# Aula 06 – Segmentação e Redes Generativas Adversárias

Prof. André Gustavo Hochuli

gustavo.hochuli@pucpr.br
aghochuli@ppgia.pucpr.br

# Tópicos

- Classificação vs Segmentação
  - Classificação
  - Detecção de Objetos
  - Segmentação

- Redes Generativas Adversárias
  - DCGAN
  - PIX2PIX

- Codificação

# Segmentação

# Classificação vs Segmentação

Is this a dog?

Image Classification

What is there in image and where?

Object Detection

Which pixels belong to which object?

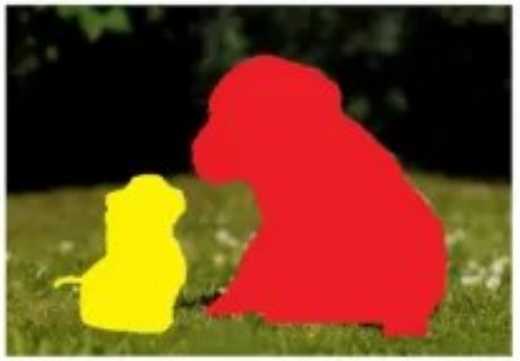
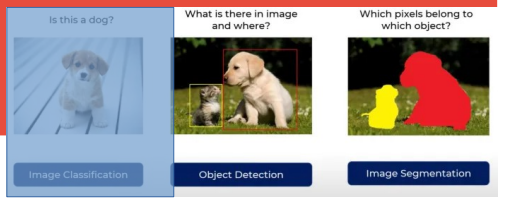
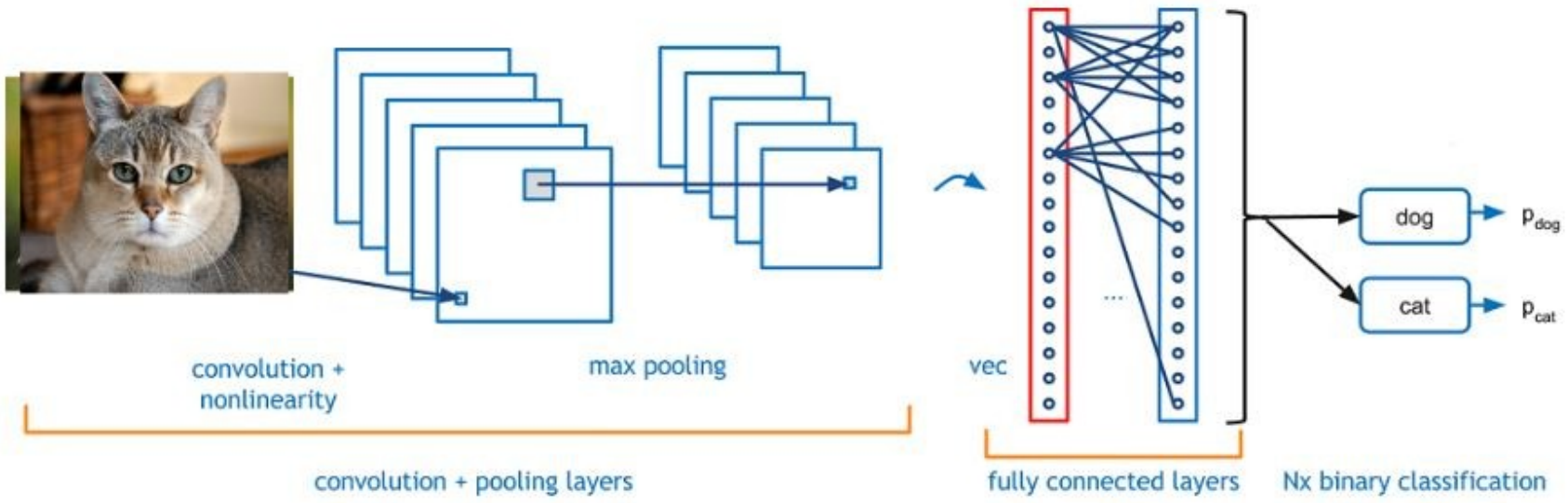


