Uncertainty-Aware 3D Human Pose Estimation from Monocular Video

Jinlu Zhang
jinluzhang@whu.edu.cn
Wuhan University

Yujin Chen
yujin.chen@tum.de
Technical University of Munich

Zhigang Tu*
tuzhigang@whu.edu.cn
Wuhan University

ABSTRACT

Estimating the 3D human pose from the monocular video is challenging mainly due to the depth ambiguity and inaccurate 2D detected keypoints. To quantify the depth uncertainty of 3D human pose via the neural network, we imbue the uncertainty modeling to depth prediction by using evidential deep learning (EDL). Meanwhile, to calibrate the distribution uncertainty of the 2D detection, we explore a probabilistic representation to model the realistic distribution. Specifically, we exploit the EDL to measure the *depth* prediction uncertainty of the network, and decompose the $x-y$ coordinates into individual distributions to model the deviation uncertainty of the inaccurate 2D keypoints. Then we optimize the depth uncertainty parameters and calibrate the 2D deviations to obtain accurate 3D human poses. Besides, to provide effective latent features for uncertainty learning, we design an encoder which combines graph convolutional network (GCN) and transformer to learn discriminative spatio-temporal representations. Extensive experiments are conducted on three benchmarks (*Human3.6M, MPI-INF-3DHP, and HumanEva-I*) and the comprehensive results show that our model surpasses the state-of-the-arts by a large margin.

CCS CONCEPTS

• **Computing methodologies** → **Motion capture**.

KEYWORDS

3D human pose; uncertainty learning

ACM Reference Format:
Jinlu Zhang, Yujin Chen, and Zhigang Tu. 2022. Uncertainty-Aware 3D Human Pose Estimation from Monocular Video. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3503161.3547773

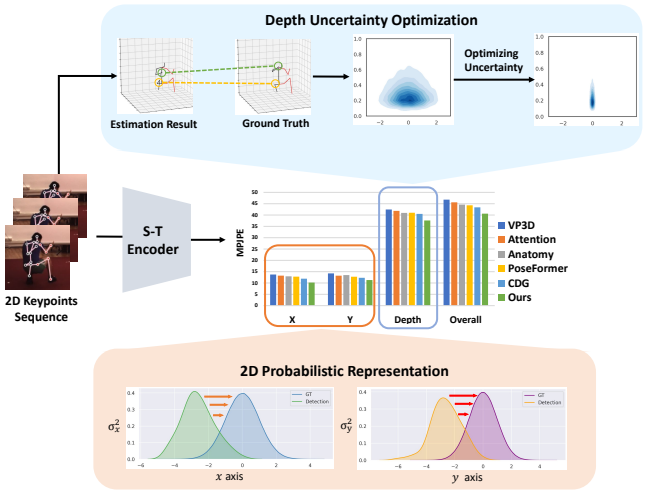

Figure 1: The *depth* and $x-y$ coordinates have different types of errors (depth ambiguity and noisy 2D data), we optimize them in terms of depth uncertainty (blue area) and 2D probabilistic representation (orange area), respectively. We compare our approach with the state-of-the-art 2D-to-3D lifting methods (VP3D [43], Attention [31], Anatomy [5], PoseFormer [63], and CDG [17]) from the perspective of different coordinates. The spatial-temporal (S-T) encoder is applied to model the spatial-temporal relationships in video.

1 INTRODUCTION

3D human pose estimation aims at reconstructing the coordinates of 3D human body joints from images or detected 2D keypoints. This task can be applied to a wide range of applications, such as skeleton action recognition [55, 61], motion retargeting [1], and human animation [57]. In recent years, this field has achieved notable progress, many solutions [4, 5, 20, 27, 31, 35, 43, 51, 60, 63], which under a 2D-to-3D lifting pipeline, have been exploited to detect 2D keypoints from images and lift them to obtain 3D coordinates.

The 2D-to-3D lifting pipeline can imbue 2D keypoints as powerful intermediate representations and produce reasonable 3D predictions, however, it has two serious issues. First, due to the lack of depth information in the 2D detection and inherent epistemic uncertainty of neural networks, the predictions are not always reliable. Consequently, if the model can know whether its outputs are reliable, the prediction could be optimized and more discriminative for downstream applications. Second, the commonly used keypoints representations are defined with human experience, but there are no such absolute correct positions. Especially for the 2D keypoints, which are captured from a single view, even the state-of-the-art detectors still produce noise 2D keypoints. Although some works [7, 52, 59] have considered the uncertainty of the noisy data and modeled it during training, the distribution deviation of the detected 2D data is ignored, which needs to be explicitly calibrated.

*Corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547773

Test Set \ Train Set	SH [39]	CPN [9]	HRNet [46]	2D GT with noise
SH [39]	56.1	56.8	58.3	67.3
CPN [9]	67.2	44.6	48.7	56.6
HRNet [46]	64.7	49.9	42.4	55.4
2D GT with noise	54.2	39.4	38.1	31.9

Table 1: Trained and evaluated the previous SOTA method [63] on different input 2D data. We apply SH [39], CPN [9], and HRNet [46] to obtain different 2D keypoints. The 2D ground truth (GT) with noise is obtained by projecting the 3D poses to 2D space and then adding the Gaussian noise to simulate the input data noise. The blue color indicates training and testing on the same detector data. The performance is better when the MPJPE is lower.

Our key observation is that these two issues of the current 2D-to-3D lifting methods lead to unreliable estimation results of neural networks during inference, and these issues are supposed to be addressed respectively.

To learn more valid depth information from monocular input, most previous works tried to utilize the temporal information of video frames by introducing various networks, such as temporal convolution network (TCN) [5, 31, 43], Transformer [26, 27, 60, 63], and graph convolution network (GCN) [4, 59]. The uncertainty of depth prediction has been considered implicitly by just adding more data augmentations [15, 58] or generating diverse hypotheses of feasible 3D poses [27, 52], while there is no method that directly estimates an exact representation to imitate the depth uncertainty.

As for the 2D detected keypoints, we find that there are different deviations among different 2D detectors, leading to distribution uncertainty of the 2D detection keypoints. It is caused by the internal distribution of different pre-trained detectors and degrades the estimation accuracy of $x-y$ coordinates. As shown in Table 1, using different 2D detected keypoints sets as the input of the 2D-to-3D lifting network results in biased 3D predictions. The estimated 3D results are positively correlated with the 2D keypoints set from the same detector while having a large degradation on test sets from other detectors. The results evaluated on the 2D ground truth (GT) have different degrees of performance improvement. Table 1 demonstrates that there are different deviations caused by different 2D detectors, and we summarize this phenomenon as 2D distribution uncertainty. However, it is often regarded as the aleatoric noise in the 3D human pose estimation task without being specially optimized. To our knowledge, there are no methods exploring to model the 2D distribution uncertainty by calibrating deviations between the input detections and the 2D GT.

Motivated by the above observations, we propose an uncertainty-aware 3D human pose estimation method, where the depth uncertainty and the 2D distribution uncertainty are modeled separately (see Figure 1). Firstly, we present a novel evidential pose estimation module to quantify the depth uncertainty of predictions during training. Specifically, we utilize a decoder to obtain the high-order evidential parameters of the estimated output. And then, we compute the prediction and determine whether it is reliable (called *evidence*) by using the decoded parameters and GT. The area of the blue region at the top-right of Figure 1 visualizes the optimization process of the depth uncertainty. With less area of blue region, the depth uncertainty on model prediction becomes lower. In this way, the uncertainty on depth prediction can be estimated and subsequently optimized by minimizing the well-designed loss function.

Secondly, the distribution uncertainty from the 2D keypoint detector is modeled by the explored probabilistic representation. We exploit different parameters of Gaussian prior distribution to represent the uncertainty of the input 2D keypoints. The parameters of the input $x-y$ coordinates are obtained by the distribution uncertainty decoder. Then the deviations of the 2D detected input could be calibrated according to the 2D annotations projected from the 3D GT. Since the 2D GT contains aleatoric uncertainty due to unavoidable annotation mistakes, we project the distribution parameters to a 1D discrete vector rather than directly supervising the parameter regression to avoid obtaining an absolute prior distribution. The proposed uncertainty-aware method on depth and 2D detection input is light-weight and plug-and-play, which means it can be easily incorporated into other 2D-to-3D methods.

