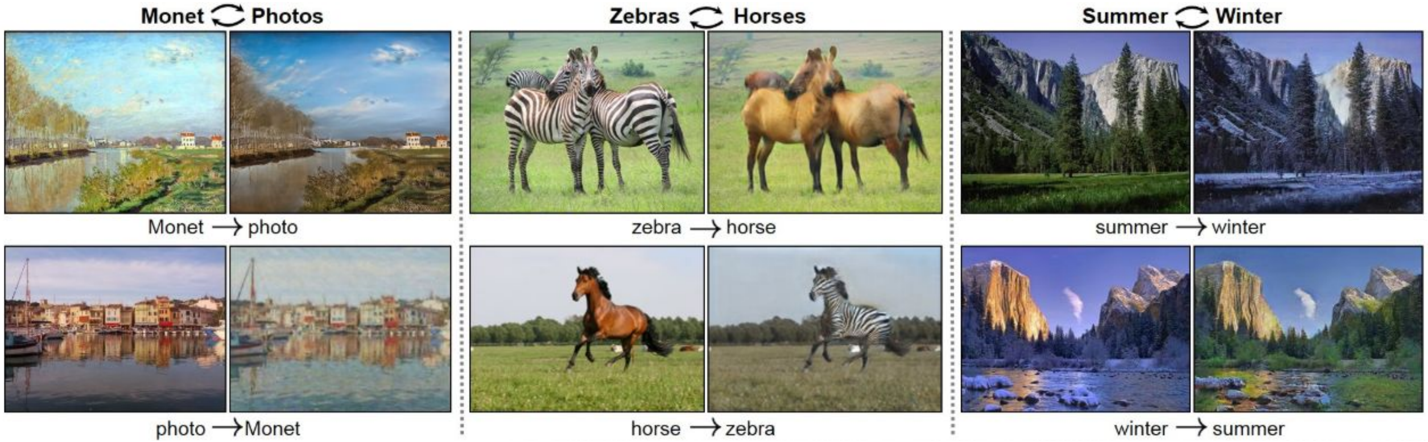
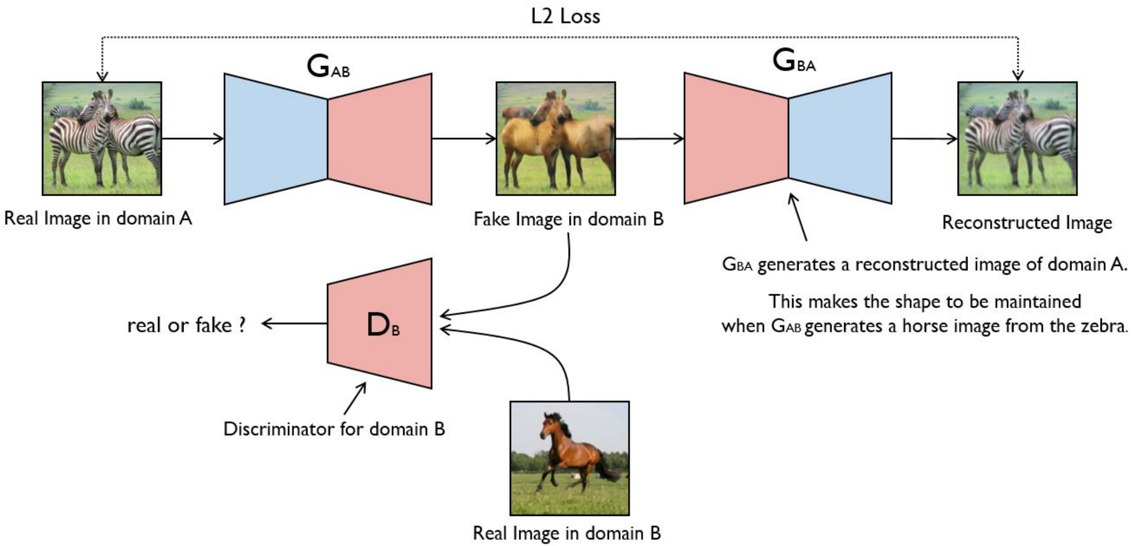
CycleGAN



