

UNIVERZA V LJUBLJANI  
FAKULTETA ZA ELEKTROTEHNIKO

# **Sistem za avtomatizirano optično odčitavanje električnih števecv**

**Magistrska naloga**

**Boštjan MUROVEC**



# Zahvala

Za strokovno pomoč in prijateljsko vzpodbudo pri nastajanju magistrske naloge se iskreno zahvaljujem svojemu mentorju prof. dr. Stanislavu Kovačiču, dipl. ing., ki me je kot podiplomskega študenta skrbno spremljal, usmerjal in mi ustvaril odlične delovne pogoje, v katerih sem z veseljem raziskoval in se strokovno izpopolnjeval.

Zahvaljujem se tudi kolegom in prijateljem Franciju Lahajnarju, dipl. ing., Marku Knezu, dipl. ing. ter Petru Roglju, dipl. ing. za prijetno in stimulatívno delovno vzdušje v laboratoriju ter za številne izmenjave strokovnih izkušenj.

Posebno mesto v zahvali posvečam prijatelju mag. Antonu Poznetu ml., dipl. ing. za strokovne nasvete in neprecenljivo pomoč pri strojni opremi, ki sem jo potreboval za raziskave.

Na zadnjem, a v nobenem pogledu manj pomembnem, mestu bi se rad iskreno zahvalil svojim staršem, ki so mi v času študija vsestransko pomagali, zaupali vame ter me vzpodbujali.

Boštjan Murovec





## Povzetek

Pričujoče delo podaja rezultate raziskav in razvoja avtomatiziranega odčitavanja električnih števecv s pomočjo kamere in osebnega računalnika. Potreba po sistemu, ki bi opravljal to nalogo, se je pojavila v tovarni, kjer električne števecv izdelujejo. Preizkus teh naprav je skoraj v celoti avtomatiziran, glavno oviro do popolne avtomatizacije pa predstavlja končni preizkus proizvodov, sestavni del katerega je odčitavanje njihovih vrednosti, ki ga v odsotnosti ustreznega sistema za odčitavanje opravlja človek. Tak način kontrole je že v osnovi manj zanesljiv od avtomatiziranega, saj je rutinsko in ponavljajoče se odčitavanje vrednosti števecv utrudljivo in zahteva od človeka veliko koncentracije. Iz tega sledi, da se je praktično nemogoče izogniti določenemu številu napak pri kontroli proizvodov, ki so težko ugotovljive in niso naknadno dokazljive. Postopek avtomatskega odčitavanja električnih števecv, kot ga v tem delu predlagamo, izpolnjuje pričakovanja industrijskega okolja po hitrem, robustnem, ponovljivem in točnem delovanju. S tem se odpira možnost popolne avtomatizacije proizvodnje teh naprav. Laboratorijski testi kažejo, da bodo končni produkt lahko uporabljali tudi serviserji električnih števecv z uporabo ročnih odčitovalnikov, kjer so delovni pogoji manj predvidljivi kot v sami proizvodnji.

Delo podrobno opisuje strukturo sistema ter postopke, ki nas od zajete slike pripeljejo do končnega rezultata. Poleg prijemov, ki smo jih pri odčitavanju uporabili, so opisane še alternativne rešitve, ki smo jih v toku raziskav preizkusili in zavrnil, ker so se izkazale za manj uspešne. Kljub temu nam pomagajo bolje razumeti delovanje sistema ter nam omogočajo popolnejši pregled nad obravnavano problematiko.



# 1 Uvod

Potreba po optičnem odčitavanju električnih števecov izvira iz potrebe po popolni avtomatizaciji proizvodnje teh naprav. Preden električni števec zapusti tovarno, ga je potrebno preizkusiti; vrednosti števecov se odčitajo pred samim pričetkom testiranja, nato se jih izpostavi električni obremenitvi, nakar ponovno sledi odčitavanje ter ugotavljanje, ali je izmerjena vrednost enaka pričakovani. Tovarna izdelava več tisoč števecov dnevno, kar pomeni, da je potrebno število odčitavanj zelo veliko. Ker do sedaj sistem avtomatskega zajema vrednosti ne obstaja, morajo to delo opravljati ljudje, kar ima za posledico precej slabosti, ki so opisane v nadaljevanju. Avtomatski zajem nam odpira možnost odpraviti te slabosti in s tem dvigniti kvaliteto proizvodnje.

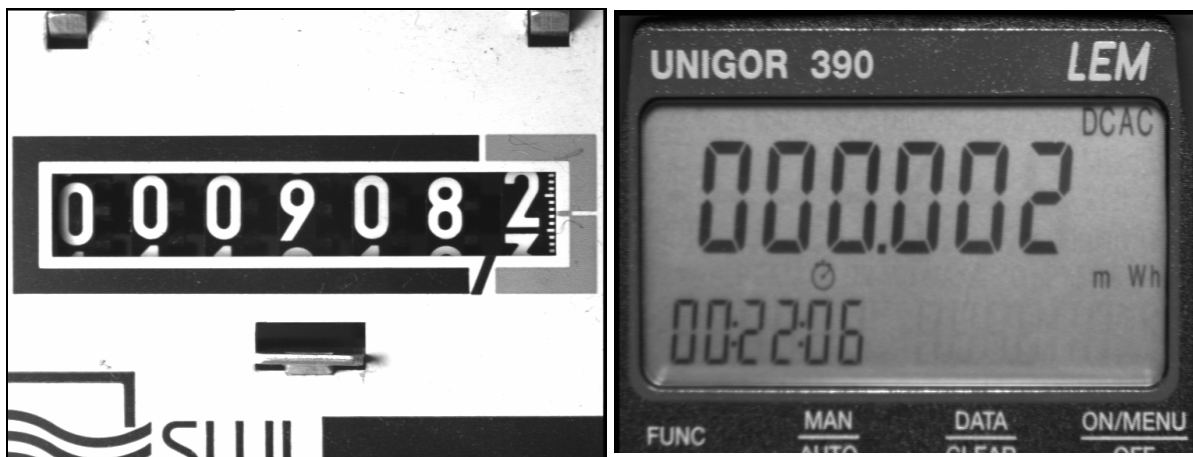
V uvodnem poglavju bomo podali razliko med električnimi števci z analognim in digitalnim prikazom (poglavje 1.1) ter opisali čelno ploščo analognega števca, ki nas v tem primeru zanima (poglavje 1.2). Predstavili bomo tudi potek preizkusa električnih števecov brez avtomatskega zajema njihovih vrednosti (poglavje 1.3), slabosti, ki iz tega izhajajo (poglavje 1.4), ter prednosti, ki jih pričakujemo od avtomatizacije postopka (poglavje 1.5). Sledi groba specifikacija zahtev sistema za odčitavanje (poglavje 1.6).

## 1.1 Električni števci in prikaz vrednosti

Glede na princip merjenja energije delimo električne števce na indukcijske in elektronske. Indukcijski prikazujejo izmerjeno vrednost s pomočjo rotacijskega števnikar (slika 1 levo), elektronski pa z LED prikazovalnikom (slika 1 desno) ali redkeje tudi z rotacijskim števnikom.

Rotacijski števnik je sestavljen iz niza kolesčkov (dekad), na katerih so natisnjene številke. Kolesček z najmanjšo vrednostjo (skrajno desni) se vrti s kotno hitrostjo, ki je premo sorazmerna trenutni moči, ki jo števec meri. Ostali kolesčki so z njim povezani preko mehanskega prenosa in naslednji se premakne za eno številko naprej, ko predhodni naredi celoten obrat. Na ta način je doseženo, da števnik prikazuje porabo električne energije, ki jo matematično izrazimo kot časovni integral trenutne moči.

Elektronski električni števci so zgodovinsko gledano novejši in imajo dolgoročne možnosti zamenjave indukcijskih. Kljub temu se v tem trenutku indukcijski števci proizvajajo v večjem številu.



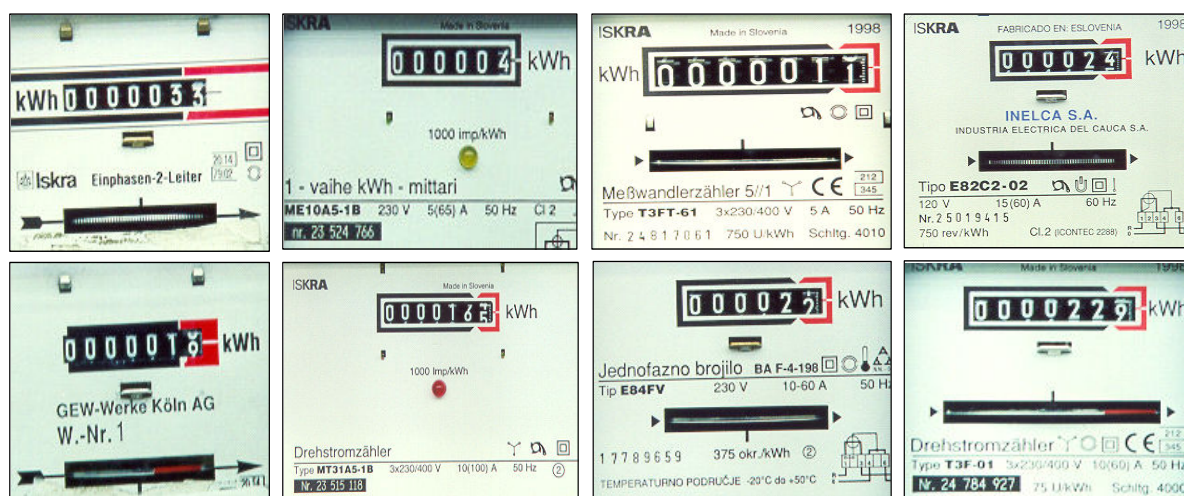
**Slika 1: Električni števec z rotacijskim števnikom (levo) ter prikazovalnikom LCD (desno)**

Magistrska naloga je posvečena avtomatskemu odčitavanju števecov z rotacijskom prikazovalnikom, kar je neprimerno težje izvedljivo kot odčitavanje prikazovalnikov LCD (poglavje 4.1). Nekateri postopki, ki smo jih razvili, so uporabni tudi pri realizaciji odčitavanja slednjih (zlasti določanje položaja števnik, poglavje 1). V nadaljevanju bomo imeli v mislih analogne števec, ko bomo uporabljali izraz električni števec.

## 1.2 Čelna plošča električnega števca

Nekaj primerov čelnih plošč vidimo na sliki 2. Opazimo lahko, da se med seboj precej razlikujejo, kljub temu pa imajo nekatere bistvene lastnosti skupne in človek lahko na vsaki izmed njih z lahkoto prepozna posamezna področja, kot so števnik, kotva in serijska številka.

S stališča optičnega odčitavanja električnih števecov sta zlasti zanimivi področji števnik in serijske številke. Področje kotve za nas ni zanimivo, ker sta njen zajem in kontrola že rešeni (omenjeno v poglavju 1.3). Problematiki odčitavanja serijske številke se v tem delu ne bomo posvečali, ker njen razvoj poteka vzporedno [L1]. Za nas je interesantno področje števnik, katerega vrednost moramo razbrati.



Slika 2: Nekaj primerov čelnih plošč števecov z enim števnikom

Opazimo, da imajo vsi števniki črno ozadje ter bele številke. Vidimo lahko tudi, da so številke predstavljene večinoma z arabskimi številkami, možno pa je, da se nahajajo na zadnjem koleščku številke, zakodirane z izbranimi vzorci (slika 3).

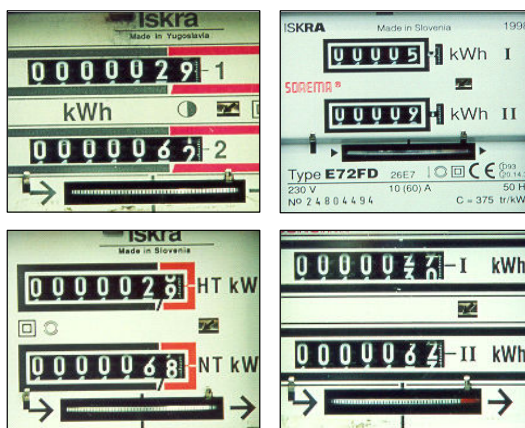


**Slika 3: Števec z arabsko (levo) ter kodirano (desno) decimalnko**

Z vidika problematike optičnega odčitavanja vrednosti se čelne plošče razlikujejo v položaju števnikov na njih, razlike pa so tudi med grafičnimi prikazi števnikov. Tako lahko opazimo, da imajo nekatere izvedbe le-teh črn ali črno-rdeč rob, druge imajo okoli sebe ozek bel pas, šele nato so obrobljene. Nekateri števniki pa nimajo pravokotne obrobe, ampak so od ostalega dela čelne plošče razmejeni z vodoravnimi črtami. Omenimo še ločevanje zadnje številke od ostalih z decimalno zarezo, ki je največkrat prisotna; njeno vlogo lahko opravljata tudi različni barvi ali pa grafične ločitve med zadnjo in ostalimi številkami sploh ni. Vse te razlike je pri odčitavanju potrebno upoštevati, sicer bi sistem deloval pravilno le z majhnim številom izvedb števnikov.

Veliko števecov ima vgrajena dva števnikov (slika 4). Njuna lega na čelni plošči variira od modela do modela. Odčitavanje teh števecov rešujemo z dvojnimi zajemom slike, pri čemer je kamera vsakič postavljena tako, da je na sliki v celoti viden ustrezen števník. Realizirali smo sicer detekcijo položaja ter odčitavanje vrednosti več števnikov, vendar smo ugotovili, da bi morali sliko zajemati pri premajhni povečavi, ki

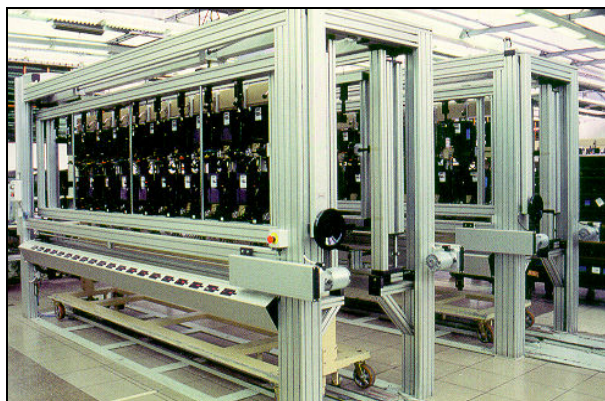
ne bi omogočala odčitavanja z zadovoljivo natančnostjo, če bi hoteli doseči, da sta oba števnika vidna v celoti.



Slika 4: Nekaj primerov čelnih plošč z dvema števnikoma

## 1.3 Končni preizkus električnih števcov

Števci, ki vstopajo v fazo preizkusa (velikostnega reda štirideset izdelkov naenkrat), se nahajajo na posebnih nosilcih in so pritrjeni v vodoravni vrsti v enakomernih presledkih. Tipično obstajata dva niza le-teh, ki se nahajata en nad drugim (slika 5).



Slika 5: Števci na nosilcih med končnim preizkusom

Pred samo izvedbo preizkusa se odčitajo vrednosti vsakega števca posebej. To opravi operater tako, da se s prenosnim terminalom premika od števca do števca, odčitava vrednosti s prostim očesom ter jih preko posebne tipkovnice vnaša v računalnik. Odčitavanju vrednosti sledi fantomska obremenitev, ki ima za posledico, da se števci pričnejo vrteti. V toku te faze se s posebnimi optičnimi glavami kontrolira pravilno delovanje kotve (vsak števec ima svojo optično glavo). Po koncu obremenitve zopet prevzame vlogo operater, ki ponovno odčita ter vnese v računalnik vrednosti vseh števecv. Na podlagi odčitavanj vrednosti ter podatkov, ki jih posredujejo optične glave, se določi, kateri števci dosegajo zahtevani točnostni razred in katere je potrebno dodatno umeriti ali celo odstraniti iz prodaje.

## **1.4 Slabosti ročnega odčitavanja**

Ob upoštevanju dejstva, da izdelava tovarena več tisoč števecv dnevno in da mora zaradi ekonomskih razlogov en človek v delovnem času odčitati veliko število le-teh, lahko zaključimo, da je to delo precej utrudljivo, zahteva od operaterja visoko stopnjo koncentracije v daljšem obdobju ter naprezanje oči. Poudariti moramo, da ni dovolj razpoznati posameznih števk, ki jih koleščki števnik prikazujejo, ampak je potrebno vsaj zadnji kolešček odčitati do posameznega razdelka natančno (vsaka števka je graduirana z desetimi razdelki, torej sto razdelkov na celoten kolešček; poglavje 4.2). Ker so koleščki oz. števke na njih zelo majhne (celotna višina števke je cca. 1 cm, kar pomeni, da je razdalja med dvema razdelkoma približno 1 mm), oči pri odčitavanju precej trpijo. Tipična frekvenca odčitavanja je en odčitek na tri do štiri sekunde (skupaj s premikanjem od števca do števca), zato morajo operaterji odčitano vrednost določiti relativno hitro.

Sklepamo lahko, da je v daljšem časovnem obdobju težko obdržati zahtevano natančnost odčitavanja (operater lahko zelo hitro napačno oceni oz. spregleda razdelek), poleg tega se lahko pojavijo napake pri samem vnosu odčitanih vrednosti v terminal. Posledica tega je določeno število dobrih števecv, ki so razglašeni za



slabe, ali (redkeje) da slabi števcji uspešno prestanejo testiranje. V prvem primeru se povečajo stroški proizvodnje, v drugem gre slab izdelek na tržišče in manjša zaupanje kupcev v blagovno znamko.

## 1.5 Prednosti avtomatskega odčitavanja

Pričakujemo, da bomo z avtomatizacijo zajema vrednosti števcjev odpravili slabosti sedanjega postopka končnega preizkusa (poglavje 1.4) in s tem izboljšali kvaliteto samega testiranja. Odčitovalni sistem bo deloval z nespremenjeno natančnostjo v daljšem časovnem obdobju (ne bo podvržen stresom, utrujenosti, upadu koncentracije, ...), odpravljena bo nevarnost napak ob vnašanju odčitanih vrednosti v terminal, odpira se tudi možnost arhiviranja slik števnikov v času zajema in s tem naknadna dokazljivost pravilnosti oz. napačnosti odčitavanja, kar je v primeru ročnega odčitavanja praktično neizvedljivo.

## 1.6 Specifikacija zahtev sistema

Da bi sistem v praksi zaživel, morajo biti njegove zmožnosti odčitavanja boljše ali vsaj enake človekovim. Poleg tega mora biti praktičen za uporabo in integracijo v industrijsko okolje. Kot pri vseh industrijskih aplikacijah je tudi v tem primeru zelo pomemben ekonomski vidik oz. cena realizacije sistema ter njegovega vzdrževanja.

Opisane zahteve lahko strnemo v naslednje točke:

- **natančnost odčitavanja:** odčitana vrednost lahko odstopa manj kot za en razdelek zadnjega koleščka števnika (poglavje 4.2);
- **hitrost odčitavanja:** v primeru realizacije sprotnega odčitavanja mora biti čas leta tega pod eno sekundo, možna je tudi izvedba zajema slik števcjev v paketu ter naknadnega odčitavanja;

- **praktičnost:** sistem mora biti čim manj občutljiv na motilne vplive okolja, poleg tega morajo biti postopki, ki jih moramo izvesti pred začetkom uporabe (npr. učenje oblike številke; poglavje 1), enostavni za izvedbo;
- **preprosta integracija v proizvodni proces:** sistem odčitavanja mora z ostalimi sistemi v proizvodnji komunicirati preko protokolov, ki jih le-ti razumejo, izdajati rezultate v primerni obliki za nadaljnjo obdelavo ter z lastno prisotnostjo čim manj spremeniti obstoječe proizvodne postopke;
- **ekonomičnost;** oprema, ki sistem sestavlja, naj bo čim cenejša. Namesto dragih in kvalitetnih kamer naj se uporablja cenejšo opremo; manjša kvaliteta strojne opreme naj se v čim večji meri kompenzira programsko, ker obstaja možnost uporabe velikega števila kamer pri zajemu enega nosilca.

Pri razvoju ročnega odčitovalnika kot pripomočka za servisiranje je potreba po hitrosti še posebej izrazita zaradi ergonomične uporabe, saj v tem primeru ni možno izvesti naknadnega odčitavanja na podlagi zajetih slik, ampak se od sistema pričakuje izdaja rezultata z minimalnimi, v kolikor je mogoče neopaznimi, zakasnitvami. V nasprotnem primeru bi bilo delo z napravo utrudljivo.

Magistrska naloga opisuje pot do zastavljenih ciljev. V nadaljnjih poglavjih si bomo natančno ogledali, na kakšen način je opisane zahteve možno realizirati.

## 2 Splošno o zasnovi sistema

Sistem optičnega odčitavanja električnih števcov je zasnovan tako, da ga lahko obravnavamo kot dva modula. Prvi obsega strojno opremo, drugi predstavlja programsko opremo, ki obdeluje vhodne podatke ter poda želeni rezultat.

Strojni del optičnega odčitovalnika sestavlja osebni računalnik, kamera za zajem slike števca, luč za osvetljavo ter pomožni pribor (senčilo, ki odpravlja motnje ostalih virov svetlobe, stojalo za kamero, razni nosilci, kabli za komunikacijo z ostalimi aparaturami, ...). Naloga strojne opreme je ob pravem trenutku zajeti sliko števca ter jo predati programski opremi, ki iz nje izloči želeno informacijo – odčita vrednost števca na sliki.

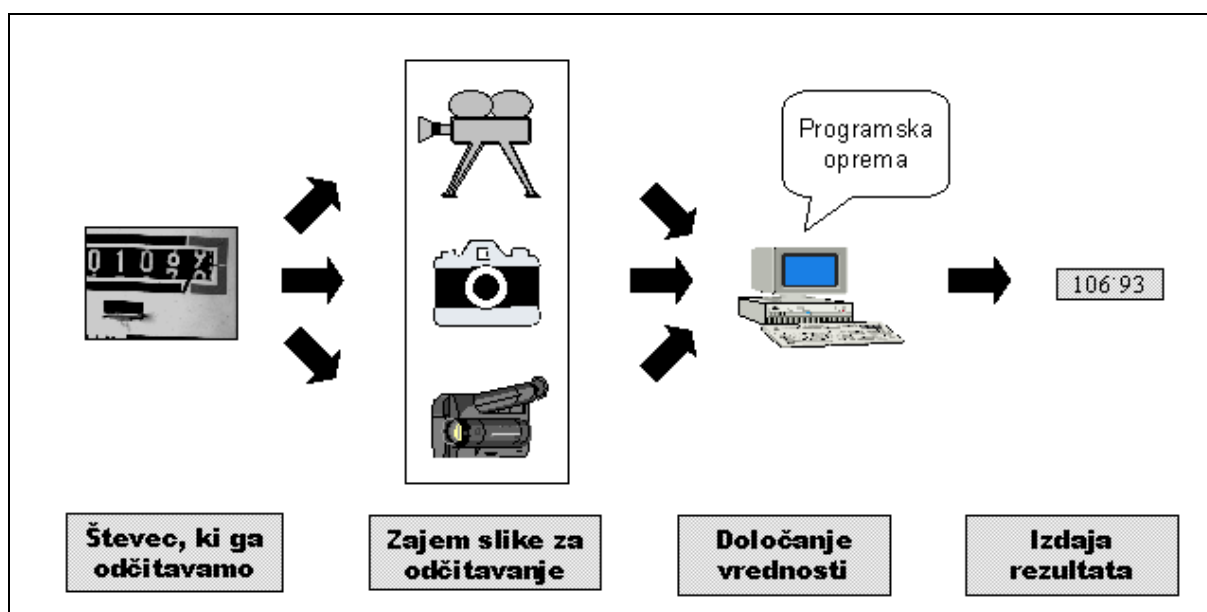
V tem poglavju bomo podali makroskopski opis sistema (poglavje 2.1); ogledali si bomo strojno opremo, ki smo jo uporabljali med razvojem in testiranjem (poglavje 2.2). Oprema podobnih zmogljivosti in karakteristik bo verjetno uporabljena tudi v končni izvedbi sistema, zato podajamo nekaj hitrostnih testov (poglavje 2.3), ki smo jih dosegli z njo. Nadalje bomo opisali način delovanja sistema, za katerega je bil le-ta predviden (končni preizkus izdelkov v tovarni, poglavje 2.4), ter ostale možnosti uporabe, ki se nakazujejo (poglavje 2.5).

## 2.1 Makroskopski opis sistema

Zaradi enostavnejše in s tem cenejše uporabe sistema je pomembno, da je programska oprema v čim večji meri realizirana neodvisno od strojne. Taka zasnova je potrebna, ker lahko slike števecov zajemamo v tovarni z eno kamero, ki se pomika od števca do števca. Druga možnost, ki se nam odpira, je uporaba posebne kamere za vsak števec na nosilcu (podobno kot pri fotoglavah za kontrolo kotve). Nenazadnje pa lahko zajemamo slike števecov tudi s prenosno kamero na servisu ali pri odčitavanju porabe električne energije v gospodinjstvih.

Prednost neodvisnosti programske opreme od načina uporabe sistema je poenostavljen razvoj (izdelati moramo samo eno verzijo programa), testiranje (zagotoviti več brezhibnih verzij je zanesljivo težje kot eno samo) ter vzdrževanje. Pripomnimo naj, da popolne neodvisnosti med strojno in programsko opremo ne moremo doseči. Izpostavimo naj dva modula, kjer je povezava izrazita: modul za delo s kamero ter kontrola zajema slike in izdaje rezultata. Prvega je potrebno spremeniti, če preidemo na drugo kartico za zajem slik; drugi realizira protokol med odčitovalnim sistemom ter zunanjim svetom in je praktično v celoti odvisen od namena uporabe.

Koncept uporabe sistema prikazuje slika 6. Izhodišče postopka predstavlja števec kot objekt odčitavanja. Njegovo sliko lahko zajamemo na različne načine (na sliki so nakazani trije primeri: serijsko odčitavanje števecov na nosilcih, zajem slik za poznejše odčitavanje, kar lahko storimo z drugo kamero, ter ročno odčitavanje na terenu ali servisu s pomočjo prenosne kamere). Zajeta slika se preda programski opremi, ki teče na osebнем računalniku, in se tu obdela – določi se vrednost, ki jo števec kaže.



**Slika 6: Splošni koncept uporabe sistema**

Vidimo, da programska oprema deluje neodvisno od načina zajema slike ter s tem načina uporabe, saj kot vhodni podatek uporabi izključno zajeto sliko, ki mora biti ustrezne kvalitete. Če to ni izpolnjeno, potrebujemo dodatne postopke predobdelave, s katerimi vhodno sliko izboljšamo. S tem koncepta neodvisno realizirane programske opreme ne pokvarimo bistveno, saj lahko fazo predobdelave dodamo na začetek postopka odčitavanja, ne da bi s tem spremenili potek nadaljnjih korakov.

Način podaje rezultata, ki sledi po uspešnem ali neuspešnem odčitavanju vrednosti (npr. ker števca ni bilo na predvidenem mestu na nosilcu), se izvrši na način, ki je specifičen namenu uporabe. V primeru končnega preizkusa v tovarni se rezultat posreduje sistemu, ki ugotovi, ali je konkretni števec dober; pri ročnem odčitavanju se vrednost sporoči v bazo ali izpiše na zaslonu naprave.

Na tem mestu ne bomo podrobno opisovali možnih načinov zajema ali izdaje rezultata, ker sistema še nismo prenesli v industrijsko okolje.

## **2.2 Strojna oprema**

V tem poglavju bomo opisali tisti del strojne opreme, ki ga potrebujemo za realizacijo sistema neodvisno od namena uporabe. Podali bomo napotke za tehnično pravilno ter ekonomsko racionalno izbiro le-te.

### **2.2.1 Osebni računalnik**

Analiza zajete slike se vrši v osebem računalniku, ki mora biti dovolj zmogljiv, da zadosti hitrostnim zahtevam sistema (poglavje 1.6). Čas odčitavanja slike lahko razdelimo na dva dela: interval, v katerem se slika zajame, in interval, v katerem poteka sama analiza in določanje vrednosti. Groba ocena razmerja trajanja zajema slike v celotnem času odčitavanja je 41% na računalniku PentiumII/266, 28% na Pentium/233 ter 22% v primeru uporabe Pentiuma/166 (tabela 1, str. 18). Meritve, ki smo jih izvedli, so pokazale, da je poleg takta procesorja zelo važen podatek tudi hitrost systemskega vodila. Razlog je v tem, da zajem slike in s tem njen prenos v delovni pomnilnik računalnika poteka s hitrostjo vodila. Poleg tega začetni postopki analize, ki trajajo večino časa, delujejo predvsem z vsebinami pomnilniških prostorov, kjer se nahaja začetna slika ter vmesni rezultati. To dejstvo je pri nakupu potrebno upoštevati, zato da ne kupimo računalnika, ki ima veliko razmerje notranjega procesorskega takta proti zunanjemu, ker samo z notranjim taktom ne pridobimo več kot cca. 50% časa.

Pri razvoju smo uporabljali računalnik PentiumII/266 z 64 MB pomnilnika, ki ima 66 MHz zunanje vodilo, kar povsem zadostuje tudi v praksi v primeru, da računalnik analizira slike zajete z eno kamero. Tudi cenovno so ti računalniki dostopni.

### **2.2.2 Kamera za zajem slike**

Sistem optičnega odčitavanja električnih števecov uporablja sivinsko sliko, zato lahko uporabljamo cenejše kamere kot v primeru barvnih slik.

V laboratoriju smo uporabljali modele: JVC TK-5130 (sivinska), SONY XC-77CE (sivinska) ter SONY XC-0035 (barvna, ki je delovala v sivinskem načinu). Vse tri kamere zagotavljajo v našem primeru uporabe dovolj kvalitetno sliko, čeprav nekatere spadajo v slabši kakovostni razred, zato smatramo, da je nakup dragih in zelo kvalitetnih kamer nepotreben.

### **2.2.3 Kartica za zajem slike**

Osebni računalnik sprejme zajeto sliko s kamere preko posebne kartice, ki signal digitalizira ter ga prenese v delovni pomnilnik računalnika. Kvaliteta zajete slike je neposredno odvisna od obeh členov: kamere in kartice, zato je smiselno, da oboje izberemo iz podobnega kakovostnega razreda. Od kombinacije kamere in kartice dodatno zahtevamo, da je število slikovnih elementov po vertikali in horizontali deljivo z osem. Zahteva izhaja izključno iz postopka odčitavanja, ki smo ga razvili, saj ob tej predpostavki nekatere algoritme lahko optimiramo na nivoju samega programiranja. Omejitev ni moteča, saj večina standardnih ločljivosti izpolnjuje to zahtevo.

V laboratoriju smo uporabljali kartico Matrox Meteor (vodilo PCI), ki zagotavlja solidno sliko z malo šuma (ločljivost slike je bila 768x576 slikovnih elementov) in ima podporo za barvno kamero, ki je v našem primeru ne potrebujemo.

## **2.3 Nekaj hitrostnih testov**

V tem poglavju podajamo grobo oceno hitrosti, ki jo lahko od sistema pričakujemo. Za meritve smo uporabili v poglavju 2.2 opisano opremo. Rezultati so prikazani v tabeli 1, pri čemer moramo poudariti, da se vrednost odčitavanja spreminja glede na vsebino slike, zato prikazanih podatkov ne smemo interpretirati, kot da podajajo natančno karakteristiko sistema. Testni vzorec, ki smo ga uporabili pri meritvah, si lahko ogledamo na sliki 7, str. 28. Natančnejšo analizo hitrosti bomo podali v poglavju 1, kjer bomo tudi podrobneje opisali pogoje testiranja.

	Zajem (s)	Odčitavanje (s)	Celotni čas (s)	% časa za zajem
PentiumII/266, 64Mb	0,08	0,12	0,20	41%
Pentium/233, 64MB	0,08	0,21	0,29	28%
Pentium/166, 64Mb	0,08	0,29	0,37	22%

**Tabela 1: Ocena hitrosti sistema na različnih računalnikih**

Celotni čas smo razbili na čas zajema slike in čas odčitavanja, zato ker prvega lahko zmanjšamo samo s hitrejšo strojno opremo, zadnjega pa tudi z boljšimi algoritmi odčitavanja in analize slike. Pri tem je pomembno, kolikšen delež celotnega časa porabi sam zajem (zadnji stolpec tabele 1), saj lahko na ta način ocenimo, koliko truda je vredno vložiti v pohitritev algoritma odčitavanja, preden pričnemo razmišljati o dražji opremi.

Iz tabele 1 je razvidno, da je čas potreben za zajem slike enak pri vseh računalnikih, ki smo jih preizkusili, ker so vsi imeli sistemsko vodilo 66 MHz in enako kartico za zajem. Pri počasnejših Pentium računalnikih čas zajema slike predstavlja majhen delež v celotnem času odčitavanja, ko pa procesorsko moč povečujemo, prične zajem predstavljati ozko grlo.

## 2.4 Uporaba sistema za končni preizkus

Glavni problem, ki ga je potrebno rešiti, preden pričnemo sistem uporabljati v tovarni za končni preizkus, je, kako dovolj hitro zajemati slike števecv na nosilcih. Ponujata se dve rešitvi: uporaba ene kamere in premični nosilec, ki jo bo premikal od števca do števca oz. uporaba ločene kamere za vsak števec posebej. Cenovno je prva varianta dosti bolj ugodna, saj ne prihranimo samo pri nakupu velikega števila kamer (devetintrideset ob predpostavki, da je kapaciteta nosilca štirideset števecv), pač pa moramo v drugem primeru za vsako kamero zagotoviti poseben računalnik in kartico za zajem slike ali pa sistemu dodati posebno strojno opremo, ki bo signale različnih kamer multipleksirala.



V primeru uporabe ene kamere hitrost odčitavanja (programske opreme) ne bi bila kritična, saj bi lahko zajeto sliko analizirali v času, ko bi se kamera pomikala do naslednjega števcu, kar zanesljivo ne bi bilo manj kot nekaj sekund, saj se mora celoten nosilec kamere pospešiti, ustaviti in njegove vibracije iznihati, preden lahko zajamemo sliko zadovoljive ostrine. Glede na to, da v tem trenutku opravljajo to delo operaterji s podobno hitrostjo, smatramo, da ima ta način uporabe sistema precej ugodnejše razmerje cena/učinek kot izvedba z več kamerami in je zato tudi zanimivejši. Možno je tudi realizirati sistem z eno kamero v večjem delu proizvodnje, dražjo izvedbo pa samo na nekaj izbranih mestih, če bi se izkazalo, da je odčitavanje na določenih delih proizvodnje ozko grlo.

