

# Probabilistic Triangulation for Uncalibrated Multi-View 3D Human Pose Estimation

Boyuan Jiang<sup>1,2</sup>, Lei Hu<sup>1,2</sup>, Shihong Xia<sup>1,2\*</sup>

<sup>1</sup>Institute of Computing Technology, Chinese Academy of Sciences;

<sup>2</sup>University of Chinese Academy of Sciences

{jiangboyuan20s, hulei19z, xsh}@ict.ac.cn

## Abstract

*3D human pose estimation has been a long-standing challenge in computer vision and graphics, where multi-view methods have significantly progressed but are limited by the tedious calibration processes. Existing multi-view methods are restricted to fixed camera pose and therefore lack generalization ability. This paper presents a novel Probabilistic Triangulation module that can be embedded in a calibrated 3D human pose estimation method, generalizing it to uncalibration scenes. The key idea is to use a probability distribution to model the camera pose and iteratively update the distribution from 2D features instead of using camera pose. Specifically, we maintain a camera pose distribution and then iteratively update this distribution by computing the posterior probability of the camera pose through Monte Carlo sampling. This way, the gradients can be directly back-propagated from the 3D pose estimation to the 2D heatmap, enabling end-to-end training. Extensive experiments on Human3.6M and CMU Panoptic demonstrate that our method outperforms other uncalibration methods and achieves comparable results with state-of-the-art calibration methods. Thus, our method achieves a trade-off between estimation accuracy and generalizability. Our code is in [https://github.com/bymaths/probabilistic\\_triangulation](https://github.com/bymaths/probabilistic_triangulation)*

## 1. Introduction

3D human pose estimation is a fundamental tool in many downstream applications of computer vision and computer graphics, such as motion recognition[13], human-computer interaction[23], movie game animation[4], virtual reality[24], etc. The development of these fields requires a more accurate, convenient, and robust 3D human pose estimation algorithm.

Recently, RGB-based pose estimation has become the

trend because of its convenience, wide application scenarios, and low cost[27]. These methods can be divided into two categories: single-view and multi-view methods. Although single-view methods[39, 37, 11, 38, 25] have more widespread application scenarios, the lack of depth information makes single-view estimation of 3D human pose an ill-posed problem, especially when dealing with occlusions and multi-person scenes[21]. Multi-view methods[18, 16, 36, 33, 2, 14, 31, 20] can achieve more accurate results than single view-based methods because multi-view cameras can restore the depth information as well as the occluded parts.

However, the multi-view approach faces several issues when applied in practice. First, multi-view methods require synchronized and calibrated cameras, which complicates the process. Second, existing datasets for multi-view human pose estimation lack the diversity of camera external parameters. This results in the trained model cannot generalize well to unseen camera parameters, thus requiring re-annotation and re-training once the camera configurations change.

To alleviate the above issues, we propose Probabilistic Triangulation module for uncalibrated 3D human pose estimation. Our module enables the network to get rid of camera pose parameters during the training process, which results in better generalization performance to be applied to wild scenes. The main idea is to use a probability distribution to model the camera pose, making the whole process differentiable. Specifically, we transform the uncalibrated 3D human pose estimation into a nonlinear optimization problem based on the reprojection error. The camera pose distribution is estimated from the 2D pose features, and then the posterior probability of the camera pose is computed by Monte Carlo sampling to iteratively update this distribution.

We conducted extensive experiments on two benchmark datasets (Human3.6M[17] and CMU Panoptic[19, 29, 35]) to demonstrate that our method outperforms other uncalibrated methods and achieves comparable results with the state-of-the-art calibration methods, which indicates that

---

\*Corresponding author

our method improves the generalized ability to camera parameters in multi-view human pose estimation while maintaining high accuracy.

In summary, our contributions are mainly three-fold.

- We propose Probabilistic Triangulation that can generalize the calibrated 3D human pose estimation algorithm to uncalibrated scenes to estimate 3D human pose in the absence of camera pose.
- We propose an approximation method for camera pose distribution and an updated distribution method based on Monte Carlo sampling. It is used to model the camera pose of uncalibrated scenes with accuracy and robustness beyond classical algorithms.
- We conducted extensive experiments on two benchmark datasets (Human3.6M and CMU Panoptic) to demonstrate the accuracy, generalization capability, and applicability of Probabilistic Triangulation to wild scenes.

## 2. Related work

### 2.1. Calibrated 3D pose estimation

3D human pose estimation is a vital problem in computer vision and graphics, where the single view-based method has been a research spotlight due to the convenience and comprehensive application scenarios of a monocular camera. To alleviate the ambiguity problem caused by occlusion and the lack of depth information, many single view-based works[37, 11, 38, 25, 3] have tried to introduce the temporal information or body-related prior during training.

Multi-view images retain more information compared to single-view ones. The association of information between different views eliminates most of the ambiguity. So it is easier to achieve high-accuracy 3D human pose estimation in multi-view settings. Fusing the information of different views is the core problem of multi-view human pose estimation. Existing multi-view methods include two processes, 2D pose estimation and 3D pose reconstruction. The former consists of a pre-trained 2D pose estimator[10, 6], and multi-view information fusion occurs in the 3D pose reconstruction. Multi-view methods are divided into two categories according to the different ways of representing human pose. The first is the point-based method[26, 5, 12], which uses the spatial coordinates of the joint points as the human pose representation. This representation makes it easier to introduce temporal and body information explicitly. The second is the heatmap-based method, which uses the heatmap as the human pose representation[16, 36, 18], and learnable triangulation represents this method[18, 33]. The point representation is more concise and explicit but loses information, while the heatmap representation retains

more information but is not easily bounded explicitly. Our proposed Probabilistic Triangulation is applicable in both representations.

