


XAI_02. CAM, Grad CAM

🕒 생성일	@2022년 8월 19일 오후 4:37
📁 유형	머신러닝/딥러닝
👤 작성자	 동훈 오

CNN visualization: CAM and Grad-CAM 설명

이번 포스팅에서는 CNN 모델이 어느 곳을 보고 있는지 를 알려주는 weak supervised learning 알고리즘 (CAM, Grad-CAM)에 대해 정리해보고자 합니다. 학습한 네트워크가 이미지를 개라고 판별할 때와 고양이라고 판별할 때, 각각 이미지에서 중요하게 생각하는 영역은 다를 것입니다. 이를 시각화해주는 알고리즘이 바로 Class Activation Map (CAM) 관련

🔗 <https://tyami.github.io/deep%20learning/CNN-visualization-Grad-CAM/>

Taeyang Yang

Study note

Data science
Machine learning
Deep learning

↳ 위의 깃헙 블로그 내용 정리

Weakly Supervised Learning 과 CAM 의 관계

학습할 이미지에 대한 정보보다 예측해야할 정보가 더 디테일한 경우, 이를 Weakly Supervsied learning 이라고 한다. 예를 들어, 훈련데이터로 이미지와 라벨 데이터만 있고 테스트로는 bbox 를 예측하고자 할 때.

모델이 이미지의 어느 부분을 주로 참고해서 분류를 하였는지 알고자 할 때, CAM 을 활용할 수 있다. 이런 점에서 CAM 은 Weakly Supervised learning 의 하위 전략이라고 고려할 수 있다.

CAM : Class Activation Map

CAM 은 2016년 'Learning Deep Features for Discriminatvie Localization' 논문에서 제시되었다.



Figure 2. Class Activation Mapping: the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps (CAMs). The CAM highlights the class-specific discriminative regions.

위 그림은 CAM 에 대한 직관적인 설명이다. Global Average Pooling ; GAP 를 통해 얻은 가중치를 CNN의 마지막 각 레이어에 곱함으로써 특정 클래스에 대해 모델이 어느 부분에 주목하는가를 시각적으로 확인할 수 있다.

