**DIPLOMARBEIT**

**NIMBUS**



Verfasser: Janina KOWATSCH

Maximilian LANGER

Markus WANKE

Betreuer: Mag. Clemens Wiesinger, MBA

Abteilung für Informatik – Software Engineering, 5AHIF

Verfassererklärung

Ich erkläre, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt, die aus anderen Werken übernommene Inhalte durch Quellenangaben erkenntlich gemacht, sowie durch Künstliche Intelligenz oder sonstige Tools generierte Inhalte unter Angabe von Prompt und Produktversion in der Quellenangabe gekennzeichnet habe.

Gegebenenfalls Kooperation mit Firmen und Sperrvermerke ergänzen.

Wien, [Veröffentlichungsdatum]

|  |  |
| --- | --- |
| Janina KOWATSCH |  |
|  | Unterschrift |

|  |  |
| --- | --- |
| Maximilian LANGER |  |
|  | Unterschrift |

|  |  |
| --- | --- |
| Markus WANKE |  |
|  | Unterschrift |

Kurzfassung

NIMBUS ist ein Schüler:innen-Projekt, das im Rahmen der Diplomarbeit an der HTL Donaustadt entwickelt wird. Es kombiniert Drohnentechnologie und Künstliche Intelligenz (KI) zur Unterstützung blinder und sehbeeinträchtigter Personen bei der sicheren Navigation in alltäglichen Umgebungen. Durch die Sensibilisierung für die Bedürfnisse blinder und sehbeeinträchtigter Menschen und die Förderung von Inklusion trägt NIMBUS wesentlich zur gesellschaftlichen Bewusstseinsbildung bei. Die technisch-spielerische Herangehensweise, wie etwa durch Showcases beim Tag der offenen Tür (TdoT) der HTL-Donaustadt, die den direkten Kontakt mit Drohnen ermöglicht, fördert nicht nur Kreativität und Innovation, sondern macht das Kennenlernen der Technologie auch zu einem spannenden Erlebnis.

Abstract

NIMBUS is a student project developed as part of a diploma thesis at HTL Donaustadt. It combines drone technology and artificial intelligence (AI) to support blind and visually impaired individuals in safely navigating everyday environments. By raising awareness of the needs of blind and visually impaired people and promoting inclusion, NIMBUS makes a significant contribution to societal awareness. The project’s technical and playful approach, such as showcases at HTL Donaustadt’s open house events, allows for direct interaction with drones, fostering creativity and innovation while making the exploration of technology an exciting experience.

Inhaltsverzeichnis

[1 Einleitung 7](#_Toc187690623)

[2 Best Practices zur Entwicklung einer barrierefreien App zur Drohnensteuerung 8](#_Toc187690624)

[2.1 Einführung 8](#_Toc187690625)

[2.2 Theoretische Grundlagen 8](#_Toc187690626)

[2.2.1 User Experience 8](#_Toc187690627)

[2.2.2 Barrierefreiheit (Accessibility) 10](#_Toc187690628)

[2.2.3 WCAG (Web Content Accessibility Guidelines) 12](#_Toc187690629)

[2.2.4 EN 301 549 13](#_Toc187690630)

[2.2.5 Accessibility Checklists und Testing Tools 14](#_Toc187690631)

[2.2.6 Plattformunabhängige Mobile-App Entwicklung 15](#_Toc187690632)

[2.2.7 Flutter für plattformunabhängige Entwicklungen 16](#_Toc187690633)

[2.2.8 Tensorflow Lite für Machine Learning auf Mobilen Geräten 17](#_Toc187690634)

[2.3 Konzept 17](#_Toc187690635)

[2.3.1 Anforderungen an die mobile App 17](#_Toc187690636)

[2.3.2 Aufbau und Struktur der App 17](#_Toc187690637)

[2.3.3 Schnittstellen 17](#_Toc187690638)

[2.3.4 Maßnahmen in Flutter für Accessibility 17](#_Toc187690639)

[2.4 Herausforderungen der Umsetzung 18](#_Toc187690640)

[2.4.1 Barrierefreiheit in den Screens 18](#_Toc187690641)

[2.4.2 Plattformunabhängige Nimbus-App 18](#_Toc187690642)

[2.4.3 Überprüfung der Barrierefreiheit 18](#_Toc187690643)

[2.4.4 Herausforderungen bei der Flutter-Integration 18](#_Toc187690644)

[2.4.5 Tensorflow Lite und Barrierefreiheit 18](#_Toc187690645)

[2.5 Fazit und Ausblick 18](#_Toc187690646)

[3 Einsatz von Künstlicher Intelligenz in Drohnen zur Hinderniserkennung für sehbeeinträchtigte Personen 19](#_Toc187690647)

[3.1 Einführung 19](#_Toc187690648)

[3.2 Theoretische Grundlagen 20](#_Toc187690649)

[3.2.1 Künstliche Intelligenz und Maschinelles lernen 20](#_Toc187690650)

[3.2.2 Bilderkennung in echtzeit 20](#_Toc187690651)

[3.2.3 Trainingsdatensätze und Training 22](#_Toc187690652)

[3.3 Konzept 23](#_Toc187690653)

[3.3.1 Systemarchitektur 23](#_Toc187690654)

[3.3.2 Verwendung von Tensorflow Lite für die Bilderkennung 26](#_Toc187690655)

[3.4 Herausforderungen der Umsetzung 26](#_Toc187690656)

[3.4.1 Trainingsdatensätze und Modellgenauigkeit 26](#_Toc187690657)

[3.5 Fazit und Ausblick 28](#_Toc187690658)

[4 Entwicklung der Drohnenhardware sowie -Firmware 30](#_Toc187690659)

[4.1 Einführung 30](#_Toc187690660)

[4.2 Theoretische Grundlagen 30](#_Toc187690661)

[4.2.1 Drohnenplattformen 30](#_Toc187690662)

[4.2.2 Antriebssystem 31](#_Toc187690663)

[4.2.3 Microcontroller 32](#_Toc187690664)

[4.2.4 Sensorik 34](#_Toc187690665)

[4.2.5 Kommunikationsmodule 34](#_Toc187690666)

[4.2.6 Autonomes Fliegen 35](#_Toc187690667)

[4.3 Konzept 36](#_Toc187690668)

[4.3.1 Konzept 1 36](#_Toc187690669)

[4.3.2 Konzept 2…… 36](#_Toc187690670)

[4.4 Herausforderungen der Umsetzung 37](#_Toc187690671)

[4.5 Fazit und AusblicK 37](#_Toc187690672)

[5 Literaturverzeichnis 38](#_Toc187690673)

[6 Abbildungsverzeichnis 39](#_Toc187690674)

[7 Tabellenverzeichnis 40](#_Toc187690675)

[8 Abkürzungsverzeichnis 41](#_Toc187690676)

[9 Anhang 42](#_Toc187690677)

# Einleitung

# Best Practices zur Entwicklung einer barrierefreien App zur Drohnensteuerung

Janina KOWATSCH

## Einführung

Barrierefreiheit ist ein zentraler Aspekt der moderner App-Entwicklung, insbesondere wenn Anwendungen für Zielgruppen mit einer Behinderung entwickelt werden. Im Zusammenhang mit der Steuerung von Drohnen stellt dies eine besondere Herausforderung dar, da es nicht nur um die Interaktion mit der App selbst, sondern auch um die Verbindung zwischen Software und Hardware geht. Ziel von diesem Kapitel ist es, Best Practices zur Entwicklung einer barrierefreien App für die Drohnensteuerung zu beschreiben. Dabei wird besonders auf die Bedürfnisse blinder und sehbeeinträchtigter Nutzer und Nutzerinnen gelegt.

## Theoretische Grundlagen

### User Experience

User Experience (UX) beschreibt das Erlebnis und die Wahrnehmung eines Nutzers oder einer Nutzerin mit einem Produkt, einer Dienstleistung oder einem System. Dies umfasst sich mit unterschiedlichen funktionalen Aspekten, wie der Nutzbarkeit (Usability), als auch mit den emotionalen Aspekten, wie die Zufriedenheit der Anwendung. Somit fördert eine gute User Experience nicht nur die Effektivität der Nutzung, sondern trägt auch dazu bei, eine Zufriedenstellung für den Nutzer zu gewährleisten.

Begriff User Interface (UI) ist eng mit der UX verbunden. Während UX das gesamte Nutzererlebnis umfasst, konzentriert sich UI auf die Gestaltung der Schnittstellen, mit denen Nutzerinnen und Nutzer interagieren. Die visuelle und funktionale Gestaltung des UI trägt maßgeblich zur Usability und damit zur UX bei [1].

#### Usability und Utility

Die Usability (Benutzerfreundlichkeit) ist ein wichtiger Teil der UX und beschreibt, wie einfach und effizient Nutzer und Nutzerinnen, sowie auch Kunden und Vorgesetzte der Nutzer, ein System verwenden können, um ihre Ziele zu erreichen [2]. Utility (Nutzen) bezieht sich hingegen auf die Frage, ob ein System überhaupt die richtigen Funktionen bereitstellt, um die Bedürfnisse der Zielgruppe zu erfüllen. [3] Zusammen bilden diese beiden Aspekte die Grundlage für ein erfolgreiches UX-Design.

#### Personas

Eine Persona ist eine Art Modell, die einen typischen Nutzer oder eine typische Nutzerin repräsentieren soll. Sie basiert auf Daten, die durch eine Nutzerforschung gesammelt wurden und beinhalten Informationen, wie Hintergrund der Person, Bedürfnisse, die Herausforderungen, die eine sehbeeinträchtigte Person durchleben muss, welche Bereiche betroffen sind und der Grad der Blindheit. Personas helfen dabei, das Nutzerverhalten zu verstehen und das Design der App gezielt auf die Bedürfnisse dieser Zielgruppe abzustimmen. [4] Die Erstellung von Personas, umfasst folgende Schritte normalerweise:

* Sammeln der Daten von verschiedenen Personen der Zielgruppe, durch Umfragen und Interviews
* Ähnliche Herausforderungen und Bedürfnisse werden zusammengefasst
* Mit diesen Informationen wird dann eine Persona in einem verständlichen Format dargestellt

Personas sind hilfreich, um Teams auf eine gemeinsame Zielgruppe auszurichten. Sie fördern das Verständnis für die Nutzerperspektive und unterstützen dabei Entscheidungen im Designprozess fundiert zu treffen. Durch die Nutzung von Personas wird vermieden, dass Designs auf subjektive Annahmen basiert und stattdessen wird ein nutzerzentrierter Ansatz verfolgt. [5]

Beispiel für eine Persona:

Persona 1: Anna, 25 Jahre, vollblind

Hintergrund: Anna wurde mit einer genetischen Erkrankung geboren, die zu vollständiger Blindheit führte. Sie hat nie Licht oder Farben gesehen.

