algorithms

MDPI

Article

Multispectral Fusion Approach for Traffic Target Detection in Bad Weather

Yajing Han [1,2] and Dean Hu [1,2,*]

[1] State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China; hanyj@hnu.edu.cn

[2] Key Laboratory of Advanced Design and Simulation Techniques for Special Equipments, Ministry of Education, Hunan University, Changsha 410082, China

* Correspondence: hudean@hnu.edu.cn; Tel.:+86-13317315065

Received: 12 October 2020; Accepted: 25 October 2020; Published: 28 October 2020

Abstract: Visual traffic surveillance using computer vision techniques can be noninvasive, automated and cost effective. Traffic surveillance systems with the ability to detect, count and classify vehicles can be employed in gathering traffic statistics and achieving better traffic control in intelligent transportation systems. This works well in daylight when the road users are clearly visible to the camera, but it often struggles when the visibility of the scene is impaired by insufficient lighting or bad weather conditions such as rain, snow, haze and fog. Therefore, in this paper, we design a dual input faster region-based convolutional neural network (RCNN) to make full use of the complementary advantages of color and thermal images to detect traffic objects in bad weather. Different from the previous detector, we used halfway fusion to fuse color and thermal images for traffic object detection. Besides, we adopt the polling from multiple layers method to adapt the characteristics of large size differences between objects of traffic targets to accurately identify targets of different sizes. The experimental results show that the present method improves the target recognition accuracy by 7.15% under normal weather conditions and 14.2% under bad weather conditions. This exhibits promising potential for implementation with real-world applications.

Keywords: color and thermal images; traffic surveillance; vehicle detection; RCNN; bad weather

1. Introduction

A traffic surveillance camera system is an important part of an intelligent transportation system [1], which monitors traffic conditions and pedestrians by cameras mounted above the driveway. Surveillance video includes a lot of information [2], such as traffic flow, lane occupancy and vehicle type, that can be further processed by a computer to get real-time traffic conditions, accurate prediction and discrimination, to ultimately improve traffic congestion, accidents, environmental pollution and other issues.

In recent years, with the development of computer vision, more and more algorithms are applied to the field of traffic surveillance [3–5]. These methods improve the efficiency of road monitoring and let people get rid of boring and tedious work in front of the monitor [6]. However, it is well known that the application context plays an important role in practical applications of computer vision, as the conditions for the camera to capture images are not ideal sometimes, especially when the visibility of the scene is impaired by insufficient lighting or bad weather conditions such as rain, snow, haze and fog [7], as shown in Figure 1. The efficiency and accuracy of identification in these conditions are greatly reduced, which is unacceptable for traffic surveillance. To solve this problem, we propose a neural network to process color and thermal images collected by surveillance equipment and to obtain information about the images. We know that the thermal image represents the difference in

thermal radiation between the objects and the background, in other words, it describes the difference in temperature between the objects and the background, which is effective in all weather conditions. In contrast, the color image represents the visible light that is reflected or emitted from the object and background, it has high spatial resolution and sharp texture details [8]. Therefore, the fusion of these two complementary types of information can effectively improve the robustness of recognition, which is very helpful for object detection and tracking in some limitations caused by weather conditions [1].

Figure 1. Some examples of poor imaging conditions, the top images are the RGB images, and the bottom images are the images after edge detect. There is a lot of interference information in the edge information under the influence of light and reflection, in these cases the object detector may miss an object or detect a wrong object.

This paper mainly introduces a dual input faster region-based convolutional neural network (RCNN) approach based on infrared and visible images for object detection inspired by thermal and visible properties. Two convolutional neural networks are used to extract the visible and infrared image features of two images, respectively, to obtain the information contained in the two images, and then we connect the feature maps obtained by the two convolution neural networks on the channel dimension. In order to solve the problem of the large change in size of the target when identifying the surveillance images, we adjusted the anchor size of the region proposal network (RPN) to identify targets of various sizes.

The remainder of this paper consists of the following parts. Section 2 discusses the related works. Section 3 introduces the model of our work. The dataset and experiment are shown in Section 4. We give the conclusion in Section 5.

2. Related Works

2.1. Object Detection

The first object detector was called the Viola Jones object detector [9], proposed by Paul Viola and Michael Jones, which was technically classified as an object detector and mainly used in face detection. It provided a real-time solution and was used in many types of computer vision software. With the combination of deep learning and computer vision, the field of object detection is developing rapidly. The first object detector model based on deep learning was the OverFeat network [10] which used convolutional neural networks (CNNs) along with a sliding window approach. It classifies the various parts of the image one by one and then combines the results to produce the final prediction set. The application of CNNs to solve detection problems led to the trend seen in recent years.

Object detector models have undergone various changes since 2012. The RCNN [11] made a landmark contribution to target detection. It uses a selective search to find around 2000 regions where

objects are most likely to be present in them. These regions are cropped from the input image and resized to 7 by 7 pixels, then fed into the object detector model. The RCNN was the first two-step approach detector. However, due to its time and space inefficiency, a better model was needed. Hence, the fast RCNN [12] was proposed soon after, which further improved upon the RCNN. The fast RCNN reduces the overhead of running a region proposal by cropping the regions from a feature map instead of an input image. Additionally, it introduces a simpler single step training pipeline and a new loss function, and this loss function is easier to train and does not suffer from the gradient explosion problem.

