

Augmented Reality

From zero to hero

Aurelijus Banelis



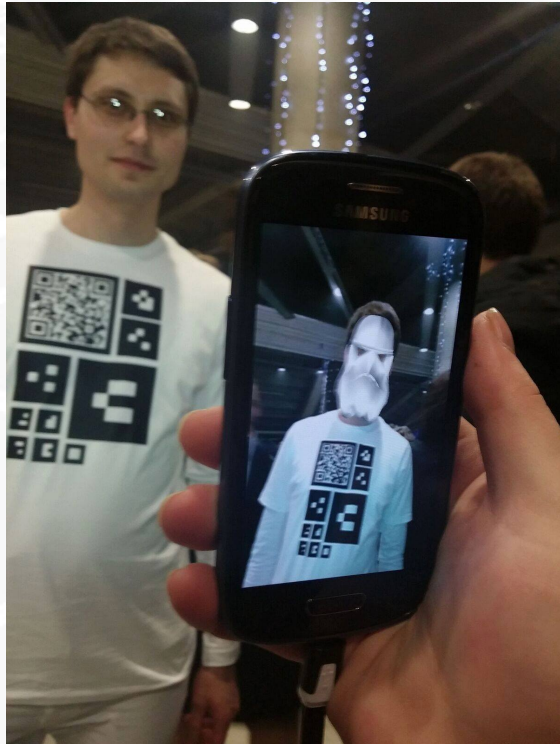
Aurelijus Banelis

aurelijus.banelis.lt

aurelijus@banelis.lt



Augmented Reality



Computer Vision

3D Graphics

Infrastructure

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
```



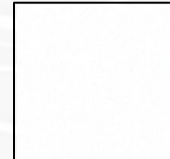
Meaning, decision

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

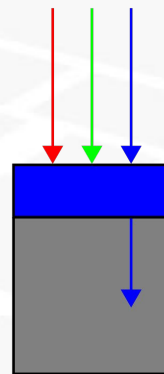
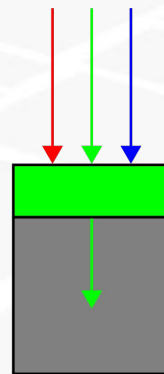
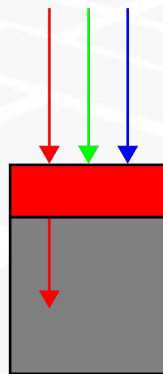
Found {
  marker="m98"
  x = 15,
  y = 44,
  accuracy=0.6
}
```

Problem: Data noise

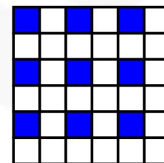
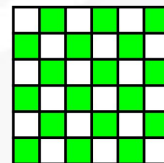
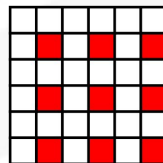
Light



White balance



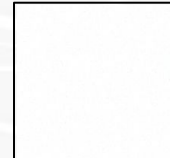
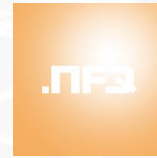
Blur, movement



