electronics

MDPI

Article

Bidirectional Temporal Pose Matching for Tracking

Yichuan Fang [†,‡], Qingxuan Shi [*,‡] and Zhen Yang [†,‡]

Hebei Machine Vision Engineering Research Center, Hebei University, Baoding 071002, China; 20217019001@stumail.hbu.edu.cn (Y.F.); 20217019047@stumail.hbu.edu.cn (Z.Y.)

* Correspondence: qingxuanshi@hbu.edu.cn

† These authors contributed equally to this work.

‡ Current address: School of Cyber Security and Computer, Hebei University, Baoding 071002, China.

Abstract: Multi-person pose tracking is a challenging task. It requires identifying the human poses in each frame and matching them across time. This task still faces two main challenges. Firstly, sudden camera zooming and drastic pose changes between adjacent frames may result in mismatched poses between them. Secondly, the time relationships modeled by most existing methods provide insufficient information in scenarios with long-term occlusion. In this paper, to address the first challenge, we propagate the bounding boxes of the current frame to the previous frame for pose estimation, and match the estimated results with the previous ones, which we call the Backward Temporal Pose-Matching (BTPM) module. To solve the second challenge, we design an Association Across Multiple Frames (AAMF) module that utilizes long-term temporal relationships to supplement tracking information lost in the previous frames as a Re-identification (Re-id) technique. Specifically, we select keyframes with a fixed step size in the videos and label other frames as general frames. In the keyframes, we use the BTPM module and the AAMF module to perform tracking. In the general frames, we propagate poses in the previous frame to the current frame for pose estimation and association, which we call the Forward Temporal Pose-Matching (FTPM) module. If the pose association fails, the current frame will be set as a keyframe, and tracking will be re-performed. In the PoseTrack 2018 benchmark tests, our method shows significant improvements over the baseline methods, with improvements of 2.1 and 1.1 in mean Average Precision (mAP) and Multi-Object Tracking Accuracy (MOTA), respectively.

Keywords: multi-person pose estimation; pose tracking; temporal association; pose matching

Citation: Fang, Y.; Shi, Q.; Yang, Z. Bidirectional Temporal Pose Matching for Tracking. *Electronics* **2024**, *13*, 442. https://doi.org/10.3390/electronics13020442

Academic Editor: Beiwen Li

Received: 11 December 2023
Revised: 15 January 2024
Accepted: 19 January 2024
Published: 21 January 2024

1. Introduction

Multi-person pose tracking is a fundamental challenge in computer vision. It can be understood as connecting the poses estimated by a pose estimator in a coherent sequence in time and assigning a unique ID to the same person. The correct association of human pose trajectories is of great help for human action recognition, human interaction understanding, motion capture, animation design, etc. In addition, online multi-person pose tracking is also applied to real-time scenarios, such as autonomous driving and virtual interaction.

Currently, most multi-person pose-tracking methods follow a two-step approach of detection and tracking. They typically perform human detection on the current frame, extract information about the position and keypoints of each detected human object, and then match the human objects in the current frame with the human objects in previous frames based on this information, thereby achieving multi-person tracking. With the development of Convolutional Neural Networks (CNNs) [1–4] and the release of large-scale datasets such as MPII [5], LSP [6] and COCO [7], the ability of multi-person pose estimation (MPPE) [1,8–10] has been improved significantly. Some previous multi-person pose-tracking methods benefit from the accuracy advantage brought by MPPE and obtain better tracking results. However, some other multi-person pose-tracking methods [11–14]

do not rely on the selection of MPPE methods but try to improve the accuracy through tracking means.

With the introduction of video datasets such as PoseTrack 2017 [15] and PoseTrack 2018 [16], multi-person pose tracking receives more and more attention. Two-step methods are still prone to losing the identity of individuals who disappear for several frames and reappear in the current frame, especially when heavy occlusion occurs [17–19]. This is not surprising; some of these methods associate individuals only based on the pose or bounding box information of the previous and current frames, and then connect the results to the identity trajectories. Reappearing individuals are always assigned with new IDs. Some other methods attempt to compensate for this deficiency with Re-id modules, but they still heavily rely on temporal information close to the current frame. When individuals disappear for a relatively long period, these methods inevitably lead to the failure of tracking these individuals.

In this paper, we propose an online top–down tracker that aims to improve the performance of multi-person pose estimation and tracking by utilizing long temporal relationships to supplement missing information in middle frames due to occlusion and other factors. Our method follows the tracking by detection scheme, first locating the human body based on the detector, then estimating the keypoints of the human body, and finally tracking these keypoints by assigning a unique ID. Differently, in addition to the pose information from the pose estimator, we incorporate the pose information propagated over time.

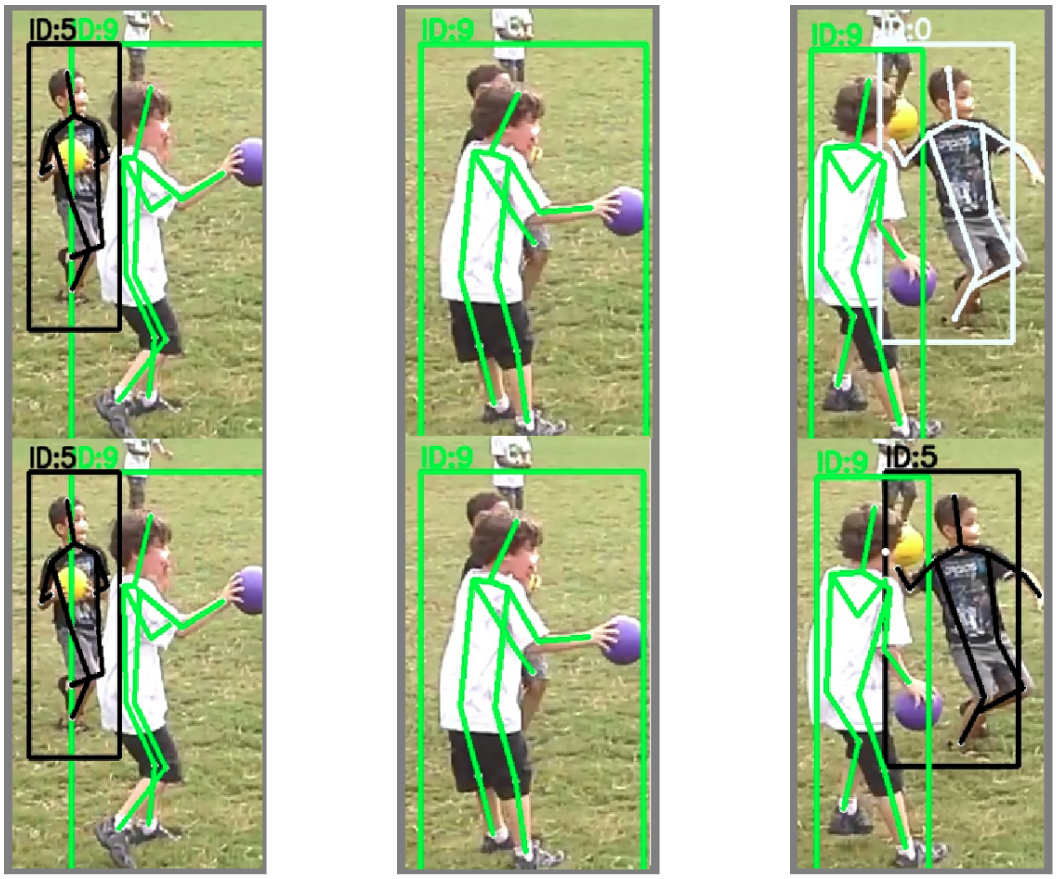
Given a video sequence, we divide the video frames into general frames and keyframes. In keyframes, we back-propagate the bounding box of the current frame to the previous frame and perform pose estimation to obtain a new pose, which is then matched with the pose results of the previous frame. We call it the BTPM module. This method avoids the problem of two poses not matching due to large movements of the same person between the previous and current frames. For some people in the video, they may reappear only after many frames due to the camera movement or person occlusion; we add an AAMF module that can trace back the tracking results of the previous frames to achieve personal Re-id. In general frames, we estimate the pose of the current frame based on the bounding box propagated from the previous frame and match it with the pose of the previous frame. We call it the FTPM module. This strategy does not consider detection clues, thereby reducing dependence on the detection results. When there is a pose that cannot be matched in the general frame, we set that frame as a keyframe and use the detection results to perform pose estimation and tracking again. We combine the two different tracking methods on general frames and keyframes, as well as the Re-id module, to achieve multi-person pose tracking. Compared with some existing methods, better mAP and MOTA results are obtained with our method. A qualitative example is shown in Figure 1. The top row is the tracking result from the baseline method; the bottom row is the tracking result from our method. And t represents the number of frames. It can be observed that our method, utilizing the Re-id module based on temporal relationships, accurately assigns IDs to human objects experiencing long-term occlusion and recurrence.

