AN UNSUPERVISED ALGORITHM TO EXTRACT FACE TEXTURE FROM VIDEO

Yue Zhou and Thomas S. Huang

Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign, Urbana, IL 61801
yuezhou,huang@ifp.uiuc.edu

ABSTRACT

Building face models is an essential task in face recognition, tracking and etc. However, most of the current techniques require hand-labelling or special machinery such as cyber-scanner to extract the face model. In the paper, we propose an unsupervised algorithm to learn the face texture from video. The proposed approach models the video sequence as a mixture of dynamic face-layers and background layers, where the dynamic face-layers may undergo 3D motions in the video. The hidden variables and their generating process is represented by probabilistic graphical model. The model is learnt by EM algorithm with variational approximation. The proposed approach offers several advantage over existing algorithms. First, it derive its learning power by a generative model which naturally represents the generating process of videos. Second, it does not require any labelling or face detection algorithm. Third, the application domain of the proposed algorithm is not restricted to extracting face texture and it can be adapted to model other objects as well. The experimental results demonstrate that our model is capable of learning the appearance model of faces with complex 3D motions in the video.

Index Terms— Video signal processing, unsupervised learning, pattern recognition

1. INTRODUCTION

Extracting face models from video sequence is an very important part for face recognition and many other tasks. While two-dimensional face models are widely used in face recognition, its limitation is obvious because generally they do not distinguish rotation angle and shape of face images. One way to overcome the limitation is to combine 2D face model from multiple views [1]. 3D face models match the observed image with a rigid or deformable 3D geometry and texture [2]. It completely separates the shape and rotation and is therefore far more desirable in face recognition. The 3D face model can be manually defined and labelled , or learnt from 3D scans of heads without texture or with texture . However, manually labelling requires much labor power and introduces subjective error and 3D scanner need specially equipment such as cyber scanner which is not popularly used.

In the paper, we propose a new unsupervised algorithm to learn a semi-3D face model. The observed face image is represented as a mixture of dynamic face-layers and background. As previous research shows, the 2D image is insufficient in modelling the face; however, a fully 3D face model introduces too many degrees of freedom and is hard to compute in an unsupervised way. We therefore adopt a semi-3D face model with dynamic foreground layers, i.e. the geometry of the face is simplified to a cylinder with face texture projected onto the cylindrical surface while the parameters of the cylinder and have to be learnt from the video. The unknown parameters of the model include size and the texture of the dynamic face-layer. For most applications we found the semi-3D model is a good enough approximation for tasks like face recognition, face tracking and pose estimation.

The proposed algorithm is mathematically formulated as the problem to learn a generative model such that the observed video is well explained and scene analysis can be performed efficiently on top of that model. Such generative model [3] [4] [5] usually includes the estimation of layer appearance, shape and motion and these variables are iteratively updated by EM algorithm [6]. In the proposed algorithm, probabilistic graphical model is used to represent the random variables and their relationship by nodes and conditional probabilities and to provide a unified framework for learning and inference.

In this paper the extraction of face model is considered as a probabilistic clustering problem, where the input data is generated by mixing different clusters. Each cluster has uni-modal density and may undergo 3D transformation. We propose a inference algorithm that jointly estimates the layer shape/appearance and 3D motion in an unsupervised manner. Theoretically, this algorithm does not requires the frames to be temporally order to compute the clusters; however, our experiments indicate that the computational complexity can be significantly improved by using motion prior which is straightforward to computed from ordered video sequences.

The rest of the paper is organized as follows: we describe the model in Section 2 and the generative model in Section 3; the learning is discussed in Section 4; we show the experimental results in Section 5 and conclude in Section 6.

2. THE DYNAMIC FACE-LAYER MODEL

Extensive research has been performed in modelling face geometry and texture. 2D face models are usually built from face images with manually labelled feature points such as eye corners and mouth corners. 3D face models are much more difficult to obtain and usually require special equipment like 3D scanner. The drawback of these approaches is that none of them are unsupervised, which is not desirable for a automatic system.

