Sim - 2 - Real GAN

Synthetic data augmentation with Adversarial Nets

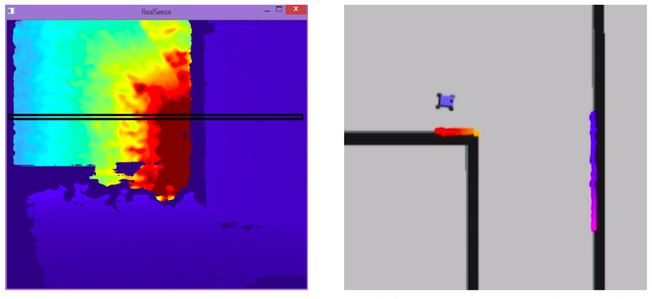
CSCI 5922 - Final Project Poster – Mike Flanigan

Introduction

This poster presents the findings, techniques tried, coolness, and occasional failure of a project aimed at achieving better deep learning robot localization through the leverage of adversarial data augmentation. The data, goal, and motivation are explained. The process used is detailed. And the results which were successful in localization, but a general failure in GAN training are discussed.

Data Gap

Simulators have become a prevalent and powerful tool in the field of robotics. Also Neural Networks, Deep Neural Networks, and Generative Adversarial Networks have become prevalent in computer science in recent years due to large performance improvements. Since various robotics problems are difficult to solve analytically but can be solved by Machine Learning algorithms with large amounts of data the pairing of simulators and machine learning technique is an active field of research.
A problem that has become a known show-stopper is the poor transfer-ability of simulation trained models to real world settings. The current logical explanation for the lack of performance transfer is the slight differences in data that are hard to model analytically in simulators. Figure 1 below shows the difference in what a simulator presents and what a robot sees in the real world at a corner of the test track.

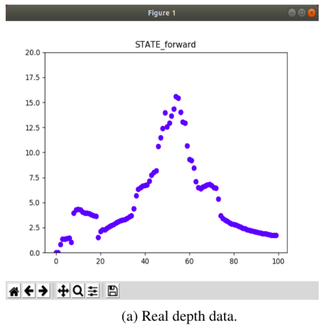


(a) Real depth data, black row indicates used data.

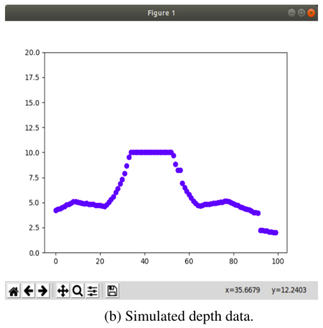
(b) Simulator environment and depth data shown.

Fig. 1: Depth data coming from a real robot and the corresponding simulator.

While the above figures give an overview of the difference between the simulator and real scenarios, it's certainly useful to visualize the actual data the networks are handling. Figure 2 below shows the single line-scan array that the localization network consumes. Immediately a human can note that there are probabilistic and structure differences between the simulated and real scan data.



(a) Real depth data.



(b) Simulated depth data.

Fig. 2: Real and simulated line scan depth data.

Technique

The goal and theory behind the use of adversarial training is to close the data gap between the simulated data and real data by learning the differences necessary and augmenting the simulated data with those differences. To further detail the learning goal; a Generator network (shown in Figure 3 below) takes in a line-scan, augments it by passing it through a convolutional neural network, and is penalized if the Discriminator network is able to identify it as from a simulator. The Discriminator network accepts both augmented simulation data and real unlabeled data and learns to distinguish the two with a binary cross entropy loss. A benefit of this process is that real data can be used to improve a simulator without having to have a human annotate labels.
Figure 3 below shows the training loop in the upper section and mostly in red, the lower section and flow with green arrows shows the pipeline for taking simulation data and passing it through the generator and then to a MLP localization network once the generator has been trained.

