Textextraktion aus nichttextuellen Datenquellen

Motivation

Im WWW sind viele nichttextuelle Datenquellen verfügbar

- PostScript (PS und PDF)
- Wordprozessor (z.B. DOC, RTF)
- Bilder (z.B. GIF, JPG)
- …

- Potentiell hoher Informationsgehalt
- Werden nicht indiziert
- Sind nicht auffindbar

Gliederung

- Textextraktion aus PostScript-Dateien
- Textextraktion aus WWW-Images
- Bezug auf die PG

Das Problem mit PostScript

PostScript ist eine Programmiersprache.

Es gibt zwei Problemebenen:

1. Es muss nicht der gesamte Textinhalt in der ps-Datei stehen

a

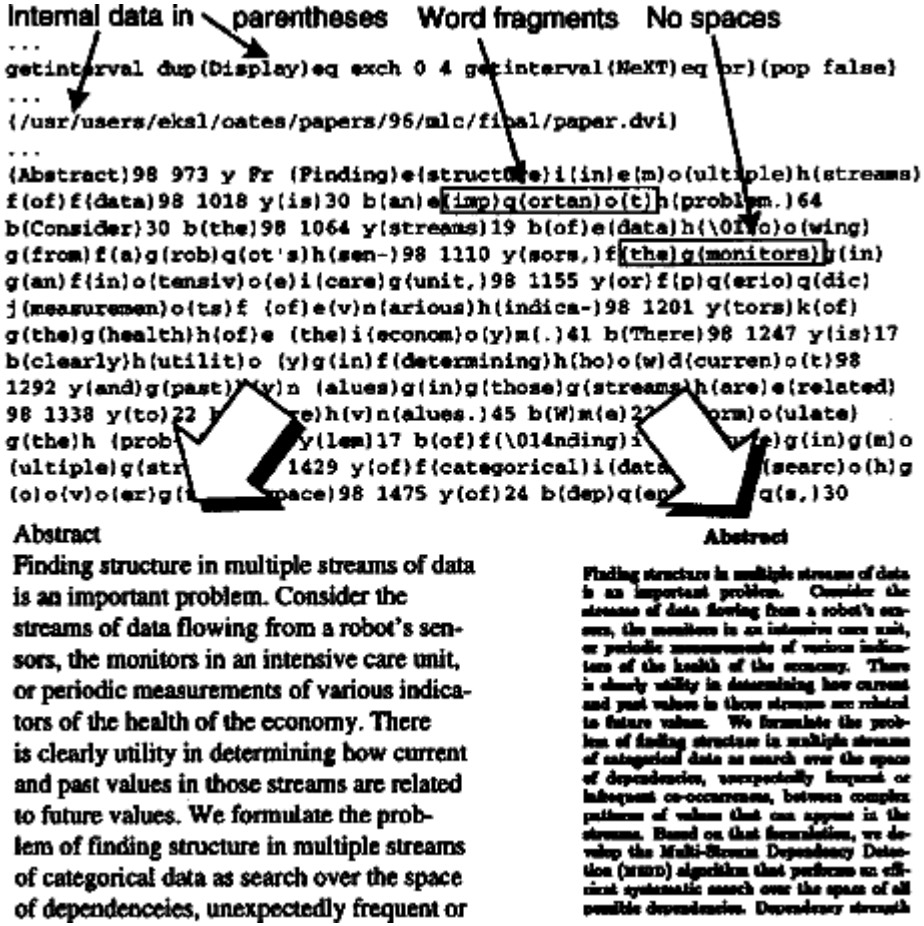
```
/fib {
  dup
  1 eq { }
       { dup 2 eq { pop 1 }
                  { dup 1 sub fib exch 2 sub fib add }
         ifelse }
  ifelse
} def

(Fib[6] = ) show 6 fib (  ) cvs show
```

b

Fib[6] = 8

Das Problem mit PostScript (2)

2. Der Text ist fragmentiert und gemischt mit Formatierungen



Abstract

Finding structure in multiple streams of data is an important problem. Consider the streams of data flowing from a robot's sensors, the monitors in an intensive care unit, or periodic measurements of various indicators of the health of the economy. There is clearly utility in determining how current and past values in those streams are related to future values. We formulate the problem of finding structure in multiple streams of categorical data as search over the space of dependencies, unexpectedly frequent or