Finally, we design a spatial-temporal encoder that combines the GCN [22] and transformer [50]. GCN is widely used in skeleton-based visual tasks [4, 55], which has good interpretability for spatial human joints structure. Therefore, we introduce GCN to keypoints embedding and output regression to better preserve the spatial correlation of joints. On the other side, transformer is applied to model the temporal relationships among input frames to improve the smoothness and reduce jitters of the output pose sequence. The ability of global sequence modeling enables the model to capture the frame-to-frame interrelationships of the input keypoints.

We evaluate the proposed method on three public benchmarks, *i.e.* Human3.6M [18], MPI-INF-3DHP [36], and HumanEva-I [45]. The quantitative and qualitative experimental results show that our method is effective to improve the performance of 3D human pose estimation. Our contributions can be summarized in three-fold:

- We propose an uncertainty-aware method to quantify and optimize the depth and 2D detection input respectively, which improves the performance on depth ambiguity and 2D keypoint errors for 3D human pose estimation.
- Evidential deep learning is introduced to quantify the prediction uncertainty of depth, and a probabilistic representation is exploited to model the distribution uncertainty of the 2D detected input.
- An effective encoder based on GCN and transformer is proposed to better model the spatial-temporal correlation of human joints.

2 RELATED WORK

3D human pose estimation in video. With the development of deep learning, many data-driven methods has shown remarkable progress. According to whether 2D human pose is represented in the 2D image space, these methods can be divided into end-to-end and 2D-to-3D lifting pipelines. The end-to-end pipelines directly regress the 3D poses from the input images with large model parameters and difficulties. There are approaches [8, 10, 42, 47, 48] followed this pipeline in the early stage. And the 2D-to-3D lifting pipeline [12, 30, 33, 35, 54, 62, 64] overcomes the issues by first estimating 2D keypoints in the RGB observations, benefiting from the outstanding performance of 2D keypoint detectors [9, 39,

46]. Therefore, we follow the 2D-to-3D lifting pipeline and apply our method to the video sequence. For the 3D human pose estimation task, the depth ambiguity challenge is commonly discussed, and there are two main solutions to solve this issue: temporal-based methods and generative-based methods. The first solution [5, 31, 43, 49] prefers to utilize the temporal information to smooth the output pose sequence. And the second solution [27, 52] generates many potential pose proposals to model the uncertainty of the depth ambiguity. However, there are no methods considering to estimate and this uncertainty during model training. Therefore, different from the above 3D pose estimation methods, we introduce the proposed uncertainty-aware method to model depth and 2D detection input uncertainty in 3D human pose.

Uncertainty learning for depth ambiguity. Uncertainty learning aims to estimate the uncertainty representation of a determined neural network. It is important for the tasks where the data sources are highly inhomogeneous or rare. In recent years, many research works have shown an increased interest in estimating uncertainty in deep neural networks (DNNs) [2, 3, 34, 44]. Some methods based on Bayesian deep learning introduced the uncertainty estimation in kinds of ways: variational inference and dropout [14, 21, 38], ensemble [23]. These methods rely on the expensive samples to estimate predictive variance [2]. The evidential deep learning overcomes above drawback and has been applied to many tasks *e.g.*, action recognition [3] and classification [44]. It directly predicts the related parameters of uncertainty without repeat sampling. We introduce the evidential deep learning method [2] and then design a more concise constraint for 3D human pose task to better model depth uncertainty. We estimate the uncertainty of model prediction on depth and then optimize it to enable our model to predict the depth uncertainty and improve the accuracy of estimation results.

Representations of human pose estimation. There are different kinds of ways to represent the human pose. Most of recent 3D pose works [6, 11, 31, 43, 63] obtained 3D joint locations directly from the neural network. It is simple and intuitive but lacks much prior, *e.g.*, human skeleton structure and probabilistic distribution of input data. Some researchers [5] proposed to decompose the 3D pose estimation task into bone direction prediction and bone length prediction. And many others [54, 59, 62] utilized the GCN [22] to model the human skeleton from the perspective of graph connectivity. Besides, the probabilistic distribution of input data is another effective way to represent human pose. And there have been many 2D human pose works [13, 24, 28, 53] to apply the heatmap as main representation of keypoints. It enables network to better learn location and corresponding probabilistic information. However, this representation of input data has been absent. Our method is similar to the pipeline of constructing the heatmap in the 2D human pose task, but we decompose the $x-y$ coordinates rather than making a two-dimensional distribution, and our input is keypoints rather than images, which helps to reduce much computing cost. Besides, we perform it on input keypoints instead image space.

3 OUR METHOD

The pipeline of the proposed method is illustrated in Figure 2. Our method enables to estimate 3D human pose with uncertainty information from the monocular video end-to-end. The input 2D keypoints are taken into the model and reconstructed to a sequence of 3D poses (*seq2seq*). Specifically, first, we use the 2D detector to produce a sequence of 2D keypoints $K = \{X_{i,j} \in \mathbb{R}^2, i \in N, j \in T\}$ in the image space, where N and T indicate the joints of the defined human skeleton and the number of frames of the input video. We take the input keypoints K to the spatial embedding layer to learn the initial relationships of joints. The embedding layer is exploited with a single GCN. Second, the spatial GCN module and the temporal transformer module are used to model the high dimensional feature in the spatial and temporal domains, respectively. The depth and 2D uncertainty are modeled in different decoders and optimized by loss functions to enable the backbone to learn the uncertainty information accurately. Last, the spatial regression head obtains the output 3D pose sequence $Z \in \mathbb{R}^{N\times T\times 3}$ from the feature $Z \in \mathbb{R}^{N\times T\times d}$, where d is the hidden dimension of the backbone. The regression head is constructed by a GCN layer to maintain the spatial relationship of the skeleton.

In brief, we introduce the uncertainty of depth prediction in subsection 3.1 and 2D detection keypoints in subsection 3.2, respectively. The spatial-temporal encoder is described in subsection 3.3.

3.1 Depth uncertainty estimation

3.1.1 ***Background derivation.*** Different from the ordinary probability, we introduce the subjective logic to measure the uncertainty of the model. Considering the training phase of pose estimation in depth dimension, the two-norm of errors (MPJPE or P-MPJPE) is considered as the optimization item normally. It can be formally written as:

$$\mathcal{L}_{norm} = \|\hat{y} - f(x, W)\|^2, \tag{1}$$

where $\hat{y}$ indicates the prediction of input data x, W and f are the parameter weights and the mapping function of neural network. The network is enabled to fit the correct prediction results and learn the distribution feature of the given training dataset. However, the model can not model the uncertainty of depth ambiguity because each prediction is seen as absolute correct during inference. To imbue the uncertainty of the model prediction on depth information, we assume the depth data following Gaussian distribution, which can be represented as $x \sim \mathcal{N}\left(\mu, \sigma^2\right)$. With the unknown μ and σ^2, we can place the Gaussian distribution on μ and inverse-chi-squared distribution on σ^2 based on probability theory:

$$(\mu \mid \sigma^2) \sim \mathcal{N}(\mu_0, \frac{\sigma^2}{k}), \sigma^2 \sim \text{Inv}-\chi^2\left(\alpha, \beta\right)), \tag{2}$$

where IG is inverse-chi-squared distribution, $(\mu_0, k, \alpha, \beta)$ indicate relative high-order parameters, and the μ_0 is the locations of poses. Therefore, the expectation and variance of unknown μ can be computed as:

$$\mathbb{E}[\mu] = \mu_0, \text{Var}[\mu] = \frac{\beta}{k(\alpha-1)}, \tag{3}$$

where the $\mathbb{E}[\mu]$ indicates the prediction of depth, and the $\text{Var}[\mu]$ represents the model uncertainty. More detailed derivations are shown in the supplementary materials. Based on the above derivations, we can obtain the depth uncertainty high-order parameters $(\mu_0, k, \alpha, \beta)$.

