Faster VoxelPose: Real-time 3D Human Pose Estimation by Orthographic Projection

Hang Ye[1*], Wentao Zhu[2,3*], Chunyu Wang[4+],
Rujie Wu[2,3], and Yizhou Wang[2,3,5]

[1] Yuanpei College, Peking University
[2] Center on Frontiers of Computing Studies, Peking University
[3] School of Computer Science, Peking University
[4] Microsoft Research Asia
[5] Inst. for Artificial Intelligence, Peking University
{yehang, wtzhu, wu_rujie, yizhou.wang}@pku.edu.cn, chnuwa@microsoft.com

Abstract. While the voxel-based methods have achieved promising results for multi-person 3D pose estimation from multi-cameras, they suffer from heavy computation burdens, especially for large scenes. We present *Faster VoxelPose* to address the challenge by re-projecting the feature volume to the three two-dimensional coordinate planes and estimating X, Y, Z coordinates from them separately. To that end, we first localize each person by a 3D bounding box by estimating a 2D box and its height based on the volume features projected to the xy-plane and z-axis, respectively. Then for each person, we estimate partial joint coordinates from the three coordinate planes separately which are then fused to obtain the final 3D pose. The method is free from costly 3D-CNNs and improves the speed of VoxelPose by ten times and meanwhile achieves competitive accuracy as the state-of-the-art methods, proving its potential in real-time applications.

Keywords: 3D Human Pose Estimation, Multi-view Multi-person

1 Introduction

Estimating 3D human pose from RGB images is a fundamental problem in computer vision. It not only paves the way for some important downstream tasks such as action recognition [36, 39, 42, 43, 50] and human-computer interaction [1, 18], but also enables a wide range of applications, *e.g.* sports analysis [5, 29] and virtual avatar animation [46, 63].

While many works [8, 24, 27, 40, 41, 59, 61] address monocular 3D pose estimation, their application in serious scenarios is limited because of the degraded accuracy [20, 34]. In addition, monocular human pose estimation struggles when occlusion occurs which is ubiquitous in natural images [7, 10]. As a result, the

[*] Equal contribution.
[+] Corresponding author.

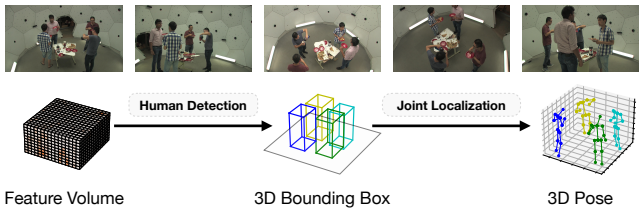

Fig. 1. Multi-view 3D Pose Estimation. Given multi-view images and camera parameters, the task aims to estimate the 3D poses of all people in the world coordinates. Similar to [38], our approach is based on the volumetric representation and detects 3D box as an intermediate step.

state-of-the-art 3D human pose estimation results are usually obtained via multi-camera systems which consist of a group of synchronized and calibrated wide-baseline cameras [2, 9, 22, 38, 47, 55, 57, 58].

Simple triangulation [14,51] can achieve accurate 3D pose estimates if the 2D poses in all views are accurate. However, 2D pose estimates may have errors in practice especially when occlusion occurs. To address the problem, voxel-based methods [17, 28, 31, 38, 47, 54] have been proposed which inversely project 2D features or heatmaps in each view to the 3D space and then fuse the multi-view features. The resulting feature volume is more robust to occlusions in individual cameras. Then they apply a 3D-CNN to estimate the 3D positions of the body joints from the feature volume. While these methods achieve very accurate results, the computation complexity increases cubically with space size. As a result, they cannot support real-time inference for large scenes such as sports stadiums or retail stores.

In this work, we present Faster VoxelPose which is about ten times faster than VoxelPose on the common benchmarks and more importantly scales gracefully to large spaces. Inspired by technical drawing where a 3D object is often unambiguously represented by three 2D orthographic projections, *i.e.* plan, elevation and section, we re-project the previously fused 3D volumetric features to the three 2D coordinate planes by orthographic projection and estimate partial coordinates, *e.g.* xy, xz and yz, of a 3D pose from each of the 2D planes, which are then fused by a tiny network to predict xyz. The main advantage of the method is that we can replace the expensive 3D-CNNs with 2D-CNNs which reduces the computation cost from $O(n^3)$ to $O(n^2)$ where n is the spatial resolution. However, the factorization brings two new challenges. First, people that are far away in the 3D space could overlap in some planes after re-projection, which may bring severe ambiguity to the corresponding features. Second, the estimation results may be inconsistent across planes so we need a strategy to aggregate the contradictory predictions.

We address the challenges from two aspects. Firstly, as shown in Fig. 1, we present *Human Detection Networks* (HDN) to estimate a *tight* 3D box for each person which is used to filter out the features of other people. By contrast,

VoxelPose [38] use a loose fixed-size 3D bounding box. In particular, we re-project the 3D feature volume to the xy plane by max-pooling along the z axis (bird's-eye view), and apply a 2D-CNN to localize people by a 2D box in the xy plane. Then, for each bounding box, we obtain a 1D "column" feature representation from the volume at the box center along the z axis, and apply a 1D-CNN to estimate the vertical position of the box center.

Then we present *Joint Localization Networks* to estimate a 3D pose for each 3D box. We first mask out the features in the volume which are outside the box to reduce the impact of other people, obtaining *person-specific* feature volume. We re-project the masked volume to the three coordinate planes and estimate the X, Y and Z coordinates, respectively. For each coordinate, we have two predictions from two planes. It is probable that the two predictions are contradictory so we propose a fusion network to learn a weight for each prediction and aggregate them to obtain the final 3D pose.

Our approach achieves competing results as the baseline method which uses 3D-CNN. But ours is about 10 times faster than it (speed improvement is larger for larger scenes). Our contributions are four-fold: 1) We design a lightweight framework for efficient training and inference of the multi-view multi-person 3D pose estimation problem. Our approach demonstrates that 3D human detection and pose estimation can be resolved on the re-projected 2D feature maps with careful design. 2) We propose a novel 3D human detector that disentangles ground plane localization and height estimation. 3) We utilize 3D bounding box for feature masking, which contributes to person-specific feature volume and improves joint localization accuracy. 4) We deploy the confidence regression networks to adaptively fuse the estimates on the re-projected planes to compensate for their individual accuracy loss. While we focus on pose estimation in this work, the idea may also benefit other voxel-based tasks such as object detection [23, 33, 60] and shape completion [45].