Abstract

Finding structure in multiple streams of data is an important problem. Consider the streams of data flowing from a robot's sensors, the monitors in an intensive care unit, or periodic measurements of various indicators of the health of the economy. There is clearly utility in determining how current and past values in those streams are related to future values. We formulate the problem of finding structure in multiple streams of categorical data as search over the space of dependencies, unexpectedly frequent or infrequent co-occurrences, between complex patterns of values that can appear in the streams. Based on that formulation, we develop the Multi-Stream Dependency Detection (MSDD) algorithm that performs an efficient systematic search over the space of all possible dependencies. Dependency strength

Einfacher Textextraktor

Der Lösungsansatz aus [1] redefiniert den PostScript ***show***-Operator:
Die extrahierten ASCII-Zeichen werden in eine Datei umgeleitet

a

```
/show { print } def
```

```
Findingstructureinmultiplestreamsofdataisanimportant
problem.Considerthestreamsofdata§owingfromarobot'
ssensors,themonitorsinanintensivecareunit,orperiodi
cmeasurementsofvariousindicatorsofthehealthofthee
conomy.Thereisclearlyutilityindeterminghowcurrenta
ndpastvaluesinthosestreamsarerelatedtofuturevalues
```

Durch Verbesserung werden Wortfragmente getrennt ausgegeben

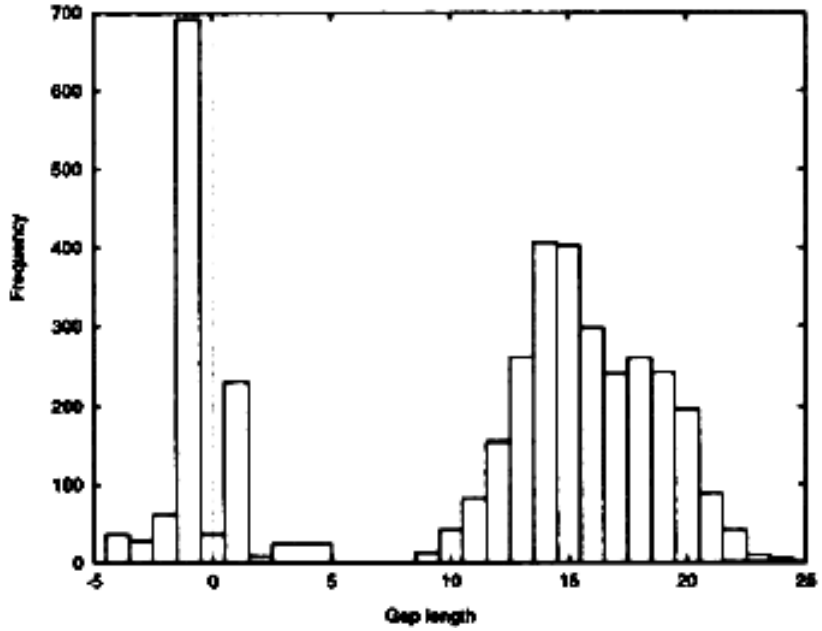
b

```
/show { print ( ) print } def
```

```
Finding structure in m ultiple streams of data is an imp
ortan t problem. Consider the streams of data §o wing
from a rob ot's sensors, the monitors in an in tensiv e
care unit, or p erio dic measuremen ts of v arious
indicators of the health of the econom y . There is
clearly utilit y in determining ho w curren t and past v
alues in those streams are related to future v alues
```

Einfacher Textextraktor (2)

Die Stellen zwischen Wortfragmenten werden heuristisch ermittelt.



