Probabilistic Selective Encryption of Convolutional Neural Networks for Hierarchical Services (Supplementary Materials)

Jinyu Tian[1], Jiantao Zhou[1,*], and Jia Duan[2]
[1]State Key Laboratory of Internet of Things for Smart City,
Department of Computer and Information Science, University of Macau
[2]JD Explore, JD
{yb77405, jtzhou}@um.edu.mo, duanjia1@jd.com

1. The derivation of formula 3

In this section, we provide detailed derivation of formula (3) in the main file. That is to show

$$\mathbb{P}\left\{\tilde{z}_\theta^{(n)} \neq 0\right\} = Sig\left(\log(p_\theta/(1-p_\theta)) - \beta \log\frac{-\gamma}{\zeta}\right). \tag{1}$$

As shown in formula (2) in the main file,

$$\begin{aligned} &s_\theta(u) = Sig((\log u - \log(1-u) + \log(p_\theta/(1-p_\theta)))/\beta), \\ &\tilde{s}_\theta(u) = s_\theta(u)(\zeta-\gamma)+\gamma, \\ &\tilde{z}_\theta = \min(1, \max(0, \tilde{s}_\theta(u)), \end{aligned} \tag{2}$$

where u is the random variable of uniform distribution $U(0,1)$; $Sig(\cdot)$ represents the sigmoid function; $\zeta > 0$ and $\gamma < 0$ are two parameters to extend the support of $\tilde{z}$ to be $[0,1]$; and p_θ is the importance of the parameter θ. The hyper-parameter β controls the approximation precision of $\tilde{z}_\theta$ to z_θ.

From equations (2), we have that

$$\mathbb{P}\left\{\tilde{z}_\theta^{(n)} \neq 0\right\} = 1 - \mathbb{P}\left\{\tilde{z}_\theta^{(n)} = 0\right\} = 1 - \mathbb{P}\left\{(\tilde{s}_\theta(u) \leq 0)\right\}. \tag{3}$$

According to the cumulative distribution function (CDF) of $\tilde{s}_\theta(u)$ [4] below

$$\mathcal{Q}(\tilde{s}_\theta(u)) = Sig((\log(\frac{\tilde{s}_\theta(u)-\gamma}{\zeta-\tilde{s}_\theta(u)}))\beta - \log(p_\theta/(1-p_\theta))), \tag{4}$$

it is easy to show that

$$\mathbb{P}\left\{\tilde{z}_\theta^{(n)} \neq 0\right\} = 1 - \mathbb{P}\left\{(\tilde{s}_\theta(u) \leq 0)\right\} = 1 - \mathcal{Q}(0) = Sig\left(\log(p_\theta/(1-p_\theta)) - \beta \log\frac{-\gamma}{\zeta}\right). \tag{5}$$

2. Overhead of permissions

In this section, we analyze the overhead of permissions $\mathbf{S}_{\hat{m}}$'s ($\hat{m} = 1, ..., M$) discussed in Section 3.3 of the main file. That is to show the size of a permission $\mathbf{S}_{\hat{m}}$ in bits is

$$(b_\kappa + 64L)\hat{m} + \frac{16L\hat{m}}{M}\phi. \tag{6}$$

As introduced in Section 3.3, a permission $\mathbf{S}_{\hat{m}} = \{\hat{\mathbf{E}}_m, \mathbf{u}_m, \mathbf{v}_m, \kappa_m, \boldsymbol{\mu}, \boldsymbol{\sigma}\}_{m=0}^{\hat{m}-1}$. Suppose the size of an integer and a float number in bits are 16 and 32, respectively. Consider each item $\{\hat{\mathbf{E}}_m, \mathbf{u}_m, \mathbf{v}_m, \kappa_m, \boldsymbol{\mu}, \boldsymbol{\sigma}\}$. The size of $\mathbf{u}_m$ ($\mathbf{v}_m$) in bits is $32L$ by noticing that $\mathbf{u}_m$ ($\mathbf{v}_m$) contains L float numbers. From the equations (7)

$$\hat{\mathbf{E}}_m = \{\hat{\mathbf{E}}_m^l\}_{l=1}^L,\ \hat{\mathbf{E}}_m^l = \{\mathbf{e}_\theta\}_{\theta \in \hat{\boldsymbol{\Theta}}_m^l}, \tag{7}$$