Our contributions are three-fold:

- We propose a Bidirectional Temporal Pose-Matching module as a new online pose-tracking framework, applicable to top–down human pose estimation methods. The novelty of this module lies in the reverse propagation of information. Unlike traditional forward propagation of temporal information, we introduce backward propagation of current frame pose information. This places the comparison of pose similarity in the previous frame, overcoming the challenge of drastic pose changes between adjacent frames.
- We propose a novel identity Re-id method called the AAMF module. Differing from the previous difficulty in reidentifying poses occluded for an extended period by relying solely on matching poses between consecutive frames, this module utilizes the

temporal relationships provided by frames with a larger span to supplement the lost pose information. This is the novelty of the AAMF Module.

- We demonstrate the effectiveness of our approach through extensive experiments. Our approach outperforms the baseline method by 2.1 mAP and 1.1 MOTA on the widely used PoseTrack 2018 [16] benchmark dataset.



(**a**) $t = 48$ (**b**) $t = 54$ (**c**) $t = 61$

Figure 1. A qualitative example.

2. Related Work

2.1. Multi-Person Pose Estimation

Recently, due to significant advancements in deep learning methods [20,21], there has been substantial improvement in the results of MPPE methods. MPPE is more difficult and challenging because it needs to figure out the number of people and their positions, and how to group keypoints for different people [22]. There are two main types of methods for MPPE based on their operation mode: top–down and bottom–up.

The top–down methods [23–29] first use a human detector [30–32] to obtain the candidate bounding boxes of each person from an image, and then apply a human pose estimator to obtain the keypoints of the human body. The bottom–up methods [33–38] first detect the human joints in the image, and then assemble the body joints into the human pose. Bottom–up methods can detect joints in complex scenes (person occlusion, camera high-speed motion, motion blur, etc.) and classify them into different human bodies, giving them an advantage in speed. However, their corresponding pose estimation performance is not ideal compared to top–down methods. On the contrary, top–down methods divide the process into two tasks: detecting bounding boxes in the image, and then estimating the pose based on reliable detection results to obtain more accurate results. Top–down methods do not require any joint grouping and provide additional bounding box information for

tracking, which bottom–up methods do not have. Therefore, we choose the top–down pose detection method and use our tracking method to complete pose tracking.

2.2. Multi-Person Pose Tracking

Expanding MPPE to videos raises the problem of multi-person pose tracking, which mainly addresses the issues of per-frame pose estimation and pose association between frames.

Bottom–up methods construct a spatiotemporal graph among detected keypoints without relying on bounding boxes. For example, Raaj et al. [18] extend the Part Affinity Field (PAF) [39] design for single image pose estimation to SpatioTemporal Affinity Fields (STAFs) in videos. Jin et al. [40] propose Spatial–Temporal Embed (ST-Embed) to learn the Spatial–Temporal Embedding of joints based on the idea of Associative Embedding [41].

Top–down methods utilize temporal context information to achieve identity association between poses or bounding boxes. The simple baseline method [17] first performs human pose estimation on single frames, and then matches them by calculating the similarity of poses using optical flow. Detect-and-Track (DAT) [42] extends Mask-RCNN [43] to 3D to form the same person's pipeline, and then associates the pipeline according to the position of the bounding boxes to realize tracking. KeyTrack [14] merges keypoint refinement techniques into pose estimation and evaluates similarity from different perspectives to achieve temporal pose association.

Additionally, pose tracking can be categorized into two working modes: online and offline. Online pose tracking refers to the real-time processing and tracking of continuous video streams or live inputs. Offline pose tracking involves the subsequent processing and tracking of recorded videos or image sequences, without the need to consider real-time constraints during the processing. Therefore, compared to offline pose tracking, online pose tracking imposes higher demands on algorithms and models.

2.3. Association of Identities

Many top–down pose-tracking methods often rely on specific pose estimators. This dependence can have an impact on the robustness and generalization of the model. In contrast, there are also methods that do not rely on the selection of the pose estimator but attempt to recover the accuracy loss caused by detection and pose estimation through different tracking methods. Alphapose [11] uses the Pose-Guided Attention (PGA) mechanism to enhance human identity features and integrates human proposal information based on bounding boxes and poses to achieve identity matching. Buizza et al. [12] use data assimilation to predict the results of the next frame and realize pose tracking. Algabri et al. [44] combine multiple features into a single joint feature and utilize an online enhancement method to continuously update features in each frame for the identification of target individuals. We utilize Bidirectional Temporal Pose Matching and Re-id methods to attain improved tracking performance.

3. Method

3.1. Overview of Our Approach

Our overall framework is shown in Figure 2. The upper part displays the tracking operations implemented on keyframes, and the lower part shows the tracking operations implemented on general frames. In the keyframes, the pose of the current frame and the pose of the previous frame are input into the BTPM module, which outputs a maximum matching value. If the matching value is less than a certain threshold, the pose of the current frame is compared with poses that are earlier in the timing in the AAMF module to output the final result. In the general frames, the pose results of the current and previous frames are input into the FTPM module. If there is a match, the tracking result is output. Otherwise, the frame is converted into a keyframe, and tracking is re-performed.

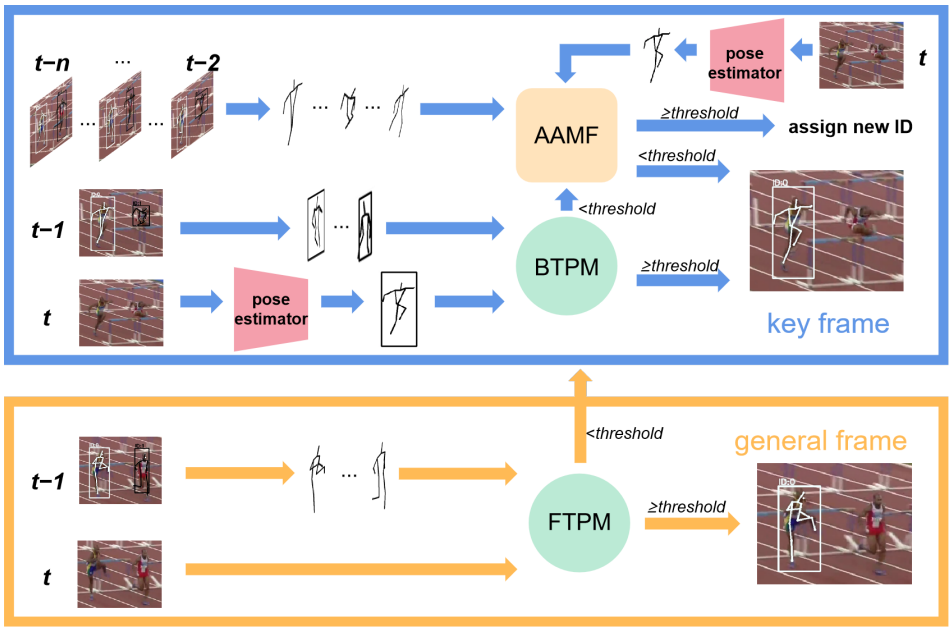


Figure 2. Overview of our method.

