

Introduction to Augmented Reality

Creating Augmented reality costume: concepts, tools, decisions, failures

Aurelijus Banelis



Aurelijus Banelis

Software developer
aurelijus.banelis.lt
aurelijus@banelis.lt



Augmented Reality



What is

Augmented Reality



To the end-user

- Real world



devrooms
5000+ hackers
608 lectures

January 2016



Fosdem

- Real world
- Camera



- Real world
- Camera
- Artificial



- Real world
- Camera
- Artificial
- Follows



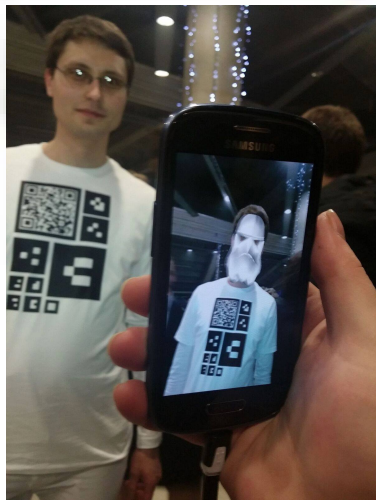
A hand holding a smartphone. The screen shows a city street scene with a colorful, glowing rectangular overlay in the center, representing augmented reality. The background is dark and out of focus.

