

NAVIGATE

(Agent Maulwurf)



Gliederung

- Zielsetzung / Idee / Erläuterung
- Machine Learning
- Umsetzung
- Ergebnisse



Zielsetzung / Idee

- Umsetzung eines “Agents”, welcher autonom durch Tunnel / Röhrensystem navigiert
- Nutzung von Unity’s “ML-Agents”-Toolkit → Machine Learning / Deep Reinforcement Learning
- Forschungsprojekt



Navigation

Was bedeutet Navigation?

→ Wikipedia sagt unter Anderem folgendes:

“Sie [die Navigation] kann [...] definiert werden als **Sich-Zurechtfinden in einem topografischen Raum**, um einen gewünschten Ort zu erreichen.”

(Wikipedia-Artikel zur Navigation)

→ **Wie gut kann eine Maschine das “Sich-Zurechtfinden” lernen?**



Machine Learning

Grundsätzlich:

- Aufgaben werden anhand von Beispielen (Training) gelernt und das Gelernte kann später auf andere Beispiele angewendet werden (Test). Die entsprechenden Daten werden im sogenannten "Model" gespeichert.

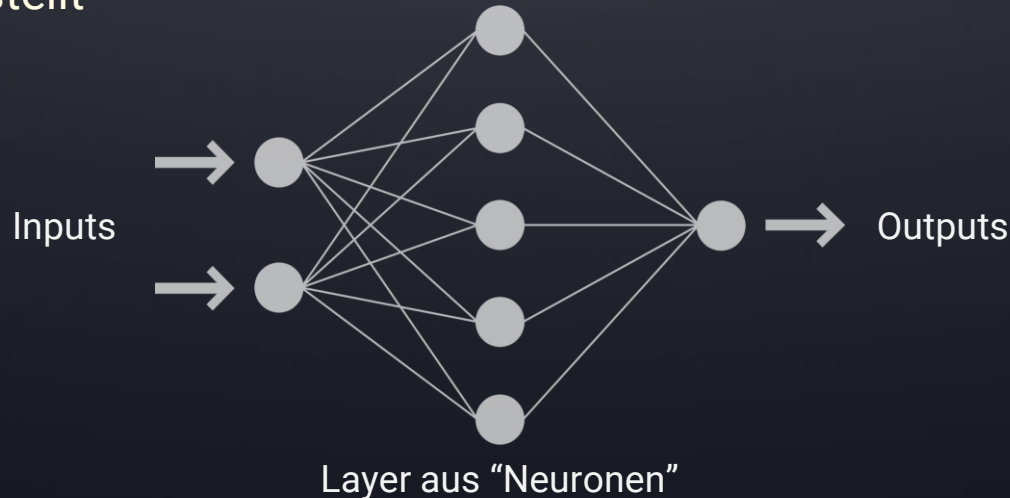


Deep Learning

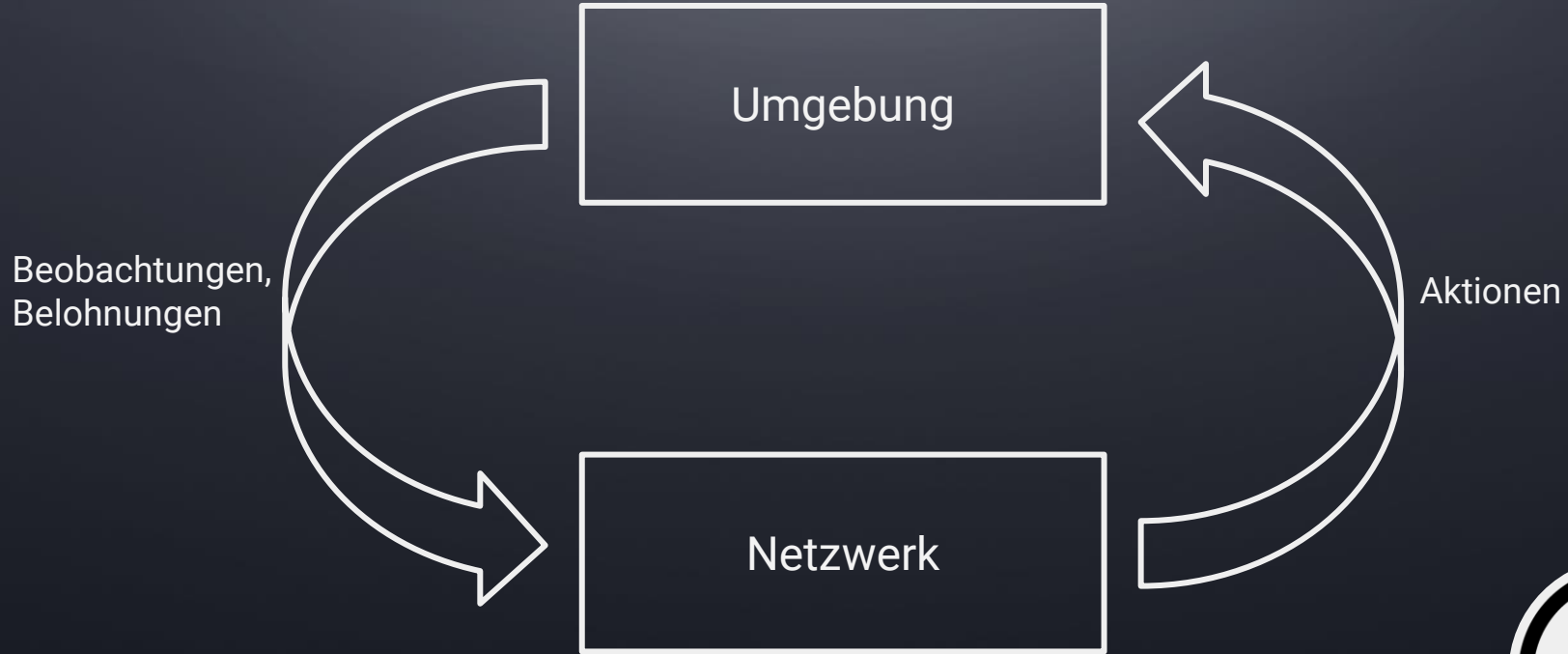
Künstliches Neuronales Netz:

- Modellhafte Nachbildung biologischer Neuronen in einem Netzwerk.

→ Das künstliche neuronale Netz wurde in unserem Projekt durch ML-Agents in tensorflow erstellt



