Lecture note 8 : Typical tasks with CNN

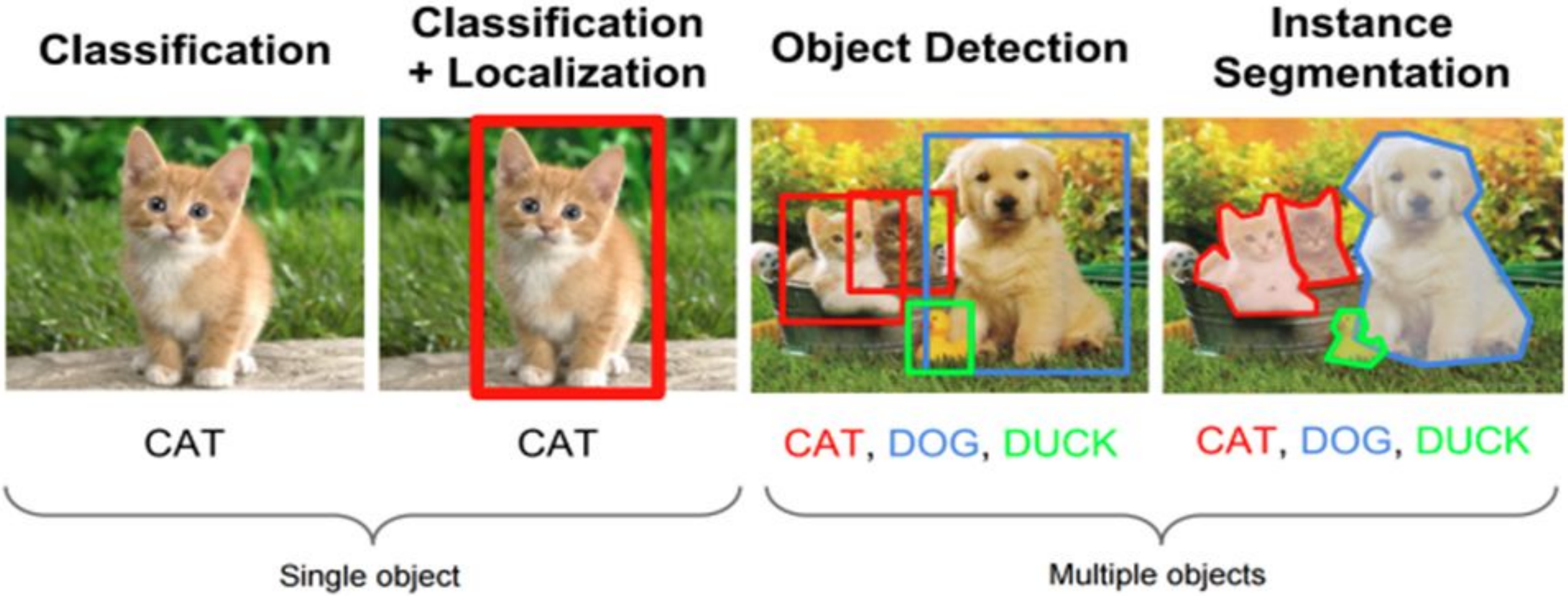
프로젝트 기반 딥러닝 이미지처리
한국인공지능아카데미 x Hub Academy

강사 : 김형욱 (hyounguk1112@gmail.com)

Agenda

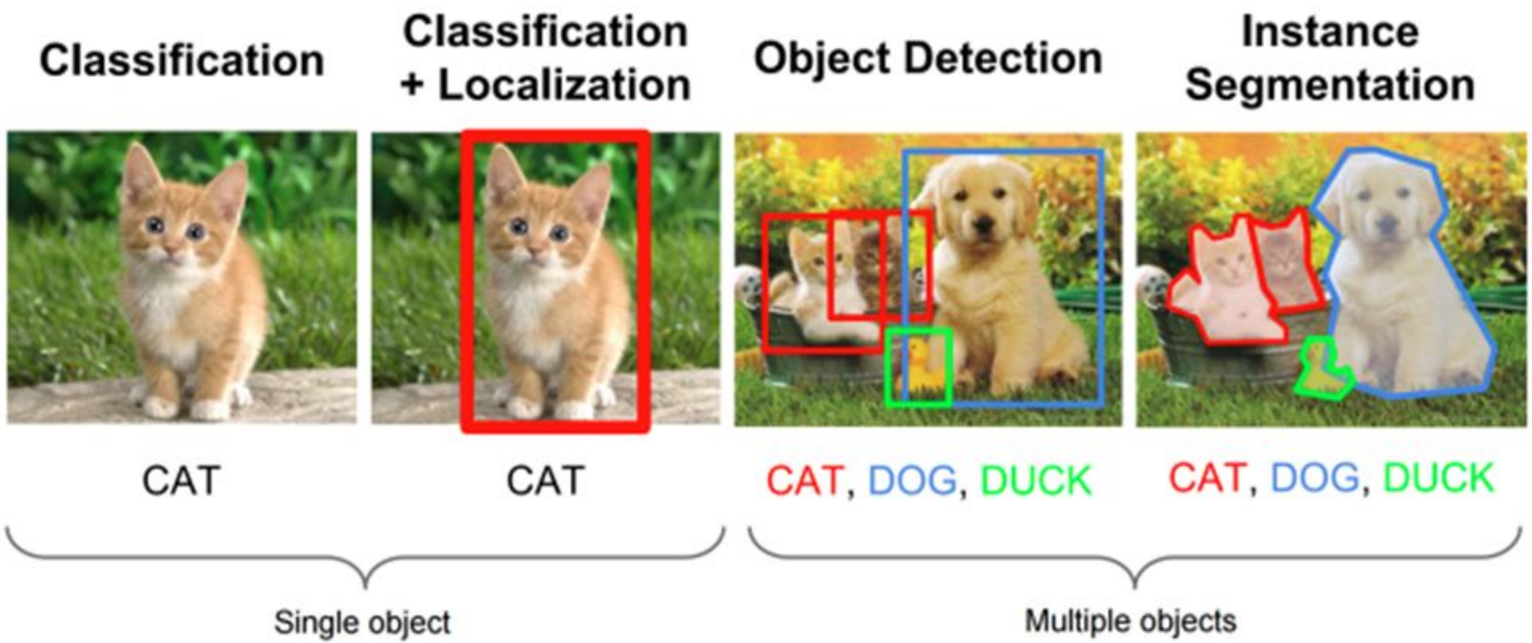
1. Regional CNN
2. Segmentation
3. GAN

Typical tasks

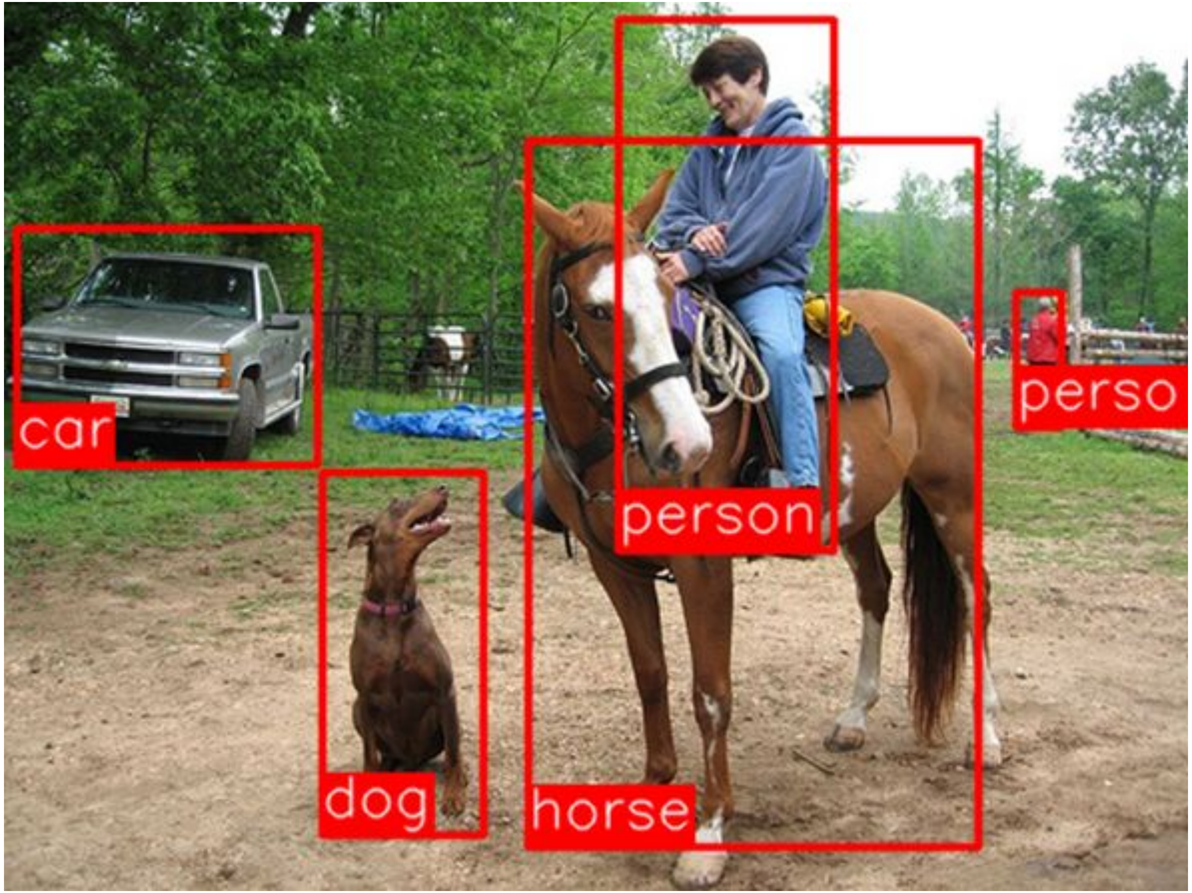


Regional CNN

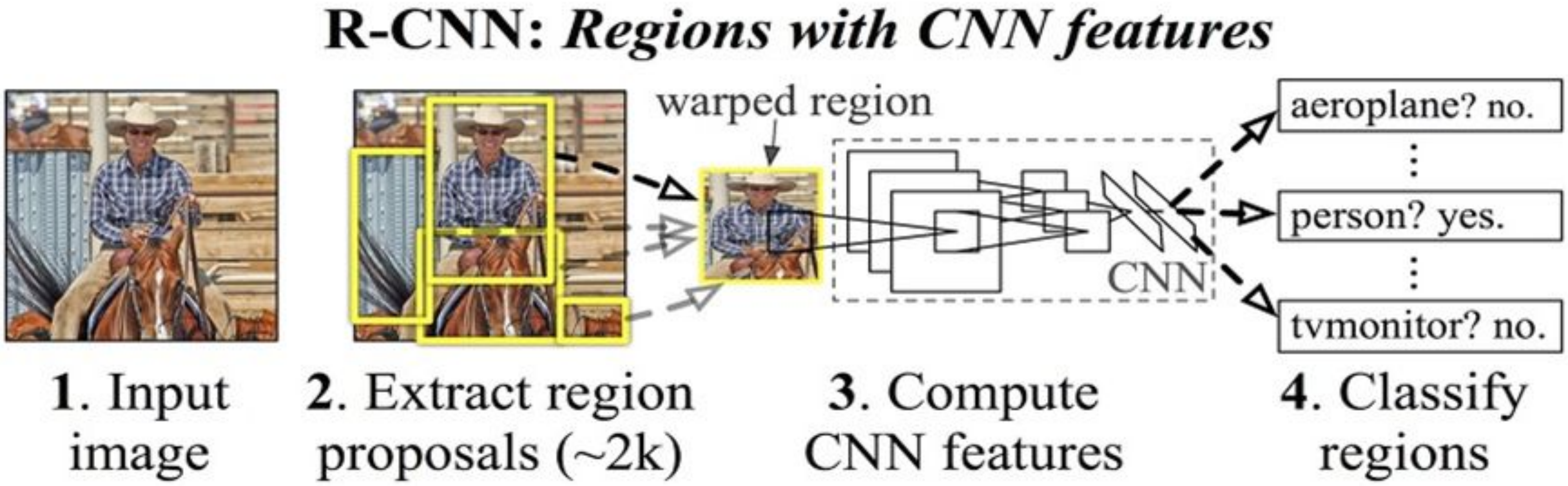
객체검출(Object detection)



객체검출(Object detection)