Specifically, for a video, we first filter out keyframes based on a fixed step size, and then set the other frames as general frames. If the t-th frame $\mathbf{F}_t$ is a keyframe, it will be input into the pose estimator to obtain the bounding boxes $\mathbf{B}_t = \{B_t^i\}_{i=1}^m$ and the poses $\mathbf{P}_t = \{P_t^i\}_{i=1}^m$. Here, m represents the number of poses in $\mathbf{F}_t$. These results are obtained by locating a bounding box around each person through a human detector and then performing pose estimation within each bounding box. Our goal is to assign a temporally continuous ID to each pose. To find optimal matching for the i-th person, we input its bounding box B_t^i and its pose P_t^i to the BTPM module together with the bounding boxes $\mathbf{B}_{t-1}$ and poses $\mathbf{P}_{t-1}$ from $\mathbf{F}_{t-1}$. BTPM outputs a maximum matching score. If the score is greater than the threshold, it indicates that the corresponding previous pose P_{t-1}^j is the best match for P_t^i, and therefore the corresponding id of P_{t-1}^j is assigned to P_t^i. Conversely, it suggests that the target may be lost due to occlusion or large-scale movement, and needs to be retrieved from a few frames further back. To this end, we input the pose P_t^i and $\left\{\mathbf{P}_{t-f}\right\}_{f=2}^n$ to the AAMF module. In practice, in order to extract more balanced temporal information, we set f to several evenly spaced numbers. The AAMF module first forms pose pairs by pairing P_t^i with each pose in $\left\{\mathbf{P}_{t-f}\right\}_{f=2}^n$. Subsequently, it calculates the distance between the center points of each pose pair and outputs the minimum value. If this value is less than a threshold, it means that the corresponding pose P_{t-f}^j from $\mathbf{F}_{t-f}$ is the best match for P_t^i, and the id of P_{t-f}^j is assigned to P_t^i. Otherwise, the AAMF module inputs all pose pairs into the Siamese graph convolutional network (SGCN) [13,45] to obtain the pose similarity difference values and outputs the minimum value. Similar to the center point matching, if this value is smaller than a certain threshold, it signifies that P_t^i has obtained the optimal match. Otherwise, it indicates that the pose may be a new pose in the current frame, and a new id is assigned to it.

If the current frame $\mathbf{F}_t$ is a general frame, it will be fed into the FTPM module together with the tracked results $\mathbf{B}_{t-1}$ and $\mathbf{P}_{t-1}$ of the previous frame $\mathbf{F}_{t-1}$. The FTPM module compares the pose estimation results between the current frame and the previous frame, and outputs the matching value between the poses. If this value is greater than a threshold, the id of the corresponding pose P_{t-1}^j is assigned to the current pose. Otherwise, the target tracking is lost in the general frame, and $\mathbf{F}_t$ is set as a keyframe, and the tracking is restarted.

3.2. Bidirectional Temporal Pose Matching

Our overall framework is shown in Figure 3. We design two distinct tracking methods for different frames. Specifically, for general frames, we want the tracking speed to be fast, so we use the forward temporal contrast of poses between adjacent frames as a tracking method, referred to as the FTPM module. In the FTPM module, we propagate the bounding box of the previous frame to the current frame and estimate the pose of the current frame. Then, we combine the new estimated pose with the pose of the previous frame to form a pose pair. The OKS result obtained from the calculation of the pose is used as the matching criteria. In contrast, for keyframes, we pay more attention to tracking accuracy, and at the same time, we need to consider some potential difficult issues between two frames, such as the large motion amplitude. In such a case, where the similarity between poses is low, simple pose comparison between the two frames may lead to tracking failure. To solve this problem, we propose a Backward Temporal Pose-Matching method, namely the BTPM module. In the BTPM module, we propagate the bounding box of the current frame to the previous frame and estimate the pose of the previous frame. Then, we combine the new estimated pose with the original poses of the previous frame to form pose pairs. Finally, in the pose-matching stage, the BTPM module relies on spatial consistency. The BTPM module utilizes the IoU and OKS results obtained from the calculation of each pair's pose as the matching criteria. Since in most cases, the pose position changes of the same person from the current and previous frames are not very large, when the optimal Intersection over Union (IoU) of two bounding boxes and the optimal Object Keypoint Similarity (OKS) [7,46] between two poses from the current frame and the previous frame exceed certain thresholds, we consider them to belong to the same person. Below is a detailed description of two modules proposed for different frames.

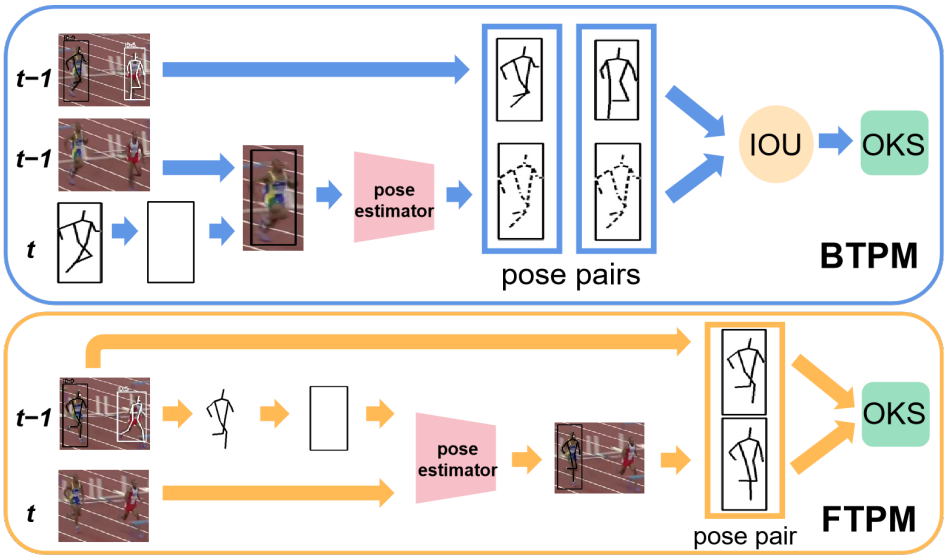


Figure 3. BTPM and FTPM.