in the main file, where $\mathbf{e}_\theta$ is a 2-D tuple to locate which layer θ belongs to and the position of θ in this layer. To represent one $\mathbf{e}_\theta$, we need two integers. Since the first coordinate of $\mathbf{e}_\theta$ in $\hat{\boldsymbol{\Theta}}_m^l$ is the same, the overhead of $\hat{\mathbf{E}}_m^l$ thus is $16 + 16\phi/M$ (note that $\hat{\boldsymbol{\Theta}}_m^l$ contains ϕ/M elements). Therefore, the total size of items $\hat{\mathbf{E}}_m$, $\mathbf{u}_m$, $\mathbf{v}_m$, and κ_m is

$$64L + 16 + \frac{16L\phi}{M} + b_\kappa, \tag{8}$$

where b_κ is the bit size of the secret key κ, and generally is determined by the key generator. Both of the mean values $\boldsymbol{\mu}$ and the standard deviations $\boldsymbol{\sigma}$ contain L float numbers. To this end, the overhead of the permission $\mathbf{S}_{\hat{m}}$ in bits is

$$(b_\kappa + 16)\hat{m} + 64L(\hat{m} + 1) + \frac{16L\hat{m}}{M}\phi. \tag{9}$$

3. Proof of Theorem 1

In this section, we provide the proof for Theorem 1. Before diving into the proof, let us introduce the definition of *equivocation* [1, 5] proposed by Shannon used for measure the information leakage.

DEFINITION. *Let $H(\hat{\mathbf{W}}^l)$ be the entropy of samples in $\hat{\mathbf{W}}^l$. Let also $I(\hat{\mathbf{W}}^l; \tilde{\boldsymbol{\Theta}}^l)$ be the mutual information between $\hat{\mathbf{W}}^l$ and $\tilde{\boldsymbol{\Theta}}^l$. The equivocation of $\hat{\mathbf{W}}^l$ by observing $\tilde{\boldsymbol{\Theta}}^l$ is defined as*

$$E(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l) = H(\hat{\mathbf{W}}^l) - I(\hat{\mathbf{W}}^l; \tilde{\boldsymbol{\Theta}}^l). \tag{10}$$

Remark 1: A large value of $E(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l)$ implies that a slight information leakage of $\hat{\mathbf{W}}^l$ when observing $\tilde{\boldsymbol{\Theta}}^l$ and vice versa.

Remark 2: For the convenience of the theoretical justification, we modify the definition (10) of equivocation into another form $\hat{E}(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l) = I(\hat{\mathbf{W}}^l; \tilde{\boldsymbol{\Theta}}^l)/H(\hat{\mathbf{W}}^l)$ so that the value of $\hat{E}(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l)$ is positively correlated with the information leakage of $\hat{\mathbf{W}}^l$. Specifically, when $\hat{E}(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l) \to 0$, the information leakage of $\hat{\mathbf{W}}^l$ via observing $\tilde{\boldsymbol{\Theta}}^l$ is negligible. In this case, the original equivocation $E(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l)$ tends to $H(\hat{\mathbf{W}}^l)$, which is the maximum of the $E(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l)$. We call $\hat{E}(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l)$ modified equivocation in the rest of the discussion.

To prove Theorem 1, we briefly discuss the distribution of dominated parameters $\hat{\boldsymbol{\Theta}}^l$, their ciphertext $\hat{\mathbf{C}}^l$, and the noise $\hat{\mathbf{W}}^l = \hat{\mathbf{C}}^l - \hat{\boldsymbol{\Theta}}^l$ added on $\hat{\boldsymbol{\Theta}}^l$. Recalling the encryption of the dominated parameters $\hat{\boldsymbol{\Theta}}^l$, both parameters in $\hat{\boldsymbol{\Theta}}^l$ and their ciphertext $\hat{\mathbf{C}}^l$ following the same Gaussian distribution. More precisely, let $\hat{\theta}$ and $\hat{c}$ be two random variables of the Gaussian distribution $\mathcal{N}(\cdot|\mu^l, \sigma^l)$, where μ^l and σ^l are mean value and standard deviation of parameters in the l-th of the pretrained model $\mathcal{F}_{\boldsymbol{\Theta}}$. $\hat{\boldsymbol{\Theta}}^l$ and $\hat{\mathbf{C}}^l$ thus are samples drawn from $\hat{\theta}$ and $\hat{c}$, respectively. Now, considering samples $\mathbf{w}_i$'s in $\mathbf{W}^l = \hat{\mathbf{C}}^l - \hat{\boldsymbol{\Theta}}^l$, it is quit clear that $\mathbf{w}_i$'s are drawn from the following random variable,

$$\hat{w} = \hat{c} - \hat{\theta}, \quad \hat{w} \sim \mathcal{N}(\hat{w}|0, 2\sigma^l). \tag{11}$$

