TEMPO: Efficient Multi-View Pose Estimation, Tracking, and Forecasting

Rohan Choudhury   Kris M. Kitani   László A. Jeni
Robotics Institute, Carnegie Mellon University

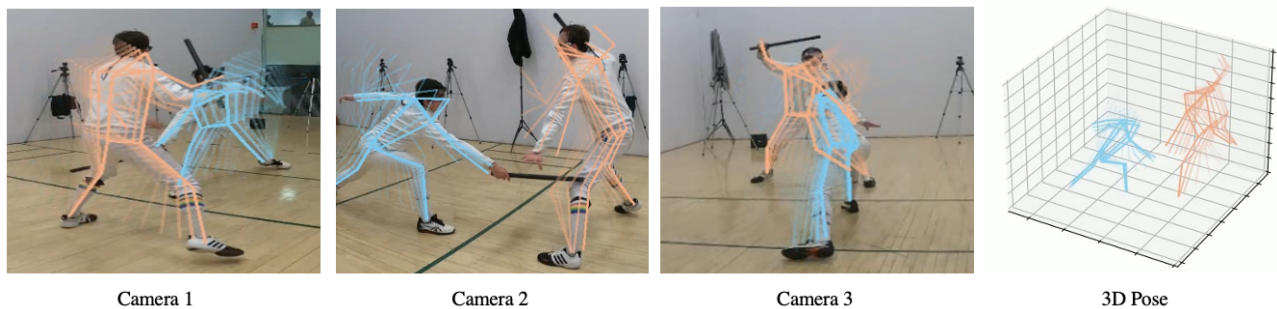{rchoudhu, kmkitani}@andrew.cmu.edu   laszlojeni@cmu.edu

Figure 1: We propose **TEMPO**: TEMporal POse Estimation, a method for multi-view, multi-person pose estimation, tracking and forecasting. TEMPO uses a recurrent architecture to learn a spatiotemporal representation, significantly improving the pose estimation accuracy while preserving speed at inference time. In each view, we show TEMPO's predicted pose skeletons over 10 frames, colored by tracker identity. Lighter colors correspond to previous frames.

Abstract

Existing volumetric methods for predicting 3D human pose estimation are accurate, but computationally expensive and optimized for single time-step prediction. We present TEMPO, an efficient multi-view pose estimation model that learns a robust spatiotemporal representation, improving pose accuracy while also tracking and forecasting human pose. We significantly reduce computation compared to the state-of-the-art by recurrently computing per-person 2D pose features, fusing both spatial and temporal information into a single representation. In doing so, our model is able to use spatiotemporal context to predict more accurate human poses without sacrificing efficiency. We further use this representation to track human poses over time as well as predict future poses. Finally, we demonstrate that our model is able to generalize across datasets without scene-specific fine-tuning. TEMPO achieves 10% better MPJPE with a 33× improvement in FPS compared to TesseTrack on the challenging CMU Panoptic Studio dataset. Our code and demos are available at https://rccchoudhury.github.io/tempo2023/.

1. Introduction

Estimating the pose of several people from multiple overlapping cameras is a crucial vision problem. Volumet-ric multi-view methods, which lift 2D image features from each camera view to a feature volume then regress 3D pose, are currently the state of the art [47, 42, 53, 23] in this task. These approaches produce significantly more accurate poses than geometric alternatives, but suffer from two key limitations. First, the most accurate methods use either 3D convolutions [47, 42, 55] or cross-view transformers [49] which are slow and prevent real-time inference. Secondly, most methods are designed for estimating pose at a single timestep and are unable to reason over time, limiting their accuracy and preventing their use for tasks like motion prediction.

We propose TEMPO, a multi-view TEMporal POse estimation method that addresses both of these issues. TEMPO uses *temporal context* from previous timesteps to produce smoother and more accurate pose estimates. Our model tracks people over time, predicts future pose and runs efficiently, achieving near real-time performance on existing benchmarks. The key insight behind TEMPO, inspired by work in 3D object detection [31, 20], is that recurrently aggregating spatiotemporal context results in powerful learned representations while being computationally efficient. To do this, we decompose the problem into three stages, illustrated in Figure 2. Given an input RGB video from multiple static, calibrated cameras, at a given timestep t we first detect the locations of each person in the scene by unproject-

ing image features from each view to a common 3D volume. We then regress 3D bounding boxes centered on each person, and perform tracking by matching the box centers with the detections from the previous timestep $t - 1$. For each detected person, we compute a spatiotemporal pose representation by recurrently combining features from current and previous timesteps. We then decode the representation into an estimate of the current pose as well as poses at future timesteps. Unlike existing work [47, 53, 42, 55], our method is able to perform temporal tasks like tracking and forecasting without sacrificing efficiency.

We evaluate our method on several pose estimation benchmarks. TEMPO achieves state of the art results on the challenging CMU Panoptic Studio dataset [25] by 10%, and is competitive on the Campus, Shelf and Human3.6M datasets. We additionally collect our own multi-view dataset consisting of highly dynamic scenes, on which TEMPO achieves the best result by a large margin. We show that our model achieves competitive results in pose tracking and evaluate the pose forecasting quality on the CMU Panoptic dataset. Additionally, multi-view pose estimation methods are almost always evaluated on the same dataset they are trained on, leading to results that are specific to certain scenes and camera configurations. We measure our method's ability to generalize across different datasets and find that our method can transfer without additional fine tuning. To summarize, our key contributions are that:

- We develop the most accurate multi-view, multi-person 3D human pose estimation model. Our model uses temporal context to produce smoother and more accurate poses.
- Our model runs efficiently with no performance degradation.
- Our model tracks and forecasts human pose for every person in the scene.
- We evaluate the generalization of our model across multiple datasets and camera configurations.

