

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 4579

**Očitavanje studentskih matičnih
brojeva unesenih zacrnjivanjem
predloška znamenki**

Jure Čular

Zagreb, srpanj 2016.

Zagreb, 17. ožujka 2016.

ZAVRŠNI ZADATAK br. 4579

Pristupnik: **Jure Čular (0036479001)**
Studij: Računarstvo
Modul: Računarska znanost

Zadatak: **Očitavanje studentskih matičnih brojeva unesenih zacrnjivanjem predloška znamenki**

Opis zadatka:

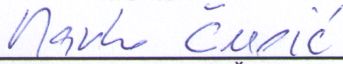
Pisanje provjera znanja u velikim grupama studenata postavlja niz izazova. Provjere znanja ponekad se provode izravno na nastavi tijekom predavanja a svrha im je pružanje brze povratne informacije studentima. Kako bi takve provjere bile bodovane u što kraćem vremenu, poželjno ih je automatski (računalno) ispravljati. Jedan od problema u tom pristupu je automatsko očitavanje studentovog matičnog broja koji na papiru s pitanjima student može unijeti na različite načine.

U okviru ovog rada potrebno je razmotriti problem lokalizacije te očitavanja matičnog broja koji je unesen zacrnjivanjem unaprijed pripremljenih predložaka znamenaka. Krenuti od pretpostavke da je na papiru otisnut niz predložaka znamenki (oblikovanih kao znamenke kod 7-segmentnih prikaznika) u koje je student unio svoj matični broj zacrnjivanjem. Potrebno je na dostupnom skupu podataka implementirati odgovarajuće prototipno rješenje opisanog problema. Za implementirani postupak potrebno je provesti vrednovanje. Radu je potrebno priložiti algoritme, izvorne kodove i rezultate uz potrebna objašnjenja i dokumentaciju. Citirati korištenu literaturu i navesti dobivenu pomoć.

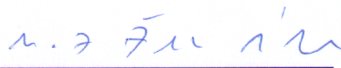
Zadatak uručen pristupniku: 18. ožujka 2016.

Rok za predaju rada: 17. lipnja 2016.

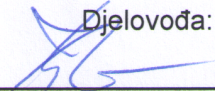
Mentor:


Doc. dr. sc. Marko Čupić

Predsjednik odbora za
završni rad modula:


Prof. dr. sc. Siniša Srbljić

Djelovođa:


Doc. dr. sc. Tomislav Hrkać

Zahvala mentoru doc. dr. sc. Marku Čupiću na pomoći i prijedlozima.

SADRŽAJ

1. Uvod	1
2. Predobrada slike	2
2.1. Pretvaranje slike u sivu	2
2.2. Binarizacija	2
2.2.1. Histogram	3
2.2.2. Otsu-ova metoda	4
2.2.3. Primjer postupka binarizacije	5
3. Segmentacija	10
3.1. Segmentacija pomoću projekcija	10
3.1.1. Projekcije	10
3.1.2. Postupak segmentacije pomoću projekcije	11
3.1.3. Nedostatci metode	15
3.2. Segmentacija pomoću traženja prvog crnog slikovnog elementa sa svake strane	16
3.2.1. Određivanje prvog crnog slikovnog elementa	16
3.2.2. Postupak segmentacije pomoću traženja prvog crnog slikovnog elementa sa svake strane	17
3.2.3. Nedostatci metode	18
4. Raspoznavanje	19
4.1. Postupak raspoznavanja znamenaka	19
4.2. Kodiranje znamenaka	24
5. Ispitni skup podataka	26
5.0.1. Pogrešno skenirani predlošci	26
5.0.2. Pogrešno zacrnjeni predlošci	26

6. Rezultati	28
6.1. Rezultati segmentacije	28
6.2. Rezultati prepoznavnja	29
6.3. Analiza rezultata	33
7. Zaključak	36
Literatura	37

1. Uvod

U svrhu rješavanja jednog od izazova koje predstavlja ispravljanje pisanih provjera, a to je identifikacija pristupnika, razmatra se rješenje očitavanja identifikacijskog broja (JMBAG) pomoću strojnog očitavanja. Umjesto korištenja prethodno označenih provjera, student bi označio svoj identifikacijski broj te bi se iz predložaka tijekom ispravljanja provjera strojno očitali identifikacijski brojevi studenata. Predložak se sastoji od predložaka sedam segmentnih znamenaka, svaka znamenka se sastoji od sedam ćelija (segmenta) koje student zacrni kako bi označio pojedinu znamenku. Takvim pristupom smanjuje se potreba za prethodnom pripremom provjera za svakog studenta.

Sustav se sastoji od tri dijela: predobrada slike, razdvajanja znamenaka odnosno segmentacija te prepoznavanje znamenaka. Programsko rješenje razvijeno je u programskom jeziku *Java*. U daljnjem tekstu opisuju se metode koje se koriste za pripremu slike za očitavanje, metode korištene za segmentaciju te metoda korištena za očitavanje. Ostatak rada organiziran je kako slijedi. U poglavlju 2 opisana je pretvorba slike u sivu sliku i binarizacija. Potom su u poglavlju 3 opisani postupci segmentiranja predložaka. Zatim u poglavlju 4 slijedi opis postupka raspoznavanja znamenaka. U poglavlju 5 opisan je ispitni skup podataka pomoću kojeg je sustav vrednovan. Zatim slijedi poglavlje 6 u kojem su predstavljeni rezultati sustava.

2. Predobrada slike

Prije očitavanja znamenaka, slike je potrebno obraditi. U postupak prethodne obrade slike spadaju postupak pretvaranja slike u sivu te postupak binarizacije slike. Proces predobrade slike prikazan je na slici 2.1.

2.1. Pretvaranje slike u sivu

Svaka učitana slika sadrži slikovne elemente koje su definirani pozicijom i intenzitetom. Intenzitet opisuje boju slikovnog elementa te se tipično sastoji od tri komponente: intenzitete crvene, plave i zelene. Svaka od komponenti ima raspon vrijednosti 0–255. Da bi slika bila siva, za svaki slikovni element mora vrijediti da komponente intenziteta imaju jednaku vrijednost. U tu svrhu koristi se jednostavna formula za izračunavanje nove vrijednosti koja vrijedi za svaku od komponenti [5].

$$V = \frac{1}{3} \cdot R + \frac{1}{3} \cdot G + \frac{1}{3} \cdot B \quad (2.1)$$

U izrazu 2.1, R predstavlja crvenu komponentu, G predstavlja zelenu komponentu, a B predstavlja plavu. Suma koeficijenata u izrazu je jednaka 1, tako da, u slučaju da su sve komponente jednake, nova vrijednost će biti jednaka staroj.

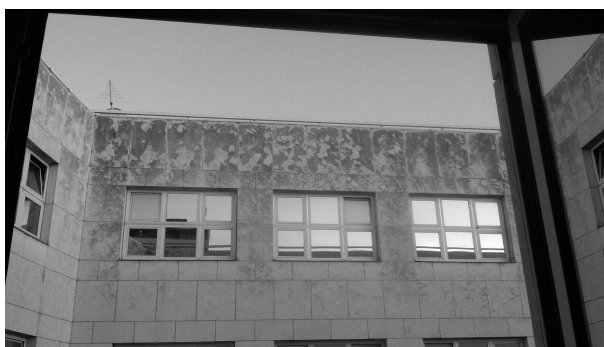
2.2. Binarizacija

Nakon što dobijemo sivu sliku, možemo započeti postupak binarizacije. Binarizaciju izvodimo kako bi svaki slikovni element slike klasificirali kao dio pozadine ili dio prvog plana. Postupak se obično sastoji od provjeravanja prelazi li jedna komponenta intenziteta¹ slikovnog elementa $I(x, y)$ određeni prag t . Ukoliko prijeđe neki prag, taj slikovni element svrstavamo kao dio pozadine (bijela boja). U slučaju da je komponenta manja od praga, svrstavamo ga kao dio prednjeg plana (crna boja). Najjednostavnija metoda binarizacije koristi unaprijed definirani prag. U praksi, takva metoda

¹Ne trebamo provjeravati sve tri komponente pošto znamo da je slika siva, pa su komponente jednake



(a) Slika u boji.



(b) Slika nakon pretvorbe u sivu.



(c) Slika nakon postupka binarizacije sive slike.

Slika 2.1: Prikaz procesa predobrade slike.

ne radi dobro pa se koriste metode koje adaptivno određuju prag. Jedna takva metoda opisana je u nastavku, a oslanja se na histogram intenziteta.

2.2.1. Histogram

Prije postupka binarizacije, potrebno je odrediti histogram pojavljivanja intenziteta slikovnog elementa sive slike. Histogram sadrži polje od 256 pretinaca, jer je raspon intenziteta slike $0 - 255$. U pretincu koji predstavlja intenzitet i nalazi se broj pojavljivanja slikovnih elemenata tog intenziteta u slici [2].

2.2.2. Otsu-ova metoda

U sklopu sustava implementirana je *Otsu-ova* metoda binarizacije (engl. *Otsu's method*). Algoritam je globalan, što znači da se za cijelu sliku određuje jedan prag, na temelju kojeg se svaki slikovni element slike klasificira u jedan od dva razreda [2]. Ovaj algoritam zahtijeva prethodno izračunat histogram pojavljivanja intenziteta slikovnih elemenata. Postoje dva načina traženja praga ovom metodom. Prvi način je traženje varijance unutar razreda, tako da se odredi težinska suma varijance prvog razreda i varijance drugog razreda te se traži prag za čiju vrijednost je varijanca unutar razreda minimalna [7]. Drugi način je traženje praga za čiju je vrijednost varijanca između razreda maksimalna. U sklopu sustava koristi se drugi način jer sadrži manje izračuna te je zato brži [2]. Postupak određivanja praga na drugi sljedeći u nastavku.

