AUDIO-DRIVEN HUMAN BODY MOTION ANALYSIS AND SYNTHESIS

Ferda Ofli [1], *Cristian Canton-Ferrer* [2], *Yasemin Demir* [1], *Koray Balcı* [3], *Joëlle Tilmanne* [4], *Elif Bozkurt* [5], *İdil Kızoğlu* [3], *Yücel Yemez* [1], *Engin Erzin* [1], *A. Murat Tekalp* [1], *Lale Akarun* [3], *A. Tanju Erdem* [5]

[1] Multimedia, Vision and Graphics Laboratory, Koç University, İstanbul, Turkey
[2] Image and Video Processing Group, Technical University of Catalonia, Barcelona, Spain
[3] Multimedia Group, Boğaziçi University, İstanbul, Turkey
[4] TCTS Lab, Faculté Polytechnique de Mons, Belgium
[5] Momentum Digital Media Technologies, İstanbul, Turkey

ABSTRACT

This project is on multicamera audio-driven human body motion analysis towards automatic and realistic audio-driven avatar synthesis. We address this problem in the context of a dance performance, where the gestures or the movements of a human actor are mainly driven by a musical piece. We analyze the relations between the audio (music) and the body movements on a training video sequence acquired during the performance of a dancer. The joint analysis provides us with a correlation model that is used to animate a dancing avatar when driven with any musical piece of the same genre.

KEYWORDS

Body motion analysis – Dance figures – Audio-driven synthesis

1. INTRODUCTION

There exists little research work reported on the problem of audio-driven human body motion analysis and synthesis. The most relevant literature is on speech-driven lip animation [1]. Since lip movement is physiologically tightly coupled with acoustic speech, it is relatively an easy task to find a mapping between the phonemes of speech and the visemes of lip movement. Many schemes exist to find such audio-to-visual mappings among which the HMM (Hidden Markov Model)-based techniques are the most common as they yield smooth animations exploiting temporal dynamics of speech. Some of these works also incorporates synthesis of facial expressions along with the lip movements to make animated faces look more natural [2, 3, 4].

Humans body motion can have many purposes: To go from one place to another, humans walk, or run. Walking is perhaps the most thoroughly studied form of body motions. Upper body motions, such as hand gestures can also have many aims: communicative, deictic or conversational. Sign language relies on hand gestures as well as upper body motions and facial expressions to convey a whole language [5]. On the other hand, some body motions express emotions. Dancing is a special type of body motion that has some predefined structure; as well as emotional aspects. Analysis of gestures in dance with the purpose of uncovering the conveyed emotions has been undertaken in recent researches [6, 7].

There are several challenges involved in audio-driven human body motion analysis and synthesis: First, there does not exist a well-established set of elementary audio and motion patterns, unlike phonemes and visemes in speech articulation. Second, body motion patterns are person dependent and open to interpretation, and may exhibit variations in time even for the same person. Third, audio and body motion are not physiologically coupled and the synchronicity in between may exhibit variations. Moreover, motion patterns may span time intervals of different length with respect to its audio counterparts. The recent works [8, 9] address the challenges similar to those mentioned above in the context of facial expression analysis and prosody-driven facial expression synthesis, using a multi-stream parallel HMM structure to find the jointly recurring gesture-prosody patterns and the corresponding audio-to-visual mapping.

In this work, our aim is to learn the predefined structure of a given dance. We analyze the relations between the music and the body movements on a training video sequence acquired during the performance of a dancer. We track the movements of the dancer using marker-based and markerless methods. We train an HMM with the basic pieces from the given genre. Then, given a music piece, we classify its genre and use the corresponding HMM for synthesis. We use a body animation software developed in this project to animate an avatar with the motion parameters produced by the HMM. We analyze the audio to extract parameters about the speed of the music and adapt the animation accordingly.

2. SYSTEM OVERVIEW

Our audio-driven body animation system is composed of multimodal analysis and synthesis blocks. In the analysis, we observe the recurring body motion patterns in the video and segment the video into partitions of meaningful dance figures. Then, we try to learn the recurring body motion patterns by training HMMs over these segments. In the mean time, we try to learn the audio by looking at its beat frequency within each time window we have for the video. We expect to have similar beat frequency values in the time windows that correspond to the same dance figure. However, beat frequency may vary throughout a single musical piece, or among different musical pieces. This variation is smaller in the former case whereas it is expected to be larger in the latter case. Therefore, the variation in beat frequency is used to determine the duration of dance figures as well as to specify the genre of the given musical piece.

In the synthesis, given a musical piece of one of the types we learn in the analysis part, we first classify the audio into the correct audio class. Then extracting the beat information of the given audio signal, we decide on which dance figure is going to be generated and how much time that the expected dance figure is going to occupy. After we obtain the outline of the dance

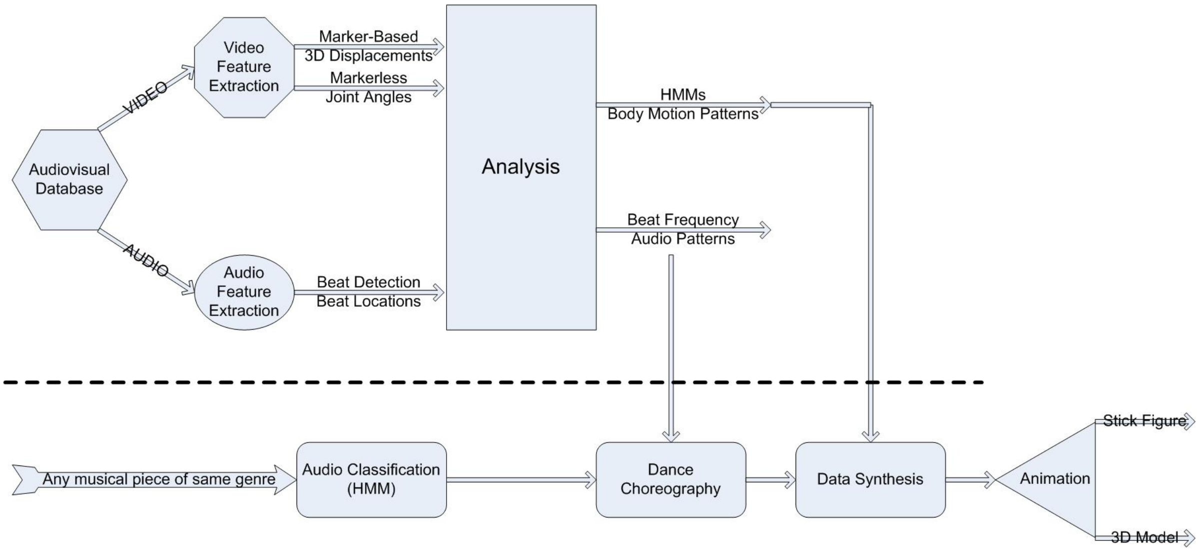


Figure 1: *Block diagram of the analysis-synthesis system.*

figure synthesis task, we start generating the dance figures using the corresponding HMMs for this specific set of dance figures. Fig. 1 shows the overall analysis-synthesis system.

