AE-PSL

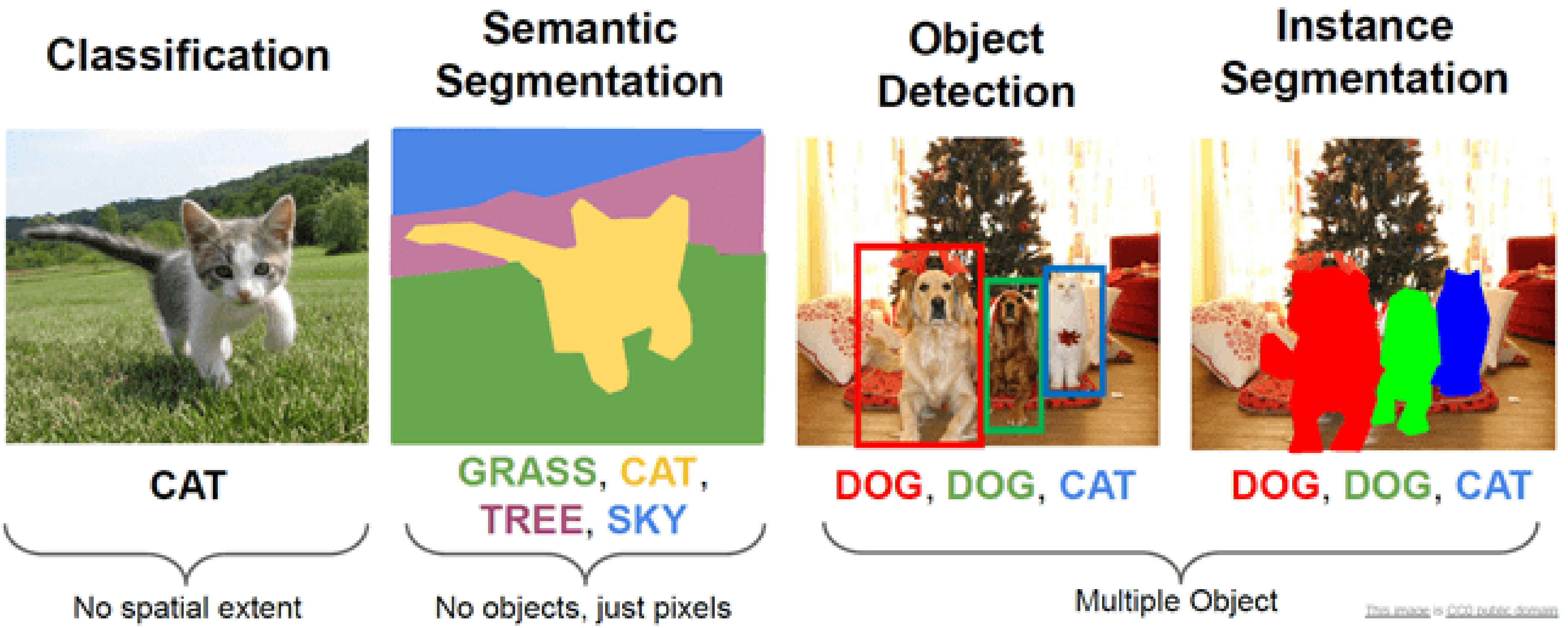
Object Region Mining with Adversarial Erasing: A Simple Classification to Semantic Segmentation Approach

카이스트
김현우

0 발표 목차

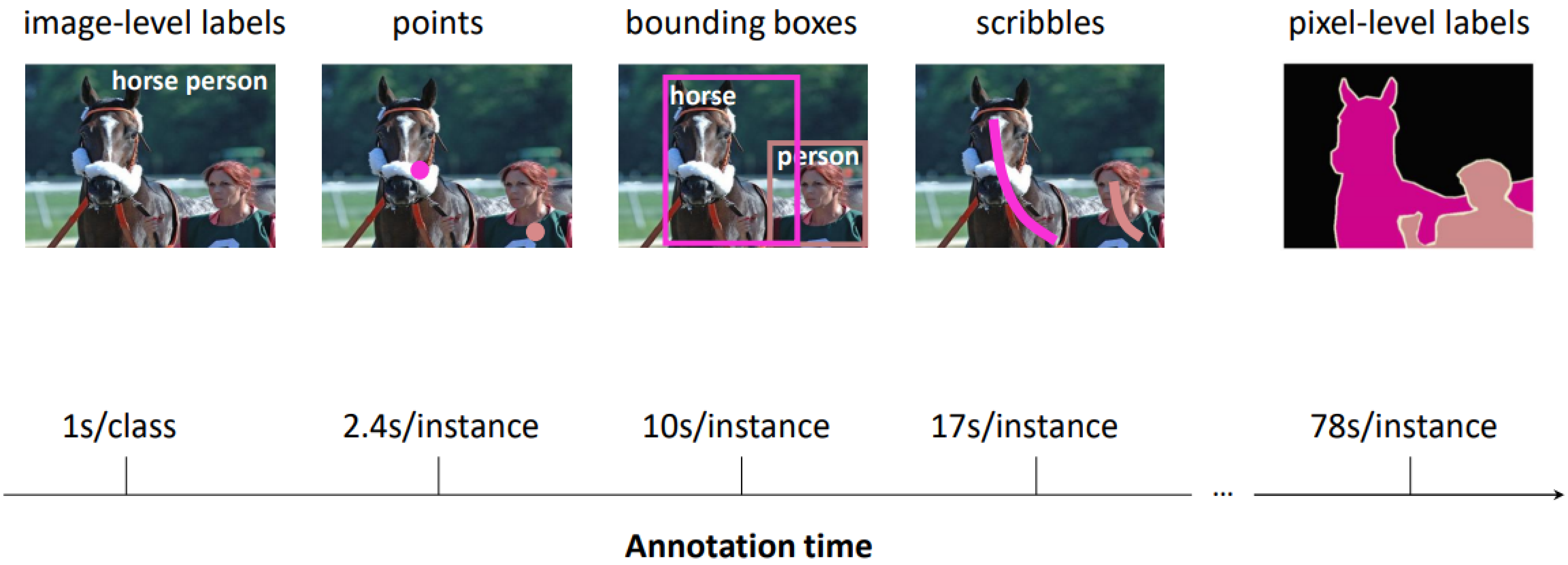
발표 목차

1 WSSS 개요



출처 : http://cs231n.stanford.edu/slides/2020/lecture_12.pdf

1 WSSS 개요



출처 : Lin CVPR16, Berman ECCV16, Hakan CVPR18 Tutorial

1 WSSS 개요

Weak Supervision

Lower degree (or **cheaper**, **simper**) annotations at **training stage** than the required outputs at the **testing stage**.

image-level labels

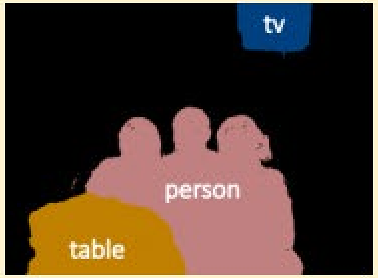
points

bounding boxes

scribbles

Training Stage (Weakly-supervised Annotations)

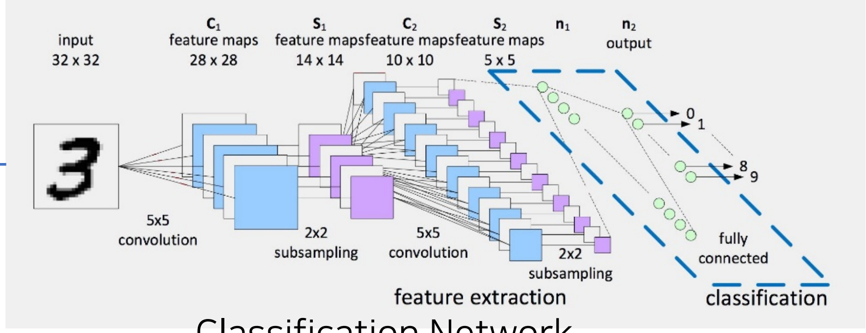
pixel-level labels



Testing Stage

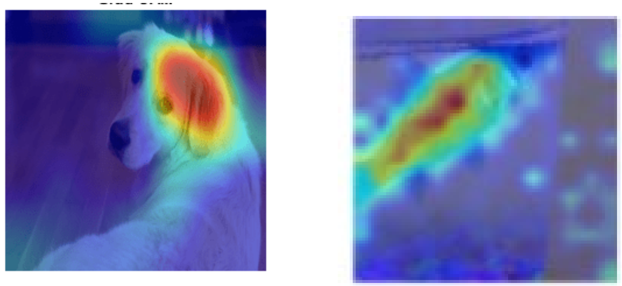
출처 : Hakan CVPR18 Tutorial

1 WSSS 개요

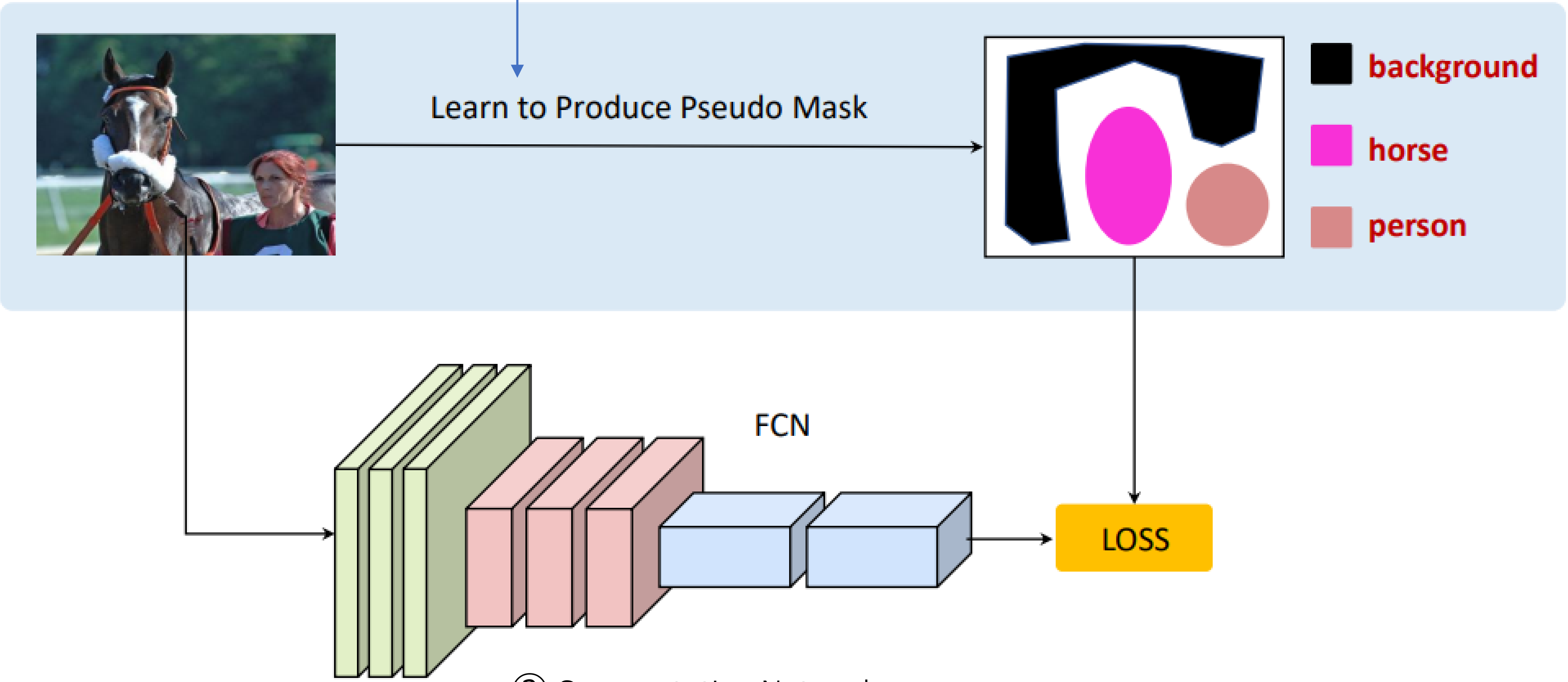
①


Classification Network

+



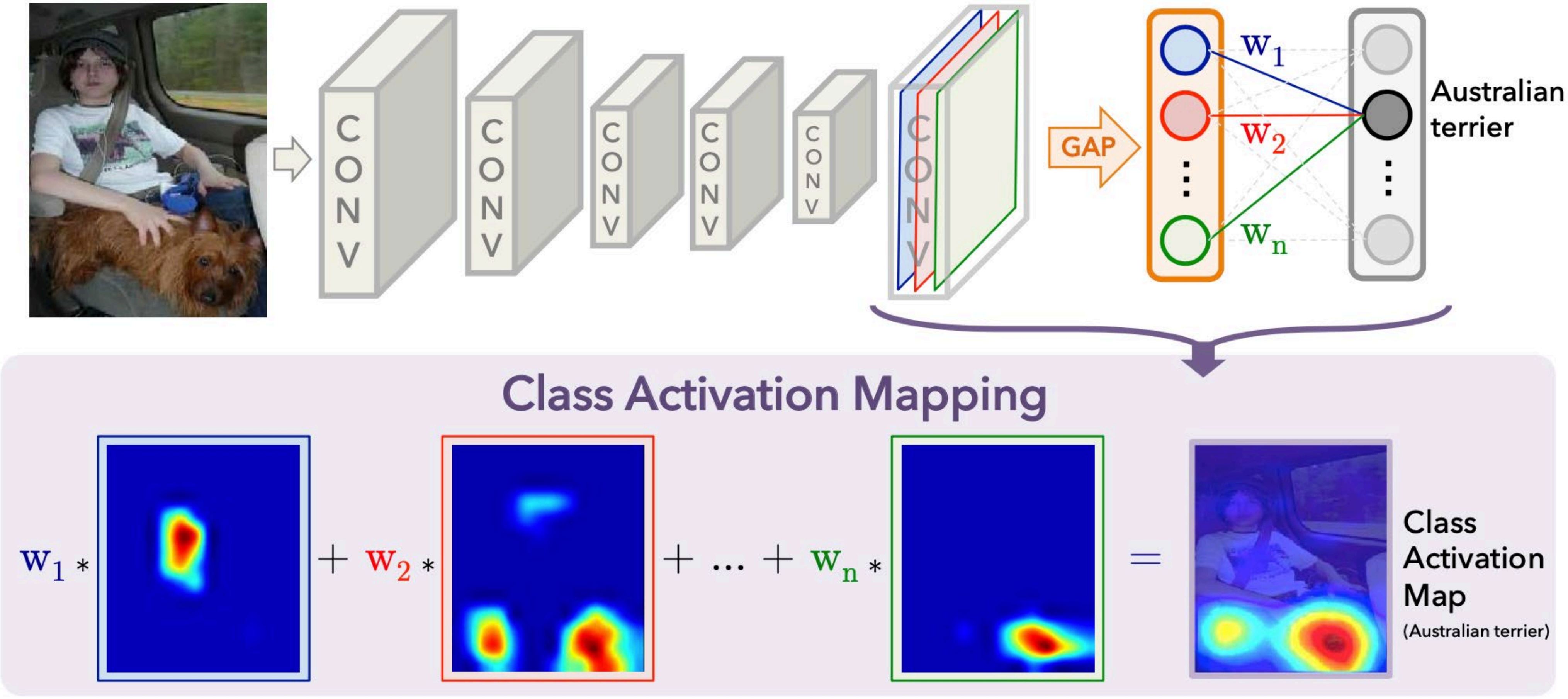
[CAM, Grad-CAM, Attention]



② Segmentation Network

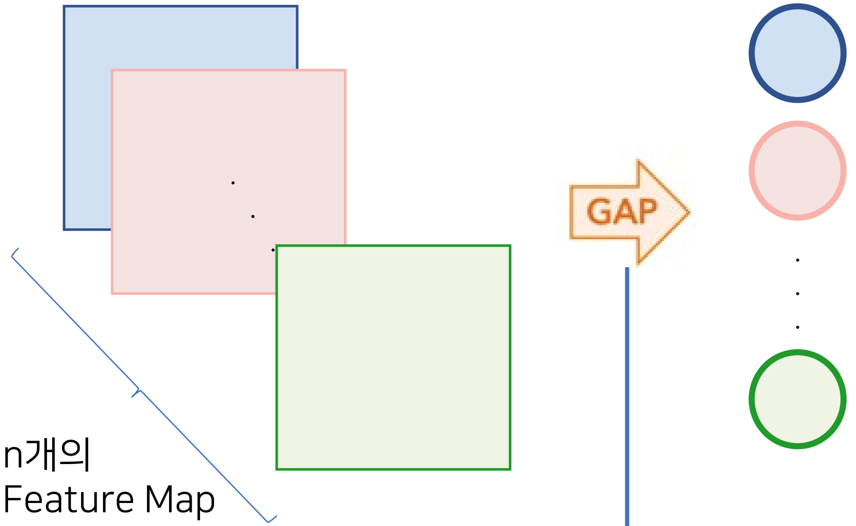
출처 : Yunchao Wei VALSE 2019 Tutorial, **[논문리뷰]Transformer Interpretability Beyond Attention Visualization** by sjinu, https://kr.mathworks.com/help/deeplearning/ug/gradcam-explains-why.html

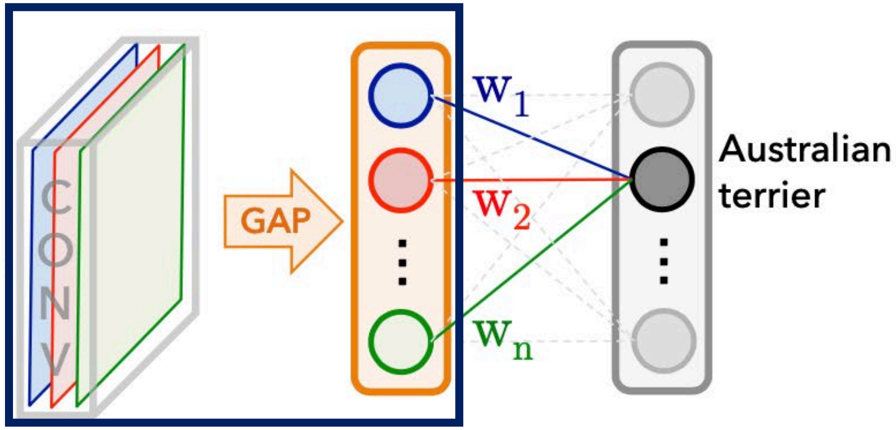
2 CAM & Grad-CAM



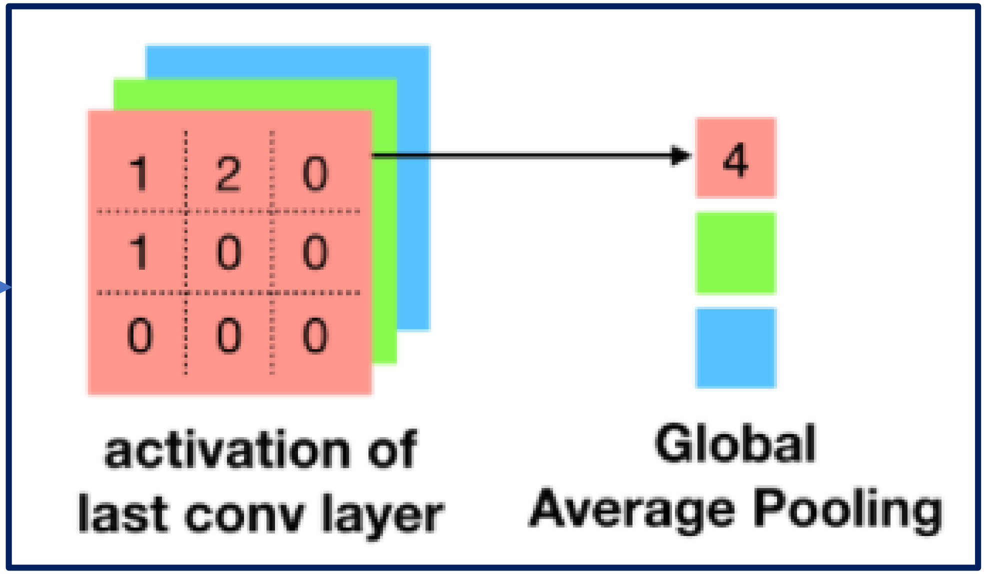
출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization)

2 CAM & Grad-CAM



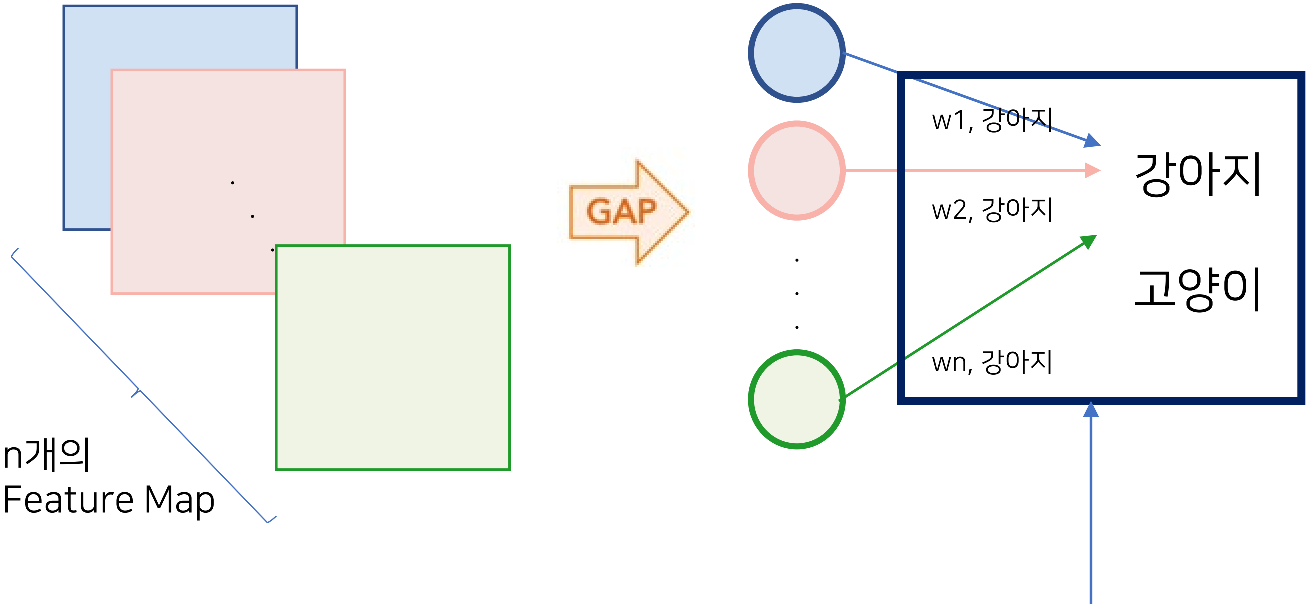


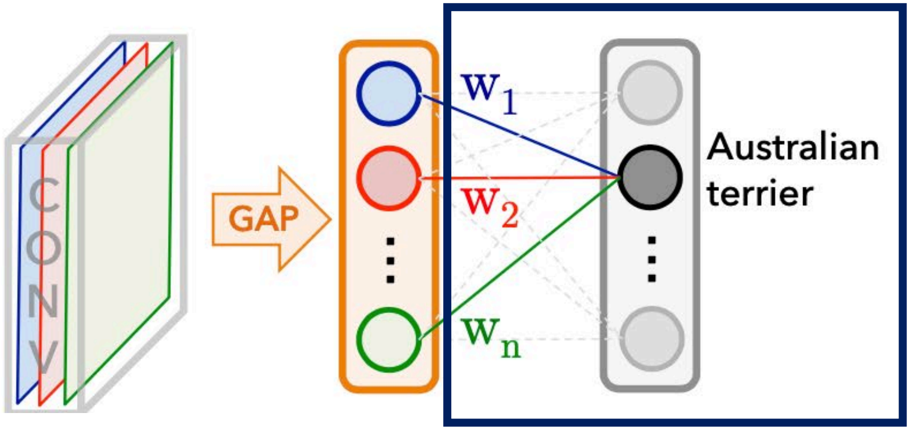
GAP의 결과물은 해당 Feature Map
의 정보를 **대표**한다고 볼 수 있음



