

TRANSFORMATION BASED OBJECT CLASSIFICATION. EIN VERFAHREN ZUR AUTOMATISCHEN INITIALISIERUNG VON AUGMENTED REALITY-ANWENDUNGEN

Hans Christian Arlt und Cornelius Malerczyk

Technische Hochschule Mittelhessen
Fachbereich Mathematik, Naturwissenschaften und Datenverarbeitung
Friedberg, Deutschland

ABSTRACT

Diese Arbeit befasst sich mit der Problemstellung der automatischen Initialisierung für Augmented Reality Applikationen im mobilen Anwendungsbereich. Es wird ein Verfahren entwickelt, welches ohne sichtliche Initialisierungsmarken eine Augmented Reality-Anwendung automatisch lädt. In der Regel wird hierfür auf Marker meist in der Form von QR-Codes zurückgegriffen. Das Verfahren soll die vorhandenen Sensoren und verschiedene Bildanalyseverfahren nutzen, um eine Klassifizierung von Objekten zu realisieren. Die Objekte müssen vorher in einer Anlernphase im System hinterlegt werden. Hierbei werden eine Vielzahl von Trainingsdaten abgelegt, die beim Erkennen dann mit den aktuellen Werten abgeglichen werden. Im Rahmen des Forschungsprojektes „PIMAR“¹ wird ein Prototyp dieses Verfahrens in einer Android-Anwendung für das mathematische Mitmach-Museum „Mathematikum“² in Gießen umgesetzt. In der Anwendung werden exemplarisch einzelne Exponate über das Verfahren erkannt und die zugehörige AR-Szene geladen.

Index Terms— Augmented Reality, AR, Klassifikation, mobile Anwendungen, Sensorik

1. EINLEITUNG

Erweiterte Realität (engl.: Augmented Reality, kurz: AR) gewinnt durch die große Verbreitung der neuen Generation von mobilen Telefonen, den sogenannten „Smartphones“, in letzter Zeit einen immer größeren Bekanntheitsgrad. AR ist eine Anreicherung von Zusatzinformationen in Form von virtuellen Daten, die in Abhängigkeit der zu betrachtenden Realität situationsgerecht angezeigt werden [1].

Die heutige Technik der mobilen Telefone ist durch die eingebauten Kameras und mehreren Sensorikbauteilen gut nutzbar, um einfache Erweiterungen im Kamerabild auf dem Display umzusetzen. Jedoch kommen die heutigen mobilen Telefone mit begrenzter Prozessorleistung und Arbeitsspeicher an ihre

Grenzen, wenn es darum geht, eine Vielzahl von aufwendigen und großen AR-Trackingdaten mit dem aktuellen Kamerabild in Echtzeit zu vergleichen. Optisches AR-Tracking kann in drei Arten unterschieden werden:

1. 2D markerbasiert (z.B. QR-Code, Bildmarken etc.)
2. Multimarkerbasiert (mehrere Marker mit festen Abständen)
3. Markerlos (2D bildbasiert oder 3D Punktwolkenbasiert)

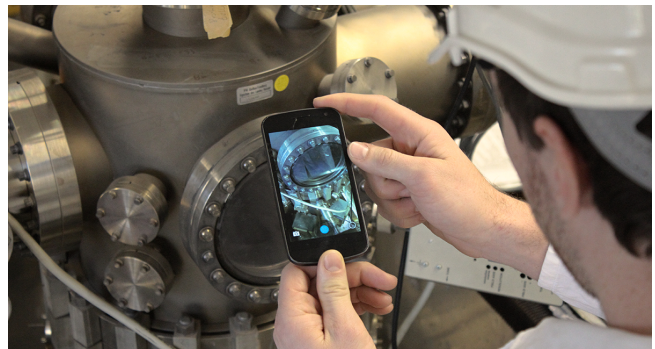


Abbildung 1. Industrielles AR-Anwendungsszenario aus PIMAR.

