ELEKTROTEHNIČKI FAKULTET
BANJA LUKA

SEGMENTACIJA KARAKTERA

Student: Ljubiša Branković
Broj indeksa: 56/03
Profesor: dr. Zdenka Babić
Asistent: mr. Vladimir Risojević

1. Uvod

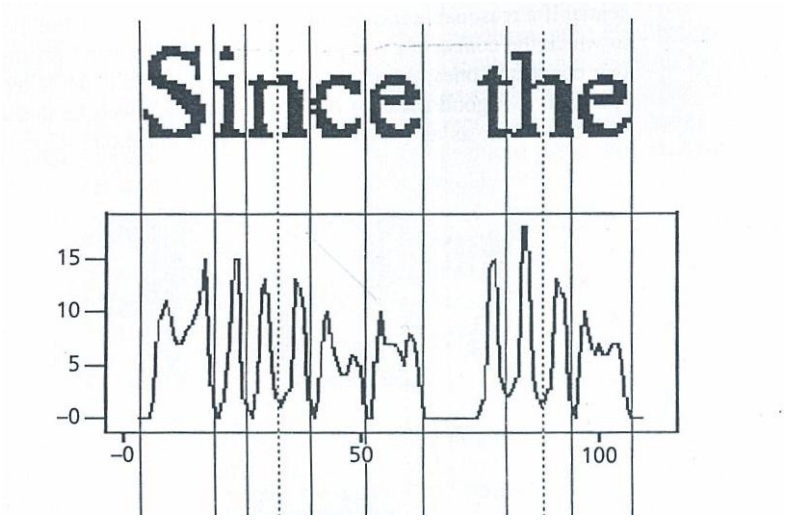
Segmentacija linija i riječi u nekom dokumentu, a zatim segmentacija pojedinačnih karaktera i simbola je važan posao u postupku optičkog prepoznavanja teksta. Većina grešaka pri OCR-u (Optical Character Recognition) su uzrokovane greškama nastalim prilikom segmentacije. Veoma često susjedni karakteri se dodiruju, ili su polja u kojima se nalaze preklopljena. Zbog toga je ispravna segmentacija date riječi u karaktere veoma složen i zahtjevan posao.

2. Realizacija

2.1. Izbor načina segmentacije

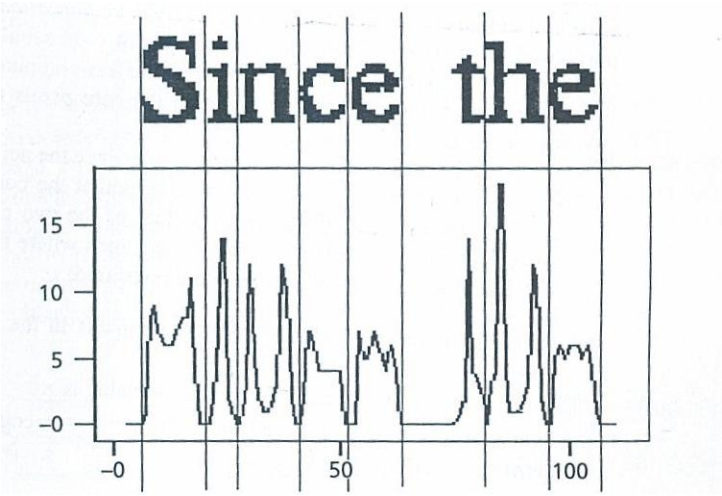
Ne postoji algoritam koji se može primjeniti na bilo koji skenirani dokument. Može se reći da segmentacija karaktera predstavlja problem koji još nije u potpunosti riješen. Najčešće se koriste metode za segmentaciju karaktera koje se baziraju na nekoj do varijacija vertikalne projekcije.

Najjednostavniji način od svih je da se nađu minimumi vertikalne projekcije i da se segmentacija izvrši na tim mjestima (slika 1.). Međutim ovdje se mogu pojaviti greške. Na osnovu ove metode može se desiti da karakter bude podijeljen na dva dijela prilikom segmentacije (isprekidana linija na slici) . Ovo se dešava jer se i u okviru datog karaktera mogu pojaviti minimumi. Oni čak mogu biti i manji od minimuma vertikalne projekcije na čijem mjestu bi trebala biti izvršena segmentacija (pune linije na slici). Na datom primjeru sa slike 1. minimum izmenđu slva *t* i *h* je veći ili jednak minimumu na slovu *h*.