Grad der Blindheit: Vollständig blind (Amaurose), keine Lichtwahrnehmung.

Betroffene Bereiche: Gesamtes Auge, keine optische Reizverarbeitung.

Herausforderungen:

* Orientierung in neuen Umgebungen
* Nutzung von Technologie ohne taktile oder auditive Unterstützung
* Zugang zu gedruckten Informationen

Bedürfnisse:

* Braille-Displays und -Bücher
* Sprachgesteuerte Assistenten und Screenreader
* Mobilitätstraining mit einem Blindenstock oder Blindenhund

#### User Journey

Eine User Journey Map ist ein visuelles Werkzeug, das die Erfahrungen eines Nutzers oder einer Nutzerin entlang der verschiedenen Kontaktpunkte mit einem Produkt oder einer Dienstleistung darstellt. Sie dokumentiert die Interaktionen, Gefühle und Herausforderungen, die Nutzer während ihres Weges erleben. [6]

#### Benutzerfeedback

Die kontinuierliche Integration von Benutzerfeedback ist ein zentraler Bestandteil des UX-Designs. Durch verschiedene Methoden wie Umfragen, Interviews, Usability-Tests oder Analysewerkzeuge können wertvolle Informationen gewonnen werden, die die Entwicklung und Optimierung von Produkten unterstützen.

*„Für eine gute User Experience sind die vier folgenden Faktoren wichtig: Utility, User Interface, Usability und Joy of Use.“* [1]

Das Zitat verdeutlicht, dass eine gute UX nicht nur auf funktionalen, sondern auch auf emotionalen und ästhetischen Dimensionen basiert. Diese Faktoren arbeiten zusammen, um ein ganzheitlich positives Nutzererlebnis zu schaffen, das sowohl die praktischen Bedürfnisse als auch die persönlichen Erwartungen der Nutzerinnen und Nutzer erfüllt.

### Barrierefreiheit (Accessibility)

* Definition von Barrierefreiheit und Abgrenzung zu User Experience.
* Arten von Behinderungen, die bei der App-Entwicklung berücksichtigt werden müssen
* TalkBack und VoiceOver: Screenreader-Technologien für Android und iOs

Barrierefreiheit (Accessibility) bezieht sich auf die Gestaltung von Produkten, Dienstleistungen und Umgebungen, die es Menschen mit Behinderungen ermöglicht, diese uneingeschränkt zu nutzen. Im Kontext der Softwareentwicklung bedeutet dies, sicherzustellen, dass digitale Anwendungen von Menschen mit verschiedenen Einschränkungen genutzt werden können. Dazu zählen visuelle, auditive, motorische und kognitive Beeinträchtigungen.

Im Unterschied dazu umfasst User Experience (UX) das gesamte Nutzungserlebnis eines Produkts, einschließlich seiner Funktionalität, Ästhetik und emotionalen Wirkung. Barrierefreiheit ist ein wesentlicher Bestandteil der UX, da sie sicherstellt, dass alle Nutzergruppen in der Lage sind, ein Produkt effektiv und zufriedenstellend zu nutzen. Ohne Barrierefreiheit kann eine positive UX für Menschen mit Behinderungen nicht erreicht werden. [7]

#### Arten von Behinderungen

Um eine inklusive App-Entwicklung zu gewährleisten, müssen folgende Arten von Behinderungen berücksichtigt werden:

* **Visuelle Beeinträchtigungen**: Dazu gehören Blindheit, eingeschränktes Sehvermögen, Farbenblindheit und andere Sehbehinderungen wie Makuladegeneration oder diabetische Retinopathie.
* **Auditive Beeinträchtigungen**: Menschen mit Hörverlust oder Taubheit können Schwierigkeiten mit rein auditiven Informationen haben.
* **Motorische Einschränkungen**: Dazu gehören Probleme mit der Feinmotorik oder Lähmungen, die die Nutzung von Touchscreens oder Eingabegeräten erschweren.
* **Kognitive Beeinträchtigungen**: Menschen mit Lernbehinderungen, Gedächtnisproblemen oder Aufmerksamkeitsdefiziten benötigen klare und einfache Navigationsstrukturen.

#### Android - TalkBack

Moderne mobile Betriebssysteme verfügen in der Regel über integrierte Bildschirmleseprogramme, die speziell für Menschen mit Sehbehinderungen entwickelt wurden. Auf Android-Geräten ist TalkBack der standardmäßige Screenreader. Diese Funktion ist kostenlos und kann einfach über die Barrierefreiheitseinstellungen aktiviert werden. TalkBack ermöglicht es Nutzerinnen und Nutzern, ihre Geräte zu bedienen, ohne auf den Bildschirm schauen zu müssen. Durch Wischen nach rechts oder links können sie zwischen den verfügbaren Elementen navigieren, während TalkBack das jeweils ausgewählte Element – wie Texte, Schaltflächen oder Apps – laut vorliest. Ein Doppeltippen auf den Bildschirm aktiviert das ausgewählte Element, z. B. das Öffnen einer App.

Die Navigation kann durch horizontales Wischen mit drei Fingern angepasst werden. TalkBack bietet verschiedene Modi, wie die Navigation nach Symbolen, Überschriften (z. B. bei Websites), Wörtern oder einzelnen Buchstaben. Es verfügt zudem über zahlreiche Steuerungsmöglichkeiten, um mit Elementen wie Schiebereglern, Eingabefeldern oder Menüs zu interagieren. Die Funktion kann individuell angepasst werden, um den spezifischen Bedürfnissen der Nutzerinnen und Nutzer gerecht zu werden. Zusätzlich kann TalkBack mit einer Braillezeile kombiniert werden, um taktile Ausgaben bereitzustellen. [8]

Android bietet darüber hinaus weitere nützliche Barrierefreiheitsfunktionen, darunter Bildbeschreibungen, Anpassungsmöglichkeiten für Schriftgrößen, Farben und Kontraste sowie die Option, lange Texte mit anpassbarer Geschwindigkeit und verschiedenen Stimmen vorlesen zu lassen – unabhängig davon, ob TalkBack aktiviert ist. Außerdem können Größe, Farbe und Kontrast der meisten Texte im System flexibel angepasst werden. [9]

#### iOS - VoiceOver

Apple-Geräte verfügen über einen eigenen Screenreader namens VoiceOver, der als Alternative zu TalkBack dient. VoiceOver ist kostenlos auf allen Apple-Geräten verfügbar und kann in den Barrierefreiheitseinstellungen aktiviert werden. Es ist nicht nur auf iPhones, sondern auch auf anderen Apple-Geräten wie iPads oder Macs verfügbar. Die Bedienung ähnelt TalkBack, da viele Gesten wie horizontales Wischen zur Navigation oder doppeltes Tippen zur Aktivierung von Elementen verwendet werden. Ein zweifaches Tippen mit zwei Fingern pausiert VoiceOver vorübergehend. VoiceOver bietet zudem erweiterte Steuerungsmöglichkeiten, etwa zum Öffnen bestimmter Seiten oder zur Rückkehr zum Startbildschirm. Die Stimme und Geschwindigkeit des vorgelesenen Textes können individuell angepasst werden, und es besteht die Möglichkeit, die Aussprache bestimmter Wörter zu korrigieren. [10]

Zusätzlich zu VoiceOver bietet Apple eine Vielzahl weiterer Barrierefreiheitsoptionen. Die Kamera kann beispielsweise, als Lupe verwendet werden, während die „Point and Speak“-Funktion Objekte im Kamerabild beschreibt. Außerdem lassen sich Textgröße und Anzeigeeinstellungen flexibel anpassen. Funktionen wie das Dimmen blinkender Lichter, die Reduzierung von Bildschirmbewegungen oder das Pausieren animierter Bilder erhöhen ebenfalls die Nutzbarkeit. iOS bietet darüber hinaus eine Zoomfunktion, mit der bestimmte Bereiche des Bildschirms vergrößert werden können. [11]

### WCAG (Web Content Accessibility Guidelines)

Die Web Content Accessibility Guidelines (WCAG), entwickelt vom World Wide Web Consortium (W3C), sind ein internationaler Standard zur barrierefreien Gestaltung von Webinhalten. Ziel der WCAG ist es, sicherzustellen, dass digitale Inhalte für alle Nutzergruppen zugänglich sind, einschließlich Menschen mit Behinderungen.

#### Prinzipien der WCAG

Die WCAG basiert auf vier grundlegenden Prinzipien:

1. Wahrnehmbarkeit (Perceivable): Inhalte und Benutzeroberflächen müssen so gestaltet sein, dass sie für alle Sinne der Nutzer wahrnehmbar sind.

* Beispiele: Textalternativen für visuelle Inhalte, ausreichender Farbkontrast.

1. Bedienbarkeit (Operable): Interaktive Elemente und Navigation müssen so gestaltet sein, dass sie von allen Nutzergruppen bedient werden können.

* Beispiele: Tastaturbedienbarkeit, Zeitsteuerung für Aktionen.

1. Verständlichkeit (Understandable): Informationen und Bedienoberflächen müssen für die Nutzer einfach und klar verständlich sein.

* Beispiele: Konsistente Navigation, klare Fehlermeldungen.

1. Robustheit (Robust): Inhalte müssen mit einer Vielzahl von Technologien und Benutzeragenten, einschließlich assistiver Technologien, kompatibel sein.

#### Ein Bild, das Text, Logo, Schrift, Symbol enthält. Automatisch generierte BeschreibungErfolgskriterien und Konformitätslevel

Abbildung : WCAG  
Level AA Seal

Die WCAG definiert Erfolgskriterien, die in drei Konformitätslevel **(Abbildung 1: WCAG  
Level AA Seal)** unterteilt sind:

1. Level A: Grundlegende Anforderungen an die Barrierefreiheit.
2. Level AA: Erweiterte Anforderungen, die häufig als gesetzlicher Standard gelten.
3. Level AAA: Höchste Anforderungen, die nicht immer vollständig umsetzbar sind.

Die Richtlinien werden regelmäßig aktualisiert, zuletzt in der Version WCAG 2.1, um neuen technologischen Entwicklungen und Nutzerbedürfnissen Rechnung zu tragen.

#### Relevanz und Anwendung

Die WCAG bilden die Grundlage vieler gesetzlicher Anforderungen, wie beispielsweise der EU-Richtlinie 2016/2102. Sie sind essenziell für die Gestaltung barrierefreier Websites und digitaler Anwendungen.

