Revisiting Orthogonality Regularization: A Study for Convolutional Neural Networks in Image Classification

TAEHYEON KIM1 and SE-YOUNG YUN2

1Kim JaeChul Graduate School of AI, KAIST, Seoul, South Korea (e-mail: potter32@kaist.ac.kr)
2Kim JaeChul Graduate School of AI, KAIST, Seoul, South Korea (e-mail: yunseyoung@kaist.ac.kr)

Corresponding author: TAEHYEON KIM (e-mail: potter32@kaist.ac.kr).

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program(KAIST)). This work was also conducted by Center for Applied Research in Artificial Intelligence (CARAI) grant funded by DAPA and ADD (UD190031RD).

ABSTRACT Recent research in deep Convolutional Neural Networks (CNN) faces the challenges of vanishing/exploding gradient issues, training instability, and feature redundancy. Orthogonality Regularization (OR), which introduces a penalty function considering the orthogonality of neural networks, could be a remedy to these challenges but is surprisingly not popular in the literature. This work revisits the OR approaches and empirically answer the question: Even when comparing various regularizations like weight decay and spectral norm regularization, which is the most powerful OR technique? We begin by introducing the improvements of various regularization techniques, specifically focusing on OR approaches over a variety of architectures. After that, we disentangle the benefits of OR in the comparison of other regularization approaches with a connection on how they affect norm preservation effects and feature redundancy in the forward and backward propagation. Our investigations show that Kernel Orthogonality Regularization (KOR) approaches, which directly penalize the orthogonality of convolutional kernel matrices, consistently outperform other techniques. We propose a simple KOR method considering both row- and column- orthogonality, of which empirical performance is the most effective in mitigating the aforementioned challenges. We further discuss several circumstances in the recent CNN models on various benchmark datasets, wherein KOR gains more effectiveness.

INDEX TERMS Deep Neural Network (DNN), kernel, orthogonality regularization, Convolutional Neural Network (CNN), regularization, image classification.

I. INTRODUCTION

Recent Convolutional Neural Networks (CNN) have evolved into more in-depth and broader structures to get more accurate results [1], [2]. However, stacking many layers causes several training issues such as vanishing/exploding gradient [3] and over-parameterization issues [4]. There has been a lot of literature for mitigating these issues such as the weight decay [5], Dropout [6], and skip connections [7]. Such training procedure refinements in loss functions and optimization methods play a major role in a solution while barely change the computational complexity.

Orthogonality Regularization (OR) is one of the most potent techniques in contributing to these advancements by enforcing weight matrices \( W \) to be semi-orthogonal (i.e., \( WW^\top = I \) or \( W^\top W = I \) where \( I \) is the identity matrix). The orthogonality on linear transformations can preserve the energy [8] and reduce the redundancy of the model’s filter responses [9]. With the orthogonality initialization scheme, vanilla CNNs can be trained even with thousands of layers [10].

Orthogonality for convolutional layers can be injected into two different ways: (1) Kernel Orthogonality Regularization (KOR) which penalizes the gram matrix of convolutional kernel to be orthogonal and (2) Convolutional Orthogonality Regularization (COR) that strictly enforces the magnitude of the input feature and that of output feature to be same by using the Block-Toeplitz (DBT) matrix. Both works generally use the Frobenius norm [8], [9], [11], [12] as the penalty function. In KOR approaches, the spectral norm is also used as the regularization while it requires expensive computational costs [13]. To circumvent the computational...
complexity issue, an efficient algorithm of KOR, called Spectral Restricted Isometry Property (SRIP), is proposed; the algorithm utilizes the power method with a shift in the weight matrix [14]. Notably, in NeurIPS2019 MicroNet Challenge, team KAIIST uses a variant of SRIP ranked 2nd and 3rd [15] in CIFAR-100 track, that facilitates the generalization without additional resources such as the parameter storage and computational complexity.

Despite its superiority, OR approaches have not attracted attention and have rarely been discussed by a few papers. In the literature, they generally show their superiority through model accuracy improvement without examining a collection of other regularization methods such as weight decay [5] and spectral norm regularization [16] under the same training settings. Furthermore, it is rarely comprehensively studied on the vanishing/exploding gradient issue, training instability, and feature redundancy compared to other regularization techniques.

Contribution In this paper, we examine OR methods along with a collection of regularization techniques which do not require additional resources for various architectures. In particular, we shed light on the impact of using OR methods for training deep CNN models. The contributions of this paper are as follows:

- We empirically observe that KOR approaches consistently outperform the other regularization techniques defined on model parameters. In particular, our proposed simple method, referred to as SRIP+, which is a modified version of SRIP considering the row-orthogonality of weight matrices, is slightly better than the other OR methods.

- We demonstrate that SRIP+ implicitly improves the norm preservation effects and mitigates the filter redundancy so that the models have more gains on training networks. We show this positive effect by analyzing the magnitudes of layer-wise responses [8], spectral norm of the convolutional kernel [16], the row-orthogonality of the kernel matrix [9], and the weight of the batch normalization layers [17].

- We investigate a bag of benefits of the SRIP+ method on parameter initialization, training acceleration, the simple modulation of SRIP+ method considering the structures of CNN models, loss landscape, and representation qualities. Especially, with such simple modulations, we can gain more effectiveness from SRIP+ on recent mobile-friendly architectures such as MobileNets [18], [19] and EfficientNet [20]. The experiments are conducted on the CIFAR10, CIFAR100, TinyImageNet, and ImageNet datasets, and the proposed approach improves the accuracy by up to $4.89\%$, $+10.85\%$, $+9.1\%$, and $+0.89\%$ on average, respectively.

