



Article

Multi-Templates Based Robust Tracking for Robot Person-Following Tasks

Minghe Cao ¹, Jianzhong Wang ^{1,*} and Li Ming ²

¹ School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China; 3120160104@bit.edu.cn

² Beijing Institute of Automation Equipment, Beijing 100074, China; minglllove@126.com

* Correspondence: cwjzwang@bit.edu.cn

Abstract: While the robotics techniques have not developed to full automation, robot following is common and crucial in robotic applications to reduce the need for dedicated teleoperation. To achieve this task, the target must first be robustly and consistently perceived. In this paper, a robust visual tracking approach is proposed. The approach adopts a scene analysis module (SAM) to identify the real target and similar distractors, leveraging statistical characteristics of cross-correlation responses. Positive templates are collected based on the tracking confidence constructed by the SAM, and negative templates are gathered by the recognized distractors. Based on the collected templates, response fusion is performed. As a result, the responses of the target are enhanced and the false responses are suppressed, leading to robust tracking results. The proposed approach is validated on an outdoor robot-person following dataset and a collection of public person tracking datasets. The results show that our approach achieved state-of-the-art tracking performance in terms of both the robustness and AUC score.

Keywords: person following; robust visual tracking; tracking reliability; response fusion; unmanned ground vehicle



Citation: Cao, M.; Wang, J.; Ming, L. Multi-Templates Based Robust Tracking for Robot Person-Following Tasks. *Appl. Sci.* **2021**, *11*, 8698. <https://doi.org/10.3390/app11188698>

Academic Editor: Manuel Armada

Received: 26 July 2021

Accepted: 16 September 2021

Published: 18 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

While the robotics techniques have not led to full automation, human–robot collaboration scenarios have arisen in diverse domains, such as manufacturing, health care, and entertainment. The major advantage of adopting person-following robots is that it reduces the demand for dedicated teleoperation. In all person-following applications, the robustness of recognizing the target to follow is the most important aspect of the following system. The perception sensors of the person-following system include a camera, laser range-finder, LiDAR, infrared and thermal sensors, and sonar. The RGB camera is widely used for its rich information, compactness, and cost-effectiveness.

To perform the following tasks, the following robot must perceive the relative position of the target in its operating environment. This can be considered as a tracking task. There are many situations in which the robot may lose track in a dynamic environment, e.g., occlusion, illumination variation, scale variation, deformation, etc. Therefore, the target must be tracked in real time without critical failures. The attempts to use visual tracking techniques have flourished over the past decade. In previous approaches, tracking algorithms detected specific features in the input feature space [1]. Schlegel et al. [2] and Hu et al. [3] developed tracking methods in the RGB image space. Shin et al. [4] presented a free tracking algorithm model using optical flow. Koide et al. [5] tracked people using color, height, and gait features. Satake et al. [6] established the distance dependence appearance model using the SIFT feature. Kwolek [7] tracked targets using a color histogram. Satake [8] adopted depth histogram features to fit with support vector machines for robust tracking. Chen et al. [9] deployed the Ada-boosting algorithm to person tracking. Wang et al. [10] adopted the kernelized correlation filter (KCF) as the tracking module in a following mission. Using

traditional features can achieve person tracking under certain circumstances but cannot work well under long-term and complex environments.

Recently, Siamese-based approaches that adopt discriminative correlation of deep features have been proposed to address these issues. In Siamese-based trackers, it is common (some stated in the papers but not implemented in the released code) to only use the first frame as the template to grant template reliability, which achieved good performance in short-term datasets such as OTB [11] and VOT [12].

However, in experiments, using a fixed template can perform well for a certain duration, but over time, the variations in appearance, illumination, scale, deformation, etc., reduce the intensity of the responses to the tracked target, and eventually, tracking is lost. The intuitive solution to this issue is to continue incorporating the latest target information, but Zhang et al. [13] proved that the tracking performance only worsens if a non-discriminative template update strategy is applied throughout the tracking process. We think the reason for that is the introduction of false-positive templates. Therefore, a tracking reliability criterion is needed to safely incorporate new templates.

Providing new target information only enhances the responses of target tracking, but false responses are still caused by similar objects. As illustrated in the ground truth score map row of Figure 1, even though the people have different appearances, they receive high responses in the score map. When the real tracking target crosses or is occluded by these objects, the tracking is easily lost with these interferences. It is also important to eliminate false responses.



Figure 1. Illustration of our score fusion procedure. The ground truth score map not only responds to the real target but also raises the responses in many other areas where the tracker is easily misled. After implementing fusion, the unrelated responses are sufficiently suppressed.

Motivated by the aforementioned analysis, we propose a robust tracking approach to enable robot person-following tasks. A scene analysis module (SAM) is proposed that leverages the statistical characteristics of the cross-correlation responses. The density distribution of the responses is estimated using a Gaussian mixture model. Based on the mutual information of the mixture components, the responses are segmented into instance-aware clusters. As a result, a tracking reliability criterion is proposed based on the size of the center cluster, and distractors that produce false responses are extracted as negative templates. By collecting the positive and negative templates, a score fusion strategy is

applied to enhance the responses of target-tracking and to eliminate false responses, leading to the robust person tracking.

Our main contributions can be summarized as follows: (1) We proposed a tracking reliability criterion based on the variance of center responses. With the criterion, the most recent reliable results can be safely extracted as positive templates, avoiding template pollution. (2) We perform a score fusion strategy that generates the final score map by combining the responses of ground truth template, positive templates, and negative templates. As a result, the target responses are enhanced and distractors are suppressed and eliminated, reducing the chance of incorrect positioning. (3) The proposed method was incorporated in two state-of-the-art approaches, SiamRPN [14] and SiamRPN++ [15], and validated on person-following-based datasets as well as public datasets. The results show that our approaches outperform their base approaches and rank high when competing with other state-of-the-art approaches.

2. Related Work

The tracking performance using traditional features is severely restricted when tracking scenarios are complex. Distinct from handcrafted features, the emergence of deep-learning-based approaches has provided a significant increase in performance. Tracking algorithms based on deep feature representations have achieved state-of-the-art accuracy.

Although these techniques perform well on benchmarks, they often suffer from tracking drift caused by the accumulation of errors. Recently, derived from the idea of tracking by detection, trackers based on the Siamese network have received wide attention. Siamese-based trackers formulate the tracking problem as a similarity learning function and predict the object location by comparing the similarity between the template image and the search image. The Siamese networks are trained offline on large-scale image pairs.

The pioneering method SiamFC [16] uses the Siamese network as a feature extractor and introduces a cross-correlation layer to generate a single channel response map. The correlation can be seen as a similarity calculation, and the response map reflects the similarity between the template and the search region. Following this similarity-learning work, Li et al. propose SiamRPN [14], which enhances the tracking performance by integrating a region proposal network (RPN) into SiamFC. The RPN has two branches: one classification branch in charge of scoring the probability, and the other, a regression branch, is responsible for estimating the coordinates of bounding boxes. Based on SiamRPN, Da-SiamRPN [17] addresses the problem of the imbalance between non-semantic negative examples and semantic distractors of training data through data augmentation. UpdateNet [13] further improves upon DaSiamRPN by incorporating a small network that learns the appearance change of tracked targets. SiamDW [18] takes advantage of deeper neural networks by eliminating the negative impact of padding. SiamRPN++ [15] further improves upon the object-tracking performance using deeper networks and multi-layer fusion, achieving better accuracy while maintaining fair speed. To eliminate the negative effects of anchors, SiamCAR [19] adopts two subnetworks for feature extraction and regression respectively and proposes an anchor-free framework; SiamBAN [20] directly classifies objects and regresses bounding boxes taking advantage of a unified fully convolutional network. The avoidance of setting pre-defined anchors can avoid the tricky hyper-parameter tuning, easing the effect of human intervention. To solve the problem that the size of the object feature region needs to be determined in advance, and the cross-correlation method either retains a lot of unfavorable background information or loses a lot of foreground information, SiamGAT [21] proposes a Graph Attention Module (GAM) to establish a partial correspondence between an object and a search region as a complete bipartite graph.