2. Related Work

3D Pose Estimation, Tracking and Forecasting Approaches for recovering and tracking 3D human pose are usually limited to monocular video. Such methods for pose estimation [46, 5, 27, 29] and pose tracking [41, 40] are highly efficient, but perform significantly worse than multi-view methods in 3D pose estimation accuracy due to the inherent ambiguity of monocular input.

Furthermore, methods in human pose forecasting [34] usually predict future motion from ground truth pose histories. Our approach follows [8, 56] and predicts pose directly from a sequence of video frames. Snipper [56] is the closest to our method and uses a spatiotemporal transformer to

jointly estimate, track and forecast pose from a monocular video. Our method differs in that it is able to produce highly accurate estimates using multi-view information while running efficiently.

Multi-View Pose Estimation Early work in multi-view human pose estimation was limited to the single-person case, with [3, 30, 37, 1] using pictorial structure models to improve over basic triangulation. More recent approaches [19, 39, 23, 7, 26] improve this result by using advanced deep architectures like 3D CNNs and transformers, and others [9, 10] introduce priors on human shape for additional performance. Our method is most similar to [23], which uses 3D CNNs to regress pose directly from a feature volume. We also follow [23, 26] in analyzing our model's transfer to multiple datasets, extending their qualitative, single-person analysis to a quantitative measurement of performance on several multi-person datasets.

In the multi-person setting, early approaches like [3, 12, 13, 52, 6] associate 2D pose estimates from each view, then fuse the matched 2D poses into 3D. Other methods aimed towards multi-person motion capture use Re-ID features [12, 52], 4D graph cuts [54], plane sweep stereo [32], cross-view graph matching [50], or optimize SMPL parameters [14] to produce 3D poses from 2D pose estimates in each view. These methods can generalize across the data sources, but typically have much less accurate predictions compared to *volumetric* methods. These first unproject learned 2D image features into a 3D volume and regress pose directly from the 3D features with neural networks. Both [23] and [47] use computationally expensive 3D CNNs for the pose estimation step. Follow up work includes Faster VoxelPose [53] which replaces these 3D CNNs with 2D CNNs for a large speedup, and TesseTrack [42], which uses 4D CNNs to reason over multiple timesteps. Our method combines the best of both: we efficiently incorporate spatiotemporal information with only 2D CNNs and a lightweight recurrent network.

3D Object Detection and Forecasting Historically, 3D object detection and instance segmentation methods for autonomous driving have led development in using multi-view images. One key similarity to our work is the aggregation of 2D image features into a single 3D volume. While [44, 48] use the same bilinear unprojection strategy as our method, several works [38, 17, 31] propose alternatives such as predicting per-pixel depth. Other works also use temporal information for detection for tracking objects through occlusion and spotting hard-to-see objects; [20, 31, 36] concretely demonstrate the benefits of incorporating spatiotemporal context. In particular, FIERY [20] uses temporal information for future instance prediction and BEVFormer [31] efficiently aggregates temporal information with a recurrent architecture, both of which inspired our method. Furthermore, [24, 16] use per-timestep supervision to track
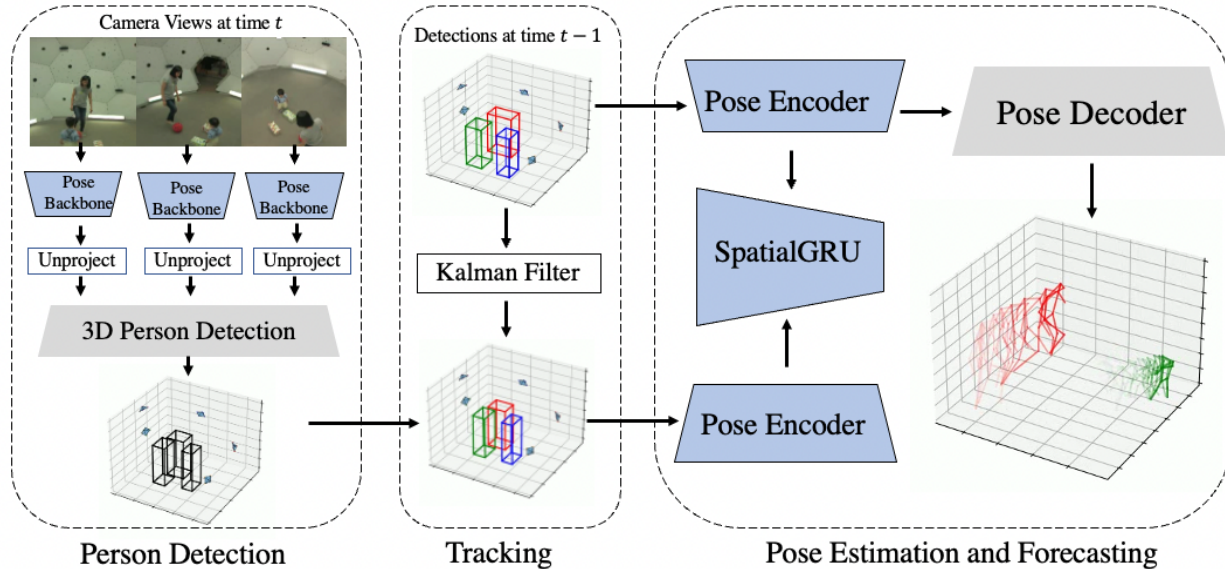
Figure 2: The overall model architecture. We begin by (1) extracting features from each image with the backbone network and unprojecting those features to a 3D volume. In step (2), we use the volume to detect each person in the scene, and (3) associate the detections from the current timestep to the previous one. We then (4) fuse the features from each person with our temporal model and produce a final pose estimate.

pixels through occlusion, an idea which we adapt for reasoning about human pose over multiple frames.