1. Prolazi se kroz svaki mogući prag od $t = 0$ do $t = L - 1$, gdje je L broj pretinaca histograma, te se za svaki t određuje:
 - (a) $w_0(t)$ i $w_1(t)$, odnosno težine razreda, dobivene izrazima 2.3 te 2.4, gdje je p_i vjerojatnost pojavljivanja intenziteta slikovnog elementa dobivena izrazom 2.2,
 - (b) $\mu_0(t)$ te $\mu_1(t)$, odnosno očekivanja svakog razreda koja se dobiju izrazima 2.5 te 2.6,
 - (c) te konačno varijanca između dva razreda $\sigma_B^2(t)$ dobivena izrazom 2.7.
2. Od svih tako dobivenih vrijednosti $\sigma_B^2(t)$, uzima se najveća te se vrijednost t za koju se dobije koristi kao prag za razdvajanje pozadine i prednjeg plana.

$$p_i(t) = \frac{\text{histogram}(i)}{\sum_{j=0}^{L-1} \text{histogram}(j)}, \quad (2.2)$$

$$w_0(t) = \sum_{i=0}^{t-1} p_i, \quad (2.3)$$

$$w_1(t) = \sum_{i=t}^{L-1} p_i = 1 - w_0(t), \quad (2.4)$$

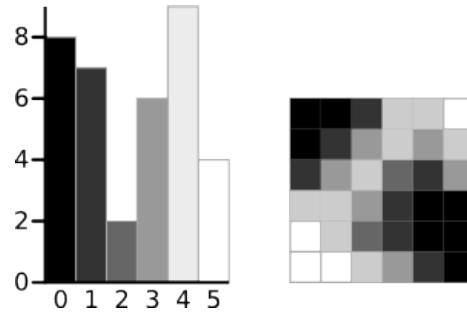
$$\mu_0(t) = \sum_{i=0}^{t-1} p_i \cdot i, \quad (2.5)$$

$$\mu_1(t) = \sum_{i=t}^{L-1} p_i \cdot i, \quad (2.6)$$

$$\sigma_B^2(t) = w_0(t) \cdot w_1(t) \cdot (\mu_0(t) - \mu_1(t))^2 \quad (2.7)$$

2.2.3. Primjer postupka binarizacije

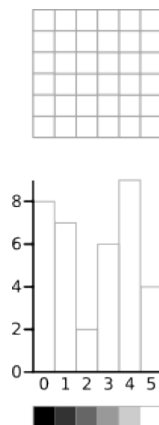
Na sljedećem primjeru jednostavne slike 2.2 i njenim histogramom, bit će prikazan postupak binarizacije *Otsu-ovom* metodom.



Slika 2.2: Primjer jednostavne sive slike (desno) koja sadrži 6 različitih intenziteta, te njoj pripadajući histogram (lijevo) [4].

Inicijalno se postavlja prag na vrijednost $t = 0$, na slici 2.3 prikazan je izgled slike nakon binarizacije uz taj prag te histogram iste slike. Vrijednosti uz prag $t = 0$ su:

- $w_0(0) = 0$,
- $w_1(0) = 1$,
- $\mu_0(0) = 0$,
- $\mu_1(0) = 2.361$,
- $\sigma_B^2(0) = 0$.



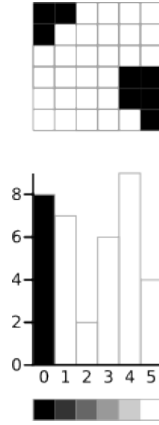
Slika 2.3: Izgled slike i histograma za prag, $t = 0$ [4].

Prvi korak je postavljanje praga $t = 1$. Na slici 2.4 vidljiv je izgled slike nakon binarizacije uz taj prag te njen histogram. Vrijednosti uz prag $t = 1$ su:

- $w_0(1) = 0.222$,

- $w_1(1) = 0.7778$,
- $\mu_0(1) = 0$,
- $\mu_1(1) = 3.0357$,
- $\sigma_B^2(1) = 1.5928$.

Najveća varijanca između razreda je trenutno $\sigma_B^2 = 1.5268$, a prag za koji je najveća je $T = 1$. Drugi korak je postavljanje praga $t = 2$. Na slici 2.5 vidljiv je izgled slike



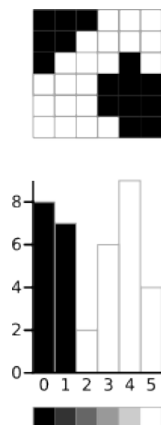
Slika 2.4: Izgled slike i histograma za prag, $t = 1$ [4].

nakon binarizacije uz taj prag. Vrijednosti uz prag $t = 2$ su:

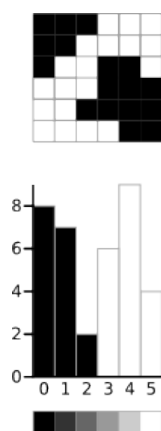
- $w_0(2) = 0.4167$,
- $w_1(2) = 0.5833$,
- $\mu_0(2) = 0.5561$,
- $\mu_1(2) = 3.7143$,
- $\sigma_B^2(2) = 2.5635$.

Najveća varijanca između razreda sada je $\sigma_B^2 = 2.5635$, a prag za koji je najveća je $T = 2$. Treći korak je postavljanje praga $t = 3$. Na slici 2.6 vidljiv je izgled slike nakon binarizacije uz taj prag te njen histogram. Vrijednosti uz prag $t = 3$ su:

- $w_0(3) = 0.4722$,
- $w_1(3) = 0.5278$,
- $\mu_0(3) = 0.6471$,
- $\mu_1(3) = 3.8947$,
- $\sigma_B^2(3) = 2.6287$.



Slika 2.5: Izgled slike i histograma za prag, $t = 2$ [4].



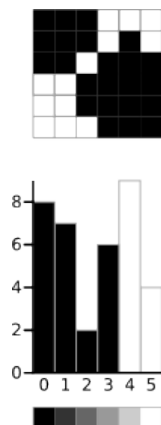
Slika 2.6: Izgled slike i histograma za prag, $t = 3$ [4].

Najveća varijanca između razreda sada je $\sigma_B^2 = 2.6287$, a prag za koji je najveća je $T = 3$. Četvrti korak je postavljanje praga $t = 4$. Na slici 2.7 vidljiv je izgled slike nakon binarizacije uz taj prag te njen histogram. Vrijednosti uz prag $t = 4$ su:

- $w_0(4) = 0.6389$,
- $w_1(4) = 0.3611$,
- $\mu_0(4) = 1.2609$,
- $\mu_1(4) = 4.3077$,
- $\sigma_B^2(4) = 2.1417$.

Najveća varijanca između razreda sada je $\sigma_B^2 = 2.6287$, a prag za koji je najveća je $T = 3$. Peti korak je postavljanje praga $t = 5$. Na slici 2.9 vidljiv je izgled slike nakon binarizacije uz taj prag te njen histogram. Vrijednosti uz prag $t = 5$ su:

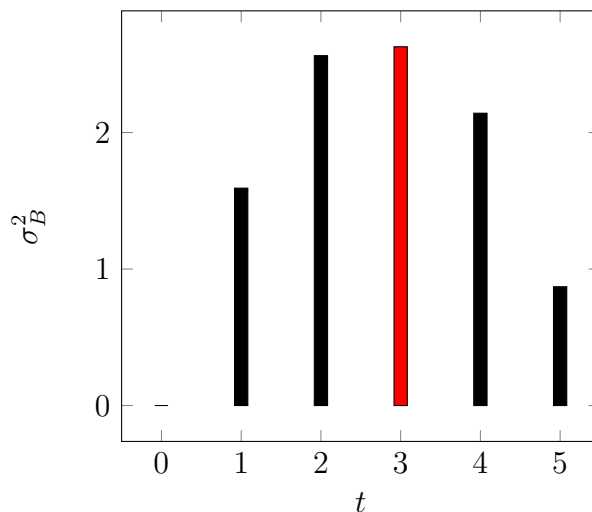
- $w_0(5) = 0.8889$,
- $w_1(5) = 0.1111$,



Slika 2.7: Izgled slike i histograma za prag, $t = 4$ [4].

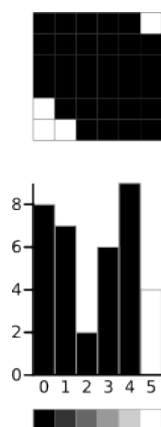
- $\mu_0(5) = 2.0313$,
- $\mu_1(5) = 5.000$,
- $\sigma_B^2(5) = 0.8705$.

Najveća varijanca između razreda sada je $\sigma_B^2 = 2.6287$, a prag za koji je najveća je $T = 3$. Graf na slici 2.8 prikazuje promjenu varijance između razreda ovisno o pragu t , crvenom bojom je označena najveća varijanca. Nakon cijelog postupka, za dani

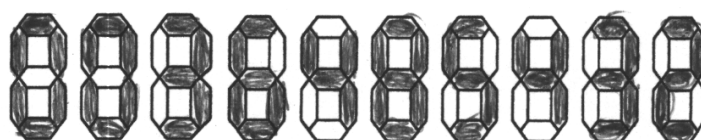


Slika 2.8: Prikaz varijance između razreda za svaki prag.

