# Pix2Pix
## with cGAN's

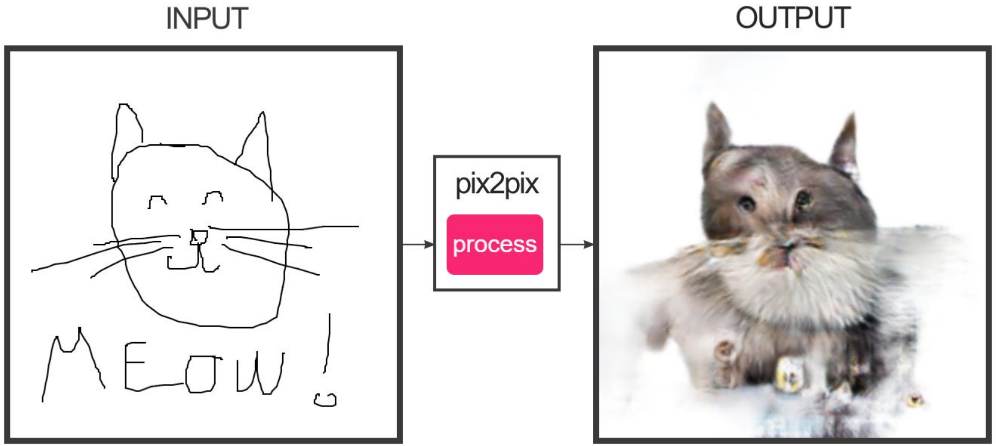


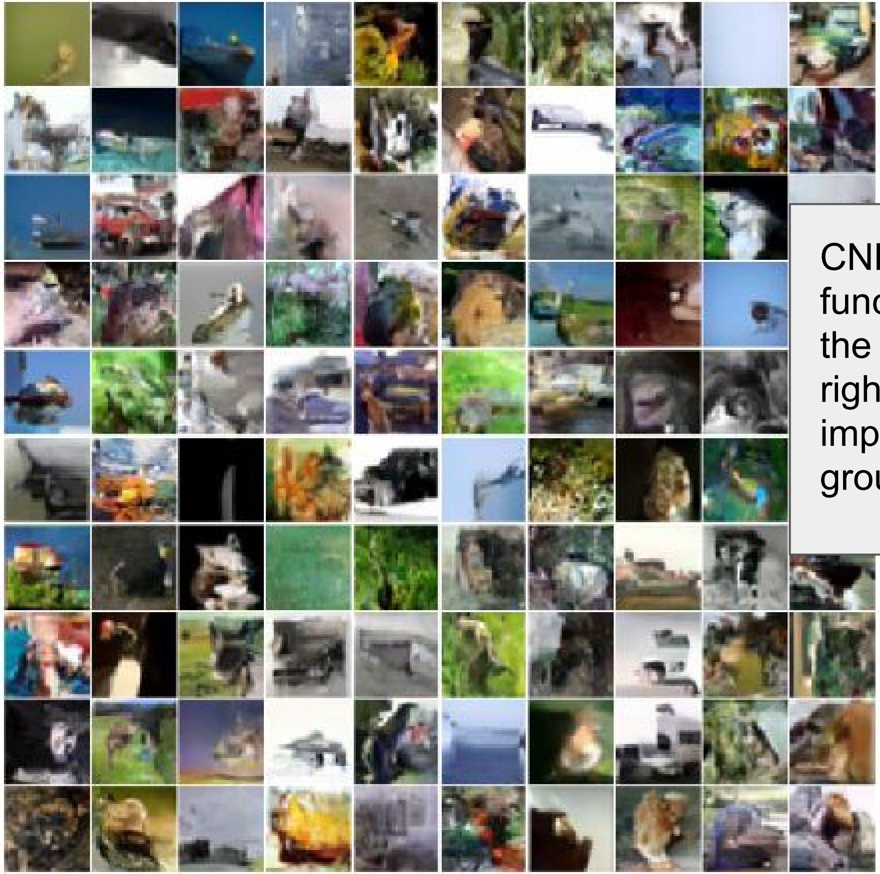
Image-to-Image Translation with Conditional Adversarial Networks

