

Technische Hochschule Deggendorf
Faculty Applied Information Technology
Studiengang Bachelor of Cyber-Security

Projekt "WeedDetector"

Projektübergabe

Vorgelegt von:

Christof Renner (22301943)
Manuel Friedl (1236626)

Prüfungsleitung:

Prof. Dr. Holger Jehle

Datum: 05. Juli 2025

Contents

1	Einführung und Projektübersicht	4
1.1	Motivation und Zielsetzung	4
1.2	Systemarchitektur	4
1.3	Technologiestack	4
2	Installation und Konfiguration	6
2.1	Systemvoraussetzungen	6
2.2	Installationsanleitung	6
2.2.1	Installation mittels Docker	6
2.2.2	Manuelle Installation (ohne Docker)	7
2.3	Konfiguration und Anpassung	7
2.3.1	Modellkonfiguration	7
2.3.2	Umgebungsvariablen	7
3	Benutzerhandbuch	8
3.1	Benutzeroberfläche	8
3.1.1	Hauptfunktionen	8
3.2	Grundlegende Bedienung	9
3.2.1	Einzelbildanalyse	9
3.2.2	Live-Kameraerkennung	9
3.2.3	Robotersteuerung	9
3.3	Erweiterte Funktionen	10
3.3.1	Anpassung der Erkennungsparameter	10
3.3.2	Interpretation der Erkennungsergebnisse	10
4	Modelltraining und -anpassung	11
4.1	Vorbereitung des Trainingsdatensatzes	11
4.1.1	Datenannotation	11
4.1.2	Datenaugmentierung	11
4.2	Durchführung des Trainings	11
4.2.1	Training starten	11
4.3	Modellbewertung und -optimierung	12
4.3.1	Bewertungsmetriken	12
4.3.2	Feinabstimmung des Modells	12
4.4	Fehlerbehandlung und Logging	12
5	Wartung und Weiterentwicklung	13
5.1	Bekannte Einschränkungen	13
5.2	Erweiterungsmöglichkeiten	13
5.3	Wartungsrichtlinien	13

6	Fazit und Ausblick	15
6.1	Projektergebnisse	15
6.2	Zukünftige Forschungsrichtungen	15
6.3	Abschließende Bewertung	15

1 Einführung und Projektübersicht

1.1 Motivation und Zielsetzung

Der **WeedDetector** ist ein innovatives Softwaresystem, das entwickelt wurde, um in landwirtschaftlichen Umgebungen automatisiert Unkräuter zu erkennen und zu klassifizieren. Die Hauptmotivation hinter diesem Projekt liegt in der zunehmenden Notwendigkeit nachhaltiger landwirtschaftlicher Praktiken und der Reduzierung des Einsatzes von Herbiziden.

Dieses Projekt entstand aus folgenden konkreten Anforderungen:

- **Präzisionslandwirtschaft:** Durch die genaue Identifikation von Unkraut können Landwirte gezielt eingreifen, anstatt flächendeckend Herbizide einzusetzen.
- **Umweltschutz:** Reduzierung des Chemikalieneinsatzes durch punktgenaue Behandlung von Unkrautflächen.
- **Automatisierung:** Integration mit Robotersystemen zur vollautomatischen Unkrautbekämpfung.
- **Kosteneinsparung:** Verringerung des Ressourcenverbrauchs und effizientere Arbeitsabläufe.

1.2 Systemarchitektur

Der WeedDetector folgt dem Model-View-Controller (MVC) Architekturmuster, um eine saubere Trennung der Verantwortlichkeiten zu gewährleisten und die Wartbarkeit zu verbessern.

1.3 Technologiestack

Der WeedDetector nutzt einen modernen Technologiestack, der für Bildverarbeitung und maschinelles Lernen optimiert ist:

- **Programmiersprache:** Python 3.13 als Hauptsprache für die gesamte Anwendung
- **Computer Vision:**
 - OpenCV 4.11.0.86 für Bildverarbeitung und Kameraintegration
 - Ultralytics YOLOv8 8.3.155 für Objekterkennung und -klassifikation
- **Benutzeroberfläche:**
 - Tkinter für die grafische Benutzeroberfläche
 - Pillow 11.2.1 für erweiterte Bildverarbeitung in der GUI

