Deep Kinematics Analysis for Monocular 3D Human Pose Estimation

Jingwei Xu*[1,2], Zhenbo Yu*[1,2], Bingbing Ni†[1,2,3], Jiancheng Yang[1,2], Xiaokang Yang[1,2], Wenjun Zhang[1,2]
[1]Shanghai Jiao Tong University, Shanghai 200240, China
[2]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[3]Huawei Hisilicon
{xjwxjw,yuzhenbo,nibingbing,jekyll4168,xkyang,zhangwenjun}@sjtu.edu.cn, nibingbing@hisilicon.com

Abstract

For monocular 3D pose estimation conditioned on 2D detection, noisy/unreliable input is a key obstacle in this task. Simple structure constraints attempting to tackle this problem, e.g.*, symmetry loss and joint angle limit, could only provide marginal improvements and are commonly treated as auxiliary losses in previous researches. It still remains challenging to fully utilize human prior knowledge in this task. In this paper, we propose to address above issue in a systematic view. Firstly, we show that optimizing the kinematics structure of noisy 2D inputs is critical to obtain accurate 3D estimations. Secondly, based on corrected 2D joints, we further explicitly decompose articulated motion with human topology, which leads to more compact 3D static structure easier for estimation. Finally, we propose a temporal module to refine 3D trajectories, which obtains more rational results. Above three steps are seamlessly integrated into deep neural models, which form a deep kinematics analysis pipeline concurrently considering the static/dynamic structure of 2D inputs and 3D outputs. Extensive experiments show that proposed framework achieves state-of-the-art performance on two widely used 3D human action datasets. Meanwhile, targeted ablation study shows that each former step is critical for the latter one to obtain promising results.*

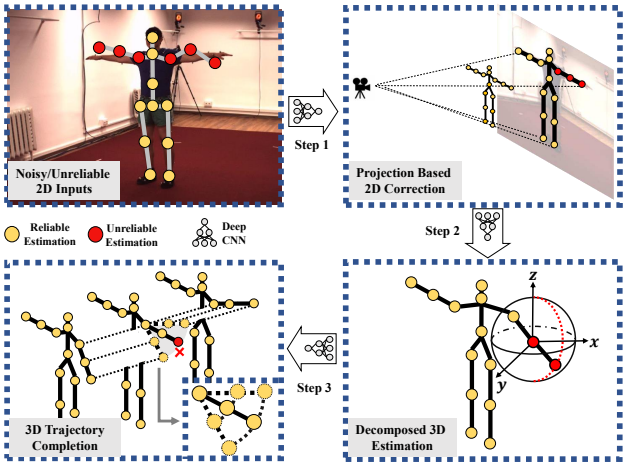

Figure 1: Overview of proposed framework. Our model pursues 3D pose estimation with more reasonable structure and more compact output space, which incorporates kinematics analysis into deep models. Step 1: Noisy/unreliable 2D Inputs (denoted as red dots) are corrected with perspective projection. Step 2: 3D poses are further estimated in a decomposed manner within more compact space. Step 3: Unreliably estimated 3D poses are excluded from previous outputs (denoted by red cross), which are finally refined as a completion task.

1. Introduction

Pose estimation is a hot-spot topic in computer vision researches [35, 51, 2, 5]. Particularly, 3D pose estimation for monocular video has drawn tremendous attention in the past decades [21, 18, 37, 33, 1, 6], which involves estimating keypoint trajectories of human subject in 3D space. This research topic possesses several valuable downstream applications, *e.g.*, action recognition [3, 23], human body reconstruction [16, 13, 47] and robotics manipulation [34].

Recently, many works [11, 5] have used 2D pose detectors to facilitate the 3D human pose estimation task. Several previous researches [28, 25, 46, 38, 7] take detected 2D keypoints as input and predict corresponding 3D joint locations from monocular video, which have promising results and require much less training resources compared to other works fed with RGB images [19, 42, 10, 54]. Our work belongs to this branch with explicit incorporation of kinematics analysis, which would be discussed in detail at latter paragraphs.

Despite considerable progress in 2D-keypoint condi-

*Equal contribution.
†Corresponding author: Bingbing Ni

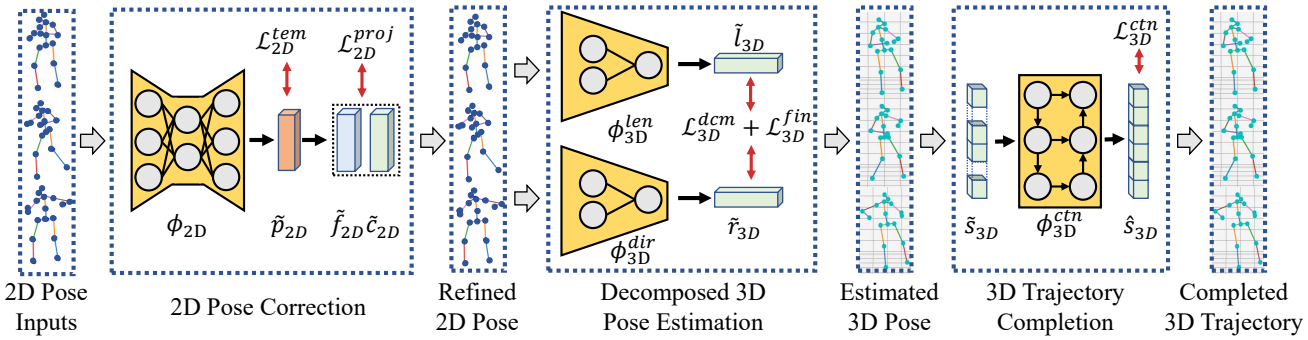

Figure 2: Detailed architecture of proposed framework. Three hierarchical modules (from left to right) correspond to 2D pose correction (ϕ_{2D}), 3D pose estimation ($\phi_{3D}^{len}, \phi_{3D}^{dir}$) and 3D trajectory refinement (ϕ_{3D}^{ctn}) respectively. $\tilde{p}_{2D}$ is corrected 2D pose and $\tilde{f}, \tilde{c}$ refer to regressed camera parameters. $\tilde{l}_{3D}$ and $\tilde{r}_{3D}$ are estimated length and direction from $\phi_{3D}^{len}, \phi_{3D}^{dir}$. $\tilde{S}_{3D}$ and $\hat{S}_{3D}$ stand for 3D pose before/after trajectory completion respectively.

tioned 3D pose estimation, several critical challenges remain yet to be solved. On one hand, 2D detections are generally noisy and unreliable due to motion blur and self-occlusion contained in video sequences. Current methods [25, 46, 33] adopt simple structure constraints, *e.g.*, symmetric bone length [33] and limited joint angle [16], to facilitate 3D joint predictions, which are insufficient to bring significant improvements for this task. On the other hand, the majority of existing approaches directly formulate this task as a coordinate regression problem, which do not fully take the inherent kinematics structure of human subject into consideration and commonly leads to invalid results.