The rest of the paper is organized as follows. After describing the preliminaries for the various regularization methods (Section II), we describe our method and experimental settings (Section III). In Section IV, we discuss our findings and the empirical evidence for each. We describe the related works in Section V. Section VI concludes the paper.

II. PRELIMINARIES

A. USEFUL DEFINITION

In this paper, we denote by function $f_\theta$ a CNN parameterized by $\theta$ and the objective function $L$. Here $f_\theta$ consists of multiple nonlinear functions, and each nonlinear function is usually divided into three parts. One is a linear transformation part with weight matrix $W_l$, the next is a non-linearity part with an element-wise activation function $\phi$ such as ReLU, and the other is a normalization part $N_l$ such batch normalization (BN). Here $l \in \{1, 2, ..., L\}$ indicates the index of the layer. Under this notation, the learnable parameters are $\theta = \{W_l, N_l| l = 1, 2, ..., L\}$. Let us denote the $l$-th layer’s output $y_l$, then the forward propagation may be derived as follows: $y_{l+1} = \phi(N_{l+1}(\text{Conv}(W_{l+1}, y_l)))$. We denote the set of weight matrices by $W = \{W_1, \ldots, W_L\}$. We may define the notion of others as follows:

Orthogonality/orthonormality The two non-zero vectors $u, v \in \mathbb{R}^d$ are orthogonal when $u^\top v = 0$ and are orthonormal when $u^\top v = 0$ and $\|u\| = \|v\| = 1$ where $\cdot \| \|$ denotes the Euclidean norm.

Semi-orthogonal We call a weight matrix $W \in \mathbb{R}^{m \times n}$ as semi-orthogonal when $WW^\top = I$ or $W^\top W = I$, i.e., either the column vectors or the row vectors form an orthonormal set.

Matrix-norms In this paper, we frequently use two different matrix norms, the Frobenius norm and the spectral norm. For any matrix $A \in \mathbb{R}^{m \times n}$, the Frobenius norm and the spectral norm are defined as follows:

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}^2}$$ and $$\|A\|_2 = \sup_{x \in \mathbb{R}^n: \|x\| = 1} \|Ax\|,$$

where $a_{ij}$ is the $(i, j)$ element of $A$.

B. KOR

KOR is a class of regularization methods that make the weight matrices of a model almost semi-orthogonal. Note that, to apply the KOR, it is generally necessary to reshape each convolutional operator into a matrix $W \in \mathbb{R}^{m \times n}$ with $m = C_{out}$ and $n = S \times H \times C_{in}$, where $S, H, C_{in}, C_{out}$ are the filter width, filter height, the number of input channels, and the number of output channels, respectively (Figure 1).

The evolution history is aligned with the matrix norm. For instance, one [8] adds the following form, referred to as Soft Orthogonality (SO), to the main objective $L(\{(x_i, y_i)\}_{i=1}^{m}; \theta)$:

$$\frac{\lambda}{|W|} \sum_{W \in \mathcal{W}} \|W^\top W - I_n\|_F^2,$$ \hspace{1cm} (1)

where $x_i$ and $y_i$ are the input data and the corresponding label, $\lambda$ is the regularization coefficient, $I_n$ is an identity matrix, and $|W|$ is the cardinality of $\mathcal{W}$. On the other side, in [14], SRIP regularization replaces the Frobenius norm distance with the spectral norm. The spectral norm is computationally...
D. RELATED REGULARIZERS

Both SO and SRIP have their counterparts without considering the orthonormality. Their counterparts only consider the Frobenious norm and the spectral norm of the weight matrices rather than the difference $W^T W - I$.

**Weight Decay (WD)** WD is a common technique for regularizing the model parameters with the Frobenious norm:

$$
\lambda \sum_{W \in W} \|W\|_F^2
$$

**Spectral Norm Regularization (SNR)** In [16], the authors showed that the generalization error of CNN models decrease when applying SNR instead of the Frobenious norm (WD):

$$
\lambda \sum_{W \in W} \sigma(W)
$$

where the spectral norm is approximated similarly to SRIP.

III. METHOD AND EXPERIMENTAL SETTINGS

A. METHOD

Both SO and SRIP aim to make $W$ have orthonormal columns by using regularization functions $\|W^T W - I\|_F$ and $\sigma(W^T W - I)$ (an approximator of $\|W^T W - I\|_2$), respectively. When the number of rows is larger than the number of columns, i.e., $m < n$, the regularization functions could not be adequate to achieve the semi-orthogonality since the rows, instead of columns, have to be orthonormal. Fortunately, SO does not need to consider whether $m < n$ or not. Always, $\|W^T W - I\|_F^2$ is minimized to $\max\{0, m - n\}$ if and only if $W$ is semi-orthogonal [9] (Proposition 1). However, SRIP has to change the regularization function when $m < n$. All the matrices satisfying $\|W\|_2 \leq \sqrt{2}$ minimize $\|W^T W - I\|_2$ to 1 when $m < n$. Thus, SRIP just acts like SNR, restricting the spectral norms (Proposition 2).

**Proposition 1.** The row orthogonality and column orthogonality are equivalent in the Frobenious sense [9], i.e., $\|W^T W - I\|_F^2 = \|W^T W - I\|_2^2 + C$, where $C$ is a constant $m - n$.

**Proposition 2.** Let $W \in \mathbb{R}^{m \times n}$ ($m < n$) be a real matrix under the $n$-RIP condition. Then, $\|W^T W - I\|_2 = 1$ and $\|WW^T - I_m\|_2 = \sup_{z \in \mathbb{R}^n, \|z\|_2 = 1} \|Wz\|_2^2 - 1 \leq \sup_{z \in \mathbb{R}^n, \|z\|_2 = 1} \|Wz\|_2 \delta W$ where $\|Wz\|_2^2 - 1 \leq \delta W$ from the definition of $n$-RIP condition.

