algorithms

MDPI

Article

Segmentation and Tracking Based on Equalized Memory Matching Network and Its Application in Electric Substation Inspection

Huanlong Zhang [1,*], Bin Zhou [1], Yangyang Tian [2] and Zhe Li [2]

[1] College of Electric and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China; binzhou4968@163.com
[2] State Grid Henan Electric Power Research Institute, Zhengzhou 450000, China; tianyangyang199306@163.com (Y.T.); lizhe4@ha.sgcc.com.cn (Z.L.)
* Correspondence: zzuli407@163.com

Abstract: With the wide application of deep learning, power inspection technology has made great progress. However, substation inspection videos often present challenges such as complex backgrounds, uneven lighting distribution, variations in the appearance of power equipment targets, and occlusions, which increase the difficulty of object segmentation and tracking, thereby adversely affecting the accuracy and reliability of power equipment condition monitoring. In this paper, a pixel-level equalized memory matching network (PEMMN) for power intelligent inspection segmentation and tracking is proposed. Firstly, an equalized memory matching network is designed to collect historical information about the target using a memory bank, in which a pixel-level equalized matching method is used to ensure that the reference frame information can be transferred to the current frame reliably, guiding the segmentation tracker to focus on the most informative region in the current frame. Then, to prevent memory explosion and the accumulation of segmentation template errors, a mask quality evaluation module is introduced to obtain the confidence level of the current segmentation result so as to selectively store the frames with high segmentation quality to ensure the reliability of the memory update. Finally, the synthetic feature map generated by the PEMMN and the mask quality assessment strategy are unified into the segmentation tracking framework to achieve accurate segmentation and robust tracking. Experimental results show that the method performs excellently on real substation inspection scenarios and three generalized datasets and has high practical value.

Keywords: substation intelligent inspection; deep learning; object segmentation and tracking; feature matching; memory updating

Citation: Zhang, H.; Zhou, B.; Tian, Y.; Li, Z. Segmentation and Tracking Based on Equalized Memory Matching Network and Its Application in Electric Substation Inspection. *Algorithms* **2024**, *17*, 203. https://doi.org/10.3390/a17050203

Academic Editor: Abdulsalam Yassine

Received: 17 April 2024
Revised: 4 May 2024
Accepted: 7 May 2024
Published: 10 May 2024

1. Introduction

The inspection of power equipment plays a pivotal role in ensuring the secure and stable operation of the power grid. The use of deep learning in power systems such as wind power prediction [1], voltage stability assessment [2], power grid synthesis [3], solar power system control [4], power intelligent inspection of [5], etc. has grown rapidly in recent years.

During the intelligent inspection in substation scenarios, inspection technologies based on UAV [6,7] and robots [8] can utilize computer vision technology for the recognition of appearance defects in substation equipment (e.g., transformers, insulators, meters, etc.), which can greatly alleviate the dangers to staff in high-risk environments such as substations. Meanwhile, the quality and efficiency of inspections in hazardous environments are improved. It requires the tracking and identification of power equipment targets in substations, and realizing real-time monitoring and analysis of equipment status. Therefore, the study of accurate and robust tracking algorithms is of great significance for substation

inspection. However, for the power equipment targets in the inspection video, the complex background, uneven light distribution, large change amplitude, occlusion, and other factors of interference, resulting in power equipment segmentation tracking accuracy, has not yet reached the requirements of the production needs.

To overcome the above problems, some trackers [9,10] employ advanced template updating mechanisms to enhance their robustness. Recently, memory network has been effectively utilized in object tracking [11–15]. The tracking method [11] introduces a method that utilizes a combination of long-term and short-term memory to learn multiple adaptive correlation filters, enhancing the robustness of object tracking. A dynamic memory network [12,13] is proposed that dynamically adjusts the template to accommodate variations in the target's appearance during the tracking process. In [14], a dual-memory selection (DMS) model is presented for reliable visual tracking, which effectively reduces tracking drift. By matching the reference frames with the query frame, STMTrack [15] transfers the reference frame information to the query frame, thereby directing the attention of the tracker to the most informative region of the current frame.

In order to obtain the optimal match between the query frame and the reference frame in the memory network, VideoMatch [16] constructs templates for foreground and background based on the information from the initial frame, and utilizes the soft matching layer to generate similarity scores. Building upon VideoMatch, FEELVOS [17] and CFBI [18] incorporate information from the initial frame and the previous frame to generate templates. To maximize the utilization of historical frames for prediction purposes, STM [19] retrieves memory by employing non-local and dense memory matching techniques, and employs the retrieved memory to locate the target in the query. Extended from STM, episodic graph memory networks [20] are exploited to update memory segmentation models. NPMCA-net [21] utilizes a combination of non-local techniques and mask propagation to accurately localize foreground targets by comparing the pixels of reference and target frames. All of the above methods belong to surjective matching, also known as one-to-many matching, which only considers the query frame options without addressing reference frame options. This matching mechanism offers flexibility as there are no limitations on the matching process. Consequently, it enables effective handling of visually diverse frames but is vulnerable to potential background distractions.

In order to prevent the problem, KMN [22] proposes a kernelized memory network, where the non-local matching between its query and memory is controlled by a Gaussian kernel that is generated by an argmax operation along the query frames, making the memory network more efficient for VOS. However, it is still surjective matching, since all pixels of the query frame can refer to the reference frames. HMMN [23] addresses the problem by introducing a kernel-guided memory matching module. Unlike the one used in KMN, it imposes temporal smoothness constraints, which is an important cue for the VOS. Additionally, a top-k guided memory matching method is present to bootstrap the prediction of more accurate target masks.

Unlike the above works, BMVOS [24] provides a bijective matching mechanism that prevents both reference frame pixels from being referenced, and, therefore, considers the best match of the reference frame pixels first. In other words, bijective matching considers both the reference frame and query frame to find the best match, and connects the pixels only when they are definite matches to each other. Thus, it can effectively eliminate background interference. Although bijective matching can solve the limitation of surjective matching, it can only replace surjective matching in the testing phase. The reason is that it is based on discrete functions (argmax or top K operation), which has an impact on stable network training. Moreover, the hyperparameters need to be carefully adjusted manually during testing.

Inspired by the work mentioned above, we propose a segmentation tracking algorithm based on pixel-level equalized memory matching network (PEMMN) for intelligent inspection of substations. The main contributions of this work are as follows:

(1) An equalized memory matching network is designed, which stores historical information of the target through a memory bank and utilizes a pixel-level equalized matching method to ensure that the detailed information of the reference frames is efficiently delivered to the current frame, so that the segmentation tracker can focus on the most informative region in the current frame.
(2) To avoid excessive consumption of memory resources and accumulation of erroneous segmentation templates, a memory storage and update strategy is designed to filter and store frames with high segmentation quality to ensure that the process of updating the memory bank is both accurate and reliable.
(3) The synthetic feature map generated by the PEMMN and the mask quality assessment strategy are unified into the segmentation tracking framework to achieve accurate segmentation and robust tracking.
(4) Experimental results show that the method performs well on both real videos of substation inspection scenarios and commonly used benchmark datasets.

2. Pixel-Level Equalized Memory Matching Network

Our method, the pixel-level equalized memory matching network (PEMMN) for segmentation tracking algorithms, will be described in detail in this section.

2.1. Feature Similarity Matching

In semi-supervised video target segmentation methods, only the segmentation mask of the first frame is given, and the target is segmented in all the remaining frames of the video. The existing SVOS methods are most often based on feature similarity matching, in which embedded features of the reference frame are compared with those of the query frame. It is necessary for VOS to perform pixel-level feature matching to ensure that the information located in localized regions is captured. We first review the process of feature similarity matching.

