

Übungsblatt 9

Übung 9.1 Implementieren Sie den *Runlength-Smearing* Algorithmus zur Page-Segmentation, wie auf den Folien Kap4.2 beschrieben. Wenden Sie Ihren Algorithmus mit verschiedenen Parametersätzen für Cx , Cy und Csm auf die Beipielbilder an.

Ein RGB-Bild, in dem die gefundenen Segmente farblich hervorgehoben sind, können Sie erzeugen mittels `rgb = cc_labeled_image.color_ccs()`.

- Welche Parametersätze funktionieren gut, welche weniger?
- Gibt es Bilder, bei denen dieser Algorithmus nicht funktioniert? Ergänzen Sie Ihr Script so, dass es die zwischendurch erzeugten Bilder speichert und schauen Sie sich diese Bilder an, um den Grund für das Versagen dieses Algorithmus zu finden.

Übung 9.2 Probieren Sie den von Gamera bereitgestellten Algorithmus *projection_cutting* auf denselben Bildern mit verschiedenen Parametersätzen Tx , Ty und *noise*.

Können Sie Unterschiede erkennen im Vergleich zum Runlength-Smearing Algorithmus?

Übung 9.3 Um die einzelnen Zeilen in CCs zu zerlegen, gibt es in Gamera das Plugin *sub_cc_analysis*:

```
segs = img.runlength_smearing()
[imgcc, segccs] = img.sub_cc_analysis(segs)
```

Das zurückgegebene Bild *imgcc* ist ein Bild, auf dem alle CCs gelabelt sind. *segccs* ist eine Liste von CC-Listen, über die iteriert werden kann mittels

```
for line in segccs:
    # jetzt sind wir in einer Zeile
    for cc in line:
        # jetzt haben wir in cc ein einzelnes CC
```

Benutzen Sie dies, um für jede Zeile durch einen Least-Squares Fit durch die Mittelpunkt aller CCs der Zeile eine Mittelgerade durch die Zeile zu legen. Zeichnen Sie diese Gerade für jede Zeile mit rot in ein RGB-Bild ein.

Hinweise:

- Der Mittelpunkt eines CC steht in der Property *center*
- Um die Parameter m und b einer gefitteten Geraden $y = m \cdot x + b$ durch eine Liste von Punkten vom Typ *Point(x,y)* zu berechnen, können Sie die Funktion *least_squares_fit* verwenden:

```
from gamera.plugins.structural import least_squares_fit
# wir brauchen nur m und b
(m, b, q) = least_squares_fit(pointlist)
```

- Die Linien können Sie mit *draw_line* zeichnen. Dazu müssen Sie je Zeile zu den Randpunkten x_{left} (Minimum aller $cc.ul_x$ der Zeile) x_{right} (Maximum aller $cc.lr_x$ der Zeile) die zugehörigen y -Werte über die Geradengleichung mit den Parametern m und b berechnen.