

# Framework zur Evaluation von Trackingbibliotheken mittels gerenderter Videos von Tracking-Targets

Jonas Diekmann, Patrick Renner, Thies Pfeiffer

Cluster of Excellence Cognitive Interaction Technology, Universität Bielefeld

Inspiration 1, 33619 Bielefeld, Germany

E-Mail: {jdiekmann,prenner,tpfeiffer}@techfak.uni-bielefeld.de

**Zusammenfassung:** Die Erkennung und Verfolgung von Objekten ist seit Jahrzehnten eine wichtige Basis für Anwendungen im Bereich der Erweiterten Realität. Muss aus der Vielzahl an Bibliotheken eine verfügbare ausgewählt werden, fehlt es häufig an Vergleichbarkeit, da es kein standardisiertes Testverfahren gibt. Gleichzeitig ist es für die Entwicklung eigener Verfahren essentiell, das Optimierungspotential genau bestimmen zu können. Die vorliegende Arbeit versucht diese Lücke zu füllen: Mithilfe von systematisch erstellten gerenderten Videos können verschiedene Verfahren und Bibliotheken bezüglich ihrer Genauigkeit und Performanz überprüft werden. Exemplarisch werden die Eigenschaften dreier Trackingbibliotheken miteinander verglichen.

**Stichworte:** Objektverfolgung, Benchmarking, BART, ALVAR, ARUCO

## 1 Einleitung

Als Entwickler von Anwendungen im Bereich der Erweiterten Realität ist man mit der Schwierigkeit konfrontiert, zwischen den mittlerweile vielzähligen Angeboten an Bibliotheken zur Erkennung und Verfolgung der Umgebung (Trackingbibliotheken) auszuwählen. Die Angebote unterscheiden sich unter anderem in technischen Details und der Leistungsfähigkeit bezüglich der Art der unterstützten erkenn- und verfolgbaren Objekte. Ein objektives Vergleichen dieser technischen Eigenschaften ist schwierig, da es an standardisierten Testverfahren mangelt und verschiedene Bibliotheken z.B. im Bereich der speziell erzeugten Tracking-Marker zueinander nicht kompatibel sind. Dadurch wird es sehr aufwändig, vergleichbare Ground-Truth Daten zur Messung der Genauigkeit oder der Effizienz der Bibliotheken zu erheben. Eine unfundierte Entscheidung kann dabei zum Risiko werden: So kann beispielsweise eine Bibliothek, die auf hohe Genauigkeit spezialisiert ist, zu einer höheren Latenz der Erkennung führen und damit ein Echtzeiterlebnis, insbesondere im HMD, schnell zerstören.

Diese Arbeit nimmt sich der Problematik an und bietet eine Lösung in Form eines Test-Frameworks zur Evaluation von Trackingbibliotheken. Um eine Vergleichbarkeit herzustellen, müssen dabei Testdatensätze erstellt werden, die eine automatische Evaluation unterschiedlicher Bibliotheken ermöglichen. Zentraler Ansatz ist es, die Testdatensätze über Rendering-Prozesse virtuell zu generieren, sodass Größe, Position und Orientierung der Zielobjekte in den Videos der Testdatensätze eindeutig bestimmt sind. Damit liegen die Ground-Truth

Daten zur Berechnung der Genauigkeit in höchster Präzision vor.

Im hier präsentierten ersten Schritt konzentrieren wir uns dabei auf Erkennung und Tracking von künstlich erstellten AR-Markern. Testdatensätze für andere Typen von Zielobjekten, z.B. Bildern für Poster-Tracker, sind jedoch auf gleiche Weise generierbar. AR-Marker wurden gewählt, da diese für viele Messeinsätze immer noch die robusteste Lösung darstellen. Bei der Erstellung der Testdatensätze können dabei verschiedene Typen von Markern einfach ausgetauscht werden. Das erarbeitete Vergleichsverfahren wird an drei Bibliotheken exemplarisch angewandt: ALVAR<sup>1</sup> [KAS10], Aruco<sup>2</sup> [GJMCMCMJ14] und die Eigenentwicklung BART<sup>3</sup>. Eine Unterstützung für weitere, insbesondere für kommerzielle Bibliotheken soll schrittweise hinzugefügt werden. Dazu wird das Test-Framework als Teilprojekt des OpenSource-Projektes BART veröffentlicht werden.

Im Folgenden wird ein Überblick über verwandte Arbeiten und aktuelle Markertracking-Bibliotheken gegeben. Im Anschluss werden relevante Untersuchungsaspekte betrachtet und der in dieser Arbeit entwickelte Ansatz wird erklärt. In Kapitel 5 werden die einzelnen Tests vorgestellt und ausgewählte Ergebnisse werden präsentiert, nachfolgend sind detailliertere Ergebnisse zu finden. Abschließend werden die Erkenntnisse der Arbeit diskutiert.

## 2 Relevante Arbeiten

Es gibt zahlreiche Arbeiten in denen Erkennungsraten oder -genauigkeiten von einzelnen Trackingbibliotheken unter realen Bedingungen evaluiert werden. In der Regel sollen dabei die Aussagen oder Verbesserungen der jeweiligen Arbeit belegt werden (z.B. [VTS14, KB99]).

