End-to-End Multi-Person Pose Estimation with Pose-Aware Video Transformer

Yonghui Yu[1*], **Jiahang Cai**[1*], **Xun Wang**[1], **Wenwu Yang**[1†]

[1]Zhejiang Gongshang University, China

Abstract

Existing multi-person video pose estimation methods typically adopt a two-stage pipeline: detecting individuals in each frame, followed by temporal modeling for single-person pose estimation. This design relies on heuristic operations such as detection, RoI cropping, and non-maximum suppression (NMS), limiting both accuracy and efficiency. In this paper, we present a fully end-to-end framework for multi-person 2D pose estimation in videos, effectively eliminating heuristic operations. A key challenge is to associate individuals across frames under complex and overlapping temporal trajectories. To address this, we introduce a novel Pose-Aware Video transformEr Network (PAVE-Net), which features a spatial encoder to model intra-frame relations and a spatiotemporal pose decoder to capture global dependencies across frames. To achieve accurate temporal association, we propose a pose-aware attention mechanism that enables each pose query to selectively aggregate features corresponding to the same individual across consecutive frames. Additionally, we explicitly model spatiotemporal dependencies among pose keypoints to improve accuracy. Notably, our approach is the first end-to-end method for multi-frame 2D human pose estimation. Extensive experiments show that PAVE-Net substantially outperforms prior image-based end-to-end methods, achieving a **6.0** mAP improvement on PoseTrack2017, and delivers accuracy competitive with state-of-the-art two-stage video-based approaches, while offering significant gains in efficiency.

Code — https://github.com/zgspose/PAVENet

Introduction

Multi-person 2D pose estimation (MPPE) aims to detect and localize anatomical keypoints of all individuals in images or videos, and is fundamental to many applications such as human-computer interaction (Li et al. 2020), behavior analysis (Sun et al. 2023), and motion capture (Zhang et al. 2020). Although MPPE is widely used in videos, most existing methods process frames independently as static images (Geng et al. 2023; Wang and Zhang 2022; Xu et al. 2022; Sun et al. 2019; Xiao, Wu, and Wei 2018; Li et al. 2021; Shi et al. 2022; Wang, Xuan, and Zhang 2024; Khirodkar et al. 2024). Recent works (Bertasius et al. 2019; Liu et al. 2021a; Jin, Lee, and Lee 2022; Feng et al. 2023b; He and Yang 2024) incorporate temporal information to better handle occlusion, motion blur, and defocus, showing the importance of leveraging video dynamics.

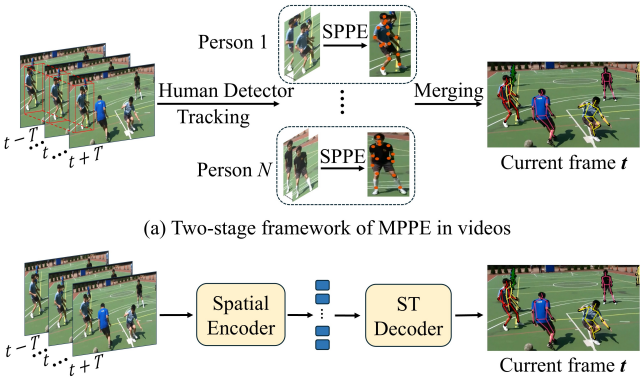

Figure 1: Comparison of two-stage and end-to-end frameworks for video-based MMPE. (a) To predict 2D poses in the current frame, existing methods (Bertasius et al. 2019; Liu et al. 2021a; Jin, Lee, and Lee 2022; Feng et al. 2023b,a; He and Yang 2024) first crop regions from consecutive frames for each human instance and then input them into a temporal model to perform single-person pose estimation (SPPE). (b) PAVE-Net achieves end-to-end video-based MMPE with a spatial encoder and spatiotemporal decoder.

Existing video-based methods all follow a detection-based pipeline, as shown in Fig. 1(a), where human instances are first detected and their regions cropped from consecutive frames before being fed into a CNN-based (Bertasius et al. 2019; Liu et al. 2021a; Feng et al. 2023b,a) or Transformer-based (Jin, Lee, and Lee 2022; He and Yang 2024) temporal model for single-person pose estimation. This framework has several drawbacks: (1) it fails to capture spatial relations among human instances, (2) its accuracy heavily depends on human detector performance (Wang and Zhang 2022; Shi et al. 2022), leading to suboptimal results in crowded scenarios, and (3) it is computationally expensive due to the separate detector, with runtime increasing as the number of people grows. As a result, these methods split the task into

*These authors contributed equally.

†Corresponding Author (wwyang@zjgsu.edu.cn).

two stages, preventing full end-to-end optimization.

To overcome the limitations of two-stage pipelines, several fully end-to-end frameworks have recently been proposed for image-based MPPE (Shi et al. 2022; Yang et al. 2023; Liu et al. 2023). These methods typically adopt an encoder-decoder transformer architecture: the encoder captures local dependencies among image feature tokens, while the decoder uses pose queries to directly infer full-body poses. A straightforward extension to video-based MPPE, as explored in (Qiu et al. 2023) for 3D pose estimation, incorporates temporal information via a spatiotemporal encoder, modeling global dependencies among feature tokens across frames. However, this extension dramatically increases computational complexity, as transformer attention scales quadratically with token count; for example, processing 5 frames increases the computational load by 25×, leading to prohibitive memory and compute costs. Furthermore, as illustrated in Fig. 1, individuals in multi-person video scenes follow independent and potentially overlapping temporal trajectories, posing a significant challenge: accurately associating identities over time is critical to prevent feature mixing and to enable effective temporal aggregation.