Image Segmentation

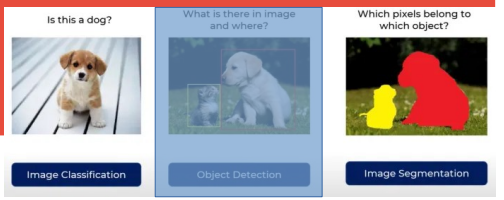
# Classificação

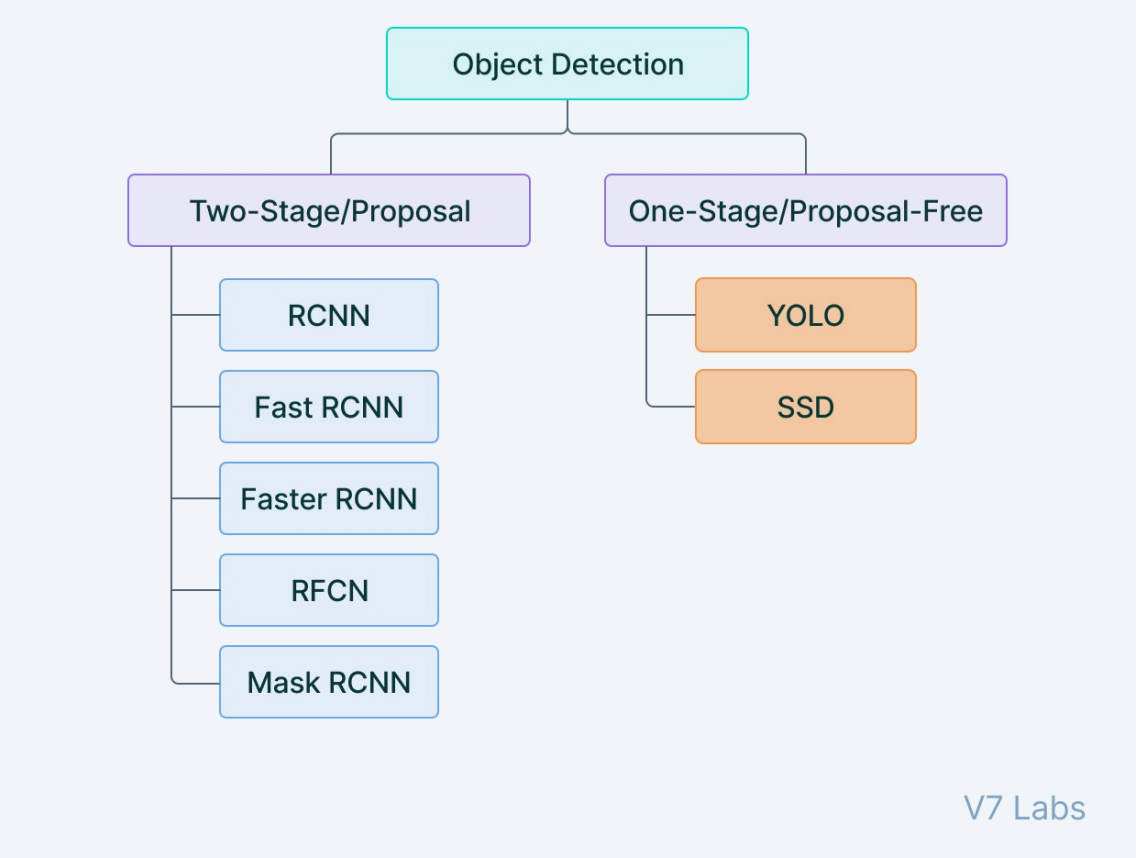


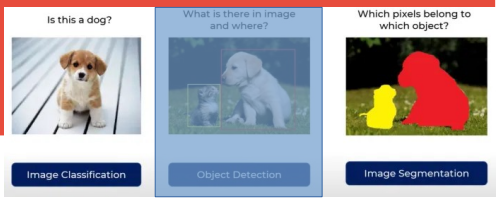
# Detecção de Objetos

# Detecção de Objetos

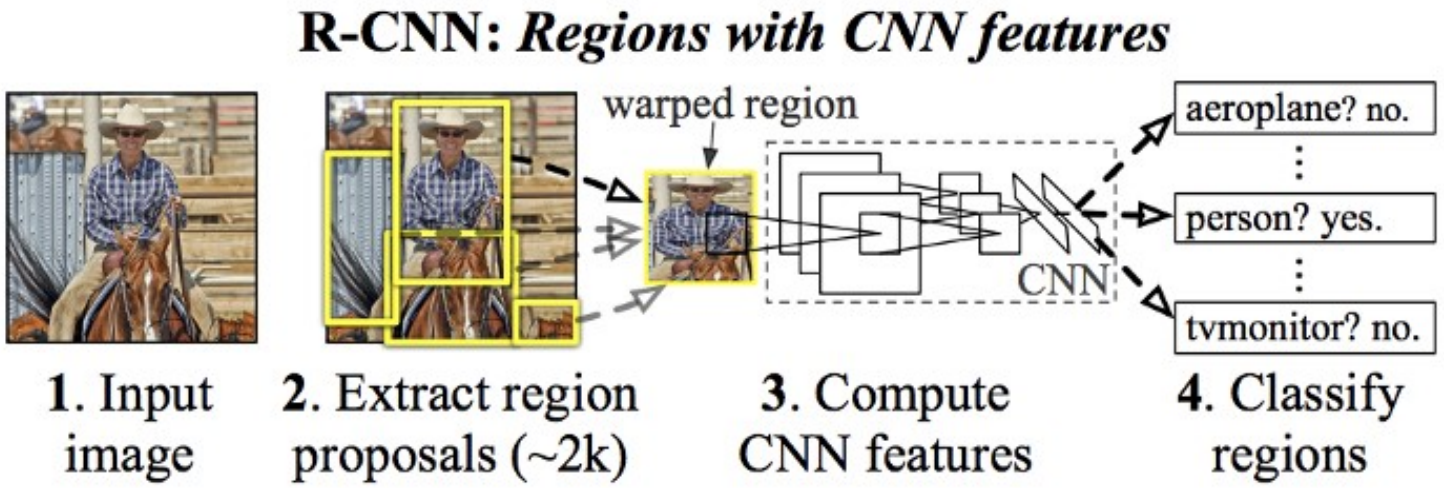


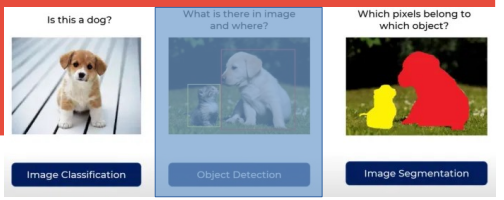
# Detecção de Objetos - RCNN



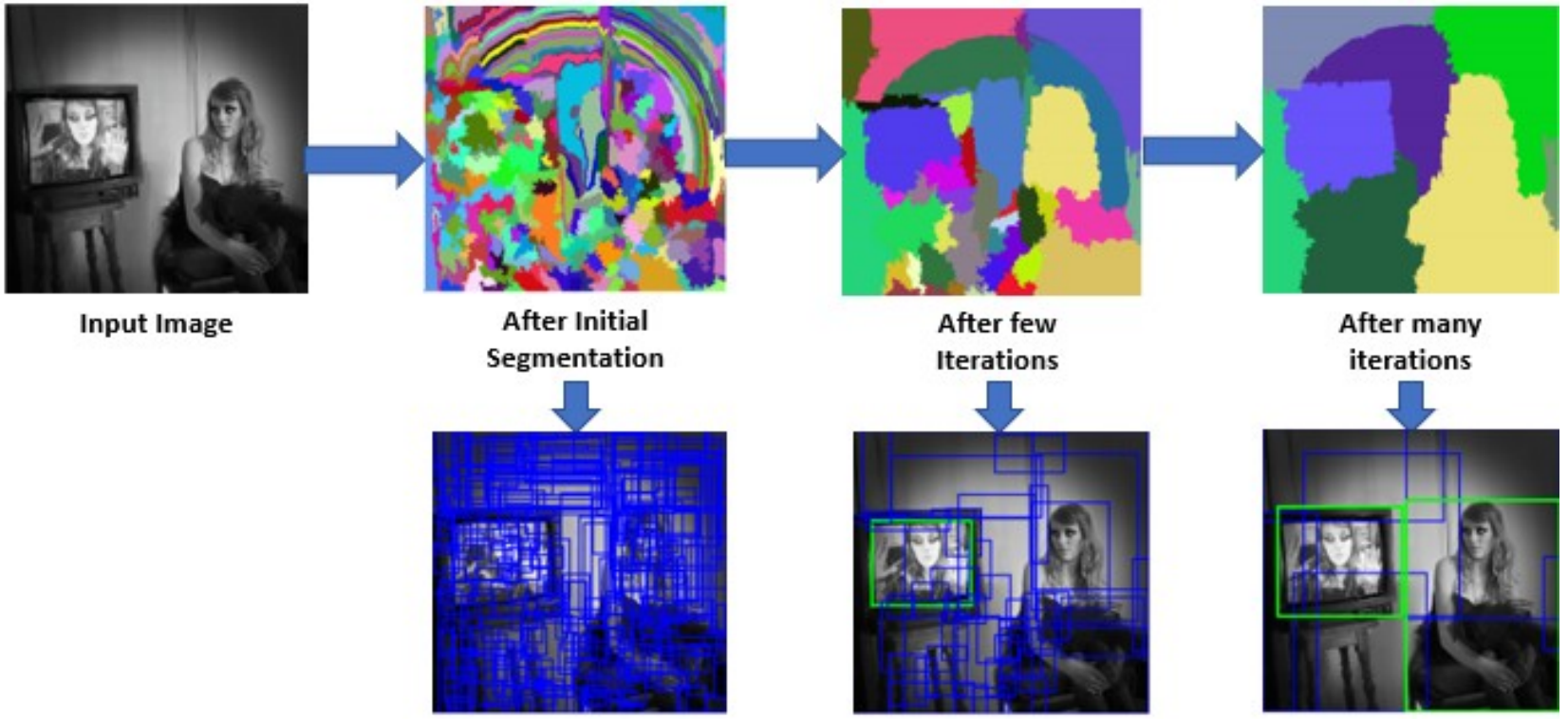

- Region Based Convolutional Neural Network (2014) - Ross Girshick
- Selective Search Algorithm (Region Proposal)
- CNN (Classification)



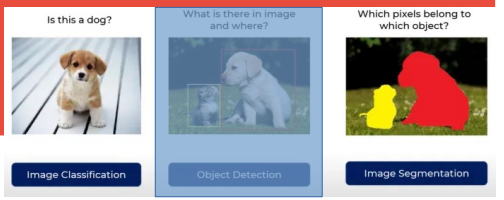
# Detecção de Objetos - RCNN



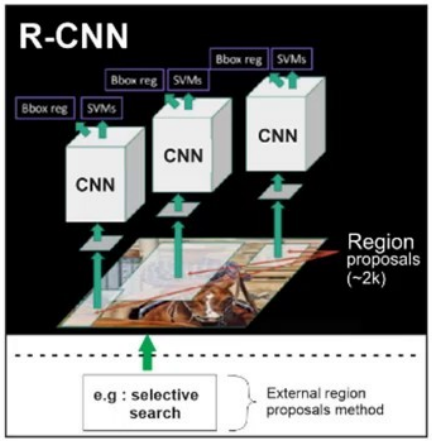

- Selective Search Algorithm (Region Proposal)

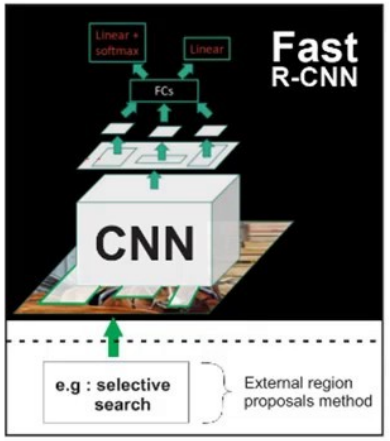


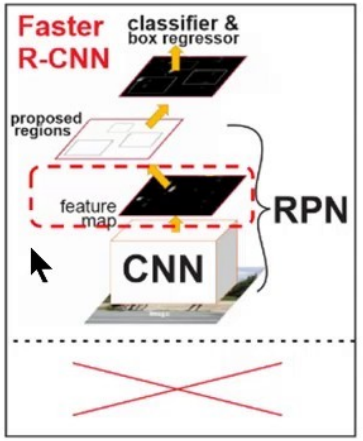
# Detecção de Objetos - RCNN




- R-CNN: Selective Search->CNN
- Fast: End-to-end (Sel. Search->ROI Pooling→FC)
- Faster: Region Proposal Network (RPN)

