Markerless Human Motion Capture from Voxel Reconstruction with Simple Human Model∗

Kazuhiro TAKAHASHI**, Yusuke NAGASAWA*** and Masafumi HASHIMOTO†
** Department of Information Systems Design, Doshisha University
1-3 Miyakodani Tatara, Kyotanabe, Kyoto 610-0321, Japan
E-mail: katakaha@mail.doshisha.ac.jp
*** Konami Digital Entertainment Co., Ltd. (ex-affiliation Graduate school of Doshisha University)
2-5-25 Umeda, Kita-ku, Osaka 530-0001, Japan
E-mail: nagasawa0707@yahoo.co.jp
† Department of Intelligent Information Engineering and Science, Doshisha University
1-3 Miyakodani Tatara, Kyotanabe, Kyoto 610-0321, Japan
E-mail: mhashimo@mail.doshisha.ac.jp

Abstract
This paper investigates a human body posture estimation method based on the back projection of human silhouette images extracted from multi-camera images. The multi-camera system is based on a server-client system with local network of 1000Base-T to achieve a voxel 3D reconstruction of human body posture in real-time. In order to extract significant points of the human body such as head, neck, shoulders, elbow joints, hands, waist, knee joints, and toes in 3D, an articulated cylindrical human model is applied to the voxel reconstruction of human body. To evaluate the proposed human body posture estimation method, 3D reconstruction experiments of human body posture and extraction experiments of human body’s significant points are carried out. The system runs in approximately real time (9 frames/sec with 50 × 50 × 50 voxel resolution) and the experimental results confirm both the feasibility and effectiveness of the proposed system in 3D human body posture estimation.

Key words : Motion Capture, Multi Camera, Voxel Reconstruction, Human Model, Image Processing

1. Introduction

Recently, demands for human motion analysis are increasing in various fields not only engineering but also entertainment, education, sports, medicine, and culture. Therefore, human motion capture has a wide range of application in human-machine interface systems, visual communications, virtual environments, and video game systems. In the interaction between humans and machines, real-time 3D human posture estimation is particularly important for designing a control strategy for an avatar or robot. In virtual reality, human motion information can be used to drive virtual environments. Many studies of human motion capture using computer vision(1)(2)(3)(4)(5)(6) have been undertaken and several kinds of visual cues utilized for human posture estimation from video images. Several convincing results have been achieved especially in 2D estimation. However, instead of working directly with 2D images, interest in using a 3D reconstruction with voxel obtained by multiple views as a basis for human posture estimation is increasing.

The shape-from-silhouette or visual hull that defines the volume of interest as the intersection of the 3D projections of its silhouettes is the most common 3D shape reconstruction method. Several studies have developed 3D shape reconstruction systems(7)(8)(9)(10) and motion capture systems based on the shape-from-silhouette approach to estimate the 3D shape and 3D posture parameters have been proposed(11)(12)(13)(14)(15)(16)(17)(18). Those systems have been used to capture accurate shape and posture parameters of a subject and work well under...
controlled conditions. However, human posture estimation with 3D reconstruction has been investigated on a large-scale system composed of synchronized multi-cameras and parallel processing on a PC cluster because of the heavy computational cost of real-time full 3D shape reconstruction and human body model fitting. On the other hand, this paper is motivated by achieving real-time 3D human posture estimation with simple system and simple estimation method of small computational cost as possible.

The authors previously proposed a human posture estimation method using a multi-camera system\(^{(19)(20)}\). As shown in Figure 1, the estimation method, which is based on projections of human silhouettes that correspond to the human body areas in the images (hereafter called a "silhouette image"), is composed of four processes: human silhouette extraction using background subtraction, back projection of the silhouette image from the 2D plane to 3D space, reconstruction of 3D human body posture using silhouette volume intersection, and estimation of 3D skeleton information. To achieve reconstruction of human body postures in real time, the multi-camera system composed of a server-client system and thread-programming technique with a dual CPU system was used in image processing. The 3D reconstruction with \(200 \times 200 \times 200\) voxel resolution ran in approximately 20 frames/sec using \(640 \times 480\) pixel resolution images from six cameras, however, estimating human body posture or the representative body points (top of the head, and tips of hands and feet) and joint positions (the elbow and knee) has not yet been investigated.