- ICCV 2017 groundbreaking paper
- First work, together with DiscoGAN, that exploits 2 GAN models for source → target and target → source mapping at the same time!
- Used as baseline method for several other models

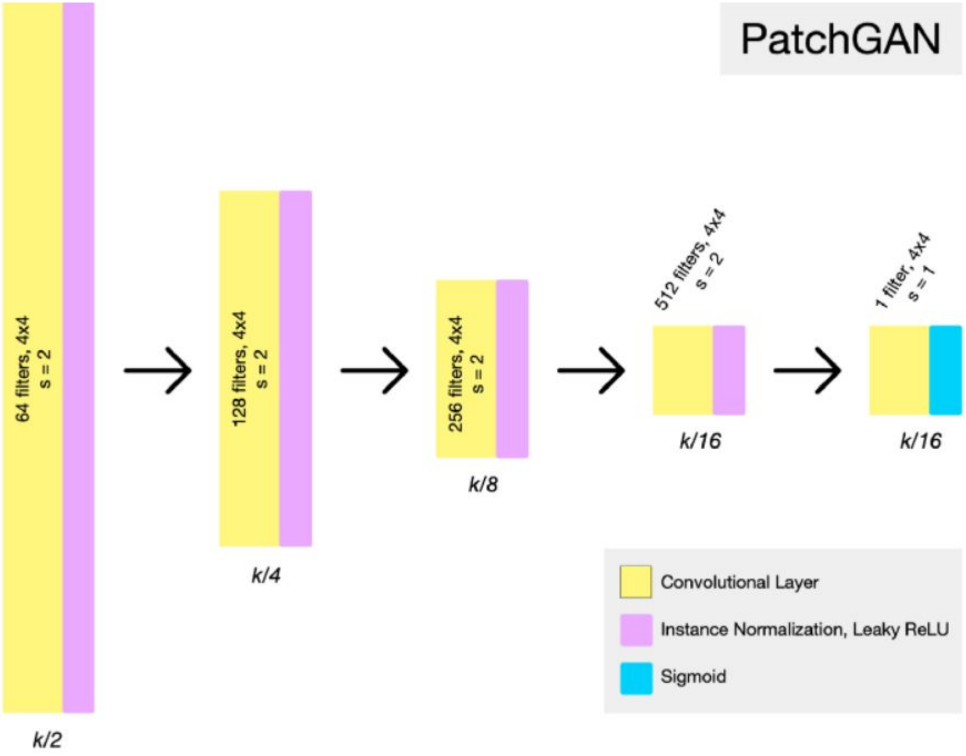
CycleGAN



- L2 reconstruction loss between input and Gab-Gba concatenation!
- Take home message: impose additional losses/constraints when you can

