Deep Pyramidal Residual Networks (PyramidNets)

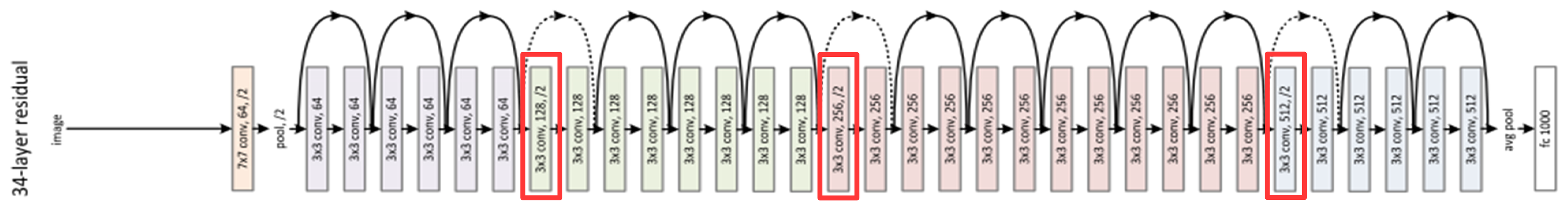
Dongyoon Han*, Jiwhan Kim*, and Junmo Kim (*authors equally contributed)

dyhan@kaist.ac.kr, jhkim89@kaist.ac.kr, and junmo.kim@kaist.ac.kr

School of Electrical Engineering, KAIST, South Korea

IEEE 2017 Conference on Computer Vision and Pattern Recognition

1.Motivation

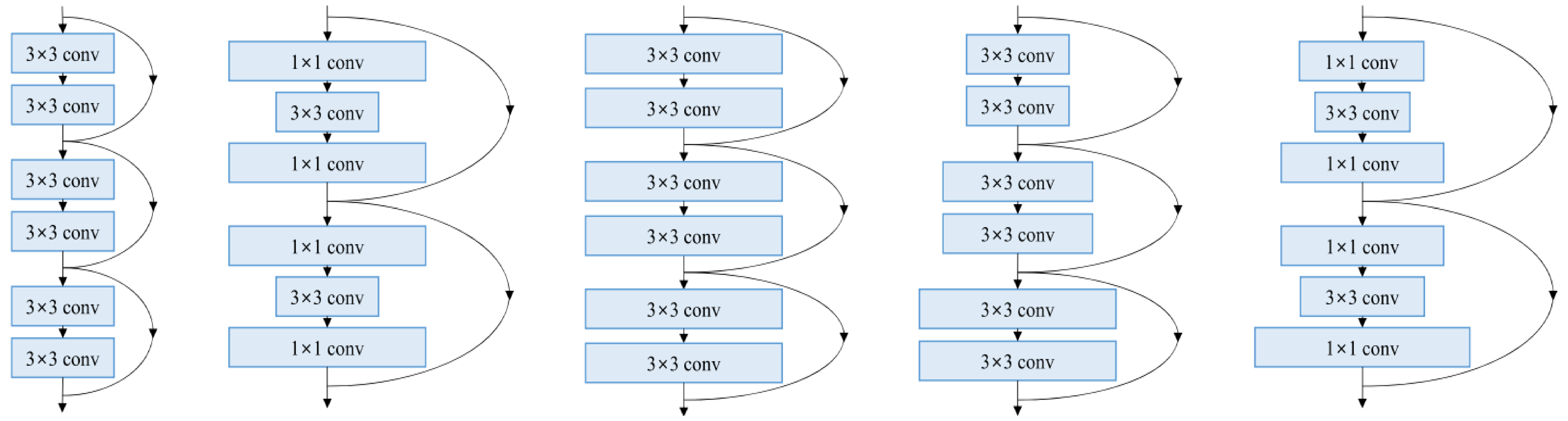


- When removing the downsampling residual units, test errors are fluctuated drastically. We believe that each downsampling unit that increases feature map dimension may play important roles on performance.
- Motivated by this, the main idea of PyramidNet is increasing the feature map dimension gradually instead of increasing it sharply at each residual unit with downsampling.
- This feature map configuration may distribute the burden concentrated of downsampling residual units, such that it is equally distributed across all units.