Based on the RCNN and fast RCNN, Ren et al. proposed the faster RCNN in [13], which is a detector that learns end to end. The faster RCNN introduced the region proposal network (RPN), which used feature maps to generate object proposals instead of the selective search. The RPN has the capability of predicting regions of multiple scales and aspect ratios across the image by using a novel concept of anchors, in which the scale invariance is an important property of computer vision systems. One of the basic tasks of object detection is to recognize multi-scale objects. A feature pyramid is the most commonly used method but it is computationally and memory intensive. The feature pyramid network [14] (FPN) provides a top-down architecture with horizontal connectivity to construct high-level semantic feature maps at various scales, and the new state-of-the-art results were obtained in object detection, segmentation and classification by integrating FPNs into the pre-existing models.

The single shot multibox detector (SSD) [15,16] was published in 2015, the major difference between the SSD and previous architectures is that it was the first one to propose training on a feature pyramid, and high accuracy and recognition speed can be obtained by the SSD. You only look once (YOLO) [17], proposed by Redmon et al. in 2016, has similar architecture to the SSD. As a new target detection method, YOLO is characterized by rapid detection and high accuracy. The authors consider the target detection task as a regression problem of the target region and category prediction, so they used a single neural network to directly predict the object boundary and category probability, and to realize the end-to-end object detection.

2.2. Computer Vision for Traffic Surveillance Systems

With the development of automatic driving and intelligent transportation technology, traffic surveillance has become increasingly important, with the advantages of being unmanned and highly efficient. At present, traffic target recognition based on RCNNs has become the mainstream, and the several abovementioned kinds of detector have been applied in the field of traffic identification. As for the actual application effect [6], the single-stage detectors (YOLO, SSD, etc.) identify quickly but with low accuracy. Moreover, YOLO has trouble identifying small goals. On the contrary, two-stage detectors (faster RCNN, etc.) possess higher accuracy but are also slower.

Another application of traffic surveillance is multi-target tracking (MOT), which is used for tracking multiple moving targets at the same time. The core of detection tracking is to correctly associate detection boundary boxes between video frames. In most cases, the correlation measures are based on appearance similarity and action consistency. In terms of association strategy, it can be divided into offline global optimization and online association. The offline global methods always use network flows [18] or probabilistic graph models [19] to solve MOT problems. This method has a strong tracking ability for targets that have been blocked by other objects for a long time. Unlike the offline global approach, the online approach [20,21] is frame based and focuses on establishing the right connection between each pair of frames, which makes it less time consuming. At present, a majority of online tracking methods have a strong anti-interference ability towards false detection signals, short-term block and lost tracing objects. But these methods do not work well for objects that have been blocked by other objects for a long time.

3. Methods

In this part, we introduce the network structure of our proposed dual input faster RCNN (D-F Net). As stated in Figure 2, the D-F Net model is two-step approach model, the same as faster RCNN. The first part is the convolutional base for feature extraction based on ResNet and the RPN. It contains two branches for processing infrared and visible images, respectively. We connect the feature maps of infrared and visible images processed by the basic network on the channel dimension, and then feed this into the RPN to find a list of regions that could contain an object. The second part is an object detector model. The regions from the above step run through the object detector model and generate the class probabilities and offset coordinates for each region.

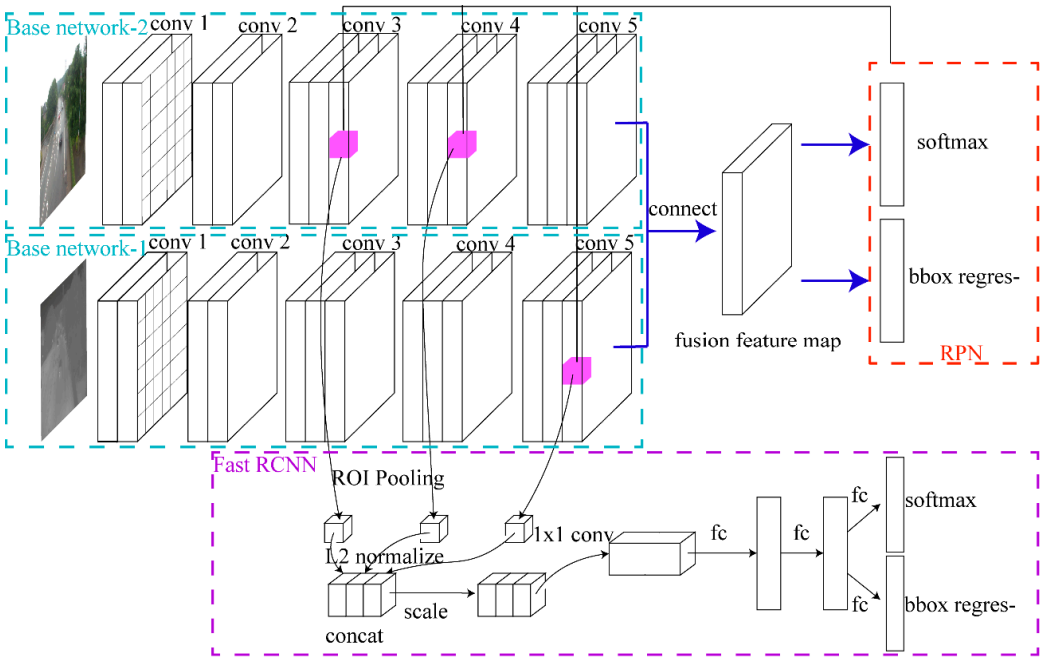


Figure 2. The architecture of dual input faster RCNN (D-F Net). It is a middle level feature fusion model.

