

Projektarbeit zum Robo-Spot

von Phil Vossieg und Felix Spangenberg

Projektarbeit zum Robo-Spot

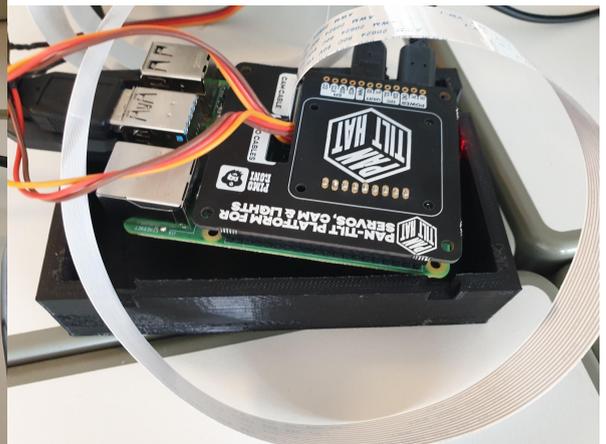
Die Idee	2
Das Design	2
Die Materialien	4
Raspberry Pi 4 Modell B	4
Teensy LC	4
Servo Motor HDKJ D3015	4
Raspberry Pi Cam V2	4
3D Druckteile	4
Pan-Tilt-Hat-Plattform	5
Funktionsweise	5
Vorgehen	5
Probleme	6
Fazit zum Projekt	6

Die Idee

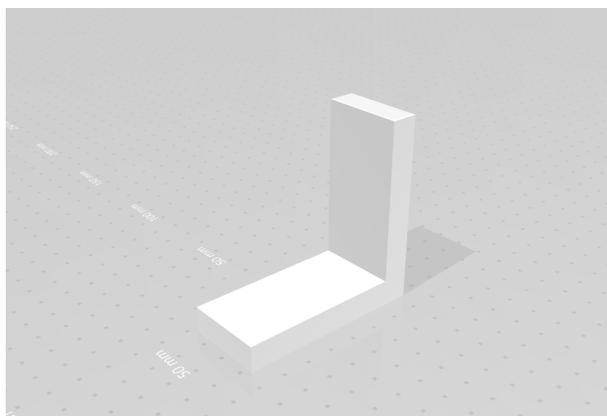
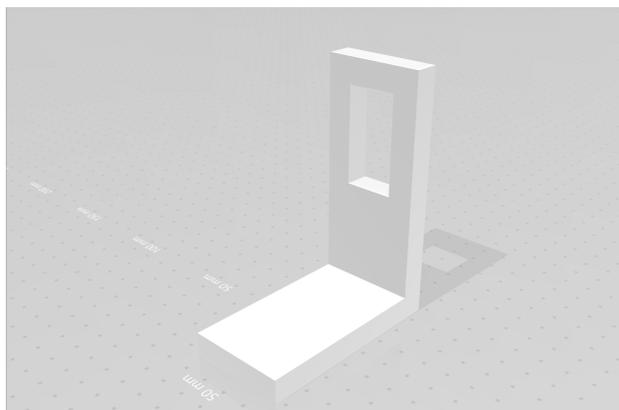
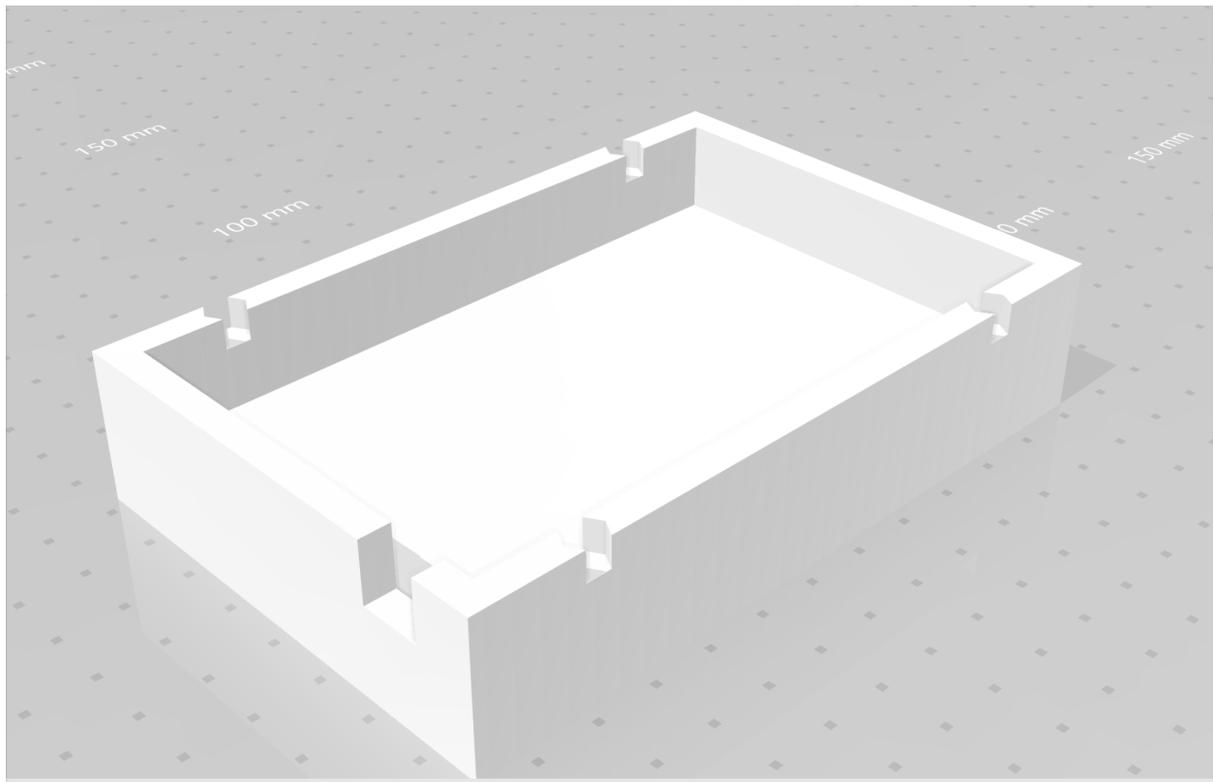
Die Idee hinter dem Robo-Spot war, einen Roboter zu bauen, der Bewegungen erkennt und diese verfolgen kann. Somit sollte der Roboter immer in Richtung des bewegenden Objekts gucken. Inspiriert wurde dies, durch das Fehlen einer automatischen Followspot im Bereich der Bühnentechnik. In unserem Fall haben wir allerdings die Lampe weggelassen, da wir nur die Grundlage für eine solche Technik haben wollten. Allerdings könnte man jederzeit eine kleine Lampe hinzufügen.

