Supplementary Material : Efficient Neural Network Compression

Hyeji Kim, Muhammad Umar Karim Khan, and Chong-Min Kyung
Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea
{hyejikim89, umar, kyung}@kaist.ac.kr

1. List of Symbols in the Paper

- I_l : number of input channels in l-th layer
- O_l : number of output filters in l-th layer
- D_l : filter window size in l-th layer
- W_l : width of output feature map in l-th layer
- H_l : height of output feature map in l-th layer

- r_l^{max} : initial maximum rank in l-th layer
- r_l : vector space of rank in l-th layer
- $r_{l,max}$: a maximum rank in vector space r_l
- r'_i : vector space of rank in a group of i-th layer for hierarchical space generation
- $r'_{i,max}$: a maximum rank in vector space r'_i

- R : a set of ranks of each layer
- R_o : a final selected rank configuration
- R_e : a rank configuration for which layer-wise rank metrics are equal for every layer
- R_{min} : a rank configuration for lower bound of combinatorial space
- R_{max} : a rank configuration for upper bound of combinatorial space
- $\mathbf{R}$: candidate rank configurations
- $\hat{\mathbf{R}}$: candidate rank configurations in min-offset space
- $\mathbf{R}_A$: selected top-N rank configurations for ENC-Inf

- $\sigma_l(d)$: d-th singular value after SVD on the parameters in l-th layer
- $\sigma'_l(r_l)$: PCA-energy for r_l; the accumulation of the first r_l diagonal entries for singular values after SVD
- $y_{p,l}$: layer-wise accuracy metric based on the PCA-energy for l-th layer
- $y_{m,l}$: layer-wise accuracy metric based on the measurement accuracy for l-th layer
- $A_p(R)$: network accuracy metric based on the PCA-energy
- $A_m(R)$: network accuracy metric based on the measured accuracy
- $A_c(R)$: network accuracy metric based on the combination of PCA-energy and measured accuracy
- f_{C-A} : a mapping function between complexity and accuracy
- f_{C-R} : a mapping function between complexity and rank configuration

- c_l : a coefficient of layer complexity in l-th layer
- c'_i : a coefficient of layer complexity in a group of i-th layer for hierarchical space generation
- $C_l(r_l)$: a complexity of l-th layer for r_l
- $C(R)$: total complexity of R
- C_{orig} : total complexity of original network model
- C_t : target complexity

- δ_s : space margin
- δ_m : complexity margin
- δ_d : average density of all vector spaces
- t_l : step size of rank in vector space r_l and r'_l
- $\mathcal{X}_l$: l-th sub-spaces

2. Low-rank Decomposition and Neural Network Compression

In CNN, the pre-trained parameters of a layer is represented by the 4-D tensor for convolutional layer and the 2-D matrix for fully-connected layer. This trained parameters can be separated into over two tensors by the tensor decomposition algorithms such as Truncated SVD, Tucker, and CP decomposition. The shape of 4-D parameter of l-th convolutional layer can be transformed by the two general strategies: (i) channel decomposition using (D_lxD_l) and (1x1) kernel window, and (ii) spatial decomposition using (D_lx1) and (1xD_l) kernel window.

As an example of spatial decomposition, a l-th convolution layer is separated into two layers with the maximum rank r_l^{max}, which makes the total number of parameters of decomposed kernels same as the original number of parameters. In our framework, we determine a rank configuration for the separated convolution and fully-connected layers. In the network compression, the decomposed kernels are truncated by the determined rank of each layer as illustrated in Fig. 1.

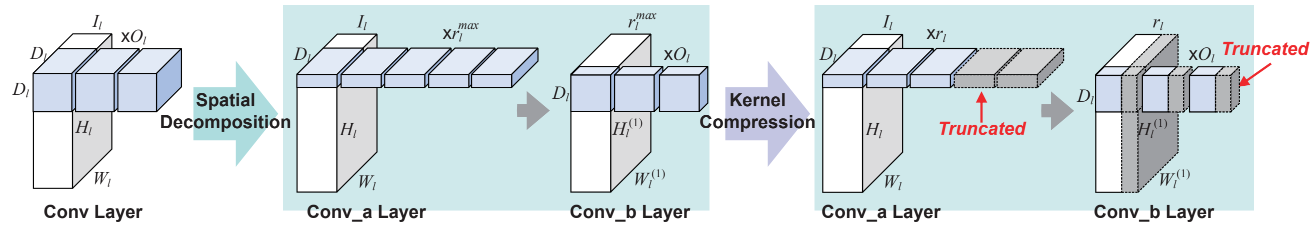

Figure 1. Spatial decomposition and compression of trained parameters. White box is the feature map, and blue box is the trained kernel parameters. After spatial decomposition, a 4-D kernel parameters in Conv layer is separated to two 4-D parameters in Conv_a and Conv_b layers. In the kernel compression, each 4-D kernel is truncated by r_l which means the number of filters is reduced by r_l in Conv_a layer, and the number of channels is reduced by r_l in Conv_b layer.

