How Does DCNN Make Decisions?

Yi Lin *, Namin Wang, Xiaoqing Ma, Ziwei Li, Gang Bai **
E-mail: * 2120180463@mail.nankai.edu.cn, ** baigang@nankai.edu.cn

Introduction

Problems :
DCNN is vulnerable to small perturbations and exhibit poor interpretability.

Results :
This phenomenon leads to the limitations of DCNN’s applications in the security and trusted computing.

Our solution:
Start with the decision-making method of DCNN to find out the reason for its low robustness and interpretability.

Introduction (cont.)

The main **contributions** of the paper are:

- We put forward the “point-wise activation” and further verify it by extensive experiments;
- We point out the effect of “point-wise activation” on DCNN’s uninterpretable classification and pool robustness and reveal the contradiction between the traditional and DCNN’s convolution kernel functions;
- We distinguish decision-making interpretability from semantic interpretability, and indicate the future improvements of DCNN.

Point-Wise Activation

The Nonlinear Activation Function

$$Out = \mathrm{ReLU}\left(Net\right) = \max\left(0, Net\right) \qquad (1)$$

max() represents the operation to take the maximum. Variables *Net* and *Out* are the input and output scalars of layers considered.

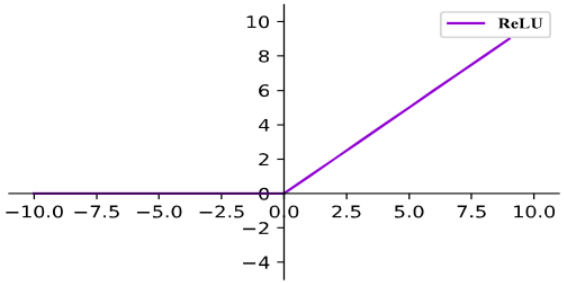


Fig.1 The image of function ReLU.

The Architecture of Convolutional Layer

$$Net = w_1 x_1 + \mathrm{L} + w_d x_d + b = \sum_{i=1}^{d} w_i x_i + b = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b \qquad (2)$$

w and ***x*** are the weight vectors that denote the convolution kernel function (namely a filter) and a local area of the input image (or feature map) considered respectively. Scalar *Net* and *b* stand for the output and a bias term.

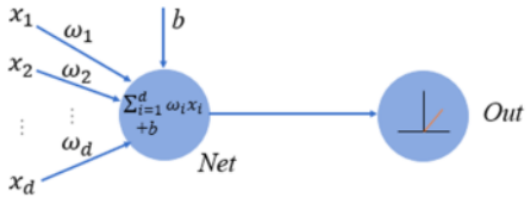


Fig.2 The architectures of the convolutional layer and function ReLU.

Point-Wise Activation(cont.)

- *The "Point-wise Activation" Hypothesis*

ReLU: Activated without restricting the number of positive terms of the products.

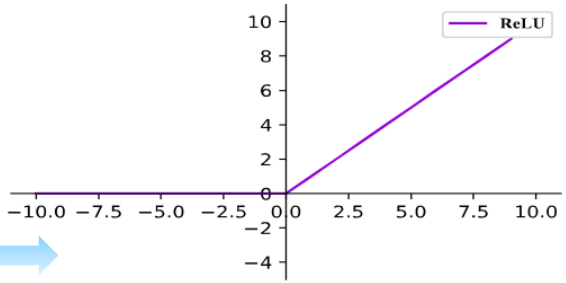


Fig.1 The image of function ReLU.

The "point-wise activation" :
Pixels(points) used to be combined as features to distinguish objects can be rather few in DCNN. (Unless otherwise stated, pixels(points) in our paper are of single-channel.)

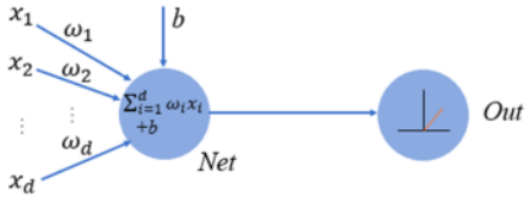


Fig.2 The architectures of the convolutional layer and function ReLU.

Experiments And Analyzes

Network Training Experiments

Dataset:CIFAR10 Model:VGG16, Resnet18

the 3σ principle:

$$P(\mu - 3\sigma < x \leq \mu + 3\sigma) = 99.7\% \qquad (3)$$

where μ and σ stand for the mean and standard deviation of the data. While P indicates the probability.

Conclusion: Due to the 3σ principle of the normal distribution, we consider trimming out parameters of small contributions in order to acquire a better understanding of the limits of current DCNN architectures.

