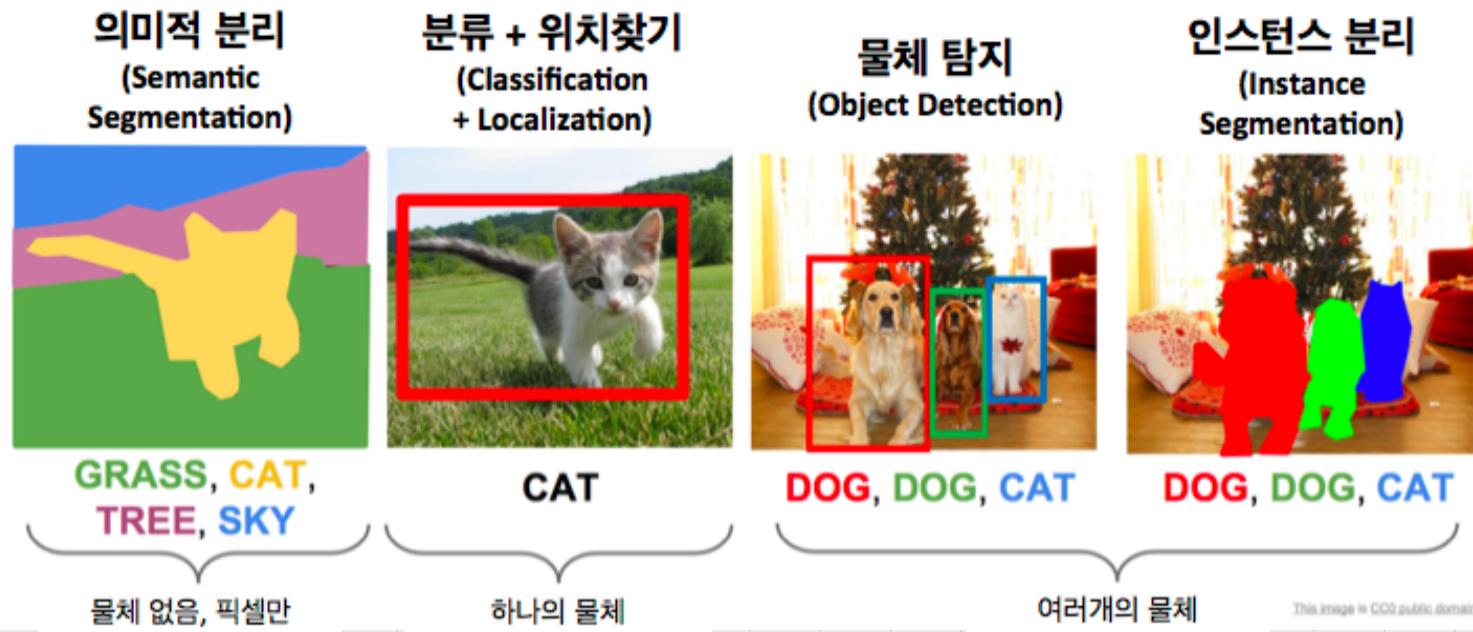


Segmentation (분리)

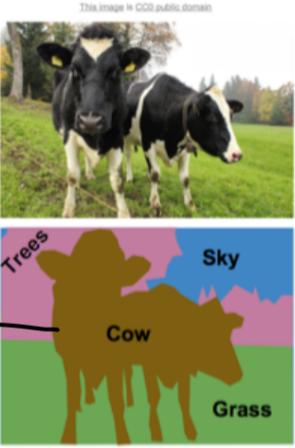
Semantic Segmentation (각각 물체가 어떤 class인지 구분)
Instance Segmentation (같은 클래스 & 인스턴스 구분)

다른 컴퓨터 시각 작업 (Computer Vision Tasks)



