Image Recoloring with conditional GANs

Luca Agosti   Nicola Bee   Lorenzo Saccaro

20 February 2023

Università degli Studi di Padova

Outline

Università degli Studi di Padova

1. Project overview
2. Perceptual Losses
3. Implemented architecture
4. Results

Introduction

- The goal of this work is to build a deep learning model able to **colorize** grayscale image
- To achieve this we use a Generative Adversarial Network (**GAN**)
- In particular we exploit the **pix2pix** framework, which utilizes a conditional GAN with *U-Net* generator and *PatchGAN* discriminator
- The re-colorizing task is studied using 20% of the **COCO dataset** that underwent elimination of black and white images.

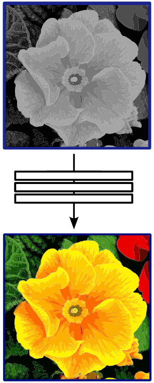

Figure: Goal: Recoloring B/W images reaching credible results

GAN, Wasserstein distance and gradient penalty

GAN:

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z)))]$$

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D)$$

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda\mathcal{L}_{L1}(G)$$

WGAN/WGAN-GP:

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma\in\Pi(\mathbb{P}_r,\mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\Big[||x,y||\Big]$$

$$\min_G \max_{D\in\mathcal{D}} \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_r}[D(\boldsymbol{x})] - \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}[D(\tilde{\boldsymbol{x}})]$$

$$\min_G \max_{D\in\mathcal{D}} \mathbb{E}_{\tilde{\boldsymbol{x}}\sim\mathbb{P}_g}[D(\tilde{\boldsymbol{x}})] - \mathbb{E}_{\boldsymbol{x}\sim\mathbb{P}_r}[D(\boldsymbol{x})] + \lambda\mathbb{E}_{\hat{\boldsymbol{x}}\sim\mathbb{P}_{\hat{\boldsymbol{x}}}}\Big[(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1)^2\Big]$$

The WGAN-GP makes the optimization of the generator **easier** and **more stable**, but it requires more epochs.

Perceptual Losses

- When comparing results obtained from different generative models we need an **objective metric** to determine the **quality** of the reproduced images.
- Perceptual losses try to solve the issues by calculating functions designed to reflect the realism of the images.
- We implemented 5 different kinds of perceptual losses: SSIM, PSNR, LPIPS, UIQI and FID.

Figure: Visual inspection of generated image quality

In the **pix2pix** framework both the discriminator and the generator see the input image (in our case the grayscale one)