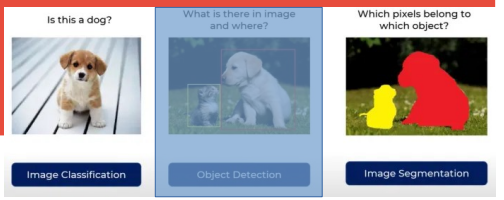


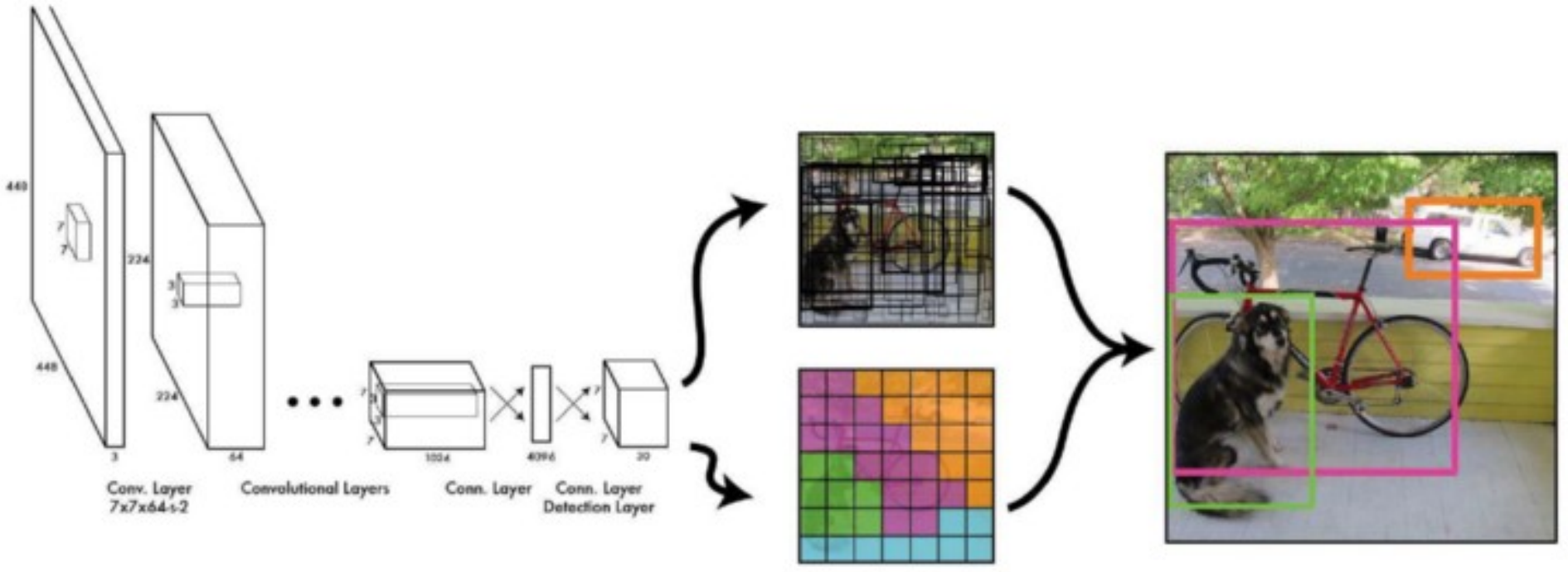




| | R-CNN | Fast R-CNN | Faster R-CNN |
|---|---|---|---|
| Test time per image | 50 seconds | 2 seconds | 0.2 seconds |
| Speed-up | 1x | 25x | 250x |
| mAP (VOC 2007) | 66.0% | 66.9% | 66.9% |

