CVVisual

Andreas Clara Erich Florian Johannes
Nikolai Raphael

20. Juni 2014

Gliederung

- Einführung in OpenCV
- Motivation
- Anwenderfeatures
- Gui-Demo
- Dokumentation
- Architektur
- API
- Ausblick

Einführung in OpenCV

Überblick

- Bildverarbeitung
- weite Verbreitung
- Matrizen als Grundlage
- Filter + Matches

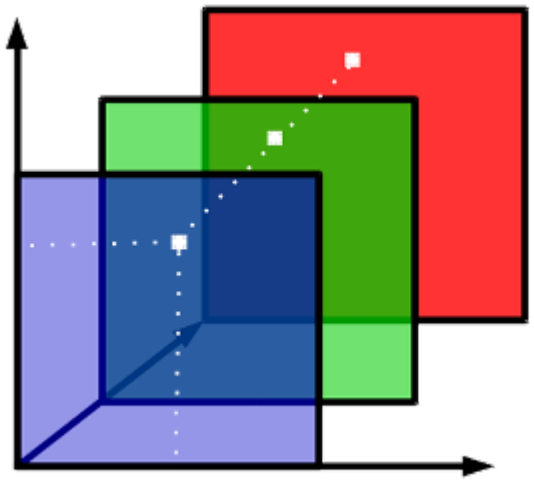
Matrizen

Bild = mehrdimensionale Matrix

Bsp. BGR-Bild: 1. Channel blau, 2. Channel grün usw.

Filter

Berechnung auf Umgebung jedes Pixels



5	7	3	5	5	5
3	2	6	7	6	5
2	3	2	4	6	6
3	3	5	6	4	5
1	4	6	2	2	4
3	4	7	5	6	5

Filter

Beispiel dilate: helle Flächen werden größer

Filter

Beispiel dilate: helle Flächen werden größer

Filter

Beispiel dilate: helle Flächen werden größer

Matches

Keypoints = charakteristische Punkte

Matches

Match = Paar aus Keypoints

Motivation

Debuggen von OpenCV

Systematisches Debugging statt „Random Code“

```
#ifdef DEBUG
        Mat img_matches;
        drawMatches( img_1, keypoints_1, img_2, keypoints_2,
                     good_matches, img_matches, Scalar::all(-1), Scalar::all(-1),
                     vector<char>(), DrawMatchesFlags::NOT_DRAW_SINGLE_POINTS );
        imshow("good matches", img_matches);
#endif
```

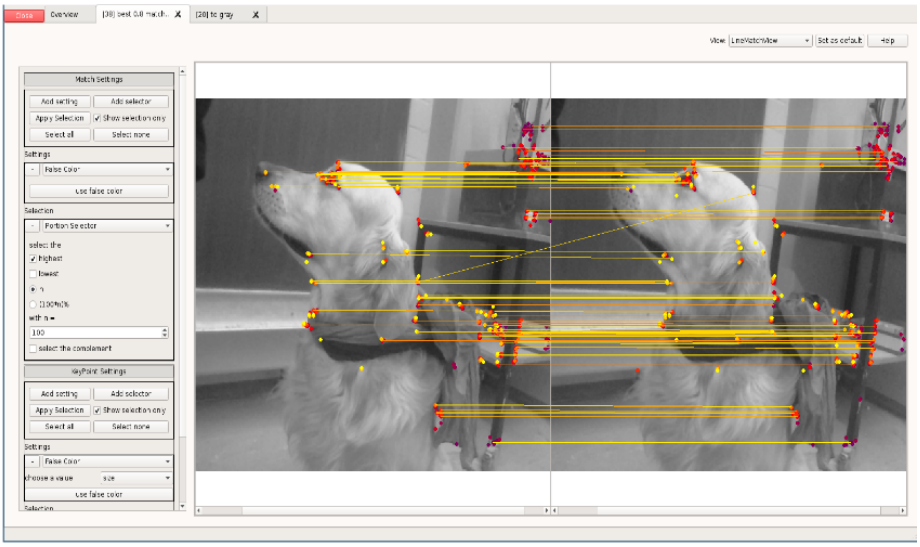
versus

```
cvv::debugMatches(img1, img2, keypoints_1, keypoints_2, good_matches);
```

