Paper Review_
SKNet : Selective Kernel Network

Selective Kernel Networks

Xiang Li[*1,2], Wenhai Wang[†3,2], Xiaolin Hu[†4] and Jian Yang[§1]

[1]PCALab, Nanjing University of Science and Technology [2]Momenta [3]Nanjing University
[4]Tsinghua University

CVPR 2019

Visual Intelligence Lab
Dept. of Applied Artificial Intelligence
Minu Ham

Abstract

신경과학 관점에서 시각 피질 뉴런의 수용 영역(Receptive Field, RF) 크기는 자극에 따라 조절되지만 표준 CNN에서는 각 계층의 인공 뉴런의 수용 영역 크기가 동일하도록 설계됨

본 논문은 CNN에서 각 뉴런이 입력 정보의 다양한 스케일에 따라 수용 영역 크기를 적응적으로 조정할 수 있도록 하는 동적 선택 메커니즘 Selective Kernel(SK) 유닛을 제안하고 여러 SK 유닛을 쌓아 Selective Kernel Networks(SKNet)이라는 심층 네트워크를 설계함

이는 뉴런이 서로 다른 커널 크기를 가진 여러 분기(branch)를 결합하고, 이러한 분기에서 얻은 정보를 바탕으로 Softmax attention을 통해 가중치를 부여하여 서로 다른 효과적인 수용 영역 크기를 가짐

ImageNet 및 CIFAR 벤치마크에서 SOTA 모델보다 더 낮은 파라미터로 우수한 성능을 보이고, 실험으로 입력에 따라 뉴런이 수용 영역 크기를 적응적으로 조정할 수 있음을 검증함

Method	#P	Top-1 error (%)
ResNet-50 [9]	25.56M	23.9
ResNet-101 [9]	44.55M	22.6
ResNet-152 [9]	60.19M	21.7
DenseNet-169 (k=32) [13]	14.15M	23.8
DenseNet-201 (k=32) [13]	20.01M	22.6
DenseNet-264 (k=32) [13]	33.34M	22.2
DenseNet-232 (k=48) [13]	55.80M	21.3
ResNeXt-50 (32×4d) [47]	25.00M	22.2
ResNeXt-101 (32×4d) [47]	44.30M	21.2
DPN-68 (32×4d) [5]	12.61M	23.7
DPN-92 (32×3d) [5]	37.67M	20.7
DPN-98 (32×4d) [5]	61.57M	20.2
SENet-50 [12]	27.7M	21.12
SENet-101 [12]	49.2M	20.58
SKNet-26	16.8M	22.74
SKNet-50	27.5M	20.79
SKNet-101	48.9M	20.19

Table S9. The top-1 error rates (%) on the ImageNet validation set with single 224×224 crop testing.

