

Article

Modified Siamese Network Based on Feature Enhancement and Dynamic Template for Low-Light Object Tracking in UAV Videos

Lifan Sun ^{1,2,*}, Shuaibing Kong ¹, Zhe Yang ³, Dan Gao ¹ and Bo Fan ¹¹ School of Information Engineering, Henan University of Science and Technology, Luoyang 471023, China; neo1999@stu.haust.edu.cn (S.K.); fanbo@haust.edu.cn (B.F.)² Longmen Laboratory, Luoyang 471000, China³ Xiaomi Technology Co., Ltd., Beijing 100102, China; yangzhe11@xiaomi.com

* Correspondence: lifan.sun@haust.edu.cn

Abstract: Unmanned aerial vehicles (UAVs) visual object tracking under low-light conditions serves as a crucial component for applications, such as night surveillance, indoor searches, night combat, and all-weather tracking. However, the majority of the existing tracking algorithms are designed for optimal lighting conditions. In low-light environments, images captured by UAV typically exhibit reduced contrast, brightness, and a signal-to-noise ratio, which hampers the extraction of target features. Moreover, the target's appearance in low-light UAV video sequences often changes rapidly, rendering traditional fixed template tracking mechanisms inadequate, and resulting in poor tracker accuracy and robustness. This study introduces a low-light UAV object tracking algorithm (SiamLT) that leverages image feature enhancement and a dynamic template-updating Siamese network. Initially, the algorithm employs an iterative noise filtering framework-enhanced low-light enhancer to boost the features of low-light images prior to feature extraction. This ensures that the extracted features possess more critical target characteristics and minimal background interference information. Subsequently, the fixed template tracking mechanism, which lacks adaptability, is enhanced by dynamically updating the tracking template through the fusion of the reference and base templates. This improves the algorithm's capacity to address challenges associated with feature changes. Furthermore, the Average Peak-to-Correlation Energy (APCE) is utilized to filter the templates, mitigating interference from low-quality templates. Performance tests were conducted on various low-light UAV video datasets, including UAVDark135, UAVDark70, DarkTrack2021, NAT2021, and NAT2021L. The experimental outcomes substantiate the efficacy of the proposed algorithm in low-light UAV object-tracking tasks.

Keywords: unmanned aerial vehicle; low-light tracking; Siamese network; feature enhancement; dynamic template



Citation: Sun, L.; Kong, S.; Yang, Z.; Gao, D.; Fan, B. Modified Siamese Network Based on Feature Enhancement and Dynamic Template for Low-Light Object Tracking in UAV Videos. *Drones* **2023**, *7*, 483. <https://doi.org/10.3390/drones7070483>

Academic Editor: Giordano Teza

Received: 31 May 2023

Revised: 17 July 2023

Accepted: 19 July 2023

Published: 21 July 2023



Copyright: © 2023 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

Visual object tracking is a fundamental task in computer vision that finds extensive applications in the unmanned aerial vehicle (UAV) domain. Recent years have witnessed the emergence of new trackers that exhibit exceptional performance in UAV tracking [1–3], which is largely attributed to the fine manual annotation of large-scale datasets [4–7]. However, the evaluation standards and tracking algorithms currently employed are primarily designed for favorable lighting conditions. In real-world scenarios, low-light conditions such as nighttime, rainy weather, and small spaces are often encountered, resulting in images with low contrast, low brightness, and low signal-to-noise ratio compared to normal lighting. These discrepancies give rise to inconsistent feature distributions between the two types of images, thereby rendering it challenging to extend trackers designed for

favorable lighting conditions to low-light scenarios [8,9], making it more challenging for UAV tracking.

Low-light UAV video sequences exhibit poor robustness and tracking drift when conventional object-tracking algorithms are employed, as illustrated in Figure 1. This paper aims to address the issue of object tracking under low-light conditions, which can be divided into two sub-problems: enhancing low-light image features and tackling the challenge of target appearance changes in low-light video sequences. First, the low contrast, low brightness, and low signal-to-noise ratio of low-light images make feature extraction more arduous compared to normal images. Insufficient feature information hampers subsequent object-tracking tasks and constrains the performance of object-tracking algorithms. Another obstacle hindering the effectiveness of object-tracking algorithms arises from the characteristics of low-light video sequences. During tracking, the target's appearance often changes, and when it becomes occluded or deformed, its features no longer correspond to the original template features, resulting in tracking drift. Such challenges are commonplace in vision object-tracking tasks and are more pronounced under low-light conditions due to the unstable lighting conditions, which serve as a crucial limiting factor for the performance of object-tracking algorithms.

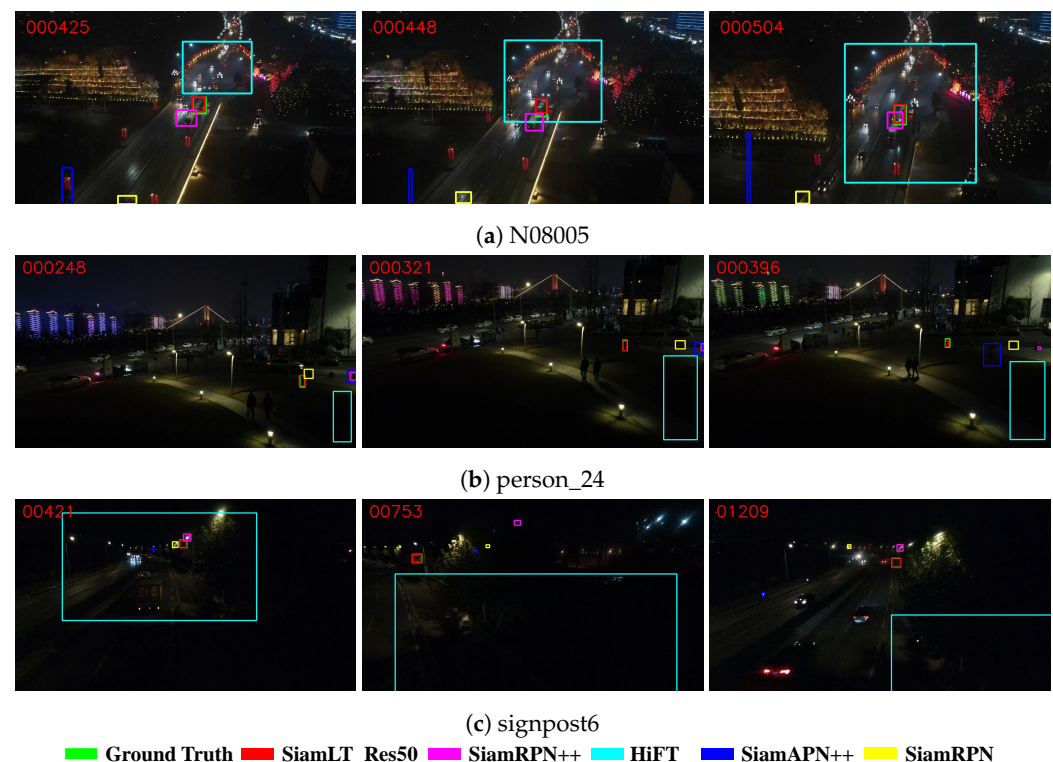


Figure 1. Trackers performance under low-light conditions.

Taking into account the aforementioned issues, this paper presents a visual object-tracking algorithm specifically designed for low-light environments. First, to enhance the quality of images in low-light settings and to mitigate the challenge of feature extraction, a low-light image enhancement algorithm is devised. This algorithm integrates deep learning and filtering techniques to accentuate crucial target features. Simultaneously, the traditional static template mechanism of Siamese networks is improved upon by employing dynamic templates that capture changes in target features for feature matching. This increases the likelihood of matching templates to the correct targets, thereby enhancing tracker performance in low-light conditions. In summary, the principal contributions of this paper are as follows:

- (1) To tackle the challenges of low contrast, low brightness, and low signal-to-noise ratio in low-light images, which hinder effective target feature extraction, an enhanced

- low-light image enhancement algorithm is proposed. An iterative noise filtering framework is developed to suppress high-intensity noise arising from low-light image enhancement and to emphasize key features in low-light images;
- (2) To address the issue of appearance changes in low-light tracking tasks, a dynamic template tracking mechanism is proposed, which surpasses the limited adaptability of traditional Siamese networks reliant on static templates to changes in target features. This enhances the tracker's robustness;
 - (3) By amalgamating a dynamic template Siamese network framework with a low-light image enhancement algorithm, two primary challenges are surmounted: extracting target features from low-light images and coping with frequent appearance changes in video sequences. Consequently, an object-tracking algorithm suitable for low-light situations is proposed to bolster tracker performance under such conditions.

2. Related Work

2.1. Low-Light Image Enhancement

The objective of low-light image enhancement is to improve the quality of images by making the details that are concealed in darkness visible. In recent years, this area has gained significant attention and undergone continuous development and improvement in various computer vision domains. Two main types of algorithms are used for low-light image enhancement, namely model-based methods and deep learning-based methods.

Model-based methods were developed earlier and are based on the Retinex theory [10]. According to this theory, low-light images can be separated into illuminance and reflectance components. The reflectance component contains the essential attributes of the image, including edge details and color information, while the illuminance component captures the general outline and brightness distribution of the objects in the image. Fu et al. [11,12] were the first to use the L2 norm to constrain illumination and proposed an image enhancement method that simultaneously estimates illuminance and reflectance components in the linear domain. This method demonstrated that the linear domain formula is more suitable than the logarithmic domain formula. Guo et al. [13] used relative total variation [14] as a constraint on illumination and developed a structure-aware smoothing model to obtain better estimates of illuminance components. However, this model has the disadvantage of overexposure. Li et al. [15] added a noise term to address low-light image enhancement under strong noise conditions. They introduced new regularization terms to jointly estimate a piecewise smooth illumination and a structure-displaying reflectance in the optimization problem of illumination and reflectance. They also modeled noise removal and low-light enhancement as a unified optimization goal. Additionally, Ref. [16] proposed a semi-decoupled decomposition model to simultaneously enhance brightness and suppress noise. Although some models use camera response characteristics (e.g., LEACRM [17]), their effects are often not ideal and require manual adjustment of numerous parameters when dealing with real scenes.

In recent years, deep learning-based methods have rapidly emerged with the advancement of computer technology. Li et al. [18] proposed a control-based method for optimizing UAV trajectories, which incorporates energy conversion efficiency by directly deriving the model from the voltage and current flow of the UAV's electric motor. EvoXBench [19] introduced an end-to-end process to address the lack of a general problem statement for NAS tasks from an optimization perspective. Zhang et al. [20] presented a low-complexity strategy for super-resolution (SR) based on adaptive low-rank approximation (LRA), aiming to overcome the limitations of processing large-scale datasets. Jin et al. [21] developed a deep transfer learning method that leverages facial recognition techniques to achieve a computer-aided facial diagnosis, validated in both single disease and multiple diseases with healthy controls. Zheng et al. [22] proposed a two-stage data augmentation method for automatic modulation classification in deep learning, utilizing spectral interference in the frequency domain to enhance radio signals and aid in modulation classification. This marks the first instance where frequency domain information has been considered to enhance

radio signals for modulation classification purposes. Meanwhile, deep learning-based low-light enhancement algorithms have also made significant progress. Chen et al. [23] created a new dataset called LOL dataset by collecting low/normal light image pairs with adjusted exposure time. This dataset is the first to contain image pairs obtained from real scenes for low-light enhancement research, making a significant contribution to learning-based low-light image enhancement algorithm research. Many algorithms have been trained based on this dataset. The retinal network, designed in [23], generated unnatural enhancement results. KinD [24] improved some of the issues in the retinal network by adjusting the network architecture and introducing some training losses. DeepUPE [25] proposed a low-light image enhancement network that learned an image-to-illumination component mapping. Yang et al. [26] developed a fidelity-based two-stage network that first restores signals and then further enhances the results to improve overall visual quality, trained using a semi-supervised strategy. EnGAN [27] used a GAN-based unsupervised training method to enhance low-light images using unpaired low/normal light data. The network was trained using carefully designed discriminators and loss functions while carefully selecting training data. SSIENet [28] proposed a maximum entropy-based Retinex model that could estimate illuminance and reflectance components simultaneously while being trained only with low-light images. ZeroDCE [29] heuristically constructed quadratic curves with learned parameters to estimate parameter mapping from low-light input and used curve projection models for iterative light enhancement of low-light images. However, these models focus on adjusting the brightness of images and do not consider the noise that inevitably occurs in real-world nighttime imaging. Liu et al. [30] introduced prior constraints based on Retinex theory to establish a low-light image enhancement model and constructed an overall network architecture by unfolding its optimization solution process. Recently, Ma et al. [31] added self-correcting modules during training to reduce the model parameter size and improve inference speed.

