International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.2 (2014), pp.265-280
http://dx.doi.org/10.14257/ijmue.2013.9.2.27

Motion Tracking, Retrieval and 3D Reconstruction from Video

Hashim Yasin, Björn Krüger, Andreas Weber
Department of Computer Science II,
University of Bonn,
Bonn, Germany.
{*yasin, kruegerb, weber*}*@cs.uni-bonn.de*

Abstract

The work at hand presents a novel data-driven framework for 3D full body human motion reconstruction from uncalibrated monocular video data. To this end, we develop a knowledge base by taking 2D samples of the motion capture library from different viewing directions. This allows later steps to handle 2D query videos without any information on the viewing direction. We detect and track features from input video sequences by utilizing low-level image based feature detection techniques like MSER and SURF. This process is stabilized by back projection of high-level 3D prior information obtained from the motion capture library to the image plane. Extraction of suitable feature sets from both, input control signals and motion capture data, enables us to retrieve the best relevant prior poses from the motion capture library by employing fast motion retrieval techniques. Finally, 3D motion sequences are reconstructed by non-linear energy minimization, that takes into account multiple prior terms. Furthermore, we propose a method to estimate camera parameters from input video itself and sampling of motion capture library.

Keywords: Feature detection and tracking, Motion retrieval, Camera parameters, 3D motion reconstruction.

1. Introduction

Motion reconstruction from uncalibrated video data has been remained a major research topic in the last decade. A lot of techniques have been proposed but one of the most widespread and successful approaches for 3D motion reconstruction is the data-driven approach that makes use of a knowledge base which might be developed from a huge motion capture library just like in our case. Although a bulk of research has already been conducted in this specific direction, yet there are still open challenges that have to be tackled by further research in order to meet the massive demands of growing industries like animated characters in movies, computer games, medical sciences and sport sciences etc.

In this paper, we present a 3D motion reconstruction approach from static uncalibrated monocular video data. As we are dealing with 2D input signals, having no information on the actual viewing direction, we first sample our motion capture database into various 2-dimensional viewing directions as combination of different azimuth and elevation angles in a preprocessing step. The input video stream is prepared to be a query to our system by

ISSN: 1975-0080 IJMUE

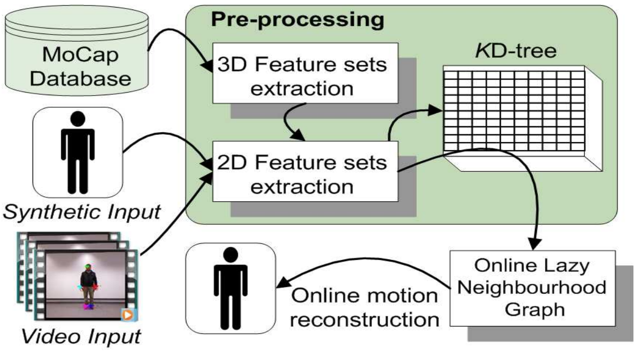

Pre-processing
MoCap Database
3D Feature sets extraction
KD-tree
2D Feature sets extraction
Synthetic Input
Video Input
Online Lazy Neighbourhood Graph
Online motion reconstruction

Figure 1. System overview diagram.

detecting and tracking feature sets. Here, we utilize Maximally Stable Extremal Regions (MSER) and Speeded Up Robust Features (SURF) feature detection techniques together with some prior knowledge obtained from the motion capture (MoCap) library. By using this prior knowledge from the database, we are able to make feature detection and tracking more robust. After getting suitable feature sets both from input signals as well as from motion capture database, we are able to perform efficient similarity search and retrieve nearest neighbours from database. For that purpose, we have developed a kd-tree data structure and the so called *online lazy neighbourhood graph* (OLNG) along the lines of Tautges et al. [1]. In our domain, we have adapted these methods to work with 2D feature sets extracted either from MoCap data or video data used as control input signal.

Additionally, we have addressed the problem of camera parameter estimation from uncalibrated monocular video input in this work. Camera parameters are estimated from the input video and from samples of MoCap database at different viewing directions. More precisely, we extract translational information from the input control video signal and the orientation information from samples of the MoCap database, that are similar according to the derived feature sets. At the end, the 3D human motion sequences are reconstructed by gradient-based energy minimization process that considers multiple energy prior terms. We have tested our developed algorithm on real video data as well as synthetically generated 2D input signals.

2. Related Work

Motion retrieval and reconstruction in 3-dimensions from 2-dimensional video input is the current strand of research. Some general methods to reconstruct human motion found in prior literature can be categorized into data-driven approaches either generative or discriminative like in [2, 3, 4, 5, 6, 7], geometric constraint based approach as in [8, 9] and physics based modeling [10, 11]. Sminchisescu et al. [2] estimate full body human motion using discriminative density propagation approach. A compact conditional Bayesian mixture of experts models is utilized in order to learn multi-modal conditional distributions. They synthesize human configurations together with rendered 2D silhouettes using MoCap data and 3D human model. Hornung et al. [12] animate 2D pictures with the help of user interaction in the form of selection of joints as well as the use of prior information in 3D motion capture database. They have presented shape deformation method in order to animate the still image projectively. Wei and Chai in [9] reconstruct 3D human pose by using a set of geometric constraints from 2D images. They estimate human skeleton as well as camera parameters with weak perspective model in order to reconstruct 3D pose. They optimize their reconstruction framework with two steps gradient based optimization process without use of some data-driven prior knowledge. Rosenhahn et al. [13] employ data-driven geometric ground plane prior constraints for human movement patterns in the process of pose tracking. Vondrak et al. [10] perform Bayesian filtering based human motion tracking by full body physics based dynamic simulation priors together with interpolation of joint data. Wei and Chai in [11] reconstruct human motion form video input data by employing physics-based modeling and minimal user interaction to annotate key intermediate frames for tracking error correction. Yasin et al. [14] first detect and track video features by constructing a dictionary of features (DoF) using MSER and SURF feature detection techniques and make data-driven 3D reconstruction from these 2D detected feature sets. Dantone et al. [15] estimate 2D human pose from still images by non-linear body part dependent joint regressor using two layered random forests. First layer classifies different independent body parts while second layer contributes to predict joint locations. Jain et al. [16] reconstruct 3D animation from 2D hand drawn animated characters and make interaction between 3D reconstructed character and a virtual world. First, they use user defined orthographies camera model and then they estimate camera by minimizing the geometric projection error. They formulate their data-driven reconstruction approach by using three energy terms like input-match term, motion prior term and regularization term.

