

Spotfinder - Entwicklung eines Lego Roboters zur autonomen Routensuche

Lasse Kramer, Elektromobilität
Otto-von-Guericke-Universität Magdeburg

Zusammenfassung—Führerlose Transportfahrzeuge nutzen zur Navigation fast ausnahmslos das Prinzip der Spurführung entlang einer Leitlinie. Niedrige Kosten sowie Robustheit und Zuverlässigkeit sind prägnante Vorteile. Da die Leitlinien fest auf oder in den Boden aufgetragen bzw. eingelassen sind, ist eine Änderung dieser nicht spontan möglich. Im Rahmen des Projektseminar Elektrotechnik/Informationstechnik (LEGO Mindstorms) der Otto-von-Guericke Universität Magdeburg wurde ein Roboter entwickelt, der eine neue Art der Navigation nutzen soll. In diesem Paper werden die Idee, konstruktive Umsetzung und der Programmablauf genauer erläutert. Nachfolgend wird auch auf die in den Erprobungen aufgetretenen Probleme eingegangen und beschrieben wie diese gelöst worden. So entstand ein Roboter, der in bis zu drei Suchradien seine direkte Umgebung nach farbigen Punkten scannt. Gefundene Markierungen werden angefahren und dann farbabhängige Aktionen ausgeführt. Zwei Abbruchkriterien führen dazu, dass der Roboter zurück zum Ausgangspunkt fährt. Es wird außerdem eine Einschätzung darüber gegeben, in wie weit der entwickelte Roboter, bzw. die neue Art der Navigation Einsatz in der Industrie finden kann.

Schlagwörter—Autonom, Farbmakierungen, FTF, Lego-Mindstorms, Matlab, Navigation

I. EINLEITUNG

AUTONOM arbeitende Roboter haben längst den Weg in Haushalte der ganzen Welt gefunden. Sie nehmen Menschen zeitintensive oder meist langweilige Arbeiten wie Saugen, Wischen oder Rasenmähen ab. Aber noch viel länger finden Roboter Einsatz in der Industrie. Hier übernehmen sie Arbeitsschritte die hohe und immer gleichbleibende Qualität voraussetzen, wie zum Beispiel das Schweißen oder Lackieren von Fahrzeugkarossen. Es treten aber auch monotone Tätigkeiten auf, die Roboter zuverlässig abarbeiten können. Hierzu zählt beispielsweise das Transportieren von Gütern innerhalb einer Fabrik und das anschließende Ausführen einer Aktion wie dem Einscannen von Barcodes zur Lagerquittierung oder dem Senden einer Bestandsmenge. Um Roboter Wege zurück legen zu lassen gibt es bereits Lösungen, auf die im Folgenden eingegangen wird. Diese bieten es aber nicht an, die Route zu ändern. Der Roboter legt demnach immer denselben Weg zurück. Es kann aber durchaus von Bedeutung sein die Route zu variieren. So ist es möglich Lagerorte flexibler zu gestalten oder auf auftretende Aufgaben mit hoher Priorität, z.B. die Überprüfung eines Bestands, spontan zu reagieren. Im Rahmen des Projektseminar Elektrotechnik/Informationstechnik (LEGO Mindstorms) vom 11.02.2019 bis 22.02.2019 an der Otto-von-Guericke Universität Magdeburg wurde sich diesem Problem angenommen und der Spotfinder entwickelt.

Ziel war es, einen Roboter zu entwickeln der durch Rotation um die eigene Achse den Boden in unmittelbarer Umgebung nach Farbmarkierungen scannt. Durch einen ausfahrbaren Farbsensor kann der Suchradius schrittweise bis zu zweimal vergrößert werden. Wird eine Farbe erkannt, fährt der Roboter diesen Punkt an und führt eine farbabhängige Aktion durch. Die Anordnung der Markierungen kann dabei nach jedem oder sogar während eines laufenden Durchgangs verändert werden. Es soll zwei Abbruchkriterien geben, zum einen der Fall, dass kein weiterer Punkt gefunden wird, zum anderen, dass der gefundene Punkt die Farbe Schwarz hat. Beim Eintreten einer dieser Fälle soll der Roboter ohne erneut scannen zu müssen, nur anhand abgespeicherter Drehwinkel und Strecken, autonom zurück zum Ausgangspunkt fahren.