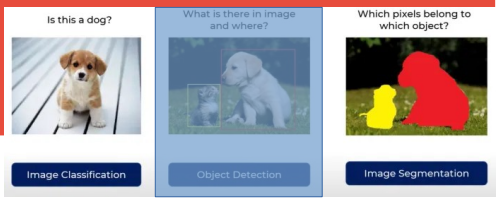
# Detecção de Objetos - Yolo

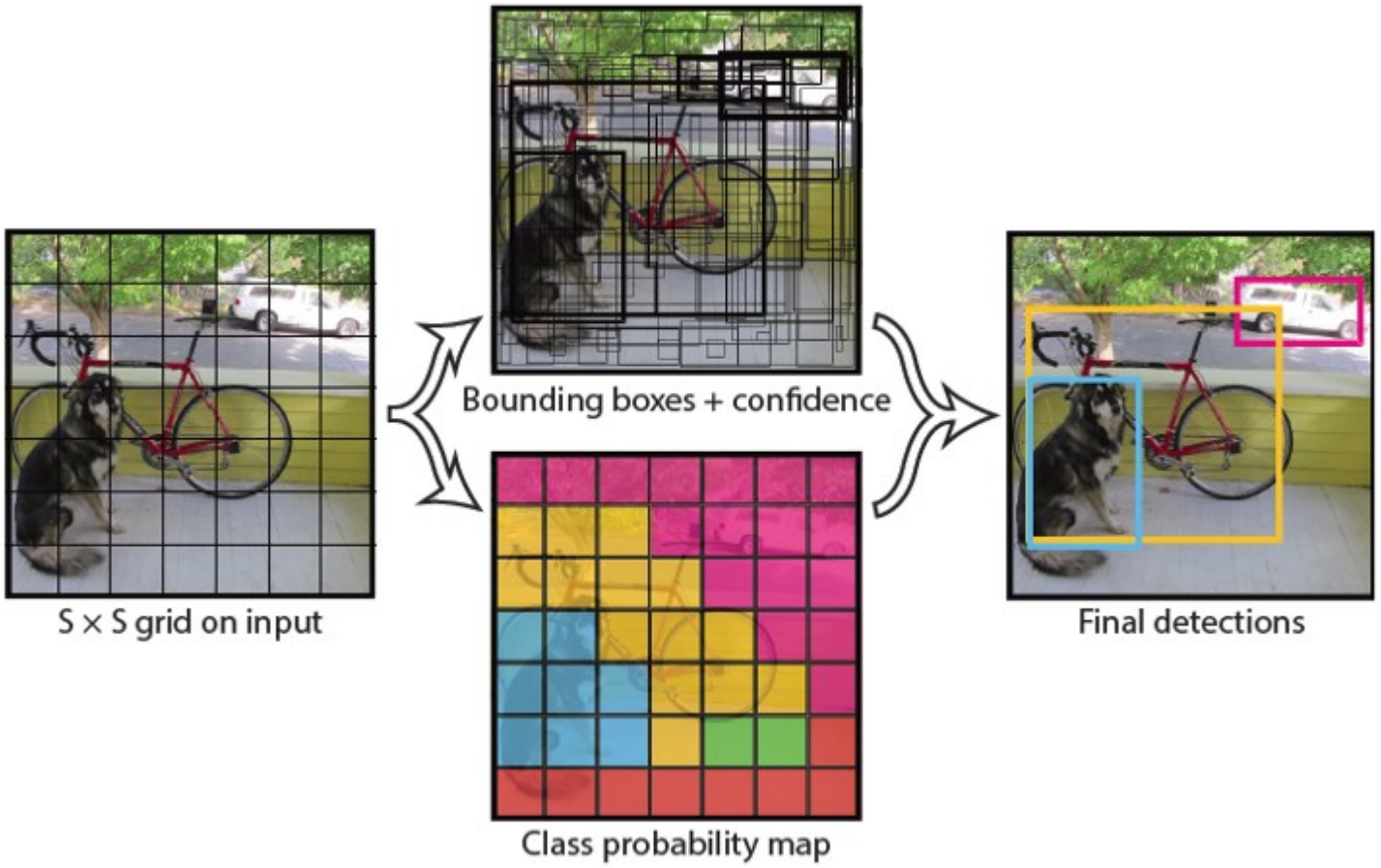



- You Look Once (2015)
  - Joseph Redmon / Ross Girshick

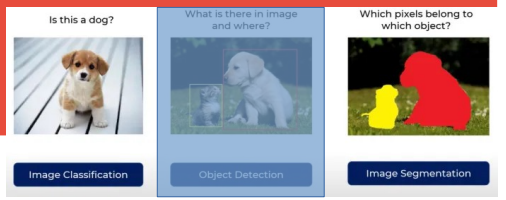


# Detecção de Objetos - Yolo



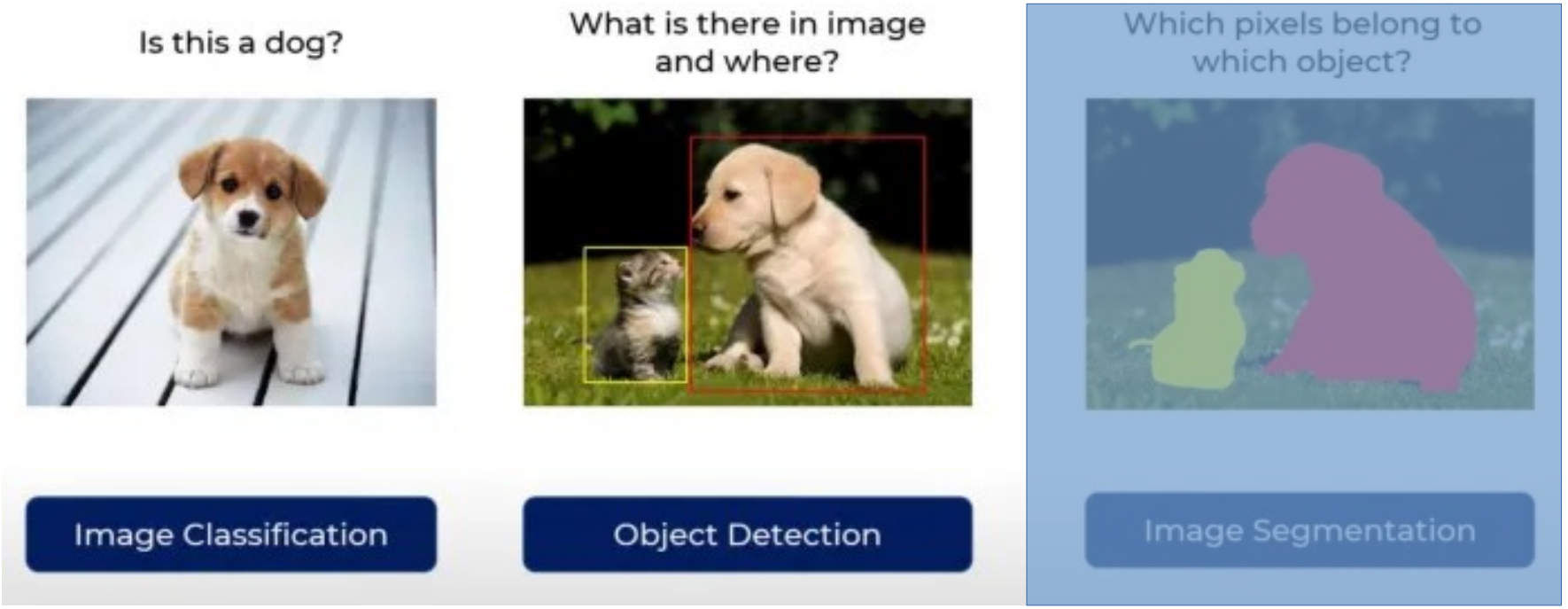



# Let’s Code



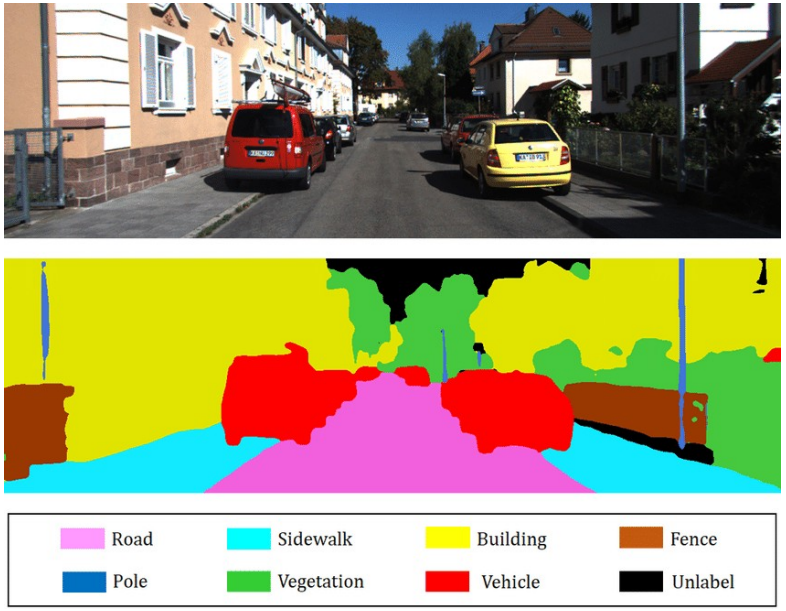

- YOLO Inference
  - COLAB *[LINK]*
  - CPU (local)
    - https://github.com/Asadullah-Dal17/yolov4-opencv-python