3. VIDEO FEATURES EXTRACTION

Body motion capture and feature extraction involves automated capture of body motion from multiview video recorded by a multicamera system. We will employ two different methods for body motion tracking in parallel. One method will be based on 3D tracking of the markers attached to the person’s body in the scene. The other method will be based on 3D reconstruction of background segmented images of the scene. We will make use of the multistereo correspondence information from multiple cameras to obtain 3D motion information in both methods. This task will provide us with a set of features of joint angles over time that expresses the alignment of the body parts of the dancer in the scene.

3.1. Marker-based Motion Capture

The motion capture process involves tracking a number of markers attached to the dancer’s body as observed from multiple cameras and extraction of the corresponding motion features. Fig. 2 demonstrates our setting for this scenario. Markers in each video frame are tracked making use of their chrominance information. The 3D position of each marker at each frame is then determined via triangulation based on the observed projections of the markers on each camera’s image plane.

3.1.1. Initialization

Markers on the subject are manually labeled in the first frame for all camera views. We change the color space from RGB to YCrCb which gives flexibility over intensity variations in the frames of a video as well as among the videos captured by cameras at different views. We assume that the distributions of Cr and Cb channel intensity values belonging to marker regions are Gaussian. Thus, we calculate the mean, μ, and the covariance, $\mathbf{\Sigma}$, over each marker region (a pixel neighborhood around the labeled point), where $\mu = [\mu_{Cr}, \mu_{Cb}]^T$ and $\mathbf{\Sigma} = (\mathbf{c}-\mu)(\mathbf{c}-\mu)^T$, $\mathbf{c}$ being $[c_{Cr}, c_{Cb}]^T$.

Let M be the number of markers on the subject and $\mathbf{W}$ be the set of search windows, where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M]$ such that each window $\mathbf{w}_m$ is centered around the location, $[x_m, y_m]^T$, of the corresponding marker. The set $\mathbf{W}$ is used to track markers over frames. Thus the center of each search window, $\mathbf{w}_m$, is initialized as the point manually labeled in the first frame and specifies the current position of the marker.

3.1.2. Tracking

To track the marker positions through the incoming frames, we use the Mahalanobis distance from $\mathbf{c}$ to $(\mu, \mathbf{\Sigma})$ where $\mathbf{c}$ is a vector containing Cr and Cb channel intensity values $[c_{Cr}, c_{Cb}]^T$ of a point $\mathbf{x}_n \in \mathbf{w}_m$. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ be the set of candidate pixels for which the chrominance distance is less than a certain threshold. If the number of these candidate pixels, N, is larger than a predefined value, then we label that marker as visible in the current camera view and update its position as the mean of the points in $\mathbf{X}$ for the current camera view. The same process is repeated for all marker points in all camera views. Hence, we have the visibility information of each marker from each camera, and for those that are visible, we have the list of 2D positions of the markers on that specific camera image plane.

Once we scan the current scene from all cameras and obtain the visibility information for all markers, we start calculating the 3D positions of the markers by back-projecting the set of 2D points which are visible in respective cameras, using triangulation method. Theoretically, it is sufficient to see a marker at least from two cameras to be able to compute its position in 3D world. If a marker is not visible at least from two cameras, then its current 3D position is estimated from the information in the previous frame.

The 3D positions of markers are tracked over frames by Kalman filtering where the filter states correspond to 3D position and velocity of each marker. The list of 3D points obtained by back-projection of visible 2D points in respective camera image planes constitute the observations for this filter. This filtering operation has two purposes:

- to smooth out the measurements for marker locations in the current frame,
- to estimate the location of each marker in the next frame and to update the positioning of each search window,

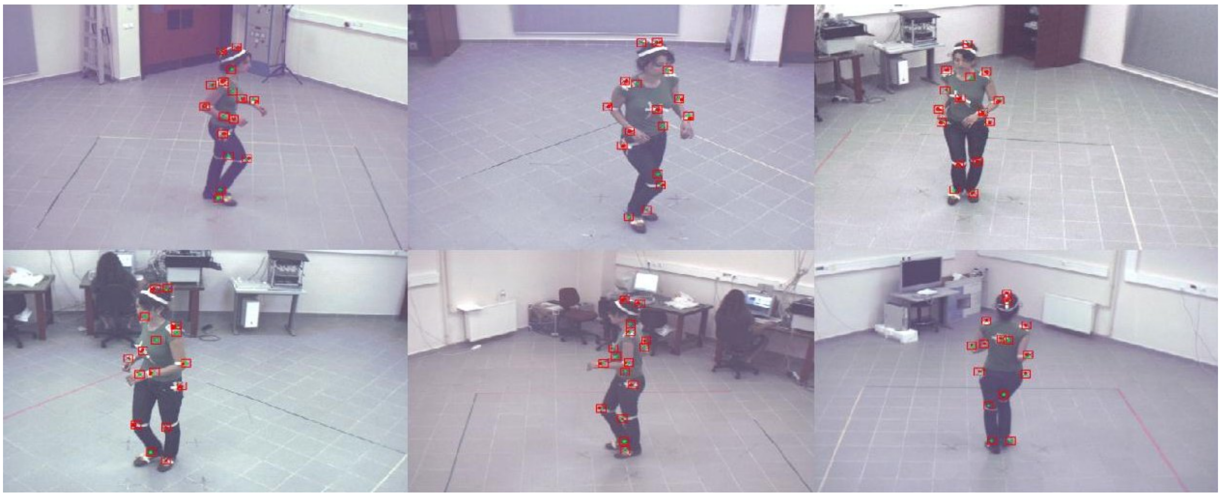


Figure 2: *An example scene.*

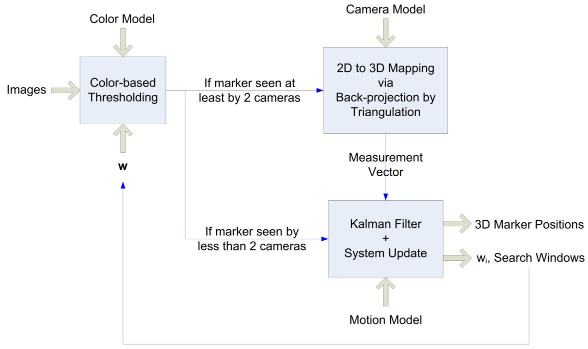

Color Model
Images
Color-based Thresholding
w
If marker seen at least by 2 cameras
Camera Model
2D to 3D Mapping via Back-projection by Triangulation
Measurement Vector
Kalman Filter + System Update
If marker seen by less than 2 cameras
3D Marker Positions
w_i, Search Windows
Motion Model

Figure 3: *Block diagram of the proposed tracking system.*

$\mathbf{w}_m$, on the corresponding image plane accordingly.