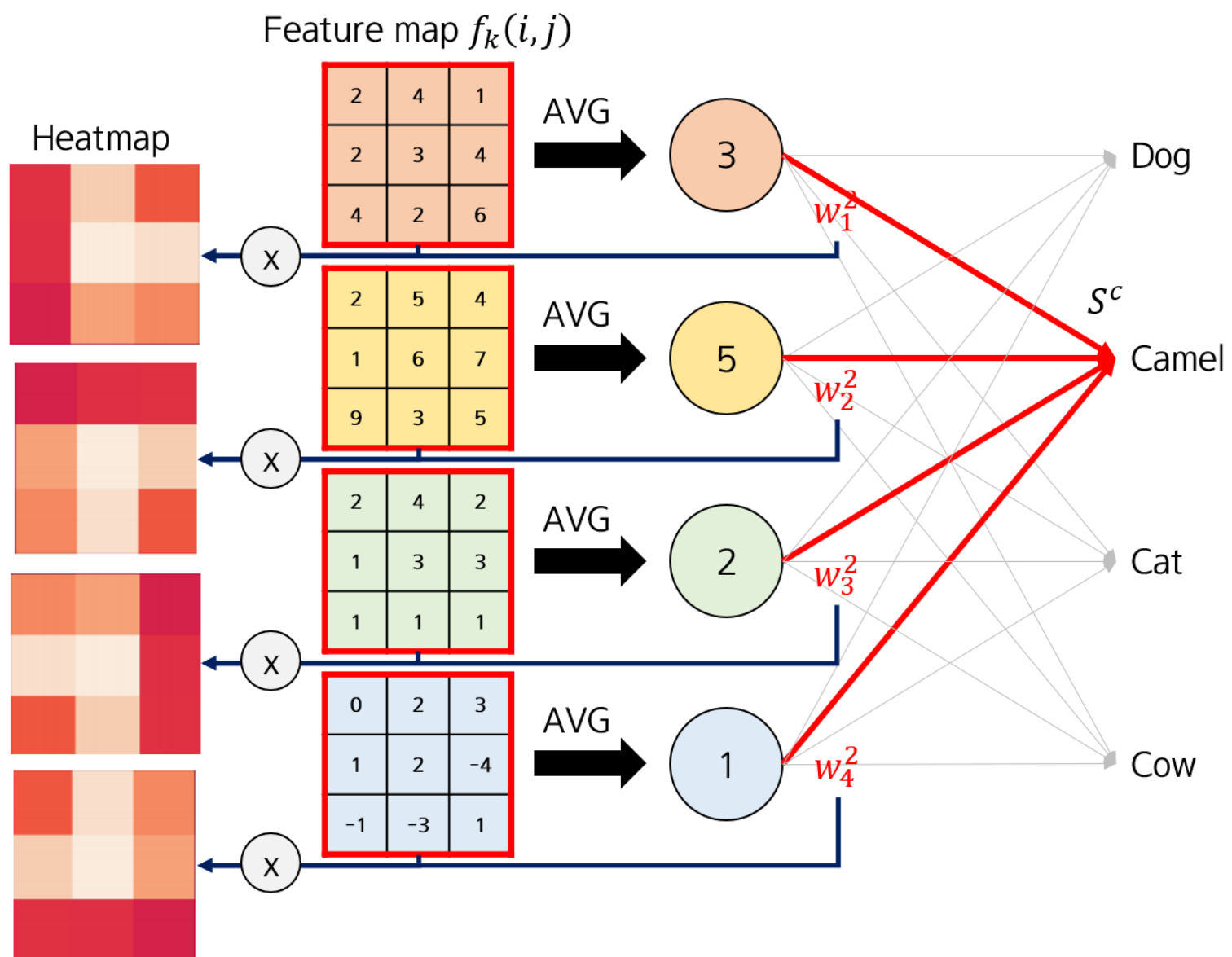
CAM 이 연산되어 시각화되기 까지의 과정은 다음의 수식과 과정을 따른다.

- 클래스 C 에 대한 CAM 