2 Related Work

2.1 Multi-view 3D Pose Estimation

For the single-person case, the key is to handle 2D pose estimation errors in individual planes. Iskakov *et al*. [17] designed differentiable triangulation which uses joint detection confidence in each camera view to learn the optimal triangulation weights. Pavlakos *et al*. [26] applied CNN with 3D PSM for markerless motion capture. Qiu *et al*. [28] used epipolar lines to guide cross-view feature fusion followed by a recurrent PSM. Epipolar transformer [15] extended [28] to handle dynamic cameras. Generally speaking, single-person 3D pose estimation has achieved satisfactory results when there are sufficient cameras to guarantee that every body joint can be seen from at least two cameras.

Multi-person 3D human pose estimation is more challenging because it needs to solve two additional sub-tasks: 1) Identifying joint-to-person association in different views. 2) Handling mutual occlusions among the crowd. To address

the first challenge, various association strategies are proposed based on re-id features [9], dynamic matching [2], 4D graph cut [56], and plane sweep stereo [22]. However, in crowded scenes, noisy 2D pose estimates would harm their accuracy. To address the second challenge, Belagiannis *et al.* [2] extended PSM for multi-person. Wang *et al.* [44] propose a transformer-based direct regression model with projective attention.

Recently, voxel-based methods [30, 38, 47, 54] are proposed to avoid making decisions in each camera view. Instead, they fuse multi-view features in the 3D space and only make the decision there. Such methods are free from pairwise reasoning of camera views and enable learning human posture knowledge in a data-driven way. However, the computation-intensive 3D convolutions prevent these approaches from being real-time and applicable to large spaces. Our method enjoys the benefit of volumetric feature aggregation, meanwhile being significantly faster and more scalable.

2.2 Efficient Human Pose Estimation

Designing efficient human pose estimators has been intensively studied for practical usage. For extracting 2D pose from images, state-of-the-art methods [21, 25, 35, 49, 52] have achieved real-time inference speed. In terms of multiview 3D pose estimation, Bultman *et al.* [6] explores an efficient system using edge sensors. Remelli *et al.* [32] and Fabbri *et al.* [12] adopt encoder-decoder networks to reduce computation, but they are not applicable to the multi-person setting. Most recently, Lin *et al.* [22] and Wang *et al.* [44] present alternative solutions to volumetric methods [38, 47, 54] and show some speed improvement. Nevertheless, these methods are capped in terms of scalability, which prevents them from being deployed to large scenes. Our method is complementary to state-of-the-art lightweight 2D pose estimators, and can further improve the speed of other volumetric methods [12, 30, 47].

3 Method

3.1 Overview

Without loss of generality, we explain our motivation with a simple case in which there is only one person. As shown in Fig. 2 (A), the input to our approach is a 3D feature volume $\mathbf{V} \in \mathbb{R}^{K \times L \times W \times H}$ which is constructed by back-projecting the 2D pose heatmaps in multiple cameras to the 3D voxel space [38]. The 2D pose heatmaps are extracted from the images using an off-the-shelf pose estimation model [37]. $L \times W \times H$ represents the number of voxels that are used to discretize the space and K represents the number of joint types. The volume approximately encodes the per-voxel likelihood of body joints.

In Fig. 2 (A), we show a 3D joint of interest, *e.g.* a shoulder joint, as $P = (X, Y, Z)$. In general, the corresponding feature volume should have a distinctive pattern around P so that it can be localized by expensive 3D-CNNs [38].

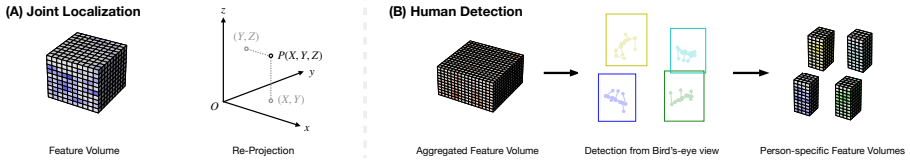

Fig. 2. Problem Decomposition. (A): Considering a single person, we re-project its feature volume to the coordinate planes with orthographic projection. The partial coordinates can be estimated by 2D CNN and assembled to 3D estimation. (B): Multi-person brings the extra challenge of ambiguity and occlusion. Nonetheless, people can be easily isolated from the bird's-eye view of the aggregated feature volume. Based on the intuitive ideas, we develop the lightweight *Joint Localization Networks* and *Human Detection Networks* respectively.

To reduce the computation cost, we re-project the volume to the three coordinate planes (*i.e.* the xy, yz, xz planes), respectively, resulting in three 2D feature maps. We can imagine that there are also distinctive patterns at the corresponding locations of each 2D feature map, *e.g.* (X, Y) at the xy plane, which can be similarly detected by 2D-CNNs. Then the 3D position of P can be assembled from the estimated coordinates in the three planes.

However, when we apply the idea to the multi-person scenario, we are confronted with new challenges. The features of different people may be mixed together after being projected to the coordinate planes even when they are far away from each other in the 3D space. This may corrupt the pose estimation accuracy. Inspired by top-down 2D pose estimation [13], the problem can be alleviated by "cropping" the person from the overall 3D space and only projecting features belonging to the person to the planes. So the remaining task is to detect each person in the 3D space efficiently. We utilize the prior that people barely overlap along the z axis, therefore they can be easily detected in the bird's-eye view as shown in Fig. 2 (B).

We take a two-phase approach to address the challenges. In the first phase, we present *Human Detection Networks* (Section 3.2) which efficiently detects all people from the bird's-eye view by 3D bounding boxes, ensuring that only the person-of-interest features are passed to the next phase. The second phase conducts fine-grained pose estimation for each person with *Joint Localization Networks* (Section 3.3), which is greatly eased since occlusion and distraction are mostly eliminated in the first phase. Importantly, all the operators in the networks are on 2D and 1D features, which boosts the speed.