2. Architecture Details

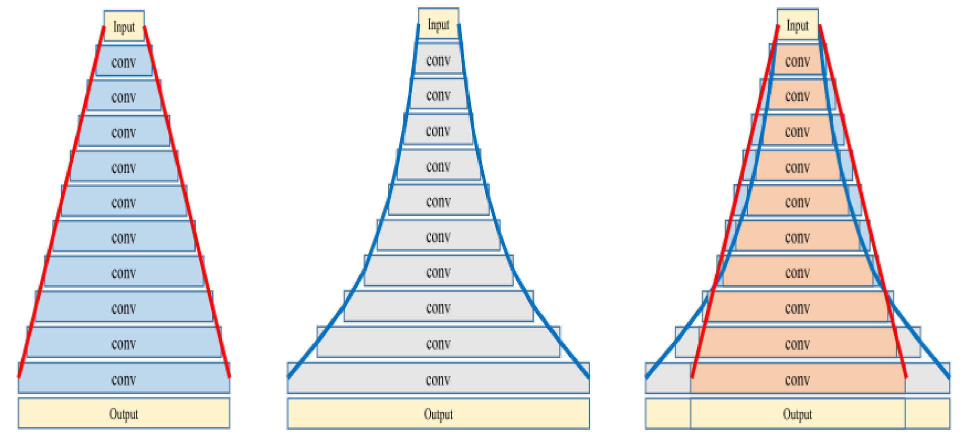
2.1. Schematic Illustration

- Pyramidal and pyramidal bottleneck residual unit are proposed:



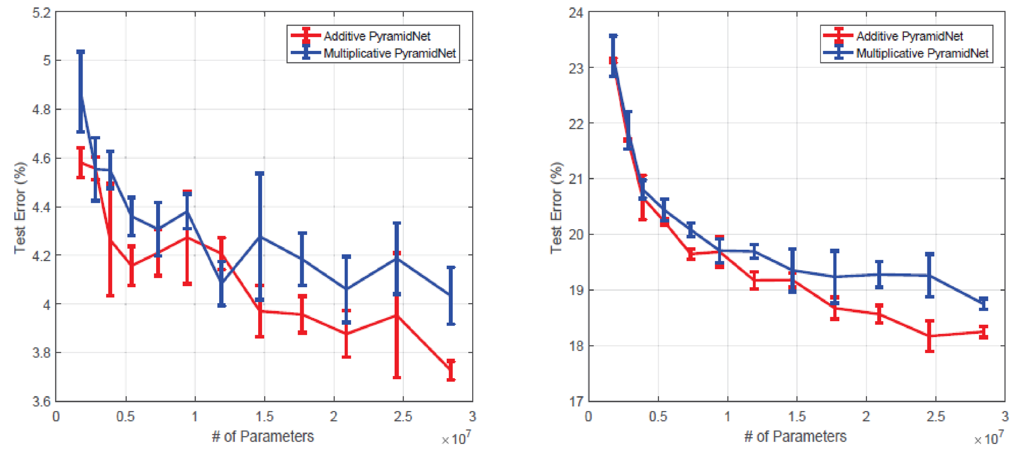
2.2. Additive vs Multiplicative PyramidNet

- **Contribution 1**: proposed ways of **feature map dimension** configuration:
  - **Additive PyramidNet**: feature map dimension increases **linearly**: $D_k = D_{k-1} + \alpha \ (\alpha \geq 0)$
  - **Multiplicative PyramidNet**: feature map dimension increases **geometrically**: $D_k = D_{k-1} \cdot \alpha \ (\alpha \geq 1)$



(a) (b) (c)

