S1 Settings

We make use of three datasets, namely CIFAR-10, BOSS1.0.1, and ImageNet for the evaluation. CIFAR-10 and ImageNet are widely used for various vision-related tasks and BOSS1.0.1 is the dataset commonly covered by the stenography fields. For the overall experiments, for CIFAR-10, we utilize its training dataset for training and the testing dataset for testing. In the case of BOSS1.0.1, it contains 10,000 images in total. We split the dataset into two: 9,000 for training and 1,000 for testing. In the case of ImageNet, for all experiments including PixelCNN++ training, we utilize its training dataset for training and its validation dataset for testing. We use $32 \times 32 \times 3$ for CIFAR-10, $128 \times 128 \times 1$ for Boss-1.0.1 dataset, and $64 \times 64 \times 3$ for ImageNet. For experimental settings, we spend approximately 1 week to train a PixelCNN++ model for each dataset using four NVIDIA P100 GPUs. For training Deep Steganography and ISGAN, we need less than 3h for each case on one NVIDIA GTX 1080 GPU. In the case of Deep Steganography and ISGAN, we train each model using the hyperparameter proposed in each paper.

S2 Where is a Secret Image Encoded?

We further examine how and where the secret image is encoded in case of the deep learning based steganographic algorithms as mentioned in Sec. 1. We could utilize the newly acquired discoveries to find a more appropriate way to destroy as much of the secret image as possible while maintaining the quality of the cover image.

As illustrated in Fig. S1, after we increase the pixel value of the R channel of a random single pixel in the stego image by 1, we evaluate its effect on the decoded secret image by measuring the residual between the secret image decoded by unmodified stego image and randomly modified stego image, respectively. An increase by 1 in any position in any color channel of the stego image has an impact across all color channels in the decoded secret image. We see that the largest impact commonly occurs on the same channel.

We also witness that the secret image is encoded in a distributive but location-limited way. In other words, the pixel information of the left top corner of the secret image is only encoded on the left top corner of the cover image and likewise for other areas. Based on this finding, we confirm that active steganalysis algorithms adjusting the stego image generated by the deep learning-based encoding algorithms should apply the alternation of all the pixels of the stego image for the definitive steganography removal.

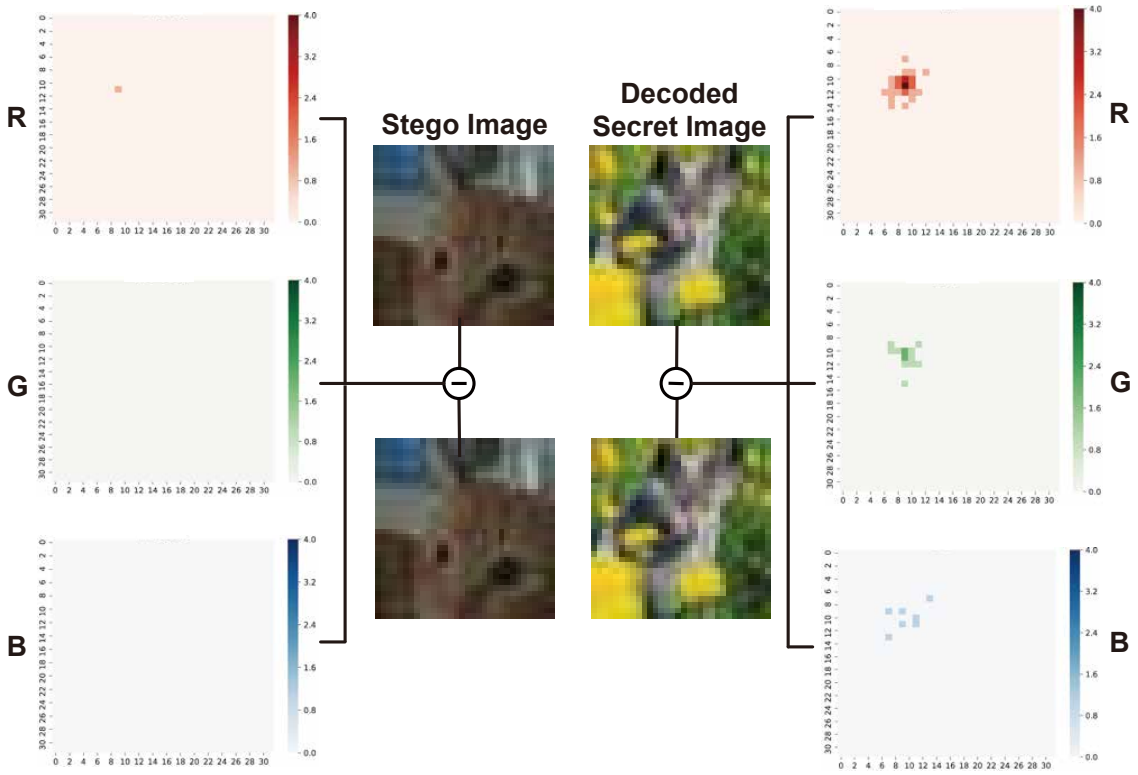

Figure S1: Impact of increasing a pixel value of the stego image by 1 on the decoded secret image. The left side of the figure is the residual on each RGB channel between the stego image and modified stego image. The right side of the figure is the residual on each RGB channel between the secret images decoded by the stego image and modified stego image, respectively.

S3 Destruction Rate

Table S1: The average of decoded rate (DC) and destruction rate (DT) of **Deep Steganography** on each dataset.

		CIFAR-10		IMAGENET		BOSS1.0.1	
	ϵ	DC	DT	DC	DT	DC	DT
Original		0.9117	0.0000	0.9657	0.0000	0.9634	0.0000
Gaussian Noise	1	0.7748	0.2138	0.9457	0.0447	0.9018	0.0917
	2	0.7592	0.2314	0.9104	0.0807	0.8460	0.1470
	4	0.7358	0.2569	0.8550	0.1360	0.7789	0.2133
	8	0.7044	0.2899	0.7975	0.1932	0.7574	0.2339
Denoising		0.6881	0.2973	0.7338	0.2649	0.7273	0.2639
Restoration		0.7809	0.2076	0.9645	0.0278	0.9406	0.0528
Our Method	1	0.7520	0.2234	0.9216	0.0708	0.8633	0.1301
	2	0.7112	0.2682	0.869	0.1390	0.8027	0.1907
	4	0.6790	0.3064	0.8199	0.1899	0.7671	0.2263
	8	0.6631	0.3223	0.7607	0.2480	0.7662	0.2259

Table S2: The average of decoded rate (DC) and destruction rate (DT) of **ISGAN** on each dataset.

