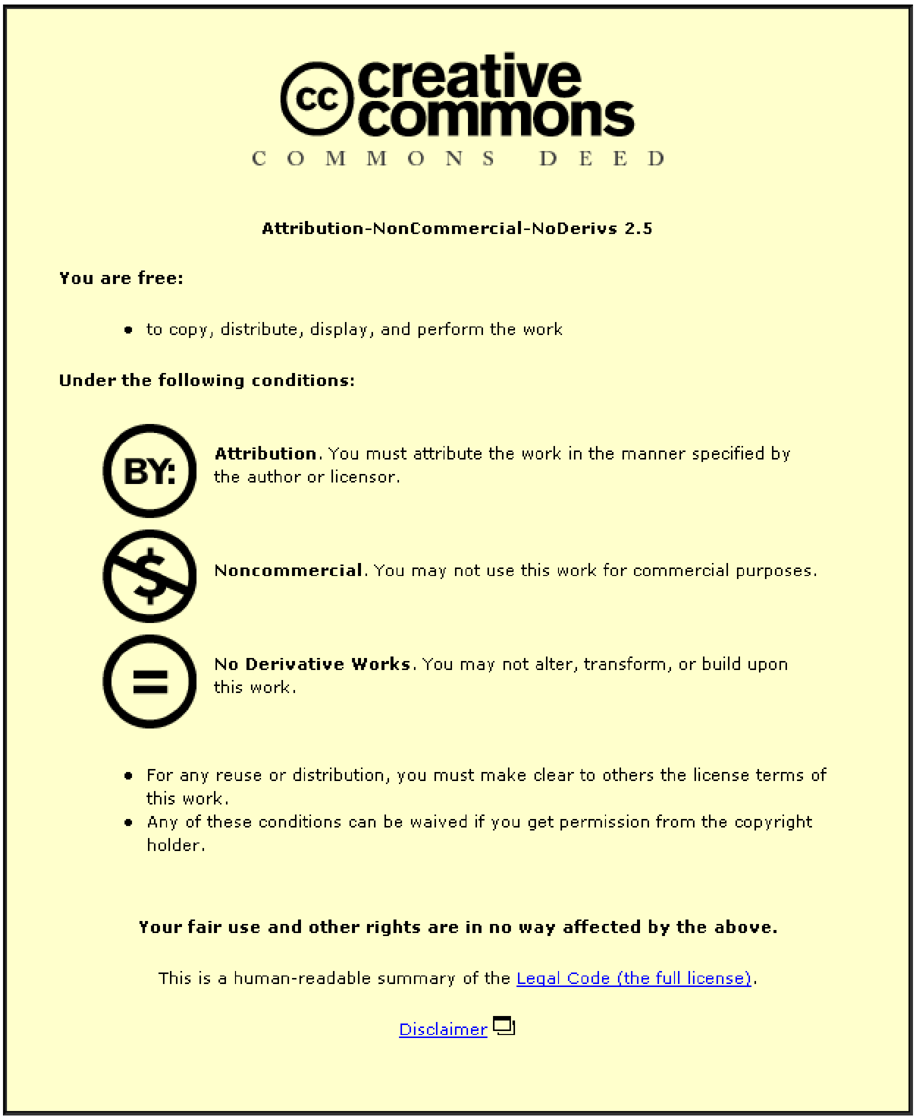


ICSP2006 Proceedings

Virtual Friend: Tracking and Generating Natural Interactive Behaviours in Real Video

Yue Zheng[1], Yulia Hicks[1], Darren Cosker[2], Dave Marshall[2], Juan C. Mostaza[3] and Jonathon A. Chambers[1]
[1]*School of Engineering,* [2]*School of Computer Science, Cardiff University, UK, CF24 3AA*
[3]*University of León, Spain*
E-mail: zhengy@cf.ac.uk

Abstract

The aim of our research is to create a "virtual friend" i.e., a virtual character capable of responding to actions obtained from observing a real person in video in a realistic and sensible manner. In this paper, we present a novel approach for generating a variety of complex behavioural responses for a fully articulated "virtual friend" in three dimensional (3D) space. Our approach is model-based. First of all, we train a collection of dual Hidden Markov Models (HMMs) on 3D motion capture (MoCap) data representing a number of interactions between two people. Secondly, we track 3D articulated motion of a single person in ordinary 2D video. Finally, using the dual HMM, we generate a moving "virtual friend" reacting to the motion of the tracked person and place it in the original video footage. In this paper, we describe our approach in depth as well as present the results of experiments, which show that the produced behaviours are very close to those of real people.

