

Face Detection & Graphic Overlays

By Jan Duldhardt



Goal

“An application which detects the users face and applies different graphics on certain facial features or on the whole face altogether”



Tools & Platform

ARCore

- Good documentation
- Camera API included
- Good performance
- Open Source (Apache 2.0)

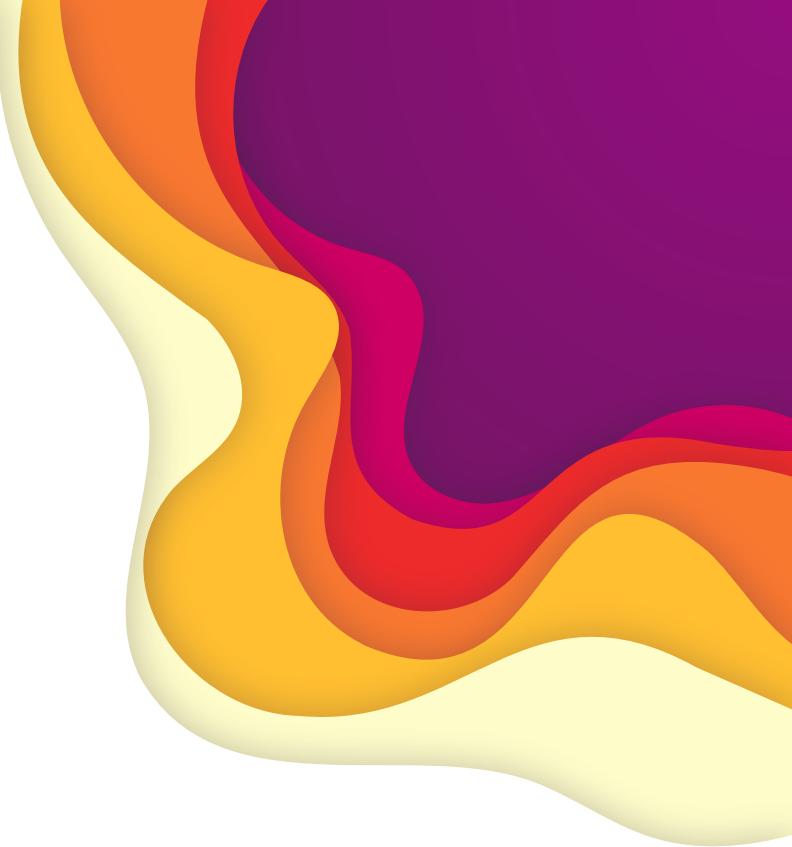


ARCore



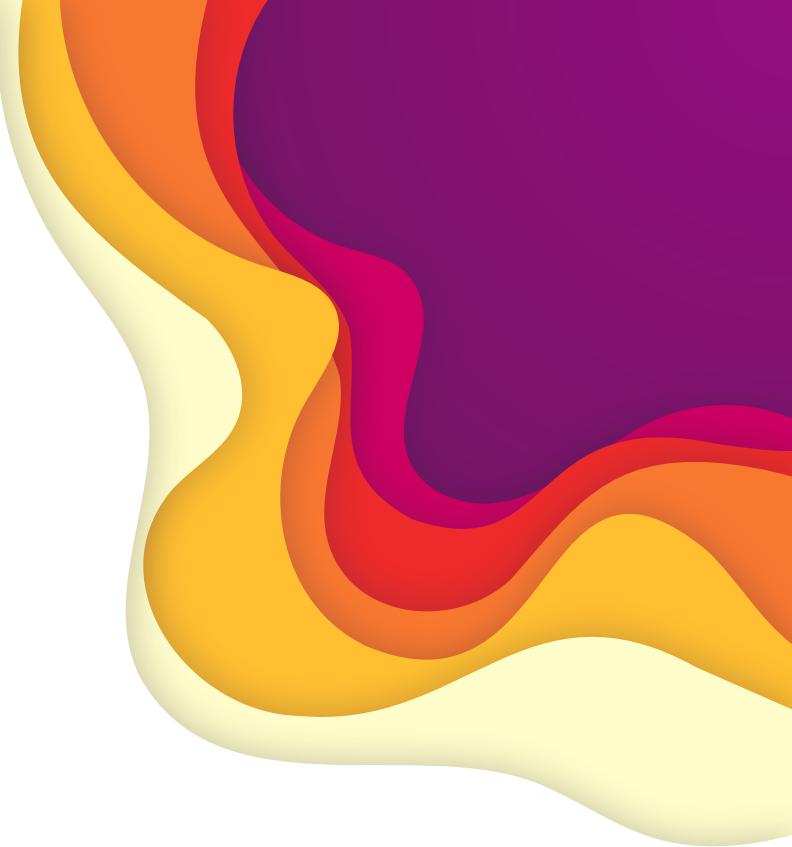
ARCore

- Motion tracking
- Environmental understanding
- Light estimation
- **Augmented faces**



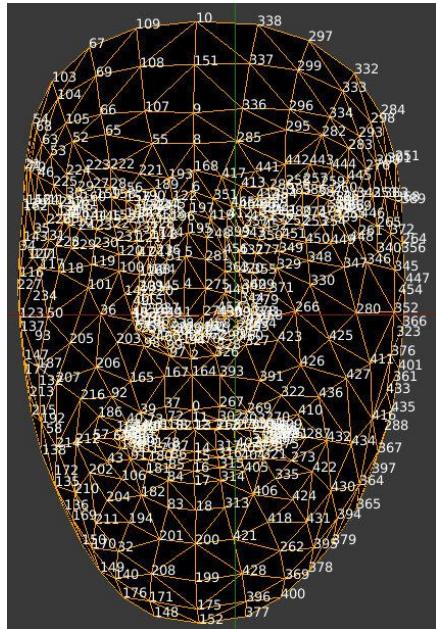
Face detection

- Scans the scene for faces
- Checks if face is tracking
- Otherwise adds to tracking list

