Effective Training of Artificial Neural Networks for Autonomous Navigation*

Outlook

- Introduction
- Backpropagation Algorithm
- Realtime Model Training
- Discussion
Introduction

- Neural Network typically requires too a large training dataset and prohibitively long training time to make it widely practical for realtime perception applications.
- Autonomous Land Vehicle In a Neural Network (ALVINN) system remedies both restraints by using backpropagation network trained with 200 low pixel images in 5 minutes before driving autonomously up to 20 mph on a variety of circumstances.
Navlab Testbed

This testbed is remodeled Chevy van. Equipped with hydraulic driving system, velocity is maintained under 20 mph for this project.
Backpropagation

- Input = 30 x 32 receiving “retina” units from video camera
- One hidden layer of 5 units
- 30-unit output layer to represent the direction Navlab should travel
- Supervised model and model output will be compared to human steering to update weights
Why 30 output units instead of conventionally 1 output?

ALVINN system is trained to generate a Gaussian distribution of activation centered around the steering direction to keep Navlab on the road. And this distribution can be deemed as probability density function for each unit to be the correct direction. For i-th unit, desired activation $x_i$ can be approximated with

$$x_i = e^{(-d_i^2/10)}$$

Where $d_i$ is distance from correct direction, and 10 expirial factor.
Traditional Approach

- 1200 synthetic road images with different weather and lighting conditions
- Takes 30 - 40 presentation of these 1200 images to get accurate driving over single laned test road
- Time consuming to generate synthetic images
- Difference between synthetic and real road images lead to poor performance in actual driving situations
- More driving situations such multilane and off road driving need to be addressed.
Improved “On-the-fly” Approach

- Use realtime images from camera as input and human steering as desired output to train back propagation model
- Downsides: 1) model never presented misalignment situation because human typically drives correctly 2) overlearning from repetitive inputs due to the scope of test drive roads available
- Solution: take an image and shift 7 times to the left and 7 times to the right in 0.25m increments to create 15 new images by software; human steering direction will shift accordingly to be sync
Shifted New Images

10 images with least errors and 5 randomly chosen images are replaced by the newly shifted 15 images in the training dataset.
Results

- Takes 50 epoches on the model
- Trained within 5 min at 4 mph driving speed on the test road
- Then Navlab can drive the same test road correctly but also roads never encountered under different conditions.
Future Work

- Increase speed limit
- Develop a single model to detect for different road situations