

UTF-8 Vorlage

Jan Fässler

3. Semester (HS 2012)

Inhaltsverzeichnis

| | | |
|----------|---|----------|
| 1 | Problemstellung | 1 |
| 1.1 | Annahmen | 1 |
| 2 | Analyse der Problemstellung | 1 |
| 2.1 | Analyse der Visitenkarten | 1 |
| 2.2 | Analyse der Handybilder | 1 |
| 3 | Testumgebung und Testdaten | 1 |
| 3.1 | Visitenkarten Testumgebung | 1 |
| 3.2 | Vergleich Tesseract-Output mit Cardscan | 1 |
| 3.3 | Tesseract Testumgebung | 1 |
| 4 | Statistiken?????? | 2 |
| 4.1 | Vergleich Präprozessverfahren | 2 |
| 4.2 | Vergleich verschiedener Fonts | 2 |
| 5 | Konfigurationen?????? | 2 |
| 6 | Fazit | 2 |
| 6.1 | Anforderungen an die Kamera | 2 |
| 7 | Quellen | 3 |
| 7.1 | Tools und Applikationen | 3 |
| 8 | Anhang | 3 |

1 Problemstellung

1.1 Annahmen

Der Benutzer kooperiert, er wird nicht versuchen

2 Analyse der Problemstellung

2.1 Analyse der Visitenkarten

2.2 Analyse der Handybilder



Ein Typisches Problem

3 Testumgebung und Testdaten

Die Grundidee ist, die Visitenkarten mit dem Cardscanner einzulesen und dessen OCR Output als Solldaten zu verwenden. Das Rohe Cardscan-Bild wird zu den anderen Testbildern hinzugefügt, es ist für unsere Anwendung der Optimalfall. Das Bild ist scharf, hat keinen Lichtverlauf und keinen Hintergrund.

Für die Texterkennung haben wir eine zusätzliche Testumgebung erstellt. Mit dieser können wir genaue Aussagen über die Qualität der Texterkennung zu verschiedenen Schriftarten.

Als Metrik wurde F-Measure eingesetzt.

3.1 Visitenkarten Testumgebung

3.2 Vergleich Tesseract-Output mit Cardscan

3.3 Tesseract Testumgebung

Die zusätzliche Testumgebung ist vergleichsweise Trivial. Auf den Testbildern ist immer der selbe Text zu sehen. Tesseract verarbeitet die Bilder und der erkannte Text wird per String-Diff mit dem Originaltext verglichen.¹

Die Testbilder wurden mit der Hilfe von Microsoft Word erstellt. Der Text wurde mit ca 30 Pixel Höhe und Weite erstellt. Wir haben versucht, möglichst weitverbreitete Schriftarten zu verwenden. Dazu haben wir von Webseiten die beliebtesten und meist gehassten Schriftarten genommen². Zusätzlich haben wir Schriftarten hinzugefügt, die speziell für Visitenkarten angepriesen werden³. Folgende Schriftarten haben wir im Test berücksichtigt:

- Agency FB

¹<http://code.google.com/p/google-diff-match-patch/>

²Quelle: absoluteographix.co.uk/bestworstfonts.asp?strID=Guest

³Quelle: www.psprint.com/resources/powerful-business-card-fonts/

- Arial
- Baskerville Old Face
- Berlin Sans
- Calibri
- Century Gothic
- Elephant
- Eras Bold
- Franklin Gothic
- Garamond
- Gill Sans
- Impact
- Rockwell
- Tahoma
- Times New Roman
- Verdana

4 Statistiken?????

4.1 Vergleich Präprozessverfahren

4.2 Vergleich verschiedener Fonts

5 Konfigurationen?????

6 Fazit

6.1 Anforderungen an die Kamera

Für eine annehmbare Texterkennung müssen die Buchstaben eine Höhe von mindestens zehn Pixel haben. Das heisst, die Kamera muss eine genügend hohe Auflösung haben. Die Kamera des *SamsungGalaxyS2* hat eine Auflösung von 8 Megapixel, das führt dazu dass die Buchstaben der Testbilder eine Höhe von 30 bis 60 Pixel haben. Diese Anforderung wird von einem Smartphone erfüllt, welches 2011 erschien. Durch die Abonnementregelung von Swisscom, Orange und Sunrise wechseln die meisten Smartphone Benutzer alle zwei Jahre auf ein aktuelles Gerät. Somit ist heute kaum mehr ein Smartphone in Betrieb, welches diese Anforderung nicht erfüllt.

Damit der Phansalkar-Algorithmus eine gute Binarisierung durchführen kann, sollte das Bild harte Kanten haben. Die Kamera sollte also ein möglichst scharfes Bild schiessen.

Wird aber die gleiche Visitenkarte unscharf photographiert, so wird die Binarisierung massiv schlechter.

Auch wenn der Text nicht gänzlich fehlt so werden die Buchstaben nur Bruchhaft binarisiert, was zu einer sehr schlechten Texterkennung führt.



Abbildung 1: Beispiel eines scharf geschossenen Bild. Die Binarisierung ist so gut wie makellos



Abbildung 2: Die gleiche Visitenkarte unscharf photographiert. Teile de Texts sind nicht mehr im Bild enthalten.

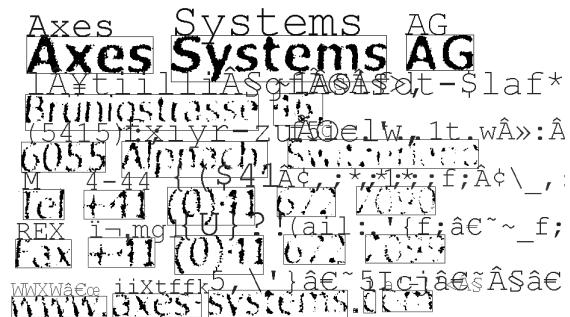


Abbildung 3: Ausschnitt des unscharfen Bildes zusammen mit dem von Tesseract erkannten Text. Der erkannte Text steht über den Boundingboxen geschrieben.

7 Quellen

1. www.psprint.com/resources/powerful-business-card-fonts/ Aufgerufen am 20.12.2013
2. absolutegraphix.co.uk/bestworstfonts.asp?strID=Guest Aufgerufen am 20.12.2013

7.1 Tools und Applikationen

- 1.

8 Anhang