

Sustav za određivanje strukture teksta na temelju položaja pojedinih znakova

Herman Zvonimir Došilović

Sveučilište u Zagrebu
Fakultet elektrotehnike i računarstva

Zagreb, srpanj 2018.

Sadržaj

1 Uvod

- Optičko raspoznavanje znakova
 - Primjene optičkog raspoznavanja znakova
 - Komponente sustava za optičko raspoznavanje znakova

2 Određivanje strukture teksta na temelju položaja pojedinih znakova

- Željena funkcionalnost
- Suradnja s OCR-sustavom
- Skup podataka za ispitivanje
- Korištenje skupa podataka za ispitivanje

3 Algoritmi za određivanje strukture teksta

- Algoritmi za određivanje linija
- Algoritmi za rastavljanje riječi

4 Mjere točnosti algoritama

5 Rezultati i analiza

6 Zaključak

Uvod

Optičko raspoznavanje znakova

- Engl. *Optical Character Recognition (OCR)*



Slika 1: OCR-sustav na mobilnom uređaju. [1]

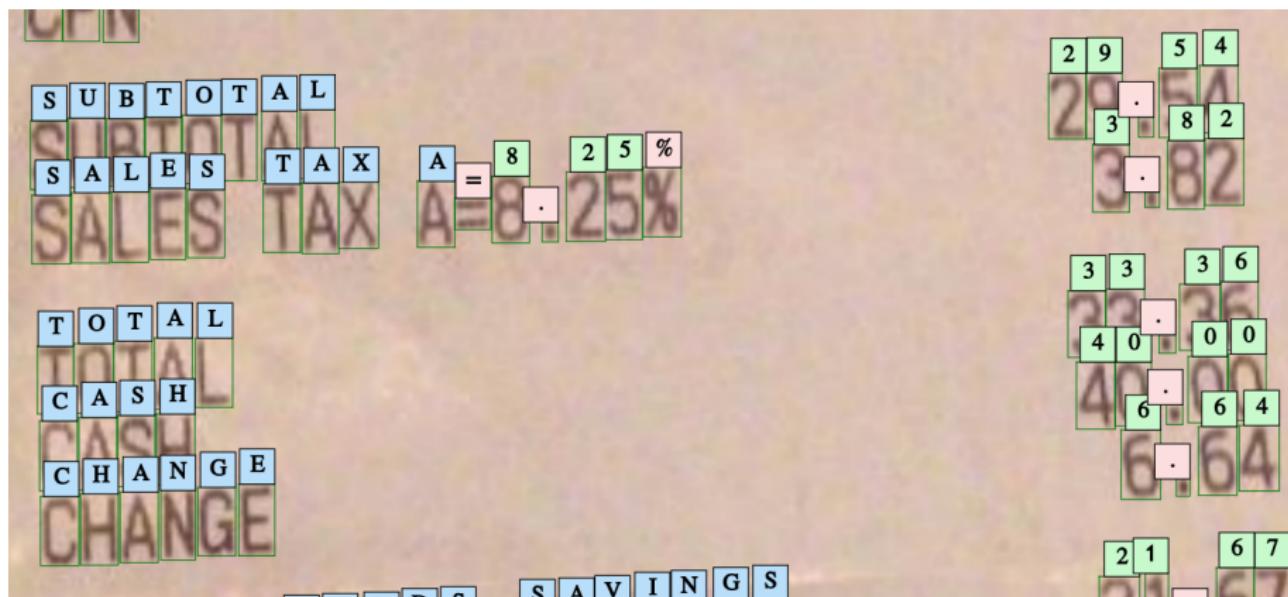
Optičko raspoznavanje znakova

Primjene optičkog raspoznavanja znakova (1)

- Bankovne aplikacije za mobilne uređaje
 - ▶ Plaćanje računa
 - ▶ Otvaranje bankovnog računa (npr. Zagrebačka banka, Revolut, N26)
- Turizam i hoteljerstvo
 - ▶ Prijava boravka u hotelima
- Registracija glasača na biralištima
- Registracija posjetitelja na raznim događajima
- Granične kontrole
- Upravljanje financijama
- Digitalizacija knjiga
- Detekcija znakova na registarskim pločicama

Optičko raspoznavanje znakova

Primjene optičkog raspoznavanja znakova (2)



Slika 2: Vizualizacija rezultata OCR-sustava.

Optičko raspoznavanje znakova

Komponente sustava za optičko raspoznavanje znakova

Optičko raspoznavanje znakova provodi se u nekoliko koraka:

- pribavljanje slike,
- predobrada,
- segmentacija znakova,
- izdvajanje značajki znakova,
- klasifikacija znakova i
- **naknadna obrada.**

Određivanje strukture teksta na temelju položaja pojedinih znakova

Željena funkcionalnost

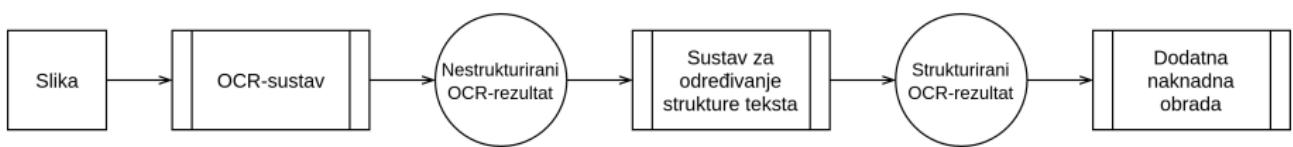
2.5 Reasoning About Efficiency

Gross reasoning about an algorithm's running time of is usually easy given a precise written description of the algorithm. In this section I will work through several examples perhaps in greater detail than necessary.

Slika 3: Vizualizacija rezultata sustava za određivanje strukture teksta.

Određivanje strukture teksta na temelju položaja pojedinih znakova

Suradnja s OCR-sustavom



Slika 4: Suradnja OCR-sustava i sustava za određivanje strukture teksta.