### 2.2. Uncalibrated 3D pose estimation

Uncalibrated 3D pose estimation is a relatively new field[31, 2, 14, 20]. Takahashi’s work[31] is a point-based method that establishes explicit constraints based on reprojection errors and the human body prior to transform the problem into an optimization problem and iteratively solve for the optimal solution. Lee’s work[20] is also a point-based method, estimating the camera pose using the 3D pose estimated from different views. FLEX[14] estimates a time-invariant bone length while estimating joint rotations frame by frame and reconstructing 3D pose by forward kinematics. In this approach, the bone length is a link to fuse the multi-view information, and the rotation estimated for each view implicitly constrains the camera pose. However, this method cannot accurately estimate the camera pose due to the lack of transformation of the global coordinate system. Therefore, the output 3D pose is camera dependent. Bartol et al. [2] use a scoring network to evaluate the weight of the current proposal’s camera pose or 3D human pose, and the final weighting is used to obtain a definite result. This method does not directly estimate the 3D human pose for uncalibrated scenes, but it can estimate the camera pose before and use other calibration methods to estimate the 3D human pose. This method does not directly estimate the 3D human pose for uncalibrated scenes, but it can estimate the camera pose before using other calibration methods to estimate the 3D human pose. Because the experiments for 3D human pose estimation in this work are for calibrated scenes, they are placed in the calibration method in the comparison experiments later in the paper.

Current methods are limited to point-based representation. The heatmap-based representation retains more information than the point-based representation. This is because the accuracy of existing 2D human pose estimators is not sufficient for direct calibration. Small errors in points can be rapidly amplified by the camera pose estimation process, making the method ineffective. Our proposed camera pose distribution can effectively increase the robustness of the camera estimation process and support heatmap-based representation.

### 2.3. Probabilistic camera pose estimation

Camera pose estimation is a fundamental problem in computer vision[15]. PNP methods solve the case of known 3D-2D point pairs, while triangulation solves the case of known 2D-2D point pairs. Bundle adjustment transforms the camera matching problem into an optimization problem starting from reprojection errors[1]. However, matching feature points and solving them using classical meth-

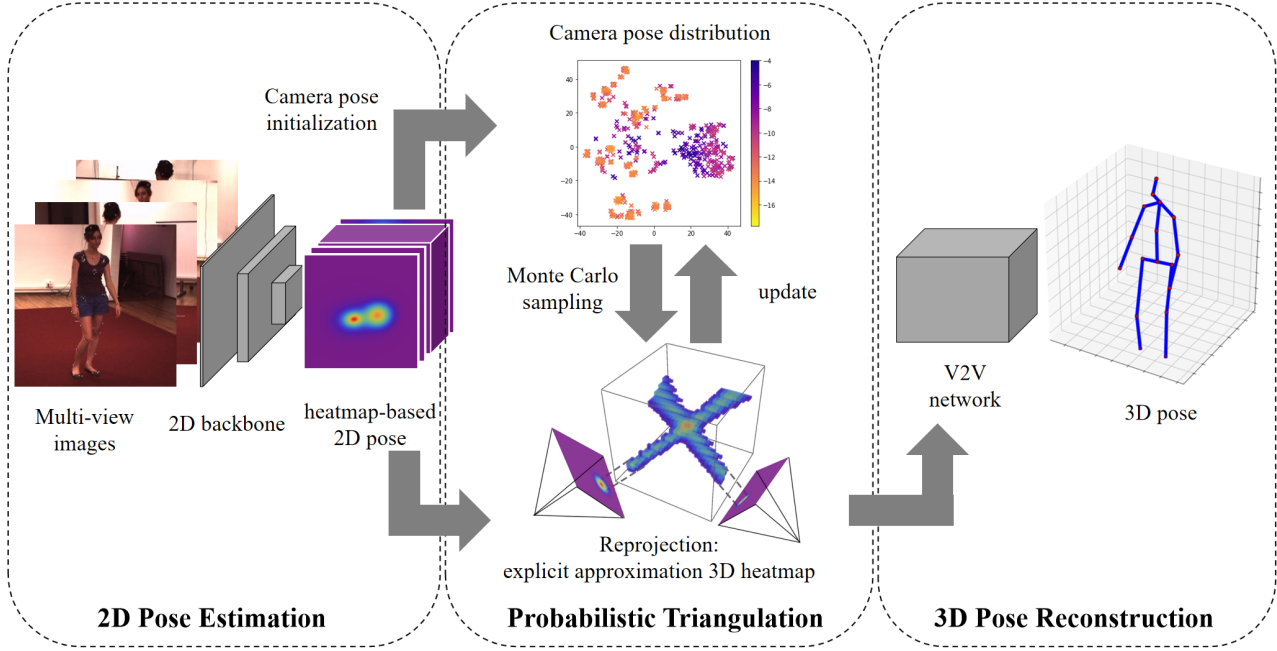


Figure 1: Probabilistic Triangulation for uncalibrated 3D human pose estimation pipeline. The inputs to the network are images from different views at the same moment. The 2D human pose heatmaps are estimated from the images and used to initialize the camera pose distribution. Then, Monte Carlo sampling is performed iteratively and each sampled camera parameter is computed by reprojection to obtain an explicit approximation of the 3D heatmap. The 3D heatmap calculates the weights of the sampled camera parameters. These weights are then used to update the pose distribution of the cameras. The weighted 3D heat map is used as input for the subsequent 3D pose reconstruction, for estimating the 3D human pose.

ods does not apply to all scenarios. In the field of object and camera pose estimation, many works try to improve the network architecture that allows an end-to-end approach for pose estimation[8, 7]. The work EPro-PnP by Chen et al[9]. proposes a probabilistic PnP layer that allows the gradient back-propagated through the pose.

Inspired by EPro-PnP, we use probability distributions to model camera pose. Compared with the camera pose represented by a single variable, the camera pose represented by the distribution has more robust performance and effectively mitigates the effect of 2D human pose detection errors.

### 3. Method

In this paragraph, we will review the multi-view uncalibrated 3D human pose estimation problem and introduce our Probabilistic Triangulation module. The overall pipeline is shown in Figure 1.