3. Initial Steps for a New CNN

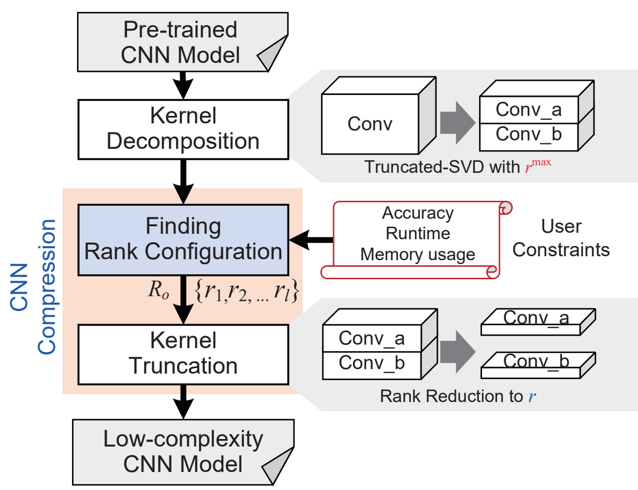

Figure 2. Overview of the proposed framework. After initial decomposition of pre-trained kernels, the proposed method generates a rank configuration corresponding to the number of filters for the separated layers. Then, the decomposed kernels are truncated by the rank r_l.

Following steps are performed :

1) Decompose each kernel by truncated-SVD with initial maximum rank r_l^{max}
   - Spatial decomposition : $r_l^{max} = I_l O_l D_l / (I_l + O_l)$
   - Channel decomposition : $r_l^{max} = I_l O_l D_l^2 / (I_l D_l^2 + O_l)$
2) Extract the partial data from the training dataset for the accuracy measuring
3) Define the layer-wise metric based on both PCA-energy $y_{p,l}(r_l)$ and accuracy measurement $y_{m,l}(r_l)$

4. Further Details of Combinatorial Space

The large number of candidate rank configurations $\mathbf{R}$ makes higher the probability of finding the optimal rank configuration. The number of $\mathbf{R}$ is directly related to the space density, which means how the unit size of rank in a layer is fine-grained. However, the higher space density and space volume incur the significant space generation time, since there are huge number of possible combinations of rank. In our paper, we overcome the huge generation time of combinatorial space from the hierarchical layer grouping.

4.1. Space density & Number of candidate rank configurations $\mathbf{R}$

There are three types of space parameters related to the space density, space volume, and number of candidate rank configurations $\mathbf{R}$. The basic effect of space parameters is follows:

- Decrease t_l:
  - (Pros) : it makes larger the number of candidate rank configurations $\mathbf{R}$
  - (Cons) : space generation takes significant time due to increasing space density
- Increase δ_s :
  - (Pros) : it makes larger the number of candidate rank configurations $\mathbf{R}$
  - (Cons) : space generation takes significant time due to increasing space volume
- Increase δ_m :
  - (Pros) : it makes larger the number of candidate rank configurations $\mathbf{R}$ without increasing space generation time
  - (Cons) : **deviation** from the target complexity is larger

In our experiments, we fix the space margin δ_s and complexity margin δ_m. The space density is only controlled by the step size of vector space t_l.

- δ_s (*fixed*) : space margin. Default = 10% of original total complexity
- δ_m (*fixed*) : complexity margin. Default = 0.5% of target complexity
- t_l (*variable*) : step size of vector space for the rank in l-th layer. Initial value = 1% of initial maximum rank

As an example of two-layered CNN, the candidate rank configurations $\mathbf{R}$ denoted in Fig. 3 as star points are lying on the linear function of C_t. Total complexity is the plane function, and a specific level of complexity is projected to the linear function on the rank dimensions.