To effectively address these challenges, we propose PAVE-Net, a fully end-to-end framework specifically designed for multi-person 2D pose estimation in videos. PAVE-Net integrates spatial-temporal dependency modeling at both the instance level (individuals) and the joint level (fine-grained body keypoints). As shown in Fig. 1(b), PAVE-Net takes as input the current frame along with several adjacent frames and outputs 2D poses for all individuals in the current frame. Specifically, it first encodes local dependencies among visual feature tokens from each frame. Next, a spatiotemporal pose decoder captures global dependencies between pose queries and tokens across frames. To ensure each pose query accurately aggregates features corresponding to the same individual over time, we introduce a novel pose-aware attention mechanism that predicts initial pose estimates and uses them to guide query-to-feature matching across frames. Finally, a spatiotemporal joint decoder explicitly analyzes dependencies among keypoints within each pose, further refining the multi-person pose estimates.

To the best of our knowledge, this is the first fully end-to-end framework for multi-person 2D pose estimation in videos. We extensively evaluate our method on three widely used video-based MPPE benchmarks: PoseTrack2017 (Iqbal, Milan, and Gall 2017), PoseTrack2018 (Andriluka et al. 2018), and PoseTrack21 (Doering et al. 2022). Experimental results show that our framework achieves substantial improvements, outperforming prior end-to-end image-based methods by a notable margin (*e.g.*, **6.0** mAP gain), while delivering accuracy comparable to state-of-the-art two-stage methods. Moreover, unlike two-stage approaches, our method eliminates the human detection step, removing heuristic operations such as RoI cropping and NMS, offering significant efficiency gains and full end-to-end differentiability. It should be noted that our work does not address temporal pose tracking, but rather presents an end-to-end framework with superior performance for 2D pose estimation in videos.

Our main contributions can be summarized as follows:

- We propose **PAVE-Net**, a novel end-to-end framework for multi-person video pose estimation that efficiently and flexibly models spatiotemporal relationships among both human instances and fine-grained body joints.
- Our method is the first end-to-end approach for multi-frame, multi-person 2D pose estimation. Unlike existing two-stage methods, it directly predicts instance-aware full-body poses, eliminating the need for human detection, RoI cropping, and NMS.
- Extensive experiments demonstrate that our method significantly outperforms prior end-to-end image-based MPPE approaches, while achieving accuracy comparable to state-of-the-art two-stage video-based methods and offering substantial efficiency gains.

Method

Given a current video frame $F(t)$ at time t containing multiple individuals, our goal is to estimate the locations of pose joints for each person by leveraging temporal dynamics from a sequence of consecutive frames $\langle F(t-T), \cdots, F(t), \cdots, F(t+T) \rangle$, where T denotes a predefined temporal span. Our method adopts the encoder-decoder transformer architecture commonly used in end-to-end image-based MPPE (Shi et al. 2022; Yang et al. 2023; Liu et al. 2023): feature tokens are extracted from video frames by the encoder, followed by a pose decoder that learns multiple pose queries to directly predict full-body poses. A joint decoder is then applied to further refine predictions at the joint level. In contrast to prior end-to-end MPPE models designed for static images (Shi et al. 2022; Yang et al. 2023; Liu et al. 2023), our approach introduces a video-based end-to-end framework that effectively and efficiently exploits temporal information.

Video Transformer Baseline

As a baseline application of the encoder-decoder transformer architecture in end-to-end video-based MPPE, as adopted in (Qiu et al. 2023) for 3D pose estimation, we employ a spatiotemporal encoder to capture global dependencies among visual feature tokens extracted from all input video frames, as illustrated in Fig. 2. For the input frame sequence $X \in \mathbb{R}^{f \times H \times W \times 3}$, f denotes the number of frames, H and W the height and width of each frame, and 3 the number of color channels. Each frame is represented as $F(t') \in \mathbb{R}^{H \times W \times 3}$, where $t' \in [t-T, t+T]$. For each $F(t')$, multi-scale features $Z(t') \in \mathbb{R}^{\frac{H}{s} \times \frac{W}{s} \times C_s}$ are extracted by a backbone network (*e.g.*, ResNet (He et al. 2016)), where s is the scale factor and C_s the feature dimensionality at that scale. $Z(t')$ is then converted into tokens $\tau(t') \in \mathbb{R}^{N \times D}$ via patch embedding with a patch size of 1×1, where N is the number of tokens and D the embedding dimension. After concatenating tokens from all frames, the sequence $X \in \mathbb{R}^{f \times H \times W \times 3}$ is transformed into $Y \in \mathbb{R}^{f \times N \times D}$, which is fed into the spatiotemporal transformer encoder.

Spatiotemporal Transformer Encoder captures global spatiotemporal dependencies among feature tokens,