We thus propose a simple variant of SRIP, which is referred to as SRIP+, to make only semi-orthogonal $W$ be a solution as in SO. To this aim, we simply consider the row-orthogonality $\sigma(WW^T - I_m)$ when $m < n$ instead of the column-orthogonality. We also modify the power of $\sigma$ in SRIP from 1 to 2 (Eq. 7) so that the penalty function becomes more significant as $W$ goes further away from semi-orthogonal.

C. CONVOLUTIONAL ORTHOGONALITY REGULARIZATION (COR)

Recently, instead of KOR, [9], [12] proposed a direct constraint that preserves the norm between the $l$-th layer output $y_l$ and $x_{l+1}$ which is the output of the linear transformation. They enforced the orthogonality of the Doubly Block-Toeplitz (DBT) matrix $\mathcal{K}_{l+1}$ directly, i.e., $x_{l+1} = \mathcal{K}_{l+1} y_l$ (Eq. 4):

$$
\frac{\lambda}{|W|} \sum_{W \in W} \|\mathcal{K}_{l+1}^T \mathcal{K}_{l+1} - I_m\|_F^2,
$$

DBT matrix $\mathcal{K}$ is different from $W$ which is not an actual linear operator during propagation. [12] showed that KOR makes a network *isometry* partially, while COR enforces the norm preservation property more strictly in most cases except for when the kernel size and stride of the convolutional filter is equal to 1.
B. EXPERIMENTAL SETTINGS

CIFAR We train all models using the standard pytorch SGD optimizer with a nesterov momentum of 0.9 for 600 epochs [21] for three times. We use the cosine annealing scheduler to control the learning rate, whose initial learning rate is 0.1 and the minimum learning rate is 0.0005 with batch size 128 [22]. We use dropout of 0.3, and $l_2$ WD 1e-5. All convolutional layers use batch-normalization layers with average decay of 0.99. We did not apply any data augmentation procedures. Other than those mentioned, the training settings covered in the original papers [18]–[20] are used.

TinyImageNet / ImageNet We train the MobileNetV2, MobilenetV3, and EfficientNet-B0 to verify our method. First, for the TinyImageNet, we use the same training recipe with the CIFAR setting. Next, for the ImageNet, we use the cosine annealing scheduler on the learning rate decay whose initial learning rate is 0.025 and the minimum learning rate is 0.00005 with batch size 256. In the first 5 steps, we increase the learning rate with gradual warmup from 0 to 0.025. Finally, other training settings covered in the original papers [18]–[20] are used.

IV. DISENTANGLING THE BENEFITS OF OR

A. COLLECTION OF DIFFERENT REGULARIZATIONS

As a survey, we compare recent regularization algorithms for the prior distribution of model parameters (Table 2). Here, we do not apply the WD when using the regularizations to see their intrinsic effects while some regularizations are simultaneously used with WD in their original papers. As shown in Table 2, penalizing the KOR approaches consistently improves model performance better than WD, SNR and COR. Furthermore, SRIP$^+$ among KOR families is the best method in the survey while the difference is marginal. We observe the same order in all the evaluated models. For training acceleration, as pointed out in [14], we empirically observe that KOR methods also enable faster acceleration in the early stages of training (Figure 2) than other regularization methods.

B. NORM PRESERVATION EFFECTS

In this subsection, we investigate why performance gains remain when deploying the KOR approaches (Table 2). Hence, we attempt to answer the following question:

What benefit do such regularizations pose?

At first glance, we investigate how such regularizations enables isometric learning during the forward dynamics [8]. Isometric property means that each layer of the network preserves the inner product for both forward and backward propagation [12]. In [10], [12], they demonstrated that the isometric property may stabilize the network activations. For the evaluation, we measure the ratio between magnitudes of forward and backward propagation in a layer-wise manner, i.e., $\frac{\|x_{l+1}\|}{\|y_l\|}$. Analogously, in [10], [16], the authors clarified that the spectral norm is also related to the network activations. For example, let us look at the course of forward propagation in a layer-wise manner, i.e., $x_{l+1} = \mathcal{K}_{l+1}y_l$. Based on the definition of the spectral norm,

$$\frac{\|x_{l+1}\|}{\|y_l\|} \leq \|\mathcal{K}_{l+1}\|_2 \tag{8}$$

As derived in Eq. 8, the spectral norm controls the tightness of fluctuation of forward propagation, even for the backward propagation in a similar manner. Depending the values of this bound it suggests that the signal in the process of propagation is affected by the spectral norm; if $\|\mathcal{K}\|$ goes to 0, the vanishing gradient issue occurs; otherwise, if it goes to $\infty$, an exploding gradient issue occurs [16].

