Sequence Based Face Characterization Using Factorized Feature Points

Xiaozhou Wei *Lijun Yin*
Department of Computer Science, SUNY at Binghamton, Binghamton, NY 13902
{xwei, lijun}@cs.binghamton.edu

Abstract

In this paper, we proposed a face tracking and reconstruction scheme for face representation, human computer interaction, and the application of security. We applied a CDOF (Color Distribution Based Optical flow) approach to track the feature points in a facial video sequence. The extracted feature points sequence are then used as an input to derive their 3D coordinates using a factorization based algorithm. In combining with our model based expression generation approach, a facial expression from a 2D input can be reconstructed as a 3D output. The feasibility, limitation and future development of the proposed scheme are discussed through the experimentation.

1. Introduction

Human-computer interaction requires automatic reading of human face to obtain expression information. Human facial expression contains both shape and intensity information, we expect to extract shape data from well observed facial features. As surveillance video of human is easier to obtain while human face shape information needs costly hardware to capture, various algorithms have been developed to extract 3D information from available 2D information, such as intensity information from single or multiple images. In this paper we studied different feature tracking and structure recovery algorithm and designed a face expression interface system based on our study.

1.1. Structure from Motion

In vision applications SFM (Structure from Motion) algorithms are widely used to reconstruct 3D shape of objects with tracked 2D information through a sequence of image frames. Among SFM algorithms, factorization method [7, 9] has been successfully used to construct a matrix with tracked 2D feature coordinates and decompose it into motion and shape matrix. Applying the Gaussian distribution and EM algorithm [2], non-rigid motion and shape can be simultaneously estimated. In order to reconstruct 3D coordinates of the 2D points, a series of feature data should be tracked in a time-varying fashion. Feature tracking [1, 5, 10, 11] is to find specified features in a sequence of motion frames. Among feature tracking algorithms, optical flow is widely used to find point matching when deformation is not obvious. Meanwhile color distribution method is robust to match features but has lower accuracy. A combined tracking method [8] is expected to overcome the disadvantage of each and gain both robustness and localization accuracy.

1.2. Structure Recovery by Other Methods

An analysis by synthesize estimation method is used in [3, 4, 16] where 3D model is reconstructed from single image. The estimate is achieved by fitting a statistical, morph-able model of 3D faces to intensity images. The model is learned and builds from a set of textured 3D scans of heads and can be .reconstructed by varying a sequence of parameters. Various fitting schemes have been developed based on this algorithm

The limitation of these methods is the model fitting is very computation intensive and takes long time to converge. Also the construction of model needs a large dataset while the final fitting result hasn't been compared to ground truth to prove its correctness.

1.3. Proposed Scheme

We propose a system to track multiple features on human face from a video input and synthesize the same expression on any given face models. In this system, the facial features in the areas of eyes, eyelids, mouth, nose and face contour are tracked. After building correspondence between feature points in videos and vertices on a 3D face model, we are able to generate similar expression on another face by animating the face model. The proposed system can be used for facial gesture interface for expressions, lip reading, speech learning and communications, and be further used

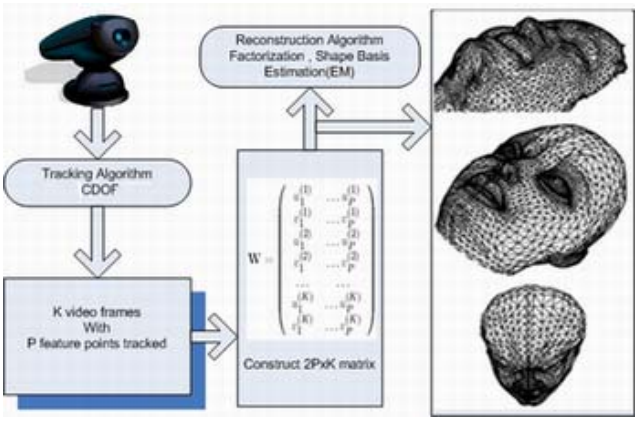

Figure 1: 2D tracking to 3D reconstruction of human face

as a non-intrusive expression reader to assist emotion identification. The 2D to 3D scheme is shown in Figure 1, where we extract 3D face shape from tracked video. Each components of the system will be described in the subsequent sections.

