Explainable Convolutional Neural Network

발표자 : 백인성

Data Mining
Quality Analytics

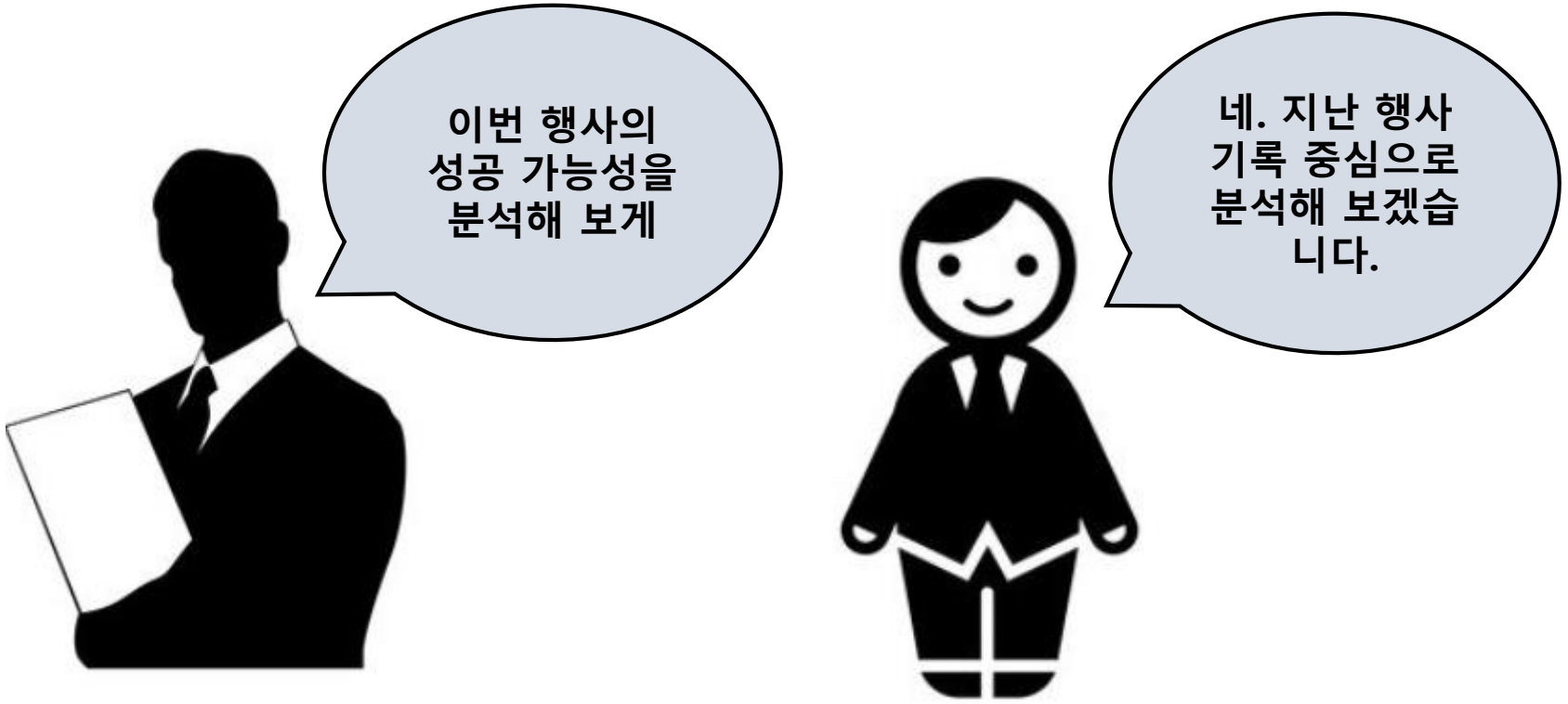
목차

1. Introduction

Introduction

❖ 예측 모델에서 해석이 왜 중요한가?

- ➢ 모델링이 핵심인 전문가는 예측 성능 향상이 주 목표이지만
- ➢ 실제 적용되는 산업 분야에서는 해석은 예측 성능만큼 중요함



[사업부 팀장] [1년차 신입사원]

Introduction

❖ 예측 모델에서 해석이 왜 중요한가?

- ➢ 데이터 분석가는 모델의 결과를 중심으로 이야기 하지만
- ➢ 실제로 사용하는 실무자들은 그 결과가 나온 이유를 더 궁금해 함

[1년차 신입사원]

[사업부 팀장]

Introduction

❖ 예측 모델에서 해석이 왜 필요한가?

- ➢ 현업 관계자와 소통을 원활하게 하기 위해
- ➢ 모델의 예측 성능에 대한 타당성을 직관적으로 보여주기 위해

[A사 팀장]

[B사 팀장]

Introduction

❖ 예측 모델의 해석이 필요한 이유는 아래의 3가지 이유로 요약할 수 있음

1. 예측 모델을 실제로 사용하는 사람들의 이해를 돕기 위해

2. 의사결정하는 사람들을 조금 더 쉽게 설득하기 위해

3. 구축한 예측 모델의 결과의 타당함을 직관적으로 보이기 위해

Introduction

- Interpretable Convolutional Neural Network(2018)
  - CNN 모델의 결과를 해석하기 위한 방법론 중 하나
  - 2018년도 CVPR에서 소개 되었고, 1년간 인용 회수는 65회

Interpretable convolutional neural networks

Q Zhang, Y Nian Wu, SC Zhu - Proceedings of the IEEE ..., 2018 - openaccess.thecvf.com

This paper proposes a method to modify a traditional convolutional neural network (CNN) into an interpretable CNN, in order to clarify knowledge representations in high conv-layers of the CNN. In an interpretable CNN, each filter in a high conv-layer represents a specific object part. Our interpretable CNNs use the same training data as ordinary CNNs without a need for any annotations of object parts or textures for supervision. The interpretable CNN automatically assigns each filter in a high conv-layer with an object part during the learning ...

☆ 99 65회 인용 관련 학술자료 전체 7개의 버전 »

Interpretable Convolutional Neural Networks

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu
University of California, Los Angeles

Abstract

This paper proposes a method to modify a traditional convolutional neural network (CNN) into an interpretable CNN, in order to clarify knowledge representations in high conv-layers of the CNN. In an interpretable CNN, each filter in a high conv-layer represents a specific object part. Our interpretable CNNs use the same training data as ordinary CNNs without a need for any annotations of object parts or textures for supervision. The interpretable CNN automatically assigns each filter in a high conv-layer with an object part during the learning process. We can apply our method to different types of CNNs with various structures. The explicit knowledge representation in an interpretable CNN can help people understand the logic inside a CNN, i.e. *what patterns are memorized by the CNN for prediction. Experiments have shown that filters in an interpretable CNN are more semantically meaningful than those in a traditional CNN. The code is available at https://github.com/zqs1022/interpretableCNN.*

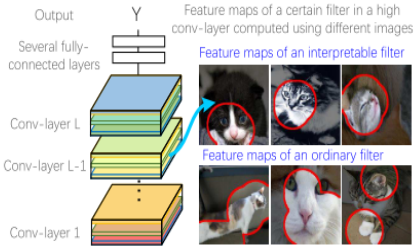


Figure 1. Comparison of a filter's feature maps in an interpretable CNN and those in a traditional CNN.

In fact, we can roughly consider the first two semantics as object-part patterns with specific shapes, and summarize the last four semantics as texture patterns without clear contours. Moreover, filters in low conv-layers usually describe simple textures, whereas filters in high conv-layers are more likely to represent object parts.

Therefore, in this study, we aim to train each filter in a

Introduction

- 논문의 저자 (Quanshi Zhang)
  - 상하이 대학교 부교수
  - 딥러닝 모델의 해석력을 높이려는 연구를 중점적으로 진행

Quanshi Zhang

张拳石

Associate Professor

John Hopcroft Center for Computer Science

School of electronic information and electrical engineering

Shanghai Jiao Tong University

Email: zqs1022 [AT] sjtu.edu.cn [知乎]

招生 Prospective Ph.D., Master, and undergraduate students:

I am looking for highly motivated students to work together on interpretability of neural networks, unsupervised and weakly-supervised learning, graph mining, and other frontier topics in machine learning and computer vision.

Please read "写给学生" and send me your CV and transcripts.

제목

Interpretable convolutional neural networks
Q Zhang, Y Nian Wu, SC Zhu
Proceedings of the IEEE Conference on Computer Vision and Pattern ...

Visual interpretability for deep learning: a survey
Q Zhang, SC Zhu
Frontiers of Information Technology & Electronic Engineering 19 (1), 27-39

Prediction of human emergency behavior and their mobility following large-scale disaster
X Song, Q Zhang, Y Sekimoto, R Shibasaki
Proceedings of the 20th ACM SIGKDD international conference on Knowledge ...

Modeling and probabilistic reasoning of population evacuation during large-scale disaster
X Song, Q Zhang, Y Sekimoto, T Horanont, S Ueyama, R Shibasaki
Proceedings of the 19th ACM SIGKDD international conference on Knowledge ...

Unsupervised skeleton extraction and motion capture from 3D deformable matching
Q Zhang, X Song, X Shao, R Shibasaki, H Zhao
Neurocomputing 100, 170-182

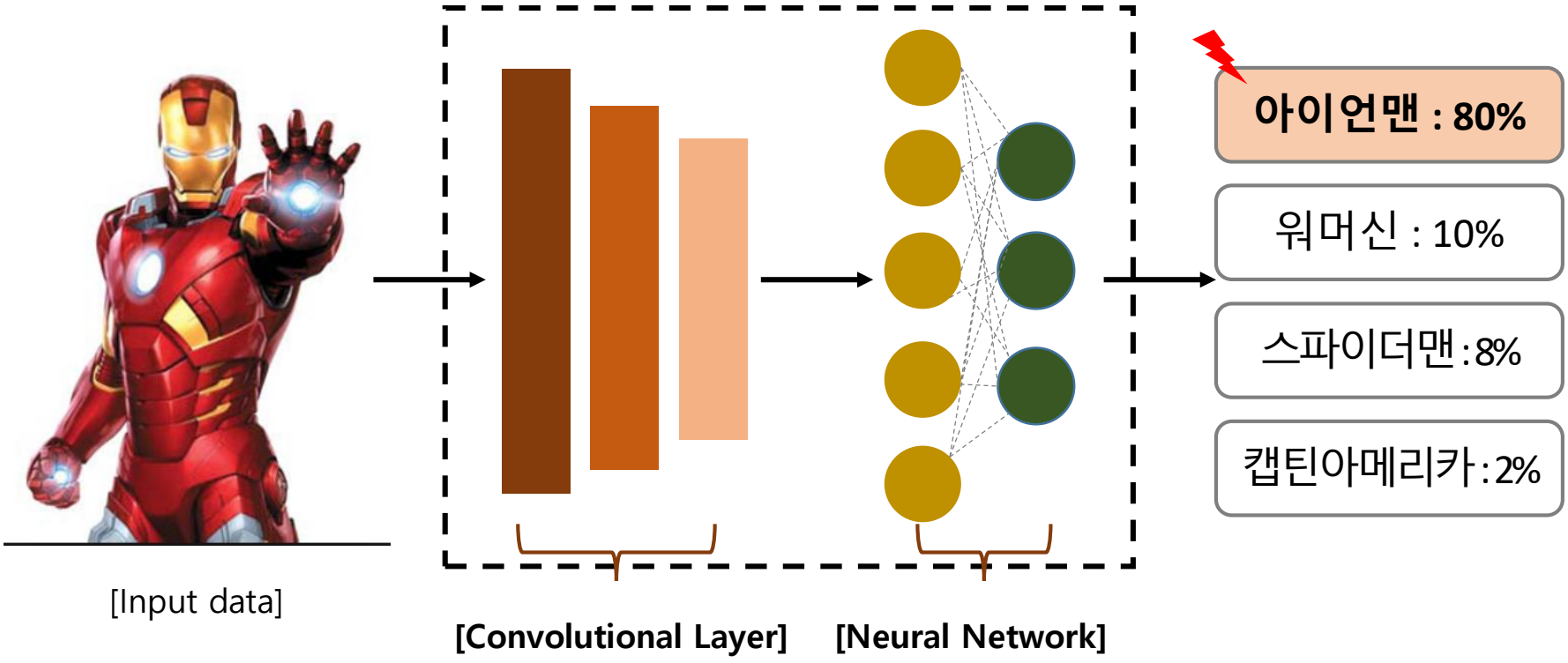
Interpreting cnn knowledge via an explanatory graph
Q Zhang, R Cao, F Shi, YN Wu, SC Zhu
Thirty-Second AAAI Conference on Artificial Intelligence