		CIFAR-10		IMAGENET		BOSS1.0.1	
	ϵ	DC	DT	DC	DT	DC	DT
Original		0.9087	0.0000	0.9264	0.0000	0.9297	0.0000
Gaussian Noise	1	0.9069	0.0278	0.9164	0.0551	0.8743	0.0984
	2	0.8804	0.0547	0.8634	0.1082	0.8540	0.1187
	4	0.8212	0.1141	0.7875	0.1840	0.7827	0.1826
	8	0.7774	0.1580	0.7406	0.2310	0.7777	0.2113
Denoising		0.6977	0.1541	0.6907	0.2808	0.7126	0.2763
Restoration		0.9089	0.0042	0.9249	0.0466	0.9287	0.0439
Our Method	1	0.9035	0.0311	0.8957	0.0759	0.8679	0.1047
	2	0.8751	0.0672	0.8494	0.1225	0.8061	0.1665
	4	0.8086	0.1342	0.7889	0.1825	0.7521	0.2131
	8	0.7982	0.1453	0.6588	0.3127	0.7029	0.2860

Table S3: The average of decoded rate (DC) and destruction rate (DT) of **LSB** on each dataset.

		CIFAR-10		IMAGENET	
	ϵ	DC	DT	DC	DT
Original		1.0000	0.0000	1.0000	0.0000
Gaussian Noise	1	0.8588	0.1411	0.8588	0.1411
	2	0.7163	0.2836	0.7163	0.2836
	4	0.6705	0.3294	0.6705	0.3294
	8	0.6632	0.3367	0.6666	0.3333
Denoising		0.7160	0.2839	0.6941	0.3058
Restoration		0.7774	0.2225	0.6929	0.3071
Our Method	1	0.7204	0.2795	0.7121	0.2878
	2	0.6642	0.3357	0.6702	0.3297
	4	0.6643	0.3356	0.6740	0.3259
	8	0.6601	0.3398	0.6665	0.3334

S4 Experimental Results on Original Images

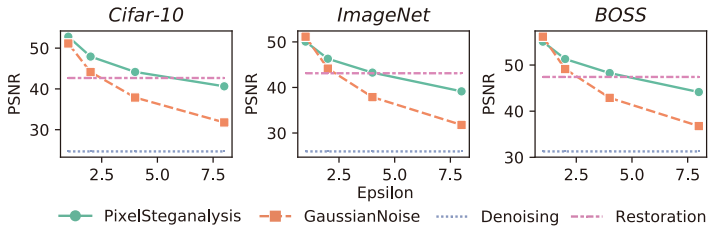

Figure S2: The degree of degradation on the original images after four active steganalysis methods. We can see that the quality of the modified images after our method is always better than the other three methods: Gaussian noise, Denoising, and Restoration. In case of restoration, the PSNR of restoration is higher than that of our method when ϵ is 8. However, the removal ability of restoration is too weak as shown in Fig. S18.

S5 Experimental Results on ImageNet

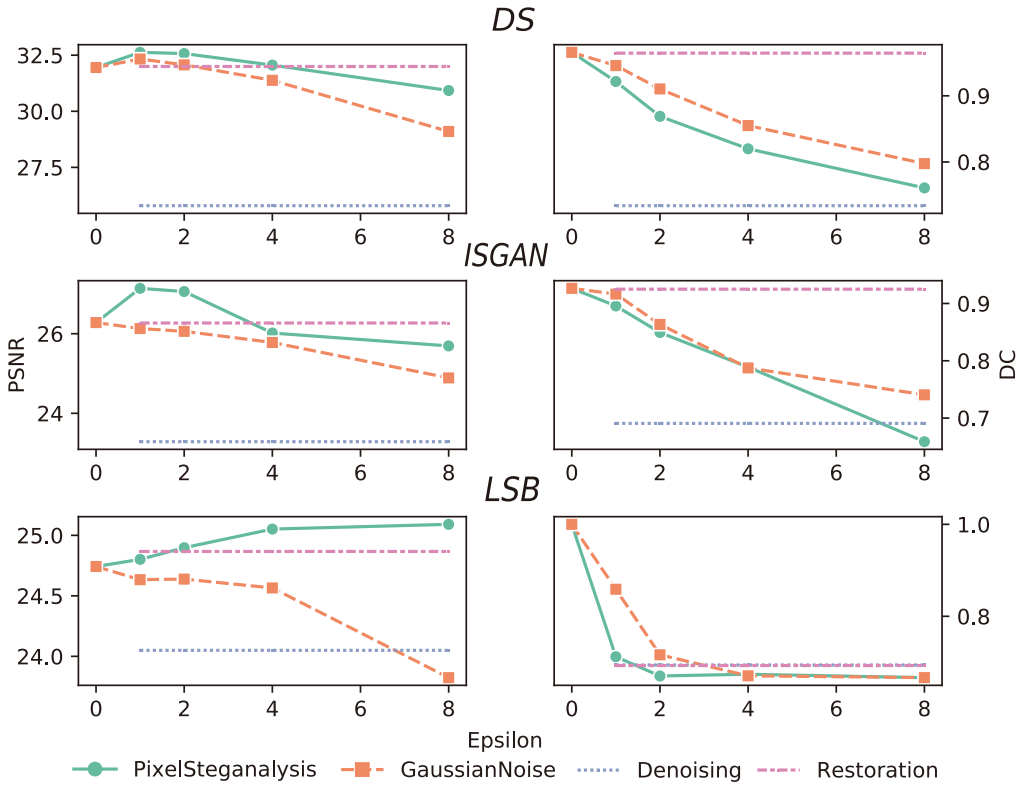

Figure S3: The comparison of our method with the three comparing methods when using the ImageNet dataset. The overall trends of the PSNR and the DC are very similar to CIFAR-10, except for one case. On ImageNet, the average PSNR value keeps increasing even in ϵ = 8. It can be interpreted that our attempt to move the distribution of the stego image towards the distribution of the original cover image quite works well.