RCNN



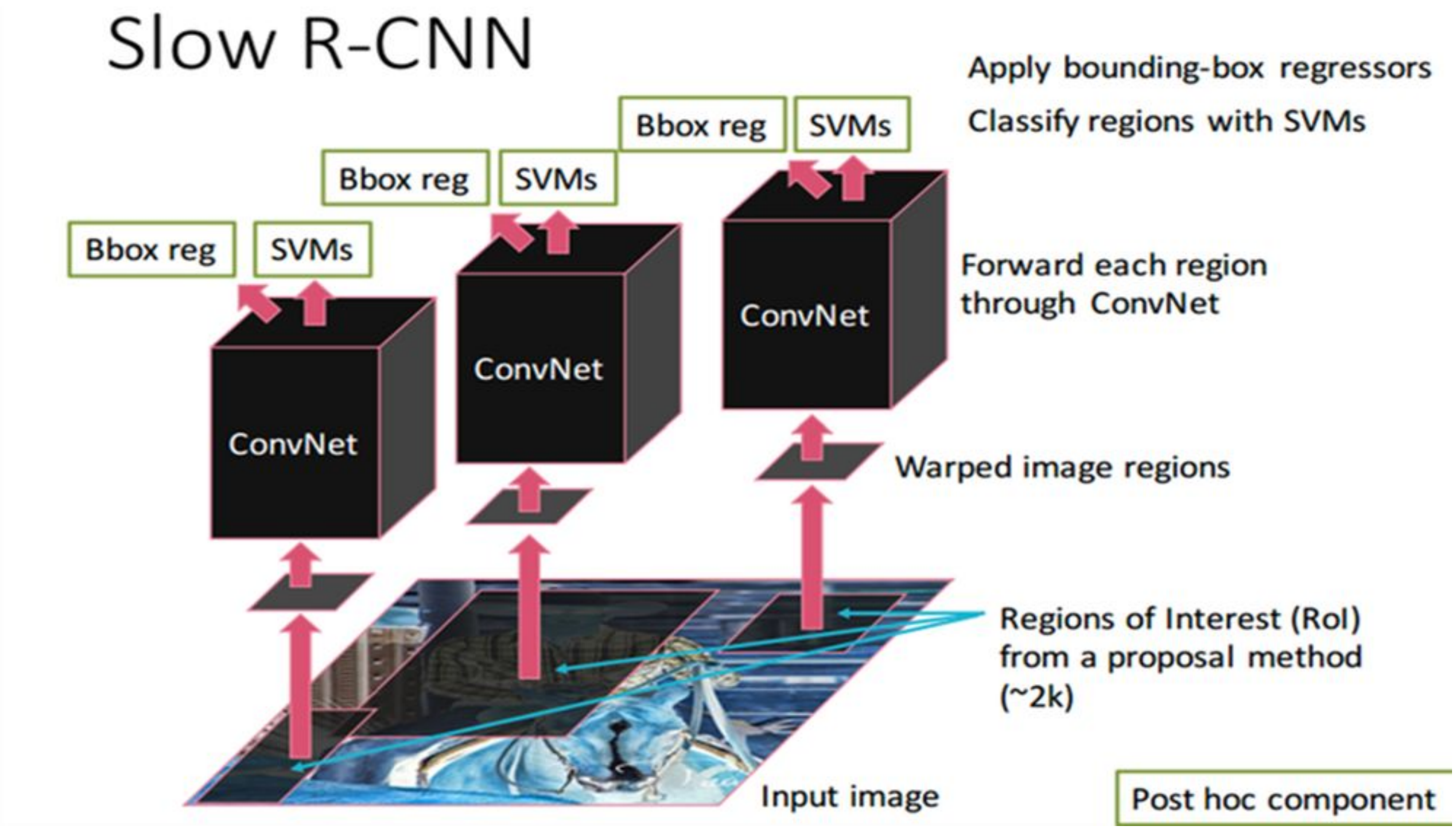
Step 2. CNN을 분류 문제에 대해 사전 지도학습(Domain specific classification learning)
Step 1. 이미지 입력
Step 2. Selective search 알고리즘을 통해 region proposal 수행
Step 3. 각 region을 동일한 사이즈로 warping하여 CNN에 입력하여 특징 검출
Step 4. 분류기를 통해 각 region을 분류
Step 5. 마지막으로 Non-maximum suppresion 알고리즘을 통해 지역을 확정함.

RCNN - Selective search

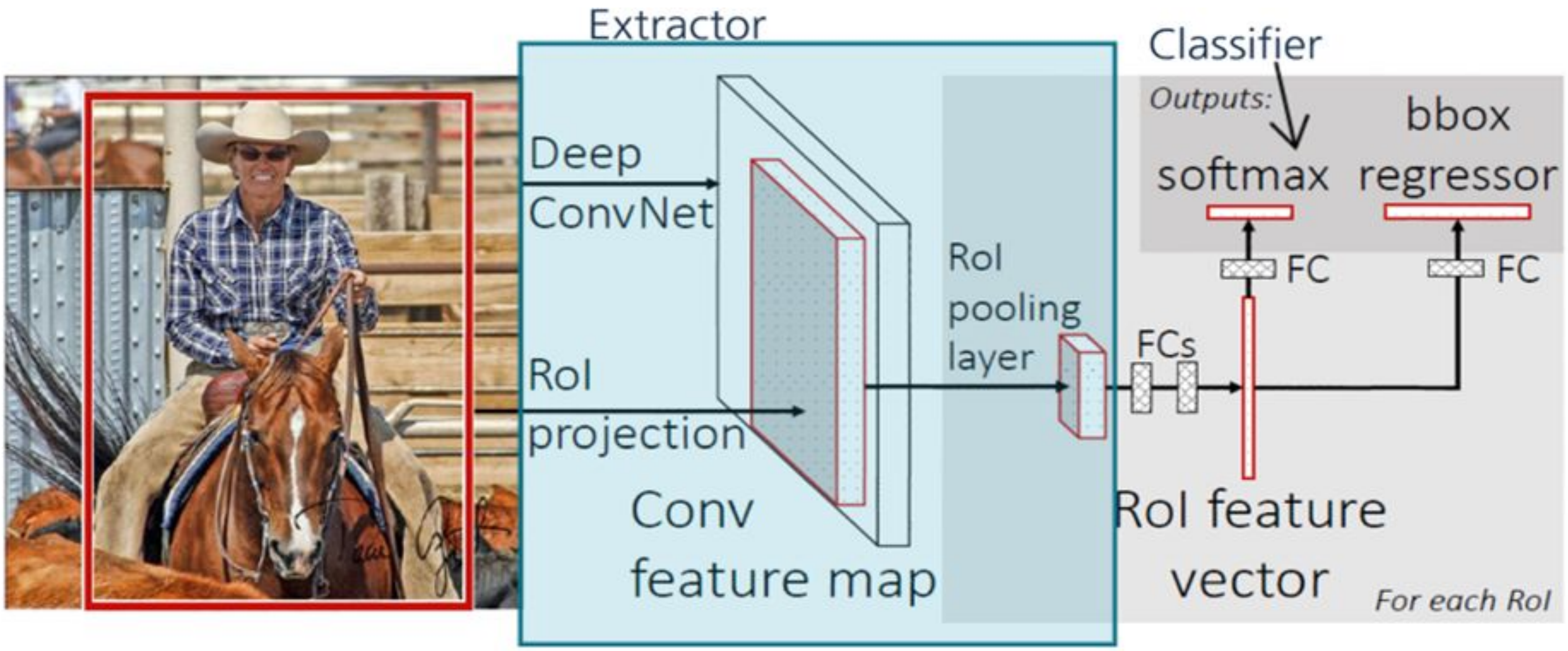
Figure 2: Two examples of our selective search showing the necessity of different scales. On the left we find many objects at different scales. On the right we necessarily find the objects at different scales as the girl is contained by the tv.

- Group regions with some rules(intensity, color..)
- In experience, S.S resulted in an average of 2403 region proposals per image with a 91.6% recall of all ground-truth bounding boxes(0.5 IoU threshold)

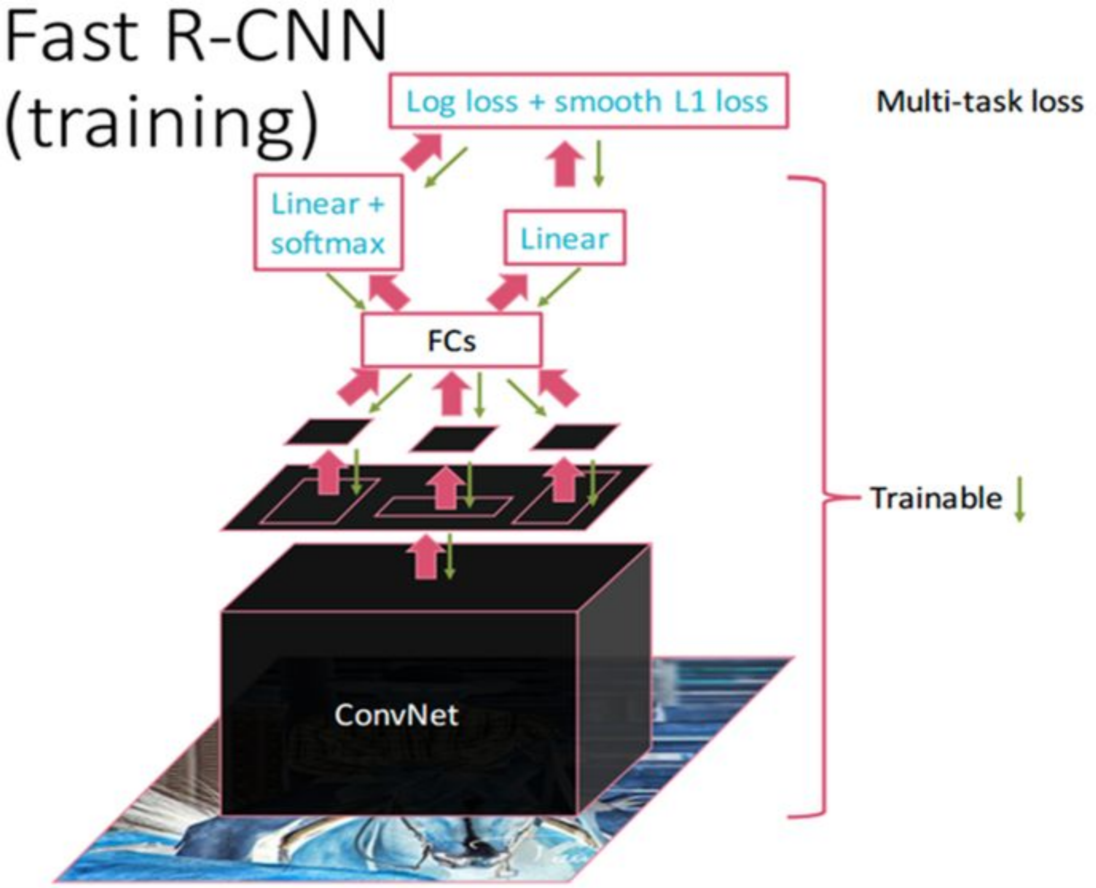
RCNN


Slow R-CNN
Apply bounding-box regressors
Classify regions with SVMs
Bbox reg
SVMs
Bbox reg
SVMs
Bbox reg
SVMs
ConvNet
ConvNet
ConvNet
Forward each region through ConvNet
Warped image regions
Regions of Interest (RoI) from a proposal method (~2k)
Input image
Post hoc component