II. VORBETRACHTUNGEN

Die Art des Spotfinders eine Route abzufahren würde wohl am ehesten bei sogenannten fahrerlosen Transportfahrzeugen (FTF) eingesetzt werden. Diese automatisierten und flurgebundenen Fördermittel werden meist in der Industrie zum Materialtransport eingesetzt. Eine einfache Möglichkeit des Fahrwerkaufbaus wird in Abbildung 1 gezeigt. Zwei starre Räder dienen zum Abstützen, das dritte Rad ist schwenkbar und wird in den meisten Fällen per Elektromotor angetrieben. So kann das FTF vor- sowie rückwärts fahren und ist linienbeweglich. Die Lastaufnahme findet je nach Anwendungszweck auf viele unterschiedliche Möglichkeiten statt und wird hier nicht weiter erläutert, da bereits Lösungen für die komplexesten Anforderungen existieren. Zu erwähnen ist nur, dass die Tragfähigkeit von einigen Kilogramm bis zu 50 Tonnen betragen kann, übliche Geschwindigkeiten liegen im Bereich von einem Meter pro Sekunde und die Einsatzdauer kann theoretisch 24 Stunden betragen. Zur Navigation wurden bereits diverse Verfahren entwickelt. Als robust und preiswert stellte sich dabei die Spurführung mittels kontinuierlicher Linie heraus. Zwei Arten werden im Folgenden genauer betrachtet. [1]

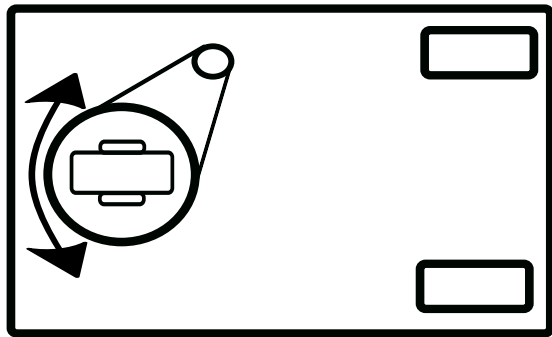
A. Induktive Leitlinie

Hierbei wird in den Boden eine von Wechselstrom durchflossene Drahtschleife verlegt, diese gibt die Fahrspur vor. Das Fahrzeug wird mit induktiven Sensoren ausgerüstet, welche dazu dienen, das wechselnde Magnetfeld zu erkennen. Ein Rechner an Bord des FTF übernimmt die Steuerung des Antriebsrades entlang dieser Leitlinie. Durch Referenzpunkte, z.B. Magnete oder Metallstücke entlang der Fahrspur, können Fehler in der Positionsbestimmung genullt werden und eine Genauigkeit von bis zu zwei Millimetern erreicht werden. [1]

B. Optische Leitlinie

Anstatt der induktiven Sensoren kommen bei dieser Art der Navigation wahlweise Kameras, Farb- oder Lichtsensoren zum Einsatz. Diese sollen eine auf dem Fußboden aufgetragene, farbige Leitlinie erkennen. Die Regelung funktioniert danach analog. Referenzpunkte können zum Beispiel durch quer zur Fahrbahn aufgetragene Farbmarkierungen gekennzeichnet werden. Hierbei können ebenso Genauigkeiten von zwei Millimetern erreicht werden. [1]

Auch im Projektseminar Lego-Mindstorms wurde sich dieser Technik schon bedient, dazu ein Verweis auf ein Projekt aus dem letzten Jahr [2].





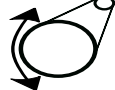
Symbole:   
 Fahrentrieb Stützrolle Lenkantrieb

Abbildung 1. Eine Möglichkeit des Fahrwerksaufbaus eines fahrerlosen Transportfahrzeugs [1]

III. HAUPTTEIL

Wie eingangs erwähnt sind die aktuellen Möglichkeiten der Navigation beschränkt auf feste Fahrspuren. Im Folgenden wird erklärt, wie der Spotfinder aufgebaut und programmiert wurde um eine flexible Routensuche zu ermöglichen. Der Aufbau wurde mit Hilfe von Lego-Mindstorms umgesetzt, der dazugehörige NXT-Brick wurde per Matlab und einer von der RWTH Aachen entwickelten Toolbox programmiert.

