**Softwareentwicklung SS2025**

**Von:** Brua Karim, Georg Settels, Vu Minh Hoang Nguyen

**Datum:** 23.06.2025

INHALT

[INHALT 2](#_Toc201605829)

[1. Allgemeines 3](#_Toc201605830)

[1.1 Projektbezug 3](#_Toc201605831)

[1.2 Story 3](#_Toc201605832)

[1.3 Aufgabe der Spielerinnen und Spieler 3](#_Toc201605833)

[1.4 Minispiele 3](#_Toc201605834)

[2. Benötigte Komponenten 4](#_Toc201605835)

[2.1 IR-Pistole (x2) 4](#_Toc201605836)

[2.2 IR-Receiver Board 4](#_Toc201605837)

[3. Verkabelung 5](#_Toc201605838)

[3.1 Software Interfaces 5](#_Toc201605839)

[3.1.1 Graphische Darstellung 5](#_Toc201605840)

[3.1.2 Graphisches Preprocessing 5](#_Toc201605841)

[3.1.3 Navigation 5](#_Toc201605842)

[3.2 Hardware Interfaces 5](#_Toc201605843)

[3.2.1 Minimale Anforderungen 5](#_Toc201605844)

[4. Installationsanleitung 6](#_Toc201605845)

[4.1 Performance Anforderungen 6](#_Toc201605846)

[4.2 Funktionen 6](#_Toc201605847)

[4.2.1 Datensatz einladen und darstellen 6](#_Toc201605848)

[ CT-Datensatz einladen in geteiltes Image Array 6](#_Toc201605849)

[4.2.2 Registrierung 6](#_Toc201605850)

[4.2.3 Planung der Strahlentherapie 6](#_Toc201605851)

[4.2.4 Speichern der Planung 6](#_Toc201605852)

[5. Ausführliche Spielanleitung 7](#_Toc201605853)

[6. Bekannte Probleme & Fixes 7](#_Toc201605854)

[6.1 Mitgeltende Unterlagen 7](#_Toc201605855)

[7. Supplementary materials 7](#_Toc201605856)

[7.1 3D-Modelle 7](#_Toc201605857)

# Allgemeines

## Projektbezug

Gitlab: <https://git.rwth-aachen.de/hoang.nguyen/multimodal-madness-gruppe5>

## Story

Das Trainingsprogramm der „Orcan Trooper“, die letzte Verteidigungslinie der Galaxis, hat ein neues Trainingssystem implementiert, um der stetig wachsenden Bedrohung des äußeren Sektors entgegenzuwirken.

Du bist einer der Auserwählten. Ein Rekrut unter vielen. Doch nur wenige werden es schaffen, die gnadenlose Abschlussprüfung zu bestehen.

Bist du bereit, die Prüfung zu bestehen und deinen Platz in den Reihen der legendären Orcan Trooper einzunehmen? Oder wirst du im Schatten der Galaxis untergehen?

Trotze dem Sturm nicht – werde Teil von ihm. Oder verliere dich in seinen Tiefen.

## Aufgabe der Spielerinnen und Spieler

Die Spielerinnen und Spieler sind in einem Trainingsprogramm und müssen gegen Mitspieler und gegen ihre eigenen Grenzen antreten. Durch Minispiele sollen körperliche und geistige Fähigkeiten gestärkt werden, um die Kadetten auf einen kommenden galaktischen Krieg vorzubereiten. Dazu erhalten bis zu zwei Spieler IR-Pistolen, mit denen man auf IR-Receiver-Boards schießen kann. Die anzuschießenden Boards werden mit LEDs markiert.

## Minispiele

* **Reaktionszeit:** (30s-60s)

x-faches kurzes Aufblinken der LED. Der Spieler muss so schnell wie es geht schießen. Der Score ist die Durchschnittszeit

* **Schnelligkeit:** (20s-30s)

In einer bestimmten Zeitspanne trifft man so viele Ziele wie möglich

* **Arbeitsgedächtnis:** (abhängig vom Spielergedächtnis)

Das Board zeigt eine Sequenz an Feldern an, die der Spieler nach dem Abspielen wiederaufrufen muss. Dazu schießt der Spieler in der selben Reihenfolge die Ziele ab

* **Stressbewältigung: (**30s-60s)

Die Ziele leuchten in zwei verschiedenen Farben auf, welche Feind und Freund signalisieren. Der Spieler muss in Echtzeit entscheiden, ob er auf die Gegner schießt, oder mit Heilpfeilen die Kameraden vorm Tod rettet.