In this paper, estimating human body posture from 3D voxel reconstruction is presented in order to extract the 3D coordinates of the significant points (the representative points and the joints). An articulated cylindrical human model is introduced into our human body posture estimation system in order to obtain 3D skeleton information of a human body. In section 2, the algorithm of a 3D human body posture estimation is described. In section 3, experimental results of estimating 3D human body postures using a multi-camera system are presented.

![Fig. 1 Processing flow of 3D human body posture estimation using multi-camera system.](image)

### 2. 3D Human Posture Estimation

#### 2.1. Background Subtraction

The background subtraction, based on a colour model of brightness and chromaticity in RGB colour space\(^{(19)}\), is carried out to extract the silhouette image. By applying the following pixel classification procedure to the difference in brightness between the background model and the current image of frame number \(k\), at the pixel in the \(i\) row, the \(j\) column \(L_{ij}(k)\) and that of chromaticity \(S_{ij}(k)\), a pixel from the current image is classified into one of the four categories, such as foreground, background, shadow, and highlight.
if \( S_{ij}(k) > \alpha_{S_{ij}} \) or \( L_{ij}(k) < \alpha_{\text{low}} \)
then the pixel is foreground
else if \( L_{ij}(k) < \alpha_{L_{ij}} \max \) or \( L_{ij}(k) > \alpha_{L_{ij}} \min \)
then the pixel is background
else if \( L_{ij}(k) < 1 \)
then the pixel is shadow
else the pixel is highlight
endif
endif
endif

Here the parameters are defined by

\[
L_{ij}(k) = \frac{\mathbf{v}_{b_i} \cdot \mathbf{v}_{c_i}(k)}{|\mathbf{v}_{b_i}|^2}
\]  

(1)

\[
S_{ij}(k) = |\mathbf{v}_{c_i}(k)| \sqrt{1 - \left( \frac{\mathbf{v}_{b_i} \cdot \mathbf{v}_{c_i}(k)}{|\mathbf{v}_{b_i}| |\mathbf{v}_{c_i}(k)|} \right)^2}
\]  

(2)

where \( \mathbf{v}_{b_i} \) is the background model vector defined by using the means of the RGB values calculated over \( N \) background image sequences, and \( \mathbf{v}_{c_i}(k) \) is the current image vector defined by using the RGB value of the current image. The threshold values are defined by considering \( L_{ij} \) and \( S_{ij} \) of an image sequence acquired by a camera that observes the background as follows: \( \alpha_{L_{ij}} \max = \max(L_{ij}) \), \( \alpha_{L_{ij}} \min = \min(L_{ij}) \), and \( \alpha_{S_{ij}} = \max(S_{ij}) \). \( \alpha_{\text{low}} \) is defined so as to satisfy the following relationship: \( 0 < \alpha_{\text{low}} < \alpha_{L_{ij}} \min \).

2.2. Silhouette Volume Intersection

Silhouette volume intersection\(^{(21)}\)\(^{(22)}\)\(^{(23)}\) is the most popular method of reconstructing 3D object shapes from multi-camera images. The silhouette volume intersection is based on the silhouette constraint that a 3D object is encased in the 3D frustum produced by the back projection of a 2D object silhouette image, and the shape of a 3D object can be approximated by intersecting frusta of the object’s silhouette images obtained via multi-cameras. To achieve the silhouette volume intersection in our system, a space carving method (SCM) is utilized because of the low memory cost in implementation. The algorithm for SCM is as follows\(^{(10)}\).

for each \( v \) involved \{voxel\} begin
\[ v := \text{‘occupied’} \]
for each \( c \) involved \{camera\} begin
project \( v \) to image plane of \( c \)
if projected point isn’t in the silhouette then \( v := \text{‘empty’} \)
go to END_INNER_LOOP
endif
end
END_INNER_LOOP
end

Here, \{voxel\} is the set of all voxels, \{camera\} is the set of all cameras, ‘occupied’ is the value of an occupied voxel, and ‘empty’ is the value of an unoccupied voxel.

