Computer Vision and Image Understanding 155 (2017) 150–161

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Single image 3D human pose estimation using a procrustean normal distribution mixture model and model transformation

Jungchan Cho[a], Minsik Lee[b], Songhwai Oh[a,*]

[a] Department of Electrical and Computer Engineering and ASRI, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Korea
[b] Division of Electrical Engineering, Hanyang University, 55 Hanyangdaehak-ro, Sangnok-Gu, Ansan, Gyeonggi-do 15588, Korea

ARTICLE INFO

Article history:
Received 13 January 2016
Revised 31 October 2016
Accepted 1 November 2016
Available online 2 November 2016

Keywords:
Human pose estimation
3D Shape recovery
3D Human pose estimation
3D Reconstruction

ABSTRACT

3D human pose estimation from a single image is an important problem in computer vision with a number of applications, including action recognition and scene understanding. However, it is still challenging due to its ill-posedness and complex non-rigid shape variations of a human body. In this paper, we use the Procrustean normal distribution mixture model as a 3D shape prior and propose a model transformation method for adjusting limb lengths of the 3D shape prior model, by which the proposed method can be applied to a novel test image. Inaccuracies of 2D part detections are handled by selecting from a diverse set of 2D pose candidates considering both the 2D part model and 3D shape model. Experimental results show that the proposed method performs favorably compared with existing methods, despite inaccuracies of 2D part detections and 3D shape ambiguities.

1. Introduction

Estimating a 3D human pose from a single image has received a significant attention in computer vision due to its wide range of potentially useful applications, such as human-computer interaction, intelligent surveillance, and scene understanding, to name a few. In particular, understanding the 3D human pose is essential for view invariant action recognition. However, recovering a 3D structure of an articulated object, such as a human body, from images is still considered as a challenging problem.

The main difficulty is due to the high degree of freedom of a complex articulated object (Bregler et al., 2000; Cho et al., 2013; Gotardo and Martinez, 2011; Paladini et al., 2009; Torresani et al., 2008; Xiao et al., 2006) and automatic recovery of a 3D human pose from a single view is even more challenging. While there are various approaches for monocular 3D human pose estimation (Agarwal and Triggs, 2004; Bo and Sminchisescu, 2009; 2010; Fan et al., 2014; Ionescu et al., 2014; Kostrikov and Gall, 2014; Pons-Moll et al., 2014; Radwan et al., 2013; Ramakrishna et al., 2012; Sigal et al., 2007; Simo-Serra et al., 2013; 2012; Wang et al., 2014; Zhou and Leonardos, 2015), our work focuses on recovering a 3D human pose using 2D part locations obtained from an image, which are more robust against changes in view point.

In general, 3D human pose estimation based on 2D body part locations is done by first detecting body parts from the image and then recovering a 3D pose using a 3D shape model of a human. However, currently available 2D part detectors cannot accurately localize key joints in all cases. In addition, recovering a 3D shape from its projection in a 2D image is inherently an ill-posed problem because different 3D shapes may generate similar 2D projections (Radwan et al., 2013; Simo-Serra et al., 2012). While there have been many efforts to estimate a 3D human pose from 2D part detections with the prior information about a 3D human body, developing a sound mathematical model is still an open issue.

The overall structure of the proposed method is shown in Fig. 1. In order to handle inaccuracies of 2D part detections, we generate a diverse set of 2D part candidates by decomposing and recombining multiple 2D part detections, and then select the one which explains the 2D part model and 3D shape model the best.

To overcome the complexity of human shapes and noisy observations, we apply the Procrustean normal distribution (PND) (Lee et al., 2013) in form of a mixture of PNDs (Cho et al., 2016): We learn the prior information about 3D configurations using a mixture of Procrustean normal distributions (Cho et al., 2016; Lee et al., 2013) and fit the mixture model (Cho et al., 2016) to 2D observations. Here, the mixture model (Cho et al., 2016) plays a role of making a set of specific pose subspaces in an unsupervised manner and, by restricting 3D pose estimation on a subspace, 3D pose estimation from a single image can be achieved.

* Corresponding author.
E-mail addresses: cjc83@snu.ac.kr (J. Cho), mleepaper@hanyang.ac.kr (M. Lee), songhwai@snu.ac.kr (S. Oh).

http://dx.doi.org/10.1016/j.cviu.2016.11.002
1077-3142/

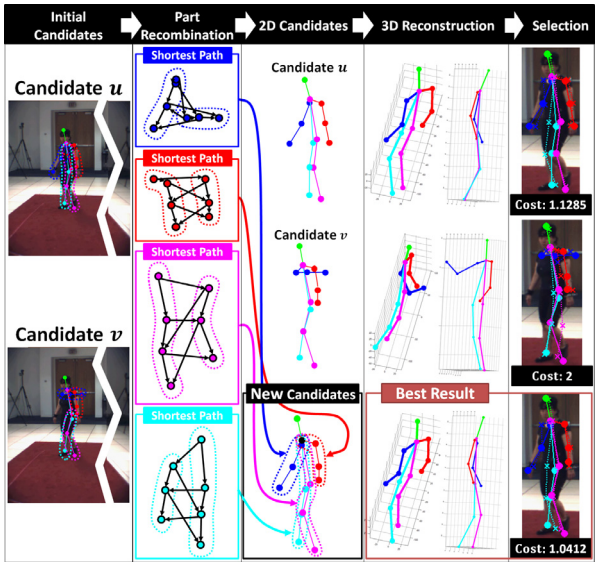


Fig. 1. An overview of the proposed method.

In prior work, the 3D human pose estimation problem often operates under the assumption that training and testing datasets are obtained from the same body part location settings, including (Cho et al., 2016). However, when a learned 3D shape model is applied to a novel image, a problem can arise since the test image may contain a human subject who has limb lengths significantly different from subjects in the training set due to different body part location setting. Hence, we focus on the 3D human pose estimation using heterogeneous datasets and propose a model transformation method, which consists of model normalization and model adaptation. In the model normalization step, we normalize limb lengths of mixture components using their mean limb lengths. The model adaptation step adjusts the normalized model to the 2D part detections in a novel image.

In our experiments, the proposed method is applied to the HumanEva dataset (Sigal et al., 2010) for testing, which is different from the CMU dataset used for training, as done in Radwan et al. (2013), and the proposed method shows better performance in many cases. In addition, we compare the proposed method with Simo-Serra et al. (2012) and Wang et al. (2014), which used the same 2D part detector (Yang and Ramanan, 2013), and show that the proposed method outperforms them by overcoming inaccuracies of 2D part detections and 3D shape ambiguities.

The main contributions of the proposed method are summarized as follows:

1. We generate a diverse set of 2D part detection candidates and choose the best candidate considering both the 2D part model and 3D shape model to overcome the inaccuracy of 2D part detection algorithms.
2. We show that a PND (Lee et al., 2013) provides a sound mathematical model for capturing the 3D shape information and it can be easily incorporated to recover a 3D shape from a single image.
3. We propose a method for adjusting limb lengths of a learned 3D shape model, such that the model can be applied to a novel test image with human subjects with limb lengths that are different from subjects in the training set.

The remainder of this paper is organized as follows: Section 2 reviews related work in 3D pose estimation from a single image. We review the Procrustean normal distribution (PND) (Lee et al., 2013) and its mixture model (Cho et al., 2016) in Section 3. Section 4 introduces a method for generating a set of diverse 2D pose candidates and Section 5 describes a method for a single view 3D human pose estimation. We then describe model transformation which consists of model normalization and model adaptation in Section 6. In Section 7, metrics for choosing a solution from a group of candidates are described. Experimental results are discussed in Section 8.

2. Related work

The problem of estimating a 3D human pose from a single image can be considered as a structured output regression problem based on 2D features, such as silhouettes (Agarwal and Triggs, 2004; Bo and Sminchisescu, 2009; 2010; Ionescu et al., 2014; Kostrikov and Gall, 2014; Pons-Moll et al., 2014; Sigal et al., 2007). Recently, Kostrikov and Gall (2014) used regression forests to infer missing depth data of image features and 3D pose simultaneously. Li and Chan (2014) proposed a deep convolutional neural network for 3D human pose estimation from monocular images. Such image feature based methods are inherently limited by the amount and quality of the training data, consequently they require a large number of training samples (2D image and 3D human pose pairs) to represent the appearance variability of different people and viewpoints. In contrast, 2D part detection based methods are proposed to fit a 3D shape model to detected 2D joint locations (Radwan et al., 2013; Simo-Serra et al., 2012; Wang et al., 2014), requiring much less training samples. However, estimating a 3D human pose in a single image, even if a set of 2D joint locations is given, is an inherently an ill-posed problem.

There have been studies in various directions to reduce the ambiguity. Lehrmann et al. (2013) proposed a non-parametric Bayesian network prior model of a human pose, which can synthesize realistic poses. Simo-Serra et al. (2014) proposed a novel approach to learn a finite mixture model on a Riemannian manifold and showed that it can be used for estimating occluded parts of a human body. Akhter and Black (2015) introduced a new prior to limit joint angles and remove invalid 3D human body poses.

When a 3D human pose is estimated directly from a single image, there can be an additional problem. Currently available 2D part detection algorithms frequently report incorrect body parts, degrading the performance of 3D pose estimation. In order to overcome inaccuracies in a part detection algorithm, Simo-Serra et al. (2012) used a stochastic sampling strategy which propagated 2D observation noises to the 3D shape space. Radwan et al. (2013) generated 2D part locations in synthetic views by regressing a set of 2D part locations from the input view to multiple oriented views. Then the pose was estimated using multi-view geometry. Wang et al. (2014) minimized the l_1-norm error between the projection of an estimated 3D pose and corresponding 2D detections, in which the 3D pose was represented as a linear combination of a sparse set of basis vectors of human poses.

In many prior work, it is assumed that a test image has the same joint placement condition as training images, often from the same dataset, making 3D pose estimation manageable. In this paper, we propose a method to overcome such limitations by allowing test images with joint placement conditions different from training images.

