Learning Deep Features for Discriminative Localization

김미주

XAI(eXplainable AI)

: 사람이 AI의 동작과 최종결과를 이해하고 올바르게 해석할 수 있고, 결과물이 생성되는 과정을 설명 가능하도록 해주는 기술

How to Visualize Neural Networks?

1. *Backpropagation Based Method (BBMs)*

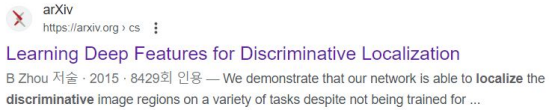
   : 입력값에 대해 `backpropagation`을 통해 각 픽셀의 기여도를 나타내는 방법

   - LRP(Layer-wise Relevance Propagation)

2. *Activation Based Methods (ABMs)*

   : 각 Convolutional layers 에서 나온 Activation들의 선형 결합한 가중치를 사용하여 설명 생성

   - **CAM**(Class Activation Map)



3. *Perturbation Based Methods (PBMs)*

   : input 에 작은 변화(purtubation)를 줘서 예측값이 어떻게 변하는지를 통해 중요도를 파악

   - LIME(Local Interpretable Model-Agnostic Explanation)

INTRODUCTION

- 최근 연구
  - 최근 연구는 object 의 location 에 대해 bounding box 와 같은 supervision 이 제공되지 않았음에도 불구하고 convolutional neural network (CNNs) 다양한 층의 convolutional units이 실제로 object detector 로서 행동한다는 것을 보여줌.

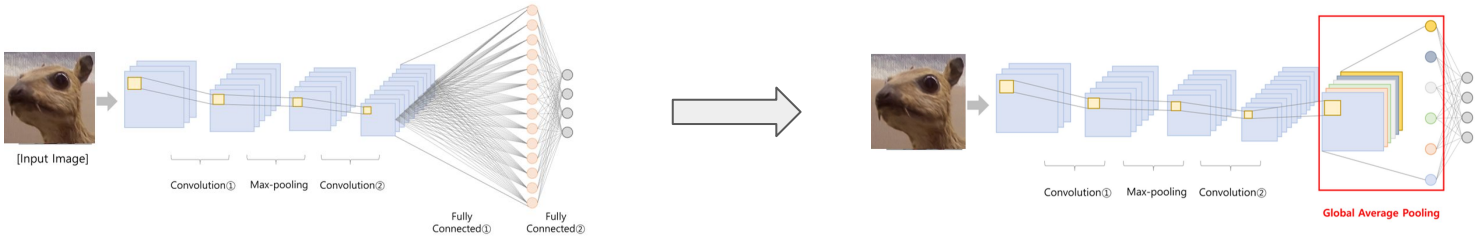
⇒ 해당 네트워크가 분류(Classification)에 대한 목표를 가지고 학습하더라도 이미지 영역을 위치화할 수 있음을 보여준다.

- Fully-Connected Layer
  - 기존의 CNN based image classification model 은 마지막 feature map 을 flatten해서 FC(Fully Connected) layer로 변환한 후 classification을 학습,

⇒ 그런데 flatten 을 하게 되면 기존의 feature map이 가지고 있던 공간 정보를 잃게 된다.

- Global Average Pooling
  - global average pooling의 이점이 단순히 regularizer의 역할을 수행하는 것 이상으로 확장될 수 있다는 사실 발견

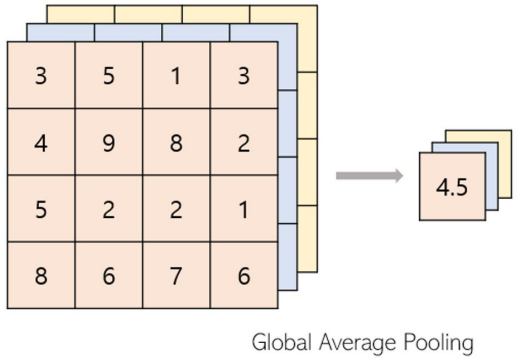
⇒ network의 구조를 약간 수정하면, network 의 localization ability 를 마지막 layer까지 보유할 수 있다는 것