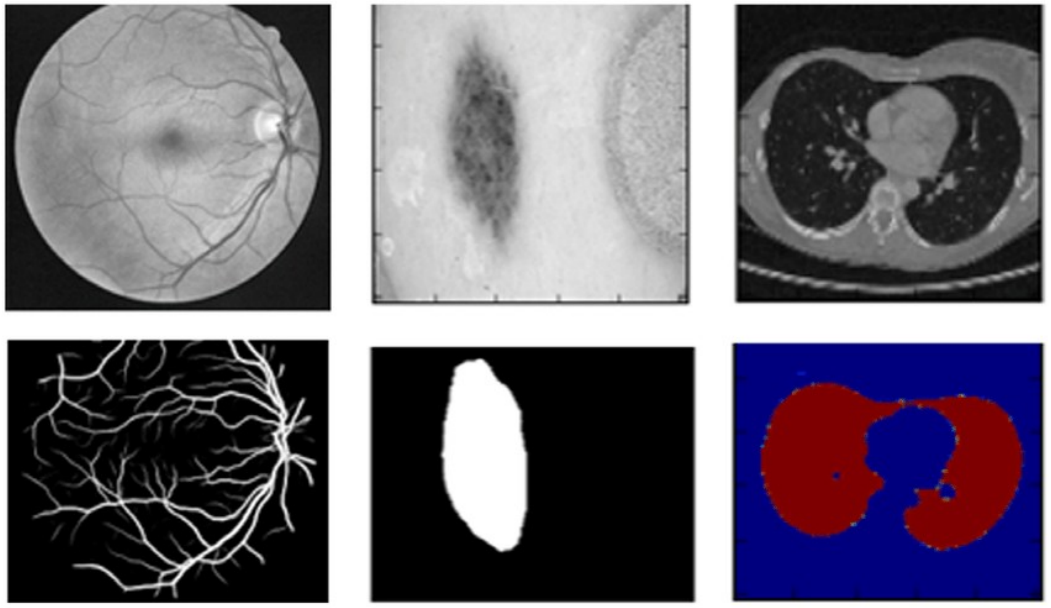
# Segmentação



# Segmentação

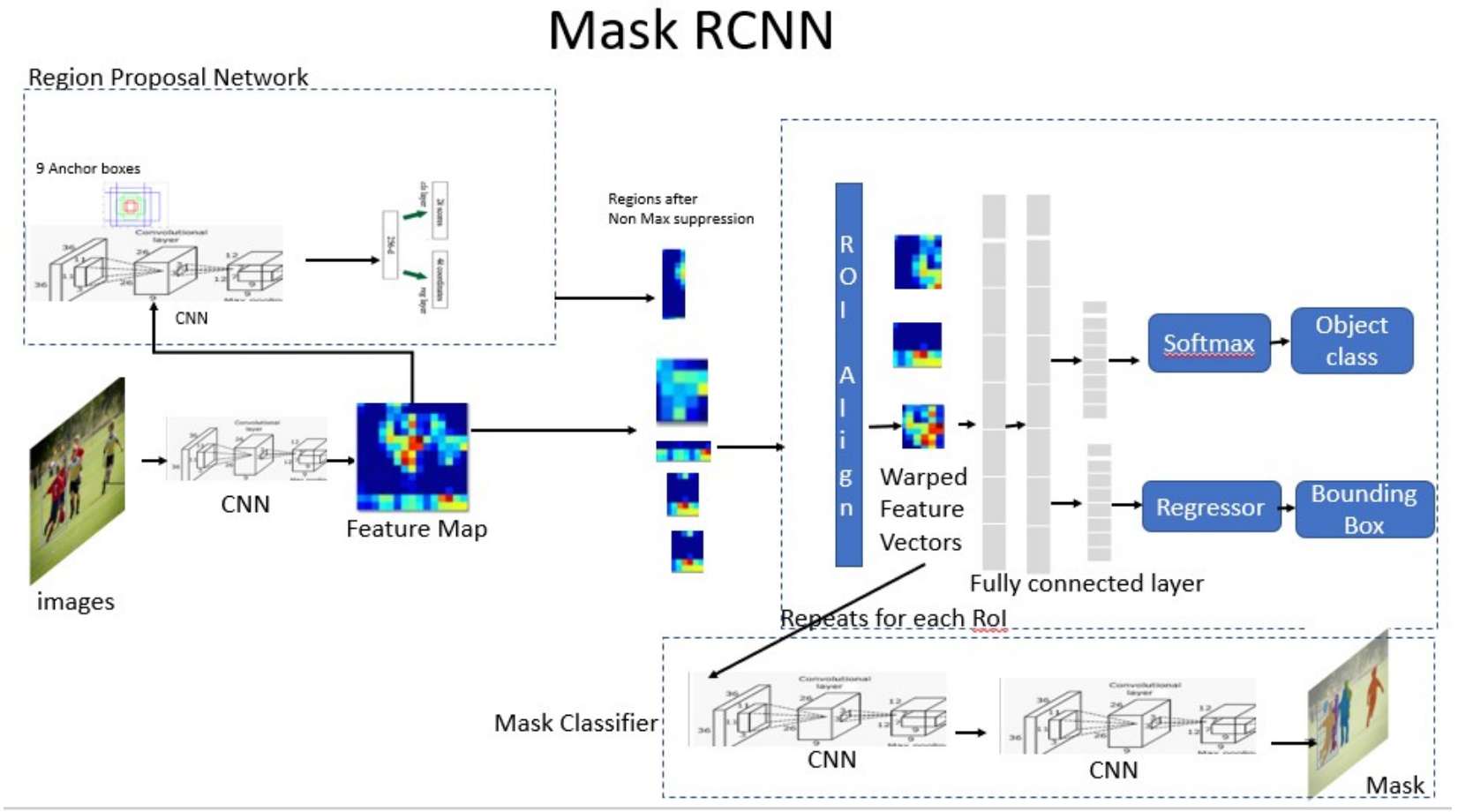

- Classificação a nível de pixel



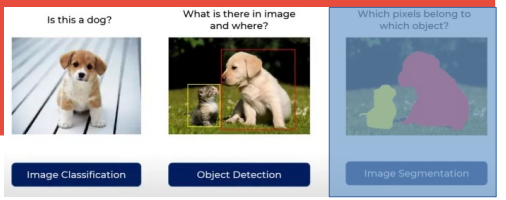



# Segmentação – Mask RCNN

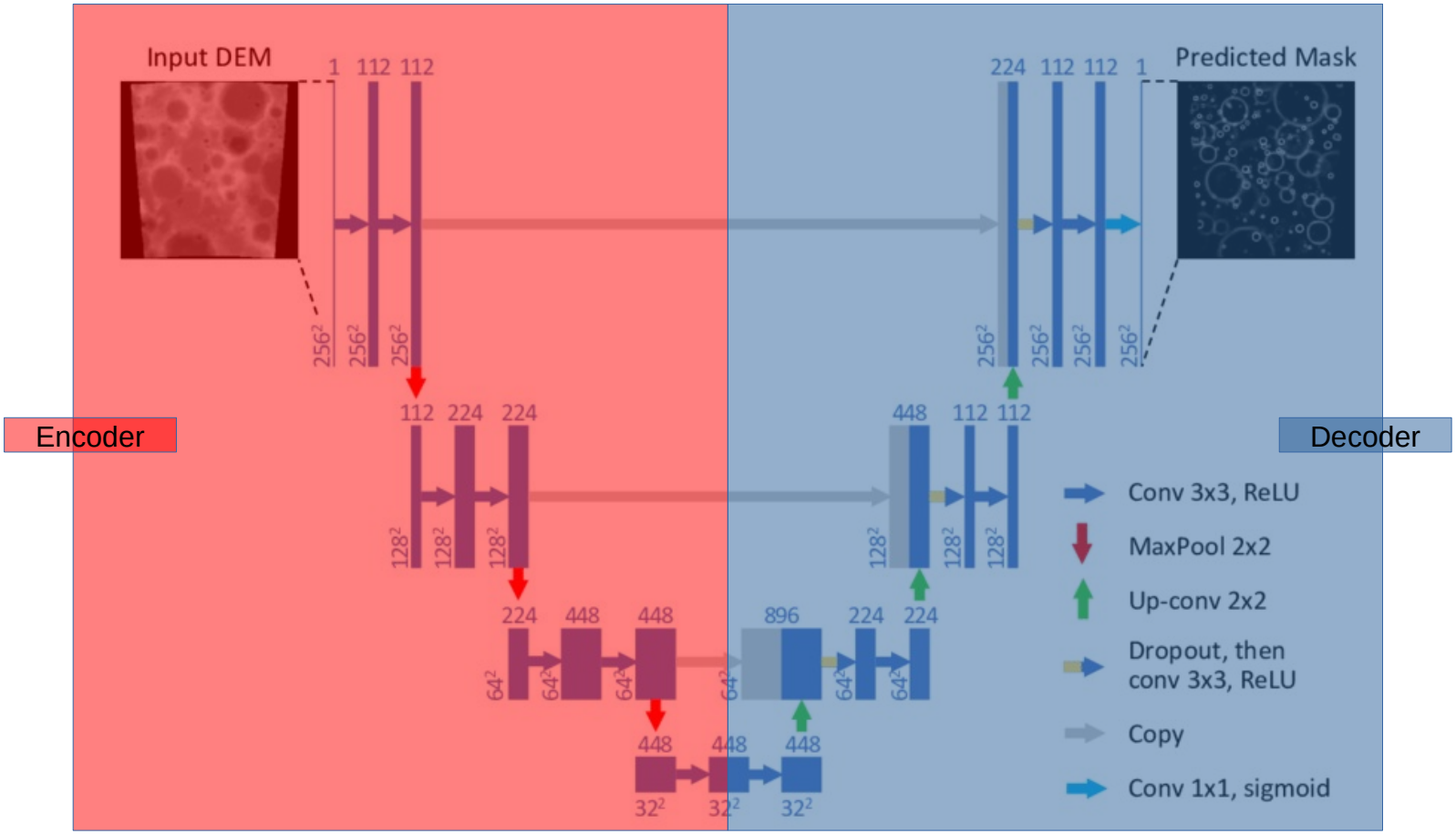

- Faster R-CNN with Binary Mask (2017)



# Segmentação - UNET




- U-Net (Encoder and Decoder)



# Let’s Code


- U-NET (Treino e Inferência)
  - *[LINK]*

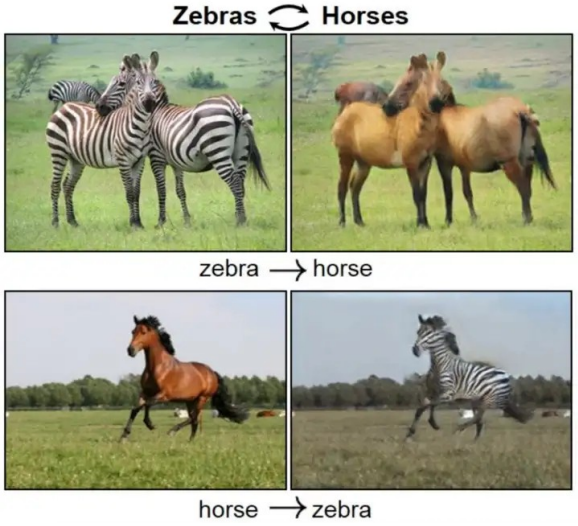
# Redes Generativas Adversárias

# Tópicos

- DCGAN
- PIX2PIX
- Prática

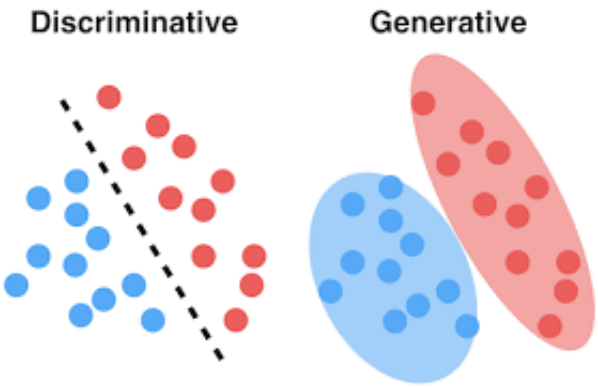
# Deep Fakes

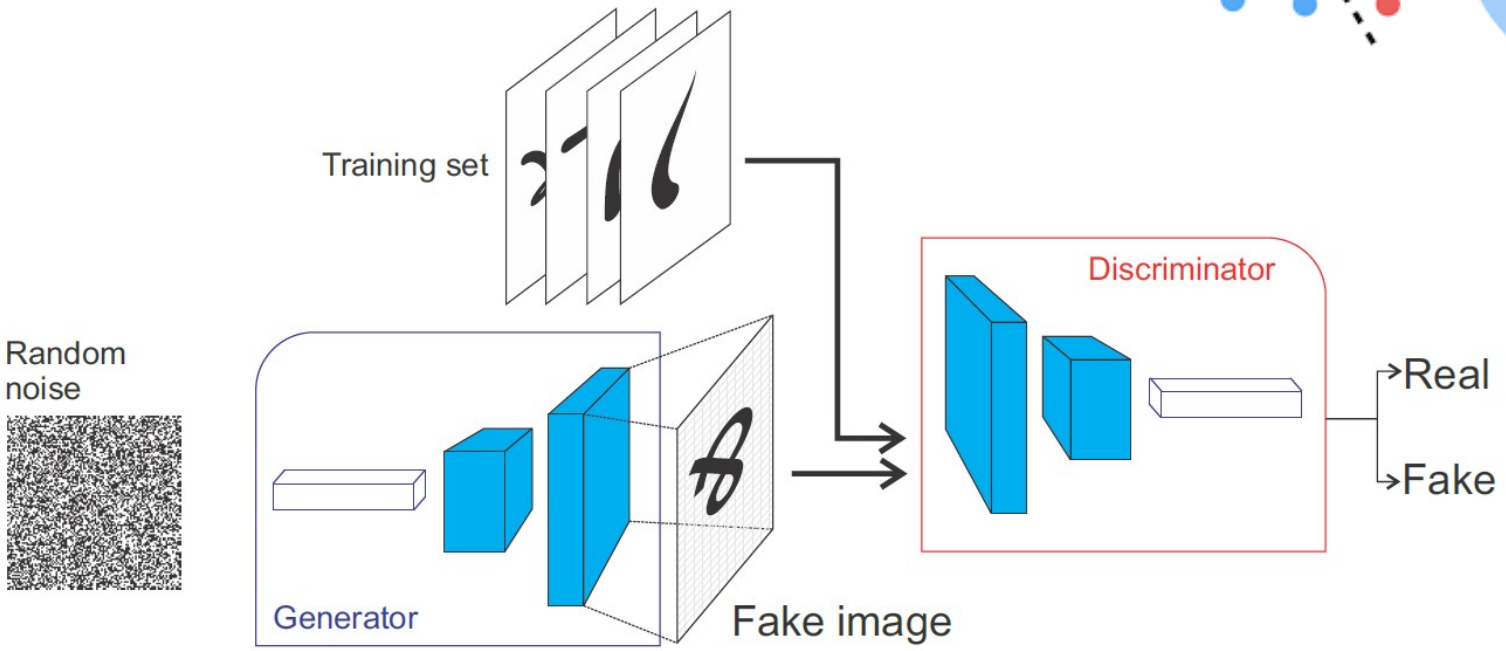
- Generalização: Dados Sintéticos gerados a partir do aprendizado da distribuição real do dado
- Aplicações
  - Filmes (Cenários Sintéticos)
  - Fotografia (Estimação de Pose, Coloração Artificial, Redução de Ruído)
  - Troca de Contexto (Zebra->Cavalo)
  - …….




