Appl Intell (2011) 35:163–177
DOI 10.1007/s10489-009-0209-4

Estimation of 3-D human body posture via co-registration of 3-D human model and sequential stereo information

Nguyen Duc Thang · Tae-Seong Kim · Young-Koo Lee · Sungyoung Lee

Published online: 20 February 2010

Abstract In this paper, we present a technique for estimating three-dimensional (3-D) human body posture from a set of sequential stereo images. We estimated the pixel displacements of stereo image pairs to reconstruct 3-D information. We modeled the human body with a set of ellipsoids connected by kinematic chains and parameterized with rotational angles at each body joint. To estimate human posture from the 3-D data, we developed a new algorithm based on expectation maximization (EM) with two-step iterations, assigning the 3-D data to different body parts and refining the kinematic parameters to fit the 3-D model to the data. The algorithm is iterated until it converges on the correct posture. Experimental results with synthetic and real data demonstrate that our method is capable of reconstructing 3-D human posture from stereo images. Our method is robust and generic; any useful information for locating the body parts can be integrated into our framework to improve the outcomes.

Keywords Estimation of 3-D human body posture · Stereo images · Articulated human body model

N.D. Thang · Y.-K. Lee · S. Lee
Department of Computer Engineering, Kyung Hee University, Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do, 446-701, South Korea

N.D. Thang
e-mail: ducthang@oslab.khu.ac.kr

T.-S. Kim (✉)
Department of Biomedical Engineering, Kyung Hee University, Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do, 446-701, South Korea
e-mail: tskim@khu.ac.kr

1 Introduction

During the last decade, automatically tracking and recovering thee-dimensional (3-D) human body posture from video footage has become one of the most interesting topics in the field of computer vision, with applications in tracking, motion analysis, and activity recognition. In healthcare applications, tracking the movements of individuals may allow clinicians and family members to detect events such as dangerous falls by elderly family members, or monitor the activities of patients for diagnosis of disease.

The traditional system used for human motion capture consists of multiple special cameras that are arranged in a studio. The subject must wear markers that can be detected by the multiple cameras in 3-D. Combining the 3-D coordinates of the recognized points, the system computes the necessary joint parameters to represent the human body. Requiring expensive devices with complicated deployment, this approach is only suitable for limited applications.

Markerless systems that estimate posture from a sequence of images are receiving more attention because they are more convenient for users in daily applications. Although using a single camera is the most convenient way of capturing data, monocular images do not provide enough information to recover 3-D posture due to ambiguity and occlusion: basically, when projecting a 3-D object into a 2-D image, we lose the depth information from the original object.

Obviously, multiple-camera systems, with cameras installed at different view angles, can capture comprehensive 3-D information. However, some disadvantages are also evident with the multiple-camera approach. The cameras need to be fixed at multiple positions for calibration and are inflexible for deployment in outdoor and mobile environments. Synchronization is also required for the high-speed hardware and the additional protocols necessary to control large video files over a network. It is possible to obtain useful information, including depth data, using a stereo camera, which consists of two lenses integrated into a unified device. Stereo cameras achieve depth perception in a manner similar to human eyesight. Due to their flexibility, stereo cameras are convenient to deploy under a variety of environmental conditions.

Here, we give an overview of previous studies that have investigated human pose estimation from video. Generally, these algorithms can be grouped into two categories, namely the *model-based* approach and the *matching-based* approach.

The model-based approach The *model-based* approach predefines the human body with a set of parameters related to the locations of body joints, the kinematic rotational angles, and the sizes of body parts, then attempts to fit the model to the observation.

One branch of methods using this approach uses only monocular images. Because of the ambiguity in depth information, most researchers simplify the problem by only locating the positions of body parts in 2-D images [10, 16, 19–21, 31]. Among these studies, some sophisticated probabilistic graphical models were designed to evaluate the compatibility between model parameters and cues from images like colors, edges, shapes, and contours. The most suitable parameters are found by message-passing based algorithms such as nonparametric belief propagation [23]. Although the problem is challenging, there are some current attempts to estimate 3-D human posture information using only monocular images [12, 26]. Similarly, in [12], the probabilistic approach is also attempted where the posterior probability distribution of the parameters is estimated by the Markov chain Monte Carlo (MCMC) method. The convergence speed of MCMC can be ensured by decomposing the Markov chain to a series of local transitions of each portion (e.g., face or limb).

Given the limitations of using a single view, the multiview approach is proposed to overcome the missing 3-D information. Integrating the 2-D cues from each image with the data from multiple cameras, Gupta et al. [7] demonstrated that their system can solve the problem of posture estimation even within self occlusion. In [11], Knossow et al. analyzed the properties of the extremal contours of elliptical cones, then analytically derived the non-linear expressions of contour velocities that can be further used to minimize the differences between model contours and contours extracted from binary image silhouettes. The shortcomings of these methods are shown by the fact that they work separately on a single image. The outcomes also need to be combined in an additional stage to obtain the precise 3-D model parameters.

Alternatively, rather than directly processing on images, some algorithms assume that the 3-D data is already available. In [24], Sundaresan et al. segmented 3-D voxels into different rigid parts, represented each with a 1-D curve, and linked these curves to form the articulated skeleton. Meanwhile, several methods rely on the gradient descent algorithm to identify the human kinematic configurations best matching observations. In [18], Plankers et al. introduced the concept of an articulated soft object to represent the human body as an explicit surface. The first and second derivatives of the body surface are used by the least-square estimator to adjust the soft object matching with the cloud of 3-D points. During model registration, to reduce ambiguity, each 3-D point is cast into one portion (i.e., a single body part) of the model using the Euclidean distance as in [14, 29]. Recently, Horaund et al. [9] applied the *datum* distance calculated from the coordinates and normal vectors of 3-D points to perform point assignments. Obviously, the use of distance alone to locate the body parts is deficient in the case where the model is diverging markedly from the data. It is necessary to exploit more cues to improve detection accuracy.

The matching-based approach The *matching-based* approach stores a number of human pose configurations in a database. From the exemplar dataset, the main features of each observation are extracted and mapped to a specific configuration. This method also determines the features from the queries and uses them to search for the most suitable poses. For 2-D images, the binary shapes including the internal and external contours can be utilized as the descriptors for each 3-D pose [1, 15]. Additionally, after background subtraction, the remaining silhouettes of the human body also exhibit potential features [22]. For stereo data, in [32], Yang et al. used the depth images (i.e., disparity images) to retrieve a human pose from the hierarchy database with 100,000 exemplars. Within the huge space of all possible human poses, the *matching-based* approach requires efficient methods for organizing and searching the exemplars stored in the database.

In this study, we capitalize on the advantages of the stereo camera to implement a novel system for recovering 3-D human body postures. Our system is built on an iterative coregistration algorithm with two steps: detecting body parts from 3-D data and fitting the model to the data after the body parts have been located. Here are the main characteristics of our algorithm that distinguish it from others: (1) As a *model-based* approach for estimating 3-D human body postures from stereo camera images, our method does not require us to build and maintain a large database of human poses. (2) Both the bottom-up approach (i.e., detecting the location of each body part) and the top-down approach (i.e., connecting different body parts in a predefined human model using

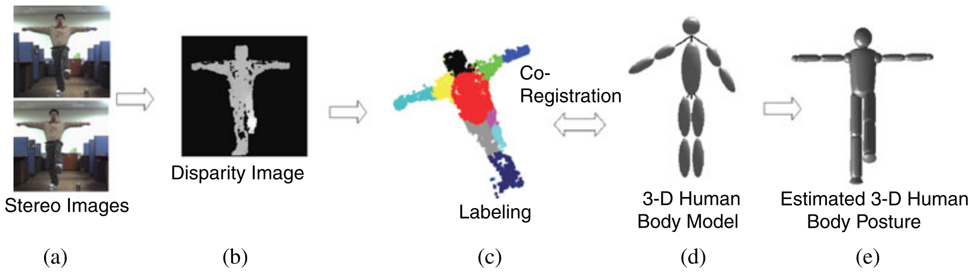


Fig. 1 Our proposed method of estimating 3-D human body posture from stereo images. (**a**) A set of stereo images. (**b**) Estimated disparity image. (**c**) Labeling the body parts of the 3-D data. (**d**) Fitting the 3-D model with the 3-D data. (**e**) Final estimated body posture

ellipsoids) are integrated in our framework. (3) Our algorithm utilizes all 3-D information, cues from RGB images, and geodesic relationships among 3-D points to better detect body parts.

The rest of this paper is organized as follows. In Sect. 2, we describe our overall system. The main algorithm of recovering human posture from 3-D data is presented in Sect. 3. The experimental results are shown in Sect. 4. Finally, we present our conclusion and discussion in Sect. 5.

2 Our methodology

The step-by-step processing stage of our system is briefly described in Fig. 1. In the preprocessing step, we estimate the disparity between the left and right images taken by a stereo camera. The 3-D location of the observed subject is reconstructed using this disparity and represented by a cloud of points in 3-D. To fit the 3-D model to the given 3-D data, we perform co-registration in two steps: labeling (E-step) and model fitting (M-step). The labeling step assigns each point to one ellipsoid and the model fitting step fits the ellipsoids to their corresponding points. This process is iterated by minimizing the discrepancies between the model and the observation, finally recovering the correct human pose. The details of our co-registration algorithm are discussed in Sect. 3.