2. [CNN]Convolutional Neural Network

[CNN]Convolutional Neural Network

❖ Convolutional Neural Network 기본 구조

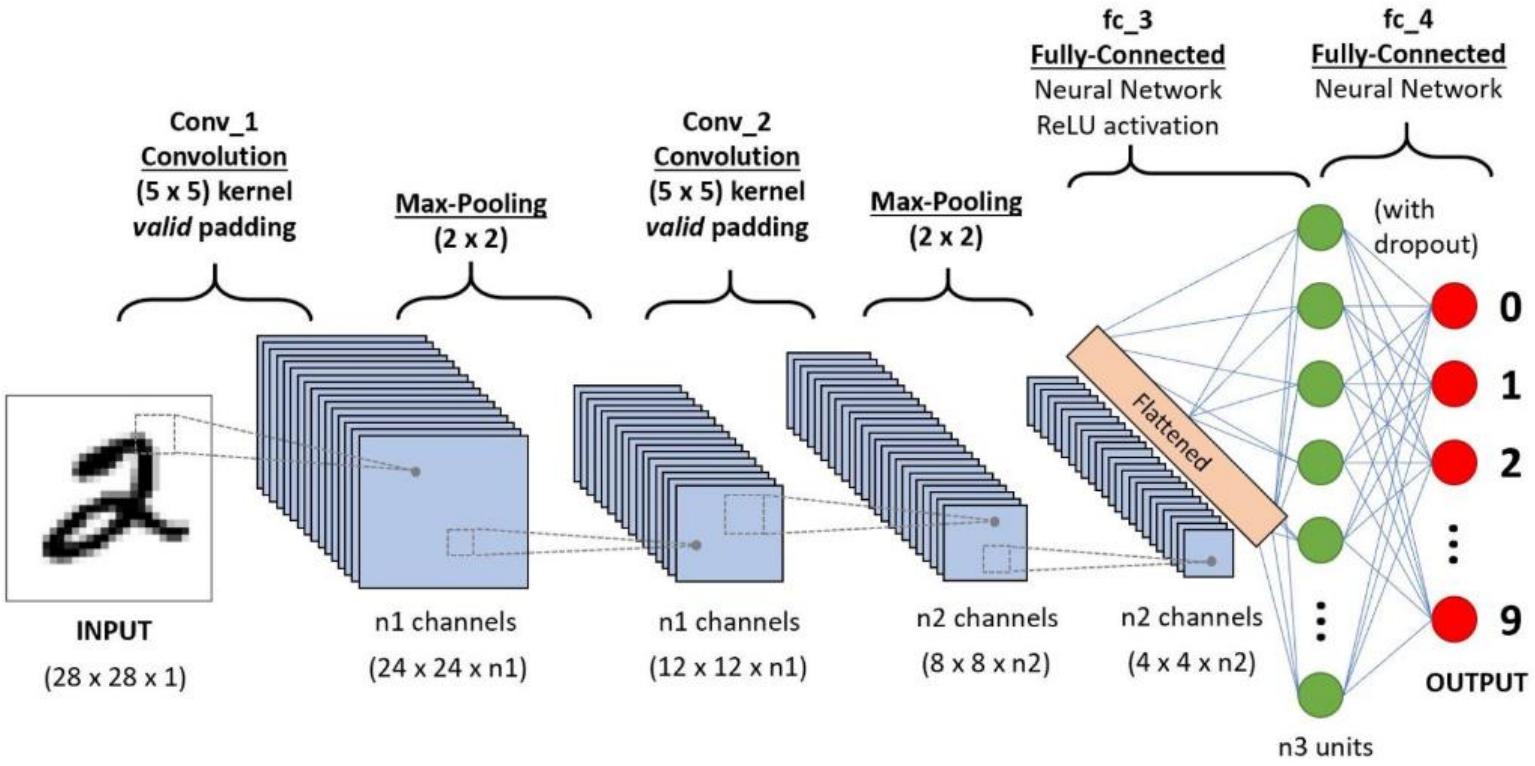
- ➢ Neural Network에 Convolution Layer를 사용한 방법론
- ➢ Object Detection, Classification 등 Visual Task에서 좋은 Performance 보임



[CNN]Convolutional Neural Network

❖ Convolutional Neural Network 세부 구조

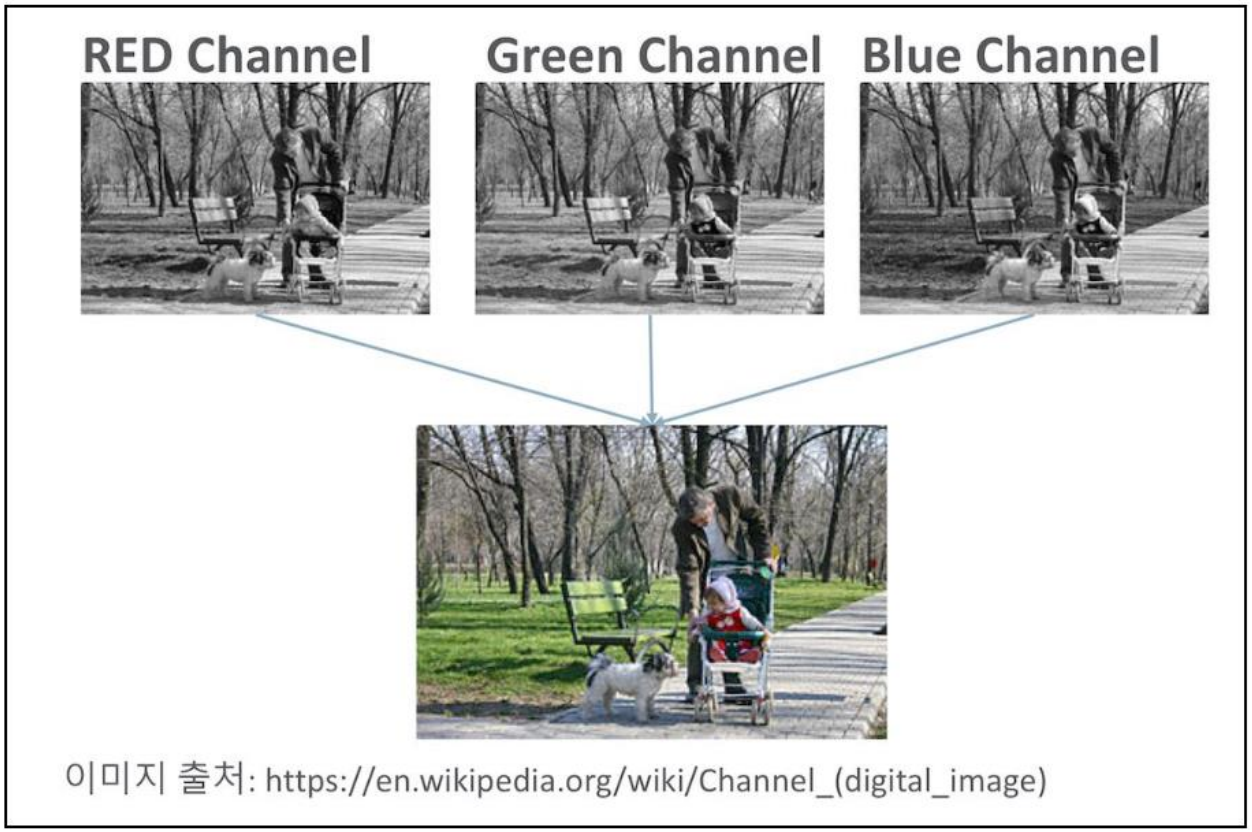
- ➢ Input data에 Convolution & Pooling을 활용하여 특징 추출
- ➢ 최종적으로 해당 Input이 어떤 Class에 속하는지 결정



