

SWEEEP

Smart Waste Elimination & Eco Protection

**S.W.E.E.P – intelligenter Müllsammelroboter für
Grünflächen und öffentliche Grünanlagen**

Ein Projekt von Roland Jindra und Jonas Koch

Logodesign: Lisa Moritz

--- Alexander von Humboldt Gymnasium Neuss ---

Mit Unterstützung des Sponsorenpools von Jugend forscht
und des Fördervereins des Alexander von Humboldt Gymnasiums

Kurzfassung unseres Projektes

Ein sehr markantes Problem in unserer modernen Gesellschaft ist Müll. Wir verbrauchen in Deutschland jede Stunde 320.000 Einweg-Becher ^{Quelle 1}. Viele davon landen in unserer Umwelt. Grünanlagen, Parks und Wiesen müssen konstant gesäubert und abgeräumt werden, damit dieser Müll keine Gefahr für die Tier- und Pflanzenwelt darstellt.

Wir haben uns diesem Problem gewidmet. Dazu entwickelten wir ein Konzept für einen Müllsammelroboter, der mithilfe von künstlicher Intelligenz dazu in der Lage sein soll, Müll auf Wiesen zu erkennen, von anderen Stoffen zu unterscheiden und anschließend mit Hilfe verschiedener Mechanismen aufzusammeln. Durch jene Abläufe soll unserer Roboter künftig und in einer weiterentwickelten Form dazu in der Lage sein, völlig autonom auf befahrbaren Grünflächen wie Parks und Wiesen, Müll zu erkennen und selbstständig aufzusammeln.

Zentrale Aspekte unseres Projektes sind hierbei die Entwicklung von:

- Bildverarbeitung durch KI,
- Input der KI in die Steuerung,
- Bau des Fahrgestelles,
- Aufnahmemechanismus für Müll,
- Interne Stromversorgung der eingebauten Technik,

sowie die Verlinkung aller Teilgebiete.

Dabei teilten wir den physischen Bau des Roboters in drei weitere Teilgebiete:

- Müll: Entwicklung und Realisierung eines Aufnahmemechanismus und einem anschließend folgenden internen Transportsystem (das Förderband in Kombination mit den Bürsten)
- Interne Elektronik: Konstruierung und Umsetzung der internen Elektronik und der dazugehörigen Stromversorgung
- Gesamtkorpus und dessen Antrieb

Inhaltsverzeichnis

1. Ideenfindung	4
2. Wie kann so eine Lösung realisiert werden?	4
2.1. Grundlegende Umsetzung	4
2.1.1. Antriebsart	4
2.1.2. Aufnahme, Transport und Lagerung des Mülls	5
2.1.2.1. Erstkontakt und Aufnahme des Mülls	5
2.1.2.2. Kontrolle des Mülls und Einführung in die innere Einheit	5
2.1.2.3. Entwicklung der Seiten- und Oberbüsten zur Unterstützung der Kontrolleinheit	5
2.1.2.4. Interner Transport des Mülls mit Hilfe eines Förderbands	6
2.1.2.5. Lagerung des Mülls innerhalb des Roboters	6
2.2. Programmiertechnisches Segment	7
2.2.1. Erläuterung unserer Vorgehensweise	7
2.2.2. Grundsätzliche Erklärung der Arbeitsweise unserer KI	7
2.2.3. Analyse des Livebildes und Weitergabe von Rohdaten	7
2.2.4. Verarbeitung durch das interne Steuersystem	7
2.2.5. Müllsuche und Ausweichautomatik	8
2.3. Umsetzung, genutzte Komponenten und Ausblick	8
2.3.1. Akkumulator und Spannungswandler	8
2.3.2. Hauptcomputer Jetson Orin Nano Super und Steuereinheit	8
2.3.3. Warum diese Komponenten?	9
2.3.3.1. Mini Computer	9
2.3.3.2. Antrieb	9
2.3.3.3. Sonstige Motoren und dazugehörige Komponenten	9
2.3.3.4. Kameras, andere Sensorik und mögliche nächste Schritte	9
2.4. Training der künstlichen Intelligenz	10
3. Problematiken	15
3.1. Finanzierung	15
3.2. Umsetzung	15
3.3. Zeit und Fristen	16
3.4. Ungeeignete Komponenten	16
Fazit des Projekts "S.W.E.E.P."	17
Quellenverzeichnis	18
KI in der Arbeit	18

1. Ideenfindung

Das Projekt SWEEP fasste in der Jugend Forscht AG am Alexander-von-Humboldt Gymnasium in Neuss zum ersten Mal Fuß. So instruierte uns unser Lehrer Philipp Kollat über die Findung einer Idee zur Lösung eines Problems. Daraufhin machten wir uns verschiedene Gedanken über die unterschiedlichsten Konzepte. Bei einem Spaziergang, zwei Tage nach Beginn der Entwicklungsphase einer Idee, kam uns die finale Idee. Als wir durch den Neusser Stadtgarten schlenderten, fiel mir auf, wie verdreckt die Wiesen waren. Von Brötchentüten über Kaffeebecher lagen verschiedenste Arten von Müll herum. Und das, obwohl regelmäßig Reinigungskräfte versuchen, jenen Müll weitestgehend zu räumen. Anschließend schlug ich vor, diesem Problem zumindest in Teilen ein Ende zu setzen. Wir entwickelten anschließend ein erstes Konzept. Unser Projekt war geboren. Eine sichere, nicht zu arbeitsintensive und komfortable Lösung zur Reinigung jener Wiesen und Grünflächen.

2. Wie kann so eine Lösung realisiert werden?

Wir stellten unser Konzept im nächsten Schritt vielen Lehrern aus unterschiedlichen Fachbereichen vor. Die konstant positive Resonanz und auch positives Feedback durch andere, außenstehende Personen, bestätigte Jonas und mich in unserer Idee, sodass wir von nun an begannen, diese fortzuentwickeln. So erstellten wir bereits in den ersten Treffen auf teils privaten Treffen erste Konzepte, die bereits auf verschiedene Aspekte und Problematiken eingingen, die auf unser Projekt zukommen könnten.