# Generative Adversarial Networks (GAN's)

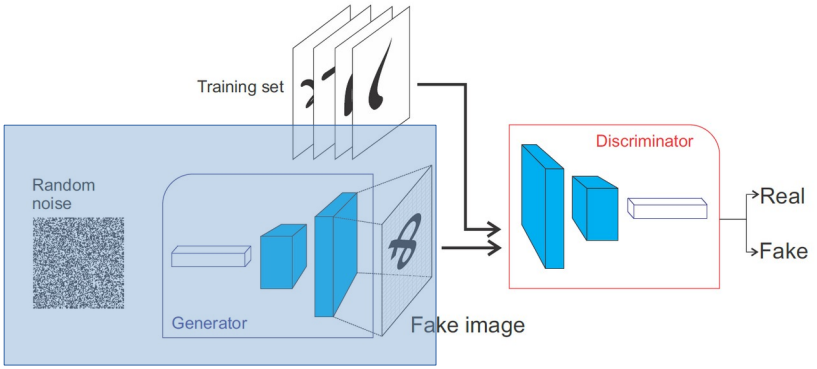
- Gerador: Ruído → Data
- Discriminador: Classificação

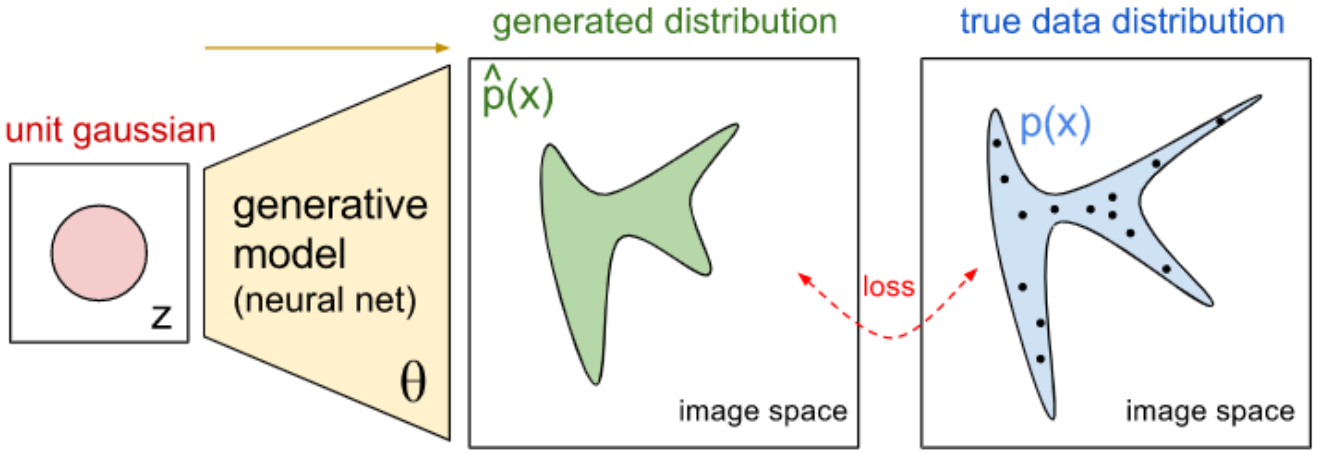


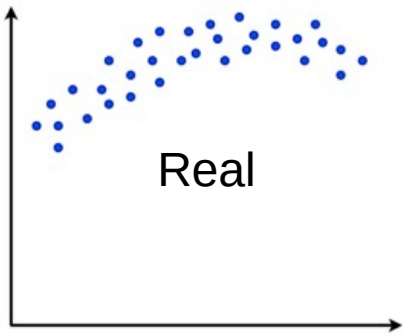
# Modelo Generativo

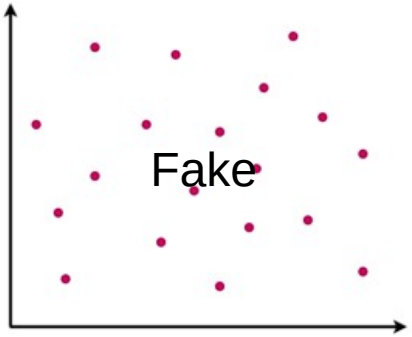
- Aprende a distribuição do dado

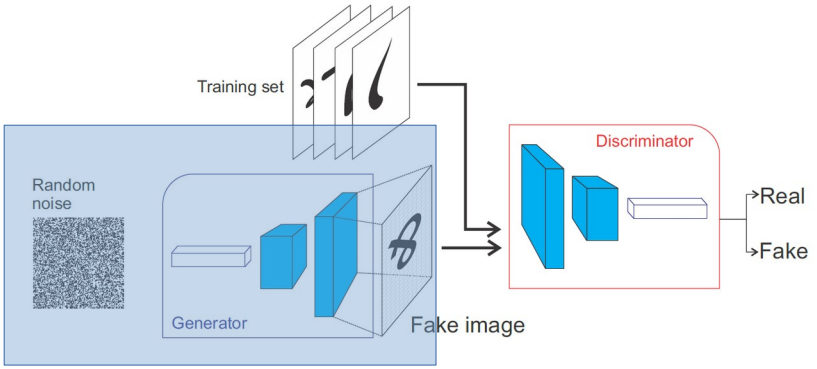


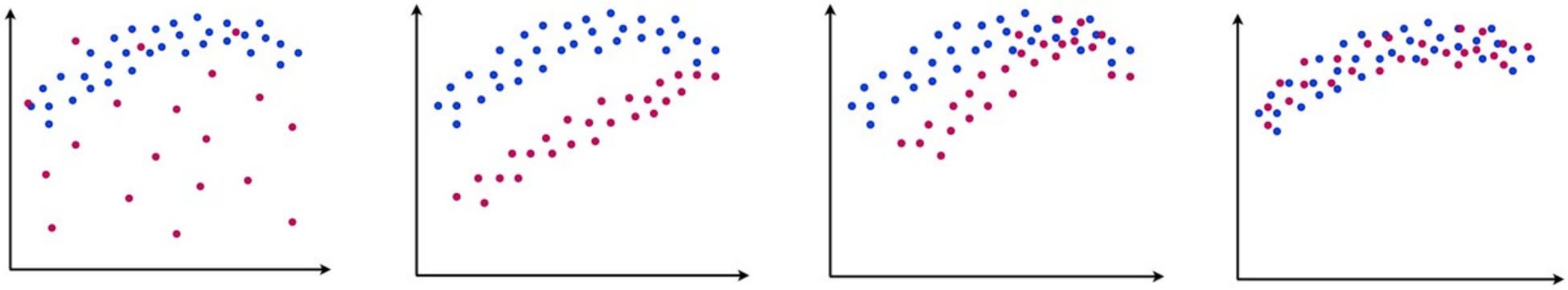



# Modelo Generativo



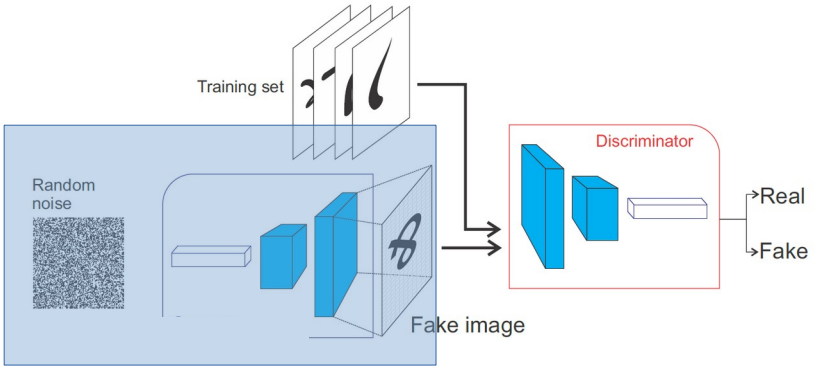