The back projection from a voxel \( \zeta^T = [X \ Y \ Z \ 1] \) in the XYZ-coordinate system onto the pixel \( \xi_q^T = [h^{(q)}X^{(q)} \ h^{(q)}Y^{(q)} \ h^{(q)}Z^{(q)}] \) of the image plane (xy-coordinate system) in the \( q \)th camera is defined by

\[
\xi_q = \mathcal{W}_q \zeta
\]  

(3)
where \( h(q) \) is a scale parameter and \( \Psi_q \) is the projection matrix of the \( q \)th camera. Since the projection matrix of each camera is determined in the calibration process, computational complexity can be reduced by maintaining the projection calculation result on a look-up table in advance.

2.3. 3D Skeleton Estimation

To estimate 3D skeleton information of human body posture, an articulated cylindrical human model as shown in Figure 2 is introduced. Here the significant points of the body, \( P_i = [X_i \ Y_i \ Z_i]^T (i = \text{Head, Neck, Shoulder1, Shoulder2, Elbow1, Elbow2, Hand1, Hand2, Waist1, Waist2, Knee1, Knee2, Toe1, and Toe2}) \), are indicated with circles. By fitting the cylindrical model to the reconstructed human body in 3D volume that is a set of voxels, the skeleton can be obtained as a set of cylinder axes as shown by the broken line in Figure 2.

2.3.1. Head and Hand Position Estimation

To begin the fitting process of the cylindrical human model, constraints that assume the initial conditions of the model are required. A Reeb graph approach(24) is sometimes useful to extract 3D volume features and the results can be utilized in the fitting of human model(25). However, the result obtained by the Reeb graph often depends on human body postures when it applies to the reconstructed human body of 3D volume and it is not suited to real-time processing since it requires great computational power(20). In this study, the positions of the head and hands are first located in order to use them as initial conditions in the fitting process of cylindrical human models. Skin color information is utilized to find the head and hands on the 3D reconstruction with ease.

**Skin color extraction**

The skin color model is obtained by using RGB colour space. First, the skin color pixel of the total number \( N_s \) is collected from each camera image in advance. Next, the skin color model vector \( v_s \) is defined by using the means of the RGB values calculated with the skin color pixels. The chromaticity \( S_{sij} \) between the input image \( u_{cij}(k) \) and \( v_s \) is defined by

\[
S_{sij}(k) = |u_{cij}(k)| \sqrt{1 - \frac{v_s \cdot v_u(k)}{|v_s||u_{cij}(k)|}}^2
\]

and the skin color area in the current image is then extracted by using the pixel classification rule: if \( S_{sij}(k) \) is less than the threshold \( \alpha_{S_s} \) and the pixel is inside of the silhouette, the pixel has skin colour. Here the threshold value is defined by considering \( S_{sij} \) with the collected skin colour pixels: \( \alpha_{S_s} = \max(S_{sij}) \).

**Detection of face and hands**

To estimate the position of face and hands, the back projection from the voxels of the reconstructed body onto the image plane of each camera is calculated. Here the constraint condition that the face and hands are always captured by \( n_0 \) cameras (\( n_0 \geq 2 \)) is assumed. A set of voxels is chosen as the candidate for face or hands if the pixels projected from the set of voxels are classified into the skin colour in \( n_c \) camera images (\( n_c \geq n_0 \)). The set of voxels that has the largest volume among the candidates is chosen.
as the face, and the others are designated as hands. The centre of gravity for the set of voxels in each candidate, \[ \{ X_i, Y_i, Z_i \vert i = \text{Face}, \text{Hand}_1, \text{Hand}_2 \}, \] is calculated by

\[
\begin{bmatrix}
X_{yi}
Y_{yi}
Z_{yi}
\end{bmatrix} = \frac{1}{M_i(0,0,0)} \begin{bmatrix}
M_i(1,0,0)
M_i(0,1,0)
M_i(0,0,1)
\end{bmatrix}
\]

(5)

where \( \mathbf{V}^{(i)}_{\text{XYZ}} \) is the set of voxels of the candidate \( i \), \( V_{\text{XYZ}} = 1 \) for volumes within the human body, and 0 for the background space. The position of face and hands, \( P_t \), is defined with the corresponding centre of gravity.