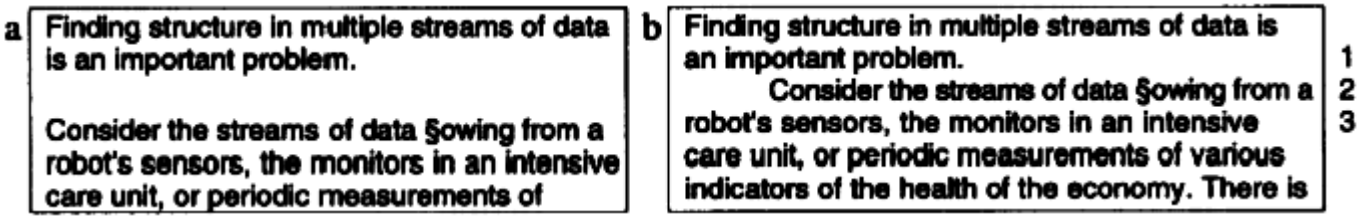

```
/X 0 def

/show {
  currentpoint pop
  X sub 5 gt { ( )  print } if
  dup print
  systemdict /show get exec
  currentpoint pop /X exch def
} def
```

Finding structure in multiple streams of data is an important problem. Consider the streams of data §owing from a robot's sensors, the monitors in an intensive care unit, or periodic measurements of various indicators of the health of the economy. There is clearly utility in determining how current and past values in those streams are related to future values.

Verbesserter Textextraktor

Die Erweiterung (in Python) basiert auf dem ersten Ansatz und behandelt folgende Fälle:

- Abstände bei großen und kleinen Schriften
- Paragraph vs. Zeilenumbruch



- Nicht-ASCII Zeichen
- Worttrennung am Zeilenende
- Seitenumkehrung

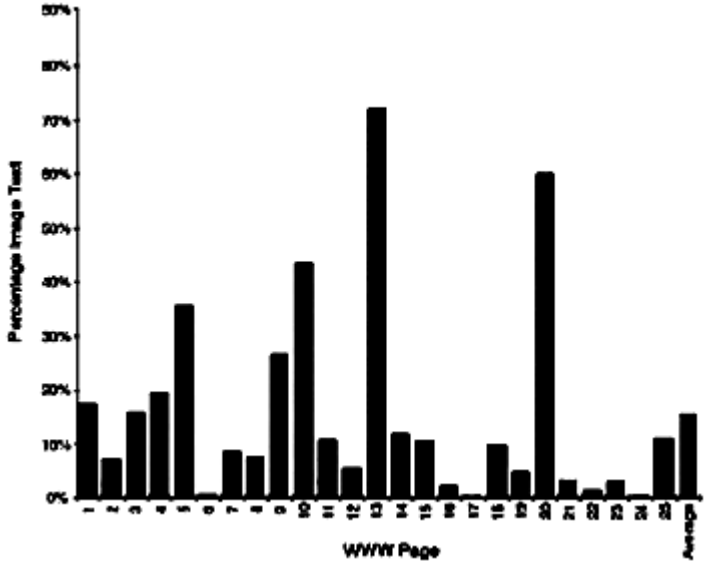
Andere Ansätze

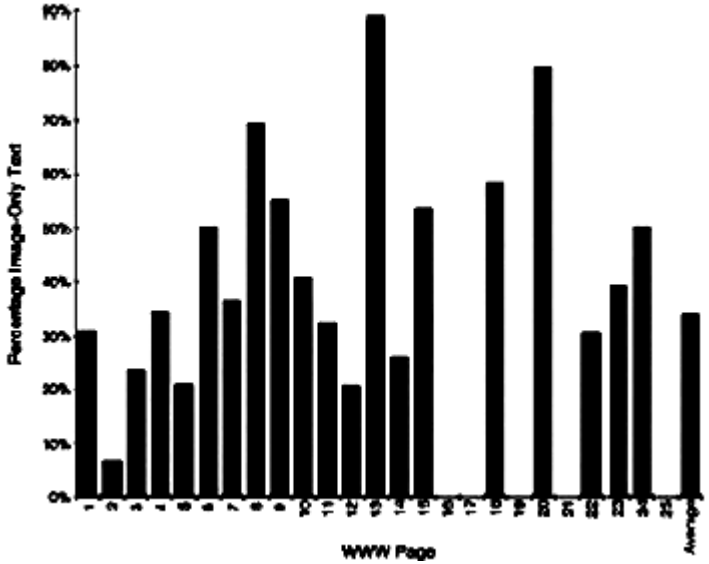
- **ps2ascii.pl** Perl-Script, extrahiert geklammerten Text
- **ps2txt** C-Programm, extrahiert geklammerten Text, spezieller Code für durch dvips generierte PostScript-Dateien
- **ps2a.sh** komplexes PostScript-Programm, optimiert für Dateien, generierten aus $\mathrm{T_EX}$
- **pstotext (DEC)** PostScript- und C-Programm von DEC, teuer, entspr. komplex und sehr langsam, aber konvertiert exzellent
- **ps2ascii** aus Ghostscript, nicht besonders zuverlässig
- **ps2html** Variante des ps2ascii, entwickelt an Johns Hopkins University für spezielle Dateien (PS aus QuarkXPress)
- **pstotext** funktioniert mit GhostScript, konvertiert auch PDF, aber mit weniger zuverlässigem Ergebnis
- **pdf2text pdf2html** Bestandteil des Pakets xpdf

