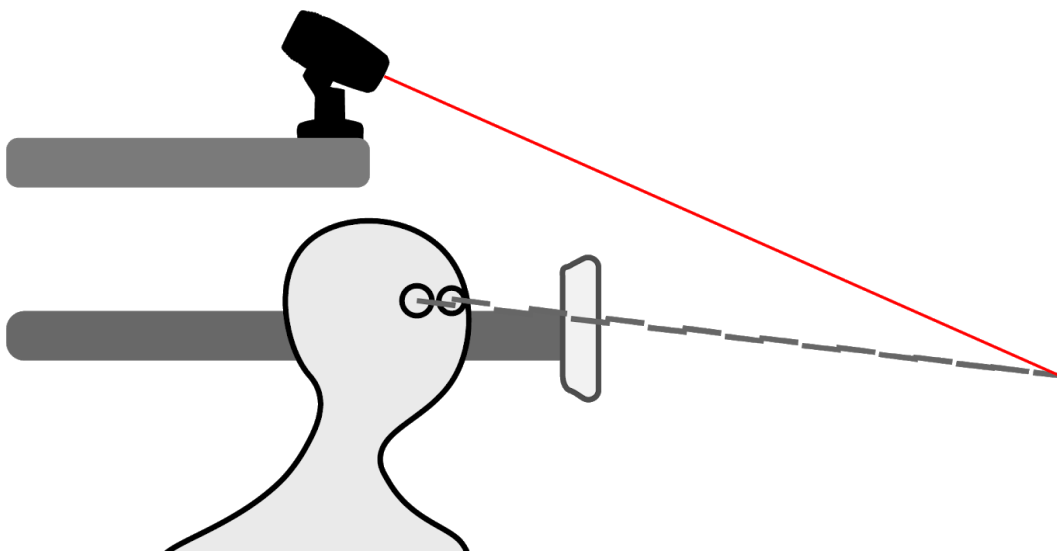


LASERVISION

Mobile Systeme/IT-Systeme MT5 S19



Name	Matrikelnummer	E-Mail
Niklas Bartsch	2216285	niklas.bartsch@haw-hamburg.de
Vincent Friedrich	2326998	vincent.friedrich@haw-hamburg.de
Jonas	2217911	jonas.vonkroge@haw-hamburg.de

Inhaltsverzeichnis

Projektziel	3
Anforderungsanalyse	3
Hardware Installation	3
iOS App	4
Technische Rahmenbedingungen	4
Hardwarebedingungen	4
Softwarebedingungen	4
Technisches Konzept	4
Hardware Installation	4
iOS App	5
Bedienkonzept	6
Abbildung 1: Hardwareinstallation	6
Aufbau	6
Kalibrierung	6
User Interface	6
Steuerung des Pan-Tilt-Gerätes	6
Zeitplan	8
Zeiteinschätzung und Aufgabenverteilung	8
Meilensteine	8

Projektziel

Das Projekt LASERVISION kombiniert die Gesichtserkennung des Apple iPhone X mit der DMX-Steuerung, die im Bereich der Lichttechnik eingesetzt wird.

Dies geschieht im Rahmen der beiden Veranstaltungen "Mobile Systeme" für Media Systems Studenten und "IT-Systeme" für Medientechnik Studenten an der HAW Hamburg, welche im Sommersemester 2019 zusammengelegt werden.

Als Endergebnis soll das Projekt die Steuerung eines DMX-fähigen Gerätes mit Kipp- und Schwenkfunktion (folgend als Pan-Tilt-Gerät bezeichnet) durch ein, über Bluetooth verbundenes, iPhone X und dessen Gesichtserkennung ermöglichen.

Somit wird erreicht, dass die Bewegungsrichtung des Pan-Tilt-Gerätes der Blickrichtung des Nutzers folgt.

Anforderungsanalyse

Hardware Installation

Vorrichtung für Ausstellung: Damit die Abstimmung der Positionierung zwischen dem iPhone, dem Pan-Tilt-Gerät und dem Nutzer während der Nutzung zuverlässig gewährleistet werden kann, wird eine Vorrichtung konstruiert. [Siehe Abbildung 1.](#)

Das iPhone wird an einer Halterung angebracht. Insgesamt soll die Vorrichtung dem Stil einer Art SciFi-Maschine entsprechen.

Der Nutzer soll die Möglichkeit bekommen, über einen externen Schalter einen simulierten Schussmechanismus auszulösen.

