Appendix

In Appendix A, we elaborate on three loss functions for object detection. In Appendix B, we describe the training details of experiments and provide additional results of digital attacks. In Appendix C, we supplement the settings and test results of three defense methods.

A. Details on Loss Functions

Given a training dataset $S = \{(x_i, l_i)\}_{i=1}^{N}$ with N training samples for the thermal infrared object detection model training, we denote each image by $x_i \in \mathbb{G}$ with $\mathbb{G}$ representing the grayscale space with only grayscale channels, the BBox information and class information of each object in the corresponding image by l_i. Let f be the object detection model (e.g., YOLO v5), and θ be the parameters. Denote the model f trained on dataset S by $f(S, \theta)$, which will return three types of information: classification probability, BBox coordinates, and confidence. Since Yolo v5 divides each image into many grids (usually 20×20, 40×40, 80×80) for detection, there are three detection boxes in each grid. Let M be the total number of detection boxes in each image. Let us assume that the object classification probability of the jth detection BBox in the ith ground truth label is $l_{cls}(i, j)$ and the classification probability obtained be the model is $f_{cls}(i, j, \theta)$. Then, the calculation formula of the classification loss L_{cls} can be represented as follows:

$$\begin{aligned} L_{cls} = &\sum_{i=1}^{N}\sum_{j=1}^{M} -l_{cls}(i,j) * \log f_{cls}(i,j,\theta) \\ &- (1 - l_{cls}(i,j)) * \log\left(1 - f_{cls}(i,j,\theta)\right). \end{aligned} \quad (15)$$

Similarly, let us assume that the confidence of jth detection bounding box in the ith ground truth label be $l_{conf}(i, j)$, and the confidence obtained by the model be $f_{conf}(i, j, \theta)$. Then, the calculation formula of the confidence loss L_{conf} can be represented as follows:

$$\begin{aligned} L_{conf} = &\sum_{i=1}^{N}\sum_{j=1}^{M} -(1 - l_{conf}(i,j)) * \log\left(1 - f_{conf}(i,j,\theta)\right) \\ &-l_{conf}(i,j) * \log f_{conf}(i,j,\theta). \end{aligned} \quad (16)$$

The formula for calculating the bounding box regression loss L_B is as follows:

$$\begin{aligned} L_B &= \sum_{i=1}^{N}\sum_{j=1}^{M} 1 - CIOU(l_B(i,j), f_B(i,j,\theta)) \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} 1 - (IOU(l_B(i,j), f_B(i,j,\theta)) - \frac{\zeta^2}{\eta^2} - \mu\nu) \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} 1 - (\frac{|l_B(i,j) \cap f_B(i,j,\theta)|}{|l_B(i,j) \cup f_B(i,j,\theta)|} - \frac{\zeta^2}{\eta^2} - \mu\nu), \end{aligned} \quad (17)$$

where $l_B(i, j)$ and $f_B(i, j, \theta)$ represent the jth ground truth bounding box and the corresponding predicted bounding box in the ith image, respectively. ζ is the distance between the center points of $l_B(i, j)$ and $f_B(i, j, \theta)$. η is the diagonal length of the smallest enclosing rectangle of $l_B(i, j)$ and $f_B(i, j, \theta)$. ν is the similarity of the aspect ratio of $l_B(i, j)$ and $f_B(i, j, \theta)$, and μ is the influence factor of ν.

B. Detailed Experiment Settings and Additional Experiment Results

Datasets and Models. $Flir\_v2\_T$ is more challenging because it has more diverse scenes with denser and more object classes. For YOLO v5, we choose the Adma algorithm as the optimizer. The initial learning rate is set to 1e-3, and the learning rate period decline rate is set to 1e-2. The batch size is set to 16, and the training epochs is set to 300. Usually, the model will converge in advance and stop training. YOLO v3 has been applied to autonomous vehicles, so it has great research significance. For the YOLO v3 model, we keep the same parameter settings as YOLO v5. The training of the Faster RCNN is divided into two stages: the freezing stage and the unfreezing stage. Both stages choose the Adma algorithm as the optimizer, and the batch size is set to 8. In the freezing phase, the learning rate is set to 1e-3 and the training epochs are 50. In the unfreezing phase, the learning rate is set to 1e-4 and the training epochs are 150.

