Appendices

We first show RPG networks could be quantized with minimal accuracy drop for compression purpose in Section A. We then provide a figure revealing log-linear DoF-accuracy relationship in Section B. We also provide proof for the orthogonal proposition in the main paper (Section C). Finally, we provide detailed comparison and discussion to a closely related work HyperNetworks [22] in Section D.

Additionally, we provide the most important code to reproduce the layer superposition experiments on ImageNet in supplementary as a tgz file. The rest of code is also ready for release, and will be released after additional internal review.

A. Quantize RPG

Quantization refers to techniques for performing computations and storing tensors at lower bitwidths than floating point precision. Quantization can reduce model size with tiny accuracy drop. Table 9 shows that with 8-bit quantization, ResNet18-vanilla has an accuracy drop of 0.3 percentage point, while our ResNet18-RPG has an accuracy drop of 0.1 percentage point. RPG models can be quantized for further model size reduction with a negligible accuracy drop.

Table 9: RPG model can be quantized with very tiny accuracy drop. With 8-bit quantization on ImageNet, ResNet18-vanilla has an accuracy drop of 0.3 percentage point, while our ResNet18-RPG has an accuracy drop of 0.1 percentage point.

	# Params	Acc before	Acc after ↓ quantization	Acc drop
R18-vanilla	11M	69.8	69.5	0.3
R18-RPG	5.6M	70.2	70.1	0.1

B. CIFAR100 Accuracy versus DoF

Fig.6 plots CIFAR100 classification accuracy versus model DoF. We observe a similar log-linear relationship as in ImageNet.

C. Proof to the Orthogonal Proposition

We provide proofs to the orthogonal proposition mentioned in Section 3 of the main paper. Suppose we have two vectors $\mathbf{f}_i = \boldsymbol{A}_i\mathbf{f}, \mathbf{f}_j = \boldsymbol{A}_i\mathbf{f}$, where $\boldsymbol{A}_i$, $\boldsymbol{A}_j$ are sampled from the $O(M)$ Haar distribution.

Proposition 1. $\mathrm{E}\left[\langle \mathbf{f}_i, \mathbf{f}_j \rangle\right] = 0.$

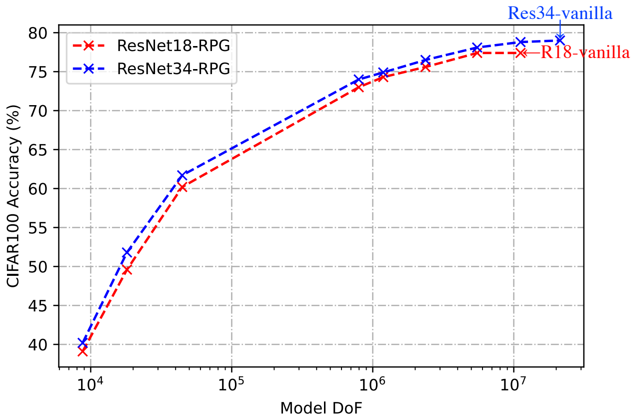

Figure 6: Log-linear DoF-accuracy relationship of CIFAR100 accuracy and model DoF on CIFAR100. RPG achieves the same accuracy as vanilla ResNet with 50% DoF.

Proof.

$$\begin{aligned}
\mathrm{E}\left[\langle \mathbf{f}_i, \mathbf{f}_j \rangle\right] &= \mathrm{E}\left[\langle \mathbf{f}_i, \mathbf{f}_j \rangle\right] \\
&= \mathrm{E}\left[\langle \boldsymbol{A}_i\mathbf{f}, \boldsymbol{A}_j\mathbf{f} \rangle\right] \\
&= \mathrm{E}\left[\langle \mathbf{f}, \boldsymbol{A}_i^T\boldsymbol{A}_j\mathbf{f} \rangle\right] \\
&= \mathbf{f}^T\mathrm{E}\left[\boldsymbol{A}_i^T\boldsymbol{A}_j\right]\mathbf{f} \\
&= 0
\end{aligned}$$

where $\boldsymbol{A}_i^T\boldsymbol{A}_j$ is equivalently a random sample from $O(M)$ Haar distribution and its expectation is clearly 0. □

Proposition 2. $\mathrm{E}\left[\langle \frac{\mathbf{f}_i}{||\mathbf{f}_i||}, \frac{\mathbf{f}_j}{||\mathbf{f}_j||} \rangle^2\right] = \frac{1}{M}.$

