Improved 3D Model Search for Facial Feature Location and Pose Estimation in 2D images

Angela Caunce
angela.caunce@manchester.ac.uk
Chris Taylor
chris.taylor@manchester.ac.uk
Tim Cootes
tim.cootes@manchester.ac.uk

Imaging Science
& Biomedical Engineering
The University of Manchester
Manchester M13 9PT, UK

This is the original version of this paper. A revised version is available at the authors website, with corrections to Figure 7

Abstract

This paper tackles the problem of accurately matching a 3D deformable face model to sequences of images in challenging real-world scenarios with large amounts of head movement, occlusion, and difficult lighting conditions. A baseline system involves searching with a set of view-dependent local patches to locate image features, and using these to update the face shape model parameters. We show here two modifications that lead to improvements in performance and can be applied in other similar systems. These are: explicitly searching for occluding boundaries, which prevents the model from rotating rather than changing shape; and a simple method for weighting the relative importance of each located match for model fit. We demonstrate the improvements on both standard test sets and on a series of difficult in-car driver videos, showing more accurate matching and fewer search failures.

1 Introduction

This paper describes improvements to a 3D model matching algorithm capable of tracking the face and estimating pose in difficult videos like that shown in Figure 1. Such videos contain significant head movement, dramatic lighting changes, and frequent occlusions, making reliable facial feature tracking a challenge.

Previously, in [1], a 3D matching approach was presented that performed at least as well as an established 2D system [2] on large datasets of near frontal images [3, 4], and surpassed it on large rotations. The system uses a sparse 3D shape model, with a matching algorithm similar to the Active Shape Model [5], in which model parameters are updated based on the best matched locations of local patches. A key component was that the appearance of each patch depended on the orientation relative to the camera. Although good results were obtained, the system still failed to match accurately to a proportion of the data.

In this paper we demonstrate that simple generic modifications can be made to such a system to improve its accuracy and reduce the number of frames on which it fails. These are: searching for edges to match to occluding boundaries, which prevents the model from

© 2010. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.
BMVC 2010 doi:10.5244/C.24.81

rotating instead of altering other pose or shape parameters; and altering the model fitting weights to give greater importance to points which are already close to their target position.

The new system is versatile, working well on several different large data sets, and proves superior to [1], reducing point-to-point errors and failures on a variety of data sets.

Figure 1: Search results. The example frames from in-car videos show many problems including: occlusion; a non-uniform background; variable illumination; low contrast; and extreme pose.

2 Search Method

The base system uses a sparse 3D shape model [5] for pose invariance, and continuously updated view-based local patches for illumination compensation. The search on each still image is initialized using the Viola-Jones (V-J) face detector [6]. The detector returns the location of a box, bounding the most likely location of a face. The 3D shape model is placed within the box adopting its default (mean) shape and either 0° rotation or a fixed angle based on prior knowledge. For example, in in-car videos the camera may be mounted below the head and the model is initialised tilted slightly upwards.

In each frame after the first in sequences, the model is initialised with the shape and pose from the previous frame. However, if the search fails, the model is reinitialised on the same frame using the V-J detector and default attributes.

2.1 Shape Model

Some authors build their model using 3D head scans [7, 8] or generate artificial examples [9]. Others use prior knowledge of face deformation [10] or an artificial head model [11] as an alternative. The model used in this work was built from 923 head meshes. Each mesh was created from a manual markup of photographs of an individual. The front and profile shots of each person were marked in detail and the two point sets were combined to produce a 3D representation for that subject (Figure 2). A generic mesh was warped [12] to fit the markup giving a mesh for each individual (Figure 2). Since the same mesh was used in each case the vertices are automatically corresponded across the set.

A subset of 238 vertices (Figure 2) was used to build the model since using the full mesh would be computationally expensive, and many vertices are in areas of little information. Points were chosen which are close to features of interest such as the eyes, nose, and mouth.