출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization), https://jsideas.net/class_activation_map/

2 CAM & Grad-CAM

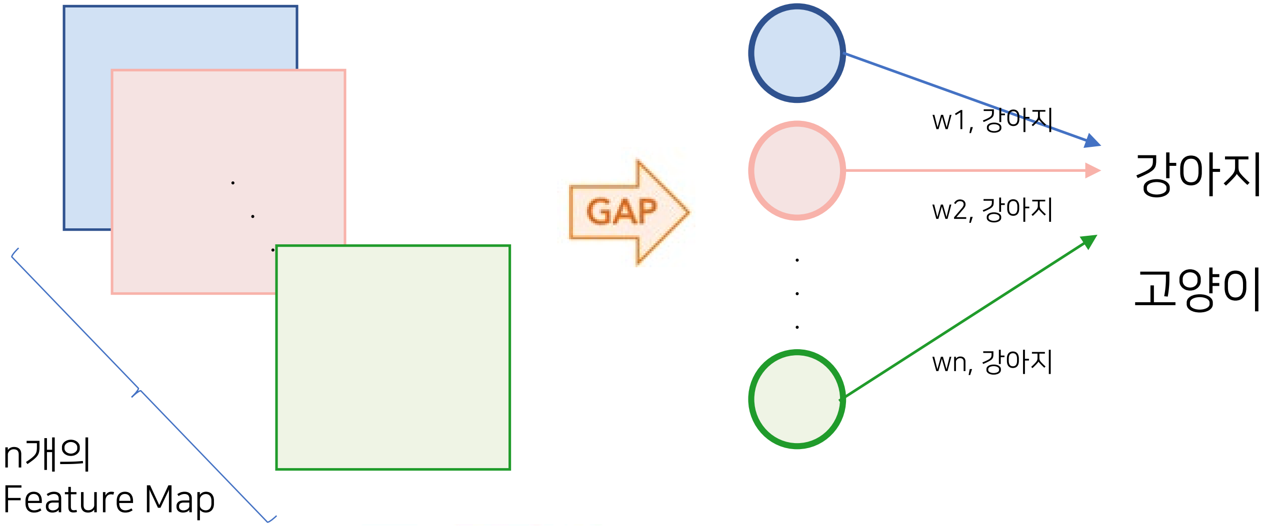


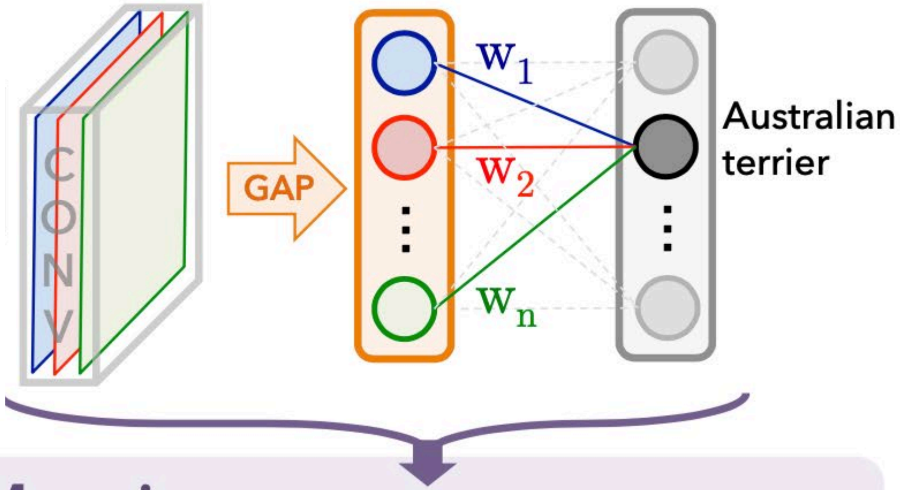


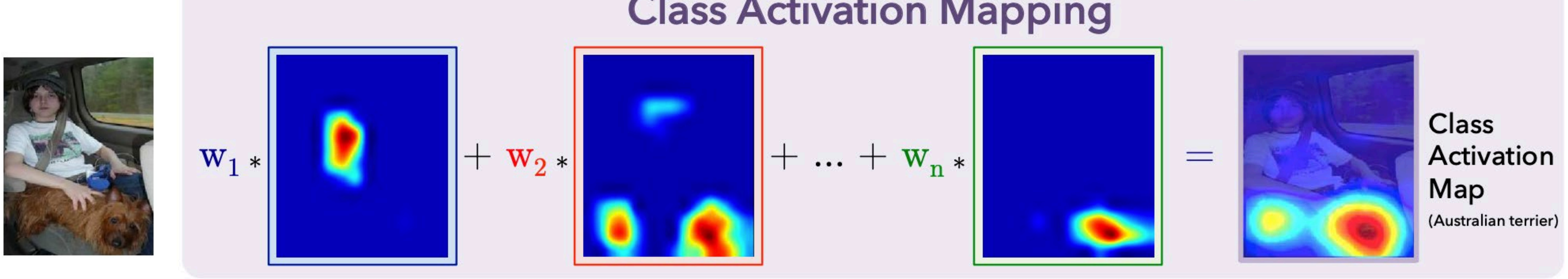
강아지와 고양이를 분류할 때, Feature Map을 대표하는 정보인
GAP가 얼마나 중요한지 (weight)에 대해서 학습

출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization)

2 CAM & Grad-CAM

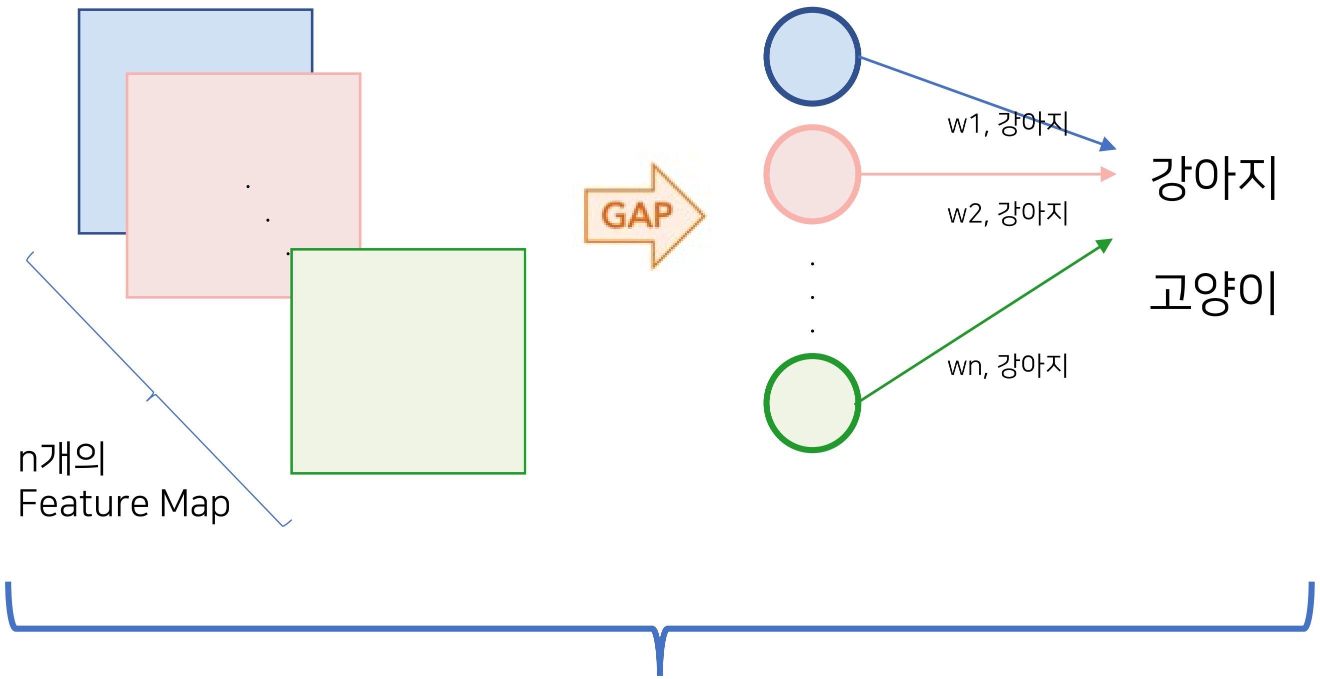






출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization)

2 CAM & Grad-CAM



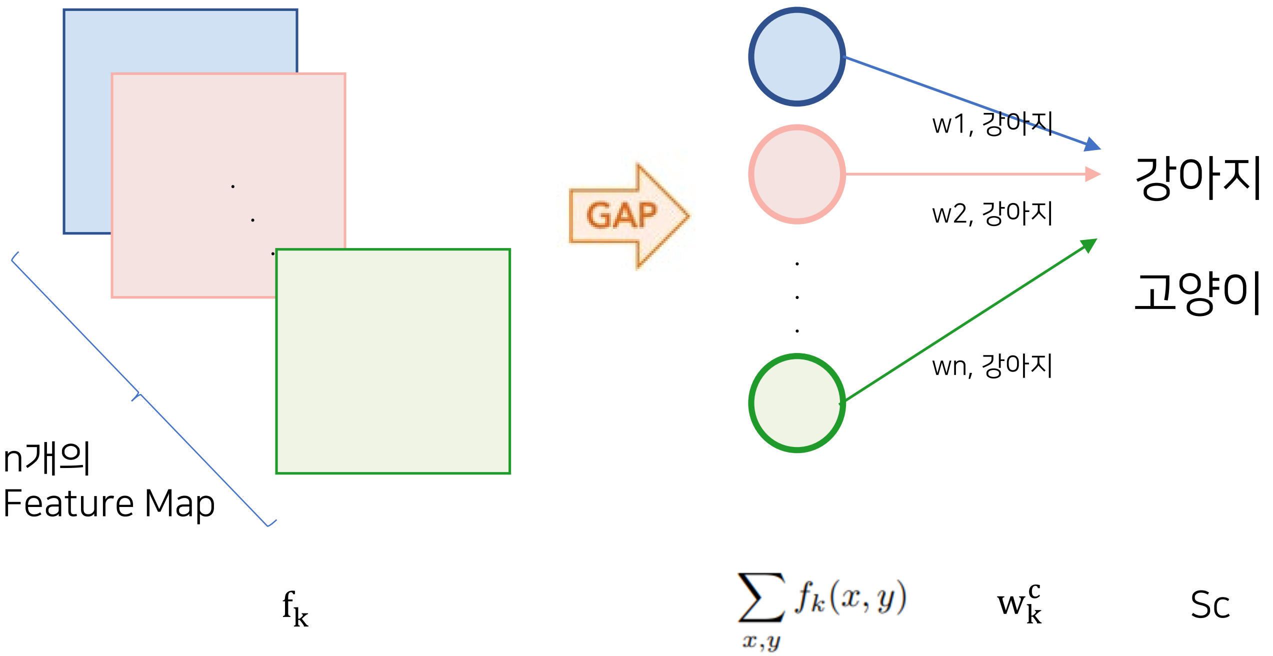
CAM의 문제점

- **마지막 레이어가 GAP**를 가져야함 -> 일반적인 다른 네트워크에 적용하기 어려움
- CAM의 결과를 **마지막 레이어에서** 밖에 뽑을 수 없음 -> 이전 레이어들이 어떻게 활성화 되고 있는지 확인하기 어려움

위의 2가지 문제점을 해결한 연구 : **Grad-CAM**

출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization)

2 CAM & Grad-CAM



$$S_c = \sum_k w_k^c \sum_{x,y} f_k(x,y)$$

Sc : 클래스 c에 대한 Score (위의 예시에서 c는 강아지)
k : Feature Map의 개수 (k=1 … n)
w_k^c : 클래스 c 예측시 사용되는 k번째 Feature Map
f_k : k번째 Feature map
x, y : Feature map의 x, y번째 좌표

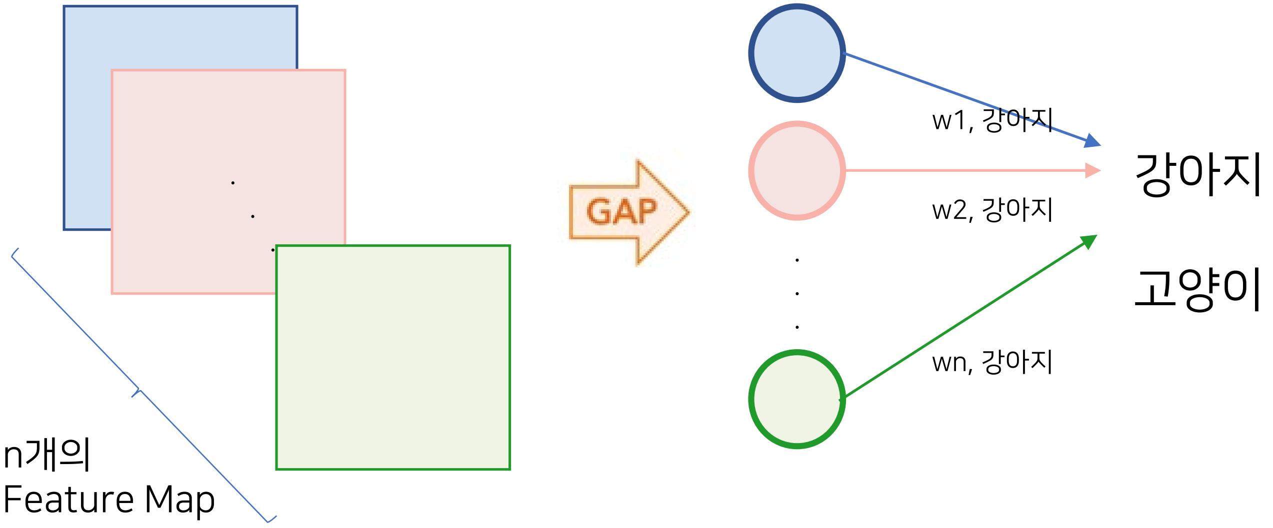
(의문 : 나누기 Feature map 크기 안써진거 같은데 이렇게 적어도 되나 …?)

f_k 는 이전 Layer에서도 추출할 수 있는데 w_k^c 는 마지막 Layer에서만 추출이 가능함. 그렇다면 w_k^c 를 대체할 만한 **요소**가 있을까? (이 요소는 다음과 같은 조건을 가져야함)

- 강아지, 고양이의 클래스를 구분하는데 얼마나 중요한지에 대한 영향
- 이전 Feature Map에서도 추출이 가능해야함

출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization)

2 CAM & Grad-CAM

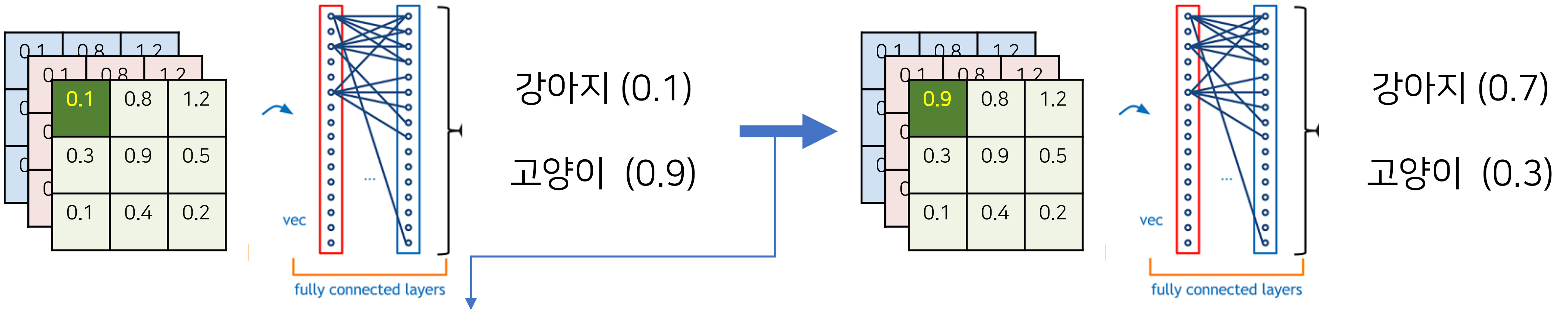


$$S_c = \sum_k w_k^c \sum_{x,y} f_k(x,y)$$

Sc : 클래스 c에 대한 Score (위의 예시에서 c는 강아지)
k : Feature Map의 개수 (k=1 ... n)
w_k^c : 클래스 c 예측시 사용되는 k번째 Feature Map
f_k : k번째 Feature map
x, y : Feature map의 x, y번째 좌표

(의문 : 나누기 Feature map 크기 안써진거 같은데 이렇게 적어도 되나 ...?)

f_k $\sum_{x,y} f_k(x,y)$ w_k^c Sc



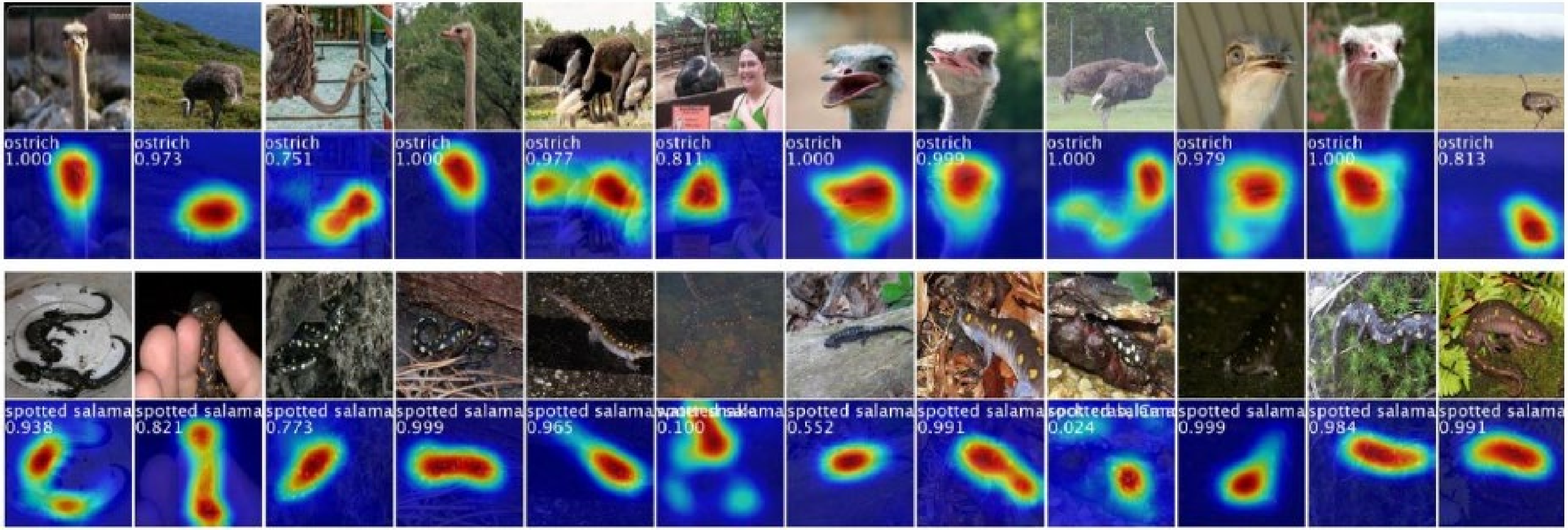
$$\sum_{x,y} \frac{\partial y_c}{\partial f_k(x,y)}$$

타겟의 스코어가 바뀔때 해당 Feature Map 요소의 중요한지
해당 Feature Map의 요소를 다 더하면 해당 Feature Map이 Class에 끼치는 중요도를 알 수 있음

출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization),
https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

3 Resulting Masks are not Sharp

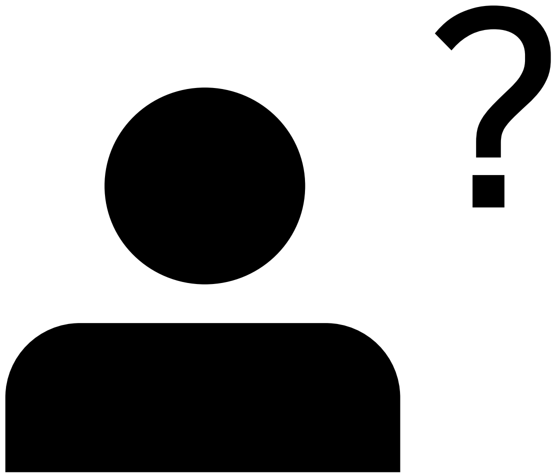
1. CAM의 결과가 Sharp하지 않음
2. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음