Augmented Reality

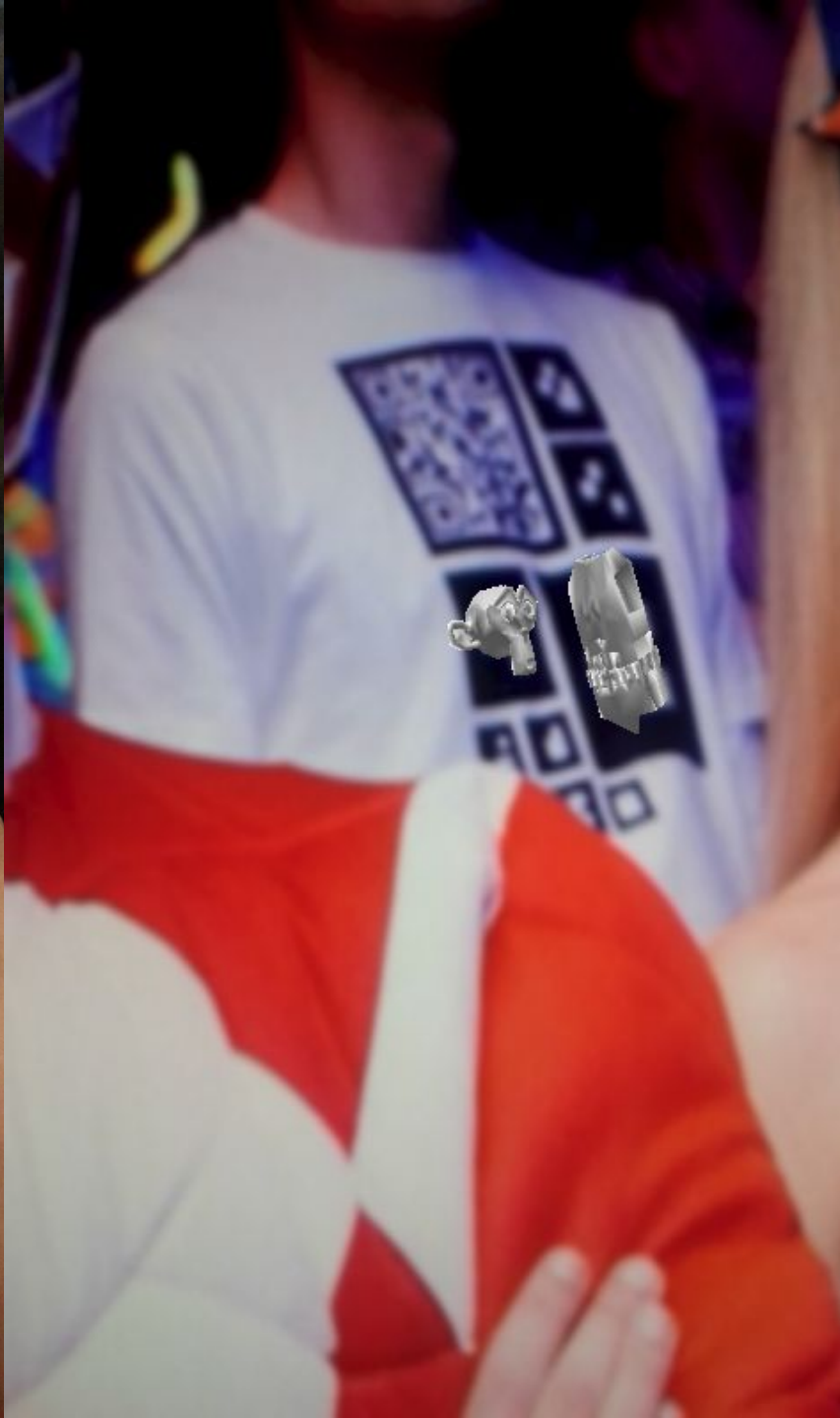