**Detection of the head** To estimate the head position, a sphere with a radius of \( r \) and the centre at the face position \( P_{\text{Face}} \) is assumed. Calculating the centre of gravity \( G_{\text{nonFace}} \) for the set of voxels that are included in the sphere but not included in the face area, the head position \( P_{\text{Head}} \) is located by shifting \( P_{\text{Face}} \) toward \( G_{\text{nonFace}} \) with the distance \( L_{\text{FtoH}} \). Since \( P_{\text{Head}} \) is the tentative position, it is recalculated after completing the torso position estimation.

2.3.2. Torso Position Estimation Assuming the initial position of the cylinder model for the torso based on the head position \( P_{\text{Head}} \), the fitting calculation of the torso cylinder \( C_{\text{Body}} \) is first carried out toward the centre of gravity \( G_{\text{All}} \) for the volumes of a human body. The next direction of fitting \( C_{\text{Body}} \) is defined by calculating the centre of gravity \( G_{\text{incBody}} \) for the set of voxels that are included in \( C_{\text{Body}} \). Next, \( C_{\text{Body}} \) is fitted toward \( G_{\text{incBody}} \), and \( G_{\text{incBody}} \) is then updated. The fitting calculation is quitted after \( G_{\text{incBody}} \) converges the constant voxel as follows.

\[
|G_{\text{incBody}}(t) - G_{\text{incBody}}(t-1)| < \epsilon_{\text{Body}}
\]

(7)

where \( t \) is the iteration number and \( \epsilon_{\text{Body}} \) is the threshold value.

**Detection of shoulders** In the converged torso cylinder \( C_{\text{Body}} \), the candidate voxels of shoulders are chosen from the set of voxels neigboured to the cross section that is located from the upper base of \( C_{\text{Body}} \), which closes to \( P_{\text{Head}} \), toward the lower base with the distance \( L_{\text{Shoulder}} \). Since it can be considered that the shape of cross section in a human torso is almost oval, the shoulder positions are located around the intersections between the length axis of the oval and its circumference. By calculating the distance between all candidate voxels: \( d_j = \sum \{ |P_i - P_j| \} \), the voxel that has the maximum distance is defined as one of the shoulder \( P_{\text{Shoulder}} \). The other shoulder \( P_{\text{Shoulder}} \) is defined with the voxel that locates in the symmetry position with respect to the centre of gravity \( G_{\text{Shoulder}} \) within the candidate voxels.

**Detection of waist** The candidate voxels for the waist are chosen from the set of voxels neigboured to the cross section that is located from the lower base of \( C_{\text{Body}} \) toward the upper base with the distance \( L_{\text{Waist}} \). By calculating the distance between all candidate voxels: \( d_j = \sum \{ |P_i - P_j| \} \), the voxel that has the maximum distance is defined as one of the waist \( P_{\text{Waist}} \). The other waist \( P_{\text{Waist}} \) is defined with the voxel that locates in the symmetry position with respect to the centre of gravity \( G_{\text{Waist}} \) within the candidate voxels. Hereafter, \( G_{\text{Waist}} \) is called the center of waist \( P_{\text{Waist}} \).

2.3.3. Head Position Recalculation Assuming the initial position of the cylindrical model for the head based on the torso cylinder \( C_{\text{Body}} \), the head position \( P_{\text{Head}} \) is recalculated. The neck position \( P_{\text{Neck}} \) is defined by the centre of the upper base of \( C_{\text{Body}} \). First, the fitting calculation of the head cylinder \( C_{\text{Head}} \) is carried out from \( P_{\text{Neck}} \) toward the parallel direction with the axis of \( C_{\text{Body}} \). In order to define the next direction of fitting \( C_{\text{Head}} \), the centre of gravity \( G_{\text{incHead}} \) for the set of voxels that are included in \( C_{\text{Head}} \) is calculated. Next, \( C_{\text{Head}} \) is fitted toward \( G_{\text{incHead}} \), and \( G_{\text{incHead}} \) is then updated. The fitting calculation is quitted after \( G_{\text{incHead}} \) converges the constant voxel as follows.

\[
|G_{\text{incHead}}(t) - G_{\text{incHead}}(t-1)| < \epsilon_{\text{Head}}
\]

(8)
where $\epsilon_{\text{head}}$ is the threshold value. As a result, $P_{\text{head}}$ is defined by the centre of the upper base of $C_{\text{head}}$.