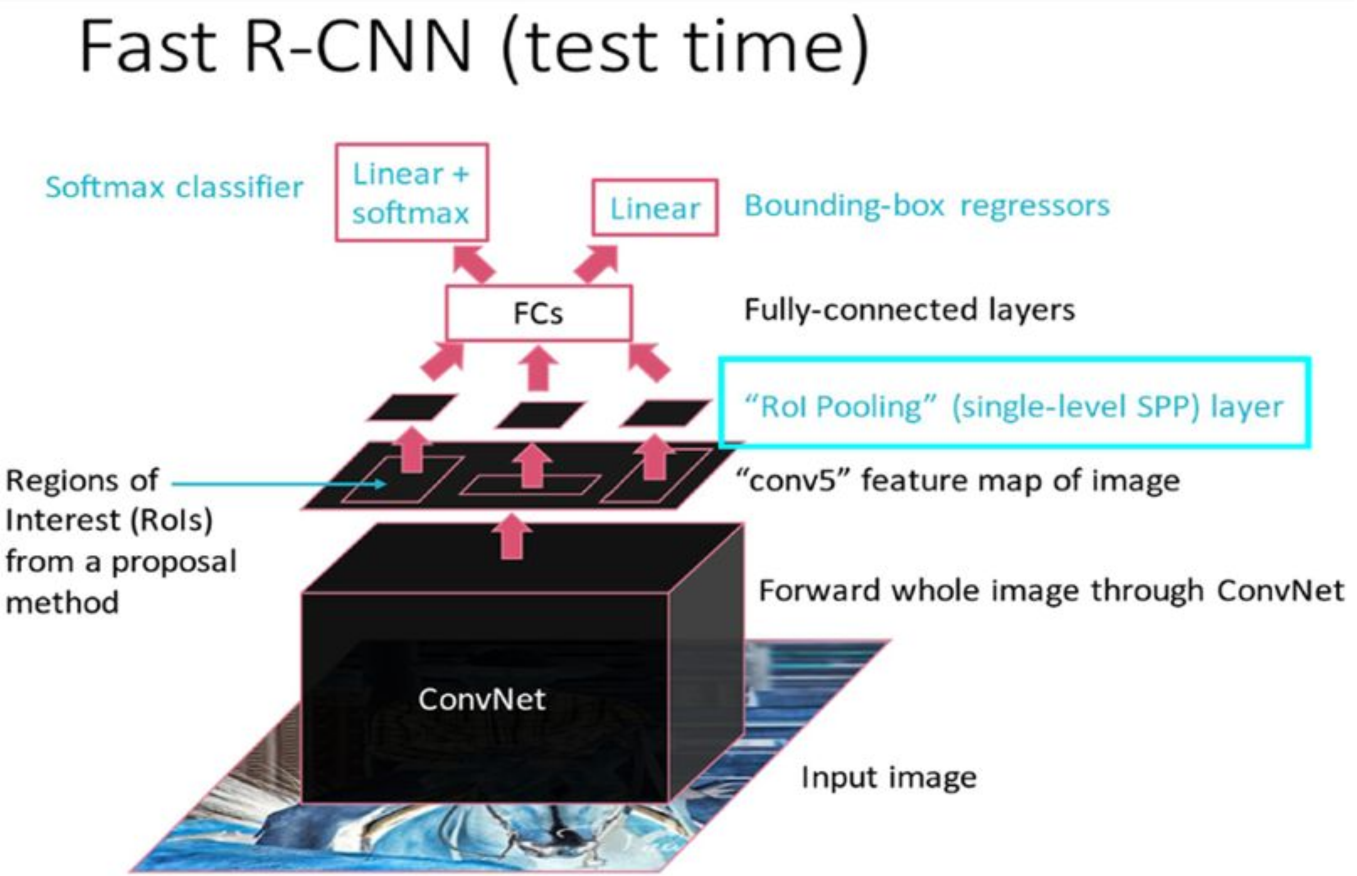
Fast RCNN : Overview



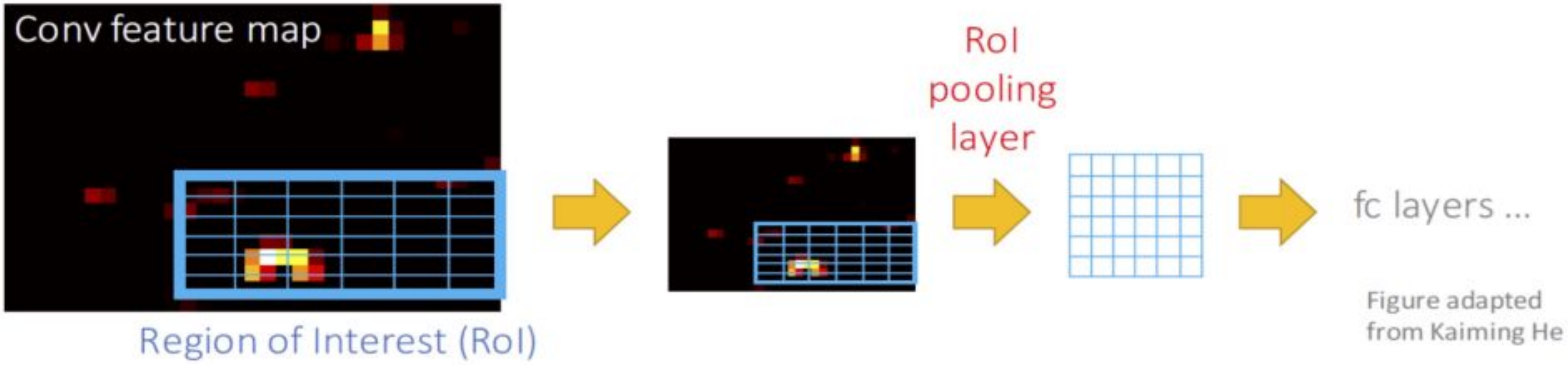
Fast RCNN : training process



Fast RCNN : testing process

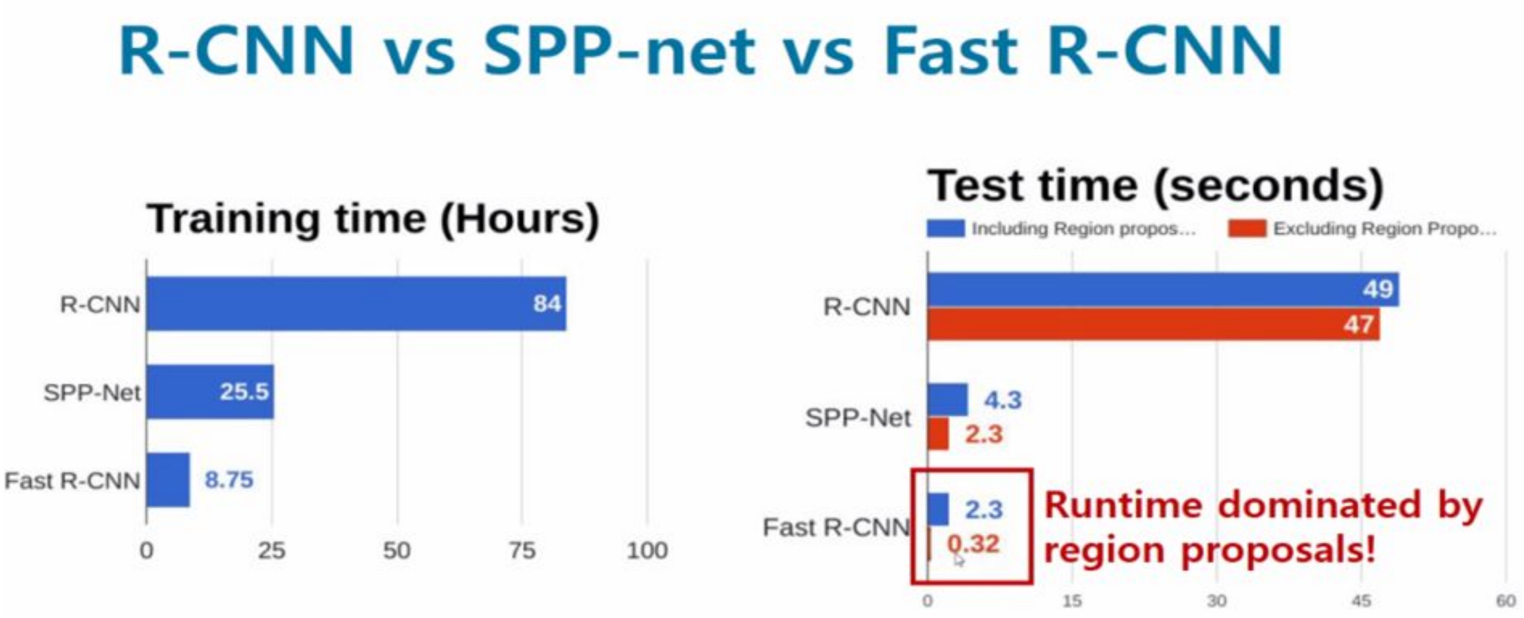


Fast RCNN : RoI pooling layer

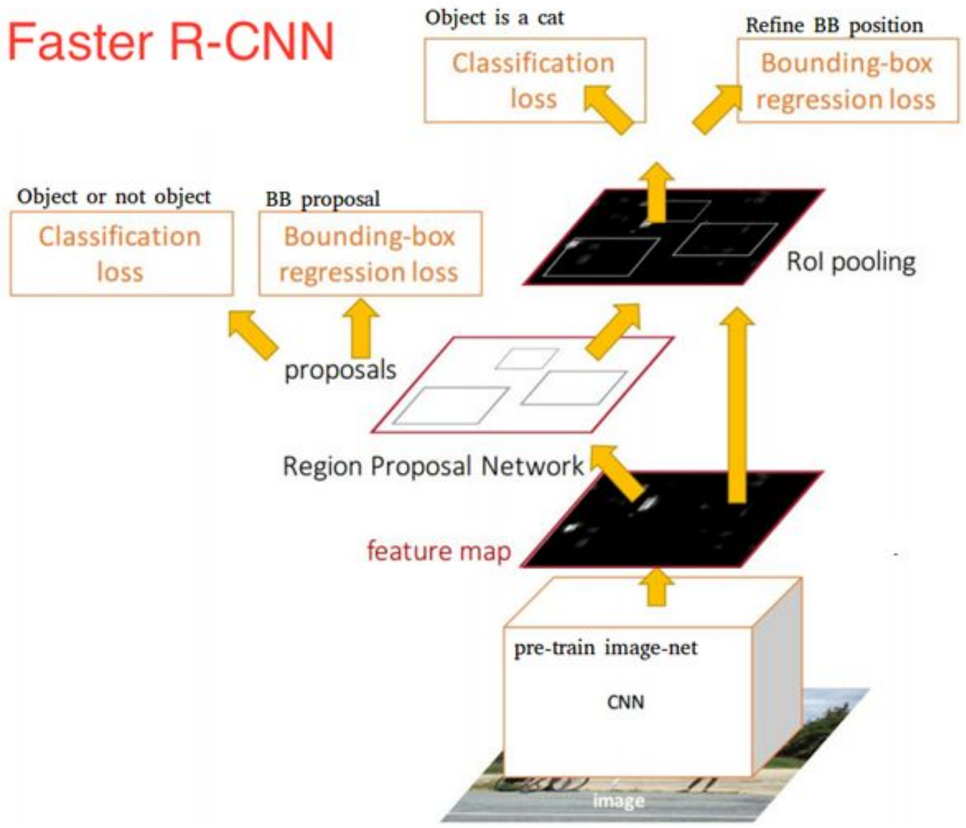


RoI in Conv feature map : 21x14 → 3x2 max pooling with stride(3, 2) → output : 7x7
RoI in Conv feature map : 35x42 → 5x6 max pooling with stride(5, 6) → output : 7x7

