

MASTERARBEIT

Augmented Reality: Frameworks und Workflows für Android und iOS

durchgeführt am
Studiengang Informationstechnik & System-Management
an der
Fachhochschule Salzburg GmbH

vorgelegt von

Florian Auer



Studiengangsleiter: FH-Prof. DI Dr. Gerhard Jöchl

Betreuer: FH Ass-Prof. DI (FH) Dr. Thomas Kurz MSc.

Salzburg, September 2017

Eidesstattliche Erklärung

Hiermit versichere ich, Florian Auer, geboren am 21.02.1992, dass die vorliegende Masterarbeit von mir selbstständig verfasst wurde. Zur Erstellung wurden von mir keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Salzburg, am 01.09.2017

Florian Auer

1510581017

Matrikelnummer

Allgemeine Informationen

Vor- und Zuname:	Florian Auer
Institution:	Fachhochschule Salzburg GmbH
Studiengang:	Informationstechnik & System-Management
Titel der Masterarbeit:	Augmented Reality: Frameworks und Workflows für Android und iOS
Schlagwörter:	Augmented Reality, Workflow, iOS, Android, Wikitude, JavaScript, Kudan, Vuforia, Mobile, AR
Betreuer an der FH:	FH Ass-Prof. DI (FH) Dr. Thomas Kurz MSc.

Abstract

The market for Augmented Reality (AR)-Applications is developing fast. In 2016, there was a turnover of about 6.1 billion US-Dollar (USD) for AR and Virtual Reality (VR)[1]. The volume of sales could be at 80 billion USD in 2025[2]. Developers of AR-Solutions are facing problems. There are many scopes, frameworks and platforms. This paper solves these problems for Android and iOS in three steps. At first, there is a comparison between the 10 most relevant frameworks. Secondly, three libraries are evaluated with the aid of a classic AR-Use Case. At last, the paper describes the implementation of a 3D-prototype. The results help to start AR-projects and reduce costs. They contain the key features of the most significant SDKs, the best workflow for writing once, running anywhere and they identify sticking points. The information in this paper can be utilized potentially generalizable and helps to lower the amount of needed resources up to 50 Percent.

Inhaltsverzeichnis

Inhaltsverzeichnis	4
Abkürzungsverzeichnis	5
Abbildungsverzeichnis	6
Tabellenverzeichnis	7
1 Einleitung	8
1.1 Augmented Reality: Begriffsklärung und Abgrenzung	8
1.2 Problemstellung und Motivation	9
1.3 Zieldefinition und Vorgehensweise	11
2 Theoretischer Vergleich relevantester SDKs	13
2.1 Relevanteste SDKs	14
2.1.1 ARLab	14
2.1.2 AR-media	16
2.1.3 CraftAR	18
2.1.4 D'Fusion	20
2.1.5 Holobuilder	21
2.1.6 Kudan AR Engine	22
2.1.7 Vuforia	24
2.1.8 Wikitude	26
2.1.9 Xloudia	28
2.1.10 XZIMG	30
2.2 Anforderungsmatrix	31
2.2.1 Zusammenfassung	33
2.2.2 Gegenüberstellung	34
2.2.3 Evaluierung	36
2.3 Ergebnisse	37
3 Machbarkeitsanalyse ausgewählter SDKs	39
3.1 Anwendungsfall	39
3.2 Kudan AR Engine	40
3.2.1 Installation	40
3.2.2 Implementierung	41

3.2.3	Ergebnisse	42
3.3	Vuforia	43
3.3.1	Installation	43
3.3.2	Implementierung	44
3.3.3	Ergebnisse	45
3.4	Wikitude	46
3.4.1	Installation	46
3.4.2	Implementierung	47
3.4.3	Ergebnisse	48
3.5	Vergleich der SDKs	50
4	Umsetzung eines Prototyps	52
4.1	Anwendungsfall	52
4.2	Vorgehen	54
4.3	Ergebnisse	56
5	Schlussbetrachtung	59
5.1	Ergebnisse dieser Arbeit	59
5.2	Diskussion der Ergebnisse	62
5.3	Zusammenfassung	64
5.4	Ausblick in die Zukunft von Augmented Reality	65
	Glossar	69
	Literatur	70

Abkürzungsverzeichnis

ADE	ARchitect Desktop Environment
API	Application Programming Interface
AR	Augmented Reality
FBX	Filmbox
HUD	Head-Up-Display
IDE	Integrated Development Environment
IMU	Inertial Measurement Unit
IoT	Internet of Things
ms	Millisekunden
REST	Representational State Transfer
SDK	Software Development Kit
SLA	Service Level Agreement
SLAM	Simultaneous Localization and Mapping
USD	US-Dollar
VR	Virtual Reality
W3T	Wikitude 3D Format

Abbildungsverzeichnis

2.1	VuMark	25
3.1	Anwendungsfall Machbarkeitsanalyse	39
4.1	Anwendungsfall Prototyp	52

Tabellenverzeichnis

2.1	Preise ARLab in Euro	15
2.2	Preise Inglobe Technologies in Euro	17
2.3	Anforderungsmatrix	31

1 Einleitung

Dieses Kapitel behandelt einerseits die computergestützte, erweiterte Realität im Allgemeinen. Andererseits legt die Passage Motivation, Ziele und Vorgehen für weitere Abschnitte der wissenschaftlichen Arbeit dar.

1.1 Augmented Reality: Begriffsklärung und Abgrenzung

Im Folgenden wird ein Überblick über AR skizziert, der Begriff erklärt und von ähnlichen Themengebieten abgegrenzt.

Begriffserklärung

Der englische Ausdruck “Augmented Reality“ lässt sich mit „erweiterte Realität“ übersetzen. Dieser Begriff impliziert die computergestützte Anreicherung der Wirklichkeit mit Informationen in Echtzeit. Als Anwendungsbeispiel dient das Einblenden von virtuellen Objekten in die physische Umgebung. Die Erweiterung der Wahrnehmung beschränkt sich größtenteils auf die visuellen Sinneseindrücke[3]. Einige Endgeräte zur Darstellung von Augmented Reality (AR) sind Smartphones, Tablets, Datenbrillen und Fahrzeugdisplays. Dadurch entstehen unterschiedlichste Verwendungsmöglichkeiten der erweiterten Realität. Als Technologie zur Umsetzung von Anwendungsfällen kann in Einzelfällen neben AR auch Virtual Reality (VR) eingesetzt werden.

Abgrenzung

Die erweiterte Wirklichkeit ist zwischen der unverfälschten menschlichen Wahrnehmung und der virtuellen Realität einzuordnen. VR bezeichnet eine komplett künstliche Abbildung einer physischen Umgebung[3].

1.2 Problemstellung und Motivation

Die erweiterte Realität umfasst ein breites Spektrum an Anwendungsgebieten. Dieses Unterkapitel stellt die damit einhergehenden Probleme dar und beschreibt die Relevanz sowie die Motivation für eine Lösung.

Probleme

Es gibt diverse Möglichkeiten der Nutzung von AR. Einige relevante Funktionen sind Marker, Image Recognition, Object Tracking und Face Tracking[4].

Die Implementierung der je nach Anwendungsfall benötigten funktionalen Anforderungen einer AR-Applikation erfordern Know-How unterschiedlichster Kategorien. Zu diesem Wissen zählen unter anderem Computer Vision, Computer Graphics oder Human-Computer Interaction (für weitere Informationen siehe [5]).

Für die Entwicklung von AR-Applikationen ohne Zuhilfenahme bestehender Komponenten entsteht, neben dem Bedarf an Know-How, ein hoher Implementierungsaufwand. Diese Entwicklungszeit beeinflusst direkt die Kosten und den Zeitpunkt der Produkteinführung. Um die Wirtschaftlichkeit von AR-Projekten zu gewährleisten beziehungsweise die Konkurrenzfähigkeit zu erhöhen, kann es von Vorteil sein, auf Softwarekomponenten von Dritten zurück zu greifen.

Ein weiteres Problem ergibt sich durch die Vielzahl von Angeboten. Mehr als 68 Hilfsmittel im Bereich der AR werden im Internet offeriert.

Relevanz

Die hohe Verbreitung und die Zukunftsträchtigkeit von AR-Anwendungen verdeutlichen die Relevanz der Probleme. Auf den diversen Plattformen gibt es unterschiedliche Applikationen, die Augmented Reality bieten. Der Google Play Store listet bei der Eingabe von “Augmented Reality” 250 Applikationen auf[6].

Das Forschungs- und Beratungsunternehmen Gartner sieht für AR, laut dem “Hype Cycle for Emerging Technologies“ von 2016, eine breite Akzeptanz des Marktes in fünf bis zehn Jahren[7].

Motivation

Das Investmentbanking- und Wertpapierhandelsunternehmen Goldman Sachs prognostiziert, dass das Marktvolumen im Jahr 2025 für AR und VR 80 Milliarden Dollar betragen wird (Im Jahr 2016 waren es 6,1 Milliarden Dollar)[1, 2]. Aufgrund der bestehenden und potenziell wachsenden Bedeutung der erweiterten Realität versucht diese Arbeit die aufgeführten Probleme zu lösen. Im Fokus steht die Vereinfachung und Verbesserung der Entwicklung von AR-Applikationen für einzelne Personen und Unternehmen. Auf Details geht das nächste Unterkapitel ein.

1.3 Zieldefinition und Vorgehensweise

Der folgende Abschnitt behandelt die Ziele der weiteren Abhandlung sowie die damit einhergehende Verfahrensweise.

Zieldefinition

Diese Arbeit gibt einen Überblick über die relevantesten Software Development Kits (SDKs) für die erweiterte Realität auf mobilen Plattformen. Das Augenmerk liegt aufgrund der Verbreitung auf den Betriebssystemen Android und iOS. Dadurch wird nicht genauer auf AR-Brillen eingegangen.

Die SDKs müssen verfügbar und sowohl Android als auch iOS unterstützen. Eine Steigerung der Relevanz erreicht das jeweilige Framework durch funktionale wie auch nicht-funktionale Aspekte, welche die Entwicklung von AR-Applikationen maßgeblich unterstützen.

Ein weiteres Ziel ist die Skizzierung einer Übersicht hinsichtlich Funktionalitäten und Besonderheiten der SDKs. Die bedeutendsten Charakteristiken eines Frameworks erfahren eine genauere Vorstellung beziehungsweise Inspizierung.

Um die unterschiedlichen Development-Kits miteinander vergleichen zu können, werden die wesentlichen Anforderungen an diese herausgearbeitet und beschrieben. Eine Gegenüberstellung dient zur Aufführung von Besonderheiten der unterschiedlichen Frameworks.

Die Validierung der Erkenntnisse aus den Vergleichen erfolgt in Form eines Prototyps mit geeignetem Anwendungsfall auf iOS und Android.

Ziel ist zudem, Besonderheiten sowie Schwierigkeiten bei der praktischen Umsetzung aufzuzeigen und zu erörtern. Dabei entsteht ein Workflow zur Applikationsentwicklung auf den Plattformen und dient zur Minimierung des Aufwands beim Einsatz der jeweiligen Frameworks. Der Arbeitsablauf setzt den Fokus auf die unter Kapitel 1.2 erläuterten Punkte.

Vorgehensweise

Zunächst werden alle 68 Frameworks nach den bedeutendsten Kriterien aus socialcompare¹ klassifiziert. Frameworks, welche die obligatorischen Eigenschaften wie zu unterstützende Betriebssysteme (Android und iOS) oder SDK-Verfügbarkeit nicht gewährleisten, erfahren keine Berücksichtigung. Es folgt eine quantitative Auswertung, wobei jedes erfüllte Kriterium eine Bewertungseinheit bedeutet. Die Aufsummierung der Punkte geschieht je Development-Kit.

Die Eigenschaften und Besonderheiten der 10 am besten bewerteten Frameworks werden zunächst beschrieben und daraufhin in einer Anforderungsmatrix dargestellt. Es erfolgt im Zuge der quantitativen Auswertung eine Gewichtung nach Kriterium und eine erneute Aufsummierung der erreichten Punkte nach Framework.

Eine Implementierung von vergleichbaren, einfachen Beispielanwendungen als Machbarkeitsstudie geschieht mit Hilfe der drei bestbewerteten SDKs. In diesem Fall findet eine qualitative Auswertung statt, indem Auffälligkeiten beziehungsweise Besonderheiten verglichen werden. Daraus ergibt sich das Framework für die abschließende Anwendungsfall-Implementierung. Die Realisierung dient zur Nutzung und Validierung des Workflows. Der Arbeitsablauf beinhaltet die Anbindung der selben SDK sowohl an iOS als auch an Android. Des weiteren wird beschrieben, inwieweit betriebssystemunabhängiger Quellcode verwendbar ist und welche Unterschiede sich hierbei ergeben.

¹<http://socialcompare.com/en/comparison/augmented-reality-sdks>

2 Theoretischer Vergleich relevantester SDKs

Dieses Kapitel stellt das Vorgehen dar, wonach eine Vielzahl von SDKs gefiltert werden. Für die in Unterkapitel 1.3 beschriebenen Ziele ergeben sich dadurch die relevantesten Hilfsmittel.

Als Quelle für einen Großteil der verfügbaren SDKs sowie einer ersten Klassifizierung dient die kollaborative Plattform namens socialcompare¹. Der Datensatz wird fortwährend von NutzerInnen gepflegt. Er bietet diverse Kriterien, um die SDKs bewerten zu können.

Die Liste an Frameworks wird mit Hilfe von diversen weiteren Referenzen[4, 5, 8, 9] validiert und auf Vollständigkeit überprüft. Um die übrigen 68 SDKs zu filtern dienen obligatorische Eigenschaften, wie zu unterstützende Betriebssysteme (Android und iOS). Development-Kits, die nicht verfügbar sind, finden keine Beachtung. Die Kriterien Android, iOS, Unity (3D), Marker, NaturalFeature, 3D Object Tracking, GPS, IMU Sensors, Visual Search, FaceTracking und ContentAPI von socialcompare werden verwendet, um die SDKs bewerten zu können. Diese Eigenschaften entsprechen prinzipiell den in Abschnitt 1.3 festgelegten Intentionen. Je Framework aufsummiert werden die Merkmale verwendet, um eine Einstufung vorzunehmen und somit eine Relevanz für diese Arbeit auszudrücken.

Im Folgenden finden die 10 SDKs mit den meisten Punkten Anwendung.

¹<http://socialcompare.com/en/comparison/augmented-reality-sdks>

2.1 Relevanteste SDKs

Diese Sektion beinhaltet diverse Informationen zu den 10 bedeutendsten Frameworks. Dabei wird auf ausgewählte Funktionalitäten, Kompatibilität, preisliche Aspekte sowie Besonderheiten eingegangen. Die aufgeführten Eigenschaften stellen eine Ergänzung zu Tabelle 2.3 dar. Für die Klärung offener Fragen bezüglich der zur Verfügung gestellten Software ist zu jedem Anbieter Kontakt aufzunehmen. Als Abkürzung für die gesamte Bezeichnung eines Development-Kits wird gegebenenfalls nur der meist verwendete Name angegeben.