Slika 1. Korištenje minimuma za određivanje lokacije karaktera

Slična tehnika vrši segmentaciju na osnovu „cijene koštanja“. Cijena koštanja se računa na osnovu broja crnih piksela u koloni. Uslov je da kolona ima „susjeda“ u istoj vrsti (liniji teksta), koja takođe ima crne piksele (slika 2.). Cijena koštanja je prikazana na donjem dijelu slike. Kolone u kojima je cijena koštanja mala su kandidati za mjesta na kojima će slika biti segmentirana. Za dati primjer segmentacija teksta u karaktere će se desiti na mjestima gdje je cijena koštanja nula. Rješenje u ovom projektu će se bazirati na ovoj tehnici, s tim da će prag odlučivanja zavisiti od veličine slova.



Slika 2. Lociranje izolovanih karaktera korištenjem cijene koštanja

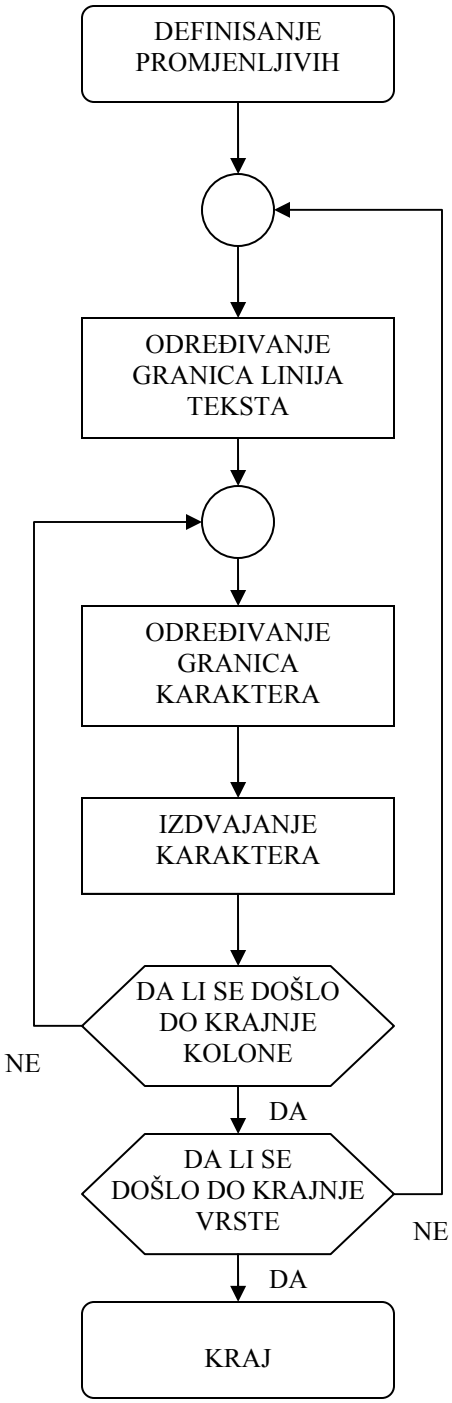
2.2. Algoritam

Kao što je rečeno postupak segmentacije u ovom projektu se zasniva na određivanju broja crnih piksela u koloni date linije teksta i poređenju sa zadatim pragom. Da bi se mogla izvršiti segmentacija karaktera, prvo je potrebno izvršiti segmentaciju linija u tekstu, kao i riječi. Radi jednostavnosti ovdje nije obavljena segmentacija riječi, već je nakon izdvajanja linija odmah izvršeno izdvajanje karaktera.

Izdvajanje linija je obavljeno na sličan način kao što je urađeno i sa izdvajanjem pojedinih karaktera. Razlika je u tome što se umjesto određivanja broja crnih piksela po kolonama date linije određuje broj crnih piksela u po vrstama, u cijelom dokumentu. Ako se smatra da nema šuma i da između redova postoji dovoljan razmak da se karakteri iz susjednih redova ne preklapaju, može se usvojiti da prag bude nula. Za označavanje granica reda koriste se dvije promjenljive, od kojih jedna označava početak reda, a druga kraj. Prije ispitivanja sledećeg reda potrebno je ispitati da li se došlo do zadnje vrste matrice.

