

# LIGHTVISION

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 von Kroge	2217911	jonas.vonkroge@haw-hamburg.de

# Inhaltsverzeichnis

Einleitung	3
Konzept	3
Umsetzung	3
Equipment und Aufbau	3
Software und Ausstellung	4
Fazit	5
Ergebnis und Verbesserungen	5
Schaltplan	7
Steckplatine	8

## Einleitung

„LightVISION“ ist ein Projekt welches im Zuge des Kurses „Mobile Systeme“ im Studiengang Media Systems bzw. IT-Systeme im Studiengang Medientechnik entstanden ist. Es wird betreut durch Prof. Dr. Torsten Edeler und Prof. Dr. Andreas Plaß. Media Systems und Medientechnik Studenten arbeiten gemeinsam an der Umsetzung eines Projektes, welches am Ende des Semesters als Ausstellungsstück des „Rundgang Finkenau“ zum Einsatz kommt.

## Konzept

Endergebnis unseres Projektes sollte es sein, ein DMX-fähiges Gerät mit Kipp- und Schwenkfunktion (folgend Pan-Tilt-Gerät) über ein iPhone X und dessen Gesichtserkennung (folgend FaceID) zu steuern, sodass die Bewegungsrichtung des Pan-Tilt-Gerätes der Blickrichtung des Nutzers folgt.

Dieses Projektziel konnten wir erfolgreich erreichen.

Das Projekt LightVISION kombiniert die Gesichtserkennung des Apple iPhone X mit der im Bereich der Lichttechnik eingesetzten DMX-Steuerung.

Im Laufe der Entwicklung, ließen wir Ideen wie den simulierten Schussmechanismus oder das Zerplatzen von Luftballons fallen, weil das Umsetzen der Idee als eine Art Spiel, auch ohne Pan-Tilt-Gerät rein virtuell realisierbar gewesen wäre und somit den Reiz einer hardware basierten Umsetzung verloren hätte. Des Weiteren wollte wir die Zeit in das Optimieren des eigentlichen Projektzieles stecken.

## Umsetzung

### Equipment und Aufbau

Als Pan-Tilt-Gerät zur Steuerung des Lasers hatten wir die engere Auswahl zwischen einem Moving Head oder einem Scanner. Zur Nutzung eines Lasers an der HAW wird ein Laserschein gefordert. Trotz Vorhandensein des besagten Scheines, entschieden wir uns aus Sicherheitsgründen gegen die Nutzung eines Lasers der Klasse 3R. Die finale Wahl fiel nicht auf einen Moving-Head, sondern auf einen technisch simpleren, aber dafür annähernd latenzfreien und schnell neigbaren Scanner.

Infolge der Anpassung des Konzeptes zur Gewährleistung der Sicherheit, wurde der Projektname von „Laservision“ zu „LightVISION“ geändert. Den Augen sollte kein Laser mehr, sondern ein Lichtkegel des Scanners folgen. Wir entschieden uns für das Standardsymbol des Scanners mit der besterkennbarsten Mitte, sodass die Augen bei der Kalibrierung einen deutlichen Fokuspunkt finden.

Den Stecker im XLR Eingang des Scanners löteten wir an den Arduino Uno, welcher über ein HM-10 Bluetoothmodul die Datenobjekte des iPhones seriell empfängt, dekodiert und als DMX Signal an den Scanner schickt.



Zur optimalen Nutzung der begrenzten Fläche, auf welche der Scanner abbildet, sollte sich eine Höhe von circa 70 cm und ein Abstand von etwa 2,5 Metern zur Wand als geeignet herausstellen. Das Lichtlabor konnte mit einem auf einer Bodenplatte verschraubten Metallrohr und einer Schelle aushelfen, womit wir den Scanner stabil in dieser Höhe anbrachten. Den Scanner konnten wir privat organisieren.

Abbildung 1: Scanner als Pan-Tilt-Gerät

Das für die FaceID-Erkennungsbibliothek nötige iPhone X wurde uns von der Firma "For Sale Mobile GmbH" zur Verfügung gestellt.

Um dieses fest zu installieren, wurde eine Smartphonehalterung auf ein Stativ geschraubt.

Der Arduino und das Bluetoothmodul wurden in einer Holzbox geschützt auf der Bodenplatte platziert.

In den ersten Versionen unserer Hardwareinstallation war die Haltestange für das iPhone fest mit dem Rohr verbunden, um einen einheitlichen Look zu bewahren. In der finalen Version stellten wir jedoch das Stativ mit dem iPhone unmittelbar neben den Scanner, um die Höhenverstellbarkeit für die verschiedenen Körpergrößen der Ausstellungsgäste zu erleichtern. Eine Einheit bildete die Installation optisch trotz dessen.