1. Introduction

In recent years, there has been a large amount of research in producing virtual worlds and populating them with virtual characters. There has been also a limited amount of research into enabling virtual characters with the ability to produce intelligent behaviour on the basis of visual analysis of the scene, which mainly was conducted in the computer vision area. Johnson et al. [1] developed a system capable of producing a 2D silhouette of a virtual person interacting with a real person in video and demonstrated it working with a handshake behaviour. Jebara et al. [2] developed a system capable of producing a dynamic human face together with speech reacting to the events around it. However, as Jebara states himself, the system exhibited only limited intelligent behaviour. Both of the above systems automatically learnt the intelligent behaviours from observed video data and represented them using HMMs [3], which are commonly used for representing temporal dynamics of the data. However, nobody has attempted to produce interactive behaviours for fully articulated 3D virtual characters until now.

In this article, we present a novel approach for generating intelligent behaviours for fully articulated 3D virtual characters on the basis of visual analysis of the motion of a real person in ordinary 2D video. To achieve the above goal, we learn a statistical model of the interactive actions between two people, from which we can derive an appropriate reacting behaviour for a virtual character given the motion of the real person.

The statistical model we apply here is a dual HMM[4], where the motion of each person is represented in its own state space, however, the transition probability matrix is shared, as explained in more detail in Section 3. Similar to Johnson [1] and Jebara[2], we learn the statistical models from real world data, however, unlike them, we use MoCap data representing articulated 3D motion of two interacting people.

To generate an intelligent behaviour for a virtual character, first of all, we need to analyse the motion of real person in video. In recent years, a variety of methods to extract 3D articulated motion of a person moving in video were developed [5]. The majority of these methods rely on some kind of a model of human body and/or motion. For tracking, many approaches used the CONDENSATION algorithm [6] or similar methods based on particle filters [7]. The problem with these approaches is the large number of particles required to track the motion, which grows exponenttially with the dimensionality of the search space. For a fully articulated motion of human body this produces unacceptably long processing times. Here, we use a method based on particle filtering, modified to avoid the high dimensionality problem, namely, annealed particle filtering [8].

The outline of the whole system can be seen in Figure 1. We outline the tracking part of the system in Section 2 and explain the behaviour generating part of the system in Section 3. The experiments together with evaluation are presented in Section 4.

0-7803-9737-1/06/$20.00 ©2006 IEEE

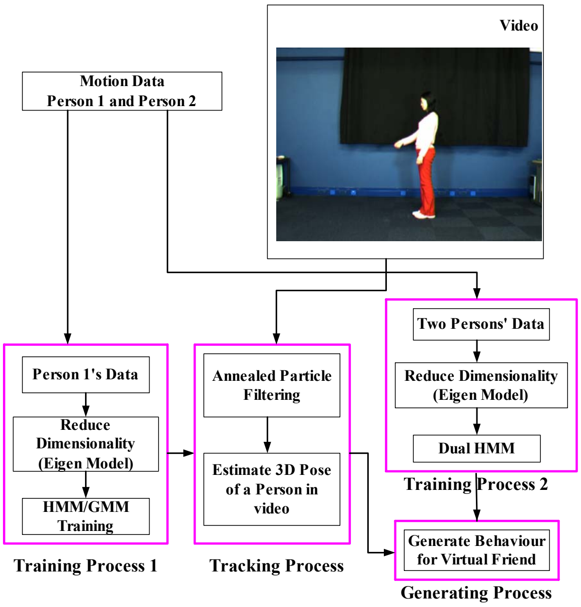

Video
Motion Data
Person 1 and Person 2
Person 1's Data
Reduce Dimensionality (Eigen Model)
HMM/GMM Training
Training Process 1
Annealed Particle Filtering
Estimate 3D Pose of a Person in video
Tracking Process
Two Persons' Data
Reduce Dimensionality (Eigen Model)
Dual HMM
Training Process 2
Generate Behaviour for Virtual Friend
Generating Process