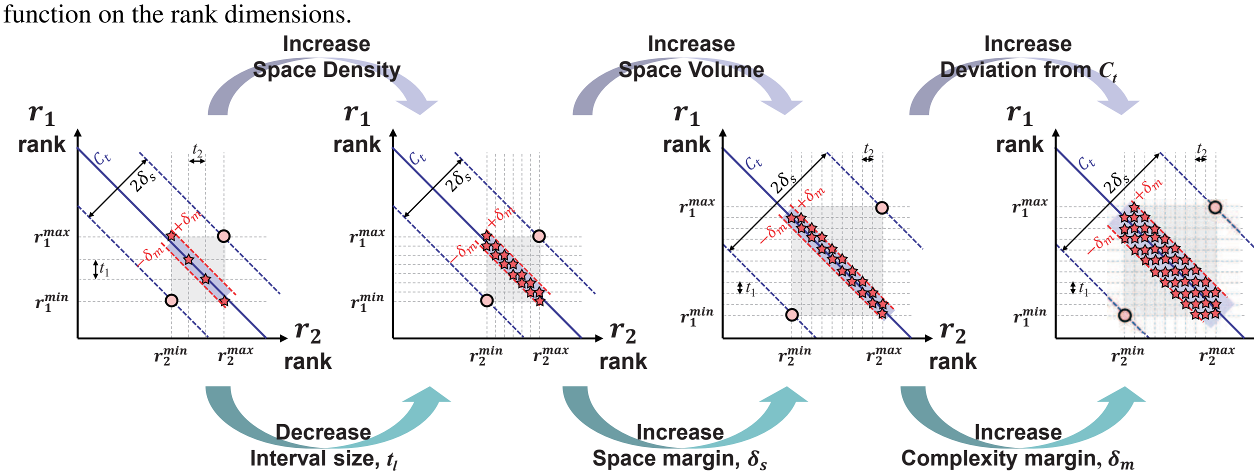

Figure 3. Dependency of space parameters $\{\delta_s, \delta_m, t_l\}$ on the number of candidate rank configurations $\mathbf{R}$.

4.2. Definition of the step size of rank t_l in vector space for l-th layer

We control the step size of rank t_l with the average density of all vector space, $\delta_d = \sum_{l=1}^{L} 1/(t_l L)$.

- t_l^0 : initial step size. $t_l^0 = round(r_l^{max} \times 0.01)$
- δ'_d : desired average density (hyper-parameter)
- α : scaling factor of initial step size t_l^0 to make the new t_l satisfying the δ'_d
  - Derived from :

$$\delta'_d \approx \alpha \times \sum_{l=1}^{L} \frac{1}{t_l^0 \times L} = \left(\frac{1}{t_1^0/\alpha} + \frac{1}{t_2^0/\alpha} + ... + \frac{1}{t_L^0/\alpha} \right) \times \frac{1}{L}$$

$$\alpha \approx \delta'_d \Big/ \left(\sum_{l=1}^{L} \frac{1}{t_l^0 \times L} \right) \quad \rightarrow \quad \text{initial } \alpha$$

$$t_l = \max(\text{round}(t_l^0/\alpha)\ ,\ 1) \quad \rightarrow \quad \text{integer value}$$

- From the initial step size t_l^0 and initial scaling factor α, we iteratively reduce α until the average density δ_d satisfies the desired value δ'_d
  - Update : $\alpha \leftarrow \alpha \times 0.99$
  - Termination condition : $\delta_d \geq \delta'_d$

4.3. Hierarchical sub-spaces

The space generation time is exponential to the number of effective vector spaces in a group, and the overall generation time is determined by the maximum space complexity. From the hierarchical space generation, we can reduce the maximum space complexity, and it can save the space generation time with the amount of complexity reduction.

From the space constraint such as target complexity, we can simplify the extraction of candidate rank configurations by hierarchically generating the combinatorial space.

- Total complexity : $C(R) = C(r_1, r_2, ..., r_L) = \sum_{l=1}^{L} C_l(r_l) = c_1 r_1 + c_2 r_2 + ... + c_L r_L$
  - Complexity of l-th layer for r_l : $C_l(r_l) = c_l r_l$
  - Coefficient of layer complexity : c_l
  - Vector space of rank (in min-offset space (Sec.5.2)) : $r_l = [0 : t_l : r_{l,max}]$
    - t_l : step size of rank in the vector space of r_l
    - $r_{l,max}$: maximum rank in the vector space of r_l with min-offset space
- Grouping rule : gather the layers having same layer complexity
- For example) if $c_2 = c_3 = c_4$:
  - Total complexity $C(R) = c_1 r_1 + c_2(r_2 + r_3 + r_4) + c_5 r_5 ... + c_L r_L = c_1 r_1 + c'_2 r'_2 + c_5 r_5 ... + c_L r_L$
  - c'_2 is equal to $c_2 = c_3 = c_4$
  - r'_2 is defined by :
    - step size of rank : $t_2 = \min(\{t_2, t_3, t_4\})$
    - maximum rank : $r'_{2,max} = \text{sum}(\{\max(r_2), \max(r_3), \max(r_4)\})$
    - total range : $r'_2 = [0 : t_2 : r'_{2,max}]$
  - if $r'_2 = k$ (*scalar*) : find the rank configurations of $\{r_2, r_3, r_4\}$ satisfying $(r_2 + r_3 + r_4) = k$
    - Sub-space $\mathcal{X}$ includes all possible combinations