이미지는 아래의 수식을 따른다.

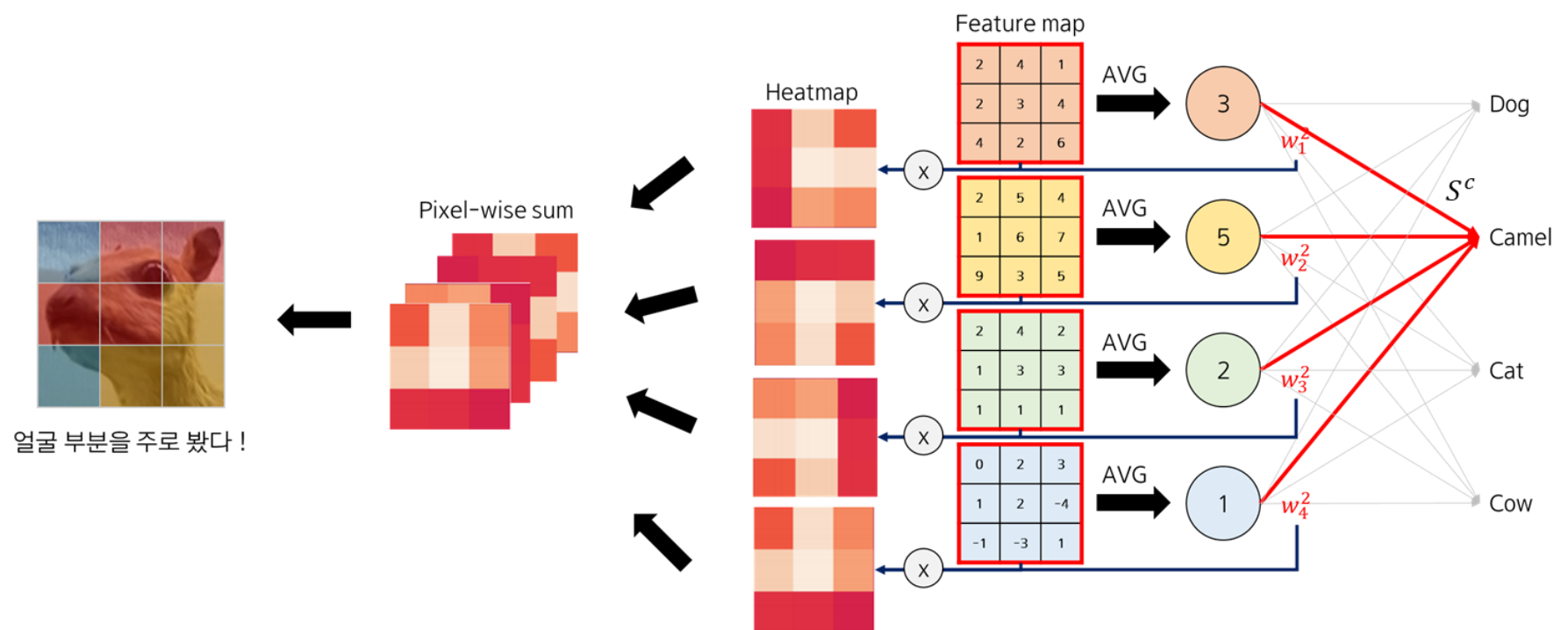
$$L_{CAM}^c(i, j) = \sum_k w_k^c f_k(i, j)$$

