

CAM & Grad-CAM

Preliminaries : CAM

- Method: CNN의 마지막 FC Layer로 되기 전에, 각 Feature map에 대한 Global average pooling을 통해 어디에 가장치를 두고 Classification 했는지 확인할 수 있음.
GAP을 지나면, 각 클래스에 대한 FC Layer를 연결한다.
각 Feature map을 통해 heatmap을 생성하고, 이 heatmap에 GAP을 통해 생성된 각 Class에 대한 Weight를 곱한다
→ 모든 heatmap을 pixel-wise sum하여 CAM Score 생성.

$$L_{CAM}^c(i, j) = \sum_k w_k^c f_k(i, j)$$

CNN의 마지막 Layer에 붙이는 이유는??

→ CNN은 초반 레이어에서 local한 정보를, 후반 레이어에서 보다 global한 정보를 갖고 있는 경우가 많기 때문에 '전체에서 일부 영역을 찾기 위해서'는 가장 마지막 레이어에 붙이는 게 낫다.

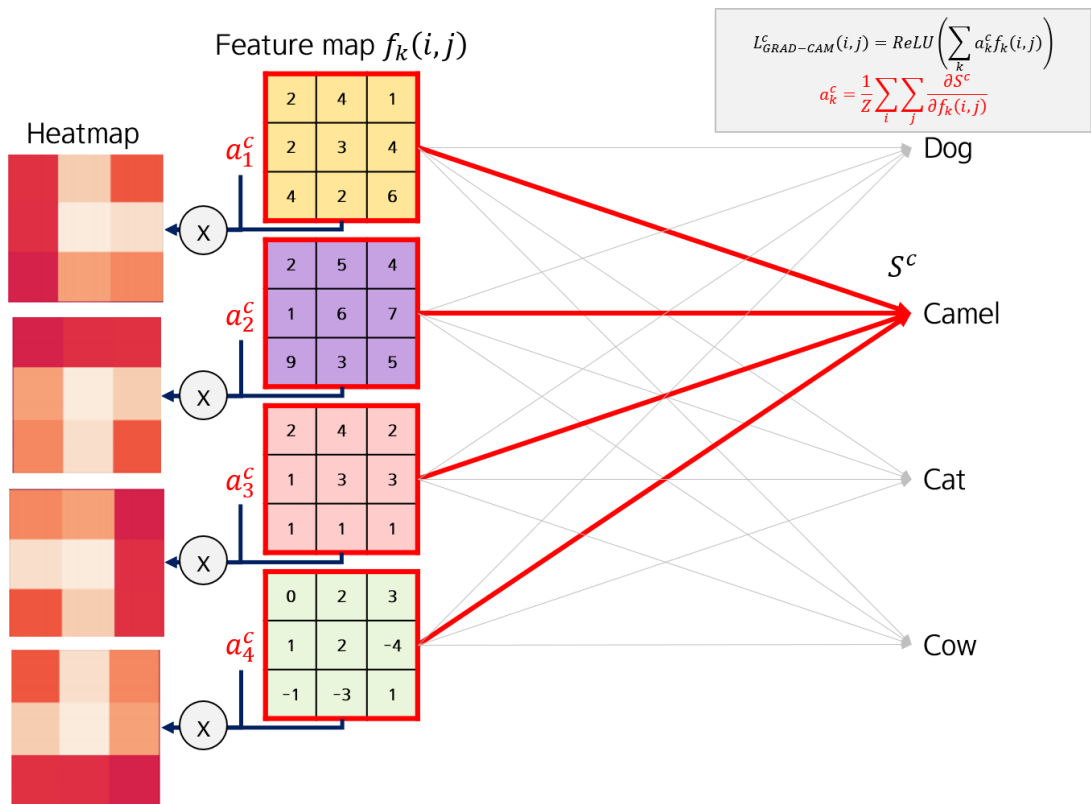
- Limitations:
 1. GAP가 반드시 필요하다
 2. Convolutional layer에 대해서만 적용 가능하다.

Introduction

CAM의 한계점을 보완 → 네트워크 구조에 국한되지 않으며, GAP Layer를 사용하지 않아도 된다.

GAP을 통해 계산한 Feature map 별 Weight를 Gradient로 대체해서 사용

Method



Camel 클래스에 대해 연결된 모든 feature map을 matmul한 S_c 에 대해서 각 feature map의 i, j 번째 픽셀의 Gradient를 평균 내어 계산한 값을 각 클래스 당, feature map만큼 가지게 된다.

k 번째 feature map f 에 대해서 Gradient를 통해 계산된 weight인 k 에 대한 a 를 곱한 값을 ReLU하여 만듦

$$L_{Grad-CAM}^c(i, j) = ReLU\left(\sum_k a_k^c f_k(i, j)\right)$$

$$a_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial S^c}{\partial f_k(i, j)}$$

→ a 를 구하면, 이를 feature map에 곱해서 heatmap을 만들고 이걸 pixel-wise sum한 다음, ReLU를 거쳐 최종 Grad-CAM Score 생성.

Results