$$\tilde{\tau}(t) = \text{STE}(Y), \tag{1}$$

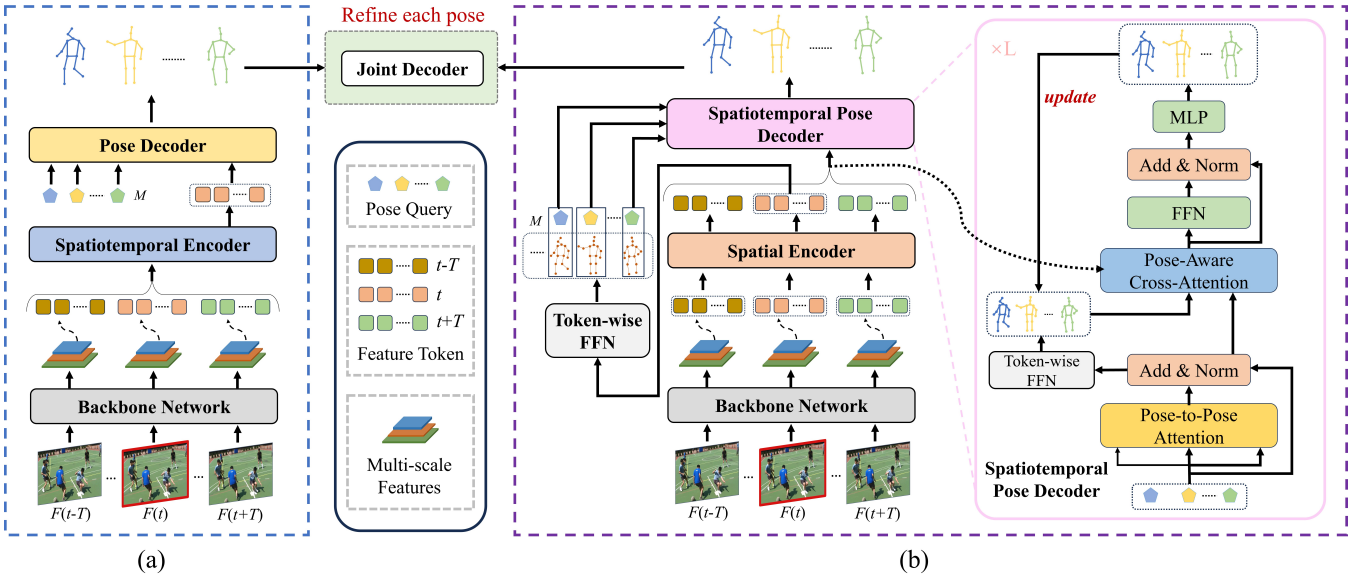

Figure 2: (a) Video transformer baseline. (b) Pose-aware video transformer (PAVE-Net) architecture. The goal is to detect all human poses in the current frame $F(t)$ by leveraging temporal dynamics from a sequence of consecutive frames $\langle F(t-T), \ldots, F(t), \ldots, F(t+T)\rangle$. PAVE-Net employs a backbone network to extract multi-scale features from each frame, which are transformed into feature tokens. A Spatial Encoder (SE) processes each frame independently to capture local dependencies within its tokens. The Spatiotemporal Pose Decoder (STPD) then models global dependencies between pose queries and feature tokens across all frames, using the top M highest-confidence poses regressed from the feature tokens of the current frame t as references. This enables accurate prediction of 2D poses for frame t, which are further refined by a joint decoder.

where $\text{STE}(\cdot)$ denotes the spatiotemporal encoder module, and $\tilde{\tau}(t) \in \mathbb{R}^{N\times D}$ represents the updated feature tokens for the current video frame t, encoding both spatial and temporal information learned by the STE. In our implementation, we adopt deformable attention (Zhu et al. 2020) for an efficient realization of the STE module, rather than standard multi-head self-attention (Vaswani et al. 2017). We stack six identical deformable attention layers, each consisting of a multi-scale deformable attention module and a simple token-wise feed-forward network. Feature tokens pass through these layers sequentially, with each producing an updated version that serves as input for the next. Additionally, each initial feature token is augmented with a learnable position embedding and a feature scale-level embedding, and their sum forms the encoder input.

Pose Decoder computes cross-attention between M learnable pose query tokens $Q^p \in \mathbb{R}^{M\times D}$ and the feature tokens of the current video frame $\tilde{\tau}(t) \in \mathbb{R}^{N\times D}$, which encode both spatial and temporal information,

$$\tilde{Q}^p = \text{PD}(Q^p, \tilde{\tau}(t)), \tag{2}$$

where $\text{PD}(\cdot)$ denotes the pose decoder module, and $\tilde{Q}^p$ are the updated query tokens, each extracting features from $\tilde{\tau}(t)$. Following the STE module, we adopt deformable attention to build the PD module for efficiency. As in (Shi et al. 2022), we stack three identical deformable cross-attention layers. In each layer, query tokens first interact through a self-attention module (*i.e.*, pose-to-pose attention), followed by deformable cross-attention to extract features from $\tilde{\tau}(t)$ (*i.e.*, feature-to-pose attention). To predict full-body poses at frame t, the decoder output $\tilde{Q}^p$ is passed through two token-wise feed-forward networks: one predicts M full-body poses and the other predicts confidence scores for each pose,

$$\begin{cases} \tilde{Q}^p \xrightarrow[\text{feed-forward network}]{\text{token-wise fully connected}} \{P_i(t)\}_{i=1}^{M} \\ \tilde{Q}^p \xrightarrow[\text{feed-forward network}]{\text{token-wise fully connected}} \{c_i(t)\}_{i=1}^{M} \end{cases}, \tag{3}$$

where $P_i(t) \in \mathbb{R}^{J\times 2}$ denotes the joint coordinates of the i-th 2D pose, with J representing the number of joints (*e.g.*, $J = 15$ for the PoseTrack datasets (Iqbal, Milan, and Gall 2017)), and $c_i(t)$ is the confidence score for the i-th pose. A joint decoder, discussed in the next section, is further used to refine predicted poses at the joint level.

Pose-Aware Video Transformer

We observe that the video transformer baseline primarily relies on the spatiotemporal encoder $\text{STE}(\cdot)$ to capture global dependencies across frames and leverage temporal information. However, compared to the encoder used in end-to-end image-based MPPE methods (Shi et al. 2022; Yang et al. 2023; Liu et al. 2023), which only captures local dependencies within a single image, the computational complexity of the spatiotemporal encoder increases substantially. For example, when T = 2 (*i.e.*, using 5 frames), its computational demand becomes 25 times higher than that of a single-frame encoder. In addition, multi-person video scenarios introduce a key technical challenge: temporal features from different individuals can easily become entangled without explicit cross-frame associations. Therefore, accurately associating identities across frames is critical for effectively aggregating temporal features corresponding to each individual.

To effectively and efficiently exploit temporal information in videos, we propose a novel Pose-Aware Video transformEr network, called PAVE-Net. As illustrated in Fig. 2, PAVE-Net consists of three main modules: the spatial encoder (SE), the spatiotemporal pose decoder (STPD), and the spatiotemporal joint decoder (STJD).

Spatial Encoder. Unlike the STE$(\cdot)$ module used in the baseline, which captures global dependencies among feature tokens across multiple frames, the spatial encoder (SE) processes each frame independently to capture local dependencies within its feature tokens,

$$\hat{\tau}(t') = \text{SE}(\tau(t')), \quad t' \in [t-T, t+T], \tag{4}$$

where SE$(\cdot)$ denotes the spatial encoder module, and $\hat{\tau}(t') \in \mathbb{R}^{N\times D}$ represents the updated feature tokens for frame t', encoding spatial information learned by SE. Since SE captures only local dependencies within each frame and operates independently across frames, its output can be reused for different frame sequences. For example, $\hat{\tau}(t)$ can be reused for pose estimation from frame $t-T$ to $t+T$. As a result, the computational complexity of SE is equivalent to that of a single-frame encoder. In our implementation, SE adopts the same architecture as the spatiotemporal encoder (STE) in Eq. 1, consisting of multi-scale deformable attention and feed-forward network (FFN) blocks.