I. Semantic Segmentation

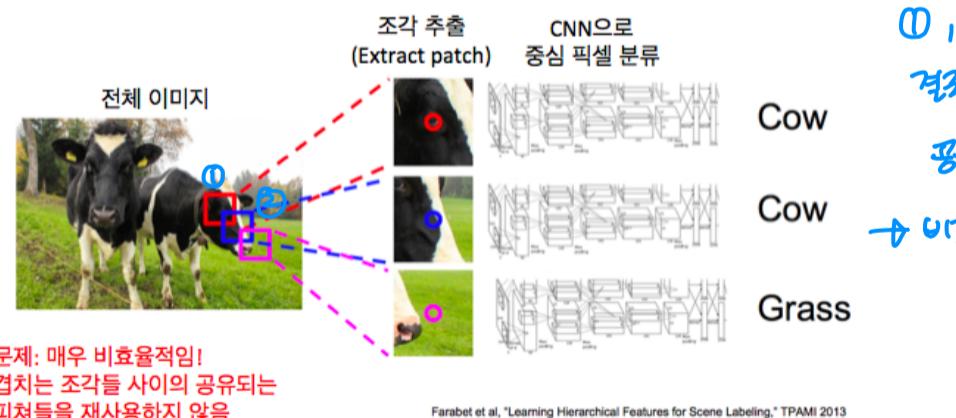
- 이미지 내의 각 픽셀을 차례로 고기 레이블 (category label)을 저장
- 인스턴스 구별 X 픽셀만 선명함.



* channel 개수
= feature map 개수
= filter의 개수
= kernel의 개수

Semantic

의미적 분리 아이디어: 슬라이딩 윈도우 (Sliding Window)



①, ② 번 경계는 복잡
결국 같은 conv 거친...
공유학수 있는 부분 많.
→ 비효율적이라는 뜻.

의미적 분리 아이디어: 슬라이딩 윈도우 (Sliding Window)



Fully Convolutional Network

FC (fully connected) 개념

모든 이미지 픽셀 통과

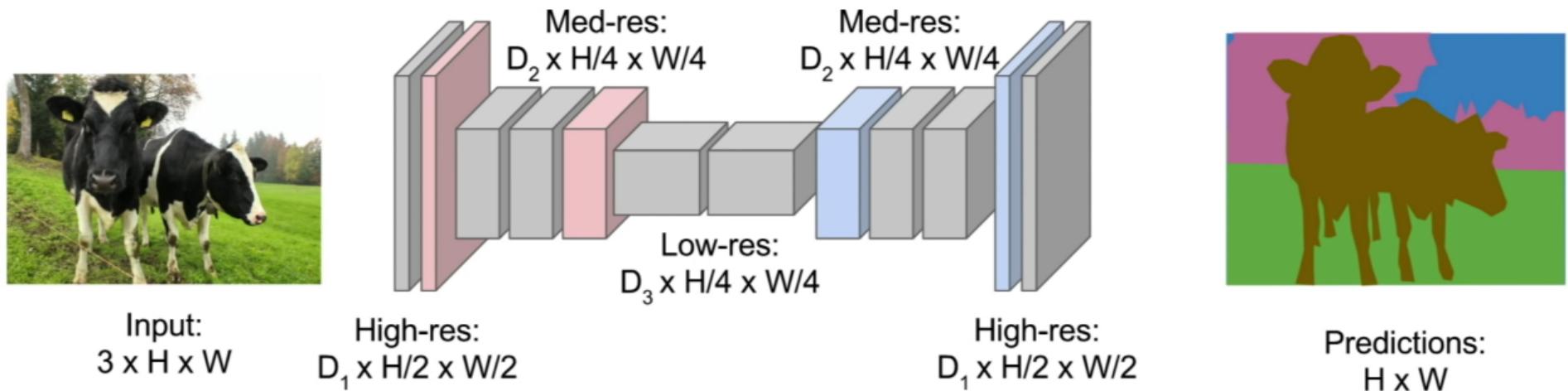
→ loss 계산위 → 역전파 업데이트

모든 픽셀들에 대한 loss를 계산하기 위해서는 train 이미지 픽셀 전부가 labeling되어 있어야 함.