Split : 다양한 커널 크기를 사용하는 다중 경로 생성

Fuse : 다중 경로에서 나온 정보를 결합 및 통합하여 선택

Select : 가중치에 따라 크기가 다중 분기의 피쳐를 통합

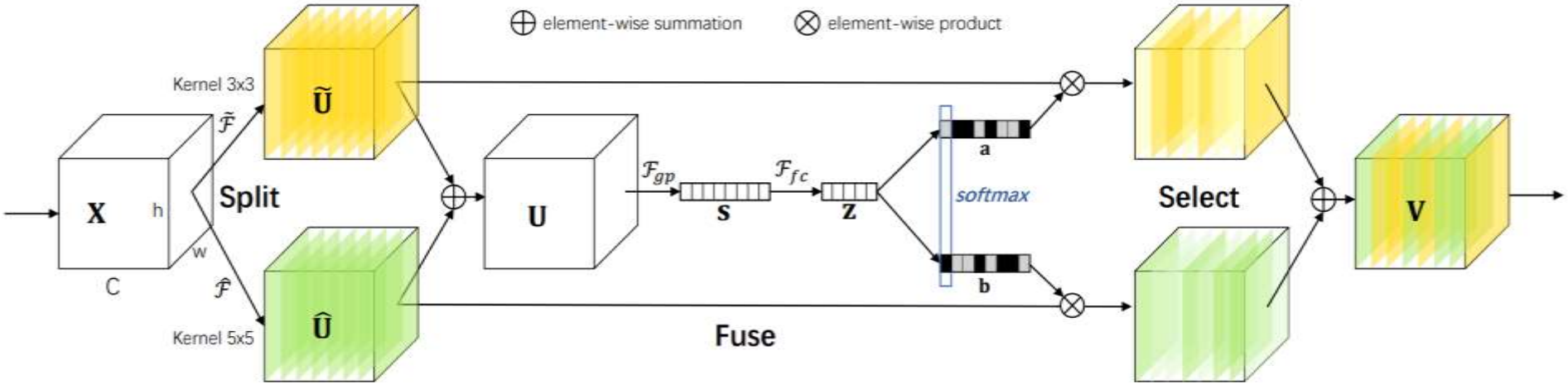


Figure 1. Selective Kernel Convolution.

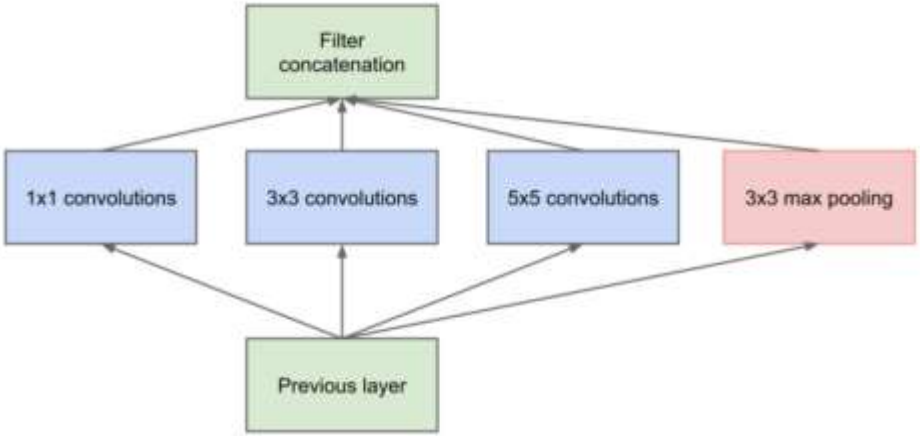
본 논문에서는 뉴런의 적응적 수용 영역(Receptive Field, RF) 동적 크기 조절을 구현하기 위해 다중 커널의 정보를 비선형적으로 통합하는 접근 방식을 제시

이를 위해 Selective Kernel(SK) 합성곱을 도입했으며, 이는 Split, Fuse, Select 라는 세 가지 operator로 구성.

SKNet의 뉴런의 RF 크기 조정 능력을 이미지에서 타겟 객체를 점점 확대하는 방식으로 검증, 타겟 객체의 크기와 큰 커널의 사용량이 비례하는 것을 확인