### 2.3.4. Arm Position Estimation

By fitting the cylindrical model of upper and lower arms, $C_{\text{upperArm}}_i$ and $C_{\text{lowerArm}}_i$ ($i = 1, 2$), the elbow position $P_{\text{Elbow}}$ can be estimated. Here $C_{\text{upperArm}}_i$ and $C_{\text{lowerArm}}_i$ are connected to $P_{\text{Shoulder}}_i$ and $P_{\text{Hand}}_i$, respectively. It is considered that the elbow locates at the distance $L_{\text{upperArm}}$ from the shoulder and the distance $L_{\text{lowerArm}}$ from the hand. However, the connection relationships between shoulder positions $P_{\text{Shoulder}}_i$ and hand positions $P_{\text{Hand}}_i$ have not been defined since the shoulder and hand positions are not classified into either left nor right. Therefore, the candidate voxels of elbows are assumed by considering the combination of $P_{\text{Shoulder}}_i$ and $P_{\text{Hand}}_i$ with the distances of $L_{\text{upperArm}}$ and $L_{\text{lowerArm}}$: \{ $P_{\text{Shoulder}}_1$, $P_{\text{Hand}}_1$ \}, \{ $P_{\text{Shoulder}}_2$, $P_{\text{Hand}}_2$ \}, \{ $P_{\text{Shoulder}}_1$, $P_{\text{Hand}}_2$ \}, and \{ $P_{\text{Shoulder}}_2$, $P_{\text{Hand}}_1$ \}. The fitting calculations of $C_{\text{upperArm}}_i$ and $C_{\text{lowerArm}}_i$ are carried out from the elbow candidate voxels for all combinations to find the maximum value of the fitting rate. The fitting rate is defined by

$$R_{\text{Arm}ij} = \sum_{V \in \{C_{\text{upperArm}}(\theta) \cap C_{\text{lowerArm}}(\theta)\}} V_{XYZ}$$

As a result, the elbow position $P_{\text{Elbow}}$ is defined with the voxel that has the maximum fitting rate and is included in $C_{\text{upperArm}}_i$.

### 2.3.5. Leg Position Estimation

#### Detection of knee

By fitting the cylindrical model of upper leg $C_{\text{upperLeg}}_i$ ($i = 1, 2$) from the waist position $P_{\text{Waist}}$, respectively, the knee position $P_{\text{Knee}}$ can be estimated. First, the fitting calculations of $C_{\text{upperLeg}}_i$ are carried out for all directions with the interval of angle $\theta$ to find the maximum value of the fitting rate. The fitting rate is defined by

$$R_{\text{upperLeg}ij} = \sum_{V \in \{C_{\text{upperLeg}}(\theta) \cap C_{\text{upperLeg}}(\theta)\}} V_{XYZ}$$

Next, the combination of $C_{\text{upperLeg}}_1$ and $C_{\text{upperLeg}}_2$ that has the maximum fitting rate is chosen. The knee position $P_{\text{Knee}}$ is defined with the voxel that locates near the centre in the base of $C_{\text{upperLeg}}$, which is not close to $P_{\text{Waist}}$.

#### Detection of toe

By fitting the cylindrical model of lower leg $C_{\text{lowerLeg}}_i$ ($i = 1, 2$) from the knee position $P_{\text{Knee}}$, respectively, the toe position $P_{\text{Toe}}$ can be estimated. The fitting calculations of $C_{\text{lowerLeg}}_i$ are first carried out for all directions with the interval of angle $\phi$ to find the maximum value of the fitting rate. The fitting rate is defined by

$$R_{\text{lowerLeg}ij} = \sum_{V \in \{C_{\text{lowerLeg}}(\theta) \cap C_{\text{lowerLeg}}(\theta)\}} V_{XYZ}$$

Then, the combination of $C_{\text{lowerLeg}}_1$ and $C_{\text{lowerLeg}}_2$ that has the maximum fitting rate is chosen. The toe position $P_{\text{Toe}}$ is defined with the voxel that locates near the centre in the base of $C_{\text{lowerLeg}}$, which is not close to $P_{\text{Knee}}$.