Deep Reinforcement Learning



Umsetzung

Plattformen



TensorFlow



unity



Umsetzung

Belohnungssystem

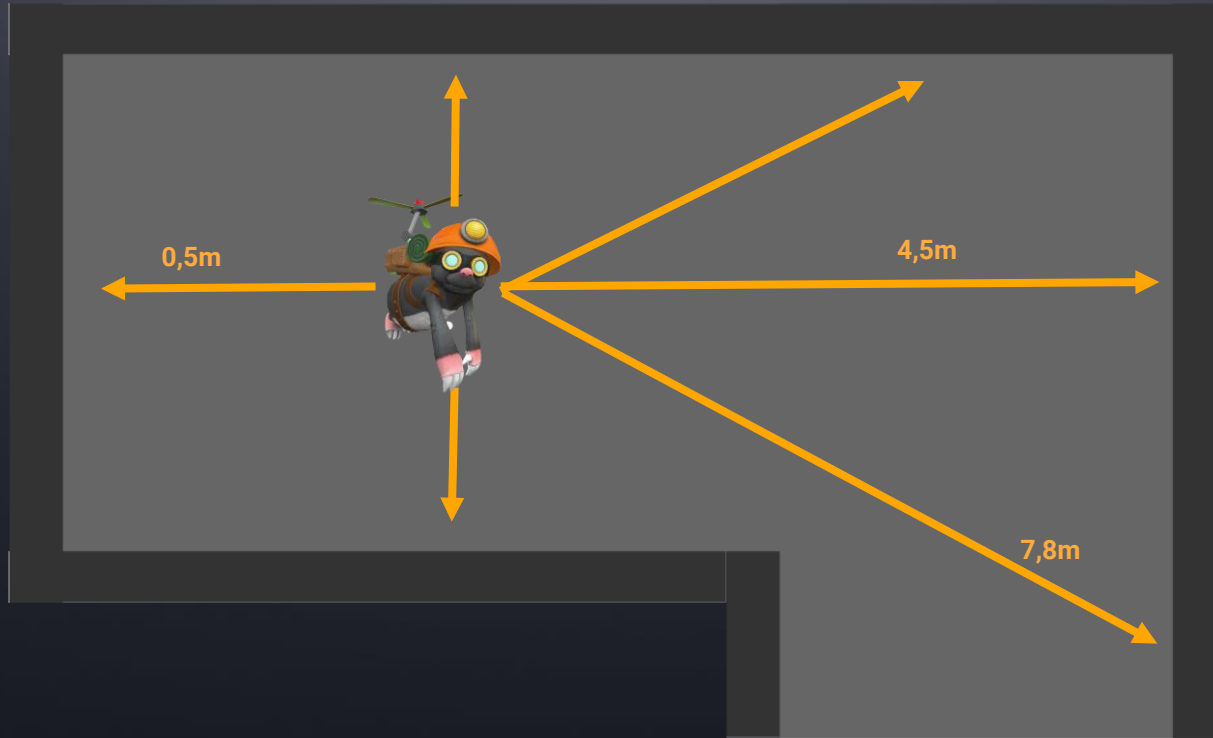
+1

-1



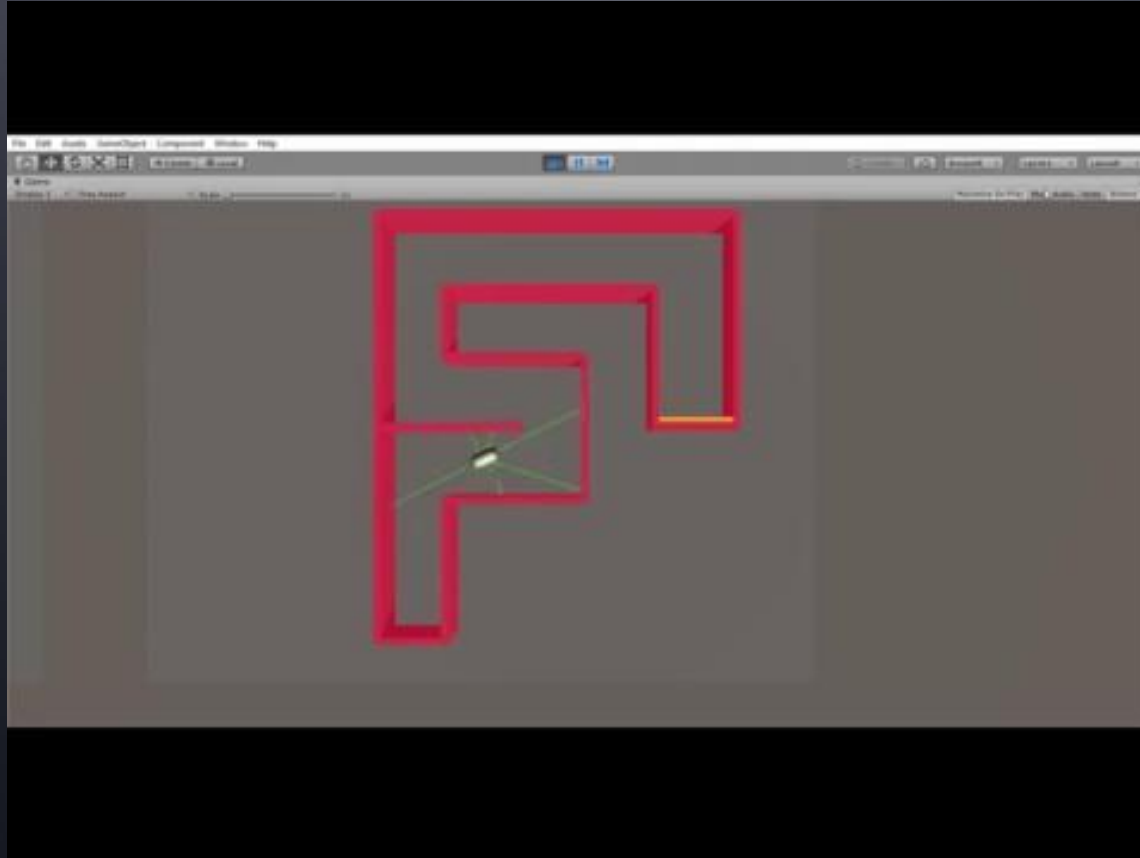
Umsetzung

Phase 1: 2D



Umsetzung

Phase 1: 2D Video



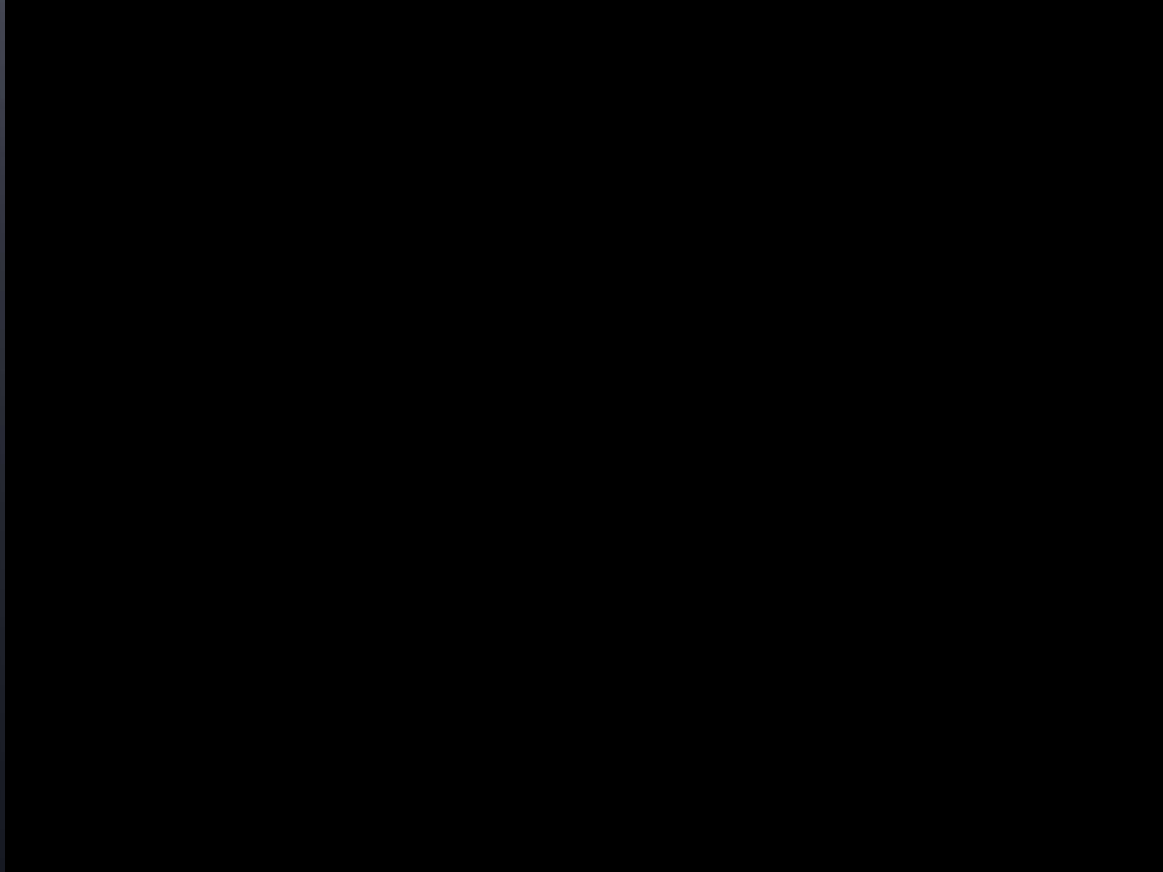
Umsetzung

Phase 2: 3D



Umsetzung

Phase 2: 3D



Umsetzung

Phase 2: Visual Observations:

- Inputs sollen keine Entfernungen mehr sein
- stattdessen sollen Bilder übergeben werden
- mit diesen soll der Agent trotzdem navigieren können (Helligkeit und Kanten)



Umsetzung

Phase 2: Visual Observations:

- Texturen sind wichtig für Orientierung
- Trotzdem soll der Agent sich in verschiedenen Umgebungen zurecht finden