Figure 1. Overview of the system

2. 3D Human Body Tracking in a Video

To track fully articulated 3D motion of a person in video from a single camera, we utilise a model of human geometry and a model of dynamics of human motion in conjunction with annealed particle filtering approach described in [8].

Our model of geometry of a human body is deliberately kept simple, however, it is sufficient for our purposes. It consists of 16 segments connecting 14 vertices positioned on the body and representing places like elbows, knees, etc. These segments are dressed in truncated cones. The silhouette of the produced model of geometry roughly resembles a human figure, which we need during the tracking process.

We represent the model of dynamics of motion of a single person in manner similar to the way we represent the interactive motion of two people, with an HMM, but this time we have only one state space. An HMM is defined as

$$\lambda = (A, B, \pi) \qquad (1)$$

where A = $\{a_{ij}\}$ is the state transition probability matrix, B =$\{b_j(o)\}$ where $b_j(o)$ is the observation density distribution at state j and π is the initial state probability distribution.

To train the model of dynamics, we use MoCap data. In our experiments we capture several sets of motion data in 3D space with 30 markers, therefore each pose is represented by a 90-dimensional vector. Such data is always constrained by physical and dynamical factors, thus we reduce its dimensionality using Principal Component Analysis (PCA), keeping 95% of the eigenenergy, and then train our HMM on a number of such vectors.

When tracking 3D pose of person in 2D video, we preprocess the video sequence by subtracting background [9] and thus obtain a sequence of binary images as shown in Figure 2. Next, we use annealed particle filtering together with HMM trained in the previous stage to estimate 3D poses of the tracked person in the video. In the experiments, we used 10 layers and 256 particles (samples) on each layer; both numbers were determined experimentally. The result of the tracking process is a sequence of 90 dimensional vectors, each estimating a 3D pose of the tracked person in the video.

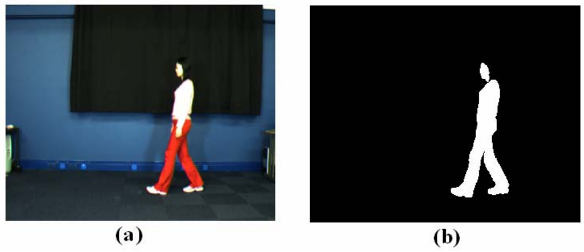


Figure 2. (a). Original image with person in the scene; (b). Binary image after subtracting background

3. Generating Behaviours for a Virtual Friend

This section describes the process of generating intelligent behaviours for a “virtual friend”. Here, we use an approach adapted from a method for generating a talking head [4].

3.1. Model of Interactive Behaviour

The model is trained on the 3D MoCap data of two real people. It is capable of representing a variety of interactive behaviours, such as, for example, shaking hands. Our model uses a dual HMM, with two sets of states. The first set of states models the poses for the first person (A), where each pose is represented by a 90-dimensional vector, and the other set of states models the poses for the second person (B). Each state in the model is modelled with a Gaussian. The first HMM is defined as:

$$\lambda_A = (T_A, B_A, \pi_A) \qquad (2)$$

where T_A is the state transition probability matrix, B_A is the observation probability distribution and π_A is the initial state probability distribution. We initialise the transition probabilities and the observation probabilities using random numbers, and then iteratively improve the estimates.

