A. Derivation of Regularization Gradient

The detail derivation of regularization gradient is given below if \( \hat{T} \neq pT \):

\[
\frac{\partial R_{\text{log}}}{\partial \theta_{l,c}} = \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial \sum_{l=1}^{L} (\text{FLOPs})_l}{\partial \theta_{l,c}}
\]

\[
= \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}}
\]

\[
= \eta_l \cdot \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}}
\]

where \( \eta_l = k^2_l \cdot \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}} \).

The result of the second line is due to the definition of (FLOPs)_l and (FLOPs) = \( k_l \cdot k_l \cdot \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}} \).

The result of the fourth line is because of STE: \( \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}} = 1 \), if \( \theta_{l,c} \in [0, 1] \). If \( \hat{T} = pT \), then \( R_{\text{log}} = 0 \), and 0 can be used as the subgradient of this point. Thus, we have the sub-gradient given in the paper:

\[
\frac{\partial R_{\text{log}}}{\partial \theta_{l,c}} = \begin{cases} 
\eta_l \cdot \frac{1}{|\hat{T} - pT| + 1} \cdot \frac{\hat{T} - pT}{|\hat{T} - pT|} \cdot \frac{\partial g_{l,c}(\hat{w}_l, h_l)}{\partial \theta_{l,c}}, & \text{if } \hat{T} \neq pT \\
0, & \text{if } \hat{T} = pT 
\end{cases}
\]

B. Detailed Choice of \( p \)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( p )</td>
<td>0.55</td>
<td>0.38</td>
<td>0.42</td>
<td>0.50</td>
</tr>
</tbody>
</table>

Table 1: Choice of \( p \) for ImageNet models. \( p \) is the remained FLOPs divided by the total FLOPs.

In this section, we will give the detail number of \( p \). In a CNN, we do not prune the first layer, the last layer and residual connections in ResNet. As a result, the actual remained FLOPs may not equal to \( p \). We list the choice of \( p \) for ImageNet models in Tab.1. For CIFAR-10 models, the unpruned FLOPs is quite small, thus, \( p \) is the same as the remained fraction of FLOPs.

C. Acceleration

The cpu run time of different models are shown in Tab.2.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Time (ms/batch)</td>
<td>113.5</td>
<td>195.7</td>
<td>331.5</td>
<td>106.8</td>
</tr>
<tr>
<td>Pruned Time (ms/batch)</td>
<td>81.4</td>
<td>126.2</td>
<td>208.4</td>
<td>67.5</td>
</tr>
<tr>
<td>Improvement (%)</td>
<td>28.3%</td>
<td>35.5%</td>
<td>38.0%</td>
<td>36.8%</td>
</tr>
</tbody>
</table>

Table 2: CPU time for different ImageNet Models. The time is measured in millisecond.

D. Discussion of Difference between Proposed Method and Trainable Gate [2]

In trainable gate (TG) [2], they propose to turn non-differentiable gate to differentiable gate inspired by [1]. They add a perturbation to the gate function to make it differentiable. Our method, on the other hand, does not modify the gate function and use STE to handle gradient calculation. Thus, the approach of making gate differentiable is different. Moreover, in their framework, the gate calculation is deterministic. Consequently, they can not sample sub-networks as we do. Their work also applies a form of constraint to limit the resource of the pruned neural network.

References