[12] [13]

### EN 301 549

Die EN 301 549 ist eine europäische Norm, die Anforderungen an die Barrierefreiheit von Informations- und Kommunikationstechnologie (IKT) definiert. Sie wurde entwickelt, um die Barrierefreiheit von IT-Produkten und -Diensten in der EU zu gewährleisten und unterstützt die Umsetzung der EU-Richtlinie 2016/2102.

#### Zielsetzung und Anwendungsbereich

Die Norm hat das Ziel, sicherzustellen, dass alle IKT-Produkte und -Dienste für Menschen mit Behinderungen zugänglich sind. Sie gilt für:

* Websites und mobile Anwendungen,
* Software,
* Hardware wie Computer, Telefone und Selbstbedienungskioske.

#### Hauptinhalte

Die EN 301 549 basiert auf den WCAG 2.1 für Webinhalte und ergänzt diese um technische Anforderungen für:

* **Hardware:** Vorgaben für physische Geräte, wie taktile Eingabegeräte oder Braillezeilen.
* **Software:** Anforderungen wie Tastaturbedienbarkeit und alternative Textbeschreibungen.
* **Dokumentation:** Verpflichtung zur Bereitstellung barrierefreier Benutzerhandbücher und Supportdienste.

Ein Bild, das Text, Schrift, Logo, Kreis enthält.

Automatisch generierte BeschreibungEine Visualisierung der EN 301 549 wird durch das sogenannte **EN-Seal** veranschaulicht. Dieses Siegel zeigt an, dass ein Produkt oder eine Dienstleistung den Anforderungen dieser Norm entspricht. In **Abbildung 2: EN 301 549 Seal** wird ein Beispiel für ein solches EN-Seal dargestellt.

Abbildung : EN 301 549 Seal

**Bedeutung**

Die EN 301 549 ist ein zentraler Standard für die barrierefreie Gestaltung von IT-Produkten und wird in Ausschreibungen der öffentlichen Hand häufig als Referenz verwendet. Sie fördert die Inklusion und gewährleistet den Zugang zu digitalen Technologien für alle.

[14] [15]

### Accessibility Checklists und Testing Tools

Die Barrierefreiheit digitaler Anwendungen ist ein zentraler Aspekt moderner Softwareentwicklung. Sie stellt sicher, dass alle Nutzer, einschließlich Personen mit Behinderungen, uneingeschränkten Zugang zu Funktionen und Informationen erhalten. Accessibility-Checklisten und Testing-Tools spielen hierbei eine entscheidende Rolle, indem sie Standards wie WCAG (**WCAG (Web Content Accessibility Guidelines)**) überprüfen und deren Einhaltung sicherstellen.

#### Vergleich von Tools zur Überprüfung der Barrierefreiheit

|  |  |  |  |
| --- | --- | --- | --- |
| Tool | Beschreibung | Vorteile | Nachteile |
| Accessibility Scanner (Android) | Google-App, die UI-Elemente in Android-Apps analysiert und Barrierefreiheitsprobleme identifiziert | Einfach zu bedienen  Erkennt Textgröße, Kontrast und Touch-Ziele  Kostenlos | Nur für Android. Eingeschränkte Funktionalität bei komplexen Designs. |
| Accessibility Insights (Android/iOS/Windows) | Plattformübergreifendes Tool zur Überprüfung von Barrierefreiheit mit automatischen und manuellen Tests | Umfassend für verschiedene Plattformen Schritt-für-Schritt-Anleitungen | Nicht mobil-exklusiv, Desktop-Tools oft stärker ausgeprägt |
| Lighthouse (Android/Chrome) | Google-Tool zur Performance- und Accessibility-Analyse von Web-Apps und PWA | Kostenlos. Gute Integration in Chrome DevTools Unterstützt PWA-Analyse | Keine native App, eher für Web-Apps geeignet.  Eingeschränkt für iOS-spezifische Probleme |
| Color Contrast Analyzer (Android/iOS) | Überprüft Farbkontraste von Apps oder Bildern auf Barrierefreiheit gemäß WCAG-Richtlinien | Einfache Überprüfung von Kontrasten und Lesbarkeit | Beschränkt auf Farbkontrast, keine umfassende Accessibility-Analyse |

### Plattformunabhängige Mobile-App Entwicklung

* Vorteile und Herausforderungen plattformunabhängiger Ansätze.
* Kuzer tabellarischer Überblick über Frameworks
* Design System Theorie <https://www.nngroup.com/articles/design-systems-vs-style-guides/>

Plattformunabhängige Ansätze zur Mobile-App-Entwicklung ermöglichen die Erstellung von Anwendungen, die auf mehreren Betriebssystemen, wie iOS und Android laufen. Dies spart Entwicklungsressourcen und ermöglicht eine breitere Zielgruppenansprache. [16]

#### Vorteile und Herausforderungen plattformunabhängiger Ansätze

**Vorteile:**

* Kostenreduktion: Eine einzige Codebasis verringert Entwicklungs- und Wartungskosten.
* Zeiteffizienz: Schnellere Markteinführung durch parallele Entwicklung für mehrere Plattformen.
* Konsistentes Design: Einheitliche Nutzererfahrung auf unterschiedlichen Geräten.

**Herausforderungen:**

* Leistungseinbußen: Frameworks, wie React Native oder Flutter sind in der Performance nativen Apps oft unterlegen.
* Plattformspezifische Anpassungen: Trotz einheitlicher Codebasis sind plattformspezifische Modifikationen häufig notwendig.
* Eingeschränkte Nutzung nativer Funktionen: Manche APIs oder Hardware-Funktionen sind schwieriger zugänglich.

[16] [17]

#### Tabellarischer Überblick über Frameworks

|  |  |  |  |
| --- | --- | --- | --- |
| Framework | Beschreibung | Programmiersprache | Besonderheiten |
| Flutter | Google-Framework mit eigener Rendering-Engine | Dart | Hoch performant, umfassende Widgets |
| React Native | Facebook-Framework für hybride Apps | Java | |  | | --- | | Hohe Community-Unterstützung |  |  | | --- | |  | |
| Xamarin | Microsoft-Lösung für C#-basierte Apps | C# | Gute Integration mit .NET |

[16] [17] [18]

#### Design System Theorie

Ein konsistentes Designsystem ist ein zentraler Bestandteil plattformunabhängiger Entwicklung. Es dient dazu, wiederverwendbare Komponenten, visuelle Richtlinien und einheitliche Interaktionsmuster zu definieren. Laut Nielsen Norman Group unterscheidet sich ein Designsystem von reinen Style Guides durch die Integration von Code-Komponenten und einer aktiven Governance-Struktur.

Plattformunabhängige Mobile-App-Entwicklung bietet erhebliche Vorteile in Bezug auf Effizienz und Reichweite, birgt jedoch Herausforderungen in der Umsetzung. Die Wahl des geeigneten Frameworks und die Nutzung eines Designsystems sind entscheidend für den Erfolg solcher Projekte.

### Flutter für plattformunabhängige Entwicklungen

* Vorteile und Einschränkungen bei der Nutzung von Flutter für barrierefreie Apps

### Tensorflow Lite für Machine Learning auf Mobilen Geräten

* Grundlagen und Vorteile der Verwendung von LiteRT / TFLite auf mobilen Geräten
* Grundlagen Navigation blinder und sehbeeinträchtigter Personen durch Hinderniserkennung über Drohne (nur UI-Perspektive)

## Konzept

### Anforderungen an die mobile App

* ausgedrückt durch (konkrete) Personas, (konkrete) User Journey) und Featureliste

### Aufbau und Struktur der App

* führt Schritt für Schritt zu den UI-Screens
* Design System (konkretisieren)
* Benutzerführung / PageFlow (konkret die wichtigsten Zusammenhänge der UI-Screens bzw. Pfade durch die App)
* Die wichtigsten Screens zur Kernfunktionalität (mit 2.3.1 Anforderungen/Hauptfunktionalitäten/Use Cases und 2.3.3 Schnittstellen gut abstimmen -> das muss "als Bild" alles zusammenpassen)

### Schnittstellen

* Kurze/grobe Beschreibungen der Schnittstellen und wichtigsten Entscheidungen, vllt auch teilweise tabellarisch darstellen und Text eher nicht-technisch halten -> die technischen Aspekte kommen dann bei den anderen beiden DA-Teilen - du bereitest hier nur den Rahmen / die Schnittstellen vor und hältst diesen Rahmen im Zuge des DA-Projektes zusammen
* Beschreibung, Entscheidungen und Konzept Navigationsprotokoll "App zu Benutzer" (Benutzernavigation durch App, Navigationshinweise: Bereitstellung von Informationen/Hinweisen/Warnungen zu Zielen, Orientierungspunkten, Richtungen und Entfernungen)
* Beschreibung und Entscheidungen "App zu Drohne" (App mit Drohne verbinden)
* Beschreibung und Entscheidungen "App zu KI-Funktion" (App mit KI-Funktion verbinden)
* Beschreibung und Entscheidungen "App zu Backend" (App mit Backend für User- und Datenmanagement verbinden, nahtlose Integration der verschiedenen Typen von Daten zwischen App und Backend, hier klar und kurz beschreiben, welche Daten zu speichern sind)

### Maßnahmen in Flutter für Accessibility

* Zusammenfassung der Entscheidungen 2.3.4 Prozess und Projektsetup Android und iOS App mit Flutter (eine Codebasis)

## Herausforderungen der Umsetzung

### Barrierefreiheit in den Screens

* Technische Herausforderungen bei der Umsetzung barrierefreier Interfaces.
* Praktische Beispiele für Screenreader-Kompatibilität
* Wie werden Sensorinformationen der Drohne barrierefrei für sehbeeinträchtigte Nutzer dargestellt?

### Plattformunabhängige Nimbus-App

* Konfigurationsschritte (Was ich wirklich dafür tue)
* Schritte und Anforderungen für die Konfiguration und Entwicklung
* Wie die Hardware- und Firmware-Komponenten aus Thema 4 Schnittstellen bereitstellen, die ich in der App anbinden kann

### Überprüfung der Barrierefreiheit

* Checker verwenden, der sagt, wie gut man das barrierefrei gemacht hat für mobile Apps
* Darstellung von Schwachstellen und Lösungen

### Herausforderungen bei der Flutter-Integration

* Darstellung von Schwachstellen und Lösungen

### Tensorflow Lite und Barrierefreiheit

* Schwierigkeiten bei der Integration von ML-Modellen und deren Accessibility
* Wie wird sichergestellt, dass ML-Funktionen wie Hinderniserkennung oder autonome Steuerung reibungslos zwischen App und Drohne funktionieren?

