

# Der Farbsortierroboter

Lara Müller, Elektro- und Informationstechnik  
Otto-von-Guericke-Universität Magdeburg

**Abstract** - Auch in diesem Jahr 2019 wurde das Lego Mindstorms Praktikum von der Otto von Guericke Universität durchgeführt. Seit 2013 konstruieren und programmieren Studenten verschiedene Roboter aus LEGO, die vor einer Jury aus Studenten vorgestellt werden. Wie auch der Farbsortierroboter, mit dem sich in diesem Paper auseinandergesetzt wird. Dabei wurden die Schwerpunkte auf die Planung der Idee und die Umsetzung des Konzeptes gelegt. Der Farbsortierroboter sollte in der Lage sein LEGO-Steine nach ihrer Farbe zu sortieren und in die entsprechenden Behälter zu sortieren. Bei der Umsetzung des Projektes traten einige Probleme beim Programmieren und beim Konstruieren auf. Dennoch ist es den Entwicklern gelungen, am Ende des zwei wöchigen Projektseminars, einen autonom farbsortierenden Roboter zu präsentieren.

Farbsortierroboter, LEGO, NXT, Otto-von-Guericke Universität

## I. EINLEITUNG

Seit Beginn der Industrialisierung ersetzen Maschinen nach und nach die Arbeit der Menschen. In den folgenden Jahren sind Maschinen essenzieller geworden und nicht mehr aus der Wirtschaft und dem privaten Haushalt weg zu denken. Heutzutage ist die Automatisierung von Abläufen prägnanter denn je, denn sie arbeiten größtenteils genauer und schneller und sparen Zeit, die der Mensch in andere Tätigkeiten stecken kann. Besonders bei älteren Menschen sind Maschinen hilfreich, da sie gesundheitlich nicht mehr in der Lage sind jede Arbeit so auszuführen wie ein gesunder und junger Mensch. Auch in der Unterhaltungsbranche sind Maschinen nicht mehr wegzudenken, zum Beispiel Smartphone oder Computer. Sogar Aktivitäten ohne Maschinen werden mit moderner Technik verbessert, wie im Fußball der Videoassistent oder die Kameraführung beim Golf. Sie vermitteln Nähe, wo eigentlich größere globale Distanzen auftreten und verbinden Kontinente und Menschen miteinander. Der wichtigste Aspekt der modernen Maschinen ist die Verringerung der Arbeit des Menschen, worauf sich der Farbsortierroboter bezieht. Dieser Roboter soll die aufwendige Arbeit des Aufräumens/sortieren übernehmen, womit Zeit gespart wird, die in andere Beschäftigungen gesteckt werden kann. Er ist vielfältig einsetzbar und kann verschiedene Prozesse im Alltag übernehmen, wie das Sortieren der Wäsche oder

Kinderspielzeugen. So kann schnell und einfach Chaos minimiert werden und Wichtiges vom Unwichtigen getrennt werden. Der Mensch muss nichts weiter tun, als die zu sortierenden Gegenstände in die Maschine zu packen.

## II. VORBETRACHTUNGEN

Damit der Farbsortierroboter etwas sortieren kann, benötigt er einen Farbsensor dessen Funktionsweise erst mal genauer betrachtet werden muss und in welcher Reihenfolge der Prozess ablaufen soll.

### A. Farbsortierroboter Prinzip

Die Idee dahinter ist ein Roboter der Gegenstände, wie Lego-Steine, nach ihrer Farbe sortiert. Zunächst soll ein Farbsensor die Farben der Gegenstände erkennen. Nach Erkennen der Farbe wird der Stein in einen Behälter transportiert, der seiner Farbe entspricht. Der Vorgang wird so oft wiederholt, bis alle Steine einsortiert sind oder der Benutzer das Programm beendet. Das Prinzip ist recht simple „Ohne großen Aufwand Ordnung zu schaffen“. Dies ist heutzutage auch schon in großen Firmen zu finden. Ein Beispiel ist bei der Tomatenernte zu finden. Bei dieser werden die geernteten Tomaten über ein Fließband laufen und nach Verfärbungen, verfaulten Stellen, Sonnenbrand, gebrochene, zu kleine oder zu große Tomaten und alle Arten von Fremdmaterial, wie Tiere, Baumwollstiele, Metall, Kunststoffe, Steine, Glas, Holz, usw. getrennt. Das Verfahren nachdem die Tomaten sortiert werden, achtet dabei nicht nur auf die Farbe, sondern auch auf andere Faktoren die beim Lego Farbsortierroboter nicht umsetzbar wären.

