remote sensing

MDPI

Article

Moving Point Target Detection Based on Temporal Transient Disturbance Learning in Low SNR

Weihua Gao [1,2], Wenlong Niu [1,*], Pengcheng Wang [1,2], Yanzhao Li [1,2], Chunxu Ren [1], Xiaodong Peng [1] and Zhen Yang [1]

[1] National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China
[2] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: niuwenlong@nssc.ac.cn

Abstract: Moving target detection in optical remote sensing is important for satellite surveillance and space target monitoring. Here, a new moving point target detection framework under a low signal-to-noise ratio (SNR) that uses an end-to-end network (1D-ResNet) to learn the distribution features of transient disturbances in the temporal profile (TP) formed by a target passing through a pixel is proposed. First, we converted the detection of the point target in the image into the detection of transient disturbance in the TP and established mathematical models of different TP types. Then, according to the established mathematical models of TP, we generated the simulation TP dataset to train the 1D-ResNet. In 1D-ResNet, the structure of CBR-1D (Conv1D, BatchNormalization, ReLU) was designed to extract the features of transient disturbance. As the transient disturbance is very weak, we used several skip connections to prevent the loss of features in the deep layers. After the backbone, two LBR (Linear, BatchNormalization, ReLU) modules were used for further feature extraction to classify TP and identify the locations of transient disturbances. A multitask weighted loss function to ensure training convergence was proposed. Sufficient experiments showed that this method effectively detects moving point targets with a low SNR and has the highest detection rate and the lowest false alarm rate compared to other benchmark methods. Our method also has the best detection efficiency.

Keywords: moving point target; low SNR; transient disturbance; temporal profile; skip connection

Citation: Gao, W.; Niu, W.; Wang, P.; Li, Y.; Ren, C.; Peng, X.; Yang, Z. Moving Point Target Detection Based on Temporal Transient Disturbance Learning in Low SNR. *Remote Sens.* **2023**, *15*, 2523. https://doi.org/10.3390/rs15102523

Academic Editor: Gwanggil Jeon

Received: 25 February 2023
Revised: 24 April 2023
Accepted: 7 May 2023
Published: 11 May 2023

1. Introduction

The detection of moving targets has important applications in security monitoring, military reconnaissance, and satellite detection [1–3]. In some scenarios, such as early warning against space debris [4] and small faint bodies in near-Earth space or against naval ships and fighters, optical remote sensing detection has the characteristics of long distance and large field of view [5]. In this condition, the fast-moving target is more like a point in the image. The point target does not have shape, size, texture, or other spatial information and may even be submerged in background and clutter, resulting in a very low space-time signal-to-noise ratio (SNR) of the target and making it difficult to detect. Therefore, the problem of moving target detection in optical remote sensing images at a long distance and under a large field of view can be transformed into the problem of moving point target detection under a low SNR, which is important for effective detection.

There are currently three detection methods based on the temporal and spatial features of moving point targets: spatial-based detection, temporal-based detection, and spatiotemporal-based detection.

1.1. Spatial-Based Detection Methods

Spatial-based detection mainly realizes detection by enhancing small targets and suppressing the background or by converting the detection problem into an optimization

problem of separating sparse and low-rank matrices. For example, the top-hat algorithm first calculates an image to estimate the background and then subtracts the background from the original image to obtain small targets [6]. The max-mean filter and max-median filter suppress clutter by filtering in four directions and then subtracting the background to obtain candidate targets [7]. Local contrast measure (LCM) and its improved algorithms, such as MPCM, HWLCM, MLCM-LEF, and WVCLCM, use local contrast information to enhance the point target and suppress the background [8–12]. In contrast to the above-mentioned methods, IPI-based methods use the background non-local self-correlation property to transform the small target detection problem into an optimization problem of the recovery of low-rank and sparse matrices and use principal component pursuit to solve the problem [13–16]. Xia et al. considered both the global sparsity and local contrast of small targets and proposed a modified graph Laplacian model (MGLM) with local contrast and consistency constraints [17]. Because a point target with a low SNR lacks effective spatial information, the above methods cannot separate the target from the background.

In recent years, with the development of deep learning, point target detection algorithms based on convolutional neural networks have emerged endlessly, including ALCNet, GLFM, ISTDU, ISTNet, MLCL, and APANet [18–23]. The principles of these CNN-based methods are predominantly similar to those of traditional methods. Multilayer neural networks are used to enhance the point targets, suppress the background, and box the target position. Although the CNN-based method has improved the feature extraction ability of the target, it still cannot achieve excellent detection for low-SNR point targets lacking spatial information. In addition, the track of the target cannot be obtained by detecting a single image. In early warning systems, it is still necessary to detect image sequences. Because CNN-based detection methods take a long time to detect image sequences, they are inefficient.

1.2. Temporal-Based Detection Methods

Temporal-based detection refers to the detection of image sequences using the target's movement information in temporal terms, such as optical flow [24], temporal difference [25], dynamic background modeling (DBM) [26,27], and tracking before detection (TBD) [28]. Optical flow uses the correlation between adjacent frames in the image sequence and the changes of pixels over time to find the corresponding relationship between moving targets in the frames in order to calculate the motion information of moving targets. This method assumes that the brightness of the target is constant, that the motion between adjacent frames is derivable, and that the motion of adjacent pixels is similar. There are numerous constraints and few scenes that satisfy this assumption. In addition, the optical flow method is time-consuming and struggles to meet real-time requirements. The temporal difference method makes use of the gradual change of the background in the image sequence to directly identify differences in the adjacent frames. If there are moving targets in the sequence, this will lead to a large difference in the intensity of the adjacent frames. However, the temporal difference is sensitive to background noise and has a poor detection effect for point targets with a low SNR. The DBM models the background in the image sequence and determines whether the pixel belongs to the foreground or background according to the established model to segment the moving target. The detection performance of this method depends on the modeling accuracy. It is difficult to distinguish the moving point target from the background under a low SNR, and the target is easily misjudged as background. Thus, this method's robustness is poor. TBD is a commonly used algorithm for detecting the traces of small moving targets. This algorithm accumulates multiple frames, searches for every possible trace of targets, and finally decides on the searched trace. Therefore, it does not need to detect every single image, but it directly outputs the target's motion trace. However, this method requires excessive time to search. Moreover, if the target is weak, the target cannot be found effectively.

1.3. Spatiotemporal-Based Detection Methods

Researchers have proposed spatiotemporal-based detection methods that combine spatial and temporal information. For example, Zhang et al. proposed a three-dimensional filtering detection method, which takes a segment of an image sequence as the input and uses multiple matching filters to suppress the background in order to ultimately obtain point targets [29,30]. Deng et al. proposed a filtering method based on spatiotemporal local contrast, which calculates spatial and temporal local contrast, respectively, and then performs filtering mapping on spatiotemporal local contrast to obtain detection results [31]. Lin et al. used the Pearson correlation coefficient to suppress the background in the time-domain window and then used the target detection algorithm based on the regional gray level to suppress the residual background and finally obtained a target motion track [32]. Zhu et al. filtered the frame first, then detected the frame's gradient to obtain the candidate targets, and finally supplemented local contrast information in temporal terms for spatiotemporal joint judgment. Yan et al. used the top-hat algorithm to separate small targets from the background, a grid-based density peak search algorithm and gray area growth algorithm to identify false alarm points, and an improved KCF algorithm to achieve target tracking for continuous frames [33]. These algorithms use the spatiotemporal information of point targets to improve the detection effect, but their assumptions on small targets are too strong and require considerable prior information.

1.4. TP-Based Detection Methods

These temporal-based or spatiotemporal-based methods only use a few frames and do not fully use the temporal information of the target, and so they do not exhibit good detection performance. Under the observation condition of staring imaging, the intensity change of a single pixel in the image sequence over time can be regarded as a profile. If a target passes a pixel, it will produce a transient disturbance in the temporal profile (TP) of that pixel. If the transient disturbance can be detected, the target will be detected. Thus, the point target detection in an image can be converted into the detection of transient disturbances in the TP. Methods based on TP have been proposed. Liu et al. estimated the background signal from the original TP and then subtracted it to obtain the target signal [34,35]. Subsequently, Liu et al. performed the nonlinear adaptive filtering of TP to extract the target signal [36]. Recently, Liu et al. used FFT and KL to calculate the similarity between the TP and waveform to detect the target signal [37]. Niu et al. proposed detection methods based on statistical distribution distance involving high-frame-rate detection [38–40]. These methods are effective for TPs with a high SNR, but for TPs with a low SNR, the target signal cannot be separated from the background signal, and the time when the target appears in the TP cannot be identified.

The transient disturbance of the target formed by the pixels can be regarded as a pattern that can be recognized by CNN-1D. Therefore, to overcome the problems of the previous methods and achieve effective moving point target detection under a low SNR, we proposed a detection framework based on transient disturbance distribution feature learning. The framework takes the image sequence as the input and directly outputs the track of the point target.

The main contributions of our work are as follows:

1. We converted the point target detection in the image into the detection of transient disturbance in the TP formed by a pixel and propose a low-SNR point target detection framework based on transient disturbance distribution feature learning.
2. In the detection framework, we designed a 1D-ResNet for transient disturbance feature learning. The 1D-ResNet can learn the distribution features of the transient disturbance and realize the classification of the TP and the location of the transient disturbance. In 1D-ResNet, skip connections are used to prevent the loss of the target signal feature. To prevent gradient disappearance and gradient explosion, the structure of the CBR-1D was designed to extract the features of the weak transient disturbance. The specially designed weighted multitask loss function ensures

training convergence. In addition, we verified the effect of network depth on the detection performance of 1D-ResNet and trained two networks: 1D-ResNet-8 and 1D-ResNet-16. The two networks deal with detection speed priority and detection rate priority, respectively.

3. We formulated the TP formed by pixels and generated a simulation dataset according to the TP formula. By combining the simulation data and real-world data, a training and verification dataset satisfying the research of moving point target detection with a low SNR is generated. Compared to other spatial-based and temporal-based methods, the proposed method exhibits the best performance in terms of its detection rate, false alarm rate, and computing efficiency. The biggest advantage of our method is that it exhibits excellent detection performance under extremely low SNRs.