2.1. Grundlegende Umsetzung

Wir befassten uns zunächst mit dem Grundgedanken, mit welcher Methode etwas, in unserer Vorstellung roboterartiges, Müll auf Wiesen erkennen und selbständig einsammeln kann. Dabei war aus verschiedensten logischen Gründen schnell der Entschluss gefasst, dass wir ein fahrendes Objekt bauen müssen.

2.1.1. Antriebsart

Die Diskussion stand nun im Raum, ob wir auf geländefähige Räder oder ein Kettenkonstrukt zurückgreifen sollten. Wir verglichen dabei Fahrzeuge, die sich in einem solchen Terrain bewegen. Schnell fiel uns auf, dass Räder zwar eine gute Option darstellen, Fahrzeuge diese aber vor allem bevorzugen, um auch auf Straßen fahren zu können. Da unser Roboter jedoch fast ausschließlich auf ungefestigtem Gelände wie Wiesen, Erde und generellen Grünflächen fahren wird, benötigt er auf gefestigten Wegen und Straßen keine guten Fahreigenschaften, wie eine schnelle Geschwindigkeit. Dabei zeigten uns auch andere, in ähnlichem Terrain operierenden Fahrzeuge, wie beispielsweise Panzer, dass ein solcher Antrieb sich bewährt. Ketten sind zudem deutlich einfacher zu bauen, anzutreiben und weniger fehleranfällig. Sie können viele verschiedene Umstände aushalten, über spitze Gegenstände und durch tiefe Pfützen fahren und haben auch mit kleineren Hügeln keine Probleme. Wir entschieden uns daher, auch wegen der Komplexität des Baues eines solchen Antriebes, für ein kettengetragenes und gegen ein auf Rädern basierendes Fahrzeug.

2.1.2. Aufnahme, Transport und Lagerung des Mülls

Bevor wir uns nun mit der äußeren Form des Roboters beschäftigen, musste die konkrete Form des Aufnahmemechanismus und der inneren Transport- und Lagereinheit festgelegt werden. Nur anhand dieser Maße kann auch die „Hülle“ um das gesamte Konstrukt designt werden. Daher beschlossen wir, zunächst diesen Punkt in Angriff zu nehmen. Dabei unterteilten wir das Aufnahme- und Transportsystem in folgende Teilpunkte:

- 2.1.2.1. Erstkontakt und Aufnahme des Mülls des Untergrunds
- 2.1.2.2. Kontrolle des Mülls und Einführung in die innere Einheit
- 2.1.2.3. Entwicklung der Seiten- und Oberbürsten zur Unterstützung der Kontrolleinheit
- 2.1.2.4. interner Transport des Mülls mit Hilfe eines Förderbandes
- 2.1.2.5. Lagerung des Mülls innerhalb des Roboters

2.1.2.1. Erstkontakt und Aufnahme des Mülls

Zur Aufnahme des Mülls überlegten wir, welche Lösung am praktikabelsten wäre. Wir entschieden uns nach der Abwägung verschiedenster Konzepte für einen kammähnliches Konstrukt. Dabei fanden wir Orientierung an modernen Staubsaugerrobotern, welche mit einem ähnlichen Konstrukt arbeiten. Dabei streifen bei uns kleine, 3D- gedruckte Zähne durch den Rasen. Diese sind angeschrägt und rampenförmig, sodass der Müll auf die Ebene des Roboters gedrückt wird.

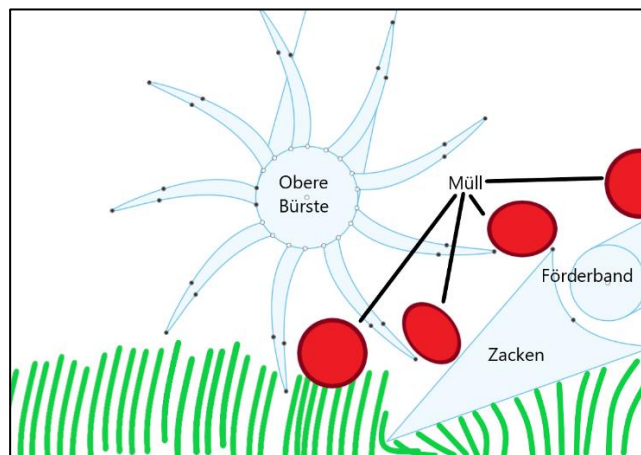


Abbildung 1: Skizze der Bürstenkonstruktion zur Aufnahme

2.1.2.2. Kontrolle des Mülls und Einführung in die innere Einheit

Um den Müll kontrolliert einzuführen und diesen lenken zu können, erweiterten wir unsere Aufnahme mit einer Bürstenkombination. Außerdem ließen wir Lücken zwischen den einzelnen Zacken, damit der Müll von dem unter ihm befindlichen Rasen separiert wird. Durch das Zusammenspiel dieser beiden Ideen konnten wir den Müll letztendlich kontrolliert von der Wiese trennen, aufnehmen und anschließend in unseren Roboter einführen, wo dieser dem internen Transportsystem übergeben wird.

2.1.2.3. Entwicklung der Seiten- und Oberbürsten zur Unterstützung der Kontrolleinheit

Da wir feststellten, dass der Müll zwar durch die Zacken recht gut kontrolliert werden kann, jedoch vor allem leichter Müll schnell dazu neigt, vor dem Roboter her geschoben zu werden, anstatt in diesen allein durch die Antriebskraft unseres Roboters geschoben zu werden, entwickelten wir ein auf Bürsten basierendes Unterstützungssystem. Dabei orientierten wir uns an schiebbaren Laubsammlern, welche ebenfalls mit ähnlichen Bürsten arbeiten, um Laub aufzunehmen. Dabei standen wir vor der Herausforderung, eine Bürstenart zu finden, welche weder zu weich ist, um schwere Gegenstände wie Glas einfach liegen zu lassen, noch zu stark ist, um auch Erde und Laub aufzunehmen. Wir entschieden

uns nach einer langen und vielseitigen Entwicklungsphase letztendlich für eine, auf dünnen 3D-gedruckten Plastikstäben basierende, Bürste. Diese sitzt, wie auf der Skizze zu sehen, direkt und sehr flach über den Zacken, um ihre unterstützende Wirkung zu erfüllen. Außerdem ist diese von einem Motor angetrieben, um die nötige Kraft auch auf die Müllobjekte übertragen zu können.