Baselines. The BAF for each category in the test dataset is calculated separately and independently. We only show the BAFs of the attacked categories — person and car.

B.1. Object-Affecting Attack

For OAA, the other two parameters we focus on are size scaling ratio and relative location. Unless otherwise specified, the *default parameters* take the following combination: $p = 192$, $q = 20\%$, $\lambda = 0.04$ and $rl =$ M0. $\lambda = 0.04$ means the length and width of the trigger to be one-fifth of the object BBox. $rl =$ M0 means the center point of the trigger coincides with the center point of the object BBox. We list the experimental results in Table 5. The well-performing parameters and results are marked in boldface.

Pixel Value (p). We found that nearly all thermal infrared images in the datasets have the highest pixel value of 255 and the lowest pixel value of 0. It can be concluded that the mapping relationship between temperature and pixels is $T \in [T_{min}, T_{max}] \xrightarrow{g(\cdot)} p \in [0, 255]$, where T_{min} and T_{max} are the lowest temperature and the highest temperature in the environment, respectively. Therefore, the attack effect of various pixel values can be explored. When the pixel value of trigger is close to the median, the backdoor model has poor detection effect of clean test images.

Size Scaling Ratio (λ). We initially set the trigger size to

Method	Parameter		BAF (%)		ASR (%)
			person	car	
OAA	*Default*		−5.60	−3.40	97.87
	λ	$4\lambda_1$	−0.60	−0.90	97.98
		$3\lambda_1$	−1.80	−1.70	97.52
		$\mathbf{2\lambda_1}$	**−1.50**	−1.80	**97.05**
		$0.5\lambda_1$	−17.50	−8.20	97.85
		$0.25\lambda_1$	−36.00	−19.60	98.09
	rl	Mout	−11.30	−5.20	96.99
		M4	−6.60	−4.00	96.86
		M3	−3.20	−2.00	97.39
		M2	**−2.90**	**−1.70**	**97.46**
		M1	−6.70	−3.50	98.10
RAA	h	100	−1.10	−0.90	96.55
		80	**−0.90**	**−0.90**	**96.65**
		60	−3.20	−1.80	96.30
		40	−4.00	−1.80	94.99
		20	−5.50	−2.00	95.15
		10	−19.90	−6.80	91.46

Table 5. The effect of other parameters on OAA and RAA.

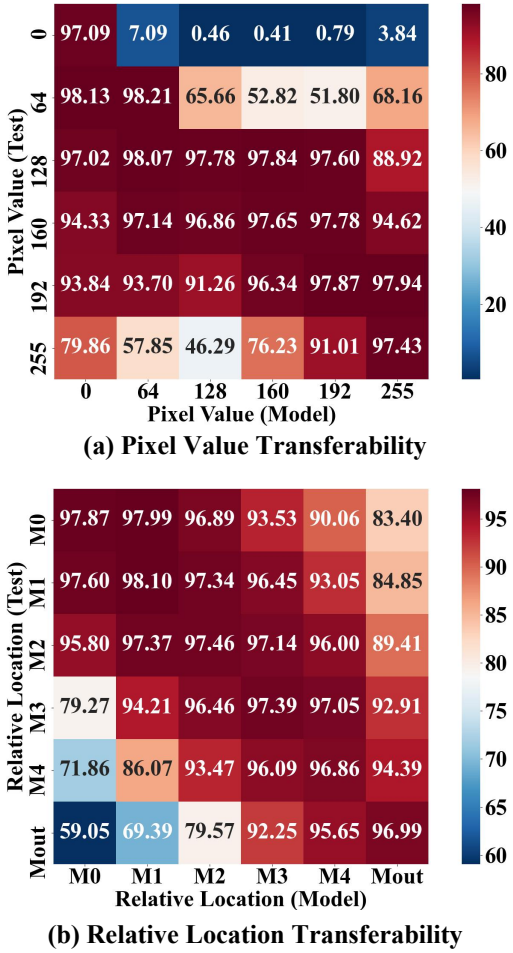
$\lambda = \lambda_1 = 0.04$. Different values of λ represent the scaling of initial trigger area. A lager trigger size will lead to more effective backdoor attacks, which is however a trade-off to the visual stealthiness. When the size scaling ratio is $0.25\lambda_1$, the BAF is low, which has a great impact on the detection effect of clean test images.