However, these algorithms have limited stability, and it is difficult to achieve sustained superior performance, particularly in unknown real scenes where unclear details and inappropriate exposure are common and without good solutions for noise in images.

2.2. Object Tracking

In recent years, object tracking algorithms can be classified into methods based on discriminative correlation filtering [32–34] and methods based on Siamese networks. Achieving end-to-end training on trackers based on discriminative correlation filtering is challenging due to their complex online learning process. Moreover, limited by low-level manual features or inappropriate pre-trained classifiers, trackers based on discriminative correlation filtering become ineffective under complex conditions.

With the continuous improvement of computer performance and the establishment of large-scale datasets, tracking algorithms based on Siamese networks have become mainstream due to their superior performance. The Siamese network series of algorithms started with SINT [35] and SiamFC [36], which treat target tracking as a similarity learning problem and train Siamese networks using large amounts of image data. SiamFC introduced a correlation layer for feature fusion which significantly improved accuracy. Based on the success of SiamFC, subsequent improvements were made. CFNet [37] added a correlation filter to the template branch to make the network shallower and more efficient. DSiam [38] proposed a dynamic Siamese network that could be trained on labeled video sequences as a whole, fully utilizing the rich spatiotemporal information of moving objects and achieving improved accuracy with an acceptable speed loss. RASNet [39] used three attention mechanisms to weight the space and channels of SiamFC features, enhancing the network's discriminative ability by decomposing the coupling of feature extraction and discriminative analysis. SASiam [40] established a Siamese network containing semantic and appearance branches. During training, the two branches were separated to maintain specificity. During testing, the two branches were combined to improve accuracy. However, these methods require multi-scale testing to cope with scale changes and cannot handle

proportion changes caused by changes in target appearance. To obtain more accurate target bounding boxes, B. Li et al. [41] introduced a region proposal network (RPN) [42] into the Siamese network framework, achieving simultaneous improvement in accuracy and speed. SiamRPN++ [43] further adopted a deeper backbone and feature aggregation architecture to exploit the potential of deep networks on Siamese networks and improve tracking accuracy. SiamMask [44] introduced a mask branch to simultaneously achieve target tracking and image segmentation. Xu et al. [45] proposed a set of criteria for estimating the target state in tracker design and designed a new Siamese network, SiamFC++, based on SiamFC. DaSiamRPN [46] introduced existing detection datasets to enrich positive sample data and difficult negative sample data to improve the generalization and discrimination ability of trackers. It also introduced a local-to-global strategy to achieve good accuracy in long-term tracking. Anchor-free methods use per-pixel regression to predict four offsets on each pixel, reducing the hyperparameters caused by the introduction of RPNs. SiamBAN [47] proposed a tracking framework, containing multiple adaptive heads, that does not require multi-scale search or predefined candidate boxes, that directly classifies objects in a unified network, and that regresses bounding boxes. SiamCAR [48] added a centrality branch to help determine the position of the target center point and further improve tracking accuracy. Recently, Transformer [49] was integrated into the Siamese framework to simulate global information and improve tracking performance.

Regarding target tracking algorithms under low-light conditions, a DCF framework integrated with a low-light enhancer was proposed in [50]. However, it is limited to hand-crafted features and lacks transferability. Ye et al. [51] developed a new unsupervised domain adaptation framework that uses a day-night feature discriminator to adversarially train a daytime tracking model for nighttime tracking. However, there is currently insufficient targeted research on this issue.

Remark 1. *Despite significant progress in target tracking algorithms, previous research has largely focused on tracking targets under normal lighting conditions, with little attention paid to tracking under unfavorable lighting conditions. Current methods still fall short of meeting performance requirements in real-world scenarios. This article addresses the challenges of tracking targets under low-light conditions and proposes a targeted approach to addressing these issues. Experimental results demonstrate the effectiveness of the proposed algorithm.*

3. Low-Light Object Tracking Algorithm

3.1. Overall Framework

In low-light conditions, trackers often encounter tracking drift problems due to factors such as low target feature saliency and target feature changes. This paper proposes a low-light adapted target tracking algorithm (SiamLT) under the Siamese network framework. The algorithm enhances the feature extraction ability of the tracker under low-light conditions by incorporating an image feature enhancement module. A dynamic template tracking mechanism is used to address the challenges of target feature changes during tracking and improve tracking accuracy under low-light conditions.

The SiamLT network architecture is illustrated in Figure 2. The network takes in a low-light image sequence as input and crops the original image to obtain a template image and a search image. The low-light feature enhancement module is applied to enhance the feature information of the template and search images. Feature extraction is then performed to obtain the template feature map and search feature map. These feature maps are fed into the RPN network to generate a classification score map and a regression score map, which are used to determine the target position and size in the current frame. To enhance the Siamese network's adaptability to changes in target features, we introduce a dynamic template tracking mechanism to the traditional Siamese network tracking framework. As shown in Figure 2, the RPN network's template input is obtained by fusing the target features from the first frame and the previous frame. This enhances the tracker's accuracy and stability under low-light conditions. The Feature Enhancement, RPN, and Dynamic Template

modules presented in Figure 2 will be thoroughly discussed in Sections 3.2, 3.3, and 3.4, respectively.

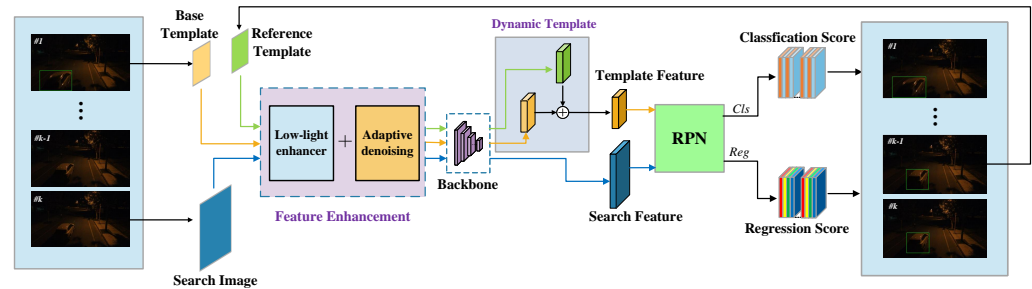


Figure 2. Overall structure diagram of the proposed algorithm.

3.2. Low-Light Image Feature Enhancement Module

The main objective of this module is to address the issues related to poor quality low-light images and difficulty in extracting target features. To this end, this paper divides the task of low-light image enhancement into two steps. The first step involves the use of an image illumination enhancement algorithm to improve the lighting conditions, adjust the brightness and contrast, and make the target features more prominent. However, during the brightness enhancement process, it is challenging to differentiate between target information and noise information in the image, and the noise intensity in the image will also increase accordingly. High-intensity noise can interfere with the extraction of key target features and hinder the tracking process. Therefore, this paper proposes an adaptive image filtering and denoising algorithm to suppress high-intensity noise in the image and reduce its interference with key target features after illumination enhancement. The two sub-modules work in tandem to process low-light images and obtain higher-quality image features, significantly improving the tracking accuracy of tracking algorithms.

3.2.1. Image Illumination Enhancement Submodule

Low-light images frequently exhibit insufficient brightness and inadequate contrast, posing challenges in extracting prominent feature information of the object and distinguishing it from the image background. Consequently, weak target features and a significant amount of interference information can lead to tracking drift in low-light conditions. To deal with these problems, this paper proposes an image illumination enhancement module that improves tracking performance by increasing the saliency of an object in the background.

The configuration of the module is illustrated in Figure 3, where I denotes the input image. Initially, the lighting conditions of the input image I are assessed. Only images with suboptimal lighting conditions are chosen for enhancement to prevent the overexposure of images exhibiting normal lighting conditions. This paper employs the log-average luminance of the image [52] as the illumination condition evaluation index. This index simplifies complex illumination information into a constant through pixel-level calculations. For a given RGB image I , the light intensity value of a single pixel $L^W(I)$ is first represented as:

$$L^W(x, y, I) = \sum_m \alpha_m \psi_m(I(x, y)), m \in \{R, G, B\} \quad (1)$$

where $\psi_m(I(x, y))$ represents the light intensity at position (x, y) in channel m of the image, such as $\psi_G(I(x, y))$ represents the light intensity value in the green channel. The channel weight parameters $\alpha_R, \alpha_G, \alpha_B$ satisfy $\alpha_R + \alpha_G + \alpha_B = 1$. Referring to [52], the logarithmic average illuminance intensity is represented as:

$$L^W(I) = \exp\left(\frac{1}{wh} \sum_{x,y} \log(\delta + L^W(x, y, I))\right) \quad (2)$$

where δ is a very small constant used to avoid the special case where L^W is 0. L^W can effectively represent the lighting conditions of the image, and there are obvious differences in L^W under different lighting conditions. Therefore, this paper uses L^W as an indicator to evaluate the lighting conditions of the image and sets a threshold to distinguish between normal-light images and low-light images.

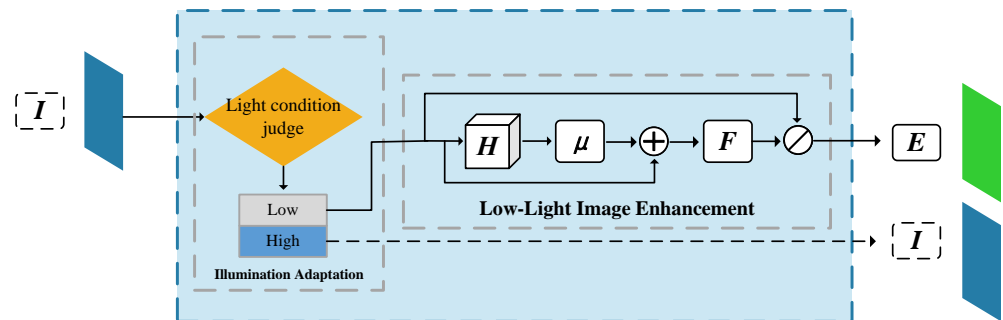


Figure 3. Schematic diagram of the illumination enhancement module.

The filtered low-light images are then processed through the low-light enhancement network for enhancement. In this study, the SCI illumination enhancement algorithm [31] is selected for the enhancement network, using its test phase. The network structure is shown in the figure. According to the retinal cortex theory, there is a relationship between low-light image y and ideal image z as follows:

$$y = z \otimes x \quad (3)$$

The network estimates the illumination component of the input image represented by “ x ”. Then, it removes the estimated illumination component from the original image to produce the enhanced image. As shown in Figure 3, the algorithm introduces a mapping relationship H_θ to learn the illumination component in low-light images, and the inference process can be written as:

$$z(y) : \begin{cases} u = H(y), \\ F = y + u, \\ z = y/F, \end{cases} \quad (4)$$

where u represents the output of the residual block, and F represents the illumination component obtained by mapping the input image.

3.2.2. Adaptive Image Filtering Denoising Algorithm

The image illumination enhancement module primarily focuses on improving the brightness of the low-light image without taking into account the feature information and noise in the image. As shown in Figure 4, during this process, enhancing valid feature information also enhances the noise information in the image, leading to an amplified noise intensity in the enhanced image compared to the original image. This high-intensity noise significantly interferes with extracting the target’s key features in the image and can result in tracking drift. To avoid tracking drift caused by the amplified noise, it is crucial to suppress the noise in the enhanced images produced by the illumination enhancement module.

There are several image denoizing methods, including filter-based methods [53,54], model-based methods [55,56], and learning-based methods [57–59]. However, model-based and learning-based methods are time-consuming, and are not suitable for real-time target-tracking tasks. On the other hand, the fast filter-based method has significant advantages in terms of speed and can effectively remove specific noise in the image. However, the selection of filter parameters, particularly the size of the filter window, has a significant impact on denoizing performance. A small filter window may not achieve satisfactory results, while a large filter window may lead to the loss of crucial information in the image.













Original						
Light level	0.083	0.072	0.052	0.069	0.080	0.053
Noise level	0.507	0.544	0.389	0.921	1.691	0.574
Enhanced						
Light level	0.329	0.247	0.186	0.275	0.265	0.229
Noise level	1.671	1.397	1.363	2.945	3.062	1.691

Figure 4. Comparison of images before and after enhancement.