In long-term tracking, robustness is a common weak point. Siam R-CNN [22] has a two-stage Siamese re-detection architecture and re-detects images by comparing region proposals with the template region. LTMU [23] proposed a meta-updater that guides the tracker update, forming a long-term tracking framework along with an online local tracker, an online verifier, and a SiamRPN-based re-detector. These methods significantly

improved tracking precision but have a low tracking frame rate even in high-end desktops. Wang et al. [24] presented a long-term target tracking method by combining adaptive discriminative correlation filters with a support vector machine-based component. SiameseRM [25] has an object-tracking framework that uses the Siamese network and adopts the Siamese instance search tracker as the re-detection network. Zhang et al. [26] deployed local–global multiple correlation filters for tracking and a Kalman filter re-detection model for re-detection when the correlation filters are unreliable. Methods such as online updater, re-detection module, hierarchical search, and multi-stage framework are commonly used to handle tracking robustness issues in long-term tracking. However, the introduced modules inevitably deteriorate the real-time performance of the approaches.

In this paper, we use SiameseRPN-based trackers as the front end of the following system. For the reason that using the first frame as the template may be easily impacted and lose the target, we adopt a Scene Analysis Module that can safely produce positive and negative targets in the tracking scenes. By fusing the scores of the templates, the target responses are enhanced and noises are suppressed, leading to robust tracking.

3. Method

3.1. Framework

Figure 2 presents the flow chart of our approach. The base tracker part is the common steps of Siamese-based trackers where for each frame; the features of the template image and search image are extracted using a shared-weight deep convolutional backbone. The two extracted features then cross-correlate and produce a score map that consists of the probabilities of similarity between the template and the search image. Usually, Siamese-based trackers employ non-maximum suppression to the scores and choose the corresponding regressed bounding box with the highest value as the tracking result, but these methods do not work well in some situations such as occlusion and appearance change.

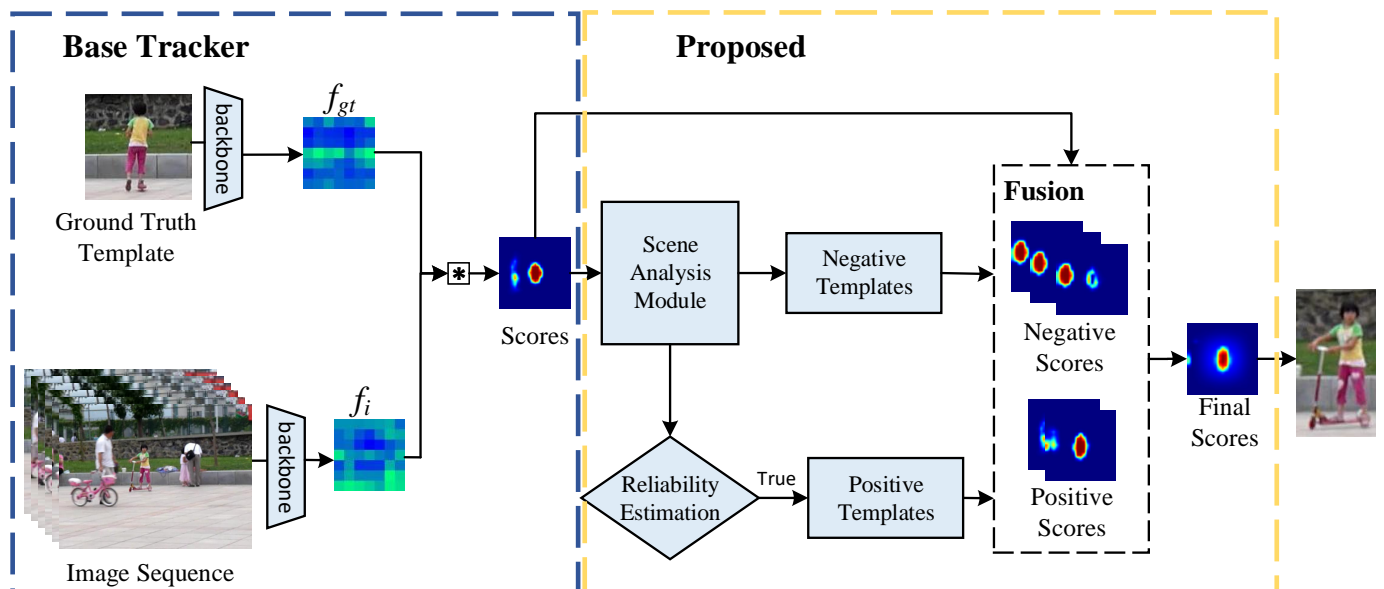


Figure 2. The framework of our approach. The base tracker part is the common framework of Siamese-based trackers. For each frame, the features of the template image and the search image are extracted using a shared-weight network. Then, a score map is generated by the cross-correlation of the two features. To further exploit the information from the score map, we take the scores as a whole and process them using our scene analysis module. The objects with highly similar responses are collected as negative templates. Next, we estimate the tracking reliability leveraging the outcomes of the SAM and collect confident tracking results as positive templates. Finally, the scores of the ground truth, negative, and positive are fused and the tracking box is regressed from the fused score map.

To improve tracking robustness, we take the scores as a whole and further exploit the information provided by the score map using our scene analysis module (SAM). The SAM analyzes the score map by estimating the score densities and segmenting the scores into instance-aware clusters. The SAM provides two contributions: first, a tracking reliability criterion is proposed using the statistical characteristics of the score distribution. If the tracking is determined to be reliable, the tracking target is extracted and collected as positive templates. Second, because of the limitation of the backbone networks, objects that are similar to the tracking target also respond with high values in the score map, which strongly interferes with tracking accuracy. Since the SAM segments scores into instance-aware clusters and each cluster represents a potential tracking target, the targets except for those being tracked are determined as false positive targets and collected as negative templates.

Finally, the score maps of the ground truth and negative and positive templates are fused together and the tracking box is regressed from the fused score map. The positive templates provide more recent information, enhancing the responses of the target-tracking. The negative templates are in charge of suppressing the interference due to similar objects.

3.2. Scene Analysis Module

In Siamese trackers, only the maximum value of the responses is used to predict the result for the candidate target position. However, the outcome may be unreliable for complicated scenes, such as out-of-view and occlusion situations. Nevertheless, the SiamRPN-based trackers provide discriminative responses on foregrounds and backgrounds (Figure 3a). After the generation of the score map, we take the map as a whole and analyze the statistical characteristics of the response distribution. The estimated distribution is further segmented into instance-aware clusters, where each cluster corresponds to a potential object that is similar to the tracked target. The distribution variance of the objects and their bounding boxes are used to establish a tracking confidence criterion and fit false positive objects.