Kada se odrede granice linija teksta, može se početi sa izdvajanjem karaktera u datoj liniji. U svakoj koloni se provjerava proj crnih piksela. Kada se prvi put desi da je broj crnih piksela veći od zadatog praga, označava se prva granica. Zatim se nastavlja sa ispitivanjem kolona, s tim što se sada ispituje da li je broj crnih piksela manji od zadatog praga. Ako se to desi postavlja se druga granica. Ovako definisane granice sada predstavljaju početak i kraj oblasti u kojoj se nalazi dati karakter. Za određivanje granica

koriste se dvije promjenljive, slično kao i kod određivanja granica linija teksta. Prije ispitivanja granica sledećeg karaktera, potrebno je ispitati da li se došlo do zadnje kolone, da ne bi došlo do greške. Nakon toga imamo određene sve granice karaktera i može se izvršiti njegovo izdvajanje. Na slici je dat dijagram toka segmentacije karaktera.



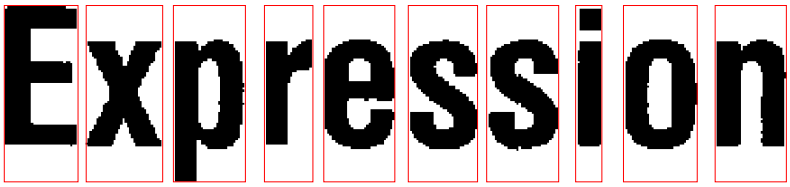
Slika 3. Dijagram toka zadatka

2.3. Detalji realizacije

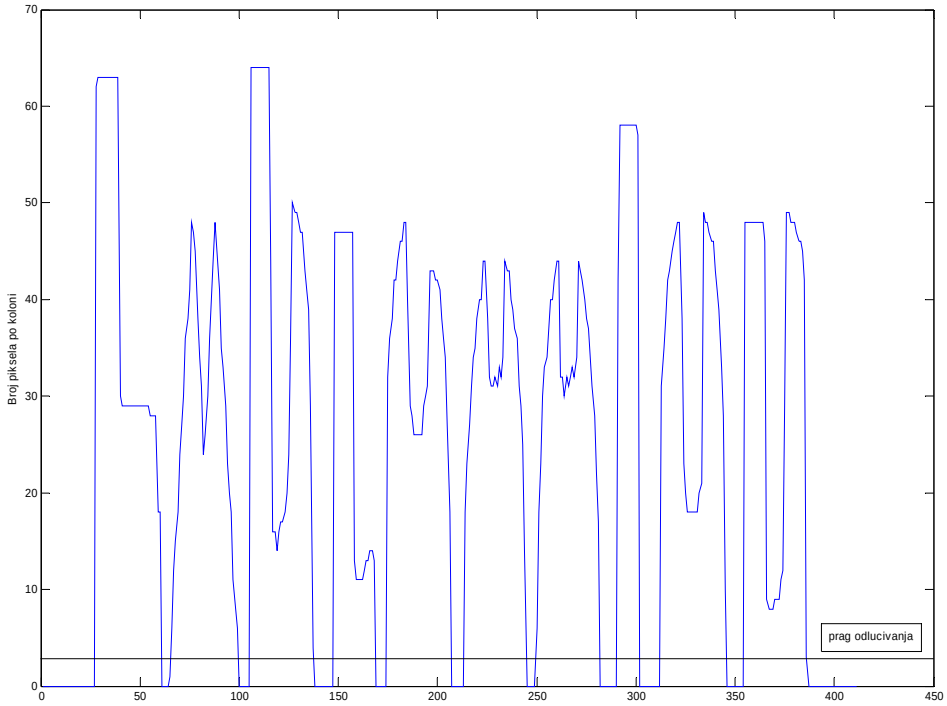
Određivanje broja piksela u koloni date linije teksta se može odrediti sabiranjem piksela invertovane slike. Sliku je potrebno invertovati, jer jedinice odgovaraju bijelim pikselima, a nule crnim. Invertovanjem se dobija obrnuta situacija, tj. pikseli koji se odnose na karaktere sada imaju vrijednost jedan, tako da se njihovim jednostavnim sabiranjem može odrediti njihov broj u datoj koloni. Naravno treba napomenuti da se podrazumijeva da slika koja se segmentira mora biti binarna.

Može se desiti da neki karakteri budu spojeni, što predstavlja problem. Ovo se može djelimično riješiti povećanjem praga odlučivanja do određene mjere. Međutim maksimalna vrijednost praga zavisi od veličine karaktera. Ako su karakteri veći, onda imaju veći broj piksela po koloni. To znači da se može koristiti veći prag odlučivanja, jer se time neće značajnije narušiti izgled karaktera. Ako bi se isti takav prag primjenio na manje kataktere, zbog manjeg broja piksela po koloni može doći do značajnog narušavanja oblika karaktera, ili čak segmentacije na pogrešnom mjestu, čime bi se onemogućilo prepoznavanje karaktera.