Hinweis auf showMatches/showKeypoints

Ziele

Visualisierung von Matritzen, Filtereffekten und Matches



Anwenderfeatures

Verwendung

```
std::string imgIdString{"imgRead"};
imgIDString += toString(imgId);
cvv::showImage(imgRead, CVVISUAL_LOCATION, imgIdString);

// convert to grayscale:
cv::Mat imgGray;
cv::cvtColor(imgRead, imgGray, CV_BGR2GRAY);
cvv::debugFilter(imgRead, imgGray, CVVISUAL_LOCATION,
                 "to gray", "SingleFilterView");
```

Übersicht

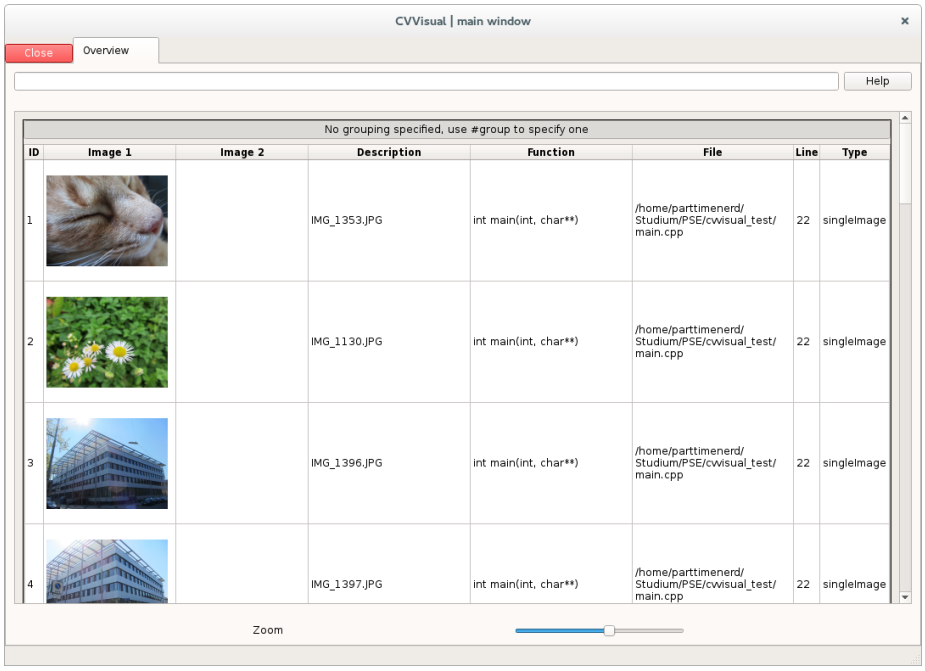
Übersicht über alle Aufrufe

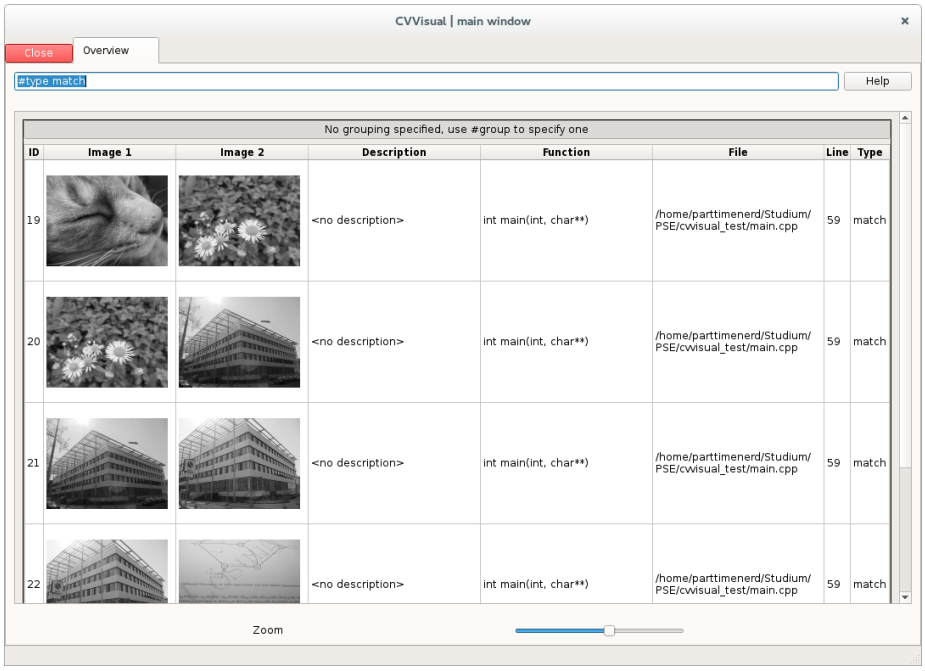
Übersicht

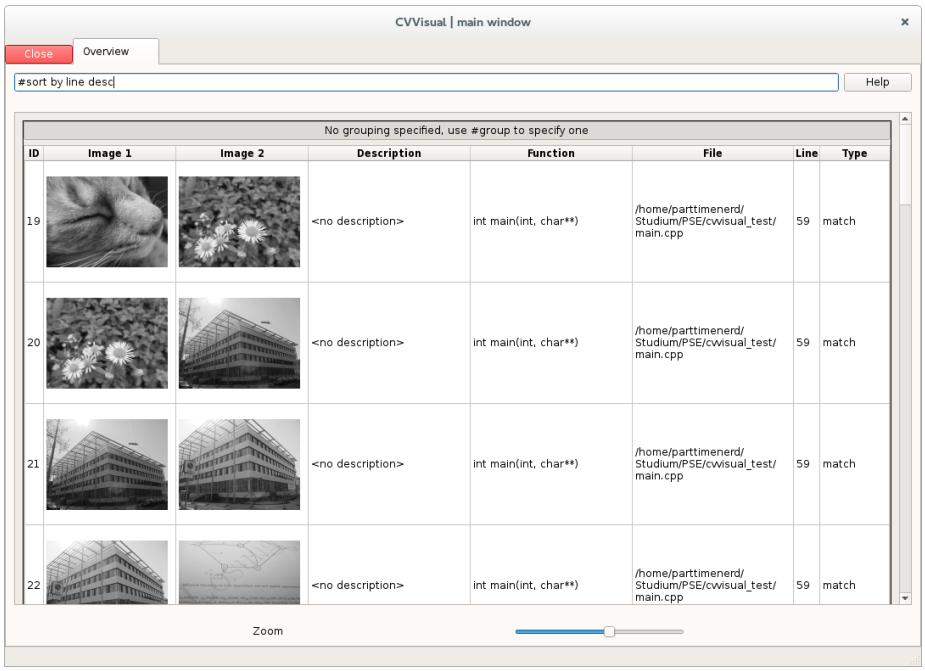
Filterbar

Übersicht

Sortierbar


CVVisual | main window
Close
Overview
Help
No grouping specified, use #group to specify one
ID
Image 1
Image 2
Description
Function
File
Line
Type
1
IMG_1353.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
