

Person Re-Identification for Robot Person Following with Online Continual Learning

Hanjing Ye¹, Jieting Zhao¹, Yu Zhan¹, Weinan Chen², Li He¹ and Hong Zhang^{1*}, *Fellow IEEE*

Abstract—Robot person following (RPF) is a crucial capability in human-robot interaction (HRI) applications, allowing a robot to persistently follow a designated person. In practical RPF scenarios, the person often be occluded by other objects or people. Consequently, it is necessary to re-identify the person when he/she re-appears within the robot’s field of view. Previous person re-identification (ReID) approaches to person following rely on offline-trained features and short-term experiences. Such an approach i) has a limited capacity to generalize across scenarios; and ii) often fails to re-identify the person when his re-appearance is out of the learned domain represented by the short-term experiences. Based on this observation, in this work, we propose a ReID framework for RPF that leverages long-term experiences. The experiences are maintained by a loss-guided keyframe selection strategy, to enable online continual learning of the appearance model. Our experiments demonstrate that even in the presence of severe appearance changes and distractions from visually similar people, the proposed method can still re-identify the person more accurately than the state-of-the-art methods.

I. INTRODUCTION

Robot person following (RPF) [1] serves as an essential function in many HRI applications, enabling a robot to follow a specified person autonomously. However, the person being followed may become occluded in various situations, such as when other objects or people obstruct the view of the robot in the working environment. Therefore, it is crucial to re-identify the person when he re-appears in the view.

Existing RPF systems can be achieved through two steps: *identify* and *follow*. In the *identify* step, the system performs tracking and possibly ReID to locate the target person, while the *follow* step involves planning and executing the control of the robot to maintain the desired relative position with the target person. In this paper, we focus on the ReID aspect, specifically re-identifying the target person after occlusion. Existing ReID methods for RPF describe a person’s appearance either with hand-crafted features [2]–[4], or with learned features [5]. However, these methods may experience poor generalization when the features are not sufficiently discriminative for re-identifying the person. Some methods [6]–[8] update the tracker online with newly acquired observations of the target person to distinguish the person from the background and other distracting individuals. Such solutions usually do not consider the appearance model of a person explicitly, leading to suboptimal ReID performance. In addition, all methods mentioned above learn from



Fig. 1. Robot person following with online continual learning. In this end, long-term experiences are utilized to learn a complete and discriminative appearance model of the target person.

short-term experiences only, i.e., the most recently observed samples. This results in limited discriminative ability when the re-appearance of the target person is out of the learned domain represented by the short-term experiences.

To solve the above problems, we propose to utilize long-term experiences to model the target person’s appearance. Specifically, we approach the person ReID in RPF as a problem of online continual learning (OCL) [9], which aims to learn the newest knowledge without forgetting long-term experiences using a size-limited long-term memory. This idea has shown promising results in existing works on dense mapping [10] and place recognition [11]. For example, IMap [10] incrementally learns a NeRF-based dense map by replaying images and poses from a sparse keyframe set, where camera poses are estimated through the tracking process. Similarly, BioSLAM [11] constructs a discriminative long-term memory to replay point clouds and positions for learning a life-long place recognition network, where positions are obtained via LiDAR odometry.

To construct a ReID framework for RPF that leverages the long-term experiences, we first construct a long-term memory to store valuable historical samples, which are selected by a loss-guided keyframe selection strategy. These long-term samples, together with the most recent samples, are utilized to optimize ReID features. As a result, the features can capture the current knowledge about the target person without forgetting historical experiences. Furthermore, these features are utilized to learn a ridge-regression-based classifier for the recognition of the target person. Lastly, a ReID lifecycle management is implemented to form a complete ReID solution, which we refer to as **OCLReID**. In our experiments, the RPF system with OCLReID can reliably re-identify and follow the target person even in situations with visually similar distracting people and different appearances after occlusion.

¹Hanjing Ye, Jieting Zhao, Yu Zhan, Li He and Hong Zhang are with Shenzhen Key Laboratory of Robotics and Computer Vision, Southern University of Science and Technology. ²Weinan Chen is with Guangdong University of Technology. *corresponding author (h Zhang@sustech.edu.cn). Code and video are available at <https://sites.google.com/view/oclrf>.

II. RELATED WORK

A. Person ReID in Robot Person Following

Person ReID is crucial for RPF, which helps re-identify the target person after occlusion. Existing ReID methods in RPF usually describe the appearance of the target person with hand-crafted features [2]–[4] or learned features [5]. Examples of hand-crafted features include color histograms [2], geometric attributes [3], and characteristics like height, gait and clothing color [4]. Alternatively, ReID can rely on features learned from a ReID dataset. For example, [5] trains a convolutional neural network (CNN) with a ReID dataset and then extracts CNN features from detected people when performing RPF. Often, these features are further utilized to construct a target classifier with the most recent observed samples, i.e., short-term experiences.

The above methods, however, often fail to re-identify the person in complicated RPF situations because hand-crafted or learned features have a limited capacity to generalize across scenarios. Besides, short-term experiences contain less information for the recognition of the target person compared to the long-term experiences. Given the fact that long-term experiences contain rich knowledge for re-identifying the person, we propose to utilize long-term experiences to optimize ReID features.