The remainder of this paper is organized as follows. Section 2 analyzes the components of the TP and establishes mathematical models for each part. The mathematical expressions for the target TP, background TP, and clutter TP are presented in Section 2. Section 3 details the moving point target detection framework, including the network architectures, model training, and the entire detection process. Section 4 presents the experimental scheme and results. We designed experiments based on four aspects and compared our method with other benchmark methods on test sequences. Section 5 discusses our method in detail and compares it with other methods, followed by network ablation experiments and visualization studies. Section 6 presents the conclusions of this study.

2. Temporal Profile Analysis

2.1. The Components of the Temporal Profile

Under the condition of staring imaging, each pixel in the image will form a *TP*, which tracks the change in pixel intensity value over time. Each *TP* is different. What is most important is the transient disturbance formed by the target passing through the pixel. Therefore, all *TP*s can be divided into two categories: background *TP* and target *TP* [34]. The *TP* of any pixel under ideal clutter-free conditions can be described as follows:

$$\overline{TP_{i,j}(k)} = \begin{cases} t_{i,j}(k), \ k_1 < k < k_2 \\ b_{i,j}(k), \ others \end{cases} \tag{1}$$

where $t_{i,j}$ and $b_{i,j}$ represent the distribution of the target *TP* and background *TP*, respectively; i and j represent the row and column index of the pixel in the image, respectively; k represents time; and k_1 and k_2 are the times when the target enters and leaves the pixel, respectively. The *TP* formation process is illustrated in Figure 1.

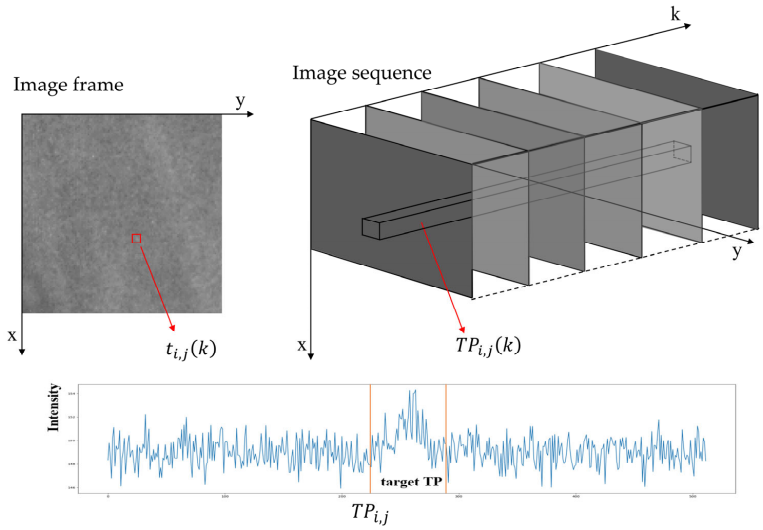

Figure 1. The formation process of *TP* in the image sequence; x and y are the horizontal and vertical coordinates of the image, respectively, and k is the frame number.

Under ideal clutter-free conditions, because the view of the detector is fixed, the background pixel intensity is constant for a short time, and the background *TP* can be considered a short-time stationary signal. However, in real image processing, the imaging results are affected by noise from different sources, including shot noise, thermal noise, photon noise, etc. In [35], additive white Gaussian noise (AWGN) was used to model these different noises. Thus, the actual *TP* can be expressed as follows:

$$TP_{i,j}(k) = AN + \overline{TP_{i,j}(k)} \tag{2}$$

where *AN* represents the AWGN.

2.2. The Target Temporal Profile

The TP of a target passing through a pixel can be regarded as a transient disturbance, and the following formula is used to describe the target *TP*:

$$t(k) = \begin{cases} s(k), & k_1 < k < k_2 \\ 0, & others \end{cases} \tag{3}$$

where $s(k)$ represents the transient disturbance caused by the appearance of the target.

The ideal imaging model of the optical system is pinhole imaging, and the light diffracts when mapping the object through the pinhole, forming a series of light–dark alternating diffraction rings. Therefore, a point in the real world will be a circle with a certain radius after imaging. Academia describes this phenomenon with a point spread function, and Pentland uses a two-dimensional Gaussian distribution to model it [41], which is defined as follows:

$$g(x, y) = Ae^{-a[(x-x_0)^2+(y-y_0)^2]} \tag{4}$$

where A represents the intensity of the target in the imaging, a represents the optical parameters of the detector, and (x_0, y_0) represents the center position of the target.

When the point target passes through a pixel, the intensity of the pixel first increases and then decreases, and a bell-shaped transient disturbance then appears on the *TP* of the pixel. The bell-shaped *TP* can be described by the following formula:

$$s(k) = Ae^{-a[v(k-k_0)]^2}, \quad k_1 < k < k_2 \tag{5}$$

where v is the moving speed of the target and $k_0 = k_1 + (k_2 - k_1)/2$ represents the time when the target center passes through the pixel.

The formation process and specific shape of target *TP* are shown in Figure 2.

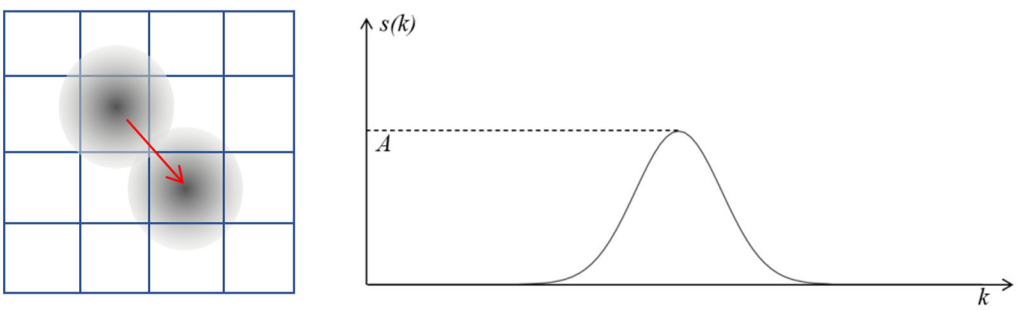

Figure 2. The formation process and the specific shape of target *TP*; k is the frame number and $s(k)$ is the intensity value of the transient disturbance; A is the maximum intensity of the point target.

Because the size of the target is smaller than the imaging spatial resolution, the target cannot completely cover the background, and the intensity of the target in imaging will be

affected by the background. Therefore, the formula of *TP* can be expressed as background distribution plus target distribution, as shown below:

$$TP_{i,j}(k) = n_{i,j}(k) + t_{i,j}(k) + b_{i,j}(k) \quad (6)$$

where $n_{i,j}(k)$ is the distribution of *AN*.

3. Detection Method

3.1. The Framework of Temporal Transient Disturbance Learning

We used CNN-1D to detect transient disturbances formed by the target. Because the transient disturbance is extremely weak, the feature extraction of the transient disturbance is difficult and the extracted features are easily lost in the network. The skip connection can directly transfer the shallow feature to deeper layers so that the network can fully learn the distribution feature of the transient disturbance and achieve high-accuracy detection. The detection framework of our method is shown in Figure 3, which includes two modules: training and detection. In the training part, we first generated the simulated *TP* by adding a bell-shaped signal to the background signal. Noise was then added to the *TP* to simulate a real situation. Next, a training dataset containing 160,000 *TP*s was generated under the experimental parameters. Subsequently, the proposed networks, 1D-ResNet-8 and 1D-ResNet-16, were trained under the same super-parameter settings. In the detection part, the trained model was used to detect the transient disturbance in *TP*s formed by pixels to detect the moving track of the point target.

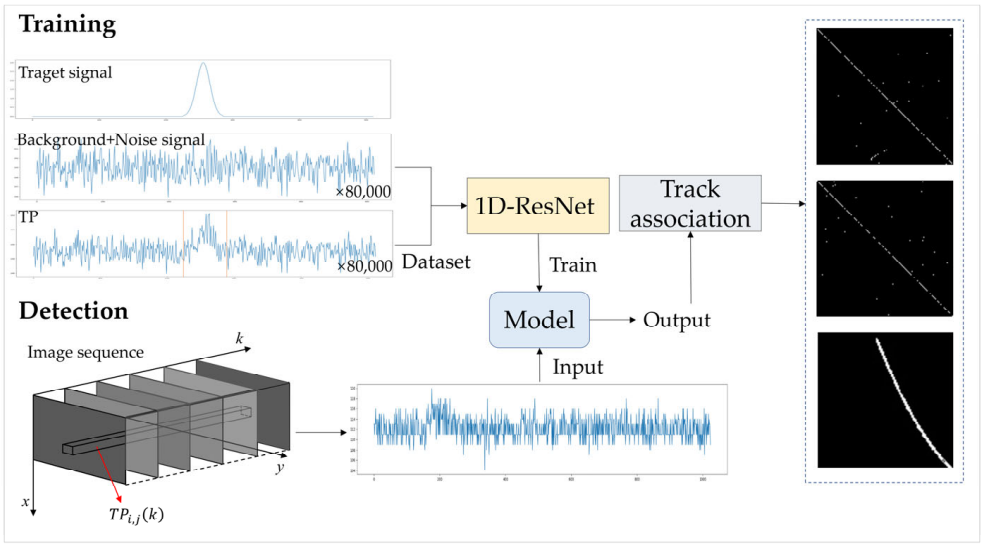

Figure 3. The detection framework of our method.

3.2. Architectures of 1D-ResNet

There are two tasks for detecting transient disturbances in a *TP*. One involves classifying the target *TP* containing a bell-shaped signal and the background *TP*. The other involves obtaining information on transient disturbances, such as the time of occurrence and the duration of the bell-shaped signal. Therefore, for these two detection tasks, inspired by classical ResNet and Darknet, we use one-dimensional ResNet as the backbone feature extraction network and CBR-1D as the basic feature extraction unit [42,43] to propose the 1D-ResNet. To verify the impact of the network layers on the detection performance, 1D-ResNet-8 and 1D-ResNet-16 were designed. The architectures of these 1D-ResNet are shown in Figure 4.