### 2.4. Tracking of Significant Points

To optimize and track the positions of the significant point, the following state space model\(^{[26]}\) is assumed for every significant point.

\[
x_{m}(t + 1) = \Phi x_{m}(t) + \Gamma u_{t}(t) \\
y_{m}(t) = H x_{m}(t) + w_{t}(t)
\]

\[
\begin{align*}
x_{m}(t) = \begin{bmatrix} X_{i}(t) & X_{i}(t-1) & Y_{i}(t) & Y_{i}(t-1) & Z_{i}(t) & Z_{i}(t-1) \end{bmatrix}^{T}, \\
y_{m}(t) = \begin{bmatrix} X_{i}(t) & Y_{i}(t) & Z_{i}(t) \end{bmatrix}^{T}, \\
u_{t}(t) = \begin{bmatrix} u_{x}(t) & u_{y}(t) & u_{z}(t) \end{bmatrix}^{T}
\end{align*}
\]

where $x_{m}(t)$ represents the state at time $t$, $\Phi$ is the state transition matrix, $\Gamma$ is the input matrix, $u_{t}(t)$ is the input vector at time $t$, $y_{m}(t)$ is the output vector at time $t$, and $w_{t}(t)$ is the measurement vector at time $t$.
Here, $x_{mi}(t)$ is the state vector, $y_{mi}(t)$ is the output vector, $t$ is the frame number, and $i$ denotes the significant point ($i = \text{Head, Neck, Shoulder1, Shoulder2, Elbow1, Elbow2, Hand1, Hand2, Waist, Waist1, Waist2, Knee1, Knee2, Toe1, and Toe2}$). The vector $u_i(t)$ is the system noise whose distribution is Gaussian distribution $N(0, \tau_i^2)$, and $w_i(t)$ is the observation noise whose distribution is $N(0, \tau_w^2)$. This state space model assumes that the variation of significant points’ velocity is smooth. By using Monte Carlo filter\(^{(27)}\), the state vector $x_{mi}(t)$ is estimated.

### 2.5. 3D Skeleton Estimation

Each cylindrical model has connection information relating to other cylinder models. According to the connection information, the significant point positions $P_i$ are connected with sticks to obtain a 3D skeleton of the human body.

### 3. Human Posture Estimation Experiment

#### 3.1. Experimental Setup

To evaluate the efficiency of our human body posture estimation method, 3D human body reconstruction and 3D skeleton estimation experiments were carried out. The multi-camera system shown in Figure 3 consists of a server and clients. The silhouette image is extracted in each client computer connected to each camera and is then transferred into the server computer that calculates the 3D reconstruction via voxel data and estimates the significant points of the body. The server-client system is achieved by using socket communication with TCP/IP, local network of 1000Base-T, and winsock2 programming.

- **Server**: DELL Precision Workstation 650, Intel(R) Xeon CPU 3.2GHz, 3.50GB RAM, Windows XP SP2.

![Fig. 3 Experimental setup of multi-camera system.](image)
The proposed method was coded in C++ language (Microsoft Visual Studio .NET) with DirectX graphic system and was implemented on personal computers. The images from the CCD cameras (IEEE 1394 camera “Dragonfly2”, Point Grey Research Inc.) were digitized into the client computers with a 640 by 480 pixel resolution via IEEE 1394 Interface in real time (frame rate of 60 Hz). While many cameras are required to achieve the 3D reconstruction with silhouette volume intersection precisely, six cameras, which are placed hexagonally, are utilized in our system. Three cameras are located near the ceiling and the other three cameras are set at middle height. The measured space in front of the cameras is $2 \times 2 \times 2$ m.