The co-ordinates of each example are concatenated into a single vector and Principal Component Analysis is applied to all the point sets to generate a statistical shape model

representation of the data. A shape example $\mathbf{x_i}$ can be represented by the mean shape $\bar{\mathbf{x}}$ plus a linear combination of the principle modes of the data concatenated into a matrix $\mathbf{P}$:

$$\mathbf{x_i} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b_i} \quad (1)$$

where the coefficients $\mathbf{b_i}$ are the model parameters for shape $\mathbf{x_i}$. The number of columns in $\mathbf{P}$ was restricted to 33 which accounts for approximately 93% of the variation in the training data. None of the subjects used in training was present in any of the test images or videos used in the experiments.

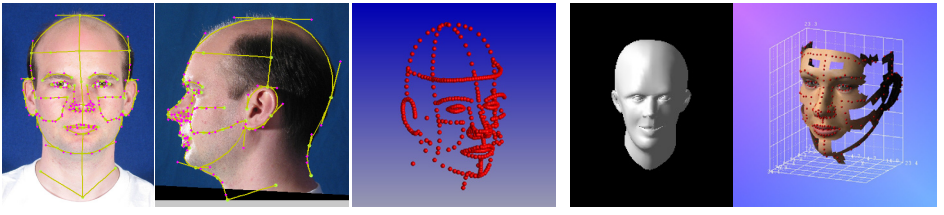

Figure 2: The model is built using manual markups. Each subject is marked in front and profile to create a 3D point set *(3 left)*. A mesh is then warped to match each subject and a subset of vertices on features of interest is extracted from each mesh *(right)*.

2.2 Locating the Feature Points

To compensate for illumination variation the system uses normalised view-based local texture patches similar to Gu and Kanade [9], but continuously updated to reflect the current model pose (Figure 3). The local patches are sampled from an average texture generated from 913 subjects (Figure 3) where each example is a face 'unfolded' from the meshes described in Section 2.1. This patch is always the same size and shape throughout the matching process (5x5 pixels) but changes content at every iteration. It is updated based on the surface normal of the point and the current orientation of the model, and represents the view of the texture at that point. It is assumed for this purpose, that the head is a globe and the texture lies tangential to the surface at each point with its major (UV) axes aligned to the lines of latitude and longitude. Variation in the texture was not modelled for these experiments.

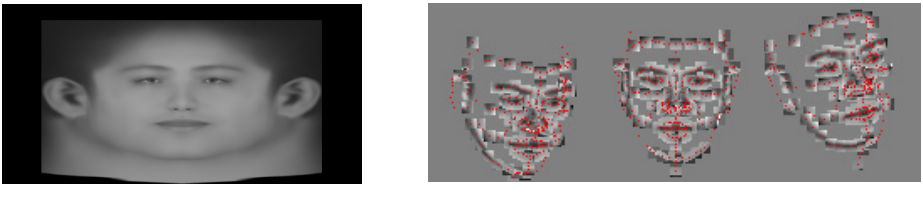

Figure 3: The patches are generated from the mean texture *(left)*. The appearance of the patches changes to match the current pose of the model *(right)*.

Only a subset of 155 points are considered in the search to speed up the algorithm and exclude points where there is less information, for example around the outside of the face. Also, of the 155, only points which have surface normals currently facing forwards (less

than 90 degrees to the view axis) are used to search. The patches are compared to the image in a neighbourhood around each point using normalised correlation. The best match value gives the new target for each point. The parameters of the model are then estimated to bring the 3D model points and their targets into alignment. To do this each 2D target is assigned the z-component of the matching model point, which assumes an orthogonal projection. This fitting is a two stage process extended to 3D from the 2D ASM, refer to [5] for full details. Firstly the points are rigidly aligned (rotation, scale, and translation) to minimise the sum of squared distances between matched points, then the shape model parameters (**b** in (1)) are updated using a least squares approximation. During the fitting, weights can be assigned to each point. The calculation of these weights is one of the areas investigated in this work and discussed in the next section.

3 Modifications

The results obtained with the basic 3D system were good and superior to an established 2D method [1]. Here we investigate two ways that such a search may be enhanced and show that large improvements are possible.

