Grid Particle Filter for Human Head Tracking Using 3D Model

Chenguang Liu [1] Jiafeng Liu [2] Jianhua Huang [3] Xianglong Tang[4]

[1]liu.cg@live.cn [2]jefferyliu@hit.edu.cn

Abstract

A new 3D head-shoulder model based particle filter is presented for finding human head in static images. Edge cues are used as the likelihood function of the proposed particle filter. The positions of head as well as its direction are evaluated simultaneously. At each time step, the proposed algorithm generates a discretized grid state space and takes a subspace of it as the searching space of the particles. Hence the number of particles is well limited and the performance of the particle filter is improved. The experimental results prove that our method is effective and robust.

Keywords: head tracking; face direction; 3D model; grid particle filter

1. Introduction

Monocular sequence based human tracking and analysis is promising for many applications, hence a number of researchers are interested in this field. The human head tracking is important in this researching field since it provides preconditions for the whole human body tracking, human face analysis, face recognition and so on.

Many algorithms for human head tracking use face characters to detect the position of head [1, 2]. Admittedly, the face is always bare and the characters are stable for tracking. But in many important applications such as surveillance, face is not available at any time. While there are also some algorithms take into account the contour of head or head-shoulder structure for head tracking [3, 4, 5, 6]. Jin *et al.* [3] fulfilled a method based on ellipse matching for human head tracking in complicated background. Yoon *et al.* [4] proposed a ‘Ω’ model and fulfilled a head tracker. However, both ellipse model and ‘Ω’ model are simple and liable to be affected by noises. Moreover, they do not provide 3D information and can not be used to judge the direction of head.

In this paper we propose a 3D head-shoulder model. This model is similar to the 2D model presented in [4, 5, 6]. But our 3D model is closer to the true structure of the human body and contains more 3D information; therefore the proposed model can better represent the deformation of the projection of the head-shoulder model. Besides, a grid particle filter is presented, which can not only track the position of head but also detect the direction of the face. The tracking result is compared to the literature [5].

2. Particle Filter

The human head tracking problem can be treated as a Bayesian problem. At time step *t* the state of head is represented by X_t, thus the probability framework of the head tracking problem is expressed as

Proceedings of the 11th Joint Conference on Information Sciences (2008)
Published by Atlantis Press

$$p(X_t \mid I_t) = cp(I_t \mid X_t)\int p(X_t \mid X_{t-1})p(X_{t-1} \mid I_{t-1})dX_{t-1} \quad (1)$$

where I_t is observation template of the image, c is a normalization constant, $p(X_t \mid I_t)$ is the posterior probability about I_t, $p(I_t \mid X_t)$ represents the likelihood measurement, $p(X_t \mid X_{t-1})$ is a dynamic model which predicts the next state of the object, $p(X_{t-1} \mid I_{t-1})$ is the prior probability which provide the prior information about the states of the head at time step $t-1$.

Generally, it's difficult to directly calculate the integral in (1). While particle filter [7] is a method to evaluate the posterior from a set of non-Gaussian and multi-model observation data. The main idea of the particle filter is to represent the posterior density through a set of weighted particles. Therefore, the calculation of the integral is avoided by using particle filtering.

The particle filtering can be treated as a process of selecting 'the most optimized particles' from an optimized particle set. A typical particle filter consists of three major steps at time step t: (i) weighted sampling process, during which N particles are selected on the distribution of the posterior at time step $t-1$. The possibility of selecting a particle is decided by its normalized weight; (ii) dynamic process, during which a dynamic model is used to update the N selected particles of which the states are predicted; (iii) measurement process, during which the observations of the current time step are obtained, and the weight of each of the particles is calculated by likelihood measurement.

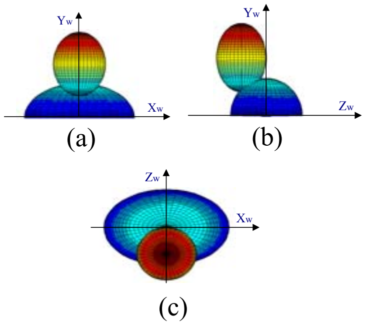

Fig. 1: Head-Shoulder model (a) Frontal View (b) Side View (c) Upper View

2.1. Head-Shoulder 3D Model

We construct a head-shoulder model $\psi(\nu, \theta_y, \theta_z)$ (Fig. 1) using two connected ellipsoids, of which the parameters are calculated by the proportion of the human body. The model ψ has three DOFs: ν represents the scale, θ_{y} represents the scrolling angle around the coordinate Y_W, and θ_{z} represents the scrolling angle around the coordinate Z_W.

