# Titel: Iwas mit Barcodes

**Ausarbeitung**

Veranstaltung:
**Begleitendes Praktikum zu Computer Vision WS 2016/17**


| | |
|---|---|
| Themensteller: | **Prof. Dr. Xiaoyi Jiang** |
| | Dimitri Berh |
| | Andreas Nienkötter |
| Betreuer: | Aaron Scherzinger |
| Verfasser: | **Aleksej Matis** |
| | **Alexander Schlüter** |
| | **Kjeld Schmidt** |

# Titel: Iwas mit Barcodes

Aleksej Matis, Alexander Schlüter, Kjeld Schmidt

24. März 2017

## Contents

# 1 Einleitung

- Motivation
- Barcodes und EAN13
- Aufteilung in Lokalisation+Boundary+Reading

# 2 Lokalisierung

Kurze Einleitung

## 2.1 Gradient+Blur

Bla bla

## 2.2 LSD

This localization method is based on the LineSegmentDetector algorithm [4], which is implemented in OpenCV 3 in the cv::LineSegmentDetector class [1]. The algorithm detects line segments in an image, such as the segments formed by the bars in a barcode. On a high level, it works by first calculating the image gradient and assigning to each pixel a unit vector perpendicular to the gradient, resulting in a *level line field*. Pixels with a gradient magnitude under a certain tolerance are discarded. Connected pixels with similar level line angles are clustered together and form a *line support region*. To each line support region, a rectangle is fit which covers the whole region. To limit the number of false detections, the rectangle is only accepted as a line segment if the number of covered pixels with aligned level lines is high enough relative to the total number of covered pixels. The exact threshold is based on a statistical model (*a contrario* method), so that the rectangle is only accepted if, in a purely random level line field, the found ratio of aligned level lines is unlikely. Before a rectangle is rejected, some variations of the rectangle's parameters are tried.

The barcode localization method is taken from Creusot et al. with minor modifications. First we run the LineSegmentDetector provided by OpenCV on the image. We use default parameters, except for the first parameter LSD_REFINE_NONE, which disables refinement of line segments. We found that line refinement tends to break up barcode bars into multiple segments, e.g. if the bars in the image are not quite straight due to kinks in the material on which the barcode is printed, or if there are glare spots on the barcode. This is undesirable, since in the following steps we rely on the length of line segments in the barcode to be similar.

Next we want to find a line segment which belongs to a barcode bar approximately in the middle of the barcode. To each detected line segment, a score is assigned based on how many other line segments might belong to bars of the same barcode as the first one. Two line segments are regarded as possibly belonging to the same barcode, if

1. their centers are not too far apart,
2. they have similar length,
3. they have similar angles and
4. their projected intersection covers most of the smaller segment.

Calculation of the first three distance measures is straightforward: Let $c_i$ be the center positions, $l_i$ the segment lengths and $\alpha_i$ the segment angles, where the index $i = 1, 2$ denotes the first or the second line segment, respectively. The used criteria are

$$d_{\text{center}} \coloneqq \frac{\|c_1 - c_2\|}{l_1} < 1, \qquad \frac{|l_1 - l_2|}{l_1} < 0.3, \qquad |\alpha_1 - \alpha_2| < 0.1\,.$$

As BIIIIIIILD shows, two lines fulfilling the first three criteria might still not be arranged like barcode bars, if they are displaced along the line segment direction. To catch these cases, we project the second line onto the first and check that the intersection is large enough relative to the length of the smaller line, see BIIIIIIILD2.

For each line segment which fulfills the criteria, Creusot et al. increment the score of the first line by one. Instead, we increment the score of the first line segment by $1 - d_{\text{center}}$. This effectively weights the score by the distance between the line segments, which favors line segments which lie in the middle of the barcode.

Finally, we select the line segment with the highest score. This line segment should correspond to a barcode bar in the middle of the barcode.

