

API-Demo

Florian

16. November 2014

1 Anwender API

- Funktionalität:
 - `debugDMatch(img1, keys1, img2, keys2, matches, ...)`
 - `debugFilter(orig, result, ...)`
 - `showImage(img, ...)`
- Meta/Boilerplate:
 - `finalShow()`
 - `FinalShowCaller` (RAII)
 - `CVVISUAL_LOCATION`
 - `setDebugFlag(bool)`
- Reinen Anwendern reicht das für alles!

2 Interne API

2.1 Allgemein

- Const-korrekt
- keine bekannten Leaks (Qt leakt immer -> Valgrind nutzlos)
- Relativ oft: `std::unique_ptr` statt besitzenden, blanken Pointern

2.2 `cvv::extend`

- `addCallType`: registriert eine Funktion die einen Tab zurück geben muss
- `addFilterView` und `addMatchView`: neue Visualisierungen für Filter und Matches
- `registerFilter`: Neue Filter für das `FilterSelectorWidget`

2.3 cvv::util

- **Reference:** Referenzen die ändern können, worauf sie zeigen; nicht besitzend
- **ObserverPtr:** Wie **Reference**, kann aber null sein
- **make_unique** und **makeRef**
- häufig benötigte Includes in **util.hpp**

2.4 Flexible Liniendicken

- „Problem“ zeigen
- Neuen Code zeigen
- Einfügen und in **init.cpp** registrieren
- „Lösung“ zeigen