## Software und Ausstellung

Die iOS-App wurde nativ in Swift entwickelt und nutzte die Apple-Programmiersprache Swift 5.0. Die Swift-Bibliotheken *CoreBluetooth* für die Bluetoothverbindung, *Codable* für die Kodierung der Datenstruktur als Text zur seriellen Übertragung, *ARKit 2.0* für die Gesichtserkennung und *SceneKit* für die 3D-Darstellung des User-Interfaces für die Kalibrierung wurden eingesetzt.

Die Werte der Blickrichtung wurden als 3D Vektoren ausgegeben, welche zur Übertragung auf den Scanner, in ein serielles Signal aus X- und Y-Werten für die Steuerung des Scannerspiegels umgerechnet werden mussten.

Zum Testen des Konzeptes wurde zuerst der Kalibrierungsprozess anhand eines roten Quadrates auf dem Videostream simuliert. Das Quadrat wandert an 4 Fixpunkte, rechts, unterhalb, links und oberhalb der Augen, der Nutzer behält das Quadrat dabei im Auge und mit jedem Bestätigen über einmaliges Tippen auf den Bildschirm wird der Blickrichtungsvektor für die entsprechende Quadratposition gespeichert. Ist der Prozess abgeschlossen, wandert das Quadrat entsprechend der Augenausrichtung zwischen den 4 Positionen und darüber hinaus umher.



Abbildung 2: Erster Prototyp der iOS-App

Dasselbe Prinzip übertragen wir auf den praktischen Aufbau, bei dem das Lichtsignal des Scanners anstelle des Quadrates an eine Wand projiziert und wird dem Blick des Nutzers folgt. Auch der Kalibrierungsprozess folgte auf der Ausstellung demselben Konzept, das Userinterface der App wurde jedoch durch ein dreidimensionales Gitternetz des Gesichtes anstelle des Kamerastreams und einer ansprechenden Optik des Menüs erweitert um die Userexperience zu verbessern und dem Nutzer klar zu visualisieren, dass das Gesicht erkannt wird.

Der Gast nähert sich dem auf Augenhöhe befindlichen iPhone und wird sofort vom Polygongitter widergespiegelt. Mimik und Augenausrichtung des virtuellen Gesichtes entsprechen der des Nutzers, welcher aufgefordert wird den Bildschirm zu berühren um die Kalibrierung zu starten. War diese erfolgreich, hat die Person das Lichtsignal an den 4 Positionen nicht aus den Augen verloren und möglichst den Kopf still gehalten, folgt das Licht nun den Augen.

Ein Hinweis darauf, dass man durch längeres Gedrückthalten den Prozess zurücksetzen und wiederholen kann, erscheint neben der Bestätigung, dass die Kalibrierung erfolgreich abgeschlossen wurde. Ebenfalls integriert, ist eine Sprachausgabe, die dem Nutzer alle beschriebenen Hinweise durch die Bedienung laut vorliest.



