Towards Cross-modality Medical Image Segmentation with Online Mutual Knowledge Distillation

Milestone 1a Presentation

Paweł Gelar, Kacper Trębacz, Mateusz Ziemła

Warsaw 04.2022

Contents

Problem introduction

Task

Our goal in this segmentation task is annotating seven cardiac substructures:

- the left ventricle blood cavity
- the right ventricle blood cavity
- the left atrium blood cavity
- the right atrium blood cavity
- the myocardium of the left ventricle
- the ascending aeorta
- the pulmonary artery

on CT (computed tomography) scans. We also have annotated MRI images.

Data

MRI, CT scans and labels are in NIFTI (.nii) format, which can be easily converted to 3d-ndarray.

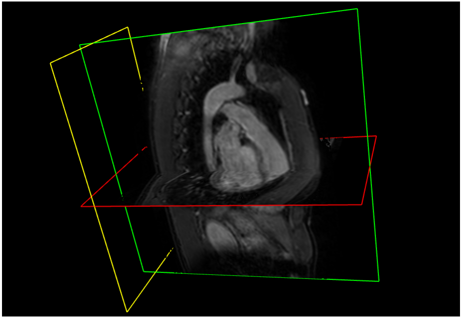

Figure: MRI scan

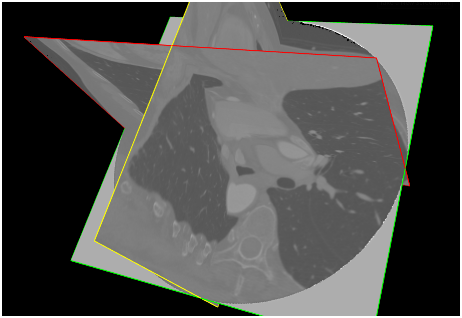

Figure: CT scan

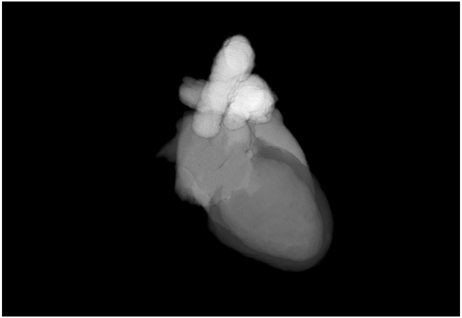

Figure: Label

Model architecture

The composed model

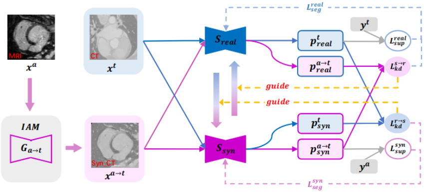

Figure: The architecture

The segmentators

The models consists of two segmentators, which train on both real and GAN-synthetized data. After feeding the data through both models, we get the predictions, denoted as p_{real} and p_{syn} respectively.

The composed model

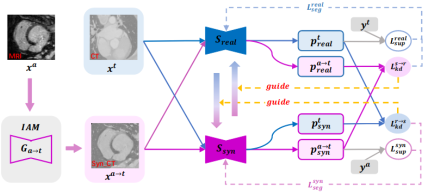

Figure: The architecture

The shared knowledge

Both models are scored on both the differences between the predicted segmentations and the actual ones, but they are also scored on the their similarity - the model focusing on the real data gets a loss term calculated from the dissimilarity of its predictions of synthetic data and the synthetic-focused model's predictions of the synthetic data.

The composed model

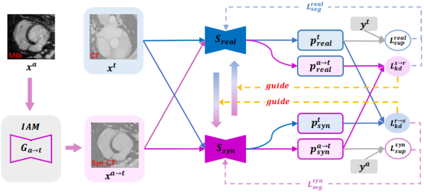

Figure: The architecture

Mutual Knowledge Distillation

This enables the models to learn from each other, and guide each other decisions, in a novel process the authors called Mutual Knowledge Distillation.

GANs

Generative Adversarial Network

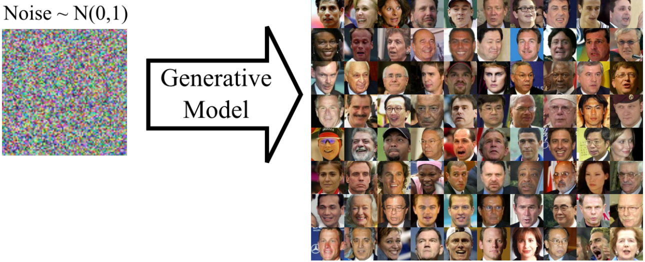

Figure: Face Generation

Generative Adversarial Network

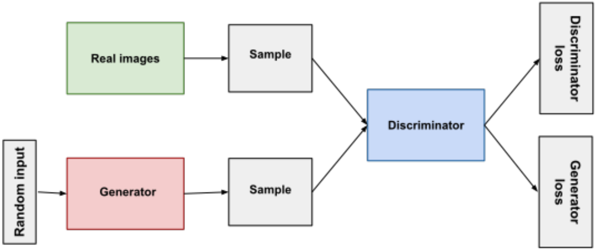

Figure: The architecture

Generative Adversarial Network

$$\min_{G} \max_{D} \mathbb{E}_{x \sim q_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z} \sim p(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))],$$