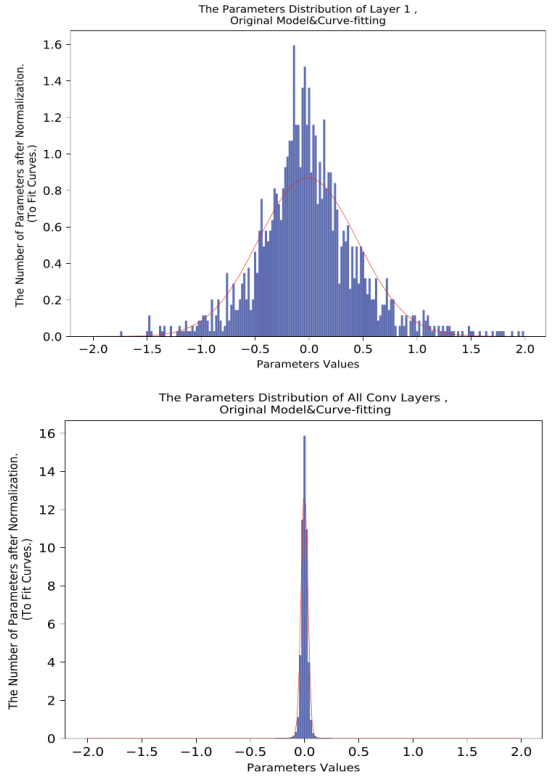


Fig.3. The parameters distributions of VGG16 model. Top: the first convolutional layer (a maximum outlier and two minimum outliers are deleted); bottom: all convolutional layers (a maximum outlier and two minimum outliers are deleted).

Experiments And Analyzes(cont.)

Network Compression Experiments

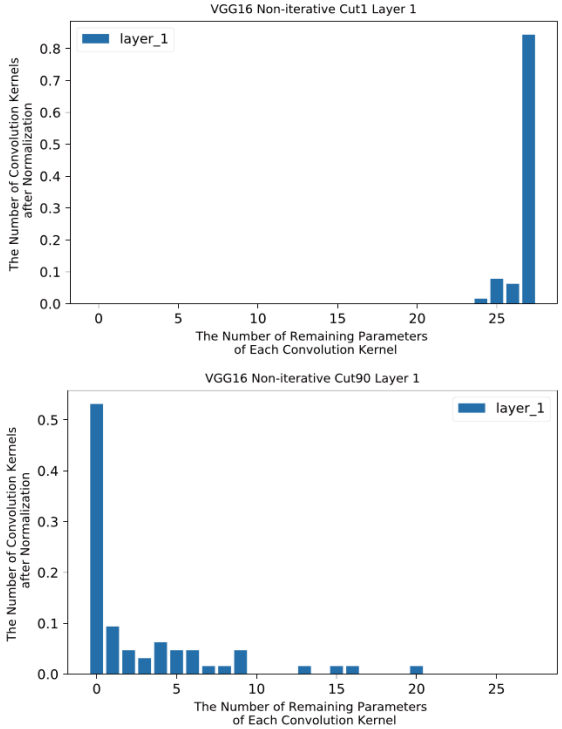


Fig. 4. The changes of the parameters in convolution kernels at different pruning rate

Dataset:CIFAR10

Model:VGG16, Resnet18

Non- iterative Pruning:
1.Set the pruning rates to 1%, 10%, 20% ... 80%, 90%, 99%, respectively.
2.Prune networks according to the pruning rates as well as using masks to record them.
3.The parameters are fine-tuned with masks, and the remaining effective parameters of each convolution kernel are counted.

Experiments And Analyzes(cont.)

Network Compression Experiments

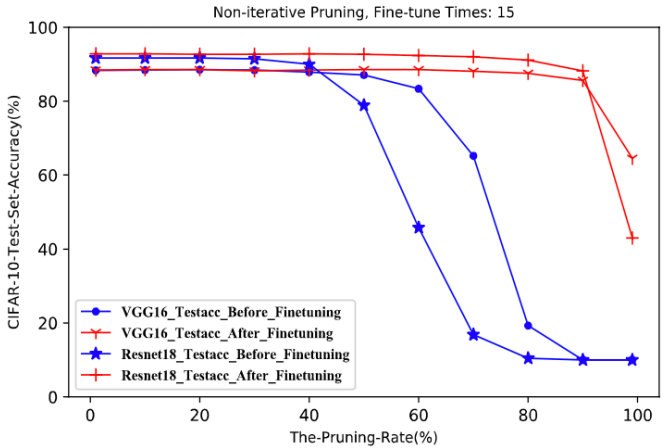


Fig. 5. Non-iterative pruning accuricies.