3. Method

Our method assumes access to calibrated time-synchronized videos from one or more cameras. At training time, we assume access to T sets of N RGB images from different cameras, while at inference time, we have a single set of N images corresponding to the current timestep. In order to enable TEMPO to transfer to new camera configurations and settings, we compute the dimensions of the space and size of the voxel volume directly from the camera matrices. We set the height of the volume to a constant $2\,\mathrm{m}$, while setting the length and width of the volume to be the bounding box of the camera extrinsics from a top-down view, and center the space at the mean of the camera locations.

3.1. Preliminaries

We briefly review the person detection and pose estimation modules used by VoxelPose [47], Tessetrack [42], and Faster VoxelPose [53] that TEMPO builds upon. We refer the reader to the original papers for further detail.

3.1.1 Person Detection

The detection module aims to estimate the location of the *root joint* as well as tight 3D bounding box for each person

in the scene. Following previous work, we define the root joint as the mid-hip. At a given time t, the detector module takes as input a set of N images, each corresponding to a different camera view of the same scene at time t. For each image, we extract features with a pretrained backbone, resulting in N feature maps $\mathbf{F}_1^t, \mathbf{F}_2^t, \ldots, \mathbf{F}_N^t$.

Given the camera matrices for each view $\mathbf{C}_1^t, \mathbf{C}_2^t, \ldots, \mathbf{C}_N^t$, we use the bilinear sampling procedure from [23, 47, 17]. For a voxel $v \in V$ with coordinates $\mathbf{x}$, we have

$$v = \sum_{i=1}^{N} \mathbf{F}_i^t(\mathbf{C}_i \mathbf{x}) \tag{1}$$

where $\mathbf{F}_i^t(\mathbf{x})$ is the feature map $\mathbf{F}_i^t$'s value at position $\mathbf{x}$, obtained by bilinear sampling. We then compute a birds-eye view representation of V by taking the maximum along the z-axis:

$$\mathbf{F}_{\mathrm{BEV}}^t = \max_z V \tag{2}$$

We use a 2D CNN to produce a 2D heatmap of $\mathbf{H}^t$ from $\mathbf{F}_{\mathrm{BEV}}^t$ of the (x, y) locations of every root joint in the scene. We then sample the top K locations from $\mathbf{H}^t$, yielding proposals $(x_1, y_1), (x_2, y_2), \ldots (x_K, y_K)$. For each proposal location, we obtain the corresponding feature column $V|_{x,y}$ and apply a 1D CNN to regress a 1D heatmap of the root joint's height, denoted $\mathbf{H}_k^t$. We then sample the maximum z coordinate for each $\mathbf{H}_k^t$, and combine these to produce a set of detections $D_t = \{(x_1, y_1, z_1), \ldots (x_K, y_k, z_K)\}$. Fi-

nally, we regress width, length and centerness from $\mathbf{F}_{\mathrm{BEV}}^t$ with a multi-headed 2D CNN to produce bounding box predictions for each proposal. The loss function for the detection module has three terms. First, L_{2D} is the distance between the predicted 2D heatmap and the ground truth, given by

$$L_{2D} = \sum_{t=1}^{T} \sum_{(x,y)} \| \mathbf{H}^t(x,y) - \mathbf{H}_{GT}^t(x,y) \| \qquad (3)$$

We also compute the loss on the predicted 1D heatmap:

$$L_{1D} = \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{z} \| \mathbf{H}_k^t(z) - \mathbf{H}_{k,GT}^t(z) \| \qquad (4)$$

Finally, we include the bounding box regression loss

$$L_{\mathrm{bbox}} = \sum_{t=1}^{T} \sum_{(i,j) \in U} \| \mathbf{S}(i,j) - \mathbf{S}_{GT}(i,j) \|_1 \qquad (5)$$

The total detection loss is the sum of the above terms, with $L_{det} = L_{2D} + L_{1D} + L_{\mathrm{bbox}}$.

3.1.2 Instantaneous Pose Estimation

For each detection D, we construct a volume of fixed size centered on the detection's center c_i, and unproject the backbone features from each camera view into the volume, as in the detection step. As in [53], we mask out all features falling outside the detection's associated bounding box B_i, resulting in a feature volume V_i^t for person i.

As shown in Figure 3, we project the feature volume V_t^i to 2D along each of the xy, yz, and xz planes, resulting in three 2D feature maps, denoted by $\mathbf{P}_{i,xy}^t$, $\mathbf{P}_{i,xz}^t$, and $\mathbf{P}_{i,yz}^t$. The intuition behind this step is that we can predict the 2D position of each joint in each plane, and fuse the predicted 2D positions back together to form a 3D skeleton. Each feature map is passed through a 2D CNN to decode a heatmap of joint likelihood for every person joint, in each of the three planes, and the 2D joint predictions from each plane are fused into 3D with a learned weighting network. We define both the loss for a predicted pose as the mean squared loss between the computed and ground truth 2D heatmaps, as well as the L_1 loss of the predicted joint locations and ground truth:

$$L_{joint,t}^k = \mathrm{MSE}(\mathbf{P}_{xy}^t, \mathbf{P}_{xy,GT}^t) + \mathrm{MSE}(\mathbf{P}_{xz}^t, \mathbf{P}_{xz,GT}^t) +$$
$$\mathrm{MSE}(\mathbf{P}_{yz}^t, \mathbf{P}_{yz,GT}^t) + \sum_{j=1}^{J} |j_{i,pred} - j_{i,GT}|$$
$$(6)$$

with MSE representing the mean squared error.