Figure 3a shows the box plot regarding how the norm of network activations is preserved over all layers. As shown in Figure 3a, all regularizations attempt to keep the ratio close to

| Dataset       | Resolution | Train data | Test data | Classes |
|---------------|------------|------------|-----------|---------|
| CIFAR10       | 32x32      | 50,000     | 10,000    | 10      |
| CIFAR100      | 32x32      | 50,000     | 10,000    | 100     |
| TinyImageNet* | 64x64      | 100,000    | 10,000    | 200     |
| ImageNet†     | 224x224    | 1,280,000  | 50,000    | 1,000   |

Table 1. Experiment datasets. For ‘*’ denoted dataset, we evaluated with the validation dataset.

| Method      | RN-50       | RN-56†      | RN-110†     | WRN-16-2  | WRN-28-4  |
|-------------|-------------|-------------|-------------|-----------|-----------|
| WD          | 93.31±0.18  | 93.11±0.15  | 93.54±0.21  | 93.82±0.11 | 94.92±0.14 |
| SNR         | 92.15±0.08  | 92.07±0.12  | 92.21±0.13  | 92.51±0.23 | 93.88±0.17 |
| COR         | 90.68±0.11  | 92.43±0.13  | 92.98±0.15  | 91.07±0.13 | 92.84±0.18 |
| SO          | 93.41±0.16  | 93.45±0.11  | 93.98±0.18  | 93.98±0.13 | 95.08±0.11 |
| SRIP        | 93.55±0.19  | 93.67±0.10  | 93.84±0.17  | 93.93±0.07 | 95.03±0.17 |
| SRIP$^+$    | 93.66±0.11  | 93.82±0.13  | 94.19±0.15  | 94.02±0.17 | 95.12±0.08 |

Table 2. Top1 accuracy of regularization algorithms on CIFAR10. † indicates the bottleneck structure, and RN and WRN are abbreviations of ResNet and Wide-ResNet, respectively.

1https://www.kaggle.com/c/tiny-imagenet
to 1, but the variance is significantly different. A network trained with WD and SNR has larger variance than other regularizations dealing with orthogonality and among them, COR is the best way to practice isometric learning. A similar consistency is also found in Figure 3b, which depicts the spectral norm of the DBT matrices. COR and all categories of KOR bound the spectral norm of the DBT matrices near 1.

Despite these consistency, the KOR approaches outperform the COR in our empirical studies. We think that this is because COR administers constraints that are strict compared to KOR approaches since it directly penalizes the DBT matrices. In a deep CNN model, most of filters seem to have advantages in norm preservation, but isometry learning rather obstructs the training of filters such as a downsampling layer whose stride is larger than 1, stem layer, etc. As evidence, when deploying the COR and WD together, we can improve the performance at a similar level of SRIP+ while the norm-preserving behavior of COR resembles that of KOR approaches. Among the KOR methods, the spectral norm performs better than the Frobenius norm, and we think this is also a similar reason that SO strictly limits the set of parameters in an orthogonal space, while SRIP and SRIP+ enable them in a wide range with a spectral norm of 1.

### C. FILTER REDUNDANCY

So far we have provided empirical evidence that the benefits of KOR are primarily caused by isometric properties. We now investigate further effects regarding how the filters in the layer (row vector of \( W \)) become as orthogonal as possible [9]. This implies that their filter responses are much less redundant, so that the model utilizes a larger capacity for the feature expressiveness. Here, we calculate the filter similarities using the absolute sum of the cosine similarities between row vectors in \( W \) (Eq. 9):

\[
\frac{1}{m(m-1)} \sum_{1 \leq i < j \leq m} \frac{W_{i,\cdot} \cdot W_{j,\cdot}}{\|W_{i,\cdot}\| \|W_{j,\cdot}\|}
\]

where \( W_{i,\cdot} \) is the \( i \)-th row vector of \( W \). Figure 3c shows the box plot for the magnitudes of Eq. 9 for all filters of ResNet-561. As shown in Figure 3c, KOR approaches and COR remove correlations among filters compared to WD and SNR and COR slightly lead to a more ideal situation.

We also focus on the channel importance of each filter through the BN weight \( \gamma \). [17] showed that since every \( \gamma \) multiplies a normalized random variable, the channel importance becomes comparable across different layers by measuring the magnitude values of \( \gamma \). When looking at the \( \gamma \) values in Figure 3d, SRIP+ seems to consider the the importance of all channels in each filter as compared to other methods. This means that a model trained with SRIP+ has fewer filters that are considered relatively redundant.

### D. SRIP+ AND ORTHOGONAL INITIALIZATION

We empirically observe that networks with the SRIP+ can be trained with Xavier [23] and Kaiming [1] initialization methods, and consequently it is important to note that with such initialization methods, the networks are not trained as the \( \lambda \) grows. Because the SRIP+ tries to change the optimization landscape, the initialization methods can no longer provide good initial points.

We demonstrate that the weight matrix cannot be trained depending on the initialization method. For example, when

Table 3. (Training accuracy / test accuracy) of a network after the first epoch of training on CIFAR10 when different initialization method is applied. We evaluate the robustness towards the \( \lambda \) of SRIP+ via the ResNet-561 model.

| \( \lambda \) | Xavier init [23] | Kaiming init [1] | Orthogonal init [24] |
|-------------|-------------------|------------------|---------------------|
| 1.0         | - / -             | - / -            | 35.37 \pm 1.11 / 41.40 \pm 2.21 |
| 0.5         | 33.51 \pm 1.61 / 46.55 \pm 6.19 | 31.70 \pm 1.66 / 44.24 \pm 4.06 | 34.01 \pm 1.91 / 47.73 \pm 2.88 |
| 0.1         | 31.18 \pm 2.87 / 43.09 \pm 2.42 | 30.00 \pm 1.51 / 43.69 \pm 2.64 | 31.83 \pm 1.44 / 44.58 \pm 2.28 |
| 0.01        | 25.88 \pm 3.46 / 39.98 \pm 4.04 | 23.37 \pm 4.90 / 37.20 \pm 6.73 | 28.29 \pm 2.54 / 41.63 \pm 2.70 |
the spectral norm of an initialized weight matrix is either very large or very close to zero, the SRIP\(^+\) term becomes so large that the network cannot be trained. [9], [10] methods can initialize all convolutional kernels to be isometry with delta dirac initialization, but it fails to calculate the approximated spectral norm since the denominator always remains zero (Eq. 3). To resolve this issue, we construct the weight matrix as a semi-orthogonal matrix at the beginning, which is known as the orthogonal initialization [24]. The initialization method proceeds as follows:

1. Make the matrix \( W \) with the normal distribution whose mean and variance is 0 and 1.

2. Compute the QR factorization and make Q uniform. For the overcomplete case \( W \in \mathbb{R}^{m \times n} (m < n) \), transpose the matrix before computing the QR factorization.