2.1.1 ARLab

Die Firma Augmented Reality Lab S.L. mit Niederlassungen in Spanien und Israel ist Herausgeber von ARLab[10]. Diese Augmented Reality-Lösung setzt sich aus mehreren, separat erhältlichen SDKs zusammen. Auf der Homepage führt das Unternehmen Image Matching, Image Tracking, Virtual Buttons sowie “AR Browser” und “3D Engine Advance” auf[11]. Als Unterstützung von ARLab wurde eine Community-Plattform namens LinkAR ins Leben gerufen[12]. Die Besonderheiten dieser Plattform sind unter der Sektion Support dieses Unterkapitels gelistet.

Ausgewählte Funktionalitäten

ARLab bietet eine überschaubare Anzahl an AR-Funktionen. Neben den bereits verfügbaren Möglichkeiten sind sowohl Object Tracking als auch Virtual Buttons angekündigt[11, 13]. Dieser Anbieter von AR-Frameworks unterstützt zwar kein Unity, jedoch bietet er eine eigene 3D Engine an[13].

Kompatibilität

Die von ARLab angebotenen SDKs setzen mindestens Android 2.3.3 beziehungsweise iOS 5.0 voraus. Weder über die Website noch durch eine Support-Anfrage gibt der Anbieter Auskunft darüber, ab welchen Versionsnummern voller Funktionsumfang besteht[14, 15, 16, 17]. Die Frameworks Image Matching und Image Tracking sind offline verwendbar, zu den anderen Komponenten werden keine Angaben gemacht[15, 17, 18].

Support

Tutorien, Beispielapplikationen, erweiterte Dokumentation etc. müssen mit einer eigenen Währung gekauft werden. Die Akquirierung der Devisen ist als Startkapitel nach der Anmeldung sowie bei diversen Aktivitäten möglich. Einige dieser Tätigkeiten sind das Erstellen einer Applikation, das kommentieren von Apps oder Profilen anderer NutzerInnen sowie bei der Partizipation im Forum[19]. Auf eine E-Mail mit offenen Fragen an das Unternehmen wurde innerhalb von fünf Werktagen reagiert. Dabei konnte eine von sieben Fragen beantwortet werden[17].

Preise

Jegliche Käufe von ARLab-Software erfordern eine kostenlose Anmeldung[19]. Die Preise für die jeweiligen SDKs sind in Tabelle 2.1 aufgeführt. Diese gelten für den einmaligen Kauf. Monatliche Kosten fallen nicht an[11]. Es ist möglich, kostenlose Testversionen auf Anfrage zu erhalten[17].

Software	Preis
AR Browser	199
Image Matching	299
Image Tracking	399
Virtual Buttons	599
3D Engine Advance	99

Tabelle 2.1: Preise ARLab in Euro
[11]

2.1.2 AR-media

Das Framework zur Entwicklung von AR-Applikationen der italienischen Firma Inglobe Technologies Srl nennt sich AR-media SDK. AR-media bietet neben den verbreitetsten AR-Funktionen einige zusätzliche Features. Auf diese wird im Folgenden eingegangen.

Ausgewählte Funktionalitäten

Neben den Funktionalitäten, worauf in dieser Arbeit alle 10 SDKs untersucht werden, bietet dieses Framework einige Besonderheiten. Inglobe Technologies stattet sein Augmented Reality Development-Kit zusätzlich mit VR - Komponenten aus. Diese sind "VR Motion" und "VR Panoramas"[20]. Zudem werden diverse Hilfsapplikationen angeboten. Die Smartphone- und Tablet-Applikation 3D Targets Mapper dient zur Erstellung von 3D Objekten reeller Gegenstände. Für die 3D-Tools Autodesk 3ds Max®, Trimble SketchUp™, Autodesk Maya, MAXON Cinema 4D, Vectorworks® oder Nemetschek Scia Engineer existiert jeweils ein Plugins namens ARPlugin. Diese ermöglichen die Konvertierung der 3D Modelle in für die AR-media SDK verwendbare Datentypen[20].

Inglobe Technologies hat ein vielseitiges Content-API im Angebot. Das Webinterface, genannt "hyperspaces" arbeitet jedoch derzeit nicht mit der Ar-media SDK zusammen. Zur vielseitigen Verwendung der bereitgestellten Inhalte muss die zugehörige Applikation als Client für iOS verwendet werden. Eine Android-Variante befindet sich noch in der Entwicklung. Hyperspaces ist getrennt von der Ar-media SDK zu erwerben[21]. Details dazu sind neben der im Folgenden beschriebenen Kompatibilität unter Sektion Preise dieses Unterkapitels zu finden.

Kompatibilität

Die AR-media SDKs sowie diverse derzeit verfügbare mobile Applikationen von Inglobe Technologies benötigen als niedrigste Version iOS 7.0 und Android 4.1. Die Ausführungen unterstützen laut dem Anbieter zudem bereits den vollen Funktionsumfang des Frameworks[22, 23].

Support

Neben dem bei den meisten Anbietern gängigen Hilfeleistungen durch ein entsprechendes Forum bietet Inglobe Technologies S.r.l. zudem E-Mail-Support an. Die technische Unterstützung ist bei den Abonnements ab 6 Monaten inkludiert[24]. Das Unternehmen reagierte auf eine E-Mail mit Fragen am nächsten Arbeitstag und lieferte in allen Punkten ausreichend Informationen[25].

Preise

Die Preise für hyperspaces und AR-media SDK beziehen sich auf Abonnements. Sie sind in Laufzeiten von 1, 6 und 12 Monate unterteilt. In Tabelle 2.2 werden die monetären Kosten dargelegt. Aufgrund diverser Ausführungen von hyperspaces wird hier nur eine Preisspanne angegeben (für Details siehe [21]). Die Tabelle beinhaltet zwar keine kostenlose Ausführung der AR-media SDK, jedoch ist es möglich, eine Testversion des Frameworks auf Anfrage zu erhalten[25].

Software	1 Monat	6 Monate	12 Monate
AR-media SDK	1000	2500	3500
hyperspaces	0 - 250	0 - 1375	0 - 2500

Tabelle 2.2: Preise Inglobe Technologies in Euro
[21, 24]

2.1.3 CraftAR

Die Firma Catchoom mit Standorten in den USA, Spanien und Australien ist der Herausgeber von CraftAR-SDKs[26]. Die Funktionalitäten der Development-Kits sind einerseits gebündelt in der Pro-SDK erhältlich. Andererseits ist es möglich, Frameworks jeweils für AR-Anwendungen und Image Recognition zu beziehen[27]. Im Webauftritt² hebt Catchoom Image Recognition stark hervor. Der Anbieter gibt an, bei dieser Funktionalität Marktführer zu sein[28]. Zur Bilderkennung existieren mehrere Implementierungsarten, worauf im nächsten Unterkapitel eingegangen wird.

Ausgewählte Funktionalitäten

Zwei verschiedene Frameworks von CraftAR sind für Image Recognition-Anwendungen verfügbar. “Image Recognition Cloud” ist ein leichtgewichtiges SDK, welches die Bilderkennung in der Cloud durchführen lässt. Als Pendant dazu ist eine auf die Rechenleistung des lokalen Geräts beschränkte Bilderkennung möglich[29]. Beide Arten der Image Recognition können mit Hilfe der Pro-SDK im laufenden Betrieb eingesetzt werden. Eine Besonderheit von CraftAR ist die Möglichkeit der Object Recognition. Jedoch wird beim Tracking der Objekte nur von zwei Dimensionen ausgegangen. Dies hat zur Folge, dass komplexere dreidimensionale Objekte schlecht bis gar nicht verfolgt werden[30]. Das “WYSIWYG content creation tool” erleichtert das Erstellen von AR-Inhalten. Die Bedienung dieses Hilfsprogramms ist ohne Entwicklerfähigkeiten möglich[31].

Kompatibilität

Die Frameworks unterstützen laut dem Catchoom-Team als niedrigste Version iOS 8 und Android 3.0. Ab diesen Betriebssystemausführungen ist zudem der vollständige Funktionsumfang verfügbar. Da Android 3.0 nur von Tablets unterstützt wird und Android 4.0.3 die nächste stabile Ausführung des Betriebssystems für Smartphones ist, dient 4.0.3 als niedrigst mögliche Version für Android-Smartphones[23]. Die ContentAPI kann mit Hilfe einer RESTful-Schnittstelle verwendet werden und ermöglicht somit eine hohe Kompatibilität mit bestehenden Systemen. Zudem bietet das Interface diverse Möglichkeiten zum Controlling der abgerufenen Inhalte[31].

²<https://catchoom.com/>

Support

Je nach Abonnement bietet Catchoom Forum-, Email-, Chat- und Telefon-Support. Eine E-Mail mit offenen Fragen zur SDK blieb bis zum Abschluss dieser Arbeit unbeantwortet. Auf die Kosten der unterschiedlichen Pläne wird im nächsten Kapitel eingegangen.

Preise

Catchoom bietet die Verwendung von Anwendungen mit CraftAR-SDKs im zeitlich begrenzten Abonnement an. Beim Veröffentlichen der Applikation können Zusatzkosten anfallen.

Zeitliche Verträge belaufen sich auf 99 bis 249 Euro pro Monat bei jährlicher Zahlung. Neben kostenlosen Test-Versionen werden auch Premium-Angebote ohne Limitierungen auf Anfrage angeboten.

Um eine Applikation veröffentlichen zu können, muss je SDK eine gegebenenfalls einmalige Pauschale entrichtet werden. Cloud Image Recognition ist kostenlos. On-Device Image Recognition und Augmented Reality sowie das All-In-One Framework werden mit 2490 Euro, 1990 Euro und 3890 Euro angeboten[29].

2.1.4 D’Fusion

Es steht kein Framework auf der Website von T-Immersion³ zur Verfügung. Darüber hinaus blieb eine Anfrage an den Support unbeantwortet. Aus diesen Gründen entspricht Total Immersion D’Fusion nicht der Zielsetzung dieser Arbeit.

³<http://www.t-immersion.com/products/dfusion-suite/dfusion-studio>

2.1.5 Holobuilder

Die Software ermöglicht es Bau- und Immobilien-ExpertInnen, 360 Grad Ansichten von Gebäuden zu erstellen. Es sind keine Programmierkenntnisse nötig. Für die Realisierung von anderen Anwendungsfällen existiert kein Framework. Aufgrund der erheblichen Einschränkungen ist Holobuilder für diese Arbeit nicht relevant[32].

2.1.6 Kudan AR Engine

Die Firma Kudan Limited unterhält Niederlassungen in Großbritannien und Japan. Sie bietet ihr AR-Framework, die Kudan AR Engine, einerseits als All-In-One SDK an. Andererseits ist es möglich, nur einen Teil des gesamten Development-Kits einzubinden. Im Folgenden wird ein Überblick über die Eigenschaften des Frameworks skizziert[33].

Ausgewählte Funktionalitäten

Funktionalitäten für Image Tracking beziehungsweise Object Tracking ohne Rendering sind als sogenannte “Computer Vision (CV) - SDK” erhältlich. Dieses im Umfang reduzierte Framework befindet sich jedoch noch in der Beta-Phase[33, 34]. Beide SDKs bieten Object Tracking und Lokalisierung, sowohl mit Marker als auch Markerlos[34]. Zudem unterstützen die Frameworks Stereokameras und diverse andere Messwertgeber, wie beispielsweise Raumentiefensensoren[34].

Kudan wird unter anderem aufgrund der robusten SLAM-Implementierung von Kunden ausgewählt[9].

Kompatibilität

Beide Kudan-Frameworks sind nicht auf eine Internetverbindung angewiesen. Die SDKs unterstützen iOS ab Version 8 und Android ab Version 4.0.2 (API Level 15). Seit diesen Ausführungen ist außerdem der volle Funktionsumfang verfügbar. Das Unity-Plugin von Kudan kann ab Unity-Version 5.2 angebunden werden[35].

Support

Kudan Limited bietet Unterstützung für Kunden in Form eines Forums und ein Support-Portal. Der Zugang für zahlende Kunden dient als Schnittstelle zu einem Ticketing System[36, 37]. Nachdem Fragen bezüglich des Frameworks auftraten, wurde eine Anfrage über das Forum getätigt. Die Antwort erfolgte noch am selben Tag. Dabei konnten genügend Informationen zu allen Punkten geliefert werden[35].

Preise

Die SDKs sind kostenlos nutzbar. Wenn jedoch eine Applikation veröffentlicht werden soll, müssen 1000 britische Pfund pro Jahr und Applikation gezahlt werden. Es ist möglich, eine Volumenlizenz zu beantragen[38].

2.1.7 Vuforia

Der Vertrieb des AR-Frameworks Vuforia erfolgt durch PTC Inc.. Der Anbieter ist ein globaler IT-Dienstleister mit den Schwerpunkten Internet of Things (IoT) und AR. Für das im Jahr 1985 gegründete Unternehmen arbeiten circa 6000 Angestellte in 30 Ländern[39].

Ausgewählte Funktionalitäten

Vuforia unterstützt zusätzlich zu den in Unterkapitel 2.2 aufgeführten Technologien diverse weitere Funktionalitäten.

Mit dem sogenannten “Extended Tracking” ermöglicht das SDK, AR-Inhalte anzuzeigen, obwohl der zugehörige Marker nicht im Kamerafokus des Endgerätes ist. Als Beispiel dient ein Wolkenkratzer, der als 3D-Objekt aus einem 2D-Bild ragt, welches als Marker verwendet wird. Wenn ein Smartphone sich dem Wolkenkratzer nähert, verschwindet das 2D-Bild aus dem Sichtfeld der Smartphone-Kamera. In diesem Fall bleibt das hohe Gebäude jedoch im Raum bestehen und kann mit Hilfe des Smartphones detailliert inspiziert werden[40].

Neben dem Extended Tracking ist es außerdem möglich, zwei Objekte gleichzeitig zu erkennen und zu verfolgen[41].

Als Marker für jegliches Auslösen von Ereignissen können neben gewöhnlichen 2D-Objekten obendrein sogenannte “VuMarks” verwendet werden. VuMarks stellen, analog zu QR- und Bar-Codes, bei für den Menschen nahezu gleichem Aussehen unterschiedliche Informationen bereit. Ein Beispiel hierfür ist Abbildung 2.1[42].

Vuforia bietet außerdem Texterkennung und eine Android-Applikation zum Scannen von 3D-Objekten, die daraufhin in eigenen Programmen verwendet werden können[40, 41].

Kompatibilität

Das Framework unterstützt als minimale Versionen iOS 7 und Android 4.1 (Details siehe [43, 44]).



Abbildung 2.1: VuMark

[42]

Aufgrund fehlender Angaben des Webauftritts und einer unbeantworteten Anfrage an den Support, können keine Angaben zu den unterstützten Ausgaben von Unity gemacht werden.

Support

Zum einen bietet PTC ein Forum für die Community an, das von Angestellten beaufsichtigt wird. Zum anderen ist Email-Support im Pro-Abonnement inkludiert[45]. Weitere Informationen zum Pro-Abonnement sind unter Sektion Preise dieses Unterkapitels zu finden. Für offene Fragen zur SDK, welche über das Kontaktformular von PTC abgesendet wurden, steht eine Antwort aus.