출처 : Tutorial by Rodrigo Benenson

3 Resulting Masks are not Sharp

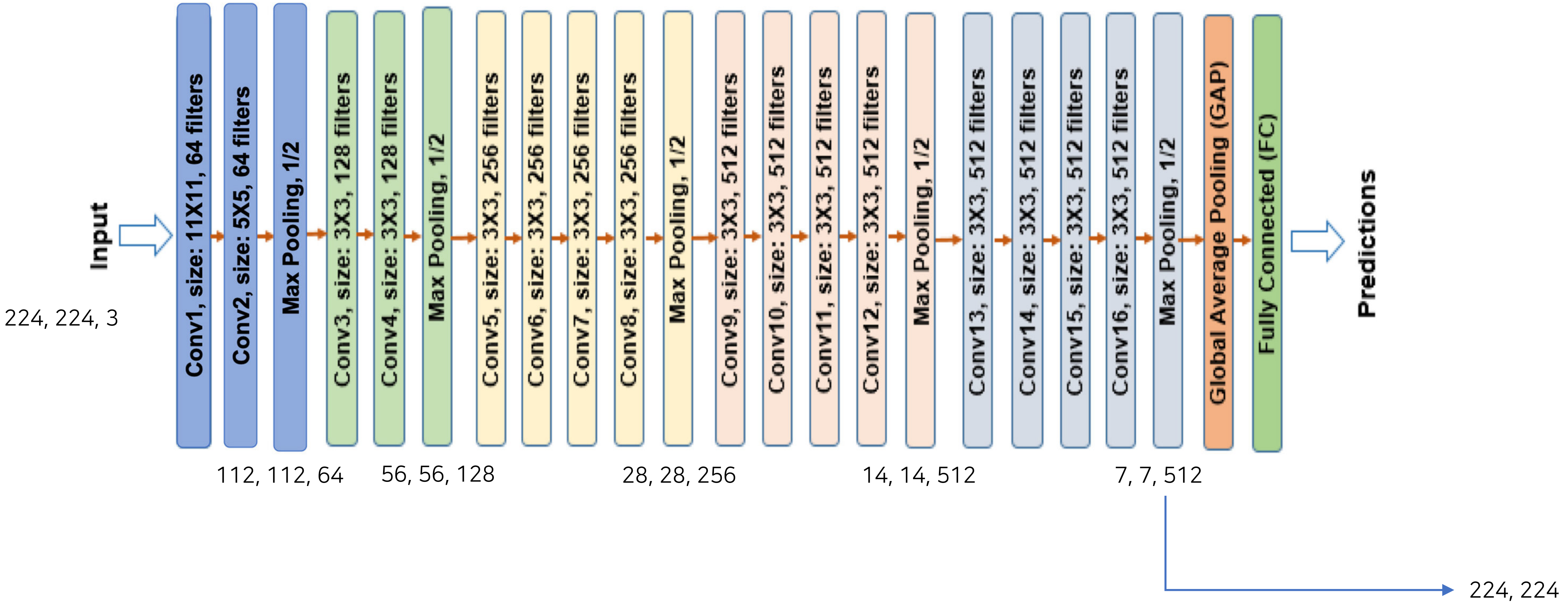
1. CAM의 결과가 Sharp하지 않음



출처 : Tutorial by Rodrigo Benenson

3 Resulting Masks are not Sharp

1. CAM의 결과가 Sharp하지 않음

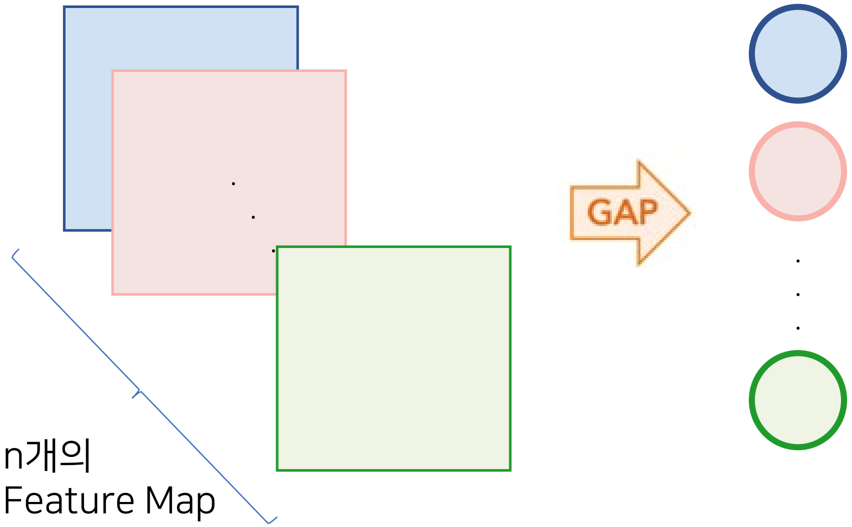


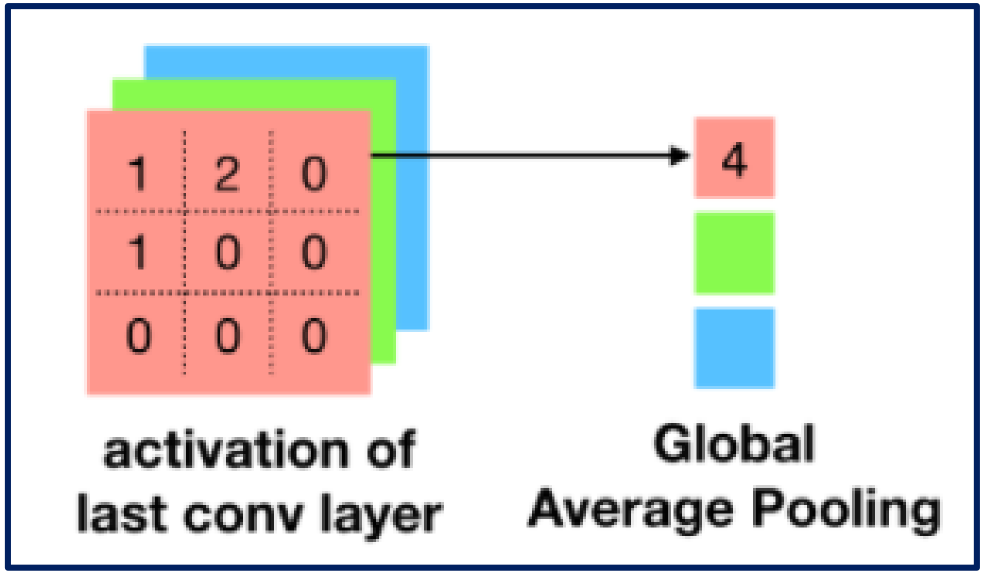
이유1. CAM의 결과를 Original Image에 맞게 매핑하려면 32배 만큼을 더 키워야함 (작은 Feature Map에서 발생한 CAM)

출처 : Deep Feature Selection and Causal Analysis of Alzheimer's Disease

3 Resulting Masks are not Sharp

1. CAM의 결과가 Sharp하지 않음



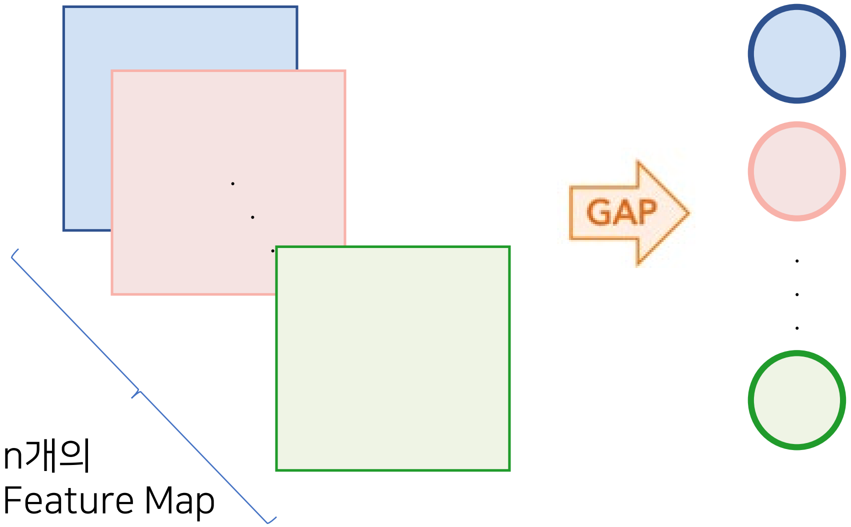


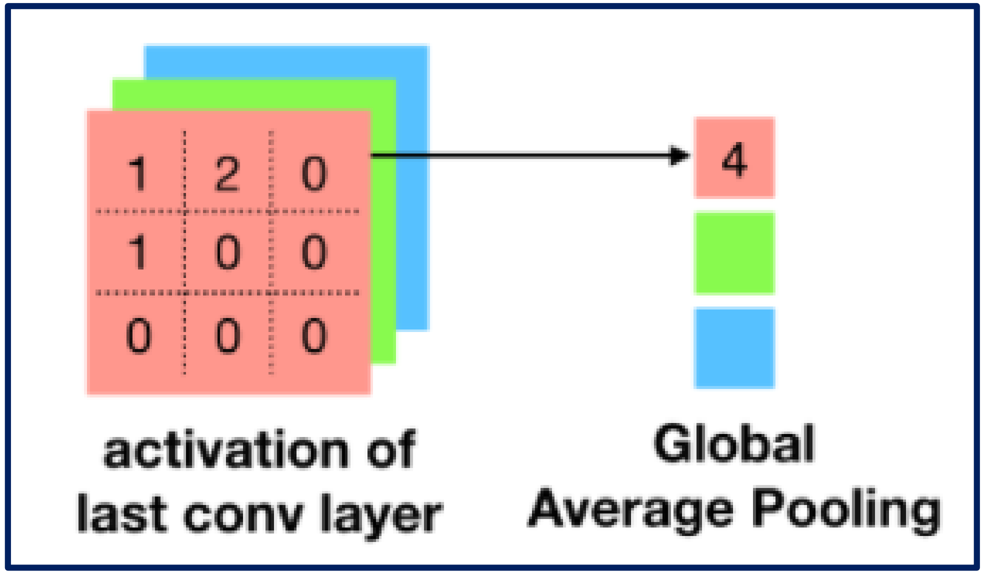
이유2. Global Average Pooling 특성상 전체 Feature Map에 대한 평균이기에 Object 영역이외의 공간도 차지하는 문제가 발생

출처 : https://jsideas.net/class_activation_map/

3 Resulting Masks are not Sharp

1. CAM의 결과가 Sharp하지 않음



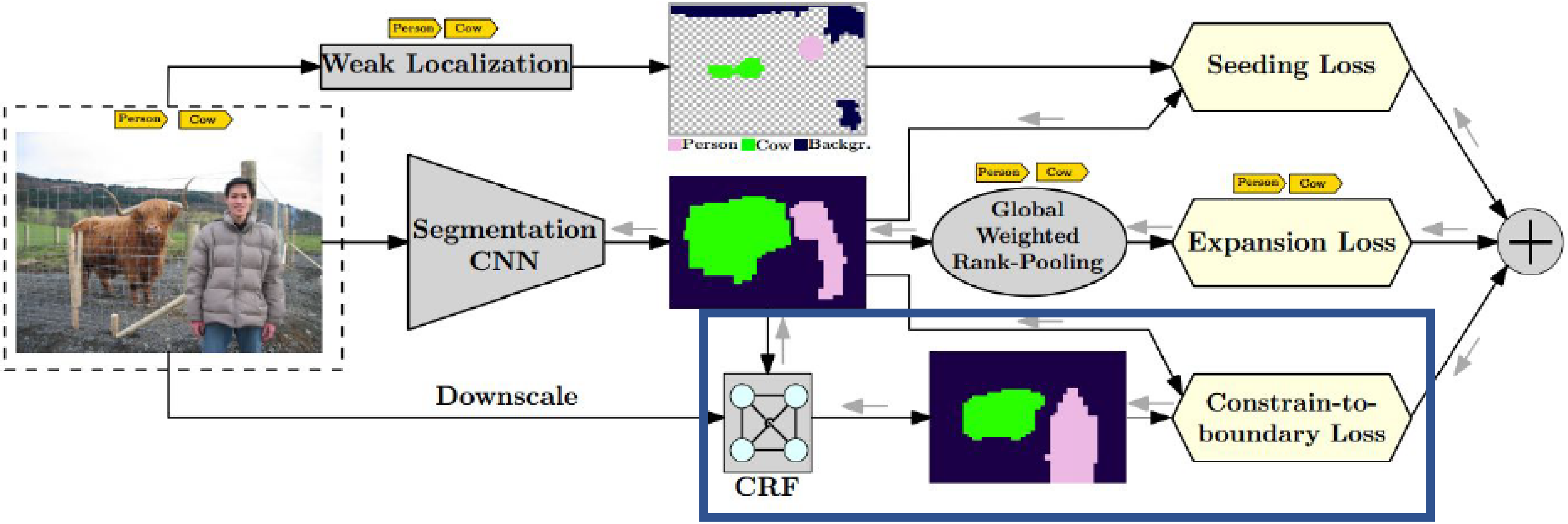


이유2. Global Average Pooling 특성상 전체 Feature Map에 대한 평균이기에 Object 영역이외의 공간도 차지하는 문제가 발생

출처 : https://jsideas.net/class_activation_map/

3 Resulting Masks are not Sharp

Boundary를 인식하는 Propagation을 쓰자 ! (Constrain-to-boundary loss)

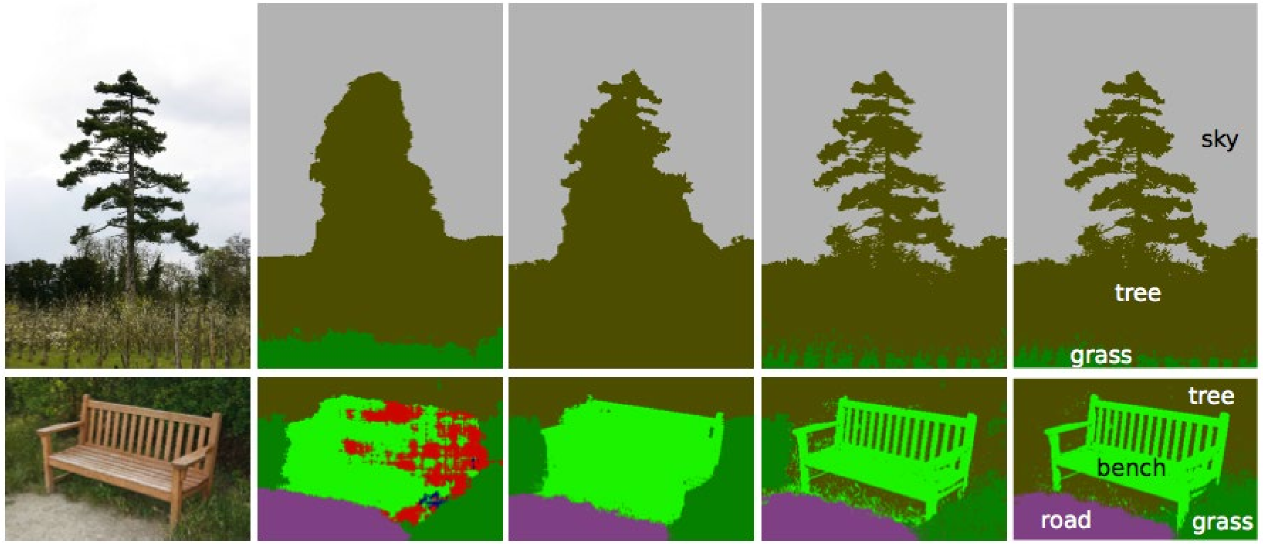


$$\min_{\theta} \sum_{(X,T)\in\mathcal{D}} \left[L_{\text{seed}}(f(X;\theta),T) + L_{\text{expand}}(f(X;\theta),T) + L_{\text{constrain}}(X, f(X;\theta)) \right].$$

$$L_{\text{constrain}}(X, f(X)) = \frac{1}{n} \sum_{u=1}^{n} \sum_{c\in\mathcal{C}} Q_{u,c}(X, f(X)) \log \frac{Q_{u,c}(X, f(X))}{f_{u,c}(X)}.$$

출처 : Kolesnikov & Lampert ECCV 2016

3 Resulting Masks are not Sharp



(a) Image (b) Unary classifiers (c) Robust P^n CRF (d) Fully connected CRF, MCMC inference, 36 hrs (e) Fully connected CRF, our approach, 0.2 seconds

$$E(\boldsymbol{x}) = \sum_i \theta_i(x_i) + \sum_{ij} \theta_{ij}(x_i, x_j) \longleftarrow$$ Fully connected model

From DCNN label probabilities

Gaussian, pairwise

$$w_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + w_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)$$

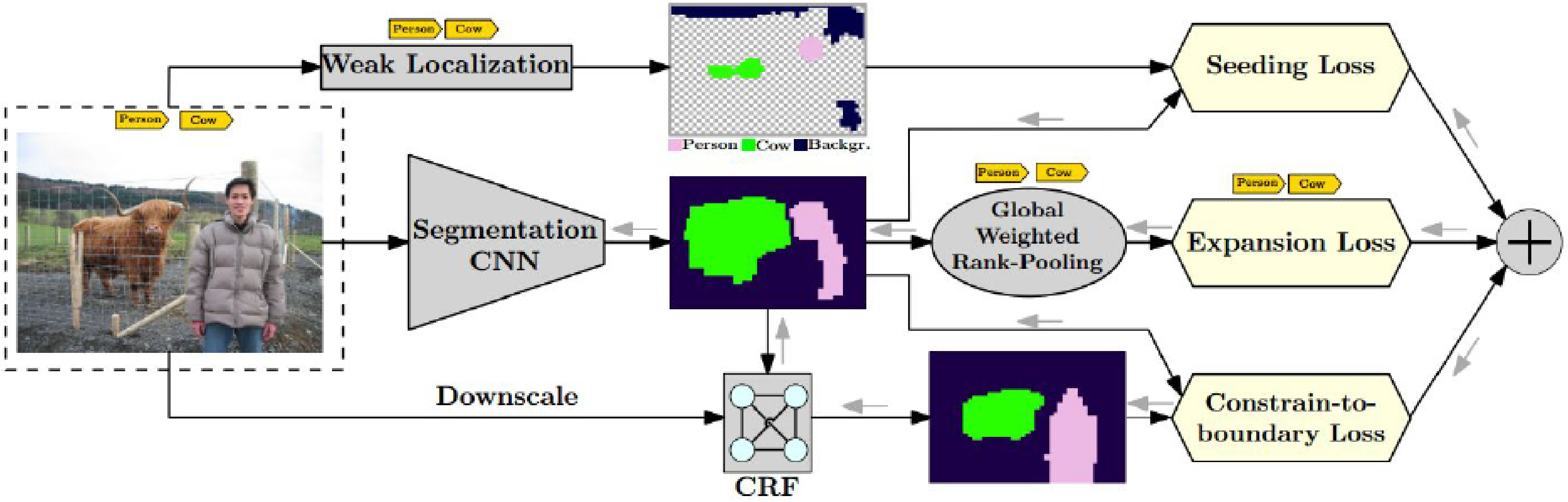
Differences in position and intensity

Just position

출처 : Inference in Fully Connected CRFs with Gaussian Edge Potentials,
DeepLab-semantic image segmentation

3 Resulting Masks are not Sharp

Boundary를 인식하는 Propagation을 쓰자 ! (Constrain-to-boundary loss)



$$L_{\text{seed}}(f(X), T, S_c) = -\frac{1}{\sum_{c\in T}|S_c|}\sum_{c\in T}\sum_{u\in S_c}\log f_{u,c}(X).$$

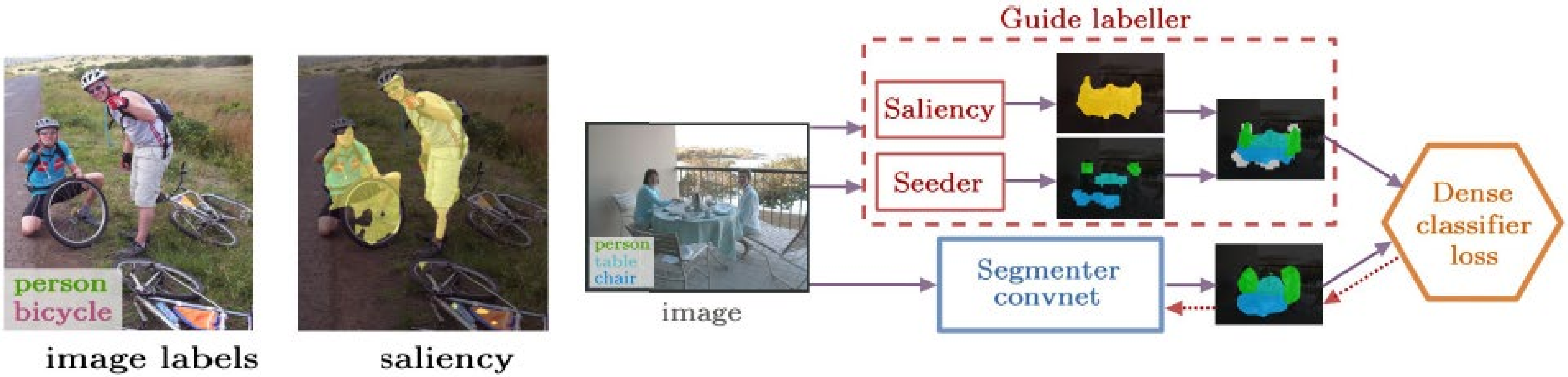
$$L_{\text{expand}}(f(X), T) = -\frac{1}{|T|}\sum_{c\in T}\log G_c(f(X); d_+) - \frac{1}{|\mathcal{C}'\backslash T|}\sum_{c\in\mathcal{C}'\backslash T}\log(1 - G_c(f(X); d_-)) - \log G_{c^{\text{bg}}}(f(X); d_{\text{bg}}). \quad (4)$$

$$L_{\text{constrain}}(X, f(X)) = \frac{1}{n}\sum_{u=1}^{n}\sum_{c\in\mathcal{C}} Q_{u,c}(X, f(X)) \log\frac{Q_{u,c}(X, f(X))}{f_{u,c}(X)}.$$

출처 : Kolesnikov & Lampert ECCV 2016

3 Resulting Masks are not Sharp

Transfer Learning을 통해서 Saliency cues를 같이 활용해보자 !!!