Schussmechanismus: Möglichkeiten zur Darstellung des Mechanismus wären

- Ein Laserstrahl, ausgehend vom Pan-Tilt-Gerät in Richtung Wand, Darstellung des Schusses über:
 - Kurzfristige Verfärbung des Lasers
 - Zerplatzen eines schwarzen Luftballons durch den Laser
 - Vorrichtungen an festen Punkten an der Wand, die auf Lichteinfluss reagieren
 - Vorrichtungen hinter Luftballons, die besagte Luftballons zum Platzen bringen - somit das Platzen des Luftballons durch den Laser nur simulieren um den Einsatz von gefährlichem Laser zu vermeiden
- Eine Nerfgun, befestigt am Pan-Tilt-Gerät, und Auslösung des Abzugs nach Betätigung des externen Schalters

iOS App

Gesichtserkennung: Die iOS App wird die Blickrichtung der Pupillen des Nutzers erkennen. So kann erreicht werden, dass das Pan-Tilt-Gerät dem Blick des Nutzers folgt.

Bluetooth-Verbindung: Die iOS App stellt eine Bluetooth-Verbindung zum Arduino her. Die erkannte Blickrichtung wird so an den Arduino übertragen und dort z.B. an das Pan-Tilt-Gerät weitergeleitet.

Technische Rahmenbedingungen

Hardwarebedingungen

Kern des Systems ist ein Arduino Uno (Mega 2560).

Dieser ist verbunden mit einem HM-10 Bluetooth 4.0-Modul, um die Verbindung zu einem iOS-Gerät zu ermöglichen.

An den Arduino ist außerdem ein Pan-Tilt-Gerät angeschlossen, welches sich über das digitale Steuerprotokoll Digital Multiplex (DMX) ansteuern lässt.
In Betracht käme beispielsweise ein Moving-Head.

Zur Gesichtserkennung wird ein iPhone X oder neuer mit FaceID verwendet.

Softwarebedingungen

Die Zielplattform der App zur Ansteuerung des Arduino ist Apple iOS 12.0 oder neuer.
Die Programmierung der App erfolgt in der Programmiersprache Swift, die Gesichtserkennung wird mithilfe des Frameworks Apple ARKit 2.0 realisiert.
Die Programmierung des Arduino erfolgt in der Programmiersprache C.

Technisches Konzept

Hardware Installation

Vorrichtung für Ausstellung: Der Arduino wird in der Vorrichtung außerhalb des Sichtfeldes des Nutzers angebracht. Das iPhone wird über ein Ladekabel mit Strom versorgt.

Arduino: Die Parameter der Gesichtserkennung werden vom Arduino für das Pan-Tilt-Gerät übersetzt und mit möglichst geringer Latenz übertragen, sodass er die Schnittstelle zwischen Gesichtserkennung und tatsächlicher Motorbewegung schließt.

Pan-Tilt-Gerät: Der Moving Head ist entweder als Mittelpunkt der Präsentation gut sichtbar installiert, oder rückt je nach Anwendung in den Hintergrund und überlässt dem Lichtsignal selbst die Aufmerksamkeit.

DMX: Die Dmx.Simple Bibliothek eignet sich für alle verbreiteten Arduinos, einschließlich des Megs. Eine Lichtsteuer-Software um aus dem Arduino einen DMX-Controller zu machen wäre beispielsweise "Q Light Commander", welche gratis und für Mac, Windows und Linux verfügbar ist.

iOS App

Gesichtserkennung: Die iOS App nutzt zur Gesichtserkennung das Apple Framework ARKit 2.0. Das Framework bietet die Möglichkeit, Teile der Gesichtsmuskulatur und die Blickrichtung der Pupillen zu erkennen. Die aktuelle Blickrichtung des Nutzers wird von dem Framework in Form eines 3D-Vektors repräsentiert. Durch einen Kalibrierungsprozess wird der Raum auf die Blickrichtung des Nutzers abgestimmt und somit errechnet, wohin der Nutzer seinen Blick richtet, damit dieser Wert später zur Ansteuerung des Pan-Tilt-Gerätes genutzt werden kann. Da das Framework hauptsächlich für AR-Anwendungen ausgelegt ist, wird in der Standardkonfiguration der Kamerastream mit angezeigt. Dies kann zu Testzwecken durchaus hilfreich sein, in der praktischen Anwendung soll stattdessen ein Steuerinterface angezeigt werden.

User Interface: Das User Interface der App wird in Xcode mit dem Interface Builder entwickelt. Grundlage der Steuerelemente in iOS ist das Cocoa Touch Framework von Apple.

Bluetooth-Verbindung: Die Bluetooth-Verbindung zum Arduino wird mithilfe des Apple Frameworks CoreBluetooth realisiert. Die Daten werden zwischen den Geräten seriell ausgetauscht. Auf Seite der App werden Werte wie Koordinaten der Blickrichtung, oder Kommandos an das Pan-Tilt-Gerät in Objektorientierter Datenstruktur behandelt, in ein Datenobjekt enkodiert und so per Bluetooth an den Arduino übermittelt. Auf dem Arduino werden diese Datenobjekte dekodiert.