2.1.2.4. Interner Transport des Mülls mit Hilfe eines Förderbands

Nachdem der Müll nun erfolgreich von uns aufgesammelt wurde, muss dieser selbstverständlich noch in den im hinteren Teil des Roboters gelegenen Müllsammelbehälter transportiert werden, wo dieser bis zu seiner umweltfreundlichen Entsorgung gelagert wird. Dazu nutzen wir ein Flurförderband. Auch dieses wird von uns in der kommenden Entwicklungsphase unseres Projektes per additiver Fertigung (3D-Druck) gefertigt. Das Konzept sieht vor, dass das planmäßig ca. 50cm breite Förderband im inneren des Roboters direkt unterhalb der Zacken beginnt, um den fallenden Müll, welcher von der Bürste in den Roboter gedrückt wird, abzufangen. Anschließend fördert dieses mit einem leichten Profil auf dem Förderband den Müll in das hintere Segment unseres Roboters, während es durch seine leicht angeschrägte Position unseren aufgesammelten Müll auf eine höhere Ebene befördert, um diesen von oben in den Müllsammelbehälter fallen lassen zu können.

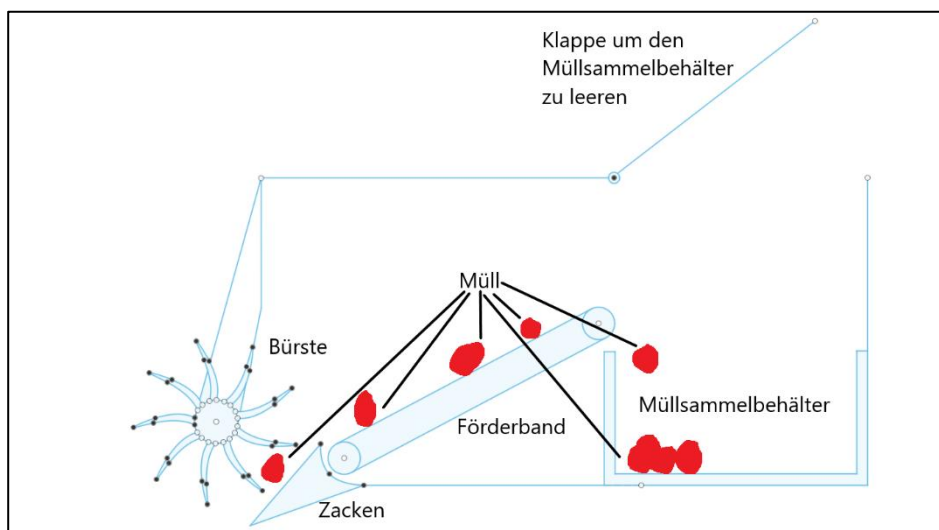


Abbildung 2: Skizze des Inneren des Roboters (unvollständig-Transport des Mülls im Fokus)

2.1.2.5. Lagerung des Mülls innerhalb des Roboters

Zur Lagerung des Mülls nutzen wir einen festen und aus Holz gebauten Behälter. Dieser lässt sich mit handelsüblichen 60L blauen Müllbeuteln bestücken. Zudem haben wir oberhalb des Behältnisses eine Luke, welche sich durch einen angebrachten Griff öffnen lässt, um den vollen Müllbeutel zu entfernen und durch einen leeren Müllbeutel zu ersetzen. In der weiteren Entwicklungsphase ist zudem geplant, in den Müllsammelbehälter einen Infra-Rot-Sensor einzubauen, welcher überprüft, ob der Beutel voll ist, um anschließend selbstständig zu einer Ausgangsposition zu fahren.

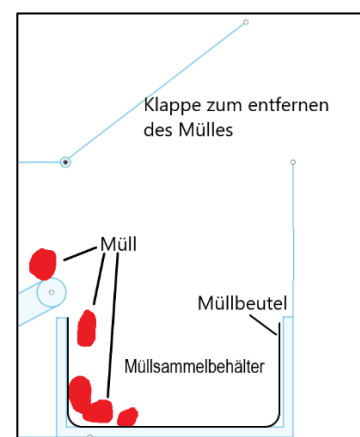


Abbildung 3: Skizze Auffangbehälter

2.2. Programmiertechnisches Segment

2.2.1. Erläuterung unserer Vorgehensweise

Einer der zentralen Punkte unseres Projekts ist die Programmierung einer künstlichen Intelligenz (im folgenden „KI“ genannt). Da wir durch die, wie in Kapitel 3 näher erläutert, sehr späte Bereitstellung eines geeigneten Rechners nur eine kurze Erprobungszeit für diese gehabt haben, wird auch dieser Sektor theoretisch und ohne Praxisnachweise abgehandelt. Außerdem ist durch die knappen Fristsetzungen der Code nicht vollständig vollendet und enthält an Stellen, welche Komponentenspezifisch geschrieben werden müssen, teilweise Lücken. Jedoch haben wir bereits eine genaue Vorstellung, wie sich unser Code gliedern wird und was den Inhalt dessen umfassen sollte.



Abbildung 4: KI-Erkennung von Müll I

2.2.2. Grundsätzliche Erklärung der Arbeitsweise unserer KI

Die KI beginnt im ersten Arbeitsschritte damit, mit Hilfe einer Logitech C270 HD-Webcam ein Live-Video (in der späteren Praxis dann in ihrer direkten Umgebung) aufzunehmen. Die KI basiert nun darauf, einzelne Bilder (in der Fachsprache „Frames“ genannt; ein Video basiert auf vielen, aufeinanderfolgenden dieser Bilder) zu analysieren. Im weiteren Verlauf kann die KI selbstständig erkennen, wo auf dem ihr gegebenen Bild ein Müllobjekt ist und dessen Koordinaten anschließend an die Steuerung unseres Roboters weitergeben. Diese wird dann in der Lage dazu sein, den Roboter zu dem Müllobjekt zu steuern.

