

Robotik trifft Kunst: Der Lightpainting-Roboter 2.0

Sunny Lou Seidler, Elektromobilität
Otto-von-Guericke-Universität Magdeburg

Zusammenfassung— Im Rahmen des LEGO-Mindstorms-Praktikums 2023 wurde der letztjährige Lightpainting-Roboter „Sally can Draw“ weiterentwickelt. Dieser ist in der Lage, mittels einer Lichtquelle zweidimensionale Formen und Zeichnungen darzustellen. Die dazu verwendeten Konzeptideen sowie deren technische Umsetzungen, Verbesserungen und Probleme werden im Folgenden chronologisch dargestellt.

Schlagwörter— LEGO-Mindstorms, Lightpainting, Roboter, Kunst, Plotter, MATLAB

I. EINLEITUNG

LIGHTPAINTING [1] ist eine Kunstform, bei der Fotografien in der Nacht oder in abgedunkelten Räumen durch gezielten Einsatz von Licht und Langzeitbelichtung entstehen. Die Künstler verwenden dabei Lichtquellen wie Taschenlampen, LEDs oder Leuchtstäbe, um die Umgebung mit Licht zu bemalen. Da der Fotograf in der Regel viele Versuche benötigt, ein Lightpainting-Bild zu erstellen, wurde bereits 2022 von den Studentinnen Michelle Horn und Anna-Lena Thalhafer, im Rahmen dieses Lego-Projekts ein Prototyp mit dem Namen „Sally can Draw“ [2] entwickelt. Dieser sollte neben einer Verkürzung der Gesamtdauer eines Lightpainting-Bildes, auch eine präzisere Linienführung des Lichts ermöglichen, um mit wenigen Versuchen das ideale Bild zu erstellen. Da der Prototyp noch viel Verbesserungspotential hat und das Endergebnis des Bildes nicht optimal aussieht, soll eine 2. Variante des Roboters entwickelt werden. Dieser soll eine stabilere Konstruktion und ein anderes Konzept in der programmtechnischen Umsetzung haben und neben einfachen Formen auch Freihandzeichnungen umsetzen können.



Abbildung 1. Beispielbild zum Lightpainting [3]

II. VORBETRACHTUNGEN

Bevor diese Änderungen am Roboter näher erklärt werden können, sind einige Vorüberlegungen zum Prototyp und zum bereitgestellten Equipment notwendig, die im Folgenden erläutert werden.

A. Hard- und Software

1) *LEGO-Mindstorms-NXT*: Für die mechanische Umsetzung bildet das LEGO-Mindstorms-NXT-Set die Basis der Hardware. Neben verschiedenen Sensoren und Bauteilen stehen auch Motoren zur Verfügung, die alle mit dem NXT-Stein verbunden werden können. Die Motoren bieten viele Einstellmöglichkeiten. Sie können sich z. B. über den Power-Wert schneller oder langsamer bewegen oder über das Tacho-Limit um eine bestimmte Gradzahl drehen.

2) *MATLAB und RWTH – Toolbox*: Die Programmierung des NXT-Steins erfolgt mit der numerischen Rechen- und Programmiersoftware *MATLAB* und der zur Verfügung gestellten *RWTH-Toolbox* aus Aachen [4], mit der die Motoren und Sensoren über den NXT-Stein angesteuert werden können.

3) *Leuchtmittel*: Für die Realisierung der Lichtstrahlen ist der *Calliope mini* [5] eine gute Alternative zur normalen LED. Er lässt sich einfach in verschiedenen webbasierten Entwicklungsumgebungen programmieren. Die Farben sind manuell einstellbar und es gibt gute Befestigungsmöglichkeiten.