S6 Experimental Results on BOSS1.0.1

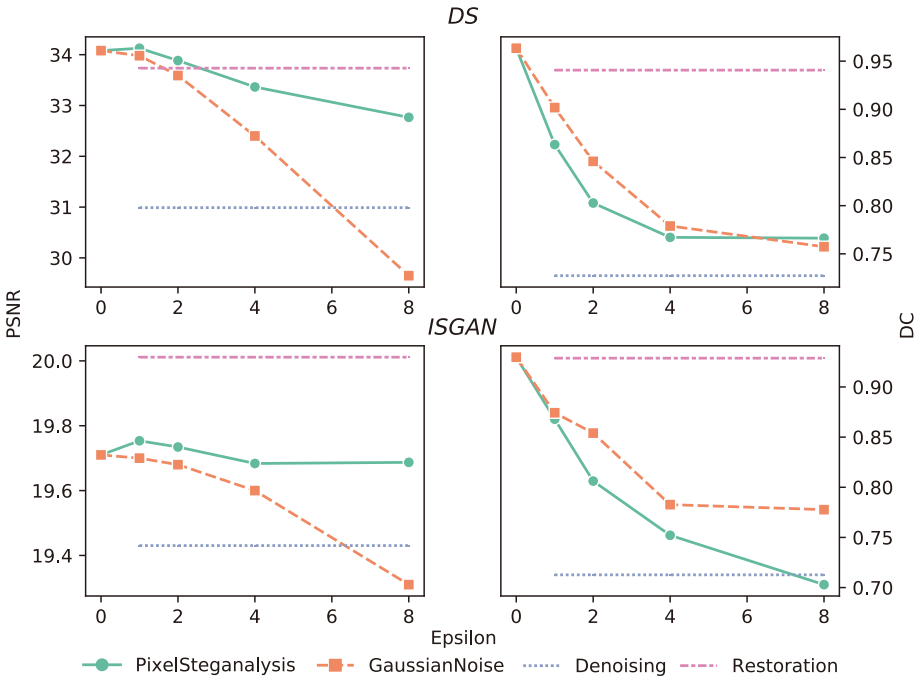

Figure S4: The comparison of our method with the three comparing methods when using the BOSS1.0.1 dataset. In case of ISGAN, we can observe that the PSNR of Gaussian noise falls rapidly as ϵ value increases.

S7 Comparison between an Edge Detector and Our Deep Neural Network

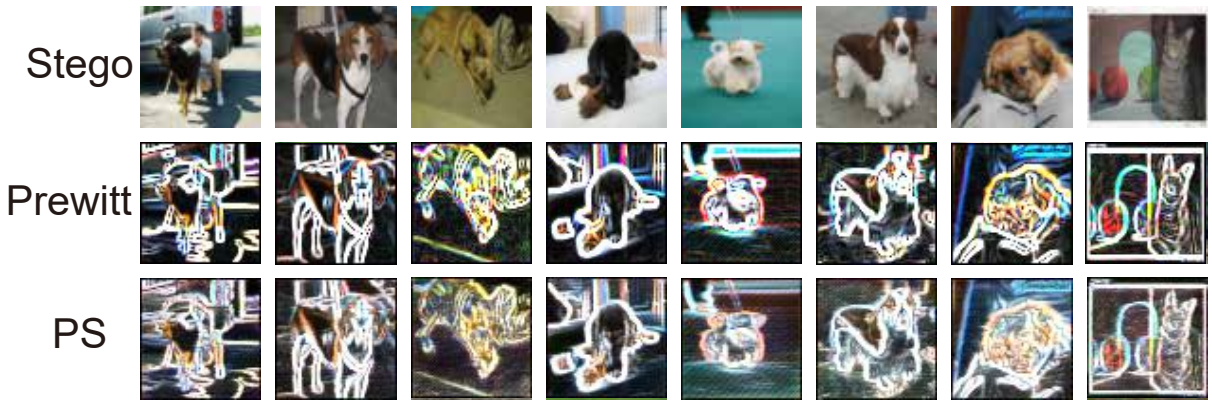

Figure S5: The comparison of our extracted edge distribution with a famous edge detector, Prewitt. Prewitt = an edge distribution of a stego image from the well-known edge detector, Prewitt. PS = an edge distribution of a stego image from our trained deep neural network. Each pair of images is alike. The good results using the extracted edge distribution show that the edge distribution is well-used for the original purpose.

S8 Residual between a Cover Image and a Stego Image

Fig. S6 shows the location where the secret image is more encoded. The residual between the cover image and stego image is much larger on the edge areas while the residual values are comparably small on the non-edge (low frequency) areas.