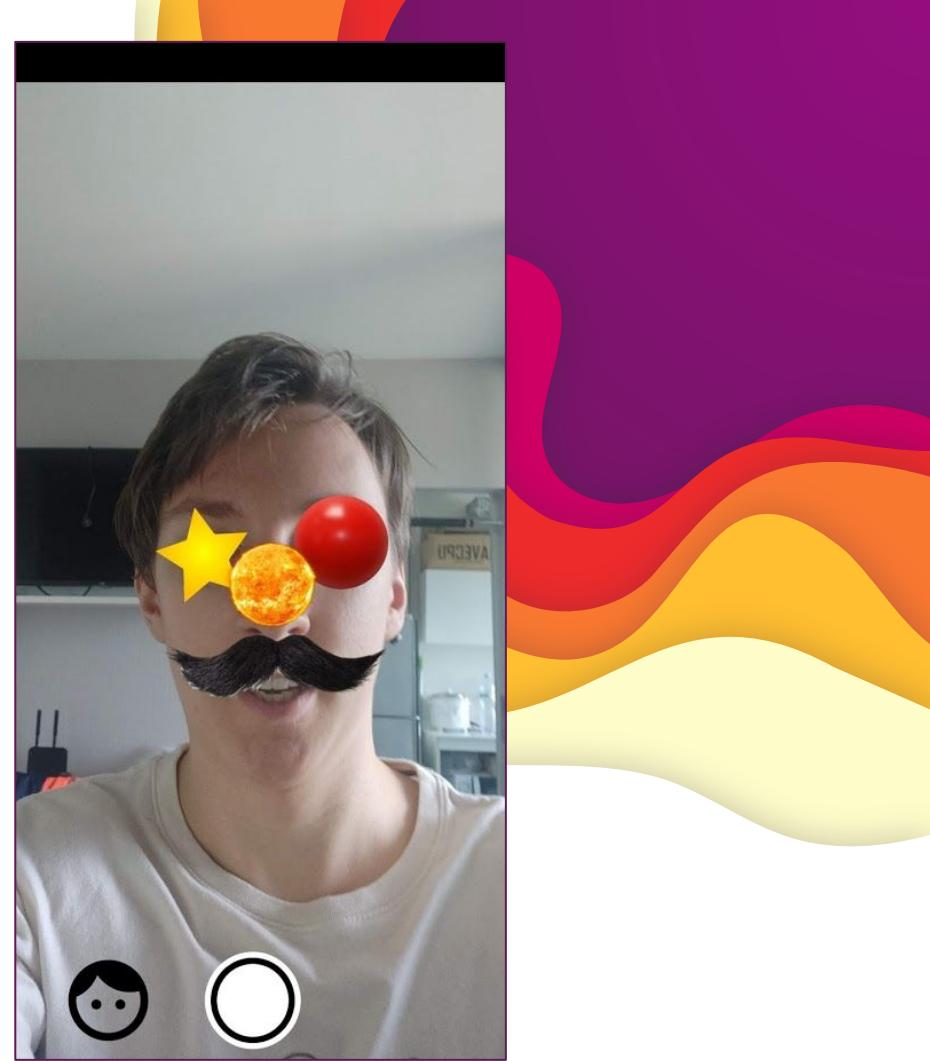
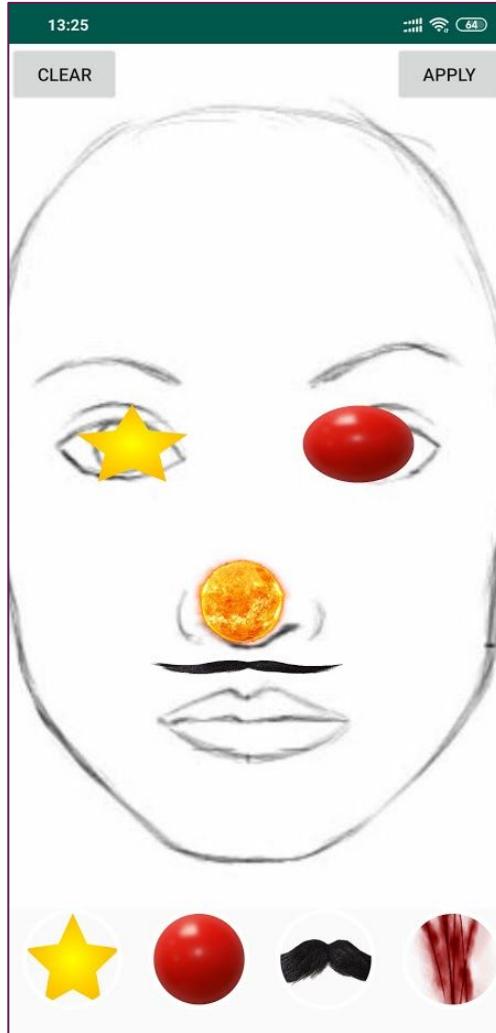


First approach: Coordinates of mesh vertices

- Apply the graphic through coordinates
- Similar to OpenCV dlib facial landmarks approach

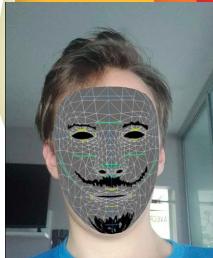
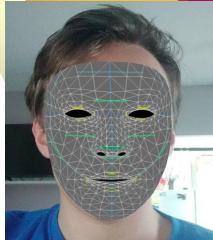
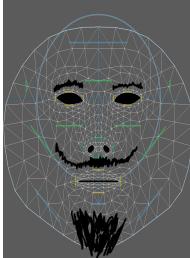
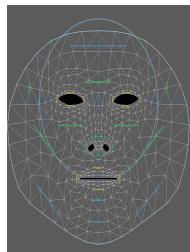


468 * 3 Points (X, Y, Z)