Multi-branch convolutional networks

GoogleNet_Inception module

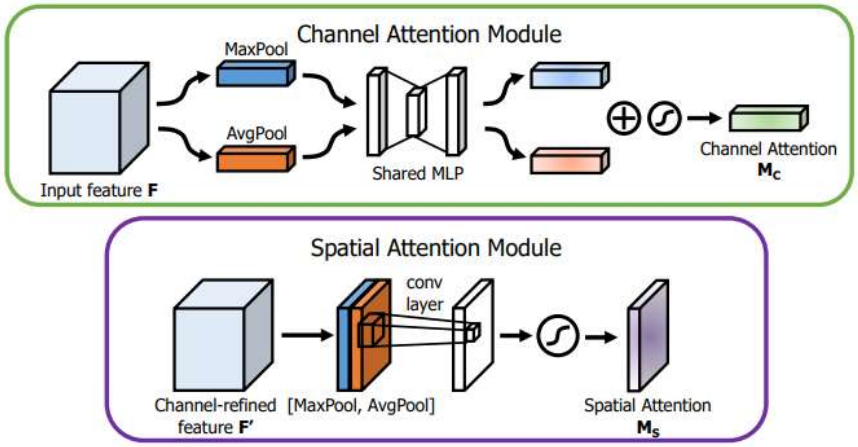


1) 무거운 맞춤형 설계가 아닌 단순한 구조

2) 여러 분기에 대한 적응형 선택 메커니즘을 활용하여 뉴런의 적응형 RF 크기를 실현

Attention Mechanisms

CBAM_Channel/Spatial Attention module

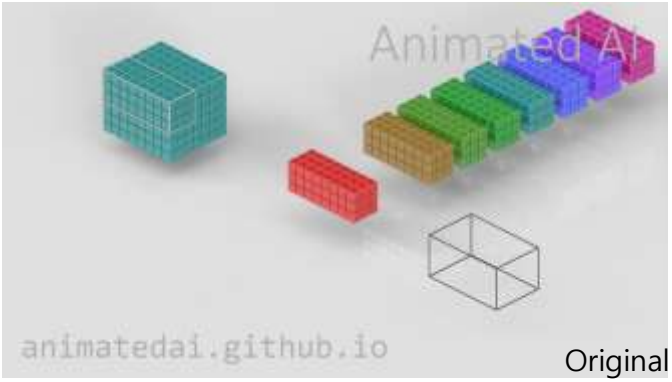


1) 뉴런의 적응적 수용 영역(Receptive Field, RF) 크기에 초점을 맞춤

2) 이를 위해 Attention 메커니즘을 도입하여 뉴런이 입력 정보에 따라 적응적으로 RF 크기를 조정할 수 있도록 설계

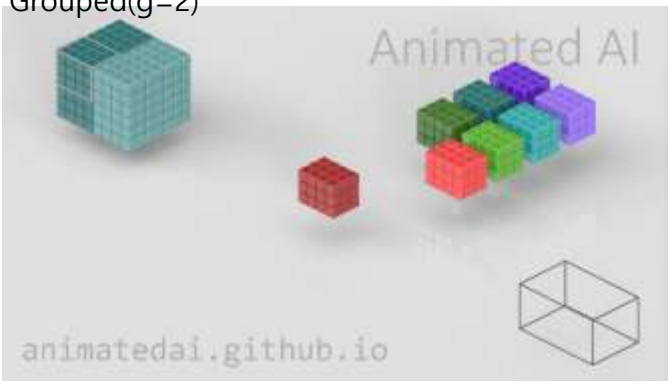
Related work

Grouped/Depthwise/Dilated Convolutions

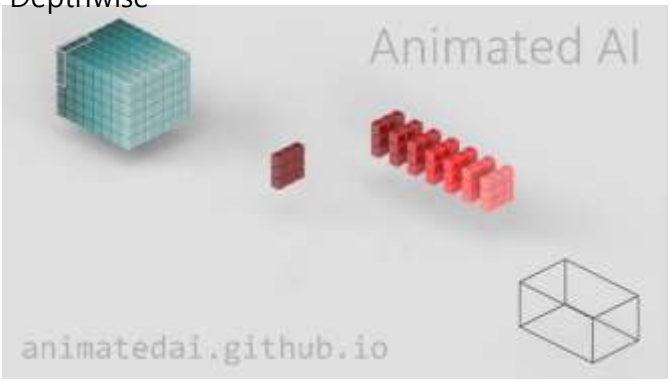


Original

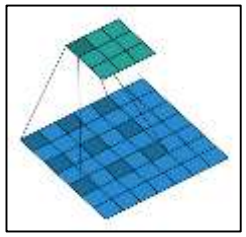
Grouped(g=2)



