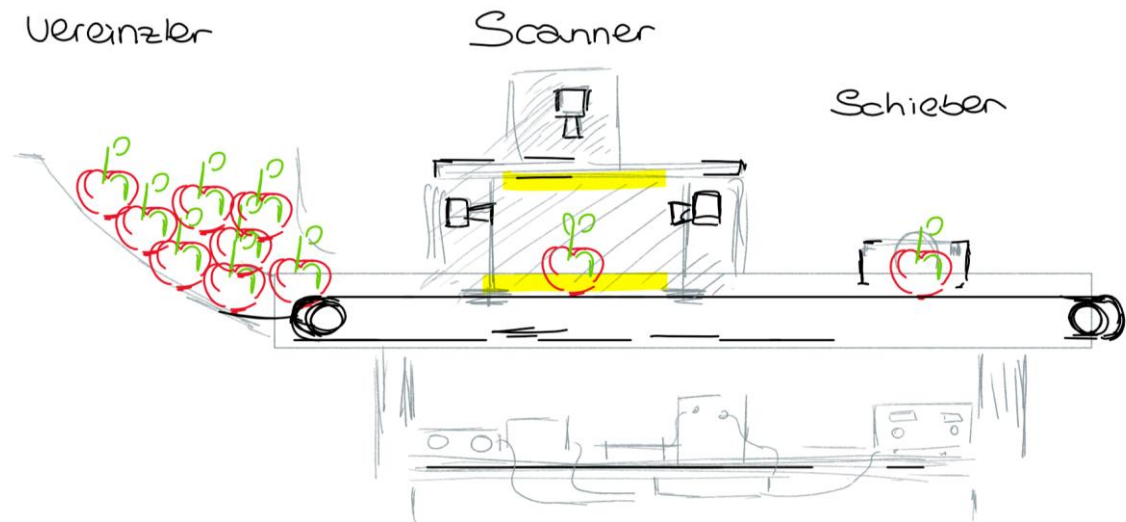


# OptiSort

KI basiert, voll automatisch und universell



Bastian Schifano

Tobias Rick

Tim Teuffel

# **1 Datenblatt**

## **1.1 Teilnehmer**

Bastian Schifano, 18 Jahre alt

Tobias Rick, 18 Jahre alt

Tim Teuffel, 18 Jahre alt

## **1.2 Schule**

Berufliches Schulzentrum Leonberg, Fockentalweg 8,

71229 Leonberg

## **1.3 Projektbetreuer**

Dipl. Ing. Dominique Haas

Dr. Rainer Hummel

Dr. Josef Kleckner

## **1.4 Thema**

OptiSort: KI basiert, voll automatisiert und universell

## **1.5 Wettbewerbssparte**

Jugend forscht

## Inhalt

<b>1</b>	<b>DATENBLATT .....</b>	<b>2</b>
1.1	TEILNEHMER .....	2
1.2	SCHULE .....	2
1.3	PROJEKT BETREUER.....	2
1.4	THEMA.....	2
1.5	WETTBEWERBSSPARTE .....	2
<b>2</b>	<b>KURZFASSUNG .....</b>	<b>4</b>
<b>3</b>	<b>IDEENFINDUNG.....</b>	<b>4</b>
3.1	PROBLEMSTELLUNG .....	4
3.2	ZIELSETZUNG .....	4
3.3	LÖSUNGSANSATZ .....	4
3.4	UNSER REALISierter LÖSUNGSANSATZ.....	5
<b>4</b>	<b>AUFBAU UND BESCHREIBUNG DES OPTISORT .....</b>	<b>5</b>
4.1	VORGEHENSWEISE.....	5
4.2	AUFBAU .....	5
4.3	KONSTRUKTION .....	6
4.4	BLOCKSCHALTBIld .....	7
4.5	HARDWARE.....	8
4.5.1	<i>Elektronik.....</i>	<i>8</i>
4.5.2	<i>Mechanik.....</i>	<i>10</i>
4.6	SOFTWARE .....	12
4.6.1	<i>Einleitung.....</i>	<i>12</i>
4.6.2	<i>Software-Probleme und Lösungen .....</i>	<i>13</i>
4.6.3	<i>Begriffsdefinitionen .....</i>	<i>14</i>
<b>5</b>	<b>FAZIT.....</b>	<b>15</b>
<b>6</b>	<b>AUSBLICK.....</b>	<b>15</b>
6.1	POTENTIELLE ANWENDUNGEN .....	15
6.2	VERBESSERUNGSMÖGLICHKEITEN .....	16
<b>7</b>	<b>ERGEBNISSE.....</b>	<b>16</b>
<b>8</b>	<b>DANKSAGUNG .....</b>	<b>17</b>
<b>9</b>	<b>LITERATURVERZEICHNIS.....</b>	<b>17</b>

## 2 Kurzfassung

Künstliche Intelligenz wird immer sichtbarer in unserer Arbeitswelt und im Alltag. Sie kann beispielsweise optische Merkmale selbst erkennen und ist so in der Lage, Dinge visuell und ohne klare Regeln zu kategorisieren. Ein sinnvolles Anwendungsbeispiel fanden wir in der Apfellagerung, wo faule Äpfel stetig aussortiert werden müssen, ehe sie Nachbarobst befallen.

Wir haben uns die Aufgabe gestellt, faule Äpfel von Guten optisch, sprich durch eine KI bzw. ein neuronales Netzwerk auszusortieren und dabei ein Maximum an Automatisierung in den Prozess zu bringen.

Was dann entstand, war der OptiSort, unsere vollautomatisierte, KI-basierte und mittlerweile auch universelle Sortiermaschine, welche durch Förderband, pneumatischem Vereinzeler und elektrischen Schieber ebenfalls mechanisch aussortiert. Die von 5 Kameras ermittelten Messdaten verarbeitet ein Computer und die Mechanik wird von einem separaten Microcontroller koordiniert.

## 3 Ideenfindung

### 3.1 Problemstellung

Unsere Idee entstand aus dem Problem des mühsamen Schlechtobst-Aussortierens bei Bekannten. Diese verbringen oft viel Zeit damit, ihr ungespritztes und deshalb häufig beschädigtes Obst kontinuierlich zu kontrollieren.

Denn: ist im Beispiel von Äpfeln ein Apfel faul, so überträgt sich die Fäule schnell auf das Nachbarobst. Umso notwendiger ist es deshalb, durch stetiges Kontrollieren Schlechtobst zeitnah auszusortieren, ehe es mehr Schaden anrichtet.

