



Roboy-made Lighthouse Sensor
60Hz, 1mm accuracy,
external object tracking

ROLLBODY

Roboy's little brother on wheels!

30.08.2022

Roboy Finals SS22

Johannes, Philipp W., Danylo, Philipp K., Emin, Deniz, Jefferson

FINALS SLIDES MUST INCLUDE

- semester team goal
 - why you are doing what you are doing
- final video
- what you achieved? how does it compare to what you've planned to achieve? how good is it?
- system design/ hardware and software architecture / important design decisions
- implementation details - the “meat” of your presentation, be specific and precise, keep in mind that your audience has a solid technical background, include concrete tools, algorithms, libraries, components you've used
- validation / verification / test results
 - (how did you measure? and how well did you do?)
- challenges (tech, organizational,etc) and learnings (I am a better engineer now, because I've learned: ...; Had I known this secret trick, my life would have been so much easier.)
- live demo (have a backup video in the slides with public access)



GENERAL HOW-TO

most importantly: make it awesome

Time limit

30 mins - Teams

10 mins - IDP, Theses

Use the template Slides at the end to put graphics etc.

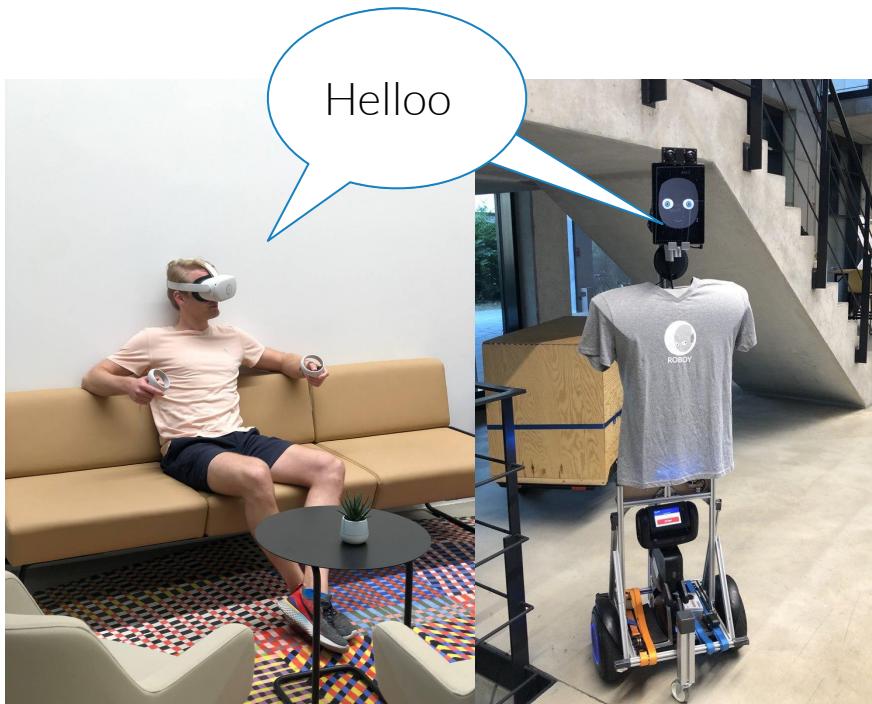
General tips

- Try not to have a boring presentation 😊
- Fonts size not smaller than 18 (if it's smaller you have too much infos on the slide)



OUR MISSION

...they see me rollin'