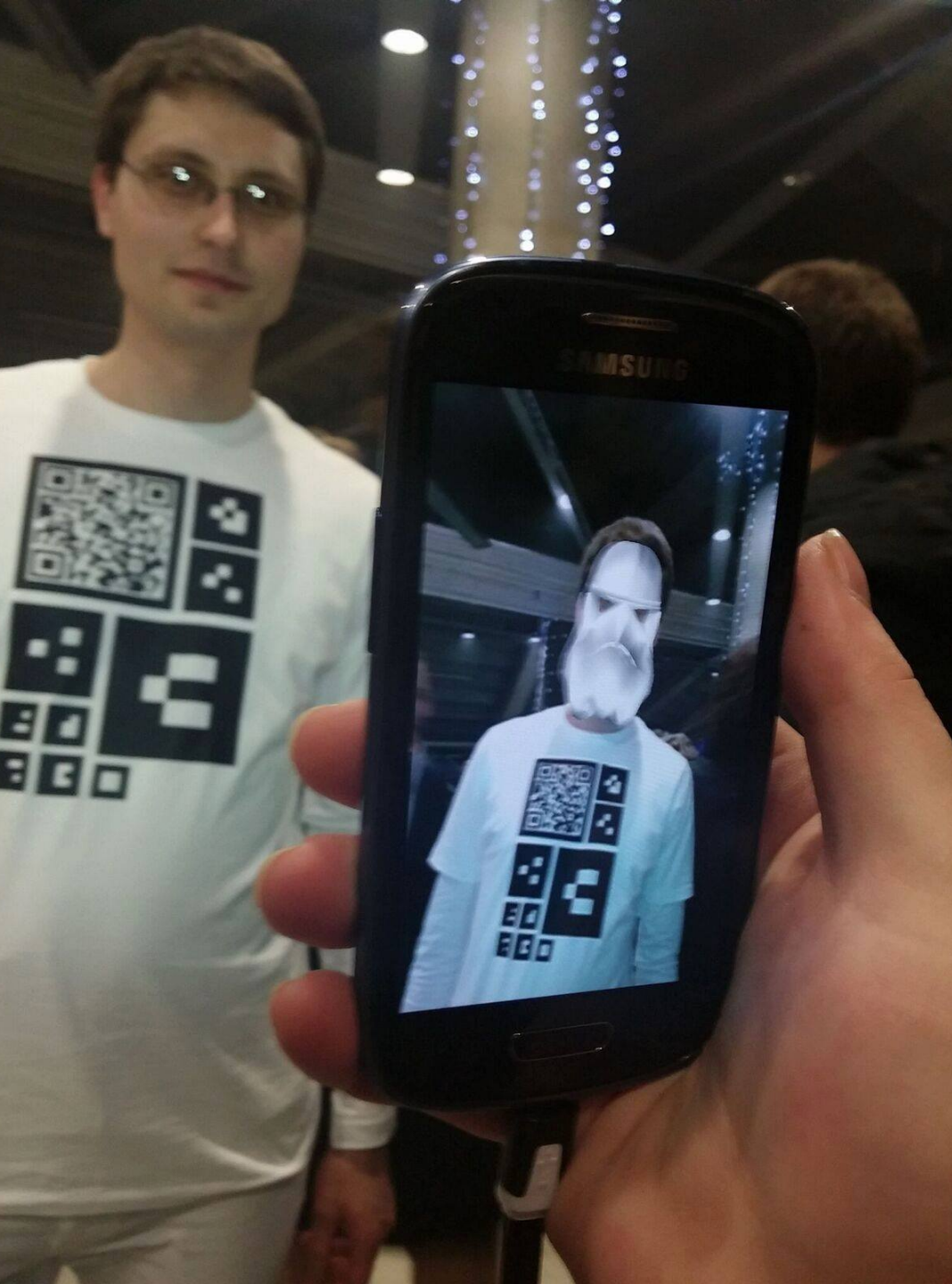
Hello world example



