AUTOMATIC FRONTAL FACE ANNOTATION AND AAM BUILDING FOR ARBITRARY EXPRESSIONS FROM A SINGLE FRONTAL IMAGE ONLY

Akshay Asthana[1] *Asim Khwaja*[1] *Roland Goecke*[1,2]

[1]RSISE, CECS, Australian National University, Canberra, Australia
[2]HCC Lab / NCBS, Faculty of Information Sciences and Engineering, University of Canberra, Australia
aasthana@rsise.anu.edu.au, asim.khwaja@anu.edu.au, roland.goecke@ieee.org

ABSTRACT

In recent years, statistically motivated approaches for the registration and tracking of non-rigid objects, such as the Active Appearance Model (AAM), have become very popular. A major drawback of these approaches is that they require manual annotation of all training images which can be tedious and error prone. In this paper, a MPEG-4 based approach for the automatic annotation of frontal face images, having any arbitrary facial expression, from a single annotated frontal image is presented. This approach utilises the MPEG-4 based facial animation system to generate virtual images having different expressions and uses the existing AAM framework to automatically annotate unseen images. The approach demonstrates an excellent generalisability by automatically annotating face images from two different databases.

***Index Terms*—** Facial modelling, Active Appearance Model (AAM), Automatic Annotation.

1. INTRODUCTION

Statistically motivated approaches for registration and tracking of non-rigid objects, such as the Active Shape Model (ASM) [1], Active Appearance Model (AAM) [2] and 3D Morphable Model (3DMM) [3] are becoming increasingly popular by virtue of their fast and efficient modelling and alignment methods. However, these approaches require manual annotation of a large set of images that can be tedious and error prone. In addition, some objects (e.g. a human face) can have certain structures (e.g. the jaw line) that may be difficult to consistently annotate due to a lack of distinct landmarks.

In [4], the problem of automatic annotation and model building is posed as an energy-minimising image coding problem. A direct pair-wise method for automatic correspondence learning was proposed in [5], in which the non-rigid registration is performed between the pre-annotated reference image and other images in the training database, so that the possible deformations can be learnt, assuming a smooth deformation between the pair of images. This work was further extended in [6] to finding automatic correspondences in monocular videos within an adaptive tracking paradigm and by exploiting the epipolar geometry constraints in stereo videos.

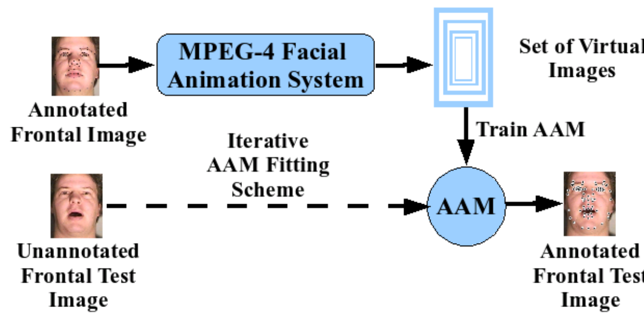


Fig. 1: Overall architecture of the proposed approach

In this paper, an approach for automatic annotation of unseen frontal images and AAM building from a single frontal annotated image is presented (Fig. 1). This approach uses the parametric representation of AAMs to its advantage and relies on generating a standard set of virtual images, exhibiting arbitrary facial expressions, that will cover most facial expressions normally possible for humans. For synthesising the set of virtual images (Sec. 2.2), the system uses the MPEG-4 standard facial animation system [7, 8]. This set of virtual images is used to train an AAM (Sec. 2.3) and an iterative AAM fitting scheme (Alg. 2) is used to locate the landmarks in the given unseen image and, hence, annotate it automatically.

The remainder of this paper is structured as follows. In Sec. 2, the process of reconstructing virtual images and automatically building an AAM for automatic annotation of unseen images is presented. The approach is experimentally validated in Sec. 3. Finally, the conclusions are given in Sec. 4.