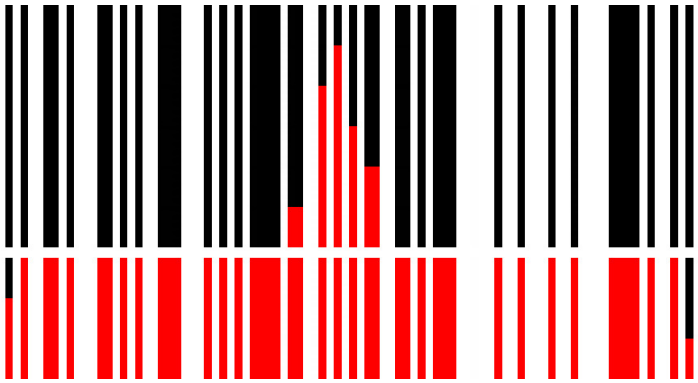

Figure 1: Wachenfeld

# 3 Boundary Detection

Einleitung

## 3.1 Wachenfeld [5]

Abbildun 1

## 3.2 LSD Bound

bla bla

## 3.3 Variation

machen wir das?

# 4 Lesen

Hier nur Template Matching

| | Fehler | Prozent Korrekt | Zeit in Sekunden |
|---|---|---|---|
| **Gradient+Blur** | 831 | 22% | 266 |
| **LSD+Wachenfeld** | 452 | 56% | 319 |
| **LSD+LSDBounds** | 41 | 97% | 1760 |
| **LSD+Variation** | 279 | 74% | 1595 |

Figure 2: Vergleich der Laufzeit und Genauigkeit der Methoden

# 5 Vergleich

Einleitun...

## 5.1 Datasets

übersicht über die Datasets

- Generiert
- Wachenfeld [5]
- ArteLab [7] [6]

## 5.2 Laufzeit

Geprüft auf dem Wachenfeld Dataset mit 1055 Bildern [5]
Laufzeit gemessen auf Debian 8 mit 4Gb Ram, 2.2 Ghz CPU, 4 Kerne und 4 Threads
Siehe Abbildung 2

# 6 Verbesserung

# 7 Fazit

# 8 Meta: Wie zitiere ich?

1. Titel des Papers bei `https://scholar.google.de/` Google Scholar suchen.
2. Bei dem Eintrag zu dem Paper unten auf zitieren klicken, dann auf BibTex.
3. Den BibTex string kopieren in die LITERATUR.BIB
4. Zitat hinzufügen durch \CITE{*name*}
5. Übersicht BibTex: `https://de.wikibooks.org/wiki/LaTeX-Kompendium:_Zitieren_mit_BibTeX`

Beispiel Templatematching [2]

Einleitung Lokalisierung Rand Lesen Testdaten Vergleich der Verfahren Verbesserung

# References

[1] G. Bradski. *cv::LineSegmentDetector Class Reference*. Mar. 20, 2017. URL: http://docs.opencv.org/3.1.0/db/d73/classcv_1_1LineSegmentDetector.html.

[2] Ling Chen, Hong Man, and Huading Jia. “On scanning linear barcodes from out-of-focus blurred images: a spatial domain dynamic template matching approach”. In: *IEEE Transactions on Image Processing* 23.6 (2014), pp. 2637–2650.

[3] Clement Creusot and Asim Munawar. “Low-computation egocentric barcode detector for the blind”. In: *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE. 2016, pp. 2856–2860.

[4] Rafael Grompone von Gioi et al. “LSD: a Line Segment Detector”. In: *Image Processing On Line* 2 (Mar. 2012), pp. 35–55. DOI: 10.5201/ipol.2012.gjmr-lsd.

[5] Steffen Wachenfeld, Sebastian Terlunen, and Xiaoyi Jiang. “Robust recognition of 1-d barcodes using camera phones”. In: *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE. 2008, pp. 1–4.

[6] Alessandro Zamberletti, Ignazio Gallo, and Simone Albertini. “Robust angle invariant 1d barcode detection”. In: *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*. IEEE. 2013, pp. 160–164.

[7] Alessandro Zamberletti et al. “Neural Image Restoration for Decoding 1-D Barcodes using Common Camera Phones.” In: *VISAPP (1)*. 2010, pp. 5–11.