Spatiotemporal Pose Decoder with Pose-Aware Attention. Since each set of feature tokens $\hat{\tau}(t')$, where $t' \in [t-T, t+T]$, encodes only local dependencies within frame t', the spatiotemporal pose decoder (STPD) must compute cross-attention between the M learnable pose query tokens $Q^p \in \mathbb{R}^{M\times D}$ and the feature tokens from all input frames to aggregate temporal features,

$$\hat{Q}^p = \text{STPD}(Q^p, \{\hat{\tau}(t')\}_{t'=t-T}^{t+T}), \tag{5}$$

where STPD$(\cdot)$ denotes the spatiotemporal pose decoder, and $\hat{Q}^p$ represents the updated query tokens, each aggregating features corresponding to the same individual across frames. STPD adopts a similar architecture to the pose decoder (PD) in Eq. 2, consisting of three stacked layers, each with self-attention among query tokens and deformable cross-attention between query tokens and multi-scale feature tokens. To ensure that each pose query token consistently aggregates features from the same individual across frames, we introduce a pose-aware attention mechanism, which guides each query to attend only to feature tokens associated with the same person throughout the temporal window.

We first predict a set of initial poses from the feature tokens $\hat{\tau}(t)$ of the current frame t, where each feature token is used to regress a full-body pose and its confidence score via two token-wise regression heads, similar to Eq. 3. The M poses with the highest confidence scores are then selected as reference poses, with each pose query token assigned a corresponding reference. Note that, unlike (Shi et al. 2022), our method does not assign query tokens to a fixed set of randomly initialized reference points.

For each pose query token $\mathbf{q}_i^p \in \mathbb{R}^D$, $i = 1, 2, \ldots, M$, we denote $P_i^0(t) \in \mathbb{R}^{J\times 2}$ as its initial reference pose. Since an individual's pose remains similar across adjacent frames, we reuse $P_i^0(t)$ as the reference for that individual across all input frames, enabling us to locate corresponding features in each frame. A token-wise FFN regresses relative offsets for each input frame with respect to the reference pose,

$$\mathbf{q}_i^p \xrightarrow[\text{FFN}]{\text{token-wise}} \{\Delta P_i(t')\}_{t'=t-T}^{t+T}, \tag{6}$$

where $\Delta P_i(t')$ is the relative offset at frame t'. Thus, for each pose query token $\mathbf{q}_i^p$, the target positions at frame t' are given by $P_i^0(t') + \Delta P_i(t')$, where $P_i^0(t') = P_i^0(t)$ for all t'. The query token then extracts relevant features from $\{\hat{\tau}(t')\}_{t'=t-T}^{t+T}$ at these positions, performing cross-attention to aggregate features corresponding to the same individual across frames. Inspired by (Zhu et al. 2020; Shi et al. 2022), we progressively refine reference poses across decoder layers: $P_i^l(t') = P_i^{l-1}(t') + \Delta P_i^l(t')$, where $\Delta P_i^l(t')$ is predicted by feeding the updated query token into a token-wise FFN, and $P_i^{l-1}(t')$ serves as the reference in the l-th decoder layer. The final predicted 2D poses for frame t are $\{P_i(t)\}_{i=1}^M$, where each $P_i(t) = P_i^3(t)$.

Spatiotemporal Joint Decoder. The joint decoder captures kinematic dependencies between articulated joints, further refining each predicted pose at the joint level. For each pose $P_i(t)$ predicted by the spatiotemporal pose decoder, the joint decoder uses its joint locations as initial reference points and refines them via cross-attention between a shared set of J learnable joint query tokens $Q^o \in \mathbb{R}^{J\times D}$ and the feature tokens from all input frames,

$$\hat{Q}^o = \text{STJD}(Q^o, \{\hat{\tau}(t')\}_{t'=t-T}^{t+T} | P_i(t)), \tag{7}$$

where STJD$(\cdot)$ denotes the spatiotemporal joint decoder, and $\hat{Q}^o$ are the updated joint query tokens, each aggregating features for the corresponding joint across frames. STJD has three layers and follows the architecture of STPD: each layer applies self-attention among joint queries (joint-to-joint attention) followed by deformable cross-attention between joint queries and multi-scale feature tokens (feature-to-joint attention). As in STPD, reference points are progressively refined layer by layer, with relative offsets regressed by a token-wise FFN from the updated joint query tokens.

Note that it is straightforward to extend the spatiotemporal joint decoder to the previously described video transformer baseline by replacing the spatial feature tokens of all input frames $\{\hat{\tau}(t')\}_{t'=t-T}^{t+T}$ learned by the SE module in Eq. 4 with the spatiotemporal feature tokens of the current frame $\tilde{\tau}(t)$ learned by the STE module in Eq. 1.

Loss Function

During training, the entire model is optimized end-to-end by minimizing the discrepancy between the ground-truth poses and all predicted poses at different stages, including the initial pose and its successive refinements for the current frame. Similar to (Shi et al. 2022), we adopt a set-based Hungarian loss to enforce a one-to-one assignment between predictions and ground-truth poses. Following (He and Yang 2024; Li et al. 2021), we replace conventional regression losses (l_1 or l_2) with the residual log-likelihood estimation loss ($\mathcal{L}_{rle}$) for pose regression in the pose decoder and joint regression in

the joint decoder. Additionally, we use the same classification loss ($\mathcal{L}_{cls}$) as in (Zhu et al. 2020) for regressing pose confidence scores. The total loss $\mathcal{L}$ is defined as

$$\mathcal{L} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{rle}\mathcal{L}_{rle}, \tag{8}$$

where λ_{cls} and λ_{rle} are weighting factors.