3.2 Human Detection Networks

We first apply HRNet [37] to estimate 2D pose heatmaps from the multiview images, and construct an aggregated feature volume $\mathbf{V} \in \mathbb{R}^{K \times L \times W \times H}$ by back-projecting the heatmaps to the 3D voxel space. Since people are usually on the ground plane and it is less probable that one person is right on top of another,

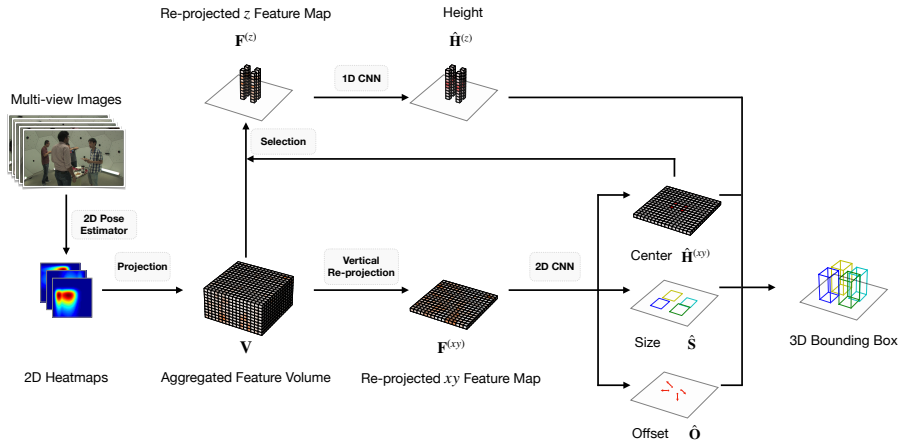

Fig. 3. Human Detection Networks. We first construct the feature volume $\mathbf{V}$ from the multi-view images. It is then projected to the xy plane to obtain the feature map $\mathbf{F}^{(xy)}$ (bird's-eye view). A Multi-branch 2D CNN estimates three feature maps encoding each person's center position, bounding box size, and center offset, respectively. We then select the 1D columns feature $\mathbf{F}^{(z)}$ from the positions with high confidence values on $\hat{\mathbf{H}}^{(xy)}$. Then a 1D CNN estimates the heatmap $\hat{\mathbf{H}}^{(z)}$ of the vertical position of the 3D box center. Finally, HDN outputs the combined 3D bounding box.

it inspires us to construct a 2D bird's-eye view representation from the feature volume for efficiently detecting people.

Detection in xy Plane We re-project the aggregated feature volume to the ground plane (xy) by performing max-pooling along the z direction and obtain $\mathbf{F}^{(xy)} \in \mathbb{R}^{K\times L\times W}$. Then we feed $\mathbf{F}^{(xy)}$ to a 2D fully convolutional network to detect the locations of people in the xy plane. The positions of all people in the plane are encoded by a 2D confidence map $\hat{\mathbf{H}}^{(xy)} \in [0,1]^{L\times W}$ whose value $\hat{\mathbf{H}}^{(xy)}_{i,j}$ represents the likelihood of human presence at the location (i,j). For training supervision, we generate the ground-truth (GT) 2D confidence map $\mathbf{H}^{(xy)}$. Its values are computed by the distance between the GT center point and each grid point using a Gaussian kernel. Specifically, the confidence value of grid point (i,j) is computed by:

$$\mathbf{H}^{(xy)}_{i,j} = \max_{1\le n\le N} \exp\{-\frac{(i-\tilde{i_n})^2+(j-\tilde{j_n})^2}{2\sigma^2}\}$$

where N denotes the number of persons and $(\tilde{i_n},\tilde{j_n})$ represents the corresponding GT position for person n. We just keep the largest scores in the presence of multiple people. The mean squared error (MSE) loss is computed by:

$$\mathcal{L}_{2d} = \sum_{i=1}^{L}\sum_{j=1}^{W} \|\mathbf{H}^{(xy)}_{i,j} - \hat{\mathbf{H}}^{(xy)}_{i,j}\|_2 \tag{1}$$

We further estimate a 2D box size for each person instead of assuming a loose constant size as in the previous work [38]. The height of the box is simply set to be 2000mm. This is critical to isolate the interference of multiple people, especially in crowded scenes. Our model generates a box size embedding at all grid points, denoted as $\hat{\mathbf{S}} \in \mathbb{R}^{2\times L\times W}$. But only those at the locations with large confidences are meaningful. We compute a ground-truth size embedding $\mathbf{S}$ based on box annotations.

During training, we only compute losses on the grid points which are adjacent to the ground-truth box centers. Specifically, for a 2D GT box center $(\tilde{x}, \tilde{y})$, we only add supervision on the discretized grid points $(\lfloor \frac{\tilde{x}}{l} \rfloor, \lfloor \frac{\tilde{y}}{w} \rfloor)$, where l represents the length of a single voxel and w denotes the width. Let $\mathbf{U}$ denote the set of the neighboring points mentioned above and suppose N is the number of persons in the image. We compute an L_1 loss at each center point in $\mathbf{U}$:

$$\mathcal{L}_{size} = \frac{1}{N} \sum_{(i,j)\in\mathbf{U}} \|\mathbf{S}_{i,j} - \hat{\mathbf{S}}_{i,j}\|_1 \tag{2}$$

In addition, to reduce the quantization error, we estimate the local offset for each root joint on the horizontal plane. Similar to size estimation, the model outputs an offset prediction at each grid point, denoted as $\hat{\mathbf{O}} \in \mathbb{R}^{2\times L\times W}$. We also generate a GT offset prediction $\mathbf{O}$ and use an L_1 loss on the neighboring points:

$$\mathcal{L}_{off} = \frac{1}{N} \sum_{(i,j)\in\mathbf{U}} \|\mathbf{O}_{i,j} - \hat{\mathbf{O}}_{i,j}\|_1 \tag{3}$$

Inspired by [62], we use a simple network structure with three parallel branches to estimate the heatmap, offset and size respectively. As shown in Fig. 3, the 2D bird's-eye features are passed through a fully-convolutional backbone network and then fed into three separate branches with identical designs, which consist of a 3×3 convolution, ReLU, and another 1×1 convolution.