Real-world human motion obeys kinematics laws involving 2D/3D correspondence and static/dynamic structure: (1) For camera-based view, 3D and projected 2D joints should follow the constraint of perspective projection. (2) For static structure, the length between two adjacent 3D joints (defined by skeleton) should be constant throughout the whole motion sequence. (3) For dynamic structure, estimated 3D trajectory formed by the same joint should be smooth and continuous. We are thus motivated to integrate all above laws forming a systematical analysis pipeline from correspondence to structure, which facilitates pursuing 3D poses within more rational solution space.

In this paper, we propose to systematically incorporate kinematics analysis into deep models for effective utilization of human prior knowledge. As illustrated in Fig. 1, we firstly refine 2D inputs to be more reliable rather than only consider the estimation accuracy of 3D keypoints [33, 1]. A novel optimization scheme is designed for 2D keypoints under the constraint of perspective projection, which mainly facilitates kinematics structure correction of noisy 2D inputs. To our best knowledge, it is the first attempt that perspective projection is used for 2D joints refinement rather than 3D counterparts. Experimental results demonstrate that above 2D optimization scheme is critical for the following 3D pose estimation. Secondly, starting from the static structure of human subject, we decompose articulated motion based on rigid body assumption, which breaks unconstrained 3D trajectories down to a tree-structured combination of 2D sphere curves with much lower dimension. More specifically, we split 3D coordinate regression problem into two sub-tasks, *i.e.*, length and direction estimation, which are complementary to each other and reduce the learning difficulty by a large margin. Finally, we notice that not all parts are equally estimated. To pursue valid dynamic structure we exclude those joints with low reliability from above predictions and the whole 3D trajectory is completed based on more reliable parts. Above three steps are seamlessly integrated into deep neural models, which form a systematic analysis pipeline, *i.e.*, our model simultaneously considers the kinematics structure of 2D inputs and 3D outputs.

We conduct detailed ablation study to demonstrate the contribution of each component of proposed framework. Further extensive experiments show that our model achieves state-of-the-art performance on two widely used 3D human motion datasets.

2. Related Work

In this section, we discuss the approaches that are based on deep neural networks for 3D pose estimation.

Holistic 3D pose estimation. With the excellent feature extraction capacity of deep neural networks, many approaches [19, 31, 44, 42, 26, 32, 27, 10, 54, 6] utilize Deep Convolutional Neural Networks to estimate 3D poses from the images or other sources (e.g., point clouds [50, 22, 49]) directly. In this paper we concentrate on the image one. Li *et al*. [19] firstly apply CNNs to jointly estimate 3D poses and detect body parts via a multi-task framework. Tekin *et al*. [42] use an overcomplete auto-encoder to learn a high-

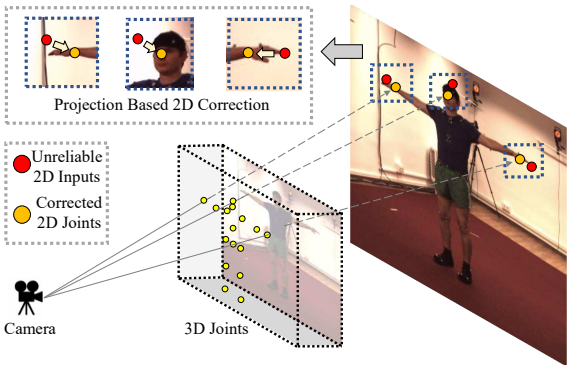

Figure 3: Example of perspective projection used to refine 2D inputs. This essentially incorporates 3D supervision signal for 2D pose training, where 2D/3D correspondence is better preserved.

dimensional latent pose representation and account for joint dependencies. Differently, [10] and [54] both utilize intermediate 3D representations and 2D counterparts to regress 3D poses. However, training deep models directly from images requires expensive computation resources and careful hyper-parameter tuning. Differently, our model starts with detected 2D joints as inputs whose pipeline is largely simplified but with comparable performance.

Two-step pose estimation. To avoid collecting 2D-3D paired data, a variety of works [55, 25, 43, 4, 9, 53, 52, 28, 38, 7] decouple the task of 3D pose estimation into two independent stages: (1) firstly predicting 2D joint location in image space using off-the-shelf 2D pose estimation methods; (2) and then learning a mapping to lift them to 3D space. Moreno *et al*. [28] learn a mapping from 2D to 3D distance matrices. A simple yet effective method [25] can directly predict 3D joint locations via deep CNNs. Regarding the prior knowledge of human structure, Wang *et al*. [46] propose a progressive approach that explicitly accounts for the distinct DOFs among the body parts. Sharma *et al*. [38] synthesize diverse anatomically plausible 3D-pose samples conditioned on the estimated 2D poses via a deep conditional variational auto-encoder based model. The generic combination [7] of Graph Convolutional Network (GCN) and Connected Network (FCN) can also improve the representation capability. These approaches mainly focus on the second stage and our method also belongs to this branch. However, few researches have paid attention on the inherent validity of 2D/3D poses in a systemic and comprehensive view, where kinematics structure and view correspondence are generally ignored.

Video pose estimation. Since most previous works operate in a single-frame setting, recently more attention [14, 21, 18, 8, 37, 33, 1, 6, 20, 36, 48] has been put on temporal information from monocular video clips. LSTMs [21] have been applied to refine 3D poses predicted from single images. There has also been work on RNN approach [18] which considers prior knowledge with body part based structural connectivity. Rayat *et al*. [37] utilize the temporal smoothness constraint across a sequence of 2D joint locations to estimate a sequence of 3D poses. Pavllo *et al*. [33] transform a sequence of 2D poses through temporal convolutions and make computational complexity independent of key-point spatial resolution. Multi-scale features for the graph-based representations [1] are critic to pose estimation by a local-to-global network architecture. Unlike existing temporal based methods, our model explicitly incorporate a kinematics analysis pipeline for 3D pose estimation.

Table 1: Influence of 2D accuracy for 3D pose estimation. Both are evaluated by mean squared error. $M = 1$ is no 2D refinement is applied.

Window Length	2D Joints Train/Test	3D Joints Train/Test
M=1	0.058 / 0.078	0.027 / 0.053
M=5	0.042 / 0.071	0.024 / 0.052
M=9	0.033 / 0.067	0.023 / 0.052

3. Method

In this section, we present detailed description of proposed method. The overall framework is illustrated in Fig. 2. In our method and experiments, we focus on pose estimation over a short video clip ($T \leq 9$).

