End-to-end-Learning Ansatz für autonomes Fahren im Miniatur Wunderland

Nils-Ole Bickel, Michel Brüger

26. Februar 2020

Inhaltsverzeichnis

ToDo:
evtl Abstract schreiben
- end-to-end learning erklären
– netzt sagt für übertragene Bilder lenkwinkel voraus
– übermittelte Lenkwinkel als Label, Netz sagt für übertragene Bilder Lenkwinkel voraus

1 Hardware

1.1 Raspberry Pi 4

1.2 Google Coral

1.3 Das Auto

1.4 Der Trainingsrechner

2 Software

2.1 Auf dem Raspberry Pi verwendete Software

2.1.1 Python

2.1.2 Tensorflow lite

2.2 Auf dem Auto verwendete Software

- irgendwie schickt es einen Videostream...

2.3 Auf dem Trainingsrechner verwendete Software

2.3.1 Anaconda

- Virtual Environments
- Python
- Tensorflow
- Tensorflow lite converter (heisst der so?)

3 Das Netz

3.1 Implementierung

Das Netzwerk besteht aus 11 Layern. Bei 6 davon handelt es sich um Convolutional Layer. Die anderen 5 sind fully-connected Layer. Die Convolutional Layer sind für die

Featureerkennunng verantwortlich. Die ersten 4 haben einen Kernel der Größe 5x5 und Strides der Größe 2x2. Die anderen Convolutional Layer haben einen Kernen der Größe 3x3 und Strides der Größe 1x1. Anschließend kommen die 5 fully-connnnected Layer. Das als Ergebnis beschreibt den Lenkwinkel in einer abstrakten Form. Genauer ist es die Länge des Vektors (X, Y), die der Hall Sensor, an der Vorderachse des Auto, bei diesem Lenkeinschlag misst.

Als Vorlage für das Netzwerk wurde das Netzwerk aus dem Paper [1] verwendet. Unsere Bilder haben eine Größe von 300x800. Dies ist deutlich größer als im Paper weshalb das Netzwerk um weitere Layer ergenzt wurde.

3.2 Training

3.2.1 Trainingsdaten

Die Trainingsdaten wurden mit Hilfe des Autos aufgenommen. Das Auto ist entlang eines Drahtes nach dem Faller Car-System durch das Miniatur Wunderland gefahren. Dabei wurden Video-Stream und zugehörige Werte des Hall Sensors aufgezeichnet. Die obere Hälfte des Bildausschnitts wird verworfen. Dies führt dazu, dass großteils die Straße auf den Bildern zu sehen ist. Die Hoffnung ist, dass dadurch das Netzwerk nicht durch Häuser oder Deckenelemente im Miniatur Wunderland abgelenkt wird und somit allgemeiner eingesetzt werden kann.

Die Bilder werden als BGR-Bilder mit 3 Kanälen zusammen mit den Daten zum Lenkwinkel in einer H5-Datei gespeichert. Dabei ist zu beachten, dass die Bilddaten bereits als float32 Werte gespeichert werden. Sonnst muss dieser rechenaufwändige Schritt während des Trainings durchgeführt werden. Eine Normierung der Bilddaten auf einen Wert zwischen 0 und 1 könnte zu einer Verbesserung der Ergebnisse vom Netzwerk führen. Allerdings würde es auf die Zeit, bis ein Bild bearbeitet wurde deutlich erhöhen, weshalb wir uns dagegen entschieden haben.

3.2.2 Optimierung

Das Netzwerk wird dahingehend Trainiert die durchschnittlichen Quadratischen Abweichung (mse) zwischen dem Ergebnis des Netzes und den gemessenen Daten zum Lenkwinkel zu minimieren. Dies führt dazu, dass große Abweichungen besonders stark Gewichtet werden und kleinere nur Geringer.

3.2.3 ...

4 Ergebnisse/Fazit

- Genauigkeit der Berechneten Lenkwinkel noch nicht erprobt...