2.1 Stereo image processing

The computation of stereo information is the preliminary processing step necessary to recover 3-D information from a pair of images. The displacements between two images are presented as a disparity image and used to estimate the depth value. With an ordinary searching technique, it exhausts $O(n^3)$ computation to obtain the complete disparity values, assuming that the size of the image is n^2. We use the fast stereo matching algorithm, Growing Correspondence Seeds (GCS) [4], which requires only a small fraction of the disparity space to improve speed and accuracy. The computation complexity becomes $O(kn^2)$ with $k \ll n$ compared with searching the entire disparity space at $O(n^3)$. Moreover, if the background is partially eliminated, we can reduce the searching time on the sparse regions. The approach we apply for the background modeling and removal is described in [30].

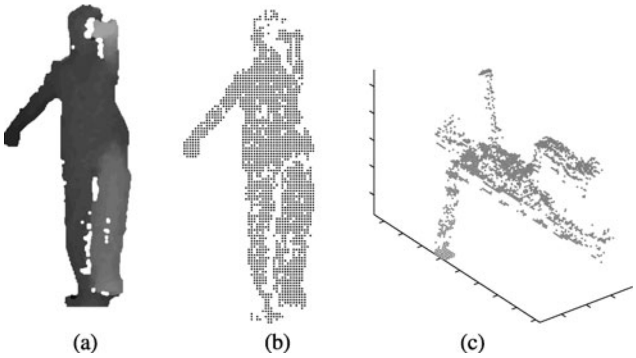


Fig. 2 Computing the 3-D stereo data. (**a**) Disparity image. (**b**) Sampling on the grid. (**c**) 3-D data

Then, the disparity image is sampled by a grid to reduce the number of points in the observed data and avoid extensive computation, as depicted in Fig. 2(b). To obtain the 3-D data, the depth value Z of each point is computed by

$$Z = \frac{fb}{d} \tag{1}$$

where f is the focus length, b is the base-line, and d is the disparity value. The two remaining coordinates X and Y are given by

$$X = \frac{uZ}{f}, \qquad Y = \frac{vZ}{f} \tag{2}$$

where u and v are the column and row index of a pixel in the disparity image.

2.2 3-D human body model

The 3-D human model includes a set of blobs represented by a set of ellipsoids. In the computation of transformation, we formulate the equation of an ellipsoid in the 4-D projective space as

$$q(X) = X^T \mathbf{Q}_\theta^T \mathbf{S}^T \mathbf{D}\mathbf{S}\mathbf{Q}_\theta X - 2 = 0 \tag{3}$$

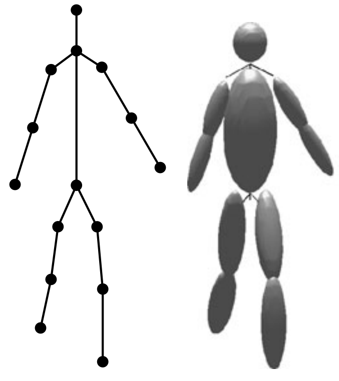


Fig. 3 3-D human body model. (**a**) Skeleton model. (**b**) Computation model with ellipsoids. (**c**) Human synthetic model with super-quadrics

where $\mathbf{D} = \text{diag}[a^{-2}, b^{-2}, c^{-2}, 1]$ configures the size of the ellipsoid, $\mathbf{S}$ locates the center of the ellipsoid in the local coordinate system, $\mathbf{Q}_\theta$ is the skeleton-induced transformation, and $X = [x, y, z, 1]^T$ is the coordinate of a 3-D point. We choose $b = a$ and $c \geq a$ to simplify the Euclidean distance computation from one point to an ellipsoid. The 4×4 transformation matrix $\mathbf{Q}_\theta$ is a matrix function of $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ where $\theta_1, \theta_2, \dots, \theta_n$ are the n kinematic parameters that control the position of each ellipsoid in the model. $\mathbf{Q}_\theta$ is not only a single transformation, but it relates to a kinematic chain of transformations through each body part. The joint between two adjacent parts has up to three rotational degrees of freedom (DOF), while the transformation from the global coordinate system to the local coordinate system at the human hip requires six DOF (i.e., three rotations and three translations). We separate $\mathbf{Q}_\theta$ to a series of independent primitives that only depend on a single parameter,

$$\mathbf{Q}_\theta = \mathbf{Q}_n(\theta_n)\mathbf{Q}_{n-1}(\theta_{n-1}) \dots \mathbf{Q}_1(\theta_1) \tag{4}$$

where $\mathbf{Q}_1(\theta_1), \mathbf{Q}_2(\theta_2), \dots, \mathbf{Q}_6(\theta_6)$ are of six DOF of the global transformation and $\mathbf{Q}_i(\theta_i) = \mathbf{Tr}_i\mathbf{R}(\theta_i)$ with $i > 6$ is the local transformation from one coordinate system i to the other $i + 1$. $\mathbf{Tr}_i$ is the translation matrix determined by the skeleton architecture and $\mathbf{R}(\theta_i)$ is the rotation matrix around the x-,y-, or z-axis. We can set $\mathbf{Tr}_i$ to be the identity matrix $\mathbf{I}_{4\times4}$ if we want to add more than one DOF to a joint.

The whole body configuration is depicted in Fig. 3. There are 14 segments of the body, nine joints (two knees, two hips, two elbows, two shoulders, and one neck), and 24 DOF (two DOF at each joint and six free transformations from the global coordinate system to the local coordinate system at the hip). Each body part may contain several ellipsoids. However, to simplify the computation, we use only one for each.

For better display and to create a synthetic human model for simulations, we also designed a model using super-quadrics as shown in Fig. 3(c). The equation of the super-quadric surface [24] without any transformation is expressed as

$$\left(\frac{x}{a_0}\right)^2 + \left(\frac{y}{b_0}\right)^2 = \left(1 + \frac{sz}{c_0}\right)\left(1 - \left(1 - \frac{2z}{c_0}^d\right)\right), \quad 0 \leq z \leq c_0 \tag{5}$$

where a_0, b_0, and c_0 determine the size of the super-quadric along the x-axis, y-axis, and z-axis, respectively.

2.3 Distance from one point to an ellipsoid

The distances between a set of points to an ellipsoid are used to measure the differences between the 3-D data and the model. For simplification, the function $q(X)$ defined in (3), which approaches zero at the ellipsoid surface and becomes larger when the point moves away from the ellipsoid, has been defined as the *algebraic distance* [18]. However, due to variation that is related to direction (e.g., with the prolate spheroid, the algebraic distance gets smaller as the point moves toward the poles), the algebraic distance cannot exactly reflect the measurement, especially for thin ellipsoids (usually representing limbs). In addition, Horaud et al. [9] proposed an alternative distance, the *datum distance*; however, as it requires normal vectors, it is very difficult to calculate this distance from the data gathered by a stereo camera alone.

The *Euclidean distance*, equal to the distance from one point to its nearest point in the ellipsoid surface, is rarely used because it requires solving a sixth-degree polynomial equation [8]. In this work, with the symmetric ellipsoid model, the calculation of Euclidean distance can be simplified: First of all, rather than computing Euclidean distance in the global coordinate system (x, y, z), the point $X_0(x_0, y_0, z_0)$ can be transformed to the local coordinate system (x', y', z') that holds the ellipsoid. In Fig. 4, let P be the plane that contains a point X_0 and the major z'-axis of the ellipsoid. The intersection between the plane P and the ellipsoid will be an ellipse. The computation of the Euclidean distance to an ellipsoid is reduced to find the distance between a point X_0 and an ellipse lying in P with only a fourth-degree polynomial equation that has an analytical solution enabling us to calculate its roots.

Moreover, the kinematic parameter $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ in (3) is updated by the gradient descent method in Sect. 3.2.2. Therefore, at each step, the point X_0 moves to $X_0 + dX_0$ with a small change dX_0 in the local coordinate system (x', y', z'). Corresponding, X_t, the nearest point of X_0 in the ellipsoid surface, also moves to $X_t + dX_t$, which can be calculated from X_0, dX_0, and X_t with some multiplication and addition.

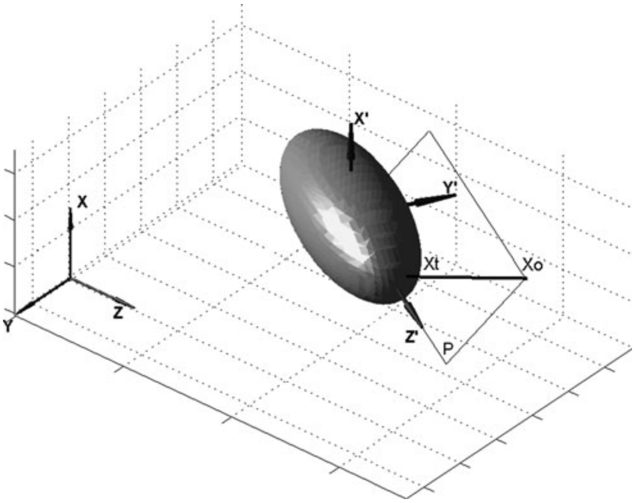


Fig. 4 The Euclidean distance from a point to an ellipsoid

The mathematical details of finding the nearest point in an ellipsoid surface to a given point are described in Appendix A.