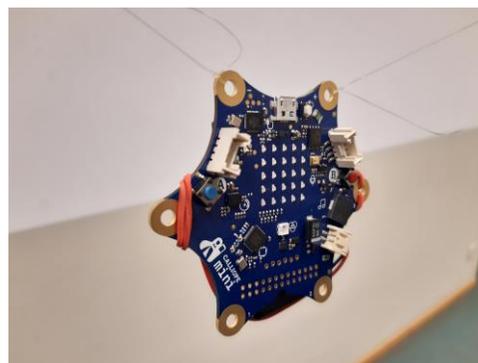


Abbildung 2. Calliope mini

B. Sally can Draw

Der Roboter besteht aus einem Metallgerüst, in dessen Mitte der NXT-Stein befestigt ist. Außen sind zwei Motoren angebracht, die den Calliope mini über eine Schnur durch Auf- und Abrollen in die gewünschte Position bringen. Die softwaretechnische Umsetzung erfolgt in zwei Schritten. Dabei handelt es sich zum einen um die Bildverarbeitung und zum anderen um den Algorithmus, der für die Bewegung der Lichtquelle verwendet werden soll. Zunächst wird das Bild eingelesen und so gefiltert, dass alle Kanten erkannt werden. Anschließend werden die Kanten segmentiert, so dass ein binäres Bild entsteht. Eine „1“ steht für eine Stelle, an der sich eine Kante befindet, ansonsten stehen dort Nullen. Mit der Hilfe des Binärbildes fährt der Roboter jeden Pixel der Kante ab, bis er wieder an der Ausgangsposition angelangt ist. Dabei berechnet das Programm für jede Position des Pixels die Fadellänge. Dazu werden die Motoren um 3° pro Pixel bewegt.

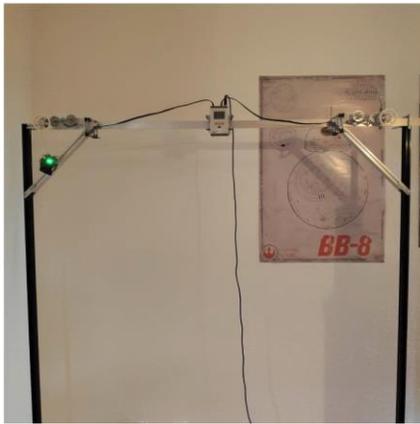


Abbildung 3. Sally can Draw

C. Potentielle Probleme

1) *Konstruktion:* Durch die Anordnung der Motoren innerhalb des Gerüsts kann der Platz nicht voll ausgenutzt werden, da die Verbindung zum NXT-Stein durch die Kabellänge begrenzt ist. Außerdem ist es schwierig, die Leistung der Motoren auf die entsprechende Seillänge einzustellen, da sie über Umwege über die Zahnräder auf die Rolle übertragen werden muss. Ein weiteres potentielles Problem ist die Instabilität der Konstruktion. Da die Klammern zur Befestigung der Motoren außen angebracht sind, kann die Rolle nach unten rutschen und das Endergebnis verschlechtern.

2) *Programmierung:* Durch die Ansteuerung der Motoren über das Tacho-Limit, werden die Formen nur sehr zeitaufwendig dargestellt und runde Objekte sind kaum realisierbar. Besser wäre es die Motoren über Power-Werte zu steuern, um eine kontinuierliche Linie zu erzeugen.

III. TECHNISCHE UMSETZUNG

Die dazu notwendigen Vorüberlegungen und Konzepte sowie die technische Umsetzung zur Verbesserung dieser Probleme werden im Folgenden dargestellt.

A. Mechanische Konstruktion

Das Grundgerüst des Lightpainting-Roboters besteht wie beim Prototyp aus einem Metallrahmen, in dessen Mitte sich oben der NXT-Stein befindet. Die beiden Motoren sind hier jedoch direkt am NXT-Stein befestigt und wickeln den Calliope mini über eine Angelschnurrolle und zwei Umlenkrollen, die außen am Rahmen befestigt sind, in die gewünschte Position. Dadurch wurde zum einen die Platzausnutzung verbessert und zum anderen das Problem der Kabellänge gelöst. Außerdem bietet das Gerüst mehr Stabilität, da die Schienen auch zur Befestigung der Bauteile dienen. In der Konstruktion befinden sich zudem 4 Tastsensoren, mit denen die Motoren per Knopfdruck vorwärts bzw. rückwärts gesteuert werden können, um den Calliope mini vor dem Start des Programms in die Startposition zu bringen.