22
singleImage
2
IMG_1130.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
22
singleImage
3
IMG_1396.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
22
singleImage
4
IMG_1397.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
22
singleImage
Zoom


CVVisual | main window
Close
Overview
#type match
Help
No grouping specified, use #group to specify one
ID
Image 1
Image 2
Description
Function
File
Line
Type
19
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
59
match
20
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
59
match
21
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
59
match
22
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvwisual_test/main.cpp
59
match
Zoom



Übersicht

Gruppierbar

Übersicht

Filter

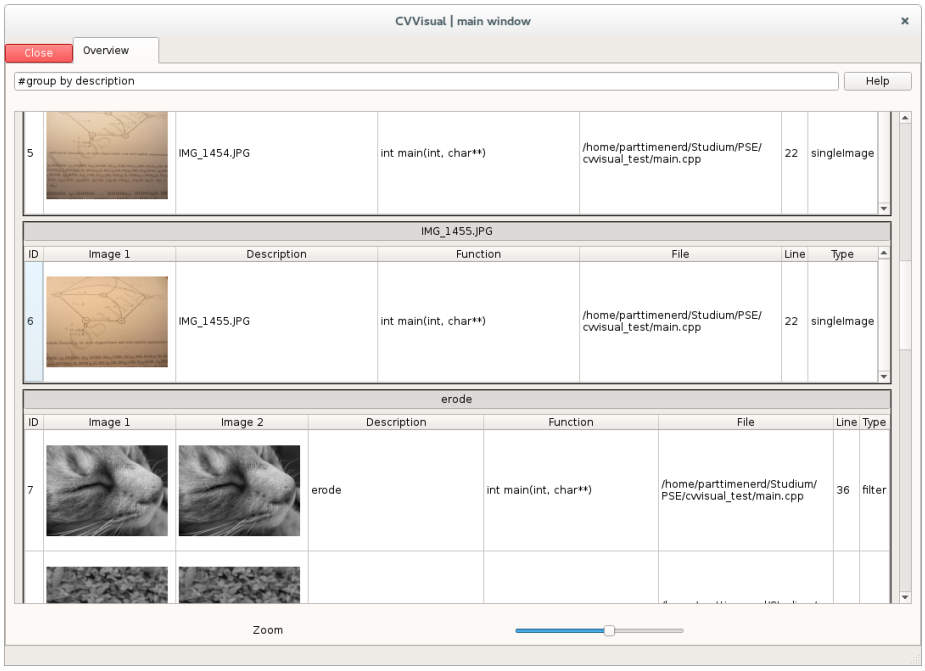
- 2 Bilder → 1 Bild
- Differenzbilder, Overlay, geänderte Pixel für Filter