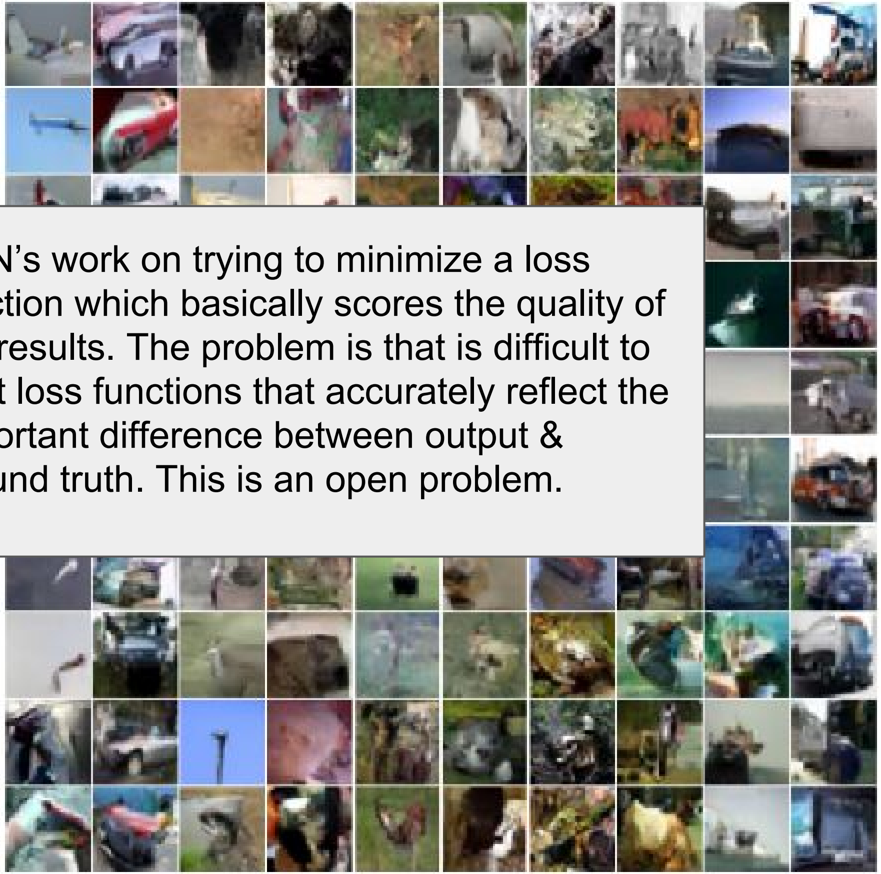
## The proposal:

image translation is a super common problem in computer vision with specialized solutions often being developed when in reality it is a general problem that can be solved by a general solution.

one set of pixels must become another set of pixels.

# CNN

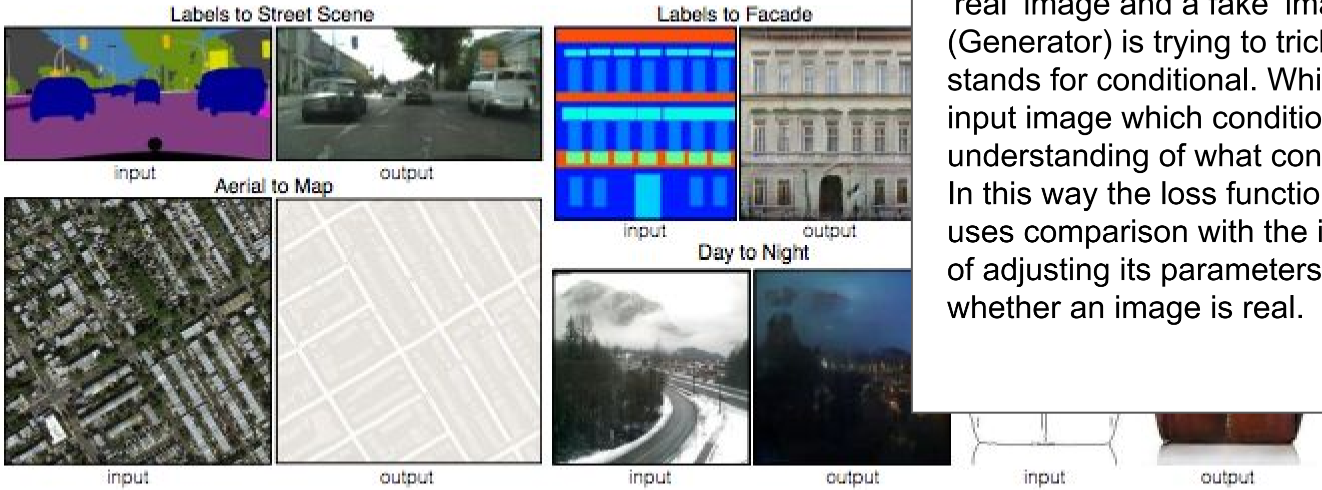




CNN’s work on trying to minimize a loss function which basically scores the quality of the results. The problem is that is difficult to right loss functions that accurately reflect the important difference between output & ground truth. This is an open problem.

