Object detection based DuckieTown agent on INTEL Movidius Neural Compute Stick

Objektum felismerést megvalósító DuckieTown ágens INTEL Movidius eszközön

Antal Balázs
a.balat1993@gmail.com

Wippelhauser András
wandris1210@gmail.com

Meleg Eszter
eszto.mg@gmail.com

Faculty of Electrical Engineering and Informatics
Budapest University of Technology and Economics

Abstract

In 2016 DuckieTown started as a class at MIT with the goal to introduce students into the field of artificial intelligence and robotics. Since then the project has grown into a worldwide initiative to realize a new vision for AI/robotics education by supporting innovative solutions. Our main goal with this project is to implement an object detection network based on the guidence of the original course material and enhance its performance with a dedicated hardware accelerator for deep neural network inferences. The greatest challenges during the process are considered to be the difficulities rooted in the diversity of the environments which we attept to solve as best we can. During the implementation we are seeking to apply as optimal solutions as possible while trying to obtain new and useful experiences for our further studies. The following document offers an insight of our work, including the obstacles we faced with and the results we reached.

Abstract

2016-ban a DuckieTown egy az MIT által szervezett kurzusként indult azzal a céllal, hogy bevezesse a tanulókat a mesterséges intelligencia és a robotika világába. Azóta a projekt egy világméretű kezdeményezéssé nőtte ki magát annak érdekében, hogy új szemléletet alakítson ki az AI és a robotikai oktatás terén az innovatív módszerek támogatásával. Az eredeti kurzus útmutatását alapul véve a fő célunk ebben a projektben egy olyan objektum detektáló neurális háló elkészítése, melynek teljesítményét egy mély neurális hálók predikcióihoz alkalmazott, dedikált hardware alapú gyorsítóeszköz növeli. A legnagyobb kihívást a megvalósítás során a környezetek sokszínűségéből adódó nehézségek jelenthetik, melyekre igyekszünk legjobb tudásunk szerint megoldást találni. Az implementáció során arra törekszünk, hogy a lehető legoptimálisabb megoldásokat alkalmazzuk miközben igyekszünk új és hasznos tapasztalatokat szerezni további tanulmányainkhoz. Az alábbi dokumentum betekintést nyújt a munkánkba, beleértve a nehézségeket amikkel szembenéztünk valamint az elért eredményeket.

vitmav45 Deep Learning in Practice with Python and LUA (Autumn 2019), BME.

1 Introduction

The following project is made as a team work assignment for the class Deep Learning in Practice with Python and LUA, autumn semester 2019. Our choice was the implementing of a DuckieTown agent which is able to detect specific objects in a well determined environment and uses an edge based accelerator for faster decision making progress. Our baseline was the excercise of the official Duckietown project documentation Unit B-5, which gives a YOLO based convolutional network as a solution of the problem.

1.1 Field and previous solutions

The base objective of the originating article is to create a labeling system which is able to recognize various types of objects related to the duckietown environment. The recognized objects are duckie bots, duckies, stop signs and so on. The study chose YOLO as learning architecture, which is basically a convolutional neural network. Using a convolutional neural network is quite adequate because it efficiently reduces the number of used neurons by scanning the image in parts. The YOLO is said to be one of the most efficient and fastest architecture for image labelling. The study made on our baselin suggests the network achieved a 70 percent average IOU during validation and demonstrates real-time detection of close objects on unseen data. (link)

2 Datasets

The datasets being used in the project are the precleaned, labelled pictures from the original DuckieTown repository. The collection of the dataset is performed manually. The pictures are taken by real duckiebots in real in-lab duckietown environments. The labelling of the dataset is basically done manually, supported by various utility scripts. Each picture has a text file too which contains the parameters of the objects being depicted. This way by merging the visual and text data we get the annotated pictures for training. In general around 1000 pictures were taken, after the cleaning process around 420 pictures are considered being useful for teaching the network. In order to enlarge the dataset, the Augmentation python framework is used. The Augmentation framework changes different properties of the original picture, like saturation or contrast. Using this technique we can expand our dataset. The originating article used around 4000 images created by this technique. The dataset is split into validation, test and training sets in the proper folders respectively.