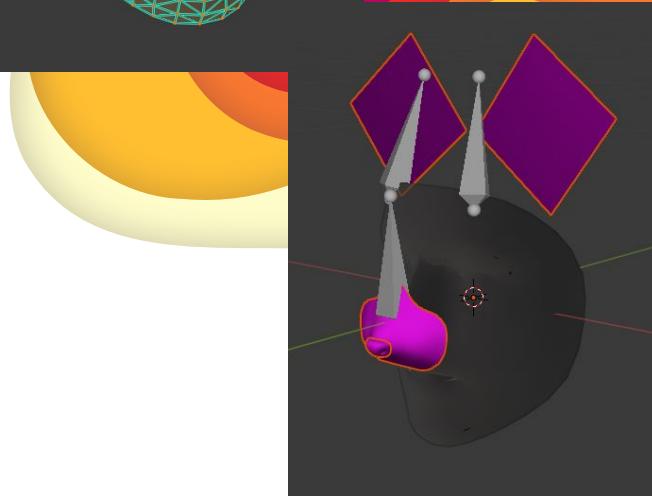
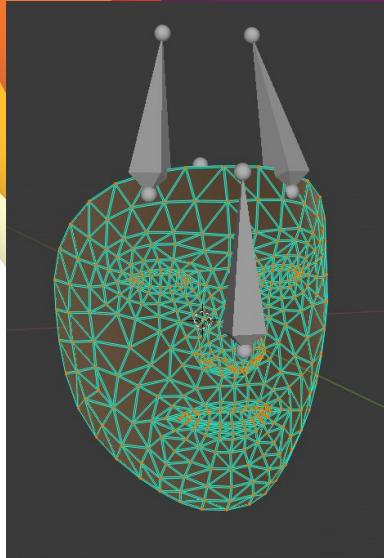
Second approach: Apply on the whole face

- Face mesh texture like a second skin
- 3D face → 2D texture



Second approach (Bonus): 3D assets

- 3 Anchor Points (NOSE_TIP, FOREHEAD_RIGHT, FOREHEAD_LEFT)
- Place 3D objects on the face (3D modeling skills required)





Requirements for usage

- Well lit environment
- No face covers
- OpenGL 3.0 and ARCore support



The product (POC)



Testing the face detection



Little head movement

Positives 292
Negatives: 0
100%



Always moving background,
Face angle changes up and down

Positives 550
Negatives: 64
90%



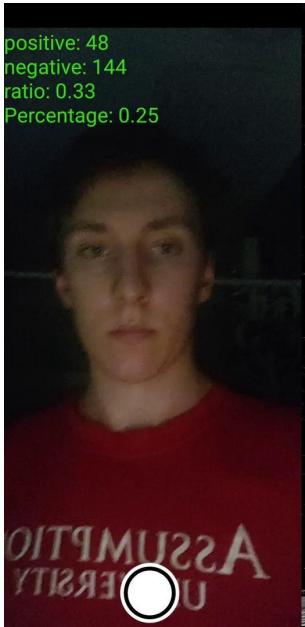
Room tour, real case scenario approach bad/good lightning lots of movement

Positives 377
Negatives: 54
87%

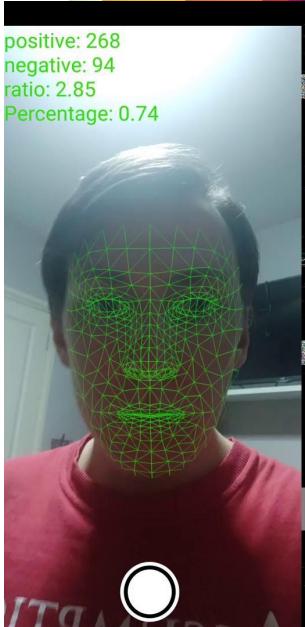
Testing bad light



Very low light, only light from display screen
P: 48 N: 144 0.25%



low light
P: 154 N: 174 0.47%



Light overexposure
P: 586 N: 234 0.71%

Testing conclusion

- Almost perfect in well lit areas
- Good detection from bad angles
- Very fast processing (framerate)
- Problems with bad lightning conditions



GitHub

<https://github.com/janduldhardt/AugmentedFacesARCore>