## Fazit und Ausblick

# Einsatz von Künstlicher Intelligenz in Drohnen zur Hinderniserkennung für sehbeeinträchtigte Personen

Maximilian LANGER

## Einführung

Die Integration von künstlicher Intelligenz (KI) in Drohnensysteme hat in den letzten Jahren eine enorme Entwicklung durchlaufen. Mit Fortschritten in der Rechenleistung, den Algorithmen des maschinellen Lernens und der Verfügbarkeit großer Mengen an Trainingsdaten ist es heute möglich, Drohnen mit Fähigkeiten auszustatten, die früher undenkbar waren. Besonders vielversprechend ist der Einsatz solcher Technologien zur Unterstützung von sehbeeinträchtigten Personen. Dieser Ansatz verfolgt nicht nur einen technologischen Fortschritt, sondern auch eine gesellschaftliche Vision.

Durch Hinderniserkennung und Navigationshilfen, die auf hochentwickelten KI-Systemen basieren, wird die Mobilität und Sicherheit dieser Bevölkerungsgruppe signifikant verbessert. Solche Systeme können dabei helfen, alltägliche Hürden zu überwinden und komplexe Umgebungen autonom zu navigieren. Die drohnenbasierte Assistenz bietet eine innovative und kosteneffiziente Alternative zu bestehenden Hilfsmittel wie Blindenhunden oder Bodenleitsystemen, die in vielen Szenarien Einschränkungen aufweisen.

In der folgenden Abhandlung werden sowohl die theoretischen Grundlagen als auch die praktischen Anwendungen der beschriebenen Technologien eingehend untersucht. Der Fokus liegt darauf, die Implementierung von KI-Modellen in Drohnensystemen detailliert zu analysieren, die damit verbundenen technischen Herausforderungen zu identifizieren und Perspektiven für eine Weiterentwicklung dieser Systeme aufzuzeigen.

Zentrale Aspekte dieser Untersuchung umfassen die technische Machbarkeit, die Integration von Softwarekomponenten sowie die kontinuierliche Optimierung der Algorithmen, um höchste Funktionalität und Zuverlässigkeit zu gewährleisten. Ein besonderes Augenmerk liegt darauf, die wissenschaftlichen und technologischen Grundlagen dieser Innovationen umfassend zu dokumentieren und deren potenziellen Beitrag zur gesellschaftlichen Verbesserung aufzuzeigen.

Der technologische Fortschritt hat nicht nur die Entwicklung kompakter und leistungsfähiger Hardware ermöglicht, sondern auch den Einsatz von Algorithmen des maschinellen Lernens zur Analyse und Verarbeitung von Umgebungsdaten revolutioniert. Durch die Kombination von Sensoren, Kameras und fortschrittlicher Software können Drohnen heute Aufgaben wie die Erkennung und Umgehung von Hindernissen autonom ausführen. Diese Fähigkeit wird insbesondere für sehbeeinträchtigte Personen von unschätzbarem Wert sein, da sie eine sicherere und selbstbestimmtere Teilnahme am öffentlichen Leben ermöglicht.

## Theoretische Grundlagen

### Künstliche Intelligenz und Maschinelles lernen

Künstliche Intelligenz (KI) beschreibt ein Gebiet der Technologie, das sich mit der Entwicklung von Systemen beschäftigt, die Aufgaben bewältigen können, die typischerweise menschliche Intelligenz erfordern. Dazu gehören das Verarbeiten von Wahrnehmungsdaten, das Treffen von Entscheidungen auf der Grundlage komplexer Informationen, das Lösen schwieriger Probleme und das Erlernen neuer Fähigkeiten durch Analyse und Erfahrung.

Ein zentraler Bestandteil der KI ist das Maschinelle Lernen (ML), ein Ansatz, bei dem Algorithmen aus bestehenden Daten Muster extrahieren und eigenständige Vorhersagen oder Entscheidungen treffen können, ohne explizit dafür programmiert zu sein. Dies geschieht durch iterative Optimierungsprozesse, bei denen Modelle trainiert werden, um die Beziehungen zwischen Eingaben und Ausgaben zu erkennen und zu verallgemeinern. ML ermöglicht es Systemen, flexibel auf neue Daten zu reagieren und sich kontinuierlich zu verbessern.

In der Hinderniserkennung werden häufig neuronale Netze eingesetzt, die durch ihre Fähigkeit, komplexe Muster in großen Datenmengen zu erkennen, überzeugen. Ein spezieller Typ von neuronalen Netzen, die sogenannten Convolutional Neural Networks (CNNs), hat sich als besonders geeignet für die Verarbeitung und Analyse visueller Daten erwiesen. CNNs bestehen aus mehreren Schichten, die jeweils unterschiedliche Aspekte der Eingabebilder analysieren. Die ersten Schichten erkennen grundlegende Merkmale wie Kanten und Texturen, während spätere Schichten komplexere Muster wie Formen und Objekte identifizieren. Dieses hierarchische Verfahren ermöglicht eine hochgradig präzise Objekterkennung.

Der Lernprozess eines CNNs basiert auf einem umfangreichen Training mit gekennzeichneten Datensätzen. Während des Trainings passt das Netzwerk seine Gewichte an, um die Beziehung zwischen den Eingabedaten und den erwarteten Ausgaben zu optimieren. Ein gut trainiertes CNN kann dadurch Hindernisse wie Straßenschilder, Bäume oder andere Objekte in Echtzeit identifizieren und klassifizieren. Diese Echtzeit-Fähigkeit ist entscheidend für Drohnen, die in sich dynamisch ändernden Umgebungen operieren und sofortige Entscheidungen treffen müssen. Darüber hinaus können durch Techniken wie Transfer Learning bereits vortrainierte Netzwerke an spezifische Anwendungsfälle angepasst werden, was die Entwicklungszeit und den Ressourcenaufwand erheblich reduziert.

### Bilderkennung in echtzeit

Die Bilderkennung in Echtzeit stellt eine der Kernkomponenten moderner KI-Systeme dar. Sie basiert auf komplexen Algorithmen, die darauf ausgelegt sind, kontinuierlich visuelle Datenströme von Kameras in einer hohen Geschwindigkeit und Genauigkeit zu analysieren. Diese Technologie ist insbesondere für Drohnen essenziell, da sie in dynamischen und unvorhersehbaren Umgebungen operieren müssen. Hierbei werden verschiedene Ansätze und Modelle verwendet, um eine präzise Objekterkennung und -klassifikation zu ermöglichen. Dabei hängt die Latenz, also die Verzögerung zwischen Eingabe und Ausgabe, auch stark von der Größe des verwendeten Modells ab, wie in der Grafik zu sehen ist. Größere Modelle wie YOLO11x bieten eine höhere Genauigkeit (mAP), haben jedoch auch eine höhere Latenzzeit im Vergleich zu kleineren Modellen wie YOLO11n. Diese Balance zwischen Geschwindigkeit und Genauigkeit ist entscheidend für die Wahl des richtigen Modells in Echtzeitanwendungen. Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte Beschreibung

Ein prominentes Beispiel ist das Modell YOLO (You Only Look Once), das auf Geschwindigkeit und Effizienz optimiert ist. Es verarbeitet ein komplettes Bild in einem einzigen Durchlauf und identifiziert dabei mehrere Objekte gleichzeitig. Diese Methode reduziert die Rechenlast erheblich, was besonders in ressourcenbeschränkten Umgebungen wie Drohnenanwendungen von Vorteil ist. YOLO verwendet Ankerboxen und Vorhersagen, um Objekte in verschiedenen Größen und Positionen zu erkennen, und bietet dabei eine Balance zwischen Genauigkeit und Geschwindigkeit.

Ein weiterer wesentlicher Aspekt der Echtzeit-Bilderkennung ist die Verarbeitung von Videodatenströmen. Drohnen erfassen kontinuierlich Daten, die analysiert werden müssen, um potenzielle Hindernisse wie Bäume, Straßenschilder oder Fußgänger zu erkennen. Diese Daten werden durch vortrainierte neuronale Netzwerke wie Convolutional Neural Networks (CNNs) verarbeitet, die speziell darauf trainiert wurden, Muster und Objekte in Bilddaten zu identifizieren.

Darüber hinaus wird die Echtzeit-Bilderkennung durch Hardwarebeschleuniger wie GPUs (Graphical Processing Units) oder TPUs (Tensor Processing Units) unterstützt, die die parallele Verarbeitung von Daten ermöglichen. Dies sorgt dafür, dass die Drohne auch in Szenarien mit hohem Datenaufkommen effektiv agieren kann. Zudem spielen Optimierungsansätze wie Quantisierung und Modellkompression eine wichtige Rolle, um die Berechnungen auf der Hardware der Drohne zu beschleunigen, ohne dabei die Genauigkeit zu beeinträchtigen.

Die Kombination dieser Technologien erlaubt es der Drohne, potenzielle Gefahren sofort zu identifizieren und in Echtzeit darauf zu reagieren. Beispielsweise kann sie automatisch ihre Flugbahn anpassen, um Kollisionen zu vermeiden, oder akustische und visuelle Warnsignale an den Benutzer senden. Solche Funktionen tragen wesentlich dazu bei, die Sicherheit und Zuverlässigkeit der Drohne in realen Anwendungsszenarien zu gewährleisten

### Trainingsdatensätze und Training

Trainingsdatensätze sind von zentraler Bedeutung für den Erfolg jedes maschinellen Lernmodells. Die Qualität und Vielfalt dieser Datensätze beeinflusst direkt die Fähigkeit des Modells, präzise und allgemeingültige Vorhersagen zu treffen. Um die Drohne optimal auf ihre Aufgabe vorzubereiten, müssen die Datensätze verschiedene Umgebungen und Hindernisse abbilden, die in realen Einsatzszenarien auftreten können. Dazu gehören sowohl statische Hindernisse wie Straßenschilder, Bäume und Mauern als auch dynamische Elemente wie Fußgänger, Fahrzeuge oder Tiere.

Die Erstellung dieser Datensätze ist ein mehrstufiger Prozess, der eine sorgfältige Planung und Ausführung erfordert. Plattformen wie RoboFlow bieten eine strukturierte Umgebung zur Annotation, Verwaltung und Optimierung von Datensätzen. Hier können Bilder gesammelt, beschriftet und nach spezifischen Kategorien wie "Straßenhindernisse" oder "Gefährliche Objekte" sortiert werden. RoboFlow ermöglicht darüber hinaus die Generierung erweiterter Datensätze durch Datenaugmentation, bei der bestehende Bilder durch Rotation, Skalierung, Farbveränderungen und andere Techniken manipuliert werden, um die Vielfalt der Daten zu erhöhen.