A diagram of the denoising module is presented to illustrate the principle of the iterative filtering algorithm. To balance the suppression of noise and the preservation of feature information, an iterative filtering algorithm is designed for image denoising. The framework of the algorithm is depicted in Figure 5, where I represents the input noisy image. As there is no precise linear relationship between the noise intensity of the image and the filter parameters, it is difficult to set the most appropriate filter parameters based on the noise intensity. Therefore, to avoid losing key information in the image, a low-intensity filter with fixed parameters is used for iterative denoising. The problem of selecting filter parameters is transformed into controlling the number of iterations. The filtering strength increases with the number of iterations. To match the number of iterations that preserves the key information and removes noise, an image information loss supervision module is set up. After each round of filtering, the output image is compared with the original image for information comparison. Large differences between the output and original images indicate that key information may have been lost. At this point, the filtering process stops, and the output image is generated. The calculation formula for the loss L is presented as follows:

$$L = 100 - PSNR(I, D) = 100 - 20 \cdot \log_{10} \left(\frac{MAX_I}{MSE} \right) \quad (5)$$

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i, j) - D(i, j)\|^2 \quad (6)$$

where I is the input image, D is the denoised image, m and n represent the length and width of the image pixel size, $I(i, j)$ and $D(i, j)$ represent the pixel values at the (x, y) coordinates of the input image and the denoised image, respectively. When the similarity between the two images is higher, the MSE value is smaller, the PSNR value is larger, and the loss L is smaller; on the contrary, when the loss L value becomes larger, it indicates that the similarity between the denoised image and the original image is lower. When the loss L of the image information falls below a certain threshold, it means that the filtering process is causing a loss of information in the image. Therefore, the iterative filtering should be stopped and the image obtained from the previous round of filtering should be output. The entire algorithm flow of the image feature enhancement module can be summarized as in Algorithm 1.

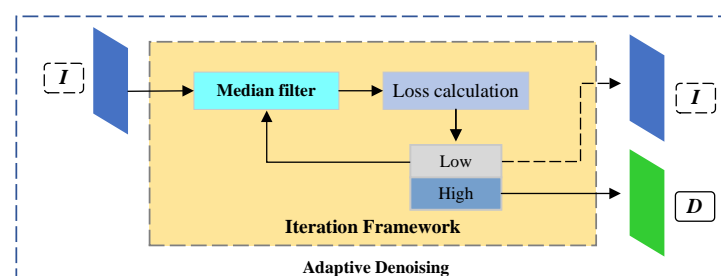


Figure 5. Schematic diagram of the adaptive denoising module.

Algorithm 1: Low-light image feature enhancement algorithm.

Input: Low-light image I ;
Output: Feature enhanced image D ;

```

1 Initialization:  $I_0 = I$ ;
2 Compute the light condition level  $L^w(I_0)$  using Equation (2);
3 if  $L^w(I) < p$  then
4   Adjust the light condition of a low-light image using SCI net;
5   for  $i = 0, 1, \dots, m$  do
6     Denoizing the image  $I_i$  to  $D_i$  using the lightweight filter;
7     Compute information loss  $L(I_0, D_i)$  using Equation (5);
8     if  $L(I_0, D_i) < q$  then
9       Let  $I_{i+1} = D_i$ ;
10    end
11    else
12      Return  $D = D_i$ ;
13    end
14  end
15  Return  $D = D_m$ 
16 end
17 else
18   Return  $D = I_0$ 
19 end

```

Remark 2. The lightweight filter used in Algorithm 1 is a median filter with a window size of 3×3 . In Algorithm 1, p is the threshold of illuminance intensity and is set to 0.148 referring to [52]. q is the threshold of image information loss. It is generally believed that the image quality is poor when the PSNR value is less than 30 dB. Therefore, we set the threshold q to 70 dB.

3.3. Bounding Box Prediction Network

To predict bounding boxes and adapt to scale changes in the target, the algorithm utilizes a region proposal network (RPN) [42]. As shown in Figure 6, the RPN includes two branches: a classification branch that differentiates between foreground and background, and a regression branch that calculates four position offset parameters relative to the anchor. After implementing the feature enhancement module and the backbone network, the template feature map $\varphi(z)$ and the search area feature map $\varphi(x)$ are obtained. The two feature maps are convolved in two branches. If there are k anchors, the classification branch and the regression branch output score maps with $2k$ and $4k$ channels, respectively:

$$\begin{aligned} A_{w \times h \times 2k}^{cls} &= [\varphi(x)]_{cls} * [\varphi(z)]_{cls} \\ A_{w \times h \times 4k}^{reg} &= [\varphi(x)]_{reg} * [\varphi(z)]_{reg} \end{aligned} \quad (7)$$

where $\varphi(z)$ is used as the convolution kernel and $*$ represents the convolution operation.

During network training, the two branches of the RPN compute loss independently, and the classification loss is determined using cross-entropy loss. p_i is the probability that the anchor is predicted as a positive sample, and p_i^* is the classification label of the anchor. If the IOU between the anchor and the true annotation box exceeds 0.6, the anchor is labeled as a positive sample with a label of 1; otherwise, if the IOU is less than 0.3, it is labeled as a negative sample with a label of 0.

$$p_i^* = \begin{cases} 0, & IOU > 0.6 \\ 1, & IOU < 0.3 \end{cases} \quad (8)$$

Then the classification loss can be written as:

$$L_{cls}(p_i, p_i^*) = -\log[p_i^* p_i + (1 - p_i^*)(1 - p_i)] \quad (9)$$

Regression uses normalized coordinate smooth L_1 loss. If A_x, A_y, A_w, A_h are used to represent the center coordinates and size of the anchor box, and T_x, T_y, T_w, T_h are used to represent the true center coordinates and size of the annotated target, the normalized distance is:

$$\begin{aligned}\delta[0] &= \frac{T_x - A_x}{A_w}, \delta[1] = \frac{T_y - A_y}{A_h} \\ \delta[2] &= \ln \frac{T_w}{A_w}, \delta[3] = \ln \frac{T_h}{A_h}\end{aligned}\quad (10)$$

Then use them to calculate the smooth L_1 loss:

$$\text{smooth}_{L_1}(x, \sigma) = \begin{cases} 0.5\sigma^2 x^2, & |x| < \frac{1}{\sigma^2} \\ |x| - \frac{1}{2\sigma^2}, & |x| \geq \frac{1}{\sigma^2} \end{cases}\quad (11)$$

The regression loss can be written as:

$$L_{\text{reg}} = \sum_{i=0}^3 \text{smooth}_{L_1}(\delta[i], \sigma)\quad (12)$$

The final loss function is:

$$\text{loss} = L_{\text{cls}} + \lambda L_{\text{reg}}\quad (13)$$

where λ is a hyperparameter used to balance the two parts of the loss.

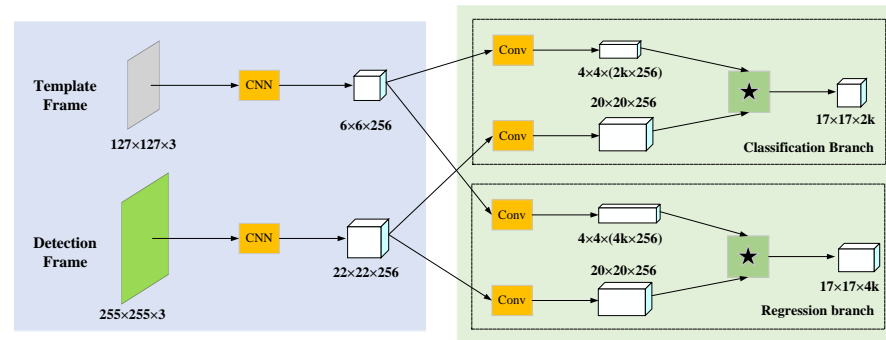


Figure 6. Structure of region proposal network.

3.4. Dynamic Template Tracking Mechanism

In typical Siamese network tracking algorithms, the initial frame's cropped target image is usually used as a fixed template [36,41], and is not updated even if the target's features change during subsequent tracking. This makes the algorithm susceptible to tracking drift when facing challenges such as occlusion, deformation, and changes in lighting. These issues frequently arise in low-light tracking tasks, where the fixed template tracking mechanism severely limits the algorithm's tracking performance under low-light conditions. To address this problem, this paper proposes a dynamic template tracking algorithm that updates templates dynamically by fusing reference templates and basic templates. This enhances the algorithm's adaptability to challenges in target feature changes during the tracking process and improves tracking accuracy and robustness under low-light conditions.

3.4.1. Template Update Method

The template update process is illustrated in Figure 7. The first step is to select the target features in the initial frame as the basic template, which is assumed to contain the most essential target features and is therefore considered the most credible. Since the target's features continue to change based on the basic template during tracking, it is necessary to update the template to make it as close as possible to the current target's features. In the second step, the target features in the previous frame are selected as the reference template.

As the target features become closer to the current frame, their similarity to the current target's features increases. Therefore, the target's features in the previous frame are best suited to reflect the changes in the target's features. The third step involves fusing the reference template with the basic template to obtain a new template that is most likely to be similar to the current target's features. This new template is used as the tracking template for the next frame. Since the true position of the target in the current frame is unknown during tracking, templates are cropped based on the tracking results from the previous frame.

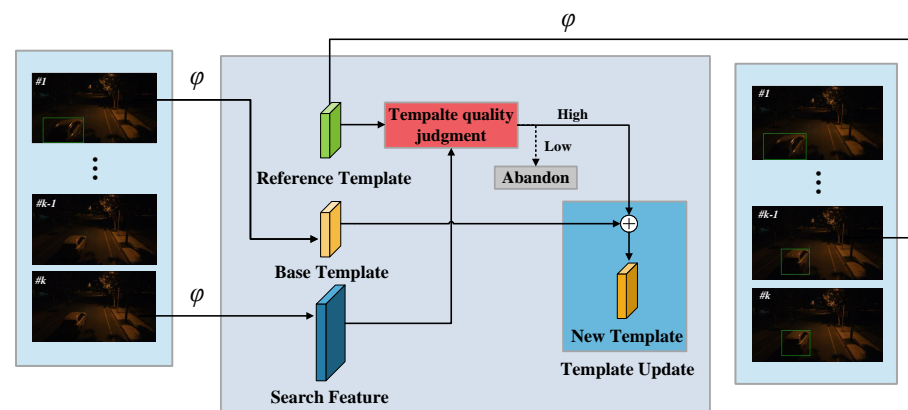


Figure 7. Schematic diagram of the template update process.

3.4.2. Template Quality Judgment Strategy

In the tracking process, occlusions often occur, and the reference template may contain interference information. Updating the template with this information can reduce its quality and negatively affect tracking performance. Additionally, frequent updates can also result in time loss. To address this issue, the proposed method filters the template before updating it and stops updating it when its quality is deemed poor. Drawing inspiration from a previous work [60], the Average Peak-to-Correlation Energy (APCE) is used to quantify the degree of occlusion of the target. Its calculation formula is as follows:

$$APCE = \frac{|F_{\max} - F_{\min}|^2}{\text{mean} \left(\sum_{w,h} (F_{w,h} - F_{\min})^2 \right)} \quad (14)$$

where F_{\max} , F_{\min} , $F_{w,h}$ and, respectively, represent the maximum value, minimum value, and corresponding value at coordinate (w, h) of the response map.

APCE is a measure of the variability of the response map and reflects the level of confidence in detecting the target. As illustrated in Figure 8, when the response map shows a clear and sharp peak with low noise, indicating the target is clearly detected within the search range, APCE increases, and the response map shows a single peak with smooth distribution. Conversely, if the target is occluded or missing, APCE decreases significantly. Based on this feature, the paper employs the APCE value to determine the extent of target occlusion and filter out high-quality templates. In the feature fusion process, the basic template is considered to be more reliable, and therefore, the reference template features are used to correct it while retaining its primary features. The inference process of the dynamic template Siamese network is outlined in Algorithm 2.

Remark 3. When filtering templates, if the APCE value for the current frame exceeds the average APCE value of previous historical frames ($APCE_{avg}$), it is deemed that the quality of the reference template is high, and thus, the tracking template may be updated.