Na slici 4. je dat tekst na koji je primjenjena segmentacija. Na slici 5. je grafički prikazana vertikalna projekcija, kao i prag odlučivanja za dati tekst sa slike 4.

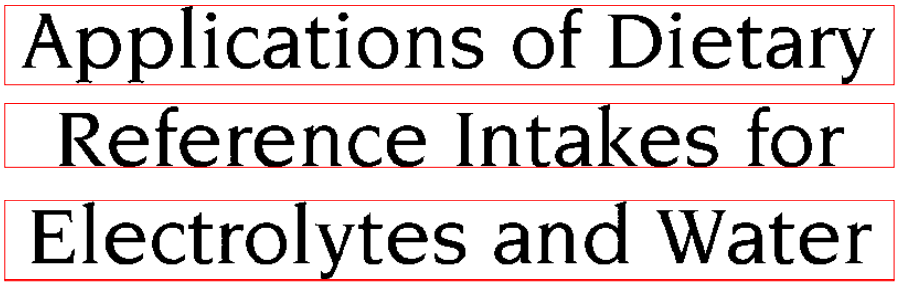


Slika 4. Tekst sa primjenjenom segmentacijom

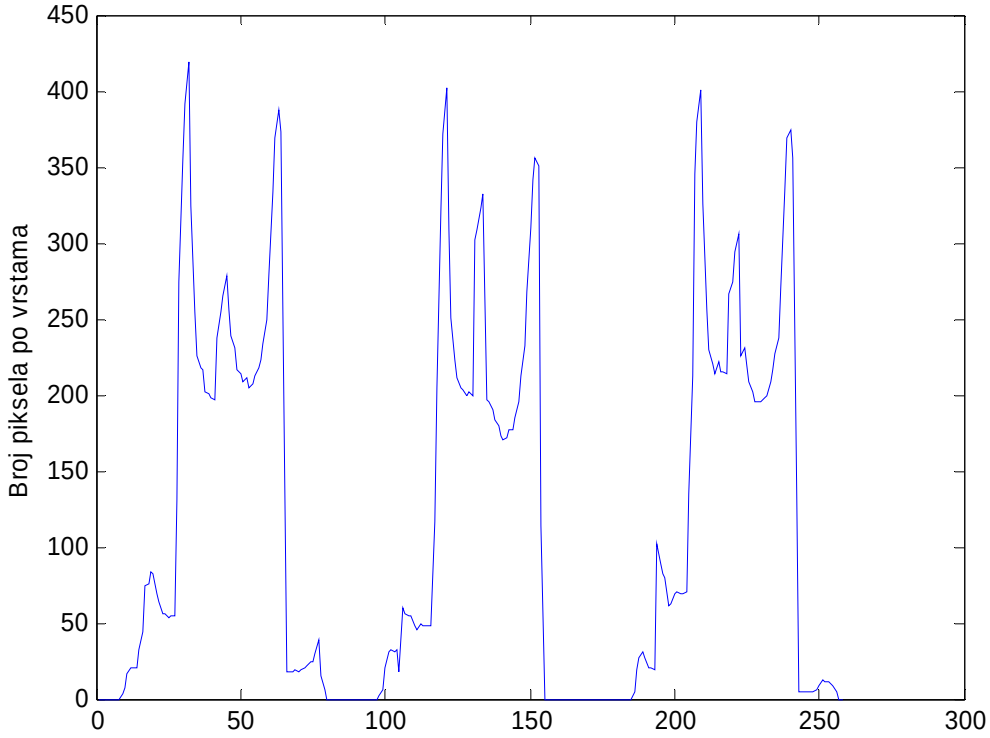


Slika 5. Vertikalna projekcija teksta

Na slici 6. je dat tekst na kome su izdvojene linije teksta. Na slici 7. je dat broj piksela po vrstama na osnovu kojeg se vrši izdvajanje linija. Postupak je sličan kao i za izdvajanje karaktera, s tim što prag odlučivanja nije promjenljiv i jednak je nuli.