B. Person ReID in Computer Vision

Person ReID has been a prominent research area in computer vision, with the main objective of identifying individuals in video surveillance systems [12]. Various methods have been proposed to solve the ReID problem. For instance, [13] introduces a hand-crafted feature that combines eight color channels (RGB, HSV, and YCbCr) and 19 texture channels to achieve viewpoint invariance. Another approach [14] involves using attribute-based features to achieve competitive ReID performance. However, in recent years, with the advancement of deep learning techniques, learned features [15] have become dominant in ReID research due to their end-to-end nature and excellent generalization. Notably, [16] proposes a CNN-based ReID method that effectively models complex photometric and geometric transformations. However, ReID with a global CNN feature can introduce distractive information in case of occlusion, posing a challenge in real-world scenarios.

To address the issue of occlusion, researchers have introduced ReID methods [17], [18] that leverage pre-defined or learned part masks to match features defined with respect to parts of a target person. Considering that an occluded human body is frequently encountered in RPF scenarios, one can use part-guided ReID features to describe a person’s appearance. Still, as mentioned before, these offline-learned features have a limited generalization ability across scenarios. To achieve generalization, we propose in this paper to learn these features incrementally within the OCL framework.

C. Online Continual Learning

OCL addresses the challenge of learning from a non-independent and identically distributed (Non-IID) stream of

data in an online manner, with the objective of preserving and extending historical knowledge [9]. The Non-IID data setting aligns with the observation scenario of our RPF system, in which the appearance of an observed individual significantly varies due to complex backgrounds and the motion of the robot and target.

Recent works in OCL can be categorized into three main families: regularization-based, parameter-isolation-based and memory-replay-based methods. Regularization-based methods [19]–[21] preserve knowledge by adding history-related constraints to the loss function during current task training, thereby balancing the loss gradient direction for old and new knowledge. However, these methods face challenges in finding the desired global optima, making it difficult to strike a balance between both types of knowledge. Parameter-isolation-based methods [22], [23] retain old knowledge by freezing the related parts of the model and only allowing the remaining parts to learn new knowledge. However, these methods are limited by the initial model capacity and require significant training time to achieve good performance. Memory-replay-based methods [24]–[27] utilize memory replays to incrementally learn old knowledge. Examples include Reservoir [27], which randomly forgets samples based on a distribution related to observation times, MIR [24], which randomly updates the memory and retrieves “the hardest” samples for model updating, and ASER [25], which utilizes an Adversarial Shapley value scoring method for memory retrieval to preserve latent decision boundaries for previously observed samples.

Recently, the benefits of memory-replay-based OCL have been demonstrated in several works [10], [11], [28] to enhance the perception ability of robot systems. We therefore adopt a memory-replay-based algorithm in the implementation of our RPF system, although our solution is not limited to any particular OCL algorithm. To the best of our knowledge, we are the first to integrate the OCL concept into an RPF system to incrementally learn ReID features from long-term experiences.

III. METHOD

A. Problem Statement and Overview

Our RPF system is an extension of our previous work [29], represented by the top half of Fig. 2. Our previous RPF system allows for accurate tracking of individuals, even in scenarios with partial occlusion. It first tracks multiple people and then identifies the target person to follow by selecting the corresponding identity (ID). However, when the target person undergoes occlusion and disappears from the camera view, his ID may be removed because no observation is associated with the ID. Therefore, re-identifying the target person after occlusion, either momentary or long-term, becomes crucial. To solve this problem in our current work, we introduce a person ReID process, which is performed by the module in the lower half of Fig. 2. Specifically, the ReID module learns an appearance model of the target person when he can be correctly identified from tracked people. Later, if and when a long-time occlusion occurs, this appearance model is utilized

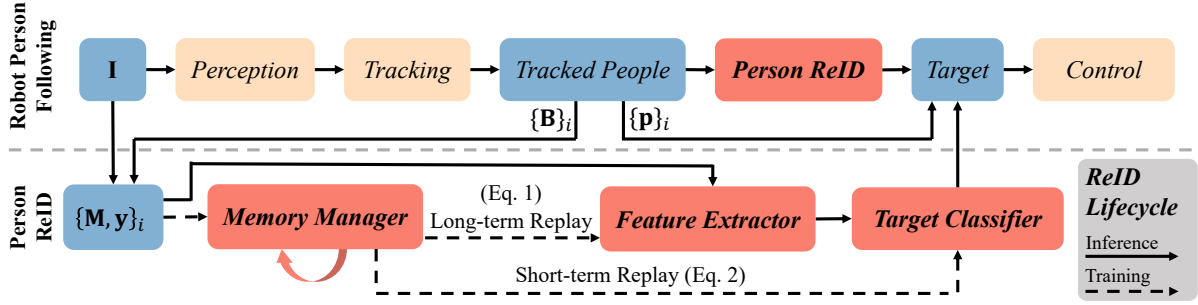


Fig. 2. The top part is the pipeline of our RPF system and the bottom part is the proposed person ReID framework. We obtain image patches $\{M\}_i$ of the tracked people using the current image I and their bounding boxes $\{B\}_i$. When the target person is consistently tracked, his label y represents positives and other people are negatives. Afterward, we add $\{M, y\}_i$ to the *memory manager* for memorization. Additionally, these patches are fed into the *feature extractor* to extract ReID features. These features are utilized by the *target classifier* to estimate the target confidence. If the target confidence is greater than a threshold, the corresponding position p is designated as the target position. In addition to the inference above process, the *memory manager* simultaneously replays long-term and short-term experiences to train the *feature extractor* and the *target classifier*, respectively. If the target person is not found among the tracked individuals, the training process pauses, and all observations $\{M, y\}_i$ become candidates for re-identification. The above training and inference processes are managed by the *ReID lifecycle*.

to re-identify the target person among all the tracked people. In our work, this appearance model comprises the *feature extractor* and the *target classifier*.