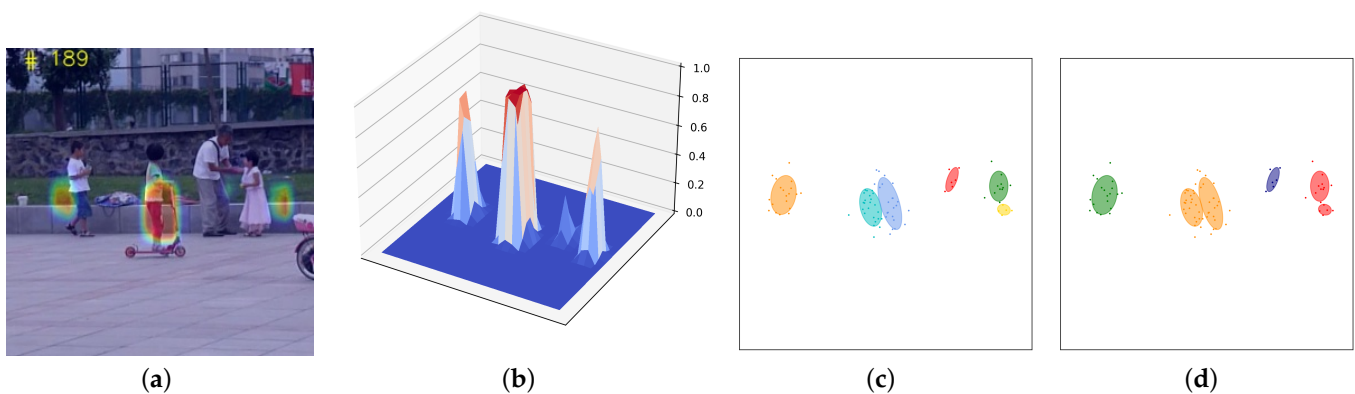


Figure 3. (a) Score map visualized by a heat map. (b) Score map in three dimensions. (c) Density estimation outcome of GMM. The GMM components are visualized using ellipses in different colors. (d) Instance segmentation. The GMM components are further segmented in instance-aware clusters, where each cluster corresponds to a potential object.

3.2.1. Density Estimation

Figure 3a shows a tracking frame and its corresponding responses after cross-correlation. The tracking frames were obtained from the OTB dataset [11]. The backbone was obtained from SiamRPN++ [15]. Figure 3b shows the responses in three dimensions. We can see that with the improvement in training [17] and the [15] network, the responses distribute densely within the potential tracking targets. Therefore, we take the responses as a distribution and first sample it using the accept-reject algorithm. Specifically, given the responding distribution $d(x)$, we select a known probability distribution $q(x)$ and a sufficiently large

constant m , such that $\forall x$, we have $mq(x) \geq d(x)$. Then, we repeatedly sample from a uniform distribution $U(0, 1)$. If the i th sample satisfies

$$u_i \leq \frac{d(x_i)}{mq(x_i)}, \quad (1)$$

we accept x_i as a sample, or reject it otherwise. Figure 3c shows the sampled points from the score map.

After sampling from the score map, we adopt the Gaussian mixture model (GMM) to estimate the probability density. The GMM is a parametric probability density function represented as the sum of Gaussian densities. The representation of GMM is

$$p(x|\Theta) = \sum_{c=1}^C \rho_c \mathcal{N}(x|\mu_c, \Sigma_c), \quad (2)$$

where $\mathcal{N}(x|\mu, \Sigma)$ is the multivariate Gaussian densities, whose parameter $\mu \in \mathbb{R}^2, \Sigma \in \mathbb{R}^{2 \times 2}$ are the mean vector and the covariance matrix. The scalar ρ_c is the weight of the Gaussian component.

Since there is no closed-form solution for the GMM, the expectation–maximization (EM) algorithm [27] is commonly used to find a solution by iteratively maximizing data likelihood until the average data log-likelihood converges to a threshold. Figure 3c depicts the fitting outcome of the example image. The components of the GMM are visualized as different-colored ellipses.

3.2.2. Instance Segmentation

The density of the response map is estimated by the GMM; however, the GMM components are not discriminative in instances. Biemann [28] adopted the Chinese whispers algorithm to solve clustering problems using undirected and weighted graphs, which can further facilitate segmenting the GMM into instance-aware mixtures.

We define $G = (V, E)$ as a graph with nodes $v_i \in V$ and weighted edges $(v_i, v_j, w_{ij}) \in E$. The adjacent matrix \mathcal{W} of graph G is a square matrix, where the entries w_{ij} denote the weight of the edges between v_i and v_j . Since the segmentation is conducted on probability densities, we use Kullback–Leibler divergence (KL divergence) as the metric to set up the weights of the graph edges.

Given two components $f = \rho_f \mathcal{N}(\mu_f, \Sigma_f)$ and $g = \rho_g \mathcal{N}(\mu_g, \Sigma_g)$ of a mixture, according to the definition, their KL divergence is given by

$$D_{KL}(f, g) = \rho_f \left(D_{KL}(\mathcal{N}(\mu_f, \Sigma_f), \mathcal{N}(\mu_g, \Sigma_g)) + \log \frac{\rho_f}{\rho_g} \right), \quad (3)$$

and a closed-form solution is derived as

$$D_{KL}(\mathcal{N}_f, \mathcal{N}_g) = \frac{1}{2} \left\{ \log \frac{|\Sigma_g|}{|\Sigma_f|} - n + \text{tr}(\Sigma_g^{-1} \Sigma_f) + (u_g - u_f)^T \Sigma_g^{-1} (u_g - u_f) \right\}$$

We take the GMM components as the nodes in G . If the KL divergence of two nodes is greater than a threshold, an edge is established and the reciprocal of its divergence is set to the corresponding position in the adjacent matrix \mathcal{W} . Then, the algorithm iteratively segments by grouping nodes that have the maximum mutual weights.

Figure 3d presents an example of an outcome of applying instance segmentation. The GMM components are segmented into four clusters, where each cluster corresponds to a potential object in Figure 3a.

3.3. Reliability Estimation

The instance segmentation clusters responses into instance-aware GMM mixtures. When tracking in reliable circumstances, each object has its own cluster as illustrated in Figure 3d, and the size of the cluster remains stable. However, when a potential occlusion occurs, the instance clusters merge together given their small divergence values, resulting in a large size variation. The upper row of Figure 4 illustrates example scenes before and after potential occlusion. We define the standard deviation matrix of the i th instance as s_i , and σ_i as the max eigenvalue of s_i . We introduce a reliability parameter τ :

$$\tau = \frac{\max\{\sigma_i\}}{0.5 * SIZE_{score}}, \quad (4)$$

where $SIZE_{score}$ is the size of the response map.

Figure 4 shows the values of τ over frames on Girl2 of the OTB dataset. When a potential occlusion occurs, τ presents a peak. We set a threshold parameter τ_t . When $\tau \geq \tau_t$, it indicates a potential occlusion (see the τ values in frames 50, 70, 100, 120, etc.); the tracking results are unreliable. Conversely, if τ satisfies $\tau \leq \tau_t$ in N_r successive tracking frames, the result is considered reliable.