[CNN]Convolutional Neural Network

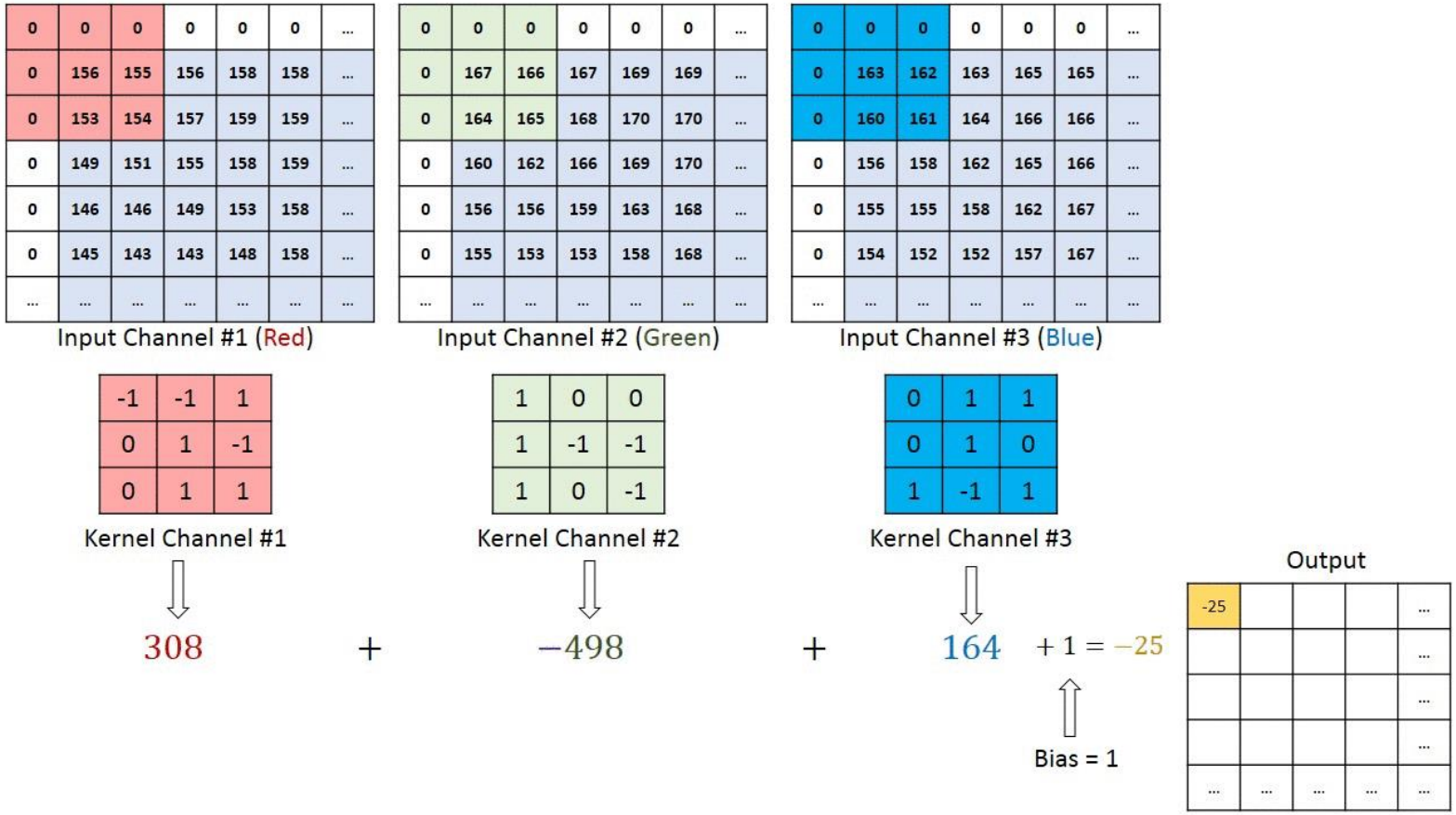
❖ Channel & Pooling

➢ Channel : Input이 이미지인 경우 그 이미지를 포현하는 R/G/B 각각을 의미함



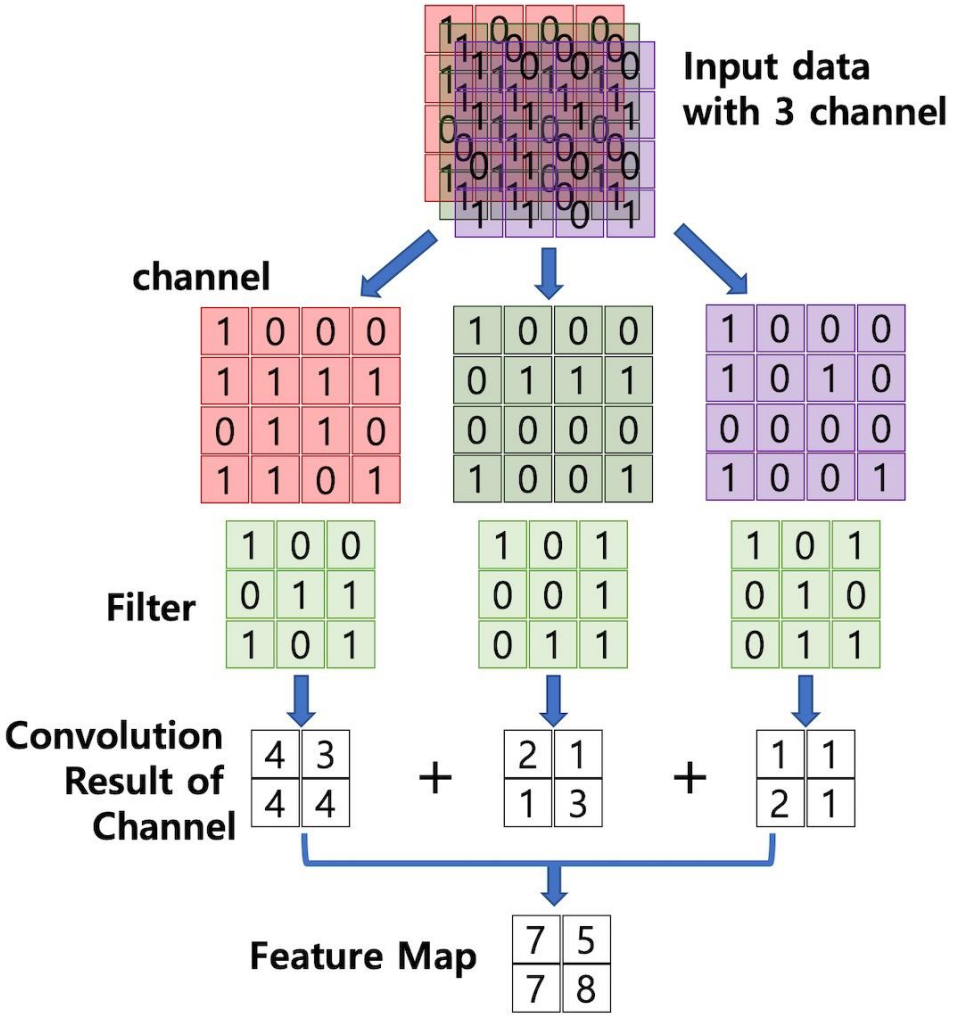
[CNN]Convolutional Neural Network

- Filter & Convolution
  - Filter : Input의 특징을 찾아내기 위한 공용 parameter
  - Convolution(합성곱) : Filter를 이동 시키면서 곱한 결과를 합산한 것



[CNN]Convolutional Neural Network

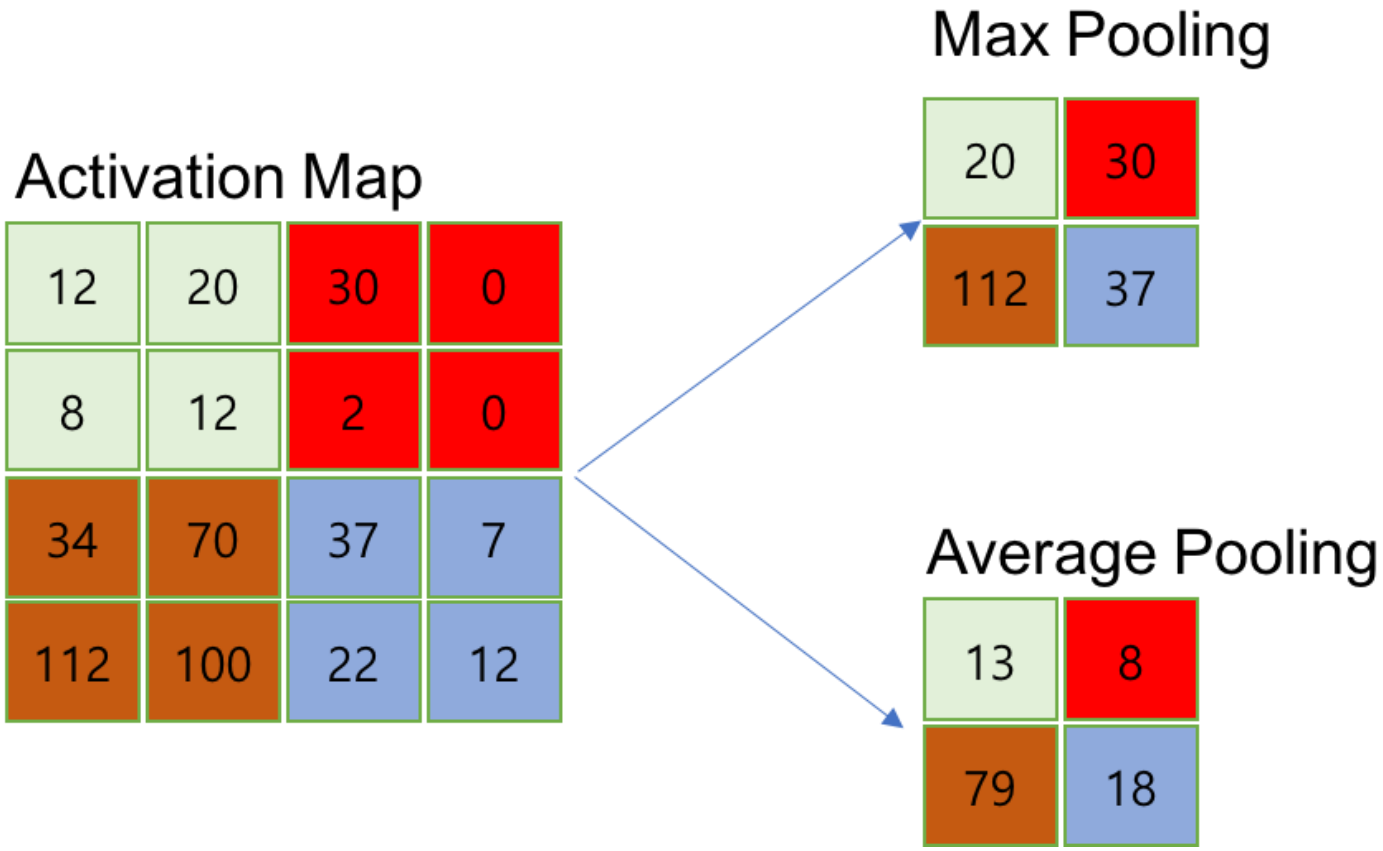
❖ Channel & Convolution & Filter 정리



[CNN]Convolutional Neural Network

❖ Channel & Pooling

➢ Pooling : Convolution Layer의 출력 값을 입력으로 받아 크기를 줄이거나 특정 데이터를 강조하는 용도로 사용됨, Filter와 달리 Parameter가 아님