3.2.1. BTPM Module

In the BTPM module, we first propagate the i-th detected box B_t^i in the current frame $\mathbf{F}_t$ backward to the previous frame $\mathbf{F}_{t-1}$ to obtain a bounding box $B_{t-1}^{'i}$. Then, we perform pose estimation in $B_{t-1}^{'i}$ and obtain the pose $P_{t-1}^{'i}$. The combination of $P_{t-1}^{'i}$ and each pose in $\mathbf{P}_{t-1}$ forms the pose pairs. For each pose pair $\left(P_{t-1}^{'i}, P_{t-1}^{j}\right)$, $j \in \{1, 2, \ldots, |\mathbf{P}_{t-1}|\}$, we compute their IoU value to remove pose pairs that are far apart in space. And then, we compare the remaining pose pairs based on their OKS to obtain a set of matching scores $\mathbf{M} = \{M_k\}_{k=1}^{K}$, where $1 \leq K \leq |\mathbf{P}_{t-1}|$. We select the maximum value $m^* = \max(\mathbf{M})$ from this set of scores as the optimal matching score. If the maximum value m^* is greater than

a threshold, we assign the tracking id^* corresponding to the maximum value to pose P_t^i. Otherwise, we execute the AAMF module. The calculation formula for OKS is as follows:

$$OKS = \frac{\sum_i e^{-\frac{d_i^2}{2s^2k_i^2}} \mu(v_i > 0)}{\sum_i \mu(v_i > 0)}, \tag{1}$$

where d_i represents the Euclidean distance between the i-th keypoints, k_i is a constant indicating the influence of the keypoint on the overall score, and v_i represents the visibility of the keypoint. In addition, μ is a Dirac delta function, whose definition is as follows:

$$\mu = \begin{cases} 1, & \text{if } v_i > 0, \\ 0, & \text{else}, \end{cases} \tag{2}$$

and s denotes the average of the bounding box areas of the two compared poses.

3.2.2. FTPM Module

In the general frame, the FTPM module determines the Region of Interest (RoI) [47] based on the j-th pose P_{t-1}^j estimated in the previous frame $\mathbf{F}_{t-1}$. Based on the constraints of the pose on the bounding box, we infer a bounding box $B_{t-1}^{'j}$ for this RoI and expand it by 20%. Compared to directly using the box detected in the current frame $\mathbf{F}_t$ or the previous frame $\mathbf{F}_{t-1}$, our method increases the information transmitted over time and minimizes losses caused by inaccurate detection. Then, we estimate the pose $P_t^{'j}$ based on the coordinate position of $B_{t-1}^{'j}$ in the current frame. If the similarity s between $P_t^{'j}$ and P_{t-1}^j exceeds the threshold, the tracking is considered successful, and the id corresponding to P_{t-1}^j from the previous frame is assigned to the pose P_t^j in the current frame. Conversely, if the similarity is less than or equal to the threshold, the tracking is considered failed.

Possible reasons for identity loss are as follows:

- P_t^i and P_{t-1}^j belong to the same person, but due to the large motion magnitude or sudden heavy occlusion during the short time period between the two frames, the difference between the two poses is too large.
- Due to the camera movement or sudden image zooming, the position offset of the same target in the two frames is too large to match the poses.
- The target person disappears in the current frame.

When a target-tracking loss occurs in the current frame, we set the current frame as a keyframe and reperform tracking. We combine BTPM modules and FTPM modules across different frames for pose tracking. This way, we can minimize the time spent on tracking while reducing the failure rate of target tracking and improving the tracking performance by transforming general frames into keyframes.

3.3. Association Across Multiple Frames

Although the BTPM module can solve the problem of mismatching two poses with a large difference in motion amplitude for the same person in the previous and current frames, it does not work when a person disappears, due to occlusion or camera movement, for a relatively long period and reappears in the current frame. As the BTPM module can only associate identities based on the temporal relationship between two adjacent frames, we add a Re-id module behind the BTPM module, called AAMF. This module not only uses the temporal relationship closer to the current frame to enhance the BTPM module but also exploits the temporal relationship further away from the current frame to compensate for the lost tracking of the BTPM module. Our AAMF module follows the overall flow as shown in Figure 4. The upper part is the center point matching module, and the lower part is the pose feature matching module. The distances between the center points of the current pose and each pose of the previous few frames are used as the criteria to assign identities.

If the best-matching result is still larger than the threshold, pose feature matching will be initiated. The feature matching module first inputs the current pose and the previous few frames' poses into the SGCN module to output the feature vectors of the poses. Then, the difference between the feature vectors is calculated, and the result with the smallest difference is selected as the matching result.

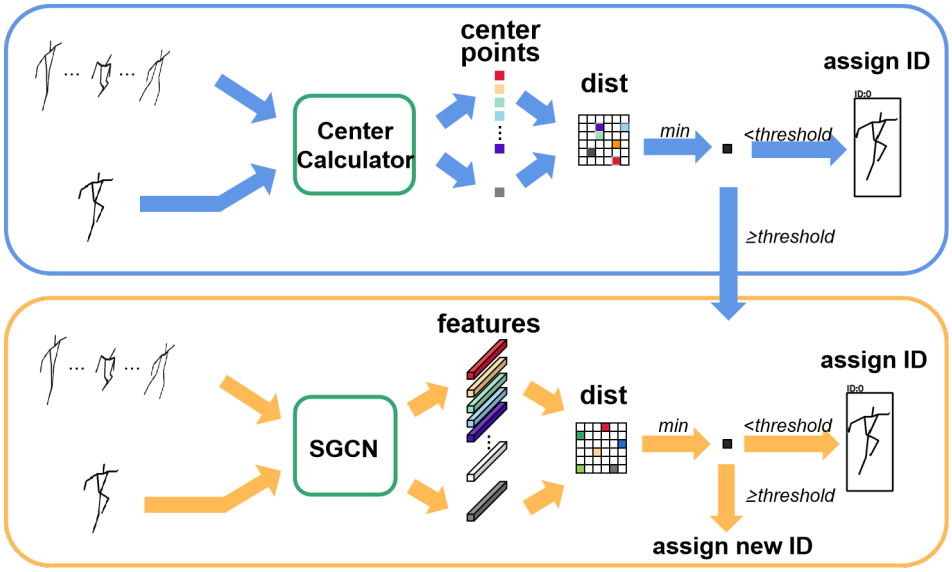


Figure 4. The AAMF module.

In the first part, to find a matching pose for the i-th pose P_t^i, we compute the center point for P_t^i from the t-th frame and poses $\left\{\mathbf{P}_{t-f}\right\}_{f=2}^{n}$ from the previous frames based on their corresponding keypoint coordinates. The method for calculating the center point is to take the average of the keypoint positions with scores greater than a certain confidence threshold. After obtaining the center points $\left\{\mathbf{C}_{t-f}\right\}_{f=2}^{n}$ of the previous frames and the center point C_t^i of P_t^i, we use a matrix D to record the distance between them. Here, D_a^b represents the distance between the center point C_a^b of P_a^b and the center point C_t^i. The number of poses in each frame is not the same, so we take the maximum number of poses as the number of columns in the matrix D and fill the missing positions with infinite values. Then, we select the pose $P_{a^*}^{b^*}$ corresponding to the smallest value in the matrix that is less than a certain threshold as the pose that best matches the current pose, and assign the id^* of $P_{a^*}^{b^*}$ to the current pose. Formally,

$$D_a^b = \left\| C_a^b - C_t^i \right\|_2, \tag{3}$$

$$(a^*, b^*) = argmin\left\{D_a^b\right\}. \tag{4}$$

While using distance comparison can handle scenarios where individuals suddenly disappear and reappear, it is still challenging to address the problem of severe position changes in images caused by camera movement and image zooming. To tackle this issue, we introduce a second part of the Re-id module based on pose features. The second part of the module supplements the Re-id of the first part. In addition, even if the human pose changes due to the camera movement and image zooming, the variation in their pose is not significant over a short period of a few frames, and the spatial consistency of the pose remains as reliable information. When the minimum distance obtained by the center point module is still greater than the threshold, we input the pose P_t^i and poses $\left\{\mathbf{P}_{t-f}\right\}_{f=2}^{n}$ from the previous frames into the SGCN module. Given that the poses change little over a short

period of time, we choose a relatively small value for n to select the previous frames for comparison so that they are closer to the current frame.