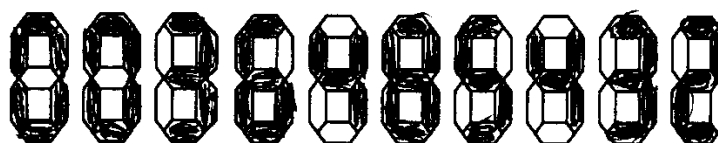
primjer najbolji prag je određen s $t = 3$. Rezultat binarizacije s tim pragom vidljiv je na slici 2.6, dok je rezultat takvog postupka na primjeru slike iz ispitnog skupa vidljiv je na slici 2.11. Slika 2.10 je slika prije postupka binarizacije navedenom metodom. Histogram slike vidljiv je na slici 2.12, gdje je crvenom crtom naznačen prag određen *Otsu-ovom* metodom.



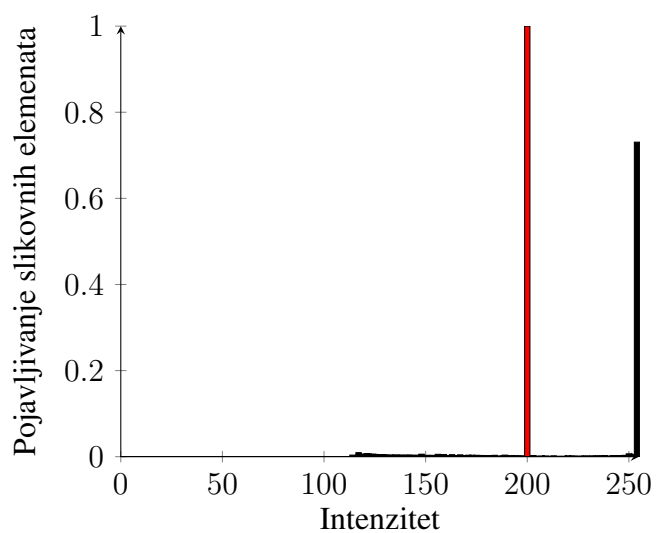
Slika 2.9: Izgled slike i histograma za prag, $t = 5$ [4].



Slika 2.10: Slika prije binarizacije.



Slika 2.11: Slika nakon binarizacije *Otsu-ovom* metodom.



Slika 2.12: Histogram slike 2.10 s naznačenim pragom binarizacije.

3. Segmentacija

Nakon obrade slike, potrebno je odijeliti svaku znamenku iz slike. Kako bi očitavanje bilo lakše potrebno je riješiti se svih praznina koje nisu potrebne za očitavanje, što uključuje bijele slikovne elemente oko svake znamenke. U sklopu sustava implementirano je više postupaka odjeljivanja znamenaka, gdje se dio postupaka zasniva na prethodnom određivanju vertikalne i horizontalne projekcije, a drugi dio na traženju prvih crnih slikovnih elemenata sa svake strane slike. Metode segmentacije koje su korištene u sklopu ovog sustava su sljedeće:

- segmentacija pomoću projekcija,
- segmentacija pomoću skaliranih projekcija, gdje su vrijednosti skalirane na interval $[0, 1]$,
- segmentacija pomoću traženja prvog crnog slikovnog elementa sa svake strane.

3.1. Segmentacija pomoću projekcija

Prva i druga metoda segmentacije znamenaka koriste prethodno određenu horizontalnu i vertikalnu projekciju.

3.1.1. Projekcije

Projekcija je suma crnih slikovnih elemenata po stupcima (broj crnih slikovnih elemenata u stupcu), odnosno retcima (broj crnih slikovnih elemenata u retku). Vertikalna projekcija je suma slikovnih elemenata po svakom stupcu slike, dok je horizontalna projekcija suma slikovnih elemenata po svakom retku slike [2]. U sklopu sustava pri određivanju projekcije crni slikovni elementi imaju vrijednost 1, a bijeli vrijednost 0. Izraz 3.1 će za dani stupac x odrediti sumu crnih slikovnih elemenata na slici visine h , dok će izraz 3.2 isto to učiniti za dani redak y , duljine w . Izraz 3.3 za dani x i y određuje je li slikovni element na poziciji (x, y) crn ili bijel. Na slici 3.2 prikazana je vertikalna projekcija slike 2.10, a na slici 3.1 horizontalna projekcija iste slike.

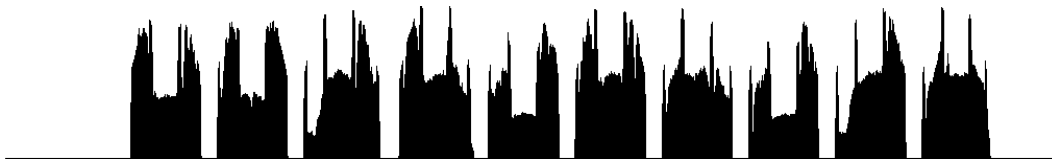
$$c(x) = \sum_{i=0}^h f(x, i) \quad (3.1)$$

$$r(y) = \sum_{j=0}^w f(j, y) \quad (3.2)$$

$$f(x, y) = \begin{cases} 0 & \text{ako } I(x, y) = 255, \\ 1 & \text{ako } I(x, y) = 0 \end{cases} \quad (3.3)$$



Slika 3.1: Horizontalna projekcija slike 2.10.



Slika 3.2: Vertikalna projekcija slike 2.10.

3.1.2. Postupak segmentacije pomoću projekcije

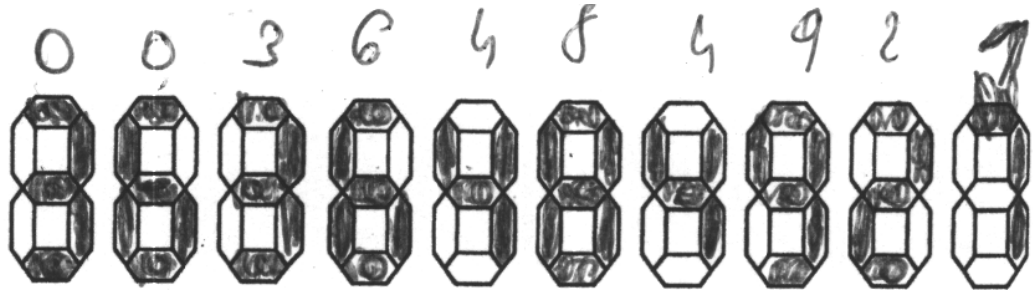
Postupak se sastoji od dva dijela. Prvi dio obuhvaća horizontalno odsijecanje, odnosno odsijecanje nebitnih slikovnih elemenata s gornje i donje strane predloška. Drugi dio obuhvaća vertikalno odsijecanje, odnosno razdvajanje svake znamenke u zasebne slike.

Za segmentaciju ovom metodom, unaprijed su postavljeni sljedeći pragovi:

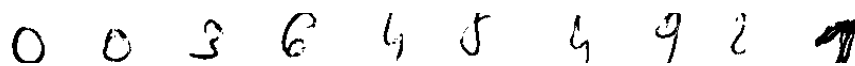
- t_h – prag zbroja crnih slikovnih elemenata u retku, kojim se određuje vertikalni početak i kraj znamenke,
- t_w – prag zbroja crnih slikovnih elemenata u stupcu, kojim se određuje horizontalni početak znamenke,
- t_{iw} – prag zbroja crnih slikovnih elemenata u stupcu, koji određuje horizontalni kraj znamenke,
- h_{min} – najmanja dopuštena visina znamenke,

– w_{min} – najmanja dopuštena širina znamenke.

Navedene pragove potrebno je ručno odrediti. Vrijednosti tih pragova ovise o rezoluciji slika i dimenzijama predložaka. U svrhu izbjegavanja smetnji na slikama uvedeni su pragovi w_{min} , h_{min} i t_{iw} . Prag h_{min} je uveden kako bi se izbjegle smetnje kao na slici 3.3. Naime bez tog praga odsječeni dio bi izgledao kao na slici 3.4.

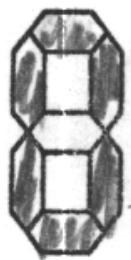


Slika 3.3: Primjer predložka sa smetnjom u gornjem dijelu slike.



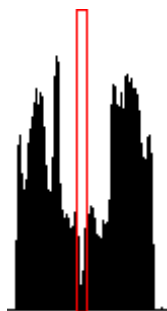
Slika 3.4: Horizontalno odsijecanje slike 3.3 bez praga h_{min} .

Prag t_{iw} je uveden kako bi se izbjegle smetnje kao na slici 3.5. Slika 3.6 prikazuje vertikalnu projekciju slike 3.5, crvenim pravokutnikom naznačen je problematičan stupac čija će suma crnih slikovnih elemenata biti manja od praga t_w . Ako bi prag t_{iw} bio jednak pragu t_w , u ovom primjeru znamenka na kojoj se nalazi ta smetnja ne bi bila odsječena jer bi se zbog rada metode smatrala smetnjom odnosno jer ne zadovoljava prag w_{min} .

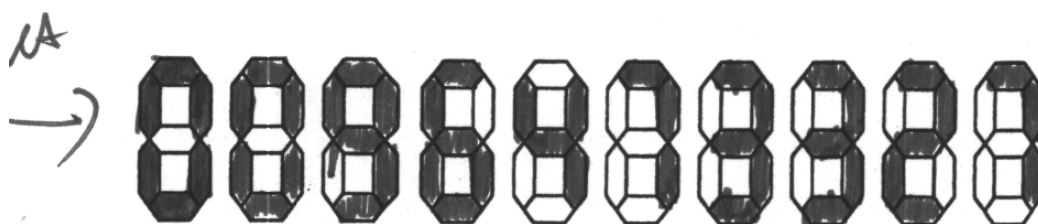


Slika 3.5: Primjer znamenke sa smetnjom.