#### 3.1. Uncalibrated 3D human pose estimation

Given that the images  $I \in \mathbb{R}^{K \times H \times W}$  from  $K$  views at a specific moment, our goal is to estimate the 3D pose  $X^{3D}$  of the target human from it. Given the camera pose  $y = \{y_k \in SE(3) | k = 1 \dots K\}$ , the optimal  $X^{3D}$  will

make the reprojection error ( $3D \rightarrow 2D$ ) small enough and fits the prior distribution of the human pose learned in the dataset. Therefore, the estimation algorithm is often divided into two parts, 2D pose estimation, and 3D pose reconstruction. In 2D pose estimation, a neural network  $M_1$  is employed to estimate the 2D pose  $X^{2D} = M_1(I)$  from each view. In the 3D pose reconstruction, the  $X^{2D}$  of different views are fused using a triangulation or a learnable triangulation method  $M_2$  to obtain  $X^{3D} = M_2(X^{2D}, y)$ .

Based on the reprojection error, the calibrated 3D human pose estimation is to solve an optimization problem:

$$\min_{X^{3D}} E [proj(X^{3D}, y) - X^{2D}] \quad (1)$$

where  $proj(\cdot)$  denotes the projection function. The neural network-based algorithm replaces  $X^{3D}$  in it with  $M_2(X^{2D}, y)$ , transforming the original optimization problem into solving for the optimal  $M_2(\cdot)$  network parameters.

When the camera pose  $y$  is an unknown parameter,  $y$  also becomes a variable to be optimized. Uncalibrated 3D human pose estimation can be similarly transformed into an optimization problem:

$$\min_{X_{3D}, y} E [proj(X^{3D}, y) - X^{2D}] \quad (2)$$

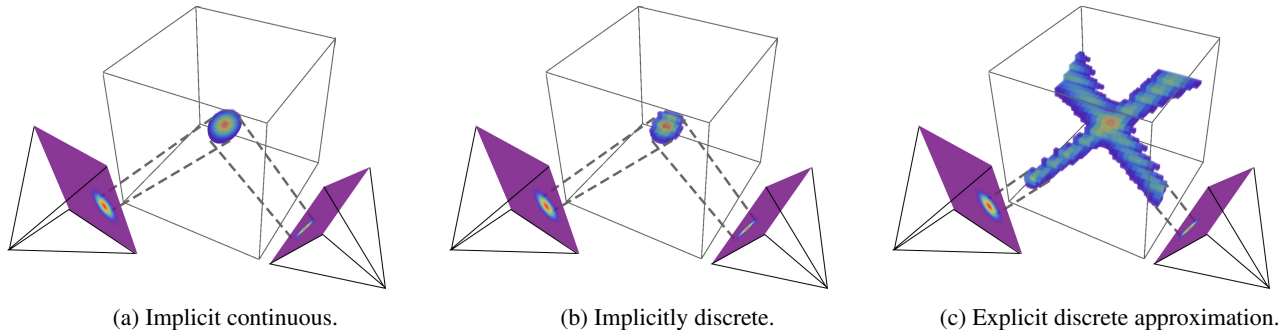


Figure 2: 3D heatmap under different definitions

This problem is tough to solve and exists multiple solutions. There are three main reasons for this:

1. If an iterative method is used to optimize  $y$  and  $X^{3D}$  alternately, the results tend to fall into local optimum.
2. If a neural network is used to regress  $y$  and then solve for  $X^{3D}$ , it will be constrained by the lack of diversity in the existing data set  $y$  to train.
3. If the neural network is used to regress  $X^{3D}$  directly from the monocular view and then to solve  $y$ , it will introduce the ill-conditioned problem of monocular scale ambiguity.

In summary, we have to extract more information from the only known quantity  $X_{2D}$  to obtain a relatively accurate estimate of  $y$ , i.e., a valid estimate  $p(y|X^{2D})$ .

We use camera pose distribution approximation and Monte Carlo sampling to solve this problem. (see Probabilistic Triangulation part of Figure 1) Specifically, we maintain a proposal distribution  $q(y)$  for approximating the posterior distribution  $p(y|X^{2D})$  of camera poses during one inference. Monte Carlo sampling is continuously performed on  $q(y)$ , and the weights are calculated from the sampled points, and then  $q(y)$  is updated according to the weights. After several iterations, the results generated from the sampled points are weighted and averaged as the input to the 3D pose reconstruction network. We refer to this process as Probabilistic Triangulation. Figure 1 shows the pipeline of uncalibrated 3D human pose estimation after adding Probabilistic Triangulation.

The Probabilistic Triangulation can be flexibly embedded into any kind of calibrated 3D human pose estimation, reducing its dependence on camera pose for realistic scenarios. The differentiable computational process makes it possible to connect the 2D pose estimation and 3D pose reconstruction parts, allowing end-to-end training.

### 3.2. Probabilistic Triangulation

We detail the Probabilistic Triangulation using the pipeline in Figure 1 as an example. In subsequent repre-

sentations,  $y$  is used to denote the camera pose and  $y_{gt}$  the ground truth of the camera pose. The 2D human pose is represented as  $X^{2D} = \{x_i^{2D} \in \mathbb{R}^{K \times H' \times W'} | i = 1 \dots N\}$ , and the 3D human pose is represented as  $X^{3D} = \{x_i^{3D} \in \mathbb{R}^{K \times L_1 \times L_2 \times L_3} | i = 1 \dots N\}$ , where  $H'$  and  $W'$  denote the height and width of the 2D heatmap, and  $L_1, L_2, L_3$  denote the length, width, and height of the 3D heatmap.

**Camera pose distribution approximation** In order to solve the optimization problem 2 with multiple solutions., we first normalized the multi-view camera pose to limit the range of solutions. Specifically, through an affine transformation of the whole coordinate system, the rotation matrix of the first view is changed into a unit matrix, the translation vector becomes a zero vector, and the translation vector of the second view has a modulus of one. The above affine changes can guarantee the uniqueness of the solution.

