Sparseness Meets Deepness: 3D Human Pose Estimation from Monocular Video

Xiaowei Zhou[†,*], Menglong Zhu[†,*], Spyridon Leonardos[†], Konstantinos G. Derpanis[‡], Kostas Daniilidis[†]
[†] University of Pennsylvania [‡] Ryerson University

Abstract

This paper addresses the challenge of 3D full-body human pose estimation from a monocular image sequence. Here, two cases are considered: (i) the image locations of the human joints are provided and (ii) the image locations of joints are unknown. In the former case, a novel approach is introduced that integrates a sparsity-driven 3D geometric prior and temporal smoothness. In the latter case, the former case is extended by treating the image locations of the joints as latent variables to take into account considerable uncertainties in 2D joint locations. A deep fully convolutional network is trained to predict the uncertainty maps of the 2D joint locations. The 3D pose estimates are realized via an Expectation-Maximization algorithm over the entire sequence, where it is shown that the 2D joint location uncertainties can be conveniently marginalized out during inference. Empirical evaluation on the Human3.6M dataset shows that the proposed approaches achieve greater 3D pose estimation accuracy over state-of-the-art baselines. Further, the proposed approach outperforms a publicly available 2D pose estimation baseline on the challenging PennAction dataset.

1. Introduction

This paper is concerned with the challenge of recovering the 3D full-body human pose from a monocular RGB image sequence. Potential applications of the presented research include human-computer interaction (cf. [37]), surveillance, video browsing and indexing, and virtual reality.

From a geometric perspective, 3D articulated pose recovery is inherently ambiguous from monocular imagery [20]. Further difficulties are raised due to the large variation in human appearance (e.g., clothing, body shape, and illumination), arbitrary camera viewpoint, and obstructed visibility due to external entities and self-occlusions. Notable successes in pose estimation consider the challenge of 2D pose recovery using discriminatively trained 2D part models coupled with 2D deformation priors, e.g., [50, 4, 49], and more recently using deep learning, e.g., [46]. Here, the 3D pose geometry is not leveraged. Combining robust image-driven 2D part detectors, expressive 3D geometric pose priors and temporal models to aggregate information over time is a promising area of research that has been given limited attention, e.g., [5, 54]. The challenge posed is how to seamlessly integrate 2D, 3D and temporal information to fully account for the model and measurement uncertainties.

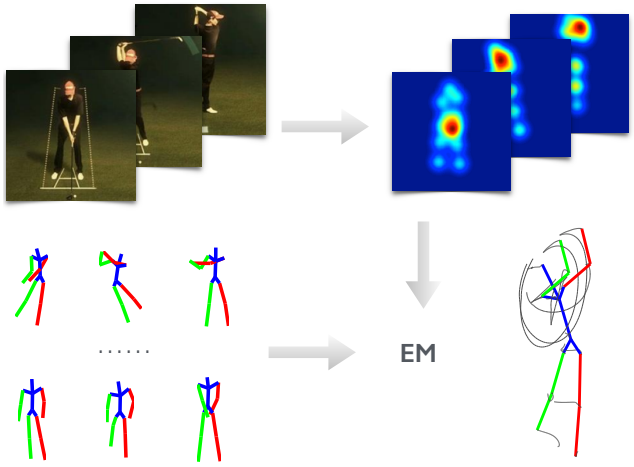

Figure 1. Overview of the proposed approach. (top-left) Input image sequence, (top-right) CNN-based heat map outputs representing the soft localization of 2D joints, (bottom-left) 3D pose dictionary, and (bottom-right) the recovered 3D pose sequence reconstruction.

This paper presents a 3D pose recovery framework that consists of a novel synthesis between discriminative image-based and 3D reconstruction approaches. In particular, the approach reasons jointly about image-based 2D part location estimates and model-based 3D pose reconstruction, so that they can benefit from each other. Further, to improve the approach's robustness against detector error, occlusion, and reconstruction ambiguity, temporal smoothness is imposed on the 3D pose and viewpoint parameters. Figure 1 provides an overview of the proposed approach. Given the input video (Fig. 1, top-left), 2D joint heat maps are generated with a deep convolutional neural network (CNN) (Fig. 1, top-right). These heat maps are combined with a sparse model of 3D human pose (Fig. 1, bottom-left) within an Expectation-Maximization (EM) framework to recover the 3D pose sequence (Fig. 1, bottom-right).

*The first two authors contributed equally to this work.

Considerable research has addressed the challenge of human motion capture from imagery [26, 41, 9, 33]. This work includes 2D human pose recovery in both single images (e.g., [50, 46, 10, 17, 45]) and video, e.g., [35, 11, 49, 29, 31, 52]. In the current work, focus is placed on 3D pose recovery in video, where the pose model and prior are expressed in their natural 3D domain.

Early research on 3D monocular pose estimation in videos largely centred on incremental frame-to-frame pose tracking, e.g., [8, 42, 38]. These approaches rely on a given pose and dynamic model to constrain the pose search space. Notable drawbacks of this approach include: the requirement that the initialization be provided and their inability to recover from tracking failures. To address these limitations, more recent approaches have cast the tracking problem as one of data association across frames, i.e., "tracking-by-detection", e.g., [5]. Here, candidate poses are first detected in each frame and subsequently a linking process attempts to establish temporally consistent poses.

Another strand of research has focused on methods that predict 3D poses by searching a database of exemplars [36, 27, 19] or via a discriminatively learned mapping from the image directly or image features to human joint locations [1, 34, 51, 16, 44]. Recently, deep convolutional networks (CNNs) have emerged as a common element behind many state-of-the-art approaches, including human pose estimation, e.g., [46, 22, 45, 23]. Here, two general approaches can be distinguished. The first approach casts the pose estimation task as a joint location regression problem from the input image [46, 22, 23]. The second approach uses a CNN architecture for body part detection [10, 17, 45, 31] and then typically enforces the 2D spatial relationship between body parts as a subsequent processing step. Similar to the latter approaches, the proposed approach uses a CNN-based architecture to regress confidence heat maps of 2D joint position predictions. The current work departs from these approaches by enforcing 3D spatial part relationships rather than 2D ones.