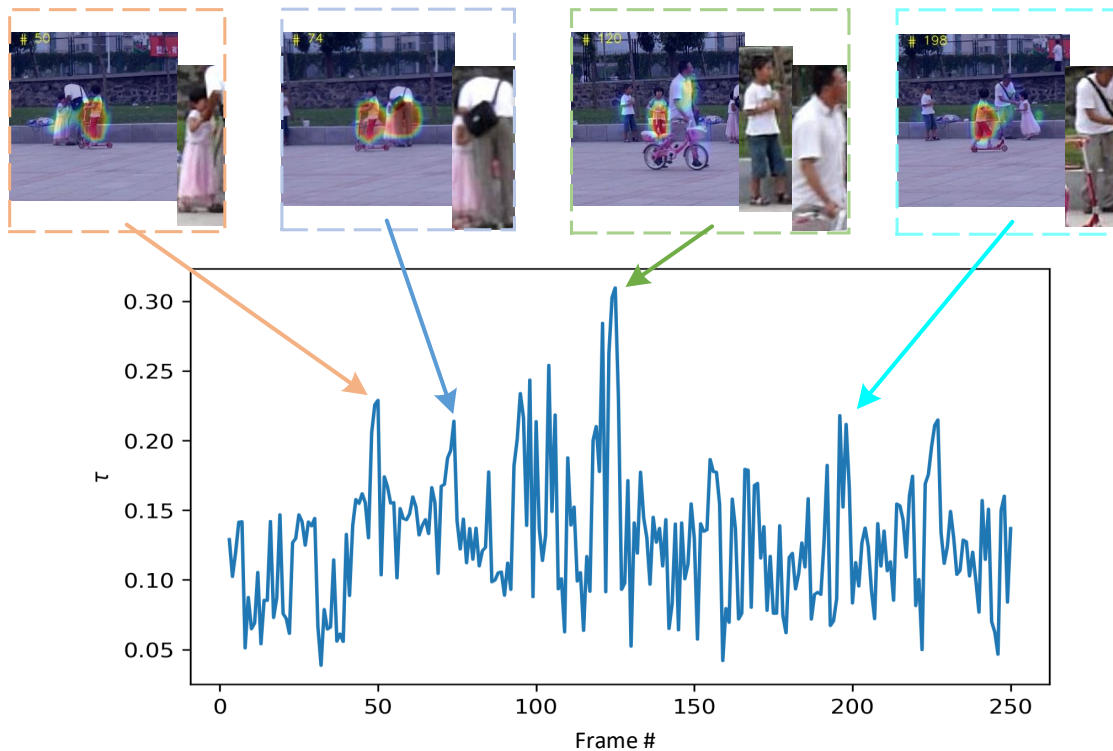


Figure 4. The plot of τ values on Girl2 of the OTB dataset (bottom row), the heated score map of potential occlusion scenes, and corresponding negative templates (top row). In the τ plot, the peaks indicate potential occlusions, leading to an unreliable tracking result.

3.4. Templates Collection

3.4.1. Positive Templates

In Section 3.3, we proposed a tracking reliability criterion. Based on this criterion, we can safely extract the latest reliable tracking target as a positive template. We define N_{pos} as the maximum number of positive templates stored during tracking tasks. The templates are stored as a queue; if the number of templates exceeds N_{pos} , we remove the top of the queue.

3.4.2. Negative Templates

The SAM segments the GMM into instance-aware mixtures. Based on the number of segmented mixtures, we can infer the number of similar objects in the current tracking scene; then, except for the tracked target, the bounding boxes of each object are regressed and the object images are cropped as negative templates. As such, the false-positive responses can be suppressed in the score map fusion step.

We set N_{neg} as the maximum number of negative templates stored during tracking tasks. When a new negative template arrives, the Euclidean distances between the new template feature and the stored template features are calculated. Then, the template that has the smallest distance is removed, and the new template is added.

3.5. Score Fusion

Equation (5) describes the score fusion process, where N_{pos} and N_{neg} are the number of collected positive templates and negative templates, respectively; f_{pos} and f_{neg} are the extracted features using the backbone network; and $\varphi(\cdot)$ is the cross-correlation procedure.

$$Score = \varphi(f_{gt}) + \sum_i^{N_{pos}} \varphi(f_{pos}^i) - \sum_j^{N_{neg}} \varphi(f_{neg}^j) \quad (5)$$

The idea of score fusion is to enhance the responses of target-tracking using positive templates and to suppress false responses using negative templates. Figure 1 presents examples of the outcome of our score fusion procedure. Each column is a tracking frame. The top to the bottom rows illustrate the score map of fusion, ground truth, and two negative templates, respectively. We see that the ground truth score map not only responds to the tracked target but also increases responses in many other areas where the tracker is easily misled. After implementing our fusion, the unrelated responses are sufficiently suppressed.

4. Evaluation

To evaluate the performance of our method, we tested it on two collections of datasets: the UGV dataset and a public dataset. The UGV dataset includes 17 image sequences of outdoor person-following tasks recorded by a small unmanned ground vehicle (UGV). The purpose of the person-following system is to reduce the workload of the teleoperator. To conduct a more comprehensive evaluation, we further selected 27 image sequences that involve person tracking from the OTB and UAV [29] datasets. Unlike other popular public tracking datasets, the sequences of the UAV dataset were captured from an aerial viewpoint of low-altitude UAVs.

In the experiments, we applied the designed algorithm to two representative Siamese trackers: SiamRPN and SiamRPN++. SiamRPN adopts AlexNet as the backbone and takes the feature of the final layer for the correlation. SiamRPN++ uses ResNet50 as the backbone and outputs features by fusing the outputs of multiple layers. We applied our framework to these two approaches and observed the performance improvement. The applied networks and pre-trained weights were obtained from <https://github.com/STVIR/pysot> (accessed in 20 April 2021). We further applied DaSiamRPN (<https://github.com/foolwood/DaSiamRPN>, accessed in 20 April 2021) and its update-based variation UpdateNet (<https://github.com/zhanglichao/updatenet>, accessed in 20 April 2021) for comparisons. Therefore, the SiamRPN and our improvement, SiamRPN++ and our improvement, and DaSiamRPN and UpdateNet shared weights respectively and can be seen as three comparing groups.

As the evaluation method, we used one pass evaluation (OPE) [11]. The OPE criterion scores tracker performance using center location error and the bounding box overlap, which yield a precision plot and a success plot according to the threshold, respectively. The success plots are calculated as the percentage of frames with an intersection-over-union (IOU) overlap exceeding a threshold and scored using the area under the curve (AUC) score.

Since our approach provides improvements in terms of robustness instead of localization accuracy, the precision plots were omitted.

The experiments were conducted on a desktop with an NVIDIA RTX3090 GPU and an Intel i7 CPU. We set the number of GMM fitting components to six. The KL divergence threshold of setting adjacent matrix was two. The threshold parameter $\tau_t = 0.19$, $N_{pos} = 2$, $N_{neg} = 3$.

4.1. UGV Dataset

The UGV dataset is a self-constructed dataset that contains images from a small unmanned ground vehicle that performed person-following tasks in outdoor environments. The robot was following a single-person target in a campus environment under varying road conditions (e.g., brick roads, cement roads, snowy roads, and grasslands) and illumination conditions (backlight, shadow, dawn, and night). The vehicle performed servo moving in accordance with the relative position of the tracked target. We set the target person being followed to pose different challenging situations such as teams wearing similar clothes, partial and full occlusion, etc. The images were collected by an Intel Realsense D435i camera that was rigidly attached to the robot. The camera collected images at 30 fps in the following tasks. We downsampled the frame rate to 10 fps in our dataset. The image resolution is 640×480 pixels. The robot was following the target person up to speeds of 2 m/s.