Moreover, given $\hat{\theta}_j \in \hat{\boldsymbol{\Theta}}^l$, the conditional distribution of random variable $\hat{w}|\hat{\theta}_j = \hat{c} - \hat{\theta}_j$ follows the distribution below

$$\hat{w}|\hat{\theta}_j \sim \mathcal{N}(\hat{w}|\mu^l - \hat{\theta}_j, \sigma^l). \tag{12}$$

Upon having the distributions (11) and (12), we now prove Theorem 1 in the main file.

Theorem 1. *The modified equivocation between $\hat{\mathbf{W}}^l$ and $\tilde{\boldsymbol{\Theta}}^l$ is of order $|\hat{\mathbf{W}}^l|^{-1/2}$. That is*

$$\hat{E}(\hat{\mathbf{W}}^l, \tilde{\boldsymbol{\Theta}}^l) = O(|\hat{\mathbf{W}}^l|^{-1/2}), \tag{13}$$

as $|\hat{\mathbf{W}}^l| \to \infty$.

Proof. By using the proposition of mutual information, we have

$$\hat{E}(\hat{\mathbf{W}}^l, \tilde{\mathbf{\Theta}}^l) = \frac{I(\hat{\mathbf{W}}^l; \tilde{\mathbf{\Theta}}^l)}{H(\hat{\mathbf{W}}^l)} = 1 - \frac{H(\hat{\mathbf{W}}^l|\tilde{\mathbf{\Theta}}^l)}{H(\hat{\mathbf{W}}^l)}. \tag{14}$$

Since $\hat{\mathbf{W}}^l$ and $\tilde{\mathbf{\Theta}}^l$ are samples of continuous random variables, we approximate the discrete entropy $H(\hat{\mathbf{W}}^l)$ and $H(\hat{\mathbf{W}}^l|\tilde{\mathbf{\Theta}}^l)$ with unequal quantization intervals, $\triangle_{\hat{w}_i}$ and $\triangle_{\tilde{\theta}_i}$ as follows [2].

$$\begin{aligned}
H(\hat{\mathbf{W}}^l|\tilde{\mathbf{\Theta}}^l) &= \sum_{j=1}^{|\tilde{\mathbf{\Theta}}^l|-1} p(\tilde{\theta}_j) H(\hat{\mathbf{W}}^l|\tilde{\theta}_j) \triangle_{\tilde{\theta}_j}, \\
&= \sum_{j=1}^{|\tilde{\mathbf{\Theta}}^l|-1} p(\tilde{\theta}_j) \left(- \sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i|\tilde{\theta}_j) \log\left(p(\hat{w}_i|\tilde{\theta}_j) \triangle_{\hat{w}_i} \right) \triangle_{\hat{w}_i} \right) \triangle_{\tilde{\theta}_j}, \\
&= \sum_{j=1}^{|\tilde{\mathbf{\Theta}}^l|-1} p(\tilde{\theta}_j) \left(- \left(\sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i|\tilde{\theta}_j) \log(p(\hat{w}_i|\tilde{\theta}_j)) \triangle_{\hat{w}_i} \right) \triangle_{\tilde{\theta}_j} - \left(\sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i|\tilde{\theta}_j) \triangle_{\hat{w}_i} \log(\triangle_{\hat{w}_i}) \right) \triangle_{\tilde{\theta}_j} \right),
\end{aligned} \tag{15}$$

and

$$\begin{aligned}
H(\hat{\mathbf{W}}^l) &= - \sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i) \log(p(\hat{w}_i) \triangle_{\hat{w}_i}) \triangle_{\hat{w}_i}, \\
&= - \left(\sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i) \log(p(\hat{w}_i)) \triangle_{\hat{w}_i} + \sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i) \log(\triangle_{\hat{w}_i}) \triangle_{\hat{w}_i} \right).
\end{aligned} \tag{16}$$