### 3.2 Zielsetzung

Wir setzen uns nun das Ziel, Schlechtobst im Beispiel von Äpfeln vollautomatisiert und zuverlässig aus einem großen Pool an Äpfeln auszusortieren, um unseren Bekannten in ihrem Problem zu helfen.

### 3.3 Lösungsansatz

Damit in Zukunft Mühen, Zeit und Kosten im privaten als auch konventionellen Obstbetrieb beim Obstsortieren gespart werden können, kamen für uns folgende Möglichkeiten in Frage:

- Ein Röntgenverfahren, welches auch Schädlinge unterhalb der Oberfläche sieht und somit sehr gründlich aber ebenso komplex, teuer und durch die Strahlung auch gefährlich sein kann.
- Ein Licht Verfahren, dass in der Theorie durch riesige Mengen Licht Obst o.Ä. durchleuchtet und deren Inneres auf eine Fläche projiziert.
- Ein Optisches Scan Verfahren mit handelsüblichen Kameras, in Komplexität und Anwendung überschaubar ist.

Alle Messdaten der entsprechenden Verfahren werden in Form eines abstrakten oder konkreten Fotos des zu untersuchenden Objekts gesammelt und müssen im zweiten Schritt folglich von einer KI bzw. einem Neuronalen Netzwerk bewertet werden. Die Kriterien bzw. Merkmale für die einzelnen Klassen muss die KI aber selbst finden

bzw. erlernen, um schlussendlich wie ein Mensch anhand von Erfahrung zu wissen, welche Indizien auf den Fotos auf welchen Zustand der Äpfel schließen.

### 3.4 Unser realisierter Lösungsansatz

Wir haben uns für das optische Scanverfahren mit handelsüblichen Kameras entschieden, da wir diese Variante am praktikabelsten fanden.

Die ermittelten Messdaten wollen wir von einer vorab trainierte KI bewerten, umso automatisiert über den Zustand der Äpfel zu urteilen.

Das Röntgenverfahren schien durch teure Komponenten und hohe Sicherheitsstandards leider nicht umsetzbar, das Lichtverfahren scheiterte in einer Vorabtestreihe am diffusen Fruchtfleisch von zum Beispiel Trauben. Die flüssige Materie im Inneren der Früchte brach das Licht so stark, dass selbst große Kerne auf der projizierten Fläche zu undeutlichen Punkten verschwammen und so jegliches Urteil über etwaige Schädlinge im Inneren unmöglich macht.

## 4 Aufbau und Beschreibung des OptiSort

### 4.1 Vorgehensweise

Bevor man sich um die Umsetzung des Projekts kümmert, versucht man theoretisch die von Anfang an auftretenden Problemstellungen zu lösen.

Bei unserem Projekt waren die großen Problemstellungen:

- Kann eine KI unsere Messdaten, sprich Fotos zuverlässig auswerten und selbst erkennen, wie z.B. ein fauler/ schlechter/ kaputter Apfel aussieht bzw. sich von einem guten Apfel unterscheidet?
- Wie viele Fotos/ Referenzbilder zum Trainieren braucht es, bis eine KI zuverlässig faule Äpfel erkennt?
- Wie viel Rechenleistung und Zeit braucht unsere KI im laufenden Betrieb?
- Wie wirken sich Apfelsorte, Lichtveränderung und Grad der Fäule auf das Urteil der KI aus?

All diese Fragen haben wir im Laufe des Projekts gelöst.

### 4.2 Aufbau

Um konstante Rahmenbedingungen (gleicher Kamerawinkel, gleiche Lichtverhältnisse, Position des zu scannenden Objekts, ...) in das Fotografieren der Äpfel zu bringen und gleichzeitig den Prozess automatisieren zu können, entschieden wir uns für folgenden Aufbau:

### 4.3 Konstruktion

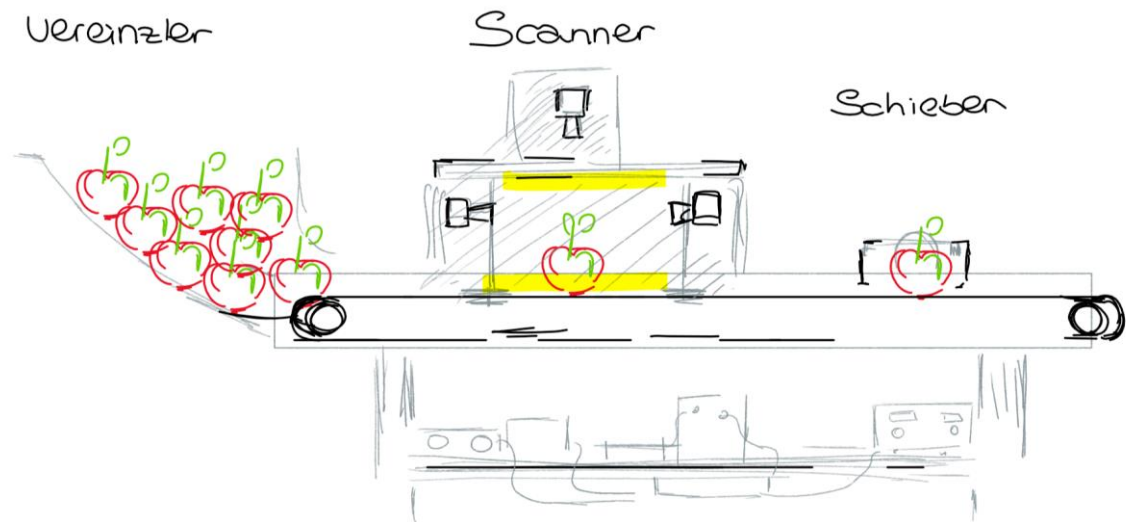


Abbildung 1 Skizze fertiger Maschine

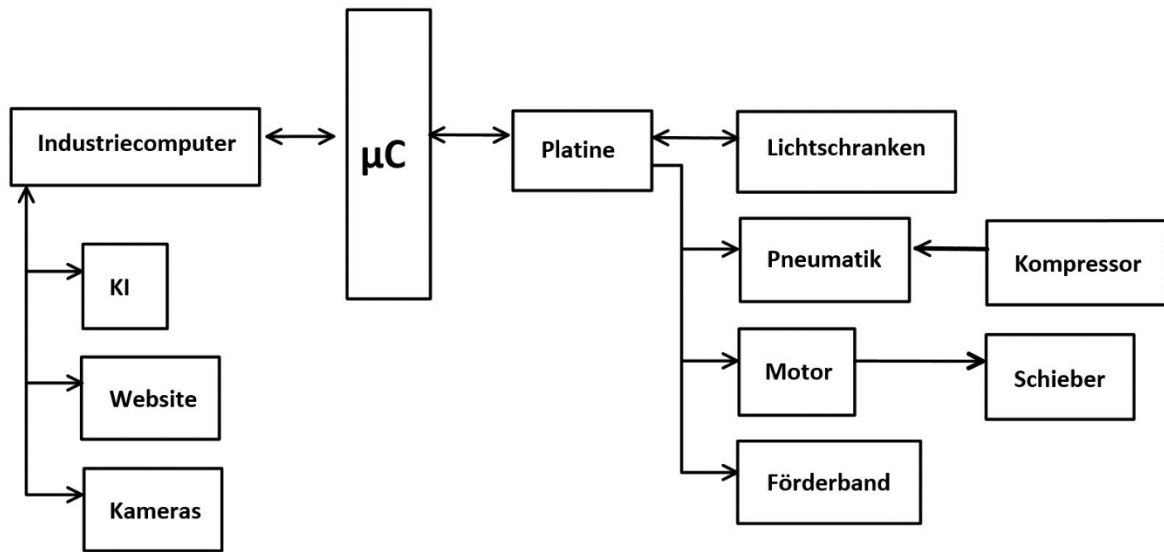
#### Beweggründe für Konstruktion

- Als Basis dient ein extern zukaufbares Förderband (spart Zeit und Geld)
- unterhalb des Förderbands befindet sich ausreichend Platz für Netzteil, Platine und Computer
- Lichtschranken, LED Beleuchtung, Kameras sowie Schieber und Vereinzelner können einfach adaptiert werden
- zu scannendes Objekt kann durch das Förderband automatisch alle Stationen anfahren

#### Maße des OptiSort

- 1700mm x 1200mm x 500mm

## 4.4 Blockschalbild



Die Maschine ist so konzipiert, dass ein Mikrocontroller (μC) verschiedene Prozesse koordiniert, um individuelle Objekte anwendungsorientiert zu sortieren. Ein konkreten Sortierprozess wird exemplarisch am Beispiel von Äpfeln erklärt:

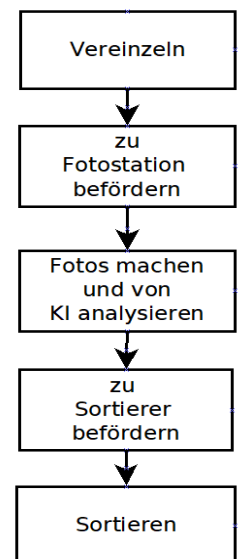
1. Zuerst wird immer genau ein Apfel auf das Förderband geschoben. Innerhalb des von uns eigens entwickelten Vereinzlers fährt dafür ein Pneumatik Zylinder hierzu aus.

2. Ein Förderband transportiert den Apfel weiter in die Fotostation. Dort stoppt der Apfel positionsgenau, da er im Inneren der Station eine Lichtschranke durchbricht.

3. Das Fotolicht wird aktiviert, der Computer wird informiert, 5 Kameras fotografieren den Apfel von 5 Seiten. Die Bilder wertet der Computer mit einer vorab trainierten KI aus.

4. Ein Förderband fährt den Apfel heraus aus der Fotostation weiter zum Sortierer. Dort stoppt der Apfel erneut positionsgenau, da er erneut eine Lichtschranke durchbricht.

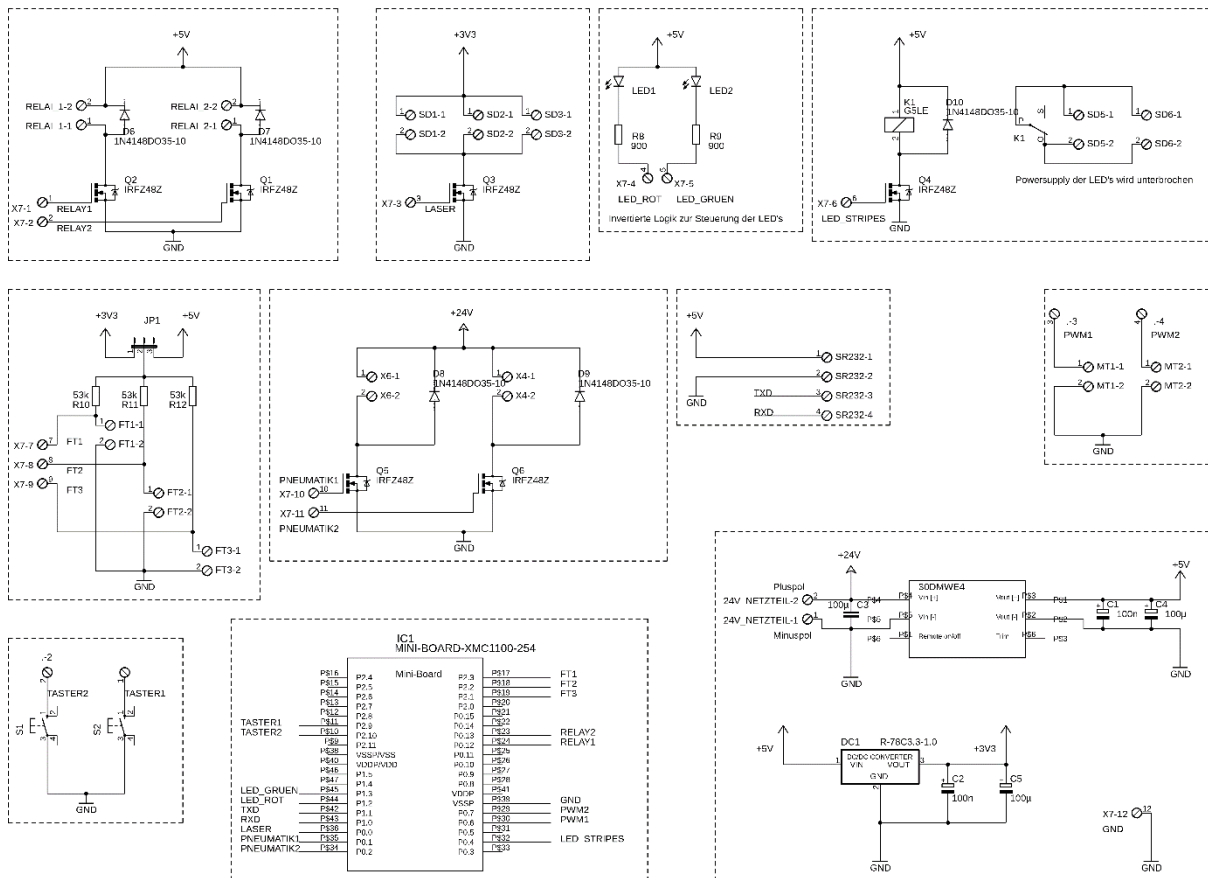
5. Das Ergebnis der KI Analyse wird dem μC mitgeteilt, der Apfel kann, sofern er als schlecht eingestuft wurde, am Sortierer vom Förderband geschoben werden. Da guteingestufte Äpfel weiter auf dem Förderband fahren, ergeben sich am Ende des Prozesses zwei getrennte Orte für die in zwei Kategorien sortierte Objektklasse oder im Falle von Äpfeln ein Eimer mit Guten und ein Eimer mit schlechten Äpfeln.