3.2. Person Tracking

We now describe how TEMPO uses temporal information. Unlike previous works, TEMPO takes as input a set of N images at time t as well as the person detections D_{t-1} and corresponding 2D pose embeddings P^{t-1} from the previous timestep.

Each proposed detection from the previous step consists of a body center c_i and a bounding box $B_i^t = (h_i^t, w_i^t)$. Given K proposals, we compute a $K \times K$ cost matrix $\mathbf{A}$ based on the IoU between B_i^t and B_k^t for all detections at time t, resulting in

$$\mathbf{A}[i][j] = \| c_i - c_j \| \qquad (7)$$

with c_i, c_j being the associated predicted locations of person i and person j's root joint.

While VoxelTrack [55] computes cost across every joint in each pose, TEMPO uses the top-down view of the associated bounding box for each person. At inference time, we use the SORT [4] tracker, which is fast and uses a simple Kalman filter with no learned Re-ID mechanism. While [42] uses a learned tracker based on SuperGlue [43], we find that SORT is faster and does not result in degraded performance.

3.3. Temporal Pose Estimation and Forecasting

After running the tracker, the input to the pose estimation stage is a set of detection proposals D^t, the previous timestep's detection proposals D^{t-1}, and the assignment matrix between the detection sets $\mathbf{A}^t$. We also assume access to pose features $\mathbf{P}_i^{t-1}$ for each person in the previous timestep. Both D^t and D^{t-1} have K proposals each.

However, $\mathbf{P}_i^{t-1}$ and $\mathbf{P}_i^t$ are centered at c_i^{t-1} and c_i^t respectively, and thus the pose features from each volume are not in the same coordinate system due to the motion of person i. To fix this, we follow the standard procedure used in temporal birds-eye view prediction [31, 20] and warp the projected features from $\mathbf{F}_i^{t-1}$ into the coordinate system of $\mathbf{P}_i^t$ with a translational warp defined by $c_i^t - c_i^{t-1}$.

After warping the previous pose features, we run a recurrent network with Spatial Gated Recurrent Units [2] (SpatialGRUs) to produce multiple embeddings: $\mathbf{F}_i^t$, representing the current pose, and $\mathbf{F}_i^{t+1}, \mathbf{F}_i^{t+2}, \ldots$, representing the pose in future frames. While [20] and [31] do not propagate gradients through time and only predict object locations and instance segments at time T, we *do* backpropagate through time by predicting the pose for each person at *every* timestep. At training time, we recurrently compute the temporal representation at each timestep $t_0, t_0 + 1, \ldots t_0 + T$, decode a pose for every timestep, and compute losses over all the predicted poses simultaneously. Thus, the final train-
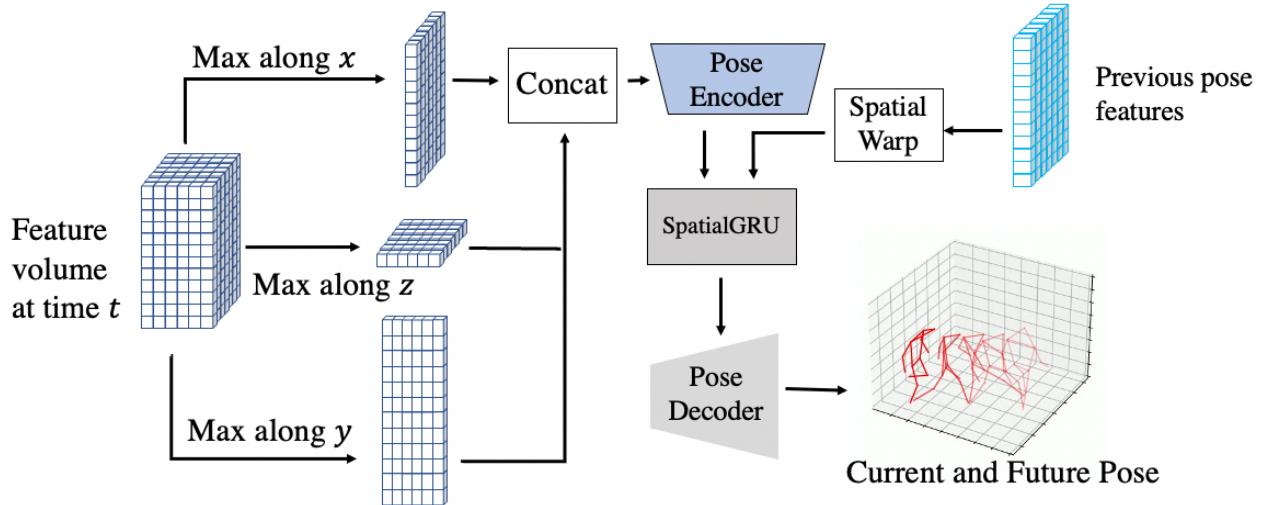
Figure 3: A closer look at the temporal representation used by our model. Following [53], we first project the feature volume to each of the three planes, and concatenate the projections channel-wise. We pass this feature map through an encoder network. We use this feature encoding as input to the SpatialGRU module, using the spatially warped pose feature from the previous timestep as a hidden state. We use the SpatialGRU module to produce features at the current and future timesteps, which we decode into human poses with the pose decoder network.

ing objective is

$$L_{\text{pose}} = \sum_{t=1}^{T} \sum_{i=1}^{K} L_{\text{joint},t}^{i} + L_{\text{joint},t+1}^{i} \qquad (8)$$

where $L_{\text{joint},t}^{i}$ is the L1 distance between the predictde and ground truth pose at time t Providing supervision to the network at every timestep allows the network to learn a representation that encodes the motion between consecutive frames while enabling temporal smoothness between predictions. As we show in Section 4.5, this technique is crucial to our model's performance.