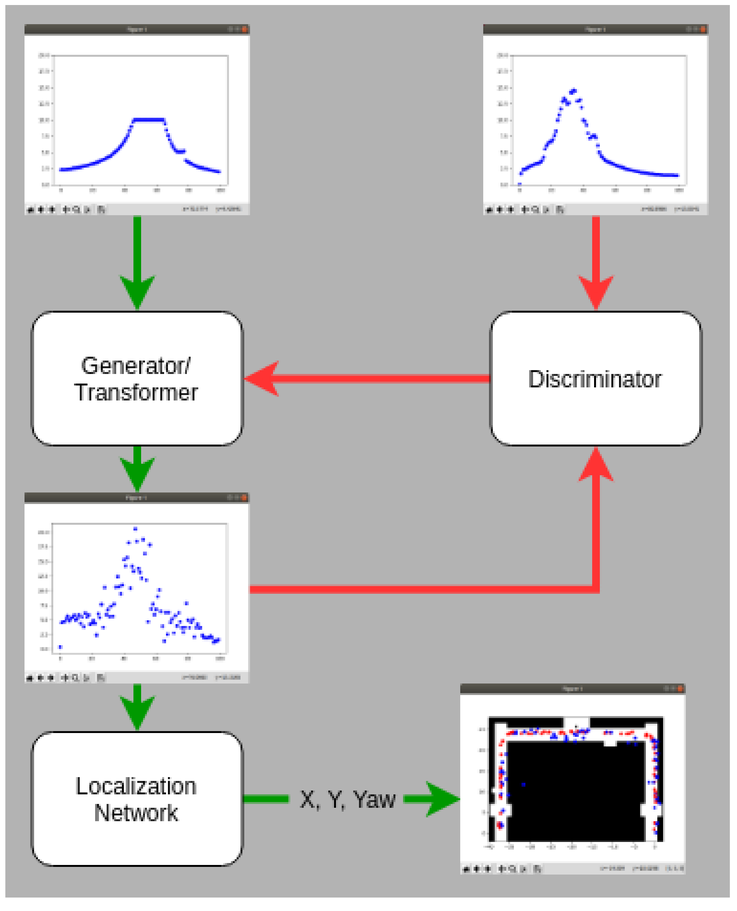


Fig. 3: Flow diagram showing adversarial training and generator use.

An important note on the above image is that the process of training the discriminator and generator networks above typically happens in batches and independently. The reason for this is that keeping the generator and discriminator at similar levels of success, with respect to tricking or catching each other, is critical for ensuring successful training.

Challenges

Training GANs is known to be a difficult process since it's easy for the networks to get out of balance. During the testing of this network if the discriminator was trained first it would quickly become so good at distinguishing fake data that the generator wouldn't receive enough positive examples to ensure good learning. One solution to this is to add noise to the output of the generator as well as to the real data. The technique that appeared to work reasonably well was to simply iterate the training starting with the generator and ensuring that the generator always stayed "a step ahead" of the discriminator during training. Figure 4 below shows the generator loss from a batch when it is trained first.

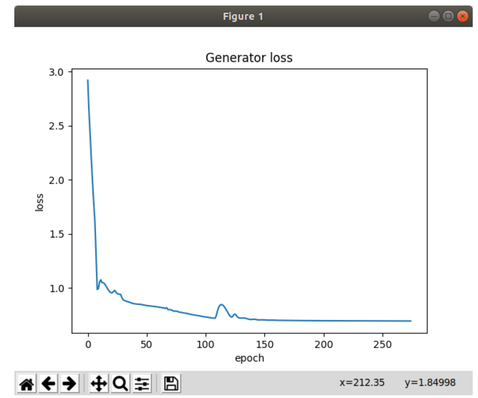


Fig. 4: Training loss during generator training.

Conclusion

While consistent and impressive GAN training turned out to be a tricky problem the approach is certainly viable but needs plenty of time to allow for tuning and ML tricks to be applied. An upside was that the pipeline was implemented and localization results from the multi-layer fully connected network were surprisingly good. Figure 5 below shows the real data labels in red and the trained network predictions in blue.

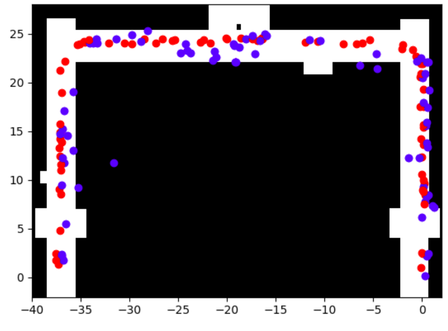


Fig. 5: Output of real and predicted robot location by network fed refined data.

All in all this was semi-successful and has been a great motivational project to drive future research on GAN data augmentation. The world of robotics and machine learning is an exciting place!