- $f_k(i, j)$: k 번째 feature image (i, j 는 x, y 축 좌표를 의미함)
- w_k^c : k 번째 feature image $f_k(i, j)$ 에서 class c 로 가는 weight

- cnn layer 의 feature map $f_k(i, j)$ 에 풀링의 출력으로 나온, 특정 클래스 c에 대한 가중치를 각각의 대응으로 곱하면, heatmap 이 아래 이미지와 같이 만들어 진다.



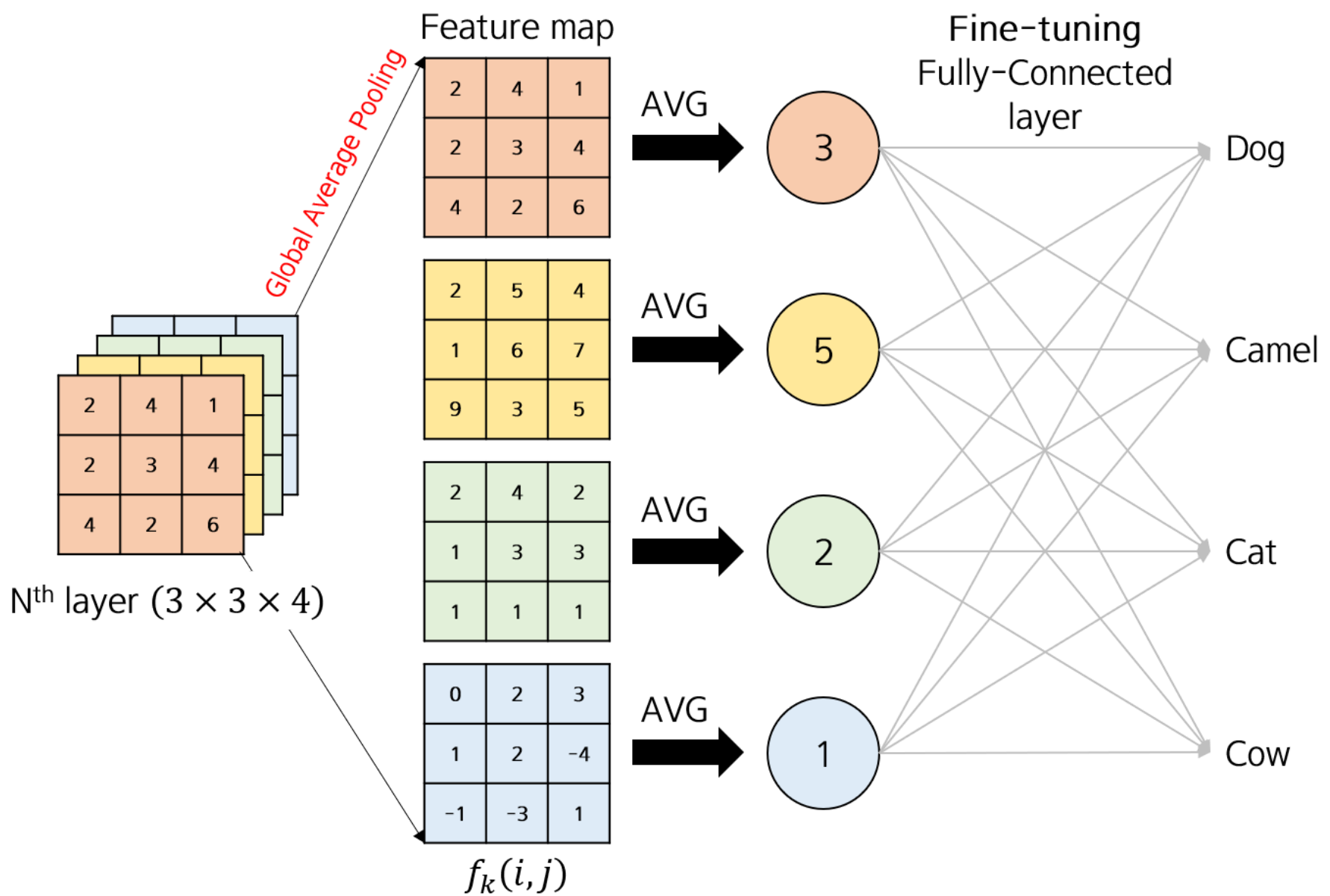
- pixel-wise sum 을 통해서 각 heatmap 을 합쳐주면 CAM 이 완성된다.