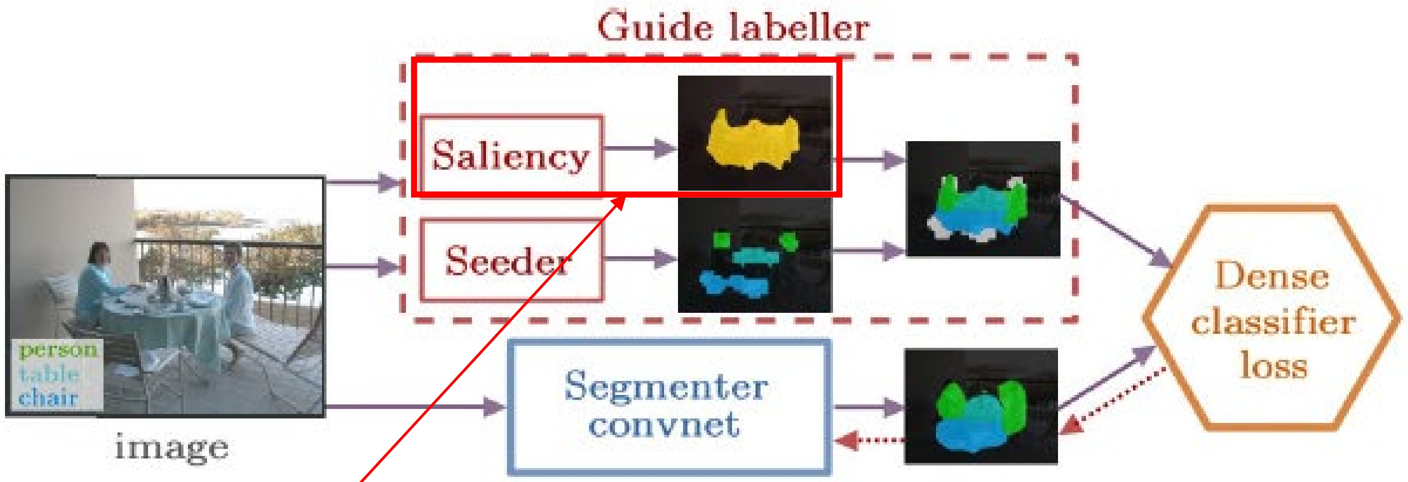
it can refer to a **spatial probability map of where a person might look first** , **a probability map of which object a person might look first** , or a **binary mask segmenting the one object a person is most likely to look first**

출처 : Oh et al. CVPR 2017

3 Resulting Masks are not Sharp

Transfer Learning을 통해서 Saliency cues를 같이 활용해보자 !!!

image labels saliency



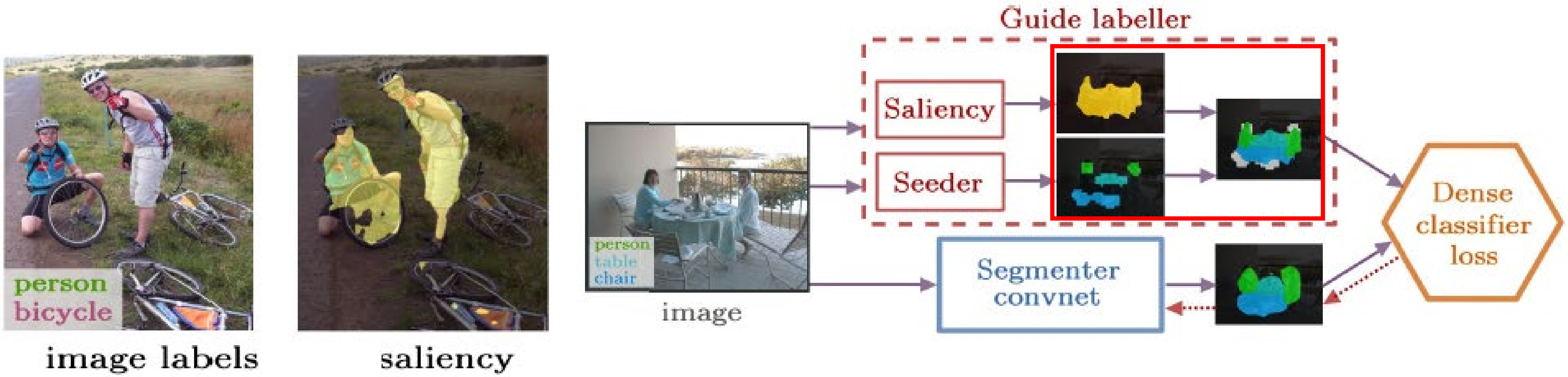
MSRA 데이터셋으로 학습한
모델로 Saliency를 추출

MSRA 데이터셋

출처 : Oh et al. CVPR 2017, MSRA10K Salient Object Database

3 Resulting Masks are not Sharp

Transfer Learning을 통해서 Saliency cues를 같이 활용해보자 !!!

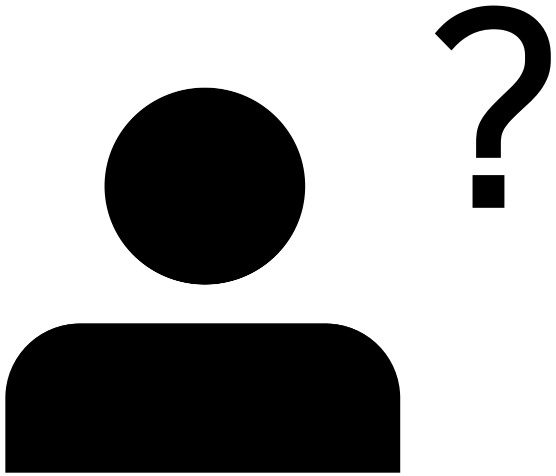


1) We treat the seeds as reliable small size point predictors of each object instance, but that might leak outside of the object.
2) We assume the saliency might trigger on objects that are not part of the classes of interest.
3) A foreground connected component R f g i should take the label of the seed touching it,
4) If two (or more) seeds touch the same foreground component, then we want to propagate all the seed labels inside it.
5) When in doubt, mark as ignore

출처 : Oh et al. CVPR 2017, MSRA10K Salient Object Database

4 Focused on Discriminative Area Only

1. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음



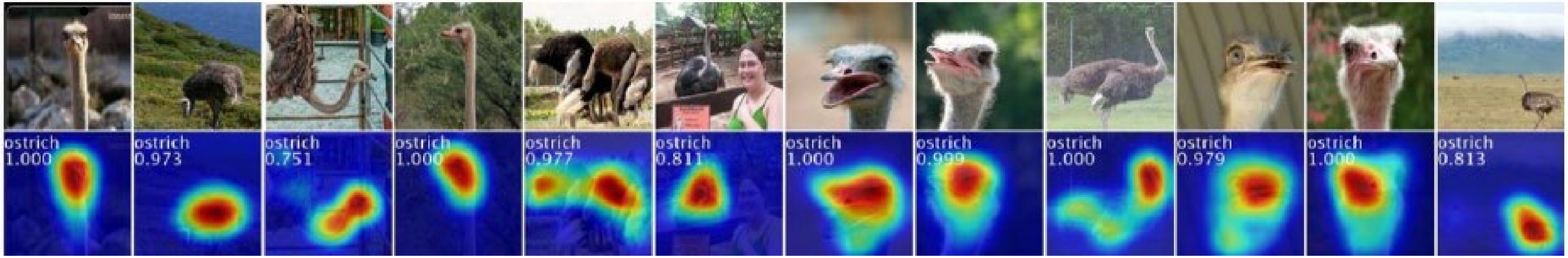
4 Focused on Discriminative Area Only

1. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음
   - 이유1. CAM의 경우 Classification을 목적으로 하기에 Object Localization과는 다름

$$\text{Loss} = -\sum_{i=1}^{\substack{\text{output}\\\text{size}}} y_i \cdot \log \hat{y}_i$$

4 Focused on Discriminative Area Only

1. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음
    - 이유1. CAM의 경우 Classification을 목적으로 하기에 Object Localization과는 다름
    - 이유2. 같은 클래스임에도 서로 다른 모습이기에 두드러진 특징을 가진 부분에 의존 (Intra-category variations)



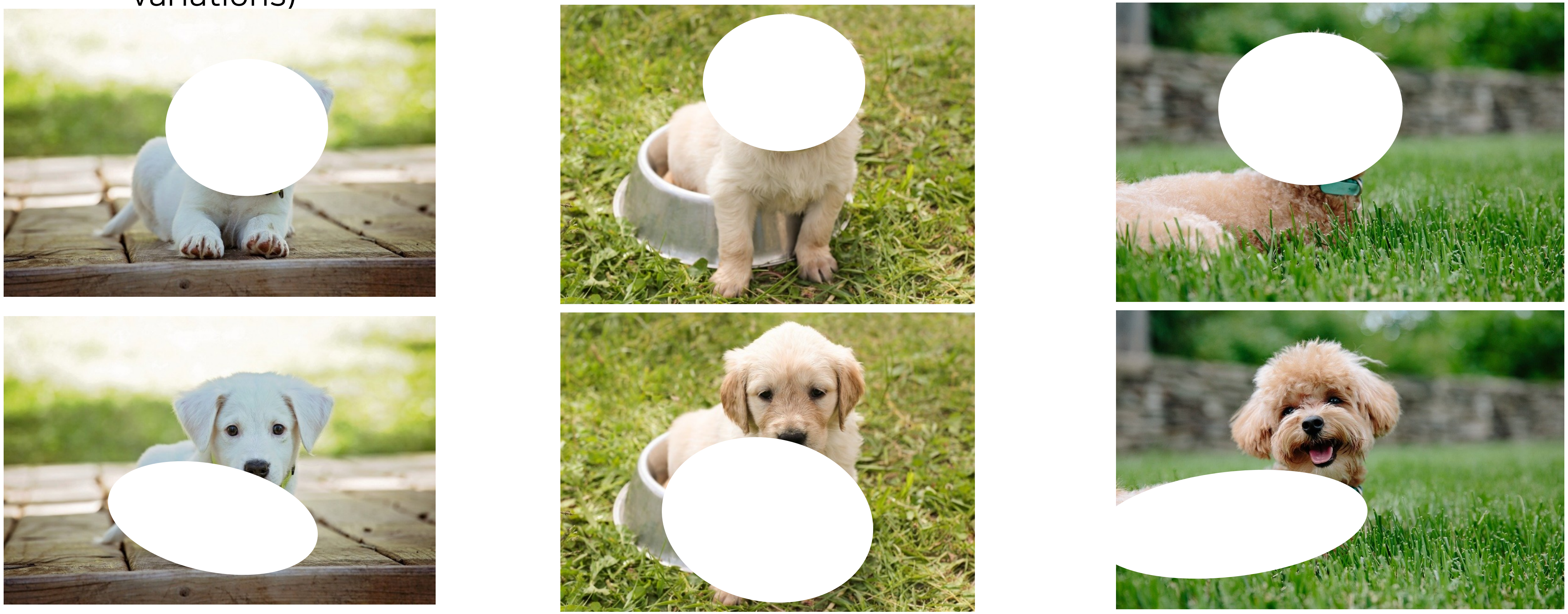
출처 : Tutorial by Rodrigo Benenson

4 Focused on Discriminative Area Only

1. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음
   - 이유1. CAM의 경우 Classification을 목적으로 하기에 Object Localization과는 다름
   - 이유2. 같은 클래스임에도 서로 다른 모습이기에 두드러진 특징을 가진 부분에 의존 (Intra-category variations)

[몸만 보는 경우]

[얼굴만 보는 경우]



출처 : Tutorial by Rodrigo Benenson

4 Focused on Discriminative Area Only

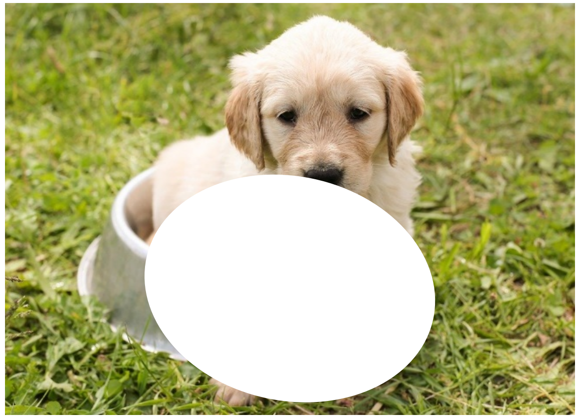
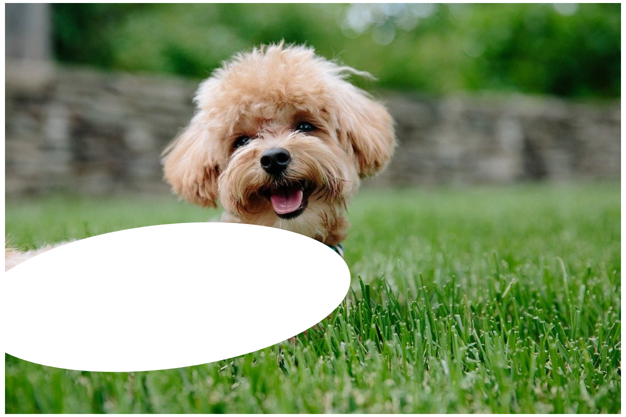
1. 오직 Discriminative 한 영역(Classification 할때 중요한 부분)만 집중하는 경향이 있음
   - 이유1. CAM의 경우 Classification을 목적으로 하기에 Object Localization과는 다름
   - 이유2. 같은 클래스임에도 서로 다른 모습이기에 두드러진 특징을 가진 부분에 의존 (Intra-category variations)

[몸만 보는 경우]



얼굴만 보는 경우가 몸통만 보는 경우보다 훨씬 강아지임을 인식하기 편함 !!!

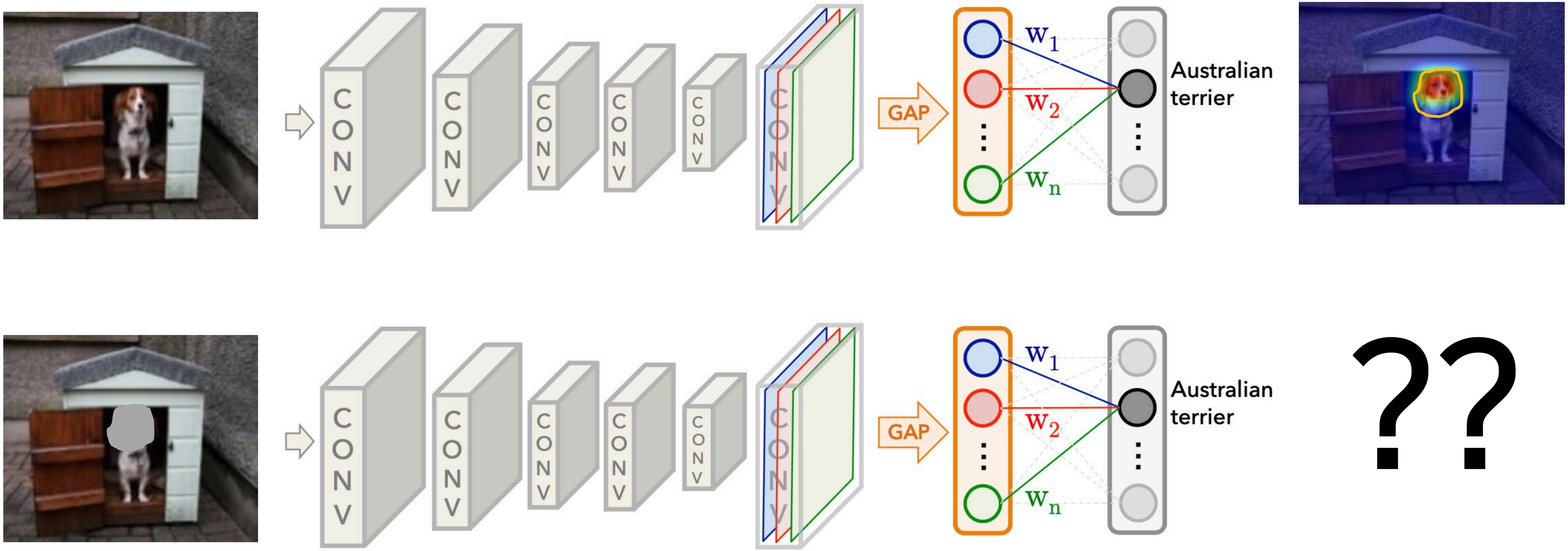
[얼굴만 보는 경우]

출처 : Tutorial by Rodrigo Benenson

4 Focused on Discriminative Area Only

Discriminative 영역만 계속해서 CAM으로 잡히는게 문제면 해당 영역을 지우고 CAM을 뽑으면 어떨까?



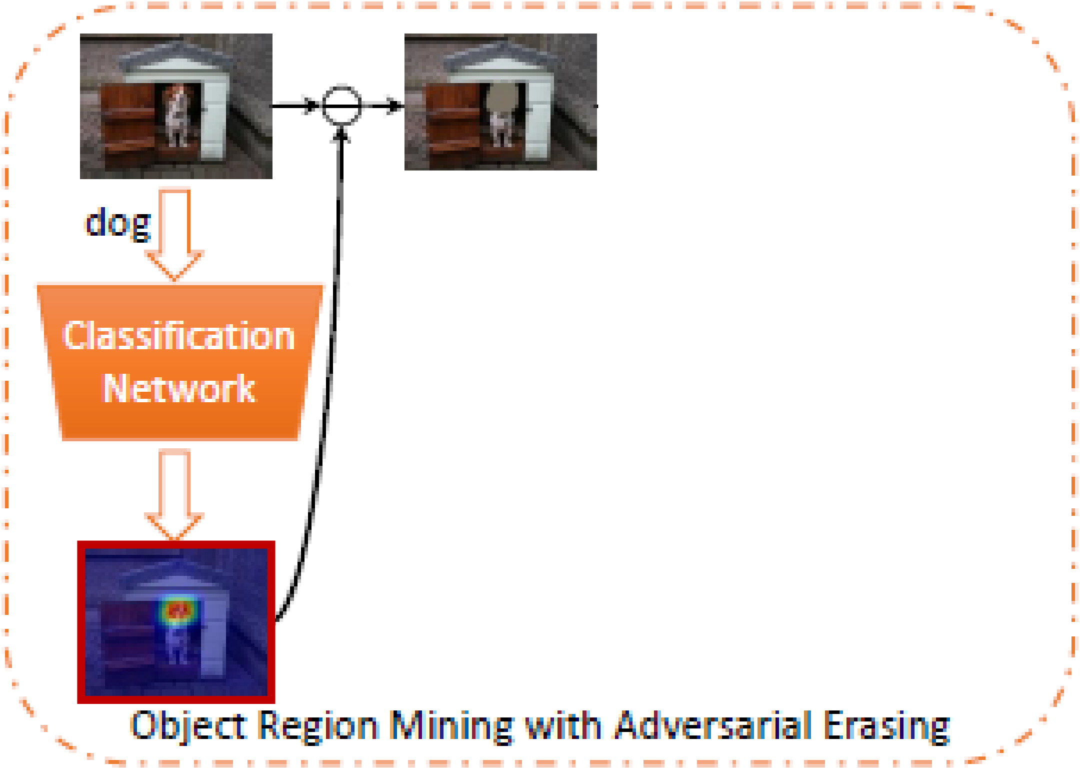
출처 : CAM(Class Activation Map-Learning Deep Features for Discriminative Localization,
Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

1. 입력 이미지를 Classification Network1을 통해 학습하고 CAM을 추출

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

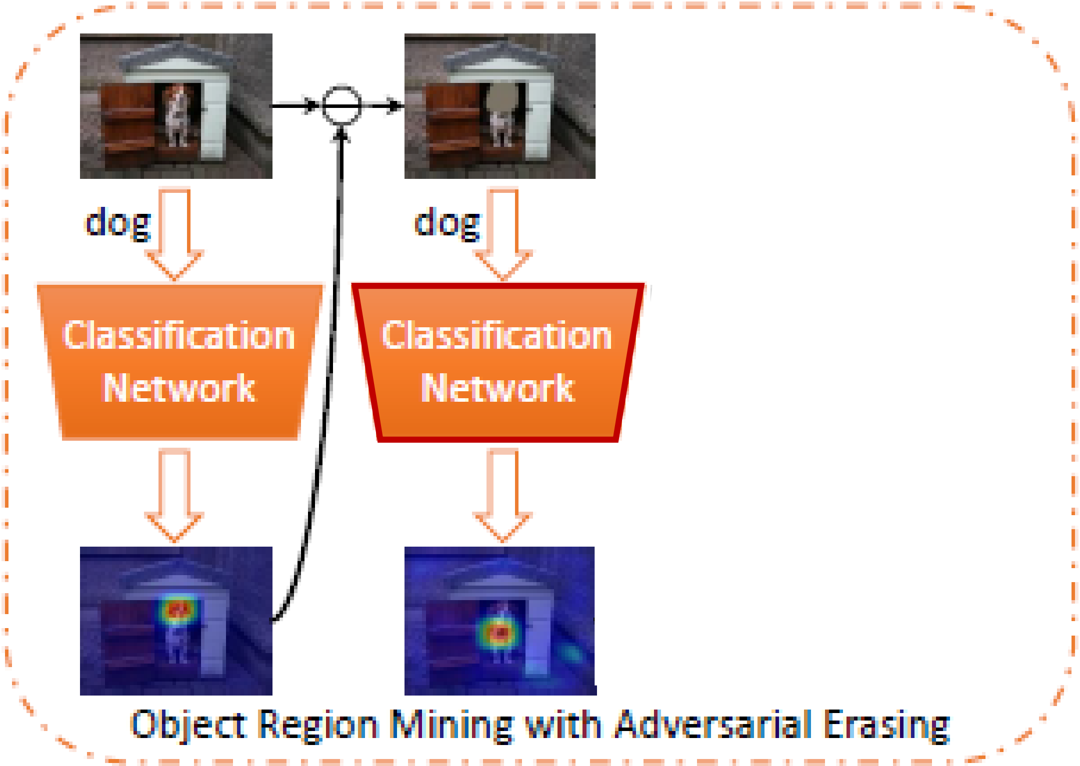


1. 입력 이미지를 Classification Network1을 통해 학습하고 CAM을 추출

2. 1에서 만든 캠의 결과를 제거 (제거 : 전체 이미지의 평균으로 대체)