3. Initialize the weight matrix to Q and the bias \( b = 0 \).

Figure 3 shows our empirical study on the robustness of the SRIP\(^+\) with respect to \( \lambda \) when the neural networks are trained with the SRIP\(^+\). We evaluate this through the training and test accuracy after the first epoch. Orthogonal initialization consistently outperforms other initialization methods in terms of training and test accuracy after the first training epoch. Furthermore, when \( \lambda \) increases, a network initialized with either Xavier and Kaiming method faces the issue that a model cannot be trained (noted it as - in Figure 3), but the orthogonal initialization diminishes this adverse effect.

### E. SRIP\(^+\) MODULATION FOR DIFFERENT LAYERS

In this subsection, we investigate the improvements by applying the SRIP\(^+\) to different categories of layers considering the structure of CNNs. We categorize the layers as follows: a fully connected layer (FC), pointwise convolution, and other convolutional layers. Because pointwise convolutions require no effort for the flattened process before deploying SRIP\(^+\), we doubt that the pointwise convolution inherently possesses different characteristics compared to other convolutional layers. Moreover, we test the SRIP\(^+\) with a C-EfficientNet [15] model as well as a ResNet model on CIFAR10 because a C-EfficientNet model consists of depthwise separable convolution unlike the ResNet series.

As shown in Table 4, SRIP\(^+\) consistently creates positive synergies with all types of layers, but the most benefited filter is dependent on the model structure. Specifically, we can find that the pointwise convolution is the most promising in the model with bottleneck structure. Furthermore, for the depthwise separable convolutions, SRIP\(^+\) enforced to both depthwise and pointwise convolutions seem to hinder the generalization because we could obtain the accuracy as 93.81\(\pm\)0.14 in C-EfficientNet-1 when turning on the KOR to all layers except the convolutions whose kernel size is greater than 1. These results slightly outperform those of the deployment on all layers.

Table 4. Comparison of different layers on CIFAR10. \(^1\) indicates a model with a bottleneck structure. WD is plugged into layers that are not spectrally regularized.

| Layer        | ResNet-50   | ResNet-56\(^1\) | C-EfficientNet-1 |
|--------------|-------------|-----------------|------------------|
| WD           | 93.31\(\pm\)0.18 | 93.11\(\pm\)0.15 | 92.52\(\pm\)0.21 |
| 1 Conv       | 93.48\(\pm\)0.12 | 93.77\(\pm\)0.22 | 93.74\(\pm\)0.12 |
| 3 Conv       | 93.37\(\pm\)0.11 | 93.46\(\pm\)0.19 | 92.58\(\pm\)0.32 |
| FC           | 93.41\(\pm\)0.17 | 93.23\(\pm\)0.25 | 92.91\(\pm\)0.18 |
| All          | 93.66\(\pm\)0.11 | 93.82\(\pm\)0.13 | 93.71\(\pm\)0.21 |

To verify the validity of this simple modulation, we systematically evaluate the SRIP\(^+\) for various CNN models whose convolutional filters consist of pointwise and depthwise convolutional layers. We deploy the algorithm denoted as mSRIP\(^+\) that applies the SRIP\(^+\) to all layers except for the depthwise convolutional layers in the model. WD is not used for all layers. For architectures, we test this strategy on MobileNets [18], [19] and EfficientNet [20]. As shown in Table 5, we can consistently improve the performance via mSRIP\(^+\). Far Interestingly, in the MobileNet-like structures, SRIP\(^+\) significantly facilitates generalization compared to conventional CNN structures such as ResNet and Wide-ResNet (Table 5) even before modulation. This result provides an intuition regarding how the performance of lightweight CNN models can be dramatically improved when trained with KOR approaches.

To go further, such consistent improvements are also obtained in the vision-transformer networks [25], [26] (Table 6). Recently, there remain an increasing demand of designing a DNN with transformer layers for computer vision [25]–[28]. Although our paper mainly targets the CNN models, the experiment shown in Table 5 is simply applied to the Feed-Forward-Network (FFN) layers in the Transformer as a pilot. We observe that even applying mSRIP\(^+\) on FFN layers in the Transformer block can lead to better optima (Table 6).

### F. SHARP AND FLAT OPTIMA

We visualize the loss surface along a random direction near the optimal parameters referred to in [29]. This tool utilizes a filter normalization technique. Figure 4 depicts the 2D loss contours of ResNet-50 models. As shown in Figure 4, the optimum of networks trained with WD has a relatively sharp curvature, while that of KOR families has a flatter minimum.

We believe that this is also a consequence of Figure 3. Because networks trained with WD have unstable network activations during propagation, the network is easily corrupted by noise on weight, so that the loss landscape becomes sharp. On the other hand, the KOR series has a flatter optimum.

### G. SHRINKAGE EFFECTS ON PENULTIMATE LAYER REPRESENTATIONS

In this subsection, we visualize the activations of the penultimate layer by the tools of [30] (Figure 5). This is to find out how energy stabilization affects the representation distribution. In detail, it depicts in 2-D via the linear projection how the activations of three classes cluster around
the template that represents the representative feature vector of corresponding classes such as the row vector of fully-connected layer. Here, we show visualization results for the classes “airplane”, “automobile”, and “bird” on CIFAR10. The first column represent examples of training dataset and the second column represent those of test dataset. Training accuracies of all models used in the visualization are almost 99.99% (Figure 5).