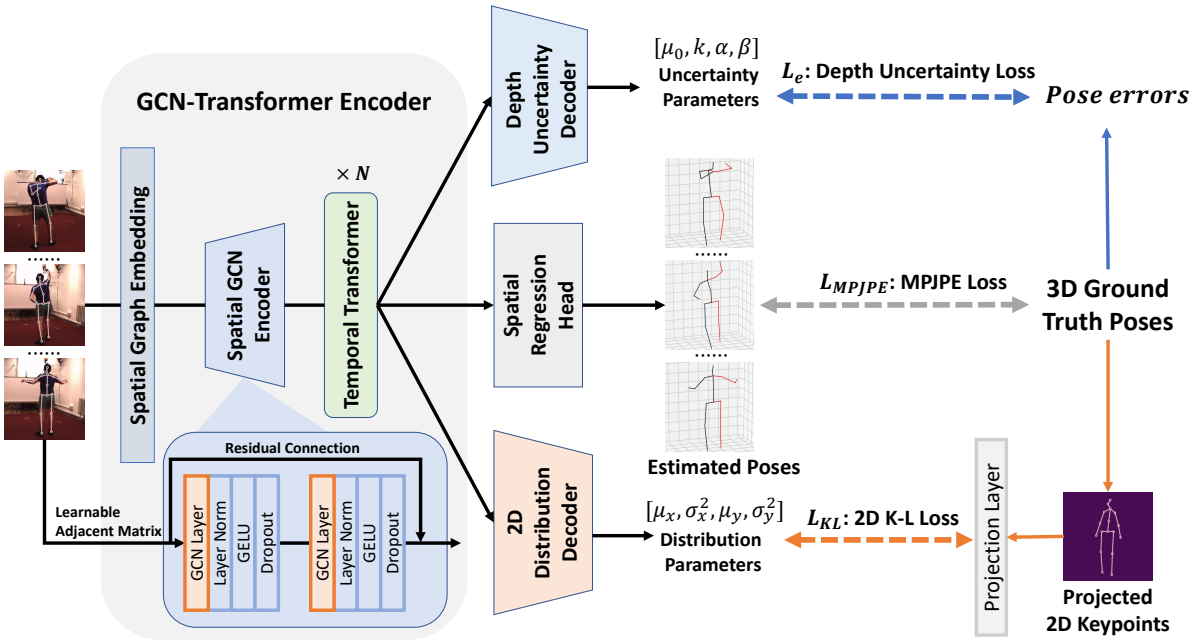

Figure 2: Overview of the proposed method. We employ the GCN-Transformer encoder to model the spatial-temporal relationships in input keypoints sequence. The encoder consists of GCN for spatial prior and transformer for temporal domain. Then the uncertainty parameters of *depth* and $x-y$ are decoded, respectively. The parameters are supervised by different loss functions to enable the model to learn uncertainty information and calibrate input distribution deviation. The MPJPE under Protocol #1 is the main metric during training.

3.1.2 ***Evidential 3D pose estimation***. In this paper, we propose a novel uncertainty estimation method to formulate the 3D human pose task from the evidential deep learning (EDL) perspective. To concisely and effectively model uncertainty, we design the constraints as follow. We first need to imbue the above distributions to encoder training to learn the probabilistic modeling. To this end, we construct the loss item for parameters (k, α, β) of uncertainty representation as $\left|\frac{\beta}{k(\alpha-1)} - \mathrm{Var}[\hat{\mu}]\right|$, where $\hat{\mu}$ indicates the depth coordinates of samples. Besides, in regression problems, due to the regression space is infinite and unbounded, we have to design the regularized item for prediction errors. We follow the previous work [2] and define the *evidence* as $(2k+\nu)$. The uncertainty and *evidence* are inversely proportional and they are used to quantify the uncertainty of model prediction. During the training phase, we regress the uncertainty of each keypoint in depth using a simple decoder. And the regression parameters of keypoint are $(\mu_0, k, \alpha, \beta)$. Then we optimize the *evidence* on prediction errors: $(|\hat{\mu} - \mu_0| \cdot (2k+\nu))$, where μ_0 is the prediction, and $|\hat{\mu} - \mu_0|$ indicates the prediction errors in depth. By optimizing *evidence* on pose errors, the encoder can learn to refine the depth prediction. Therefore the final loss of depth uncertainty can be formulated as:

$$\mathcal{L}_e = \left|\frac{\beta}{k(\alpha-1)} - \mathrm{Var}[\hat{\mu}]\right| + (|\hat{\mu} - \mu_0| \cdot (2k+\nu)) \tag{4}$$

In this way, we can not only predict the depth coordinate location μ_0, but also estimate the related parameters (k, α, β) to illustrate the model uncertainty. During inference we can chose to regress the evidential parameters or not because the encoder has already modeled uncertainty of depth prediction implicitly.

3.2 Distribution uncertainty estimation of 2D detection

Given a 2D detection keypoint sequence K with N joints and T frames, we decompose the $x-y$ into two individual representation to learn the distribution of the input data. We assume that there is an approximate Gaussian distribution for x and y of each keypoint, and take the Gaussian distribution as the prior of 2D GT projected from 3D data:

$$G(p|\mu, \sigma^2) = \mathbf{M} \odot \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(p-\mu)^2}{2\sigma^2}\right), p \in \mathbb{R}^s \tag{5}$$

where p is the s size discrete coordinate vector like $(x_o, x_1, ..., x_s)$ or $(y_o, y_1, ..., y_s)$, $\mathbf{M}$ is the learnable projection matrix, μ and σ^2 indicate the coordinate locations and variance of the prior distribution. The parameterized matrix $\mathbf{M}$ enables the model to learn aleatoric noise of GT caused by annotation mistakes during training. We use *Eq.* 5 for both x and y coordinate representation. During training, we apply the 2D distribution decoder to obtain the discrete vectors $\hat{p} \in \mathbb{R}^{N\times 2\times s}$ of each coordinate of the keypoints. The decoder is constructed by a MLP with Layer Norm (LN). We use the Kullback-Leibler (KL) divergence to constrain 2D distributions between the

input data and projected 2D GT, which is formulated as:

$$\begin{aligned}\mathcal{L}_{KL}(\hat{p}\|p) &= -\int \hat{p}(m)\ln p(m)dm - \left(-\int p(m)\ln \hat{p}(m)dm\right) \\ &= -\int \hat{p}(m)\ln\left[\frac{p(m)}{\hat{p}(m)}\right]dm,\end{aligned} \tag{6}$$

where m is input coordinates x or y, and $\hat{p}, p$ are predicted distribution of input data and projected distribution of 2D ground truth. In this way, we can model the uncertainty of input data from the probabilistic perspective.

3.3 GCN-Transformer encoder (GTE)

As shown in Figure 2, the backbone of model consists of two main modules: spatial GCN module and temporal transformer module. The two modules effectively learn the spatial and temporal correlations in video sequence to provide robust feature for uncertainty modeling in Section 3.1 and Section 3.2.

3.3.1 ***Spatial GCN module.*** We exploit the GCN with a learnable parameterized matrix and residual connection based on the vanilla GCN model [22] to model the spatial correlation of human joints. The operation of GCN layer on input Z^l can be formulated as:

$$G_{conv}(X^{l+1}) = \sigma\left(W_j Z^l (W_m \odot D^{-\frac{1}{2}} A D^{-\frac{1}{2}})\right), \tag{7}$$

where $X^{l+1} \in \mathbb{R}^{N\times d_{out}}$ indicates output of l + 1-th layer, $W_j \in \mathbb{R}^{d_{in}\times d_{out}}$ represents the learnable transform matrix with input (d_{in}) and output (d_{out}) channel dimension, and $W_m \in \mathbb{R}^{N\times N}$ indicates the parameter matrix used for learning different weight of predefined skeleton adjacent matrix A, and it is symmetrically normalized.

The spatial graph embedding layer is performed by the proposed GCN when the input Z^l is set to 2D keypoints $K \in \mathbb{R}^{T\times N\times 2}$, where T is paralleled. And the spatial GCN as shown in the Figure 2 is exploited by a residual connection. It enables model to efficiently learn both initial spatial features and high-level features and speed up convergence. The complete spatial GCN module can be written as:

$$Z^{l+2} = Z^l + GELU(LN(G_{conv}(Z^{l+1}))). \tag{8}$$

We choose the GELU function to keep it similar to the temporal module in Section 3.3.2. The designed spatial GCN module is lightweight in computation cost and parameters, and it can be easily extended.