Most closely related to the present paper are generic factorization approaches for recovering 3D non-rigid shapes from image sequences captured with a single camera [7, 3, 14, 57, 12], i.e., non-rigid structure from motion (NRSFM), and human pose recovery models based on known skeletons [20, 43, 47, 30, 21] or sparse representations [32, 15, 2, 55, 56]. Much of this work has been realized by assuming manually labeled 2D joint locations; however, there is some recent work that has used a 2D pose detector to automatically provide the input joints [40, 48] or solved 2D and 3D pose estimation jointly [39, 54].

Contributions: The proposed approach advances the state-of-the-art in the following three ways. First, in contrast to prediction methods (e.g., [16, 23]), the proposed approach does not require synchronized 2D-3D data, as captured by motion capture systems. The proposed approach only requires readily available annotated 2D imagery (e.g., the "in-the-wild" PennAction dataset [53]) to train a CNN part detector and a separate 3D motion capture dataset (e.g., the CMU MoCap database) for the pose dictionary. Second, in comparison to other 3D reconstruction methods (e.g., [32, 2]), the proposed approach considers an arbitrary pose uncertainty. Finally, in contrast to prior work that consider two disjoint steps (i.e., detection of 2D joints and subsequent lifting the detections to 3D), the current approach combines these steps by casting the 2D joint locations as latent variables. This allows us to leverage the 3D geometric prior to help 2D joint localization and to rigorously handle the 2D estimation uncertainty in a statistical framework.

2. Models

In this section, the models that describe the relationships between 3D poses, 2D poses and images are introduced.

2.1. Sparse representation of 3D poses

The 3D human pose is represented by the 3D locations of a set of p joints, which is denoted by $\boldsymbol{S}_t \in \mathbb{R}^{3\times p}$ for frame t. To reduce the ambiguity for 3D reconstruction, it is assumed that a 3D pose can be represented as a linear combination of predefined basis poses:

$$\boldsymbol{S}_t = \sum_{i=1}^{k} c_{it}\boldsymbol{B}_i, \tag{1}$$

where $\boldsymbol{B}_i \in \mathbb{R}^{3\times p}$ denotes a basis pose and c_{it} the corresponding weight. The basis poses are learned from training poses provided by a motion capture (MoCap) dataset. Instead of using the conventional active shape model [13], where the basis set is small, a sparse representation is adopted which has proven in recent work to be capable of modelling the large variability of human pose, e.g., [32, 2, 55]. That is, an overcomplete dictionary, $\{\boldsymbol{B}_1, \cdots, \boldsymbol{B}_k\}$, is learned with a relatively large number of basis poses, k, where the coefficients, c_{it}, are assumed to be sparse. In the remainder of this paper, $\boldsymbol{c}_t$ denotes the coefficient vector $[c_{1t}, \cdots, c_{kt}]^\top$ for frame t and $\boldsymbol{C}$ denotes the matrix composed of all $\boldsymbol{c}_t$.

2.2. Dependence between 2D and 3D poses

The dependence between a 3D pose and its imaged 2D pose is modelled with a weak perspective camera model:

$$\boldsymbol{W}_t = \boldsymbol{R}_t\boldsymbol{S}_t + \boldsymbol{T}_t\boldsymbol{1}^\top, \tag{2}$$

where $\boldsymbol{W}_t \in \mathbb{R}^{2\times p}$ denotes the 2D pose in frame t, and $\boldsymbol{R}_t \in \mathbb{R}^{2\times 3}$ and $\boldsymbol{T}_t \in \mathbb{R}^2$ the camera rotation and translation, respectively. Note, the scale parameter in the weak perspective model is removed because the 3D structure, $\boldsymbol{S}_t$,

can itself be scaled. In the following, $\boldsymbol{W}$, $\boldsymbol{R}$ and $\boldsymbol{T}$ denote the collections of $\boldsymbol{W}_t$, $\boldsymbol{R}_t$ and $\boldsymbol{T}_t$ for all t, respectively.

Considering the observation noise and model error, the conditional distribution of the 2D poses given the 3D pose parameters is modelled as

$$\Pr(\boldsymbol{W}|\theta) \propto e^{-\mathcal{L}(\theta;\boldsymbol{W})}, \tag{3}$$

where $\theta = \{\boldsymbol{C}, \boldsymbol{R}, \boldsymbol{T}\}$ is the union of all the 3D pose parameters and the loss function, $\mathcal{L}(\theta; \boldsymbol{W})$, is defined as

$$\mathcal{L}(\theta; \boldsymbol{W}) = \frac{\nu}{2} \sum_{t=1}^{n} \left\| \boldsymbol{W}_t - \boldsymbol{R}_t \sum_{i=1}^{k} c_{it} \boldsymbol{B}_i - \boldsymbol{T}_t \mathbf{1}^\top \right\|_F^2, \tag{4}$$

with $\|\cdot\|_F$ denoting the Frobenius norm. The model in (3) states that, given the 3D poses and camera parameters, the 2D location of each joint belongs to a Gaussian distribution with a mean equal to the projection of its 3D counterpart and a precision (i.e., the inverse variance) equal to ν.

2.3. Dependence between pose and image

When 2D poses are given, it is assumed that the distribution of 3D pose parameters is conditionally independent of the image data. Therefore, the likelihood function of θ can be factorized as

$$\Pr(\boldsymbol{I}, \boldsymbol{W}|\theta) = \Pr(\boldsymbol{I}|\boldsymbol{W}) \Pr(\boldsymbol{W}|\theta), \tag{5}$$

where $\boldsymbol{I} = \{\boldsymbol{I}_1, \cdots, \boldsymbol{I}_n\}$ denotes the input images and $\Pr(\boldsymbol{W}|\theta)$ is given in (3). $\Pr(\boldsymbol{I}|\boldsymbol{W})$ is difficult to directly model, but it is proportional to $\Pr(\boldsymbol{W}|\boldsymbol{I})$ by assuming uniform priors on $\boldsymbol{W}$ and $\boldsymbol{I}$, and $\Pr(\boldsymbol{W}|\boldsymbol{I})$ can be learned from data.