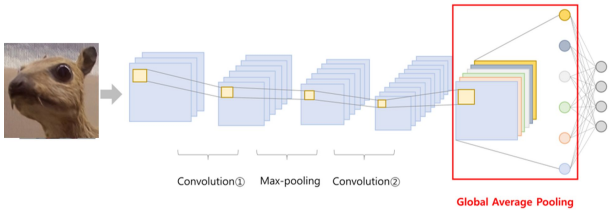
Global Average Pooling(GAP)

- pooling 의 역할
  - CNN 은 많은 convolution layer 를 쌓기 때문에 필터(parameter)의 수가 많다. ⇒ 많은 feature map 들이 쌓임.
  - 높은 차원을 다루기 위해 많은 수의 파라미터가 필요한데, 파라미터가 많아지면 overfitting 이 발생할 수 있음.
    ⇒ 필터에 사용되는 파라미터 수를 줄이기 위해 pooling layer 필요
  - 결국 feature map을 다운샘플링하여 특성 맵의 크기를 줄이는 풀링 연산이 이루어진다.

- Global Average Pooling(GAP)
  - 모든 값을 고려하여 평균 값을 사용
  - 본 논문에서는 CNN의 마지막 FC layer를 Global Average Pooling 으로 대체하여 overfitting을 방지할 수 있는 regularization의 역할을 하며, 위치정보를 손실하지 않을 수 있도록 함.



Global Average Pooling



INTRODUCTION

- 실험 결과
  - object categorization에 대해 학습된 CNN은 action classification을 위한 discriminative regions을 사람보다는 사람이 상호작용하고 있는 개체의 위치로 잘 찾고 있는 것을 볼 수 있음.
  - global average pooling layer 를 통해 classification-trained CNN 은 image를 classification 할 수 있을뿐만 아니라 class-specific image regions 를 single forward-pass 로 localization 가능

Class Activation Mapping

: CNNs에 있는 global average pooling을 사용해 class activation maps (CAM)를 만들어내는 절차 설명

CAM architecture

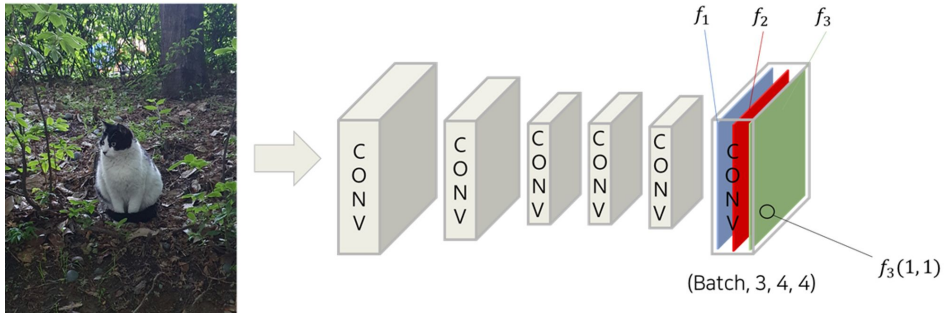
(Convolutional feature map → global average pooling (GAP) → softmax layer)

- global average pooling GAP : last convolutional layer 각 unit의 feature map 에 대해 global average pooling을 수행하여 spatial average 출력
- weighted sum(GAP output * 각 class weight) = softmax input: final output 을 생성하는데 사용

- CAM: last convolutional layer의 feature map 에 fine-tuning 된 각 class weight 를 곱해주면 channel 개수만큼 heatmap 생성 → heatmap 을 모두 pixel-wise sum

Class Activation Mapping

1. $f_k(x,y)$: spatial location (x, y) 에서 last convolutional layer의 feature map unit k 의 activation



