# Submission

“Please handing presentation, code, images and copies of papers read. You should do this via the University Dropoff System and e-mail links to John and Mark”

WE DON’T HAND IN A MODEL -> We trained it at home and kept the workspace until the presentation.

# Introduction

Project aim: Photorealistic background generation

The process involved is called image-to-image translation that translates one image into another. HOWEVER, when it comes to changing the background from, for example, a day to a night scene, EXAMPLE HERE there are many possible solutions for that task, since “night” is quite unspecific for any model.  
Therefore, we had a look into an approach providing a distribution of image solutions. It uses a Generative adversarial network (GAN) where two neural network compete with each other into a zero-sum game to create a result for which the model cannot tell if it was created artificially and or is a real image. This was proposed in a paper from these 4 chaps (Phillip Isola Jun-Yan Zhu Tinghui Zhou Alexei A. Efros) from the University of California, Berkeley.

And, here are our results …

# Demo

1. Day to Night **images X**
2. Night to Day **images X**
3. Day to night **video X**
4. Blueprint to map **Video X**
5. Shoes edges2shoes -> SPECIAL: draw the shoe in class! As **VIDEO**

Encoded sample = pix2pix with noise

# Limitations:

1. Very dependent on training data -> e.g. tries to fit the image to the more landscapes/ city images from the training data
2. And, with the input providing limited features, especially in night scenes, it’s hard to estimate the ground truth

# Related work: pix2pix (one image to one output image)

… how that works broadly

Using the architecture from their paper

# Implementation

Steps for our project:

1. Research: Reading the paper and other versions of it and understand their architecture
2. For those publications were solutions were available: Understand the solution OR pick one with libraries that we can use and understand, here: pytorch and tensorflow
3. Bringing pre-trained models to work and use them on their and our input data
4. Training our own model from their architecture implementation with their data (and own labelled data, if available)
5. Rebuilding the GAN with our tweaks and changes and train it

Difficulties:

* Not really compatible for windows, so first hurdle to make it work for windows
  + Used older version of Pytorch not available for windows …
* Using over 14 GB of RAM to create training and test files out of input in tensorflow

First Results (their architecture, trained on their data, generator mapped into wrong domain):

1. Facades bad output images
2. Facades good output

# Examples from other publications

1. Nvidea Faces
2. Edges2Cats Online tool
3. Cycle GANs

# Limit of the Code slide

“We have noticed a tendency to depend on code preexisting in matlab or downloaded from the web. This is not a good idea as you have no idea what the limit of the code is. For this project anything you don't write must be tested and you must include details of the testing at the end of the final presentation (extra slides after you have finished the talk)”

# Sources

**GANs**:

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, “Generative Adversarial Networks”, eprint, 2014, [arXiv: 1406.2661](https://arxiv.org/abs/1406.2661)

**Pix2pix using CAN**(1 image to 1 image)**:**   
P. Isola, J. Y. Zhu, T. Zhou and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, [arXiv: 1611.07004](https://arxiv.org/abs/1611.07004), GitHub: <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>

**BicycleGAN using GANs** (1 image to distribution):

J.Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, E. Shechtman, “Toward Multimodal Image-to-Image Translation”, 2017 in NIPS, 2017, [arXiv: 1711.11586](https://arxiv.org/abs/1711.11586),  
GitHub: <https://github.com/junyanz/BicycleGAN>

**CycleGAN using GANs** (1 image to 1 image with no labelled inputs, only domain specific inputs):

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", in IEEE International Conference on Computer Vision (ICCV), 2017, [arXiv:1703.10593](https://arxiv.org/abs/1703.10593), GitHub: <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>

**Nvidea:**

T. Karras, T. Aila, S. Laine, J. Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, 2018 ICLR Conference on Learning Representations, Vancouver, 2017, [arXiv: 1710.10196](https://arxiv.org/abs/1710.10196), url: <https://arxiv.org/abs/1710.10196>,   
GitHub: <https://github.com/tkarras/progressive_growing_of_gans>