

# NEUERUNGEN IN ARKIT

VON MOTION CAPTURE BIS ZUM „RAUMSCANNER“

MATTHIAS WEBER - DEVELICIOUS STUDIOS WEBER - 2021



# ÜBER MICH

Studium der Naturwissenschaftlichen Informatik an der Uni Bielefeld

Als wissenschaftlicher Mitarbeiter an der Uni Lübeck, TU Freiberg und dem Fraunhofer FKIE gearbeitet

Dabei viel mit VR in Kontakt gekommen

Währenddessen interaktive mediale Installationen mit Künstlern

Freiberuflicher Entwickler seit 2010, entwickelt iOS (und in Grenzen auch Android) Apps und C++/Qt Software unter Linux&Windows, Drohnen, MeinMagenta App und Baker Hughes Mentor

Weiterhin großes Interesse an VR/AR



# WAS IST AUGMENTED REALITY

Paul Milgrem et al. 1994: Realitäts-Virtualitäts-Kontinuum, von realer Umgebung, über erweiterter Realität (Augmented Reality, AR) und erweiterter Virtualität zu virtueller Realität (VR)

Computergestützte Erweiterung der Realität, kann alle Sinneswahrnehmungen betreffen, ist aber in der Regel nur auf Darstellung begrenzt



# NEUERUNGEN

„Weltwissen“, Bilddetektion, Objektdetektion, Meshdetektion, Multiuser Experience

Neues Fileformat: USDZ, unkomprimierter Container für 3D Daten, Reality Converter, Scene Capture

People Occlusion

Global Anchors, App Clips

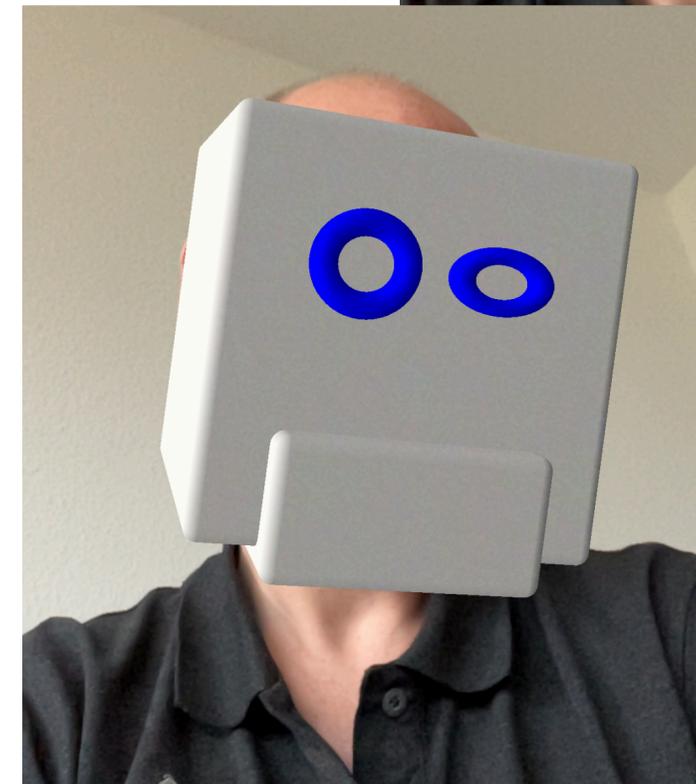
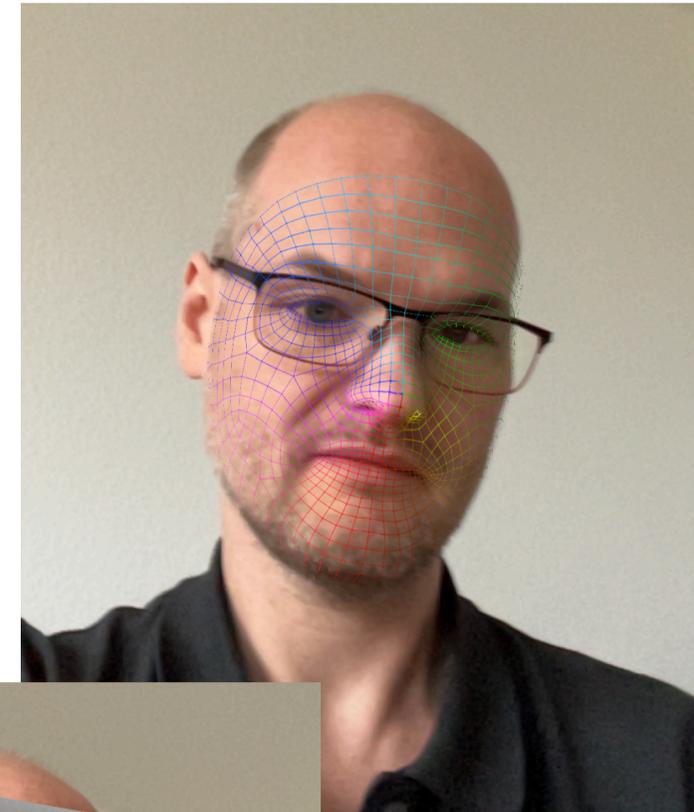
Facial Capture / Motion Capture