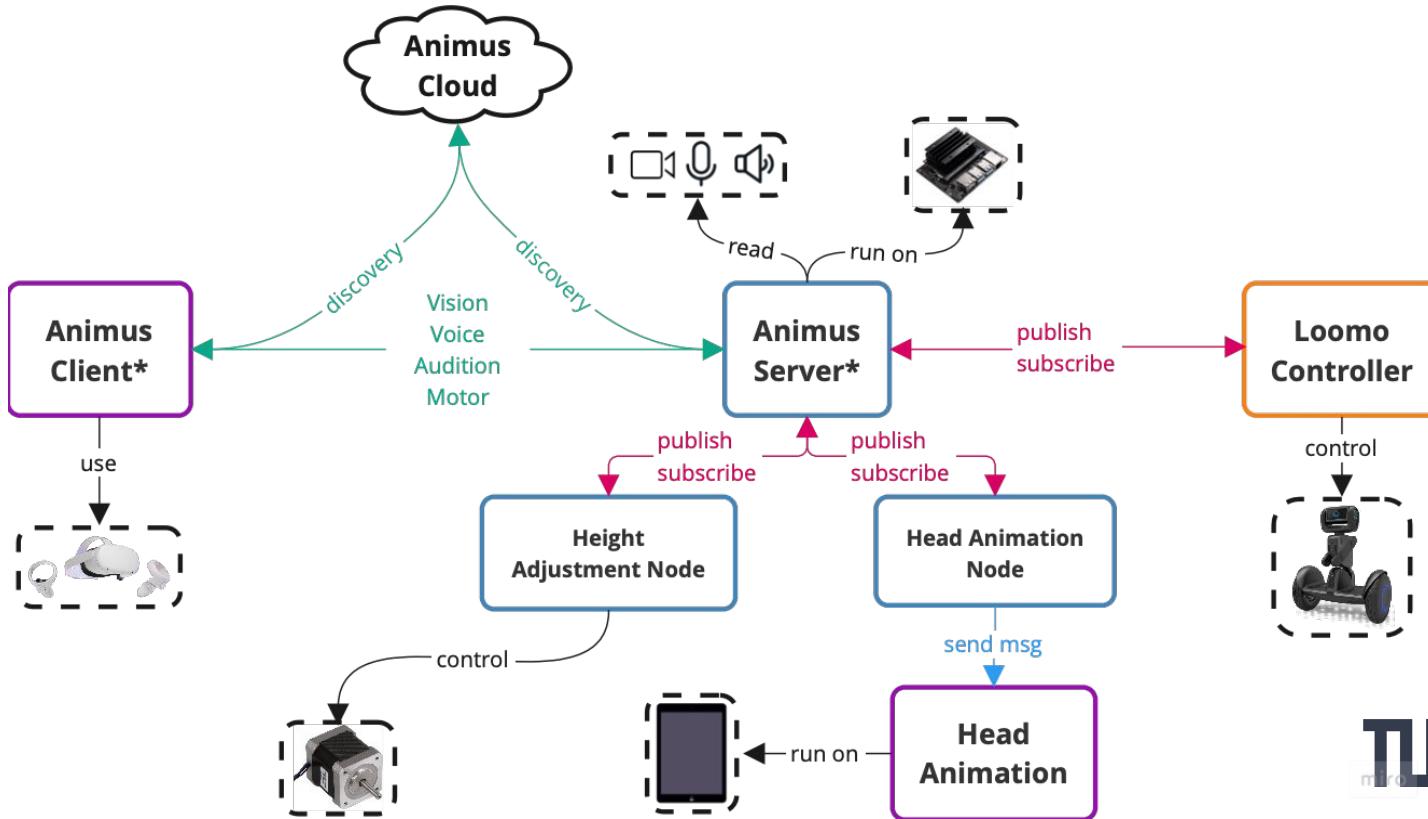


FINAL VIDEO

ROLLBODY

Overview

... before we dig in

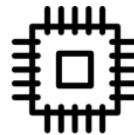


HARDWARE

Structure & Power



MECHANICAL
ENGINEERING

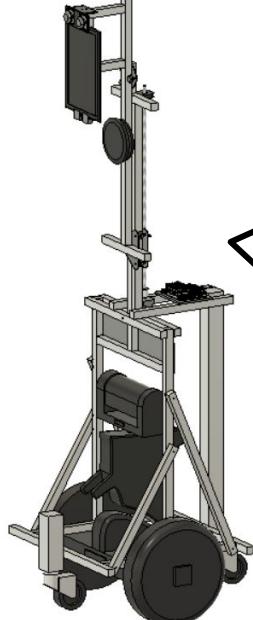


ELECTRICAL
ENGINEERING



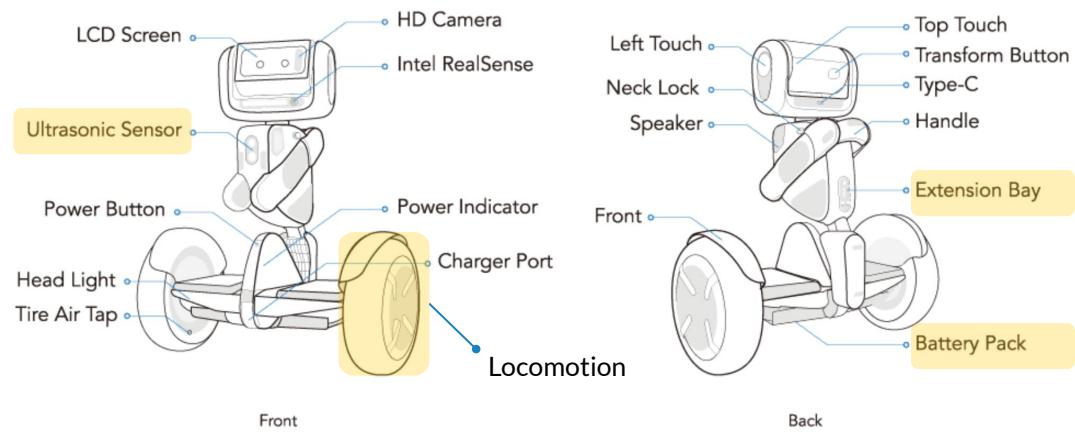
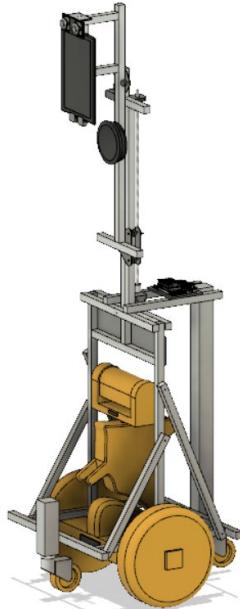
COMPONENTS

... that actually worked for us



LOOMO

a Segway for Developers



<http://developer.segwayrobotics.com/developer/documents/segway-robot-overview.html>



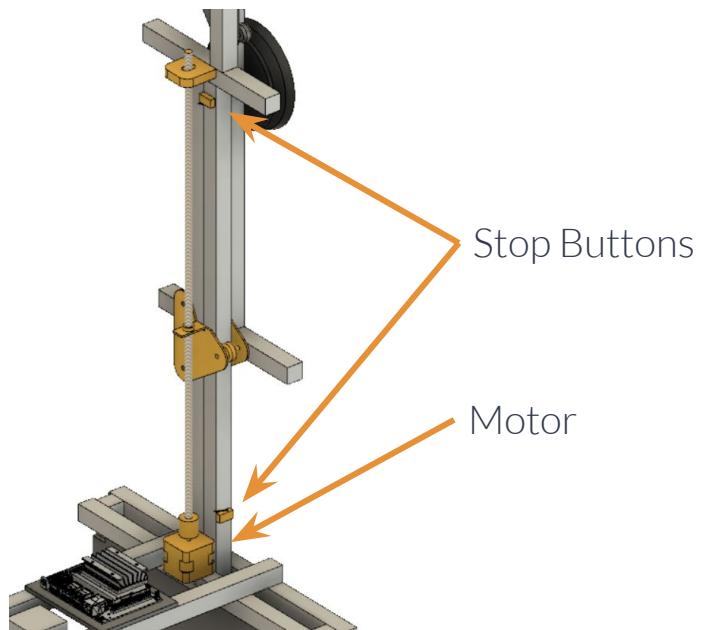
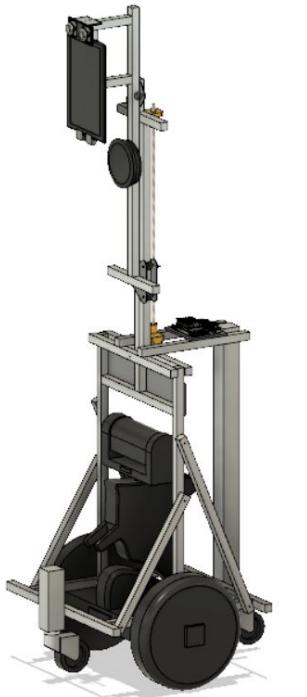
Rigid structure

Base and Head



HEIGHT ADJUSTMENT

Stepper motor & driver

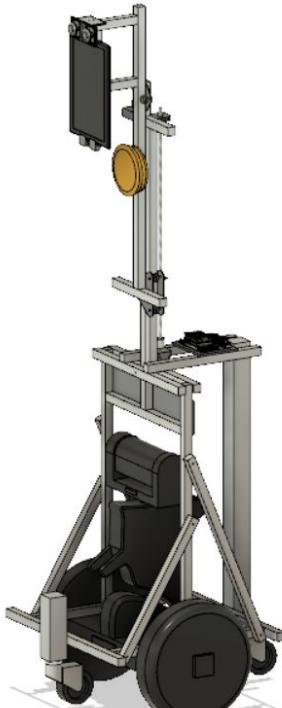


- Nema 17 step motor
- End trigger buttons for stopping motor in extreme positions
- Driver: L298N