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



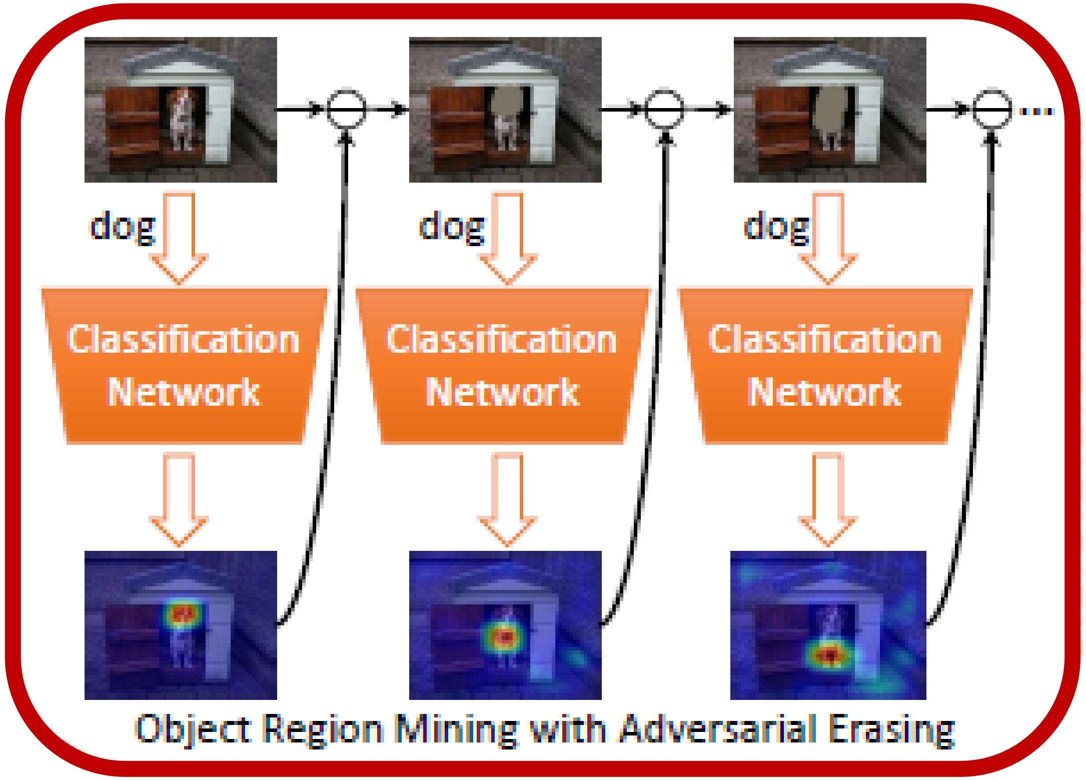
1. 입력 이미지를 Classification Network1을 통해 학습하고 CAM을 추출

2. 1에서 만든 캠의 결과를 제거 (제거 : 전체 이미지의 평균으로 대체)

3. 1번의 네트워크와 **독립**인 새로운 네트워크 Classification Network2로 학습을 진행해서 CAM 결과를 생성

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



1. 입력 이미지를 Classification Network1을 통해 학습하고 CAM을 추출

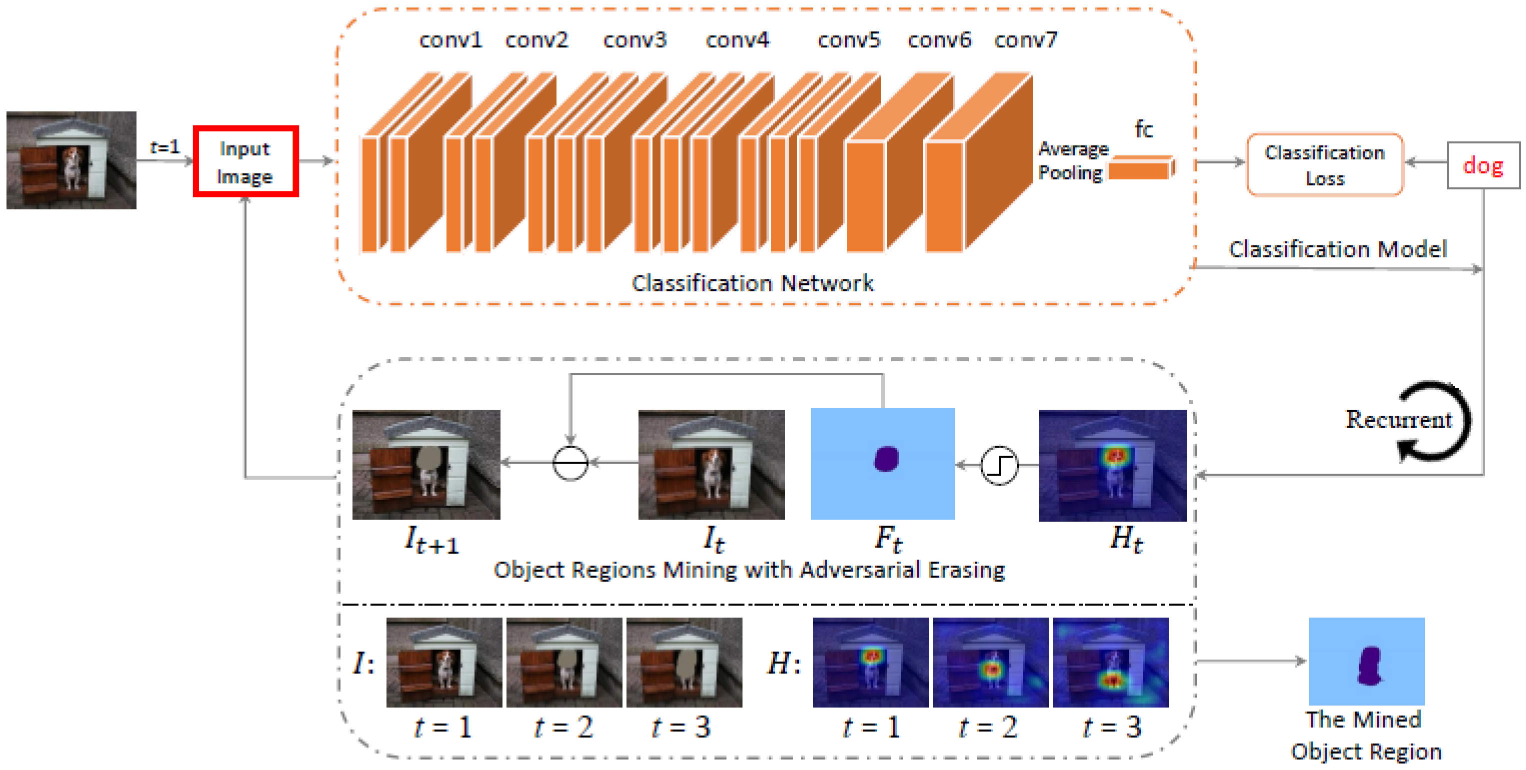
2. 1에서 만든 캠의 결과를 제거 (제거 : 전체 이미지의 평균으로 대체)

3. 1번의 네트워크와 **독립**인 새로운 네트워크 Classification Network2로 학습을 진행해서 CAM 결과를 생성

4. 학습이 완전히 끝날때까지 위의 과정을 반복

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: $\boxed{\text{Training data } \mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N},}$ threshold δ.

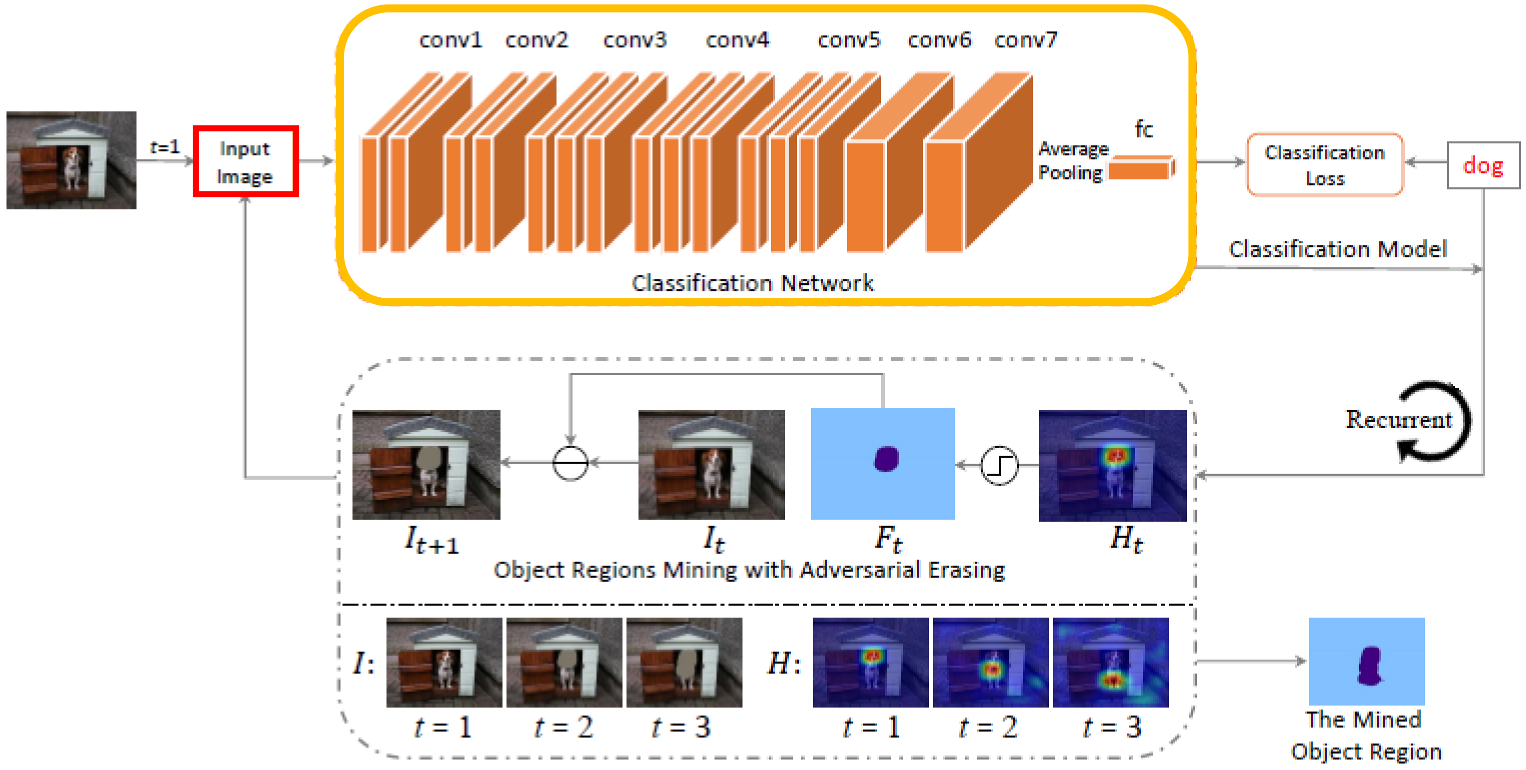
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

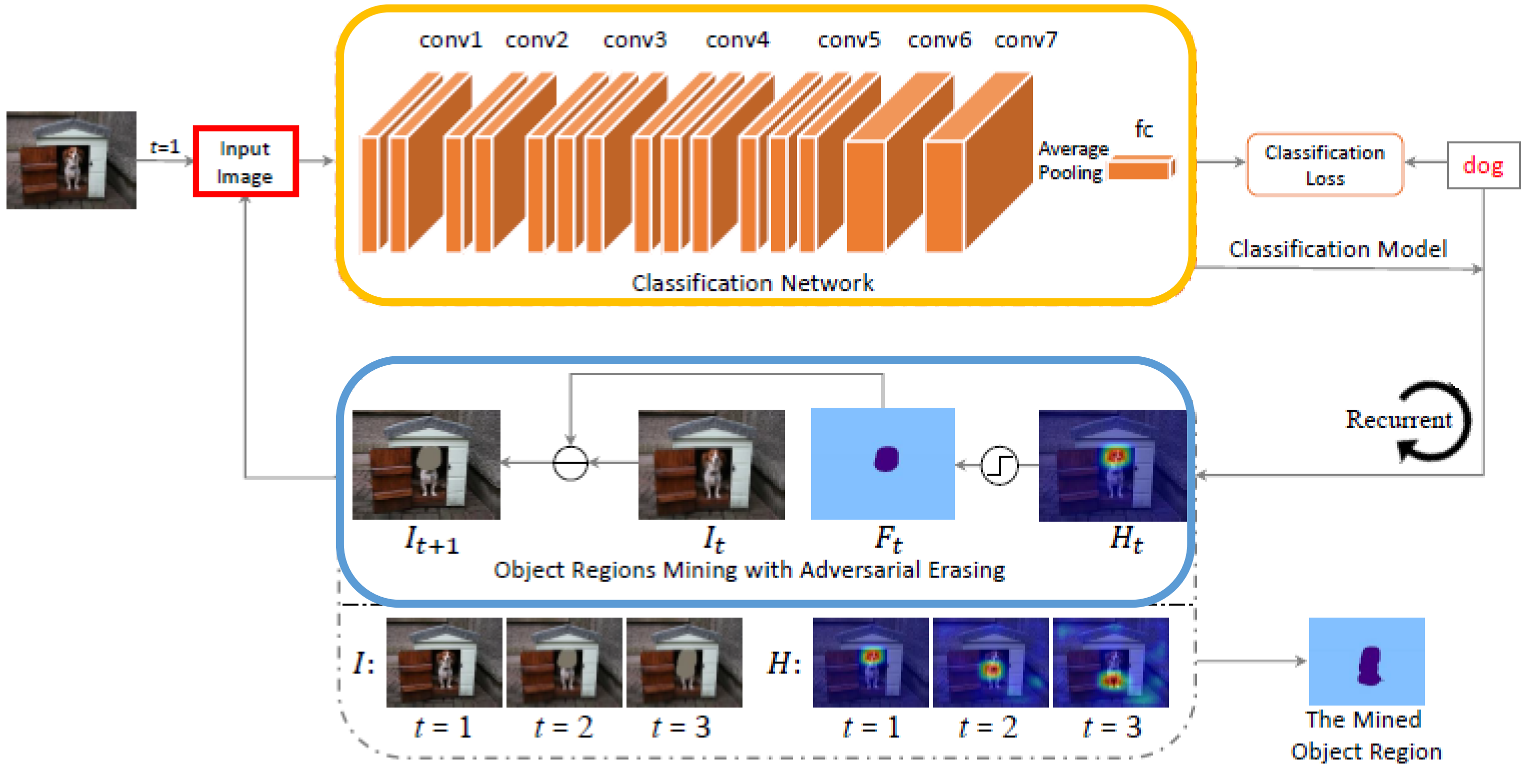
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

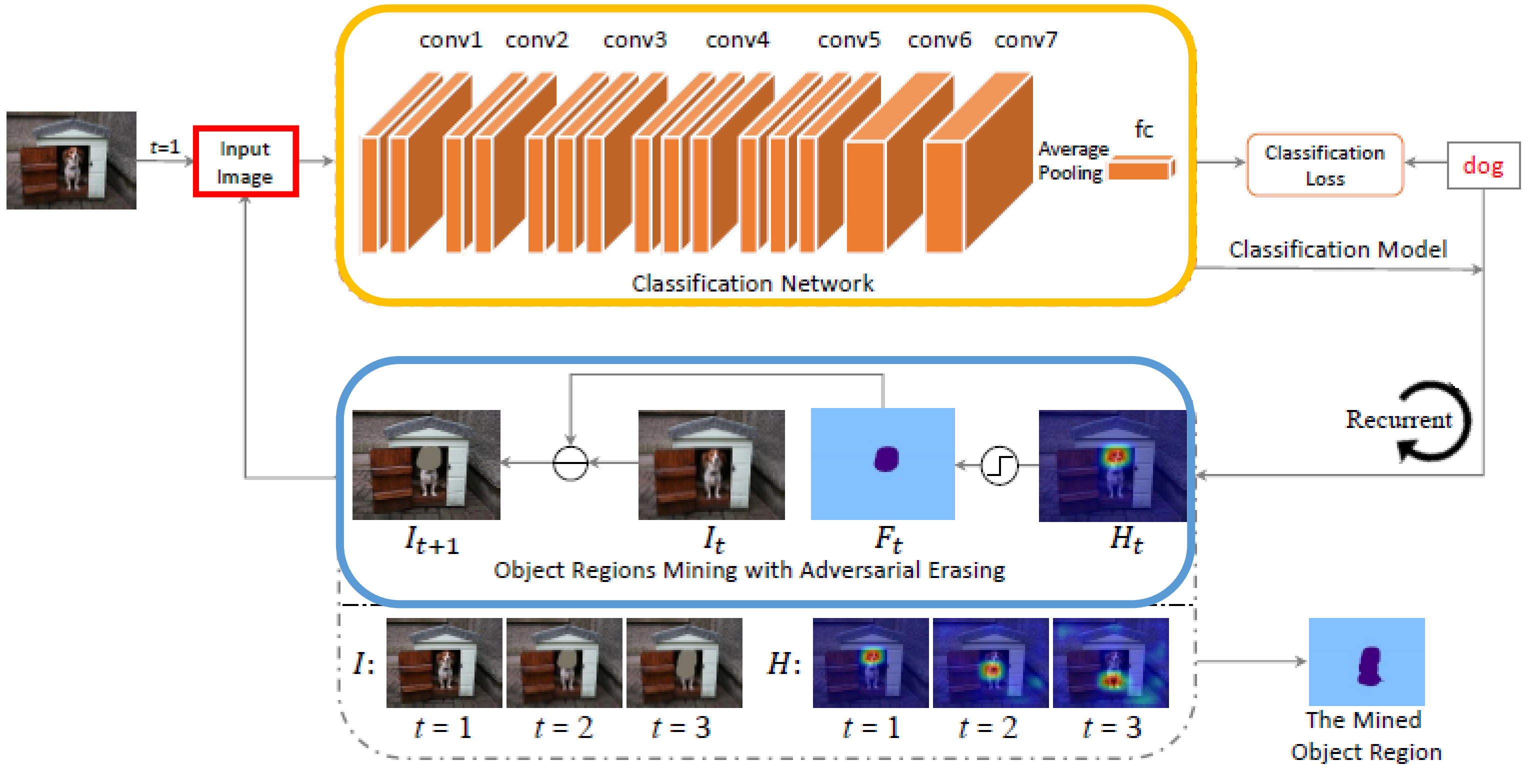
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

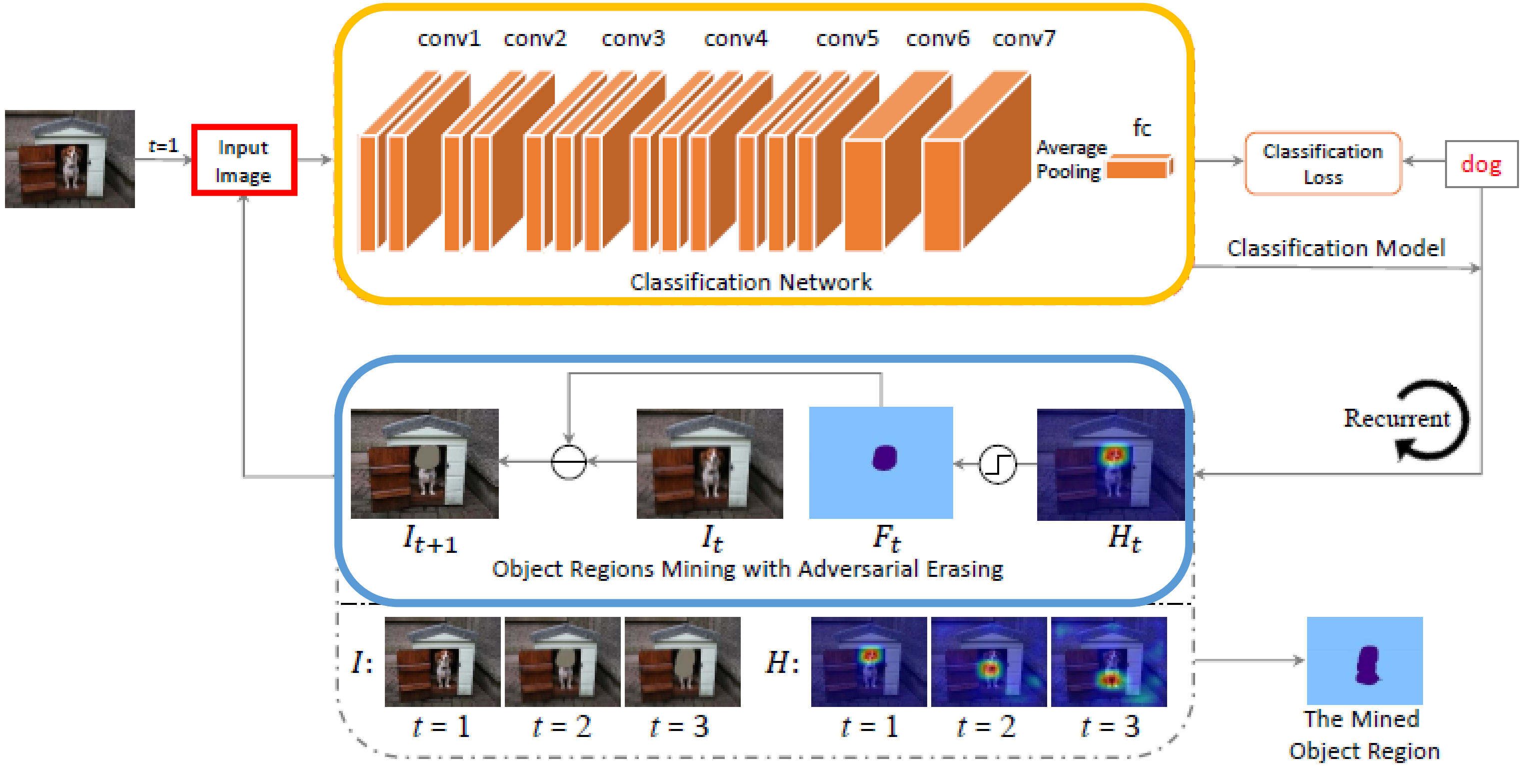
while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