Experiments

Experimental Settings

We evaluated our model on three widely used video benchmark datasets: PoseTrack2017 (Iqbal, Milan, and Gall 2017), PoseTrack2018 (Andriluka et al. 2018), and PoseTrack21 (Doering et al. 2022). Each dataset contains dynamic video sequences with complex scenes, including significant occlusions and rapid motion in crowded environments. To assess performance, we used the Average Precision (AP) metric (Sun et al. 2019; Bertasius et al. 2019; Liu et al. 2021a; Li et al. 2021), where AP is computed for each keypoint and the mean Average Precision (mAP) is obtained by averaging AP over all keypoints. Each result is obtained through 2–4 runs. Our model was implemented in PyTorch, with the backbone network pre-trained on the COCO dataset. The temporal span T was set to 1, *i.e.*, one preceding frame and one subsequent frame, totaling two auxiliary frames. For additional implementation details, please refer to Appendix B of the supplementary material.

Comparison with State-of-the-art Methods

We begin with a comprehensive performance comparison against state-of-the-art methods on the PoseTrack2017 dataset, and subsequently extend our evaluation to the PoseTrack2018 and PoseTrack21 datasets.

Results on the PoseTrack2017 Dataset

Comparison with Image-based End-to-End Methods. To thoroughly evaluate the effectiveness of our proposed end-to-end method for video input, we first compare it against state-of-the-art image-based end-to-end methods, specifically PETR (Shi et al. 2022) and GroupPose (Liu et al. 2023). To ensure a comprehensive evaluation, we employ three backbone networks: ResNet-50, HRNet-W48, and Swin-L, applying each approach using identical pre-trained models for these backbones. Additionally, we replace the conventional regression losses (l_1 or l_2) with the residual log-likelihood estimation loss in our re-implementations of PETR and GroupPose to ensure consistency and fairness.

Quantitative results: As shown in Table 1, our video-based method consistently achieves substantial performance gains across all backbones. For example, it surpasses PETR (Shi et al. 2022) by **6.0** mAP using ResNet-50 and by **4.7** mAP using HRNet-W48. These results highlight the importance of leveraging temporal cues from adjacent frames, which image-based methods inherently lack.

Qualitative results: By effectively leveraging temporal dependencies across consecutive frames, our video-based end-to-end framework demonstrates improved robustness in challenging scenarios such as occlusions and motion blur, which are common in real-world videos (see Fig. 3). These qualitative results clearly show that our method achieves significantly better performance than prior image-based end-to-end methods, further underscoring the importance of temporal cues for video-based tasks.

Method	Bkbone	Head	Should.	Elbow	Wrist	Hip	Knee	Ankle	Mean
Two-Stage (Top-Down)									
Image-Based									
SimBase. (2018)	ResNet-152	81.7	83.4	80.0	72.4	75.3	74.8	67.1	76.7
HRNet (2019)	HRNet-W48	82.1	83.6	80.4	73.3	75.5	75.3	68.5	77.3
Video-Based									
PoseTrack (2018)	ResNet-101	67.5	70.2	62.0	51.7	60.7	58.7	49.8	60.6
FastPose (2019)	ResNet-101	80.0	80.3	69.5	59.1	71.4	67.5	59.4	70.3
STEmbed (2019)	ResNet-152	83.8	81.6	77.1	70.0	77.4	74.5	70.8	77.0
PoseWarp. (2019)	HRNet-W48	81.4	88.3	83.9	78.0	82.4	80.5	73.6	81.2
DCPose (2021a)	HRNet-W48	88.0	88.7	84.1	78.4	83.0	81.4	74.2	82.8
DetTrack (2020)	HRNet-W48	89.4	89.7	85.5	79.5	82.4	80.8	76.4	83.8
FAMIPose (2022)	HRNet-W48	89.1	89.5	84.8	79.0	84.2	82.3	74.9	83.9
TDMI (2023b)†	HRNet-W48	90.6	91.0	87.2	81.5	85.2	84.5	78.7	85.9
DiffPose (2023a)†	HRNet-W48	89.0	91.2	87.4	83.5	85.5	87.2	80.2	86.4
DSTA (2024)	ResNet-50	87.3	86.8	80.0	71.9	78.6	75.8	65.4	78.6
DSTA (2024)	HRNet-W48	87.6	88.1	84.8	80.1	83.6	82.8	75.1	83.4
DSTA (2024)	ViT-H	88.1	88.3	86.4	81.1	84.2	84.2	76.3	84.3
End-to-End									
Image-Based									
PETR (2022)	ResNet-50	80.5	80.8	71.3	62.1	73.4	68.5	61.2	71.7
GroupPose (2023)	ResNet-50	82.4	82.1	73.3	64.3	74.4	70.7	63.7	73.6
PETR (2022)	HRNet-W48	82.4	83.2	74.4	70.8	74.5	72.3	66.9	75.4
GroupPose (2023)	HRNet-W48	83.3	84.3	77.8	70.3	75.6	72.8	66.8	76.3
PETR (2022)	Swin-L	83.3	84.3	78.3	71.3	76.4	73.4	67.6	76.8
GroupPose (2023)	Swin-L	83.9	84.7	78.8	70.6	77.5	74.4	68.7	77.4
Video-Based									
PAVE-Net (Ours)	ResNet-50	86.5	87.4	78.9	69.3	78.2	73.8	65.8	77.7
PAVE-Net (Ours)	HRNet-W48	87.1	88.4	80.9	73.9	80.3	76.9	69.9	80.1
PAVE-Net (Ours)	Swin-L	88.2	89.1	81.7	74.8	81.6	78.5	71.8	81.3

Table 1: Comparison with SOTA methods on the PoseTrack2017 validation set. '†' indicates results using 4 auxiliary frames; otherwise, 2 auxiliary frames are used.