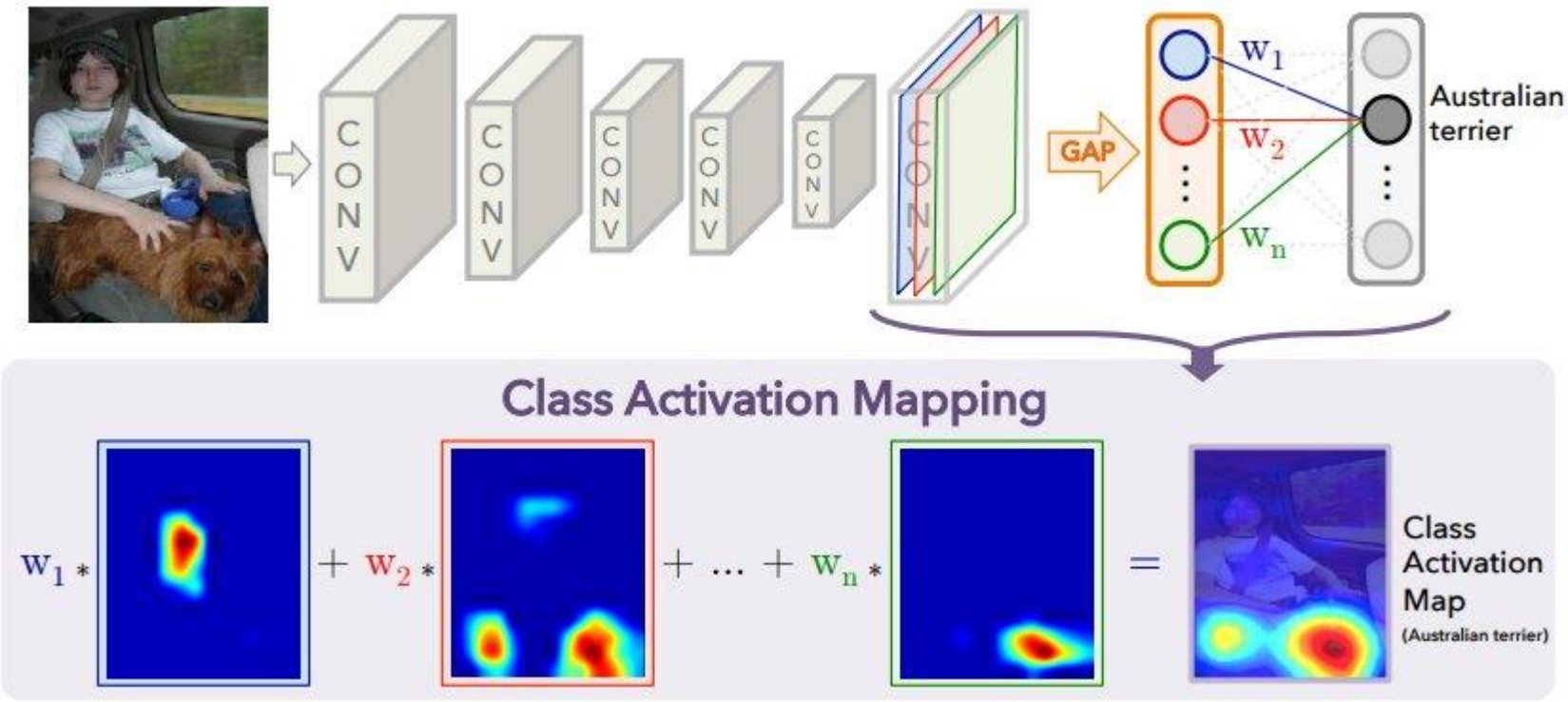


3. [CAM]Class Activation Map

[CAM]Class Activation Map

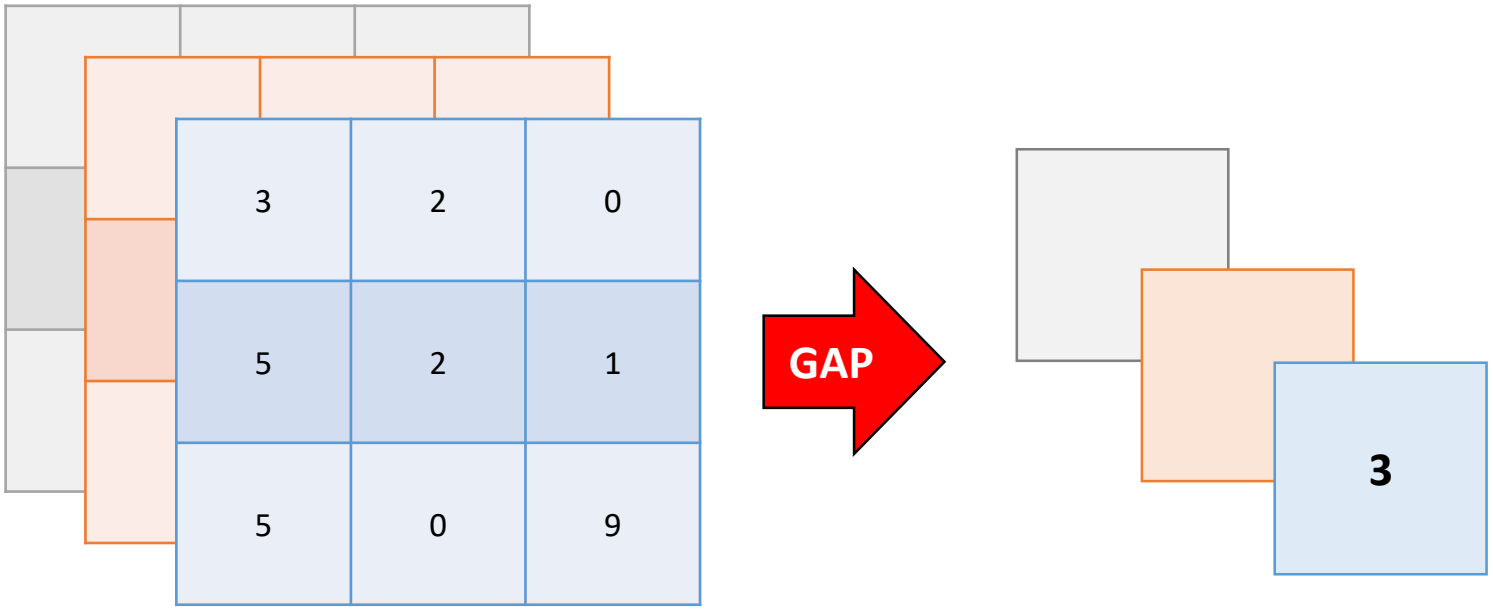
❖ Class Activation Map 구조

- ➢ CNN 구조에서 어떤 부분이 Class 결정에 큰 영향을 주었는지 확인 가능
- ➢ Flatten 대신 GAP을 적용, 마지막 판별 전까지 데이터 위치 정보 훼손되지 않음



[CAM]Class Activation Map

- GAP(Global Average Pooling)
  - CNN 마지막에 Flatten 방법 대신 사용
  - 결과는 Class개수와 동일, 채널별 평균값에 Softmax 함수 적용

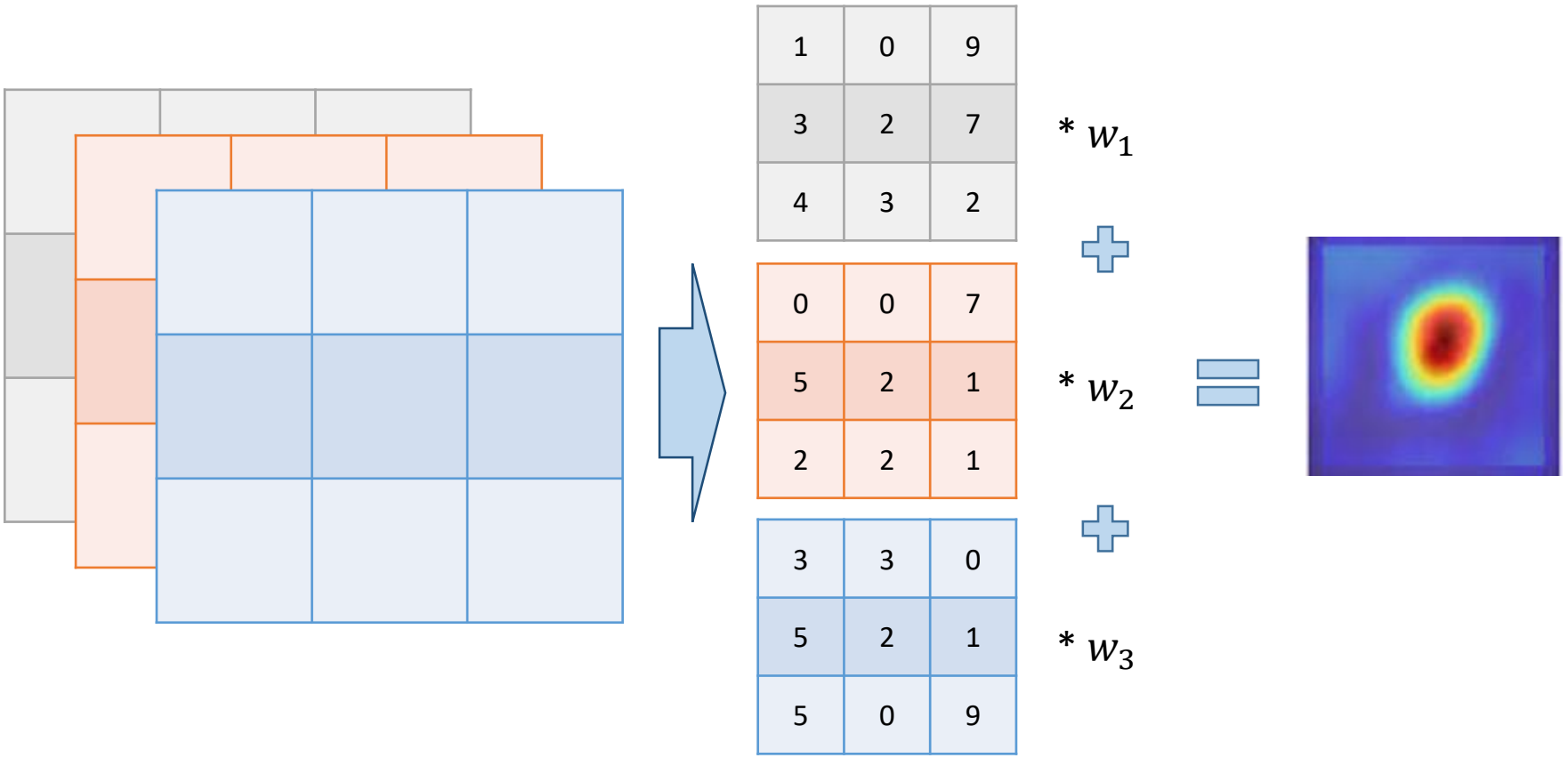


[3x3x3(Channel)]

[1x1x3(channel)]

[CAM]Class Activation Map

- Class Activation Map
  - 해당 채널의 각 값에 Weight 값을 곱한 수치를 중요도로 해석
  - 위처럼 계산된 모든 채널 값을 더하여 N x N 행렬을 Heat-map으로 그림