Prag w_{min} uveden kako bi se izbjegle smetnje kao na slici 3.7. Bez tog praga dobiven je niz prikazanih na slici 3.8. Nakon odsijecanja zbog smetnje dobije se dodatna slika koja ne sadrži znamenku. Zbog toga je uvedena provjera zadovoljava li odsečena znamenka pretpostavljenu minimalnu duljinu, odnosno prag w_{min} .



Slika 3.6: Vertikalno projekcija slike 3.5.



Slika 3.7: Primjer predloška sa smetnjom na lijevoj strani slike.

Najbolji rezultati prilikom segmentacije predložaka ispitnog skupa dobivaju se s pragovima postavljenim na sljedeće vrijednosti:

- $t_h = 80$ slikovnih elemenata,
- $t_w = 15$ slikovnih elemenata,
- $t_{iw} = 5$ slikovnih elemenata,
- $h_{min} = 40$ slikovnih elemenata,
- $w_{min} = 40$ slikovnih elemenata.

Navedene vrijednosti dobivene su eksperimentalno za dani ispitni skup podataka te nisu otporne na promjenu rezolucije. Ukoliko se predlošci skeniraju drugačijom rezolucijom, pragovi će se morati postaviti na drugačije vrijednosti.

Druga metoda koristi isti postupak, razlika je u pragovima i vrijednostima projekcija. Vrijednosti pragova t_h , t_w , t_{iw} i projekcija su skalirane na interval $[0, 1]$. Projekcije se jednostavno skaliraju tako da se suma crnih slikovnih elemenata u retcima



Slika 3.8: Vertikalno odsijecanje slike 3.7 bez praga w_{min} , pojedine znamenke su odvojene sivom crtom.

podijeli s duljinom slike za horizontalnu projekciju, a za vertikalnu se suma crnih slikovnih elemenata u stupcima podijeli s visinom slike. Najbolji rezultati na danom ispitnom skupu dobiju se za sljedeće vrijednosti pragova:

- $t_h = 0.08$,
- $t_w = 0.1$,
- $t_{iw} = 0.03$,
- $h_{min} = 40$,
- $w_{min} = 40$.

Za horizontalno odsijecanje koristi se horizontalna projekcija. Postupak horizontalnog odsijecanja je prikazan u nastavku.

1. Postavi $y1 = 0$ te $y2 = 0$, gdje su $y1$ i $y2$ retci koji određuju horizontalne granice odsijecanja.
2. Obidi sve retke od 0 do h , gdje je h visina slike, te za svaki redak:
 - ako je $r(y1) \geq t_h$, gdje je t_h unaprijed određen prag, te ako je $y1 \geq y2$ povećaj $y1$ za 1,
 - inače ako je $y2 > y1$ i $r(y2) \geq t_h$, onda povećaj $y2$ za 1,
 - inače ako je $y2 > y1$ i $r(y2) < t_h$ i $y2 - y1 > h_{min}$, gdje je h_{min} unaprijed postavljena minimalna visina znamenke, onda izađi iz petlje,
 - inače povećaj $y1$ za 1.
3. Odsjeci sliku od visine $y1$ do $y2$.

Nakon što se uklone nebitni dijelovi slike s gornje i donje strane horizontalnim odsijecanjem, slijedi vertikalno odsijecanje kojim se odjeljuju znamenke u zasebne slike. Postupak vertikalnog odsijecanja je prikazan u nastavku.

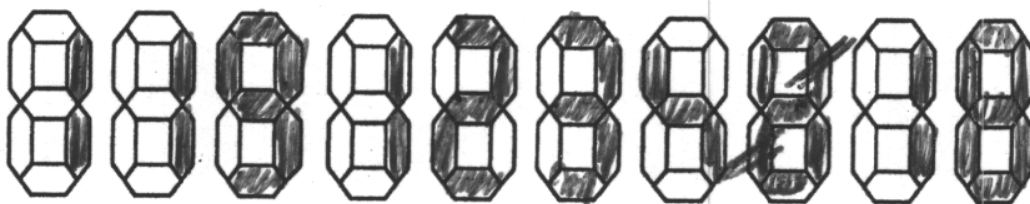
1. Postavi $x1 = 0$ te $x2 = 0$, gdje su $x1$ i $x2$ stupci koji određuju vertikalne granice odsijecanja.
2. Obidi sve stupce od 0 do w , gdje je w duljina slike, te za svaki stupac:
 - ako je $c(x1) \geq t_w$, gdje je t_w unaprijed određen prag, te ako je $x1 \geq x2$ povećaj $x1$ za 1,
 - inače ako je $x2 > x1$ i $c(x2) > t_{iw}$, onda povećaj $y2$ za 1,

- inače ako je $x_2 > x_1$ i $c(x_2) < t_w$ i $x_2 - x_1 > w_{min}$, gdje je w_{min} unaprijed postavljena minimalna duljina znamenke, odsjeci sliku od stupca x_1 do x_2 te postavi $x_1 = x_2 + 1$,
- inače povećaj x_1 za 1.

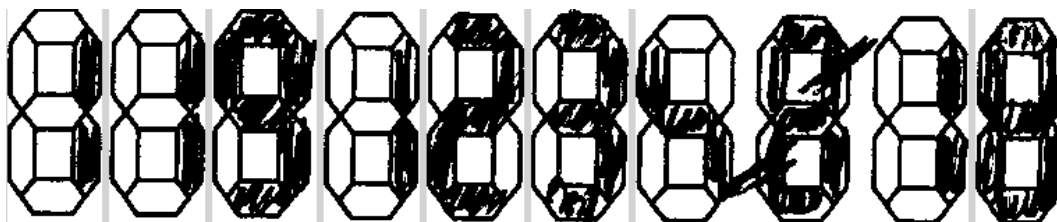
Nakon što se obavi i horizontalno odsijecanje dobije se niz slika koje sadrže po jednu znamenku.

3.1.3. Nedostatci metode

Opisana metoda ima nedostatak. Zbog ovisnosti o unaprijed određenim pragovima metoda neće razdvajati predloške koji imaju drugačiju dimenziju od predložaka iz ispitnog skupa, što je očigledno pošto pragovi w_{min} i h_{min} određuju minimalnu duljinu odnosno visinu znamenke, tako da se predlošci manje visine ili duljine neće segmentirati. Drugi nedostatak je činjenica da se ovim postupkom ne mogu segmentirati



Slika 3.9: Primjer predloška sa smetnjom zbog greške prilikom zacrnjivanja.



Slika 3.10: Vertikalno odsijecanje slike 3.9.

predlošci koji sadrže smetnje poput one prikazane na slici 3.9. Naime, zbog praga t_{iw} , dvije znamenke će ostati spojene nakon segmentacije. Rezultat segmentacije predloška na slici 3.9 prikazan je na slici 3.10. Očito je da će zbog greške koja je nastala prilikom zacrnjivanja tri znamenke ostati spojene u jednoj slici.

3.2. Segmentacija pomoću traženja prvog crnog slikovnog elementa sa svake strane

Za ovu metodu potrebno je prethodno odrediti prva pojavljivanja crnog slikovnog elementa sa svake strane.

3.2.1. Određivanje prvog crnog slikovnog elementa

Za treću metodu segmentacije potrebno je odrediti, za svaki stupac i redak, prvo pojavljivanje crnog slikovnog elementa što znači da je potrebno odrediti prvo pojavljivanje crnog slikovnog elementa s lijeve ($l(y)$), desne ($d(y)$), gornje ($v(x)$) i donje ($b(x)$) strane slike. Za traženje prvog crnog slikovnog elementa od dna, jednostavno se pusti vertikalna crta od dna slike prema vrhu te se traži prvo sjecište s crnim slikovnim elementom, i tako po cijeloj duljini slike. Odnosno, potrebno je po cijeloj duljini slike početi od dna i za svaki stupac naći prvo pojavljivanje crnog slikovnog elementa. Za traženje prvog crnog slikovnog elementa od vrha, izvodi se jednak postupak, samo što se počinje od vrha za svaki stupac i kreće prema dnu. Kod traženja prvih crnih slikovnih elemenata s lijeve i desne strane, postupno se prolazi po visini slike i traži se prvo pojavljivanje slikovnog elementa u retku s lijeve strane, odnosno desne. Sljedeći isječak koda u programskom jeziku *Java* prikazuje postupak traženja prvog slikovnog elementa od dna.

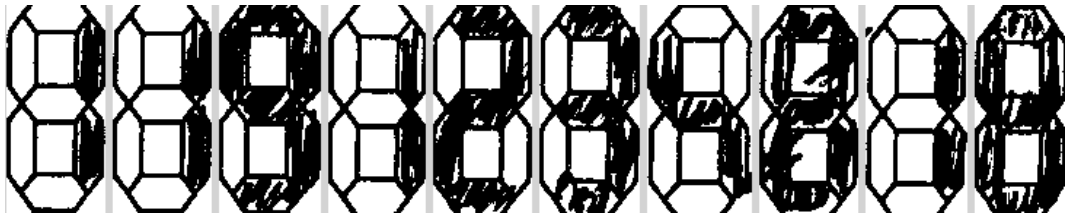
```
private double[] findBot(BufferedImage image) {  
    \\ Stvori niz koji sadrzi pozicije prvih crnih slikovnih elemenata  
    double[] projection = new double[image.getWidth()];  
    \\ Prodi po cijeloj duljini slike  
    for (int i = 0; i < image.getWidth(); i++) {  
        \\ Prodi po cijeloj visini slike od dna do vrha  
        for (int j = image.getHeight()-1; j >= 0; j--) {  
            \\ Ako je slikovni element (i,j) crn, ili  
            \\ ako je j na kraju stupca postavi trenutnu  
            \\ poziciju od dna za ovaj stupac i izadi iz petlje  
            if (j == 0 || (image.getRGB(i, j) & 0xFF) == 0) {  
                projection[i] = image.getHeight() - j;  
                break;  
            }  
        }  
    }  
}
```

```

    }
    return projection;
}

```

Prednost ove metode naspram prethodno navedenih je što može segmentirati predloške koji sadrže smetnju kao na slici 3.9, rezultat segmentacije navedene slike ovom metodom je prikazan na slici 3.11.



