Weiyang Liu, Zhen Liu, Zhiding Yu, Bo Dai, Rongmei Lin, Yisen Wang, James M.Rehg, Le Song

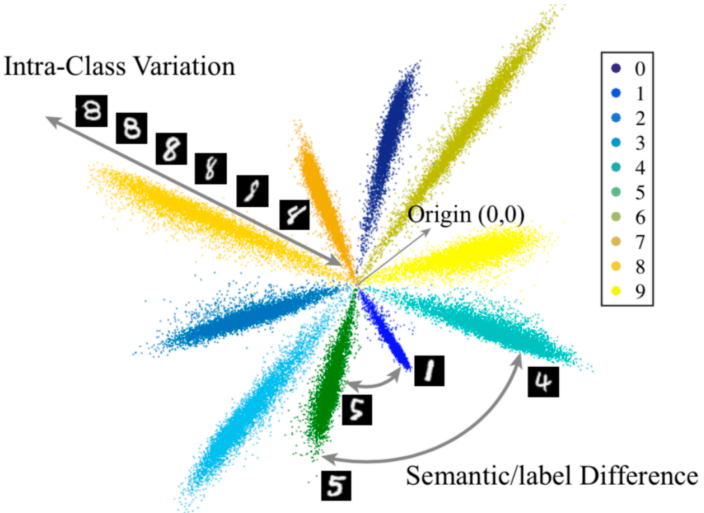
Georgia Institute of Technology, NVIDIA, Emory University, Tsinghua University, Ant Financial

Abstract

- 이 논문에서는 기존 w 와 x 의 내적을 'decouple'하여 intra-class variation 과 semantic difference 에 대하여 조금 더 학습을 잘 하기 위한 방법을 제안하였다.
- decouple 을 수행하기 위한 다양한 operators 를 제안하였다.
- 이렇게 decouple 하게 되면 신경망이 조금 더 빠르게 수렴하고, 강인해지도록 만드는 효과가 있다. (뒤의 실험 결과를 보면 알 수 있겠지만, 실제 Adversarial attack 에 강인해지는 효과가 있음을 보여주었다.)

Introduction

- 기존 convolution 신경망의 $< w, x >= w^T x$ 는 intra-class variation 과 semantic difference(inter-class variation) 를 하나의 couple 형태로 측정하게 된다.
- 위의 결과처럼 couple 형태로 측정하게 되면 다음과 같은 질문에 대하여 알기 어려워진다. 2개의 샘플의 내적 결과값이 크다면, '2개의 샘플의 semantic/label difference 를 가진것인지?' 'large intra-class variation 을 가진것인지?' (즉, 같은 클래스 내부에서 분리가 된것인지 , 다른 클래스의 성격을 갖고 있는 것인지에 대한 판단이 어렵다)
- 따라서, 기존 $w^T x$ 의 식을 $||w||\ ||x|| \cos(\theta_{(w,x)})$ 로 분리하였다. angle은 semantic/label difference 를 나타내게 되고, feature의 norm은 intra-class variation을 나타내게 된다.



- $||w||\ ||x|| \cos(\theta_{(w,x)})$ 의 수식에서 $h(||w||\ ||x||)$ 을 magnitude function, $g(\theta_{(w,x)})$ 을 anuglar function 으로 정의한다. 이 때, $h(||w||\ ||x||)$ = $||w||\ ||x||$, $g(\theta_{(w,x)})$ = $\cos(\theta_{(w,x)})$ 을 나타낸다.
    - manitude function 은 intra-class variation 을 나타내고, angular function 은 semantic difference 를 나타낸다.