Each camera pose is represented as a 7-dimensional vector, where 3D rotations are represented using unit quaternions and 3D translations are represented using 3D vectors. To better express the camera distribution, the angular central Gaussian distribution is introduced to represent the 3D rotation [34], and the multivariate t-distribution is introduced to represent the 3D translation. The  $p(y|X^{2D})$  is calculated for each sampling point  $y$  to obtain the covariance matrix of the angular central Gaussian distribution, and the mean and covariance matrix of the multivariate t-distribution, which in turn updates the parameters. The initial parameters of the distributions are first estimated based on the 2D human pose using the eight-point method. Then, a portion is sampled around the initial value and another portion is sampled around the entire camera domain. The distribution is initialized by the sampling result. Since the input of a single frame may cause a large error in the eight-point method, and initializing the distribution in this way increases the robustness of the algorithm.

**Monte Carlo sampling** We assume that the 2D heatmap is the marginal distribution of the 3D heatmap along the projection direction, and the problem can be transformed into an optimization problem based on the reprojection error:

$$\min_{y, X^{3D}} \sum_i^N \int_{\sigma_i} \left\| x_i^{2D}(\sigma_i) - \int_{proj(z, y_i) = \sigma_i} x^{3D}(z) dz \right\|_2 d\sigma_i \quad (3)$$

where  $z$  denotes the body element in 3D space,  $\sigma_i$  denotes the area element of the  $i$ th camera plane, and  $proj(\cdot)$  denotes the projection function. A more general heatmap representation is adopted here, where  $x^{3D}$  and  $x^{2D}$  are considered as the probability density functions of the heatmap distribution. In practice, the heatmap is often stored in a discrete way, and the integrals need to be discretized as well. However, it is difficult to establish the constraints between  $y$  and  $X^{3D}$  explicitly, making the above optimization problem difficult to solve.

Therefore, we use an approximate 3D heatmap explicitly defined as:

$$\begin{aligned} x^{3D}(z) &= \frac{1}{N} \sum_{proj(z, y_i) = \sigma_i}^N x_i^{2D}(\sigma_i) \\ x_i^{2D}(\sigma_i) &= \max_{proj(z, y_i) = \sigma_i} x^{3D}(z) \end{aligned} \quad (4)$$

This definition is similar to the volume of the learnable triangulation, with the advantage that  $X^{3D}$  can be computed explicitly and differentially from  $y$  and  $X^{2D}$ . It also has some physical meaning as far as the location of the extreme points taken is similar to the original representation. The visualization of the 3D heatmap under different definitions is shown in Figure 2.

The generated  $X^{3D}$  retains the information of the heatmap and can be used as input for subsequent networks. Thus, the problem is simplified to:

$$\min_y \sum_i^N \underbrace{\left\| \sum_z (x^{3D}(z) x_i^{2D}(\sigma_i))^2 (x_i^{2D}(\sigma_i) - x^{3D}(z)) \right\|_2}_{f_i(y)} \quad (5)$$

The reprojection error is employed as the negative logarithm of the likelihood function  $p(X_{2D}|y)$ .

$$p(X^{2D}|y) = \exp - \sum_i^N \|f_i(y)\|_2 \quad (6)$$

Suppose that the camera pose  $p(y)$  is subject to a uniform distribution,  $p(y|X^{2D})$  can be calculated by Bayesian formula:

$$p(y|X^{2D}) = \frac{p(X^{2D}|y)}{E[p(X^{2D}|y)]} \quad (7)$$

The calculation of denominator expectation can be estimated by Monte Carlo sampling approximation. The specific calculation formula is as follows.

$$E[p(X^{2D}|y)] \approx \frac{1}{M} \sum_j^M q(y_j) p(X^{2D}|y), y_j \sim q(y) \quad (8)$$

where  $M$  denotes the total number of samples. Then,  $q(y)$  is updated using the computed  $p(y|X^{2D})$ .

Finally, the normalized  $q(y)$  is used as the weight and the  $X^{3D}$  generated by all sampled points  $y$  is weighted and averaged as the input to the subsequent 3D pose reconstruction network.

$$X^{3D} = \frac{\sum_j q(y_j) X_j^{3D}}{\sum_j q(y_j)} \quad (9)$$

### 3.3. Training Loss

The entire training process include two loss terms:  $\mathcal{L}_{3d}$  and  $\mathcal{L}_{cam}$ .

We measure the difference between the true distribution of camera pose  $z(y)$  and the posterior distribution  $p(y|X^{2D})$  using KL divergence.

$$\begin{aligned} \mathcal{L}_{cam} &= D_{KL}(z(y) \| p(y|X^{2D})) \\ &= -\frac{1}{2} \sum_i^N \|f_i(y_{gt})\|_2 + \log E[p(X^{2D}|y)] \end{aligned} \quad (10)$$

where the expectation of the second half is also obtained by the calculation of the equation 8.

The 3D human pose estimation loss is the same as previous works[18], which encourages the predicted 3d poses to be close to the ground truth. The formula can be summarized as follow:

$$\mathcal{L}_{3d} = \|\text{softargmax}(X^{3D}) - X_{gt}\|_1 - \beta \cdot \log(X^{3D}(X_{gt})) \quad (11)$$

In end-to-end training, we simultaneously optimize these two loss terms to train the networks, and the total loss function can be written as:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{cam} + \lambda_2 \mathcal{L}_{3d} \quad (12)$$

where the  $\lambda_1 = 1$  and  $\lambda_2 = 0.1$  in our experiment setting.