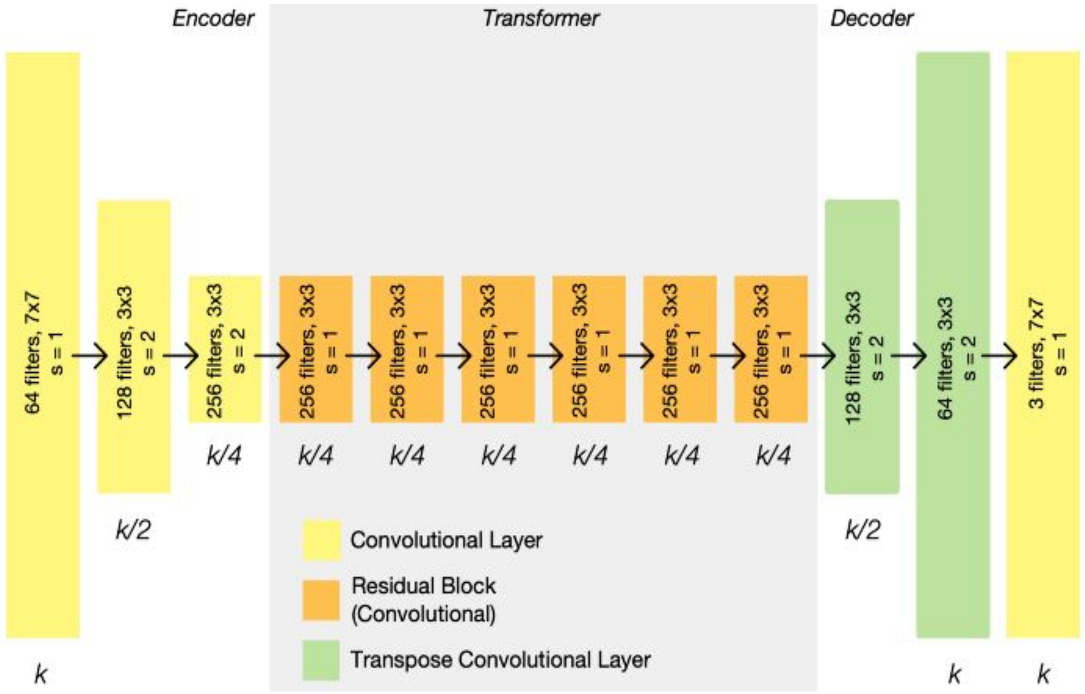
CycleGAN - discriminator

- No BN and ReLU - use Instance Norm and LeakyReLU instead!
- The binary classification is performed on patches with a simple trick