3 Estimating 3-D human body posture from 3-D stereo data

This section presents our algorithm to estimate 3-D human body posture from the 3-D stereo data. First, we establish a comprehensive conditional probabilistic distribution between the human pose specified by the kinematic parameter $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ and the given 3-D data and RGB image. Then, we show how to estimate the optimal kinematic parameter θ^* that maximizes the distribution by the EM algorithm. The estimated parameter θ^* will correspond to the most suitable human posture with the given information.

3.1 Probabilistic relationship between the model parameters and the stereo data

We use $D = (X_1, X_2, \ldots, X_M)$ to denote M points of the 3-D data and I for the RGB image. Since our model is created with multiple ellipsoids, the supplementary variables are introduced to determine to which part of the body (i.e., ellipsoid) each point should belong. Let $V = (v_1, v_2, \ldots, v_M)$ denote the body part assignments or labels of each point. The posterior probability of the label V and the model parameter θ given the 3-D data and RGB image is expressed by

$$P(V, \theta | I, D) \propto P(V)P(I|V)P(D|V)P(D|V, \theta). \quad (6)$$

The elements of (6) are sequentially defined in the following sections.

3.1.1 Smoothness energy

The smoothness prior $P(V)$ is derived in the form of the Potts model [3],

$$P(V) = \prod_{i=1}^{M} \prod_{j \subset \mathcal{N}_i} P(v_i, v_j) \quad (7)$$

where $\mathcal{N}_i$ is a set of neighbors of point i and $P(v_i, v_j)$ is,

$$P(v_i, v_j) = \begin{cases} e^{\gamma} & \text{if } v_i = v_j, \\ 1 & \text{if } v_i \neq v_j \end{cases} \quad (8)$$

where γ (in our case $\gamma = 0.5$) is a real positive constant. $P(v_i, v_j)$ is used to drive the label of each point toward the same label of its neighbors. This causes the labeling results to become smooth and eliminates the outliers. The simplest way to locate the neighbors bounded by the radius d of one point is via a mask. We predefine the binary mask based on the distance d and perform an operation via the AND operator with the binary silhouette to find the neighbors of each point. We set $d = 2$ for all of our experiments.

3.1.2 Image likelihood

Some partial regions in the RGB image can provide extra information to identify the body components. Generally, the image likelihood term is derived as

$$P(I|V) = \prod_{i=1}^{M} \phi(I|v_i). \quad (9)$$

One might utilize the shape of the binary silhouettes or texture information to detect body parts. In our approach, we apply face detection to locate the head. Potential face areas are ascertained by detecting skin in the HSV color space and thermal infrared domains [5]. Some regions lying outside the binary silhouette or having unsuitable shapes (too small or appearing to be limbs) are considered outliers and removed. Estimation of the binary silhouette that relies only on background subtraction is not enough to obtain the correct result due to the effects of lighting conditions and shadows. As shown in Fig. 5, after the stereo computation, based

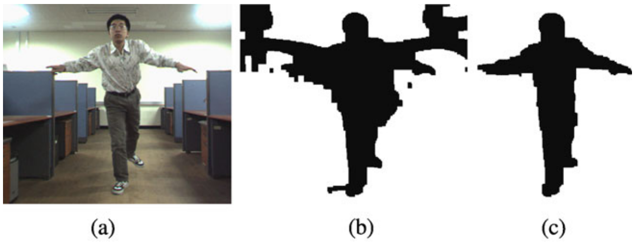
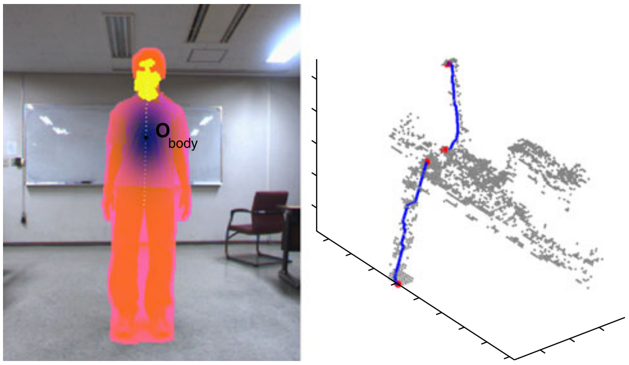


Fig. 5 Binary silhouette extraction. (**a**) Input image. (**b**) Background substraction. (**c**) Refined silhouette



(a) (b)

Fig. 6 Illustration of the factors that affect label assignments. (**a**) Image likelihood for detecting the face and torso. (**b**) Geodesic distance preserved with human movements

on the estimated distance between the person and the camera, some pixels remaining outside the ranges are removed to refine the silhouette. $\phi(I|v_i = head)$ evaluating the likelihood of point i to be assigned the label '*head*' gets a value of e^c $(c = 1)$ for the pixel marked as '*faces*' and a value of one in other cases.

Together with face detection, an additional function $f(\mathbf{x}_i)$ (related to the concept of soft objects [18]) is defined to estimate the torso location. If we let the center of the body O_{body} lie at a middle point between the center of the face and the center of the silhouette. $f(\mathbf{x}_i)$ is computed in the following way:

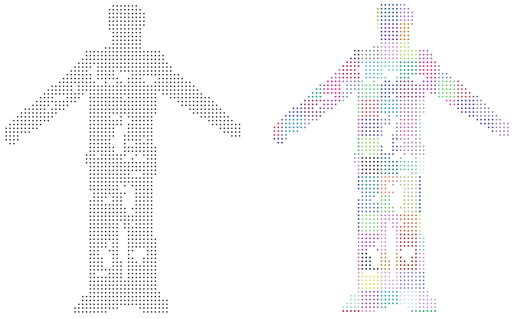
$$f(\mathbf{x}_i) = \kappa e^{-d(\mathbf{x}_i)} \tag{10}$$

where $d(\mathbf{x}_i)$ is the algebraic distance from the point $\mathbf{x}_i = [x, y, 1]^T$ to the ellipse with the centroid O_{body} and κ $(\kappa = e)$ is a positive constant. In the coordinate system attached to the origin O_{body}, $d(\mathbf{x}_i) = \mathbf{x}_i{}^T \mathbf{Q}_e^T \mathbf{D}_e \mathbf{Q}_e \mathbf{x}_i - 1$ where $\mathbf{D}_e$ and $\mathbf{Q}_e$ are the 3×3 matrices that determine the shape and orientation of the ellipse. The likelihood for identifying a single point as '*torso*' is given by

$$\phi(I|v_i = torso) = \begin{cases} f(\mathbf{x}_i) & \text{if } d(\mathbf{x}_i) \le 1, \\ 1 & \text{otherwise.} \end{cases} \tag{11}$$

3.1.3 Pairwise geodesic relationship among 3-D points

The Euclidean distance between any two points is only preserved within a rigid object. With a non-rigid object like the human body, the Euclidean distance will be changed due to the non-linear deformations of various body parts while the object is moving. However, with regard to the geodesic distance between a pair of points in space, this distance always retains its value during the movement of a human body. The preservation of geodesic distance has been successfully applied by the ISOMAP algorithm [27] to determine the manifold of high-dimensional data in a lower dimension. Here, we attempt to represent the geodesic relationships between each point and others in our probabilistic model. Some constraints are established to restrict the probability of incorrect label assignments. Assigning the pixels into groups called cells, as illustrated in Fig. 7, can help us save the computational time. All of the elements belonging to the same cell receive the same geodesic constraints. The geodesic distance between two cells is approximated by the shortest path distance in a graph using Dijkstra's algorithm [6]. The compatible probability $P(D|V)$ of the 3-D data with the geodesic constraints is given by



(a) (b)

Fig. 7 Assigning points into cells. (**a**) Sampling on the grid. (**b**) Points grouped by cells

$$P(D|V) = \prod_{i=1}^{M} \prod_{j_c=1}^{M_c} P_{geo}(D|v_i, v_{j_c}),$$

$$P_{geo}(D|v_i, v_{j_c}) = \begin{cases} e^{-\alpha} & d(v_{i_c}, v_{j_c}) < d_{min}(v_{i_c}, v_{j_c}), \\ e^{-\beta} & d(v_{i_c}, v_{j_c}) > d_{max}(v_{i_c}, v_{j_c}) \end{cases} \tag{12}$$

where i_c is the cell that holds pixel i, $d(v_{i_c}, v_{j_c})$ is the geodesic distance between the cell i_c and j_c, M_c is the number of cells, and α and β are two positive constants. Two values, $d_{min}(v_{i_c}, v_{j_c})$ and $d_{max}(v_{i_c}, v_{j_c})$, define the lower and upper bounds for the geodesic distance between a pair of labels. Two related labels that are too far or too close are penalized to decrease the belief in those labels. The constant values α and β are taken to be $\alpha = \beta = 0.04 M_c / M_{v_{j_c}}$, limiting the maximum number of cells which can ascertain to the label of the pixel i to 4% of the total. The maximum number of cells receiving the same label v_{j_c}, $M_{v_{j_c}}$, appears in the denominator as a normalized constant to ensure that $\prod_{j_c=1}^{M_c} P_{geo}(D|v_i, v_{j_c})$, the total effects to the pixel i, is approximately invariant to the size of body parts.