In vielen dieser Arbeiten wird beschrieben, dass ein gewisser Aufwand betrieben wird, um die zu untersuchenden Faktoren stabil, kontrolliert und reproduzierbar zu halten. In der erwähnten Arbeit von Verband, Theys und Schuetter [VTS14] werden unter anderem Kontrastveränderungen untersucht, die in der Realität durch ungünstige Lichtverhältnisse erzeugt werden. Die Autoren stellen dabei diese Kontrastunterschiede durch von Anfang an heller oder dunkler bedruckte Marker her.

Die Arbeit von Kato und Billingham [KB99] dreht sich um die Kalibrierung von Kameras an am Kopf getragenen AR-Systemen und die anschließende Detektion von AR-Markern. Im Rahmen dieser Untersuchungen wird auch die Erkennungsgenauigkeit untersucht, was durch die Beobachtung realer Marker in unterschiedlichen Entfernungen und Neigungen geschieht.

Insgesamt liegt es nahe, die benötigten Markeraufnahmen nicht in der Realität durchzuführen, sondern mithilfe von Techniken aus der virtuellen Realität zu simulieren. Diese Methode wurde unseres Wissens erstmalig von Pentenrieder, Meier und Klinker [PMK06] beschrieben. In dieser Arbeit wird die Genauigkeit eines Markertrackingsystems in Abhängigkeit des Kamerawinkels und der Pixelgröße der Marker auf der Bildebene beschrieben, was prinzipiell gleichbedeutend ist mit der Untersuchung des Neigungswinkels und der Entfer-

---

<sup>1</sup><http://virtual.vtt.fi/virtual/proj2/multimedia/alvar/>

<sup>2</sup><http://www.uco.es/investiga/grupos/ava/node/26>

<sup>3</sup><http://bart.multimodal-interaction.org>

nungen [KB99]. Erwähnenswert ist die Feststellung der Autoren, dass sich die fokale Länge, die Auflösung und die Entfernung des Markers zu einem Parameter zusammenfassen lassen, der die Markergröße in Bild-Pixeln angibt. Außerdem geben sie weitere Parameter an, die ihnen untersuchenswert erscheinen: Bildstörungen, Beleuchtung und Verzerrungen. Das Ergebnis ihrer Arbeit zeigt, dass die Untersuchung in kleinen Intervallen erfolgen muss, da eine "W-form" für die Abhängigkeit des Kamerawinkels gefunden wurde. Außerdem steigt bei ihnen die Abweichung der Genauigkeit mit zunehmendem Kamerabstand entlang der Projektionsrichtung.

## 2.1 Überblick über bekannte Trackingbibliotheken

Dieser Abschnitt gibt einen kurzen Überblick über die wichtigsten am Markt befindlichen Bibliotheken und erläutert die hier betrachtete Auswahl.

Viele der auf dem Markt befindliche Trackingbibliotheken sind eher dem Bereich der Forschung zuzuordnen. Darunter der Pionier *ARToolkit* [KB99], *ALVAR* (VTT Technical Resarch Centre of Finland) [KAS10], *Studierstube* (Technische Universität Graz), und *ARUCO* (Universidad de Córdoba) [GJMSMCMJ14]. Andere wie *Qualcomm Vuforia* und *Metaio SDK* werden kommerziell angeboten.

Diese Arbeit betrachtet neben der Eigenentwicklung BART (*Bielefeld Augmented Reality Tracker*), die Bibliotheken ARUCO (auch *ArUco* für *Augmented Reality* und die *Universidad de Cordoba*) sowie ALVAR (*A Library for Virtual and Augmented Reality*).

ALVAR wird wegen des langjährigen Entwicklungs- und Forschungszeitraums (seit 2009) und seiner bewiesenen Marktreife für kommerzielle Produkte gewählt. ARUCO dagegen verwendet ein ähnliches Prinzip wie BART und wird aktiv weiterentwickelt. Ein weiteres Kriterium für die Wahl dieser drei Bibliotheken ist, dass alle auf OpenCV aufbauen, was eine hohe Vergleichbarkeit im Bereich der Messung des Ressourcenverbrauches verspricht.

## 3 Untersuchungsaspekte

Im Folgenden werden einige Aspekte im Zusammenhang mit Trackingsoftware vorgestellt, für deren Untersuchung wir die Erstellung von Videos virtueller AR-Marker für geeignet halten.

Wie in [PMK06] gezeigt, lassen sich Translationen und Rotationen untersuchen. Darüber hinaus gibt es mit Kontrast bzw. Farbabhängigkeiten einen weiteren Bereich, der schon für diesen Ansatz vorgeschlagen wurde (vgl. Kap. 2). Hinzu kommen Verdeckungen, sowohl längerfristige Teilverdeckungen (z.B. durch Finger) als auch kurzzeitige Verdeckungen ganzer Marker. Außerdem halten wir Rauschen, unruhige Umgebungen, falsche markerähnliche Artefakte, sowie echte Hintergrundbilder und Videos für realisierbar. Zudem soll in sogenannten Massentests das Verhalten der Bibliotheken bei großen Markerzahlen untersucht werden.

