

## ÜBUNG 4: AUTOMATISCHE REGISTRIERUNG ZWEIER BILDER MIT SIFT UND RANSAC

### Praktikum Computer Vision

#### 1. Direkte Lineartransformation

- a. Implementieren Sie die direkte Lineartransformation zur Schätzung einer Homographie zwischen zwei Bildern eines Panoramas. Normieren Sie Ihre Homographie immer so, dass der letzte untere Eintrag in der Homographie-Matrix 1 ist.
- b. Wählen Sie aus einem überlappenden Bildpaar (am Besten aus Übung 3) mindestens 6 Korrespondenzpunkte per Hand aus, d.h. bestimmen Sie wie in Übung 3 die Pixelposition der korrespondierenden Punkte in beiden Bildern. Nehmen Sie 4 Korrespondenzen zur Schätzung der Homographie per DLT und testen Sie anhand der anderen Korrespondenzen die Korrektheit Ihrer Homographie, indem Sie mithilfe der Homographie die Bildposition des Punktes im neuen Bild berechnen und dann den mittleren euklidischen Abstand zu dem korrespondierenden Punkt bestimmen (diese Routine wird später noch für RANSAC gebraucht).
- c. Aufbauend auf Übung 1, schreiben Sie eine Routine zur indirekten Umbildung eines Bildes mithilfe einer Homographie und testen Sie sie anhand der geschätzten Homographie aus Aufgabe 1 b. Schreiben Sie ein Programm zur Registrierung zweier Bilder.
- c. Implementieren Sie nun die normierte DLT wie in Vorlesung 5 beschrieben. Wiederholen Sie alle vorigen Tests für die normierte DLT.

#### 2. RANSAC

- a. Implementieren Sie den RANSAC-Algorithmus für die normierte DLT. Dazu wählt der Algorithmus aus einem Satz von Korrespondenzen zufällig eine Vierergruppe aus, berechnet die Homographie mithilfe der normierten DLT und berechnet den mittleren euklidischen Abstand der übrigen Korrespondenzen.
- b. Ergänzen Sie Ihre Trainingsdaten aus Aufgabe 1 um mehrere Fehlkorrespondenzen und testen Sie Ihre Implementierung von RANSAC. Kommt die korrekte Homographie heraus? Wurde ein korrektes *Consensus set* gefunden?

### 3. Automatisiertes Matching mit RANSAC

a. Falls noch nicht geschehen, installieren Sie OpenCV. Laden Sie das auf der Vorlesungsseite zur Verfügung gestellte Modul *sift.py* und installieren Sie es im selben Verzeichnis, in dem auch Ihr Code liegt. Konvertieren Sie Ihre Bilder in Grauwertbilder mit `cv2.cvtColor(im1,cv2.COLOR_BGR2GRAY)`. Das Extrahieren der *interest points* und der Deskriptoren geschieht durch folgenden Code:

```
import cv2, sift

# .. Bild einlesen und konvertieren
im = cv2.cvtColor(im,cv2.COLOR_BGR2GRAY)

location_list, descriptor_list = sift.detect_and_compute(im)
```

Das Matching, d.h. das Auffinden korrespondierender Punkte mit ähnlichem SIFT-Deskriptor und Darstellung geschieht über

```
matches = sift.match_twosided(descriptor_list_img1,descriptor_list_img2)
sift.plot_matches(im1,im2, location_list_img1, location_list_img2,matches,True)
```

Testen Sie Ihren Code an dem Bildpaar aus Aufgabe 1 und 2. Wenn die Berechnung von SIFT und Matching korrekt funktioniert, können Sie nun automatisch eine Liste von Korrespondenzen erstellen.

b. Testen Sie Ihre automatisch gefundenen Korrespondenzen mit RANSAC (500–1000 Iterationen) und normierter DLT. Kommt die gleiche oder zumindest eine ähnliche Homographie heraus? Welchen mittleren quadratischen Fehler erhalten Sie? Erstellen Sie ein Programm zur automatischen Registrierung zweier Bilder und testen Sie dieses an allen Bildpaaren aus Übungsblatt 3.