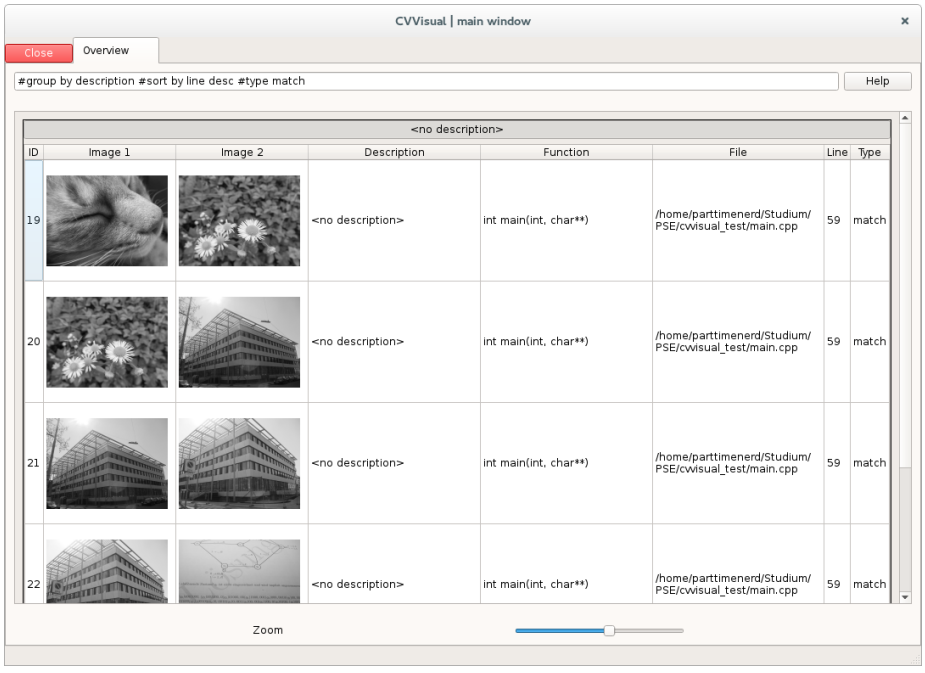
Filter

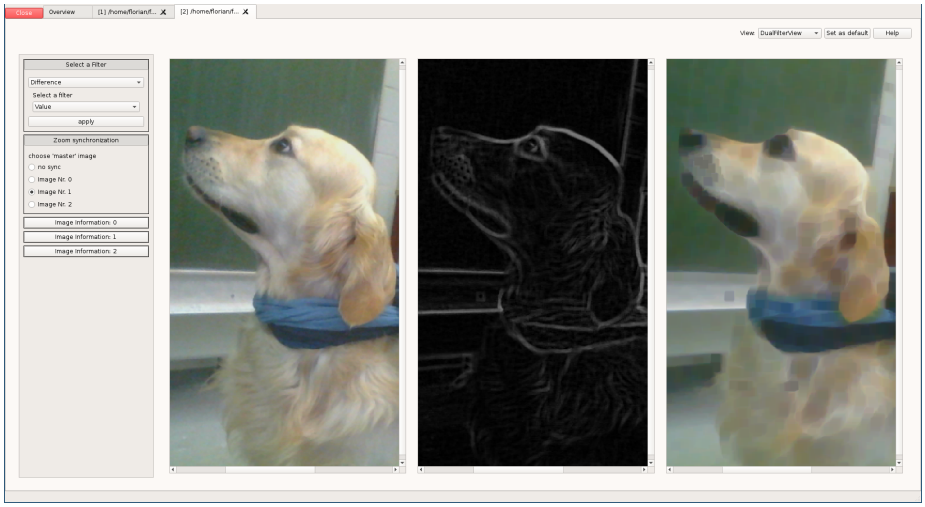
- 1 Bild → 1 Bild
- Nachträgliche Anwendung weiterer Filter