The camera calibration was carried out by using Microsoft easy camera calibration tool(28). In the background subtraction processing, thread-programming technique with a dual CPU system is used: the image is divided into two parts, and then the processing is carried out on each part in parallel by thread programming. Figure 4 shows an example of a background subtraction result from the six cameras. In this experiment, 100 image sequences were utilized to calculate the threshold values.

3.2. Experimental Results

In the experiment, the motion was a human walking in place. The cylindrical human model was chosen as shown in Table 1 and the parameters were $L_{\text{Foot}} = 3$, $L_{\text{Shoulder}} = 4$, and $L_{\text{Waist}} = 1$. In each significant point, the number of particle for Monte Carlo filter was 2000 and the initial condition of the state vector was gaussian distribution, which has the mean of $P_i(1)$ and whose covariance matrix is unit matrix. The model parameter were $\tau_2^u = 2$ and $\tau_0^2 = 0.14$. These parameters were defined by trial and error.

Figure 5 (top to bottom) shows examples of the 3D reconstruction of body posture with $50 \times 50 \times 50$ voxel resolution, the fitting result of cylindrical human model, the 3D skeleton with the estimated significant points, and the 3D representation of body posture with a CG human model. In the 3D reconstructions, the voxel’s colours are calculated in average from the camera images. In the fitting results, the voxel included in one of cylinders is coloured with one of six colours that correspond to the cylinders, while the voxel coloured with black is not classified into any cylinder. In the skeleton images, the estimated significant points are indicated with small squares. As shown in Figure 5, the significant points of a body and 3D

<table>
<thead>
<tr>
<th>Model</th>
<th>Radius [voxel]</th>
<th>Height [voxels]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Torso</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>Upper and Lower arm</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>Upper and Lower leg</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>
Fig. 5 Example of an image sequence on estimating human body posture (top to bottom: Frame number, 3D reconstruction with voxel, fitting of cylindrical human model, human body skeleton and significant points, and representation of human body posture with CG model).

Table 2 Computational cost.

<table>
<thead>
<tr>
<th>Processing</th>
<th>Resolution</th>
<th>Frame rate [frames/sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background subtraction</td>
<td>640 × 480 (pixel)</td>
<td>40</td>
</tr>
<tr>
<td>3D reconstruction</td>
<td>50 × 50 × 50 (voxel)</td>
<td>38</td>
</tr>
<tr>
<td>100 × 100 × 100 (voxel)</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>200 × 200 × 200 (voxel)</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3D skeleton estimation</td>
<td>θ = φ = π/4 (rad)</td>
<td>9</td>
</tr>
<tr>
<td>(50 × 50 × 50 voxel resolution)</td>
<td>θ = φ = π/12 (rad)</td>
<td>3.8</td>
</tr>
<tr>
<td></td>
<td>θ = φ = π/20 (rad)</td>
<td>1.7</td>
</tr>
</tbody>
</table>

skeleton can be extracted successfully from the 3D voxel reconstruction in every frame. Table 2 shows the computational costs. The 3D human body posture reconstruction of 200 × 200 × 200 voxel resolution can be achieved with 20 frames/sec in average, however, it is hard to
Table 3  Processing speed of fitting cylindrical model with 50×50×50 voxel resolution.

<table>
<thead>
<tr>
<th>Processing</th>
<th>Time [msec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head and hand position estimation</td>
<td>6.88</td>
</tr>
<tr>
<td>Torso position estimation</td>
<td>27.10</td>
</tr>
<tr>
<td>Head position recalculation</td>
<td>2.89</td>
</tr>
<tr>
<td>Arm position estimation</td>
<td>40.84</td>
</tr>
<tr>
<td>Leg position estimation</td>
<td></td>
</tr>
<tr>
<td>$\theta = \phi = \pi/4$ (rad)</td>
<td>29.49</td>
</tr>
<tr>
<td>$\theta = \phi = \pi/12$ (rad)</td>
<td>182.40</td>
</tr>
<tr>
<td>$\theta = \phi = \pi/20$ (rad)</td>
<td>494.05</td>
</tr>
</tbody>
</table>

![Fig. 6 Estimation error of significant point.](image)

We can only achieve the processing of 9 frames/sec with 50×50×50 voxel resolution when both $\theta$ and $\phi$ equal $\pi/4$. In the bottom of Figure 5, motion control of the CG human model is shown as an application example of posture estimation. Here the CG human model is designed by POSER 6 for Windows (e frontier). The joint angles of the CG human model are calculated by using the estimated 3D coordinates of the body’s significant points to control the posture of the CG model. To achieve practical use of the estimation system, however, real-time processing of 30 frames/sec with 200×200×200 voxel resolution would be required at least.

