Danijel Radošević
Fakultet organizacije i informatike Varaždin
E-mail : Danijel.Radosevic@foi.hr

UDK: 681.3.06
Prethodno priopćenje

Postupci i problemi optičkog prepoznavanja teksta

U radu se obrađuje problematika optičkog prepoznavanja teksta kao načina unosa podataka u računalo. Opisani su osnovni postupci optičkog prepoznavanja teksta, a posebno prepoznavanje na temelju predložaka i prepoznavanje na temelju svojstava oblika. Navedeni su i osnovni problemi prepoznavanja. Posebno je opisan program Megaznak, koji je napisao autor ovog članka, a namjena mu je testiranje raznih postupaka optičkog prepoznavanja teksta.

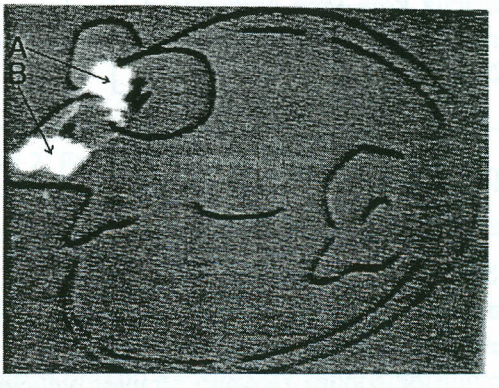
Ključne riječi: računalo, program, optičko prepoznavanje teksta (OCR), skener.

1. Uvod

Sposobnost prepoznavanja različitih objekata iz svoje okoline svojstvena je čovjeku i drugim živim bićima. Tako bez problema prepoznajemo razne predmete, slike, zvukove, odnosno simbole kao što su slova, brojke, prometni znakovi i slično. To je u većini slučajeva već toliko automatizirano da obično nismo ni svjesni postupka kojim prepoznajemo objekte iz naše okoline.

Slijedeći ilustrativan primjer treba pokazati da je prepoznavanje kod čovjeka analitički postupak i da ne prepoznajemo "odjednom", nego se i tu mogu izdvojiti pojedine faze.

Što predstavlja ovaj crtež [3; 112] ?



Slika 1. Prepoznavanje kao analitički postupak

Prema [3, str. 111] na slici se može vidjeti ili ljudsko lice, ili miš. Pokazalo se da ljudi koji na slici prepoznaju ljudsko lice, svoj pogled pretežno usmjeravaju na

područje A. Tamo se mogu prepoznati "naočale", ispod su "nos" i "usta", s desne strane može se vidjeti "uho", a i sve je zaokruženo, tako da sve skupa podsjeća na ljudsko lice. Naprotiv, ljudi koji na slici prepoznaju miša, svoj pogled pretežno usmjeravaju na područje B. Tamo se može prepoznati "mišja glava", iznad koje se nalaze "uši" (umjesto "naočala"), "usta" i "uho" kod "ljuskog lica" sada predstavljaju prednje i stražnje mišje "noge", a dobro se vidi i "rep" u donjem dijelu crteža.

Problem je naravno puno složeniji kada ga pokušamo riješiti uz pomoć računala. Naime, dok čovjek najčešće i ne razmišljajući prepoznaje, kod računala se svaka faza prepoznavanja mora vrlo precizno raščlaniti, i na kraju "pretočiti" u odgovarajući programski kod u nekom programskom jeziku.

Do sada su najbolji rezultati postignuti u optičkom prepoznavanju teksta (engl. Optical Character Recognition - OCR). Na tom području dostupni su danas već vrlo napredni komercijalni alati, koji pretvaraju bitmapu teksta unešenog preko optičkog čitača (skenera) u tekst koji se dalje može obrađivati u nekom od standardnih programa za obradu teksta. Time se izbjegava često dugotrajno i naporno unošenje tekstova u računalo preko tipkovnice.

U ovom radu bit će objašnjeni osnovni postupci i problemi optičkog prepoznavanja teksta.

2. Glavni problemi optičkog prepoznavanja teksta

Osnovni problem kod optičkog prepoznavanja teksta na računalu je taj što klasično računalo (u pravilu) ne može istovremeno obrađivati više podataka koje sadrži jedna slika, u ovom slučaju bit-mapa teksta, koja je dobivena pomoću optičkog čitača, ili nekog drugog uređaja. To je otprilike kao da na neki tekst koji želimo pročitati stavimo neprozirni papir na kojem se nalazi prozorčić kroz koji možemo vidjeti tek djelić teksta, ili točnije, tek djelić jednog znaka u tekstu! Ipak, problem je rješiv, pod uvjetom da uspijemo na pravi način definirati putanju prozorčića u zavisnosti od informacija koje kroz prozorčić dobijemo, te ako uspijemo ispravno interpretirati te informacije. Zbog toga je do sada razvijeno više raznih postupaka koji su primjenjivi na računalu.