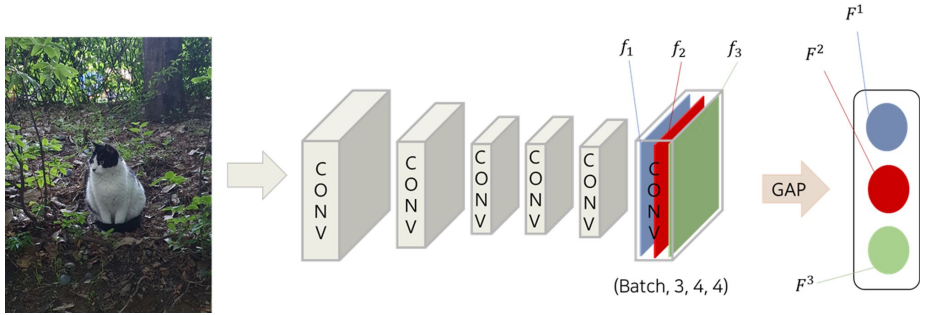
- last convolutional layer의 feature map크기를 (Batch, 3, 4, 4)라고 가정 (channel: 3, 가로, 세로: 4)

  → 여기서 channel이 3 이므로, k는 3까지 존재

  ex. 3번째 channel의 (1, 1)의 activation: $f_3(1,1)$

Class Activation Mapping

2. $F_k = \sum_{x,y} f_k(x,y)$: unit k 에 대해 global average pooling 을 수행한 결과



- global average pooling을 하게 되면, f_k에서 **4x4** 였던 데이터를 평균 내서 하나의 값으로 생성

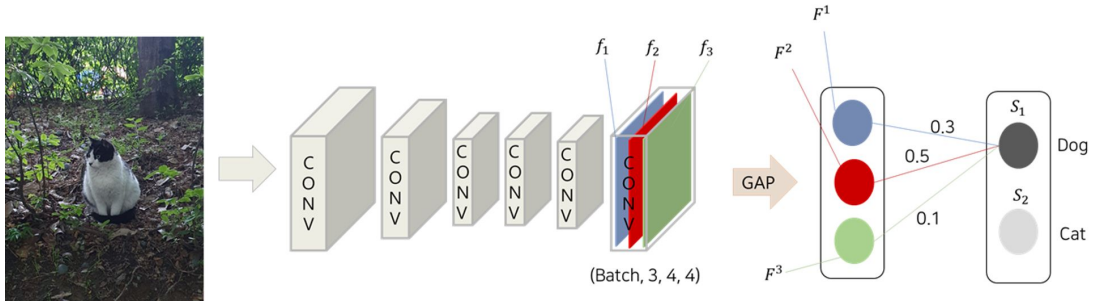
  → 각 데이터들은 F_1, F_2, F_3가 된다.

  ex. $\mathbf{F_1}$: $\{f_1(1,1)+f_1(1,2)+...+f_1(4,4)\} \div 16$ (1번째 channel의 global average pooling output)

Class Activation Mapping

3. $S_c=\sum_k w_k^c F_k$: 주어진 class c 에 대한 softmax input [class 1(Dog) 에 대한 softmax input]