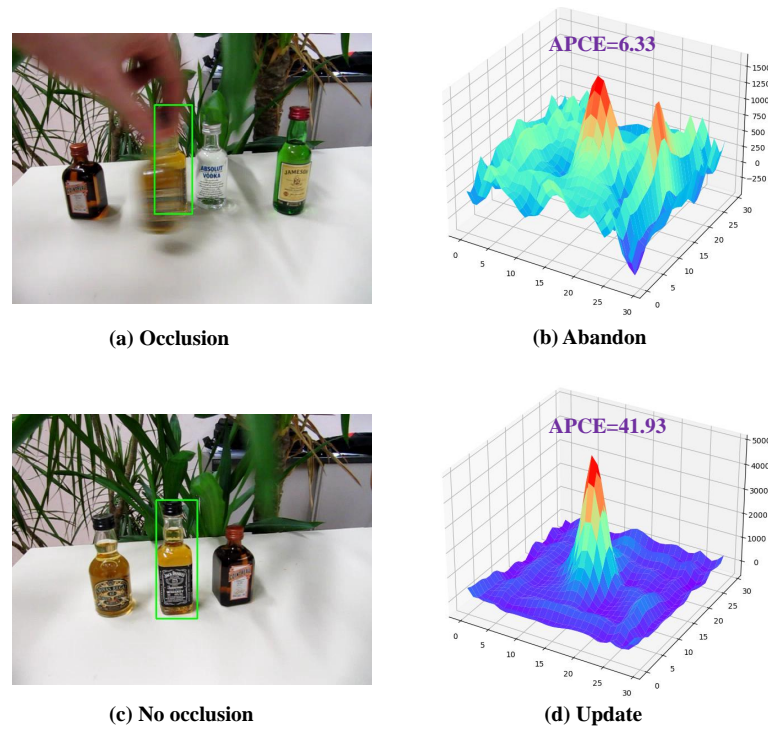


Figure 8. Effect diagram of the APCE.

Algorithm 2: Dynamic template update algorithm.

Input: The first frame I_0 and the ground truth (x_0, y_0, w_0, h_0) of the target;

Output: The predicted bounding box (x_i, y_i, w_i, h_i) of the target in the i th frame;

- 1 Crop the base template region and extract the feature T_b from the first frame according to the ground truth;
 - 2 Initialization: $T_0 = T_b$, $APCE_{avg} = 0$;
 - 3 **for** $i = 1, 2, \dots, n$ **do**
 - 4 Crop the search region and extract the feature X_i from the current frame according to the last frame's tracking result;
 - 5 Compute the classification scores and regression scores using T_i and X_i ;
 - 6 Compute the response map M_i by convolving X_i with T_b as convolution kernel;
 - 7 Compute the APCE of M_i using Equation (14);
 - 8 **if** $APCE_{M_i} > APCE_{avg}$ **then**
 - 9 Crop the reference template region and extract the feature T_r from the current frame;
 - 10 Update the template using $T_i = T_b + T_r$;
 - 11 **end**
 - 12 **else**
 - 13 Abandon the reference template T_r and let $T_i = T_b$;
 - 14 **end**
 - 15 Update $APCE_{avg}$;
 - 16 **end**
-

4. Experiment and Discussion

This section primarily focuses on the experimental validation process of the algorithm. First, it introduces the experimental environment and training methodologies. Following this, performance testing and analysis are conducted using the public datasets UAVDark135 [52], UAVDark70 [50], DarkTrack2021 [8], NAT2021 [51], and NAT2021L [51]. The experimental results are objectively compared from two standpoints: quantitative and qualitative analyses. The quantitative analysis employs two evaluation metrics: tracking precision and success rate. Tracking precision is defined as the ratio of the number of frames in which the average Euclidean distance between the target's center position tracked by

the algorithm and the manually labeled true position in the video sequence is below a specified threshold to the total number of frames. The tracking success rate, on the other hand, is determined by the proportion of frames in which the overlap score between the target bounding box predicted by the algorithm and the manually labeled true bounding box in the video sequence surpasses a predetermined threshold to the overall number of frames. Qualitative analysis involves annotating the tracking results of algorithms in video images and comparing the tracking effects more intuitively based on visual impact. Lastly, ablation experiments are conducted by dissecting the algorithm to verify the effectiveness of each component. This study designs two versions of algorithms that utilize AlexNet and ResNet50 as backbone networks, respectively, to demonstrate the performance of algorithms under varying network complexities.

4.1. Experimental Details

The algorithmic environment in this study is constructed on a computer hardware platform featuring an Intel® Xeon® Silver 4110 2.1 GHz CPU and an NVIDIA GeForce RTX2080 GPU and is implemented using PyTorch programming. The comparison algorithm is replicated based on the original text and assessed under the same conditions.

The model training employs a phased approach. First, the original tracking network, without incorporating a low-light enhancement network, is trained independently. The data utilized in this training phase is sourced from the ImageNet VID [61], GOT10K [5], and YouTube-BB [6] datasets. The pre-training parameters of the feature extraction network are derived from the model trained on the ImageNet dataset. During training, the template image size is set at 127 px × 127 px, and the search image size is 255 px × 255 px. The initial learning rate is established at 0.01, the batch size is 16, and the Adam optimizer is employed. Subsequently, the low-light enhancement module is integrated into the network to achieve a comprehensive network structure and attain end-to-end performance. The SCI network parameters are loaded from models trained on the LOL [23] and LSRW [62] datasets.

4.2. Quantitative Analysis

To validate the effectiveness of the proposed algorithm, this study conducts performance evaluations on four low-light datasets: UAVDark135, UAVDark70, DarkTrack2021, NAT2021, and NAT2021L, and compares its performance with other contemporary tracking algorithms. This research considers a variety of target tracking algorithms for performance comparison, which include shallow network-based methods (SiamAPN [63], SiamAPN++ [64], HiFT [65], SiamRPN), deep network-based methods (SiamRPN++, SiamBAN, SiamCAR); approaches employing anchor boxes (SiamAPN, SiamAPN++, HiFT, SiamRPN, SiamRPN++), anchor-free techniques (SiamBAN, SiamCAR); and nighttime tracking methods (UDAT-CAR [51], UDAT-BAN [51]). The actual performance of the algorithm is assessed through comprehensive comparison.

4.2.1. UAVDark135

UAVDark135 comprises 135 video sequences captured by standard drones at night. It encompasses various tracking scenarios, such as intersections, T-junctions, roads, highways, and a diverse range of tracking objects, including people, boats, buses, cars, trucks, athletes, and houses. The dataset has a total of 125,466 frames, with an average of 929 frames, a maximum of 4571 frames, and a minimum of 216 frames, making it suitable for large-scale evaluation. UAVDark135 offers five common challenge attributes in drone tracking, namely viewpoint change (VC), fast motion (FM), low resolution (LR), occlusion (OCC), and illumination variation (IV).

Overall evaluation: Figure 9 presents the tracking precision curve and success rate curve derived from the experimental results. It is evident that the SiamLT_Res50 proposed in this study has achieved the highest level in terms of tracking precision and success rate, with a tracking precision of 0.707 and a success rate of 0.550. In comparison to the second-

ranked SiamCAR, the precision is enhanced by 6.6%, and the success rate is improved by 5.0%. Simultaneously, our AlexNet version of the algorithm, SiamLT_Alex, has also attained the best performance among algorithms of the same level, surpassing SiamAPN, SiamAPN++, HiFT, and SiamRPN. Relative to SiamRPN, its performance has significantly improved, with an increase of 22.8% and 23.0% in precision and success rate, respectively.

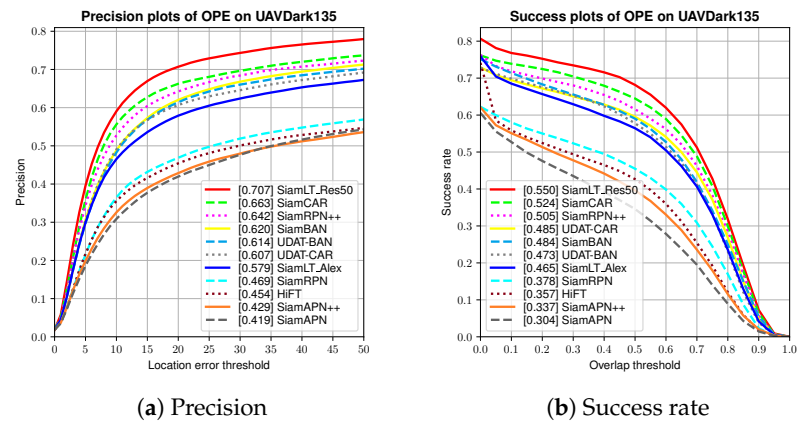


Figure 9. Overall evaluation on UAVDark135 dataset.

Attribute evaluation: Figures 10 and 11 display the test results under five distinct attributes. It can be observed that the SiamLT_Res50 method proposed in this study has attained the highest level in multiple attributes. The precision values are 0.684 (FM), 0.683 (IV), 0.716 (LR), 0.666 (OCC), and 0.674 (VC), while the success rate values are 0.536 (FM), 0.523 (IV), 0.519 (LR), 0.505 (OCC), and 0.547 (VC). Among these, the performance in fast motion (FM), occlusion (OCC), and viewpoint change (VC) are the most exceptional, with precision surpassing the second place by 8.7%, 8.8%, and 8.4%, respectively.

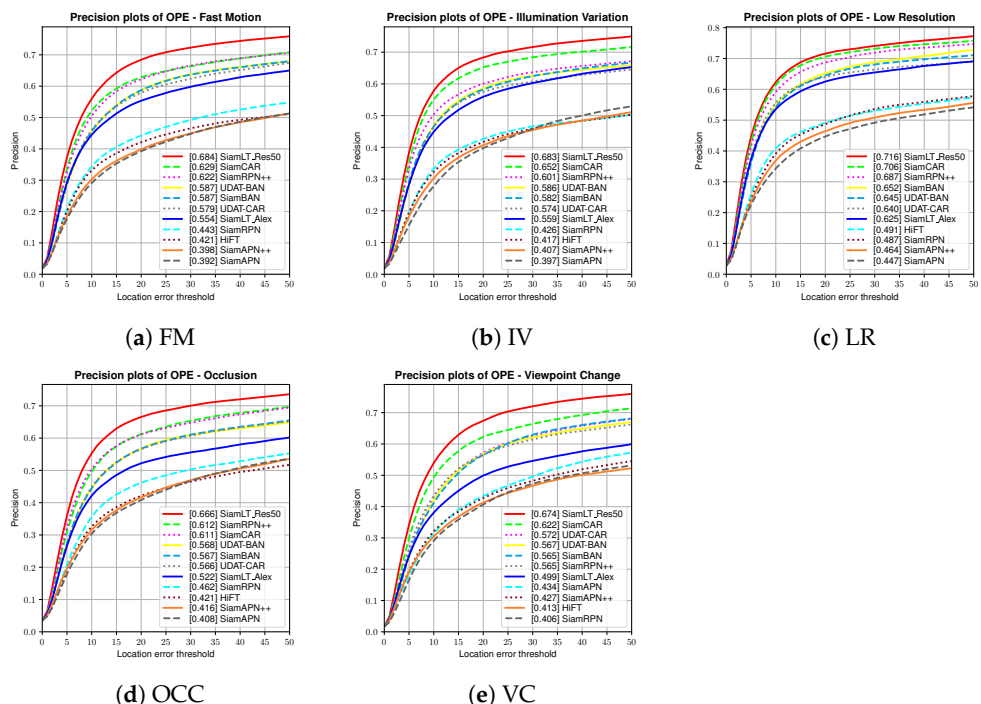


Figure 10. Precision comparison of attributes on UAVDark135 dataset.

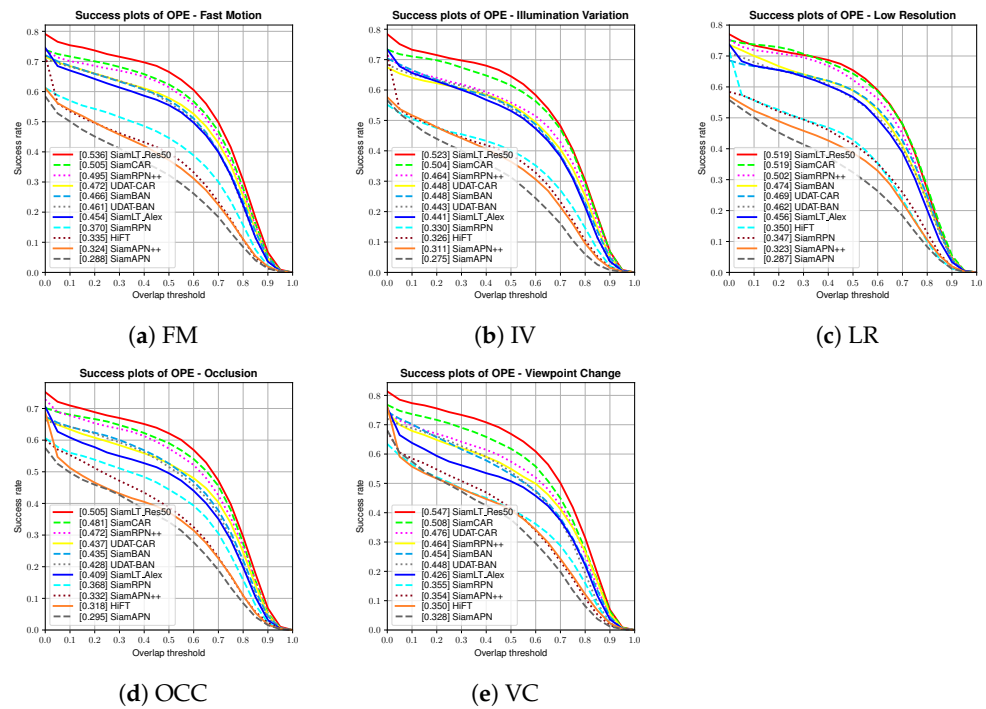


Figure 11. Success rate comparison of attributes on UAVDark135 dataset.