3.1. Convolutional Base and RPN

Different from the traditional faster RCNN network, our base convolutional network consists of two parts, the infrared image feature extract convolutional network (base network 1), and the visible image feature extract convolutional network (base network 2). In the base network, we use Inception ResNet [22] to generate the initial feature map. In detail, we use fine-tuned ResNet-34 and ResNet-50 as base network 1 and base network 2, respectively. In order to reduce the number of weights, we adjusted the dimensions of ResNet-34 and ResNet-50, as shown in Table 1. In addition, base network 2 is pretrained on the ImageNet dataset, and base network 1 uses Xavier for weight initialization. After the convolution processing of the two images, the corresponding feature maps of the two images are obtained, and then the two feature maps are connected along the channel axis and computed with a 1×1 convolution to obtain the fusion feature map, as shown in Figure 3a. The region proposal network is used to generate proposals from the input 512-dimensional fusion feature maps extracted by convolutional base, as shown in Figure 3b.

Table 1. Overview of base network.

Layer Name	Base Network 1	Base Network 2
Conv_1	7×7, 64, Stride 2 3×3 max pool, Stride 2	
Conv_2	$\begin{bmatrix} 3\times3 & 32 \\ 3\times3 & 32 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1 & 32 \\ 3\times3 & 32 \\ 1\times1 & 64 \end{bmatrix} \times 3$
Conv_3	$\begin{bmatrix} 3\times3 & 64 \\ 3\times3 & 64 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1 & 64 \\ 3\times3 & 64 \\ 1\times1 & 128 \end{bmatrix} \times 4$
Conv_4	$\begin{bmatrix} 3\times3 & 128 \\ 3\times3 & 128 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1 & 128 \\ 3\times3 & 128 \\ 1\times1 & 256 \end{bmatrix} \times 6$
Conv_5	$\begin{bmatrix} 3\times3 & 256 \\ 3\times3 & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1 & 256 \\ 3\times3 & 256 \\ 1\times1 & 512 \end{bmatrix} \times 3$

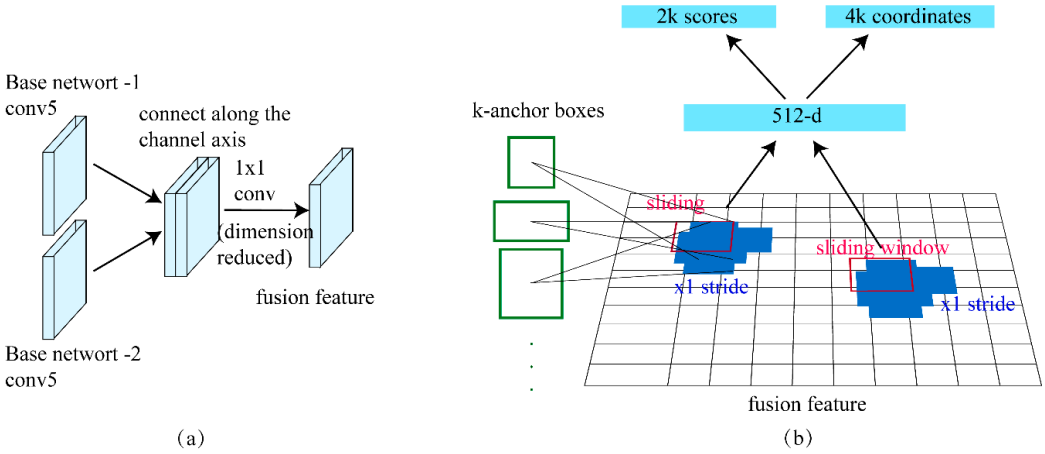


Figure 3. (**a**) We concatenate each pooled feature along the channel axis and reduce the dimension with a 1×1 convolution; (**b**) the region proposal network.

A series of anchor boxes are output by a shared small network that slides over the fusion feature map of the RPN. The shared small network consists of an intermediate layer and a full connection layer. The convolution kernel size is 3×3 in the RPN. In traffic surveillance, the size and shape of the objects vary greatly. Thus, the aspect ratios are set as 1:1, 0.4:1 and 2.2:1, while the scales are set as 128^2, 256^2 and 512^2 pixels.

The RPN generates high-quality region proposals, and the fast RCNN learns those features and performs classification. The loss function for an image is defined as:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (1)$$

where i is the anchor's index in a mini batch, P_i^* is the ground truth label (when a proposal is an object, $P_i^* = 1$, otherwise $P_i^* = 0$) and P_i is the anchor i's predicted probability of being an object. t_i^* is the ground truth box of a positive anchor and t_i is a vector indicating four parameterized coordinates of the predicted bounding box. N_{cls} and N_{reg} are two normalization parameters, L_{cls} is classification loss, which is log over two classes (object versus not object). L_{reg} is regression loss, the term $p_i^* L_{reg}$ means the regression loss that is active only for a positive anchor.