3. Postupak prepoznavanja teksta

Postupak prepoznavanja teksta možemo u grubo podijeliti na tri faze:

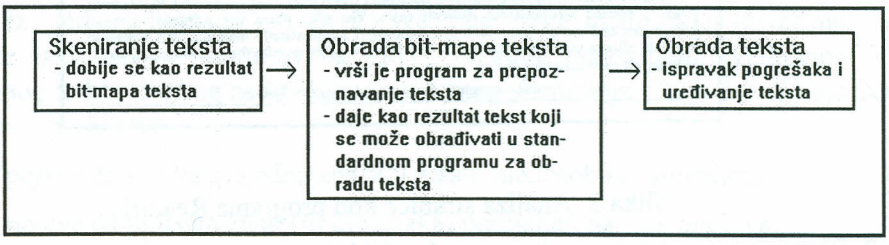
a.) Dobivanje bit-mape teksta. To se najčešće radi tako da se tekst koji se želi prepoznavati stavi u optički čitač (skener), preko kojega se bit-mapa teksta unosi u računalo. Obično se koriste ručni skeneri i stolni skeneri. Stolni skeneri daju znatno bolje rezultate jer mogu odjednom zahvatiti stranicu veličine A4 formata i nije ih potrebno rukom povlačiti preko teksta. Potrebna rezolucija skeniranja iznosi obično 200-300 DPI, a ovisi uglavnom o veličini znakova od kojih se sastoji tekst. Tako će za sitnije znakove biti potrebna veća rezolucija skeniranja. Način skeniranja je u pravilu

jednobojno skeniranje (line-art), tako da se dobivena slika sastoji samo od crnih i bijelih piksela. Osim skeniranja ponekad se primijenjuje unos bit-mape teksta u računalo preko modema, odnosno telefaksa. U tom slučaju radi se zapravo o prepoznavanju teksta koji nam netko upućuje s drugog računala, odnosno s telefaksa.

b.) Obrada bit-mape teksta pomoću programa za prepoznavanje teksta. Ovo je središnja faza prepoznavanja cilj koje je dobivanje teksta u ASCII formatu ili formatu nekog od standardnih programa za obradu teksta. Od programa za prepoznavanje teksta očekuje se u pravilu točnost prepoznavanja od najmanje 98%. Najbolji komercijalni programi mogu ostvariti točnost i preko 99%, ali uvjet za to su i kvaliteta skeniranog teksta, te kvaliteta skeniranja. Većina novijih programa za prepoznavanje teksta može prepoznavati i naša specifična slova (č,ć,đ,š i ž).

Drugi važan zahtjev je brzina prepoznavanja. Da bi program za prepoznavanje teksta uopće bio upotrebljiv, mora prepoznavati tekst barem onom brzinom kojom bi čovjek taj tekst čitao. To iznosi nekoliko stotina znakova u minuti. Ispunjenje tog zahtjeva uglavnom nije dovedeno u pitanje jer se na današnjim osobnim računalima obično postiže prepoznavanje od najmanje tisuću znakova u minuti.

c.) Obrada teksta dobivenog u fazi (b) u nekom od standardnih programa za obradu teksta. U toj fazi potrebno je najprije ispraviti pogreške nastale prilikom prepoznavanja teksta. U tome nam može pomoći program za provjeru gramatičke ispravnosti teksta (speller). Na žalost, takvi programi većinom su pisani za engleski jezik. Bilo je nekih pokušaja da se naprave i za hrvatski jezik, ali autoru ovog članka nije poznat ni jedan koji bi bio dostupan na tržištu. Nakon toga potrebno je urediti izgled teksta, odnosno podesiti margine na stranici, poravnanje teksta, font, stil i ostalo što je potrebno da bi tekst dobio svoj konačan oblik.



Slika 2. Faze u procesu prepoznavanja teksta

3.1. Obrada bit-mape teksta

Obrada bit-mape teksta je središnja faza prepoznavanja teksta, u kojoj dolaze do izražaja sve kvalitete, odnosno nedostaci programa za prepoznavanje teksta. U slijedećim poglavljima bit će opisani postupci u okviru te faze.

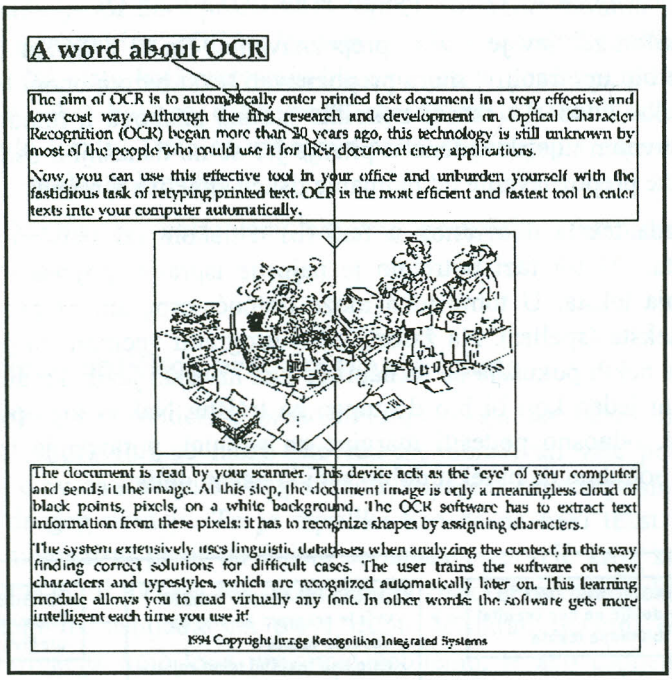
3.1.1. Analiza stranice

Prvi korak u prepoznavanju teksta je prema [7; 203] podjela stranice u blokove teksta, prema tipografskim svojstvima, kao što su lijevo i desno poravnanje. Također

je potrebno izdvojiti slike iz teksta. Ova faza prisutna je kod novijih i naprednijih alata za prepoznavanje teksta i najčešće se naziva analiza stranice (engl. Page Analisys).