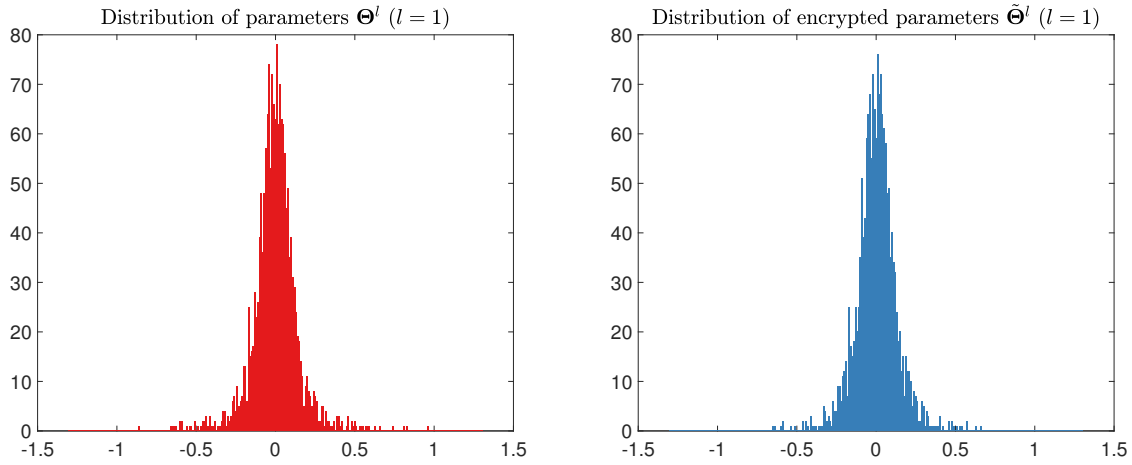
Here, without loss the generality, we sort $\hat{\mathbf{W}}^l = \{\hat{w}_i\}_{i=1}^{|\hat{\mathbf{W}}^l|}$ and $\tilde{\mathbf{\Theta}}^l = \{\tilde{\theta}_i\}_{i=1}^{|\tilde{\mathbf{\Theta}}^l|}$ with ascending order. That is, $\hat{w}_{i+1} > \hat{w}_i$ and $\tilde{\theta}_{i+1} > \tilde{\theta}_i$. Thus, $\triangle_{\hat{w}_i} = \hat{w}_{i+1} - \hat{w}_i$, and $\triangle_{\tilde{\theta}_j} = \tilde{\theta}_{j+1} - \tilde{\theta}_j$.

According to the definition of definite integral, and note that $\hat{w} \sim \mathcal{N}(\hat{w}|0, 2\sigma^l)$, as $|\hat{\mathbf{W}}^l| \to \infty$, we have

$$\begin{aligned}
\lim_{\triangle_{\hat{w}_i} \to 0} - \sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i) \log(p(\hat{w}_i)) \triangle_{\hat{w}_i} &= \int_{-\infty}^{+\infty} -p(\hat{w}) \log(p(\hat{w})) \mathrm{d}\hat{w}, \\
&= - \int_{-\infty}^{+\infty} - \frac{\exp(\frac{\hat{w}^2}{4(\sigma^l)^2})}{2\sigma^l\sqrt{2\pi}} \log(-\frac{\exp(\frac{\hat{w}^2}{4(\sigma^l)^2})}{2\sigma^l\sqrt{2\pi}}) \mathrm{d}\hat{w}, \\
&= \frac{1}{2} \log(8\pi e(\sigma^l)^2).
\end{aligned} \tag{17}$$

Similarly, by noticing that $\hat{w}|\hat{\theta}_j \sim \mathcal{N}(\hat{w}|\mu^l - \hat{\theta}_j, \sigma^l)$ as (12), we have

$$\begin{aligned}
\lim_{\triangle_{\hat{w}_i} \to 0} - \sum_{i=1}^{|\hat{\mathbf{W}}^l|-1} p(\hat{w}_i|\tilde{\theta}_j) \log(p(\hat{w}_i|\tilde{\theta}_j)) \triangle_{\hat{w}_i} &= \int_{-\infty}^{+\infty} -p(\hat{w}|\tilde{\theta}_j) \log(p(\hat{w}|\tilde{\theta}_j)) \mathrm{d}\hat{w}, \\
&= \int_{-\infty}^{+\infty} - \frac{\exp(-\frac{(\hat{w}+\tilde{\theta}_j-\mu^l)^2}{2(\sigma^l)^2})}{\sigma^l\sqrt{2\pi}} \log(\frac{\exp(-\frac{(\hat{w}+\tilde{\theta}_j-\mu^l)^2}{2(\sigma^l)^2})}{\sigma^l\sqrt{2\pi}}) \mathrm{d}\hat{w}, \\
&= \frac{1}{2} \log(2\pi e(\sigma^l)^2).
\end{aligned} \tag{18}$$



