Robot Person Following Under Partial Occlusion

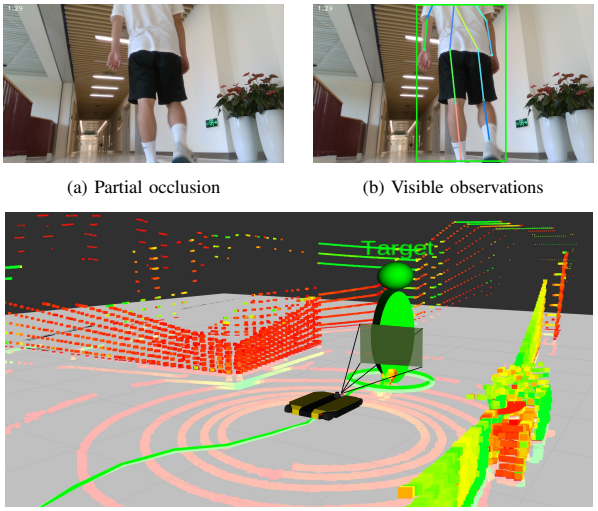
Hanjing Ye[1], Jieting Zhao[1], Yaling Pan[2], Weinan Chen[1], Li He[1] and Hong Zhang[1*]

***Abstract*— Robot person following (RPF) is a capability that supports many useful human-robot-interaction (HRI) applications of a mobile robot. However, existing solutions to person following often assume a full observation of the tracked person. As a consequence, they cannot track the person reliably under partial occlusion where the assumption of full observation is not satisfied. In this paper, we focus on the problem of robot person following under partial occlusion caused by a limited field of view of a monocular camera. Based on the key insight that it is possible to locate the target person when one or more of his/her joints are visible, we propose a method in which each visible joint contributes a location estimate of the followed person. Experiments show that, even under partial occlusion, the proposed method can still locate the person more reliably than the existing methods. In combination with this person location module, our RPF system achieves SOTA results in a public person following dataset. As well, the application of our method is demonstrated in real experiments on a mobile robot.**

I. INTRODUCTION

Robot person following (RPF) [1] is a capability that supports many useful HRI [2] applications. Typically, the person being followed can become partially occluded in various situations due to, for example, other objects or people in the robot working environment. Therefore, the ability of following a person under partial occlusion is essential. RPF can be achieved with a distance measurement sensor such as LiDAR and RGB-D camera. Methods [3]–[8], for example, firstly track multiple people with the help of a distance measurement sensor. Once a target person is selected in the field of view of the robot sensor, they will follow the person based on the tracked location. Such solutions, however, can be expensive due to the high cost of a distance measurement sensor. In addition, distance sensors lack textural information, and this prevents them from resolving data association effectively.

Alternatively, one can resort to vision to deploy a low-cost mobile robot for person following. [9] uses a vision-based single object tracker (SOT) [10]–[12] to track the person in the image space, and relies on visual servo to follow the person. To track the person in the robot space for robust control, [13] proposes a method under the assumption that the neck of the person is always visible. Effectively, it can be characterized as a *single-joint-based* method. However, with a camera of limited field-of-view (FoV), neck visibility cannot always be guaranteed even though the followed person may still be partially visible, as shown in the example in Fig. 1(a). In addition, many *deep-learning-based* methods can be used for person location estimate, such as monocular depth prediction [14], monocular 3D bounding-box detection [15], etc. These methods, however, are known to experience poor generalization in terms of adapting to the case of partial person observation.



(a) Partial occlusion (b) Visible observations

(c) Target person location and following

Fig. 1. (a) An example of partial occlusion caused by a limited FoV of a camera while performing person following. Existing methods often fail to locate the person in such challenging scenario. (b) The proposed method is based on these visible observations, including the bounding box and 2D joints. (c) Person following under partial occlusion (the pointcloud is used for visualization only).

Related to our study is the topic of 2D pose estimation [16] [17], which has been well developed in recent years. It has been demonstrated that human joints can be detected reliably even under partial observation as shown in Fig. 1(b). We can therefore exploit these well-detected joints of a partially occluded person in his location estimation. This idea of estimating an object or a person from partial observation is not new in 3D object or human pose estimation [18]–[20]. To design a solution, one can first build a prior global model describing the structural constraints between the parts of the whole object/person, and this model can be created in the form of an implicit representation by a neural network [18], a parametric model like SMPL (skinned multi-person linear model) [19], or a CAD model [20]. Then given a partial

*corresponding author (hzhang@sustech.edu.cn).