2.2.3. Analyse des Livebildes und Weitergabe von Rohdaten

Die KI analysiert das ihr gegebene Bild auf 9,4 Millionen unterschiedliche Parameter wie beispielsweise Farbe, Form und Lichtreflexion, an welchen die künstliche Intelligenz mit einer prozentualen Wahrscheinlichkeit angeben kann, ob und wenn ja, wo sich in dem Bild Müll befindet. Ein Beispielversuch dazu lässt sich in folgendem Bild sehen.

Sobald die KI mit einer genügend hohen Wahrscheinlichkeit Müll identifiziert hat, gibt diese die Koordinaten jenes Mülls an ein weiteres internes System zur Steuerung der einzelnen Motoren und somit des gesamten Roboters weiter, welches die gegebenen Daten weiterverarbeitet und aus diesen Steuerbefehlen für die Ansteuerung der Motoren erstellt. Während dieses internen Vorgangs haben wir aus Effizienzgründen vor einzubauen, dass die KI bereits an der Analyse, welche Dauer wir im Übrigen mit 40ms (0,04 Sek) ansetzen, des nächsten Bilds arbeitet.



Abbildung 5: KI-Erkennung von Müll II

2.2.4. Verarbeitung durch das interne Steuersystem

Zunächst sorgt das Steuersystem durch eine ständige Aktualisierung des Standortes des Mülls im Bild dafür, dass sich der Roboter so dem Müll frontal zentriert ausrichtet. Wir überprüfen dies, indem wir den

Roboter einfach so lange drehen lassen, bis das Müllobjekt von der KI in der Bildmitte erkannt wird. Die KI gibt dabei immer nur die Information weiter, ob der Müll sich rechts oder links befindet. Wurde der Roboter nun erfolgreich mittig zentriert zu dem Müllobjekt ausgerichtet, folgt im nächsten Schritt die vertikale Bewegung. Hierbei gibt die KI auch wieder lediglich die Information weiter, auf welcher Höhe im Bild sich der Müll befindet. Nun lässt die Steuereinheit den Roboter so lange frontal geradeaus (durch die vorherige Zentrierung also direkt Richtung Müll) fahren, bis sich das Müllobjekt unmittelbar vor der Aufnahme des Roboters befindet. Ab diesem Punkt aktiviert die Steuereinheit, nach Bestätigung durch die KI, nun die Bürste und das Flurförderband. Durch diese zeitpunktangepasste Aktivierung und eine, somit nicht durchgehend drehende Bürste, ist der Roboter erstens energieeffizienter, und kann mit der gleichen Menge an Strom mehr Arbeit verrichten, und zweitens lässt sich damit verhindern, dass die Bürste während der Fahrt unnötige Objekte, wie beispielsweise Laub, in das innere unseres Roboters bürstet. Das Flurförderband und die Bürste werden anschließend für eine gewisse Zeitspanne aktiviert gelassen, bevor diese bis zu ihrem nächsten Einsatz abgeschaltet werden. Wir können uns in der weiteren Entwicklung an dieser Stelle eine Sensorbasierte, intelligentere Lösung vorstellen, die aktiv überprüft, ob sich noch Müll auf dem Flurförderband befindet, und dieses variabel aktivieren und deaktivieren kann.

2.2.5. Müllsuche und Ausweichautomatik

Sollte die KI keinen Müll in ihrem Sichtfeld wahrnehmen, wird sie diese Information an die Steuereinheit weitergeben. Diese wird den Roboter anschließend dazu anleiten, in geraden Linien auf und ab zu fahren und dabei in regelmäßigen Abständen eine Drehung zu vollführen. Dies soll dazu führen, dass der Roboter bisher unergründete Bereiche befährt, um neuen Müll zu sammeln. Außerdem wird mit Hilfe von LiDAR Sensoren, welche sich in einem 360° Sichtfeld um den Roboter befinden, überprüft, ob und wenn ja wo und wie schnell sich Objekte in der Umgebung des Roboters befinden. So hat der Roboter sowohl eine automatische Notabschaltung bei Annäherung von Personen sowie die Fähigkeit, Hindernisse wie Bäume und Büsche selbstständig zu erkennen und zu umfahren.

2.3. Umsetzung, genutzte Komponenten und Ausblick

2.3.1. Akkumulator und Spannungswandler

Das Herzstück unseres Roboters ist eine 100 Amperestunden große 12,8V Batterie. Mit dieser kann der Roboter planmäßig mindestens 2,5 Stunden bei vollem Betrieb arbeiten. An den Akkumulator werden wir einen 12V auf 5V Spannungswandler anschließen, welcher für die Stromversorgung unseres Hauptcomputers, dem Jetson Orin Nano Super, zuständig sein wird. Dieser Spannungswandler wird zwingend benötigt, da die Ausgabespannung des Akkus nicht mit der geforderten Eingabespannung des Jetson Orin Nano Super übereinstimmt.

2.3.2. Hauptcomputer Jetson Orin Nano Super und Steuereinheit

Der Jetson Orin Nano Super (im folgenden „Jetson“ genannt) ist die Hauptrecheneinheit unseres Roboters. Dieser ist für die Verarbeitung der Bilder mit Hilfe der KI zuständig, sowie für die Weitergabe der ermittelten Koordinaten im Bild an die zweite Recheninstanz in unserem Roboter, ein PCA9682 Microcontroller, der aus den im gegebenen Koordinaten Steuerbefehle für die Motoren

berechnet, und diese an einen Motorcontroller weiterleitet, welcher letztendlich die finalen Steuersignale (in Form von Stromstärke) an die Motoren lässt.

2.3.3. Warum diese Komponenten?

2.3.3.1. Mini Computer

Wir haben uns nach langer Recherchearbeit und vielen Suchanfragen auf den unterschiedlichsten Portalen für den, wie oben erwähnten „Jetson Orin Nano Super“ entschieden, da dieser derzeit mit Abstand einen der stärksten Microcomputer für das uns zur Verfügung gestellte Förderpensum von 300€ ist.