AUDITION

Speaker & Microphone



- **Jabra speaker**
- Speaker
- Omnidirectional microphone

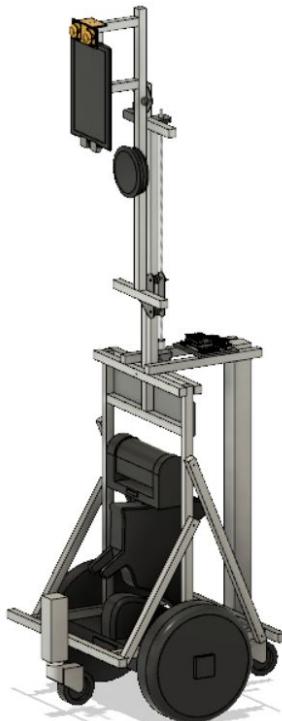
<https://www.jabra.com/de/business/speakerphones/jabra-speak-series/jabra-speak-410>

Jabra Speaker



Vision

Let there be light



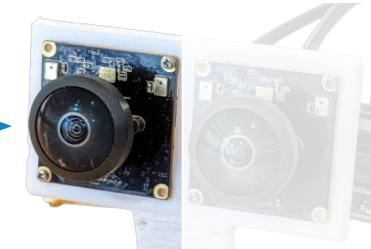
- **Key Features:** Stereo Vision, Adaptive Image



Logitech C720 [5]



ZED Mini [6]



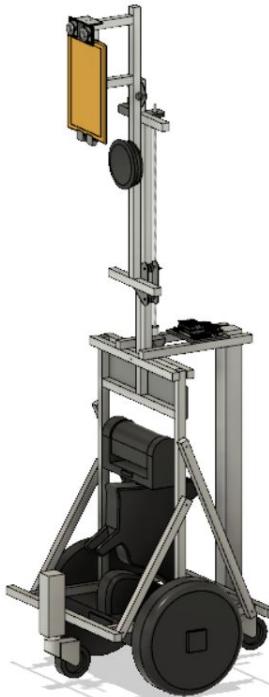
PiCam360 [7]

<https://www.picam360.com/>



“HEAD” TABLET

nicer than speaking to a void



- **Android tablet**
- Displays were the operator is looking
- Shows if the operator is connected/disconnected



JETSON NANO

Rollbody's brain

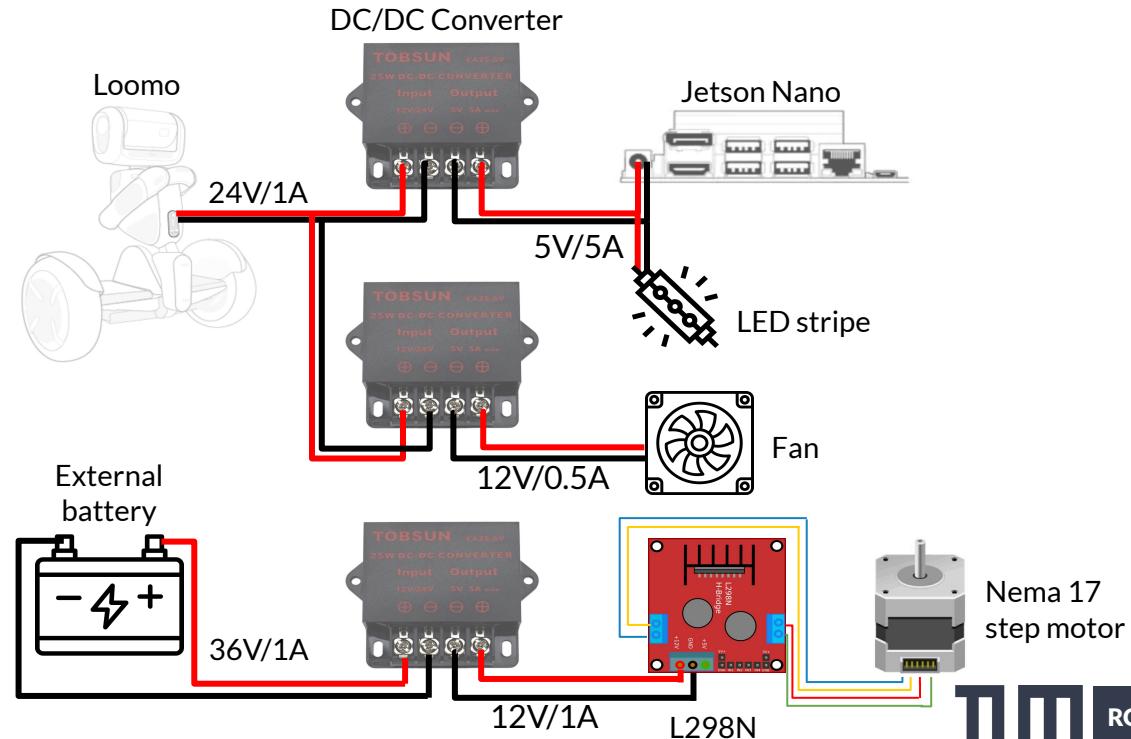
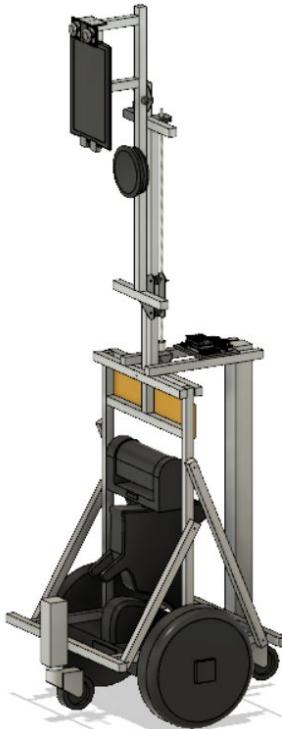