For bounding box regression, the parameterization of the four coordinates can be written as:

$$t_x = \frac{(x - x_a)}{w_a} \quad t_y = \frac{(y - y_a)}{h_a} \quad (2)$$

$$t_w = \log\left(\frac{w}{w_a}\right) \quad t_h = \log\left(\frac{h}{h_a}\right) \tag{3}$$

$$t_x^* = \frac{(x^* - x_a)}{w_a} \quad t_y^* = \frac{(y^* - y_a)}{h_a} \tag{4}$$

$$t_w^* = \log\left(\frac{w^*}{w_a}\right) \quad t_h^* = \log\left(\frac{h^*}{h_a}\right) \tag{5}$$

where x and y represent the coordinates of the box center, w and h represent the weight and height of the bounding box, respectively, and x, x_a and x^* are the predicated box, anchor box and ground truth box, respectively (likewise for y, w, h y, w, h). This can be thought of as bounding box regression from an anchor box to a nearby ground truth box.

3.2. Object Detector Model

We know that the detection of small targets is challenging for the fast RCNN, because the fast RCNN is based on the last convolution layer conv5_3 for object detection. The receptive field in the feature map is quite large, which is insufficient to encode object information. In addition, the deeper the convolutional layer, the more information each pixel on the feature map contains that is not the object. In this paper, we adopt the layers of base network 2 conv3_3, conv4_3 and base network 1 conv5_3 to extract features where the high-resolution information of the lower-level layer will not be lost in terms of small-scale objects. The region proposals and input feature maps can be collected through an region of interest (ROI) pooling layer. ROI pooling layer is characterized by the non-fixed size of feature maps. The input of three network connections is the feature map of base network 2 conv3_3, conv4_3 and base network 1 conv5_3. The base network 2 feature map of conv3_3 and conv4_3 is used to obtain relevant information about small and medium objects, and the base network 1 feature map of conv5_3 is used to obtain relevant information about large objects. In addition, by connecting the middle-level and low-level layers of base network 2 and the high-level layers of base network 1, the characteristics of infrared and optical images can be fully utilized.

According to the Inside–Outside NET [23], as more features are connected, we must consider issues of dimensionality and amplitude. We know that the final feature's shape must be $512 \times 7 \times 7$ so that it can be processed by the first final fully connected layer. The three ROIs corresponding to object proposal are fed into three corresponding ROI pooling layers, the number and scale of channels differ on each layer of convolution base, with a higher scale on a lower layer. To satisfy the shape constraint, we concatenate three pooled features along the channel axis and reduce the dimension by a 1×1 convolution. Besides, the L2 normalization is applied to normalize amplitudes for each tensor ahead of concatenation [24], and then to scale each tensor independently. For a d-dimensional input vector $x = (x_1, x_2, \cdots, x_d)$, L2 normalization is expressed by the following equation:

$$\|x\|_2 = \sqrt{\left(\sum_{i=1}^{d} |x_i|^2\right)} \tag{6}$$

$$\hat{x} = \frac{x}{\|x\|_2} \tag{7}$$

where x is the input pixel vector, d is the dimension of each ROI polling tensor and $\hat{x}$ is the normalized pixel vector.

In the last step of the object detector model, the 7×7 feature vector is input into two fully connected layers that branch into two sibling layers and, finally, the softmax layers for object classification and the regression function for bounding box regression results.

4. Procedure and Results

4.1. Dataset and Data Augmentation

The dataset is traffic surveillance in [7], collected by Chris H. Bahnsen et al. This dataset is focused on collecting traffic surveillance video in rainfall and snowfall, with 22 videos from seven different traffic intersections, and each monitor video is five minutes long. The illumination of the scenes varies from broad daylight to dusk and night. The features of scenes are obscured by the glare of the headlights of cars and streetlamps, reflected in puddles and blurred by raindrops on the camera lens.

This dataset uses optical and thermal infrared cameras to capture video sequences of road users. We have selected 600 frames randomly from each five-minute sequence and any road user in these frames is annotated on an instance level with a corresponding category label. In total, 15,600 frames are annotated, containing 33,297 objects in six categories (car, truck, bike, motorbike, bus, person). We divide 15,600 pairs of infrared and visible images into a training set and a testing set according to the ratio of 8:2. Moreover, all the images in this dataset are normalized to the size of 640×480 pixels and input to the convolutional neural networks. Some examples of the dataset are shown in Figure 4.