Određivanje strukture teksta na temelju položaja pojedinih znakova

Skup podataka za ispitivanje

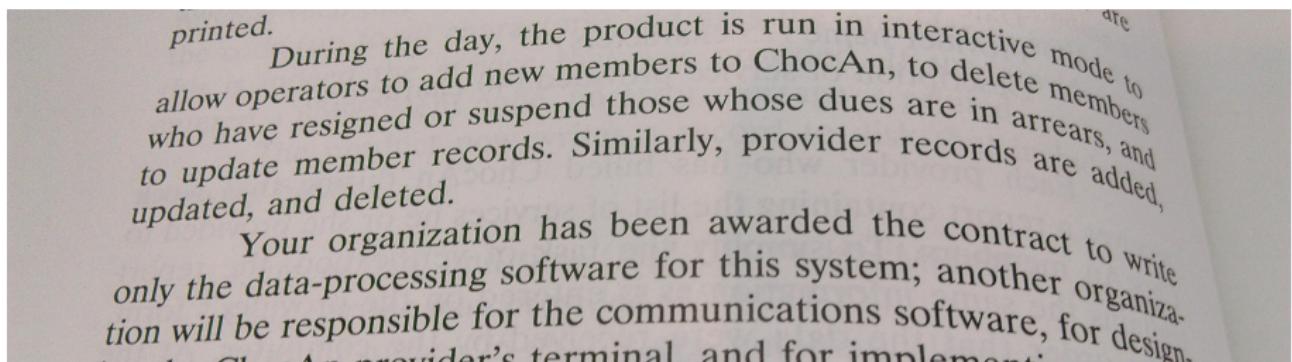
Skup podataka za ispitivanje sastoji se od:

- slika,
- ulaznih datoteka u formatu JSON i
- očekivanih izlaznih datoteka.

Skup podataka za ispitivanje

Slike (1)

- **Ručno** označene i klasificirane slike.
- 100 slika računa iz trgovina (ukupno 85068 znakova).
- 34 slike sadržaja iz knjiga (ukupno 25092 znaka).



Slika 5: Primjer slike sadržaja iz knjige.

Skup podataka za ispitivanje

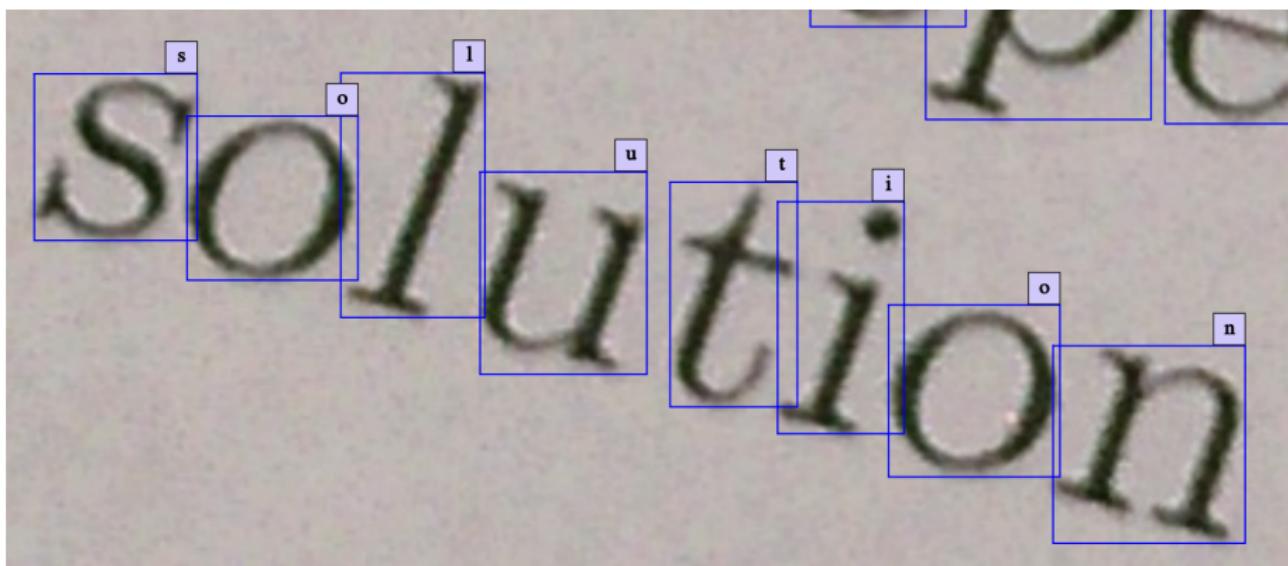
Slike (2)



Slika 6: Primjer slike računa iz trgovine.

Skup podataka za ispitivanje

Slike (3)



Slika 7: Primjer slike s ukošenim tekstrom.

Skup podataka za ispitivanje

Ulazne datoteke

- Slike **ne predstavljaju** ulaz u sustav za određivanje strukture teksta.
- Ulazne datoteke u formatu JSON predstavljaju nestrukturirani OCR-rezultat.
- Za svaki znak poznate su sljedeće informacije:
 - ▶ x - horizontalna pozicija gornjeg lijevog kuta,
 - ▶ y - vertikalna pozicija gornjeg lijevog kuta,
 - ▶ $width$ - širina,
 - ▶ $height$ - visina i
 - ▶ $value$ - Unicode vrijednost.

Skup podataka za ispitivanje

Očekivane izlazne datoteke

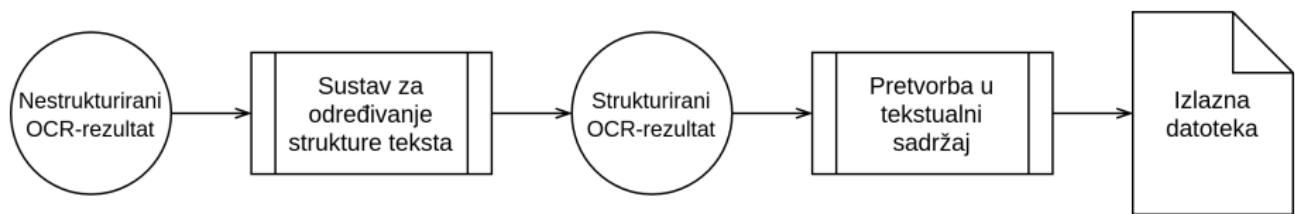
Isječak 1: Ispravni tekstualni sadržaj slike 6.

```
1 POINTS TO $35 REWARD 8770
2 BALANCE REWARDS ACCT # ****2463
3 OPENING BALANCE 20820
4 EVERYDAY POINTS - RETAIL 410
5 CLOSING BALANCE 21230
6 ****
7 Walgreens 01875
8 ACCT 7681
9 SEQUENCE 1875220350
10 PAYMENT FROM PRIMARY
11 Get the flu shot that helps provide
12 a lifesaving vaccine to a child in need
13 Get a Shot. Give a Shot.® It's that easy
14 Learn more at the pharmacy.
```



Skup podataka za ispitivanje

Korištenje skupa podataka za ispitivanje (1)



Slika 8: Postupak dobivanja izlazne datoteke iz strukturiranog OCR-rezultata.