Markerlose Tracking-Systeme sind die anspruchsvolleren Verfahren. Zur Erkennung werden Bilder (2D Markerless) oder reale Objekte (3D Markerless) benutzt. Hierbei spielt die natürliche Struktur des Bildes oder die Beschaffenheit und Form des Objektes eine Rolle.³ Beim 3D Markerless-Verfahren werden Punktwolken mit Tiefeninformationen benutzt, um Objekte oder auch Räumlichkeiten zu erkennen und zu verfolgen (siehe Abbildung 2). Im Projekt PIMAR ist die Zielsetzung, ohne Benutzereinwirkung ortsgebunden unterschiedliche markerlose Tracking-Szenarien zu laden. Es gibt

¹<http://pimar.mnd.thm.de>

²www.mathematikum.de

³www.wearear.de/sensor-fusion-augmented-reality-tracking/ Stand: 18. November 2014

verschiedene SDKs auf dem Markt, die es ermöglichen nach mehreren Punktwolken zu suchen und das jeweilige Tracking zu starten bzw. die gewünschte Information anzuzeigen. In dem SDK von metaio⁴, welches im Projekt PIMAR zur Verwendung kommt, ist die maximale Anzahl von Trackingdateien nach eigenen Angaben auf 15 limitiert.

Jedoch machen sich schon kleinere Anzahlen sehr stark



Abbildung 2. Markerless 3D-Tracking: Punktwolken (links), Tracking und Anzeige eines virtuellen Objektes (rechts).

in einer sinkenden Framerate der Anzeige beim Tracking bemerkbar. Der Grund dafür liegt auf der Hand: Für jedes potentielle Szenario, dessen virtuelle Objekte dargestellt werden sollen, muss permanent eine angelernete Punktwolke im Gerätespeicher gehalten und wiederholt mit den aktuellen Kamerabildern abgeglichen werden. Die große Datenmenge fordert hier nicht nur entsprechend viel Speicher, sondern insbesondere auch eine hohe Prozessorlast. Erst wenn das System eine der 3D-Punktwolken erfolgreich zuordnen kann, können die anderen Punktwolken verworfen und die Verfolgung der Szene und deren virtuelle Überlagerung gestartet werden.

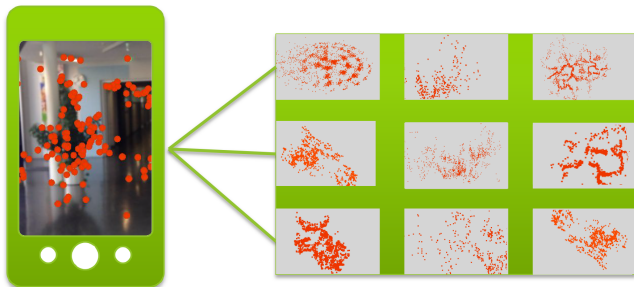


Abbildung 3. Gleichzeitiger Abgleich mehrerer 3D-Punktwolken.

In vielen Fällen werden deshalb zur Initialisierung QR-Codes verwendet⁵ oder der Benutzer muss über eine grafische Benutzeroberfläche in der Anwendung das jeweilige Szenario starten. Um dieses Problem zu lösen, wurden Überlegungen getroffen, wie man mit der vorhandenen Technik eine automatische Initialisierung vor das eigentliche AR-Tracking

⁴www.metaio.com

⁵<http://goo.gl/DNqJT8> Stand: 19. November 2014

setzen kann. Daraus wurde dann das Verfahren „Transformation Based Object Classification“ kurz TBOC entwickelt und in einer mobilen prototypischen Applikation für das mathematische Mitmach-Museum „Mathematikum“ umgesetzt. Es wird dafür verwendet, unterschiedliche Exponate im Museum zu erkennen und daraufhin passende AR-Szenen zu laden.

2. VERWANDTE ARBEITEN

Mit der Problematik der automatischen Initialisierung und der damit verbundenen Positions- und Orientierungsbestimmung für AR-Tracking beschäftigt sich beispielsweise die Hochschule Reutlingen, die sich für eine AR-Schnitzeljagd für interessierte Schüler im Fachbereich Informatik mit der Thematik der Positionsbestimmung gerade in Innenräumen auseinander gesetzt haben [2]. Im Außenbereich kann üblicherweise mittels GPS-Ortung eine ungefähre Position bestimmt werden. In Innenräumen greifen die Entwickler aus Reutlingen auf sogenannte LLA-Marker (Latitude, Longitude, Altitude) zurück, die es innerhalb des Frameworks von metaio ermöglichen, eine Position zu bestimmen.