Das Ergebnis der KI-Auswertung sowie die aufgenommenen Bilder werden ebenso temporär auf der Weboberfläche angezeigt.

## 4.5 Hardware

### 4.5.1 Elektronik





### Diese Schaltung wurde entwickelt, um folgendes zu erreichen

- digitale Ansteuerung von Förderband, LED Stripes, Lasern, Pneumatik Magnet Ventil, Motortreiber
- Einlesen der Signale von Fototransistoren für Bestimmung der Position der Äpfel via Lichtschranken Prinzip
- Einlesen der Signale zweier Taster (für Programm Reset o.Ä.)
- visuelle Anzeige von Informationen durch zwei farbige LEDs
- Spannungswandlung von 24V auf 5V und 3,3V für einheitlichen industrietypischen Spannungsbetrieb
- platzsparende Unterbringung des XMC  $\mu$ C direkt auf dem Board, steckbar.
- Universelles Design: alle internen Anschlüsse des  $\mu$ C auch über Steckerleiste erreichbar, HIGH Signal wahlweise 3.3V oder 5V -> falls statt XMC ein Raspberry Pi verwendet werden sollte

### Die wichtigsten Komponenten der Elektrotechnik

#### Lichtschranke

Um an zwei Stellen innerhalb der Maschine die Position der Äpfel zu messen, wurden Lichtschranken nach dem Sender/ Empfänger Prinzip realisiert, die aus folgenden Komponenten bestehen:

- Lasermodul 650nm 0.95mW TRU COMPONENTS
- Fotodiode BPW 21 820nm 55° OSRAM

Wie in Abb. 2 zu sehen, schaltet die Fotodiode bei auftreffendem roten Laserlicht mit 650nm Wellenlänge noch zu 90% durch. Im Vergleich zu alternativen IR-Emittoren kann das menschliche Auge bei unserer gewählten Lösung das Licht des Lasers erkennen, was ein einstellen der Lichtschranken erleichtert. Des Weiteren bietet ein Laser durch seine Lichtbündelung eine höhere Reichweite, sodass Emittor und Sensor weiter entfernt voneinander angebracht werden können.

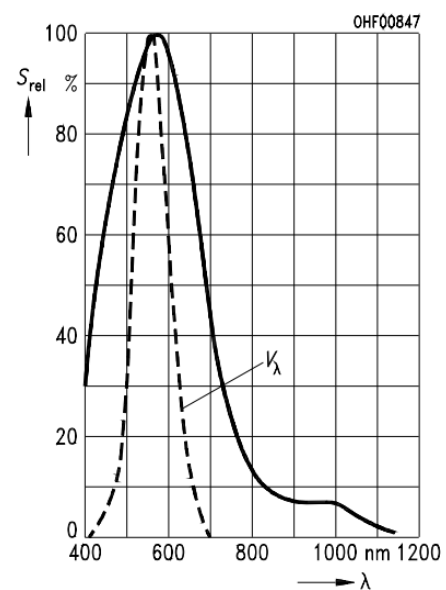


Abbildung 2 Relative Spektrale Empfindlichkeit Fotodiode BPW 21

#### IRFZ 48N MOSFET

Um den  $\mu$ C durch hohe Output Ströme für z.B. Relais nicht unnötig zu belasten, werden 6 MOSFETS angesteuert, die mit dem 5/ 24V Spannungsnetz die jeweiligen Bauteile steuern. Der großzügig dimensionierte und einheitlich verwendete IRFZ 48N MOSFET schaltet bereits mit 5V XMC Output Spannung sehr gut durch (siehe Abb. 4) und kann große Ströme bei geringer Drain Source Voltage effizient schalten (siehe Abb. 4).

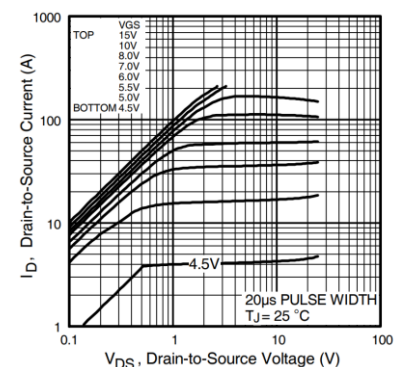


Abbildung 3 aus Datasheet IRFZ 48N

#### G5LE-1 5DC Relais

Um das Förderband oder die LED Stripes digital anzusteuern, benötigt es galvanisch getrennte Schalter also Relais. Die galvanische Trennung verhindert Störsignale auf unserer Platine und garantiert einen sicheren Betrieb, da beispielsweise die Steuerungselektronik des Förderbands bei 230V Spannung schaltet und eigentlich für den Betrieb mit mechanischem Kippschalter vorgesehen ist.

Die Relais, mit denen der mechanische Kippschalter des Förderbands digitalisiert, befinden sich aber aus Gründen der Sicherheit auf einer zweiten, externen Platine. Das auf der Hauptplatine angebrachte Relais dient dem An/Aus-Schalten der LED Stripes.

Die verwendeten G5LE-1 5DC Wechsler Relais zeichnen sich durch kompakte Bauform, geringer Spulendauerleistung [400mW], 5V DC Betriebsspannung und zugelassenen 250V AC Schaltspannung aus.

#### *XMC1100*

Der XMC1100 Mikrocontroller wurde aufgrund seiner ausgewogenen Eigenschaften zur Steuerung von Abläufen innerhalb der Maschine gewählt. Die integrierten Peripherieeinheiten und Low-Power-Funktionen sind ideale Eigenschaften für die Steuerung von Motoren und Sensoren. Zudem erleichtern umfassende Entwicklungstools, wie zum Beispiel der Debugger die Programmierung und Integration in unser System.