4.2.2. UAVDark70

UAVDark70 comprises 70 manually annotated sequences captured at night using professional-grade drones. This dataset takes into account the unique tracking challenges associated with drone footage, such as low resolution (LR), fast motion (FM), illumination variation (IV), viewpoint change (VC), and occlusion (OCC).

Overall evaluation: Figure 12 displays the tracking precision curve and success rate curve based on the experimental results. It is evident that the SiamLT_Res50 method proposed in this study has achieved the highest performance in both tracking precision and success rate, with values of 0.770 and 0.566, respectively. In comparison to the second-ranked SiamRPN++, the precision improved by 6.4%, and the success rate increased by 7.2%. Simultaneously, our Alexnet-based algorithm, SiamLT_Alex, has outperformed its counterparts, surpassing SiamAPN, SiamAPN++, HiFT, and SiamRPN. Relative to SiamRPN, its performance has significantly improved, with increases of 15.6% and 18.0% in precision and success rate, respectively.

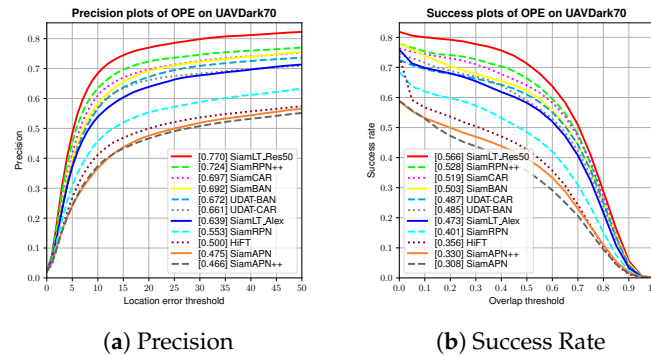


Figure 12. Overall evaluation on UAVDark70 dataset.

Attribute evaluation: Figures 13 and 14 illustrate the test results under five distinct attributes. It is clear that the SiamLT_Res50 method proposed in this study has attained the highest performance in multiple attributes. The precision values are 0.735 (FM), 0.838 (IV),

0.777 (LR), 0.769 (OCC), and 0.738 (VC), while the success rate values are 0.535 (FM), 0.580 (IV), 0.573 (LR), 0.571 (OCC), and 0.536 (VC). Notably, the performance in illumination variation (IV) and occlusion (OCC) is more remarkable, with precision surpassing the second-best results by 8.7% and 8.3%, respectively, and success rates exceeding the second-best results by 9.2% and 8.8%, respectively.

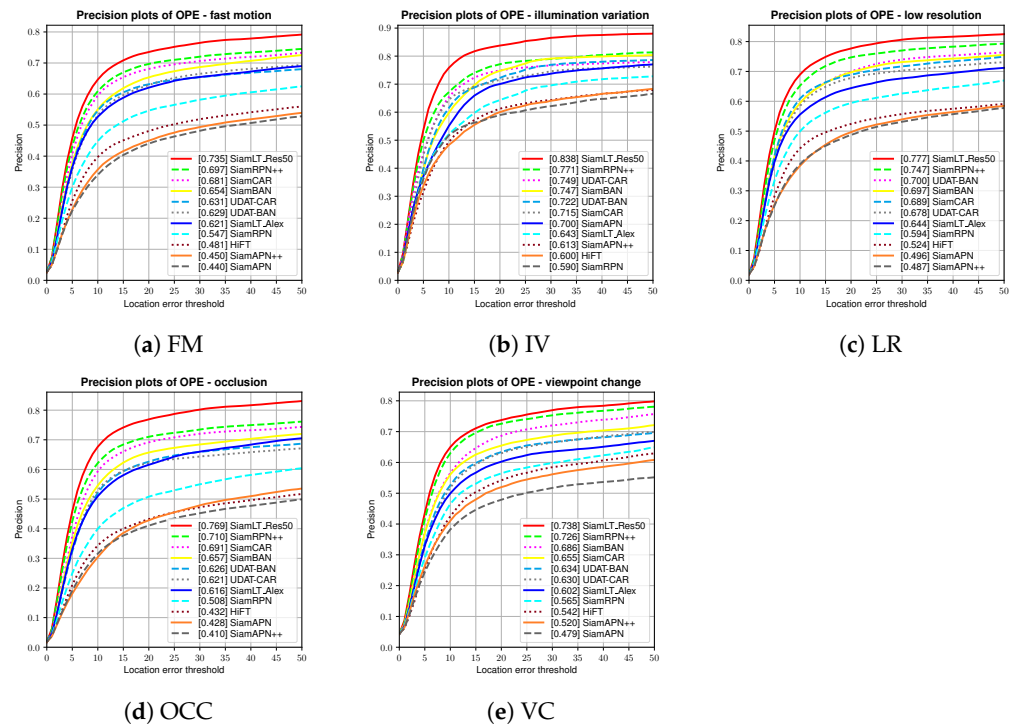


Figure 13. Precision comparison of attributes on UAVDark70 dataset.

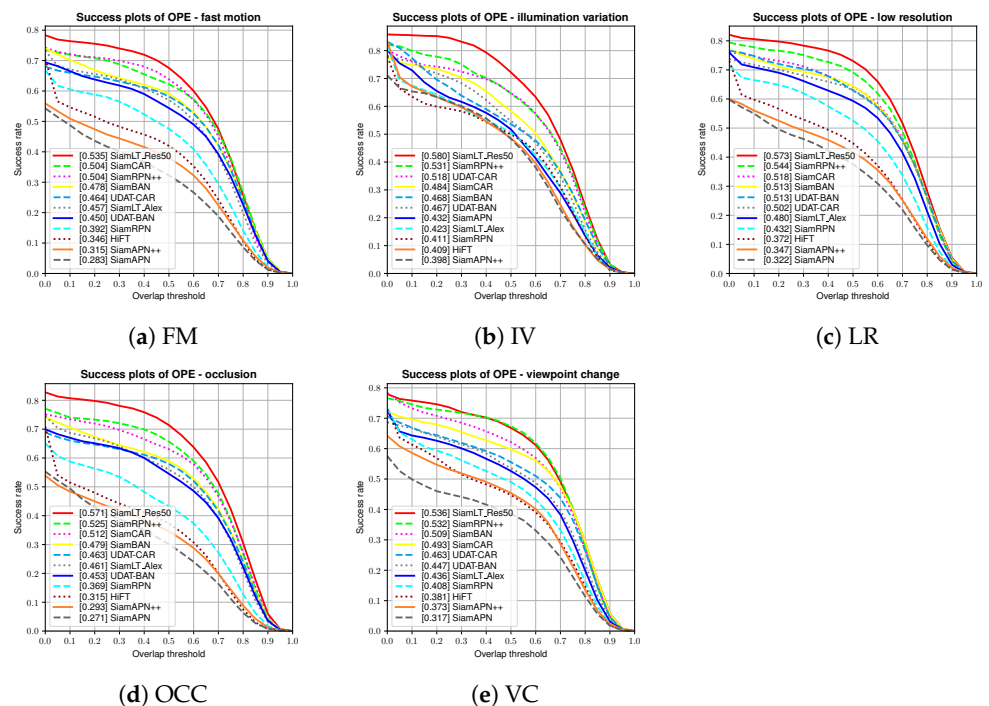


Figure 14. Success rate comparison of attributes on UAVDark70 dataset.

4.2.3. DarkTrack2021

The DarkTrack2021 dataset comprises 110 challenging sequences, totaling 100,377 frames. The sequences' shortest, longest, and average lengths are 92 frames, 6579 frames, and 913 frames, respectively. The tracking objects encompass people, buses, cars, trucks, trams, dogs, buildings, and more, covering a diverse range of real-world drone night tracking tasks. DarkTrack2021 involves numerous scenes with various challenges, including perspective changes, fast motion, occlusion, low resolution, low brightness, and out-of-field-of-view occurrences. This dataset does not categorize video sequences into attributes.

Figure 15 presents the tracking precision curve and success rate curve based on the experimental results. It is evident that the SiamLT_Res50 proposed in this study has achieved the highest level in terms of tracking precision and success rate, with values of 0.659 and 0.505, respectively. In comparison to SiamCAR, the precision and success rate are improved by 6.1% and 4.3%, respectively. Simultaneously, it is noteworthy that our AlexNet version of the algorithm, SiamLT_Alex, has also attained a high level, ranking second in precision and fifth in success rate, surpassing the ResNet-based SiamBAN and UDAT-BAN algorithms. This demonstrates that the algorithm framework proposed in this study can still perform well when combined with shallow networks. It proves that the low-light image feature enhancement and dynamic template Siamese network mechanism proposed in this research can effectively augment the network's capacity to extract low-light image features and address challenges in low-light scenarios.

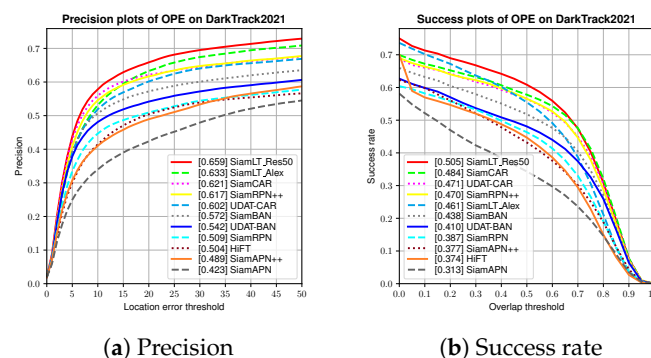


Figure 15. Evaluation on DarkTrack2021 dataset.

4.2.4. NAT2021

The NAT2021 dataset comprises 180 nighttime aerial tracking sequences, categorized by diverse targets (e.g., cars, trucks, people, groups, buses, buildings, and motorcycles) and activities (e.g., cycling, skating, running, and ball sports), totaling over 140,000 frames. The test sequences incorporate 12 distinct attributes: aspect ratio change (ARC), background clutter (BC), fast motion (FM), full occlusion (FOC), out-of-view (OV), similar object (SOB), viewpoint change (VC), illumination variation (IV), and low ambient intensity (LAI). To better comprehend the impact of illumination on tracking algorithms, the dataset introduces the novel attribute of low ambient intensity (LAI). The average pixel intensity of the local region centered on the object is calculated as the illuminance intensity for the current frame, and the average illuminance level of the sequence is regarded as the ambient intensity of the tracking scene. Sequences with ambient intensity below 20, which make object identification difficult for the naked eye, are labeled as LAI attributes.

Overall evaluation: Figure 16 illustrates the tracking precision curve and success rate curve based on the experimental outcomes. The SiamLT_Res50 method proposed in this study attains the highest levels in both tracking precision (0.706) and success rate (0.491). In comparison to the second-ranked UDAT-CAR, the precision has improved by 4.3%, and the success rate by 3.2%. Concurrently, our Alexnet version of the algorithm, SiamLT_Alex, also achieves the best performance among algorithms of a similar magnitude, outperforming SiamAPN, SiamAPN++, HiFT, and SiamRPN. The precision and success rate has increased by 1.0% and 2.7%, respectively, compared to SiamRPN.

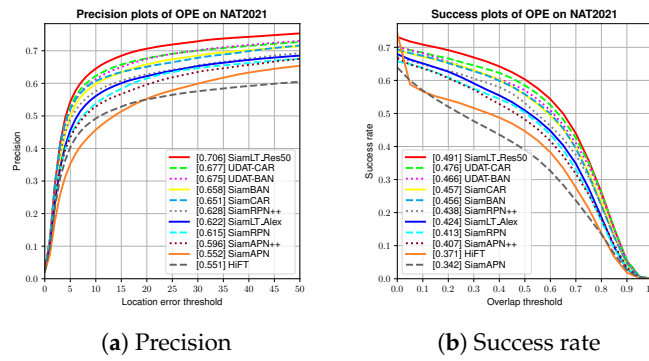


Figure 16. Overall evaluation on NAT2021 dataset.