GAP이 끝나면 뒤에는 각 클래스로 연결되는 Fully-connected layer를 붙여 fine-tuning 수행



- 하나의 이미지를 개와 고양이를 분류하는 이진 분류 문제로 가정: class c = 2

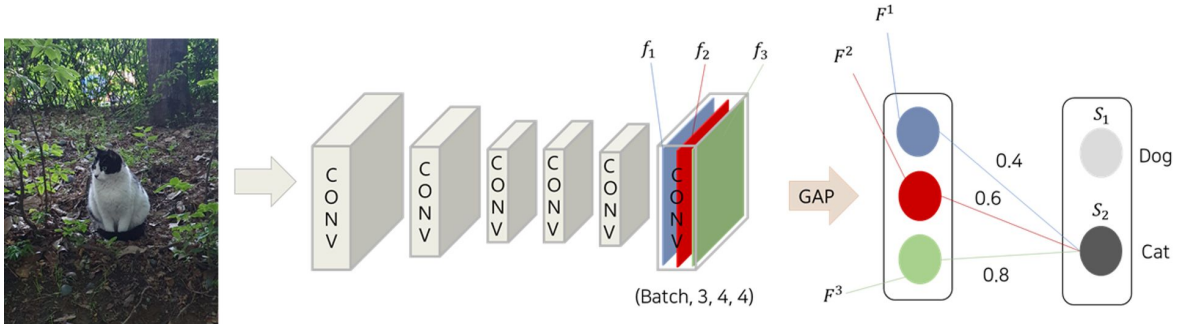
  → Global Average Pooling을 통해서 얻은 F^k를 weight와 곱해서 최종적으로 softmax의 input인 S_c 를 얻음.

  $S_1 = F_1 \times 0.3(w^1_1) + F_2 \times 0.5(w^1_2) + F_3 \times 0.1(w^1_3)$

Class Activation Mapping

3. $S_c = \sum_k w_k^c F_k$: 주어진 class c 에 대한 softmax input [class 2(Cat) 에 대한 softmax input]

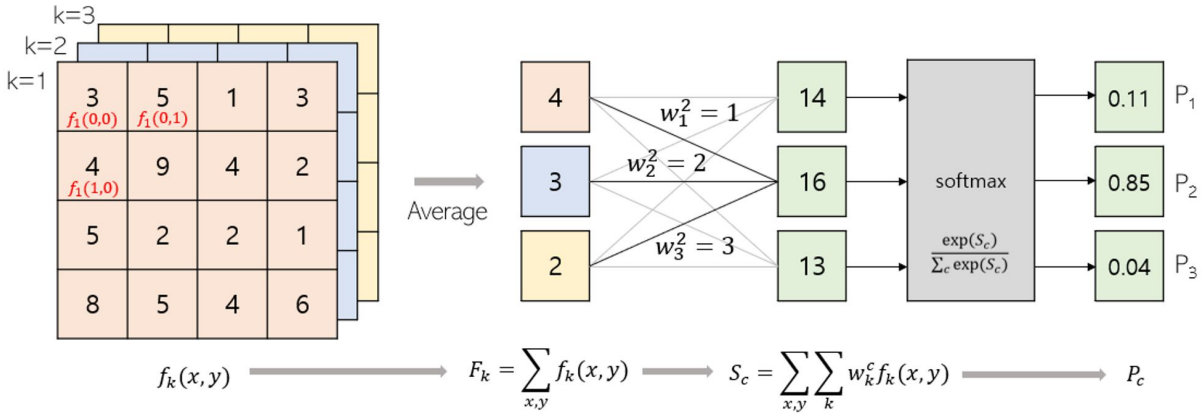
   GAP이 끝나면 뒤에는 각 클래스로 연결되는 Fully-connected layer를 붙여 fine-tuning 수행



$S_2 = F^1 \times 0.4(w^2_1) + F^2 \times 0.6(w^2_2) + F^3 \times 0.8(w^2_3)$

4. $P_c = \exp(S_c) \div \sum_c \exp(S_c)$: class c 에 대한 softmax output

Class Activation Mapping



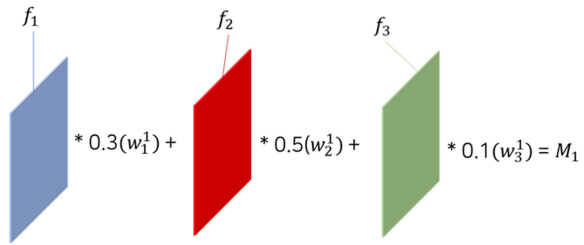
- $M_c(x,y)=\sum_k w_k^c f_k(x,y)$: class c 에 대한 class activation map

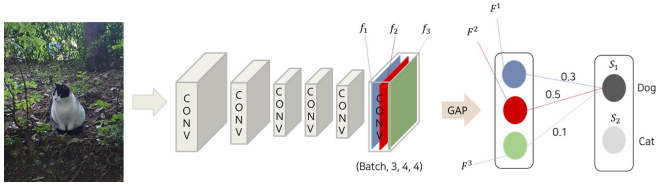
  s_c 의 수식을 변형하여 각 spatial element 나타내는 식은 아래와 같다.