The SGCN module outputs 128-d feature vectors V_t^i and V_{t-f}^j for the corresponding poses. The feature vectors intrinsically encode the spatial relationships between human joints. Subsequently, similar to the center point module, we calculate the difference between the features of P_t^i and the poses of the previous frames, and record the differences in a matrix Ds. We select the pose $P_{a^*}^{b^*}$ corresponding to the minimum value in matrix Ds that is smaller than a certain threshold as the pose that best matches the current pose and assign the id^* of $P_{a^*}^{b^*}$ to P_t^i. a^* and b^* can be obtained as follows:

$$Ds_a^b = \left\| V_a^b - V_t^i \right\|_2, \tag{5}$$

$$(a^*, b^*) = argmin\left\{ Ds_a^b \right\}. \tag{6}$$

4. Experiments

4.1. Datasets

We mainly conduct experiments on the Posetrack 2017 [15] and Posetrack 2018 [16] datasets, which are two large-scale datasets for human pose estimation and tracking. They include challenging video sequences with various human actions in real-world scenarios, such as collisions, occlusion, and background clutter. The Posetrack 2017 dataset contains 250 video sequences for training and 50 video sequences for validation. Posetrack 2018 adds more videos, with 593 video sequences for training and 74 video sequences for validation. In each video sequence, each pose is annotated with 15 keypoints, each of which is associated with a unique ID for that pose. The training videos have dense annotations for the middle 30 frames of each video, while the validation videos have annotations for every 4th frame in addition to the middle frames. We use the combined training set of PoseTrack 2017 and COCO [7] for training, and evaluate on the validation set of Posetrack 2018.

4.2. Evaluation Metrics

We evaluate our method in terms of both human pose estimation and tracking. For human pose estimation, we use mAP [46,48] as the evaluation metric, while for tracking, we use MOTA [16,49]. MOTA considers three issues: false negatives (FNs), false positives (FPs), and the identity switch (IDSW) rate. The following is the formula for calculating MOTA for each body joint i, where t represents the current time step and GT stands for the Ground Truth:

$$MOTA^i = 1 - \frac{\sum_t (FN_t^i + FP_t^i + IDSW_t^i)}{\sum_t GT_t^i}. \tag{7}$$

We independently calculate these two evaluation metrics for each body joint, and then obtain the final results by averaging the results for each joint. Since the evaluation of MOTA requires the filtering of joints with a certain threshold, the performance of human pose tracking is evaluated based on the filtered joints. We evaluate the performance of human pose estimation using the evaluation results of all joints and filtered joints. These performance results provide both the overall results of the human pose estimation and the balanced results between the pose estimation and pose tracking. Additionally, since the tracking process only affects the IDSW metric in MOTA, where smaller IDSW values indicate fewer identity switches during tracking and better MOTA results, we also evaluate the performance of our method and other methods in terms of IDSW, calculated as follows:

$$IDSW^i = \frac{\sum_t IDSW_t^i}{\sum_t GT_t^i}. \tag{8}$$

4.3. Implementation Details

We train the pose estimation network with a batch size of 24 for 260 epochs on the merged dataset of PoseTrack 2017 and COCO [7]. The initial learning rate is set to 0.0005, and the learning rate is halved every 60 epochs. We then fine-tune the network for 40 epochs on PoseTrack 2017 to increase the accuracy of the head and neck keypoint regression, which is different from the COCO dataset. We use FPN [50] as the human bounding box detector and use MSRA152 [17] as the pose estimator.

4.4. Comparison with Other Methods of Identity Association

4.4.1. Quantitative Results

In this section, we compare our method with other methods that associate the pose estimation results to achieve real-time tracking. The comparison between our method and other methods in terms of human pose estimation and pose tracking on the Posetrack 2018 validation set is shown in Tables 1 and 2. Table 3 illustrates the comparison between our method and other approaches in terms of IDSW. A smaller IDSW value indicates fewer identity switches during the tracking process. Note that AP^T represents the result of the human pose estimation (with filtering), where the threshold is used for filtering low-confidence joints for pose tracking. In Table 1, compared to our baseline method LightTrack [13], our method improves the mAP of all keypoints by 1.8, and AP^T by 2.1. This is because our module reduces the situation where the pose estimation results are not ideal. In terms of the MOTA evaluation metric shown in Table 2, our method has a similar inference time to the baseline method but achieves a higher MOTA by 1.1, and is better than most other tracking methods. Furthermore, Alphapose [11] exhibits a shorter inference time, possibly attributed to the utilization of lower-resolution images, thereby reducing computational demands. Compared to the other methods in the table, our approach exhibits slight differences in mAP and MOTA. However, our method outperforms them in terms of both IDSW and AP^T, indicating its ability to better maintain individual identities during the tracking process. There are also some methods, such as [51], that achieve good results on the Posetrack 2018 dataset. However, they have some enhancement methods in the estimation module, and our comparison is mainly focused on the performance improvement brought by the tracking methods, so our method is not directly compared with them.

Table 1. Comparison of mAP results on the Posetrack 2018 validation set.

	mAP			**AP^T**
Method	**Wri.**	**Ank.**	**Total**	**Total**
STAF [18]	64.7	62.0	-	70.4
Alphapose-UNI [11]	-	-	-	74.0
Keytrack [14]	79.2	76.5	81.6	74.3
MDPN [52]	74.1	69.9	75.0	71.7
Baseline [13]	73.3	70.9	77.2	72.4
ours	75.3	71.7	79.0	74.5

Table 2. Comparison of MOTA results on the Posetrack 2018 validation set.

Method	**Wri.**	**Ank.**	**Total**	**fps**
STAF [18]	-	-	60.9	3
Alphapose-UNI [11]	-	-	64.4	10.9
Keytrack [14]	-	-	66.6	1.0
MDPN [52]	49.0	45.1	50.6	-
Baseline [13]	-	-	64.6	0.7
Ours	59.2	58.3	65.7	0.5

Table 3. Comparison of IDSW results on the Posetrack 2018 validation set.

Method	Wri.	Ank.	Total
Keytrack [14]	0.8	0.8	0.8
Optical Flow [17]	1.1	1.1	1.1
Ours	0.6	0.7	0.7

4.4.2. Qualitative Results

Our qualitative results on the PoseTrack 2018 dataset are visualized in Figures 5 and 6. Figure 5 shows the comparison between our method and LightTrack [13] on the pose estimation results, where (a) represents the pose estimation result of the $t-1$ frame, (b) is the pose estimation result of LightTrack in the t frame, and (c) is the result of our method in the t frame. From Figure 5, it can be observed that in certain occluded scenarios, our method achieves improved pose estimation results through the use of tracking. This is because in general frames, the bounding box is propagated from the previous frame, and when encountering occluded scenes, it is prone to suboptimal pose estimation results, making it difficult to match with the pose from the previous frame. After transitioning to keyframes, utilizing the detected bounding boxes allows for better pose results. Figure 6 shows the visualization of our tracking results. The bounding boxes, poses, and identity IDs are color-coded according to the predicted trajectory IDs by our model. Bounding boxes of different depths of color represent different identities. From the figures, we can see that our method can perform well in the task of multi-person pose estimation and tracking.

(**a**) $t-1$ frame (**b**) LightTrack in t frame (**c**) our method in t frame

Figure 5. The results of comparing our method with LightTrack.