Attribute evaluation: Figures 17 and 18 display the test results under five distinct attributes. The proposed SiamLT_Res50 method achieves the highest levels in multiple attributes, with particularly exceptional performance in fast motion (FM), out-of-view (OV), and low ambient intensity (LAI) attributes. The precision surpasses the second place by 7.6% (FM), 8.5% (OV), and 6.4% (LAI), respectively, and the success rate exceeds the second place by 5.1% (FM), 7.0% (OV), and 2.9% (LAI), respectively. Notably, the excellent performance in the low ambient intensity (LAI) attribute demonstrates the unique advantage of the algorithm proposed in this study for low-light tracking tasks.

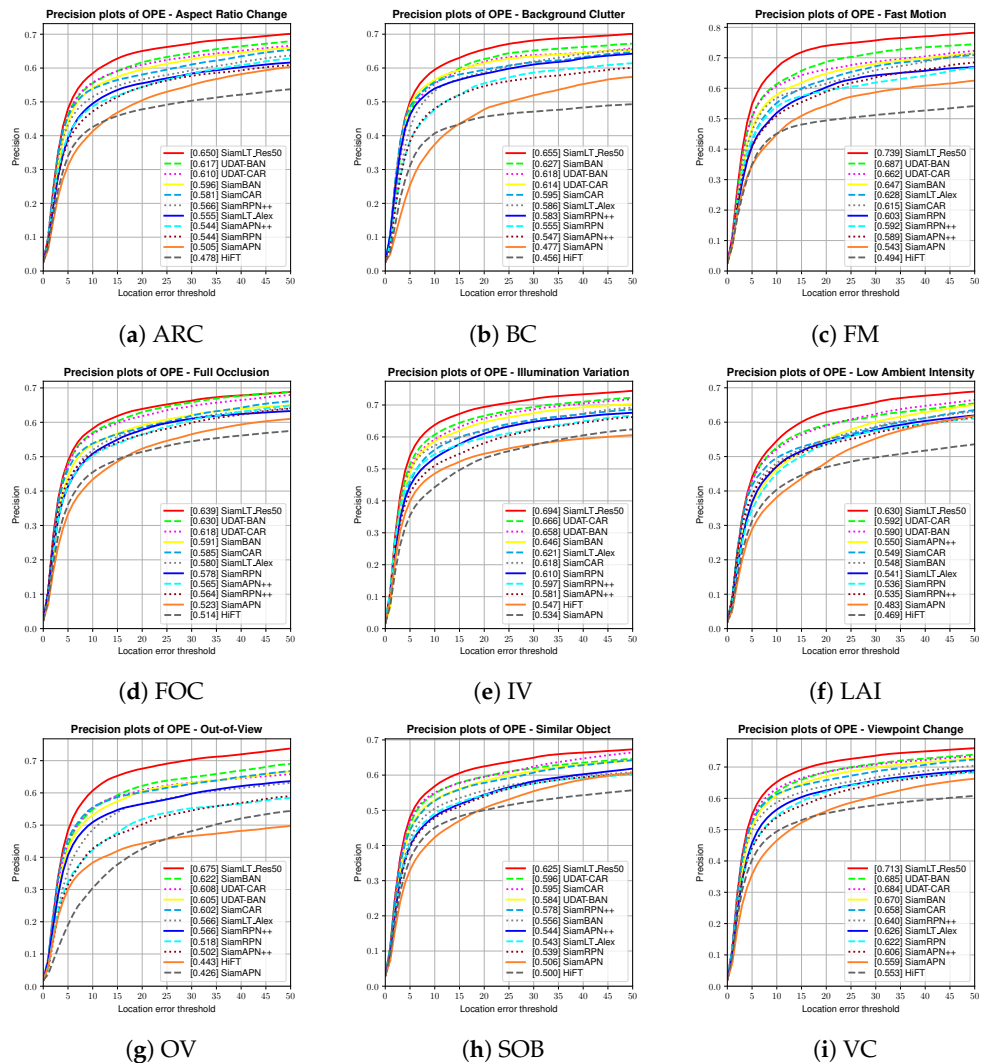


Figure 17. Precision comparison of attributes on NAT2021 dataset.

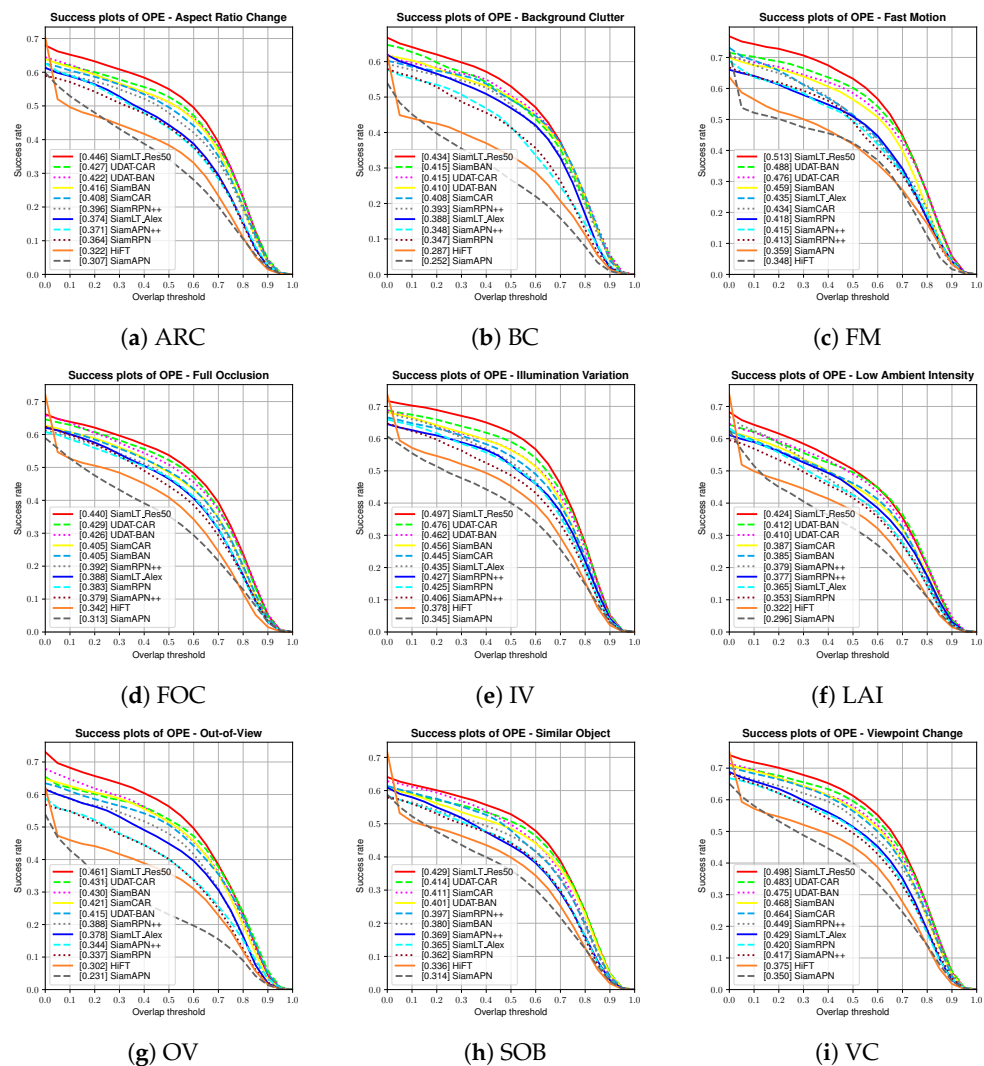


Figure 18. Success rate comparison of attributes on NAT2021 dataset.

4.2.5. NAT2021L

The NAT2021L dataset comprises 23 long video sequences, with sequence categories including various targets (e.g., cars, trucks, pedestrians, crowds, buildings, and motorcycles) and activities (e.g., cycling, skating, and running), involving challenges such as aspect ratio change (ARC), background clutter (BC), fast motion (FM), full occlusion (FOC), out-of-view (OV), similar object (SOB), viewpoint change (VC), illumination variation (IV), and low ambient intensity (LAI). Furthermore, a single video length exceeded 1400 frames, making it more challenging.

Overall evaluation: Figure 19 illustrates the tracking precision curve and success rate curve based on the experimental outcomes. The SiamLT_Res50 method proposed in this study attains the highest levels in both tracking precision (0.579) and success rate (0.420). In comparison to the second-ranked UDAT-CAR, the precision has improved by 9.5%, and the success rate by 7.4%. Concurrently, our Alexnet version of the algorithm, SiamLT_Alex, also achieves the best performance among algorithms of a similar magnitude, outperforming SiamAPN, SiamAPN++, HiFT, and SiamRPN. The precision and success rate has increased by 10.0% and 10.7%, respectively, compared to HiFT.

Attribute evaluation: Figures 20 and 21 display the test results under five distinct attributes. The proposed SiamLT_Res50 method achieves the highest levels in multiple attributes, with particularly exceptional performance in fast motion (FM), illumination variation (IV), and out-of-view (OV) attributes. The precision surpasses the second place

by 12.0% (FM), 9.9% (IV), and 15.1% (OV), respectively, and the success rate exceeds the second place by 13.9% (FM), 10.8% (IV), and 10.6% (OV), respectively.

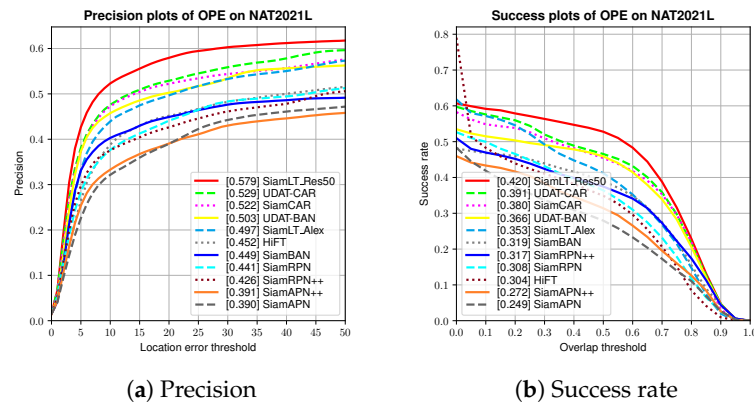


Figure 19. Overall evaluation on NAT2021L dataset.

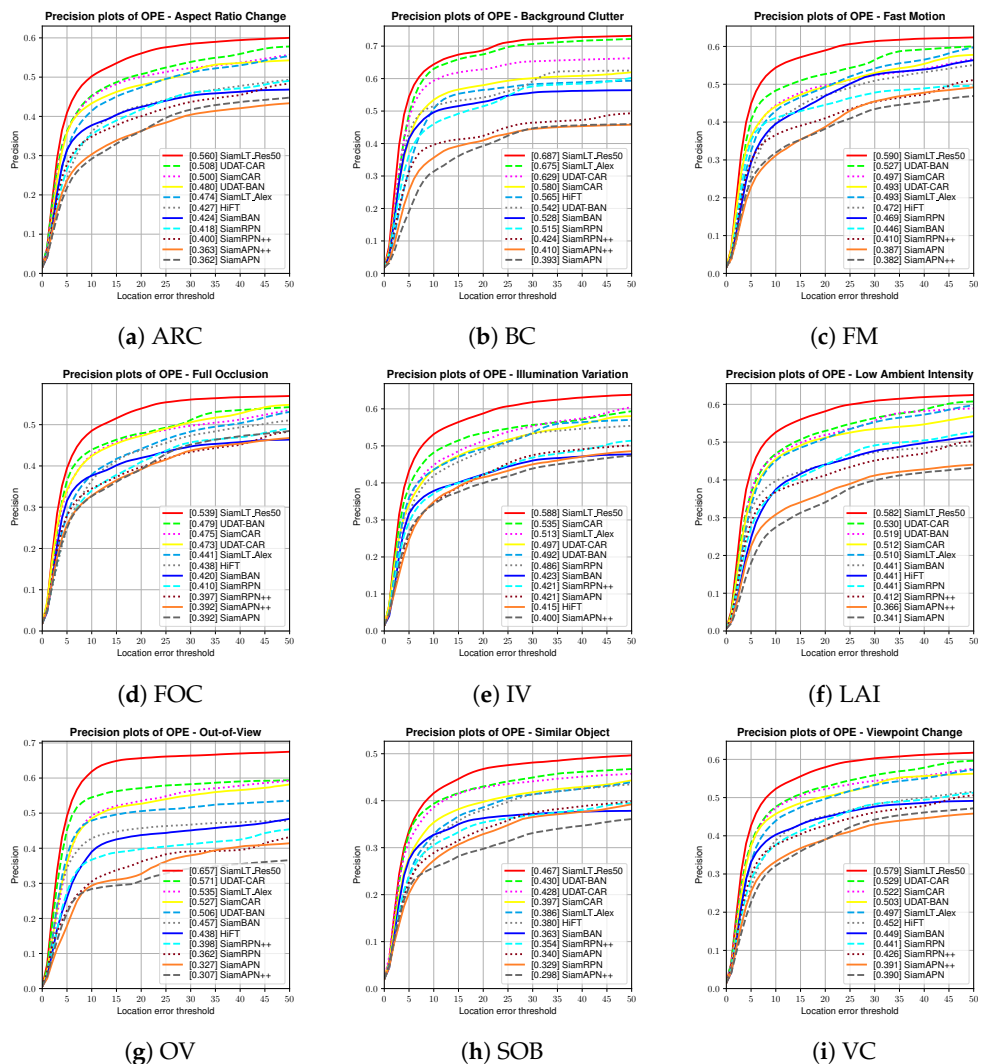


Figure 20. Precision comparison of attributes on NAT2021L dataset.