Gerhard Reitmayr und Tom W. Drummond beschreiben in ihrem Artikel „Initialisation for Visual Tracking in Urban Environments“ die Kombination von GPS- und visuellem Tracking. Bei diesem Verfahren spielt die Sensorfusion mit dem Orientierungssensor die entscheidende Rolle zur Bestimmung der Szenerie. Der Schwerpunkt der Arbeit liegt bei der Reinitialisierung des Trackings [3].

Yunpeng Li et al. schreiben in ihrer Arbeit „Location Recognition using Prioritized Feature Matching“ über eine schnelle und einfache Ortbestimmungs- und Bildlokalisierungsmethode [4]. Aus einer großen Sammlung von Bildern werden mit *Image Matching*-Verfahren und *Structure from Motion*-Techniken 3D-Modelle erzeugt. Mit Hilfe eines SIFT (*Scale Invariant Feature Transform*)-Verfahrens werden Übereinstimmungen zwischen den Bildern des verwendeten SfM und der rekonstruierten 3D-Geometrie gesucht. Ähnliche Ansätze für Ortsbestimmung für große Gebiete, wie zum Beispiel einer Stadt, mit Hilfe von großen Bilddatenbanken werden in der Arbeit von Grant Schindler et al. „City-Scale Location Recognition“ [5] sowie in der Arbeit von Duncan Robertson und Roberto Cipolla „An Image-Based System for Urban Navigation“ aufgezeigt [6].

In der Arbeit von Niels Henze et al. „What is That? Object Recognition from Natural Features on a Mobile Phone“ wird ein markerloses Objekterkennungsverfahren beschrieben, welches mit mehreren Kamerabildern von einem mobilen Telefon, Bilder und sogar Objekte erkennt [7]. Das Verfahren kombiniert dabei ein vereinfachtes SIFT-Verfahren, einen skalierbaren *Vocabulary Tree* und einen einfachen *Feature Matching*-Algorithmus. In den meisten Arbeiten wird bei der Ortsbestimmung bzw. Blickrichtungsbestimmung der Schwerpunkt auf GPS-Unterstützung gelegt. Aus dieser Arbeit ist erkennbar, dass die Fusion einzelner Sensoren eines

Mobiltelefons eine wichtige Grundlage bei der Klassifizierung spielen kann. Eine sehr ähnliche Herangehensweise an die Problematik zeigt sich in der Arbeit von Niels Henze [7]. Bei dem in dieser Arbeit vorgestellten Verfahren soll jedoch auf die Nutzung großer Bilddatenbanken und somit auf den Zugriff des Internets verzichtet werden.



Abbildung 4. Überblick über das Verfahren der AR-Initialisierung in ner Museumsanwendung auf einem Tablet.

3. TRANSFORMATION BASED OBJECT CLASSIFICATION

Um die Echtzeitfähigkeit der Erkennung bei einer Vielzahl von unterschiedlichen Objekten zu gewährleisten, müssen alle sensorischen Werte auf ein Minimum reduziert werden. Dafür werden alle Sensordaten inklusive der Kamerabilder zur Laufzeit in einen Parameterraum übertragen. Mit dieser *Transformation Based Object Classification* (kurz: TBOC) verwendet das Verfahren die im Smartphone integrierte Sensorik, um bei gestarteter Applikation automatisch die entsprechenden Objekte zu detektieren und damit in einem folgenden Schritt die notwendigen Modelldaten für die Objekterkennung und -verfolgung und für die 3D-Überlagerungen selektiv zu bestimmen. Es handelt sich um ein statistisch begründetes Verfahren, das Objekte durch Extraktion von geeigneten Sensorparametern als Merkmalsvektoren in einem Wahrscheinlichkeitsnetz klassifiziert und die gewonnenen Daten in einer Modelldatei speichert. Dieses Modell wird dann zur Klassierung von zu erkennenden Objekten verwendet. TBOC ist also in zwei Phasen unterteilt: In einer ersten Offline-Trainingsphase werden alle Objekte, die später zur Laufzeit der Anwendung durch den Benutzer erkannt werden sollen, automatisch sensorisch vermessen und die Trainingsergebnisse in einer Datenbank abgespeichert, die zusammen mit der Anwendung selbst auf dem Smartphone des Anwenders installiert wird. Diese einmalige Trainingsphase benötigt in der Praxis nur wenige Minuten pro Objekt. Die zweite Phase bedient sich des aus den Trainingsdaten gewonnenen statistischen Modells, um zur Laufzeit der Anwendung unter