As shown in Figure 5, the clusters of all representations are definite, but the cohesion and separation of each method’s clusters is clearly different. We observe that networks trained with SRIP+ exhibit a decrease in both inter- and intra-cluster divergence, respectively. Namely, their clusters are more dense and well separated. In both visualizations of training and test dataset, their clusters from networks trained with SRIP+ form almost equilateral triangle, whereas the triangular structure is less clear in the case of the networks trained with WD.

V. RELATED WORKS

A. KOR

Orthonormality, which is applied in linear transformations between layers of networks, is one of the most interesting debates in the optimization field [8], [11], [24]. This technique can guarantee that the activation energy is not amplified [16] while spectrally regularizing the weight matrices of a network. Therefore, it can stabilize the distribution of network activations [8], [11]. Many researchers have attempted

| Dataset | Network          | WD           | SRIP+ | mSRIP+ | # Params |
|---------|------------------|--------------|-------|--------|----------|
| C10/C100| C-EfficientNet-1 | 92.52±0.21 / 68.79±0.81 | 93.71±0.25 / 73.11±0.39 | 93.81±0.19 / 73.24±0.19 | 0.22 M |
|         | C-EfficientNet-2 | 93.44±0.25 / 71.74±0.54 | 95.08±0.24 / 77.39±0.31 | 95.21±0.24 / 77.54±0.32 | 0.39 M |
|         | C-EfficientNet-3 | 93.60±0.24 / 73.26±0.51 | 95.43±0.20 / 78.88±0.17 | 95.60±0.17 / 79.11±0.19 | 0.55 M |
| MobileNetV2 | 83.72±1.02 / 52.57±1.20 | 86.98±0.38 / 61.57±0.88 | 87.14±0.87 / 60.38±1.10 | 87.10±0.41 / 61.76±0.78 | 2.3 M |
| MobileNetV3 | 80.47±1.10 / 45.17±1.08 | 86.85±0.36 / 63.18±0.87 | 87.01±0.30 / 63.32±0.61 | 4.2 M |
| EfficientNet-B0 | 81.19±0.76 / 56.40±1.41 | 54.34±0.24 / 72.94 | 54.42±0.53 / 72.98 | 2.4 M / 3.4 M |
|         | MobileNetV2     | 46.39±0.41 / 72.00 | 52.87±1.31 / 75.91 | 53.21±1.01 / 76.09 | 4.4 M / 5.4 M |
|         | MobileNetV3     | 47.89±0.67 / 75.20 | 54.05±0.83 / 76.87 | 54.33±0.67 / 77.10 | 4.3 M / 5.3 M |
|         | EfficientNet-B0 | 47.03±0.51 / 76.30 | 54.50±0.24 / 76.94 | 54.60±0.53 / 77.04 | 2.5 M / 3.5 M |

Table 5. Test accuracy on CIFAR10, CIFAR100, TinyImageNet and ImageNet. C10, C100, TIN and IN indicate the CIFAR10, CIFAR100, TinyImageNet and ImageNet, respectively. mSRIP+ indicates the SRIP+ with our modulations for the categories of layers and initialization. The models marked “±6” are trained from scratch. Models trained on IN are run once.

| Network          | WD | SRIP+ | mSRIP+ | # Params |
|------------------|----|-------|--------|----------|
| ViT-Small-Patch16 [25] | 81.40 | 81.45 | 81.53 | 22.1M |
| DeiT-Small-Patch16 [26] | 81.20 | 81.31 | 81.33 | 22.4M |

Table 6. Test accuracy on ImageNet. WD is plugged into layers that are not spectrally regularized. Training recipes are followed in [25], [26].

Figure 4. 2-D visualization of solutions for the ResNet-561 model on CIFAR-10 by using different regularization methods [29]. Each line in contour indicates the loss values. The more sparse the spacing between the lines in the contour, the more robust the weight is to noise, indicating that it has a flatter minima.

Figure 5. Visualization of penultimate layer’s activations of ResNet-561 on CIFAR10. (first column) the visualization of training dataset, and (second column) that of test dataset. Each blue, orange, green color indicates the classes “airplane”, “automobile”, and “bird”.

(a) Training w/ WD
(b) Test w/ WD
(c) Training w/ SRIP+ (d) Test w/ SRIP+
to utilize the orthogonality for training, for example, by regularizing the singular values of each weight matrix in a narrow band around the value near 1 [13] or by penalizing the gram matrix of each weight matrix to be identity [8], [31] or by orthogonal initialization [8], [24]. Various types of orthogonality regularization tools were proposed [14], and the authors empirically found the best among them.

**B. COR**

[10] provided the conditions for dynamical isometry and the convolution operator that can preserve the norm, but this method is limited to represent a subset of orthogonal space. Recently, [9], [12] proposed the same COR methods by regularizing the orthogonality of the DBT matrix. In [12], the authors showed that the isometry property can train deep vanilla networks without normalization nor skip connections. In [9], the performance of the existing networks was improved. However, neither work stand alone for training deep networks without WD.

**VI. CONCLUSION**

In this paper, we disentangle the benefits of the OR methods with exhaustive studies. We empirically observe that the OR series based on the penalty function of convolutional kernel weight matrices, called KOR, facilitate generalization with norm preservation property during the propagation, and such methods are sufficient for training deep convolutional neural networks even without WD. Among those, our simple method SRIP⁺ further improves the performance of various models, specifically MobileNet-like models with a simple modulation on the layer to be regularized. We verify these findings on various benchmark dataset. Furthermore, we stabilize the SRIP⁺ training via robust orthogonal initialization. As a qualitative study, we empirically observe that OR methods lead to flat optima. We believe that such OR approaches, especially SRIP⁺, should be considered as a cornerstone method.