Skup podataka za ispitivanje

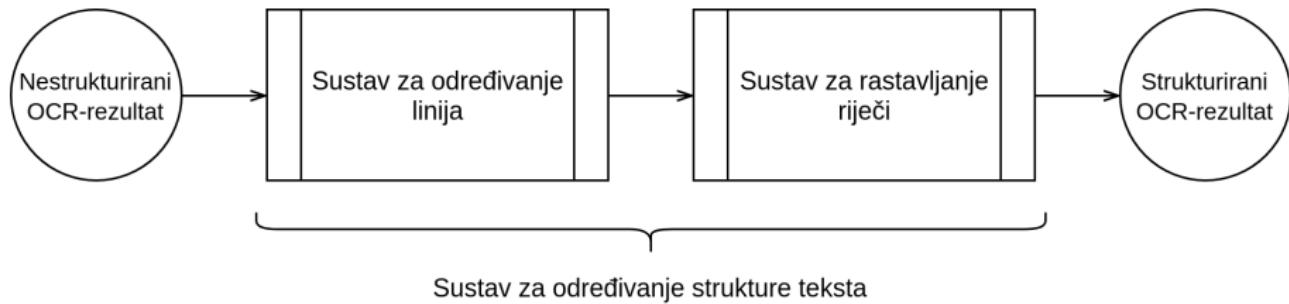
Korištenje skupa podataka za ispitivanje (2)

Isječak 2: Pseudokôd algoritma za ispis OCR-rezultata.

```
1 def ispisi(ocrRezultat)
2   for linija u ocrRezultat.linije
3     for znak u linija
4       print(znak.value)
5     end
6     print("\n")
7   end
8 end
```

Algoritmi za određivanje strukture teksta

Komponente sustava za određivanje strukture teksta



Slika 9: Komponente sustava za određivanje strukture teksta.

Algoritmi za određivanje strukture teksta

Algoritmi za određivanje linija

Known Uses

The Interpreter pattern is widely used in compilers implemented with object-oriented languages, as the Smalltalk compilers are. SPECTalk uses the pattern to interpret descriptions of input file formats [Sza92]. The QOCA constraint-solving toolkit uses it to evaluate constraints [HHMV92].

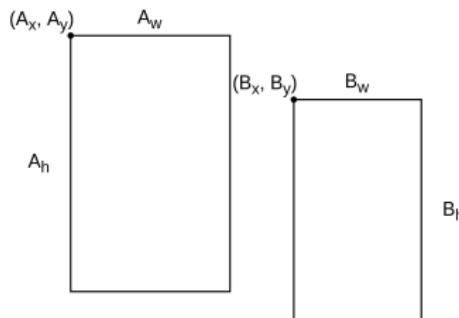
Slika 10: Vizualizacija detektiranih linija u sadržaju iz knjige.

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (1)

- Temelji se na pretpostavci da dva susjedna znaka koje se nalaze u istoj liniji ostvaruju maksimalno vertikalno preklapanje.
- Vertikalno preklapanje dva znaka definira se izrazom:

$$\text{overlap}(A, B) = \frac{\max(0, \min(A_y + A_h, B_y + B_h) - \max(A_y, B_y))}{\min(A_h, B_h)} \quad (1)$$



Slika 11: Omeđujući pravokutnici znakova.

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (2)

- Algoritam izgrađuje linije.
- Na početku, algoritam uzlazno sortira sve znakove po horizontalnoj x vrijednost.
- Promatrani znak pridružuje se onoj liniji s kojom ostvari maksimalno preklapanje:

$$I_{max} = \arg \max_{I \in L} \{overlap(A, I_{-1})\} \quad (2)$$

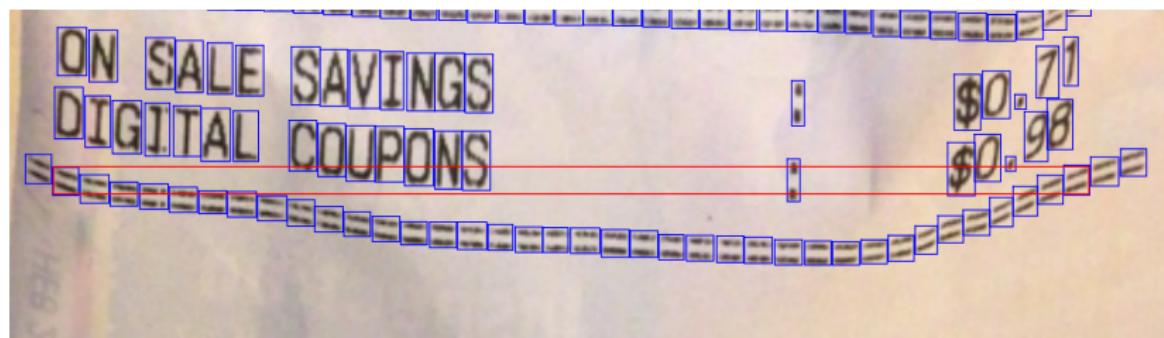
- Za preklapanje vrijednosti 0, u skup L dodaje se nova linija i promatrani znak postaje početak te linije.