<https://developer.nvidia.com/embedded/jetson-nano-developer-kit>

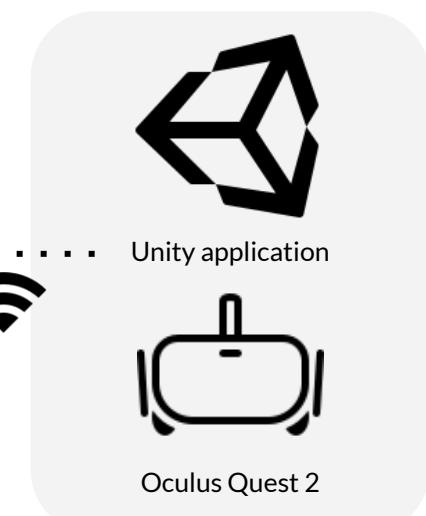
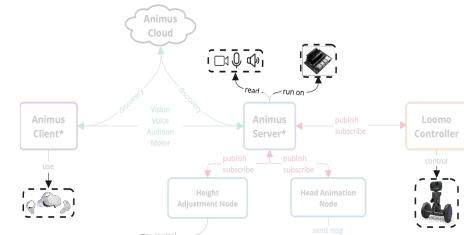
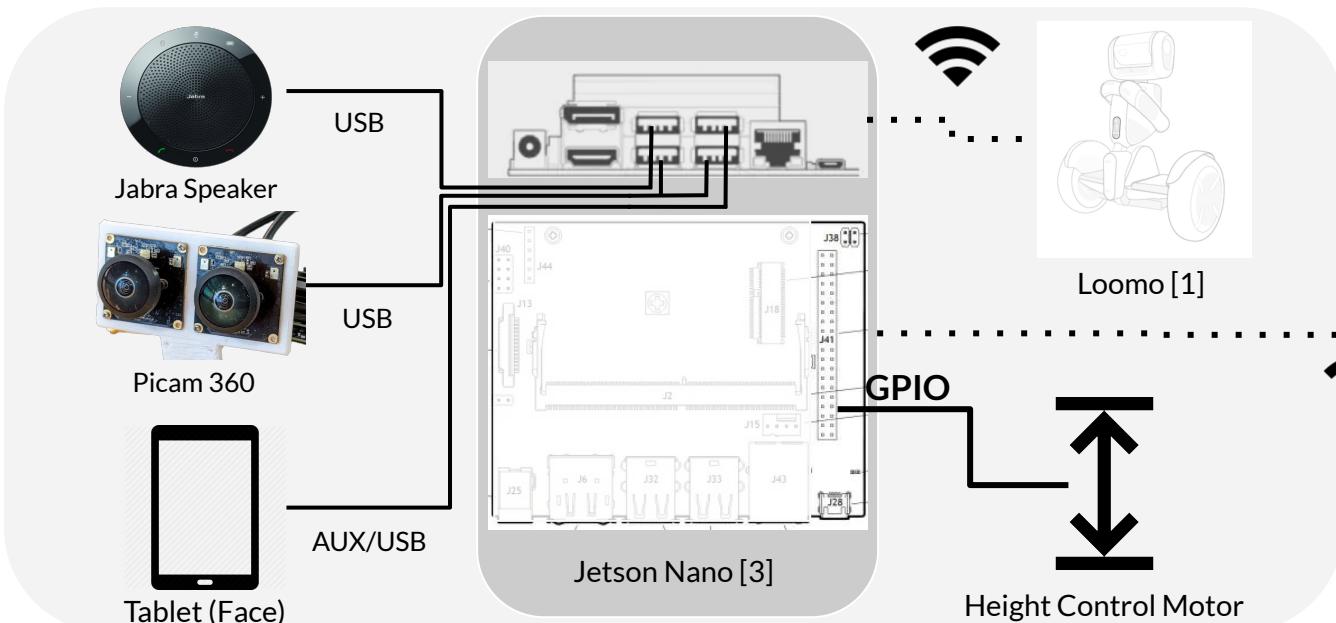