3.1.4 Reconstruction error

To co-register the ellipsoid model with the observations, we need to minimize the differences between them. The last term accounts for the compatible probability between the model specified by θ and the data D consisting a set of points $X_1, X_2, \ldots, X_M$. Let's denote $d(X_i, \theta, v_i)$ as the Euclidean distance between a point $X_i(x_i, y_i, z_i)$ and an ellipsoid v_i, as we already discussed in Sect. 2.3. $P(D|V,\theta)$ is defined as

$$P(D|V,\theta) = \prod_{i=1}^{M} e^{-\frac{d^2(X_i,\theta,v_i)}{2\sigma^2}} \tag{13}$$

where σ denotes the variance (σ^2 is chosen to be 0.1 in our experiments). The distance between the point X_i and the ellipsoids is also one of the factors that decides the body segment of X_i. Hence, in a sequence of frames, the estimated model from the current frame presents a good initial model to derive the label on the next frame.

3.2 Estimating the model parameters

Our main goal is to find the optimal kinematic parameter θ^* that maximizes the posterior probability of θ given the data. This problem can be rewritten as

$$\theta^* = \operatorname*{argmax}_{\theta} \sum_{V} P(V, \theta | I, D) \tag{14}$$

where V is considered the latent variable in this framework. The EM algorithm is a good choice for estimating the optimal values of the probabilistic problem with the appearances of unobserved variables. By introducing the distribution $Q(V)$ over the variable V [2], the problem in (14), equivalent with maximizing $\sum_V \log P(V,\theta|I,D)$, can be decomposed into

$$\operatorname*{argmax}_{\theta,Q} \sum_{V} Q(V) \log \frac{P(V,\theta|I,D)}{Q(V)} - \sum_{V} Q(V) \log \frac{P(V|\theta,I,D)}{Q(V)}. \tag{15}$$

The EM algorithm is an iterative procedure whose each iteration consists of the following two main steps:

(i) Assuming that the current value of θ is θ_{old}, the E-step evaluates the analytical expression of posterior distribution $Q(V)$ as

$$Q(V) = P(V|\theta, I, D). \tag{16}$$

(ii) The M-step maximizes

$$E_{Q_{old}(V)}[\log P(V,\theta|I,D)] \tag{17}$$

with respect to θ where $Q_{old}(V)$ is found from the previous E-step.

We provide the technical details of the E-step and M-step in the next sections, 3.2.1 and 3.2.2.

3.2.1 The E-step

The true distribution of $Q(V)$ in (16) is intractable to compute. Therefore, we perform the mean field approximation of $Q(V)$, which can be expressed as,

$$\log P(V|\theta, I, D) \propto \sum_{i=1}^{M} f_i(v_i) + \sum_{i=1}^{M} \sum_{j\subset\mathcal{N}_i} f_{ij}(v_i, v_j) + \sum_{i=1}^{M} \sum_{j_c=1}^{M_c} g_{ij}(v_i, v_{j_c}). \tag{18}$$

In this equation, $f(v_i)$ is the sum of the logarithms of the image likelihood term in (9) and the reconstruction error term in (13). $f(v_i, v_j)$ is determined by the logarithm of the compatible probability from the Potts model in (7). The pairwise $g(v_i, v_{j_c})$ is determined by the logarithm of the geodesic potential in (12), such that

$$g(v_i, v_{j_c}) = \log P_{geo}(D|v_i, v_{j_c}). \tag{19}$$

As in [28], the belief $q_i(v_i) = P(v_i|\theta, I, D)$ is iteratively updated until convergence:

$$q_{i_{step+1}}(v_i) = \frac{1}{Z_{i_{step}}(v_i)} \exp\left\{ \sum_{j_c=1}^{M_c} \sum_{v_{jc}} q^{j_c}_{step}(v_{j_c}) g_{ij}(v_i, v_{j_c}) + \sum_{j\subset\mathcal{N}_i} \sum_{v_j} q_{j_{step}}(v_j) f_{ij}(v_i, v_j) + f_i(v_i) \right\} \tag{20}$$

where $q^{j_c}_{step}(v_{j_c}) = E[q_{j_{step}}(v_{j_c})]$ is an average belief of all pixels $j \subset$ the cell j_c and $Z_{step}(v_i) = \sum_{v_i} q_{i_{step}}(v_i)$ is a normalization factor. To reduce the amount of computation required, set $q^{j_c}_{step}(v_{j_c} = \varepsilon) = 1$ for the maximum probability of the cell j_c pertaining to the ellipsoid ε and $q^{j_c}_{step}(v_{j_c}) = 0$ for $v_{j_c} \neq \varepsilon$. $f_i(v_i)$ is used to initialize the value of $q_i(v_i)$ in the first iteration, where

$$q_{i_0}(v_i) = \frac{1}{Z_{i_0}(v_i)} \exp\{f_i(v_i)\}. \tag{21}$$

In Fig. 8, we show the results of running the E-step on two examples in which the label of each point is selected by the label with the maximum belief. At the first iteration, using only the image likelihood and the distance provides incorrect labeling results because some pixels belonging to an arm are near to the torso or the head. After the E-step converges (three or four iterations), we obtain a correct labeling assignment.

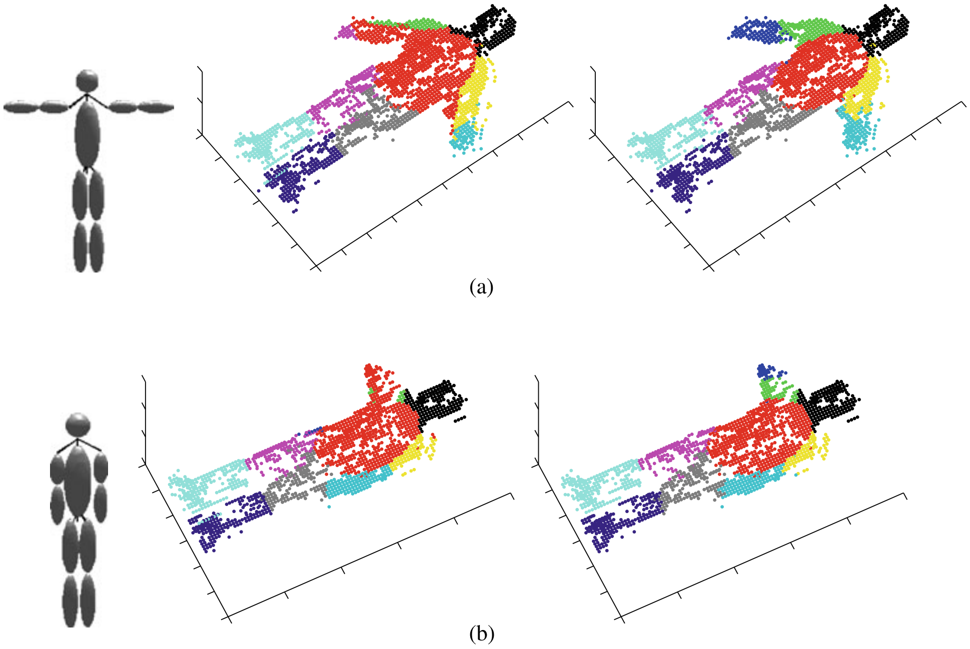


Fig. 8 The results of running the E-step on two examples (**a**) and (**b**). Corresponding from *left* to *right*: the initial human models, the label assignments found by the first iteration of the E-step, and the last iteration

3.2.2 The M-step

Once the distribution of the random variable v_i has been obtained, the kinematic parameter θ becomes the solution of the following optimization problem:

$$\operatorname*{argmax}_{\theta} E_{Q(V)}[\log P(D|\theta, V)]. \tag{22}$$

Here, the components independent of θ in (6) are eliminated. By taking the logarithm of $P(D|\theta, V)$, (22) can be rewritten as

$$-\operatorname*{argmax}_{\theta} \sum_{\varepsilon=1}^{N_\varepsilon} \sum_{i=1}^{M} q_i(v_i = \varepsilon) d^2(X_i, \theta, v_i = \varepsilon) \tag{23}$$

where N_ε is the number of ellipsoid, $d(X_i, \theta, v_i = \varepsilon) = \|X_i - Z_i(\theta)^\varepsilon\|^2$, and $Z_i(\theta)^\varepsilon$ is the nearest point of X_i lying on the surface of the ellipsoid ε. We formulate (23) in an alternative way as

$$\operatorname*{argmin}_{\theta} \sum_{\varepsilon=1}^{N_\varepsilon} \sum_{i=1}^{M} q_i(v_i = \varepsilon) \|X_i - Z_i(\theta)^\varepsilon\|^2. \tag{24}$$

For simplification of the M-step, set $q_i(v_i = \varepsilon) = 1$ for the maximum probability that the point i pertains to the ellipsoid ε and $q_i(v_i) = 0$ for $v_i \neq \varepsilon$. The least square problem with a nonlinear function like (24) can be efficiently solved by the Levenberg-Marquardt method. This estimator requires the computation of the Jacobian matrix $\mathbf{J}$ of $Z_i(\theta)^\varepsilon$ with respect to θ [13, 17, 25] that is explained in Appendix B.

4 Experiments

We have evaluated our proposed techniques with simulated and real data.