## **2.5 Drugi načini uporabe**

V tem trenutku se nakazujeta dve dodatni možnosti uporabe, ki sta tehnično in realizacijsko gledano zelo podobni. Sistem bi lahko opremili s prenosno kamero in računalnikom. Na ta način bi dobili ročni odčitovalnik, ki bi ga lahko uporabljali serviserji pri kontroli in umerjanju števcu ali inkasanti na terenu. S tehničnega stališča je razlika samo v načinu predaje in tolmačenja rezultatov. Npr.: pri umerjanju izdelkov bi lahko sistem avtomatično ugotovil, za koliko je potrebno števec korigirati, ter mogoče tudi sam opravil korekturo, če bi v ta namen realizirali tudi poseben izvršni člen, ki bi bil sposoben spreminjati nastavitve števcu. Pri odčitavanju porabe v gospodinjstvih bi sistem odčitavanja hkrati z vrednostjo števcu poslal bazi podatkov kodo porabnika. S tem bi dobili sistem popolnoma avtomatiziranega obračunavanja porabe električne energije.



## 3 Izbira metod obdelave slik

Na področju računalniškega vida je bilo objavljeno veliko metod za obdelavo, analizo in razumevanje vizualne informacije. Nekatere so bile razvite kot splošno namenske in z njimi lahko obdelujemo široko množico slik, druge so specializirane za določeno področje uporabe v povezavi s slikami z vnaprej pričakovanimi lastnostmi, ker se naslanjajo na *a priori* znanje o njihovi vsebini. Pogosto poteka razvoj določene aplikacije računalniškega vida tako, da se že znane metode vzame kot izhodišče ter se jih prilagodi oz. optimira za specifičen namen uporabe.

Na tem mestu podajamo kratek pregled postopkov obdelave digitalnih slik, ki so sorodni in vsaj delno uporabni pri obdelavi slikovnih vzorcev električnih števecov ter pojasnujemo odločitve o njihovi izbiri za reševanje našega problema.

V poglavju 3.1 nakažemo možnosti za uporabo postopkov predobdelave zajetih slik. Uporabo upravljanja pri segmentaciji slike opisujemo v poglavju 3.2. Nazadnje omenjamo postopke optičnega razpoznavanja znakov (*ang. OCR - optical character recognition*), ki se uporabljajo pri razpoznavanju besedila ali odčitavanju serijskih števil in drugih oznak (poglavje 3.3).

### 3.1 Predobdelava digitalnih slikovnih vzorcev

Večina aplikacij računalniškega vida uporablja ustaljeno shemo predobdelave vhodnih slik, ki so lahko zajete s kamero ali katerokoli drugo napravo za zajem (npr. scanner, rentgen, ...). Prva faza obdelave je obnavljanje zajete slike (poglavje 3.1.1),

ki ga uporabljamo v primeru, ko poznamo matematični model popačenja, ki ga vnaša slikovni podsistem. Če temu ni tako oz. če ocenimo, da moramo slikovni vzorec nadalje izboljšati, uporabljamo postopke izboljšanja kakovosti (poglavje 3.1.2).

### **3.1.1 Obnavljanje zajete slike**

Prva faza obdelave je obnavljanje zajete slike, s katero skušamo zmanjšati popačenja, ki jih v vzorec vnese sistem senzorjev. Največkrat ga izvajamo, ko je vzorec geometrijsko popačen (npr. relativni premik objekta glede na kamero med zajemom, napake leč, ...). Obnavljanje temelji na poznavanju matematičnega modela popačenja in izvajanju temu inverzne operacije. Podrobni opis najdemo v [4], kjer sta podana tudi modela popačenja v primeru neostre slike (slikovna ravnina leži izven gorišča optičnega sistema) ter nejasne slike zaradi premika snemalne naprave med zajemom.

Predvideni načini uporabe sistema za optično odčitavanje električnih števecov (poglavji 2.4 in 2.5) temeljijo na predpostavki, da je sistem pred uporabo izostren in da kamera med zajemom ne spreminja svoje lege glede na števec. Zato smatramo, da postopkov obnavljanja slike v osnovni izvedbi sistema ne potrebujemo. Če bi se izkazalo, da temu ni tako (npr. kamera bi se neprestano gibala med zajemi slik števecov na nosilcu), lahko to fazo obdelave naknadno dodamo sistemu pred ostale korake postopka odčitavanja (poglavje 5.2).

### **3.1.2 Izboljšanje kakovosti slike**

Postopki izboljšanja kakovosti ne potrebujejo natančnega matematičnega modela popačenja. Njihova paleta je široka [4]. Podmnožica teh postopkov je glajenje slike z množico filtrov, s katerimi zmanjšujemo šum na sliki, poudarjamo robove na njej, ali izboljšamo kako drugo globalno lastnost slike (sprememba kontrasta, izostritev, ...).

Sistem optičnega odčitavanja električnih števecv uporablja od tega le najpreprostejše zmanjšanje šuma; pri izdelavi podvzorčene slike (poglavje 6.1) povprečimo področja 8x8 slikovnih elementov, kar deluje kot nizkopasovni filter.

## 3.2 Segmentacija slike

Sliko električnega števca, zajeto z namenom odčitavanja vrednosti števnik, lahko opredelimo kot zapleten vzorec, ki ga kot takega ne moremo razpoznavati s prileganjem, t.j. s primerjanjem zajete (in predobdelane) slike z vnaprej naučenimi primeri slik (učno množico vzorcev) [4]. Odčitavanje vrednosti na tak način bi zahtevalo neobvladljivo veliko število predhodno zajetih slik števca. Obsežnost naloge zmanjšamo tako, da na sliki lociramo posamezne števk (poglavje 1), jih poravnamo, t.j. postavimo v normalno stanje (poglavje 1), ter jih razpoznavamo (jim določamo vrednosti) z eno od možnih izvedb prileganja. Slike posameznih števk torej obravnavamo kot preproste vzorce.

Da bi posamezne števk lahko locirali, moramo izvesti segmentacijo slike. Poslužujemo se upragovljanja (postopek, kjer vse slikovne elemente s pomočjo praga osvetljenosti razvrstimo v dva razreda: temno in svetlo področje), saj je slikovni vzorec električnega števca zelo primeren za tako razčlenjevanje, v kar se lahko prepričamo s pogledom na sliko 7, str. 28. Vidimo, da so števk na sliki izrazito svetle, njihov okoliški prostor pa je temen. S pravilno postavitvijo praga lahko večino slikovnih elementov na področju števnik pravilno razvrstimo med števk in vmesni prostor (slika 24 desno, str. 50).

Pri upragovljanju uporabljamo globalni prag (njegova vrednost je odvisna samo od svetilnosti slikovnih elementov), ker je časovno manj zahteven od lokalnega (ki je odvisen tudi od okolice slikovnega elementa) ali dinamičnega (ki je nadalje odvisen od prostorskih koordinat). Poskusi kažejo, da globalni prag zadostuje namenu

uporabe, t.j. razčlenjevanju slike električnega števca. V [4] je podrobno opisan postopek določanja optimalnega praga s pomočjo histograma porazdelitve relativnih frekvenc sivih nivojev svetilnosti slike. Mi smo za naš sistem razvili nekoliko drugačen postopek (poglavje 6.2), ki je časovno manj zahteven in pri uporabi s splošnimi slikami daje slabše rezultate, ker ne določi optimalnega praga, ampak samo zadovoljiv približek, ki pa v našem primeru zadostuje (kar potrjujejo poskusi), saj so histogrami slik električnih števecv izrazito bimodalni z zelo majhnim številom slikovnih elementov med obema glavnima modulusoma (sliki 13, str. 42 in 16, str. 45).

S pomočjo binarizirane (in podvzorčene, poglavje 6.1) slike uspešno izvedemo detekcijo števnik (poglavje 1) ter lociramo posamezne številke (poglavje 1).

### **3.3 Razpoznavanje znakov s postopki OCR**

Postopki optičnega razpoznavanja znakov (*ang. optical character recognition - OCR*) [2, 3, 1] so se uveljavili v aplikacijah, ki so povezane z razpoznavanjem slikovnih vzorcev, na katerih so predstavljeni simboli (številke, znaki abecede in drugo). Tem slikam želimo prirediti oznake simbolov, ki so na njih narisane. Primeri takih aplikacij so razpoznavanje serijske številke električnega števca [1] in pretvorba besedil predstavljenih s pomočjo slike v ASCII obliko.

Glede na namen uporabe in na cilje, ki jih aplikacije optičnega razpoznavanja znakov želijo doseči, lahko postopke OCR delimo na industrijske in neindustrijske [1]. Od industrijskega OCR zahtevamo veliko robustnost in zanesljivost odčitavanja. Izdani rezultat ne vsebuje samo prepoznanih simbolov, ampak tudi morebitna obvestila o verjetnosti napake. Neindustrijski OCR ima cilje drugače zastavljene. Razpoznati skuša čim večji odstotek znakov, četudi zaradi tega žrtvuje zanesljivost.

Izvedb razpoznavanja znakov je veliko, skupna značilnost vseh postopkov pa je določanje značilke simbola na sliki (in s tem manjšanje redundantne informacije

vhodnega vzorca) ter njihovo primerjanje z značilkami predhodno zapomnjenih primerov simbolov (vzorci učne množice). Rezultat postopka je oznaka simbola iz učne množice, katerega značilke se (po vnaprej določenem kriteriju) najbolje ujemajo z značilkami razponavanega simbola (oz. pri industrijskem OCR tudi podatek o napaki, če se razpoznavani simbol ne ujema dovolj z nobenim iz učne množice). Značilke, ki se uporabljajo v praksi, so precej občutljive na to, ali je razpoznavani znak na sliki viden v celoti, kar pomeni, da pri delno manjkajočih znakih uspešnost razpoznavanja močno upade.





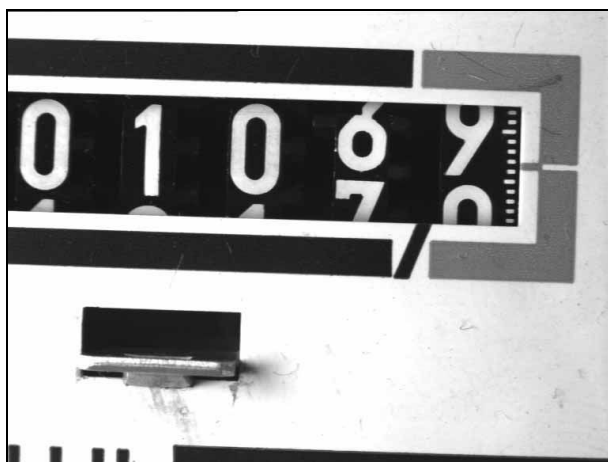
## 4 Odčitavanje vrednosti

Na tem mestu se bomo posvetili problematiki odčitavanja vrednosti števca ob predpostavki, da je njegova slika že zajeta in posamezne številke na njej uspešno locirane. Predstavili bomo glavne probleme, ki jih je za uspešno odčitavanje vrednosti števk potrebno rešiti, ter ustrezne rešitve tudi nakazali.

V poglavju 4.1 bomo opisali problem odčitavanja vrednosti in podali utemeljitev, zakaj klasični postopki optičnega razpoznavanja znakov v našem primeru niso uporabni. Poglavje 4.2 formalno predpiše natančnost, ki jo mora sistem doseči. Postopek, ki dosega ustrezne karakteristike in ga predlagamo namesto uporabe OCR, je opisan v poglavju 4.3.

### 4.1 Problematika natančnega odčitavanja

V uvodnem poglavju (1.4) smo že omenili, da pri odčitavanju vrednosti števca ne zadostuje razpoznati posameznih števk (jih opisati s simboli od 0 do 9) ter jih združiti v končni rezultat, temveč potrebujemo natančnejšo določitev vrednosti. Da bi omenjeno zahtevo lažje pojasnili, si oglejmo sliko 7, ki prikazuje števec, katerega vrednost moramo odčitati. Vidimo, da prikazujejo levi trije kolesčki števnik številke, ki so vidne v celoti, zadnji dve pa se nahajata v vmesnih položajih. Prva tri mesta bi zato lahko razpoznali s postopki OCR, zadnjih dveh pa ne, saj OCR v splošnem ne deluje zadovoljivo, če znaki, ki jih razpoznavamo, niso vidni v celoti (poglavje 3.3). Poleg tega si s podatkom, da je vrednost zadnje številke 9 ali 0, ne moremo veliko pomagati, ker z njegovo pomočjo ne moremo izdati rezultata potrebne natančnosti.



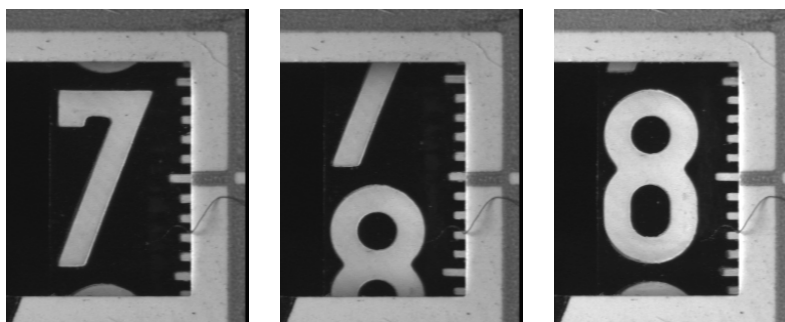
**Slika 7: Primer slike števca, ki mu moramo odčitati vrednost**

Vrednost, ki jo na podlagi prikazane slike želimo razpoznati, je 106`93 kWh ali 106`94 kWh (pri tem smo upoštevali, da zadnje mesto števnik predstavlja desetinke merske enote). Pri uporabi postopkov OCR in ob predpostavki, da bi z njihovo pomočjo lahko pravilno določili simbole vsem števkam (tudi tistim, ki niso vidne v celoti), bi lahko izdali naslednje rezultate: 106`9 kWh, 106`0 kWh, 107`9 kWh ter 107`0 kWh. Ne samo, da noben od naštetih rezultatov ni zadosti natančen, tudi med seboj so si precej različni in brez dodatne informacije nimamo možnosti ugotoviti, kateri je najbližji dejanski vrednosti. Zaradi tega bi bila karakteristika odčitovalnika nezvezna, ponovljivost pa slaba v položajih koleščka, ki so blizu preskokov z ene vrednosti številke na drugo.

Na tem mestu naj omenimo, da bo neodvisno od uporabljene metode odčitavanja vsaka karakteristika nezvezna že zaradi dejstva, da je ločljivost zajete slike omejena in zaradi tega lahko zaznamo samo končno število različnih položajev koleščka. Poleg tega je rezultat izdan v digitalni obliki, kar pomeni, da se je v postopku odčitavanja izvršila kvantizacija ustrezne analogne vrednosti. Pomembno pri karakteristiki je, da je njena nezveznost na vseh mestih manjša od dopustne napake odčitavanja (v našem primeru dovoljujemo absolutno napako sistema 0`01 kWh; poglavje 4.2), za kar ocenjujemo, da postopki OCR ne morejo doseči.

## 4.2 Zahtevana natančnost odčitavanja

Zahtevana natančnost odčitka sledi neposredno iz graduacije zadnje številke, s katero je povezano število različnih položajev kolesčka, ki jih nedvoumno lahko zaznamo s prostim očesom. Slika 8 prikazuje povečano sliko zadnjega kolesčka v treh različnih položajih. Vidimo, da je vsaka številka razdeljena na deset razdelkov in vsaj toliko različnih položajev le-te lahko ločimo.



**Slika 8: Povečana slika številke z dobro vidno graduacijo decimalnega mesta**

Dejanska ločljivost je nekoliko večja, saj lahko zaznamo tudi položaje kolesčka med dvema razdelkoma (na polovici, četrtini, ...), a različni odčitovalci si verjetno ne bi bili več enotni pri določanju odčitane vrednosti v teh primerih; zato smatramo, da je štetje razdelkov največ, kar lahko človeško oko objektivno in enolično oceni pri odčitavanju vrednosti. Absolutna napaka, ki jo sistem ne sme prekoračiti, je torej  $0,01$  kWh, kjer je upoštevano, da zadnja številka predstavlja desetinke merske enote.

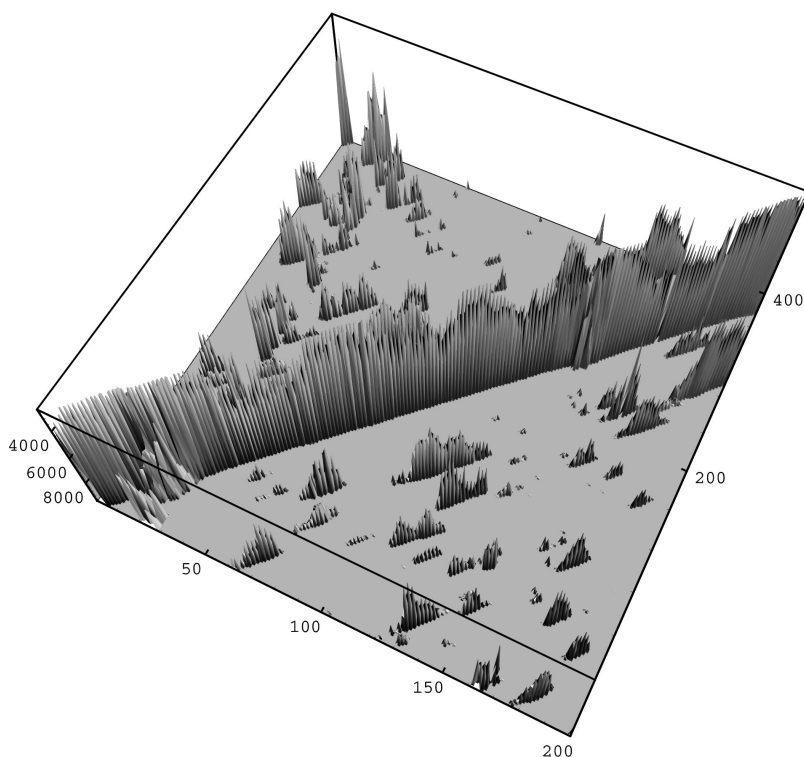
## 4.3 Doseganje zelene natančnosti

Postopek odčitavanja, ki ga v praksi uporabljamo, je naslednji. Na sliki lociramo posamezne številke (poglavje 1) ter jih ločeno primerjamo s sliko raztegnjenega kolesčka števnik, ki je prikazan na sliki 9 (obrnjen  $90^\circ$  protiurno).



**Slika 9: Primer traku, ki predstavlja raztegnjeno sliko kolesčka**

Sliko posamezne števkke pomikamo vzdolž traku ter za vsak položaj ugotovimo razdaljo med njima (razdaljo definiramo kot število točk, v katerih se sliki razlikujeta). Kjer je razdalja najmanjša, je slika števkke najbolj podobna traku. Iz tega podatka lahko neposredno ugotovimo, v katerem položaju se kolesček na sliki nahaja in s tem kolikšna je njegova vrednost. V ustreznost te metode se lahko prepričamo s pomočjo slike 10, kjer so prikazane razdalje med sliko števkke v vseh položajih kolesčka in trakom preko njegove celotne dolžine.



**Slika 10: Prikaz razdalj med sliko števkke v vseh položajih kolesčka in trakom preko njegove celotne dolžine**

Odmik na sliki vzdolž sprednje stranice podaja zasuk kolesčka (njegovi položaji so oštevilčeni od 1 do 200, ker smo ga zajeli v prav toliko različnih pozicijah; pri vsakem polnem razdelku ter v položajih na njihovi polovici). Odmik vzdolž stranske stranice podaja položaj traku, pri kateri je izračunana razdalja med njim in sliko številke (položaji na traku so oštevilčeni od 1 do 400, ker smo izbrali tako velikost števk na njem, da smo imeli na voljo dvakrat več položajev traku kot zajetih pozicij kolesčka). Razdalje so ponazorjene z višino funkcije, pri čemer je os usmerjena navzdol ter zaradi večje preglednosti vrednosti večje od 8000 popravljene na to vrednost. Na sliki jasno izstopa izredno strm greben po njeni sredini, ki kaže, da se številke v vseh položajih kolesčka najbolje prilegajo traku na tistem mestu, ki je analogen kolesčkovi poziciji.

Formula za oceno vrednosti številke je naslednja:

$$v = k(x - x_0)$$

kjer pomenijo oznake:

- $v$  vrednost številke na sliki,
- $k$  konstanta traku,
- $x$  oddaljenost mesta najboljšega ujemanja od začetka traku,
- $x_0$  začetek številke "0" na traku.

Vidimo, da je v nasprotju s postopki OCR dobljena vrednost kolesčka dosti bolj korelirana z njegovim zasukom in je natančnost določanja položaja omejena samo z ločljivostjo oz. kakovostjo zajete slike in traku, ki ga uporabljamo. Omenimo še, da tak postopek uporabljamo za vse številke na sliki (OCR ne uporabljamo niti za skrajno levo številko), ker se lahko zaradi prenosa med dekadami prav vsi kolesčki nahajajo v kateremkoli vmesnem položaju.



## 5 Postopek odčitavanja vrednosti

Primerjanje slik števk s trakom za določitev njihovih vrednosti (poglavji 4.3 in 1) je ključna faza odčitavanja števcov, na kateri temelji celotna implementacija zamisli, sama zase pa še zdaleč ne reši celotnega problema. Kje na sliki se številke nahajajo? Koliko jih je? Ali je števec sploh prisoten na sliki? Na ta in na še precej drugih vprašanj moramo odgovoriti, preden lahko primerjavo s trakom sploh izvedemo. Nazadnje je potrebno dobljene vrednosti posameznih števk združiti v končni rezultat, kar je težja naloga, kot se na prvi pogled zdi.

V tem poglavju si bomo ogledali izhodiščne kriterije, ki so ključno vplivali na zasnovo sistema ter nas usmerjali pri odločitvah med raziskavami (poglavje 5.1). Temu sledi generalni opis celotnega postopka odčitavanja (poglavje 5.2).

### 5.1 Zasnova sistema

Sistem optičnega odčitavanja električnih števcov je bil razvit z namenom uporabe v industrijskem okolju in kot tak mora strogo izpolnjevati pogoje pravilnega delovanja. Gotovo je njegov najkompleksnejši del programska oprema, ki realizira odčitavanje. Doseganje brezhibnega delovanja programov pri velikosti izvorne kode reda polovice megabajta je težavno opravilo, ki pa ga lahko precej poenostavimo s pravilno zasnovo. Izpostavimo naj naslednja prijema:

- **modularna zasnova:** postopek odčitavanja je zasnovan kot niz korakov (modulov), od katerih vsak izgleda kot črna škatla, ki z ostalimi moduli komunicira preko majhnega števila dobro definiranih mest;
- **uporaba odprte zanke:** izogibali smo se zaprtim zankam, ob prisotnosti katerih se je dosti težje prepričati, da sistem pravilno deluje v vseh pogojih, na katere lahko naletimo.

Zaradi modularne zasnove je določanje in lociranje napak bistveno lažje, saj lahko spremljamo potek izvajanja od modula do modula ter opazujemo, na katerem nivoju dobimo nepričakovane rezultate. Prav tako je delo olajšano v odsotnosti zaprtih zank, saj bi se napake lahko pojavile šele po večjem številu iteracij (večina pod zelo redkimi in specifičnimi pogoji), zato bi bilo njihovo odkrivanje bistveno težje in dolgotrajnejše.

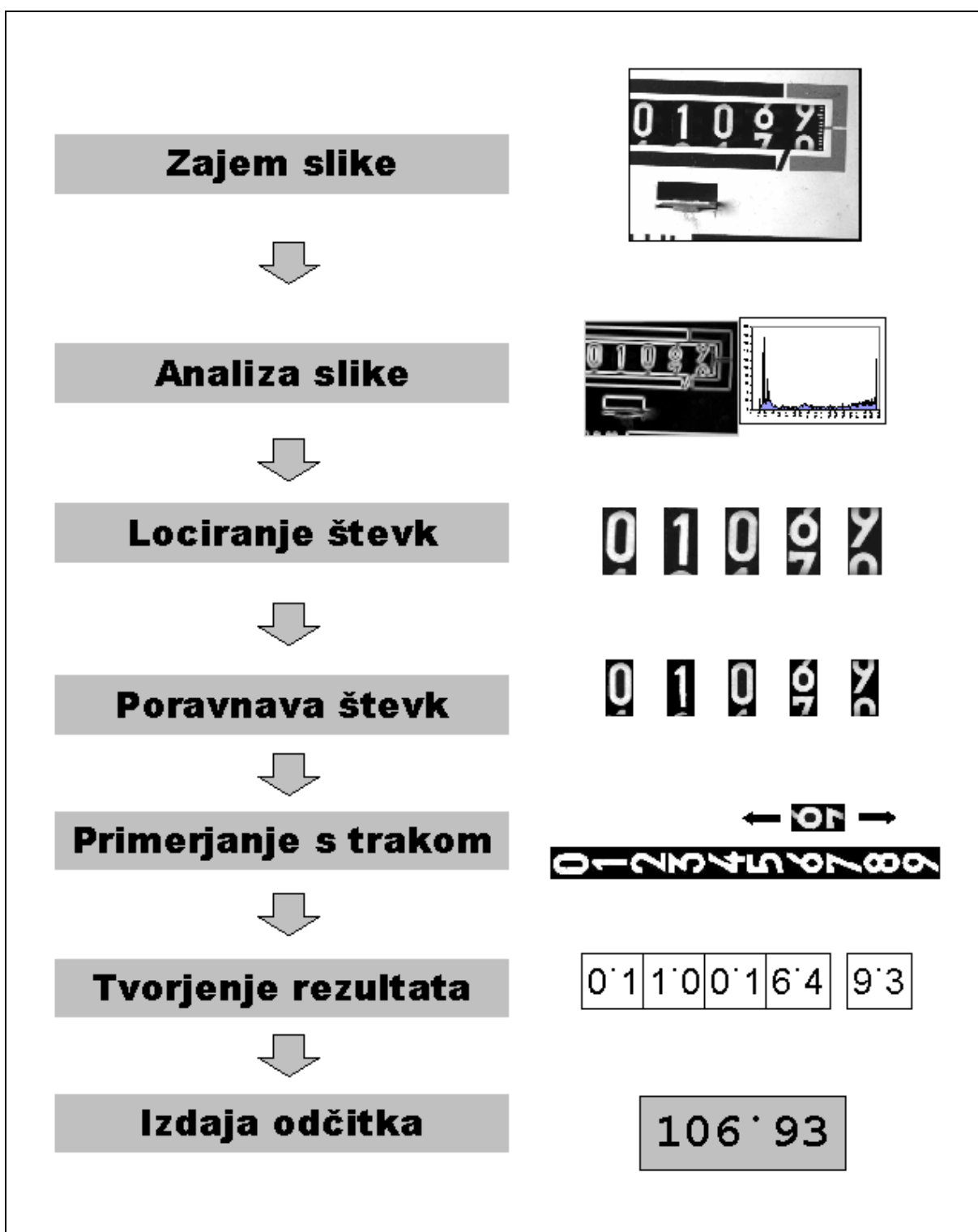
## 5.2 Opis postopka

Bločni diagram postopka odčitavanja je prikazan na sliki 11. Opazimo lahko, da je zasnovan po načelih, ki smo jih opisali v prejšnjem razdelku: posamezni koraki postopka dobijo rezultate samo od predhodnega nivoja ter pripravijo rezultate za naslednjega.

Vloga posameznih korakov je naslednja:

- **zajem slike števca:** korak, ki je specifičen načinu uporabe sistema (poglavja 2.1, 2.4, 2.5), ima nalogo ob pravem trenutku zajeti sliko ustrezne kvalitete. Ključno vlogo pri tem ima ustrezna strojna oprema (poglavje 2.2). Problematika zajema slike je tehnični problem, zato je v tem delu ne obravnavamo;





Slika 11: Shematski prikaz postopka odčitavanja

- **analiza slike:** začetna obdelava slike ter določanje njenih globalnih parametrov je prvi korak postopka odčitavanja. Kot rezultat nam vrne podvzorčeno in s tem podatkovno manj obsežno sliko (poglavje 6.1), ki jo uporabimo za hitrejše lociranje položajev števk. Tudi vrednost praga binarizacije (poglavje 6.2), ki ga potrebujemo pri nadaljnji obdelavi, se določi na tem nivoju;
- **lociranje števk:** princip določanja vrednosti posameznim števkom na sliki, ki ima zadovoljivo natančnost, smo že opisali (poglavje 4.3). Njihova lokacija na sliki in samo število le-teh pa se od zajema do zajema spreminja, zato njihovih leg ne moremo vnaprej predvideti, ampak jih moramo s posebnim postopkom določiti pri vsakem odčitavanju posebej (poglavje 6);
- **poravnava števk:** števki s trakom ne moremo primerjati takoj, ko določimo njihovo lokacijo, saj je števec lahko zajet pri drugačni povečavi, kot je bil ob izgradnji traku (ob učenju oblik števk; poglavje 10.1) in je lahko zavrten za nezanemarljiv kot (primer na sliki 7, str. 28, kjer je števec opazno zavrten v smeri ure). Z ustrezno poravnavo (poglavje 1) dosežemo, da slike števk izgledajo zadosti podobne traku oz. so zadovoljivo podobne števkom, ki bi jih zajeli pod enakimi pogoji, kot smo zajeli slike za izgradnjo traku;
- **primerjanje s trakom:** vsaki števk se določi njena vrednost (poglavji 4.3 in 1). Števk ni opisana samo z ustreznim simbolom (od 0 do 9), ampak s precej natančnejšo vrednostjo, katero dobimo z linearno interpolacijo vrednosti števk, med katerima leži slika neznane števk (natančnost je omejena s kvaliteto traku in slike števk);
- **tvorjenje rezultata:** vrednosti posameznih števk nas kot take ne zanimajo. Da bi ugotovili, katero vrednost števec kaže, moramo posamezne vrednosti števk združiti v končni rezultat. Vrednosti, dobljene v predhodnem koraku, ne moremo enostavno pomnožiti z dekadnimi utežmi posameznih mest in sešteti, ker je pri tem potrebno upoštevati mehanske prenose med kolesčki ter paziti na

---

nepretrgano in ponovljivo karakteristiko brez preskokov v vseh možnih položajih, v katerih se lahko posamezne številke nahajajo (poglavje 1);

- **izdaja odčitka:** korak je specifičen načinu uporabe in odvisen od strojne opreme, ki jo v ta namen uporabljamo. V najpreprostejšem primeru (ročni odčitovalnik) sistem enostavno prikaže vrednost na ustreznem zaslonu ali drugi napravi, ki služi v ta namen. Pri končnem preizkusu v tovarni pa je stvar nekoliko težja. Realizirati moramo ustrezen protokol pošiljanja podatkov, ker hkrati lahko deluje več odčitovalnih mest, ki centralnemu računalniku sporočajo rezultate. Poleg tega moramo poskrbeti za preverjanje pravilnosti podatkov in detekcijo izpada omrežja.

V nadaljevanju bomo posamezne faze odčitavanja podrobno opisali.



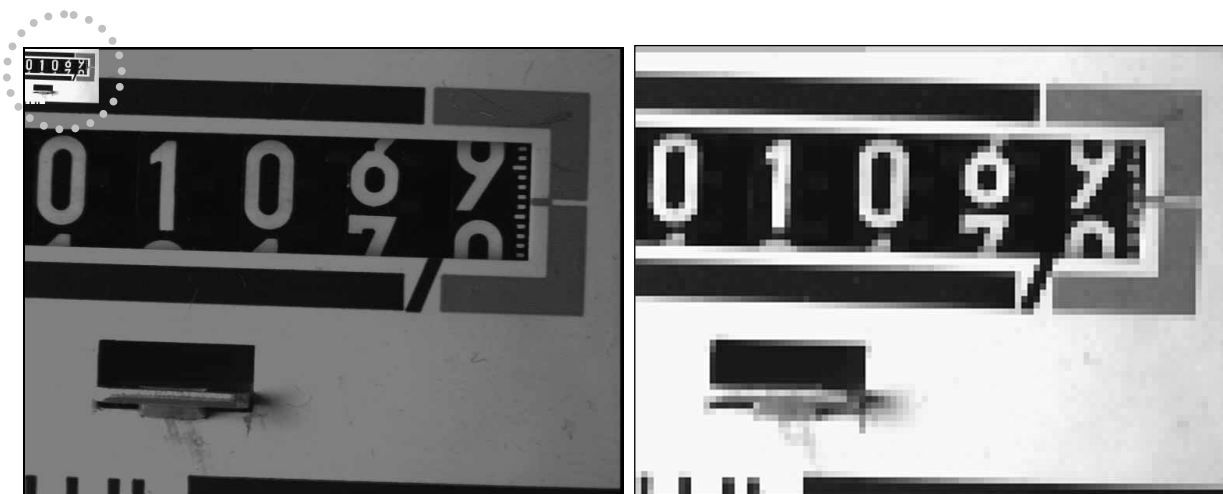
## 6 Začetna analiza slike

Začetna analiza nam nudi prvi vpogled v zajeto sliko. Z njeno pomočjo opravimo nizkonivojsko obdelavo, kjer še ne razpoznavamo ali interpretiramo njene vsebine kot prepoznane ali izločene objekte, pač pa se operacije vršijo neodvisno od pomenske vsebine slike. Naloge, ki jih moramo na tem mestu opraviti, so filtriranje s ciljem, da zmanjšamo šum, prilagoditev na različne stopnje osvetljenosti ter začetna redukcija odvečne informacije. Prvi dve zahtevi nam povečata robustnost postopka, zadnja pa nam ga pohitri.

Začetna analiza nam priskrbi podvzorčeno sliko (poglavje 6.1), ki je od prvotne precej manjša (vsebuje manj podatkov), zato jo z namenom pohitritve odčitavanja uporabljamo pri detekciji števnik namesto originalne; dobimo tudi vrednost praga binarizacije (poglavje 6.2), s katerim iščemo lokacije števk in robove števnik na sliki.

### 6.1 Tvorjenje podvzorčene slike

Tvorjenje podvzorčene slike je v našem primeru preprost postopek. Pravokotnim področjem 8x8 točk priredimo na podvzorčeni sliki vrednost njihovega povprečja (natančneje navzdol zaokrožen celoštevilčni del aritmetične sredine). Pri tem smo predpostavili, da je število točk na sliki po horizontali in vertikali deljivo z osem (poglavje 2.2.3). Podvzorčeno sliko, prikazano preko levega zgornjega kota (nekoliko zatemnjene) originalne, si lahko ogledamo na sliki 12 levo. Zajeto sliko, ki je bila podlaga za tvorjenje podvzorčene slike, prikazuje slika 7, str. 28.