Slika 3.11: Vertikalno odsijecanje slike 3.9.

3.2.2. Postupak segmentacije pomoću traženja prvog crnog slikovnog elementa sa svake strane

Slično kao za prethodnu metodu, i ova metoda koristi unaprijed određene pragove. Za ovu metodu koriste se sljedeći pragovi:

- t_{dno} – prag pojavljivanja prvog crnog slikovnog elementa od dna,
- t_{vrh} – prag pojavljivanja prvog crnog slikovnog elementa od vrha,
- t_{lijevo} – prag pojavljivanja prvog crnog slikovnog elementa s lijeva,
- t_{desno} – prag pojavljivanja prvog crnog slikovnog elementa s desna.

Ova metoda se također sastoji od dva dijela: horizontalnog i vertikalnog odsijecanja. Prvo se izvodi horizontalno odsijecanje kako je opisano u nastavku.

1. Postavi $y1 = 0$ te $y2 = 0$, $y1$ i $y2$ su retci koji određuju horizontalne granice odsijecanja.
2. Obidi sve retke od 0 do h , gdje je h visina slike, te za svaki redak:
 - ako je $l(y1) < t_{lijevo}$ te ako je $d(y1) < t_{desno}$, gdje su $l(y1)$ te $d(y1)$ pozicije prvog crnog slikovnog elementa s lijeve i desne strane za dani redak $y1$, onda povećaj $y1$ za 1,
 - inače ako je $y2 > y1$ i $l(y2) < t_{lijevo}$ te $d(y2) < t_{desno}$, onda povećaj $y2$ za 1,

- inače ako je $y2 > y1$ te $l(y2) \geq t_{lijevo}$ ili $d(y2) \geq t_{desno}$, onda izađi iz petlje,
- inače povećaj $y1$ za 1,

3. odsjeci sliku od visine $y1$ do $y2$.

Nakon horizontalnog odsijecanja slijedi vertikalno odsijecanje, kako je opisano u nastavku.

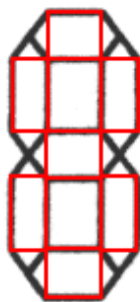
1. Postavi $x1 = 0$ te $x2 = 0$, $x1$ i $x2$ su stupci koji određuju vertikalne granice odsijecanja.
2. Obiđi sve stupce od 0 do w , gdje je w duljina slike, te za svaki stupac:
 - ako je $v(x1) < t_{vrh}$ te ako je $b(x1) < t_{dno}$, gdje su $v(x1)$ te $b(x1)$ pozicije prvog crnog slikovnog elementa s vrha i dna za dani stupac $x1$, onda povećaj $x1$ za 1,
 - inače ako je $x2 > x1$ i $v(x2) < t_{vrh}$ te $b(x2) < t_{dno}$, onda povećaj $x2$ za 1,
 - inače ako je $x2 > x1$ te $v(x2) \geq t_{vrh}$ ili $b(x2) \geq t_{dno}$, odsjeci sliku od stupca $x1$ do $x2$ te postavi $x1 = x2 + 1$,
 - inače povećaj $x1$ za 1.

3.2.3. Nedostatci metode

Nedostatak ove metode je ovisnost o unaprijed postavljenim pragovima t_{dno} , t_{vrh} , t_{lijevo} te t_{desno} . Ukoliko prvo pojavljivanje crnih slikovnih elemenata znamenaka gdje se nalaze znamenke prelaze neki od pragova, segmentacija neće raditi ispravno.

4. Raspoznavanje

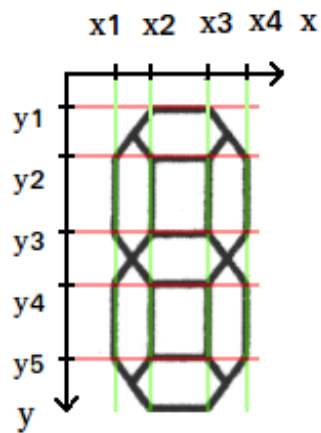
Nakon segmentacije slike, slijedi raspoznavanje znamenaka. Postupkom segmentacije svaka će znamenka biti odvojena u zasebnu sliku. Slika 4.1 je primjer nezacrnjene znamenke. Crvenim pravokutnicima označena su područja interesa, a raspoznavanje znamenaka vrši se analizom tih područja.



Slika 4.1: Primjer predloška za zacrnjivanje s naznačenim područjima interesa.

4.1. Postupak raspoznavanja znamenaka

Postupak raspoznavanja znamenaka u sklopu sustava radi tako da se za svako područje interesa traži suma crnih slikovnih elemenata koja se podijeli s veličinom područja interesa. Tako se dobije postotak popunjenosti. Postotak popunjenosti uspoređuje se s unaprijed zadanim pragom popunjenosti t_p . Ako popunjenost prelazi određeni prag, segment se smatra zacrnjenim. Prije određivanja popunjenosti potrebno je odrediti pozicije područja interesa. Na slici 4.2 su naznačene koordinate gornjeg lijevog kuta područja interesa, a u tablici 4.1 upisane su segmentima pripadajuće koordinate. Pošto se koordinate neće uvijek nalaziti na istom mjestu, potrebno ih je odrediti dinamički. Iz slike 4.1 vidljivo je kako se na dvije trećine visine, te jednoj trećini visine slike nalazi praznina. To znanje moguće je iskoristiti za određivanje $x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4$ te y_5 koordinata. Postupak određivanja koordinata x_1, x_2, x_3 te x_4 je prikazan na sljedećem isječku koda napisanog u programskom jeziku *Java*.



Slika 4.2: Znamenka s naznačenim koordinatama područja interesa.

```

int y = image.getHeight()*2/3;
int x1 = 0;
int x2 = image.getWidth()/2;
int x3 = image.getWidth()/2;
int x4 = image.getWidth()-1;

while(x1 < image.getWidth() &&
    (image.getRGB(x1,y) & 0xFF) != 0){
    x1++;
}

while(x2 > 0 &&
    (image.getRGB(x2,y) & 0xFF) != 0){
    x2--;
}

while(x3 < image.getWidth() &&
    (image.getRGB(x3,y) & 0xFF) != 0){
    x3++;
}

while(x4 > 0 &&
    (image.getRGB(x4,y) & 0xFF) != 0){
    x4--;
}

```

```
}
```

Navedeni programski kod očekuje postojanje praznine na jednoj trećini slike. Traženje koordinate $x1$ započinje tako da se postavi $x1 = 0$ te se uvećava za jedan dok ne dođe do prvog crnog slikovnog elementa ili kraja slike. Nadalje, $x2$ i $x3$ se postave na vrijednost $x2 = x3 = \text{duljina_slike}/2$, jednaki uvjeti vrijede kao i za $x1$, samo što se $x2$ smanjuje za 1, a $x3$ se povećava. Na kraju $x4$ se postavi na vrijednost $x4 = \text{duljina_slike}$ te se umanjuje za 1 dok se ne zadovolji prethodno navedeni uvjet.

Sljedeći isječak koda također pisan u *Javi* prikazuje traženje koordinata $y1$, $y2$, $y3$, $y4$ te $y5$.

```
int x = image.getWidth()/2;
int y1 = 0;
int y2 = image.getHeight()*1/3;
int y3 = image.getHeight()*1/3;
int y4 = image.getHeight()*2/3;
int y5 = image.getHeight()*2/3;
int y6 = image.getHeight()-1;

while(y1 < image.getHeight() &&
      (image.getRGB(x,y1) & 0xFF) != 0){
    y1++;
}

while(y2 > 0 &&
      (image.getRGB(x,y2) & 0xFF) != 0){
    y2--;
}

while(y3 < image.getHeight()/2 &&
      (image.getRGB(x,y3) & 0xFF) != 0){
    y3++;
}

while(y4 > image.getHeight()/2 &&
      (image.getRGB(x,y4) & 0xFF) != 0){
    y4--;
```

```
}
```

```
while(y5 < image.getHeight() &&  
      (image.getRGB(x,y5) & 0xFF) != 0){  
y5++;
```

```
while(y6 > 0&&  
      (image.getRGB(x,y6) & 0xFF) != 0){  
y6--;  
}
```