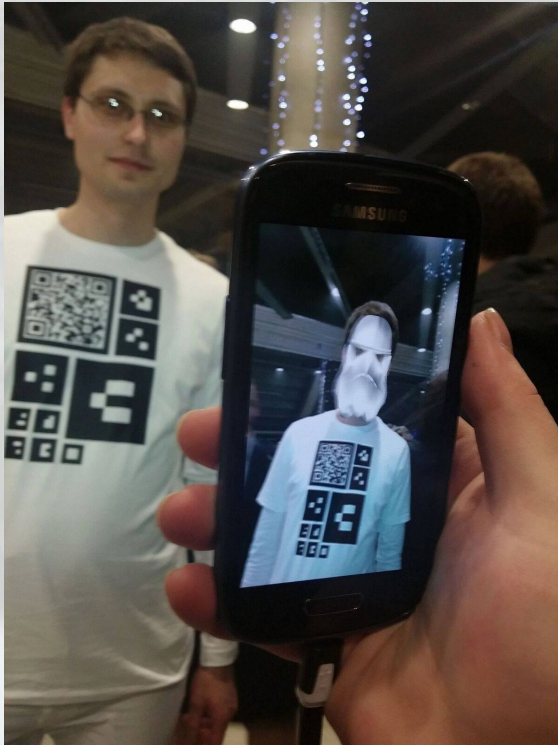
≠



Complete product



Augmented Reality



Computer Vision

3D Graphics

Infrastructure

RAW data → Objects

**Data from
camera**

**Noise,
Useful**

**Analyse,
Compare**

**Meaning,
decision**

RAW data → Objects

Data from camera

Noise, Useful

Analyse, Compare

Meaning, decision

125142365814152
451542247582451
424125384864535
315315313547876
815453151313515
748448343818131



RAW data → Objects

Data from camera

125142365814152
451542247582451
424125384864535
315315313547876
815453151313515
748448343818131



Noise, Useful

001000011100010
010100001110010
000001010110101
001001000101111
101010010000101
101001000101000



Analyse, Compare

Meaning, decision

RAW data \rightarrow Objects

Data from
camera

```
125142365814152
451542247582451
424125384864535
315315313547876
815453151313515
748448343818131
```



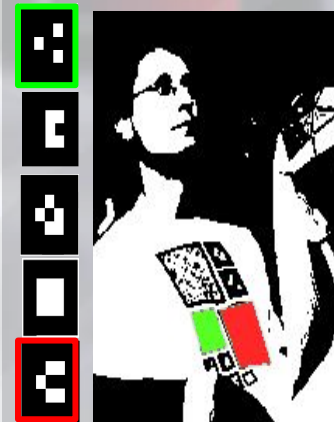
Noise,
Useful

```
001000011100010
010100001110010
000001010110101
001001000101111
101010010000101
101001000101000
```