2.3.3.2. Antrieb

Für den Hauptantrieb der Ketten haben wir uns für zwei leistungsstarke 150-Watt-12-Volt-Gleichstrommotoren entschieden. Diese Motoren werden typischerweise in Elektro-Rollern verwendet und sind daher ideal für einen Roboter, der viel Kraft braucht, da er auf schwierigem Untergrund zurechtkommen muss. Zudem sind diese für stärkere Witterungsbedingungen ausgelegt und äußerst effizient. Wir benötigen zwei dieser Motoren, da wir jeweils eine unserer Ketten mit einem Motor in Bewegung versetzen. Der Grund für die Entscheidung zu unpräziseren Motoren war, dass diese sich zwar nicht exakt so genau ansteuern lassen, wie es bei Motoren mit integriertem Computer wäre, dafür aber in den Punkten Preis und Einfachheit der Implementierung in den übrigen Code klare Vorteile haben. Außerdem sind Gleichstrommotoren im Verhältnis zu anderen Arten von Motoren recht preiswert.

2.3.3.3. Sonstige Motoren und dazugehörige Komponenten

Bei jeglichen anderen Motoren wie beispielsweise für die Bürste und das Förderband haben wir uns für 12V 50W 57BYA-BLDC-Motoren entschieden. Diese Entscheidung beruht auf den Anforderungen, welche wir an die Motoren haben. Jene müssen keine sehr große Kraft erzeugen können, aber eine präzise Steuerung ermöglichen. Zudem bringen diese Motoren aufgrund ihrer Bauart den Vorteil mit sich, dass sie sehr energieeffizient sind und dabei eine hohe Langlebigkeit aufweisen. Ein kleiner, für uns ein wenig Extraaufwand bedeutender Nachteil dieser Motoren ist jedoch, dass man andere Motorcontroller für diese braucht. Wir haben uns dabei für die Bee BLHeli-s Feder 20A entschieden, da diese sehr akkurate Steuerung der Motoren für einen kleinen Anschaffungspreis bieten.

Die Verkabelung erfolgt mit 1,5 mm² Kupferkabeln, welche genügend Strom für die Anwendungszwecke im Roboter liefern können.

2.3.3.4. Kameras, andere Sensorik und mögliche nächste Schritte

Seine Umgebung wird unser Roboter wie in Kapitel 2.2 erläutert zunächst mit einer, planmäßig in der weiteren Entwicklung mit zwei Kameras analysieren. Dabei wird eine bereits in der jetzigen Bauplanung auf dem Roboterdach installiert werden, um ihre Umgebung nach Müll abzusuchen. Die zweite Kamera kann in einer weiteren Entwicklungsphase im Inneren des Roboters angebracht werden, sodass der Roboter mit Hilfe dieser, auch KI-gestützten Kamera, erkennen könnte, ob der Müll auf dem Förderband

angekommen ist und um welchen Müll es sich handelt. So wären genaueste Rückschlüsse auf die Vermüllung möglich, um beispielsweise Firmen auf ihr Produkt hinzuweisen.

Beide Kameras sind dabei Logitech C270 HD-Webcam, da diese eine kostengünstige Möglichkeit sind, ein qualitativ gutes Videosignal zu erhalten.

Zudem besteht die Möglichkeit, dass wir, einen GPS-Sensor einbauen, um ein per Karte festgelegtes Gebiet einstellen zu können, in welchem sich der Roboter bewegen darf. So würden wir sowohl die Sicherheit als auch die Effizienz des Roboters erhöhen, da dieser eine Stelle nur einmalig abfährt, und so nicht wie im bisherigen Plan einfache Linien abfährt. Zudem besteht so keine Möglichkeit mehr, dass der Roboter seinen festgelegten Bereich verlässt.

2.4. Training der künstlichen Intelligenz

Zu Beginn recherchierten wir, welches KI-Modell zur Bilderkennung am besten für unseren Anwendungsfall geeignet ist. Dazu haben wir die Ergebnisse von verschiedenen Modellen verglichen und dabei insbesondere Wert auf die benötigte Leistung des Computers gelegt. Bei dieser Recherche sind wir schnell auf das KI-Modell „Yolo“ (kurz für „You only look once“) gestoßen. Dieses Modell bietet den großen Vorteil, dass es verschiedene Versionen aus verschiedenen Generationen gibt, welche verschiedene Leistungsklassen und damit verbunden Ressourcenverbrauch abdecken.

Im nächsten Arbeitsschritt stand im Fokus, die für uns am besten passende Versionen von Yolo zu finden. Es standen nach einer kurzen Phase der Abwägung zwei Versionen der KI zur Auswahl: „yoloV5s“ und „yolov11s“. Beide Versionen haben eine hohe Genauigkeit im Verhältnis zu der benötigten Leistung, wobei „yoloV11“ um einiges präziser, dafür aber langsamer arbeitet.

Daraufhin begannen wir mit der Suche nach einem passenden Set an Bildern, um die KI zu trainieren. Diese werden immer benötigt, wenn sich ein neuer Anwendungsfall für ein KI-Modell ergibt, da diese standartmäßig nicht sonderlich gut in der Identifizierung von Müll sind. Wir entschieden uns für das vorgefertigte Bilderset „TACO“, da dieses kostenlose Set mit 4000 Bildern von Müll in der Umwelt, das größte, kostenlos verfügbare seiner Art ist.

Nachdem wir nun ein Dataset gefunden haben, und klar war, welche künstliche Intelligenz wir nutzen, begannen wir damit uns zu informieren, wie wir jene KI trainieren können. Nach einiger Recherche sind wir zu dem Schluss gekommen, dass wir die KI entweder online über einen Service Googles oder lokal auf einem eigenen Computer trainieren werden müssen. Wir haben uns dazu entschieden, das Training lokal laufen zu lassen, da wir bereits in dem Besitz eines ausreichend starken Computers sind. Dieser Umstand hat den Vorteil, dass wir die KI zudem auf einer Grafikkarte trainieren konnten, die von der gleichen Firma ist wie der Hauptrechner (Jetson) ist, und diese daher mit den gleichen Treibern programmiert werden können.

Für das Training der KI installierten wir „Torchvision“, damit die Computersprache Python die Grafikkarte benutzen kann.