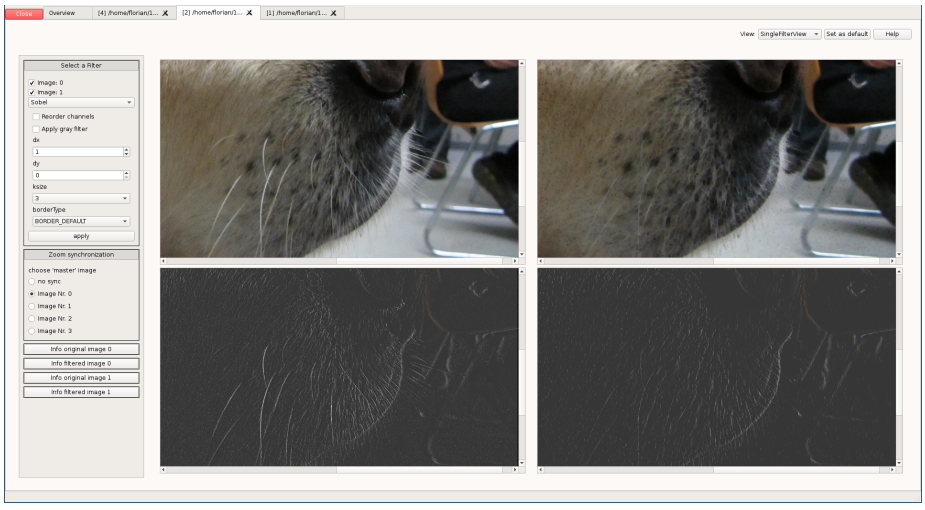
Matches

- Anzeigen / Filtern von Keypoints / Matches

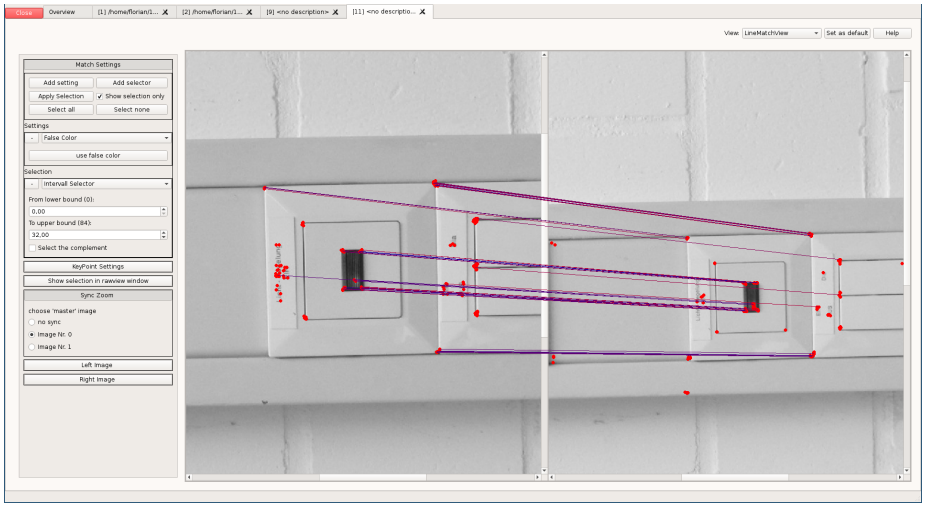

CVVisual | main window
Close
Overview
#group by description
Help
5
IMG_1454.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
22
singleImage
IMG_1455.JPG
ID
Image 1
Description
Function
File
Line
Type
6
IMG_1455.JPG
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
22
singleImage
erode
ID
Image 1
Image 2
Description
Function
File
Line
Type
7
erode
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
36
filter
Zoom


CVVisual | main window
Close
Overview
#group by description #sort by line desc #type match
Help
<no description>
ID
Image 1
Image 2
Description
Function
File
Line
Type
19
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
59
match
20
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
59
match
21
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
59
match
22
<no description>
int main(int, char**)
/home/parttimenerd/Studium/PSE/cvvisual_test/main.cpp
59
match
Zoom


Close
Overview
View
DualFilterView
Set as default
Help
Select a Filter
Difference
Select a filter
Value
apply
Zoom synchronization
choose 'master' image
no sync
Image Nr. 0
Image Nr. 1
Image Nr. 2
Image information: 0
Image information: 1
Image information: 2