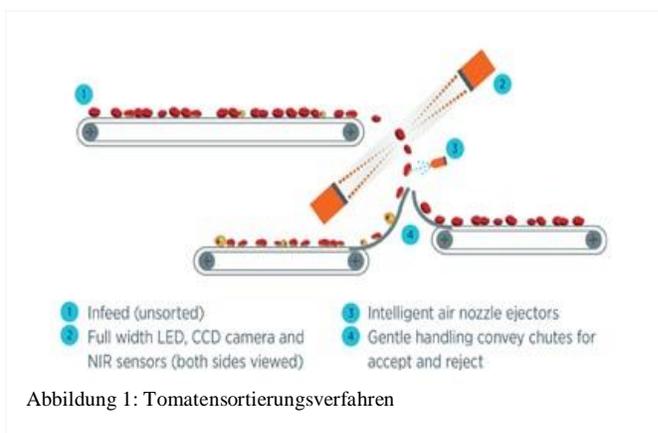


Abbildung 1: Tomatensortierverfahren

### B. Farbsensor

Zum Erkennen der Farben kommt der RGB-Sensor (Farbsensor) von Lego zum Einsatz. Der Sensor strahlt das zu scannende Objekt mit den Farben rot, blau und grün an. Um die Farbe genau zuzuordnen erfasst ein Sensor die reflektierten Farben. Wird mehr grünes Licht zurück geworfen, wird der Gegenstand als grün anerkannt. Das gleiche Prinzip wird ebenfalls bei rot und blau angewendet. Der Farbsensor kann auch noch gelb (rot und grün gleichstark reflektiert, blau keine Reflektion), weiß (alle 3 Farben gleichstark reflektiert) und schwarz (keine Farbe wird reflektiert) erkennen. Weitere Farben, wie Orange, werden bei den 6 genannten Farben (rot, grün, blau, gelb, schwarz, weiß) untergeordnet. Mit dem Wissen das es 6 zu sortierende Farben gibt braucht der Behälter 6 verschiedene Fächer für je eine Farbe.

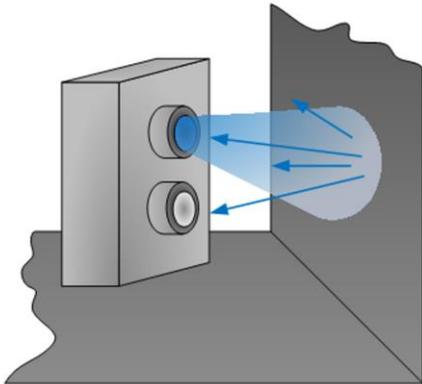


Abbildung 2: Funktionsweise Farbsensor

### C. Starten und pausieren des Programms

Das System soll auf Knopfdruck starten und stoppen. Im Falle eines Falls soll ein Notaus-Schalter existieren, um schwerere Probleme zu verhindern. Zur Auswahl der Umsetzung stand ein Start/Stop Knopf in der GUI (Graphical User Interface) oder ein Schalter aus einem Tastsensor. Diese Funktion ist hilfreich um zu verhindern, dass sich Gegenstände verkeilen und das System unnötig Strom verbraucht.

## III. HAUPTTEIL

### A. Farberkennung

Begonnen wurde mit der Einrichtung des Farbsensors in MATLAB. Dafür muss man den Sensor ansprechen und einen Modus wählen. Von den mehreren zur Auswahl stehenden Modi des Farbsensors wurde der Modus „FULL“ verwendet. Die Farben werden größtenteils richtig erkannt und in einem Textfeld ausgegeben. Um die Genauigkeit des Sensors zu verbessern, bekam der Sensor einen Kasten, der äußere Störfaktoren verhindern und genauere Ergebnisse liefern soll.