Relative Location (rl). We set the original location of the trigger to be rl = M0. Taking half of the diagonal of the trigger as a unit length, M1 means to move the trigger along the diagonal to the upper left corner by a unit length. The M2, M3 and M4 represent moving the trigger by 2, 3 and 4 unit lengths respectively. The Mout means to coincide the lower left vertex of the trigger with the upper left vertex of the object BBox. Adding a trigger near the middle part of the line connecting the center point and the upper left vertex of the object BBox can achieve a better attack effect.

B.2. Range-Affecting Attack

For RAA, we explore another parameter: trigger size. We set other parameters as $p = 192$, $q = 20\%$, ar = 150. The results are listed in Table 5.

Trigger Size (h). With the trigger width fixed at 6, we conduct experiments using various trigger heights. Shorter trigger heights result in fewer occupied detection grids by the trigger, posing challenges in establishing abnormal associations. Remarkably, a trigger height of 10 exacerbates the adverse effect of the backdoor model on the detection performance of clean test images.



(a) Pixel Value Transferability

(b) Relative Location Transferability

Figure 9. ASR (%) of pixel value transferability and relative location transferability.

B.3. Attack Transferability

Considering that the temperature may not be constant in the real world, we test the pixel value transferability. The setting the pixel values of trigger to 0, 64, 128, 160, 192, 255 respectively, we get six backdoor models. With the same trigger implanted in the test set, we get six test sets. Finally, the six backdoor models are used to test the six test sets respectively. Since the location of the trigger cannot be fixed precisely, we also focus on relative location transferability. Similar to pixel value transferability, we only change the relative location, and the rest of parameters are set with *default parameters*. The results are shown in the Figure 9. In practice, we can choose parameters with better transferability to achieve better attack effects.

B.4. Comparative Experiment

The triggers and backdoor attacks designed in [4] are used to attack TIOD, and the parameters such as poisoning ratio and attack setting are consistent with the *default parame-*

Attack Purpose →	OAA Misclassification			RAA Misclassification			OAA Disappearance			RAA Disappearance		
Method ↓	BAF (%)		ASR (%)	BAF (%)		ASR (%)	BAF (%)		ASR (%)	BAF (%)		ASR (%)
	person	car		person	car		person	car		person	car	
BadDet	−1.50	−1.80	29.72	−23.60	−51.00	12.65	−1.30	−0.90	45.35	−2.80	−54.50	1.47
Ours	−0.10	−0.90	97.09	−0.90	−0.90	97.44	−0.50	−5.00	98.54	+0.60	−0.60	95.66

Table 6. Comparative experiments with traditional methods applied to our TIOD backdoor attack task.

ters in our paper. We classify BadDet's methods according to the attack purposes and conduct comparative experiments between the BadDet's method and our method, focusing on similar attack purposes. When applying the BadDet's method, the ar in RAA is set to the global range of the image, which is consistent with the setting of BadDet. When applying our method, the ar in RAA is set to 150. For images with complex backgrounds and numerous small objects, the setting of the global range can significantly impact the accuracy of identifying clean samples. As shown in Table 6, it can be seen that VLOD backdoor attack methods are not suitable for the TIOD task. The reasons are twofold: 1) the RGB triggers cannot effectively activate the backdoor in the thermal infrared domain, and 2) the area that the trigger can affect is limited, as evident in the parameter ar experiment results presented in Table 2.

