Skript ML Projekt

**1. Folie**

Zu beginn unser Projekt:

Der Husky Roboter, der eine Kamera besitzt, soll auf einfache Kommandos reagieren, die über die Kamera entgegengenommen werden. Erster Ansatz dafür war ein Modell einzulernen, dass auf die 4 Phasen einer Ampel reagiert.

**2. Folie**

So haben wir uns erst an die Suche nach einem geeigneten Datensatz gemacht. So sind wir ziemlich schnell auf Kaggle gestoßen. Dort hat man eine große Auswahl an Datensätzen, die auch oft gelabelt sind und mit weiteren Informationen versehen sind.

Erste Gedanken: Größe des Datensatzes. Probleme eines zu kleinen Datensatzes:

* Overfitting: statt allgemeine Trends zu lernen und auf unbekannte Daten schlecht
* Vielfalt: Auch das Problem unseren ersten Datensatzes. Abdeckung aller Lichtverhältnisse, Winkel, Hintergründe und sehr wichtig bei Objekterkennung die **Größe.**
* Genauigkeit: Durch zu wenig Datenpunkte geringe Sicherheit

**3. Folie**

Wir haben uns zu beginn für den Datensatz traffic lights entschieden. Die Daten waren bereits aufgeteilt in Trainigsdaten und Validationsdaten mit jeweils nochmals einer Unterteilung in die 4 Klassen.

Einen Datensatz mit 4 Klassen haben wir sowieso gesucht, da am Ende die Aufgabe war mit jeweils einer Klasse **Go Stop Left Right** zu realisieren.

Dies sind Beispielbilder aus dem Datensatz. Die Probleme die wir hier hatten sind folgende:

* Es war ein sehr kleiner Datensatz mit gerade mal 100 Bilder was natürlich schnell ins Overfitting führt und schlecht generalisiert wird
* Die Bilder waren bereits ausgeschnitten was dazu führt dass das Modell sie nicht in einer Umgebung erkennt mit einem Hintergrund, oder nur teilweise zu sehen sind oder noch wichtiger nicht erkennt bei einer Größenänderung

Kleine Anmerkung: aufgrund eines Kommas im Dateinamen konnten manche Bilder nicht Eingelsen werden, wenn man anfangs dachte es liegt am Code lag es am Ende doch am Komma, womit man sich mit einem einfachen Python Skript Abhilfe schaffen kann um die Dateinamen zu ändern. Würde manuell zu lange dauern.

**4. Folie**

Neuer Datensatz mit Dosen, Glasflaschen, Milchbehälter und PET Flaschen.

Vorteile hier ist der komplexe Hintergrund, sehr großer Datensatz und dass sich die zu erkennenden Objekte nicht immer in der Mitte befinden.

**Zusatz:** Textdateien mit YOLO Format vorhanden wobei

Die erste Zahl `1` repräsentiert die Klassen-ID. In diesem Fall handelt es sich um die Klasse mit der ID 1

2. Die zweite Zahl `0.5703125` gibt den X-Koordinaten-Mittelwert des erkannten Objekts relativ zur Breite des Bildes an.

3. Die dritte Zahl `0.5366032210834554` gibt den Y-Koordinaten-Mittelwert des erkannten Objekts relativ zur Höhe des Bildes an.

4. Die vierte Zahl `0.1796875` gibt die tatsächliche Breite des erkannten Objekts relativ zur Breite des Bildes an.

5. Die fünfte und letzte Zahl `0.17715959004392387` gibt die tatsächliche Höhe des erkannten Objekts relativ zur Höhe des Bildes an.

Einbindung von YOLO jedoch nach Recherche sehr komplex und komplett neues Modell nötig, jedoch ist es nicht von nutzen zu wissen wo sich das Objekt befindet, für unseren Fall reicht es, dass ein Objekt der 4 Klassen erkannt wird.

**5. Folie**

Der vorhandene Code wird an den neuen Datensatz angepasst und erste Tests werden damit durchgeführt.

Erhebung eigener Daten zum Beispiel mit einem Video in dem alle 4 Klassen vorkommen und Tests Frame für Frame ob Objekt erkannt wird.