

# Augmented Reality basierte Überlagerung von Sichteseinschränkungen an Flurförderzeugen

Augmented reality based overlay of visibility constraints on industrial trucks

Alexander Poschke <sup>1</sup>  
Lukas Jütte <sup>2</sup>  
Benjamin Küster <sup>1</sup>  
Malte Stonis <sup>1</sup>  
Ludger Overmeyer <sup>2</sup>

<sup>1</sup> IPH – Institut für Integrierte Produktion Hannover gGmbH

<sup>2</sup> Institut für Transport- und Automatisierungstechnik  
Fakultät für Maschinenbau,  
Leibniz Universität Hannover

**I**n diesem Artikel wird ein Verfahren zur Überlagerung von Sichteseinschränkungen, basierend auf dem Prinzip der Augmented Reality, vorgestellt. Das Verfahren beruht auf einer Überlagerung des tatsächlichen Bedienerblickfeldes mit Informationen aus einer rekonstruierten Szene. Die rekonstruierte Szene wird über den sichteseinschränkenden Komponenten als Hologramm eingeblendet. Das vorgestellte Verfahren gliedert sich in die Komponenten Positionsbestimmung, Datenübertragung und Visualisierung. Diese Softwarekomponenten werden detailliert vorgestellt. Im Hinblick auf den späteren Einsatz des Systems in einem Flurförderzeug werden zudem die Echtzeitfähigkeit der Datenübertragung, die Genauigkeit der Visualisierung und die Robustheit der Positionsbestimmung untersucht.

[Schlüsselwörter: Bildverarbeitung, Flurförderzeuge, Fahrerassistenzsystem, Augmented Reality, Hinderniserkennung]

**T**his article presents a method for superimposing vision constraints based on the principle of augmented reality. The method is based on an overlay of the actual operator's field of view with information from a reconstructed scene. The reconstructed scene is superimposed as a hologram only over the vision-restricting components. The presented method is divided into position determination, data transmission and visualization. These software components are presented in detail. In view of the later use of the system in an industrial truck, the real-time capability of the data transmission, the accuracy of the visualization and the robustness of the position determination are also investigated.

[Keywords: image processing, forklift trucks, driver assistance system, augmented reality, obstacle detection]

## 1 EINLEITUNG

Flurförderzeuge (FFZ) stellen einen wichtigen Bestandteil der intralogistischen Lieferkette dar. Im Jahr 2019 wurden allein in Deutschland 36.080 Unfälle mit Flurfördermitteln, davon 14.788 mit Gabelstaplern, gemeldet [Deu20]. Dabei gelten Sichteseinschränkungen des Fahrers als eine wesentliche Unfallursache [Zei12]. Mit konstruktiven Mitteln lässt sich das Sichtfeld des Fahrers jedoch nur schwer verbessern.

Ein häufig verwendeter Ansatz zur Reduzierung des Unfallrisikos sind Gabelzinkenkameras, die in der Regel einfach nachgerüstet werden können. Auf einem Bildschirm in der Fahrerkabine werden die aufgenommenen Umgebungsbilder angezeigt. Hieraus ergeben sich zwei Nachteile. Zum einen können die Bildschirme eine zusätzliche Sichteseinschränkung darstellen. Zum anderen müssen die Bilder durch den Fahrer aufgrund der abweichenden Perspektive interpretiert werden.

Im Rahmen des Projekts ISI-WALK wurde die Interpretation solcher Aufnahmen erleichtert, indem dem Fahrer Verfahrensempfehlungen zur Ausrichtung der Gabelzinken zur Verfügung gestellt werden [OM13, Kle18]. Das Assistenzsystem liefert einen Beitrag zur Reduzierung des Unfallrisikos bei der Ein- und Auslagerung, indem der Fahrer bei der Feinpositionierung des Staplers unterstützt wird. Als Gabelzinkenkamera wurde ein 3D-Kamerasystem, das nach dem Time-of-Flight-Prinzip arbeitet, verwendet. Eine automatisierte Paletten- und Regalerkennung wurde eingesetzt.

Durch den Einsatz von Augmented Reality (AR) bietet sich die Möglichkeit der Reduzierung des Sicherheitsrisikos bei der Bedienung von Flurförderzeugen, ohne zusätzliche Sichteseinschränkungen für den Fahrer zu erzeugen. Im Projekt „Einsatz der Augmented-Reality-Technologie zur

Unterstützung des Fahrers von Flurförderzeugen“ wurde bereits AR im FFZ eingesetzt. Sichteinschränkungen wurden nicht kompensiert. Stattdessen wurden zusätzliche Informationen, wie bspw. die Navigation zum Lagerplatz eingeblendet. Die Informationen wurden auf die Frontscheibe des FFZ projiziert [Wil15].

Zur Kompensation der Sichteinschränkungen wurden in anderen Anwendungen bereits transparente Cockpits realisiert. In der Automobilbranche erfolgte bspw. eine Projektion auf die sichteinschränkende Komponente [Yos08]. Der Projektor war am Kopf des Fahrers befestigt. Die Kopfposition des Fahrers musste stets bekannt sein, um ein perspektivisch korrektes Bild anzuzeigen. Die Kopfposition wurde durch einen Messarm ermittelt. In einem weiteren Projekt wurden Weitwinkel-Stereo-Kameras zur Umgebungsaufnahme verwendet und auf einer Virtual Reality (VR)-Brille angezeigt [Fra19].