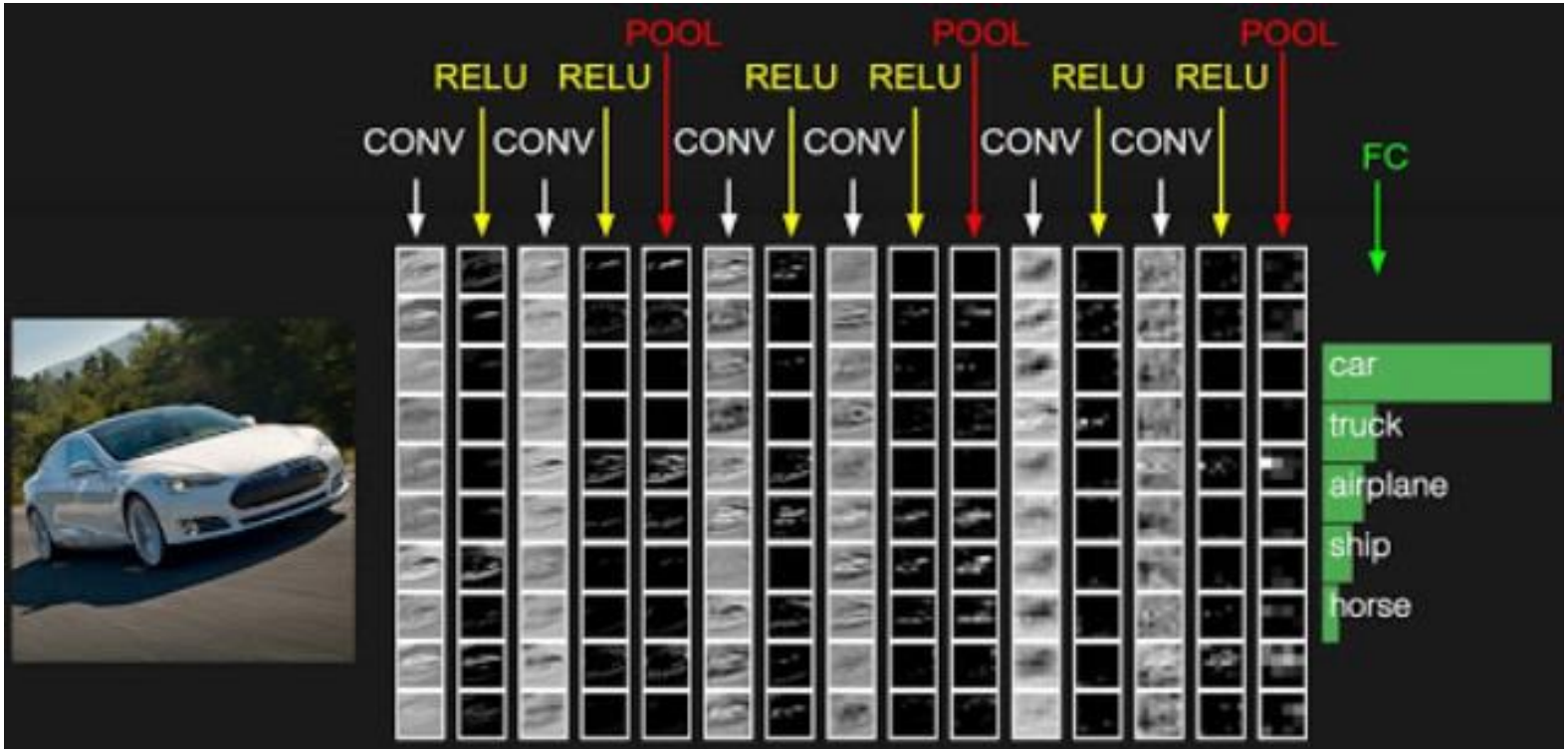


4. Interpretable Convolutional Neural Network

Interpretable Convolutional Neural Network

❖ 핵심 컨셉

- ➢ CNN의 기본구조를 변형하지 않으면서 해석력을 높인 방법론
- ➢ Input의 정보를 최대로 압축해서 담고 있는 마지막 Conv-Layer만 사용



Interpretable Convolutional Neural Network

- 일반적인 CNN filter와 Interpretable filter의 비교
  - 일반적인 filter를 사용한 경우에는 중요 feature를 정확히 찾지 못함
  - Interpretable Filter를 사용한 경우에는 얼굴 중심의 중요 feature를 찾음

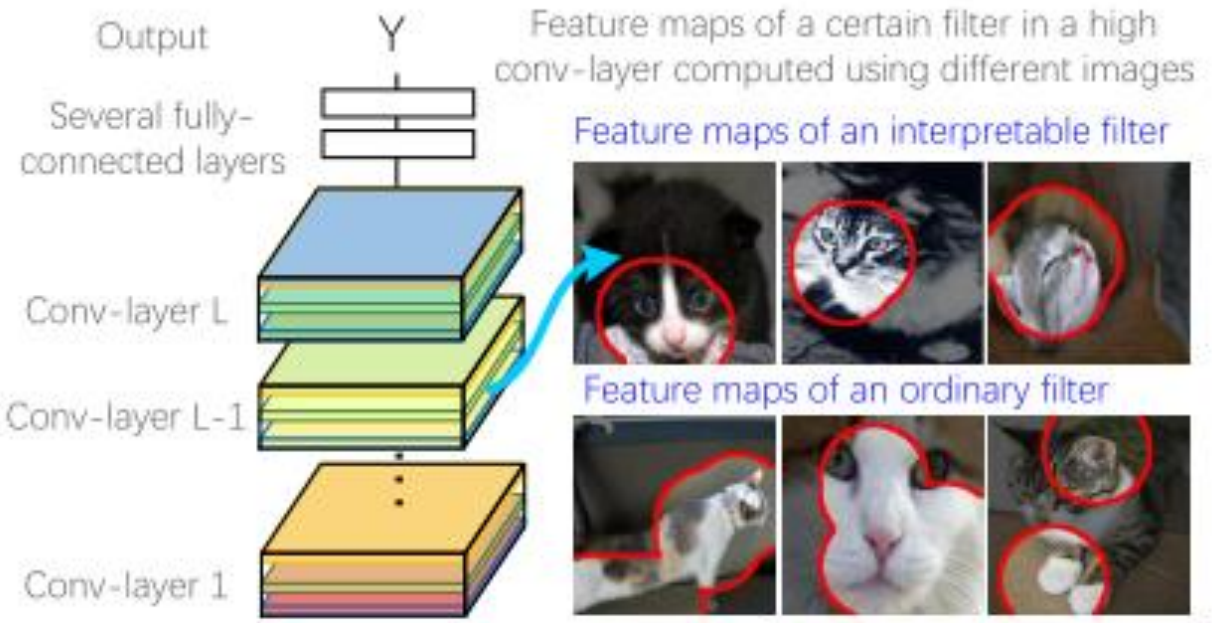
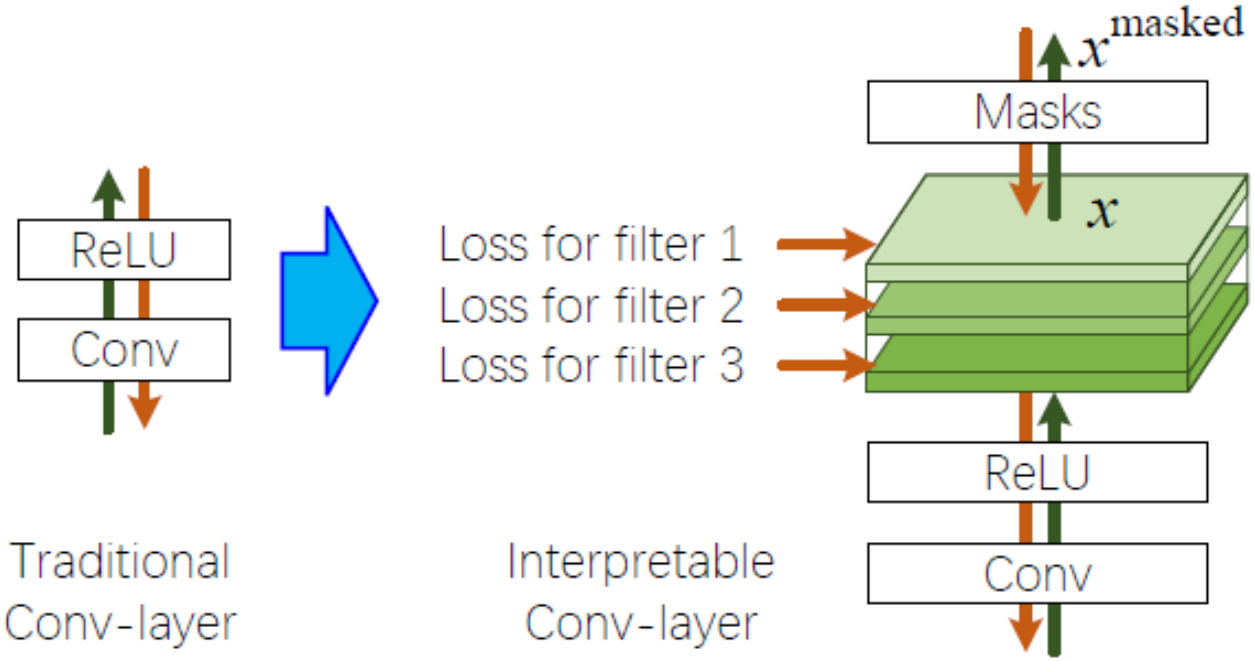


Figure 1. Comparison of a filter's feature maps in an interpretable CNN and those in a traditional CNN.

Interpretable Convolutional Neural Network

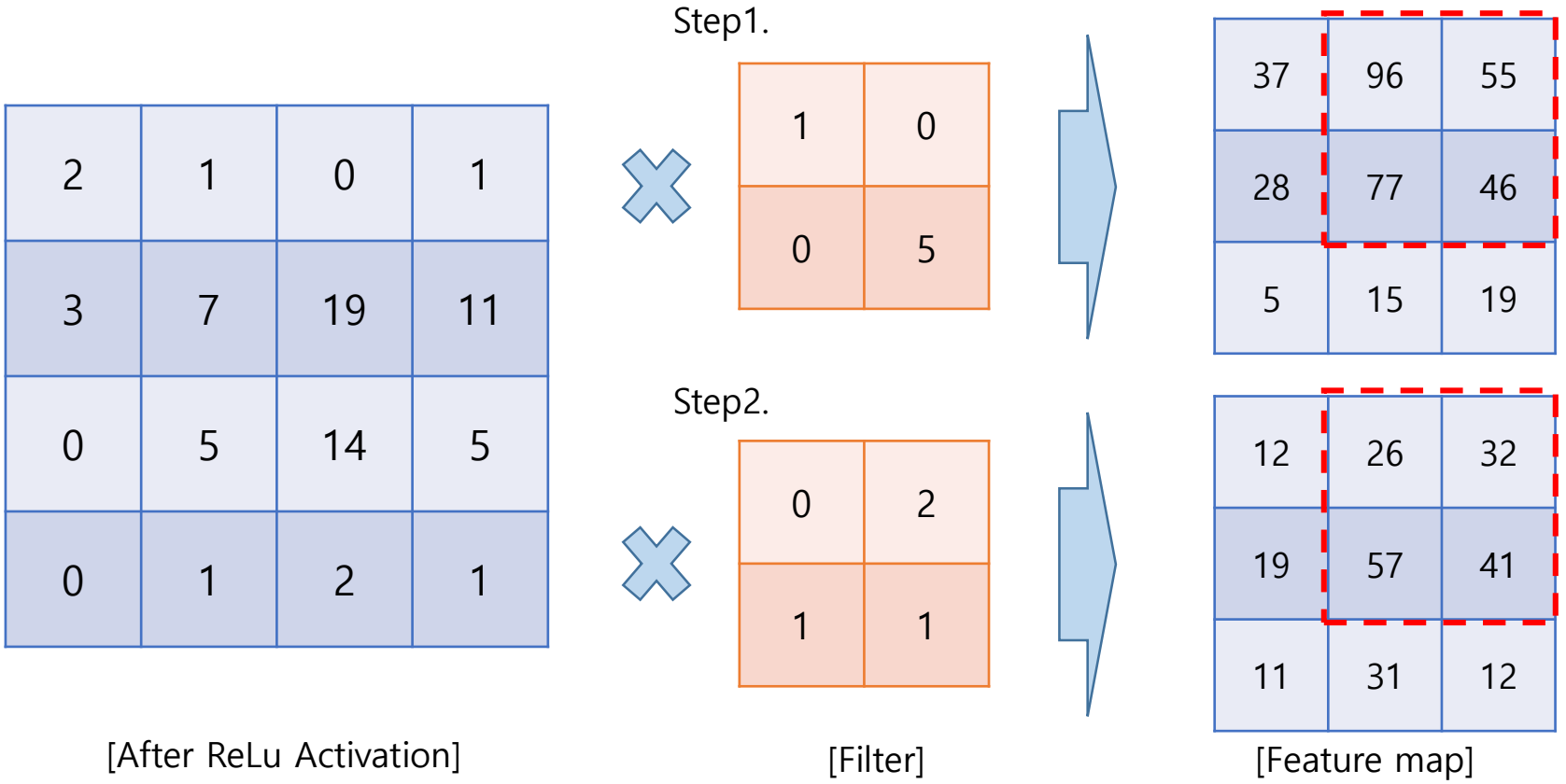
- 전통적인 CNN 개선 방법
  - 마지막 Layer의 Filter별 각각 Loss를 더함
  - 각 Filter는 중요한 부분을 구별할 수 있는 Power를 지님