Jeder dieser Aspekte beeinflusst zahlreiche Messgrößen. Hier stellt sich zunächst die Frage, ob ein bestimmter Marker überhaupt detektiert wird. Dazu gehört die allgemeine Er-

kennungsrate bezüglich eines Videos, die Zahl und das Profil der Fehlerkennungen, sowie das Profil der erkannten Marker. Anschließend können Betrachtungen zur Genauigkeit der Bestimmung von Position und Orientierung der jeweiligen Marker angestellt werden. Darüber hinaus ist für einen Vergleich von Bibliotheken der Ressourcenverbrauch von entscheidender Bedeutung. Hierbei sind vor allem die benötigte Rechenzeit und der Verbrauch an Arbeitsspeicher relevant.

Das Vorhaben jede dieser Messgrößen bei allen genannten Aspekten zu untersuchen, würde den Rahmen dieses Papers sprengen. Wir werden uns daher auf einige Aspekte beschränken, bei denen jeweils nur eine Auswahl der genannten Messgrößen betrachtet wird

## 4 Ansatz dieser Arbeit

Für die verschiedenen zu testenden Aspekte wurden mittels Blender – einer Software für die Erstellung von 3D Animationen – jeweils eigene Testvideos erstellt und zu jedem eine CSV-Datei mit den Markerpositionen und -orientierungen für jedes Einzelbild exportiert. Abbildung 1 zeigt eine Auswahl der verschiedenen Tests.

Die Testsuite mit allen Videos und CSV-Dateien wird mit der Veröffentlichung dieses Beitrags auf unserer Webseite zur Verfügung gestellt. Aus Platzgründen können hier nur ausgewählte Aspekte vorgestellt werden. Das Material auf der Webseite umfasst den vollen Umfang der Testsuite als auch die Gesamtergebnisse der evaluierten Bibliotheken.

### 4.1 Erstellung der Testdaten

Es wird Blender verwendet, um damit virtuelle Marker zu platzieren und daraus Videos mit den zu testenden Eigenschaften zu erzeugen (Abb. 1). Dabei kommen der Standardrenderer und das Standardkammermodell von Blender zum Einsatz, in dem eine fokale Länge und eine Sensorweite festgelegt werden. Neben dem Export des Videos wird gleichzeitig eine CSV-Datei mit den Ground-Truth Daten der Positionen und Orientierungen der Marker pro Frame des Videos erzeugt.

### 4.2 Anbindung der Trackingbibliotheken

Den Trackingbibliotheken wird jeweils das gesamte Video zur Verfügung gestellt. Folglich wird die Verteilung der Rechenzeit auf die Einzelbilder von der Bibliothek selbst gesteuert. Dadurch ist es auch möglich, dass Bibliotheken für die Analyse eines Videos mehr Zeit verwenden, als deren Laufzeit beträgt. Dieser Ansatz erlaubt eine korrekte Interpretation der Erkennungsraten, da von der Bibliothek alle Bilder analysiert werden konnten. Zugleich muss bei der Wahl für eine höhere zeitliche Auflösung abgeschätzt werden, ob noch eine Echtzeiterkennung möglich ist. Dafür wird die Rechenzeit der Bibliotheken ermittelt (vgl. Kap. 6.2) und verglichen.