In each ReID period, we capture image patches $\{M\}_i$ of the tracked individuals using the current image I and their corresponding bounding boxes $\{B\}_i$. When the target person is consistently tracked, his label y represents a positive sample, while labels for other people are negatives. Subsequently, these patches $\{M\}_i$ are fed into the *feature extractor* for extracting ReID features (Sec. III-B) and these features are further utilized by the *target classifier* to estimate the target confidence (Sec. III-C). If the target confidence is greater than a threshold, the corresponding position p is designated as the target position.

In addition to the inference process mentioned above, we add $\{M, y\}_i$ to the *memory manager* (Sec. III-D) for performing memory-replay-based OCL. Specifically, as shown in Fig. 3, the *feature extractor* is incrementally optimized with long-term experiences ($m_{lt} \cup m_{st}$) in an OCL manner through Eq. 1. Besides, the *target classifier* is learned with short-term experiences m_{st} through Eq. 2. If the target person is not found among the tracked people, the training process pauses, and all observations $\{M, y\}_i$ become candidates for re-identification. Above training and inference processes are managed by the *ReID lifecycle*, which is explained in detail in Algorithm 1.

B. Feature Extractor

First, we extract ReID features for the tracked people with a feature extractor. Given an image I and a person's bounding box B , we extract his image patch, denoted as M . Subsequently, we learn a feature extractor f by a standard CNN, which extracts local features from M . To represent partially visible human bodies, we further transform these local features into features associated with the body parts [17]. These features are denoted as $F \in \mathbb{R}^{N \times C}$, where N represents the number of body parts and C is the size of feature dimension. In previous RPF works [3]–[5], the feature extractor is learned offline and fixed, under the assumption of independent and identically distributed (IID)

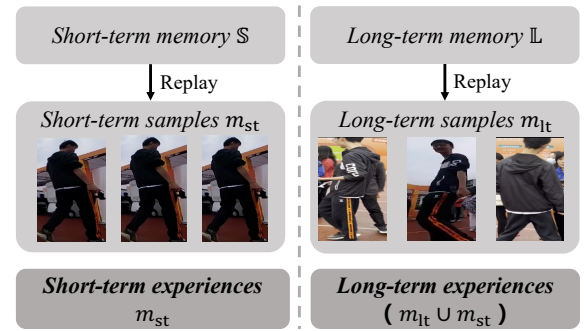


Fig. 3. Long-term experiences, responsible for training the feature extractor, consist of short-term samples m_{st} and long-term samples m_{lt} , with m_{lt} being sampled from the long-term memory L . This memory contains sparse yet valuable historical samples, maintained by the *memory manager*. Short-term experiences m_{st} , used for learning the target classifier, are sampled from the short-term memory S . This memory stores the most recent observed samples, representing the latest knowledge.

observations. However, this assumption may not be valid in an application such as our RPF. For instance, it may not hold when the target person's appearance is non-discriminative in the pre-defined feature space, or when re-appearance of the target person is different from the appearance model learned from short-term experiences. These two problems are commonly referred to as distribution drift [9].

To address these problems, we adopt the concept of OCL [9]. Instead of utilizing a fixed feature extractor, we continually update the feature extractor with long-term experiences. Due to the requirement of efficient learning, OCL demands that the model is trained with only one limited batch at a time, and other batches are not included. In addition, OCL requires that the batch should contain current and historical samples. Therefore, we typically maintain a long-term memory, denoted as L , to store a subset of historical samples. In every ReID period, L replays only one batch, denoted as $m_{lt} \subset L$. Besides, the most recent observed K samples, denoted as m_{st} , are included to represent the current knowledge. Our OCL formulation thus can be represented as follows:

$$\arg \min_{\theta_f} \sum_{(M, y) \in \{m_{st} \cup m_{lt}\}} \mathbb{E}_{(M, y)} [\mathcal{L}_F(f(M; \theta_f), y)] \quad (1)$$

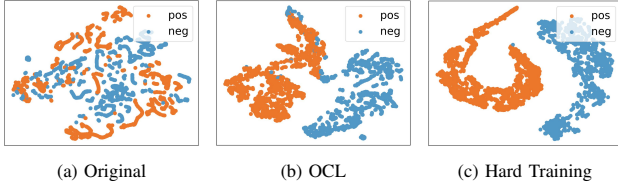


Fig. 4. Feature distribution of the target person (positive) and other distracting people (negative) across all observed samples at the end of sequence. (a) Original distribution which shows that it is hard to discriminate positives and negatives with a fixed feature extractor. (b) Distribution of learning in an OCL manner. (c) Distribution of learning in a hard training manner, which is the most ideal situation.

where \mathbf{M} and \mathbf{y} represents a person’s image patch and label, respectively. f is the feature extractor to be learned, θ_f is the parameter of f , and \mathcal{L}_F is the loss function. In this work, inspired by [17], we apply a mixed loss, combining the cross entropy loss and the triplet loss, for learning a representation that is robust to occlusions.