Textextraktion aus WWW-Images

Ergebnis einer kleinen Stichprobe (25 Websites) [2]:

- Im Durchschnitt 15 % der ganzen Textinformation einer Website steht in Images.
- Im Durchschnitt 34 % davon steht bei vielen Websites nur in Images, und nirgendwo sonst auf der Website.





Textextraktion aus WWW-Images (2)

GIF‘s und JPEG‘s sind im Internet am meisten verbreitet.

- GIF (Graphics Interchange Format): 256 Farben, verlustfrei

- JPEG: RGB-Farben, verlustbehaftet, besser für Fotos

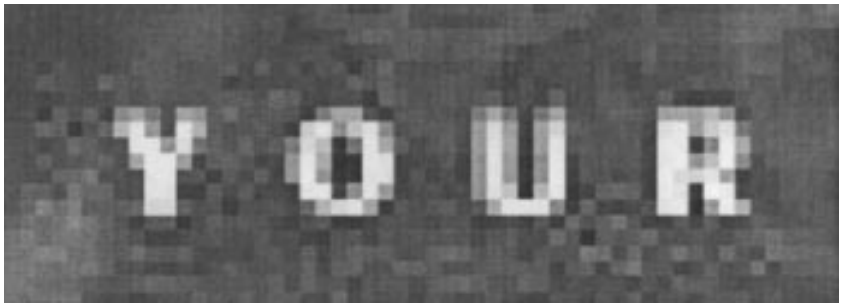
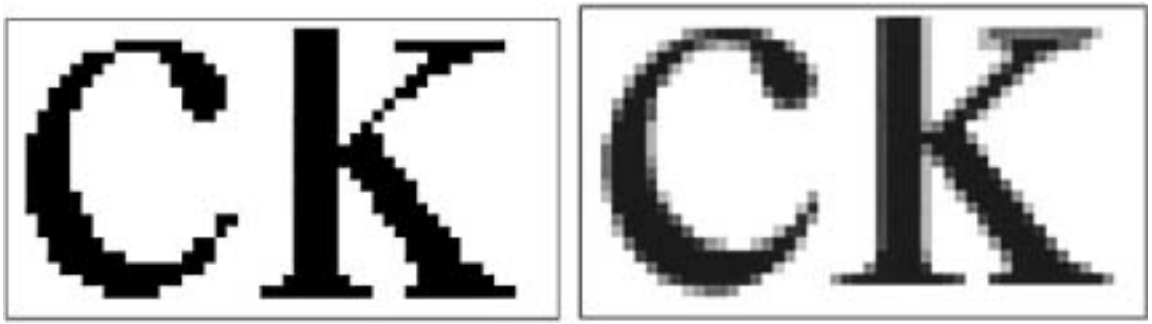