  1. feature map $f_k(i,j)$에 각 class에 대한 가중치 w^c_k 를 곱해주면 K 개의 heatmap 생성
  2. K 개의 heatmap 이미지를 모두 pixel-wise sum = CAM

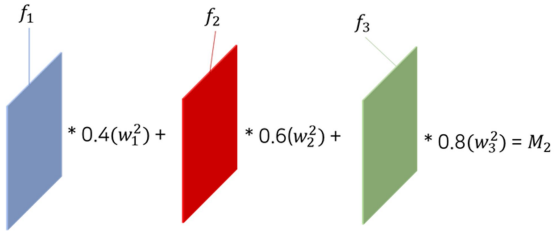
Class Activation Mapping

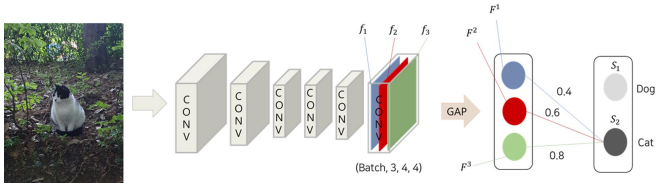
- Dog class activation map





- Cat class activation map





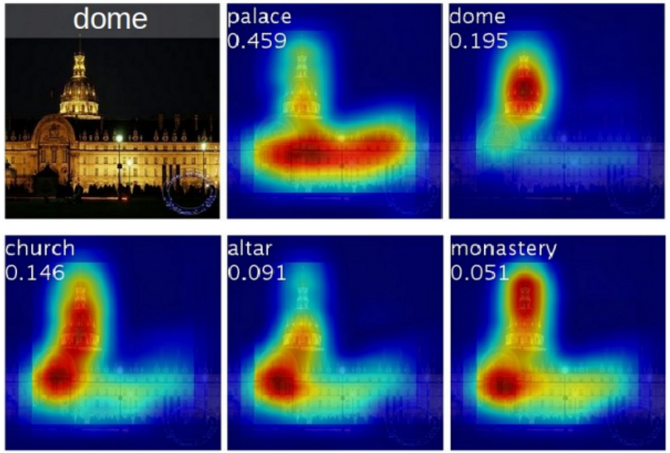
- M_c : class c 에 대한 class activation map

1. feature map $f_k(i,j)$에 각 class에 대한 가중치 w^c_k 를 곱해주면 channel K 개의 heatmap 생성
2. K 개의 heatmap 이미지를 모두 pixel-wise sum = CAM

Class activation map 을 input image 의 사이즈로 upsampling 해줌으로써, 특정한 category와 가장 관련이 있는 image regions을 식별할 수 있게 된다.

Class Activation Mapping

- 주어진 이미지의 top 5 predicted categories 에 대해 생성된 CAMs 결과



- predicted class and its score: class activation map 위에 표시
- 실험 결과
  - highlighted regions 이 predicted class 에 따라 다르다는 것을 관찰
  - ex. dome: 뒤쪽 원형 부분 활성화, palace: 아래쪽 평평한 부분 활성화
- 하나의 이미지에 대해서 map 생성 시 서로 다른 classes c를 사용했을 때 CAM에서의 차이를 보여준다.
  → 주어진 이미지에 대해서도 다른 category에 대한 discriminative region이 다르다는 것을 확인할 수 있다.