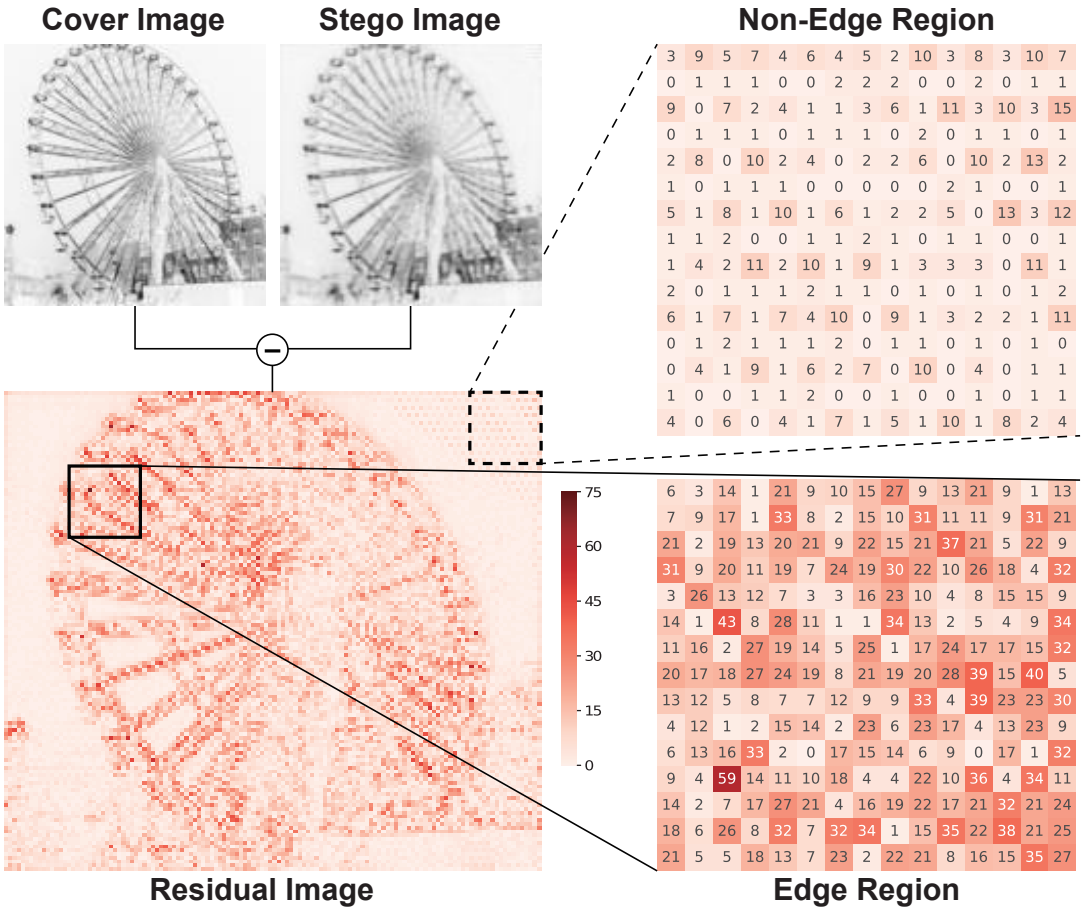

Figure S6: Residual image between the cover and stego images generated via Deep Steganography of the BOSS1.0.1 dataset. There are two zoomed areas showing different characteristics. In the case of non-edge regions, the residual values are less than 15 and commonly 1 or 2. On the other hand, the residual values on edge regions are very large.

S9 Denoising and Restoration

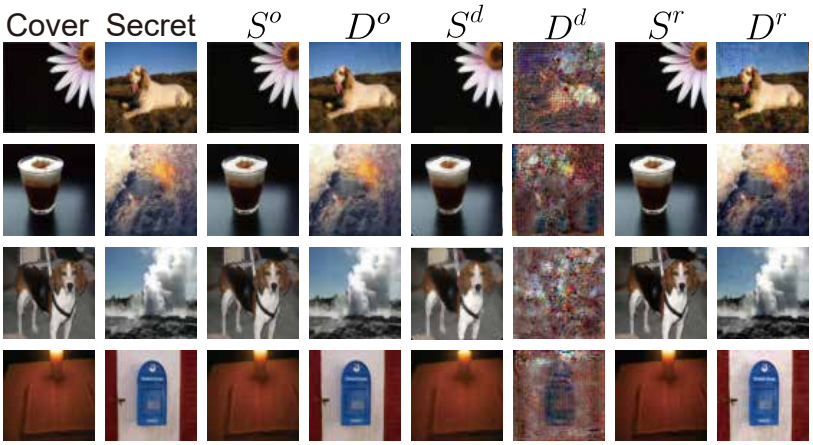

Figure S7: The effect of denoising and restoration **(Deep Steganography on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^d = stego image modified by median filter, D^d = secret image decoded from S^d, S^r = stego image modified by wiener restoration, and D^r = secret image decoded from S^r

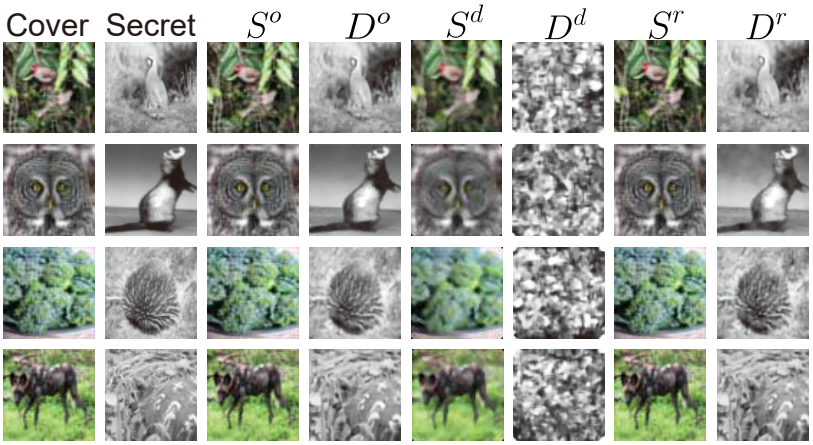

Figure S8: The effect of denoising and restoration **(ISGAN on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^d = stego image modified by median filter, D^d = secret image decoded from S^d, S^r = stego image modified by wiener restoration, and D^r = secret image decoded from S^r

S10 Ablation Study Examples

We provide a perceptual sample out of the edge condition examples of the ablation study in Table S4 in Fig. S9.

Table S4: Ablation results of our proposed methods. DC = decoded rate, and DT = destruction rate. In the case of PSNR, SSIM and DA, the larger the value, the better. DC is the opposite.

	ϵ	PSNR	SSIM	DC	DT
Our Method w/o Edge Detection	1	35.66	0.9834	0.7761	0.2117
	2	35.72	0.9837	0.7523	0.2365
	4	35.39	0.9811	0.7375	0.2720
Our Method	1	35.89	0.9839	0.7691	0.2184
	2	35.85	0.9842	0.7258	0.2626
	4	35.67	0.9822	0.6923	0.3001