Given the image data, the 2D distribution of each joint is assumed to be only dependent on the current image. Thus,

$$\Pr(\boldsymbol{I}|\boldsymbol{W}) \propto \Pr(\boldsymbol{W}|\boldsymbol{I}) = \Pi_t \Pi_j h_j(\boldsymbol{w}_{jt}; \boldsymbol{I}_t), \tag{6}$$

where $\boldsymbol{w}_{jt}$ denotes the image location of joint j in frame t, and $h_j(\cdot; \boldsymbol{Y})$ represents a mapping from an image $\boldsymbol{Y}$ to a probability distribution of the joint location (termed heat map). For each joint j, the mapping h_j is approximated by a CNN learned from training data. The details of CNN learning are described in Section 4.

2.4. Prior on model parameters

The following penalty function on the model parameters is introduced:

$$\mathcal{R}(\theta) = \alpha \|\boldsymbol{C}\|_1 + \frac{\beta}{2} \|\nabla_t \boldsymbol{C}\|_F^2 + \frac{\gamma}{2} \|\nabla_t \boldsymbol{R}\|_F^2, \tag{7}$$

where $\|\cdot\|_1$ denotes the ℓ_1-norm (i.e., the sum of absolute values), and ∇_t the discrete temporal derivative operator. The first term penalizes the cardinality of the pose coefficients to induce a sparse pose representation. The second and third terms impose first-order smoothness on both the pose coefficients and rotations.

3. 3D pose inference

In this section, the proposed approach to 3D pose inference is described. Here, two cases are distinguished: (i) the image locations of the joints are provided (Section 3.1) and (ii) the joint locations are unknown (Section 3.2).

3.1. Given 2D poses

When the 2D poses, $\boldsymbol{W}$, are given, the model parameters, θ, are recovered via penalized maximum likelihood estimation (MLE):

$$\begin{aligned} \theta^* &= \operatorname*{argmax}_{\theta} \ \ln \Pr(\boldsymbol{W}|\theta) - \mathcal{R}(\theta) \\ &= \operatorname*{argmin}_{\theta} \ \mathcal{L}(\theta; \boldsymbol{W}) + \mathcal{R}(\theta). \end{aligned} \tag{8}$$

The problem in (8) is solved via block coordinate descent, i.e., alternately updating $\boldsymbol{C}$, $\boldsymbol{R}$ or $\boldsymbol{T}$ while fixing the others. The update of $\boldsymbol{C}$ needs to solve:

$$\boldsymbol{C} \leftarrow \operatorname*{argmin}_{\boldsymbol{C}} \ \mathcal{L}(\boldsymbol{C}; \boldsymbol{W}) + \alpha \|\boldsymbol{C}\|_1 + \frac{\beta}{2} \|\nabla_t \boldsymbol{C}\|_F^2, \tag{9}$$

where the objective is the composite of two differentiable functions plus an ℓ_1 penalty. The problem in (9) is solved by accelerated proximal gradient (APG) [28]. Since the problem in (9) is convex, global optimality is guaranteed. The update of $\boldsymbol{R}$ needs to solve:

$$\boldsymbol{R} \leftarrow \operatorname*{argmin}_{\boldsymbol{R}} \ \mathcal{L}(\boldsymbol{R}; \boldsymbol{W}) + \frac{\gamma}{2} \|\nabla_t \boldsymbol{R}\|_F^2, \tag{10}$$

where the objective is differentiable and the variables are rotations restricted to $SO(3)$. Here, manifold optimization is adopted to update the rotations using the trust-region solver in the Manopt toolbox [6]. The update of $\boldsymbol{T}$ has the following closed-form solution:

$$\boldsymbol{T}_t \leftarrow \text{row mean} \left\{ \boldsymbol{W}_t - \boldsymbol{R}_t \sum_{i=1}^{k} c_{it} \boldsymbol{B}_i \right\}. \tag{11}$$

The entire algorithm for 3D pose inference given the 2D poses is summarized in Algorithm 1. The iterations are terminated once the objective value has converged. Since in each step the objective function is non-increasing, the algorithm is guaranteed to converge; however, since the problem in (8) is nonconvex, the algorithm requires a suitably chosen initialization (described in Section 3.3).

3.2. Unknown 2D poses

If the 2D poses are unknown, $\boldsymbol{W}$ is treated as a latent variable and is marginalized during the estimation process. The marginalized likelihood function is

$$\Pr(\boldsymbol{I}|\theta) = \int \Pr(\boldsymbol{I}, \boldsymbol{W}|\theta) d\boldsymbol{W}, \tag{12}$$

Algorithm 1: Block coordinate descent to solve (8).

Input: $\boldsymbol{W}$; // 2D joint locations
Output: $\boldsymbol{C}, \boldsymbol{R}, \boldsymbol{T}$; // pose parameters

initialize the parameters ; // Section 3.3
while *not converged* **do**
  update $\boldsymbol{C}$ by (9) with APG;
  update $\boldsymbol{R}$ by (10) with Manopt;
  update $\boldsymbol{T}$ by (11);
end

where $\Pr(\boldsymbol{I}, \boldsymbol{W}|\theta)$ is given in (5).

Direct marginalization of (12) is extremely difficult. Instead, an EM algorithm is developed to compute the penalized MLE. In the expectation step, the expectation of the penalized log-likelihood is calculated with respect to the conditional distribution of $\boldsymbol{W}$ given the image data and the previous estimate of all the 3D pose parameters, θ':

$$\begin{aligned} Q(\theta|\theta') &= \int \{\ln \Pr(\boldsymbol{I}, \boldsymbol{W}|\theta) - \mathcal{R}(\theta)\} \Pr(\boldsymbol{W}|\boldsymbol{I}, \theta') d\boldsymbol{W} \\ &= \int \{\ln \Pr(\boldsymbol{I}|\boldsymbol{W}) + \ln \Pr(\boldsymbol{W}|\theta) - \mathcal{R}(\theta)\} \Pr(\boldsymbol{W}|\boldsymbol{I}, \theta') d\boldsymbol{W} \\ &= \text{const} - \int \mathcal{L}(\theta; \boldsymbol{W}) \Pr(\boldsymbol{W}|\boldsymbol{I}, \theta') d\boldsymbol{W} - \mathcal{R}(\theta). \end{aligned} \tag{13}$$