4.5. Ablation Study

4.5.1. Performance of Different Pose Estimators

In this section, we experiment with the adaptability of our tracking method to different pose estimators. Except for MSRA152 [17], we also try training CPN101 [53] and HRNet [23] as human pose estimators. CPN101 adopts the same training method as MSRA152. When training the HRNet model, we first train the HRNet model with 300 epochs on the COCO dataset, then we fine-tune it for 40 epochs on the PoseTrack 2018 training set with the learning rate reduced to 1×10^{-5} and 1×10^{-6} at the 15th and 30th epochs, respectively. Although CPN101 and HRNet also perform well as a human pose estimator, their adaptability is not as good as MSRA152 for our tracking method, so our experimental comparisons are conducted using MSRA152 as the pose estimator. Our estimator experimental comparison results are shown in Table 4.

Figure 6. The visualizations of our method's results.

Table 4. Ablation study of different pose estimators in pose estimation.

	Estimation (AP^T)			Tracking (MOTA)		
Pose Estimator	**Wri.**	**Ank.**	**Total**	**Wri.**	**Ank.**	**Total**
CPN101	67.6	65.3	71.8	56.7	53.3	62.2
HRNet	72.4	66.1	74.7	59.2	54.7	64.4
MSRA152	68.9	67.3	74.5	59.2	58.3	65.7

4.5.2. Performance of Different Modules

Here, we evaluate different components of our method and quantify the impact of different components on the overall performance through ablation experiments on the PoseTrack 2018 validation set. The results in Table 5 show that the BTPM module significantly improves the MOTA results. This improvement is attributed to the capability of BTPM to increase pose matching compared to Spatial Consistency (SC) and overcome the issue of large pose variations for the same individual between consecutive frames. Furthermore, the complete model with the AAMF module (CPM + SGCN) can further improve the performance of MOTA, with a total improvement of 1.4 in MOTA. Among them, CPM denotes the Center Point Matching module. The improvement is more significant for the CPM method, which increases by 1.1. This is because the center point matching on the long temporal sequence overcomes the problem of identity loss caused by occlusion. SC is compared using IoU in our experiment.

Table 5. Ablation study on the MOTA results of different modules in the PoseTrack 2018 validation set.

Method	**Wri.**	**Ank.**	**Total**
SC	23.2	22.5	26.8
SC + BTPM	58.2	56.6	64.3
SC + BTPM + CPM	59.2	57.7	65.4
SC + BTPM + CPM + SGCN	59.2	58.3	65.7

4.5.3. Performance of Different Step Sizes between Keyframes

In this section, we evaluate the impact of the step size between keyframes on the results. In our experiments, we first select keyframes based on a fixed step size, and then set the other frames as general frames. We set the step size between two keyframes as a variable x and conduct experiments with different x values. The experimental results on

AP^T and MOTA evaluation metrics are shown in Figures 7 and 8. From Figure 7, we can see that the mAP with filtering has a slight improvement when the step size increases from 2 to 3. We speculate that this is because there are more general frames, and the FTPM module based on the generated bounding boxes from keypoints can perform better. As the step size continues to increase, there are more general frames, and the positional offset of the same pose between two adjacent frames will gradually increase. This will lead to a decrease in the accuracy of the generated bounding boxes from the previous frame's pose, which ultimately leads to a decrease in AP^T. From the results of the keypoint tracking in Figure 8, as the step size of the frame increases, the result of MOTA gradually decreases; therefore, frequent keyframes can help improve the performance of MOTA. Finally, to better evaluate our tracking model, we prefer to achieve better tracking results and set the value of x to 2.

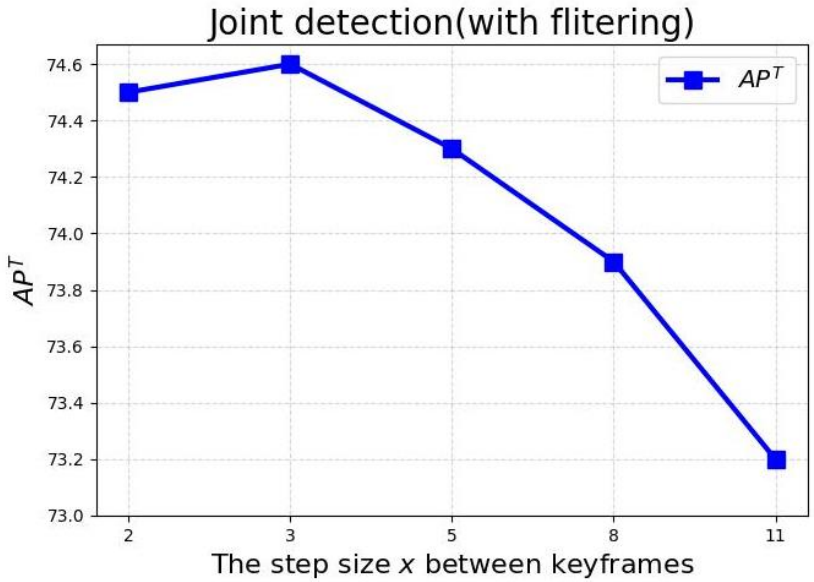


Figure 7. Results of different step sizes x in joint detection (with filtering).

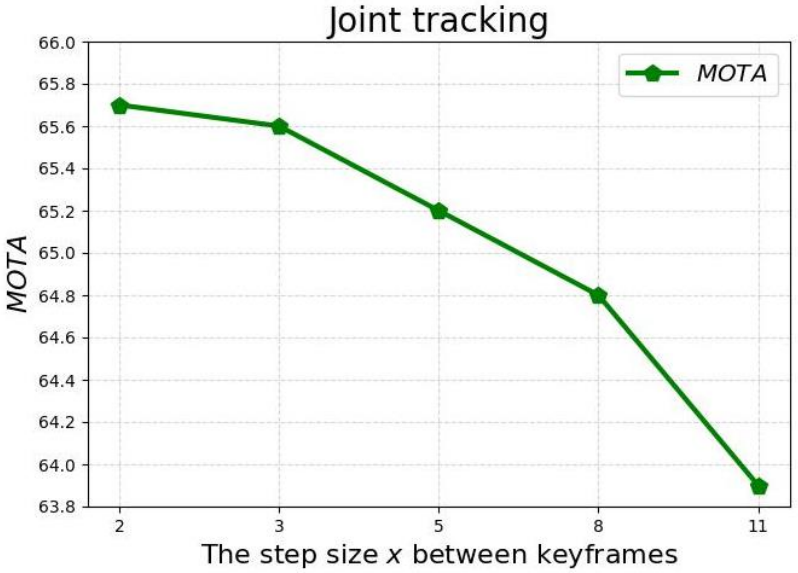


Figure 8. Results of different step sizes x in joint tracking.

4.5.4. The Setting of Threshold and Coefficient Values

In this chapter, we evaluate the setting of different thresholds and coefficients in various modules. Figure 9 illustrates the threshold settings for BTPM module, FTPM module, and center point matching in the AAMF module. Here, the blue dashed line

represents the results of using different thresholds for pose matching with OKS in the FTPM module. The red and green dashed line respectively depict the results of using different thresholds for OKS and IoU in the BTPM module. The yellow dashed line represents the parameter setting for center point matching in the AAMF module.

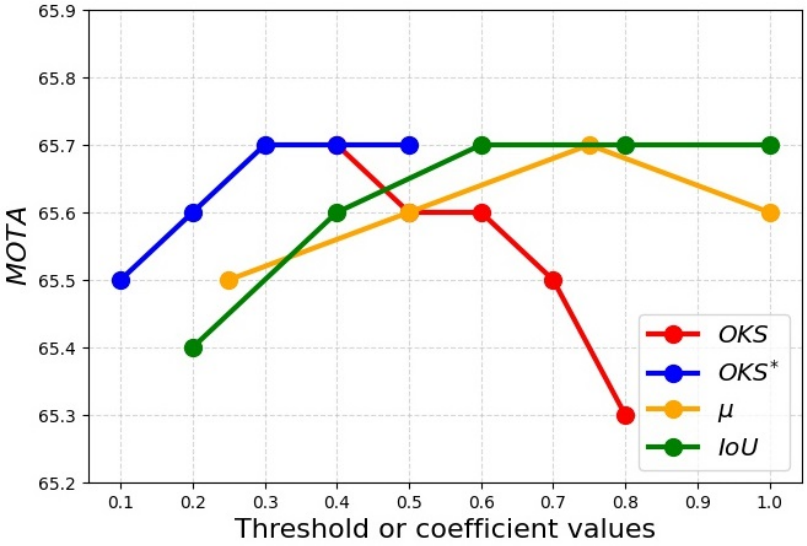


Figure 9. Results of different threshold or coefficient values. $*$ represents OKS in FTPM.