In der Automobilbranche wurde zudem die Überblendung des vorausfahrenden Fahrzeugs beim Überholvorgang untersucht [GOF12]. Ziel war, die Sicht auf die davor liegende Fahrstrecke zu ermöglichen. Hierfür nimmt das vorausfahrende Fahrzeug ein nach vorne gerichtetes Kamerabild auf und sendet dieses an das Fahrzeug mit Überholabsicht. Kameras am überholenden Fahrzeug werden genutzt, um die Position des vorausfahrenden Fahrzeugs zu bestimmen und mit dem empfangenen Bild zu überlagern. Die Aufnahme wurde dem Fahrer mit Überholabsichten auf einem transparenten Bildschirm vor der Windschutzscheibe [GVF12] oder auf einer AR-Brille eingeblendet [Fer13]. Für letzteres wurde die in der Brille integrierte Kamera genutzt, um das vorausfahrende Fahrzeug zu erkennen.

Ziel des Forschungsprojekts „Virtuelle Sichtverbesserung und intuitive Interaktion durch Erweiterte Realität an Flurförderzeugen („ViSIER“) ist es, ein auf AR basierendes Bedienerassistenzsystem für Gabelstapler zu realisieren, mit dessen Unterstützung Sichteinschränkungen des Fahrers durch Fahrzeugkomponenten und Ladung ausgeglichen werden. Die Umgebungsaufnahme erfolgt durch strategisch am FFZ platzierte Kameras. Die Visualisierung der rekonstruierten Szene für den Bediener erfolgt auf der AR-Brille HoloLens 2 (Microsoft).

Für Landmaschinen wurde ein AR-basiertes Fahrerassistenzsystem entwickelt, das beim Manövrieren die Sicherheit deutlich erhöhen soll [Fri19]. Das System setzt ebenfalls eine HoloLens 2 zur Kompensation von Sichteinschränkungen und Einblendung zusätzlicher Informationen ein.

## 2 METHODE

In diesem Artikel wird eine Software zur Sichteinschränkungskompensation basierend auf dem Prinzip der AR vorgestellt. Im Folgenden findet zunächst eine

Beschreibung des Demonstratorfahrzeugs und der wichtigsten Hard- und Softwarekomponenten statt. Weiterhin wird die konzipierte Visualisierung detailliert erläutert und hinsichtlich ihrer Genauigkeit und Echtzeitfähigkeit untersucht. Die Genauigkeit und Echtzeitfähigkeit ist entscheidend für den Einsatz auf einem FFZ, um eine sowohl zeitlich als auch räumlich immersive Einblendung zu realisieren.

### 2.1 DEMONSTRATORFAHRZEUG

Als Demonstrator FFZ dient ein Gegengewichtstapler der Firma Linde (E16). Die Kameraanbaupositionen sind in Abbildung 1 dargestellt. Zwei Gabelzinkenkameras (Q1, Q2), eine Basiskamera (Q3) und zwei Mastkameras (Q4, Q5) werden eingesetzt. Die Basiskamera ist fest mit der Fahrerkabine verbunden und bewegt sich somit nicht durch Kippung und Anhebung des Mastes. Die Gabelzinkenkameras sind in hochintegrierten Gabelzinken der Firma Vetter verbaut. Diese ermöglichen die Verwendung verschiedener Kameras sowie eine Anpassung der Kamerawinkel über Adapterplatten.

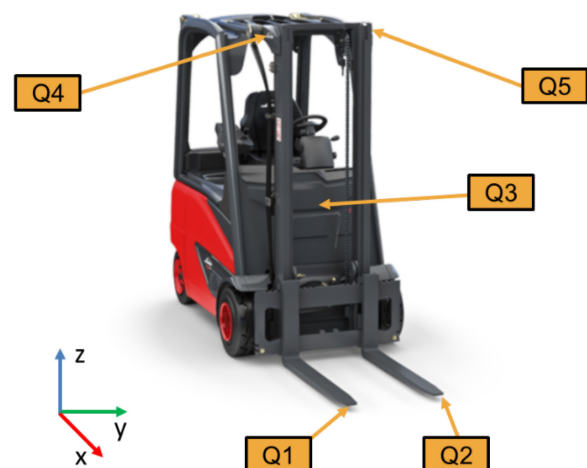


Abbildung 1. Kameraanbaupositionen am Demonstrator FFZ

### 2.2 VERWENDETE HARD- UND SOFTWARE

Bei den verwendeten Kameras handelt es sich um Intel RealSense D435i RGB-D-Kameras. Neben den RGB-Daten liefern diese auch einen Tiefenstream. Die daraus erstellten Punktwolken sollen die perspektivisch korrekte Transformation der aufgenommenen Bilder in das Fahrersichtfeld erleichtern.

Zur Visualisierung der Bildinformationen für den Fahrer des FFZ wird eine Microsoft HoloLens 2 eingesetzt. Diese setzt als Display durchsichtige holografische Linsen ein und weist ein Blickfeld von 52° (diagonal) auf. Zum Tracking von sechs Freiheitsgraden sind vier Kameras für sichtbares Licht, eine inertielle Messeinheit (Inertial

Measurement Unit, IMU) sowie ein Time-of-Flight-Tiefensensor integriert. Bedient wird das System über Hand-Gesten, Pupillenerkennung und Sprachbefehle.

Für die Umsetzung der Darstellungsberechnung wird die Unity-Engine und somit die Unity Entwicklungsumgebung verwendet. Hinsichtlich Stabilität und Leistung ist eine auf der HoloLens 2 laufende Applikation vorteilhaft und wird dementsprechend implementiert.

Zur Datenübertragung wird eine App der universellen Windows-Plattform auf einem externen PC umgesetzt. Diese App nutzt die Bibliotheken Microsoft.MixedReality.WebRTC\_UWP, Newtonsoft.Json, OpenCvSharp4 und Microsoft.NETCore.UniversalWindowsPlatform.