Abbildung 3: Finaler Kalibrierungsprozess der iOS-App

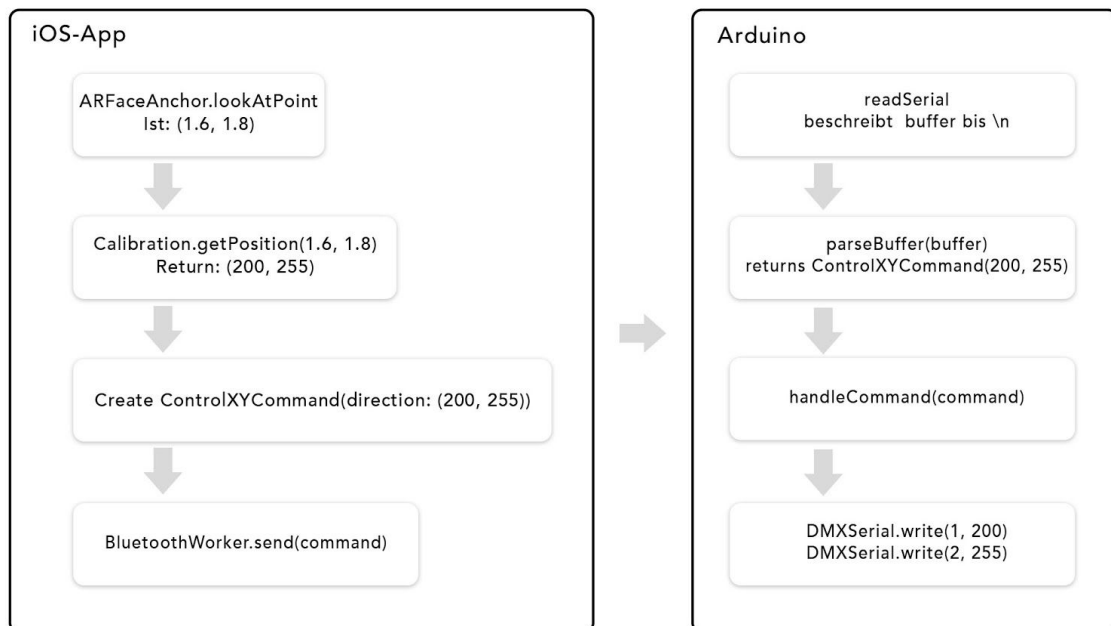


Abbildung 4: Visualisierung der Kommunikation zwischen iOS-App und Arduino nach Kalibrierung

Während des Rundgangs wurde deutlich, dass verschiedene Gesichter und Menschen sehr unterschiedliche Ergebnisse erzielten. Unter Anderem weil es schwer fiel den Kopf nicht zu bewegen, oder auch bei der Kalibrierung nicht auf den Bildschirm zu schauen. Aber auch Augenabstand und individuelle Gesichtszüge erzielten bei FaceID unterschiedliche Erkennungswerte. Bei einem Teil der Nutzer klappte die Erkennung auf Anhieb und erzeugte Überwältigung, das Licht folgte ihrem Blick sehr genau. Ein anderer Teil musste das Gerät mehrmals kalibrieren und erreichte eher ein tendenzielles Folgen der Blickrichtung.

Innerhalb des Bereichs zwischen den Kalibrierungspunkten funktionierte dies meist besser als außerhalb davon, was die Begeisterung der Rundgänger jedoch nicht milderte. Einige Nutzer hatten auch Probleme mit der Erkennung, da sie während der Nutzung zu sehr ihren Kopf bewegten, was die Erkennungsgenauigkeit von FaceID minderte.



Abbildung 5: Das Projektplakat für die Ausstellung "Rundgang Finkenau"

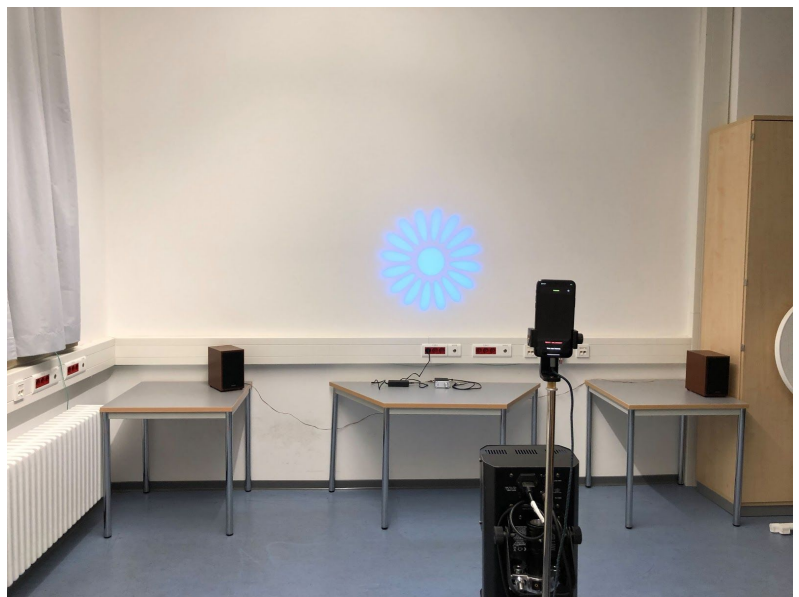


Abbildung 6: Die finale Installation bei der Ausstellung "Rundgang Finkenau"



## Fazit

### Ergebnis und Ausblick

Wir haben erfolgreich einen LED Scanner über ein iPhone X und dessen Gesichtserkennung gesteuert, sodass die Bewegungsrichtung des Scanners der Blickrichtung des Nutzers folgt.

Sinnvolle Weiterentwicklung wäre es, die Kopfbewegung in die Blickrichtung mit einzuberechnen, anstatt nur die Augenposition zu berücksichtigen, oder eine Vorrichtung, in die der User seinen Kopf arretiert, aufzubauen.

