

2022

Praktikum Bildverarbeitung

Aufgabenblatt 1

Rotation von Bildern

Anforderungen:

- Die Aufgabe wird in Python programmiert.
- Die Aufgabe wird von jedem Studierenden einzeln erstellt!
- Der Teilnehmer kommt zur Abnahme auf den Dozenten zu. Die Abnahme erfolgt für jeden Teilnehmer einzeln. Die Kenntnis des Quellcodes wird erwartet.
- Programmcode wird auf Ilias hochgeladen. Die Lokation wird im Praktikum bekanntgegeben. Das File hat folgendes Format:
 - `<Name>_<Vorname>_<Matrikelnummer>_Aufgabe_1.py`
- **Die Frist für die Abnahme und das Hochladen der Files wird im Praktikum bekanntgegeben.**
- **Die hochgeladenen Files werden nach der Frist nochmals kontrolliert. Erst nach dieser Kontrolle gilt die Aufgabe als vollständig bestanden.**

Einleitung

Zur Veranschaulichung von Funktionalitäten in der Bildverarbeitung soll die Bibliothek OpenCV verwendet werden. Eine Beschreibung von OpenCV findet sich in [1]. Ein Tutorial gibt es in [2]. In der Bildverarbeitung wird häufig Ausschnitte eines vorhandenen Bildes benötigt. Diese Ausschnitte müssen dabei in einem bestimmten Format und Größe vorliegen. Innerhalb von Python/OpenCV verwendet man dafür n-dimensionale Arrays aus der Bibliothek numpy (numpy.ndarray). Unter Verwendung von Formaten wie z.B. jpeg, png, jfif oder gif soll ein Bild mit `cv2.imshow()` aus der Python/OpenCV Bibliothek angezeigt werden.

Weiter kommen in der Bildverarbeitung oftmals Transformationen zum Einsatz. Ein Beispiel dafür ist die Rotation. Ein Bild wird hier um einen Punkt mit einem Winkel rotiert. Dabei fällt auf, dass die Ränder, bzw. Ecken des Bilds nach der Rotation schwarz erscheinen. Diese schwarzen Bereiche sind u.U. nicht gewollt und können z.B. über das Herauskopieren eines rechteckigen Bereiches aus einem größerem Original-Bild vermieden werden.

OpenCV enthält u.a. eine Library [3], um Ausschnitte von Gesichtern und Augen zu erkennen. Die Funktionen liefern dabei Rechtecke oder Punkte zurück, um die Gesichter und Augen im Bild anzuzeigen.

Aufgabe

Es soll eine Anwendung programmiert werden, die ein Bild eines Gesichts um einen Winkel rotiert. Es soll dabei beachtet werden, dass keine schwarzen Ränder entstehen. Nach dem Laden des Bilds soll der Benutzer die Augen des Gesichts anklicken, und dabei sollen rote Punkte erscheinen. Die Anwendung soll ein Winkel zwischen der Gerade entlang der Augen und der x-Achse berechnen. Das Bild des Gesichts soll um diesen Winkel rotiert werden, damit die Gerade entlang der Augen parallel zur x-Achse befindet. Koordinaten des Mauszeigers und Mouse-Clicks können mit `setMouseCallback()` [5] ermittelt werden.

Die User Stories sind in der folgenden Tabelle aufgelistet:

Als	will ich	damit
Teilnehmer	selbstständig alle erforderlichen Libraries installieren	der Teilnehmer diese und die folgenden Praktikumsaufgaben bearbeiten kann.
Teilnehmer	selbstständig alle erforderlichen Libraries und Funktionalitäten studieren	er seine Expertise mit Bildverarbeitung erweitern kann.
Anwendung	ein Bild eines Gesichts mit <code>cv2.imshow()</code> darstellen	der Benutzer das Bild weiterverarbeiten kann.
Benutzer	auf beide Augen des Bilds klicken	die Anwendung rote Kreise auf den Augen darstellt.
Anwendung	das Bild um einen Winkel rotieren	die Gerade entlang der Augen parallel zur x-Achse sich befindet.
Anwendung	nur einen Ausschnitt des Bildes entnehmen	keine schwarzen Ränder sichtbar sind.
Benutzer	die Taste „s“ bedienen	die Anwendung das rotierte Bild als neues Bild auf der Festplatte speichert.
Benutzer	die Taste „q“ bedienen	sich die Anwendung schließt.

Tabelle: User Stories

Links

- [1]: <https://docs.opencv.org/3.0-beta/index.html>
- [2]: https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html
- [3]: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html
- [4]: <https://github.com/opencv/opencv/tree/master/data/haarcascades>
- [5]: https://docs.opencv.org/3.4/db/d5b/tutorial_py_mouse_handling.html