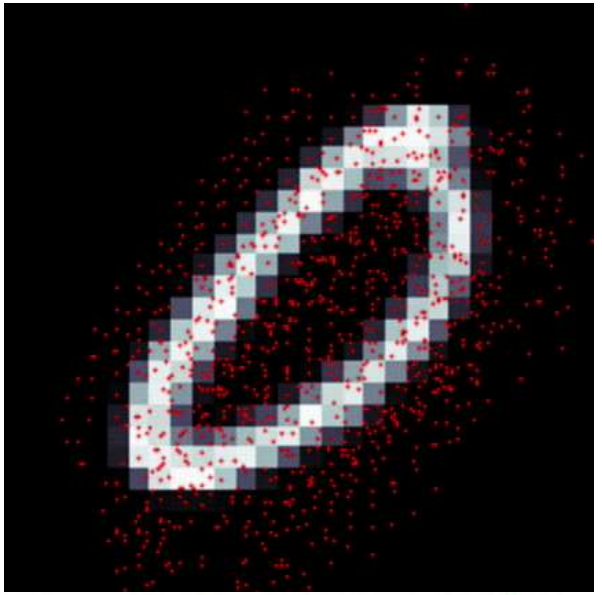
Depthwise



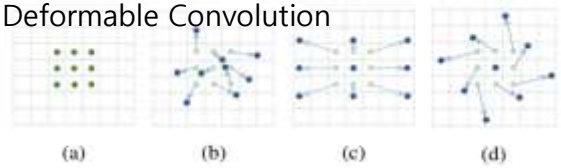
Dilated(d=2)

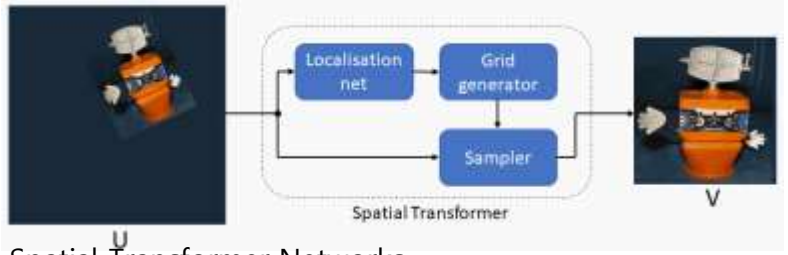


Dynamic Convolution



Deformable Convolution





Spatial Transformer Networks

Selective Kernel Convolution

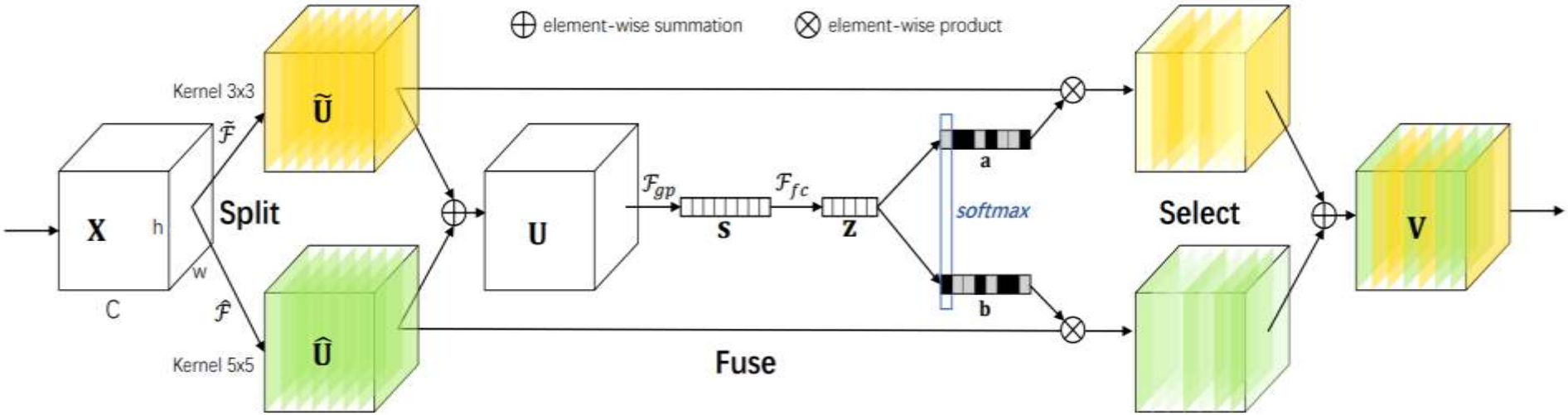


Figure 1. Selective Kernel Convolution.