Perspective

Problem: Data noise

Light



White balance



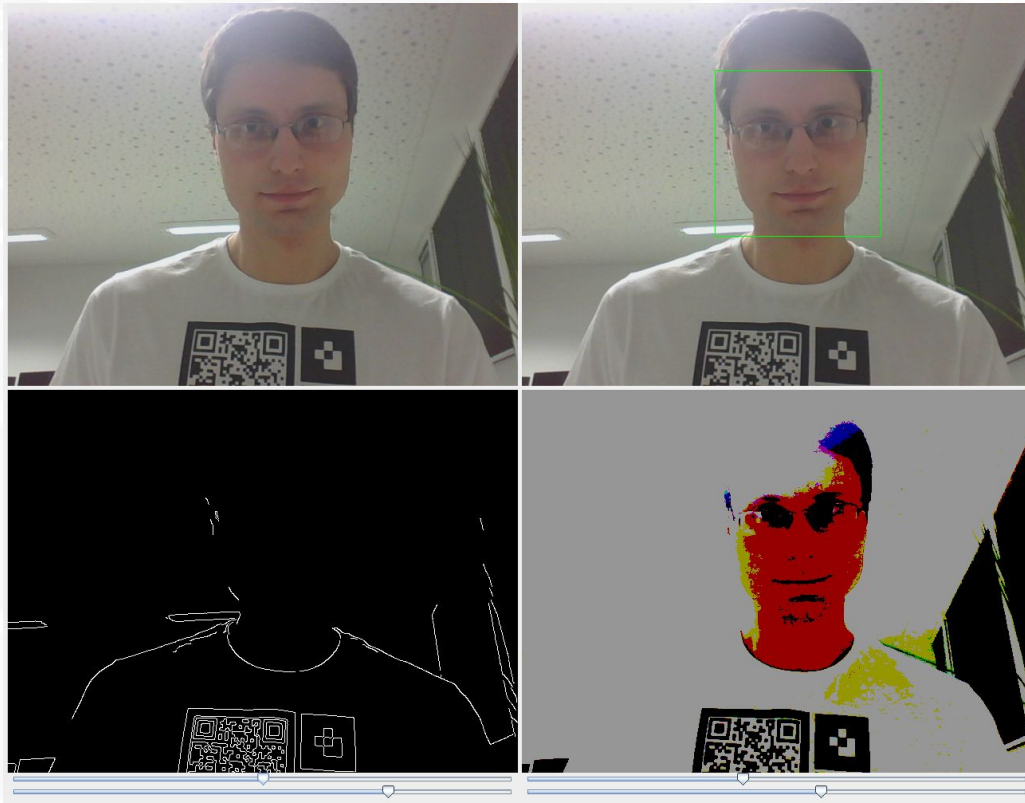
Blur, movement



Perspective



Demo time



- Face recognition in action
- Different filters for easier RAW data parsing

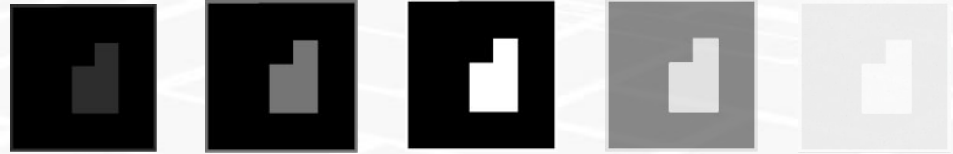
Based on OpenCV example: <https://github.com/Itseez/opencv/tree/master/samples/java/sbt>

Custom code used in demo: <https://gist.github.com/aurelijusb/504e2b85cd252371c713>

Marker - easy to track

Contrast

Min-max threshold



White & black

Average



Square, border

$y = ax + b$



Asymmetric

SIFT

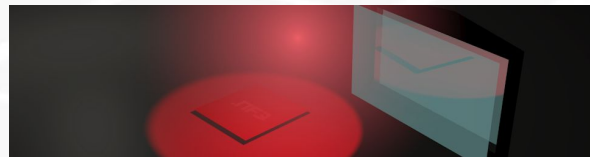


Enhance input devices

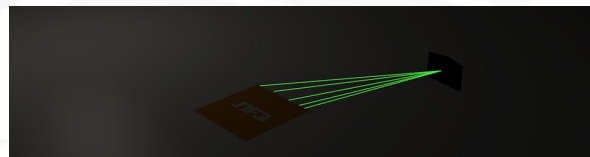
WebCam



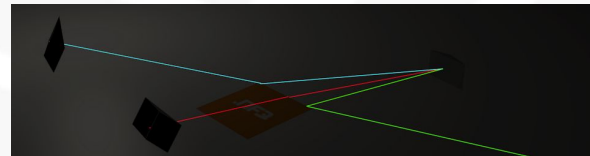
Infrared Cam



Lazer/echo



Mirrors (Mocap)



Including redundancy



Low resolution

Large markers for
camera to recognise

Multiple small markers
for error recovery

Folding

if marker1 and marker2

CV \rightarrow 3D

Frame from camera

Captured preview image from device's camera



Data from ARToolkit

```
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)  
}
```