We therefore propose to use dynamic face-layer (foreground) and background layer model to solve this problem. Face videos usually include out-plane rotation in a very wide angle and directly representing face with a 2D layers is not feasible. To our best knowledge, most previous research handles out-plane motion by the probabilistic variation of the model or shape/appearance deformation without explicitly computing the out-plane motion. This does not work either in modelling the face because the 3D motion of face is large and can not be explained by model variation and deformation. A model with 3D motion parameters is apparently necessary for the face data. However, with too many parameters in the model we face the *curse of dimensionality* and the problem becomes intractable. We therefore propose a semi-3D generative model which gives us a good balance between representation power and computational complexity.

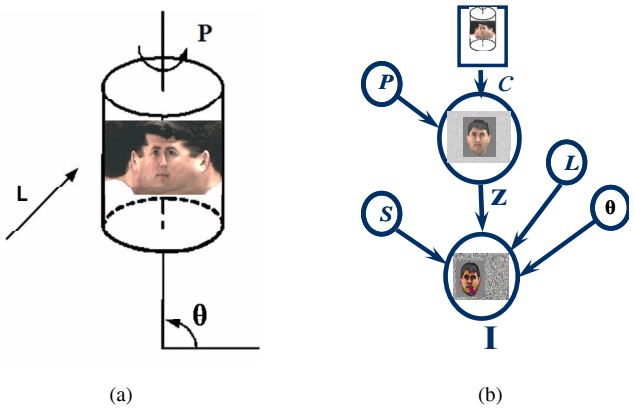
The shape of the 3D face layer is modelled as a cylinder with face appearance projected on the cylinder surface. The texture of the cylindrical layer is stored in a 2D unfolded image. The shape parameters of the cylindrical layer include its height h and radius r. The motion parameters include translation L, in-plane rotation θ and out-plane rotation P. The motion parameters of the model are illustrated in Fig 1(a), where the human face is represented as a cylindrical dynamic face-layer and the face texture is stored in a 2D unfolded image of the cylinder.

In this paper, for simplicity, we only consider the situation where there are only one dynamic face-layer and background layer respectively. However, the proposed model can be extended to arbitrary number of layers in a straightforward way.

We denote the dynamic face-layer as f and the background layer as b. The shape mask m of the object layer is a rectangular binary map and can be easily computed given the object size and motion. The generating process of the observed video frame I at each pixel is modelled as:

$$I = m_T \cdot f_T + (1 - m_T) \cdot b + w \quad (1)$$

where w is zero-mean Gaussian random noise, T is the transformation of the dynamic face-layer including translation, in-plane and out-plane rotation; m_T is the shape mask after transformation T ($m_T \in 0, 1$), and f_T is the foreground layer after transformation T. The equations shows that the observed image is modelled as the mixture of dynamic face-layers, background layers and observation noise. The dynamic face-layers and background layers are associated with a shape parameter that consists the mixture weight at each pixel.



(a) (b)

Fig. 1. The generative model of the dynamic face-layer. (a)The motion model; (b)The generative process. See text for details.

3. THE GENERATIVE MODEL FOR FACE-LAYERS WITH 3D MOTION

The generative model of the face layer is illustrated in Fig 1(b). The generating process is an extension to the Transformed Mixture of Gaussian model (TMG) in [7]. The model is represented by direct probabilistic graphical model where each node represents a hidden variable and each edge represents a conditional probability.

The cluster c and transformation $T = \{L, P, \theta\}$ are drawn from the parameter space according to their prior distributions; a 2D latent face image Z is generated by 3D-to-2D projection given the cluster mean, variance and out-plane rotation P; then the translation L and in-plane rotation θ are applied to get the transformed latent image; finally the observed image is obtained by mixing the zero-mean Gaussian noise with the transformed latent image.