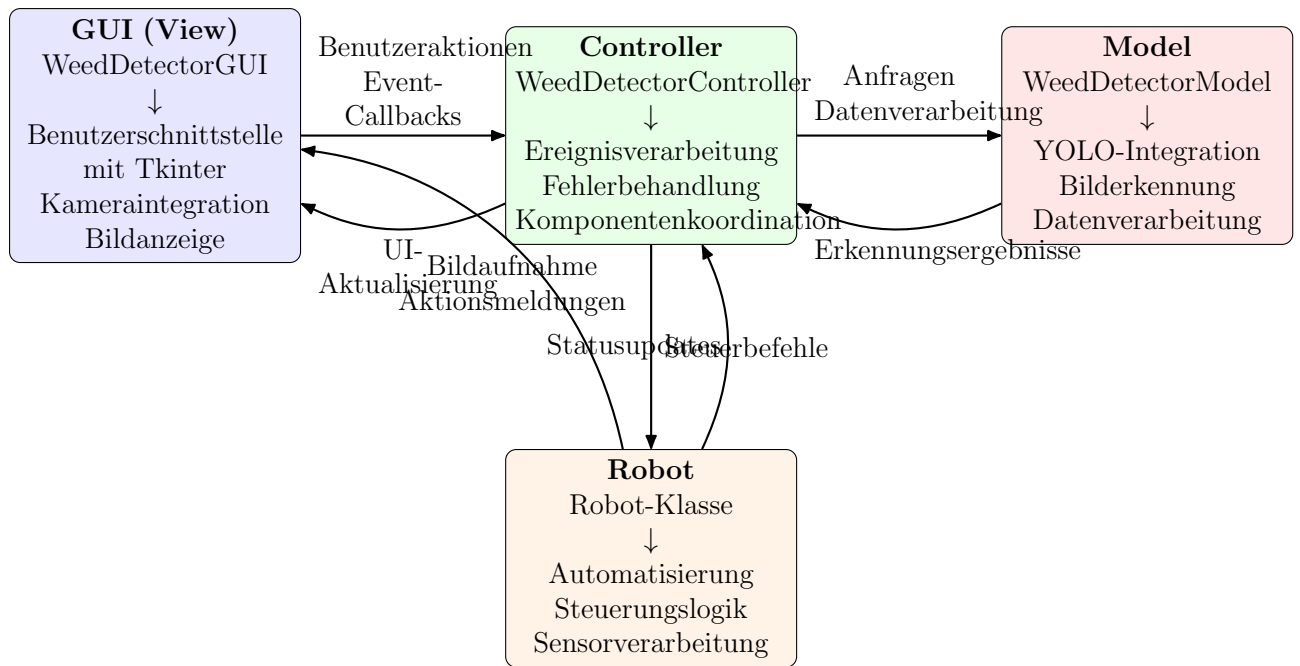


Figure 1.1: Detaillierte Architektur des WeedDetector-Systems

- **Containerisierung:** Docker für einheitliche Entwicklungs- und Produktionsumgebungen
- **Kontinuierliche Integration:** GitHub Actions für automatisierte Tests und Builds

2 Installation und Konfiguration

2.1 Systemvoraussetzungen

Für den erfolgreichen Betrieb des WeedDetector-Systems werden folgende Mindestsystemvoraussetzungen empfohlen:

- **Hardware:**
 - CPU: Quad-Core Prozessor, 2.5 GHz oder höher
 - RAM: Mindestens 8 GB (16 GB empfohlen)
 - Grafikkarte: CUDA-fähige GPU mit mindestens 4 GB VRAM für optimale Leistung
 - Festplattenspeicher: Mindestens 2 GB freier Speicherplatz
 - Kamera: USB-Webcam oder integrierte Kamera für Live-Erkennung
- **Software:**
 - Betriebssystem: Linux (Ubuntu 22.04 LTS empfohlen), Windows 10/11 oder macOS
 - Docker: Version 24.0 oder höher
 - X11-Server (für GUI-Anzeige aus dem Container)

2.2 Installationsanleitung

Die Installation des WeedDetector-Systems erfolgt primär über Docker, um eine konsistente Umgebung zu gewährleisten und Abhängigkeitsprobleme zu vermeiden.

2.2.1 Installation mittels Docker

1. **Docker installieren:** Falls noch nicht geschehen, installieren Sie Docker gemäß der offiziellen Anleitung für Ihr Betriebssystem (<https://docs.docker.com/get-docker/>).