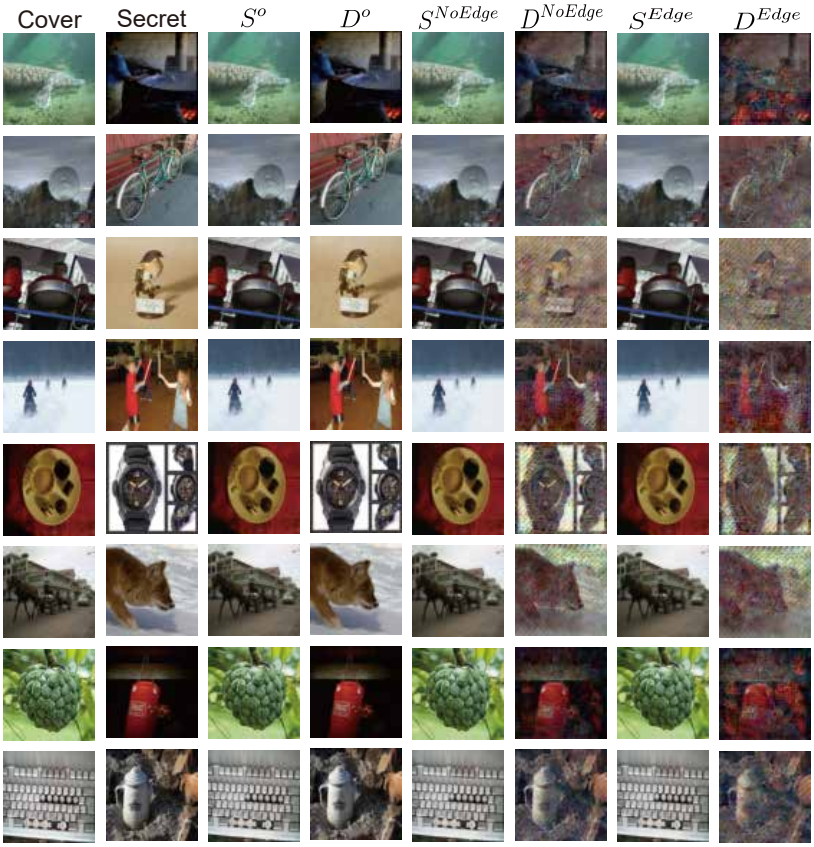

Figure S9: Ablation study: no edge detection examples (**Deep Steganography on ImageNet**). S^o = stego image, D^o = decoded secret image, S^{NoEdge} = stego image modified by our method but without an edge condition, D^{NoEdge} = secret image decoded from S^{NoEdge}, S^{Edge} = stego image modified by our method with an edge condition, and D^{Edge} = secret image decoded from S^{Edge}

S11 Comparison at the Same Decoded Rate

We provide a random sample of the examples of how our method and the conventional method, Gaussian noise, differ in the efficiency at the same decoded rate in Fig. S10 and S11 as described in Sec. 5.1.

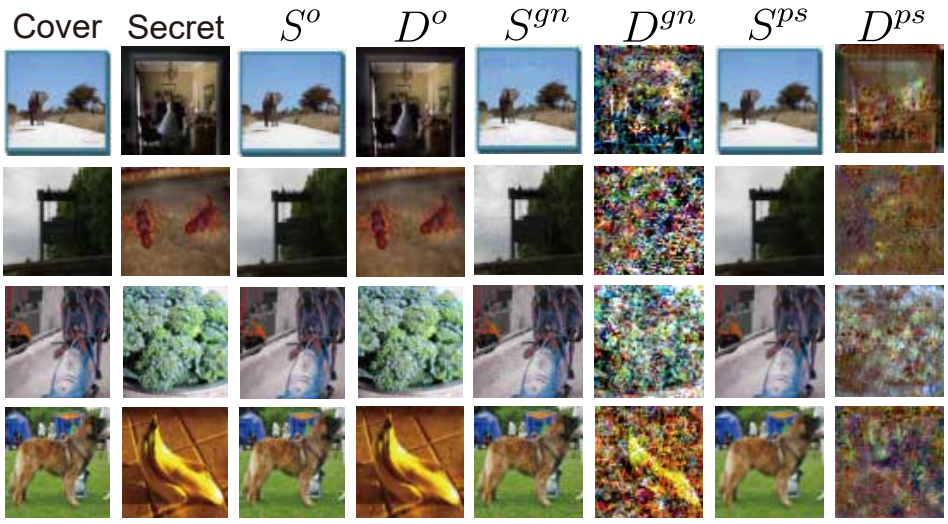

Figure S10: Comparison of the image quality of the stego image modified by our method with that of the stego image modified by Gaussian noise when the decoded rate between the two secret images is the almost same **(Deep Steganography on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

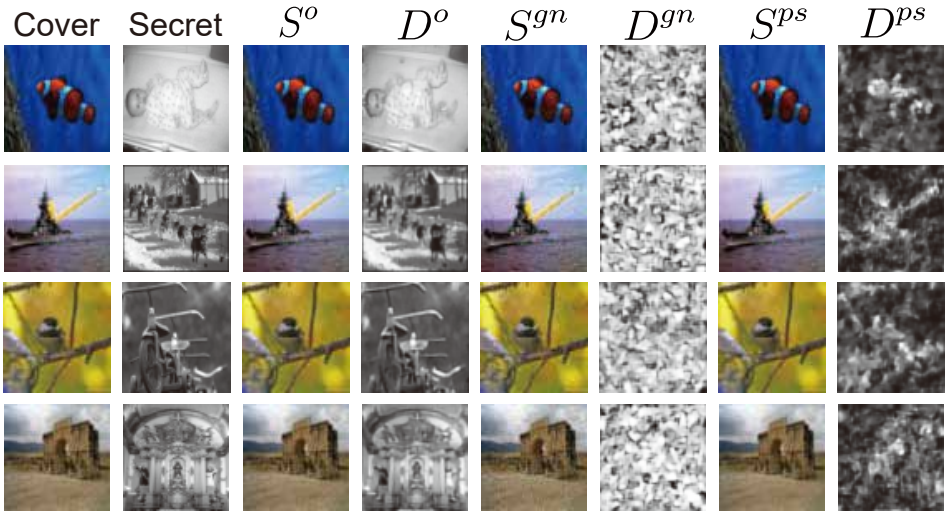

Figure S11: Comparison of the image quality of the stego image modified by our method with that of the stego image modified by Gaussian noise when the decoded rate between the two secret images is the almost same **(ISGAN on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

S12 Comparison at the Same PSNR

In Fig. S5, S8, S7, S8, S9, and S10, we provide a random sample of the examples of how our method and the conventional method, Gaussian noise, differ in the efficiency at the same PSNR for all the given cases as described in Sec. 5.1 and Fig. 4.