(a) (b)

Fig. 1: Distribution of parameters of the 1-st layer of VGG19, before and after encrypted.

Here the calculation of integral in (17) and (18) can be found in reference [2]. Therefore,

$$
\begin{aligned}
&\lim_{|\tilde{\mathbf{\Theta}}^l|\to\infty}\lim_{|\hat{\mathbf{W}}^l|\to\infty}\hat{E}(\hat{\mathbf{W}}^l,\tilde{\mathbf{\Theta}}^l)=1-\lim_{|\tilde{\mathbf{\Theta}}^l|\to\infty}\lim_{|\hat{\mathbf{W}}^l|\to\infty}\frac{H(\hat{\mathbf{W}}^l|\tilde{\mathbf{\Theta}}^l)}{H(\hat{\mathbf{W}}^l)},\\
&=1-\lim_{|\tilde{\mathbf{\Theta}}^l|\to\infty}\lim_{|\hat{\mathbf{W}}^l|\to\infty}\sum_{j=1}^{|\tilde{\mathbf{\Theta}}^l|-1}p(\tilde{\theta}_j)\frac{\log(2\pi e(\sigma^l)^2)-2\sum_{i=1}^{|\hat{\mathbf{W}}^l|-1}p(\hat{w}_i|\tilde{\theta}_j)\log(\triangle_{\hat{w}_i})\triangle_{\hat{w}_i}}{\log(8\pi e(\sigma^l)^2)-2\sum_{i=1}^{|\hat{\mathbf{W}}^l|-1}p(\hat{w}_i)\log(\triangle_{\hat{w}_i})\triangle_{\hat{w}_i}}\triangle_{\tilde{\theta}_j},\\
&=1-\lim_{|\tilde{\mathbf{\Theta}}^l|\to\infty}\frac{\log(2\pi e(\sigma^l)^2)-\sum_{j=1}^{|\tilde{\mathbf{\Theta}}|-1}p(\tilde{\theta}_j)\mathbb{E}_{\hat{w}|\tilde{\theta}_j}(\log(\triangle_{\hat{w}}))\triangle_{\tilde{\theta}_j}}{\log(8\pi e(\sigma^l)^2)-2\mathbb{E}(\log(\triangle_{\hat{w}}))},\\
&=1-\frac{\log(2\pi e(\sigma^l)^2)-2\mathbb{E}(\log(\triangle_{\hat{w}}))}{\log(8\pi e(\sigma^l)^2)-2\mathbb{E}(\log(\triangle_{\hat{w}}))},\\
&=\frac{\log 4}{\log(8\pi e(\sigma^l)^2)-2\mathbb{E}(\log(\triangle_{\hat{w}}))},
\end{aligned}
\tag{19}
$$

where $\mathbb{E}(\log(\triangle_{\hat{w}}))$ denotes the expectation of $\log(\triangle_{\hat{w}})$. Consequently, based on the the Jensen's inequality, we have

$$
\begin{aligned}
\lim_{|\tilde{\mathbf{\Theta}}^l|\to\infty}\lim_{|\hat{\mathbf{W}}^l|\to\infty}\hat{E}(\hat{\mathbf{W}}^l,\tilde{\mathbf{\Theta}}^l)&=\frac{\log 4}{\log(8\pi e(\sigma^l)^2)-2\mathbb{E}(\log(\triangle_{\hat{w}}))},\\
&\leq\frac{\log 4}{\log(8\pi e(\sigma^l)^2)-\log(\mathbb{E}(\triangle_{\hat{w}})^2)}.
\end{aligned}
\tag{20}
$$