4.1 Experiments with simulated data

In generating the simulated data, we manually defined some joint angle trajectories as depicted by the dashed lines in Fig. 10. Only the rotational angles corresponding to the elbow and shoulder joints were tested in our experiments; the values of other rotational angles were fixed. From the predefined angle trajectories, we created a sequence of human postures and their disparity images up to 110 frames. Some samples of the disparity images are shown in the first and third rows in Fig. 9.

We applied our algorithm to recover the human postures from the synthetic disparity images. Due to the nature of simulated data, the cues of RGB images were not available, so we eliminated them from computation. Some samples of the recovered human poses are depicted in the second and fourth rows in Fig. 9. To validate our algorithm, we plotted the estimated angle trajectories as solid lines to compare against the synthetic angle trajectories plotted as the dashed line in Fig. 10. The results show the good estimation of the kinematic parameters achieved by our method.

4.2 Experiments with real data

Experiments were implemented with stereo data acquired by the stereo camera (Bumblebee 2.0 of Point Grey Research).

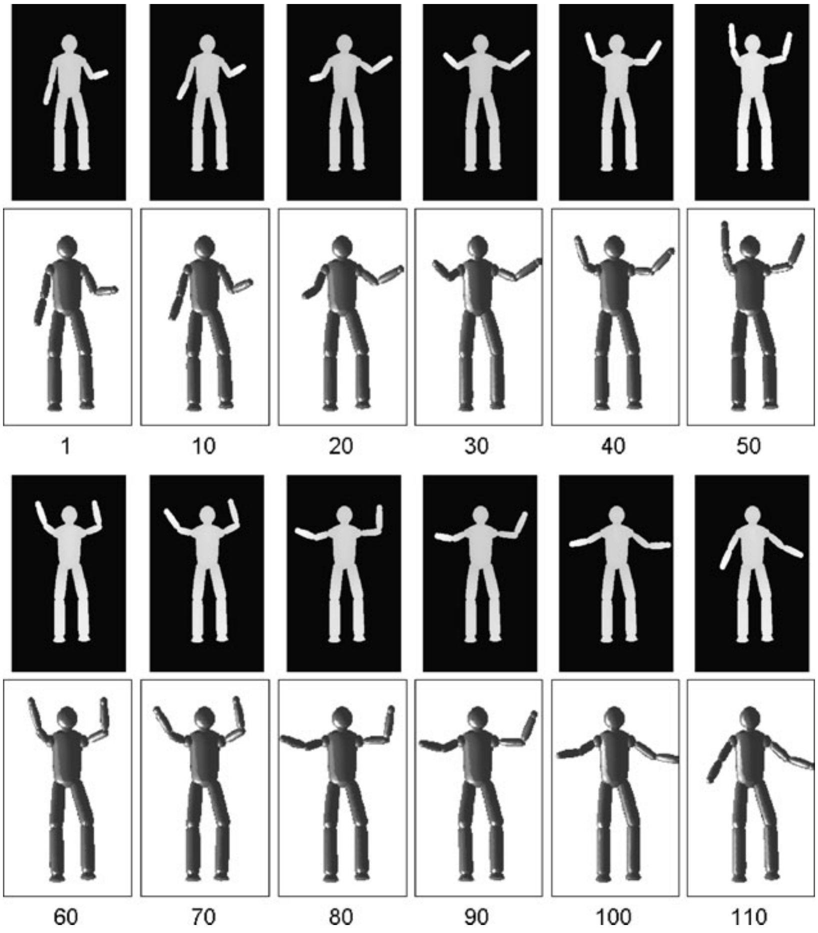


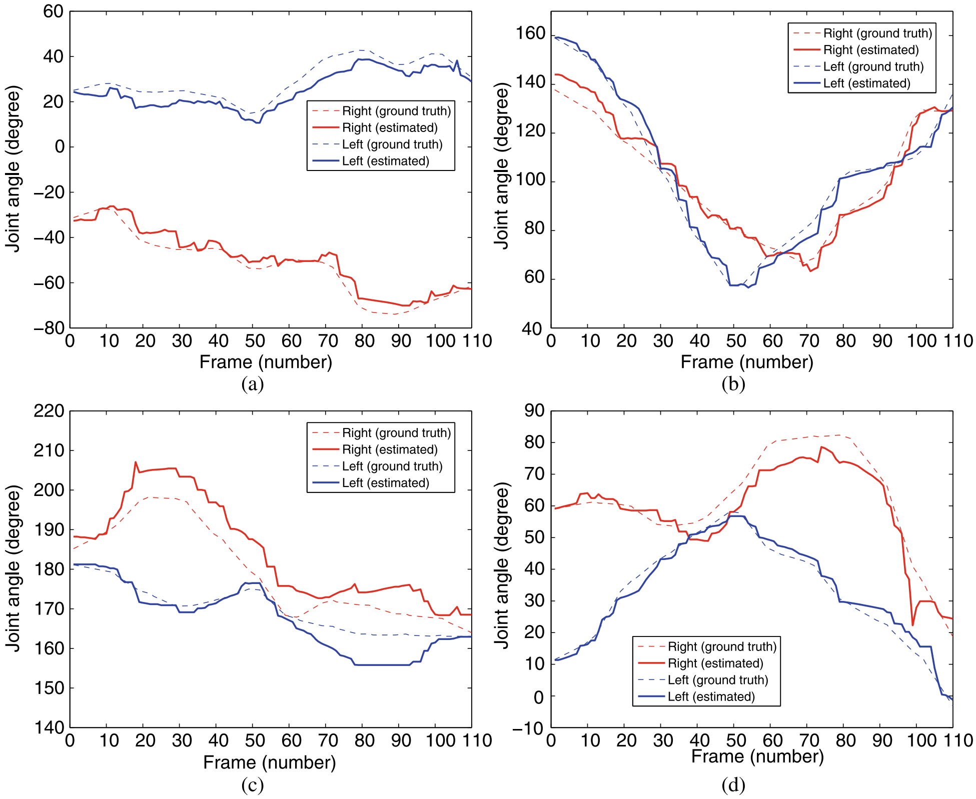
Fig. 9 The results of recovering human postures (*the second* and *fourth rows*) from the synthetic disparity images (*the first* and *third rows*). The number below each picture indicates the frame index number

In Sect. 2.1, we described the use of the GCS algorithm to extract the disparity image and compute the 3-D data for each frame. The subjects were asked to perform some distinguishable activities about 2–4 meters from the camera, producing several video sequences. The reconstructed body poses were validated by visually checking the trajectories of certain joint angles.

In the first experiment, we assessed the movements of elbows in both horizontal and vertical directions, as shown in Fig. 11. In each figure, the sequence of activities is illustrated in a video stream from top to bottom in a column. Observing the real pictures, the angle changes between the upper arm and lower arm were approximately 90°. In Fig. 12, the recovered angle of the second joint precisely reflects the arm motion in the real data. The joint angles may receive positive or negative values, depending on the way that two joint angles at the elbow are combined to drive the arm movements.

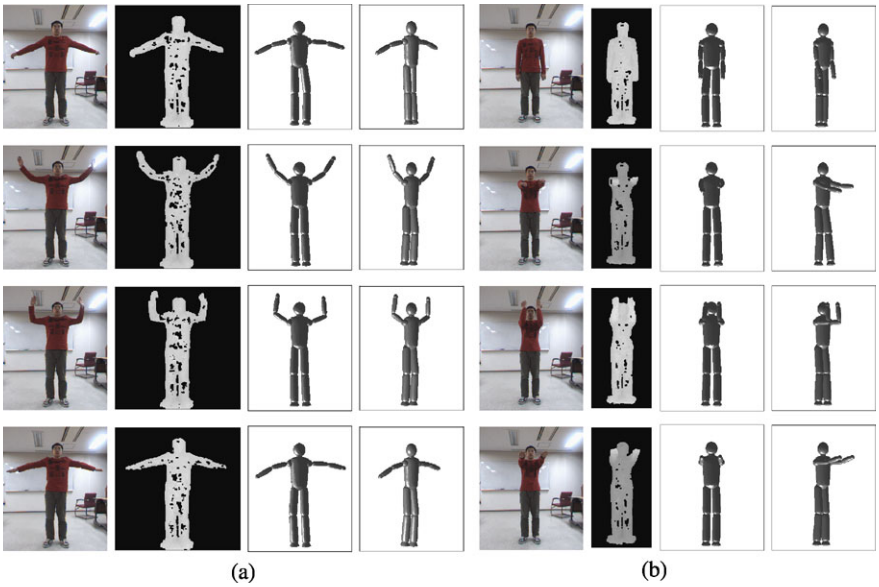
In the next test dataset, as shown in Fig. 13(a), the activity of the person in the video was related to the movements of the knee joint. The right leg was lifted until it made a 90° angle between the upper leg and lower leg, then this was followed by the same motion of the left leg. The kinematic motion parameters were estimated and are depicted in Fig. 15(a). One may notice that the switching between the two legs happens from frame 70 to frame 80.