Krüger et al. [17] develop lazy neighbourhood graph (LNG) for similarity search into the motion capture database. They employ 15-dimensional feature sets based on positions of hands, feet and head for fast similarity search. Later on, Tautges et al. [1] enhance the LNG into incremental online version named online lazy neighbourhood graph (OLNG) and reconstruct human motions using sparse accelerometer data. They reconstruct human motions with the help of data-driven prior model which measures a-priori likelihood of the input motion into MoCap database.

3. Feature Design and Tracking

In this section, we demonstrate how we search into motion capture library for similar motion segments of video input or synthetically generated input, with the help of some suitable feature sets. We discuss both scenarios one by one in detail as follows.

3.1. Motion Capture Data

We make no any assumption regarding the 2D view direction, so we have sampled our database at various 2D viewing directions. For database sampling and construction of our knowledge base, we first extract 3D feature sets $\mathcal{F}^{15}_{3\mathrm{D}}$ based on positions of hands, feet and head like Krüger et al. have done in [17]. We project these 3D feature sets $\mathcal{F}^{15}_{3\mathrm{D}}$ onto 2D plane at different viewing directions similar to Yasin et al. [14] and as a consequence, we get 2D feature sets $\mathcal{F}^{10}_{2\mathrm{D}}$. To achieve previously mentioned goal, we have specified azimuth angles from 0 to 350 degrees with 10 degree step size and elevation angles from 0 to 90 degrees with step size 15 degree, as shown in Figure 2. The extracted 2D feature sets are further normalized by translating the mean (center of mass) of the feature sets to its origin of the coordinate system. In this way, we get such type of feature sets in our knowledge

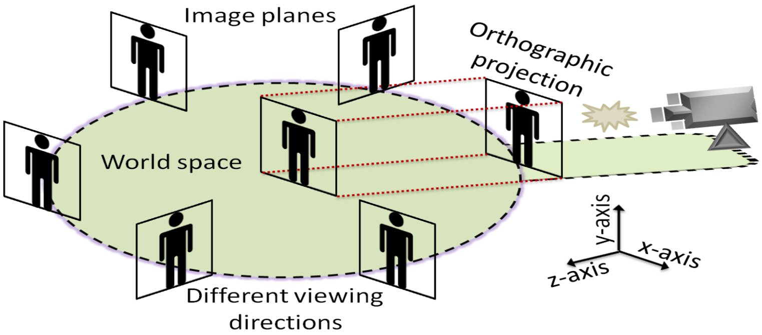


Figure 2. Sampling of MoCap database onto 2D image plane at different viewing directions on which basis kd-tree has been developed.

base which are comparable to the feature sets extracted from input video, where we have no any root node and articulated skeleton of the performing actor. As a result, we can perform comparison between both types of feature sets, extracted either from MoCap data or input video stream. Sampling of database or more precisely developing of knowledge base by projection of 3D feature sets at different viewing directions supports us in order to estimate orientation information of the performing actor in video input stream.

Our developed system takes into account only the positions of hands, feet and head to access the relevant information from MoCap database, so we have prepared our query in the format of 2D feature sets $\mathcal{F}^{10}_{\text{2D}}$ based on positions of hands, feet and head. The 2D feature sets, synthetically generated from the motion clip of the MoCap database and given as input, are named here as *synthetic data*.

3.2. Video Data

Similarly, in case of video data, we prepare our query input signal first based on extracted feature sets from video input streaming. For that purpose, we detect and track the positions of hands, feet and head only in order to develop video based feature sets $\mathcal{F}^{10}_{\text{vid}}$.

3.2.1. Feature Detection and Tracking

For feature detection and tracking, we employ together low-level image based feature detection techniques like MSER and SURF, and high-level 3D prior information available in MoCap database. First, we extract features by utilizing these MSER and SURF feature detection techniques and develop a dictionary of features (DoF) along the lines of Yasin et al. [14]. We stabilize feature detection process by 3D prior information retrieved from MoCap database by developing a kd-tree and the online lazy neighbourhood graph. This retrieved prior information (k-nearest neighbors) is back projected onto current image frame by a weak perspective camera model to make the feature detection and tracking more robust. The continuous process of detection and tracking is performed by matching feature sets of the current image frame with already extracted feature sets of the previous frames collected in DoF, and with addition of back projection of extracted 3D prior information.

For high-level feature detection and tracking, we formulate Bayes decision function $D = \{d_1, \ldots, d_I\}$ to obtain similar features' patterns $\mathbf{x}$ from current image frame of input video,

$$\mathrm{d}_i(\mathbf{x}) = p(\mathbf{x}|w_i)P(w_i). \tag{1}$$

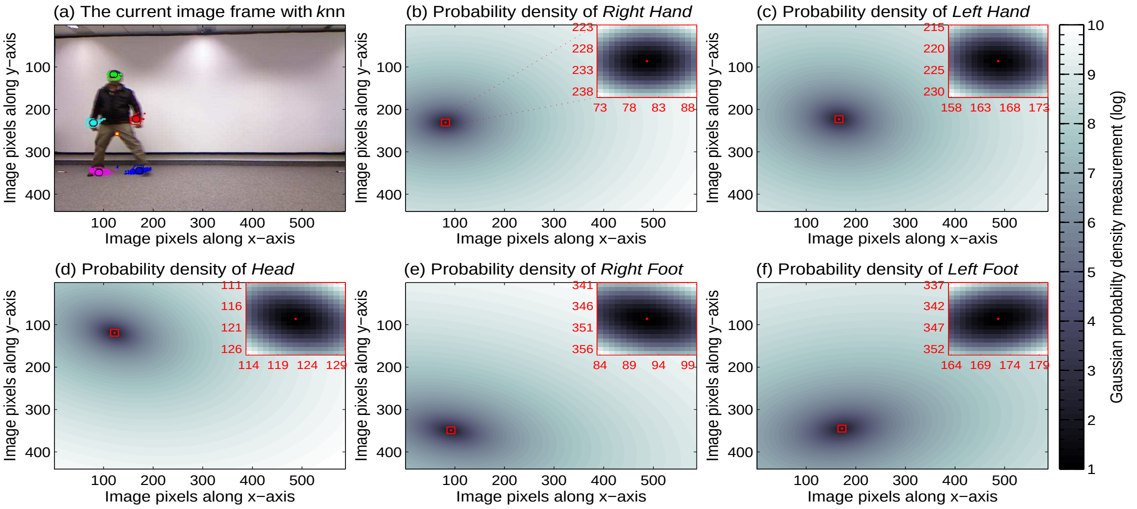


Figure 3. This figure shows probability density measurement of each end effector, used in process of feature detection and tracking. The current image frame extracted from video input query, with $k$nn back projected into image plane is represented in (a). The Probability density measurement of each end effectors is represented in: (b) *Right Hand*. **(c)** *Left Hand*. **(d)** *Head*. **(e)** *Right Foot*. **(f)** *Left Foot*.