[1]Hanjing Ye, Jieting Zhao, Li He and Hong Zhang are with Shenzhen Key Laboratory of Robotics and Computer Vision, Southern University of Science and Technology (SUSTech), and the Department of Electronic and Electrical Engineering, SUSTech.

[2]Yaling Pan and Weinan Chen are with the Biomimetic and Intelligent Robotics Lab, Guangdong University of Technology.

This work was supported by the Pearl River Talent Recruitment Program under Grant No.2019QN01X761 and the National Natural Science Foundation of China (62103179).

Source Code: https://github.com/MedlarTea/Mono-RPF.

observation, the 3D poses can be inferred from the prior global model.

To extend the above idea to our study of RPF, without loss of generality, we build a prior model consisting of the heights of a person's joints relative to the ground plane in the Cartesian space. With the assumption of a standing person and the ground plane calibration, the person's location in the Cartesian space can be inferred from a reprojection of any joint observation in the image plane given the known height of the joint from the prior model [21]. In our method, which we refer to as *visible-joints-based* method, each visible joint contributes a location estimate. Furthermore, to mitigate noise in joint measurements, we propose to perform data association in the bounding-box-like measurement space. An identification module and a robot controller are also implemented to form a complete RPF system. In the end, our system can follow a target person reliably even under partial occlusion, as shown in Fig. 1(c).

In summary, our contribution is a simple yet effective *visible-joints-based* RPF method that can locate the person under partial occlusion. It tracks a person by his visible joints and performs data association in the bounding-box-like measurement space. We conduct extensive experiments to show that the proposed method can locate the person under partial occlusion where existing methods often fail. The complete RPF system achieves SOTA results in a public person following dataset.

II. RELATED WORK

A. Monocular Person Tracking in Robot Person Following

A person following robot usually involves three modules: a tracking module, an identification module and a robot controller. The tracking module locates the person assuming smooth motion. The identification module usually maintains an appearance model useful for re-identifying the lost target person. Given the person's estimated location, the robot controller computes a motion command for the robot to maintain a desired relative location with the tracked person. Obviously, the accuracy of the person location estimate by the tracking module affects the performance of the robot controller.

Many existing works in people tracking [3]–[8] use distance measurement sensors, which can be expensive and have difficulty in dealing with cluttered indoor environments due to the lack of textural information. Some works are based on monocular vision. [9] tracks a person in the image space through SOT techniques, and realizes person following with visual servo by keeping the person in the center of the image view. Image-space-based tracking is convenient to implement but it is not effective compared to the position-based tracking as the person and the robot move physically in the Cartesian space, not the image space [22].

For estimating a person's location in the robot coordinate frame, inspired by well-known techniques in video surveillance [21] [22], [13] proposes a *single-joint-based* method to locate the person by assuming that the neck of the person is always visible. However, this assumption cannot always be satisfied even though the person may still be partially visible. Given the fact that some joints can still be observed under partial occlusion, we propose to use visible joints to track the person instead of a specific joint to alleviate the problem caused by partial occlusion.

B. Person Location Estimation

A RPF system must contain a module for person location estimation, and the current leading methods in solving this problem are mostly deep-learning-based. These methods employ a deep neural network to infer the location of a person from the observed image in an end-to-end fashion. MonoLoc [23] uses a neural network to detect 2D joints of a person first, and then locates the person by a multi-task-trained neural network whose input is the set of 2D joint locations. EPro-PnP [24] describes the pose of a person in the form of a 3D bounding box by integrating learnable 2D-3D correspondences. RootNet [25] develops a top-down pose estimation solution that computes the 3D poses including their joints of multiple people with respect to the camera coordinate frame.

All the methods mentioned above are entirely based on deep learning, although MonoLoc [23] consists of two modules, one for joint detection and the other person location, in a way that is similar to our solution. In addition, their training datasets usually involve a full observation of the objects of interest. However, in the person following scenario, a partial observation of the person often occurs, which these methods cannot deal with effectively. In our work, we adopt a hybrid approach where a 2D learning-based pose detector [17] provides joint measurements to a subsequent model-based person location estimator in our tracking module. The main advantages of this approach are that: 1) 2D pose detectors, compared to the *deep-learning-based* methods, are shown to be more robust with respect to partial occlusion and 2) our proposed model-based location estimator is able to utilize these well-detected 2D joints to infer the person's location.

C. 2D Human Pose Detection

Related to our work is the topic of 2D human pose estimation [26], which has been well developed in recent years. It aims at localizing human joints (keypoints) in an image. There are two paradigms in this topic: top-down [17] [27] and bottom-up [28] [29]. A top-down method first detects bounding boxes, then localizes 2D joints within them, whereas a bottom-up method detects joints first and then groups them into a full body. We adopt a top-down method (AlphaPose [17]) in the tracking module of our *visible-joints-based* RPF system, due to its good performance even under partial occlusion.