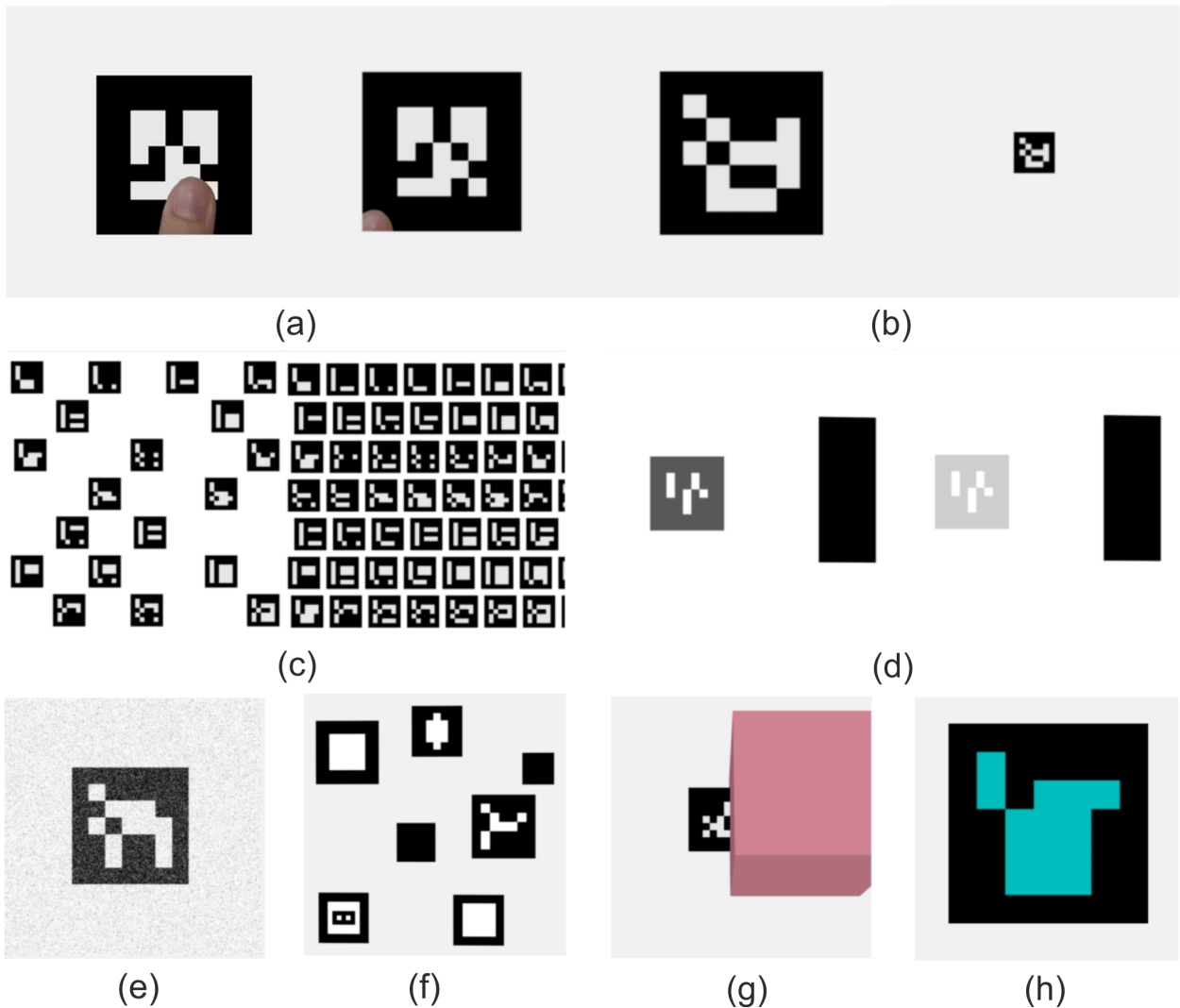


Abbildung 1: Beispiele aus den verschiedenen Test-Videos: (a) Teilweise Verdeckung, (b) Bewegung, (c) Massenmarker, (d) Kontraste, (e) Rauschen, (f) Falsche Marker, (g) Größere Verdeckungen, (h) Farbverfälschungen

### 4.3 Ansatz für die Ermittlung der Ressourcenverbrauchswerte

Die CPU-Zeit wird extern aus den Prozessdeskriptoren ausgelesen. Sie bezieht sich auf das gesamte Programm und beinhaltet neben der Zeit für die Auswertung der Videos einen Overhead für das Laden der Videos und das Schreiben der CSV-Dateien, der, da alle Bibliotheken OpenCV nutzen, vergleichbar ist (zumindest bei vergleichbaren Erkennungsraten; siehe Tabelle 3). Der Arbeitsspeicherverbrauch wird ebenfalls extern ermittelt und in Abständen von etwa 300 ms abgefragt. Dadurch können Extremwerte verpasst werden. Insbesondere bei Videos mit stark schwankendem Arbeitsspeicherverbrauch (z.B. die Massentests) sind die Werte daher als Näherung anzusehen. Das Testsystem verfügt über eine 3.4 GHz CPU und 7,8 GB Arbeitsspeicher, von denen 6 GB plus Auslagerungsspeicher (5 GB) für die Bibliotheken nutzbar sind. Die Rechenzeit für das Ein- und Auslagern ist nicht in der ermittelten CPU-Zeit enthalten.

## 4.4 Ansatz für die Analyse

Die von den Bibliotheken zurückgegebenen erkannten Marker, ihre Transformation und die gemessenen Ressourcenverbrauchswerte werden anschließend aufbereitet und verglichen.

Hierbei lassen sich die erkannten Marker und die ermittelten Positionen direkt mit den von Blender gelieferten Ground-Truth Daten verrechnen. Dadurch werden Informationen über die Genauigkeit der Positionsbestimmung, die Erkennungsrate, die Zahl der Fehlerkennungen und das Profil der erkannten, nicht erkannten und falsch erkannten Marker gewonnen (vgl. Kap. 3). Die von den Bibliotheken zurückgegebenen Rotationsmatritzen werden für die Vergleiche in Eulerwinkel umgewandelt.

Zu beachten ist bei der Interpretation der Erkennungsraten, dass viele der Videos zu den Testen das Bild kontinuierlich verschlechtern oder verbessern. Kleine Abweichungen zwischen den verschiedenen Bildwiederholungsraten sind daher der Messmethode geschuldet, da die Genauigkeit, mit der eine Erkennungsgrenze ermittelt wird proportional zur Bildwiederholungsrate ist.

## 4.5 Koordinatensystem und Einheiten

Es wird ein linkshändiges Koordinatensystem mit linkshändigen Drehrichtungen benutzt; die Achsen wurden in Metern skaliert. Sofern nichts anderes angegeben, wurden die Marker mit  $1\text{ m} \times 1\text{ m}$  Seitenlänge simuliert. Wie in [PMK06] vorgeschlagen, können die Ergebnisse auf kleinere Markerlängen mit geringerem Abstand umgerechnet werden.

# 5 Tests

Im Folgenden werden einzelne Tests und ihre Ergebnisse genauer beschrieben.