2. Facial Feature Tracking

For most of vision applications, finding corresponding feature points through a sequence of image frames is always the first essential part. Many algorithms have been developed for this purpose. There are intensity based methods such as KLT algorithm, correlation matching and optical flow based approaches. Also there are model or parameter based method for tracking specified objects, for example using adaptive appearance model to track known objects such as human body and face. The second group of methods usually has limited application scope as it is difficult to build database or parameter set for arbitrary orientation of tracking objects. Thus we are more interested on general purpose tracking methods which utilize pixel based information only.

2.1. Color Distribution Based Optical Flow

A Color distribution-based matching, such as the correlation matching, is effective but has relative poor localization accuracy. On the contrary, optical-flow based registration has certain accuracy but is noise sensitive and requires very close alignment, which means it is not robust when there is distortion between image patches. Motivated by the work done in [8], we applied a combination strategy to tracking facial features over time. We combined the advantages of color distribution method and optical-flow method so that the point matching can be both robust and accurate. To do so, a uniform parameterization is needed for the late transformation and combination. In color image the information can be presented in a five dimensional space, including the two spatial coordinates and the three components of color representation, such as RGB or LUV channels. In optical flow method, mapping error is measured between reference patch in one image and transformed patch in another image [8], which can be described as

$$e_{of}(x_i,\theta) = I_2[m(x_i,\theta)] - I_1(x_i) \qquad (2.1)$$

θ is an eight-dimensional vector which contains spatial coordinates, rotation angles, scaling factors and projective deformations. To obtain an updating rule for current estimate θ, a Taylor expansion of error is used

$$e_{of}(x_i,\theta') \approx I_2(m_i) + J\delta\theta - I_1(x_i) \qquad (2.2)$$

J is a Jacobian matrix containing elements as the derivatives of each color component with respect to each transformation parameter. To estimate θ the error is minimized over all pixels in the neighborhood centered at x_0. Using a generalized M-estimation method the optimization criterion can be expressed as:

$$\Gamma_{of} = \sum_{i=1}^{n_x} K_e\left(\frac{x_i - x_0}{h}\right)\rho\left(\left\| \frac{J\delta\theta + e_{of}}{\sigma} \right\|\right) \qquad (2.3)$$

where *h* represents the spatial bandwidth and σ is the scale of the color error, K_e is the radial symmetric Epanechnicov kernel and ρ(u) is the bi-weight loss function. Color information of neighborhood in first image will be described by a discrete color density distribution p. B is a

sample of color space. For example if there are 16 quantization steps, B has 4096 (16^3) samples. The distribution p is derived from the data using kernel density estimation. For each bin the density is accumulation of elements that are proportional to c_i-c_u, multiplied by a constant. c_i is the color of pixel x_i, and c_u the color associated with the uth bin in B.

To make the color distribution dependent on the image deformation, contribution of each pixel's color is weighted dependent on pixel location also. Thus an elliptically symmetric kernel can be used. Based on these derivations the color distribution of the pixels m_j in the neighborhood N_{m0} centered on m_0 in the second image are defined as:

$$q\ (\theta) = C_q \sum_{j=1}^{n_m} K\ (\frac{x(m_j,\theta) - x_0}{h}) K_e(\frac{c_j - c_u}{b}) \qquad (2.4)$$

C_q is normalization constant. Therefore the error between two components of the color distribution will be square root of p subtracted by square root of q.

A least squares minimization is then equivalent to maximizing the correlation coefficient between the two distributions. Consider the asymmetry nature of the task, the optimization criterion of color distribution is defined as

$$\Gamma_{cd} = \sum_{\beta} \sum_{u \in \beta} \rho_{cd} (\frac{g^T \delta\theta + e_{cd}(u,\beta,\theta)}{\sigma \pm}) \quad (2.5)$$

As both color distribution and optical flow method use the same five-dimensional space, the kernels and loss functions both have normalized, it is easy to integrate Equation (2.3) and (2.5) into one by calculating the linear summation of two criterions. More detailed theory and derivation can be obtained from [8].