2. AUTOMATIC ANNOTATION FRAMEWORK

The proposed framework uses the MPEG-4 standard facial animation system [7, 8]. MPEG-4 is a collection of methods to define audio and visual digital data in a compressed form and to model objects like human faces. The advantage of using a MPEG-4 based facial animation system is that it has a

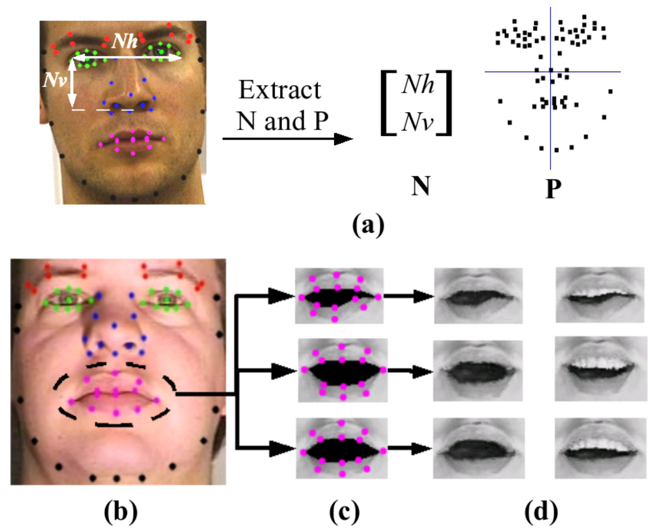


Fig. 2: (a) Extracting Normalised and Point Vectors. (b) Face partitioning: Different regions represented by different colours of the landmarks. (c) Some examples of reconstructed mouth regions. (d) 2 variations in handling the missing information about the oral cavity

standard set of rules defining most facial expressions possible for humans and require little data to generate the desired facial expression [9].

2.1. MPEG-4 based Facial Animation System

This system [9] presents an approach to generate the facial animations using Facial Definition Parameters[1] (FDP) and Facial Animation Parameters[2] (FAP). The FDPs represent the face shape and texture, while FAPs represent a complete set of basic facial actions that enable the system to animate most facial expressions. There are 68 FAPs in the MPEG-4 standard that contain 2 high-level FAPs (viseme FAP and expression FAP) and 66 low-level FAPs. Any particular expression can be animated by using a linear combination of predefined viseme or expression FAPs, while low-level FAPs express the motion of different facial parts (e.g. jaw, eyebrow, lips etc). The low-level FAPs can easily perform the functions of high-level FAPs, but can also generate random expressions that cannot be defined by high-level FAPs. Our approach mainly deals with low-level FAPs.

2.2. Reconstruction of Virtual Images

Given a single annotated frontal face image, the goal here is to use the shape and texture information from the given image and reconstruct virtual images having arbitrary movements of all the facial parts under the cropped face area according to

---

Algorithm 1 Reconstruction of Virtual Images

Notations:

$\mathbf{N} = [N_h; N_v]^T$ - Normalisation Vector used to normalise the shape vector w.r.t. varying shape & size of different faces.

$\mathbf{P}(or\ \mathbf{P'}) = [x_1; y_1; \ldots; x_n; y_n]^T$ - Point Vector representing n landmarks in the normalised frame.

Require: Single Annotated Frontal Face Image *Img*.

1: Extract $\mathbf{N}$ and $\mathbf{P}$ as shown in Fig. 2(a).

2: Generate $\mathbf{P}^{'}$ by moving the landmarks representing a facial feature (e.g. mouth) by distance $d \in [0, 0.3]$ in the normalised frame following the FAPs scheme for that particular facial feature.

3: De-normalise $\mathbf{P}^{'}$ using the $\mathbf{N}$ for real world coordinates.

for $i = 1$ to n

$x_i^{real} = x_i^{'}.(N_h)$ where . denotes multiplication

$y_i^{real} = y_i^{'}.(N_v)$

end for

4: Warp the texture from *Img* to the new locations using Piece-wise Affine Warping (PAW) [10] and fill the invalid pixel locations with their nearest neighbours. Fig. 2(c) shows the reconstructed virtual image.

5: In case of missing texture information (e.g. oral cavity), warp the mean texture from the corresponding region of an average face model using PAW. Fig. 2(d) shows the warped mean texture of the oral cavity without and with teeth in the final reconstructed virtual images.