## 5.1 Tests zu Bewegungen

In diesem Test werden Marker kontinuierlich in verschiedene Raumrichtungen bewegt. Alle anderen Faktoren werden möglichst optimal gehalten (kein Rauschen, einfarbiger weißer Hintergrund). In drei Videos werden Bewegungen in den jeweiligen Raumachsen getestet. Mit zunehmender Entfernung werden die Marker sehr klein: Die maximal gesetzte Entfernung entspricht der 3.000-fachen Markerbreite, was in der höchsten Auflösung gerade noch 4 sichtbaren Pixeln entspricht. Insgesamt zielt dieser Test primär auf die Trackingfähigkeit ab.

### 5.1.1 Ergebnisse

Bei der Translation entlang der Z-Achse ist auffällig, dass die Erkennung von BART bei Markergrößen kleiner als  $76 \times 76$  Pixeln Lücken aufweist, hier scheint das Tracking nicht gut genug zu greifen. ALVAR ist besser und erkennt den Marker bei  $1080 \times 48$  bis zu einer Markergröße von  $6 \times 6$  Pixeln. Dabei hört eine auftretende Fehlerkennung bereits ab  $140 \times 140$  Pixeln

Auflösung	1080p48	1080p24	1080p12	Sonstige
Fehlerkennungen	198/201	96/100	45/50	0

Tabelle 1: Fehlerkennungen durch ALVAR bei verschiedenen Auflösungen

auf, was darauf schließen lässt, dass ALVAR etwa ab dieser Größe primär auf Tracking setzt. ARUCO erkennt nur bis zu einer Markergröße von etwa 76x76 Pixeln. Diese Erkennung ist lückenlos, allerdings haben in diesem Bereich des Videos auch die anderen Bibliotheken eine lückenlose Erkennung.

Bei kleineren Bildauflösungen verschlechtern sich die Erkennungsraten nur leicht. Die unterschiedlichen Auflösungen weisen jedoch einen unterschiedlichen Ressourcenverbrauch auf.

Als Besonderheit weist ALVAR bei der Bewegung in X-Richtung in der höchsten Auflösung bei allen Bildwiederholraten viele Fehlerkennungen auf (vgl. Tabelle 1).

## 5.2 Tests zu Rotationen

Mit fünf Videos wird die Erkennung von Rotationen getestet (Rotationen um Y-Achse, Z-Achse und YZ-Achse). In drei Videos werden dabei kontinuierliche Markerrotationen verwendet. In zwei Videos werden die Marker 12 mal pro Sekunde getauscht, um gezielt den Anteil des Trackings zu testen.

### 5.2.1 Ergebnisse

Bei der Drehung um die Z-Achse fällt auf, dass alle Bibliotheken keine nennenswerten Abweichungen bei der Bestimmung der Translation entlang der Y-Achse aufweisen. Die Abweichung entlang der Z-Achse liegt für BART bei 0.04 m und für Aruco und Alvar deutlich darüber, jedoch weiterhin auf niedrigem Niveau. Auch die Streuung ist bei dieser Rotationsform insgesamt eher gering.

Bei den anderen Rotationsachsen sind die Marker unterschiedlich gut sichtbar und die verschiedenen Bibliotheken unterscheiden sich im Bereich in dem sie noch den Marker erkennen. Die Grenze liegt bei allen Bibliotheken etwas über 80° (siehe Tab. 2). Für ALVAR ist hier der Winkelbereich angegeben, in dem überhaupt ein Marker erkannt wurde, wenn auch mit hohen Fehlererkennungen, wie bereits oben beschrieben.

## 5.3 Tests zum Kontrast

In diesen Tests wurde der Kontrast der Marker verändert. In den ersten vier Videos wird der Marker dazu der Reihe nach aufgehellt, der helle Teil abgedunkelt und beides gemischt. Die Videos fünf bis acht wiederholen dieses Verhalten mit zwei größeren schwarzen und weißen Artefakten im Hintergrund, um die Qualität des Binarisierungsverfahrens zu testen. Die

Bibliothek	BART		ALVAR		ARUCO	
	min.	max.	min.	max.	min.	max.
X-Achse	-83,2°	81,8°	-84,7°	84,9°	-82,9°	82,9°
Y-Achse	-82,9°	82,9°	-84,9°	84,7°	-82,4°	82,9°

Tabelle 2: Maximal erkannte Winkel

Videos neun bis zwölf wiederholen wiederum die Videos fünf bis acht, wobei die Identifikationsnummer des Markers 12 mal pro Sekunde ausgetauscht wird um eine Neuerkennung zu erzwingen.

### 5.3.1 Ergebnisse

Auch in diesen Videos gibt es Fehlerkennungen bei ALVAR. So wird in Kontrast 1 und Kontrast 2 in der Bildauflösung 1080p ein Marker mit seiner richtigen und einer weiteren falschen Identifikationsnummer erkannt.