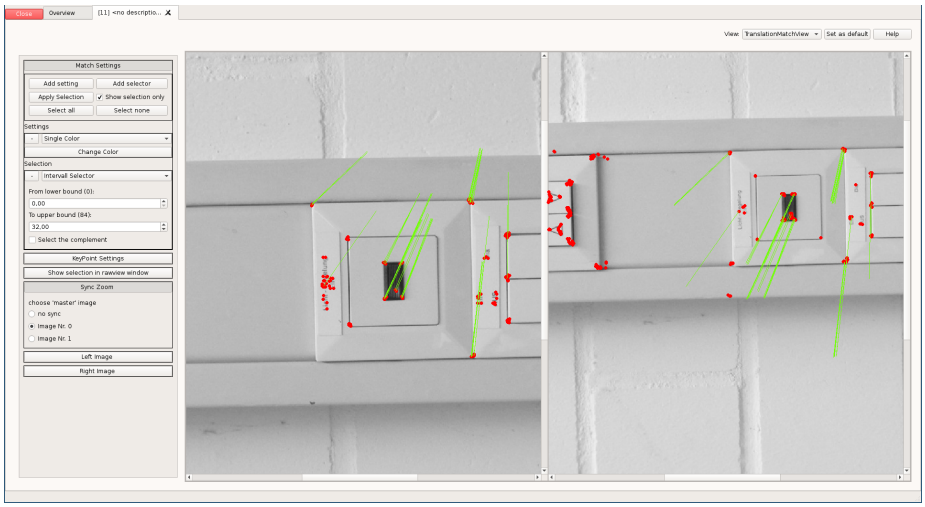

Close
Overview
View
SingleFilterView
Set as default
Help
Select a Filter
Image: 0
Image: 1
Sobel
Reorder channels
Apply gray filter
dx
1
dy
0
ksize
3
borderType
BORDER_DEFAULT
apply
Zoom synchronization
choose 'master' image
no sync
Image Nr. 0
Image Nr. 1
Image Nr. 2
Image Nr. 3
Info original image 0
Info filtered image 0
Info original image 1
Info filtered image 1

- Anzeige der Verbindungen von Keypoints



Matches

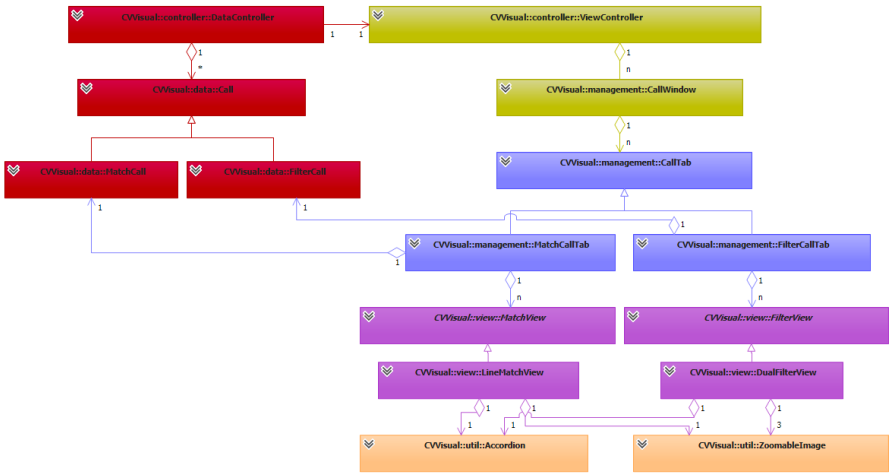
- Anzeigen / Filtern von Keypoints / Matches
- Anzeige der Translation von Keypoints



GUI-Demo

Architektur

Entwurf



Signals/Slots & Templates

```
class SlotQString : public QObject
{   Q_OBJECT
public:
    SlotQString(const std::function<void(QString)> &f,QObject *parent = nullptr)
        : QObject{ parent },function_{f}
    {   if (!f)
            throw std::invalid_argument{ "invalide function" };
    }
public slots:
    void slot(QString t) const
        {function_(t);}
    private:
    std::function<void(QString)> function_;
};
```