where $P(w_i)$ is the prior probability of the class w_i with I number of classes $W = \{w_1, \ldots, w_I\}$ and $p(\mathbf{x}|w_i)$ is the h-dimensional Gaussian probability density function of the pattern vector and is calculated with Mahalanobis distance as,

$$p(\mathbf{x}|w_i) = \frac{1}{(2\pi)^{\frac{h}{2}}\sqrt{|\Omega_i|}} \cdot e^{-\frac{1}{2}(\mathbf{x}-\mu_i)^T\Omega_i^{-1}(\mathbf{x}-\mu_i)}. \tag{2}$$

where $|\Omega|$ is the determinant of covariance matrix Ω, μ is the mean vector, $(\mathbf{x}-\mu_i)^T$ is the transpose of the difference between the features' pattern vector $\mathbf{x}$ and the mean vector μ, and h is the dimensions of feature vector $\mathbf{x}$. Some examples of the Gaussian probability density measurement for each end effector (right hand, left hand, right foot, left foot and head) has been explored in the Figure 3, where the Gaussian probability density in log has been color coded. The darker region shows higher probability density of the end effector in the current image frame. On the basis of Bayes decision function, we select those pixels of the current image frame for feature sets, that explore the largest Bayes decision function value. By combining low-level (MSER and SURF) and high-level (back projection of 3D prior information retrieved from MoCap database) feature detection techniques, the feature sets retrieval equation has become in the form of,

$$\mathcal{F}^{10}_{\text{vid}} = \alpha\Psi + \beta\Phi. \tag{3}$$

where Ψ represents the feature sets obtained through MSER and SURF feature detection techniques, Φ represents the feature sets obtained from back-projection of 3D prior information, α and β are the user defined weights.

3.2.2. Camera Model

In this paper, we assume static monocular uncalibrated camera and deal with a weak perspective camera model. We estimate our camera model parameters from input video signal as well as from our knowledge base developed by sampling of the MoCap database at different viewing directions. We estimate parameterized camera projection matrix ρ which consists of intrinsic calibration K, extrinsic orientation $\mathsf{R}_{(\alpha,\beta,\gamma)}$ and translation $\mathsf{T}_{(\mathsf{x},\mathsf{y},\mathsf{z})}$.

$$\rho = \mathsf{K}\left[\mathsf{R}_{(\alpha,\beta,\gamma)} \mid \mathsf{T}_{(\mathsf{x},\mathsf{y},\mathsf{z})}\right]. \tag{4}$$

Intrinsic Camera Parameters, K. For intrinsic camera parameters with 4 degree of freedom in our case, we have computed the focal length by employing the 3D and 2D information of first frame. The skew coefficient is fixed to be zero. The scaling factor is updated across the M number of frames. We consider mean of J number of feature sets $\mathcal{F}^{10}_{\mathrm{vid}}$ to coincide principal points $\mathsf{O} = \{\mathsf{o}^1_\mathsf{x}, \ldots, \mathsf{o}^\mathsf{M}_\mathsf{x}, \mathsf{o}^1_\mathsf{y}, \ldots, \mathsf{o}^\mathsf{M}_\mathsf{y}\}$, which are updated regularly across M number of image frames. In other words, the principle point O^m at frame m is,

$$\mathsf{O}^\mathsf{m} = \left[\tfrac{1}{\mathsf{J}} \textstyle\sum_{\mathsf{j}=1}^{\mathsf{J}} (\mathcal{F}^{10}_{\mathrm{2D}})_\mathsf{j}\right]^\mathsf{m}. \tag{5}$$

Extrinsic Camera Parameters, $\mathsf{R}_{(\alpha,\beta,\gamma)}$, $\mathsf{T}_{(\mathsf{x},\mathsf{y},\mathsf{z})}$. In case of extrinsic camera parameters which has 6 degree of freedom, we extract orientation information $\mathsf{R}_{(\alpha,\beta,\gamma)}$ from our knowledge base and translation information $\mathsf{T}_{(\mathsf{x},\mathsf{y},\mathsf{z})} = \{\mathsf{t}^1_\mathsf{x}, \ldots, \mathsf{t}^\mathsf{M}_\mathsf{x}, \mathsf{t}^1_\mathsf{y}, \ldots, \mathsf{t}^\mathsf{M}_\mathsf{y}, \mathsf{t}^1_\mathsf{z}, \ldots, \mathsf{t}^\mathsf{M}_\mathsf{z}\}$ from input video. On the basis of regularly updated principle point, scaling factor and focal length, we get an estimation of the translation. For example, translation along x and y-directions $\mathsf{T}^\mathsf{m}_{(\mathsf{x},\mathsf{y})}$ at current image frame m is calculated as,

$$\mathsf{T}^\mathsf{m}_{(\mathsf{x},\mathsf{y})} = \mathsf{O}^\mathsf{m} - \mathsf{O}^1. \tag{6}$$

where $\mathsf{O}^1 = \{\mathsf{o}^1_\mathsf{x}, \mathsf{o}^1_\mathsf{y}\}$ is the principal point of the first frame and $\mathsf{O}^\mathsf{m} = \{\mathsf{o}^\mathsf{m}_\mathsf{x}, \mathsf{o}^\mathsf{m}_\mathsf{y}\}$ is the principal point at frame number m. This translation is added to the normalized reconstructed pose to get relevant reconstructed translated motion.

We estimate orientation information from our knowledge base which is developed on the basis of sampling of MoCap database at different viewing directions (azimuth and elevation angles) as mentioned earlier. We obtain N number of best poses from the MoCap database through our developed knowledge base at various viewing directions and build a histogram on the basis of these retrieved indices of best poses. The top three peaks of the histogram are selected to be candidates of the current azimuth angle of the performing actor. That azimuth angle has been selected among the candidates, which is very close to already selected azimuth angles for previously two reconstructed poses. We have smoothen these selected azimuth angles with low-pass filter as well.