→ 원본 size를 계속 줄이시킬 / 모든 픽셀에 대해 처리

Semantic Segmentation Idea: Fully Convolutional

Design network as a bunch of convolutional layers, with **downsampling** and **upsampling** inside the network!

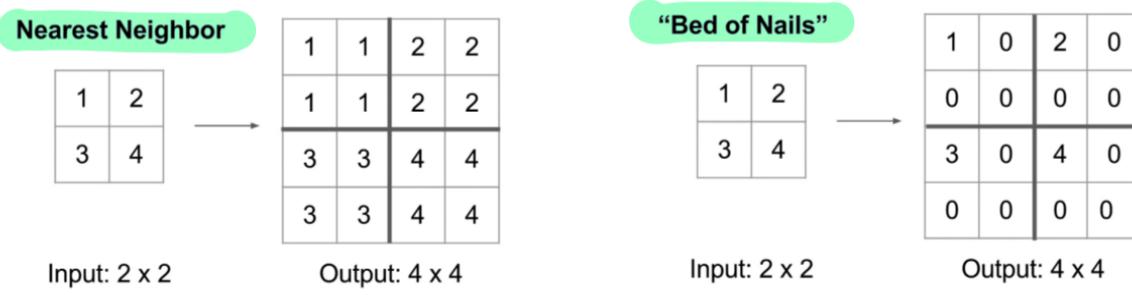


→ **설명** : downSampling은 해줄거, upSampling은 뭘까.

upsampling?

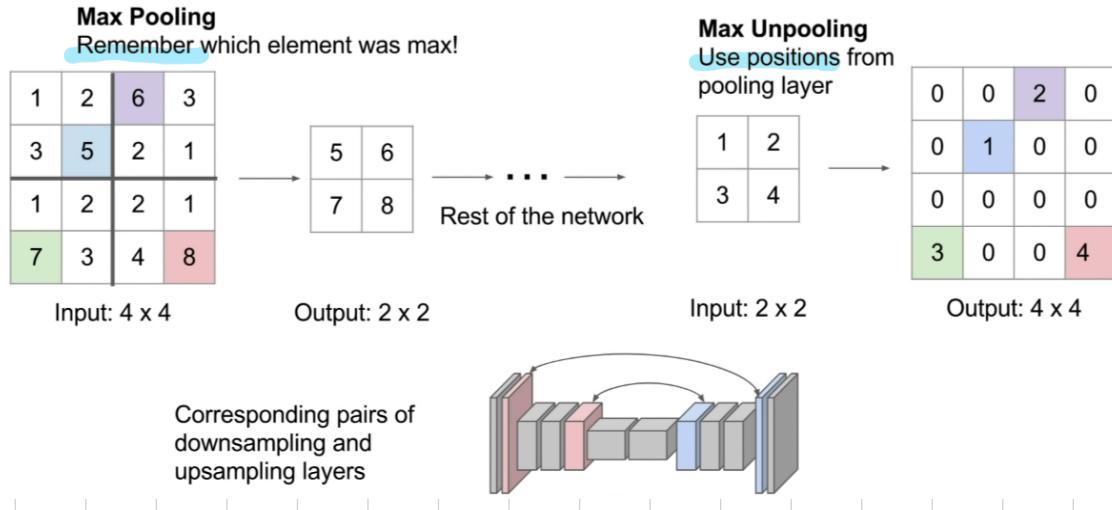
① **unpooling**

In-Network upsampling: “Unpooling”



② **Max Unpooling**

In-Network upsampling: “Max Unpooling”