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

CAM 의 제한점

Global Average Pooling 을 사용하게 되면 뒷부분을 다시 또 fine tuning 해야 하며, 마지막 CNN layer 의 feature map 에 대해서만 Heat map 을 추출할 수 있다는 점에서 한계가 있다.



GAP 를 사용하려면 각 클래스로 분류하기 위한 Fine tuning 이 이루어져야 한다.

Grad-CAM : Gradient-weighted CAM

CAM 이 2016년 CVPR 에서 발표된 이후, 다음 해 2017년 ICCV 에서 Grad-CAM 이 발표되며 CAM 의 제한점을 해결했다. (<https://arxiv.org/pdf/1610.02391.pdf>)

CNN 모델에서 Gradient 는 특정 클래스 C에 대해 특정 Input K가 주는 영향력이라고 고려할 수 있다. 그런 점에서 GAP 의 출력값인, 가중치를 gradient 로 대체할 수 있게 된다.

그렇다면, Grad-CAM 을 위해서는 GAP 레이어를 사용하지 않아도 된다.

CAM의 수식은 다음과 같으며,

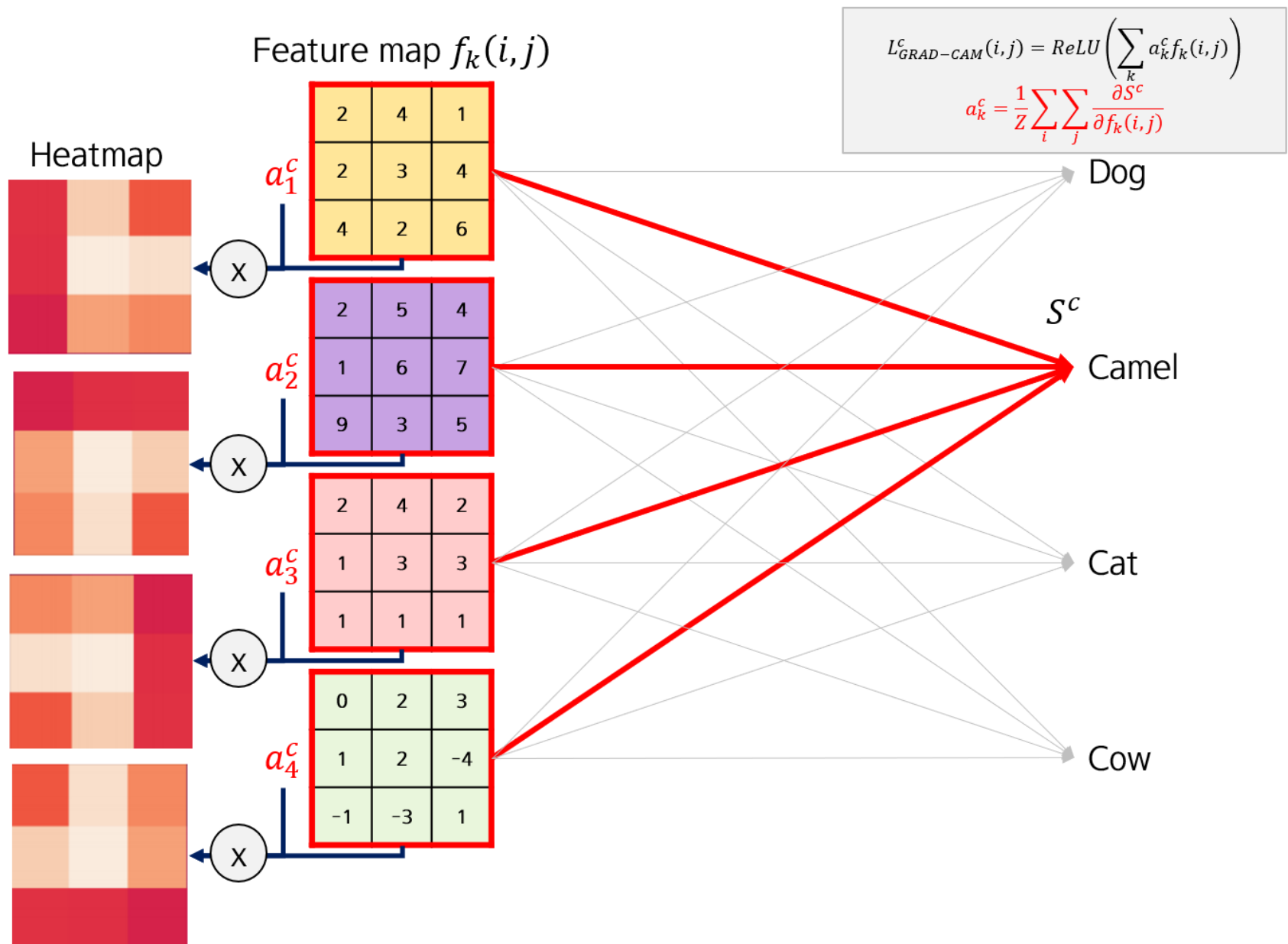
$$L_{CAM}^c(i, j) = \sum_k w_k^c f_k(i, j)$$

Grad-CAM 은 위의 수식에서 w_k^c 를 a_k^c 로 바꾸고, ReLU를 추가한다.