### B. Einordnung der Steine

Die erste Idee zum Einordnen der Steine war eine Rampe, die mittels Winkel, die Steine in die richtigen Fächer einsortiert. Dabei stellte sich heraus, dass damit zu wenige Fächer (maximal 3) angesteuert werden würden. Eine andere Möglichkeit war ein Wagen am Ende des Fließbandes, der vor und zurück fährt. Damit konnten beliebig viele Fächer angesteuert und die benötigte Anzahl konnte gebaut werden. Die zweite Möglichkeit wurde umgesetzt, weil eine hohe Anzahl von Fächern für das Projekt benötigt wird und nur der Wagen diese bietet.

### C. Mechanische Umsetzung

Um das Projekt umzusetzen bekam jede Gruppe einen Lego Mindstorm-Baukasten. Zusätzlich gab es einen NXT, zwei Motoren, einen Farbsensor und einen Tastsensor. Für die Grundlage wurde ein Gummiband über Ketten gespannt, um ein Fließband zu erzeugen, womit die Steine transportiert werden konnten. Dieses Fließband wurde dann statisch so fixiert, dass der Farbsensor nun auch angebaut werden kann. Der Farbsensor wurde am hinteren Ende des Fließbandes so angebracht, dass er von oben auf das Fließband blickt und so die Steine einzeln scannen kann. Dann bekam der Sensor seinen Kasten, welcher noch stabiler gebaut wurde, um ein umkippen zu verhindern. Nachdem Sensor und Fließband standen, wurde mit Zahnrädern ein Motor am Fließband befestigt und die Geschwindigkeit eingestellt. Anschließend wurde noch ein weiteres Fließband mit dem anderen verbunden. Der Unterschied zum ersten war, dass die Übersetzung langsamer war. Durch das langsamere erste Laufband fallen die Lego-Steine vereinzelt auf das zweite und der Sensor bekommt immer einen Stein unter sein Blickfeld. Teil A des Farbroboters war fertig, nun folgt Teil B mit dem Wagen und den Fächern. Das Model des Wagens zum Auffangen der Steine ähnelt dem Güterwaggon eines Zuges. Es wurden lange Streben benutzt, um das Grundgerüst zu bauen. Daran werden anschließend jeweils 4 Räder befestigt. Zuletzt erhält der Wagen einen Motor, der die Hinterräder antreibt. Damit der Wagen immer im richtigen Abstand zum Fließband steht, wurde ein Platzhalter angebaut, wodurch die Ausrichtung immer stimmte. Kleine kosmetische Bauteile wurden noch hinzugefügt, wie eine Rampe gleich zum Anfang, um leichter die Steine auf das Fließband zu bekommen und noch eine weitere Rampe am Ende des schnelleren Fließbandes, beim Farbsensor, damit die Steine leichter und zielgenauer in die Fächer des Wagens fallen. Soweit ist der Roboter fertig und es fehlt nur noch der Programmiererteil.

### D. Funktionsweise des Programms

Die Einführung von der Programmiersprache MATLAB wurden in der ersten Wochen des zwei wöchigen Praktikums durchgeführt. In der zweiten Woche lag der Fokus auf den einzelnen Projekten der Gruppe. Der erste Schritt lag in der Erstellung eines Programmablaufplanes [1], um eine Übersicht zu bekommen, was die grundlegenden Schritte im Programm sind. Begonnen wird mit dem Tastsensor. Ist der Sensor nicht gedrückt (also 0), so fragt das Programm immer wieder den Tastsensor ab, bis er gedrückt wird (also 1 anzeigt). Sobald der