B.5. Temperature Modulated Triggering

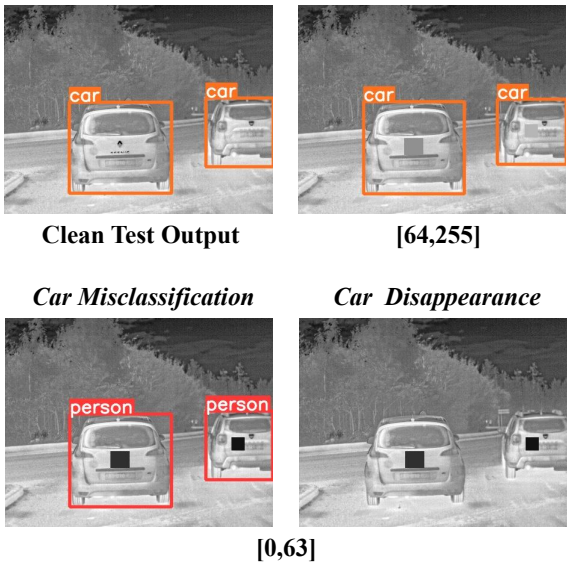


Figure 10. Examples of control the temperature in the digital world. The pixel range of the attack is $[0, 63]$.

For OAA, as shown in Figure 10, we conduct experiments using temperature-controlled backdoor attacks, taking advantage of the temperature-sensitive characteristics of TIOD. For RAA, as shown in Figure 11, We control the attack range using triggers' temperature.

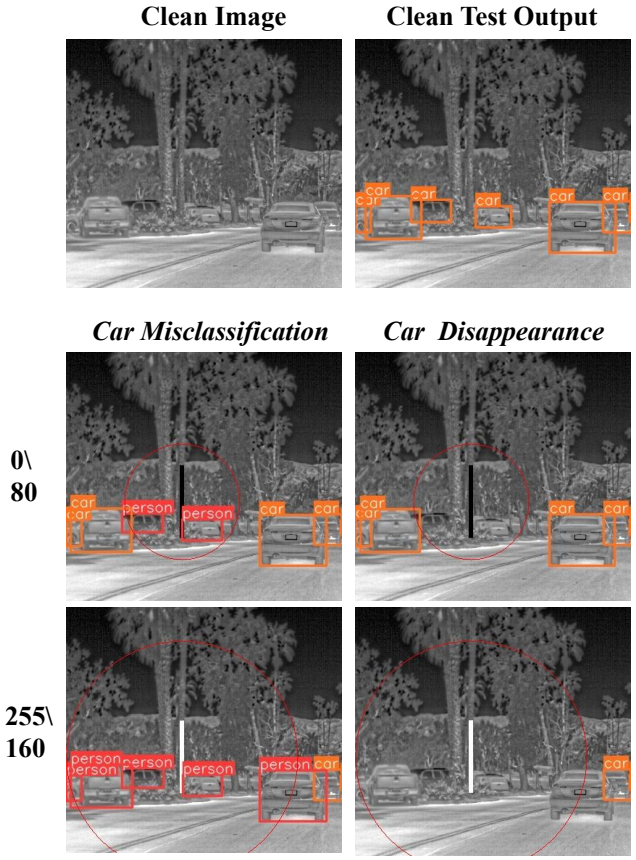


Figure 11. Examples of temperature control the attack range in the digital world. The settings of $p \setminus ar$ are listed on the far left. The affecting range of RAA is marked as the red circle.

C. Additional Experiment Results of Potential Countermeasures

Here, we provide additional experiment results of three potential countermeasures. In particular, we report our attack performance against these popular defense methods

Pruning and Fine-Pruning. We select one backdoor model randomly as the candidate backdoor model. Concretely, the candidate backdoor model is the backdoor model chosen randomly from the temperature control experiment, where the attack pixel range is [192, 255] and the attack purpose is *object disappearance*. This candidate backdoor model detects that the BA of person and car is 78.3% and 81.7% respectively, and the ASR is 95.41%. As