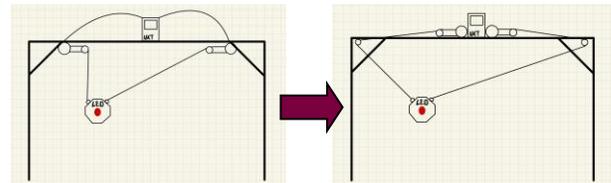


Abbildung 4. Aufbau: 1. Konzept (links), finales Konzept (rechts)

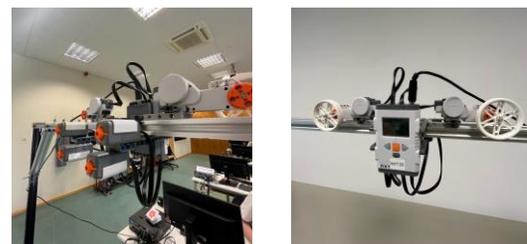


Abbildung 5. Gesamtkonstruktion (oben), manuelle Steuerung (links), NXT-Baustein und zwei Motoren mit Rollen (rechts)

B. Vorgehensweise für die Programmierung

Um eine präzise Linienführung zu realisieren und die Seillängen so anzupassen, dass der Lightpainting-Roboter die Form möglichst genau abfährt, müssen die in der Abbildung 3 dargestellten geometrischen Zusammenhänge angewendet werden. Daraus ergeben sich die Gleichungen (1) und (2), die für jede Koordinate der abzufahrenden Form die Längen der Seile berechnen. Aus diesen Längenänderungen und der zeitlichen

Differenz von Koordinate zu Koordinate ergeben sich die jeweiligen Geschwindigkeiten. Damit man diese an den Motor übertragen kann, muss zunächst eine Motorkalibrierung durchgeführt werden, die die Winkelgeschwindigkeit pro Power berechnet.

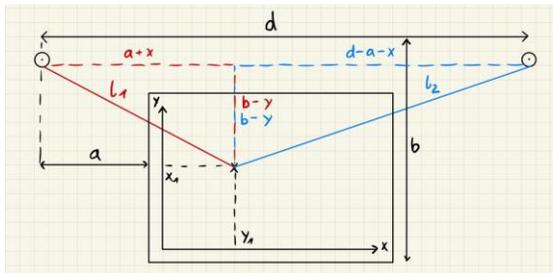


Abbildung 6. Geometrische Zusammenhänge

$$l_1 = \sqrt{(a+x)^2 + (b-y)^2} \quad (1)$$

$$l_2 = \sqrt{(d-a-x)^2 + (b-y)^2} \quad (2)$$

C. Motorkalibrierung

Die Kalibrierung wird mittels einer For-Schleife umgesetzt. Dabei wird bei jedem Schleifendurchlauf der Power-Wert des Motors erhöht und proportional dazu auch das Tacho-Limit. Sobald der Motor nach dem Start das Tacho-Limit erreicht, wird die Zeit gestoppt und ein neuer Schleifendurchlauf beginnt. Die Schleife endet, wenn der Motor einen Power-Wert von 100 % angenommen hat. Aus den aufgezeichneten Werten kann die Winkelgeschwindigkeit pro Power berechnet werden. Da sich der Motor nur bis ca. 80 % linear verhält, wie in Abbildung 4 zu sehen ist, ist dies auch der maximale Wert, der eingestellt werden kann. Andernfalls werden die linearen Abweichungen zu groß.