Electronics plates

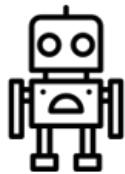
power supply



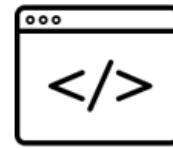
CONNECTIONS



SOFTWARE



ROBOTICS

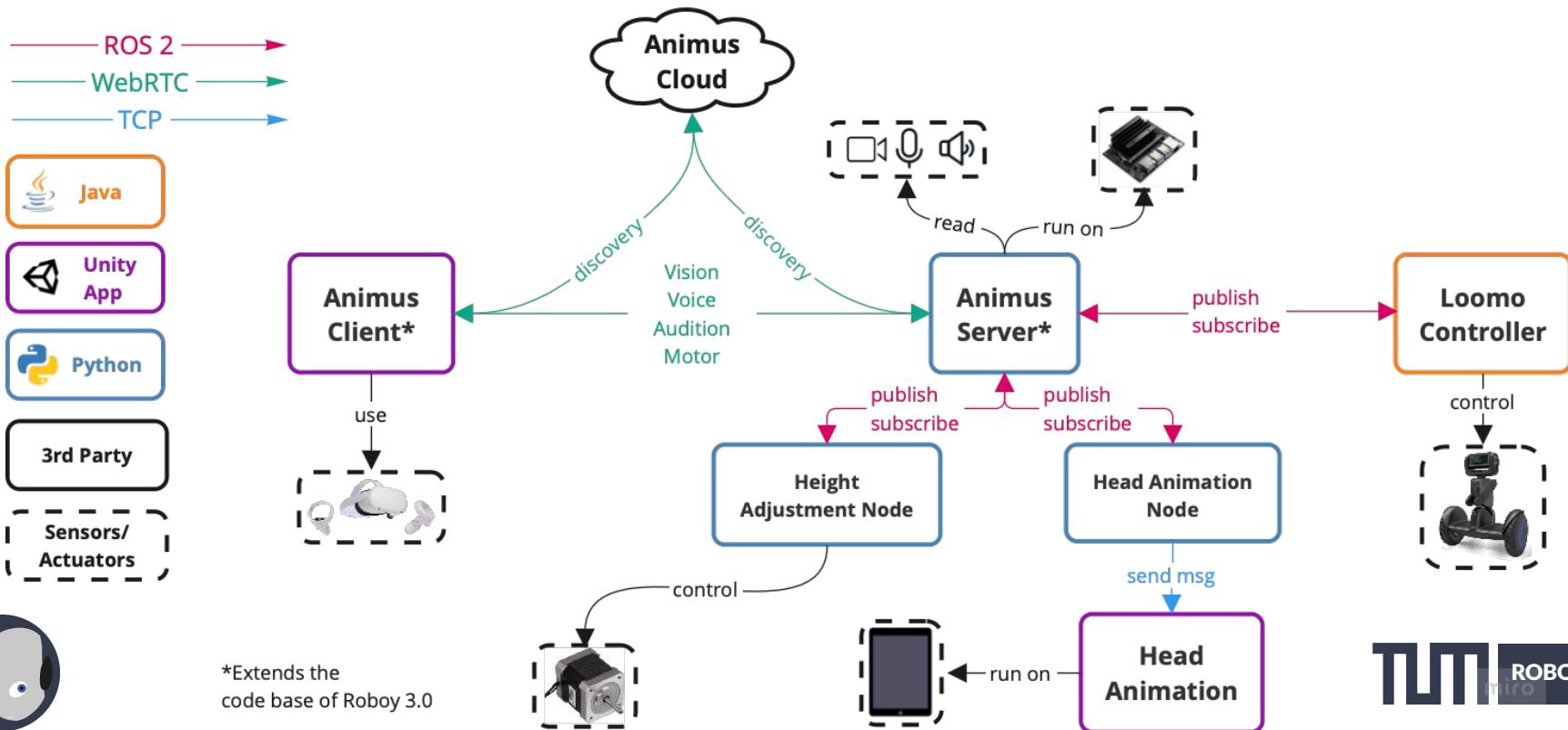


COMPUTER
SCIENCE



SOFTWARE ARCHITECTURE

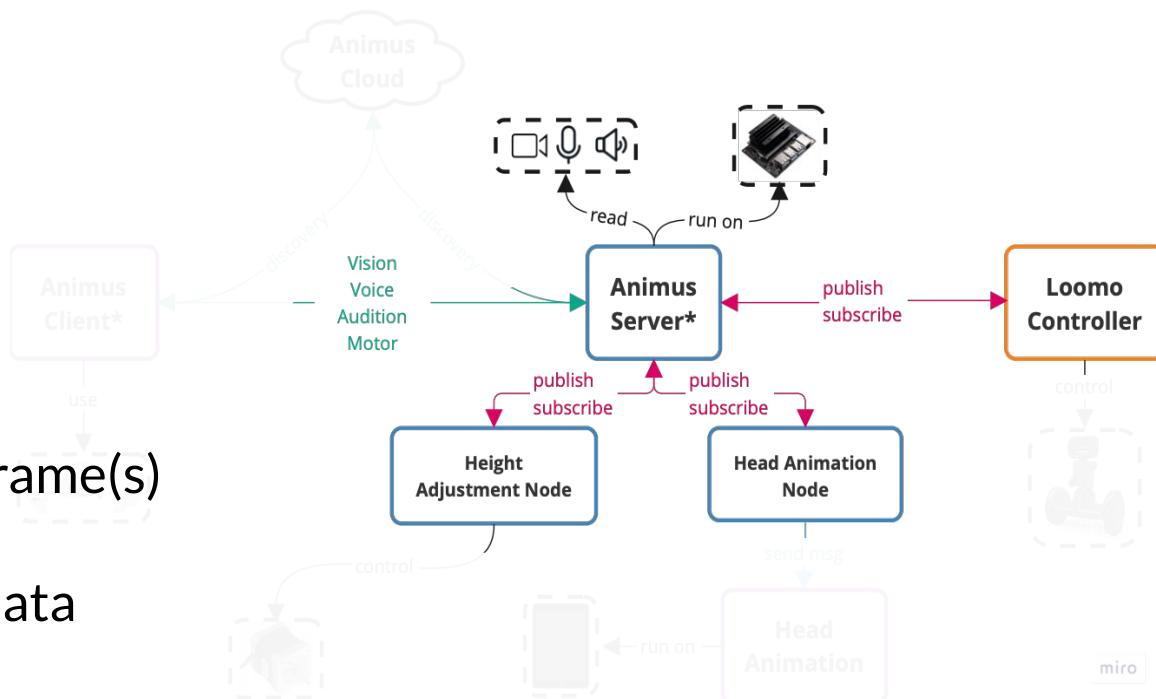
Connecting the Operator with Rollbody



ANIMUS SERVER

the backbone of Rollbody

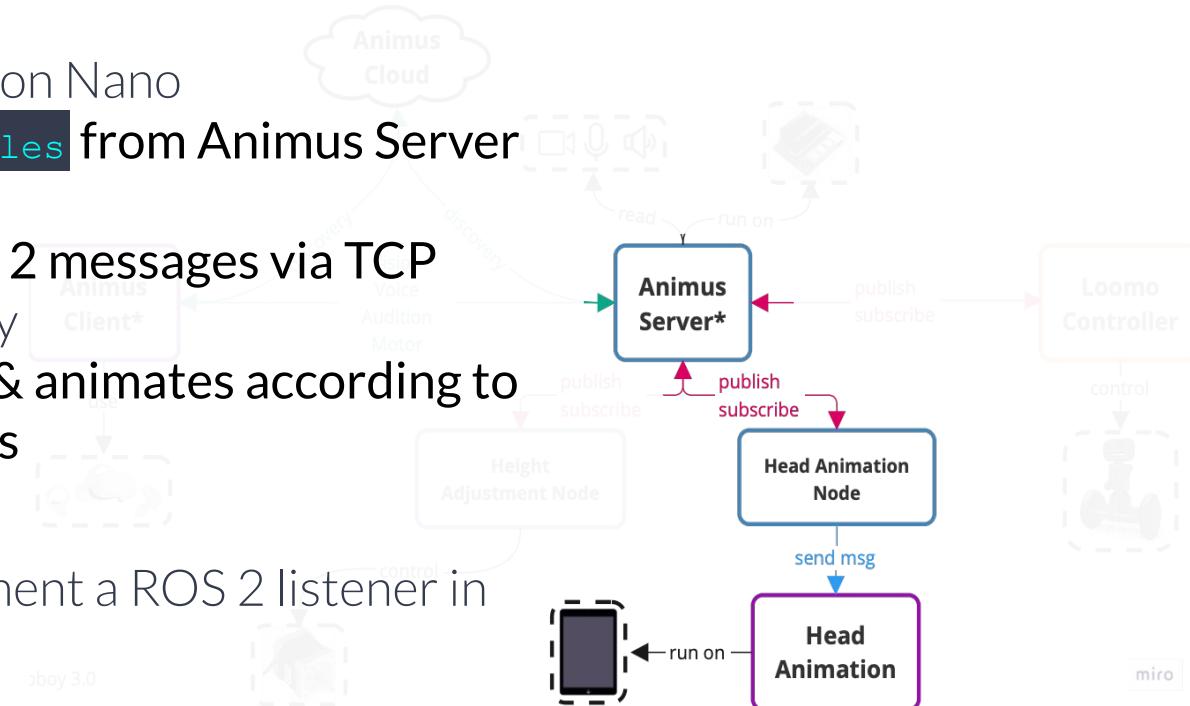
- Is a ROS 2 node
- Inputs from Client:
 - Operator's speech
 - [driving_cmds, height_adjust, head_angles]
- Outputs to Client:
 - Captured camera frame(s)
 - Recorded sound
 - Ultrasonic sensor data



Head Animation App

Leveraging the advantage of ROS 2

- ROS 2 node on the Jetson Nano
 - Listens to `head_angles` from Animus Server
 - Runs a TCP server
 - Forwards the ROS 2 messages via TCP
- Head animation in Unity
 - Runs a TCP client & animates according to incoming messages
- 🤔 Why not just implement a ROS 2 listener in the Unity app?