While training, we run the network T times, for each input timestep. However, at inference time, we save the previous embeddings and only receive input images at a single timestep t, significantly reducing the computational burden and allowing our model to use temporal information without sacrificing efficiency.

4. Experiments

4.1. Datasets and Metrics

Panoptic Studio The CMU Panoptic Studio dataset [25] is a large multi-view pose estimation dataset with several synchronized camera sequences of multiple interacting subjects. Following prior work [47, 53, 49, 32], we use five HD cameras, specifically cameras 3, 6, 12, 13, and 23. We also use the same training and test split as these works, omitting the sequence `160906_band3` due to data corruption.

Human 3.6M The Human3.6M dataset [22, 21] consists of videos of a single subject in an indoor studio with four static cameras. Each video has a professional actor performing a specific action. We follow the training-test split of prior works, using subjects 9 and 11 for validation and the others for training, while omitting corrupted sequences.

Campus and Shelf The Campus and Shelf datasets[3] contain approximately 4000 frames of a single scene. While these datasets are commonly used for benchmarking in previous work, they are missing many annotations. We follow previous work [47, 53, 49] and adopt a synthetic heatmap-based scheme for training.

EgoHumans We include the newly collected EgoHumans multi-view dataset [28]. This benchmark consists of approximately 1 hour of video of up to 5 subjects performing highly dynamic activites, such as playing tag, fencing, or group assembly. It contains videos from up to eight fisheye cameras and includes both egocentric and exocentric camera and pose data.

Metrics For the Panoptic Studio, Human3.6M, and Ego-Humans datasets, we report the mean per-joint position error (MPJPE). We additionally report the Average Precision (AP_K) on the Panoptic and EgoHumans datasets and report MPJPE for Human3.6M in line with previous work [37, 35, 23, 42]. On the Shelf and Campus dataset we report the Percentage of Correct Parts (PCP3D). For pose forecasting, we measure the MPJPE between the predicted pose and the ground truth pose 0.33s into the future, matching previous work [56].

Method	Backbone	Resolution	AP_{25} ↑	AP_{50} ↑	AP_{100} ↑	AP_{150} ↑	MPJPE (mm) ↓	FPS (s)↑
VoxelPose[47]	ResNet-50	960×512	83.59	98.33	99.76	99.91	17.68	3.2
Faster VoxelPose [53]	ResNet-50	960×512	85.22	98.08	99.32	99.48	18.26	**31.1**
PlaneSweepPose [32]	ResNet-50	960×512	92.12	98.96	99.81	99.84	16.75	4.3
MvP [32]	ResNet-50	960×512	**92.28**	96.6	97.45	97.69	15.76	3.6
Ours	ResNet-50	960×512	89.01	**99.08**	**99.76**	**99.93**	**14.68**	29.3
VoxelPose[47]	HRNet	384×384	82.44	**98.55**	99.74	99.92	17.63	2.3
Faster VoxelPose [53]	HRNet	384×384	81.69	98.38	99.67	99.83	18.77	22.4
MvP [49]	HRNet	384×384	**90.41**	96.32	97.39	97.89	16.34	2.8
TesseTrack† [42]	HRNet	384×384	86.24	98.29	99.72	99.50	16.92	0.6
Ours	HRNet	384×384	89.32	98.48	**99.73**	**99.94**	**15.99**	20.3

Table 1: Pose estimation results on the CMU Panoptic dataset. Our method achieves the best MPJPE and AP while running at speed comparable to Faster VoxelPose. We evaluate our methods at 384×384 resolution on the Panoptic dataset, as well as the higher resolution used in other methods. We mark TesseTrack [42] with a † as their reported results are on a different data split, and the results in this table are from our best reproduction of the method, which is not public.

	Shelf				Campus			
Method	Actor-1	Actor-2	Actor-3	Average	Actor-1	Actor2	Actor3	Average
Belagiannis et al. [3]	66.1	65.0	83.2	71.4	82.0	72.4	73.7	75.8
Ershadi et al. [15]	93.3	75.9	94.8	88.0	94.2	92.9	84.6	90.6
Dong et al.[12]	98.8	94.1	97.8	96.9	97.6	93.3	98.0	96.3
VoxelPose [47]	99.3	94.1	97.6	97.0	97.6	93.8	98.8	96.7
Faster VoxelPose[53]	99.4	96.0	97.5	97.6	96.5	94.1	97.9	96.2
PlaneSweepPose[32]	99.3	**96.5**	98.0	97.9	98.4	93.7	**99.0**	97.0
MvP [49]	99.3	95.1	97.8	97.4	98.2	94.1	97.4	96.6
TEMPO (Ours)	99.0	96.3	**98.2**	**98.0**	97.7	**95.5**	97.9	**97.3**

Table 2: PCP3D accuracy on the Campus and Shelf datasets. We follow the protocol of previous methods and train our backbone on synthetic heatmaps of ground-truth poses. Our method achieves results comparable to the state-of-the-art.