3. Preliminaries

3.1. Procrustean normal distribution

The Procrustean normal distribution (PND) (Lee et al., 2013) is a special case of the Gaussian distribution which describes the distribution of non-rigid shape deformation. The PND does not contain

changes due to rigid motion, since the PND is defined on a space which is orthogonal to the space spanned by rigid motion.[1]

Let $\mathbf{X} \in \mathbb{R}^{3\times n_p}$ be a centered 3D shape satisfying $\mathbf{X1} = \mathbf{0}$, where n_p is the number of landmarks and let $\mathbf{Y} \in \mathbb{R}^{3\times n_p}$ be an aligned 3D shape of $\mathbf{X}$, *i.e.*, $\mathbf{Y} = s\mathbf{R}\mathbf{X}$, where $s \in \mathbb{R}$ and $\mathbf{R} \in \mathbb{R}^{3\times 3}$ are the scale and rotation satisfying modified generalized Procrustes analysis (GPA) constraints (Lee et al., 2013), respectively. Then the probability density function of a PND random matrix $\mathbf{Y} \sim \mathcal{N}_P(\overline{\mathbf{Y}}, \mathbf{\Sigma})$ is defined as follows:

$$p(\mathbf{Y}) \propto \frac{1}{|\mathbf{\Sigma}_R|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\mathbf{v}^T\mathbf{Q}\mathbf{\Sigma}_R^{-1}\mathbf{Q}^T\mathbf{v}\right)\delta(\mathbf{Q}_N^T\mathbf{v}),$$
$$\mathbf{\Sigma}_R = \mathbf{Q}^T\mathbf{\Sigma}\mathbf{Q}, \quad \mathbf{v} = \mathbf{vec}(\mathbf{Y} - \overline{\mathbf{Y}}), \tag{1}$$

where $\overline{\mathbf{Y}} \in \mathbb{R}^{3\times n_p}$ is the mean shape of $\mathbf{Y}$, δ is the Dirac delta function, and $\mathbf{Q}_N \in \mathbb{R}^{3n_p\times 7}$ is a subspace for rigid motion, which can be computed from $\mathbf{Y}$ (Lee et al., 2013). Here, the degree of freedom of rigid shape variations is seven in a three dimensional space, *i.e.*, one for scale, three for rotation, and three for translation, respectively. $\mathbf{Q} \in \mathbb{R}^{3n_p\times(3n_p-7)}$ plays a role of removing rigid shape variations, *i.e.*, $\mathbf{Q}^T\mathbf{Q}_N = \mathbf{0}$, and the covariance matrix of non-rigid shape variations $\mathbf{Q}^T\mathbf{v}$ is denoted by $\mathbf{\Sigma}_R \in \mathbb{R}^{(3n_p-7)\times(3n_p-7)}$.

The PND has the following property (Lee et al., 2013): Let $\mathbf{Y} \sim \mathcal{N}_P(\overline{\mathbf{Y}}, \mathbf{\Sigma})$ be a PND random matrix and $\mathbf{Y}' = \mathbf{R}\mathbf{Y}$ for an orthogonal matrix $\mathbf{R}$. Then, $\mathbf{Y}'$ is also a PND random matrix and distributed as $\mathbf{Y}' \sim \mathcal{N}_P(\mathbf{R}\overline{\mathbf{Y}}, (\mathbf{I}\otimes\mathbf{R})\mathbf{\Sigma}(\mathbf{I}\otimes\mathbf{R}^T))$. If a distribution satisfies the properties of the PND except the scale constraint, it is called a scaled PND and denoted by $\mathcal{N}_P^s$.

3.2. Procrustean normal distribution mixture model

The Procrustean normal distribution mixture model (PNDMM) (Cho et al., 2016) is an extension of the PND to model complex non-rigid shape variations. A PNDMM can be represented as

$$p(\mathbf{X}) = \sum_{k=1}^{K} \pi_k p(\mathbf{X}|c_k = 1), \tag{2}$$

where K is the number of mixture components (Cho et al., 2016). The mixing probability for the kth component is defined as $\pi_k = p(c_k = 1)$, where $\pi_k \geq 0$, such that $\sum_k \pi_k = 1$ and $c_k \in \{0, 1\}$ indicates which mixture component has generated the sample. $p(\mathbf{X}|c_k = 1)$ is a PND corresponding to the kth component, which is defined as $\mathcal{N}_P(\mathbf{Y}_k|\overline{\mathbf{X}}_k, \mathbf{Q}_k\mathbf{\Sigma}_{R_k}\mathbf{Q}_k^T)$ (Lee et al., 2013), where $\overline{\mathbf{X}}_k$, $\mathbf{\Sigma}_{R_k}$, and $\mathbf{Q}_k$ are the mean of aligned 3D shapes, the covariance matrix for non-rigid variations, and the projection matrix to the linear subspace of non-rigid shapes, respectively, and $\mathbf{Y}_k = s_k\mathbf{R}_k\mathbf{X}$ is an aligned shape using scale s_k and rotation $\mathbf{R}_k$.

4. Candidate generation

We use a diverse set of candidate poses and select the best result to reduce 3D reconstruction failures due to incorrect 2D part detection, whereas existing approaches, such as Radwan et al. (2013); Simo-Serra et al. (2012) and Wang et al. (2014), use a single pose with the highest detection score for 3D reconstruction. Although there exists a method for generating multiple 2D part detections, such as the N-best extension (Park and Ramanan, 2011), we have found that resulting candidates often do not contain a good 2D pose. Hence, we generate additional candidates by recombining initial candidates obtained from the N-best extension (Park and Ramanan, 2011).

[1] In this paper, we use **0** to denote both a matrix and a vector of zeros.

4.1. Initial pose generation

Yang and Ramanan (2013) have proposed a 2D human pose estimation method by representing human body parts as a mixture of pictorial structures. Let $G = (V, E)$ be a relational tree, where V is a set of body parts and E is a set of edges connecting body parts. Then the score of a specific pose configuration is represented as follows (Yang and Ramanan, 2013):

$$S(I, z) = \sum_{j\in V} \Phi_j(I, z_j) + \sum_{(i,j)\in E} \Psi_{ji}(z_j, z_i), \tag{3}$$

where

$$z_j = (l_j, t_j)$$
$$\Phi_j(I, z_j) = w_j^{t_j} \cdot \phi(I, l_j) + b_j^{t_j}$$
$$\Psi_{ji}(z_j, z_i) = w_{ji}^{t_j,t_i} \cdot \psi(l_j - l_i) + b_{ji}^{t_j,t_i}.$$

For an edge (i, j), i denotes a parent node and j denotes a child node. Here, l_j is the location of part j and t_j is the configuration type for part j, *e.g.*, different hand appearances due to its orientation. The first sum represents the sum of local appearance scores computed by pre-trained template $w_j^{t_j}$ and HOG (Dalal and Triggs, 2005) descriptor $\phi(I, l_j)$ extracted at location l_j in image I. The second sum encodes shape deformations by $w_{ji}^{t_j,t_i}$, which is often interpreted as a spring between adjacent parts, and $\psi(l_j - l_i)$, which is the relative location of part j with respect to part i. $b_j^{t_j}$ and $b_{ji}^{t_j,t_i}$ are trained offsets. We can efficiently find z^* which maximizes $S(I, z)$ using dynamic programming (Yang and Ramanan, 2013) by sequentially optimizing from leaf nodes to the root node. The method has been extended to generate N-best candidates (Park and Ramanan, 2011), which can be used to find multiple detections.

Considering that 3D pose estimation using a single image can be highly ambiguous, it is important to use accurate 2D part detection results. To avoid reconstructing 3D poses based on incorrect part detection results, we utilize 2D pose candidates with high scores. After performing part specific non-maximum suppression, we select n_c 2D pose candidates with the highest pose detection scores. Additional pose candidates are generated from n_c candidates using the part recombination step described below.

4.2. Part recombination

For each pose candidate selected from the N-best extension (Park and Ramanan, 2011), we decompose a pose into four segments: a left arm, right arm, left leg, and right leg (see Fig. 1). All generated segments share the common neck and head and the positions of the neck and head are obtained from the part detection with the highest score. For each segment, a new segment is generated using corresponding segments from all candidates by solving a shortest path problem (see Fig. 1 under Part Recombination).

For each segment, if there is an edge $e_{u_i,u_j} = (u_i, u_j)$, we introduce new directed edges e_{u_i,v_j} for all candidates u and v, where u_i and v_j are the ith and jth joints from the uth and vth candidates, respectively (see Fig. 1). For segment s, let E_s be a set of all edges introduced for the segment. Let $\mathcal{P}_s$ be a set of all possible paths in E_s. Then we solve the following shortest path problem:

$$\min_{path\in\mathcal{P}_s} \sum_{(u_i,v_j)\in path} f(e_{u_i,v_j}), \tag{4}$$

where f consists of a part cost and neighborhood cost.

Part cost: We define the part cost for part i using the detection score of appearance, *i.e.*, $S_i = \Phi_i(I, z_i)$ in (3). We look for a path in

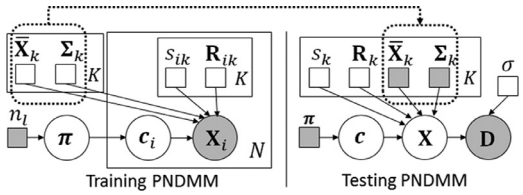


Fig. 2. A graphical illustration of the 3D human pose model (Section 5). In training a PNDMM, a 3D shape model is learned with true 3D shapes. In testing, 3D human pose estimation from 2D part locations is regarded as a fitting problem of the PNDMM (Cho et al., 2016) with known parameters.

E_s with high part detection scores, hence, we define an edge cost as follows:

$$f_p(e_{u_i,v_j}) = -S_i^u - S_j^v, \tag{5}$$

where S_i^u is the score for part i of candidate u.