Head Animation Demo

Communication is also non-verbal



LOOMO ANDROID APPLICATION

...looks easy, is not

- Our hand:
 - Loomo: Android 5.1, Java SDKs ¹
 - Jetson Nano: Ubuntu 18/20,
Python Server, ROS
- First solution:
 - ROS or Socket communication ²
 - ros-java, loomo-ros-controller ³
- Upgrade to ROS2:
 - ros2-java ⁴
 - docker-build dependencies ⁵

mit-acl / android_loomo_ros_core

Public

forked from segway-robotics/android_core

Android app to enable ROS on a Loomo robot

MIT license

22 stars 19 forks

YasuChiba / ros2-android-build

Public



Build rcljava(ros2-java) for android.

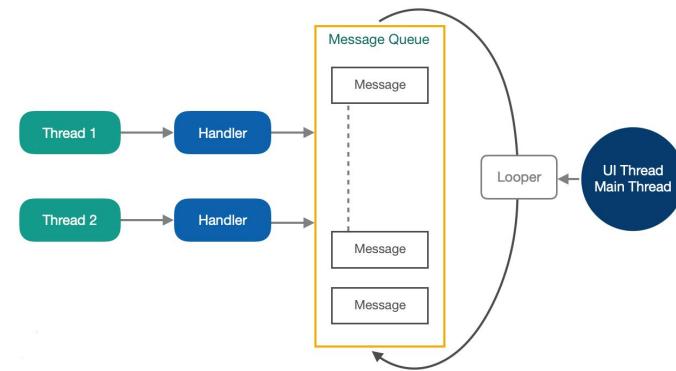
7 stars 5 forks



LOOMO ANDROID APPLICATION

...looks easy, is not

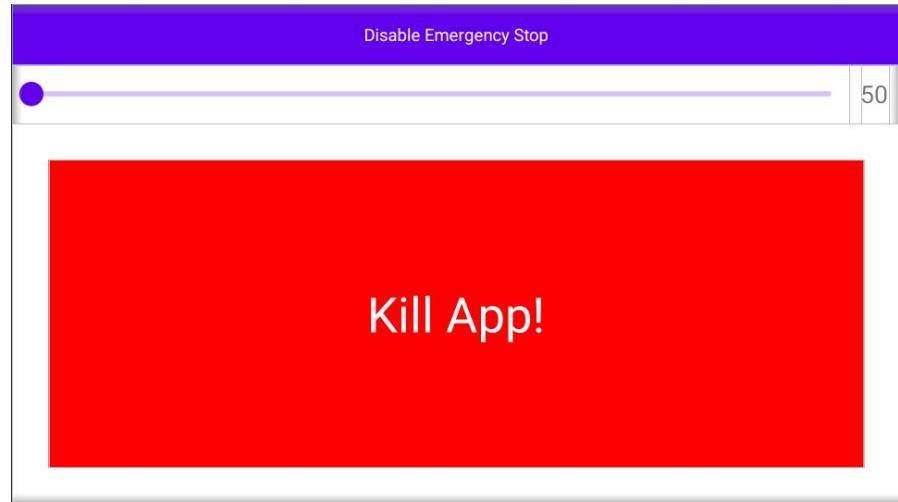
- Making use of Sensor Data
 - Android threading:
Handler, Looper, Message¹
- Project Management²
 - Open Source on Github
 - Releases at each milestone
 - Dependencies also released



LOOMO ANDROID APPLICATION

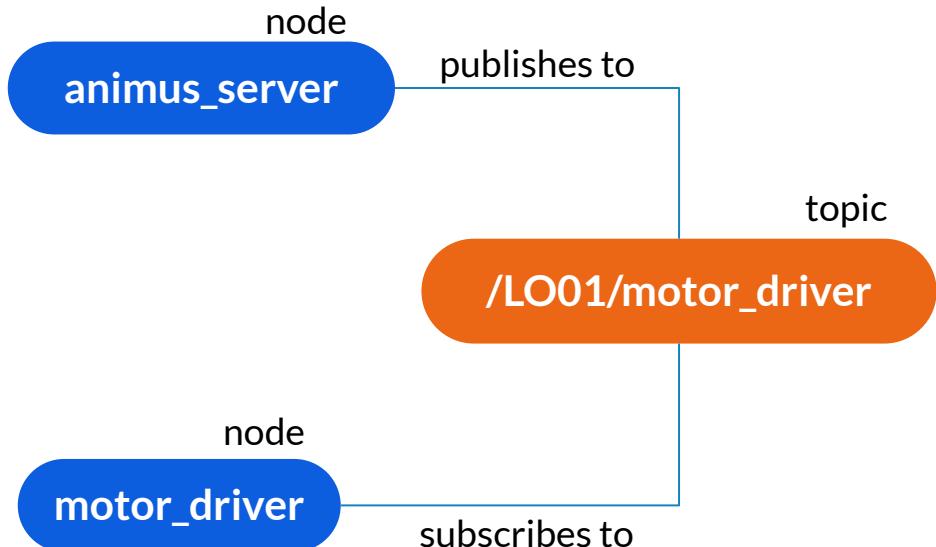
...looks easy, is not

- Safety Concerns
 - Obstacle Warning
 - Auto-Stopping
 - Kill App



HEIGHT ADJUSTMENT

... ever wanted to be very tall?



- Server receives motor command for head up/down
- Publishes to topic
- motor_driver receives the message
- Moves if buttons are not triggered



UNITY APPLICATION

and Animus Client



Basis:

- TeleportVR (Roboy3.0)
- with
RoboyAnimusClient.cs

Changes:

- 180° stereo vision
- Control to drive around
- Obstacle warning for the front
- Replacing the tutorial with an help button & loading screen



