KI-Bot für Mario Kart Game

Studiengang: BSc in Informatik | Vertiefung: Computer Perception and Virtual Reality
Betreuer: Prof. Dr. Jürgen Eckerle
Experte: Dr. Federico Flueckiger (Eidgenössisches Finanzdepartement EFD)

Ein Agent soll selbständig lernen eine Strecke im Mario Kart Spiel zu befahren. Im Umfang dieser Arbeit wurde eine künstliche Intelligenz entwickelt die ausschliesslich durch die Bildausgaben des Mario Kart Spiels lernen soll eine Strecke zu befahren. Diese Aufgabe wurde mit dem Q Learning Verfahren für ein vereinfachtes Spiel, «Vector Racer», gelöst.

Reinforcement Learning

Beim Reinforcement Learning interagiert ein Agent mit seiner Umgebung indem er verschiedene Aktionen ausführt. Mit Sensoren kann der Agent seine Umgebung wahrnehmen um zu entscheiden, welche Aktion er ausführen soll. Mit jeder Aktion verändert der Agent seine Umgebung und erhält eine Belohnung. Reinforcement Learning ist ein Lernverfahren mit dem Ziel, eine Aktionsfolge zu finden, die den Erwartungswert der akkumulierten Belohnung maximiert. In dieser Arbeit wurde Q-Learning verwendet, bei dem die sogenannte Q-Funktion jedem Zustands-Aktions-Paar die erwartete Belohnung zuordnet. Die Q-Werte können in einer Tabelle abgelegt werden. Die Q-Werte sind zu Beginn nicht bekannt, sondern müssen schrittweise durch aufeinanderfolge Lernzyklen, bei denen der Agent wiederholt Aktionssequenzen wählt, approximiert werden. In unserem Beispiel entspricht eine Aktionsfolge das Befahren einer (teilweise zufällig gewählten) Route.

Vector Racer

Vector Racer dient als vereinfachte Alternative zu Mario Kart. Da es in keiner simulierten Umgebung laufen muss und mehr Kontrolle über die Wahl der Strecken und den Spielzustand vorhanden ist, eignet sich das neue Spiel gut als Alternative. Jedem Spieler wird ein Fahrzeug zugewiesen, welches die vorgegebene Strecke am schnellsten absolvieren muss. In jedem Schritt können beide Komponenten des Geschwindigkeitsvektors des Fahrzeuges um +1, -1 oder +0 angepasst werden, was insgesamt 9 Möglichkeiten bietet. Im Vergleich zu Mario Kart unterscheiden sich die möglichen Aktionen sowie die Spielfeldansicht. Obwohl die Aktionen sich unterscheiden, ist dies für den Agenten irrelevant. Der Agent lernt mit den vorgegebenen Aktionen und führt dieselben wieder aus. Die Spielansicht welche als Spielzustand verwendet wird, muss ähnlich dargestellt werden. Bei Vector Racer wird die Spielansicht ähnlich gewählt wie bei Mario Kart. Hierbei werden die Spielfelder vor dem Fahrzeug in einem Raster betrachtet, wobei dessen Skalierung und Grösse angepasst werden kann.

Praktische Versuche

Während dem Projekt wurden viele praktischen Versuche unternommen. Dabei wurden die Parameter des Lernalgorithmus sowie die Skalierung und Grösse der Spielansicht verändert. Eine grosse Hürde bei den praktischen Versuchen war unter anderem die Belohnung des Fahrzeuges. Es hat sich schliesslich herausgestellt, dass der Lernalgorithmus schneller zu einem Ergebnis führt, wenn das Fahrzeug abhängig von der Geschwindigkeit für die Schritte belohnt wird und beim Verlassen der Strecke bestraft wird. Für die Versuche wurden zwei Strategien verwendet um die Aktionswahl des Agenten zu bestimmen: softmax und epsilon-greedy. Bei softmax wird zufällig eine der möglichen Aktionen gewählt, wobei Aktionen mit einer besseren Bewertung mit grösserer Wahrscheinlichkeit gewählt werden. Bei epsilon-greedy werden am Anfang des Lernprozesses die Aktionen zufällig gewählt, wobei alle dieselbe Wahrscheinlichkeit haben. Gegen Ende des Lernprozesses wird die Aktion mit der Besten Belohnung öfters ausgewählt.

Fazit

Dem Agenten sind zahlreiche Erfolge gelungen, bzw. er konnte die vorgegebenen Strecken erfolgreich lernen. Der Rechenaufwand für den Lernprozess ist sehr hoch, bei bestimmten Strecken benötigte es teilweise Stunden. Der Rechenaufwand ist stark abhängig von der gewählten Parametrierung sowie die Belohnungsfunktion. Er lässt sich reduzieren wenn die Q-Tabelle durch ein neuronales Netz ersetzt worden wäre. Der Agent ist sehr empfindlich auf die Parameter wie die Spielfeldgrösse. Sind die Parameter falsch eingestellt, wird der Agent überhaupt keinen Lernerfolg verzeichnen können. Bei den Strategien softmax und epsilon-greedy hat sich herausgestellt, dass bei softmax die Rechnungen aufwänderig sind, der Agent jedoch schneller lernt. Der Aufwand beider Strategien gleicht sich aus.

Casimir Benjamin Platzer