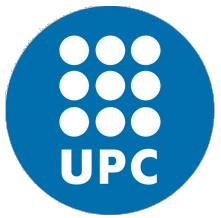
DEEP LEARNING FOR COMPUTER VISION

TEAM 2 - Colorization with Conditional GANs



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Itziar Sagastiberri
João Ramos
Mireia Gartzia

OVERVIEW

Objectives

Datasets

Architecture

Training

Results

Conclusions and Lessons Learned

OBJECTIVES

Understanding and testing Conditional GANs

Image colorization using Conditional GANs

Transfer learning on a new dataset

DATASETS

- Facades: http://cmp.felk.cvut.cz/~tylecr1/facade/

  # training samples: 200

  # testing samples: 178

- Cat_Dataset: https://www.kaggle.com/crawford/cat-dataset

  # training samples: 198

  # testing samples: 104

ARCHITECTURE

GANs (Generative Adversarial Networks).

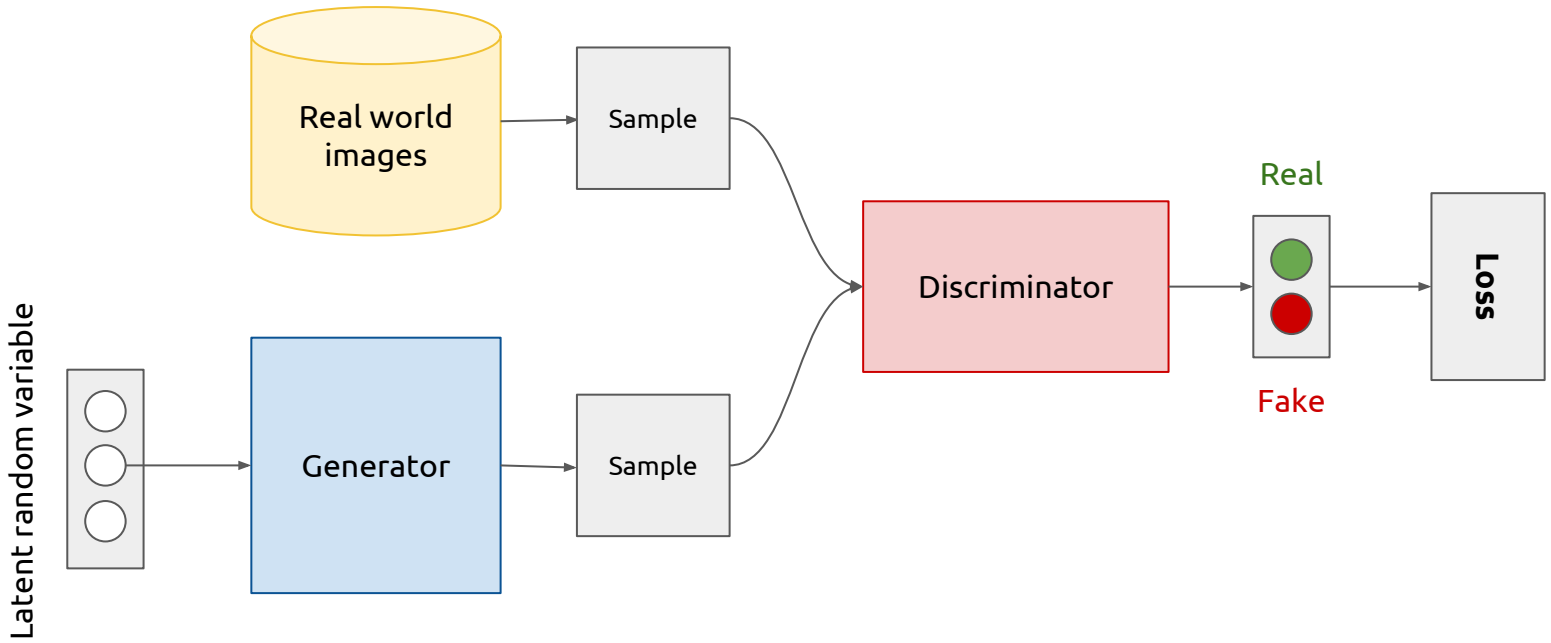
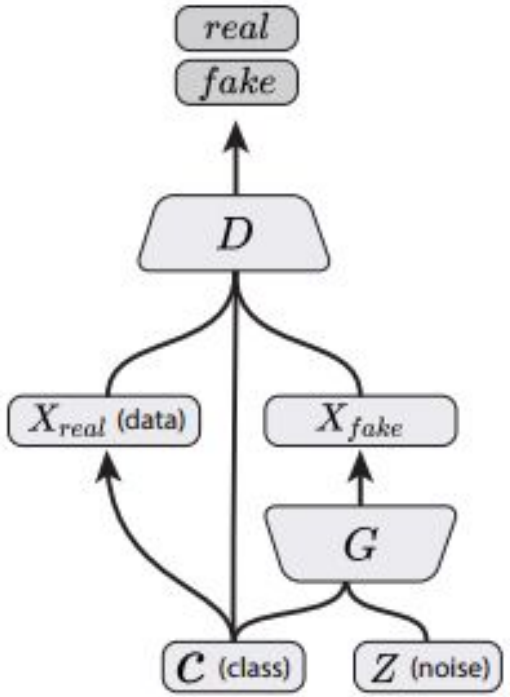