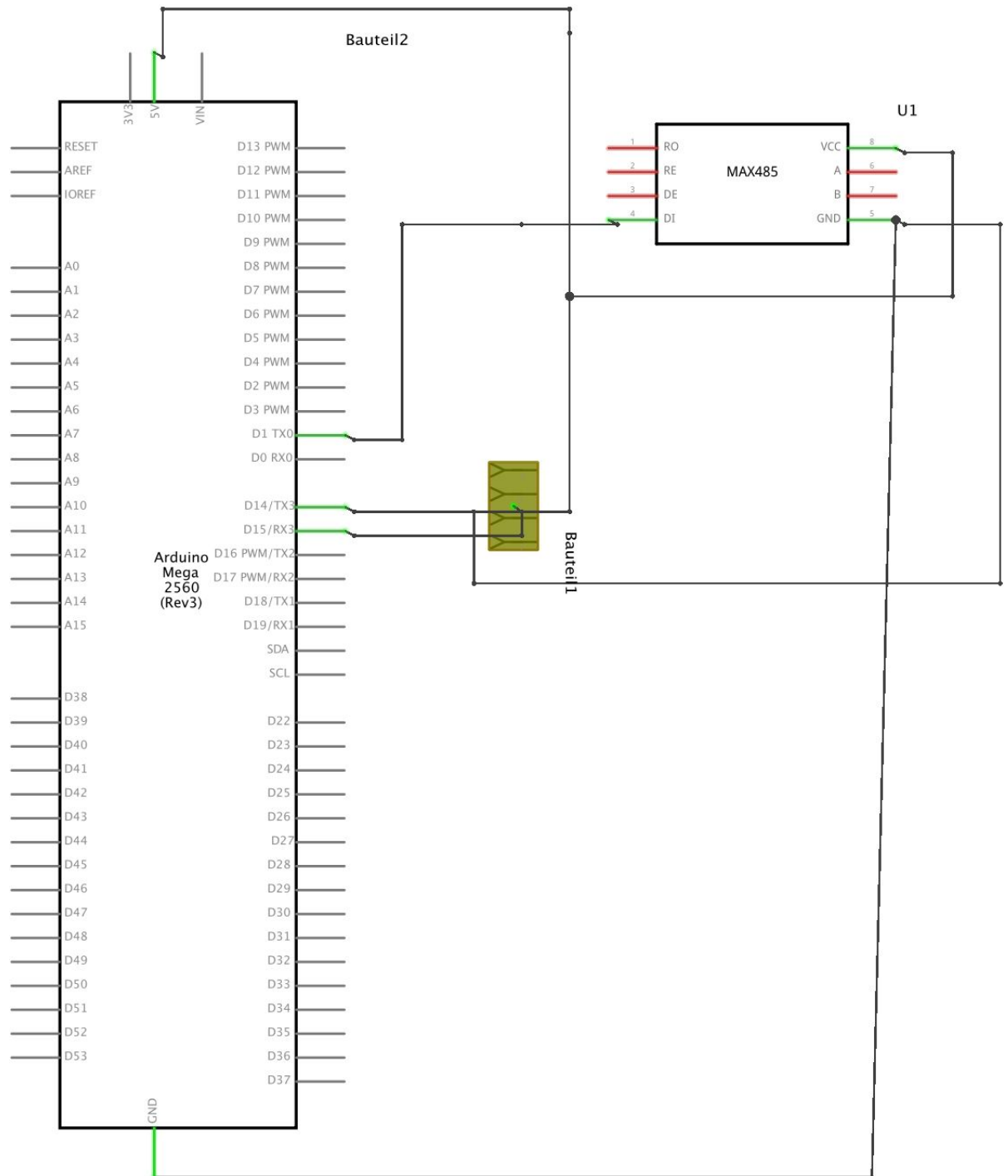
Ebenso könnte die nicht lineare Geschwindigkeit des Lichtsignals auf der Wand, welche abhängig vom Einfallswinkel und der damit verbunden Entfernung zum Scanner ist, durch sinnvolle Umrechnung kompensiert werden. Genauso interessant wäre es aber auch, anstelle einer flachen Wand, eine Kuppel als Bestrahlungsfläche zu konstruieren, sodass Abstand und Winkel zwischen Scanner und Fläche stets gleich bleiben.

Würde unser Konzept weiterentwickelt und optimiert werden, könnte Bühnen- oder Überwachungstechnik mit deutlich flexibleren und lebendig wirkenden Scheinwerfern aufgerüstet werden. Beispielsweise Spotlights, die so hartnäckig wie das menschliche Auge das Ziel im Fokus behalten, oder Stirnlampen, welche sogar den Pupillen folgend Licht ins Dunkel bringen.

Gesichtserkennungssoftware hat in vielen Bereichen eine spannende Zukunft vor sich, besonders faszinierend jedoch im Bereich; "LightVISION".



## Schaltplan



fritzing

## Steckplatine