Das Training eines Modells umfasst drei Hauptphasen, die durch den Einsatz von Epochen und Batches unterstützt werden. Eine **Epoche** bezeichnet einen vollständigen Durchlauf durch den gesamten Trainingsdatensatz. Innerhalb einer Epoche wird das Modell auf jede Instanz der Daten trainiert, was hilft, das Modell kontinuierlich zu verbessern. Allerdings ist das Verarbeiten eines großen Datensatzes in einem einzigen Schritt oft nicht praktikabel, da es hohe Speicher- und Rechenressourcen erfordert. Daher werden die Daten in kleinere **Batches** aufgeteilt. Ein **Batch** ist eine Teilmenge des gesamten Datensatzes, die während eines einzelnen Trainingsschritts verarbeitet wird. Die Verwendung von Batches beschleunigt den Trainingsprozess, da das Modell schneller aktualisiert werden kann, und hilft, Speicherbeschränkungen zu überwinden. Darüber hinaus ermöglicht die Batch-Verarbeitung die Glättung der Gradientenänderungen, was zu stabileren Optimierungsprozessen führt. Die Kombination aus mehreren Epochen und der Batch-Verarbeitung sorgt dafür, dass das Modell iterativ lernt und seine Vorhersagefähigkeit sukzessive verbessert.

Das Training eines Modells umfasst dabei drei Hauptphasen:

1. **Datenvorverarbeitung:** In diesem Schritt werden die Rohdaten aufbereitet, um sicherzustellen, dass sie für das Modell verwendbar sind. Dies beinhaltet die Normalisierung der Bilddaten, das Entfernen von Rauschen und die Skalierung der Bilder auf eine einheitliche Größe. Zusätzlich wird darauf geachtet, dass die Daten hinsichtlich ihrer Klassenverteilung ausgeglichen sind, um ein Ungleichgewicht zu vermeiden, das zu verzerrten Vorhersagen führen könnte. Tools wie „cvat.ai“ helfen dabei.
2. **Modellentwicklung:** Hierbei wird ein neuronales Netzwerk erstellt und initialisiert. Moderne Ansätze nutzen oft Transfer Learning, bei dem ein vortrainiertes Modell wie YOLO oder MobileNet auf die spezifische Anwendung angepasst wird. Diese Methode reduziert die Trainingszeit erheblich und verbessert die Genauigkeit, da das vortrainierte Modell bereits allgemeine Merkmale erkannt hat.
3. **Modelloptimierung:** Das Modell wird durch wiederholte Iterationen verbessert. Dies geschieht durch die Minimierung eines Fehlers, der mithilfe einer Loss-Funktion berechnet wird. Algorithmen wie Stochastic Gradient Descent (SGD) oder Adam werden eingesetzt, um die Gewichte des Modells schrittweise anzupassen. Während dieses Prozesses werden auch Techniken wie Early Stopping oder Cross-Validation angewendet, um Überanpassung (Overfitting) zu vermeiden.

Darüber hinaus ist die kontinuierliche Aktualisierung der Trainingsdaten entscheidend, da die Drohne mit neuen oder sich verändernden Umgebungen konfrontiert wird. Die Implementierung eines aktiven Lernansatzes, bei dem das Modell Feedback von realen Einsätzen verwendet, kann die Genauigkeit und Zuverlässigkeit weiter verbessern. Solche Ansätze stellen sicher, dass das System robust bleibt und sich an neue Herausforderungen anpassen kann, wodurch die Sicherheit und Effizienz des Drohnenbetriebs gewährleistet wird.

## Konzept

### Systemarchitektur

Das Herzstück der Systemarchitektur der Drohne ist das KI-Modell, das in unserem Fall mithilfe von YOLO11x entwickelt wurde. Dieses Modell wird mit benutzerdefinierten Datensätzen trainiert, die von der Plattform RoboFlow bereitgestellt werden. Nach dem Training wird das Modell in das TensorFlow Lite (TFLite)-Format konvertiert, um es auf der Drohne effizient auszuführen. Die gesamte Entwicklung und das Training des Modells erfolgen lokal auf einem privaten Rechner mit NVIDIA-Grafikkartenunterstützung, wobei CUDA (Compute Unified Device Architecture) genutzt wird.

**1. Datensatzerstellung mit RoboFlow**

RoboFlow bietet eine umfassende Plattform zur Erstellung und Verwaltung von Trainingsdatensätzen:

* **Annotation:** Bilder werden gesammelt und präzise annotiert, um spezifische Hindernisse wie Straßenschilder, Fahrzeuge, Bäume oder Fußgänger zu markieren. Diese Annotationen bilden die Grundlage für ein genaues Training des Modells.
* **Datenaugmentation:** RoboFlow ermöglicht die Erweiterung der Datensätze durch Techniken wie Rotation, Skalierung, Farbmodifikation und Zufügung von Rauschen. Diese Schritte erhöhen die Vielfalt der Daten und verbessern die Generalisierungsfähigkeit des Modells.
* **Export und Integration:** Die fertig aufbereiteten Datensätze werden in einem YOLO-kompatiblen Format exportiert und für den Trainingsprozess bereitgestellt.

**2. Training des Modells mit YOLO11x**

YOLO11x ist ein modernes Framework für Objekterkennung, das für hohe Geschwindigkeit und Genauigkeit optimiert ist. Der Trainingsprozess umfasst folgende Schritte:

* **Architektur:** YOLO11x verwendet eine tiefenoptimierte Architektur mit Convolutional Neural Networks (CNNs), um räumliche und kontextuelle Merkmale der Bilder zu extrahieren.
* **Hyperparameter:** Kritische Parameter wie Lernrate, Batchgröße und Epochenanzahl werden fein abgestimmt, um das Modell optimal zu trainieren.
* **Transfer Learning:** Vortrainierte Gewichte können verwendet werden, um die Trainingszeit zu reduzieren und die Genauigkeit zu verbessern.
* **Evaluierung:** Nach Abschluss des Trainings wird das Modell mit separaten Testdatensätzen evaluiert, um sicherzustellen, dass es in verschiedenen Umgebungen robust funktioniert.

**3. Nutzung von CUDA für beschleunigtes Training**

Der Trainingsprozess wird auf einem privaten Rechner mit NVIDIA-Grafikkarten durchgeführt, wobei CUDA (Compute Unified Device Architecture) genutzt wird. CUDA ist eine parallele Rechenplattform und Programmierschnittstelle, die von NVIDIA entwickelt wurde, um die leistungsstarke Verarbeitung von Deep-Learning-Modellen zu ermöglichen.

* **Voraussetzungen:**
  + Eine NVIDIA-GPU mit CUDA-Unterstützung (z. B. RTX-Serie oder Quadro-GPUs).
  + Installation der CUDA-Toolkits und zugehöriger Treiber.
  + Integration von Frameworks wie TensorFlow oder PyTorch, die CUDA für die Beschleunigung nutzen.
* **Vorteile der CUDA-Nutzung:**
  + **Schnellere Verarbeitung:** GPUs bieten durch paralleles Rechnen eine deutlich schnellere Verarbeitung im Vergleich zu CPUs, insbesondere bei großen Datenmengen und komplexen neuronalen Netzwerken.
  + **Effiziente Ressourcennutzung:** CUDA optimiert die Auslastung der GPU und reduziert die Trainingszeit erheblich.
  + **Flexibilität:** Der Trainingsprozess kann an die Hardware angepasst werden, sodass die maximale Leistung der GPU genutzt wird.
* **Ablauf:**
  + Daten werden in Batches auf die GPU geladen.
  + Das Modell wird schrittweise optimiert, indem die Gewichte mit Algorithmen wie Adam oder Stochastic Gradient Descent (SGD) angepasst werden.
  + CUDA-Bibliotheken wie cuDNN (CUDA Deep Neural Network) werden für die Beschleunigung der Convolutional-Schichten genutzt.

**4. Konvertierung in TFLite**

Nach dem erfolgreichen Training wird das Modell in das TFLite-Format konvertiert, um es auf ressourcenbeschränkten Geräten wie der Drohne auszuführen:

* **Optimierung:**
  + Die Größe des Modells wird reduziert, indem Parameter quantisiert und unnötige Schichten entfernt werden.
  + Diese Optimierungen stellen sicher, dass das Modell auch auf Hardware mit geringer Rechenleistung effizient ausgeführt werden kann.
* **Integration:** Das TFLite-Modell wird direkt auf die Drohne geladen, wo es die Echtzeitanalyse der Bilddaten übernimmt.

**5. Integration in die Drohnenarchitektur**

Das trainierte und optimierte Modell wird in die Drohnenarchitektur integriert, die aus mehreren Subsystemen besteht:

* **Kameras:** Hochauflösende Kameras erfassen kontinuierlich Bilder der Umgebung.
* **Prozessor:** Ein leistungsstarker Prozessor analysiert die Bilddaten mithilfe des TFLite-Modells in Echtzeit.
* **Steuerungssystem:** Basierend auf den Ergebnissen des Modells passt das Steuerungssystem die Flugbahn der Drohne an, um Hindernissen auszuweichen.
* **Feedback-System:** Warnungen oder Hinweise werden über akustische Signale oder Vibrationen an den Benutzer weitergegeben.

**6. Vorteile der Systemarchitektur**

* **Effizienz:** Durch die Nutzung von CUDA und TFLite werden sowohl die Trainingszeit als auch die Ausführungszeit des Modells optimiert.
* **Flexibilität:** Die modulare Architektur ermöglicht einfache Updates und Anpassungen des Modells.
* **Robustheit:** Die Verwendung von benutzerdefinierten Datensätzen und einem leistungsstarken Framework wie YOLO11x gewährleistet eine hohe Genauigkeit und Zuverlässigkeit.

Durch diese Kombination aus modernster Technologie und optimierter Entwicklung bietet die Drohne eine leistungsfähige und effiziente Lösung für die Hinderniserkennung im Alltag.

Die Systemarchitektur stellt sicher, dass alle Komponenten effizient zusammenarbeiten, um eine sichere und effektive Navigation der Drohne zu gewährleisten. Durch den modularen Aufbau können zukünftige Erweiterungen und Verbesserungen problemlos integriert werden, wodurch die Drohne anpassungsfähig und zukunftssicher bleibt.

### Verwendung von Tensorflow Lite für die Bilderkennung

TensorFlow Lite ist eine speziell entwickelte, kompakte Version von TensorFlow, die für den Einsatz auf mobilen und eingebetteten Geräten optimiert wurde. Diese Optimierung reduziert den Ressourcenbedarf erheblich und ist daher ideal für Drohnenanwendungen geeignet, bei denen Rechenleistung und Energieverbrauch begrenzt sind. Das Framework unterstützt die Ausführung von vortrainierten Modellen direkt auf Edge-Geräten wie Drohnenprozessoren oder mobilen Geräten, wodurch die Notwendigkeit entfällt, Daten an externe Server zu senden. Dies minimiert Latenzzeiten und erhöht gleichzeitig die Zuverlässigkeit und Geschwindigkeit der Entscheidungen.