### 3 SOFTWARE

Die übergeordneten Softwarekomponenten sind in Abbildung 2 dargestellt. Die Szenenrekonstruktion läuft auf dem externen PC [JPO20, JP21]. Realsense-Kameras senden die RGB-D-Streams an den PC. Dieser verbindet die Streams der Kameras zu einer gemeinsamen Punktwolke und erstellt unter Verwendung der Position des Fahrers im Gabelstapler einen perspektivisch korrekten 2D-Video-Stream. Zur Datenübertragung läuft eine Universal Windows Platform (UWP)-App auf dem externen PC, der auch zur Szenenrekonstruktion genutzt wird. Diese App sendet den Videostream an die HoloLens 2 und empfängt die Positionsdaten von dieser. Auf der HoloLens läuft eine App zur Bestimmung der Position des Fahrers sowie zur Visualisierung des Ausgabebildes.

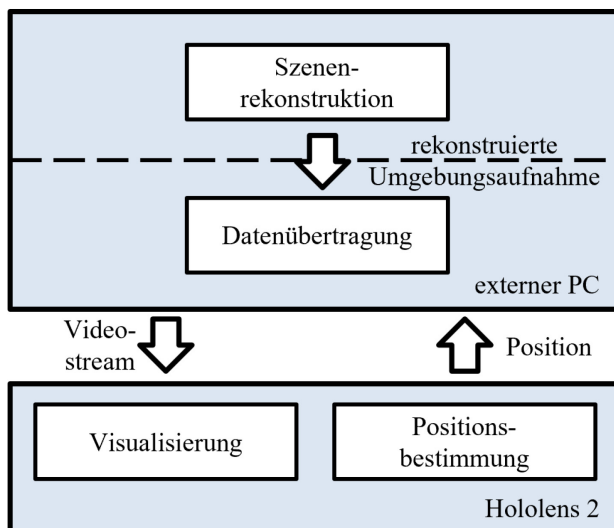


Abbildung 2. Übergeordnete Softwarekomponenten

#### 3.1 POSITIONSBESTIMMUNG

Die Positionsbestimmung der HoloLens 2 und somit des Fahrers erfolgt über QR-Codes. Microsoft stellt eine Bibliothek mit grundlegenden Funktionen zur Verfügung.

Diese ermöglicht die QR-Code Erkennung und stellt Informationen, wie die Größe und Position sowie die hinterlegten Daten zu den erkannten QR-Codes bereit. In jedem Frame wird geprüft, ob ein QR-Code im aufgenommenen Bild der internen HoloLens 2 Kamera vorliegt. Um die QR-Erkennung für die Positionierung der Hologramme nutzbar zu machen, wurde der QRCodeManager implementiert. Die Funktionsweise des QRCodeManagers sowie der Ablauf der Positionserkennung sind in Abbildung 3 dargestellt.

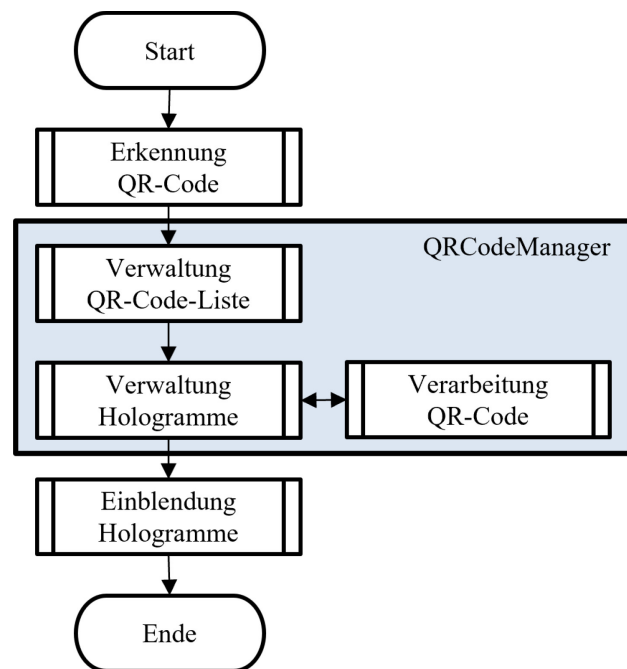


Abbildung 3. Ablauf der Positionserkennung

Erkannte QR-Codes werden vom QRCodeManager in einer Liste verwaltet. Zur Verwaltung anzuzeigender Hologramme wird eine Abfrage an die Verarbeitung der QR-Codes gestellt. Dort werden den erkannten QR-Codes hinterlegte Geometrien zugeordnet. Die Hologramme werden an den erkannten Positionen der QR-Codes instanziiert. Abschließend erfolgt die Einblendung der Hologramme auf der HoloLens 2, wie in Abbildung 4 dargestellt.



Abbildung 4. Einblendung der Hologramme

Zur Zuordnung der Hologramme zu den erkannten QR-Codes wird der hinterlegte Text in den QR-Codes ausgewertet. Zur Darstellung der Hologramme können entweder vordefinierte, bei der Kompilierung hinterlegte oder dynamisch aus primitiven Formen zusammengesetzte Geometrien ausgewählt werden. Bei den hinterlegten Geometrien handelt es sich um die Säulen, den Hubmast, das Dach und die Konsole inklusive Lenkrad des Staplers. Die primitiven Formen können Würfel, Kugeln oder Zylinder sein. Diese Formen können zur vereinfachten Darstellung von Hologrammen für aufzunehmende Lasten eingesetzt werden. Das Zentrum des jeweiligen primitiven Körpers befindet sich im Ursprung des QR-Code-Koordinatensystems, die voreingestellten Maße betragen 1 m x 1 m x 1 m und es ist keine Rotation vorhanden. Diese Werte können jedoch über den hinterlegten Text des QR-Codes manipuliert werden.