$F_{i,t}$: t번째 학습단계에서 이미지 i에 대한 Object를 담아두는 변수

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

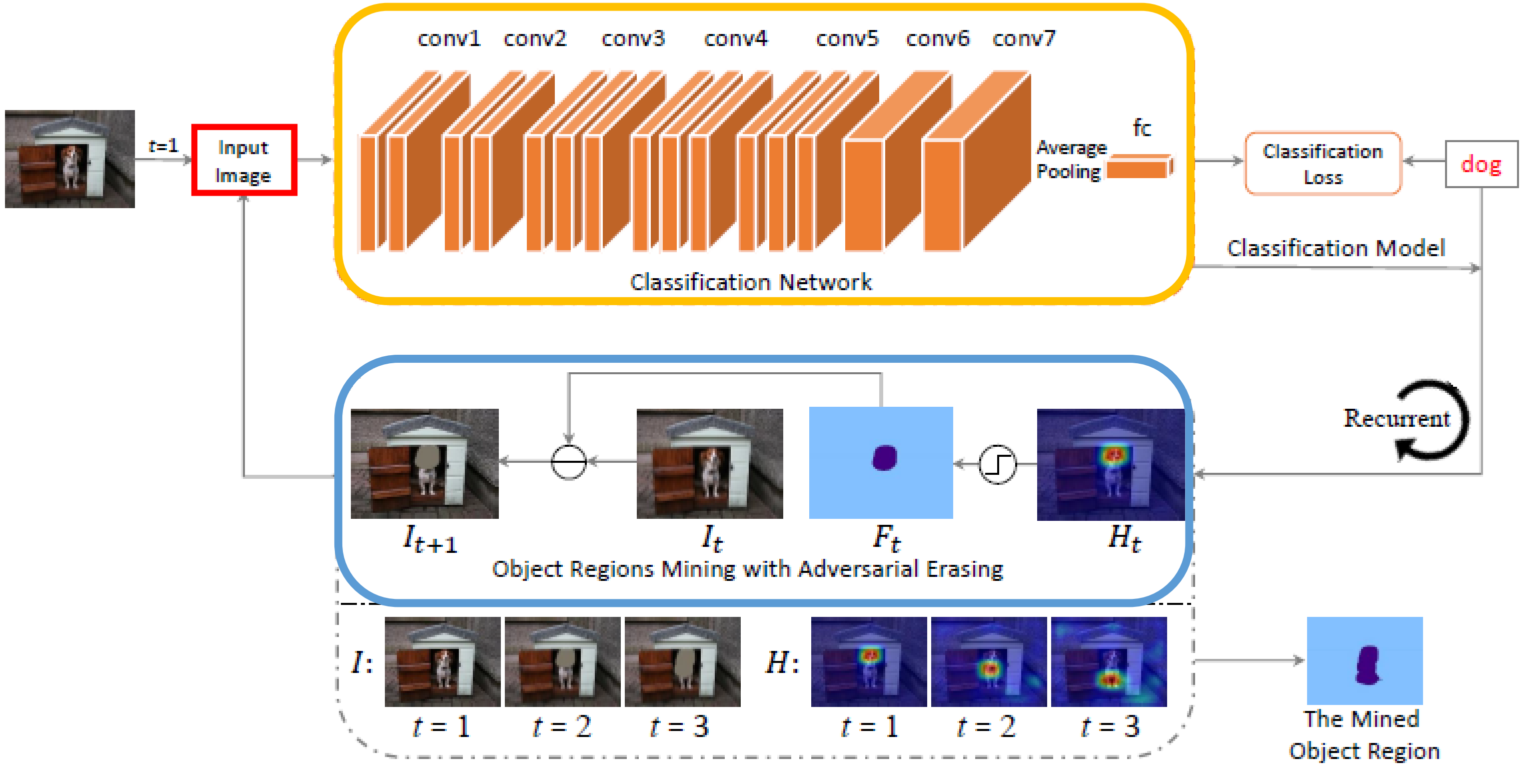
Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

$H_{i,t}^{c}$: t번째 학습에서 i번째 이미지에 대한 클래스 c의 CAM 결과

해당 이미지의 경우 강아지에 대한 클래스(c)만 있어서
For 문에서 강아지 클래스에 대한 CAM 결과만 생성됨

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

Initialize: $F_i = \emptyset(i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \emptyset$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H^c_{i,t}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H^c_{i,t}$ are larger than δ.
      Update the mined regions $F^c_{i,t} = F^c_{i,t} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

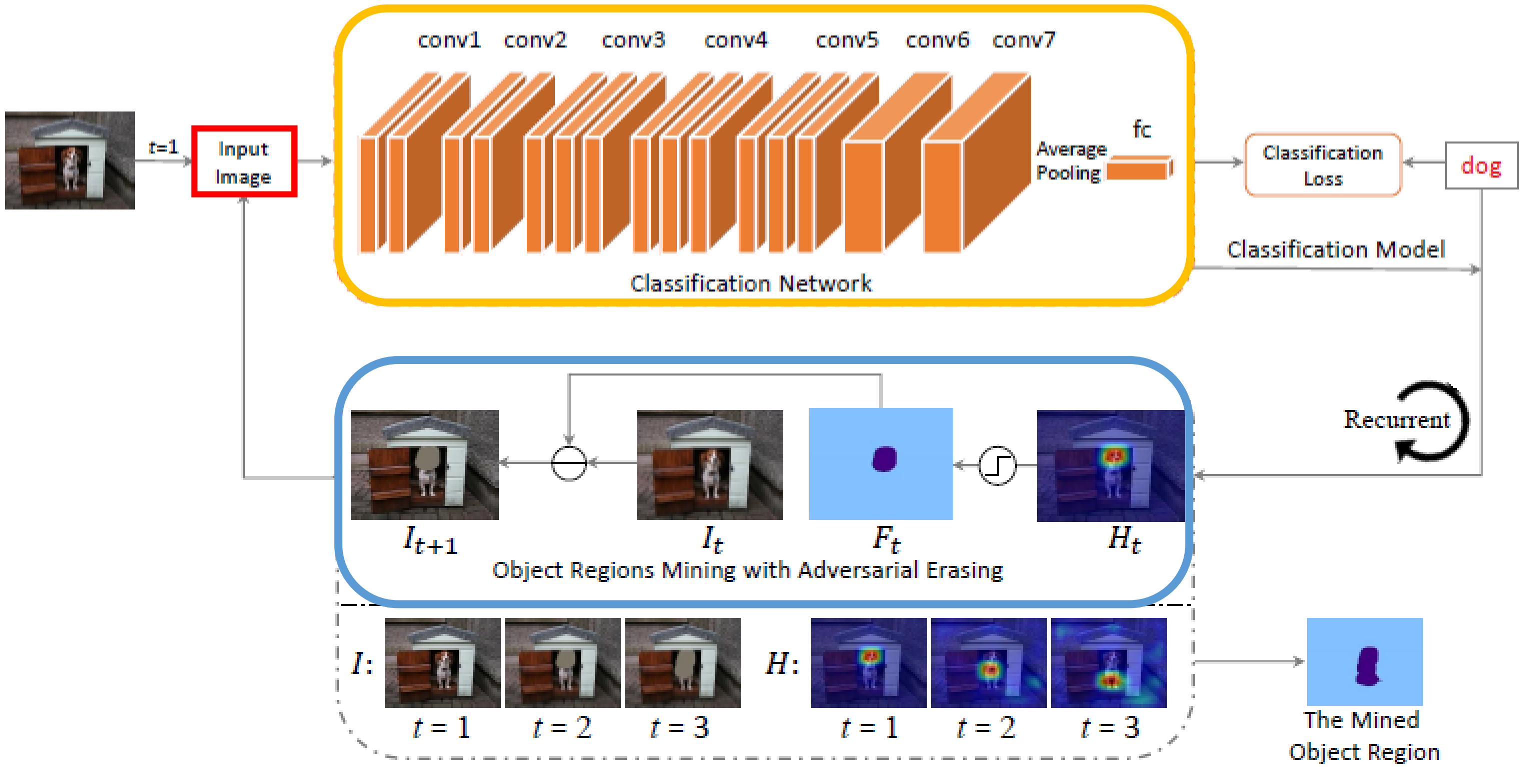
$H^c_{i,t}$: t번째 학습에서 i번째 이미지에 대한 클래스 c의 CAM 결과

해당 이미지의 경우 강아지에 대한 클래스(c)만 있어서 For 문에서 강아지 클래스에 대한 CAM 결과만 생성됨

이후, 임의의 threshold인 δ 보다 작은 값을 가지는 영역은 모두 제거해서 중요한 부분만 추출 (R)

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while

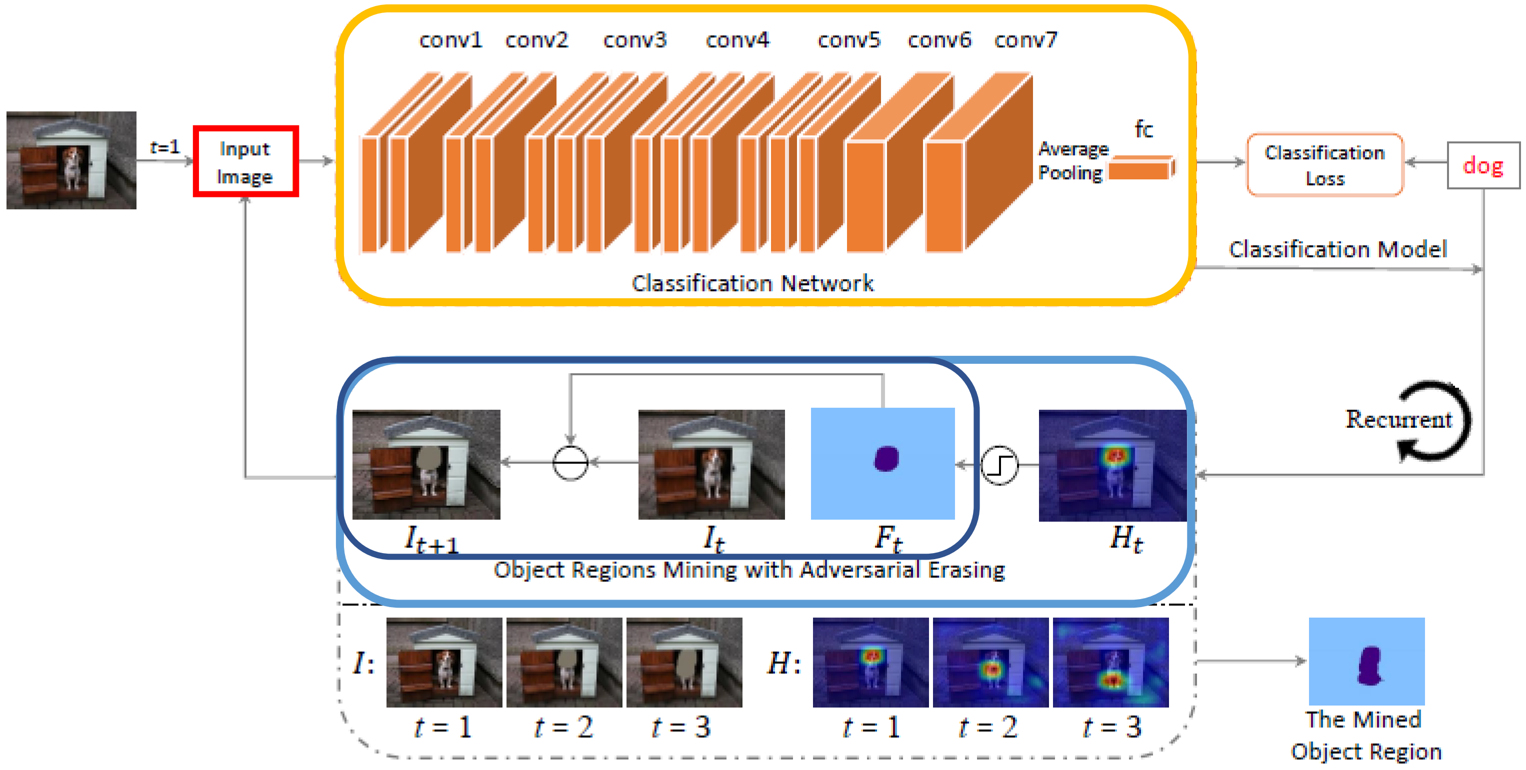
Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

$H_{i,t}^{c}$: t번째 학습에서 i번째 이미지에 대한 클래스 c의 CAM 결과

$F_{i,t}^{c}$ $\cup$ R $=$ $F_{i,t}^{c}$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.

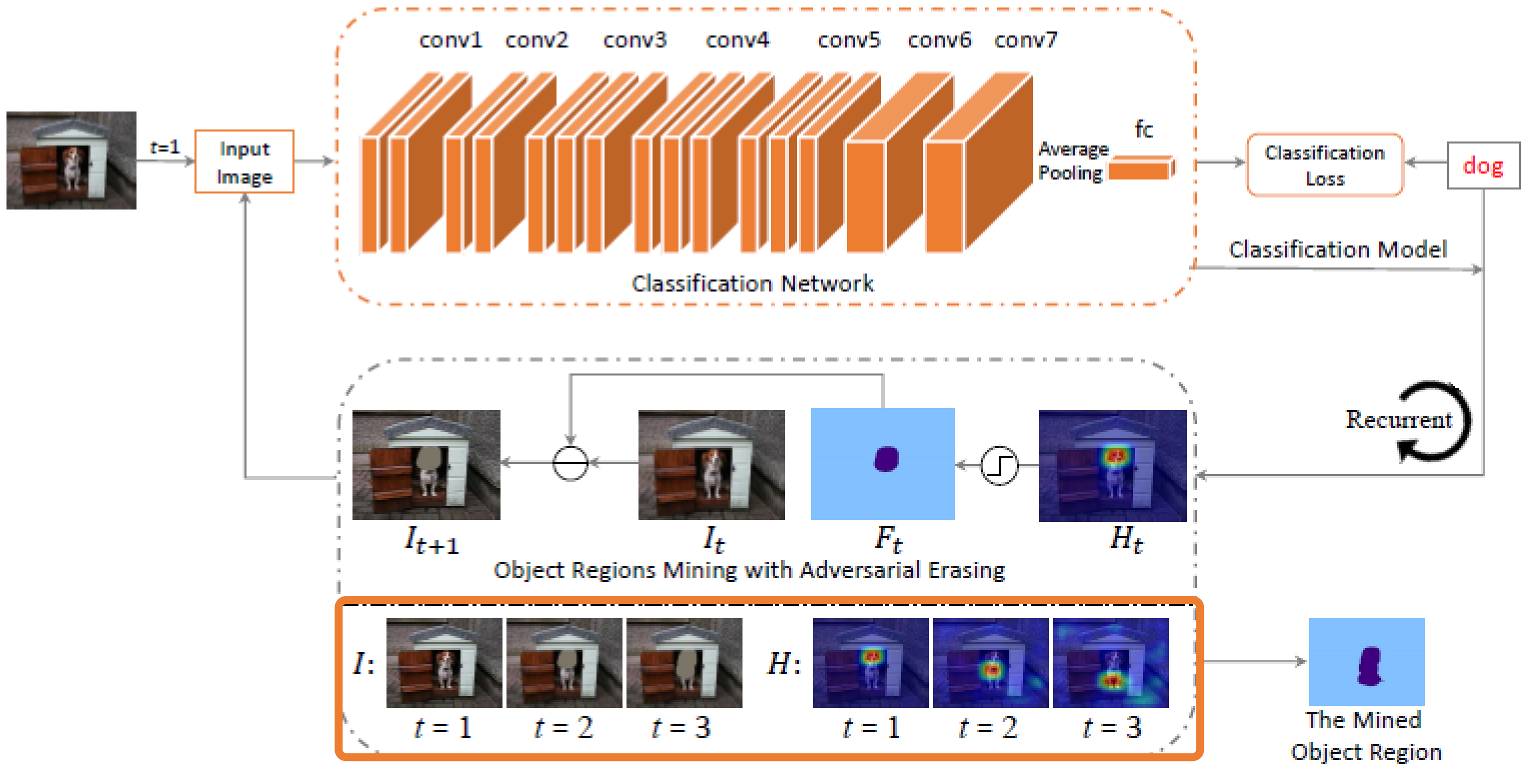
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

```
while (training of classification is success) do
    Train the classification network M_t with I.
    for I_i in I do
        Set F_{i,t} = ∅.
        for c in O_i do
            Calculate H^c_{i,t} by CAM(I_{i,t}, M_t, c) [34].
            Extract regions R whose corresponding pixel
            values in H^c_{i,t} are larger than δ.
            Update the mined regions F^c_{i,t} = F^c_{i,t} ∪ R.
        end for
        Update the mined regions F_i = F_i ∪ F_{i,t}.
        Erase the mined regions from training image
        I_{i,t+1} = I_{i,t} \ F_{i,t}.
    end for
    t = t + 1.
end while
```

Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^N$, threshold δ.
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.

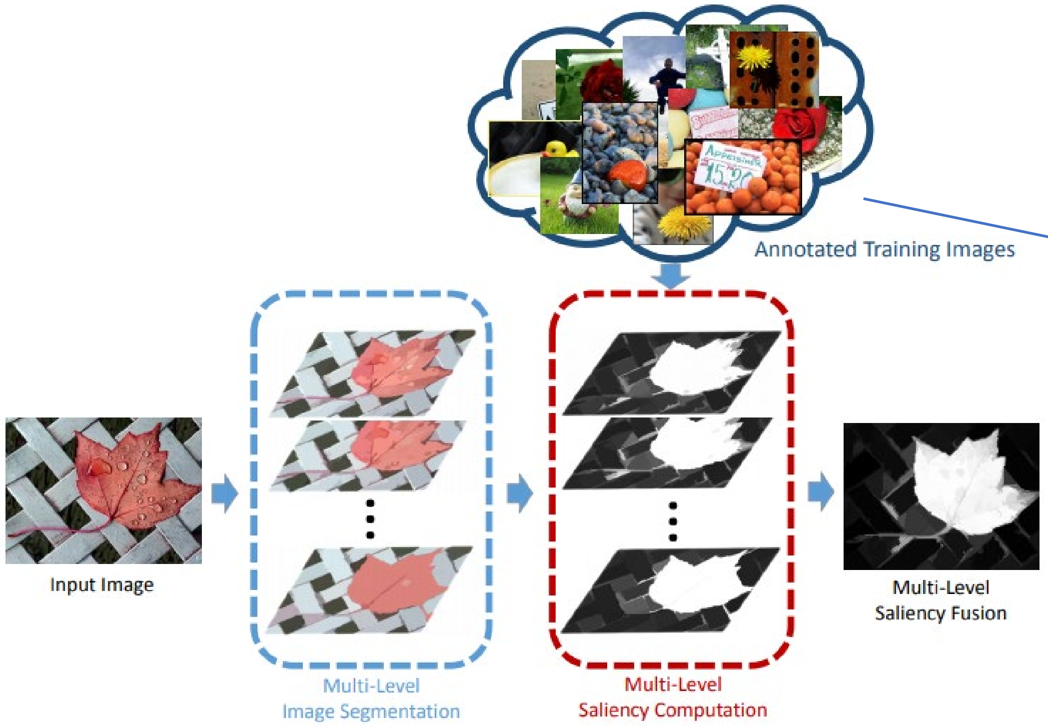
```
while (training of classification is success) do
    Train the classification network M_t with I.
    for I_i in I do
        Set F_{i,t} = ∅.
        for c in O_i do
            Calculate H^c_{i,t} by CAM(I_{i,t}, M_t, c) [34].
            Extract regions R whose corresponding pixel
            values in H^c_{i,t} are larger than δ.
            Update the mined regions F^c_{i,t} = F^c_{i,t} ∪ R.
        end for
        Update the mined regions F_i = F_i ∪ F_{i,t}.
        Erase the mined regions from training image
        I_{i,t+1} = I_{i,t} \ F_{i,t}.
    end for
    t = t + 1.
end while
```

Output: $\mathcal{F} = \{F_i\}_{i=1}^N$

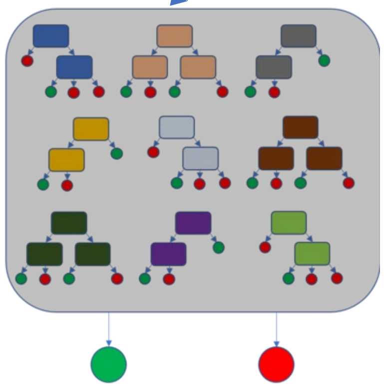
출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

Object의 위치는 이제 CAM을 통해서 추출하는 것은 알겠는데, 배경은 그러면 어떻게 해결하지…?



Color and texture features			Differences of features		Contrast	Backgroundness
	features	dim	definition	dim		
$\mathbf{a}_1$	average RGB values	3	$d(\mathbf{a}_1^{R_i}, \mathbf{a}_1^S)$	3	$c_1 \sim c_3$	$b_1 \sim b_3$
$\mathbf{h}_1$	RGB histogram	256	$\chi^2(\mathbf{h}_1^{R_i}, \mathbf{h}_1^S)$	1	c_4	b_4
$\mathbf{a}_2$	average HSV values	3	$d(\mathbf{a}_2^{R_i}, \mathbf{a}_2^S)$	3	$c_5 \sim c_7$	$b_5 \sim b_7$
$\mathbf{h}_2$	HSV histogram	256	$\chi^2(\mathbf{h}_2^{R_i}, \mathbf{h}_2^S)$	1	c_8	b_8
$\mathbf{a}_3$	average L*a*b* values	3	$d(\mathbf{a}_3^{R_i}, \mathbf{a}_3^S)$	3	$c_9 \sim c_{11}$	$b_9 \sim b_{11}$
$\mathbf{h}_3$	L*a*b* histogram	256	$\chi^2(\mathbf{h}_3^{R_i}, \mathbf{h}_3^S)$	1	c_{12}	b_{12}
$\mathbf{r}$	absolute response of LM filters	15	$d(\mathbf{r}^{R_i}, \mathbf{r}^S)$	15	$c_{13} \sim c_{27}$	$b_{13} \sim b_{27}$
$\mathbf{h}_4$	max response histogram of the LM filters	15	$\chi^2(\mathbf{h}_4^{R_i}, \mathbf{h}_4^S)$	1	c_{28}	b_{28}
$\mathbf{h}_5$	histogram of the LBP feature	256	$\chi^2(\mathbf{h}_4^{R_i}, \mathbf{h}_5^S)$	1	c_{29}	b_{29}



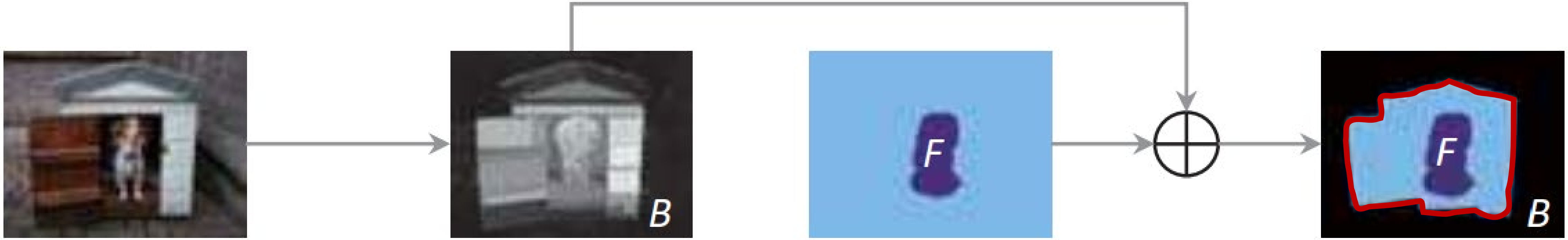
Random Forest

P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graphbased image segmentation," IJCV, vol. 59, no. 2, 2004.