3.1. Projection based 2D Pose Correction

2D Temporal Refinement. Given a monocular video clip with length of T time stamps, we first apply pretrained 2D pose detector (*e.g*., CPN [5]) to obtain 2D joints. 2D detections on single frame are generally noisy and unreliable due to motion blur and occlusion [11]. We first utilize a temporal CNN model (denoted as ϕ_{2D} as shown in Fig. 2) to refine 2D initial inputs (denoted as $\tilde{\mathbf{P}} = \{\tilde{\mathbf{p}}_t\}_{t=1}^{T}, \tilde{\mathbf{K}} = \{\tilde{\mathbf{k}}_t\}_{t=1}^{T}, \tilde{\mathbf{p}}_t \in \mathbb{R}^{J\times 2}, \tilde{\mathbf{k}}_t \in \mathbb{R}^{J}$.) Here J refers to the number of joints for single human. Specifically, $\tilde{\mathbf{p}}_t = [\tilde{\mathbf{a}}_t, \tilde{\mathbf{b}}_t]$ where $\tilde{\mathbf{a}}_t$ and $\tilde{\mathbf{b}}_t$ represent 2D coordinates and $\tilde{\mathbf{k}}_t$ is corresponding confidence score. We adopt MSE loss weighted by confidence score $\tilde{\mathbf{K}}$ for trained as follows:

$$\mathcal{L}_{2D}^{Tem} = \sum_{t=1}^{T} \tilde{\mathbf{k}}_t \sqrt{|\mathbf{a}_t - \tilde{\mathbf{a}}_t|_2^2 + |\mathbf{b}_t - \tilde{\mathbf{b}}_t|_2^2}, \qquad (1)$$

where $\mathbf{a}_t, \mathbf{b}_t$ refers to the ground truth 2D joints. Intuitively, this procedure utilizes temporal smoothness to refine detected 2D joints.

Limitation: Train/Test Imbalance. However, above optimization does not directly facilitate performance boosting for final 3D pose estimation. We conduct verification experiments (on Human3.6M dataset [15]) to support our statements. Specifically, we adopt single-frame 3D pose estimation model [33] with refined 2D joints as inputs. As shown in Tab. 1, estimation accuracy of 2D and 3D joints on both train and test set are reported. We can observe that with increase of window length, the training accuracy boosts for both 2D and 3D joints. But the improvement on 3D test set is marginal (last column). We attribute this for **imbalanced inputs on Train/Test set**. The large accuracy gap between train and test inputs (first two columns) is unknown to the following 3D pose estimation model, which leads to sub-optimal results.

Solution: Projection Constraint. To this end, we propose to refine the 2D inputs in a different point of view, *i.e.*, 2D/3D correspondence is critical for rational pose estimation (as shown in Fig. 3). We denote 3D joints as $\mathbf{S} = \{\mathbf{s}_t\}_{t=1}^{T}, \mathbf{s}_t = [\mathbf{x}_t, \mathbf{y}_t, \mathbf{z}_t] \in \mathbb{R}^{J\times 3}$. For each time stamp t, projected 2D joints $\mathbf{p}$ and 3D joints $\mathbf{s}$ should obey perspective projection as follows:

$$\mathbf{a} = \frac{\mathbf{x}}{\mathbf{z}} f_x + c_x, \mathbf{b} = \frac{\mathbf{y}}{\mathbf{z}} f_y + c_y, \tag{2}$$

where $\mathbf{f} = [f_x, f_y]$ and $\mathbf{c} = [c_x, c_y]$ are focal length and point respectively. We omit subscript t for simplicity. Our main idea is to recover $\mathbf{f}$ and $\mathbf{c}$ from refined 2D inputs and ground truth 3D joints during training. Intuitively, well estimated $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{c}}$ indicate the projection correspondence is generally preserved. The remaining problem is how to obtain $\tilde{\mathbf{f}}$ and $\tilde{\mathbf{c}}$ efficiently.

Linear squares regression is a simple yet effective approach for above issue. Take the projection x-axis coordinates for example. As indicated by Eqn. 2, paired data points $(\mathbf{a}, \frac{\mathbf{x}}{\mathbf{z}}) = \{(a^j, \frac{x^j}{z^j})\}_{j=1}^{J}$ involving single human subject follows a straight line whose intercept and slope correspond to f_x and c_x respectively. For simplicity we omit the dominator, *i.e.*, $\frac{x^j}{z^j} : x^j$. Given refined 2D detections $\tilde{\mathbf{p}}$ and ground truth 3D joints $\mathbf{s}$, we estimate f_x and c_x as follows:

$$\tilde{f}_x = \frac{\bar{a}(\sum_{j=1}^{J} x^j)^2 - \bar{x}\sum_{j=1}^{J} x^j \tilde{a}^j}{\sum_{j=1}^{J}(x^j)^2 - J\bar{x}^2}, \tilde{c}_x = \bar{a} - \tilde{f}_x \bar{x}, \tag{3}$$

where $\bar{a} = \frac{1}{J}\sum_{j=1}^{J} \tilde{a}^j, \bar{x} = \frac{1}{J}\sum_{j=1}^{J} x^j$. We obtain ground truth f_x and c_x the same as Eqn. 3 with ground truth $\mathbf{p}$ and $\mathbf{s}$ as inputs. We adopt L1 loss for training as follows:

$$\mathcal{L}_{2D}^{proj} = |f_x - \tilde{f}_x| + |c_x - \tilde{c}_x|. \tag{4}$$

Furthermore, estimated $\tilde{f}_x$ and $\tilde{c}_x$ at each time stamp should be constant for the whole monocular video clip, which is also utilized for training. More specifically, we randomly select two time stamps, where estimated $\tilde{f}_x$ and $\tilde{c}_x$ are penalized with L1 difference. As shown in Fig. 2, ϕ_{2D} adopts an encoder-decoder architecture, the temporal length of outputs are equal to that of inputs. For detailed architecture please refer to supplementary material.

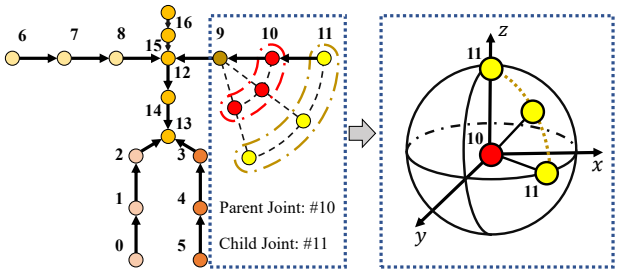

Figure 4: Illustration of articulated motion for a pair of parent-children joints. The motion trajectory of children joint relative to its parent degenerates to spherical curve.

Note that datasets used in our paper (Human3.6M [15] and HumanEva [39] Dataset) are recorded by cameras with fixed and almost identical ($\mathbf{f}$,$\mathbf{c}$), which indicates a **global projection relation** existing between all 2D and 3D poses. The refinement model is trained towards **a single projection relation**. If video sequences are recorded by cameras with varied ($\mathbf{f}$,$\mathbf{c}$), we could feed the ($\mathbf{f}$,$\mathbf{c}$) as inputs to guarantee the generalization ability of proposed model. As intrinsic parameters, ($\mathbf{f}$,$\mathbf{c}$) are generally low cost to obtain [33]. This part is leaved as future work worth studying.