Fast RCNN : Weak point

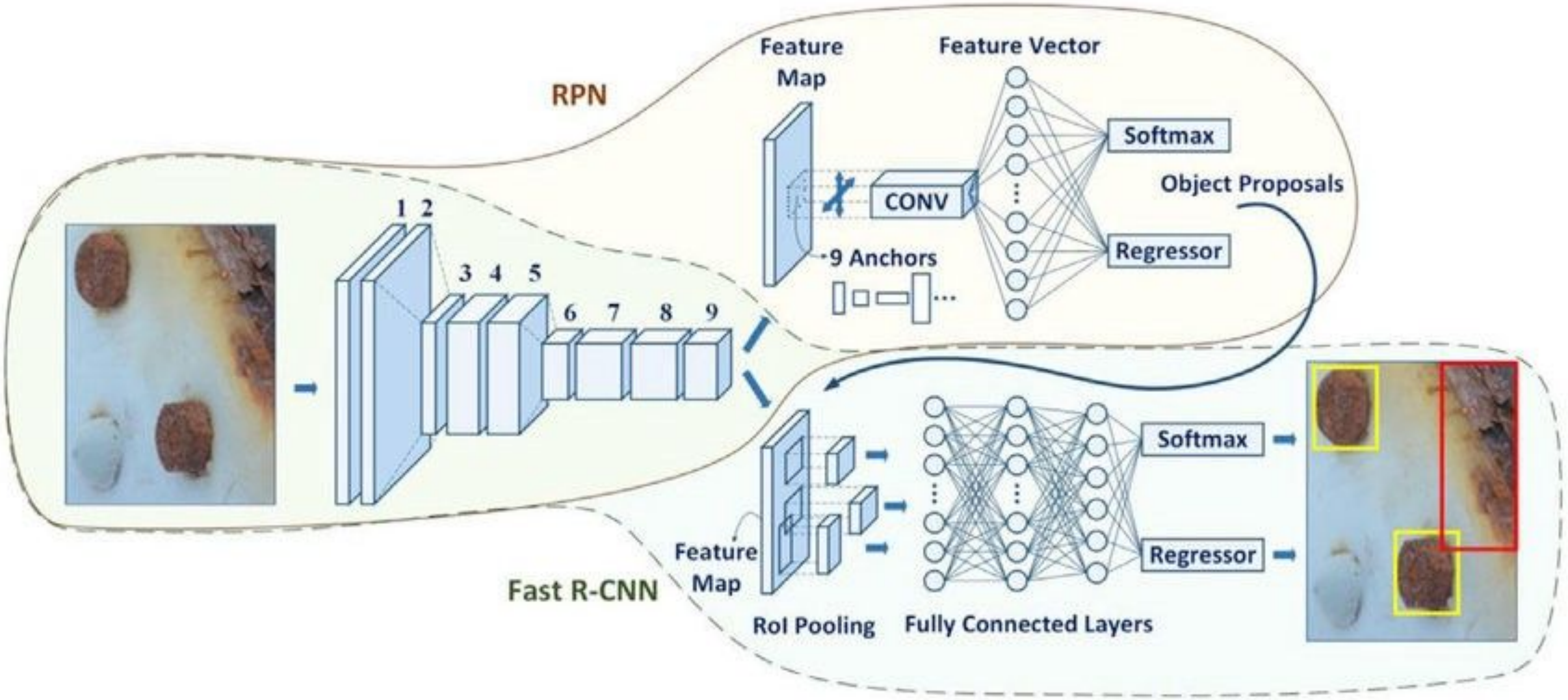


Faster RCNN



Fast RCNN의 병목(Bottle-neck) 부분이었던 Selective search를 신경망 구조로 변경

Faster RCNN

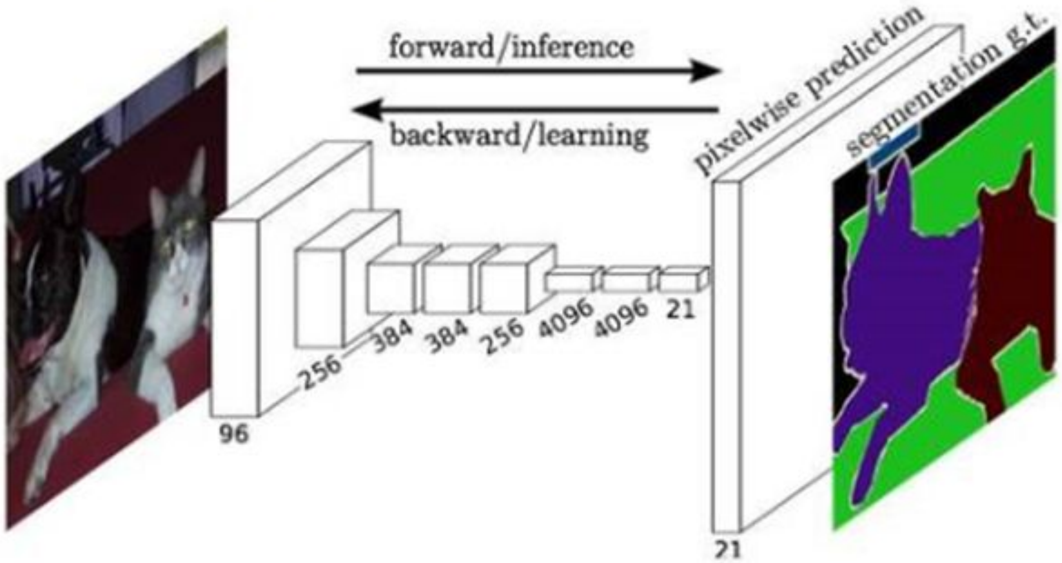


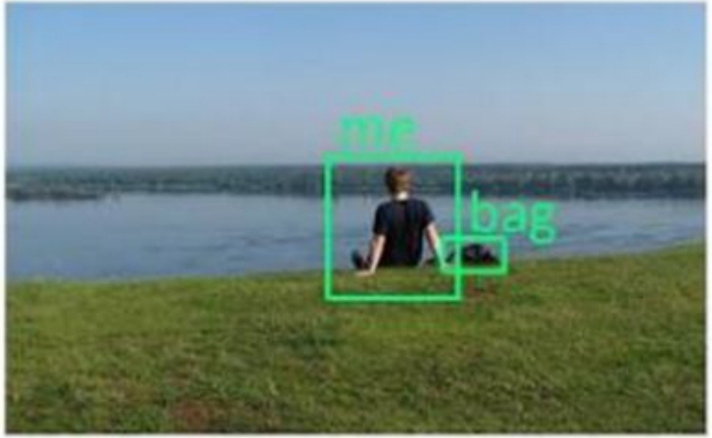
Ref : towardsdatascience.com

Segmentation

Fully Convolutional Networks for Semantic Segmentation

Jonathan Long, Evan Schelhamer, Trevor Darrell
UC Berkeley





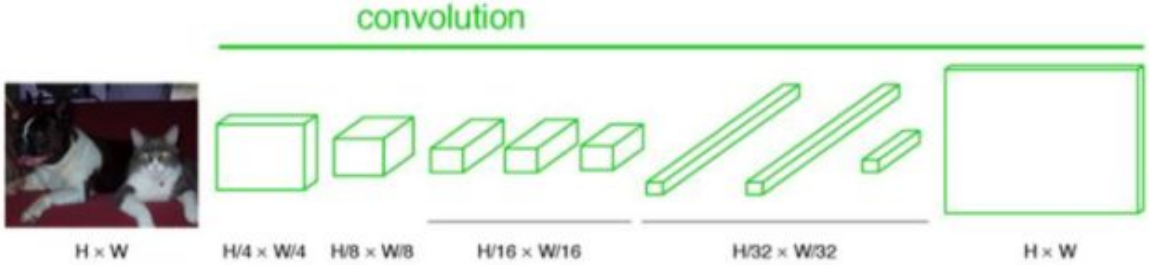
Object Detection
: addresses the problem of localization of objects of the certain classes

e.g. Sliding window of varying size and classify sub-images defined by the window

Semantic Segmentation(or pixel classification)
: associates one of the defined class labels to each pixel

e.g. pixels are classified with regard to their local features, such as color and/of texture features. Markov Random Fields could be used to incorporate inter-pixel relations.

upsampling output



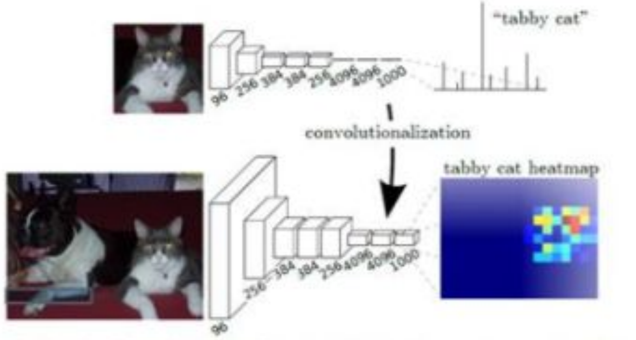
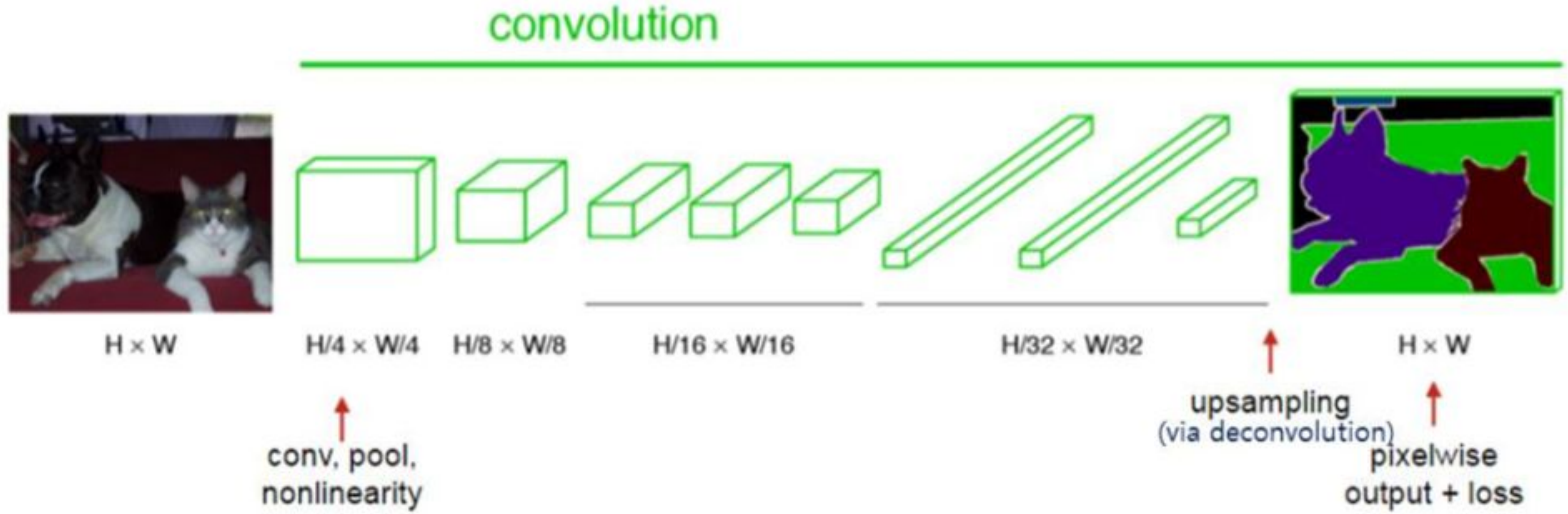
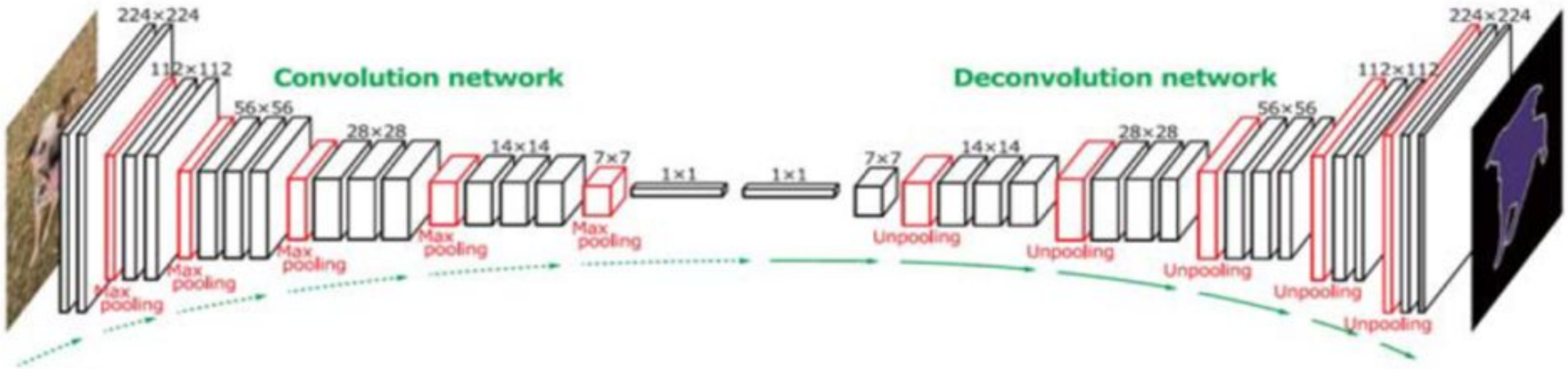


Figure 2. Transforming fully connected layers into convolution layers enables a classification net to output a heatmap. Adding layers and a spatial loss (as in Figure 1) produces an efficient machine for end-to-end dense learning.