Ansonsten fällt auf, dass ALVAR in den Videos mit möglichem Tracking etwas bessere Erkennungsraten aufweist, als die anderen beiden Bibliotheken. In den Videos ohne Trackingmöglichkeit sind BART und ARUCO besser, wobei die Erkennungsraten von BART die von ARUCO leicht übersteigen. Dabei zeigt sich grundsätzlich, dass die Möglichkeit einen Marker bei schlechter werdenden Lichtverhältnissen zu tracken, die Erkennungsraten verbessert. BART und ARUCO zeigen zudem, dass die Möglichkeit vier aufeinanderfolgende Bilder eines Markers bei gleichen Lichtverhältnissen auszuwerten bessere Erkennungsraten liefert, als zwei oder ein aufeinanderfolgendes Bild (vgl. Kap. 6.1)

## 5.4 Tests zur Anzahl der Marker

In diesen Videos werden nach und nach immer mehr unterschiedliche Marker eingeblendet. Im ersten Video ist dabei die Entfernung der Marker unterschiedlich, um eine Adaption durch die Software zu vermeiden. Dieser Test bietet allerdings wegen der daraus resultierenden Platzprobleme nur 41 Marker an. Die nächsten drei Videos platzieren jeweils bis zu 448 Marker in einem fixen Abstand von fast 30 m oder fast 40 m vor der Kamera. Dabei werden die Marker zunächst so platziert, dass ein größerer Abstand zu den Seiten anderer Marker eingehalten wird, bis bei 47 Markern ein diagonales Muster erreicht ist. Dann liegt bei jedem Marker maximal eine Seite in unmittelbarer Nachbarschaft zu einer anderen Seite und bis 89 Marker ist dies mit drei Seiten realisiert.

Dadurch soll verhindert werden, dass der Test bei Problemen mit der Erkennung benachbarter Marker gar keine Ergebnisse liefert. Andererseits sollten Bibliotheken, die bei einem der genannten Bilder aussteigen zusätzlich auf Probleme mit zu dicht nebeneinander liegenden Markern untersucht werden um diese Ursache ausschließen zu können. Die beiden vorhergehenden Videos blenden die Marker immer an der gleichen Stelle ein, was somit ein



Tracking von Markern ermöglicht. Um aussagekräftige Ergebnisse über das Verhalten der Bibliotheken bei der Neuerkennung von Markern zu erreichen werden im dritte Video die jeweils eingeblendeten Marker ausgelost und im vierten Video die IDs der Marker vertauscht. Dieser auf Pseudozufallszahlen basierende Wechsel wird 12 mal pro Sekunde durchgeführt, was der langsamsten gerenderten zeitlichen Auflösung entspricht.

#### **5.4.1 Ergebnisse**

Die Tests offenbaren eine Abhängigkeit beim Arbeitsspeicherverbrauch. Während ALVAR für das erste Video in der Auflösung 1080p48 mit nahezu konstanten 1.000 MB auskommt, steigt der Arbeitsspeicherverbrauch bei BART von anfangs etwa 1.177 MB auf über 6.000 MB an. Auch bei ARUCO steigt der Arbeitsspeicherverbrauch im Laufe der Analyse des Videos leicht an, er bleibt mit 914 bis 1.108 MB jedoch deutlich unter dem von BART.

In der in dieser Arbeit standardmäßig verwendeten Konfiguration liefert BART sehr schlechte Erkennungsraten. So wurden in dem zweiten Video nur etwa 5% der Marker detektiert. Dies liegt an einem eingestellten Timeout, der die Analyse eines Bildes nach einer vorgegebenen Zeit abbricht. Trotz dieses Timeouts verbraucht BART für die Analyse der Videos aus dieser Kategorie die meiste Rechenzeit. Ein wenig Linderung schafft hier die Wahl einer kleineren Auflösung, da so mehr Marker in der gleichen Rechenzeit detektiert werden können. Im dritten Video erkennt ARUCO keinen einzigen Marker, da alle Marker zu weit von der Kamera entfernt sind (vgl. Kap. 5.1).

Die Rechenzeiten für diese Tests lagen auf dem Testsystem alle über der Laufzeit des Videos. Der Massentest 3 hat z.B. in der Auflösung 1080p48 eine Laufzeit von etwa 37 Sekunden. Die reinen Rechenzeiten liegen jedoch bei 46 Sekunden (ARUCO) 110,3 Sekunden (ALVAR) und 183,9 Sekunden (BART). Eine Echtzeiterkennung dieser hohen Markerzahlen erfordert somit deutlich schnellere CPUs (vgl. Kap. 6.2).

### **5.5 Tests zu Verdeckungen**

Das hier verwendete Video zeigt vier verschieden starke Verdeckungen durch „Finger“, die jeweils über ein Viertel des Videos beibehalten werden. Dabei kommt die erste Verdeckung von unten mittig und die zweite Verdeckung deckt die linke untere Ecke ab. Die dritte Verdeckung kommt ebenfalls von links unten, lässt die Ecke jedoch unangetastet und die vierte Verdeckung kommt von links und von rechts gleichzeitig (Ecken bleiben frei).

#### **5.5.1 Ergebnisse**

In BART zeigt sich, dass die ersten beiden Verdeckungen in allen Auflösungen erkannt werden, BART jedoch an der Erkennung der anderen Verdeckungen scheitert. Dies zeigt, dass die Verdeckung großer Bereiche für die Erkennung problematischer ist, als die Verdeckung der Ecken. Auch ALVAR zeigt prinzipiell das gleiche Verhalten. Bei der dritten Verdeckung

kommt es allerdings zu einer Fehlerkennung einer anderen Markeridentifikationsnummer. Der Marker wird dabei an der korrekten Stelle angenommen. ARUCO scheint gegen Verdeckungen sensibel zu sein und hat bei diesen Verdeckungen den Marker nicht erkannt.