Der Prozess zur Integration in TensorFlow Lite beginnt mit der Konvertierung des trainierten KI-Modells aus einem Standardformat wie TensorFlow oder PyTorch. Dabei werden Optimierungstechniken wie Quantisierung angewendet, um die Modellgröße zu reduzieren, ohne die Genauigkeit wesentlich zu beeinträchtigen. Quantisierung wandelt beispielsweise 32-Bit-Fließkommazahlen in 8-Bit-Integer um, was sowohl den Speicherbedarf als auch die Berechnungsanforderungen drastisch verringert.

Ein weiterer Vorteil von TensorFlow Lite ist die Unterstützung von Hardwarebeschleunigung, wie sie durch TPUs (Tensor Processing Units) oder GPUs (Graphical Processing Units) bereitgestellt wird. Diese Hardwareintegration ermöglicht eine hochperformante Echtzeitverarbeitung, was für Aufgaben wie die Hinderniserkennung entscheidend ist. Zusätzlich können Entwickler benutzerdefinierte Operatoren und Delegates verwenden, um die Effizienz weiter zu steigern und die Funktionalität an spezifische Anforderungen der Drohne anzupassen.

## Herausforderungen der Umsetzung

### Trainingsdatensätze und Modellgenauigkeit

Eine der größten Herausforderungen bei der Entwicklung eines präzisen KI-Modells ist die Sicherstellung seiner Genauigkeit, die stark von der Qualität und Vielfalt der verwendeten Trainingsdaten abhängt. Trainingsdatensätze sind die Grundlage für das maschinelle Lernen und bestimmen, wie gut ein Modell verschiedene Szenarien abbilden und Vorhersagen treffen kann. Die Erstellung solcher Datensätze ist ein komplexer Prozess, der mehrere wichtige Aspekte umfasst:

1. **Arten von Trainingsdatensätzen:**
   * **Statische Datensätze:** Diese umfassen Bilder oder Videos von stationären Objekten wie Straßenschildern, Bäumen oder Mauern. Sie werden verwendet, um das Modell auf Hindernisse vorzubereiten, die in urbanen oder natürlichen Umgebungen auftreten können.
   * **Dynamische Datensätze:** Hierbei handelt es sich um Szenarien mit beweglichen Elementen wie Fußgängern, Fahrzeugen oder Tieren. Diese sind entscheidend, um das Modell auf Echtzeit-Interaktionen vorzubereiten.
   * **Edge-Fälle:** Spezifische Szenarien, die selten auftreten, wie Hindernisse bei schlechten Lichtverhältnissen, Regen oder Schnee. Diese Datensätze verbessern die Robustheit des Modells.
   * **Simulierte Datensätze:** Daten, die in virtuellen Umgebungen generiert werden, um Szenarien zu simulieren, die in der realen Welt schwer nachzustellen sind. Simulierte Daten können ergänzend eingesetzt werden, um das Spektrum der Trainingsdaten zu erweitern.
2. **Erstellung und Annotation der Datensätze:**
   * **Erfassung:** Bilder und Videos müssen in unterschiedlichen Umgebungen, unter variierenden Licht- und Wetterbedingungen gesammelt werden, um eine breite Abdeckung sicherzustellen.
   * **Annotation:** Jedes Objekt im Datensatz wird von menschlichen Annotatoren markiert und klassifiziert. Dies umfasst das Umreißen von Objekten mit Bounding-Boxen oder Masken, sowie das Hinzufügen von Labels wie "Person", "Auto" oder "Straßenschild".

Ein Bild, das Fahrzeug, Landfahrzeug, draußen, Rad enthält.

Automatisch generierte Beschreibung

* + **Datenaugmentation:** Um die Vielfalt der Daten zu erhöhen, werden Techniken wie Rotationen, Skalierungen, Farbmodifikationen und Hinzufügung von Rauschen angewendet. Dies hilft dem Modell, robustere Merkmalsmuster zu lernen.

1. **Bestimmung der Modellgenauigkeit:**
   * **Metriken zur Evaluierung:**
     + **Präzision und Recall:** Diese Metriken bewerten, wie gut das Modell Objekte erkennt, ohne falsche Positiv- oder Negativerkennungen zu erzeugen.
     + **mAP (mean Average Precision):** Diese umfassende Metrik wird für die Bewertung von Objekterkennungsmodellen wie YOLO verwendet und gibt die durchschnittliche Genauigkeit über verschiedene Klassen und Schwellenwerte hinweg an.
     + **F1-Score:** Dieser Score kombiniert Präzision und Recall, um die Gesamtgenauigkeit des Modells darzustellen.
   * **Testdatensätze:** Separate Datensätze, die nicht im Training verwendet wurden, dienen der Evaluierung des Modells unter realistischen Bedingungen.
2. **Kontinuierliche Überwachung und Aktualisierung:**
   * **Feedback-Schleifen:** Daten aus realen Einsätzen der Drohne werden gesammelt und analysiert, um die Leistung des Modells zu bewerten. Fehler oder Schwächen können identifiziert und durch erneutes Training behoben werden.
   * **Aktives Lernen:** Dieser Ansatz priorisiert die Annotation neuer Daten, die für das Modell besonders schwierig oder uneindeutig sind. Dadurch wird das Modell gezielt verbessert.
   * **Versionierung:** Jedes neue Modell und seine Trainingsdaten werden versioniert, um den Fortschritt zu dokumentieren und gegebenenfalls zu einem vorherigen Zustand zurückzukehren.
3. **Ressourcen und Herausforderungen:**
   * **Zeit und Kosten:** Die Erstellung hochwertiger Trainingsdatensätze erfordert erhebliche Ressourcen, sowohl für die Datenerfassung als auch für die Annotation. Automatisierte Tools können den Prozess beschleunigen, erfordern jedoch eine sorgfältige Überprüfung.
   * **Ethik und Datenschutz:** Bei der Verwendung von Daten mit Personenbezug müssen Datenschutzgesetze wie die DSGVO eingehalten werden. Daten sollten anonymisiert und sicher gespeichert werden.

Durch den gezielten Einsatz qualitativ hochwertiger und vielseitiger Trainingsdatensätze sowie durch die kontinuierliche Überwachung der Modellgenauigkeit kann die Drohne ihre Hinderniserkennungsfähigkeiten stetig verbessern und an neue Herausforderungen angepasst werden.

## Fazit und Ausblick

Der Einsatz von künstlicher Intelligenz (KI) in Drohnen stellt einen Meilenstein dar, um das Leben sehbeeinträchtigter Menschen signifikant zu verbessern. Durch die innovative Kombination von Algorithmen zur Echtzeit-Bilderkennung, leistungsstarken KI-Modellen und hochwertigen Trainingsdatensätzen können Drohnen eine revolutionäre Lösung für die Hinderniserkennung bieten. Diese Technologie ermöglicht nicht nur eine präzise Analyse von Umgebungen, sondern bietet auch die Flexibilität, sich dynamisch an verschiedene Szenarien anzupassen.

Der Schlüssel liegt in der kontinuierlichen Weiterentwicklung der Algorithmen und der Optimierung der Systemarchitektur. Zukünftige Arbeiten sollten sich darauf konzentrieren, die Effizienz der Modelle weiter zu steigern, etwa durch fortschrittliche Quantisierungsmethoden oder die Integration neuer Hardwarebeschleuniger wie TPUs. Zudem könnten benutzerdefinierte Datensätze, die spezifisch für individuelle Benutzerbedürfnisse erstellt werden, die Personalisierung der Drohnenfunktionen erheblich verbessern.

Darüber hinaus wird die Erweiterung der Funktionalität ein zentraler Faktor sein. Dazu gehören die Integration zusätzlicher Sensorik wie Lidar oder Infrarot für eine noch robustere Hinderniserkennung sowie die Entwicklung von Schnittstellen zu externen Systemen, um die Drohnen in Smart-City-Infrastrukturen einzubinden. Die kontinuierliche Überwachung und das Feedback aus realen Einsatzszenarien werden dabei helfen, Schwächen zu identifizieren und gezielt zu beheben.

Langfristig könnte diese Technologie nicht nur die Mobilität und Sicherheit sehbeeinträchtigter Menschen verbessern, sondern auch in anderen Bereichen wie der Logistik, im Katastrophenschutz oder in der urbanen Verkehrsüberwachung eingesetzt werden. Mit der richtigen Kombination aus Innovation, Forschung und Nutzerfeedback hat die KI-gestützte Drohnentechnologie das Potenzial, eine wesentliche Rolle in einer inklusiven und vernetzten Zukunft Entwicklung der Drohnenhardware sowie -Firmware

# 

# Entwicklung der Drohnenhardware sowie -Firmware

## Einführung

Die Entwicklung der Drohnenhardware und -Firmware bildet das technische Fundament für das Projekt Nimbus. Dieses Kapitel fokussiert sich auf die wesentlichen technischen Komponenten und deren Interaktion, um eine präzise Navigation sowie eine zuverlässige Hinderniserkennung zu gewährleisten. Ziel ist es, eine Hardwareplattform zu schaffen, die den spezifischen Anforderungen von blinden und sehbeeinträchtigten Personen gerecht wird und gleichzeitig höchste Sicherheitsstandards einhält.

Die Auswahl der Hardwarekomponenten, einschließlich der Drohnenplattform, Sensorik und Mikrocontroller, wurde sorgfältig auf die Herausforderungen des Projekts abgestimmt. Besondere Berücksichtigung fanden dabei Aspekte wie Robustheit, Energieeffizienz und Modularität, um eine flexible und nachhaltige Entwicklung zu ermöglichen. Durch den Einsatz fortschrittlicher Technologien wie LIDAR-Sensoren, hochauflösenden Kameras und GPS-Systemen wird eine präzise Umgebungserkennung und -analyse ermöglicht.

Ein weiterer Fokus dieses Kapitels liegt auf der Firmware-Entwicklung, die eine nahtlose Integration der Hardwarekomponenten mit der Navigations- und Hinderniserkennungssoftware sicherstellt. Die Firmware übernimmt essenzielle Aufgaben wie die Verarbeitung von Sensordaten, die Steuerung der Motoren sowie die Kommunikation mit der mobilen App. Durch den Einsatz bewährter Entwicklungsframeworks und kontinuierlicher Tests wird eine zuverlässige und skalierbare Systemarchitektur geschaffen.