The dataset contains 17 image sequences that vary in the appearance of the tracking targets, the appearance and number of distractors, road conditions, weather, and experiment duration. The detailed information of each subset is provided in Table 1. The distractor information states the attributes of different subsets, including campus environments with pedestrians (PED), the number of pedestrians with different-colored clothes actively interfering (#DAI), number of pedestrians with similar-colored clothes actively interfering (#SAI), illumination variation (IV), and low illumination (LI).

Table 1. Detailed information of the UGV dataset.

Set Name	Road Condition	Distractor Info	Duration
UGV01	Brick Road	PED	25 s
UGV02	Brick Road	PED	19 s
UGV03	Asphalt Road	PED	20 s
UGV04	Asphalt Road	1DAI	184 s
UGV05	Cement Road	1SAI	325s
UGV06	Asphalt Road	IV,1SAI	247 s
UGV07	Asphalt Road	1DAI	423 s
UGV08	Snowy Asphalt Road	1DAI,1SAI	529 s
UGV09	Grassland	PED	53 s
UGV10	Grassland	2DAI	76 s
UGV11	Grassland & Asphalt Road	PED	301 s
UGV12	Snowy Asphalt Road	PED	201 s
UGV13	Cement Road	LI	206 s
UGV14	Cement Road & Grassland	LI	82 s
UGV15	Cement Road & Grassland	LI	76 s
UGV16	Cement Road & Grassland	LI,PED	537 s
UGV17	Cement Road & Grassland	LI,DAI	146 s

4.1.1. Results and Analysis

The results are divided into two groups. The short-term group presents the results of the sequences that are less than 100 s. Furthermore, the long-term group gives the results of the rests. Figure 5 illustrates the success plots of short-term tasks. Figure 6 presents the qualitative results of UGV02, UGV03, and UGV14.

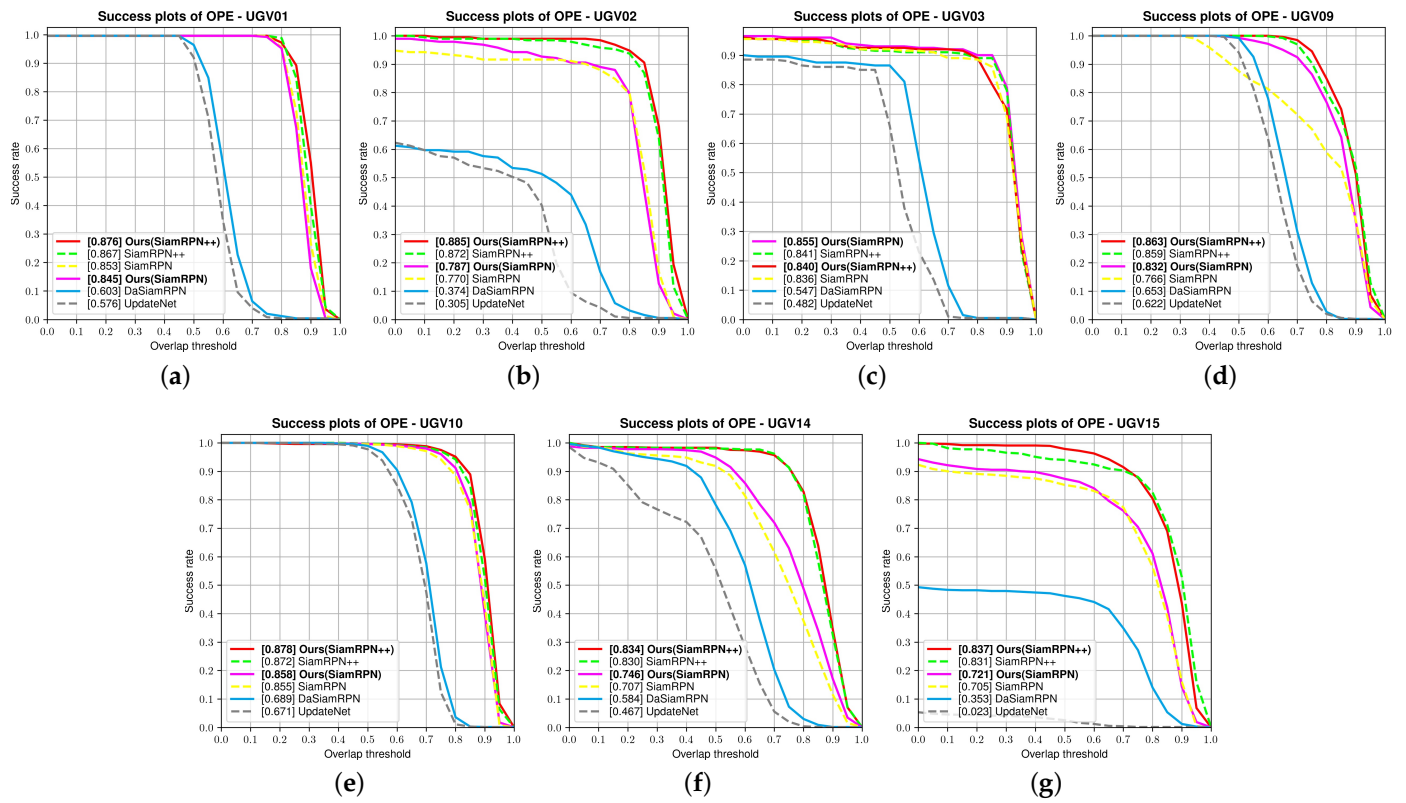


Figure 5. Success plots of the short-term following.

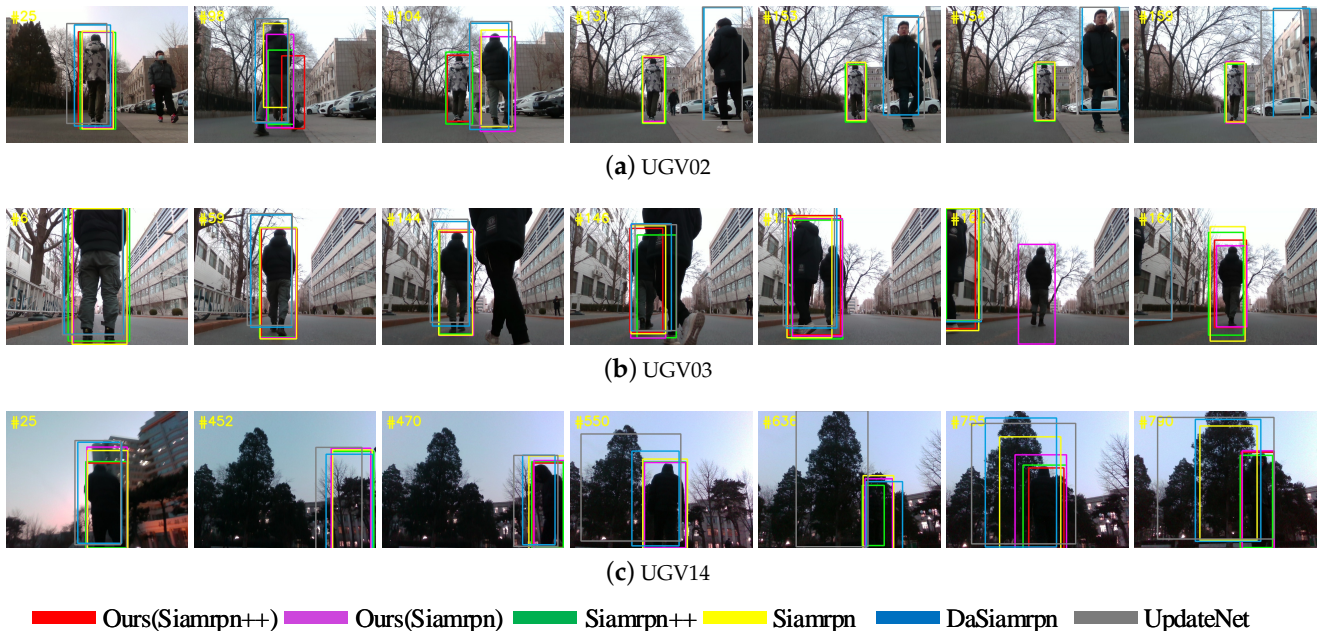


Figure 6. Qualitative results of short-term tasks.