3.3.2 ***Temporal transformer module.*** The temporal transformer module in the proposed encoder is applied to learn temporal correlation across the sequence of frames. We follow the vanilla multi-head attention [50] to obtain global correlations among frames. The multi-head self-attention in temporal transformer performs on frame-level feature, which is the output of spatial GCN module $Z \in \mathbb{R}^{N\times d_{out}}$. We take the paralleled frame dimension in spatial GCN and then make the joints paralleled, the input of module turns out to be $Z \in \mathbb{R}^{T\times d_t}$, where $d_t = d_{out}$. The operation of each head can be formulated as:

$$Attention(Q, K, V) = \text{Softmax}\left(QK^\top/\sqrt{d_t}\right)V, \tag{9}$$

where $\{Q, K, V\} \in \mathbb{R}^{T\times d_t}$ are projected from the input Z, and T indicates the number of tokens, which is equal to frames of input sequence. We repeat the temporal transformer module for n times to model efficient global temporal relationships of the input sequence.

3.4 Loss Functions

The total loss function of the proposed method consisting of tree items is written as follow:

$$\mathcal{L}_{total} = \mathcal{L}_{MPJPE} + \mathcal{L}_e + \mathcal{L}_{KL}. \tag{10}$$

With the internal error computing of uncertainty loss $\mathcal{L}_{evidence}$ on depth dimension, we can supervise the estimation result of the depth dimension. Therefore the MPJPE loss $\mathcal{L}_{MPJPE}$ can be only applied to supervise 2D coordinates. The constraint of 2D distribution $\mathcal{L}_{KL}$ is the last loss function we apply during training.

4 EXPERIMENTS

4.1 Implementation Details

The proposed method was implemented on the Pytorch [40] and the experiments were conducted on a single NVIDIA RTX 2080Ti GPU. The 2D detector could be any off-the-shelf models, we choose CPN [9] and HRNet [46] as our detectors, because CPN [9] has been widely applied, and HRNet [46] has better detection performance. We apply the AdamW [32] as our model optimizer with about 125 training epochs and 2048 batch size. The learning rate is initially set to 5×10^{-4} and 8×10^{-5} for the GCN module and the transformer module respectively to obtain better and faster convergence during training. The learning rate is decayed by the exponential strategy every 2000 iteration steps. The 2D data from 2D keypoint detectors (CPN [9], HRNet [46]) and the 2D GT are applied in the experiments to analyze the performance of our method. We conduct two settings to the input 2D keypoint sequence with length $T = 81$ and $T = 300$ on H36M, the former is set for comparing with the previous SOTAs, and the latter is introduced to achieve better performance. For the other benchmarks 3DHP and HumanEva-I, we set T to 27 and 43 respectively following the previous works [11, 43, 63].

4.2 Datasets and Metrics

Human3.6M (H36M) [19]. Following the previous approaches [4, 33, 35, 62, 63], we take H36M as one of our evaluation datasets, which is the most widely used benchmark, containing 3.6 million video frames captured from four synchronized cameras with different poses at 50 Hz. We adopt the 17-joint pose and use the five subjects S1, S5, S6, S7, and S8 for training, and the other two subjects S9, S11 for testing. The Mean Per Joint Position Error (MPJPE) metric is computed under Protocol #1 (MPJPE between the ground truth (GT) and the estimated 3D poses) and Protocol #2 (aligned with the GT by the rigid transformation, also named P-MPJPE).
MPI-INF-3DHP (3DHP) [36] is a recent large-scale 3D human pose benchmark, which consists of both constrained indoor and complex outdoor scenes, including 8 actions performed by 8 actors that are recorded by 14 camera views. We follow the previous works [17, 29, 37, 51] to split the training set and the testing set. We report the area under the curve (AUC), percentage of correct keypoints (PCK), and MPJPE as our evaluation metrics.

Protocol #1		Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	**Average**
Pavlakos *et al.* [42]	CVPR2017	67.4	71.9	66.7	69.1	72.0	77.0	65.0	68.3	83.7	96.5	71.7	65.8	74.9	59.1	63.2	71.9
Martinez *et al.* [35] (SH)	ICCV2017	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
Pavlakos *et al.* [41]	CVPR2018	48.5	54.4	54.4	52.0	59.4	65.3	49.9	52.9	65.8	71.1	56.6	52.9	60.9	44.7	47.8	56.2
Pavllo *et al.* [43] (CPN, T=243)(†)	CVPR2019	45.2	46.7	43.3	45.6	48.1	55.1	44.6	44.3	57.3	65.8	47.1	44.0	49.0	32.8	33.9	46.8
Cai *et al.* [4] (CPN, T=7)(†)	ICCV2019	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
Yeh *et al.* [56](†)	NeurIPS2019	44.8	46.1	43.3	46.4	49.0	55.2	44.6	44.0	58.3	62.7	47.1	43.9	48.6	32.7	33.3	46.7
Liu *et al.* [31] (CPN, T=243)(†)	CVPR2020	41.8	44.8	41.1	44.9	47.4	54.1	43.4	42.2	56.2	63.6	45.3	43.5	45.3	31.3	32.2	45.1
Wang *et al.* [51] (CPN, T=96)(†)	ECCV2020	40.2	42.5	42.6	41.1	46.7	56.7	41.4	42.3	56.2	60.4	46.3	42.2	46.2	31.7	31.0	44.5
Xu *et al.* [54](T=1)	CVPR2021	45.2	49.9	47.5	50.9	54.9	66.1	48.5	46.3	59.7	71.5	51.4	48.6	53.9	39.9	44.1	51.9
Zeng *et al.* [59](†)	ICCV2021	43.1	50.4	43.9	45.3	46.1	57.0	46.3	47.6	56.3	61.5	47.7	47.4	53.5	35.4	37.3	47.9
Zheng *et al.* [63](CPN, T=81)(†)	ICCV2021	41.5	44.8	39.8	42.5	46.5	51.6	42.1	42.0	53.3	60.7	45.5	43.3	46.1	31.8	32.2	44.3
Chen *et al.* [5](†)(CPN, T=243)	TCSVT2021	41.4	43.5	40.1	42.9	46.6	51.9	41.7	42.3	53.9	60.2	45.4	41.7	46.0	31.5	32.7	44.1
Hu *et al.* [17](CPN, T=96)(†)	MM2021	38.0	43.3	39.1	39.4	45.8	53.6	41.4	41.4	55.5	61.9	44.6	41.9	44.5	31.6	29.4	43.4
Ours(CPN, T=81)(†)		39.6	43.0	37.7	40.5	42.2	50.6	41.1	41.9	49.1	54.8	41.8	43.9	42.6	31.8	30.6	42.1
Ours(CPN, T=300)(†)		37.9	41.9	36.8	39.5	40.8	49.2	40.1	40.7	47.9	53.3	40.2	41.1	40.3	30.8	28.6	40.6
Wang *et al.* [51](HRNet, T=96)(†)	ECCV2020	38.2	41.0	45.9	39.7	41.4	51.4	41.6	41.4	52.0	57.4	41.8	44.4	41.6	33.1	30.0	42.6
Wehrbein *et al.* [52](HRNet, T=200)(†)	ICCV2021	38.5	42.5	39.9	41.7	46.5	51.6	39.9	40.8	49.5	56.8	45.3	46.4	46.8	37.8	40.4	44.3
Hu *et al.* [17](CPN, T=96)(†)	MM2021	35.5	41.3	36.6	39.1	42.4	49.0	39.9	37.0	51.9	63.3	40.9	41.3	40.3	29.8	28.9	41.1
Ours(HRNet, T=300) (†)		35.1	40.2	36.1	38.9	40.0	44.7	39.2	37.8	45.8	53.7	39.2	39.9	38.6	29.9	28.3	39.2