RegisterHelper

- Ermöglicht die Auswahl von Funktionen über eine Combobox

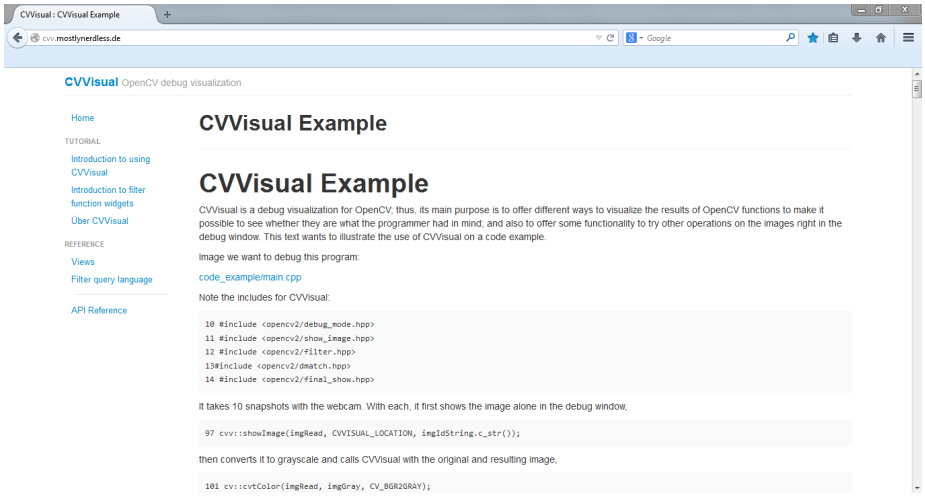
- Funktionen werden über eine API Funktione registriert
- Wird in der API Demo vorgestellt

```
cvv::qtutil::registerMatchSettings<cvv::qtutil::SingleColorMatchPen>("Single Color");
```

```
template <class Setting>
bool registerMatchSettings(const QString &name)
{
    return MatchSettingsSelector::registerElement(
        name, [](std::vector<cv::DMatch> univers)
    {
            return std::unique_ptr<MatchSettings>{ new Setting{univers}};
    });
}
```

Dokumentation

Tutorials, Beispiele

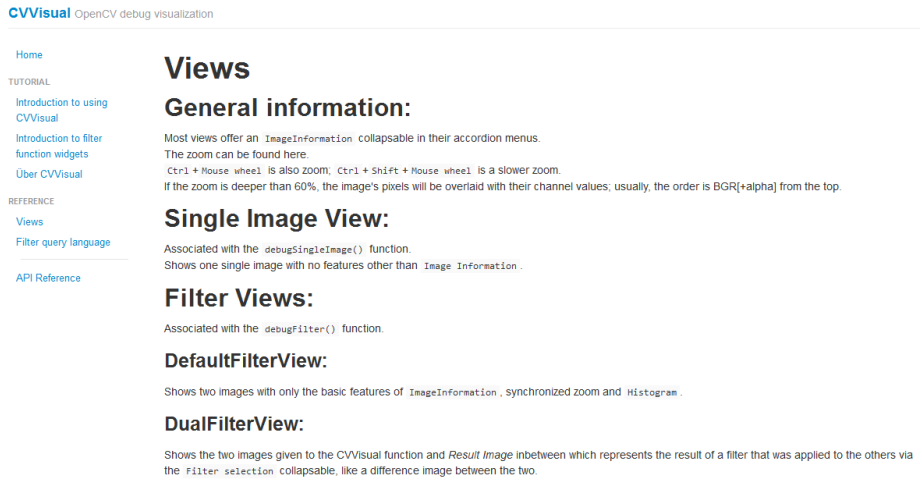


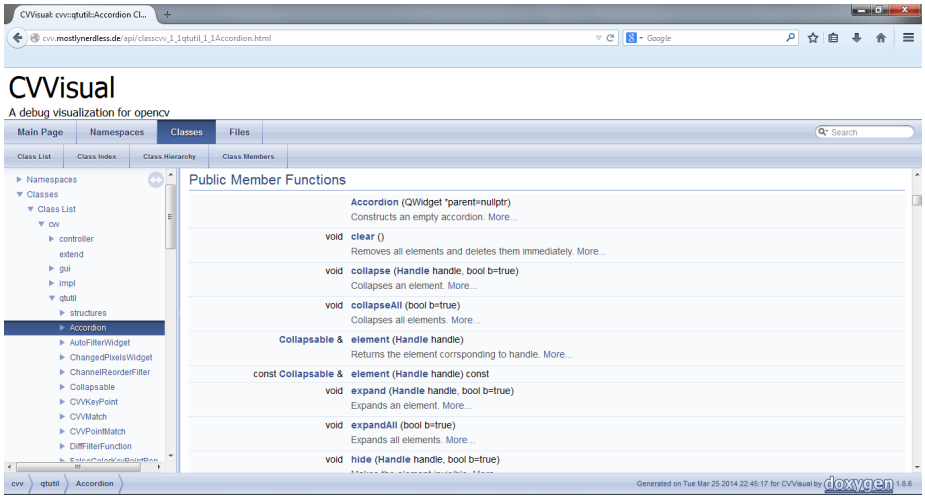
Kurzdokumentation

Wird von der Hilfefunktion des Programms benutzt.