4. Motion Retrieval

As we are dealing with 2-dimensional scenario, so we sample our database into different views as mentioned in Subsection 3.1. To retrieve similar poses from the MoCap database, 2-dimensional feature sets, extracted either from real video data or synthetically generated sequences, are given as input to the system. For this similarity search into database, we develop a kd-tree, build upon 2-dimensional feature sets $\mathcal{F}^{10}_{\mathrm{2D}}$, and a graph structure, online lazy neighbourhood graph, in a similar fashion as described in [1, 17]. In our case, we are dealing with sparse continuous stream of 2D input data (video data or synthetic data)

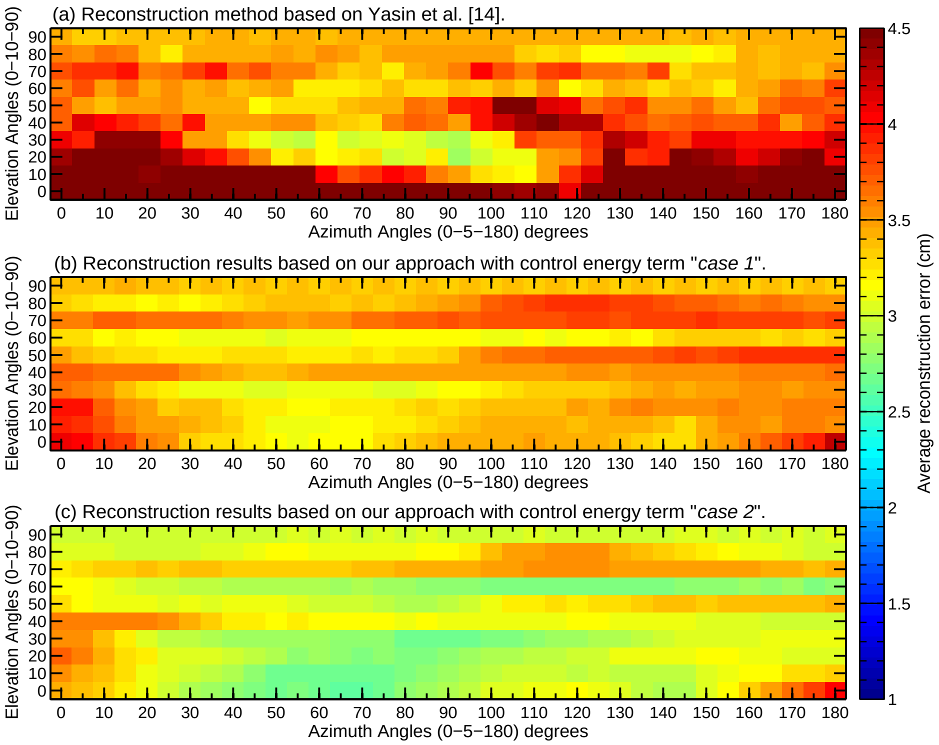


Figure 4. Average reconstruction error graph for walking motion, with different viewing directions — azimuth angles (0-5-180) and elevation angles (0-10-90). (a) Yasin et al. method [14]. (b) Our reconstruction method with control energy term *case 1*. **(c) Our reconstruction method with control energy term** *case 2*.

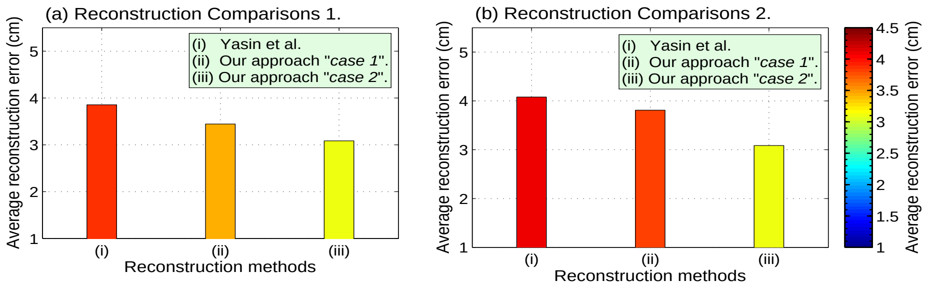


Figure 5. (a) Reconstruction methods' comparison by taking average of all reconstructions at different views for same motion class. (b) Reconstruction methods' comparison by taking average of all reconstructions of different motions at fixed azimuth and elevation angles (60 and 45 degrees).

rather than using accelerometer input data as in [1]. As a result of similarity search, we obtain k-nearest neighbours ($k$nn) from MoCap database. We select the best N poses from these retrieved nearest poses through OLNG by considering step sizes and minimum cost of the paths of k-nearest neighbours. The cost attached to the pose's path is the result of the input feature sets' similarities. In the construction of OLNG, we consider only those paths which have minimum costs. On the cost basis, we have build the penalty normalized weights as,

$$W_n^t = 1 - \frac{G_n^t - min(G^t)}{max(G_n^t - min(G^t))} \qquad n = 1, 2, \ldots, N. \tag{7}$$

where W_n^t and G_n^t are the associated weights and costs for the selected paths at current frame t respectively. The online version of lazy neighbourhood graph imposes the fact that there is no any need to construct whole graph structure for every frame cycle, rather it is more efficient to build graph incrementally considering previously constructed paths of minimum cost. In our experiments, the value of k for nearest neighbours is fixed to be 2^{12} and the value of N for best poses is set to be 256. These values might vary depending on the size of the MoCap database.

5. Online Motion Reconstruction

We have developed our reconstruction approach on the basis of low dimensional 2D feature sets obtained from video data and MoCap data. We retrieve similar poses in the form of joints angle configurations $Q^t = \{\mathbf{q}_1^t, \ldots, \mathbf{q}_N^t\}$ as well as in relevant positional information $X^t = \{\mathbf{x}_1^t, \ldots, \mathbf{x}_N^t\}$ from MoCap database for current frame t. Like Tautges et al. [1], we have taken into account square root kernel in order to estimate probability density for local modeling in contrast to multivariate normal distribution. Our data-driven approach takes 2D feature sets and reconstructs into input like motion frame by frame. We have formulated our reconstruction methodology as energy minimization problem and solved with gradient descent based optimization.

$$P_{\text{rec}} = argmin(w_p E_p + w_j E_j + w_s E_s + w_c E_c). \tag{8}$$

where P_{rec} is the reconstructed pose and w_p, w_j, w_s and w_c are user defined weights associated with energy terms and are computed accordingly.