In order to track the changes of two joint angles at the same time, we considered the sequence of activities in Fig. 13(b). We assumed that the whole arm laid along the x-axis and that the two joint angles of the shoulder were related to the rotation of the arm around the z-axis and x-axis, respectively. One can observe both trajectories of the two measured joint angles from Fig. 15(b), with the upper curves reflecting the rotational angles around the z-axis and the lower curves reflecting the rotational angles around the x-axis. To explain the meaning of the plot, we visualize the overall progress in Fig. 14. First, the whole arms were rotated around the z-axis from 180° to 360° $(+\pi)$, corresponding to the vertical movement within the frames 1–45. At the second stage, the second joint angles changed their values from 180° to 270° $(+\pi/2)$, while the arms retained their positions from frames 45 to 60. Finally, to be horizontally extended to the left or right side, the two arms were continuously rotated around the z-axis (the first joint angles)



(a) (b) (c) (d)

Fig. 10 A comparison between the estimated and the ground-truth joint angles in the simulated experiments (synthetic data). (**a**) and (**b**) show two joint angles of the shoulders. (**c**) and (**d**) show two joint angles of the elbows



(a) (b)

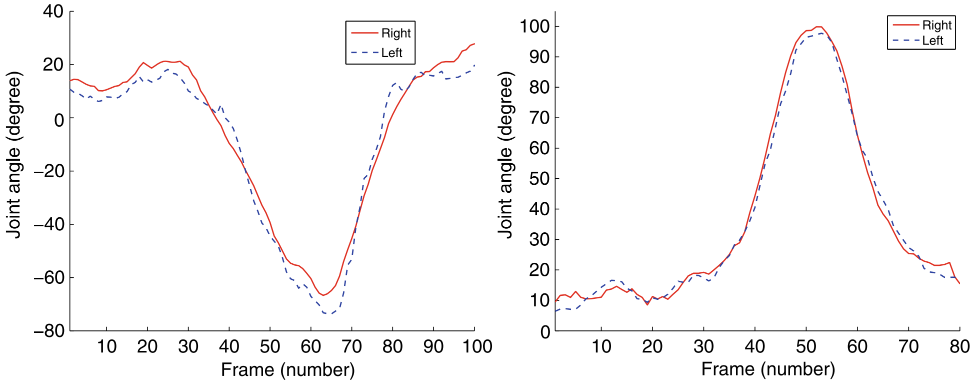
Fig. 11 Real experiments with elbow motion in two different directions. (**a**) Horizontal movements. (**b**) Vertical movements. From *left* to *right*: the RGB images, disparity images, and reconstructed human models (front view and +45° view)

from 360° to 270° ($-\pi/2$) or 450° ($+\pi/2$), corresponding to frames 60 to the end.

To quantitatively evaluate the reconstruction errors of these experiments, we needed to generate ground-truth using the given data. Applying the same method presented in [7, 12], the locations of some distinct points (e.g., hands, elbows, or shoulders) were hand-labeled in the RGB images. We used the 3-D information from these points to calculate the necessary ground-truth angles between two limbs. The angles reconstructed by the kinematic parameters were compared against the ground-truth by the average error ϵ_θ

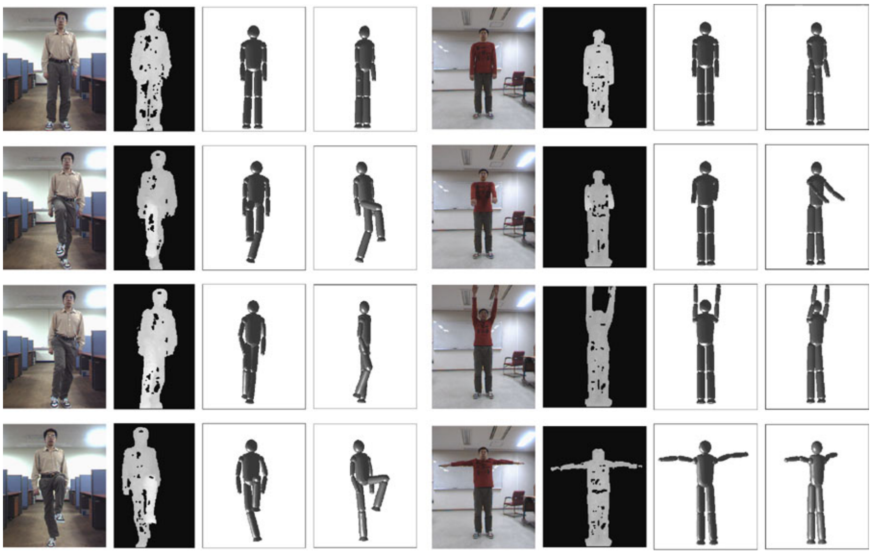
$$\epsilon_\theta = \frac{\sum_{t=1}^{n} |\theta_t^{est} - \theta_t^{grd}|}{n} \tag{25}$$

Fig. 12 The estimation of the second joint-angle trajectories for *the left* and *right* elbows corresponding to: (**a**) horizontal elbow movement and (**b**) vertical elbow movement



(a) (b)

Fig. 13 Real experiments with other motions: (**a**) Knee movements. (**b**) Shoulder movements. From *left* to *right*: the RGB images, disparity images, and reconstructed human models (front view and +45° view)



(a) (b)

Fig. 14 The changes in two joint-angles during the movements of the shoulders (experiment depicted in Fig. 13(b))

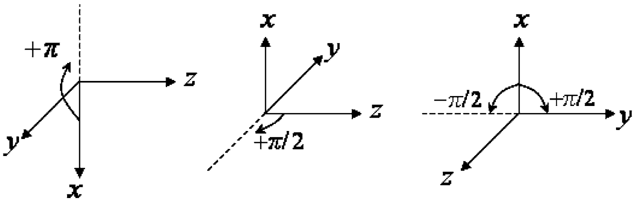
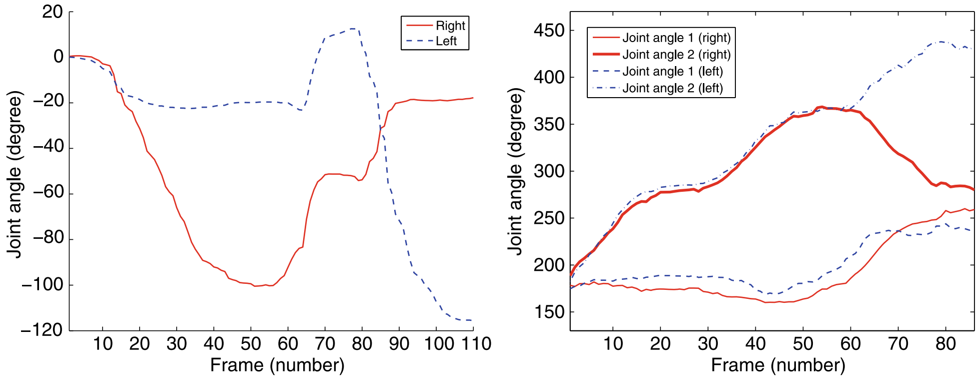
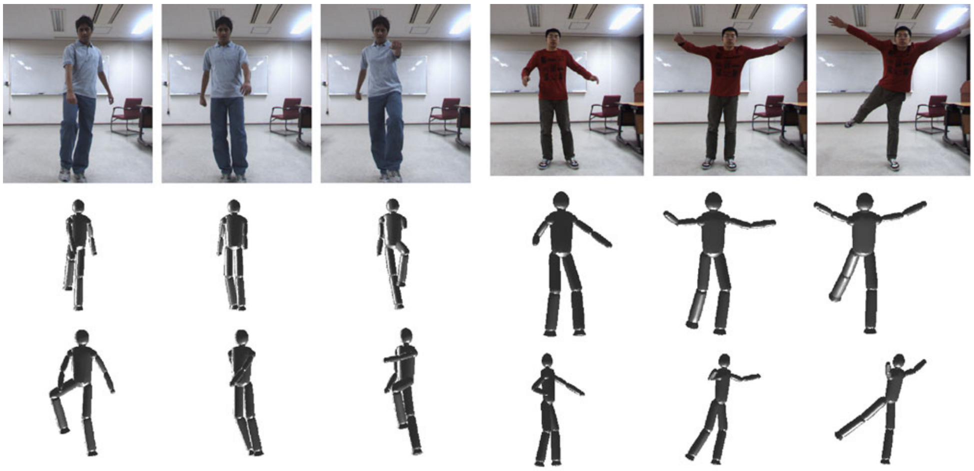


Fig. 15 The estimation of the joint-angle trajectories for *the left* and *right* sides of: (**a**) knee movements and (**b**) shoulder movements



(a) (b)

Fig. 16 The qualitative evaluation of the reconstructed human body postures from: (**a**) walking sequences and (**b**) arbitrary activity sequences.



(a) Reconstructed human body postures with the front view and -45^0 view.

(b) Reconstructed human body postures with the front view and $+45^0$ view.

Table 1 The average reconstruction error of the joint angles of the first four experiments. Note that these experiments only consider the local movements of some body limbs

Experiment	Elbow movement (horizontal direction)		Elbow movement (vertical direction)		Knee movement		Shoulder movement			
Evaluated angle	Upper arm & lower arm		Upper arm & lower arm		Upper leg & lower leg		Whole arm & x-axis		Whole arm & z-axis	
Average reconstruction	Left	Right	Left	Right	Left	Right	Left	Right	Left	Right
error (°)	8.21	7.58	6.79	7.64	8.03	13.81	5.66	5.72	9.08	9.97

where n is the number of frames, t is the frame index, θ_t^{grd} is the ground-truth, and θ_t^{est} is the estimated angle. In particular, the shoulder movements were related to two kinematic parameters, and therefore the correct arm directions were validated by measuring the angles between the arms and the x-axis or z-axis. The coordinate system (x, y, z) in this case had the x-axis and z-axis aligned with the vertical and horizontal directions of the image plane, respectively. The average errors of all four experiments are given in Table 1.