### 3.2 ÜBERTRAGUNG ZWISCHEN EXTERNEM PC UND AR-GERÄT

Der Ablauf der Datenübertragung zwischen dem externen PC und der Microsoft HoloLens 2 ist in Abbildung 5 dargestellt. Nach dem Start der Software erfolgt zunächst die Konfiguration der Datenübertragung. Hierfür wird eine Videoquelle ausgewählt und die Datenübertragung über den DataChannel optional ausgewählt. Als Videoquelle kann entweder eine direkt an den PC angeschlossene externe Kamera oder eine lokal gespeicherte bzw. durch Interprozesskommunikation (IPC) zur Verfügung gestellte Videodatei dienen.

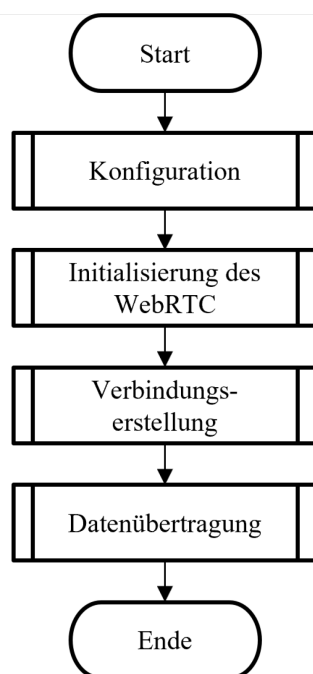


Abbildung 5. Aufbau der Datenübertragung auf externem PC

Die Initialisierung des WebRTC wird über ein Peer-Connection-Objekt realisiert. In der Konfiguration der

PeerConnection werden Informationen wie Interactive Connectivity Establishment (ICE)-Informationen und die Session Description Protocol (SDP)-Semantik definiert. Ein Transceiver wird erstellt und gemäß der Konfiguration mit dem Sender verbunden. Dieser bietet einen Kanal zur Datenübertragung, welcher auch bidirektional funktioniert. Das PeerConnection-Objekt lädt die Empfängerseite der Datenübertragung dazu ein, eine Verbindung herzustellen. Die Konfigurationsinformationen werden über Signaling an die Empfängerseite gesendet. Die Informationen werden im PeerConnection-Objekt der Empfängerseite gespeichert, dieses so konfiguriert und der entsprechende Transceiver erstellt. Die Konfigurationsinformationen werden anschließend zum externen PC zurückgesendet. Die Verbindung wird nach Erhalt der Antwort automatisch erstellt. Die Datenübertragung ermöglicht beiden Seiten das automatische Senden und Empfangen von Daten gemäß ihren jeweiligen Konfigurationen.

Für die Verwendung einer am PC angeschlossenen Kamera stellt WebRTC eine Application Programming Interface (API)-Methode bereit. Der Benutzer muss unter Windows ggf. die Berechtigung der Verwendung der Kamera durch die Software zustimmen. Beim Anschluss mehrerer Kameras kann eine als Videoquelle aus einer Liste der verfügbaren Geräte ausgewählt werden.

Die Übertragung einer lokalen Videodatei, die in Echtzeit abgespielt wird, ist in drei Schritte unterteilt. Im ersten Schritt wird die lokale Videodatei ausgewählt. Im zweiten Schritt werden die Videoframes aus der Videodatei Frame für Frame gelesen. Der dritte Schritt besteht darin, ein Videoframe zu erstellen, der per WebRTC gesendet werden kann, da die im vorherigen Schritt extrahierten Daten nicht direkt gesendet werden können. Die Daten werden in einer Struktur übergeben, die alle erforderlichen Informationen für einen Videoframe enthält, wie z. B. die Höhe und Breite des Videos sowie eine Steuernachricht.

Im vorliegenden Anwendungsfall soll in der Regel eine vorverarbeitete Umgebungsaufnahme verarbeitet werden und nicht direkt die Aufnahme einer externen Kamera. Wenn die Bereitstellung der Umgebungsaufnahme über eine separate Software erfolgt, bietet sich eine IPC auf Betriebssystemebene an. Eine praktische Methode der IPC ist die Verwendung von Sockets.

Der DataChannel hat die Fähigkeit, beliebige Daten zu senden und zu empfangen. Da die Daten, die von einem DataChannel gesendet werden können, vielfältig sind, ist auch die Vor- und Nachverarbeitung der Daten unterschiedlich. Daher werden häufig vorkommende Datentypen im Programm voreingestellt. Die Datentypen können vom Benutzer in der Benutzeroberfläche ausgewählt werden. Damit können der Sender und Empfänger die entsprechende Verarbeitung vorbereiten. Abhängig von den Anforderungen des Anwendungsszenarios kann der Datentyp

die Steuerungsinformationen, Bilder, Videos, Punktwolken oder Dateien umfassen.

Zum Sicherstellen der Integrität der Daten kann ein Header hinzugefügt werden. Der Header besteht aus einem Zeitstempel, einem Index und Hash-Wert. Ein Zeitstempel kann verwendet werden, um die Größe der Verzögerung abzuschätzen. Der Index der Daten kann zur Überwachung auf Verlust, Überlastung oder Duplizierung verwendet werden. Der Hash-Wert kann verwendet werden, um anzuzeigen, ob eine Beschädigung der Daten aufgetreten ist. Um eine flexible Kommunikation zu ermöglichen, kann die ausgewählte Datenstruktur in den zu übertragenden Daten enthalten sein. Dadurch kann dem Empfänger mitgeteilt werden, welche Methode zum Parsen der Daten verwendet werden soll.