Given two frames and coordinates of features which need to be matched in first frame, by minimizing the combined optimization criterion across two frames we will find the coordinates of moved features in the second frame. Repeat this procedure we will have a continuous tracking of feature points through a number of frames. To enhance the accuracy, tracking can be done not only on neighbor frames but also frames separate from each other for more than one frame. Tracking results can then be compared and combined to get an accurate result

3. 3D Coordinates Derivation

After obtaining the tracked feature vectors, we can further derive their 3D coordinates for later model adaptation. A number of approaches have been developed to solve the structure from motion problem. The factorization method is famous for its ability to recover both motion and shape of rigid objects under orthographic projections. Extensions have been developed for this method to use multi-body factorization, scaled orthographic projection or weak-perspective model [9, 7] to recover non-rigid motion information. For instance Lorenzo et al [2] developed an algorithm which is based on factorization theory. It uses Gaussian distribution to represent object shape in varied time instant, thus shape motion can be modeled as rigid component combined with a non-rigid deformation. EM (Expectation Maximization) algorithm is then used to simultaneously estimate 3D shape and motion of tracked objects at each time frame. We extend this approach for our unique application for 3D facial features reconstruction.

3.1. Factorization method

Suppose for each frame the object could be described as a key-frame basis set as S_1, S_2,... S_k. S_i is a 3xP matrix and P denotes the points in each frame. A shape thought K frames can be represented by a linear combination [9]:

$$S = \sum_{i=1}^{N} l_i \cdot S_i \qquad S_i \in R^{3\times P}, l_i \in R \qquad (3.1)$$

Under orthographic projection the P points can be projected into 2D:

$$W = R \cdot (\sum_{i=1}^{N} l_i \cdot S_i) + T \qquad (3.2)$$

where T, R is the 3D camera motion matrix. Assume S is centered at origin we subtracting the mean of each 2D point so that T can be eliminated. Track through K frames and P points we will have a 2KxP matrix W.

We can prove that W has rank 3N and can be factored into two matrices where Q contains the pose in each time frame and B has the key-frame basis shapes. The factorization can be done using singular value decomposition:

$$W^{2K\times P} = U \cdot D \cdot V = Q^{2K\times 3N} \cdot B^{3N\times P} \qquad (3.3)$$

With continuously tracking features we can get the shape of rigid object, while expression changing of human is non-rigid motion. Thus factorization method needs to be improved to obtain shape from non-rigid motion.

3.2. Non-rigid feature reconstruction using Gaussian distribution and EM Method

The problem of interpreting non-rigid shape and motion can be formulated quite similar to the previous section, i.e.,

$$P_t = R_t(S_t + D_t) + N \quad (3.4)$$

where P is the projected points matrix, R and D denotes the rotation and translation. S is the object shape matrix and N is zero-mean Gaussian noise with variance σ^2. The rigid motion of the object and the rigid motion of the camera are interchangeable. The goal is to estimate the time-varying shape S and motion from the observed projection P. Therefore to model non-rigid deformations, this method [7] assumes that the shape is produced by adding deformations to a shape average:

$$S_t = \bar{S} + \sum_{k=1}^{K}(V_k m_{k,t}) \quad (3.5)$$

where m is scalar weights that indicate the contribution of the deformations to each shape. S and V are referred to as the shape basis. The problem then become how to simultaneously estimate the right motion and shape PDF, that is, estimate R, D, θ, σ^2 to maximize this probability density function.