III. METHODOLOGY

We introduce a *visible-joints-based* method by utilizing visible joints to track the target person instead of a specific joint, which is our main contribution. Together with an identification module and a robot controller, a complete RPF

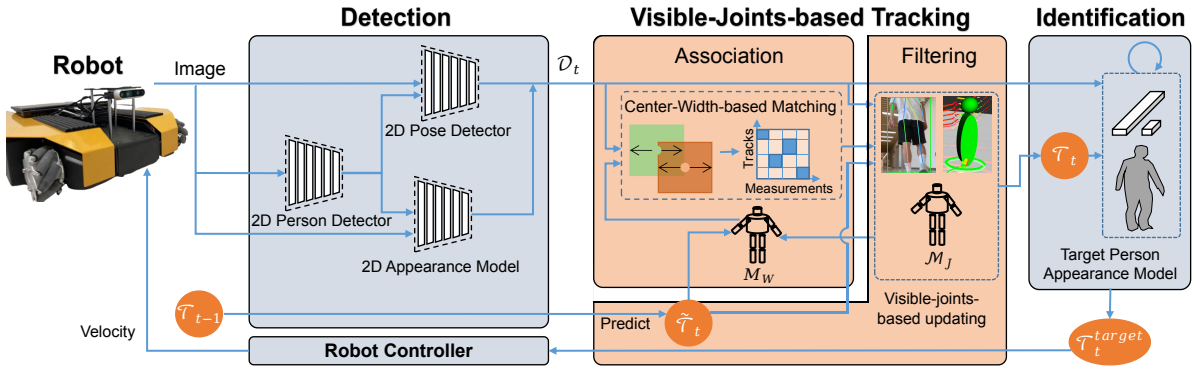


Fig. 2. Our proposed *visible-joints-based* RPF system is composed of a Detection module(Sec. III-A) including 2D person detector, 2D pose detector and appearance extraction model, a Tracking module with track initialization (Sec. III-B), filtering (Sec. III-C), and data association (Sec. III-D), a Identification module and a Robot Controller. $\mathcal{D}$,$\mathcal{T}$,$\mathcal{T}^{target}$ represents detections, tracks and target person track respectively. $\mathcal{M}_J$ is the prior global model consisting of heights of joints and M_W represents the person's real width. This system can locate, track, and follow the target person even under partial occlusion.

system is formed. A general overview of our system is shown in Fig. 2.

We follow the idea of previous works [18]–[20]. First, we build a prior global model, and then infer the person's location from visible joints. Given detection results (Sec. III-A) in each frame, our method follows the general process of the Kalman filter. In track initialization (Sec. III-B), with detections of the first frame, we first build a prior model $\mathcal{M}_J$ consisting of heights of joints, which are critical for locating the person by any visible joint measurements. Then in the following frames, a person's location is estimated by this prior model with any visible joints. In the filtering stage (Sec. III-C), tracks are updated with associated joint measurements where each visible joint contributes a location estimate. In the data association (Sec. III-D), we associate tracks and joint measurements in the bounding-box-like space due to the stability of the bounding-box detection. In addition, the identification module is used to identify our target person. Lastly, with a robot controller, person following is performed in the robot frame based on the target person's location.

As in existing works [13] [22], the robot frame is set on the ground plane. The calibrated camera only has a tilt angle with zero roll and pan angle relative to the robot frame.

A. Detection

First, we detect people and output their bounding boxes (by YOLOX [30]), 2D poses and appearance features of which the first two results are used for tracking and the last one for the target person's identification. Specifically, we use AlphaPose [17] for 2D pose estimation due to its good performance even under partial occlusion. We exploit these well-detected joints of a partially occluded person in his location estimate.

In each frame, only the centers of shoulder, hip, knees and ankles are used as our observations, which are approximated by a line that is vertical to the ground. Together with bounding box and appearance features, in the image plane, our detection result of a person is represented as: $\mathcal{D} = \{\mathcal{B}, \mathcal{P}, \mathcal{F}\}$, where $\mathcal{B} = \{[u, v], w, h\}$ defines the center point, the width and the height of the bounding box, $\mathcal{P}$ is the visible joint measurements or a subset of $\{\mathbf{p}_{neck}, \mathbf{p}_{hip}, \mathbf{p}_{knee}, \mathbf{p}_{ankle}\}$ where $\mathbf{p}_{joint} \in \mathbb{R}^2$, and $\mathcal{F}$ is the appearance features.