2. **Repository klonen:**

```
1 git clone https://github.com/SS25-SoftwareEngineering-WeedDetector.git
2 cd SS25-SoftwareEngineering-WeedDetector
```

3. **Docker-Image erstellen:**

```
1 docker build -t weed-detector .
```

4. **Anwendung starten:**

```
1 docker run --device /dev/video0:/dev/video0 --net=host -e  
    DISPLAY=$DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix weed-  
    detector
```

2.2.2 Manuelle Installation (ohne Docker)

Für Entwicklungs- oder Testzwecke kann das System auch manuell installiert werden:

1. Python-Umgebung einrichten:

```
1 python -m venv venv  
2 source venv/bin/activate # Unter Windows: venv\Scripts\  
    activate
```

2. Abhängigkeiten installieren:

```
1 pip install -r requirements.txt
```

3. Anwendung starten:

```
1 python app/main.py
```

2.3 Konfiguration und Anpassung

Der WeedDetector bietet verschiedene Konfigurationsmöglichkeiten, um das System an spezifische Anforderungen anzupassen:

2.3.1 Modellkonfiguration

Die Standardkonfiguration verwendet ein vortrainiertes YOLOv8-Modell. Das System sucht automatisch nach trainierten Modellen in folgenden Pfaden:

- runs/detect_train/weights/best.pt
- data/weights/best.pt
- data/models/best.pt
- models/best.pt

Wenn kein benutzerdefiniertes Modell gefunden wird, wird auf das Standard-YOLOv8-Modell zurückgegriffen.

2.3.2 Umgebungsvariablen

Folgende Umgebungsvariablen können beim Start des Docker-Containers konfiguriert werden:

- QT_X11_NO_MITSHM=1: Verhindert Probleme mit X11-Shared-Memory
- QT_DEBUG_PLUGINS=1: Aktiviert Debug-Ausgaben für Qt-Plugins
- OPENCV_VIDEOIO_PRIORITY_V4L2=0: Priorität für Video4Linux2-Backend
- OPENCV_VIDEOIO_DEBUG=1: Aktiviert Debug-Informationen für OpenCV Video-I/O

3 Benutzerhandbuch

3.1 Benutzeroberfläche

Die Benutzeroberfläche des WeedDetector-Systems ist in mehrere funktionale Bereiche unterteilt:

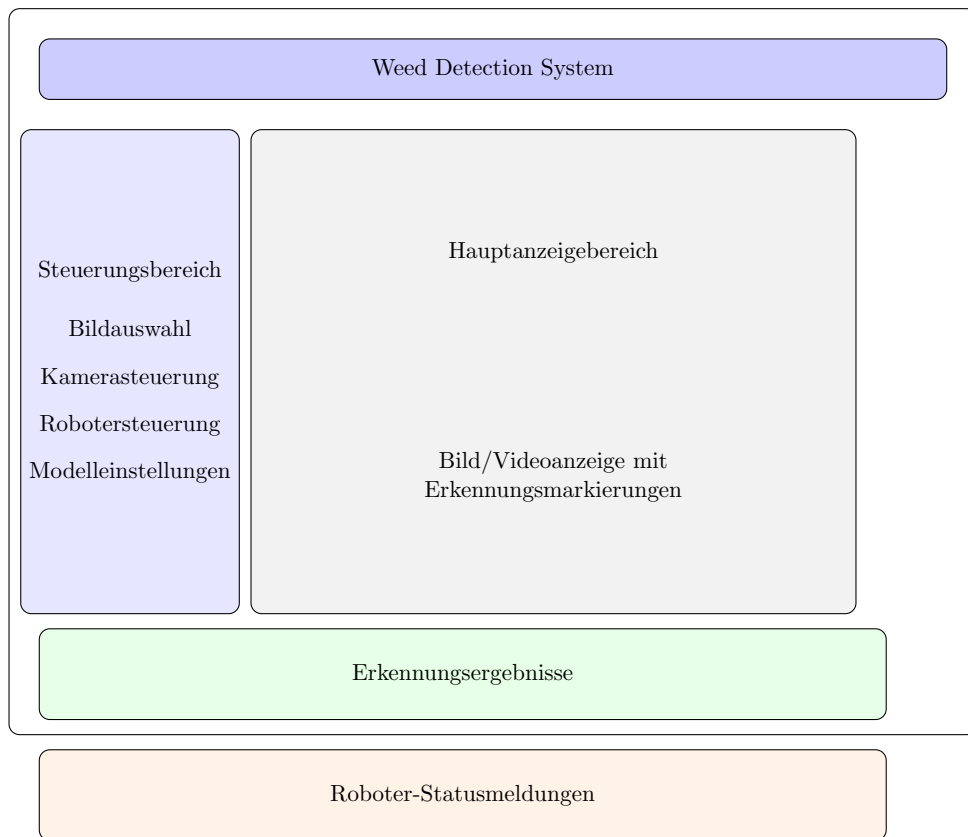


Figure 3.1: Benutzeroberfläche des WeedDetector-Systems

3.1.1 Hauptfunktionen

Die Benutzeroberfläche bietet folgende Hauptfunktionen:

1. **Bildauswahl:** Ermöglicht das Laden eines Bildes aus dem Dateisystem zur Analyse.
2. **Kamerasteuerung:** Aktiviert die Echtzeit-Bilderfassung über die angeschlossene Kamera.
3. **Robotersteuerung:** Startet und stoppt die automatisierte Unkrauterkennerung und -behandlung.

4. **Empfindlichkeitseinstellung:** Ermöglicht die Anpassung des Konfidenzwertes für die Erkennung (0,05-0,95).
5. **Erkennungsanzeige:** Zeigt erkannte Unkräuter mit Begrenzungsrahmen, Mittelpunktmarkierungen und Beschriftungen an.
6. **Ergebnisprotokoll:** Listet detaillierte Informationen zu erkannten Objekten, einschließlich Koordinaten und Konfidenzwerten.
7. **Roboter-Aktionsprotokoll:** Dokumentiert alle Aktionen und Statusmeldungen des Robotersystems.

3.2 Grundlegende Bedienung

3.2.1 Einzelbildanalyse

Um ein einzelnes Bild auf Unkraut zu analysieren:

1. Klicken Sie auf die Schaltfläche **”Select Image”**.
2. Wählen Sie eine Bilddatei aus dem Dateiauswahldialog.
3. Das System führt automatisch eine Erkennung durch und zeigt die Ergebnisse an.
4. Passen Sie bei Bedarf den Konfidenzwert an, um die Erkennungsempfindlichkeit zu ändern.

3.2.2 Live-Kameraerkennung

Für die Echtzeit-Unkrauterkenennung mit der Kamera:

1. Klicken Sie auf die Schaltfläche **”Start Camera”**.
2. Die Kameraansicht wird im Hauptanzeigebereich angezeigt.
3. Erkannte Unkräuter werden in Echtzeit markiert und im Ergebnisbereich aufgelistet.
4. Zum Beenden der Kameraerkennung klicken Sie auf **”Stop Camera”**.

3.2.3 Robotersteuerung

Zur Aktivierung des automatisierten Robotersmodus:

1. Stellen Sie sicher, dass die Kamera aktiviert ist oder aktivieren Sie sie.
2. Klicken Sie auf die Schaltfläche **”Start Robot”**.
3. Der Roboter beginnt mit der automatischen Unkrauterkenennung und -behandlung.
4. Alle Roboterhandlungen werden im Roboter-Statusbereich protokolliert.
5. Bei Erkennung von Unkraut stoppt der Roboter automatisch für eine genauere Analyse.
6. Zum manuellen Stoppen des Roboters klicken Sie auf **”Stop Robot”**.

3.3 Erweiterte Funktionen

3.3.1 Anpassung der Erkennungsparameter

Die Erkennungsgenauigkeit kann durch Anpassung des Konfidenzwertes optimiert werden:

- **Niedriger Konfidenzwert (0,05-0,25):** Erkennt mehr potenzielle Unkräuter, kann aber zu falschen Erkennungen führen.
- **Mittlerer Konfidenzwert (0,25-0,50):** Bietet eine ausgewogene Erkennungsrate für die meisten Anwendungen.
- **Hoher Konfidenzwert (0,50-0,95):** Reduziert Falscherkennungen, kann aber auch tatsächliche Unkräuter übersehen.

3.3.2 Interpretation der Erkennungsergebnisse

Die Erkennungsergebnisse werden in folgenden Formaten dargestellt:

- **Visuelle Markierungen:**
 - Rote Begrenzungsrahmen um erkannte Unkräuter
 - Grüne Kreuze an den Mittelpunkten der erkannten Objekte
 - Textbeschriftungen mit Klasse, Koordinaten und Konfidenzwert
- **Textbasierte Ausgabe:**
 - Nummerierte Liste erkannter Objekte
 - Klassenname jedes erkannten Objekts
 - X/Y-Koordinaten des Mittelpunkts
 - Konfidenzwert der Erkennung