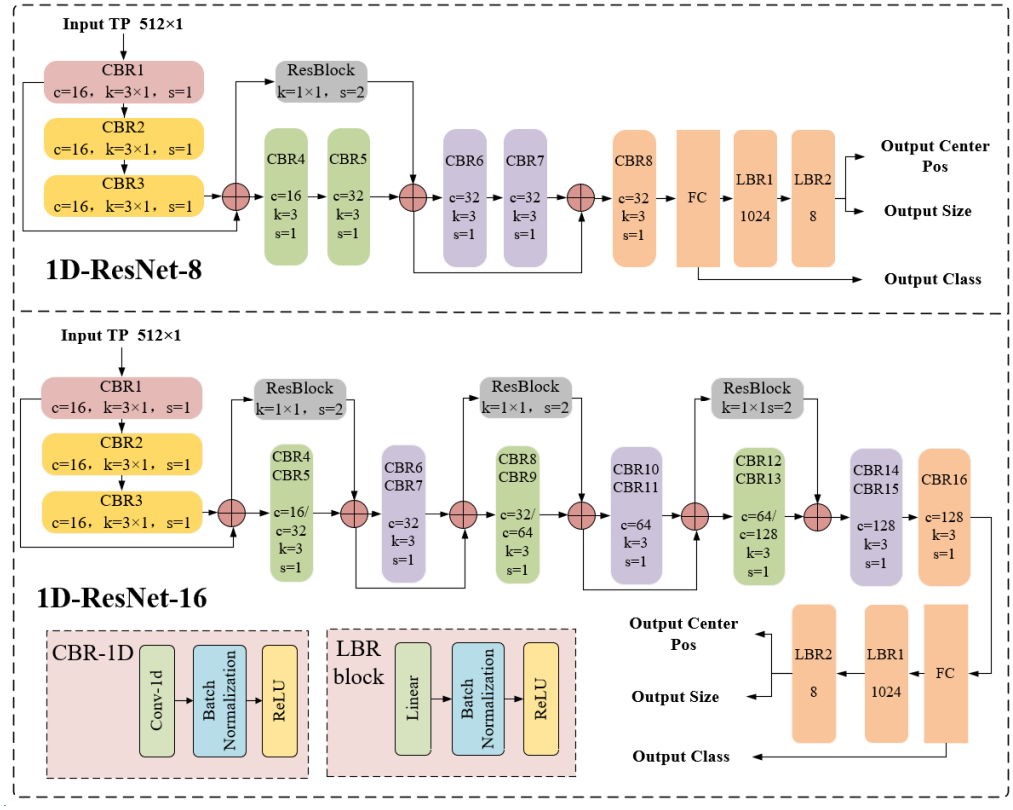

Figure 4. The architectures of 1D-ResNet.

Both networks are composed of input, backbone, neck, and output. A *TP* with a size of 512 × 1 was the input for the network. Backbone was used to extract the features of the *TP*. The neck connects the backbone and the output and to provide higher-dimensional features for the output. Finally, three outputs are obtained. If a bell-shaped signal exists, the outputs are the class of *TP*, the center position, and the size of the bell-shaped signal. Otherwise, we obtain three zero outputs.

In the training network stage, the *TP* class is easy to identify, as it is the first output. Meanwhile, identifying the center position and size is difficult. Therefore, two LBR blocks are set behind the convolution layer as the neck to further extract the features. Each LBR block includes a linear layer, batch normalization (BN), and ReLU.

Several skip connections were used to transmit the feature from the shallow layer to the deeper layer in order to avoid the loss of the transient disturbance feature. The CBR-1D includes a one-dimensional convolution layer (Conv1D), BN, and ReLU. Conv1D was used to extract local features in the *TP* and then normalize the extracted features. Finally, ReLU was used to activate the features. This can inhibit the change in the data distribution, accelerate the convergence speed, and avoid the problems of gradient disappearance and gradient explosion.

3.3. Training the 1D-ResNet

3.3.1. Generate the Dataset

Point targets with a SNR below 3 dB will have no obvious spatial features; therefore, the SNR research range was established as −3 dB to 3 dB. Because the actual *TP* under a specific SNR is difficult to obtain and label, the features of the target and ground *TP* were combined to generate a dataset through simulation. To enable the network to fully learn the features of *TP* within the research SNR range, *TP*s were generated between −4 dB and 4 dB.

During *TP* simulation, a bell-shaped target signal is generated according to Formula (5) and the location where the target signal appears is set randomly. To verify the effect of

the target signal size on the detection performance, the target signal size range was set to 10~110 and signals of different sizes were generated in equal proportions. A constant was randomly set as the background signal. The two signals were superimposed to obtain the simulated *TP*. Finally, AWGN was added to the *TP* simulation. To ensure that the model exhibits good performance on *TP*s with different SNRs and different target signal sizes, we set the number of TPs to be equal for each SNR and size range.

After generating the dataset, we divided it into training and validation sets at a ratio of 8:2, respectively. The composition of the *TP*s in the dataset is shown in Figure 5. The left figure shows the distribution of *TP*s under different SNRs and the right figure shows the distribution of *TP*s with different sizes under the same SNR.

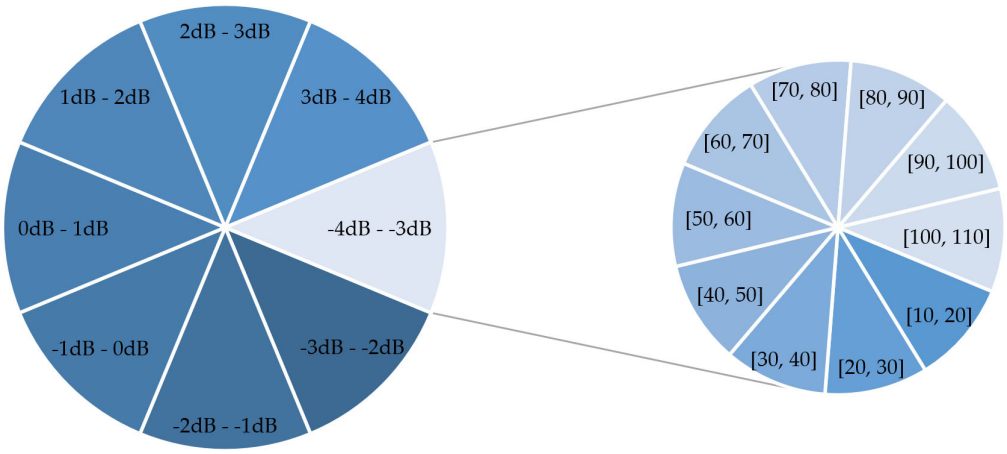

Figure 5. The composition of *TP*s in the dataset.

3.3.2. Loss Function

As the network trained in this study is a multitask learning network, the loss function is composed of three parts: classification loss, center position loss, and size loss. The classification loss uses binary cross-entropy loss, and the center position loss and size loss use the mean square error loss. Because there are significant differences in the order of magnitude of these three parts of the loss function, it is necessary to manually set their weights to prevent imbalance loss, and the final weighted loss function is shown in Formula (7).

$$Loss = w_1 * Loss_C + w_2 * Loss_P + w_3 * Loss_s \tag{7}$$

where $Loss_C$ represents the classification loss, $Loss_P$ represents the center position loss, and $Loss_s$ represents size loss. The formulas for these three parts are as follows:

$$\&Loss_C = -\frac{1}{N}\sum_{i=1}^{N}\left[C_i \log\left(\widehat{C}_i\right) + (1 - C_i)\log\left(1 - \widehat{C}_i\right)\right] \tag{8}$$

$$Loss_P = -\frac{1}{2N}\sum_{i=1}^{N}\left(P_i - \widehat{P}_i\right)^2 \tag{9}$$

$$Loss_W = -\frac{1}{2N}\sum_{i=1}^{N}\left(W_i - \widehat{W}_i\right)^2 \tag{10}$$

where C_i represents the category label, $\widehat{C}_i$ represents the predicted category, P_i represents the center point position label, $\widehat{P}_i$ represents the predicted center position, W_i represents the size label, and $\widehat{W}_i$ represents the predicted size label. N represents the number of TPs in a batch.

3.3.3. Training the Networks

PyTorch was used to build the network architectures and the training environment. The training equipment used was a workstation with an NVIDIA GeForce GTX 1080 ti GPU and 32 GB of memory.

During training, the random seed was set to 3407, the Adam optimizer was used, the parameter penalty coefficient was set to 1×10^{-5}, the learning rate was initially set to 1×10^{-4}, and the batch size was set to 2000. During training, rough training was first conducted for 10 epochs, then the learning rate was reduced 10-fold and fine-tuning was performed. If the loss of the validation set did not decrease within 10 epochs, the training ended. The loss optimization of network training is shown in Figure 6.

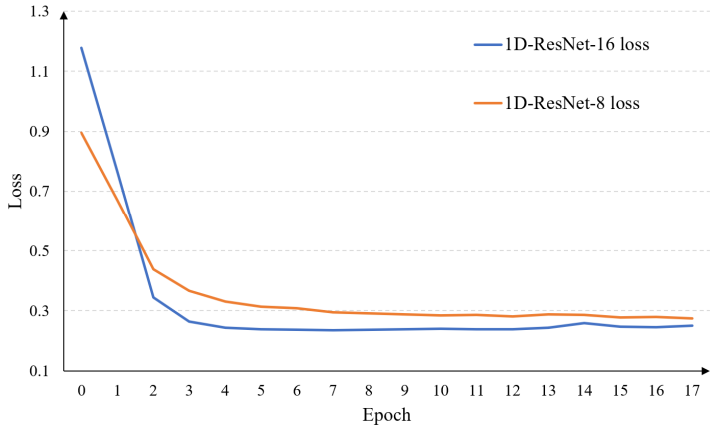

Figure 6. The loss optimization of network training. Epoch represents the number of iterations in the two networks' training.

Figure 6 shows that the two networks converged after five epochs of training, and the training effect of 1D-ResNet-16 was slightly better than that of 1D-ResNet-8.

3.4. The Moving Point Target Trajectory Detection Process

The detection process of our proposed framework is as follows:

1. Input an image sequence and obtain its *TP* for each pixel.
2. Pre-process the *TP*s, standardize the *TP*s, and divide the *TP* segments according to the network input size.
3. Load the trained model, input the *TP*s into 1D-ResNet in batches, and obtain the outputs.
4. Determine whether the *TP*s exist in the transient disturbance caused by the target according to the specified threshold value. If a *TP* exists, its pixel is considered to be in the foreground; otherwise, it is considered to be in the background.
5. Unify all foreground pixels and output the motion track of the target.