- Decoupling 관점에서 바라보면, 기존 CNN은 norm의 곱형태로 intra-class variation 을 선형적으로 모델링할 수 있고, semantic difference를 cosine angle으로 설명할 수 있다는 강력한 가정을 만들어낸다. 그렇지만, 모든 측면에서 이 모델링이 최적으로 동작하지는 못한다.
- DCNet(Decoupled Network) 은 다음과 같은 4가지 측면에서 장점을 가진다.
    - intra class variation 과 semantic difference를 더 잘 모델링 할 수 있을 뿐만 아니라 이러한 특징들을 바꾸지 않고 직접적으로 학습할 수 있도록 해준다.
    - bounded magnitude function 을 사용함으로써, 수렴속도를 빠르게 할 수 있다. (bounded magnitude function 은 뒤에서 설명할 예정이지만, softmax 처럼 단순히 값의 범위를 정해놓는다고 생각하면 편하다.)
    - Adversarial attacks 에 강하다. bounded 된 magnitude 에서 각 클래스의 feature를 사용하기 때문에 강인하다.
    - decoupled operators 는 매우 유연하게 사용 가능하며, architecture-agnostic 하다. (VGG, GoogleNet등 어디서든 사용가능하다.)
- 이 논문에서는 2가지의 decoupled operators를 제안한다.
    - Bounded operators
        - 조금 더 빠르게 수렴하며, adversarial attack 에 강인하다.
    - Unbounded operators
        - 더 많은 representational power 를 가진다.
    - 추가적으로, operator radius 라는 개념을 제안한다.
        - operator radius 는 $\|x\|$ 의 입력에 따라 변화하는 magnitude function $h()$ 의 미분의 critical change를 나타낸다.

Decoupled Networks

Reparametrizing Convolution via Decoupling

- Conventional : $f(w,x) =< w,x >= w^T x$
- Decoupled form : $f(w,x) = \|w\| \|x\| \cos(\theta_{(w,x)})$
- Decoupled general form : $f_d(w,x) = h(\|w\|, \|x\|) \cdot g(\theta_{(w,x)})$

On Better Modeling of the Intra-class Variation

- angular function 은 오직 angle 만 입력으로 받기 때문에, 설계 하기에 상대적으로 쉽다.
- magnitude function은 w, x 의 norm을 입력으로 받기 때문에, 설계 하기에 상대적으로 어렵다.
- $\|w\|$ 은 kernel 그 자체의 가중치이기 때문에, 입력의 intra-class variation 보다는 kerenl 자체에 중요성을 가진다. 따라서, 우리는$\|w\|$ 를 magnitude function 에 포함시키지 않는다. (모두 같은 중요도로 할당 시킨다)
- $\|w\|$ 을 모두 같은 중요도로 할당시키면, 가능한 많은 kernel에 기반하여 network 가 decision을 하기 때문에, 일반화 성능이 증가한다. 하지만, representational power 는 감소할 수 있다.
- $\|w\|$ 을 다시 magnitude function 으로 가져와서 사용하면, weighted decoupled operators 로 사용가능하다. (weighted decoupled operators 를 만드는 부분에 대해서는 뒷 절에서 다룰 예정이다.)

Bounded Decoupled Operators

- $|f_d(w,x)| \leq c$ where c : *positive constant* 처럼 bounded 되어있는 decoupled operators 에 대하여 설명한다. (설명의 편의성을 위하여 magnitude function 에서 weight의 norm 은 제외한다.)
- **Hyperspherical Convolution**
  - $h(\|w\|, \|x\|) = \alpha$ 라고 가정하면, 우리는 아래 수식처럼 decoupled 된 hyperspherical convolution(SphereConv) 를 얻게된다.

    $f_d(w,x) = \alpha \cdot g(\theta_{(w,x)}), \alpha > 0$

  - α 는 output의 scale을 조절하며, $g(\theta_{(w,x)})$ 은 unit hypersphere 의 geodesic distance 에 의해 결정된다. 일반적으로 결과물은 -1 부터 1 까지의 값으로 결정된다. 따라서, 최종 결과물은 $[-\alpha, \alpha]$ 로 정해진다. 주로, α 의 값으로 1을 사용하는데, 이때는 SphereConv와 유사해진다.
  - 기하학적으로, SphereConv는 w 와 x 를 hypersphere 로 사영시킨 후에 ($g(\theta) = cos(\theta)$ 일 때) 내적을 수행한다.
  - Sphereconv에 따르면, 네트워크의 수렴속도가 증가한다.
- **Hyperball Convolution**
  - Hyperball Conv. 는 $h(\|w\|, \|x\|) = \alpha min(\|x\|, \rho)/\rho$ 를 사용한다.