A. Konstruktion

Im Fokus standen eine möglichst kompakte Bauweise, die aber genug Stabilität beim Rotieren bietet. Auch auf eine ordentliche Führung der Kabel zu den Sensoren und Motoren wurde geachtet. Der Roboter steht auf drei Rädern, wobei zwei von ihnen separat mit Motoren angetrieben werden können um eine Rotation um die eigene, vertikale Achse zu ermöglichen. Das dritte, kleine Rad am hinteren Ende dient lediglich zum Abstützen und ist frei rotierbar gelagert, so dass es alle Bewegungen des Roboters zulässt. Ein weiterer Motor ermöglicht es einen Arm auszufahren, an dem sich ein Farbsensor befindet der den Boden scannt. Der Sensorarm wird durch Schienen horizontal und durch einen Spanmechanismus vertikal geführt. Ein zweiter Farbsensor befindet sich mittig zwischen den angetriebenen Rädern im Drehpunkt des Roboters.

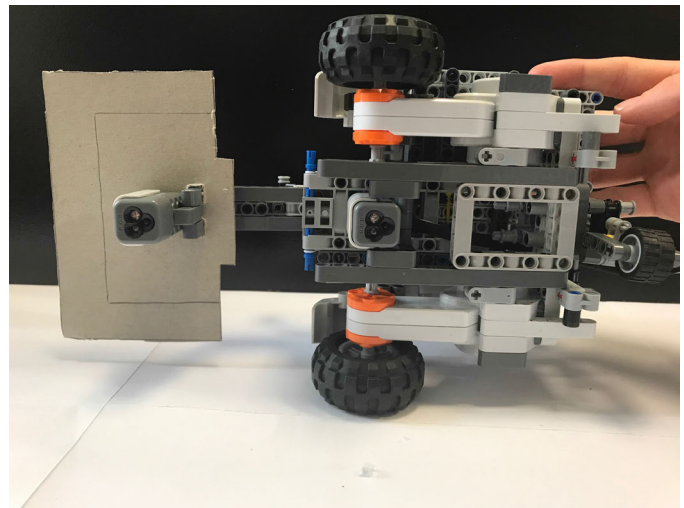


Abbildung 2. Unterseite des Spotfinders mit beiden Farbsensoren, den Antriebsrädern und dem Stützrad

Beide Farbsensoren sind in Abbildung 2 zu sehen. Außerdem wurden ein Ultraschall- und ein Lichtsensor verbaut. Diese dienen der farbabhängigen Ausführung von Aktionen (Tabelle I). Abbildung 4 zeigt den fertigen Aufbau des Spotfinders.

B. Programmablauf

Zu Beginn jedes Programmstarts befindet sich der Sensorarm im eingefahrenen Zustand. Der Roboter dreht sich um seine eigene Achse und stoppt sobald eine farbige Markierung auf dem Boden gefunden wurde oder eine volle Umdrehung stattgefunden hat. Da der zuletzt angefahrne Punkt zwangsläufig beim nächsten Scan gefunden wird muss immer überprüft werden ob die gefundene Markierung bereits angefahren wurde. Hierzu werden die Motordrehwinkel ausgewertet um somit die relative Drehung des Roboters festzustellen. Wurde eine Markierung bei 180° Roboterdrehung gefunden, handelt es sich um den zuletzt angefahrenen Punkt, dieser wird ignoriert und der Scan läuft weiter.

Wurde keine Markierung gefunden wird überprüft in welcher Position sich der Sensorarm befindet. Bis zu zweimal kann der Arm ausgefahren werden um den Suchradius zu vergrößern. Ist eine gültige Markierung gefunden, werden die Motordrehwinkel in eine Matrix an Stelle $i \times 1$ (i ist der Zähler der aktuell gefundenen Spots) geschrieben. Die Drehwinkel werden auf null zurückgesetzt und der Arm in Ausgangsposition gefahren. Jetzt fährt der Roboter geradeaus bis der Sensor auf der Drehachse über dem Farbpunkt steht. Der zurück gelegte Weg wird in die Matrix an Stelle $i \times 2$ geschrieben. So steht fortlaufend in den Zeilen der Matrix die Drehwinkel und Strecken aller gefunden Markierungen. Folgende Matrix ist beim Dreh des Demovideos (siehe Anhang) für die Abschlusspräsentation im Rahmen des Projektseminars entstanden.