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

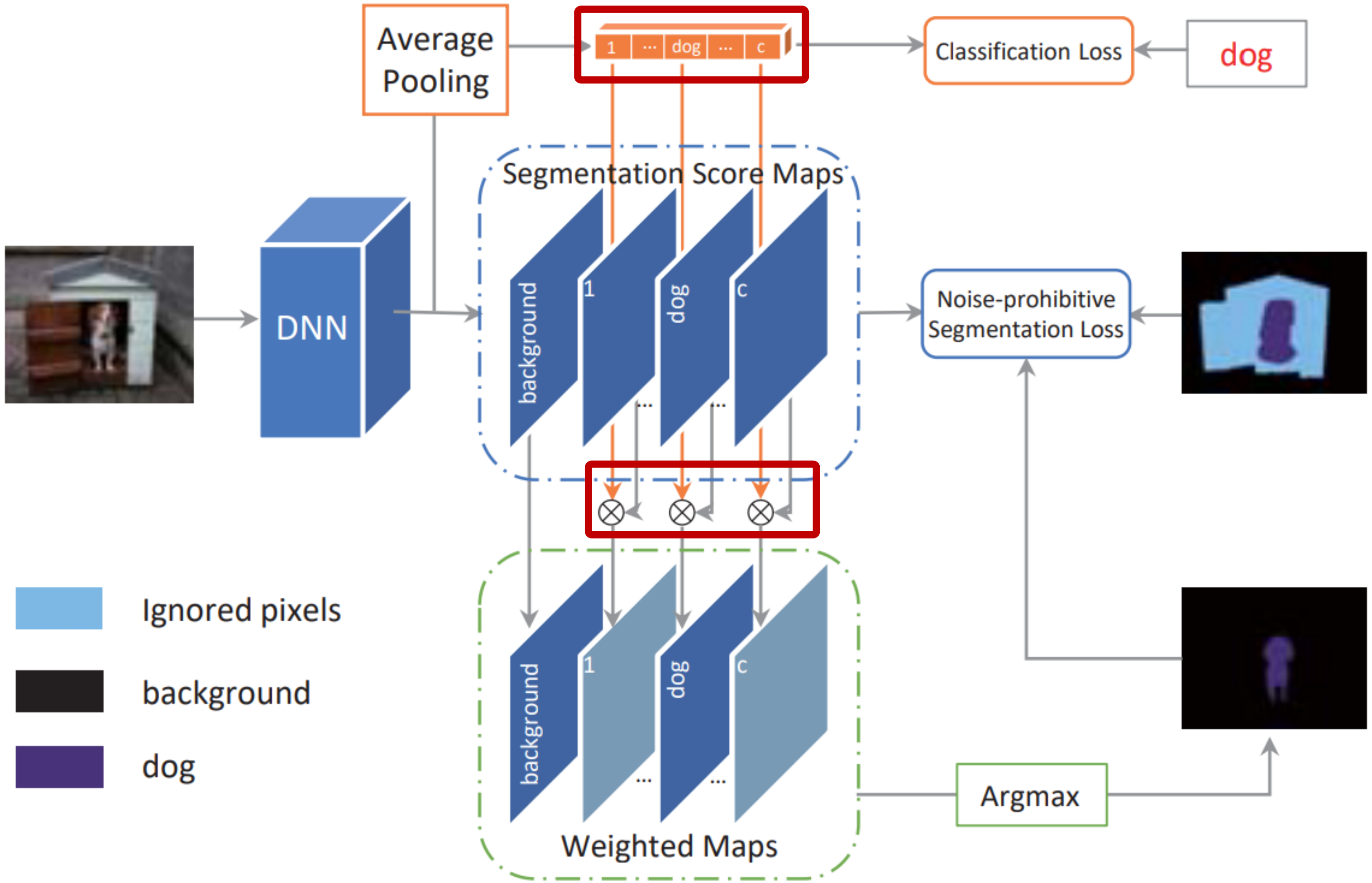
하늘색 강아지 집의 경우처럼 잘못 분류된 경우에 대해서 어떻게 해결하지?



출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

Classification의 Probability를 가중치로해서 Segmentation Score에 Weight를 주자 !!!

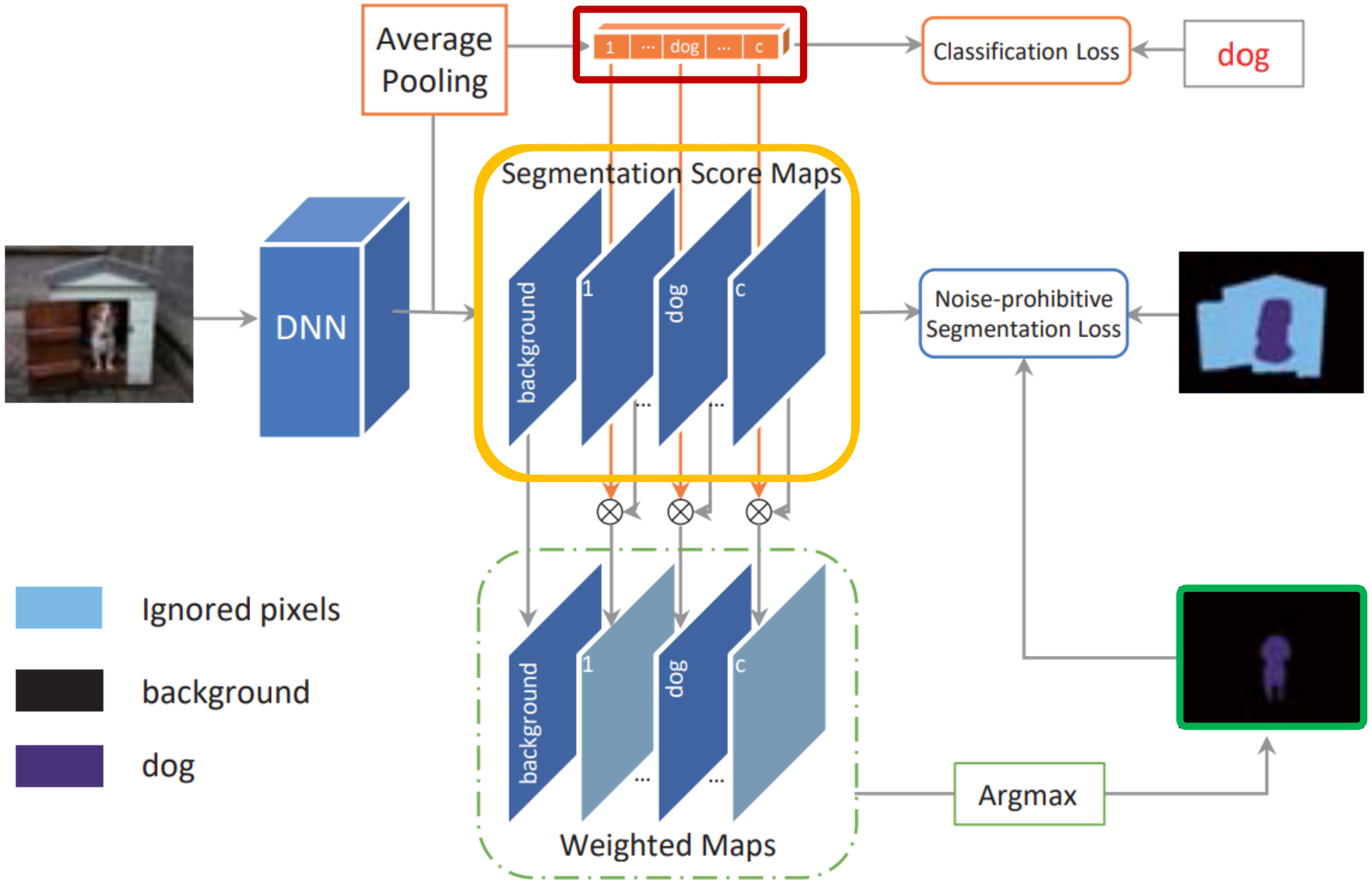


출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

Classification의 Probability를 가중치로해서 Segmentation Score에 Weight를 주자 !!!

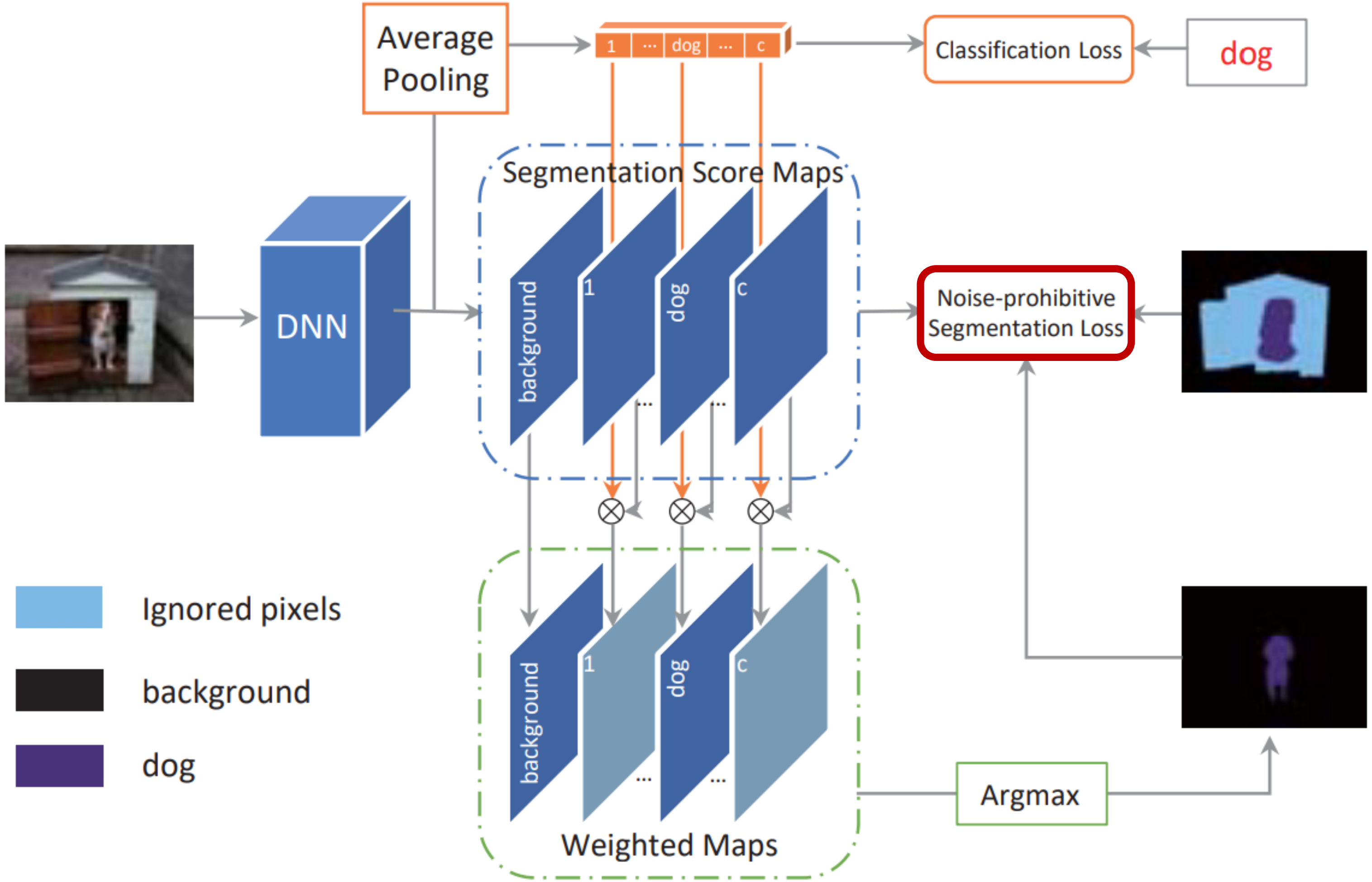
$$\boxed{\hat{S}} = \max\{\boxed{[1, v]} \otimes \boxed{f(I;\theta)}\}$$



출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

Classification의 Probability를 가중치로해서 Segmentation Score에 Weight를 주자 !!!

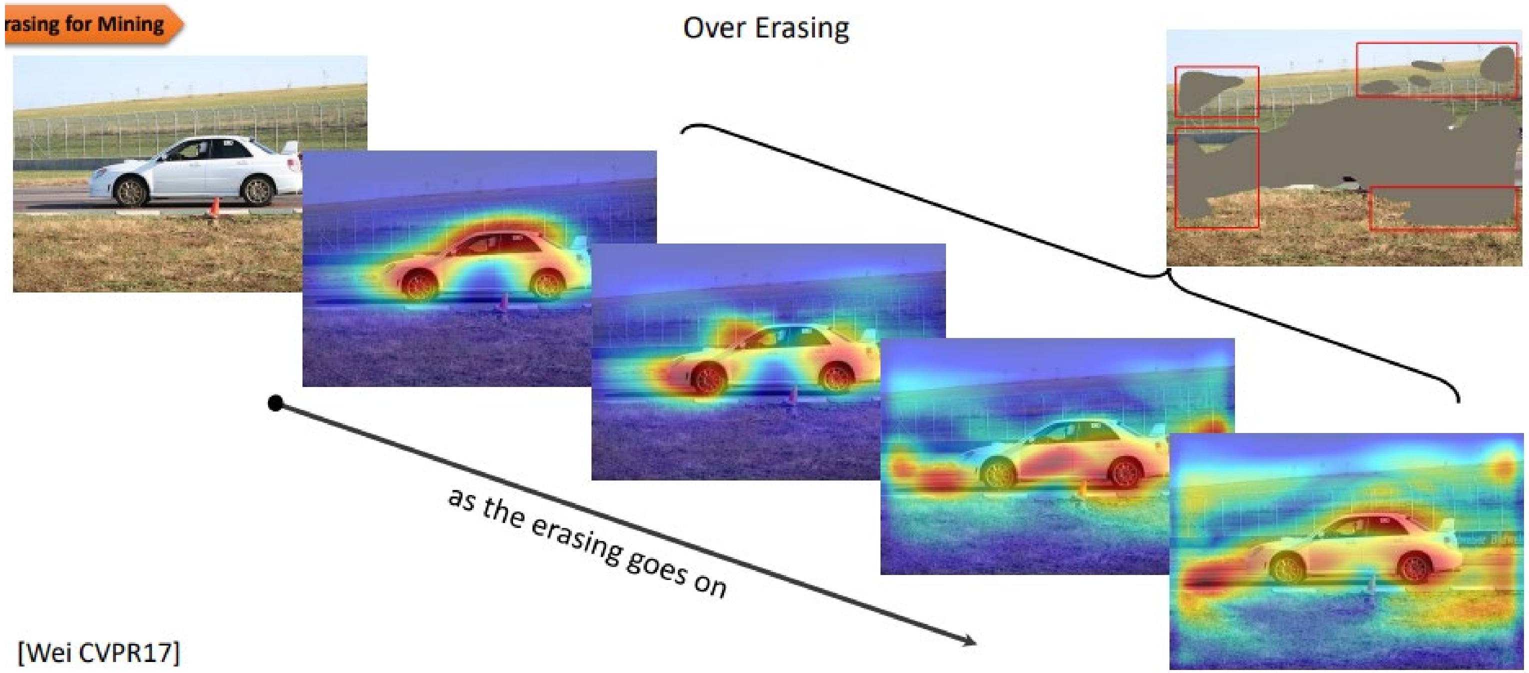


$$\hat{S} = \max\{[1, \boldsymbol{v}] \otimes f(I;\theta)\}$$

$$\min_{\theta} \sum_{I \in \mathcal{I}} J(f(I;\theta), S) + J(f(I;\theta), \hat{S})$$

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only



[Wei CVPR17]

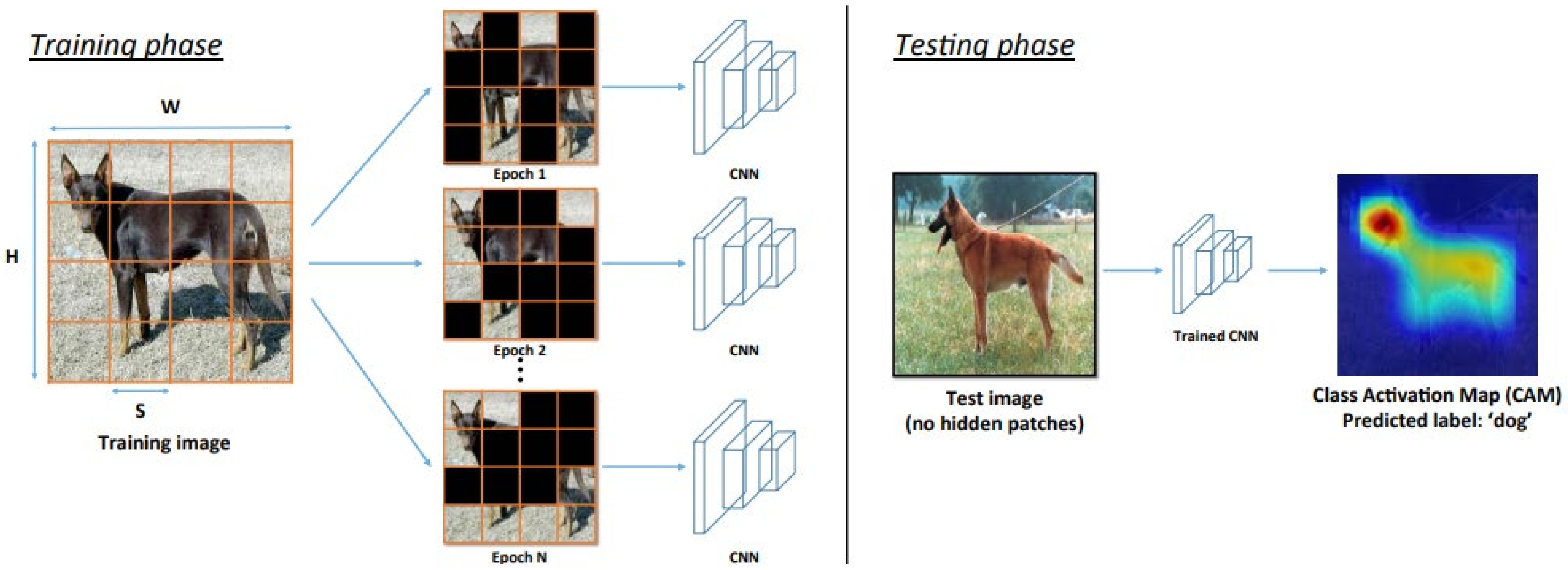
Algorithm 1 Object Regions Mining with AE

Input: Training data $\mathcal{I} = \{(I_i, \mathcal{O}_i)\}_{i=1}^{N}$, threshold δ.
Initialize: $F_i = \varnothing (i = 1, \cdots, N)$, $t = 1$.
while (training of classification is success) **do**
  Train the classification network M_t with $\mathcal{I}$.
  for I_i in $\mathcal{I}$ **do**
    Set $F_{i,t} = \varnothing$.
    for c in $\mathcal{O}_i$ **do**
      Calculate $H_{i,t}^{c}$ by CAM$(I_{i,t}, M_t, c)$ [34].
      Extract regions R whose corresponding pixel values in $H_{i,t}^{c}$ are larger than δ.
      Update the mined regions $F_{i,t}^{c} = F_{i,t}^{c} \cup R$.
    end for
    Update the mined regions $F_i = F_i \cup F_{i,t}$.
    Erase the mined regions from training image $I_{i,t+1} = I_{i,t} \backslash F_{i,t}$.
  end for
  $t = t + 1$.
end while
Output: $\mathcal{F} = \{F_i\}_{i=1}^{N}$

[문제]
1. 독립인 네트워크를 여러개 학습해야함
2. 언제까지 네트워크를 학습해야하는지에 대한 기준이 애매함 -> 그로인해 Over Erasing 문제 발생