## 4. Experiments

### 4.1. Dataset & Metrics

We conducted experiments on two of the most commonly used 3D human pose datasets available today, Human3.6M[17] and CMU Panoptic[19, 29, 35]. We used

Method	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Mean
Calibration method																
Remelli et al.[28]	27.3	32.1	25.0	26.5	29.3	35.4	28.8	31.6	36.4	31.7	31.2	29.9	26.9	33.7	30.4	30.2
Bartol et al.[2]	27.5	28.4	29.3	27.5	30.1	28.1	27.9	30.8	32.9	32.5	30.8	29.4	28.5	30.5	30.1	29.1
He et al. [16]	25.7	27.7	23.7	24.8	26.9	31.4	24.9	26.5	28.8	31.7	28.2	26.4	23.6	28.3	23.5	26.9
Qiu et al. [26]	24.0	26.7	23.2	24.3	24.8	22.8	24.1	28.6	32.1	26.9	31.0	25.6	25.0	28.0	24.4	26.2
Ma et al. [22]	24.4	26.4	23.4	21.1	25.2	23.2	24.7	33.8	29.8	26.4	26.8	24.2	23.2	26.1	23.3	25.8
Iskakov et al.[18]	19.9	20.0	18.9	18.5	20.5	19.4	18.4	22.1	22.5	28.7	21.2	20.8	19.7	22.1	20.2	20.8
Uncalibration method																
Gordon et al.[14]	<b>22.0</b>	<b>23.6</b>	<b>24.9</b>	<b>26.7</b>	<b>30.6</b>	35.7	25.1	<b>32.9</b>	<b>29.5</b>	32.5	32.6	26.5	34.7	26.0	27.7	30.2
Ours method	24.0	25.4	26.6	30.4	32.1	<b>20.1</b>	<b>20.5</b>	36.5	40.1	<b>29.5</b>	<b>27.4</b>	<b>27.6</b>	<b>20.8</b>	<b>24.1</b>	<b>22.0</b>	<b>27.8</b>

Table 1: Results of the evaluation on the Human3.6M dataset. The table shows the MPJPE in millimeters for the published state-of-the-art calibrated and uncalibrated methods.

Method	MPJPE,mm
Calibrated method	
Iskakov et al. Algebraic[18]	21.3
Iskakov et al. Volumetric[18]	13.7
Uncalibrated method	
Bartol et al. [2]	25.4
Our method	24.2

Table 2: Results of the MPJPE on the CMU Panoptic dataset ( using four cameras).

the 17-joint MPJPE (Mean Per Joint Position Error) as a metric to evaluate our method. In training, we normalize the ground truth according to the method in section 3.2 to ensure the uniqueness in the camera pose solution. In validation, the estimated 3D human pose is denormalized to the original coordinate system to ensure the correctness of the MPJPE calculation.

## 4.2. Implementation details

We added the Probabilistic Triangulation based on the work learnable triangulation by Iskakov et al.[18] to realize uncalibrated 3D human pose estimation. The same experimental parameter settings as the learnable triangulation method were used. As with learnable triangulation, our method uses predictions from the algebraic triangulation. Our method can also be fine-tuned and be trained in an end-to-end manner.  $q(y)$  is initialized using a uniform distribution. The number of iterations is four, and the number of Monte Carlo samples is 256.

## 4.3. Comparisons with State-of-the-art

We compared the existing state-of-the-art calibrated and uncalibrated 3D pose estimation algorithms. The evaluation results on the Human3.6M dataset and the CMU Panoptic dataset are shown in Table 1 and Table 2, respectively. The experimental results show that the MPJPE of our method has outperformed the state-of-the-art uncalibrated approaches and achieved comparable performance to calibrated methods. It is worth noting that our method achieves better results in most scenarios of uncalibrated setting (see Table 1), but the results are poor in the Purch. and Sit. sub-datasets. This is because there are more occlusions in these two subsets, resulting in the 2D joint estimation being inaccurate, which further affects the estimation of the camera parameter distribution. Therefore, the final accumulated error affects the 3D human pose estimation performance. Inferring the camera parameters from the features of the detected human inevitably leads to estimation results depending on the capability of the human feature detector. However, our method still outperforms existing uncalibrated methods in terms of overall MPJPE for both datasets.

## 4.4. Evaluation on Camera Pose Estimation

To evaluate the capability of our camera pose estimation, we have conducted comparisons with some classical camera parameter estimation algorithms on the Human3.6M dataset. Due to the large parallax between different camera views, we choose 2D human keypoints estimated from 20 consecutive frames as image feature points to solve the camera pose parameters. We compare our camera estimation with Bundle Adjustment[32] and RANSAC 8-point algorithm[30]. Bundle Adjustment choose to minimize the reprojection error using the LM algorithm to find the camera parameters and the RANSAC 8-point algorithm uses the