# cGAN’s

cGAN’s learn a loss function that adapts to the data as it goes. Noise is generated then filtered (discriminated) based on the relationship to the ground truth. As in a traiditional GAN the D (discriminator) is trying to distinguish between a ‘real’ image and a fake ‘image’ and the G (Generator) is trying to trick it. The c in cGAN stands for conditional. Which accounts for the input image which conditions the discriminator’s understanding of what constitutes a ‘real’ image. In this way the loss function is ‘learned’ as it uses comparison with the input image as a way of adjusting its parameters for determining whether an image is real.



PIX2PIX approach to cGAN's

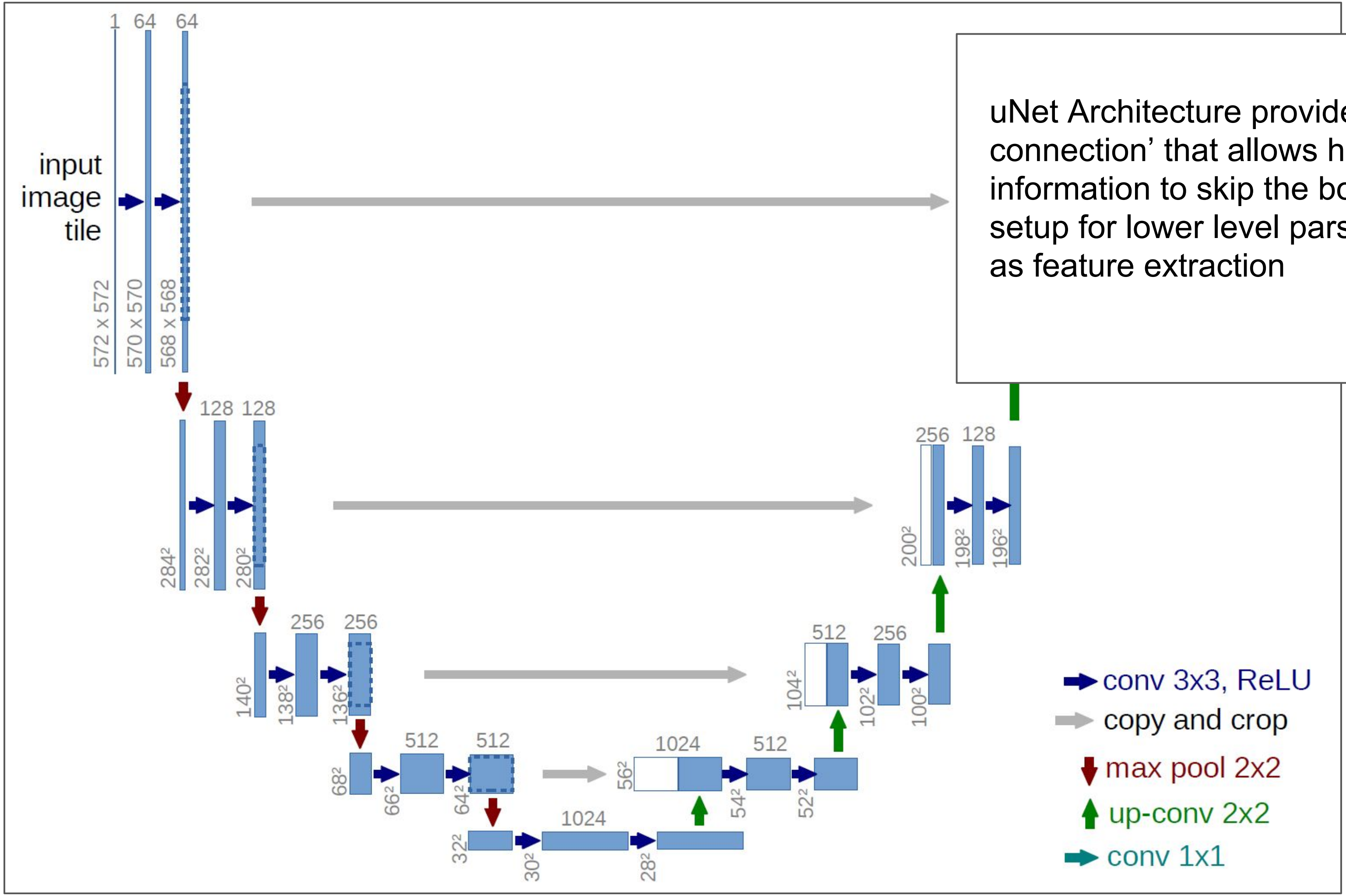
uNET architecture

train on fewer images for a faster input/output relationship - this is achieved through an architecture that has simultaneous training paths, a narrow and a wide

patchGAN


Figure 6: Patch size variations. Uncertainty in the output manifests itself differently for different loss functions. Uncertain regions become blurry and desaturated under L1. The 1x1 PixelGAN encourages greater color diversity but has no effect on spatial statistics. The 16x16 PatchGAN creates locally sharp results, but also leads to tiling artifacts beyond the scale it can observe. The 70x70 PatchGAN forces outputs that are sharp, even if incorrect, in both the spatial and spectral (coforfulness) dimensions. The full 256x256 ImageGAN produces results that are visually similar to the 70x70 PatchGAN, but somewhat lower quality according to our FCN-score metric (Table 2). Please see https://phillipi.github.io/pix2pix/ for additional examples.

