

## PUMA: Web-AR-Techniken als Ergänzung des Physikunterrichts

Stefan Kraus, Thomas Trefzger

Lehrstuhl für Physik und ihre Didaktik, Julius-Maximilians-Universität Würzburg  
stefan.kraus@physik.uni-wuerzburg.de

### Kurzfassung

Augmented Reality (AR) als Verschmelzung von virtuellen Elementen mit der Realität ist inzwischen bekannt. Doch wieso sollte diese in der Schule Einzug halten? Eine flexible Augmented Reality-Simulation, die nicht starre Vorgänge, sondern die Physik hinter dem Versuch simuliert und dabei noch nicht einmal eine Installation auf dem Endgerät voraussetzt, erscheint hier als ideale Ergänzung zu konventionellen Experimenten. Der Beitrag gibt einen Überblick über die Möglichkeiten des Einsatzes von WebAR und die dahinterstehenden Techniken bis hin zu einem groben Ausblick auf das geplante Forschungsvorhaben.

### 1. Simulationen haben Grenzen

Reale Experimente haben Vorteile, die sich durch eine Simulation nicht ersetzen lassen: So bietet ein Realexperiment immer eine haptische Komponente, die feinmotorische Fähigkeiten der Experimentierenden fördert und fordert. Zudem können sich die Schülerinnen und Schüler auf den Versuch als solchen konzentrieren, ohne durch Herausforderungen in der Bedienung der Simulationssoftware abgelenkt zu werden. Zudem kann keine Simulation alle Feinheiten der Realität komplett abbilden. Im Realexperiment erleben die Lernenden die Physik, wie sie wirklich ist.

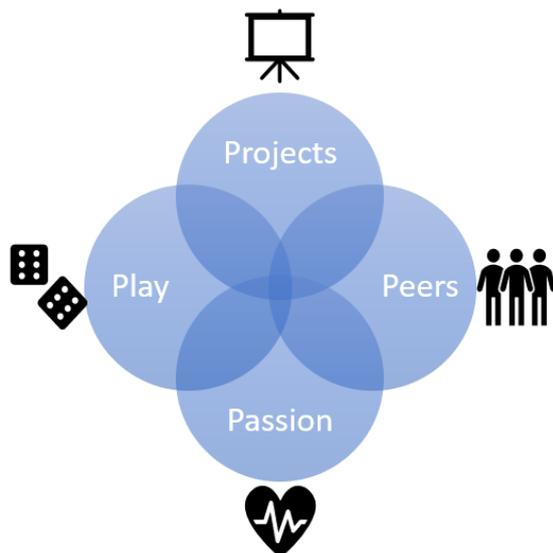


Abb.1: Give P's a chance [1]

Angelehnt an das Modell „Give P's a chance“ von Mitchel Resnick lebt der Physikunterricht von spielerischen Experimenten, Projekten, der Arbeit im Team und den Emotionen, die damit verbunden sind [1]. Es wird viel dafür getan, den Physik-Unterricht in diesen Bereichen zu stärken. Zahlreiche

Experimente sind allerdings dem Präsenzunterricht vorbehalten. Häufig wird teures Equipment benötigt, das den Schülerinnen und Schülern nicht nach Hause mitgegeben werden kann. Auch ist das Material oft nur in begrenztem Maß vorhanden. Dies führt zur Verwendung von Simulationen, die allerdings meist sehr stark auf einen bestimmten Versuch fokussiert sind und kaum Raum für kreatives Experimentieren bieten.

### 2. Web-AR: AR mit Webtechniken



Abb.2: Web-AR als Kombination von Webtechniken und Augmented Reality

Native AR-Applikationen erfordern die Installation auf einem mobilen Endgerät. Dies ermöglicht der dezidierten Software eine qualitativ hochwertige Nutzererfahrung, jedoch stellt die Installation eine Hürde im Schulalltag da. So haben viele Schülerinnen und Schüler keine Installationsrechte auf ihren Smartphones oder es fehlt schlicht an ausreichendem Speicherplatz. WebAR-Anwendungen hingegen laufen im Browser und werden als normale Website geladen. Eine Installation ist nicht notwendig. Dies vereinfacht den Schuleinsatz wesentlich.