UNITY ANIMUS CLIENT

Controls and data



- Video-stream as .yuv
- Sound
- Proprioception matrix for the obstacle warning

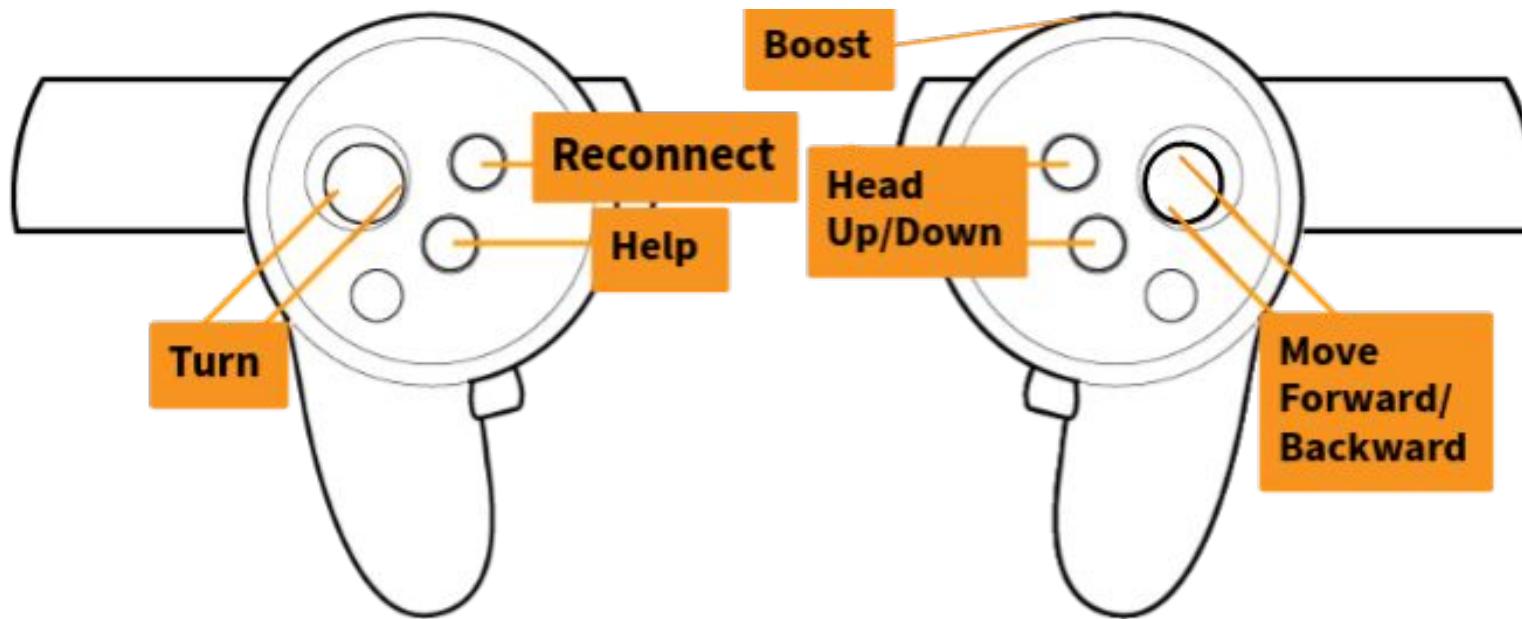
- Microphone sound
- Motor control matrix

```
778     // motorMsg.Data:  
779     // [0] -> time_stamp in h  
780     // [1] -> ang_vel  
781     // [2] -> lin_vel  
782     // [3] -> height_adjustment  
783     // [4] -> head_axisW  
784     // [5] -> head_axisX  
785     // [6] -> head_axisY  
786     // [7] -> head_axisZ  
787     // [8] -> booster
```



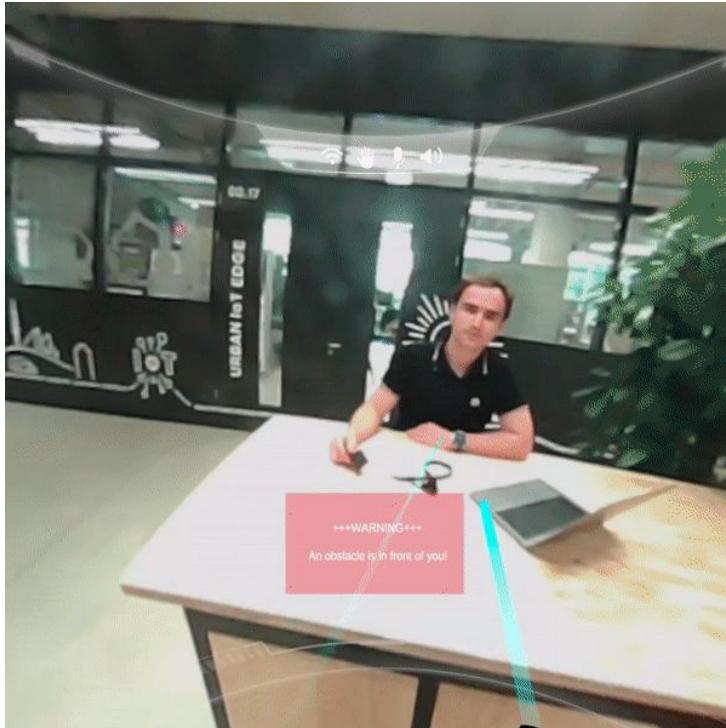
UNITY CONTROLS

your movements in the other end



UNITY OBSTACLE WARNING

for the front view



UNITY 180° STEREO VISION

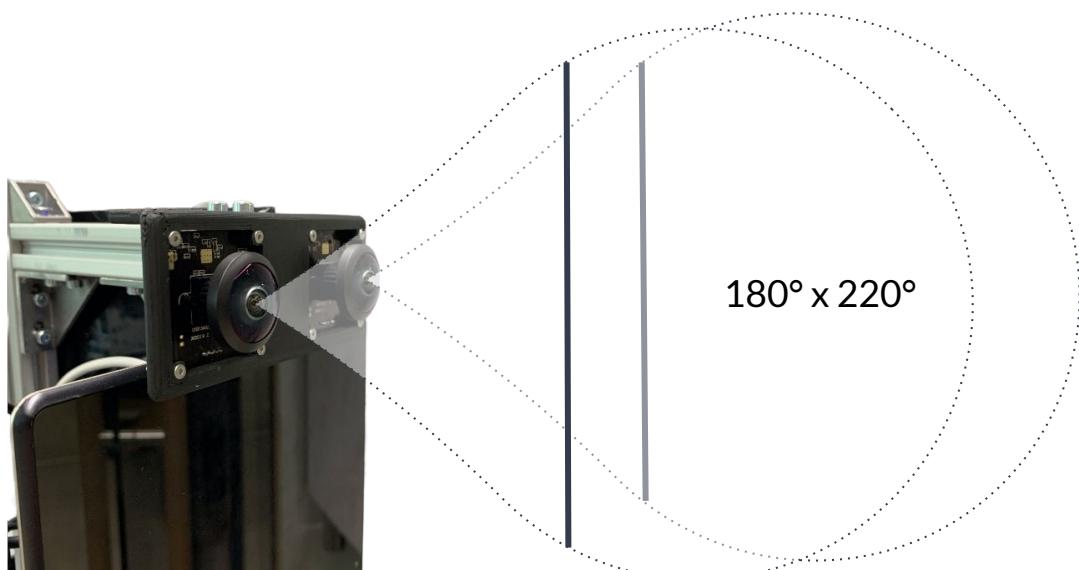
FOV of a human with poor eyesight

- Goal: 360° stereo vision
- Why:
 - Improvement of the sense of space
 - > Security
 - > localization
 - > Interaction