The second HMM is defined as

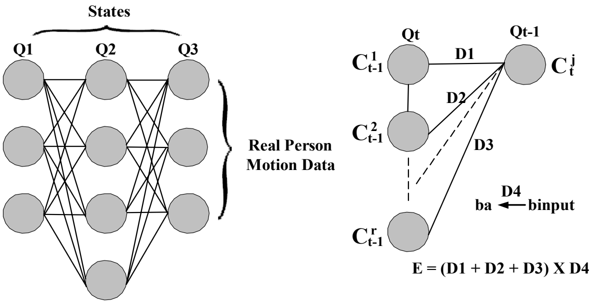
$$\lambda_S = (T_A, B_S, \pi_A) \qquad (3)$$

It has the same transition matrix as λ_A. The means and covariances of the Gaussians are calculated from the data set representing the motion of the second person.

Using this dual HMM, we can estimate a sequence of 3D poses for a virtual character given a sequence of 3D poses of the tracked person in video as described in the next section.

3.2. Generating Interactive Behaviours

Now it is possible, given as input a sequence of 3D poses of a person (tracking result from Section 2), to generate a corresponding sequence of poses for the “virtual Friend”. To achieve this goal, we defined a trellis data structure. Figure 3(a) illustrates an example of an initial trellis structure [4]. Each column of the trellis structure represents motion data of person B corresponding to the motion of our virtual character. We assign an error value (the Mahalanobis distance) to each element in each column, and then we work backward through the trellis to choose the motion data for the “virtual friend” with the lowest cost for each time *t*.



(a) Trellis Data Structure **(b) Errors for t = 1,...,M**

Figure 3. (a). Initial Trellis data structure for generating motion data. (b). Error Calculation for the generating motion data

Figure 3(b) shows how errors are calculated according to the trellis. C_t^j is the data vector for person B at time *t*, b_{input} is the new input signal (the tracking result from previous section) and b_a is the data vector for person A, which has the same location as the data vector for person B. For state 1 (*t = 1*) of the trellis structure, we only have the distance between the new input signal and person A data (denoted as D_4). For other states *t = 2, ..., M* (M is the number of frames of the new input signal), we calculate the distances between C_t^j in a column at time *t* and C_{t-1}^j in a column at time *t - 1* (denoted as D_1, D_2 and D_3). Thus, error can be obtained as $E = D_4$ for state 1 and $E = (D_1 + D_2 + D_3) \times D_4$ for other state. When errors are calculated, we work backwards through the trellis and choose the motion data for the “virtual friend” in each column with the lowest error at time *t*. Through this process, the interactive behaviour for “virtual friend” is obtained.

4. Results

In this section, we conducted three experiments, each with a different type of motion: two people shaking hands, one person pulling another person and one person pushing another person.

4.1. Experiments

In each experiment, our training data consisted of four MoCap data sequences with two people, totalling around 600 pose vectors. Each experiment consisted of the following steps.

1. Perform PCA on the data set representing the motion of one person, keeping 95% of the eigenenergy and project the data set in the reduced dimensionality eigenspace.
2. Train an HMM (to be used for tracking of a person in video) on that data set with a number of states ranging between 20 and 40, chosen experimentally.
3. Perform PCA on the data set representing the motion of two people, keeping 95% of the eigenenergy and project the data set in the reduced dimensionality eigenspace.
4. Train a dual HMM on the above data set depicting interactive behaviour of two people.
5. Track articulated 3D motion of a person in a video using annealed particle filtering. The best results were obtained using annealed filtering with 256 particles and 10 annealing layers.
6. Generate response behaviour for a “virtual friend” using a dual HMM as described in Section 3.
7. Put the “virtual friend” into a video sequence.

Selected frames from the generated video sequences are shown in Figures 4 and 5. A selection of generated video sequences are available from www.cardiff.ac.uk/schoolsanddivisions/academicschools/engin/cdsp/people/yue_zheng/ICSP2006/index.html.