Im Folgenden werden einzelne Zeilen Code erklärt, um diese nachvollziehen zu können. Die Erklärung sind mit einem „#“ markiert.

Danach klonen wir das „YOLOv5 Modell“ von der online Plattform „GitHub“ mit folgendem Code:

```
git clone https://github.com/ultralytics/yolov5.git
cd yolov5
pip install -r requirements.txt
```

Danach mussten wir das „TACCO“ Dataset anpassen, da in diesem zwischen 18 verschiedenen Arten von Müll unterschieden wird, wir jedoch nur die allgemeine Information benötigen, ob es sich bei einem Objekt um Müll handelt oder nicht. Diesen Prozess haben wir mit einem Python-code automatisiert, da die schiere Anzahl der Bilder eine manuelle Abänderung nur schwer möglich macht. Dazu nutzen wir folgenden Code:

```
import os # Modul zum Arbeiten mit Dateisystempfaden und Verzeichnissen importieren

# Pfad zu den YOLO-Label-Dateien
labels_path = "path/to/labels"

# Alle Dateien und Verzeichnisse im angegebenen Pfad rekursiv durchlaufen
for root, _, files in os.walk(labels_path):
    # Durch alle Dateien im aktuellen Verzeichnis iterieren
    for file in files:
        # Nur Dateien mit der Endung ".txt" verarbeiten
        if file.endswith(".txt"):
            # Vollständigen Pfad zur Datei erstellen
            file_path = os.path.join(root, file)

            # Die Datei im Lesemodus öffnen und alle Zeilen einlesen
            with open(file_path, 'r') as f:
                lines = f.readlines()

            # Die Datei im Schreibmodus öffnen, um die bearbeiteten Zeilen zu speichern
            with open(file_path, 'w') as f:
                for line in lines:
                    # Zeile aufteilen, um die einzelnen Teile (Klassen-ID und Koordinaten) zu erhalten
                    parts = line.strip().split()
                    # Die Klassen-ID (erstes Element) auf '0' setzen
                    parts[0] = '0'
                    # Die bearbeitete Zeile wieder in die Datei schreiben
                    f.write(" ".join(parts) + "\n")

# Ausgabe, um den Abschluss des Prozesses zu bestätigen
print("Alle Klassen wurden auf 'müll' (0) umgestellt.")
```

Danach passten wir vereinzelt mit folgendem Code Dateien wie die „data.yml an

```
python train.py --img 640 --batch 64 --epochs 200 --data data.yml --weights yolov5s.pt --device 0
```

Nach circa drei Stunden erfolgreichem Training waren wir fertig und konnten die KI ein erstes Mal an einem normalen Bild testen:



Abbildung 7: KI-Erkennung von Müll III



Abbildung 6: Beispiel einer fehlerhaften Erkennung durch die KI

```
python detect.py --weights runs/train/exp/weights/best.pt --img 640 --source path/to/test/images
```

Wir waren jedoch mit dem Ergebnis des aktuellen Modelles nicht zufrieden, da es wie auf den Bildern zu erkennen ist nicht sehr präzise ist.

Daher haben wir auch noch ein „yolov11“ Modell trainiert, da dieses zwar mehr Leistung braucht, aber auch stärker ist. Dazu haben wir es wieder erst heruntergeladen:

```
git clone https://github.com/ultralytics/yolov11.git
cd yolov11
pip install -r requirements.txt
```

Und mit Python trainiert, da dies für jenes Modell leichter geht.

```
from ultralytics import YOLO
```

```
model = YOLO("yolo11n.yaml").load("yolo11s.pt")
model.train(data="C:\\Users\\Nutzer\\Downloads\\dataset\\data.yaml", epochs=200, imgsz=640)
```

Nachdem wir die KI diesmal in Python getestet haben, waren wir mit dem Ergebnis zufrieden.

```
import ultralytics
from ultralytics import YOLO
import os

# Pfad zum Speicherordner
save_path = r"C:\\Users\\Nutzer\\Downloads\\results"

# Stelle sicher, dass der Ordner existiert
os.makedirs(save_path, exist_ok=True)

#Stellt sicher, dass das Programm nur einmal ausgeführt wird
if __name__ == "__main__":
    #lädt das richtige Modell
    model = YOLO("C:/Users/Nutzer/runs/detect/train6/weights/best.pt")
    #lädt die Bilder
    results = model("C:/Users/Nutzer/Downloads/test_images")

# Verarbeitet die Ergebnisse
for i, result in enumerate(results): # Verwende enumerate für den Zähler
    #erstellt die Boxen
    boxes = result.boxes
    #erstellt die Dateinamen
    filename = os.path.join(save_path, f"{i}_result.jpg")
    #Speichert die Ergebnisse
    result.save(filename)
```

Danach haben wir versucht, unsere Logitech C270 Webcam an den Computer anzuschließen und ein Live Video mit der KI auszugeben. Dies haben wir mit diesem Code erfolgreich geschafft.

```
from ultralytics import YOLO
import cv2
import time

# YOLO-Modell laden
model = YOLO("C:/Users/jonas/runs/detect/train6/weights/best.pt")

# Eingabegröße des Modells auf 1280x736 setzen (Vielfaches von 32)
model.overrides['imgsz'] = (736, 1280)

# Kamera öffnen (ID = 0)
```

```

cap = cv2.VideoCapture(0)

# Kameraauflösung auf 1280x736 setzen (angepasste Höhe)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 1280)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 736)

# Prüfen, ob die Kamera geöffnet werden konnte
if not cap.isOpened():
    print("Fehler: Kamera konnte nicht geöffnet werden.")
    exit()

print("Kamera geöffnet. Drücke 'q', um die Anwendung zu beenden.")

# Startzeit für FPS-Berechnung
start_time = time.time()

while True:
    # Kamerabild einlesen
    ret, frame = cap.read()
    if not ret:
        print("Fehler beim Lesen des Kamerastreams.")
        break

    # YOLO-Vorhersage auf dem Kamerabild
    results = model.predict(source=frame, conf=0.5, save=False, show=False) # `conf` bestimmt die
    Mindestwahrscheinlichkeit

    # Ergebnisse auf das Bild zeichnen
    annotated_frame = results[0].plot() # Annotiertes Bild (mit Bounding Boxes und Labels)

    # FPS berechnen
    fps = 1 / (time.time() - start_time)
    start_time = time.time()

    # FPS auf das Bild schreiben
    cv2.putText(
        annotated_frame,
        f"FPS: {fps:.2f}",
        (10, 30),
        cv2.FONT_HERSHEY_SIMPLEX,
        1,
        (0, 255, 0),
        2,
        cv2.LINE_AA,
    )

    # Annotiertes Bild anzeigen
    cv2.imshow("YOLO Live - 1280x736", annotated_frame)

    # Schleife mit Taste 'q' beenden
    if cv2.waitKey(1) & 0xFF == ord('q'):
        print("Anwendung beendet.")
        break

# Ressourcen freigeben
cap.release()

