APPENDIX A
EXPERIMENT SETTING DETAILS

A. The VERIWild dataset and data preprocess

Our experiments are based on the VERIWild dataset, which contains 416,314 images of 40,671 vehicles' identities for training. The testing dataset consists of three sub-test datasets, the small, medium, and large test datasets, which contain 3,000, 5,000, and 10,000 vehicle identification, and 38,861, 64,389, 128,517 images respectively. We evaluate the prediction and defense performance on the three testing datasets.

Figure 6 illustrates the vehicle re-identification framework used for training and inference. We first pre-process the input data by resizing and random flipping. Then we deploy ResNet-18 [20] as the backbone. ResNet-18 includes 4 residual blocks and 17 convolution layers. The features are aggregated via average pooling. We choose a linear layer as the prediction head. In the training phase, we include triplet loss and cross-entropy loss in the loss function and train the target model. In the inference phase, given two input images, we calculate the cosine similarity between two predicted embeddings and make predictions.

B. Experiments hyper-parameter settings

We first train the target model for 90 epochs using a learning rate of 0.0003. Then we perform the proposed defense. In the proposed defense, we retrain the model for 90 epochs with a learning rate 0.0003 for the model and 10^{-6} for the soft mask. We set batch size as 512 and the threshold $\theta = 0$ for pruning. We set hyper-parameters $\beta = 0.0004$, $\lambda_1 = 1$, $\lambda_2 = 10$, $\alpha_1 = 5$, $\alpha_2 = 0.2$, $\alpha_3 = 0.01$, $\alpha_4 = 0.005$.

C. Defense baseline details

The three defense baseline details are shown below:

- **Noise Defense**. We add Gaussian noise to the input data to obstruct the adversary's performance. We generate Gaussian noises using different standard deviations 0.05, 0.10, 0.20, 0.30, 0.40, 0.50 and zero-mean to explore a better balance between utility and privacy.
- **Dropout Defense**. We randomly drop out the value for the intermediate output to improve the user's privacy. We adopt different dropout ratios (0.05, 0.10, 0.20, 0.30, 0.40, 0.50) in the defense.
- **Skip Defense**. We randomly skip some values on convolution layers in the neural network. We set the skip ratio to 0.05, 0.10, 0.20, 0.30, 0.40, and 0.50, which represent how many values in convolution layers shall be skipped in the network.

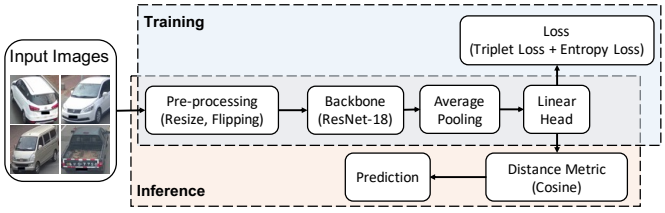

Fig. 6: Vehicle re-identification framework used in the experiments.

Dataset	Defenses	Attack Accuracy
VERIWild	No Defense	27.00%
	APP [7]	0.00%
	Noise	0.00%
	Dropout	0.00%
	Skip connection	0.00%
	PATROL	0.00%
VERI	No Defense	14.50%
	APP [7]	0.00%
	Noise	0.00%
	Dropout	0.00%
	Skip connection	0.00%
	PATROL	0.00%

TABLE VII: The attack accuracy for the classification model on no defense and defend models. The results of the VERIWild dataset are the average attack accuracy on the VERIWild small, medium, and large datasets. We confirm the attack accuracy metric is not meaningful for every model with defense.

APPENDIX B
ADDITIONAL EXPERIMENT RESULTS FOR PATROL COMPARED WITH BASELINE DEFENSE AND ABLATION STUDY

In this section, we show more details about the experiments of PATROL with different hyper-parameters settings. We give both the decreasing ratio compared with no compression, no defending model, and the real model's prediction accuracy, PSNR, and SSIM values.

The PATROL method comparing to adding noise, dropout, and skip connection defense methods on three VERIWild testing datasets and VERI testing dataset shown in Figure 3 in the main text.

The different curves show the performance of different defenses. The original model without defenses is presented in a blue dashed line. The PSNR and SSIM value for the baseline is 17.22 and 0.42 for the VERIWild dataset and 21.85, 0.63 for the VERI dataset. We selected different hyper-parameter settings (different values for β, λ_1, λ_2, α_1, α_2, α_3, and α_4.) to explore the trade-off between privacy and utility. The points in the curves represent the different hyper-parameter settings of PATROL model or different dropout, noise adding, and skipping ratios for the defense baselines.

A. Effectiveness of Pruned Model Structures.

Effectiveness of Pruned Model Structures. We consider another structure pruning method, block-wise pruning, where the entire convolution block can be removed from the target network. In our experiment, we add the soft mask at the end of each basic convolution block of ResNet-18 to implement block-wise pruning. Table VIII demonstrates that the channel-wise pruning method yields higher prediction accuracy and better defense performance compared to the

Pruning Method	Prediction Acc. Drop	PSNR Drop	SSIM Drop
Channel-wise	**3.1%**	**11.9%**	**10.9%**
Block-wise	10.5%	9.2%	9.5%
Dropout defense	10.4%	3.9%	7.1%

TABLE VIII: Comparison of PATROL using channel-wise and block-wise pruning. Channel-wise pruning achieves better defense performance and higher prediction accuracy. Both pruning methods in PATROL outperform dropout defense (the best defense baseline).

block-wise pruning method. This is mainly due to the trainable masks. The block-wise only has 8 trainable masks which is hard to balance the trade-off between prediction accuracy and defense performance after pruning. Despite its limitations, block-wise pruning has demonstrated some advantages over existing defenses. In light of the strong defense performance of the dropout defense among the existing defenses, we have included it in the table for comparison.

B. The experiments results for classification base metrics of different defense methods

In table VII, we present the results of the prediction values for the classification network on different defense methods. If there is no defense for the target model, the attack accuracy (the accuracy of the classification model) reaches 26.00% for VERIWild and 14.50% for the VERI dataset. After we apply the defense methods, the attack accuracy for every defense method is 0%, which makes the comparison very difficult. Hence, we do not provide the attack accuracy of the classification model in the comparison of PATROL and baselines.

C. More observation of Pruning Effectiveness

The reconstructing images' PNSR and SSIM from the original model without defense in the first scenario is 18.15 and 0.45, and in the second scenario is 17.16 and 0.42.

An interesting thing is when the edge device deploys more layers for the edge-side model, the privacy protection effectiveness of the pruning method decreases. In the Small Edge Device Scenario, PATROL with a low pruning ratio, deploying one more residual block than the original model on the edge device, can reduce the PNSR and SSIM value by 0.84 and 0.04 when excluding the effect of the adversarial reconstruction training and Lipschitz regularization (Compared to the defense result between PATROL with a low pruning ratio and the model with Adversarial and Lipschitz defense only). However, in the Large Edge Device Scenario, when excluding the effect of adversarial reconstruction training and Lipschitz regularization and deploying one more residual block deploying on the edge device, PATROL with a low pruning ratio could only reduce the PNSR and SSIM by 0.56 and 0.01, the same observation also appears in every comparison model. The observation shows that the privacy protection of pruning method can reduce more PSNR and SSIM when fewer layers are on the edge-side model before pruning, which indicates the defensive pruning method is more effective on an edge device with small memory.