Protocol #2		Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	**Avg.**
Wang *et al.* [51](CPN, T=96)(†)	ECCV2020	31.8	34.3	35.4	33.5	35.4	41.7	31.1	31.6	44.4	49.0	36.4	32.2	35.0	24.9	23.0	34.5
Liu *et al.* [31](CPN, T=243)(†)	CVPR2020	32.3	35.2	33.3	35.8	35.9	41.5	33.2	32.7	44.6	50.9	37.0	32.4	37.0	25.2	27.2	35.6
Zheng *et al.* [63](CPN, T=81)(†)	ICCV2021	34.1	36.1	34.4	37.2	36.4	42.2	34.4	33.6	45.0	52.5	37.4	33.8	37.8	25.6	27.3	36.5
Hu *et al.* [17](CPN, T=96)(†)	MM 2021	29.8	34.4	31.9	31.5	35.1	40.0	30.3	30.8	42.6	49.0	35.9	31.8	35.0	25.7	23.6	33.8
Ours(CPN, T=81)(†)		31.7	34.8	30.7	34.2	33.2	39.1	32.5	32.3	39.6	45.7	33.4	35.1	33.1	25.8	25.6	33.7
Ours(CPN, T=300)(†)		30.3	34.6	29.6	31.7	31.6	38.9	31.8	31.9	39.2	42.8	32.1	32.6	31.4	25.1	23.8	32.5
Wang *et al.* [51](HRNet)(†)	ECCV2020	28.4	32.5	34.4	32.3	32.5	40.9	30.4	29.3	42.6	45.2	33.0	32.0	33.2	24.2	22.9	32.7
Wehrbein *et al.* [52](HRNet, T=200)(†)	ICCV2021	27.9	31.4	29.7	30.2	34.9	37.1	27.3	28.2	39.0	46.1	34.2	32.3	33.6	26.1	27.5	32.4
Hu *et al.* [17](CPN, T=96)(†)	MM 2021	27.7	32.7	29.4	31.3	32.5	37.2	29.3	28.5	39.2	50.9	32.9	31.4	32.1	23.6	22.8	32.1
Ours(HRNet, T=300)(†)		27.2	31.6	27.8	31.2	30.1	34.4	28.6	29.1	36.2	46.2	31.1	32.8	30.2	22.7	21.6	30.7

Table 2: The MPJPE (mm) results of detailed comparison with the state-of-the-arts on the H36M dataset under Protocol #1 (Top table) and Protocol #2 (Bottom Table). T is the number of input sequence length of 2D keypoints, (†) represents using the temporal information. The best and second-best results are highlighted in red and blue color, respectively.

HumanEva-I [45] records three subjects from three camera views at 60 Hz. In the same manner of [31, 63], we conduct training and testing on the dataset with two actions (*Walk* and *Jog*) in subjects S1, S2, and S3. The MPJPE and P-MPJPE are applied to evaluate the proposed method.

4.3 Comparison to the State-of-the-arts

4.3.1 ***Results on H36M.*** To evaluate the effectiveness of our method, we first quantitatively compared our method with the state-of-the-arts on the H36M benchmark in Table 2. The input 2D keypoints are obtained from CPN [9] and HRNet [46]. The previous methods which we compared with include both the image-based and video-based (mark with †) approaches. It can be seen that the video-based methods mostly have better performance than the image-based methods, because the former utilize more temporal information in videos. As shown in Table 2, our method outperforms the previous works by a large margin under both Protocol #1 and Protocol #2. Specifically, for the CPN [9] detector data, our approach with $T = 300$ achieves the best result of average MPJPE of 40.6mm under Protocol #1 (3.7mm improvement compared to SOTA [17]). We also get the second-best when we shorten the input sequence length to $T = 81$. Besides, for the HRNet [46] detector data, we also obtain the best performance on the better input keypoints (2.0mm improvement compared to SOTA [17]). More importantly, it can be observed that our method performs even better on hard actions (*e.g.*, Smoke, Sit, and SittingDown).

To further explore the upper bound of the proposed method, we compare our model with the prior methods with the GT 2D keypoints as input. This can eliminate the effect of the quality of the 2D detection data for the 2D-to-3D lifting methods. As shown in Table 3, our method significantly outperforms all the others in terms of MPJPE under the Protocol #1. It demonstrates that if a more powerful 2D pose estimator is available, our U-CondDGCN is able to produce more accurate 3D human poses.

4.3.2 ***Results on 3DHP.*** Table 4 shows the detailed results of different methods on the 3DHP benchmark. It can be seen that our approach achieves significantly better PCK and AUC scores and MPJPE than other methods including both the image-based and video-based methods. The result demonstrates that our method has strong generalization ability, which is beneficial from the imbuing of uncertainty representation into model learning. Besides, GCN-based methods [17, 51] also obtain good performance, which have a strong modeling ability of human structure. But the pure CNN-based method [31] gets a low accuracy because it without considering the skeleton prior and uncertainty of the poses.

4.3.3 ***Results on HumanEva-I.*** Table 5 shows results on HumanEva-I and the generalization to a much smaller dataset. We followed the setting of [43, 63] to conduct the 2D detection data (based on the pre-trained Mask R-CNN 2D detector). The high error on "Walk" of the third column is due to corrupted mocap data [43]. Our proposed method outperforms the previous works in average MPJPE under Protocol #1, which demonstrates there is a great ability of

Protocol #1		Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Pur.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg.
Pavllo *et al.* [43](T=243)	CVPR2019	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	37.2
Liu *et al.* [31](T=243)	CVPR2020	34.5	37.1	33.6	34.2	32.9	37.1	39.6	35.8	40.7	41.4	33.0	33.8	33.0	26.6	26.9	34.7
Wang *et al.* [51](T=96)	ECCV2020	23.0	25.7	22.8	22.6	24.1	30.6	24.9	24.5	31.1	35.0	25.6	24.3	25.1	19.8	18.4	25.6
Zheng *et al.* [63](T = 81)	ICCV2021	30.0	33.6	29.9	31.0	30.2	33.3	34.8	31.4	37.8	38.6	31.7	31.5	29.0	23.3	23.1	31.3
Hu *et al.* [17](CPN, T=96)	MM 2021	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22.7
Ours(T=81)		25.6	27.8	24.5	25.7	24.9	29.9	28.6	27.4	29.9	29.0	26.1	25.0	25.2	18.7	19.9	25.9
Ours(T=300)		22.1	23.1	20.1	22.7	21.3	24.1	23.6	21.6	26.3	24.8	21.7	21.4	21.8	16.7	18.6	22.0

Table 3: Quantitative comparison of MPJPE in millimeters (mm) on Human3.6M under Protocol #1 using 2D ground truth (GT) keypoints as input. All of our comparison methods are based on video, which can better compare the upper bounds of the approaches. The best results are highlighted in red.

Method		PCK[↑]	AUC[↑]	MPJPE[↓]
Mehta *et al.* [37]	ACM TOG 2017	79.4	41.6	-
Lin *et al.* [29](†)	BMVC2019	83.6	51.4	79.8
Li *et al.* [25]	CVPR2020	81.2	46.1	99.7
Wang *et al.* [51](†)	ECCV2020	86.9	62.1	68.1
Gong *et al.* [15]	CVPR2021	88.6	57.3	73.0
Zheng *et al.* [63](†)	ICCV2021	88.6	56.4	77.1
Hu *et al.* [17]	MM 2021	97.9	69.5	42.5
Ours(T=27)(†)		98.2	70.1	53.3

Table 4: Quantitative comparison on MPI-INF-3DHP with PCK, AUC, and MPJPE metrics. ↑ indicates the higher, the better, while ↓ indicates the lower, the better. The best and second-best results are highlighted in red color. † indicates the video-based methods.

Method	DUE	2D-PD	GTE	MPJPE (mm)	Params (M)
Baseline (d = 64, depth = 8)	×	×	×	67.6	76.51
Baseline (d = 128, depth = 8)	×	×	×	59.8	305.06
Baseline (d = 256, depth = 8)	×	×	×	*nan*	1218.30
Configure 1(d = 256, depth = 8)	×	×	✓	49.8	4.68
Configure 2(d = 256, depth = 8)	×	✓	✓	43.9	4.68
Ours (d = 256, depth = 8)	✓	✓	✓	40.6	4.68

Table 6: Ablation study on each component introduced to our method. The evaluation is performed on the H36M dataset with MPJPE (mm). DUE indicates the *Depth Uncertainty Estimation* component, the 2D-PD is *2D Probabilistic Distribution* module, and the GTE represents the *GCN-Transformer Encoder* applied in the proposed method.

our model to learn general feature even in a small dataset. Besides, pure transformer-based method [63] performs badly because of the limitation of transformer in small datasets. We also introduce the transformer but keep it as light-weight to get the ability that utilizes it not only in large datasets.