Proof.

$$\begin{aligned}
\mathrm{E}\left[\langle \frac{\mathbf{f}_i}{||\mathbf{f}_i||}, \frac{\mathbf{f}_j}{||\mathbf{f}_j||} \rangle^2\right] &= \frac{\mathrm{E}\left[\langle \boldsymbol{A}_i\mathbf{f}, \boldsymbol{A}_j\mathbf{f} \rangle^2\right]}{||\mathbf{f}||_2^2||\mathbf{f}||_2^2} \\
&= \mathrm{E}\left[\langle \boldsymbol{A}\frac{\mathbf{f}}{||\mathbf{f}||}, \frac{\mathbf{f}}{||\mathbf{f}||} \rangle^2\right],
\end{aligned}$$

where $\boldsymbol{A} = \boldsymbol{A}_i^T\boldsymbol{A}_j \sim O(M)$ Haar distribution

Due to the symmetry,

$$= \mathrm{E}\left[\langle \boldsymbol{A}\frac{\mathbf{f}}{||\mathbf{f}||}, (1,0,0,\ldots,0)^T \rangle^2\right]$$

Let $\mathbf{g} = \boldsymbol{A}\frac{\mathbf{f}}{||\mathbf{f}||}$,

$$\begin{aligned}
&= \mathrm{E}\left[g_1^2\right] \\
&= \frac{1}{M}
\end{aligned}$$

since $\mathbf{g}$ is a random unit vector and $\mathrm{E}\left[\sum_{k=1}^M g_k^2\right] = \sum_{k=1}^M \mathrm{E}\left[g_k^2\right] = 1.$ □

D. Comparison to HyperNetworks

HyperNetworks [22] share similarity with RPG as both methods reduce model DoF. Specifically, HyperNetworks rely on learnable modules to generate network parameters. We compare with them and report results in Table 10. On CIFAR100 with the embedding dimension of 64 and the same model size, HyperNetworks has 68x FLOPs as our RPG, yet 10 percentage points lower than RPG in accuracy.

Table 10: RPG outperforms HyperNetworks [22] with same DoF on CIFAR100. HyperNetworks has 68x FLOPs as our RPG, yet 10 percentage points lower than RPG in accuracy.

	model DoF	**FLOPs**	**CIFAR100 Acc.**
HyperNet [22]	632k	2.49G	61.3%
RPG	632k	36.7M	71.6%

RPG can be considered as an extreme and minimal version of HyperNetworks, one without a network. However, RPG's unique design and implementation delivers the following advantages over HyperNetworks:

1. HyperNetworks add substantial FLOPs to the network and render it less practical. Given a network architecture, RPG adds minimal to no additional computation, as the permutation and sign reflection can be efficiently implemented. However, HyperNetworks use a weight generation network to generate the primary network weights. A hypernet mainly uses matrix multiplication and introduces substantial FLOPs. In the table below, we analyze FLOPs of HyperNetwork for ResNet18 with the embedding dimension of 64. FLOPs of a vanilla-Res18 for ImageNet (224 input size) and CIFAR100 (32 input size) are 1.8G and 36.7M, whereas the weight generation part of the HyperNet-Res18 takes 2.45G FLOPs. This means the weight generation FLOPs are 1.4 times of vanilla-Res18 for ImageNet and 67 times of that of CIFAR100. Empirically, we find the training and inference time HyperNet-Res18 is around 70x larger than vanilla-Res18.

2. HyperNetworks do not have an arbitrary DoF (number of reduced parameters). RPG uses a model ring of a size (model DoF) that can be arbitrarily determined. In HyperNetworks, the weight generation network uses the same hyper-weight and requires embedding to be of a certain size so that the matrix multiplication can be used for generating primary network weights. Therefore, the model DoF or reduced number of parameters cannot be arbitrarily determined. In other words, RPG decouples the model DoF (actual parameters) and the network architecture, while HyperNetworks have model DoF and architecture tightly coupled together, a highly restrictive limitation.

3. Weights generated by HyperNetworks may be coupled and not optimized for different layers. HyperNetworks use only one weight generation network parameterized by hyper-weight to generate all primary network weights. This may not be optimal as different layers of the primary network may need different weight generation networks. Additionally, matrix multiplication is used for generating weights, and the generated primary network weights may be coupled. On the other hand, RPG has destructive weight sharing, which improves the network performance by decoupling cross-layer network weights. We will add these results and discussions in the revision to clarify the differences between RPG and HyperNetworks.