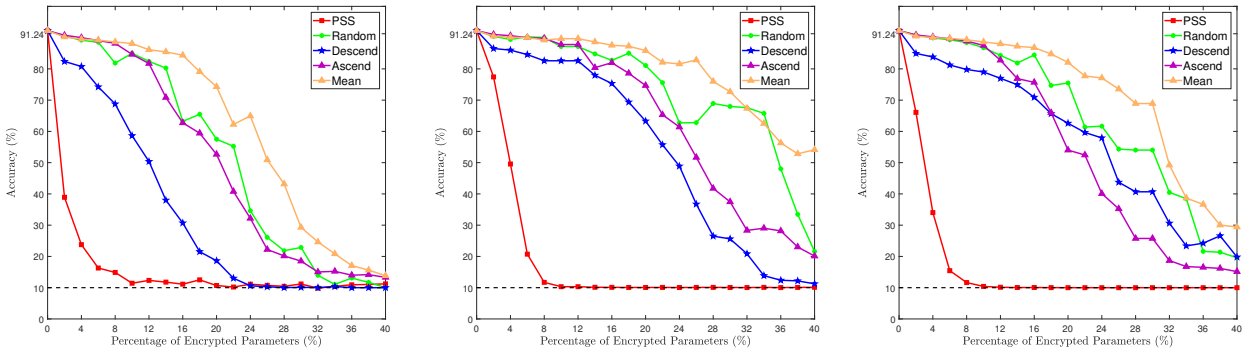
Note that $|\hat{\mathbf{W}}^l|\to\infty$ implies $|\tilde{\mathbf{\Theta}}^l|\to\infty$. Thus, we have

$$
\lim_{|\hat{\mathbf{W}}^l|\to\infty}\hat{E}(\hat{\mathbf{W}}^l,\tilde{\mathbf{\Theta}}^l)\leq\frac{\log 4}{\log(8\pi e(\sigma^l)^2)-\log(\mathbb{E}(\triangle_{\hat{w}})^2)}.
\tag{21}
$$

The upper bound in (21) depends on the term $\mathbb{E}(\triangle_{\hat{w}})$, which is the average difference between successive Gaussian samples. The work [3] has pointed out that the term $\mathbb{E}(\triangle_{\hat{w}})\to 0$ as $|\hat{\mathbf{W}}^l|\to\infty$ with order $O(e(|\hat{\mathbf{W}}^l|^{1/4}))$. Thus $\log(\mathbb{E}(\triangle_{\hat{w}})^2)\to\infty$ with order $O(|\hat{\mathbf{W}}^l|^{1/2})$. Consequently, we have that $\hat{E}(\hat{\mathbf{W}}^l,\tilde{\mathbf{\Theta}}^l)\to 0$ with order $O(|\hat{\mathbf{W}}^l|^{-1/2})$ as $|\hat{\mathbf{W}}^l|\to\infty$. □

Model	Epochs	Batch Size	Optimizer	Weight Decay	Momentum
VGG19	300	128	SGD	0.005	0.9
DnCNN	50	128	SGD	0.0001	0.9

Table 1: Hyper-parameters for training VGG19 and DnCNN



(a) (b) (c)

Fig. 2: The classification accuracy of the protected VGG19 on CIFAR10 with respect to different percentages of encrypted parameters. (a) 1-st, 2-nd, 4-th, 5-th, 7-th and 8-th layers are encrypted; (b) 1-st, 2-nd, 9-th, and 12-th layers are encrypted; (c) 1-st, 2-nd, 4-th, and 9-th layers are encrypted.

3.1. Experimental justification of the imperceptibility of ciphertext

The Theorem 1 theoretically supports the imperceptibility of ciphertext $\hat{\mathbf{C}}^l$ of dominated parameters $\hat{\mathbf{\Theta}}^l$. In this section, we further empirically demonstrate the imperceptibility of $\hat{\mathbf{C}}^l$ by justifying that the noise $\hat{\mathbf{W}}^l = \hat{\mathbf{C}}^l - \hat{\mathbf{\Theta}}^l$ added on the dominated parameters $\hat{\mathbf{\Theta}}^l$ will not cause the change of distribution of parameters. Fig. 1 shows the distributions of parameters (the 1-st layer of VGG19) $\mathbf{\Theta}^l$ and its partially encrypted version $\tilde{\mathbf{\Theta}}^l$, in which parameters $\hat{\mathbf{\Theta}}^l$ in $\mathbf{\Theta}^l$ are contaminated by $\hat{\mathbf{W}}^l$. As can be seen, the distribution of parameters $\mathbf{\Theta}^l$ is almost the same as that $\tilde{\mathbf{\Theta}}^l$. In other words, attackers cannot distinguish the original parameters $\mathbf{\Theta}^l$ from its partially encrypted version $\tilde{\mathbf{\Theta}}^l$. As a result of this indistinguishability, they cannot capture the ciphertext $\hat{\mathbf{C}}^l$ of $\hat{\mathbf{\Theta}}^l$ from $\tilde{\mathbf{\Theta}}^l$ by treating $\hat{\mathbf{C}}^l$ as abnormal values out of the distribution of unencrypted parameters.