1. Split

$$X \in R^{H_0 \times W_0 \times C_0}$$
$$F_e : X \to Ue \in R^{H \times W \times C} (\text{커널크기} : 3 * 3)$$
$$F_b : X \to U_b \in \mathbb{R}^{H \times W \times C} (\text{커널크기} := 3 * 3, \textit{dilation} = 2)$$

Operator :
- Grouped / Depthwise Convolution
- Batch Normalization
- ReLU

2. Fuse

$$U = U_e + U_b$$
$$s_c = F_{gp}(U_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} U_c(i,j)$$
$$z = F_{fc}(s) = \delta(B(Ws))$$

Operator :
- Elemnet-wise Summation
- Global Average Pooling
- Dimensionality Reduction

3. Select

$$a_c = \frac{e^{A_c z}}{e^{A_c z} + e^{B_c z}}, \quad \frac{e^{B_c z}}{e^{A_c z} + e^{B_c z}}$$
$$V_c = a_c \cdot U_e^c + b_c \cdot U_b^c,$$
$$V = [V_1, V_2, \ldots, V_C]$$
$$V \in R^{H \times W \times C} : \textit{Final Output}$$

Operator :
- Softmax Attention

Network Architecture

M=2: 두 개의 경로(3×3, 5×5 커널)
G=32: 각 경로의 그룹 수
r=16: fc layer 축소 비율

Output	ResNeXt-50 (32×4d)	SENet-50	SKNet-50
112 × 112	7 × 7, 64, stride 2		
56 × 56	3 × 3 max pool, stride 2		
56 × 56	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128, G=32 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128, G=32 \\ 1\times1, 256 \\ fc, [16, 256] \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 128 \\ SK[M=2, G=32, r=16], 128 \\ 1\times1, 256 \end{bmatrix} \times 3$
28 × 28	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, G=32 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, G=32 \\ 1\times1, 512 \\ fc, [32, 512] \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 256 \\ SK[M=2, G=32, r=16], 256 \\ 1\times1, 512 \end{bmatrix} \times 4$
14 × 14	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, G=32 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, G=32 \\ 1\times1, 1024 \\ fc, [64, 1024] \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 512 \\ SK[M=2, G=32, r=16], 512 \\ 1\times1, 1024 \end{bmatrix} \times 6$
7 × 7	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, G=32 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, G=32 \\ 1\times1, 2048 \\ fc, [128, 2048] \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 1024 \\ SK[M=2, G=32, r=16], 1024 \\ 1\times1, 2048 \end{bmatrix} \times 3$
1 × 1	7 × 7 global average pool, 1000-d fc, softmax		
#P	25.0M	27.7M	27.5M
GFLOPs	4.24	4.25	4.47

SKNet은 ResNeXt를 기반으로 설계
ResNeXt는 Grouped Convolution 사용

Architecture :
- 1×1 합성곱
- SK Convolution
- 1×1 합성곱

SKNet-50은 ResNeXt-50에 비해 파라미터 수가 10%, GFLOPs가 5% 증가

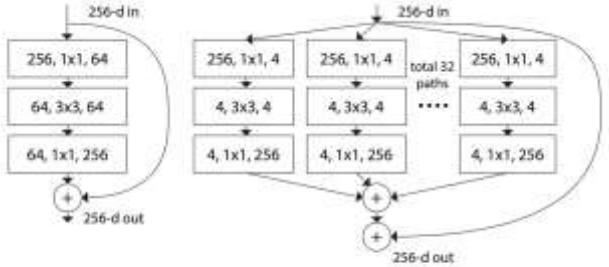


Figure 1. **Left**: A block of ResNet [14]. **Right**: A block of ResNeXt with cardinality = 32, with roughly the same complexity. A layer is shown as (# in channels, filter size, # out channels).