Figure S12: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(Deep Steganography on CIFAR-10)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

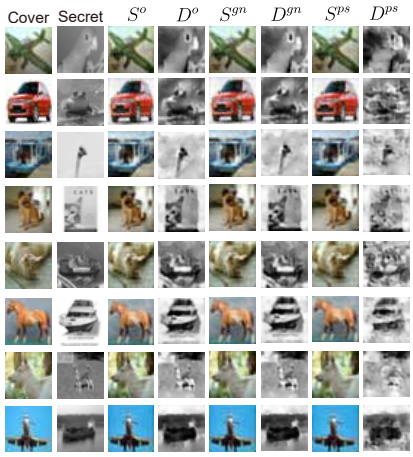

Figure S13: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(ISGAN on CIFAR-10)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

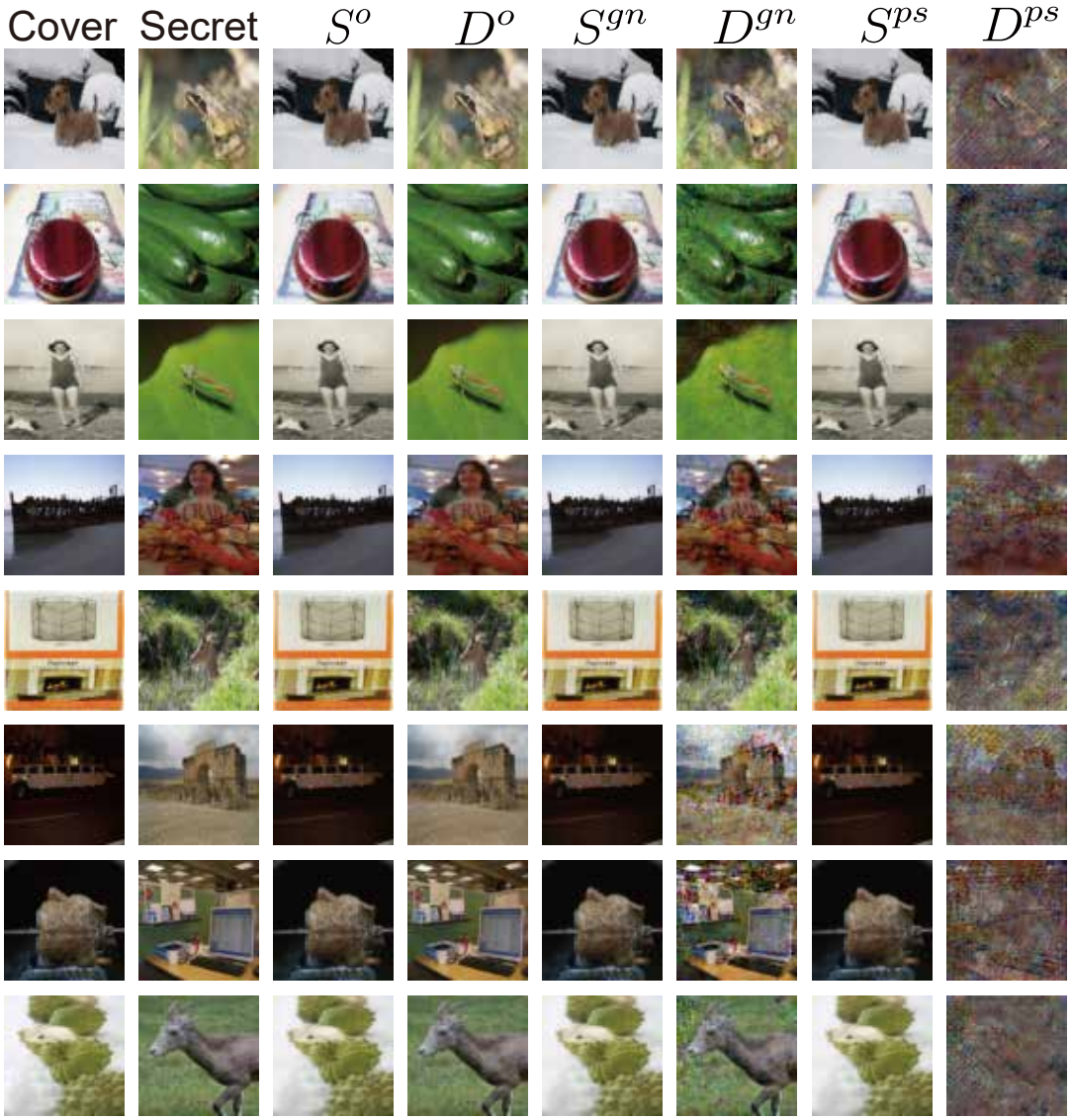

Figure S14: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(Deep Steganography on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