From Figure 9, we can observe that in the BTPM module, the accuracy of the tracking gradually increases with the increase in the threshold when using IoU for pose matching and eventually stabilizes. This is because when the threshold is too low, obtaining a matching result solely based on IoU is possible without the involvement of other modules, resulting in relatively low accuracy. As there is an optional range for the highest value, in our experiments, we chose the median, setting it to 0.8. Contrary to the IoU line, when the threshold for OKS in the BTPM module is high, the tracking accuracy decreases. This is because the same person in consecutive frames may have a considerable displacement, even if the optimal matching value is not necessarily greater than the threshold, resulting in a missed optimal match and a decrease in accuracy. Therefore, we set the threshold for OKS in the BTPM module to 0.4. From the blue line chart, we can observe that in the FTPM module, when the OKS threshold is low, the tracking accuracy is correspondingly low. With the increase in the threshold, the accuracy also increases, but it becomes constant after 0.3. This is because when the threshold is low, even some incorrect matching results are considered successful by the model, increasing the number of incorrect matches and consequently reducing the tracking accuracy. Therefore, in our experiments, the OKS threshold in the FTPM module is set to 0.3. In the first step of the AAMF module, the threshold setting of the center point-matching module follows the following formula:

$$\tau = \lambda \times \min(w, h). \tag{9}$$

Here , τ represents the threshold, w and h denote the width and height of the bounding box of a specific pose to be matched, and λ represents the coefficient. Its value is related to the model's results as indicated by the yellow curve in Figure 9. From the graph, it can be observed that as the coefficient λ increases, the tracking accuracy also increases, reaching a maximum at 0.75. Similar to OKS, when the coefficient is small, there might be erroneous matching results, and when it is large, there might be missed correct matching results. Therefore, in our experiments, we set the coefficient λ to 0.75.

In the AAMF module, different values of the number of frames n to be traced back are used for different calculation modules. For the center point module, when n is too small, it may not cover the frame before the person disappears, while when n is too

large, the temporal information becomes redundant. Therefore, we set n according to the following formula:

$$n = \max(2 + \gamma d), \tag{10}$$

where γ = [0, 1, 2, 3], and the values of d are illustrated in Figure 10. The red line represents the relationship between the tracking results and the values of d in the center point module, while the blue line represents the pose feature module. Figure 10 indicates that in the center point module, as the value of d increases, the MOTA value also increases, benefiting from the inclusion of long-term temporal information for tracking. However, as d continues to increase, the MOTA value decreases. This is because, at larger distances, poses may not provide reliable temporal information. On the contrary, it might lead to some erroneous matches. Therefore, we set d to 3, with the corresponding n being 11. As shown in the blue line chart, in the pose feature module, as the number of frames increases, the magnitude of the pose variations also increases, leading to a decrease in the performance of pose feature matching. Consequently, the MOTA value decreases with the increase in d. Therefore, we set d to 1, with the corresponding n being 5. As shown by the yellow curve, we conducted a comparative experiment on the number of γ, and the results indicate that as γ increases, MOTA gradually increases and reaches its maximum when γ is greater than or equal to 4, remaining constant thereafter. This is because when γ is too small, the center point module cannot provide sufficient long-term information. Therefore, in our experiments, we set the quantity of γ to 4.

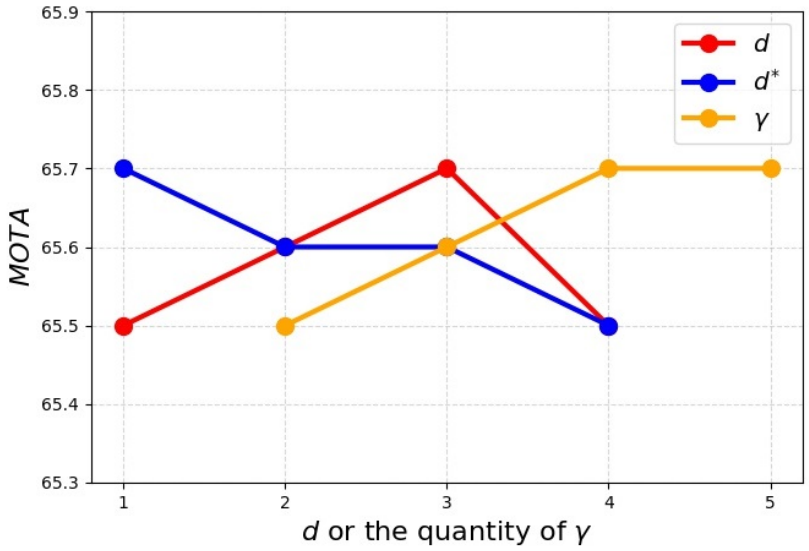


Figure 10. Results of different values for d, the quantity of γ. $*$ represents d in the pose feature module.

5. Conclusions

In this paper, we propose a new top–down approach for human pose tracking in videos. In our method, through propagating pose information bidirectionally and matching, we conduct temporal association. Additionally, we utilize a Re-id module that takes advantage of the long temporal relationship to supplement missing tracking information in previous frames due to occlusions, thus improving the results of multi-person pose estimation and tracking. Our method has good generalizability because it does not depend on the choice of the pose estimator. We outperform most tracking methods in terms of keypoint estimation and tracking. Finally, we demonstrate the accuracy of our method by presenting the visualization results on the PoseTrack 2018 dataset.

Our method still exhibits certain limitations in terms of real-time performance. Due to certain computational complexities or design flaws in the model, our approach may demonstrate a relatively slow response when dealing with real-time data streams, rendering the model inadequate for applications with stringent real-time requirements. Therefore,

in future work, we aim to further optimize our tracking model, reduce computational complexities, and design a pose estimator more suitable for the current tracking task to enhance both the real-time performance and tracking accuracy.