### 3.3 VISUALISIERUNG AUF DEM AR-GERÄT

Ziel des Projekts „ViSIER“ ist die Überlagerung von Sichteinschränkungen durch die dahinter aufgenommene Umgebung. Der Fahrer des FFZ soll die Einblendungen nur auf den tatsächlichen Sichteinschränkungen sehen. Im Rest des Sichtfeldes soll die reale Umgebung durch das See-Through-Display der HoloLens 2 wahrgenommen werden. Hierfür müssen die Umrisse der Sichteinschränkungen im aktuellen Sichtfeld des Anwenders bekannt sein. Die positionierten dreidimensionalen Geometrien dienen nur dazu, auch bei sich verändernder Position des Anwenders im Gabelstapler stets den Umriss der Sichteinschränkungen, ohne umfangreiche semantische Bilderkennung bestimmen zu können. Die Umrisse bilden eine Schablone, die über einen Videostream gelegt werden kann. So werden die Bilddaten nicht auf dem gesamten Bildschirm der AR-Brille abgespielt, sondern nur in den notwendigen Bereichen. Abbildung 6 verdeutlicht das Konzept der sich aus Umrisse ergebenden Schablonen.

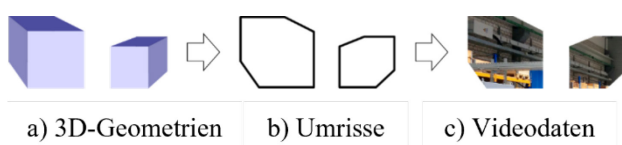


Abbildung 6. Schablonenprinzip

Die in Unity standardmäßig integrierten Shader sind für die Darstellung nach dem Schablonenprinzip ungeeignet. Die Shader dienen in der Regel der dreidimensionalen Berechnung mit Schatten und Licht-Reflektion oder Effekten wie Partikeln. Hierfür werden Texturen statisch auf die Flächen der Körper gelegt. Für den vorliegenden Anwendungsfall eignet sich ein Vertex-/Fragment-Shader. Der Vertex-Teil des Shaders dient der dreidimensionalen Beeinflussung der Polygone sowie der Transformation der globalen Polygonpositionen in das Clip-Space-Koordinatensystem, ein System mit Ursprung im Blickfeld des Betrachters. Im Fragment-Teil wird definiert, in welcher Farbe und Transparenz die Pixel für den Bildschirm ausgegeben werden.

## 4 GENAUIGKEIT UND ECHTZEITFÄHIGKEIT

### 4.1 GENAUIGKEIT DER EINBLENDUNG

Die Genauigkeit der Einblendung ist von entscheidender Bedeutung, um die Sichteinschränkungen für den Fahrer zu überblenden. Geometrisch ungenaue Einblendungen irritieren den Fahrer, anstatt die Sicht zu verbessern. Um die Genauigkeit der Einblendung zu ermitteln, wurde ein Testaufbau außerhalb des FFZ aufgebaut. QR-Codes mit Geometrieinformationen werden auf einfachen, quaderförmigen Gegenständen platziert. Die Gegenstände werden mit Maßstäben versehen. Das Ablesen des visuellen Maßstabs zur Bestimmung der Abweichung von Einblendung vom realen Objekt wird mit zunehmender Entfernung ungenau und zudem durch das See-Through-Display der HoloLens 2 erschwert. Abbildung 7 zeigt zwei Aufnahmen im Testaufbau.

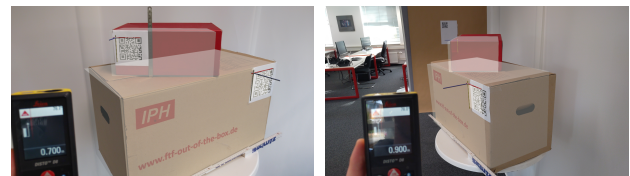


Abbildung 7. Untersuchung der Genauigkeit

Aufgrund des ungenauen Ablesens des Maßstabs erfolgt eine qualitative Bewertung. Die Hologramme werden nicht exakt deckungsgleich auf den Objekten platziert. Bei Betrachtung aus verschiedenen Winkeln verändern sich Position und Rotation der Hologramme relativ zum Objekt. Zwischen den Einblendungen von linkem und rechten Auge besteht eine Abweichung. Steigt die Entfernung zum Objekt über einen Meter nimmt die Positionsabweichung mit der Zeit zu. Erfolgt nach einer Bewegung der HoloLens aus der Szene heraus kein erneutes Tracking des QR-Codes verstärkt sich dieser Effekt.

### 4.2 ROBUSTHEIT

Die zunehmende Abweichung der Einblendung vom realen Objekt bei Verlust des Trackings der QR-Codes verdeutlicht die Notwendigkeit, die Robustheit des Trackings zu untersuchen. Die Untersuchung erfolgt zunächst ebenfalls in einer Testumgebung außerhalb des FFZ. QR-Codes mit einer Kantenlänge von 4 bis 15 cm werden an einer Pinnwand befestigt. Die QR-Codes werden aus verschiedenen Winkeln mit der HoloLens betrachtet. Die maximale Erkennungsdistanz, aus der die QR-Codes reproduzierbar erkannt und somit Einblendungen vorgenommen werden, wird bestimmt. Bei einer Seitenlänge der QR-Codes von 4 cm ist unabhängig vom Betrachtungswinkel keine Erkennung möglich. Bei Winkeln größer als 45° ist unabhängig von der Seitenlänge keine Erkennung möglich. Jeweils fünf Messungen wurden durchgeführt. Die Messergebnisse sind in Abbildung 8 dargestellt. Dem Diagramm sind die Mittelwerte, Minimum und Maximum für die senkrechte



Betrachtung ( $0^\circ$ ,  $30^\circ$  und  $45^\circ$ ) zu entnehmen. Der Kasten umfasst jeweils den Bereich vom ersten bis dritten Quartil.