3.2. Decomposed 3D pose estimation

Based on refined 2D joints, our second model (denoted as ϕ_{3D}^{dcm}) predicts corresponding 3D keypoints. As discussed in Sec. 1, 3D articulated motion involved in monocular video clip is structure constrained. Fig. 4 depicts this kinematics law more explicitly. The motion trajectory of children joint relative to its parent (defined by human skeleton) forms a spherical curve.

Explicit Pose Decomposition. Motivated by this, we decompose original coordinate regression problem into two complementary sub-tasks, *i.e.*, length and direction regression. Specifically, for a 3D sequence $\mathbf{S} \in \mathbb{R}^{T\times J\times 3}$, we first obtain the relative coordinate according to predefined skeleton topology, *i.e.*, $\Delta\mathbf{S}^{j_c} = \mathbf{S}^{j_c} - \mathbf{S}^{j_p}$, where j_c refers some child joint and j_p indicates its parent joint (shown in Fig. 4). Suppose the skeleton defined by joint pair $\{j_c, j_p\}$ is of length l^{j_c}, $\Delta\mathbf{S}^{j_c}$ could be rewritten as $\Delta\mathbf{S}^{j_c} = \{l^{j_c}\mathbf{r}_t^{j_c}\}_{t=1}^{T}$, where $\mathbf{r}_t^{j_c}$ is the unit vector representing direction of skeleton between (j_c, j_p) at time stamp t. l^{j_c} is kept constant across the whole video clip. Therefore we predict l^{j_c} and $\mathbf{r}_t^{j_c}$ as intermediate results, which are composed together according to human skeleton for final estimation (Eqn. 5).

Global-local Combined 2D inputs. We extend inputs

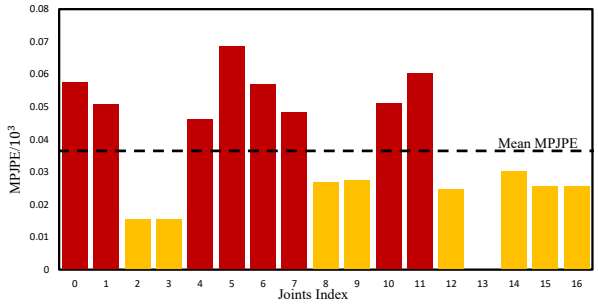

Figure 5: Per-joint estimation error in terms of MPJPE. Notably, the estimation error of four limbs (indexed by 0/1/4/5/6/7/10/11 with red bar) is significantly higher than other joints. Dashed black line indicates mean MPJPE for all joints.

$\tilde{\mathbf{P}}$ with similar manner as $\mathbf{S}$, *i.e.*, $[\tilde{\mathbf{P}}, \Delta\tilde{\mathbf{P}}, ||\Delta\tilde{\mathbf{P}}||_2^2] \in \mathbb{R}^{T\times J\times 5}$. $\Delta\tilde{\mathbf{P}} \in \mathbb{R}^{T\times J\times 2}$ is obtained the same as $\Delta\mathbf{S}$ and $||\Delta\tilde{\mathbf{P}}||_2^2 \in \mathbb{R}^{T\times J\times 1}$ is calculated pixel length for each skeleton. Above operation combines both global and local information as inputs, which provides more structured information for 3D pose estimation.

Length & Direction Estimation. We adopt a two-stream architecture for length and direction estimation, where two sub-modules are denoted as ϕ_{3D}^{len} and ϕ_{3D}^{dir} respectively. For length estimation, we first extract pose feature for each time stamp, and then conduct feature aggregation at middle level representation, which are finally used to predict skeleton length $\tilde{\mathbf{l}} = \{\tilde{l}^j\}_{j=1}^J$. In contrast, skeleton direction is estimated independently at each time stamp and the unit vector $\tilde{\mathbf{r}}_t = \{\tilde{\mathbf{r}}_t^j\}_{j=1}^J$ is obtained through a L2 normalization layer. For detailed architecture of ϕ_{3D}^{len} and ϕ_{3D}^{dir}, please refer to supplementary material. Up to now we obtain $\Delta\tilde{\mathbf{S}} = \tilde{\mathbf{l}}\tilde{\mathbf{R}}$, where $\tilde{\mathbf{R}} = \{\tilde{\mathbf{r}}_t\}_{t=1}^T$. Final estimation $\tilde{\mathbf{S}}$ is obtained through iterative summarization over $\Delta\tilde{\mathbf{S}}$ according to human topology as follows:

$$\tilde{\mathbf{S}}^{j_c} = \Delta\tilde{\mathbf{S}}^{j_c} + \tilde{\mathbf{S}}^{j_p}. \tag{5}$$

Correspondingly, we adopt intermediate loss functions to facilitate the training procedure. For length estimation, we use L1 loss (denoted as $\mathcal{L}_{3D}^{len} = |\tilde{\mathbf{l}} - \mathbf{l}|$) to obtain more accurate results. For direction estimation, a cosine similarity loss (denoted as $\mathcal{L}_{3D}^{dir} = \langle\tilde{\mathbf{r}}, \mathbf{r}\rangle - 1$) is applied on $\tilde{\mathbf{R}}$, which penalizes angle difference rather than distance. For final composed prediction, we use L2 loss (denoted as $\mathcal{L}_{3D}^{fin}$) for training. All three loss functions are shown as follows. For detailed architecture of ϕ_{3D}^{dir} and ϕ_{3D}^{len} please refer to supplementary material.

$$\mathcal{L}_{3D}^{dcm} = \mathcal{L}_{3D}^{len} + \mathcal{L}_{3D}^{dir}, \mathcal{L}_{3D}^{fin} = ||\tilde{\mathbf{S}} - \mathbf{S}||_2^2. \tag{6}$$

Discussion on Pose Decomposition. The most related work for this part is Sun *et al*. [41], which conducts pose estimation in an compositional way. However, our work differentiates from Sun *et al*. [41] in following three aspects: (1) Our work concentrate on monocular pose estimation rather than single image. (2) Instead of direct coordinate regression, we explicitly decomposes output space with rigid body structure, whose dimension is reduced from $3 \times T \times J$ to $2 \times T \times J + J$, where the first part is about direction estimation while the second part corresponds to length estimation. More compact output space facilitates 3D pose estimation model to obtain more rational results by a large margin. (3) We design targeted losses $\mathcal{L}_{3D}^{dir}$ and $\mathcal{L}_{3D}^{len}$ based on above length/direction decomposition, which reduces the learning difficulty by a large margin.

3.3. Pose Refinement as Trajectory Completion

Based on pose decomposition we estimate the skeleton direction at each time stamp independently, which needs further refinement. In this section, we consider above problem as trajectory completion task, where refinement is applied on unreliably estimated joints.