Weakly-supervised Object Localization

: ILSVRC 2014 benchmark dataset 에 대해 CAM 의 localization ability 평가

weakly supervised learning

: 학습할 이미지에 대한 정보보다 예측해야할 정보가 더 디테일한 경우

⇒ CAM: 이미지에 대한 label만 갖고 CNN 모델을 학습했지만, 결과적으로 이미지의 어느 부분을 주로 보고 label을 예측했는가?

Setup

- model
  - 해당 실험에서는, AlexNet, VGGnet, GoogLeNet을 사용하여 실험을 진행

    해당 network는 원래 fully-connected layer가 있지만, 이를 제거하고 GAP + softmax layer로 대체하여 실험 진행

⇒ AlexNet-GAP, VGGnet-GAP and GoogLeNet-GAP 생성

Classification

⇒ object classification 에 대한 결과를 보여주며 CAM이 classification performance 에 영향을 미치지 않는다는 것을 입증

- Classification: original vc GAP networks

Table 1. Classification error on the ILSVRC validation set.

Networks	top-1 val. error	top-5 val. error
VGGnet-GAP	33.4	12.2
GoogLeNet-GAP	35.0	13.2
AlexNet*-GAP	44.9	20.9
AlexNet-GAP	51.1	26.3
GoogLeNet	31.9	11.3
VGGnet	31.2	11.4
AlexNet	42.6	19.5
NIN	41.9	19.6
GoogLeNet-GMP	35.6	13.9

❑ object category 와 bounding box location 을 정확히 식별해야 하므로 localization 에서 high performance 를 달성하기 위해 classification이 잘 수행이 되어야한다.

❑ GoogLeNet-GAP 과 GoogLeNet-GMP 이 유사한 성능을 보임.

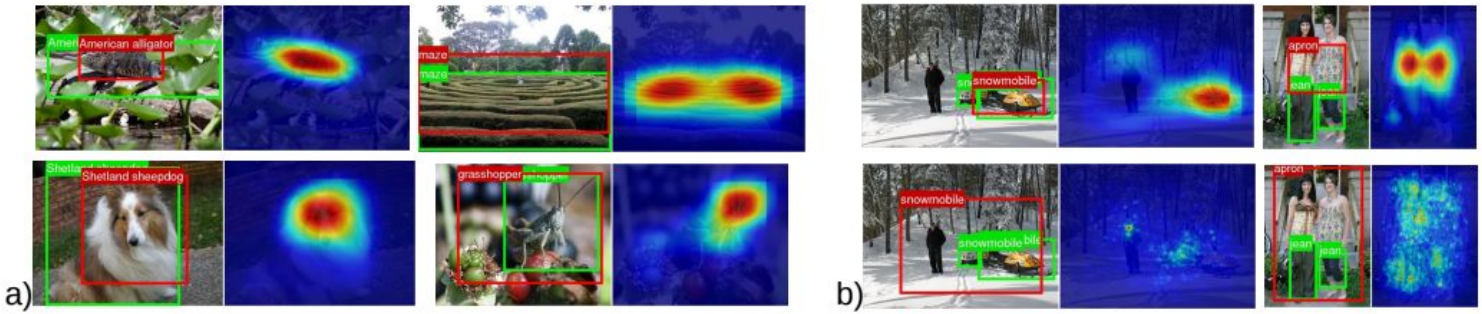
→ classification performance 가 GAP networks 에 대해 보존된다는 것을 알 수 있음.