5.1. Pose Energy Term

It measures a-priori likelihood of the synthesized joints angle configuration, called as synthesized pose, from the MoCap library. The pose has been synthesized in low dimensional principal component analysis (PCA) space from N best retrieved poses and then synthesized pose gets minimized from these retrieved N the best poses as well. This energy term demonstrates that how likely the synthesized pose is according to the prior joints angle configurations already exists in the MoCap database. As mentioned earlier, we formulate symmetric square root kernel function $\mathcal{K}$ to estimate probability density as,

$$P_p \propto \sum_{n=1}^{N} w_n^t \cdot \mathcal{K}(|\tilde{\mathbf{q}}_n^t - \mathbf{q}^t|). \tag{9}$$

where w_n^t are the normalized weights mentioned in Section 4, $\tilde{\mathbf{q}}_n^t$ is the joints angle configurations of the best retrieved poses including previously reconstructed pose and $\mathbf{q}^t$ is the joints angel configuration obtained in a PCA space at current frame t. For energy term, the Equation 9 is reformulated as,

$$E_p = \sum_{n=1}^{N} w_n^t \cdot \sqrt{|\tilde{\mathbf{q}}_n^t - \mathbf{q}^t|}. \tag{10}$$

5.2. Joint Energy Term

This energy term play a vital role in 2D and 3D positions correspondence and minimize the unwanted artifacts arises due to 2D-3D transformation. It basically compels the joints positions, resulted from the forward kinematic of the synthesized pose, according to the prior true joints positional information of the best extracted poses from the MoCap library,

$$E_j = \sum_{n=1}^{N} w_n^t \cdot \sqrt{|\tilde{\mathbf{x}}_n^t - \mathbf{x}^t|}. \tag{11}$$

where $\tilde{\mathbf{x}}_n^t$ is the joints positions of the N best poses and $\mathbf{x}^t$ is the position vectors of the current synthesized pose at frame t.

5.3. Smooth Energy Term

In order to avoid the jittering and jerkiness effects, smoothness energy term is introduced. It imposes smoothness in a way that newly reconstructed pose has been bound to be according to the previously two reconstructed poses as well as the prior knowledge about smoothness between neighbouring candidates exists in MoCap database. Mathematically,

$$E_s = \sum_{n=1}^{N} w_n^t \cdot \sqrt{|\tilde{\mathbf{S}}_n^t - \mathbf{S}^t|}. \tag{12}$$

where $\tilde{\mathbf{S}} = \tilde{\mathbf{x}}^t - 2\tilde{\mathbf{x}}^{t-1} + \tilde{\mathbf{x}}^{t-2}$ with position vectors $\tilde{\mathbf{x}}^t$, $\tilde{\mathbf{x}}^{t-1}$ and $\tilde{\mathbf{x}}^{t-2}$ of the N best retrieved poses; and $\mathbf{S} = \mathbf{x}^t - 2\mathbf{x}^{t-1} + \mathbf{x}^{t-2}$ with position vectors $\mathbf{x}^t$, $\mathbf{x}^{t-1}$ and $\mathbf{x}^{t-2}$ of the reconstructed poses at frames t, $t-1$ and $t-2$ respectively.

5.4. Control Energy Term

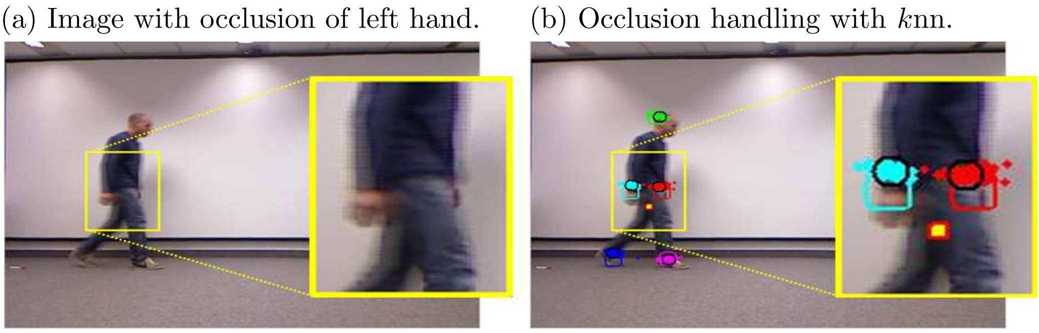
This energy term compels the synthesized motion to be according to the performed input motion. More precisely, with control energy term, we minimize the distance between feature sets of the input motion and the synthesized motion. We have performed control energy minimization in two ways as;

Case 1. In first case, we are dealing with 3-dimensional scenario. We extract 3D feature sets of hands, feet and head of current synthesized pose, represented as $\hat{\mathbf{x}}_{3d}$, and get minimized with the extracted 3D feature sets of the already reconstructed previous pose as,

$$E_c^1 = \sqrt{|\hat{\mathbf{x}}_{3d}^t - \hat{\mathbf{x}}_{3d}^{t-1}|}. \tag{13}$$

Case 2. In case 2, we project 3D feature sets of hands, feet and head of the current synthesized pose into 2D plane at specific view direction which is obtained from the process of camera parameters estimation, and normalize these 2D feature sets too. The estimated 2D feature sets $\hat{\mathbf{x}}_{est,2d}^t$ from input motion t are then get subtracted from these normalized 2D feature sets $\hat{\mathbf{x}}_{2d}^t$ of current frame t as,

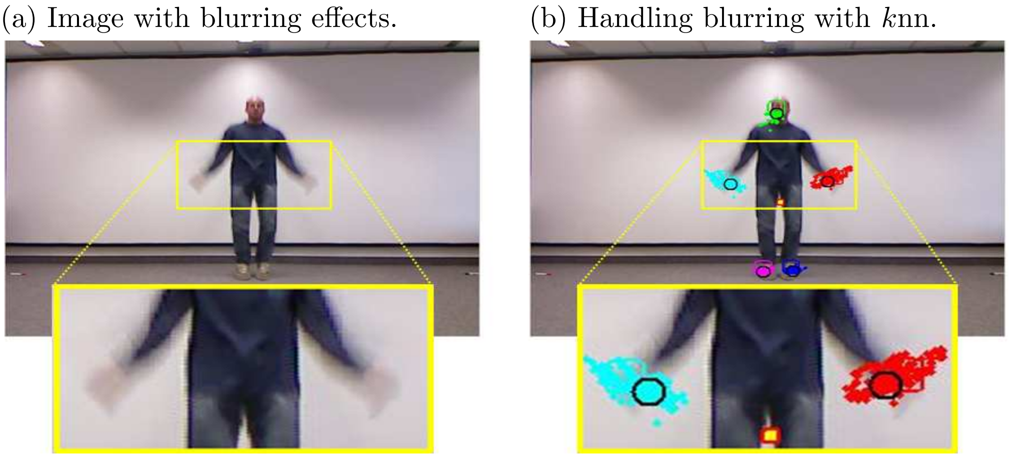
$$E_c^2 = \sqrt{|\hat{\mathbf{x}}_{2d}^t - \hat{\mathbf{x}}_{est,2d}^t|}. \tag{14}$$



(a) Image with occlusion of left hand.