einer Vielzahl von Objekten das am wahrscheinlichsten relevante zu bestimmen und das entsprechende AR-Modell zu initiieren.

4. INITIALISIERUNG DURCH KLASSIFIZIERUNG

Für die Klassifikation von Objekten soll ein schnelles und effizientes Verfahren eingesetzt werden. Es soll auch keine weiteren Hilfsmittel bzw. Marker (Veränderungen am Objekt oder der Umgebung) verwenden, sondern lediglich Sensorwerte nutzen, die das Mobiltelefon liefert sowie die Analyse von Bildinformationen der Kamera. Unter der Voraussetzung, dass diese Klassifikationen über eine gemeinsame Wahrscheinlichkeitsverteilungsfunktion zusammenhängen, kann mit Hilfe von Bayes-Klassifikator-Verfahren eine Entscheidung für ein Objekt getroffen werden [8].

4.1. Sensorik

Es bietet sich an, für die Klassifizierung die Sensorik zu verwenden, die auch eine Relevanz bei dem späteren AR-Tracking hat. Herangezogen werden der Orientierungssensor (Magnetometer) und die Telefonausrichtung über den Drehbewegungssensor (Gyrometer). So werden die Blickrichtung und auch die Neigung (Pitch & Roll) des Gerätes in der Klassifikation berücksichtigt. Das GPS liefert in Innenräumen meist keine verlässlichen Ergebnisse. Jedoch sollten, wenn verlässliche GPS-Daten vorliegen, diese auch in die Klassifikation integriert werden. Als wichtiger weiterer Sensor wird in dieser Arbeit das Kamera-System des Smartphones verwendet. Die Kamera ist in der Lage, verlässliche Informationen über Bildauswertungen für die Objekterkennung beizusteuern. Allerdings soll im folgenden nicht die Kamera selbst als sensorische Einheit angesehen werden. Vielmehr ist eine geeignete Auswahl an bildanalytischen Verfahren wichtig für eine erfolgreiche Klassifizierung von Objekten.

4.2. Bildanalyse

Neben den Sensorwerten werden für die Klassifikation Werte einbezogen, die über zwei verschiedene Bildverarbeitungsalgorithmen ermittelt werden. Das Verfahren soll möglichst mit geringem Datenvolumen, also ohne große Bilddatenbanken, und ohne Internetverbindung auskommen, so dass alle benötigten Daten direkt auf dem Anwendungsgerät abgelegt werden können. Unter diesen Voraussetzungen werden keine Bilder abgespeichert, sondern nur die jeweiligen RGB-Werte ermittelt und in die Klassifikation einbezogen.

4.2.1. Minimal-Histogramme

Das Minimal-Histogramm-Verfahren ermittelt eine Farbverteilung des Kamerabildes in Farbkanälen. Das Histogramm wird jeweils in RGB-Werte aufgeteilt und die von der Kamera

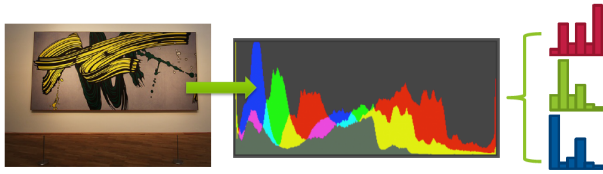


Abbildung 5. Exemplarische Darstellung der Erstellung reduzierter RGB-Histogramme.

vorgegebene Unterteilung von 256 Balken auf einen beliebigen kleineren Wert reduziert (siehe Abbildung 5). Die daraus resultierenden Werte werden dann als Trainingsdaten abgespeichert oder zur Klassifikation bereitgestellt. Die Reduktion der Histogrammwerte auf einen minimalen Wert von bis zu nur drei Werten pro Farbkanal hat mehrere Vorteile. Zum einen reduziert sich mit nur wenigen Parametern der Rechenaufwand während der Klassifikation und ermöglicht eine hohe Bildwiederholrate der Kamera, zum anderen lassen die nur wenigen Werte ausreichend Farbabweichung in den Kamerabildern zu, wie sie typischerweise unter realen Bedingungen durch sich ändernde Lichtverhältnisse vorkommen.