Comparison with Video-based Methods. Current state-of-the-art methods for video-based human pose estimation predominantly adopt a two-stage top-down framework: first detecting human instances frame by frame, then applying temporal modeling for single-person pose estimation. By explicitly focusing on individual subjects, these top-down methods have achieved superior performance. When using identical backbone networks for feature extraction, our proposed approach achieves results comparable to these state-of-the-art top-down methods. For example, with a ResNet-50 backbone, our method achieves an mAP of **77.7**, closely matching the 78.6 mAP reported by DSTA (He and Yang 2024) using the same backbone. Our approach is flexible and readily integrates with various backbones; when using the stronger Swin-L (Liu et al. 2021b) backbone, our method further improves performance, reaching **81.3** mAP.

It is important to emphasize that methods leveraging temporal cues, including PoseWarper (Bertasius et al. 2019), DCPose (Liu et al. 2021a), DetTrack (Wang, Tighe, and Modolo 2020), FAMI-Pose (Liu et al. 2022), TDMI (Feng

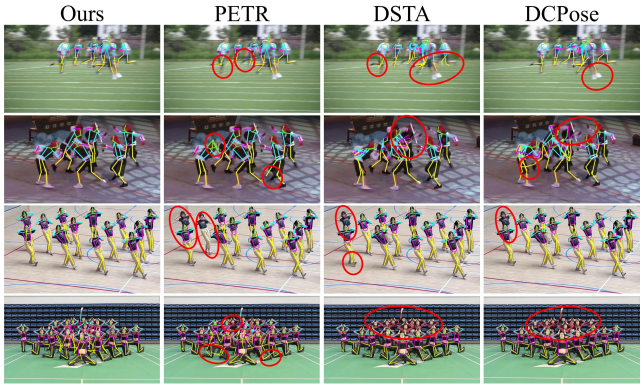

Figure 3: Qualitative comparison of our PAVE-Net, PETR (Shi et al. 2022), DSTA (He and Yang 2024), and DCPose (Liu et al. 2021a), highlighting challenges such as occlusions, motion blur, and crowded scenarios. The top two rows are from the PoseTrack dataset, while the bottom two rows are from in-the-wild videos. Inaccurate predictions are marked with red solid circles. Better viewed with zoom.

et al. 2023b), DiffPose (Feng et al. 2023a), DSTA (He and Yang 2024), and our PAVE-Net, consistently outperform single-frame methods such as SimpleBaseline (Xiao, Wu, and Wei 2018), HRNet (Sun et al. 2019), PETR (Shi et al. 2022), and GroupPose (Liu et al. 2023). This reaffirms the critical role of temporal cues from adjacent frames for more accurate and robust pose estimation in video scenarios.

Qualitative results: Notably, existing two-stage video-based methods rely heavily on human detector performance and often struggle in challenging scenarios such as crowded environments, leading to degraded performance (see Fig. 3). In contrast, our end-to-end video-based method removes the need for explicit human detection and consistently produces accurate and robust pose estimates even in these difficult conditions, as demonstrated in Fig. 3.

Inference Time Comparison. We assess inference time, with results summarized in Table 2. For a fair comparison, all methods use the same HRNet-W48 backbone, and two auxiliary frames are employed for all video-based approaches. Existing video-based MPPE methods, which follow a two-stage top-down framework, exhibit a significant increase in inference time as the number of people in the scene grows. In contrast, our end-to-end PAVE-Net maintains consistent inference time regardless of the number of individuals, demonstrating near-constant scalability with respect to scene complexity. Moreover, top-down approaches require a separate human detector, further increasing computational overhead. As a result, compared to existing video-based methods, PAVE-Net achieves substantially lower inference time, particularly in crowded scenes. For example, in 20-person scenes, our method reduces inference time by **79%** compared to DCPose (Liu et al. 2021a) and **76%** compared to DSTA (He and Yang 2024). Notably, even compared to image-based methods, our approach demonstrates comparable inference efficiency. Additional comparisons using other backbones (*i.e.*, ResNet-50 and Swin-L) are provided in Appendix C of the supplementary material.

This efficiency and scalability are especially valuable for industrial applications, where not only reliability but also real-time video processing is critical.

Method	Number of Persons 1	3	5	10	20
Two-Stage (Top-Down)					
DCPose (Liu et al. 2021a)	150	204	292	431	721
DSTA (He and Yang 2024)	122	181	265	418	631
End-to-End					
PETR (Shi et al. 2022)†	116				
GroupPose (Liu et al. 2023)†	89				
PAVE-Net (Ours)	153				

Table 2: Inference time (*ms*) with HRNet-W48 backbone, measured on an A800. '†' denotes image-based methods.

Method	mAP	Inference Time (*ms*)
Baseline	74.5	336
PAVE-Net-STE	76.9	378
PAVE-Net	**77.7**	**132**

Table 3: Ablation of different design strategies in PAVE-Net.

Results on the PoseTrack2018/21 Datasets
We further evaluate our model on the PoseTrack2018 and PoseTrack21 datasets. Due to space constraints, detailed results are provided in Appendix D of the supplementary material. These results clearly show that our approach consistently outperforms image-based end-to-end methods across all backbones. Moreover, our method achieves performance comparable to state-of-the-art two-stage video-based approaches. Specifically, with the ResNet-50 backbone, our method achieves **76.5** and **76.2** mAP on the two datasets, respectively. Using the Swin-L backbone, performance further improves by **3.6** and **3.5** points, respectively.

Ablation Study

We conduct ablation experiments on the PoseTrack2017 validation set to assess the impact of each component, using ResNet-50 as the backbone for all evaluations.

Video Transformer Baseline *vs.* PAVE-Net. As shown in Fig. 2, the video transformer baseline employs a spatiotemporal encoder (STE) to capture global dependencies among feature tokens across multiple frames. While straightforward, this design incurs significantly higher computational complexity and may entangle temporal features from different individuals. In contrast, PAVE-Net adopts a two-phase strategy: it first encodes local spatial dependencies within individual frames (SE), then employs a pose-aware spatiotemporal decoder to efficiently aggregate features corresponding to the same individual across consecutive frames. Table 3 presents the performance comparison between the video transformer baseline and PAVE-Net. PAVE-Net achieves a substantial reduction in inference time, attaining **132** *ms* versus 336 *ms* for the baseline, while simultaneously improving performance by **3.2** mAP points (**77.7** vs. 74.5). Moreover, we also experimented with replacing the SE module in PAVE-Net with the STE module. This not only led to a slight drop in accuracy (**76.9**) but also substan-

STPD		STJD		mAP
#Params	GFLOPs	#Params	GFLOPs	
12.34M	7.31	5.43M	9.72	
✗		✗		61.4
✓		✗		74.3
✓		✓		**77.7**

Table 4: Ablation of different modules in PAVE-Net.