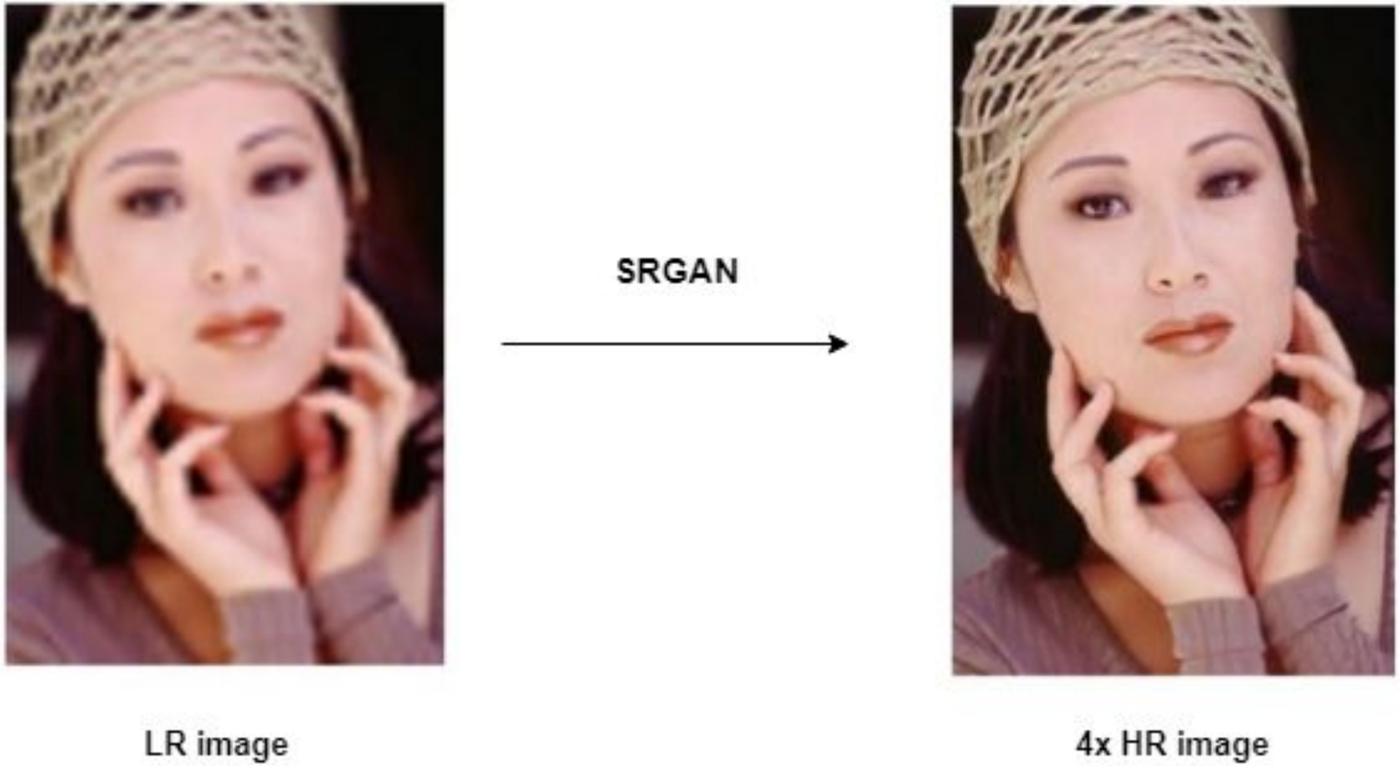


CycleGAN - generator

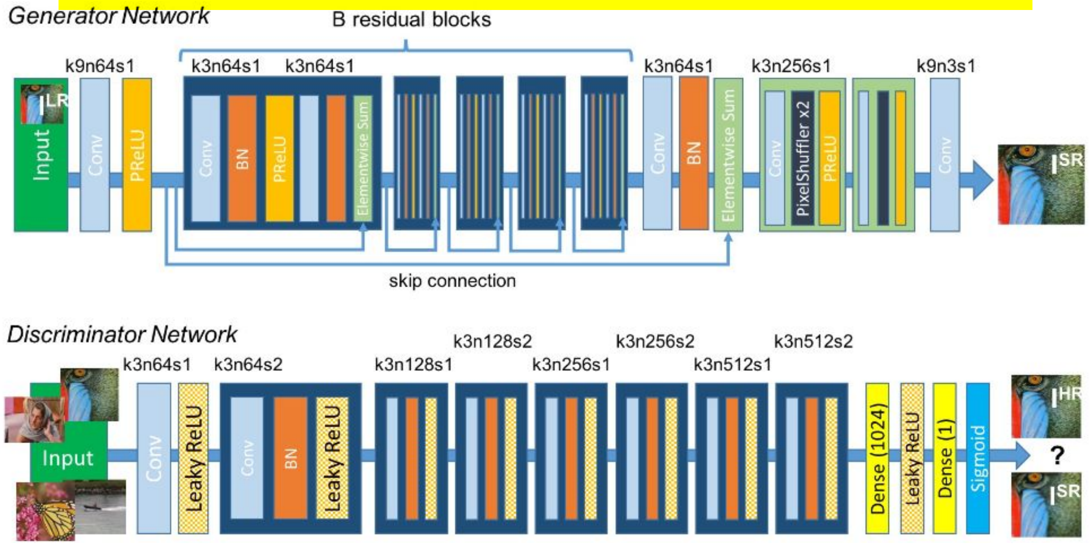
- Encoder-Transformer-Decoder approach to gently map image-to-image
- The residual approach helps a lot



Super Resolution GAN



Super Resolution GAN



- Usual residual technology in generator, but bigger discriminator
- Pretrained generator!
- It exploits an L2 reconstruction loss on generated images not pixel based, but Vgg16-features based!