Abbildung 6. Darstellung der regionalen RGB-Durchschnittsberechnung in 25 Feldern.

4.2.2. Regionale RGB-Mittelwerte

Neben der zuvor beschriebenen allgemeinen Farbverteilung über reduzierte Histogramme kommt außerdem die Ermittlung einer Regionen-spezifische Farbverteilung als Sensorik zum Einsatz. Das Verfahren basiert auf dem „Naive Similarity Finder“ Algorithmus von R. Santos aus seinem *Java Image Processing Cookbook*⁶ und verwendet ursprünglich Bildregionen und deren Farbinformationen, um die Ähnlichkeit von zwei verschiedenen Bildern zu schätzen. Das Verfahren ermittelt zunächst in einer 5x5-Matrix relevante Regionen im Kamerabild. In jedem der 25 Felder wird der RGB-Farbmittelwert berechnet (siehe Abbildung 6). Die daraus resultierenden RGB-Werte werden ebenfalls zur Speicherung der Trainingsdaten und zur Klassifikation bereitgestellt.

⁶<http://goo.gl/X8im6q> Stand: 26. November 2014

Während bei Santos die Mittelwerte zweier Bilder direkt miteinander verglichen werden, werden die RGB-Mittelwerte in der hier beschriebenen Anwendung zunächst für die in der offline-Phase aufgenommenen Referenzbilder ermittelt und als Teil der Parametervektoren für die statistische Klassifizierung der Objekte verwendet.

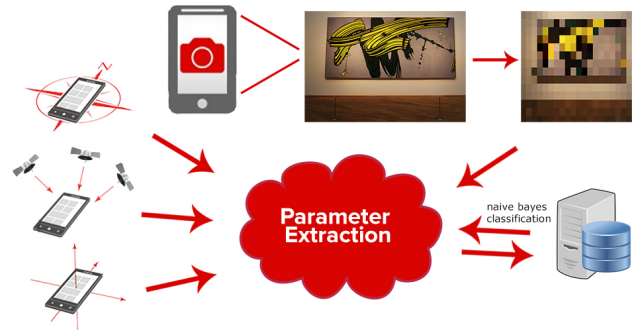


Abbildung 7. Funktionale Umsetzung des TBOC Verfahrens.

5. FUNKTIONALE UMSETZUNG

Für die Klassifizierung und Klassierung der Parametervektoren wird das von der *University of Waikato*, Neuseeland entwickelte *Data Mining*-Paket Weka verwendet [9]. Die Open Source-Software liefert alle notwendigen statistischen Implementierungen und lässt sich ohne die Verwendung der Klassen für grafische Benutzeroberflächen leicht auf beispielsweise Android übertragen.

Das TBOC-Verfahren ist in zwei Phasen unterteilt. Die Anlernphase, die offline stattfindet und in der die Exponate trainiert werden und die gesammelten Daten (Attribute) in eine Datei vom Typ „Attribute-Relation File Format“ kurz ARFF geschrieben werden. Das ARFF-Format ist eine einfache Textdatei, die mit einer Reihe von Attributen Instanzen beschreibt⁷. Die zweite Phase ist die Online-Phase des Verfahrens und implementiert die Erkennungsphase, in der ein neuer Parametervektor der aktuellen Sensorsituation mit dem aus dem ARFF-Datei generierten statistischen Modell mittels einem Bayes-Klassifikator verglichen wird und eine Wahrscheinlichkeitsverteilung und somit eine Klassifikation des Objektes ermöglicht. Wie schon beschrieben, sollen die Sensorikwerte des Magnetometers sowie des Gyrometers für die Klassifikation benutzt werden. In der Anlernphase werden diese in einem Intervall ausgelesen und in einem Vektor zwischengespeichert, um sie anschließend zu mitteln. Die Anzahl der Intervalldurchläufe wurde auf mindestens sechs Durchläufe festgelegt, damit genügend Werte vorhanden sind um bei der Mittelung Fehler sowie ausreißende Datenwerte zu eliminieren. Wie in Abbildung 7 erkennbar, werden die von der Kamera aufgezeichneten Bilder in dem