The image of frame i and its features are denoted by $I^i \in R^{H0\times W0\times 3}$ and $X^i \in R^{H\times W\times C}$, respectively. The predicted mask and the mask after downsampling are denoted by $M^i \in R^{H0\times W0\times 2}$ and $m^i \in R^{H\times W\times 2}$, containing the background and foreground channels. Given X^i, X^k, and m^k, M^i can be obtained, where $k \in [0,\, i-1]$ refers to a reference frame.

In order to mine the relation between reference frame and query frame, that is, to obtain the best match between them, the reference frame mask is transferred to the query frame, and the query frame mask is predicted. First, similarity computation is performed. If p and q are the positions of individual spatial pixels in the reference frame and query frame features, then the similarity score can be expressed as follows:

$$S(p,q) = \frac{N(X_p^k)\cdot N(X_q^k)+1}{2}, \tag{1}$$

where "$\cdot$" and N indicate the matrix inner product and channel L2 normalisation, respectively. In order to be in the same range as the scores of the mask, linear normalisation is applied to transform the similarity scores to be in the range of 0 to 1. Then the similarity matrix $S \in R^{H'W'\times HW}$ can be calculated.

After reshaping the reference frame mask, the similarity matrix S is multiplied with it to enhance the target information, as in (2).

$$\begin{aligned} S_{BG} &= S \otimes M_0^k \\ S_{FG} &= S \otimes M_1^k \end{aligned} \tag{2}$$

where $\otimes$ denotes Hadamard product. Next, the S_{BG} and S_{FG} are subjected to a query maximization operation to obtain the foreground and background matching scores. By concatenating them along the channel, the final matching score map of frame i can be represented as $S^i \in [0,1]^{H\times W\times 2}$. The visualization flowchart is shown in Figure 1a.

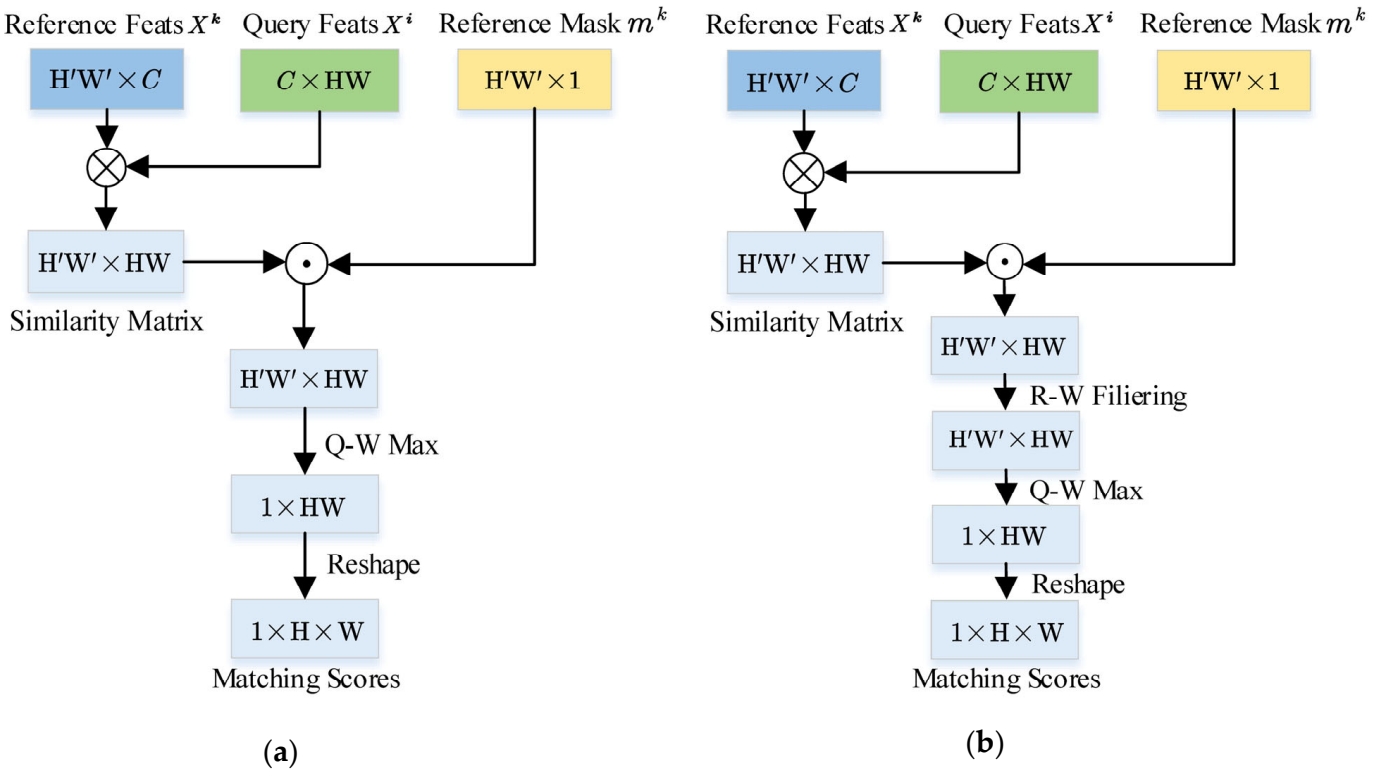

Figure 1. Pipelines of (**a**) surjective matching method and (**b**) bijective matching method.

From the above surjective matching process, it can be seen that the query frame matches some pixels of specific spatial positions in the reference frame, without considering the best match with the reference frame. Thus, there will be cases where some reference frame pixels are not referenced, and some pixels are referenced multiple times. The method is robust to scale and appearance variations but is also susceptible to background interference.

Bijective matching differs from traditional surjective matching by conducting a reference-wise top K operation before identifying the optimal matches between the query frame and the reference frame, thereby eliminating background distractors from the selection process, as shown in Figure 1b.

To ensure that all connected pixels are uniformly distributed, only the largest K scores in the query frame are maintained, and the rest are excluded. Replace the similarity score of the missing connection with the minimum value in each pixel of the reference frame. The query frame can refer to the reference frame if and only if the query frame is selected by at least one reference frame pixel. This shows that, compared with surjective matching, the implementation of the bijective matching algorithm is more strict so as to ensure the information transfer between the two frames is more reliable. Bijective methods can reflect the capability of the reference frame in feature similarity scores but can only replace surjective matching in the testing phase because it is based on a discrete function, which is not conducive to stable network training. Moreover, it requires a manual adjustment of parameters to achieve the best results.

2.2. Pixel-Level Equalized Memory Matching

Due to these limitations of existing bijective matching methods, an equalized matching mechanism is introduced, which is completely independent of surjective matching but can still catch bijective matching. It can function both independently as a bold branch and simultaneously with surjective matching. Similarly to the matching mechanism above, the feature similarity between p and q is calculated first by using the following formula:

$$S(p,q) = X_p^k \cdot X_q^i. \tag{3}$$

Then, the similarity matrix S is calculated as described above. To represent bijection in the similarity matrix S, the softmax operates along the query dimension as follows:

$$S_p \leftarrow Softmax(S_p). \tag{4}$$

From the above process, for each pixel in the reference frame, the scores of all pixels in the query frame are normalized such that their sum becomes 1. This ensures that all reference frame pixels contribute equally to the prediction of the query frame. Therefore, if a reference frame pixel is referenced multiple times, its score will be reduced proportionally as they have to share the same total score. Recognizing that chaotic referencing of these pixels can introduce significant errors, it is effective to strictly suppress their matching scores to minimize visual interference. Once the similarity matrix S is modulated accordingly, the remaining process follows a surjective matching approach, as shown in Figure 2.