Neighborhood cost: The neighborhood cost introduces constraints on limb lengths and it is defined as

$$f_n(e_{u_i,v_j}) = \left| dist(l_i^u, l_j^v) - \bar{l}_{ij}^{train} \right|, \tag{6}$$

where $dist(a, b)$ is the Euclidean distance between a and b, l_i^u is the location of part i of candidate u, and $\bar{l}_{ij}^{train}$ is a reference length of the limb obtained from the 2D part training data.

We normalize cost values between 0 and 1 since the ranges of two costs are not the same. Let f_p' and f_n' be the normalized part cost and neighborhood cost, respectively. The shortest path problem (4) is solved with

$$f(e_{u_i,v_j}) = f_p'(e_{u_i,v_j}) + f_n'(e_{u_i,v_j}). \tag{7}$$

When all segments are generated by solving (4), we combine them to make an additional candidate pose with the common neck position.

Candidate generation: In order to generate a diverse set of candidate poses, we perform the part recombination step repeatedly with a different set of initial candidates. The first recombined candidate is found using all n_c initial candidates and the second recombined candidate is found using $n_c - 1$ initial candidates by removing the candidate with the highest detection score. We repeat the process until at least two initial candidates are remained. In total, $n_c - 1$ recombined candidates are generated and will be considered for 3D reconstruction along with n_c initial candidates.

5. 3D Human pose model

Since we can consider a single view 3D human pose estimation as a fitting problem of a PNDMM if parameters of the mixture model is known, we learn a 3D shape model using true 3D shapes and apply the model to estimate a 3D pose from a single image as shown in Fig. 2.

5.1. 3D Shape model learning

In this section, we describe how to learn a PNDMM using true 3D shapes (Training PNDMM in Fig. 2). While the process is similar to the algorithm described in Cho et al. (2016), 3D shapes are learned from 3D data in our case, unlike 2D observations considered in Cho et al. (2016). Hence, the learning algorithm from Cho et al. (2016) is adjusted as follows to learn a PNDMM from 3D observations.

Given N training 3D shapes, let us define the joint distribution $p(\mathbf{X}, \mathbf{c})$ as

$$p(\mathbf{X}, \mathbf{c}) = \prod_{i=1}^{N} \prod_{k=1}^{K} \{p(\mathbf{X}_i|c_{ik} = 1)p(c_{ik} = 1)\}^{c_{ik}}, \tag{8}$$

where i and k correspond to the ith training sample and the kth component, respectively, $p(c_{ik} = 1) = \pi_k$, and $p(\mathbf{X}_i|c_{ik} = 1)$ is the kth PND. Following the expectation-maximization (EM) framework, the parameters of (8) can be learned by maximizing the expected complete log-likelihood:

$$\Upsilon(\Phi|\Phi^{old}) = \sum_i \sum_k w_{ik} \ln(p(\mathbf{X}_i, c_{ik} = 1|\Phi)) \tag{9}$$

where $\Phi = \{s_{ik}, \mathbf{R}_{ik}, \bar{\mathbf{X}}_k, \mathbf{\Sigma}_{R_k}, \mathbf{Q}_k, \pi_k | i = 1, \ldots, N, k = 1, \ldots, K\}$, N is the number of samples, and K is the number of components. Because the posterior distribution of c_{ik} plays the role as a weight for the component indicated by $\mathbf{c}_i$, we denote it as $w_{ik} = p(c_{ik} = 1|\mathbf{X}_i, \Phi^{old})$.

In practice, we do not know the number of mixture components. In order to estimate the number of components, we introduce a Dirichlet-type prior on π based on the minimum message length (MML) principle (Figueiredo and Jain, 2002): $p(\pi) \propto \exp\left(-\frac{n_l}{2}\sum_k \ln \pi_k\right)$ where $\pi = \{\pi_1, \ldots, \pi_K\}$, n_l is the user defined hyper-parameter, and the maximum a posteriori (MAP) estimate is the same as maximizing the following:

$$\Upsilon(\Phi|\Phi^{old}) = \sum_i \sum_k w_{ik} \ln(p(\mathbf{X}_i, c_{ik} = 1|\Phi)) + \ln(p(\pi)). \tag{10}$$

The superscript *old* denotes the parameter set obtained from the previous M-step in the EM iteration procedure and we will omit the superscript (*old*) if no confusion arises. The MAP estimate of (10) can be solved as follows.

In E-step: We estimate $w_{ik} = p(c_{ik} = 1|\mathbf{X}_i, \Phi^{old})$ given the current estimates of parameter Φ^{old} and observation $\mathbf{X}_i$. Using Bayes' rule, we have

$$w_{ik} = \frac{\pi_k p(\mathbf{X}_i|c_{ik} = 1, \Phi)}{\sum_j \pi_j p(\mathbf{X}_i|c_{ij} = 1, \Phi)}, \tag{11}$$

where $p(\mathbf{X}_i|c_{ik} = 1, \Phi)$ can be calculated as described in Lee et al. (2013) with $\mathbf{Y}_{ik} = \mathbf{s}_{ik}\mathbf{R}_{ik}\mathbf{X}_i$.

In M-step: The MAP solution of parameters are obtained using the posterior distribution of c_{ik}, *i.e.*, w_{ik} computed from the E-step. The parameters can be obtained by alternatively updating one parameter at a time for each component (see Cho et al. (2016) for details):

- The scale and rotation parameters, s_{ik} and $\mathbf{R}_{ik}$, for Procrustes alignment are calculated by

$$\mathbf{X}_i\bar{\mathbf{X}}_k^T = \mathbf{U}_{ik}\mathbf{\Lambda}^{ik}\mathbf{V}_{ik}^T, \quad \mathbf{R}_{ik} = \mathbf{V}_{ik}\mathbf{U}_{ik}^T,$$
$$s_{ik} = 1/\mathrm{tr}(\mathbf{R}_{ik}\mathbf{X}_i\bar{\mathbf{X}}_k^T) = 1/\mathrm{tr}(\mathbf{\Lambda}_{ik}), \tag{12}$$

  where $\mathbf{U}_{ik}\mathbf{\Lambda}^{ik}\mathbf{V}_{ik}^T$ is the singular value decomposition of $\mathbf{X}_i\bar{\mathbf{X}}_k^T$.
- The mean shape is calculated by

$$\bar{\mathbf{X}}_k = \frac{\sum_i w_{ik}s_{ik}\mathbf{R}_{ik}\mathbf{X}_i}{\|w_{ik}s_{ik}\mathbf{R}_{ik}\mathbf{X}_i\|_F}. \tag{13}$$

  $\mathbf{Q}_{N_k}$ and $\mathbf{Q}_k$ can be updated by using the new $\bar{\mathbf{X}}_k$ as done in Lee et al. (2013) and Cho et al. (2016). The covariance matrix for non-rigid shapes is

$$\mathbf{\Sigma}_{R_k} = \frac{\sum_i w_{ik}\mathbf{h}_{ik}\mathbf{h}_{ik}^T}{\sum_i w_{ik}}, \tag{14}$$

  where $\mathbf{h}_{ik} = \mathbf{Q}_k^T(s_{ik}(\mathbf{I} \otimes \mathbf{R}_{ik})\mathbf{vec}(\mathbf{X}_i) - \mathbf{vec}(\bar{\mathbf{X}}_k))$.
- Finally, the parameter for a mixing probability is

$$\pi_k = \frac{\max\left(0, \sum_i w_{ik} - \frac{n_l}{2}\right)}{\sum_k \max\left(0, \sum_i w_{ik} - \frac{n_l}{2}\right)}. \tag{15}$$

  Here, we are performing component annihilation (Figueiredo and Jain, 2002) by removing components with $\Sigma_i w_{ik} \leq n_l/2$.

Each component is updated sequentially (Cho et al., 2016; Figueiredo and Jain, 2002) and then the resulting parameters $\bar{\mathbf{X}}_k$, $\mathbf{\Sigma}_{R_k}$, and $\mathbf{Q}_k$ can be used as a prior model when we fit a PNDMM to a 2D shape from a single image in next section, *i.e.*, $\bar{\mathbf{X}}_k^{train} = \bar{\mathbf{X}}_k$, $\mathbf{\Sigma}_{R_k}^{train} = \mathbf{\Sigma}_{R_k}$, and $\mathbf{Q}_k^{train} = \mathbf{Q}_k$.

5.2. 3D Shape model fitting

Unlike the prior model learning step discussed in Section 5.1, observations for our problem is not a 3D shape $\mathbf{X}$, but a 2D shape $\mathbf{D} \in \mathbb{R}^{2\times n_p}$. We treat $\mathbf{X}$ as a hidden variable and estimate $\mathbf{X}$ from a 2D observation. Let $\widehat{\mathbf{D}} \in \mathbb{R}^{3\times n_p}$ be an observation matrix which is obtained from $\mathbf{D}$ by filling the first two rows using $\mathbf{D}$, after removing translations, and setting the third row (depth) to zeros. Then, we can regard $\widehat{\mathbf{D}}$ as a noisy observation sample from $\mathbf{X}$ (Cho et al., 2016; Lee et al., 2013):

$$\mathbf{vec}(\widehat{\mathbf{D}}) = \mathbf{F}\mathbf{vec}(\mathbf{X}) + \mathbf{u}, \tag{16}$$

where $\mathbf{vec}(\cdot)$ is the vectorization operator, $\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, and $\mathbf{F}$ is a matrix to remove translations and fills the z-coordinate, *i.e.*, the depth information, with zeros. (For more detail, see Lee et al. (2013).)