    $$f_d(w,x) = \alpha \cdot \frac{min(\|x\|, \rho)}{\rho} \cdot g(\theta_{w,x})$$

  - ρ 는 saturation threshold 를 조정하게 된다.
    - 만약, $\|x\|$ 가 ρ 보다 커지게 된다면, magnitude 는 saturation 되면서 α 값을 출력한다.
    - 반대의 경우에, magnitude function 은 $\|x\|$ 를 따라서 선형적으로 증가하게 된다.
  - BallConv 는 SphereConv 에 비해 조금 더 유연하고 강인하다. SphereConv는 angle값만을 이용하기 때문에 w 와 같은 방향의 x 일 경우에 무조건 maximum output이 나오게 된다. 만약 매우 norm이 매우 작은 x 가 입력으로 들어오게 될 경우 이를 amplify 하는 형태가 되는데, 이는 perturbation 에 취약한 형태가 될 수 있다는 것을 의미한다.
  - 반면에 BallConv는 x 의 norm이 작을 경우에, output도 작아질 수 있도록 한다. 또한, x 의 norm이 작다는 것은 local patch가 정보가 별로 없고, 강조되면 안좋을 것을 뜻하기도 한다.
- **Hyperbolic Tangent Convolution**
  - TanhConv는 BallConv에서 사용하였던 stepfunction 을 hyperbolic tangent function 으로 대체하였다.

    $f_d(w,x) = \alpha\, tanh(\frac{\|x\|}{\rho}) \cdot g(\theta_{w,x})$

  - ρ 의 값은 decay curve를 조정하는데 사용된다.
  - BallConv의 smooth version 으로 바라볼 수 있다.

Unbounded Decoupled Operators

- **Linear Convolution (LinearConv)**
  - 가장 간단한 Unbounded decoupled operator의 형태중 하나는 LinearConv 이다.

    $f_d(w,x) = \alpha \ \|x\| \cdot g(\theta_{w,x})$

  - LinearConv는 weights를 hypersphere에 사영시키고, slope를 컨트롤 할 수 있는 파라미터가 있다는 점에서 기존 convolution 과 다르다.
- **Segmented Convolution (SegConv)**
  - SegConv 는 조금 더 flexible 한 multi-range linear function 이다. LinearConv와 BallConv는 SegConv에 포함될 수 있다.

$$f_d(w,x) = \begin{cases} \alpha \|x\| \cdot g(\theta_{(w,x)}), & 0 \leq \|x\| \leq \rho \\ (\beta \|x\| + \alpha\rho - \beta\rho) \cdot g(\theta_{(w,x)}), & \rho < \|x\| \end{cases}$$

- **Logarithm Convolution (LogConv)**
  - unbounded 형태의 smooth decoupled operator 이다.