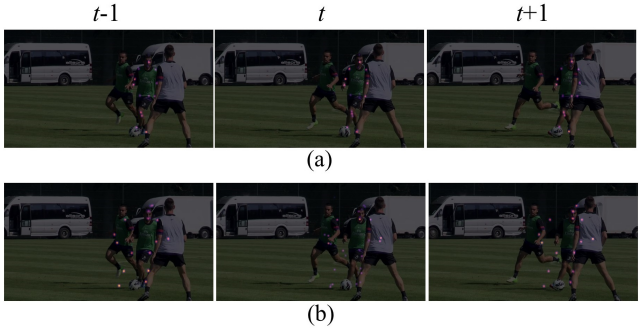

Figure 4: Features attended by the query token of the central target person across consecutive frames, highlighted with colored circles. While our pose-aware attention focuses exclusively on the target person's features (a), features from other individuals are mistakenly attended without it (b). Best viewed with zoom.

tially increased inference time to **378** *ms*. These results indicate that the SE module provides a more favorable trade-off between accuracy and efficiency within our framework.

Impact of Different Modules in PAVE-Net. Table 4 evaluates the contribution of each module in our approach. PAVE-Net consists of three key components: the spatial encoder, the spatiotemporal pose decoder (STPD), and the spatiotemporal joint decoder (STJD). The STPD predicts 2D poses for the current frame, which are then refined by the STJD. As shown, both STPD and STJD exhibit relatively relatively few and low computational cost, making them lightweight and suitable for real-time applications. Using only the STPD yields an mAP of 74.3, while adding the STJD further improves performance by **3.4** points, achieving 77.7 mAP. To further analyze the framework, we conduct an ablation where both STPD and STJD are removed. In this configuration, only feature tokens from the current frame, processed by the spatial encoder, are used to directly regress full-body poses. This reduces PAVE-Net to an image-based method where feature-to-joint misalignment tends to occur, as noted in (Geng et al. 2021; Shi et al. 2022), leading to a significant performance drop to just **61.4** mAP.

Impact of Pose-aware Attention Mechanism. In the spatiotemporal pose decoder (STPD), we introduce a novel pose-aware attention mechanism that leverages reference poses predicted from the feature tokens $\hat{\tau}(t)$ of the current frame t to ensure that each query token attends to feature tokens associated with the same individual across multiple frames. We also experimented with using randomly initialized learnable parameters as reference poses, as done in (Shi et al. 2022). As shown in Fig. 4, pose-aware attention ensures that each query token effectively aggregates features corresponding to the same individual across consecutive frames, whereas without it, features from different individuals can become mixed. Consequently, the accuracy drops to merely 34.6 mAP, resulting in a substantial performance decline of up to **43.1** points.

Number of Layers	1	2	3	4	5
mAP (%)	67.4	74.8	77.7	77.2	77.0

Table 5: Different layer numbers in STPD and STJD.

These results validate that precise temporal association of the same individuals is essential for preventing feature mixing and for enabling effective temporal information aggregation. They also demonstrate the efficacy of our proposed pose-aware attention mechanism in establishing reliable temporal associations across frames.

Number of Layers in STPD and STJD. Table 5 presents the ablation study on the number of layers used in STPD and STJD. As the number of layers increases, model accuracy initially improves, as expected. However, when the number exceeds 3, the model becomes overly complex, leading to overfitting and a decline in accuracy. Moreover, increasing the number of layers also raises computational cost. Considering the trade-off between accuracy and efficiency, we choose 3 layers as the default setting.

Auxiliary Frames. In this ablation study, we examine the impact of varying the number of auxiliary frames. As shown in Table 6, increasing auxiliary frames consistently improves performance across different backbone networks. This aligns with our intuition that additional frames provide complementary temporal information, thereby enhancing pose estimation accuracy for the key frame.

#Auxiliary Frame	ResNet-50	HRNet-W48	Swin-L
2 {-1, +1}	77.7	80.1	81.3
4 {-2, -1, +1, +2}	**78.2**	**80.5**	**81.7**

Table 6: Different number of auxiliary frames. '-' denotes previous frames, while '+' denotes subsequent frames.

Conclusion

We present PAVE-Net, the first fully end-to-end framework for multi-person 2D pose estimation in videos, eliminating the need for heuristic steps such as NMS and RoI cropping. By combining local spatial encoding, global spatiotemporal fusion, and a pose-aware attention mechanism, PAVE-Net enables robust temporal feature aggregation across frames. Extensive experiments demonstrate that PAVE-Net significantly outperforms image-based end-to-end methods, while achieving accuracy comparable to state-of-the-art two-stage video-based approaches, with substantial efficiency gains.

Acknowledgments

This work was supported by "Pioneer" and "Leading Goose" R&D Program of Zhejiang Province (2024C01167) and the Fundamental Research Funds for the Provincial Universities of Zhejiang (FR24005Z).

References

Andriluka, M.; Iqbal, U.; Insafutdinov, E.; Pishchulin, L.; Milan, A.; Gall, J.; and Schiele, B. 2018. PoseTrack: A Benchmark for Human Pose Estimation and Tracking. In *CVPR*, 5167–5176.

Bertasius, G.; Feichtenhofer, C.; Tran, D.; Shi, J.; and Torresani, L. 2019. Learning Temporal Pose Estimation from Sparsely Labeled Videos. In *NIPS*.