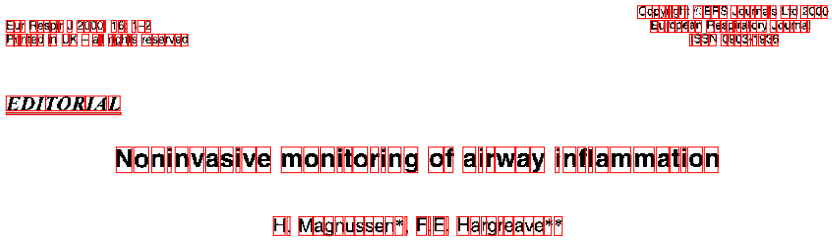
Slika 6. Izdvajanje linija



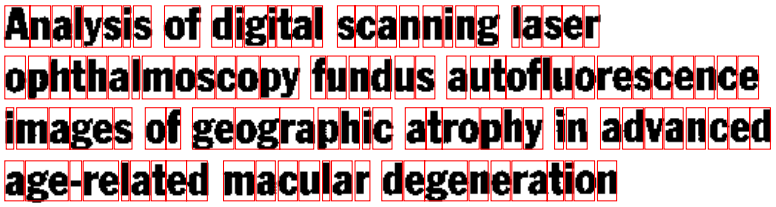
Slika 7. Horizontalna projekcija teksta

3. Rezultati segmentacije

Na slikama su dati rezultati segmentacije karaktera u nekim dokumentima.



Slika 8.



Slika 9.

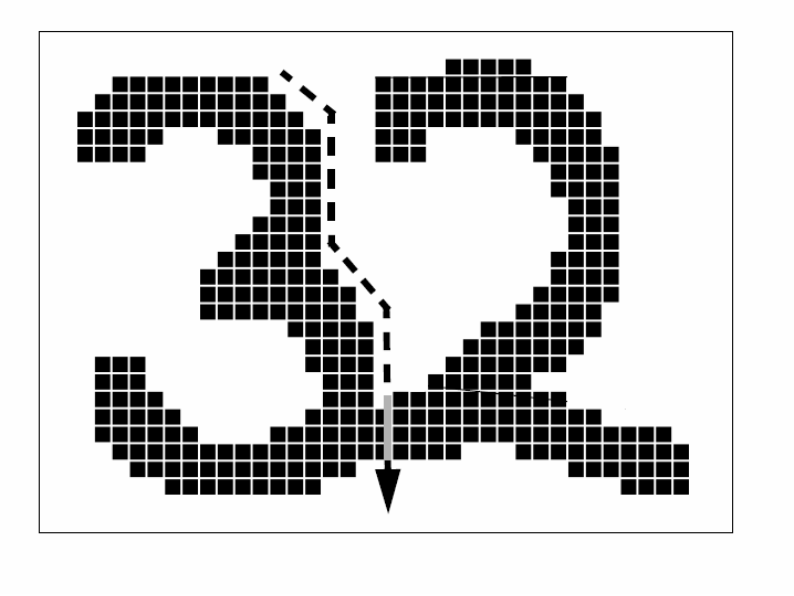
Sa slike 8. se može vidjeti da su karakteri prilično dobro segmentirani. Sledeći primjer (slika 9.) pokazuje slabosti ovog algoritma, jer karakteri koji se dodiruju ili preklapaju nisu razdvojeni. Rješenje ovog problema biće razmatrano u sledećem poglavlju, bez ulaženja u praktično rješenje.

4. Diskusija i prijedlozi daljeg poboljšanja

Može se zaključiti da dato riješenje prilično dobro segmentira štampani tekst, koji nije previše narušen šumom ili greškama prilikom obrade. Ako je kvalitet dokumenta loš mogu se pojaviti greške prilikom segmentacije. Ovo se naročito odnosi na slučajeve kada su karakteri spojeni.

Takođe greške se mogu javiti ako se vrši segmentacija kosog teksta. Ovo se dešava zbog toga što se odlučivanje vrši na osnovu horizontalnih linija, pa neki karakteri budu presječeni, ili i nakon segmentacije ostanu spojeni.

Problem spojenih karaktera se ne može uspješno riješiti na ovaj način. Rješenje može biti korištenje drugih informacija. Moguće je koristiti kontekst, kao osnovu za optimizaciju vjerovatnoća „prepoznavanja“ niza povezanih komponenti. Upotreba konteksta znači provjeravanje riječi, da bi se utvrdilo da li one postoje u rječniku.



Slika 10. Drop-falling tehnika

Jedno od boljih rješenja za segmentaciju kosih ili spojenih slova može biti neki od „drop-falling“ algoritama. Oni se zasnivaju na praćenju oblika karaktera, a „rez“ se obavlja na mjestu minimuma.

5. Literatura

[1] James R. Parker, Algorithms for imageprocessing and computer vision, John Willey and sons

[2] http://web.mit.edu/profit/PDFS/salkhanthesis.pdf

[3] http://www.iitk.ac.in/ime/veena/PAPERS/s.pdf