```

`cv2.destroyAllWindows()`



Abbildung 8: KI-Erkennung von Müll IV



Abbildung 9: KI-Erkennung von Müll V

3. Problematiken

Bei der Entwicklung unseres Roboters kamen uns verschiedene Probleme und Hürden auf uns zu, welche sich in folgende Kategorien unterteilen ließen:

- 3.1 Finanzierung
- 3.2 Umsetzung
- 3.3 Zeit und Fristen
- 3.4 Ungeeignete Komponenten

3.1. Finanzierung

Wie sicherlich bei einem Projekt dieser Größenordnung üblich, ist die Finanzierung aller benötigten Materialien eine wichtige und für die weitere Projektentwicklung äußerst relevante zu klärende Frage. Wir gingen an diese recht systematisch heran und entwickelten zur ungefähren Einschätzung der Kostenfrage mehrere Versionen von Kostenaufstellungen, die verschiedenste Modelle und Materialien in den unterschiedlichsten Qualitätsstufen beinhalteten. Jene Kostenaufstellungen waren selbstverständlich mit hohem Rechercheaufwand und einer Vielzahl von theoretischen Konzepten unseres Roboters verbunden. In dem anschließenden Prozess der Entwicklung befassten Jonas und ich uns in enger Zusammenarbeit mit unserem Lehrpersonal, den zuständigen Ehrenamtlern des AvHG-Fördervereins und Mitarbeitenden des Jugend Forscht Sponsoren- und Förderpools mit einer möglichst präzisen und qualitativ angemessenen finalen Kostenaufstellung, welche wir zu Förderungszwecken anschließend über den betreuenden Lehrer Philipp Kollat an Mitarbeiter des Sponsorenpools weitergaben. Nach einigen Wochen Korrespondenz, vielen Rückfragen und Antworten, welche unsere weitere Arbeit zu diesem Zeitpunkt leider so gut wie zum Erliegen brachte, stand fest, dass das Projekt SWEEP mit einer genehmigten Fördersumme von 1572€ gefördert wird. Die war für uns eine äußerst wichtige Nachricht und ein wichtiger Schritt hin zu einem funktionierenden Endprodukt. Zudem erklärte sich der Förderverein unserer Schule dazu, weitere Ausgaben, welche nicht in der Kostenaufstellung berücksichtigt oder vergessen wurden, mit einer Fördersumme von 200€ zu unterstützen. Damit war der Fortbestand unseres Projektes gesichert und der Weg hin zu einem finalen Ergebnis geebnet.

Das Problem der Finanzierung war somit gelöst!

3.2. Umsetzung

Nach Beendigung der theoretischen Entwicklungsphase unseres Roboters, stand als nächstes die Frage der Umsetzbarkeit im Raum. Viele der von uns theoretischen Ideen mussten und müssen zum jetzigen Zeitpunkt noch mit den uns wie in Kapitel 3.1 erst kürzlich zur Verfügung gestellten Materialien in die Realität umgewandelt werden. Da wie erwähnt durch die späte Bereitstellung jeglicher Fördermittel der physisch-bautechnische Teil unseres Projektes bis kurz vor Abgabe dieser Projektarbeit fast vollständig pausiert war, und wir lediglich an der theoretischen Ausarbeitung auf Papier, dem informieren über verschiedene Materialien sowie der Programmierung an der KI arbeiten konnten, haben wir zu dem jetzigen Zeitpunkt nur einen sehr kleinen Teil unseres Projektes praktisch umsetzen können. Jedoch haben wir unsere Zeit vor allem auf diese Projektarbeit fokussiert, um mit dieser einen möglichst akkuraten Überblick verschaffen zu können, welche weiteren Schritte noch anstehen, und wohin jene Fördermittel planmäßig fließen werden.

3.3. Zeit und Fristen

Wie in Kapitel 3.2 bereits angerissen, stellte vor allem die verspätete Bereitstellung der Förderung das größte aller Probleme für uns und unser Projekt dar. Durch den langwierigen Prozess von Beantragung der Förderung bis zur letztendlichen Genehmigung, welche erst einen Monat vor Ende der Abgabefrist final genehmigt wurde, wodurch die ersten Teile nach den Weihnachtsfeiertagen eintrafen, haben wir für die Entwicklung der physischen Komponenten unseres Projektes bis zum jetzigen Zeitpunkt, nur einen Zeitraum von etwa zwei Wochen gehabt. Da wir der Jury die theoretische Größe unseres Roboters nach Fertigstellung jedoch realistisch näherbringen wollen, wurde auch diese schriftliche Arbeit mit näher kommen der Abgabefrist zunehmend zu einer großen Baustelle, wodurch das praktische Bauen noch weiter in den Hintergrund gerückt ist. Daraus resultierend haben wir, wie in Kapitel 3.2 erwähnt, nur einen äußerst kleinen Teil eines, wie in dieser Arbeit hoffentlich anschaulich dargestellt, so viel größeren Projektes, in die Realität umsetzen können.

3.4. Ungeeignete Komponenten

Es war schnell klar, dass wir unser Projekt auf einem Jetson Nano machen werden, jedoch noch nicht auf welchen, da es entweder den Jetson Orin Nano für 600€ mit ca. 37.000 Gigaflops gibt oder den Jetson Nano B02 mit nur ca. 450 Gigaflops, welcher allerdings auch nur 290€ gekostet hat. Das Problem war, dass wir zunächst nur ca. 300€ Budget hatten für den Computer hatten, weshalb wir uns für diesen entschieden haben, auch wenn dies zur Folge hat, dass wir nur Ca 1 Bild die Sekunde analysiert bekämen, was die Programmierung deutlich schwerer machen würde.