Das Design

Das Design besteht aus einem sogenannten Zwei-Achsen-Gimbal welche sich um die horizontale und vertikale Achse bewegen kann. Auf das Gimbal haben wir dann eine kleine Kamera montiert, welche Daten an den Raspberry Pi liefert. Der Raspberry Pi liegt in einer externen "Schale", die mit Hilfe von 3D Drucker angefertigt wurde. In dieser Schale befindet sich ebenfalls die Motortreiber Platine, die auf dem Raspberry Pi drauf ist.



Projektarbeit zum Robo-Spot



Die Materialien

[Raspberry Pi 4 Modell B](#)

Das Nutzen dieses Raspberry Pi Modells, hat keinen speziellen Grund. Der Raspberry bildet zusammen mit der Pi-Cam eine gute Grundlage für unser Projekt. Da wir so oder so einen Raspberry für unsere Anforderungen gebraucht hätten, haben wir uns entschieden, diesen hier zu nutzen.

[Teensy LC](#)

Der Teensy hat zwar in der letztendlichen Projektarbeit keine Anwendung gefunden, da er kaputt gegangen ist. Allerdings haben wir ihn trotzdem hier genannt, weil es zum einen trotzdem möglich gewesen wäre, das Projekt mit dem Teensy zu vollenden und andererseits möglich gewesen wäre unseren Ablauf zu erläutern. Dort war der Teensy nämlich ein wichtiger Schritt.

[Servo Motor HDKJ D3015](#)

Die Servomotoren, die wir genutzt haben, waren noch herumliegende Motoren. Sie waren allerdings für unser Vorhaben ideal geeignet und somit mussten wir keine neuen Motoren bestellen.

[Raspberry Pi Cam V2](#)

Die Raspberry Pi Cam dient der Objekterkennung. Genommen haben wir diese, weil sie mit dem Pi kompatibel ist und wir so nicht weitere umständliche Umwege gehen mussten.

[3D Druckteile](#)

Wir haben auf 3D Druckteile zurückgegriffen, um so den Roboter nach unseren Vorstellungen zu gestalten. Hätten wir auf bereits vorhanden Materialien zurückgegriffen, wären wir in unserem Design und unserer Gestaltung des Bots stark eingeschränkt gewesen.

Pan-Tilt-Hat-Platform

Diese Platine war unser Notrettung. Wir haben sie verwendet, weil wir sie noch Zuhause hatten und somit sofort nutzen konnten. Sie hat keine Funktion, außer dass sie die Schnittstelle zwischen dem Raspberry Pi und den Servos bildet.

Funktionsweise

Der Raspberry Pi, welcher als Computer agiert, steuert die Pan-Tilt-Hat-Platform an. Diese steuert wiederum die Servos an. Die Objekterkennung läuft somit über den Raspberry und die Pan-Tilt-Hat-Platform dient lediglich der Schnittstelle zwischen Servos und Raspberry. Das Programm erkennt ein Objekt und trackt es mit Hilfe von einer [Object Detection](#). Dieses getrackte Objekt wird dann auf dem Display angezeigt.