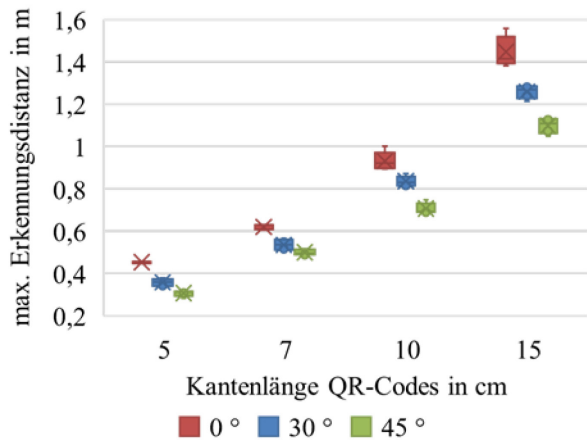


Abbildung 8. Maximale Erkennungsdistanz der QR-Codes

Die Messungen im Testumfeld suggerieren, dass eine robuste Erkennung im Stapler bei einer QR-Code Kantenlänge von 10 cm möglich ist, da die Distanz des Fahrers zu den Säulen kleiner als 50 cm ist. QR-Codes dieser Größe können an den Säulen des Staplers angebracht werden, ohne zusätzliche Sichteinschränkungen darzustellen.

Tests im FFZ zeigen, dass die Beleuchtung im Stapler häufig auch bei geringen Distanzen zum QR-Code nicht ausreicht, um diesen zu erkennen. Dementsprechend wird eine externe Beleuchtungsquelle eingesetzt, um die Erkennung sicherzustellen. Zudem tritt bei der Fahrt eine verzögerte und weniger robuste Erkennung der QR-Codes auf. Dies hat zur Folge, dass der Fahrer mit dem Stapler durch die Hologramme hindurchfährt und diese erst bei erneuter Erkennung der QR-Codes hinterher springen.

### 4.3 ECHTZEITFÄHIGKEIT

Neben der geometrischen Genauigkeit der Einblendung, muss diese auch zeitlich passend sein, um den Fahrer nicht zu irritieren, sondern zu unterstützen. Zur Untersuchung der Echtzeitfähigkeit wird eine Latenzmessung der Datenübertragung durchgeführt. Hierfür wird ein Alienware-m15r3-Laptop verwendet. Während der Versuche laufen keine unnötigen Prozesse auf dem Rechner und der Hololens. Der Rechner verfügt über ausreichend Rechenleistung und Speicherkapazität, um die Videoframes in einer sehr kurzen Zeitspanne zu öffnen und die Pixel jedes Frames zu traversieren. Demgegenüber verfügt die Hololens 2 über eine begrenzte Rechen- und Speicherleistung. Wenn das Video eine bestimmte Größe erreicht oder der Algorithmus eine bestimmte Komplexität erreicht, wird die Ausführung des Programms verzögert.

Zur Messung der Latenz der Videoübertragung werden Versuche mit einer Webcam und einer Intel RealSense D435i durchgeführt. Die Auflösung der Webcam beträgt  $1280 \times 720$  px, die Auflösung der RealSense

$1920 \times 1080$  px. Um den Einfluss des Übertragungsmediums auf die Übertragungsleistung zu überprüfen, wird die Videoübertragung über WLAN, ein USB-Kabel und ein Ethernet Kabel durchgeführt. Die WLAN-Umgebung wird von einem WLAN-Router mit einer maximalen Übertragungsrate von 600 Mbps erstellt. Beim USB-Kabel handelt es sich um eine USB C-A Verbindung. Diese arbeitet nach dem USB 3.1 Standard mit einer maximalen Übertragungsrate von 5 Gbps. Da die Hololens nur über einen USB C-Stecker verfügt, wird zur Übertragung über Ethernet ein USB zu Ethernet-Adapter verwendet. Die Versuchsergebnisse der Versuche mit der RealSense sind in Abbildung 9 dargestellt.

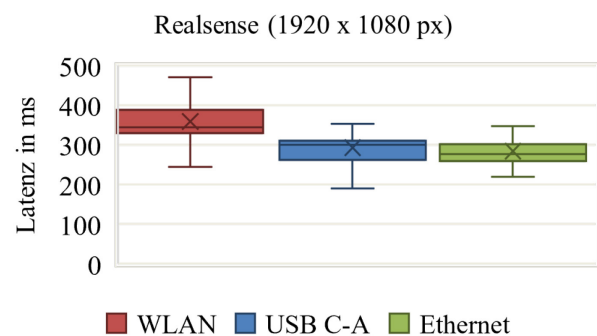


Abbildung 9. Latenz bei Übertragung des RealSense Videos ( $1920 \times 1080$  px)

Die Latenz der Videoübertragung der RealSense ist für die Übertragung über Ethernet am geringsten und unterliegt den geringsten Schwankungen. Der Mittelwert liegt bei 284,6 ms. Die Übertragung über USB liefert ähnliche Ergebnisse. Der Mittelwert liegt bei 293,4 ms. Die Werte der Übertragung über WLAN liegen deutlich über denen der kabelgebundenen Übertragung. Der Mittelwert liegt bei 359,3 ms. Unabhängig vom Übertragungsmedium sind die Werte der Latenz zu hoch, um eine zeitlich passende Einblendung auf der Hololens 2 zu erreichen. Dementsprechend wurde eine weitere Versuchsreihe mit einer Webcam mit geringerer Auflösung ( $1280 \times 720$  px) durchgeführt. Die Versuchsergebnisse der Versuche mit der Webcam sind in Abbildung 10 dargestellt.