Za određivanje y koordinata također se očekuje postojanje praznine na jednoj trećini te dvije trećine slike. Postupak je sličan određivanju x koordinata. $y1$ se postavi na vrijednost $y1 = 0$ te se uvećava za jedan dok ne naiđe na crni slikovni element ili ne dođe do kraja slike. $y2$ i $y3$ se postavljaju na vrijednost $y2 = y3 = visina_slike \cdot 1/3$. Jednak uvjet vrijedi kao i za $y1$. $y2$ se postepeno smanjuje, dok se $y3$ povećava. $y4$ i $y5$ se postavljaju na vrijednost $y4 = y5 = visina_slike \cdot 2/3$. $y4$ se povećava za jedan, a $y5$ se smanjuje za jedan sve dok se ne zadovolji prethodno naveden uvjet. Osim tih koordinata potrebno je još i odrediti širine i visine područja interesa, što se može odrediti iz dobivenih koordinata. Izraz 4.2 opisuje način dobivanja širine segmenata a, d, g , dok izrazi 4.3 te 4.4 opisuju dobivanje širina segmenata b, c te e, f . Pomoću izraza 4.5, 4.6, 4.7, 4.8, 4.9 dobiju se visine segmenata a, b, f, c, e, g te d .

$$w_a = w_d = w_g = x3 - x2 \quad (4.1)$$

$$w_b = w_c = x2 - x1 \quad (4.2)$$

$$w_e = w_f = x4 - x3 \quad (4.3)$$

$$h_a = y2 - y1 \quad (4.4)$$

$$h_b = h_f = y3 - y2 \quad (4.5)$$

$$h_c = h_e = y5 - y4 \quad (4.6)$$

$$h_g = y4 - y3 \quad (4.7)$$

$$h_d = y6 - y5 \quad (4.8)$$

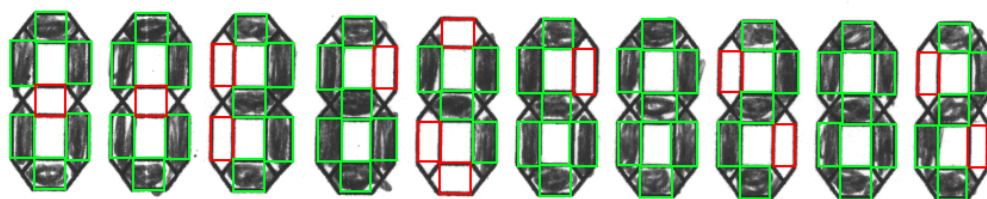
$$(4.9)$$

Najbolji rezultati prepoznavanja dobiveni su postavljanjem praga popunjenosti na vrijednost $t_p = 0.29$. Na slici 4.3 prikazan je niz znamenaka, na kojima su označena

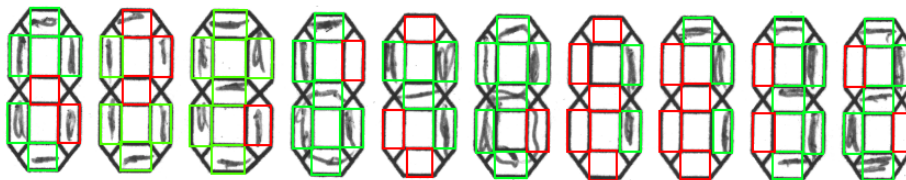
Tablica 4.1: Segmenti s odgovarajućim koordinatama područja interesa.

Segment	Koordinate
a	(x2, y1)
b	(x3, y2)
c	(x3, y4)
d	(x2, y5)
e	(x1, y4)
f	(x1, y2)
g	(x2, y3)

područja interesa. Zelenom bojom su naznačeni segmenti koji su prepoznati kao ispunjeni, dok su crvenom naznačeni neispunjeni segmenti. Za razliku od slike 4.3 koja je dobro naznačena, na slici 4.4 segmenti nisu dobro ispunjeni. Iako je čovjeku sasvim jasno koji brojevi su naznačeni na predlošku, ova metoda s unaprijed određenim pragom ispunjenosti neće prepoznati dio segmenata kao ispunjene.



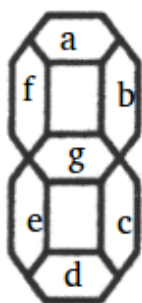
Slika 4.3: Dobro zacrnjeni predložak s označenim područjima interesa.



Slika 4.4: Loše zacrnjeni predložak s označenim područjima interesa.

4.2. Kodiranje znamenaka

Potrebno je predvidjeti sve načina označavanja sedam segmentnih znamenaka. Na slici 4.5 označeni su svi segmenti slovima a, b, c, d, e, f i g . U tablici 4.2 za svaku znamenku napisana je n-torka koja se sastoji od prethodno navedenih slova, svaka n-torka tvori skup označenih segmenata koji čine odgovarajuću znamenku.

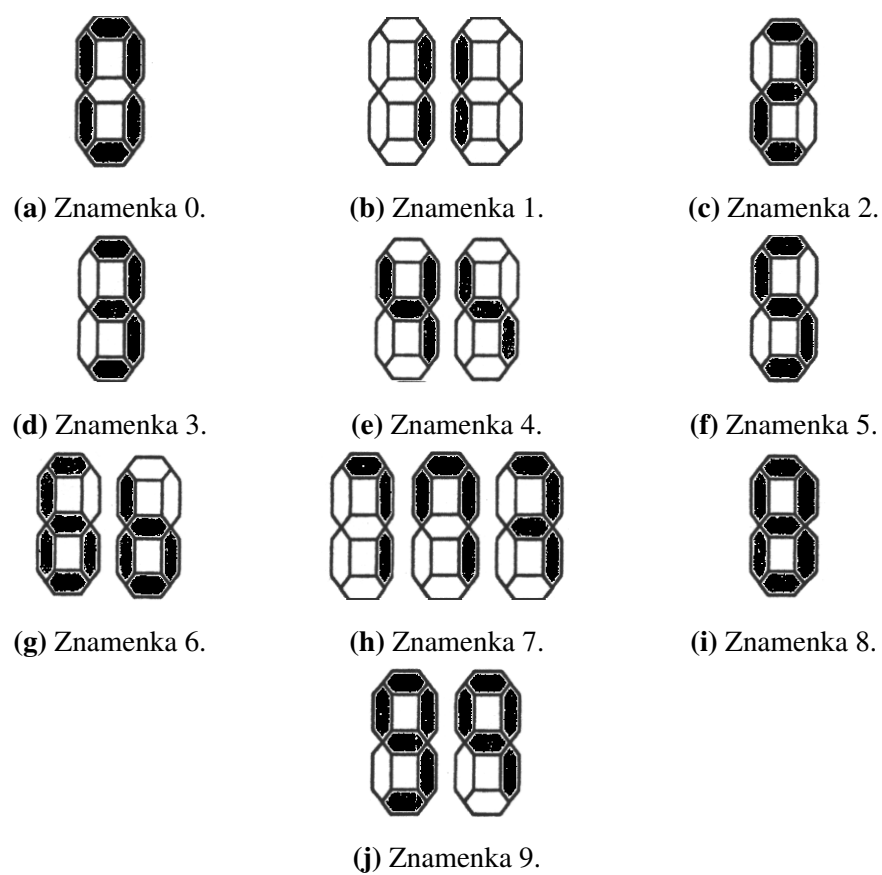


Slika 4.5: Sedam segmentna znamenka s označenim segmentima.

Tablica 4.2: Kodiranje znamenaka.

Znamenka	Zacrtnjeni segmenti
0	(a, b, c, d, e, f)
1	$(b, c), (e, f)$
2	(a, b, d, e, g)
3	(a, b, c, d, g)
4	$(b, c, f, g), (c, f, g)$
5	(a, c, d, f, g)
6	$(a, c, d, e, f, g), (c, d, e, f, g)$
7	$(a, b, c), (a, b, c, f), (a, b, c, g)$
8	(a, b, c, d, e, f, g)
9	$(a, b, c, d, f, g), (a, b, c, f, g)$

Na slici 4.6 naznačene su sve kombinacije zacrnjenih segmenata koje odgovaraju n-torkama naznačenim u tablici 4.2. Budući da osobe drugačije označavaju pojedine znamenke, na danom ispitnom skupu vidljiva su odudaranja od standardnih označavanja naznačenih u tablici 4.2. Znamenka koja nije označena kako je naznačeno u tablici 4.2 smatra se nevažecom.



Slika 4.6: Primjeri načina zacrnjivanja svih znamenaka.

5. Ispitni skup podataka

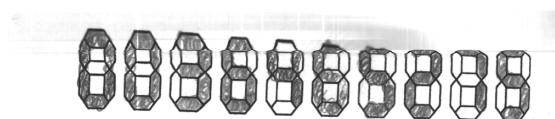
U prošlom poglavlju prikazani su neki od nizova znamenaka iz ispitnog skupa podataka koji su pogrešno zacrnjeni ili je zbog drugih problema otežano očitavanje. Kod analize rezultata svi su predlošci ispitnog skupa pregledani, te su uzeti u obzir. U ispitnom skupu nalaze se 522 predložka, od kojih 56 predložaka nije dobro ispunjeno, odnosno barem jedna znamenka nije dobro zacrnjena, a jedan predložak nije dobro skeniran.

Tablica 5.1: Raspodijela dobrih i loših predložaka.

	Loši predlošci	Dobri predlošci	Ukupno
Broj predložaka	57	465	522
Broj predložaka u postotcima	10.15%	89.85%	

5.0.1. Pogrešno skenirani predlošci

Ukoliko dođe do greške prilikom skeniranja može doći do izobličenja oblika predložaka te će prepoznavanje biti otežano ili nemoguće. Primjer loše skeniranog predložka prikazan je na slici 5.1. Vidljivo je kako je predložak izobličen što će otežati očitavanje. Od svih predložaka u ispitnom skupu podataka, jedan predložak je loše skeniran.