By continually learning from these long-term experiences ($m_{st} \cup m_{lt}$), the feature extractor incrementally acquires current knowledge while retaining previous experiences. This can be demonstrated by Fig. 4, which shows that training the feature extractor in an OCL manner leads to the target person’s features being distinguishable from others throughout the observed samples in the sequence. This ability of incremental learning enables the robot to re-identify the target person if their re-appearance exists in the previous experiences.

After feature extraction, previous works [17], [18] usually achieve ReID by averaging the similarities of features across all query-gallery pairs, assuming that the query feature and the gallery features are strictly in the same feature space. This requires one to re-extract features with the latest feature extractor from all samples in the memory buffer. However, for the purpose of effective RPF, this approach is not feasible due to the large size of our long-term memory. Therefore, to ensure efficient ReID processing, we leverage short-term experiences to learn a classifier (Sec. III-C).

C. Target Classifier

We learn a target classifier g using short-term experiences, denoted by m_{st} and defined as the most recent observed pairs of image patches and labels $\{\mathbf{M}, \mathbf{y}\}$. These pairs are replayed from a short-term memory \mathbb{S} , representing the latest knowledge about the target person. The learning process can thus be expressed as follows:

$$\arg \min_{\theta_g} \sum_{(\mathbf{M}, \mathbf{y}) \in m_{st}} \mathbb{E}_{(\mathbf{M}, \mathbf{y})} [\mathcal{L}_C(g(f(\mathbf{M}); \theta_g), \mathbf{y})] \quad (2)$$

where f is the current feature extractor without optimization during the training of the classifier g . θ_g is the parameter of g , and \mathcal{L}_C is the loss function for the classifier training. For g , we employ the ridge regression (RR) model with L2 regularization as our classifier, although any other classifiers that are capable of efficient optimization and inference can also be employed. Specifically, we train N RR models where each model is represented as $\mathbf{W}_i \in \mathbb{R}^{1 \times C}$ corresponding to a part-level classifier. The target confidence s is estimated

Algorithm 1: ReID Lifecycle of OCLReID

Input: Current image \mathbf{I} and tracked people $\{\mathbf{B}, \mathbf{p}\}_i$ representing bounding boxes and positions, target person’s identity id , target confidence s , short-term memory \mathbb{S} , long-term memory \mathbb{L} , feature extractor f and target classifier g

Output: Target person’s position $\{\mathbf{p}\}_{id}$ in the current frame

- 1 Extract image patches \mathbf{M} from \mathbf{I} and \mathbf{B} ;
- 2 Construct the observation set $\{\mathbf{M}, \mathbf{y}\}_i$ where $\mathbf{y} = 1$ if $i == id$, otherwise $\mathbf{y} = 0$;
- 3 Extract features \mathbf{F} from \mathbf{M} with f ;
- 4 **if** $id \in \{i\}$ **then**
- 5 Estimate s of the target person based on Eq. 3;
- 6 **if** $s > \delta_{sw}$ **then**
- 7 Consider $\{i\}$ as identities of negative tracks;
- 8 $\{\mathbf{M}, \mathbf{y}\}_{id} \rightarrow \mathbb{L}$ if it is a keyframe based on Eq. 5;
- 9 $\{\mathbf{M}, \mathbf{y}\}_{\bar{i}} \rightarrow \mathbb{L}$ based on FILO rule;
- 10 Consolidate \mathbb{L} with OCL techniques if \mathbb{L} is full;
- 11 $\{\mathbf{M}, \mathbf{y}\}_{id} \rightarrow \mathbb{S}$, $\{\mathbf{M}, \mathbf{y}\}_{\bar{i}} \rightarrow \mathbb{S}$ based on FILO rule;
- 12 Sample m_{st} and m_{lt} from \mathbb{S} and \mathbb{L} , respectively;
- 13 Train f with m_{st} and m_{lt} based on Eq. 1;
- 14 Train g with m_{st} based on Eq. 2;
- 15 **Return** target position $\{\mathbf{p}\}_{id}$;
- 16 **else**
- 17 Let $id = -1$, indicates id switch between the target person and other people;
- 18 **Return** \emptyset ;
- 19 **else**
- 20 Estimate s of the i_{th} person based on Eq. 3;
- 21 **if** $s > \delta_{reid}$ for consecutive ζ_{reid} frames **then**
- 22 Let $id = i$, indicates successful target person ReID;
- 23 **Return** target person’s position $\{\mathbf{p}\}_{id}$;
- 24 **else**
- 25 **Return** \emptyset ;

by averaging the outputs from all part-level classifiers:

$$s = \frac{\sum_{i=0}^N \mathbf{W}_i \mathbf{F}_i^T}{N} \quad (3)$$

where $\mathbf{F}_i \in \mathbb{R}^{1 \times C}$ represents the i_{th} part feature of \mathbf{F} where $\mathbf{F} = \tilde{f}(\mathbf{M})$. Each RR model \mathbf{W}_i is optimized with the most recent K features extracted from m_{st} :

$$\arg \min_{\mathbf{W}_i} \|\mathbf{W}_i \mathbf{X}_i^T - \mathbf{y}\|_2^2 + \lambda \|\mathbf{W}_i\|_2^2 \quad (4)$$

where $\mathbf{X}_i = \{\mathbf{F}_i^1, \mathbf{F}_i^2, \dots, \mathbf{F}_i^K\} \in \mathbb{R}^{K \times C}$ represents the features of the i_{th} part. \mathbf{y} indicates the labels and λ is a regularization parameter. The optimal solution, which is obtained using linear least squares, is given by $\mathbf{W}_i^* = (\mathbf{X}_i^T \mathbf{X}_i + \lambda \mathbf{I})^{-1} \mathbf{X}_i^T \mathbf{y}$.