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

4 Focused on Discriminative Area Only

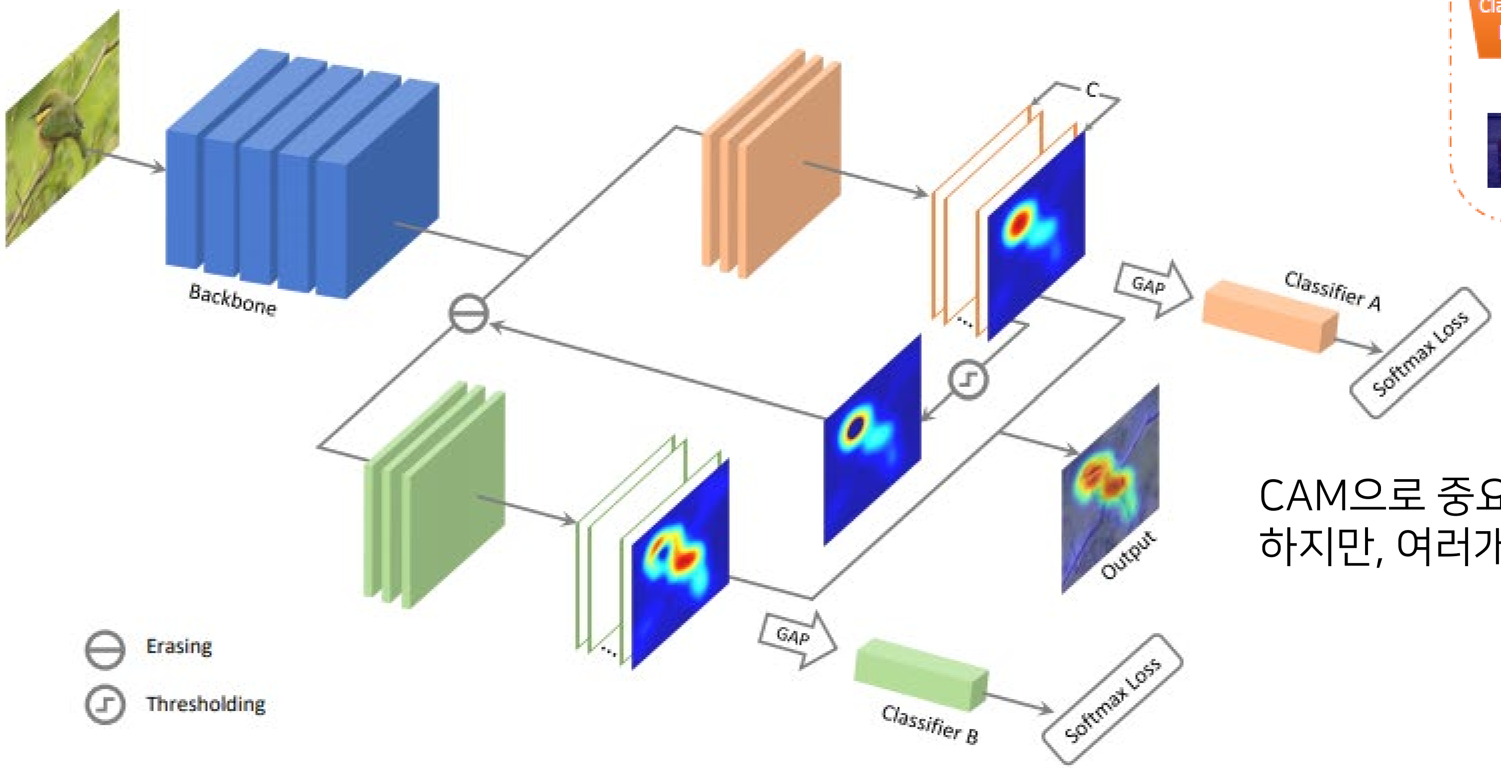


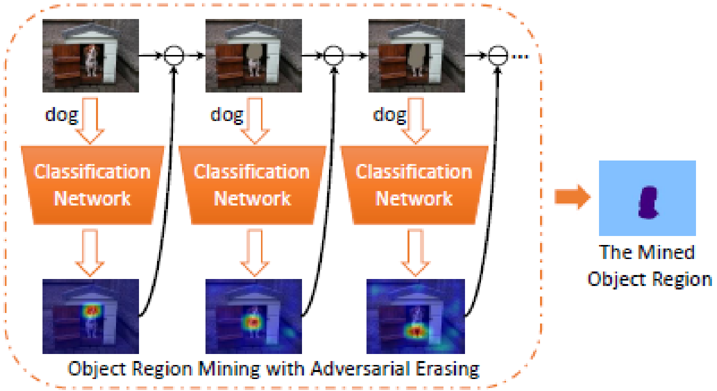
1. 입력 이미지에서 Random 하게 패치를 숨기는 전략을 사용
2. 그로인해 한개의 네트워크로도 위의 논문과 비슷한 효과를 얻을 수 있음

-> 하지만, 랜덤하게 입력이미지를 가리는게 전체 객체 영역을 CAM으로 탐지한다는 보장을 못하는게 한계

출처 : Singh & Lee Hide-and-Seek, 2017

4 Focused on Discriminative Area Only





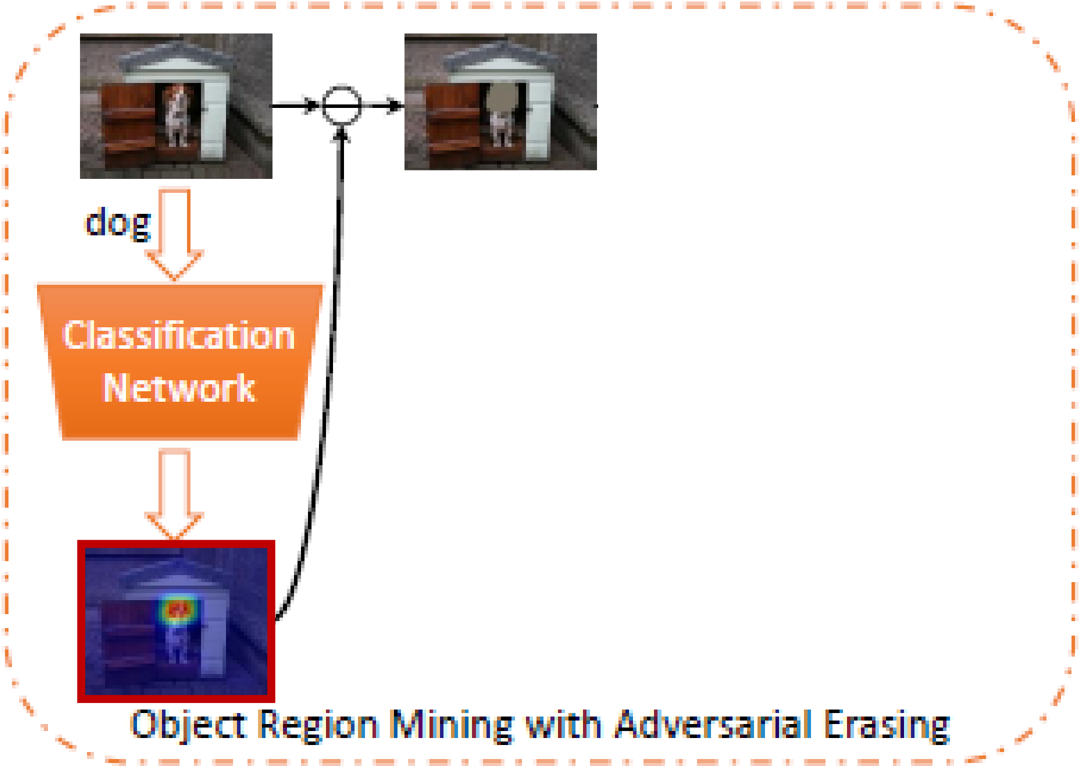
CAM으로 중요한 영역을 지우는 것에 대해서는 동의.
하지만, 여러개의 네트워크는 제거!

하나의 네트워크에서 Classifier 부분만 2개를 두는 전략을 사용

[AE-PSL] Object Region Mining With Adversarial...	Weakly-supervised Learning	Adversarial Erasing	Proposed
[ACoL] Adversarial Complementary Learning for...	Weakly-supervised Learning	Adversarial Erasing	Proposed

출처 : Adversarial Complementary Learning for Weakly Supervised Object Localization, Xiaolin Zhang et al., 2018

5 Discussion



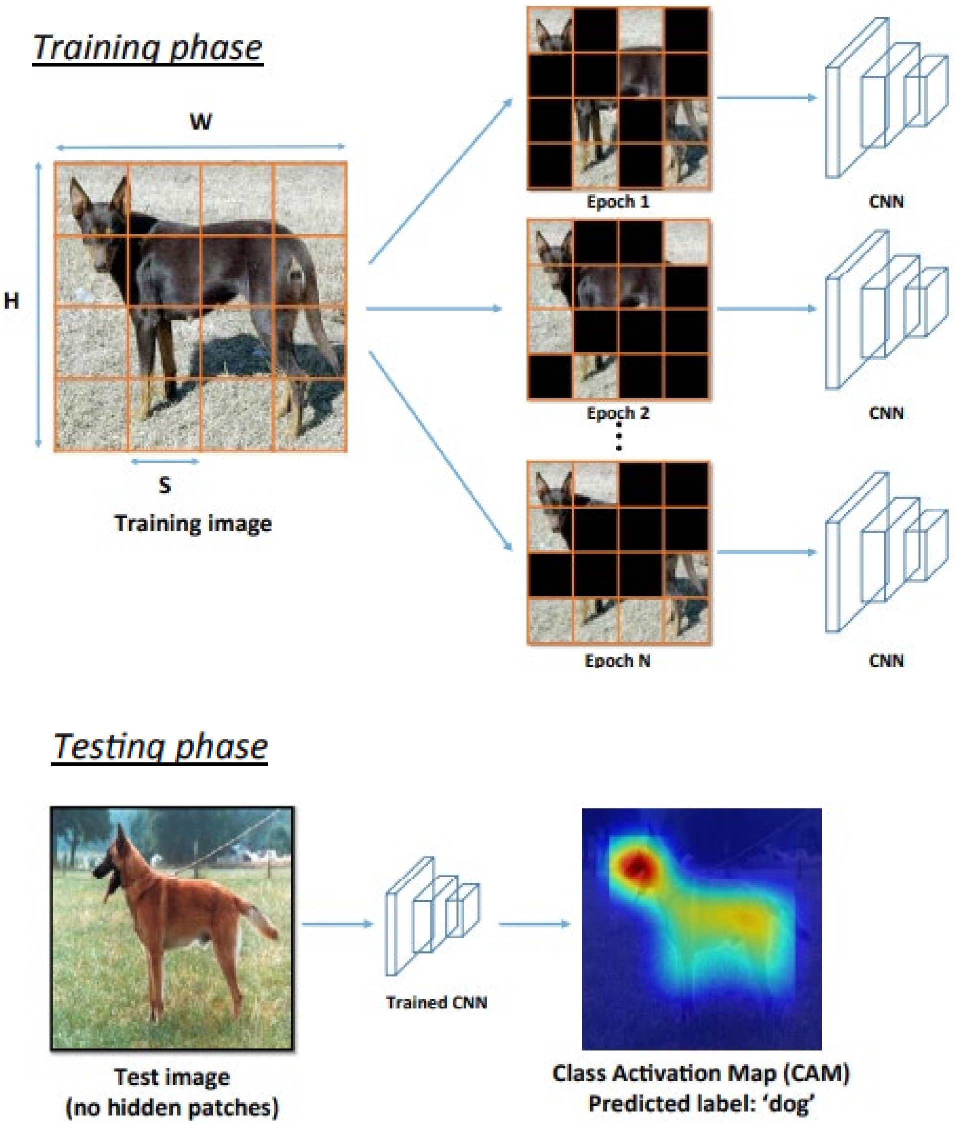
1. 입력 이미지를 Classification Network1을 통해 학습 하고 CAM을 추출

2. 1에서 만든 캠의 결과를 제거
   (제거 : 전체 이미지의 평균으로 대체)

   왜? 굳이 평균으로 해당 부분을 대체해야할까??

출처 : Object Region Mining with Adversarial Erasing_ A Simple Classification to Semantic Segmentation Approach, Yunchao Wei et al., 2017

5 Discussion

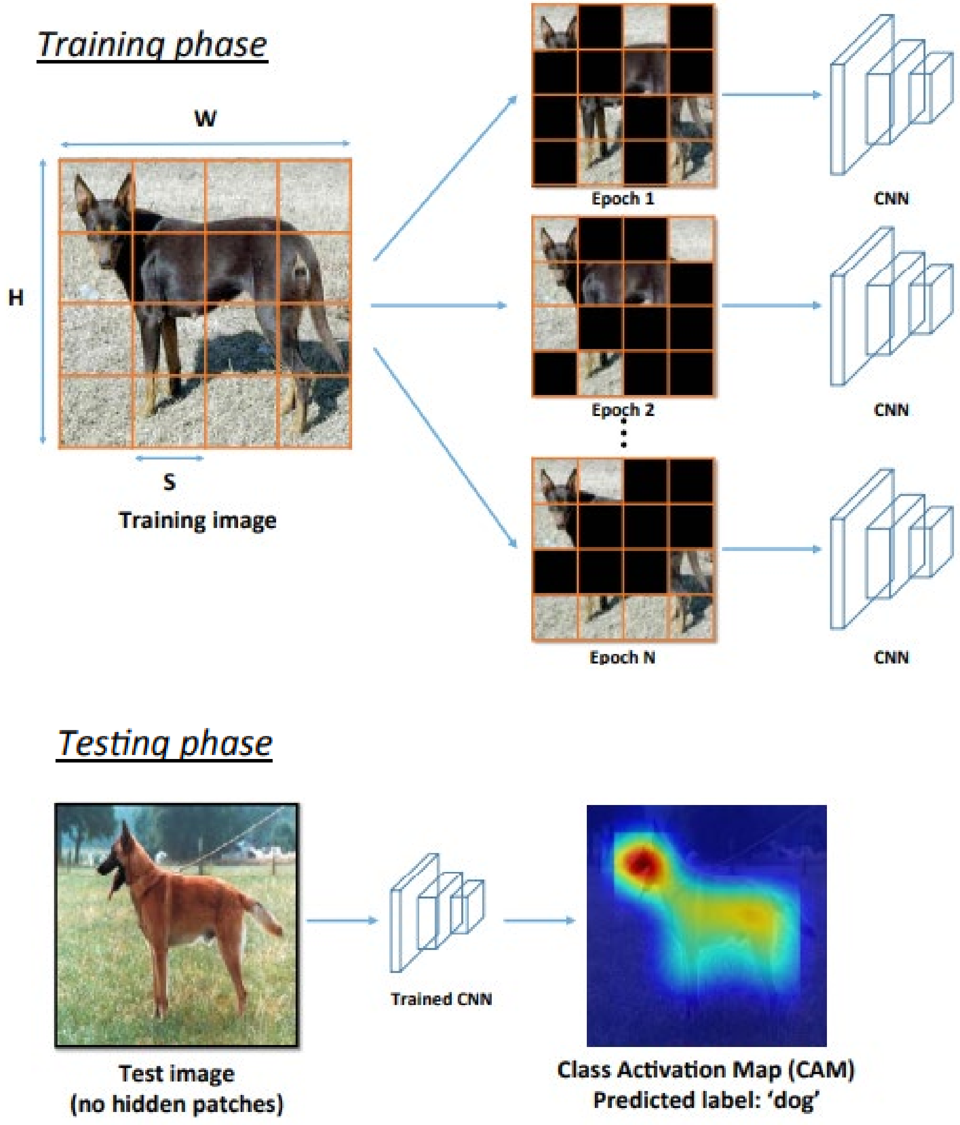


During testing, F will always be completely within a visible patch, and thus its output will be $\sum_{i=1}^{k \times k} \mathbf{w}_i^\top \mathbf{x}_i$. This matches the expected output during training in only the first case. For the remaining two cases, when F is completely or partially within a hidden patch, the activations will have a distribution that is different to those seen during testing.

We resolve this issue by setting the RGB value $\mathbf{v}$ of a hidden pixel to be equal to the mean RGB vector of the images over the entire dataset: $\mathbf{v} = \mu = \frac{1}{N_{pixels}} \sum_j \mathbf{x}_j$, where j indexes all pixels in the entire training dataset and N_{pixels} is the total number of pixels in the dataset. Why would this work? This is because in expectation, the output of a patch will be equal to that of an average-valued patch: $\mathbb{E}[\sum_{i=1}^{k \times k} \mathbf{w}_i^\top \mathbf{x}_i] = \sum_{i=1}^{k \times k} \mathbf{w}_i^\top \mu$. By replacing $\mathbf{v}$ with μ, the outputs of both the second and third cases will be $\sum_{i=1}^{k \times k} \mathbf{w}_i^\top \mu$, and thus will match the expected output during testing (i.e., of a fully-visible patch).[1]

출처 : Singh & Lee Hide-and-Seek, 2017

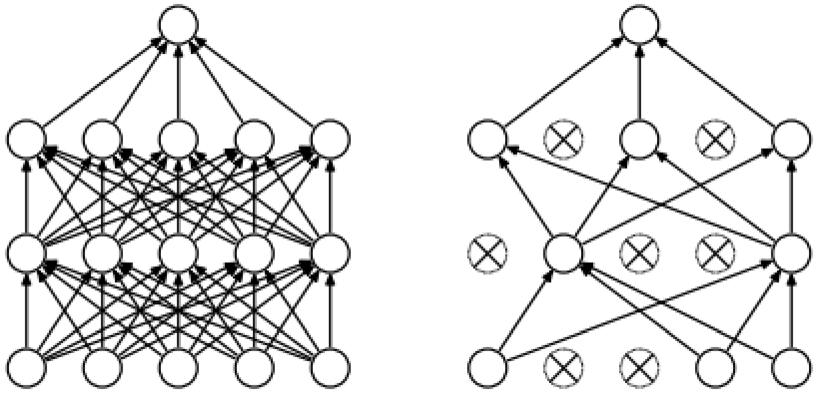
5 Discussion



This process is related to the scaling procedure in dropout [35], in which the outputs are scaled proportionally to the drop rate during testing to match the expected output during training. In dropout, the outputs are dropped uniformly across the entire feature map, independently of spatial location. If we view our hiding of the patches as equivalent to "dropping" units, then in our case, we cannot have a global scale factor since the output of a patch depends on whether there are any hidden pixels. Thus, we instead set the hidden values to be the expected pixel value of the training data as described above, and do not scale the corresponding output. Empirically, we find that setting the hidden pixel in this way is crucial for the network to behave similarly during training and testing.

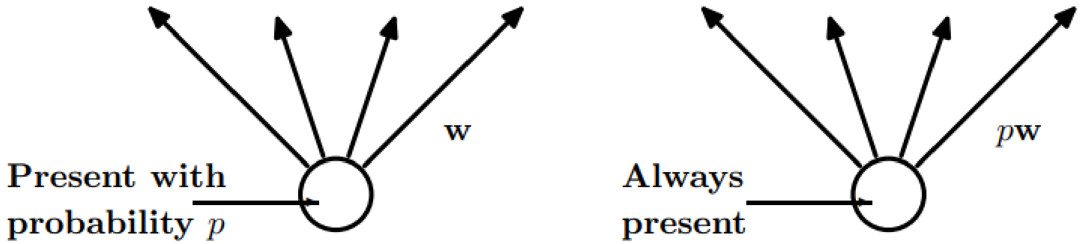
출처 : Singh & Lee Hide-and-Seek, 2017

5 Discussion



(a) Standard Neural Net　(b) After applying dropout.

$$h' = \begin{cases} 0 & \text{확률 } p \text{ 인 경우} \\ \frac{h}{1-p} & \text{그 외의 경우} \end{cases}$$



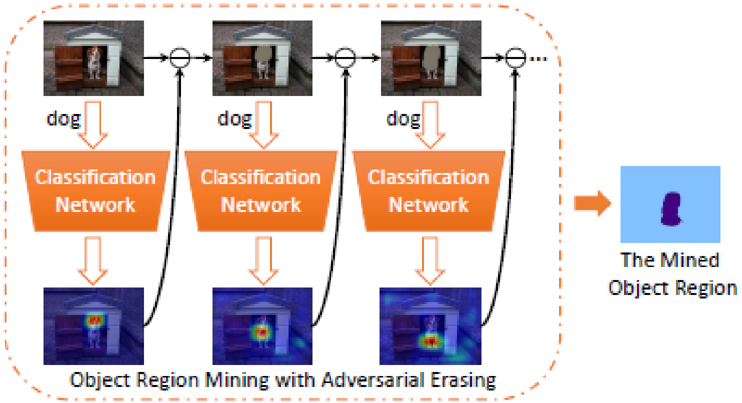
(a) At training time　(b) At test time

Figure 2: **Left**: A unit at training time that is present with probability p and is connected to units in the next layer with weights $\mathbf{w}$. **Right**: At test time, the unit is always present and the weights are multiplied by p. The output at test time is same as the expected output at training time.

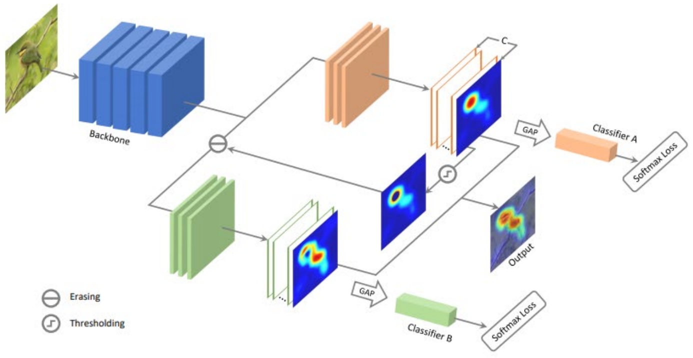
At test time, it is not feasible to explicitly average the predictions from exponentially many thinned models. However, a very simple approximate averaging method works well in practice. The idea is to use a single neural net at test time without dropout. The weights of this network are scaled-down versions of the trained weights. If a unit is retained with probability p during training, the outgoing weights of that unit are multiplied by p at test time as shown in Figure 2. This ensures that for any hidden unit the *expected* output (under the distribution used to drop units at training time) is the same as the actual output at test time. By doing this scaling, 2^n networks with shared weights can be combined into a single neural network to be used at test time. We found that training a network with dropout and using this approximate averaging method at test time leads to significantly lower generalization error on a wide variety of classification problems compared to training with other regularization methods.

출처 : Dropout: A Simple Way to Prevent Neural Networks from Overfitting

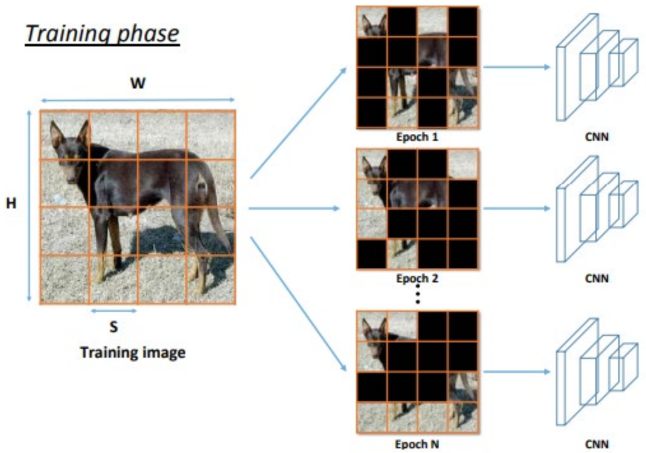
5 Discussion



-> 평균으로 대체



-> 0으로 대체



-> 평균으로 대체

출처 : Dropout: A Simple Way to Prevent Neural Networks from Overfitting