For UGV02 and UGV03, we can see that the tracking boxes usually drift when a distracting person passes through and temporarily occludes the target. For UGV14, the variation in light severely impacts the trackers' performance. In general, our approach performs well and provides improvements in both AUC scores and robustness.

Figure 7 presents the success plots of long-term following tasks. Our approach ranks high amongst all considered methods. Without fine-tuning the network, the outcomes of

DaSiamRPN and UpdateNet are poor on our dataset. UpdateNet, which is the update-based variation of DaSiamRPN, does not provide an improvement over DaSiamRPN. Despite the already excellent performance of SiamRPN and SiamRPN++, the AUC score of our approach is improved. The qualitative results are provided in Figure 8.

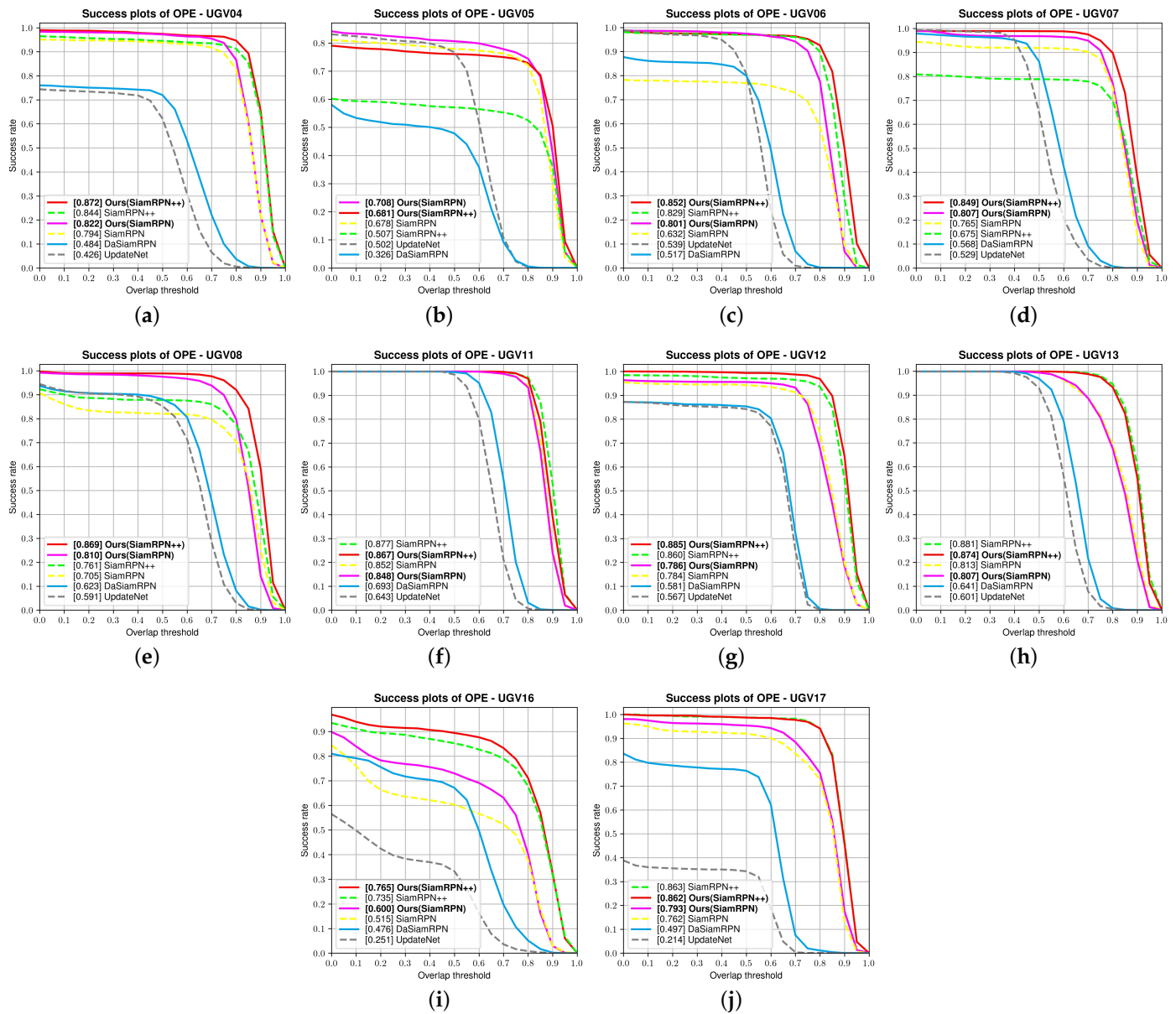


Figure 7. Success plots of long-term following.

In the robot following situations, the robot moves accordingly based on the target location command given by the tracking result: if the tracker tracks the wrong target, the real target will soon be out of view, leading to failure of the following mission. Even though the tracker sometimes does not return a precise bounding box of the following target, a rough result (IOU > 0) can still maintain the target within the tracking view, allowing a chance to recover the target. Therefore, we define tracking robustness as the percentage of the bounding box that satisfies IOU > 0 (namely, the value of the success rate when the overlap threshold = 0 in the success plots). In our opinion, discussing tracking robustness in the following missions is even more meaningful than the AUC score. Since the SiamRPN-based approaches regress bounding boxes from pre-defined anchors, the size adjustment is minor; we demonstrate that the robustness criterion will not be biased by

the large-area bounding boxes. As shown in the success plots, our approach provides a substantial gain in terms of tracking robustness compared to the base trackers.



Figure 8. Qualitative results.

For UGV04 (Figure 7a) and UGV07 (Figure 7d), even though the target and distracting person have sufficient disparity in appearance (Figure 8a,d), the tracking box drifts to the distracting person frequently after crossing. By eliminating the responses of the distractors, our approach provides a substantial improvement compared to the other approaches.

For UGV05 and UGV06, the target and distractor person dress similarly (both wearing a black coat), and the following tasks were conducted under intense light variation (see Figure 8b,c). With the light change, the target appearance varies significantly. The competing trackers were impacted and their results presented a random pattern. By continuing to obtain the latest positive templates, our approach distinguishes the target and distractor more robustly, resulting in stable and better performance.

In UGV08 (Figure 7e), the following was conducted on a cloudy day, without light variation; our approach produced stable results and outperformed the others even with a similar-appearance distracting person.

UGV16 (Figure 7i) and UGV17 (Figure 7j) were conducted in the evening (Figure 8f,g); similar to UGV14 and UGV15, the success rates of the other approaches were heavily decayed. Our approach exhibited its strength in these situations, where both AUC score and tracking robustness outperformed the corresponding approaches.

For other UGV tasks, our approach yielded better or comparable results.