40	258
151	255
63	333
146	239
158	255

Anhand dieser Matrix fährt der Roboter später zum Ausgangspunkt zurück. Auch jetzt werden die Zähler der Motoren zurückgesetzt.

Der Farbsensor zwischen den Antriebsrädern kann jetzt die Farbe scannen und die zugehörige Aktion wird ausgeführt. Alle Aktionen in Abhängigkeit der gescannten Farbe können Tabelle I entnommen werden. Die entsprechenden Texte werden dem Nutzer im Command Window von Matlab ausgegeben. Abbildung 5 im Anhang zeigt ein Beispiel für die Ausgabe. Dieser Ablauf kann beliebig oft wiederholt werden, bis einer der zwei folgenden Abbruchbedingungen eintritt:

1. Die letzte, schwarze Markierung wurde erreicht.
2. Der Suchradius wurde bereits zweimal erweitert und auch beim dritten Scan konnte kein Farbpunkt gefunden werden.

Jetzt wird ein neues Programm gestartet, welches auf die Matrix zurückgreift und den Roboter autonom, ohne erneut die Markierungen scannen zu müssen zum Ausgangspunkt zurückfahren lässt. Beide Farbsensoren werden abgestellt. Zuerst dreht sich der Roboter um 180°. Beginnend mit der letzten Zeile der Matrix wird nun zuerst der in Spalte zwei gespeicherte Weg gefahren und danach dreht sich der Roboter um den in der ersten Spalte gespeicherten Winkel. Diese Schleife wird fortlaufend bis zur ersten Zeile wiederholt. Zum Schluss dreht sich der Roboter erneut um 180° und steht dann in seiner Ausgangslage.

C. Erprobung

Die zum Ziel gesetzte Stabilität des Spotfinders konnte nach wenigen Programmdurchläufen erzielt werden. Es zeigte sich, dass der ausfahrbare Arm in den Rotationen zu viel Spiel aufwies und so der optimale Abstand zum Untergrund nicht eingehalten werden konnte. Seitliche Schienen zur horizontalen Führung und ein Spannmechanismus, der den Arm permanent vertikal fixiert, lösten dieses Problem.

In den Erprobungen zeigte sich schnell, dass die Farbsensoren hohe Fehleranfälligkeit aufweisen. Schatten, Verschmutzungen oder Knicke im Untergrund wurden als Farbmarkierungen erkannt. Außerdem wurden dunkle Farben wie Grün oder Blau oft als Schwarz erkannt. So kam es immer wieder zu vorzeitigen Programmabbrüchen. Eine Ursache dafür wurde in dem einfallendem und vom Untergrund reflektierendem Licht gesehen. Die Farbsensoren arbeiten in dem sie weißes Licht aussenden und das verbleibende reflektierte Licht auswerten. Tritt zu viel Licht aus der Umgebung in den Prozess, kommt es zu fehlerhaften Scans. Um möglichst viele Fehlerquellen auszuschließen wurde der Versuchsaufbau auf ein weißes Plakat aufgetragen und eine Umgebung gewählt, in der Licht kaum direkt einfällt um die starke Schattenbildung zu minimieren. Des Weiteren wurden die Farbmarkierungen anstatt aus reflektierendem Papier nun aus mattem Karton ausgeschnitten. Der im Drehpunkt des Spotfinders liegende Sensor scannt aufgrund der Vielzahl an umliegenden Teilen permanent im Schatten. Um gleiche Bedingungen unter den Sensoren zu schaffen wurde eine Pappe am Arm, über den dort befestigten Sensor angebracht. Diese Pappe ist in Abbildung 2 und 4 zu sehen.

All diese Maßnahmen führten dazu, dass Programmabläufe vollständig durchgeführt werden konnten. Dabei wurden