#### *Sypro 8MP USB Kamera.*

Für gute Bilder der zu scannenden Objekte wurde eine Kamera mit manuellem Fokus, manueller Blende und Zoomobjektiv gesucht. Die verwendete Kamera besitzt alle Eigenschaften und lässt sich dadurch optimal auf die konstante Lichtsituation und die ähnlichen Objekte einstellen.

#### *Cytron 10A 5-30V Zweikanal-DC-Motortreiber*

Um den DC Motor des Schiebers zu steuern, braucht es einen entsprechend dimensionierten DC-Motortreiber. Der hier genutzte Controller ist für Motoren von 5-30V Betriebsspannung dimensioniert, kann entsprechend hohe Ströme von 10 A Dauer- und 30 A Spitze steuern, verfügt über gleich zwei separate Motor Kanäle und lässt sich pro Kanal via PWM Signal über nur einen Pin steuern.

## **4.5.2 Mechanik**

### **4.5.2.1 Apfel Vereinzeler**

Um ein ganzes Magazin von Äpfeln automatisch sortieren zu lassen, wurde ein Vereinzeler entwickelt, der das händische Auflegen einzelner Äpfel überflüssig macht und stattdessen automatisch immer nur einen Apfel einzeln auf das Förderband legt. Damit zeigt der Vereinzeler exemplarisch am konkreten Beispiel Apfel, welches Grad an Automatisierung in dem OptiSort steckt und wie die individuellen Einzellösungen für konkrete Sortieraufgaben realisiert werden können.

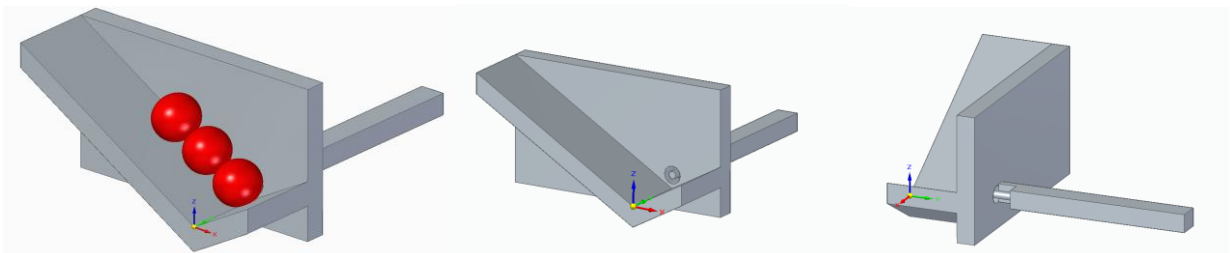


Abbildung 4: Skizze mit Äpfeln

Abbildung 5: Skizze Vorderseite

Abbildung 6: Skizze Rückseite

Wie in Abbildung 4 zu sehen, müssen für das Vereinzeln die Äpfel in eine Art Magazin hintereinander gebracht werden. Die Äpfel rollen nun immer an den untersten Punkt in der Rampe also auf einen einheitlichen Tiefpunkt unabhängig ihrer Größe. Dort in der Rückwand ist ein Zylinder eingelassen ist (siehe Abb. 6), der nun punktuell immer den untersten Apfel nach vorne auf das Förderband schiebt. Fährt der Zylinder nach erfolgreichem Vereinzeln wieder in seine Grundposition, rollt sofort der nächste Apfel durch die schiefe Ebene nach.

#### 4.5.2.2 Foto Station

Um mindestens 5/6 der Oberfläche des zu scannenden Objekts zu sehen, stellen wir uns einen klassischen Würfel vor und richten je eine Kamera auf die fünf zu sehenden Stirnseiten.

Eine Kamera wird für eine Draufsicht von oben über das zu scannende Objekt montiert (Halter Typ 1). Vier weitere Kameras befinden sich horizontal rund um die Scanposition (Halter Typ 2). Für alle Kameras wurden eigene Halterungen in SolidEdge entworfen und anschließend via 3D Druck Verfahren schulintern gefertigt (siehe Abb. 7-10). Typ 2 der Halterungen hängt in den vier Ecken einer quadratischen Aluminiumplatte über der Scanstation und bieten den Kameras durch eine 45° Drehung den Blick auf das Zentrum.

- Das Design ist gleichzeitig stabil und materialsparend.
- Die Kameras werden durch zwei Gewindeschrauben gehalten und sind dadurch neigbar.
- Die Halterung ist mit vier Schrauben mit der oberen Platte verbunden.

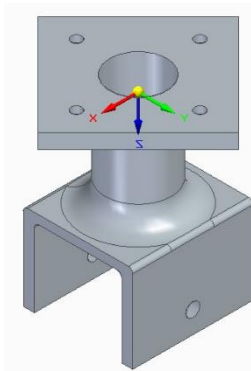


Abbildung 7: Typ 2

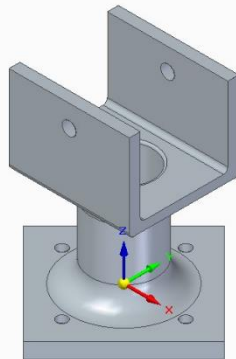


Abbildung 8: Typ 2

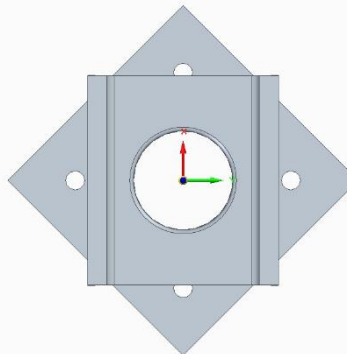


Abbildung 9: Typ 2

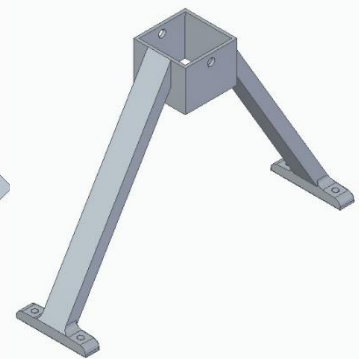


Abbildung 10: Typ 1

#### 4.5.2.3 Schieber

Damit der Weg der zu sortierenden Objekte sich je nach Urteil der KI auch verändern kann, wurde ein Schieber entwickelt, der die auszusortierenden Objekte an einer Stelle nach der Scanstation vom Förderband schiebt. Damit ergeben sich zwei unterschiedliche Endpunkte für die zu sortierenden Objekte: einmal das Förderbandende und einmal die Position seitlich rechts vom Förderband.