Fig. 3 summarizes the overall system. Having updated the list of 3D marker positions for the current frame and estimated the location of the search windows for the next frame, we move on to the next frame and search the marker positions within the new search windows. This algorithm is repeated for the whole video.

3D positions of each marker is used to extract the euler angles for each joint by using inverse kinematic chain structure. The derivation of the euler angles that belong to neck is given exactly here and the other derivations results are given in Fig. 5 which are calculated in an analogous way with the given neck angles. In Fig. 4, the derivation of the angles in neck are delineated. The given vector **p** demonstrates the vector from neck joint to head centroid in the torso-centered coordinate system, these are defined as:

$$\vec{p} = R_0^{-1} \times (\vec{p}_{neck} - \vec{p}_{headCen}). \tag{1}$$

$\vec{R}_0$ is the rotation matrix of torso $\vec{v}_0$ is the vector along z axis, $\vec{v}_1$ is the projection on x-z plane and $\vec{v}_2$ is the projection on y-z plane.

We can write the vectors as:

$$\vec{v}_0 = \begin{bmatrix} 0 & 0 & \sqrt{x^2+y^2+z^2} \end{bmatrix}^T \tag{2}$$

,

$$\vec{v}_1 = \begin{bmatrix} x & 0 & \sqrt{y^2+z^2} \end{bmatrix}^T \tag{3}$$

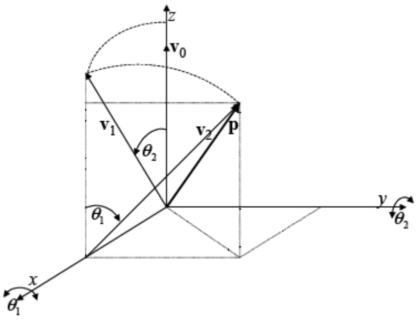


Figure 4: *The vector that connects neck joint and head centroid.*

,

$$\vec{v}_2 = \begin{bmatrix} 0 & y & z \end{bmatrix}^T \tag{4}$$

and with these given vectors we can calculate the angles for neck as given below:

$$\Theta 1 = -\arccos((e_3 v_2) \div (\|v_2\|)) sign(y) \tag{5}$$

$$\Theta 2 = \arccos((e_3 v_1) \div (\|v_1\|)) sign(x) \tag{6}$$

3.2. Markerless Motion Capture

Retrieving the body configuration in terms of its defining parameters, i.e. joint angles, from unlabeled video data presents a number of challenges. The main advantage of this technique is that no intrusive markers are required. However, the precision of the output may no be as accurate as the one obtained from marker based techniques since the input data is corrupted by noise.

3.2.1. 3D Data generation

For a given frame in the video sequence, a set of N images are obtained from the N cameras. Each camera is modeled using a pinhole camera model based on perspective projection with camera calibration information available. Foreground regions from input images are obtained using a segmentation algorithm based on Stauffer-Grimson's background learning and substraction technique [10] as shown in Fig. 6b.

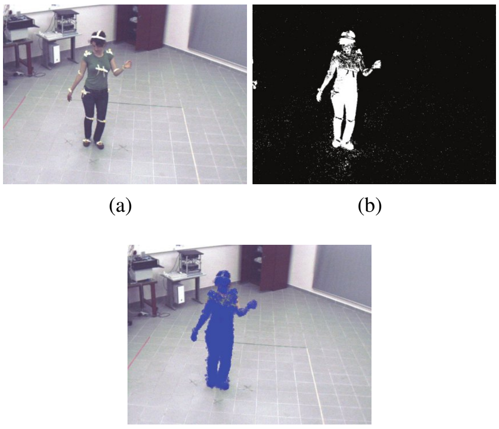
Angle	Formula	Vector $\mathbf{v}$	Vector $\mathbf{p} = \begin{bmatrix} x & y & z \end{bmatrix}^T$
Neck: θ_1	$\theta_1 = -\arccos(\mathbf{e}_3\mathbf{v})\mathrm{sgn}(y)$	$\mathbf{v} = \frac{1}{\sqrt{y^2+z^2}} \begin{bmatrix} 0 & y & z \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_1 - \mathbf{p}_{10})$
Neck: θ_2	$\theta_2 = \arccos(\mathbf{e}_3\mathbf{v})\mathrm{sgn}(x)$	$\mathbf{v} = \frac{1}{\sqrt{x^2+y^2+z^2}} \begin{bmatrix} x & 0 & \sqrt{y^2+z^2} \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_1 - \mathbf{p}_{10})$
Left shoulder: θ_3	$\theta_3 = \arccos(\mathbf{e}_2\mathbf{v})\mathrm{sgn}(z)$	$\mathbf{v} = \frac{1}{\sqrt{y^2+z^2}} \begin{bmatrix} 0 & y & z \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_2 - \mathbf{p}_{11})$
Right shoulder: θ_4	$\theta_4 = \arccos(\mathbf{e}_2\mathbf{v})\mathrm{sgn}(z)$	$\mathbf{v} = \frac{1}{\sqrt{y^2+z^2}} \begin{bmatrix} 0 & y & z \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_3 - \mathbf{p}_{12})$
Left shoulder: θ_5	$\theta_5 = -\arccos(\mathbf{e}_2\mathbf{v})\mathrm{sgn}(x)$	$\mathbf{v} = \frac{1}{\sqrt{x^2+y^2+z^2}} \begin{bmatrix} x & \sqrt{y^2+z^2} & 0 \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_2 - \mathbf{p}_{11})$
Right shoulder: θ_6	$\theta_6 = \arccos(\mathbf{e}_2\mathbf{v})\mathrm{sgn}(x)$	$\mathbf{v} = \frac{1}{\sqrt{x^2+y^2+z^2}} \begin{bmatrix} x & \sqrt{y^2+z^2} & 0 \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_3 - \mathbf{p}_{12})$
Left shoulder: θ_7	$\theta_7 = -\arccos(\mathbf{R}_3\mathbf{R}_5\mathbf{e}_3(\mathbf{v}_2 \times \mathbf{v}_1))\cdot$ $\mathrm{sgn}(\mathbf{v}_1(\mathbf{v}_2 \times (\mathbf{v}_2 \times \mathbf{v}_1)))$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_2 - \mathbf{p}_{15})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_6 - \mathbf{p}_{15})$	
Right shoulder: θ_8	$\theta_8 = -\arccos(\mathbf{R}_4\mathbf{R}_6\mathbf{e}_3(\mathbf{v}_1 \times \mathbf{v}_2))\cdot$ $\mathrm{sgn}((\mathbf{v}_2 \times (\mathbf{v}_1 \times \mathbf{v}_2))\mathbf{v}_1)$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_3 - \mathbf{p}_{16})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_7 - \mathbf{p}_{16})$	
Left elbow: θ_9	$\theta_9 = \pi - \arccos(\mathbf{v}_1\mathbf{v}_2)$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_2 - \mathbf{p}_{15})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_6 - \mathbf{p}_{15})$	
Right elbow: θ_{10}	$\theta_{10} = -\pi + \arccos(\mathbf{v}_1\mathbf{v}_2)$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_3 - \mathbf{p}_{16})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_7 - \mathbf{p}_{16})$	
Left hip: θ_{11}	$\theta_{11} = \arccos(-\mathbf{e}_3\mathbf{v})\mathrm{sgn}(y)$	$\mathbf{v} = \frac{1}{\sqrt{y^2+z^2}} \begin{bmatrix} 0 & y & z \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_4 - \mathbf{p}_{13})$
Right hip: θ_{12}	$\theta_{12} = \arccos(-\mathbf{e}_3\mathbf{v})\mathrm{sgn}(y)$	$\mathbf{v} = \frac{1}{\sqrt{y^2+z^2}} \begin{bmatrix} 0 & y & z \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_5 - \mathbf{p}_{14})$
Left hip: θ_{13}	$\theta_{13} = -\arccos(-\mathbf{e}_3\mathbf{v})\mathrm{sgn}(x)$	$\mathbf{v} = \frac{1}{\sqrt{x^2+y^2+z^2}} \begin{bmatrix} x & 0 & \sqrt{y^2+z^2} \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_4 - \mathbf{p}_{13})$
Right hip: θ_{14}	$\theta_{14} = -\arccos(-\mathbf{e}_3\mathbf{v})\mathrm{sgn}(x)$	$\mathbf{v} = \frac{1}{\sqrt{x^2+y^2+z^2}} \begin{bmatrix} x & 0 & \sqrt{y^2+z^2} \end{bmatrix}^T$	$\mathbf{p} = \mathbf{R}_0^T(\mathbf{p}_5 - \mathbf{p}_{14})$
Left knee: θ_{15}	$\theta_{15} = \pi - \arccos(\mathbf{v}_1\mathbf{v}_2)$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_8 - \mathbf{p}_{19})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_4 - \mathbf{p}_{19})$	
Right knee: θ_{16}	$\theta_{16} = \pi - \arccos(\mathbf{v}_1\mathbf{v}_2)$	$\mathbf{v}_1 = \mathbf{R}_0^T(\mathbf{p}_9 - \mathbf{p}_{20})$ $\mathbf{v}_2 = \mathbf{R}_0^T(\mathbf{p}_5 - \mathbf{p}_{20})$	