Which joints should be refined? Not all joints are estimated equally. As shown in Fig 5, we can observe that the estimation error of four limbs (*i.e*., joints 0/1/4/5/6/7/10/11 also as shown in Fig. 4) is significantly higher than others. Therefore, we focus on four-limb regularization. Meanwhile, the confidence score $\tilde{\mathbf{K}}$ (mentioned in Sec. 3.1) produced by 2D detectors is an informative indicator for unreliable estimation of joints location. To this end, we optimize four-limb joints assigned with low 2D confidence score.

Trajectory Completion with Reliable Estimation. Given estimated 3D joints $\tilde{\mathbf{S}} \in \mathbb{R}^{T\times J\times 3}$, we first apply a dropout layer [40] directly on the joints associated with four limbs. The dropout rate is $1-\tilde{\mathbf{K}}$ rather than a constant value, *i.e*., unreliable joints are excluded from $\tilde{\mathbf{S}} \in \mathbb{R}^{T\times J\times 3}$ which are further completed with reliable ones. The completion model is denoted as ϕ_{3D}^{ctn} consisting of a bi-directional LSTM [12] network (shown in Fig. 2). We denote completed output as $\hat{\mathbf{S}}$, which trained as follows:

$$\mathcal{L}_{3D}^{ctn} = ||\hat{\mathbf{S}} - \mathbf{S}||_2^2 + ||\mathcal{H}(\hat{\mathbf{S}}) - \mathcal{H}(\mathbf{S})||_2^2 + ||\mathcal{F}(\hat{\mathbf{S}}) - \mathcal{F}(\mathbf{S})||_2^2, \tag{7}$$

where $\mathcal{H}$ and $\mathcal{F}$ refer to first and second order difference over temporal axis respectively. Intuitively, high-order continuity facilitates better modelling of dynamic structure for human subject.

3.4. Implementation Details

Our model is end-to-end trainable and the overall loss function is $\mathcal{L}_{2D}^{tem} + 0.1\mathcal{L}_{2D}^{proj} + 0.1\mathcal{L}_{2D}^{pel} + \mathcal{L}_{3D}^{dcm} + \mathcal{L}_{3D}^{fin} + 0.1\mathcal{L}_{3D}^{ctn}$. We adopt PyTorch [29] to implement our proposed framework. During training phase, learning rate, learning decay and weight decay are set to $1e^{-3}, 0.93, 1e^{-4}$ respectively. Dropout rate is set to 0.25 except the one mentioned

Protocol 1: MPJPE	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg
Martinez *et al*. [25] ICCV'17 (T=1)	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Luvizon *et al*. [24] CVPR'18 (T=1)	49.2	51.6	47.6	50.5	51.8	60.3	48.5	51.7	61.5	70.9	53.7	48.9	57.9	44.4	48.9	53.2
Hossain & Little [37] ECCV'18 (T=5)	48.4	50.7	57.2	55.2	63.1	72.6	53.0	51.7	66.1	80.9	59.0	57.3	62.4	46.6	49.6	58.3
Lee *et al*. [18] ECCV'18 (T=3)	40.2	49.2	47.8	52.6	50.1	75.0	50.2	43.0	55.8	73.9	54.1	55.6	58.2	43.3	43.3	52.8
Pavllo *et al*. [33] CVPR'19 (T=1)	47.1	50.6	49.0	51.8	53.6	61.4	49.4	47.4	59.3	67.4	52.4	49.5	55.3	39.5	42.7	51.8
Pavllo *et al*. [33] CVPR'19 (T=9)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	49.8
Cai *et al*. [1] ICCV'19 (T=1)	46.5	48.8	47.6	50.9	52.9	61.3	48.3	45.8	59.2	64.4	51.2	48.4	53.5	39.2	41.2	50.6
Cai *et al*. [1] ICCV'19 (T=7)	44.6	47.4	45.6	48.8	50.8	59.0	47.2	43.9	57.9	61.9	49.7	46.6	51.3	37.1	39.4	48.8
Ours, 1-frame	40.6	47.1	45.7	46.6	50.7	63.1	45.0	47.7	56.3	63.9	49.4	46.5	51.9	38.1	42.3	49.2
Ours, 7-frames	38.2	44.4	42.8	43.7	47.6	60.3	42.0	45.4	53.2	60.8	46.4	43.5	48.5	34.6	38.6	46.3
Ours, 9-frames	37.4	43.5	42.7	42.7	46.6	59.7	41.3	45.1	52.7	60.2	45.8	43.1	47.7	33.7	37.1	45.6

Protocol 2: PA-MPJPE	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg
Sun *et al*. [41] ICCV'17 (T=1)	42.1	44.3	45.0	45.4	51.5	53.0	43.2	41.3	59.3	73.3	51.0	44.0	48.0	38.3	44.8	48.3
Fang *et al*. [9] AAAI'18 (T=1)	38.2	41.7	43.7	44.9	48.5	55.3	40.2	38.2	54.5	64.4	47.2	44.3	47.3	36.7	41.7	45.7
Pavlakos *et al*. [30] CVPR'18 (T=1)	34.7	39.8	41.8	38.6	42.5	47.5	38.0	36.6	50.7	56.8	42.6	39.6	43.9	32.1	36.5	41.8
Hossain & Little [37] ECCV'18 (T=5)	35.7	39.3	44.6	43.0	47.2	54.0	38.3	37.5	51.6	61.3	46.5	41.4	47.3	34.2	39.4	44.1
Pavllo *et al*. [33] CVPR'19 (T=1)	36.0	38.7	38.0	41.7	40.1	45.9	37.1	35.4	46.8	53.4	41.4	36.9	43.1	30.3	34.8	40.0
Cai *et al*. [1] ICCV'19 (T=1)	36.8	38.7	38.2	41.7	40.7	46.8	37.9	35.6	47.6	51.7	41.3	36.8	42.7	31.0	34.7	40.2
Cai *et al*. [1] ICCV'19 (T=7)	35.7	37.8	36.9	40.7	39.6	45.2	37.4	34.5	46.9	50.1	40.5	36.1	41.0	29.6	33.2	39.0
Ours, 1-frame	33.6	37.4	37.0	37.6	39.2	46.4	34.3	35.4	45.1	52.1	40.1	35.5	42.1	29.8	35.3	38.9
Ours, 7-frames	31.7	35.3	35.0	35.3	36.9	44.2	32.0	33.8	42.5	49.3	37.6	33.4	39.6	27.6	32.5	36.7
Ours, 9-frames	31.0	34.8	34.7	34.4	36.2	43.9	31.6	33.5	42.3	49.0	37.1	33.0	39.1	26.9	31.9	36.2

Table 2: Quantitative comparisons of Mean Per Joint Position Error (MPJPE) in millimeter between the estimated pose and the ground-truth on Human3.6M under P1 and P2, where T denotes the number of input frames used in each method. Lower is better and best is bold highlighted.