Algorithm 1: Location with A Visible Joint

Input: Camera intrinsic $\mathbf{K}$; in the camera frame, ground plane with normal $\mathbf{N}$ at a distance γ with respect to the optical center ; a joint measurement $\mathbf{p} = [u, v]$ in the image plane and corresponding prior height h where the height of ankle is 0.

Output: Person's location $\mathbf{X}$ in the camera frame.

Let $\bar{\mathbf{p}} = [u, v, 1]^\top$
$\mathbf{r} = \mathbf{K}\bar{\mathbf{p}}$ // ray from the optical center to the joint
$\rho = \mathbf{N}^\top \mathbf{r}$ // projection of ray on the normal
$\mathbf{X} = \frac{|\gamma - h|}{|\rho|} \cdot \mathbf{r}$ // obtain the scale of the ray
Return: $\mathbf{X}$

B. Track Initialization

In order to initialize the person's location with visible joints, we first build a joint-height-based prior model from the first image frame. With the assumption of a standing person, the model is represented in the Cartesian space as:

$$\mathcal{M}_J = \{h_{neck}, h_{hip}, h_{knee}\}, \tag{1}$$

which denotes the height of the person's neck, hip and knees relative to the ankle, which is assumed to be on the ground.

In the camera frame, the ground plane is already calibrated with a known unit normal $\mathbf{N} \in \mathbb{R}^3$ at a distance $\gamma > 0$ with respect to the optical center. The person's location is represented by the location of his ankle $\mathbf{X} \in \mathbb{R}^3$ with $\mathbf{N}^\top \mathbf{X} + \gamma = 0$ [31]. Then other joints locations can be approximated as $\mathbf{X}_j \approx \mathbf{X} + h_j \cdot \mathbf{N}$ [32] where $h_j \in \mathcal{M}_J$. We assume a full body can be observed in the first frame, while in the following frames, our method only needs to observe one of the four joints. So with a full-body observation $\mathcal{P}_{all} = \{\mathbf{p}_{neck}, \mathbf{p}_{hip}, \mathbf{p}_{knee}, \mathbf{p}_{ankle}\}$ in the image plane and corresponding variable $\mathcal{X}_{all} = \{\mathbf{X}_{neck}, \mathbf{X}_{hip}, \mathbf{X}_{knee}, \mathbf{X}\}$ in the camera frame, we initialize our prior model $\mathcal{M}_J$ by the

minimization of the reprojection error defined by :

$$\mathcal{X}^*_{all} = \underset{\mathcal{X}_{all}}{\operatorname{argmin}} \sum_i ||\mathbf{p}_i - g(\mathbf{X}_i)||^2_2, \tag{2}$$

where $g(\mathbf{X}_i)$ denotes the 2D projection function from the camera frame to the image plane, i.e., $g : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. After minimization, the person's location is initialized and the prior model $\mathcal{M}_J$ is obtained.

With $\mathcal{M}_J$, the person's location can be estimated with any visible joints using a Kalman filter. To begin, the person's location is initialized as in Algorithm 1, which is based on the well-known technique [21]. Empirically, we prefer to initialize the person's location by neck, hip, knee and ankle, with the importance listed in a descending order due to their different levels of measurement noise. Experimentally, when using AlphaPose [17], we observe that: 1) the upper body is more stable than the lower body for a walking person [33]; 2) the shoulder is more distinguishable than the hip because the shoulder is near the background; and 3) the movement range of the knee is smaller than the movement range of the ankle. The location estimate is updated (introduced in the following subsection) with all visible joints for n_{init} consecutive frames to obtain a stable and accurate person track. Further, this person track would be deleted if it has not been updated for n_{delete} consecutive frames.

C. Filtering

Here, we adopt the unscented Kalman filter (UKF) as our filtering framework for its advantages in dealing with non-Gaussian noise and non-linear observation function. In our UKF-based filtering, the state to be estimated only includes the person's location and velocity on the ground plane, which are enough to satisfy the need for person following. Specifically, the state is defined as: $\mathbf{s}_t = [x_t, y_t, v_{x,t}, v_{y,t}]^\top$. Here, a constant velocity motion model is utilized for the prediction. So, our predicted state is:

$$\mathbf{s}_{t+1} = \mathbf{s}_t + \Delta t \cdot [v_{x,t}, v_{y,t}, 0, 0]^\top, \tag{3}$$

where Δt is the time interval between two consecutive frames. Then we update the filter state with all visible joints so that each visible joint can contribute a location estimate. In other words, the update step of the Kalman filter depends on the joints that are visible. As defined in Sec. III-A, in the image plane, visible joint measurements $\mathcal{P}_t$ is a subset of $\{\mathbf{p}_{neck,t}, \mathbf{p}_{hip,t}, \mathbf{p}_{knee,t}, \mathbf{p}_{ankle,t}\}$. Let $\mathcal{X}_{vis,t}$ be the corresponding visible joint locations in the camera frame. Then in the image plane, our observation model used by the UKF is defined with the measurements of visible joints:

$$\mathbf{z}_t = [g(\mathbf{X}_{j,t})]^\top, \mathbf{X}_{j,t} \in \mathcal{X}_{vis,t}. \tag{4}$$