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (3)

- Ponekad je poželjno izmjeriti preklapanje ne samo sa zadnjim znakom u liniji, nego i sa zadnjih nekoliko znakova:

$$l_{max} = \arg \max_{I \in L, i \in [1, \min(|I|, c_1)]} \{overlap(A, I_{-i})\} \quad (3)$$



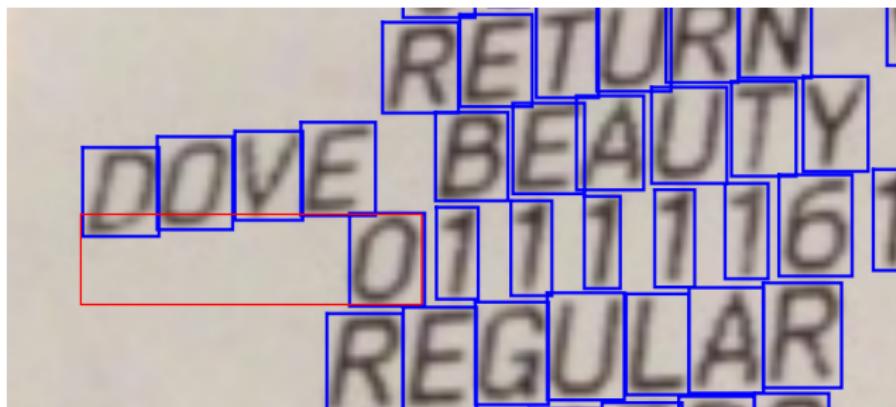
Slika 12: Valovite linije otežavaju određivanje linija.

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (4)

- Uvodi se dodatan uvijet koji će odlučiti hoće li promatrani znak pripasti liniji s kojom ostvaruje maksimalno preklapanje:

$$\max_{i \in [1, \min(|I_{\max}|, c_1)]} \{overlap(A, I_{\max-i})\} > c_2 \quad (4)$$



Slika 13: Preklapanje početaka dviju linija.

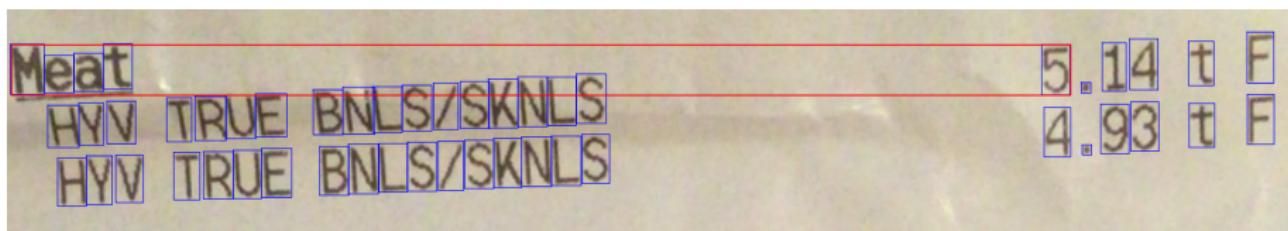
Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (5)

- Za rješavanje problema lažnog pozitivnog preklapanja definiraju se dvije funkcije:

$$f_1(x) = \frac{1}{1 + c_3 \cdot x} \quad (5)$$

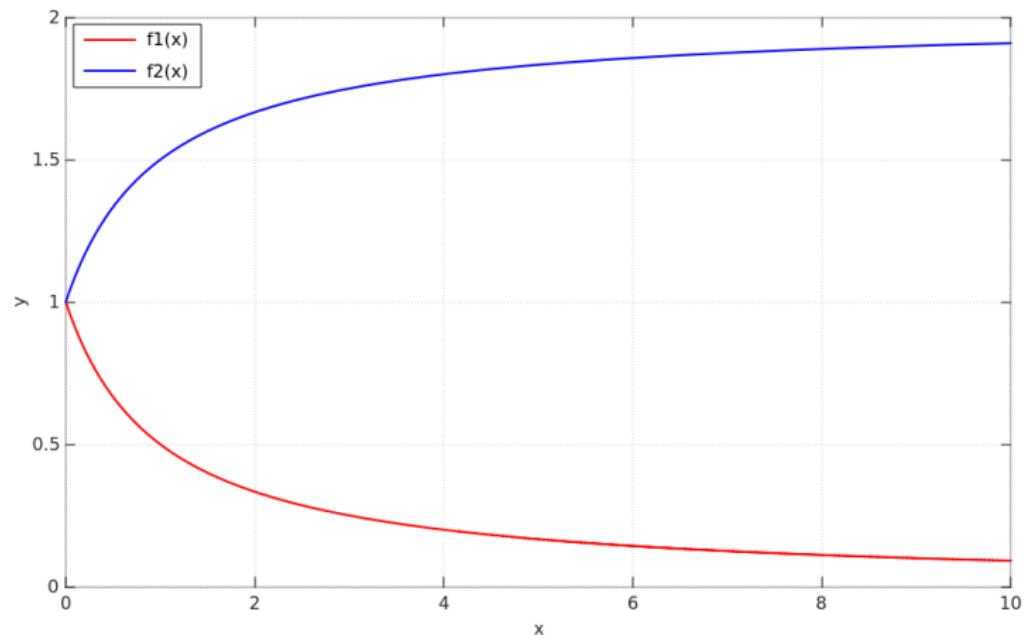
$$f_2(x) = 1 + \frac{c_4 \cdot x}{1 + c_4 \cdot x} \quad (6)$$



Slika 14: Lažno pozitivno preklapanje.

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (6)



Slika 15: Grafovi funkcije f_1 (crveno) i funkcije f_2 (plavo).

Algoritmi za određivanje linija

Algoritam temeljen na maksimalnom preklapanju znakova (7)

Iterirajući po skupu L računamo preklapanje sa svakom linijom. Neka znak u jednom trenutku pripada liniji l s kojom ostvaruje preklapanje p_l . U idućem koraku iteracije računamo preklapanje znaka s linijom k . Postavljamo nove uvjete za pridruživanje znaka liniji k .