Interpretable Convolutional Neural Network

❖ Filter 별로 계산하는 이유

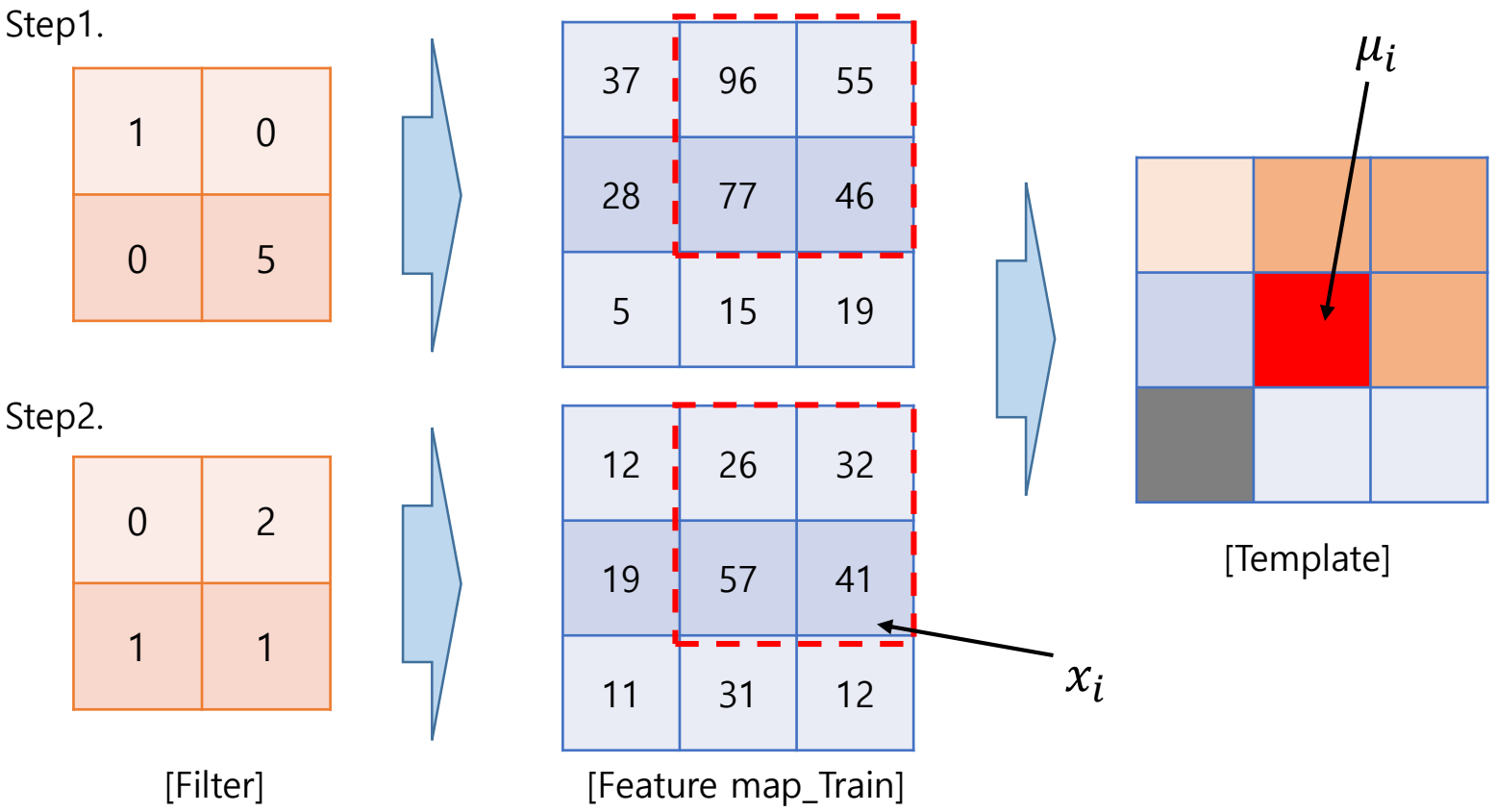
- ➢ 반복된 Filter별 계산값 통해 더 객관적으로 중요 Feature를 선별할 수 있음
- ➢ Filter는 이미지의 특정 한 부분을 통해 활성화 함



Interpretable Convolutional Neural Network

❖ Template 생성

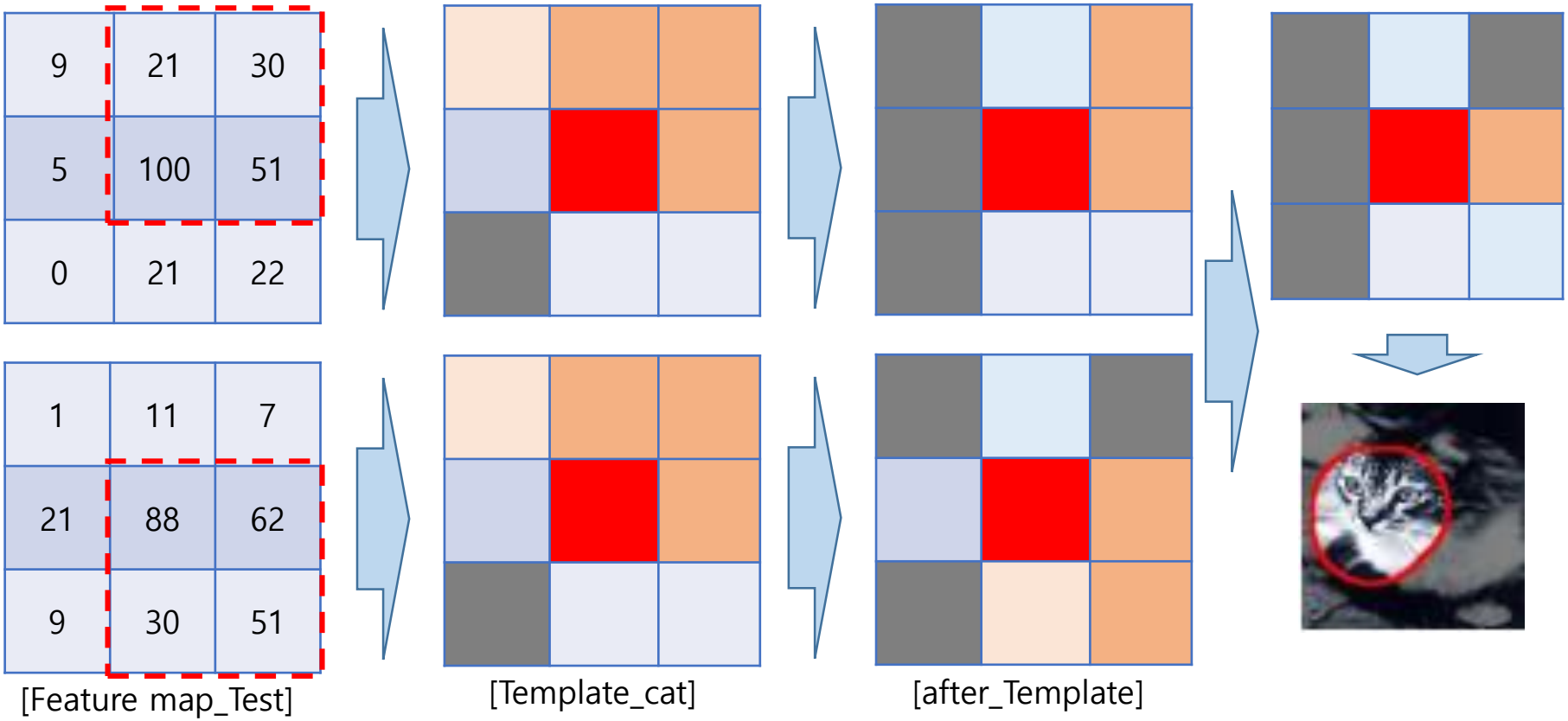
- ➢ 트레이닝 데이터에서 같은 Class(ex 고양이)로 분류되는 데이터로 학습
- ➢ 공통적으로 가장 값이 큰 곳을 μ_i로 지정한 Template을 구성



Interpretable Convolutional Neural Network

❖ Template 생성

- ➢ Test Data의 Top conv-layer의 값을 Template 값과 비교
- ➢ 학습을 진행하면서 비활성화 되는 부분을 중첩하면 중요 Feature 확인 가능



Interpretable Convolutional Neural Network

❖ Loss Function

➢ Filter별 Loss Function Term이 추가 됨

$$\begin{aligned}\mathbf{Loss}_f &= -\,MI(\mathbf{X};\mathbf{T}) \quad \text{for filter } f \\ &= -\sum_T p(T)\sum_x p(x|T)\log\frac{p(x|T)}{p(x)} \qquad (1)\end{aligned}$$

$$\forall T \in \mathbf{T}, \qquad p(x|T) = \frac{1}{Z_T}\exp\left[tr(x\cdot T)\right] \qquad (2)$$

Interpretable Convolutional Neural Network

- Loss Function
  - Filter별 Loss Function Term이 추가 됨

$$\frac{\partial \mathbf{Loss}}{\partial x_{ij}} = \lambda \frac{\partial \mathbf{Loss}_f}{\partial x_{ij}} + \frac{1}{N} \sum_{i=k}^{N} \frac{\partial \mathbf{L}(\hat{y}_k, y_k^*)}{\partial x_{ij}} \qquad (3)$$

λ is a weight

Sum of all local filter loss

Cross-entropy loss

$$\frac{\partial \mathbf{Loss}_f}{\partial x_{ij}} = \frac{1}{Z_T} \sum_T p(T) t_{ij} e^{tr(x \cdot T)} \Big\{ tr(x \cdot T) - \log \left[Z_T p(x) \right] \Big\}$$

$$\approx \frac{p(\hat{T}) \hat{t}_{ij}}{Z_{\hat{T}}} e^{tr(x \cdot \hat{T})} \Big\{ tr(x \cdot \hat{T}) - \log Z_{\hat{T}} - \log p(x) \Big\} \qquad (4)$$