D. Data Association

Compared to the bounding-box information $\mathcal{B}$ in the detection $\mathcal{D}$, 2D joints are noisy due to occlusion and motion blur, leading to incorrect data association. Therefore, we build a width model M_W as the person's real width to project the person's location $\mathbf{X}$ to the measurement $[u, w]^T$, consisting of the horizontal component of the bounding-box center and the width of the bounding box for data association. Specifically, with the cylinder assumption of a person's shape, we have the person's width as:

$$M_W = \mathbf{X}_z \cdot w / f_x, \tag{5}$$

where w is the width of the bounding box and f_x is the focal length of the camera. Then we can get the expected observation as:

$$\bar{u} = g(\mathbf{X})|_x, \quad \bar{w} = f_x \cdot M_W / \mathbf{X}_z, \tag{6}$$

where $g(\mathbf{X})|_x$ represents the horizontal component of the pixel. This bounding-box-like space as $[u, w]^T$ can be considered as a cropped version of IoU (Intersection over Union) where the vertical component and the height of the bounding box are not used for their ambiguities. Our distance metric is defined as:

$$d = ((\bar{u} - u)^2 + (\bar{w} - w)^2)^{1/2}, \tag{7}$$

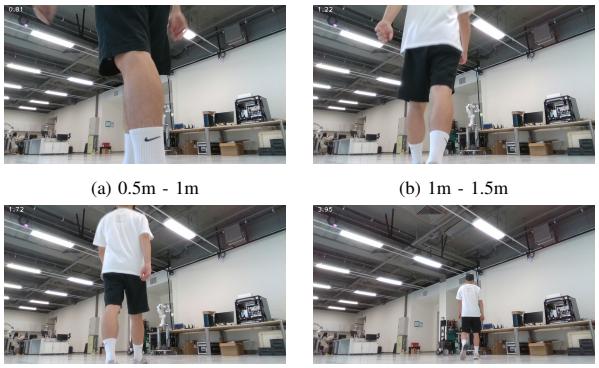
where $\bar{u}, \bar{w}$ are from Eq. 6 and u, w are from the detection set $\mathcal{D}$. Finally a global nearest neighbor method is utilized to match the measurements to the tracks. After updating through Eq. 4, the person's width is updated through Eq. 5 for adapting to the new observation.

IV. EXPERIMENTS

To verify the performance of our proposed *visible-joints-based* method and the whole RPF system, we conduct experiments on different datasets. In this section, we first introduce the datasets, the baselines and implementation details of our experiments. Secondly, the effectiveness of our *visible-joints-based* method is demonstrated by a comparison with the *single-joint-based* and *deep-learning-based* methods on a custom-built dataset. Lastly, we show the superiority of our RPF system on a public person following dataset.

A. Datasets

In this work, two datasets are used in the experiments, named as monocular person location dataset (MPLD) and monocular person following dataset (MPFD). MPLD is a self-built dataset for the location accuracy evaluation of our proposed *visible-joints-based* method and MPFD [34] is a public person following dataset for the target person tracking ability evaluation of the whole RPF system. MPLD involves sequential frames and location of a partially occluded person. All ground truth locations are provided by a motion capture system or a LiDAR sensor and four sequences are collected. The distance between the person and the robot is between 0.5m - 6.0m, which is appropriate for a service robot. Some examples are shown in Fig. 3 where an occluded body is often observed when the person partially occurs in the FoV. MPFD contains eleven sequences that are captured by a camera with challenging situations involving illumination change, appearance change and occlusion due to people crossing. MPFD is designed for evaluating the performance of target person tracking in the image space, so its ground



(a) 0.5m - 1m (b) 1m - 1.5m

(c) 1.5m - 3m (d) 3m - 6m

Fig. 3. Examples of our self-built dataset for evaluating the location accuracy of existing methods. From (d) to (a), with the distance decreasing, more joints are occluded. Our method can accurately locate the person under such partial occlusion.

truth positions are the target person's bounding boxes in all frames.