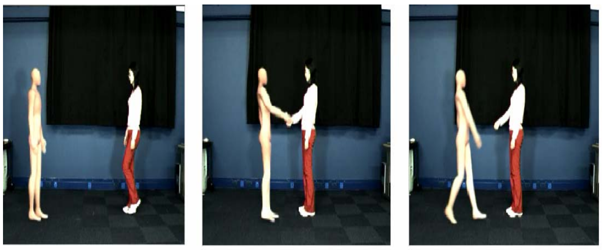


Figure 4. Interactions with virtual friend (handshake)

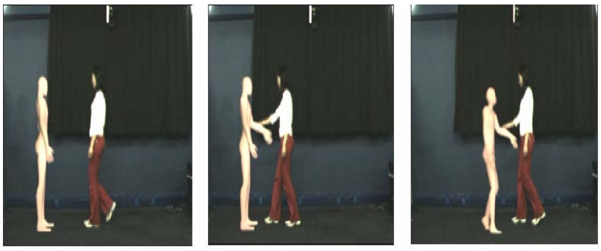


Figure 5. Interactions with virtual friend (pushing)

4.2. Evaluation

Here, our goal was to evaluate how convincing the generated motion was from the point of view of an independent observer. For this purpose, we generated six test video sequences, three of which showed original MoCap data collected from two people performing handshake, pushing, and pulling actions. In the remaining three sequences the MoCap data of the second person was substituted with motion data generated using the algorithm described in this article. You can download these video sequences from the same web page as the videos produced in the previous section. Each of the video sequences was shown to ten independent observers. The observers were told that the videos showed the motion data of two people performing some action and were asked to identify which action that was. They were also asked to comment if they noticed anything strange or unusual about the motion of the people. All ten subjects were able to identify the actions in all six videos correctly. Five out of ten commented on the motion of the generated character for shaking hands behaviour who seemed to float as described by one of the observers. However, three out of ten people also commented on the motion of the real person during pushing behaviour, who seemed to them wobbly. Also one observer out of ten commented that during the pushing behaviour with a generated character one person did not touch another. However, the last comment can be argued since the observers could not see the bodies of the people. Finally, one of the observers though that one of the real characters was bent too much. From these results, we conclude that the generated behaviours looked very similar to the real behaviours as they received approximately the same amount of comments. With exception of, perhaps, generated "shaking hands" motion, the observers did not notice anything unusual about the generated motion compared against the real motion.

5. Conclusion and Future work

In this article, we described a novel approach for generating intelligent behaviours for fully articulated 3D virtual characters on the basis of visual analysis of the real person present in the scene. In the past, to the best of our knowledge, nobody has done this for fully articulated 3D motion.

To this end, we trained a dual HMM representing interactive behaviours of two people. We tracked 3D articulated motion of a real person in video using annealed particle filtering. Then we used the obtained data in conjunction with dual HMM to generate the responsive behaviour for a virtual character. Finally, the virtual character performing the generated motion can be placed back into the original video sequence (Figures 4 and 5). The evaluation experiments showed that generated motion was very similar to real motion.

In the future, we are going to develop our model to represent more complex behaviours, for example, using hierarchical HMMs.

References

[1] N. Johnson and A. Galata and D. Hogg, "The Acquisition and Use of Interaction Behaviour Model", *IEEE CVPR Proceedings*, pp. 866-871, June, 1998.

[2] T. Jebara and A. Pentland, "Statistical Imitative Learning from Perceptual Data", *Proceedings of the 2nd International Conference on Developm-ent and Learning*, pp. 191-196, June, 2002.

[3] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *IEEE Proceedings*, vol. 77, no. 2, pp. 257-286, February, 1989.

[4] D. Cosker, D. Marshall, P. L. Rosin and Y. Hicks, "Speech Driven Facial Animation using a Hidden Markov Coarticulation Model", *IEEE ICPR*, vol. 1, pp. 314-321, August, 2004.

[5] D. M. Gavrila, "The visual analysis of human motion: a survey", *IEEE CVPR Proceedings,* vol. 73, pp. 82-98, January , 1999.

[6] M. Isard and A. Blake, "CONDENSATION - Conditional Density Propagation for Visual Tracking", *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.

[7] M. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, February, 2002.

[8] J. Deutscher, A. Blake and I. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering", *IEEE CVPR Proceedings*, vol. 2, pp. 126-133, 2000.

[9] T. Horprasert, D. Harwood, L. S. Davis, "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection", *In Proceedings IEEE ICCV'99 FRAME-RATE Workshop, Greece*, pp. 1-19, September, 1999.