Preise

Das Vuforia-Framework ist zur Entwicklung frei verfügbar. Kosten entstehen erst, wenn eine Applikation veröffentlicht werden soll. In der Standard-Ausgabe kostet die SDK einmalig 499 USD. Bei Verwendung der Cloudangebote von Vuforia werden 99 USD pro Monat veranschlagt. Zudem ist eine Pro-Variante auf Anfrage erhältlich. Diese beinhaltet zusätzlich unter anderem das “VuMark Generation Application Programming Interface (API)” und das “Advanced Camera API”[46].

2.1.8 Wikitude

Die Wikitude GmbH mit Hauptsitz in Salzburg ist Herausgeber der Wikitude SDK[47]. Das All-In-One-AR Framework bietet eine Fülle an Funktionalitäten, wobei im Folgenden neben den Aufgeführten in Abschnitt 2.2 nur auf die für diese Arbeit Relevantesten eingegangen wird[48, 49].

Ausgewählte Funktionalitäten

Dieses Framework sowie die bereit gestellten Tools bieten diverse zusätzliche Möglichkeiten zu den gängigen AR-Funktionen. Mit Wikitude Studio können Daten zur Laufzeit in die frei verfügbare Wikitude-Applikation sowie in eigene Software integriert werden. Durch die Verwendung des Webbrowsers sind ferner keine Programmierkenntnisse nötig[48].

Die unter AR-Frameworks verbreitete Möglichkeit der Bilderkennung am Gerät unterstützt Wikitude zudem in der Cloud. Die Besonderheit hierbei ist jedoch, dass die sogenannte “Cloud Recognition” darüber hinaus auf Servern möglich ist, die NutzerInnen jener SDK zur Verfügung stellen[50]. Als zusätzliche Erweiterung der Objekterkennung bietet Wikitude neben SLAM ebenfalls Extended Tracking [51].

Der Funktionsbereich namens “Sensor-based AR (Geo-AR)” ist darüber hinaus mit Geo Fencing ausgestattet[50].

Die Erweiterbarkeit des Frameworks wird durch eine API bewerkstelligt. Diese sogenannte “Plugins-API” ermöglicht es EntwicklerInnen, unter anderem auf Kamerabilder als auch Zusatzinformationen zuzugreifen und parallel zu verarbeiten[52].

Kompatibilität

Neben dem Angebots des nativen Application Programming Interface, Unity, Xamarin, Cordova und Titanium gibt es ein Javascript API. Dieses Framework ermöglicht zusätzlich zu den durch die nativen Implementierungen unterstützten Funktionalitäten den Zugriff auf ortsbasierte Services[49].

Die Javascript API wird ab Android 4.0.3 und iOS 8.3 unterstützt, die native Ausführung ab Android 4.0.3 und iOS 8.0[49].

Die Wikitude SDK unterstützt Unity ab Unity-Version 5.0[53].

Support

Die verbreiteten Support-arten über Forum und Email werden von Wikitude unterstützt. Außerdem sind ein Ticketing-System, Telefonsupport, benutzerdefinierte Service Level Agreements (SLAs) und Entwicklertraining verfügbar. Die im Preis inbegriffenen Leistungen variieren je nach Art des Abonnements, worauf im nächsten Unterkapitel näher eingegangen wird[54]. Auf E-Mails zu offenen Fragen zur SDK antwortete der Hersteller am gleichen Tag, bei Bedarf zudem auf Deutsch.

Preise

Neben der kostenlosen Testversion stellt die Wikitude GmbH vier verschiedene Abonnements bereit. SDK Pro, SDK Pro 3D, Cloud und Enterprise. Die Kosten belaufen sich auf 2490 Euro bis 4490 Euro pro Jahr. Die Höhe der Enterprise-Variante wird nur auf Anfrage bekannt gegeben[54].

2.1.9 Xloudia

Das Framework namens Xloudia vertreibt LM3LABS, Inc.. Das Unternehmen ist in Japan, Frankreich und den USA ansässig. LM3LABS hat sich mit den fokussierten Technologien Computer Vision, Deep Learning, AR, VR und Neuronal Networks auf Dienstleistungen im Zusammenhang mit Interaktivität spezialisiert[55, 56, 57, 58]. Bei Xloudia liegt das Hauptaugenmerk weniger auf die Abdeckung der AR-Funktionalitäten. Das Framework wurde für die Erkennung vieler Bilder und die Unterstützung des elektronischen Handels unter Verwendung mobiler Geräte optimiert[59]. Auf Details geht das folgende Unterkapitel ein.

Ausgewählte Funktionalitäten

Die Bilderkennung soll Übereinstimmungen in circa 200 Millisekunden (ms) bei ausreichend schneller Internetverbindung liefern. Die Identifizierung kann aus einer Vielzahl von Fotos (Größenordnung eine Million) erfolgen. Dazu wird bei Video- sowie Kamera-Streams jedes dritte Bild in die Xloudia Cloud geladen[59].

Zur stetigen Verbesserung von Genauigkeit und Geschwindigkeit der Identifizierung basiert die “Xloudia Cloud” auf Machine Learning. Dadurch werden überdies Angaben von der Nähe eines Bildes zu anderen Bildern sowie eine Klassifizierung möglich. Diese Einteilung bezeichnet LM3LABS als “Xloudia Labelling”[59, 60, 61].

Xloudia stellt neben diesen Funktionalitäten außerdem einige weniger verbreitete AR-Features bereit. “Xloudia Color” ermöglicht es, Farben von Objekten zu erkennen. “Xloudia TV” kann einen zuvor indexierten Video-Stream identifizieren. “Xloudia Beacon” integriert iBeacons in die Applikation[59, 62].

Die Verwaltung dieser Funktionalitäten und die Auswertungen in Echtzeit können über das “Unified Dashboard” vorgenommen werden. Diese Content-API bietet diverse Möglichkeiten, wie beispielsweise die Änderung von bestimmten Obergrenzen, die Anzeige von Nutzungsvorhersagen und der Visualisierung von Daten. Zur Änderung von Bildern und zum Download von Daten kann eine Representational State Transfer (REST) API angebunden werden[61].

Kompatibilität

Xloundia setzt aus Sicherheitsgründen stets die aktuellste Version des Betriebssystems voraus. Aus diesem Grund sind keine Informationen über die unterstützten Ausführungen der Plattformen, weder von iOS noch Android, vorhanden. Dies gilt zudem für Unity[63].

Support

Neben dem Support über ein Forum, ermöglicht das Unternehmen die Unterstützung durch ein Ticketing System. Ab einem bezahlten Abonnement der Software ist der sogenannte “Personalised Support” inkludiert. Darunter versteht LM3LABS die Zuordnung von TechnikerInnen zu jeder Klientin und jedem Klienten. Je nach Höhe des monatlichen Abonnements steht eine Mitarbeiterin oder ein Mitarbeiter zwischen zwei und acht Stunden im Monat zur Verfügung[61, 64]. Eine E-Mail mit offenen Fragen wurde an den sogenannten “Customer Service”[64] gesendet, wodurch innerhalb von drei Werktagen alle Punkte beantwortet werden konnten. Auf Informationen zu den Preisen wird im Folgenden eingegangen.

Preise

Wie bereits unter Sektion Support dieses Unterkapitels erwähnt, wird bei Xloundia das Abonnement monatlich abgerechnet und beträgt je nach Umfang zwischen 250 und 349 USD. Der kostenlose Test des Frameworks vor dem Kauf ist möglich, jedoch ist dieser auf ein Monat beschränkt. Im Gegensatz zu diversen anderen SDKs ist es während der Testphase möglich, das Ticketing System zu verwenden[64, 65].

2.1.10 XZIMG

Die XZIMG Limited mit Sitz in Hong Kong ist der Herausgeber von mehreren AR-Frameworks, XZIMG Augmented Face, XZIMG Augmented Vision und XZIMG Magic Face[66, 67].

Ausgewählte Funktionalitäten

Die SDKs unterstützen vergleichsweise wenig Funktionalitäten, da sich die Firma XZIMG auf Face Recognition und Face Tracking spezialisiert hat[66]. Neben Recognition und Tracking bietet XZIMG mit dem Framework “Magic Face” obendrein das Ersetzen des Gesichts sowie die Einblendung von beliebigen Objekten oder Make-up[68]. Ergänzend wird ein Framework zur Bilderkennung angeboten[69].

Kompatibilität

Das Framework setzt Unity 5, Android 4.1 oder mobile Apple-Geräte ab dem iPhone 3GS voraus[67, 68, 69]. Auf eine Anfrage zu minimalen Versionen mit vollem Funktionsumfang und der niedrigsten Ausführung von iOS gab XZIMG keine Antwort.

Support

Es existiert weder ein Forum noch Unterstützung für die Testversion der SDKs. Bei jedem gekauften Framework sind ein Jahr Support sowie Updates mit einbegriffen[67, 68, 69]. Auf eine Anfrage über das Kontaktformular mit offenen Fragen zum Framework ist bis zur Fertigstellung dieser Arbeit keine Antwort eingegangen.

Preise

Wie bereits unter Sektion Support dieses Abschnitts aufgezeigt sind Testversionen der SDKs verfügbar. Die Vollversionen sind einmalig für circa 1600 bis 11500 Euro zu erwerben[67, 68, 69].

2.2 Anforderungsmatrix

Im Folgenden wird eine Übersicht der unter Kapitel 2.1 aufgeführten Frameworks dargestellt. Der Überblick beinhaltet die bedeutendsten Eigenschaften, Support und Plattformen der jeweiligen SDKs im Sinne der Ziele dieser Arbeit (siehe Unterkapitel 1.3). Nicht als maßgeblich eingestufte Development-Kits sind nicht aufgeführt.

		ARLab	AR-Media	CraftAR	Kudan	Vuforia	Wikitude	Xloundia	XZIMG
Eigenschaften	Content-API		O	✓		✓	✓	✓	
	Image Matching	✓	✓	✓	✓	✓	✓	✓	
	Object Tracking		✓	O	✓	✓	✓		✓
	SLAM				✓		✓		
	Indoor Navigation		O						
	Visual Search			✓	✓			✓	
	IMU Sensors		✓		✓	✓	✓		
	Face Tracking								✓
	Virtual Buttons					✓			
	Offline			✓	✓		✓	O	✓
Support	Forum	✓	✓	✓	✓	✓	✓	✓	
	Email		✓	✓		✓	✓	✓	✓
	Ticketing System				✓		✓	✓	
	Telefon			✓			✓		
	Entwickler Training						✓		
Plattformen	iOS	5.0	7.0	7.1	8.0	✓	8.0	✓	iPhone 3GS
	Android	2.3.3	4.1	4.2	4.0.2	✓	4.0.3	✓	4.1
	Unity		✓	✓	✓	✓	✓	✓	✓
	Xamarin						✓		
	Titanium						✓		
	Cordova			✓			✓		

Tabelle 2.3: Anforderungsmatrix

Die Tabelle 2.3 stellt ausgewählte Charakteristiken der jeweiligen Frameworks komprimiert dar. Die Informationen stammen vom Webaufttritt bzw. von E-Mails des jeweiligen Herausgebers. Für Details sind die Quellen im Literaturverzeichnis aufgeführt. Die SDKs werden meist stetig weiterentwickelt, wodurch diese Übersicht eine Momentaufnahme ist.

Neben den Zahlen zur Angabe der Versionsnummer bedeutet das Symbol “✓”, dass das Framework eine gewisse Eigenschaft unterstützt. Bei eingeschränkter Funktionalität wird das Symbol “O” verwendet. Die Zahlen, die sich auf iOS und Android beziehen, sagen aus, dass das jeweilige SDK ab dieser Versionsnummer des Betriebssystems

unterstützt wird. Falls keine Nummer angegeben wird, existieren keine genaueren Informationen des Herausgebers. In diesem Fall ist davon auszugehen, dass die Software mindestens auf der aktuellsten Version des Betriebssystems lauffähig ist. Die Fachbegriffe der Tabelle werden im Glossar erläutert.

2.2.1 Zusammenfassung

Dieses Unterkapitel fasst die unterschiedlichen Merkmale der in Tabelle 2.3 dargestellten Frameworks zusammen.

Die Features von acht ausgewählten SDKs wurden dargestellt. Bei einem Blick auf die Tabelle kann zunächst festgestellt werden, dass keines dieser Hilfsmittel alle aufgeführten Eigenschaften besitzt. Nur eines unterstützt alle Entwicklungsframeworks sowie Support-Arten.

Eigenschaften

Jedes der Development-Kits bietet unterschiedliche Features, wobei manche Frameworks eine deutlich geringere Anzahl als Andere vorsehen. Die Funktionalität Image Matching findet sich jedoch in fast jedem Portfolio der aufgeführten SDKs wieder. Im Gegensatz dazu wird Indoor Navigation von nahezu keinem Hersteller angeboten (nur AR-Media eingeschränkt). Welches Framework mit welchen weiteren Charakteristiken aufwartet, variiert stark. Zudem sind SDKs aufgeführt, die auf wenige Features beschränkt sind.

Support

Nahezu jeder Hersteller offeriert Support via E-Mail und Forum. Unterstützung in telefonischer Form oder als Entwickler Training wird selten angeboten. Hier sind CraftAR und Wikitude zu erwähnen, wobei letzteres alle aufgeführten Kanäle anbietet.

Plattform

Wie in Unterkapitel 1.3 definiert, werden iOS und Android von jedem Framework nativ unterstützt. Jedoch sind diese SDKs nur ab bestimmten Betriebssystemversionen anwendbar. Jene Frameworks, zu denen Angaben bestehen, setzen bis auf eine Ausnahme Android 4.x sowie iOS zwischen 7 und 8 voraus.

Zusätzlich zu den nativ unterstützten Betriebssystemen kommen meist Entwicklungsframeworks hinzu. Ein Plugin für Unity wird von den meisten Herstellern zur Verfügung gestellt. Weitere SDK-Ausführungen für andere Entwicklungsplattformen werden im Gegensatz dazu nur selten angeboten.

2.2.2 Gegenüberstellung

Im Folgenden wird auf Besonderheiten ausgewählter SDKs der Tabelle 2.3 eingegangen und diese anhand der Anforderungsmatrix miteinander verglichen.

ARLab

Das Framework, das die niedrigsten Betriebssystemversionen unterstützt und somit in der Theorie auf den meisten Geräten funktioniert, ist ARLab. Die Auslieferung dessen findet mit den wenigsten Features statt. Im Detail bedeutet das, dass ARLab von den ausgewählten Eigenschaften allein Image Matching beinhaltet. Zudem werden im Vergleich zu den meisten anderen Herstellern weniger Support-Optionen angeboten. Das SDK stellt als einziger kein Unity-Plugin bereit.

AR-Media

AR-Media kann Applikationen mit einer durchschnittlichen Anzahl an zusätzlichen Features bereichern. Dies gilt zudem beim Support und den Plattformen. Als Alleinstellungsmerkmal ist Indoor Navigation zu erwähnen, die zumindest eingeschränkt möglich ist und von der Konkurrenz nicht angeboten wird.