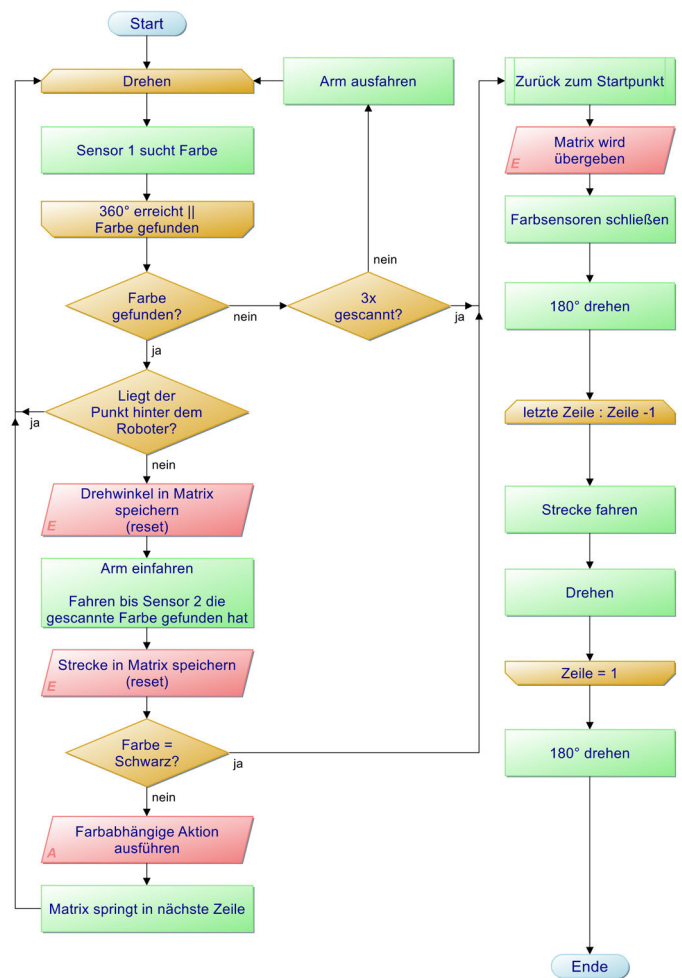


Abbildung 3. Programmablaufplan

die farbabhängigen Aktionen zuverlässig ausgeführt und die Rückkehr zum Ausgangspunkt erfolgte präzise. Durch eine Abfrage der Motorwinkel beim Drehen des Roboters ist es gelungen, den zuletzt angefahrenen Farbpunkt zu ignorieren.

Tabelle I
FARBABHÄNGIGE AKTIONEN

Farbe	Sensor	Aktion
Rot	Ultraschall	Entfernungsmessung in Fahrtrichtung Der NXT-Brick spielt einen Ton ab Textausgabe im Command Window von Matlab
Grün		
Blau		
Gelb	Licht	Helligkeitsmessung Der Spotfinder fährt zum Ausgangspunkt
Schwarz		

IV. ERGEBNISDISKUSSION

Die zu Beginn formulierten Ziele wurden erfolgreich umgesetzt. Der Spotfinder kann selbstständig Farbmarkierungen erkennen und diese anfahren. Die farbabhängigen Aktionen werden zuverlässig ausgeführt und beim Eintreten eines Abbruchkriteriums fährt der Roboter autonom zurück. In den Erprobungen zeigte sich jedoch, dass dafür ein speziell modularer Versuchsaufbau nötig ist. Ein erfolgreicher Einsatz des Spotfinders ist also noch stark von äußeren Bedingungen

abhängig. Deutlich wurde aber, dass dafür nicht das Programm, sondern die eingesetzten Farbsensoren die Fehlerquellen verantwortlich sind. Durch präzisere Modelle könnten Erfolge auch in weniger anspruchsvollen Umgebungen erzielt werden.

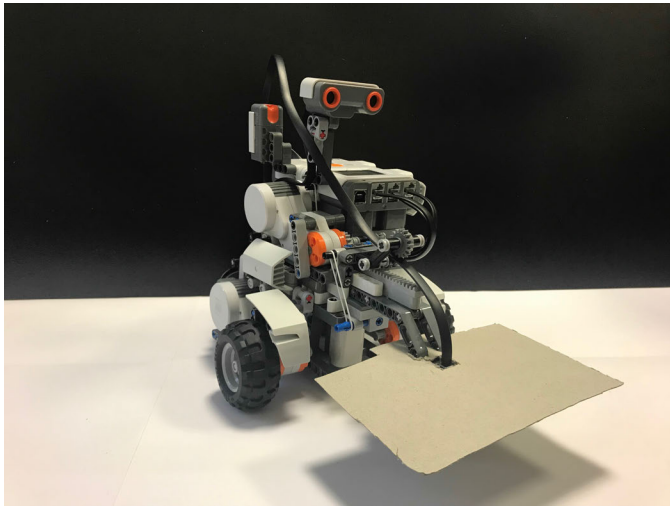


Abbildung 4. Ergebnis der zweiwöchigen Entwicklung: Der Spotfinder

V. ZUSAMMENFASSUNG UND FAZIT

Im Rahmen des Projektseminars ist es gelungen eine neue Art der Navigation und autonomen Routensuche mittels Lego-Mindstorms umzusetzen. Es ist ein Roboter entwickelt worden, der autonom eine Route aus farbigen Punkten abfährt, in Abhängigkeit der gescannten Farbe die spezifische Aktion ausführt und beim Eintreten einer Abbruchbedingung zuverlässig zum Ausgangspunkt zurückkehrt.