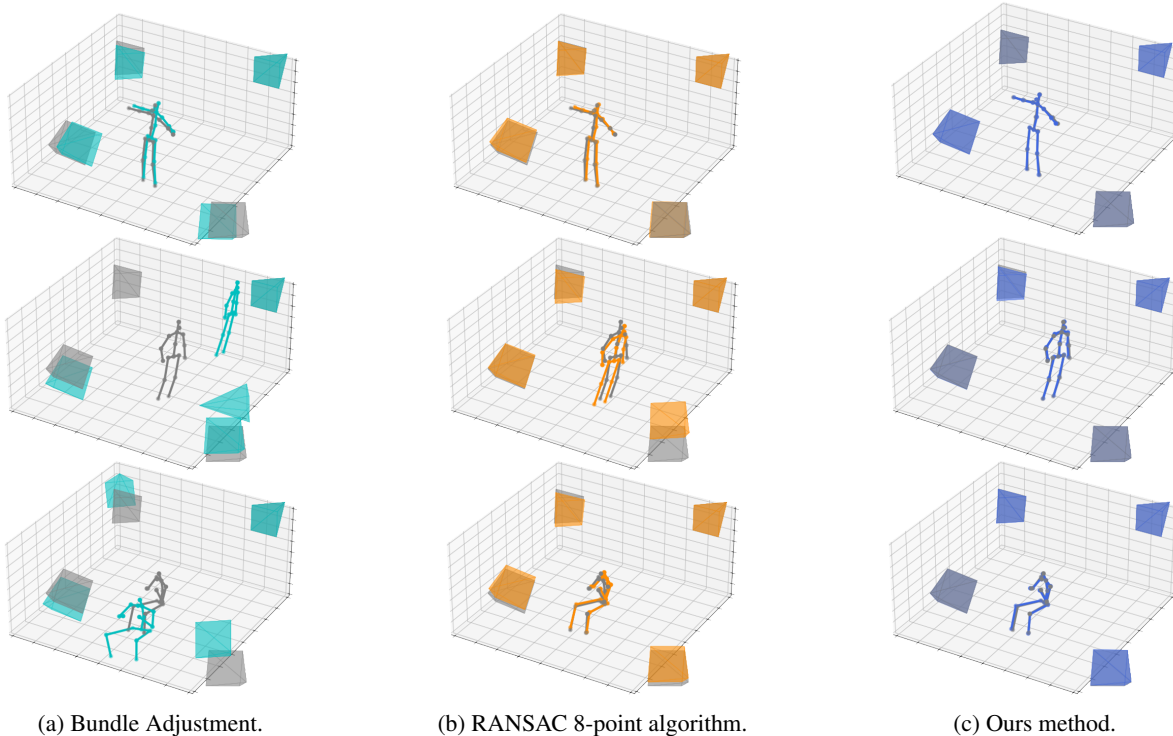


Figure 3: Visualization of the camera pose parameters and 3D human pose obtained by different algorithms. The quadrilateral cone indicates the camera pose parameters, the gray part indicates the ground truth, and the colored part indicates the prediction results of different algorithms.

8-point method to find the fundamental matrix after screening out the outliers by RANSAC, and then uses the eigenvalue decomposition to find the camera pose. We provide some visualization examples in Figure 3. From the Figure 3, we can see that Bundle Adjustment often falls into the local optimal solution, resulting in poor camera poses. The RANSAC 8-point algorithm can remove outliers, but the estimation of the camera pose is sensitive to 2D human pose estimation, which will lead to accumulated errors when reconstructing 3D human poses. The 3D human and camera pose estimation results are more robust and accurate when introducing our Probabilistic Triangulation (see (c) of Figure 3)

Table 3 shows more detailed results of camera pose evaluation, using the same experimental setup as [2]. Among the evaluation metrics used: 1. 3D estimation error(mm)  $E_{3D} = E[\|x^{\hat{3}D} - x^{3D}\|_2]$ ; 2. Projection 2D error(mm)  $E_{2D} = E[\|x^{\hat{2}D} - x^{2D}\|_2]$ ; 3. Camera rotation error  $E_R = E[\|\hat{q} - q\|_2]$ ; 4. Camera translation error (mm)  $E_t = E[\|\hat{t} - t\|_2]$ .

#### 4.5. Cross-dataset experiments

To test the generalization ability of Probabilistic Triangulation across datasets, we segment CMU Panoptic and

methods	$E_{3D} \downarrow$	$E_{2D} \downarrow$	$E_R \downarrow$	$E_t \downarrow$
BA	736.82	130.18	0.025	46.87
RANSAC 8-point	97.50	50.14	0.012	5.30
Our methods	<b>27.80</b>	<b>5.98</b>	<b>0.0043</b>	<b>2.71</b>

Table 3: Quantitative experiments on camera pose estimation on the Human3.6M dataset.

Human36M into five sub-datasets in the same way as Bartol et al.[2]. Cross-dataset experiments are conducted between these sub-datasets two by two. It was found that there was no significant loss of accuracy between the CMU Panoptic sub-datasets even after segmentation (this is because the distribution between the sub-datasets of the CMU Panoptic dataset is very similar). However, the results trained from the CMU dataset have a significant loss of accuracy on Human3.6M. Table 4 shows the generalization ability of Probabilistic Triangulation and Bartol et al.[2] from the CMU dataset to the Human3.6M dataset. Probabilistic Triangulation exhibits stronger generalization.

#### 4.6. Ablation study on the number of cameras

We show the results of the ablation study in Table 5. After training on 4-view data, our method generalizes well to

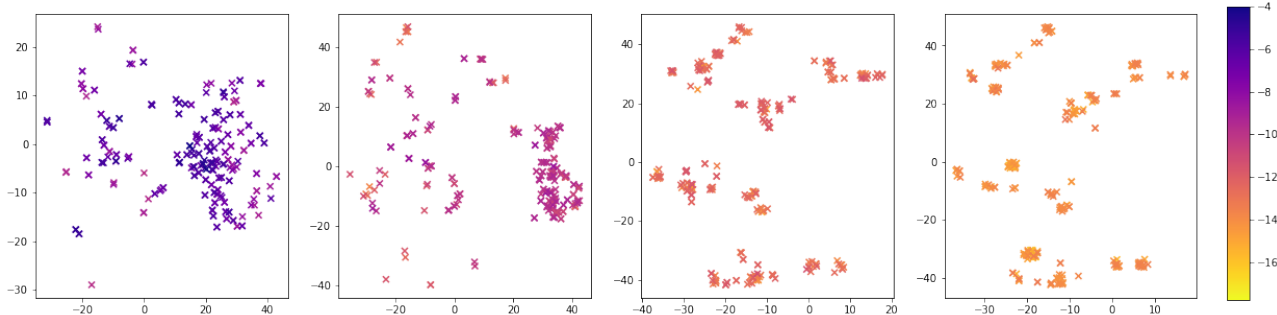


Figure 4: Visualization of the sampled points in four iterations after dimensionality reduction using t-SNE. From left to right corresponds to the sampled values after four iterations. The value of the sampled point is the logarithmic sampling loss, and the smaller the value means that the sampled point is closer to the true value.

Method	Train	Self-Test	H36M-Test	diff
Bartol et al.[2]	CMU1	25.8	33.5	30%
	CMU2	26.0	33.4	28%
	CMU3	25.0	31.0	24%
	CMU4	25.1	32.5	28%
Ours method	CMU1	24.6	30.8	25%
	CMU2	23.7	29.1	23%
	CMU3	24.1	29.2	21%
	CMU4	24.9	31.1	25%

Table 4: Experiments on generalization ability between datasets (MPJPE in mm). The CMU Panoptic dataset was split into four sub-datasets with different number of cameras and different spatial distributions of cameras. The models were trained on the four datasets separately, and the results were evaluated on their own test set and the Human3.6M test set. The last column represents the cross-data accuracy decline rate.

