanded. Finally (iii), less certain matches ($t_l < t < t_h$) which lie close to an existing pose are validated by tracking. These would typically represent unseen facial expressions. To validate such matches, temporal coherence of the video is exploited: a tracker is run from frames with certain matches, ending at the candidate frame. The tracker used is a colour version of a deformable region tracker [4]. If the position of the tracked region agrees with the detected face, then the model is updated. Figure 6 shows an example of successful tracking over wide pose variation.



a) Basil b) Manuel c) Sybil

Fig. 7. ROC curves for three characters in 1,500 key frames. Successful identification requires correct detection, pose estimation, and recognition. In all cases, unsupervised model update improves the accuracy of the model.

3 Experimental results

The algorithm was tested on 1,500 key-frames taken one per second from the episode 'A Touch of Class' of the sitcom 'Fawlty Towers'. We evaluated detection of three of the main characters: Basil, Sybil and Manuel. The task was to detect the frames containing each character, and identify the image position and pose of the face correctly. Correctness was measured by the distance to ground truth points marked on the eyes, nose and ears according to pose, requiring distance of all predicted points to be less than 0.3 of the inter-ocular distance. Corresponding points for the model (for testing purposes only) were obtained by back-projecting the ground truth points onto the ellipsoid during training and model update. Pose of the ground truth faces in the video covers poses of around +/-60° azimuth, +/-30° elevation and +/-45° in-plane rotation. Faces vary in scale from 15 to 200 pixels. The values of the thresholds t_l and t_h were determined from a validation set and kept fixed throughout the experiments.

Figure 7 shows ROC curves for each of the three characters. Note that we treat the problem as one of detection rather than 1-of-m classification since we do not know *a priori* all the characters in the video. For each character, curves are shown for the initial model and two runs of the model update procedure. The number of texture maps after model update varied for each character, due to the varying number of frames in which the character appears and differences in pose variation between characters, and is shown in the legend. The graphs show clear improvement in the accuracy of the model after update, for example in the Basil model the equal error rate decreases from 30% to 15% after two rounds of update. At this stage, characters can be detected in 75–95% of frames at a false positive rate of 10%. These results are extremely promising given the difficulty of the task. It is interesting to observe that the performance on Sybil is notably better than the other characters; this is the 'moustache problem' - the moustache is a strong visual feature shared between Basil and Manuel, and indeed three other secondary characters in the episode, which gives much scope for confusion.

4 Discussion

We have presented methods for detecting and identifying characters in video across wide variations in pose and appearance by combining a simple 3-D model with view-dependent texture mapping. Placing the views of the face in a common reference frame allows more efficient search than possible with an unorganized collection of images, and provides a basis for automatic model update. Use of a simple 3-D model rather than a detailed face model [3] avoids introducing severe rendering artifacts due to incorrect modelling of self-occlusion, and multiple texture maps allow facial expressions to be modelled, which is challenging for 3-D models with a fixed texture map.

Acknowledgements Thanks to EC Project CogViSys for funding.

References

[1] S. Basu, I. Essa, and A. Pentland. Motion regularization for model-based head tracking. In *Proc. ICPR*, pages 611–616, 1996.

[2] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE PAMI*, 19(7):711–720, 1997.

[3] V. Blanz, S. Romdhani, and T. Vetter. Face identification across different poses and illumination with a 3D morphable model. In *Proc. AFGR*, 2002.

[4] Y. Chen, T. Huang, and Y. Rui. Optimal radial contour tracking by dynamic programming. In *Proc. ICIP*, 2001.

[5] T. F. Cootes, K. Walker, and C. J. Taylor. View-based active appearance models. In *Proc. AFGR*, pages 227–232, 2000.

[6] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *Proc. ICCV*, 2003.

[7] S. Eickeler, F. Wallhoff, U. Iurgel, and G. Rigoll. Content-Based Indexing of Images and Video Using Face Detection and Recognition Methods. In *Proc. ICASSP*, 2001.

[8] V. Ferrari, T. Tuytelaars, and L. Van Gool. Wide-baseline multiple-view correspondences. In *Proc. CVPR*, pages 718–725, 2003.

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

[10] B. Heisele, T. Serre, M. Pontil, and T. Poggio. Component-based face detection. In *Proc. CVPR*, pages 657–662, 2001.

[11] N. Krahnstoever and R. Sharma. Appearance management and cue fusion for 3D model-based tracking. In *Proc. CVPR*, pages 249–254, June 2003.

[12] S. Z Li, L. Zhu, Z. Q. Zhang, A. Blake, H. J. Zhang, and H. Shum. Statistical learning of multi-view face detection. In *Proc. ECCV*, 2002.

[13] M. C. Lincoln and A. F. Clark. Pose-independent face identification from video sequences. In *Proc. BMVC.*, 2001.

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

[15] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Proc. CVPR*, 2000.

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

[17] W. Zhao, R. Challappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35:399–458, 2003.