Navigationsarten wie die Spurführungen nach induktiver oder optischer Leitlinie wird der Spotfinder aber nicht ablösen können. Die hohe Fehlerrate der Farbsensoren trotz konstruktiver Maßnahmen wäre in der Praxis wohl ein Ausschlusskriterium im Vergleich zur sehr genauen Spurführung. Die laborähnliche Umgebung, die für die reibungslosen Probendurchläufe geschaffen werden musste, ist in der Praxis nicht umsetzbar. Außerdem zeigte sich, dass durch die Vielzahl an Scans der Programmablauf sehr viel Zeit kosten kann, Zeit, die in der Industrie nicht vorhanden sein wird. Dennoch hat der Spotfinder durchaus prägnante Vorteile. Es ist ohne Probleme möglich, die Route nach oder sogar während eines Programmablaufs zu verändern. Dazu müssen nur die farbigen Markierungen verschoben werden. Des Weiteren ist die kompakte Bauweise zu nennen. Trotz drei Motoren und vier Sensoren ist der Spotfinder sehr klein und kann problemlos portabel eingesetzt werden. Sinnvoll wäre es sicherlich den Ultraschall- und Lichtsensor zu ersetzen. Barcodescanner, GPS-Sender oder Kameras wären Beispiele für anwendungsorientierte Lösungen. Lego allein bietet hier nicht genug Auswahl, Matlab hingegen lässt es zu zum Beispiel Webcams einzubinden.

Es muss das Ziel sein den Spotfinder weiterzuentwickeln. Fraglich ist ob drei Suchradien ausreichen, mit mehr Scans könnte ausnahmslos die gesamte umliegende Fläche nach Markierungen überprüft werden. Auch der maximale Suchradius

kann je nach Anwendung durch einen Arm anderer Länge angepasst werden. Eine Idee, die aus Zeitmangel während des Projektseminars nicht umgesetzt wurde, ist das Erstellen einer grafischen Benutzeroberfläche (Abk. GUI von englisch graphical user interface). Mittels einer GUI könnte der Nutzer des Spotfinders vor jedem Programmablauf per drop-down Menü den Farben Aktionen zuordnen. Eine im Programmcode festgelegte Zuordnung würde so umgangen werden. Des Weiteren könnte in der GUI ein Textfeld eingebunden werden, in dem die Ausgabe stattfindet, dies geschieht bis jetzt noch im Command Window von Matlab.

ANHANG

Im Rahmen des Projektseminars wurde dieses Video gedreht, es zeigt einen kompletten Ablauf des Spotfinders: <https://www.youtube.com/watch?v=JccdCNDJkYo>. [3]

```
Helligkeit:
          468

Entfernung nach vorne in cm:
          89

Blue like the ocean
Black Spot found, FINISH!!!
Ich fahre nach Hause, das ist meine Route:
          40      258
          151     255
           63     333
          146     239
          158     255

Ich bin zurueck!
```

Abbildung 5. Text des Command Window aus Matlab nach dem Programmablauf aus dem Demovideo

LITERATURVERZEICHNIS

- [1] WIKIPEDIA, THE FREE ENCYCLOPEDIA: *Fahrerlose Transportfahrzeuge*. https://de.wikipedia.org/wiki/Fahrerloses_Transportfahrzeug. Version: März 2019
- [2] SATTLER, BENNET: *Bau einer autonomen Raumraupe*. <http://journals.ub.uni-magdeburg.de/ubjournals/index.php/LEGO/article/view/552/527>. Version: Februar 2018
- [3] MAGDOWSKI, MATHIAS: *Spotfinder aus dem Lego-Praktikum 2019 an der Otto-von-Guericke-Universität Magdeburg*. <https://www.youtube.com/watch?v=JccdCNDJkYo>. Version: Februar 2019