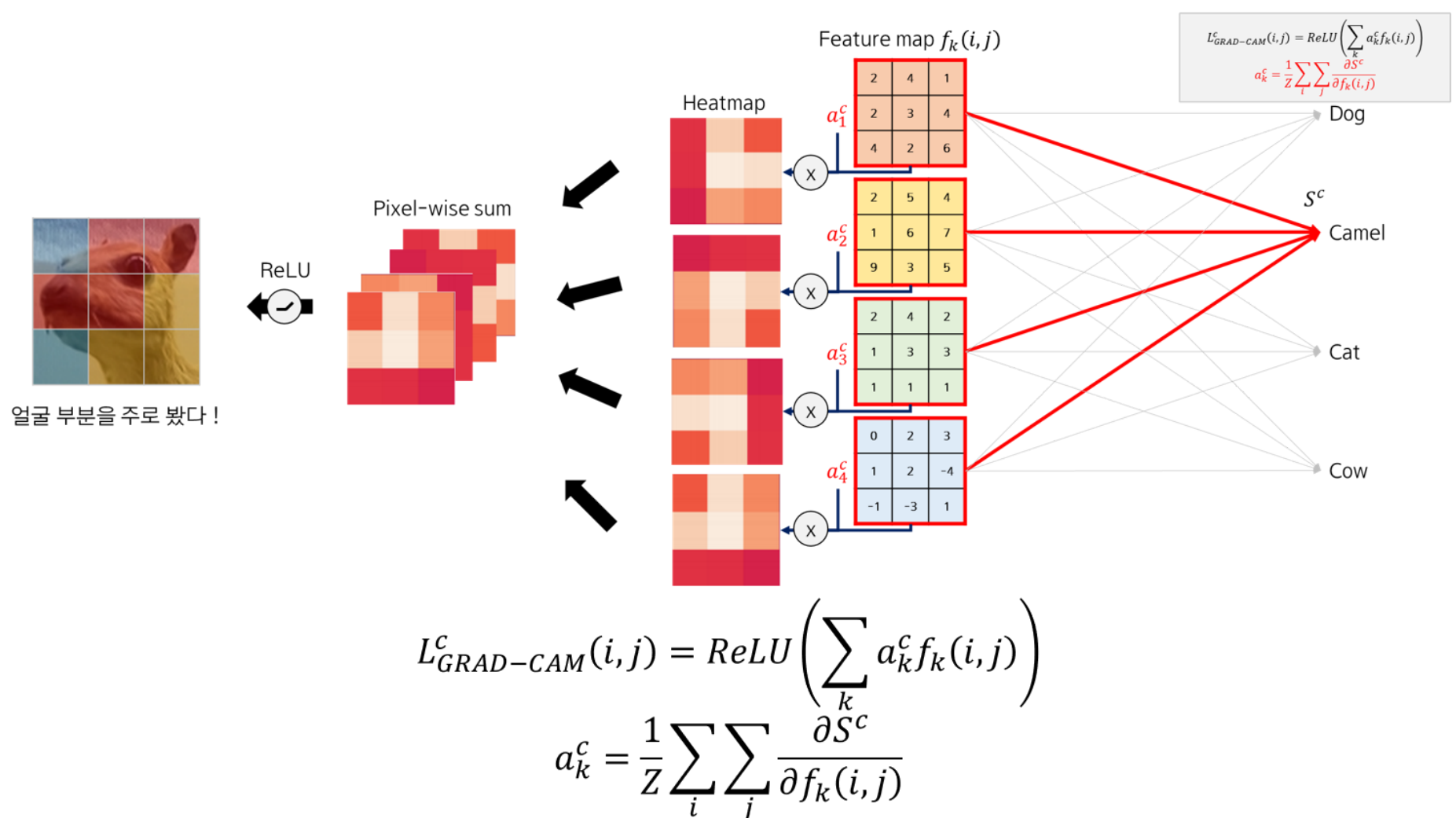
a_k^c 를 엄밀히 말하면, k 번째 feature map $f_k(i, j)$ 의 각 원소 i, j 가 클래스 c에 대한 행렬곱 값 S^c 에 주는 영향력의 평균이라고 고려할 수 있다.

$$L_{Grad-CAM}^c(i, j) = ReLU\left(\sum_k a_k^c f_k(i, j)\right)$$
$$a_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial S^c}{\partial f_k(i, j)}$$

다음의 이미지는 특정 클래스에 대해, gradient 가 feature map 에 곱해져서 Heat map 이 만들어지는 과정을 설명한다.