Analyse,
Compare

```
001000011100010
010100001110010
000001010110101
001001000101111
101010010000101
101001000101000
```



Meaning,
decision

RAW data → Objects

Data from
camera

```
125142365814152
451542247582451
424125384864535
315315313547876
815453151313515
748448343818131
```



Noise,
Useful

```
001000011100010
010100001110010
000001010110101
001001000101111
101010010000101
101001000101000
```



Analyse,
Compare

```
001000011100010
010100001110010
000001010110101
001001000101111
101010010000101
101001000101000
```

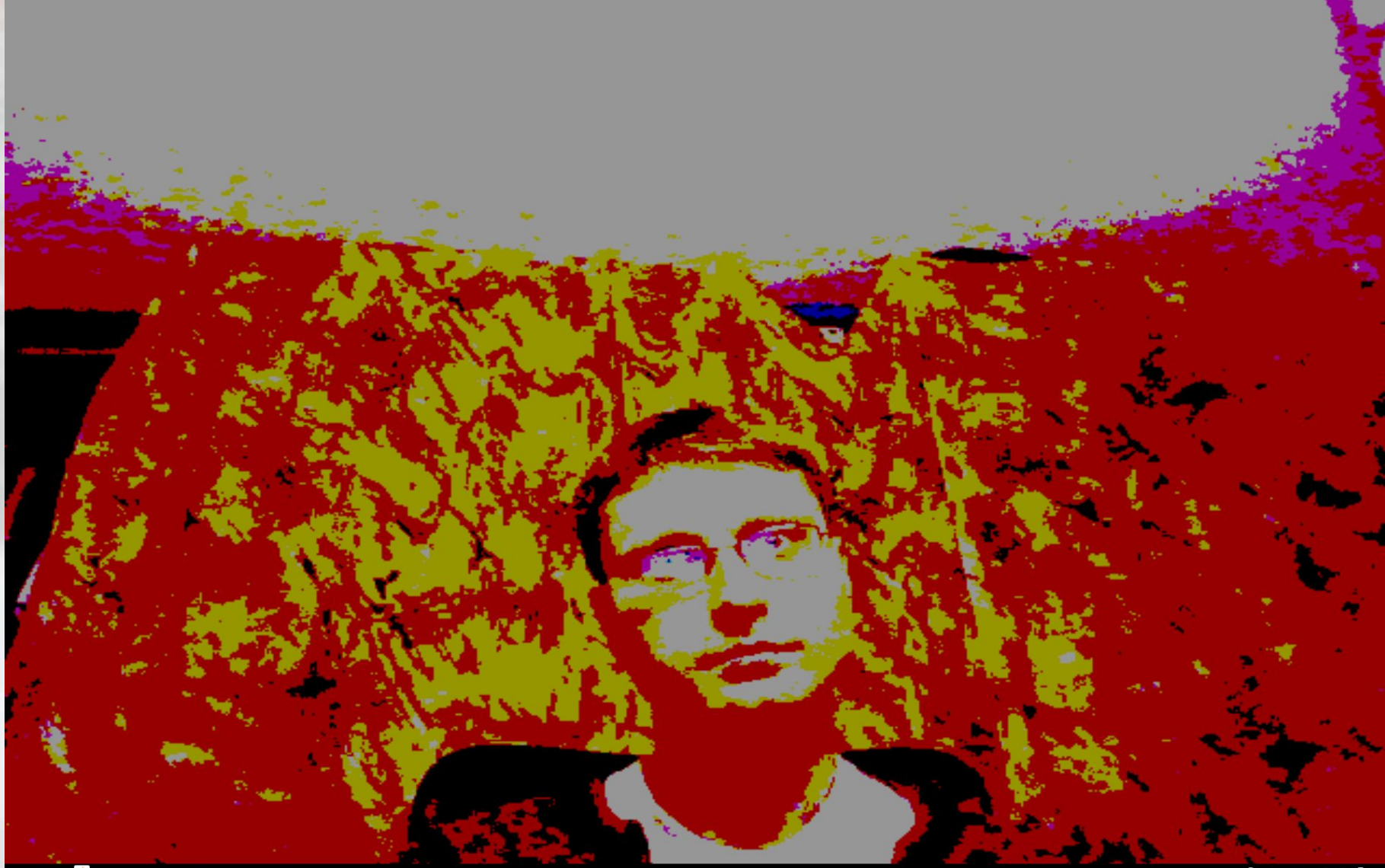


Meaning,
decision

```
Found {
  marker="m12"
  x = 10,
  y = 20,
  accuracy=0.8
}
```

```
Found {
  marker="m98"
  x = 15,
  y = 44,
  accuracy=0.6
}
```

Noises by nature



So filters in use



Including redundancy

Computer Vision

3D Graphics

Infrastructure



Low light

Low resolution

Folding

Similarity

if marker1 and marker2

CV → 3D

**Frame from
camera**

**Data from
ARToolkit**

**3D models
to be drawn**

Combined

+

+



⇒

CV → 3D

Frame from camera

Data from ARToolkit

3D models to be drawn

Combined

Captured preview image from device's camera



+

+



CV → 3D

Frame from camera

Data from ARToolkit

3D models to be drawn

Combined

Captured preview image from device's camera



+

```
Found {  
  marker="m12"  
  rot = (10, 20, 5)  
  scale = 0.3  
  pos = (5, 4, 22)  
}
```

+

```
Found {  
  marker="m98"  
  rot = (5, 33, 44)  
  scale = 0.4  
  pos = (22, 8, 3)  
}
```



CV → 3D

Frame from camera

Data from ARToolkit

3D models to be drawn

Combined

Captured preview image from device's camera



+

```
Found {  
  marker="m12"  
  rot = (10, 20, 5)  
  scale = 0.3  
  pos = (5, 4, 22)  
}
```

+

```
Found {  
  marker="m98"  
  rot = (5, 33, 44)  
  scale = 0.4  
  pos = (22, 8, 3)  
}
```

⇒

OpenGL ES 1.1



CV → 3D

Frame from camera

Data from ARToolkit

3D models to be drawn

Combined

Captured preview image from device's camera



+

```
Found {  
  marker="m12"  
  rot = (10, 20, 5)  
  scale = 0.3  
  pos = (5, 4, 22)  
}
```

```
Found {  
  marker="m98"  
  rot = (5, 33, 44)  
  scale = 0.4  
  pos = (22, 8, 3)  
}
```