This fitting problem can be solved by using the EM algorithm based on the trained PNDMM, which is the same as the fitting problem in Cho et al. (2016) with the prior information of $(\bar{\mathbf{X}}_k, \mathbf{\Sigma}_{R_k})$. However, the contribution of a single data sample has little effects on calculating the PND model parameters ($\bar{\mathbf{X}}_k^{train}$, $\mathbf{\Sigma}_{R_k}^{train}$, $\mathbf{Q}_k^{train}$) in our case. Hence, we fix those parameters and only update s_k, $\mathbf{R}_k$, π_k, and σ^2 in the M-step, as done in Cho et al. (2016), which corresponds to Testing PNDMM in Fig. 2. After finishing EM iterations, the final posterior mean shape corresponding to the PND component with the highest weight w_k is used as a reconstructed 3D shape $\widehat{\mathbf{X}}$.

6. Model transformation

6.1. Model normalization

Given a novel image, the limb length of a subject may differ from the limb lengths of subjects in the training set. To handle this issue, we normalize the limb length information in the trained PNDMM model.

Let $l_{ij}^k = \|\bar{\mathbf{X}}_k(i) - \bar{\mathbf{X}}_k(i)\|_2$ be the limb length between part i and part j, where $\bar{\mathbf{X}}_k(i)$ is the ith column vector of $\bar{\mathbf{X}}_k$. We calculate mean lengths between body parts as $\bar{l}_{ij} = \frac{1}{K}\sum_k l_{ij}^k$ and adjust lengths l_{ij}^k between body parts to $\bar{l}_{ij}$ in all components of the trained PNDMM. The length adjustment process uses the following fact.

Proposition 1. *Let* $\mathbf{J}$ *be an* $n_p \times (n_p - 1)$ *full column rank matrix satisfying* $\mathbf{1}^T\mathbf{J} = \mathbf{0}$. *Then* $\mathbf{J}\mathbf{J}^+ = \mathbf{I} - \frac{1}{n_p}\mathbf{1}\mathbf{1}^T$.[2]

Proof.

$$\mathbf{J} = \mathbf{U}\mathbf{S}\mathbf{V}^T (\text{sinular value decomposition}). \tag{17}$$

$$\mathbf{J}\mathbf{J}^+ = \mathbf{U}\mathbf{S}\mathbf{V}^T\mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T = \mathbf{U}\mathbf{U}^T. \tag{18}$$

Since $\mathbf{1}^T\mathbf{J} = \mathbf{0}$, $\mathbf{1}$ is in the null space of $\mathbf{J}^T$. Then

$$\begin{bmatrix}\mathbf{U} & \frac{1}{\sqrt{n_p}}\mathbf{1}\end{bmatrix}^T\begin{bmatrix}\mathbf{U} & \frac{1}{\sqrt{n_p}}\mathbf{1}\end{bmatrix} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix} = \mathbf{I}, \tag{19}$$

[2] $\mathbf{J}^+$ is the psedudo-inverse of $\mathbf{J}$.

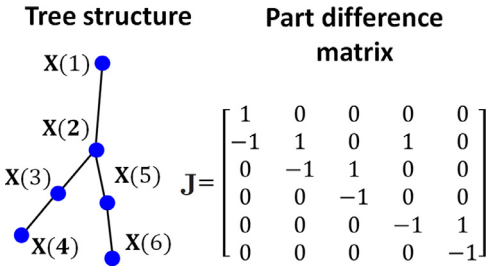


Fig. 3. An example of a tree structure and the corresponding part different matrix $\mathbf{J}$.

where $\mathbf{U}$ is a left-singular vector of $\mathbf{J}$. Therefore, $\begin{bmatrix}\mathbf{U} & \frac{1}{\sqrt{n_p}}\mathbf{1}\end{bmatrix}$ is a full rank orthogonal matrix. Then

$$\mathbf{I} = \begin{bmatrix}\mathbf{U} & \frac{1}{\sqrt{n_p}}\mathbf{1}\end{bmatrix}\begin{bmatrix}\mathbf{U} & \frac{1}{\sqrt{n_p}}\mathbf{1}\end{bmatrix}^T = \mathbf{U}\mathbf{U}^T + \frac{1}{n_p}\mathbf{1}\mathbf{1}^T.$$

Hence, we have $\mathbf{J}\mathbf{J}^+ = \mathbf{U}\mathbf{U}^T = \mathbf{I} - \frac{1}{n_p}\mathbf{1}\mathbf{1}^T$. □

Proposition 1 shows that $\mathbf{J}\mathbf{J}^+$ only removes a translation component of a given shape and preserves angles between body parts, thereby preserving the pose, *i.e.*, $\mathbf{X}\mathbf{J}\mathbf{J}^+ = \mathbf{X}(\mathbf{I} - \frac{1}{n_p}\mathbf{1}\mathbf{1}^T)$.

Let $\mathbf{J}$ be a part difference matrix defined by a tree structure, such that $\mathbf{1}^T\mathbf{J} = \mathbf{0}$ (an example is shown in Fig. 3). Then $\mathbf{J}$ is a full column rank matrix. Let $\mathbf{T}_d^k \in \mathbb{R}^{(n_p-1)\times(n_p-1)}$ be a diagonal matrix, where each diagonal entry is a length scale between $\bar{l}_{ij}$ and l_{ij}^k with appropriate indices i and j given by $\mathbf{J}$. Let $\mathbf{T}_s^k$ be a swapped matrix of $\mathbf{T}_d^k$ by swapping entries based on physically symmetric pairs of a human body (in Fig. 3, l_{23} and l_{25} can be swapped with l_{34} and l_{56}, respectively). Then we can obtain a physically symmetric diagonal matrix $\mathbf{T}^k = \frac{\mathbf{T}_d^k + \mathbf{T}_s^k}{2}$. Finally, the normalized mean shape for the kth PND is $\bar{\mathbf{X}}_k^{nor} = \bar{\mathbf{X}}_k^{train}(\mathbf{J}\mathbf{T}^k\mathbf{J}^+)$.

However, the new mean shape matrix $\bar{\mathbf{X}}_k^{nor}$ does not satisfy the scale constraint of the modified generalized Procrustes analysis of the PND (Lee et al., 2013). We therefore correct $\mathbf{T}^k$ by a scale factor $s_c^k = 1/\|\bar{\mathbf{X}}_k^{nor}\|_F$ and use $\mathbf{T}_c^k = s_c^k\mathbf{T}^k$ instead. Using the transformation matrix $\mathbf{T}_c^k$, we obtain the normalized mean shape $\bar{\mathbf{X}}_k^{nor} = \bar{\mathbf{X}}_k^{train}(\mathbf{J}\mathbf{T}_c^k\mathbf{J}^+)$ and the normalized covariance matrix for non-rigid variations $\mathbf{\Sigma}_{R_k}^{nor} = \mathbf{Q}_k^{norT}\left((\mathbf{J}\mathbf{T}_c^k\mathbf{J}^+)^T \otimes \mathbf{I}\right)\mathbf{\Sigma}_k^{train}\left((\mathbf{J}\mathbf{T}_c^k\mathbf{J}^+) \otimes \mathbf{I}\right)\mathbf{Q}_k^{nor}$, where $\mathbf{\Sigma}_k^{train} = \mathbf{Q}_k^{train}\mathbf{\Sigma}_{R_k}^{train}\mathbf{Q}_k^{trainT}$ and $\mathbf{Q}_k^{train}$ and $\mathbf{Q}_k^{nor}$ are calculated from $\bar{\mathbf{X}}_k^{train}$ and $\bar{\mathbf{X}}_k^{nor}$, respectively, according to the definition of the PND (Lee et al., 2013). The limb length adjustment is explained in Algorithm 1.

Algorithm 1 Limb length adjustment.

Require:
1: Part difference matrix $\mathbf{J}$
2: Transformation matrix $\mathbf{T}$
3: Model parameters: $\bar{\mathbf{X}}$, $\mathbf{\Sigma}_R$, and $\mathbf{Q}$

Ensure: Adjusted model parameters: $\bar{\mathbf{X}}'$, $\mathbf{\Sigma}_R'$, and $\mathbf{Q}'$
1: $\bar{\mathbf{X}}' = \bar{\mathbf{X}}(\mathbf{J}\mathbf{T}\mathbf{J}^+)$
2: $\mathbf{T}_c = s_c\mathbf{T}$, where $s_c = 1/\|\bar{\mathbf{X}}'\|_F$
3: $\bar{\mathbf{X}}' = \bar{\mathbf{X}}(\mathbf{J}\mathbf{T}_c\mathbf{J}^+)$ and calculate $\mathbf{Q}'$ using [21]
4: $\mathbf{J}' = (\mathbf{J}\mathbf{T}_c\mathbf{J}^+)^T \otimes \mathbf{I}$
5: $\mathbf{\Sigma}_R' = \mathbf{Q}'^T\mathbf{J}'\mathbf{Q}\mathbf{\Sigma}_R\mathbf{Q}^T\mathbf{J}'\mathbf{Q}'$

6.2. Model adaptation

While the model normalization step adjusts limb lengths among PNDMM components, we also need to adjust limb lengths when reconstructing from a novel test image to effectively handle the limb length difference problem. We address this issue using Algorithm 1 as a pre-processing step by adjusting model parameters $\overline{\mathbf{X}}_k^{nor}$, $\mathbf{\Sigma}_{R_k}^{nor}$, and $\mathbf{Q}_k^{nor}$ to a reconstructed 3D shape $\widehat{\mathbf{X}}$ obtained from the first iteration of the EM algorithm described in Section 5.2.

7. Result selection

Since we perform 3D reconstruction with $(2n_c - 1)$ 2D pose candidates obtained from Section 4, we need to select the best reconstruction result among them using three measures described below.

Score of a reprojected 2D shape (r_S): The 3D reconstruction algorithm includes a parameter σ to handle the observation noise, which allows a reprojected 2D shape to differ from an input 2D shape. (Dotted lines in the selection of Fig. 1 are reprojected 2D part locations.) To check whether a reconstructed 3D shape is explained by a 2D part detector, we calculate scores of reprojected 2D shapes obtained from reconstructed 3D shapes using (3) and the score is denoted as r_S. A higher score means that the reprojected 2D shape is well explained by the 2D part detector and the 3D model.