4.3.1 VGG-16

- Total top groups : 8
- Grouped vector spaces :
  - G1 $= r_2$, G2 $= r_3$, ..., G5 $= r'_6$, G6 $= r_8$, G7 $= r'_9$, G8 $= r'_{11}$
- Sub-spaces & Space complexity:
  - Space complexity = $O(n^N)$, N : number of effective vector spaces in a group
  - $\mathcal{X}_1 = \{r_2, r_3, r_4, r_5, r'_6, r_8, r'_9, r'_{11}\} \rightarrow O(n^8)$
  - $\mathcal{X}_2 = \{r_6, r_7\} \rightarrow O(n^2)$
  - $\mathcal{X}_3 = \{r_9, r_{10}\} \rightarrow O(n^2)$
  - $\mathcal{X}_4 = \{r_{11}, r_{12}, r_{13}\} \rightarrow O(n^3)$
- Accuracy metric : A_c

Layer (l)	I_l	O_l	D_l	W_l	Complexity of a Layer (C_l)	Group
1	3	64	3	224	4566016	-
2	64	64	3	224	19267584	**G1**
3	64	128	3	112	7225344	**G2**
4	128	128	3	112	9633792	**G3**
5	128	256	3	56	3612672	**G4**
6	256	256	3	56	4816896	**G5**
7	256	256	3	56	4816896	
8	256	512	3	28	1806336	**G6**
9	512	512	3	28	2408448	**G7**
10	512	512	3	28	2408448	
11	512	512	3	14	602112	**G8**
12	512	512	3	14	602112	
13	512	512	3	14	602112	

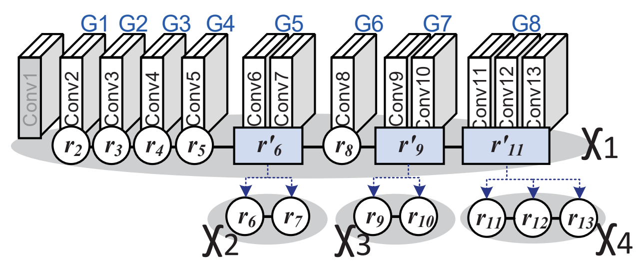

Figure 4. Sub-spaces of VGG-16. Compexity of a layer $C_l = I_l O_l D_l^2 W_l^2$, where I_l is the number of input channels, O_l is the number of output filters, D_l is the filter window size, and W_l is the size of output feature map. There are 4 sub-spaces, $\{\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \mathcal{X}_4\}$. The maximum complexity of space generation is redued from $O(n^{12})$ to $O(n^8)$.