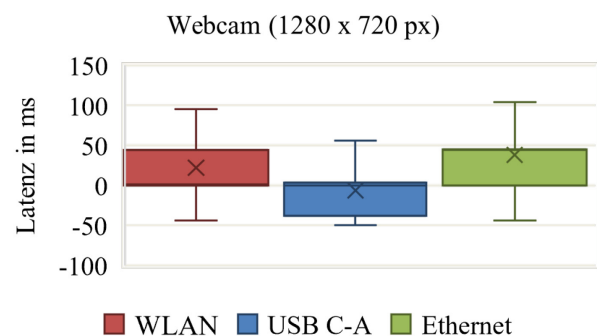


Abbildung 10. Latenz bei Übertragung des RealSense Videos ( $1280 \times 720$  px)

Die Latenz der Videoübertragung der Webcam ist für die Übertragung über USB am geringsten und unterliegt den geringsten Schwankungen. Der Mittelwert liegt bei -6,5 ms. Die Übertragung über Ethernet und WLAN liefern ähnliche Ergebnisse. Der Mittelwert liegt bei der WLAN-Übertragung bei 22,1 ms und bei der Ethernet Übertragung bei 38,1 ms. Die Schwankungen sind gegenüber der Übertragung über USB deutlich erhöht. Die negativen Werte bei der Latenzmessung sind in der Messmethode begründet. Mit der Kamera der Hololens 2 wird der Bildschirm des Rechners aufgenommen, während das empfangene Video als Hologramm angezeigt wird. Der Rechner überträgt ein Video einer Stoppuhr, welches von der Realsense oder Webcam aufgenommen wird. Gleichzeitig spielt der Rechner das übertragene Video ab. Die negativen Werte entstehen, wenn der Rechner länger benötigt, das übertragene Video darzustellen, als die Übertragungszeit inklusive der Zeit, welche die Hololens zum Einblenden des Videos als Hologramm benötigt. Die Ergebnisse zeigen, dass die reine Übertragungszeit über WebRTC bei ausreichend kleiner Auflösung des Aufnahmeapparates den Anforderungen an die Echtzeitfähigkeit genügt.

## 5 AUSBLICK

Im nächsten Schritt des Projekts werden die hier vorgestellten Softwarekomponenten an die bestehende Szenenrekonstruktion angebunden. Hierfür wird eine IPC über Sockets verwendet. Zur Umgebungsaufnahme des FFZ werden Realsense-Kameras verwendet. Diese liefern 30 Fps bei einer Auflösung von 640 x 480 px. Der verwendete PC verfügt über einen Intel Core i7-9700 mit bis zu 4.7 GHz und eine Nvidia Quadro RTX4000. Die Berechnungszeit zur Szenenrekonstruktion wird dennoch kritischer eingeschätzt als die Übertragungszeit zur Hololens 2. Um die Robustheit der Positionserkennung zu erhöhen, wird ein externes, kamerabasiertes Head Tracking-System eingesetzt. Dieses ist weniger stör anfällig als die QR-Code-Erkennung über die integrierte Kamera der Hololens.

Parallel wird ein Ansatz verfolgt, bei dem keine eigenständige App auf der Hololens 2 läuft. Stattdessen finden sämtliche Berechnungen auf dem Rechner statt. Die Hololens 2 dient als reines Anzeigemedium. Dies wird über das in Unity integrierte Holographic Remoting umgesetzt. In der Unity-Umgebung können die Realsense-Kameras ebenso wie das Staplermodell direkt eingebunden werden. Der entwickelte Shader kann übernommen werden. Erste Untersuchungen zeigen, dass die Latenz und Framerate bei dieser Umsetzung ein immersives Erlebnis für den Fahrer ermöglichen. Wesentliche Nachteile dieser Methode sind, dass die USB-Verbindung über Holographic Remoting instabil und parallel kein Zugriff auf die internen Kameras der Hololens 2 möglich ist.

## 6 ZUSAMMENFASSUNG

Der vorliegende Artikel beschreibt die Visualisierung von Echtzeitvideos auf einer Microsoft Hololens 2 für das übergeordnete Forschungsvorhaben „Virtuelle Sichtverbesserung und intuitive Interaktion durch Erweiterte Realität an Flurförderzeugen (ViSIER)“. Die entwickelten Softwarekomponenten ermöglichen es, Videodaten in Echtzeit zu übertragen und deckungsgleich darzustellen. Dabei werden nur die sichteinschränkenden Bereiche überlagert, indem ein Schablonenprinzip angewendet wird. Dies wird durch einen entwickelten Shader in Unity ermöglicht. Die Echtzeitfähigkeit der Datenübertragung wurde in Messungen untersucht und kann mit einer Latenz von im Mittel 5,7 ms bei einer Übertragung über USB und einer Auflösung von 1280 x 720 px als echtzeitfähig eingestuft werden. Zudem wurde die Genauigkeit der Einblendung auf der Hololens 2 untersucht. Diese nimmt bei zunehmender Entfernung zum Objekt und keinem erneuten Tracking zu. Insbesondere nach einer nahezu 180 Grad-Kopfdrehung des Fahrers während der Rückwärtsfahrt, könnte anschließend eine größere Abweichung vorliegen. Die Robustheit des Trackings über QR-Codes führt insbesondere bei der Fahrt zu fehlerhaften Positionierungen.

## 7 FÖRDERHINWEIS

Diese Veröffentlichung wurde im Rahmen des IGF-Vorhabens 20158 N „Virtuelle Sichtverbesserung und intuitive Interaktion durch Erweiterte Realität an Flurförderzeugen (ViSIER)“ der Forschungsgemeinschaft Intralogistik/ Fördertechnik und Logistiksysteme (IFL) e. V. über die Arbeitsgemeinschaft industrieller Forschungsvereinigungen „Otto von Guericke“ e. V. (AiF) im Rahmen des Programms zur Förderung der Industriellen Gemeinschaftsforschung (IGF) vom Bundesministerium für Wirtschaft und Energie (BMWi) aufgrund eines Beschlusses des Deutschen Bundestages gefördert