Normalized reprojection error (r_R): We use different datasets to train a 2D part detector and a 3D model, moreover, the test dataset can be different from the training datasets. Therefore, there can be a bias caused by differences in landmark locations among datasets. It can cause a large reprojection error, even if the 3D reconstruction is close to the ground truth. To address such problem, we propose a normalized reprojection error as follows:

$$r_R(\mathbf{D}, \mathbf{X}) = \frac{1}{n_p} \sum_i \sqrt{\mathbf{\Gamma}(i)^T \mathbf{P}_{orth}^T \mathbf{\Sigma}_{r_i}^{-1} \mathbf{P}_{orth} \mathbf{\Gamma}(i)}, \tag{20}$$

where i is a body part index, $\mathbf{\Gamma}(i) = \mathbf{D}(i) - \mathbf{P}_{orth}\mathbf{X}(i)$, $\mathbf{D}(i)$ is the ith column vector of $\mathbf{D}$, $\mathbf{X}(i)$ is the ith column vector of $\mathbf{X}$, $\mathbf{P}_{orth} = \left[\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right]$ is an orthographic projection matrix. That is, r_R corresponds to the mean of Mahalanobis distances calculated using 2D part reprojections. We align the body part positions in training data to body part positions detected in a test image, *i.e.*, $\mathbf{D}^{train}$ is aligned to $\mathbf{D}^{test}$. The aligned 2D shapes and its 3D reconstruction result are denoted as $\widetilde{\mathbf{D}}^{train}$ and $\widetilde{\mathbf{X}}^{train}$, respectively. As the error between $\widetilde{\mathbf{D}}^{train}$ and the projection of $\widetilde{\mathbf{X}}^{train}$ corresponds to an error caused by the bias, $(\widetilde{\mathbf{D}}^{train}(i) - \mathbf{P}_{orth}\widetilde{\mathbf{X}}^{train}(i))$ is used to compute the sample covariance matrix $\mathbf{\Sigma}_{r_i}$.

Model transformation error (r_T): We use a normalized 3D model by the mean limb lengths as discussed in Section 6.1 and it is adapted to a 2D shape. We therefore calculate the model transformation error, which is defined by the Mahalanobis distance between the mean transformation matrix $\overline{\mathbf{T}}_c = \frac{1}{K}\sum_k \mathbf{T}_c^k$ obtained from model normalization and a transformation matrix $\mathbf{T}_a$ obtained from the current 3D model adaptation process of Section 6.2. With the transformation matrix which is a diagonal matrix and each element is a length scale, the Mahalanobis distance between $\mathrm{diag}(\overline{\mathbf{T}}_c)$ and $\mathrm{diag}(\mathbf{T}_a)$ can be calculated, where $\mathrm{diag}(\mathbf{A})$ denotes a column vector consisting of diagonal elements of a matrix $\mathbf{A}$.

$$r_T(\mathbf{T}_a, \overline{\mathbf{T}}_c) = \sqrt{\mathrm{diag}(\mathbf{T}_a - \overline{\mathbf{T}}_c)^T \mathbf{\Sigma}_{\mathbf{T}_c}^{-1} \mathrm{diag}(\mathbf{T}_a - \overline{\mathbf{T}}_c)}, \tag{21}$$

where $\mathbf{\Sigma}_{\mathbf{T}_c}$ is the sample covariance of $\mathbf{T}_c^k$ from training samples.

Result selection: We again normalize each error between 0 and 1. The normalized errors are denoted as r'_S, r'_R, and r'_T and the final error can be represented as

$$error = (1 - r'_S) + r'_R + r'_T. \tag{22}$$

This normalized sum can be interpreted as error voting and we choose a candidate with the lowest error for the final 3D reconstruction.

If there are significant biases in part locations between training and testing sets, a weighted sum of the 2D shape and projected 3D shape can further improve the result:

$$\mathbf{D}^{new} = \eta\mathbf{D} + (1 - \eta)\mathbf{P}_{orth}\widehat{\mathbf{X}}, \tag{23}$$

where $0 \le \eta \le 1$ is a relative weight for emphasizing observation $\mathbf{D}$ and $\widehat{\mathbf{X}}$ is a 3D reconstruction result. With the new observation $\mathbf{D}^{new}$, we do 3D reconstruction described in Sections 5 and 6, again. The observation correction and 3D reconstruction can iteratively proceed. If we select a big value for η, the correction will be slow and the 3D reconstruction result becomes close to the observation.

8. Experiments

8.1. Implementation details

There are some implementation issues that need to be addressed. One issue is related to the part detection. Since the part detector from Yang and Ramanan (2013) was trained with twisted 2D shapes to capture appearance of poses consistently, *i.e.*, some left and right legs were swapped in the training set, it gives detections with twisted positions of legs. Such a detection is not suitable for accurate 3D reconstruction, so we detect twisted detections by comparing the angle θ_l between a left shoulder-neck and a left hip-neck with the angle θ_r between a left shoulder-neck and a right hip-neck. If $\theta_l > \theta_r$, we swap two leg positions.

Another issue is related to 3D reconstruction. A single test image allows w_k to have a small value, which causes proper PND components removed in early steps. If proper PND components are removed at the early stage of the EM iteration, the fitting process cannot give an accurate 3D reconstruction result. To solve this problem we perform the power normalization: $w_k = \frac{\sqrt{\pi_k p(\mathbf{D}|c_k=1,\Phi)}}{\sum_l \sqrt{\pi_l p(\mathbf{D}|c_l=1,\Phi)}}$. The weight does not affect PND parameters $\overline{\mathbf{X}}_k^{train}$ and $\mathbf{\Sigma}_{R_k}^{train}$ and the role of weight parameter w_k during the fitting process in Section 5.2 is to select a reconstruction result with the highest posterior.

The variance of the observation noise in Section 5.2 is initialized as $\sigma^2 = 10^{-6}$ and it is updated as part of the EM algorithm (Cho et al., 2016).

8.2. Evaluation of the 3D pose estimation

To check and analyze 3D reconstruction performances of the proposed method without the effect by inaccuracies of 2D part detections, we have performed experiments with known 2D landmark positions. In the CMU Mocap database,[3] we randomly selected a subset of 3D human poses from five different action categories by 23 subjects: *walking, jumping, running, boxing*, and *climbing* with 14 landmark points. The parameter n_l in (15) for a PNDMM was set to $2n_R$, where n_R is the dimension of a non-rigid shape space defined by a PND, hence, $2n_R$ is the minimum number of samples for a PND component is not singular. The number of PND mixtures in Section 5.1 was initialized at $K = 120$.

For the generalizability evaluation of the proposed method, we performed 23 rounds of experiments by selecting a subject as testing data and using the remaining subjects as training data. We excluded *climbing* from training and only used it for testing, since

[3] http://mocap.cs.cmu.edu/subjects.php.

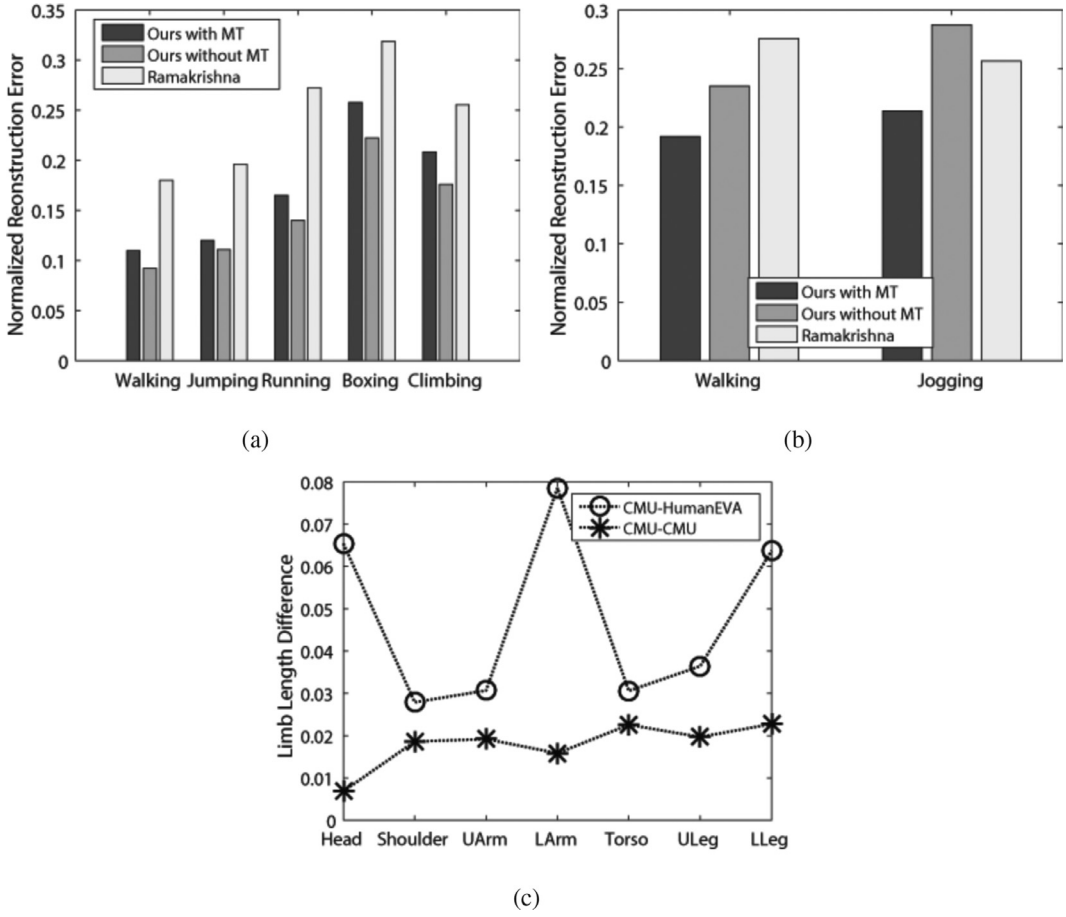


Fig. 4. (a) and (b) are the normalized reconstruction errors tested on CMU Mocap database and HumanEva dataset, respectively. 'MT' denotes model transformation. (c) shows the differences of mean limb lengths of 3D human poses between training and testing data. 'CMU-CMU' denotes both training and testing data come from the CMU Mocap database. 'CMU-HumanEVA' denotes training and testing data come from the CMU Mocap database and HumanEVA dataset, respectively. 'U' and 'L' in the x-axis label denote 'Upper' and 'Lower', respectively.