Protocol 2	Walk			Jog			Box		
PA-MPJPE	S1	S2	S3	S1	S2	S3	S1	S2	S3
Pavlakos *et al*. [31]	22.3	19.5	29.7	28.9	21.9	23.8	–	–	–
Pavlakos *et al*. [30]	18.8	12.7	**29.2**	23.5	15.4	14.5	–	–	–
Lee *et al*. [18]	18.6	19.9	30.5	25.7	16.8	17.7	42.8	48.1	53.4
Pavllo *et al*. [33]	13.9	10.2	46.6	20.9	13.1	13.8	23.8	33.7	32.0
Ours,9-frames	**13.2**	**10.2**	29.9	**12.6**	**12.3**	**13.0**	**13.2**	**18.1**	**20.4**

Table 3: Prediction accuracy on HumanEva Dataset [39] in terms of Protocol # 2 evaluation. Note that we train one model with all three actions (*i.e*., Walk, Jog and Box) models. Lower is better and best is bold highlighted.

in Sec. 3.3. We adopt the same strategy for BN momentum decay as in [33]. Adam Optimizer [17] is used for all modules. The whole model is trained with 200 epoches.

4. Experiments

4.1. Datasets & Evaluation Metrics

Human3.6M Datasets [15]. In our work, we follow the experimental setup in previous researches [10, 45, 33]. More specifically, we use Subject 1 / 5 / 6 / 7 / 8 for training and Subject 9 / 11 for testing. Without access to action labels and camera parameters, all video sequences are used for training one single model.

	MPJPE(P1)	PA-MPJPE(P2)
Baseline	51.8	40.0
2D Refine	49.9	39.4
2D Refine + 3D Decompose	47.1	37.5
2D Refine + 3D Decompose + 3D Completion	45.6	36.2

Table 4: Ablation study on the contribution of proposed three modules in terms of both P1 and P2. Note that Baseline refers to single-frame results of Pavllo *et al*. [33].

HumanEva-I Datasets [39]. Compared to Human3.6M dataset [15], HumanEva-I [39] is more light-weight containing three erect actions: Walk, Jog, Box. We follow the same data preprocessing strategy used in [33] for train/test split. We report the estimation accuracy with $T = 9$.

Evaluation Metrics. Following the majority of previous works [24, 37, 18, 1, 33] we evaluate our model in terms of mean per joint position error (MPJPE, P1 for short) commonly denoted as protocol #1. Several researches [41, 9, 30, 37, 33, 1] estimate 3D pose after alignment involving rotation and translation (PA-MPJPE, P2 for short), which is termed as protocol #2. Both protocols are utilized in our work.

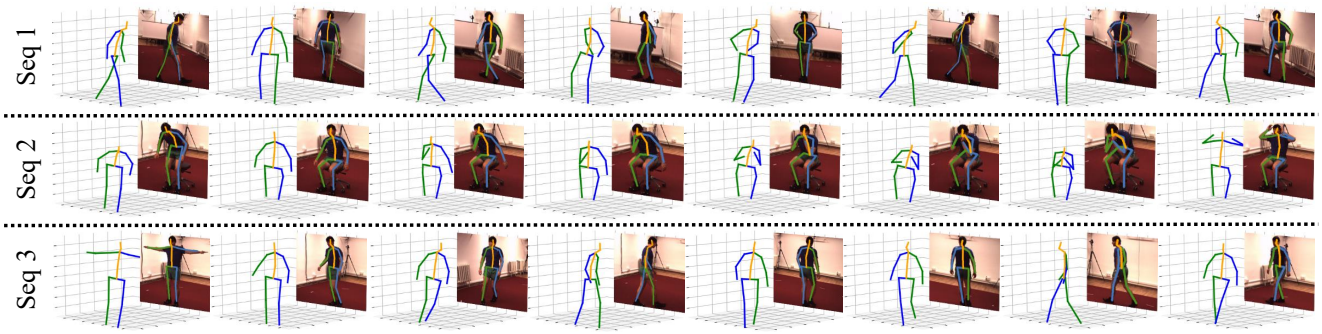

Figure 6: Visualization of predicted 3D poses on monocular video clip. Each row corresponds to one action sequence.

4.2. Quantitative Evaluation

Results on Human3.6M Dataset [15]. As shown in Tab. 2, we report pose estimation results in terms of Protocol #1 and #2. Note that previous works evaluate this task with different temporal length. For fairness we compare with them under the same inputs. When $T = 1$ trajectory completion model ϕ_{3D}^{ctn} is nonfunctional. Benefited from 2D correction and 3D decomposition, our model outperforms prior art (Cai *et al*. [1]) under both P1 and P2 evaluation when $T = 1$. Boosted by explicit kinematics analysis, the estimation accuracy outperforms Pavllo *et al*. [33] by 8.4%(4.2mm) when $T = 9$. A notable performance improvement over Cai *et al*. [1] is also presented for $T = 7$, *i.e*., 46.3mm v.s. 48.8mm. Regarding evaluation with action classes, our final model ($T = 9$) achieves best performance on the majority of them. Especially on several relatively harder actions, e.g, sitting and sitting down with severe occlusion, our model is robust enough to obtain better results compared to Pavllo *et al*. [33] which merely considers temporal smoothness rather than estimation reliability. This is further validated in our own model. From $T = 1$ to $T = 9$, estimation accuracy is enhanced constantly under both P1 and P2 evaluation, which is mainly facilitated by decomposed 3D pose estimation and 3D trajectory completion.

Results on HumanEva Dataset [39]. As shown in Tab. 3, we report pose estimation results in terms of Protocol #2. Compared to Human3.6M Dataset [15], HumanEva Dataset [39] is relatively easier to learn, where estimation accuracy is near saturated. Still, performance gain is observed for all three actions over Pavllo *et al*. [33]. By explicitly incorporating kinematics analysis into deep models, reasonable spatial-temporal structure is well preserved and output space is more compact, which finally lead to higher estimation accuracy.

4.3. Qualitative Evaluation

We further present direct visualization results of monocular 3D pose estimation. As illustrated in Fig. 6, three sequences with diverse actions are presented. Meanwhile, for each time stamp we demonstrate both refined 2D pose and corresponding 3D prediction. Our model makes it to produce visually natural estimation which is mainly benefited from explicit kinematics constraint. For example, sitting down sequence (second row in Fig. 6) is well estimated for both 2D and 3D joints, where the structure of four limbs is properly handled by our model (requiring awareness of human topology). More visual results are provided in supplementary material for reference.

	1-Frame	3-Frames	5-Frames	7-Frames	9-Frames
Pavllo *et al*. [33]	51.80	–	–	–	49.80
Cai *et al*. [1]	50.62	49.08	48.86	48.78	–
Ours	49.21	47.87	46.83	46.26	45.61

Table 5: Prediction accuracy with different input length and in terms of MPJPE. We compare our model with Pavllo *et al*. [33] and Cai *et al*. [1].