Doering, A.; Chen, D.; Zhang, S.; Schiele, B.; and Gall, J. 2022. PoseTrack21: A Dataset for Person Search, Multi-Object Tracking and Multi-Person Pose Tracking. In *CVPR*, 20931–20940.

Feng, R.; Gao, Y.; Elden Tse, T. H.; Ma, X.; and Chang, H. J. 2023a. DiffPose: SpatioTemporal Diffusion Model for Video-Based Human Pose Estimation. In *ICCV*, 14815–14826.

Feng, R.; Gao, Y.; Ma, X.; Tse, T. H. E.; and Chang, H. J. 2023b. Mutual Information-Based Temporal Difference Learning for Human Pose Estimation in Video. In *CVPR*, 17131–17141.

Geng, Z.; Sun, K.; Xiao, B.; Zhang, Z.; and Wang, J. 2021. Bottom-Up Human Pose Estimation Via Disentangled Keypoint Regression. In *CVPR*, 14671–14681.

Geng, Z.; Wang, C.; Wei, Y.; Liu, Z.; Li, H.; and Hu, H. 2023. Human Pose as Compositional Tokens. In *CVPR*, 660–671.

He, J.; and Yang, W. 2024. Video-Based Human Pose Regression via Decoupled Space-Time Aggregation . In *CVPR*, 1022–1031.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *CVPR*, 770–778.

Iqbal, U.; Milan, A.; and Gall, J. 2017. PoseTrack: Joint Multi-person Pose Estimation and Tracking. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4654–4663.

Jin, K.-M.; Lee, G.-H.; and Lee, S.-W. 2022. OTPose: Occlusion-Aware Transformer for Pose Estimation in Sparsely-Labeled Videos. In *2022 IEEE International Conference on Systems, Man, and Cybernetics*, 3255–3260.

Jin, S.; Liu, W.; Ouyang, W.; and Qian, C. 2019. Multi-person articulated tracking with spatial and temporal embeddings. In *CVPR*, 5664–5673.

Khirodkar, R.; Bagautdinov, T.; Martinez, J.; Zhaoen, S.; James, A.; Selednik, P.; Anderson, S.; and Saito, S. 2024. Sapiens: Foundation for Human Vision Models. In *ECCV*, 1–22.

Li, J.; Bian, S.; Zeng, A.; Wang, C.; Pang, B.; Liu, W.; and Lu, C. 2021. Human Pose Regression with Residual Log-likelihood Estimation. In *ICCV*, 11005–11014.

Li, Y.-L.; Liu, X.; Lu, H.; Wang, S.; Liu, J.; Li, J.; and Lu, C. 2020. Detailed 2D-3D Joint Representation for Human-Object Interaction. In *CVPR*, 10163–10172.

Liu, H.; Chen, Q.; Tan, Z.; Liu, J.-J.; Wang, J.; Su, X.; Li, X.; Yao, K.; Han, J.; Ding, E.; Zhao, Y.; and Wang, J. 2023. Group Pose: A Simple Baseline for End-to-End Multi-person Pose Estimation. In *ICCV*, 14983–14992.

Liu, Z.; Chen, H.; Feng, R.; Wu, S.; Ji, S.; Yang, B.; and Wang, X. 2021a. Deep Dual Consecutive Network for Human Pose Estimation. In *CVPR*, 525–534.

Liu, Z.; Feng, R.; Chen, H.; Wu, S.; Gao, Y.; Gao, Y.; and Wang, X. 2022. Temporal Feature Alignment and Mutual Information Maximization for Video-Based Human Pose Estimation. In *CVPR*, 10996–11006.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021b. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *ICCV*, 9992–10002.

Qiu, Z.; Yang, Q.; Wang, J.; Feng, H.; Han, J.; Ding, E.; Xu, C.; Fu, D.; and Wang, J. 2023. PSVT: End-to-End Multi-Person 3D Pose and Shape Estimation with Progressive Video Transformers. In *CVPR*, 21254–21263.

Shi, D.; Wei, X.; Li, L.; Ren, Y.; and Tan, W. 2022. End-to-End Multi-Person Pose Estimation with Transformers. In *CVPR*, 11059–11068.

Sun, K.; Xiao, B.; Liu, D.; and Wang, J. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. In *CVPR*, 5686–5696.

Sun, Z.; Ke, Q.; Rahmani, H.; Bennamoun, M.; Wang, G.; and Liu, J. 2023. Human Action Recognition From Various Data Modalities: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3200–3225.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All You Need. In *NIPS*, 6000–6010.

Wang, D.; Xuan, S.; and Zhang, S. 2024. LocLLM: Exploiting Generalizable Human Keypoint Localization via Large Language Model. In *CVPR*, 614–623.

Wang, D.; and Zhang, S. 2022. Contextual instance decoupling for robust multi-person pose estimation. In *CVPR*, 11060–11068.

Wang, M.; Tighe, J.; and Modolo, D. 2020. Combining detection and tracking for human pose estimation in videos. In *CVPR*, 11088–11096.

Xiao, B.; Wu, H.; and Wei, Y. 2018. Simple Baselines for Human Pose Estimation and Tracking. In *ECCV*.

Xu, Y.; Zhang, J.; Zhang, Q.; and Tao, D. 2022. Vitpose: Simple vision transformer baselines for human pose estimation. *NIPS*, 35: 38571–38584.

Yang, J.; Zeng, A.; Liu, S.; Li, F.; Zhang, R.; and Zhang, L. 2023. Explicit Box Detection Unifies End-to-End Multi-Person Pose Estimation. In *ICLR*.

Zhang, J.; Zhu, Z.; Zou, W.; Li, P.; Li, Y.; Su, H.; and Huang, G. 2019. Fastpose: Towards real-time pose estimation and tracking via scale-normalized multi-task networks. *arXiv preprint arXiv:1908.05593*.

Zhang, Y.; An, L.; Yu, T.; Li, X.; Li, K.; and Liu, Y. 2020. 4D Association Graph for Realtime Multi-Person Motion Capture Using Multiple Video Cameras. In *CVPR*, 1321–1330.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv preprint arXiv:2010.04159*.