climbing action is performed by a single subject. During 23 rounds of evaluations, the average number of training samples is 15,613. For testing, we generated 2D landmark positions by orthogonally projecting 3D data into a 2D image plane with a random camera motion and reduced the frame rate to 20 frame per second (fps). We compared our algorithm with the algorithm developed by Ramakrishna et al. (2012), which was trained on the same training data. For the performance evaluation of 3D pose estimation, we performed the Procrustean alignment to the estimated 3D pose and the ground truth and then computed the average normalized reconstruction error:

$$e = \frac{1}{N_t} \sum_i \frac{\|\widehat{\mathbf{X}}_i - \mathbf{X}_i^*\|_F}{\|\mathbf{X}_i^*\|_F}, \tag{24}$$

where $\mathbf{X}_i^*$ and $\widehat{\mathbf{X}}_i$ are the ith ground truth and the reconstructed shape, respectively, and N_t is the number of tested images. In our implementation using MATLAB on a PC with Intel i7-4790K CPU, the average computation time for a sample in testing was 319.4 ms. Fig. 4(a) and (b) show that the proposed method outperforms (Ramakrishna et al., 2012). However, the effect of model transformation is different: in Fig. 4(a), model transformation makes a negative effect on 3D pose estimation while Fig. 4(b) shows that model transformation significantly decreases 3D pose estimation errors.

To investigate what makes two different results, we analyzed differences of the mean limb lengths of 3D human poses between training and testing data after scale normalization, *i.e.*, $\|\mathbf{X}\|_F = 1$. Fig. 4(c) shows differences of limb lengths of 3D human poses between training and testing data are significantly larger when the HumanEva dataset is used as test data. This may be due to the different landmark settings for different datasets when data were collected. We therefore conclude that model transformation is useful in cases where there are large biases between training and testing data, *i.e.*, the proposed model transformation method is useful in heterogeneous datasets with possibly different landmark settings.

We also investigated the effect of the hyper-parameter n_l, which is shown in Fig. 5. As shown in (15), n_l influences the required number of samples for a component not to be annihilated. In our experiments, the final numbers of components with increasing n_l were reduced from 100 to 39. Although n_l influences the final number of components and the final performance, we set n_l to its minimum possible value of $2n_R$ to avoid an additional parameter tuning step.

8.3. Effects of 2D candidate selection

In this section, we show the benefits of using a diverse set of candidates. We use the 2D part detector from Yang and Ramanan (2013), which was trained using the PARSE dataset (Ramanan, 2006). Using the 2D part detector, we generate four initial candidates ($n_c = 4$) from N-best extension and then three additional candidates were generated using recombination. The part detection results were converted from 26 part locations to 14 part locations. The 3D shape prior model was learned using the CMU Mocap dataset with 14 landmark points. We randomly selected five

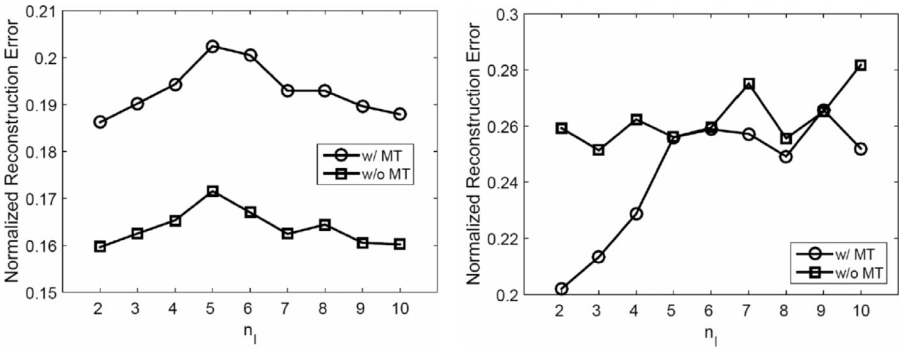


Fig. 5. Performance variations due to n_l. (a) When training and testing data are all from the CMU Mocap database. (b) When training data are from the CMU Mocap database and testing data are from the Human Eva dataset.

Table 1
2D part detection performance on the LSP Dataset (Johnson and Everingham, 2010).

Method	Torso	Head	ULeg	LLeg	UArm	LArm	Total
Yang (Yang and Ramanan, 2013)	82.9	79.1	61.9	53.2	46.0	29.8	54.4
Ours	83.1	79.5	62.5	54.6	46.5	31.5	55.3

frames from each sequence for learning the 3D shape model, similarly to Radwan et al. (2013), and the total number of samples for training a 3D model was 12,564. The number of PND mixtures in Section 5.1 was initialized to $K = 120$ and reduced to $K = 75$ after training. For 2D candidate selection, we calculated the sample covariance matrix of (20) using three nearest samples after alignment since samples far away are less predictive on the current observation.

We tested the trained joint 2D and 3D model on 1000 test samples of the LSP dataset (Johnson and Everingham, 2010) and evaluated the performance of 2D part detections obtained by the proposed 2D candidate selection. The performance measure is the percentage of correct parts (PCP), which is the standard evaluation metric (Eichner et al., 2009).[4] Table 1 shows the performance of 2D part detections selected by the proposed method is better than Yang and Ramanan (2013). We can conclude that the proposed 2D candidate selection improves the 2D part detection performance by considering 3D information.

8.4. Quantitative evaluation of the joint 2D and 3D pose estimation

The 2D part detector (Yang and Ramanan, 2013) was trained using the PARSE dataset (Ramanan, 2006) and the 3D shape model was trained using the CMU database as done in Section 8.3. The post iteration (23) was performed with $\eta = 0.9$ until the current reprojected score is less than the previous reprojected score and the minimum and maximum number of post iterations are 10 and 30, respectively.

The 3D human pose errors were evaluated using the mean error and its standard deviation in millimeter (mm) after the Procrustes alignment as done in Radwan et al. (2013) to compare the proposed method with part detection based methods (Radwan et al., 2013; Simo-Serra et al., 2012; Wang et al., 2014). Following the experiment setup in Radwan et al. (2013); Simo-Serra et al. (2012) and Wang et al. (2014), we evaluated our algorithm on the *walking* and *jogging* actions in the HumanEva dataset (Sigal et al., 2010) with the same sequences used in Radwan et al. (2013).

[4] In Table 1, Total is the ratio of the number of detected body parts to the total number of body parts (in percentage).

Table 2
Reconstruction errors (mm) with different settings (training set: CMU Mocap, test set: HumanEva).

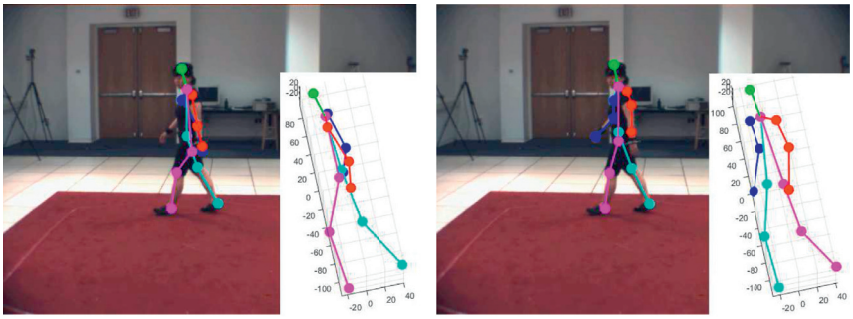
Walking	S1	S2	S3	All
Single w/ MT[a]	77.3 (20.1)	89.0 (25.1)	101.2 (30.5)	85.4 (24.6)
Single w/o MT[a]	74.9 (17.4)	84.3 (23.2)	105.3 (29.8)	82.3 (23.0)
N-best w/ MT[a]	71.3 (23.0)	85.7 (22.1)	89.8 (13.7)	80.5 (23.0)
N-best w/o MT[a]	74.5 (17.8)	83.8 (23.0)	88.2 (18.5)	80.6 (21.2)
Recomb. w/ MT[a]	70.0 (19.2)	84.7 (21.2)	88.2 (13.8)	79.3 (21.2)
Recomb. w/o MT[a]	73.8 (20.1)	83.0 (22.5)	85.4 (19.3)	79.7 (21.6)
Jogging	**S1**	**S2**	**S3**	**All**
Single w/ MT[a]	94.2 (24.8)	98.4 (27.6)	111.7 (30.9)	101.2 (28.5)
Single w/o MT[a]	101.5 (29.8)	102.0 (26.5)	118.1 (29.7)	106.8 (28.0)
N-best w/ MT[a]	99.2 (29.4)	94.9 (24.2)	103.3 (34.2)	98.8 (29.0)
N-best w/o MT[a]	97.8 (32.0)	94.0 (21.2)	111.4 (30.3)	100.5 (28.4)
Recomb. w/ MT[a]	98.8 (28.1)	92.1 (23.4)	103.2 (32.4)	97.5 (27.9)
Recomb. w/o MT[a]	100.0 (22.1)	97.5 (21.6)	114.4 (31.7)	103.4 (26.1)

[a] Without post iterations.

