Aerial Drone Urban Scene Image-to-Image Translation

By: TeVaughn Shaw

Goal

To generate synthetic urban scene images and make them as realistic to their target images with the use of other GAN-based models for evaluation.

Background

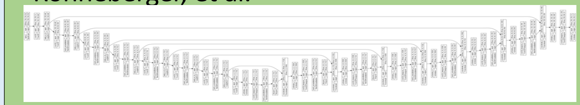
- Drones have been an inexpensive relief for businesses in recent years. Mainly in the sense of surveying and monitoring huge areas over time in job fields such as construction, real estate, and agriculture.
- Not only are drones cost efficient, but less threat to public safety with recent technology improvements over the years that produces high quality imaging and video.
- This project used Pix2Pix GAN training to produce synthetic images using the semantic drone dataset by the Institute of Computer Graphics and Vision.

Conditional-GAN Model Overview

- A conditional-GAN requires an additional input compared to a regular GAN architecture.
- RGB segmented imaging was used as input to the generator submodel while the discriminator is trained separately on domain of real images to determine whether the samples provided are real or fake.

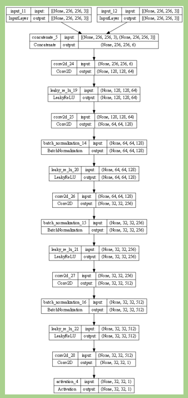
Model Architecture

Basic U-Net Generator Architecture inspired by Ronneberger, et al.



Consists of 8 encoding and decoding layers with skip connections to concatenate information.

PatchGAN Discriminator Architecture inspired by Isola et al., 2018.



Unlike CNN models that provide probabilities of the input of whole image, PatchGAN will classify whether input is real, or fake based on a small dimension or "patch" of the image. 70x70 patch is recommended to be the best for forcing sharp output even for spatial and spectral dimensions in images.(Isola et al., 2018)

Model Results

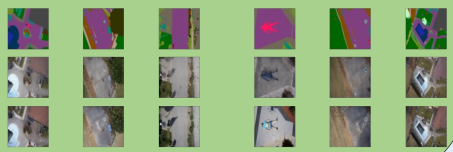
- 400 training images.
- 100 epochs initial run, total 40,000 training steps.
- Batch size = 1, means 400 training steps.
- Generator model saved every 10 epochs or 4,000 training steps.

After 40,000 iterations | After 32,000 iterations

Input

Generated

Target



Project Summary

- Due to time constraints, the project wasn't able to get to the enhancement stages on the generated images.
- There was an issue with the loss functions fluctuating through the iterations and that the generator did not show any gradual learning.
- Although the generated images do resemble their original images, there is noise present and information missing such as shadows, linings, boarders, etc.

Future Steps

- Explore project with smaller epochs.
- Change architecture of Generator.
- Fix PatchGAN on the discriminator so output will be 30x30.
- Develop visual metrics once accurate loss functions have been made.
- Build metric on noise and color to evaluate.
- Find a way to save generated images (NumPy array data types).
- Pass generated images through enhancement GAN to compare results of noise, color, details, etc.

Special Thanks

- Institute of Computer Graphics and Vision for use of dataset: http://dronedataset.icg.tugraz.at
- Jason Brownlee for adapting code for project use: https://machinelearningmastery.com/how-to-develop-a-pix2pix-gan-for-image-to-image-translation/