Postupak analize stranice najčešće je interaktivan, odnosno, računalo daje svoj "prijedlog" analize stranice, kao što vidimo na slici 3, a zatim korisnik može eventualno unijeti korekcije. Tako se mogu promijeniti područja koja obuhvaćaju pojedini blokovi teksta, redoslijed prepoznavanja, odnosno neki dijelovi teksta mogu se izuzeti od prepoznavanja.

Uloga ove faze je priprema za prepoznavanje teksta jer se u njoj određuje što zapravo treba prepoznavati.



Slika 3. Analiza stranice kod programa Readiris

3.1.2. Prepoznavanje teksta unutar pronađenih blokova

Nakon što su postupkom analize stranice označeni blokovi teksta koje treba prepoznavati, potrebno je izvršiti slijedeće faze prepoznavanja :

a.) izdvojiti pojedine znakove u bit-mapi teksta,

b.) pridružiti svakom znaku odgovarajući kod (npr. ASCII kod) i

c.) od dobivenih kodova sastaviti tekst.

3.1.2.1. Izdvajanje pojedinih znakova u bit-mapi teksta

Izdvajanje pojedinih znakova u bit-mapi teksta je faza prepoznavanja u kojoj treba identificirati pojedini predmet prepoznavanja - jedan znak. To je prema

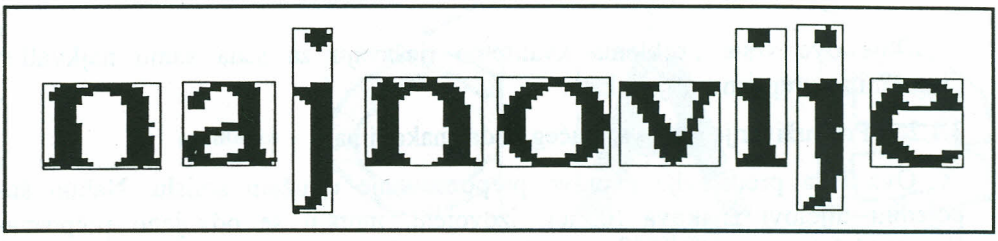
iskustvima autora ovog članka najosjetljivija faza prepoznavanja teksta, i to zbog toga što većina pogrešaka kod prepoznavanja teksta ima svoj uzrok nekoj pogrešci koja je nastala u ovoj fazi.

Izdvajanje pojedinih znakova u bit-mapi teksta sastoji se od dvije faze :

a.) Izdvajanje svakog pojedinog oblika (nakupine "tinte") iz okoline i

b.) Izdvajanje pojedinog znaka iz okoline.

Ovdje je problem to što se mnogi znakovi sastoje od više dijelova, npr. znak "%" sastoji se od tri dijela : dva kružića i kosa crta između njih. Potrebno je izdvojiti svaki pojedini dio znaka, prepoznati ga (tj. pridružiti mu odgovarajući kod), a zatim tako prepoznate dijelove znaka sastaviti i pridružiti im jedinstveni kod.



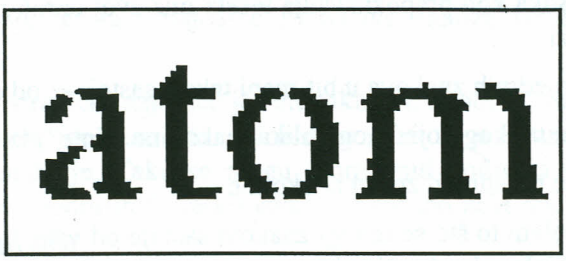
Slika 4. Izdvajanje pojedinih znakova u tekstu

Pojedini oblik izdvaja se iz okoline najčešće tako da se pođe od pretpostavke da jedan oblik čine sve međusobno povezane točke iste boje. Budući da je tekst jednobojno skeniran, da je pozadina u pravilu bijela, to jedan oblik ćine sve međusobno povezane crne točke. To znači da se npr. za slovo "a" pretpostavlja da se sastoji iz "jednog komada", tj da su sve crne točke međusobno povezane. To je najčešće tako, ali ne mora uvijek biti, i to zbog nedovoljno kvalitetnog otiska skeniranog teksta ili zbog nedovoljno kvalitetnog skeniranja. Tako dolazi najćešće do dvije vrste problema:

a.) pojave da su dva susjedna znaka u tekstu međusobno "slijepljeni" i

b.) pojave da je jedan znak (ili dio znaka) podijeljen na više dijelova.

Slika 5. "Slijepljeni" znakovi ("k" i "r")



Slika 6. Jedan znak podijeljen je na više dijelova (slovo "a" na tri, "o" i "m" na dva dijela)

Obje ove vrste problema kvalitetno rješavaju za sada samo najkvalitetniji programi za prepoznavanje teksta.