This formulation can efficiently regress the classification boundary since the sizes of both the short-term memory and feature dimensions are small. Furthermore, it can also generalize to distinguish historical samples, although the classifier is trained with short-term experiences only. This is because learned features are discriminative enough to establish a clear classification boundary with a few samples. This can be observed from Fig. 4 (b) and further verified in the experiments.

D. Memory Manager

To leverage long-term experiences, we establish a long-term memory denoted as \mathbb{L} , responsible for storing valuable samples, i.e., pairs of image patches and labels. When

presented with a new sample, the memory manager employs a *keyframe selection* strategy to decide whether to add this sample to the memory buffer. Once the buffer reaches its capacity, *memory consolidation* takes effect to create space by purging certain samples. In addition to the sample insertion and removal, the process of selecting samples for replay during model optimization (Eq. 1) is equally important and is overseen by the *memory replay* mechanism. In the following, we will introduce our *keyframe selection* strategy, as well as the *memory replay and consolidation* processes.

1) *Keyframe Selection*: Adding the newest sample directly to \mathbb{L} may not be appropriate because the appearance of the target person in adjacent frames is often similar, and therefore, it may not provide additional information. Since information in images is temporally correlated and therefore highly redundant, we insert a keyframe to \mathbb{L} only if it is informative. To this end, inspired by [10], we employ a *loss-guided* keyframe selection strategy to assess the significance of the incoming sample. Specifically, every time a new target sample is added to \mathbb{L} and the feature extractor is optimized, we save a duplicate of the latest feature extractor f and record the loss from this optimization as l_t . The subsequent sample $\{\mathbf{M}, \mathbf{y}\}_{id}$ will then be used to optimize the duplicated f . If the optimization loss is larger than the previous loss l_t by a margin, this sample will be added to \mathbb{L} . This process can be expressed as:

$$\delta = \mathcal{L}_F(f(\mathbf{M}), \mathbf{y}) - l_t \quad (5)$$

If $\delta > \delta_l$, the sample is added to \mathbb{L} , indicating that the forthcoming sample contributes additional information to the learned feature extractor.

2) *Memory Replay and Consolidation*: To preserve valuable experiences, we follow the standard technique of memory-replay in OCL to replay samples for our feature extractor learning or consolidate the memory by removing non-informative samples. We show that, with existing OCL techniques, we can mitigate the forgetting problem and further help enhance person ReID ability. Specifically, the consolidation process is triggered when \mathbb{L} is full. For example, it would prefer to forget non-discriminative samples [11] or randomly selected samples based on a dynamic-change random distribution [27].

The complete ReID lifecycle of our OCLReID framework is summarized in Algorithm 1. The algorithm optimizes the appearance model upon successful identification of the target person, identification recognized when the target *id* exists within the tracked individuals and the target confidence s surpasses the threshold δ_{sw} . When the target person is lost, the algorithm re-identifies him from all observed individuals. An individual is considered as the target person if his estimated confidence has surpassed a threshold δ_{sw} for consecutive ζ_{reid} frames.

IV. EXPERIMENTS

Through comprehensive experiments, we evaluate the OCLReID in RPF and conduct an ablation analysis to determine the effectiveness of different components.



Fig. 5. Examples of the custom-built dataset. It contains challenging situations of similar appearance of distracting people and different appearance after occlusion (lower back body vs. whole front body).

A. Experimental Setup

1) *Dataset*: We conduct experiments on a public dataset [6] and a custom-built dataset. Both datasets consist of image sequences with the ground truth provided in the form of bounding boxes around the target person. The public dataset, named *icvs*, includes challenging scenarios such as quick multi-people-crossing, illumination changes, and appearance variations. However, this public dataset lacks scenarios that require person ReID, such as occlusion and similar appearances of distracting people (as shown in Fig. 5). To address this limitation, we created a custom dataset that includes these challenging scenarios. The custom dataset comprises four sequences named *corridor1*, *corridor2*, *lab-corridor*, and *room*.

2) *Implementation Details*: For all experiments, we set the following default parameters: memory sizes $|\mathbb{S}| = 64$ and $|\mathbb{L}| = 512$, a batch size of 64 for each replay including long-term and short-term relays, a regularization parameter $\lambda = 1.0$ for RR, a keyframe selection threshold $\delta_l = 0.02$, an id switch threshold $\delta_{sw} = 0.35$, a ReID threshold $\delta_{reid} = 0.7$ and a number of consecutive frames $\zeta_{reid} = 5$. In this paper, for representing the part-level features, we define ten parts: $\{\text{front, back}\} \times \{\text{head, torso, legs, feet, whole}\}$. For orientation estimation, we employ MonoLoc [35] trained on the MEBOW dataset [36], to infer the orientation using detected joint positions from AlphaPose [37]. These joint positions are also utilized to estimate the visible parts. We use ResNet18 as our feature extractor, pre-trained on the MOT16 [38] dataset. During OCL for the ResNet18, only the layers after *conv3* are trainable (including *conv3*).

All evaluations are conducted on a computer with an Intel® Core™ i9-12900K CPU and NVIDIA GeForce RTX 3090. For real robot experiments, we use a Unitree Go1 quadruped robot (see Fig. 1) with Intel NUC 11 mini PC powered by Core i7-1165G7 CPU and NVIDIA GeForce RTX2060-laptop GPU. A dual-fisheye Ricoh camera is mounted on the robot, providing cropped perspective images with a resolution of 640×480 and a frequency of 30Hz.