Pruned Network Layers	Ratio of Pruned Neurons	Pruning			Fine-Pruning		
		BA (%)		ASR (%)	BA (%)		ASR (%)
		person	car		person	car	
21-24	50%	77.30	80.90	95.41	19.30	25.20	9.24
	80%	77.40	80.70	95.49	23.00	33.10	22.11
	95%	77.40	79.90	95.49	23.20	30.10	12.37
19-24	50%	78.10	81.10	95.57	27.00	38.20	19.27
	80%	74.80	75.30	95.47	24.70	36.80	29.42
	95%	74.40	74.50	95.42	22.70	33.90	13.28
17-24	50%	**26.20**	**42.70**	**63.33**	23.70	34.80	65.44
	80%	10.00	14.90	0.00	21.30	34.50	19.65
	95%	1.50	0.10	0.00	24.90	29.40	14.14

Table 7. Evaluation results of **Pruning** and **Fine-Pruning**.

shown in Table 7, for **Pruning**, we prune the back-end network and gradually increase the number of pruned layers. Then, we change the pruning ratio to adjust the number of pruned neurons. For **Fine-Pruning**, we use the clean dataset to fine-tune the model obtained in **Pruning** for 20 rounds. In practice, the poisoned dataset and training parameters are not available, so the implementation of these two defense methods will be more difficult.

Neural Cleanse. We randomly select two backdoor models as the objects of the defense test. For OAA, we choose the backdoor model trained with parameters $p = 255$, $q = 20\%$, $\lambda = 0.04$, and $rl =$ M0. For RAA, we choose the backdoor model trained with parameters $p = 192$, $q = 20\%$, $ar = 150$, and $h = 100$. We show the detection results for the attack purpose to misclassify car as person in Figure 12. Since the attacked label is car, we list the results about the car. The recovered triggers for the remaining