## Theoretische Grundlagen

### Ein Bild, das medizinische Ausrüstung, Projektor enthält. Automatisch generierte BeschreibungDrohnenplattformen

**Tello Drohne:**

Die Tello Drohne ist eine kompakte, leichtgewichtige Drohne mit einer Flugzeit von etwa 13 Minuten. Ihre einfache Bedienbarkeit und kostengünstige Hardware machen sie zu einer beliebten Wahl für Einsteiger und Bildungszwecke. Die Tello kann über ein Smartphone oder Controller angesteuert werden und bietet grundlegende Programmiermöglichkeiten über die Tello SDK-Schnittstelle. Sie verfügt jedoch weder über GPS noch fortschrittliche Sensorik, was ihre Eignung für autonome und präzise Navigation stark einschränkt. Aufgrund der begrenzten Ein Bild, das Spielzeug, Flugzeug enthält.

Automatisch generierte BeschreibungModularität ist sie für Projekte mit spezifischen technischen Anforderungen weniger geeignet.

**QAV250 mit Pixhawk 6C Mini:**

Der QAV250 ist eine modulare Plattform, die mit dem leistungsfähigen Pixhawk 6C Mini Flugcontroller kombiniert wird. Diese Konfiguration ermöglicht maximale Anpassbarkeit, von der Integration fortschrittlicher Sensoren wie LiDAR und Kameramodule bis hin zur Erweiterung mit GPS-Modulen. Der Pixhawk 6C Mini unterstützt ArduPilot und PX4, die Open-Source-Lösungen für Flugsteuerung und Missionsplanung. Diese bieten Unterstützung für Funktionen wie Wegpunkt-Navigation, „Return to Home“ und autonome Flüge. Die Steuerung erfolgt über MAVLink-Protokolle, was eine robuste und skalierbare Kommunikationsschnittstelle gewährleistet. Dank der Flexibilität des QAV250 können unterschiedliche Designs für spezifische Anforderungen realisiert werden, erfordert jedoch einen hohen Entwicklungsaufwand.

### Antriebssystem

Die Auswahl des geeigneten Antriebssystems ist entscheidend für die Leistung und Zuverlässigkeit einer Drohne. In der Regel werden bürstenlose Gleichstrommotoren (Brushless DC-Motoren) eingesetzt, die für ihre Effizienz und Langlebigkeit bekannt sind.

#### Motoren: Unterschiede zwischen Brushed und Brushless Motoren

**Brushed Motoren**:   
Brushed Motoren nutzen mechanische Bürsten und einen Kommutator, um elektrische Energie an den Rotor zu übertragen. Diese Bauweise führt zu erhöhter Reibung und Wärmeentwicklung, was die Effizienz und Lebensdauer reduziert. Sie sind einfacher zu steuern und günstiger in der Herstellung, aber aufgrund des Verschleißes der Bürsten nicht für den dauerhaften oder intensiven Betrieb geeignet. Daher finden sie hauptsächlich in kostengünstigen Drohnen Anwendung.

**Brushless Motoren**:   
Brushless Motoren verwenden Permanentmagnete im Rotor und ersetzen die mechanischen Bürsten durch elektronische Steuerungen (ESCs). Dies reduziert den Verschleiß und steigert die Effizienz erheblich. Brushless Motoren bieten eine bessere Leistung, eine längere Lebensdauer und arbeiten deutlich leiser. Aufgrund dieser Eigenschaften sind sie der Standard in modernen Drohnen und eignen sich besonders für präzise Steuerung sowie Anwendungen mit hohen Leistungsanforderungen.

#### Propeller: Typen und deren Eigenschaften

**Zwei-Blatt-Propeller**: Ein Bild, das Propeller enthält.

Automatisch generierte Beschreibung  
Diese Propeller sind effizient, da weniger Luftwiderstand erzeugt wird. Sie zeichnen sich durch eine höhere Energieeffizienz und eine geringere Lautstärke aus, was sie ideal für längere Flüge und leisere Einsätze macht.

**Drei-Blatt-Propeller**: Ein Bild, das Propeller, Schwarzweiß, monochrom enthält.

Automatisch generierte Beschreibung  
Mit einem zusätzlichen Blatt bieten sie mehr Schub und bessere Manövrierbarkeit. Allerdings erzeugen sie etwas mehr Luftwiderstand und sind daher weniger effizient als Zwei-Blatt-Propeller. Die Lautstärke ist etwas höher, das Geräuschprofil jedoch stabil. Sie werden häufig in FPV-Drohnen eingesetzt, die präzise Steuerung erfordern.

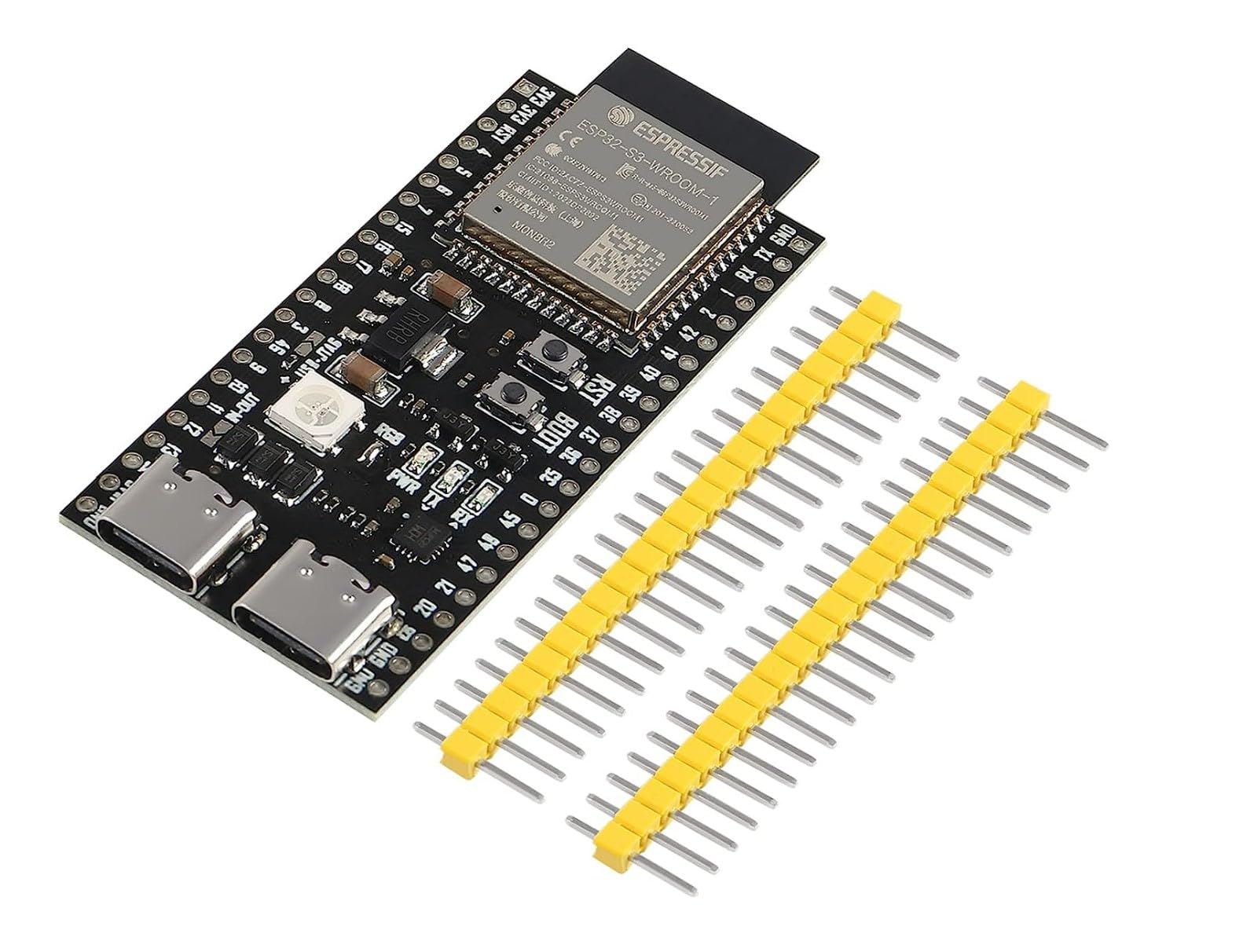
**Vier- oder Mehrblatt-Propeller**:Ein Bild, das Propeller enthält.

Automatisch generierte Beschreibung  
Diese Propeller erzeugen den höchsten Schub und sorgen für eine sehr stabile Fluglage, was besonders für Kameradrohnen nützlich ist. Allerdings sind sie weniger effizient und lauter als Varianten mit weniger Blättern.

### Microcontroller

Für die Hardware-Architektur von Nimbus wurde eine Analyse der verfügbaren Mikrocontroller durchgeführt, wobei sowohl der ESP32 als auch der Arduino evaluiert wurden. Der ESP32 wurde als Hauptkomponente ausgewählt, da er eine hohe Verarbeitungsleistung mit integrierter WLAN- und Bluetooth-Funktionalität kombiniert. Diese Eigenschaften sind entscheidend für die Echtzeit-Hinderniserkennung und die Kommunikation mit der mobilen App.

Der Arduino Uno wurde ebenfalls als Alternative betrachtet, bietet jedoch Einschränkungen hinsichtlich der Rechenleistung und drahtlosen Konnektivität. Für einfache Steuerungsaufgaben und Prototypen eignet sich der Arduino hervorragend, allerdings ist seine Erweiterbarkeit für anspruchsvollere Anwendungen wie maschinelles Lernen oder KI-basierte Navigation begrenzt.



Vorteile des ESP32 gegenüber Arduino:

* **Konnektivität**: Integriertes WLAN und Bluetooth (BLE), was eine nahtlose Kommunikation ermöglicht.
* **Rechenleistung**: Dual-Core-Architektur mit einer Taktfrequenz von bis zu 240 MHz, verglichen mit 16 MHz beim Arduino Uno.
* **Speicher**: Der ESP32 verfügt über 520 KB SRAM und ermöglicht den Anschluss an externen Flash-Speicher, während der Arduino Uno nur 2 KB SRAM bietet.
* **Vielseitigkeit**: Unterstützt eine breite Palette an Sensoren und Modulen, einschließlich Kameras und Ultraschallsensoren, was für die Hinderniserkennung essenziell ist.