4.1.2. Statistical Significance

The test results above present improvements of our approaches over their base approaches in general cases. We further perform a statistical test to see if the improvements are statistically significant. Specifically, we set the null hypothesis as the subtraction of the paired data comes from a normal distribution with mean equal to zero and unknown variance. Then, the paired-sample *t*-test is employed. If the *p* value falls below 0.05 significance level, the null hypothesis is rejected or accepted otherwise.

Table 2 presents the statistical significance condition and the corresponding *p* value of the three comparing groups. The results show that the improvements are statistically significant.

Table 2. Statistical significance of three comparative approaches.

	DaSiamRPN vs. UpdateNet	SiamRPN vs. Ours (SiamRPN)	SiamRPN++ vs. Ours(SiamRPN++)
Statistical Significance	Yes	Yes	Yes
<i>p</i> value	0.0249	0.0048	0.0241

4.1.3. Compare with State-of-the-Art

We additionally compare our approaches with several latest state-of-the-art approaches: SiamCAR [19], SiamGAT [21], SiamBAN [20], DiMP [30], and PrDiMP[31]. The overall success plot of the dataset is depicted in Figure 9.

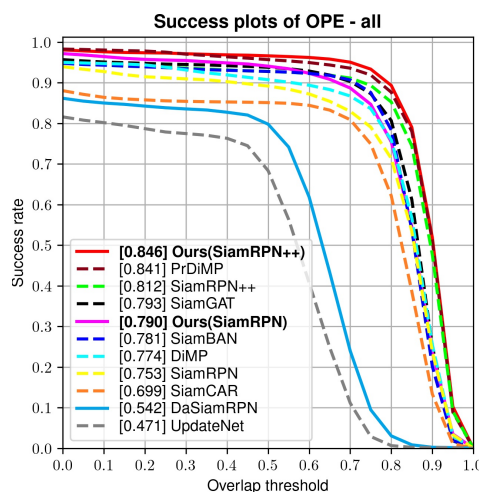


Figure 9. Overall success plot of the state-of-the-art approaches.

Compared with the base methods, our approaches bring substantial gain in terms of both AUC score and robustness. Among all 11 competing methods, our two approaches ranked first and fifth respectively in AUC scores; in terms of robustness, they ranked second and third respectively. The PrDiMP method also shows good tracking performance in the dataset.

4.2. Public Dataset

All approaches are further tested on a dataset composed of 27 public datasets involving person tracking. The selected sequences, their source, and the results of tracking robustness are listed in Table 3. The uparrow and downarrow indicate the relative improvement provided by our approach compared to the base approaches. The red, green, and blue denote the methods that ranked first, second, and third in the experimental results, respectively.

Table 3. The tracking robustness on public datasets. The uparrow and downarrow indicate the relative improvement. The red, green, and blue denote the methods that ranked first, second, and third in the experimental results.

Source	Set Name	DaSiamRPN vs. UpdateNet	SiamRPN vs. Ours (SiamRPN)	SiamRPN++ vs. Ours (SiamRPN++)	SiamCAR & SiamGAT	DiMP & PrDiMP	SiamBAN
OTB100	Girl2	9.20/13.53	70.60 / 95.67 ↑	85.27 / 89.67 ↑	89.53 / 97.20	61.47 / 95.40	27.73
	Human3	2.12 / 2.12	76.15 / 93.88 ↑	5.59 / 94.35 ↑	27.21 / 99.53	2.18 / 90.81	2.00
	Human4-2	51.57 / 51.57	52.02 / 82.31 ↑	90.85 / 98.50 ↑	51.72 / 64.92	99.85 / 56.52	85.61
	Jogging-1	98.05 / 98.37	95.11 / 100.00 ↑	96.74 / 96.74 -	96.09 / 97.39	98.05 / 99.35	91.53
	Walking	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.76 / 99.76	99.76 / 99.76	99.76
	Walking2	55.80 / 55.80	65.40 / 76.00 ↑	67.00 / 72.20 ↑	99.80 / 53.60	50.00 / 72.10	53.60
	Woman	100.00 / 95.81	98.49 / 99.33 ↑	100.00 / 100.00 -	99.16 / 99.66	67.84 / 100.00	99.83
UAV123	bike1	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	85.67 / 99.97	99.97 / 99.97	99.97
	group1_1	73.07 / 73.14	73.07 / 72.69 ↓	98.65 / 88.92 ↓	99.47 / 95.27	99.92 / 99.85	99.10
	group1_2	54.69 / 81.15	56.55 / 94.84 ↑	34.49 / 98.99 ↑	94.84 / 99.75	99.92 / 99.92	81.15
	group1_3	86.96 / 82.85	66.83 / 69.81 ↑	82.71 / 87.03 ↑	53.65 / 70.80	57.69 / 98.37	88.80
	group1_4	65.65 / 67.23	83.35 / 73.45 ↓	93.89 / 79.13 ↓	53.32 / 88.62	99.89 / 99.79	75.87
	group2_1	10.69 / 10.69	10.69 / 10.69 -	10.69 / 10.69 -	10.69 / 10.69	10.80 / 10.69	10.69
	group2_2	98.03 / 98.03	58.73 / 98.03 ↑	97.92 / 98.03 ↑	97.92 / 97.92	97.92 / 97.92	92.60
	group2_3	81.05 / 79.85	81.93 / 82.37 ↑	80.61 / 82.26 ↑	82.37 / 82.37	82.26 / 69.66	56.85
	group3_1	100.00 / 2.87	100.00 / 100.00 -	99.81 / 100.00 ↑	99.94 / 99.68	99.94 / 99.94	99.94
	group3_2	21.25 / 21.25	21.25 / 84.54 ↑	21.25 / 87.63 ↑	21.25 / 21.25	21.25 / 21.25	21.02
	group3_3	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.94 / 99.94	99.94 / 99.94	94.82
	group3_4	42.54 / 42.54	42.54 / 83.69 ↑	42.54 / 71.35 ↑	42.54 / 93.62	42.54 / 90.85	42.54
	person11	36.62 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.86 / 99.86	39.67 / 99.86	99.86
	person18	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.86 / 99.93	99.93 / 99.93	99.93
	person19_1	86.48 / 87.13	85.92 / 92.36 ↑	86.48 / 92.36 ↑	87.05 / 86.48	86.48 / 92.20	92.28
	person19_2	42.87 / 42.87	42.87 / 94.84 ↑	94.84 / 94.84 -	94.58 / 94.77	42.87 / 94.77	94.77
	person20	100.00 / 100.00	100.00 / 100.00 -	93.33 / 96.80 ↑	94.11 / 99.94	100.00 / 100.00	98.09
person4_1	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.93 / 99.93	99.93 / 99.93	99.93	
person4_2	100.00 / 100.00	100.00 / 100.00 -	100.00 / 100.00 -	99.92 / 99.92	99.92 / 99.92	99.92	
person9	47.66 / 19.82	18.76 / 94.25 ↑	63.99 / 88.96 ↑	94.10 / 18.76	19.21 / 94.10	6.96	
Overall	Mean	69.05 / 67.65 ↓	74.08 / 88.85 ↑	79.51 / 89.94 ↑	80.52 / 84.13	73.30 / 88.26	74.63
	Statistical Significance	No	Yes	Yes	-	-	-

From the table, in most of the subsets, our approach provides extra improvement compared to the base approaches. In general, the mean tracking robustness increases by 14.77% and 10.43% compared to SiamRPN and SiamRPN++, respectively. Furthermore, the improvements are statistically significant. Among all the comparison methods, the average robustness of our proposed approaches ranks first and second respectively. In addition, except for group2_1, our approaches converge to high robustness in all datasets, showing strong target capture ability of our approaches.