4. Experiments and Analysis

To evaluate the feasibility and performance of the proposed method, extensive experiments were conducted, including a *TP* simulation experiment, image-sequence simulation experiment, real-world experiment, and comparison experiment.

- The *TP* simulation experiment directly detects the simulated *TP* and evaluates the classification and positioning performance of the method under ideal conditions using the accuracy of the receiver operating characteristic (ROC) and intersection over union (IOU).
- To further fit the real scene and test the performance of the detection framework, we established image-sequence simulation experiments. A simulated moving point target was added to the real background image sequence. Simulation sequences were used as the input data of the detection framework.

- We shot the movement process of the point target outdoors and conducted a real-world experiment based on these data.
- To verify the performance of the proposed method, we compared it with that of other benchmark methods.

4.1. Details of Image Sequences in Experiments

In the experiments, we used seven image sequences, three of which were simulated. The other four were real-world data taken outdoors. The details of the image sequences used in the experiments are listed in Table 1.

Table 1. The details of image sequences.

Sequences	Resolution	Scenes	Speed (Pixels/Frame)	Frames	SNR (dB)
Sequence 1	128 × 128	Asphalt Road	0.0125	10,240	1.22
Sequence 2	128 × 128	Pure Sky	0.0125	10,240	0.6
Sequence 3	128 × 128	Complex Scene	0.0625	2048	1.09
Sequence 4	100 × 100	Asphalt Road	0.0122	8192	1–5
Sequence 5	100 × 100	Asphalt Road	0.0244	4096	1–5
Sequence 6	100 × 100	Asphalt Road	0.0488	2048	1–5
Sequence 7	100 × 100	Asphalt Road	0.0977	1024	1–5

In the image-sequence simulation experiment, we used asphalt roads, pure sky, and a complex scene to simulate space-based and ground-based detection. The backgrounds of sequence 1 and sequence 2 are simple, while the background of sequence 3 is more complex and has scenes such as sky, mountains, buildings, etc., in the background. In sequences 1–3, we added a point moving target that was 1–3 pixels in size to these background image sequences. This point target moves from the upper-left corner to the lower-right corner of the image sequence.

To verify the performance of the proposed method in a real image sequence, we used a high-speed camera to capture outdoor image sequences. We tracked the movement of a glass ball from a height of approximately 50 m at 20,000 fps. The diameter of the glass ball was 1.5 cm, and the SNR was approximately 1–5 dB. To facilitate the experimental analysis, we obtained 8192 frames from the original sequence and established a window of 100 × 100 pixels for the target to pass through.

After obtaining the original sequence 4, to verify the impact of the target's stay time on the detection effect on a single pixel, we down-sampled sequence 4 to obtain sequences 5–7.

4.2. TP Simulation Experiment

The experiments in this section were conducted in two ways to verify the detection effect of our method on *TP*s with different SNRs and target signals of different sizes. ROC and IOU accuracy rates were used to evaluate the classification and positioning capabilities of the method, respectively.

The ROC curve is a graphical representation of the performance of a binary classification model as the discrimination threshold is varied. The x-axis represents the false positive rate (FPR), which is the ratio of false positives (incorrectly classified negative samples) to the total number of negative samples. The y-axis represents the true positive rate (TPR), which is the ratio of true positives (correctly classified positive samples) to the total number of positive samples. In this paper, positive samples refer to *TP*s containing the target signal, while negative samples refer to TPs without the target signal. Each point on the ROC curve reflects the sensitivity of the classifier to different discrimination thresholds. The larger the area under the curve (AUC) covered under the ROC curve, the better the detection performance of the method.

The center position and size of the transient disturbance form the bounding box. If the IOU is greater than 0.5, the positioning is considered correct. The calculation method of the

IOU of the predicted and true bounding boxes is shown in Equation (11). The higher the accuracy of the IOU, the better the positioning performance of the method.

$$IOU = \frac{\min(E_T, E_P) - \max(S_T, S_P)}{\max(E_T, E_P) - \min(S_T, S_P)} \tag{11}$$

where E_T and E_P represent the right boundary of the true bounding box and predicted bounding box, respectively, and S_T and S_P represent the left boundary of the true bounding box and predicted bounding box, respectively.

4.2.1. The Detection Performance under Difference SNR

To verify the influence of SNR on detection performance, simulation *TP*s under different SNRs were generated. Under each SNR, the size of the target signal is set between 10 and 110 in equal proportion. The ROC curves drawn using the two networks under different SNRs are shown in Figure 7, and the AUC and accuracy of the IOU are shown in Table 2.

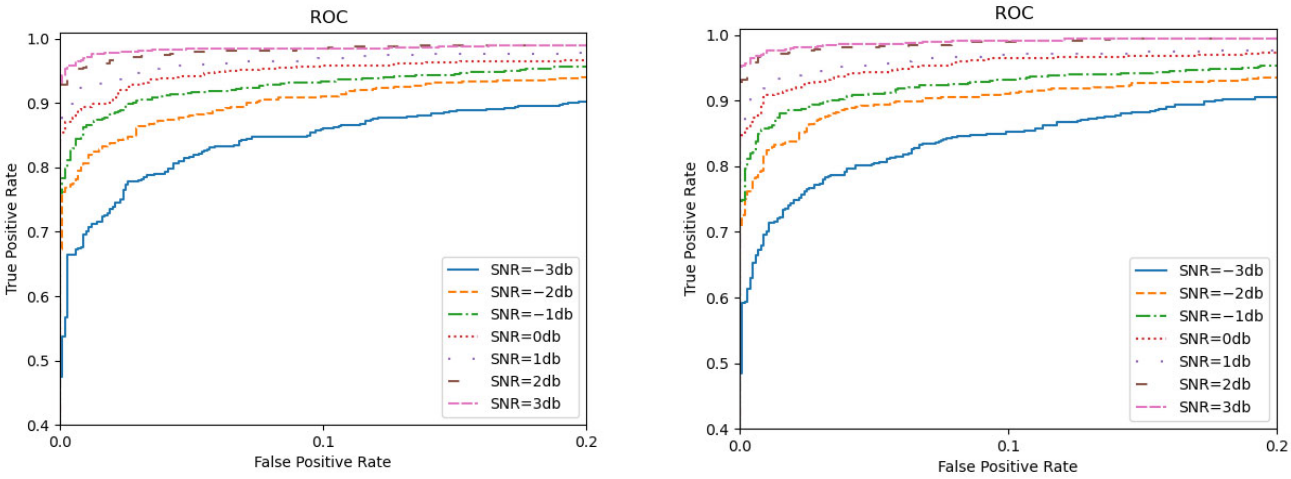

Figure 7. The detailed ROC of two networks under different SNRs. (**a**) ROC of 1D-ResNet-8. (**b**) ROC of 1D-ResNet-16.

Table 2. AUC and accuracy of IOU under different SNRs.

SNR	AUC		Accuracy of IOU	
	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16
−3 dB	0.9393	0.9389	0.5350	0.6460
−2 dB	0.9591	0.9593	0.6130	0.7000
−1 dB	0.9724	0.9719	0.6840	0.7320
0 dB	0.9807	0.9832	0.7350	0.7770
1 dB	0.9867	0.9859	0.7340	0.7670
2 dB	0.9941	0.9957	0.7730	0.7810
3 dB	0.9955	0.9963	0.7880	0.8020

As is shown in Figure 7 and Table 2, the classification performance of the two models reached a good level, and all ROCs covered over 90% of the area. With a decrease in the SNR, the classification performance worsens. The accuracy of the IOU also decreases with a decrease in the SNR. This is because transient disturbances under low SNR are very weak and can easily be submerged in the background. During the detection process, the target signal is prone to clutter interference, resulting in classification and positioning errors.

4.2.2. The Detection Performance under Different Target Signal Sizes

To verify the influence of target signal size on detection performance, in the experiment, simulated TPs with different sizes were generated, in which the SNR was set at an equal ratio of –3 dB to 3 dB under each size. The ROC drawn by the two networks under different target signal sizes are shown in Figure 8, and the AUC and accuracy of IOU are shown in Table 3.

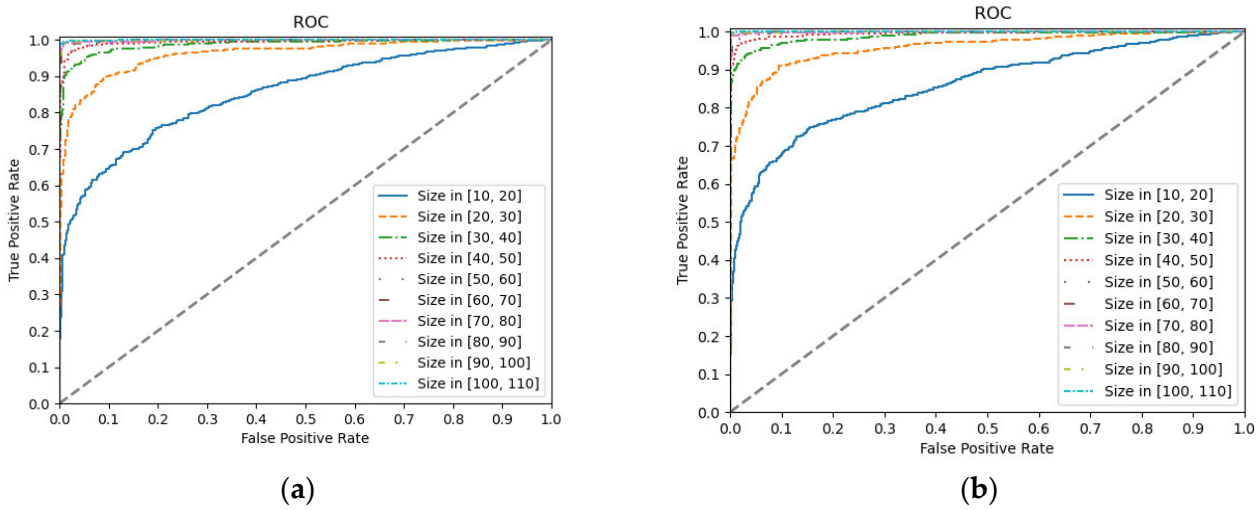

Figure 8. The ROC of two networks under different target signal sizes. (**a**) ROC of 1D-ResNet-8. (**b**) ROC of 1D-ResNet-16.