3.1.2.2. Pridruživanje odgovarajućeg koda znakovima

Ova faza predstavlja zapravo prepoznavanje u užem smislu. Nakon što su pojedini dijelovi znakova (oblici) izdvojeni, moraju se odvojeno prepoznavati. Komercijalni programi koriste uglavnom dvije vrste postupaka, koji se često međusobno kombiniraju :

a.) prepoznavanje na temelju predložaka i

b.) prepoznavanje na temelju svojstava oblika

3.1.2.2.1. Prepoznavanje na temelju predložaka

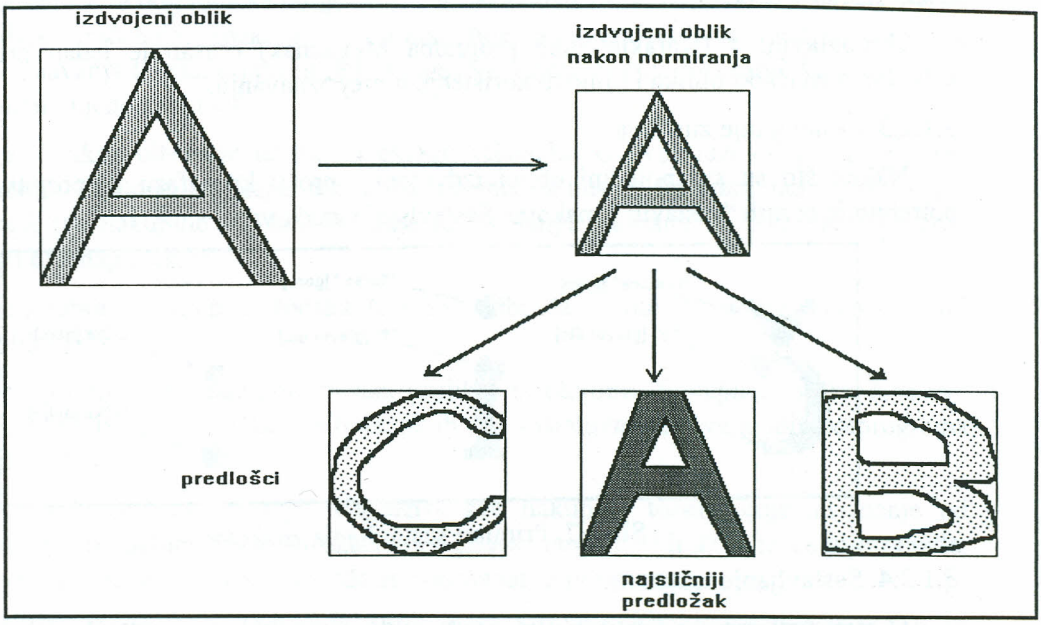
Prepoznavanje na temelju predložaka temelji se na korištenju gotovih predložaka s kojima se uspoređuju svi oblici izdvojeni iz bit-mape teksta. Obično se radi o binarnim matricama veličine do 16 * 16 (program ExperVision prema [7; 204] koristi binarne matrice veličine 10*10). Veća matrica omogućuje zapisivanje više detalja oblika, ali njena obrada duže traje, pa usporava prepoznavanje. Svakoj takvoj matrici pridružen je kod (obično ASCII kod) znaka koji ta matrica predstavlja.

Da bi se svaki izdvojeni pojedinačni oblik mogao usporediti s predlošcima, potrebno je provesti njegovo normiranje, tj. transformaciju koja će ga učiniti usporedivim s predlošcima.

Nakon normiranja oblika potrebno je provesti postupak uspoređivanja binarnih matrica normiranog oblika i svakog pojedinog predloška[1]. Kao rezultat tog uspoređivanja dobije se stupanj sličnosti, koji se obično izražava u postocima. Svrha

[1]Ponekad se nastoji izbjeći uspoređivanje normiranog oblika sa svim pojedinim predlošcima, ponajviše zbog povećanja brzine procesa prepoznavanja. U tom slučaju potrebno je izlučiti neke karakteristike normiranog oblika na temelju kojih se može s većom ili manjom sigurnošću zaključiti da je uspoređivanje s nekim predlošcima suvišno (vidi: poglavlje 4, Karakteristike programa Megaznak).

tog postupka je utvrđivanje stupnja sličnosti sa svakim pojedinim predloškom, te pronalaženje najsličnijeg predloška i njegovog koda.



Slika 7. Prepoznavanje na temelju predložaka

3.1.2.2.2. Prepoznavanje na temelju svojstava oblika

Prepoznavanje na temelju svojstava oblika zasniva se na utvrđivanju određenih svojstava oblika, odnosno njihovih osnovnih značajki. Prema [1,str. 23] te osnovne značajke moraju zadovoljavati slijedeće kriterije :

a.) Niska dimenzionalnost. Osnovne značajke uzorka moraju biti relativno malobrojne zato što njihov broj utječe na vrijeme prepoznavanja uzoraka, na složenost postupka razvrstavanja i na broj uzoraka za učenje koji su potrebni za zadovoljavajuće prepoznavanje.

b.) Dovoljna informacija. Osnovne značajke uzorka moraju sadržavati dovoljno informacije za djelotvorno razvrstavanje uzoraka[2]. Proces raspoznavanja oblika možemo predočiti preslikavanjem mjernog prostora, koji je višedimenzionalan, u jednodimenzionalni prostor odluke. Može se dogoditi da samo jedna značajka može sadržavati dovoljnu informaciju za raspoznavanje oblika.

c.) Geometrijska postojanost. Osnovna je geometrijska pretpostavka koju upotrebljavamo u gotovo svim postupcima raspoznavanja da mala udaljenost među uzorcima u prostoru uzoraka ujedno znači i malu razliku u svojstvima objekata raspoznavanja mjerodavnima za raspoznavanje.