	top-1 err (%)		#P	GFLOPs
	224×	320×		
ResNeXt-50	22.23	21.05	25.0M	4.24
AttentionNeXt-56 [44]	21.76	–	31.9M	6.32
InceptionV3 [43]	–	21.20	27.1M	5.73
ResNeXt-50 + BAM [32]	21.70	20.15	25.4M	4.31
ResNeXt-50 + CBAM [45]	21.40	20.38	27.7M	4.25
SENet-50 [12]	21.12	19.71	27.7M	4.25
SKNet-50 (ours)	**20.79**	**19.32**	27.5M	4.47
ResNeXt-101	21.11	19.86	44.3M	7.99
Attention-92 [44]	–	19.50	51.3M	10.43
DPN-92 [5]	20.70	19.30	37.7M	6.50
DPN-98 [5]	20.20	18.90	61.6M	11.70
InceptionV4 [41]	–	20.00	42.0M	12.31
Inception-ResNetV2 [41]	–	19.90	55.0M	13.22
ResNeXt-101 + BAM [32]	20.67	19.15	44.6M	8.05
ResNeXt-101 + CBAM [45]	20.60	19.42	49.2M	8.00
SENet-101 [12]	20.58	18.61	49.2M	8.00
SKNet-101 (ours)	**20.19**	**18.40**	48.9M	8.46

Table 2. Comparisons to the state-of-the-arts under roughly identical complexity. 224× denotes the single 224×224 crop for evaluation, and likewise 320×. Note that SENets/SKNets are all based on the corresponding ResNeXt backbones.

	top-1 err. (%)	#P	GFLOPs
ResNeXt-50 (32×4d)	22.23	25.0M	4.24
SKNet-50 (ours)	**20.79** (1.44)	27.5M	4.47
ResNeXt-50, wider	22.13 (0.10)	28.1M	4.74
ResNeXt-56, deeper	22.04 (0.19)	27.3M	4.67
ResNeXt-50 (36×4d)	22.00 (0.23)	27.6M	4.70

Table 3. Comparisons on ImageNet validation set when the computational cost of model with more depth/width/cardinality is increased to match that of SKNet. The numbers in brackets denote the gains of performance.

ImageNet Classification

SKNet-50은 ResNeXt-101보다 0.32% 더 나은 성능

ResNeXt-101은 파라미터가 60%, GFLOPs가 80% 더 많음

InceptionNet과 파라미터가 비슷하지만 1.5% 이상의 성능 향상을 달성

또한, 약간 더 적은 파라미터를 사용하면서도 SENet 대비 0.3~0.4%의 성능이 향상됨

ResNeXt(32×4d)와 비교했을 때, 파라미터와 계산량이 약간 증가

공정한 비교를 위해 ResNeXt의 복잡도를 SKNet과 맞춤

ResNeXt의 깊이(depth), 너비(width), **카디널리티(cardinality)**를 증가시켜 SKNet의 복잡도에 맞춘 경우, 성능 향상은 제한적

- ResNeXt-50에서 "wider" 모델로 변경: 0.10% 향상
- ResNeXt-56으로 더 깊게 변경: 0.19% 향상
- ResNeXt-50 (36×4d): 0.23% 향상

SKNet-50은 1.44% 성능 향상

Settings			top-1 err. (%)	#P	GFLOPs	Resulted Kernel
Kernel	D	G				
3×3	3	32	20.97	27.5M	4.47	7×7
3×3	2	32	**20.79**	27.5M	4.47	5×5
3×3	1	32	20.91	27.5M	4.47	3×3
5×5	1	64	**20.80**	28.1M	4.56	5×5
7×7	1	128	21.18	28.1M	4.55	7×7

Table 6. Results of SKNet-50 with different settings in the second branch, while the setting of the first kernel is fixed. "Resulted kernel" in the last column means the approximate kernel size with dilated convolution.

K3	K5	K7	SK	top-1 err. (%)	#P	GFLOPs
✓				22.23	25.0M	4.24
	✓			25.14	25.0M	4.24
		✓		25.51	25.0M	4.24
✓	✓			21.76	26.5M	4.46
✓	✓		✓	20.79	27.5M	4.47
✓		✓		21.82	26.5M	4.46
✓		✓	✓	20.97	27.5M	4.47
	✓	✓		23.64	26.5M	4.46
	✓	✓	✓	23.09	27.5M	4.47
✓	✓	✓		21.47	28.0M	4.69
✓	✓	✓	✓	20.76	29.3M	4.70

Table 7. Results of SKNet-50 with different combinations of multiple kernels. Single 224×224 crop is utilized for evaluation.