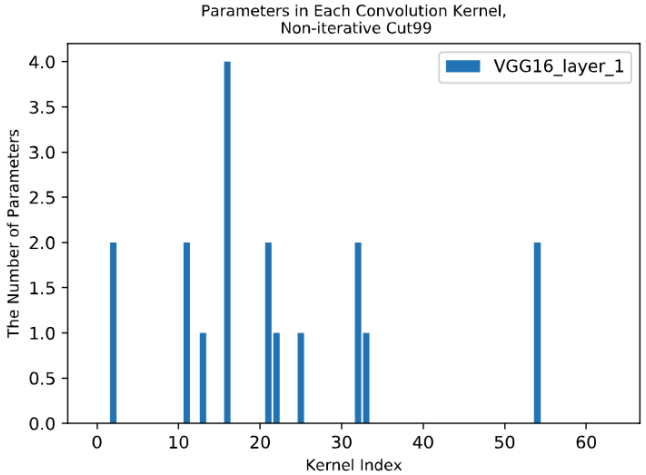


Fig. 6. The parameters number in each convolution kernel of VGG16 pruning model’ s first convolutional layer(accuricy:64.58%).

Conclusion: DCNN’ s decision-making mainly determined by a few critical pixels. These pixels are diifficult to construct enough features that people can understood.

Experiments And Analyzes(cont.)

Adversarial Attack Experiments

	OriginAcc	One-Pixel	Three-Pixels	Five-Pixels
Resnet	86.61%	31.2%	74.6%	78.6%

TABLE I. **THE RESULTS OF CONDUCTING ONE-PIXEL(THREE CHANNELS), THREE-PIXELS(THREE CHANNELS) AND FIVE-PIXELS(THREE CHANNELS) NON-TARGETED ATTACK[21] ON THE RESNET.** ORIGINACC IS THE ACCURACY ON THE NATURAL TEST DATASET(CIFAR-10). ONE-PIXEL, THREE-PIXELS AND FIVE-PIXELS INDICATE THE ACCURACY OF CONDUCTING NON-TARGETED ATTACKS AT CORRESPONDING TIMES ON 500 SAMPLES RANDOMLY EXTRACTED ON THE CIFAR-10 DATASET.

	AllConv	NiN	VGG16
OriginAcc	85.6%	87.2%	83.3%
Non-targeted	68.71%	71.66%	63.53%

TABLE II. **THE RESULT【1】OF CONDUCTING ONE-PIXEL ATTACK ON THREE DIFFERENT TYPES OF NETWORKS.** ORIGINACC IS THE ACCURACY ON THE NATURAL TEST DATASET(CIFAR-10). NON-TARGETED INDICATE THE ACCURACY OF CONDUCTING NON-TARGETED ATTACKS.

one- pixel attack[1]:generating one-pixel adversarial perturbations based on differential evolution (DE) .

Conclusion: DCNN enables very few pixels play a key role in classifications. The results verify the conclusion of network compression experiments again.

[1]J. Su, D. V.Vargas, and S. Kouichi. "One pixel attack for fooling deep neural networks," IEEE Transactions on Evolutionary Computation, 2017.

Experiments And Analyzes(cont.)

Single Value Experiments

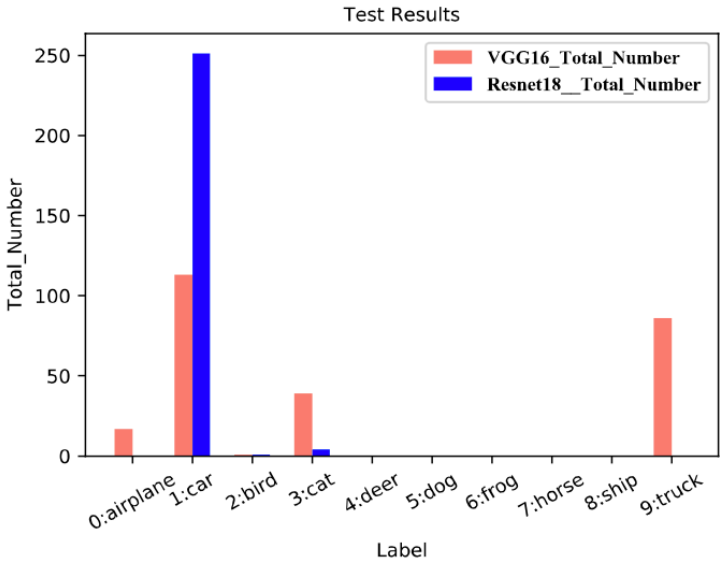


Fig. 7. Test results of VGG16 and Resnet18 models.