**Slika 12: Prikaz podvzorčene slike v zgornjem levem kotu originalne (levo) ter podvzorčena slika, povečana na velikost originalne (desno)**

Že na prvi pogled vidimo, da je redukcija podatkov na tako dobljeni sliki velika (1:64). Povprečenje slikovnih elementov pa ne zreducira samo prvotne informacije, pač pa deluje tudi kot preprost nizkopasovni filter, zato je na podvzorčeni sliki šum zmanjšan in v nadaljevanju lahko uporabljamo postopke obdelave, ki so manj robustni in bolj občutljivi na šum, so pa zato hitrejši (to izkoriščamo pri detekciji števnik ter pri hitrem določanju področja, kjer iščemo vertikalne meje števca – poglavji 7.3 in 8.1). Slika 12 desno prikazuje podvzorčeno sliko, povečano na velikost originalne (v nadaljevanju bomo vse podvzorčene slike prikazovali pri tej povečavi).

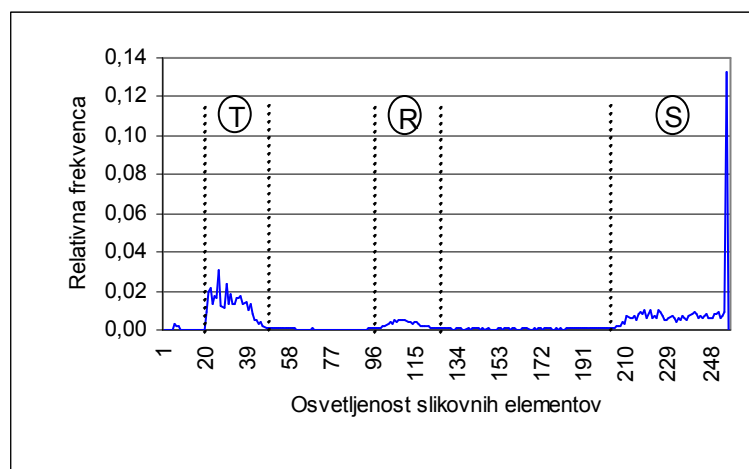
Velikost področja 8x8 slikovnih elementov smo izbrali na podlagi kvalitativne ocene rezultatov poskusov s področji velikosti 2x2, 4x4, 8x8, 16x16, 32x32 in 64x64 slikovnih elementov. Ocenjujemo, da manjša področja od 8x8 slikovnih elementov nezadovoljivo filtrirajo šum na podvzorčeni sliki, medtem ko večja lahko z nje zabrišejo pomembne detajle. Podvzorčeno sliko potrebujemo za grobo lociranje števnik in položajev posameznih števk na sliki (poglavje 7.3), zato njene ločljivosti ne smemo pretirano manjšati. Zlasti pri manjših povečavah zajema se lahko zgodi, da se ob uporabi večjih področij posamezni deli števk na podvzorčeni sliki sploh ne pojavijo, kar za reševanje našega problema ni sprejemljivo.

Izbira velikosti področja je povezana z ločljivostjo zajete slike. Področja velikosti 8x8 slikovnih elementov so primerna za ločljivost, s katero smo v laboratoriju zajemali slike (768x576 slikovnih elementov). V primeru uporabe slikovnih sistemov, ki zajemajo slike pri drugačnih ločljivostih, priporočamo izgradnjo podvzorčenih slik s spremenjeno velikostjo področij. Pri izbiri velikosti je bistveno, da se razmerje dimenzij originalne in podvzorčene slike čim bolj ohrani (npr. pri ločljivosti 256x256 bi verjetno dobili boljše rezultate z uporabo področji velikosti 4x4 slikovnih elementov). Tudi omejitve na področja kvadratne oblike ni striktno nujna. Uporabljamo jo zato, ker omogoča pohitritev nekaterih postopkov.

## 6.2 Določanje praga binarizacije

V literaturi [4, 5, 6, 7] obstaja veliko postopkov določanja optimalnega praga binarizacije (s pomočjo histograma porazdelitve relativnih frekvenc sivih nivojev svetilnosti slike, z detekcijo robov na gradientni sliki, z upoštevanjem sosednosti slikovnih elementov, s spremljanjem poteka osvetljenosti, ...) ter tudi različni tipi le-tega (globalni, lokalni, dinamični, ...). Vsak postopek je primeren za reševanje določenega tipa problemov ter ima svoje prednosti in slabosti (npr. večja časovna zahtevnost postopka, ki operira nad gradientno sliko v primerjavi s postopkom, ki uporablja samo histogram, zato pa so robovi področij na binarizirani sliki pravilnejši in šum binarizacije manjši).

V našem primeru izbira postopka ni kritična, saj že bežen pogled na zajeto sliko električnega števca (slika 7, str. 28) razkrije, da je vzorec večinoma dvonivojski in zaradi tega njegov histogram izrazito bimodalni, kar lahko natančneje opazujemo na sliki 13.



**Slika 13: Histogram zajetega slikovnega vzorca, na katerem so označena posamezna zanimiva področja (T označuje temno področje, R vsebuje slikovne elemente rdeče barve, S označuje svetlo področje)**

Opazimo, da na histogramu izstopata dve področji. Območje osvetljenosti slikovnih elementov od cca. 19 do 50 (področje na sliki označeno s črko T) predstavlja temna področja števnik, t.j. vmesne prostore med posameznimi števki ter njihova ozadja. Na drugem koncu histograma pa opazimo področje, ki predstavlja bele površine čelne plošče (interval cca. od 200 do 255; označeno s črko S) in je precej bolj izrazito od temnega, ker čelna plošča predstavlja večji del celotne zajete slike. V območju osvetljenosti od 200 do 240 se nahajajo slikovni elementi števka, za katere se izkaže, da so v povprečju osvetljene rahlo šibkeje kot čelna plošča. Poleg tega so natisnjene na koleščke t.j. ukrivljene površine, zato je njihova osvetljenost prostorsko spremenljiva, kar rezultira v širšem področju histograma približno enakomerne osvetlitve. V intervalu od cca. 91 do 127 lahko opazimo še dokaj neizrazito področje, ki predstavlja rdečo barvo (označeno s črko R), ki obkroža zadnjo števk in s tem poudarja, da gre za decimalno mesto.

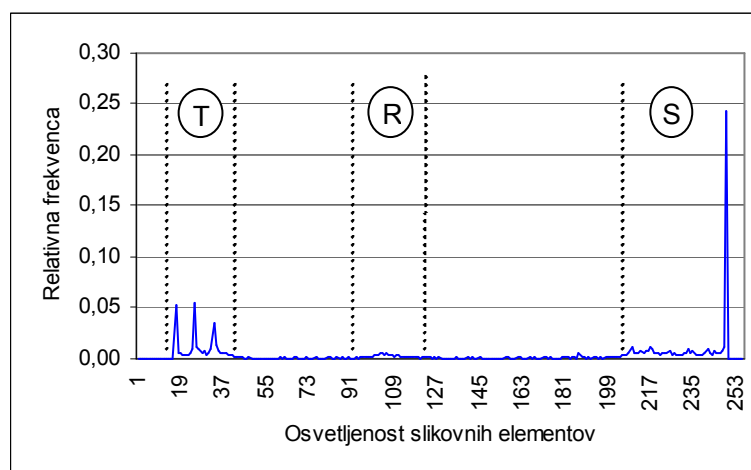
Našteta področja oz. intervali niso stalni, ampak se spreminjajo v odvisnosti od tega, kako močno osvetlimo vzorec, ko ga zajemamo, ter od tipa čelne plošče (npr. nekatere sploh nimajo rdeče barve). V laboratoriju lahko vedno dosežemo tako



osvetlitev, da se omenjena področja raztezajo preko celotnega območja (od 0 do 255), v praksi pa to ni vedno možno, zato tudi praga binarizacije ne moremo vnaprej določiti fiksno, ampak moramo njegovo vrednost ugotoviti za vsako sliko posebej (ali v primeru velike časovne zahtevnosti za celotno serijo slik, ki so zajete pod enakimi pogoji).

### 6.2.1 Postopek določanja praga binarizacije

Prag binarizacije zajete slike bomo določili z analizo podvzorčene (slika 12 desno) zaradi večje hitrosti izvajanja postopka. Njen histogram si lahko ogledamo na sliki 14.



**Slika 14: Histogram podvzorčene slike (T označuje temno področje, R vsebuje slikovne elemente rdeče barve, S označuje svetlo področje)**

Vidimo, da se histograma podvzorčene in prvotne slike (primerjava s sliko 13) po obliki ne razlikujeta dosti, le vrhovi posameznih področij so nekoliko manj izraziti. Podobnost oblik ne preseneča, saj podvzorčenje ne vpliva na distribucijo osvetljenosti slikovnih elementov (v mejah napake, ki jo le-to povzroči) in histogram podvzorčene slike predstavlja nepristransko oceno poteka histograma osnovne slike.

Razlika, ki jo s prostim očesom skoraj ne moremo opaziti, lahko pa sklepamo nanjo, je rahlo nižji povprečni nivo osvetljenosti na novi sliki. Razlog je v tem, da smo povprečne vrednosti osvetljenosti posameznih področij 8x8 slikovnih elementov zaokrožili navzdol (zaradi hitrejše izvedbe podvzorčenja) in s tem povzročili majhno pristranskost njihovih ocen. Omenjeni pojav ne moti zaradi majhnega števila slikovnih elementov med obema glavnima modulusoma histograma, s čimer praga binarizacije originalne slike (ki ga določamo preko podvzorčene) ni potrebno določiti zelo natančno.

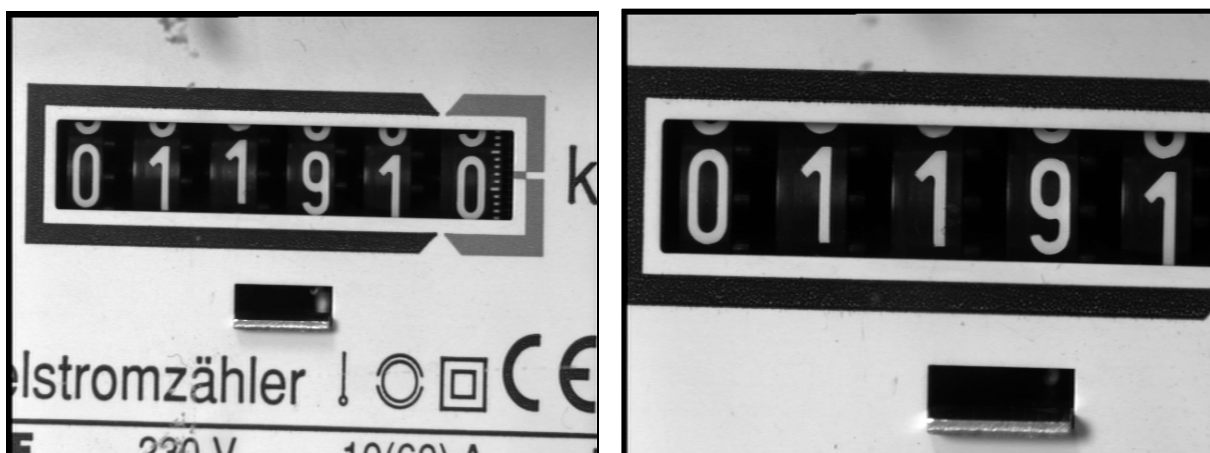
Prag binarizacije nam bo predstavljala srednja vrednost sivih nivojev:

$$SV = \sum_{i=0}^{255} i \cdot p_i$$

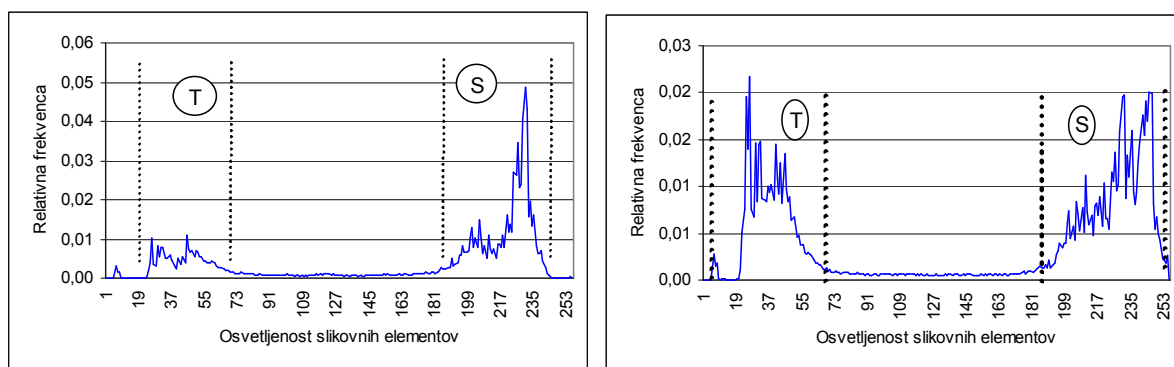
kjer pomenijo oznake:

- $SV$  srednja vrednost histograma,
- $i$  sivi nivo slikovnih elementov,
- $p_i$  ocena verjetnosti slikovnega elementa s sivim nivojem  $i$ .

Pri tem ne moremo uporabiti histograma celotne slike, ampak dobljenega tako, da bo vseboval samo relativne frekvence robnih točk. S tem bomo dosegli neobčutljivost postopka na različne povečave števnika, med katerimi se razmerje svetlih in temnih površin močno spreminja. Npr. ko sliko zajemamo z manjšo povečavo (števnik na sliki zavzema manjšo površino – slika 15 levo prikazuje zajem števca pri višini števnika 100 slikovnih elementov), na vzorcu močno prevladuje bela površina čelne plošče, ko pa povečavo večamo (na sliki 15 desno prikazan števnik zavzema po višini 150 slikovnih elementov), predstavljajo črna področja med števki vedno večji delež slikovnih elementov in s tem se tudi močno spremenita deleža modulusov na histogramih, ki sta prikazana na sliki 16.

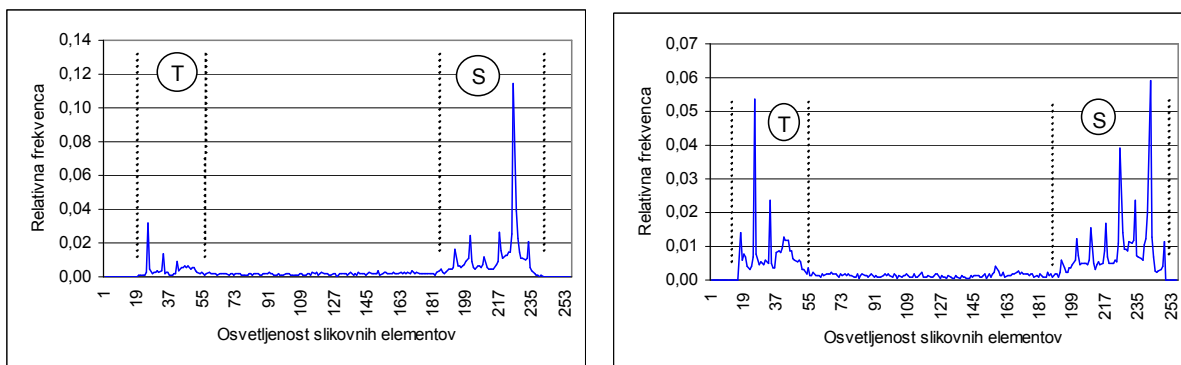


**Slika 15: Primer dveh slik istega števca, zajetih pri različnih povečavah (višini števca sta od leve proti desni 100 in 150 slikovnih elementov)**

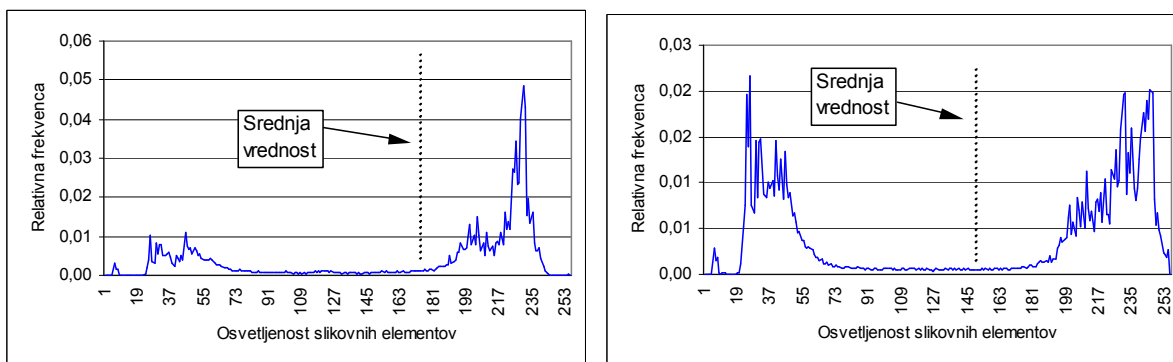


**Slika 16: Histograma slik števca, zajetega pri različnih povečavah (levo višina števca 100, desno 150 slikovnih elementov)**

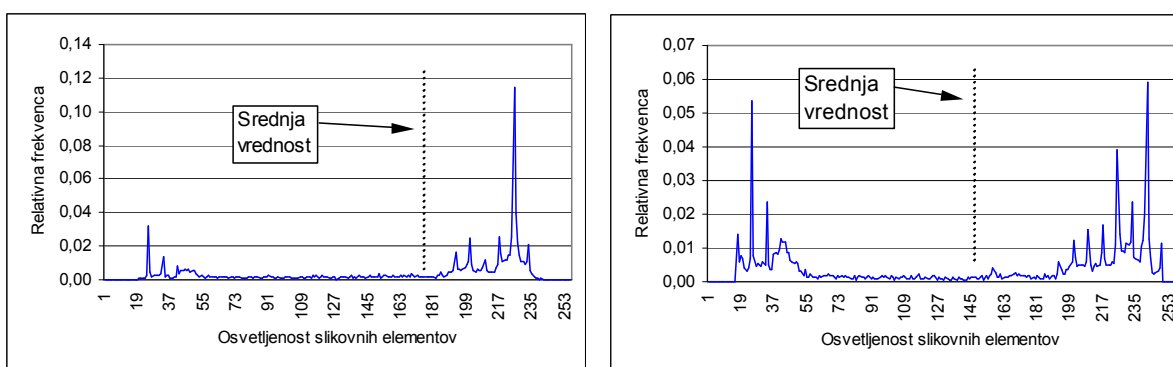
Vidimo, da je pri manjši povečavi (slika 16 levo) razmerje temnega proti svetlemu področju (območje označeno s črko T proti območju označenem s črko S) ustrezno manjše v primerjavi z razmerjem pri večji povečavi (slika 16 desno). Ta pojav se seveda odraža tudi na podvzorčeni sliki (slika 17) in ima za posledico premik srednje vrednosti histograma tako originalne kot tudi podvzorčene slike (sliki 18 in 19). Podatki so številčno predstavljeni v tabeli 2.



**Slika 17: Histograma podvzorčenih slik števca, zajetega pri različnih povečavah (levo višina števnik 100, desno 150 slikovnih elementov)**



**Slika 18: Histograma slik števca, zajetega pri višini 100 (levo) in 150 (desno) slikovnih elementov z vrisanimi srednjima vrednostima**



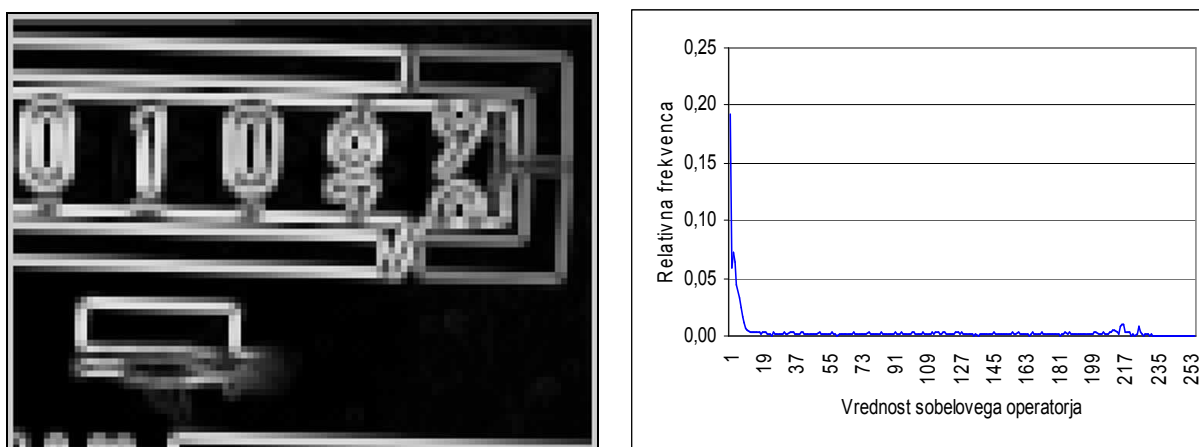
**Slika 19: Histograma podvzorčenih slik števca, zajetega pri višini 100 (levo) in 150 (desno) slikovnih elementov z vrisanimi srednjima vrednostima**

	Povečava 100 s.e.		Povečava 150 s.e.	
	Zajeta slika	Podvzorčena slika	Zajeta slika	Podvzorčena slika
Površina temnega področja	0,235	0,190	0,372	0,343
Površina svetlega področja	0,674	0,629	0,564	0,523
Razmerje temnega proti svetlemu	0,349	0,302	0,660	0,656
Srednja vrednost histograma	169	167	149	146

**Tabela 2: Pregled nekaterih karakteristik histogramov, zajete in podvzorčene slike pri dveh različnih povečavah**

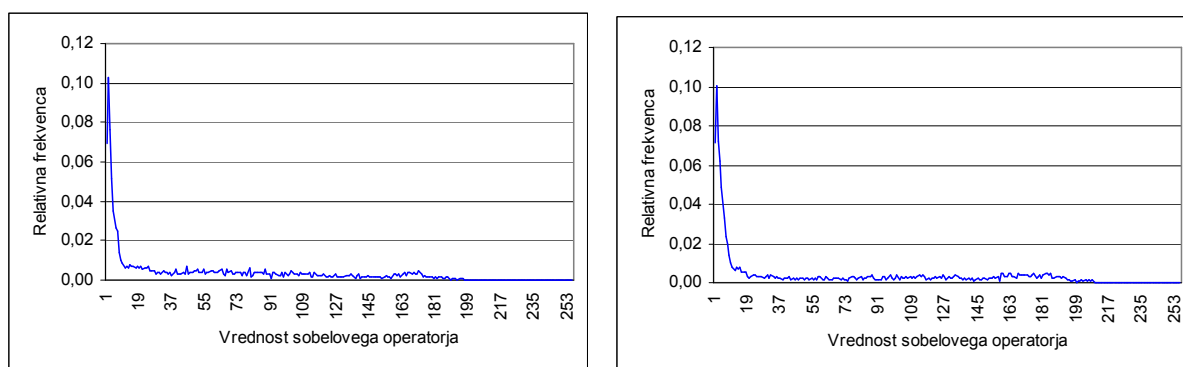
S tabele 2 je razvidno, da se tiste lastnosti histograma originalne slike, ki nas zanimajo, ohranijo tudi na histogramu podvzorčene slike. Tako se deleža temnega in svetlega področja ter njihovo relativno razmerje pri podvzorčenju ne spremenita bistveno, prav tako se v večji meri ohranita srednji vrednosti histogramov. Vidimo pa, da se ti parametri precej spreminjajo v odvisnosti od povečave, kar je nezaželeno.

Če bi prag binarizacije določali direktno s pomočjo srednje vrednosti histograma osnovne ali podvzorčene slike, bi bil naš postopek občutljiv na te spremembe parametrov. Namesto tega s pomočjo sobelovega operatorja določimo, kje na podvzorčeni sliki se nahajajo robne točke in samo te upoštevamo pri tvorjenju histograma. Sobelova slika in njen histogram sta prikazana na sliki 20.



**Slika 20: Rezultat sobelovega operatorja nad podvzorčeno sliko (levo) ter njegov histogram (desno)**

Slika 21 prikazuje histograma sobelove slike še za primera slik, ki sta zajeti pri različnih povečavah (originala sta prikazana na sliki 15). Histogram sobelove slike ima v primeru slike števnikova zelo karakteristično obliko. Na samem začetku ima izrazit vrh, ki predstavlja točke, v okolici katerih so spremembe osvetljenosti zelo majhne.



**Slika 21: Histograma sobelovih slik števnikov, zajetih pri različnih povečavah (levo višina števnikova 100, desno 150 slikovnih elementov)**

Vrh je izrazit, ker večji del slike zasedajo relativno velike ploskve skoraj konstantne osvetljenosti, in je posledica šuma, ki povzroča, da se vrednosti slikovnih elementov, ki prikazujejo homogene ploskve, nekoliko spreminjajo.

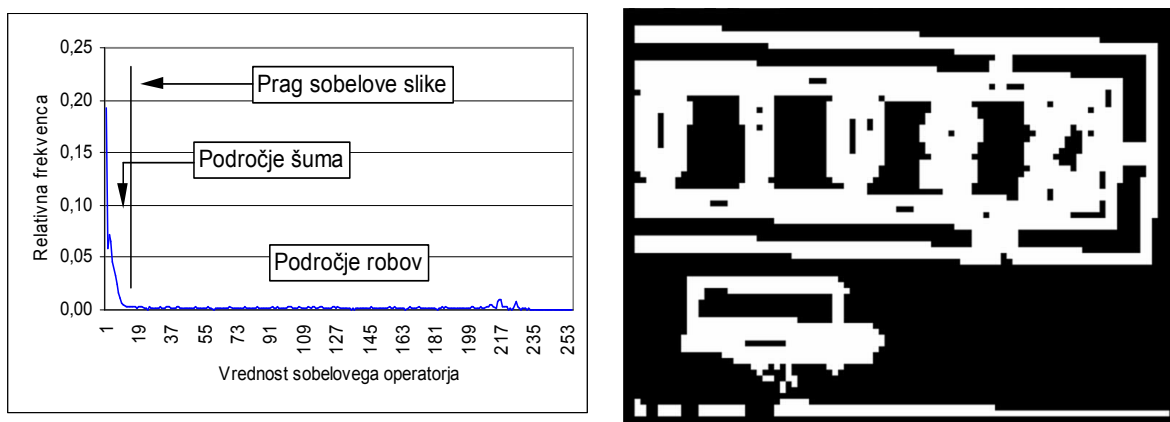
Robne točke poiščemo tako, da določimo vrednost sobelovega operatorja, od katere naprej smatramo točke za robne, t.j. tiste, katerih osvetljenost se je spremenila zaradi prehoda iz enega območja slike v drugega in ne zaradi šuma.

To storimo tako, da pregledujemo sobelov histogram od začetka, dokler mu vrednost pada v posamezni točki ali v katerikoli točki, ki je od trenutne oddaljena za določeno vrednost (poskuse smo izvajali z oddaljenostimi od 8 do 16; pri vseh vrednostih smo dobili zelo podobne rezultate). Ko upadanja ni več, smo dosegli rob začetnega hriba in s tem določili iskano vrednost praga sobelove slike (slika 22 levo). Vse točke, ki

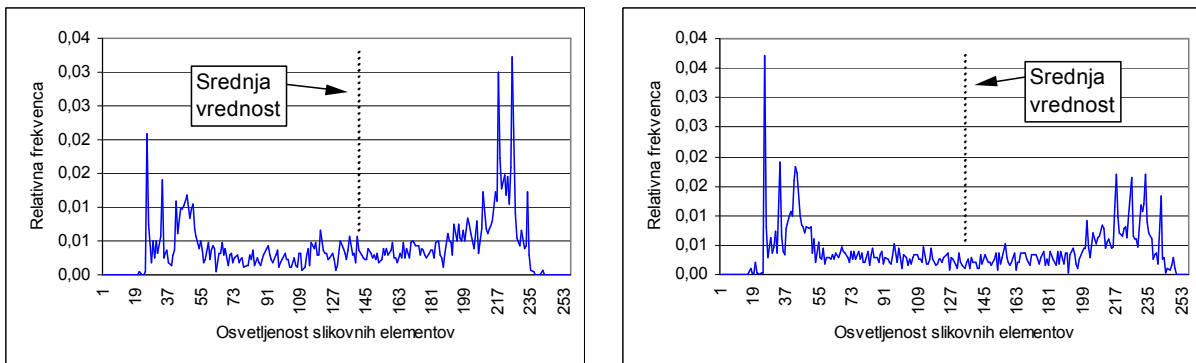
imajo vrednost sobelovega operatorja večjo od roba hriba, so robne točke podvzorčene slike (slika 22 desno), ki jih upoštevamo pri izgradnji histograma.

Za primera števecv s slike 15 sta histograma prikazana na sliki 23. Kvalitativno in kvantitativno (tabela 3) lahko ocenimo, da smo precej zmanjšali vpliv različnih povečav. Histograma na sliki 23 imata bolj izenačena deleža temnega in svetlega področja kot histograma na sliki 17. S tako dobljenim histogramom lahko prag binarizacije podvzorčene in s tem tudi osnovne slike določimo z izračunom njegove srednje vrednosti.

Podatki so prikazani v tabeli 3, kjer vidimo, da smo uspeli zmanjšati vpliv različne povečave na razmerje med temnim in svetlim področjem. Tudi srednja vrednost se manj premika in je postavljena bolj v središče med oba modusa histograma.



**Slika 22: Sobelov diagram z vrisanim pragom (levo) ter prikaz robnih točk podvzorčene slike (desno)**

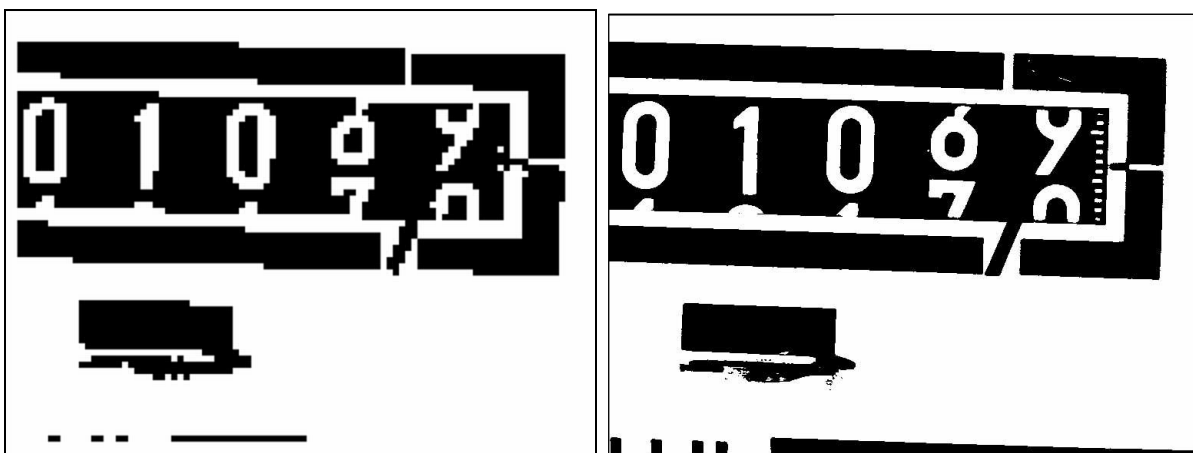


**Slika 23: Histograma robnih točk slikovnega vzorca števecv, zajetega pri višini števnik 100 (levo) in 150 (desno) slikovnih elementov**

	Povečava 100 s.e.		Povečava 150 s.e.	
	Zajeta slika	Podvzorčena slika	Zajeta slika	Podvzorčena slika
Površina temnega področja	0,190	0,269	0,343	0,348
Površina svetlega področja	0,629	0,426	0,523	0,383
Razmerje temnega proti svetlemu	0,302	0,631	0,656	0,909
Srednja vrednost histograma	146	142	136	132

**Tabela 3: Primerjava parametrov histograma robnih točk s histogramom podvzorčene slike**

V ustreznost dobljenega praga binarizacije se lahko prepričamo na sliki 24, kjer sta prikazani binarizirana podvzorčena ter osnovna slika.



**Slika 24: Binarizirana podvzorčena (levo) ter originalna (desno) slika**



V zvezi z določanjem praga binarizacije pojasnimo še šum binarizacije, ki ga večkrat omenjamo v nadaljevanju. Pojem opisuje nezaželen pojav določenega števila izoliranih točk ali manjših področij bele barve na predhodno navidez homogeni črni podlagi ali obratno. Pojav praktično vedno spremlja binarizacijo z vpragovljanjem in je tem izrazitejši, čim manj poudarjene moduse ima histogram. V primeru slikovnega vzorca števnika pojav skoraj ni opazen, povsem odsoten pa tudi ni. Zato pri nadaljnji obdelavi osnovne slike uporabljamo postopke, ki so temu prilagojeni.

S tem, ko smo generirali podvzorčeno sliko ter poiskali njen prag binarizacije, je začetna analiza slike končana. Oboje potrebujemo v nadaljnjih korakih odčitavanja števca.