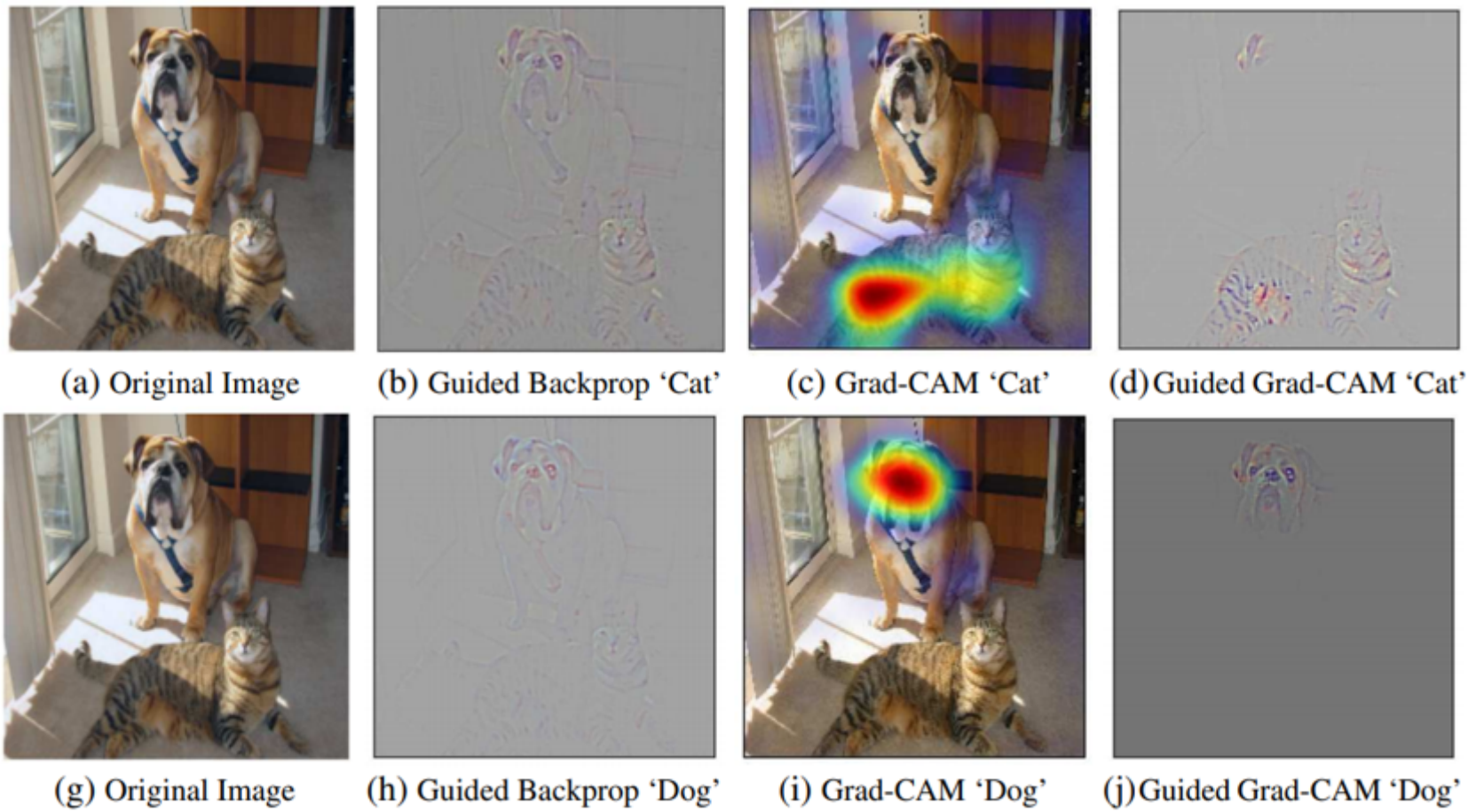


최종적으로 heap map 들이 합쳐져 Grad-CAM으로 시각화되는 과정은 다음의 이미지와 같다.



Grad-CAM 이 적용된 예시

하나의 이미지에 여러 클래스의 객체가 있을 경우, 각 객체의 어느 부분에 주목해서 특정 클래스로 분류했는지 Grad-CAM 을 사용하면 확인할 수 있다.



CAM, Grad-CAM 구현 파일

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/cd08dab0-3256-400b-89b4-6227b93cfa5e/CAM_Grad_CAM.ipynb