Steganographic Passport: An Owner and User Verifiable Credential for Deep Model IP Protection Without Retraining

Supplementary Material

S1. Architecture of the key-based invertible steganographic network

The key-based invertible steganographic network (ISN) consists of eight unit blocks $\{\mathcal{B}_1, \mathcal{B}_2, ..., \mathcal{B}_8\}$, each of which shares the same network architecture but learns with different weights. In each unit block, there are three sub-blocks $\{\aleph_1, \aleph_2, \aleph_3\}$, each also sharing the same architecture with different weights. Each of $\{\aleph_1, \aleph_2, \aleph_3\}$ is a typical Dense Block, where each layer receives feature maps from all preceding layers. Taking $\aleph_1$ as the representative, it consists of 5 convolutional layers $\{C_1, C_2, ..., C_5\}$. C_1 takes an input with 12 channels and outputs with 32 channels, with a kernel size of 3×3, stride of 1, and padding of 1. C_2 to C_4 incrementally increase the input channels by concatenating the output of the previous layers. For instance, we fed C_2 with $(32 + 32)$ channels, while C_3 with $(32 + 2 \times 32)$, and so on. C_5 outputs with 12 channels, consolidating the concatenated features of $\{C_1, C_2, ..., C_4\}$. A LeakyReLU is applied after each convolutional layer, except C_5. Given an input image with the shape of $\{3, H, W\}$, we will perform discrete wavelet transform (DWT) on it and then feed the transformed image, with the shape of $\{12, H/2, W/2\}$, to the key-based ISN. For clarity, we define the notations used to train the key-based ISN in Tab. A1.

With x_c and x_s as the input, the forward hiding pass $\mathcal{H}_s$ of the key-based ISN can be formulated as follows:

$$x_c{}^{(k+1)} = x_c^{(k)} + \aleph_1\left(x_e^{(k)}\right), \tag{20}$$

$$x_e^{(k+1)} = x_e^{(k)} \odot \exp\left(\lambda \cdot \aleph_2\left(x_c^{(k+1)}\right)\right) + \aleph_3\left(x_c^{(k+1)}\right), \tag{21}$$

where $x_c^{(1)} = x_c$, $x_e^{(1)} = x_e$, k denotes the k-th block, and λ controls the weight of the exponential operation. We take $x_s = x_c^{(8)}$ and $z = x_e^{(8)}$ as the outputs of the forward hiding calculation, where z is the lost information and is discarded.

By retaining the weights of all the modules from the forward pass, and given the key image $\mathbf{k}_s$, the reverse pass of key-based ISN can be formulated as follows:

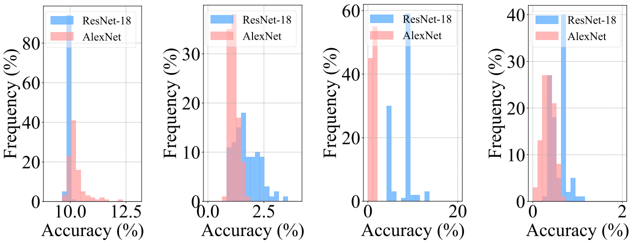
$$\hat{z}^{(k-1)} = \left(\hat{z}^{(k)} - \aleph_3\left(x_s{}^{(k)}\right)\right) \oslash \exp\left(\lambda \cdot \aleph_2\left(x_s^{(k)}\right)\right), \tag{22}$$

$$x_s^{(k-1)} = x_s^{(k)} - \aleph_1\left(\hat{z}^{(k-1)}\right), \tag{23}$$

where $x_s^{(8)} = x_s$, $\hat{z}^{(8)} = \mathbf{k}_s$, $\oslash$ denotes element-wise division operation. We take $\widehat{x_c} = x_s^{(1)}$ and $\widehat{x_e} = \hat{z}^{(1)}$ as the output of the reverse pass. By conducting inverse discrete wavelet transform (IDWT) on $\widehat{x_e}$, we can obtain the revealed hidden image in the spatial domain.

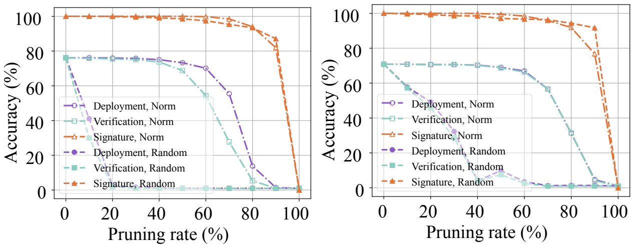
Table A1. Notations and Representations.

Notation	Representation
x_c	Cover images (randomly sampled owner-side passport images)
x_s	Stego images (generated user-side passport images)
x_e	Secret images (randomly sampled user's ID images)
$\widehat{x_e}$	Revealed secret images (revealed user's ID images)
$\widehat{x_c}$	Revealed cover images (revealed owner-side passport images)



(a) CIFAR-10 (b) CIFAR-100 (c) Caltech-101 (d) Caltech-256

Figure A1. Performance of our approach under random passport attacks across various datasets, with models incorporating BN.



(a) CIFAR-100 with BN (b) CIFAR-100 with GN

Figure A2. Performance of our method under random and ℓ_1 norm pruning attacks.

S2. Supplementary experimental results

Ownership ambiguity attack robustness by trials and errors.

In the existing passport scheme, the most direct way to maliciously claim ownership is by trying randomly selected passports. As the strict avalanche criterion of the SHA hash function ensures that the forged signature will be different from the genuine signature in approximately 50% of bits, the random passport ambiguity attack can never be successful for our method.

To simulate this attack, we randomly select 100 images from the same distribution as the genuine passport to use as forged owner-side passports. From the frequency distributions shown in Fig. A1, no random passports can gain successful ownership verification over the four datasets and the two networks. Specifically, for ResNet-18 on the Caltech-101 dataset, we observed a peak accuracy of 13.17%. However, this is significantly below that task's $\tau_f = 73.56\%$, and the AD of 58.81% has far exceeded $\tau_d = 0.05\%$.

Removal attack robustness of pruning with more different settings. Fig. A2 illustrates the performance trends of our method under random and ℓ_1 norm pruning attacks on ResNet-18 with more different settings. The same phenomenon as the main experiment is observed. That is, the SA does not degrade much until the model performance has dropped dramatically. This indicates that our method is highly resilient against pruning attacks.