It can be easily shown that

$$\int \mathcal{L}(\theta; \boldsymbol{W}) \Pr(\boldsymbol{W}|\boldsymbol{I}, \theta') d\boldsymbol{W} = \mathcal{L}(\theta; \mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta']) + \text{const}, \tag{14}$$

where $\mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta']$ is the expectation of $\boldsymbol{W}$ given $\boldsymbol{I}$ and θ':

$$\begin{aligned} \mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta'] &= \int \Pr(\boldsymbol{W}|\boldsymbol{I}, \theta') \, \boldsymbol{W} \, d\boldsymbol{W} \\ &= \int \frac{\Pr(\boldsymbol{I}|\boldsymbol{W}) \Pr(\boldsymbol{W}|\theta')}{Z} \, \boldsymbol{W} \, d\boldsymbol{W}, \end{aligned} \tag{15}$$

and Z is a scalar that normalizes the probability. The derivation of (14) and (15) is given in the supplementary material. Both $\Pr(\boldsymbol{I}|\boldsymbol{W})$ and $\Pr(\boldsymbol{W}|\theta')$ given in (6) and (3), respectively, are products of marginal probabilities of $\boldsymbol{w}_{jt}$. Therefore, the expectation of each $\boldsymbol{w}_{jt}$ can be computed separately. In particular, the expectation of each $\boldsymbol{w}_{jt}$ is efficiently approximated by sampling over the pixel grid.

In the maximization step, the following is computed:

$$\begin{aligned} \theta &\leftarrow \operatorname*{argmax}_{\theta} Q(\theta|\theta') \\ &= \operatorname*{argmin}_{\theta} \; \mathcal{L}(\theta; \mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta']) + \mathcal{R}(\theta), \end{aligned} \tag{16}$$

which can be solved by Algorithm 1.

The entire EM algorithm is summarized in Algorithm 2 with the initialization scheme described next in Section 3.3.

Algorithm 2: The EM algorithm for pose from video.

Input: $h_j(\cdot; \boldsymbol{I}_t), \forall j, t$; // heat maps
Output: $\theta = \{\boldsymbol{C}, \boldsymbol{R}, \boldsymbol{T}\}$; // pose parameters

initialize the parameters ; // Section 3.3
while *not converged* **do**
  $\theta' = \theta$;
  // Compute the expectation of $\boldsymbol{W}$
  $\mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta'] = \int \frac{1}{Z} \Pr(\boldsymbol{I}|\boldsymbol{W}) \Pr(\boldsymbol{W}|\theta') \, \boldsymbol{W} \, d\boldsymbol{W}$;
  // Update θ by Algorithm 1
  $\theta = \operatorname{argmin}_{\theta} \; \mathcal{L}(\theta; \mathbb{E}[\boldsymbol{W}|\boldsymbol{I}, \theta']) + \mathcal{R}(\theta)$;
end

3.3. Initialization

A convex relaxation approach [55, 56] is used to initialize the parameters. In [55], a convex formulation was proposed to solve the single frame pose estimation problem given 2D correspondences, which is a special case of (8). The approach was later extended to handle 2D correspondence outliers [56]. If the 2D poses are given, the model parameters are initialized for each frame separately with the convex method proposed in [55]. Alternatively, if the 2D poses are unknown, for each joint, the image location with the maximum heat map value is used. Next, the robust estimation algorithm from [56] is applied to initialize the parameters.

4. CNN-based joint uncertainty regression

A CNN is used to learn the mapping $\boldsymbol{Y} \mapsto h_j(\cdot; \boldsymbol{Y})$, where $\boldsymbol{Y}$ denotes an input image and $h_j(\cdot; \boldsymbol{Y})$ represents a heat map for joint j. Instead of learning p networks for p joints, a fully convolutional neural network [24] is trained to regress p joint distributions simultaneously by taking into account the full-body information.

During training, a rectangular patch is extracted around the subject from each image and is resized to 256×256 pixels. Random shifts are applied during cropping and RGB channel-wise random noise is added for data augmentation. Channel-wise RGB mean values are computed from the dataset and subtracted from the images for data normalization. The training labels to be regressed are multi-channel heat maps with each channel corresponding to the image location uncertainty distribution for each joint. The uncertainty is modelled by a Gaussian centered at the annotated joint location with variance $\sigma = 1.5$. The heat map resolution is reduced to 32×32 to decrease the CNN model size which allows a large batch size in training and prevents overfitting.

The CNN architecture used is similar to the SpatialNet model proposed elsewhere [31] but without any spatial fu-

sion or temporal pooling. The network consists of seven convolutional layers with 5×5 filters followed by ReLU layers and a last convolutional layer with $1 \times 1 \times p$ filters to provide dense prediction for all joints. A 2×2 max pooling layer is inserted after each of the first three convolutional layers. The network is trained by minimizing the l_2 loss between the prediction and the label with the open source Caffe framework [18]. Stochastic gradient descent (SGD) with momentum of 0.9 and a mini-batch size of 128 is used.

During testing, consistent with previous 3D pose methods (e.g., [23, 44]), a bounding box around the subject is assumed and the image patch in the bounding box $\boldsymbol{I}_t$ is cropped in frame t and fed forward through the network to predict the heat maps, $h_j(\cdot; \boldsymbol{I}_t)$, $\forall j = 1, \ldots, n$.

5. Empirical evaluation

5.1. Datasets and implementation details

Empirical evaluation was performed on two datasets – Human3.6M [16] and PennAction [53].

The Human3.6M dataset [16] is a recently published large-scale dataset for 3D human sensing. It includes millions of 3D human poses acquired from a MoCap system with corresponding images from calibrated cameras. This setup provides synchronized videos and 2D-3D pose data for evaluation. It includes 11 subjects performing 15 actions, such as eating, sitting and walking. The same data partition protocol as in previous work was used [23, 44]: the data from five subjects (S1, S5, S6, S7, S8) was used for training and the data from two subjects (S9, S11) was used for testing. The original frame rate is 50 fps and is downsampled to 10 fps.

The PennAction dataset [53] is a recently introduced in-the-wild human action dataset containing 2326 challenging consumer videos. The dataset consists of 15 actions, such as golf swing, bowling, and tennis swing. Each of the video sequences is manually annotated frame-by-frame with 13 human body joints in 2D. In evaluation, PennAction's training and testing split was used which consists of an even split of the videos between training and testing.