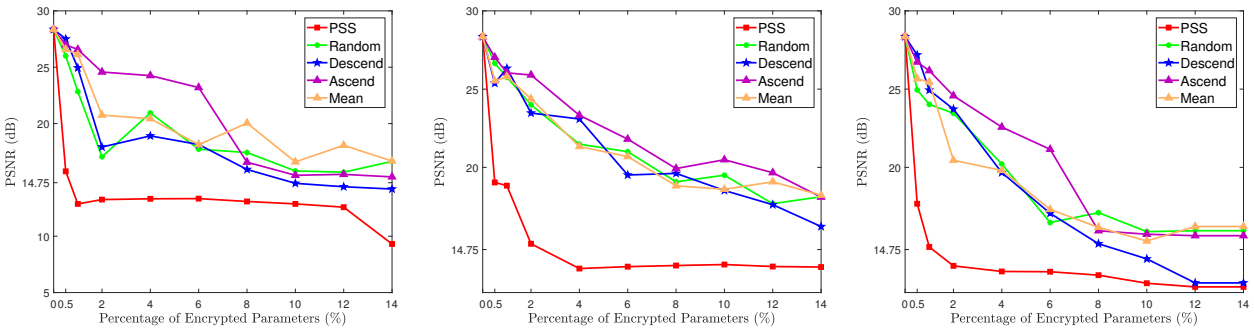
4. Experimental configurations and more results

In this section, we describe detailed explanations about all the experiments described in Section 5 of the main file. Also, we provide more experimental results.

4.1. Detailed experimental configurations

Configuration of pretrained models: We consider two CNN models for experiments in the main file: VGG19 for classification and DnCNN for denoising. We train VGG19 on 30000 images from CIFAR10 and train DnCNN on 300 noisy images scaled into 180×180 as the recommendation of the original work [6]. The initial learning rate for training VGG19 is 0.0001 and is decayed by a factor of 0.1 once the epoch reaches one of the milestones $[50, 70, 90, 100]$. Simialry, for DnCNN, the learning rate was decayed from 0.1 to 0.0001 with milestones $[10, 20, 30]$. All experiments are running on the platform with two GPUs (NVIDIA 1080 8G). Detailed about other hyper-parameters, such as weight decay, of the training of the two models are listed in Table 1.

Hyper-parameters of our method: The most important hyper-parameter in our method is the weighting factor λ of the problem (1) in the main file. We choose the weighting factor λ so that the magnitude of the cost term is consistent with that of the regularization term. More precisely, when solving the problem (1) to 1-st, 2-nd, 5-th, and 9-th layers of VGG19, λ's are 0.01, 0.001, 0.0001, and 0.0001, respectively. For considering layers (6-th, 9-th, and 12-th) of DnCNN, λ's are 0.1, 0.1, and 0.1. To solve the optimization problem (1), we initialize the importance of parameters, i.e. p_θ's, with equal values 0.5.



(a) (b) (c)

Fig. 3: The denoising performance of the protected DnCNN with respect to different percentages of encrypted parameters. (a) 9-th, 12-th, and 18-th layers are encrypted; (b) 21-th, 24-th, and 27-th layers are encrypted; (c) 9-th, 15-th, 18-th, and 24-th layers are encrypted.