In some subsets, the tracking performance is heavily impacted by the long-term or large-scale occlusion of roofs, buildings, and trees; the anti-occlusion strategy of the aerial viewpoint needs to be further explored.

5. Conclusions

In this paper, a robust visual tracking approach aiming to enable robot person-following tasks was proposed. To solve the problem where fixed templates cannot adapt to the robustness demand of long-term tracking, a multi-templates tracking method was proposed. The confident templates and distract templates are yielded during tracking leveraging the distribution of the central responses. By merging the responses of ground-truth templates, confident templates, and distract templates, the responses of target-tracking are enhanced and false responses are suppressed, leading to robust tracking. The proposed method was incorporated into two state-of-the-art approaches, SiamRPN and SiamRPN++, and validated on a robot person-following dataset as well as a collection of public person-tracking datasets. The results showed that our approaches outperform their base approaches in terms of both AUC score and tracking robustness. Furthermore, the approaches are compared with seven state-of-the-art methods. In the UGV dataset, among 11 approaches, they rank first and fifth in terms of AUC score and second and third in terms of tracking robustness. In the public dataset, they rank first and second.

Based on the proposed robust visual tracking approach, in the future, we will continue to explore human–robot interaction and failure recovery methods to construct an autonomous and control terminal-free person following system, so that it can be applied to facilitate police patrols, factory manufacturing, and other scenarios.

Author Contributions: Conceptualization, M.C.; data curation, L.M.; funding acquisition, J.W.; methodology, M.C.; project administration, J.W.; software, M.C.; visualization, M.C. and L.M.; writing—original draft, M.C.; writing—review and editing, M.C., J.W., and L.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Defense Industrial Technology Development Program (JCKY2019602C015).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: Not applicable.

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

References

1. Islam, M.J.; Hong, J.; Sattar, J. Person-following by autonomous robots: A categorical overview. *Int. J. Robot. Res.* **2019**, *38*, 1581–1618. [[CrossRef](#)]
2. Schlegel, C.; Illmann, J.; Jaberg, H.; Schuster, M.; Wörz, R. Vision based person tracking with a mobile robot. In Proceedings of the Ninth British Machine Vision Conference (BMVC), Southampton, UK, 14–17 September 1998; pp. 418–427.
3. Hu, C.; Ma, X.; Dai, X. A robust person tracking and following approach for mobile robot. In Proceedings of the 2007 International Conference on Mechatronics and Automation, Harbin, China, 5–8 August 2007; pp. 3571–3576.
4. Shin, J.; Kim, S.; Kang, S.; Lee, S.W.; Paik, J.; Abidi, B.; Abidi, M. Optical flow-based real-time object tracking using non-prior training active feature model. *Real-Time Imaging* **2005**, *11*, 204–218. [[CrossRef](#)]
5. Koide, K.; Miura, J. Identification of a specific person using color, height, and gait features for a person following robot. *Robot. Auton. Syst.* **2016**, *84*, 76–87. [[CrossRef](#)]
6. Satake, J.; Chiba, M.; Miura, J. A SIFT-based person identification using a distance-dependent appearance model for a person following robot. In Proceedings of the 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guangzhou, China, 11–14 December 2012; pp. 962–967.
7. Kwolek, B. Person following and mobile camera localization using particle filters. In Proceedings of the Fourth International Workshop on Robot Motion and Control (IEEE Cat. No. 04EX891), Puzszykow, Poland, 20–20 June 2004; pp. 265–270.
8. Satake, J.; Miura, J. Robust stereo-based person detection and tracking for a person following robot. In Proceedings of the ICRA Workshop on People Detection and Tracking, Kobe, Japan, 12–17 May 2009; pp. 1–10.
9. Chen, B.X.; Sahdev, R.; Tsotsos, J.K. Person following robot using selected online ada-boosting with stereo camera. In Proceedings of the 2017 14th conference on computer and robot vision (CRV), Edmonton, AB, Canada, 16–19 May 2017; pp. 48–55.
10. Wang, M.; Liu, Y.; Su, D.; Liao, Y.; Shi, L.; Xu, J.; Miro, J.V. Accurate and real-time 3-D tracking for the following robots by fusing vision and ultrasonic information. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 997–1006. [[CrossRef](#)]

11. Wu, Y.; Lim, J.; Yang, M.H. Online Object Tracking: A Benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013.
12. Kristan, M.; Leonardis, A.; Matas, J.; Felsberg, M.; Pfugfelder, R.; Čehovin Zajc, L.; Vojir, T.; Bhat, G.; Lukežič, A.; Eldesokey, A.; et al. The sixth Visual Object Tracking VOT2018 challenge results. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
13. Zhang, L.; Gonzalez-Garcia, A.; Weijer, J.V.D.; Danelljan, M.; Khan, F.S. Learning the Model Update for Siamese Trackers. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019.
14. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.
15. Li, B.; Wu, W.; Wang, Q.; Zhang, F.; Xing, J.; Yan, J. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 4282–4291.
16. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 850–865.
17. Zhu, Z.; Wang, Q.; Li, B.; Wu, W.; Yan, J.; Hu, W. Distractor-aware siamese networks for visual object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 101–117.
18. Zhang, Z.; Peng, H. Deeper and wider siamese networks for real-time visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4591–4600.
19. Guo, D.; Wang, J.; Cui, Y.; Wang, Z.; Chen, S. SiamCAR: Siamese fully convolutional classification and regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6269–6277.
20. Chen, Z.; Zhong, B.; Li, G.; Zhang, S.; Ji, R. Siamese Box Adaptive Network for Visual Tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6668–6677.
21. Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; Shen, C. Graph Attention Tracking. In Proceedings of the The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 19–25 June 2021.
22. Voigtlaender, P.; Luiten, J.; Torr, P.H.; Leibe, B. Siam r-cnn: Visual tracking by re-detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6578–6588.
23. Dai, K.; Zhang, Y.; Wang, D.; Li, J.; Lu, H.; Yang, X. High-performance long-term tracking with meta-updater. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6298–6307.
24. Wang, J.; Yang, H.; Xu, N.; Wu, C.; Zhao, Z.; Zhang, J.; Wu, D.O. Long-term target tracking combined with re-detection. *EURASIP J. Adv. Signal Process.* **2021**, *2021*, 2. [[CrossRef](#)]
25. Li, D.; Yu, Y.; Chen, X. Object tracking framework with Siamese network and re-detection mechanism. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 261. [[CrossRef](#)]
26. Zhang, J.; Liu, Y.; Liu, H.; Wang, J. Learning Local–Global Multiple Correlation Filters for Robust Visual Tracking with Kalman Filter Redetection. *Sensors* **2021**, *21*, 1129. [[CrossRef](#)] [[PubMed](#)]
27. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B (Methodol.)* **1977**, *39*, 1–22.
28. Biemann, C. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. In Proceedings of the TextGraphs: The First Workshop on Graph Based Methods for Natural Language Processing, Morristown, NJ, USA, 9 June 2006; pp. 73–80.
29. Mueller, M.; Smith, N.; Ghanem, B. A benchmark and simulator for uav tracking. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 445–461.
30. Bhat, G.; Danelljan, M.; Gool, L.V.; Timofte, R. Learning discriminative model prediction for tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–20 June 2019; pp. 6182–6191.
31. Danelljan, M.; Gool, L.V.; Timofte, R. Probabilistic regression for visual tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 7183–7192.