The algorithm in [56] was used to learn the pose dictionaries. The dictionary size was set to $K = 64$ for action-specific dictionaries and $K = 128$ for the nonspecific action case. For all experiments, the parameters of the proposed model were fixed ($\alpha = 0.1$, $\beta = 5$, $\gamma = 0.5$, $\nu = 4$ in a normalized 2D coordinate system).

5.2. Evaluation with known 2D poses

First, the evaluation of the 3D reconstructability of the proposed method with known 2D poses is presented. The generic approach to 3D reconstruction from 2D correspondences across a sequence is NRSFM. The proposed method is compared to the state-of-the-art method for NRSFM [14] on the Human3.6M dataset. A recent baseline method for single-view pose reconstruction Projected Matching Pursuit (PMP) [32] is also included in comparison.

	Original	Synthesized
PMP [32]	89.50	84.16
NRSFM [14]	72.98	48.88
Single frame initialization	50.04	48.08
Optimization by Algorithm 1	**49.64**	**47.57**

Table 1. 3D reconstruction given 2D poses. Two input cases are considered: original 2D pose data from Human3.6M and synthesized 2D pose data with artificial camera motion. The numbers are the mean per joint errors (mm) in 3D.

The sequences of S9 and S11 from the first camera in the Human 3.6M dataset were used for evaluation and frames beyond 30 seconds were truncated for each sequence. The 2D orthographic projections of the 3D poses provided in the dataset were used as the input. Performance was evaluated by the mean per joint error (mm) in 3D by comparing the reconstructed pose against the ground truth. As the standard protocol for evaluating NRSFM, the error was calculated up to a similarity transformation via the Procrustes analysis. To demonstrate the generality of the proposed approach, a single pose dictionary from all the training pose data, irrespective of the action type, was used, i.e., a non-action specific model. The method from Dai et al. [14] requires a predefined rank K. Here, various values of K were considered with the best result for each sequence reported.

The results are shown in the second column of Table 1. The proposed method clearly outperforms the NRSFM baseline. The reason is that the videos are captured by stationary cameras. Although the subject is occasionally rotating, the "baseline" between frames is generally small, and neighboring views provide insufficient geometric constraints for 3D reconstruction. In other words, NRSFM is very difficult to compute with slow camera motion. This observation is consistent with prior findings in the NRSFM literature, e.g., [3]. To validate this issue, an artificial rotation was applied to the 3D poses by 15 degrees per second and the 2D joint locations were synthesized by projecting the rotated 3D poses into 2D. The corresponding results are presented in the third column of Table 1. In this case, the performance of NRSFM improved dramatically. Overall, the experiments demonstrate that the structure prior (even a non-action specific one) from existing pose data is critical for reconstruction. This is especially true for videos with small camera motion, which is common in real world applications. The temporal smoothness helps but the change is not significant since the single frame initialization is very stable with known 2D poses. Nevertheless, in the next section it is shown that the temporal smoothness is important when 2D poses are not given.

	Directions	Discussion	Eating	Greeting	Phoning	Photo	Posing	Purchases
LinKDE [16]	132.71	183.55	132.37	164.39	162.12	205.94	150.61	171.31
Li et al. [23]	-	136.88	96.94	124.74	-	168.68	-	-
Tekin et al. [44]	102.39	158.52	87.95	126.83	118.37	185.02	114.69	107.61
Proposed	**87.36**	**109.31**	**87.05**	**103.16**	**116.18**	**143.32**	**106.88**	**99.78**
	Sitting	SittingDown	Smoking	Waiting	WalkDog	Walking	WalkTogether	Average
LinKDE [16]	151.57	243.03	162.14	170.69	177.13	96.60	127.88	162.14
Li et al. [23]	-	-	-	-	132.17	69.97	-	-
Tekin et al. [44]	136.15	205.65	118.21	146.66	128.11	**65.86**	**77.21**	125.28
Proposed	**124.52**	**199.23**	**107.42**	**118.09**	**114.23**	79.39	97.70	**113.01**

Table 2. Quantitative comparison on Human 3.6M datasets. The numbers are the mean per joint errors (mm) in 3D evaluated for different actions of Subjects 9 and 11.

	3D (mm)	2D (pixel)
Single frame initialization	143.85	15.00
Optimization by Algorithm 2	125.55	10.85
Perspective adjustment	**113.01**	**10.85**
No smoothness	120.99	11.25
No action label	116.49	10.87

Table 3. The estimation errors after separate steps and under additional settings. The numbers are the average per joint errors for all testing data in both 3D and 2D.

5.3. Evaluation with unknown poses: Human3.6M

Next, results on the Human3.6M dataset are reported when 2D poses are not given. The proposed method is compared to three recent baseline methods. The first baseline method is LinKDE which is provided with the Human3.6M dataset [16]. This baseline is based on single frame regression. The second one is from Tekin et al. [44] which extends the first baseline method by exploring motion information in a short sequence. The third one is a recently published CNN-based method from Li et al. [23].

In this experiment, the sequences of S9 and S11 from all cameras were used for evaluation. The standard evaluation protocol of the Human3.6M dataset was adopted, i.e., the mean per joint error (mm) in 3D is calculated between the reconstructed pose and the ground truth in the camera frame with their root locations aligned. Note that the Procrustes alignment is not allowed here. In general, it is impossible to determine the scale of the object in monocular images. The baseline methods learned the scale from training subjects. For a fair comparison, the reconstructed pose by the proposed method was scaled such that the mean limb length of the reconstructed pose was identical to the average value of all training subjects. As the alignment to the ground truth was not allowed, the joint error was largely affected by the camera rotation estimate, and empirically the misalignment was largely due to the adopted weak perspective camera model. To compensate the misalignment, the rotation estimate was refined for each frame with a perspective camera model (the 2D and 3D human pose estimates were fixed) by a perspective-n-point (PnP) algorithm [25]

The results are summarized in Table 2. The table shows that the proposed method achieves the best results on most of the actions except for "walk" and "walk together", which involve very predictable and repetitive motions and might favor the direct regression approach [44]. In addition, the results of the proposed approach have the smallest variation across all actions with a standard deviation of 28.75 versus 37.80 from Tekin et al.

