

2021

Praktikum Bildverarbeitung

Aufgabenblatt 1

Kernfunktionalitäten in der Bildverarbeitung

Anforderungen:

- Die Aufgabe wird in Python programmiert.
- Die Aufgabe wird von jedem Studierenden einzeln erstellt!
- Der Studierende kommt zur Abnahme auf den Dozenten zu. Die Abnahme erfolgt für jeden Studierenden 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 oder gif soll ein Bild mit cv2.imshow() aus der Python/OpenCV Bibliothek angezeigt werden.

Weiter kommen in der Bildverarbeitung oftmals Transformationen zum Einsatz, um die Perspektive des Bildes zum Betrachter zu verändern. Ein Beispiel dafür ist die perspektivische Projektion. Reale Bilder von z.B. rechteckigen Objekten sind in den meisten Fällen nicht aus der Vogelperspektive aufgenommen, sondern aus einer beliebigen Richtung. Dadurch sind die realen parallelen Kanten eines Rechtecks auf den Bildern nicht parallel dargestellt. Werden die Kanten verlängert, dann treffen diese auf einen sogenannten Fluchtpunkt zusammen. OpenCV bietet die Funktion cv2.warpPerspective [5] an, um Objekte von einer Raute in ein Rechteck zu transformieren. Für die Transformation wird eine Matrix benötigt, welche durch die Funktion cv2.getPerspectiveTransform [5] erzeugt werden kann.

Aufgabe

Es soll eine Anwendung programmiert werden, die ein frei verfügbares Bild mit Wally [7], unter Berücksichtigung der Urheberrechte, ausliest und dann mit Hilfe von Tastenkommandos weiterverarbeitet.

Nachdem Laden des Bilds soll der Benutzer die Ecken eines Rechtecks um den Kopf von Wally mit der Maus anklicken. Nach jedem Klick erscheint dann ein roter Kreis. Koordinaten des Mauszeigers können mit `setMouseCallback()` [6] ermittelt werden. Nachdem der Benutzer auf alle viermal geklickt hat, speichert die Anwendung die vier Koordinaten in eine Polygon-Variable. Die vier Koordinaten sollten dann ein Polygon formen, und kein Rechteck. Die Anwendung transformiert den Bildausschnitt, repräsentiert durch das Polygon, auf ein Rechteck, welches als neues Bild dem Benutzer dargestellt wird. Der Benutzer speichert das Bild, und kann ein neues Bild mit Mausklicks erzeugen, oder auch die Anwendung schließen.

Die User Stories sind in der folgenden Tabelle aufgelistet:

Als	will ich	damit
Studierender	selbstständig alle erforderlichen Libraries installieren	der Studierende diese und die folgenden Praktikumsaufgaben bearbeiten kann.
Studierender	selbstständig alle erforderlichen Libraries und Funktionalitäten studieren	er seine Expertise mit Bildverarbeitung erweitern kann.
Anwendung	ein Bild mit <code>cv2.imshow()</code> darstellen	der Benutzer das Bild weiterverarbeiten kann.
Benutzer	die viermal um den Kopf von Wally auf dem Bild anklicken	die Anwendung rote Kreise darstellt und die Koordinaten der roten Kreise in ein Polygon speichert.
Anwendung	das gespeicherte Polygon mit vier Ecken perspektivisch transformieren	die Anwendung das transformierte Polygon als Rechteck mit <code>cv2.imshow()</code> darstellt.
Benutzer	die Taste „q“ bedienen	die Anwendung das transformierte Rechteck als neues Bild auf der Festplatte speichert.
Benutzer	Mindestens zehn unterschiedliche Bilder mit Köpfen von Wally auf der Festplatte speichern	der Studierende die Bilder für die nächsten Aufgaben verwenden kann.
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://pypi.org/project/imutils/>

- [4]: https://docs.opencv.org/3.4/d2/de8/group_core_array.html
- [5]: https://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html
- [6]: https://docs.opencv.org/3.0-beta/modules/highgui/doc/user_interface.html
- [7]: https://en.wikipedia.org/wiki/Where%27s_Wally%3F