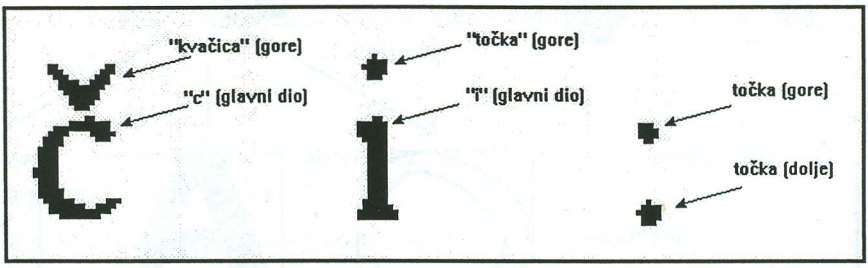
[2]Predmet prepoznavanja općenito nazivamo uzorkom. Navedeni kriteriji koje moraju zadovoljavati znaèajke oblika vrijede i za ostale vrste uzoraka (npr. izgovorene rijeèi, slike i sl.).

d.) Postojanost značajki. Značajke su namijenjene za uspoređivanje uzoraka, pa zbog toga treba uzeti u obzir njihovu vremensku postojanost, odnosno nepostojanost. Ovaj kriterij naveden je ovdje radi potpunosti jer se uglavnom ne odnosi na raspoznavanje teksta.

U poglavlju 4 (Karakteristike programa Megaznak) opisan je jedan primjer izdvajanja značajki oblika i njihovo korištenje u prepoznavanju.

3.1.2.3. Sastavljanje znakova

Nakon što su svi pojedini oblici izdvojeni i prošli kroz fazu prepoznavanja, potrebno je iz njih "sastaviti" znakove. Sastavljanje znakova



Slika 7. Primjeri sastavljanja znakova

3.1.2.4. Sastavljanje teksta

U ovoj fazi potrebno je dobivene znakove urediti u redove teksta. Ponekad nije sasvim jednostavno izdvojiti redove teksta, npr. ako je tekst skeniran ukoso, pa treba korigirati nagib teksta(taj problem osobito je izražen kod tekstova koji su skenirani ručnim skenerom). Drugi problem može biti razdvajanje riječi u tekstu, odnosno određivanje broja znakova razmaka koji se moraju pridružiti prepoznatim znakovima. Jedan jednostavan postupak sastavljanja teksta opisan je u poglavlju 4.1., točka e.

4. Karakteristike programa Megaznak

Program Megaznak napisao je autor ovog teksta sa svrhom ispitivanja raznih postupaka optičkog prepoznavanja teksta. Program je napisan u programskom jeziku Turbo Pascal, a radi pod operativnim sustavom MS DOS. Od strojne opreme zahtijeva Pc računalo opremljeno procesorom 80286 ili jačim i VGA ili EGA grafičkom karticom. Poželjno je također i barem 4 Mb radne memorije računala, radi ubrzavanja vrlo intenzivne komunikacije s diskom.

Ulaz za program je datoteka u PCX grafičkom formatu, koja sadrži bit-mapu skeniranog teksta. Način skeniranja je jednobojno skeniranje (line-art), a rezolucija između 200 i 300 DPI.

4.1. Faze prepoznavanja teksta kod programa Megaznak

Optičko prepoznavanje teksta kod programa Megaznak podijeljeno je u slijedećih nekoliko faza :

a.) Pretvaranje bitmape skeniranog teksta iz formata PCX u interni format. Između više različitih grafičkih formata za bit-mapu teksta izabran je PCX jer on predstavlja kompromis između zahtjeva za što boljom kompresijom bit-mape i brzine njene obrade. Zbog toga se taj grafički format koristi za pohranjivanje bit-mapi teksta na disku računala. Ipak, i taj format nije pogodan radi specifičnih zahtjeva prepoznavanja teksta, u prvom redu mogućnosti brze obrade bit-mape, koja uključuje i izmjenu njenog sadržaja.

b.) Prikaz bit-mape teksta na ekranu računala. U ovoj fazi dolazi do prikaza pojedinih dijelova bit-mape teksta na ekranu računala. Program tada pronalazi i izdvaja sve nakupine točaka, obrađuje ih i uklanja s ekrana, ali i iz datoteke koja sadrži bit-mapu teksta.

c.) Obrada nakupina točaka (oblika). Obrada nakupina točaka sastoji se od nekoliko podfaza:

c.1.) Izdvajanje nakupine točaka (oblika) iz okoline. Nakupinu točaka čine sve međusobno povezane točke iste boje, pa ih je korištenjem odgovarajućih potprograma moguće izdvojiti.