Ein Beispiel kann über den QR-Code auf der nächsten Seite abgerufen werden. Anschließend muss die Kamera auf den „A“-Marker gerichtet werden (optimiert für das Querformat).

### 3. Die Applikation

Ziel ist die Entwicklung einer WebAR-Anwendung zur Simulation von Optik-Versuchen für den schulischen Physikunterricht. Dabei legen die

Schülerinnen und Schüler mehrere ausgedruckte Marker auf den Tisch, worüber Lichtquellen und Gegenstände wie Linsen und Spiegel eingeblendet werden. Die Anordnung der Marker ist völlig frei wählbar. Der Strahlenverlauf wird physikalisch modelliert und so realistisch wie möglich wiedergegeben. Dabei kommt neben der darstellerischen auch eine haptische Komponente zum Tragen, die das Experimentiererlebnis verstärken soll. Zudem ist es möglich, Parameter der beteiligten Objekte wie die Wellenlänge oder Brennweite während des Versuchs zu ändern und dabei in physikalische Grenzbereiche vorzustoßen.

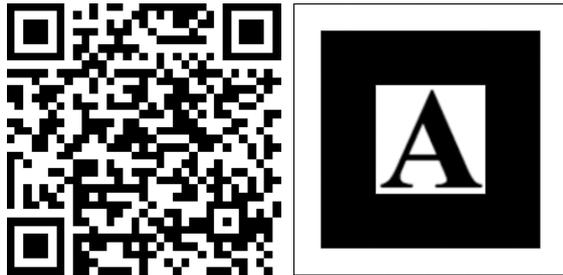


Abb.3: QR-Code und Target für eine statische Grafik-Demonstration der THREE.js-Komponenten

#### 4. Eingesetzte Techniken

Die Entwicklung von Augmented Reality-Anwendungen mittels Web-Techniken erlebt aktuell noch ein Nischendasein, erfreut sich allerdings zunehmender Bedeutung. Die eingesetzten Frameworks bauen auf die WebXR-API auf, die inzwischen von allen gängigen Mobile-Browsern unterstützt wird. Programmiert wird mittels HTML-basierten Bibliotheken wie A-Frame [3] und JavaScript-Bibliotheken wie AR.js [8], das THREE.js [4] zur 3D-Darstellung und JSARToolkit5 [5] zur Umsetzung der Augmentierung einsetzt. Im Hintergrund stehen hier die bekannten Programmierschnittstellen WebGL [6] und WebRTC [7].

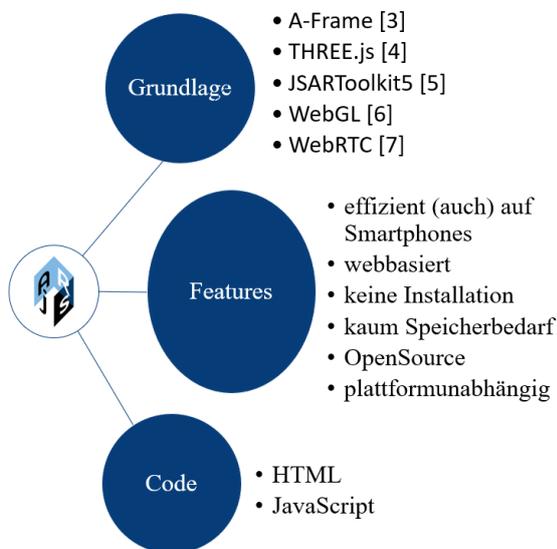


Abb.4: Web-AR-Frameworks, -Features & -Sprachen

Der Einsatz dieser OpenSource-Bibliotheken ermöglicht eine performante Darstellung der augmentierten Inhalte (auch) auf Smartphones innerhalb von Browsern (webbasiert). Dadurch ist keine Installation notwendig, es wird nur wenig Speicher genutzt und die Applikation ist in höchstem Maße plattformunabhängig.