Abb.: JPEG‘s Kompressionsartefakte

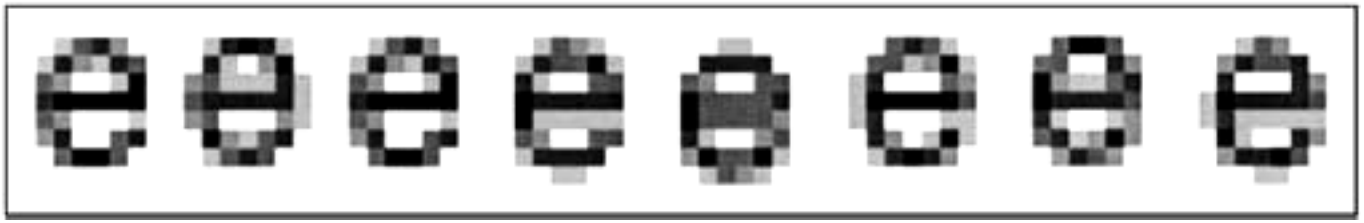
Die Ausnutzung der besonderen Eigenschaften eines Graphikformats kann die Ergebnisse der Texterkennung verbessern.

Allgemeine Probleme

- Kleine Auflösung (üblich 72 dpi)
- Anti-Aliasing



- Räumliche Mustereffekte (spartial sympling effects)



Allgemeine Probleme (2)

- Images mit schwer erkennbaren text

- Dynamische Images (GIF89a Standard)
- Images mit „Multizeichen“

Verfahren

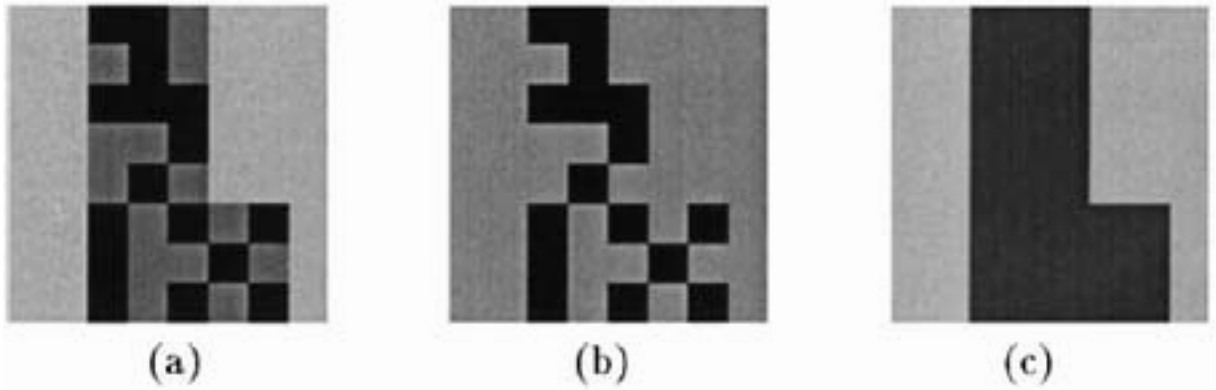
Aktuelle OCR-Technologie erkennt Text nur auf einem einheitlichen Hintergrund [3].

Alternative Verfahren benutzen bis zu drei Schritte:

- Farbclustering (color clustering)
- Zeichenentdeckung (character detection)
- Layoutanalyse (layout analysis)

Beispiel des Farbclusterings

a) Buchstabe „L“
b) Clusterung, basiert auf RGB-Distanz
c) Clustering, basiert auf Kombination von RGB- /räumliche Distanz



Bezug auf die Projektgruppe

PostScript (PS und PDF):

- Public Domain Programme verfügbar
- Zur Verbesserung der Treffer der von uns eingesetzten Suchmaschine
- Umwandlung in HTML-Format (mit Qualitätsverlusten)

WWW-Images:

- Keine Public Domain Programme verfügbar
- Ggf. Ressourcen-intensiv

Literatur

[1] G. G. Nevill-Manning, T. Reed, I.H. Witten. Extracting Text from PostScript. In *Software-Practice and Experince*, vol. 28(5), 481-491, 1998.

[2] D. Lopresti, J. Zhou. Locating and Recognizing Text in WWW Images. In *Information Retrieval 2*, 177-206, 2000.

[3] V. Wu, R. Manmatha, E. M. Riseman. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, No. 11, 1999

[4] D. Byers. Full-text Indexing of Non-textual Resorces. In *Computer Networks and ISDN Systems* 30 (1998), 141-148