**References**

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[2] S. Zagoruyko and N. Komodakis, “Wide residual networks,” CoRR, vol. abs/1605.07146, 2016. [Online]. Available: http://arxiv.org/abs/1605.07146

[3] B. Hanin, “Which neural net architectures give rise to exploding and vanishing gradients?” in *Advances in neural information processing systems*, 2018, pp. 582–591.

[4] Z. Allen-Zhu, Y. Li, and Y. Liang, “Learning and generalization in over-parameterized neural networks, going beyond two layers,” in *Advances in neural information processing systems*, 2019, pp. 6158–6169.

[5] A. Lewkowycz and G. Gur-Ari, “On the training dynamics of deep networks with $l_2$ regularization,” arXiv preprint arXiv:2006.08643, 2020.

[6] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] D. Xie, J. Xiong, and S. Pu, “All you need is beyond a good init: Exploring better solution for training extremely deep convolutional neural networks with orthornormalization and modulation,” CoRR, vol. abs/1703.01827, 2017. [Online]. Available: http://arxiv.org/abs/1703.01827

[9] J. Wang, Y. Chen, R. Chakraborty, and S. X. Yu, “Orthogonal convolutional neural networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 505–11 515.

[10] D. Xie, J. Xiong, and S. Pu, “Orthogonal weight normalization: Solution to optimization over multiple dependent stiefel manifolds in deep neural networks,” CoRR, vol. abs/1709.06079, 2017. [Online]. Available: http://arxiv.org/abs/1709.06079

[11] L. Huang, X. Liu, B. Lang, A. W. Yu, and B. Li, “Orthogonal weight normalization: Solution to optimization over multiple dependent stiefel manifolds in deep neural networks,” CoRR, vol. abs/1709.06079, 2017. [Online]. Available: http://arxiv.org/abs/1709.06079

[12] H. Qi, C. You, X. Wang, Y. Ma, and J. Malik, “Deep isometric learning for visual recognition,” in *International Conference on Machine Learning*, PMLR, 2020, pp. 7824–7835.

[13] K. Jia, “Improving training of deep neural networks via singular value bounding,” CoRR, vol. abs/1611.06013, 2016. [Online]. Available: http://arxiv.org/abs/1611.06013

[14] N. Bansal, X. Chen, and Z. Wang, “Can we gain more from orthogonality regularizations in training deep cnns?” CoRR, vol. abs/1810.09102, 2018. [Online]. Available: http://arxiv.org/abs/1810.09102

[15] T. Kim, J. Kim, and S. Yun, “Efficient model for image classification with regularization tricks,” in *Proceedings of the 2019 Competition and Demonstration Track, ser. Proceedings of Machine Learning Research*, H. J. Escalante and R. Hadsell, Eds., vol. 123. Vancouver, CA: PMLR, 08–14 Dec 2020, pp. 13–26. [Online]. Available: http://proceedings.mlr.press/v123/kim20a.html

[16] Y. Yoshida and T. Miyato, “Spectral norm regularization for improving the generalizability of deep learning,” 2017.

[17] J. Yu, X. Lu, Z. Liu, and Z. Wang, “Rethinking the smaller-norm-less informative assumption in channel pruning of convolution layers,” arXiv preprint arXiv:1802.00124, 2018.

[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[19] M. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan et al., “Searching for mobilenetv3,” arXiv preprint arXiv:1905.02244, 2019.

[20] M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” arXiv preprint arXiv:1905.11946, 2019.

[21] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, “On the importance of initialization and momentum in deep learning,” in *Proceedings of the 30th International Conference on International Conference on Machine Learning*, ser. ICML ’13. JMLR.org, 2013, pp. III–1139–III–1147. [Online]. Available: http://dl.acm.org/citation.cfm?id=3042817.3043064

[22] I. Loshchilov and F. Hutter, “SGDR: stochastic gradient descent with restarts,” CoRR, vol. abs/1808.03983, 2016. [Online]. Available: http://arxiv.org/abs/1608.03983

[23] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[24] A. M. Saxe, J. L. McClelland, and S. Ganguli, “Exact solutions to the nonlinear dynamics of learning in deep linear neural networks,” arXiv preprint arXiv:1312.6120, 2013.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” arXiv preprint arXiv:2010.11929, 2020.

[26] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” in *International Conference on Machine Learning*, PMLR, 2021, pp. 10 347–10 357.

[27] A. Steinr, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, “How to train your vit? data, augmentation, and regularization in vision transformers,” arXiv preprint arXiv:2106.10270, 2021.

[28] Z. Dai, H. Liu, Q. V. Le, and M. Tan, “Coatnet: Marrying convolution and attention for all data sizes,” arXiv preprint arXiv:2106.04803, 2021.
[29] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, “Visualizing the loss landscape of neural nets,” in Neural Information Processing Systems, 2018.

[30] R. Müller, S. Kornblith, and G. E. Hinton, “When does label smoothing help?” CoRR, vol. abs/1906.02629, 2019. [Online]. Available: http://arxiv.org/abs/1906.02629

[31] R. Balestriero and richard baraniuk, “A spline theory of deep learning,” in Proceedings of the 35th International Conference on Machine Learning, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. Stockholmsmässan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 374–383. [Online]. Available: http://proceedings.mlr.press/v80/balestriero18b.html

APPENDIX A DETAILS OF ORTHOGONALITY EQUIVALENCE ON SPECTRAL NORM IN ??