Detection in z Axis The remaining task is to estimate the center height for each proposal. Firstly, we obtain the proposals with P largest confidences on the 2D heatmap $\hat{\mathbf{H}}^{(xy)}$ after applying non-maximum suppression (NMS). We set $P = 10$ in all the experiments. Subsequently, we extract the corresponding 1D "columns" for each proposal from the aggregated feature volume $\mathbf{V}$, denoted as $\mathbf{F}^{(z)} \in \mathbb{R}^{P\times K\times H}$, which is then fed into a 1D fully convolutional network to regress the height. Similar to 2D detection, our model generates 1D heatmap estimation $\hat{\mathbf{H}}^{(z)} \in [0,1]^{P\times H}$, indicating the likelihood of human presence at every possible height. We compute a GT 1D heatmap $\mathbf{H}^{(z)}$ for each proposal based on its center height using the Gaussian distribution. Likewise, we use an MSE loss here:

$$\mathcal{L}_{1d} = \frac{1}{P} \sum_{p=1}^{P} \sum_{k=1}^{H} \|\mathbf{H}^{(z)}_{p,k} - \hat{\mathbf{H}}^{(z)}_{p,k}\|_2 \tag{4}$$

Finally, we select the height with maximum confidence and by combining it with the 2D box center, offset and size, we can obtain the 3D bounding box. The overall confidence score for each box is computed by multiplying the scores of the 2D heatmap and 1D outputs. According to the exponential property of the Gaussian function, it can be regarded as an approximate of the 3D Gaussian distribution. We set a threshold for confidence scores to select the valid proposals. To sum up, the overall training objective is as follows:

$$\mathcal{L}_{HDN} = \mathcal{L}_{2d} + \lambda_{size}\mathcal{L}_{size} + \lambda_{off}\mathcal{L}_{off} + \lambda_{1d}\mathcal{L}_{1d} \tag{5}$$

where we set $\lambda_{size} = 0.02$, $\lambda_{off} = 0.1$ and $\lambda_{1d} = 1$.

3.3 Joint Localization Networks

Person-specific Feature Volume. With the bounding box of each person, we construct its fine-grained feature volume to predict the final 3D pose. We first crop a smaller feature volume $\mathbf{V}'$ from $\mathbf{V}$ centered at the box center with a fixed size (*i.e.* 2m $\times$ 2m $\times$ 2m). It suffices to cover arbitrary poses and maintains the relative scale of the motion space. The space is then divided into $L' \times W' \times H'$ voxels. Now the key step is to **zero out** the features outside the estimated bounding box to get the person-specific feature volume $\mathbf{V_s}$. This masking mechanism reduces the distraction of other people and enables safe volume re-projection in the following stage.

Joint Localization. To reduce the computational cost, we re-project $\mathbf{V_s}$ onto three orthogonal 2D planes, *i.e.* the *xy* plane, *xz* plane and *yz* planes in the world coordinate systems. Let $\mathbf{P}^{(xy)} \in \mathbb{R}^{K\times L'\times W'}$, $\mathbf{P}^{(xz)} \in \mathbb{R}^{K\times L'\times H'}$ and $\mathbf{P}^{(yz)} \in \mathbb{R}^{K\times W'\times H'}$ denote the re-projected feature maps corresponding to the three planes, respectively. Again, we use max-pooling for feature projection.

Subsequently, they are concatenated as a batch and fed to a 2D CNN for joint localization, as shown in Fig. 4. Note that we set the same granularity of voxels on different axes to enable parallel estimation, *i.e.* $L' = W' = H'$. The 2D CNN produces a joint-wise heatmap estimation for each re-projection plane, denoted as $\hat{\mathbf{H}}^{(t)}(t \in \{xy, xz, yz\})$ in the same shape of $\mathbf{P}^{(t)}$. To reduce the quantization error, we compute the center of mass of $\hat{\mathbf{H}}^{(t)}$ instead of taking the maximum responses. Specifically, the estimated positions $\hat{\mathbf{J}}^{(t)} \in \mathbb{R}^{K\times 2}$ are computed by:

$$\hat{\mathbf{J}}^{(xy)} = \sum_{i=1}^{L}\sum_{j=1}^{W}(i,j)\cdot\hat{\mathbf{H}}^{(xy)}_{i,j}, \hat{\mathbf{J}}^{(xz)} = \sum_{i=1}^{L}\sum_{k=1}^{H}(i,k)\cdot\hat{\mathbf{H}}^{(xz)}_{i,k}, \hat{\mathbf{J}}^{(yz)} = \sum_{j=1}^{W}\sum_{k=1}^{H}(j,k)\cdot\hat{\mathbf{H}}^{(yz)}_{j,k} \tag{6}$$

We supervise the estimations with the ground-truth 2D location $\mathbf{J}^{(t)} \in \mathbb{R}^{K\times 2}$ on each plane. An L_1 loss is computed by:

$$\mathcal{L}_{hm} = \sum_{t}\sum_{k=1}^{K}\|\mathbf{J}^{(t)}_k - \hat{\mathbf{J}}^{(t)}_k\|_1 \tag{7}$$

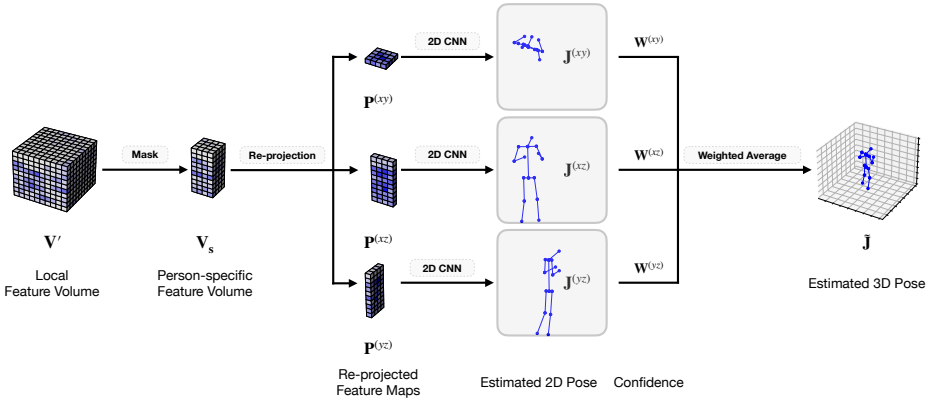

Fig. 4. Joint Localization Networks. For each person, we first construct its local feature volume $\mathbf{V}'$. The person-specific feature volume $\mathbf{V_s}$ is obtained by masking $\mathbf{V}'$ with the detected 3D box. We re-project $\mathbf{V_s}$ to three orthogonal coordinate planes to get the 2D feature maps $\mathbf{P}^{(t)}$. A shared 2D pose estimator regresses the joint locations $\mathbf{J}^{(t)}$ for each plane, and a confidence network computes the corresponding weights $\mathbf{W}^{(t)}$. Finally, the 3D pose $\tilde{\mathbf{J}}$ is computed by weighting $\mathbf{J}^{(t)}$ with $\mathbf{W}^{(t)}$ in a pairwise manner. $(t \in \{xy, xz, yz\})$

