

Projekt Ausarbeitung

Kamerakalibrierung anhand eines Punktgitters

geschrieben von

Vera Brockmeyer (Matrikelnr. 11077082)
Artjom Schwabski (Matrikelnr. 11113320)

Weiterführende Themen der Bildverarbeitung in SS 2017

Betreuer:

Prof. Dr. Dietmar Kunz
Institute for Media- and Phototechnology

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Abstrakt | 3 |
| 2 | Einleitung | 4 |
| 2.1 | Motivation | 4 |
| 3 | Stand der Wissenschaft | 5 |
| 3.1 | Kamerakalibrierung mit Punktraster | 5 |
| 4 | Materialien | 6 |
| 4.1 | Hardware | 6 |
| 4.2 | Software | 6 |
| 4.2.1 | Klassen | 6 |
| 4.2.2 | ImageJ | 8 |
| 5 | Methode | 9 |
| 5.0.3 | Levenberg-Marquard-Approximation | 9 |
| 5.0.4 | Affine Transformation | 9 |
| 5.0.5 | Kamerakalibrierung | 9 |
| 6 | Auswertung | 10 |
| 7 | Reflexion | 11 |
| 8 | Zusammenfassung | 12 |

1 Abstrakt

XXXX

2 Einleitung

Vera

2.1 Motivation

Vera

3 Stand der Wissenschaft

???

3.1 Kamerakalibrierung mit Punktraster

XXXX

4 Materialien

XXX

4.1 Hardware

4.2 Software

4.2.1 Klassen

Artjom

Im folgenden werden die Methoden der einzelnen Klassen erläutert. Die vollständige UML zur besseren Verständlichkeit der Klassenbeziehungen ist der Abb. 1 zu entnehmen.

point_grid_radial_affin_distor_ Hauptklasse der Anwendung. Implementiert das Interface *PluginFilter* um über ImageJ aufgerufen werden zu können.

Die Klasse besitzt folgende Methoden und deren Funktion:

| | |
|---------------------------------|--|
| run | Main-Methode des PlugIns in der die Optimierung aufgerufen wird |
| setup | Konstruktor-Methode des PlugIns in dem die Bildreferenz gespeichert wird |
| readData | Liest aus einer in ImageJ geöffneten Textdatei Punkt-Paare ein für Start- und Ziel-Koordinaten |
| computeDrawRadialTransformation | |
| drawTargets | Zeichnet Punkte an den übergebenen Ziel-Koordinaten in das übergebene Bild |
| computeDrawAffineTransformation | |
| computeRadius2Center | Berechnet anhand der Parameter den Abstand zum Gittermittelpunkt |
| compute_radial_dist_coef | Berechnet mit dem LevenbergMarquadt Optimierer die Koeffizienten der Radialen Verzerrung der übergebenen Punkt und gibt die Koeffizienten zurück |

Tabelle 1: Methoden der point_grid_radial_affin_distor_ Klasse

SimplePair Eine Einfache Klasse zum Speichern der Vorgabe- und Ziel Koordinaten und des Abstandes zum Mittelpunkt.

RadialDistFunction Klasse zum Erzeugen der Funktionen für den Optimierer.

| | |
|--------------------|--|
| RadialDistFunction | Konstruktor der Klasse. Es wird ein SimplePoint Array erwartet welcher Koordinaten-Paare für Start- und Ziel-Koordinaten enthält. |
| realTargetPoints | Gibt ein Array aus welches nur die Ziel-Koordinaten enthält. Dieses wird für den Optimierer benötigt. |
| retMVF | Funktion zur Modellierung der Radialen Verzerrung für den Optimierer. Berechnet zu den Vorgegeben Koeffizienten und einer Start-Koordinate die Ziel-Koordinate |
| retMMF | Jacobi-Matrix-Funktion zur Berechnung der Ableitung nach den einzelnen vom Optimierer vorgegebenen Koeffizienten |

Tabelle 2: Methoden der RadialDistFunction Klasse

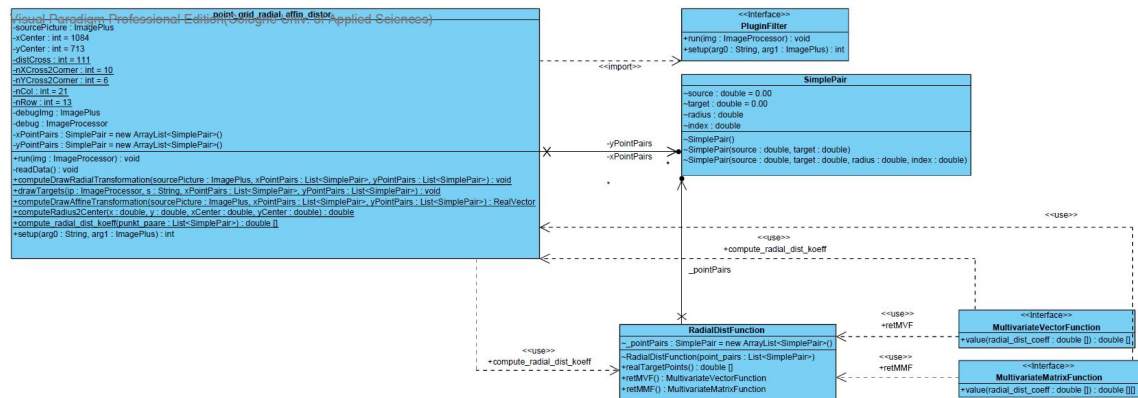


Abbildung 1: UML Klassendiagramm

UnwrapJ ist ein für ImageJ entwickeltes Plugin, das die elastische Registrierung von zwei Bildern ermöglicht, indem es ein Quellbild verformt, so dass es einem Zielbild ähnelt. Es stehen drei Betriebsarten zur Verfügung:

1. ein vollautomatischer Modus;
2. ein vollständig interaktiver Modus, bei dem die Verformung durch die Position einer beliebigen Anzahl von Landmarken eindeutig bestimmt ist;
3. ein gemischter Modus, bei dem interaktive Landmarken nur verwendet werden, um eine ansonsten automatische Registrierungsprozedur anzuzeigen.

Das Deformationsmodell besteht aus kubischen Splines, die Glätte und Vielseitigkeit gewährleisten. Das Registrierungskriterium enthält einen Vektor-Spline-Regularisierungstermin, um die Deformation physisch realistisch zu beschränken.[1]

In dieser Anwendung wird es jedoch nicht zur Registrierung sondern nur zum erzeugen von Landmarken genutzt. Diese werden in eine Textdatei gespeichert, welche im programmierten Plug-In eingelesen und verwendet wird.

4.2.2 ImageJ

5 Methode

XXXX

5.0.3 Levenberg-Marquard-Approximation

5.0.4 Affine Transformation

5.0.5 Kamerakalibrierung

6 Auswertung

XXXX

7 Reflexion

Vera

8 Zusammenfassung

Vera

Literatur

- [1] Biomedical Imaging Group. Unwarpj: An imagej plugin that performs a spline-based elastic registration of two images., Sept. 2017.