Interpretable Convolutional Neural Network

- Loss Function
  - Filter별 Loss Function Term이 추가 됨

$$\mathbf{Loss}_f = -H(\mathbf{T}) + H(\mathbf{T}' = \{T^-, \mathbf{T}^+\}|\mathbf{X}) + \sum_x p(\mathbf{T}^+, x) H(\mathbf{T}^+|X = x) \quad (5)$$

$$H(\mathbf{T}' = \{T^-, \mathbf{T}^+\}|\mathbf{X}) = -\sum_x p(x) \sum_{T \in \{T^-, \mathbf{T}^+\}} p(T|x) \log p(T|x) \quad (6)$$

Equation 6: Low inter-category entropy

$$H(\mathbf{T}^+|X = x) = \sum_\mu \tilde{p}(T_\mu|x) \log \tilde{p}(T_\mu|x) \quad (7)$$

Equation 7: Low spatial entropy

Interpretable Convolutional Neural Network

- Masking Data
  - 기존 Data에 Mask(=Template)를 씌운 이미지 결과

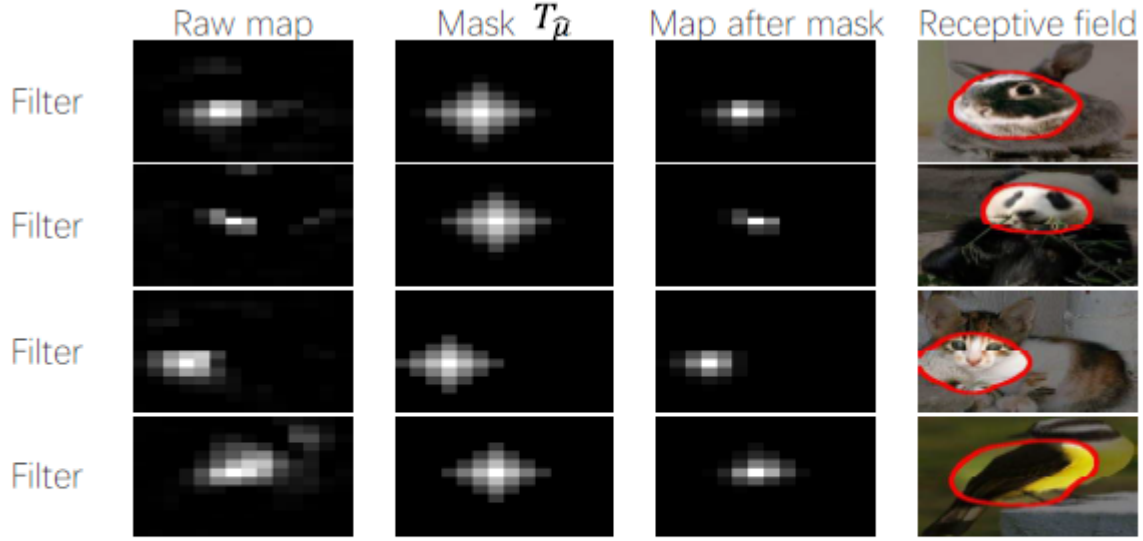
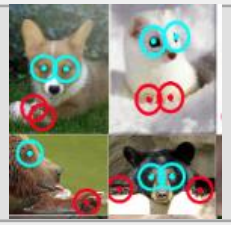


Figure 4. Given an input image I, from the left to the right, we consequently show the feature map of a filter after the ReLU layer x, the assigned mask $T_{\hat{\mu}}$, the masked feature map x^{masked}, and the image-resolution RF of activations in x^{masked} computed by [40].

Interpretable Convolutional Neural Network

❖ 실험 세팅(Data Set)

- ILSVRC 2013 DET Animal-Part dataset
- CUB200-2011 dataset
- Pascal VOC Part dataset

Data	내용	이미지
Animal-Part	Heads and legs of 30 animals	
CUB200-2011	Bird image of 200 pieces	
Pascal VOC Part	107 object landmarks in six animal categories	

Interpretable Convolutional Neural Network

❖ 실험 결과(Animal-Part dataset - instability)

Normalized distance :

$$d_1(p_k, \hat{\mu}) = \frac{\|p_k - p(\hat{\mu})\|}{\sqrt{w^2 + h^2}}$$

Relative location deviation :

$$D_{f,k} = \sqrt{var_{I[d_I(p_k, \hat{\mu})]}}$$

	gold.	bird	frog	turt.	liza.	koala	lobs.	dog	fox	cat	lion	tiger	bear	rabb.	hams.	squi.
AlexNet	0.161	0.167	0.152	0.153	0.175	0.128	0.123	0.144	0.143	0.148	0.137	0.142	0.144	0.148	0.128	0.149
AlexNet, interpretable	**0.084**	**0.095**	**0.090**	**0.107**	**0.097**	**0.079**	**0.077**	**0.093**	**0.087**	**0.095**	**0.084**	**0.090**	**0.095**	**0.095**	**0.077**	**0.095**
VGG-16	0.153	0.156	0.144	0.150	0.170	0.127	0.126	0.143	0.137	0.148	0.139	0.144	0.143	0.146	0.125	0.150
VGG-16, interpretable	**0.076**	**0.099**	**0.086**	**0.115**	**0.113**	**0.070**	**0.084**	**0.077**	**0.069**	**0.086**	**0.067**	**0.097**	**0.081**	**0.079**	**0.066**	**0.065**
VGG-M	0.161	0.166	0.151	0.153	0.176	0.128	0.125	0.145	0.145	0.150	0.140	0.145	0.144	0.150	0.128	0.150
VGG-M, interpretable	**0.088**	**0.088**	**0.089**	**0.108**	**0.099**	**0.080**	**0.074**	**0.090**	**0.082**	**0.103**	**0.079**	**0.089**	**0.101**	**0.097**	**0.082**	**0.095**
VGG-S	0.158	0.166	0.149	0.151	0.173	0.127	0.124	0.143	0.142	0.148	0.138	0.142	0.143	0.148	0.128	0.146
VGG-S, interpretable	**0.087**	**0.101**	**0.093**	**0.107**	**0.096**	**0.084**	**0.078**	**0.091**	**0.082**	**0.101**	**0.082**	**0.089**	**0.097**	**0.091**	**0.076**	**0.098**

	horse	zebra	swine	hippo	catt.	sheep	ante.	camel	otter	arma.	monk.	elep.	red pa.	gia.pa.	**Avg.**
AlexNet	0.152	0.154	0.141	0.141	0.144	0.155	0.147	0.153	0.159	0.160	0.139	0.125	0.140	0.125	0.146
AlexNet, interpretable	**0.098**	**0.084**	**0.091**	**0.089**	**0.097**	**0.101**	**0.085**	**0.102**	**0.104**	**0.095**	**0.090**	**0.085**	**0.084**	**0.073**	**0.091**
VGG-16	0.150	0.153	0.141	0.140	0.140	0.150	0.144	0.149	0.154	0.163	0.136	0.129	0.143	0.125	0.144
VGG-16, interpretable	**0.106**	**0.077**	**0.094**	**0.083**	**0.102**	**0.097**	**0.091**	**0.105**	**0.093**	**0.100**	**0.074**	**0.084**	**0.067**	**0.063**	**0.085**
VGG-M	0.151	0.158	0.140	0.140	0.143	0.155	0.146	0.154	0.160	0.161	0.140	0.126	0.142	0.127	0.147
VGG-M, interpretable	**0.095**	**0.080**	**0.095**	**0.084**	**0.092**	**0.094**	**0.077**	**0.104**	**0.102**	**0.093**	**0.086**	**0.087**	**0.089**	**0.068**	**0.090**
VGG-S	0.149	0.155	0.139	0.140	0.141	0.155	0.143	0.154	0.158	0.157	0.140	0.125	0.139	0.125	0.145
VGG-S, interpretable	**0.096**	**0.080**	**0.092**	**0.088**	**0.094**	**0.101**	**0.077**	**0.102**	**0.105**	**0.094**	**0.090**	**0.086**	**0.078**	**0.072**	**0.090**

Table 3. Location instability of filters ($\mathbf{E}_{f,k}[D_{f,k}]$) in CNNs that are trained for single-category classification using the ILSVRC 2013 DET Animal-Part dataset [36]. Filters in our interpretable CNNs exhibited significantly lower localization instability than ordinary CNNs.

Interpretable Convolutional Neural Network

❖ 실험 결과(Pascal VOC Part Dataset - instability)

Normalized distance :

$$d_1(p_k, \hat{\mu}) = \frac{\|p_k - p(\hat{\mu})\|}{\sqrt{w^2 + h^2}}$$

Relative location deviation :

$$D_{f,k} = \sqrt{var_{I[d_I(p_k, \hat{\mu})]}}$$

	bird	cat	cow	dog	horse	sheep	**Avg.**
AlexNet	0.153	0.131	0.141	0.128	0.145	0.140	0.140
AlexNet, interpretable	**0.090**	**0.089**	**0.090**	**0.088**	**0.087**	**0.088**	**0.088**
VGG-16	0.145	0.133	0.146	0.127	0.143	0.143	0.139
VGG-16, interpretable	**0.101**	**0.098**	**0.105**	**0.074**	**0.097**	**0.100**	**0.096**
VGG-M	0.152	0.132	0.143	0.130	0.145	0.141	0.141
VGG-M, interpretable	**0.086**	**0.094**	**0.090**	**0.087**	**0.084**	**0.084**	**0.088**
VGG-S	0.152	0.131	0.141	0.128	0.144	0.141	0.139
VGG-S, interpretable	**0.089**	**0.092**	**0.092**	**0.087**	**0.086**	**0.088**	**0.089**

Table 4. Location instability of filters ($\mathbf{E}_{f,k}[D_{f,k}]$) in CNNs that are trained for single-category classification using the Pascal VOC Part dataset [3]. Filters in our interpretable CNNs exhibited significantly lower localization instability than ordinary CNNs.

Interpretable Convolutional Neural Network

❖ 실험 결과(CUB200-2011 dataset- instability)

Normalized distance :

$$d_1(p_k, \hat{\mu}) = \frac{\|p_k - p(\hat{\mu})\|}{\sqrt{w^2 + h^2}}$$

Relative location deviation :

$$D_{f,k} = \sqrt{var_{I[d_I(p_k, \hat{\mu})]}}$$

Network	Avg. location instability
AlexNet	0.150
AlexNet, interpretable	**0.070**
VGG-16	0.137
VGG-16, interpretable	**0.076**
VGG-M	0.148
VGG-M, interpretable	**0.065**
VGG-S	0.148
VGG-S, interpretable	**0.073**

Table 5. Location instability of filters ($\mathbf{E}_{f,k}[D_{f,k}]$) in CNNs for single-category classification using the CUB200-2011 dataset.