3) *Baselines*: In our evaluation, we assess the person ReID ability of our OCLReID framework using the same approach as previous RPF works [5], [6], [29], where RPF is treated as a special case of object tracking. Specifically, we integrate the ReID module into the RPF system to evaluate the person ReID capability by assessing the person tracking performance of the entire RPF system. We compare our method with popular baselines, which include both feature-based methods and learning-based methods. The difference between them lies in the fact that the former does not update the feature extractor, while the latter does. In our experi-

TABLE I. Tracking mean accuracy (%) (t-mAcc) of the baseline and our method in the custom-built dataset[†] and the public dataset^{††}. F represents a feature-based method and L indicates a learning-based method. For any feature-based method, the tracking performance is always better when combining with our OCLReID method, compared to Koide’s ReID approach [5].

Methods	Type	t-mAcc (%)				
		<i>corridor1</i> [†]	<i>corridor2</i> [†]	<i>lab-corridor</i> [†]	<i>room</i> [†]	<i>icvs</i> ^{††}
SiamRPN++ [30]	L	44.8	55.9	46.1	42.6	93.6
STARK [31]	L	44.3	83.8	73.1	65.8	96.5
SORT [32]	F	42.0	20.2	29.5	58.3	40.4
SORT w/ Koide’s method [5]	F + ReID	52.8	70.0	87.0	85.8	90.2
SORT w/ Our OCLReID	F + ReID	92.8	95.6	92.7	96.8	96.8
OC-SORT [33]	F	42.0	20.2	29.5	58.3	56.6
OC-SORT w/ Koide’s method [5]	F + ReID	50.4	74.6	79.5	31.7	91.6
OC-SORT w/ Our OCLReID	F + ReID	92.8	95.3	92.9	96.8	96.7
ByteTrack [34]	F	42.0	20.2	29.5	58.3	88.6
ByteTrack w/ Koide’s method [5]	F + ReID	48.3	67.7	68.3	60.2	92.0
ByteTrack w/ Our OCLReID	F + ReID	93.5	94.9	96.0	96.8	97.0

ments, we compare our approach against SORT [32], OC-SORT [33], and ByteTrack [34] as feature-based methods, and SiamRPN++ [30] and STARK [31] as learning-based methods. As in previous RPF works [5], [6], [29], where a feature-based method and a ReID module are combined to create a complete RPF system, we integrate Koide’s ReID module [5] and our OCLReID module into the same feature-based methods, respectively, to evaluate their ReID capability. To investigate the impact of different methods of memory consolidation (as illustrated in Sec. III-D) on person ReID ability, we conduct experiments involving three methods: BioSLAM [11], MIR [39], and Reservoir [27]. These methods are employed to assess whether any form of memory consolidation can enhance the performance of person ReID.

B. Person Tracking Evaluation

1) *Metric*: The evaluation metrics of person tracking rely on those employed in previous RPF studies [5], [6], [29]. We assess tracking performance in the image space using the tracking mean accuracy (t-mAcc) as the evaluation metric, which is defined as follows:

$$\text{t-mAcc} = \frac{1}{N} \sum_{i=0}^N a_i \quad (6)$$

where N represents the number of frames within a sequence and a_i is a binary indicator. It equals 1 if the distance between the recognized and ground-truth bounding boxes is less than 50 pixels, and 0 otherwise.

2) *Experimental Results*: The results are shown in Table I. We observe that with any feature-based method, our OCLReID can achieve the highest tracking accuracy on all sequences compared to other baselines. Specifically, when utilizing ByteTrack [34], the RPF system with our OCLReID surpasses the second-best method by 45.2% on *corridor1*, 12.7% on *corridor2*, 20.0% on *lab-corridor*, 31.3% on *room* and 0.5% on *icvs*. Furthermore, compared to Koide’s ReID method [5], our OCLReID can achieve a better improvement of tracking performance on all sequences independently of which RPF method is used. For instance, when combined

TABLE II. Ablation study on *corridor2* and *lab-corridor* in terms of ReID mean accuracy at the end of training (r-mEAcc) (%) and the tracking mean accuracy (t-mAcc) (%). The number in the brace represents the counts of sample insertion to the long-term memory buffer. All r-mEAcc values are averages of three runs.

Methods	<i>corridor2</i>		<i>lab-corridor</i>	
	r-mEAcc ↑	t-mAcc ↑	r-mEAcc	t-mAcc
w/o long-term replay	59.2 ± 0.0	20.2	31.7 ± 0.0	54.2
w/o short-term replay	96.0 ± 0.8	45.1	89.3 ± 13.6	68.6
w/o KF selection	96.8 ± 0.2	94.2 (3470)	93.0 ± 0.8	95.1 (4116)
BioSLAM [11]	94.9 ± 2.0	94.9	79.0 ± 22.5	93.8
MIR [39]	94.7 ± 0.8	95.4	86.1 ± 14.3	96.1
Reservoir [27]	96.5 ± 0.4	94.9 (388)	94.0 ± 0.7	96.0 (224)

with ByteTrack, our method improves the tracking accuracy by 45.2% on *corridor1*, 27.2% on *corridor2*, 27.7% on *lab-corridor*, 36.6% on *room* and 5.0% on *icvs* compared to Koide’s ReID method.