Image taken from Kevin McGuinnes's Generative Models slides.

ARCHITECTURE

Conditional GAN.

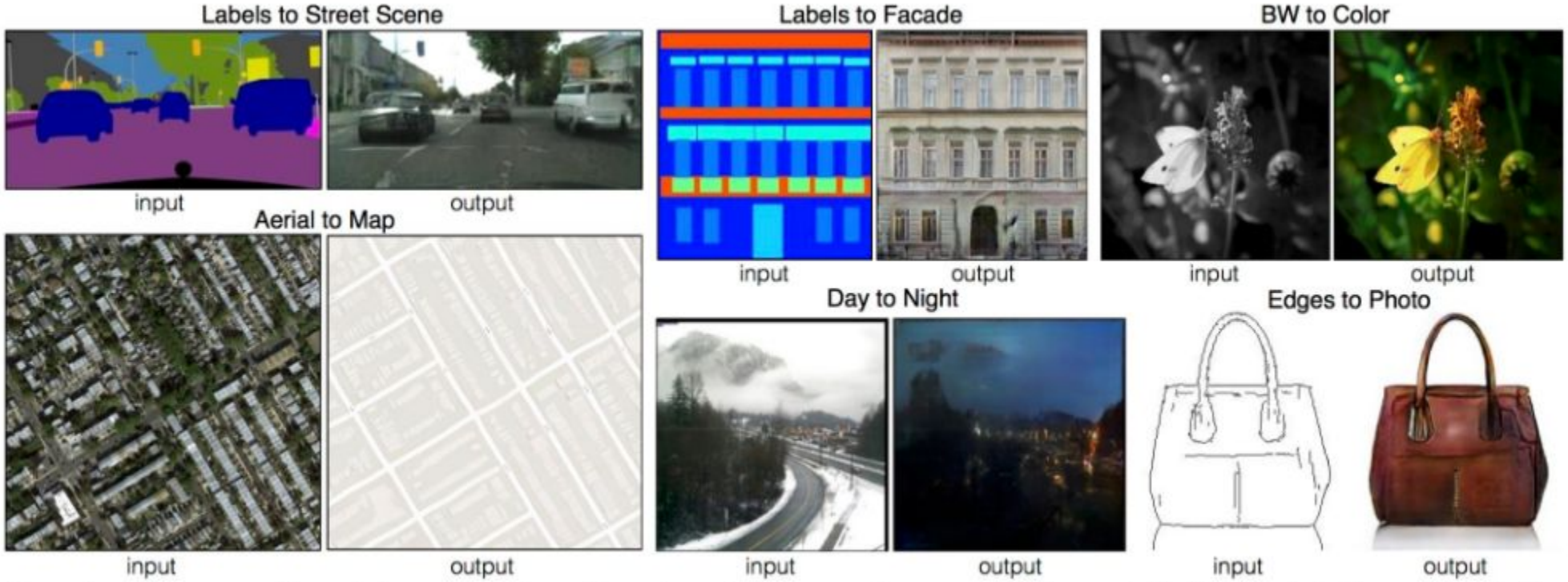
- Unlike an unconditional GAN, the generator has an input condition (image, label, …)



Conditional GAN
(Mirza & Osindero, 2014)

ARCHITECTURE

Pix2pix - Image to Image Translation



Example results on several image-to-image translation problems. In each case we use the same architecture and objective, simply training on different data.