    $f_d(w,x) = \alpha \, log(1 + \|x\|) \cdot g(\theta_{w,x})$

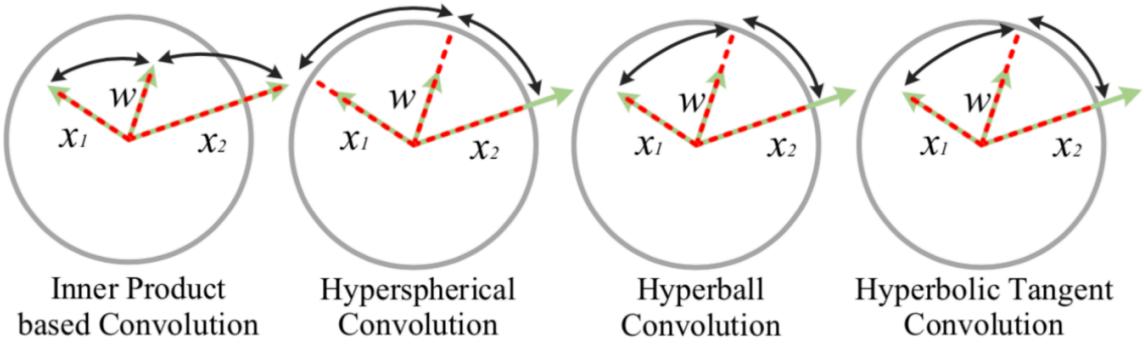
Properties of Decoupled Operators

- **Operator Radius**
  - Operator radius 는 magnitude function에서 gradient가 변하는 점을 표기하기 위해 필요하며, 앞으로 ρ 로 표기한다.
  - Operator radius 는 magnitude function 의 2 단계를 미분하며, 이 2 단계는 서로 다른 gradient 범위를 갖고 있기 때문에, optimization 과정동안 서로 다르게 행동하게 된다.
  - BallConv 의 경우 $\|x\|$ 가 ρ 보다 작을경우, magnitude function 은 활성화되며, $\|x\|$ 와 함께 선형적으로 증가할 것이다. 하지만 $\|x\|$ 가 ρ 보다 크게 될 경우, magnitude function 은 deactivated 되며, 상수값을 결과로 낼 것이다.
  - SegConv 의 경우, $\|x\| = \rho$ 는 magnitude function의 기울기가 변하는 점을 나타낼 것이다.
- **Boundedness**
  - Decoupled operator 의 boundedness 는 convergence speed 와 robustness 에 영향을 줄 것이다.
    - Bounded operators 는 SGD를 사용하는 신경망 트레이닝에서 조금 더 좋은 problem conditioning을 이끌어 줄 수 있다.
    - Bounded operators 는 출력값의 variance를 작게 만들어줄수 있으며, internal covariate shift problem 을 해결해 줄 수 있다.
      - Internal covariate shift problem ?
        - Gradient vanishing problem 의 요인중에 하나이다.
        - Covariate shift는 train data와 test data의 data distribution 이 다른 현상을 의미한다.

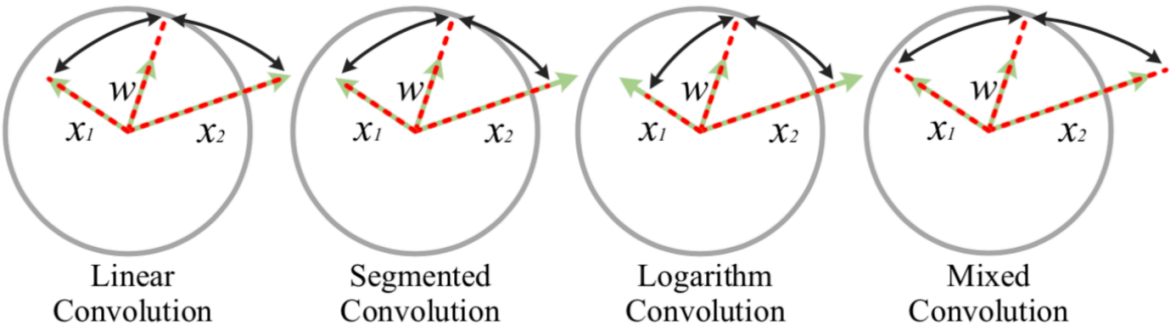
        - 각 layer들의 입력 distribution 이 consistent 해야하는데, 이렇지 못한 현상을 internal covariate shift 라고 정의한다.
        - BN(Batch Normalization)을 사용하여 internal covariate shift 를 감소시킬 수 있고, learning rate를 조금 더 크게 사용 가능하다.
    - Neural network 에서의 Lipschitz constant를 제약하여 전체 신경망이 smooth하도록 만들 수 있다. (Lipschitz regularity of deep neural networks: analysis and efficient estimation 논문 참조)
        - Lipschitz constant of neural network는 neural network가 adversarial perturbation에 강인하게 대응할 수 있는지에 대하여 관계가 있다.
    - 하지만, Unbounded operators가 approximation power, flexibility 측면에서는 bounded operators 보다 좋은 효과를 가진다.
- **Smoothness**
    - Magnitude function의 smoothness는 approximation ability, convergence 와 관련된다.
    - Smooth magnitude function 은 조금 더 좋은 approximation rate를 가지고, 안정적이고 빠른 convergence 를 할 수있게 한다. 하지만, computationally expensive 하기 때문에, smooth function 으로 approximation 하는 것은 어렵다.

Geometric Interpretations

- 모든 Decoupled Operator는 아래 그림과 같이 표현된다.
- Bounded Decoupled Operators



- Unbounded Decoupled Operators



Angular Function

- Linear angular activation

$$g(\theta_{(w,x)}) = -\frac{2}{\pi}\theta_{(w,x)} + 1$$

- Cosine angular activation

$$g(\theta_{(w,x)}) = \cos(\theta_{(w,x)})$$

- Sigmoid angular activation

$$g(\theta_{(w,x)}) = \frac{1+\exp(\frac{-\pi}{2k})}{1-\exp(\frac{-\pi}{2k})} \cdot \frac{1-\exp(\frac{\theta_{(w,x)}}{k} - \frac{\pi}{2k})}{1+\exp(\frac{\theta_{(w,x)}}{k} - \frac{\pi}{2k})}$$

- Square cosine angular activation

$$g(\theta_{(w,x)}) = sign(\cos(\theta)) \cdot \cos^2(\theta)$$

- 아래 그림은 Magnitude function 과 Angular function 의 input 에 따른 변화를 나타낸다.