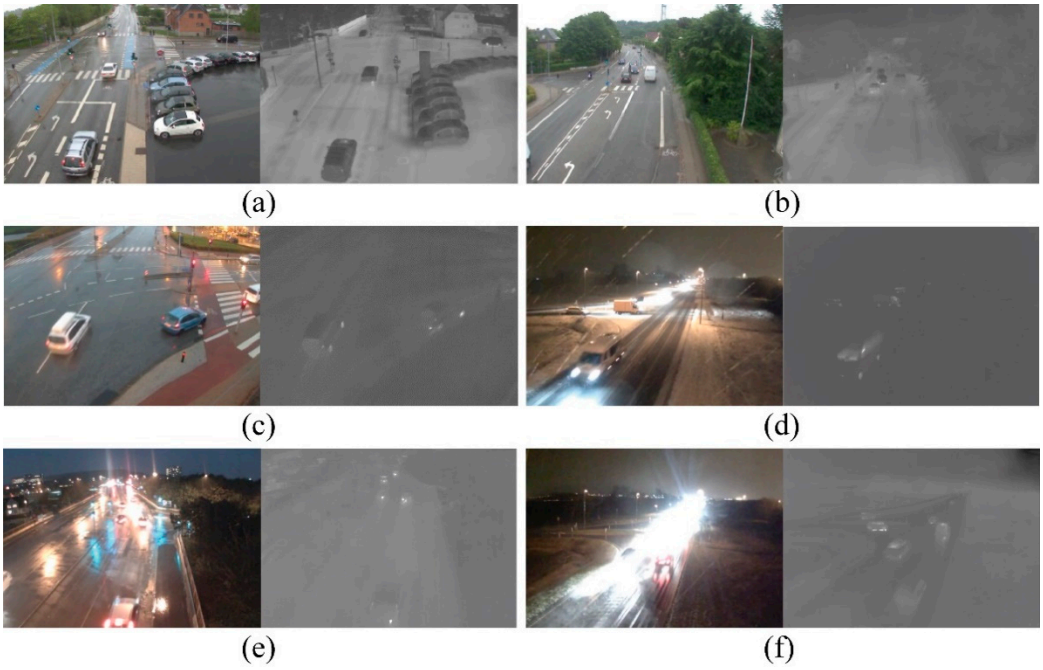


Figure 4. Some examples of images of traffic surveillance, the visible image is on the left and the infrared image is on the right. In (**a**–**c**), there are scenes ranging from sunny to rainy in the daytime; in (**d**–**f**), there are bad lighting conditions and bad visibility of the scenes; (**c**) snow; (**d**) rain and reflections; (**f**) raindrops on the camera lens.

There are 15,600 pairs of photos in the dataset, but some categories of images are insufficient, such as pedestrian, motorbike and bicycle, which account for a small proportion. In order to reduce the risk of overfitting, we use data augmentation to enrich the original dataset. The spatial transformation of coordinates is used to enlarge our dataset, and the transform of coordinates can be expressed by the following formula:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{T} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{8}$$

where (x, y) are pixel coordinates in the original image and $(x'x', y'y')$ are the corresponding pixel coordinates of the transformed images. The most common type of coordinate transformation is affine transformation, the general form is as follows:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{9}$$

This transformation can scale, rotate, translate or shear an image by changing the value of the elements of matrix A. We apply multiple transformations to the image by multiplying different matrices A. The matrix A that we use for affine transformation is shown in Appendix A.

4.2. Implementation Details

Our experiments are based on the open source framework of Mxnet [25] with a Python interface, and the model introduced in Section 3. Fast RCNN and RPN modules are fine-tuned end-to-end with Pascal Voc_2007 and Pascal Voc_2013 [26]. In our experiment, the threshold of the intersection-over-union is set to 0.7 and the learning rate is set to 0.005 for the first 40 k iterations and 0.001 for the last 40 k iterations. In addition, the weight decay and momentum are set to 0.0005 and 0.9, respectively. Two pairs of infrared and visible images make up a batch. Other network hyper-parameters of our approach are the same as those in traditional faster RCNN. The infrared and visible images in our dataset have the same perspective, and there is no need for image registration. If two images have different perspectives, they should be registered before training. Our experiments were performed on a 64 bit Ubuntu 16.04 computer with CPU Intel(R) Core (TM) i7-6700K CPU@ 4.00 GHz and NVIDIA GeForce GTX 1050. During the training process, we use the feature map connection approach, as introduced in Section 3.

4.3. Experimental Results

In this section, we evaluate our approach by comparing it with one multispectral model and two single-spectral models. In order to evaluate the performance of our proposed multispectral recognition method, we also trained three other object detectors, including two faster RCNN models trained by color or thermal images only and a vanilla ConvNet. To facilitate the distinction, we named the faster RCNN trained by visible images as F-rgb and the faster RCNN trained by thermal images as F-ther. The intersection-over-union (IOU), non-maximum suppression (nms), weight decay, momentum and other parameters of reference models are consistent with our model.

Figure 5 shows the curve of the miss rate and false positives per image under all-weather conditions of the four models, where the miss rate is the number of objects not recognized by the detector divided by the total number of ground truth and the false positives (FPs) are those negative samples identified by the detector as positive samples. It can be seen from the figure that neither single spectral recognizer is as effective as the fused recognizer. Their miss rate and FPs per image are high in most cases, and the comparison of the two fused recognizers shows that our method is an improvement over vanilla ConvNet. The reason why we use vanilla ConvNet as a comparison here is that it performs better in the fusion detector, especially for pedestrian recognition [27].

During the testing process, we quantify the performance of our approach by computing the mean average precision (mAP). In particular, we evaluate the D-F Net on six different weather conditions (sunny day, clear night, rainy day, rainy night, reflection and blur) to simulate the working environment. The calculation results are shown in Table 2, it can be seen from the data that the present approach has improved performance in different environments, the mAP of our approach is 75.05% for good weather and 65.15% for bad weather, while the mAP of vanilla ConvNet is 67.9% for good weather and 50.95% for bad weather. According to the results, the accuracy of the present method is improved by 7.15% for good weather and 14.2% for bad weather compared with vanilla ConvNet.