Lemma 1. Let \( W \in \mathbb{R}^{M \times N} \) be a real matrix. Then, for given orthogonal matrices \( P \in \mathbb{R}^{M \times M} \) and \( Q \in \mathbb{R}^{N \times N} \). Then,
\[
\|PWQ\|_2 = \|W\|_2
\]

Proof. This can be derived from the orthogonal invariance of the 2-norm. Orthogonal matrix \( P \) preserves the vector 2-norm for any real vector \( x \in \mathbb{R}^M \),
\[
\|Px\|_2 = \sqrt{x^T P^T Px} = \sqrt{x^T x} = \|x\|_2
\]

Therefore, from the Eq. equation 11,
\[
\|PWQ\|_2 = \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \|PWQx\|_2 \\
= \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \|P(WQx)\|_2 \\
= \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \|WQx\|_2 \\
= \|Q^TW\|_2
\]

Similarly, \( \|Q^TW\|_2 = \|W\|_2 \).

Lemma 2. Let \( A \in \mathbb{R}^{N \times N} \) be a diagonal matrix, i.e., \( A = diag(\lambda_1, \lambda_2, \ldots, \lambda_N) \). Then,
\[
\|A\|_2 = \max |\lambda_i|
\]

Proof.
\[
\|A\|_2 = \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \|Ax\|_2 \\
= \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \sqrt{\sum_{i=1}^{n} \lambda_i^2 x_i^2} \quad (12)
\]
\[
\leq \max |\lambda_i| \sup_{x \in \mathbb{R}^N : \|x\|_2 = 1} \sqrt{\sum_{i=1}^{n} x_i^2} = \max |\lambda_i| \quad (16)
\]

As a next step, we provide a proof of the spectral norm equivalence, i.e., \( \|WW^T - I_M\|_2 = \|W^TW - I_N\|_2 \). Recall the RIP condition of \( W \):

Lemma 3. Let \( W \in \mathbb{R}^{M \times N} (M > N) \) be a real matrix. Then,
\[
\|WW^T - I_M\|_2 \geq \|W^TW - I_N\|_2 \quad (15)
\]

Proof. We can decompose the \( W \) with singular value decomposition as follows:
\[
W = U \Sigma V
\]
We can derive the left handside of Eq. equation 15.
\[
\|WW^T - I_M\|_2 = \|U \Sigma V V^T \Sigma U^T - I_M\|_2 \\
= \|U \Sigma \Sigma^T U^T - I_M\|_2 \\
= \|U \Sigma \Sigma^T U^T - UI_M U^T\|_2 \\
= \|U (\Sigma \Sigma^T - I_M) U^T\|_2 \\
= \|\Sigma \Sigma^T - I_M\|_2 (\text{from lemma 1})
\]

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
The right side hand of Eq. equation 15 is similarly derived as follows:

\[ \| W W^T - I_N \|_2 = \| V^T \Sigma^T U^T U \Sigma V - I_N \|_2 \]

\[ = \| V^T \Sigma^T \Sigma V - I_N \|_2 \]

\[ = \| V^T (\Sigma^T \Sigma - I_N) V \|_2 \]

\[ = \| \Sigma^T \Sigma - I_N \|_2 \text{(from lemma 1)} \]

Then, we can have the following inequality based on lemma 2:

\[ \| W W^T - I_M \|_2 = \| W W^T - I_N \|_2 \]

\[ \leq \max \{ \| W W^T - I_N \|_2, 1 \} \]

(18)

(from lemma 2)

\[ \geq \| W^T W - I_N \|_2 \]

Therefore,

\[ \| W^T W - I_N \|_2 \geq \| W^T W \|_2 - \| I_N \|_2 \]

\[ = \sup_{z \in \mathbb{R}^N : \| z \|_2 = 1} \| (W^T W - I_N) z \|_2 \]

(19)

As a result, \( \| W^T W - I_N \|_2 = \) both \( \| W W^T - I_M \|_2 \) and \( \| W^T W - I_N \|_2 \) are equal to \( \delta_W \geq 0 \).

Analogously, lemma 3 can be shown when \( M < N \) case, i.e., let denote the singular values of \( W \) by \( \sigma_1(W) \geq \sigma_2(W) \geq \cdots \geq \sigma_{min(M,N)}(W) \). Then, the largest value of either \( \Sigma^T \Sigma \) or \( \Sigma \Sigma^T \) is \( \sigma_1^2(W) \) and since the Proposition 2 implies that \( \| W^T W - I_n \|_2 \) may be flawed for reasons: a network may not train the semi-orthogonality of \( W \) via backpropagation since \( \| W^T W - I_n \|_2 \) is always equal to 1 regardless of the \( W \) when \( m < n \). Furthermore, in such case, each column vector of \( W \) can not be orthonormal each other because its rank is at most \( m \).

As a proof, under the N-RIP condition,

\[ \| W^T \|_2^2 - 1 - \| z \|_2^2 = 1 \]

(20)

where \( z \) is \( N \)-sparse vector.

\[ \sup_{z \in \mathbb{R}^N : \| z \|_2 = 1} \| W^T \|_2^2 - 1 = \sup_{z \in \mathbb{R}^N : \| z \|_2 = 1} \| z^T W^T W z - z^T z \|

\[ = \sup_{z \in \mathbb{R}^N : \| z \|_2 = 1} \| z^T (W^T W - I_N) z \|

\[ = \sup_{z \in \mathbb{R}^N : \| z \|_2 = 1} \| z^T (\Sigma^T \Sigma - I_N) z \|

\[ = \| W^T W - I_N \|_2 \]

\[ \leq \delta_W < 1 \]

(21)

From lemma 3, \( \| W W^T - I_M \|_2 = 1 \).

***