복잡도가 유사한 조건에서 두 번째 커널 분기의 RF를 확장하는 두 가지 방법:

- Dilation D 증가: 그룹 수 G를 고정한 상태에서 D를 증가
- 필터 크기와 그룹 수 G를 동시에 증가: 필터 크기와 G를 동시에 확대

두 번째 분기에 대한 최적 설정:

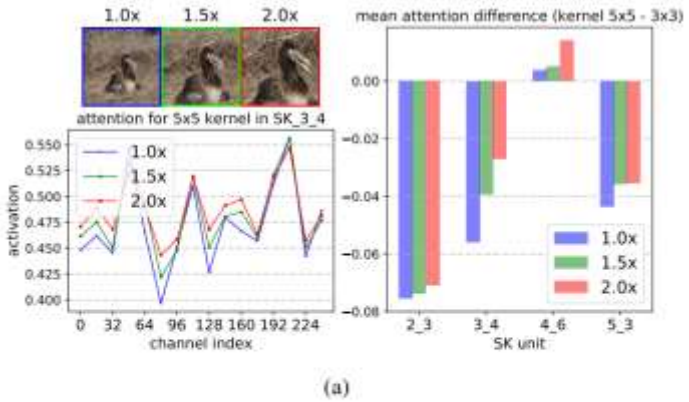
- 5×5 커널 크기(마지막 열)
- 다중 스케일 정보의 집계(aggregation of multi-scale information)가 유익하다는 것을 입증

두 가지 최적 구성:

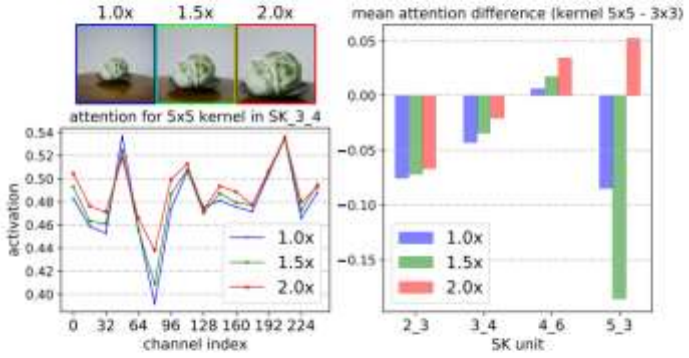
- 5×5 커널 크기와 D=1
- 3×3 커널 크기와 D=2-모델 복잡도가 약간 더 낮음

다중 커널 조합 분석 결과:

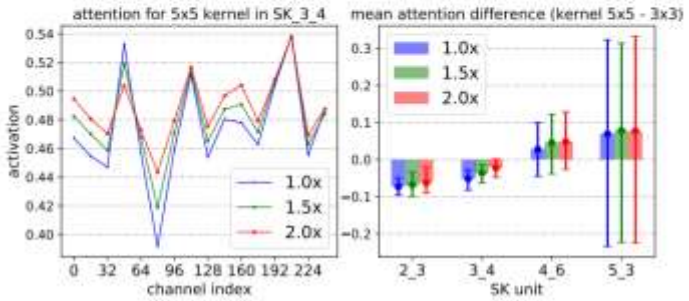
- 단일 커널: 단일 커널(K3, K5, K7)을 사용하는 경우 성능이 떨어짐
- 다중 커널 조합: K3+K5+K7: 가장 성능이 높음
  - 그러나 K3+K5 조합과 비교하여 큰 이점이 없음



(a)



(b)



(c)

목표 객체의 크기가 커짐에 따라 SKNet의 커널 선택 Attention이 어떻게 변화하는지 :