+

OpenGL ES 1.1

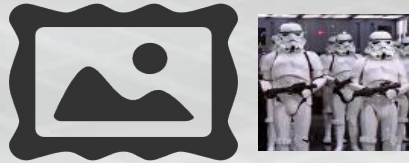


⇒

- Frame
- Transform
- Draw 3D



Idea → 3D models



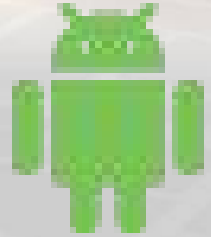
Idea / 2D pictures



3D Modeling soft



Wavefront

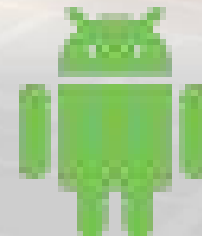


Android App

Idea → 3D models



.obj ⇒

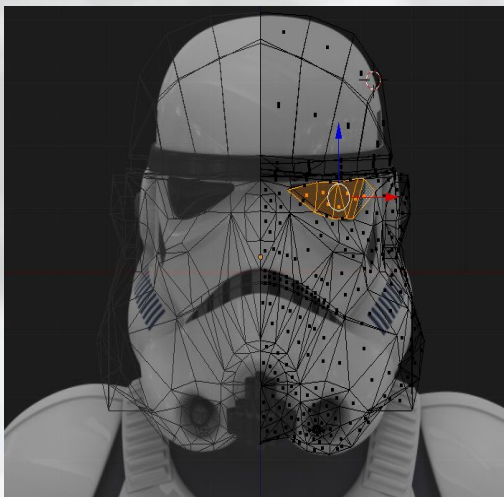


Idea / 2D pictures

3D Modeling soft

Wavefront

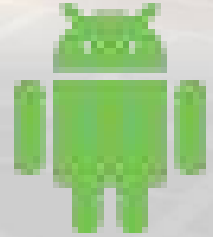
Android
App



Idea → 3D models



.obj ⇒

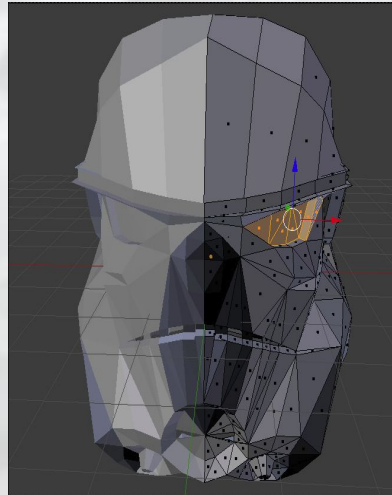
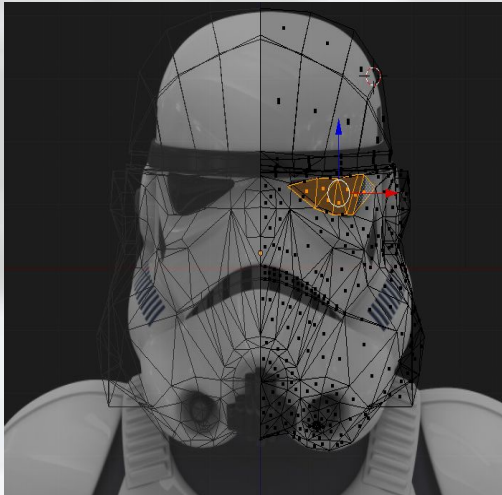


Idea / 2D pictures

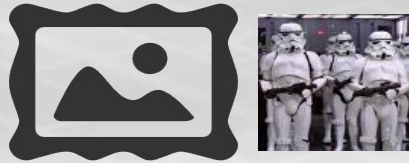
3D Modeling soft

Wavefront

Android
App



Idea → 3D models



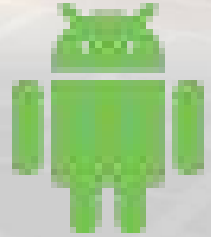
Idea / 2D pictures



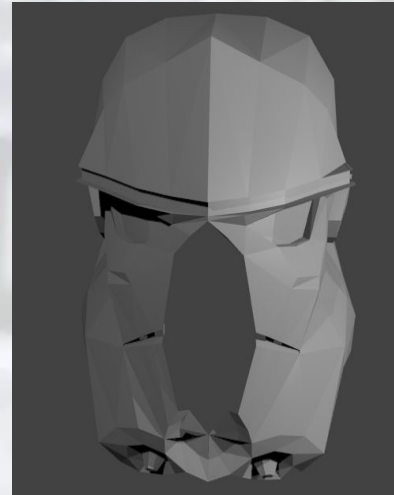
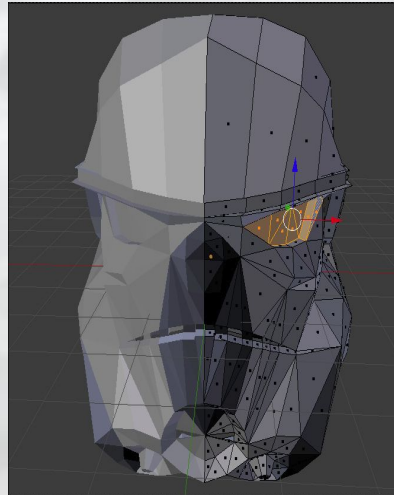
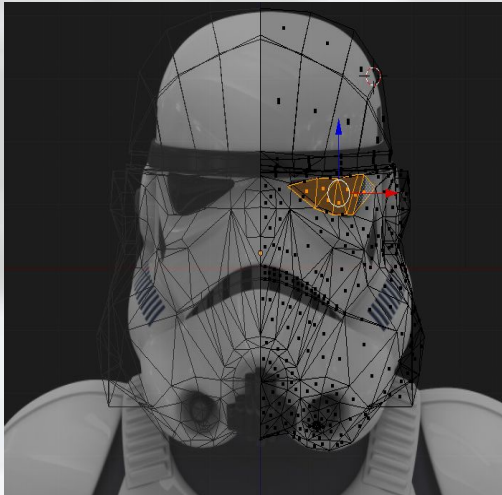
3D Modeling soft