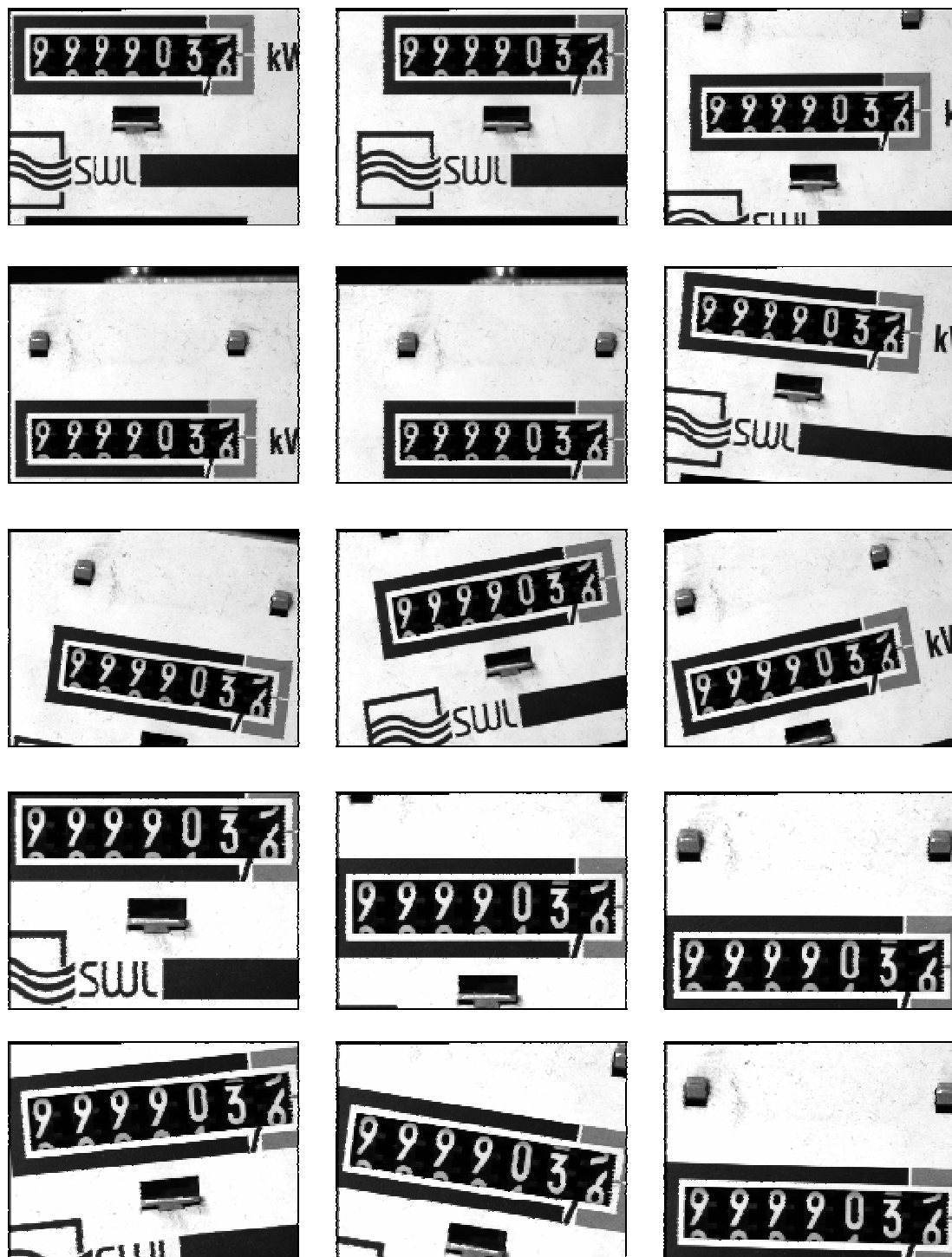
## 7 Detekcija položaja števnika

Kvalitetna in robustna detekcija števnika ter s tem povezano lociranje posameznih števk na zajeti sliki je brez dvoma ključni korak celotnega postopka odčitavanja vrednosti števecov. Poznavanje natančne lokacije števk je predpogoj za uspešno in smiselno primerjavo njihovih slik s trakom. Praktično vse aplikacije industrijskega vida nam lahko potrdijo, da je segmentacija s kamero zajete slike ter s tem povezano razpoznavanje zanimivih področij na njej najtežja faza celotnega postopka in če je izvedena zadovoljivo, deluje v večini primerov dobro tudi sistem kot celota [4].

V tem poglavju si bomo ogledali nekaj slik števecov, na katerih se števnik nahaja na različnih položajih znotraj slikovnega vzorca, ter z njihovo pomočjo predpisali, kako naj naš sistem na te situacije reagira (poglavji 7.1 in 7.2). V nadaljevanju bomo postopek, ki te zahteve izpolnjuje, tudi natančno opisali (poglavje 7.3). Nazadnje bomo v poglavju 7.4 pokazali primer števnika, kjer postopek odpove.

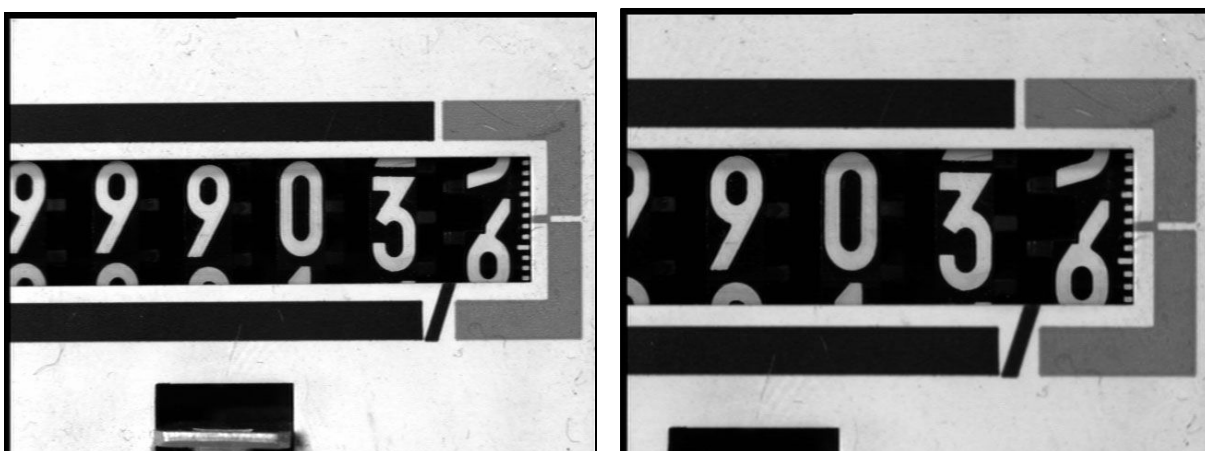
### 7.1 Števnik in njegov položaj na sliki

Da se bolje seznanimo s problematiko detekcije števnika, si oglejmo sliko 25, kjer je isti števec zajet v različnih položajih, ki so zadovoljivi za odčitavanje. Opazimo lahko, da se med slikami ne spreminja samo lokacija števnika, ampak tudi njegova rotacija in povečava. Želimo, da postopek segmentacije pravilno prepozna števnik ter locira števke na njem kljub prisotnosti prikazanih razlik med posameznimi zajemi.



Slika 25: Različni položaji števniaka znotraj zajete slike

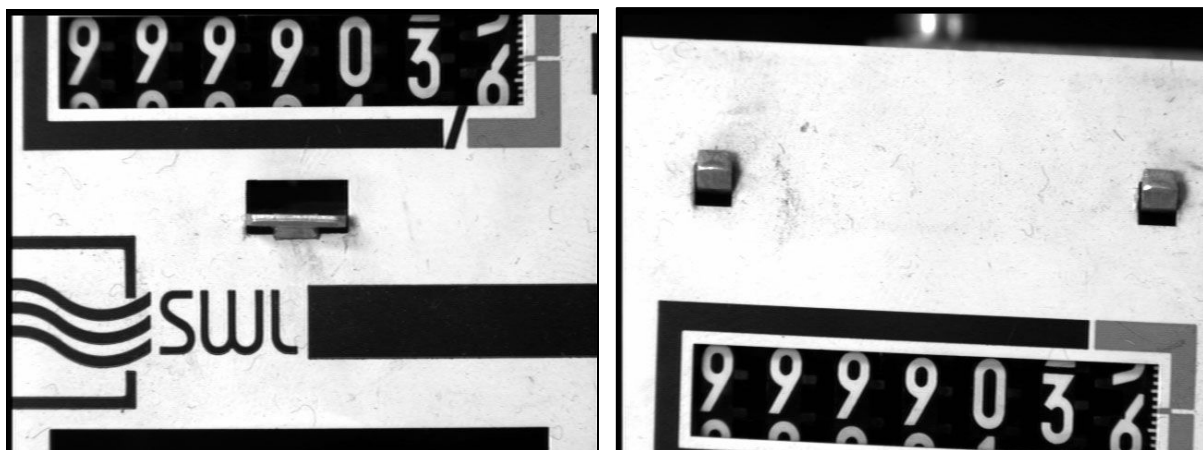
Poudarimo naj, da se na število števok kot enega od parametrov števnik (oz. posameznega tipa števca), ki se ne spreminja, ne moremo zanesti in ga uporabiti pri detekciji prisotnosti števnik, saj so lahko števci zajeti pri tako veliki povečavi (slika 26), da vse števk na sliki niso vidne (na sliki 26 levo je uporabnih pet števok, ker skrajno levo ni vidna v celoti, na desni sliki pa so uporabne štiri; vseh števok na števniku je sedem, kar lahko vidimo na sliki 25).



**Slika 26: Primer dveh slik števca zajetega pri tako veliki povečavi, da nekatere števk niso vidne ali niso vidne v celoti**

To pride v poštev pri preciznih odčitavanjih, kjer želimo zajeti sliko pri večji ločljivosti, da lahko natančneje odčitamo vrednost. Fantomska obremenitev števcov med končnim preizkusom (poglavje 1.3) navadno ne traja dovolj dolgo, da bi se vsem števokam spremenila vrednost, zato skrajno leve ostanejo na vrednosti 0 (oz. na predhodni vrednosti). Rezultat odčitavanja (natančneje: razlika vrednosti po in pred obremenitvijo) se ne spremeni, če le-te na sliki niso vidne. V zvezi s tem naj omenimo še problem delno vidnih števok na levi strani, saj se pri delnem zajemu številčnice pogosto zgodi, da je skrajno levo števk vidna le delno (na sliki 26 sta oba primera taka). Takih števok ne moremo uporabiti pri primerjavi s trakom, saj lahko dobimo napačen rezultat. Postopek segmentacije mora te števk zavrniti kot neustrezne.

Naslednje vprašanje, ki se postavlja, je, kdaj sliko števnika detektirati kot ustrezno. Slika 27 prikazuje dva položaja števca, kjer je števnik prisoten, opazimo pa, da se v vseh primerih vsaj delno dotika zgornjega ali spodnjega roba slike.



**Slika 27: Primer zajete slike števca, kjer se števnik dotika zgornjega (levo) ali spodnjega (desno) roba**

V takih primerih želimo, da postopek detekcije sliko zavrne kot neustrezno, saj ne moremo zanesljivo ugotoviti, kje se nahaja začetek števnik in s tem sredine števk, katere potrebujemo za določanje lokacij, na katerih se številke najboljše ujemajo s trakom (določimo lahko, kje se njihove slike najboljše ujemajo, vrednost, ki jo razpoznamo pa je različna od prave za toliko, kolikor vrednosti bi prispevala napaka ocene sredine števnik; poglavje 11.1).

V nasprotju z vertikalnimi mejami pa za horizontalne ne postavljamo takih omejitev zaradi že omenjene možnosti zajema slik pri tako veliki povečavi, da števnik ni viden v celoti, s tem pa tudi ni vidna vsaj ena horizontalna meja. Izkaže se, da lahko števec uspešno odčitavamo tudi brez ene ali obeh vidnih horizontalnih mej.

## 7.2 Specifikacija zahtev za detekcijo

V prejšnjem poglavju smo opisali razne situacije, na katere lahko naletimo pri zajemu slike števca. Na tem mestu bomo povedano strnili v seznam zahtev, katere naj bi postopek za detekcijo števnik zadovoljil.

Zahteve so naslednje:

1. števnik želimo prepoznati ne glede na to, koliko števk je na sliki vidnih (postavljena je zahteva po minimalnem številu števk zaradi postopka detekcije, v naši implementaciji tri v celoti vidne števk; poglavje 7.3);
2. ne želimo biti odvisni od povečave oz. velikosti števnik na sliki. Določena je minimalna velikost (poglavje 7.3), pod katero nima smisla iti, ker bi bil rezultat odčitavanja nenatančen in nezanesljiv. Maksimalna velikost pa je predpisana s tem, da morajo biti vsaj tri števk vidne v celoti;
3. števnik je lahko za znaten kot zarotiran so– ali protiurno (nekaj primerov je prikazanih na sliki 25);
4. prepoznati moramo samo števk, ki so na sliki vidne v celoti; delno vidne moramo izločiti kot neustrezne;
5. vertikalne meje števnik morajo biti v celoti vidne (števnik se ne sme dotikati zgornjega ali spodnjega roba zajete slike),
6. vidne horizontalne meje števnik ne smejo biti pogoj za uspešno detekcijo;
7. pri detekciji se ne želimo naslanjati na same slike števk oz. pisavo in obliko, ker imamo tudi števkce s posebnimi kodami na decimalnem mestu (slika 3, str. 8) in števkce, kjer se namesto arabskih uporabljajo lokalne pisave (npr. števkci, ki so namenjeni izvozu v islamske države).

V nadaljevanju opisujemo postopek, ki izpolnjuje omenjene zahteve in ga lahko uspešno uporabljamo za detekcijo različnih tipov čelnih plošč (vendar ne vseh; poglavje 7.4).

## 7.3 Postopek detektiranja števnik

Da bi uspešno in hitro detektirali števniki ob pogojih, ki smo jih navedli v prejšnjih dveh poglavjih, potrebujemo vsaj eno njegovo karakteristično lastnost, ki jo je relativno preprosto prepoznati. Izkaže se, da lahko v ta namen uporabimo zaporedje temnih področij med števki. S slike 24 na str. 50 lahko človek brez problemov prepozna števniki samo z opazovanjem te njegove lastnosti, in to tako na originalni (desno) kot tudi na podvzorčeni sliki (levo). Postopek, s katerim lahko to opravi računalnik, je možno dokaj enostavno realizirati.

V prvem koraku izdelamo sliko, ki izhaja iz binarizirane podvzorčene slike, kjer je za vsako temno točko določeno, koliko je oddaljena od najbližje svetle točke. Merilo oddaljenosti je najmanjše število točk, ki jih moramo prepotovati, preden pridemo do svetle točke. Pri tem lahko potujemo v vseh osmih smereh, torej tudi diagonalno. Rezultat si lahko ogledamo na sliki 28, kjer je večja oddaljenost temne točke od najbližje svetle prikazana grafično z različno intenziteto osvetljenosti (svetlejša točka pomeni večjo oddaljenost). Siva barva, ki prevladuje na sliki (zlasti na spodnjem delu), označuje svetle točke na binarizirani podvzorčeni sliki.



**Slika 28: Grafični prikaz odmika temnih točk od svetlih**



V nadaljevanju moramo najbolj oddaljene točke razširiti v pravokotna temna področja ter tako dobiti seznam pravokotnih temnih področij na sliki. Postopek je naslednji.

***Postopek izgradnje seznama pravokotnih temnih področij na sliki***

1. inicializiramo seznam temnih področij;
2. na sliki poiščemo najbolj oddaljeno temno točko. Če je njena oddaljenost od najbližje svetle točke manjša od vnaprej predpisane vrednosti, postopek prekinemo, sicer nadaljujemo z naslednjim korakom. Če obstaja več točk z enako oddaljenostjo, izberemo tisto, katere vsota oddaljenosti njenih sosedov je največja. V primeru, da je tudi takih točk več, izberemo katerokoli izmed njih;
3. temno točko, izbrano v prejšnjem koraku, razširimo v pravokotno področje, ki ga dodamo na seznam temnih področij (kako to storimo, je opisano spodaj; "Postopek razširjanja izbrane točke v pravokotno regijo");
4. vsem točkam v pravokotnem temnem področju s koraka 3 pripišemo oddaljenost 0, da preprečimo ponovno širjenje in s tem večkratno uvrščanje iste regije na seznam, ter se vrnemo na korak 2.

Komentar postopka:

- v koraku 2 želimo izbrati točko, ki se nahaja čim bližje sredini temnega področja, saj lahko na ta način njeno razširjanje vanj poenostavimo (postopek je opisan v nadaljevanju). S predlaganim načinom dobimo suboptimalno rešitev, ki se izkaže kot zadovoljiva, saj so temna področja števnik pravokotniki z omejenim intervalom razmerij med višino in širino (podrobna analiza sledi v nadaljevanju), torej lahko predvidimo, za koliko odstopajo od kvadrata in s tem vemo, da se vse točke z največjo oddaljenostjo nahajajo v bližini njihovih središč;
- s tem ko predpišemo v koraku 2 najmanjšo oddaljenost temne točke, ki jo še razširimo v temno področje, smo določili najmanjšo povečavo, pri kateri števnik še prepoznamo, tako kot zahteva poglavje 7.2.

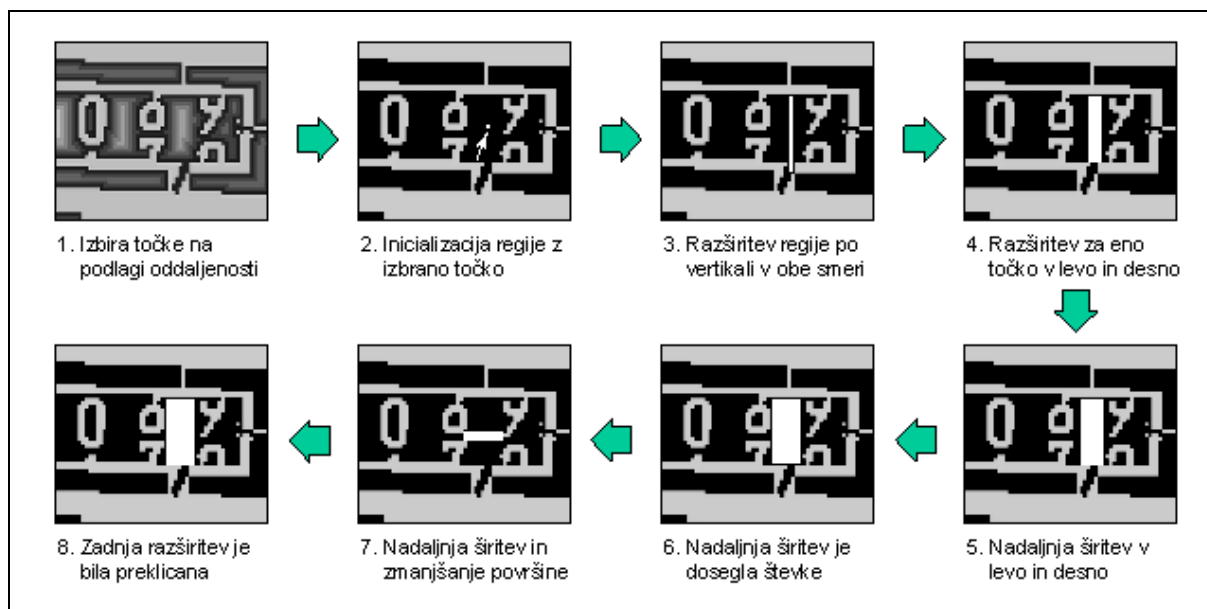
**Postopek razširjanja izbrane točke v pravokotno regijo**

1. strukturo "temna regija" inicializiramo tako, da obsega izbrano točko;
2. "temno regijo" razširjamo vertikalno v obe smeri, dokler ni zaobjeta nobena svetla točka ali dokler ne dosežemo roba slike;
3. na vrednost DA postavimo logični spremenljivki "razširitev\_v\_levo" in "razširitev\_v\_desno";
4. če ima spremenljivka "razširitev\_v\_levo" vrednost DA, skušamo "temno regijo" razširiti za en pixel v levo, tudi če jo moramo po višini ustrezno zmanjšati, da ne zaobjamemo nobene svetle točke. Razširitev obdržimo, če ima rezultirajoča regija večjo površino kot prejšnja, sicer jo prekličemo in spremenljivko "razširitev\_v\_levo" izbrišemo (postavimo na logično NE);
5. analogno storimo z razširitvijo v desno;
6. če ima vrednost DA katerakoli od spremenljivk "razširitev\_v\_levo" ali "razširitev\_v\_desno", se vrnemo na korak 4, sicer postopek prekinemo.

**Komentar postopka:**

- v koraku 2 razširimo začetno točko v regijo, ki je široka en slikovni element ter visoka, kolikor nam to dopušča okolica izbrane točke. V primeru temne regije števnikar smo omejeni z njegovim zgornjim in spodnjim robom. Če bi točko razširjali horizontalno, bi se lahko zapletli v številke, kar bi imelo za posledico, da se temna regija ne bi raztezala po celotni višini števnikar in postopek združevanja temnih regij (opisan v nadaljevanju) bi jo zaradi prevelike razlike v višini izločil;
- v korakih 4 in 5 skušamo regijo razširiti, da bi se horizontalno raztezala od predhodne do naslednje številke. Preprost kriterij, da regijo razširjamo, dokler se njena površina veča, se izkaže za zadovoljivega.

Postopek je ilustriran na sliki 29, kjer lahko opazujemo zaporedje korakov, ki nas pripeljejo od izbrane točke do temne regije.



**Slika 29: Potek razširjanja izbrane točke v temno regijo**

V posameznih korakih prikazanih na sliki 29 je dogajanje naslednje:

1. s pomočjo slike, ki prikazuje oddaljenosti vseh temnih točk od najbližje svetle točke, izberemo ustrezno točko, ki jo bomo razširili v temno regijo;
2. začetna temna regija je inicializirana tako, da zaobjema izbrano točko (na sliki je prikazana kot bela pika, ki je še dodatno označena s puščico);
3. začetno regijo razširimo vertikalno navzgor in navzdol, dokler ne zadenemo svetle točke (na sliki je prikazan primer, ko zadenemo meji števnik);
4. regijo razširimo v levo in desno za eno točko. Opazimo, da smo jo morali na spodnjem koncu skrčiti po višini, ker se je razširjala v črno zarezo, ki predstavlja decimalno vejico;
5. nadaljnja širitev v levo in desno ne zahteva zmanjšanja višine;
6. nadaljnja širitev je dosegla števki na obeh straneh (ni nujno, da se vedno zgodi istočasno);
7. ob nadaljnji širitvi regije v levo se mora njena površina zmanjšati, ker jo moramo drastično skrčiti po višini, da ne prekrijemo delov obeh števk na levi strani;
8. zaradi zmanjšanja površine je bila zadnja razširitev preklicana in rezultat je regija, ki smo jo dobili v koraku 6.

Rezultat pravkar opisanih postopkov nad celotno sliko si lahko ogledamo na sliki 30, kjer so dobljena temna področja pobarvana s svetlo barvo.



**Slika 30: Temna področja na podvzorčeni sliki**

Opazimo, da smo na seznam temnih področij uvrstili vsa temna področja, ki pripadajo števniku (se nahajajo med števkami), t.j. tista področja, ki jih potrebujemo za njegovo detekcijo, poleg teh pa vsebuje seznam še polno nezaželenih področij, ki jih moramo v nadaljevanju odpraviti.

Postopek izločanja poteka tako, da za vsako področje izračunamo razmerje med njegovo višino in širino. Če je to razmerje večje od vnaprej predpisane vrednosti (področje je previsoko glede na svojo širino) ali manjše od druge vrednosti (področje je preširoko glede na svojo višino), ga zavržemo. To pravilo temelji na dejstvu, da področja, ki se nahajajo med števki, ne morejo biti neomejeno raztegnjena, saj so horizontalno omejena s števki, vertikalno pa z mejami števnik, v kar se lahko prepričamo z analizo slike 30. Vidimo, da je večina ostalih področij precej bolj raztegnjenih (v primeru na sliki 30 je večina od njih precej širokih v primerjavi njihovo višino).

Tabela 4 prikazuje nekaj temnih področij števnik, ki smo jih dobili z zajemom slike pri različnih povečavah (označenih od A do D) in rotacijah števnik. Podatki so znotraj posamezne povečave urejeni po naraščajočem razmerju V/Š.

Povečava	Št. rotacije	Širina področja	Višina področja	Površina področja	Razmerje Š/V	Razmerje V/Š
A	1	8	13	104	0,62	1,63
A	2	8	14	112	0,57	1,75
A	3	7	13	91	0,54	1,86
A	4	7	14	98	0,50	2,00
A	5	7	15	105	0,47	2,14
A	6	6	14	84	0,43	2,33
A	7	6	15	90	0,40	2,50
A	8	5	14	70	0,36	2,80
A	9	5	15	75	0,33	3,00
A	10	4	14	56	0,29	3,50
B	1	9	15	135	0,60	1,67
B	2	9	16	144	0,56	1,78
B	3	8	15	120	0,53	1,88
B	4	8	16	128	0,50	2,00
B	5	8	17	136	0,47	2,13
B	6	7	16	112	0,44	2,29
B	7	7	17	119	0,41	2,43
B	8	6	16	96	0,38	2,67
B	9	6	17	102	0,35	2,83
B	10	5	17	85	0,29	3,40
B	11	4	16	64	0,25	4,00
C	1	10	19	190	0,53	1,90
C	2	10	20	200	0,50	2,00
C	3	9	19	171	0,47	2,11
C	4	9	20	180	0,45	2,22
C	5	8	20	160	0,40	2,50
C	6	7	20	140	0,35	2,86
C	7	6	20	120	0,30	3,33
C	8	5	20	100	0,25	4,00
D	1	10	19	190	0,53	1,90
D	2	10	20	200	0,50	2,00
D	3	9	19	171	0,47	2,11
D	4	9	20	180	0,45	2,22
D	5	8	20	160	0,40	2,50
D	6	7	20	140	0,35	2,86
D	7	6	20	120	0,30	3,33
D	8	5	20	100	0,25	4,00

**Tabela 4: Temna področja ter njihovo razmerje med višino in širino pri različnih povečavah in kotih zasuka števnik na zajeti sliki**

S pregledom zadnjega stolpca tabele 4 se lahko prepričamo, da se razmerje med višino in širino temnih področij nahaja nekje med 1·6 in 4·0 ter je relativno invariantno na povečavo, s katero je števnik zajet. Tabela 5 in graf na sliki 31 nam prikazujeta povzetek teh podatkov.

Številka povečave	Minimalno razmerje Š/V	Maksimalno razmerje Š/V	Minimalno razmerje V/Š	Maksimalno razmerje V/Š
A	0,29	0,62	1,63	3,50
B	0,25	0,60	1,67	4,00
C	0,25	0,53	1,90	4,00
D	0,25	0,53	1,90	4,00
Vse	0,25	0,62	1,63	4,00

**Tabela 5: Povzetek minimalnih in maksimalnih vrednosti razmerja med višino in širino temnih področij števnika**



**Slika 31: Grafični prikaz razmerij med višino in širino temnih področij števnika po posameznih povečavah**

Eksperimentiranje z večjo množico vzorcev nam pokaže, da lahko vsa temna področja binarizirane podvzorčene slike izločimo kot neustrezna oz. tista, ki ne pripadajo števniku, če se njihovo razmerje med višino in širino ne nahaja v intervalu od 1·1 do 4·3. S tem izločimo večino odvečnih področij, ne pa vseh.

Preostala področja med seboj primerjamo po velikosti in njihovih medsebojnih legah na sliki ter tako ugotovimo, katera pripadajo števniku in katera so za nas moteča. Postopek je naslednji.

***Postopek izločanja nezaželenih temnih področij***

1. seznam temnih področij razvrstimo po višinah temnih področij od najvišjega do najnižjega;
2. če je število področij na seznamu manjše od vnaprej predpisane vrednosti (4), postopek prekinemo in ugotovimo, da na sliki ni števnik;
3. inicializiramo seznam "kandidati" ter nanj prenašamo temna področja po vrsti od začetka, dokler ne naletimo na tako, ki se po višini razlikuje od največjega več kot za vnaprej predpisano vrednost;
4. če je število področij na seznamu "kandidati" manjše od vnaprej predpisane vrednosti (enake kot v koraku 2), ga zavržemo in se vrnemo nazaj na korak 2;
5. seznam "kandidati" razvrstimo po horizontalni koordinati centrov področij od najmanjše do največje;
6. inicializiramo seznam "temna področja števnik". Prvo področje na seznamu "kandidati" vpišemo v spremenljivko "trenutni" ter ga uvrstimo na seznam "temna področja števnik" in izbrišemo s seznama "kandidati";
7. naslednje področje na seznamu "kandidati" vpišemo v spremenljivko "naslednji" in ga izbrišemo s seznama "kandidati", če obstaja, sicer nadaljujemo s korakom 10;
8. če se vertikalna koordinata področja "naslednji" razlikuje po vrednosti od vertikalne koordinate področja "trenutni" za manj kot tretjino višine področja "trenutni", področje uvrstimo na seznam "temna področja števnik", sicer nadaljujemo na koraku 10;
9. spremenljivko "naslednji" prepisemo v spremenljivko "trenutni" in se vrnemo na korak 7;
10. če je število področij na seznamu "temna področja števnik" manjše od vnaprej predpisane vrednosti (enake kot v koraku 2), ga zavržemo in se vrnemo na korak 6 v primeru, da seznam "kandidati" ni prazen, v nasprotnem primeru postopek prekinemo.

Komentar postopka:

- v koraku 2 predpisano najmanjše število temnih regij na seznamu izhaja iz zahteve po treh v celoti vidnih števkih na sliki (poglavje 7.2), kar pomeni dve področji med njimi ter še po eno na vsaki strani;
- v koraku 3 zgrajen seznam "kandidati" predstavlja tista temna področja, ki imajo dovolj podobne višine, da bi lahko pripadali istemu števniku. S pogledom na sliko 30 se lahko prepričamo, da imajo vsa temna področja števnikova skoraj enako višino;
- v koraku 4 zavržemo seznam "kandidati", če ne zadosti istemu pogoju kot seznam temnih področij v koraku 2, ker ne vsebuje področij števnikova, ki bi zadostil zahtevanim pogojem (poglavje 7.2);
- nadaljnji koraki skušajo ugotoviti, ali so temna področja krajevno ustrezno razporejena po sliki, da bi lahko pripadala istemu števniku.

Na seznamu "temna področja števnikova" se nahajajo temna področja med števki razvrščena od leve proti desni. Rezultat si lahko ogledamo na sliki 32, kjer so temna področja števnikova označena s sivo barvo.



**Slika 32: Temna področja števnikova**



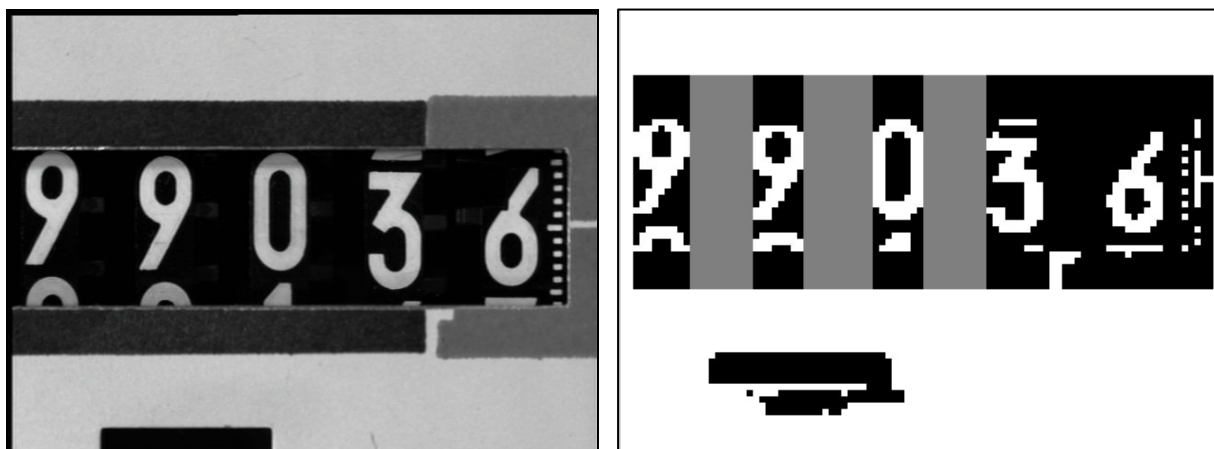
V primeru, da bi bil rezultat postopka prazen seznam, bi to pomenilo, da se na sliki števniki ne nahaja. Če bi se katerokoli temno področje dotikalo zgornjega ali spodnjega roba slike, bi vzorec razglasili za napačen, tako kot zahtevajo specifikacije v poglavju 7.2.

S tem, ko smo dobili temna področja števnik, je njegova detekcija končana ter grobo določena tudi njegova lokacija.

## 7.4 Omejitve opisanega postopka

Pravkar opisani postopek temelji na določenem številu predpostavk, ki morajo biti izpolnjene, sicer rezultati ne bodo uporabni. Njegov namen ni predvideti vseh mogočih situacij, predvsem ne takih, ki se ob načinu uporabe, za katerega je namenjen, ne morejo pojaviti. Npr. ker postopek temelji izključno na iskanju temnih področij med števki, ne pregleduje samih števk (tako kot smo zahtevali v poglavju 7.2), zato pa bo detektiral števec, ki ima števke prelepljene z nalepkami, kot da ni nič narobe – situacija, ki se v normalnih okoliščinah ne more pojaviti. Ravno tako bi ga lahko zmedli, če bi na belo površino postavili tri ali več manjših pravokotnih temnih površin ter jih razporedili v vrsto. Te omejitve niso resne, ker se ob smotrnem načinu uporabe ne morejo pojaviti.

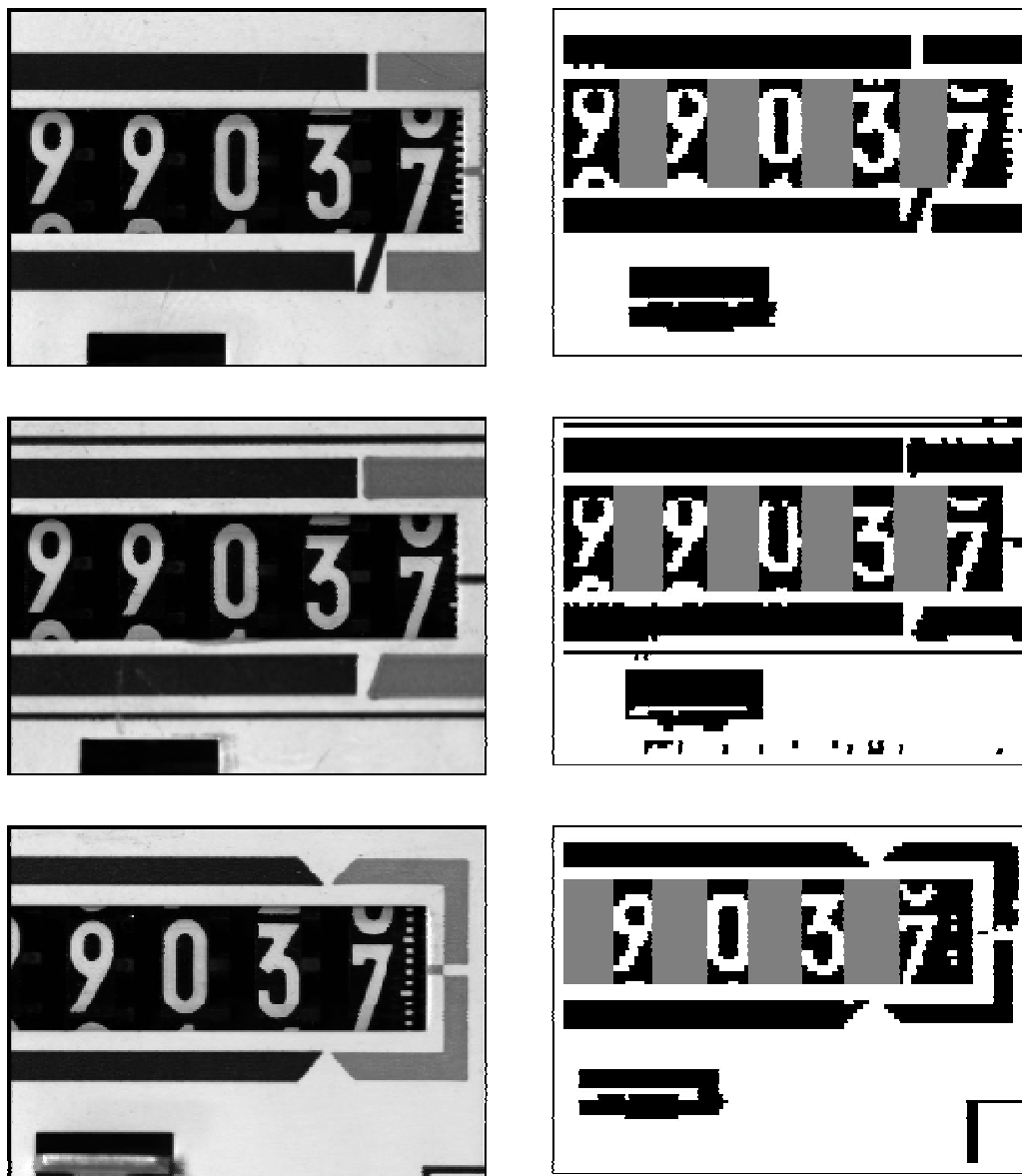
Resnejša omejitev je zahteva, da je števniki dobro ločen od ostalega dela čelne plošče z belo površino, česar predvsem nekateri starejši števcji ne izpolnjujejo. Tak primer si lahko ogledamo na sliki 33 levo, kjer vidimo, da se decimalna vejica, ki je narisana enostavno tako, da je črna barva, s katero je narisana rob števnik, prekinjena. Rezultat razvrščanja in izločanja temnih regij si lahko ogledamo na sliki 33 desno, kjer vidimo, da eno temno področje manjka, kar ima za posledico, da nadaljnji postopki analize ne uspejo zaznati zadnje števke (poglavje 1).