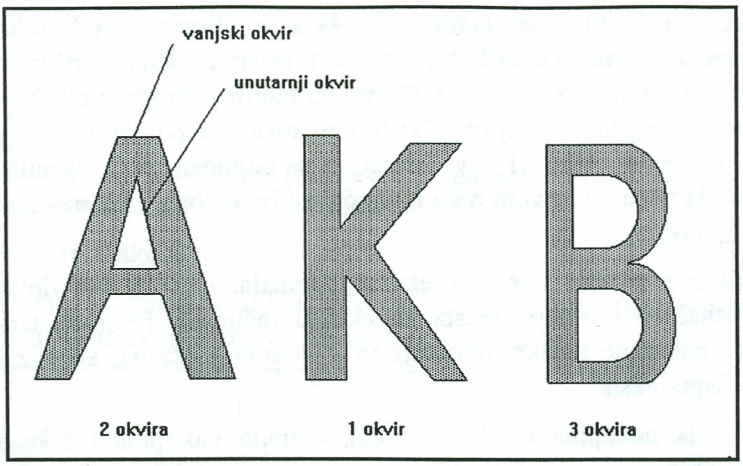
c.2.) Uklanjanje smetnji. Megaznak sve nakupine točaka koje su manje od zadanog minimalnog broja točaka tretira kao smetnje i isključuje iz daljnje obrade. Zadani minimalni broj točaka može se podešavati u programu.

c.3.) Normiranje. Izdvojeni oblik mora biti usporediv s predlošcima koji se nalaze u memoriji računala. Zbog toga se vrši kopiranje oblika u standardni okvir koji je kod Megaznaka veličine 16 * 16 točaka. Oblici koji ne stanu u standardni okvir transformiraju se (vrši se preračunavanje koordinata točaka), a ostali se kopiraju bez transformacije.

c.4.) Izdvajanje osnovnih značajki oblika. Izdvajaju se dvije osnovne značajke (slika 8) :

- broj okvira znaka i
- veličina vanjskog okvira.

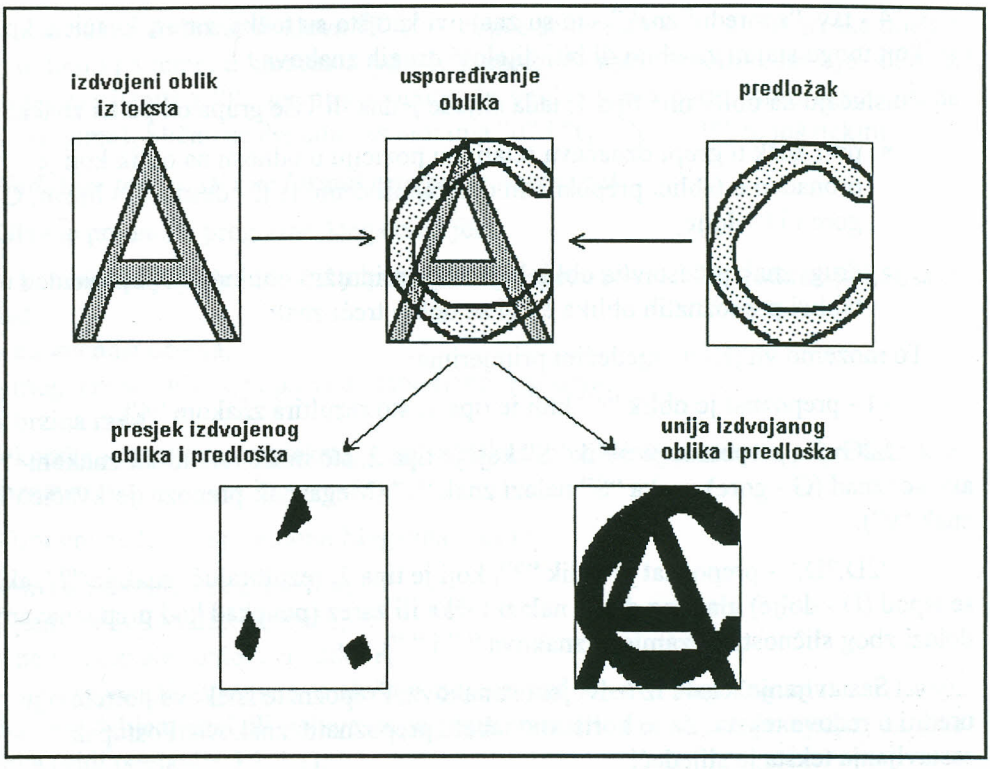
Podaci o ove dvije značajke oblika nalaze se u memoriji računala uz svaki znak koji služi kao predložak za prepoznavanje. Izdvajanjem osnovnih značajki oblika dobivamo filter pomoću kojega postižemo da više ne moramo ispitivati stupanj sličnosti izdvojenog oblika sa svim predlošcima, nego samo s onima koji sadrže osnovne značajke izdvojenog oblika. Pri tome se broj okvira oblika mora podudarati, dok veličina vanjskog okvira (izražena u broju rubnih točaka koje čine okvir) smije odstupati do 20%. Time se značajno povećava brzina, ali i točnost prepoznavanja teksta.



Slika 8. Izdvajanje osnovnih značajki oblika.

c.5.) Uspoređivanje izdvojenog oblika s predlošcima za prepoznavanje. Izdvojeni oblik uspoređuje se sa svim zapisanim predlošcima kod kojih odgovaraju osnovne značajke oblika. Uspoređuje se najprije veličina izdvojenog oblika (prije normiranja) sa veličinom predloška (također prije normiranja). Ukoliko se ni visina, ni širina izdvojenog oblika ne razlikuju od visine, odnosno širine predloška za više od granične vrijednosti, koja iznosi oko 15%, tada se prelazi na utvrđivanje stupnja podudarnosti binarnih matrica ova dva oblika (slika 9).