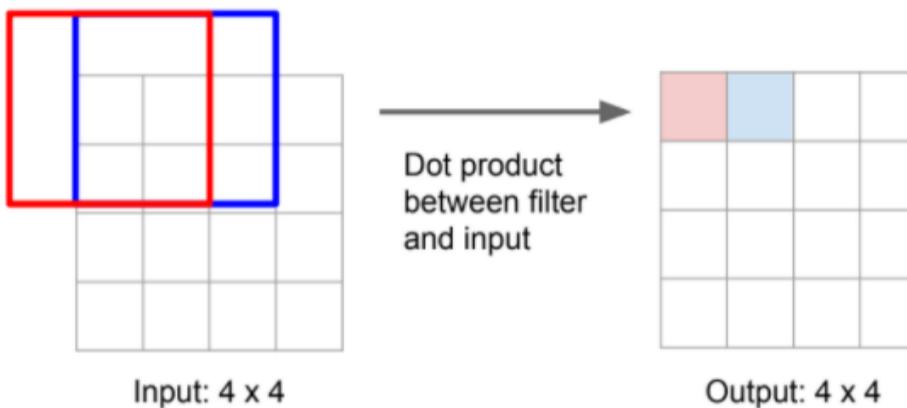


①, ②는 고정되어 있다(?)

③ Transpose Convolution

Learnable Upsampling: Transpose Convolution

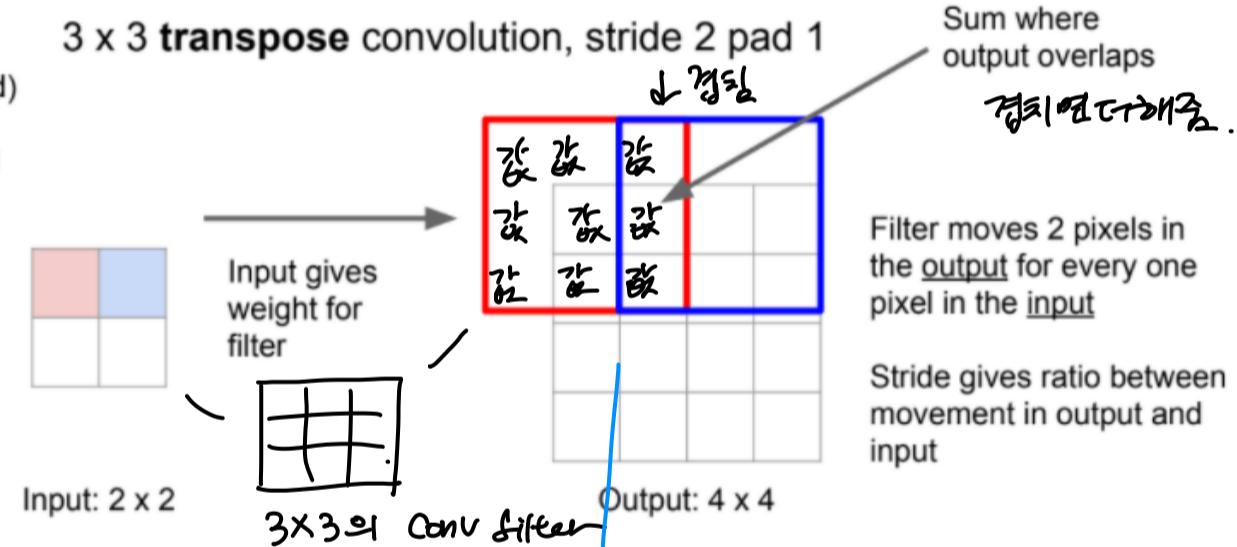
Recall: Normal 3 x 3 convolution, stride 1 pad 1