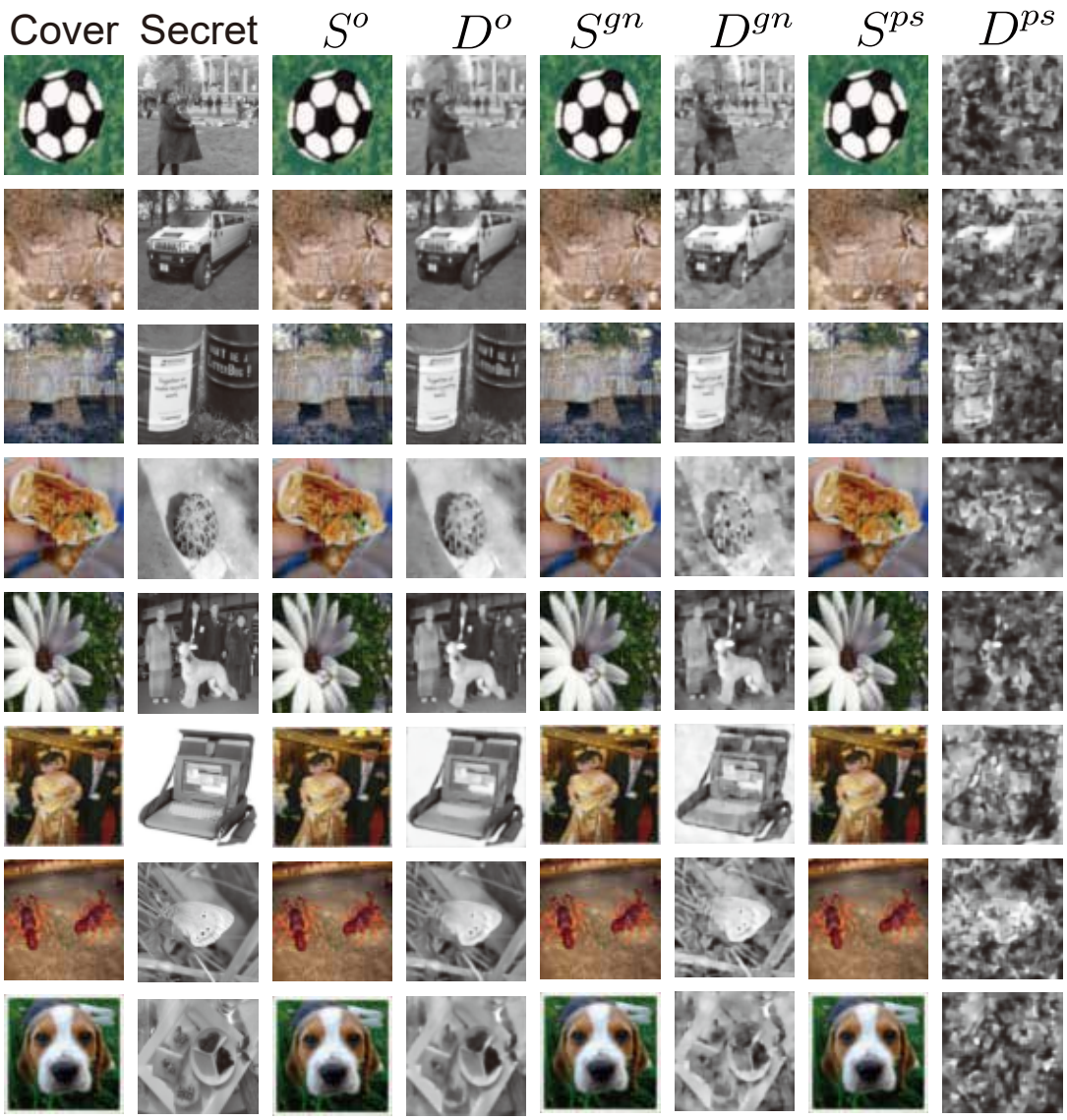

Figure S15: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(ISGAN on ImageNet)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

Figure S16: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(Deep Steganography on BOSS1.0.1)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

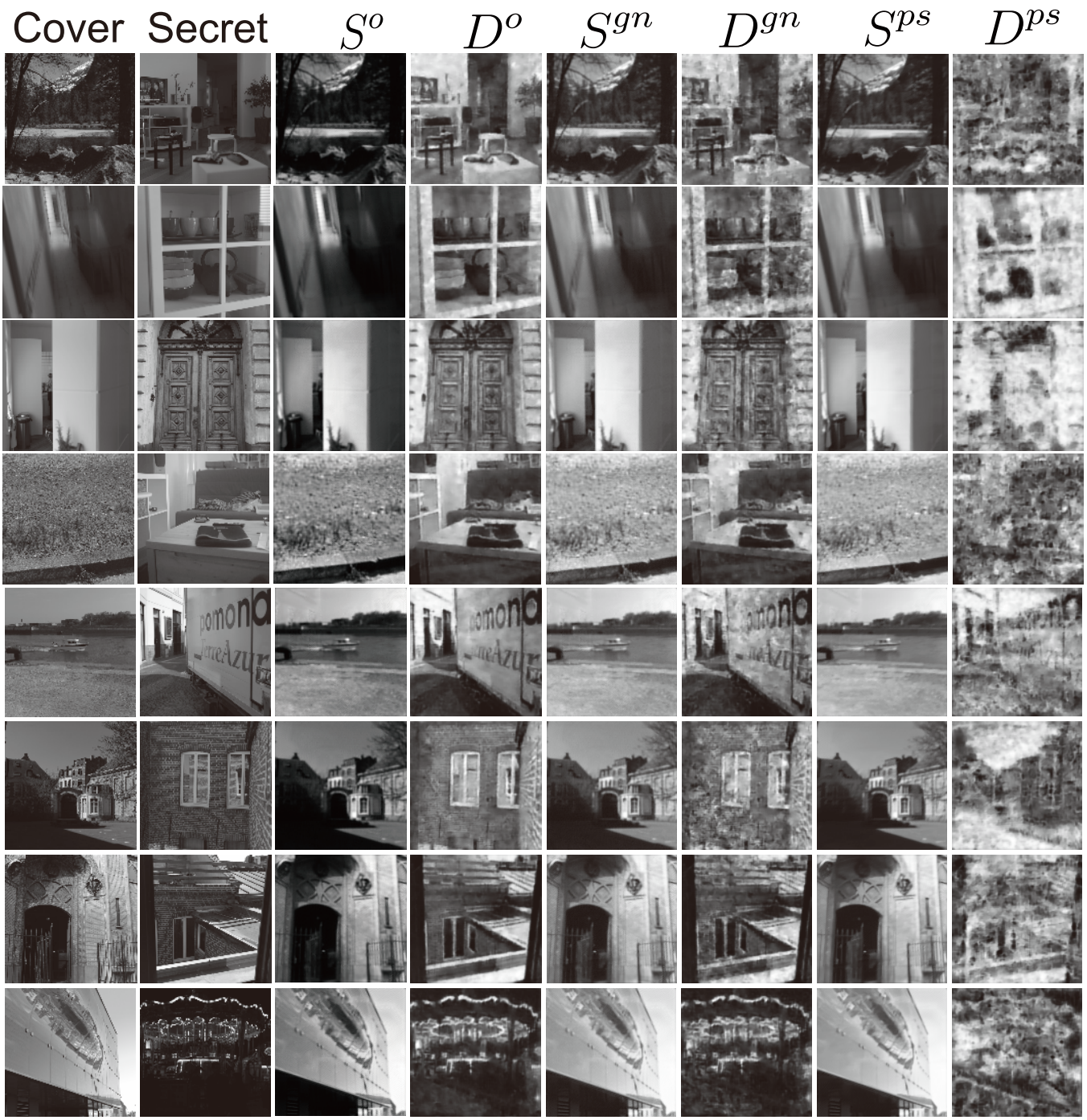

Figure S17: Comparison of the destructed degree of the secret image decoded by the stego image from our method with that of the secret image decoded by the stego image from Gaussian noise when the PSNR between the two stego images is the almost same **(ISGAN on BOSS1.0.1)**. S^o = stego image, D^o = decoded secret image, S^{gn} = stego image modified by Gaussian noise, D^{gn} = secret image decoded from S^{gn}, S^{ps} = stego image modified by our method, and D^{ps} = secret image decoded from S^{ps}