Um die genaue Position zu ermitteln, wann das Objekt am Schieber ist, wird erneut eine Lichtschranke nach dem Sender/ Empfänger Prinzip verwendet. Da die Lichtschranke jedoch im 90° Winkel zur Förderband Laufrichtung misst, muss der Schieber im Winkel von 45° über die Diagonale schieben, weswegen sich der Hub grob um den Faktor  $\sqrt{2}$  verlängert (Pythagoras).

Der Schieber selbst besteht dabei aus einer radial gelagerten Zahnstange, einem DC Motor und einem induktiven Sensor zur Ermittlung der Position. Die Schaufel des Schiebers kommt aus dem 3D Drucker.

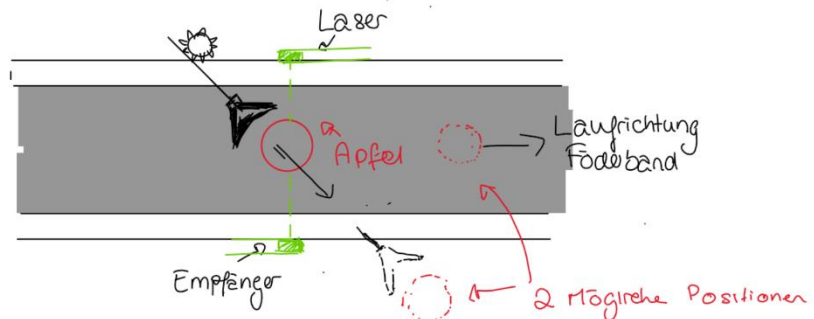


Abbildung 11: Skizze Draufsicht Förderband

#### 4.5.2.4 Verkleidung/ Blenden

Um die Maschine für den Transport zu diversen Messen robuster zu bauen, wurden Blenden entworfen. Die gewünschten Aluminium-Platten in 8mm Stärke können jedoch nicht schulintern von uns gefertigt werden, also wurde eine Kooperation mit der Firma Trumpf gestartet. Die benötigten Bleche wurden erneut in CAD designend und Trumpf für die Fertigung überreicht.

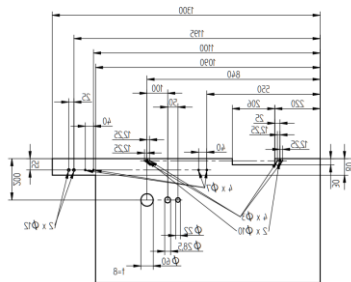


Abbildung 12: Rückseite

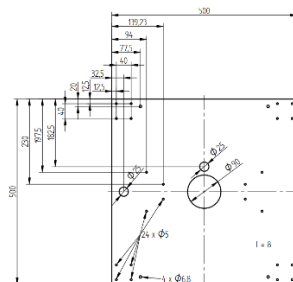


Abbildung 13: Kamera Mount Platte

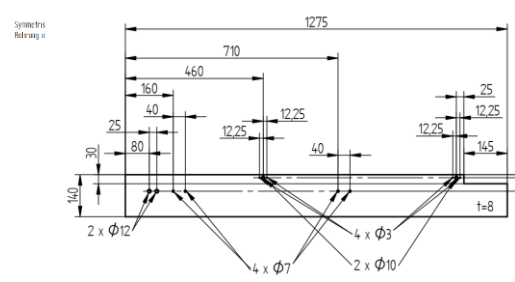


Abbildung 14: Stirnseite

## 4.6 Software

### 4.6.1 Einleitung

Das Projekt nutzt Software, die auf künstlicher Intelligenz (KI) und maschinellem Lernen basiert, um Objekte automatisch zu sortieren. Diese Software, die mit dem Flask-Framework entwickelt wurde, stellt das Herzstück der Maschine dar und koordiniert alle ihre Funktionen. Sie ist flexibel und anpassbar, was bedeutet, dass sie nicht nur für das Sortieren von Äpfeln, sondern für eine Vielzahl von Objekten eingesetzt werden kann.

Im Zentrum der Software steht TensorFlow Lite, eine für ressourcenbeschränkte Geräte optimierte Version von TensorFlow. Diese Technologie ermöglicht es der Software, Bilder, die von mehreren Kameras aufgenommen werden, effizient zu verarbeiten. Sobald ein Objekt, wie ein Apfel, auf das Förderband gelegt wird, nehmen die Kameras Bilder aus verschiedenen Winkeln auf. Diese Bilder werden dann von TensorFlow Lite analysiert. Das neuronale Netzwerk, das Teil von TensorFlow Lite ist, wurde so trainiert, dass es Merkmale wie Farbe, Form und Zustand der Objekte erkennen und interpretieren kann.

Nach der Bildanalyse trifft die KI eine Entscheidung darüber, ob das Objekt den festgelegten Kriterien entspricht. Diese Entscheidung wird dann an einen Mikrocontroller weitergegeben, der die mechanischen Komponenten der Maschine steuert. Der Mikrocontroller, ein wichtiger Bestandteil des Systems, ist dafür verantwortlich, die Bewegungen des Förderbands und des Schiebers zu koordinieren, sodass die Objekte entsprechend der Klassifizierung durch die KI sortiert werden.

Die Ergebnisse der KI-Analyse und die durchgeführten Aktionen werden auf einer Weboberfläche angezeigt, die mit HTML und JavaScript erstellt wurde. Diese Benutzeroberfläche bietet eine interaktive und benutzerfreundliche Erfahrung, indem sie Echtzeitinformationen über den Sortierprozess bereitstellt. Nutzer können die Klassifizierungsergebnisse überwachen und bei Bedarf Anpassungen vornehmen.

Ein besonders beeindruckendes Merkmal der Software ist ihre Trainierbarkeit. Nutzer können die KI mit eigenen Bildern trainieren, wodurch die Maschine für unterschiedliche Sortieraufgaben optimiert wird. Dieser Prozess wird durch das Hochladen neuer Bilder und das Trainieren des neuronalen Netzwerks mit diesen Bildern ermöglicht. So kann die Maschine kontinuierlich verbessert und für verschiedene Anwendungen angepasst werden.

## 4.6.2 Software-Probleme und Lösungen

### **Problem: Langsame Bildaufnahme**

In der ursprünglichen Konfiguration des Projekts gab es ein Problem bei der Aufnahme von Bildern: Es war zu langsam. Dieses Problem trat auf, weil die Kameraaufnahmen sequenziell, das heißt eine nach der anderen, durchgeführt wurden. In einem System, das mehrere Kameras gleichzeitig verwendet, führt dieser Ansatz zu erheblichen Verzögerungen. Jede Kamera musste warten, bis die vorherige ihre Aufnahme abgeschlossen hatte, bevor sie mit ihrer eigenen Aufnahme beginnen konnte.