In Table 3, 3D reconstruction and 2D joint localization results are provided under several setup variations of the proposed approach. Note that the 2D errors are with respect to the normalized bounding box size 256×256. The table shows that the convex initialization provides suitable initial estimates, which are further improved by the EM algorithm that integrates joint detection uncertainty and temporal smoothness. The perspective adjustment is important under the Human3.6M evaluation protocol, where Procrustes alignment to the ground truth is not allowed. The proposed approach was also evaluated under two additional settings. In the first setting, the smoothness constraint was removed from the proposed model by setting $\beta = \gamma = 0$. As a result, the average error significantly increased. This demonstrates the importance of incorporating temporal smoothness. In the second setting, a single CNN and pose dictionary was learned from all training data. These models were then applied to all testing data without distinguishing the videos by their action class. As a result, the estimation error increased, which is attributed to the fact that the 3D reconstruction ambiguity is greatly enlarged if the pose prior is not restricted to an action class.

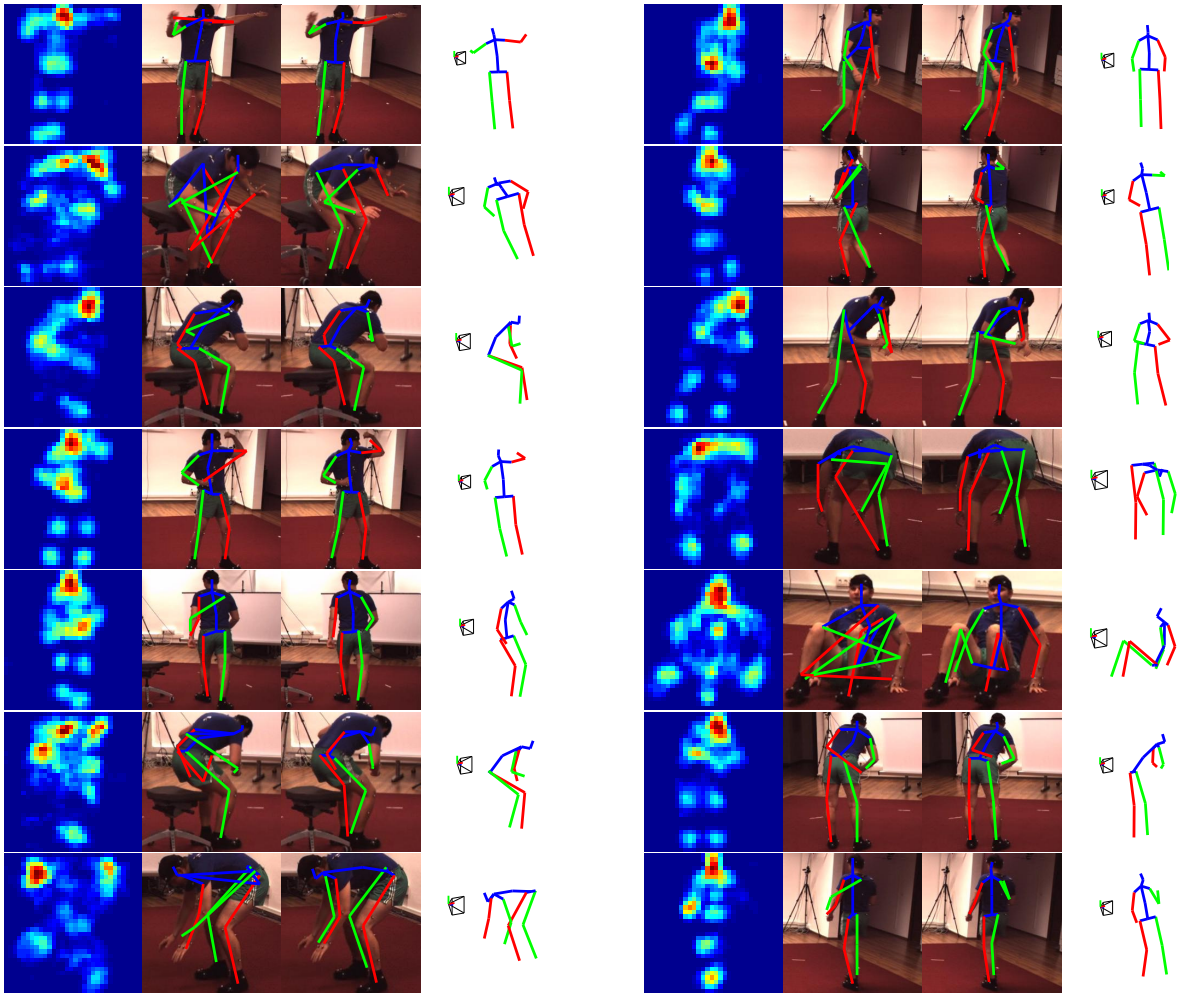

Figure 2. Example frame results on Human3.6M, where the errors in the 2D heat maps are corrected after considering the pose and temporal smoothness priors. Each row includes two examples from two actions. The figures from left-to-right correspond to the heat map (all joints combined), the 2D pose by greedily locating each joint separately according to the heat map, the estimated 2D pose by the proposed EM algorithm, and the estimated 3D pose visualized in a novel view. The original viewpoint is also shown.

Figure 2 visualizes the results of some example frames. While the heat maps may be erroneous due to occlusion, left-right ambiguity, and other uncertainty from the detectors, the proposed EM algorithm can largely correct the errors by leveraging the pose prior, integrating temporal smoothness, and modelling the uncertainty.

5.4. Evaluation with unknown poses: PennAction

Finally, the applicability of the proposed approach for pose estimation with in-the-wild videos is demonstrated. Results are reported using two actions from the PennAction dataset: "golf swing" and "tennis forehand", both of which are very challenging due to large pose variability, self-occlusion, and image blur caused by fast motion. For the proposed approach, the CNN was trained using the annotated training images from the PennAction dataset, while the pose dictionary was learned with publicly available MoCap data[1]. Due to the lack of 3D ground truth, quantitative 2D pose estimation results are reported and compared with the publicly available 2D pose detector from Yang and Ramanan [50]. The baseline was retrained on the PennAction dataset. Note that the baseline methods considered in Section 5.3 are not applicable here since they require synchronized 2D image and 3D pose data for training.

To measure joint localization accuracy, both the widely used per joint distance errors and the probability of correct keypoint (PCK) metrics are used. The PCK metric measures the fraction of correctly located joints with respect to a threshold. Here, the threshold is set to 10 pixels which is roughly the half length of a head segment.