CraftAR

Bei nahezu allen aufgeführten Kriterien befindet sich das offline verwendbare Development-Kit im Mittelfeld. Hervorzuheben ist zum Einen die telefonische Erreichbarkeit von ServicemitarbeiterInnen. Zum Anderen kann zur Entwicklung von Software das Framework Cordova eingesetzt werden. Die Charakteristiken bietet jeweils nur ein Konkurrent - in beiden Fällen Wikitude.

Kudan

Kudan ermöglicht es EntwicklerInnen, verhältnismäßig viele Arten der erweiterten Realität zu implementieren. Dabei gilt zu beachten, dass iOS erst ab der Betriebssystemversion 8.0 verwendet werden kann. Der Hersteller des SDKs leistet keine Hilfe über

den Kommunikationskanal E-Mail. Jedoch können Probleme durch ein Ticketing System dargelegt werden. Neben Wikitude bietet Kudan die einzige Implementierung von SLAM.

Vuforia

Das Framework ist der einzige Anbieter von Virtual Buttons. Eine Einschränkung ist, dass der Hersteller keine Informationen bezüglich unterstützten Android- und iOS-Versionen bereit stellt. Dadurch kann ohne ausgiebige Tests nur die neueste Betriebssystemausführung als lauffähig deklariert werden.

Wikitude

Im Gegensatz zu ARLab mit der deutlichen Beschränkung auf wenig Merkmale, steht Wikitude. Dieses Hilfsmittel bietet neben Kudan die meisten funktionalen und nicht-funktionalen Eigenschaften. Zudem werden alle aufgeführten Support-Arten und Plattformen bereit gestellt. Für Xamarin und Titanium sind keine der inspizierten Frameworks außer Wikitude verfügbar. Jedoch setzt Wikitude nahezu die höchsten Betriebssystemversionen voraus.

Xloundia

Xloundia ermöglicht im Bezug zu den konkurrierenden Frameworks eine geringere Abdeckung der Eigenschaften. Zudem sind keine Features implementiert, die andere Development-Kits nicht bieten würden. Analog zu Vuforia fehlen Informationen des Herstellers zu unterstützten Android- und iOS-Versionen.

XZIMG

Das Hilfsmittel ermöglicht vergleichsweise wenig AR-Funktionalitäten. Image Matching wird im Gegensatz zu allen anderen untersuchten SDK nicht unterstützt. Die Information zur niedrigsten iOS-Version ist zudem ein einschränkender Faktor. Der Hersteller gibt nur "iPhone 3GS" an. Das Smartphone wurde mit der Betriebssystemversion 3.0 ausgeliefert, erhielt jedoch Updates bis mindestens 6.1.3 [70, 71]. Als Alleinstellungsmerkmal dient hingegen Face Tracking.

2.2.3 Evaluierung

Dieses Unterkapitel stellt das Auswahlverfahren für die drei Frameworks dar, die in Kapitel 3 verwendet werden.

Jedes der in Tabelle 2.3 aufgeführten SDKs bietet Vor- und Nachteile. Viele Features können zu erhöhter Komplexität der Anbindung des Development-Kits führen. Ein weiterer Nachteil ist die mögliche Inanspruchnahme von zusätzlichem Speicher auf dem mobilen Gerät. Neben Kosten, Anbindungsaufwand und Support-AnsprechpartnerInnen für jedes einzelne Framework überwiegt jedoch die Intention dieser Arbeit. Das Ziel ist, einen Workflow zu erarbeiten, der so allgemein gültig wie möglich ist. Dadurch werden Hilfsmittel mit den meisten Features präferiert.

Um drei SDKs im folgenden Kapitel analysieren zu können, kommt ein quantitatives Vorgehen zum Einsatz. Je Development-Kit werden alle uneingeschränkten Features in der Tabelle mit gleicher Gewichtung addiert. Um jedoch die Vergleichbarkeit der Frameworks gewährleisten zu können, gilt eine Implementierung von definierten, AR-typischen Eigenschaften als Voraussetzung. Diese sind Image Matching und Object Tracking.

Aufgrund dieses Vorgehens werden Kudan, Vuforia und Wikitude ausgewählt.

2.3 Ergebnisse

Im Folgenden werden die auffälligsten Unterschiede und Erfahrungen aus dem Kapitel 2 dargestellt.

Der Fokus ist als auffälligster Gegensatz zu nennen. Die Frameworks haben zum Teil unterschiedliche Schwerpunkte, wobei beispielsweise ARLab und XZIMG zu erwähnen sind. Diese wurden hauptsächlich für die Anwendungsgebiete Image Matching und Face Tracking konzipiert[13, 66]. Neben diesen eng abgegrenzten Funktionsumfängen, sind auch SDKs für ein weites Einsatzspektrum verfügbar. An dieser Stelle ist Wikitude anzuführen[48].

Jedoch kann von einem einzigen Development-Kit nicht jedes Feature abgedeckt werden. Deshalb ist es für speziellere Funktionalitäten oftmals nötig, auf andere Hilfsmittel zurück zu greifen. Beispiele hierfür sind unter anderem “Xludia Color”, womit Farben von Gegenständen bestimmt werden können, und “Wikitude Plugins API”[52, 59]. Bestimmte AR-Anwendungsfälle wie Virtual Buttons oder Face Tracking unterstützt nur Vuforia beziehungsweise XZIMG.

Je Hersteller werden diese gebündelten Charakteristiken unterschiedlich unterteilt. Mancher gliedert die Features nach den nicht-funktionalen Eigenschaften cloud und offline (z.B. CraftAR)[27]. Andere setzen auf die Differenzierung je Anwendungsgebiet (z.B. Kudan, Xludia oder XZIMG)[33, 64, 72]. Die Mehrzahl der untersuchten Hersteller offerieren ihre Software auch als Gesamtpaket[27, 51].

Bei der Auswahl eines Development-Kits muss die Betriebssystemversion der Zielgeräte erörtert werden. Die niedrigsten Plattformauführungen variieren je Hersteller. Zudem geben Vuforia und Xludia keine Informationen dazu preis, wodurch nur von einer Unterstützung der aktuellsten Betriebssystemversion ausgegangen werden kann.

Bis auf ARLab ist von jedem inspiziertem Framework ein Plugin für Unity erhältlich. Als einziges aufgeführtes Hilfsmittel ermöglicht Wikitude die Entwicklung unter Xamarin und Titanium.

Neben den Anbietern von SDKs existieren Unternehmen, die anpassbare AR-Funktionalitäten ohne Programmieraufwand ermöglichen. An dieser Stelle kann beispielsweise Holobuilder genannt werden, worauf aus bereits unter Abschnitt 2.1.5 genannten Gründen nicht näher eingegangen wird.

Testversionen von diversen Frameworks sind meist kostenlos. Sollte dies nicht expli-

zit auf der Homepage des jeweiligen Distributors ersichtlich sein, kann eine Anfrage beim Herausgeber nützen. Wenn ein frei verfügbares Development-Kit zum Test zur Verfügung steht, sind in der Regel Unterlagen zur Verwendung der Software kostenlos verfügbar. Eine Ausnahme stellt ARLab dar, da außer der Dokumentation Hilfen wie Tutorien oder Quellcodebeispiele gebührenpflichtig sind[19].

Für manche NutzerInnen der Frameworks ist die Größe des Softwareherstellers in Anbetracht des Risikomanagements von Bedeutung. Nicht jeder Anbieter macht dazu Angaben, jedoch zeigen die verfügbaren Daten hohe Unterschiede auf. XZIMG Limited führt auf der eigenen Homepage zwei MitarbeiterInnen an, wobei PTC Inc., der Herausgeber von Vuforia, nach eigenen Angaben 6000 MitarbeiterInnen beschäftigt[39, 72].

3 Machbarkeitsanalyse ausgewählter SDKs

In diesem Kapitel werden die drei relevantesten Frameworks miteinander verglichen. Schwerpunkte sind die Dauer und Komplexität der Installation und Implementierung sowie Unterschiede im Verhalten der AR-Inhalte. Als Plattform dienen Android und iOS. Voraussetzungen sind die Entwicklungsumgebungen Android Studio und Xcode sowie installierte und konfigurierte Android- als auch iOS-SDKs. Die Installationsworkflows befinden sich im Anhang. Der Sourcecode der Projekte ist auf einem Datenträger abgespeichert, der dieser Arbeit beiliegt.

3.1 Anwendungsfall

Es wird zunächst ein Anwendungsfall definiert, der zu ersten praktischen Erfahrungen mit den Frameworks dient. Dabei sollen Gemeinsamkeiten und Unterschiede beim gesamten Workflow deutlich werden.

Zur Vergleichbarkeit der drei SDKs wird eine Funktionalität ausgewählt, die alle Frameworks unterstützen. Dabei handelt es sich um Image Recognition in Kombination mit AR-Inhalten. Sobald die Applikation ein Target erkennt, zeigt das Endgerät ein Bild im Vordergrund relativ zur Position des Targets an.

Als zu erkennendes, zweidimensionales Objekt wird ein Foto der Fachhochschule Salzburg verwendet. Der Abzug liegt für diverse Gegenüberstellungen (siehe Unterkapitel 3.5) in 20 cm x 10 cm vor. Als AR-Inhalt dient das Logo der Fachhochschule, das im oberen mittleren Bereich eingeblendet werden soll (siehe Abbildung 3.1).



Abbildung 3.1: Anwendungsfall Machbarkeitsanalyse

3.2 Kudan AR Engine

Dieses Unterkapitel befasst sich mit der sogenannten “Kudan AR Engine”, im Folgenden mit Kudan abgekürzt. Es werden die Installation unter Android als auch iOS beschrieben, ein Programm je Plattform mit diesem Framework erstellt und resultierende Ergebnisse dargelegt.

3.2.1 Installation

Die Homepage des Herausgebers legt die ersten Schritte zur Verwendung dieses Frameworks knapp, jedoch für Android- und iOS-EntwicklerInnen ausreichend detailliert dar. Weitere Einzelheiten zur nachfolgend beschriebenen Installation können den Homepages[73, 74] oder den Installationsworkflows von Kudan im Anhang entnommen werden.

Die Leitfäden beinhalten nur wenige Aufgaben, die in kurzer Zeit vollzogen sind. Nachdem zunächst ein neues Projekt mit Android Studio beziehungsweise Xcode angelegt wurde, ist es möglich das SDK einzubinden. Kudan Limited stellt das Framework über den eigenen Webauftritt¹ frei zur Verfügung. Die derzeit verfügbaren Versionen des SDKs sind für Android 1.4.2 beziehungsweise für iOS 1.4. Des Weiteren müssen benötigte Berechtigungen in den entsprechenden Dateien eingetragen werden. Kudan Limited sieht vor, dass der Compiler gewisse Teile des Frameworks nicht komprimiert.

Je nachdem, ob eine Test-Applikation erstellt werden soll oder das SDK gekauft wurde, setzt das Framework den passenden “Development License Key” zur Authentifizierung voraus. Der Schlüssel ist für Tests ohne Anmeldung frei verfügbar[75]. Weitere Informationen bezüglich der Testversion, sowie zu Preisen sind dem Abschnitt 2.1.6 zu entnehmen. Die zum “Development License Key” zugehörige “Application-ID” muss zwingend korrekt in der Datei `build.gradle` (Android) beziehungsweise im Projekt unter iOS angegeben werden. Ansonsten zeigt das Gerät keine AR-Inhalte an und gegebenenfalls tritt kein Fehler in der Applikation auf. Es ist zu beachten, dass diese Information nicht in der Anleitung von Kudan vermerkt ist.

Um die Funktionalitäten von Kudan schlussendlich zu verwenden, muss die anzuzeigende Android Activity von der Kudan-Activity “ARActivity” beziehungsweise der “ViewController” von “ARCameraViewController” (iOS) erben. Jegliche Anzeigen von Augmented Reality - Inhalten sollten laut Kudan in einer parameterlosen Methode namens “setup” (Android) beziehungsweise `setupContent` (iOS) statt finden.

¹<https://www.kudan.eu/download-kudan-ar-sdk/>

3.2.2 Implementierung

Die Implementierung des Anwendungsfalls beschreiben zum Teil die Quellen [76, 77].

Zunächst wird ein Target angelegt. Für diesen Einstiegspunkt stellt Kudan die Klasse `ARImageTrackable` bereit. Um das in Unterkapitel 3.1 beschriebene Szenario abzubilden, wird das Foto des Gebäudes der Fachhochschule Salzburg in Verbindung mit dem `ARImageTrackable` verwendet. Die Erkennung und Nachverfolgung des zweidimensionalen Objekts verwaltet der `ARImageTracker`, dem das `ARImageTrackable` hinzuzufügen ist.

Das anzuzeigende Bild, welches in diesem Fall ein Logo der Fachhochschule Salzburg ist, wird in Form der Kudan-spezifischen Klasse `ARImageNode` instantiiert und dem Target hinzugefügt.

3.2.3 Ergebnisse

Dieses Unterkapitel geht auf die Ergebnisse bei der Verwendung von Kudan für den Anwendungsfall ein.

Installation und Implementierung

Die Installation und die Implementierung des Anwendungsfalls konnten mit Kudan in wenigen Arbeitsschritten und dadurch geringem Arbeitsaufwand durchgeführt werden. In Java wie auch in Objective-C sind sechs Quellcode-Zeilen nötig. Die aufzuwendende Zeit kann je nach Erfahrungsstand variieren.

Verwendung der Applikation

Die Applikation startet auf einem Mittelklasse-Smartphone in unter drei Sekunden und erkennt Targets in weniger als einer Sekunde.

Für die initiale Erkennung des Targets sind circa 75 Prozent des Bildes nötig. Das Einblenden des Logos wird jedoch nach der Initialisierung auch bei geringem Abstand und extremen Winkeln unterstützt. Beim Bild der Fachhochschule liegt die minimale Distanz bei circa 6 cm. Als Target wird ein Ausdruck von 20 cm x 10 cm verwendet.

Das Logo bleibt bei mäßig starken Änderungen der Perspektive stabil auf dem Target angeheftet.

Fazit

Im Allgemeinen macht Kudan den Eindruck, als wäre es für diesen und ähnliche Anwendungsfälle entwickelt worden. Die Komplexität ist aus Nutzersicht gering und der Anwendungsfall lässt sich gut abbilden. In diesem Fall kann Kudan somit ohne Einschränkungen empfohlen werden.

3.3 Vuforia

Im Folgenden wird die Vuforia SDK in Version 6.2.6 samt Beispielprogrammen unter Android und iOS installiert. Daraufhin sind die Implementierung des Anwendungsfalls (siehe Abschnitt 3.1) und die Ergebnisse zusammengefasst. Als Abkürzung für die Vuforia SDK dient gegebenenfalls Vuforia.

3.3.1 Installation

Der Herausgeber PTC Inc. beschreibt in einer “Getting Started”-Anleitung die Installation von Vuforia in Verbindung mit Beispielprogrammen. Dadurch haben EntwicklerInnen die Möglichkeit, benötigte Klassen und Methoden zu nutzen und nicht verwendete zu löschen[78, 79]. Aufgrund der Empfehlung von PTC Inc. wird jenes Vorgehen im weiteren Verlauf beschrieben.