Interpretable Convolutional Neural Network

❖ 실험 결과(Multi Category - instability)

Normalized distance :

$$d_1(p_k, \hat{\mu}) = \frac{\|p_k - p(\hat{\mu})\|}{\sqrt{w^2 + h^2}}$$

Relative location deviation :

$$D_{f,k} = \sqrt{var_{I[d_I(p_k, \hat{\mu})]}}$$

Dataset	ILSVRC Part [36]	Pascal VOC Part [3]	
Network	Logistic log loss[4]	Logistic log loss[4]	Softmax log loss
VGG-16	–	0.128	0.142
interpretable	–	**0.073**	**0.075**
VGG-M	0.167	0.135	0.137
interpretable	**0.096**	**0.083**	**0.087**
VGG-S	0.131	0.138	0.138
interpretable	**0.083**	**0.078**	**0.082**

Table 6. Location instability of filters ($\mathbf{E}_{f,k}[D_{f,k}]$) in CNNs that are trained for multi-category classification. Filters in our interpretable CNNs exhibited significantly lower localization instability than ordinary CNNs in all comparisons.

Interpretable Convolutional Neural Network

❖ 실험 결과(Multi Category - instability)

Normalized distance :

$$d_1(p_k, \hat{\mu}) = \frac{\|p_k - p(\hat{\mu})\|}{\sqrt{w^2 + h^2}}$$

Relative location deviation :

$$D_{f,k} = \sqrt{var_{I[d_I(p_k, \hat{\mu})]}}$$

	bird	cat	cow	dog	horse	sheep	**Avg.**
AlexNet	0.153	0.131	0.141	0.128	0.145	0.140	0.140
AlexNet, interpretable	**0.090**	**0.089**	**0.090**	**0.088**	**0.087**	**0.088**	**0.088**
VGG-16	0.145	0.133	0.146	0.127	0.143	0.143	0.139
VGG-16, interpretable	**0.101**	**0.098**	**0.105**	**0.074**	**0.097**	**0.100**	**0.096**
VGG-M	0.152	0.132	0.143	0.130	0.145	0.141	0.141
VGG-M, interpretable	**0.086**	**0.094**	**0.090**	**0.087**	**0.084**	**0.084**	**0.088**
VGG-S	0.152	0.131	0.141	0.128	0.144	0.141	0.139
VGG-S, interpretable	**0.089**	**0.092**	**0.092**	**0.087**	**0.086**	**0.088**	**0.089**

Table 4. Location instability of filters ($\mathbf{E}_{f,k}[D_{f,k}]$) in CNNs that are trained for single-category classification using the Pascal VOC Part dataset [3]. Filters in our interpretable CNNs exhibited significantly lower localization instability than ordinary CNNs.

Interpretable Convolutional Neural Network

❖ 실험 결과(Single & Multi Category)

- ➢ Classification Accuracy = Single category is bad, Multi-category is better
- ➢ 명확한 이유에 대해서는 저자도 언급이 없었음

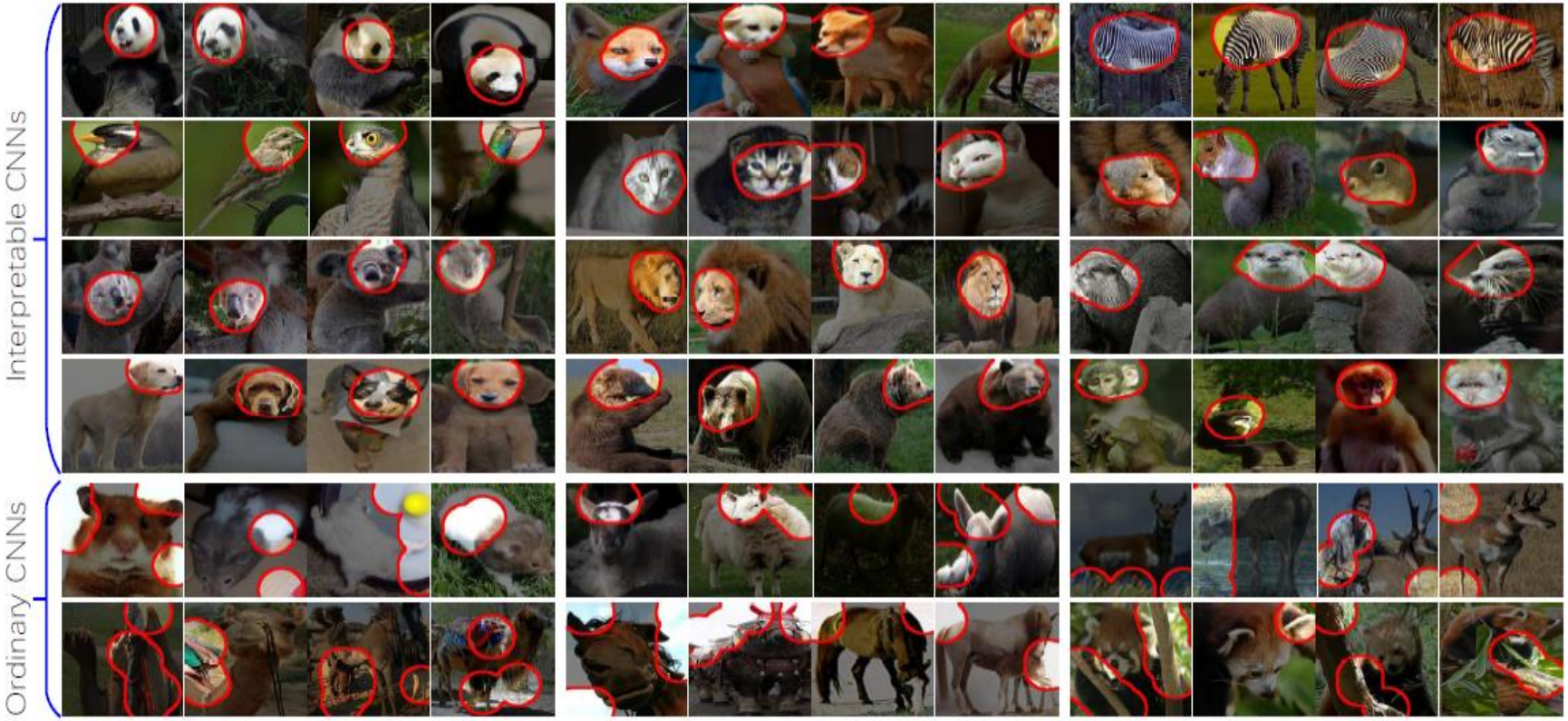
	multi-category			single-category		
	ILSVRC Part	VOC Part		ILSVRC Part	VOC Part	CUB200
	logistic[4]	logistic[4]	softmax			
AlexNet	–	–	–	**96.28**	**95.40**	**95.59**
interpretable	–	–	–	95.38	93.93	95.35
VGG-M	96.73	93.88	81.93	**97.34**	**96.82**	**97.34**
interpretable	**97.99**	**96.19**	**88.03**	95.77	94.17	96.03
VGG-S	96.98	94.05	78.15	**97.62**	**97.74**	**97.24**
interpretable	**98.72**	**96.78**	**86.13**	95.64	95.47	95.82
VGG-16	–	97.97	89.71	**98.58**	**98.66**	**98.91**
interpretable	–	**98.50**	**91.60**	96.67	95.39	96.51

Table 7. Classification accuracy based on different datasets. In single-category classification, ordinary CNNs performed better, while in multi-category classification, interpretable CNNs exhibited superior performance.

Interpretable Convolutional Neural Network

❖ 실험 결과(Image Data)

➢ 일반적인 CNN에 비해 이미지의 특징을 더 잘 추출하고 있음



Interpretable Convolutional Neural Network

❖ 실험 결과(Image Data)

➢ Top conv-layer만 사용 했지만 중요 feature를 잘 선별하고 있었음

Figure 6. Heat maps for distributions of object parts that are encoded in interpretable filters. We use all filters in the top conv-layer to compute the heat map.

5. Conclusion

Conclusion

❖ Conclusion

① 기본 CNN 구조를 크게 변경한 것이 아니기 때문에 다른 구조에도 적용 가능

② 특별한 주석도 필요 없다. Filter 값들이 그것을 표현

③ Loss function의 변경이나 Training Set의 대한 변경도 필요 없음

④ 하지만 Single Category 사례처럼 모델의 성능을 조금 떨어뜨릴 수 있음

❖ Critic

① 사실 기본적인 구조를 변경하지 않았다 것을 제외하고는 큰 의미 찾지 못함

② CAM과 대비했을 때 효율적인지 의문스러움

③ Classification의 성능이 좋아지고, 나빠진 근거가 불충분

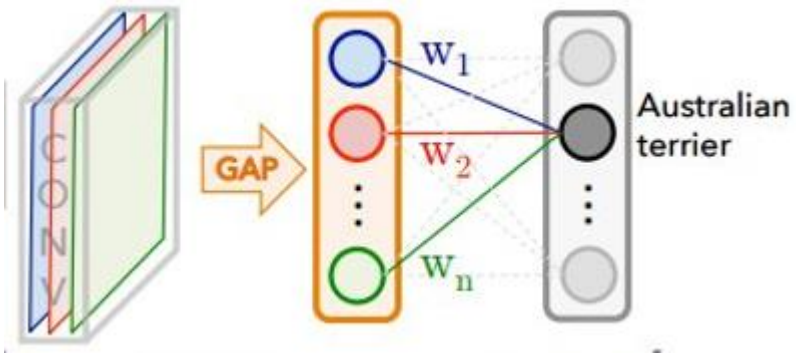
④ 개인적으로는 Cite 횟수에 대해 Cross-Check 해야겠다고 생각함

감사합니다.

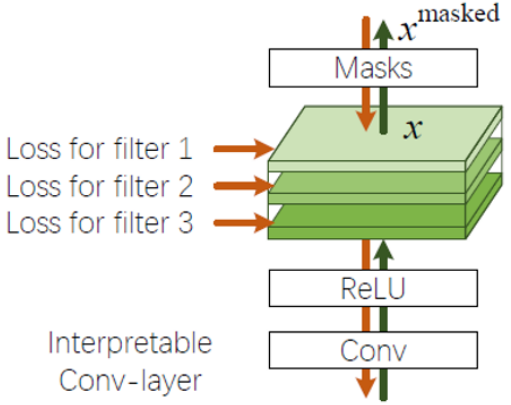
Appendix.

❖ CAM vs Interpretable CNN

- ➢ CAM은 마지막 Gap Layer에 Weight * Channel을 합친 값
- ➢ Interpretable CNN은 기본 CNN의 구조 변경 없이 Filter에 Loss 만 계산



[CAM Last Layer]



[Interpretable CNN Last Layer]

❖ 출처

- Figure: Basic architecture of Convolutional Neural Networks(https://pydeeplearning.weebly.com/blog/basics-of-convolutional-neural-networks)
- https://bcho.tistory.com/1149(조대협의 블로그)
- https://blog.ees.guru/50 (EeS의 연구실)
- Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu(2018). Interpretable Convolutional Neural Networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8827-8836)
- http://taewan.kim/post/cnn/
- https://medium.com/@kamwohng/interpretable-convolutional-neural-network-3f7ef6c9b7ae