Bedienkonzept

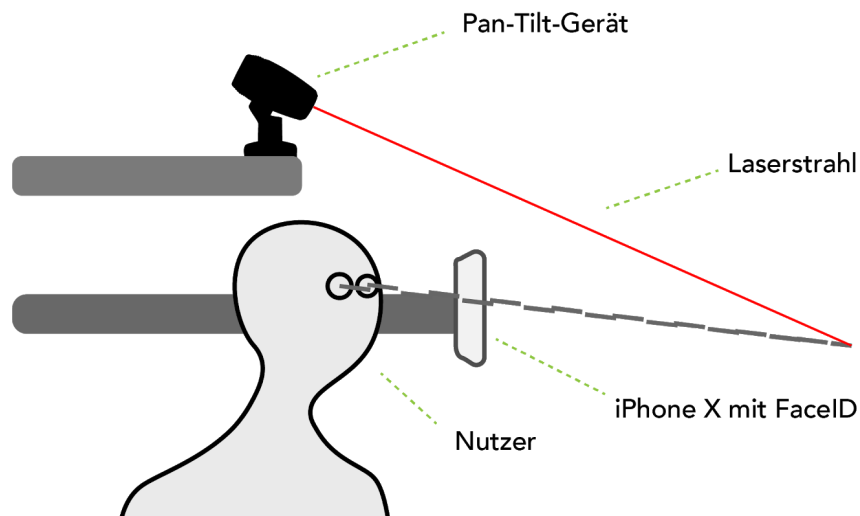


Abbildung 1: Hardwareinstallation

Aufbau

Der Nutzer wird seinen Kopf in einer Vorrichtung platzieren. Vor ihm wird das iPhone X angebracht sein. Über dem Kopf des Nutzers wird das Pan-Tilt-Gerät angebracht sein.

Kalibrierung

Die Anwendung auf dem iPhone startet mit einem Kalibrierungsprozess, um die individuelle Augenposition des Nutzers auf den Raum abzustimmen. Der Nutzer wird dabei verschiedene fixe Punkte im Raum für einige Sekunden anvisieren müssen.

User Interface

Das User Interface der Anwendung auf dem iPhone nach der Kalibrierung wird neben einem Button zur erneuten Kalibrierung und einem Button zum An- und Abschalten des Pan-Tilt-Gerätes einen Einstellungsbildschirm für die Bluetooth-Verbindung mit dem Arduino haben. Die Möglichkeiten der visuellen Begleitung, wie z.B. eine Animation beim Aufladen der Waffe, müssen während der praktischen Nutzung des Systems hinsichtlich der Sinnhaftigkeit und Intuition mit dem Konzept verglichen und evaluiert werden.

Steuerung des Pan-Tilt-Gerätes

Nach erfolgreicher Kalibrierung per App und bei aktiver Bluetooth-Verbindung, wird die Position des Pan-Tilt-Gerätes sich in Echtzeit der Augenposition des Nutzers anpassen.

Einen Schuss wird der Nutzer über einen externen Schalter auslösen können. Der Schalter wird kein Bestandteil des User Interface der App werden, um Probleme der Kalibrierung durch Positionsveränderungen des iPhones zu vermeiden.

Zeitplan

Zeiteinschätzung und Aufgabenverteilung

Thema	Zuständige Person	Einschätzung der benötigten Stunden
Konzeption	Niklas, Jonas und Vincent	16h
Programmierung - iOS - Arduino	Niklas und Vincent Jonas	60h 60h
Testing		20h
Dokumentation	Niklas, Jonas und Vincent	24h
Projektmanagement	Vincent	20h
Gesamt:		200h

Meilensteine

Datum	Event	Arbeitsziel
9.4	Konzeptbesprechung	Vorstellung des Konzeptes
16.4		Research <ul style="list-style-type: none">• ARKit Framework• DMX Moving Head Steuerung
23.4		Bluetooth Verbindung und Transfer von Daten
30.4		Steuerung des Movingheads durch UI Regler
7.5		Hardware Installation konzipieren und umsetzen
14.5		Gesichts- und Augenerkennung
21.5		Steuerung des Moving Heads durch die Augenrichtung
28.5		Fine-Tuning des Prototypen

4.6	Generalprobe	
11.6		Finetuning
18.6		Projektdokumentation
25.6	Abgabe	Abgabe der Dokumentation
2.7	Präsentationsvorbereitung	
9.7	Präsentationsvorbereitung	
11.7	Rundgang	