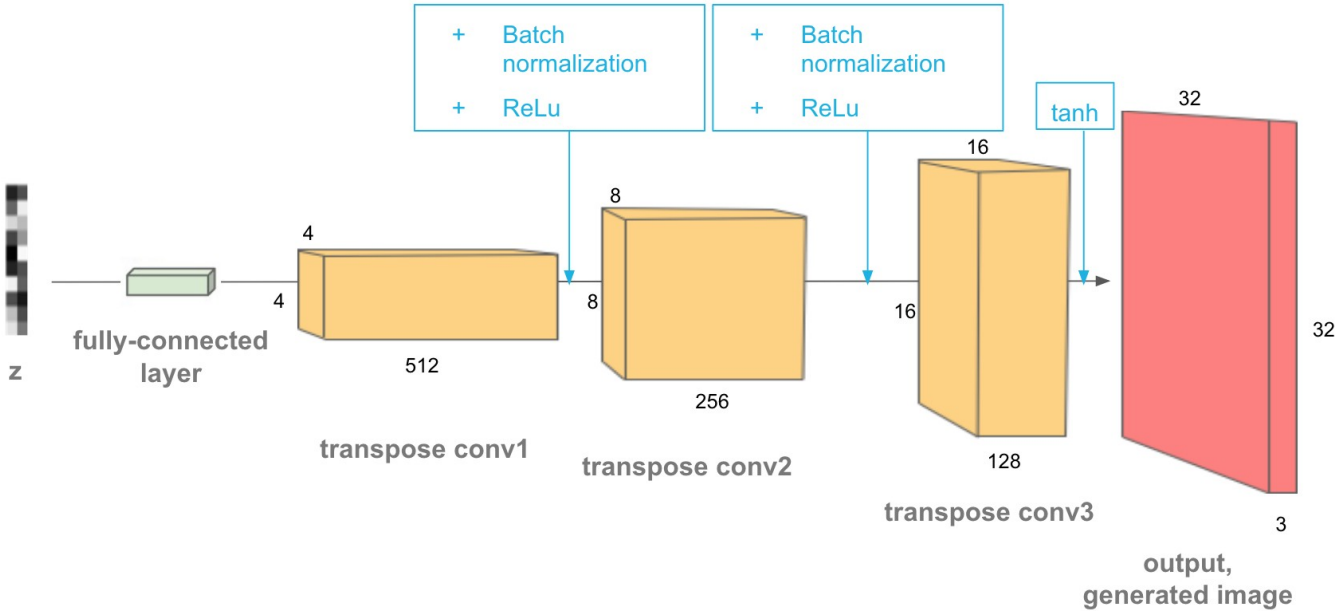



Training

# Modelo Generativo Profundo

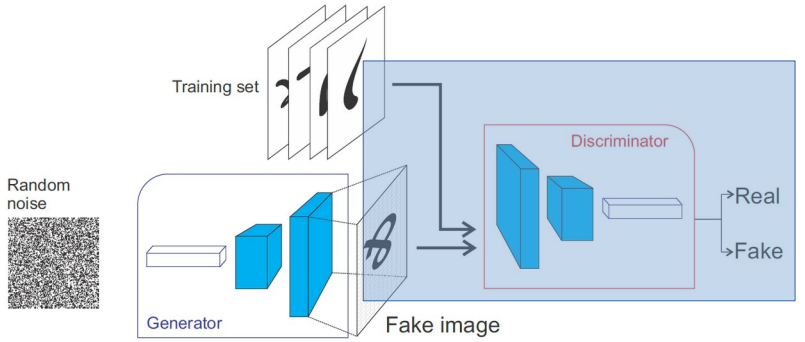
- Camadas Des-Convolucionais (upsampling)
  - Ruído → Imagens Sintéticas

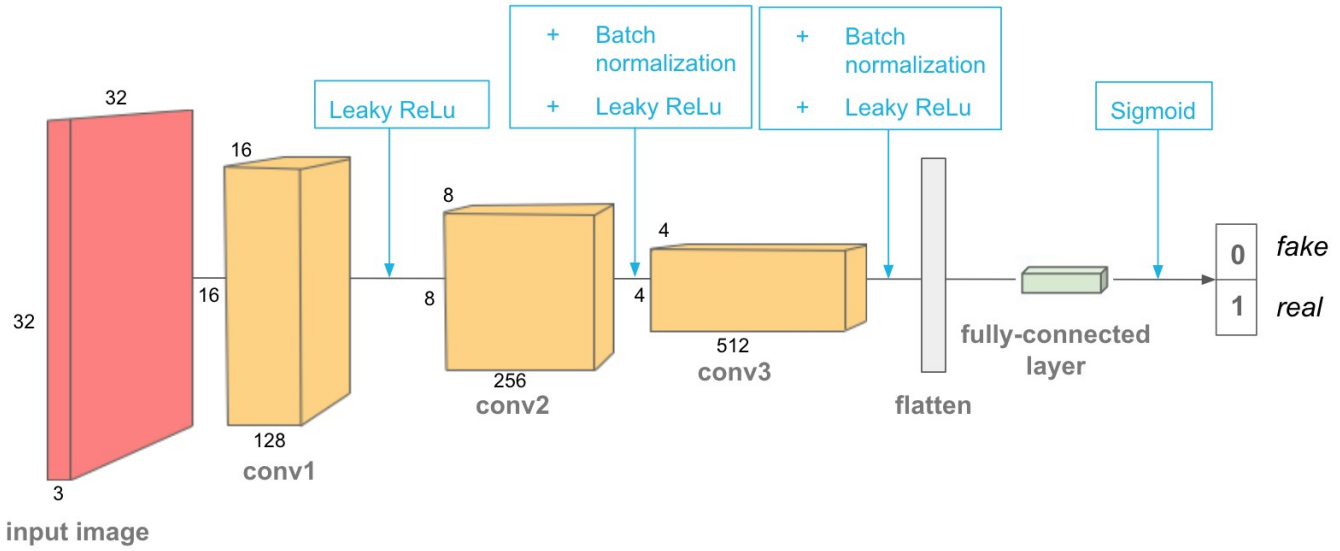


# Modelo Discriminante

- Classificação: Falso ou Real ?
  - CNN

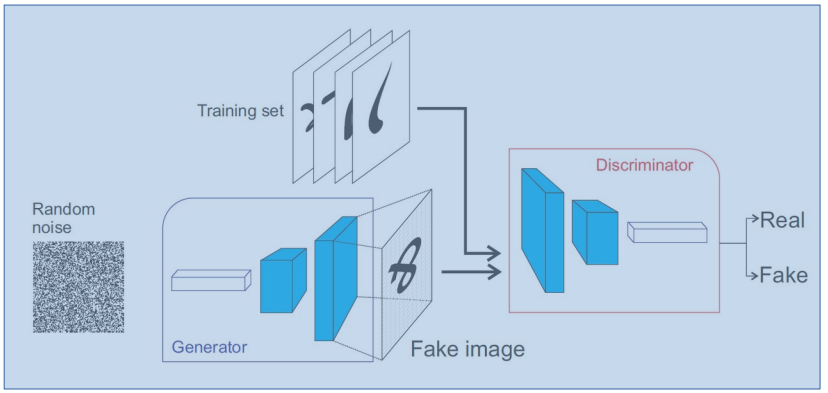


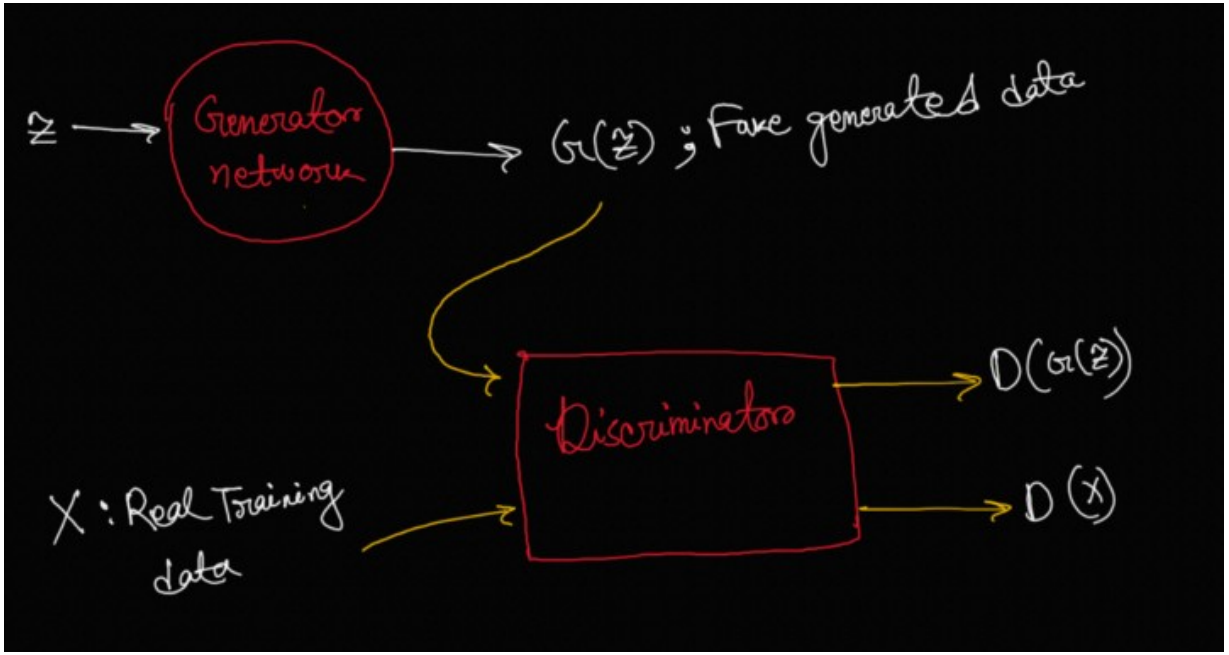



# Treinamento Adversário

- Erro Adversário (Min-Max)