Der zu Beginn erforderliche Download² des Frameworks ist nur für registrierte NutzerInnen möglich. Die kostenlose Registrierung ist auch für einen API-Schlüssel zum Verwenden der SDK erforderlich. Nach dem Download der Software wird das heruntergeladene Archiv in das Verzeichnis kopiert, in dem sich das Framework befindet.

NutzerInnen können die Beispielprogramme herunterladen³ und zum lokalen Unterverzeichnis namens “Samples” des kopierten Vuforia-Verzeichnisses hinzufügen.

Nach der Generierung eines individuellen API-Schlüssels wird dieser zum Projekt hinzugefügt. Somit ist die Installation abgeschlossen, für Details siehe Informationen von PTC[78, 79] oder Workflows im Anhang.

²<https://developer.vuforia.com/downloads/sdk>

³<https://developer.vuforia.com/downloads/samples>

3.3.2 Implementierung

Als Ausgangspunkte für die Realisierung des Anwendungsfalls (siehe Unterkapitel 3.1) wird, wie von PTC Inc. empfohlen, ein Beispielprojekt verwendet. Es ist möglich, das Paket “Core Features” von der Website⁴ herunterzuladen. Im weiteren Verlauf findet die Version 6.2.10 ihre Verwendung.

Einen ähnlichen Anwendungsfall als den in Abschnitt 3.1 beabsichtigten, beinhaltet das Teilprojekt “Image Targets”. Hier werden dreidimensionale Teekannen vor definierten Hintergründen angezeigt. Aufgrund der Verwendung von 3D-Objekten sind diverse Änderungen nötig.

Da Vuforia keine zweidimensionalen Gebilde nutzt, sondern für drei Dimensionen konzipiert wurde, ist die Vorgehensweise der Darstellung eine Andere. Um 2D-Bilder als 3D-Objekte zu visualisieren, wird zunächst eine neue Klasse von Körpern erzeugt. Diese bildet eine Fläche als Quader mit vernachlässigbarer Tiefe ab[80]. Daraufhin kann das Bild, in diesem Fall das Logo der Fachhochschule Salzburg, auf einer Instanz der Quader-Klasse angezeigt werden.

Im Ausgangszustand werden transparente Anteile von Darstellungsschichten schwarz abgebildet. Dies lässt sich jedoch mit einer Anweisung beheben[81].

Über ein Webinterface können Datenbanken erstellt und mit Targets ausgestattet werden. Das erstellte File wird heruntergeladen und anschließend in das Projekt eingebunden[82].

⁴<https://developer.vuforia.com/downloads/samples?d=windows-31-22-4816>

3.3.3 Ergebnisse

Dieses Unterkapitel geht auf die Ergebnisse der Installation und der Anwendungsfallimplementierung mit Hilfe von Vuforia ein.

Installation und Implementierung

Bei der Realisierung wird klar, dass ein Einstieg ohne Beispielprojekt zeitaufwändig ist. PTC kapselt nur einen Teil der Komplexität vor NutzerInnen der SDK ab. Dadurch erhöht sich der Einarbeitungsaufwand, aber es bietet auch mehr Möglichkeiten zur Modifikation. Für diesen häufig verbreiteten Anwendungsfall sind jedoch, trotz Beispielprojekt, einige Anpassungen nötig.

Verwendung der Applikation

Auf einem Smartphone der Mittelklasse dauert der Start der Applikation und die Ladezeit bis erste Targets fokussiert werden können circa sechs Sekunden. Zur Erkennung eines Bildes und der darauf folgenden Anzeige eines Objekts benötigt die Applikation bei diesem Anwendungsfall weniger als 300 ms.

Vuforia erkennt Targets früh. Es müssen initial weniger als 50 Prozent eines Bildes im Fokus der Kamera sein, um dieses zweifelsfrei zu erkennen. Sobald ein Objekt registriert wurde, können Perspektiven eingenommen werden, aus welchen keine initiale Identifizierung mehr möglich wäre. Vuforia unterstützt dabei kleine Winkel und minimale Entfernungen von circa 8 cm beim Target des Anwendungsfalls mit der Größe von 20 cm x 10 cm.

Fazit

Vuforia eignet sich für Anwendungsfälle mit Fokus auf Bilderkennung und gegebenenfalls einhergehender Einblendung von AR-Inhalten. Die hohe Anpassbarkeit geht jedoch mit erhöhter Komplexität einher, speziell im Hinblick auf die Möglichkeiten im dreidimensionalen Raum.

3.4 Wikitude

Dieses Unterkapitel stellt einen Überblick über die Installation des Frameworks in Version 6.0.1, die Implementierung des Anwendungsfalls und die resultierenden Ergebnisse dar. Die Wikitude SDK, oder gegebenenfalls als Wikitude abgekürzt, liegt für Android und iOS als native Ausführung sowie als JavaScript-API vor. Aufgrund des erweiterten Funktionsumfangs findet im Folgenden die JavaScript-Ausführung ihre Verwendung. Das native Framework unterstützt zwar die Erkennung von Bildern, jedoch nicht die Positionierung eines Objekts relativ zum Target. Die native SDK ist somit für den Anwendungsfall dieses Kapitels nicht die erste Wahl. Für weitere Unterschiede zwischen nativem und JavaScript-Framework siehe Quelle [83].

3.4.1 Installation

Die initiale Anbindung des Development-Kits, worauf dieser Abschnitt eingeht, ist der Beschreibung auf der Homepage des Anbieters zu entnehmen[83]. Der detaillierte Installationsworkflow befindet sich im Anhang.

Nach dem Download der SDK⁵ wird diese dem Projekt neben gegebenenfalls benötigten weiteren Dateien hinzugefügt. Im Anschluss dessen findet die Eintragung der sogenannten “Application-ID” statt. Ausschließlich bei gekauften Ausführungen ist diese ID von Nöten. Es bedarf der Erteilung diverser Berechtigungen, damit die Applikation auf Hardware des Gerätes zugreifen kann.

Als Eigenheit der JavaScript-API empfiehlt die Wikitude GmbH unter Android und iOS die Verwendung der Klasse `com.wikitude.architect.ArchitectView`. Die Installation ist nach der Einbindung des API-Schlüssels fertiggestellt.

⁵<https://www.wikitude.com/download/>

3.4.2 Implementierung

Dieser Abschnitt beschreibt die Implementierung des Anwendungsfalls aus Unterkapitel 3.1) unter Verwendung der JavaScript-SDK. Als Ausgangspunkt jener Applikation wird ein Beispielprojekt von Wikitude verwendet.

Das Unternehmen liefert das Projekt bereits mit dem Framework⁶ der Version 6.0.1 aus. Der Anwendungsfall ist in “Image Recognition - Image On Target” implementiert.

Nachdem alle nicht benötigten Features für diese Realisierung entfernt sind, wird als Target das Foto der Fachhochschule mit Hilfe des Webinterfaces⁷ angelegt. Dazu muss das Bild hochgeladen, in eine Datenbank eingefügt und selbige heruntergeladen werden.

Die folgende Implementierung erfolgt im Gegensatz zum Betriebssystem-Standard nicht in Java beziehungsweise Objective-C/Swift, sondern in JavaScript. Das Target in Form der Datenbank-Datei wird in das Projekt eingebunden und verwendet. Das Logo der Fachhochschule muss als `AR.Resource` angelegt werden.

Um das Größenverhältnis der beiden Objekte an den Anwendungsfall anzupassen, erhält das Logo einen Skalierungsfaktor von 0.25.

⁶https://wikitude-web-hosting.s3.amazonaws.com/sdk/6_0_1/WikitudeSDK_Android_6-0-1_2017-02-16_08-59-59.zip

⁷<https://targetmanager.wikitude.com/>

3.4.3 Ergebnisse

Dieses Unterkapitel beschreibt die Ergebnisse, die sich bei der Verwendung von Wikitude für den Anwendungsfall ergeben.

Installation und Implementierung

Zunächst wurde die native Android-SDK von Wikitude verwendet. Diese unterstützt EntwicklerInnen, wie bereits in Sektion 3.4 erwähnt, nur mit Computer Vision und enthält keine Funktionalität zur Anzeige und Positionierung von Inhalten im Vordergrund des Targets. Zur Vergleichbarkeit der drei Frameworks wird für diesen Abschnitt die JavaScript-SDK eingesetzt.

Diese Ausführung des Frameworks unterstützt Quellcode der Sprache JavaScript für die Anwendungslogik. Der restliche Teil des Projekts setzt sich aus Android - beziehungsweise iOS - typischen Dateien zusammen. Bei der Realisierung werden circa 30 Zeilen Sourcecode in JavaScript benötigt.

Die Komplexität des Projekts hält sich demnach in Grenzen. Die Implementierung des Anwendungsfalls ist dadurch sowohl ohne Beispielprojekt mit einer gewissen Einarbeitungszeit möglich.

Verwendung der Applikation

Das Starten der Applikation und die Initialisierung dauern auf einem Mittelklasse-Smartphone weniger als vier Sekunden. Die Erkennung des Targets im Rahmen des Anwendungsfalls wird im Durchschnitt unter 300 ms bewerkstelligt. Bei diesem Image Matching - Prozess muss sich jedoch beinahe das gesamte Target im Fokus der Kamera befinden.

Sobald das Bild erkannt wird, zeigt das Endgerät den AR-Inhalt bis zu einer minimalen Entfernung von 9 cm an. Als Target dient die Front der Fachhochschule Salzburg auf einem 20 x 10 cm großen Foto. Nach der Erkennung werden zudem extreme Winkel unterstützt.

Fazit

Die Verwendung von JavaScript bringt gegebenenfalls eine Umstellung für EntwicklerInnen mit sich. Der Sourcecode bietet jedoch die Möglichkeit der Wiederverwendung auf allen Plattformen. Dadurch wird der Workflow der Implementierung stark verkürzt. Als einziger negativer Punkt ist die benötigte Fokussierung des gesamten Targets hervorzuheben. Somit eignet sich Wikitude nur gut für Anwendungsfälle mit kleineren oder aufgeteilten Targets.

3.5 Vergleich der SDKs

Im Folgenden werden die Frameworks miteinander verglichen. Diese Gegenüberstellung betrifft die Vor- und Nachteile sowie Eigenheiten, die bei den Realisierungen des Anwendungsfalls aufgetreten sind. Zur Vergleichbarkeit wird stets das selbe Foto der Fachhochschule Salzburg sowie das selbe Android-Gerät verwendet.

Dauer der Realisierung

Generell erfolgt die Installation sowie Implementierung bei allen drei SDKs in wenigen Arbeitsschritten. Bei vorhandenen Programmierkenntnissen im Bereich der jeweiligen Plattform und unter Verwendung entsprechender Anleitungen dauert die Einrichtung weniger als 15 Minuten je Framework. Kudan ist im Vergleich zu beiden anderen Development-Kits schneller eingerichtet.

Bei der Implementierung des Anwendungsfalls nach dem beschriebenen Vorgehen (siehe Kapitel 3) gibt es deutlichere Unterschiede im Zeitaufwand beziehungsweise den Arbeitsschritten. Da die Dauer der Implementierung besonders bei Problemen stark von der Erfahrung der Entwicklerin oder des Entwicklers abhängt, werden die Frameworks nach Anzahl und Aufwand der Arbeitsschritte miteinander verglichen.

Bei der Nutzung von Kudan kann der Anwendungsfall mit Abstand am schnellsten implementiert werden. Nach der Installation sind sechs Zeilen Sourcecode sowie die beiden Bilddateien nötig.

Die Realisierung mit der Wikitude Javascript SDK dauert zwar länger als mit Kudan, jedoch ist der Zeitaufwand signifikant geringer als mit Vuforia. Bei letzterem Framework besteht der hauptsächliche Aufwand darin, eine Klasse zu schaffen, welche ein zweidimensionales Bild als dreidimensionales Objekt darstellt. Falls die Implementierungen auf mehreren Plattform ausführbar sein sollen, bietet sich Wikitude mit der JavaScript-SDK durch wiederverwendbaren Quellcode an.

Komplexität

Die Komplexität der Verwendung eines Frameworks bestimmt maßgeblich die Einarbeitungszeit. Da dies stark mit der Dauer der Realisierung zusammen hängt, tritt hier zudem Kudan positiv hervor. Wird von gleicher Expertise der Entwicklerin oder des

Entwicklers in allen verwendeten Programmiersprachen ausgegangen, ist die Realisierung mit Vuforia deutlich komplexer als mit Wikitude.

Ein entscheidender Punkt bei der Wahl des Frameworks ist, dass mit der Vielschichtigkeit auch die Möglichkeiten steigen. Somit sinkt die Anpassbarkeit tendenziell bei geringerer Komplexität.

Augmented Reality - Eigenschaften

Die Dauer der Erkennung des Targets und die einhergehende Anzeige des Logos liegt bei Vuforia und Wikitude unter 300 ms. Mit Kudan kostet dieser Prozess nahezu eine Sekunde.

Ein weiterer Aspekt der AR-Funktionalitäten ist die benötigte Abdeckung des Targets durch die Kamera. Hierbei handelt es sich um den prozentualen Anteil des Bildes, der über die Kamera des Geräts aufgenommen werden muss, um das Target zu erkennen. Bei weniger als 100 Prozent des gesamten Bildes im Fokus des Geräts folgern Kudan und Vuforia das Foto. Wikitude setzt nahezu das komplette Bild voraus. Kudan erkennt das Target bei circa 75 Prozent, Vuforia ist auf weniger als 55 Prozent angewiesen.

Alle drei Frameworks können erkannte Bilder bei extremen Blickwinkeln und geringen Entfernungen verfolgen und AR-Inhalte anzeigen. Vuforia sticht bei den niedrigsten und höchsten Blickwinkeln hervor, wobei die beiden anderen SDKs gleichauf sind. Kudan führt bei einer minimalen Entfernung zum Target von 6 cm. Vuforia und Wikitude erreichen eine Anzeige des Logos bis circa 8 beziehungsweise 9 cm.

Fazit

Alle drei Frameworks haben bei dieser Realisierung ihre Stärken und Schwächen, welche je nach Anwendungsfall abgewägt werden müssen. Für geringere Anforderungen und schnelle Ergebnisse eignet sich Kudan. Bei komplexeren Anwendungsfällen empfiehlt sich eine Auswahl je nach Fokus der Applikation, Anzahl der zu unterstützenden Betriebssysteme sowie weiterer geplanter Funktionalitäten (siehe Unterkapitel 2.2).

Im Hinblick auf die Ziele dieser Arbeit erfolgt die Implementierung eines Prototyps mit Hilfe der Wikitude JavaScript SDK. Die Umsetzung wird im folgenden Kapitel beschrieben.

4 Umsetzung eines Prototyps

Dieses Kapitel beschreibt die Erstellung einer Referenzimplementierung. Der Prototyp dient zur Erarbeitung und Validierung eines passenden Workflows für die Entwicklung von AR-Applikationen auf Android und iOS. Dabei wird ein geeigneter Anwendungsfall gesucht, das Vorgehen erläutert und die Ergebnisse dargestellt.

4.1 Anwendungsfall

Im weiteren Verlauf werden die bedeutendsten funktionalen Eigenschaften der Software beschrieben. Der Fokus liegt auf der Abbildung eines verbreiteten AR-Anwendungsfalls. Dabei werden Unterschiede zwischen den Plattformen und die Zeitersparnis durch die plattformübergreifende Implementierung deutlich.