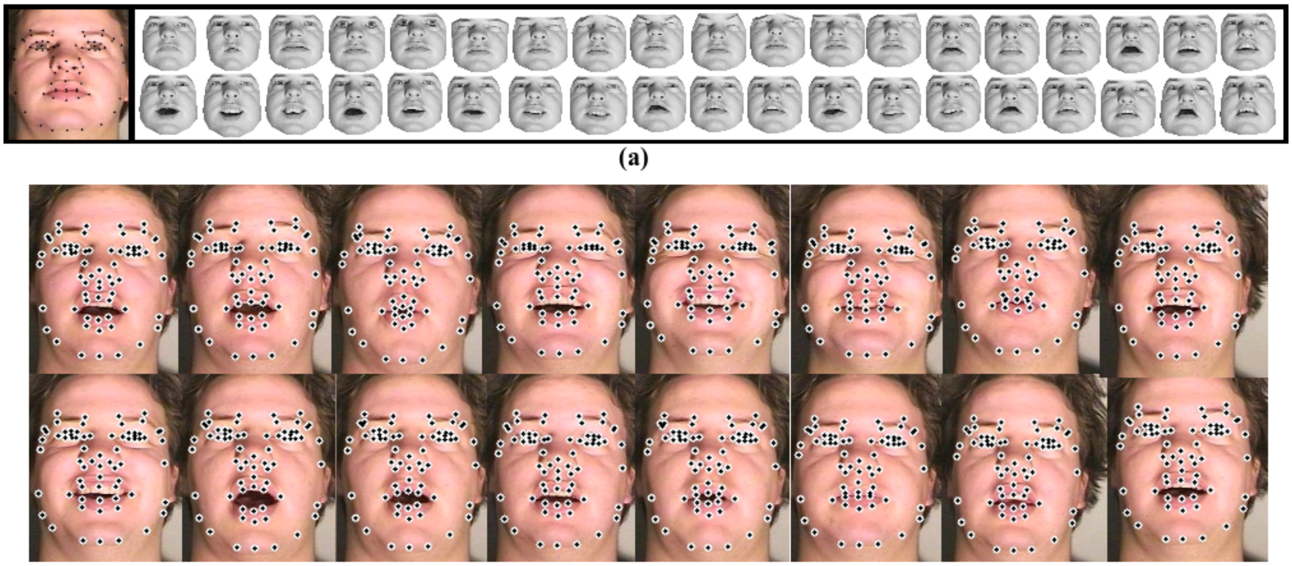
---

the FAP description. The virtual images can then be used to automatically build an AAM for fitting / tracking. For the experiments presented in this paper, the cropped face area contained six different facial features: eyebrows, eyes, nose, mouth, cheeks and jaw. In the MPEG-4 standard, these features are described by 54 low level FAPs and are used for reconstructing the set of virtual images. In some cases, e.g. for the mouth region, the texture information for the oral cavity may be incomplete or totally unavailable. To complete a virtual image that might need this missing information, we use the mean texture of the oral cavity from an average face model to warp the texture into this region. For the experiments presented in this paper, we use two different variants of the mean texture for the oral cavity that provide texture information for the oral cavity alone and for the oral cavity with teeth visible. Fig. 2(c) shows the subset of virtual images reconstructed by the combination of FAPs 4, 5, 8, 9, 10, 11, 51, 52, 55, 56, 57 and 58. A similar method is adopted for completing the missing texture information for the eye region (Fig. 3(a)). The step-by-step procedure is given in Alg. 1.