4.2. Implementation Details

Following [47, 53, 56, 49] we use a ResNet-50 backbone pre-trained on the Panoptic dataset and follow [23] by also using a ResNet-50 backbone pre-trained on Human3.6M for the dataset-specific pose estimation results. We use HRNet [45] pre-trained on COCO as the model backbone for the generalization experiments, as done in [42] rather than pose estimation backbones that are trained on multi-view datasets. All methods are trained on 8 NVIDIA A100 GPUs with batch size of 2 per GPU. We use Adam with a learning rate of 3e-4, with weight decay of 1e-4 and a linear decay schedule for 10 epochs. We measure FPS using a single A100 GPU, and our code is based off the MMPose [11] library. Additional architectural details are in the supplement.

4.3. Pose Estimation Results

We first compare the results of our methods other state-of-the art methods. On the Panoptic Studio dataset, we report results following [53, 47, 49] and use 960×512 images and initialize from a ResNet-50 [18, 51] checkpoint pretrained on the Panoptic dataset. We also evaluate our method using an HRNet [45] backbone pretrained on COCO with 384×384 images for a fair comparison with TesseTrack. In Table 1 we provide a complete comparison across the state-of-the-art methods, with baselines trained on the same image resolutions and backbones for completeness. TEMPO achieves significantly lower MPJPE across both resolutions and backbones, while running at 29.3 FPS, competitive with Faster VoxelPose. We attribute this performance increase to the the smoother and more temporally consistent skeletons our model produces due to its incorporation of temporal context and temporal supervision. We also show in 2 that our model achieves performance competitive with the state of the art on the Campus and Shelf datasets.

Method	MOTA	IDF1	MPJPE
VoxelTrack [55]	98.45	98.67	-
Snipper [56]	93.40	85.50	40.2
TEMPO (Ours)	98.42	93.62	**38.5**

(a) Evaluation of tracking and forecasting on the CMU Panoptic dataset. Our method outperforms [56] by using multi-view information and is competitive with VoxelTrack. [55]

Training Dataset			MPJPE (mm)		
Panoptic	H.6M	EgoHumans	Panoptic	H.6M	EgoHumans
✓			14.68	62.96	119.8
✓	✓		36.56	25.3	108.3
✓	✓	✓	42.8	32.3	48.9

(b) Evaluating TEMPO's ability to transfer. Unlike previous methods, TEMPO is able to train on multiple datasets and can perform reasonably given enough training data. While our model is able to effectively transfer to Human3.6M, it has difficulty with EgoHumans due likely due to its use of fisheye cameras.
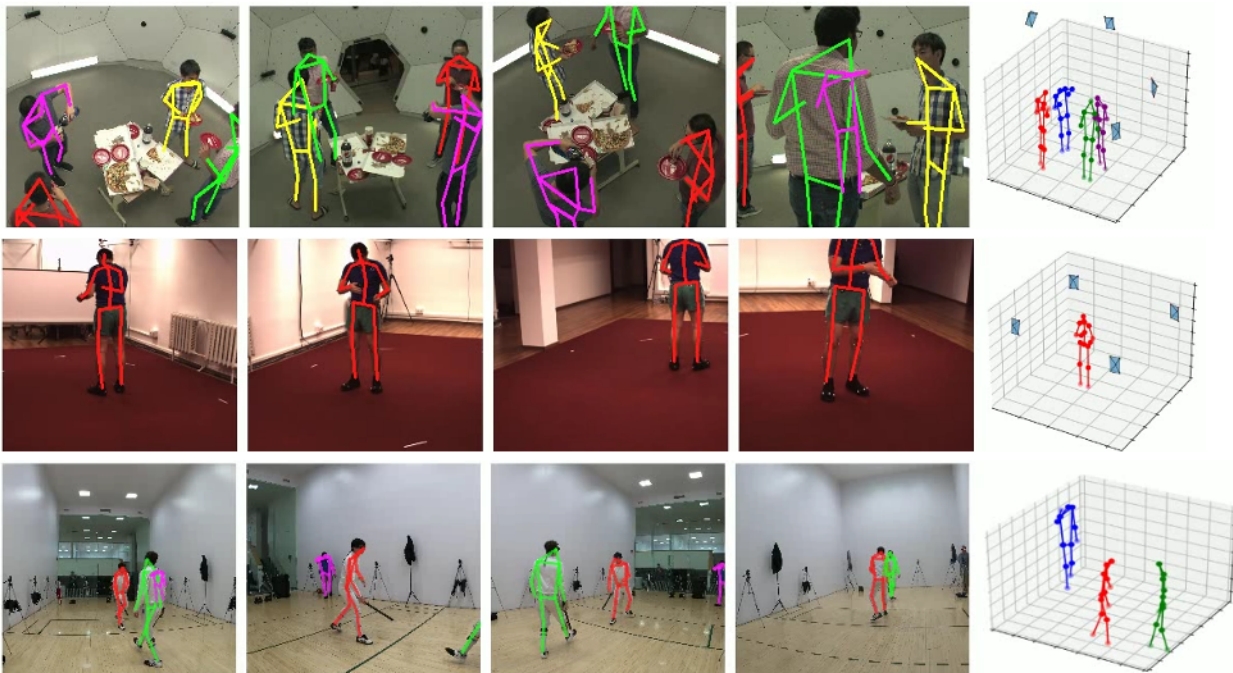

Figure 4: Samples of our model's pose estimation performance on the Panoptic Studio, Human3.6M, and EgoHumans datasets. TEMPO predicts accurate poses and tracks them over time.