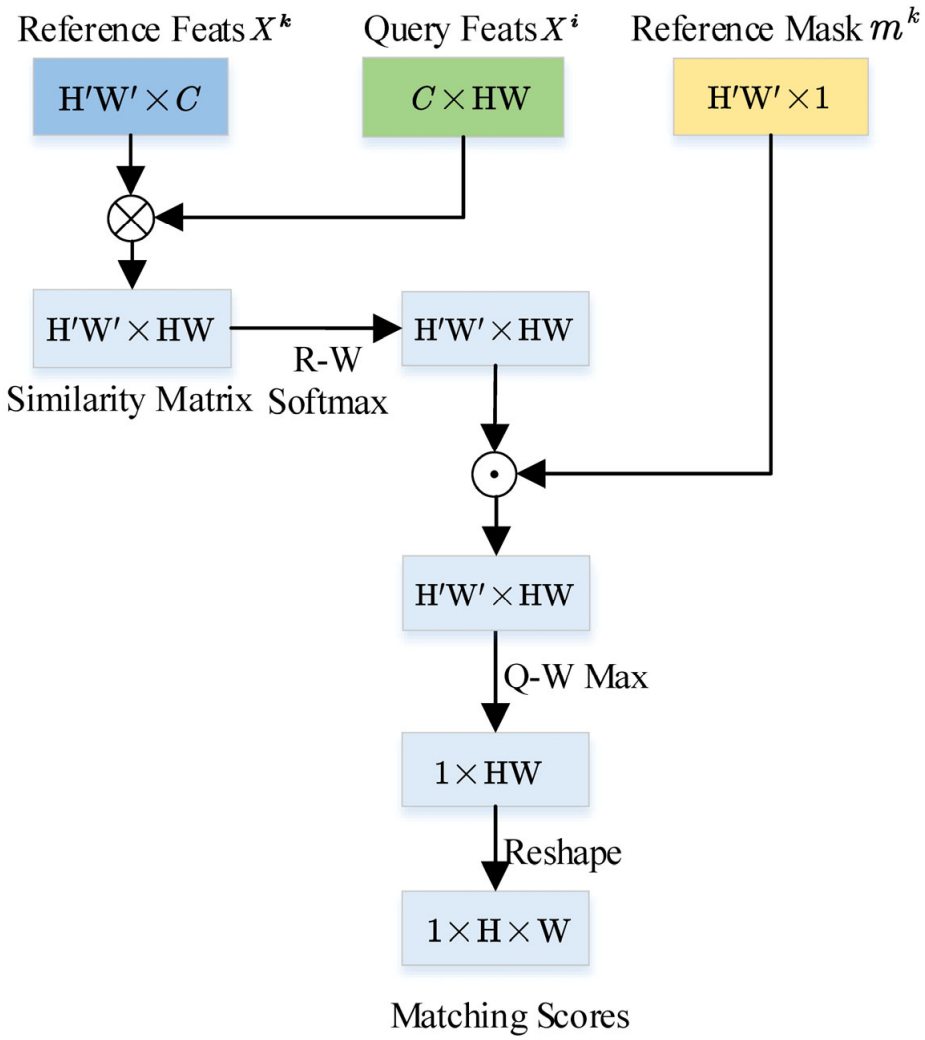

Figure 2. Pipelines of the equalized memory matching method.

2.3. Pixel-Wise Memory Storage and Update

To prevent memory explosion and accumulation of segmentation template errors, inspired by [25], a mask quality evaluation module (MQEM) is introduced for evaluating the segmentation results of each frame and determining whether the frame can be added as a reference frame to the memory library. Since the hardware cannot withstand the increasing memory requirements, when the memory reaches a certain limit, we perform dynamic updates to prevent memory explosion problems.

The model consists of a fractional encoder, four convolutional layers, and two fully connected (FC) layers. The same feature extraction network as the memory network can be used to share the convolutional network parameters. Query image $I^i \in R^{H0\times W0\times 3}$ and its segmentation mask $M^i \in R^{H0\times W0\times 2}$ act as inputs to the evaluation model, and the feature map $f^i \in R^{H0/16\times W0/16\times C}$ is obtained. f^i is then fed into the convolutional layers followed by the FC layer, and the fraction S^A is finally output. The process is as follows.

$$\begin{aligned} f^i &= Enc(I^i \oplus M^i) \\ S^A &= Fc(Conv(f^i)) \end{aligned} \tag{5}$$

where $\oplus$ indicates the concat operation, and i denotes the index of the current query frame.

With the quality evaluation module, the memory network can optionally add those frames whose mask quality score is greater than the threshold σ to ensure that the stored information is more accurate and reliable and improve the efficiency and quality of information storage. Therefore, the negative impact of noisy data on memory network is avoided.

To prevent memory explosions, the size of the storage needs to be limited and dynamically updated to accommodate new scenarios. The memory is dynamically updated when it reaches a particular storage limit so that different video scenes can be handled. As a measure of consistency with time between each reference frame and the current frame, we calculate the time consistency score S^C in the following manner:

$$S_k^C = e^{-|i-k|}, \tag{6}$$

where k and i represent the index of a reference frame and current frame, respectively.

Using the time consistency score S^C and the accuracy score S^A, we can calculate the score for each reference frame in the memory bank as follows:

$$S_k^R = S_k^A + S_k^C, \tag{7}$$

where k is the index of a reference frame, and S_k^A and S_k^C are the accuracy score and consistency score of the reference frame with index k, respectively.

The reference score is used as a basis to move out those memory frames with low scores so that the memory storage can be dynamically updated.

2.4. Intelligent Inspection Segmentation and Tracking

Once the target of the electrical equipment to be tracked is identified, the intelligent inspection device begins the segmentation and tracking process. Our network is based on the segmentation tracker SiamMask, the structure of which is shown in Figure 3. The whole framework is divided into three parts, which are the pixel-level equalized memory matching network (PEMMN), the mask quality evaluation module (MQEM), and the process for generating mask and tracking results. First, the feature extraction network of the base segmentation tracker (i.e., ResNet50) is utilized to extract the reference feature X^k and the query feature X^i from the reference and query frame images, respectively. Then, the downsampled reference frame mask m^k, the reference features X^k, and the query features X^i are introduced as inputs to a pixel-level equalized memory matching method. By comparing the reference features with the current frame features and retrieving target-related information from the features of the stored frames, the obtained matching score is concatenated with the query frame feature X^i to generate a composite feature map X'. Further, the composite feature X' is fed into three branches of the base segmentation tracker, namely, a classification branch (Cls), a regression branch (Reg), and a segmentation branch (Segmentation), which generates a binary mask. The segmentation branch introduces the low-level features of the query frame during the up-sampling process to make the edges of its segmentation result finer. From this process, the query frame mask M^i and the bounding box of the target object (i.e., tracking result) can be inferred. Finally, the current query frame

mask M^i is input into the mask quality evaluation module (MQEM) together with the query frame image to obtain the evaluation score S_t^A, and if it is higher than the threshold σ, the query frame mask M^i is added to the memory bank to dynamically update the memory bank.