Figure 5: *The formulas for calculation of joint euler angles.*

Redundancy among cameras is exploited by means of a Shape-from-Silhouette (SfS) technique [11]. This process generates a discrete occupancy representation of the 3D space (voxels). Each voxel is labelled as foreground or background by checking the spatial consistency of its projection on of the N segmented silhouettes. The data obtained with this 3D reconstruction is corrupted by spurious voxels introduced due to wrong segmentation, camera calibration inaccuracies, etc. A connectivity filter is introduced in order to remove these voxels by checking its connectivity consistency with its spatial neighbors. An example of the output of the whole 3D processing module is depicted in Fig. 6c. For the research presented whithin this paper, it is assumed that only one person is present in the scene. Let us refer to the obtained voxel data as $\mathcal{V}$.

3.2.2. Human Body Model

In order to analyze the incoming data $\mathcal{V}$, an articulated body model will be used. This body model allows exploiting the underlying antropomorphic structure of the data; let us refer to this model as $\mathcal{H}$. Model based analysis of humans as been already addressed in the literature in [12, 13]. The employed model is formed by a set of joints and links representing the limbs, head and torso of the human body and a given number of degrees of freedom (DoF) are assigned to each articulation (joint). Particularly, our model has 22 DoF to properly capture the possible movements of the body: position of the center of the torso (3 DoF), rotation of the torso (3 DoF), rotation of the neck (2 DoF), rotation of the shoulders (3+3 DoF), rotation of the elbows (1+1 DoF), rotation of the hips (2+2 DoF) and rotation of the ankles (1+1 DoF). An example of this body model is depicted in Fig. 7.


(a)

(b)

(c)

Figure 6: *3D data generation. In (a), one of the original N images. In (b), the foreground/background binary segmentation and, in (c), the projection of the voxels defining the input 3D data.*

The equations driving the behavior of the joints rely on kinematic chains formulated by means of exponential maps [14, 15].

3.2.3. Human Body Tracking

Particle Filtering (PF) [16] algorithms are sequential Monte Carlo methods based on point mass (or "particle") representations of probability densities. These techniques are employed to tackle estimation and tracking problems where the variables involved do not hold Gaussianity uncertainty models and linear dynamics. In the current scenario, the hidden state to be estimated, that is the set of 22 DoF of the human body model, falls in the aforementioned conditions hence particle filtering techniques may efficiently retrieve this state vector.

Two major issues must be addressed when employing PF: likelihood evaluation and propagation model. The first one, establishes the observation model, that is how a given configuration of a the body matches the incoming data. The propagation model is adopted to add a drift to the angles of the particles in order to progressively sample the state space in the following iterations [16]. For complex PF problems involving a high dimensional state space such as in this articulated human body tracking task [13], an underlying motion pattern is employed in order to efficiently sample the state space thus reducing the number of particles required. This motion pattern is represented by the kinematical constrains and physical limits of the joints of the human body.

Likelihood evaluation being a crucial step is described as follows. For any particle, a volumetric representation of the human body using hyper-ellipsoids is generated, $\tilde{\mathcal{H}}$ (see Fig. 7 for an example). Within this representation, every limb of the generated model is denoted as $\mathcal{L}_k$. Likelihood

$$p(\tilde{\mathcal{H}}^j|\mathcal{V}) = \prod_{k=0}^{K} \mathcal{L}_k \cap \mathcal{V}, \quad (7)$$

where $\cap$ operator denotes the volumetric intersection between the limb of the model $\mathcal{L}_k$ and the incoming data $\mathcal{V}$. Individual likelihoods of each limb are assumed to be independent in order to generate the global human body likelihood function. Current research involves employing more informative measures including color information.