.obj ⇒

Wavefront



Android App



Idea → 3D models



.obj

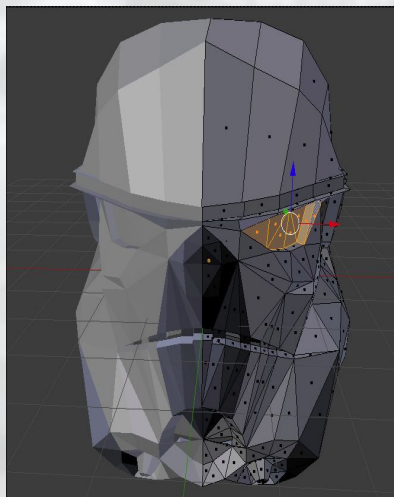
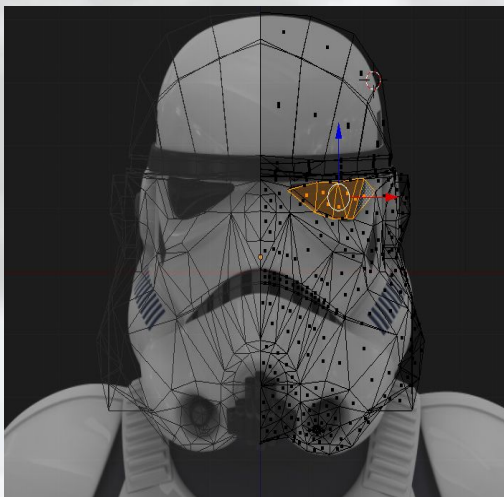


Idea / 2D pictures

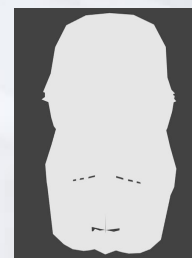
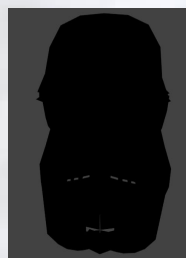
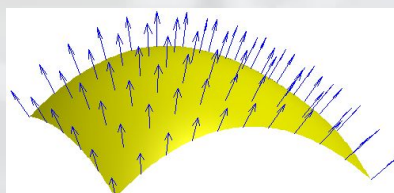
3D Modeling soft

Wavefront

Android App



Normals



Debug

Invest in 3D rendering

Debug separately

Copying ARToolKit native libraries takes time

Implement custom widgets for debugging
(faster than reloading whole code)

Consider native implementation

Garbage collection between Java native and *Buffer* structures takes time (slow loading)

End-to-end

ARToolKit

Wraps computer vision, starting point

Android (SDK)

Custom logic (e.g. helmet position)

Blender

Generating 3D models

QR + Server

Telling, how to find application

Draw on fabric

Some shops requires vector images



Questions?

References and useful links

- <http://aurelijus.banelis.lt/presentations/augmetned-reality-2015/ag-costume.pdf>
- <https://youtu.be/Klafn8hFi8c?t=4m41s>

- <http://artoolkit.org/>
- <https://github.com/ltseez/opencv>
- O'REILLY Learning OpenCV
- https://en.wikipedia.org/wiki/Scale-invariant_feature_transform
- https://en.wikipedia.org/wiki/Normal_%28geometry%29
- <https://unity3d.com/>
- <https://www.blender.org/>
- <https://inkscape.org/en/>