### **Lösung: Multithreading**

Die Lösung für dieses Problem fanden wir im Multithreading, einer Programmiermethode, die es ermöglicht, mehrere Prozesse gleichzeitig auszuführen. Anstatt die Kameras nacheinander zu aktivieren, ermöglicht Multithreading das gleichzeitige Aktivieren und Betreiben aller Kameras. Somit sind wir von einer Scannzeit von ca. 10 Sekunden auf ca. 0,5 Sekunden gekommen.

### **Problem: Kommunikationsherausforderungen**

Ein anderes Problem in der ursprünglichen Konfiguration des Projekts war die Kommunikation zwischen dem Mikrocontroller und dem Computer. Beide Geräte sind über eine serielle Schnittstelle verbunden, was eine direkte und zeitnahe Kommunikation erfordert. Die Herausforderung bestand darin, eine effiziente und zuverlässige Kommunikationsmethode zu implementieren, bei der beide Geräte immer dann bereit sind, Nachrichten zu empfangen und zu senden, wenn es das Gegenüber erfordert.

### **Lösung: Koordinierte Kommunikationsprotokolle**

Die Lösung für dieses Problem wurde durch die Einführung koordinierter Kommunikationsprotokolle erreicht. Diese Protokolle regeln, wann und wie die Geräte miteinander kommunizieren.

Umsetzung im System:

- **Mikrocontroller als Initiator:** Der Mikrocontroller spielt die Rolle des Initiators in der Kommunikation. Er sendet ein Signal an den Computer, sobald ein Objekt für den Scan bereit ist. Dieses Signal dient als Auslöser für den Beginn des Scanvorgangs.
- **Warten auf Antwort:** Nach dem Senden des Signals wartet der Mikrocontroller auf eine Antwort des Computers. Diese Wartezeit ist entscheidend, um sicherzustellen, dass der Computer genügend Zeit hat, die Bilder zu verarbeiten und eine Entscheidung über das Objekt zu treffen.
- **Computer in ständiger Bereitschaft:** Der Computer befindet sich in einem ständigen Wartezustand, um das Signal des Mikrocontrollers zu empfangen. Sobald dieses Signal empfangen wird, beginnt der Computer sofort mit der Bildaufnahme und -verarbeitung.
- **Rückmeldung und Koordination:** Nachdem der Computer die Bilder verarbeitet und eine Entscheidung getroffen hat, sendet er eine Rückmeldung an den Mikrocontroller. Diese Rückmeldung enthält Informationen darüber, was mit dem Objekt geschehen soll (z.B. ob es aussortiert werden soll). Der Mikrocontroller empfängt diese Information und führt die entsprechenden Aktionen aus.

### Problem: Organisation und Training der KI

Ein weiteres Problem betraf das Training der Künstlichen Intelligenz (KI). Die Herausforderung bestand darin, eine große Anzahl von Bildern für das Training der KI hochzuladen und dabei den Überblick über die verschiedenen Klassen und Bilder zu behalten. Dies war wichtig, um sicherzustellen, dass die KI korrekt und effektiv trainiert wird.

### Lösungsansatz: Web-basierte Bildupload- und Trainingsprozesse

Die Lösung für dieses Problem wurde durch die Entwicklung eines webbasierten Systems realisiert, das es Nutzern ermöglicht, Bilder einfach hochzuladen und die KI im Hintergrund zu trainieren.

#### Umsetzung im System

- **Webbasierte Uploads:** Die Nutzer laden Bilder über eine Weboberfläche hoch. Diese Bilder werden entsprechend ihren Klassen organisiert, um dem System zu helfen, zwischen verschiedenen Objektkategorien zu unterscheiden.
- **Hintergrundtraining:** Nach dem Hochladen der Bilder wird das Training der KI im Hintergrund initiiert. Dieser Prozess verwendet die hochgeladenen Bilder, um das KI-Modell zu trainieren, sodass es effektiv zwischen verschiedenen Klassen unterscheiden kann.
- **Erstellung einer Zip-Datei:** Nach Abschluss des Trainings wird automatisch eine Zip-Datei erstellt. Diese Datei enthält sowohl die Namen der verschiedenen Klassen als auch das trainierte KI-Modell.
- **Wiederverwendung des KI-Modells:** Wenn Nutzer das selbst trainierte Modell verwenden möchten, muss lediglich die erstellte Zip-Datei hochladen. Das System extrahiert dann die notwendigen Informationen und das Modell aus dieser Datei und setzt es für den Sortiervorgang ein.

### 4.6.3 Begriffsdefinitionen

- **Flask-Framework:** Flask ist ein sogenanntes Webframework, das in der Programmiersprache Python geschrieben ist. Ein Webframework ist eine Sammlung von Werkzeugen und Bibliotheken, die das

Entwickeln von Webanwendungen vereinfachen. Flask ist besonders für seine Einfachheit und Flexibilität bekannt, was es Entwicklern ermöglicht, schnell und effizient Webseiten und Online-Anwendungen zu erstellen.

- **TensorFlow Lite:** TensorFlow ist eine Bibliothek für maschinelles Lernen und KI. TensorFlow Lite ist eine leichtere Version davon, die speziell für leistungsschwächere Geräte, wie Mobiltelefone oder Mikrocontroller, entwickelt wurde. Sie ermöglicht es, KI-Modelle zu nutzen, um Muster in Daten zu erkennen und Entscheidungen zu treffen – in diesem Fall, um Bilder zu analysieren und zu klassifizieren.
- **Neuronales Netzwerk:** Ein neuronales Netzwerk ist ein Algorithmus, der nach dem Vorbild des menschlichen Gehirns konzipiert ist. Es besteht aus einer Reihe von Knoten (Neuronen), die miteinander verbunden sind und Daten verarbeiten können. Durch Training lernt das Netzwerk, Muster in den Daten zu erkennen und spezifische Aufgaben auszuführen, wie z.B. das Erkennen von Objekten auf Bildern.
- **HTML und JavaScript:** HTML (Hypertext Markup Language) und JavaScript sind Technologien, die zur Erstellung von Webseiten verwendet werden. HTML wird für die Struktur und das Layout der Seite verwendet, während JavaScript für interaktive Elemente zuständig ist. Mit JavaScript können Webseiten auf Nutzereingaben reagieren, ohne neu geladen werden zu müssen.