Table 4 summarizes the quantitative results. The initial-

[1]Data sources: http://mocap.cs.cmu.edu and http://www.motioncapturedata.com

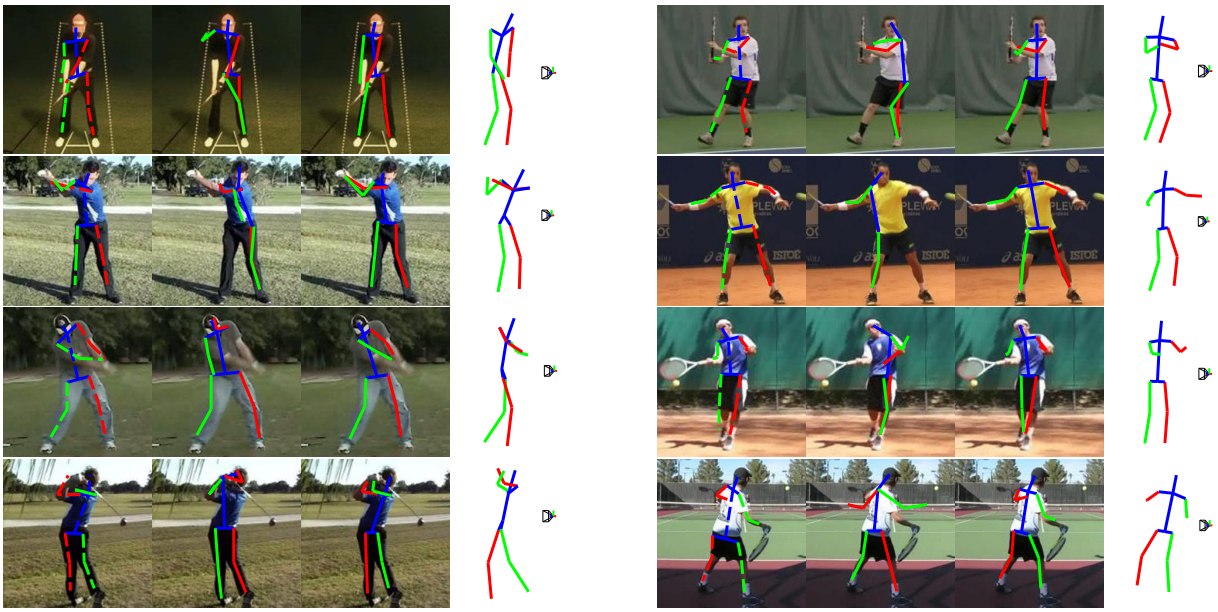

Figure 3. Example results on PennAction. Each row includes two examples. In each example, the figures from left-to-right correspond to the ground truth superimposed on the image, the estimated pose using the baseline approach [50], the estimated pose by the proposed approach, and the estimated 3D pose visualized in a novel view. The original viewpoint is also shown.

	Baseline	Initial	Optimized
Golf	24.78 / 0.38	18.73 / 0.45	**14.03 / 0.54**
Tennis	29.15 / 0.40	25.75 / 0.42	**20.99 / 0.45**

Table 4. 2D pose errors on PennAction. Each pair of numbers correspond to the per joint distance error (pixels) and the PCK metric. The baseline is the retrained model from Yang and Ramanan [50]. The last two columns correspond to the errors after initialization and EM optimization in the proposed approach.

ization step alone outperformed the baseline. This demonstrates the effectiveness of CNN-based approaches, which has been shown in many recent works, e.g., [46, 31]. The proposed EM algorithm further improves upon the initialization results by a large margin by integrating the geometric and smoothness priors. Several example results are shown in Figure 3. It can be seen that the proposed method successfully recovers the poses for various subjects under a variety of viewpoints. In particular, compared to the baseline, the proposed method does not suffer from the well-known "double-counting" problem for tree-based models [50] due to the holistic 3D pose prior.

5.5. Running time

The experiments were performed on a desktop with an Intel i7 3.4G CPU, 8G RAM and a TitanZ GPU. The running times for CNN-based heat map generation and convex initialization were roughly 1s and 0.6s per frame, respectively; both steps can be easily parallelized. The EM algorithm usually converged in 20 iterations with a CPU time less than 100s for a sequence of 300 frames.

6. Summary

In summary, a 3D pose estimation framework from video has been presented that consists of a novel synthesis between a deep learning-based 2D part regressor, a sparsity-driven 3D reconstruction approach and a 3D temporal smoothness prior. This joint consideration combines the discriminative power of state-of-the-art 2D part detectors, the expressiveness of 3D pose models and regularization by way of aggregating information over time. In practice, alternative joint detectors, pose representations and temporal models can be conveniently integrated in the proposed framework by replacing the original components. Experiments demonstrated that 3D geometric priors and temporal coherence can not only help 3D reconstruction but also improve 2D joint localization. Future extensions may include incremental algorithms for online tracking-by-detection and handling multiple subjects.

Supplementary material: The MATLAB code, evaluation on the HumanEva I dataset, demonstration videos, and other supplementary materials are available at: `http://cis.upenn.edu/~xiaowz/monocap.html`.

Acknowledgments: The authors are grateful for support through the following grants: NSF-DGE-0966142, NSF-IIS-1317788, NSF-IIP-1439681, NSF-IIS-1426840, ARL MAST-CTA W911NF-08-2-0004, ARL RCTA W911NF-10-2-0016, ONR N000141310778, and NSERC Discovery.