The shape and motion can be estimated while learning the parameters of the PDF $p(S|\theta)$ over shapes [7]. A vectorized form of first equation is

$$\begin{aligned} &vec(P_t) = vec(R_t S_t + R_t D_t + N_t) = \\ &M_t z_t + \bar{f}_t + T_t + vec(N_t) \end{aligned} \quad (3.6)$$

where $M_t=[\text{vec}(RtV1),...,\text{vec}(Rt\ V_k]$, $z_t=[z_{\{1,t\}},...,z_{\{K,t\}}]^T$, $f_t=\text{vec}(R_t\ S)$. The marginal distribution over shape is a Gaussian.

One parameter in Gaussian presentation here contains the model parameters S, V_k, R_t, D_t, σ^2. With the definition and derivation we had, a generalized EM (expectation maximization) algorithm is derived to estimate the motion and deformation model given a set of point tracks P (or f).

Two steps are involved in the EM algorithm. In the E-step for each frame t given the current motion and shape estimates the distribution over z_t can be estimated, which can be computed afterwards.

In the M-step, the motion parameters can be estimated by minimizing an energy function. To update the parameters, corresponding partial derivative of the expected log likelihood need to be computed. Parameter update rules have been set for shape basis, noise variance, translation and rotation.

By recursively execute E-step and M-step we will finally reach a solution which gives the current shape estimation. Therefore shape of objects having non-rigid motion of facial features in certain expressions can be derived by this method.

4. Model Instantiation Using Reconstructed 3D Features

The amount of feature points that can be tracked is depends on the quality of images and computation cost. In experiments we use a single video camera to capture the facial sequence. Apparently the number of points which are tractable under this circumstance is limited. A more efficient algorithm which can track more points with higher accuracy could greatly help to reconstruct the 3D models with better details.

Currently we have discussed about 3D reconstruction of certain tracked points only. However, the face shape representation requires the estimation of all the points in the face region in addition to a few of feature points. In order to reconstruct the whole face region in a 3D space, the modification-based approaches [3, 4, 13] could be applied by using existing face models with our limited tracking points(e.g., morphable models). Here we use our developed 3D generic model fitting algorithm to infer the 3D coordinates of the non-feature points. We use a rule based interpolation strategy to derive the 3D position of all points on the generic model, given the 3D feature points being reconstructed. And finally, the individualized model is generated to represent the 3D facial expression of the individual person frame by frame [14].

5. Experiments and Discussion

Video sequences showing single person's frontal head-and-shoulder are used in our experiments. Figure 2 shows four sample frames with tracked features from a sample human face expression video. The features are

selected from regions which can reflect 3D shape of human face more obviously, such as eye corner, nose top, outline of mouth and face. They are marked with black dots in each frame as shown below.

In the experiment, we define the tracking failure as the tracked point distracts from the original position by 5 pixels away. We found that the tracked feature points gradually eliminate with respect to time. Especially when the motion is obviously large and the deformation are relatively strong. We have a few failure cases when face has quick movement or large rotation. To solve this problem, we select feature points that have higher probability of tracking through the frame sequences and use it as clue to infer motion of other feature points.

Figure 2: Example of tracking feature points from images of CMU face expression database [15]

From 2D projection of 3D points on a face model, we construct the projection matrix and apply the factorization algorithm to derive the 3D coordinates of each tracked points. By factorizing the constructed matrix, we obtain both the motion matrix and the shape matrix. Figure 3 shows one 3D reconstructed example with the recovered shape matrix. Due to image quality and error from factorization calculation, the z axis value shows errors which are parallel to the ground truth z value. A study has been done to reduce this error and facilitate with 3D reconstruction.

Furthermore, in order to reconstruct the 3D coordinates of feature points effectively, it is necessary to track a large number of frames. Insufficient number of frame will not provide enough parameters to successfully factorize shape and motion information. In the experiments, we tracked 200 frames for reconstructing 122 3D feature points for each subject. The reconstructed 3D points exhibit the basic shape of the individual person's face. Then we use these data to drive a 3D model to create a complete individualized facial model. Note that based on the detected feature points, we can further transfer the 3D feature points to different person's facial model, and achieve the expression mapping from one subject to the other (Figure 4, detailed feature transferring algorithm can be found in [14]). Currently, this procedure runs off-line due to the calculation cost; and we expect to generate dynamic facial expression on-line and in real time with Gaussian basis shape and EM algorithm for non-rigid deformations in the future.