3.1 Weighting

When the model is fitted to the target points each point is given a weighting, $\boldsymbol{w_j}$, based on the match obtained. The original method [1] used a combination of the match value at the target and its difference from that at the current position.

$$w_j = m_j * (1 - \frac{v_j}{\max\{v_j\}_{\forall j}}); \qquad m_j = \begin{cases} 1 & v_c = 0 \\ 1 - \dfrac{v_j}{v_c} & j \neq c \\ 1 - \dfrac{v_c}{\min\{v_k\}_{\forall k \neq c}} & j = c \end{cases} \tag{2}$$

where: v_j is the match value at the target; m_j is a match weight; c indicates the current point position; v_c is the match value at that point; and the v_k are the match values at all points in the neighbourhood. We examined the efficacy of this scheme by comparing it to one based on the match value alone (the second multiplier in (2)), also to uniform weights (value 1.0), and we investigated a new method which ignores the match value and is based on the distance between the current point and the target point d_j:

$$w_j = 1 - \frac{d_j}{\max\{d\}_{\forall j}} \tag{3}$$

The aim of this latter scheme is to keep in place points that are already matched, rather than allowing them to be pulled away by spurious matches in other parts of the model.

This may be because that part of the model is less well positioned, or because the subject is occluded in that area of the image and a good match cannot be found.

3.2 Searching for Occluding Boundaries

Searching using only the patches does not take advantage of important image information from the occluding boundaries. In some cases the search result was poor because the model did not expand to match the size of the face, or because the mouth of the model was confused with the nose, or because the model rotated to match the features rather than translating or changing shape. To correct for this we added an explicit search for occluding boundaries. By using the surface normals of the original mesh, points which are likely to be on the boundary can be identified, because they are currently facing approximately 90 degrees to the viewing direction. For each of these points the search is directed along a line in the image perpendicular to the boundary to find the strongest edge. The surface normal determines the direction of the search which is 3 pixels wide in both directions from the point (Figure 4).

The length of the profile is the same as the neighbourhood size used in the patch matching, which is reduced as the search resolution is increased. Since the target point is on the strongest edge, the 'match value' is the normalised edge strength.

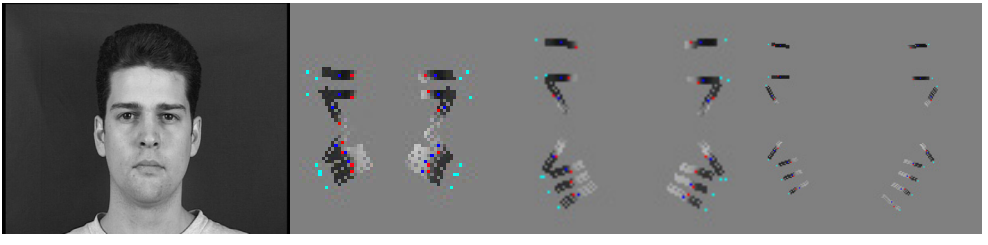

Figure 4: Those points facing at approximately 90 degrees to the view angle search along a profile for the strongest edge instead of matching a local patch. The appearance of the profiles at different resolutions is shown. The light spot is the surface normal direction, the other points indicate the current model position and the strongest edge.

4 DataSets

To test the modifications presented here we have used 6 data sets with a variety of challenges.

4.1 Still Images

Two large publicly available datasets were used, neither of which contain large variations in pose (Figure 5):

- XM2VTS [4]: 2344 images of 295 individuals at 720x576 pixels. This database has the least challenging type of images. The subjects are posed against a fairly uniform backdrop and in general have neutral expressions and near frontal poses.

- BioID [3]: 1520 images of 23 individuals at 384x286 pixels. This database shows much more natural poses (although still mainly near frontal) and a variety of expressions. The data was acquired in an office setting so has cluttered backgrounds.

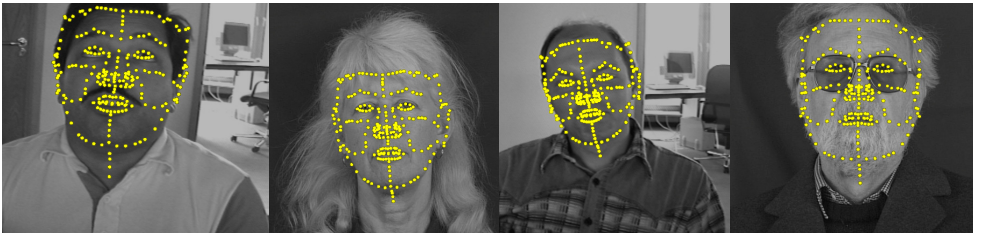

Figure 5: The still images show many problems including: glasses; facial hair; low contrast; and a cluttered background.