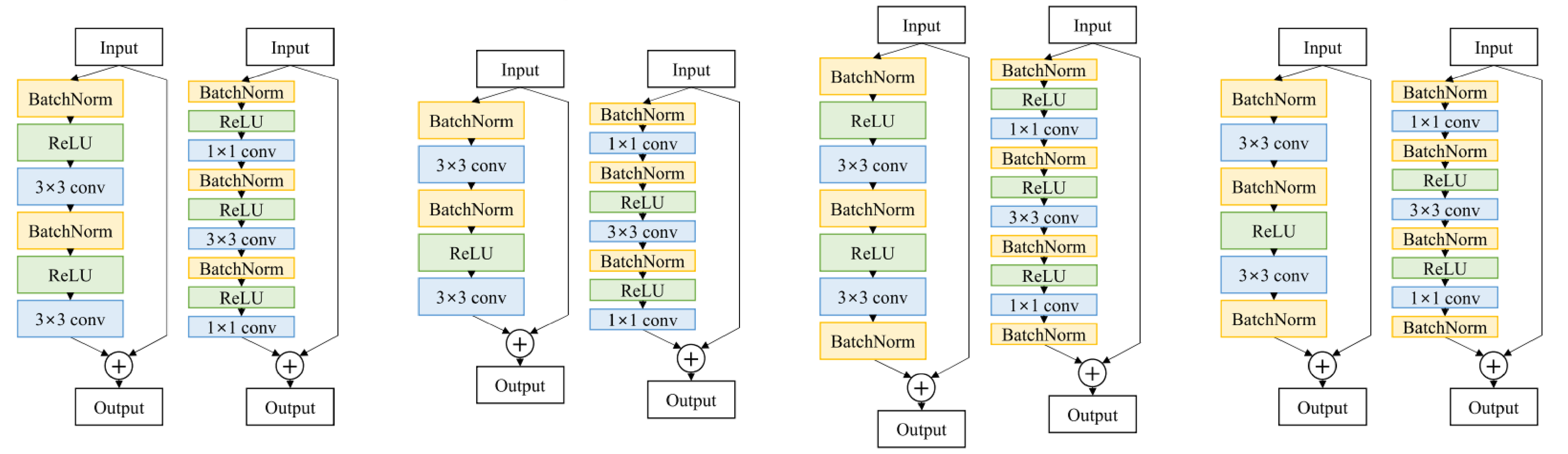
· Visual illustrations of (a) additive PyramidNet, (b) multiplicative PyramidNet, and (c) a comparison of (a) and (b).



· Test error curves with error bars of additive and multiplicative PyramidNet, according to the different number of parameters.

2.3. A New Building Block

- **Contribution 2**: a **new building block** is proposed:



(a) (b) (c) (d)

- (a) - the building block of **original** pre-activation ResNet
- (b) - the building block of pre-activation ResNet, in which the first **ReLU is removed**.
- (c) - the building block of pre-activation ResNet, in which **a new Batch Normalization (BN) layer is added** after the last convolution layer.
- (d) - **(b) + (c) (first ReLU is removed and additional BN is added)**

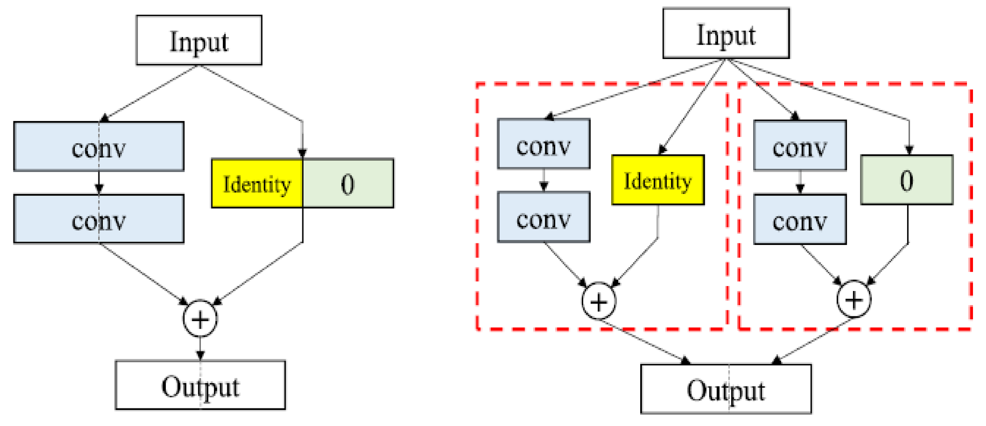
ResNet Architecture	CIFAR-10	CIFAR-100
(a) Pre-ResNet	5.82	25.06
(b) Removing the first ReLU	5.31	24.55
(c) BN after the final conv	5.74	24.54
(d) Removing the first ReLU, BN after the final conv	**5.29**	**23.74**
PyramidNet Architecture	**CIFAR-10**	**CIFAR-100**
(a) Pre-ResNet	5.15	24.40
(b) Removing the first ReLU	4.81	23.43
(c) BN after the final conv	4.96	23.89
(d) Removing the first ReLU, BN after the final conv	**4.62**	**23.31**
PyramidNet (bottleneck) Architecture	**CIFAR-10**	**CIFAR-100**
(a) Pre-ResNet	4.61	21.10
(b) Removing the first ReLU	4.45	20.40
(c) BN after the final conv	4.56	20.44
(d) Removing the first ReLU, BN after the final conv	**4.26**	**20.32**