Localization

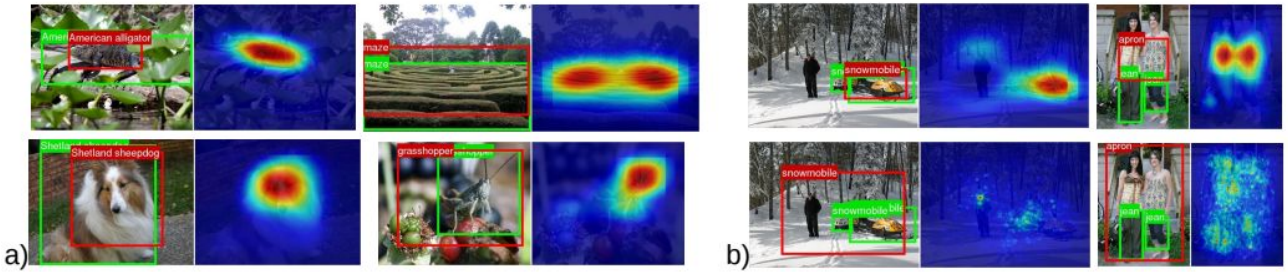
⇒ weakly-supervised object localization 에 효과적이라는 것을 입증

- Localization 수행을 위해 bounding box 와 관련 object category를 생성 필요
- object category 생성 방법
  - CAM 에서 bounding box 를 만들기 위해 thresholding 을 통해 heatmap 을 segmentation

  → CAM의 max value의 20%보다 높은 값들을 가지는 구역만 잘라낸 뒤에, 이 segmentation map을 포함할 수 있는 가장 큰 bounding box 생성



Localization



a) Examples of localization from GoogleNet-GAP.
b) Comparison of the localization from GooleNet-GAP (upper two) and the backpropagation using AlexNet (lower two).
ground-truth boxes, predicted bounding boxes from the class activation map

Table 2. Localization error on the ILSVRC validation set. *Backprop* refers to using [22] for localization instead of CAM.

Method	top-1 val.error	top-5 val. error
GoogLeNet-GAP	**56.40**	**43.00**
VGGnet-GAP	57.20	45.14
GoogLeNet	60.09	49.34
AlexNet*-GAP	63.75	49.53
AlexNet-GAP	67.19	52.16
NIN	65.47	54.19
Backprop on GoogLeNet	61.31	50.55
Backprop on VGGnet	61.12	51.46
Backprop on AlexNet	65.17	52.64
GoogLeNet-GMP	57.78	45.26

- GoogLeNet-GAP
  - top-5 에서 43% 의 가장 낮은 localization error 를 달성함으로써 baseline approaches 를 능가

Localization

CAM vs weakly-supervised vs fully-supervised CNN methods

Table 3. Localization error on the ILSVRC test set for various weakly- and fully- supervised methods.

Method	supervision	top-5 test error
GoogLeNet-GAP (heuristics)	weakly	**37.1**
GoogLeNet-GAP	weakly	42.9
Backprop [22]	weakly	46.4
GoogLeNet [24]	full	26.7
OverFeat [21]	full	29.9
AlexNet [24]	full	34.2

- CAM을 weakly-supervised, fully-supervised CNN methods 와도 비교하기 위해 GoogLeNet-GAP 을 ILSVRC test set 에 대해 실험
- heuristic : bounding box selection strategy 를 다르게 → 성능 향상
- 결과: 인상적인 수치이긴하나, weakly-supervised GoogLeNet-GAP 을 fully-supervised GoogLeNet 과 비교했을 때 더 성능을 높여야 한다.

Conclusion

- 마지막 FC layer를 **Global Average Pooling** 으로 대체한 CNN을 통해 Class Activation Mapping (CAM) 제시

  ⇒ bounding box annotation 을 사용하지 않고도 object localization 가능

- CNN에 의해 감지된 discriminative object parts를 강조하여 predicted class score를 시각화할 수 있다.
- weakly supervised object localization 관점에서 global average pooling CNNs 이 object localization을 수행할 수 있음을 보여줌.

참고자료

How to Explain AI (AI를 설명하는 방법) (datanetworkanalysis.github.io)

Global Average Pooling 이란 - gaussian37

[포테이토 논문 리뷰] Learning Deep Features for Discriminative Localization (tistory.com)

8. Learning Deep Features for Discriminative Localization(CAM) - paper review :: 헤헤 (tistory.com)