Image Matching

Als Ausgangspunkt eignet sich ein zweidimensionales Bild. Dieses kann in der Praxis beispielsweise in einer Zeitschrift abgedruckt sein. Sobald die Software das Target erkennt, wird ein dreidimensionales Objekt auf dem Smartphone-Bildschirm angezeigt. Die Applikation blendet den Körper direkt auf dem Bild ein und fixiert diesen. Das ermöglicht es NutzerInnen, durch die Verwendung anderer Kamerawinkel, diverse Aspekte des virtuellen Objekts zu inspizieren (siehe Abbildung 4.1).



Abbildung 4.1: Anwendungsfall Prototyp

3D - Transformation

Die Anzeige des virtuellen Gegenstands kann durch Touch-Gesten verändert werden. Zur Rotation um die Z-Achse und der Skalierung des Objekts benötigen AnwenderInnen jeweils zwei Finger.

Objekt Modifikation

Neben einer Änderung der Darstellung können Teile des Körpers umgestaltet werden. Dabei ist es möglich, einzelne Farben des Objekts anzupassen. Beim Berühren eines bestimmten Details des Gegenstands führt die Applikation Operationen aus. Diese sind die Animierung des Fragments, das Ersetzen des Abschnitts durch einen weiteren sowie das Abspielen von Klängen.

Extended Tracking

Das von Wikitude unterstützte Extended Tracking verbessert die Verfolgung des Targets nach dem Erkennen, auch wenn dieses sich nicht mehr im Sichtfeld befindet. Außerdem wird die Notwendigkeit für ein Neuladen des Objekts reduziert. Für weitere Informationen zu Extended Tracking siehe Abschnitt 2.1.7.

Object Matching

Zusätzlich zum zweidimensionalen Bild kann auch ein dreidimensionales Target verwendet werden. Dadurch wird es der Anbieterin oder dem Anbieter der Applikation ermöglicht, Gegenstände zu erkennen und die virtuellen Pendants dazu anzuzeigen.

4.2 Vorgehen

Im Folgenden wird die konkrete Implementierung des Prototyps dargestellt. Die Anleitung zu 3D Modellen (siehe Dokumentation[84]) und der dazugehörige Quellcode¹ dienen als Basis. Die erstellte Software befindet sich auf dem Datenträger, der dieser Arbeit beiliegt.

Projektinitialisierung

Zunächst wird sowohl für Android als auch für iOS ein neues Projekt angelegt. Diese beiden betriebssystemspezifischen Strukturen greifen auf den selben Dateordner zu. In jenem Verzeichnis befinden sich die betriebssystemunabhängigen Komponenten.

Dadurch kann ein Großteil der Applikation einmal entwickelt und auf allen Plattformen verwendet werden.

3D - Transformation

Interaktionen mit mehreren Fingern auf dem Objekt lösen diverse Events aus. Die Dokumentation von Wikitude behandelt Ereignisse für Rotation, Skalierung und Verschiebung im zweidimensionalen Raum[85]. Diese Logik wird mit einigen Anpassungen im Dreidimensionalen eingesetzt.

Objekt Modifikation

Das Framework bietet die Möglichkeit, Events beim berühren ausgewählter Teile des Körpers auszulösen. Beim Touch auf beispielsweise die Motorhaube eines Fahrzeugs spielt das Smartphone ein Motorengeräusch mit Hilfe der Klasse `AR.Sound` ab.

Die Farbe des Körpers soll änderbar sein. Die Modifikation lässt sich zum einen durch das Ersetzen des gleichen Objekts mit anderer Farbe bewerkstelligen. Zum anderen können mit Hilfe von 3D-Grafiksoftware Texturen hinterlegt werden. Diese Variante der Farbänderung wird von Wikitude empfohlen, findet aber im Prototypen aufgrund der in Unterkapitel 4.3 erwähnten Punkte keine Anwendung. In beiden Fällen ist 3D-Grafiksoftware sowie das zugehörige Know-How notwendig. Als Voraussetzung gelten diese zudem bei den folgenden Modifikationen.

¹<https://github.com/Wikitude/wikitude-sdk-samples>

Animationen von Teilen eines Objekts wie beispielsweise das Öffnen von Türen eines Fahrzeugs werden mit 3D-Grafiksoftware erstellt. Das Programm speichert die Bewegung zusammen mit dem Körper. Dadurch konvertiert der Wikitude 3D Encoder die Sequenzen zusätzlich und stellt sie der Applikation innerhalb einer Wikitude 3D Format (W3T) - Datei zur Verfügung. Wie auch beim Abspielen von Tönen können die Animationen beim Berühren bestimmter Elemente des Objekts gestartet werden.

Neben der Bewegung von Teilen des Körpers dient die Grafiksoftware auch zum Auswechseln von Komponenten. Bestandteile, wie etwa Felgen und Reifen eines Fahrzeugs, werden bis zur Unsichtbarkeit verkleinert. Daraufhin zeigt die Software jeweils ein neues Rad am Platz des Alten an.

Extended Tracking

Eine Änderung von normalem Tracking zu Extended Tracking bedarf mit Wikitude dem Austausch von drei Zeilen Sourcecode. Neben dem Setzen einer booleschen Variablen ist es nötig, zwei spezielle Events zu verwenden[85].

Object Matching

Die Erkennung eines dreidimensionalen Objekts ist mit Wikitudes JavaScript-Framework nicht möglich. Dieses Feature bietet lediglich die native SDK.

4.3 Ergebnisse

Dieses Unterkapitel behandelt diverse Erkenntnisse, die während der Implementierung des Prototyps gewonnen werden konnten. Die Beobachtungen fließen zudem soweit möglich in den Entwicklungsworkflow mit ein.

Entwicklungsumgebung

Die Programmiersprache Javascript, die das Framework Wikitude vorgibt, wird nicht aktiv von Android Studio oder Xcode unterstützt. Dadurch sind diverse Hilfsmittel der Entwicklungswerkzeuge nicht einsetzbar.

Nur einige Klassen kennzeichnet das Integrated Development Environment (IDE) farblich. Eine automatische Quellcode-Vervollständigung greift ausschließlich für bereits verwendete Ausdrücke. Die Entwicklungsumgebungen validieren die Syntax des Javascript-Codes nicht und zeigen keine Kompilier- oder Laufzeitfehler an. Darüber hinaus ist Debuggen mit diesen Tools nicht möglich.

Es gibt jedoch zusätzliche Mittel, die die Fehlersuche unterstützen. Wikitude bietet das ARchitect Desktop Environment (ADE), um im Desktop-Browser eine Übersicht instantiiertter Objekte zu erhalten und Events auszulösen[86]. Eine weitere Möglichkeit ist das Debuggen auf dem Gerät mit der Klasse `WebView`. Dieses Vorgehen ist nur auf Android-Geräten verfügbar und setzt den Internet-Browser Chrome voraus. Weitere Details können Quelle [87] entnommen werden.

Verwendung von 3D-Grafiken

Der Anwendungsfall erfordert diverse Modifikationen von dreidimensionalen Objekten. Fundierte Erfahrungen mit 3D-Grafiksoftware sind dadurch eine Grundvoraussetzung. Außerdem treten Eigenheiten in Verbindung mit dem Framework auf, die im weiteren Verlauf dargelegt werden.

Die AR-Umgebung verwendet eine eigene Beleuchtung, wodurch keine Lichtquelle bei der Grafikerstellung hinzugefügt werden sollte.

In bestimmten Fällen exportiert die Software namens “Blender” fehlerhafte Animationen. Um Modelle vor der Konvertierung zu validieren, empfiehlt Wikitude das Programm “FBX Reviewer”[88].

Objekte, die in der 3D-Grafiksoftware “Cinema4D” als unsichtbar gekennzeichnet sind, werden im 3D Encoder und in der Wikitude-Applikation angezeigt. Eine Alternative für das Verbergen von Komponenten sind Skalierungsfaktoren nahe Null.

Zur Änderung der Farbe eines Körpers empfiehlt Wikitude die Verwendung von Texturen. Diese abweichenden Farbtöne können jedoch im 3D Encoder nicht angezeigt werden[89]. Der Prototyp umgeht dieses Problem, indem die Businesslogik zwei unterschiedlich-farbige Objekte instantiiert, die Koordinaten zentral speichert und nur einen dieser Körper anzeigt.

Plattformabhängigkeit

Der betriebssystemunabhängige Teil wird in Javascript beschrieben und ist ohne Abänderung unter iOS und Android lauffähig. Bei der Implementierung des Anwendungsfalls tritt je Plattform jedoch verschiedenartiges Verhalten auf. Wird die gewünschte Reaktion unter Android erreicht, impliziert der selbe Sourcecode nicht die gleiche Abhandlung unter iOS.

Als Beispiel dient zum einen die fehlende Klanguausgabe von `AR.Sound.play()`, welche `onLoad()` durch vorheriges Laden behebt. Zum anderen sei das Ruckeln bei der Transformation mit Touchgesten genannt[90].

Weitere Auffälligkeiten

Der Wikitude 3D Encoder unterstützt neben dem Hauseigenen nur das Dateiformat Filmbox (FBX).

Bei der Verwendung von Extended Tracking erkennt das Framework den Blickwinkel nicht verlässlich. Dadurch werden dreidimensionale Objekte zwar relativ zum Target dargestellt. Die Anzeige erfolgt jedoch nicht nach der Position der Kamera im Raum. Extended Tracking eignet sich somit ausschließlich für das Visualisieren von zweidimensionalen Bildern.

Fazit

Mittels dieser Erkenntnisse können folgende Punkte für einen optimierten Workflow extrahiert werden:

In Anbetracht der Möglichkeit des Debuggens empfiehlt sich die Entwicklung unter Android. Um jedoch Inkonsistenzen zu vermeiden, muss stets zusätzlich auf iOS getestet werden.

Der Zeitaufwand bei der Implementierung für zwei Betriebssysteme mit Javascript halbiert sich nur bedingt. Es können plattformabhängige Unterschiede auftreten, die Zeit zur Behebung erfordern. Zudem ist zu beachten, dass jeweils für iOS und Android ein Projekt zum starten der gemeinsamen Logik entwickelt werden muss.

5 Schlussbetrachtung

Dieses abschließende Kapitel stellt die Ergebnisse der Arbeit dar. Nach der Diskussion der Resultate erfolgt ein Ausblick wie sich das Thema Augmented Reality entwickeln könnte.

5.1 Ergebnisse dieser Arbeit

Die wichtigsten Feststellungen sind in die drei Hauptkapitel untergliedert. Einen weiteren Punkt stellt der Workflow dar, auf den gesondert eingegangen wird.

Theoretischer Vergleich

Der Vergleich der 10 relevantesten Frameworks verdeutlicht diverse Unterschiede zwischen den Hilfsmitteln. Neben Komponenten zur Programmierung eigener Applikationen können auch Software-Lösungen bezogen werden, die ohne Eigenentwicklung verwendbar sind.

Aufgrund des breiten Anwendungsgebiets von AR setzt jeder Hersteller andere Schwerpunkte. Zum einen gibt es Ansätze, so viele Anwendungsbereiche wie möglich abzudecken. Zum anderen spezialisieren sich manche Anbieter auf einzelne Einsatzgebiete. Jedoch konnte kein Framework identifiziert werden, das alle verbreiteten Features von AR-Anwendungen anbieten würde.

Einige Hersteller unterteilen die angebotenen Funktionalitäten in mehrere Komponenten. Die Aufteilung kann beispielsweise nach Anwendungsbereich, Ressourcenbedarf oder Programmiersprache erfolgen.

Zur Validierung der SDK stellt nahezu jeder Anbieter eine kostenlose Version seiner Software zur Verfügung. Gegebenenfalls muss zuvor eine Anfrage an den Support erfolgen.

Machbarkeitsanalyse

Für einen, in den meisten Fällen anwendbaren, Workflow liegt das Augenmerk auf Komponenten mit möglichst vielen Funktionalitäten. Somit heben sich Kudan, Vuforia und Wikitude (JavaScript SDK) hervor.

Die Installation der Frameworks ist in wenigen Arbeitsschritten vollzogen. Je nach Fokus und Komplexität der Software variiert der Aufwand für die Implementierung des Anwendungsfalls. Das gewählte Szenario, das die Image Recognition impliziert, wird von den drei SDKs unterstützt.

Jede Software erkennt das Target, wenn sich dieses im Sichtfeld der Kamera befindet. Bei Verwendung von Wikitude muss sich das zu erkennende Bild komplett im Fokus befinden, bei Kudan und Vuforia nur zu circa 75 beziehungsweise 55 Prozent.

Kudan sticht als Tool mit dem geringsten Zeitbedarf für diesen Einsatz hervor. Die Realisierung kann mit sechs Zeilen Sourcecode durchgeführt werden.

Vuforia legt das Augenmerk auf dreidimensionale Objekte und verlangt dadurch für den Anwendungsfall erheblich mehr Anpassungen.

Wikitude kann in Anbetracht des Aufwands zwischen Kudan und Vuforia eingeordnet werden. Die Logik wird nicht in nativer Programmiersprache, sondern in JavaScript beschrieben.

Prototyp

Aufgrund der Anzahl an Features und der Wiederverwendbarkeit von JavaScript, erfolgt die Implementierung des Prototyps mit Hilfe von Wikitude.

Die Verwendung von JavaScript schränkt die Möglichkeiten der Entwicklungsumgebungen ein. Einige Funktionalitäten, die bei nativem Sourcecode verfügbar sind, werden nicht unterstützt. Dazu zählen Debuggen, Quellcode Vervollständigung und Quellcode-Syntax Validierung. Die Fehlersuche zur Laufzeit ermöglicht die Verwendung von Chrome und Android.

Zu beachten sind in diversen AR-Anwendungsfällen zudem dreidimensionale Objekte. Diese müssen gekauft oder erstellt und gegebenenfalls modifiziert werden. Der Bedarf an Know-How und Ressourcen steigt somit.

Workflow

Der optimale Workflow zur Erstellung von iOS- und Android-Applikationen setzt zunächst jeweils ein Projekt mit Hilfe der jeweiligen Entwicklungsumgebung voraus. Zur

automatisierten Übernahme von Änderungen wird die Logik für beide Plattformen in einer zentralen JavaScript-Datei festgelegt und von beiden Projekten referenziert.

In Anbetracht der Möglichkeit des Debuggens empfiehlt sich die Entwicklung unter Android. Um jedoch Inkonsistenzen zu vermeiden, muss stets zusätzlich auf iOS getestet werden.

Der Zeitaufwand bei der Implementierung für zwei Betriebssysteme mit JavaScript halbiert sich nur bedingt. Es können plattformabhängige Unterschiede auftreten, die Zeit zur Behebung erfordern. Zudem gilt zu beachten, dass jeweils für iOS und Android ein Starter in nativer Programmiersprache entwickelt werden muss.

5.2 Diskussion der Ergebnisse

Im weiteren Verlauf legt dieses Unterkapitel die positiven und negativen Aspekte ausgewählter Ergebnisse dar.