S13 How much the encoded secret image is removed according to ϵ

When ϵ = 2, it is sometimes impossible to recognize the object of the image. However, ϵ = 4 seems to fit if wanted to the reconstructed secret image to be close to noise. Since there may be little visual degradation of the input image with ϵ = 4, we recommend choosing the ϵ by considering all conditions.

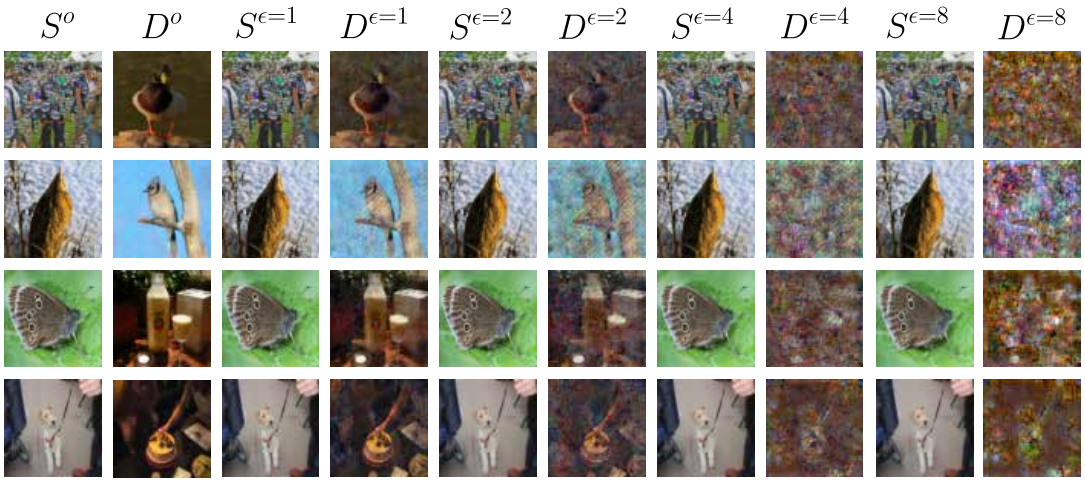

Figure S18: The degree of removal of the secret image encoded in the stego image according to ϵ **(Deep Steganography on ImageNet)**. S^o = stego image, D^o = decoded secret image, $S^{\epsilon=1}$ = stego image modified by our method at ϵ = 1, $D^{\epsilon=1}$ = secret image decoded from $S^{\epsilon=1}$, $S^{\epsilon=2}$ = stego image modified by our method at ϵ = 2, and $D^{\epsilon=2}$ = secret image decoded from $S^{\epsilon=2}$, $S^{\epsilon=4}$ = stego image modified by our method at ϵ = 4, $D^{\epsilon=4}$ = secret image decoded from $S^{\epsilon=4}$, $S^{\epsilon=8}$ = stego image modified by our method at ϵ = 8, and $D^{\epsilon=8}$ = secret image decoded from $S^{\epsilon=8}$

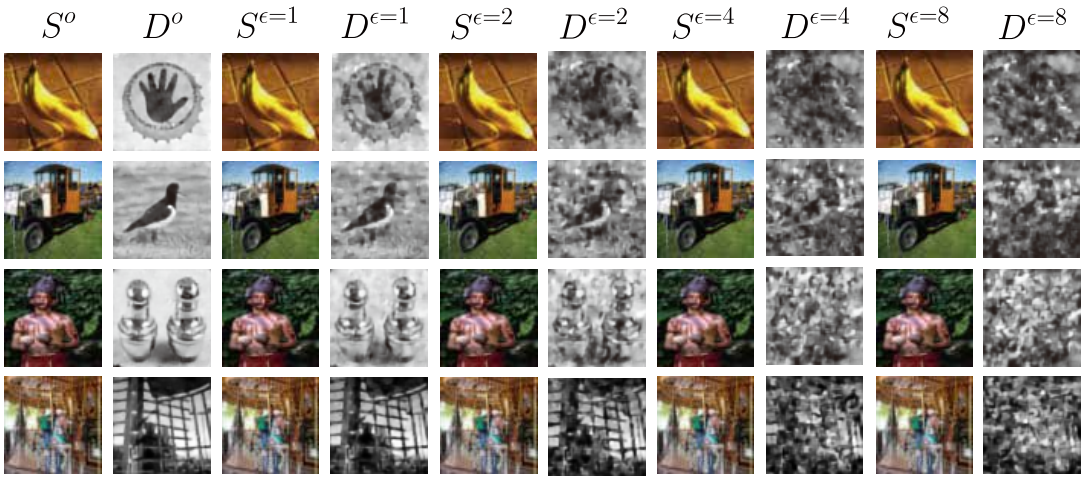

Figure S19: The degree of removal of the secret image encoded in the stego image according to ϵ **(ISGAN on ImageNet)**. S^o = stego image, D^o = decoded secret image, $S^{\epsilon=1}$ = stego image modified by our method at ϵ = 1, $D^{\epsilon=1}$ = secret image decoded from $S^{\epsilon=1}$, $S^{\epsilon=2}$ = stego image modified by our method at ϵ = 2, and $D^{\epsilon=2}$ = secret image decoded from $S^{\epsilon=2}$, $S^{\epsilon=4}$ = stego image modified by our method at ϵ = 4, $D^{\epsilon=4}$ = secret image decoded from $S^{\epsilon=4}$, $S^{\epsilon=8}$ = stego image modified by our method at ϵ = 8, and $D^{\epsilon=8}$ = secret image decoded from $S^{\epsilon=8}$