Protocol #1	Walk			Jog			Avg.
Pavllo *et al.* [43](T=81)	13.1	10.1	39.8	20.7	13.9	15.6	18.9
Zheng *et al.* [63](T=43)	16.3	11	47.1	25	15.2	15.1	21.6
Zheng *et al.* [63](T=43, FT)	14.4	10.2	46.6	22.7	13.4	13.4	20.1
Ours(T=43)	13.1	10.7	37.9	21.2	17.2	18.8	18.1

Table 5: Comparison on the HumanEva-I benchmark under Protocol #1. FT indicates pre-training on the H36M dataset and then fine-tuning on the HumanEva-I dataset. The best results are highlighted in red color.

4.4 Ablation Study

To evaluate the impact and performance of each component in our model, we evaluate their effectiveness in this subsection. The Human3.6M dataset and the CPN [43] detector are utilized to provide the 2D keypoints.

4.4.1 ***Impact of model components.*** As shown in Table 6, we set the input sequence length to 300 and use the PoseFormer [63] as the baseline, then we change the dimension size of channel d and model depth (number of encoder layers) $depth$. Besides, we regress the 3D pose sequence each time to construct the *seq2seq* model, which is also different from [63]. It can be observed that with the increment of the model depth and dimension size of channel, the model parameters of the baseline increase rapidly and becomes too large to train efficiently. After enabling the GCN-Transformer Encoder (GTE) module in the Configuration 2, the model parameter is greatly reduced and can be easier to train. This mainly because the combination of GCN and transformer is effective, and the spatial human structure prior modeling ability of GCN is efficient. With the 2D Probabilistic Distribution (2D-PD) component turning in the Configuration 2, the performance obtains a significant growup. Finally, we apply the DUE (Depth Uncertainty Estimation) to model the uncertainty information on the depth dimension, and our method achieves the best performance under Protocol #1. The ablation study on each component reveals that the uncertainty-aware information plays an important role in the 2D-to-3D lifting task. Moreover, the GTE provides the ability on efficiently modeling the spatial and temporal relationships.

4.4.2 ***Impact of the model parameter setting.*** We conduct the ablation study on the dimension size of channel d, the number of GCN layers L_{GCN}, and the number of transformer layer $L_{Transformer}$. We choose 27-frames setting in this study from the perspective of experimental efficiency. As shown in Table 7, the number of GCN layers brings few parameters, the main increase of model parameters comes from the dimension and the number of transformer layers. We finally choose the parameter combination ($d = 256, L_{GCN} = 2, L_{Transformer} = 8$) to better balance the computation efficiency and performance.

d	L_{GCN}	$L_{Transformer}$	Params (M)	MPJPE (mm)	Es
128	1	2	0.43	57.4	50
256	1	2	1.52	49.2	80
512	1	2	5.66	48.9	150
256	**2**	2	1.67	48.2	80
256	4	2	1.98	48.2	100
256	8	2	2.59	48.1	100
256	2	4	2.57	46.1	120
256	2	**8**	4.68	43.1	120
256	2	16	8.90	42.8	300

Table 7: Ablation study on different parameters. The evaluation is performed on the H36M benchmark with MPJPE (mm). *Es* indicates training epochs to get convergence. The best choice of parameter combination is highlighted in bold.

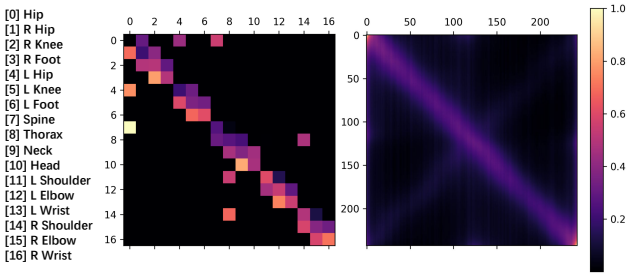

Figure 3: Visualization of spatial learned adjacent matrix among body joints and temporal attention map among input frames. The weights are normalized to $[0, 1]$, and the light color indicates higher weight.

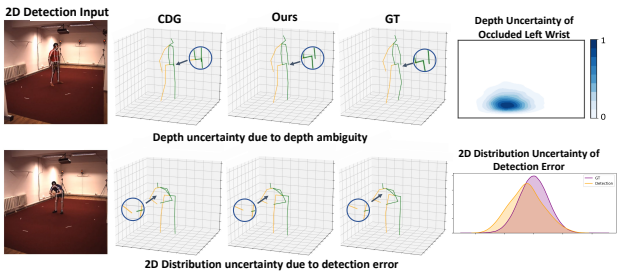

Figure 4: Qualitative comparison between our method and the SOTA approach CDG [17] on H36M test set S9, S11. The blue circles and arrows highlight locations where our method clearly has better results. The uncertainty of depth and 2D detection input are shown at the right column.

5 QUALITATIVE RESULTS

Qualitative results of spatial and temporal relationship modeling. We conduct visualization of the relationships on spatial and temporal domains to verify the effectiveness of the proposed encoder. A video sequence of specific action (*Sitting*) in H36M benchmark is selected for illustration. We visualize the average spatial learned adjacent matrix of joints in the GCN module and the temporal attention map of frames in the transformer module, respectively. The spatial learned adjacent matrix has a shape of $N \times N$, where N is the number of joints of the pre-defined skeleton, while the temporal attention map has a shape of $T \times T$, where T is the length of the input frames. As shown in Figure 3, there are different weight distributions in the spatial and temporal domains. For the spatial matrix (left of Figure 3), it is clear that the matrix mainly aggregates information among their near neighbor joints, some nodes can aggregate long-range information. For the temporal attention map (right of Figure 3), the videos with fast movement like *Sitting*, transformer captures attention mainly from itself and its neighboring frames, thus the weight distribution is more smooth than the spatial matrix.

Qualitative comparison with SOTA methods. Our method can be chosen to estimate the 3D pose predictions with/without corresponding uncertainty parameters. For some applications, which take 3D pose estimation as the upstream task (*e.g.,* skeleton action recognition), the uncertainty parameters may help them to better learn features of the estimated pose. We enable our model to visualize the uncertainty parameters and estimated poses for better observation. As shown in Figure 4, with more complex poses, predictions of the model have larger uncertainty, and the blue area is larger. This demonstrates that our method is able to predict accurate uncertainty information on depth for some self-occluded keypoints or complex poses. Therefore, the final 3D pose optimized with the uncertainty information is reasonable and meaningful. And for the 2D detection error, our method can also calibrate it from the perspective of probabilistic distribution, as shown in the second row of Figure 4.

6 CONCLUSION

In this paper, we proposed an uncertainty-aware method to optimize the *depth* and $x-y$ coordinates for 3D human pose estimation, respectively. The employed evidential deep learning is able to quantify the uncertainty of depth ambiguity. Moreover, the explored 2D probabilistic representation can efficiently model the distribution uncertainty of the 2D detection input and calibrate its deviations. The efficient GCN-Transformer based encoder enables our model to provide effective spatial and temporal correlations in keypoints sequence for the uncertainty estimation. Importantly, this work opens a new baseline of uncertainty learning on 3D human pose estimation from the monocular video. Extensive experiments demonstrate that our method achieves the state-of-the-art performance.

Limitation. One limitation of the proposed method is that the performance is relatively degraded when dealing with fast-motion in-the-wild videos, which is affected by the poor 2D keypoints. The failure cases are shown in the supplementary materials.

Acknowledgement. This work was supported by the National Natural Science Foundation of China under Grant 62106177 and the Joint Fund of the Ministry of Education of China under Grant 8091B032156. The numerical calculation was supported by the super-computing system in the Supercomputing Center of Wuhan University.

REFERENCES

[1] Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 62–1.

[2] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. 2020. Deep evidential regression. *Advances in Neural Information Processing Systems* 33 (2020), 14927–14937.

[3] Wentao Bao, Qi Yu, and Yu Kong. 2021. Evidential deep learning for open set action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13349–13358.

[4] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. 2019. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2272–2281.