3-view and 2-view because the network has seen rich information during training. There is some performance degradation when the model is trained on 2-view and tested on 3-view and 4-view as shown in Table 5.

	2-view	3-view	4-view
Training on 2-view	29.5	42.4	39.2
Training on 4-view	32.6	29.3	27.8

Table 5: Ablation study on Human3.6M for camera numbers (MPJPE, mm).

#### 4.7. Ablation study on each module

In order to demonstrate the irreplaceable role that probabilistic triangularization plays in camera pose estimation and 3D human reconstruction in uncalibrated 3D human

pose estimation. We design two ablation experiments to remove the influence of the rest of the network on the results and to demonstrate the role of Probabilistic Triangulation better.

First, we use 2D heatmaps ground truth as the input of Probabilistic Triangulation to eliminate the effect of 2D backbone on the results. Also, the upper bound of our method is shown (see Table 6 middle). The table shows that our method achieves comparable results to the calibration method [18] when given the 2D ground truth.

The 3D error still cannot reach zero under the condition of using 2D ground truth because there is a theoretical upper limit to the accuracy of the voxel structure. However, the 2D human pose estimator error cannot be avoided, and the better robustness of the voxel structure to error is the reason it was chosen.

In addition, we demonstrate the necessity of introducing the camera pose estimation module by comparing it with Iskaov et al. (V2V network and triangulation). We rotated the world coordinate system ( $0^\circ$  to  $180^\circ$  randomly around the vertical axis and  $0^\circ$  to  $45^\circ$  randomly around the pitch) to change the values of the camera’s external parameters, resulting in a new test set called "Test\*" in Table 6. We can find that only using a V2V network and triangulation will make the network overfit the camera poses, resulting in performance degradation. Our method does not require the input of camera poses, so changes in the world coordinate system do not affect our method’s final pose estimation results. It demonstrates that our camera estimation module, i.e., the main contribution, is necessary and reasonable.

#### 4.8. Visualization of Sampling Process

In order to visualize the sampling process in the Probabilistic Triangulation, the 256 sampling points of each iteration are embedded by t-SNE and plotted in Figure 4. From the figure, as the iteration proceeds, the approximation of the camera distribution becomes more and more ac-



	Baseline	2D GT	Test*
Iskakov et al.	20.8	16.3	25.12
Our methods	27.8	18.5	27.8

Table 6: Pose estimation results (MPJPE, mm) on different 2D and test settings. The middle two columns show the results of 3D pose estimation using the 2D estimator and 2D GT, respectively. The rightmost column shows the estimation results on the Test\* set.

curate, the sampled points are closer to the ground truth, and the sampling loss keeps decreasing. At the same time, the whole camera pose space is not smooth, and there are many local optimums and pseudo-solutions. The 2D human pose estimator generates noise, making the classical camera pose estimation more likely to fall into local optimal solutions. Distributed representation of camera pose (Probabilistic Triangulation) can alleviate this problem.

Furthermore, from Figure 4, the sampling points are concentrated around a few extreme points as the iteration parameters increase. This is because an incremental solution exists to the camera pose estimation, so there are multiple extreme points. However, only one of them has real physical significance, whereas, in the projection process, these extreme points produce the same projection result. The distribution also tends to converge to a single extreme point if an initial value is used to initialize the camera pose distribution, and the sampling points of the distribution are spread out to appear near multiple extreme points if a uniform distribution is used to initialize the distribution.

#### 4.9. Generalize to Wide Scenes

To demonstrate the generalization of our Probabilistic Triangulation in the wild scenes, we use a model trained from the CMU Panoptic dataset and test it on the natural scenes. Two iPhones with a 26mm primary camera were used for the shooting, and the intrinsic parameters were read using the swift function. The results are shown in Figure 5. Our method remains in effect even when the camera space location and the number of cameras change.

### 5. Conclusions and limitations

We propose a novel Probabilistic Triangulation module for multi-view 3D human pose estimation, which enables the existing calibrated methods to reduce the dependence on calibration and can be applied to uncalibrated wild scenes. Extensive experiments demonstrate that the 3D human pose estimation based on our Probabilistic Triangulation can achieve state-of-the-art performance when compared to uncalibrated methods. And the employment of camera pose distribution and Monte Carlo sampling alleviates the problem of easy access to local optimums and pseudo-solutions

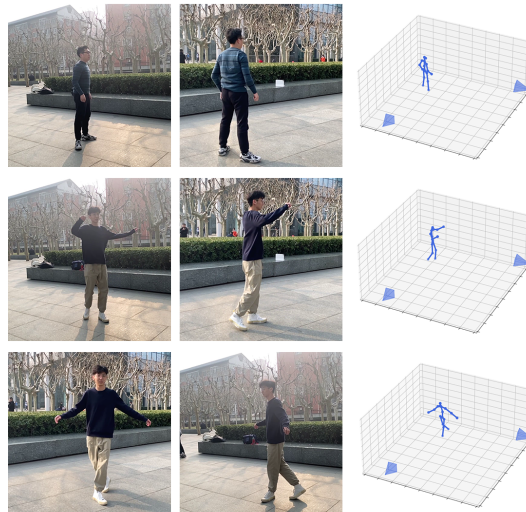


Figure 5: Our method demonstrates the results in a non-calibrated wild scene. The first and second columns represent the two iPhone camera views, and the third column shows the estimated camera pose and 3D human pose.

in camera pose estimation. Since our Probabilistic Triangulation does not take the camera pose parameters as input, it is possible to build the multi-view 3D human pose estimation system in wild scenes and achieve a high-accurate capture. In the wild scenes, the Probabilistic Triangulation is not affected by the spatial location changes and the number of cameras. It can be directly applied to uncalibrated scenes after training with calibrated data.