#### 5. Forschungsinteresse

Die Forschungsfragen sind noch nicht abschließend formuliert. Jedoch bietet es sich an, die Praxistauglichkeit von WebAR-Anwendungen, insbesondere den Einsatz auf verschiedensten Plattformen zu testen. Daneben stellen sich auch Fragen zum fachdidaktischen Einsatz einer solchen Applikation:

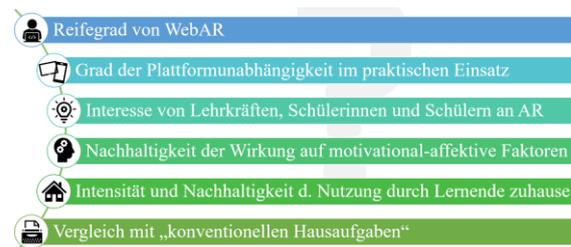


Abb.5: Mögliche Forschungsgebiete

#### 6. Einsatzzweck

Zweck der WebAR-Applikation soll es keinesfalls sein, reale Experimente zu ersetzen. Vielmehr soll sie eine Ergänzung darstellen und dabei ihre Vorteile primär in der Vor- und Nachbereitung, aber auch im Unterricht ausspielen. Sowohl finanziell als auch in der Verfügbarkeit des Versuchs sollen hierdurch in vielfältiger Weise Barrieren abgebaut werden. Experimente, die sonst nur in einem relativ streng vorgegebenen Rahmen im Unterricht stattfinden, werden nun zu den Schülerinnen und Schülern nach Hause gebracht. Gefahren von Realexperimenten auch über die Optik hinaus (z. B. radioaktive Strahlenbelastung) lassen sich durch Simulationen vermeiden. Der eine oder andere Versuch wird somit durchführbar. Da sich Parameter im Gegensatz zum Realexperiment beliebig wählen lassen, können physikalische Grenzbereiche ausgelotet werden und z. B. Brennweiten während des Versuchs verändert werden. Auch sonst schwer umsetzbare Versuche wie eine Luftlinse unter Wasser lässt sich problemlos darstellen. Der Einsatz könnte auch im Unterricht denkbar sein, hier aber nur, wenn sich ein klarer Vorteil gegenüber dem Realexperiment ergibt.

#### 7. Weiteres Vorgehen

Nach der aktuellen Konzeptionsphase soll die erste lauffähige Beta-Version im Herbst 2022 zur Diskussion stehen. Mit dieser soll im Laufe des Schuljahrs eine Prästudie mit Befragungen auf Fortbildungen, Lehrveranstaltungen der Universität und Tagungen durchgeführt werden, die zur Konzeption der Hauptstudie im darauffolgenden

Schuljahr führt. Hier ist ein größer angelegter Einsatz in Schulklassen unter der Nutzung von Online-Auswertungstechniken geplant.

## 8. Projekt „PUMA“

Diese Arbeit ist Teil des Projekts „PUMA“ (Physikunterricht mit Augmentierung) des Lehrstuhls für Physik und ihre Didaktik an der Universität Würzburg. Hier entstehen weitere Projekte zur Elektrizitätslehre inkl. Magnetismus. Weitere Informationen finden Sie auf unserer Website. [2]

## 9. Literatur

- [1] Resnick, Mitchel (2014): Give P's a chance: Projects, peers, passion, play, <https://web.media.mit.edu/~mres/papers/constructionism-2014.pdf>
- [2] Website des Projekts „Puma“ am Lehrstuhl für Physik und Ihre Didaktik der Universität Würzburg: <https://www.physik.uni-wuerzburg.de/pid/physik-didaktik/augmented-reality/> (Stand 3/2022)
- [3] Dokumentation des Frameworks A-Frame: <https://aframe.io/docs/1.3.0/introduction/> (Stand 3/2022)
- [4] Dokumentation des Frameworks THREE.js: <https://threejs.org/> (Stand 3/2022)
- [5] Repository des Frameworks jsARToolkit5: <https://github.com/artoolkitx/jsartoolkit5> (Stand 3/2022)
- [6] Mozilla: Dokumentation der JavaScript-Programmierschnittstelle WebGL: [https://developer.mozilla.org/de/docs/Web/API/WebGL\\_API](https://developer.mozilla.org/de/docs/Web/API/WebGL_API) (Stand 3/2022)
- [7] Google Developers: Dokumentation von WebRTC: <https://webrtc.org/getting-started/overview> (Stand 3/2022)
- [8] Etienne, Jerome: Dokumentation der Frameworks AR.js: <https://ar-js-org.github.io/AR.js-Docs/> (Stand 3/2022)