Znak će pripasti liniji k ako je zadnji znak linije k bliži promatranom znaku od zadnjeg znaka linije l i ako vrijedi:

$$p_k > p_l \cdot f_1(\hat{d}_l(l_{-1}, k_{-1})) \quad (7)$$

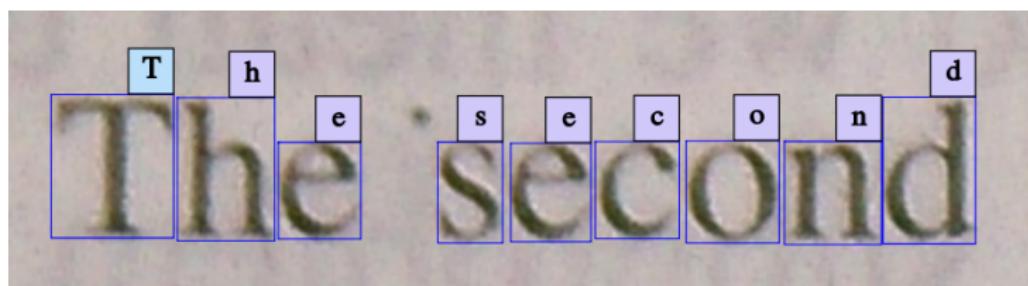
Znak će pripasti liniji k ako je zadnji znak linije k dalji od promatranog znaka nego zadnji znak linije l i ako vrijedi:

$$p_k > p_l \cdot f_2(\hat{d}_l(l_{-1}, k_{-1})) \quad (8)$$

Algoritmi za određivanje strukture teksta

Algoritmi za rastavljanje riječi

- Na ulaz primaju OCR-rezultat s grupiranim linijama.
- Trebaju ubaciti znakove bjeline između znakova za koje smatra da su završetak prethodne i početak iduće riječi.
- Razvijena su tri algoritma za rastavljanje riječi:
 - ▶ algoritam temeljen na prosječnoj širini znaka,
 - ▶ algoritam temeljen na prosječnoj relativnoj udaljenosti i
 - ▶ algoritam temeljen na prosječnoj udaljenosti centara.



Slika 16: Linija s dvije riječi.

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj širini znaka

- Najjednostavniji razvijeni algoritam.
- Temelji se na pretpostavci da je širina razmaka između riječi proporcionalna prosječnoj širini znakova u promatranoj liniji.
- Prosječna širina znaka u liniji / računa se na sljedeći način:

$$\overline{w_l} = \frac{\sum_{A \in l} A_w}{|l|} \quad (9)$$

- Algoritam ubacuje znak bjeline između znakova A i B ako vrijedi:

$$d(A, B) > \overline{w_l} \cdot c_1 \quad (10)$$

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj relativnoj udaljenosti (1)

- Temelji se na pretpostavci da je udaljenost znaka A s vrijednosti (engl. *value*) A_v proporcionalna prosječnoj udaljenosti koju svi znakovi s vrijednost A_v ostvaruju sa svojim susjedima.

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj relativnoj udaljenosti (2)

FITOPRUCCCI	SNDWCH PEP	8.81 A *
FL FITOPRUCCCI	HARD SALAMI	9.34 A *
FL HUT PEPPER CHS	YLW AMERICAN CHSE	2.40 A *
	M	2.40 A *
<u>FROZEN/DAIRY</u>		
TGIF CHED CH POPPERS	M	7.98 A *
2 @ 3.99		
<u>GROCERY</u>		
SUNCHIPS GARD SALSA	M	3.29 A *
1 @ 2 FOR 6.58		
SUN CHIPS FRCH ONION	M	3.29 A *
1 @ 2 FOR 6.58		
FL BOLD SNACK MIX		3.58 A *
2 @ 1.79		
WISE ONION GARLIC	M	2.99 A *
12PK CN DT A&W R/BR	M	5.79 B *
<u>PRODUCE</u>		
CLEM TANGERINES 3LB	M	4.99 A *
<u>Savings</u>		
You saved:		
SUNCHIPS GARD SALSA	-0.29	
SUN CHIPS FRCH ONION	-0.29	
WISE ONION GARLIC	-1.11	
TGIF CHED CH POPPERS	-0.60	
CLEM TANGERINES 3LB	-1.00	
12PK CN DT A&W R/BR	-0.80	
FL HUT PEPPER CHS	-0.80	



Slika 17: Slika računa iz trgovine.

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj relativnoj udaljenosti (3)

- Skup svih susjeda znaka A definiramo kao skup svih znakova B koji su različiti od A , koji pripadaju istoj liniji kao i A , i za koje vrijedi:

$$\hat{d}_c(A, B) < c_1. \quad (11)$$

- Skup svih susjeda vrijednosti v definiramo kao:

$$s(v) = \bigcup_{A \in C} \{S(A) | A_v = v\} \quad (12)$$

- Prosječna udaljenost između znakova A , za koje vrijedi $A_v = v$, računa se na sljedeći način:

$$\overline{d}_c(v) = \frac{\sum_{B \in C, B_v=v} \left[\sum_{D \in S(B)} d_c(B, D) \right]}{|s(v)|}$$

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj relativnoj udaljenosti (4)

- Algoritam ubacuje znak bjeline između znakova A i B ako vrijedi:

$$d_c(A, B) > \overline{d}_c(A_v) \cdot c_2 \quad \vee \quad d_c(A, B) > \overline{d}_c(B_v) \cdot c_2 \quad (14)$$

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj udaljenosti centara (1)

- Temelji se na pretpostavci da je širina razmaka između riječi proporcionalna s prosječnom udaljenosti centara između susjednih znakova.
- Skup svih susjeda znaka A definira se kao i u algoritmu temeljenom na prosječnoj relativnoj udaljenosti.
- Prosječna udaljenost centara između susjednih znakova u liniji I definira se na sljedeći način:

$$\overline{d}_c(I) = \frac{\sum_{A \in I} \left[\sum_{B \in S(A)} d_c(A, B) \right]}{\left| \bigcup_{A \in I} S(A) \right|} \quad (15)$$

Algoritmi za rastavljanje riječi

Algoritam temeljen na prosječnoj udaljenosti centara (2)