Table 3. AUC and accuracy of IOU under different target signal sizes.

Size	AUC		Accuracy of IOU	
	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16
[10, 20]	0.8541	0.8574	0.0614	0.1214
[20, 30]	0.9614	0.9575	0.2486	0.3143
[30, 40]	0.9873	0.9882	0.4371	0.4914
[40, 50]	0.9937	0.9951	0.6586	0.6643
[50, 60]	0.9966	0.9979	0.7543	0.7671
[60, 70]	0.9986	0.9993	0.8314	0.8771
[70, 80]	0.9988	0.9991	0.9129	0.9343
[80, 90]	0.9980	0.9993	0.9529	0.9529
[90, 100]	0.9991	0.9997	0.9614	0.9771
[100, 110]	0.9997	0.9999	0.9914	0.9871

As is shown in Figure 8 and Table 3, with an increase in the target signal size, the classification and positioning capabilities of the two models show a significant improvement trend. For classification tasks, when the target signal size was less than 20, the classification performance was very poor, whereas when the target signal size increased to 40, the AUC of both models reached over 99%.

For positioning tasks, the IOU accuracy exhibited a more obvious trend with an increase in the target signal size. When the size was increased to 70, the accuracy increased to over 90%.

From the experimental results, we can see that the size of the target signal is a crucial factor for our methods. The longer the moving target stays on a single pixel, the more sufficient are the motion features and the better the performance of the proposed method. Therefore, the detection performance can be improved by increasing the frame rate of the detector.

4.2.3. *TP* Simulation Experiment Analysis

The SNR and size of the target signal are important factors that affect detection performance. The higher the SNR and the larger the proportion of the target signal, the better the model detection performance. The proportion of the target signal has a greater impact on the detection effect than the SNR. The SNR cannot be significantly improved; however, the proportion of the target signal can be further improved by increasing the frame rate of the detection equipment.

Among the two networks, although 1D-ResNet-16 has an additional eight layers of CBR-1D and 228,160 parameters compared to 1D-ResNet-8, the improvement of the model's detection performance is very small, the classification performance gap is small, and the IOU accuracy rate is less than two percentage points higher than that of 1D-ResNet-8. This proves that for weak transients, deeper network layers do not lead to greater performance improvement; however, deeper networks lead to greater computing consumption, which is contradictory to real-time detection performance.

4.3. Image-Sequence Simulation Experiment

We used our method to detect sequences 1–3. The detection results for the two networks are presented in Figure 9 and Table 4.

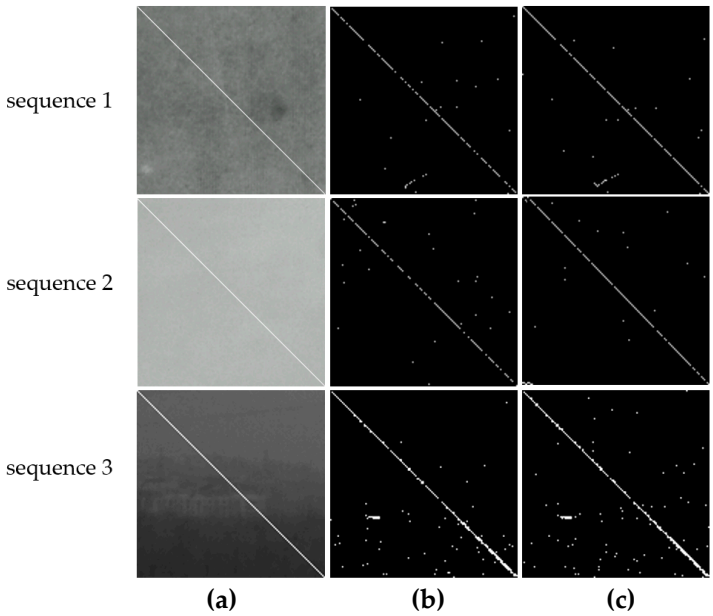

Figure 9. The detection results of the two networks in simulated image sequences. (**a**) The ground truth of the image sequences. (**b**) The detection results of 1D-ResNet-8. (**c**) The detection results of 1D-ResNet-16.

Table 4. The detection results of the simulated image sequences.

Sequences	Detection Rate		False Alarm Rate		Efficiency (ms/Frame)	
	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16
Sequence 1	72.67%	88.28%	0.15%	0.17%	1.77	2.97
Sequence 2	69.53%	87.50%	0.16%	0.11%	1.83	2.96
Sequence 3	78.12%	79.69%	0.58%	0.51%	3.01	4.19

From Figure 9 and Table 4, we can observe that both networks show good detection performance for all three sequences. Although there were some false alarm points, the moving track (main diagonal) of the target was clear. Additionally, these false alarm points can be removed through post-processing.

Compared to 1D-ResNet-8, the detection rate of 1D-ResNet-16 is higher, but the time consumption of 1D-ResNet-8 is lower. In an actual detection task, we should use 1D-ResNet-16 if the detection rate is more important. However, if the detection speed is more important, 1D-ResNet-8 should be used.

4.4. Real-World Experiment

We used our method to detect real-world sequences. The detection results are shown in Figure 10 and Table 5.

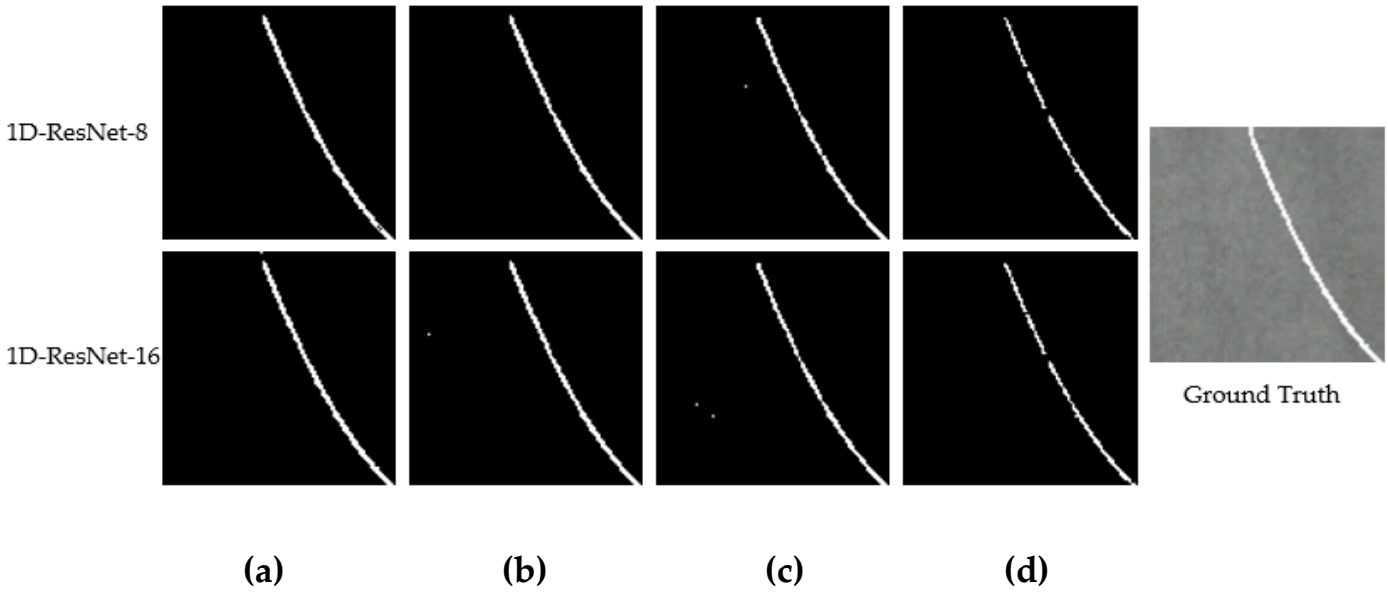

Figure 10. The detection results of the two networks on real-data sequences. (**a**) The detection results of the two networks on sequence 4. (**b**) The detection results of the two networks on sequence 5. (**c**) The detection results of the two networks on sequence 6. (**d**) The detection results of the two networks on sequence 7.

Table 5. The detection results on real data sequences.

Sequences	Detection Rate		False Alarm Rate		Efficiency (ms/Frame)	
	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16	1D-ResNet-8	1D-ResNet-16
Sequence 4	96.00%	97.00%	0.00%	0.00%	1.63	2.27
Sequence 5	96.00%	96.00%	0.00%	0.01%	1.79	2.58
Sequence 6	95.00%	95.00%	0.01%	0.02%	2.55	3.15
Sequence 7	90.00%	91.00%	0.00%	0.00%	3.62	4.49

The results show that both networks have relatively good detection performance on the sequences, both of which completely detect the moving track of the glass ball. With an increase in the de-sampling fold, the stay frames of the target in a single pixel become shorter and the detection performance worsens. In this experiment, 1D-ResNet-16 had no significant advantage over 1D-ResNet-8 in terms of detection performance. Therefore, 1D-ResNet-8 can meet the detection requirements when the SNR is high.

4.5. Contrast Experiments with the Benchmark Methods

To verify the performance of the proposed method, we compared it with some benchmark methods, including MaxMean [7], IPI [13], LCM [8], Kernel [38], ICLSP [35], NAF [36], and TRLCM [44]. MaxMean, IPI, and LCM are spatial-based methods, whereas Kernel, ICLSP, NAF, and TRLCM are temporal-based methods.

Sequences 1–4 were used for comparison. The results are presented in Figure 11 and Tables 6 and 7.