We compare pose estimation performance on the Human3.6M and EgoHumans datasets in Table 4 . Our results perform comparably to the state-of-the art. Notably [23] uses ground-truth locations and cropped bounding boxes from input views, while our method is able to match performance despite simultaneously detecting people in the scene. Furthermore, our method significantly outperforms others in the more challenging EgoHumans benchmark, suggesting that temporal context is crucial for handling rapid motion.

4.4. Pose Tracking and Forecasting

We compared the performance of TEMPO's tracking to VoxelTrack [55] in Table 3a. Our tracker is competitive but performs slightly worse, which is expected due to its lack of learned Re-ID features. To our knowledge, pose forecasting has not been attempted for the multi-view case, so we compare against the closest forecasting method, Snipper [56], which estimates future pose from monocular video. Our method takes as input 4 previous timesteps and predicts the pose over the next 3 timesteps, which is 0.33 seconds into the future, matching prior work [56]. Our model achieves state-of-the-art performance, shown in Table 3a. We conducted an additional experiment to measure the performance of our model in transfer on different datasets. The standard practice in monocular 3D pose estimation is to train on and evaluate performance on a combination of multiple datasets. However, the predominant paradigm for

Method	Human 3.6M	EgoHumans
Martinez et al. [35]	57.0	-
Pavlakos et al. [37]	56.9	-
Kadkhodamohammadi et al. [26]	49.1	-
Ma et al [33]	24.4	-
VoxelPose [47]	19.0	46.24
Faster VoxelPose [53]	19.8	48.50
MvP [49]	18.6	41.72
Iskakov et al. [23]	**17.7**	-
TEMPO (Ours)	18.5	**36.74**

Table 4: Pose Estimation results on the Human3.6M and EgoHumans datasets in MPJPE (mm). Our method is competitive woth the state-of-the art on Human3.6M and surpasses current methods by a significant margin on EgoHumans.

Method	T	Forecasting	Warping	Per-t loss	MPJPE ↓(mm)
(a)	3				17.83
(b)	3			✓	15.03
(c)	3		✓	✓	14.94
(d)	3	✓	✓	✓	14.68
(e)	4	✓	✓	✓	14.90
(f)	5	✓	✓	✓	14.82

Table 5: Ablation study on various components to our model. The most important component to performance was the per-timestep supervision. Warping the previous feature also improved performance. We observed the forecasting and slightly increasing the length of the input history had no noticeable effect on performance.

Cameras	1	2	3	4	5
MPJPE	51.32	32.13	19.22	17.34	14.68

Table 6: Ablation on number of cameras. We observe that performance decreases with decreasing number of cameras.

evaluating multi-view pose estimation methods has been to train a single model only on a single dataset and evaluate it on the same dataset.This severely limits the potential of these models to generalize across different settings and thus be deployed in real-world scenarios. Similar to [23], we evaluate the performance of our model trained on multiple combinations of annotated datasets, and report the results for each combination in Table 3b. We run our model with no fine tuning and the same voxel size of 10cm^3 across each dataset. Our method is able to transfer and successfully tracks and predicts pose, but performs noticeably worse, likely due to being trained on a single camera configuration. In particular, the CMU Panoptic training dataset uses 5 cameras, whereas Human3.6M uses 4 and EgoHumans uses 8 fisheye cameras. The model has the most trouble generalizing to EgoHumans, likely due to the significantly larger indoor space and different camera models. We find that TEMPO performs better in transfer after including training data from each dataset, especially on the EgoHumans dataset, suggesting that future works should include diverse multi-view data from different camera configurations and camera models in order to better generalize.

4.5. Ablations

We train ablated models to study the impact of individual components in our method. All our experiments are conducted on the CMU Panoptic dataset, with results shown in table 5. We find that using temporal information slightly helps, but with per-timestep supervision, the model is able to greatly improve. Warping the pose between timesteps further improves the model. We hypothesize that the im-

provement from warping is small due to the relative lack of motion in the Panoptic dataset - the distance between body centers in consecutive timesteps is usually small. We also measure the effect of the history length on performance and found no significant difference. While intuitively, larger history lengths should provide more context, the GPU memory constraints of our method prevent investigating $T > 5$. We also ablated on the number of cameras on the Panoptic dataset. We found that the MPJPE increases with the number of cameras, matching the findings of [47, 23] and [42].

5. Conclusions

Understanding human behavior from video is a fundamentally temporal problem, requiring accurate and efficient pose estimation algorithms that can reason over time. We presented the first method that satisfies these requirements, achieving state-of-the-art results over existing multi-view pose estimation benchmarks via temporal consistency as a learning objective. Our model is also highly efficient, relying on recurrence to maintain a temporal state while enabling pose tracking and forecasting. TEMPO represents a step closer towards general-purpose human behavior understanding from video.

Acknowledgements

References

[1] Sikandar Amin, Mykhaylo Andriluka, Marcus Rohrbach, and Bernt Schiele. Multi-view pictorial structures for 3d human pose estimation. In *Bmvc*, volume 1. Bristol, UK, 2013.

[2] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015.

[3] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures revisited: Multiple human pose estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(10):1929–1942, 2016.

[4] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016.

[5] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 561–578. Springer, 2016.

[6] Lewis Bridgeman, Marco Volino, Jean-Yves Guillemaut, and Adrian Hilton. Multi-person 3d pose estimation and tracking in sports. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019.

[7] 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.

[8] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 387–404. Springer, 2020.