- The ablation experiment shows that the building block of (d) could be the most promising unit:
  - Basic block: **BN-conv3-BN-ReLU-conv3-BN**
  - Bottleneck block: **BN-conv1-BN-ReLU-conv3-BN-ReLU-conv3-BN-ReLU-conv3-BN**

2.4. Zero-padded Identity-Mapping Shortcut

- PyramidNets' feature map dimension differs among individual residual units: Identity shortcut cannot be used.
- Zero-padded identity-mapping shortcut is adopted:
  - Concatenation of Identity and zeros.
  - A major advantage: no extra parameters (easier optimization, less overfitting).

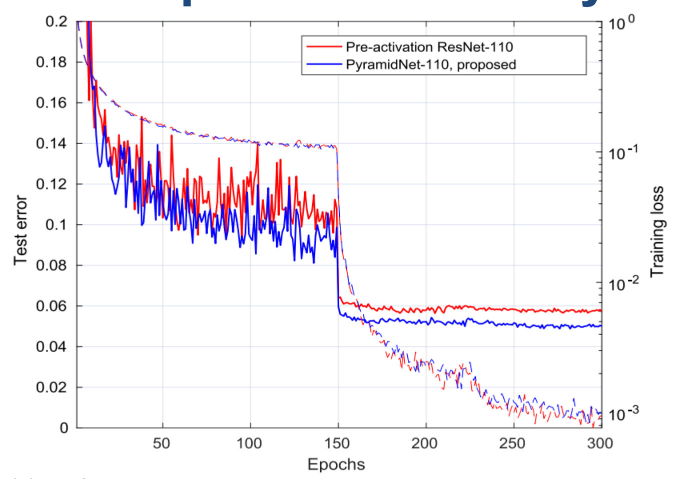
Shortcut Types	CIFAR-10	CIFAR-100
(a) Identity mapping with projection shortcut	5.03	23.48
(b) Projection with zero-padded shortcut	6.84	31.29
(c) All projection shortcut	6.98	31.62
(d) Identity mapping with zero-padded shortcut	**4.70**	**22.77**



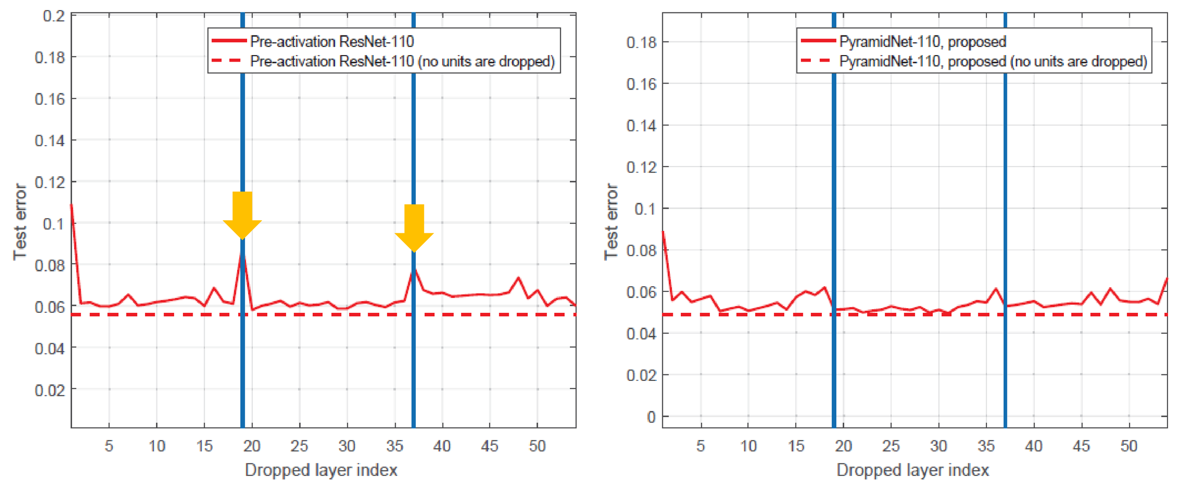
(a) (b)

3. Experiment

3.1. Experimental Study



(a) Performance comparison between pre-activation ResNet and PyramidNet on CIFAR-10 daaset. Dashed and solid lines denote the training loss and test error, respectively.