## LITERATUR

- [Deu20] Deutsche Gesetzliche Unfallversicherung (DGUV): Arbeitsunfallgeschehen 2019, Statistik, Berlin, 2020.
- [Fer13] Ferreira, P. et al.: Augmented Reality driving supported by Vehicular Ad Hoc Networking. In: 2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Adelaide, Australien, 2013, S. 253-254.
- [Fra19] Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung IOSB: Tote Winkel ade: Virtual Reality macht das Cockpit transparent. <https://www.iosb.fraunhofer.de/de/presse/presseinformationen/2019/transparentes-cockpit.html>,
- [Fri19] Fritzmeier CABS: Augmented Reality für die Agrartechnik – Fritzmeier CABS präsentiert neues Sicherheitssystem, 2019. <https://www.landtechnikmagazin.de/sonstige-Landtechnik-Bild-Intelligent-Security-View-ISV-Fritzmeier-CABS-AR-basiertes-Fahrerassistenzsystem-Sicherheit-erhoehen-38214-8498.php>
- [GOF12] Gomes, P.; Olaverri-Monreal, C.; Ferreira, M.: Making Vehicles Transparent Through V2V Video Streaming. In: Transactions on Intelligent Transportation Systems, IEEE, 13. Bd. (2012), H. 2, S. 930-938.
- [GVF12] Gomes, P.; Vieira, F.; Ferreira, M.: The See-Through System: From implementation to test-drive. In: 2012 IEEE Vehicular Networking Conference (VNC), Seoul, Südkorea, 2012, S. 40-47.
- [JPO20] Jütte, L.; Poschke, A.; Overmeyer, L.: Kompensation von Sicht Einschränkungen an Flurförderzeugen. In: Tagungsband zum 16. Fachkolloquium der Wissenschaftlichen Gesellschaft für Technische Logistik e. V. (WGTL), 2020, S.93-100. ISBN: 987-3-00-066746-6.
- [JP21] Jütte, L.; Poschke, A.: Einschränkungsfreie Sicht im Gabelstapler. In: phi – Produktionstechnik Hannover informiert, Newsletter Nr. 31 / Juni 2021, ISSN: 2198-1922.
- [Kle18] Kleinert, S.: Einsatz skalenübergreifender Time-of-Flight-Aufnahmen für Assistenzsysteme in Flurförderzeugen, Dissertation. TEWISS Verlag, Garbsen, 2018. – ISBN 978-3-95900-233-2
- [OM13] Overmeyer, L.; Mänken, F.: Intelligente Schnittstellen in wandlungsfähigen Lieferketten – ISI-WALK. Abschlussbericht. PZH-Verlag, Garbsen, 2013. – ISBN 978-394-458-616-8
- [Wil15] Willibald, A. G. Hrsg.: Einsatz der Augmented-Reality-Technologie zur Unterstützung des Fahrers von Flurförderzeugen. Forschungsbericht zu dem IGFVorhaben. Technische Universität München Institut für Maschinen- und Fahrzeugtechnik. Lehrstuhl für Förder-technik und Materialfluss, München, 2015. – ISBN 978-3-941702-53-0
- [Yos08] Yoshida, T. et al.: Transparent Cockpit: Visual Assistance System for Vehicle Using Retro-reflective Projection Technology. In: 2008 IEEE Virtual Reality Conference, Reno, USA, 2008, S. 185-188.
- [Zei12] Zeit, F.; Butz, S.; Schwarzmann, K.; Stöckl, K.: Handbuch für Gabelstaplerfahrer, Berufsgenossenschaft Handel und Warendistribution BGHW, Bonn, 2012.
- 
- Alexander Poschke, M. Sc.**, (\*1992) Project Engineer at Institut für Integrierte Produktion Hannover (IPH) gGmbH. Between 2011 and 2018, he studied mechanical engineering at the Leibniz Universität Hannover. The focus of his work is the Augmented reality based overlay of visibility constraints on industrial trucks.
- Address: Institut für Integrierte Produktion Hannover gGmbH, Hollerithallee 6, 30419 Hannover, Germany, Phone: +49 511 27976-229, Fax: +49 511 27976-888, E-Mail: [poschke@iph-hannover.de](mailto:poschke@iph-hannover.de)
- Lukas Jütte, M. Sc.**, (\*1993) studied mechanical engineering at the Leibniz Universität Hannover. Since October 2019 he works as a research associate at the Institute of Transport und Automation Technology, Leibniz Universität Hannover. The focus of his work is the scene reconstruction for visibility obstructions on industrial trucks. Since January 2021 Lukas Jütte is the team leader of the department of transport technology.
- Address: Institute of Transport and Automation Technology, Leibniz Universität Hannover, An der Universität 2,



30823 Garbsen, Germany Phone: +49 511 762-18173,  
Fax: +49 511 762-4007, E-Mail: lukas.juette@ita.uni-han-  
nover.de

**Dr.-Ing. Benjamin Küster** (\*1988) studied industrial engineering at the Leibniz Universität Hannover. From November 2014 to August 2017, he worked as a project engineer at the IPH - Institut für Integrierte Produktion Hannover gGmbH in the department of production automation. In 2020, he received his doctorate with a thesis on “Automated quality assessment of 8D reports by methods of computational linguistics”. Since September 2017 Benjamin Küster is leader of the department of production automation.

**Dr.-Ing. Malte Stonis** (\*1979) studied mechanical engineering at the Leibniz Universität Hannover with a focus on vehicle systems and biomedical engineering. He has been working at IPH - Institut für Integrierte Produktion Hannover gGmbH since 2006, initially as a project engineer in the field of process technology and from 2008 as head of department. In 2011, he received his doctorate with a thesis on "Multidirectional forging of flat aluminium parts". Since September 2016 Malte Stonis is coordinating managing director of the IPH.

**Prof. Dr.-Ing. Ludger Overmeyer** (\*1964) studied electrical engineering at the University of Hannover between 1984 and 1991. In 1996 he finished his doctorate in mechanical engineering at the University of Hannover. From 1997 to 2001 he worked as project manager, division manager and head of research and development at Mühlbauer AG in Roding. Since 2001 Ludger Overmeyer is Professor of Transport and Automation Technology of Leibniz Universität Hannover.