[5] Tianlang Chen, Chen Fang, Xiaohui Shen, Yiheng Zhu, Zhili Chen, and Jiebo Luo. 2021. Anatomy-aware 3D Human Pose Estimation with Bone-based Pose Decomposition. *IEEE Transactions on Circuits and Systems for Video Technology* (2021).

[6] Yujin Chen, Zhigang Tu, Liuhao Ge, Dejun Zhang, Ruizhi Chen, and Junsong Yuan. 2019. So-handnet: Self-organizing network for 3d hand pose estimation with semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6961–6970.

[7] Yujin Chen, Zhigang Tu, Di Kang, Linchao Bao, Ying Zhang, Xuefei Zhe, Ruizhi Chen, and Junsong Yuan. 2021. Model-based 3d hand reconstruction via self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10451–10460.

[8] Yujin Chen, Zhigang Tu, Di Kang, Ruizhi Chen, Linchao Bao, Zhengyou Zhang, and Junsong Yuan. 2021. Joint hand-object 3d reconstruction from a single image with cross-branch feature fusion. *IEEE Transactions on Image Processing* 30 (2021), 4008–4021.

[9] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. 2018. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7103–7112.

[10] Yu Cheng, Bo Yang, Bo Wang, and Robby T Tan. 2020. 3d human pose estimation using spatio-temporal networks with explicit occlusion training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 10631–10638.

[11] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509* (2019).

[12] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. 2019. Optimizing Network Structure for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[13] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. 2017. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE international conference on computer vision*. 2334–2343.

[14] Yarin Gal, Jiri Hron, and Alex Kendall. 2017. Concrete dropout. *Advances in neural information processing systems* 30 (2017).

[15] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. 2021. PoseAug: A Differentiable Pose Augmentation Framework for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8575–8584.

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969.

[17] Wenbo Hu, Changgong Zhang, Fangneng Zhan, Lei Zhang, and Tien-Tsin Wong. 2021. Conditional directed graph convolution for 3d human pose estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 602–611.

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7 (July 2014), 1325–1339. https://doi.org/10.1109/TPAMI.2013.248

[19] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7 (jul 2014), 1325–1339.

[20] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. 2019. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7718–7727.

[21] Durk P Kingma, Tim Salimans, and Max Welling. 2015. Variational dropout and the local reparameterization trick. *Advances in neural information processing systems* 28 (2015).

[22] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=SJU4ayYgl

[23] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems* 30 (2017).

[24] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. 2021. Human pose regression with residual log-likelihood estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11025–11034.

[25] Shichao Li, Lei Ke, Kevin Pratama, Yu-Wing Tai, Chi-Keung Tang, and Kwang-Ting Cheng. 2020. Cascaded Deep Monocular 3D Human Pose Estimation With Evolutionary Training Data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[26] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, Pichao Wang, and Wenming Yang. 2022. Exploiting temporal contexts with strided transformer for 3d human pose estimation. *IEEE Transactions on Multimedia* (2022).

[27] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. 2021. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. *arXiv preprint arXiv:2111.12707* (2021).

[28] Yanjie Li, Shoukui Zhang, Zhicheng Wang, Sen Yang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. 2021. Tokenpose: Learning keypoint tokens for human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 11313–11322.

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

[30] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. 2020. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *European Conference on Computer Vision*. Springer, 318–334.

[31] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. 2020. Attention mechanism exploits temporal contexts: Real-time 3D human pose reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5064–5073.

[32] Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam. (2018).

[33] Xiaoxuan Ma, Jiajun Su, Chunyu Wang, Hai Ci, and Yizhou Wang. 2021. Context Modeling in 3D Human Pose Estimation: A Unified Perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6238–6247.

[34] Andrey Malinin and Mark Gales. 2018. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems* 31 (2018).

[35] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. 2017. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE international conference on computer vision*. 2640–2649.

[36] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. 2017. Monocular 3D Human Pose Estimation In The Wild Using Improved CNN Supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE. https://doi.org/10.1109/3dv.2017.00064

[37] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. 2017. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–14.

[38] Dmitry Molchanov, Arsenii Ashukha, and Dmitry Vetrov. 2017. Variational dropout sparsifies deep neural networks. In *International Conference on Machine Learning*. PMLR, 2498–2507.

[39] Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*. Springer, 483–499.

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[41] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. 2018. Ordinal Depth Supervision for 3D Human Pose Estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[42] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. 2017. Coarse-To-Fine Volumetric Prediction for Single-Image 3D Human Pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[43] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 2019. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7753–7762.

[44] Murat Sensoy, Lance Kaplan, and Melih Kandemir. 2018. Evidential deep learning to quantify classification uncertainty. *Advances in Neural Information Processing Systems* 31 (2018).

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

[46] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5693–5703.

[47] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. 2018. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 529–545.

[48] Bugra Tekin, Artem Rozantsev, Vincent Lepetit, and Pascal Fua. 2016. Direct prediction of 3d body poses from motion compensated sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 991–1000.

[49] Zhigang Tu, Zhisheng Huang, Yujin Chen, Di Kang, Linchao Bao, Bisheng Yang, and Junsong Yuan. 2022. Consistent 3D Hand Reconstruction in Video via self-supervised Learning. *arXiv preprint arXiv:2201.09548* (2022).

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[51] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2020. Motion guided 3d pose estimation from videos. In *European Conference on Computer Vision*. Springer, 764–780.

[52] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. 2021. Probabilistic Monocular 3D Human Pose Estimation With Normalizing Flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11199–11208.

[53] Bin Xiao, Haiping Wu, and Yichen Wei. 2018. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*. 466–481.

[54] Tianhan Xu and Wataru Takano. 2021. Graph Stacked Hourglass Networks for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16105–16114.

[55] Sijie Yan, Yuanjun Xiong, and Dahua Lin. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*.

[56] Raymond Yeh, Yuan-Ting Hu, and Alexander Schwing. 2019. Chirality nets for human pose regression. *Advances in Neural Information Processing Systems* 32 (2019), 8163–8173.

[57] Jae Shin Yoon, Lingjie Liu, Vladislav Golyanik, Kripasindhu Sarkar, Hyun Soo Park, and Christian Theobalt. 2021. Pose-Guided Human Animation From a Single Image in the Wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15039–15048.

[58] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. 2020. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *European Conference on Computer Vision*. Springer, 507–523.

[59] Ailing Zeng, Xiao Sun, Lei Yang, Nanxuan Zhao, Minhao Liu, and Qiang Xu. 2021. Learning Skeletal Graph Neural Networks for Hard 3D Pose Estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11436–11445.

[60] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. 2022. MixSTE: Seq2seq Mixed Spatio-Temporal Encoder for 3D Human Pose Estimation in Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13232–13242.

[61] Jiaxu Zhang, Gaoxiang Ye, Zhigang Tu, Yongtao Qin, Jinlu Zhang, Xiangjian Liu, and Shixu Luo. 2020. A spatial attentive and temporal dilated (SATD) GCN for skeleton-based action recognition. *CAAI Transactions on Intelligence Technology* (2020).

[62] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N. Metaxas. 2019. Semantic Graph Convolutional Networks for 3D Human Pose Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[63] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 2021. 3D Human Pose Estimation With Spatial and Temporal Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11656–11665.

[64] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. 2017. Towards 3d human pose estimation in the wild: a weakly-supervised approach. In *Proceedings of the IEEE International Conference on Computer Vision*. 398–407.

A DETAILED DERIVATIONS

As mentioned in Section 3.1.1, following the previous work [44], we assume that the depth prediction follows Gaussian distribution with unknown mean μ and variance σ^2:

$$\left(\mu \mid \sigma^2\right) \sim \mathcal{N}\left(\mu_0, \frac{\sigma^2}{k}\right), \sigma^2 \sim \text{Inv}-\chi^2(\alpha, \beta), \tag{11}$$

where the parameters of the distributions are $(\mu_0, k, \alpha, \beta)$. Therefore, the model prediction can be formatted as:

$$\mathbb{E}[\mu] = \int_{\mu=-\infty}^{\infty} \mu p(\mu) \mathrm{d}\mu = \mu_0, \tag{12}$$

where the μ_0 indicates the prediction on depth. And and uncertainty of depth prediction is represented as:

$$\begin{aligned} \text{Var}[\mu] &= \int_{\mu=-\infty}^{\infty} \mu^2 p(\mu) \mathrm{d}\mu - (\mathbb{E}[\mu])^2 \\ &= \mu_0^2 - \frac{\sigma^2}{k} - \mu_0^2 \\ &= \mu_0^2 - \frac{\frac{\beta}{\alpha-1}}{k} - \mu_0^2 \\ &= \frac{\beta}{v(\alpha-1)}, \quad \alpha > 1. \end{aligned} \tag{13}$$

In this way, we can take the parameters $(\mu_0, k, \alpha, \beta)$ to represent the uncertainty of model prediction on depth.