The relation between the world coordinate space W and the image coordinate space F is shown in Fig. 2, and the plane $\mathrm{X_wO_wY_w}$ in W and the plane $\mathrm{X_FO_FY_F}$ in F are parallel with each other.

Now, we project the 3D model $\psi(\nu, \theta_y, \theta_z)$ to the image plane and then obtain the 2D edge template $\kappa(x, y, \nu, \theta_y, \theta_z)$, where x and y represent the coordinates of template κ in the image plane, ν, θ_y, θ_z represent the three DOFs of the 3D model ψ corresponding to κ.

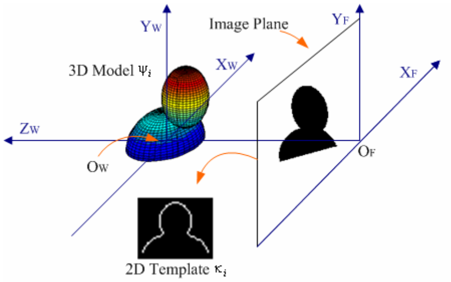

Fig. 2: The relation between the world coordinate system W and the image coordinate system F

The particle in 5 dimensional state space is defined as $\phi\left(x, y, \nu, \theta_y, \theta_z, \omega\right)$, where ω represent the weight of the particle, vector $\left(\nu, \theta_y, \theta_z\right)$ corresponds to a 3D model ψ, and (x, y) represents the position of the 2D template in the image plane. The coordinate of the head in the 2D template image is calculated beforehand; therefore, when the position of the 2D template is evaluated the position of the head in the image plane can be calculated consequently.

By tuning the value of the three DOFs of model $\psi(\nu, \theta_y, \theta_z)$, we can obtain various configurations of the model. The set of different configurations consists of a 3D model database S. Given that the sample rate of ν, θ_y and θ_z are $\Delta\nu$, $\Delta\theta_y$ and $\Delta\theta_z$, respectively. The model database S is represented as

$$\begin{cases} \mathrm{S} = \left\{ \begin{array}{l} \psi_i(\nu^i, \theta_y^i, \theta_z^i) \mid \nu^i \in \mathrm{P}, \theta_y^i \in \mathrm{G}, \theta_z^i \in \mathrm{Q} \\ i = 1 \ldots N \end{array} \right\} \\ \mathrm{P} = \left\{ \nu \mid \nu = \nu_0 + (i-1)\Delta\nu, i = 1 \ldots N_{\mathrm{P}} \right\} \\ \mathrm{G} = \left\{ \theta_y \mid \theta_y = \theta_{y,0} + (i-1)\Delta\theta_y, i = 1 \ldots N_{\mathrm{G}} \right\} \\ \mathrm{Q} = \left\{ \theta_z \mid \theta_z = \theta_{z,0} + (i-1)\Delta\theta_z, i = 1 \ldots N_{\mathrm{Q}} \right\} \end{cases} \qquad (2)$$

where P, G and Q are the sets of the corresponding DOFs, respectively. There are $N = N_{\mathrm{P}} \times N_{\mathrm{G}} \times N_{\mathrm{Q}}$ head-shoulder model in S. The model database S optimized the searching space of the particle filtering. The state space could be discretized to the grid state space based on S.

2.2. Grid particle Filter

The proposed grid particle filter discretize the 5 dimensional state space into a grid state space which is used as the searching space of particle $\phi\left(x, y, \nu, \theta_y, \theta_z, \omega\right)$. Each node in the discretized state space corresponds to a particle. Thus, the continues state space is discretized to a grid state space and the 'discretizing grade' depends on the sample rate on each dimensions. The bigger the sample rate is, the more efficient the performance is, the smaller the precision of evaluating the configuration is, or vice versa.

The proposed method does not need manually initialization. The grid particle filter contains three major steps at time step t: (1) Weighted sampling process: selecting *M* particles with biggest weights from the particle set Φ_{t-1} which is generated at time step $t-1$; (2) Prediction process: after the tracking process at time step $t-1$, the mean value $\bar{\phi}_{t-1}\left(\bar{x}, \bar{y}, \bar{\nu}, \bar{\theta}_y, \bar{\theta}_z, \omega\right)$ is evaluated. As a matter of fact, the difference between two continues frames is very small, therefore, we use $\bar{\phi}_{t-1}$ to predict the particles at time step t. In the 5 dimensional grid state space, a super ellipsoid Ω is obtained and its center is $\bar{\phi}_{t-1}$. Then the N_t particles surrounded by the ellipsoid in the grid state space are predicted new particles. The volume of the ellipsoid can decide the value of N_t but the volume is decided by the width of each dimension. In order to increase the computing effectiveness, we limit the number of particles by setting reasonable value of the width of each dimension. We use (3) to go through the N_t new particles.

$$\begin{cases} \eta = \overline{\eta} - w + i\triangle\eta \\ \eta \in \left[\overline{\eta} - w,\ \overline{\eta} + w\right] \end{cases} \quad (3)$$

where η represents the value of one dimension on which the width of the super ellipsoid Ω is $2w+1$, $\triangle\eta$ represents the interval between each pair of particles in the discretized state space.