2.3. Automatic AAM building and Fitting

The reconstructed virtual images are used to train an AAM and existing AAM fitting methods can be easily utilised to locate the facial features in the original images and, hence,

[1] http://www.dsp.dist.unige.it/~pok/RESEARCH/MPEG/fdpspec.htm
[2] http://www.dsp.dist.unige.it/~pok/RESEARCH/MPEG/fapspec.htm



(a) **(b)**

Fig. 3: (a) Set of virtual images reconstructed from a single sample image, (b) Automatic annotation results of sample images

annotate them automatically. Since the reconstructed virtual images have no information about the area outside the convex hull of the canonical shape or the background, a model trained on these images can be vulnerable to a changing background, ill-defined borders and poor initialisation [11]. To deal with changing backgrounds, we use the existing *Simultaneous Inverse Compositional* method (SIC) [12] - a generative fitting method - where the update model is generated directly from *background free* components (i.e. the mean appearance and their modes of variation) and has no specialisation to any particular background. To deal with ill-defined borders and poor initialisation, we use an iterative fitting scheme in which we initialise the model at several different positions and choose the fitting result that gives the minimum residual texture error. This iterative fitting scheme is explained in Alg. 2.

3. EXPERIMENTS AND DISCUSSION

We conducted experiments on the AVOZES [13] and CMU PIE databases [14]. Overall, 240 images across 10 subjects from the AVOZES database and 600 images across 10 subjects from the CMU PIE database were manually annotated with 69 landmarks each to provide a ground-truth. Now given a single annotated frontal image of a subject, virtual images were reconstructed (Sec. 2.2) and were used to train an AAM (Sec. 2.3) that performed the AAM fitting (Alg. 2), hence, automatically annotating the unseen images of the subject, having any random facial expression. Fig. 3 shows a subset of reconstructed virtual images for a sample speaker from AVOZES and the automatic annotation results obtained from the AAM trained on these virtual images for this subject.

To evaluate the performance of the proposed automatic annotation framework, the pixel error per landmark was computed between manual and automatic annotations for every image. It should be noted here that consistently annotating the outer boundary of the face is highly error prone due to a lack of distinct features. Therefore, we computed the pixel error per landmark by excluding the 13 landmarks that represent the outer boundary for each face. Fig. 4 shows the error distribution for the two datasets (AVOZES and CMU PIE). Assuming the face cropped area to be $16 \times 18\text{cm}^2$ in the real

Algorithm 2 Iterative Fitting Scheme

Notations:

$\mathcal{E}_r(x, y)$ - Residual texture error obtained by AAM fitting when model is initialised at location (x, y)

$\mathbf{s}_{(x,y)}$ - Shape vector obtained by AAM fitting when model is initialised at location (x, y)

$\delta_x \mathrm{X} \delta_y$ - Size of the initialisation search window.

Input: Face Image, *I*, to be annotated.

```
Initialise the fitting procedure at the centre, C(x, y), of
  the bounding box obtained by the face detector.
for i = (−δx/2) to +δx do
    for j = −δy to +δy do
        Compute the residual texture error Er(x + i, y + j)
        if Er(x + i, y + j) < MIN(Er) then
            MIN(Er) = Er(x + i, y + j)
            s = s(x+i,y+j)
        end if
    end for
end for
```

Output: Shape vector, $\mathbf{s}$, representing the landmarks for *I*.

Databases	AVOZES	CMU PIE
Avg. Cropped Area ($pixel^2$)	190×220	140×150
Mean Error (pixel/landmark)	2.24	1.26
St. Dev. (pixel/landmark)	0.52	0.36
Assumed Cropped Area (mm^2)	160×180	160×180
Mean Error (mm/landmark)	1.88	1.49
St. Dev. (mm/landmark)	0.42	0.43

Table 1: Overall annotation errors obtained for the AVOZES and CMU PIE Databases

world, we computed the approximate real world error (mm per landmark) for both the two datasets (Table 1).

4. CONCLUSION

An approach for the automatic annotation of frontal face images with any random expression from a single annotated frontal image has been presented. The framework exhibits impressive accuracy in annotating images from the AVOZES and CMU PIE databases with an annotation error of 2.24 ± 0.52 pixel/landmark ($\approx 1.88 \pm 0.42$ mm/landmark) and 1.26 ± 0.36 pixel/landmark ($\approx 1.49 \pm 0.43$mm/landmark), respectively. In future, we plan to work on automatically annotating the single frontal image that currently needs to be annotated manually, thus making the process completely automatic.