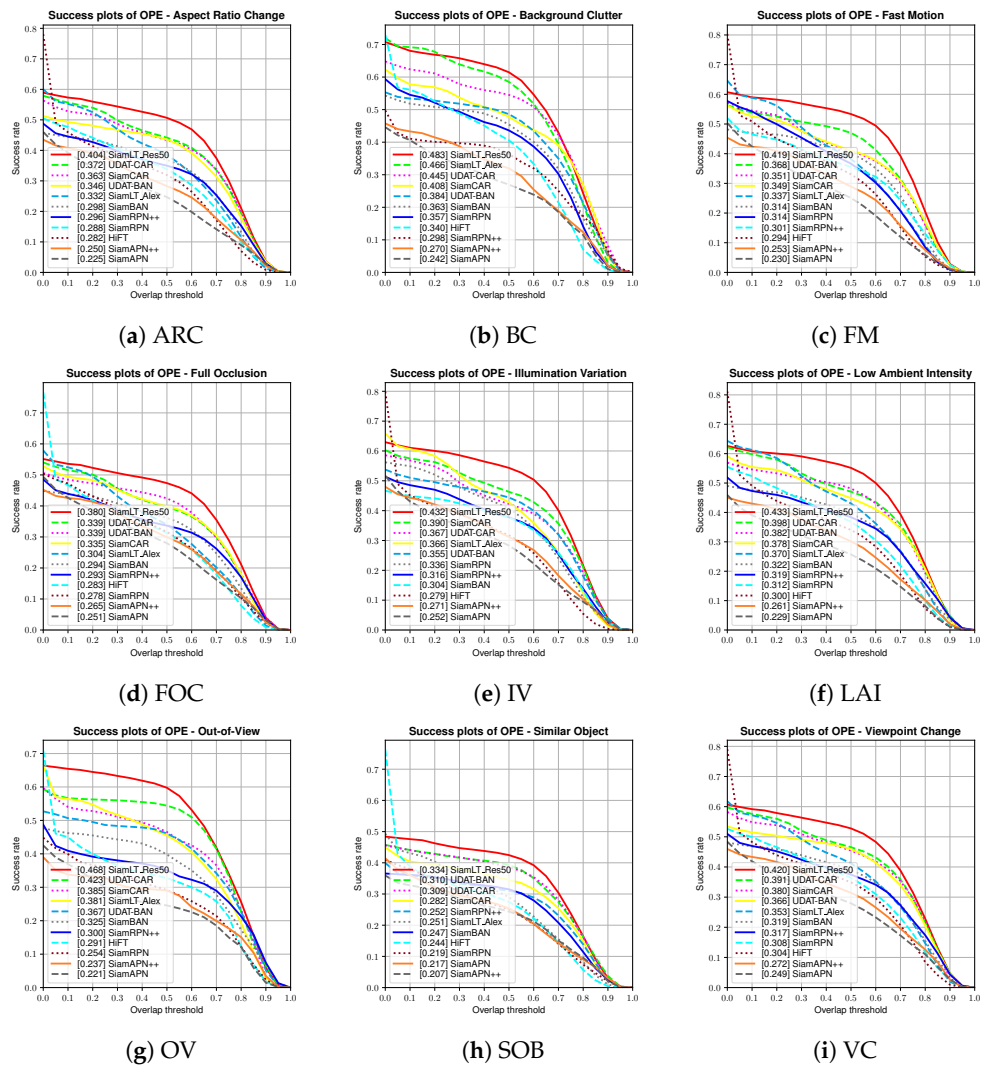


Figure 21. Success rate comparison of attributes on NAT2021L dataset.

4.3. Qualitative Analysis

To visually compare the tracking performance of the algorithm, the tracking results are marked with a bounding box in the image. This study selects four video sequences from the UAVDark135 and DarkTrack2021 datasets to illustrate the experimental results as follows:

- (1) **bike6:** This video sequence involves a person riding a bicycle as the tracking target, with interference from another similar target also riding a bicycle around the target. The first row of the visualization results in Figure 22 shows that other tracking algorithms fail to identify and drift when the two targets are in close proximity. Conversely, the SiamLT_Res50 algorithm proposed in this paper maintains stable tracking of the target.
- (2) **group1:** This video sequence features two crowds of pedestrians, with the tracking target being a pedestrian from the right crowd. The second row of the visualization results in Figure 22 demonstrates that some tracking algorithms lost the target and tracked the interfering target next to it during the tracking process. The SiamLT_Res50 algorithm proposed in this paper has strong anti-interference ability and consistently tracks the correct target.
- (3) **running:** This video sequence shows two running people, with the tracking target being the person on the left. The third row of Figure 22 presents the visualization results, indicating that some algorithms drift and track the wrong target as the relative

position of the two targets changes. In contrast, the SiamLT_Res50 algorithm proposed in this paper accurately tracks the correct target throughout the tracking process.

- (4) **person_22**: This video sequence depicts several people walking in different directions, with the tracking target being one of them walking from right to left. The fourth row of visualization results in Figure 22 reveals that some algorithms fail to overcome the challenge of similar objects blocking each other during their movement and drift while tracking. The SiamLT_Res50 algorithm proposed in this paper achieves stable and accurate tracking of the correct target throughout the tracking process.

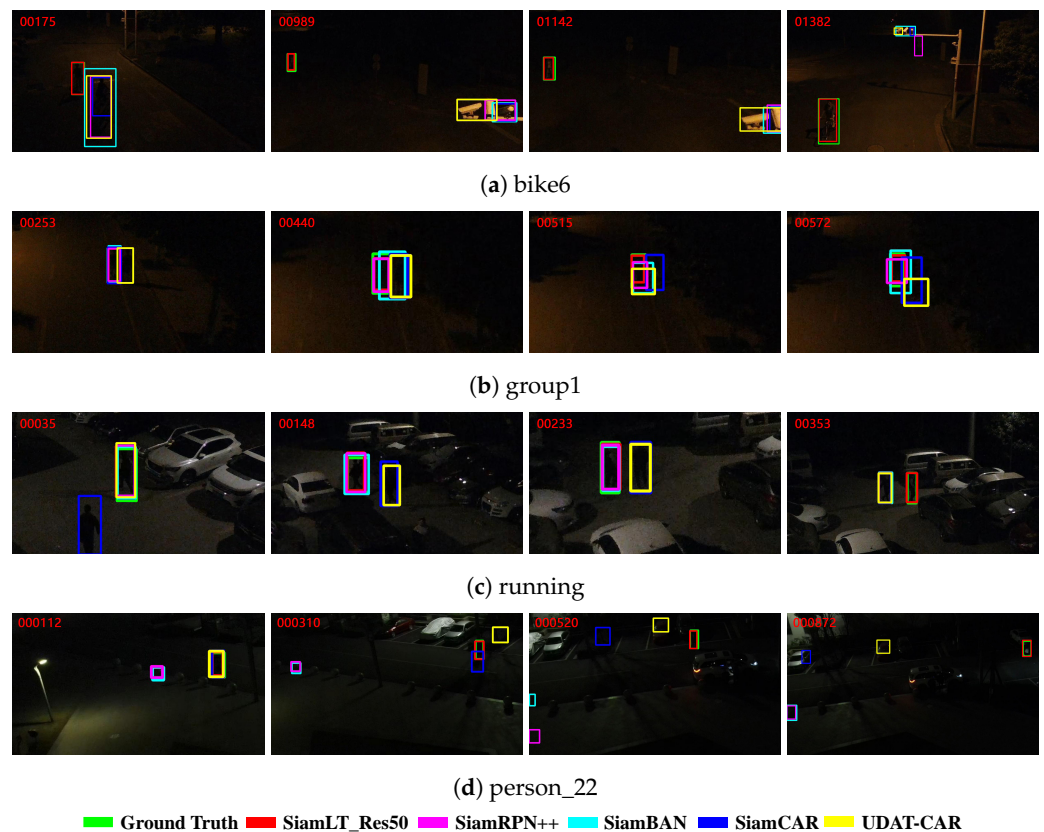


Figure 22. Visualization of experimental results.

4.4. Ablaton Study

To evaluate the individual effectiveness of each component of the proposed algorithm, this study decomposes the algorithm and conducts comparative experiments on the UAVDark135 and DarkTrack2021 datasets.

Table 1 presents the comparison results, where a \checkmark symbol denotes the usage of a module, a \times symbol indicates its non-usage, and the best results are marked in bold. The table indicates that the algorithm employing all modules achieves the best performance, with an accuracy increase of 10.1% and a success rate increase of 8.9% on the UAVDark135 dataset when compared to the baseline algorithm. Similarly, on the DarkTrack2021 dataset, the accuracy and success rate increased by 6.8% and 7.4%, respectively, compared to the baseline algorithm. Moreover, the addition of each module demonstrated some degree of enhancement in the algorithm's performance. These results imply that the low-light enhancement module (LE), image adaptive denoizing module (AD), dynamic template mechanism (DT), and APCE template screening strategy all contribute significantly to the algorithm's performance and play an irreplaceable role.

Table 1. Results of ablation experiments on modules.

LE	AD	DT	APCE	UAVDark135		DarkTrack2021	
				Precision	Success	Precision	Success
×	×	×	×	0.642	0.505	0.617	0.470
✓	×	×	×	0.657	0.510	0.630	0.478
✓	✓	×	×	0.660	0.515	0.643	0.493
✓	✓	✓	×	0.702	0.548	0.647	0.498
✓	✓	✓	✓	0.707	0.550	0.659	0.505

To further validate the reliability of two important thresholds p (illuminance intensity threshold) and q (information loss threshold), this paper conducted ablation experiments on the UAVDark70 dataset using the proposed method.

Upon observing the results in Table 2, it becomes apparent that when the value of p exceeds 0.5, the performance of the tracker remains unchanged. This suggests that our discriminator has entirely lost its functionality and cannot differentiate between light and dark conditions in the photos. From a starting point of 0.4, as the value of p decreases, the tracker's performance gradually improves. This indicates that the discriminator can distinguish certain photos, and this ability progressively strengthens until it reaches its peak at 0.148. However, at a value of p of 0.1, the tracker's performance begins to decline again. This signifies that the discriminator has become too lenient and now misclassifies some low-light photos as normal ones. Consequently, it can be inferred that the most suitable value for p lies between 0.1 and 0.2. Hence, the adoption of a threshold of 0.148 in [57] is both reasonable and effective.

Table 2. Results of ablation experiment on illuminance intensity threshold.

p	0.1	0.148	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
success rate	0.515	0.528	0.512	0.506	0.505	0.506	0.506	0.506	0.506	0.506
precision	0.709	0.724	0.705	0.698	0.698	0.700	0.700	0.700	0.700	0.700

The results in Table 3 indicate a notable decline in tracker performance when q is less than 40 dB or greater than 70 dB. This decline occurs because excessively small values of q cause a significant loss of key target features during the denoising process. Conversely, excessively large values fail to remove noise from the image, making it difficult to extract target feature information and resulting in decreased tracking performance. It can be observed that when the value of q ranges between 40 dB and 70 dB, the tracker's performance remains quite similar. However, in target tracking tasks, maintaining the integrity of target feature information is crucial. Therefore, adhering to the principle of preserving as much feature information as possible, we opt for setting the information loss threshold at 70 dB to achieve optimal tracking effectiveness.