**Slika 33: Primer čelne plošče, kjer postopek odpove (levo) ter rezultat razvrščanja in izločanja temnih področij števnik (desno)**

Opazimo, da se temna področja po višini ne končajo na istih mestih kot številke, kar pa ne moti, kot bomo videli v nadaljnjih poglavjih.

Sklepamo lahko, da postopek uspešno detektira številnik ter pravilno prepozna vse njegove temne regije, če je le-ta z belim robom ločen od ostalega dela čelne plošče, kar si lahko ogledamo na sliki 34. V prvem primeru vidimo, da ne moti niti črna decimalna vejica, ki se dotika števnik. Moti samo bela, ki se razširja v notranjost črnih področij kot na sliki 33.



Slika 34: Primeri čelnih plošč, ki jih opisani postopek pravilno detektira



## 8 Lociranje števk

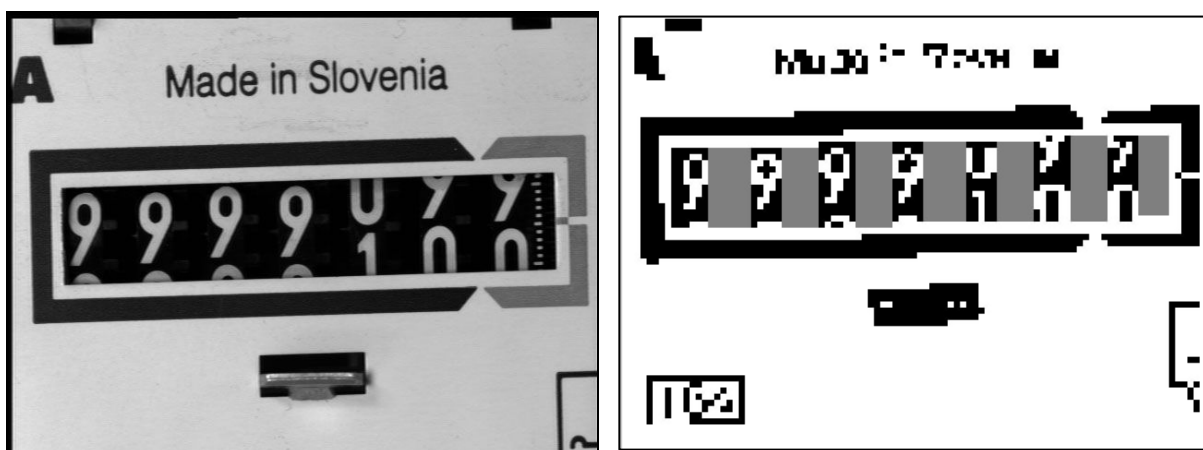
V prejšnjem poglavju smo detektirali števnik ter mu v grobem določili položaj na slikovnem vzorcu. Kot rezultat detekcije dobimo seznam temnih področij, ki se nahajajo med števki. Ta področja v nadaljevanju uporabimo za določitev regij, kjer se nahajajo številke. To pa ni edina naloga, ki jo moramo na tem mestu opraviti. Da bi uspešno izvedli poravnavo števk za primerjavo s trakom (poglavje 1), moramo natančno poznati višino števnika na sliki ter njegovo rotacijo glede na horizontalno os. Z območij, kjer se nahajajo številke, moramo tudi odstraniti morebitne robove števnika, da jih med poravnavo ne bi pripisali samim števkom.

V poglavju 8.1 bomo natančno določili vertikalne meje števnika (parametre premic, ki števnik obdajata na zgornjem in spodnjem robu), s čimer pridobimo tudi informacijo o njegovi višini ter rotaciji. Sledi detekcija prisotnosti številke pred prvo in za zadnjo temno regijo (poglavje 8.3), ker na teh mestih ne moremo vnaprej predpostaviti njunih prisotnosti oz. odsotnosti. V poglavju 8.4 tvorimo seznam regij s števki. Nazadnje bomo še opisali, kako s področij števk odstranimo moteče robove števnika (poglavje 8.5).

### 8.1 Vertikalni meji števnika

Vertikalni meji števnika sta fiktivni premici, ki števnik na zgornjem in spodnjem robu ločujeta od ostalega dela čelne plošče. Ti premici moramo sedaj poiskati. Postopek temelji na določitvi večjega števila robnih točk števnika ter iskanja regresijske premice skozi njih. Seznam temnih področij števnika bomo uporabili kot osnovni

napotek, kje iskati te točke, saj je v vmesnih prostorih med števki najlažje določiti, kje ima števec rob. Na sliki 35 levo vidimo primer slike števca, kjer si lahko brez težav predstavljamo, kje obe vertikalni meji števca potekata. Na desni strani slike 35 pa lahko opazujemo pripadajoča temna področja števca.

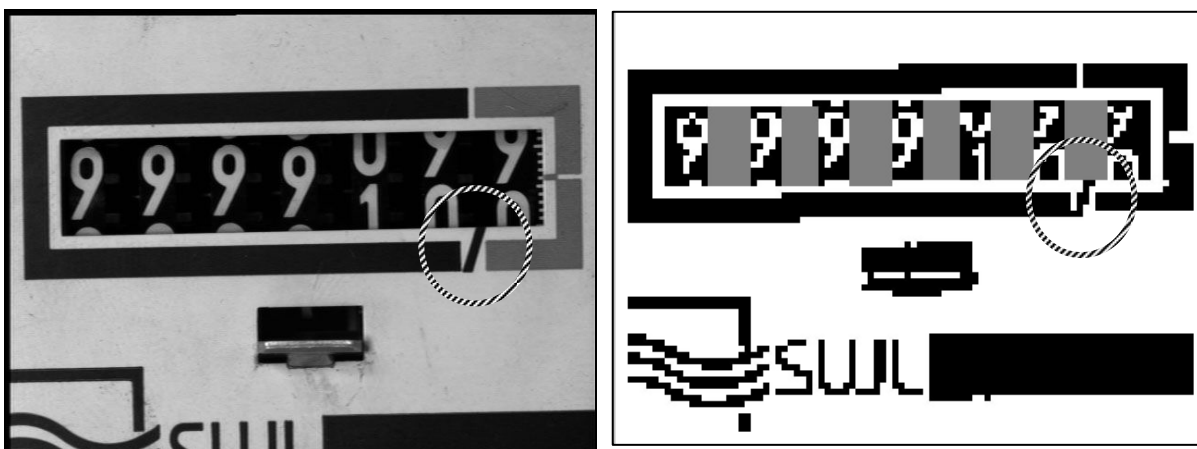


**Slika 35: Števec z lepimi predstavljivimi vertikalnimi mejami (levo) ter pripadajočimi temnimi področji med števki (desno)**

Vidimo, da so temna področja med števki dober napotek za iskanje točk, ki se nahajajo na vertikalnih mejah števca, saj te točke ležijo v bližini njihovega zgornjega in spodnjega roba. Področja, kjer se nahajajo števke, za detekcijo mej niso primerna, saj s preprostimi postopki preiskovanja slike ne moremo vedeti, kje se številka konča in kje se prične meja števca v primeru, ko se števke nahajajo v takih položajih, da med njimi in robom števca ni nobenega vmesnega prostora (na sliki 35 to velja vsaj za zadnje tri števke na zgornjem robu in za vse na spodnjem).

### **8.1.1 Izločanje neprimernih temnih področij za iskanje mej**

Kljub temu, da so temna področja skoraj idealna za iskanje mej, nam majhen problem predstavljajo čelne plošče, kjer je decimalno mesto od ostalih ločeno z decimalno vejico, ki je vrisana s črno barvo prav do samega roba števca, kar lahko vidimo na sliki 36 levo. Pripadajoča temna področja so prikazana na isti sliki desno.



**Slika 36: Primer števnik z motečo decimalno vejico (levo) ter pripadajoča temna področja (desno)**

Opazimo, da je spodnji rob skrajno desnega temnega področja (na slikah poudarjen s krožnico) neprimeren za iskanje meje števnik, saj na mestu, kjer se nahaja decimalna vejica, ne moremo sklepati na njen potek z enostavnimi postopki preiskovanja slike.

Temne regije moramo torej najprej pregledati, če so ustrezne za iskanje robnih točk. Preiskujemo samo njihov spodnji rob, saj se decimalna vejica nikoli ne pojavi na zgornji strani števnik. Testiranje poteka na naslednji način.

***Postopek testiranja ustreznosti temne regije***

1. od spodnjega levega oglišča temne regije preiskujemo binarizirano podvzorčeno sliko vertikalno navzdol, dokler ne naletimo na svetlo točko (v veliki večini primerov je to kar prva točka, ki se nahaja neposredno pod ogliščno točko temne regije);
2. enako storimo tudi pod spodnjim desnim ogliščem;
3. zapomnimo si tisto od obeh dobljenih točk, ki je bolj oddaljena od zgornjega roba slike;
4. pri vseh ostalih točkah, ki pripadajo spodnjemu robu temne regije, postopamo enako;

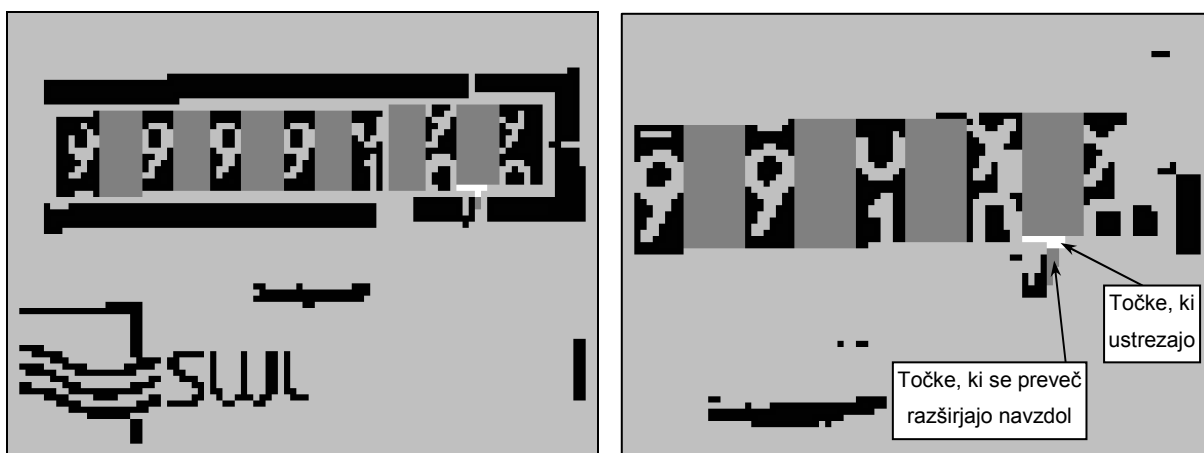
5. temna regija ni primerna za iskanje robnih točk, če je katerakoli v koraku 4 dobljena točka za več kot en slikovni element bolj oddaljena od zgornjega roba slike kot točka zapomnjena v koraku 3. V nasprotnem primeru je regija primerna.

#### Komentar postopka:

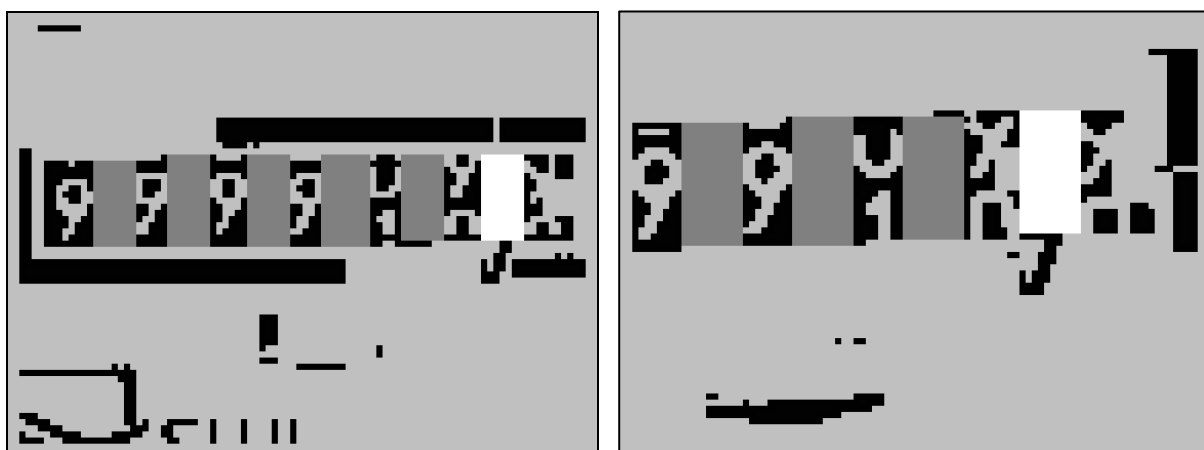
- v korakih 1 in 2 smo s preiskovanjem binarizirane podvzorčene slike določili dve točki, skozi kateri poteka grobi približek premice, ki predstavlja vertikalno spodnjo mejo števnik;
- v koraku 3 določimo vodoravno premico, ki se nahaja pod oz. se dotika premice, dobljene v korakih 1 in 2 v celotnem področju temne regije;
- v korakih 4 in 5 testiramo, ali se pod spodnjim robom temne regije nahaja moteča decimalna zareza tako, da ugotavljamo, ali se na kateremkoli mestu temna regija števnik razširja v območje pod vodoravno premico;
- v koraku 5 dopuščamo toleranco ene točke (testiramo, če je katerakoli dobljena točka oddaljena več kot za en slikovni element od zgornje meje) zaradi šuma binarizacije, ki je vedno prisoten pri tako enostavnem postopku binarizacije, kot ga mi izvajamo (poglavje 6.2).

Na slikah 37 in 38 si lahko ogledamo rezultat preiskovanja temnih regij, kjer smo neustrezno tudi uspešno detektirali. Na sliki 37 smo z belo in pod njimi s sivo barvo označili preiskane točke. Prisotnost samo belih točk pomeni, da je regija ustrezna, vsaj ena siva točka pod belimi pa je dovolj, da regijo izločimo kot neustrezno. Izločeno temno regijo si lahko ogledamo na sliki 38, kjer je prikazana z belo barvo. Na isti sliki se lahko prepričamo tudi, da so vse preostale regije primerne za določanje točk na vertikalnih mejah števnik.





Slika 37: Izločanje neustreznih temnih regij za določanje mej števnik (levo manjša, desno večja povečava)



Slika 38: Izločena neustrezna temna regija za določanje mej števnik (levo manjša, desno večja povečava)

### 8.1.2 Iskanje točk na vertikalnih mejah števnik

Točke, ki se nahajajo na obeh mejah števnik, iščemo v bližini zgornjega in spodnjega roba temnih regij, za katere smo v predhodnem poglavju ugotovili, da so primerne. Postopek je tak, da te regije najprej skrčimo za dva slikovna elementa na vsaki strani. To storimo zato, da ne dobimo prevelikega števila točk (horizontalna skrčitev) in da pričnemo preiskovanje zanesljivo znotraj števnik (vertikalna skrčitev).

Točk na mejah števnikar ne iščemo več na podvzorčeni sliki, ampak na originalni, zato moramo temne regije ustrezno preračunati na nove koordinate (zgornjo levo točko pomnožimo z osem, spodnjo desno pa pomnožimo z osem ter prištejemo sedem). Skrčena in preračunana temna področja so prikazana na sliki 39 kot beli pravokotniki z vrisanimi diagonalnimi križi.



**Slika 39: Področja, kjer bomo iskali točke na mejah števnikar (levo manjša, desno večja povečava)**

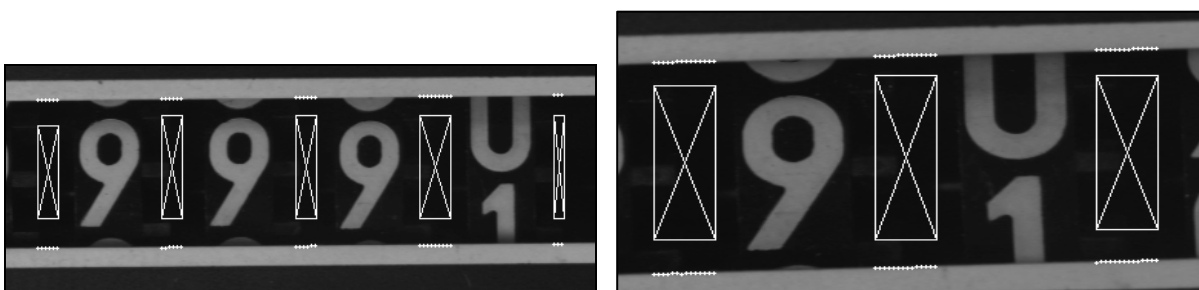
Preiskovanje se prične v zgornji levi točki prve regije in poteka navzgor, dokler ne dosežemo slikovnega elementa, ki ima večjo osvetljenost od praga binarizacije, ali dokler ne dosežemo roba slike (v tem primeru postopek prekinemo, ker se števec dotika roba slike; poglavje 7.2). Ker originalne slike nismo filtrirali, se moramo prepričati, da dobljena točka res pripada meji števnikar in ni posledica šuma. V ta namen preiskovanje nadaljujemo še določeno število točk (v našem primeru 4) naprej in če so vse svetlejše od praga binarizacije, točko razglasimo za mejno in jo uvrstimo na seznam točk, ki pripadajo zgornji meji števnikar. Postopek ponovimo za vsak tretji slikovni element (lahko bi za vsak ali vsak drugi, a izkaže se, da bi s tem dobili večje število točk, ki pa ne prispevajo dodatne informacijske vsebine, saj so premice dobljene z manjšim številom le-teh praktično identične), ki se nahaja na zgornjem robu skrčene temne regije. Nato pa enako še za spodnji rob, le da sliko

preiskujemo navzdol in točke uvrščamo na seznam točk, ki pripadajo spodnji meji števnik. To ponavljamo za vsako primerno temno regijo.

Rezultat postopka sta dva seznama točk (sliki 40 in 41), skozi katera lahko potegnemo regresijski premici, ki predstavljata vertikalni meji števnik (sliki 42 in 43).



**Slika 40: Točke, ki pripadajo vertikalnim mejam števnik (levo manjša, desno večja povečava)**



**Slika 41: Povečan prikaz točk, ki pripadajo vertikalnim mejam števnik (levo manjša, desno večja povečava)**



Slika 42: Prikaz vertikalnih mej števecva (levo manjša, desno večja povečava)



Slika 43: Povečan prikaz vertikalnih mej števecva (levo manjša, desno večja povečava)

## 8.2 Parametri števnika

V parametrični obliki določeni vertikalni meji števnika nam omogočata natančno določitev dveh njegovih parametrov, ki ju potrebujemo pri poravnavi števk (poglavje 1). To sta zasuk glede na horizontalno os ter povečava, pri kateri smo števnik zajeli (oz. parameter, s katerim določimo relativno razmerje med povečavami različnih zajemov).

Zasuk števnika glede na horizontalno os izračunamo kot povprečno vrednost zasukov obeh vertikalnih mej, povečavo pa kot razliko njunih svobodnih členov:

$$R_{\xi} = \frac{\arctg(k_1) + \arctg(k_2)}{2} \qquad P_{\xi} = n_2 - n_1$$

kjer pomenijo oznake:

- $R_{\xi}$  rotacija števnika (v radianih),
- $k_1, k_2$  smerna koeficienta zgornje in spodnje vertikalne meje,
- $P_{\xi}$  merilo povečave števnika,
- $n_1, n_2$  svobodna člena zgornje in spodnje vertikalne meje.

Povečava števnika se neposredno in linearno odraža v njegovi višini, t.j. isti števnik, ki je zajet pri dvakrat večji povečavi, bo na sliki zavzemal po višini dvakrat večje število slikovnih elementov. Ker se izhodišče koordinatnega sistema slikovnih elementov prične v zgornjem levem kotu slike, ima parameter  $n_1$  manjšo vrednost kot  $n_2$  in je merilo povečave vedno pozitivno.

### 8.3 Detekcija prve in zadnje števk

Natančno določeni vertikalni meji števnik sta predpogoj za določitev zgornjih in spodnjih mej območjem, na katerih se nahajajo številke. Brez njih bi bilo nemogoče ugotoviti, kje se številka konča ter prične rob števnik, zlasti v tistih položajih koleščka, kjer med njima ni nobenega temnega področja (zadnje tri številke na sliki 35, str. 72). S tem, ko smo uspeli določiti parametre fiktivnih premic, ki potekata po obeh vertikalnih mejah števnik, smo tudi določili, kje so vertikalni robovi območij s števkami.

Naslednje vprašanje, na katerega moramo odgovoriti, je, koliko števk je na sliki prisotnih ter kje se nahajajo horizontalno znotraj področja števnik. S pogledom na sliko 34, str 69 lahko ugotovimo, da se med vsakima temnima področjema števnik nahaja po ena številka, torej lahko prisotnost in približni položaj večine števk določimo že na podzorceni sliki. Na ta način pa ne moremo določiti, kaj se nahaja pred prvim in za zadnjim temnim področjem števnik. Izkaže se, da se lahko na teh mestih številke nahajajo ali pa ne. Nekaj možnih situacij prikazuje slika 44.



**Slika 44: Različne situacije pred prvo in za zadnje temno regijo števnik**

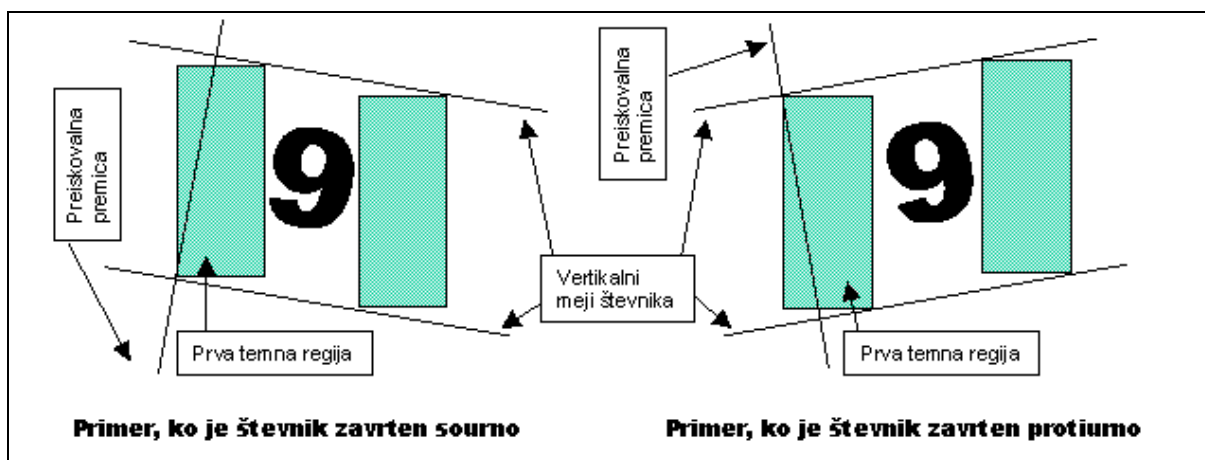
Zgornji levi primer na sliki 44 nam kaže situacijo, ko se pred prvim in za zadnjim temnim področjem nahaja dodatna številka. Zgornji desni primer je ravno nasproten;

obe strani sta brez dodatnih števk. Spodaj levo vidimo situacijo, ko se dodatna številka nahaja samo na desni strani. Spodaj desno pa je številnik zajet tako, da na desni strani ne smemo prepoznati dodatne številke, ker ni vidna v celoti.

Vidimo, da samo z analizo temnih področij ne moremo potrditi ali izključiti prisotnosti dodatnih števk. Odgovor moramo poiskati z analizo originalne slike na področju pred prvo in za zadnjo temno regijo. Postopamo na naslednji način.

***Postopek iskanja številke pred prvo temno regijo***

1. določimo preiskovalno premico, ki je pravokotna na vertikalni meji številnika ter poteka skozi tisto oglišče levega roba prve temne regije, ki zagotavlja, da se noben del premice v območju med mejama številnika ne nahaja levo od levega roba regije (slika 45);
2. v območju med obema vertikalnima mejama številnika preiskujemo originalno sliko vzdolž preiskovalne premice s ciljem, da ugotovimo, ali se na njej nahaja vnaprej določeno število zaporednih točk, ki so svetlejše od praga binarizacije;
3. če smo v prejšnjem koraku ugotovili, da tako zaporedje točk obstaja, razglasimo položaj preiskovalne premice za začetek področja, kjer se lahko številka nahaja, in nadaljujemo s korakom 6;
4. preiskovalno premico premaknemo nekaj točk (3) v levo;
5. če preiskovalna premica v območju med vertikalnima mejama številnika ne pade izven slike, se vrnemo na korak 2, sicer ugotovimo, da se pred prvo temno regijo številka ne nahaja in postopek končamo;
6. preiskovalno premico premaknemo za eno točko v levo;
7. če preiskovalna premica v območju med vertikalnima mejama številnika pade vsaj delno izven slike, smatramo, da številka pred prvo regijo ni prisotna ali vidna v celoti, ter postopek prekinemo;
8. vzdolž preiskovalne premice iščemo zaporedje vnaprej določenega števila točk, ki so svetlejše od praga binarizacije;
9. če tako zaporedje obstaja, se vrnemo na korak 6, sicer smo pred prvim temnim področjem našli številko, ki je v celoti vidna.



**Slika 45: Princip določanja preiskovalnih premic**

Komentar postopka:

- vse temne regije so preračunane na velikost originalne slike;
- v koraku 2 in 8 iščemo zaporedje točk, ki so svetlejše od praga binarizacije, in ne ene same. Razlog je v tem, da originalna slika ni filtrirana in se na tak način izogibamo vplivu šuma binarizacije;
- v koraku 4 premikamo preiskovalno premico po več točk naenkrat zaradi pohitritve izvajanja. Ker iščemo začetek morebitno prisotne števk in je najmanjša širina le-te cca. 30 točk, lahko to počnemo brez bojazni, da bi števko preskočili;
- če smo pri izvajanju postopka dosegli korak 6, pomeni, da smo dobili možni začetek števk. Od tu naprej iščemo njen konec, zato moramo preiskovalno premico premikati po eno točko v levo, da pravilno detektiramo tudi števke, ki se nahajajo zelo blizu levega roba slike, t.j. so od njega oddaljene samo en slikovni element;
- pri tem postopku predpostavljamo (in s tem delno kršimo pravilo 7 v poglavju 7.2), da noben simbol števk ni po vertikali pretrgan (t.j. ni sestavljen iz dveh po vertikali razmejenih delov).

Analogno preiskujemo originalno sliko še za zadnjim temnim področjem števnik. Ko je postopek končan, poznamo lokacije vseh področij, kjer se števke nahajajo.



## 8.4 Tvorjenje seznama območij s števki

Območja števk, ki smo jih predhodno detektirali, hranimo kot seznam pravokotnih področij originalne slike. Področja niso poljubno orientirana, saj zahtevamo, da potekajo njihove stranice vzporedno oz. pravokotno z robovi slike (slika 46).



**Slika 46: Območja števk pri različnih rotacijah števnik (od leve proti desni: vodoravni položaj, zasuk protiurno, zasuk sournjo)**

Na sliki 46 vidimo, da se pravokotna področja ne nahajajo več v isti vodoravni liniji, ko je števnik zarotiran (srednji in desni primer), so pa še vedno vsa orientirana v isto smer, t.j. njihovi zgornji in spodnji robovi so vzporedni z zgornjim in spodnjim robom slike, njihovi levi in desni pa z levim in desnim robom slike. Razlog je v tem, da so postopki obdelave takih področij hitrejši in preprostejši.

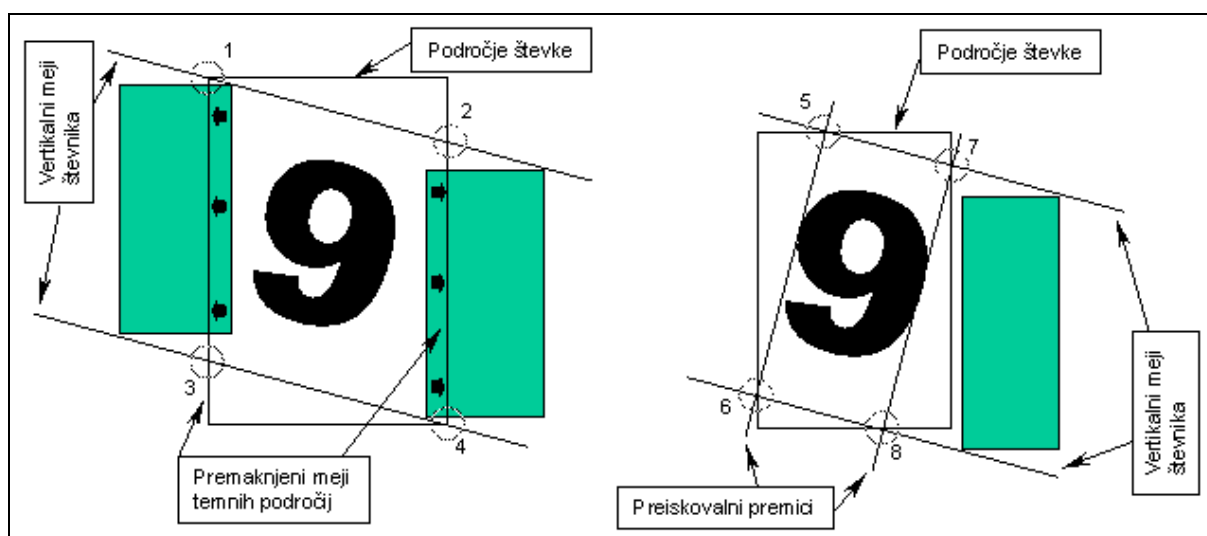
Postopek določanja področij števk je različen za števke med temnima področjema števnik in za tiste, ki se nahajajo pred prvo oz. za zadnjo temno regijo (slika 47).

**Števka se nahaja med temnima regijama števnik**

1. **levi rob** določimo tako, da desni rob predhodnega temnega področja premaknemo za osem slikovnih elementov v levo. Razlog je v tem, da so bila temna področja generirana z analizo podvzorčene slike, kar pomeni, da se lahko v najslabšem primeru na originalni sliki za toliko prekrivajo z območji števk;
2. **desni rob** določimo tako, da levi rob naslednjega temnega področja premaknemo v desno za osem slikovnih elementov;
3. **zgornji rob** poteka skozi eno od točk, kjer se levi ali desni rob števke seka z zgornjo vertikalno mejo števnik, in sicer tisto, ki je bližja zgornjemu robu slike (na sliki 47 se v točkah 1 in 2 sekata levi oz. desni rob z zgornjo mejo števnik; zgornja meja števke poteka skozi točko 1);
4. **spodnji rob** določimo povsem analogno z zgornjim; opazujemo spodnjo vertikalno mejo števnik in izberemo tisto točko, ki je bližja spodnjemu robu slike (na sliki 47 sta prikazani točki 3 in 4; rob števke poteka skozi točko 4).

**Števka se nahaja pred prvo oz. za zadnjo temno regijo števnik**

1. **levi rob** določimo tako, da poiščemo točki, v katerih se preiskovalna premica na začetku števke seka z vertikalnima mejama števnik; levi rob števke poteka skozi točko, ki je bližja levemu robu slike (na sliki 47 sta taki točki 5 in 6; levi rob poteka skozi točko 6);
2. **desni rob** lociramo z analognim postopkom. Določimo točki, v katerih se preiskovalna premica na koncu števke seka z vertikalnima mejama števnik; desni rob števke poteka skozi točko, ki je bližja desnemu robu slike (na sliki 47 sta taki točki 7 in 8; desni rob poteka skozi točko 7);
3. **zgornji rob** poteka skozi tisto ob obeh točk, v kateri levi in desni rob sekata zgornjo vertikalno mejo števnik, ki je bližja zgornjemu robu slike (na sliki 47 sta taki točki 5 in 7; zgornji rob poteka skozi točko 5);
4. **spodnji rob** poteka skozi tisto od obeh točk, v kateri levi in desni rob sekata spodnjo vertikalno mejo števnik, ki je bližja spodnjemu robu slike (na sliki 47 sta taki točki 6 in 8; spodnji rob poteka skozi točko 8).