Adaptive Weighted Fusion. The quality of $\mathbf{P}^{(t)}$ and the difficulty of pose estimation naturally vary with the re-projection plane and human pose, thus we hope the model could learn to discriminate and balance the estimations from different planes automatically. To achieve this, we introduce a lightweight confidence regression network. We assume that the pattern of $\hat{\mathbf{H}}^{(t)}$ could reflect the quality of 2D pose estimation. Therefore, the estimated heatmaps $\hat{\mathbf{H}}^{(t)}$ are fed into a shared confidence regression network. Inspired by [58], we adopt a simple design for the confidence regression network, consisting of a convolutional layer, a global average pooling layer and one fully-connected layer.

The network generates joint-wise fusion weight for each plane, denoted as $\mathbf{W}^{(t)} \in \mathbb{R}^K$. We then use the Softmax function for normalization in a pair-wise manner and obtain the final 3D prediction $\tilde{\mathbf{J}} \in \mathbb{R}^{K\times 3}$. Specifically, for the joint k, the final estimations can be computed by:

$$\begin{aligned}
\tilde{\mathbf{J}}_{k,1} &= \text{softmax}(\mathbf{W}_k^{(xy)}, \mathbf{W}_k^{(xz)}) \cdot (\hat{\mathbf{J}}_{k,1}^{(xy)}, \hat{\mathbf{J}}_{k,1}^{(xz)}) \\
\tilde{\mathbf{J}}_{k,2} &= \text{softmax}(\mathbf{W}_k^{(xy)}, \mathbf{W}_k^{(yz)}) \cdot (\hat{\mathbf{J}}_{k,2}^{(xy)}, \hat{\mathbf{J}}_{k,1}^{(yz)}) \\
\tilde{\mathbf{J}}_{k,3} &= \text{softmax}(\mathbf{W}_k^{(xz)}, \mathbf{W}_k^{(yz)}) \cdot (\hat{\mathbf{J}}_{k,2}^{(xz)}, \hat{\mathbf{J}}_{k,2}^{(yz)})
\end{aligned} \tag{8}$$

where $\hat{\mathbf{J}}_{k,1}^{(xy)}$ denotes taking the first component of the 2D estimated coordinates of $\hat{\mathbf{J}}_k^{(xy)}$, namely the component on the x-axis, and the other notations have similar interpretations. Let $\mathbf{J}$ denote the GT 3D pose, we use an L_1 loss to train

the confidence regression network:

$$\mathcal{L}_{conf} = \sum_{k=1}^{K} \|\mathbf{J}_k - \tilde{\mathbf{J}}_k\|_1 \tag{9}$$

Now we get the overall training objective of JLN as follows. In our experiments, we set $\lambda_{conf} = 1$.

$$\mathcal{L}_{JLN} = \mathcal{L}_{hm} + \lambda_{conf}\mathcal{L}_{conf} \tag{10}$$

4 Experiments

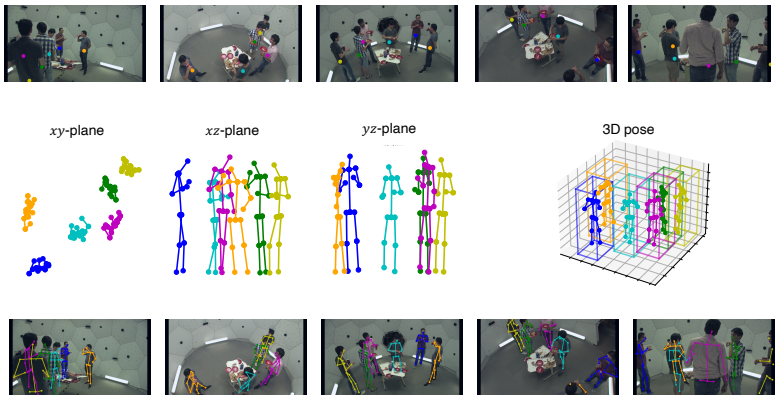

Fig. 5. Qualitative Results on the CMU Panoptic Dataset. The first row illustrates the estimated root joints in HDN. The second row shows the estimated 2D poses on the three orthogonal re-projection planes and the fused 3D pose in JLN. The last row shows the 2D back-projection of the estimated 3D pose to each camera view.

4.1 Setup

Datasets. The Shelf [2] dataset captures four people disassembling a shelf using five cameras. We select the frames of test set following previous works [22, 38]. The Campus [2] dataset captures multiple people interacting with each other in an outdoor environment shot by three cameras. The CMU Panoptic [19] dataset captures multiple people engaging in social activities. We use the same training and testing sequences captured by five HD cameras as in [22, 38].

Training Strategies. Due to incomplete annotations of Shelf and Campus, we use synthetic 3D poses to train the model for the two datasets, following [22, 38].

Table 1. Quantitative Evaluation of HDN. We measure the mean center error, precision and recall rate to evaluate the quality of human center detection and offset regression. The IoU score is computed between the estimated horizontal bounding box and GT, which additionally reflects the precision of bounding box size estimation.

Mean Center Error (mm)	Precision	Recall	IoU
53.73	0.9982	0.9985	0.757

For the Panoptic dataset, we first finetune the 2D heatmap estimation network. Then we fix the 2D network and train the 3D networks following [38].

Evaluation Metrics. Following the common practice, we compute the Percentage of Correct Parts (PCP3D) metric on Shelf and Campus. Specifically, we pair each GT pose with the closest estimation and calculate the percentage of correct parts. For the Panoptic dataset, we adopt the Average Precision (AP_K) and Mean Per Joint Position Error (MPJPE) as metrics, which reflect the quality of multi-person 3D pose estimation more comprehensively. In addition, we measure the inference time and frame per second (FPS) on the Panoptic dataset.