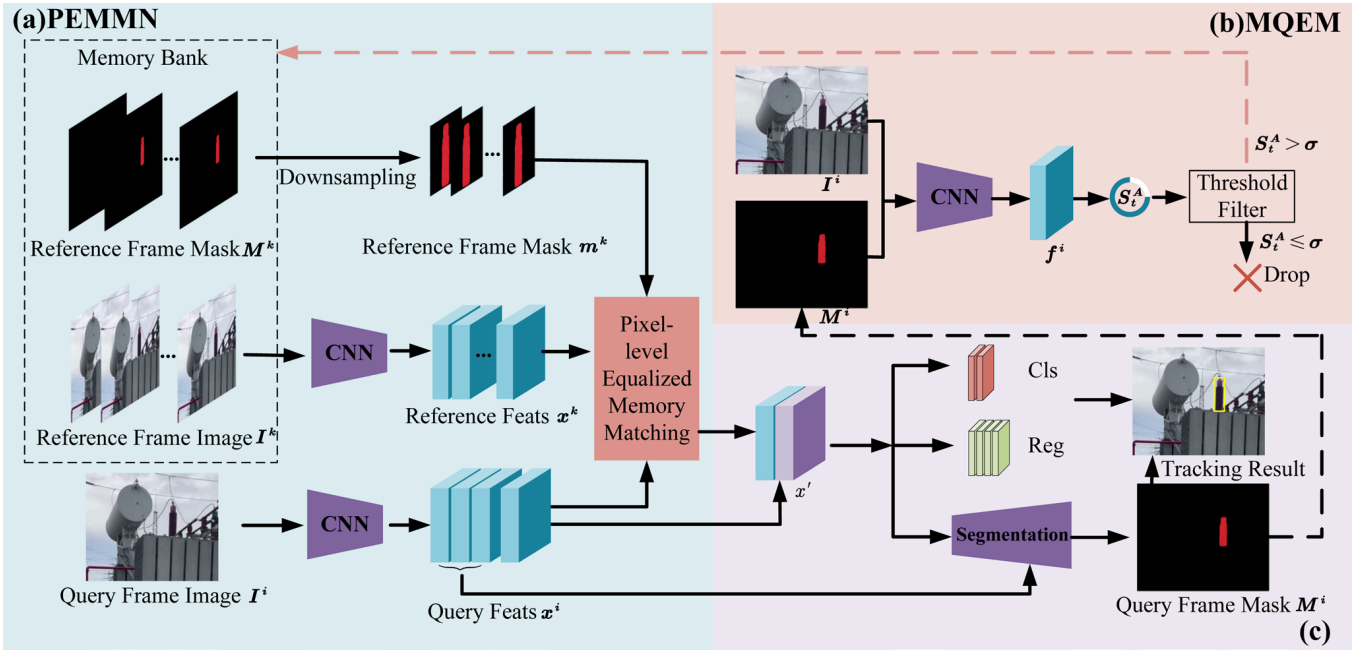

Figure 3. The architecture of the method we propose. (**a**) The pixel-level equalized memory matching network (PEMMN), in which the query frame is matched to the dynamically stored reference frame using our matching method. (**b**) MQEM is the mask quality evaluation module to decide if the mask of the query frame can be included in the memory. (**c**) The process involves combining the enhanced feature X' with the base segmentation tracking framework to generate the mask of the current frame and the tracking box of the target.

3. Results and Discussion

3.1. Experimental Setup and Datasets

The experiment is carried out in Python3.7 with PyTorch framework on a PC equipped with an Intel i7-10700 2.90 GHz CPU, 16 GB RAM, and an NVIDIA GeForce GTX 1650 GPU. In this work, we employ the first four layers of the pre-trained ResNet50 as the backbone network to extract features. The target search region is cropped to 255×255. A binary mask output is obtained by thresholding the predicted segmentation at 0.15, the size of which is changed to 15×15 and then stored as a value in the memory network. The memory storage threshold σ is set to 0.8 by default.

During training, the pre-training of the backbone network ResNet50 is performed on the dataset ImageNet [26] classification task. Using the stochastic gradient descent (SGD) method, a warmup period was first performed where the learning rate progressively elevates from 10^{-3} to 5×10^{-3} across the initial 5 epochs, followed by a logarithmic reduction to the learning rate of 5×10^{-4} sustained over the subsequent 15 epochs. The datasets used for our algorithmic model to be trained are ImageNet [26] and COCO [27]. We then conducted experiments on several commonly used tracking datasets, including OTB-100 [28], TC128 [29], and UAV123 [30], all three of which are test sets. Moreover, we performed experiments on actual captured substation inspection videos to validate the effectiveness of our method. These videos are collected from real scenarios, which can provide a closer test environment. By conducting experiments on these inspection videos, we can better evaluate the applicability and robustness of the proposed methods in actual scenarios.

3.2. Experiments on the Common Benchmark

The proposed method is comprehensively evaluated on three widely-used tracking benchmarks (OTB100, TC-128, and UAV123). A summary table is presented, as shown in Table 1, to compare our algorithm with other state-of-the-art algorithms, mainly in terms of both success rate (SR) and precision (P).

Table 1. State-of-the-art comparison on OTB100, TC128, and UAV123. Best in **bold**, second best underlined.

Methods	OTB100 [28]		TC128 [29]		UAV123 [30]	
	SR	P	SR	P	SR	P
DaSiamRPN [31]	0.658	0.880	-	-	-	-
TADT [32]	0.656	0.854	-	-	-	-
GradNet [33]	0.639	0.861	-	-	-	-
DeepSRDCF [34]	0.636	0.851	-	-	-	-
CFNet [35]	0.587	0.778	-	-	-	-
SiamFC [36]	0.587	0.772	0.489	0.672	-	-
SESiamFC [37]	0.650	0.864	-	-	-	-
SiamDW [38]	-	-	-	-	0.536	0.776
SiamDWfc [38]	0.627	0.828	-	-	-	-
SiamRPN [39]	0.629	0.847	-	-	0.581	0.772
SiamMask [40]	0.649	0.842	0.540	0.725	0.602	0.794
SiamFC++ [41]	-	-	0.566	0.763	-	-
SiamRPN++ [42]	-	-	0.577	0.775	0.611	0.804
SiamGAT [43]	-	-	0.559	0.753	-	-
SiamCAR [44]	-	-	-	-	0.615	0.804
SiamBAN [45]	-	-	-	-	0.604	0.795
Ocean [46]	-	-	0.557	0.752	0.621	0.823
HCFT [47]	-	-	0.495	0.692	-	-
RMIT [48]	-	-	0.551	0.761	-	-
ADMT [49]	-	-	-	-	0.535	0.754
Ours	**0.670**	**0.881**	**0.580**	**0.779**	**0.625**	**0.827**

In addition, success and precision plots are created, ranking them based on area under the success curve and precision. The success plot shows the relationship between the probability of successfully tracking the target and the probability of false alarms at different success rate thresholds. The precision plot provides a detailed representation of the trade-off between precision and the confidence score threshold in a tracking algorithm. It visualizes how the precision of the algorithm changes as the confidence score threshold varies.

3.2.1. Evaluation on the OTB100 Dataset

OTB100 dataset is a classical tracking benchmark with 100 sequences covering a diverse range of challenges such as scale variation, background clutter, and fast motion. We display the results on OTB100. Here we compare our tracker with 11 recent state-of-the-art methods: DaSiamRPN [31], TADT [32], GradNet [33], DeepSRDCF [34], CFNet [35], SiamFC [36], SESiamFC [37], SiamDWfc [38], SiamRPN [39], and SiamMask [40], as shown in Table 1. In order to show the curves in the graphs more clearly, we selected some representative algorithms to be plotted, as shown in Figure 4. Our model reaches 88.1% in the AUC score and 67% in the success score. When our equalized memory matching network is equipped to the baseline segmentation and tracker SiamMask, the improvement is 3.9 and 2.1 points gains.

Figure 5 displays the performance of success rates and precision rates for our method and other advanced methods in challenging scenarios such as background clutters, fast motion, illumination variation, scale variation, etc. Our tracker outperforms most in terms of performance. Notably, our method performs best in the face of background clutter

and illumination variation. In our algorithm, we propose a pixel-level equalized memory matching network to constrain the background clutter and enhance the target. Furthermore, it is obvious that our method outperforms other approaches in the case of fast motion and scale variations. The reason is that we combine the historical mask information with the current frame features.