Above results indicate that our OCLReID can: i) utilize stable tracking results from a feature-based tracking method to learn a discriminative appearance model. This model further assists the RPF system in re-identifying the target person after occlusion; and ii) achieve better ReID performance compared to existing baselines. This could be attributed to its ability to leverage long-term experiences to construct a more complete appearance model. This incrementally learned appearance model captures more knowledge about the target person, enabling successful ReID even in challenging RPF scenarios.

C. Online Continual Learning Evaluation

1) *Metric*: The evaluation of OCL [9] aims to assess how well the model remembers previous knowledge, which is essential for person ReID in RPF, as previous knowledge contain potentially matching experiences for future ReID. Additionally, incrementally remembering previous knowledge might result in a more generalized feature extractor. Specifically, we treat the OCL evaluation for person ReID as a classification task, where we assume that the true identity of the target person is known in each frame, and the model incrementally learns with known labels. For evaluation purposes, we divide each sequence into eight segments, each representing different levels of distribution drift. During

incremental learning, after each segment is learned, the model is evaluated on previously seen segments. Similar to [9], we use the ReID mean accuracy at the end of training (r-mEAcc) as our OCL evaluation metric:

$$\text{r-mEAcc} = \frac{1}{8} \sum_{j=0}^8 a_{8,j} \quad (7)$$

where $a_{8,j}$ represents the average accuracy on the j^{th} segment with the model having been learned from all eight segments. Higher r-mEAcc values indicate that the model retains more of the previous knowledge during incremental learning.

2) *Experimental Results:* The results are shown in Table II. The row of “w/o long-term replay” excludes the long-term replay process from the framework while retaining the classifiers for online learning and ReID. Compared to the performance of original setup (*Reservoir*), its r-mEAcc and t-mAcc drop by 37.3% and 74.7%, respectively, on *corridor2*. This underscores the importance of *long-term replay* for incrementally memorizing long-term experiences and constructing a appearance model at a long-term scale. Moreover, this long-term appearance model significantly improves the person tracking ability, resulting in higher t-mAcc values for all OCL methods that perform above 92.0% accuracy on both sequences.

The row of “w/o short-term replay” removes the classifiers but retains the feature extractor updating with long-term experiences. It constructs a long-term buffer to store the extracted features and re-identifies the target person using the matching strategy with these features (similar to occluded person ReID [17]). This experiment shows that although r-mEAcc is high with 96.0% and 89.3% on *corridor2* and *lab-corridor*, respectively, t-mAcc is low with 45.1% and 68.6%. This result suggests the importance of learning a classifier using short-term experiences. This could be attributed to the fact that the feature extractor is incrementally optimized using samples from different distributions. Therefore, features in the long-term buffer are not strictly in the same feature space. On the other hand, short-term features extracted from short-term experiences exhibit local consistency within the same feature space, which is essential for building a stable classifier. The evaluation of the loss-guided keyframe (KF) selection strategy shows that, *Reservoir* achieves similar r-mEAcc and t-mAcc compared to “w/o KF selection”. However, it stores significantly fewer samples in the long-term memory with 388 frames vs. 3470 on *corridor2* and 224 frames vs. 4116 on *lab-corridor*. This indicates that our loss-guided KF selection strategy can help insert sparse and valuable samples to the memory without compromising the ReID performance.

In summary, the above experiments demonstrate that *long-term replay* can leverage long-term experiences to improve the ReID ability of the RPF system. With these discriminative features, *short-term replay* can thus learn a good classifier for re-identifying the target person.

D. Discussion

As shown in Table I, the proposed OCLReID framework achieves the best identification performance with an average accuracy of 95.6%. Compared to Koide’s ReID method [5], it exhibits superior ReID performance, which helps the RPF system consistently track the target person even in case of occlusion and similar appearance of distracting people. These results indicate that our OCLReID framework can incrementally learn from long-term experiences and help build a more complete and discriminative appearance model.

Results of “w/o long-term replay” and “w/o short-term replay” shown in Table II highlight the importance of the combination of fast learning from short-term experiences and slow consolidation from long-term experiences in our RPF system. This dual-mechanism design draws inspiration from the dual process theory mentioned in [40], which suggests that a successful robot-environment interaction requires both an intuitive system and a reasoning system. In our work, the intuitive system is constructed by a clear classification boundary optimized with short-term experiences. The reasoning system is built through incremental learning in an OCL manner with long-term experiences, which incrementally reasons about the discriminative information of the target person.

From Table II, we observe that different OCL methods perform differently in terms of r-mEAcc, with *BioSLAM* achieving a performance of 79.0%, *MIR* 86.1%, and *Reservoir* 94.0%. While we do not design an OCL method in this paper to specifically address the catastrophic forgetting problem for RPF, it is a promising and practical direction for better incrementally learning from long-term experiences on a resource-limited robot. Additionally, we observe that although the OCL ability of *BioSLAM* [11] is worse than *Reservoir* [27] with r-mEAcc of 79.0% vs. 94.0% on *lab-corridor*, its tracking accuracy only drops by 2.2%. This indicates that not all historical knowledge needs to be memorized for person ReID in some situations. However, we claim that maximizing the enhancement of ReID ability at a long-term scale is still necessary as it ensures a more complete appearance model for dealing with complex ReID situations.