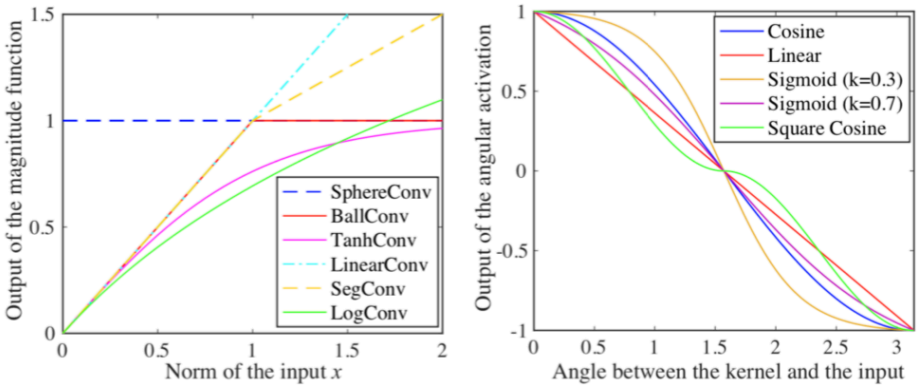


Figure 3: Magnitude function ($\rho = 1$) and angular activation function.

Weighted Decoupled Operators

- Weighted Decoupled Operator 는 앞서 언급했던 operator의 성능이 더 좋았기 때문에, 간단하게만 살펴볼 예정이다.
- Linearly Weighted Decoupled Operator
  - 앞서서 본 Linear Decoupled Operator 에 Weight의 norm 이 곱해진 형태이다.

$$f_d(w,x) = \alpha \|w\| \cdot g(\theta_{(w,x)})$$

- Nonlinearly Weighted Decoupled Operator
  - tanh conv operator에 Weight의 norm 이 포함된 형태이다.

$$f_d(w,x) = \alpha \tanh(\frac{1}{\rho}\|x\| \cdot \|w\| \cdot g(\theta_{(w,x)}))$$

- 위의 수식은 아래와 같은형태로도 표현이 가능하다. 아래 수식은 tanh 가 2번 연산되기 때문에, 비선형성이 조금 더 증가되는 효과가 있다. 물론 그 만큼 학습을 하기 어려워지는 문제도 있을 것이다.

$$f_d(w,x) = \alpha \tanh(\frac{1}{\rho}\|w\|) \cdot \tanh(\frac{1}{\rho}\|x\|) \cdot g(\theta_{(w,x)})$$

Learnable Decoupled Operators

- 하이퍼 파라미터를 학습 가능하게 만든다는 것은 Representational power 를 증가시킬 수 있지만, 더 많은 Training data가 요구되어야 한다는 Trade-Off 문제를 가지고 있다.
- 이 Trade-Off 문제를 효율적으로 해결하기 위하여, 학습가능한 하이퍼 파라미터는 ρ(operator radius) 1개만 설정하였다.

Experiments

- 아래 표는 Weighted Decoupled 의 형태가 성능향상에 도움이 되지 않는다는 사실을 보여준다.

Method	Error
Linearly Weighted Decoupled Operator	22.95
Nonlinearly Weighted Decoupled Operator (Eq. (15))	23.03
Nonlinearly Weighted Decoupled Operator (Eq. (16))	23.38
Decoupled Operator (Standard Gradients)	23.09
Decoupled Operator (Weight Projection)	**21.17**
Decoupled Operator (Weighted Gradients)	21.45

- 아래 표는 Weight Project과, Weight Gradients 방식이 성능향상에 도움이 된다는 사실을 보여준다.