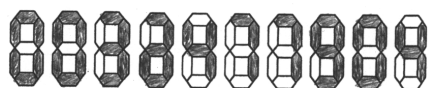


Slika 5.1: Primjer loše skenirane slike.

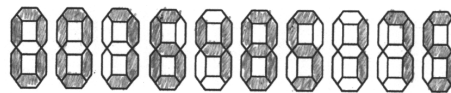
5.0.2. Pogrešno zacrnjeni predlošci

Ispitni skup sadrži veći broj pogrešno zacrnjenih predložaka kao što je prikazano u tablici 5.1. U skup pogrešno zacrnjenih predložaka stavljeni su svi predlošci koji sadrže

krivo zacrnjenu znamenku, odnosno zacrnjeni segmenti ne tvore ni jednu od znamenaka. Slijede dva primjera predložaka koji sadrže znamenke pogrešno zacrnjene znamenke. Na slici 5.2a očekivani niz znamenaka je 0036477904, ali četvrta znamenka je



(a) Očekivani niz 0036477904.



(b) Očekivani niz 0036485175.

Slika 5.2: Primjeri pogrešno zacrnjenih predložaka.

krivo zacrnjena, segmenti popunjeni na taj način ne pripadaju ni jednom skupu navedenom u tablici 4.2. Na slici 5.2b očekivani niz znamenaka je 0036485175, no treća znamenka nije važeća. Tako zacrnjene znamenke sustav ne može prepoznati jer segmenti popunjeni na taj način ne tvore važeće znamenke.

6. Rezultati

Ispitivanje sustava je provedeno na ispitnom skupu podataka od 522 predložaka. Svaki predložak sadrži po 10 znamenaka. U prošlom poglavlju opisan je ispitni skup podataka te je podijeljen u dvije skupine (dobri i loši predlošci). Podatci navedeni u daljnjem tekstu prikazuju rezultate postupka prepoznavanja provedenom na skupu svih predložaka i skupu dobrih predložaka. Nadalje prepoznavanje je provedeno sa različitim metodama segmentacije navedenim u trećem poglavlju te su dobiveni rezultati podijeljeni u tri skupine:

1. prepoznavanje sa segmentacijom pomoću projekcija,
2. prepoznavanje sa segmentacijom pomoću skaliranih projekcija,
3. prepoznavanje sa segmentacijom pomoću traženja prvog crnog slikovnog elementa.

6.1. Rezultati segmentacije

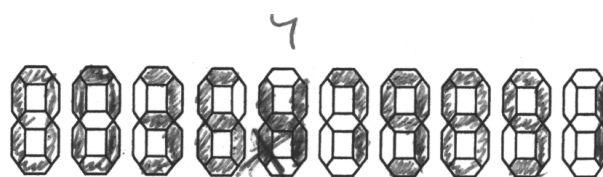
Tablica 6.1 prikazuje uspješnost segmentacije na cijelom ispitnom skupu. Vidljivo je kako su segmentacije pomoću projekcija uspješnije od segmentacije pomoću traženja prvog crnog slikovnog elementa. Kao kriterij određivanja uspješnosti segmentacije uzet je broj odsiječenih znamenaka. Budući da se za svaki predložak očekuje 10 znamenaka, ukoliko nakon odsijecanja broj znamenaka nije 10 segmentacija se smatra neuspješnom.

Slike 6.1a i 6.1b prikazuju jedan od loše segmentiranih predložaka prvom i drugom metodom.

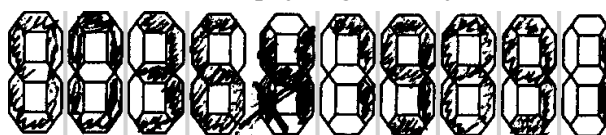
Tablica 6.2 prikazuje uspješnost segmentacije na skupu dobrih predložaka.

Tablica 6.1: Uspješnost segmentacije cijelog ispitnog skupa.

	uspješno	neuspješno	ukupno	metoda
Broj predložaka	518	4	522	Prva metoda segmentacije
Broj predložaka u postotcima	99.23%	0.77%		
Broj predložaka	518	4	522	Druga metoda segmentacije
Broj predložaka u postotcima	99.23%	0.77%		
Broj predložaka	507	15	522	Treća metoda segmentacije
Broj predložaka u postotcima	97.12%	2.88%		



(a) Predložak prije segmentacije.



(b) Predložak sa slike 6.1a nakon segmentacije.

Slika 6.1: Loše segmentiran predložak prije i nakon segmentacije.

Tablica 6.2: Uspješnost segmentacije dobrih predložaka ispitnog skupa.

	uspješno	neuspješno	ukupno	metoda
Broj predložaka	465	0	465	Prva metoda segmentacije
Broj predložaka u postotcima	100%	0%		
Broj predložaka	465	0	465	Druga metoda segmentacije
Broj predložaka u postotcima	100%	0%		
Broj predložaka	453	12	465	Treća metoda segmentacije
Broj predložaka u postotcima	97.41%	2.59%		

6.2. Rezultati prepoznavnja

Tablice 6.3 i 6.4 prikazuju rezultate dobivene prepoznavanjem predložaka na razini niza znamenaka, kao kriterij određivanja uspješnosti prepoznavanja uzeta je točnost cijelog niza znamenaka. Ukoliko jedna znamenka niza nije jednaka očekivanoj, prepoznavanje se smatra neuspješnim.

Tablica 6.3: Uspješnost prepoznavanja predložaka cijelog ispitnog skupa na razini niza znamenaka.

	uspješno	neuspješno	ukupno	metoda
Broj predložaka	446	76	522	Prva metoda segmentacije
Broj predložaka u postotcima	85.44%	14.56%		
Broj predložaka	447	75	522	Druga metoda segmentacije
Broj predložaka u postotcima	85.63%	14.37%		
Broj predložaka	305	160	465	Treća metoda segmentacije
Broj predložaka u postotcima	65.59%	34.41%		

Tablica 6.4: Uspješnost prepoznavanja dobrih predložaka ispitnog skupa na razini niza znamenaka

	uspješno	neuspješno	ukupno	metoda
Broj predložaka	409	56	465	Prva metoda segmentacije
Broj predložaka u postotcima	87.96%	12.04%		
Broj predložaka	410	55	465	Druga metoda segmentacije
Broj predložaka u postotcima	88.17%	11.83%		
Broj predložaka	336	186	522	Treća metoda segmentacije
Broj predložaka u postotcima	64.37%	35.63%		

Tablice 6.5 te 6.6 prikazuju rezultate dobivene prepoznavanjem predložaka na razini znamenke. Ukoliko prepoznata znamenka ne odgovara očekivanoj, prepoznavanje na razini znamenke smatra se neuspješnim.

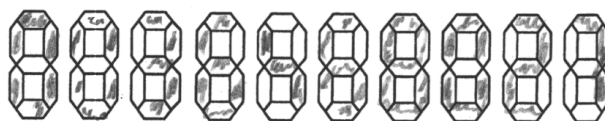
Tablica 6.5: Uspješnost prepoznavanja predložaka cijelog ispitnog skupa na razini znamenke.

	uspješno	neuspješno	ukupno	metoda
Broj znamenaka	5060	120	5180	Prva metoda segmentacije
Broj znamenaka u postotcima	97.68%	2.32%		
Broj znamenaka	5061	119	5180	Druga metoda segmentacije
Broj znamenaka u postotcima	97.70%	2.30%		
Broj znamenaka	4747	323	5070	Treća metoda segmentacije
Broj znamenaka u postotcima	93.63%	6.37%		

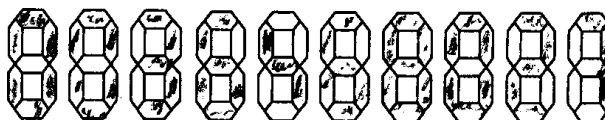
Slike 6.2 prikazuju predloške koji nisu dobro prepoznati drugom metodom prije i nakon binarizacije. Sljedeće tablice prikazuju matrice zabune dobivene segmentacijom

Tablica 6.6: Uspješnost prepoznavanja dobrih predložaka ispitnog skupa na razini znamenke.

	uspješno	neuspješno	ukupno	metoda
Broj znamenaka	4559	91	4650	Prva metoda segmentacije
Broj znamenaka u postotcima	98.04%	1.96%		
Broj znamenaka	4560	90	4650	Druga metoda segmentacije
Broj znamenaka u postotcima	98.06%	1.94%		
Broj znamenaka	4251	279	4530	Treća metoda segmentacije
Broj znamenaka u postotcima	93.84%	6.16%		

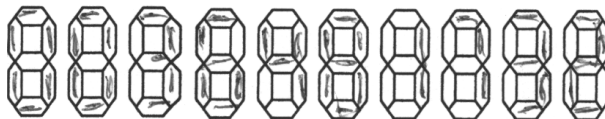


(a) Predložak prije binarizacije.

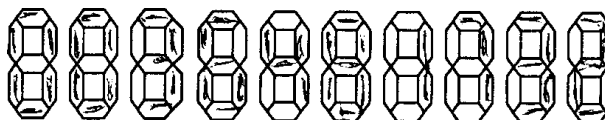


(b) Predložak sa slike 6.2a nakon binarizacije.

Dobiven je niz 003?468027, gdje znak "?" označava nevažeću znamenku.



(c) Predložak prije binarizacije.



(d) Predložak sa slike 6.2c nakon binarizacije.

Dobiven je niz ??3?46173?, gdje znak "?" označava nevažeću znamenku.

Slika 6.2: Predlošci koji nisu dobro prepoznati prije i nakon binarizacije.

pomoću projekcija. Matrice 6.7 i 6.8 prikazuju matrice zabune na svim predlošcima ispitnog skupa i na dobrom ispitnom skupu.

