



Abschlusspräsentation IV Visualizer

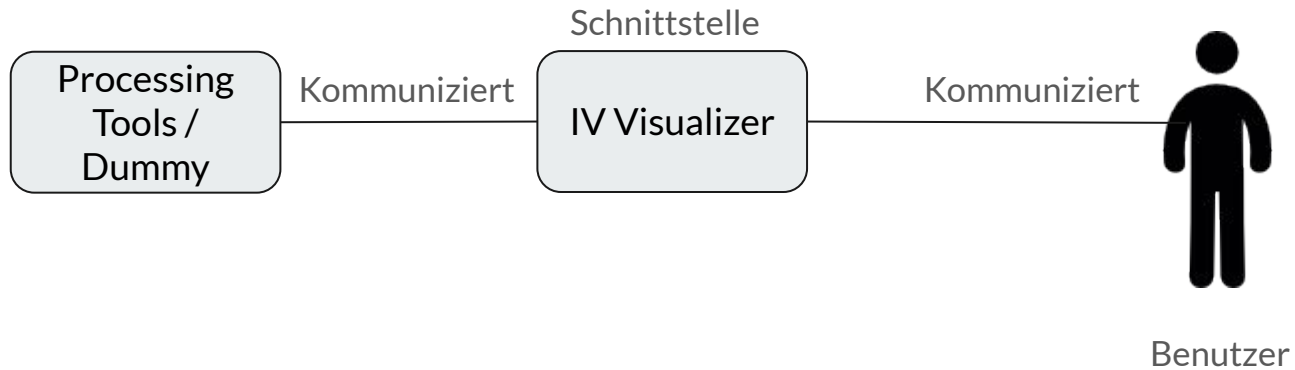
Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung IOSB, Josua Benjamin Eyl, Lukas Friedrich, Max Bretschneider, Nathaniel Till Hartmann, Robin Köchel

Betreut von: Mickael Cormier M.Sc., Stefan Wolf M.Sc

**“Visualisierung und Konfiguration
von verteilten KI-Verfahren
an einer zentralen Stelle”**

—

Aufbau



Anforderungen

- ✓ Laden der Videostreams von mehreren Processing Tools
- ✓ Prozessierung der Bilder mittels eines entfernten KI-Objekt Detektors
- ✓ Anzeigen der Ergebnisse im Visualizer
- ✓ Logs
- ✓ speichern und exportieren von Videos
- ✓ Konfiguration der Processing Tools
- ✗ erstellen von Diagrammen

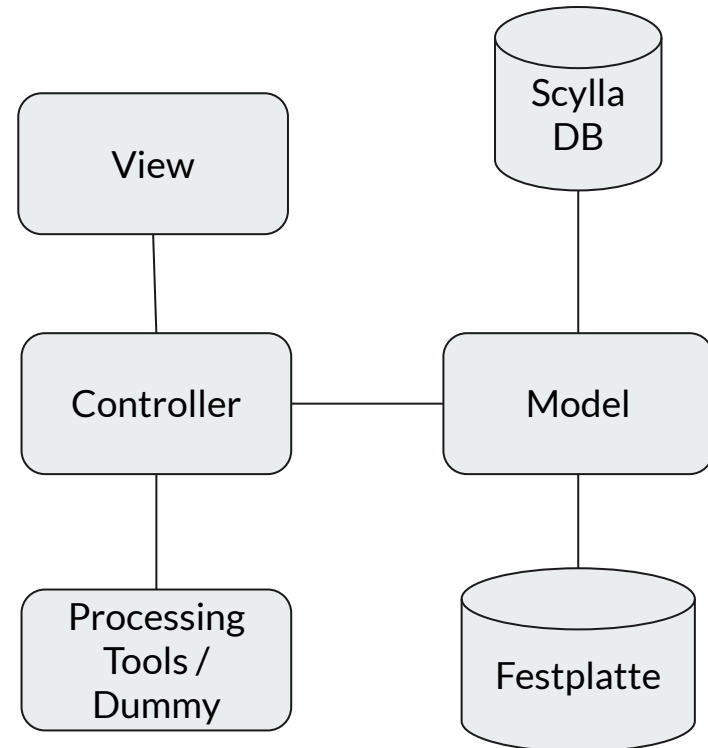
- ✓ Verwendung von C++ und Docker
- ✓ QT Framework für GUI
- ✓ erweiterbar
- ✓ skalierbar



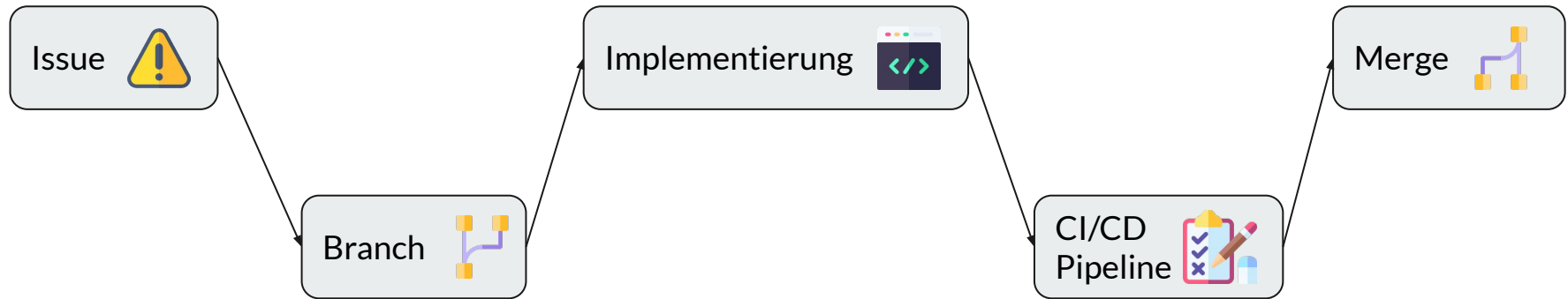
Modellierung der Software

Vorteile

- skalierbar
- erweiterbar
- austauschbar



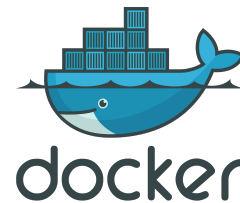
Implementierung & Qualitätssicherung





Tools & Tech Stack

- C++20
- CMake
- Docker compose
- Scylla Database





Tools & Tech Stack

- GitLab
- Valgrind
- Google Test Framework





Statistik

- Frontend:
 - Codezeilen: 7880
 - Davon Tests: 729
 - Anzahl Commits: 330
- Backend:
 - Codezeilen: 3115
 - Davon Tests: 364
 - Anzahl Commits: 291

Demo des IV Visualizers

Fragen?

Quellen:

https://upload.wikimedia.org/wikipedia/commons/1/18/ISO_C%2B%2B_Logo.svg

https://d1.awsstatic.com/acs/characters/Logos/Docker-Logo_Horizontal_279x131.b8a5c41e56b77706656d61080f6a0217a3ba356d.png

https://upload.wikimedia.org/wikipedia/commons/thumb/1/18/ISO_C%2B%2B_Logo.svg/1822px-ISO_C%2B%2B_Logo.svg.png

<https://upload.wikimedia.org/wikipedia/commons/thumb/1/13/Cmake.svg/2048px-Cmake.svg.png>

<https://cdn.worldvectorlogo.com/logos/docker.svg>

<https://www.scylladb.com/wp-content/uploads/brand-logo-vert@2x.png>

https://upload.wikimedia.org/wikipedia/commons/thumb/e/e1/GitLab_logo.svg/2560px-GitLab_logo.svg.png

https://valgrind.org/images/st-george_sm.png