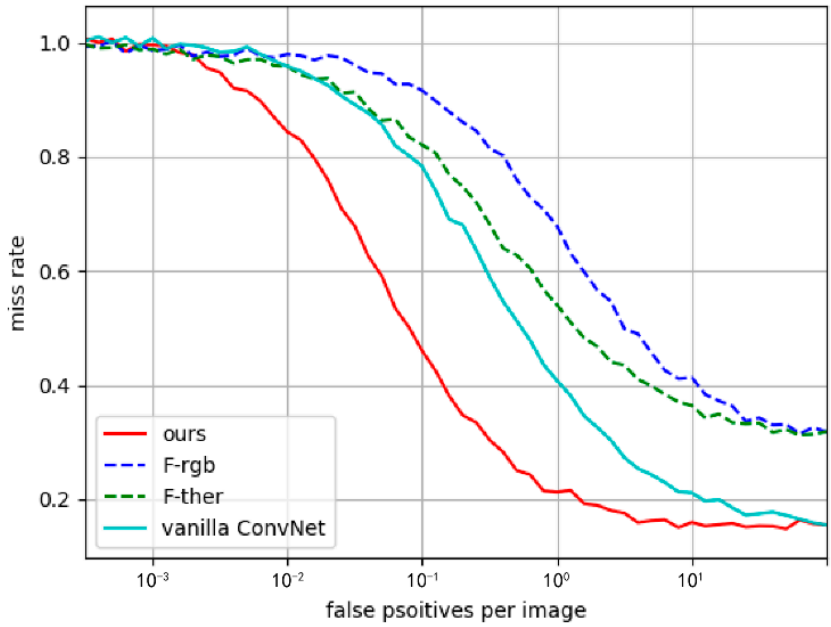


Figure 5. Comparison of all-day detection results reported on the test set.

Table 2. Comparison of two multispectral detectors under different weather conditions.

		Good Weather		Bad Weather			
		Sunny Day	**Clear Night**	**Rainy Day**	**Rainy Night**	**Reflection**	**Blur**
mAP	Ours	79.8	70.3	69.2	64.9	65.4	61.1
	Vanilla ConvNet	75.3	60.5	66.4	54.2	43.3	39.9
Average mAP	Ours	75.05			65.15		
	Vanilla ConvNet	67.9			50.95		

Figure 6a compares the training process of loss function between the vanilla ConvNet and our D FRCNN. It can be seen that in the training process, the D-F Net model can converge rapidly, and the loss function is smaller than the vanilla ConvNet after about 30,000 iterations, in which the vanilla ConvNet shows signs of overfitting. Figure 6b is the comparison of receiver operating characteristic curve (ROC) between our approach and the vanilla ConvNet. We know that the area enclosed by the ROC curve and coordinate axis is an index to measure the performance of the model. The larger the area, the better the recognition performance of the detector. We can see from Figure 6b that our proposed method has more reliable performance and better robustness than vanilla ConvNet under the premise of considering complex weather conditions, because it is more accurate under the same recall rate.

Figure 7 shows the detection results for some bad weather conditions, these images were selected from the test set, including sunny, rainy, snowy and blurry lens conditions. As we can see from these test results, our method can still ensure the accuracy of recognition under some special circumstances by adopting multispectral information fusion. In addition, we adopt the layers of base network 2 conv3_3, conv4_3 and base network 1 conv5_3 to extract features, and the detector is also able to recognize targets with a large size difference (such as the pedestrian and the bus in the picture on the top left).

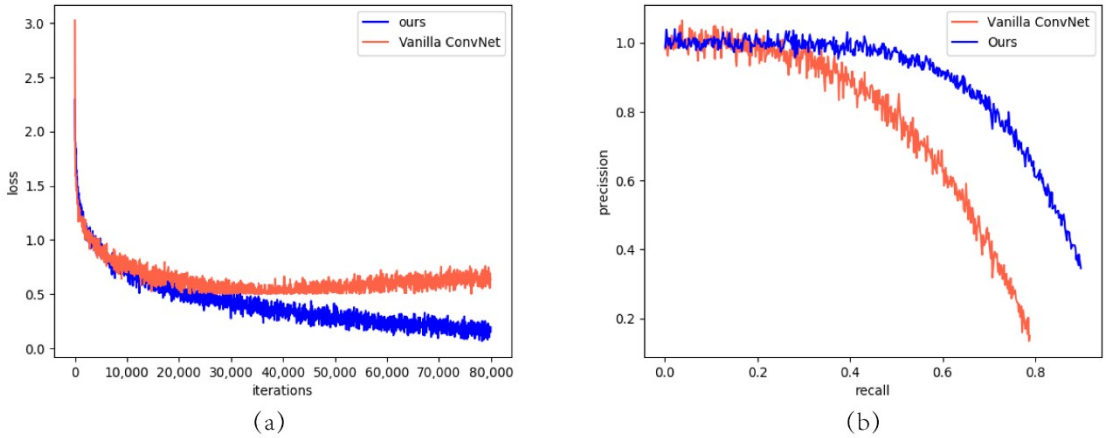


Figure 6. D-F Net performance compared with the vanilla ConvNet: (**a**) Loss function curve of 80,000 iterations; (**b**) precision–recall curve.