4.2 Real-World Video Sequences

Three video sequences of 3 different drivers were analysed. These were taken with an in-car camera fixed below head level, behind the steering wheel. Each sequence is 2000 frames long but only every 10th frame is manually marked. Due to the movement of the steering wheel, the head is obscured in some of these, which reduced the numbers of frames suitable for evaluation to 150, 156, and 136. Since the camera was known to be situated below eye-level the model was initialised at a 40 degree upwards pitch when searching these sequences. As these are real in-car videos there is a great deal of lighting variation, within and between frames, and, although the driver is facing forwards for a large part of the time, there is a wider variation in poses than in the still images. Figure 1 illustrates the typical problems.

4.3 Artificial Images with Known Poses

This publicly available dataset was devised to assess the ability of face tracking methods to deal with large poses. It comprises of a series of images of artificial subjects with known attitudes. They were created using a full mesh statistical shape model of the head and matching texture model. 20 synthetic subjects were used, posed against a real in-car background. Figure 6 shows some examples. The heads were posed as follows:

- Heading +/- (*r/l*) 90º in 10 degree intervals (right and left as viewed)
- Pitch +/- (*d/u*) 60º in 10 degree intervals
- Roll +/- (*r/l*) 90º in 10 degree intervals

The data set contains 977 images with known feature point positions. For each rotation direction, the images were presented to the system as a sequence starting at zero each time.

Figure 6: Some examples of artificial subjects *(top)* and poses *(bottom)*.

4.4 Assessment

All of these data sets have annotations but the same features are not located in each and there are differences between the annotations and the model points. Because of this only a subset of 12 points was used for point-to-point error evaluation. The points chosen are located on the better defined features, common to all sets: the ends of the eyebrows (least well localised manually); the corners of the eyes (well localised); the corners of the mouth (well localised); and the top and bottom of the mouth (moderately well localised).

The system is initialised using the V-J face detector. On the still image databases this is done on every image. We assessed the detector's performance by comparing the box returned by the algorithm to the 12 points used in the evaluation. If any points fell outside the box the detection was considered a failure. It was found that the detector failed on 8% of the BioID data set. These examples were excluded from the analysis since the method requires initialisation in the location of the face.

Where errors are quoted these are the median of the average point to point errors for each example. In most cases this is a percentage of the inter-ocular distance, which normalises for the fact that the head size can vary across the data set. This is particularly true of the BioID images. For the artificial images this is presented as pixels but the inter-ocular distance is approximately 100 pixels.

5 Results

The graphs of Figure 7 show the effect each modification, or combination, had on median average point-to-point error and the number of failures. The circles in the graphs show the relative number of failures and their vertical position indicates the error value. Failure is defined by an average point to point error of over 15 percent or pixels. Since the eyes are easiest to locate manually the errors are reported on all 12 feature points and separately for just the eye corners. The errors over all points are universally larger than those for the eyes, they are shown on the top row of each graph. The result(s) judged to be most successful are shaded. Method codes are as follows: **B**asic Method (2); **P**rofile Search; Equal weighting of **1**.0; Weighting using **M**atch Value; Weighting using **D**istance (3)[1].

The improvements on the XM2VTS data set are not as dramatic as those on the BioID set which are much more difficult cases and have therefore benefitted more from the

[1] Refer to the supplementary video for an example search sequence comparing B and combined PD methods

modifications. In fact, the failures in the BioID set (all points) were approximately halved. Interestingly, all the data sets showed improvements when both modifications were added together (PD), but the responses to the changes individually were not consistent. In most cases, the new distance weighting scheme is obviously superior. It is likely that the boundary search scheme is beneficial in only a small number of cases and this is why it has a less pronounced effect. Figure 8 shows typical situations where the boundary search has improved performance. One is that, because of the extra degrees of freedom in 3D, the model can sometimes rotate to bring the features into alignment rather than changing shape, size, or position. The other situation is where the mouth matches to the nose and the model shrinks rather than expanding to fit the face.