Progressive growing GAN

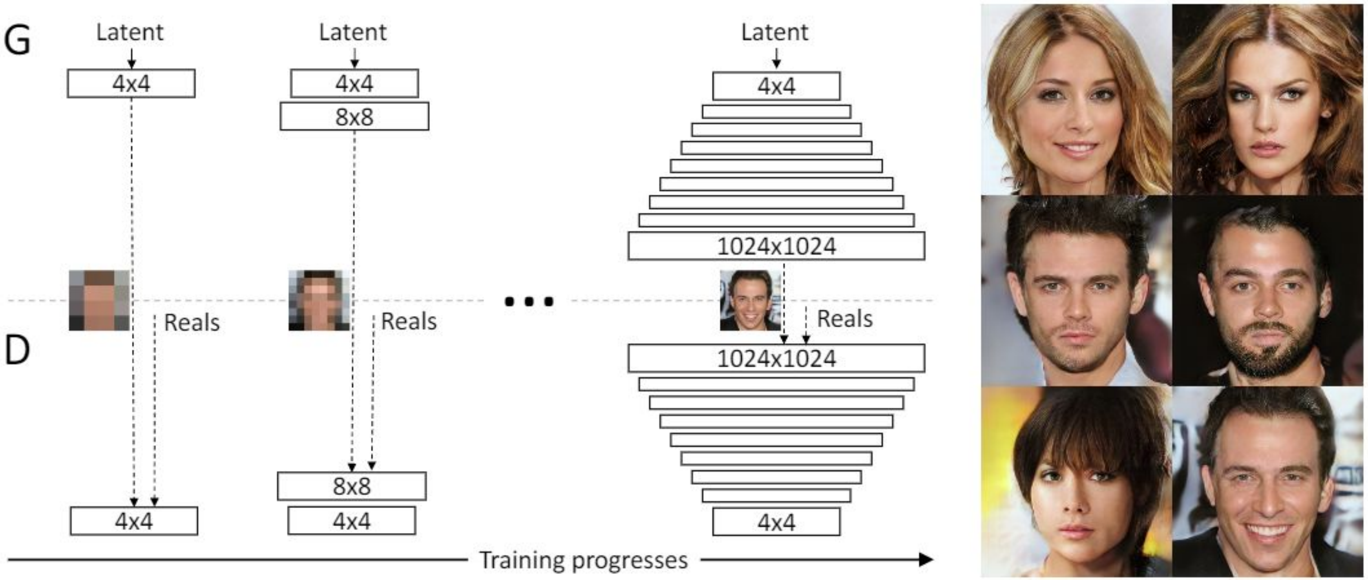
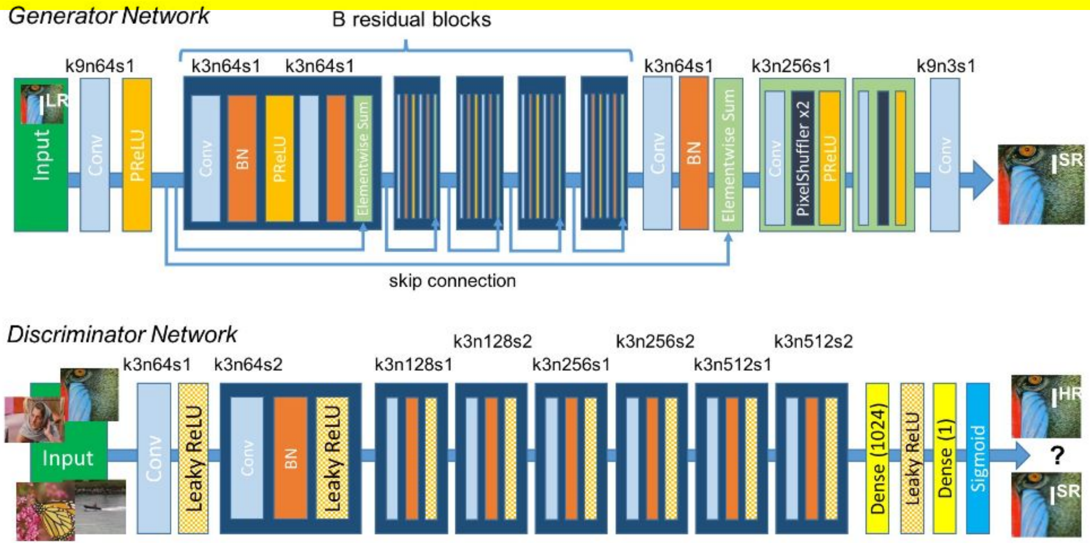


Figure 1: Our training starts with both the generator (G) and discriminator (D) having a low spatial resolution of 4×4 pixels. As the training advances, we incrementally add layers to G and D, thus increasing the spatial resolution of the generated images. All existing layers remain trainable throughout the process. Here $\boxed{N \times N}$ refers to convolutional layers operating on $N \times N$ spatial resolution. This allows stable synthesis in high resolutions and also speeds up training considerably. One the right we show six example images generated using progressive growing at 1024×1024.

Progressive growing GAN



- The devil is in the details:
  - Minibatch discrimination for added variety
  - Smooth linking process of new layers
  - Pixelwise feature vector normalization in generator

Progressive growing GAN