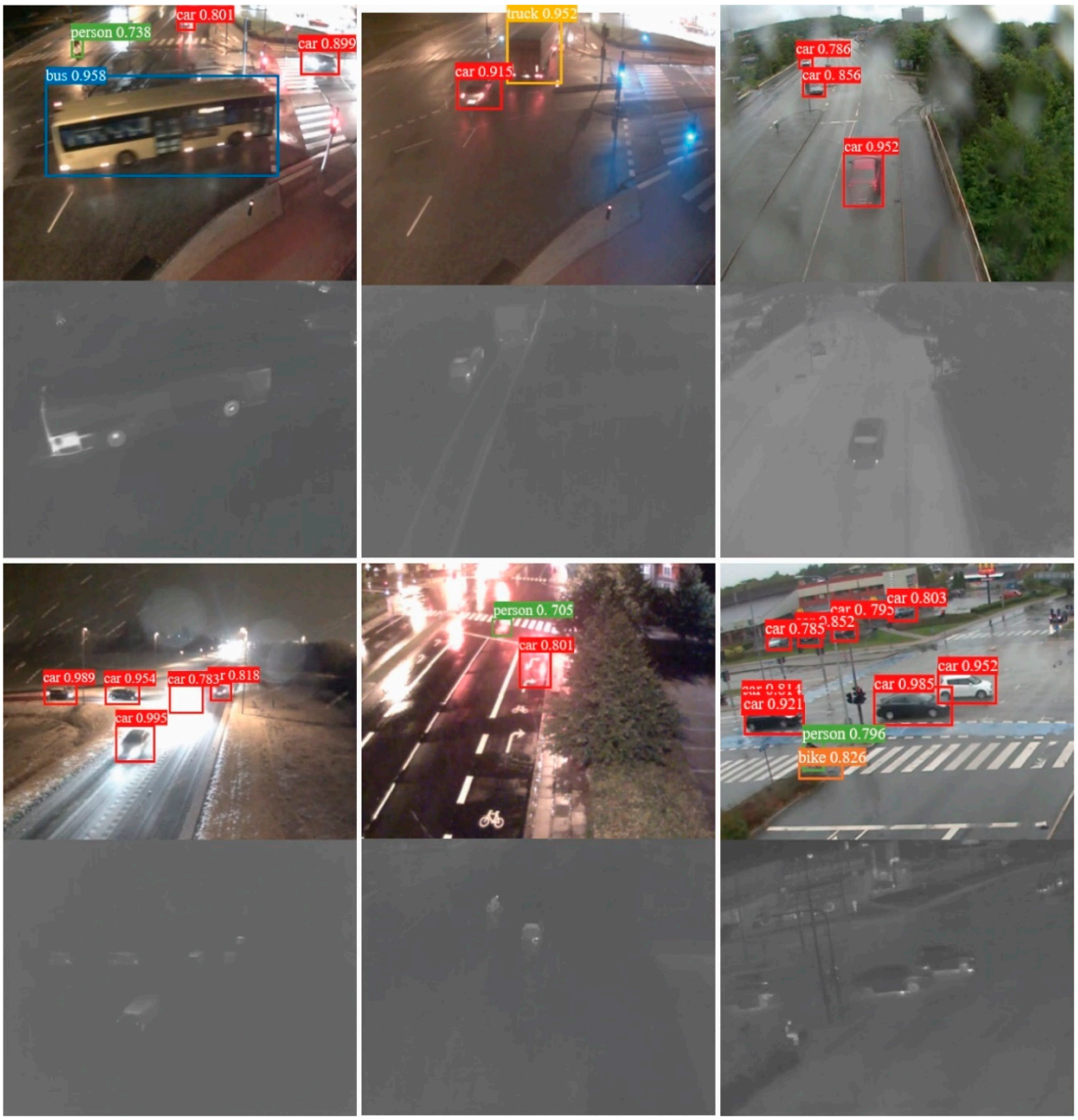


Figure 7. Some traffic object detection results for good weather and bad weather.

5. Discussion

In this paper, a new approach is proposed for traffic surveillance object detection in bad weather based on a faster RCNN. We used two convolutional neural networks to obtain high-quality object proposals from the information of infrared and visible images. With the addition of infrared images, the performance of regression and classification has been improved to some extent, especially in severe weather environments. Besides, we adopt the polling from multiple layers method to adapt the characteristics of large size differences of objects in traffic targets so as to accurately identify targets of different sizes. Our dual input faster RCNN can also be applied to other object detection tasks with two information sources.

In the future, we will explore how to improve the time efficiency of D-RCNN and optimize the regional recommendation network with the homogeneity of thermal radiation images. On the other hand, we will try to use probability theory and the multispectral fusion recognition method to track the target along the timeline and establish connections between objects identified in different frames, so that it can better serve intelligent transportation systems.

Author Contributions: Conceptualization, D.H. and Y.H.; methodology, Y.H.; software, Y.H.; validation, D.H.; formal analysis, Y.H.; investigation, D.H.; resources, D.H.; data curation, Y.H.; writing—original draft preparation, Y.H.; writing—review and editing, D.H.; visualization, Y.H.; supervision, D.H.; project administration, D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank the School for providing the computing equipment and Zhihua Zhong for his guidance.

Conflicts of Interest: This article has no conflict of interest with any organization or individual.