(b) Occlusion handling with $k$nn.

Figure 6. The process of occlusion handling by 3D prior information. (a) Image frame shows occlusion of left hand which is totally disappeared. (b) Image frame shows that how occlusion has been handled by back projection of 3D prior $k$nn.



(a) Image with blurring effects.

(b) Handling blurring with $k$nn.

Figure 7. The process of handling blurring effect by 3D prior information. (a) Image frame showing blurring effect for the hands' positions. (b) Image frame shows that how blurring effect is handled by back projection of 3D prior $k$nn.

6. Results and Analysis

We utilize HDM05 [18] database as MoCap library which is publically accessible. There are five different actors who perform various types of motion. HDM05 is a heterogeneous database with sampling rate 120Hz. We have down sampled our database to sampling rate 30 Hz to make it equivalent to the sampling rate of input video stream. With 30 Hz frame rate, the database consists of more or less 381,157 frames. For video input query, we have recorded motions using Kinect RGB camera with resolution 587×440 pixels and frame rate 30 frames per second. We have testified our approach on variety of motions like straight walking, side walking, walking in a circle, jumping jack and cartwheel motions etc.

6.1. Synthetic Data

We extract 2D data from the MoCap data and name it as synthetic data as mentioned earlier in Subsection 3.1. This synthetically generated 2D data is then given as input to the system for 3D reconstruction. We have evaluated our approach in different ways as;

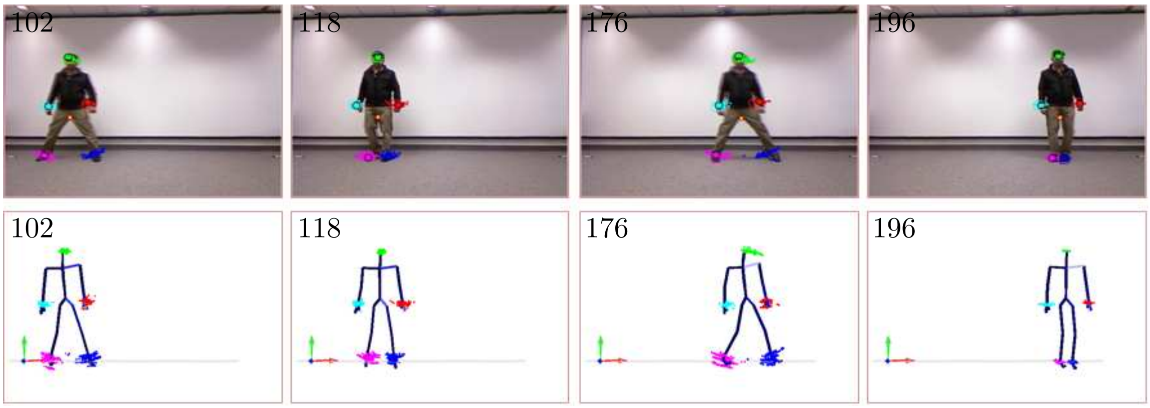


Figure 8. Tracking and reconstruction results of our approach *case 2* of *side-walking-left* input video motion with extracted $k$nn from MoCap library.

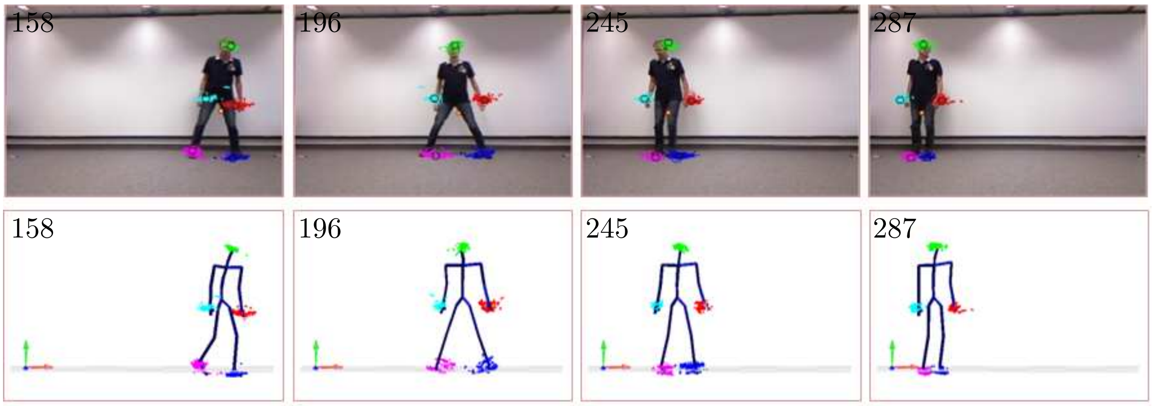


Figure 9. Tracking and reconstruction results of our approach *case 2* of *side-walking-right* input video motion with extracted $k$nn from MoCap library.

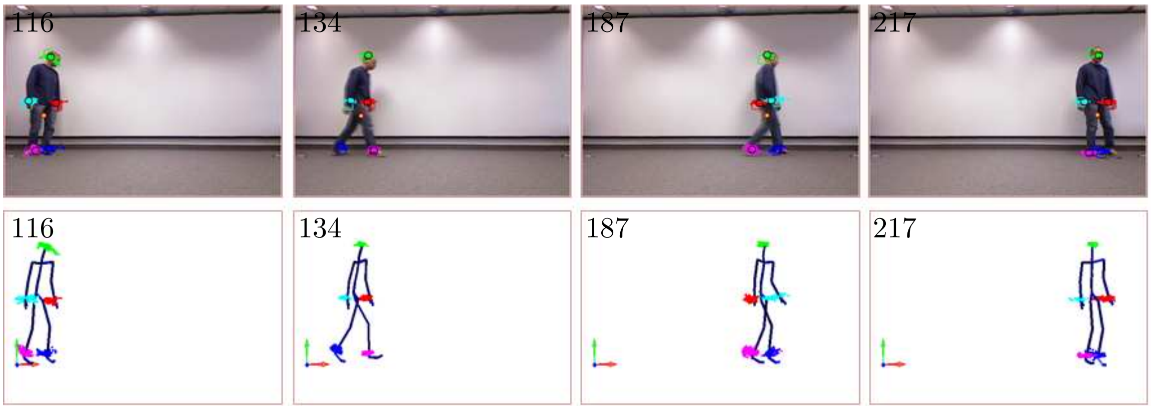


Figure 10. Tracking and reconstruction results of our approach *case 2* of *walking-left* input video motion with extracted $k$nn from MoCap library.