B. Baselines

For evaluating the location accuracy of our method, we conduct a comparison with *single-joint-based* and *deep-learning-based* methods whose mechanisms have been introduced in Sec. II-B. All compared methods locate the person in the camera frame. They are named as follows:

- **Single-joint-based** [13] tracks the person only when the full body is observed including his neck.
- **MonoLoco** [23] could locate the person directly by a neural network with the 2D pose of the person as input.
- **Mono3DBox** predicts the 3D bounding box of the person based on EPro-PnP [24] of which the bottom center is utilized as the estimated location.
- **Mono3DPose** performs person location by predicting his 3D root location through RootNet [25].
- **MonoDepth**, based on MiDas [35], first obtains the scene depth and then the distance to the person with the corresponding bounding box and the depth map. Subsequently, the location of the person can be obtained with reprojection.

All estimated locations are transformed to the ground plane and smoothened by the UKF in our evaluation. The main purpose of this comparative experiment is to establish that our solution to pose estimation is superior to these SOTA methods in the case of partial occlusion due mostly to the assumption of full-body observation by these SOTA methods during their training.

To evaluate the target tracking ability of our RPF system, we compare our system with popular MOT (multiple object tracking) and SOT baselines, including QDTrack [36], Bytetrack [37], SiamRPN++ [38] and STARK [39]. For MOT methods, target person initialization is performed by selecting the ID of the desired bounding box. SOT methods select the person by the ground truth 2D bounding box of the person in the first frame.

C. Implementation Details

In all experiments, bounding-box detection model relies on YOLOX [30] and 2D pose estimation on AlphaPose [17]. The feature extraction model used for appearance description is similar to that in DeepSORT [40]. The minimization problem in the track initialization (Eq. 2) is solved by the Levenberg-Marquardt algorithm. In all experiments, we use the following parameter settings: $n_{init} = 2$, $n_{delete} = 10$.

All evaluations are run on a computer with Intel® Core™ i7-10700F CPU @ 2.90GHz and NVIDIA GeForce RTX 2060. For real robot experiments, a Clearpath Dingo-O and a laptop with Intel(R) Core(TM) i5-10200H CPU @ 2.40GHz and NVIDIA GeForce RTX 1650 are used. A Realsense D435i with 1280×720 resolution and 30Hz frequency is mounted on the robot with 32° tilt angle and a height of 0.19m relative to the ground plane.

D. Experimental Results

1) Evaluation of visible-joints-based person location estimate: This experiment is conducted in MPLD, and three metrics are used for evaluation: 1) average location error (ALE) is calculated in the Euclidean space of the ground plane, 2) recall is the ratio of the number of recognized frames to that of all frames, and 3) weighted location error (WLE) is the proportion of ALE to recall, which can evaluate the overall effectiveness of location considering both accuracy and robustness. Results are shown in Table I. We can observe that, compared to other methods, our method achieves the lowest WLE in all sequences with 0.11m, 0.09m, 0.10m and 0.08m respectively.

The *single-joint-based* method achieves a 0.10m ALE and a 64.0% recall in Sequence I where a full body can be observed occasionally. While in other sequences where only a partially occluded body is observed, the *single-joint-based* method always fails to locate the target person as expected. MonoLoco and Mono3DBox also act poorly with low recall and high ALE, especially in Sequence II to IV. Mono3DPose and MonoDepth can recognize almost all frames with about a 100% recall but with a higher ALE compared to our method.

2) Evaluation of our RPF system: The evaluation of our *visible-joints-based* RPF system is conducted independently from considering robot control. To evaluate its performance of target person tracking, we compare it with other baselines in MPFD. These baselines are popular object tracking methods. Accuracy is used as our evaluation metric by considering a recognized target person's bounding box as a true positive if the distance between the centers of its bounding box and the ground truth bounding box is less than 50 pixels. As is shown in Table II, in MOT and SOT baselines, STARK achieves good performance with a 96.5% accuracy. Our method achieves the best accuracy at 97.5%. This indicates that in the person following scenario, our method is able to track and identify the target person as well as the popular object tracking methods and even slightly better. Another observation is that if the location method of our RPF system changes to the *single-joint-based* one, the accuracy would

TABLE I. Comparison of location performance between our method and esisting baselines in MPLD. All sequences are captured within a distance range of 0.5m-2m, except sequence I is within 0.5m-6m. † indicates deep-learning-based methods. **ALE** (m) represents average location error, **Recall** is the ratio of the number of recognized frames and that of all frames, and **WLE** (m) is the proportion of ALE to recall. ✗means the algorithm fails to locate the target person if it cannot recognize the frame or its ALE is larger than 5 meters. Our method achieves the lowest WLE in all sequences with 0.11m, 0.09m, 0.10m and 0.08m respectively.