Scene Map / Scene Depth

# FACE CAPTURE

Immer abfragen, ob das Device die Konfiguration unterstützt

```
guard ARFaceTrackingConfiguration.isSupported else { return }
let configuration = ARFaceTrackingConfiguration()
if #available(iOS 13.0, *) {
    configuration.maximumNumberOfTrackedFaces =
        ARFaceTrackingConfiguration.supportedNumberOfTrackedFaces
}
sceneView.session.run(configuration, options: [.resetTracking,
        .removeExistingAnchors])
```



# MOTION CAPTURE

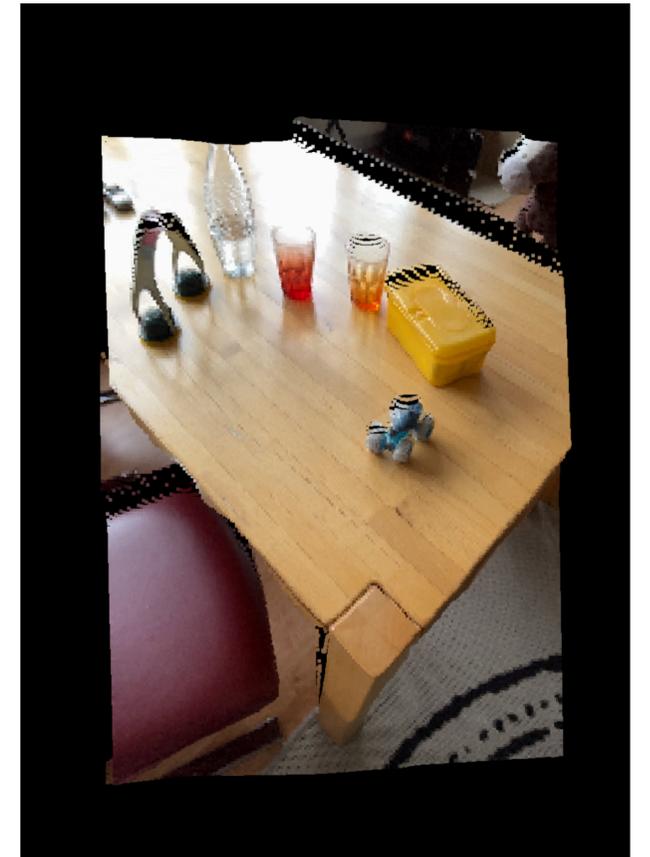
```
// If the iOS device doesn't support body tracking, raise a developer error for
// this unhandled case.
guard ARBodyTrackingConfiguration.isSupported else {
    fatalError("This feature is only supported on devices with an A12 chip")
}

// Run a body tracking configuration.
let configuration = ARBodyTrackingConfiguration()
arView.session.run(configuration)

arView.scene.addAnchor(characterAnchor)
```

# LIDAR – SCENE DEPTH

```
guard ARWorldTrackingConfiguration.supportsFrameSemantics([.sceneDepth,  
    .smoothedSceneDepth]) else { return }  
// Enable both the `sceneDepth` and `smoothedSceneDepth` frame semantics.  
let config = ARWorldTrackingConfiguration()  
config.frameSemantics = [.sceneDepth, .smoothedSceneDepth]  
arSession.run(config)
```



Und nun ...

**... ein bisschen Live-Coding**

# ZUSAMMENFASSUNG UND FRAGEN

ARKit bietet inzwischen sehr vielfältige Einsatzmöglichkeiten, von Mesherkennung über Szenenverständnis zu Tracking, 3D-Scanner, usw.

Einfache Konfiguration

Viel Doku, Videos (WWDCs)

Manchmal schwierig, die Daten zu bekommen, die man braucht

# LINKS

ARKit: <https://developer.apple.com/documentation/arkit>

Displaying a Point Cloud Using Scene Depth: [https://developer.apple.com/documentation/arkit/environmental\\_analysis/displaying\\_a\\_point\\_cloud\\_using\\_scene\\_depth](https://developer.apple.com/documentation/arkit/environmental_analysis/displaying_a_point_cloud_using_scene_depth)

ARKit 5: <https://developer.apple.com/augmented-reality/arkit/>

WWDC 2021: <https://developer.apple.com/videos/play/wwdc2021/10073/>