⁷<http://goo.gl/dUZBj3> Stand: 26. November 2014

Intervall auf eine quadratische Vorschaubildgröße skaliert, um so eine schnelleren Berechnung der Bildverarbeitungsalgorithmen zu gewährleisten. Auf das Vorschaubild werden dann nacheinander die beiden Algorithmen angewendet und die zurückgegebenen Werte ebenfalls in einen Vektor geschrieben, um am Ende des Intervalls daraus die jeweiligen Mittelwerte errechnet. Die einzelnen errechneten Mittelwerte werden anschließend nach Benutzerbestätigung in die ARFF-Datei geschrieben und dabei einem Exponat zugeordnet. In der zweiten Phase wird aus der ARFF-Datei das zugehörige Modell generiert. Um die Applikation beim Laden der Erkennungsphase zu entlasten, wurde festgelegt, dass das Generieren des Modells nur automatisch erfolgen soll, wenn kein Modell vorhanden ist. Wenn danach Änderungen in der ARFF-Datei vorgenommen werden, kann das Modell über ein Benutzermenü bei Bedarf aktualisiert werden. Die Erkennungsphase ist der Anlernphase sehr ähnlich, auch hier werden über ein Intervall Daten gesammelt, aber nach der Mittelung werden diese in einem Vergleichsvektor gespeichert, der dann mit dem Modell mittels der *NaiveBayes*-Methode verglichen wird. Je nach erkanntem Objekt kann nun die passende AR-Szene geladen werden.

6. ANWENDUNGSSZENARIEN

Für das prototypisch implementierte Verfahren wurden einige Testanwendungen entwickelt. Neben industriellen Szenarien wie beispielsweise einer Anwendung für die Wartungsunterstützung von Großmaschinen oder touristischen Szenarien zur Augmentierung von wichtigen Gebäuden wurde auch eine Anwendung im musealen Kontext entwickelt.

Für das Mathematikum in Gießen wurden einige der dort ausgestellten Exponate in die Anwendung aufgenommen, um dem Besucher entweder bei der Lösung des gezeigten Problems zu unterstützen oder ihm Zusatzinformationen bereitzustellen. Für das Anwendungsszenario *Pi-Kreis* erkennt die

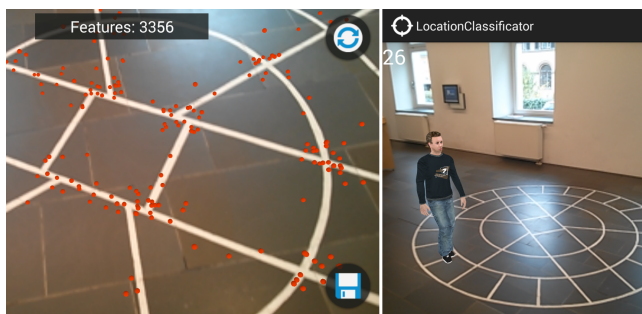


Abbildung 8. Anwendungsszenario *Pi-Kreis* im Mathematikum. Punktwolke augmentiert (links) und überlagerte Animation eines Avatars (rechts).