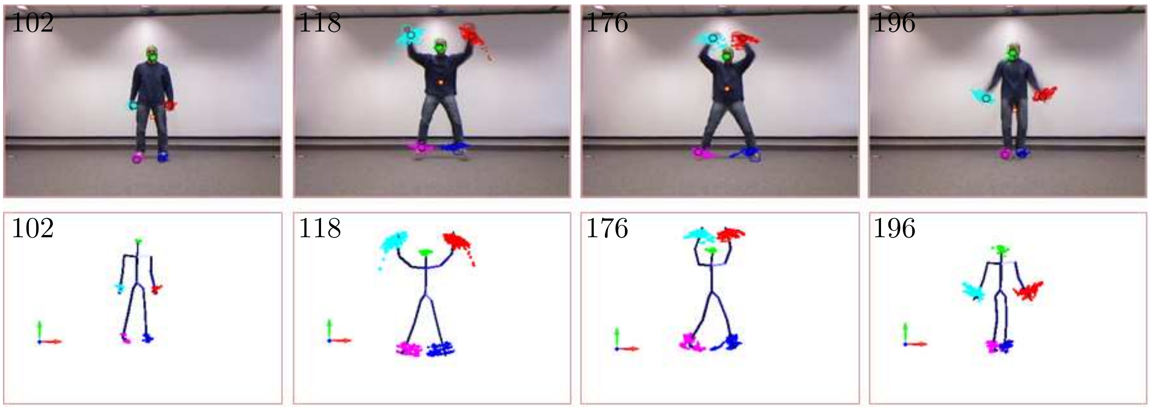


Figure 11. Tracking and reconstruction results of our approach *case 2* **of** *jumping jack* **input video motions with extracted** k**nn from MoCap library.**

Firstly, we have constructed the average reconstruction error graphs where azimuth view directions are given along x-axis and elevation angles are drawn along y-axis. The reconstruction error has been reported by the color range from red to blue as shown in Figure 4. We have made comparison of our approach with the results mentioned in Yasin et al. [14]. We have checked the reconstruction results at various combinations of azimuth angles and elevation angles. The azimuth angles range from 0 to 180 degree with 5 degree step size combined with elevation angles which range from 0 to 90 degree with 10 degree step size. From the results, it is quite obvious that the reconstruction has been improved when we employ proposed approach *case 1* with 3D feature sets for approximately every combination of azimuth and elevation angles. We further improve our results when we use proposed approach *case 2* with normalized 2D feature sets $\hat{\mathbf{x}}^t_{2d}$ and estimated 2D feature sets $\hat{\mathbf{x}}^t_{est,2d}$. The developed system with *case 2* outperforms and shows the best reconstruction results for more or less every view direction (combination of azimuth and elevation angles), as obvious in Figure 4. The proposed approach *case 2* with 2D scenario executes the fact that knowing the 2D view direction and then performing energy minimization in 2-dimensions enhances the reconstruction results as compared to the approach *case 1* with 3D scenario when we are dealing with monocular camera.

Secondly, we check our methods' performance by combining the all reconstruction results at different viewing directions and perform comparison as shown in Figure 5 (a). We take average of reconstruction results at all combination of azimuth (starting from 0 degree to 180 degree with 5 degree step size) and elevation angles (from 0 degree till 90 degree with 10 degree step size) for the same motion class. From the results, it is clear that the proposed approach with *case 2* gives the best results.

Thirdly, we explore our results other way around by taking average of the reconstruction results of different types of motions like walking, jumping jack and cartwheel motions and keep the azimuth angle fixed to 60 degree and elevation angles fixed to 45 degree and then make comparison again as in Figure 5 (b). From these various experiments, we conclude that proposed approach *case 2* with normalized 2D feature sets comparatively performs better, when synthetically generated 2D data has been given as input query. The further details of the reconstruction results based on synthetically generated data can be seen in supplementary material found in [19].

6.2. Video Data

We have also assessed our proposed framework on uncalibrated monocular video data. For real video data, we have first detect and track video based feature sets and prepare input query. Thus, the ultimate reconstructed sequences depends upon not only the reconstruction methodology but also the fact that how accurately the feature sets are detected and tracked in order to prepare the input query. For feature set detection and tracking, we have improved detection as well as tracking results through 3D prior existing knowledge available in motion capture library as compared to the work of Yasin et al. [14]. We extract 3D prior knowledge and back project them into 2-dimensional image plane. In this way, we are able to handle occlusion and blurring effects somehow. Some examples have been drawn in Figure 6 and Figure 7. In Figure 6, the left hand of the performing actor is completely occluded but with back projection of 3D prior knowledge, we are able to track occluded left hand accurately. Similarly, in Figures 7 the hands have lost their structure due to fast movement and blurring effects and low-level image based feature detection techniques mistrack the positions of hands but the use of 3D prior k-nearest neighbours bound the feature detection process to detect and track hands successfully. In this way, we have dealt with occlusion and blurring effects successfully to some extant but still sometime mistracking of feature sets may occur due to occlusion, illumination and blurring effects which affect reconstruction results too. We have observed from the experiments that the mistracking is roughly 20-25 percent on an average of the total number of frames without use of 3D prior existing knowledge, which is reduced to less than roughly 8-10 percent on an average by the use of 3D data. The mistracked feature sets are then corrected manually.

We estimate camera parameters form video input as well as knowledge base. To this end, we sample our knowledge base at different viewing directions (azimuth and elevation angles). From experiments, we observe that we get more accurate orientation information when we sample MoCap database with smaller step sizes of azimuth and elevation angles.

After detection and tracking of 2D video feature sets, we have reconstructed input like motion on the basis of these video based feature sets. No doubt, the improvement in the process of detection and tracking enhances the accuracy in reconstruction results, but our proposed reconstruction approach also contributes a lot in making the reconstruction accurate. This is quite obvious when we use the synthetically generated data as input query. In case of real input video query, the proposed approach *case 2* gives the best reconstruction results as well. Some examples of tracking and reconstruction of different types of motion are shown in Figures 8 to 11. More results of feature detection, tracking and reconstruction based on video input query can be seen in supplementary material found in [19].

7. Conclusion and Future Work

We have proposed an efficient data driven reconstruction approach from video data by constructing kd-tree data structure and online lazy neighbourhood graph, taking into account just the positions of hands, feet and head. Our proposed framework first design and normalize feature sets from both video input as well as MoCap data to make them comparable. From input video stream, we have detected and tracked feature sets by the use of 3D prior knowledge already available in MoCap library. In this way, we have made low-level image based feature detection and tracking process more robust. We estimate camera parameters successfully for 2D and 3D correspondence from video itself and from knowledge base which is created by sampling of MoCap data at different viewing directions. The experimental results indicates that our proposed approach has outperformed comparatively. Our system performs reconstruction with frame rate approximately 5-6 frames per second.