4.2 Evaluation and Comparison

Evaluation of HDN. We first evaluate the performance of the Human Detection Networks qualitatively. As Fig.5 shows, our model is able to detect the human centers and estimate the 3D bounding boxes as intended, despite the fact that severe occlusion occurs in all views. Accurate 3D bounding boxes help to isolate the persons for the fine-grained joint localization. In addition, we quantitatively measure the performance of HDN in terms of both center position and bounding box. As Tab.1 shows, our HDN localizes the root joint well, and the regressed bounding boxes overlap with GT mostly. The mean center error is larger than the MPJPE of JLN because JLN involves detailed pose estimation on a finer voxel granularity. Still, the center precision suffices to provide a reasonable 3D bounding box for joint localization.

Evaluation of JLN. We compare the 3D pose estimation performance with the state-of-the-art (SOTA) multi-view multi-person 3D pose estimation methods on Shelf and Campus [2]. While the proposed method is primarily optimized for inference efficiency and makes several approximations, it performs competitively with SOTA as shown in Tab.2. On the Shelf dataset, it outperforms the SOTA volumetric approach VoxelPose [38] which features fully 3D convolutional architecture. We also train and test on the Panoptic [19] dataset following the most recent works [22, 38]. As shown in Tab. 3, our method receives an extra per-joint error of about 2mm. We argue that the error margin is within an acceptable range given the speed-accuracy trade-off in real-time applications.

Table 2. Comparison with SOTA on Campus and Shelf. We compute the PCP3D (Percentage of Correct Parts) metrics following previous work. A part is considered correct if its distance with GT is at most half of the limb length.

	Shelf				Campus			
Method	Actor1	Actor2	Actor3	Average	Actor1	Actor2	Actor3	Average
Belagiannis *et al.* [2]	66.1	65.0	83.2	71.4	82.0	72.4	73.7	75.8
Belagiannis *et al.* [4]	75.0	67.0	86.0	76.0	83.0	73.0	78.0	78.0
Belagiannis *et al.* [3]	75.3	69.7	87.6	77.5	93.5	75.7	84.4	84.5
Ershadi-Nasab *et al.* [11]	93.3	75.9	94.8	88.0	94.2	92.9	84.6	90.6
Dong *et al.* [9]	98.8	94.1	97.8	96.9	97.6	93.3	98.0	96.3
Huang *et al.* [16]	98.8	96.2	97.2	97.4	98.0	94.8	97.4	96.7
Tu *et al.* [38]	99.3	94.1	97.6	97.0	97.6	93.8	98.8	96.7
Lin *et al.* [22]	99.3	96.5	98.0	97.9	98.4	93.7	99.0	97.0
Wang *et al.* [44]	99.3	95.1	97.8	97.4	98.2	94.1	97.4	96.6
Ours	99.4	96.0	97.5	97.6	96.5	94.1	97.9	96.2

Table 3. Comparison with SOTA on Panoptic. For efficiency metrics, We measure the average per-sample inference time on Panoptics test set (5 camera views, 3.41 person per frame). The measurement is done on a Linux machine with GPU GeForce RTX 2080 Ti and CPU Intel(R) Xeon(R) CPU E5-2699A v4 @ 2.40GHz. Batch size is set to be 1 for all methods.

Method	AP_{25}	AP_{50}	AP_{100}	AP_{150}	MPJPE	Time	FPS
VoxelPose [38]	83.59	98.33	99.76	99.91	17.68mm	316.0ms	3.2
PlaneSweepPose [22]	92.12	98.96	99.81	99.84	16.75mm	234.3ms	4.3
MvP [44]	92.28	96.60	97.45	97.69	15.76mm	278.8ms	3.6
Ours	85.22	98.08	99.32	99.48	18.26mm	32.2ms	31.1

Efficiency. We first compare the inference speed of our method to the SOTA methods, and then conduct an in-depth efficiency analysis. The speed results of other methods are obtained using their official codes on the same hardware as ours. For a fair comparison, we set the batch size to be one for all methods during inference following [44] to simulate the real-time use case where data arrives frame by frame. The batch size of PlaneSweepPose [22] was set to be 64 in the original paper so their reported speed is different from the one reported in this paper. For all the methods, the off-the-shelf 2D pose estimator time is not measured following [22, 44]. The results on the Panoptic dataset are shown in Tab. 3. Our approach shows a considerable advantage in terms of inference speed and supports real-time inference.

The inference time broken down per module is shown in Fig. 6. The "others" parts mainly consist of data preparation and feature volume construction. For HDN, the time cost is independent of the number of cameras and persons. For JLN, the theoretical computation complexity is linear to the number of persons. In practice, feature maps of different persons are concatenated as a batch and inferred in a single feedforward. As we only use the re-projected 2D feature maps, the batch size could be large enough to cover very crowded scenes. In general, the time cost of our method is mainly determined by voxel granularity. By using $2\times$ coarser voxel, its computation complexity could be reduced to $\frac{1}{4}$. The voxel granularity selection serves as a trade-off between speed and accuracy.

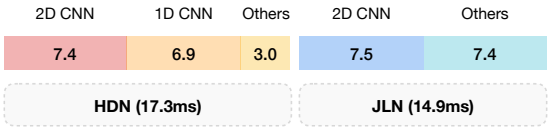

Fig. 6. Time Cost Visualization. We visualize the average inference time cost for each module on the Panoptic test set in milliseconds (ms). It takes 32.2ms in total.

Finally, we analyze the scalability of our method and compare it with the existing methods. Consider applying the algorithms to a challenging scenario that is much larger and more crowded than the current datasets [2,19]. In order to retain a reasonable coverage, the number of cameras needs to grow proportionally [48,53]. VoxelPose [38] uses massive 3D convolution operations that are computation-intensive, and its efficiency disadvantage would be enlarged when scaling. PlaneSweepPose [22] needs to enumerate the depth planes for every pair of camera views and persons. As a result, the computation complexity increases in polynomials regarding the number of cameras and persons. For example, simply shifting from Campus (3 persons, 3 cameras) to Shelf (4 persons, 5 cameras) slows PlaneSweepPose by 2.6× according to [22] (1.3× for our method). MvP [44] uses projective attention to integrate the multi-view information, and its time cost also grows quadratically as camera number increases. As previously analyzed, our method does not involve explicit view-person association, and its speed is mainly affected by the granularity of space division. We argue that the above characteristics make our method more scalable to large, crowded scenes than the previous methods. We deployed our model to a basketball court and a retail store where the space size is 16m×16mm with 12 cameras and 10 people. Our inference time increases by 28.8% compared to that on Panoptic (8m × 8m, 5 cameras, 3.4 person).