Appendix A

Table A1. Affine matrices used for data enhancement.

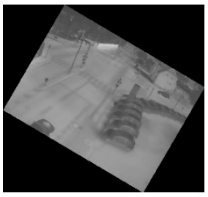
Transformation Type	Affine Matrix	Coordinate Transform	Example
Identify	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x' = x + t_x$ $y' = y + t_y$	
Rotation	$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x\cos\theta - y\sin\theta$ $y' = x\cos\theta + y\sin\theta$	

Table A1. *Cont.*

Transformation Type	Affine Matrix	Coordinate Transform	Example
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = s_v y + x$ $y' = y$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	

References

1. Zhang, B.; Zhou, Y.; Pan, H. Vehicle Classification with Confidence by Classified Vector Quantization. *IEEE Intell. Transp. Syst. Mag.* **2013**, *5*, 8–20. [CrossRef]
2. Park, S.H.; Jung, K.; Hea, J.K.; Kim, H.J. Vision-Based Traffic Surveillance System on the Internet. In Proceedings of the Third International Conference on Computational Intelligence and Multimedia Applications, New Delhi, India, 23–26 September 1999.
3. Tang, Y.; Zhang, C.; Gu, R.; Li, P.; Yang, B. Vehicle Detection and Recognition for Intelligent Traffic Surveillance System. *Multimed. Tools Appl.* **2017**, *76*, 5817–5832. [CrossRef]
4. Kumar, T.; Kushwaha, D.S. Traffic Surveillance and Speed Limit Violation Detection System. *J. Intell. Fuzzy Syst.* **2017**, *32*, 3761–3773. [CrossRef]
5. Hsieh, J.W.; Yu, S.H.; Chen, Y.S.; Hu, W.F. An Automatic Traffic Surveillance System for Vehicle Tracking and Classification. *Trans. Intell. Transp. Syst.* **2006**, *7*, 175–187. [CrossRef]
6. Mao, T.; Zhang, W.; He, H.; Lin, Y.; Kale, V.; Stein, A.; Kostic, Z. Aic2018 Report: Traffic Surveillance Research. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018.
7. Bahnsen, C.H.; Moeslund, T.B. Rain Removal in Traffic Surveillance:Does It Matter? *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 2802–2819. [CrossRef]
8. Ma, J.; Ma, Y.; Li, C. Infrared and Visible Image Fusion Methods and Applications: A Survey. *Inf. Fusion* **2019**, *45*, 153–178. [CrossRef]
9. Viola, P.; Jones, M. Rapid Object Detection Using a Boosted Cascade. In Proceedings of the CVPR 2001, Kauai, HI, USA, USA, 8–14 December 2001.
10. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. Overfeat: Integrated Recognition, Localization and Detection. In Proceedings of the ICLR 2014, Banff, AB, Canada, 14–16 April 2014.
11. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the CVPR 2014, Columbus, OH, USA, 23–28 June 2014.
12. Girshick, R. Fast R-CNN. In Proceedings of the ICCV 2015, Santiago, Chile, 7–13 December 2015.
13. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the NIPS 2015, Montreal, QC, Canada, 7–12 December 2015; pp. 1440–1448.
14. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the CVPR 2017, Honolulu, Hawaii, 21–26 July 2017.
15. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. *SSD: Single Shot MultiBox Detector*; Springer: Cham, Switzerland, 2016.
16. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. Dssd: Deconvolutional Single Shot Detector. *arXiv* **2016**, arXiv:1701.06659.
17. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the CVPR 2016, Las Vegas, NV, USA, 27–30 June 2016; p. 1.

18. Pirsiavash, H.; Ramanan, D.; Fowlkes, C.C. Globally-Optimal Greedy Algorithms for Tracking a Variable Number. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 21–23 June 2011.
19. Yang, B.; Huang, C.; Nevatia, R. Learning Affinities and Dependencies for Multi-Target Tracking Using Acrf Model. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 21–23 June 2011.
20. Choi, W. Near-Online Multi-Target Tracking with Aggregated Local Flow Descriptor. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015.
21. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017.
22. Bebis, G.; Gyaourova, A.; Singh, S.; Pavlidis, I. Face Recognition by Fusing Thermal Infrared and Visible Imagery. *Image Vis. Comput.* **2006**, *24*, 727–742. [CrossRef]
23. Kumar, P.; Mittal, A.; Kumar, P. Fusion of Thermal Infrared and Visible Spectrum Video for Robust Surveillance. In *Computer Vision, Graphics and Image Processing*; Springer: Berlin/Heidelberg, Germany, 2006.
24. Wagner, J.; Fischer, V.; Herman, M.; Behnke, S. Multispectral Pedestrian Detection Using Deep Fusion Convolutional Neural Networks. In Proceedings of the European Symposium on Artificial Neural Networks, Bruges, Belgium, 27–29 April 2016.
25. Liu, J.; Zhang, S.; Wang, S.; Metaxas, D.N. Multispectral Deep Neural Networks for Pedestrian Detection. In Proceedings of the British Machine Vision Conference, York, UK, 19–22 September 2016.
26. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-V4, Inception-Resnet and the Impact of Residual Connections on Learning. In Proceedings of the National Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
27. Bell, S.; Lawrence Zitnick, C.; Bala, K.; Girshick, R. Inside-Outside Net:Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.