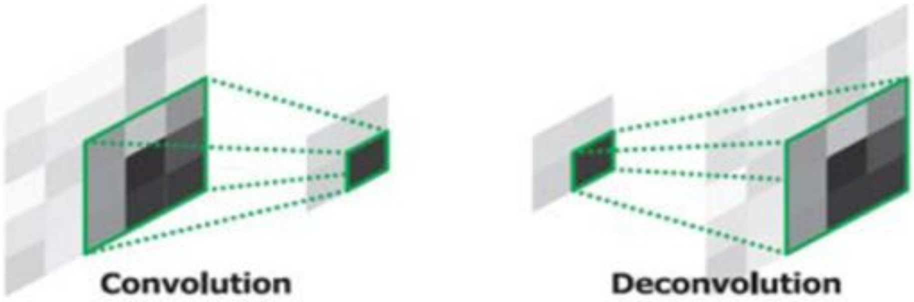
end-to-end, pixels-to-pixels network



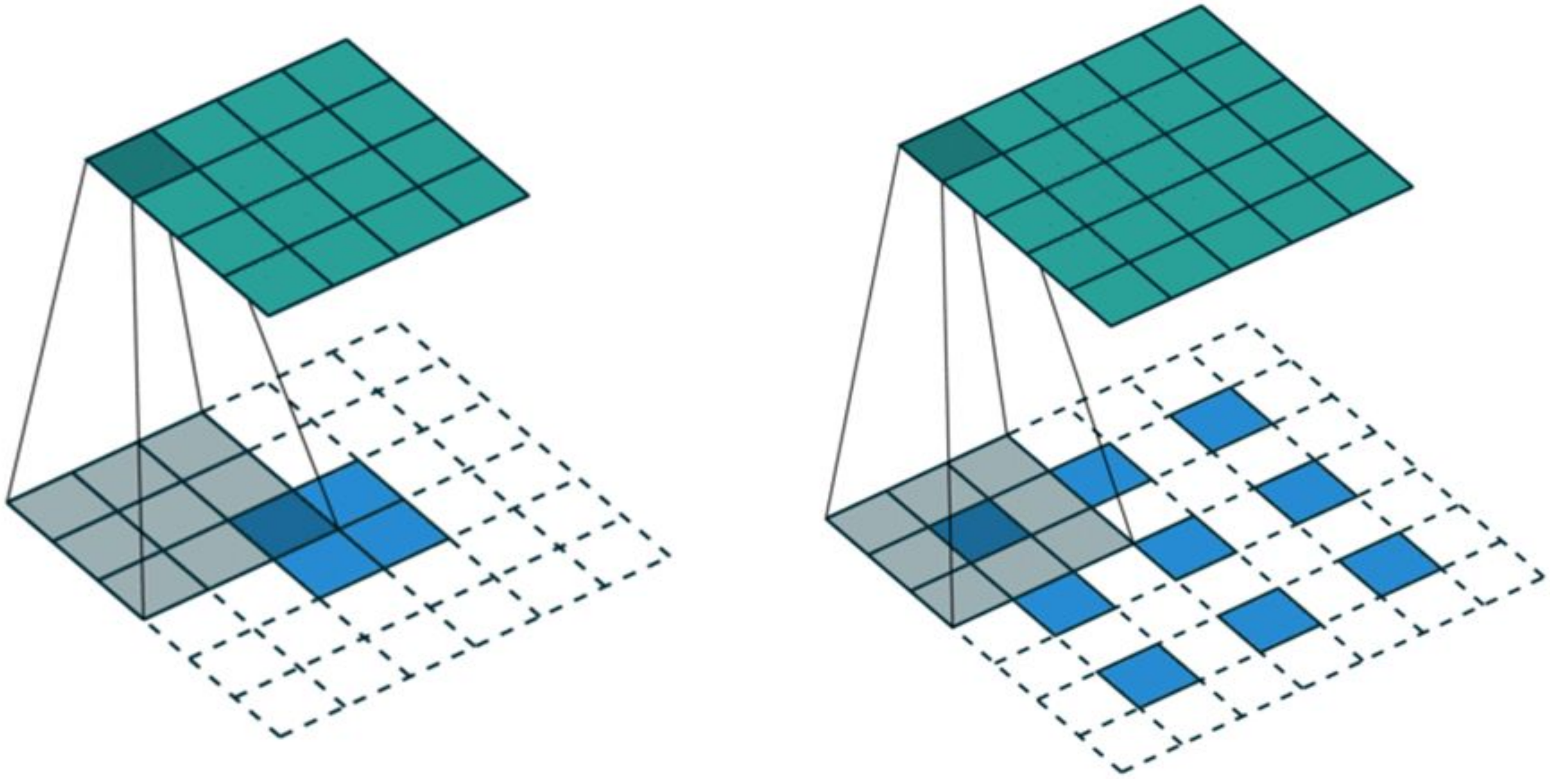

Convolution network
Deconvolution network
224×224
112×112
56×56
28×28
14×14
7×7
1×1
1×1
7×7
14×14
28×28
56×56
112×112
224×224
Max pooling
Max pooling
Max pooling
Max pooling
Max pooling
Unpooling
Unpooling
Unpooling
Unpooling
Unpooling

- Unpooling
  - Place activations to pooled location
  - Preserve structure of activations

- Deconvolution
  - Densify sparse activations
  - Bases to reconstruct shape

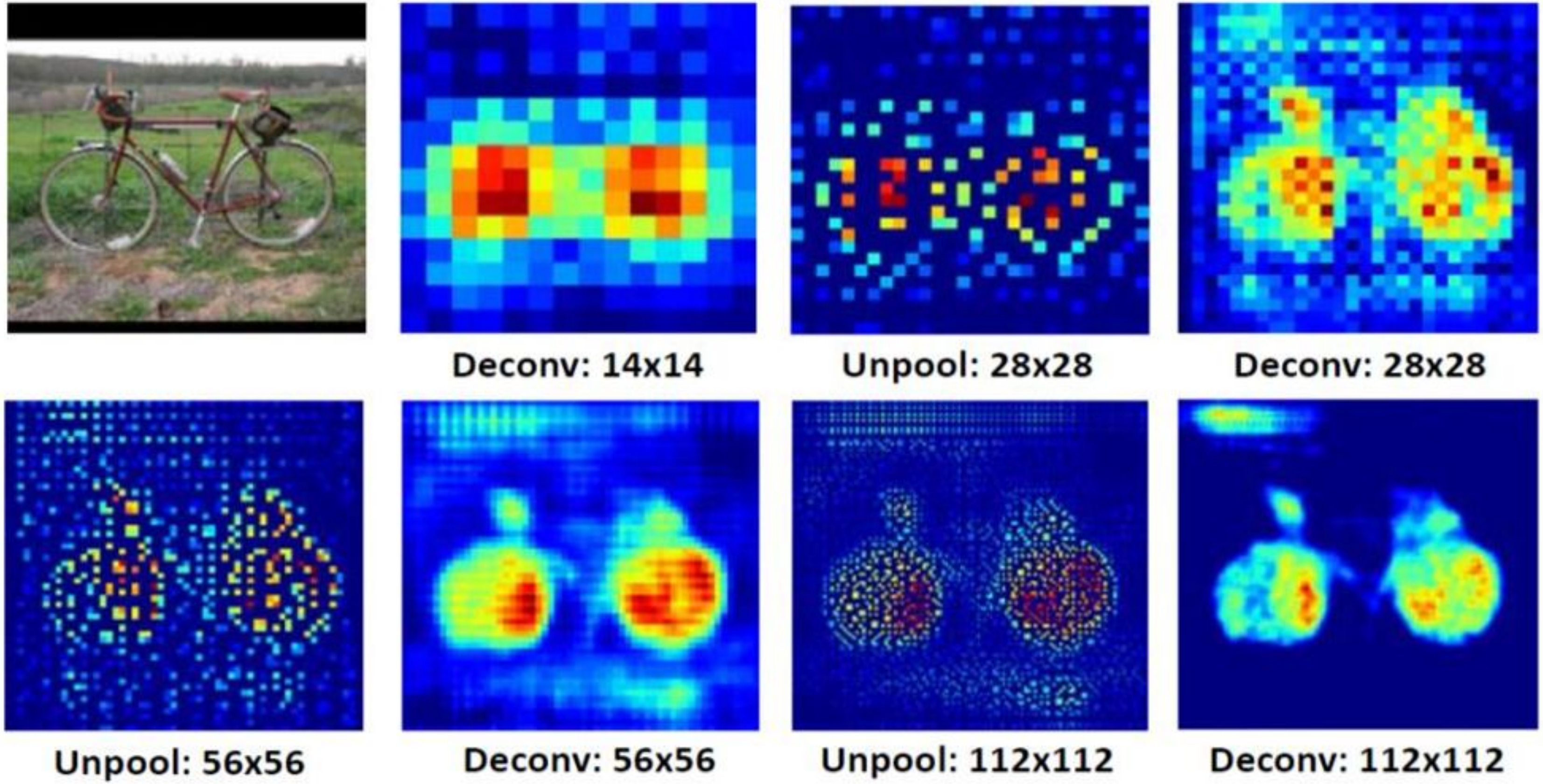


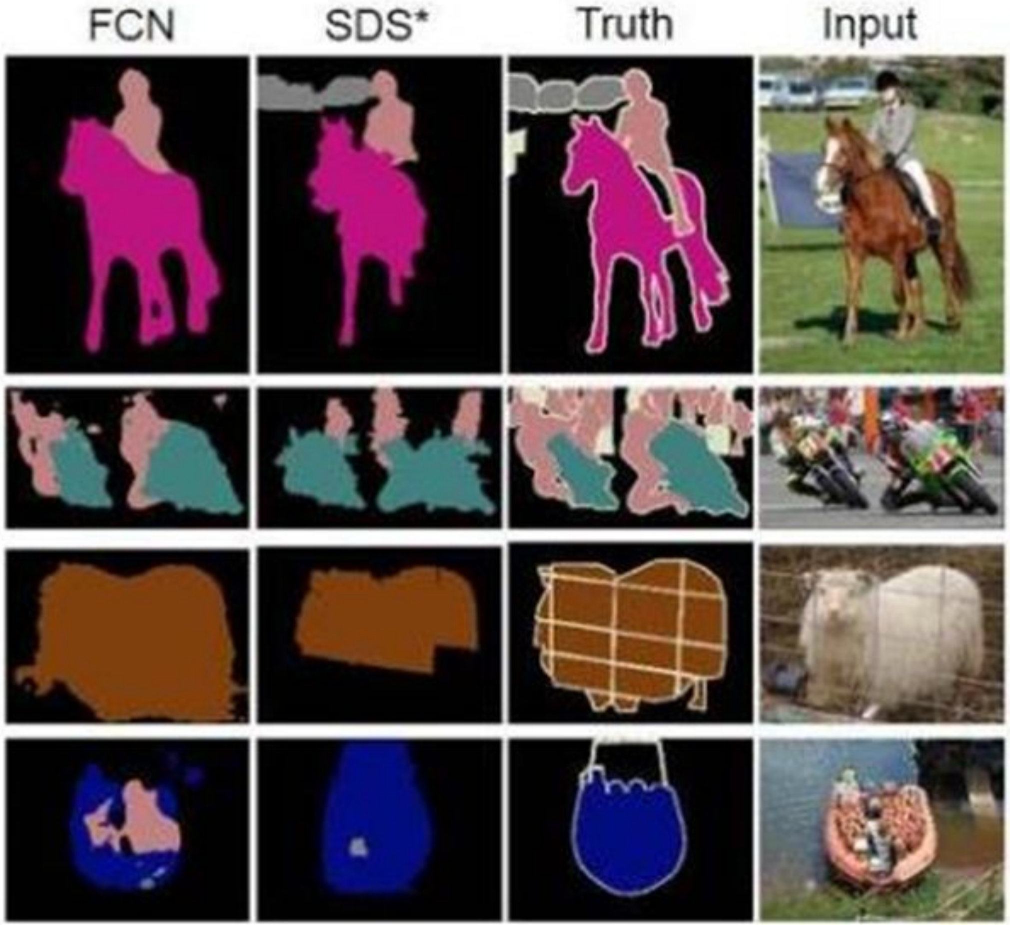
- ReLU
  - Same with convolution network



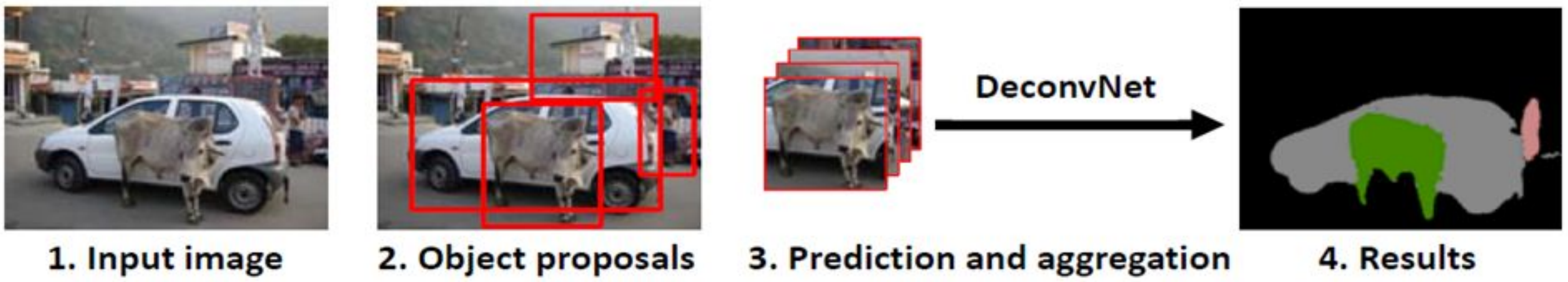
Deconvolution , Convolution transposed, Full convolution

Visualization of activations




FCN
SDS*
Truth
Input

Instance segmentation



GAN

- Ian Goodfellow의 NIPS 2016에서 발표한 논문 'Generative Adversarial Network' 에서 소개되어 많은 후속 연구 및 높은 관심을 받음.