Referenz:

- Mit Hilfe von Doxygen





API

Anwender API

- Triviale Benutzung auch in C++98
- Sehr klein und übersichtlich

Interne API

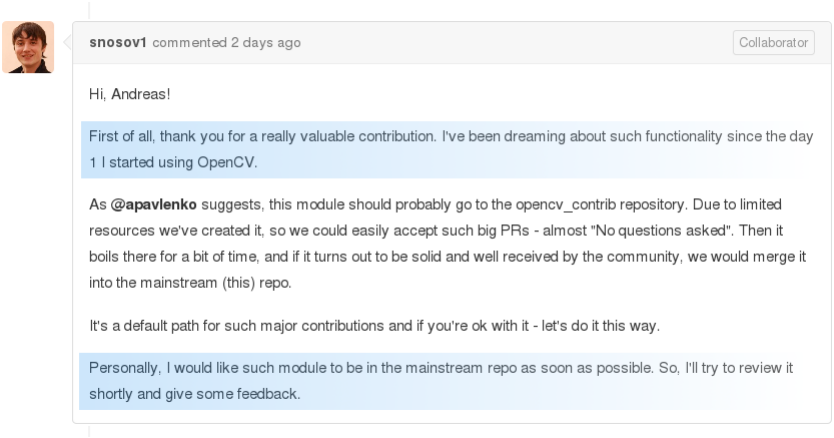
- Erweiterung über Funktionen in `cvv::extend`

- Leichtes, zentralisiertes Hinzufügen von Visualisierungen, Filtern, Views,. . .

Ausblick

Rezeption

Projekt schien von der OpenCV-Community wohlwollend aufgenommen zu werden



snosov1 commented 2 days ago — Collaborator

Hi, Andreas!

First of all, thank you for a really valuable contribution. I've been dreaming about such functionality since the day 1 I started using OpenCV.

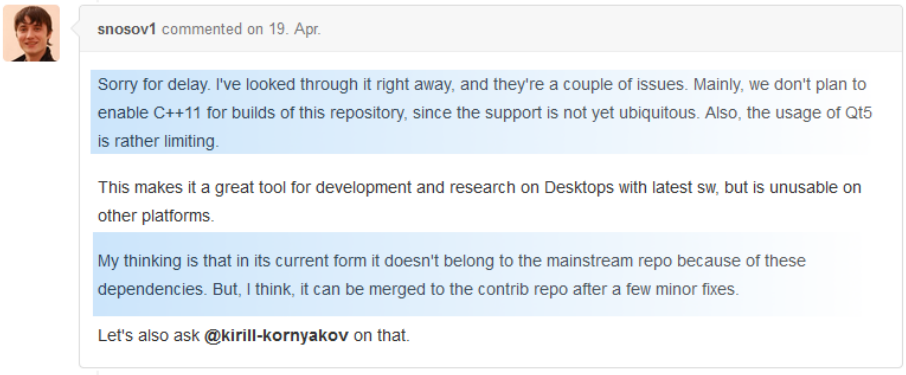
As **@apavlenko** suggests, this module should probably go to the opencv_contrib repository. Due to limited resources we've created it, so we could easily accept such big PRs - almost "No questions asked". Then it boils there for a bit of time, and if it turns out to be solid and well received by the community, we would merge it into the mainstream (this) repo.

It's a default path for such major contributions and if you're ok with it - let's do it this way.

Personally, I would like such module to be in the mainstream repo as soon as possible. So, I'll try to review it shortly and give some feedback.

Rezeption

Nach aktuellem Stand aber aufgrund C++11 und Qt5 keine Aufnahme ins Haupt-Repo



snosov1 commented on 19. Apr.

Sorry for delay. I've looked through it right away, and they're a couple of issues. Mainly, we don't plan to enable C++11 for builds of this repository, since the support is not yet ubiquitous. Also, the usage of Qt5 is rather limiting.

This makes it a great tool for development and research on Desktops with latest sw, but is unusable on other platforms.

My thinking is that in its current form it doesn't belong to the mainstream repo because of these dependencies. But, I think, it can be merged to the contrib repo after a few minor fixes.

Let's also ask **@kirill-kornyakov** on that.

Links

- Github: https://github.com/CVVisualPSETeam/CVVisual/
- Dokumentation: https://cvv.mostlynerdless.de/
- Doxygen: https://cvv.mostlynerdless.de/api/