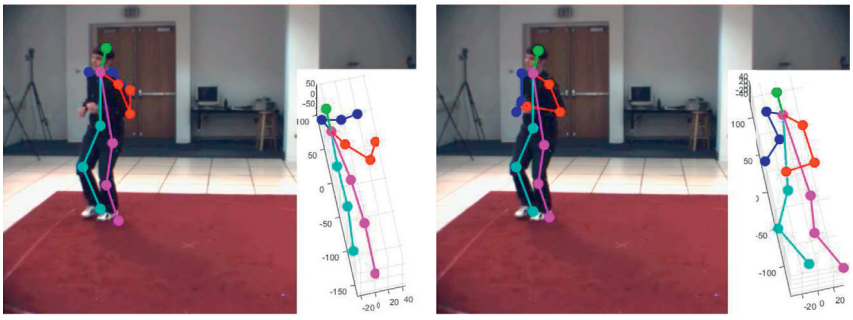
Unlike the pose estimated from a single 2D pose candidate, our algorithm select the result that is well explained by both 2D and 3D models and it reduces the effects of inaccuracies in 2D part detection results, thereby improving the performance. We show the performance as mean errors with standard deviations in parentheses in Table 2. In Table 2, '*N*-best' is the case when *N* candidates from the *N*-best algorithm (Park and Ramanan, 2011) are used and 'Recomb.' indicates the case which uses a combination of n_c best candidates from the *N*-best algorithm and $n_c - 1$ candidates from the recombination step of the proposed method. In our experiments, $N = 7$ and $n_c = 4$. 'S1', 'S2', and 'S3' indicate three subjects, respectively, and 'All' is the mean error of all subjects.

In this table, we can see that the reconstruction results based on multiple 2D candidates are better than those based on a single detection. Moreover, Table 2 shows that the results including recombined candidates are better than those using *N*-best algorithm alone. Fig. 6 shows examples of 3D pose estimation results obtained from the proposed method. The figure shows that we can better estimate the 3D pose from a single image by considering multiple 2D pose candidates with a good 3D shape model. Note that model transformation is useful when the training and testing datasets are different (see Fig. 7 for examples).

We investigated reasons why 3D human pose estimation errors of jogging sequences were bigger than those of walking sequences. We found that some frames of jogging action made severe occlusions on arms, and the incorrect joint locations deteriorate 3D human pose estimation as shown in Fig. 8. In cases with model transformation and post iterations in Table 3, the maximum errors of jogging sequences for S1, S2, and S3 were 151.7, 156.5, and 183.2,

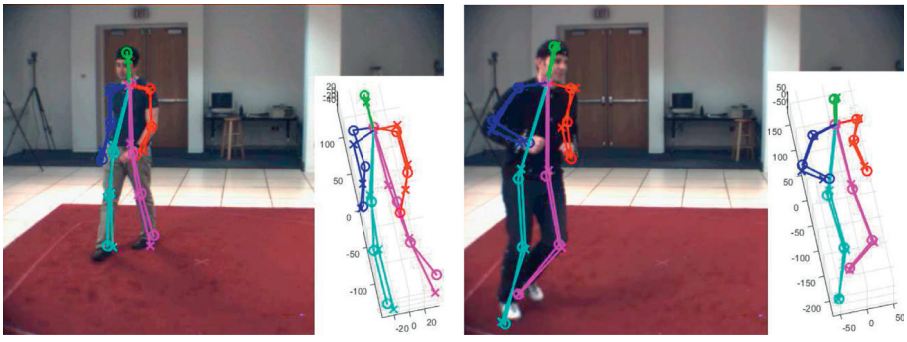


(a) Walking, S1, #400, Single (b) Walking, S1, #400, Proposed



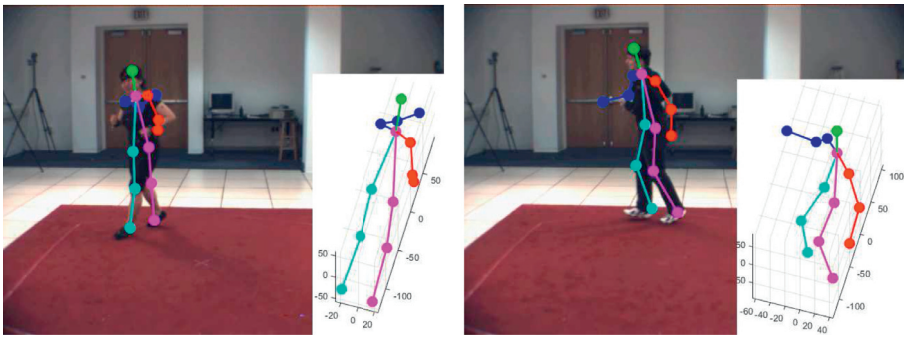
(c) Jogging, S3, #225, Single (d) Jogging, S3, #225, Proposed

Fig. 6. Examples of 3D pose estimation results from the Human Eva dataset. (a) and (c) are single detection based results without model transformation and post iteration. (b) and (c) are results from the proposed method using multiple candidates, model transformation, and post iterations. '#' denotes the frame number.



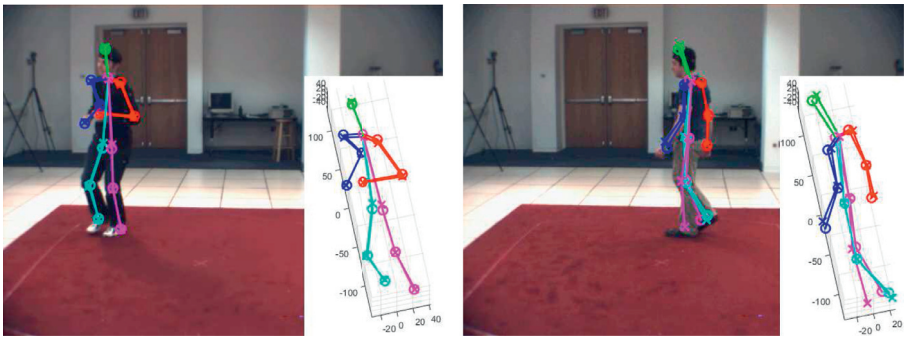
(a) Waking, S2, #120 (b) Jogging, S3, #290

Fig. 7. Examples where model transformation improves 3D pose estimation performances (when the training and testing datsets are different). Here, the 'O' and 'X' marks indicate a joint location estimated with model transformation and the one without model transformation, respectively.



(a) Jogging, S1, #155 (b) Jogging, S3, #185

Fig. 8. Failed examples of 3D human pose estimation from the jogging sequence. 3D Pose estimation errors of (a) and (b) are 151.7 and 183.2, respectively. A jogging action frequently makes arms occluded and the incorrect locations of arms deteriorate 3D human pose estimation.



(a) Waking, S2, #15 (b) Jogging, S3, #225

Fig. 9. Examples where model transformation increases 3D pose estimation errors (when the training and testing datsets are the same). Here, the 'O' and 'X' marks indicate a joint location estimated with model transformation and the one without model transformation, respectively.

Table 3
Reconstruction errors (mm) on the HumanEva dataset (training set: CMU Mocap).

Walking	S1	S2	S3	All
Ours (CMU) w/ MT[a]	71.7 (19.8)	80.7 (20.8)	87.8 (12.4)	77.8 (20.4)
Ours (CMU) w/o MT[a]	72.6 (21.3)	84.5 (21.3)	85.6 (19.7)	80.0 (20.7)
(Radwan et al., 2013)	75.1 (35.6)	99.8 (32.6)	93.8 (19.3)	89.8 (Est.)
Jogging	S1	S2	S3	All
Ours (CMU) w/ MT[a]	93.5 (27.2)	87.9 (22.2)	99.0 (30.4)	93.0 (26.6)
Ours (CMU) w/o MT[a]	99.6 (23.8)	97.8 (22.0)	115.2 (30.6)	103.7 (26.3)
(Radwan et al., 2013)	79.2 (26.4)	89.8 (34.2)	99.4 (35.1)	89.5 (Est.)

[a] With post iterations.

Table 4
Reconstruction errors (mm) on the HumanEva dataset (training set: HumanEva).

Walking	S1	S2	S3	All
Ours w/ MT (HumanEva)[a]	55.5 (27.2)	65.5 (30.9)	84.7 (18.9)	63.0 (29.6)
Ours w/o MT (HumanEva)[a]	42.4 (24.5)	56.9 (29.9)	72.7 (15.6)	52.6 (28.3)
Simo-Serra et al. (2012)	99.6 (42.6)	108.3 (42.3)	127.4 (24.0)	–
Wang et al. (2014)	71.9 (19.0)	75.7 (15.9)	85.3 (10.3)	–
Jogging	S1	S2	S3	All
Ours w/ MT (HumanEva)[a]	60.8 (26.8)	53.1 (24.5)	76.4 (35.5)	62.6 (30.2)
Ours w/o MT (HumanEva)[a]	59.3 (31.6)	57.3 (31.1)	61.3 (33.4)	59.1 (31.6)
Simo-Serra et al. (2012)	107.2 (41.5)	93.1 (41.1)	115.8 (40.6)	–
Wang et al. (2014)	62.6 (10.2)	77.7 (12.1)	54.4 (9.0)	–

[a] With post iterations.

respectively, while those of walking sequences for S1, S2, and S3 were 112.0, 138.6, and 109.4, respectively. Because 3D pose estimation results depends on the quality of 2D part detection results, we can conclude that the large errors of some frames significantly contributed to the average errors of jogging sequences.

We compared our results with the reported performance in Radwan et al. (2013); Simo-Serra et al. (2012) and Wang et al. (2014). We first compared our algorithm with Radwan et al. (2013) which have evaluated their algorithm on the HumanEva dataset (Sigal et al., 2010) with a model trained the CMU Mocap dataset. Table 3 shows that our algorithm outperforms (Radwan et al., 2013) in many cases.[5] Since the results of 'Recomb. w/ MT' in Table 2 are obtained without post iterations and the results of 'Ours (CMU)' in Table 3 are obtained with post iterations, the difference corresponds to the error reduced by post iterations.