Tastsensor gedrückt wird, startet das Fließband und dreht solange, bis ein Stein unter dem Farbsensor liegt. Dann scannt der Farbsensor den Stein und stoppt den Motor der Fließbänder. Über eine if-Bedingung geht das Programm jede Farbe durch. Wenn er die richtige gefunden hat, folgt folgendes. Sobald das Fließband steht, bewegt sich der Wagen um einen gewissen Rotationswinkel und der Motor des Wagens stoppt. Wenn der Wagen steht, startet der Motor des Fließbandes wieder und der Stein rutscht über die Rampe in ein Fach der entsprechenden Farbe. Der Motor des Fließbandes läuft weiter, bis wieder ein Stein unter dem Farbsensor steht. Währenddessen bewegt sich der Wagen zu seiner Ausgangsposition zurück. Der Prozess wird solange durchlaufen, bis der Tastsensor nicht mehr gedrückt wird oder der NXT von MATLAB getrennt wird. Der Grund für den ständigen Durchlauf ist eine while-Schleife in der der Quellcode steht. Zu Beginn der Schleife wird der Tastsensor mit einer If-Bedingung abgefragt wie bei den Farben.

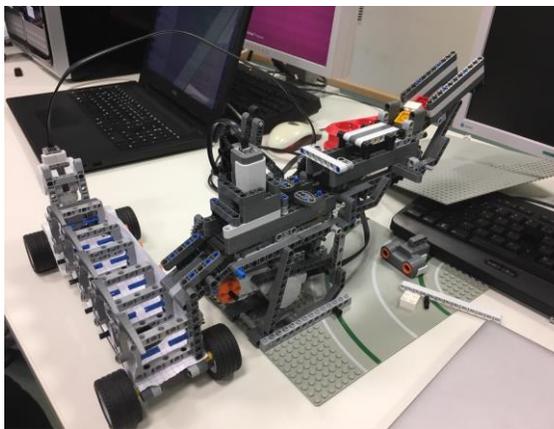


Abbildung 3: Farbsortierroboter

#### IV. ERGEBNISDISKUSSION

Am Ende der zwei Wochen des Projektseminars wurde ein funktionierender Farbsortierroboter gebaut der Lego-Steine nach ihrer Farbe sortiert. Es wurden einige Tests durchgeführt wie Dauer des Transportes eines Steins von Start bis in sein Fach. Im Schnitt braucht ein Stein 13 Sekunden und daraus lässt sich schließen das 50 Steine um die 11 Minuten brauchen, sofern alles ohne Probleme abläuft. Bei einem weiteren Test

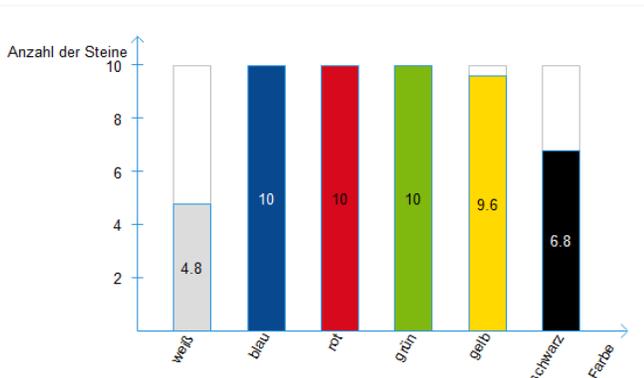


Abbildung 4: Fehlerquote der verschiedenen Farben

wurde die Fehlerquote ermittelt und in einem selbsterstellten Diagramm dargestellt.

Bei dem Test wurden von jeder Farbe 10 Steine verwendet, welche vom Roboter einsortiert wurden. Dabei kam heraus, dass grün, rot, blau zu 100% korrekt zugeordnet wurden und gelb nur mit 96%. Jedoch gab es bei weiß und schwarz die größten Abweichungen. Dies hat folgenden Grund: um eine Farbe richtig zu erkennen, werden die Steine beleuchtet. Wenn der weiße Stein an der Reihe ist wirft dieser viel von dem blauen und grünen Licht zurück. Dabei bleibt das rote Licht auf der Strecke. So wird weiß, bei grün oder blau meist eingeordnet. Das gleiche gilt für schwarz. Wenn die Oberfläche der Steine nicht glatt, sondern rau sei, so würde der Sensor genauer arbeiten können, wodurch weiß und schwarz genauer einsortiert werden könnten.