UNITY 180° STEREO VISION

FOV of a human with bad eyes



UNITY 180° STEREO VISION

FOV of a human with poor eyesight

Challenges:

- Resolution vs. Latency
- Shader & Shapes
- Price

Solution:

- Lower resolution -> lower latency
- Shapes as the lense of the cameras
- Specific shader in Unity
- 2x PiCam360



VR TUTORIAL

... which we deprecated in the meantime

Original:



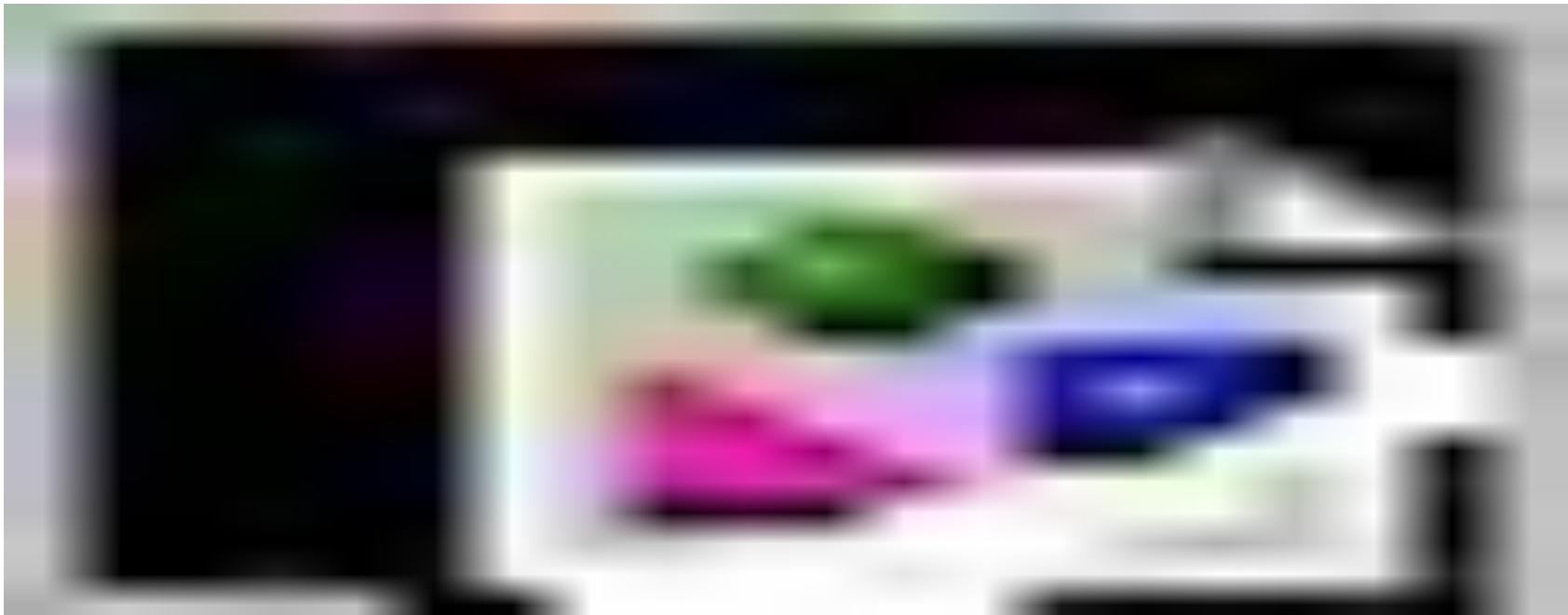
Final:



REDUCING VISION LATENCY

abandoned CV projects

- Solution 1: extract the FOV on the robot side and only send this
- **Issue:** hard to do in OpenCV (dewarping, perspective projection, etc.)



REDUCING VISION LATENCY

abandoned CV projects

- Solution 2: send 1 high-res (FOV) and 8 low-res patches
 - ~35 % improvement of end-to-end vision latency ($4.74\text{s} \rightarrow 3.1\text{s}$)
 - Issue: decompression algorithm in C# → basically just a time issue :(



VALIDATION AND TESTING



VALIDATION & TESTING

... of components

- Various tests
 - Network: Speed Tests (80-150mbps)
 - Modalities: Latency Tests
 - Vision: ~ 0.5s
 - Vision: Depth Perception Tests
- Human Evaluation
 - through us
 - and user feedback (at the midterm & “Festival der Zukunft”)



USER FEEDBACK

... an excerpt

- What needs to be improved in your opinion?

It would be great if the field of vision would move with my head movements.



Higher resolution of vision, listening could be improved to hear people not directly in front of the robot.
Improve listening to recognise where sound comes from



- What other functions do you expect in the next version?

It would be great that the robot displays my photo so others can identify me.



Moving robot extremities - 4, sound from 360 degrees - 6

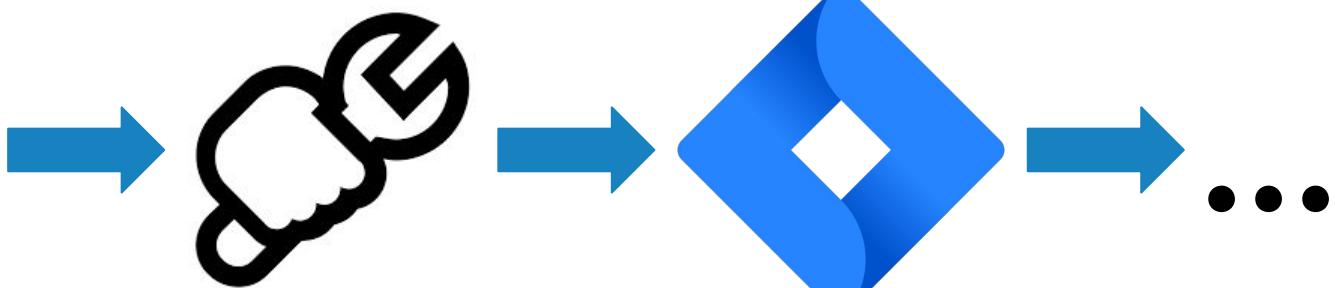


PLANNING PHASES

from kickoff to the final presentation



Planning Phase at
our Kickoff



Working Phase

Replanning Phase



PLANNED VS ACHIEVED

- Main goal:
 - Telepresence robot that can drive, see, communicate ✓
- Lost on the way due to refocus or difficulties:
 - Stereo vision AND control of field of view ✗
 - Tutorial game how to control Rollbody in Unity ✗
 - Tutorial Description how to control Rollbody in Unity ✓
 - Obstacle avoidance ✗
 - Obstacle Warning with Emergency stop ✓
 - Emotion control ✗
 - Voice ✓
 - Voice activated face ✓



THAT'S IT

... now test Rollbody yourself!