Sljedeće tablice prikazuju matrice zabune dobivene segmentacijom pomoću skaliranih projekcija. Matrice 6.9 i 6.10 prikazuju matrice zabune na svim predlošcima ispitnog skupa i na dobrom ispitnom skupu.

Sljedeće tablice prikazuju matrice zabune dobivene segmentacijom pomoću traže-

Tablica 6.7: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	1087	0	0	0	0	0	0	1	13	0	11
1	0	307	0	0	0	0	0	3	0	0	2
2	0	0	309	0	0	0	0	0	0	0	10
3	1	0	0	685	0	0	0	0	1	8	10
4	0	0	0	0	634	0	1	0	1	4	6
5	0	0	0	0	0	241	1	0	1	0	3
6	0	0	0	1	0	1	685	0	4	0	10
7	0	0	0	0	0	0	0	273	0	1	2
8	1	0	0	1	1	0	2	0	463	0	8
9	0	0	0	0	0	0	0	0	2	267	2
?	0	0	0	0	0	0	1	0	3	2	110

Tablica 6.8: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	989	0	0	0	0	0	0	1	7	0	10
1	0	291	0	0	0	0	0	3	0	0	2
2	0	0	297	0	0	0	0	0	0	0	9
3	0	0	0	629	0	0	0	0	1	7	9
4	0	0	0	0	595	0	0	0	1	4	5
5	0	0	0	0	0	211	1	0	0	0	3
6	0	0	0	0	0	0	628	0	4	0	8
7	0	0	0	0	0	0	0	248	0	0	2
8	1	0	0	1	0	0	2	0	432	0	6
9	0	0	0	0	0	0	0	0	2	240	1
10	0	0	0	0	0	0	0	0	0	0	0

nja prvog crnog slikovnog elementa. Matrice 6.11 i 6.12 prikazuju matrice zabune na svim predlošcima ispitnog skupa i na dobrom ispitnom skupu.

Tablica 6.9: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	1087	0	0	0	0	0	0	1	13	0	11
1	0	307	0	0	0	0	0	3	0	0	2
2	0	0	309	0	0	0	0	0	0	0	10
3	1	0	0	685	0	0	0	0	1	8	10
4	0	0	0	0	634	0	1	0	1	4	6
5	0	0	0	0	0	241	1	0	1	0	3
6	0	0	0	1	0	1	685	0	4	0	10
7	0	0	0	0	0	0	0	273	0	1	2
8	1	0	0	1	1	0	2	0	462	0	9
9	0	0	0	0	0	0	0	0	2	267	2
?	0	0	0	0	0	0	1	0	3	2	110

Tablica 6.10: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	989	0	0	0	0	0	0	1	7	0	10
1	0	291	0	0	0	0	0	3	0	0	2
2	0	0	297	0	0	0	0	0	0	0	9
3	0	0	0	629	0	0	0	0	1	7	9
4	0	0	0	0	595	0	0	0	1	4	5
5	0	0	0	0	0	211	1	0	0	0	3
6	0	0	0	0	0	0	628	0	4	0	8
7	0	0	0	0	0	0	0	248	0	0	2
8	1	0	0	1	0	0	2	0	431	0	7
9	0	0	0	0	0	0	0	0	2	240	1
?	0	0	0	0	0	0	0	0	0	0	0

6.3. Analiza rezultata

Iz tablica 6.1 i 6.2 vidljivo je kako se najbolja segmentacija postiže prvom i drugom metodom. Što se tiče raspoznavanja, najbolji rezultati postignuti su uz segmentaciju pomoću skaliranih projekcija, što je vidljivo iz 6.4 i 6.3. Iz matrica zabune 6.7, 6.8, 6.9

Tablica 6.11: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	1051	0	0	0	0	0	0	1	13	0	24
1	0	204	0	0	0	0	0	101	0	0	6
2	1	0	297	0	0	1	0	0	0	0	14
3	0	0	0	661	0	0	0	3	1	8	12
4	0	0	0	0	563	0	0	0	3	47	17
5	0	0	0	0	0	237	0	0	3	0	6
6	0	0	1	0	0	0	657	0	3	0	24
7	0	0	0	0	0	0	0	266	0	1	3
8	1	0	1	0	1	2	3	0	447	1	11
9	0	0	0	0	0	0	0	0	1	256	4
?	0	0	0	0	0	0	1	0	2	2	108

Tablica 6.12: Matrica zabune. Redak predstavlja očekivani razred slike, a stupac razred slike dobiven klasifikacijom.

	0	1	2	3	4	5	6	7	8	9	?
0	958	0	0	0	0	0	0	1	7	0	19
1	0	187	0	0	0	0	0	96	0	0	6
2	0	0	285	0	0	0	0	0	0	0	13
3	0	0	0	604	0	0	0	3	1	7	9
4	0	0	0	0	528	0	0	0	2	45	14
5	0	0	0	0	0	206	0	0	0	0	6
6	0	0	0	0	0	0	601	0	2	0	23
7	0	0	0	0	0	0	0	241	0	0	3
8	1	0	0	0	0	2	3	0	414	1	11
9	0	0	0	0	0	0	0	0	1	227	3
?	0	0	0	0	0	0	0	0	0	0	0

i 6.10 vidljivo je kako je najčešća kriva klasifikacija znamenke 0 u znamenku 8 te znamenke 3 u znamenku 9. Nadalje iz matrica zabune 6.11 i 6.12 je vidljivo kako segmentacija pomoću traženja prvog crnog slikovnog elementa loše utječe na raspoznavanje. Najčešće greške klasifikacije su klasifikacija znamenke 1 u znamenke 7 te znamenke 4 u znamenku 9. Što je rezultat nedostatka postupka raspoznavanja znamenki. Dobiveni

rezultati pokazuju da se najviše znamenaka točno prepoznaje metodom segmentacije pomoću skaliranih projekcija.

7. Zaključak

Predstavljen je sustav koji bi omogućio lakšu pripremu ispita. U sklopu sustava postignuta je točnost od 85.63% na cijelom ispitnom skupu od 522 predložaka, dok je na skupu dobrih predložaka postignuta točnost od 88.17% na razini niza znamenaka. Što se tiče točnosti na razini samih znamenaka postignuta je točnost od 97.7% na cijelom ispitnom skupu, a na skupu dobrih predložaka točnost je 98.06%. Od svih metoda koje su se koristile za segmentaciju u ovom radu, najboljom se pokazala metoda koja koristi skalirane projekcije.

Očigledno je kako sustav trenutno ne može imati stvarnu upotrebu zbog dva razloga. Prvi razlog je činjenica da trenutno sustav nema dovoljno dobru točnost, a drugi razlog je činjenica da je veći broj predložaka iz ispitnog skupa podataka krivo ispunjen što dovodi u pitanje je li ovakva vrsta predložaka dobro rješenje problema. Za daljnje unaprjeđenje očitavanja smatram da bi za segmentaciju i raspoznavanje trebalo koristiti metode koje ne ovise o unaprijed postavljenim pragovima.

LITERATURA

- [1] I. Bonačić, T. Herman, T. Krznar, E. Mangić, G. Molnar, i M. Čupić. Optical character recognition of seven-segment display digits using neural networks. 2009.
- [2] M. Cheriet, N. Khanna, C. L. Liu, i C. Y. Suen. *Character Recognition Systems*. John Wiley & Sons, 2007.
- [3] Rakhi P Ghugardare, Sandip P Narote, P Mukherji, i Prathamesh M Kulkarni. Optical character recognition system for seven segment display images of measuring instruments. U *TENCON 2009-2009 IEEE Region 10 Conference*, stranice 1–6. IEEE, 2009.
- [4] A. Greensted. Otsu thresholding. <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html> (10.4.2016), 2010.
- [5] T. Helland. Seven grayscale conversion algorithms. <http://www.tannerhelland.com/3643/grayscale-image-algorithm-vb6> (10.4.2016), 2011.
- [6] M. Kokot. Izvedba biblioteke za strojno očitavanje rukom popunjavanj sedamsegmentnih znamenki. Diplomski rad, Fakultet elektrotehnike i računarstva, Unska 3, 2014.
- [7] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, Siječanj 1979. ISSN 0018-9472. doi: 10.1109/TSMC.1979.4310076.

Očitavanje studentskih matičnih brojeva unesenih zacrnjivanjem predloška znamenki

Sažetak

U sklopu rada realiziran je sustav za očitavanje studentskih brojeva unesenih zacrnjivanjem predloška koji ima oblik sedam segmentnog prikaznika. Sustav se sastoji od tri dijela. Prvi dio je predobrada slike, što uključuje pretvaranje slike u sivu sliku te postupak binarizacije. Drugi dio se sastoji od segmentacije odnosno izvlačenja slika pojedinih znamenaka iz cijelog predloška. Treći dio se sastoji od raspoznavanja pojedinih znamenaka. Sustav je ostvaren u programskom jeziku *Java*.

Ključne riječi: binarizacija, segmentacija, prepoznavanje predložaka, sedam segmentni predložak.

Automatic Recognition of Student Identifiers from Digit Templates

Abstract

A system for character recognition of student identifiers was constructed as a part of this theses. System consists of three parts. First part is pre-processing of an image, which includes conversion of an image into grey-scale and binarisation. Second part consists of segmentation or extraction of individual digits from template. Third part is the recognition of individual digits from image. The system was implemented in *Java* programming language.

Keywords: binarisation, segmentation, template recognition, seven segment digit template.