Dataset:CIFAR10

Model:VGG16, Resnet18

Inputs: A series of images of a single pixel value. (the pixel values of those images are integers that increase from 0 to 255, for a total of 256.)

Conclusion: As long as the pixel exists, DCNN can classify regardless of whether it has intelligible features such as shapes or not.

Experiments And Analyzes(cont.)

Hard Sample Experiments.

Fig. 8. The results for hard sample experiments. Middle: an image with all pixels initialized to 1. Left: the image after being modified which is classified by VGG16-like model as category 1 (car) with the probability no less than 0.9. Right: the image after being modified which is classified by Resnet18-like model as category 1 (car) with the probability no less than 0.9.

$$\mathbf{w}_0^{(new)} = \mathbf{w}_0 - 0.001 \times \mathbf{Grad}(\mathbf{w}_0) \quad (4)$$

$$\mathbf{img}^{(new)} = \left(\mathbf{img} * \mathbf{w}_0^{(new)}\right) / \mathbf{w}_0 \quad (5)$$

Dataset: CIFAR10

Model:VGG16-like, Resnet18-like

(Add a dot product layer (dot product between the input image and the tensor $\mathbf{w}_0$) to the input of the VGG16 and Resnet18 framework, respectively.)

Algorithm:

1. Use an image **img** with all pixels initialized to 1 and assign a category to the image artificially.
2. Calculate loss and gradient **Grad**(**w**0) .
3. Use the gradient **Grad**(**w**0) to update **w** to **w**(new) by (4).
4. Follow the (5), we can get a newly changed image **img** (new) .
5. Replace **img** with **img** (new) , then go to the first step. Loop until the output category turns to the assigned one.
6. The resulting **img**(new) of which we need is generated.

Experiments And Analyzes(cont.)

- Hard Sample Experiments.

Fig. 8. The results for hard sample experiments. Middle: an image with all pixels initialized to 1. Left: the image after being modified which is classified by VGG16-like model as category 1 (car) with the probability no less than 0.9. Right: the image after being modified which is classified by Resnet18-like model as category 1 (car) with the probability no less than 0.9.

Conclusion: It is rather difficult for human observers to recognize the distribution of pixel sets used by DCNN.

Discussions

The Semantic Uncertainty of DCNN’ s Convolution Results

- Convolution kernel functions in traditional computer vision, such as Gaussian smooth function, Sobel operators, are artificially designed in advance based on mathematical equations.

Contradiction

- The parameters in DCNN’ s convolution kernel function are updated to reduce the loss between the truth labels and real outputs.

- We consider the semantic uncertainty of DCNN’ s convolution results as the root cause of why DCNN's decision cannot be explained.

Discussions

The Robustness of DCNN’s Decisions

- From the perspective of visual decision-making, human observers tend to recognize objects based on their shapes, while DCNN based on pixel sets.

- From the perspective of high-dimensional space, pixel-based classifications are of pool robustness for each sampled point can be described as the content of category manifold boundary**.** Moreover, as far as “point-wise activation” concerned, DCNN’s classifications can be greatly affected by a few pixels, while "a few" itself means more susceptible to perturbations.

Discussions

About Interpretability and Improvements

- For interpretability, we consider dividing it into two types. One is decision-making interpretability, the other is semantic interpretable.

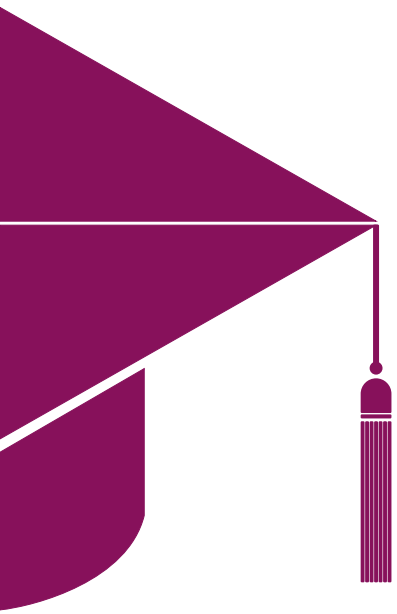
- For decision-making interpretability, most improvements are conducted by segmentation and visualization, which are to find the useful part in the original image for decision-making.

- For semantic interpretability, we hope to establish a relationship with human knowledge so that the classifications are semantically interpretable to humans.

CONCLUSION

CONCLUSION

- We provide the "point-wise activation" hypothesis and consider it contributes to the poor robustness of DCNN largely.
- The convolution kernel function of DCNN is different from the others used in the field of traditional computer vision for the methods to determine the parameters differ.
- In order to make credible decisions, DCNN's decision-making mechanism need to evolve towards the direction of semantics in the future.



Thank You!