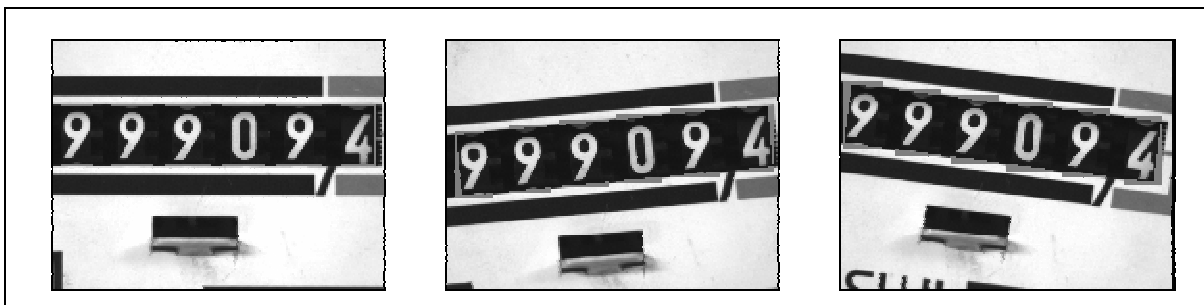


**Slika 47: Princip določanja področij s števki (levo: števka se nahaja med temnima področjema števnik, desno: števka se nahaja pred prvim temnim področjem)**

Postopek izvedemo za vsako števko, ki smo jo predhodno detektirali. Rezultat je seznam pravokotnih področij na sliki, kjer se številke nahajajo. Prikazana so na sliki 46.

## 8.5 Odstranjevanje motečih robov števnik

Težava, ki se pojavi zaradi zahteve po vzporednosti stranic področij s slikovnimi, je, da področja števk vsebujejo tudi robove števnik. Na sliki 48 smo s sivo barvo prikazali območja, kjer se nezaželen efekt lahko pojavi. Zaradi slabe vidljivosti (območja so majhna) smo na sliki 49 nekatere številke prikazali povečano.



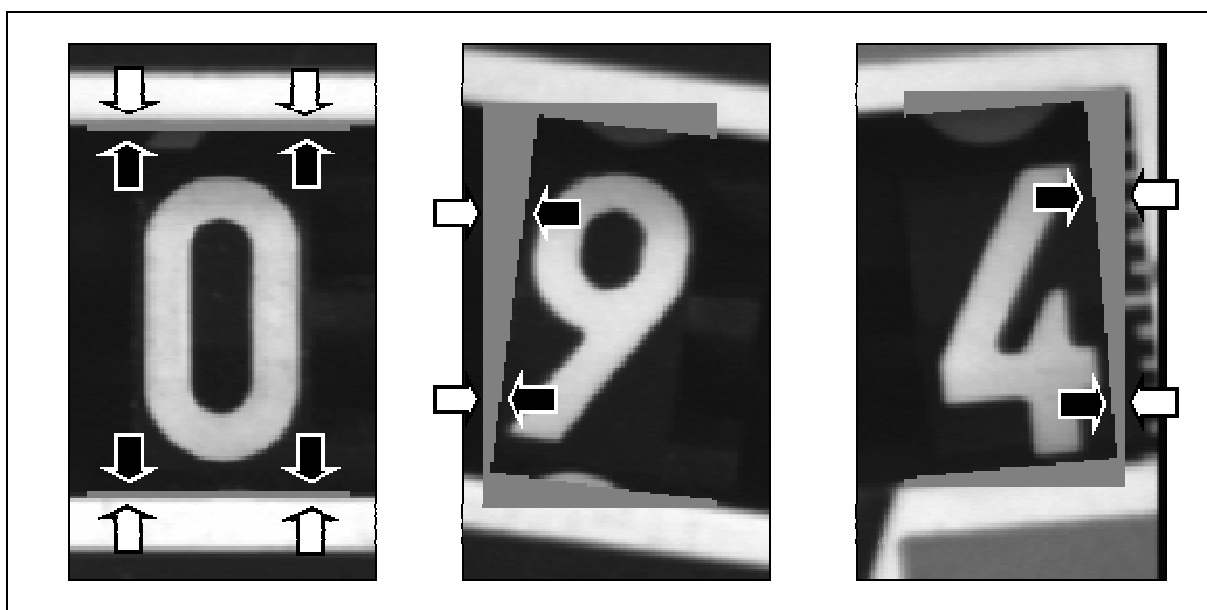
**Slika 48: Moteči robovi števnik**



**Slika 49: Povečan prikaz motečih robov števnik**

Vidimo, da se pri števkih, ki se nahajajo znotraj temnih področij števnik, motnja pojavi samo na zgornjem in spodnjem robu, pri skrajno levi in desni števk pa se lahko pojavi tudi na levi oz. desni strani (slika 50). Te motnje moramo odstraniti, da nam pozneje ne bodo kvarile poteka primerjave s trakom (poglavje 11.1).

Vse, kar moramo storiti, je z originalne slike pobrisati trikotna področja med zgornjo mejo števnik in zgornjo mejo območij posameznih števk. Podobno na spodnji strani pobrišemo trikotna območja med spodnjo mejo števnik in spodnjo mejo območij, kjer se nahajajo števk (na slika 50 levo sta ti področji označeni s puščicami). Števk, ki se nahaja pred prvo temno regijo števnik (če obstaja), moramo še dodatno pobrisati trikotno območje med levo mejo regije in zadnjo preiskovalno premico (slika 50 v sredini). Podobno tudi števk za zadnjo temno regijo (slika 50 desno) pobrišemo območje med zadnjo preiskovalno premico in desnim robom njenega območja. Rezultat si lahko ogledamo na sliki 51.



**Slika 50: Moteči robovi števnik (levo: številka se nahaja med temnima področjema števnik, sredina in desno: številka se nahaja pred prvim oz. za zadnjim temnim področjem)**



**Slika 51: Odstranjeni moteči robovi števnik**

S tem je postopek lociranja števk končan. Številke, ki so na sliki vidne v celoti, smo popisali z nizom območij, kjer se le-te nahajajo, ter s teh območij odstranili vse elemente števnik, ki bi s števkami lahko interferirali v postopku poravnave (poglavje 1).



## 9 Poravnava števk

V prejšnjem poglavju smo števkom določili položaje na sliki z namenom, da jim bomo v nadaljevanju pripisali numerične vrednosti. Postopek, ki je bil nakazan v poglavju 4.3, temelji na primerjavi bitnih kart števk s sliko raztegnjenega koleščka števnik. Direktna primerjava pa je možna le v primeru, da so števke za izgradnjo traku (poglavje 1) zajete pod enakimi pogoji, kot se vrši odčitavanje vrednosti števnik. To je v nekaterih primerih možno zagotoviti (npr. zajem za izgradnjo traku in tudi vsi zajemi za odčitavanje vrednosti so opravljeni s števci, pritrjenimi na istem togem nosilcu). V splošnem pa ocenjujemo, da je taka zahteva moteča, nepraktična in bi skrčila možne načine uporabe sistema (npr. ročni odčitovalnik bi bil povsem neuporaben, saj od operaterja ne moremo pričakovati, da bo napravo za zajem poravnal s prosto roko na manj kot eno kotno stopinjo natančno). Problem rešujemo tako, da števke poravnamo, preden jim določimo vrednosti; njihove slike skušamo prilagoditi (normirati) tako, da bi bile čim bolj podobne slikam, ki bi jih zajeli pod vnaprej določenimi pogoji.

V tem poglavju se bomo seznanili s problematiko poravnave števk (poglavje 9.1), predstavili naš pristop k njenemu reševanju (poglavje 9.2) ter implementirani postopek tudi podrobno opisali (poglavje 9.3). Nazadnje bomo v poglavju 9.4 prikazali, na kakšen način poravnavo uporabljamo.

## 9.1 Problematika poravnave števk

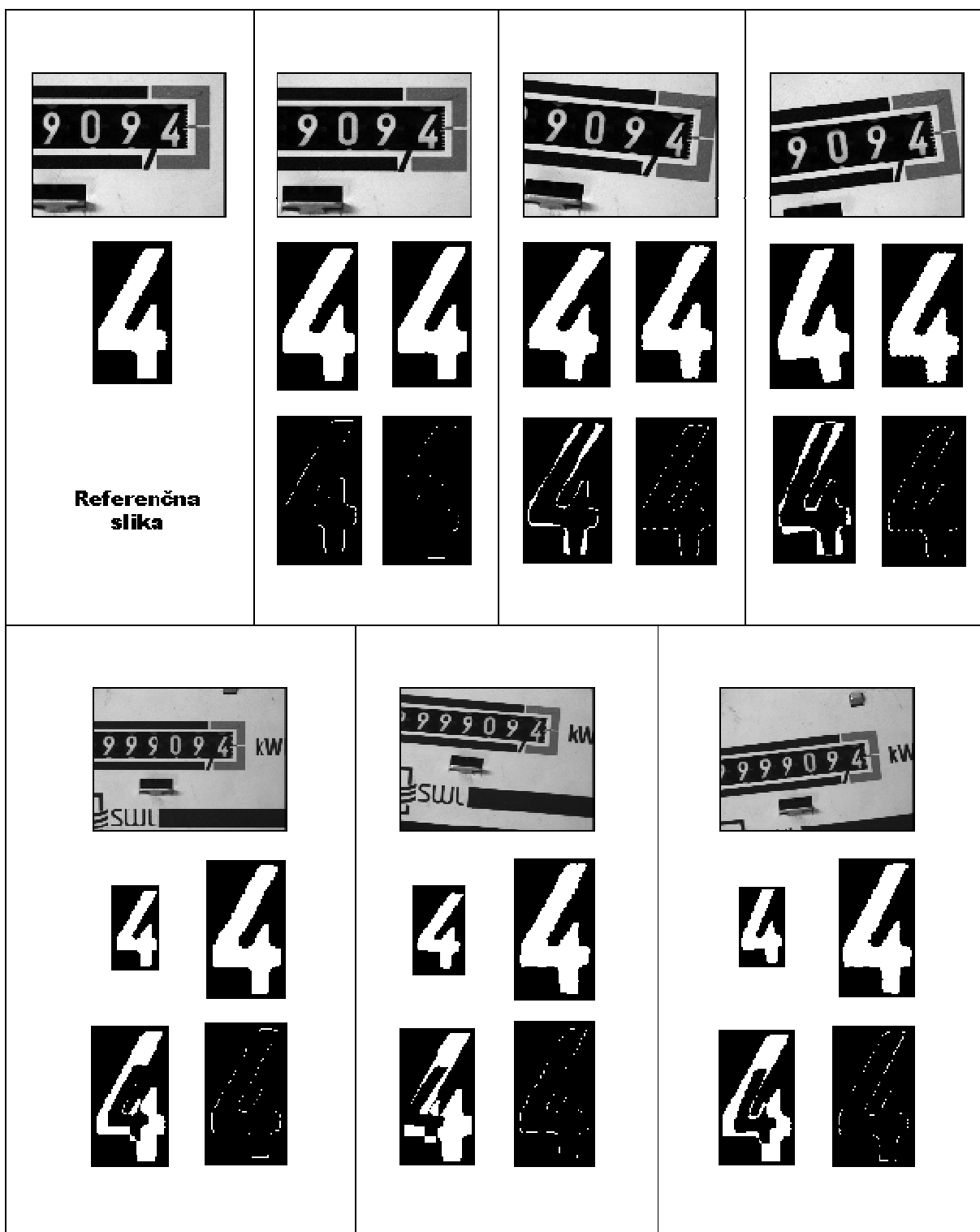
V zadnjih treh poglavjih smo opisali postopke analize slikovnih vzorcev, ki smo jih namenoma zasnovali tako, da delujejo v širokem območju pogojev, pod katerimi lahko slike števecv zajamemo. Tako smo s postopki začetne analize slike (poglavje 1) uspeli delno odpraviti vplive, katerim so slike podvržene zaradi sprememb osvetlitve števecv pri različnih zajemih. Postopka detekcije števnika (poglavje 1) in lociranja števk (poglavje 1) sta zasnovana tako, da nalogo opravita kljub različnim povečavam in rotacijam števca na sliki. Neobčutljivost na te parametre želimo ohraniti skozi celoten postopek odčitavanja vrednosti.

Pripomnimo naj, da je neobčutljivost na spremenjeno osvetlitev vzorca zagotovljena s pravilno postavitvijo praga binarizacije skozi vse faze postopka; občutljivosti na rotacijo in povečavo pa nismo mogli odpraviti že s tem, da smo detekcijo zasnovali neobčutljivo nanju. To nalogo moramo rešiti s postopkom poravnave števk.

Moteči vpliv rotacije in spremenjene povečave števnika je prikazan na sliki 52, kjer lahko opazujemo njun učinek na sliko zadnje (skrajno desne) številke števnika. Zgornji levi okvirček prikazuje referenčno sliko števca ter številko, s katero bomo primerjali ostale. Drugi primeri v zgornji vrsti so zajeti pri enaki povečavi in spremenjenem zasuku (od leve proti desni: zasuka ni, števec je zarotiran so- in protiurno). Spodnja vrsta prikazuje zajeme pri manjši povečavi ter z enakimi zasuki. Pod slikami števnikov sta prikazani neporavnana in poravnana (po postopku, ki ga opisujemo v nadaljevanju) slika zadnje številke, pod njima pa lahko vidimo izključujočo disjunkcijo (ang. exclusive or) med njima in referenčno številko.

Vidimo, da pride ob spremenjeni rotaciji in še posebej povečavi do znatnih odstopanj neporavnanih števk od referenčne. Postopek poravnave pa je v vseh primerih uspel njihove slike uskladiti, tako da lahko opazimo manjša odstopanja samo na robovih.





Slika 52: Vpliv rotacije in povečave števcia na slike števk ter učinek poravnave

## 9.2 Klasičen in naš pristop k poravnavi števka

Poravnavo števka bi lahko izvedli s klasičnimi postopki transformacije slikovnih vzorcev, t.j. s preprosto izvedbo rotacije in povečave slike števke za ustrezen kot in faktor povečave [4, 6]. Prednost takega pristopa bi bila preprosta in lahko razumljiva izvedba postopka, slabost pa večja časovna zahtevnost, saj na ta način ne bi mogli izkoristiti *a priori* znanja o vsebini slike. Vseeno si ta postopek oglejmo, da ga bomo lahko primerjali z našim.

### 9.2.1 Klasičen postopek transformacije slike

Klasična izvedba rotacije in povečave slike je taka, da najprej inicializiramo novo sliko ustrezne velikosti, ki bo vsebovala rezultat transformacije. Nato se za vsak slikovni element nove slike izračuna (inverzna transformacija k tisti, ki jo izvajamo), na katero mesto originalne slike se preslika. Ker se to mesto v splošnem ne ujema natančno z nobenim njenim slikovnim elementom (slikovni elementi imajo celoštevilске koordinate, rezultati transformacije pa imajo tudi decimalni del), vrednost slikovnega elementa na novi sliki določimo z interpolacijo vrednosti večih slikovnih elementov originalne slike. Pri tem največkrat uporabljamo bilinearno interpolacijo (slika 53), v nekaterih primerih pa tudi interpolacije višjega reda. Podroben opis postopka najdemo v [1].

Iz povedanega sledi, da moramo za vsak slikovni element nove slike izračunati njegovo inverzno transformacijo (v primeru dvodimenzionalne rotacije in povečave to pomeni štiri množenja in dve seštevanji) po naslednji enačbi:

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} S_x \cos(\phi) & -S_y \sin(\phi) \\ S_x \sin(\phi) & S_y \cos(\phi) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

kjer pomenijo oznake:

- $x, y$  koordinati slikovnega elementa, ki ga transformiramo,
- $x_1, x_2$  transformirani koordinati slikovnega elementa,
- $\phi$  kot rotacije,
- $S_x, S_y$  faktorja povečave po x in y osi.

Ker sta rotacija in povečava konstantni za vse slikovne elemente, ki jih transformiramo, je transformacijska matrika konstantna in jo lahko izračunamo pred začetkom transformacije.

Enačbe za bilinearno interpolacijo, ki jo moramo ravno tako izvesti za vsak transformiran slikovni element, so naslednje (ilustracija na sliki 53):

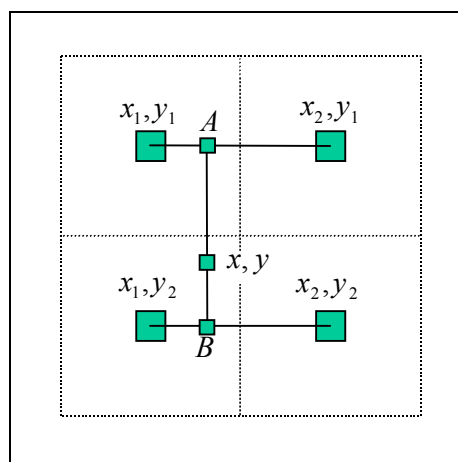
$$V(A) = \frac{V(x_2, y_1) - V(x_1, y_1)}{(x_2 - x_1)} \cdot (x - x_1) + V(x_1, y_1)$$

$$V(B) = \frac{V(x_2, y_2) - V(x_1, y_2)}{(x_2 - x_1)} \cdot (x - x_1) + V(x_1, y_2)$$

$$V(x, y) = \frac{V(B) - V(A)}{(y_2 - y_1)} \cdot (y - y_1)$$

kjer pomenijo oznake:

- $x, y$  koordinati točke, kateri določamo vrednost,
- $x_i, y_i; i = 1, 2$  koordinate slikovnih elementov, ki so točki najbližji,
- $V(A), V(B)$  vmesna rezultata interpolacije v eni smeri,
- $V(x, y)$  interpolirana vrednost slikovnega elementa, ki jo iščemo.



**Slika 53: Ilustracija bilinearne interpolacije**

To pomeni dodatnih devet seštevanj, tri deljenja in tri množenja. Za izvedbo izračunov moramo tudi prebrati vrednosti štirih slikovnih elementov originalne slike. Pripomnimo naj, da število slikovnih elementov originalne slike narašča kvadratno s povečavo (natančneje z višino števk na sliki), pri kateri števec zajemamo. Zaključimo lahko, da je klasičen postopek poravnave časovno potraten, zato smo razvili svojega, ki je specializiran za poravnavo števk in ga ni možno uporabiti za transformacijo splošnih slik.

### 9.2.2 Naš pristop k transformaciji slike števke

Postopek, ki smo ga razvili, osnovne slike ne transformira, pač pa neparavnano sliko števke uporabi kot navodilo za sintezo nove. Glavna pohitritev v primerjavi s predhodno opisanim postopkom je v tem, da transformiramo samo robove števk na sliki, medtem ko ostale dele njihovih površin zapolnimo s poplavljanjem področij. Manjša časovna zahtevnost takega pristopa izhaja iz tega, da večine slikovnih elementov na sliki ni potrebno transformirati (nobenega, ki se nahaja izven ali v notranjosti števke); poleg tega vrednosti na novi sliki ne računamo z bilinearno interpolacijo. Število robnih točk, ki jih transformiramo, pa narašča linearno s povečavo števca oz. višino števke.

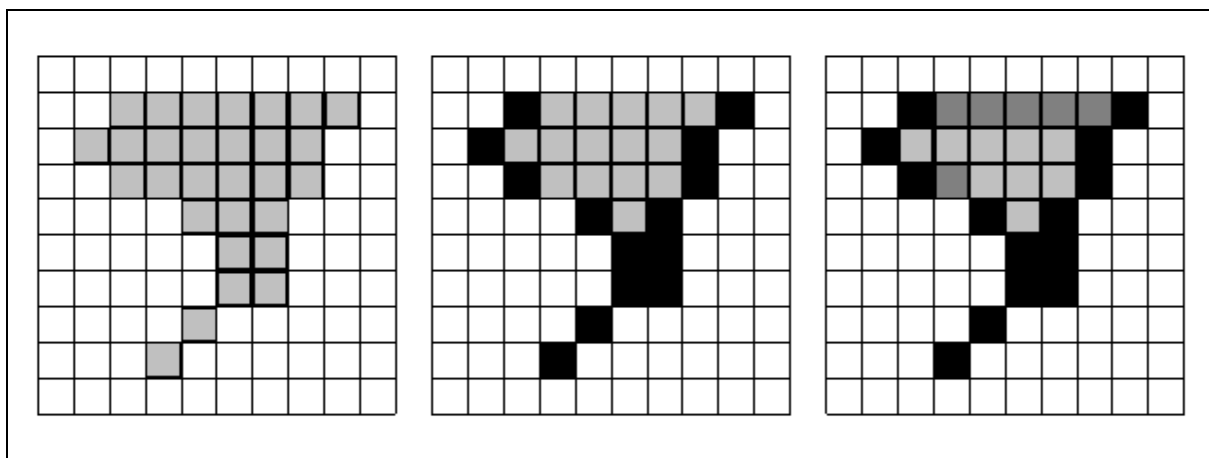
## 9.3 Implementirani postopek poravnave

Omenili smo že, da naš postopek temelji na transformaciji robov ter poplavljanju površin števk. Zato v prvem koraku označimo na sliki, katere točke so robne in katere pripadajo notranjosti števk. Postopek je naslednji.

### *Postopek označevanja robnih in notranjih točk števk*

1. originalno sliko preiskujemo v območju števk po vrsticah od leve proti desni, dokler ne naletimo na slikovni element, ki ima večjo osvetljenost od praga binarizacije;
2. točko označimo kot robno;
3. nadaljujemo preiskovanje originalne slike, dokler ne naletimo na točko, ki ima osvetljenost manjšo od praga binarizacije, ali dokler ne naletimo na rob območja števk;
4. točko, ki se nahaja levo od dobljene točke, označimo kot robno;
5. vse točke med obema robnima točkama označimo kot notranje točke;
6. vračamo se na korak 1, dokler ne pregledamo celotnega področja števk, nato nadaljujemo s korakom 7;
7. originalno sliko preiskujemo v območju števk po stolpcih od zgoraj navzdol, dokler ne naletimo na slikovni element, ki je označen kot robni ali notranji;
8. če je slikovni element označen kot notranji, ga spremenimo v robnega;
9. nadaljujemo preiskovanje originalne slike, dokler ne naletimo na točko, ki ni označena niti kot notranja niti kot robna, oz. dokler ne naletimo na rob slike;
10. če točka nad točko, dobljeno v koraku 9, ni označena kot robna, jo spremenimo v njo;
11. vračamo se na korak 7, dokler ne pregledamo celotnega področja števk.

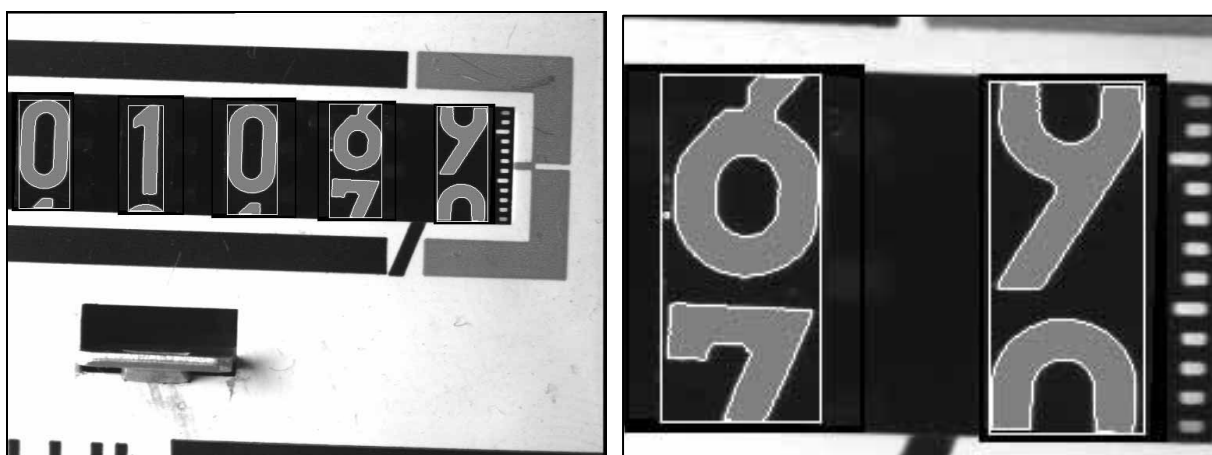
Slika 54 prikazuje vmesni in končni rezultat označevanja točk na sliki.



**Slika 54: Označevanje točk na sliki kot robne in notranje (od leve proti desni: začetno stanje, rezultat horizontalnega preiskovanja, končni rezultat)**

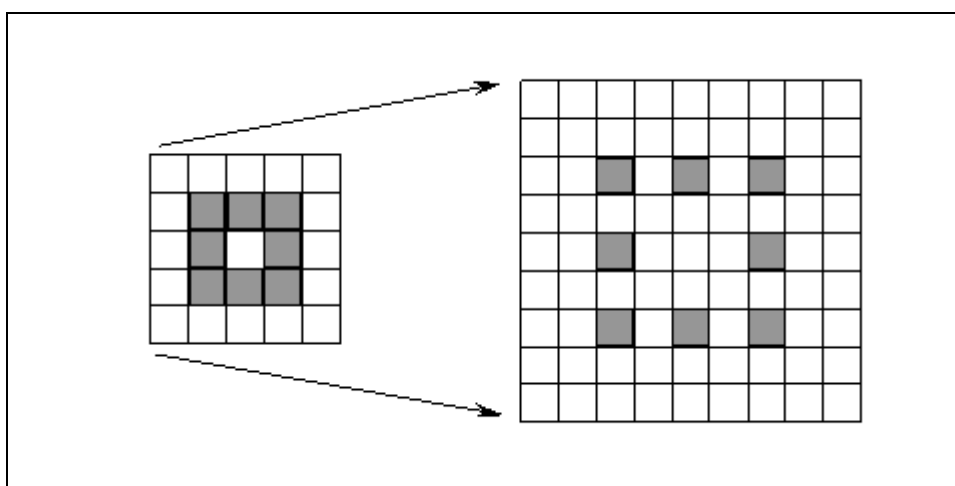
Na levi risbi slike 54 se nahaja namišljeno začetno stanje. Slikovni elementi, ki so svetlejši od praga binarizacije, so prikazani sivo, ostali so beli. Srednja risba prikazuje vmesni rezultat izvajanja korakov od 1 do 6. S črno barvo so označene točke, ki pripadajo robu območja, s sivo pa notranje. Opazimo, da ta del postopka detektira vse točke, ki pripadajo levim in desnim robovom območja; točke, ki pripadajo zgornjim in spodnjim, pa so označene kot notranje. Pomankljivost odpravimo z izvajanjem korakov od 7 do 11. Rezultat vidimo na desni risbi, kjer so novo dobljene točke označene s temnejšo sivo barvo (zaradi nazornosti; dejansko sta obe skupini točk označeni kot robni in jih ne ločujemo med sabo).

Rezultat postopka nad realistično sliko si lahko ogledamo na sliki 55, kjer smo postopek izvedli za vsako števko posebej. Robne točke števk so označene z belo, notranje pa s sivo barvo. Okoli števk smo tudi narisali pravokotnike, ki predstavljajo dejanska področja, v katerih se števke nahajajo. Dobili smo jih tako, da smo si med izvajanjem postopka zapomnili skrajno levo, desno, zgornjo in spodnjo točko, ki smo jo razglasili za robno. Natančno poznavanje teh področij, nam postopek dodatno pohitri, ker nam večjega dela območij izven števk ni potrebno več preiskovati.



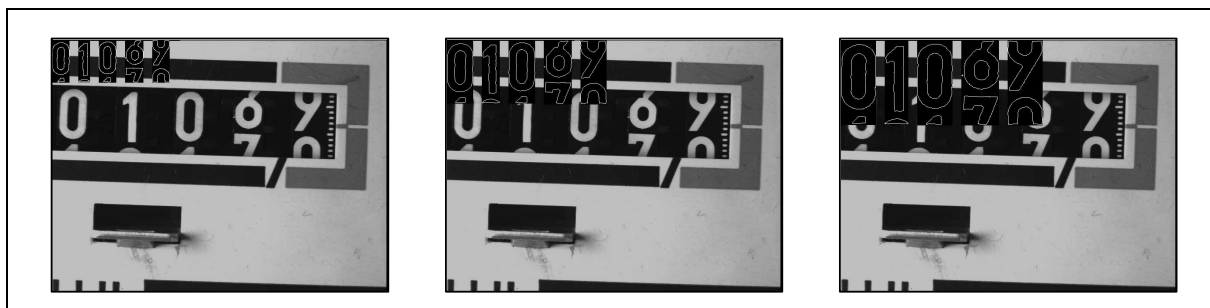
**Slika 55: Prikaz robnih in notranjih točk števk (levo so vidne vse števkke na sliki, desno sta zadnji dve prikazani povečano)**

Naslednji korak poravnave je transformacija robov. Naloge ne moremo rešiti tako, da vse robne točke, ki smo jih dobili v prejšnjem koraku, preslikamo s pomočjo transformacijske enačbe. Težava takega pristopa se pojavi pri slikah, ki so glede na original povečane. V takem primeru bi dobili pretrgane krivulje robov (princip je prikazan na sliki 56). To za nas ni sprejemljivo, saj bomo notranjosti števk poplavljeni, zato morajo biti robovi sklenjene krivulje, sicer bi hkrati poplavlili tudi zunanosti števk in s tem njihove slike povsem pokvarili.

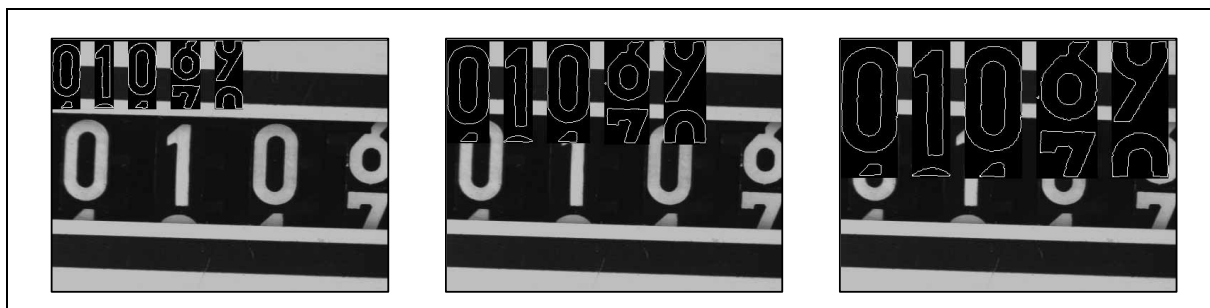


**Slika 56: Pretrganje roba pri veliki povečavi slike**

Postopek preslikave modificiramo tako, da preslikane točke na novi sliki povezujemo med seboj, če se njihovi originali dotikajo oz. so si sosednji (tudi diagonalno). S tem zagotovimo, da so preslikani obrisi števk nepretrgane konture. Rezultat preslikave robov si lahko ogledamo na slikah 57 in 58.



**Slika 57: Transformirani robovi števk (od leve proti desni: višine izločenih števk so 100, 150 in 200 slikovnih elementov)**



**Slika 58: Povečan prikaz transformiranih robov števk (od leve proti desni: višine izločenih števk so 100, 150 in 200 slikovnih elementov)**

Konture števk na slikah 57 in 58 smo dobili tako, da smo rotacijo določili kot nasprotno vrednost rotacije števnikar (poglavje 8.2), faktorje povečave pa smo izbrali tako, da so transformirane števkke visoke 100, 150 in 200 slikovnih elementov. Vidimo, da so števkke postavljene pokončno, kljub temu da so na originalni sliki zarotirane skupaj s števnikom. Pri večjih povečavah opazimo tudi nazobčanost robov, ki so posledica tega, da smo slike sosednjih točk povezovali med seboj z



ravno črto in ne s krivuljo višjega reda. Izkaže se, da ta slabost postopka ni moteča za odčitavanje vrednosti in je le lepotna napaka.

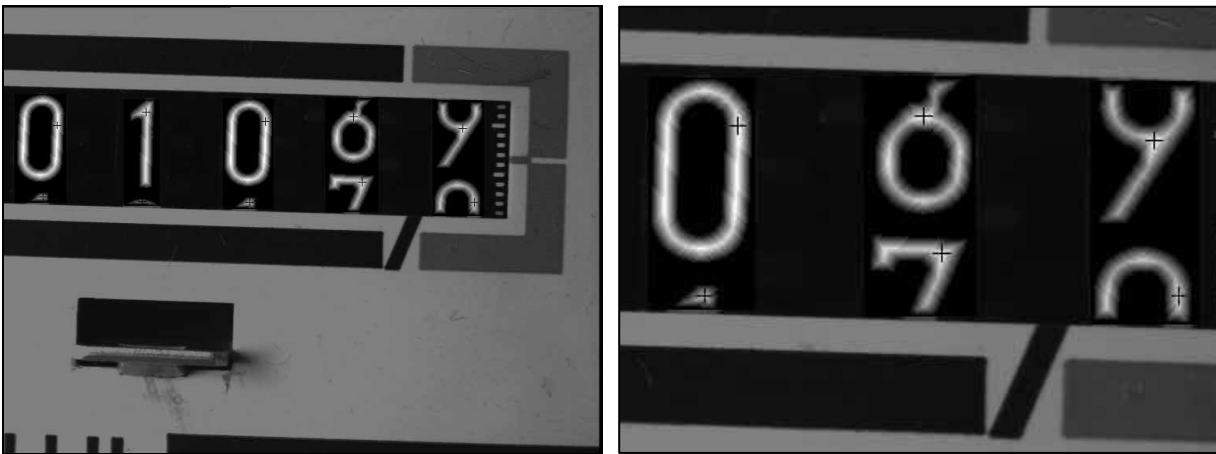
Števke na poravnani sliki moramo še poplaviti, da bo rezultat preslikave tak, kot si ga želimo. V ta namen moramo določiti točke, kjer poplavljanje začnemo. Da zagotovimo robustnost postopka, moramo začetne točke izbrati čim bolj v notranjosti področij števk, da se začetna točka poplavljanja ne bi preslikala v zunanost števke (to je zlasti nevarno pri velikih pomanjšavah, kjer bi točka v bližini roba števke lahko bila preslikana na njen rob), s čimer bi povzročila poplavljanje napačnih področij. Točke, ki so za pričetek poplavljanja najbolj ustrezne, poiščemo tako, da vsaki točki določimo, koliko je oddaljena od roba števke, ter izberemo tisto, ki je oddaljena najbolj. Če je takih točk več, izberemo tisto, katere vsota oddaljenosti sosedov je največja; povsem enako, kot smo v poglavju 7.3 določali točke, ki smo jih razširjali v temne regije. Števke pričnemo poplavljanje na tistih mestih, kamor se te točke transformirajo po enaki formuli, kot smo jo uporabili za transformacijo robov. Postopek je naslednji.