For future work, we are planning to track and reconstruct human motion in more complex scenarios like human motion in outdoor environments with non-static background and moving camera. The low-level image feature extraction techniques might be made more robust and efficient. The camera model parameters can be made more precise and accurate by performing the process of optimization for estimation of intrinsic as well as extrinsic camera parameters frame by frame. Another important aspect of future work might be to extend the the reconstruction scenario of human motions to quadruped motions where firstly we have to select some different types of feature sets on which basis the process of reconstruction can be performed.

References

[1] J. Tautges, A. Zinke, B. Krüger, J. Baumann, A. Weber, T. Helten, M. Müller, H.-P. Seidel, and B. Eberhardt, “Motion reconstruction using sparse accelerometer data,” ACM Trans. Graph., vol. 30, pp. 18:1–18:12, May **(2011)**.

[2] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, “Discriminative Density Propagation for 3D Human Motion Estimation,” in IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 390–397, **(2005)**.

[3] Y.-L. Chen and J. Chai, “3d reconstruction of human motion and skeleton from uncalibrated monocular video,” in Computer Vision - ACCV 2009, vol. 5994 of Lecture Notes in Computer Science, pp. 71–82, Springer Berlin Heidelberg, **(2010)**.

[4] J. Valmadre and S. Lucey, “Deterministic 3d human pose estimation using rigid structure,” in Proceedings of the 11th European conference on computer vision: Part III, ECCV’10, (Berlin, Heidelberg), pp. 467–480, Springer-Verlag, **(2010)**.

[5] M. Salzmann and R. Urtasun, “Implicitly constrained gaussian process regression for monocular nonrigid pose estimation,” in Advances in Neural Information Processing Systems 23, pp. 2065–2073, **(2010)**.

[6] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt, “A data-driven approach for realtime full body pose reconstruction from a depth camera,” in IEEE 13th International Conference on Computer Vision (ICCV), pp. 1092–1099, IEEE, Nov. **(2011)**.

[7] V. Ramakrishna, T. Kanade, and Y. Sheikh, “Reconstructing 3d human pose from 2d image landmarks,” Computer Vision–ECCV 2012, pp. 573–586, **(2012)**.

[8] H. S. Park and Y. Sheikh, “3d reconstruction of a smooth articulated trajectory from a monocular image sequence,” in Proceedings of the 2011 International Conference on Computer Vision, ICCV ’11, (Washington, DC, USA), pp. 201–208, IEEE Computer Society, **(2011)**.

[9] X. K. Wei and J. Chai, “Modeling 3d human poses from uncalibrated monocular images,” in ICCV, pp. 1873–1880, **(2009)**.

[10] M. Vondrak, L. Sigal, and O. C. Jenkins, “Physical simulation for probabilistic motion tracking,” 2012 IEEE Conference on Computer Vision and Pattern Recognition, vol. 0, pp. 1–8, **(2008)**.

[11] X. Wei and J. Chai, “Videomocap: modeling physically realistic human motion from monocular video sequences,” ACM Trans. Graph., vol. 29, pp. 42:1–42:10, July **(2010)**.

[12] A. Hornung, E. Dekkers, and L. Kobbelt, “Character animation from 2d pictures and 3d motion data,” ACM Trans. Graph., vol. 26, Jan. **(2007)**.

[13] B. Rosenhahn, C. Schmaltz, T. Brox, J. Weickert, and H.-P. Seidel, “Staying well grounded in markerless motion capture,” in Proceedings of the 30th DAGM symposium on Pattern Recognition, (Berlin, Heidelberg), pp. 385–395, Springer-Verlag, **(2008)**.

[14] H. Yasin, B. Krüger, and A. Weber, “Model based full body human motion reconstruction from video data,” in 6th International Conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications (MIRAGE 2013), June **(2013)**.

[15] M. Dantone, J. Gall, C. Leistner, , and L. V. Gool., “Human pose estimation using body parts dependent joint regressors,” in CVPR, June **(2013)**.

[16] E. Jain, Y. Sheikh, M. Mahler, and J. Hodgins, “Three-dimensional proxies for hand-drawn characters,” ACM Trans. Graph., vol. 31, pp. 8:1–8:16, Feb. **(2012)**.

[17] B. Krüger, J. Tautges, A. Weber, and A. Zinke, “Fast local and global similarity searches in large motion capture databases,” in 2010 ACM SIGGRAPH / Eurographics Symposium on Computer Animation, SCA ’10, (Aire-la-Ville, Switzerland, Switzerland), pp. 1–10, Eurographics Association, July **(2010)**.

[18] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, “Documentation Mocap Database HDM05,” Tech. Rep. CG-2007-2, Universität Bonn, June **(2007)**.

[19] Department of Computer Graphics, University of Bonn, “Motion Tracking, Retrieval and 3D Reconstruction from Video,” **(2013)**. `http://cg.cs.uni-bonn.de/vmotrec`.

Authors

Hashim Yasin

He received his MS degree in Computer Software Engineering from National University of Science and Technology (NUST), Pakistan and Master degree in Computer Science from University of Bonn, Germany. He is currently doing his PhD in Computer Science from University of Bonn, Germany, in the research group of Prof. Dr. Andreas Weber on scholarship of Higher Education Commission, Pakistan and German Academic Exchange Service (DAAD). His research interests include video based motion tracking and reconstruction, image/video processing, computer vision and animation.

Björn Krüger

He received a Diploma (2006) and a doctorate degree (2012) in computer science from the University of Bonn. His research focuses on human motion data in computer animation. In this area his works cover a wide range starting from motion capturing and motion analysis up to motion synthesis.

Andreas Weber

He studied mathematics and computer science at the Universities of Tbingen, Germany and Boulder, Colorado, U.S.A. From the University of Tbingen he received his MS in Mathematics (Dipl.-Math) in 1990 and his PhD (Dr. rer. nat.) in computer science in 1993. From 1995 to 1997 he was working with a scholarship from Deutsche Forschungsgemeinschaft as a postdoctoral fellow at the Computer Science Department of Cornell University. From 1997 to 1999 he was a member of the Symbolic Computation Group at the University of Tbingen, Germany. From 1999 to 2001 he was a member of the research group Animation and Image Communication at the Fraunhofer Institut for Computer Graphics. Since April 2001 he has been professor of computer science at the University of Bonn, Germany.