ARCHITECTURE

- Our Generator has 6 blocks, each one composed by:
  - Convolutional Layer
  - Normalization Layer
  - ReLu Layer

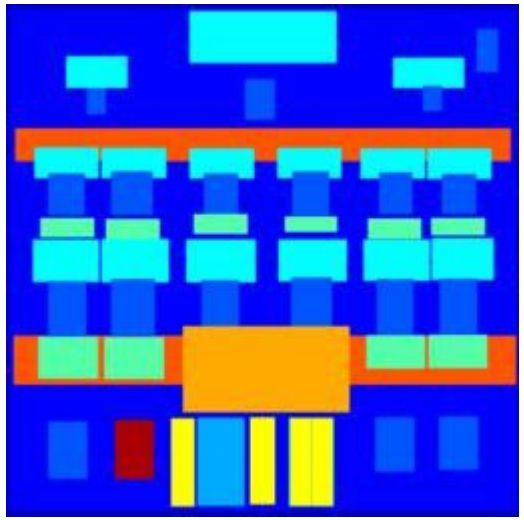
- The Discriminator is composed by 3 layers:
  - Convolutional Layer
  - Normalization Layer
  - LeakyReLu Layer

http://cs231n.github.io/neural-networks-1/

TRAINING

Source code: https://github.com/mrzhu-cool/pix2pix-pytorch.git

Source code example: labels to facades



Label

Facade

TRAINING

1. Colorization (from BW to color) train and test with facades dataset

epochs = 200, # learning rate = 0.0002 (default)

Original

Black and white

Generated

https://github.com/telecombcn-dl/2018-dlcv-team2

TRAINING

2. Colorization (from BW to color) test with cat dataset

Original

Generated

Original

Generated

https://github.com/telecombcn-dl/2018-dlcv-team2

TRAINING

3. Transfer Learning

epochs = 50, # learning rate = 0.0002 (default)