***Postopek zapolnjevanja transformiranih robov števk***

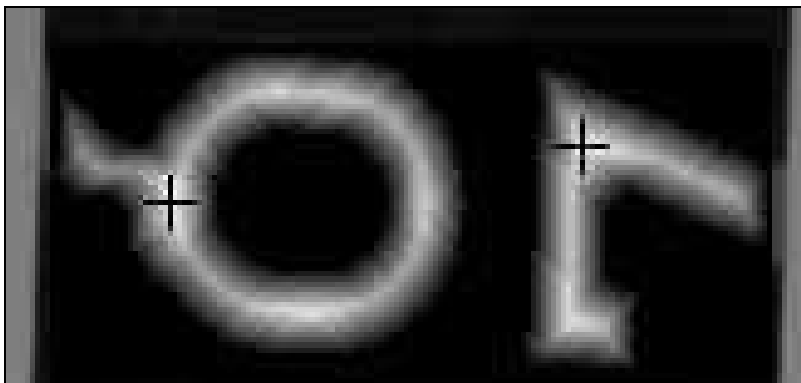
1. Na originalni sliki določimo vsaki notranji točki njeno oddaljenost od prve robne točke. Razdalja je definirana kot najmanjše število notranjih točk, ki jih moramo prepotovati, da pridemo do robne točke; pri tem lahko potujemo v vseh osmih smereh, torej tudi diagonalno (sliki 59 in 60);
2. poiščemo točko z največjo oddaljenostjo. Če je takih točk več, izberemo tisto, katere vsota oddaljenosti sosedov je največja. Če je tudi takih točk več, izberemo katerokoli;
3. poplavimo originalno sliko s pričetkom v izbrani točki;
4. izbrano točko transformiramo in poplavimo poravnano sliko s pričetkom v transformirani točki;
5. dokler so na sliki točke z dovolj veliko oddaljenostjo, se vračamo na korak 2, sicer prekinemo.

Komentar postopka:

- V koraku 3 poplavljam originalno sliko s pričetkom v izbrani točki, da bi videli, kolikšen del transformirane slike bomo z njo poplavlili. Celotno področje, ki je poplavljen, ne pride več v poštev pri izbiri naslednje točke. Na ta način zagotovimo poplavljanje cele števk ne glede na to, iz koliko pretrganih območij je sestavljena z najmanjšim številom poplavljanj, ter preprečimo neskončno zanko, ki bi neprestano izbirala isto točko.

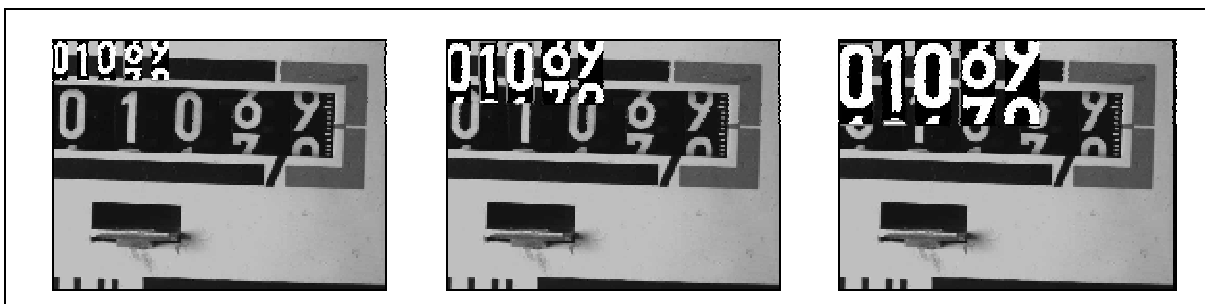


**Slika 59: Oddaljenost notranjih točk števk od njihovih robov (večja osvetljenost točke ponazarja večjo oddaljenost) in optimalne točke poplavljanja (prikazani s temnimi križci) števk (levo je prikazana zajeta slika v celoti, desno vidimo povečane zadnje tri števk)**

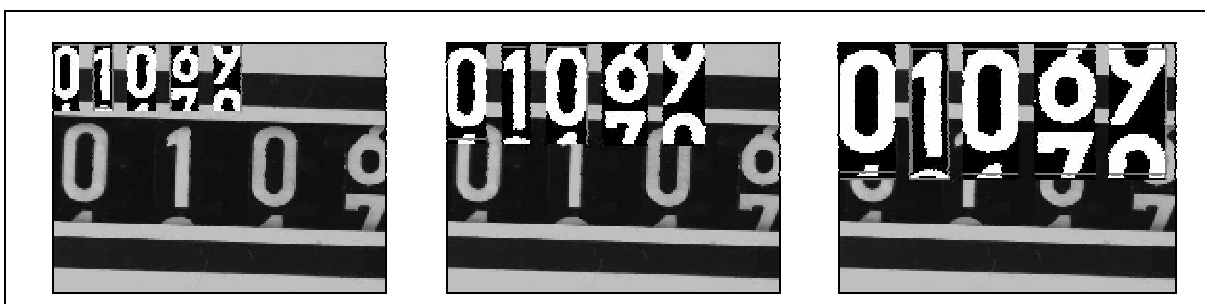


**Slika 60: Močno povečan in za 90° zarotiran prikaz optimalnih izvorov**

Rezultat postopka si lahko ogledamo na slikah 61 in 62, kjer vidimo, da smo konture s slik 57 in 58 uspešno zapolnili. S tem je kalibracija števk končana.



Slika 61: Poravnane števk (od leve proti desni: njihove višine so 100, 150 in 200 slikovnih elementov)



Slika 62: Povečan prikaz poravnanih števk (od leve proti desni: njihove višine so 100, 150 in 200 slikovnih elementov)

## 9.4 Način uporabe poravnave

Možnost spremeniti zasuk in povečavo števk sama po sebi ni zadosten pogoj za uspešno izvedbo poravnave. Pomembna je tudi izbira pravih vrednosti teh parametrov. Na tem mestu si bomo ogledali, kako postopamo pri poravnavi, ko pripravljamo slike števk za izgradnjo traku (poglavje 1), ter pri določanju števkinih vrednosti (poglavje 1).

V obeh primerih izberemo vrednost zasuka števk kot nasprotno vrednost rotacije števnika. S tem dosežemo, da so poravnane števk postavljene pokončno ne glede na to, kakšen je zasuk števnika na sliki.

Pri izgradnji traku izberemo vrednost povečave manjšo ali enako ena (slike števk pomanjšamo ali pustimo enake velikosti, pri katerih smo jih zajeli). Vrednosti, večje od ena, ne priporočamo uporabljati, saj s tem postane trak večji, določanje vrednosti z njegovo pomočjo pa dolgotrajnejše, ker je čas računanja razdalje med njim in števk večji. Ob tem pa se informacija, ki jo trak vsebuje ne poveča, saj bi morali števec zajeti pri večji povečavi, če bi hoteli imeti na njem več detajlov in s tem natančnejše odčitavanje. Pomanjševanje števk pri izgradnji pa je smiselno, saj imamo s tem možnost izbirati med večjo hitrostjo odčitavanja ali natančnejšim odčitkom (poglavje 1).

Ko števec odčitavamo, je izbira povečave preprosta. Zagotoviti moramo enako velikost števk, kot smo jo izbrali pri izgradnji traku. Faktor povečave je torej razmerje med višino zajete števk in višino števk na traku.

## 10 Izgradnja traku

V poglavju 4.3 smo nakazali princip določanja vrednosti številke. Gre za primerjavo njene slike s trakom, ki predstavlja raztegnjeno sliko kolesčka števnik. V predhodnih poglavjih opisani postopki nam locirajo posamezne številke na slikovnem vzorcu ter njihove slike ustrezno poravnajo, tako da so primerne za neposredno primerjavo s trakom. Da bi posameznim številkam določili vrednosti, potrebujemo samo še ustrezen trak. Izgradnji le-tega pravimo faza učenja določenega tipa števca. Potrebno jo je izvršiti samo enkrat za posamezni tip aparata, nato lahko izvedemo poljubno število odčitavanj njegovih vrednosti. Da bi trak uspešno zgradili, potrebujemo določeno število slik kolesčka v različnih položajih (zasukih okoli svoje osi). Od načina zajema teh slik je odvisno, ali bo izdelava traku praktična in enostavna, t.j. ali jo bodo lahko izvajali operaterji v proizvodnji.

V poglavju 10.1 bomo opisali način zajemanja slik za izgradnjo traku. Postopek, ki iz njih sestavi trak, bomo predstavili v poglavju 10.2.

### 10.1 Zajem slik za izgradnjo traku

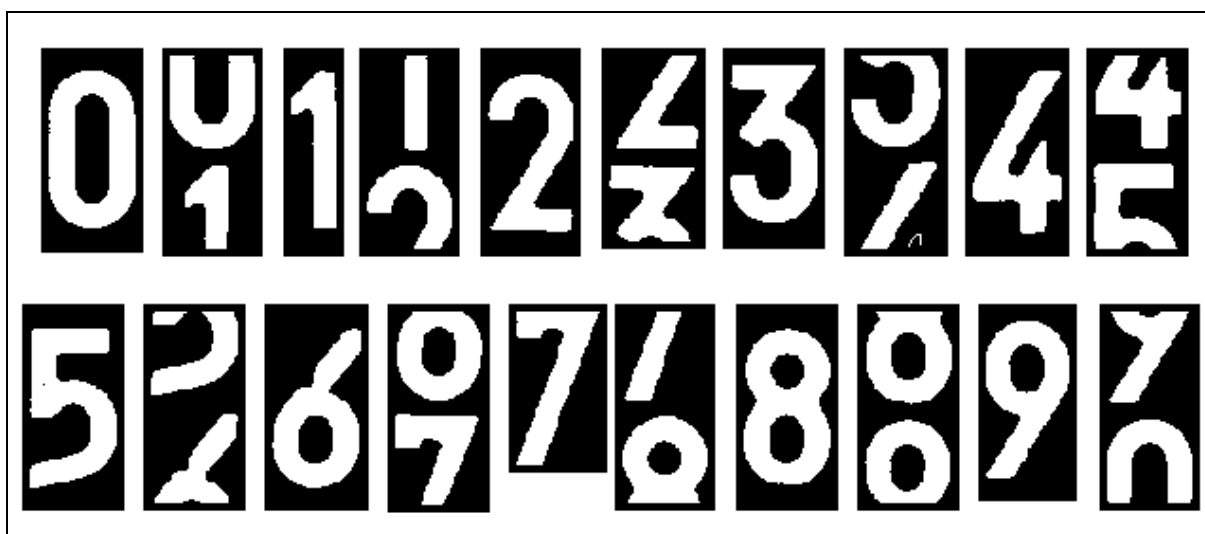
Trak lahko sestavljamo na različne načine. Prva možnost, ki se nam odpira, je počasno vrtenje kolesčka okoli njegove osi ter zajem ozkih odsekov slike, katere vertikalno združujemo v sliko celotnega traku. Kljub preprosti ideji je ta postopek v praksi težko izvedljiv. Zajeti moramo veliko število vzorcev kolesčka (če so pasovi, ki jih zajemamo široki en razdelek, moramo zajeti sto slik), ki mora biti ob tem zelo natančno pozicioniran, ker bi se majhne napake pri vsakokratni poziciji akumulirale

preko celotnega obrata, kar bi trak po vertikali deformiralo. Napaka v izgradnji bi se direktno odražala pri odčitavanju vrednosti števk (poglavje 11.1).

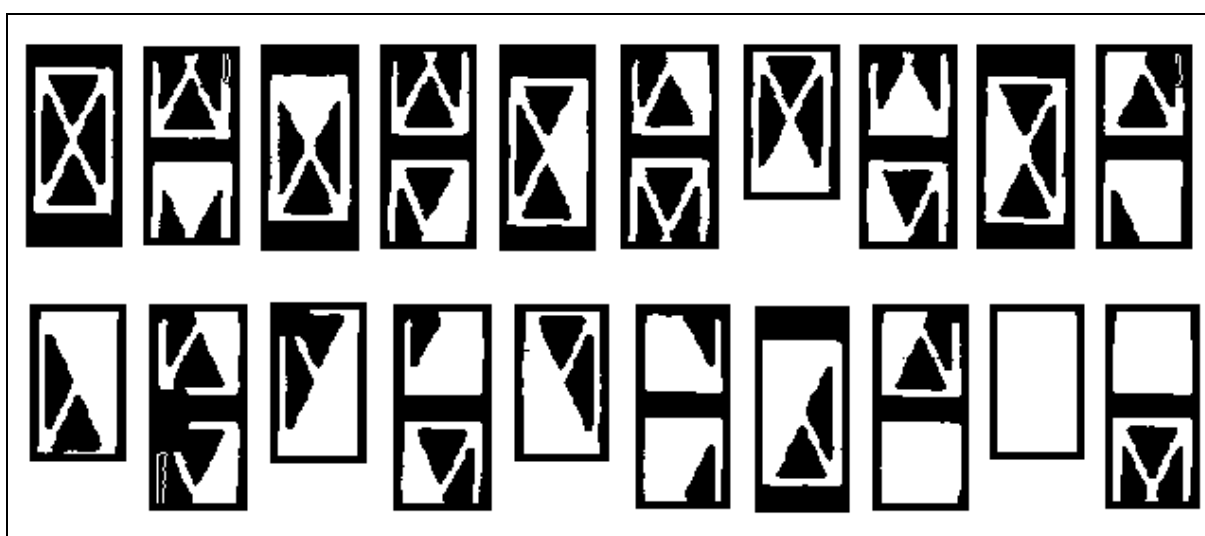
Postopek lahko izboljšamo tako, da ugotavljamo medsebojno odvisnost v sekvenci zajetih slik. Ker se rotacija koleščka ne spreminja veliko med posameznimi zajemi, bodo večji deli zajetih slik redundantni (nosili isto informacijo). To lahko izkoristimo za relativno natančno izgradnjo traku tudi v primeru slabo pozicioniranega koleščka, saj lahko iščemo medsebojne položaje zaporedno zajetih slik na mestih, kjer se le-te najboljše prilegajo med sabo. Trak torej ne sestavljamo po vertikali uniformirano, pač pa slike nanj nanašamo tako, da so med seboj skladne. Poskusi kažejo, da lahko trakove, ki jih dobimo na tak način, uspešno uporabimo za natančno določanje vrednosti števk (poglavje 1).

Naslednje vprašanje, ki se zastavlja, je, koliko slik potrebujemo, t.j. kolikšen del koleščka lahko verodostojno prenesemo na trak z eno zajeto sliko. Naš postopek trak uspešno zgradi z dvajsetimi zajetimi slikami. Prvih deset prikazuje cele številke, ostale pa vmesne položaje med njimi. Primera slik sta prikazana na slikah 63 in 64. Prva slika prikazuje odseke koleščka, ki ima natisnjene arabske številke, s pomočjo katerih prikazuje izmerjeno vrednost. Na drugi smo zajeli odseke koleščka, ki ima številke zakodirane s posebnimi vzorci (primer takega števca si lahko ogledamo na sliki 3, str. 8)

Z drugimi števili slik nismo eksperimentirali. Večje število smatramo za nepotrebno, saj rezultati kažejo, da so z našim postopkom zgrajeni trakovi dovolj kvalitetni (poglavje 1).



Slika 63: Primer zajetih slik kolesčka za izgradnjo traku



Slika 64: Primer zajetih slik za izgradnjo traku pri uporabi kolesčka, ki prikazuje vrednosti s pomočjo posebnih kod namesto števk

Teoretično najmanjše število slik je deset, kjer so številke zajete tako, da so vidne v celoti, pri čemer med slikami ni nobene redundantne informacije, ki bi omogočala odpravljanje napak v položajih kolesčka ob zajemu. Tudi samo širino razmaka med števki s tako majhnim številom slik ne moremo določiti, zato sklepamo, da je v

praksi potrebnih vsaj dvanajst do štirinajst slik. Naš postopek jih uporablja nekaj več in si s tem zagotovi dovolj veliko redundanco informacije, da lahko trak sestavi tudi pri večjih odstopanjih položajev kolesčka od idealnih. Za vse trakove, ki smo jih do sedaj zgradili, smo kolesček ročno nastavljali na ustrezne položaje, kar pomeni, da tega nismo opravili precizno. Kljub temu so zgrajeni trakovi ustrezni za nalogo, ki jo opravljajo.

## 10.2 Postopek izgradnje traku

Jedro postopka predstavlja prileganje slik dvajsetih odsekov traku (sliki 63 in 64) med seboj ter njihov vnos na trak. Žal se izkaže, da moramo postopek večkrat ponavljati. Razlog je v tem, da običajno ne moremo kompenzirati rotacijske napake v enem prehodu. Podatek o rotaciji številke na zajeti sliki dobimo s pomočjo vertikalnih mej števnika (poglavje 8.1), kar je zadosti natančno za poravnavo posamezne številke, katere vrednost iščemo z že narejenim trakom. Za samo izgradnjo pa je natančnost premajhna, ker se drobne napake vzdolž traku seštevajo in rezultat je tak, da je celotni trak za opazen kot zarotiran oz. odmaknjen od idealne pokončne oblike (slika 65).



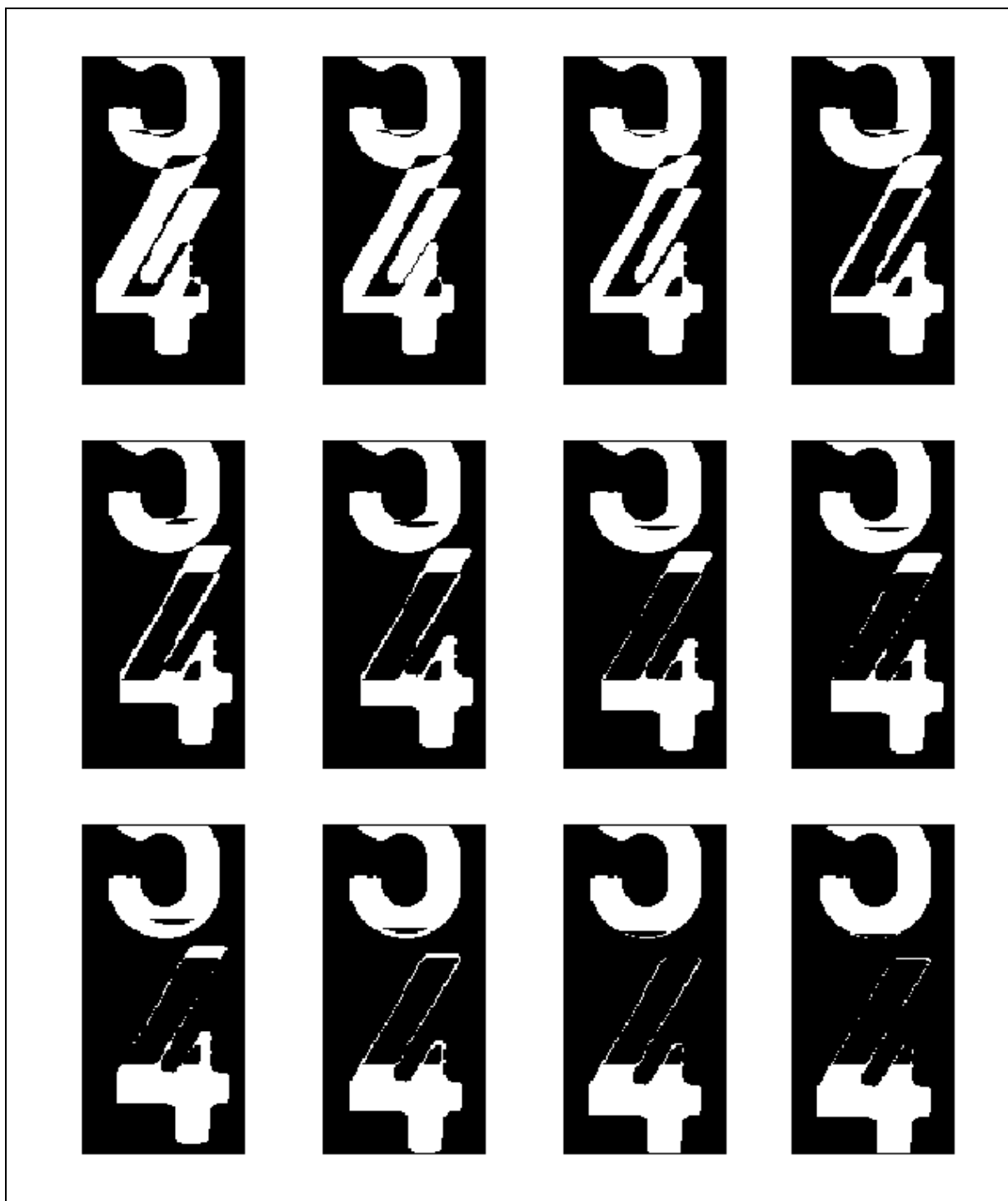
**Slika 65: Primer traku, ki nima kompenzirane rotacijske napake**

Podrobna proučitev izvora napake nam je pokazala, da je vzrok v nenatančni izvedbi števnika, zaradi česar vertikalne osi števk niso povsem pravokotne na njegove vertikalne meje. Napako zmanjšamo tako, da trak zgradimo večkrat. Vsakič ocenimo, za kakšen kot je zarotiran in na podlagi ocene spremenimo kot rotacije, s katerim poravnamo slike za naslednjo izgradnjo. Končni rezultat je trak, ki je natančnejši, čeprav na njem še vedno lahko opazimo zasuk, ki pa pri določanju vrednosti števk ne moti. Celoten postopek izgradnje je torej naslednji.

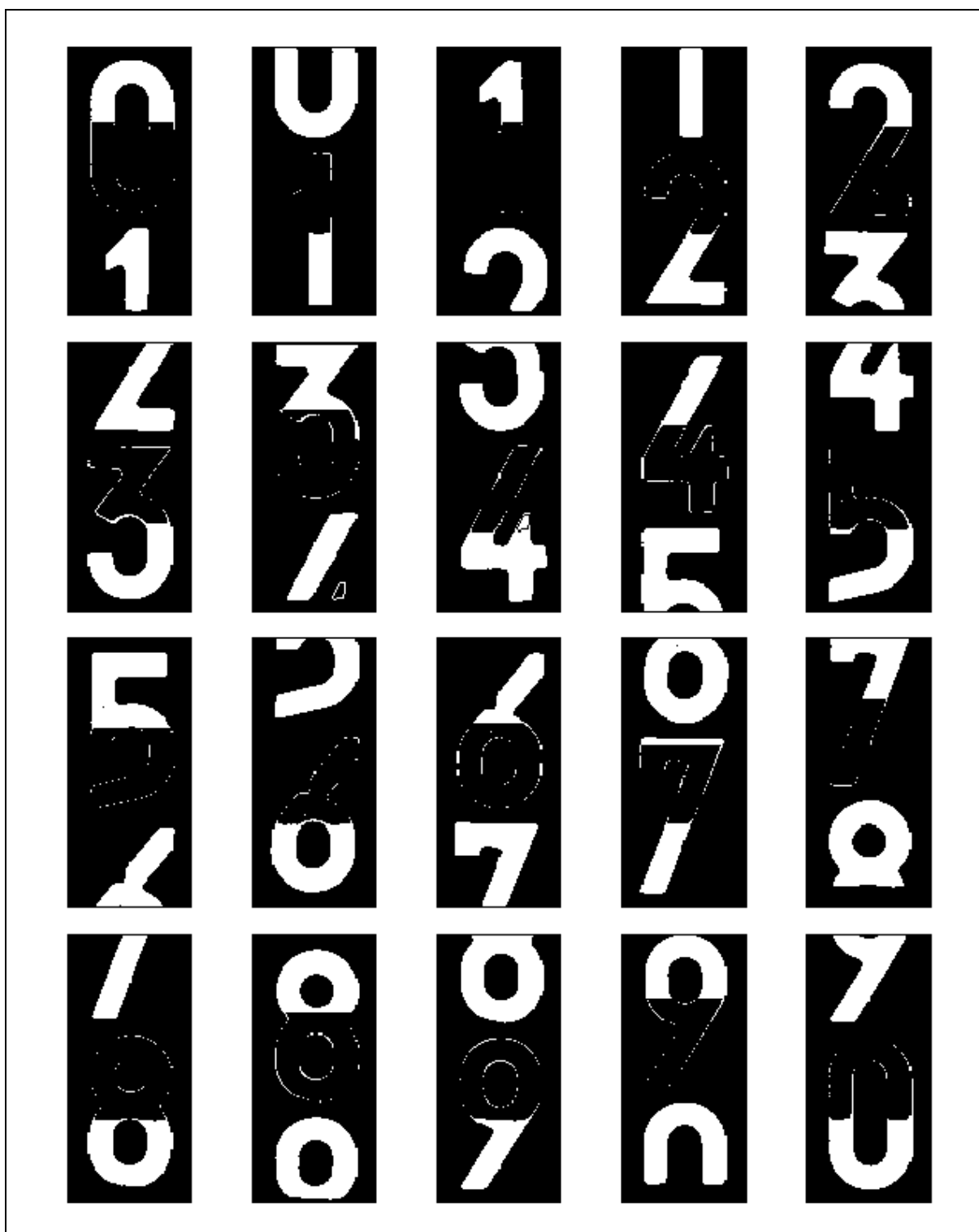


**Postopek izgradnje traku iz dvajset zajetih slik kolesčka števnik**

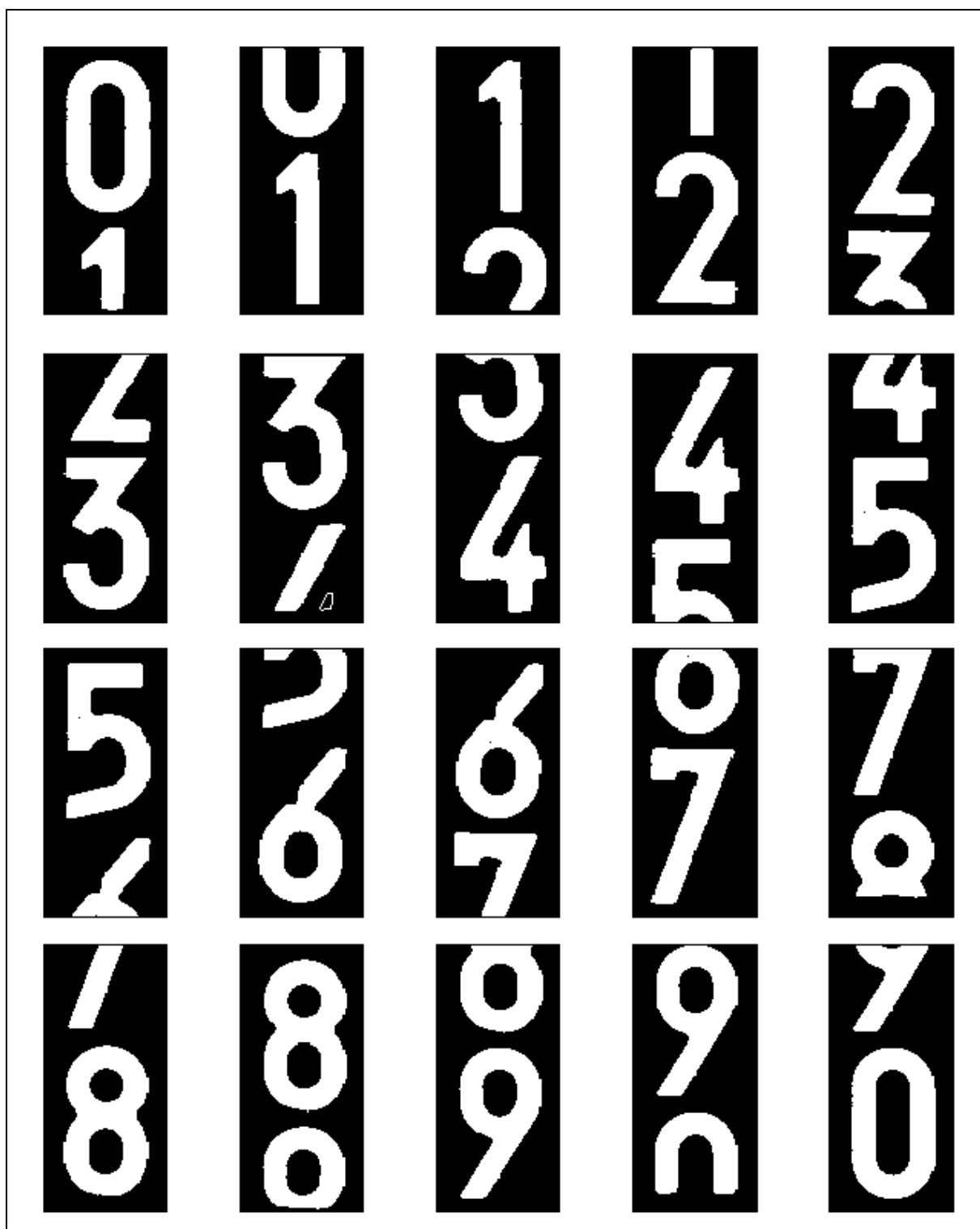
1. Inicializacija postopka: spremenljivko "Število\_preostalih\_izgradenj" postavimo na 3, spremenljivki "Zadnja\_kompenzacija" in "Celotna\_kompenzacija" postavimo na 0;
2. poravnava posameznih slik za izgradnjo z izbrano povečavo in rotacijo, ki je vsota nasprotno vrednosti zasuka števnik in "Zadnje\_kompenzacije";
3. prileganje posameznih parov slik med seboj (primer poteka med vmesnim položajem števk tri in štiri ter sliko cele števk štiri je prikazan na sliki 66, rezultat poravnave vseh parov pa si lahko ogledamo na slikah 67 in 68);
4. poravnane slike vnesemo na trak tako, da na začetku pustimo prazen prostor približno v velikosti ene števk, nato vrišemo sliko števk 9 (t.j. predzadnjo in zadnjo zajeto sliko; sliki 63 in 64), nato slike števk od 0 do 9 (zajete slike od prve do predzadnje) ter še eno sliko števk 0 in 1 (zadnja in prva zajeta sliko);
5. določimo rotacijsko napako traku po postopku, opisanem v nadaljevanju ("Postopek ocenjevanja rotacijske napake traku");
6. če je to prvi prehod, si napako zapomnimo v spremenljivki "Najmanjša\_napaka" ter shranimo kopijo dobljenega traku v strukturi "Zgrajen\_trak". "Zadnjo\_kompenzacijo" in "Celotno\_kompenzacijo" nastavimo na nasprotno vrednost ocenjene napake in nadaljujemo s korakom 2;
7. če je ocenjena napaka po absolutni vrednosti manjša od "Najmanjše\_napake", dobljeni trak shranimo v "Zgrajen\_trak", "Število\_preostalih\_izgradenj" nastavimo na 3, "Zadnjo\_kompenzacijo" nastavimo na preostalo napako, "Celotni\_kompenzaciji" preostalo napako prištejemo ter nadaljujemo s korakom 2;
8. zmanjšamo "Število\_preostalih\_izgradenj" za 1 in če doseže vrednost 0, postopek prekinemo;
9. "Zadnjo\_kompenzacijo" delimo z dva ter njeno vrednost odštejemo od "Celotne\_kompenzacije". Nadaljujemo s korakom 2.



Slika 66: Potek prileganja dveh sosednjih slik za izgradnjo traku (zgoraj levo: izhodiščni položaj prileganja; po vrsticah sledijo od leve proti desni vmesni položaji; spodaj desno: končni položaj, kjer se sliki najbolj ujemata)



Slika 67: Paroma prilagojene slike za izgradnjo traku (izključujoča disjunkcija)



Slika 68: Paroma prilagojene slike za izgradnjo traku (navadna disjunkcija)

Komentar postopka:

- v prvem koraku inicializirana spremenljivka "Število\_preostalih\_izgradenj" predstavlja oceno, kolikokrat bomo postopek izgradnje še ponovili. V koraku 8 to spremenljivko zmanjšamo za ena v primeru, da ponovni preizkus ni dal boljših rezultatov, sicer je ne manjšamo. Na ta način ponavljamo postopek, dokler se rezultat izboljšuje. Spremenljivka "Celotna\_kompenzacija" ima vrednost kota zasuka, ki ga prištejemo k rotaciji števnik v trenutnem koraku. V prvem poskusu ima vrednost 0, ker še nimamo informacije o tem, za koliko moramo zasuk popraviti. V spremenljivko "Zadnja\_kompenzacija" shranimo spremembo "Celotne\_kompenzacije" ob zadnji izgradnji. Uporabimo jo v koraku 9 v primeru, ko je novi rezultat slabši od starega, da poskusimo kompenzirati s polovično vrednostjo;
- v koraku 4 sestavimo trak. Nanj dodamo tri redundantne številke: na začetku 9, na koncu pa 0 in 1. S tem smo dosegli navidez krožno in nepretrgano obliko traku, ki poenostavi postopek določanja vrednosti števk (poglavje 11.1).

Rotacijsko napako traku bomo ocenili s primerjavo odmikov obeh slik, ki predstavljata številko 0 od levega roba traku. Ta metoda je ugodna, ker se ne opira na oblike števk, ampak samo ugotavlja, koliko je zgornja slika zamaknjena v primerjavi s spodnjo (slika 69). Pri tem izkoriščamo redundantno informacijo, ki je na traku prisotna zaradi večkratnega vrisa nekaterih števk na začetek in konec. Če tega ne bi bilo, bi morali iskani odmik določati s pomočjo *a priori* znanja o oblikah števk. Postopek je naslednji.