Method	Error
Linearly Weighted Decoupled Operator	22.95
Nonlinearly Weighted Decoupled Operator (Eq. (15))	23.03
Nonlinearly Weighted Decoupled Operator (Eq. (16))	23.38
Decoupled Operator (Standard Gradients)	23.09
Decoupled Operator (Weight Projection)	**21.17**
Decoupled Operator (Weighted Gradients)	21.45

- 아래 표는 Batch Normalization 없이도, 기존 CNN baseline 의 성능 보다 좋다는 사실을 보여준다.
  Decoupled Operator 를 통하여 internal covariate shift problem 을 완화시켰기 때문에, BN 없이도 학습이 안정적으로 되는 것을 확인할 수 있다.
  (CIFAR 10, CIFAR 100 데이터에서 실험을 진행하였다.)

Method	Linear	Cosine	Sq. Cosine
CNN Baseline	-	35.30	-
LinearConv	33.39	31.76	N/C
TanhConv	**32.88**	31.88	**34.26**
SegConv	34.69	**30.34**	N/C

- 아래 표는 ReLU 없이도 학습이 잘 진행된다는 사실을 보여준다. Decoupled Operator가 비선형성을 증가시켜 주기 때문에 ReLU가 없이도 학습이 가능하다. (CIFAR 10, CIFAR 100 데이터에서 실험을 진행하였다.)

Method	Cosine w/o ReLU	Sq. Cosine w/o ReLU	Cosine w/ ReLU	Sq. Cosine w/ ReLU
Baseline	58.24	-	26.01	-
SphereConv	33.31	25.90	26.00	26.97
BallConv	**31.81**	25.43	25.18	26.48
TanhConv	32.27	25.27	25.15	26.94
LinearConv	36.49	24.36	**24.81**	25.14
SegConv	33.57	**24.29**	24.96	**25.04**
LogConv	33.62	24.91	25.17	25.85
MixConv	33.46	24.93	25.27	25.77

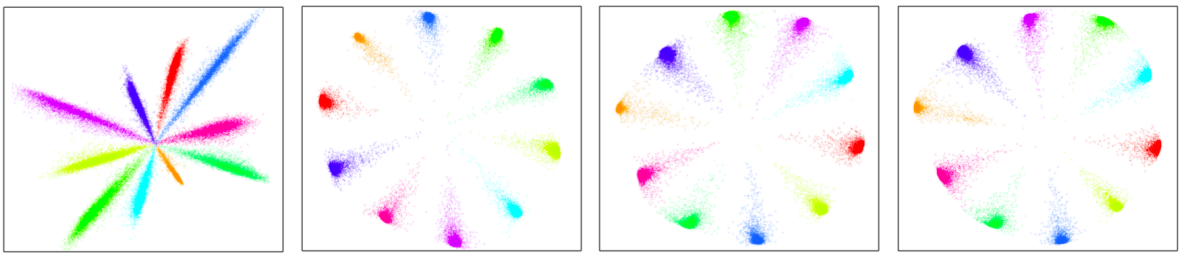
Table 3: Testing error rate (%) of plain CNN-9 on CIFAR-100. Note that, BN is used in all compared models. Baseline is the original plain CNN-9.

- 아래 표는 Adversarial attack 에 대해서 기존 baseline보다 성능이 우수하다는 사실을 보여준다. (Black-box attack, White-box attack 에 대해서 실험을 수행하였고, CIFAR-10 데이터를 사용했다.)

Attack	Target models			
	Baseline	SphereConv	BallConv	TanhConv
None	85.35	88.58	91.13	**91.45**
FGSM	50.90	**56.71**	49.50	50.61
BIM	36.22	**43.10**	27.48	29.06
None	83.70	87.41	87.47	**87.54**
FGSM	77.57	76.29	78.67	**80.38**
BIM	78.55	77.79	80.59	**82.47**

Attack	Target models			
	Baseline	SphereConv	BallConv	TanhConv
None	85.35	88.58	91.13	**91.45**
FGSM	18.82	43.64	50.47	**52.60**
BIM	8.67	8.89	7.74	**10.18**
None	83.70	87.41	87.47	**87.54**
FGSM	78.96	**85.98**	82.20	81.46
BIM	7.96	**35.07**	17.38	19.86

- 아래 그림은 DCNet 이 기존 CNN Network 보다 inter-class 를 잘 분리하고, inter-class 를 잘 뭉치게 하는 것을 보여준다.



Baseline CNN (Original Convolution) DCNet (SphereConv) DCNet (BallConv) DCNet (TanhConv)

Conclusion

- Decoupled Operator를 통해 기존 CNN 보다 효율적으로 학습할 수 있다는 것을 보여주었다.