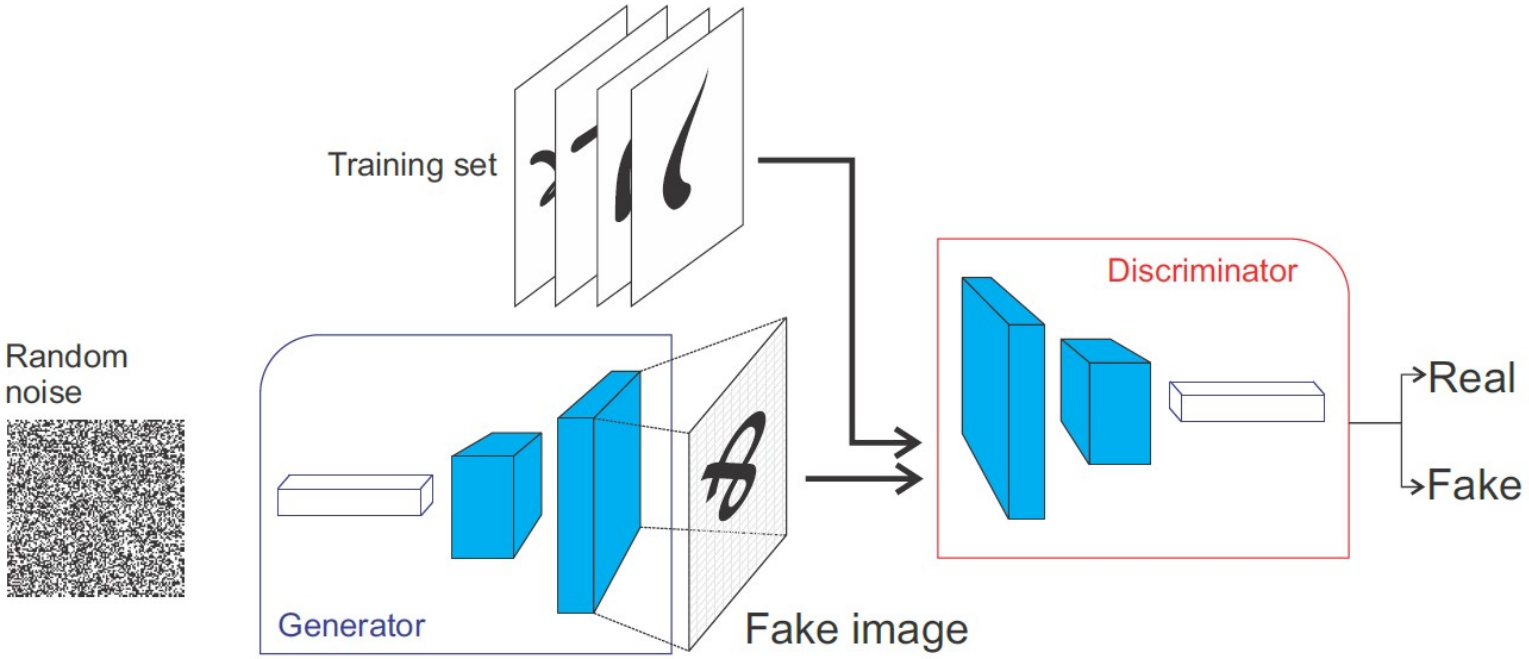
$$\min_G \max_D V(D,G) = \mathbb{E}_{x \sim p_{data}}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$



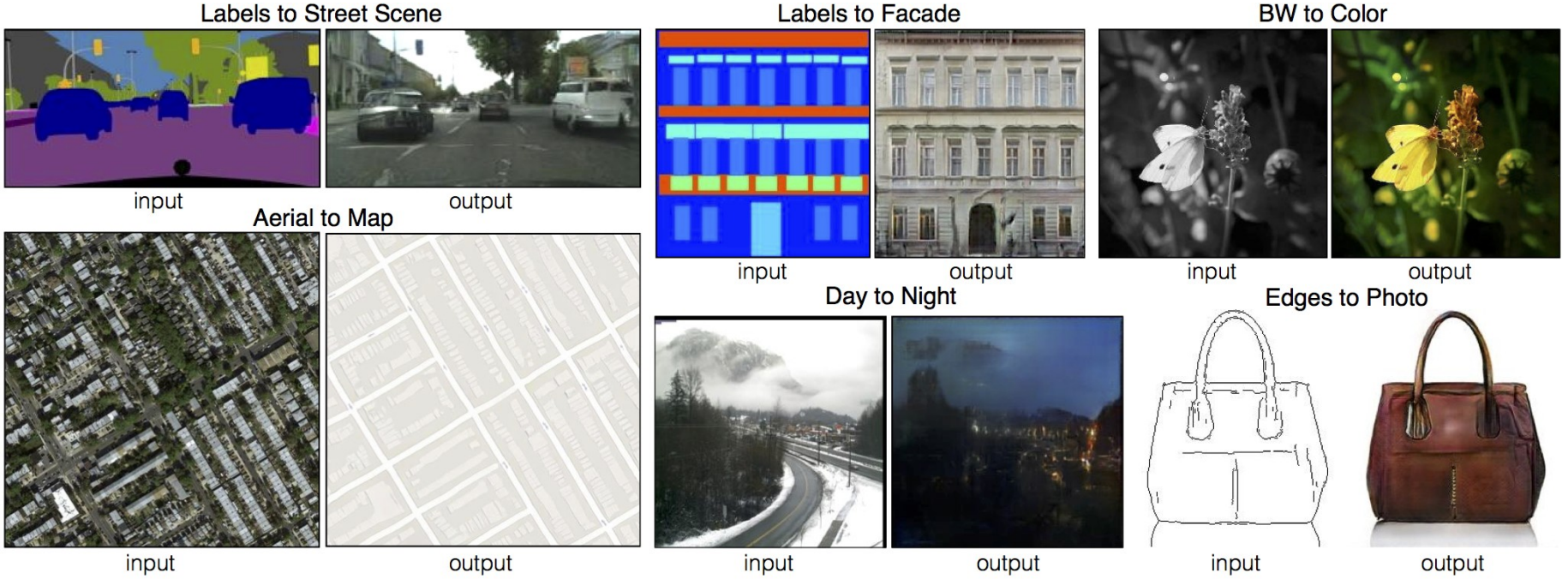
# Let’s Code

- LINK: Lecture_12-DCGAN.ipynb



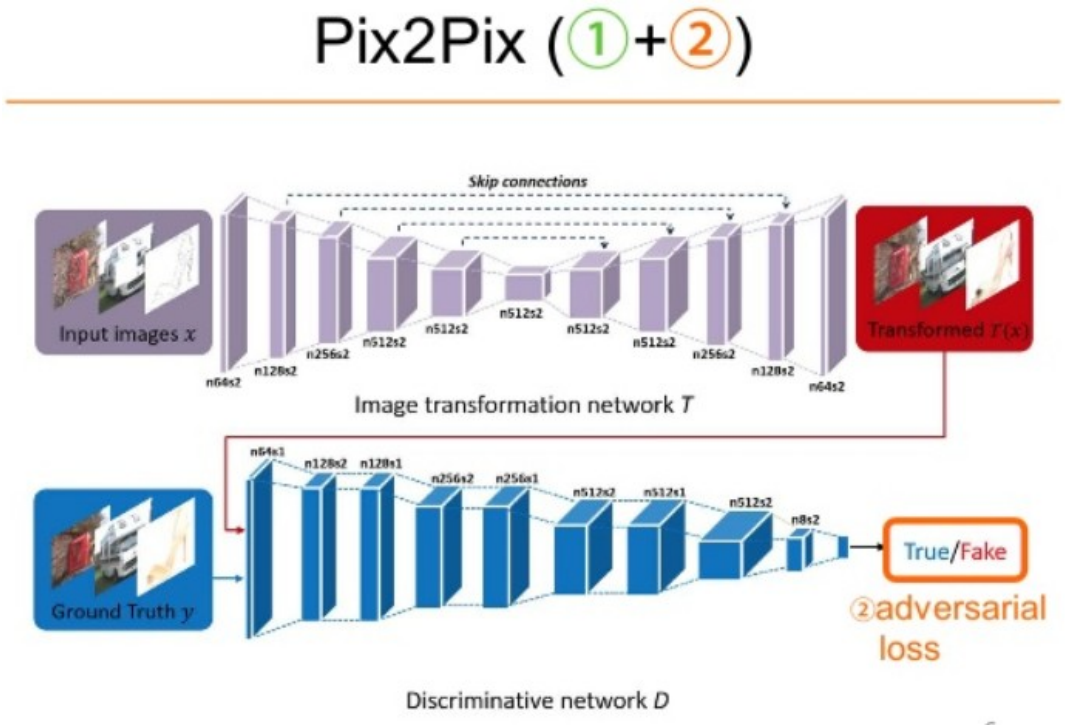
# Pix2Pix

- Transformação de Contexto (Image Translation)



# Pix2Pix

- Modelo Generativo: Arquitetura Encoder-Decoder (i.e U-Net)
- Base de Dados Pareada (Origem->Destino)




# Let’s Code

- LINK: Lecture_12-pix2pix.ipynb