Given the above generative process, we define the observation likelihood of the image as:

$$P(I|T, m, f, b) = N(I; m_T \cdot f_T + (1 - m_T) \cdot b, \sigma_o^2) \quad (2)$$

and the likelihood of the video sequence is:

$$\prod_t P(I_t|T_t, m, f, b) \quad (3)$$

where I_t, T_t are the observed frame and transformation at time t. The likelihood of one image frame is a Gaussian with

mean of the mixture of background and foreground. The overall likelihood of the video is the product of the likelihood for each frames. We further assume that all pixels in one frame are independent and the likelihood of one frame can be factorized to the product of the likelihood of each pixel:

$$P(I|T,m,f,b) = \prod_i N(I(x_i); m_T \cdot f_T(x_i) + (1-m_T)\cdot b(x_i), \sigma_o^2(x_i)) \quad (4)$$

where x_i is the set of all the pixels in the frame.

The TMG model in [7] and the flexible sprites model in [4] only use the 2D linear transforms of the dynamic foreground layers. However, the 3D dynamic face-layer has nonlinear transformations such as out-plane rotation and projection. This introduces a problem because the projection from 3D surface to 2D image is a multiple to one mapping, i.e., multiple pixels in the 3D surface can correspond to one pixel in the 2D image. In our algorithm we use the linear interpolation of the four nearest neighbors.

4. LEARNING THE MODEL

4.1. Variational EM

We use expectation maximization (EM) [6] to learn the model parameters under the framework of maximum likelihood (MLE). The EM algorithm is an iterative algorithm that maximizes the probability of the observed data according to the model. Exact EM is intractable in our model because of its large parameter space. We instead choose variational method [8] to solve this problem approximately.

In variational EM, we assume that posterior has a factorized form:

$$\prod_t P(I_t|T_t,r,h,f,b) = \prod_t q(r)\cdot q(h)\cdot q(T_t) \quad (5)$$

where we assume $q(r) = N(r;\phi_r,\sigma_r^2)$, $q(h) = N(h;\phi_h,\sigma_h^2)$ and σ_r^2, σ_h^2 are known.

The variational bound of the log-likelihood is:

$$F = \sum_t \sum_{T_t} \sum_{r,h} q(r,h)\cdot q(T_t)\cdot ln\frac{P(I_t|T_t,r,h,f,b)}{q(r)\cdot q(h)\cdot q(T_t)} \quad (6)$$

As its name suggests, an EM algorithm consists of two steps. In the generalized EM algorithm, F is optimized with respect to the q distribution in E step and with respect to the model parameters in the M step until it finally converges. We briefly illustrate the EM algorithm as follows:

1. Given a guess of model parameter θ, we lower-bound the objective function $F(\theta)$ with a function $G(\theta, q)$
2. Find q distribution that maximize the lower bound. This is the Expectation-step.
3. Maximize the lower bound with respect to the parameter θ. This is the Maximization-step.
4. Go to 2 until it converges.

The EM algorithm is guaranteed to converged to a local optimum on a given input. There may be multiple local optima, and only one of these need to be a global optimum. Our experiments suggest that such local optima can be effectively reduced with good initialization and random restart.

4.2. E step

In the E step, the q distributions are optimized. Because the mapping from 3D cylinder to 2D image is a nonlinear transformation, a closed form update of ϕ_r and ϕ_h can not be found. We instead used gradient ascent to do the optimization:

$$\phi_r^{(new)} = \phi_r^{(old)} + \lambda\cdot\frac{\partial F}{\partial \phi_r^{(old)}} \quad (7)$$

$$\phi_h^{(new)} = \phi_r^{(old)} + \lambda\cdot\frac{\partial F}{\partial \phi_r^{(old)}} \quad (8)$$

where λ is the learning rate. To deal with the local maximum problem, random restart is applied. Given the estimation of ϕ_r and ϕ_h, we are able to compute the shape mask $m(r,h)$, which is a function of r and h.

$q(T_t)$ is updated by setting the derivative of D with respect to $q(T_t)$ to 0. Here we also apply a Lagrange multiplier to ensure that $\sum_{T_t} q(T_t) = 1$:

$$q(T_t) = \frac{1}{N_T}\exp\{-\frac{1}{2\sigma_o^2} [m_{T_t}(r,h)\cdot(I_t - f_{T_t})^2 + (1-m_{T_t}(r,h))\cdot(I_t-b)^2]\} \quad (9)$$

where m_{T_t}, f_{T_t} are the shape mask m and the foreground appearance f transformed by T_t.