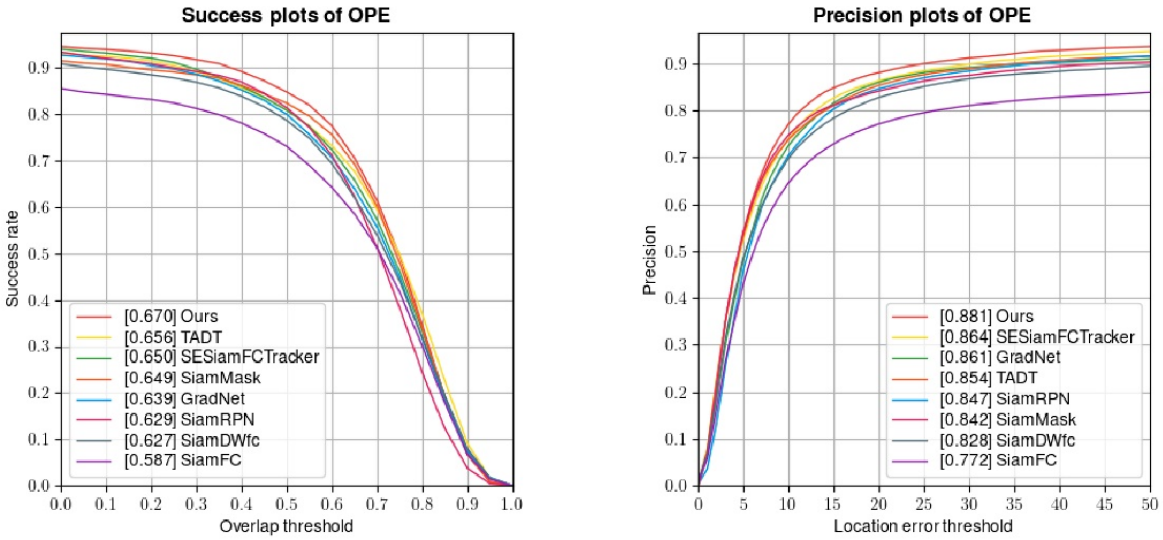

Figure 4. Success and precision plots on OTB100.

3.2.2. Evaluation on the TC-128 Dataset

Unlike OTB, the TC-128 dataset consists of 128 color video sequences which offer more intricate and demanding tracking tasks. To demonstrate the versatility of our algorithm, we ran further tests on the TC-128 dataset. We conducted quantitative comparisons between our trackers and several state-of-the-art trackers, including SiamFC [36], SiamMASK [40], SiamFC++ [41], SiamRPN++ [42], SiamGAT [43], Ocean [46], HCFT [47], and RMIT [48]. The tracking results are illustrated in Table 2. Our approach ranks first in success and precision rate, with a precision score of 77.9% and a success rate AUC score of 58%, respectively. In comparison with the baseline method SiamMask, we have a success rate of 4% and a precision rate of 5.4% higher, respectively.

Our tracker outperforms the memory tracker RMIT in both success rate and precision. While RMIT also utilizes target appearance information and leverages memory content about the target through a memory residual branch, our approach differs in terms of the content stored in the memory. Instead, we store mask information relevant to the historical frames to capture the robust target representation. As a result, both our algorithm and RMIT demonstrate superior performance compared to other trackers.

Table 2. Comparison of experimental results on the TC-128 dataset.

Tracker	Success Rate	Precision
Ours	0.580	0.779
SiamRPN++	0.577	0.775
SiamFC++	0.566	0.763
SiamGAT	0.559	0.753
Ocean	0.557	0.752
RMIT	0.551	0.761
SiamMASK	0.540	0.725
HCFT	0.495	0.692
SiamFC	0.489	0.672

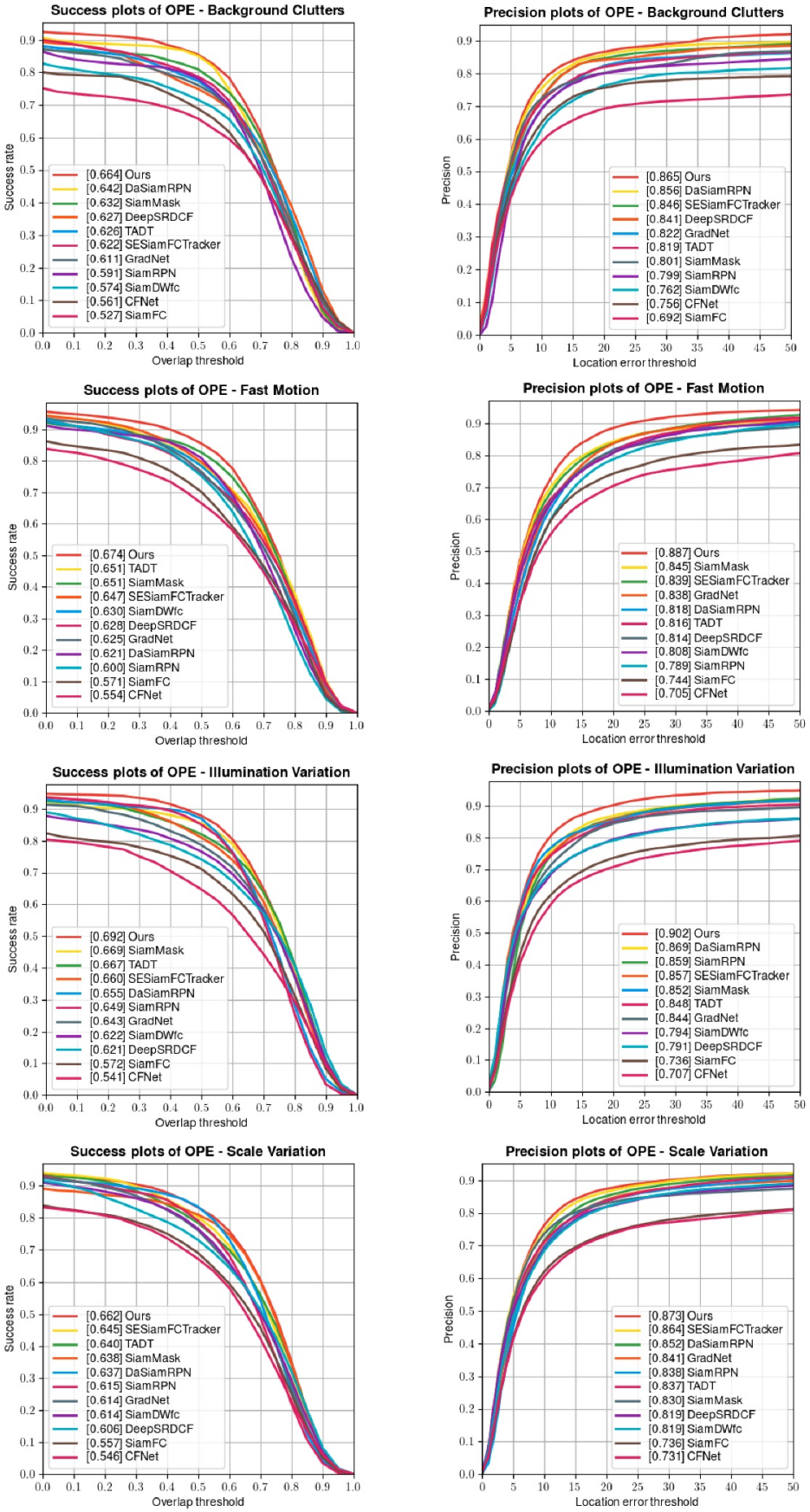

Figure 5. Success and precision plots of four attributes challenges from OTB100.

3.2.3. Evaluation on the UAV123 Dataset

The UAV123 dataset comprises 123 challenging video sequences captured exclusively by unmanned aerial vehicles (UAVs). Each UAV123 video sequence, recorded from low altitudes, consists of an average of 915 frames. There are major challenges for trackers due to the inherent instability of the UAV perspective and the frequent variations in target distance. Consequently, the low resolution of many objects in the video leads to particularly high tracking requirements for this dataset.