- 동일한 이미지에서 중심 객체(새)가 1.0×, 1.5×, 2.0×로 확대된 샘플.
- 채널별 Attention 변화: 특정 SK 유닛(SK_3_4)에서 5×5 커널에 대한 채널별 Attention 값을 나타냄.빨간색(2.0×) 그래프에서 5×5 커널의 Attention 값이 대체로 증가. 이는 목표 객체가 커질수록 SKNet이 더 큰 수용 영역(Receptive Field)을 필요로 한다는 것을 의미.
- 평균 Attention 차이: 5×5와 3×3 커널 간의 Attention 차이를 SK 유닛별로 비교.
- 1.0×에서 2.0×로 객체가 커질수록 큰 커널(5×5)에 대한 선호도가 증가.

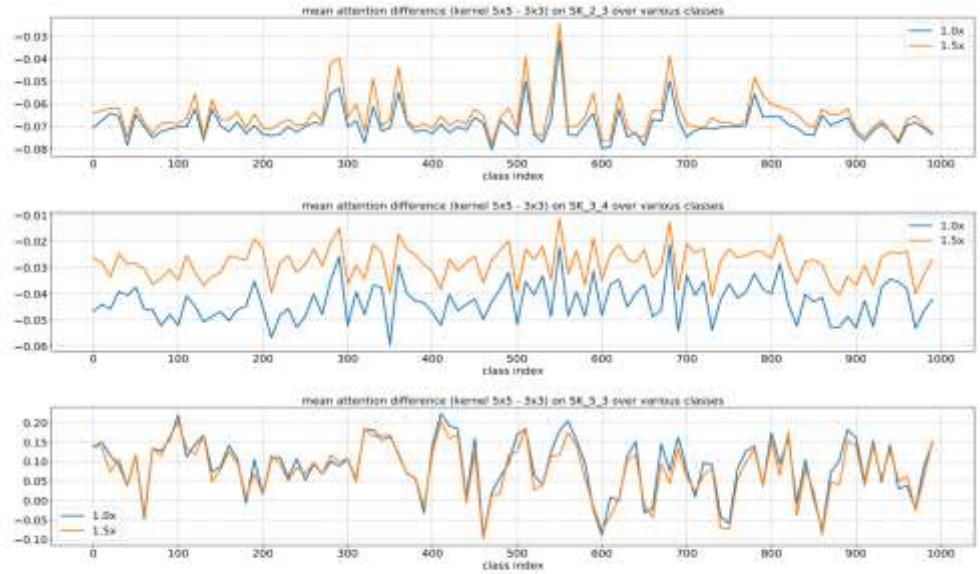


Figure 4. Average mean attention difference (mean attention value of kernel 5×5 minus that of kernel 3×3) on SK units of SKNet-50, for each of 1,000 categories using all validation samples on ImageNet. On low or middle level SK units (e.g., SK_2_3, SK_3_4), 5×5 kernels are clearly imposed with more emphasis if the target object becomes larger (1.0x → 1.5x).

(1) SK_2_3 (초기 단계) 특징: 객체 크기가 증가하면 5×5 커널에 대한 Attention 증가.

초기 SKNet은 객체 크기 변화에 따라 더 넓은 수용 영역(큰 커널)을 선호.

- SK_2_3과 같은 초기 레이어는 입력 이미지의 세부적이고 국소적인 정보를 처리하며, 객체 크기에 민감하게 반응.

(2) SK_3_4 (중간 단계) 특징: 5×5 커널의 Attention 차이가 SK_2_3에 비해 작지만, 여전히 객체 크기가 커질수록 큰 커널에 대한 선호도가 증가.

의미: 중간 레이어에서도 다중 스케일 정보를 처리하며, 객체 크기 변화에 따른 적응적 선택을 유지.

(3) SK_5_3 (고수준 단계) 특징: 5×5와 3×3 커널 간의 Attention 차이가 감소하거나 거의 없어짐. 객체 크기 변화(1.0× → 1.5×)에 따른 큰 커널 선호도가 약해짐.

의미: 고수준 레이어는 더 추상적인 전역 정보를 처리, 특정 스케일에 덜 의존. 객체 크기 변화에 따른 반응이 약화됨.

Thanks