4.4. Ablation Study

Analysis on all modules. Recall that our model consists of three modules: ϕ_{2D}, ϕ_{3D}^{dcm} and ϕ_{3D}^{ctn}. To validate the contribution of each module, we present corresponding ablation study on Human3.6M Dataset [15]. As shown in Tab. 4, both accuracy on P1 and P2 are reported. Baseline refers to single frame estimation model of Pavllo *et al*. [33]. We can notice that each module offers positive contribution under evaluation of P1 and P2. Notably, the most significant improvement comes from 3D decomposition module ϕ_{3D}^{dcm}, which benefits from ϕ_{2D} with more reasonable 2D/3D correspondence and further leads to more compact output space.

Analysis on temporal length. As shown in Tab. 5, we report the estimation accuracy w.r.t. different input lengths. We can notice that with the increase of temporal horizon, our model constantly performs better than those with shorter inputs. For the identical input length, our model also produces more accurate results than prior arts, *i.e*., Pavllo *et al*. [33] and Cai *et al*. [1].

Analysis on projection based 2D correction. One remaining but critical problem is: how does projection based

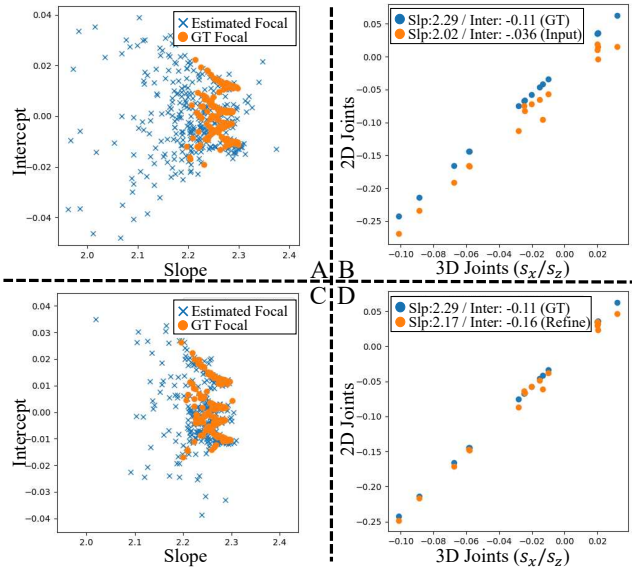

Figure 7: Analysis on projection based 2D correction. A/C correspond to f_x, c_x estimation without/with constraint of perspective projection. B/D correspond to two typical examples of projection correspondence between 2D and 3D joints without/with constraint of perspective projection.

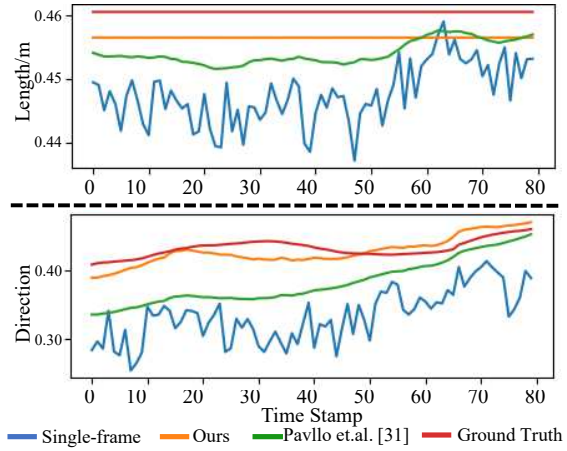

Figure 8: Analysis on decomposed 3D estimation. The length (upper part) and direction (lower part) estimation of left shank (skeleton between joint 4 and joint 5) are presented.

2D correction facilitate 3D pose estimation? As shown in Fig 7(A and C), we plot estimated/ground thruth focal length f_x and point c_x on test set for evaluation. Fig 7A corresponds to training without $\mathcal{L}_{2D}^{proj}$ while Fig 7C shows results trained with it. Orange dot corresponds to ground truth and blue cross are estimated results. Boosted by $\mathcal{L}_{2D}^{proj}$, the estimated projection structure in camera view is more accurate. Moreover, we present two typical examples of correspondence between 2D and 3D joints in Fig. 7 (B and D). Following Eqn. 2, X-axis represents $\mathbf{x}/\mathbf{z}$ while Y-axis is $\mathbf{a}$. Notation is used the same as Fig. 7 (A and C). All joints belong to one single frame. Orange dots referring ground truth lie in a straight line. Similarly, Fig. 7D depicts test results trained with $\mathcal{L}_{2D}^{proj}$ while Fig. 7B not. It clearly shows that 2D correction based on projection correspondence is more reasonable and accurate.

	1-Frame	3-Frames	5-Frames	7-Frames	9-Frames
Full model w/o ϕ_{3D}^{ctn}	7.80	5.11	4.03	3.57	3.29
Full model	7.80	3.70	2.45	2.13	2.01

Table 6: Prediction accuracy (Human3.6M Dataset [15]) with different input length in terms of MPJVE [33]. The first row corresponds to training without trajectory model ϕ_{3D}^{ctn}, while the second row refers to full model. Note that ϕ_{3D}^{ctn} is nonfunctional when $T = 1$, whose estimation accuracy is identical to the first row.

Analysis on decomposed 3D estimation. To analyze the contribution of ϕ_{3D}^{dcm}, we present both length and direction results shown in Fig. 8. Red line corresponds to ground truth, blue line is single-frame estimation, green line refers to the results of Pavllo *et al*. [33] and orange line is ours. For length estimation, our model is robust to the variation of 2D poses. One the contrary, both single-frame estimation and the model of Pavllo *et al*. [33] fails to produce valid length estimation which should keep constant throughout the whole video clip. Based on accurate length estimation, our model is capable of finding joint angle within more compact space. As shown in Fig. 8B where the direction is calculated as angle between left shank and Y-axis, our model performs better than all other baselines by a large margin, i.e, estimated direction is closer to ground truth (red line) with smaller variation.

Analysis on 3D trajectory completion. Following Pavllo *et al*. [33], we evaluate trajectory estimation accuracy in terms of MPJVE, *i.e*., velocity error, to further validate the contribution of ϕ_{3D}^{ctn}. As shown in Tab. 6, facilitated by ϕ_{3D}^{ctn} our model achieves lower velocity error with all experimented temporal lengths (from $T = 3$ to $T = 9$).

5. Conclusion

In this paper, we propose a deep kinematics analysis framework for monocular 3D pose estimation. By explicitly incorporating kinematics regularization into deep models, we achieves more reliable estimation with noisy 2D joints as inputs. Extensive experiments show that our model achieves state-of-the-art performance on two widely used 3D human action datasets.