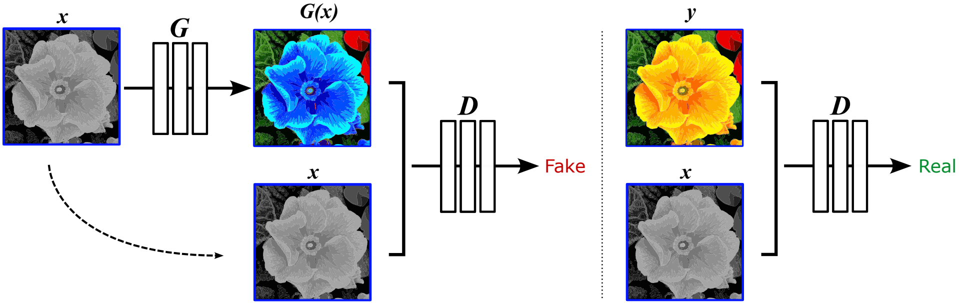

Figure: pix2pix high level schema

Generator and Discriminator

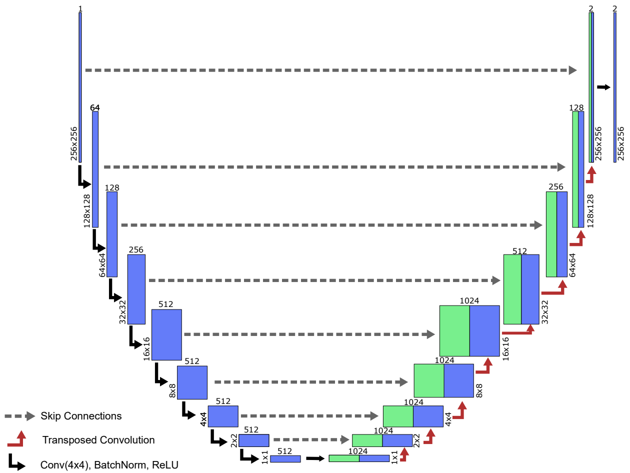

Figure: U-Net Generator

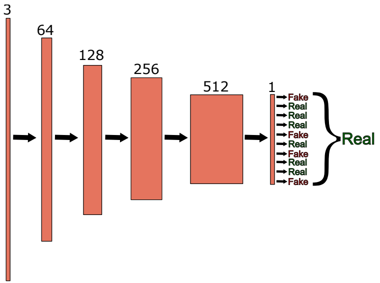

Figure: PatchGAN Discriminator

PixelTCL

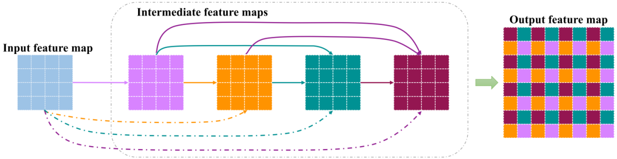

Figure: Pixel TCL main idea

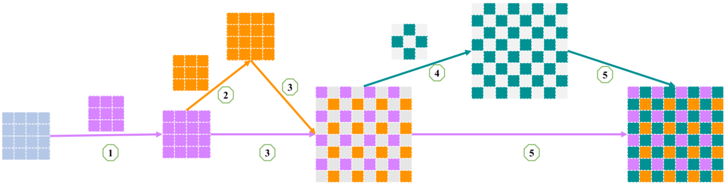

Figure: Efficient implementation

Results - generated images

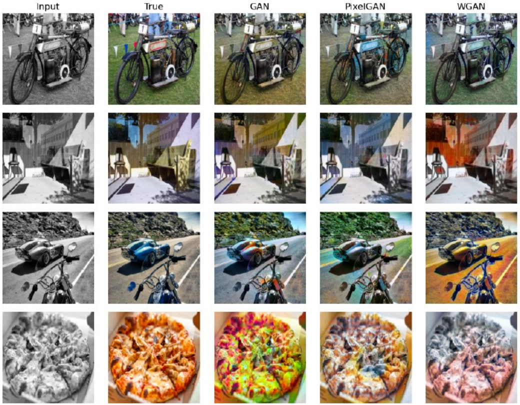

Figure: Generated images from the train set

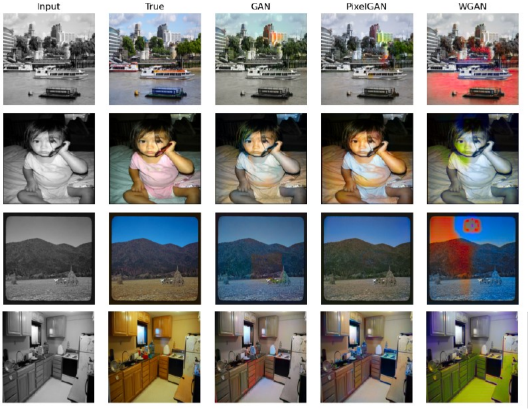

Figure: Generated images from the test set

Results - perceptual losses

Università degli Studi di Padova

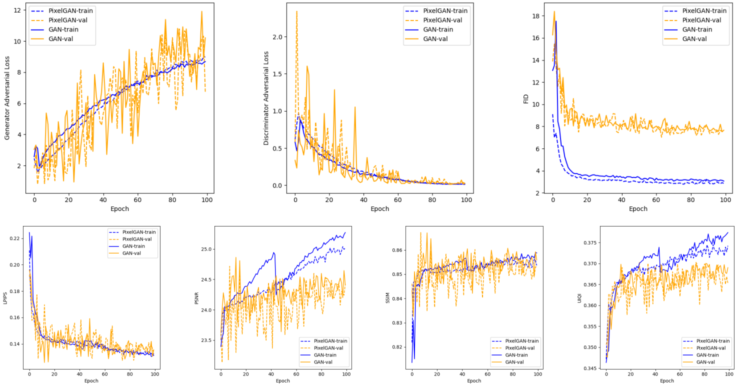

Figure: Generator and discriminator adversarial loss for GAN and PixelTCL (PixelGAN) (top left and center top). Evolution of perceptual losses during training (top right and bottom row)

	GAN	PixelGAN	WGAN
FID	2.52	2.28	3.21
SSIM	0.86	0.86	0.86
PSNR	24.44	24.44	23.83
LPIPS	0.14	0.14	0.16
UIQI	0.37	0.37	0.35

Table: Perceptual losses evaluated on the test set

Final remarks

- Training of GANs is not a simple task: finding an **equilibrium** between generator and discriminator requires lots of trials and errors
- Despite multiple experiments, tuning of optimizers, and changes to the architecture, we were unable to make the WGAN-GP implementation able to produce meaningful results
- The FID is the perceptual loss that best achieves our requests: there is a clear **correlation** between the metric values and the generated image quality
- The dataset we use is very **complex** and **diverse**: this helps in obtaining a model which should be able to perform well on a large variety of inputs but at the same time makes training more demanding

Future works

- Perform more experiments to determine the feasibility of the WGAN-GP implementation for the pix2pix framework
- Combine and further develop the strategies implemented in this work
- Incorporate the perceptual losses in the training procedure to **monitor quantitavely** the progress
- Design **additional loss terms** to generate more colorful/saturated images

References

pix2pix paper: Phillip Isola et. Al. “Image to Image Translation with Conditional Adversarial Networks”, CVPR 2017
wgan paper: Arjovsky et. Al., ”Wasserstein GAN”, 2017
wgan-gp paper: Gulrajani et. Al., ”Improved Training of Wasserstein GANs”, 2017
pixelTCL paper: Gao et. Al. ”Pixel transposed convolutional networks”, IEEE transactions on pattern analysis and machine intelligence 2013
perceptual losses paper: Alaa Abu-Srhan et. Al., ”The effect of loss function on conditional generative adversarial networks”, Journal of King Saud University - Computer and Information Sciences 2022

code available at https://github.com/lorenzo-saccaro/NNDL-recoloring-GAN