To imbue the practical meanings to these parameters in probability distribution during training, we apply the variance of the sample mean to supervise the depth uncertainty, which is a different and more simple design than previous works [2]:

$$\mathbb{L}_u = \left| \frac{\beta}{k(\alpha-1)} - \text{Var}[\hat{\mu}] \right|, \tag{14}$$

where the $\hat{\mu}$ indicates the samples in the training batch.

B ABLATION STUDY ON UNCERTAINTY ESTIMATION

To further explore the effect of the uncertainty estimation module, we make the ablation studies on depth uncertainty and 2D distribution uncertainty. As presented in Table 8, removing the depth uncertainty estimation (DUE) module leads to 3.3 mm increase in error, and without the 2D probabilistic distribution (2D-PD), the proposed method increases 4.1 mm in errors. Without these two uncertainty-aware modules, the performance falls 9.0 mm in terms of MPJPE. The experiment demonstrates the benefit of uncertainty estimation module in our proposed framework. Moreover, it also shows the performance of our proposed encoder.

C ABLATION STUDEY ON INPUT SEQUENCE LENGTH

Our method utilizes the temporal information to improve the estimation performance by employing the GCN-Transformer encoder. Basically, the longer input sequence of the model, the better performance it obtains, because of more available temporal information. Figure 5 illustrates the results of our method with different lengths of input frames. It can be observed that the proposed method gets larger gains with more frames fed into the model. The error has a significant decrease of 32.2% from the single-frame setting to 300-frames setting with three input data, which indicates the effectiveness of our method in capturing long-range dependency across frames with the large receptive field. There is no much significant improvement when the input sequence length is longer than 300, thus we pick 300-frame setting as our final choice. Besides, the ablation study also shows that our method can work well on single frame, which means the temporal transformer module is automatically abandoned in this case.

Method	MPJPE (mm ↓)	Δ
Ours (DUC + 2D-PD)	40.6	–
w/o DUE	43.9	3.3
w/o 2D-PD	44.7	4.1
w/o DUC & 2D-PD (only GTE)	47.6	7.0

Table 8: Ablation study on uncertainty-aware modules in the proposed method. The DUC indicates the depth uncertainty estimation, 2D-PD is 2D probabilistic distribution, and GTE is the proposed GCN-Transformer encoder. The evaluation is performed on H36M with the MPJPE metric under Protocol #1. ↓ indicates lower is better.

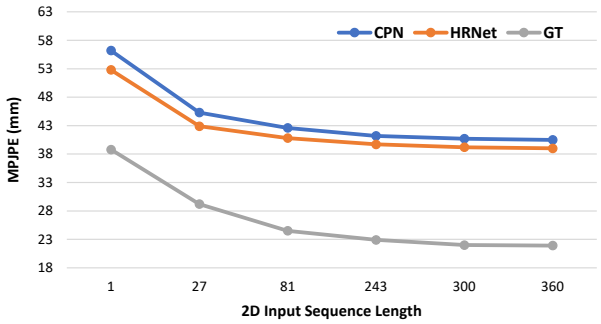

Figure 5: Ablation study on different 2D detection sequence lengths with MPJPE (mm) under Protocol #1. We apply different input data (CPN [9], HRNet [46], and 2D ground truth.)

D DETAIL COMPARISON WITH SOTAS

To further illustrate the differences between the proposed method and previous SOTAs, we compare our method with these works [5, 17, 31, 43, 63]. The comparison items include the parameters, FLOPs, FPS, and MPJPE. The experiment setting follows the PoseFormer [63]. As shown in Table 9, our model achieves the best performance and highest inference speed on H36M dataset with longer input sequence length while relatively small parameters and low computing cost. This experiment shows that our method can be easily applied to multi-paralleled tasks and is able to maintain real-time.

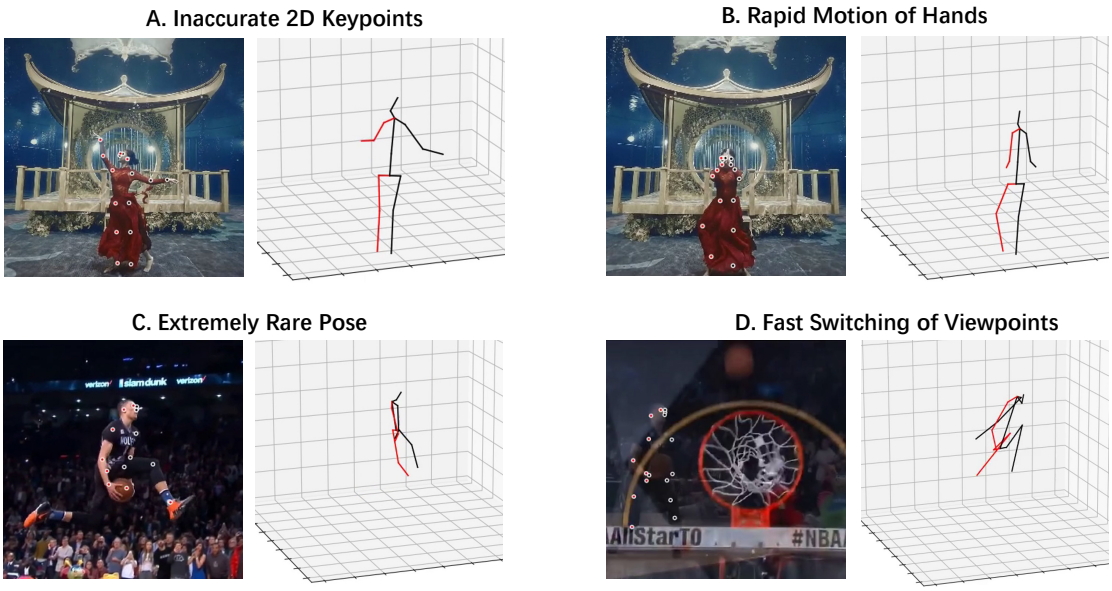

Figure 6: Failure cases. We show the input image, 2D detection keypoints, and corresponding estimated 3D poses in the wild. See Section E for details.

Method	T	Parameters (M)	FLOPs (M)	MPJPE	FPS
Pavllo *et al.* [43]	81	12.79	25.48	47.7	1121
Pavllo *et al.* [43]	243	16.95	33.87	46.8	863
Liu *et al.* [31]	243	11.25	-	45.1	66
Chen *et al.* [5]	81	45.53	88.9	44.6	315
Chen *et al.* [5]	243	59.18	116	44.1	264
Zheng *et al.* [63]	81	9.60	1358	44.3	269
Hu *et al.* [17]	96	3.42	-	43.4	289
Ours	300	4.68	148	40.6	1720

Table 9: Comparison on parameters, FLOPs, MPJPE, and inference speed (FPS). The evaluation is performed on H36M under Protocol #1.

E FAILURE CASES

While our method can estimate accurate human pose results. We also observe some failure cases when inference in the wild. We follow the VideoPose3D [43] to render the estimated 3D pose in the wild. It [43] provides a convenient interface to generate 2D keypoint predictions from videos without manually extracting individual frames. The Mask R-CNN [16] is utilized to generate the 2D keypoints. As shown in Figure 6 A., the estimation accuracy tends to be lower in some inaccurate 2D keypoints. For the rapid motion of end of limbs, our method fails to estimate each frame (Figure 6 B.). Extremely rare poses will also lead to inaccurate 3D pose results (Figure 6 C.), which is due to poor supervision on limited training data. And some videos with fast switching of views (Figure 6 C.) leads to failure case because of broken temporal consistency.