#### A. Probleme

Bei der Umsetzung des Projektes kam es zu Problemen. Ein Problem entstand, als der Code geschrieben wurde, denn beim ersten Versuch gab der NXT die ganze Zeit einen Ton von sich. Dies war ein Zeichen dafür, dass zu viele Befehle auf einmal an den NXT gesendet wurden und er dadurch überlastet war. Das Problem wurde durch Befehle wie „WaitFor()“ unterbunden und das Programm lief wie geschnitten Brot. Ein zweites Problem tauchte beim Start und Stopp Knopf in der GUI auf. Durch die while-Schleife wird immer wieder das Programm durchlaufen. Wenn der Stopp-Knopf betätigt wird, so stoppt das System nur für den Moment, indem der Knopf gedrückt wurde. Um das zu lösen, wurde ein Tastsensor als Schalter eingebaut, der noch in die while-Schleife kommt, wodurch das System vor der Farbabfrage den Schalter abfragt. So kann das System mit dem Schalter gestartet und gestoppt werden. Das war es noch nicht mit den Problemen, ein letztes gibt es noch. Beim ungünstigen Hinlegen der Steine auf das Fließband kann es hin oder wieder passieren, dass sich die Steine verkeilen und den Roboter auseinander drücken, oder zu Steingeschossen werden. Dieses Problem entstand allein durch die Art des Reinlegens der Steine. Die spartanische Lösung war es die Steine einzeln reinzulegen. Es gibt auch die Möglichkeit, das Problem zu nutzen und jemanden mit den Steingeschossen abzuschießen. Beim Bau des Roboters wurde festgestellt, dass ein Lego-Baukasten allein nicht ausreicht und mindestens ein zweiter benötigt wurde.

#### V. ZUSAMMENFASSUNG UND FAZIT

Das Projekt hat in dem Umfang mir viel Freude bereitet und dabei noch eine neue Programmiersprache gelehrt. Das Seminar hat die Zusammenarbeit zwischen den Kommilitonen gefördert und wir halfen uns gegenseitig bei Problemen. Die Arbeitsteilung im Team funktionierte einwandfrei.

Wenn das Projektseminar länger als zwei Wochen ginge, könnte man eine komplexere Maschine bauen, auch wenn es manchmal die Menge an Legoteilen nicht zulässt. Den Farbsortiere könnte man in größere Dimensionen übertragen und ihn aus richtigen Bestandteilen der Industrie bauen umso auch andere Gegenstände zu scannen und zu sortieren. Lego Mindstorms ist eine gute Möglichkeit um Kinder und

Jugendliche das Programmieren beizubringen und sie auf einen Weg bringen, der im späteren Leben noch sehr nützlich wird.

## ANHANG

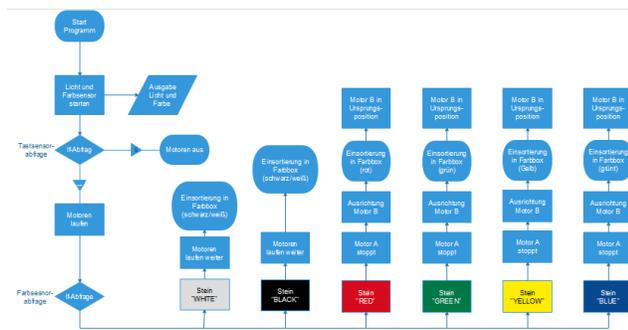


Abbildung 5: Programmablaufplan

### Literaturverzeichnis

Informationen aus dem Internet:

- <https://www.tomra.com/de-de/sorting/food/your-produce/fruit/tomatoes>

Abbildungen :

1. <https://www.tomra.com/-/media/images/sorting-solutions/food/working-principles/iris-wp.ashx?&h=210&hash=65C5F92B8C4D781AC8C79560477CB15CB527896D>
2. Vortrag Sensor von OVGU mit Snipping Tool ausgeschnitten
3. Eigene Aufnahme
4. Eigene Aufnahme
5. Eigene Aufnahme