## 6 Gesamttabellen

### 6.1 Erkennungsraten

Die Tabellen 3, 4 und 5 zeigen die ermittelten Erkennungsraten für einige ausgewählte Videos. Ein Strich bedeutet dabei, dass leider keine Daten vorliegen.

Bibliothek	BART			ALVAR			ARUCO		
	48	24	12	48	24	12	48	24	12
Bewegung 2	99,0 %	93,0 %	78,0 %	99,5 %	100 %	100 %	99,5 %	100 %	100 %
Bewegung 3	37,5 %	37,6 %	38,0 %	48,0 %	48,0 %	48,0 %	24,3 %	24,2 %	24,0 %
Rotationen 2	76,4 %	77,1 %	78,4 %	79,8 %	80,0 %	81,4 %	75,9 %	76,6 %	77,5 %
Rotationen 4	73,3 %	73,0 %	73,0 %	10,7 %	9,0 %	9,0 %	72,6 %	73,0 %	73,0 %
Kontrast 2	99,8 %	100 %	100 %	99,8 %	100 %	100 %	99,8 %	100 %	100 %
Kontrast 7	91,6 %	92,1 %	93,1 %	99,8 %	100 %	100 %	88,7 %	88,7 %	90,1 %
Kontrast 9	91,4 %	24,9 %	29,5 %	27,3 %	26,7 %	31,2 %	90,2 %	24,9 %	28,6 %
Kontrast 10	32,2 %	8,9 %	11,6 %	8,0 %	8,4 %	11,6 %	24,4 %	7,1 %	9,8 %
Massen 1	17,0 %	12,9 %	12,8 %	99,8 %	99,8 %	99,4 %	96,4 %	96,4 %	95,9 %
Massen 2	3,2 %	3,2 %	3,2 %	99,8 %	100 %	99,7 %	93,8 %	93,9 %	93,6 %
Massen 3	1,2 %	0,9 %	0,6 %	56,9 %	47,5 %	38,1 %	0,0 %	0,0 %	0,0 %

Tabelle 3: Erkennungsraten der drei Bibliotheken bei einer Auflösung von 1920x1080 Pixeln

Bibliothek	BART			ALVAR			ARUCO		
	48	24	12	48	24	12	48	24	12
Bewegung 2	99,0 %	100 %	88,0 %	99,5 %	100 %	100 %	99,5 %	100 %	100 %
Bewegung 3	36,4 %	36,2 %	36,8 %	45,5 %	45,6 %	45,6 %	24,1 %	24,0 %	24,0 %
Rotationen 2	75,7 %	76,1 %	76,5 %	80,3 %	80,0 %	81,4 %	75,4 %	76,1 %	76,5 %
Rotationen 4	72,3 %	70,5 %	73,0 %	8,7 %	10,5 %	11,0 %	72,3 %	72,5 %	73,0 %
Kontrast 2	99,8 %	100 %	100 %	99,8 %	100 %	100 %	99,8 %	100 %	100 %
Kontrast 7	91,6 %	91,6 %	92,1 %	99,8 %	100 %	100 %	91,4 %	91,1 %	90,1 %
Kontrast 9	91,4 %	24,9 %	29,5 %	27,3 %	26,7 %	31,2 %	91,4 %	24,9 %	29,5 %
Kontrast 10	32,2%	8,9%	11,6%	8,6%	8,4%	11,6%	26,4%	7,1%	9,8%
Massen 1	16,8 %	16,8 %	16,7 %	99,8 %	99,8 %	99,3 %	96,4 %	96,4 %	95,9 %
Massen 2	4,7 %	4,7 %	3,2 %	99,8 %	100 %	99,7 %	93,8 %	93,9 %	93,6 %
Massen 3	1,0 %	0,7 %	0,5 %	56,9 %	47,5 %	38,1 %	0,0 %	0,0 %	0,0 %

Tabelle 4: Erkennungsraten der drei Bibliotheken bei einer Auflösung von 1280x720 Pixeln

Bibliothek	BART			ALVAR			ARUCO		
	48	24	12	48	24	12	48	24	12
Bewegung 2	99,5 %	100 %	100 %	99,5 %	100 %	100 %	99,5 %	100 %	100 %
Bewegung 3	29,9 %	30,0 %	30,0 %	36,7 %	36,8 %	40,8 %	24,0 %	24,0 %	24,0 %
Rotationen 2	74,9 %	74,6 %	75,5 %	79,6 %	80,0 %	80,4 %	73,7 %	73,2 %	76,5 %
Rotationen 4	70,8 %	71,0 %	71,0 %	10,7 %	12,5 %	12,0 %	70,6 %	70,0 %	71,0 %
Kontrast 2	99,8 %	100 %	100 %	99,8 %	100 %	100 %	99,8 %	100 %	100 %
Kontrast 7	91,4 %	91,6 %	92,1 %	99,8 %	100 %	100 %	91,4 %	91,6 %	92,1 %
Kontrast 9	91,4 %	24,9 %	29,5 %	27,3 %	26,7 %	31,2 %	91,4 %	24,9 %	29,5 %
Kontrast 10	32,2 %	8,9 %	11,6 %	8,6 %	8,4 %	11,6 %	25,5 %	7,6 %	9,8 %
Massen 1	15,3 %	15,1 %	12,8 %	99,8 %	99,8 %	99,3 %	96,4 %	96,4 %	95,9 %
Massen 2	6,7 %	6,0 %	4,7 %	99,8 %	100 %	99,7 %	93,8 %	93,9 %	93,6 %
Massen 3	1,7 %	1,3 %	1,0 %	54,9 %	45,7 %	36,5 %	0,0 %	0,0 %	0,0 %