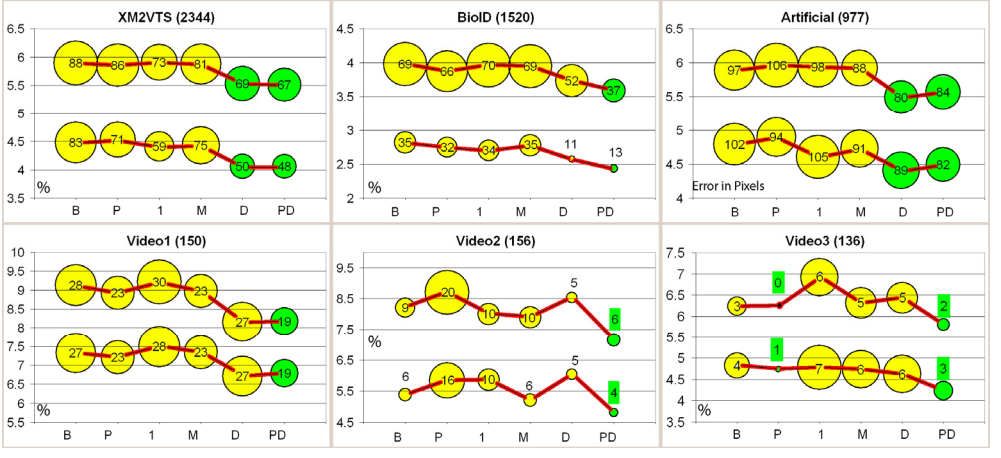

Figure 7: The point to point errors and failures for the different schemes. For each dataset the number of images is shown in brackets. The explanation of the codes on the horizontal axis is in section 5. The vertical axis is the median average error as % inter-ocular distance or pixels, as shown. The vertical position of the bubble indicates the error value and its size is the number of failures, also shown inside. The top row in each graph shows the results for all points, the bottom shows those for the eyes only. The darker bubbles are those results which were deemed best in each case

6 Discussion

It has been shown that these simple generic modifications have an impact on search success. Altering the point weighting scheme to ignore match value but use target distance has an important and surprising effect, and performance can further be improved by introducing additional matching methods such as boundary detection, which can prevent cases of spurious rotation and shrinkage.

Since these modifications are non-specific we suggest they can be considered in other search algorithms of this type which use weighted model fitting, and where occluding boundaries may vary by view and are not explicitly modeled, particularly in 3D where the added degrees of freedom can cause ambiguities.