UAVs are widely employed in the field of intelligent inspection. Taking advantage of this similarity, we further validate the effectiveness of our method on the UAV123 dataset. Figure 6 shows the tracking results compared to six other Siamese trackers including, SiamRPN [35], SiamMASK [40], SiamRPN++ [42], SiamCAR [44], SiamBAN [45], Ocean [46], SiamDW [48] and ADMT [49]. Our tracker achieves a success score of 0.625 and a precision score of 0.827, which still outperforms recent competitive Siamese trackers.

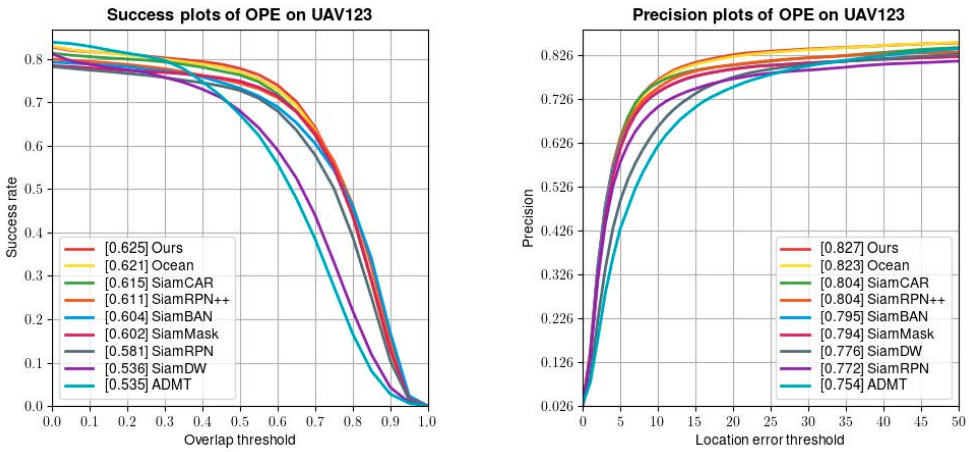

Figure 6. Success and precision plots on UAV123.

3.3. Experiments on the Self-Built Actual Inspection Dataset

Since there is no publicly standardized transformer substation intelligent inspection dataset available, and to more convincingly evaluate the performance of the proposed method, we gathered videos of real inspection scenarios and compiled a small dataset for the experiment. The self-built dataset contains scenes of electrical equipment such as instruments, insulators, and liquidometers taken by inspection robots and UAVs. The same evaluation metric described above for the benchmark dataset is utilized to evaluate the overall performance of the proposed approach.

In the comparison experiment, classic tracking networks are selected, including TADT [32], SiamRPN [39], SiamMASK [40], SiamCAR [44], and Ocean [46]. The tracking effect is shown in Figure 7. Our approach ranks first in success and precision rate, with a precision score of 89.3% and a success rate AUC score of 74.9%, respectively. Also, the proposed method achieves a processing speed of up to 33 FPS, which exceeds the minimum requirement of 30 FPS for real-time segmentation, and real-time tracking of substation detection is realized.

Figure 8 shows a qualitative comparison of our algorithm and baseline on the real inspection scene videos. Our algorithm demonstrates great advantages in segmentation accuracy in four substation inspection scenarios. The method has smoother segmented contours than the baseline segmentation tracking algorithm, which improves the tracking accuracy and is an effective network that can be used for real-time segmentation and tracking. The green bounding box is the ground truth, and the segmentation and tracking results are shown with the pink mask and the yellow bounding box.

In order to compare and analyze the tracking ability of the proposed methods even further, the tracking results of the different methods are considered to be demonstrated in four real scenarios, as shown in Figure 9. It can be seen that the algorithm in this paper is

able to obtain better tracking results with strong robustness and real-time performance for tracking power equipment in substation scenarios.

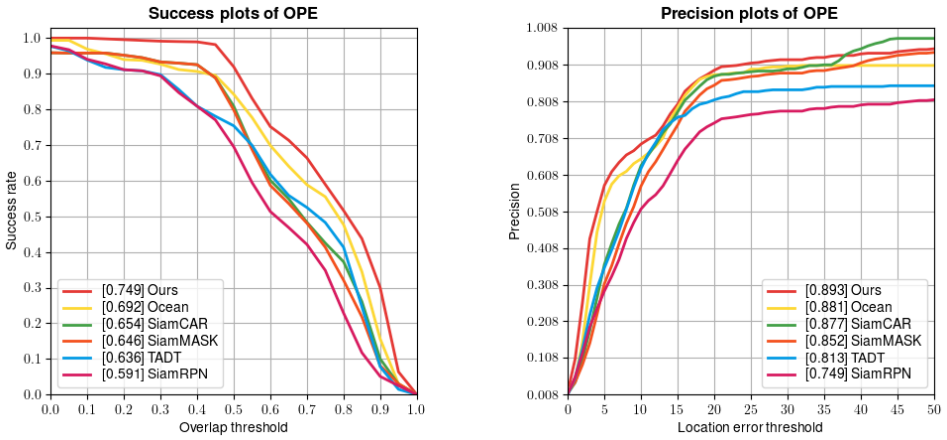

Figure 7. Success and precision plots on the self-built actual inspection dataset.