Anwendung den Raum, in dem ein weißes Kreismuster auf den dunklen Bodenfliesen angebracht ist. Der Besucher soll die Schritte zählen, die er auf dem Kreis für eine Umrundung

benötigt, genauso wie die Anzahl für dessen Durchmesser, um aus dem Quotienten die Zahl Pi anzunähern. Die AR-Anwendung unterstützt dabei das Verständnis, dass je größer die Zahlen werden, die Schätzung genauer wird, indem Animationen eines Menschen und einer Maus mit kleineren und deshalb mehr Schritten dem Kamerabild überlagert werden können. Im Anwendungsszenario *Deutschlandkarte* soll der

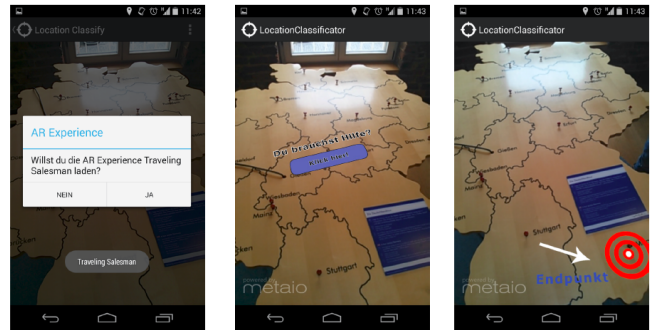


Abbildung 9. Anwendungsszenario *Deutschlandkarte*. Augmentierte Hilfefunktion eines *Traveling Salesman*-Problems.

Museumsbesucher die kürzeste Strecke zwischen elf Städten finden. Die am Startpunkt befestigte Schnur reicht nur bis zur letzten Stadt, wenn dieses kombinatorische Problem richtig gelöst wird. Das als NP-vollständig bekannte Problem des Handlungsreisenden kann den Museumsbesucher schnell frustrieren und enttäuschen. Mit der AR-Anwendung ist der Besucher aber in der Lage, gezielt Hilfe anzufordern und sich die Karte im Kamerabild mit entweder den erste Stationen der Reise oder aber mit der Zielstadt erweitern zu lassen.

7. ERGEBNISSE

Das Verfahren wurde mit verschiedenen Prototypen sowohl unter Laborbedingungen als auch unter realen Bedingungen in einer Museumsanwendung (Mathematik in Gießen) getestet. Daraus entstand eine finale Prototyp-Applikation, die in der Lage ist, Objekte in Innenräumen und Außenbereichen zu erkennen. Hierfür sind keine explizit angebrachten Marker wie beispielsweise QR-Codes nötig. Das hier entwickelte markerlose Verfahren ermittelt die Objektbestimmung über eine statistische Wahrscheinlichkeitsverteilung anhand eines Vergleiches von Sensordaten und einem zuvor angelearnen Datenmodell. Hierfür wird auf einen Bayes-Klassifikator zurückgegriffen. In den vorläufigen Testverfahren ergaben sich Erkennungsraten von über 90%. Für die Aufnahme von Datenvektoren zur Klassierung der Objekte wurde im Schnitt eine Zeit von drei bis fünf Sekunden gemessen. Eine ausführliche Testreihe und anschließende Evaluation wird in einer weiteren Arbeit durchgeführt. Jedoch sind die Resultate der vor Ort im Museum durchgeführten Tests schon ein erster deutlicher Hinweis darauf, dass der verwendete Algorithmus im gegebenen Szenario unter realen Bedingungen

gewünschte Ergebnisse erzielt. Kleinere Störungen, wie teilweise Verdeckungen der Objekte durch Personen etc. haben keine Auswirkungen auf das System und die Erkennungsrate. Somit ist eine Benutzung im alltäglichen Museumsbetrieb sichergestellt.