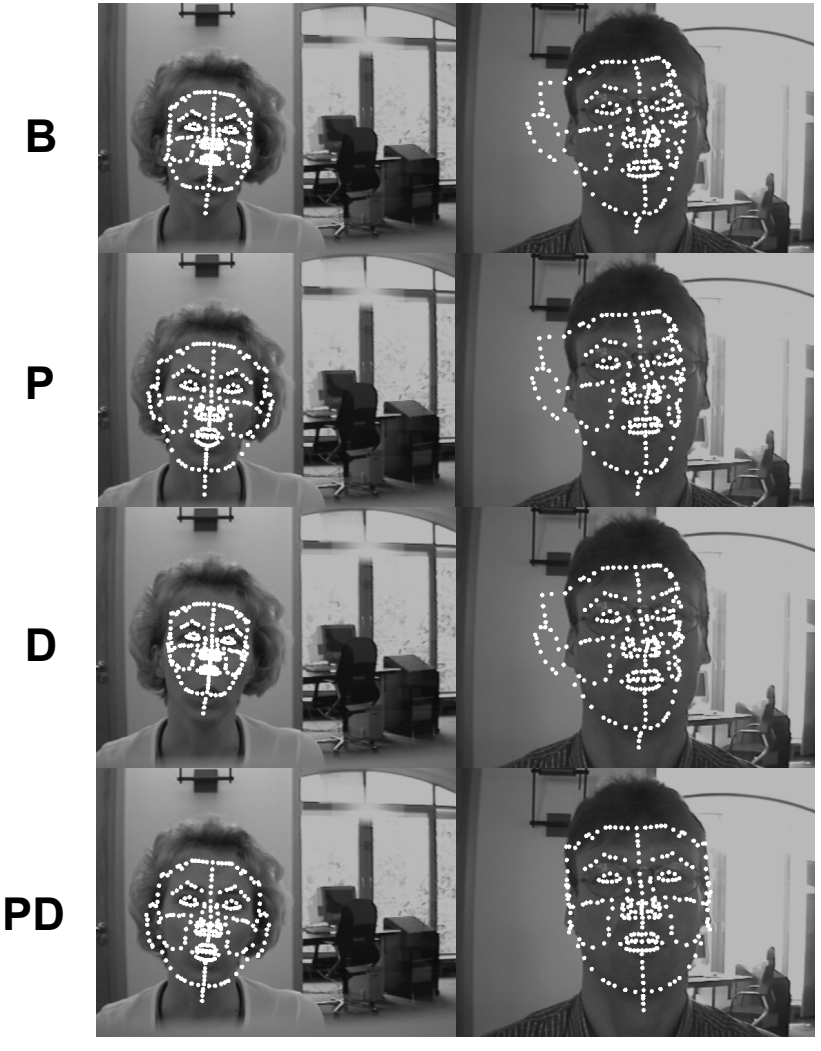

Figure 8: Examples where occluding boundary search helped. Left: The model is too small and the mouth has been matched to the nose. Introducing boundary search has allowed it to expand to the correct size. Right: The model has rotated to try and match the features – this is a case where both modifications worked together. The method codes are as in section 5.

Acknowledgements

This project is funded by Toyota Motor Europe, who provided the driver videos. We would like to thank Genemation Ltd for the 3D data markups and head textures.

References

[1] A. Caunce, D. Cristinacce, C. Taylor, and T. Cootes, "Locating Facial Features and Pose Estimation Using a 3D Shape Model," *Proceedings of International Symposium on Visual Computing*, Las Vegas, 750-761, 2009.

[2] D. Cristinacce and T. Cootes, "Automatic feature localisation with constrained local models," *Pattern Recognition*, vol. 41, pp. 3054-3067, 2007.

[3] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz, "Robust Face Detection Using the Hausdorff Distance," *Proceedings of International Conference on Audio- and Video-based Biometric Authentication*, Halmstaad, Sweden, 2001.

[4] K. Messer, J. Matas, J. Kittler, and K. Jonsson, "XM2VTSDB: The Extended M2VTS Database," *Proceedings of International Conference on Audio- and Video-based Biometric Person Authentication*, Washington DC, USA, 1999.

[5] T. F. Cootes, D. H. Cooper, C. J. Taylor, and J. Graham, "Active Shape Models - Their Training and Application," *Computer Vision and Image Understanding*, vol. 61, pp. 38-59, 1995.

[6] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol. 57, pp. 137-154, 2004.

[7] V. Blanz and T. Vetter, "A Morphable Model for the Synthesis of 3D Faces," *SIGGRAPH*, 1999.

[8] S. Romdhani, J. Ho, T. Vetter, and D. J. Kriegman, "Face Recognition Using 3-D Models: Pose and Illumination," *Proceedings of the IEEE*, vol. 94, pp. 1977-1999, 2006.

[9] L. Gu and T. Kanade, "3D Alignment of Face in a Single Image," *Proceedings of International Conference on Computer Vision and Pattern Recognition*, New York, 2006.

[10] C. Vogler, Z. Li, A. Kanaujia, S. Goldenstein, and D. Metaxas, "The Best of Both Worlds: Combining 3D Deformable Models with Active Shape Models," *Proceedings of International Conference on Computer Vision*, 2007.

[11] F. Dornaika and J. Ahlberg, "Fitting 3D face models for tracking and active appearance model training," *Image and Vision Computing*, vol. 24, pp. 1010-1024, 2006.

[12] F. L. Bookstein, "Principal Warps: Thin-Plate Splines and the Decomposition of Deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 567-585, 1989.