4.3 Ablation Study

Table 4. Ablation Study Results. Our full approach is (a). From (b) to (e), we study the effect of volume feature masking, weighted fusion and camera views respectively.

Method	#Views	Mask	Weighted	AP_{25}	AP_{50}	AP_{100}	AP_{150}	MPJPE
(a)	5	✓	✓	85.22	98.08	99.32	99.48	18.26mm
(b)	5		✓	72.05	96.75	99.10	99.39	21.07mm
(c)	5	✓		77.23	97.61	99.18	99.48	20.11mm
(d)	4	✓	✓	73.95	97.02	99.21	99.35	21.12mm
(e)	3	✓	✓	53.68	91.89	97.40	98.30	26.13mm

Table 5. Influence of Voxel Granularity. We additionally report the MACs (Multiply–Accumulate Operations) and number of parameters of the networks.

JLN Voxels	AP_{25}	AP_{50}	AP_{100}	AP_{150}	MPJPE	MACs	Parameters
$64 \times 64 \times 64$	85.22	98.08	99.32	99.48	18.26mm	8.670G	1.236M
$48 \times 48 \times 48$	78.76	97.14	98.99	99.14	19.66mm	4.877G	1.210M
$32 \times 32 \times 32$	73.20	97.37	98.93	99.08	20.47mm	2.167G	1.190M

We train some ablated models to study the impact of the individual factors. All the ablation experiments are evaluated on CMU Panoptic [19], and the results are shown in Tab. 4.

Feature Masking. In (b), we remove the masking step and directly use the local feature volume $\mathbf{V}'$ in JLN. This is equivalent to using a fixed bounding box size as [38]. The degraded performance indicates that the masking mechanism indeed reduces the ambiguity and helps joint localization.

Adaptive Weighted Fusion. In (c), we simply take the mean of the estimated coordinates from different planes to compute the final result. The performance gap suggests that the learned confidence weights emphasize the more reliable estimations as intended.

Number of Cameras. In (d)-(e), we compare the performance under different camera numbers. The accuracy drops with fewer camera views as the feature volume coverage is weakened.

Granularity of Voxels. We study the impact of voxel granularity on both efficiency and accuracy. Tab. 5. shows models trained with different JLN voxel sizes. By reducing the number of voxels (effectively increasing the individual voxel size), the error increases slightly, while the inference efficiency additionally improves. It inspires us to balance the trade-off between speed and accuracy in real usage.

5 Conclusion

In this paper, we present a novel method for 3D human pose estimation from multi-view images. Our pipeline uniquely integrates the feature volume re-projection to both human detection and joint localization, which substitutes the computation-intensive 3D convolutions. Experiment results prove the effectiveness of the proposed HDN and JLN. The accelerated inference demonstrates the potential of our method in real-time applications, especially for large scenes.

Acknowledgement

This work was supported in part by MOST-2018AAA0102004 and NSFC-62061136001.

References

1. Ahuja, K., Ofek, E., Gonzalez-Franco, M., Holz, C., Wilson, A.D.: Coolmoves: User motion accentuation in virtual reality. IMWUT (2021)
2. Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3d pictorial structures for multiple human pose estimation. In: CVPR (2014)
3. Belagiannis, V., Amin, S., Andriluka, M., Schiele, B., Navab, N., Ilic, S.: 3d pictorial structures revisited: Multiple human pose estimation. IEEE transactions on pattern analysis and machine intelligence (2015)
4. Belagiannis, V., Wang, X., Schiele, B., Fua, P., Ilic, S., Navab, N.: Multiple human pose estimation with temporally consistent 3d pictorial structures. In: ECCV (2014)
5. Bridgeman, L., Volino, M., Guillemaut, J.Y., Hilton, A.: Multi-person 3d pose estimation and tracking in sports. In: CVPR Workshops (June 2019)
6. Bultmann, S., Behnke, S.: Real-time multi-view 3d human pose estimation using semantic feedback to smart edge sensors. RSS (2021)
7. Cheng, Y., Yang, B., Wang, B., Yan, W., Tan, R.T.: Occlusion-Aware networks for 3D human pose estimation in video. In: ICCV (2019)
8. Ci, H., Wang, C., Ma, X., Wang, Y.: Optimizing network structure for 3d human pose estimation. In: ICCV (2019)
9. Dong, J., Jiang, W., Huang, Q., Bao, H., Zhou, X.: Fast and robust multi-person 3d pose estimation from multiple views (2019)
10. Dong, J., Shuai, Q., Zhang, Y., Liu, X., Zhou, X., Bao, H.: Motion capture from internet videos. In: ECCV (2020)
11. Ershadi-Nasab, S., Noury, E., Kasaei, S., Sanaei, E.: Multiple human 3d pose estimation from multiview images. Multimedia Tools and Applications (2018)
12. Fabbri, M., Lanzi, F., Calderara, S., Alletto, S., Cucchiara, R.: Compressed volumetric heatmaps for multi-person 3d pose estimation. In: CVPR (2020)
13. Fang, H.S., Xie, S., Tai, Y.W., Lu, C.: RMPE: Regional multi-person pose estimation. In: ICCV (2017)
14. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, USA, 2 edn. (2003)
15. He, Y., Yan, R., Fragkiadaki, K., Yu, S.I.: Epipolar transformers. In: CVPR. pp. 7779–7788 (2020)
16. Huang, C., Jiang, S., Li, Y., Zhang, Z., Traish, J.M., Deng, C., Ferguson, S., Xu, R.Y.D.: End-to-end dynamic matching network for multi-view multi-person 3d pose estimation. In: ECCV (2020)
17. Iskakov, K., Burkov, E., Lempitsky, V., Malkov, Y.: Learnable triangulation of human pose (2019)
18. Jansen, Y., Hornbæk, K.: How relevant are incidental power poses for hci? In: CHI (2018)
19. Joo, H., Liu, H., Tan, L., Gui, L., Nabbe, B., Matthews, I., Kanade, T., Nobuhara, S., Sheikh, Y.: Panoptic studio: A massively multiview system for social motion capture. In: ICCV (2015)
20. Li, C., Lee, G.H.: Generating multiple hypotheses for 3d human pose estimation with mixture density network. In: CVPR (2019)
21. Li, Z., Ye, J., Song, M., Huang, Y., Pan, Z.: Online knowledge distillation for efficient pose estimation. ICCV (2021)
22. Lin, J., Lee, G.H.: Multi-view multi-person 3d pose estimation with plane sweep stereo. In: CVPR (2021)