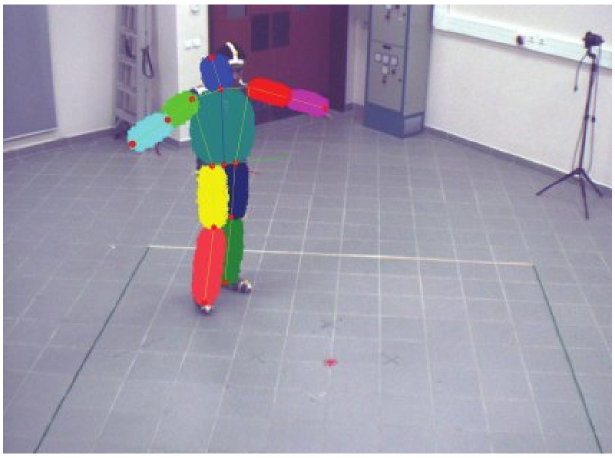


Figure 7: *Articulated human body model with 22 DoF and hyper-ellipsoids to represent the limbs.*

4. AUDIO FEATURES EXTRACTION

An appropriate set of features will be extracted from the audio signal that is synchronized with the body motion parameters. The mel frequency cepstral coefficients (MFCC) along with additional prosodic features can be considered as audio features. Audio feature extraction will be performed using the well known HTK Tool.

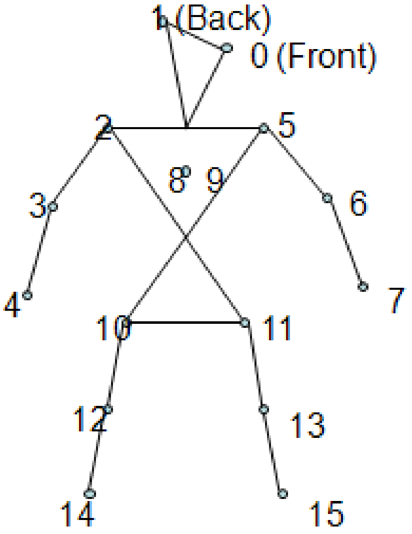


Figure 8: *Markers positions (10 to 15 for lower body, 2 to 7 for upper body).*

5. ANALYSIS

The feature sets resulting from body motion and audio will jointly be analyzed to model the correlation between audio patterns and body motion patterns. For this purpose, we plan to use a two-step HMM-based unsupervised analysis framework as proposed in [5]. At the first step, the audio and motion features will separately be analyzed by a parallel HMM structure to learn and model the elementary patterns for a particular performer. A multi-stream parallel HMM structure will then be employed to find the jointly recurring audio-motion patterns and the corresponding audio-to-visual mapping. All the simulations at this second step will be implemented by using the HTK Toolkit.

The body motion synthesis system will take an audio signal as an input and produce a sequence of body motion features, which are correlated with the input audio. The synthesis will be based on the HMM-based audio-body motion correlation model derived from the multimodal analysis. The synthesized body motion will then be animated on an avatar.

5.1. Video Analysis

Human body motion analysis will be tackled through HMMs. Dance motion can be addressed by analyzing patterns that are repeated sequentially by the dancer and a set of HMMs is trained separately for each dance figure. Data employed to train the HMMs are the normalized 3D positions of some landmarks defined on the body (typically the body joints) and tracked along time by means of the two vision based analysis systems. For each figure, two sub-HMMs are defined to better capture the dynamics behavior of the upper and lower part of the body. The HMM modelling the upper part of the body addresses the arms movement (described by the (x, y, z) positions of the six landmarks placed in shoulders, elbows and wrists) while the other HMMs accounts for the legs (described by the (x, y, z) position for the six landmarks placed in hips, knees and ankles) (Fig 8).

To start evaluating the performance of the system presented in this report, a simple HMM is adopted. Typically, dance figures always contain a very concrete sequence of movements hence a left-right HMM structure is employed (Fig 9). Each of the parameters is represented by a single Gaussian function

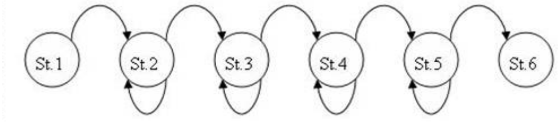


Figure 9: *Simple left-right HMM structure.*

and one full covariance matrix is computed for each state. This rather simple scheme leads to satisfactory results hence no further complexity is added to the simple. All computation as been done by means of the ”Hidden Markov Model Toolkit” (HTK) package developed at the Cambridge University. This packages allowed us to efficiently model the HMM structures employed within this project.

5.2. Audio Analysis

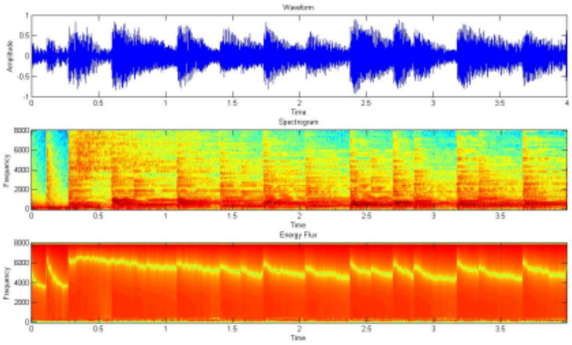


Figure 10: *From top to bottom: time waveform, spectrogram and spectral energy flux of four seconds of Salsa music audio file.*

Both for Salsa and Belly dance music audio files we measure tempo in terms of beats per minute (BPM) using the estimation algorithm suggested in [17]. Tempo estimation can be broken down into three sections: onset detection, periodicity estimation and beat location estimation. Onset detection aims to point out where musical notes begin and tempo is established by periodicity of the detected onsets. It is straight forward to detect beat locations after periodicity estimation.

First, we detect onsets based on the spectral energy flux of the input audio signal that signify the most salient features of the audio file as shown Fig. 10. Onset detection is important, since beat tends to occur at onsets. Algorithm works best for four second analysis window with 50% overlap. The below figure belongs to the first four seconds analysis window of the Salsa music audio file.

Next, we estimate the periodicity of the detected onsets using the autocorrelation method. The distance between the largest peak in the interval from 300 ms to 1 s of the autocorrelated signal and its origin gives the periodicity value. Once the periodicity is determined we can calculate tempo in terms of number of beats per minute, which lies between the values 60 and 200 BPM. For Salsa music audio file we estimate tempo as 185 BPM and for the Belly dance it is 134 BPM.

Furthermore, we estimate beat locations regarding the periodicity value in the previous step. We generate an artificial pulse train with the estimated periodicity and cross - correlate with the onset sequence where maximum value of the cross - correlation gives the starting beat location. Successive beats are expected in every beat period *T*.

Analyzing the results from labeling of the dance figures in the video frames, we conclude that each Salsa figure corresponds to 8 beats in the Salsa music audio file and each Belly dance figure corresponds to 3 beats in the Belly dance music. We also use this information during the synthesis time to determine the beginning and ending frames of a dance figure.