Zur Software [hier klicken](https://github.com/2Friendly4You/OptiSort) [https://github.com/2Friendly4You/OptiSort]

## 5 Fazit

Wir haben eine vollautomatisierte Sortiermaschine entwickelt, die mit 95% Wahrscheinlichkeit die Äpfel richtig kategorisiert und durch ihre Weboberfläche so Bedienungsfreundlich geworden ist, dass auch Laien ganz einfach die Maschine auf neue Sortieraufgaben umtrainieren und anschließend verwenden können. Damit haben wir sogar mehr als unser ursprüngliches Ziel erreicht, da wir die Maschine nun auch universell gestaltet haben. Weiter verlaufen die automatisierten Prozesse innerhalb der Maschine so verlässlich und präzise, dass wir Potential in der Industrie sehen. Die dafür nötigen mechanischen und softwareseitigen Innovationen wie der Einzeler und der Sortierer stammen ebenso wie die selbstgeschriebene KI alle von uns. Aktuell benötigt das Scannen circa eine Sekunde, der Transport 10 Sekunden, wodurch wir bei größerem Magazin circa 200 Äpfel pro Stunde überprüfen können.

## 6 Ausblick

### 6.1 Potentielle Anwendungen

Überall dort, wo bisher noch Menschen optisch Dinge (aus)sortieren, gibt es eine Anwendung für den OptiSort in seiner bisherigen oder einer abgewandelten Form. Damit ermöglichen wir das Beschleunigen von Prozessen und eine Kostenminimierung bei teuren Arbeitskräften. Gleichzeitig erlaubt unsere benutzerfreundliche Web-Oberfläche bereits Laien, die KI auf neue individuelle Sortieraufgabe einzulernen.

Unser OptiSort kann bei Bauern wie im privaten Bereich jetzt schon die eingelagerten Äpfel überprüfen, ohne dass eine teure Arbeitskraft anwesend sein muss.



## 6.2 Verbesserungsmöglichkeiten

Der OptiSort lässt sich noch in vielerlei Hinsicht verbessern und erweitern. Einige dieser Bereiche sind:

- $\mu$ C und Computer durch einen gemeinsamen RaspberryPi ersetzen, um Gesamtkosten zu senken
- Komplettes Motherboard/ Haupt-Platine erneut ohne Lötbrücken bzw. Fehlerfrei fertigen lassen
- Zu scannendes Objekt nach 1. Scan rotieren und erneut scannen, um die relative Blindheit des Scanners zu minimieren (bisher sieht der Scanner nur 90% der Oberfläche) Vereinzeler komplett aus lebensmittelnutralem Edelstahl oder Aluminium für hygienischere Anwendung in Lebensmittelindustrie fertigen
- Weitere Vereinzeler Konstruktionen für neue Anwendungen fertigen

## 7 Ergebnisse

Durch das Projekt haben wir gelernt, Verantwortung zu übernehmen, strukturiert und zielstrebig zu arbeiten, uns Wissen anzueignen und mit Problemen konstruktiv umzugehen. Besonders die folgenden Erkenntnisse haben wir aus unserem Projekt mitgenommen:

- Die Elektrotechnik als auch die Informatik ist komplex: umso wichtiger ist es, konstruktiv mit Problemen umzugehen, wenn die neu gefertigte Platine doch nicht auf Anhieb funktioniert. Oft sind es Logikfehler im Schaltplan, manchmal ist es aber auch wirklich ein Bauteil defekt. Die Fehlersuche fordert dabei jedes Mal aufs Neue Ausdauer doch endete immer lehrreich! Umso schöner ist dann aber das Gefühl, wenn endlich alles funktioniert.
- Ein komplexes Projekt schafft man nicht allein. Doch das war auch nie das Ziel. Wichtig ist also kontinuierlich den Austausch im Team zu halten und durch effiziente Arbeitsteilung die Stärke des Einzelnen optimal zu nutzen. Dadurch kann auch an mehreren Stellen am Projekt gleichzeitig gearbeitet werden.
- Die in der Aufgabenstellung gesetzten Ziele wurden erfolgreich erreicht, umgesetzt und teilweise übertroffen. Die eigens entwickelte Elektronik und Software wurde in das Förderband erfolgreich integriert und mit weiterer eigens konstruierten Mechanik kombiniert. Während des Entstehungsprozess flossen auch viele weitere Ideen mit ein, die den OptiSort verbesserten. Für eine vermarkungsfähige Version sehen wir genügend Potential.



## 8 Danksagung

Wir möchten uns zunächst einmal bei den Lesern dieser Dokumentation für Ihr Interesse an diesem Projekt bedanken.

Vielen Dank an die Mikro Makro Mint Stiftung des Landes Baden-Württemberg, die durch ihre finanzielle Förderung das Projekt erst ermöglicht hat. Danke an die Firma Trumpf Ditzingen, die uns im Fertigen der Seitenverkleidungen großzügig unterstützt hat.

Ebenso gilt unser Dank Herr Zipfel, der uns mit seinem Sammelsurium an Teilen und guten Ratschlägen stets zur Seite stand. Auch Herr Dr. Kleckner unterstütze uns bei Hardware Fragen immer und übernahm schließlich die Fertigung der Schieberkonstruktion aus Metall nachdem unser 3D gedruckte Mechanismus zu viel Reibung aufwies. In der Organisation des gesamten Projekts und in der Formulierung von Texten hatten wir die Hilfe von Herr Dr. Hummel, wofür wir uns recht herzlich bedanken! Und zuletzt bedanken wir uns bei unserem Klassenlehrer Dipl. Ing. Dominique Haas für seine Unterstützung bei den Bestellungen aller Teile und der Hilfestellung bei Problemen in der Elektrotechnik.

## 9 Literaturverzeichnis

- [1] R. S. Joachim Steinwendner, Neuronale Netze programmieren mit Python, Bonn: Rheinwerk Computing, 2020.
- [2] P. K. Johannes Ernesti, Python 3: Das umfassende Handbuch, Bonn: Rheinwerk Computing, 2020.
- [3] P. Perrotta, Machine Learning für Softwareentwickler, Heidelberg: dpunkt.verlag, 2020.
- [4] Tensorflow, „Transfer learning and fine-tuning,“ [Online]. Available: [https://www.tensorflow.org/tutorials/images/transfer\\_learning](https://www.tensorflow.org/tutorials/images/transfer_learning). [Zugriff am 13.1.2024].