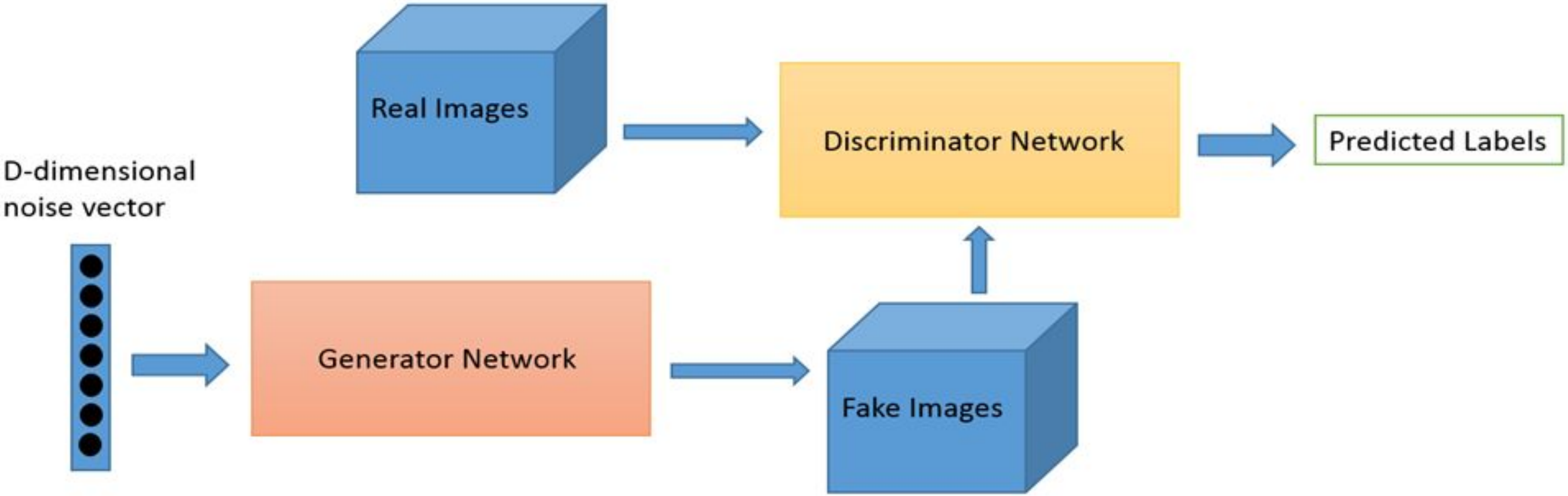
- 이전 Christian Szegedy의 Adversarial training Algorithm에서 영향을 받았다.

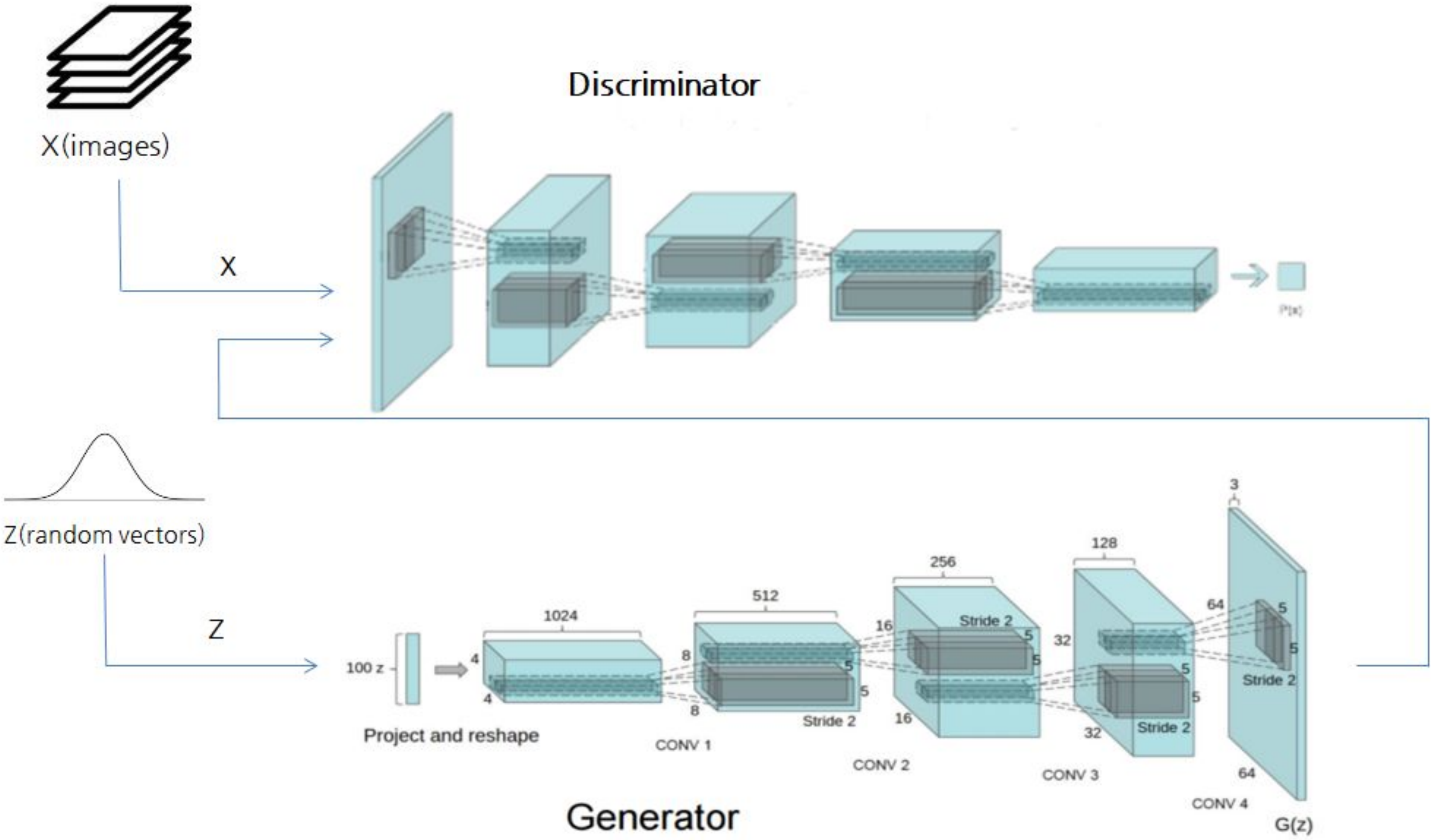
- A super dataset augmenting system (able to create more data from the original data)

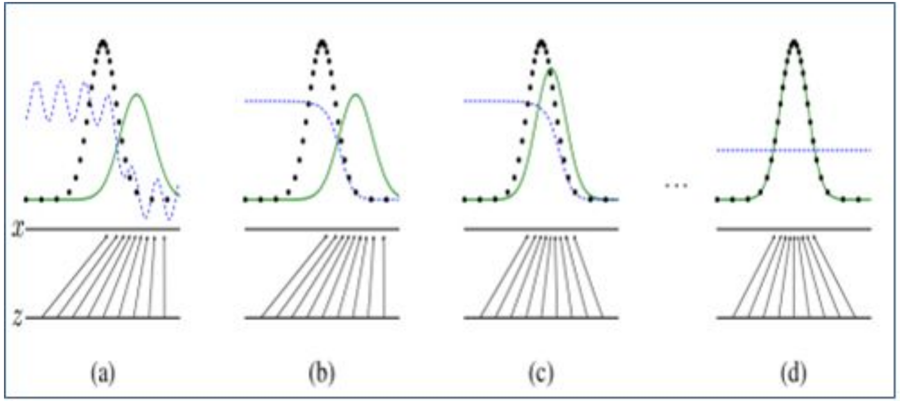
- "what I cannot create, I do not understand."

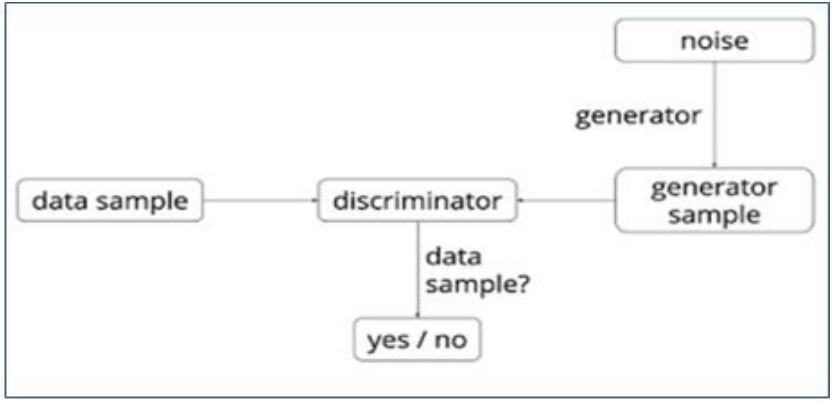
  - *Richard Feynman*

- 지폐위조범과 경찰의 경쟁으로 많이 비유한다.
- 지폐위조범(Generator)은 경찰을 최대한 속이려고 하고, 다른 한편에서는 경찰(Discriminator)이 위조된 지폐를 감별(Classify)하려 노력한다.
- 이런 경쟁 속에서 두 그룹 모두 속이고 구별하는 서로의 능력이 발전하게 되고 결과적으로는 진짜 지폐와 위조 지폐를 구별할 확률이 50%에 이르면 실제와 똑같을 정도로 정교한 지폐를 생성하게 될 것이다.

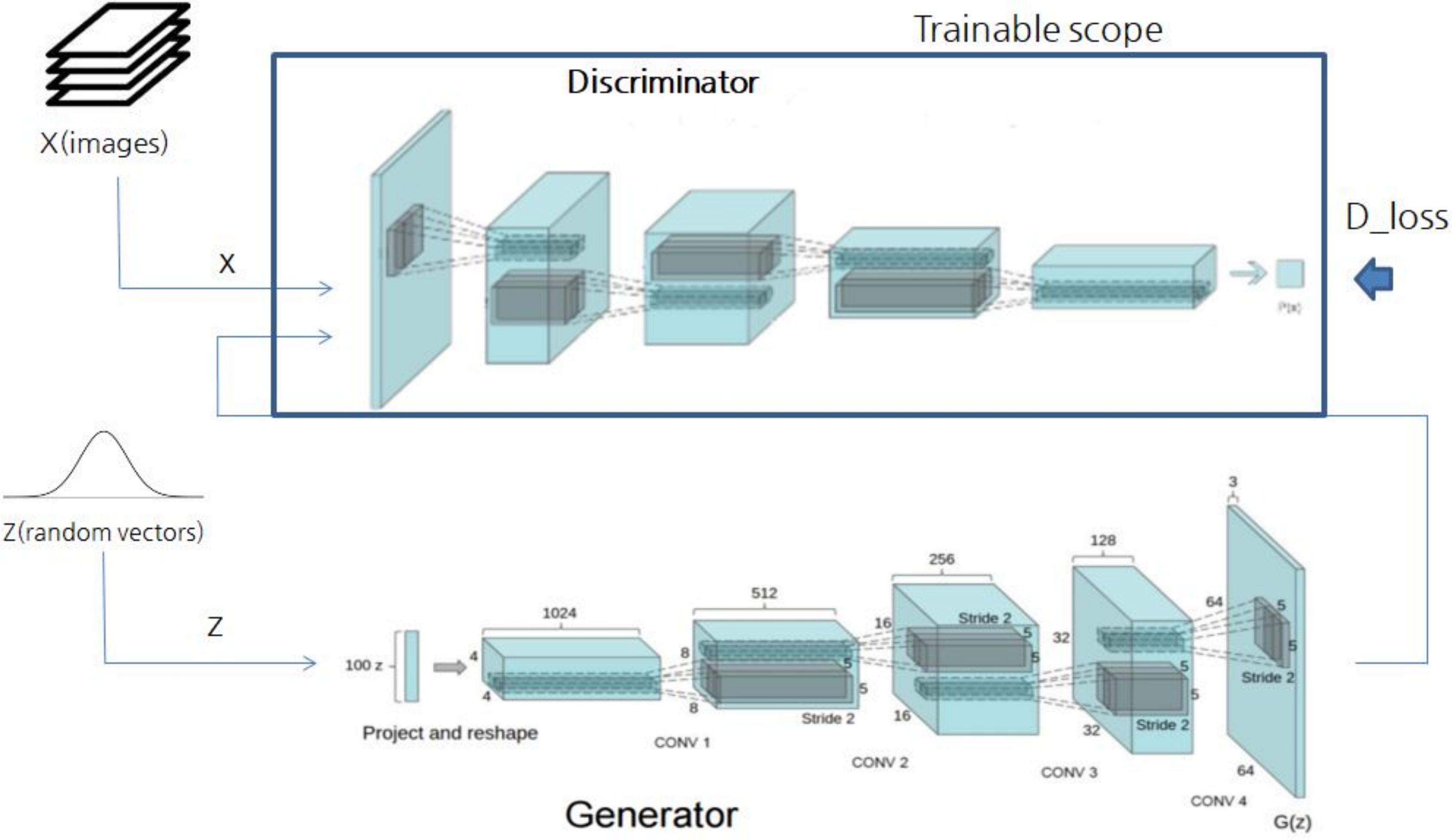

Real Images
Discriminator Network
Predicted Labels
D-dimensional
noise vector
Generator Network
Fake Images