6. SYNTHESIS

Given a audio track, the generated classifier is used to classify the tracks as belly or salsa. The classified track beats is extracted by the method explained in section 4 and the beat periods of each track are used to generate figures that are learnt by HMMs in the motion analysis part. The appropriate HMM that would generate figures with the given audio track is used by the synthesizer. We have one HMM model for salsa which generates the basic figure with the given salsa track. Belly dance sequence is more complicated than the salsa dance sequence. It yields three independent figures with just one beat period. An HMM is trained for this scenario. We will generate a coupled HMM with individual HMM models in each state that correspond to different figures that are recognized during training. The state transitions are determined according to the dancer sequence and the transition probabilities are calculated from the co-occurrence matrices of audio beat numbers and video labels.

6.1. Audio Classification

This part is a simple music genre classification problem. We have two types of music audio files where one is Salsa and the other is Belly dance. We use supervised HMMs and the well - known Mel Frequency Cepstral Frequency coefficients (MFCCs) to discriminate the 16 kHz, 16 bit mono PCM wavefiles. The music audio signals are analyzed over 25 ms Hamming window for each 10 ms frame. Finally, 13 MFCC coefficients together with the accelaration and the delta parameters, adding up to 39 features, form the audio feature vectors. Use of MFCCs as the only audio feature set is sufficient for the classification problem, since we have only two kinds of audio files. For the extraction of parameters and classification steps, we use HTK toolkit.

Using the HMMs generated in the analysis step we first classify the input music audio files as Salsa or Belly dance as depicted in Fig. 11, below. Then, we estimate the beat signal for the detected music audio file following the steps onset detection, periodicity estimation and beat location. Next, we identify the beat segmentation times in the music audio and determine the duration (in terms of frame numbers) of figures to be performed during the animation. Precalculated beats per frame information that we got in the analysis section is used for this purpose. For example, for Salsa, each figure corresponds to a time segment of eight beats, so by multiplying the start and end time of the each segment with the number of frames per second (30 in our case), we simply get the beginning and ending frame numbers for Salsa dance figures.

6.2. Body Motion Parameters Generation

Once we have the list of durations and types of consecutive dance figures in a file, we can use that file to generate the appropriate values for the animation parameters according to the mean and standard deviation values of the corresponding HMM states. This file basically determines how much time each dance figure takes in the sequence. This helps us to allocate exactly the necessary amount of time to perform each dance figure.

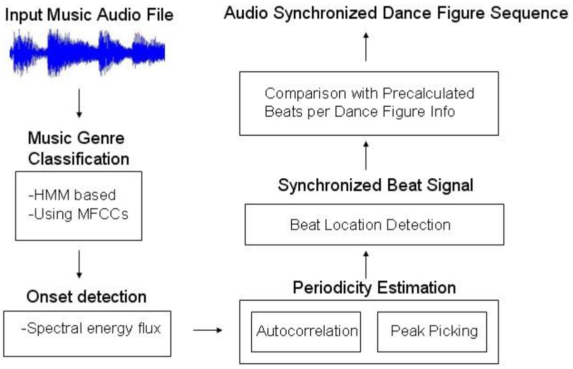


Figure 11: *Audio processing steps in the synthesis part.*

7. VISUALIZATION

For avatar model, we used a free 3D model named Douglas F. Woodward shown in Figure 12 with 9599 vertices and 16155 faces. The model comes with segmented hierarchy, which let us create a kinematic chain of segments in a conventional directed acyclic graph (DAG) structure.

We decided to implement a generic synthetic body representation and animation tool instead of relying on a single model. Our tool, namely Xbody, can open models in 3DS format and display the DAG and submesh info and enables labeling of the segments for animation as can be seen in Figure 12. For rendering, Xbody relies on OpenGL and existing Xface [18] codebase. We implemented an additional forward kinematics pipeline for rendering and animation of DAG.

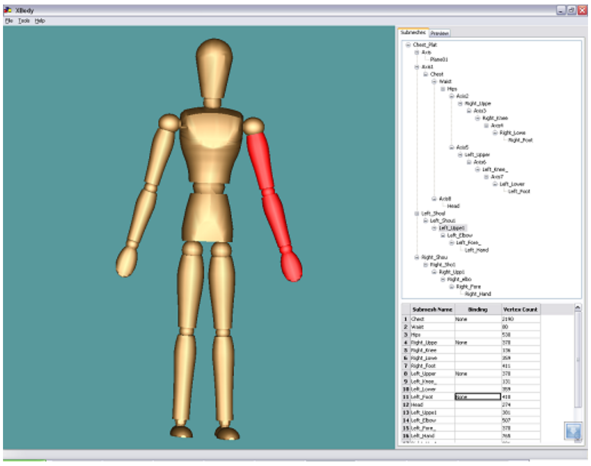


Figure 12: *Xbody DAG view and labelling pane.*

As for animation, the generated data from analysis and synthesis part can be fed to Xbody and animated with the same frame per second of video. The previewing interface of the tool enables us to inspect each frame by entering the frame number and using rotation, zooming in/out and panning the model on the screen. In Figure 13, previewing of frame 180 for markerles tracking analysis result for "Zeybek" dance is shown. The tool can also export the animation as video in avi format.

As its current state, Xbody can be used for better analyzing the results of motion tracking algorithms and HMM based motion generation. In the future, we plan to improve on Xbody and implement full support for MPEG-4 Body Animation by parsing BAP and BDP formats.

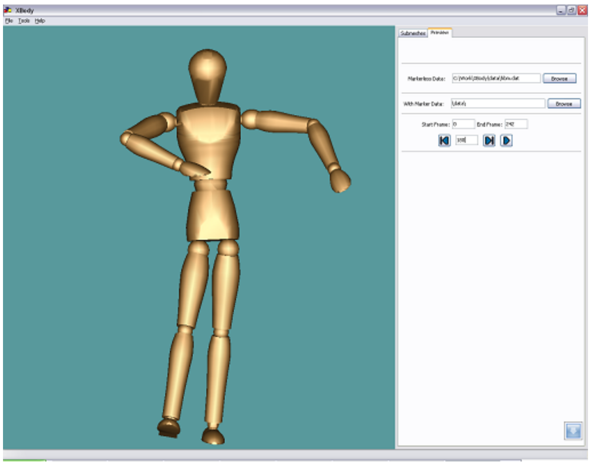


Figure 13: *Xbody preview pane.*

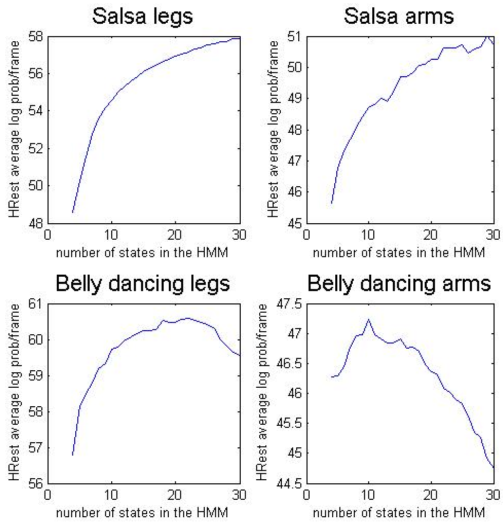


Figure 14: *Evolution of the logarithmic probability per frame for the 4 HMMs types.*

8. EXPERIMENTS AND RESULTS

As we modeled two dance figures (salsa and belly dancing) and that the whole body movement modeling was splitted into two HMMs, we had four HMMs to train.