Table 3. Results of ablation experiment on information loss threshold.

q	10	20	30	40	50	60	70	80	90
success rate	0.402	0.415	0.481	0.511	0.511	0.511	0.528	0.483	0.477
precision	0.622	0.637	0.652	0.702	0.702	0.709	0.724	0.676	0.667

Remark 4. In summary, the SiamLT algorithm proposed in this paper exhibits outstanding performance in low-light tracking tasks, and each module effectively improves the algorithm's tracking performance under low-light conditions.

5. Conclusions

This research article presents a novel UAV visual object-tracking algorithm for low-light environments. The proposed approach employs an iterative filtering framework to enhance low-light images and incorporates it into a Siamese network for feature extraction. Additionally, the authors improve the traditional fixed template mechanism of Siamese networks by introducing a dynamic template update strategy to handle target feature changes during tracking. The algorithm's efficacy is evaluated on multiple low-light UAV video datasets, namely UAVDark135, UAVDark70, DarkTrack2021, NAT2021, and NAT2021L, and the results demonstrate the proposed method's effectiveness in improving tracking precision and robustness under low-light conditions. The authors use a conventional RPN network for bounding box prediction, but in future work, they aim to explore tracking algorithms based on anchor-free methods to further enhance the algorithm's performance.

Author Contributions: L.S. and S.K. conceived of the idea and developed the proposed approaches. Z.Y. advised the research. D.G. helped edit the paper. B.F. improved the quality of the manuscript and the completed revision. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 62271193), the Aeronautical Science Foundation of China (No. 20185142003), Natural Science Foundation of Henan Province, China (No. 222300420433), Science and Technology Innovative Talents in Universities of Henan Province, China (No. 21HASTIT030), Young Backbone Teachers in Universities of Henan Province, China (No. 2020GGJS073), and Major Science and Technology Projects of Longmen Laboratory (No. 231100220200).

Data Availability Statement: Not applicable.

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

References

1. Xie, X.; Xi, J.; Yang, X.; Lu, R.; Xia, W. STFTTrack: Spatio-Temporal-Focused Siamese Network for Infrared UAV Tracking. *Drones* **2023**, *7*, 296. [\[CrossRef\]](#)
2. Memon, S.A.; Son, H.; Kim, W.G.; Khan, A.M.; Shahzad, M.; Khan, U. Tracking Multiple Unmanned Aerial Vehicles through Occlusion in Low-Altitude Airspace. *Drones* **2023**, *7*, 241. [\[CrossRef\]](#)
3. Yeom, S. Long Distance Ground Target Tracking with Aerial Image-to-Position Conversion and Improved Track Association. *Drones* **2022**, *6*, 55. [\[CrossRef\]](#)
4. Fan, H.; Bai, H.; Lin, L.; Yang, F.; Chu, P.; Deng, G.; Yu, S.; Huang, M.; Liu, J.; Xu, Y.; et al. Lasot: A high-quality large-scale single object tracking benchmark. *Int. J. Comput. Vis.* **2021**, *129*, 439–461. [\[CrossRef\]](#)
5. Huang, L.; Zhao, X.; Huang, K. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1562–1577. [\[CrossRef\]](#)
6. Real, E.; Shlens, J.; Mazzocchi, S.; Pan, X.; Vanhoucke, V. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5296–5305.
7. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F.F. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
8. Ye, J.; Fu, C.; Cao, Z.; An, S.; Zheng, G.; Li, B. Tracker meets night: A transformer enhancer for UAV tracking. *IEEE Robot. Autom. Lett.* **2022**, *7*, 3866–3873. [\[CrossRef\]](#)
9. Ye, J.; Fu, C.; Zheng, G.; Cao, Z.; Li, B. Darklighter: Light up the darkness for uav tracking. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 3079–3085.
10. Rahman, Z.u.; Jobson, D.J.; Woodell, G.A. Retinex processing for automatic image enhancement. *J. Electron. Imaging* **2004**, *13*, 100–110.
11. Fu, X.; Liao, Y.; Zeng, D.; Huang, Y.; Zhang, X.P.; Ding, X. A probabilistic method for image enhancement with simultaneous illumination and reflectance estimation. *IEEE Trans. Image Process.* **2015**, *24*, 4965–4977. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Fu, X.; Zeng, D.; Huang, Y.; Zhang, X.P.; Ding, X. A weighted variational model for simultaneous reflectance and illumination estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2782–2790.

13. Guo, X.; Li, Y.; Ling, H. LIME: Low-light image enhancement via illumination map estimation. *IEEE Trans. Image Process.* **2016**, *26*, 982–993. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Xu, L.; Yan, Q.; Xia, Y.; Jia, J. Structure extraction from texture via relative total variation. *ACM Trans. Graph. (TOG)* **2012**, *31*, 1–10. [\[CrossRef\]](#)
15. Li, M.; Liu, J.; Yang, W.; Sun, X.; Guo, Z. Structure-revealing low-light image enhancement via robust retinex model. *IEEE Trans. Image Process.* **2018**, *27*, 2828–2841. [\[CrossRef\]](#)
16. Hao, S.; Han, X.; Guo, Y.; Xu, X.; Wang, M. Low-light image enhancement with semi-decoupled decomposition. *IEEE Trans. Multimed.* **2020**, *22*, 3025–3038. [\[CrossRef\]](#)
17. Ren, Y.; Ying, Z.; Li, T.H.; Li, G. LECARM: Low-light image enhancement using the camera response model. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 968–981. [\[CrossRef\]](#)
18. Li, B.; Li, Q.; Zeng, Y.; Rong, Y.; Zhang, R. 3D trajectory optimization for energy-efficient UAV communication: A control design perspective. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 4579–4593. [\[CrossRef\]](#)
19. Lu, Z.; Cheng, R.; Jin, Y.; Tan, K.C.; Deb, K. Neural architecture search as multiobjective optimization benchmarks: Problem formulation and performance assessment. *arXiv* **2023**, arXiv:2208.04321.
20. Zhang, Y.; Luo, J.; Zhang, Y.; Huang, Y.; Cai, X.; Yang, J.; Mao, D.; Li, J.; Tuo, X.; Zhang, Y. Resolution enhancement for large-scale real beam mapping based on adaptive low-rank approximation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5116921. [\[CrossRef\]](#)
21. Jin, B.; Cruz, L.; Gonçalves, N. Deep facial diagnosis: Deep transfer learning from face recognition to facial diagnosis. *IEEE Access* **2020**, *8*, 123649–123661. [\[CrossRef\]](#)
22. Zheng, Q.; Zhao, P.; Li, Y.; Wang, H.; Yang, Y. Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification. *Neural Comput. Appl.* **2021**, *33*, 7723–7745. [\[CrossRef\]](#)
23. Panareda Busto, P.; Gall, J. Open set domain adaptation. In Proceedings of the IEEE international Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 754–763.
24. Zhang, Y.; Guo, X.; Ma, J.; Liu, W.; Zhang, J. Beyond brightening low-light images. *Int. J. Comput. Vis.* **2021**, *129*, 1013–1037. [\[CrossRef\]](#)
25. Wang, R.; Zhang, Q.; Fu, C.W.; Shen, X.; Zheng, W.S.; Jia, J. Underexposed photo enhancement using deep illumination estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6849–6857.
26. Yang, W.; Wang, S.; Fang, Y.; Wang, Y.; Liu, J. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 3063–3072.
27. Jiang, Y.; Gong, X.; Liu, D.; Cheng, Y.; Fang, C.; Shen, X.; Yang, J.; Zhou, P.; Wang, Z. Enlightengan: Deep light enhancement without paired supervision. *IEEE Trans. Image Process.* **2021**, *30*, 2340–2349. [\[CrossRef\]](#)
28. Zhang, Y.; Di, X.; Zhang, B.; Wang, C. Self-supervised image enhancement network: Training with low light images only. *arXiv* **2020**, arXiv:2002.11300.
29. Guo, C.; Li, C.; Guo, J.; Loy, C.C.; Hou, J.; Kwong, S.; Cong, R. Zero-reference deep curve estimation for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1780–1789.
30. Liu, R.; Ma, L.; Zhang, J.; Fan, X.; Luo, Z. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 10561–10570.
31. Ma, L.; Ma, T.; Liu, R.; Fan, X.; Luo, Z. Toward fast, flexible, and robust low-light image enhancement. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5637–5646.
32. Hare, S.; Golodetz, S.; Saffari, A.; Vineet, V.; Cheng, M.M.; Hicks, S.L.; Torr, P.H. Struck: Structured output tracking with kernels. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 2096–2109. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 1409–1422. [\[CrossRef\]](#) [\[PubMed\]](#)
34. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista, J. High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 583–596. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Tao, R.; Gavves, E.; Smeulders, A.W. Siamese instance search for tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1420–1429.
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, 8–10 and 15–16 October 2016; Proceedings, Part II 14; pp. 850–865.
37. Kiani Galoogahi, H.; Fagg, A.; Lucey, S. Learning background-aware correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1135–1143.
38. Yang, T.; Chan, A.B. Learning dynamic memory networks for object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 152–167.

39. Wang, Q.; Teng, Z.; Xing, J.; Gao, J.; Hu, W.; Maybank, S. Learning attentions: Residual attentional siamese network for high performance online visual tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4854–4863.
40. He, A.; Luo, C.; Tian, X.; Zeng, W. A twofold siamese network for real-time object tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4834–4843.
41. 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, 18–23 June 2018; pp. 8971–8980.
42. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [[CrossRef](#)]
43. 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, 15–20 June 2019; pp. 4282–4291.
44. Wang, Q.; Zhang, L.; Bertinetto, L.; Hu, W.; Torr, P.H. Fast online object tracking and segmentation: A unifying approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1328–1338.
45. Xu, Y.; Wang, Z.; Li, Z.; Yuan, Y.; Yu, G. Siamfc++: Towards robust and accurate visual tracking with target estimation guidelines. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12549–12556.
46. 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.
47. 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, 13–19 June 2020; pp. 6668–6677.
48. 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, 13–19 June 2020; pp. 6269–6277.
49. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. [[CrossRef](#)]
50. Li, B.; Fu, C.; Ding, F.; Ye, J.; Lin, F. ADTrack: Target-aware dual filter learning for real-time anti-dark UAV tracking. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 496–502.
51. Ye, J.; Fu, C.; Zheng, G.; Paudel, D.P.; Chen, G. Unsupervised domain adaptation for nighttime aerial tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 8896–8905.
52. Li, B.; Fu, C.; Ding, F.; Ye, J.; Lin, F. All-day object tracking for unmanned aerial vehicle. *IEEE Trans. Mob. Comput.* **2022**, *22*, 4515–4529. [[CrossRef](#)]
53. Chen, T.; Ma, K.K.; Chen, L.H. Tri-state median filter for image denoising. *IEEE Trans. Image Process.* **1999**, *8*, 1834–1838. [[CrossRef](#)]
54. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 60–65.
55. Gu, S.; Zhang, L.; Zuo, W.; Feng, X. Weighted nuclear norm minimization with application to image denoising. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2862–2869.
56. Pang, J.; Cheung, G. Graph Laplacian regularization for image denoising: Analysis in the continuous domain. *IEEE Trans. Image Process.* **2017**, *26*, 1770–1785. [[CrossRef](#)]
57. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [[CrossRef](#)]
58. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622. [[CrossRef](#)] [[PubMed](#)]
59. Guo, S.; Yan, Z.; Zhang, K.; Zuo, W.; Zhang, L. Toward convolutional blind denoising of real photographs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1712–1722.
60. Wang, M.; Liu, Y.; Huang, Z. Large margin object tracking with circulant feature maps. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4021–4029.
61. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
62. Hai, J.; Xuan, Z.; Yang, R.; Hao, Y.; Zou, F.; Lin, F.; Han, S. R2rnet: Low-light image enhancement via real-low to real-normal network. *J. Vis. Commun. Image Represent.* **2023**, *90*, 103712. [[CrossRef](#)]
63. Fu, C.; Cao, Z.; Li, Y.; Ye, J.; Feng, C. Siamese anchor proposal network for high-speed aerial tracking. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 510–516.

64. Cao, Z.; Fu, C.; Ye, J.; Li, B.; Li, Y. SiamAPN++: Siamese attentional aggregation network for real-time UAV tracking. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 3086–3092.
65. Cao, Z.; Fu, C.; Ye, J.; Li, B.; Li, Y. Hift: Hierarchical feature transformer for aerial tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 15457–15466.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.