X(images)
Discriminator
X
Z(random vectors)
Z
100 z
Project and reshape
1024
4
4
512
8
8
5
5
Stride 2
CONV 1
256
16
16
Stride 2
5
5
CONV 2
128
32
32
5
5
Stride 2
CONV 3
3
64
5
5
Stride 2
64
CONV 4
G(z)
Generator

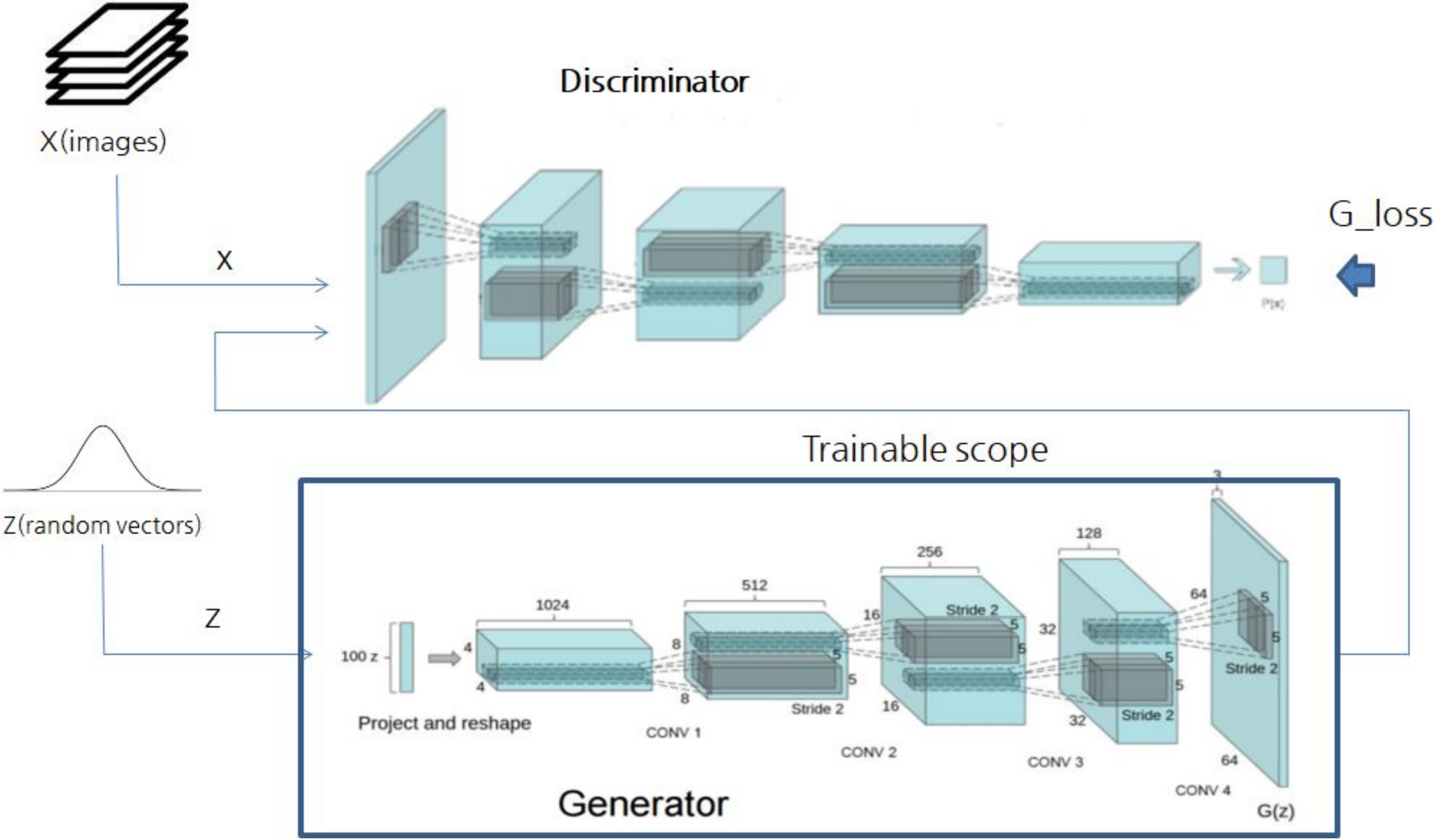


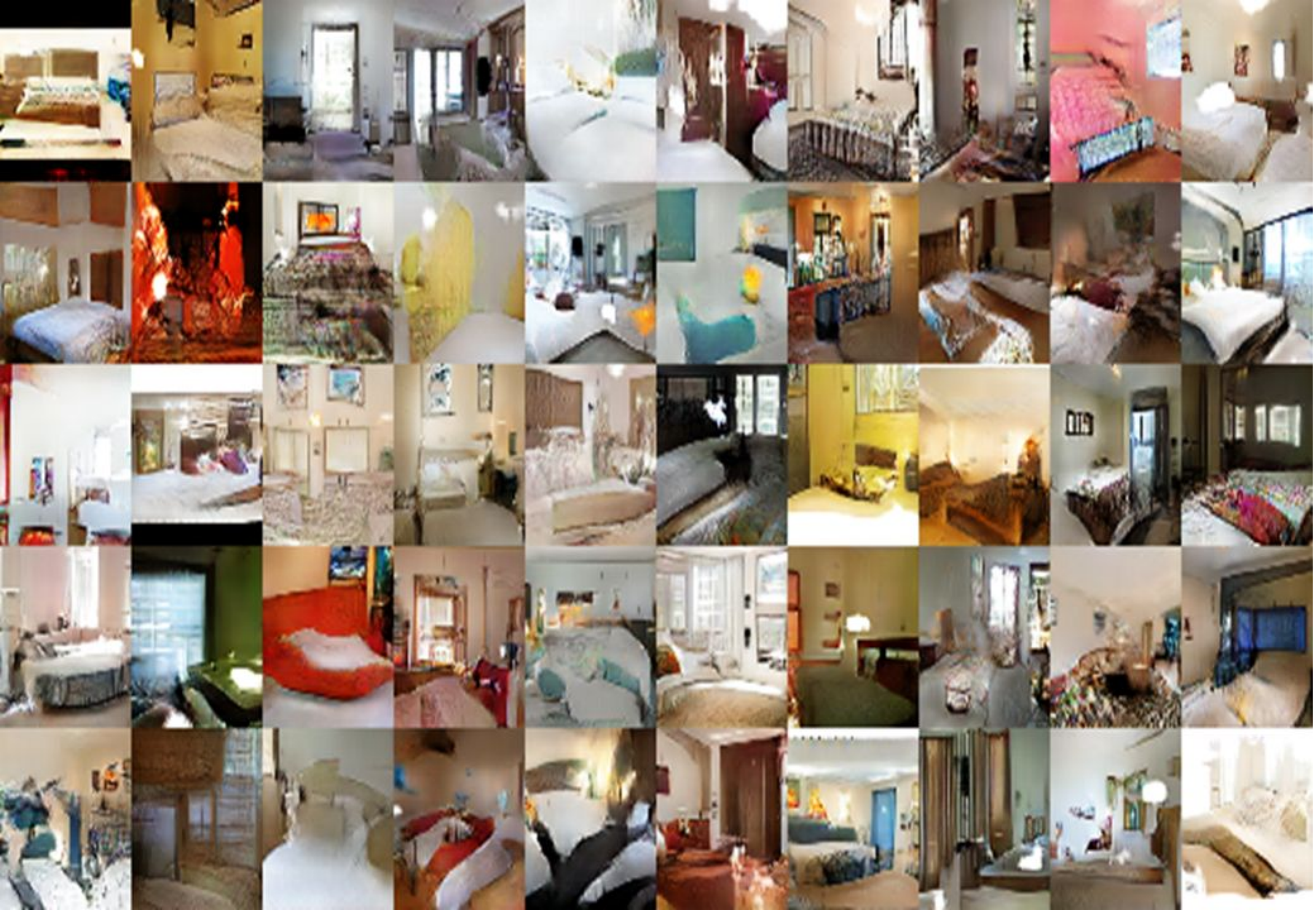


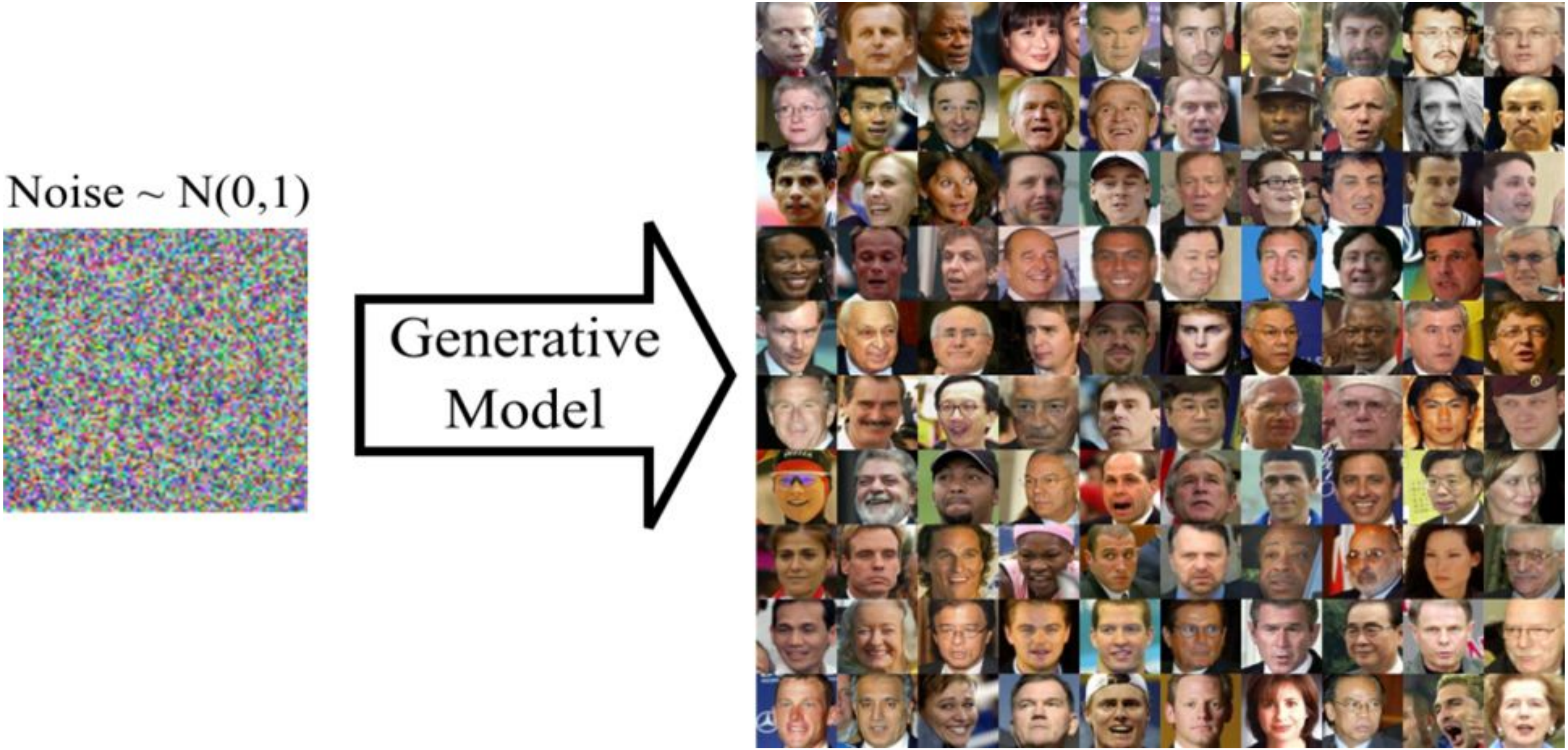

```
G_sample = generator(Z)
D_real, D_logit_real = discriminator(X)
D_fake, D_logit_fake = discriminator(G_sample)

D_loss = -tf.reduce_mean(tf.log(D_real) + tf.log(1. - D_fake))
G_loss = -tf.reduce_mean(tf.log(D_fake))
```