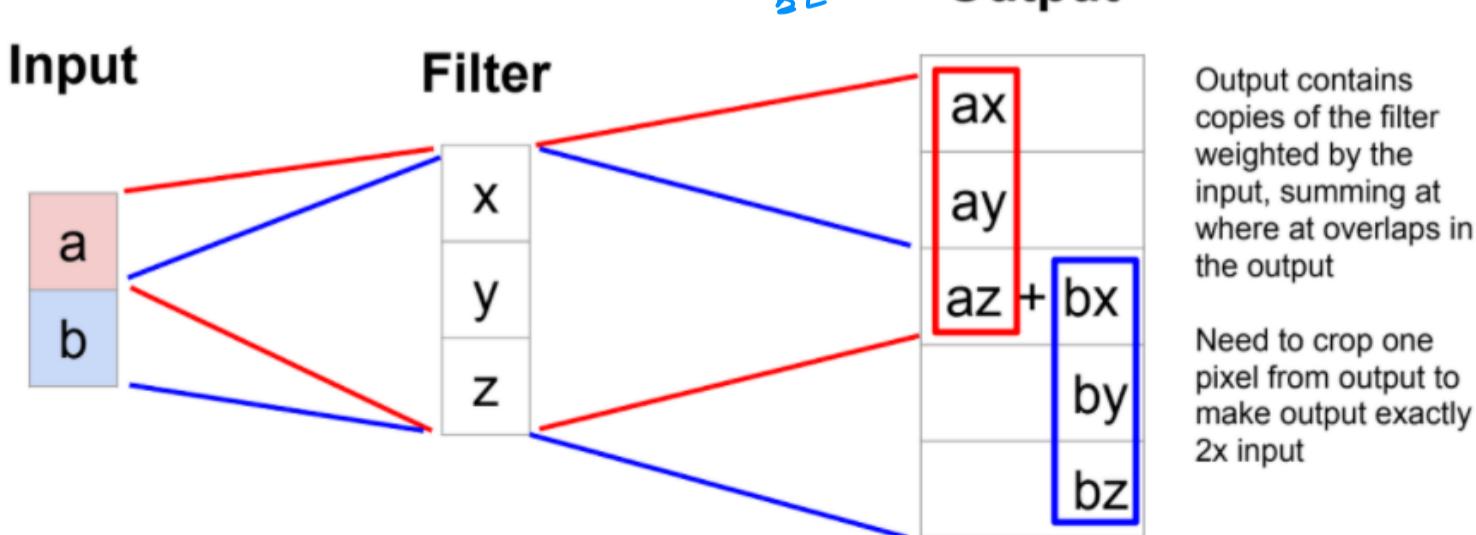
기존과 있는 절차를
Stride 2로는 흔히 사용
Conv
↑ transpose 학습 가능

Learnable Upsampling: Transpose Convolution

Other names:
 -Deconvolution (bad)
 -Upconvolution
 -Fractionally strided convolution
 -Backward strided convolution