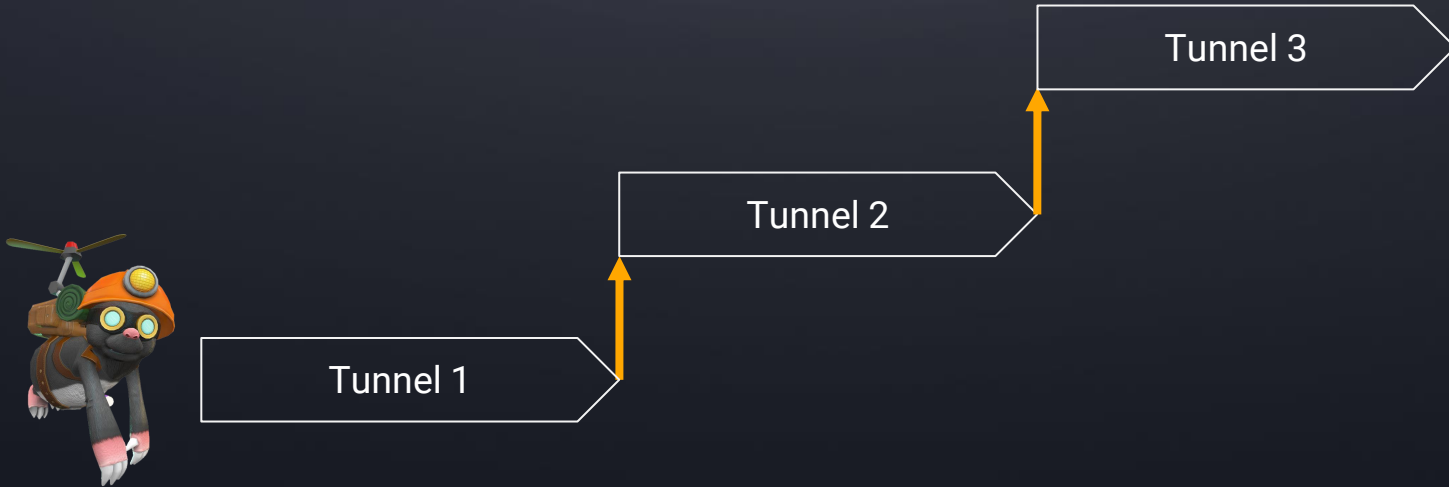
→ ständige Änderungen der Texturen während des Trainings zur Generalisierung



Umsetzung

Phase 2: Curriculum Learning:

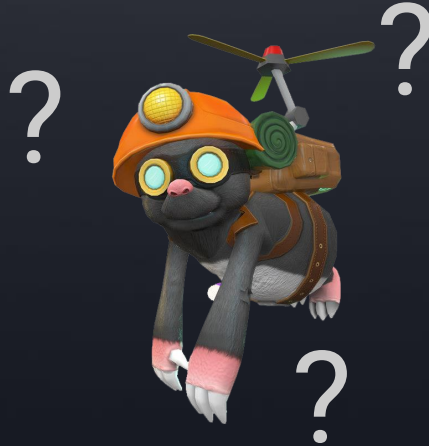
- komplexe Aufgaben können stufenweise erlernt werden
- ist eine Bedingung erfüllt (z. B. Aufgabe geschafft), beginnt die nächste Stufe



Umsetzung

Phase 2: Curiosity:

- es werden keine bzw. wenige Rewards explizit von uns vergeben
- eine intrinsische Motivation wird eingeführt
- der Agent bekommt Rewards wenn “unvorhergesehene” Ereignisse eintreten
- so wird der Agent angehalten zu Erkunden - er ist neugierig



Umsetzung

Phase 2: Recurrent Neural Network:

- Der Agent (das Netzwerk) hat ein Gedächtnis in dem er Dinge abspeichern kann
- ein lang anhaltendes Kurzzeitgedächtnis (Long Short-Term Memory - LSTM)



Umsetzung

Phase 2: Imitation Learning:

- Der Agent (Student) “imitiert” die Aktionen eines Menschen (Teacher)
- so lernt er die Aufgabe zu lösen
- eine Abstraktion der Aufgabe ist hier schwieriger



Ergebnisse

Vector Observations	einfach, effektiv, ressourcenschonend
Visual Observations	aufwendig, ressourcenintensiv, keine sehr guten Resultate
Curriculum Learning	geeignet für komplexe Aufgaben, verringert Zeitaufwand
Curiosity	für Aufgaben bei denen es schwierig ist ein Belohnungssystem zu modellieren
Recurrent Neural Network	sehr ressourcenintensiv, Aufgaben bei denen Information später noch benötigt werden
Imitation Learning	nicht gut zur Abstraktion von Aufgaben geeignet



Ergebnisse

Das Spiel:

- Model über Raycasts trainiert
- unbekannte Umgebungen nach Training kein Problem
- auch Hindernissen kann ausgewichen werden
- stark wenn menschliche kognitive und physische Fähigkeiten versagen



Ergebnisse



Vielen Dank für Ihre Aufmerksamkeit!