The training of the HMM was performed using the HTK function HERest. It takes as input the data parameters file in HTK format, a prototype of the HMM containing its structure, and the transcription of the data file. In our case, we trained only one HMM at a time, and the transcription is only a succession of the same HMM name.

At the end of each training iteration, HERest gives an average logarithmic probability per frame. The evolution of this parameter enables us to follow the progression of the learning process and the accuracy of the trained model.

As we had no prior knowledge of the optimal number of states, we trained HMMs with an increasing number of states (from 4 to 30) and compared their average logarithmic probability per frame. The evolution of this parameter for the four types of HMMs we trained is shown in figure 14.

As the arms are hardly moving in the belly dance and as the salsa motion pattern is much more complicated than the belly

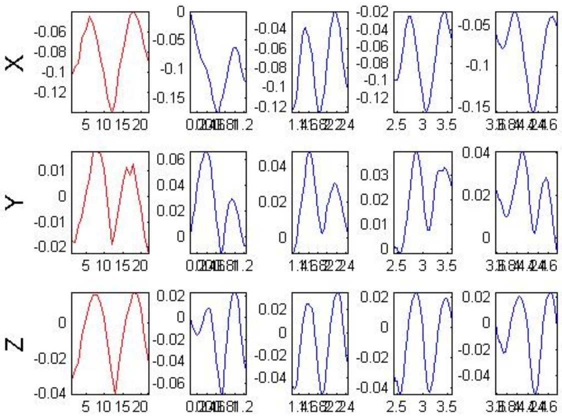


Figure 15: *Comparison of the means of the Gaussian distributions for 22 states of the belly dance HMM (in red) to the evolution of the 3 corresponding parameters during 4 dance figures (in blue) (x,y and z values of the left hip) during one salsa figure.*

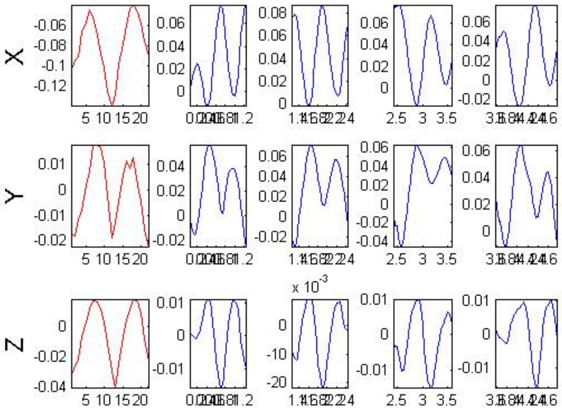


Figure 16: *Comparison of the HMM Gaussian means and the corresponding parameter (left knee) in 4 occurrence of the belly figure where the inter occurrence variability is very high*

dance one, we can see that the number of states required is linked to the complexity of the motion to model. For belly dancing, the optimal number is clear (around 20 for the legs and 10 for the arms), but in salsa there is no decrease in the logarithmic probability parameter before 30 states. For the salsa motion, we decided to keep around 20 states for both legs and arms as saturation began around that number and that the number of states (and thus the complexity of the HMM model) has to be kept reasonable.

In order to verify that the modeling of the data parameters was correct, we compared, for each parameter, the evolution of the mean of its Gaussian distribution across the states to the evolution of the same parameter for a few occurrence of the dance figure in the training dataset. The shape of the evolution can clearly be recognized (Fig. 15), even if some parameters vary highly between two occurrences of the same dance figure in the training set and are thus more difficult to model (Fig. 16).

9. CONCLUSIONS AND FUTURE WORK

In this research work, we first developed an automated human body motion capture system based solely on image processing and computer vision tools using standard digital video cameras. Second we provided a framework for joint analysis of loosely correlated modalities such as motion and audio and demonstrate how this framework can be used for audio-driven motion synthesis.

10. SOFTWARE NOTES

As a result of this project, the following resources are available:

- Dance databases
- Color-marker-based motion tracking software

11. REFERENCES

[1] T. Chen, "Audiovisual speech processing", *IEEE Signal Processing Magazine*, vol. 18, no. 1, pp. 9–21, 2001. 61

[2] C. Bregler, M. Covell, and M. Slaney, "Video Rewrite: driving visual speech with audio", in *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 353–360, ACM Press/Addison-Wesley Publishing Co., 1997. 61

[3] M. Brand, "Voice puppetry", in *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 21–28, ACM Press/Addison-Wesley Publishing Co., 1999. 61

[4] Y. Li and H.-Y. Shum, "Learning dynamic audio-visual mapping with input-output Hidden Markov models", *IEEE Transactions on Multimedia*, vol. 8, no. 3, pp. 542–549, 2006. 61

[5] O. Aran, I. Ari, A. Benoit, A. H. Carrillo, F. Fanard, P. Campr, L. Akarun, A. Caplier, M. Rombaut, and B. Sankur, "SignTutor: An Interactive Sign Language Tutoring Tool", in *Proceedings of eNTERFACE 2006, The Summer Workshop on Multimodal Interfaces, Dubrovnik, Croatia*, 2006. 61

[6] A. Camurri, B. Mazzarino, M. Ricchetti, R. Timmers, and V. G, *Multimodal Analysis of Expressive Gesture in Music and Dance Performances*, vol. 2915/2004. Heidelberg: Springer Berlin, 2004. 61

[7] F. Ofli, Y. Demir, E. Erzin, Y. Yemez, and A. M. Tekalp, "Multicamera Audio-Visual Analysis of Dance Figures", *IEEE International Conference on Multimedia and Expo, 2007. ICME 2007*, 2007. 61

[8] F. Ofli, E. Erzin, Y. Yemez, and A. M. Tekalp, "Estimation and Analysis of Facial Animation Parameter Patterns", *to appear in IEEE International Conference on Image Processing, 2007. ICIP 2007*, 2007. 61

[9] M. E. Sargin, E. Erzin, Y. Yemez, A. M. Tekalp, A. T. Erdem, C. Erdem, and M. Ozkan, "Prosody-Driven Head-Gesture Animation", *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007*, vol. 2, pp. 677–680, 2007. 61

[10] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking", in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 252–259, 1999. 63