Gleichzeitig gibt somit der Raspberry Pi die Informationen an die Servos bzw. sagt diesen, dass sie sich drehen sollen und um wie viel Grad sie dies tun sollten, sobald das Objekt (z.B ein Mensch) sich zu weit an den Rand des Bildes nähert. Somit bleibt das Objekt immer relativ zentral im Bild.

Vorgehen

Zu Beginn hatten wir erst überlegt, was eine gute Idee wäre. Wir haben uns dann für das Bauen eines Trackers entschieden, der in der Lage sein soll, Bewegungen zu erkennen und diese zu verfolgen. Dann ging es erstmal auf die Umsetzungssuche. Da wir auf dem Bereich unerfahren waren, mussten wir überlegen, wie wir es hinkriegen können, unsere Idee umzusetzen. Wir entschieden uns, ähnliche Projekte zu begutachten und Vergleiche mit unserem zu ziehen. Wir fanden letztendlich ein Projekt, welches eigentlich für eine [Sentry Gun](#) gedacht war. Es verfolgte schließlich ein ähnliches Ziel wie wir, nämlich das Tracken eines Objektes.

Wir fingen an uns ein Design für den Robo-Spot auszudenken und kamen schnell zu dem Schluss, dass wir 3D Druck Teile verwenden um unsere Vorstellung bestmöglich umzusetzen. Um Zeit zu sparen, entschieden wir uns dazu ein bereits vor gelerntes [Tensorflow](#) Programm zu nutzen, welches bereits Objekte erkennen und identifizieren konnte. Damit dieses Programm problemlos laufen kann, nahmen wir einen Raspberry Pi. Da wir nun schon einen Raspberry hatten, nahmen wir direkt eine Pi Cam um uns Umwege zu ersparen.

Für das Gimbal selbst, nahmen wir noch zwei herumliegende Servomotoren, die die Funktion perfekt erfüllen sollten.

Nachdem der Tensy kaputt ging, aufgrund einer zu hohen Spannung (unklar wo diese herkam) , mussten wir neu planen. Da dieses Problem erst kurz vor Abgabe des Projektes auftrat, musste schnell eine neue Lösung gefunden werden. Wir griffen somit auf eine andere Art von Motortreiber zurück, welche wir noch Zuhause rumliegen hatten und somit sofort nutzen und einbauen konnten. Wir schlossen also die Servos an die Platine an und die Platine an der Raspberry Pi und somit war die Hardware fertig.

Als Code entschieden wir uns auch einen bereits angefertigten [Tensorflow](#) Code zu nutzen welcher schon eine leichte [Object detection](#) gelernt hatte. Wir optimierten den Code, wodurch eine bessere [Object detection](#) entstand. Anschließend schrieben wir einen Code, welcher es möglich machte, dass durch die [Object detection](#) eine Person oder ein Objekt getrackt werden kann. Danach mussten wir noch einen Code schreiben, welcher nun dafür sorgte, dass die Servomotoren richtig angesteuert werden, wobei einige GitHub projekte uns zur Hilfe kamen und wir Teile aus diesen nutzen konnten.

Nach Hochladen des gesamten [Code packages](#) konnten wir somit das erste Mal einen erfolgreichen Testlauf absolvieren.

Unser Robo-Spot war das erste Mal in der Lage ein bewegendes Objekt zu verfolgen.

Probleme

Ein entscheidendes Problem ist aufgetreten, das unsere Planung erstmal über den Haufen warf. Unser Teensy und somit auch die Motortreiber Platine, die eigentlich zusammen die Schnittstelle zwischen Raspberry und Servos bieten sollte, ist bei einem Testlauf kaputt gegangen und funktionierte nicht mehr.

Somit haben wir auf herumliegende Materialien zurückgegriffen und improvisiert. Dadurch kam die Pan-Tilt-Hat-Plattform ins Spiel.

Im Nachhinein hatte das den Vorteil, dass wir durch diese Plattform ein kleineres Design hatten als mit dem Teensy und der Platine.

Ein weiteres Problem war danach, dass der Code auf dem Raspberry Pi lief, jedoch die Motoren nicht angesteuert wurden. Nachdem wir uns den Code nochmal angeschaut haben, welcher keine Fehler enthielt, suchten wir im Internet nach Lösungen, wobei wir auf den [SPI](#) Port sowie den [I2c](#) Port gestoßen sind. Beim Aufsetzen des Raspberry Pis hatten wir diese nicht aktiviert, welche jedoch für das Projekt aktiv sein mussten, um mit dem Pan-Tilt-Hat funktioniere zu können. Nachdem wir die Ports Aktiviert hatten lief das Programm und der Roboter.

Fazit zum Projekt

Rückwirkend war das Projekt trotz seiner Komplexität sehr spannend und es hat uns Spaß gemacht es zu entwickeln.

Wir sind sehr zufrieden mit dem Ergebnis und denken, dass das Projekt noch Vielversprechendes in der Zukunft bringen könnte.