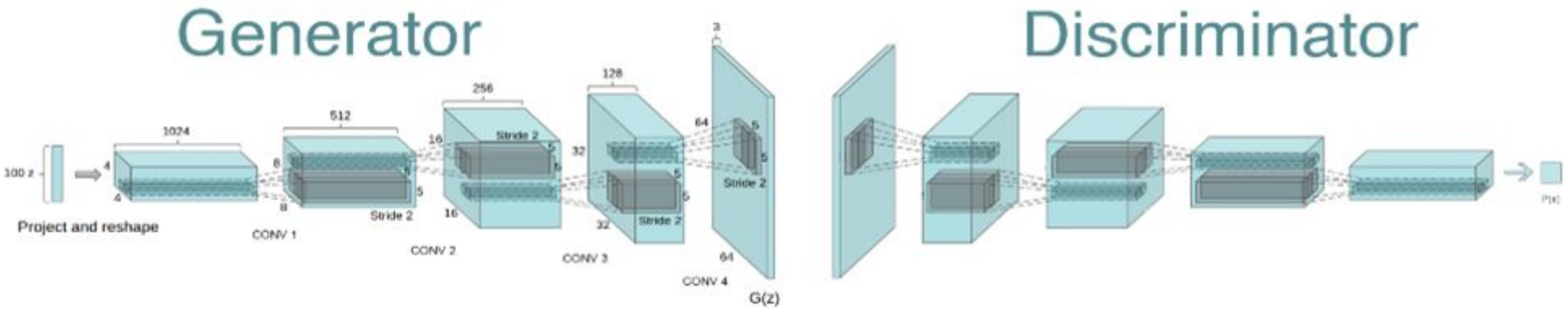


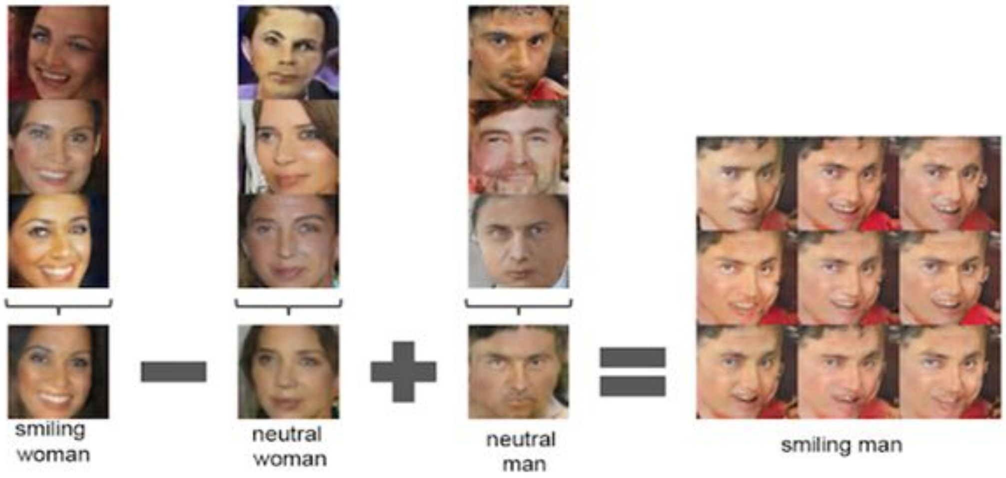
Trainable scope
X(images)
Discriminator
X
D_loss
Z(random vectors)
Z
100 z
Project and reshape
1024
4
4
512
8
8
Stride 2
CONV 1
256
16
16
Stride 2
5
5
CONV 2
128
32
32
5
5
Stride 2
CONV 3
3
64
5
5
Stride 2
64
CONV 4
G(z)
Generator


X(images)
Discriminator
X
G_loss
Trainable scope
Z(random vectors)
Z
100 z
Project and reshape
1024
4
4
512
8
8
Stride 2
CONV 1
256
16
16
Stride 2
5
5
CONV 2
128
32
32
Stride 2
CONV 3
3
64
64
Stride 2
CONV 4
G(z)
Generator

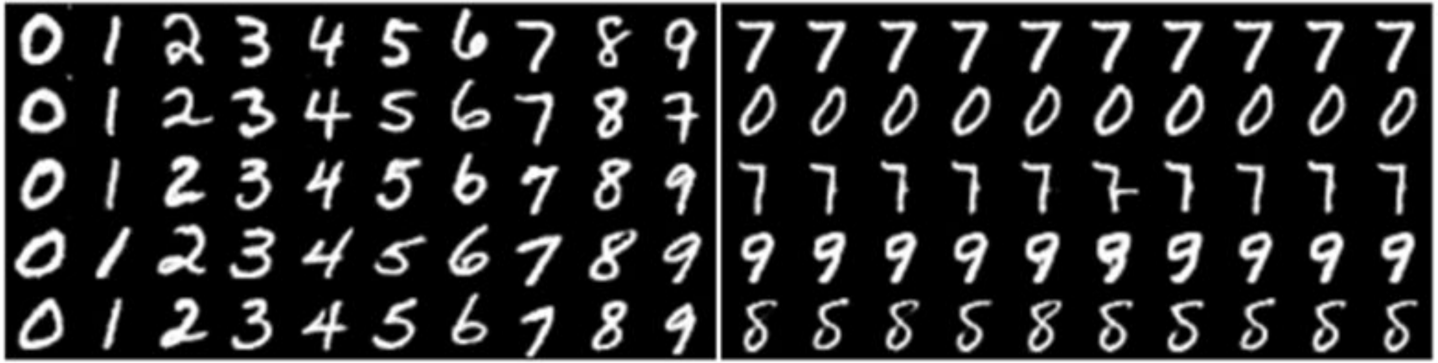



Noise ~ N(0,1)
Generative Model


Generator
Discriminator
100 z
Project and reshape
1024
512
256
128
Stride 2
CONV 1
CONV 2
CONV 3
CONV 4
64
3
G(z)

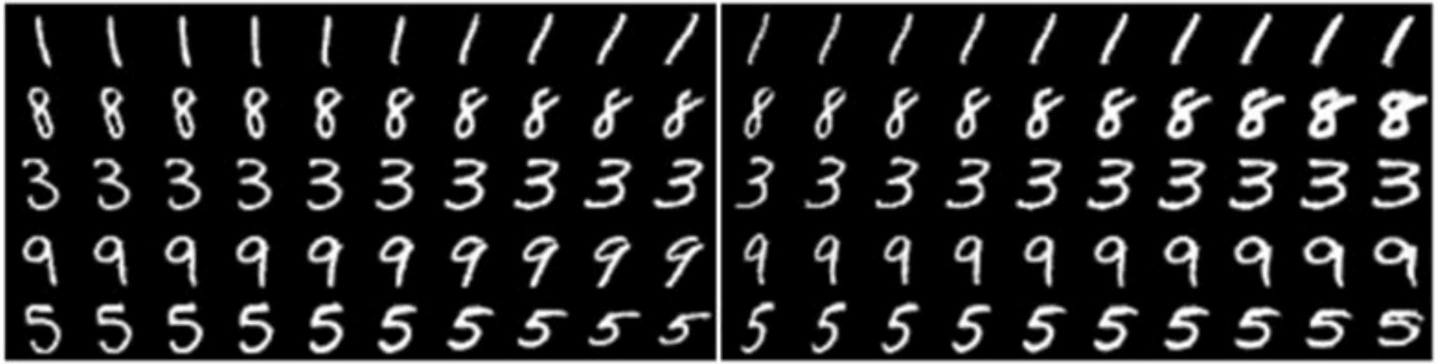

smiling woman
neutral woman
neutral man
smiling man

Disentangled features of GAN



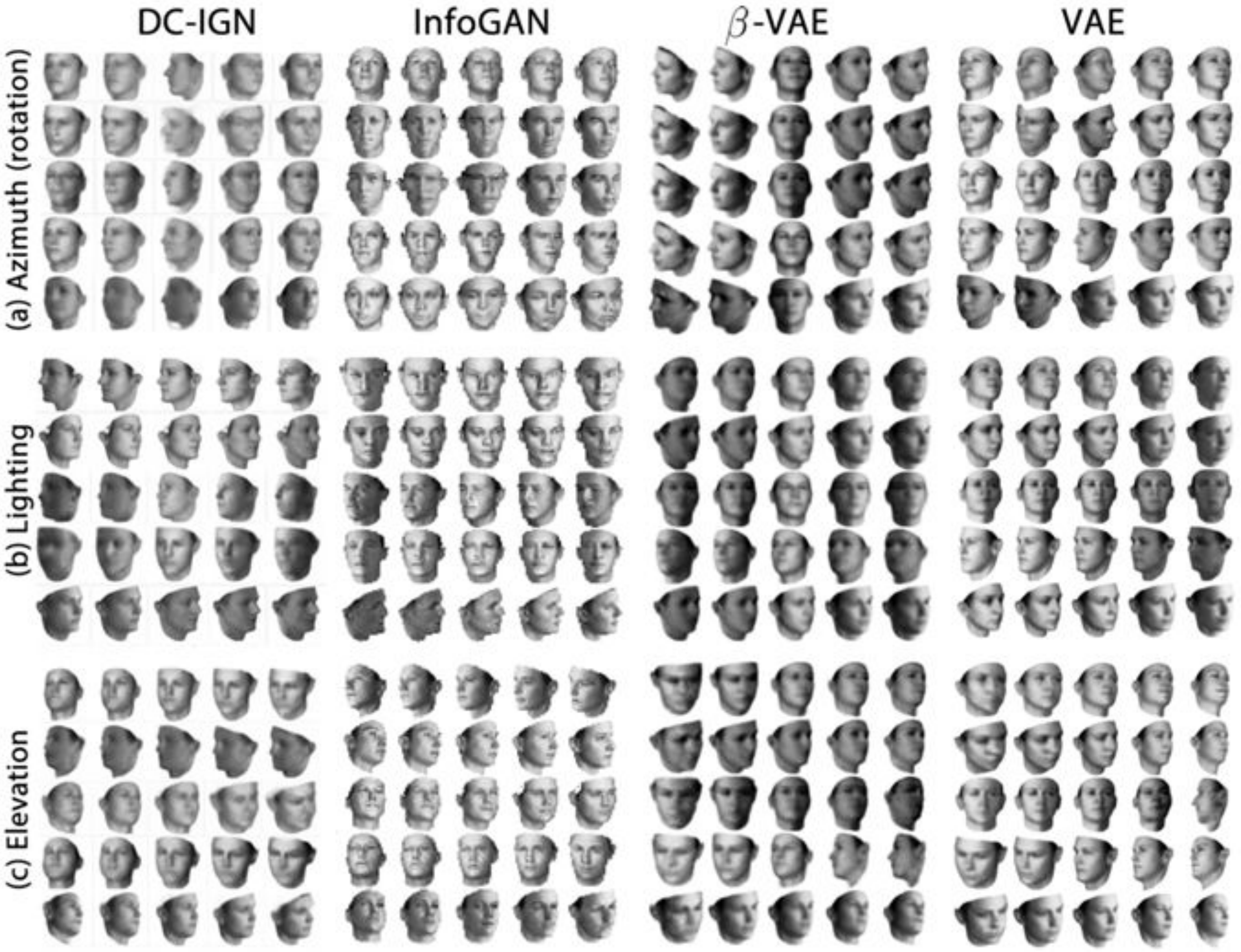
(a) Varying c_1 on InfoGAN (Digit type)

(b) Varying c_1 on regular GAN (No clear meaning)



(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)

(d) Varying c_3 from -2 to 2 on InfoGAN (Width)


DC-IGN
InfoGAN
β-VAE
VAE
(a) Azimuth (rotation)
(b) Lighting
(c) Elevation

Thank you!