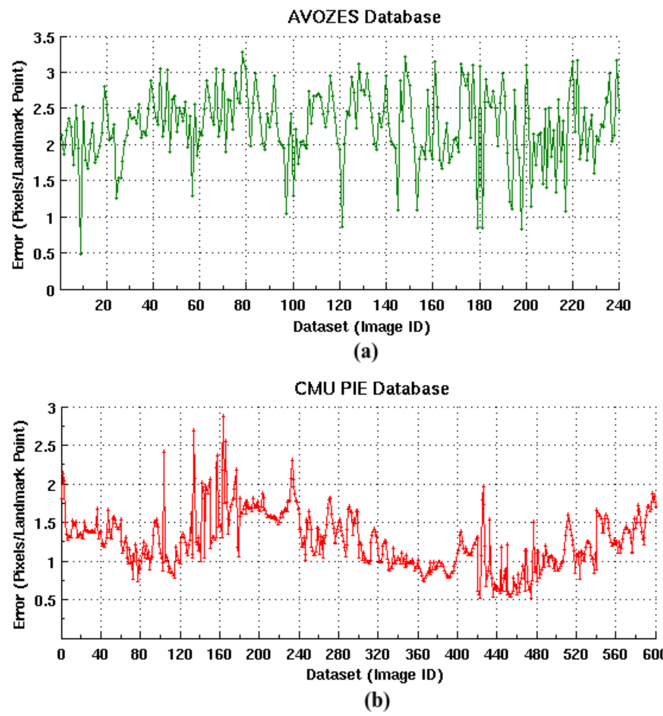


Fig. 4: Automatic Annotation Results from (a) AVOZES Database, (b) CMU PIE Database

5. REFERENCES

[1] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active shape models - their training and applications," *CVIU*, vol. 61, no. 1, pp. 38–59, Jan. 1995.

[2] G. Edwards, C.J. Taylor, and T.F. Cootes, "Interpreting Face Images Using Active Appearance Models," in *Proc. FG'98*. Apr. 1998, pp. 300–305, IEEE.

[3] V. Blanz and T. Vetter, "Face Recognition Based on Fitting a 3D Morphable Model," *IEEE PAMI*, vol. 25, no. 9, pp. 1063–1074, Sept. 2003.

[4] S. Baker, I. Matthews, and J. Schneider, "Automatic Construction of Active Appearance Models as an Image Coding Problem," *IEEE PAMI*, vol. 26, no. 10, pp. 1380–1384, Oct. 2004.

[5] J. Saragih and R. Goecke, "Learning Active Appearance Models from Image Sequences," in *Proc. VisHCI 2006*, Nov. 2006, vol. 56 of *CRPIT*, pp. 51–60.

[6] J. Saragih and R. Goecke, "Monocular and Stereo Methods for AAM Learning from Video," in *Proc. CVPR 2007*, June 2007, DOI: 10.1109/ICCV.2007.4409106.

[7] ISO/IEC 14496-1:2001, *Coding of Audio-Visual Objects : Systems*.

[8] ISO/IEC 14496-2:2001, *Coding of Audio-Visual Objects : Visual*.

[9] F. Lavagetto and R. Pockaj, "The Facial Animation Engine: towards a high-level interface for the design of MPEG-4 compliant animated faces," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 277–289, 1999.

[10] I. Matthews and S. Baker, "Active Appearance Models Revisited," *IJCV*, vol. 60, no. 2, pp. 135–164, 2004.

[11] M.B. Stegmann, *Active Appearance Models: Theory, Extensions & Cases*, Ph.D. thesis, Technical University of Denmark, DTU, 2000.

[12] S. Baker, R. Gross, and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework: Part 3," Tech. Rep., RI, Carnegie Mellon University, USA, 2003.

[13] R. Goecke and B. Millar, "The Audio-Video Australian English Speech Data Corpus AVOZES," in *Proc. ICSLP2004*, Jeju, Korea, 2004, vol. III, pp. 2525–2528.

[14] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression Database," *IEEE PAMI*, vol. 25, no. 12, pp. 1615–1618, 2003.