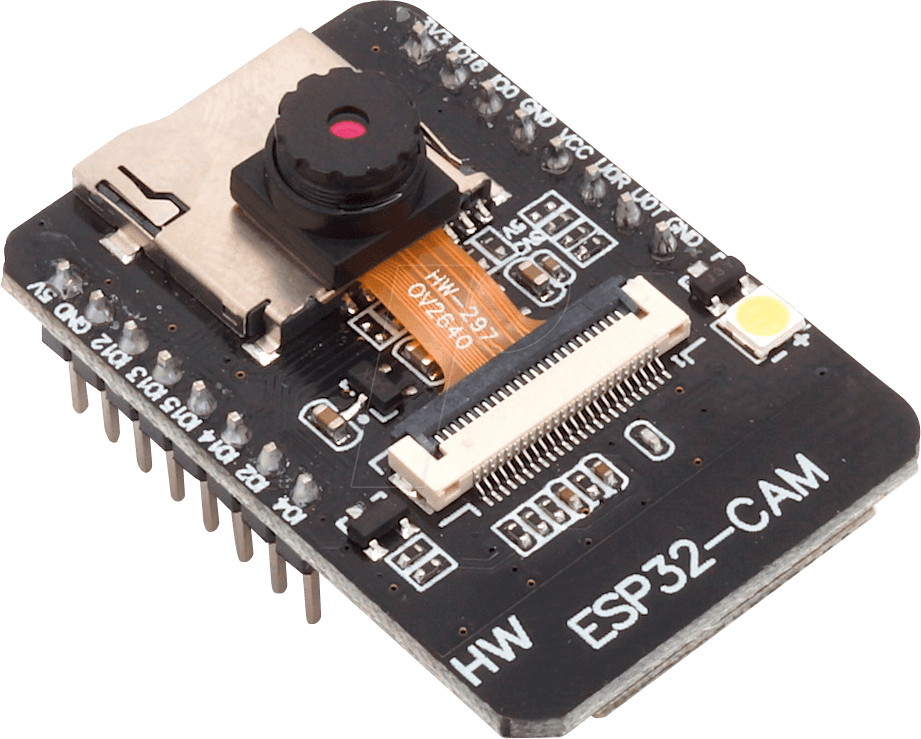
Der ESP32 wurde gewählt, da er höhere Rechenleistung, integriertes WLAN/BLE, ideal für Echtzeit-Hinderniserkennung und mobile Kommunikation. Seine Vielseitigkeit und Energieeffizienz übertreffen Alternativen wie den Arduino Uno, wodurch er optimal für die Anforderungen von Nimbus ist.

### Sensorik

**Ultraschallsensor** **Ein Bild, das Elektronik, Lautsprecher enthält.

Automatisch generierte Beschreibung**

Ultraschallsensoren nutzen Schallwellen, um Entfernungen zu messen. Sie senden Ultraschallimpulse aus, die von Hindernissen reflektiert werden. Die Zeit, die die Wellen für die Rückkehr benötigen, wird zur Entfernungsmessung verwendet. Diese Sensoren sind besonders nützlich für die Erkennung von Objekten in kurzen bis mittleren Entfernungen. In der Drohne dienen sie zur präzisen Navigation in engen Räumen und zur Hindernisvermeidung. Rechtlich ist sicherzustellen, dass die emittierten Schallfrequenzen den Gesundheitsstandards entsprechen und keine Störungen für Tiere verursachen.

**Kamera**

Kameras dienen als visuelle Sensoren, die Bilder und Videos aufnehmen, um Hindernisse zu erkennen und die Umgebung zu analysieren. Durch KI-basierte Bildverarbeitung, wie sie mit TensorFlow Lite in der Nimbus-Drohne realisiert wird, können Objekte wie Fußgänger oder Fahrzeuge identifiziert werden. Kameras sind besonders effektiv bei der Erkennung visueller Hinweise wie Straßenschilder. Datenschutzrechtlich muss sichergestellt werden, dass keine personenbezogenen Daten erfasst oder gespeichert werden, insbesondere in öffentlichen Bereichen.

Zusammen bieten diese Sensoren eine synergetische Grundlage für präzise Navigation, wobei sie sich in Funktion und Reichweite ergänzen. Rechtliche Rahmenbedingungen, wie die Einhaltung von Datenschutz- und Sicherheitsvorgaben, spielen eine zentrale Rolle bei der Implementierung.

### Kommunikationsmodule

Die Kommunikationsmodule sind ein zentraler Bestandteil der NIMBUS-Architektur, da sie die Interaktion zwischen Drohne, mobiler App und Peripheriegeräten wie dem ESP32 oder Pixhawk ermöglichen. Die Auswahl und Integration dieser Technologien gewährleistet die zuverlässige Übertragung von Daten in Echtzeit und optimiert die Benutzerfreundlichkeit.

**WLAN**

Das WLAN-Modul auf dem ESP32 dient der Datenübertragung mit höherer Bandbreite, z. B. für Bilddaten und umfangreiche Navigationsinformationen. Mithilfe von Wi-Fi Direct können Drohne und Smartphone direkt miteinander kommunizieren, ohne ein externes Netzwerk zu benötigen. Dies ermöglicht eine stabile Verbindung, selbst in Umgebungen ohne Internetzugang. Für komplexe Navigationsaufgaben und die Synchronisation von Daten mit dem Backend ist WLAN entscheidend, da es Echtzeitdatenübertragungen und Interaktionen mit der App unterstützt.

**MAVLink**

MAVLink (Micro Air Vehicle Communication Protocol) ist das Standardprotokoll für die Kommunikation zwischen Drohne und Flight Controller (z. B. Pixhawk). Es erlaubt den Austausch von Telemetriedaten, Befehlen und Missionsparametern. Die Integration von MAVLink ermöglicht Funktionen wie Echtzeit-Hinderniserkennung und autonome Navigation. Darüber hinaus unterstützt MAVLink die Verbindung mit Bodenstationen, wodurch weiterführende Kontrollmöglichkeiten, wie Waypoint-Missionen und Statusüberwachung, bereitgestellt werden.

WLAN und MAVLink bilden die zentrale Kommunikationsbasis von NIMBUS. WLAN ermöglicht die Echtzeitübertragung großer Datenmengen, während MAVLink präzise Steuerung und Telemetriedaten bereitstellt. Diese Kombination gewährleistet zuverlässige Navigation und Benutzerfreundlichkeit.

### Autonomes Fliegen

**Definition**

Autonomes Fliegen beschreibt die Fähigkeit eines unbemannten Luftfahrzeugs (UAV, engl. Unmanned Aerial Vehicle), komplexe Flugmanöver und Navigation ohne menschliche Steuerung auszuführen. Dies wird durch fortschrittliche Technologien wie Sensorfusion, Künstliche Intelligenz (KI) und präzise Steuerungsalgorithmen ermöglicht. Ziel ist es, dem Nutzer die direkte Steuerung abzunehmen und eine sichere, effiziente und unabhängige Fortbewegung des UAVs zu gewährleisten.

**Navigation**

Die Navigation eines autonomen UAVs erfolgt durch die Kombination verschiedener Technologien, um Position, Orientierung und Umgebung kontinuierlich zu analysieren und darauf basierend Entscheidungen zu treffen. Zwei wesentliche Methoden kommen dabei zum Einsatz:

* **GPS**

Das Global Positioning System (GPS) bildet die Grundlage für die Positionsbestimmung. Es ermöglicht der Drohne, ihre geographische Position und Höhe mit hoher Präzision zu bestimmen. GPS ist besonders in offenen, Outdoorumgebungen effektiv und wird durch unterstützende Systeme wie Differential-GPS ergänzt, um eine Genauigkeit im Zentimeterbereich zu erreichen.

* **SLAM**

Simultaneous Localization and Mapping (SLAM) wird eingesetzt, um in Echtzeit Karten der Umgebung zu erstellen und die Position der Drohne innerhalb dieser Karte zu bestimmen. Diese Technologie ist besonders nützlich in Umgebungen, in denen GPS-Signale schwach oder nicht verfügbar sind, wie z. B. in Innenräumen oder urbanen Canyons. SLAM kombiniert Daten aus verschiedenen Sensoren, darunter Kameras und LiDAR, und verarbeitet diese mit Hilfe von KI-Algorithmen.

**Tests in der Simulation**

Vor der physischen Erprobung in realen Umgebungen wird das autonome Flugverhalten umfassend in Simulationsumgebungen getestet. Diese Tests haben mehrere Ziele:

1. **Validierung der Navigationsalgorithmen**: Die Drohne wird virtuellen Szenarien ausgesetzt, um die Zuverlässigkeit der GPS- und SLAM-basierten Navigation zu prüfen.
2. **Gefahrenanalyse**: Potenzielle Sicherheitsrisiken wie Kollisionen oder Softwarefehler können identifiziert und behoben werden, bevor die Drohne in reale Szenarien eingeführt wird.
3. **Optimierung der Flugparameter**: Die Leistung der Drohne in unterschiedlichen Umgebungen wird simuliert, um ihre Flugparameter (z. B. Geschwindigkeit, Höhenkontrolle) zu optimieren​.

**Rechtliche Aspekte**

Der Betrieb autonomer Drohnen unterliegt strengen gesetzlichen Vorschriften, die den sicheren Einsatz im öffentlichen Raum gewährleisten sollen. Wichtige rechtliche Rahmenbedingungen umfassen:

* **Flugzonen-Beschränkungen**: Autonome Drohnen dürfen nur in genehmigten Lufträumen fliegen. In urbanen Gebieten sind zusätzliche Genehmigungen erforderlich.
* **Datenschutzbestimmungen**: Da Drohnen Umgebungsdaten sammeln und verarbeiten, müssen sie die Datenschutzgesetze, insbesondere die DSGVO, einhalten.
* **Kennzeichnung und Registrierung**: Jede Drohne muss eine eindeutige Identifikationsnummer tragen und bei den zuständigen Behörden registriert sein.
* **Versicherungspflicht**: Der Betreiber ist verpflichtet, eine Haftpflichtversicherung abzuschließen, um Schäden abzudecken, die durch die Drohne verursacht werden könnten.

## Konzept

### Konzept 1

### Konzept 2……

## Herausforderungen der Umsetzung

* Falscher Akku bestellt
  + neuer Akku bestellt
* Electronic Speed Controller (ESC) ist fehlerhaft geliefert worden
  + musste ausgetauscht werden

## Fazit und AusblicK

# Literaturverzeichnis

# Abbildungsverzeichnis

[Abbildung 1: WCAG Level AA Seal 12](#_Toc187688996)

[Abbildung 2: EN 301 549 Seal 13](#_Toc187688997)

# Tabellenverzeichnis

[Tab. 1: Quellentypen 22](#_Toc340231527)

[Tab. 2: Formatvorlagen 23](#_Toc340231528)

# Abkürzungsverzeichnis

| **Abkürzung** | **Bedeutung** |
| --- | --- |
| UX | User Experience, das Erlebnis und die Wahrnehmung eines Produkts. |

# Anhang