Predložak za koji se ustanovi najveći stupanj podudarnosti s izdvojenim oblikom smatramo najsličnijim, pa se njegov kod pridružuje izdvojenom obliku i zapisuje u tabelu prepoznatih oblika, zajedno s koordinatama koje označavaju poziciju na kojoj se unutar bit-mape teksta nalazi prepoznati oblik. Ipak pokazalo se da dolazi do pogrešaka u prepoznavanju nekih znakova čije su bit-mape slične, tako je npr. znakove "e" i "o" dosta teško razlikovati. Zbog toga, Megaznak u slučaju prepoznavanja tih znakova (te još nekih, kao npr. "f" i "t", "n" i "u", itd.) pokreće postupak za njihovo razlikovanje. Postupak se sastoji od provjeravanja onih dijelova tih znakova koji se međusobno najviše razlikuju. Kod znakova "e" i "o" to je središnji dio znaka. Rezultat ovog postupka može biti korekcija prethodnog rezultata prepoznavanja.



Slika 9. Utvrđivanje stupnja podudarnosti binarnih matrica oblika. Broj točaka u presjeku podijeljen s brojem točaka u uniji izdvojenog oblika i predloška daje stupanj sličnosti oblika s predloškom.

d.) Sastavljanje znakova iz izdvojenih oblika. Neki znakovi, kao što su npr. "A","8","$" i drugi sastoje se od samo jednog oblika, neki od dva oblika (npr. "Č","!",";", itd.), a neki i od tri oblika (npr. "%"). Program Megaznak prepoznaje svaki izvojeni oblik i pridružuje mu odgovarajući kod, koji zapisuje u tabelu prepoznatih oblika. Na kraju je potrebno znakove "sastaviti", tj. svim oblicima koji čine jedan znak pridružiti jedinstveni kod, te takav "sastavljeni" znak upisati u tabelu prepoznatih znakova, zajedno s njegovim koordinatama unutar bit-mape teksta. To se radi pomoću odgovarajućih pravila. Način zapisivanja pravila za sastavljanje znakova je slijedeći :

- prvi znak predstavlja oblik koji je pronađen u tablici prepoznatih oblika i koji ulazi u sastav znaka,

- drugi znak predstavlja tip oblika:
  1 - znak se sastoji iz samo jednog oblika,
  2 - znak se sastoji, ili se može sastojati od dva oblika,
  3 - znak se sastoji ili se može sastojati od tri oblika i

4 - tzv. "sporedni znak" - to su znakovi kao što su točka, zarez, kvačica, kružić i sl. koji mogu stajati zasebno ili biti dijelovi drugih znakova,

- u slučaju da oblik nije tipa 1, tada slijede jedna ili više grupa od po tri znaka:
  - prvi znak u grupi označava relativnu poziciju u odnosu na oblik koji je pronađen u tablici prepoznatih oblika, pri čemu je E - desno, L - lijevo, G - gore i D - dolje,
  - drugi znak predstavlja oblik koji treba pridružiti obliku koji je pronađen u tablici prepoznatih oblika da bi se dobio treći znak.

To možemo vidjeti u slijedećim primjerima:

41 - prepoznat je oblik "4" koji je tipa 1, što rezultira znakom "4",

S2GvŠ - prepoznat je oblik "S" koji je tipa 2, što može rezultirati znakom "Š", ako se iznad (G - gore) znaka "S" nalazi znak "v" (Megaznak prepoznaje kvačicu kao znak "v").

?2D.?D,? - prepoznat je oblik "?", koji je tipa 2, rezultirajući znak je "?", ako se ispod (D - dolje) glavnog dijela nalazi točka ili zarez (ponekad kod prepoznavanja dolazi zbog sličnosti do zamjene znakova "." i ",").

e.) Sastavljanje teksta iz izdvojenih znakova. Prepoznate znakove potrebno je urediti u redove teksta. Za to koristimo tabelu prepoznatih znakova. Postupak sastavljanja teksta je slijedeći :

1. Tabela prepoznatih znakova sortira se uzlazno prema ordinati, tako da dobijemo sve znakove u tabeli poredane od vrha prema dnu bit-mape teksta,

2. Svi znakovi od početnog do posljednjeg čija ordinata gornjeg ruba je manja od ordinate početnog znaka uvećana za prosječnu širinu reda teksta čine jedan red iupisuju se u tabelu reda,

3. Tabela reda sortira se uzlazno prema apscisi, time su svi znakovi poredani od prvog do zadnjeg u redu,

4. Znakovi se prepisuju u tekstualnu datoteku koja sadrži konačan rezultat prepoznavanja - gotovi tekst u formatu ASCII. Pri tom je potrebno na odgovarajuća mjesta u tekstu ubaciti znak razmaka, što se čini na svim onim mjestima u redu gdje je razmak između dva znaka veći od polovice prosječne širine znaka.