To confirm the effectiveness of the proposed system, we conducted the following experi-

achieve extraction of significant points in real-time with such a voxel resolution in our current system. Table 3 shows the averaged processing speed in the fitting the cylindrical model with 50×50×50 voxel resolution. Here the measurement of processing speed was repeated 15 times. We can only achieve the processing of 9 frames/sec with 50×50×50 voxel resolution when both $\theta$ and $\phi$ equal $\pi/4$. In the bottom of Figure 5, motion control of the CG human model is shown as an application example of posture estimation. Here the CG human model is designed by POSER 6 for Windows (e frontier). The joint angles of the CG human model are calculated by using the estimated 3D coordinates of the body’s significant points to control the posture of the CG model. To achieve practical use of the estimation system, however, real-time processing of 30 frames/sec with 200×200×200 voxel resolution would be required at least.

To confirm the effectiveness of the proposed system, we conducted the following experi-

achieve extraction of significant points in real-time with such a voxel resolution in our current system. Table 3 shows the averaged processing speed in the fitting the cylindrical model with 50×50×50 voxel resolution. Here the measurement of processing speed was repeated 15 times. We can only achieve the processing of 9 frames/sec with 50×50×50 voxel resolution when both $\theta$ and $\phi$ equal $\pi/4$. In the bottom of Figure 5, motion control of the CG human model is shown as an application example of posture estimation. Here the CG human model is designed by POSER 6 for Windows (e frontier). The joint angles of the CG human model are calculated by using the estimated 3D coordinates of the body’s significant points to control the posture of the CG model. To achieve practical use of the estimation system, however, real-time processing of 30 frames/sec with 200×200×200 voxel resolution would be required at least.

To confirm the effectiveness of the proposed system, we conducted the following experi-
First, the real positions of significant points $P_{\text{real}}$ at each frame were obtained manually by one subject from the 3D reconstruction, and the estimation error $e_i(t) = |P_{\text{real}}(t) - P_i(t)|$ was then evaluated. Here the estimation error should be considered as a relative value since the real positions include some errors caused by the manual extraction. Figure 6 shows the estimation error in each significant point for the sequence shown in Figure 5. The mean of error with all significant points for the sequence $\mu_e$ is 2.1 voxels and its variance $\sigma^2_e$ is 0.2, while $\mu_e$ is 2.2 voxels and $\sigma^2_e$ is 0.3 without tracking the significant points. Although the improvement in this experimental example is very small, this result shows that the tracking of significant points has a possibility to improve the human body posture estimation.

4. Conclusions

This paper investigated a human body posture estimation method based on back projection of human silhouette images extracted from multi-camera images. The multi-camera system is based on a server-client system with local network of 1000Base-T to achieve voxel reconstruction of 3D human body posture in real-time. To extract significant points of the body in 3D space, the articulated cylindrical human model was applied to the voxel reconstruction of the body and the 3D skeleton was estimated. To evaluate our proposed estimation method, 3D reconstruction experiments of human body posture and extraction experiments of human body’s significant points were carried out. The experimental system with six CCD cameras and seven personal computers was constructed and the system ran in approximately real time (9 frames/sec with 50×50×50 voxel resolution). The experimental results confirmed both the feasibility and effectiveness of the proposed system.

There is still more work to do: increasing the number of cameras is necessary to improve the accuracy of 3D reconstruction, optimizing the implementation of estimation algorithms should also be investigated to improve the processing speed. Application to gesture recognition with the proposed method is another one of our important future tasks.

Acknowledgments

This study was supported in part by the Academic Frontier Research Project on “New Frontier of Biomedical Engineering Research” of Ministry of Education, Culture, Sports, Science and Technology.

References


