

Vortrag

Aufgabe/Anforderung

- Autonomes Erkennungs/Überwachungssystem
- soll wild tiere erkennen und benachrichtigung+bild schicken
- nicht nur bewegung erkennen sondern auch was -> NN/DeepLearning
 - nur relevante daten senden/benachrichtigen

kurz was is NN:

- **was:** NN lernt in gr Datenmenge Zusammenhänge und kann diese generalisieren so das es sie auch für neue daten anwenden kann
- **wie:** input daten mit zugehörigen outputs (hier gelabelte bilder) in Modell, dieses lernt iterativ die zusammenhänge
- vgl normal programm vs ml algorithmus

Equipment:

- raspi für steuerung und server conn,
- ncs2 für inferenz,
- camera modul für tag und nacht(infrarot) aufnahmen

NCS2 und MYRIAD Chip:

- grob die funktionsweise (hardware schnelle berechnung von NN operationen)
- Anwendungen/Vorteile für edge Systeme
 - vgl zu cloud basiert zb in handy
- workflow mit opencv toolkit, mögliche frameworks, vgl zu herkömmml workflow
- asynchrone inferenz

Training des Models

- Sammeln und aufbereiten der Daten
 - von Open Images folgende klassen geladen: Deer, ...
 - validierungs split
 - Augmentierung: gegen overfitting (netz lernt train datas nur auswendig)
 - Graustufen: für infrared
- Auswahl und Training des Modells
 - CNNs -> für reine klassifikation
 - Objekterkennung verwendet CNN + box erkennung:
 - Single Shot vs Two Stages
 - mit Tensorflow Object Detection Api als Framework mit folgende Modellen trainiert

- SSD und RCNN mit verschiedenen Backbone CNNs
- Evaluierung des Trainings
 - Loss/mAP (während Training) auf *Test Daten*: wie wird berechnet/was bedeutet
 - Inferenz Ergebnisse (nach Training) auf *Val Daten*
 - Geschwindigkeit vs Genauigkeit

Applikation

- Einfrieren des Models und Convertierung zu IR
- Logik mit Bewegungsmelder, speichern der Bilder, parallele und asynchrone Inferenz
- Tcp Verbindung, wann senden
- Real World Ergebnisse zeigen