[11] G. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler, "A real time system for robust 3D voxel reconstruction of human motions", in *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 714–720, 2000. 64

[12] F. Caillette and T. Howard, "Real-Time Markerless Human Body Tracking with Multi-View 3-D Voxel Reconstruction", in *Proceedings of British Machine Vision Conference (BMVC)*, vol. 2, pp. 597–606, September 2004. 64

[13] J. Deutscher and I. Reid, "Articulated Body Motion Capture by Stochastic Search", *International Journal of Computer Vision*, vol. 61, pp. 185–205, Feb. 2005. 64, 65

[14] M. J. Bregler, C., "Tracking people with twists and exponential maps", in *IEEE Conf. on Computer Vision and Pattern Recognition*, 1998. 64

[15] F. Sebastin-Grassia, "Practical parameterization of rotations using the exponential map", *J. Graph. Tools*, vol. 3, no. 3, pp. 29–48, 1998. 64

[16] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking", *Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on]*, vol. 50, no. 2, pp. 174–188, 2002. 64, 65

[17] M. Alonso, B. David, and G. Richard, "Tempo and Beat Estimation of Music Signals", in *Proceedings of ISMIR 2004, Barcelona, Spain*, 2004. 66

[18] K. Balci, E. Not, M. Zancanaro, and F. Pianesi, "Xface open source project and smil-agent scripting language for creating and animating embodied conversational agents", in *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, (New York, NY, USA), pp. 1013–1016, ACM, 2007. 67

12. BIOGRAPHIES

Lale Akarun is a professor in computer engineering of Boğaziçi University, İstanbul. Her research interests are in face recognition and gesture recognition. She is currently involved in FP6 projects Biosecure and SIMILAR. She would like to participate in projects in Biometrics and in Human computer interaction.
Email: akarun@boun.edu.tr

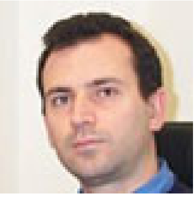
Koray Balcı received his B.S degree in Electrical and Electronics Engineering and M.S. degree in Cognitive Sciences from Middle East Technical University (METU), Ankara, Turkey, in 2000 and 2003. Since 2005, he is a PhD student at Boğazici University, İstanbul, Turkey. His research topic is human body animation. He is also a research consultant in Bruno Kessler Foundation (formerly ITC-irst) in Trento, Italy since 2003. He has participated in European projects PF-STAR and NetCarity and implemented Xface 3D Facial Animation toolkit.
Email: koraybalci@boun.edu.tr

Elif Bozkurt is a researcher at Momentum Digital Media Technologies, Gebze, Kocaeli, Turkey. She received her BS degree in Telecommunications Engineering from Sabancı University, İstanbul, Turkey, in July 2004. Since August 2004 she is with the Momentum Digital Media Technologies. Her research interests are speech synthesis, speech - driven 3D facial animation analysis and synthesis and emotion recognition from speech.
Email: ebozkurt@momentum-dmt.com

Cristian Canton-Ferrer received the Electrical Engineering degree from the Technical University of Catalonia (UPC), Barcelona, Spain, in 2003. He is currently working towards the PhD degree at the Image and Video Processing Group in the UPC. He made his Ms. Thesis at the Signal Processing Institute at the Swiss Federal Institute of Technology (EPFL). He has contributed to European projects such as IST CHIL and NoE SIMILAR and MUSCLE. His research interests focuses in multiview image analysis, gesture recognition, human body motion and gait analysis.
Email: ccanton@gps.tsc.upc.edu

Yasemin Demir received her B.S degree in Telecommunication Engineering from İstanbul Technical University (İTÜ), İstanbul Turkey in 2006. Since 2006, she is a master student at Koç University, İstanbul, Turkey. Her research topic is audio-visual analysis of dance figures.
Email: ydemir@ku.edu.tr

A. Tanju Erdem is the CTO and co-founder of Momentum Digital Media Technologies. He received a B.S. degree in electrical and electronics engineering and a B.S. degree in physics, both in 1986, from Boğaziçi University. He received an M.S. degree in 1988 and a Ph.D. degree in 1990, both in electrical engineering, from University of Rochester. Prior to Momentum, he was with the Research Laboratories of Eastman Kodak Company from 1990 to 1998. He holds 9 U.S. patents in the field of video processing and 3D face animation and has authored and co-authored more than 50 technical publications in these fields.
Email: terdem@momentum-dmt.com



Engin Erzin is an assistant professor in the Electrical and Electronics Engineering and the Computer Engineering Departments of Koç University, İstanbul, Turkey. His research interests include speech signal processing, pattern recognition, and adaptive signal processing. Erzin received a PhD, MS, and BS from the Bilkent University, Ankara, Turkey, all in electrical engineering.
Email: eerzin@ku.edu.tr

İdil Kızoğlu is a B.S. student in Computer Engineering Department at Boğazici University, İstanbul, Turkey.
Email: idilkizoglu@boun.edu.tr

Ferda Ofli received B.Sc. degree in Electrical and Electronics Engineering, and B.Sc. degree in Computer Engineering from Koç University, İstanbul, Turkey in 2005. He is now a M.Sc. student in Electrical and Computer Engineering Department and a member of Multimedia, Vision and Graphics Laboratory at Koç University. He is currently taking part in European projects, SIMILAR NoE and 3DTV NoE. His research interests include image and video processing, specifically, object segmentation and tracking, facial expression analysis, human body modeling, motion capture and gait/gesture analysis.
Email: fofli@ku.edu.tr

A. Murat Tekalp is a professor at Koç University. His research interests are in digital image and video processing. Tekalp received an MS and a PhD in electrical, computer, and systems engineering from Rensselaer Polytechnic Institute. He received a Fulbright Senior Scholarship in 1999 and the Tübitak Science Award in 2004. Tekalp, editor in chief of the EURASIP journal Signal Processing: Image Communication, authored Digital Video Processing (Prentice Hall, 1995). He holds seven US patents and is a Fellow of the IEEE.
Email: mtekalp@ku.edu.tr

Joëlle Tilmanne holds an Electrical Engineering degree from the Faculté Polytechnique de Mons (FPMs, Belgium) since June 2006. She did her master thesis in the field of sleep signals analysis, at Lehigh University (USA). She is pursuing a PhD thesis in the TCTS lab of FPMs since September 2006, in the field of HMM based motion synthesis.
Email: joelle.tilmanne@fpms.ac.be

Yücel Yemez is an assistant professor in the Computer Engineering Department at Koç University. His current research is focused on various fields of computer vision and 3D computer graphics. Yemez received a BS from Middle East Technical University, Ankara, Turkey, and an MS and PhD from Boğaziçi University, İstanbul, Turkey, all in electrical engineering.
Email: yyemez@ku.edu.tr