Measurement process: each new particle $\phi_t\left(x, y, \nu, \theta_y, \theta_z, \omega\right)$ uniquely corresponds to a 2D template $\kappa_t\left(x, y, \nu, \theta_y, \theta_z\right)$. An observation template Z_t is the image region covered by κ_t. The likelihood between Z_t and the 2D model κ_t is calculated as the weight of the particle ϕ_t. After N_t new particles' weights have been updated, the particle set Φ_t is generated at time step t. By finding *M* particles with the biggest weights and calculating the mean value of them, the position of the head at the current time step is evaluated.

2.3. Likelihood measurement

Edge cue is used to construct the likelihood function which is shown as below:

$$\omega_e = \frac{N_c}{N_m} \quad (4)$$

where N_m represents the number of pixels on the edge of the human body in the 2D template image. $N_c = \sum_{i=1}^{N_m} f(i)$ represents the number of matched pixels comparing the 2D edge template image with the observation image, $f(i)$ represents matching result of the *i*th pixel on the edge of the 2D template. The matching process is defined as:

$$\begin{cases} f(i) = 0 & \text{if} \sum G(x, y) = 0 \\ f(i) = 1 & \text{if} \sum G(x, y) \neq 0 \end{cases} \quad (5)$$

where $G(x, y)$ represents the gray level of the point (x, y) in a window which is decided by the center point P_0'. Given the coordinates of the *i*th pixel on the edge of the 2D template (x_0, y_0), P_0' is the pixel at the same position on the observation edge image. The window's size is 3×3 pixels.

2.4. Detection of Face Direction

After the grid particle filtering, we obtain the state vector $\left(\overline{x}, \overline{y}, \overline{\nu}, \overline{\theta}_y, \overline{\theta}_z\right)$ of the head at time step *t*. Where vector $\left(\overline{\nu}, \overline{\theta}_y, \overline{\theta}_z\right)$ depict the 3D configuration of the head-shoulder structure and the face direction is represented by $\overline{\theta}_y$.

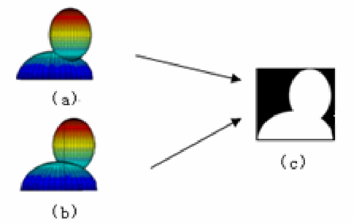

Fig. 3: The ambiguity of projection. (a) 3D head-shoulder model (face); (b) 3D head-shoulder model (back); (c) the projection of (a) and (b).

However, when projecting the 3D head-shoulder model to the image plane, the depth ambiguity will occur (Fig. 3). The head-shoulder model database doesn't have the model of back view ($\theta_y \notin \left[-90,\ 90\right]$). Therefore, when the body is facing back (Fig. 3(b)), the tracker will get a wrong tracking result (Fig. 3(a)).

As a matter of fact, the color difference between the face and the hair. Hence color could be used to distinguish the face and the back of head. The 2D S-V (Saturation and Value) color histogram $\tilde{H}_{face}$ and $\tilde{H}_{back}$ could be generated after the learning process beforehand. After the tracking process is finished, the position of head is calculated too. Then the histogram H_{head} of a region of the head is calculated based on its position. Bhattacharyya distance [8] is used to describe the distance between H_{head} and the two histograms $\tilde{H}_{face}$ and $\tilde{H}_{back}$. The face direction error is amended according to Eq. (6).

$$\overline{\theta}_y = \begin{cases} \overline{\theta}_y & \text{if } B(H_{head}, \tilde{H}_{face}) \le B(H_{head}, \tilde{H}_{back}) \\ 180 + \overline{\theta}_y & \text{if } B(H_{head}, \tilde{H}_{face}) > B(H_{head}, \tilde{H}_{back}) \end{cases} \quad (6)$$

where $B(H_{head}, \tilde{H}_{face})$ and $B(H_{head}, \tilde{H}_{back})$ represent the Bhattacharyya distances between the corresponding histograms.