V. CONCLUSION

We approach person ReID in RPF as a problem of online continual learning, enabling the RPF system to learn incrementally from long-term experiences. As a result, the framework achieves a complete and discriminative appearance model, allowing for effective ReID even in challenging scenarios, such as frequent appearance changes, occlusion, and distracting people with similar appearances. Compared to existing baselines, our OCLReID framework achieves state-of-the-art performance in person ReID within RPF scenarios.

Future directions for OCLReID involve the exploration of methods to consolidate valuable samples, maximizing the learning of appearance representations without forgetting previous knowledge. Additionally, strategies to balance efficient ReID and incremental memorization within crowded

environments will be investigated. These endeavors will contribute to the enhancement of the robustness and efficacy of the OCLReID framework in real-world applications.

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] Y. Yoon, W.-h. Yun, H. Yoon, and J. Kim, "Real-time visual target tracking in rgb-d data for person-following robots," in *2014 22nd International Conference on Pattern Recognition*, 2014, pp. 2227–2232.
- [3] X. Chen, J. Liu, J. Wu, C. Wang, and R. Song, "Lopf: An online lidar-only person-following framework," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–13, 2022.
- [4] 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.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] B. X. Chen, R. Sahdev, and J. K. Tsotsos, "Person following robot using selected online ada-boosting with stereo camera," in *2017 14th Conference on Computer and Robot Vision (CRV)*, 2017, pp. 48–55.
- [9] Z. Mai, R. Li, J. Jeong, D. Quispe, H. Kim, and S. Sanner, "Online continual learning in image classification: An empirical survey," *Neurocomputing*, vol. 469, pp. 28–51, 2022.
- [10] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6229–6238.
- [11] P. Yin, A. Abuduweili, S. Zhao, C. Liu, and S. Scherer, "Bioslam: A bio-inspired lifelong memory system for general place recognition," *arXiv preprint arXiv:2208.14543*, 2022.
- [12] Q. Leng, M. Ye, and Q. Tian, "A survey of open-world person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 4, pp. 1092–1108, 2019.
- [13] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features." *ECCV (1)*, vol. 2008, pp. 262–275, 2008.
- [14] Z. Shi, T. M. Hospedales, and T. Xiang, "Transferring a semantic representation for person re-identification and search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [15] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 6, pp. 2872–2893, 2021.
- [16] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.
- [17] V. Somers, C. De Vleeschouwer, and A. Alahi, "Body part-based representation learning for occluded person re-identification," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 1613–1623.
- [18] Y. Li, J. He, T. Zhang, X. Liu, Y. Zhang, and F. Wu, "Diverse part discovery: Occluded person re-identification with part-aware transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2898–2907.
- [19] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [20] J. Schwarz, W. Czarnecki, J. Luketina, A. Grabska-Barwinska, Y. W. Teh, R. Pascanu, and R. Hadsell, "Progress & compress: A scalable framework for continual learning," in *International conference on machine learning*. PMLR, 2018, pp. 4528–4537.
- [21] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.
- [22] A. Mallya and S. Lazebnik, "Packnet: Adding multiple tasks to a single network by iterative pruning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 7765–7773.
- [23] S. Lee, J. Ha, D. Zhang, and G. Kim, "A neural dirichlet process mixture model for task-free continual learning," in *International Conference on Learning Representations*, 2020.
- [24] R. Aljundi, E. Belilovsky, T. Tuytelaars, L. Charlin, M. Caccia, M. Lin, and L. Page-Caccia, "Online continual learning with maximal interfered retrieval," in *Advances in Neural Information Processing Systems*, 2019, vol. 32, pp. 11 849–11 860.
- [25] D. Shim, Z. Mai, J. Jeong, S. Sanner, H. Kim, and J. Jang, "Online class-incremental continual learning with adversarial shapley value," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 9630–9638.
- [26] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio, "Gradient based sample selection for online continual learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [27] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, and M. Ranzato, "Continual learning with tiny episodic memories," 2019.
- [28] J. Frey, M. Mattamala, N. Chebrolov, C. Cadena, M. Fallon, and M. Hutter, "Fast traversability estimation for wild visual navigation," *arXiv preprint arXiv:2305.08510*, 2023.
- [29] H. Ye, J. Zhao, Y. Pan, W. Cherr, L. He, and H. Zhang, "Robot person following under partial occlusion," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7591–7597.
- [30] 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.
- [31] 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.
- [32] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 3464–3468.
- [33] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-centric sort: Rethinking sort for robust multi-object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9686–9696.
- [34] 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.
- [35] 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 (ICCV)*, October 2019.
- [36] C. Wu, Y. Chen, J. Luo, C.-C. Su, A. Dawane, B. Hanzra, Z. Deng, B. Liu, J. Z. Wang, and C.-h. Kuo, "Mebow: Monocular estimation of body orientation in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [37] 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.
- [38] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," 2016.
- [39] R. Aljundi, L. Caccia, E. Belilovsky, M. Caccia, M. Lin, L. Charlin, and T. Tuytelaars, *Online Continual Learning with Maximally Interfered Retrieval*. Red Hook, NY, USA: Curran Associates Inc., 2019.
- [40] N. Roy, I. Posner, T. Barfoot, P. Beaudoin, Y. Bengio, J. Bohg, O. Brock, I. DePATIE, D. Fox, D. Koditschek, et al., "From machine learning to robotics: challenges and opportunities for embodied intelligence," *arXiv preprint arXiv:2110.15245*, 2021.