[9] Zhuo Chen, Xu Zhao, and Xiaoyue Wan. Structural triangulation: A closed-form solution to constrained 3d human pose estimation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 695–711. Springer, 2022.

[10] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2262–2271, 2019.

[11] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. `https://github.com/open-mmlab/mmpose`, 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[13] 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.

[14] Zijian Dong, Jie Song, Xu Chen, Chen Guo, and Otmar Hilliges. Shape-aware multi-person pose estimation from multi-view images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11158–11168, 2021.

[15] Sara Ershadi-Nasab, Erfan Noury, Shohreh Kasaei, and Esmaeil Sanaei. Multiple human 3d pose estimation from multiview images. *Multimedia Tools and Applications*, 77:15573–15601, 2018.

[16] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 59–75. Springer, 2022.

[17] Adam W. Harley, Zhaoyuan Fang, Jie Li, Rares Ambrus, and Katerina Fragkiadaki. Simple-BEV: What really matters for multi-sensor bev perception? In *arXiv:2206.07959*, 2022.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[19] Yihui He, Rui Yan, Katerina Fragkiadaki, and Shoou-I Yu. Epipolar transformers. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 7779–7788, 2020.

[20] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance prediction in bird's-eye view from surround monocular cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15273–15282, October 2021.

[21] Catalin Ionescu, Fuxin Li, and Cristian Sminchisescu. Latent structured models for human pose estimation. In *2011 International Conference on Computer Vision*, pages 2220–2227. IEEE, 2011.

[22] 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, jul 2014.

[23] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *International Conference on Computer Vision (ICCV)*, 2019.

[24] Joel Janai, Fatma Guney, Anurag Ranjan, Michael Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *Proceedings of the European conference on computer vision (ECCV)*, pages 690–706, 2018.

[25] 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 *ICCV*, 2015.

[26] Abdolrahim Kadkhodamohammadi and Nicolas Padoy. A generalizable approach for multi-view 3d human pose regression. *Machine Vision and Applications*, 32(1):6, 2021.

[27] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. corr abs/1712.06584 (2017). *arXiv preprint arXiv:1712.06584*, 1(3), 2017.

[28] Rawal Khirodkar, Aayush Bansal, Lingni Ma, Richard Newcombe, Minh Vo, and Kris Kitani. Egohumans: An egocentric 3d multi-human benchmark. *arXiv preprint arXiv:2305.16487*, 2023.

[29] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5253–5263, 2020.

[30] Ilya Kostrikov and Juergen Gall. Depth sweep regression forests for estimating 3d human pose from images. In *BMVC*, volume 1, page 5. Nottingham, UK, 2014.

[31] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022.

[32] Jiahao Lin and Gim Hee Lee. Multi-view multi-person 3d pose estimation with plane sweep stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11886–11895, June 2021.

[33] Haoyu Ma, Zhe Wang, Yifei Chen, Deying Kong, Liangjian Chen, Xingwei Liu, Xiangyi Yan, Hao Tang, and Xiaohui Xie. Ppt: token-pruned pose transformer for monocular and multi-view human pose estimation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 424–442. Springer, 2022.

[34] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9489–9497, 2019.

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

[36] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection, 2022.

[37] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Harvesting multiple views for marker-less 3d human pose annotations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6988–6997, 2017.

[38] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Proceedings of the European Conference on Computer Vision*, 2020.

[39] 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.

[40] Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. Tracking people with 3d representations. *arXiv preprint arXiv:2111.07868*, 2021.

[41] Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. Tracking people by predicting 3d appearance, location and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2740–2749, 2022.

[42] N Dinesh Reddy, Laurent Guigues, Leonid Pischulin, Jayan Eledath, and Srinivasa G. Narasimhan. Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[43] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020.

[44] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2446, 2019.

[45] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[46] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11179–11188, 2021.

[47] Hanyue Tu, Chunyu Wang, and Wenjun Zeng. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *European Conference on Computer Vision (ECCV)*, 2020.

[48] Hsiao-Yu Fish Tung, Ricson Cheng, and Katerina Fragkiadaki. Learning spatial common sense with geometry-aware recurrent networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2595–2603, 2019.

[49] Tao Wang, Jianfeng Zhang, Yujun Cai, Shuicheng Yan, and Jiashi Feng. Direct multi-view multi-person 3d human pose estimation. *Advances in Neural Information Processing Systems*, 2021.

[50] Size Wu, Sheng Jin, Wentao Liu, Lei Bai, Chen Qian, Dong Liu, and Wanli Ouyang. Graph-based 3d multi-person pose estimation using multi-view images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11148–11157, 2021.

[51] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision (ECCV)*, 2018.

[52] Yan Xu and Kris Kitani. Multi-view multi-person 3d pose estimation with uncalibrated camera networks. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022.

[53] Hang Ye, Wentao Zhu, Chunyu Wang, Rujie Wu, and Yizhou Wang. Faster voxelpose: Real-time 3d human pose estima-

tion by orthographic projection. In *European Conference on Computer Vision (ECCV)*, 2022.

[54] Yuxiang Zhang, Liang An, Tao Yu, Xiu Li, Kun Li, and Yebin Liu. 4d association graph for realtime multi-person motion capture using multiple video cameras. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1324–1333, 2020.

[55] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenyu Liu, and Wenjun Zeng. Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2613–2626, 2022.

[56] Shihao Zou, Yuanlu Xu, Chao Li, Lingni Ma, Li Cheng, and Minh Vo. Snipper: A spatiotemporal transformer for simultaneous multi-person 3d pose estimation tracking and forecasting on a video snippet, 2022.