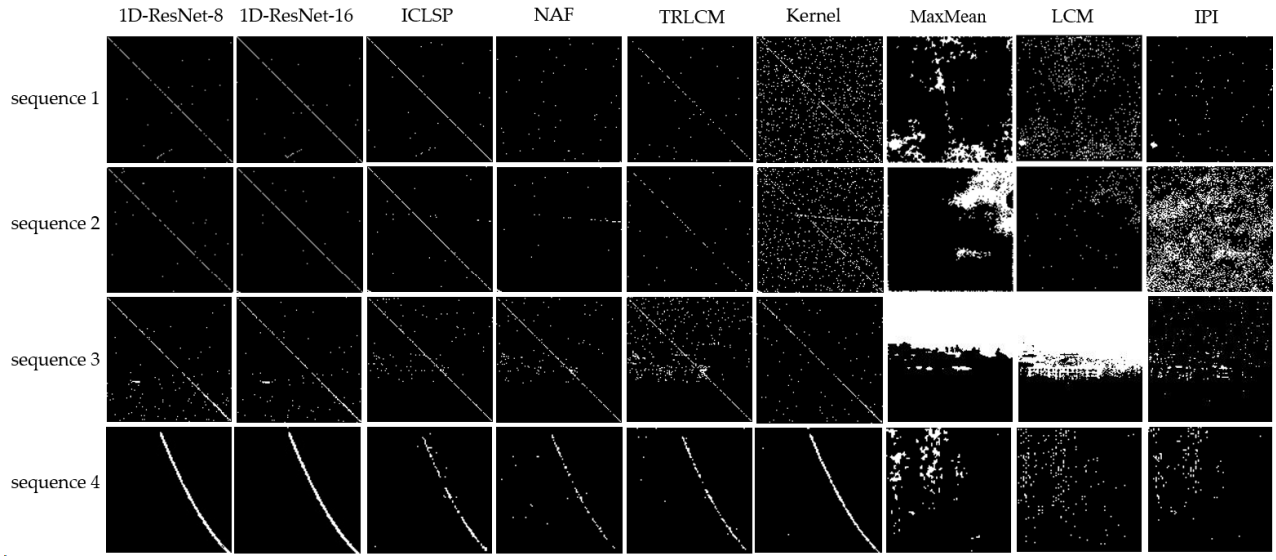

Figure 11. The detection results of the proposed methods and benchmark methods.

Table 6. The detection rates of the proposed methods and the benchmark methods.

Method	Detection Rate			
	Sequence 1	**Sequence 2**	**Sequence 3**	**Sequence 4**
1D-ResNet-16	**88.28%**	**87.50%**	**79.69%**	**97.00%**
1D-ResNet-8	72.67%	69.53%	78.13%	96.00%
ICLSP	88.28%	82.81%	78.13%	68.00%
NAF	1.56%	0.78%	73.43%	52.00%
TRLCM	35.16%	29.69%	71.88%	68.00%
Kernel	67.97%	60.16%	66.41%	95.00%
MaxMean	10.16%	1.56%	46.09%	8.00%
LCM	6.25%	0.78%	28.12%	9.00%
IPI	1.56%	23.43%	4.69%	3.00%

Table 7. The false alarm rates of the proposed methods and the benchmark methods.

Method	False Alarm Rate			
	Sequence 1	**Sequence 2**	**Sequence 3**	**Sequence 4**
1D-ResNet-16	0.15%	0.16%	**0.51%**	**0.00%**
1D-ResNet-8	0.17%	**0.11%**	0.58%	**0.00%**
ICLSP	**0.14%**	0.19%	0.54%	0.13%
NAF	0.29%	0.23%	0.61%	0.15%
TRLCM	0.20%	0.18%	2.22%	0.14%
Kernel	5.38%	4.71%	0.61%	0.02%
MaxMean	8.87%	13.81%	43.21%	4.80%
LCM	5.73%	1.29%	33.03%	2.13%
IPI	0.70%	26.00%	1.51%	1.54%

Figure 11 shows that the temporal-based methods can detect low-SNR point targets in an image sequence, whereas the spatial-based methods cannot detect the target track.

Among the temporal-based methods, our method has the best performance, followed by ICLSP. Although ICLSP exhibits similar performance to our method on simulation sequences, its detection effect is far inferior to that of our method on real-world low-SNR sequences. The Kernel method can better detect a real sequence with a high SNR, but there are many false alarm points for the simulation sequence with a low SNR. This shows that our method not only has excellent detection ability for moving point targets with a low SNR but also has good robustness for real point targets.

Table 8 shows the computational efficiency of all methods. These methods are implemented on a computer with an AMD Ryzen 7 1700 CPU and a Nvidia GeForce GTX 1080 ti GPU. From Table 8, it can be seen that our method has the fastest detection speed. The detection speed of 1D-ResNet-8 is faster than that of 1D-ResNet-16, as 1D-ResNet-8 has fewer parameters. In the future, we will improve the network by proposing lightweight networks to further improve the detection speed.

Table 8. The computing efficiency of all methods.

Method	Environment	Computing Efficiency (ms/Frame)			
		Sequence 1	Sequence 2	Sequence 3	Sequence 4
1D-ResNet-16	python3.9+cuda11.7	**2.97**	**2.96**	**4.19**	**2.27**
1D-ResNet-8	python3.9+cuda11.7	**1.77**	**1.83**	**3.01**	**1.63**
ICLSP	python3.9	30.84	32.35	29.82	18.58
NAF	python3.9	1194.15	1203.66	1060.80	761.13
TRLCM	python3.9	580.75	528.75	478.37	355.44
Kernel	python3.9	1586.91	1287.25	3589.67	1481.46
MaxMean	matlab2018	246.33	244.61	226.59	147.11
LCM	matlab2018	429.53	428.21	418.99	258.24
IPI	matlab2018	612.23	528.25	760.69	237.29

5. Discussion

In this section, we discuss our method in detail and compare it with other methods to illustrate its advantages and disadvantages. After that, we discuss the results of our ablation experiments to verify the effects of various parts of 1D-ResNet. Finally, we discuss the results of our visualization research on the network to verify whether it learned the features of transient disturbances.

5.1. Analysis of All Methods

In this section, we analyze the characteristics, advantages, and disadvantages of all methods, as shown in Table 9.

Table 9. The characteristics, advantages, and disadvantages of all methods.

Method	Characteristics	Advantages	Disadvantages
1D-ResNet-16	Batch detection of transient disturbances in *TP* using 1D-ResNet-16 on GPU	Best detection ability and fast detection speed for low-SNR point targets; Few hyperparameters	The detection speed is slower than that of 1D-ResNet-8
1D-ResNet-8	Batch detection of transient disturbances in *TP* using 1D-ResNet-8 on GPU	Good detection ability and fastest detection speed for low-SNR point targets; Few hyperparameters	The detection performance is slightly worse than that of 1D-ResNet-16
ICLSP	Calculate the deviation distribution between *TP* and CLSP on the CPU to detect the target *TP*	Good detection ability for low-SNR point targets	Poor detection performance for real data; Slow detection speed; More hyperparameters
NAF	Using a nonlinear filter to extract the target *TP* on CPU	Moving point target with a higher SNR can be detected	Very slow detection speed; Unable to detect low-SNR targets; More hyperparameters
TRLCM	Using temporal local contrast information to detect target *TP* on CPU	Can detect low-SNR point targets	Very slow detection speed; Poor detection performance for low-SNR targets; More hyperparameters

Table 9. *Cont.*

Method	Characteristics	Advantages	Disadvantages
Kernel	Calculate the statistical distribution distance between the target and background to detect target *TP* on CPU	Can detect low-SNR point targets; Few hyperparameters	Very slow detection speed; Poor detection performance for low-SNR targets
MaxMean	Detect each image based on local maximum mean on CPU	Simple detection theory; Few hyperparameters	Very slow detection speed; Completely unable to detect low-SNR targets
LCM	Detect each image based on local contrast on CPU	Simple detection theory; Few hyperparameters	Very slow detection speed; Completely unable to detect low-SNR targets
IPI	Based on the non-local autocorrelation characteristics of the background, transform the small target detection into an optimization problem of recovering low-rank and sparse matrices and use stable principal component pursuit to solve this problem on CPU	Simple detection theory; Few hyperparameters	Very slow detection speed; Completely unable to detect low-SNR targets

The two networks we propose have the best detection performance and fastest detection speed for low-SNR moving point targets. Of the two networks, 1D-ResNet-16 has the best detection performance, while 1D-ResNet-8 has the fastest detection speed.

Other *TP*-based detection methods (ICLSP, NAF, TRLCM, and Kernel) can also detect the motion trajectory of targets, but their detection rate and false alarm rate are not as good as those of our methods, and these methods require more time for detection.

Other spatial-based methods (MaxMean, LCM, and IPI) are completely unable to detect point targets under a low SNR.

5.2. Ablation Experiments

In this section, we conducted ablation experiments to verify the superiority of the 1D-ResNet and CBR-1D. Due to the similarity of the two network structures (1D-ResNet-16 only has eight more layers than 1D-ResNet-8), this section is based on 1D-ResNet-8 only.

5.2.1. Network Structure Study

We first removed the skip connections from the network and then replaced the basic structural unit CBR with Conv-1D (Conv1D and ReLU) and CBL (Conv1D, BN, and LeakyReLU). We then removed the LBR module from the network. The ablation experiment we designed is shown in Table 10.

Table 10. The networks of ablation experiments.

Network	Skip Connection	Basic Structural Unit	LBR
CNN-1D	✗	Conv-1D	✗
ResNet-1D	✓	Conv-1D	✗
ResNet-1D LBR	✓	Conv-1D	✓
ResNet-1D CBL+LBR	✓	CBL-1D	✓
Ours	✓	CBR-1D	✓

We did not weigh the loss function to see the performance of these networks. The loss optimization of all networks is shown in Figure 12.