8. ZUSAMMENFASSUNG UND AUSBLICK

In dieser Arbeit wurde das Verfahren TBOC vorgestellt und die Funktionsweise beschrieben. Das Verfahren ermöglicht eine markerlose Initialisierung zum Laden von komplexen AR-Szenarien ohne Benutzereinwirkung. Wenn reale Objekte als AR-Referenz dienen, wird auf das sogenannte 3D Markerless-Verfahren zurückgegriffen. Bei diesem Verfahren werden Punktwolken mit Tiefeninformationen verwendet. Die unterschiedlichen AR-SDKs bieten zwar die Möglichkeit mehrere Punktwolken zur selben Zeit zu vergleichen und zu erkennen, jedoch führt dies schon bei wenigen Punktwolken zu Leistungseinbrüchen und somit niedrigen Frameraten. Um den Benutzer der Anwendung nicht in den Prozess der Initialisierung einzubinden, in dem er entweder über einen sogenannten QR-Code oder eine Menüstruktur immer das gewünschte AR-Szenario starten muss, wurde mit dem TBOC Verfahren eine Möglichkeit geschaffen, eine automatische Objekterkennung vor die eigentliche AR-Szene zu setzen, die dann je nach erkanntem Objekt die passende Szene lädt. Für die Erkennung und die sogenannte Klassifizierung der Objekte wird mittels der Sensorik der Mobiltelefone und einer Analyse des Kamerabildes mit zwei verschiedenen Bildverarbeitungsalgorithmen ein Datenvektor erstellt, der mittels der „NaiveBayes“-Methode eine Wahrscheinlichkeitsverteilung vornimmt. Für die Klassifikation mittels eines Bayes-Klassifikators ist es notwendig, dass für die zu erkennenden Objekte bekannte Klassifikatoren durch Trainingsdaten angelernt werden. Um Fehler zu minimieren, werden die Werte in einem Intervall gesammelt und anschließend gemittelt. Das Verfahren wurde im Rahmen des Forschungsprojektes PIMAR exemplarisch in einem Prototypen für das Mathematikum in Gießen umgesetzt und dort vor Ort auf seine Praktikabilität getestet.

Mit diesem Verfahren ist es möglich, in Innenräumen, sowie auch im Außenbereich, Objekte zu erkennen und in einer mobilen Anwendung virtuelle Inhalte entsprechend zu laden. Dies konnte am Prototypen für das mathematische Mitmach-Museum *Mathematikum* gezeigt werden, bei dem nicht nur AR-Szenen bei der Erkennung von Exponaten geladen werden, sondern auch sogenannte Minispiele direkt in der Anwendung. So zeigt sich mit diesem Anwendungsszenario, dass in einem Museum mit dem in dieser Arbeit gezeigten Verfahren Objekte und Räume erkannt werden können, um diesen beliebige Informationen zuzuordnen. Diese Informationen können direkt auf dem Exponat angezeigt werden (AR) oder einfach nur mit Hilfe von Text und Bild in der Anwendung angezeigt werden.

9. DANKSAGUNG

Diese Arbeit wurde im Rahmen des Forschungsprojekts *PIMAR, Platform Independent Mobile Augmented Reality* entwickelt. PIMAR (HA-Projekt-Nr.: 355/12-45) wird im Rahmen von Hessen Modellprojekte aus Mitteln der LOEWE - Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz, Förderlinie 3: KMU-Verbundvorhaben gefördert.

10. REFERENZEN

- [1] Olaf Oehme, *Ergonomische Untersuchung von kopfbasierten Displays für Anwendungen der erweiterten Realität in Produktion und Service.*, Ph.D. thesis, RWTH Aachen University, 2004.
- [2] Bernhard Thiersch, “Entwurfsentscheidungen zur augmentedreality-schnitzeljagd augmenture,” in *Informatics Inside: Grenzen überwinden - Virtualität erweitert Realität*. 2011, pp. 17–21, Hochschule Reutlingen.
- [3] Gerhard Reitmayr and T.W. Drummond, “Initialisation for visual tracking in urban environments,” in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, Nov 2007, pp. 161–172.
- [4] Yunpeng Li, Noah Snavely, and Daniel P. Huttenlocher, “Location recognition using prioritized feature matching,” in *ECCV (2)*. 2010, vol. 6312 of *Lecture Notes in Computer Science*, pp. 791–804, Springer.
- [5] G. Schindler, M. Brown, and R. Szeliski, “City-scale location recognition,” in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, June 2007, pp. 1–7.
- [6] Duncan Robertson and Roberto Cipolla, “An image-based system for urban navigation,” in *Proceedings of the British Machine Vision Conference*. 2004, pp. 819–828, BMVA Press.
- [7] Niels Henze, Torben Schinke, and Susanne Boll, “What is that? object recognition from natural features on a mobile phone,” in *Proceedings of the Workshop on Mobile Interaction with the Real World*, Sep 2009.
- [8] Thomas M. Mitchell, *Machine Learning*, McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997.
- [9] I.H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*, The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2005.