(b) When dropping a single layer, no fluctuation phenomenon is observed with PyramidNet. Bold vertical lines denote the location of residual units through downsampling.

3.2. Experimental Results

Network	# of Params	Output Feat. Dim.	Depth	CIFAR-10	CIFAR-100
NIN	-	-	-	8.81	35.68
Highway	-	-	-	7.72	32.39
ResNet-110	1.7M	64	110	6.43	25.16
ResNet-1001	10.2M	256	1,001	-	27.82
ResNet-1202	19.4M	64	1,202	7.93	-
Pre-activation ResNet-164	1.7M	256	164	5.46	24.33
Pre-activation ResNet-1001	10.2M	256	1,001	4.62	22.71
Stochastic Depth	1.7M	64	110	5.23	24.58
Stochastic Depth	10.2M	64	1,202	4.91	-
Wide-ResNet-40 (width ×4)	8.7M	256	40	4.97	22.89
Wide-ResNet-28 (width ×10)	36.5M	640	28	4.17	20.50
DenseNet ($k = 24$)	27.2M	2,320	100	3.74	19.25
DenseNet-BC ($k = 40$)	25.6M	2,190	190	**3.46**	**17.18**
PyramidNet ($\alpha = 48$)	1.7M	64	110	4.58±0.06	23.12±0.04
PyramidNet ($\alpha = 84$)	3.8M	100	110	4.26±0.23	20.66±0.40
PyramidNet ($\alpha = 270$)	28.3M	286	110	**3.73±0.04**	**18.25±0.10**
PyramidNet (bottleneck, $\alpha = 270$)	27.0M	1,144	164	**3.48±0.20**	**17.01±0.39**

Network	# of Params	Output Feat. Dim.	Augmentation	Train crop	Test Crop	Top-1	Top-5
ResNet-152	60.0M	2,048	scale	224 × 224	224 × 224	23.0	6.7
Pre-ResNet-152	60.0M	2,048	scale + aspect ratio	224 × 224	224 × 224	22.2	6.2
Pre-ResNet-200	64.5M	2,048	scale + aspect ratio	224 × 224	224 × 224	21.7	5.8
Wide-ResNet-50-bottleneck	68.9M	2,048	scale + aspect ratio	224 × 224	224 × 224	21.9	6.0
PyramidNet-200 ($\alpha = 300$)	62.1M	1,456	scale + aspect ratio	224 × 224	224 × 224	**20.5**	**5.3**
PyramidNet-200 ($\alpha = 300$, dropout)	62.1M	1,456	scale + aspect ratio	224 × 224	224 × 224	**20.5**	**5.4**
PyramidNet-200 ($\alpha = 450$, dropout)	116.4M	2,056	scale + aspect ratio	224 × 224	224 × 224	**20.1**	**5.4**
ResNet-200	64.5M	2,048	scale	224 × 224	320 × 320	21.8	6.0
Pre-ResNet-200	64.5M	2,048	scale + aspect ratio	224 × 224	320 × 320	20.1	4.8
Inception-v3	-	2,048	scale + aspect ratio	299 × 299	299 × 299	21.2	5.6
Inception-ResNet-v1	-	1,792	scale + aspect ratio	299 × 299	299 × 299	21.3	5.5
Inception-v4	-	1,536	scale + aspect ratio	299 × 299	299 × 299	20.0	5.0
Inception-ResNet-v2	-	1,792	scale + aspect ratio	299 × 299	299 × 299	19.9	4.9
PyramidNet-200 ($\alpha = 300$)	62.1M	1,456	scale + aspect ratio	224 × 224	320 × 320	**19.6**	**4.8**
PyramidNet-200 ($\alpha = 300$, dropout)	62.1M	1,456	scale + aspect ratio	224 × 224	320 × 320	**19.5**	**4.8**
PyramidNet-200 ($\alpha = 450$, dropout)	116.4M	2,056	scale + aspect ratio	224 × 224	320 × 320	**19.2**	**4.7**

Code is available at https://github.com/jhkim89/PyramidNet (Torch)