Figure: Loss

Conditional GAN

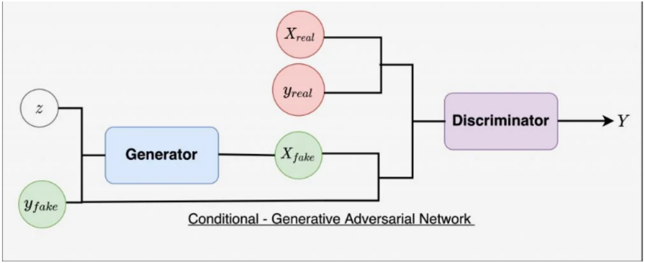

Figure: Conditional GAN architecture

Conditional GAN

Figure: Image from Edges

Cycle GAN

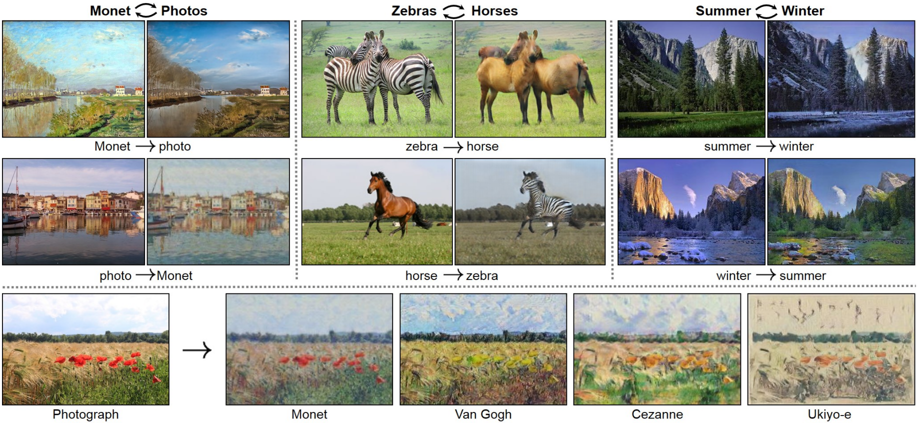

Figure: Cycle GAN Examples

Cycle GAN

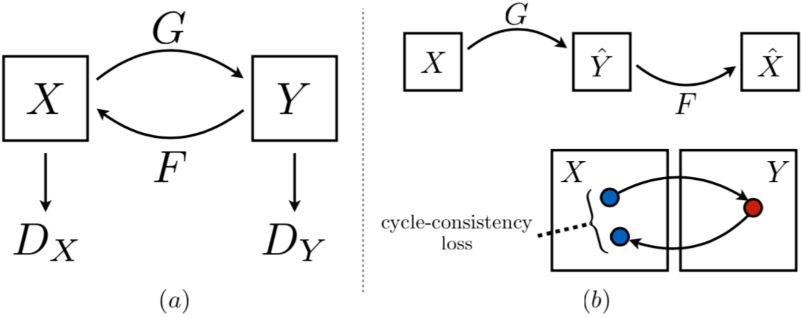

Figure: Cycle GAN Architecture

Cycle GAN

$$\begin{aligned}\mathcal{L}_{\text{cyc}}(G, F) =& \mathbb{E}_{x\sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] \\ &+ \mathbb{E}_{y\sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1].\end{aligned} \tag{2}$$

Figure: The Loss

Execution Plan

Optimistic Schedule :)

- 14.04 - Overview of the Literature
- 21.04 - Paweł - Dataset, Kacper - GAN, Mateusz - Unet
- 28.04 model definition
- 05.04 training on dataset 1
- 12.04 training on dataset 2
- 19.04 creating overleaf file for paper
- 05.05 delivering final paper

Pesymistic Schedule :)

- 14.04 - Overview of the Literature
- 21.04 - realising that it is too difficult and switching to testing k-plet loss (:
- Later - who knows

Other Paulina's Questions :)

- Which tools and technologies will be used? - PyTorch, dvc, Weights & Biases
- Other dataset - GTA and real life street photos segmentation
- Other experiments - ablation study (?)

Images sources

- https://developers.google.com/machine-learning/gan/gan_structure
- https://miro.medium.com/max/1400/0*0sy8_NFeJXzqgqFs.png
- https://learnopencv.com/conditional-gan-cgan-in-pytorch-and-tensorflow/
- https://arxiv.org/pdf/1611.07004.pdf
- https://media.arxiv-vanity.com/render-output/5766162/image/cyclegan.png
- https://junyanz.github.io/CycleGAN/images/teaser.jpg
- https://i.imgur.com/OurehZ5.png

Thank you!