Tabelle 5: Erkennungsraten der drei Bibliotheken bei einer Auflösung von 630x360 Pixeln

## 6.2 Ressourcenverbrauch (in der Auflösung 1080p48)

Tabelle 6 zeigt den Ressourcenverbrauch bei der Auswertung in der höchsten Auflösung.

Bibliothek	BART			ALVAR			ARUCO		
	RAM in MB		cpu-Zeit	RAM in MB		cpu-Zeit	RAM in MB		cpu-Zeit
	min.	max.	in s	min.	max.	in s	min.	max.	in s
Bewegung 2	317	2430	21,0	1051	1064	6,7	0	1106	5,5
Bewegung 3	317	6880	59,5	32	1064	26,2	309	1106	21,0
Kontrast 2	317	3585	37,7	1065	1074	12,3	286	978	9,8
Kontrast 4	317	3657	32,9	1058	1074	10,3	309	978	9,7
Massen 1	1177	6055	75,5	1032	1065	31,5	914	1108	32,8
Massen 2	317	6618	110,3	1053	1077	53,9	0	1109	71,4
Massen 3	317	6524	183,9	1049	1081	110,3	914	920	46,3

Tabelle 6: Ressourcenverbrauch bei ausgewählten Videos

## 7 Diskussion

Diese Arbeit hat gezeigt, dass es möglich ist, mit dem Ansatz der virtuell simulierten Marker viele Eigenschaften von Trackingbibliotheken zu testen. Dabei können unter gezielten Bedingungen eine große Menge an Daten zu verschiedenen Eigenschaften, wie die allgemeine Erkennungsrate, die Translations- und Rotationsgenauigkeit, der Ressourcenverbrauch und das genaue Erkennungsprofil, gewonnen werden.

Wie hier an drei Bibliotheken beispielhaft gezeigt, lassen sich damit die unterschiedlichen Stärken und Schwächen der Bibliotheken herausarbeiten, was einen Vergleich in Bezug

auf die Anforderungen eines speziellen Projektes möglich macht. Die Testfälle sind dabei systematisch so erstellt, dass sich auch Rückschlüsse auf einzelne Prozessschritte der Bibliotheken (z.B. Binarisierung, Rauschunterdrückung, Tracking) ziehen lassen. Dies liefert bei Eigenentwicklungen, wie z.B. BART, konkrete Hinweise zur Optimierung.

Darüber hinaus wurde für die behandelten Eigenschaften ein Testbett beschrieben, das auch für den Vergleich mit weiteren Bibliotheken geeignet ist. Leider konnten wir auf Grund des Umfangs der Einreichung nur ausgewählte Testfälle und ausgeählte Bibliotheken präsentieren und jeweils nur einen Teil der gewonnenen Daten darstellen. Das Test-Framework mit den Testdaten und dem notwendigen Programmcode wird als OpenSource-Veröffentlichung zugänglich gemacht. Es ist ab Veröffentlichung unter <http://bart.multimodal-interaction.org> erreichbar.

## Acknowledgement

Diese Arbeit wurde anteilig gefördert durch den Exzellenzcluster Cognitive Interaction Technology 'CITEC' (EXC 277) an der Universität Bielefeld, der durch die Deutsche Forschungsgemeinschaft (DFG) finanziert wird.

## Literatur

- [GJMSMCMJ14] GARRIDO-JURADO, S., R. MUÑOZ-SALINAS, F.J. MADRID-CUEVAS und M.J. MARÍN-JIMÉNEZ: *Automatic generation and detection of highly reliable fiducial markers under occlusion*. Pattern Recognition, 47(6):2280 – 2292, 2014.
- [KAS10] KORKALO, O., M. AITTALA und S. SILTANEN: *Light-weight marker hiding for augmented reality*. In: *Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on*, Seiten 247–248, 2010.
- [KB99] KATO, HIROKAZU und MARK BILLINGHURST: *Marker tracking and HMD calibration for a video-based augmented reality conferencing system*. In: *Proceedings of IWAR 99*, 1999.
- [PMK06] PENTENRIEDER, KATHARINA, PETER MEIER und GUDRUN KLINKER: *Analysis of Tracking Accuracy for Single-Camera Square-Marker-Based Tracking*. In: *Proceedings des Workshops der GI-Fachgruppe VR/AR*, 2006.
- [VTS14] VERBANDT, MAARTEN, BART THEYS und JORIS DE SCHUTTER: *Robust marker-tracking system for vision-based autonomous landing of VTOL UAVs*. In: *Proceedings of the International Micro Air Vehicle Conference and Competition 2014*. Delft University of Technology, 2014.