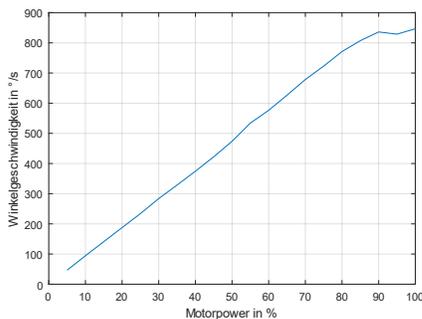


Abbildung 7. Winkelgeschwindigkeit in Abhängigkeit der Motorpower

D. Programmablauf

Der Benutzer hat die Möglichkeit, das Programm so einzustellen, dass die maximale Leistung des Motors abgerufen wird, so dass der Prozess in kürzester Zeit abgeschlossen werden kann. Er kann aber auch eine beliebige Zeit wählen, zu der das Programm beendet sein soll. Außerdem besteht die

Möglichkeit, zwischen verschiedenen Formen und Größen zu wählen oder Freihandzeichnungen [6] in das Programm einzufügen. Nach der Auswahl wird die Form geplottet und für die jeweiligen Koordinaten werden die Längendifferenzen mit den Gleichungen (1) und (2) berechnet. Daraus ergeben sich die Winkeländerungen für die Motoren und mit dem Wert der Winkelgeschwindigkeit pro Power und der zeitlichen Differenz von Koordinate zu Koordinate, die sich aus den beiden Optionen am Anfang ergibt, die Power-Werte für die Motoren.

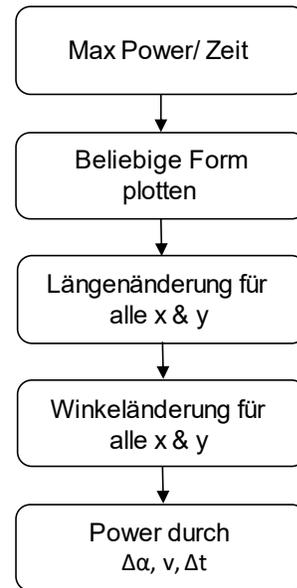


Abbildung 8. Ablaufplan des Algorithmus

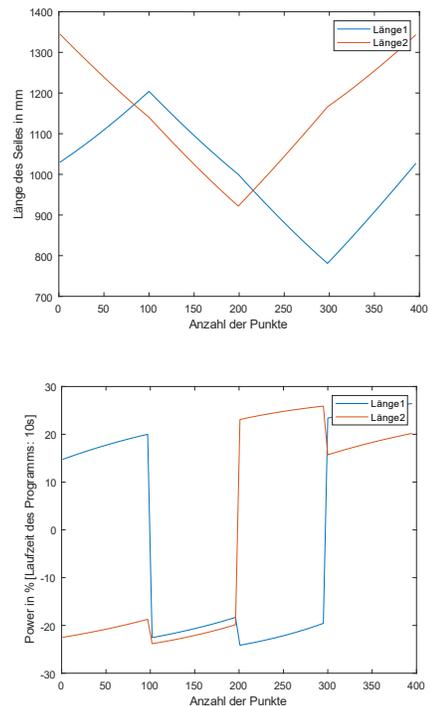


Abbildung 9. Länge der Seile (oben) und Power-Werte der Motoren (unten) beim Quadrat (a = 50 cm, 100 Punkte pro Seite)

IV. ERGEBNISDISKUSSION

Am Ende des Projekts ist ein Lightpainting-Roboter entstanden, bei dem die Funktionen und der technische Aufbau des Prototyps verbessert wurden. Neben der maximalen Ausnutzung der Fläche und der Verbesserung der Stabilität des Gestells werden nun Endergebnisse mit klaren Linienführungen und vollständiger Wiedergabe der Formen erzielt. In der Abbildung 10 ist auch zu sehen, dass der Roboter neben dem Kreis, Stern, Quadrat und dem Dreieck auch Freihandzeichnungen darstellen kann. Das Problem dabei ist jedoch, dass die Ergebnisse meist sehr verwackelt sind, weil die Kamera nur maximal 30 Sekunden mit Langzeitbelichtung aufnehmen kann und der Roboter daher mit hektischen Bewegungen versucht, das Bild in dieser Zeit abzufahren. Neben diesem Problem gab es bei der Entwicklung des Programms noch viele weitere Schwierigkeiten. Zum einen stellten die Berechnungen für die Motorsteuerung eine große Herausforderung dar, aber auch das Plotten der Formen und die unzähligen Bugs während der Programmierphase kosteten viel Mühe und Zeit bei der Bearbeitung.