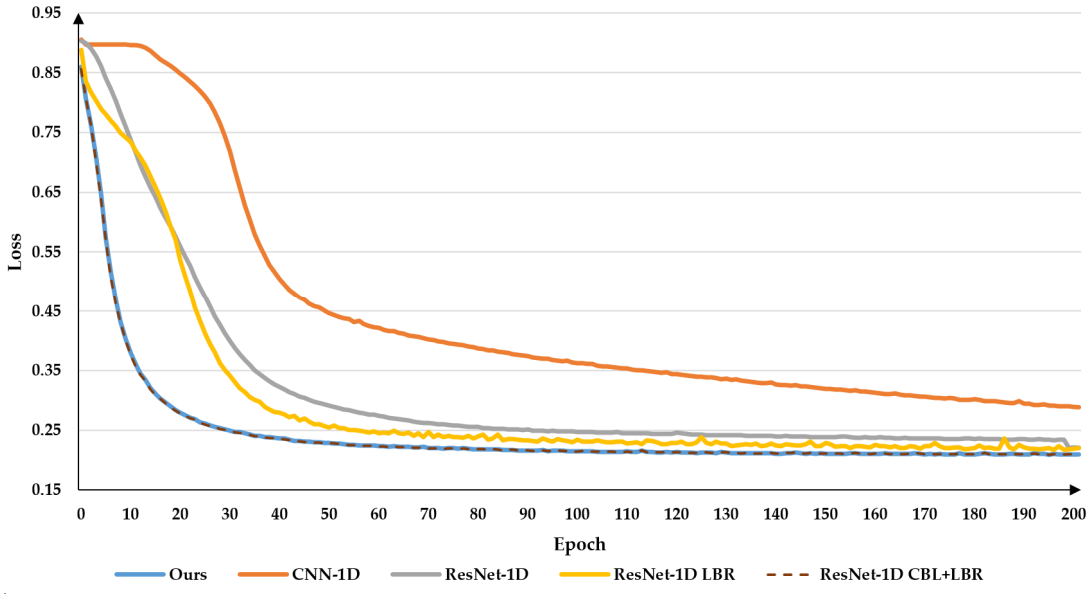

Figure 12. The loss optimization of all networks in training.

From Figure 12, we can see that the performance of the CNN-1D network is the worst, but after adding skip connections, the performance of ResNet-1D is significantly improved. This indicates that skip connections are very helpful for optimizing network loss. After adding the LBR module, the loss of ResNet-1D LBR further decreased. The addition of BN to the basic structural unit accelerates the convergence speed of the network. However, we can also see that replacing the activation function (ReLU or LeakyReLU) does not affect the network optimization.

Next, we use these networks to test the TP and verify its detection performance. The experimental data are *TP* with SNR = 0 dB and target signal size = 80. The experimental results are shown in Figure 13 and Table 11.

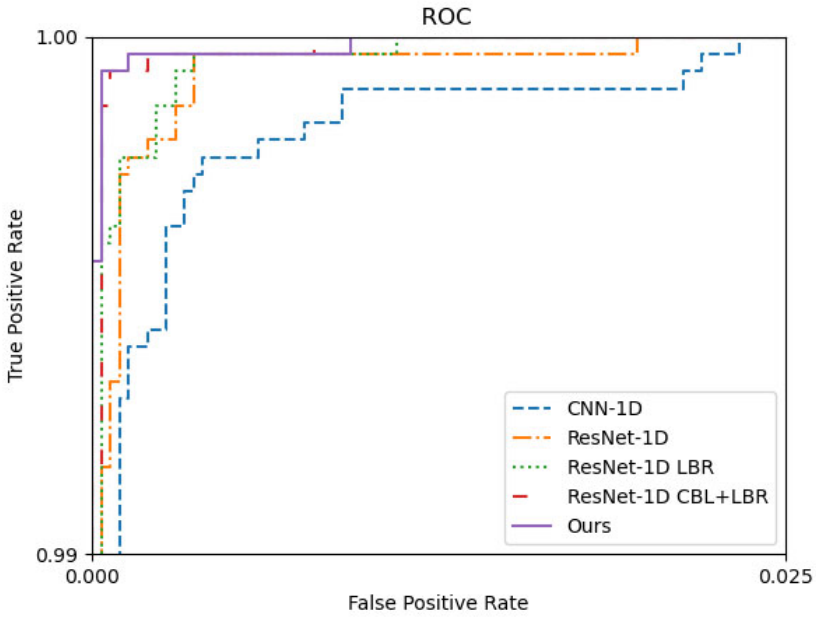

Figure 13. The detailed ROC of different networks.

Table 11. The AUC and accuracy of IOU of different networks.

Network	AUC	Accuracy of IOU
CNN-1D	$0.9999 + 0.3956 \times 10^{-4}$	64.58%
ResNet-1D	$0.9999 + 0.7801 \times 10^{-4}$	70.64%
ResNet-1D LBR	$0.9999 + 0.8678 \times 10^{-4}$	87.14%
ResNet-1D CBL+LBR	$0.9999 + 0.9389 \times 10^{-4}$	82.51%
Ours	$0.9999 + 0.9522 \times 10^{-4}$	86.97%

From the experimental results, it can be seen that all networks have good classification ability, but our network has the highest AUC. The positioning ability of networks without skip connections and LBR modules is poor. After adding skip connections, the network positioning ability is improved but not by very much. The addition of the LBR module greatly improves the positioning performance of the network. This indicates that skip connections can transfer the transient disturbance features extracted from shallow layers to deeper layers, preventing feature loss. Additionally, the LBR module can extract higher dimensional features, which helps to better locate transient disturbances. ResNet-1D LBR with no BN in its basic structural unit has the best positioning performance, but it is only 0.17% higher than that of our network. Adding BN will not affect the performance of the network in theory, but it can accelerate the convergence speed of the network.

5.2.2. Network Visualization

In this section, we conduct visualization research on the network to verify whether it has learned the distribution features of transient disturbances. Grad-CAM [45] (Gradient-weighted Class Activation Mapping) was used to visualize the network in order to verify whether the network has learned the features of the *TP*. The intensity of the target signal was set to 3, the size was set to 60, and its SNR was controlled at 3 dB. The chosen visualization layers were CBR5, CBR9, CBR13, and CBR16. The visualization results are shown in Figure 14, where the blue line is the original *TP* and the orange line is the heatmap calculated using Grad-CAM. The larger the value of the heatmap, the more interested the network is.

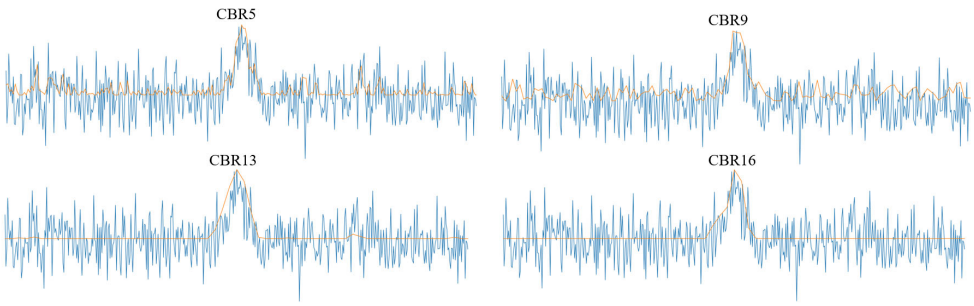

Figure 14. The heatmap of 1D-ResNet-16.

Figure 14 shows that the heatmap has the highest value at the target signal, proving that the network has fully learned the distribution features of the transient disturbance; however, the heatmap of the shallow layers also contains a lot of clutter. With an increase in the network depth, the clutter gradually decreases and the network learns more features of the transient disturbance. Therefore, the Grad-CAM visualization of the network shows that the network proposed in this study has interpretability.

6. Conclusions

To resolve the problem of moving point target detection at a low SNR, we converted the problem of point target detection into the problem of transient disturbance detection in the *TP* formed by each pixel. For the transient disturbance detection problem, we propose a detection framework to learn the distribution features of the transient disturbances. In

this framework, we first formulated different types of *TP* and generated a training dataset. Then, two networks, 1D-ResNet-8 and 1D-ResNet-16, were designed, which can adapt to the situation of detection speed priority and detection rate priority. Of the two networks, 1D-ResNet-16 has better detection performance than 1D-ResNet-8, but it requires more time. For detection tasks with high real-time requirements, 1D-ResNet-8 is a better choice. Adequate experiments showed that our *TP* model is correct and that our method is effective. Compared to other benchmark methods, the proposed method has obvious advantages when it comes to improving the detection rate and reducing the false alarm rate at a low SNR. Our method also has the fastest detection speed. In addition, we conducted ablation experiments to verify the superiority of our network and the CBR-1D structure, and the experimental results showed that all the modules of our proposed network were necessary. Network visualization research proved that our network learned the features of transient disturbances well.

Moreover, we studied the factors that affect detection performance and found that the size of the target signal had a greater impact on the detection results than the SNR of the *TP*. The detection performance of our method can be improved by increasing the sampling frame rate of the camera.

The method proposed in this study has the potential to be deployed in space-based or ground-based intelligent detection equipment. In the future, we will continue to study the problem of moving point target detection to propose a more efficient and stable detection method in order to make further contributions to this research field.

Author Contributions: Conceptualization, W.G. and P.W.; methodology, W.G.; software, W.G. and P.W.; validation, W.G., P.W. and Y.L.; formal analysis, W.G.; investigation, W.G., Y.L. and C.R.; resources, W.N.; data curation, W.G. and P.W.; writing—original draft preparation, W.G. and P.W.; writing—review and editing, W.G. and P.W.; visualization, W.G. and Y.L.; supervision, W.N., X.P. and Z.Y.; project administration, W.N., X.P. and Z.Y.; funding acquisition, W.N. and X.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Youth Innovation Promotion Association, Grant NO. E1213A02, and the Key Research Program of Frontier Sciences, CAS, Grant NO. 22E0223301.

Data Availability Statement: Not applicable.