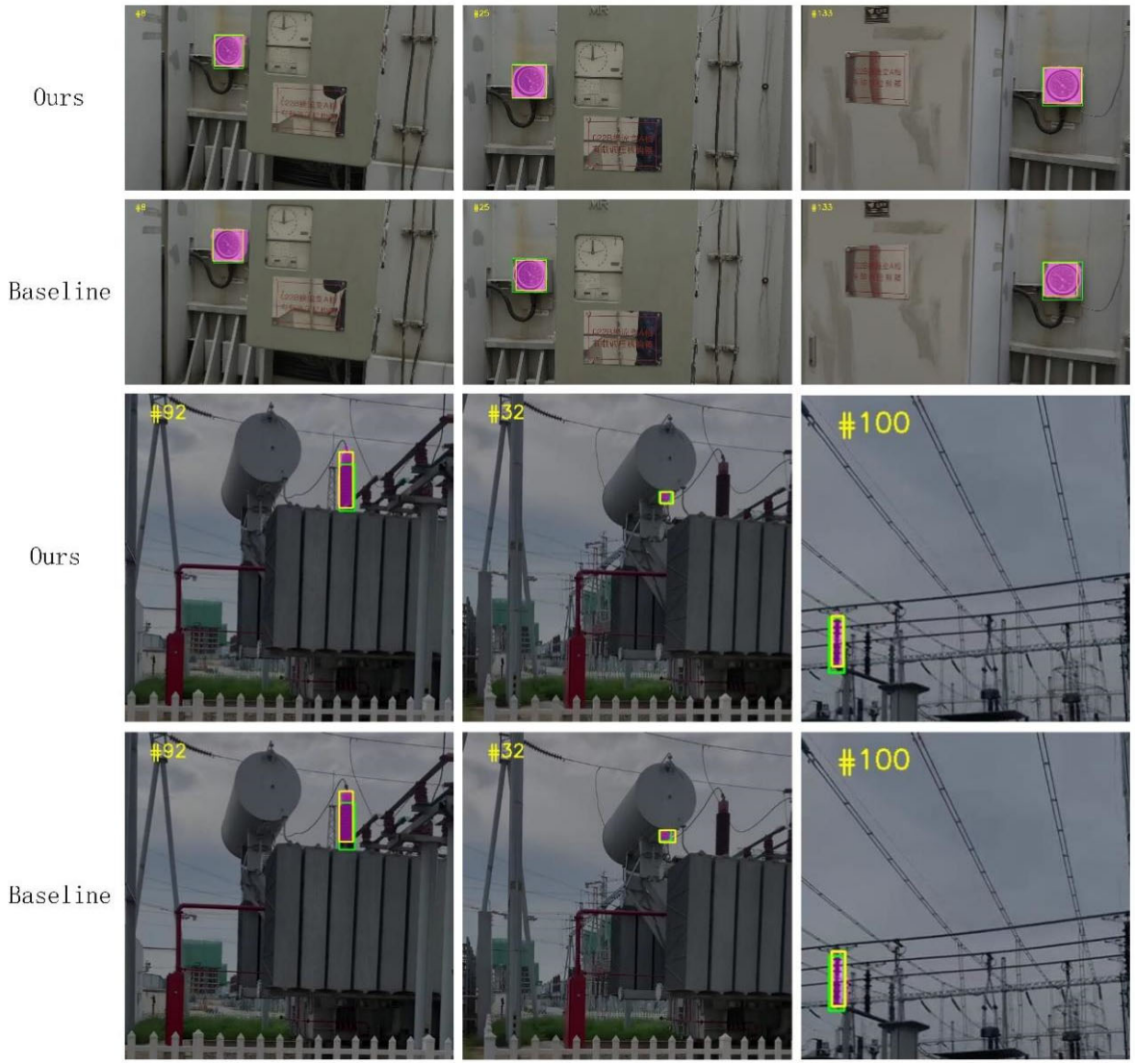

Figure 8. Segmentation and tracking results of our method and baseline on real inspection scene videos. The green bounding box is the ground truth, and the segmentation and tracking results are shown with the pink mask and the yellow bounding box.

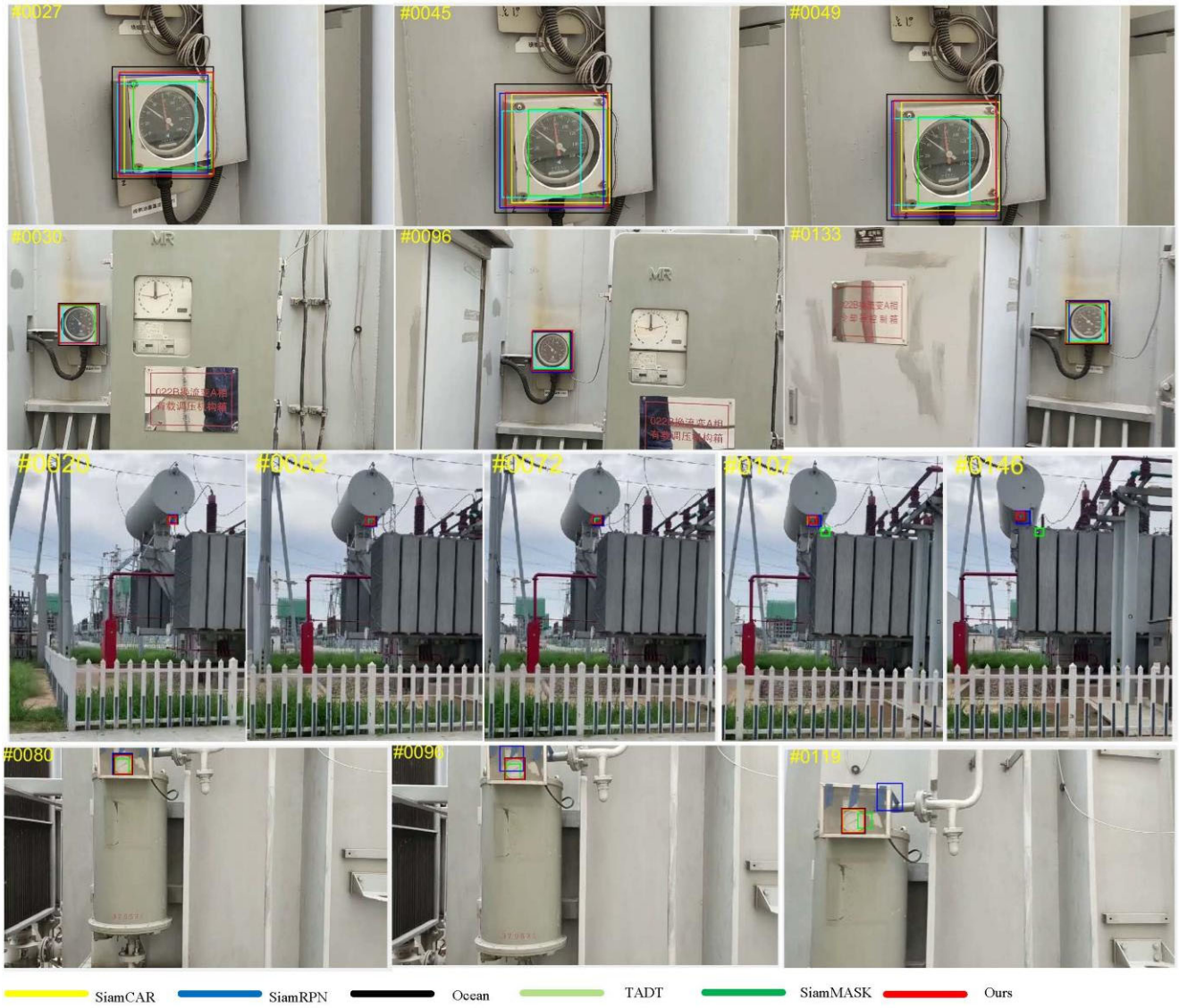

Figure 9. Tracking results of different tracking methods.

4. Conclusions

Accurate target segmentation and robust tracking are of great significance in substation intelligent inspection. A segmentation tracking algorithm based on pixel-level equalized memory matching network (PEMMN) is proposed. Experimental results show that the method exhibits excellent performance on both real substation inspection scenarios and three benchmark datasets. This work can not only help to ensure effective detection and localization of power equipment and improve the accuracy and effectiveness of inspection, but also help to reduce false alarms and missed inspections and alleviate manual intervention, thus improving work efficiency. Applying the target segmentation and tracking algorithm proposed in this paper to UAV inspection can realize accurate monitoring and tracking of power equipment from an aerial perspective and improve the coverage and efficiency of inspection. It can also be combined with artificial intelligence technology and big data analysis to realize intelligent fault diagnosis and prediction capability and improve the intelligent level of power inspection.

Author Contributions: Conceptualization, H.Z. and B.Z.; methodology, B.Z.; software, B.Z. and Y.T.; validation, B.Z. and Y.T.; formal analysis, Z.L.; investigation, H.Z. and B.Z.; resources, H.Z. and B.Z.; data curation, Y.T. and Z.L.; writing—original draft preparation, B.Z.; writing—review and editing, H.Z.; visualization, B.Z. and Y.T.; supervision, H.Z. and Z.L.; project administration, H.Z.

and B.Z.; funding acquisition, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China under Grant (62272423, 62072416, 62006213, 62102373) and the Excellent Youth Science Foundation of Henan Province (2300421055).

Data Availability Statement: Data are contained within the article.