Auswahl der SDKs

Die Selektion der Frameworks bis zur Festlegung auf drei Komponenten für die Machbarkeitsanalyse erfolgt nach der Anzahl der Funktionalitäten. Die SDKs mit den meisten Features werden bevorzugt. Dadurch steigt die Wahrscheinlichkeit für eine Verwendung in der Praxis. Das Vorgehen bringt sowohl Vor- als auch Nachteile mit sich.

Die Anwendungsbereiche von AR sind vielseitig. Zum einen bietet kein Framework alle möglichen Anforderungen. Zum anderen setzt die Abdeckung eines Features nicht voraus, dass der Anwendungsfall zufriedenstellend unterstützt wird. Aufgrund bereits genannter Gründe kann es notwendig sein, weitere SDKs in ein Projekt einzubinden.

Die Beschränkung auf ein einziges Framework bringt eine Reduktion der Komplexität mit sich. Diese Verringerung wirkt auf unterschiedliche Weise. Unter anderem durch die Senkung der Anzahl von Support-AnsprechpartnerInnen oder die Kostenersparnis bei Softwarekomponenten und Anbindungen.

Für den Fall, dass die Verwendung von Kudan, Vuforia oder Wikitude in einem individuellen Fall nicht zielführend sein sollte, stellt das Unterkapitel 2.2 eine Übersicht über weitere Hilfsmittel zur Verfügung. Eine Referenzimplementierung mit Hilfe von jedem Framework würde die Vorgaben für diese Arbeit überschreiten.

JavaScript

Das Wikitude JavaScript SDK ermöglicht es, plattformunabhängigen Sourcecode zu verwenden.

Daraus ergeben sich zudem negative Aspekte. Die Entwicklungsumgebungen Xcode und Android Studio wurden nicht für JavaScript entwickelt und bieten deshalb für die Programmiersprache nur bedingt Unterstützung. Einen weiteren Nachteil kann die Expertise bedeuten. Die Sprache erfordert für die Verwendung spezielles Know-How.

Als zusätzlicher negativer Aspekt wird die fehlende native Kompilierung von JavaScript

gesehen. Der Sourcecode muss in einer eigenen Umgebung ablaufen. Aufgrund des Mehraufwands können speziell bei komplexen Projekten Leistungseinbußen entstehen. Bei der Implementierung des Prototyps wurden jedoch keine Mängel festgestellt.

Die Vorteile von JavaScript stehen hauptsächlich in Verbindung mit der nicht vorhandenen Betriebssystembindung. Außerdem wird zur Erstellung von Logik kein Wissen über native Entwicklungssprachen benötigt. Ferner ist ein Einsatz des Sourcecodes auf anderen Plattformen als Android und iOS möglich.

Um Alternativen zu JavaScript aufzuzeigen, wurden weitere Frameworks verglichen und Referenzimplementierungen erstellt. Die Vorteile im Bezug auf den Workflow überwiegen jedoch.

Anwendungsfall

Der Anwendungsfall mit dreidimensionalen Objekten bietet positive und negative Aspekte.

Letztere äußern sich in der Art des Aufwands. Der erhebliche Zeitbedarf für Erstellung, Modifikation und Animation von Objekten steht nur bedingt in Zusammenhang mit den Zielen dieser Arbeit. Zudem stellt der Spezialfall nur einen kleinen Teil des Einsatzgebiets dar.

Die Verwendung von Körpern dient jedoch zur Identifizierung von generellen Hürden, die auch bei anderen Anwendungsfällen auftreten. Dieses Einsatzgebiet wird außerdem von diversen Softwareherstellern adressiert.

Trotz der Nachteile eignet sich der Fall zur Überprüfung der Plattformunabhängigkeit sowie für Einblicke in die 3D-Modellierung.

5.3 Zusammenfassung

AR-Applikationen sind weit verbreitet und auf einer Vielzahl von Smartphones einsetzbar. Es gibt unterschiedlichste Anwendungsgebiete und Funktionalitäten. Kein Framework unterstützt alle Features und jedes SDK hat einen anderen Fokus. Dabei variieren unter anderem Einsatzgebiet, Anzahl der Funktionen, Leichtgewichtigkeit, Art der Anbindung und Cloud-Nutzung.

Das Hauptaugenmerk der Software-Komponente bestimmt maßgeblich Komplexität und Wiederverwendbarkeit der Realisierung eines konkreten Anwendungsfalls.

Wikitude JavaScript SDK

Eine Abdeckung der meisten Ausstattungsmerkmale sowie der Wiederverwendbarkeit von Quellcode ermöglicht die Wikitude JavaScript SDK. Die Verwendung einer zusätzlichen, nicht nativen Programmiersprache bringt sowohl Vor- als auch Nachteile mit sich.

Positive Aspekte sind Wiederverwendbarkeit und Reduzierung des nötigen plattform-spezifischen Know-Hows. Als negativ einzustufen sind eingeschränkte Features der Entwicklungsumgebungen (Debuggen, Quellcode Vervollständigung und Quellcode-Syntax Validierung).

Workflow

Eine Entwicklung von AR-Applikationen lässt sich mit Hilfe eines Workflows definieren. JavaScript ermöglicht bei zwei Betriebssystemen eine erhebliche Einsparung an Sourcecode gegenüber Frameworks mit nativen Programmiersprachen. Bei zwei Plattformen kann diese Reduktion bis zu 50 Prozent betragen. Dennoch gilt zu beachten, dass je nach Betriebssystem unterschiedliches Verhalten auftreten kann. Das Testen auf Android sowie iOS ist deshalb unabdingbar.

5.4 Ausblick in die Zukunft von Augmented Reality

Das Forschungs- und Beratungsunternehmen Gartner sieht für AR laut dem “Hype Cycle for Emerging Technologies“ von 2016 eine breite Akzeptanz des Marktes in fünf bis zehn Jahren[7].

Die Lauffähigkeit der häufigsten AR-Applikationen auf nahezu allen Smartphones und Tablets unterstützt die hohe Verbreitung. Zusätzlich fördert nutzbringende Software die Akzeptanz. Einige bereits populäre Anwendungen decken unterschiedlichste Bereiche ab. Zu diesen gehören die Anzeige von ortsbasierten Objekten, Echtzeitübersetzung mit Texterkennung, Fahrzeugkonfiguration, Simulation von Innenraumeinrichtung und Indoor Navigation. Die genaue Lokalisierung im Raum ohne Satellitenunterstützung mit gängiger Smartphone-Ausstattung gestaltet sich kompliziert[91].

Um Tiefeninformationen des Raums zu erhalten und somit eine relative Position im Raum bestimmen zu können, statten bestimmte Hersteller Smartphones mit Stereokameras aus. Android-Geräte, bei denen die spezielle Hardware verbaut wird, sind beispielsweise das Lenovo Phab 2 Pro und das Asus ZenFone AR. Das Projekt “Tango” von Google bietet Hilfsmittel speziell für zertifizierte Android-Geräte mit Stereokameras[92].

Neben Google zeigt außerdem Apple Interesse an der erweiterten Realität. Der iPhone-Hersteller kaufte in den vergangenen Jahren diverse Unternehmen in dem Bereich[93].

Endgeräte in Form von Smartphones und Tablets dienen derzeit als Hauptplattform für AR. Jedoch steigt die Verbreitung von zusätzlichen visuellen Ausgabegeräten wie Head-Up-Displays (HUDs) oder Datenbrillen. Das Investmentbanking- und Wertpapierhandelsunternehmen Goldman Sachs prognostiziert, dass das Marktvolumen im Jahr 2025 für AR und VR 80 Milliarden Dollar betragen wird (Im Jahr 2016 waren es 6,1 Milliarden Dollar)[1, 2].

Augmented Reality spielt neben dem Konsumentenmarkt eine entscheidende Rolle im industriellen Bereich. Unterschiedlichste Unternehmen beginnen, Einsatzgebiete für die erweiterte Realität in Prozessen zu erforschen. Als Beispiel kann “Pick-by-Vision” genannt werden. Hierbei handelt es sich um die datenbrillengestützte Kommissionierung in Lagerhallen (Details siehe [94]).

Zusammenfassend lässt sich schlussfolgern, dass die Zeit der erweiterten Realität bereits begonnen hat und AR für die Zukunft hoch gehandelt wird.

Glossar

Android Quelloffenes Betriebssystem für mobile Geräte. 2

Android Activity Die Android Activity stellt eine Bildschirmseite dar. 40

Android Studio Von Google empfohlene Entwicklungsumgebung zur Erstellung von Android-Applikationen. 40

Community Gemeinschaft von Menschen, die sich mit dem gleichen Bereich beschäftigen. 25

Computer Vision Computergestützte Verarbeitung von visuellen Informationen. 9, 28, 48

Content-API Eine Content-API bezeichnet die Schnittstelle der Software zur Integration von spezifischen Inhalten. 16, 28, 31

Cordova Das Open-Source Mobile Development Framework Apache Cordova unterstützt bei der Entwicklung von Plattformübergreifenden Applikationen mit Web Technologien. Für weitere Informationen siehe [95]. 26, 31, 34

Deep Learning Ein Zweig von Machine Learning, wobei eine mehrere Schichten umfassende Architektur verwendet wird, um Beziehungen zwischen Eingangsdaten und Ergebnisdaten zu erkennen. Für mehr Informationen siehe [96]. 28

Extended Tracking Bezeichnet die erweiterte visuelle Erkennung und Nachverfolgung von Bildern oder Objekten. Dabei muss sich das Target nicht im Fokus der Kamera befinden. Zusätzlich dazu werden Informationen aus dem unmittelbaren Umfeld des Objekts zur Wiedererkennung genutzt. 24, 26, 53, 55, 57

Face Recognition Die visuelle Erkennung von menschlichen Gesichtern. 30

Face Tracking Kombiniert Face Recognition mit der kontinuierlichen Positionsbestimmung eines Gesichts im Raum. 30, 31, 35, 37

Framework Programmiergerüst. Wird als Synonym für SDK verwendet. 19, 21

Geo Fencing Virtuelle, geografische Eingrenzung eines Bereichs. Ermöglicht das Auslösen von Aktionen wenn ein Gerät in ein definiertes Areal eindringt oder dieses verlässt. 26

iBeacon Dient zur Lokalisierung in Räumen basierend auf Bluetooth. Wurde von Apple Inc. entwickelt. 28

Image Matching Erkennung eines Bildes im Raum mit Hilfe eines hinterlegten Bildes. 14, 15, 31, 33–37, 48, 52

Image Recognition Erkennung eines Bildes. 18, 39, 60

Image Tracking Bezeichnet die visuelle Erkennung und Nachverfolgung von Bildern. 14, 15, 22

IMU Sensors Inertial Measurement Unit (IMU) ist eine Kombination von Trägheitssensoren wie Drehraten- und Beschleunigungssensoren[97]. 31

Indoor Navigation Navigation im Raum ohne Satellitenunterstützung. 31, 33, 34, 65

iOS Von Apple Inc. entwickeltes Betriebssystem für Smartphones und Tablets. 2

Machine Learning Machine Learning dient zur Entwicklung und zur Verwendung von Algorithmen, um Vorhersagen aus unklassifizierten Daten treffen zu können[96]. 28

Marker Definierter, markanter Punkt im Raum. Dient zur Positionsbestimmung. 22, 24

Markerlos Positionsbestimmung, die ohne zuvor definierten markanten Punkt möglich ist. 22

Neuronal Networks Deutsch: (künstliche) Neuronale Netze. Ein Mechanismus für die künstliche Intelligenz, der dem neuronalen Netz des Gehirns nachempfunden ist. 28

Object Matching Erkennung eines Objekts im Raum mit Hilfe einer hinterlegten Beschreibung des Objekts. 53, 55

Object Recognition Erkennung eines dreidimensionalen Objekts. 18

- Object Tracking** Die Erkennung und kontinuierliche Positionsbestimmung eines beweglichen Objektes im Raum. 14, 22, 31, 36
- Plugin** Software-Komponente, die zur Laufzeit eingebunden und verwendet werden kann. 16, 22, 33, 34, 37
- Rendering** Generierung einer Grafik aus einem Modell oder Daten. 22
- RESTful** Stellt eine Implementierung des Architekturstils REST dar. 18
- SLAM** Simultaneous Localization and Mapping (SLAM) dient zur Kartografierung der Umgebung sowie der Lokalisierung der erstellten Kartografie[98]. 22, 26, 31, 35
- Stereokamera** Apparatur zur Aufnahme von Bildern mit Tiefeninformationen. 22, 65
- Target** Objekt, das durch Computer Vision erkannt werden soll. 39, 41, 42, 44, 45, 47–49, 51–53, 57, 60
- Textur** Oberflächenschicht eines Modells, die beliebige Farben und Strukturen darstellt. 54
- Ticketing System** Software zur Organisation und Bearbeitung von Kundenanfragen. 22, 29, 35
- Titanium** Appcclerator Titanium ermöglicht es, mobile Applikationen für unter anderem iOS, Android und Windows Phone mit JavaScript zu entwickeln[99]. 26, 31, 35
- Unity** Unity ist eine Plattform zur Spieleentwicklung, optimiert für 3D[100]. 14, 22, 26, 31, 33, 34, 37
- Virtual Buttons** Virtuelle Buttons dienen dazu, Ereignisse beim Berühren virtueller Objekte auszulösen. 14, 15, 31, 35, 37
- Visual Search** Durchsucht einen Videostream nach einem oder mehreren definierten Objekten[101]. 31

Xamarin Die Xamarin Plattform dient als Framework zur Plattformübergreifenden Entwicklung von iOS, Android, Windows und MacOS/OSX Anwendungen in C#[102]. 26, 31, 35

Xcode Entwicklungsumgebung von Apple Inc.. 40

Literaturverzeichnis

- [1] IDC Research, Inc. Worldwide Spending on Augmented and Virtual Reality Forecast to Reach \$13.9 Billion in 2017, According to IDC. Abgerufen am 18.05.2017. [Online]. Verfügbar: <http://www.idc.com/getdoc.jsp?containerId=prUS42331217>
- [2] Goldman Sachs & Co. LLC, „Profiles in Innovation - Virtual & Augmented Reality,” S. 1–30. [Online]. Verfügbar: <http://www.goldmansachs.com/our-thinking/pages/technology-driving-innovation-folder/virtual-and-augmented-reality/report.pdf>
- [3] R. Dörner, W. Broll, P. Grimm, und B. Jung, *Virtual und Augmented Reality*. Springer Vieweg, 2014.
- [4] R. Sood, *Pro Android Augmented Reality*. Apress, 2012.
- [5] D. Schmalstieg und T. Hollerer, *Augmented Reality: Principles and Practice*. Addison Wesley, 2016.
- [6] Google Inc. Android-Apps auf Google Play. Abgerufen am 28.05.2017. [Online]. Verfügbar: <https://play.google.com/store/search?q=Augmented%20Reality&c=apps&hl=de>
- [7] Gartner, Inc. Gartner’s 2016 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage. Abgerufen am 14.01.2017. [Online]. Verfügbar: <http://www.gartner.com/newsroom/id/3412017>
- [8] J. Grubert und D. R. Grasset, *Augmented Reality for Android Application Development*. Packt Publishing, 2013.
- [9] E. Offermann. There are dozens more Augmented Reality SDKs than you think! Here are seven great ones. Abgerufen am 15.01.2017. [Online]. Verfügbar: <https://www.linkedin.com/pulse/dozens-more-augmented-reality-sdks-than-you-think-here-offermann>
- [10] Augmented Reality Lab S.L. ARLab Company. Abgerufen am 16.01.2017. [Online]. Verfügbar: <http://arlab.com/company>
- [11] ——. ARLab Pricing. Abgerufen am 16.01.2017. [Online]. Verfügbar: <http://arlab.com/pricing>
- [12] ——. ARLab FAQ. Abgerufen am 16.01.2017. [Online]. Verfügbar: <http://developers.arlab.com/faq>
- [13] ——. ARLab Products. Abgerufen am 22.01.2017. [Online]. Verfügbar: <http://arlab.com/products>