3. Experimental Results

We perform our method on the HumanEva database [9] downloaded from Brown University. Several sequences, including walking, jogging and boxing, are used to test the proposed head tracking method.

When generating the 3D model database, the corresponding parameters are: $\nu \in [0.5,\ 2.0]$, $\theta_y \in [-90,\ 90]$ and $\theta_z \in [-20,\ 20]$; the corresponding sample rate is: $\triangle\nu = 0.1$, $\triangle\theta_y = 45$ and $\triangle\theta_z = 5$, hence totally 720 models are generated. When performing the prediction process of the grid particle filter, the values of the parameters on each of the dimensions are shown as Table 1.

η / Value	x_t	y_t	ν_t	$\theta_{y,t}$	$\theta_{z,t}$
Width w	$15\,\nu_{t-1}$ （Pixel）	$15\,\nu_{t-1}$ （Pixel）	0.2	45°	10°
Interval $\triangle\eta$	2（Pixel）	2（Pixel）	0.1	45°	5°

Table 1. The values of the dimensions when performing the prediction process at time step *t*.

The ground truth is annotated manually on the test sequences. The average tracking error is 6.97 pixels, and the detected face direction of the test sequences is shown as Fig. 4, Fig.5 and Fig. 6. The experiment is done on an Intel Core2 2.0 GHz PC and the average computation time is 0.41 second per frame. From the results we can see that our method is able to deal with the self occlusion problem. In literature [5] the average tracking error of head is 9.28 pixels and the time is more than 2 seconds per frame on an Intel-Xeon 2.8 GHz PC. Comparing to [5] on human head tracking, our method achieve a better and more effective result.

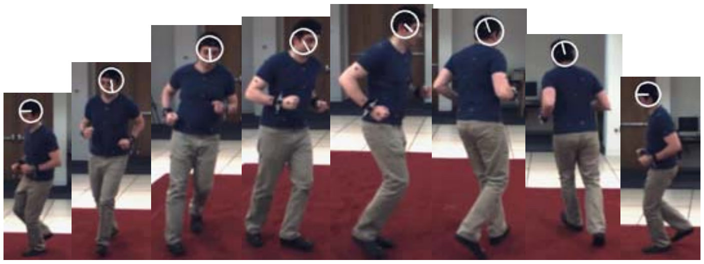

Fig. 5: Tracking result of the jogging sequence.

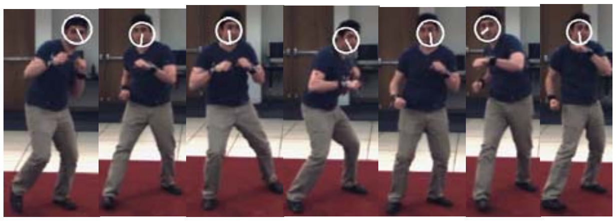

Fig. 6: Tracking result of the boxing sequence

4. References

[1] GangWang-Jian, Eng Thiam Lim, and Ronda Venkateswarlu, "stereo head/face tracking and pose estimation," Seventh International Conference on Control, Automation, Robotics, and Vision, Singapore, 2-5 December 2002 : 1609-1614.

[2] O Sileye, J Odobez, "A probabilistic framework for joint head tracking and pose estimation," ICPR of the 17th International Conference on Pattern Recoqnition, United Kingdom, 23-26 August 2004 : 264-267.

[3] Yonggang Jin, F Mokhtarian, "Data fusion for robust head tracking by particles," 2nd joint IEEE international workshop on VS-PETS, Beijing, 15-16 October 2005 : 33-40.

[4] Hosub Yoon, Dohyung Kim, Suyoung Chi and Youngjo Cho. A robust human head detection method for human tracking. Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on Oct. 2006 : 4558 – 4563

[5] Mun Wai Lee, Ram Nevatia, "Body part detection for human pose estimation and tracking," Motion and Video Computing, 2007. WMVC '07. IEEE Workshop on Feb. 2007 : 23 – 23

[6] Mun Wai Lee, I. Cohen, "A model-based approach for estimating human 3D poses in static images," Pattern Analysis and Machine Intelligence, IEEE Transactions on 2006, 28(6) : 905 – 916

[7] M A Isard, A Blake, "CONDENSATION conditional density propagation for visual tracking," International Journal of Computer Vision, 1998, 29(1) : 5-28.

[8] F Aherne, N Thacker and P Rockett, "The Bhattacharyya metric as an absolute similarity measure for frequency coded data," Kybernetika,1998, 34 (4): 363-368

[9] L Sigal, M J Black, "Humaneva: Syncronized video and motion capture dataset for evaluation of articulated human motion," Brown Univertsity TR, 2006.