Considering that Simo-Serra et al. (2013); 2012) and Wang et al. (2014) are trained from the HumanEva dataset and tested on the same HumanEva set, the learned 3D models could be biased toward subjects in the HumanEva dataset, resulting in smaller reconstruction errors. In Simo-Serra et al. (2013), input images were cropped based on the ground truth body joints and parameters were initialized using the ground truth body joints, which is an extremely strong assumption and, therefore, Simo-Serra et al. (2013) is not compared below. We tested the proposed method trained by the HumanEva dataset (Sigal et al., 2010) and compared with Simo-Serra et al. (2012) and Wang et al. (2014) in Table 4.[6] As in the experiments of Section 8.2, model transformation can give negative effects when there is no bias (see Fig. 9 for examples). Although this is a negative effect of model transformation, the reconstruction errors are significantly reduced regardless the effect of model transformation, compared with Table 3. Here, our algorithm shows excellent reconstruction results in most cases and outperforms compared methods (Simo-Serra et al., 2012; Wang et al., 2014). In Table 4, we have noticed that the standard deviation of Wang et al. (2014) is lower than the proposed method and Simo-Serra et al. (2012). We believe that this is due to the fact that Wang et al. (2014) uses an l_1-norm based objective function to fit a 3D model to 2D objects. It is known that the l_1-norm is robust against occlusion (Wright et al., 2009). It will be an interesting future work direction to consider an l_1-norm regularizer to make the proposed method more robust against self-occlusions.

9. Conclusions

We have proposed a method for estimating a 3D human pose from a single novel image. The problem is challenging due to inaccuracies of 2D part detectors and the complexity of human poses. To address these issues, we consider multiple 2D pose candidates with respect to a recently proposed 3D shape model, a Procrustean normal distribution mixture model (PNDMM). We have also introduced model transformation which is incorporated into the 3D shape prior model, such that the proposed method can be applied to a novel test image. Experimental results have shown that the proposed method can provide excellent 3D reconstruction results when tested on a novel test image, despite inaccuracies of 2D part detections and 3D shape ambiguities. Although the proposed method have used (Yang and Ramanan, 2013) for 2D part detection, a different 2D part detector can be utilized to improve its performance.

[5] Since we only obtained sequence numbers for tested images from the author of Radwan et al. (2013), we represented the performance by all subjects as estimated values.

[6] The sequence numbers for tested images are not clear in Simo-Serra et al. (2012); Wang et al. (2014), hence, we do not report reconstruction errors for all subjects in Table 4 for Simo-Serra et al. (2012) and Wang et al. (2014).

Acknowledgements

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2013R1A1A2065551, NRF-2015R1A2A1A15052493).

References

Agarwal, A., Triggs, B., 2004. 3D human pose from silhouettes by relevance vector regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2004.1315258.

Akhter, I., Black, M.J., 2015. Pose-conditioned joint angle limits for 3D human pose reconstruction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Bo, L., Sminchisescu, C., 2009. Structured output-associative regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2009.5206699.

Bo, L., Sminchisescu, C., 2010. Twin Gaussian processes for structured prediction. Int. J. Comput. Vision 87 (1–2), 28–52. doi:10.1007/s11263-008-0204-y.

Bregler, C., Hertzmann, A., Biermann, H., 2000. Recovering non-rigid 3D shape from image streams. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2000.854941.

Cho, J., Lee, M., Choi, C.-H., Oh, S., 2013. EM-GPA: generalized procrustes analysis with hidden variables for 3D shape modeling. Comput. Vision Image Understanding 117 (11), 1549–1559. doi:10.1016/j.cviu.2013.07.009.

Cho, J., Lee, M., Oh, S., 2016. Complex non-rigid 3d shape recovery using a procrustean normal distribution mixture model. Int. J. Comput. Vision 117 (3), 226–246. doi:10.1007/s11263-015-0860-7.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2005.177.

Eichner, M., Ferrari, V., Zurich, S., 2009. Better appearance models for pictorial structures. In: Proceedings of the British Machine Vision Conference doi:10.5244/C.23.3.

Fan, X., Zheng, K., Zhou, Y., Wang, S., 2014. Pose locality constrained representation for 3D human pose reconstruction. In: Proceedings of the European Conference on Computer Vision doi:10.1007/978-3-319-10590-1_12.

Figueiredo, M.A.T., Jain, A.K., 2002. Unsupervised learning of finite mixture models. IEEE Trans. Pattern. Anal. Mach. Intell. 24 (3), 381–396. doi:10.1109/34.990138.

Gotardo, P.F., Martinez, A.M., 2011. Non-rigid structure from motion with complementary rank-3 spaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/TPAMI.2007.70752.

Ionescu, C., Carreira, J., Sminchisescu, C., 2014. Iterated second-order label sensitive pooling for 3D human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2014.215.

Johnson, S., Everingham, M., 2010. Clustered pose and nonlinear appearance models for human pose estimation. In: Proceedings of the British Machine Vision Conference doi:10.5244/C.24.12.

Kostrikov, I., Gall, J., 2014. Depth sweep regression forests for estimating 3D human pose from images. In: Proceedings of the British Machine Vision Conference.

Lee, M., Cho, J., Choi, C.-H., Oh, S., 2013. Procrustean normal distribution for non-rigid structure from motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2013.169.

Lehrmann, A., Gehler, P., Nowozin, S., 2013. A non-parametric Bayesian network prior of human pose. In: Proceedings of the IEEE International Conference on Computer Vision.

Li, S., Chan, A.B., 2014. 3D human pose estimation from monocular images with deep convolutional neural network. In: Proceedings of the Asian Conference on Computer Vision.

Paladini, M., Bue, A.D., Stošic, M., Dodig, M., Xavier, J., Agapito, L., 2009. Factorization for non-rigid and articulated structure using metric projections. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2009.5206602.

Park, D., Ramanan, D., 2011. N-best maximal decoders for part models. In: Proceedings of the IEEE International Conference on Computer Vision doi:10.1109/ICCV.2011.6126552.

Pons-Moll, G., Fleet, D., Rosenhahn, B., 2014. Posebits for monocular human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Radwan, I., Dhall, A., Goecke, R., 2013. Monocular image 3D human pose estimation under self-occlusion. In: Proceedings of the IEEE International Conference on Computer Vision doi:10.1109/ICCV.2013.237.

Ramakrishna, V., Kanade, T., Sheikh, Y., 2012. Reconstructing 3D human pose from 2D image landmarks. In: Proceedings of the European Conference on Computer Vision doi:10.1007/978-3-642-33765-9_41.

Ramanan, D., 2006. Learning to parse images of articulated bodies. In: Proceedings of the Advances in Neural Information Processing Systems.

Sigal, L., Balan, A., Black, M.J., 2007. Combined discriminative and generative articulated pose and non-rigid shape estimation. In: Proceedings of the Advances in Neural Information Processing Systems.

Sigal, L., Balan, A.O., Black, M.J., 2010. Humaneva: synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. Int. J. Comput. Vision 87 (1–2), 4–27. doi:10.1007/s11263-009-0273-6.

Simo-Serra, E., Quattoni, A., Torras, C., Moreno-Noguer, F., 2013. A joint model for 2D and 3D pose estimation from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2013.466.

Simo-Serra, E., Ramisa, A., Alenya, G., Torras, C., Moreno-Noguer, F., 2012. Single image 3D human pose estimation from noisy observations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2012.6247988.

Simo-Serra, E., Torras, C., Moreno-Noguer, F., 2014. Geodesic finite mixture models. In: Proceedings of the British Machine Vision Conference.

Torresani, L., Hertzmann, A., Bregler, C., 2008. Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors. IEEE Trans. Pattern Anal. Mach. Intell. 30, 878–892. doi:10.1109/TPAMI.2007.70752.

Wang, C., Wang, Y., Lin, Z., Yuille, A.L., Gao, W., 2014. Robust estimation of 3D human poses from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2014.303.

Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y., 2009. Robust face recognition via sparse representation. IEEE Trans. Pattern Anal. Mach. Intell. 30, 201–227. doi:10.1109/TPAMI.2008.79.

Xiao, J., Chai, J., Kanade, T., 2006. A closed-form solution to non-rigid shape and motion recovery. Int. J. Comput. Vision 67, 233–246. doi:10.1007/978-3-540-24673-2_46.

Yang, Y., Ramanan, D., 2013. Articulated human detection with flexible mixtures of parts. IEEE Trans. Pattern Anal. Mach. Intell. 35 (12), 2878–2890. doi:10.1109/TPAMI.2012.261.

Zhou, X.H.K.D.X., Leonardos, S., 2015. 3D shape estimation from 2D landmarks: A convex relaxation approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition doi:10.1109/CVPR.2015.7299074.

Jungchan Cho is a Ph.D. student in the Department of Electrical and Computer Engineering at the Seoul National University, Seoul, Korea. His research interests include pattern recognition and computer vision. He received the B.S. degree in the School of Electrical and Electronics Engineering from Chung-Ang University, Seoul, Korea, in 2010.

Minsik Lee received the B.S. and Ph.D. degrees from the School of Electrical Engineering and Computer Science, Seoul National University, Korea, in 2006 and 2012, respectively. During 2012 and 2013, he was a postdoctoral researcher in the same school. In 2014, He joined Seoul National University as a BK21 Assistant Professor. Currently, he is an assistant professor at Hanyang University, Ansan, Korea. His research interests include shape and motion analysis, deformable models, computer vision, pattern analysis, and their applications.

Songhwai Oh received the B.S. (Hons.), M.S., and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, CA, USA, in 1995, 2003, and 2006, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Seoul National University, Seoul, Korea. Before his Ph.D. studies, he was a Senior Software Engineer at Synopsys, Inc., Mountain View, CA, USA, and a Microprocessor Design Engineer at Intel Corporation, Santa Clara, CA, USA. In 2007, he was a Post-Doctoral Researcher with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, USA. From 2007 to 2009, he was an Assistant Professor of Electrical Engineering and Computer Science in the School of Engineering, University of California, Merced, CA, USA. His current research interests include cyberphysical systems, robotics, computer vision, and machine learning.