Doch bereits 2 Tage nachdem dieser bestellt wurde, hat Nvidia den Jetson Orin Nano Super vorgestellt, welcher zum gleichen Preis wie der B02 Mehr Leistung als der Orin Nano hat. Dieser war allerdings sehr schnell ausverkauft und sollte erst wieder Ende Januar 25 verfügbar sein. Daher war dann die Frage, ob man den B02 zurückschicken sollte, um den anderen vorzubestellen, aber nicht zu wissen, wann genau dieser kommt. Daher entschieden wir uns erst, den B02 zu behalten. Als jedoch klar wurde, wie groß die Hürde sein würde entschieden wir uns eine Rücksendung zu beantragen und den Orin Nano Super vorzubestellen.

Fazit des Projekts "S.W.E.E.P."

Dieses Projekt zeigt eindrucksvoll, wie Technologie zur Lösung eines Umweltproblems genutzt werden kann. Mit einer KI-basierten Müllerkennung, autonomen Navigation und einer effizienten Mechanik wurde ein innovativer Ansatz entwickelt.

Dabei konnten wir zahlreiche Erfolge erzielen. Zunächst gelang uns erfolgreich die Entwicklung und das anschließende Training einer künstlichen Intelligenz, die Müll erkennt und in der Lage ist, den Roboter präzise zu diesem zu navigieren. Auch die erfolgreiche Implementierung und der anschließend einwandfrei funktionierende Einsatz leistungsstarker Komponenten wie des Jetson Orin Nano Super, gelangen uns nach langer und ausgeprägter Recherchearbeit und einer langen, sehr theoretischen Arbeitsphase, in diesem Projekt.

Zu dieser gehörten in ihren Kernpunkten sowohl eine detaillierte Planung für die Programmierung und den Aufbau unseres künftigen Codes sowie eine genaue Abstimmung der internen Elektronik und eine Überprüfung der Kompatibilität dieser untereinander.

Wir stießen bei der Entwicklung dieses Projektes dennoch auf große Herausforderungen. Zum einen führten die zeitlichen Einschränkungen, welche uns gegeben waren, zu einem aktuell nicht vollendeten Ergebnis bei Abgabe dieser Arbeit. Eng mit dem zeitlichen Aspekt verbunden ist zusätzlich noch die verzögerte Bereitstellung der Finanzierung gewesen, welche jegliche physische Entwicklung des Roboters bremste und die praktische Umsetzung folglich nahezu komplett zum Erliegen brachte.

Für den kommenden Zeitraum bis zum Veranstaltungstermin des Jugend-forscht-Wettbewerbes 2025 haben wir uns große Ziele gesetzt. In unserem ersten bevorstehenden Schritt steht die Fertigstellung des Codes, anschließende Testläufe dieses sowie das Erproben möglicher, teils physischer, teils aber auch materieller Erweiterungen wie einem GPS-Sensor, oder das Arbeiten mit einer zweiten Kamera. Diese sollen die Effizienz des Roboters steigern, seine Sicherheit noch weiter erhöhen und zusätzlich eine Möglichkeit der Datensammlung über den von uns gesammelten Müll ermöglichen, um künftig eventuell Firmen auf ihre Produkte hinweisen zu können.

"S.W.E.E.P." hat das Potenzial, eine nachhaltige, effiziente, kostengünstige und personalarme Lösung für die Säuberung von Parks, Wiesen oder öffentlichen Grünanlagen zu bieten. Wir werden unser Bestes geben, der Allgemeinheit auch in der weiteren Entwicklung unseres Projekts ein Maximum dieses Potenzial durch unsere Fähigkeiten zur Verfügung zu stellen.

Quellenverzeichnis

Informationen in der Projektarbeit

1. <https://www.bmu.de/faq/wie-viele-einweg-becher-werden-jaehrlich-verbraucht-wie-viele-landen-davon-in-der-umwelt>

Yolo Modelle und Dataset

2. <https://github.com/ultralytics/yolov5>
3. <https://github.com/ultralytics/ultralytics>
4. <https://docs.ultralytics.com/models/yolo11/>
5. <https://docs.ultralytics.com/models/yolov8/>
6. <https://www.kaggle.com/datasets/vencerlanz09/taco-dataset-yolo-format>
7. <http://tacodataset.org/>
8. <https://de.shaip.com/offerings/open-datasets/>

Jetson Nano

9. <https://www.nvidia.com/de-de/autonomous-machines/embedded-systems/>
10. <https://www.nvidia.com/de-de/autonomous-machines/embedded-systems/jetson-nano/product-development/>
11. <https://concurrent-rt.com/partners/technology/nvidia/>
12. <https://www.youtube.com/watch?v=S9L2WGf1KrM>

Programmierung und Training

13. <https://www.youtube.com/@CodeWithAarohi>
14. <https://www.youtube.com/watch?v=mUybgOISxxA>
15. <https://discord.gg/Jq6wBnW7>

Technische Komponenten

16. <https://www.omc-stepperonline.com/de/12v-3000rpm-0-16nm-50w-5-90a-57x57x53-5mm-buerstenloser-gleichstrommotor-57bya54-12-01>
17. <https://info.sycotec.eu/elektromotoren-arten/>
18. <https://de.rs-online.com/web/content/discovery-portal/produktatgeber/dc-motoren-leitfaden>
19. <https://elektro.turanis.de/html/prj147/index.html>
20. <https://image.dfrobot.com/image/data/DRI0018/BTS7960.pdf>

KI in der Arbeit

Zu Recherchezwecken wurden in dieser Arbeit auch Informationen verarbeitet, welche von künstlicher Intelligenz bereitgestellt wurden. Jedoch wurde zum Schreiben dieser Arbeit NICHT auf die Hilfe künstlicher Intelligenz zurückgegriffen und alle Fakten wurden per Hand überprüft.