Methods	**I**			**II**			**III**			**IV**		
	ALE ↓	***Recall*** ↑	***WLE*** ↓	***ALE***	***Recall***	***WLE***	***ALE***	***Recall***	***WLE***	***ALE***	***Recall***	***WLE***
Single-joint-based [13]	**0.10**	0.64	0.17	0.12	0.04	3.28	✗	✗	✗	✗	✗	✗
MonoLoco† [23]	1.07	0.48	2.22	✗	✗	✗	✗	✗	✗	✗	✗	✗
Mono3DBox† [24]	0.25	0.66	0.38	0.59	0.21	2.76	2.10	0.09	22.25	1.59	0.39	4.12
Mono3DPose† [25]	0.36	0.95	0.38	0.51	**1.00**	0.51	0.95	**1.00**	0.95	1.11	**1.00**	1.11
MonoDepth† [35]	0.57	**1.00**	0.57	0.32	**1.00**	0.32	0.36	**1.00**	0.36	0.22	**1.00**	0.22
Ours	0.11	**1.00**	**0.11**	**0.09**	0.98	**0.09**	**0.10**	**1.00**	**0.10**	**0.08**	0.92	**0.08**

TABLE II. Evaluation of target person tracking between our method and other object tracking baselines in MPFD. Our *single-joint-based* RPF system achieves the best performance with a 97.5% accuracy. And the bounding-box-like data association (bbox-like DA) metric is more appropriate for our method, which can help improve 2.1% accuracy compared to the joints-based data association (joints-based DA) metric.

Methods	**Type**	**Accuracy (%)**
QDTrack [36]	MOT	48.0
Bytetrack [37]	MOT	88.6
SiamRPN++ [38]	SOT	93.6
STARK [39]	SOT	96.5
Single-joint w/ bbox-like DA	RPF	92.0
Visible-joints w/ joints-based DA	RPF	95.4
Visible-joints w/ bbox-like DA	RPF	**97.5**

drop to 92.0%. This is because our method can handle the situation of partial occlusion better.

3) Bounding-box-like Data Association vs. Joints-based Data Association: In this experiment, we compare the bounding-box-like data association method with a joints-based one to demonstrate that the bounding-box-like metric is a better data association metric for our *visible-joints-based* method. In previous studies, it is common to associate tracks and measurements in the observation space. Here, our observation is the joint measurements in the image space. When associate tracks and measurements in the observation space constructed by the joint measurements instead of the observation space constructed by the bounding-box-like measurements, the accuracy would drop from 97.5% to 95.4% as shown in Table II. This is because joint measurements are noisy under occlusion and motion blur. This would cause incorrect data association, leading to ID switches or track loss.

E. Discussion

As is shown in Table I, the proposed method achieves the best location performance with a 0.10m ALE, a 98% recall and a 0.10m WLE on average. Compared to the *single-joint-based* method, it can accurately locate the person even under partial occlusion. Such result indicates that: 1) 2D learning-based pose estimator (AlphaPose [17]) can perform well under partial occlusion; and 2) our model-based method is able to utilize these well-detected joints to locate the partially occluded person. Compared to the *deep-learning-based* baselines, our method is also superior because the learning-based ones are sensitive to environmental change for lacking enough supervision of 3D information. On the other hand, our hybrid approach is not dependent on the labeled 3D training set, yet outperforms the baselines. In conclusion, due to the combination of the robust 2D learning-based pose estimator and our model-based location estimator, our *visible-joints-based* method is able to locate the person accurately in the person following scenario, especially under partial occlusion.

From Table II, we can observe that our method achieves the highest accuracy at 97.5% , which is higher than that of the MOT and SOT baselines. Such result indicates that, in person following scenario, our method is reliable for not only locating the person accurately but also tracking the person persistently as well as these well-designed object tracking methods. The accuracy of our RPF system would drop by 5.1% if the location method changes to the *single-joint-based* one. This is because in MPFD, there are many situations of partial occlusion, such as people crossing, occlusion by the corner wall and occlusion due to limited FoV. In above cases, the *single-joint-based* method would lose the person while our method can persistently locate the person for better handling these situations of partial occlusion.

V. CONCLUSION

In this paper, for performing robot person following under partial occlusion, we propose a practical *visible-joints-based* location method to locate the person with the observation of any of his four joints: neck, hip, knee and ankle. This is made possible by estimating the person's location with a prior model of the tracked person and associating tracks and measurements in a bounding-box-like space. Our method can locate the person accurately and persistently even under partial occlusion. Compared to baselines, our RPF system achieves SOTA target person tracking performance across multiple evaluation metrics.