Figure 16 shows the results of testing our algorithm on some free movements. The subjects performed complicated activities with all of their arms and legs. Here, we depict only three images out of the sequence and their estimated poses in the second and third rows with two alternative view angles. The 3-D locations of the body parts and the correct human poses were successfully identified. In these experiments, it is more convenient to evaluate the estimated whole body pose, rather than the local changes of individual limbs. The average distance between each 3-D point and the nearest ellipsoid of the reconstructed model can be considered the overall error measurement of the reconstructed pose in each frame. The average distance D_t of the frame t is computed by

$$D_t = \frac{\sum_{i=1}^{M} d_t(i)}{M} \tag{26}$$

where $d_t(i)$ is the Euclidean distance from the point i to the nearest ellipsoid and M is the number of points. The means and standard derivations of D_t in the two last sequences are provided in Table 2.

Table 2 The mean and standard derivation of the average distance (the average Euclidean distance between a set of 3-D points of the observed data and the ellipsoids of the reconstructed model) of the last two sequences

Sequences	Walking	Arbitrary activity
Mean (m)	0.062	0.037
Std. dev. (m)	0.003	0.002

5 Conclusion and discussion

This paper describes a novel approach for recovering 3-D human body posture information from images acquired by a stereo camera. The technique is formulated in a unique probabilistic framework with the combination of various potentials including the smoothness term, image likelihoods, reconstruction errors, and geodesic distance constraints, and implemented by the two-stage optimization of the EM algorithm. Through experiments using synthetic and real data, we demonstrated that our algorithm can reconstruct human body posture from stereo video even for complicated movements. Analyzing the performance, we detected an average error of about 6–14° of the estimated kinematic angles and an average distance (i.e., difference) of about 0.04–0.06 m between the reconstructed body model and the given 3-D data. However, in certain cases, some limitations of our technique might exist. For example, the self occlusion of the human body may affect the accuracy of the algorithm. Moreover, solving the inverse kinematic problem by the Levenberg-Marquardt least-square estimator in the M-step may return the local optima.

For future development, we plan to utilize other cues such as edges, binary silhouette shapes, or optical flows to better detect the body parts and handle occlusion. A tracking algorithm using a particle filter can be used to discover the density distribution of kinematic parameters by sampling and can also take the temporal information into account. In fact, integrating a tracking algorithm into our probabilistic framework might allow faster recovery of posture information and overcome the local optima problem.

In practice, to be useful for a broad range of applications, camera perspectives also need to be considered in the implementation of this system. A frontal view, as examined in the present study, is suitable for various applications related to human-machine interaction such as games and remote controls using body gestures. In order to be applied to tracking and monitoring applications, cameras might be installed with proper view angles to obtain wider views of subjects and to avoid occlusions. In such cases, the global transformation of the whole body from the global coordinate system estimated by our algorithm should be able to handle variation in camera perspectives. Also, the known location of the cameras represented by the perspective transformation matrix could be integrated into the computation of the global transformation matrix to improve the accuracy and speed of the algorithm.

Acknowledgements This research was supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2009-(C1090-0902-0002)). This work also was supported by a Korea Science and Engineering Foundation (KOSEF) grant funded by the Korea government (MEST No. 2008-1342), and was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (2009-0076798).

Appendix A: Locating the nearest point in an ellipsoid surface to a given point

In Fig. 4, the transformation of X_0 into the local coordinate system (x', y', z') attached to the ellipsoid is obtained as

$$[x'_0, y'_0, z'_0, 1]^T = \mathbf{SQ}_\theta [x_0, y_0, z_0, 1]^T. \tag{27}$$

In the 2-D coordinate system of the plane P (the origin of the plane P lies at the centroid of the ellipsoid), these coordinates are converted to $(\sqrt{x'^2_0 + y'^2_0}, z'_0)$. The intersection between the plane P and the ellipsoid will be an ellipse with the major axis c and the minor axis a. Hence, the nearest point X_t belonging to the ellipsoid surface of X_0 in the plane P has the 2-D coordinate (u, v) as the roots of the equation

$$\begin{aligned} &f(u, v) = \frac{u^2}{a^2} + \frac{v^2}{c^2} - 1 = 0, \\ &\Big(\sqrt{x'^2_0 + y'^2_0} - u\Big)\frac{\partial f(u, v)}{\partial v} = (z'_0 - v)\frac{\partial f(u, v)}{\partial u}. \end{aligned} \tag{28}$$

This equation can be converted to a fourth-degree polynomial equation to find u and v. The coordinate of X_t in (x', y', z') is given by

$$x'_t = u\frac{x'_0}{\sqrt{x'^2_0 + y'^2_0}}, \qquad y'_t = u\frac{y'_0}{\sqrt{x'^2_0 + y'^2_0}}, \qquad z'_t = v. \tag{29}$$

We expand the updated rules for computing $X_t(x'_t, y'_t, z'_t)$ when X_0 moves to $X_0 + dX_0$ in the local coordinate system (x', y', z'). Let $k = a^2/c^2$ be a constant. Let

$$\begin{aligned} &u = \sqrt{x'^2_t + y'^2_t}, v = z'_t, \\ &r'_0 = \sqrt{x'^2_0 + y'^2_0}, \\ &\cos\gamma = x'_0/r'_0, \qquad \sin\gamma = y'_0/r'_0. \end{aligned} \tag{30}$$

The new value of $X_t(x'_t, y'_t, z'_t)$ corresponding to $X_0 + dX_0 = (x'_0 + dx'_0, y'_0 + dy'_0, z'_0 + dz'_0)$ in the local coordinate system (x', y', z') is computed by

$$\begin{aligned} &\xi = \frac{kv(\cos\gamma dx'_0 + \sin\gamma dy'_0) - udz'_0}{(1-k)(kv^2 - u^2) - k(vz'_0 + ur'_0)}, \\ &u = u - kv\xi, \qquad v = v + u\xi, \\ &x'_t = u\cos\gamma, \qquad y'_t = u\sin\gamma, \qquad z'_t = v. \end{aligned} \tag{31}$$

Note that when the point moves from outside into the ellipsoid or vice versa, x'_t, y'_t, and z'_t need to be recomputed from (28) and (29).

Transforming $X_t(x'_t, y'_t, z'_t)$ back to the global coordinate system (x, y, z), the coordinate of X_t is given by

$$[x_t, y_t, z_t, 1]^T = \mathbf{Q}_\theta^{-1}\mathbf{S}^{-1}[x'_t, y'_t, z'_t, 1]^T.$$

Appendix B: Computation of the Jacobian matrix for the inverse kinematic problem

In this appendix, we focus on the computation of the Jacobian matrix $\mathbf{J}$ of $Z_i(\theta)^\varepsilon$ with respect to θ. Assuming that the ellipsoid ε depends on the n_ε parameters $\theta_1, \theta_2, \ldots, \theta_{n_\varepsilon}$, $[Z_i(\theta)^\varepsilon, 1]^T$ must satisfy equation (3) with $\theta = (\theta_1, \theta_2, \ldots, \theta_{n_\varepsilon})$. Because $Z_i(\theta)^\varepsilon$ belongs to an ellipsoid surface in the global coordinate system, we apply a series of transformations to $Z_i(\theta)^\varepsilon$ to get one point $Z0_i^\varepsilon$, independent of θ, lying in an ellipsoid surface in the local coordinate system

$$\begin{aligned}&\mathbf{S}\mathbf{Q}_{n_\varepsilon}(\theta_{n_\varepsilon})\mathbf{Q}_{n_\varepsilon-1}(\theta_{n_\varepsilon-1})\ldots\mathbf{Q}_1(\theta_1)[Z_i(\theta)^\varepsilon,1]^T = [Z0_i^\varepsilon,1]^T\\ &\text{or}\quad [Z_i(\theta)^\varepsilon,1]^T = \mathbf{Q}_1(\theta_1)^{-1}\mathbf{Q}_2(\theta_2)^{-1}\ldots\mathbf{Q}_{n_\varepsilon}(\theta_{n_\varepsilon})^{-1}\\ &\qquad\qquad \mathbf{S}^{-1}[Z0_i^\varepsilon,1]^T.\end{aligned} \tag{32}$$

The Jacobian matrix $\mathbf{J}$ consists of n_ε columns, where each column i, $\partial Z_i(\theta)^\varepsilon/\partial\theta_i$ is given by

$$\begin{aligned}\left[\frac{\partial Z_i(\theta)^\varepsilon}{\partial\theta_i}, 0\right]^T &= \mathbf{Q}_1(\theta_1)^{-1}\mathbf{Q}_2(\theta_2)^{-1}\ldots\frac{\partial\mathbf{Q}_i(\theta_i)^{-1}}{\partial\theta_i}\\ &\quad \mathbf{Q}_{i+1}(\theta_{i+1})^{-1}\ldots\mathbf{Q}_{n_\varepsilon}(\theta_{n_\varepsilon})^{-1}\\ &\quad \mathbf{S}^{-1}[Z0_i{}^\varepsilon, 1]^T\\ &= \mathbf{Q}_1(\theta_1)^{-1}\mathbf{Q}_2(\theta_2)^{-1}\ldots\frac{\partial\mathbf{Q}_i(\theta_i)^{-1}}{\partial\theta_i}\\ &\quad \mathbf{Q}_i(\theta_i)\ldots\mathbf{Q}_2(\theta_2)\mathbf{Q}_1(\theta_1)[Z_i(\theta)^\varepsilon, 1]^T.\end{aligned} \tag{33}$$