23. Liu, F., Liu, X.: Voxel-based 3d detection and reconstruction of multiple objects from a single image. In: NeurIPS (2021)
24. Ma, X., Su, J., Wang, C., Ci, H., Wang, Y.: Context modeling in 3d human pose estimation: A unified perspective. In: CVPR (2021)
25. Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.P., Xu, W., Casas, D., Theobalt, C.: Vnect: Real-time 3d human pose estimation with a single rgb camera (2017)
26. Pavlakos, G., Zhou, X., Derpanis, K.G., Daniilidis, K.: Harvesting multiple views for marker-less 3D human pose annotations. In: CVPR (2017)
27. Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3d human pose estimation in video with temporal convolutions and semi-supervised training. In: CVPR (2019)
28. Qiu, H., Wang, C., Wang, J., Wang, N., Zeng, W.: Cross view fusion for 3d human pose estimation. In: ICCV (2019)
29. Ramanathan, V., Huang, J., Abu-El-Haija, S., Gorban, A., Murphy, K., Fei-Fei, L.: Detecting events and key actors in multi-person videos. In: CVPR (2016)
30. Reddy, N.D., Guigues, L., Pischulini, L., Eledath, J., Narasimhan, S.: Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In: CVPR (2021)
31. Reddy, N.D., Guigues, L., Pishchulin, L., Eledath, J., Narasimhan, S.G.: Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15190–15200 (2021)
32. Remelli, E., Han, S., Honari, S., Fua, P., Wang, R.: Lightweight multi-view 3d pose estimation through camera-disentangled representation. In: CVPR (2020)
33. Rukhovich, D., Vorontsova, A., Konushin, A.: Imvoxelnet: Image to voxels projection for monocular and multi-view general-purpose 3d object detection. In: WACV (2022)
34. Sharma, S., Varigonda, P.T., Bindal, P., Sharma, A., Jain, A.: Monocular 3d human pose estimation by generation and ordinal ranking. In: ICCV (2019)
35. Shen, X., Yuan, G., Niu, W., Ma, X., Guan, J., Li, Z., Ren, B., Wang, Y.: Towards fast and accurate multi-person pose estimation on mobile devices. In: IJCAI (2021)
36. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Skeleton-based action recognition with directed graph neural networks. In: CVPR (2019)
37. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR (2019)
38. Tu, H., Wang, C., Zeng, W.: Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In: ECCV (2020)
39. Wang, C., Flynn, J., Wang, Y., Yuille, A.: Recognizing actions in 3d using action-snippets and activated simplices. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
40. Wang, C., Wang, Y., Lin, Z., Yuille, A.L.: Robust 3d human pose estimation from single images or video sequences. IEEE transactions on pattern analysis and machine intelligence **41**(5), 1227–1241 (2018)
41. Wang, C., Wang, Y., Lin, Z., Yuille, A.L., Gao, W.: Robust estimation of 3d human poses from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2361–2368 (2014)
42. Wang, C., Wang, Y., Yuille, A.L.: An approach to pose-based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 915–922 (2013)
43. Wang, C., Wang, Y., Yuille, A.L.: Mining 3d key-pose-motifs for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2639–2647 (2016)

44. Wang, T., Zhang, J., Cai, Y., Yan, S., Feng, J.: Direct multi-view multi-person 3d human pose estimation. Advances in Neural Information Processing Systems (2021)
45. Wang, X., , M.H.A.J., Lee, G.H.: Voxel-based network for shape completion by leveraging edge generation. In: ICCV (2021)
46. Weng, C.Y., Curless, B., Kemelmacher-Shlizerman, I.: Photo wake-up: 3d character animation from a single photo. In: CVPR (2019)
47. Wu, S., et al: Graph-Based 3D Multi-Person pose estimation using Multi-View images. In: ICCV (2021)
48. Xu, J., Zhong, F., Wang, Y.: Learning multi-agent coordination for enhancing target coverage in directional sensor networks. In: Advances in Neural Information Processing Systems (2020)
49. Xu, L., Guan, Y., Jin, S., Liu, W., Qian, C., Luo, P., Ouyang, W., Wang, X.: Vipnas: Efficient video pose estimation via neural architecture search. In: CVPR (2021)
50. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI (2018)
51. Yu, Z., Yoon, J.S., Lee, I.K., Venkatesh, P., Park, J., Yu, J., Park, H.S.: Humbi: A large multiview dataset of human body expressions. In: CVPR (2020)
52. Zhang, F., Zhu, X., Ye, M.: Fast human pose estimation. In: CVPR (2019)
53. Zhang, S., Staudt, E., Faltemier, T., Roy-chowdhury, A.K.: A camera network tracking (camnet) dataset and performance baseline. In: WACV (2015)
54. Zhang, Y., Wang, C., Wang, X., Liu, W., Zeng, W.: Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild. arXiv preprint **arXiv:2108.02452**
55. Zhang, Y., Wang, C., Wang, X., Liu, W., Zeng, W.: Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
56. Zhang, Y., An, L., Yu, T., Li, x., Li, K., Liu, Y.: 4d association graph for realtime multi-person motion capture using multiple video cameras. In: CVPR (2020)
57. Zhang, Z., Wang, C., Qin, W., Zeng, W.: Fusing wearable imus with multi-view images for human pose estimation: A geometric approach. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2200–2209 (2020)
58. Zhang, Z., Wang, C., Qiu, W., Qin, W., Zeng, W.: Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. International Journal of Computer Vision **129**(3), 703–718 (2021)
59. Zheng, C., Zhu, S., Mendieta, M., Yang, T., Chen, C., Ding, Z.: 3d human pose estimation with spatial and temporal transformers. ICCV (2021)
60. Zhong, Y., Zhu, M., Peng, H.: VIN: voxel-based implicit network for joint 3d object detection and segmentation for lidars (2021)
61. Zhou, K., Han, X., Jiang, N., Jia, K., Lu, J.: Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation. In: ICCV (2019)
62. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. In: CVPR (2019)
63. Zhu, L., Rematas, K., Curless, B., Seitz, S., Kemelmacher-Shlizerman, I.: Reconstructing nba players. In: ECCV (2020)