- Algoritam ubacuje znak bjeline između znakova A i B ako vrijedi:

$$d_c(A, B) > \overline{d}_c(l) \cdot c_2 \quad (16)$$

Mjere točnosti algoritama

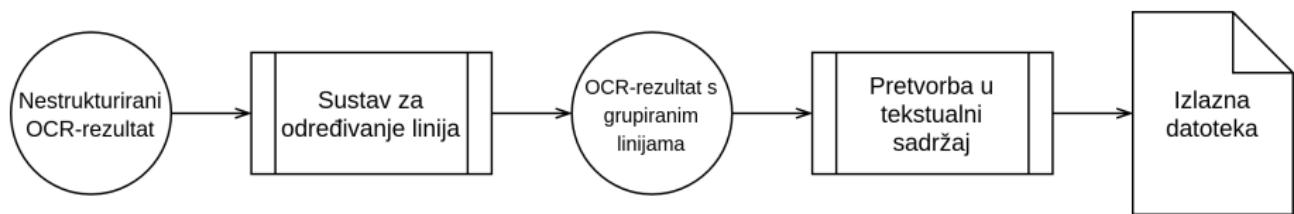
Mjere točnosti algoritama za određivanje linija (1)

- Dobiveni OCR-rezultat s grupiranim linijama pretvara se u tekstualnu datoteku koristeći algoritam prikazan u isječku 2.
- Sadržaj tekstualne datoteke uspoređuje se sa sadržajem očekivane izlazne datoteke iz koje su izbačeni znakovi bjeline.
- Točnost, odnosno dobrota (engl. *fitness*) algoritama računa se pomoću Levenshteinove udaljenosti između nizova znakova dviju datoteka:

$$f(a, b) = 1 - \frac{d_{\text{Levenshtein}}(a, b)}{\max(|a|, |b|)} \quad (17)$$

Mjere točnosti algoritama

Mjere točnosti algoritama za određivanje linija (2)



Slika 18: Postupak dobivanja izlazne datoteke.

Mjere točnosti algoritama

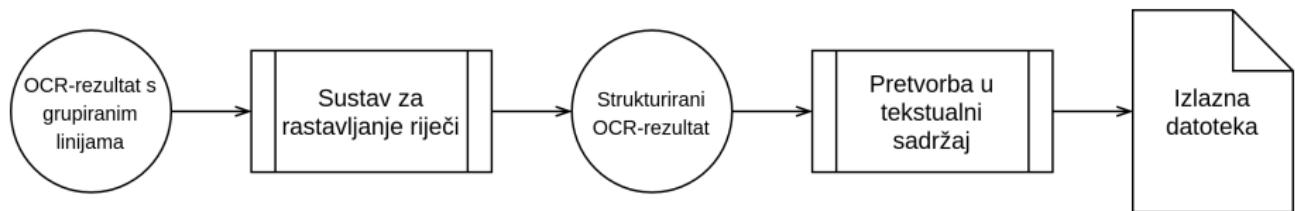
Mjere točnosti algoritama za rastavljanje riječi (1)

- Ulaz u sustav za rastavljanje riječi je OCR-rezultat s grupiranim linijama dobiven algoritmima za određivanje linija.
- Dobiveni OCR-rezultat predstavlja strukturirani OCR-rezultat koji je izlaz iz sustava za određivanje strukture teksta.
- Strukturirani OCR-rezultat pretvara se u tekstualnu datoteku koristeći algoritam prikazan u isječku 2.
- Sadržaj tekstualne datoteke uspoređuje se sa sadržajem očekivane izlazne datoteke.
- Točnost, odnosno dobrota (engl. *fitness*) algoritama računa se pomoću izraza 17.

Mjere točnosti algoritama

Mjere točnosti algoritama za rastavljanje riječi (2)

- Točnost algoritama za rastavljanje riječi predstavlja točnost sustava za određivanje strukture teksta.



Slika 19: Postupak dobivanja izlazne datoteke.

Rezultati i analiza

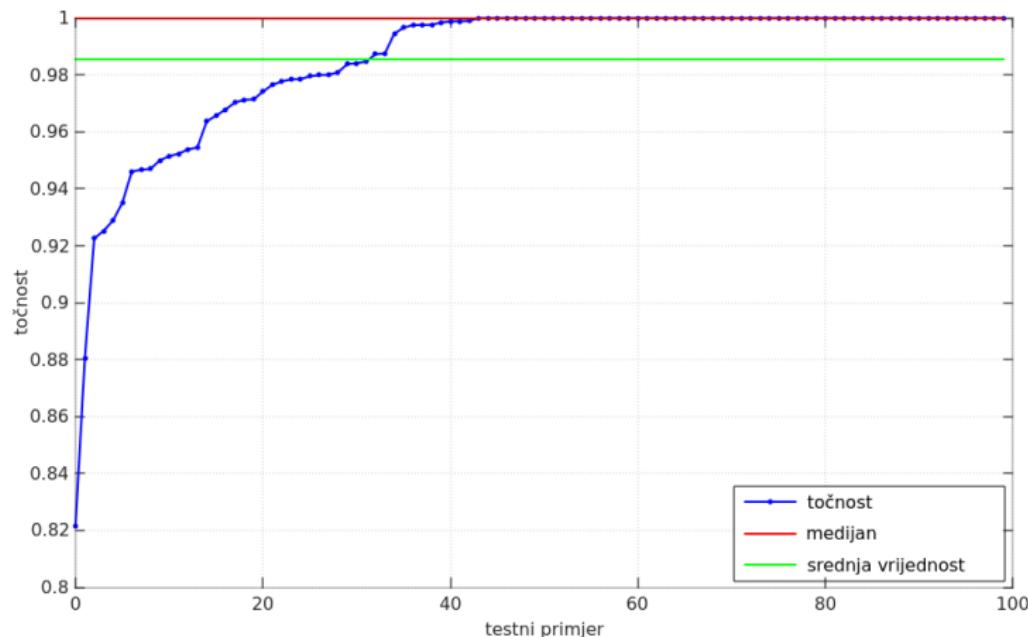
Rezultati algoritama za određivanje linija (1)

Tablica 1: Točnost algoritma za određivanje linija.