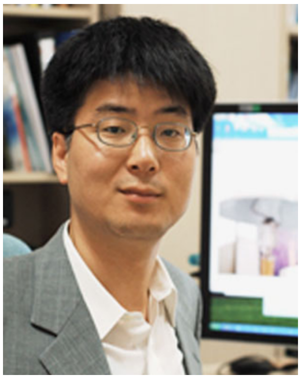
References

1. Agarwal A, Triggs B (2006) Recovering 3D human pose from monocular images. IEEE Trans Pattern Anal Mach Intell 28(1):44–58
2. Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin
3. Boykov Y, Veksler O, Zabih R (2001) Fast approximate energy minimization via graph cuts. IEEE Trans Pattern Anal Mach Intell 23(11):1222–1239
4. Cech J, Sara R (2007) Efficient sampling of disparity space for fast and accurate matching. In: IEEE conf on computer vision and pattern recognition, Minneapolis, MN, US, June 2007, pp 1–8
5. Conaire CO, O'Connor NE, Smeaton AF (2007) Detector adaption by maximising agreement between independent data sources. In: IEEE conf on computer vision and pattern recognition, Minneapolis, MN, USA, June 2007, pp 1–6
6. Dijkstra EW (1959) A note on two problems in connexion with graphs. Numer Math 1:269–271
7. Gupta A, Mittal A, Davis LS (2008) Constraint integration for efficient multiview pose estimation with self-occlusions. IEEE Trans Pattern Anal Mach Intell 30(3):493–506
8. Heckbert PS (1994) Graphics gems IV. Academic Press, San Diego
9. Horaud R, Niskanen M, Dewaele G, Boyer E (2009) Human motion tracking by registering an articulated surface to 3D points and normals. IEEE Trans Pattern Anal Mach Intell 31(1):158–163
10. Hua G, Yang M, Wu Y (2005) Learning to estimate human pose with data driven belief propagation. In: IEEE conf on computer vision and pattern recognition, vol 2, San Diego, CA, USA, June 2005, pp 747–754
11. Knossow D, Ronfard R, Horaud R (2008) Human motion tracking with a kinematic parameterization of extremal contours. Int J Comput Vis 79(3):247–269
12. Lee MW, Cohen I (2006) A model-based approach for estimating human 3D poses in static images. IEEE Trans Pattern Anal Mach Intell 28(6):905–916
13. McCarthy JM (1990) Introduction to theoretical kinematics. MIT Press, Cambridge
14. Menier C, Boyer E, Raffin B (2006) 3D skeleton-based body pose recovery. In: Third international symposium on 3D data processing, visualization, and transmission, Chapel Hill, NC, USA, June 2006, pp 389–396
15. Mori G, Malik J (2006) Recovering 3D human body configurations using shape contexts. IEEE Trans Pattern Anal Mach Intell 28(7):1052–1062
16. Mori G, Ren X, Efros AA, Malik J (2004) Recovering human body configurations: combining segmentation and recognition. In: IEEE conf on computer vision and pattern recognition, vol 2, Washington, DC, USA, July 2004, pp 326–333
17. Murray RM, Li Z, Sastry SS (1994) A mathematical introduction to robotic manipulation. CRC Press, Boca Raton
18. Plankers R, Fua P (2003) Articulated soft objects for multiview shape and motion capture. IEEE Trans Pattern Anal Mach Intell 25(9):1182–1187
19. Ramanan D, Forsyth DA, Zisserman A (2007) Tracking people by learning their appearance. IEEE Trans Pattern Anal Mach Intell 29(1):65–81
20. Ren X, Berg AC, Malik J (2005) Recovering human body configurations using pairwise constraints between parts. In: IEEE international conf on computer vision, vol 1, Beijing, China, October 2005, pp 824–831
21. Roberts TJ, McKenna SJ, Ricketts IW (2007) Human pose estimation using partial configurations and probabilistic regions. Int J Comput Vis 73(3):285–306
22. Rosales R, Sclaroff S (2000) Specialized mappings and the estimation of human body pose from a single image. In: IEEE workshop on human motion (HUMO), Austin, TX, USA, December 2000, pp 19–24
23. Sudderth EB, Ihler AT, Freeman WT, Willsky AS (2003) Nonparametric belief propagation. In: IEEE conf computer vision and pattern recognition, vol 1, Madison, WI, USA, June 2003, pp 605–612
24. Sundaresan A, Chellappa R (2008) Model driven segmentation of articulating humans in Laplacian Eigenspace. IEEE Trans Pattern Anal Mach Intell 30(10):1771–1785
25. Sundaresan A, Chellappa R, RoyChowdhury R (2004) Multiple view tracking of humans modelled by kinematic chains. In: IEEE conf on image processing, vol 2, Singapore, October 2004, pp 1009–1012
26. Taylor CJ (2000) Reconstruction of articulated objects from point correspondences in a single uncalibrated image. Comput Vis Image Underst 80(3):349–363
27. Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 209(5500):2319–2323

28. Toyoda T, Hasegawa O (2008) Random field model for integration of local information and global information. IEEE Trans Pattern Anal Mach Intell 30(8):1483–1489
29. Urtasun R, Fua P (2004) 3D human body tracking using deterministic temporal motion models. In: European conference on computer vision, vol 3, Prague, Czech Republic, May 2004, pp 92–106
30. Wang L, Tan T, Ninh H, Hu W (2003) Silhouette analysis-based gait recognition for human identification. IEEE Trans Pattern Anal Mach Intell 25(12):1505–1518
31. Wang R, Leow WK (2005) Human body posture refinement by nonparametric belief propagation. In: IEEE conf. on image processing, vol 3, Genoa, Italy, September 2005, pp 1272–1275
32. Yang HD, Lee SW (2007) Reconstruction of 3D human body pose from stereo image sequences based on top-down learning. J Pattern Recogn 40(11):3120–3131

Nguyen Duc Thang received his B.E. degree in Computer Engineering from Posts and Telecommunications Institute of Technology, Vietnam. He is currently working toward his M.S. leading to Ph.D. degree in the Department of Computer Engineering at Kyung Hee University, South Korea. His research interests include artificial intelligence, computer vision, and machine learning.

Tae-Seong Kim received the B.S. degree in Biomedical Engineering from the University of Southern California (USC) in 1991, M.S. degrees in Biomedical and Electrical Engineering from USC in 1993 and 1998 respectively, and Ph.D. in Biomedical Engineering from USC in 1999. After his postdoctoral work in cognitive sciences at the University of California, Irvine in 2000, he joined the Alfred E. Mann Institute for Biomedical Engineering and Dept. of Biomedical Engineering at USC as a Research Scientist and Research Assistant Professor. In 2004, he moved to Kyung Hee University in South Korea where he is currently an Associate Professor in the Biomedical Engineering Department. His research interests have spanned various areas of biomedical imaging including Magnetic Resonance Imaging (MRI), functional MRI, E/MEG imaging, DT-MRI, transmission ultrasonic CT, and Magnetic Resonance Electrical Impedance Imaging. Lately he has started research work in proactive computing at the u-Lifecare Research Center where he serves as Vice Director. Dr. Kim has been developing advanced signal and image processing methods, pattern classification and machine learning methods, and novel medical imaging and rehabilitation instruments and technologies. Dr. Kim has published more than 50 peer reviewed papers and 100 proceedings, and holds 3 international patents. He is a member of IEEE, KOSOMBE, and Tau Beta Pi, and listed in Who's Who in the World ('09,'10).



Young-Koo Lee received his B.S., M.S., and Ph.D. degrees in Computer Science from the Korea Advanced Institute of Science and Technology, South Korea. He is a professor in the Department of Computer Engineering at Kyung Hee University, South Korea. His research interests include ubiquitous data management, data mining, and databases. He is a member of IEEE, the IEEE Computer Society, and the ACM.

Sungyoung Lee received his B.S. from Korea University, Seoul, South Korea. He got his M.S. and Ph.D. degrees in Computer Science from Illinois Institute of Technology (IIT), Chicago, Illinois, USA in 1987 and 1991 respectively. He has been a professor in the Department of Computer Engineering, Kyung Hee University, South Korea since 1993. He is a founding director of the Ubiquitous Computing Laboratory, and has been affiliated with a director of Neo Medical ubiquitous-Life Care Information Technology Research Center, Kyung Hee University since 2006. Before joining Kyung Hee University, he was an assistant professor in the Department of Computer Science, Governors State University, Illinois, USA from 1992 to 1993. His current research focuses on Ubiquitous Computing and applications, Context-aware Middleware, Sensor Operating Systems, Real-Time Systems and Embedded Systems. He is a member of the ACM and IEEE.