5. Postupak se ponavlja od koraka (2) dok se ne obuhvate svi znakovi u tabeli prepoznatih znakova.

Kod sastavljanja teksta mogu nastati problemi, osobito ako je tekst skeniran ukoso. Tada može doći do pogrešnog sastavljanja redova teksta. Zbog toga Megaznak omogućuje korekciju nagiba redova teksta na način da korisnik unosi faktor korekcije, koji može biti pozitivan ili negativan, ovisno o smjeru nagiba teksta.

f.) Korekcija prepoznatog teksta. Do nekih pogrešaka u toku prepoznavanja teksta dolazi dosta često i teško ih je ispraviti u prethodnim fazama prepoznavanja. Na

primjer, u nekim fontovima slova "I" i "l" uopće se ne razlikuju, ali se greška obično može ukloniti na temelju konteksta (ako je prepoznata riječ "svjetlo", tada je očito da "I" treba zamijeniti s "l", jer nije moguće da veliko slovo bude usred riječi pisane malim slovima). Slično se događa i sa slovima "o" i "O", "p" i "P", te još nekim.

4.2. Sadašnje prednosti i nedostaci programa Megaznak

Glavne prednosti programa Megaznak jesu:

- kombiniranje više različitih postupaka prepoznavanja, što znatno poboljšava točnost,
- mogućnost učenja,
- mogućnost zadavanja pravila sastavljanja znakova,
- brzina rada i
- korekcija prepoznatog teksta, gdje se naknadno otkloni jedan dio pogrešaka kod prepoznavanja.

Osnovni nedostaci programa Megaznak jesu :

- ne radi direktno sa skenerom,
- nema prave analize stranice,
- ne raspoznaje fontove ni stilove,
- ne prepoznaje ispravno u slučaju pojave kad su dva susjedna znaka u tekstu međusobno "slijepljena" ili pojave da je jedan znak (ili dio znaka) podijeljen na više dijelova (vidi poglavlje 3.1.2.1),
- određeni nedostaci kod sastavljanja redova teksta, jer ne može automatski izvršiti korekciju, ako je tekst skeniran ukoso.

5. Zaključak

U radu su navedena autorova dosadašnja dostignuća na području optičkog prepoznavanja teksta, uključujući i program Megaznak, koji je napisan radi testiranja raznih postupaka prepoznavanja. Ovom području trebat će posvetiti ubuduće više pažnje jer se optičkim prepoznavanjem teksta znatno kvalitetnije nego do sada rješava velik dio problema unosa podataka u računalo, što danas sve više postaje usko grlo u primjeni računala, osobito u području obrade teksta.

Rječnik upotrijebljenih pojmova

ASCII - American Standard Code for Information Interchange - američki standardni kod za razmjenu informacija. Tekst u ASCII formatu može se obrađivati u skoro svakom programu za obradu teksta.

DPI - engl. Dot Per Inch - mjera za rezoluciju skeniranja. Predstavlja broj točaka na koji će se rastaviti jedan Inch nekog predloška za skeniranje (obično slika ili tekst). Tako rezolucija od 200 DPI znači da će se svaki kvadratni Inch predloška za skeniranje rastaviti na matricu od 200 * 200 točaka. Ponekad se, ali rjeđe,

horizontalna i vertikalna rezolucija mogu razlikovati. Tada se moraju navesti obje rezolucije, a obje se izražavaju u DPI.

Font je prema [5; 133] "skup svih znakova raspoloživih za odgovarajući oblik i izgled grafičkog prikaza znakova".

Jednobojno skeniranje (line-art) je prema[6; 15] "način rada u kojem skener za svaku analiziranu točku (piksel) odredi samo da li je crn ili bijel, a dobivena bitmapa sastoji se samo od crnih i bijelih piksela. Pohranjivanje slike zahtijeva 1 bit po svakom pikselu".

OCR (engl. Optical Character Recognition) - optičko prepoznavanje znakova.

Optički čitač (skener) je uređaj za unos slika iz tradicionalnih izvora (dokumenti, fotografije, tisak, i sl.) u obliku bit-mape u računalo.

Piksel (kratica od engl. picture element - element slike) je jedna točka u bit mapi slike koja se obrađuje na računalu.

Program za provjeru gramatičke ispravnosti teksta (speller) traži svaku pojedinu riječ iz nekog teksta u rječniku, te upozorava u slučaju da riječ ne postoji u rječniku.

Literatura

[1] L. Gyergyek, N. Pavešić, S. Ribarić: Uvod u raspoznavanje uzoraka, Tehnička knjiga, Zagreb, 1988.

[2] M. Kantardžić, A. Delić: Principi raspoznavanja uzoraka, Svjetlost, Sarajevo, 1984.

[3] B. Petz: Psihologija u ekonomskoj propagandi, Društvo ekonomskih propagandista Hrvatske, Zagreb, 1974.

[4] D. Radošević : Prepoznavanje dvodimenzionalnih oblika i izgovorenih riječi korištenjem sličnih algoritama. U: Zbornik radova Fakulteta organizacije i informatike, Varaždin, 1990.

[5] F. Ružić : Multimedija : integracija zvuka, slike i teksta uz pomoć osobnog računala, Mozaik knjiga, Zagreb, 1994.

[6] K. Vlaši : Priručnik o skeniranju, Kristal print, Zagreb, 1995.

[7] Peter Wayner. Optimal Character Recognition. Byte 12/93, str. 203 - 210.

Primljeno: 1996-02-21

Radošević D. Optical character recognition - procedures and problems

Summary

By increasing the data processing power of modern computers, the biggest problem in data processing becomes data entry, the necessary phase in all data processing. This paper concerns Optical Character Recognition (OCR). OCR is a method by which we can often solve problems of data entry when we already have our documents on standard media like paper.