	Min.	Sred.	Med.	Maks.	Udio s maks. toč.
Računi	0,82	0,99	1	1	0,57
Knjige	0,67	0,98	1	1	0,64

Rezultati i analiza

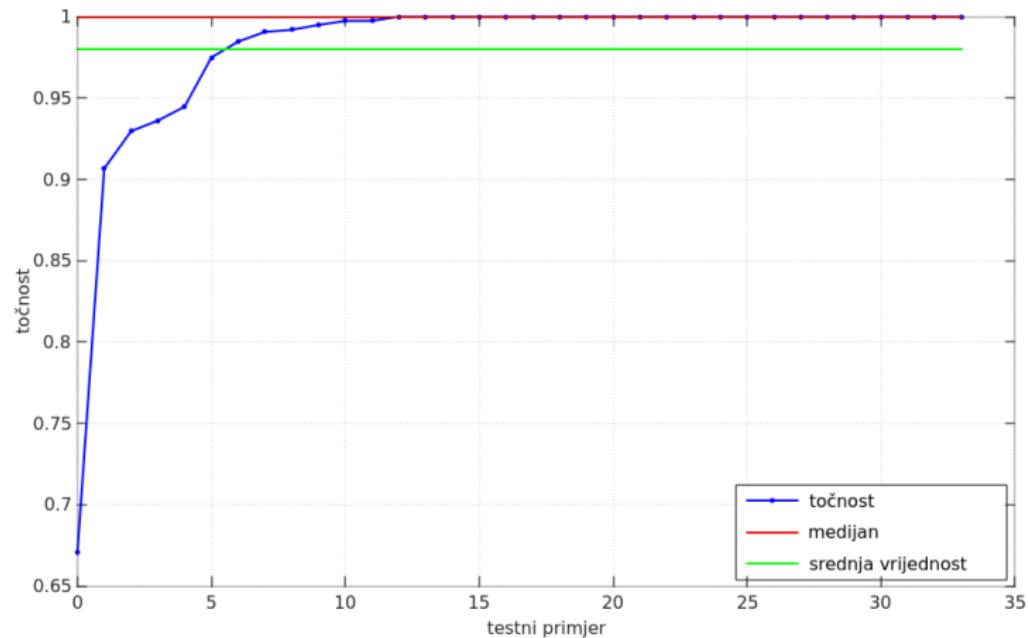
Rezultati algoritama za određivanje linija (2)



Slika 20: Graf točnosti algoritma za određivanje linija u sadržaju s računa iz trgovine.

Rezultati i analiza

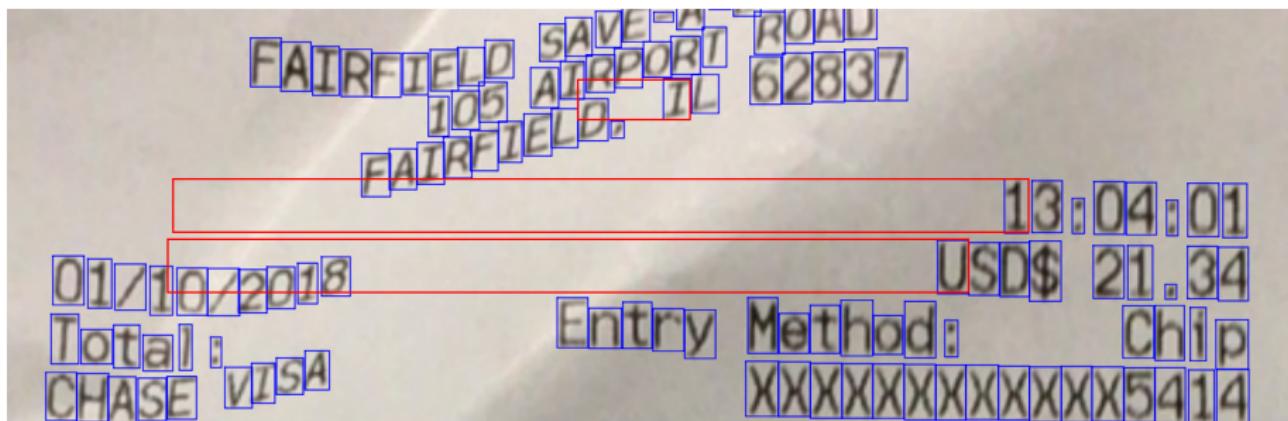
Rezultati algoritama za određivanje linija (3)



Slika 21: Graf točnosti algoritma za određivanje linija u sadržaju iz knjige.

Rezultati i analiza

Analiza algoritma za određivanje linija



Slika 22: Neispravna detekcija linija u računima zbog prevelike zakrivljenosti sadržaja.

Rezultati i analiza

Rezultati algoritama za rastavljanje riječi (1)

Tablica 2: Točnost algoritama za rastavljanje riječi u sadržaju s računa iz trgovine.

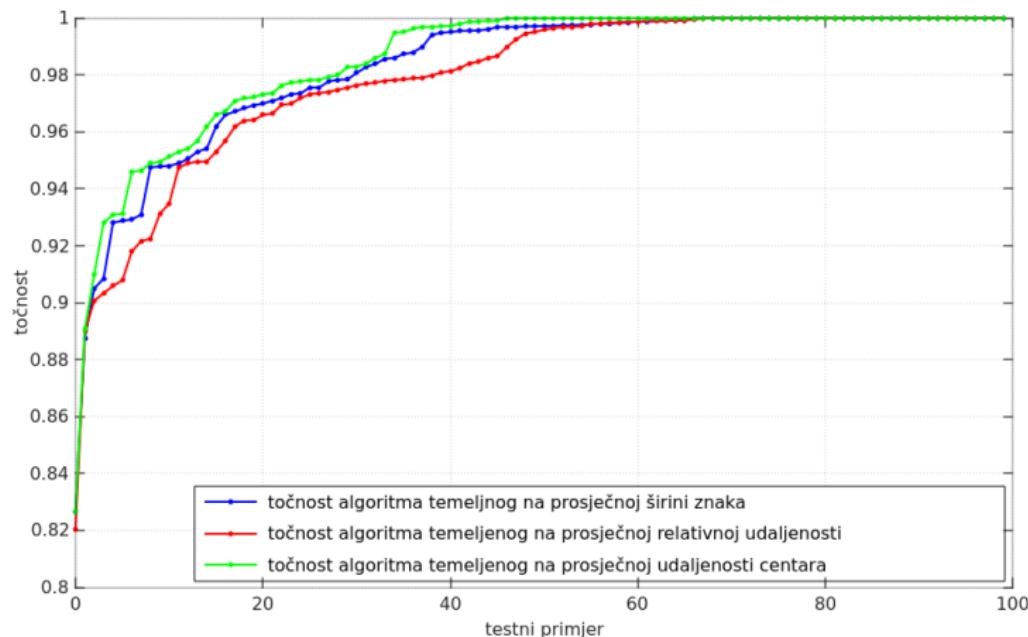
	Min.	Sred.	Med.	Maks.	Udio s maks. toč.
<i>avgcharwidth</i>	0,83	0,98	0,99	1	0,34
<i>avgreldist</i>	0,82	0,98	0,99	1	0,33
<i>avgcenterdist</i>	0,83	0,99	1	1	0,54