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

References

1. Liu, Y.Q.; Qin, H.; Zhang, Z.D.; Pei, S.Q.; Jiang, Z.Q.; Feng, Z.K.; Zhou, J.Z. Probabilistic spatiotemporal wind speed forecasting based on a variational Bayesian deep learning model. *Appl. Energy* **2020**, *260*, 114259. [CrossRef]
2. Zhang, M.; Li, J.; Li, Y.; Xu, R. Deep Learning for Short-Term Voltage Stability Assessment of Power Systems. *IEEE Access* **2021**, *9*, 29711–29718. [CrossRef]
3. Khodayar, M.; Wang, J.H.; Wang, Z.Y. Deep generative graph distribution learning for synthetic power grids. *arXiv* **2019**, arXiv:1901.09674.
4. Hamdi, H.; Regaya, C.B.; Zaafouri, A. A sliding-neural network control of induction-motor-pump supplied by photovoltaic generator. *Prot. Control Mod. Power Syst.* **2020**, *5*, 1. [CrossRef]
5. Hui, X.; Bian, J.; Zhao, X.; Tan, M. Vision-based autonomous navigation approach for unmanned aerial vehicle transmission-line inspection. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1729881417752821. [CrossRef]
6. Constantin, A.; Dinculescu, R.N. UAV development and impact in the power system. In Proceedings of the 2019 8th International Conference on Modern Power Systems (MPS), Cluj Napoca, Romania, 21–23 May 2019; pp. 1–5.
7. Zormpas, A.; Moirogiorgou, K.; Kalaitzakis, K.; Plokamakis, G.A.; Partsinevelos, P.; Giakos, G.; Zervakis, M. Power transmission lines inspection using properly equipped unmanned aerial vehicle (UAV). In Proceedings of the IEEE International Conference on Imaging Systems and Techniques, Krakow, Poland, 16–18 October 2018.
8. Alhassan, A.B.; Zhang, X.; Shen, H.; Xu, H. Power transmission line inspection robots: A review, trends and challenges for future research. *Int. J. Electr. Power Energy Syst.* **2020**, *118*, 105862. [CrossRef]
9. 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, Seoul, Republic of Korea, 27–28 October 2019; pp. 6181–6190.
10. 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, 13–19 June 2020; pp. 7181–7190.
11. Ma, C.; Huang, J.B.; Yang, X.; Yang, M.H. Adaptive correlation filters with long-term and short-term memory for object tracking. *Int. J. Comput. Vis.* **2018**, *126*, 771–796. [CrossRef]
12. Yang, T.; Chan, A.B. Learning dynamic memory networks for object tracking. In *Presented at European Conference on Computer Vision*; Springer: Cham, Switzerland, 2018.
13. Yang, T.; Chan, A.B. Visual Tracking via Dynamic Memory Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 360–374. [CrossRef] [PubMed]
14. Li, G.; Peng, M.; Nai, K.; Li, Z.; Li, K. Reliable correlation tracking via dual-memory selection model. *Inf. Sci.* **2020**, *518*, 238–255. [CrossRef]
15. Fu, Z.; Liu, Q.; Fu, Z.; Wang, Y. STMTrack: Template-free Visual Tracking with Space-time Memory Networks. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13769–13778.
16. Hu, Y.-T.; Huang, J.-B.; Schwing, A.G. Videomatch: Matching based video object segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 54–70.
17. Voigtlaender, P.; Chai, Y.; Schroff, F.; Adam, H.; Leibe, B.; Chen, L.-C. FEElVOS: Fast end-to-end embedding learning for video object segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9473–9482.
18. Yang, Z.; Wei, Y.; Yang, Y. Collaborative Video Object Segmentation by Multi-Scale Fore-ground-Background Integration. *IEEE Transac.-Tions. Pattern Anal. Mach. Intell.* **2022**, *44*, 4701–4712.
19. Oh, S.W.; Lee, J.-Y.; Xu, N.; Kim, S.J. Video Object Segmentation Using Space-Time Memory Networks. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9225–9234.
20. Lu, X.; Wang, W.; Danelljan, M.; Zhou, T.; Shen, J.; Van Gool, L. Video object segmentation with episodic graph memory networks. In Proceedings of the Computer Vision-ECCV2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 661–679.
21. Yu, S.; Xiao, J.; Zhang, B.; Lim, E.G.; Zhao, Y. Fast pixel-matching for video object segmentation. *Signal Process. Image Commun.* **2021**, *98*, 116373. [CrossRef]
22. Seong, H.; Hyun, J.; Kim, E. Kernelized memory network for video object segmentation. In Proceedings of the Computer Vi-sion-ECCV2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 629–645.

23. Oh, S.W.; Lee, J.-Y.; Lee, S.; Lee, S.; Kim, E. Hierarchical Memory Matching Network for Video Object Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 12889–12898.
24. Cho, S.; Lee, H.; Kim, M.; Jang, S.; Lee, S. Pixel-Level Bijective Matching for Video Object Segmentation. In Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, Hawaii, USA, 3–8 January 2022; pp. 1453–1462.
25. Huang, Z.; Huang, L.; Gong, Y.; Huang, C.; Wang, X. Mask scoring R-CNN. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6409–6418.
26. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
27. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision-ECCV2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
28. Wu, Y.; Lim, J.; Yang, M.-H. Online Object Tracking: A Benchmark. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2411–2418.
29. Liang, P.; Blasch, E.; Ling, H. Encoding Color Information for Visual Tracking: Algorithms and Benchmark. *IEEE Trans. Image Process.* **2015**, *24*, 5630–5644. [CrossRef] [PubMed]
30. Mueller, M.; Smith, N.; Ghanem, B. A benchmark and simulator for uav tracking. In Proceedings of the Computer Vision-ECCV 2016, Zurich, Switzerland, October 2016; pp. 445–461.
31. 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, Zurich, Switzerland, October 2018; pp. 103–119.
32. Li, X.; Ma, C.; Wu, B.; He, Z.; Yang, M.-H. Target-Aware Deep Tracking. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1369–1378.
33. Li, P.; Chen, B.; Ouyang, W.; Wang, D.; Yang, X.; Lu, H. GradNet: Gradient-Guided Network for Visual Object Tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27–28 October 2019; pp. 6162–6171.
34. Danelljan, M.; Häger, G.; Khan, F.S.; Felsberg, M. Convolutional Features for Correlation Filter Based Visual Tracking. In Proceedings of the 2015 IEEE International Conference on Computer Vision Workshop, Santiago, Chile, 7–13 December 2015; pp. 621–629.
35. Zhang, G.; Li, Z.; Li, J.; Hu, X. Cfnet: Cascade fusion network for dense prediction. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 August 2023.
36. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the Computer Vision-ECCV 2016 Workshops, Amsterdam, The Netherlands, 11–14 October 2016; pp. 850–865.
37. Sosnovik, I.; Moskalev, A.; Smeulders, A. Scale Equivariance Improves Siamese Tracking. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 2764–2773.
38. Zhang, Z.; Peng, H. Deeper and Wider Siamese Networks for Real-Time Visual Tracking. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4586–4595.
39. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High Performance Visual Tracking with Siamese Region Proposal Network. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8971–8980.
40. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1328–1338.
41. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. *AAAI Conf. Artif. Intell.* **2019**, *34*, 2159–5399. [CrossRef]
42. 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 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4277–4286.
43. Guo, D.; Shao, Y.; Cui, Y.; Wang, Z.; Zhang, L.; Shen, C. Graph attention tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9538–9547.
44. Guo, D.; Wang, J.; Cui, Y.; Wang, Z.; Chen, S. SiamCAR: Siamese Fully Convolutional Classification and Regression for Visual Tracking. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6268–6276.
45. Chen, C.; Shen, X.; Li, G.; Zhang, S.; Ji, R. Siamese box adaptive network for visual tracking. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 6667–6676.
46. Zhang, Z.; Peng, H.; Fu, J.; Li, B.; Hu, W. Ocean: Object-aware anchor-free tracking. In Proceedings of the 2020 Computer Vision–ECCV: 16th European Conference, Glasgow, UK, 31 August 2020; pp. 23–28.
47. Ma, C.; Huang, J.-B.; Yang, X.; Yang, M.-H. Hierarchical convolutional features for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3074–3082.

48. Zhang, H.; Zhang, J.; Nie, G. Residual memory inference network for regression tracking with weighted gradient harmonized loss. *Inf. Sci.* **2022**, *597*, 105–124. [CrossRef]
49. Zhang, H.; Liang, J.; Zhang, J.; Xian, P. Attention-driven memory network for online visual tracking. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, 1–14. [CrossRef]