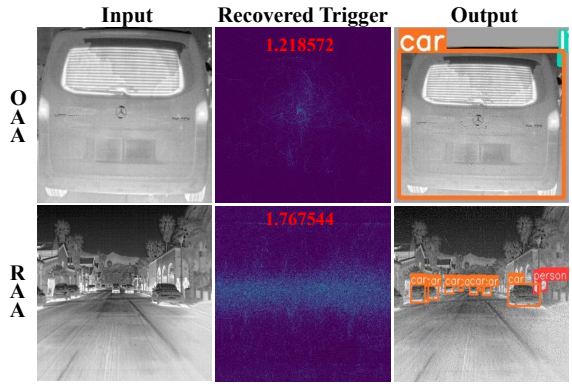
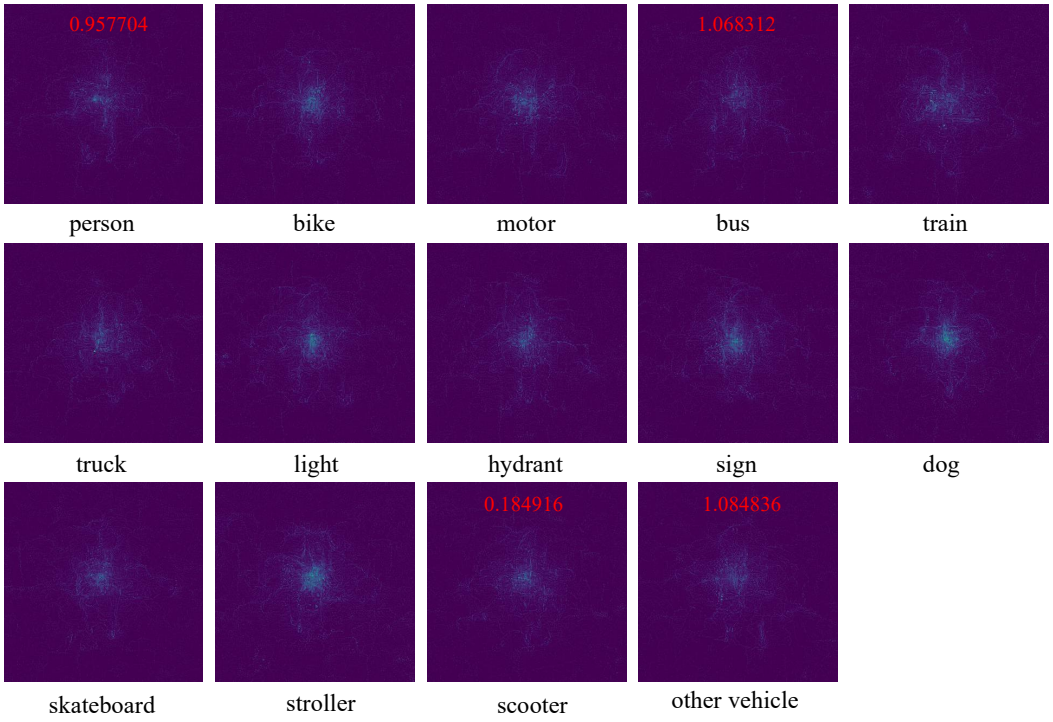
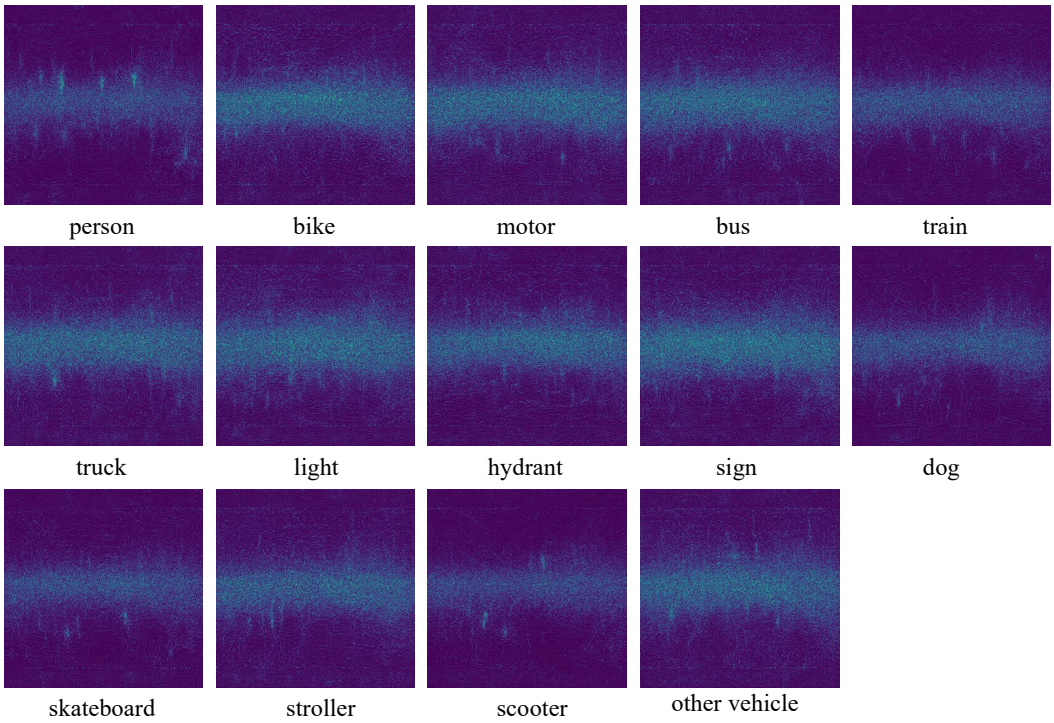


Figure 12. Evaluation results of NC. The Input is clean images fed to NC. The red characters are the anomaly indices (value > 2 considered as trigger detected) detected by NC for the attacked label. The Output is detection results of the backdoor model on images with recovered triggers injected.

classes are shown in Figure 13. For classes with anomaly indices, we mark the anomaly indices with red characters in the image. The results show that the **Neural Cleanse** adapted to image classification cannot provide satisfactory defense against our proposed backdoor attack methods.



(a) OAA



(b) RAA

Figure 13. The recovered triggers for remaining classes.