Acknowledgment This work was supported by National Science Foundation of China (61976137, 61527804, U1611461, U19B2035), STCSM(18DZ1112300). This work was also supported by National Key Research and Development Program of China (2016YFB1001003). The authors would like to give a personal thanks to the Student Innovation Center of SJTU for providing GPUs.

References

[1] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *ICCV*, October 2019.

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, pages 1302–1310, 2017.

[3] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017.

[4] Ching-Hang Chen and Deva Ramanan. 3d human pose estimation= 2d pose estimation+ matching. In *CVPR*, pages 7035–7043, 2017.

[5] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, pages 7103–7112, 2018.

[6] Yu Cheng, Bo Yang, Bo Wang, Wending Yan, and Robby T. Tan. Occlusion-aware networks for 3d human pose estimation in video. In *ICCV*, October 2019.

[7] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing Network Structure for 3D Human Pose Estimation. In *ICCV*, October 2019.

[8] Huseyin Coskun, Felix Achilles, Robert DiPietro, Nassir Navab, and Federico Tombari. Long short-term memory kalman filters: Recurrent neural estimators for pose regularization. In *ICCV*, pages 5524–5532, 2017.

[9] Hao-Shu Fang, Yuanlu Xu, Wenguan Wang, Xiaobai Liu, and Song-Chun Zhu. Learning pose grammar to encode human body configuration for 3d pose estimation. In *AAAI*, 2018.

[10] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Gerard Pons-Moll, and Christian Theobalt. In the wild human pose estimation using explicit 2d features and intermediate 3d representations. In *CVPR*, pages 10905–10914, 2019.

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *YCCV*, pages 2980–2988, 2017.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *NC*, 9(8):1735–1780, 1997.

[13] Zhongyue Huang, Jingwei Xu, and Bingbing Ni. Human motion generation via cross-space constrained sampling. In *IJCAI*, pages 757–763, 2018.

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[15] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 36(7):1325–1339, 2014.

[16] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018.

[17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[18] Kyoungoh Lee, Inwoong Lee, and Sanghoon Lee. Propagating lstm: 3d pose estimation based on joint interdependency. In *ECCV*, pages 119–135, 2018.

[19] Sijin Li and Antoni B Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *ACCV*, pages 332–347, 2014.

[20] Jiahao Lin and Gim Hee Lee. Trajectory space factorization for deep video-based 3d human pose estimation. *arXiv preprint arXiv:1908.08289*, 2019.

[21] Mude Lin, Liang Lin, Xiaodan Liang, Keze Wang, and Hui Cheng. Recurrent 3d pose sequence machines. In *CVPR*, pages 810–819, 2017.

[22] Jinxian Liu, Bingbing Ni, Caiyuan Li, Jiancheng Yang, and Qi Tian. Dynamic points agglomeration for hierarchical point sets learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7546–7555, 2019.

[23] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C. Kot. Global context-aware attention LSTM networks for 3d action recognition. In *CVPR*, pages 3671–3680, 2017.

[24] Diogo C Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *CVPR*, pages 5137–5146, 2018.

[25] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, pages 2640–2649, 2017.

[26] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3DV*, pages 506–516, 2017.

[27] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *TOG*, 36(4):44, 2017.

[28] Francesc Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *CVPR*, pages 2823–2832, 2017.

[29] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS*, 2017.

[30] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *CVPR*, pages 7307–7316, 2018.

[31] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*, pages 7025–7034, 2017.

[32] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *CVPR*, pages 459–468, 2018.

[33] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3D human pose estimation in video with temporal convolutions and semi-supervised training. In *CVPR*, pages 7753–7762, 2019.

[34] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: example-guided deep reinforcement learning of physics-based character skills. *ACM Trans. Graph.*, 37(4):143:1–143:14, 2018.

[35] Tomas Pfister, James Charles, and Andrew Zisserman. Flowing convnets for human pose estimation in videos. In *ICCV*, pages 1913–1921, 2015.

[36] Huy Hieu Pham, Houssam Salmane, Louahdi Khoudour, Alain Crouzil, Pablo Zegers, and Sergio A Velastin. A unified deep framework for joint 3d pose estimation and action recognition from a single rgb camera. *arXiv preprint arXiv:1907.06968*, 2019.

[37] Mir Rayat Imtiaz Hossain and James J Little. Exploiting temporal information for 3d human pose estimation. In *ECCV*, pages 68–84, 2018.

[38] Saurabh Sharma, Pavan Teja Varigonda, Prashast Bindal, Abhishek Sharma, and Arjun Jain. Monocular 3d human pose estimation by generation and ordinal ranking. In *ICCV*, October 2019.

[39] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *IJCV*, 87(1-2):4–27, 2010.

[40] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.

[41] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *ICCV*, pages 2602–2611, 2017.

[42] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. Structured prediction of 3d human pose with deep neural networks. In *BMVC*, 2016.

[43] Bugra Tekin, Pablo Márquez-Neila, Mathieu Salzmann, and Pascal Fua. Learning to fuse 2d and 3d image cues for monocular body pose estimation. In *ICCV*, pages 3941–3950, 2017.

[44] Bugra Tekin, Artem Rozantsev, Vincent Lepetit, and Pascal Fua. Direct prediction of 3d body poses from motion compensated sequences. In *CVPR*, pages 991–1000, 2016.

[45] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *CVPR*, pages 7782–7791, 2019.

[46] Jue Wang, Shaoli Huang, Xinchao Wang, and Dacheng Tao. Not all parts are created equal: 3d pose estimation by modeling bi-directional dependencies of body parts. In *ICCV*, October 2019.

[47] Yichao Yan, Bingbing Ni, Wendong Zhang, Jingwei Xu, and Xiaokang Yang. Structure-constrained motion sequence generation. *IEEE Trans. Multimedia*, 21(7).

[48] Yichao Yan, Jingwei Xu, Bingbing Ni, Wendong Zhang, and Xiaokang Yang. Skeleton-aided articulated motion generation. In *ACM MM*, pages 199–207, 2017.

[49] Y. Yan, N. Zhuang, b. ni, J. Zhang, M. Xu, Q. Zhang, Z. ZHENG, S. Cheng, Q. Tian, y. xu, X. Yang, and W. Zhang. Fine-grained video captioning via graph-based multi-granularity interaction learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019.

[50] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. Modeling point clouds with self-attention and gumbel subset sampling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3323–3332, 2019.

[51] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning feature pyramids for human pose estimation. In *ICCV*, pages 1290–1299, 2017.

[52] Hashim Yasin, Umar Iqbal, Bjorn Kruger, Andreas Weber, and Juergen Gall. A dual-source approach for 3d pose estimation from a single image. In *CVPR*, pages 4948–4956, 2016.

[53] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *CVPR*, pages 3425–3435, 2019.

[54] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation. In *ICCV*, October 2019.

[55] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *CVPR*, pages 4966–4975, 2016.