Author Contributions: Methodology, Q.S. and Y.F.; software, Y.F.; validation, Y.F., Z.Y. and Q.S.; formal analysis, Y.F.; investigation, Y.F., Z.Y. and Q.S.; writing—original draft preparation, Y.F.; writing—review and editing, Y.F., Z.Y. and Q.S.; visualization, Y.F. and Z.Y.; supervision, Q.S.; funding acquisition, Q.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by The Natural Science Foundation of Hebei Province grant number F2019201451.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The derived data supporting the findings of this study are available from the corresponding author on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wei, S.E.; Ramakrishna, V.; Kanade, T.; Sheikh, Y. Convolutional pose machines. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition , Las Vegas, NV, USA, 27–30 June 2016; pp. 4724–4732.
2. Newell, A.; Yang, K.; Deng, J. Stacked hourglass networks for human pose estimation. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Part VIII 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 483–499.
3. Yang, W.; Li, S.; Ouyang, W.; Li, H.; Wang, X. Learning feature pyramids for human pose estimation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1281–1290.
4. Ke, L.; Chang, M.C.; Qi, H.; Lyu, S. Multi-scale structure-aware network for human pose estimation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 713–728.
5. Andriluka, M.; Pishchulin, L.; Gehler, P.; Schiele, B. 2d human pose estimation: New benchmark and state of the art analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 3686–3693.
6. Johnson, S.; Everingham, M. Clustered pose and nonlinear appearance models for human pose estimation. In Proceedings of the BMVC, Aberystwyth, UK, 31 August–3 September 2010; Volume 2, p. 5.
7. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; Part V 13, pp. 740–755.
8. Andriluka, M.; Roth, S.; Schiele, B. Monocular 3d pose estimation and tracking by detection. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, San Francisco, CA, 13–18 June 2010; pp. 623–630.
9. Pishchulin, L.; Andriluka, M.; Gehler, P.; Schiele, B. Poselet conditioned pictorial structures. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 588–595.
10. Zhang, W.; Zhu, M.; Derpanis, K.G. From actemes to action: A strongly-supervised representation for detailed action understanding. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2248–2255.
11. Fang, H.S.; Li, J.; Tang, H.; Xu, C.; Zhu, H.; Xiu, Y.; Li, Y.L.; Lu, C. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 7157–7173. [CrossRef] [PubMed]
12. Buizza, C.; Fischer, T.; Demiris, Y. Real-time multi-person pose tracking using data assimilation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 449–458.
13. Ning, G.; Pei, J.; Huang, H. Lighttrack: A generic framework for online top-down human pose tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 13–19 June 2020; pp. 1034–1035.
14. Snower, M.; Kadav, A.; Lai, F.; Graf, H.P. 15 keypoints is all you need. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6738–6748.
15. Iqbal, U.; Milan, A.; Gall, J. Posetrack: Joint multi-person pose estimation and tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2011–2020.
16. Andriluka, M.; Iqbal, U.; Insafutdinov, E.; Pishchulin, L.; Milan, A.; Gall, J.; Schiele, B. Posetrack: A benchmark for human pose estimation and tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5167–5176.

17. Xiao, B.; Wu, H.; Wei, Y. Simple baselines for human pose estimation and tracking. In Proceedings of the European conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 466–481.
18. Raaj, Y.; Idrees, H.; Hidalgo, G.; Sheikh, Y. Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4620–4628.
19. Jin, S.; Ma, X.; Han, Z.; Wu, Y.; Yang, W.; Liu, W.; Qian, C.; Ouyang, W. Towards Multi-Person Pose Tracking: Bottom-Up and Top-Down Methods. 2017. Available online: https://jin-s13.github.io/papers/BUTD.pdf (accessed on 1 January 2024)
20. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
21. Li, Z.; Xue, M.; Cui, Y.; Liu, B.; Fu, R.; Chen, H.; Ju, F. Lightweight 2D Human Pose Estimation Based on Joint Channel Coordinate Attention Mechanism. *Electronics* **2023**, *13*, 143. [CrossRef]
22. Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N.; Shah, M. Deep learning-based human pose estimation: A survey. *ACM Comput. Surv.* **2023**, *56*, 1–37. [CrossRef]
23. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.
24. Huang, J.; Zhu, Z.; Guo, F.; Huang, G. The devil is in the details: Delving into unbiased data processing for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5700–5709.
25. Yang, S.; Quan, Z.; Nie, M.; Yang, W. Transpose: Keypoint localization via transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 11802–11812.
26. Zhou, M.; Stoffl, L.; Mathis, M.; Mathis, A. Rethinking pose estimation in crowds: Overcoming the detection information-bottleneck and ambiguity. *arXiv* **2023**, arXiv:2306.07879.
27. Li, Y.; Zhang, S.; Wang, Z.; Yang, S.; Yang, W.; Xia, S.T.; Zhou, E. Tokenpose: Learning keypoint tokens for human pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 11313–11322.
28. Yuan, Y.; Fu, R.; Huang, L.; Lin, W.; Zhang, C.; Chen, X.; Wang, J. Hrformer: High-resolution transformer for dense prediction. *arXiv* **2021**, arXiv:2110.09408.
29. Xu, Y.; Zhang, J.; Zhang, Q.; Tao, D. Vitpose: Simple vision transformer baselines for human pose estimation. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 38571–38584.
30. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1440–1448. . [CrossRef] [PubMed]
31. Chen, K.; Pang, J.; Wang, J.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Shi, J.; Ouyang, W.; et al. Hybrid task cascade for instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4974–4983.
32. Zhu, X.; Hu, H.; Lin, S.; Dai, J. Deformable convnets v2: More deformable, better results. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9308–9316.
33. Cheng, B.; Xiao, B.; Wang, J.; Shi, H.; Huang, T.S.; Zhang, L. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5386–5395.
34. Cheng, Y.; Ai, Y.; Wang, B.; Wang, X.; Tan, R.T. Bottom-up 2D pose estimation via dual anatomical centers for small-scale persons. *Pattern Recognit.* **2023**, *139*, 109403. [CrossRef]
35. Qu, H.; Cai, Y.; Foo, L.G.; Kumar, A.; Liu, J. A Characteristic Function-Based Method for Bottom-Up Human Pose Estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 13009–13018.
36. Jin, S.; Liu, W.; Xie, E.; Wang, W.; Qian, C.; Ouyang, W.; Luo, P. Differentiable hierarchical graph grouping for multi-person pose estimation. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 718–734.
37. Li, J.; Wang, Y.; Zhang, S. PolarPose: Single-stage multi-person pose estimation in polar coordinates. *IEEE Trans. Image Process.* **2023**, *32*, 1108–1119. [CrossRef] [PubMed]
38. Geng, Z.; Sun, K.; Xiao, B.; Zhang, Z.; Wang, J. Bottom-up human pose estimation via disentangled keypoint regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14676–14686.
39. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
40. Jin, S.; Liu, W.; Ouyang, W.; Qian, C. Multi-person articulated tracking with spatial and temporal embeddings. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5664–5673.

41. Newell, A.; Huang, Z.; Deng, J. Associative Embedding: End-to-End Learning for Joint Detection and Grouping. *Adv. Neural Inf. Process. Syst.* **2017**. Available online: https://patrick-llgc.github.io/Learning-Deep-Learning/paper_notes/associative_embedding.html (accessed on 1 January 2024)
42. Girdhar, R.; Gkioxari, G.; Torresani, L.; Paluri, M.; Tran, D. Detect-and-track: Efficient pose estimation in videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 350–359.
43. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
44. Algabri, R.; Choi, M.T. Online Boosting-Based Target Identification among Similar Appearance for Person-Following Robots. *Sensors* **2022**, *22*, 8422. [CrossRef] [PubMed]
45. Yan, S.; Xiong, Y.; Lin, D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
46. Ruggero Ronchi, M.; Perona, P. Benchmarking and error diagnosis in multi-instance pose estimation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 369–378.
47. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
48. Hoiem, D.; Divvala, S.K.; Hays, J.H. Pascal VOC 2008 Challenge. 2009. Available online: https://www.researchgate.net/publication/228388312_Pascal_VOC_2008_Challenge (accessed on 1 January 2024)
49. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. *arXiv* **2016**, arXiv:1603.00831.
50. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
51. Yu, D.; Su, K.; Sun, J.; Wang, C. Multi-person pose estimation for pose tracking with enhanced cascaded pyramid network. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
52. Guo, H.; Tang, T.; Luo, G.; Chen, R.; Lu, Y.; Wen, L. Multi-domain pose network for multi-person pose estimation and tracking. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
53. Chen, Y.; Wang, Z.; Peng, Y.; Zhang, Z.; Yu, G.; Sun, J. Cascaded pyramid network for multi-person pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7103–7112.