4.3. M step

In the M step, the model parameters are updated given the estimation of q distribution from the E step. The update for background layer appearance is:

$$b = \frac{\sum_t q(T_t)\cdot(1-m_{T_t}(r,h))\cdot I_t}{\sum_t q(T_t)\cdot(1-m_{T_t}(r,h))} \quad (10)$$

The appearance of foreground layer is updated as:

$$f = \frac{\sum_t q(T_t)\cdot m_{T_t}(r,h)\cdot T_t(I_t)}{\sum_t q(T_t)\cdot m_{T_t}(r,h)} \quad (11)$$

where $T_t(I_t)$ is the observed image I_t transformed by T_t.

In M step the appearance and mask are updated to their expected value under q distribution computed in E step.

Fig. 2. The first two rows are the training images, the bottom image is the learnt object appearance in a 2D unfolded image.

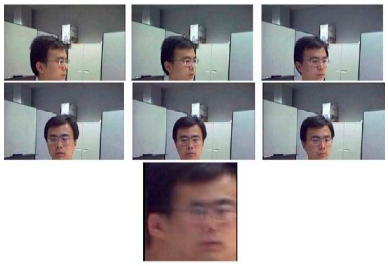


Fig. 3. The first two rows are the training images, the bottom row is the learnt face appearance in a 2D unfolded image.

5. EXPERIMENTAL RESULTS

Our experiments are carried out in the following way. The videos are first downsampled to 3 frames per second. The dynamic face-layer is initialized by the difference of fist and fifth frames. The background layer is initialized by the average of all the frames.

5.1. Extract the texture of a 3D object

In this experiment we use 65 frame in total. The sequence has a can rotating and translating in a static background scene. Our algorithm successfully extract the texture of this object. The object appearance is shown in Fig 2.

5.2. Modelling the face texture

In this experiment the proposed algorithm is applied to face video. The training data includes 35 image frames in which the face rotated around 90 degree angle. The learnt appearance model is presented in Fig 3.

5.3. Computational complexity comparison with ordered and unordered frame sequence

We compare the computational complexity of the proposed algorithm given naturally-ordered video frames and shuffled video frames. The computation is reduced by 93% for 'can' sequence and 96% for 'face' sequence if the video sequence is naturally ordered. This is because in that case the proposed algorithm can utilize motion prior to reduce the parameter space in EM.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we extend the flexible sprite model [4] and TMG model [7] by introducing dynamic face-layers in the task of extracting face textures from videos. The learning and inference is performed by variational EM algorithm. Our experiments show the proposed algorithm is capable of extracting the textures of faces undergoing 3D motions out of the scenes from video sequences.

Future work may include building a more generalized dynamic 3D layer model with flexible geometry, developing robust algorithms for non-rigid motion and illumination change.

7. REFERENCES

[1] Thomas Vetter and Tomaso Poggio, "Linear object classes and image synthesis from a single example image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 733–742, 1997.

[2] Volker Blanz and Thomas Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1063–1074, 2003.

[3] Nebojsa Jojic, Nemanja Petrovic, Thomas S. Huang, and Brendan J. Frey, "Transformed hidden markov models: Estimating mixture models of images and inferring spatial transformations in video sequence.," in *CVPR*, 2000, pp. 2026–2033.

[4] Nebojsa Jojic and Brendan J. Frey, "Learning flexible sprites in video layers.," in *CVPR (1)*, 2001, pp. 199–206.

[5] Brendan J. Frey, Nebojsa Jojic, and Anitha Kannan, "Learning appearance and transparency manifolds of occluded objects in layers.," in *CVPR (1)*, 2003, pp. 45–52.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society series B*, vol. 39, pp. 1–38, 1977.

[7] Brendan J. Frey and Nebojsa Jojic, "Transformation-invariant clustering using the em algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 1, pp. 1–17, 2003.

[8] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul, "An introduction to variational methods for graphical models," *Mach. Learn.*, vol. 37, no. 2, pp. 183–233, 1999.