The main limitation of our method is that it can only be applied to single-person scenes. Extending to multi-person scenarios requires taking into account the cross-view matching problem, which is difficult to solve in the absence of a camera pose. In addition, although the Probabilistic Triangulation does not require the camera pose as input during validation, the ground truth of the 3D human pose and the camera pose are used in the training process, and these data need to be obtained from the dataset of the calibration scene. In fact, the ground truth of camera pose is not directly involved in the training, but indirectly guides the 2D pose estimation network learning through cross-view information. The ground truth of the 3D human pose influences the 2D pose estimation network and enables the 3D pose reconstruction network to learn the human prior and integrate the information generated by previous sampling.

**Acknowledgements** This work was supported by National Key Research and Development Program of China (NO. 2022YFB3303202).

## References

- [1] Sameer Agarwal and Keir Mierle. Others, “ceres solver,”. Available: <http://ceres-solver.org>, 2015.
- [2] Kristijan Bartol, David Bojanić, Tomislav Petković, and Tomislav Pribanić. Generalizable human pose triangulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11028–11037, 2022.
- [3] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures for multiple human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1669–1676, 2014.
- [4] Anastasiia Borodulina. Application of 3d human pose estimation of motion capture and character animation. 2019.
- [5] Simon Bultmann and Sven Behnke. Real-time multi-view 3d human pose estimation using semantic feedback to smart edge sensors. *arXiv preprint arXiv:2106.14729*, 2021.
- [6] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.
- [7] Bo Chen, Alvaro Parra, Jiewei Cao, Nan Li, and Tat-Jun Chin. End-to-end learnable geometric vision by backpropagating pnp optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8100–8109, 2020.
- [8] Hansheng Chen, Yuyao Huang, Wei Tian, Zhong Gao, and Lu Xiong. Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10379–10388, 2021.
- [9] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2781–2790, 2022.
- [10] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.
- [11] Yu Cheng, Bo Yang, Bo Wang, and Robby T Tan. 3d human pose estimation using spatio-temporal networks with explicit occlusion training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10631–10638, 2020.
- [12] Junting Dong, Wen Jiang, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Fast and robust multi-person 3d pose estimation from multiple views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7792–7801, 2019.
- [13] Wenbin Du, Yali Wang, and Yu Qiao. Rpan: An end-to-end recurrent pose-attention network for action recognition in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 3725–3734, 2017.
- [14] Brian Gordon, Sigal Raab, Guy Azov, Raja Giryes, and Daniel Cohen-Or. Flex: Parameter-free multi-view 3d human motion reconstruction. *arXiv preprint arXiv:2105.01937*, 2021.
- [15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [16] Yihui He, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu. Epipolar transformer for multi-view human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1036–1037, 2020.
- [17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 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):1325–1339, 2013.
- [18] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7718–7727, 2019.
- [19] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3334–3342, 2015.
- [20] Sang-Eun Lee, Keisuke Shibata, Soma Nonaka, Shohei Nobuhara, and Ko Nishino. Extrinsic camera calibration from a moving person. *IEEE Robotics and Automation Letters*, 7(4):10344–10351, 2022.
- [21] Wu Liu and Tao Mei. Recent advances of monocular 2d and 3d human pose estimation: A deep learning perspective. *ACM Computing Surveys (CSUR)*, 2022.
- [22] Haoyu Ma, Liangjian Chen, Deying Kong, Zhe Wang, Xingwei Liu, Hao Tang, Xiangyi Yan, Yusheng Xie, Shih-Yao Lin, and Xiaohui Xie. Transfusion: Cross-view fusion with transformer for 3d human pose estimation. *arXiv preprint arXiv:2110.09554*, 2021.
- [23] Tewodros Legesse Munea, Yalew Zelalem Jembre, Halefom Tekle Weldegebriel, Longbiao Chen, Chenxi Huang, and Chenhui Yang. The progress of human pose estimation: a survey and taxonomy of models applied in 2d human pose estimation. *IEEE Access*, 8:133330–133348, 2020.
- [24] Štěpán Obdržálek, Gregorij Kurillo, Jay Han, Ted Abresch, and Ruzena Bajcsy. Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. In *Medicine Meets Virtual Reality 19*, pages 320–324. IOS Press, 2012.
- [25] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 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*, pages 7753–7762, 2019.
- [26] Haibo Qiu, Chunyu Wang, Jingdong Wang, Naiyan Wang, and Wenjun Zeng. Cross view fusion for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4342–4351, 2019.

- [27] Daniele Regazzoni, Giordano De Vecchi, and Caterina Rizzi. Rgb cams vs rgb-d sensors: Low cost motion capture technologies performances and limitations. *Journal of Manufacturing Systems*, 33(4):719–728, 2014.
- [28] Edoardo Remelli, Shangchen Han, Sina Honari, Pascal Fua, and Robert Wang. Lightweight multi-view 3d pose estimation through camera-disentangled representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6040–6049, 2020.
- [29] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multi-view bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.
- [30] Frédéric Sur, Nicolas Noury, and Marie-Odile Berger. Computing the uncertainty of the 8 point algorithm for fundamental matrix estimation. In *19th British Machine Vision Conference-BMVC 2008*, page 10, 2008.
- [31] Kosuke Takahashi, Dan Mikami, Mariko Isogawa, and Hideaki Kimata. Human pose as calibration pattern; 3d human pose estimation with multiple unsynchronized and uncalibrated cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1775–1782, 2018.
- [32] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings*, pages 298–372. Springer, 2000.
- [33] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *European Conference on Computer Vision*, pages 197–212. Springer, 2020.
- [34] David E Tyler. Statistical analysis for the angular central gaussian distribution on the sphere. *Biometrika*, 74(3):579–589, 1987.
- [35] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10974, 2019.
- [36] Jianfeng Zhang, Yujun Cai, Shuicheng Yan, Jiashi Feng, et al. Direct multi-view multi-person 3d pose estimation. *Advances in Neural Information Processing Systems*, 34:13153–13164, 2021.
- [37] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. 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*, pages 13232–13242, 2022.
- [38] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3d human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11656–11665, 2021.
- [39] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: Unified pretraining for human motion analysis. *arXiv preprint arXiv:2210.06551*, 2022.