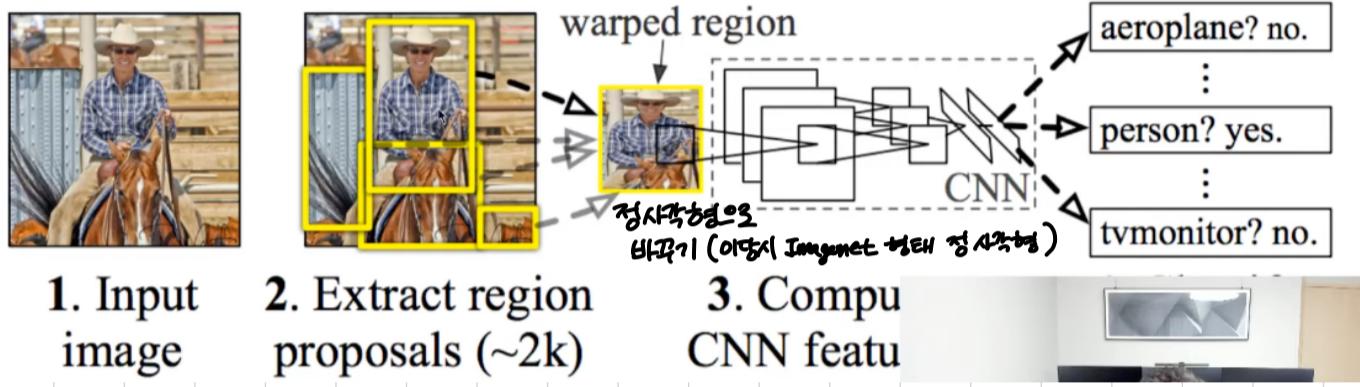
Transpose Convolution: 1D Example



2. R-CNN (Region)

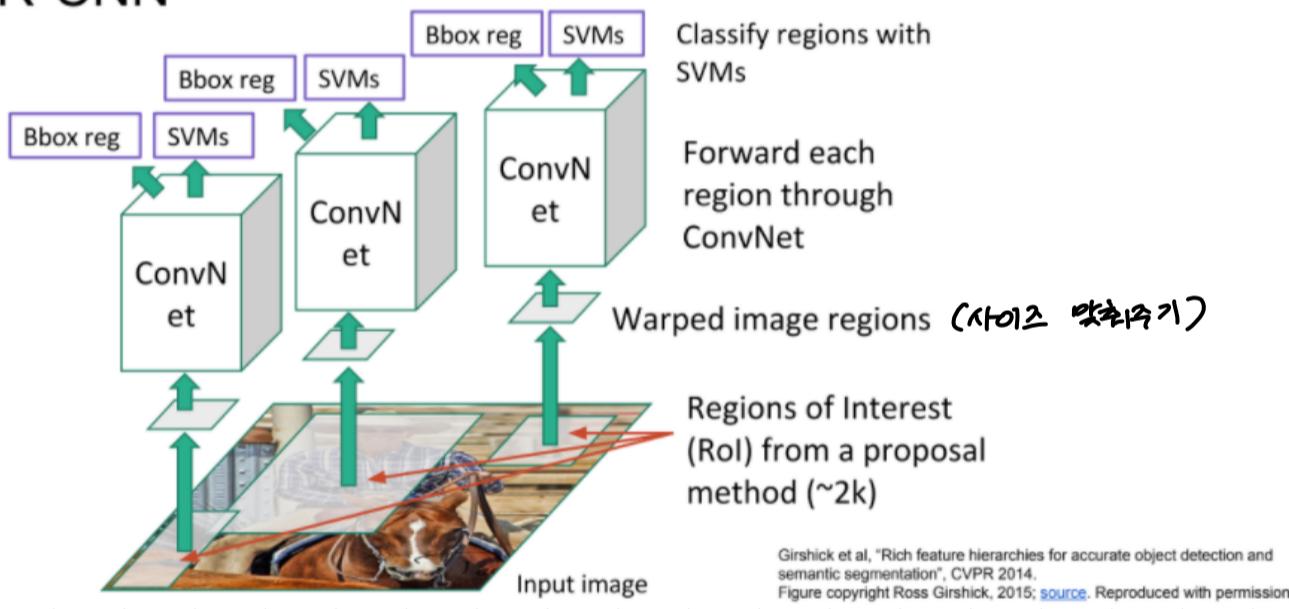
- 객체 탐지에 사용된 초기 모델 (*Segmentation* 뷔다는...) *(detection)*
- 주요 객체들을 바운딩 박스로 표현하여 정확히 식별하는게 목표
- Selective Search를 통해 다양한 크기의 박스를 만들고, region proposal 영역 생성
- region proposal 영역을 warp하여 표준화된 크기로 변환
- AlexNet을 개량한 CNN 모델을 이용하고, 마지막 층에 SVM을 통해 객체 분류

R-CNN: Regions with CNN features



* ROI : Regions of interest

R-CNN



- ROI에 CNN 수행

fast R-CNN

- R-CNN의 단점인 느린 속도를 빠른 속도로 개선
- ROI(Region of Interest) 풀링을 통해 한 이미지의 subregion에 대한 forward pass 값을 공유
- R-CNN은 CNN 모델로 image feature를 추출, SVM 모델로 분류, Regressor 모델로 bounding box를 맞추는 작업으로 분류되어 있지만, Fast R-CNN은 하나의 모델로 동작
- Top layer에 softmax layer를 둬서 CNN 결과를 class로 출력
- Box regression layer를 softmax layer에 평행하게 두어 bounding box 좌표를 출력

Fast R-CNN (Training)