Tablica 3: Točnost algoritama za rastavljanje riječi u sadržaju iz knjiga.

	Min.	Sred.	Med.	Maks.	Udio s maks. toč.
<i>avgcharwidth</i>	0,6	0,96	0,97	1	0,2
<i>avgreldist</i>	0,61	0,92	0,93	0,961436	0,6
<i>avgcenterdist</i>	0,59	0,93	0,95	1	0,02

Rezultati i analiza

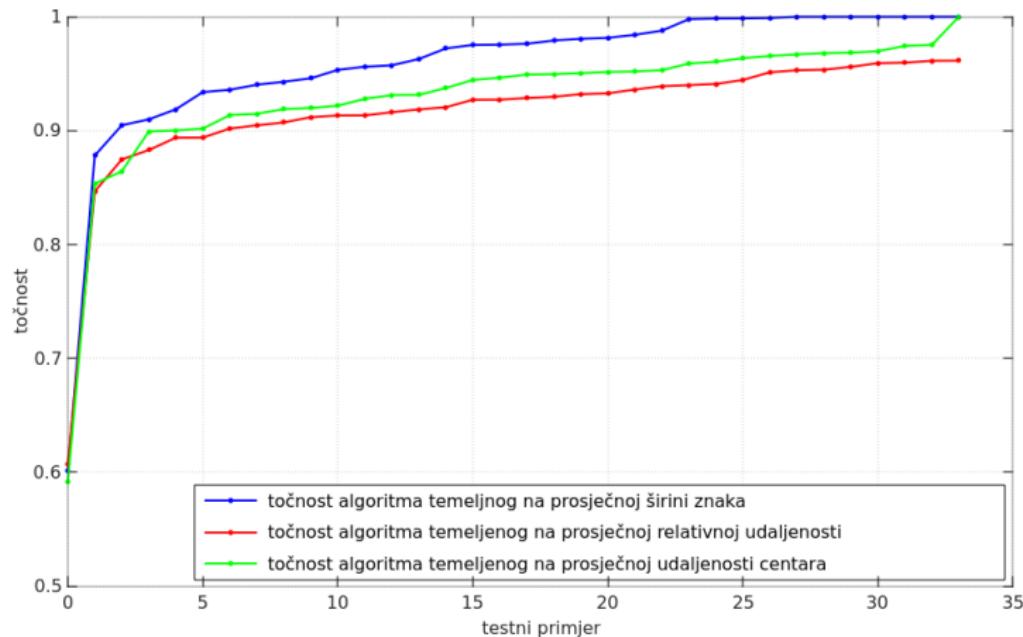
Rezultati algoritama za rastavljanje riječi (2)



Slika 23: Graf točnosti algoritama za rastavljanje riječi u sadržaju s računa iz trgovine.

Rezultati i analiza

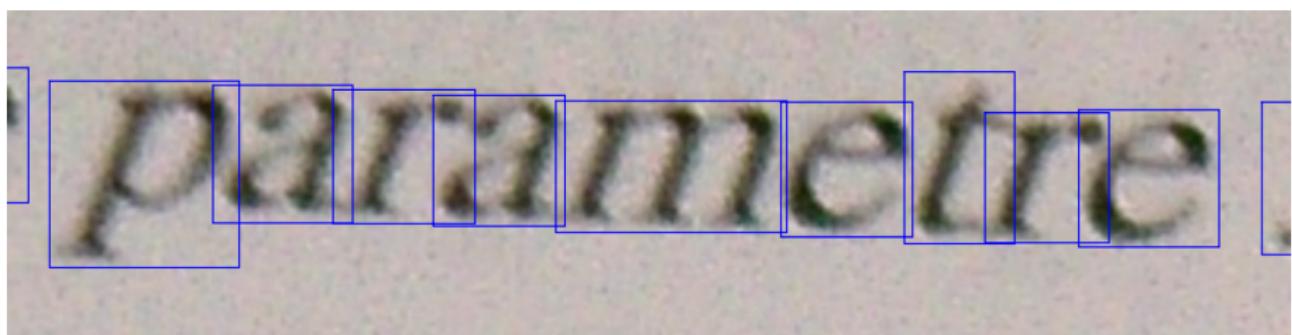
Rezultati algoritama za rastavljanje riječi (3)



Slika 24: Graf točnosti algoritma za rastavljanje riječi u sadržaju iz knjige.

Rezultati i analiza

Analiza algoritama za rastavljanje riječi



Slika 25: Horizontalno preklapanje znakova.

Zaključak

- Rezultati su zadovoljavajući.
- Analiza pokazuje kako postoji još prostora za poboljšanje.
- U budućem radu preporuča se:
 - ▶ sastavljanje novog skupa podataka i definiranje novih mjera točnosti i
 - ▶ automatizirano traženje optimalnih parametara.

Literatura

[1] Microblink Ltd. DeepOCR Technology, 2018. URL

<https://microblink.com/technology>. Pristupano:
03.07.2018.