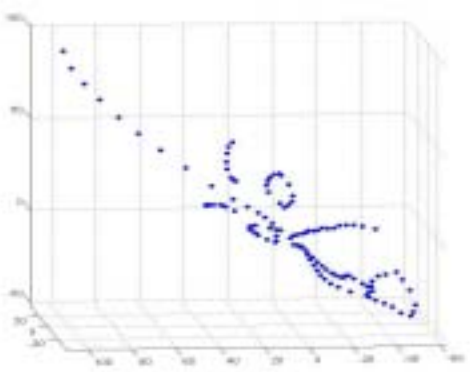

Figure 3: Reconstructed 3D face points

Figure 4: face expression mapping with interpolation

6. Limitation and Future Work

The model instantiation is greatly affected by the feature tracking results. Currently, the feature tracking is still in its preliminary stage. In the large motion situation, the algorithm loses the track of the correct feature positions. Future work could be to improve the tracking algorithm using the statistical based analysis algorithm by exploring the face 3D surface features. In addition, we will evaluate the accuracy of 3D reconstruction results by comparing them with the ground true data in the future.

7. Acknowledgement

We would like to thank the NSF and the NYSTAR for the support, and thank Dr. Qiang Ji and Dr. Zhiwei Zhu of RPI for the support and discussion of this work.

References

[1] C. Tomasi and T. Kanade, “Shape and Motion from image streams under orthography: a factorization method”, *IJCV1992.*
[2] L. Torresani, A. Hertzmann and C. Bregler, “Learning Non-Rigid 3D Shape from 2D Motion”, *NIPS 2004*.
[3] S. Romdhani and T. Vetter, ”Efficient, Robust and Accurate Fittingof a 3D Morphable Model”, ICCV, 2003
[4] V. Blanz and T. Vetter, ”A Morphable Model for the Synthesis of 3D Faces”, *SIGGRAPH*, 1999.
[5] C. Tomasi and T. Kanade, ”Detection and Tracking of point features”.*Carnegie Mellon University Technical Report*, (CMU-CS-91-132), April 1991
[6] Jean-Yves Bouguet, ”Pyramidal Implementation of the Lucas Kanade Feature Tracker description of the algorithm”. *OpenCV Documents, Intel Corp.*, 1999
[7] C. Bregler, A. Hertzmann and H. Biermann, ”Recovering Non-rigid 3D Shape from Image Streams”. *IEEE CVPR2000.*
[8] B. Georgescu and P. Meer, ”Point Matching under Large Image Deformations and Illumination Changes”. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, June 2004.
[9] J. Xiao, J. Chai,and T. Kanade, ”A Closed-Form Solution to Non-rigid Shape and Motion Recovery”. *IEEE CVPR2004.*
[10] J. Shi and C. Tomasi, ”Good Features to Track”. *IEEE Computer Vision and Pattern recognition*, 1994.
[11] T. Marks, J. Hershey, J. Roddey, and J. Movellan, ”3D Tracking of Morphable Objects Using Conditionally Gaussian Nonlinear Filters”. *Generative Model Based Vision Workshop on IEEE CVPR04*, 2004.
[12] J. Noh and U. Neumann, *Expression Cloning*, SIGGRAPH01,pp277-288.
[13] D. Terzopoulosa and K. Waters, ”Analysis and Synthesis of Facial Image Sequences Using Physical and Anatomical Models.” *IEEE Trans. PAMI*, 15(6), 1993
[14] X. Wei, Z. Zhu, L. Yin, and Q. Ji, *"A real time face tracking and animation system"*, First IEEE Workshop on Face Processing in Video, in conjunction with IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'04).
[15] http://vasc.ri.cmu.edu/idb/html/face/facial_expression/index.html
[16] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Trans .Pattern Anal. Mach. Intell.*, 2003.