-
- [14] ——. ARLab Doc. Abgerufen am 23.01.2017. [Online]. Verfügbar: <http://www.arlab.com/doc/>
- [15] ——. ARLab Doc IM. Abgerufen am 23.01.2017. [Online]. Verfügbar: http://www.arlab.com/doc/arlab/product/image_matching/android/getting_started/introduction
- [16] ——. ARLab Doc IM iOS. Abgerufen am 23.01.2017. [Online]. Verfügbar: http://www.arlab.com/doc/arlab/product/image_matching/ios/samples/box_office/intro
- [17] F. Calvo, „Re: Visitor Message,” [Email] Francisco Calvo <francisco.calvo@arlab.com>.
- [18] Augmented Reality Lab S.L. ARLab Doc IT. Abgerufen am 23.01.2017. [Online]. Verfügbar: http://www.arlab.com/doc/arlab/product/image_tracking/start
- [19] ——. ARLab LinkAR Home. Abgerufen am 22.01.2017. [Online]. Verfügbar: <http://developers.arlab.com/me>
- [20] Inglobe Technologies Srl. AR.media Developer Portal. Abgerufen am 20.01.2017. [Online]. Verfügbar: <http://dev.inglobetechnologies.com/features>
- [21] ——. AR.media Hyperspaces Prices. Abgerufen am 17.01.2017. [Online]. Verfügbar: <http://hyperspaces.inglobetechnologies.com/buy/>
- [22] Apple Inc. „hyperspaces“ im App Store. Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://itunes.apple.com/de/app/hyperspaces/id954983584?mt=8>
- [23] Y. Quispe, „Inquiry- CATCHOOM- AR & IR,” [Email] Catchoom Sales <sales@catchoom.com>.
- [24] Inglobe Technologies Srl. AR.media Developer Portal. Abgerufen am 20.01.2017. [Online]. Verfügbar: <http://dev.inglobetechnologies.com/buy/>
- [25] —, „Re: Questions Master-Thesis AR-SDK,” [Email] Software <software@inglobetechnologies.com>.
- [26] Catchoom Technologies S.L. Contact Catchoom - Image recognition & Augmented Reality. Abgerufen am 20.01.2017. [Online]. Verfügbar: <http://info.catchoom.com>
- [27] ——. CraftAR Mobile SDKs - Comparison. Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://catchoom.com/product/craftar/mobile-sdks-comparison/>
- [28] ——. Catchoom - Image Recognition and Augmented Reality. Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://catchoom.com/>

-
- [29] ——. CraftAR Pricing - Augmented Reality and Image Recognition. Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://catchoom.com/product/craftar/pricing/>
- [30] ——. Does CraftAR work with non-planar objects? Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://support.catchoom.com/customer/en/portal/articles/1785949-does-craftar-work-with-non-planar-objects->
- [31] ——. CraftAR: Augmented Reality and Image Recognition toolbox. Abgerufen am 20.01.2017. [Online]. Verfügbar: <https://catchoom.com/product/craftar/augmented-reality-and-image-recognition/>
- [32] HoloBuilder, Inc. HoloBuilder: Go Where the Work Is with Digital Job Walks! Abgerufen am 23.01.2017. [Online]. Verfügbar: <http://landing.holobuilder.com/construction>
- [33] Kudan Limited. Download CV SDK. Abgerufen am 23.01.2017. [Online]. Verfügbar: <https://www.kudan.eu/download-kudan-cv-sdk/>
- [34] ——. SDK Features. Abgerufen am 23.01.2017. [Online]. Verfügbar: <https://www.kudan.eu/kudan-sdk-features/>
- [35] "Luke Kudan", Kudan Limited. Basic Questions. Abgerufen am 24.01.2017. [Online]. Verfügbar: <http://forum.kudan.eu/t/basic-questions/75/2>
- [36] Kudan Limited. Support. Abgerufen am 24.01.2017. [Online]. Verfügbar: <https://www.kudan.eu/sdk-support/>
- [37] ——. Get Support - Service Desk. Abgerufen am 24.01.2017. [Online]. Verfügbar: <https://kudaneu.atlassian.net/servicedesk/customer/portal/1>
- [38] ——. Pricing. Abgerufen am 24.01.2017. [Online]. Verfügbar: <https://www.kudan.eu/sdk-pricing/>
- [39] PTC Inc. About Us. Abgerufen am 25.01.2017. [Online]. Verfügbar: <http://www.ptc.com/about>
- [40] ——. Vuforia. Abgerufen am 25.01.2017. [Online]. Verfügbar: <https://www.vuforia.com/Features>
- [41] ——. Object Recognition FAQ. Abgerufen am 27.01.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/FAQ/Object-Recognition-FAQ>
- [42] ——. VuMark. Abgerufen am 27.01.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Training/VuMark>
- [43] ——. Getting Started with Vuforia for iOS Development. Abgerufen am 22.05.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Solution/Getting-Started-with-Vuforia-for-iOS-Development>

-
- [44] ——. Getting Started with Vuforia for Android Development. Abgerufen am 22.05.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Solution/Getting-Started-with-Vuforia-for-Android-Development>
- [45] ——. Support. Abgerufen am 27.01.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/FAQ/Support>
- [46] ——. Vuforia Pricing. Abgerufen am 27.01.2017. [Online]. Verfügbar: <https://developer.vuforia.com/pricing>
- [47] Wikitude GmbH. Imprint - Wikitude. Abgerufen am 28.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/imprint/>
- [48] ——. Wikitude. Abgerufen am 28.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/>
- [49] ——. Wikitude SDK iOS Native API 2.0.0 Documentation. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/external/doc/documentation/latest/iosnative/supporteddevices.html>
- [50] ——. Wikitude SDK Features - Wikitude. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/products/wikitude-sdk-features/>
- [51] ——. Augmented Reality SDK - Wikitude - Wikitude. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/products/wikitude-sdk/>
- [52] ——. Wikitude SDK Plugins API - Wikitude. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/products/wikitude-sdk-features/wikitude-sdk-plugins-api/>
- [53] ——. Unity documentation - Devzone. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/developer/documentation/unity>
- [54] ——. Products - Wikitude. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://www.wikitude.com/store/>
- [55] LM3LABS, Inc. Expertise. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://corp.lm3labs.com/technologies-2/services/>
- [56] ——. About. Abgerufen am 29.01.2017. [Online]. Verfügbar: <https://www.xludia.com/about>
- [57] ——. Privacy & Terms. Abgerufen am 29.01.2017. [Online]. Verfügbar: <https://www.xludia.com/privacy-terms>
- [58] ——. Jobs. Abgerufen am 29.01.2017. [Online]. Verfügbar: <http://corp.lm3labs.com/company-2/careers/>

-
- [59] ——. Home. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/>
- [60] ——. Blog. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/blog/>
- [61] ——. Technology. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/tech>
- [62] ——. Xloudia TV. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/xloudia-tv>
- [63] ——. „YOur Questions regarding Xloudia,” [Email] info info
<info@lm3labs.com>.
- [64] ——. Pricing. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/pricing/>
- [65] ——. Start Today. Abgerufen am 29.01.2017. [Online]. Verfügbar:
<https://www.xloudia.com/start-today>
- [66] XZIMG Limited. Home. Abgerufen am 04.02.2017. [Online]. Verfügbar:
<http://www.xzimg.com/Home>
- [67] ——. Products. Abgerufen am 04.02.2017. [Online]. Verfügbar:
<http://www.xzimg.com/Products?nav=product-XAF>
- [68] ——. Products. Abgerufen am 04.02.2017. [Online]. Verfügbar:
<http://www.xzimg.com/Products?nav=product-XMF>
- [69] ——. Products. Abgerufen am 04.02.2017. [Online]. Verfügbar:
<http://www.xzimg.com/Products?nav=product-XAV>
- [70] Yvonne Göpfert. Der Blick fürs Kleingedruckte: Kamera mit Makro des iPhone 3G S. Abgerufen am 20.07.2017. [Online]. Verfügbar:
<https://www.pcwelt.de/produkte/Der-Blick-fuers-Kleingedruckte-Kamera-mit-Makro-des-iPhone-3G-S-Kult-Handy-im-Test-1257325.html>
- [71] Apple Inc. iOS 6.1.3 Software-Update. Abgerufen am 20.07.2017. [Online]. Verfügbar:
https://support.apple.com/kb/DL1646?locale=de_DE&viewlocale=de_DE
- [72] XZIMG Limited. About. Abgerufen am 09.02.2017. [Online]. Verfügbar:
<http://www.xzimg.com/About>
- [73] Kudan Limited. Getting Started Android. Abgerufen am 11.02.2017. [Online]. Verfügbar: https://wiki.kudan.eu/Getting_Started_Android
- [74] ——. Getting Started iOS. Abgerufen am 13.03.2017. [Online]. Verfügbar:
https://wiki.kudan.eu/Getting_Started_iOS

-
- [75] ——. Development Licence Keys. Abgerufen am 11.02.2017. [Online]. Verfügbar: https://wiki.kudan.eu/Development_License_Keys
- [76] ——. Marker Basics Android. Abgerufen am 21.02.2017. [Online]. Verfügbar: https://wiki.kudan.eu/Marker_Basics_Android
- [77] ——. Marker Basics iOS. Abgerufen am 10.03.2017. [Online]. Verfügbar: https://wiki.kudan.eu/Marker_Basics_iOS
- [78] PTC Inc. Installing the Android SDK for Android Studio. Abgerufen am 12.02.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Solution/Installing-the-Android-SDK-for-Android-Studio>
- [79] ——. How to Compile and Run an Android Sample in Android Studio. Abgerufen am 12.02.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Solution/How-to-Compile-and-Run-an-Android-Sample-in-Android-Studio>
- [80] stackoverflow.com. Vuforia: How to change teapot with a text in the image targets example? - Stack Overflow. Abgerufen am 21.02.2017. [Online]. Verfügbar: <http://stackoverflow.com/questions/17939607/vuforia-how-to-change-teapot-with-a-text-in-the-image-targets-example/32632414>
- [81] PTC Inc. Transparent textures. Abgerufen am 21.02.2017. [Online]. Verfügbar: <https://developer.vuforia.com/forum/ios/transparent-textures>
- [82] ——. Vuforia Target Manager. Abgerufen am 21.02.2017. [Online]. Verfügbar: <https://library.vuforia.com/articles/Training/Getting-Started-with-the-Vuforia-Target-Manager>
- [83] Wikitude GmbH. Android - Devzone. Abgerufen am 13.02.2017. [Online]. Verfügbar: <http://www.wikitude.com/developer/documentation/android>
- [84] ——. Wikitude SDK Android 6.1.0 Dokumentation. Abgerufen am 29.04.2017. [Online]. Verfügbar: <http://www.wikitude.com/external/doc/documentation/latest/android/3dmodels.html#3d-models>
- [85] ——. Wikitude SDK Android 6.1.0 Dokumentation. Abgerufen am 29.04.2017. [Online]. Verfügbar: <http://www.wikitude.com/external/doc/documentation/latest/android/advancedimagerecognition.html#advanced-image-recognition>
- [86] ——. Wikitude SDK Android 6.1.0 Dokumentation. Abgerufen am 03.05.2017. [Online]. Verfügbar: <http://www.wikitude.com/external/doc/documentation/latest/android/ade.html>
- [87] Google Inc. Getting Started with Remote Debugging Android Devices | Web | Google Developers. Abgerufen am 03.05.2017. [Online]. Verfügbar: https://developers.google.com/web/tools/chrome-devtools/remote-debugging/?utm_source=dcc&utm_medium=redirect&utm_campaign=2016q3

-
- [88] Wikitude GmbH. Wikitude SDK Android 6.1.0 Dokumentation. Abgerufen am 03.05.2017. [Online]. Verfügbar: <http://www.wikitude.com/external/doc/documentation/latest/android/assetsworkflow.html#3d-assets-workflow>
- [89] ——. change texture of 3D model : Wikitude. Abgerufen am 03.05.2017. [Online]. Verfügbar: <https://support.wikitude.com/support/discussions/topics/5000077437>
- [90] ——. Javascript iOS 3D Object manipulation jerky : Wikitude. Abgerufen am 03.05.2017. [Online]. Verfügbar: <https://support.wikitude.com/support/discussions/topics/5000082546?page=1>
- [91] M. Werner, *Ubiquitous Navigation: Skalierbare ortsbezogene Dienste in Gebäuden*. Logos Berlin, 2012.
- [92] Google Inc. Tango. Abgerufen am 18.05.2017. [Online]. Verfügbar: https://get.google.com/intl/de_ALL/tango/
- [93] Mac & i. "High on AR": Apple-Chef betont Bedeutung von Augmented Reality. Abgerufen am 18.05.2017. [Online]. Verfügbar: <http://www.heise.de/mac-and-i/meldung/High-on-AR-Apple-Chef-betont-Bedeutung-von-Augmented-Reality-3351497.html>
- [94] Technische Universität München. Pick-by-Vision. Abgerufen am 18.05.2017. [Online]. Verfügbar: http://www.fml.mw.tum.de/fml/index.php?Set_ID=258
- [95] The Apache Software Foundation. Architectural overview of Cordova plattform. Abgerufen am 07.02.2017. [Online]. Verfügbar: <https://cordova.apache.org/docs/en/6.x/guide/overview/index.html>
- [96] D. J. F. Wiley, *R Deep Learning Essentials*. Packt Publishing Ltd, 2016.
- [97] O. J. Woodman, „An introduction to inertial navigation,” University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-696, Aug. 2007. [Online]. Verfügbar: <http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf>
- [98] S. Thrun und J. J. Leonard, „Simultaneous Localization and Mapping,” in *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008, S. 871–889.
- [99] Appcelerator Inc. Mobile App Development & MBaaS Products. Abgerufen am 07.02.2017. [Online]. Verfügbar: <http://www.appcelerator.com/mobile-app-development-products/>
- [100] Unity Technologies. Unity - Game Engine. Abgerufen am 07.02.2017. [Online]. Verfügbar: <https://unity3d.com>
- [101] J. Duncan und G. W. Humphreys, „Visual search and stimulus similarity,” Vol. 96, Nr. 3, S. 433–458.

- [102] Xamarin Inc. Mobile Application Development to Build Apps in C#. Abgerufen am 07.02.2017. [Online]. Verfügbar: <https://www.xamarin.com/platform>