# Benötigte Komponenten

## IR-Pistole (x2)

* IR-LED TSAL6200
* Arduino Nano 3.3V / 5V
* Button
* Piezo-Speaker
* Laser-Modul KY-008
* Vibrationsmotor
* USB-B zu USB 3.0 Kabel
* 3x M3 20mm Schrauben für Plastik
* 2x Transistoren 2N2222A
* Halbes Breadboard GND/VCC Rail
* Jumper Kabel
  + M-to-M :
  + F-to-M :
  + F-to-F :
* Widerstände
  + 3x 1,2kΩ

## IR-Receiver Board

* IR- Receiver TSOP4840
* Arduino Nano 3.3V / 5V
* Button
* Piezo-Speaker
* Laser-Modul KY-008
* Vibrationsmotor
* USB-B zu USB 3.0 Kabel
* 3x M3 20mm Schrauben für Plastik
* Jumper Kabel
  + M-to-M :
  + F-to-M :
  + F-to-F :
* Widerstände
  + 3x 1,2kΩ

# Verkabelung

## Software Interfaces

### Graphische Darstellung

* Qt 6.8
* **Widget**-Klasse für GUI
* QImage für Bilddarstellung

### Graphisches Preprocessing

* **CTDataset** Library zum Laden von Bilddaten, Tiefenbuffern, 3D Belichtungsrenderung

### Navigation

* QMouseEvent für Verwaltung der Mausaktionen
* QWheelEvent für Verwaltung der Mausradaktionen

## Hardware Interfaces

### Minimale Anforderungen

* 4GB RAM
* 2-Kern-CPU (für Multithreading)
* 100MB freier Speicherplatz

# Installationsanleitung

## Performance Anforderungen

* Min. 60 fps bei Ansicht der Bilddaten und Navigierung der Schichten  
  (<= 16,6 ms / Aktualisierung einer Schicht)
* Max. 0,5s für Laden des Tiefenbuffers mithilfe von Multithreading

## Funktionen

### Datensatz einladen und darstellen

### CT-Datensatz einladen in geteiltes Image Array

* Darstellung der Pixelwerte mithilfe von QImage
* Eingrenzung der Grauwerte mithilfe von internem Windowing der HU-Werte
* Durch Slider kann der Nutzer Zentrum, Breite und Segmentierungs-Threshold einstellen
* Durch Buttons kann der Nutzer die Sichtperspektive (Front, Back, Top, Bottom, Right, Left) auf das 3D-Modell einstellen

### Registrierung

* Aufnahme des Punktes der Hautoberfläche wird durch Mausklicken und Mausdragging des Nutzers auf der 2D/3D Ansicht registriert
* Der geklickte Punkt wird mithilfe eines farbigen Kreises in der Ansicht markiert
* Die Koordinaten des angeklickten Punktes werden auf der Ansicht angezeigt

### Planung der Strahlentherapie

* Definition des Zielbereiches durch Auswählen des Mittelpunktes mithilfe der Maus. Modifizieren der Größe des Zielbereiches durch Ziehen des Randes vom Kreis.
* Definition der No-Go-Area durch Auswählen des Mittelpunktes mithilfe der Maus.   
  Modifizieren der Größe des Zielbereiches durch Ziehen des Randes vom Kreis.
* Wechsel zwischen Bearbeitung von Zielbereich und No-Go-Area durch Button.
* Anzeige der Planung in 2D Ansicht mit grünem Kreis für Zielbereich und rotem Kreis für No-Go-Area
* Anzeige der Planung in 3D Ansicht mit grüner Sphäre für Zielbereich und rotrm Sphäre für No-Go-Area

### Speichern der Planung

* Koordinaten der Registrierung werden in txt file in der ersten Zeile gespeichert
* Koordinaten des Zielbereiches werden in txt file in der zweiten Zeile gespeichert
* Koordinaten der No-Go-Area werden in txt file in der dritten Zeile gespeichert

# Ausführliche Spielanleitung

Die Software wird am Ende des Semesters von jedem Studierenden als Prüfungsleistung abgegeben.

# Bekannte Probleme & Fixes

## Mitgeltende Unterlagen

* Foliensätze und Aufgabenzettel, die während des Semesters ausgegeben wurden

# Supplementary materials

## 3D-Modelle

IR-Gun-Shell: <https://git.rwth-aachen.de/hoang.nguyen/multimodal-madness-gruppe5/-/blob/main/project/media/3d%20Models/ir-gun.3mf?ref_type=heads>

IR-LED-Hülse: <https://git.rwth-aachen.de/hoang.nguyen/multimodal-madness-gruppe5/-/blob/main/project/media/3d%20Models/ir-gun-shaft.3mf?ref_type=heads>

IR-Gun-Trigger: <https://git.rwth-aachen.de/hoang.nguyen/multimodal-madness-gruppe5/-/blob/main/project/media/3d%20Models/ir_gun_trigger.3mf?ref_type=heads>