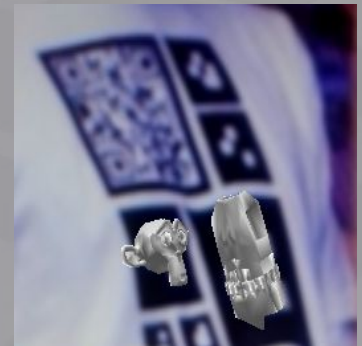
3D models to be drawn

OpenGL ES 1.1



Combined

- Frame
- Transform
- Draw 3D



Idea → 3D models



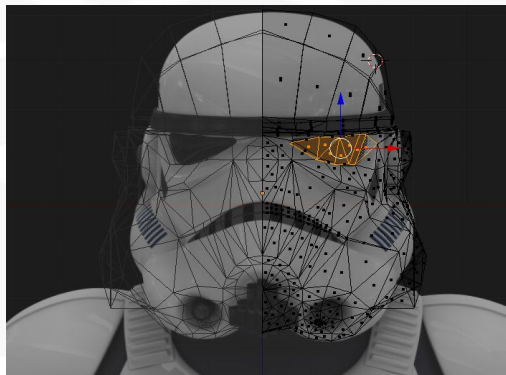
Idea / 2D pictures



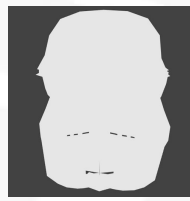
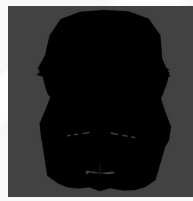
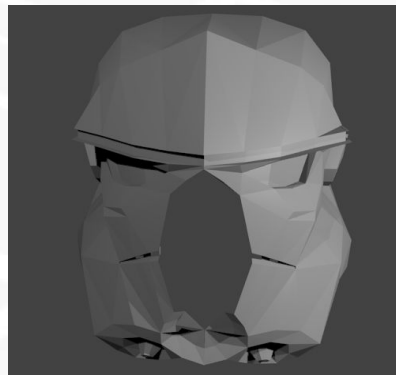
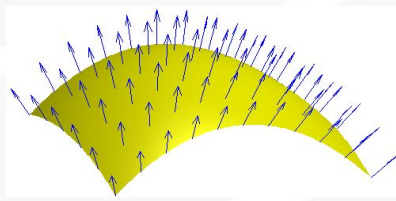
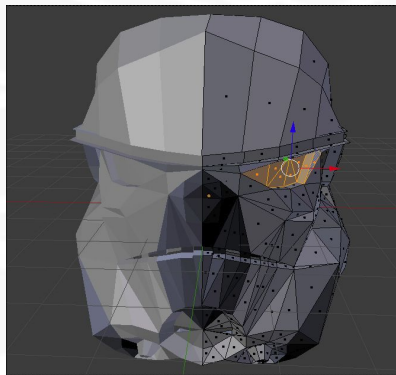
3D Modeling soft



Android App



Normals



Debug

3D models → device



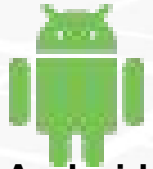
3D Modeling soft



WaveFront
(.obj)



Custom
parser



Android
App

OpenGL ES 1≠2

Unity did not worked for me

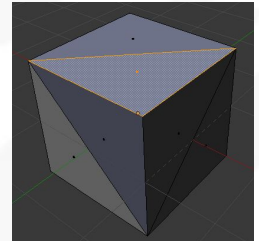
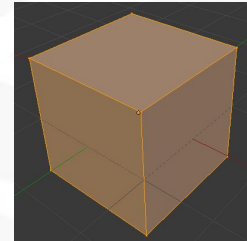
Hard to find ready
made .obj parser

glVertexPointer ≠
examples in C++

Triangulate

There is option in Blender exporter

Assigning
vertices to faces
and normals



Invest in 3D rendering

Debug separately

Copying ARToolKit native libraries takes time

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

Consider native implementation

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

End-to-end

ARToolKit

Open Source library for
Augmented reality

- Generate markers
- Native libraries

Android (SDK)

Custom logic in Mobile
Application

- OpenGL ES
- Device to test on

Blender

Generating 3D models

- WaveFront
exporter

QR + Server

Server from where to
download app

- QR code generator
- Simple HTML

Draw on fabric

Print markers/QR code
on T-shirt

- Vectorized image
- Other cloths

Conclusion

Open Source

All used software/tools was based on free or open source products

Concepts

Just using libraries not enough. Need concepts for better decisions

Early stage

Augmented Reality ecosystem is still young. Improvements in native code

Questions?



**You feedback is
always welcome**

**So next presentation could be
even better ;)**

References and useful links

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