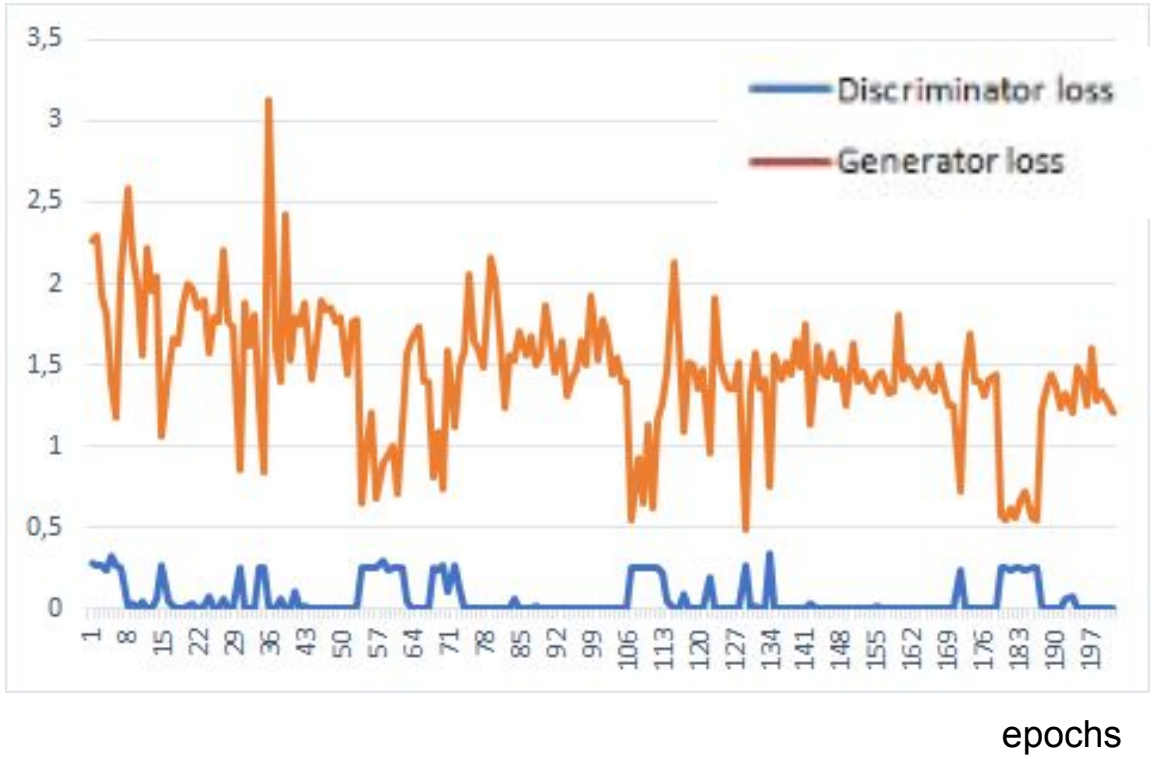
Original

Transfer

https://github.com/telecombcn-dl/2018-dlcv-team2

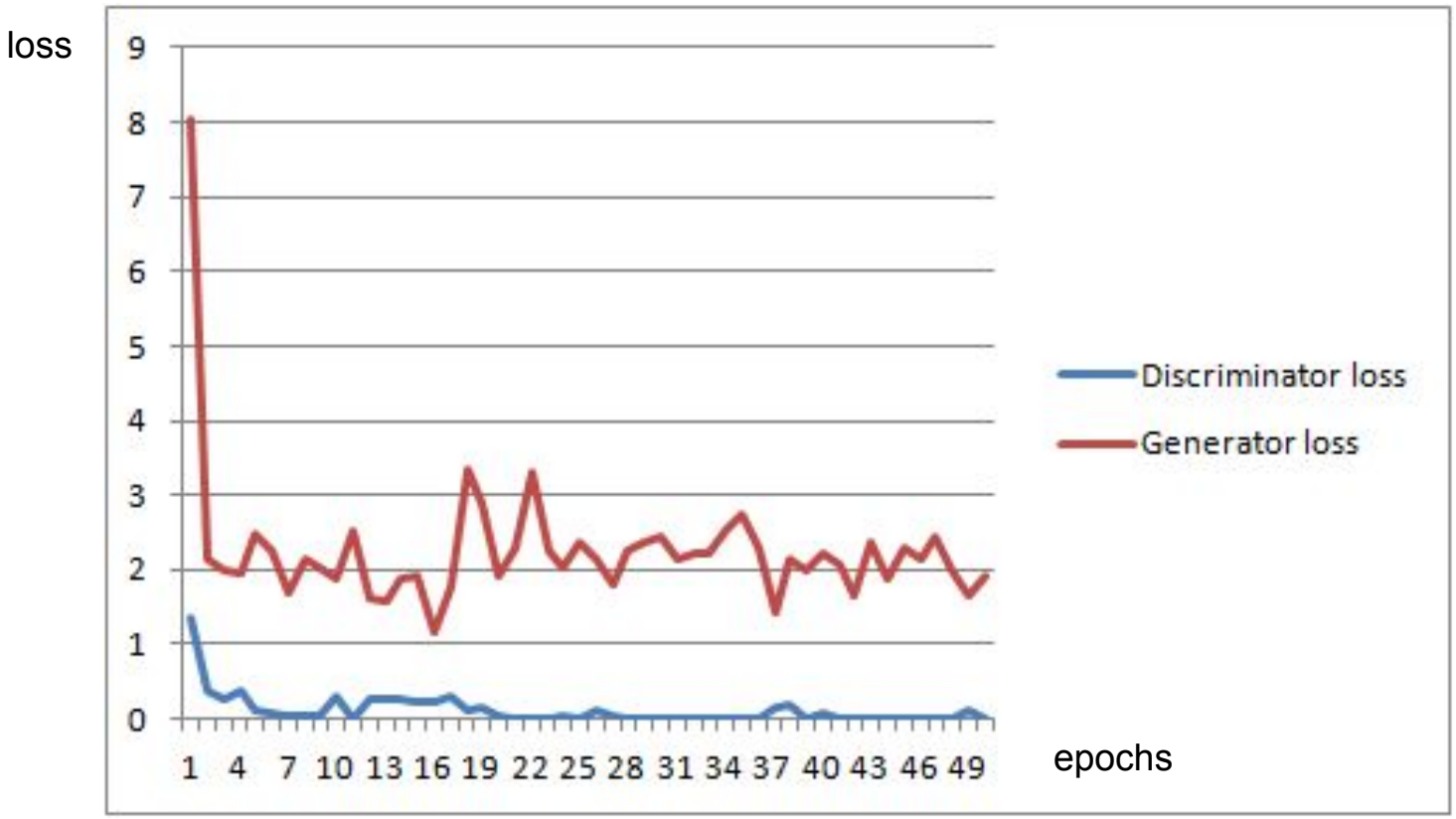
RESULTS

- Colorization trained for 200 epochs, with 200 facades training images:
  - Loss log:



RESULTS

- Transfer learning for cat dataset trained for 50 epochs, with 198 training images:
  - Loss log:



CONCLUSIONS AND LESSONS LEARNED

- Pix2pix gives overall good results in different image-to-image translation tasks
- Loss is a bit counter intuitive in GANs
- Transfer learning works better with similar datasets
- To improve the results of transfer learning → play with hyperparameters
- Imagenet, tiny imagenet: too long to train even with GPU

THANK YOU FOR YOUR ATTENTION !

Any questions ?

https://github.com/telecombcn-dl/2018-dlcv-team2/blob/master/train.py