4.3.2 ResNet-56

- Total top groups : 5
- Grouped vector spaces
  - Level-1 (Top) :
    - $\mathcal{X}_1 = \{\mathrm{G1} = r'_2, \quad \mathrm{G2} = r_{20}, \quad \mathrm{G3} = r'_{21}, \quad \mathrm{G4} = r_{38}, \quad \mathrm{G5} = r'_{39}\}$
  - Level-2 :
    - $\mathrm{G1}(\mathcal{X}_2) = \{\text{G1-1} = r'_{2,1}, \text{G1-2} = r'_{2,2}, ..., \text{G1-5} = r'_{2,5}\}$
    - $\mathrm{G3}(\mathcal{X}_3) = \{\text{G3-1} = r'_{21,1}, \text{G3-2} = r'_{21,2}, ..., \text{G3-5} = r'_{21,5}\}$
    - $\mathrm{G5}(\mathcal{X}_4) = \{\text{G5-1} = r'_{39,1}, \text{G5-2} = r'_{39,2}, ..., \text{G5-5} = r'_{39,5}\}$
  - Level-3 :
    - $\text{G1-1}(\mathcal{X}_6) = \{r_2, r_3, r_4, r_5\}, \text{G1-2}(\mathcal{X}_7) = \{r_6, r_7, r_8, r_9\}, ..., \text{G1-5}(\mathcal{X}_{10}) = \{r_{18}, r_{19}\}$
    - $\text{G3-1}(\mathcal{X}_{11}) = \{r_{21}, r_{22}, r_{23}, r_{24}\}, \text{G3-2}(\mathcal{X}_{12}) = \{r_{25}, r_{26}, r_{27}, r_{28}\}, ..., \text{G3-5} = r_{37}$
    - $\text{G5-1}(\mathcal{X}_{15}) = \{r_{39}, r_{40}, r_{41}, r_{42}\}, \text{G5-2}(\mathcal{X}_{16}) = \{r_{43}, r_{44}, r_{45}, r_{46}\}, ..., \text{G5-5} = r_{55}$
- Space complexity:
  - Space complexity = $O(n^N)$, $\quad N$: number of effective vector spaces in a group
  - Level-1 (Top) : $\mathcal{X}_1 \rightarrow O(n^5)$
  - Level-2 : $\mathcal{X}_2 \rightarrow O(n^5), \mathcal{X}_3 \rightarrow O(n^5), \mathcal{X}_4 \rightarrow O(n^5)$
  - Level-3 :
    - $\mathcal{X}_5 \rightarrow O(n^4), \mathcal{X}_6 \rightarrow O(n^4), \mathcal{X}_7 \rightarrow O(n^4), \mathcal{X}_8 \rightarrow O(n^4), \mathcal{X}_9 \rightarrow O(n^2)$
    - $\mathcal{X}_{10} \rightarrow O(n^4), \mathcal{X}_{11} \rightarrow O(n^4), \mathcal{X}_{12} \rightarrow O(n^4), \mathcal{X}_{13} \rightarrow O(n^4)$
    - $\mathcal{X}_{14} \rightarrow O(n^4), \mathcal{X}_{15} \rightarrow O(n^4), \mathcal{X}_{16} \rightarrow O(n^4), \mathcal{X}_{17} \rightarrow O(n^4)$
- Accuracy metric : A_p

Layer (l)	I_l	O_l	D_l	W_l	Complexity of a Layer (C_l)	Top Group	Sub Groups
1	3	16	3	32	44032	-	-
2	16	16	3	32	98304		**G1-1**(2,3,4,5) **G1-2**(6,7,8,9)
…	-	-	-	-	-	**G1**	**G1-3**(10,11,12,13) **G1-4**(14,15,16,17)
19	16	16	3	32	98304		**G1-5**(18,19)
20	16	32	3	16	36864	**G2**	-
21	32	32	3	16	49152		**G3-1**(21,22,23,24) **G3-2**(25,26,27,28)
…	-	-	-	-	-	**G3**	**G3-3**(29,30,31,32) **G3-4**(33,34,35,36)
37	32	32	3	16	49152		**G3-5**(37)
38	32	64	3	8	18432	**G4**	-
39	64	64	3	8	24576		**G5-1**(39,40,41,42) **G5-2**(43,44,45,46)
…	-	-	-	-	-	**G5**	**G5-3**(47,48,49,50) **G5-4**(51,52,53,54)
55	64	64	3	8	24576		**G5-5**(55)

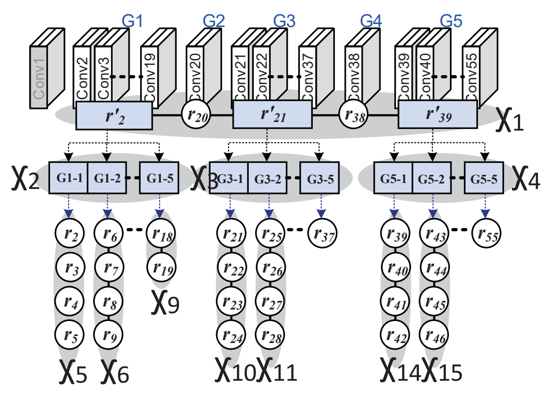

Figure 5. Sub-spaces of ResNet-56. Compexity of a layer $C_l = I_l O_l D_l^2 W_l^2$, where I_l is the number of input channels, O_l is the number of output filters, D_l is the filter window size, and W_l is the size of output feature map. There are 17 sub-spaces, $\{\mathcal{X}_1, \mathcal{X}_2, ..., \mathcal{X}_{17}\}$. The maximum complexity of space generation is redued from $O(n^{54})$ to $O(n^5)$.