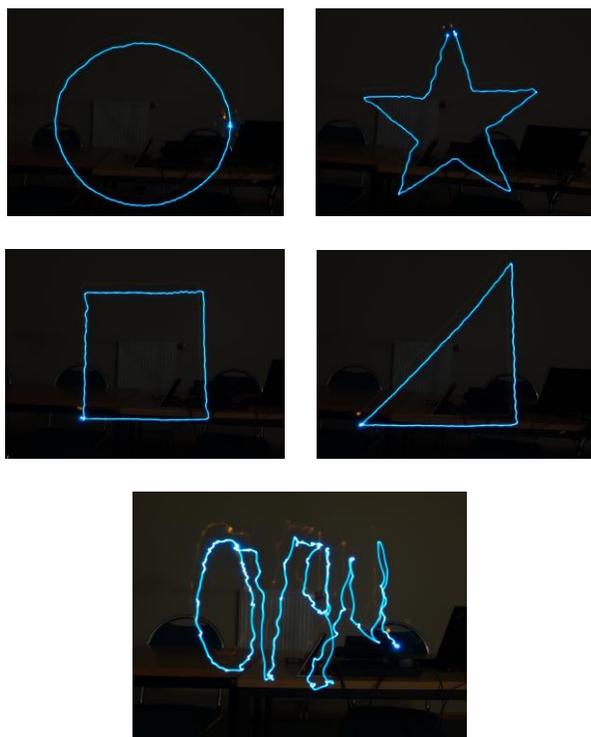


Abbildung 10. Ergebnisbilder: vier vorgefertigte Formen (oben), Freihandzeichnung (unten)

V. ZUSAMMENFASSUNG UND FAZIT

Ziel des Projektes war es, die Idee des Lightpainting-Roboters aus dem Jahr 2022 umzusetzen und weiterzuentwickeln. Dieses Ziel wurde in vollem Umfang erreicht. Sowohl die konstruktiven als auch die softwaretechnischen Verbesserungen werden aufgezeigt. Darüber hinaus werden aufgetretene Probleme und Endergebnisse dargestellt. Für die Zukunft bietet der Roboter jedoch noch Verbesserungspotential. Zum einen wäre es möglich, dass der Roboter während der Bewegung für eine

gewisse Zeit nicht zeichnet, da der Calliope mini über Bluetooth gesteuert werden kann. So könnten alle möglichen Zeichnungen und Schriftarten realisiert werden. Eine weitere Verbesserungsidee, die allerdings schwieriger umzusetzen ist, wäre die Erweiterung von 2- in 3-Dimensionale, indem man die Schiene mit weiteren Motoren beweglich macht.

LITERATURVERZEICHNIS

- [1] WIKIPEDIA: *Light Painting*. https://de.wikipedia.org/wiki/Light_Painting. Version: Dezember 2022
- [2] MAGDOWSKI Mathias SCHALLSCHMIDT Thomas, GERLACH Thomas, PANNICKE Enrico: *LEGO-Praktikum. Entwickeln, programmieren, optimieren: Berichte der Studierenden zum Projektseminar Elektrotechnik/Informationstechnik*. <https://journals.ub.ovgu.de/index.php/LEGO/article/view/2072/2065>. Version: August 2022
- [3] MUNDUS: *Praxistest: Light-Painting*. <https://www.foto-mundus.de/blog/praxistests/praxistest-light-painting-fujifilm-x-t10-fuji-xf10-24mm-f4.0-r-ois>. Version: Januar 2016
- [4] BEHRENS, Alexander: *RWTH - Mindstorms NXT Tool-box*. <https://www.mathworks.com/matlabcentral/fileexchange/18646-rwth-mindstorms-nxt-toolbox>. Version: Oktober 2011
- [5] CALLIOPE GGBH: *Calliope mini, technische Daten*. <https://calliope.cc/calliope-mini/technische-daten>. Version: Februar 2023
- [6] MATHWORKS: *How can I draw with mouse as we do in paint?* <https://de.mathworks.com/matlabcentral/answers/320227-how-can-i-draw-with-mouse-as-we-do-in-paint>. Version: Februar 2023