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

References

1. Zhou, Q.; Yao, X.; Wang, C.; Hu, J.; Liu, P.; Lin, J. Adaptive Moving Ground-Target Detection Method Based on Seismic Signal. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 2503705. [CrossRef]
2. Du, J.; Lu, H.; Zhang, L.; Hu, M.; Deng, Y.; Shen, X.; Li, D.; Zhang, Y. DP-MHT-TBD: A Dynamic Programming and Multiple Hypothesis Testing-Based Infrared Dim Point Target Detection Algorithm. *Remote Sens.* **2022**, *14*, 5072. [CrossRef]
3. Eysa, R.; Hamdulla, A. Issues on infrared dim small target detection and tracking. In Proceedings of the 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), Xiangtan, China, 10–11 August 2019; pp. 452–456.
4. Bernhard, P.; Deschamps, M.; Zaccour, G. Large Satellite Constellations and Space Debris: Exploratory Analysis of Strategic Management of the Space Commons. *Eur. J. Oper. Res.* **2023**, *304*, 1140–1157. [CrossRef]
5. Chen, L.; Chen, X.; Rao, P.; Guo, L.; Huang, M. Space-Based Infrared Aerial Target Detection Method via Interframe Registration and Spatial Local Contrast. *Opt. Lasers Eng.* **2022**, *158*, 107131. [CrossRef]
6. Zhou, J.; Lv, H.; Zhou, F. Infrared small target enhancement by using sequential top-hat filters. In Proceedings of the International Symposium on Optoelectronic Technology and Application 2014: Image Processing and Pattern Recognition, Beijing, China, 13–15 May 2014; Sharma, G., Zhou, F., Eds.; Spie-Int Soc Optical Engineering: Bellingham, WA, USA, 2014; Volume 9301, p. 93011L.
7. Deshpande, S.D.; Er, M.H.; Ronda, V.; Chan, P. Max-Mean and Max-Median Filters for Detection of Small-Targets. *Proc. SPIE Int. Soc. Opt. Eng.* **1999**, *3809*, 74–83. [CrossRef]
8. Chen, C.L.P.; Li, H.; Wei, Y.; Xia, T.; Tang, Y.Y. A Local Contrast Method for Small Infrared Target Detection. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 574–581. [CrossRef]
9. Wei, Y.; You, X.; Li, H. Multiscale Patch-Based Contrast Measure for Small Infrared Target Detection. *Pattern Recognit.* **2016**, *58*, 216–226. [CrossRef]

10. Du, P.; Hamdulla, A. Infrared Small Target Detection Using Homogeneity-Weighted Local Contrast Measure. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 514–518. [CrossRef]
11. Xia, C.; Li, X.; Zhao, L.; Shu, R. Infrared Small Target Detection Based on Multiscale Local Contrast Measure Using Local Energy Factor. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 157–161. [CrossRef]
12. He, Y.; Li, M.; Wei, Z.; Cai, Y. Infrared small target detection based on weighted variation coefficient local contrast measure. In Proceedings of the Pattern Recognition and Computer Vision, Pt. III, Beijing, China, 29 October–1 November 2021; Ma, H., Wang, L., Zhang, C., Wu, F., Tan, T., Wang, Y., Lai, J., Zhao, Y., Eds.; Springer International Publishing Ag: Cham, Switzerland, 2021; Volume 13021, pp. 117–127.
13. Gao, C.; Meng, D.; Yang, Y.; Wang, Y.; Zhou, X.; Hauptmann, A.G. Infrared Patch-Image Model for Small Target Detection in a Single Image. *IEEE Trans. Image Process.* **2013**, *22*, 4996–5009. [CrossRef]
14. Dai, Y.; Wu, Y.; Song, Y.; Guo, J. Non-Negative Infrared Patch-Image Model: Robust Target-Background Separation via Partial Sum Minimization of Singular Values. *Infrared Phys. Technol.* **2017**, *81*, 182–194. [CrossRef]
15. Guo, J.; Wu, Y.; Dai, Y. Small Target Detection Based on Reweighted Infrared Patch-Image Model. *IET Image Process.* **2018**, *12*, 70–79. [CrossRef]
16. Rawat, S.S.; Verma, S.K.; Kumar, Y. Reweighted Infrared Patch Image Model for Small Target Detection Based on Non-ConvexScript Capital Lp-Norm Minimisation and TV Regularisation. *IET Image Process.* **2020**, *14*, 1937–1947. [CrossRef]
17. Xia, C.; Li, X.; Zhao, L.; Yu, S. Modified Graph Laplacian Model with Local Contrast and Consistency Constraint for Small Target Detection. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 5807–5822. [CrossRef]
18. Dai, Y.; Wu, Y.; Zhou, F.; Barnard, K. Attentional Local Contrast Networks for Infrared Small Target Detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 9813–9824. [CrossRef]
19. Ma, T.; Yang, Z.; Wang, J.; Sun, S.; Ren, X.; Ahmad, U. Infrared Small Target Detection Network with Generate Label and Feature Mapping. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 6505405. [CrossRef]
20. Hou, Q.; Zhang, L.; Tan, F.; Xi, Y.; Zheng, H.; Li, N. ISTDU-Net: Infrared Small-Target Detection U-Net. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 7000805. [CrossRef]
21. Ju, M.; Luo, J.; Liu, G.; Luo, H. ISTDet: An Efficient End-to-End Neural Network for Infrared Small Target Detection. *Infrared Phys. Technol.* **2021**, *114*, 103659. [CrossRef]
22. Yu, C.; Liu, Y.; Wu, S.; Hu, Z.; Xia, X.; Lan, D.; Liu, X. Infrared Small Target Detection Based on Multiscale Local Contrast Learning Networks. *Infrared Phys. Technol.* **2022**, *123*, 104107. [CrossRef]
23. Lv, G.; Dong, L.; Liang, J.; Xu, W. Novel Asymmetric Pyramid Aggregation Network for Infrared Dim and Small Target Detection. *Remote Sens.* **2022**, *14*, 5643. [CrossRef]
24. Hossen, M.K.; Tuli, S.H. A surveillance system based on motion detection and motion estimation using optical flow. In Proceedings of the 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), Beijing, China, 29 October–1 November 2016; pp. 646–651.
25. Singla, N. Motion Detection Based on Frame Difference Method. *Int. J. Inf. Comput. Technol.* **2014**, *4*, 1559–1565.
26. Sun, T.; Qi, Y.; Geng, G. Moving Object Detection Algorithm Based on Frame Difference and Background Subtraction. *J. Jilin University. Eng. Technol. Ed.* **2016**, *46*, 1325–1329. [CrossRef]
27. Yi, K.M.; Yun, K.; Kim, S.W.; Chang, H.J.; Choi, J.Y. Detection of moving objects with non-stationary cameras in 5.8 ms: Bringing motion detection to your mobile device. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2023; IEEE: Piscataway, NJ, USA; pp. 27–34.
28. Aprile, A.; Grossi, E.; Lops, M.; Venturino, L. Track-before-Detect for Sea Clutter Rejection: Tests with Real Data. *IEEE Trans. Aerosp. Electron. Syst.* **2016**, *52*, 1035–1045. [CrossRef]
29. Li, M.; Zhang, T.X.; Yang, W.D.; Sun, X.C. Moving Weak Point Target Detection and Estimation with Three-Dimensional Double Directional Filter in IR Cluttered Background. *Opt. Eng.* **2005**, *44*, 107007. [CrossRef]
30. Zhang, T.; Li, M.; Zuo, Z.; Yang, W.; Sun, X. Moving Dim Point Target Detection with Three-Dimensional Wide-to-Exact Search Directional Filtering. *Pattern Recognit. Lett.* **2007**, *28*, 246–253. [CrossRef]
31. Deng, L.; Zhu, H.; Tao, C.; Wei, Y. Infrared Moving Point Target Detection Based on Spatial-Temporal Local Contrast Filter. *Infrared Phys. Technol.* **2016**, *76*, 168–173. [CrossRef]
32. Ping-yue, L.; Lin, C.; Sun, S. Dim Small Moving Target Detection and Tracking Method Based on Spatial-Temporal Joint Processing Model. *Infrared Phys. Technol.* **2019**, *102*, 102973. [CrossRef]
33. Zhu, S.; Yang, D.; Jia, P.; Li, J.; Chai, X. Design and Implementation of Space-Time Combined Infrared Small Target Detection Algorithm. *Laser Infrared* **2021**, *51*, 388–392.
34. Liu, D.; Zhang, J.; Dong, W. Temporal Profile Based Small Moving Target Detection Algorithm in Infrared Image Sequences. *Int. J. Infrared. Milli Waves* **2007**, *28*, 373–381. [CrossRef]
35. Liu, D.; Li, Z. Temporal Noise Suppression for Small Target Detection in Infrared Image Sequences. *Optik* **2015**, *126*, 4789–4795. [CrossRef]
36. Liu, D.; Li, Z.; Wang, X.; Zhang, J. Moving Target Detection by Nonlinear Adaptive Filtering on Temporal Profiles in Infrared Image Sequences. *Infrared Phys. Technol.* **2015**, *73*, 41–48. [CrossRef]
37. Liu, X.; Li, L.; Liu, L.; Su, X.; Chen, F. Moving Dim and Small Target Detection in Multiframe Infrared Sequence with Low SCR Based on Temporal Profile Similarity. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 7507005. [CrossRef]

38. Wu, Y.; Yang, Z.; Niu, W.; Zheng, W. A Weak Moving Point Target Detection Method Based on High Frame Rate Image Sequences. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018.
39. Niu, W.; Zheng, W.; Yang, Z.; Wu, Y.; Vagvolgyi, B.; Liu, B. Moving Point Target Detection Based on Higher Order Statistics in Very Low SNR. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 217–221. [CrossRef]
40. Niu, W.; Fan, M.; Han, X.; Deng, H.; Guo, Y.; Zheng, W.; Yang, Z.; Peng, X. Moving point target detection based on temporal analysis of pixels in very low SNR. In Proceedings of the Seventh Symposium on Novel Photoelectronic Detection Technology and Applications, Kunming, China, 5–7 November 2021; Su, J., Chu, J., Jiang, H., Yu, Q., Eds.; SPIE Internationl Society of Optical Engineering: Bellingham, WA, USA, 2021; Volume 11763, p. 11763A7.
41. Pentland, A.P. A New Sense for Depth of Field. *IEEE Trans. Pattern Anal. Mach. Intell.* **1987**, *PAMI-9*, 523–531. [CrossRef]
42. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016; IEEE Computer Society: Washington, DC, USA, 2016; pp. 770–778.
43. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement 2018. *arXiv* **2018**, arXiv:1804.02767.
44. Han, J.; Zhang, X.; Jiang, Y.; Dong, X.; Li, Z.; Li, N. Small moving target detection in infrared sequences by using the multi-scale temporal relative local contrast. In Proceedings of the Advances in Guidance, Navigation and Control, Tianjin, China, 23–25 October 2020; Yan, L., Duan, H., Yu, X., Eds.; Springer: Singapore, 2022; pp. 4433–4445.
45. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.