References

[1] A. Agarwal and B. Triggs. Recovering 3D human pose from monocular images. *PAMI*, 28(1):44–58, 2006. 2

[2] I. Akhter and M. J. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *CVPR*, 2015. 2

[3] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *PAMI*, 33(7):1442–1456, 2011. 2, 5

[4] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *CVPR*, 2014. 1

[5] M. Andriluka, S. Roth, and B. Schiele. Monocular 3D pose estimation and tracking by detection. In *CVPR*, 2010. 1, 2

[6] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre. Manopt, a Matlab toolbox for optimization on manifolds. *JMLR*, 15:1455–1459, 2014. 3

[7] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *CVPR*, 2000. 2

[8] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *CVPR*, 1998. 2

[9] M. A. Brubaker, L. Sigal, and D. J. Fleet. Video-based people tracking. In *Handbook of Ambient Intelligence and Smart Environments*, pages 57–87. Springer, 2010. 2

[10] X. Chen and A. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*, 2014. 2

[11] A. Cherian, J. Mairal, K. Alahari, and C. Schmid. Mixing body-part sequences for human pose estimation. In *CVPR*, pages 2361–2368, 2014. 2

[12] J. Cho, M. Lee, and S. Oh. Complex non-rigid 3D shape recovery using a Procrustean normal distribution mixture model. *IJCV*, pages 1–21, 2015. 2

[13] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models–Their training and application. *CVIU*, 61(1):38–59, 1995. 2

[14] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. *IJCV*, 107(2):101–122, 2014. 2, 5

[15] X. Fan, K. Zheng, Y. Zhou, and S. Wang. Pose locality constrained representation for 3D human pose reconstruction. In *ECCV*, 2014. 2

[16] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments. *PAMI*, 36(7):1325–1339, 2014. 2, 5, 6

[17] A. Jain, J. Tompson, M. Andriluka, G. Taylor, and C. Bregler. Learning human pose estimation features with convolutional networks. In *ICLR*, 2014. 2

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 5

[19] H. Jiang. 3D human pose reconstruction using millions of exemplars. In *ICPR*, 2010. 2

[20] H. Lee and Z. Chen. Determination of 3D human body postures from a single view. *CVGIP*, 30(2):148–168, 1985. 1, 2

[21] S. Leonardos, X. Zhou, and K. Daniilidis. Articulated motion estimation from a monocular image sequence using spherical tangent bundles. In *ICRA*, 2016. 2

[22] S. Li and A. B. Chan. 3D human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, 2014. 2

[23] S. Li, W. Zhang, and A. B. Chan. Maximum-margin structured learning with deep networks for 3D human pose estimation. In *ICCV*, 2015. 2, 5, 6

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 4

[25] C.-P. Lu, G. D. Hager, and E. Mjolsness. Fast and globally convergent pose estimation from video images. *PAMI*, 22(6):610–622, 2000. 6

[26] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *CVIU*, 104(2):90–126, 2006. 2

[27] G. Mori and J. Malik. Recovering 3D human body configurations using shape contexts. *PAMI*, 28(7):1052–1062, 2006. 2

[28] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2007. 3

[29] D. Park and D. Ramanan. Articulated pose estimation with tiny synthetic videos. In *ChaLearn Workshop on Looking at People, CVPR*, 2015. 2

[30] H. S. Park and Y. Sheikh. 3D reconstruction of a smooth articulated trajectory from a monocular image sequence. In *ICCV*, pages 201–208, 2011. 2

[31] T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *ICCV*, 2015. 2, 4, 8

[32] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3D human pose from 2D image landmarks. In *ECCV*, 2012. 2, 5

[33] D. Ramanan. Part-based models for finding people and estimating their pose. In *Visual Analysis of Humans - Looking at People*, pages 199–223. Springer, 2011. 2

[34] M. Salzmann and R. Urtasun. Implicitly constrained Gaussian process regression for monocular non-rigid pose estimation. In *NIPS*, 2010. 2

[35] B. Sapp, D. J. Weiss, and B. Taskar. Parsing human motion with stretchable models. In *CVPR*, pages 1281–1288, 2011. 2

[36] G. Shakhnarovich, P. A. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *ICCV*, 2003. 2

[37] J. Shotton, A. W. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011. 1

[38] L. Sigal, M. Isard, H. W. Haussecker, and M. J. Black. Loose-limbed people: Estimating 3D human pose and motion using non-parametric belief propagation. *IJCV*, 98(1):15–48, 2012. 2

[39] E. Simo-Serra, A. Quattoni, C. Torras, and F. Moreno-Noguer. A Joint Model for 2D and 3D Pose Estimation from a Single Image. In *CVPR*, 2013. 2

[40] E. Simo-Serra, A. Ramisa, G. Alenyà, C. Torras, and F. Moreno-Noguer. Single Image 3D Human Pose Estimation from Noisy Observations. In *CVPR*, 2012. 2

[41] C. Sminchisescu. 3D human motion analysis in monocular video techniques and challenges. In *AVSS*, 2007. 2

[42] C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3D human tracking. In *CVPR*, 2003. 2

[43] C. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *CVIU*, 80(3):349–363, 2000. 2

[44] B. Tekin, X. Sun, X. Wang, V. Lepetit, and P. Fua. Predicting people's 3D poses from short sequences. *arXiv preprint arXiv:1504.08200*, 2015. 2, 5, 6

[45] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, 2014. 2

[46] A. Toshev and C. Szegedy. DeepPose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 1, 2, 8

[47] J. Valmadre and S. Lucey. Deterministic 3D human pose estimation using rigid structure. In *ECCV*, 2010. 2

[48] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao. Robust estimation of 3D human poses from a single image. In *CVPR*, 2014. 2

[49] B. Xiaohan Nie, C. Xiong, and S.-C. Zhu. Joint action recognition and pose estimation from video. In *CVPR*, 2015. 1, 2

[50] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 1, 2, 7, 8

[51] T. Yu, T. Kim, and R. Cipolla. Unconstrained monocular 3D human pose estimation by action detection and cross-modality regression forest. In *CVPR*, 2013. 2

[52] D. Zhang and M. Shah. Human pose estimation in videos. In *ICCV*, 2015. 2

[53] W. Zhang, M. Zhu, and K. G. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*, 2013. 2, 5

[54] F. Zhou and F. D. la Torre. Spatio-temporal matching for human detection in video. In *ECCV*, 2014. 1, 2

[55] X. Zhou, S. Leonardos, X. Hu, and K. Daniilidis. 3D shape estimation from 2D landmarks: A convex relaxation approach. In *CVPR*, 2015. 2, 4

[56] X. Zhou, M. Zhu, S. Leonardos, and K. Daniilidis. Sparse representation for 3D shape estimation: A convex relaxation approach. *arXiv preprint arXiv:1509.04309*, 2015. 2, 4, 5

[57] Y. Zhu, D. Huang, F. De la Torre, and S. Lucey. Complex non-rigid motion 3D reconstruction by union of subspaces. In *CVPR*, 2014. 2