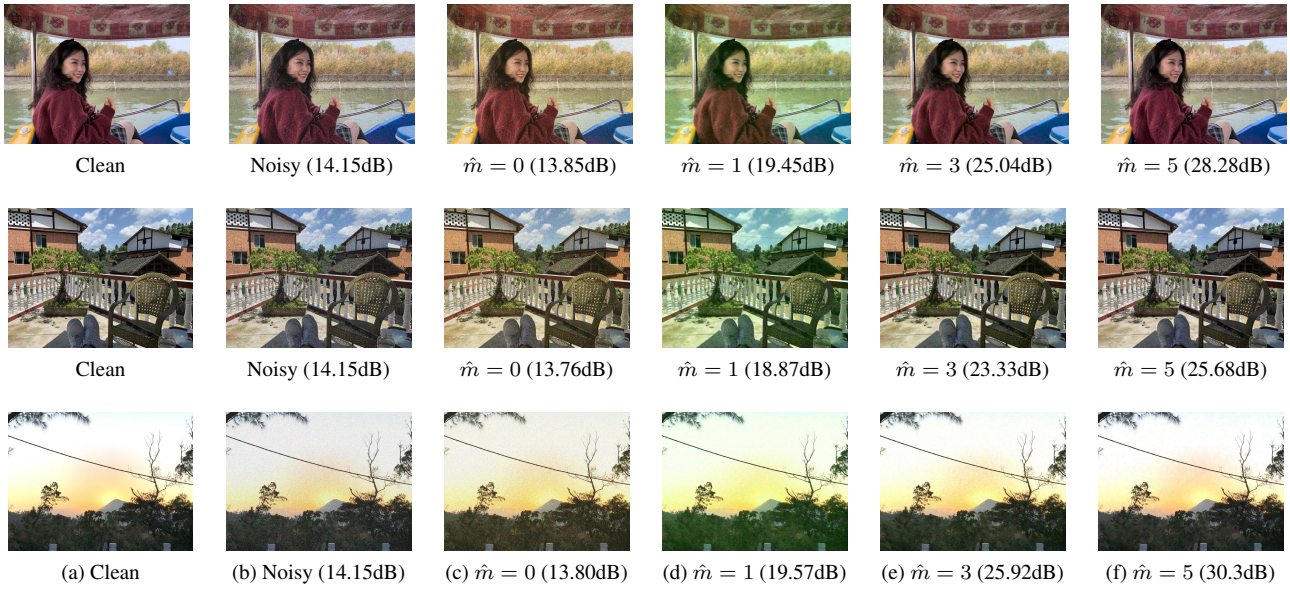


Fig. 4: Images to illustrate the hierarchical performance of the protected DnCNN. (a) Clean image; (b) Noisy image; (c) The output image of the protected DnCNN without permission; (d-f) Denoising images of decrypted DnCNN with different permissions $\mathbf{S}_{\hat{m}}$'s.

Note that samples for solving this problem is the training data of the pretrained model. There also are three parameters β, ζ, and γ in the problem (1). We set them to be 0.66, 1.1, and -0.1, respectively, as the recommendation of the work in [4].

4.2. Effectiveness of the proposed SE

In Section 5.1 of the main file, we have verified the effectiveness of the proposed SE on protecting VGG19 and DnCNN by encrypting several layers. Here we provide more experimental results under different encrypted layers to further demonstrate the effectiveness of the proposed SE. As shown in Fig. 2 (a)-(c), one can see that the classification accuracy of the VGG19 protected by our method degrades to the worst case with less encrypted parameters than that of competing parameter selection strategies. A similar phenomenon of the protected DnCNN can be observed from Fig. 3 (a)-(c), in which the denoising performance of DnCNN protected by our SE degrades much faster than DnCNN protected by competitors.

4.3. Hierarchical performance of the released model

In this section, we provide more visualized results to illustrate the hierarchical performance of the released DnCNN, as shown in Fig. 4. Similar to the results in Section 5.2 of the main file, the higher the permission $\hat{m}$, the better the denoising performance of the decrypted model will show.

References

[1] François Cayre, Caroline Fontaine, and Teddy Furon. Watermarking security: theory and practice. *IEEE Transactions on Signal Processing*, 53(10):3976–3987, 2005. 2

[2] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999. 3, 4

[3] H Leon Harter. Expected values of normal order statistics. *Biometrika*, 48(1-2):151–165, 1961. 4

[4] Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l_0 regularization. In *International Conference on Learning Representations*, pages 1–12, 2017. 1, 6

[5] Claude E Shannon. Communication theory of secrecy systems. *The Bell system technical journal*, 28(4):656–715, 1949. 2

[6] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 5