only recognizes issues with large patches of image (70x70px) this provides speed and less noise/blur in the final image, smaller sections can be used but that can result in a tiling effect




uNet Architecture provides a ‘skip connection’ that allows higher level information to skip the bottlenecks setup for lower level parsing such as feature extraction


Figure 5: Adding skip connections to an encoder-decoder to create a "U-Net" results in much higher quality results.

In other image to image translators they achieve a high resolution input to high resolution output by creating a successive series of layers that downsample features until the middle where there is a bottleneck and then upsampling happens. U-Net avoids doing this by with the skip function. Higher level features that are relatively continuous between input and output bypass layers of downsampling. Thus reducing load in the bottleneck. U-Net architecture greatly increases 'realism' of output with comparably fewer images needed for training.

For testing of the efficacy of their system they used Amazon Mechanical Turk with the following outcome:

| Loss | Photo → Map<br>% Turkers labeled *real* | Map → Photo<br>% Turkers labeled *real* |
|---|---|---|
| **L1** | $2.8\% \pm 1.0\%$ | $0.8\% \pm 0.3\%$ |
| **L1+cGAN** | $6.1\% \pm 1.3\%$ | $\mathbf{18.9\% \pm 2.5\%}$ |

Table 3: AMT "real vs fake" test on maps↔aerial photos.

| Method | % Turkers labeled *real* |
|---|---|
| **L2 regression from [46]** | $16.3\% \pm 2.4\%$ |
| **Zhang et al. 2016 [46]** | $\mathbf{27.8\% \pm 2.7\%}$ |
| **Ours** | $22.5\% \pm 1.6\%$ |

Table 4: AMT "real vs fake" test on colorization.


Figure 8: Example results on Google Maps at 512x512 resolution (model was trained on images at 256x256 resolution, and run convolutionally on the larger images at test time). Contrast adjusted for clarity.

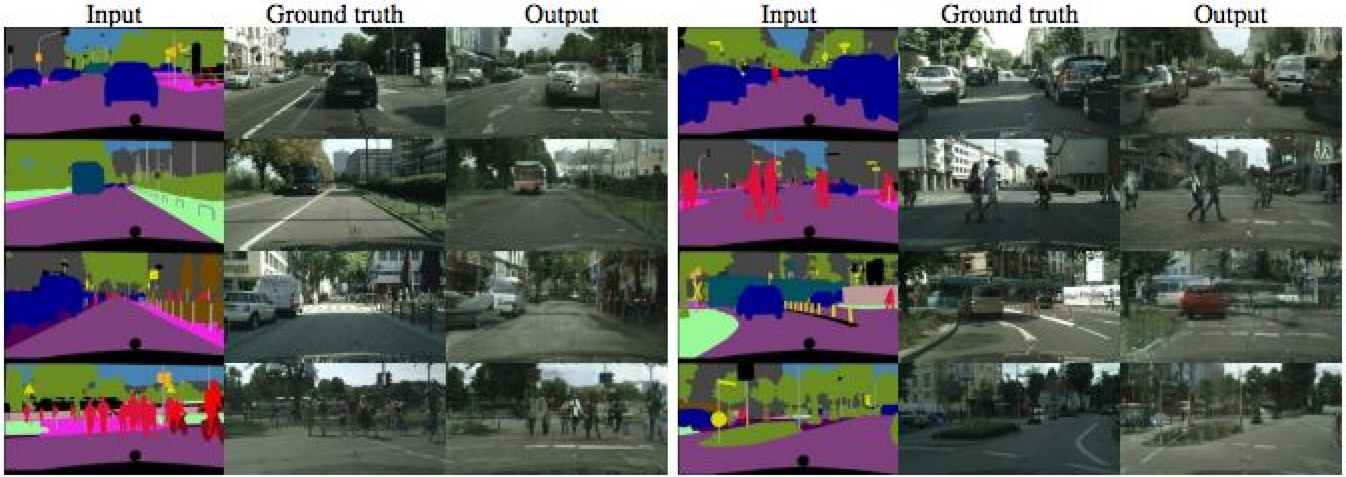



Figure 11: Example results of our method on Cityscapes labels→photo, compared to ground truth.


Figure 15: Example results of our method on automatically detected edges→shoes, compared to ground truth.