REFERENCES

[1] M. J. Islam, J. Hong, and J. Sattar, "Person-following by autonomous robots: A categorical overview," *The International Journal of Robotics Research*, vol. 38, no. 14, pp. 1581–1618, 2019.

[2] M. A. Goodrich and A. C. Schultz, *Human-robot interaction: a survey*. Now Publishers Inc, 2008.

[3] A. Leigh, J. Pineau, N. Olmedo, and H. Zhang, "Person tracking and following with 2d laser scanners," in *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 726–733.

[4] Y. Sung and W. Chung, "Hierarchical sample-based joint probabilistic data association filter for following human legs using a mobile robot in a cluttered environment," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 3, pp. 340–349, 2015.

[5] J. Yuan, S. Zhang, Q. Sun, G. Liu, and J. Cai, "Laser-based intersection-aware human following with a mobile robot in indoor environments," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 354–369, 2018.

[6] M. Wang, D. Su, L. Shi, Y. Liu, and J. V. Miro, "Real-time 3d human tracking for mobile robots with multisensors," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5081–5087.

[7] K. Koide and J. Miura, "Identification of a specific person using color, height, and gait features for a person following robot," *Robotics and Autonomous Systems*, vol. 84, pp. 76–87, 2016.

[8] T. Linder, S. Breuers, B. Leibe, and K. O. Arras, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 5512–5519.

[9] M. Zhang, X. Liu, D. Xu, Z. Cao, and J. Yu, "Vision-based target-following guider for mobile robot," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9360–9371, 2019.

[10] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu, "Improving multiple object tracking with single object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2453–2462.

[11] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4282–4291.

[12] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "Atom: Accurate tracking by overlap maximization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4660–4669.

[13] K. Koide, J. Miura, and E. Menegatti, "Monocular person tracking and identification with on-line deep feature selection for person following robots," *Robotics and Autonomous Systems*, vol. 124, p. 103348, 2020.

[14] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth prediction," in *The International Conference on Computer Vision (ICCV)*, October 2019.

[15] D. Park, R. Ambrus, V. Guizilini, J. Li, and A. Gaidon, "Is pseudo-lidar needed for monocular 3d object detection?" in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[16] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[17] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2353–2362.

[18] Y. Cheng, B. Yang, B. Wang, W. Yan, and R. T. Tan, "Occlusion-aware networks for 3d human pose estimation in video," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 723–732.

[19] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *European conference on computer vision*. Springer, 2016, pp. 561–578.

[20] V. N. Nguyen, Y. Hu, Y. Xiao, M. Salzmann, and V. Lepetit, "Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6771–6780.

[21] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," *International Journal of Computer Vision*, vol. 80, no. 1, pp. 3–15, 2008.

[22] W. Choi and S. Savarese, "Multiple target tracking in world coordinate with single, minimally calibrated camera," in *European Conference on Computer Vision*. Springer, 2010, pp. 553–567.

[23] B. Lorenzo, K. Sven, and A. Alexandre, "Perceiving humans: From monocular 3d localization to social distancing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7401–7418, 2022.

[24] H. Chen, P. Wang, F. Wang, W. Tian, L. Xiong, and H. Li, "Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[25] G. Moon, J. Y. Chang, and K. M. Lee, "Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 10 132–10 141.

[26] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1653–1660.

[27] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5693–5703.

[28] A. Newell, Z. Huang, and J. Deng, "Associative embedding: End-to-end learning for joint detection and grouping," *Advances in neural information processing systems*, vol. 30, 2017.

[29] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[30] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[31] Y. Ma, S. Soatto, J. Košecká, and S. Sastry, *An invitation to 3-d vision: from images to geometric models*. Springer, 2004, vol. 26.

[32] X. Fei, H. Wang, L. L. Cheong, X. Zeng, M. Wang, and J. Tighe, "Single view physical distance estimation using human pose," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 406–12 416.

[33] L. Bertoni, S. Kreiss, and A. Alahi, "Monoloco: Monocular 3d pedestrian localization and uncertainty estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6861–6871.

[34] B. X. Chen, R. Sahdev, and J. K. Tsotsos, "Integrating stereo vision with a cnn tracker for a person-following robot," in *International Conference on Computer Vision Systems*. Springer, 2017, pp. 300–313.

[35] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, 2022.

[36] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, and F. Yu, "Quasi-dense similarity learning for multiple object tracking," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2021.

[37] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[38] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "Siamrpn++: Evolution of siamese visual tracking with very deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4282–4291.

[39] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, "Learning spatio-temporal transformer for visual tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 448–10 457.

[40] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.