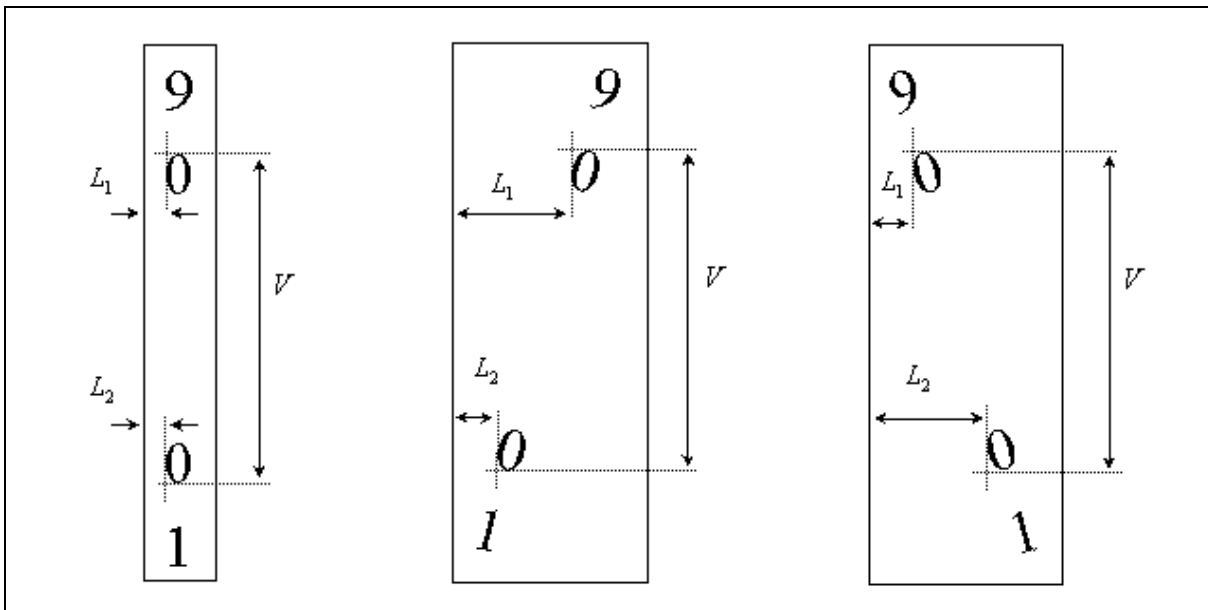
**Postopek ocenjevanja rotacijske napake traku**

1. določimo odmik levega roba zgornje številke 0 od levega roba traku (označimo z  $L_1$ );
2. določimo odmik levega roba spodnje številke 0 od levega roba traku (označimo z  $L_2$ ). Količino  $(L_1 - L_2)$  označimo s  $H$

3. določimo višino traku med obema ničloma kot razliko spodnjega roba spodnje in zgornjega roba zgornje ničle (označimo z  $V$ );
4. izračunamo zasuk traku po naslednji formuli:  $Z = \arctg((L_1 - L_2)/V)$ .

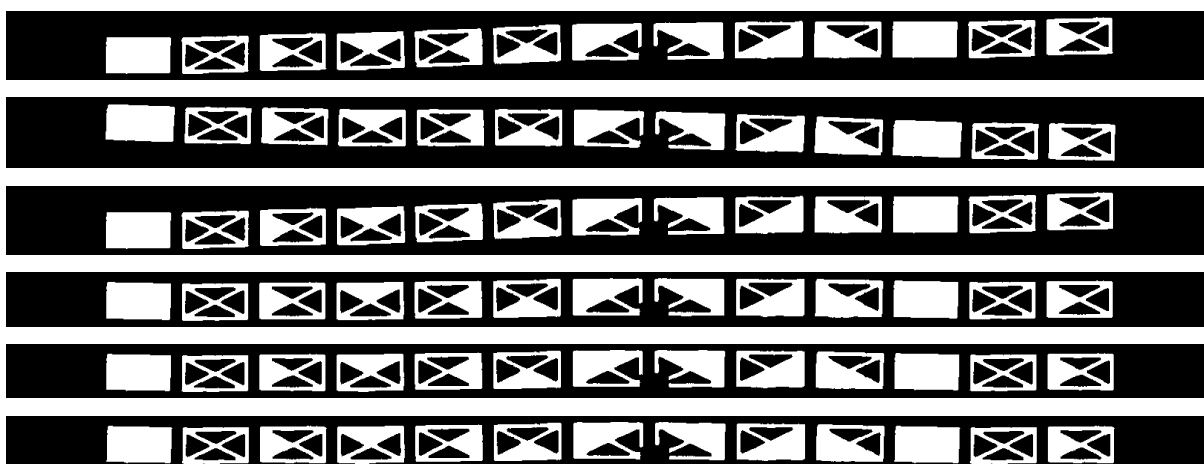
Komentar postopka:

- če trak ne bi imel rotacijske napake, bi bila leva robova obeh ničel enako oddaljena od levega roba traku, zato je količina  $(L_1 - L_2) = H$  lahko merilo za napako rotacije. Njen predznak govori tudi o tem, v katero smer je trak zasukan (pozitivno, če je zasuk souren);
- da dobimo napako izraženo v radianih, kar potrebujemo za določanje parametrov poravnave števk ob naslednji izgradnji, moramo določiti še višino traku ter izračunati arcus tangens razmerja  $H/V$ .



**Slika 69: Ocenjevanje rotacijske napake traku (od leve proti desni: trak brez napake, rotacija sournu, rotacija protiurno)**

Rezultate posameznih prehodov z različnimi kompenzacijami rotacijske napake pri izgradnji traku s slikami kod si lahko ogledamo na sliki 70. Slika 71 prikazuje končni rezultat izgradnje traku s slikami števk in kodirnih vzorcev.



**Slika 70: Potek kompenziranja rotacijske napake traku (od zgoraj navzdol: nekompenziran trak, vmesni rezultati in kompenziran trak)**

Implementirani postopek določanja vrednosti števk potrebuje tudi položaje posameznih števk na traku (za vsako od njih lokacije vseh štirih robov ter njihovo sredino v vertikalni smeri). Postopek, s katerim določimo te podatke, je naslednji.

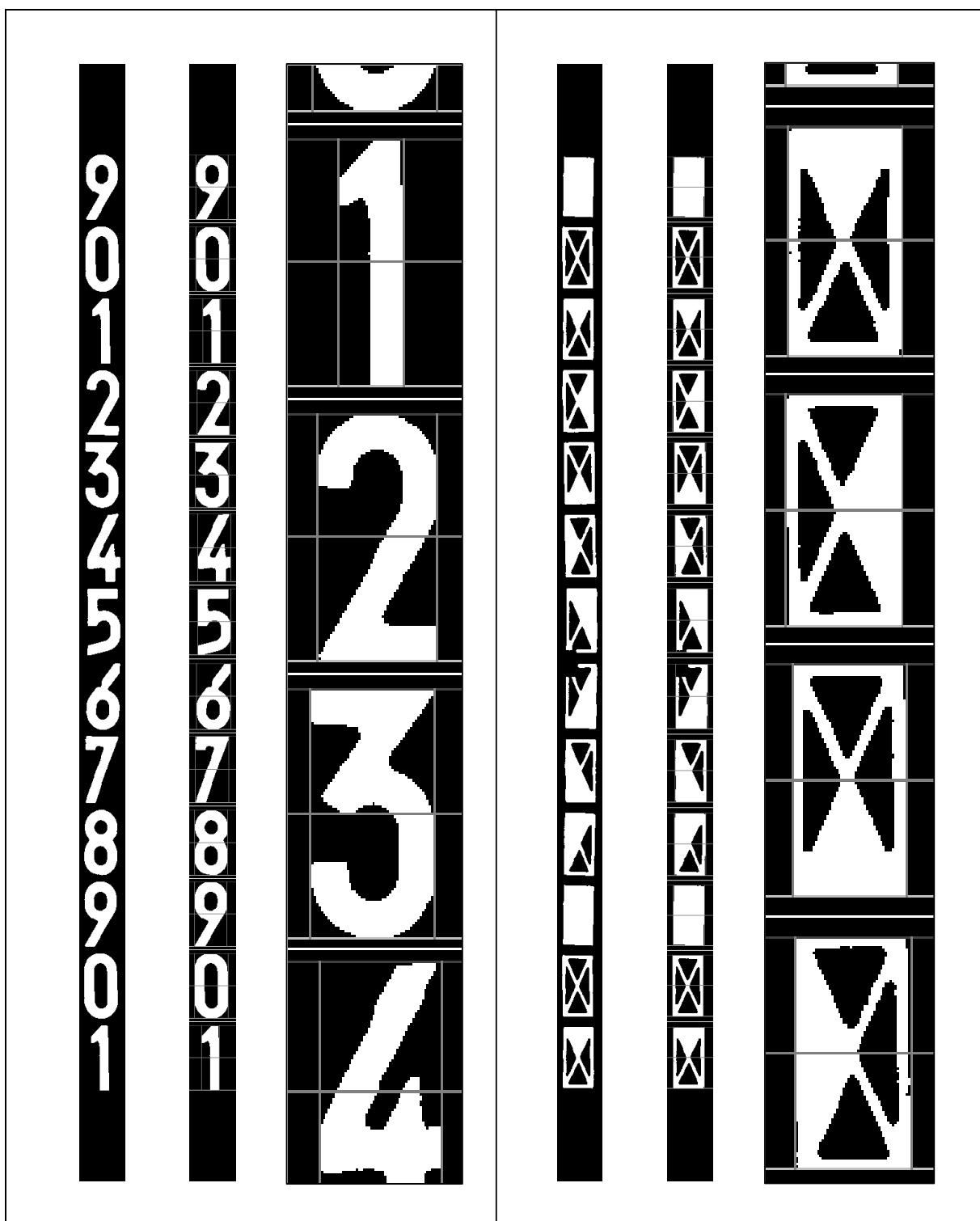
***Postopek določanja položaja posameznih števk na traku***

1. inicializiramo seznam zgornjih, spodnjih, levih in desnih robov števk ter njihovih sredin. Prvo vrstico traku označimo kot naslednjo, ki jo preiskujemo;
2. trak preiskujemo od naslednje vrstice navzdol, dokler ne naletimo na svetlo točko (trak je binarna slika, torej iščemo točko, ki je različna od 0);
3. vrstico (označimo jo z  $ZR$ ) damo na seznam zgornjih robov števk;
4. trak preiskujemo od naslednje vrstice navzdol, dokler ne naletimo na tako, ki nima nobene svetle točke;
5. vrstico pred njo (označimo jo s  $SR$ ) damo na seznam spodnjih robov števk;
6. vrstico, ki se nahaja v sredini obeh pravkar dobljenih robov števk ( $S\check{S} = \frac{ZR + SR}{2}$ ), damo na seznam sredin števk;
7. v območju med  $ZR$  in  $SR$  vrstico preiskujemo trak po stolpcih od leve proti desni, dokler ne naletimo na svetlo točko;

8. dobljeni stolpec damo na seznam levih robov števk;
9. v območju med *ZR* in *SR* vrstico preiskujemo trak po stolpcih od desne proti levi, dokler ne naletimo na svetlo točko;
10. dobljeni stolpec damo na seznam desnih robov števk;
11. korake od 2 do 10 ponavljamo za vse števke vrisane na trak;
12. inicializiramo seznam sredine praznih prostorov med števki;
13. sredine praznih prostorov med dvema sosednjima števki izračunamo kot aritmetično sredino med spodnjim robom prve ter zgornjim robom naslednje. Vrednosti vpišemo na seznam sredin praznih prostorov.

Rezultat postopka je šest seznamov, na katerih se nahajajo vsi štirje robovi števk, njihove sredine ter sredine praznih prostorov med njimi. Grafično so prikazani na sliki 70, kjer smo dobljene robove vrisali na trak.





**Slika 71: Končni rezultat postopka izgradnje traku, prikazan brez in z vrisanimi lokacijami posameznih števk (levo: trak s števki, desno: trak s kodami)**

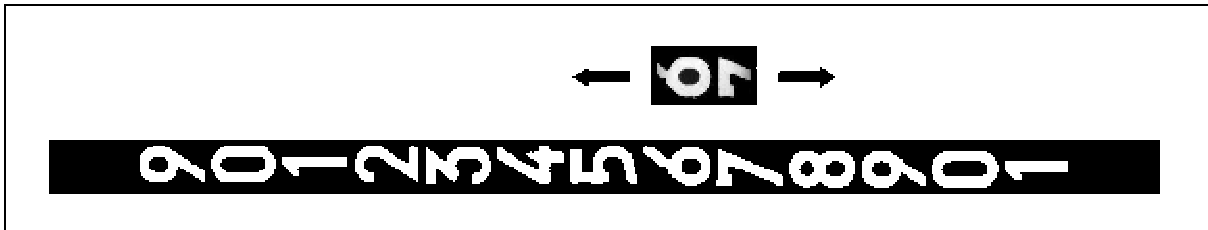
# 11 Določanje vrednosti števkam

Primerjava števk s trakom ter določanje njihovih vrednosti je osrednji korak celotnega postopka odčitavanja vrednosti števca. Vsi dosedaj opisani postopki so samo večali robustnost in uporabnost sistema s tem, da so omogočali njegovo uporabo v širšem območju okoliških parametrov. V strogo predpisanih pogojih bi bilo mogoče implementirati sistem tudi brez začetne analize slike ter lociranja števk (pri dobro definirani osvetljenosti, položaju in lokaciji števnik na sliki bi bili ti postopki odvečni), določanju vrednosti števkam pa se ne moremo izogniti.

V tem poglavju bomo opisali postopek določanja vrednosti števk, ki temelji na primerjavi njihove poravnane slike s trakom (poglavje 11.1). Uporabo te metode smo utemeljili v poglavju 4.3, kjer smo za naš primer uporabe pokazali njene prednosti v primerjavi z industrijskimi OCR postopki.

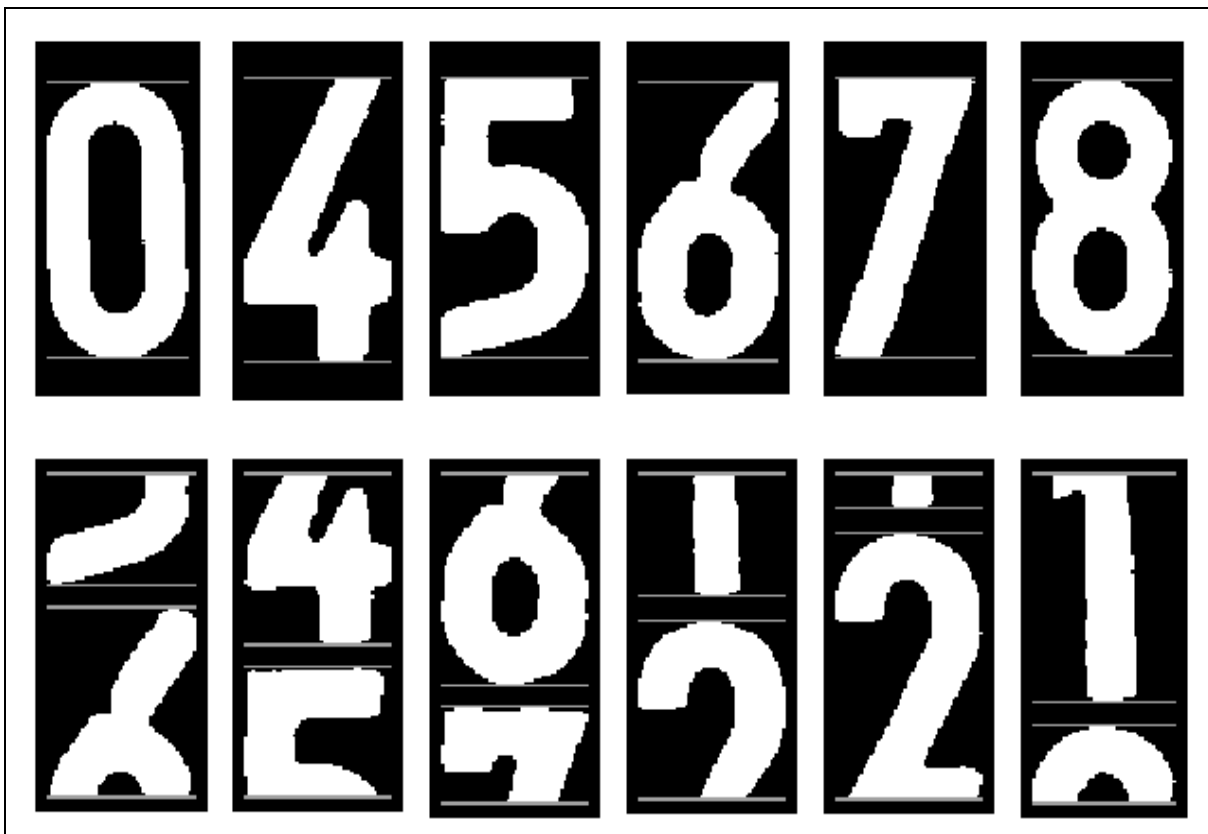
## 11.1 Primerjava števk s trakom

Vhodna podatka postopka primerjave sta slika števke, kateri določamo vrednost, in ustrezen trak skupaj s seznamom lokacij števk na njem (slika 72). Najpreprostejša izvedba določanja vrednosti števke bi bila primerjava njene slike po celi dolžini traku. Položaj najboljšega ujemanja slike s trakom oz. njegov odmik od začetka prve števke nič je direktno koreliran z iskano vrednostjo (poglavje 4.3). V praksi uporabljamo nekoliko spremenjeno metodo določanja vrednosti, ki je hitrejša, zmanjša vpliv netočnosti pri izgradnji in s tem poveča natančnost odčitavanja.



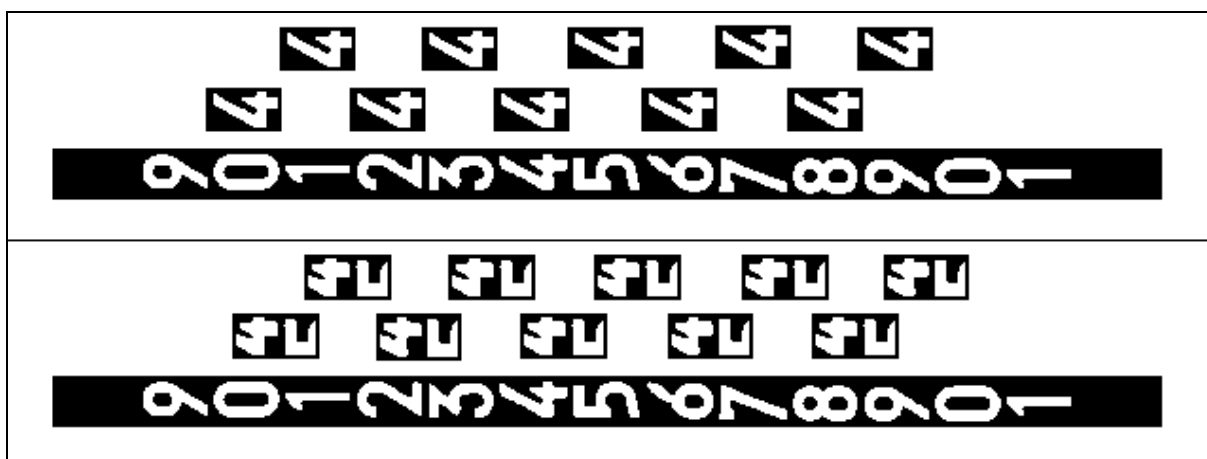
**Slika 72: Neznana številka ter ustrezen trak, ki vstopata v postopek primerjave**

Implementirani postopek najprej pregleda sliko številke ter ugotovi, ali je na njej prikazan položaj kolesčka, kjer se vidi določena številka v celoti, ali pa gre za vmesni položaj med dvema števkama (slika 73).



**Slika 73: Primeri slik, kjer so številke vidne v celoti (zgornja vrsta) in vmesnih položajev med njimi (spodnja vrsta)**

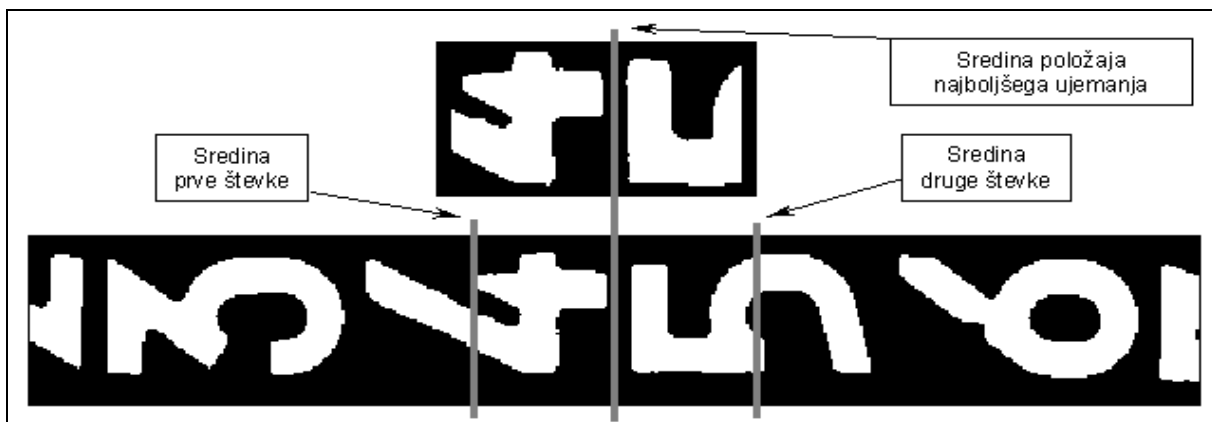
Ker položaje posameznih števk na traku poznamo (poglavje 10.2), lahko določimo tudi, kje se nahajajo vmesni položaji med njimi. Trak torej pregledujemo samo na mestih, kjer se nahajajo številke v celoti, če se na sliki nahaja cela številka oz. samo v vmesnih položajih, če je na sliki vmesni položaj med dvema števkama (slika 74).



**Slika 74: Območja na traku, ki jih pregledujemo v primeru slike cele številke (zgoraj) ter vmesnega položaja med dvema števkama (spodaj)**

V vsakem položaju si zapomnimo lokacijo na traku, kamor pade sredina primerjalne slike, ter razdaljo med njo in trakom. Merilo razdalje je število slikovnih elementov, v katerih se slika razlikuje s trakom na mestu, kjer ju primerjamo. Na mestu, kjer je dobljena razdalja najmanjša, se slika številke najbolj prilega traku. S pomočjo tega podatka lahko določimo vrednost številke na sliki.

Seznam sredin števk na traku (zgradili smo ga v poglavju 10.2 skupaj s trakom) preiščemo in ugotovimo, med sredini katerih dveh števk pade sredina slike neznanе številke (slika 75).



**Slika 75: Ujemanje neznane številke s trakom ter sredini sosednjih števk**

Neznano vrednost številke določimo z linearno interpolacijo vrednosti obeh števk, med katerima se slika nahaja. Izraz je naslednji:

$$V_s = \frac{V_2 - V_1}{S_2 - S_1} \cdot (S - S_1) + V_1$$

kjer pomenijo oznake:

- $V_s$  vrednost številke, ki jo iščemo,
- $V_1, V_2$  vrednosti števk, med kateri pade sredina neznane slike,
- $S_1, S_2$  sredini števk na traku, med kateri pade sredina neznane slike,
- $S$  sredina neznane številke na mestu, kjer se s trakom najbolj prilaga.

S tem, da ne uporabljamo interpolacijskega intervala preko celotnega traku, ampak samo med dvema števkama, smo povečali natančnost določanja vrednosti neznane številke. Vertikalne deformacije traku izven tega območja ne vplivajo na dobljeni rezultat. Nanj vplivajo samo netočnosti traku med števkama, kamor pade neznana slika. Če bi vrednost številke interpolirali preko celotnega traku, bi se posamezne deformacije traku med števkami seštevale in končni odčitek bi bil manj natančen zlasti v območju sredine traku.

## 12 Končni odčitek števca

Vrednosti posameznih števk nam same zase ne pomagajo dosti. Šele s pravilno interpretacijo in združitvijo njihovih vrednosti v končni rezultat smo dosegli zastavljeni cilj. Naivno bi bilo misliti, da lahko odčitek števca tvorimo tako, da seštejemo vrednosti posameznih števk, pomnožene z njihovimi dekadnimi utežmi (desetinke, enice, desetice, ...). Tako ravnanje bi imelo za posledico napačne vrednosti odčitkov ter v nekaterih položajih števca nezvezno karakteristiko.

V tem poglavju se bomo seznanili s problemom, ki nam otežuje tvorjenje rezultata in je posledica neidealne mehanske izvedbe števca ter neizbežne negotovosti pri določanju vrednosti števk (poglavje 12.1). Predlagali bomo rešitev, ki se težavi uspešno izogne (poglavje 12.2).

### 12.1 Težava pri tvorjenju rezultata

Morebitna napaka se pojavi izključno kot posledica slabo pogojenega prehoda vrednosti števk z 9`99 na vrednost 0`00, t.j. v bližini polnega obrata koleščka, kjer vrednost posamezne števk nezvezno preskoči nazaj na nič. Zaradi neidealnih razmer ne moremo pričakovati, da se bo vrednost vseh dekad spremenila natanko ob istem času. Delno je to oteženo zaradi mehanske izvedbe števca, kjer so dekade povezane med seboj z zobniškimi prenosi, ki imajo določene tolerance oz. karakteristike, ki odstopajo od idealnih. Drugi razlog je samo odčitavanje vrednosti s pomočjo traku (težava bi obstajala pri kakršnemkoli načinu odčitavanja, ker nobene

meritve, razen štetja, ne moremo izvesti povsem natančno [8]), kjer lahko števko v neposredni bližini vrednosti nič prepoznamo kot 0`00 ali 9`99.

Posledica tega se kaže na naslednji način. Npr.: desetinke imajo vrednost 0`00, enice pa vrednost 2`00. Obstaja nevarnost, da to odčitamo na štiri različne načine. Prva možnost je vrednost 2`00, kar tudi želimo razbrati. Ostale možnosti so naslednje. V primeru, da bi naš postopek zaradi neidealnih razmer določil enicam vrednost 1`99 ter tvoril rezultat z odstranjevanjem decimalnega dela pri vseh razen pri zadnji dekadi (vsak kolešček razen zadnjega prispeva samo celoštevilski del, ker so decimalke vsebovane v dekadah desno od njega), bi dobili odčitek 1`00. Podobno bi z vrednostjo desetink 9`99 odčitali vrednost 2`99 ali 1`99.

Zaključimo, da lahko v primeru najnižje utežene števke, ki se nahaja blizu ničle, dobimo dva različna rezultata. Če se višje dekade nahajajo v bližini celoštevilске vrednosti (npr. 0`00, 0`99, 1`00, 1`99, ...), se število možnosti še dodatno poveča. Da bo sistem pravilno in ponovljivo odčitaval vrednosti števecv, moramo težavo odpraviti.

## 12.2 Preprečevanje preskokov

Problem lahko rešimo tako, da vrednosti števk ne interpretiramo izolirano, ampak upoštevamo njihove medsebojne odvisnosti, t.j. mehanski prenos med koleščki. S primerjavo vrednosti sosednjih števk lahko brez težav ugotovimo, kdaj je prišlo do prezgodnjega preskoka vrednosti. Če ima npr. števka vrednost 9`99 in naslednja levo od nje vrednost 5`00, vemo, da predhodna vrednost zaostaja, saj se v idealnih razmerah ne more zgoditi, da bi se druga dekada obračala hitreje od prve, ker slednja potiska drugo naprej in ne obratno.

Postopek, ki to upošteva, pravilno odčitava vrednosti kljub preskokom in zagotavlja zvezno in ponovljivo karakteristiko. Opisan je v nadaljevanju. Kot vhod sprejme



seznam vrednosti posameznih števk, ki smo jih dobili v predhodnem poglavju, njegov rezultat pa je odčitek števca.

***Postopek združevanja vrednosti posameznih števk v končni rezultat***

1. Inicializacija postopka: spremenljivko "Utež\_števke" nastavimo na  $0 \cdot 1$ , logično spremenljivko "Prenos" nastavimo na DA, celoštevilsko "Delni\_prenos" pa na 0;
2. vrednost skrajno desne števke (kolesček z najmanjšo utežjo) prepisemo v spremenljivko "Trenutna\_števka";
3. spremenljivki "Trenutna\_števka" določimo celoštevilčni in decimalni del ter ju vpišemo v spremenljivki "Celi\_del" in "Decimalni\_del";
4. če je to prva (t.j. skrajno desna) števka, jo pomnožimo s spremenljivko "Utež\_števke" ter vpišemo v spremenljivko "Rezultat\_odčitka". Nadaljujemo s korakom 19;
5. če ima spremenljivka "Prenos" vrednost NE, nadaljujemo s korakom 16;
6. če je "Delni\_prenos" večji od 0, ga zmanjšamo za 1;
7. če je vrednost spremenljivke "Zadnja\_števka" manjša od  $9 \cdot 7$  in "Delni\_prenos" enak 0, nadaljujemo s korakom 12;
8. če je "Decimalni\_del" večji ali enak od  $0 \cdot 3$ , nadaljujemo s korakom 18;
9. "Celi\_del" zmanjšamo za 1;
10. če je "Celi\_del" manjši od 0, ga nastavimo na 9 in postavimo "Delni\_prenos" na vrednost 2;
11. nadaljujemo s korakom 18;
12. če je vrednost spremenljivke "Zadnja\_števka" večja od  $9 \cdot 3$ , nadaljujemo s korakom 18;
13. če je "Decimalni\_del" manjši ali enak  $0 \cdot 7$  ali ima "Polovični\_prenos" vrednost različno od 0, nadaljujemo s korakom 18;
14. "Celi\_del" povečamo za ena in če ima potem vrednost večjo od 9, ga nastavimo na 0;
15. nadaljujemo s korakom 18;
16. če je "Decimalni\_del" manjši od  $0 \cdot 5$ , nadaljujemo s korakom 18;
17. "Celi\_del" povečamo za ena; če je njegova vrednost večja od 9, ga nastavimo na 0;
18. spremenljivko "Rezultat\_odčitka" povečamo za produkt med spremenljivkama "Vrednost\_števke" in "Celi\_del";

19. spremenljivko "Vrednost\_števke" pomnožimo z 10;
20. če ima spremenljivka "Trenutna\_števka" vrednost manjšo od 9`0 in ima "Polovični\_prenos" vrednost 0, nastavimo spremenljivko "Prenos" na NE;
21. vrednost "Trenutna\_števka" prepisemo v spremenljivko "Zadnja\_števka";
22. če je na seznamu še kaka števka, katere nismo upoštevali, vrednost naslednje številke vpišemo v spremenljivko "Trenutna\_števka" in se vrnemo na korak 3.

#### Komentar postopka:

- v koraku 3 razbijemo vrednost posamezne številke na celi in decimalni del. Prvi prispeva vrednost h končnemu rezultatu (razen v primeru zadnje številke, ki prispeva svojo celotno vrednost), drugi pa nam pove, ali moramo vrednost številke prilagoditi zaradi preskoka oz. prezgodnjega prenosa;
- spremenljivka "Prenos" se na začetku postopka nastavi na DA in ostane v tem stanju, dokler se ne pojavi prva števka, ki se ne nahaja v bližini vrednosti nič (to se testira v koraku 20). Do takrat moramo mehanske prenose upoštevati, od tu naprej pa ne več;
- spremenljivka "Delni\_prenos" indicira stanje, ko je vrednost predhodne številke večja od 9`7 in vrednost trenutne manjša ali enaka 0`3, t.j., ko je trenutna števka preskočila naprej. V primeru take situacije se delni prenos nastavi na 2 in z vsako številko zmanjša za 1, ker se mora vpliv preskoka upoštevati tudi pri naslednji dekadi, saj bi morala trenutna števka imeti vrednost, večjo od 9`7, če do prenosa ne bi prišlo. "Delni\_prenos" upoštevamo samo v primeru, da ima "Prenos" vrednost DA, saj v nasprotnem primeru prenosa ni potrebno upoštevati, ker smo vmes imeli dekada, ki se ni nahajala v bližini ničle;

Rezultat postopka je odčitek števca, ki je robusten in neobčutljiv na neidealno mehansko izvedbo števnika in katerega lahko brez nadaljnjih modifikacij uporabimo kot končni rezultat celotnega postopka odčitavanja.

## **13 Merilne karakteristike**

V predhodnih poglavjih smo opisali postopek optičnega odčitavanja električnih števcov. Na tem mestu pa nas zanimajo merilne karakteristike sistema, zlasti natančnost in hitrost odčitavanja. Poznavanje teh parametrov je nujno potrebno za uspešno rabo v industrijskem okolju, saj brez njih ne moremo ustrezno načrtovati integracije sistema v proizvodni proces.

V tem poglavju bomo opisali naš pristop k izvajanju meritev (poglavje 13.1) in določanju karakteristik sistema (poglavje 13.2) ter s pomočjo rezultatov meritev (poglavje 13.3) ocenili točnost odčitkov ter hitrost odčitavanja (poglavje 13.4).

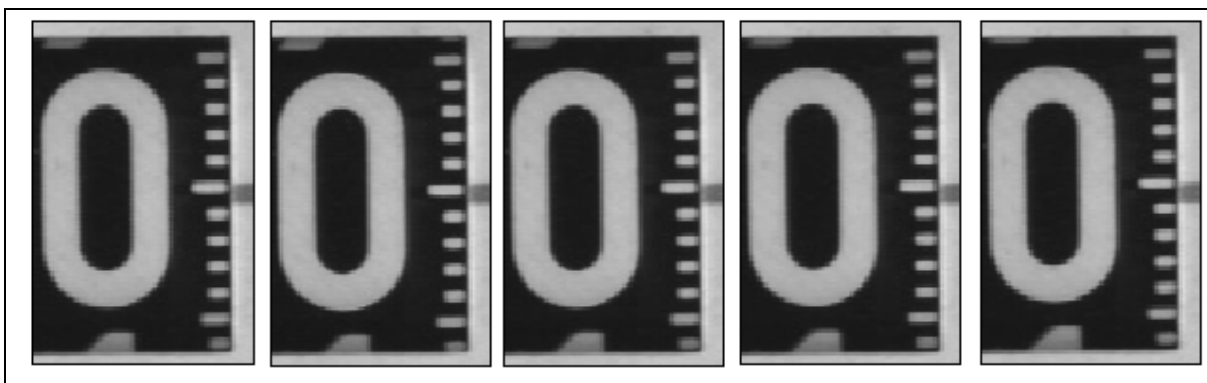
### **13.1 Izvajanje meritev**

Najvažnejši parameter sistema, kateremu smo namenili tudi največ pozornosti, je natančnost odčitavanja. Ocenjevanje le-tega je potekalo na dva načina. Najprej smo se prepričali, da je izhod sistema zvezna funkcija vrednosti števnika (poglavje 13.1.1) in je pogrešek odčitka manjši od enega razdelka (kar smo si v poglavju 4.2 zadali kot cilj). Kljub temu, da smo že na tej stopnji ugotovili zadovoljivo delovanje sistema, smo skušali natančneje določiti, kakšen je dejanski pogrešek odčitka ter kako nanj vplivajo različni pogoji, pod katerimi lahko odčitavanje izvajamo (poglavje 13.1.2).

#### **13.1.1 Meritve s števnikom, nastavljanim s koračnim motorjem**

Natančnost odčitavanja smo v grobem ocenili tako, da smo števniku (katerega začetno stanje smo poznali, t.j. odčitali s prostim očesom) s pomočjo koračnega

motorja počasi spreminjali vrednost ter spremljali rezultate merjenja z našim sistemom. Mehanski prenos med koračnim motorjem in števnikom nam je zagotavljal petindvajset korakov za vsak razdelek, kar bi v idealnih razmerah zadostovalo za zadovoljivo določitev točnostnega razreda sistema. Neidealna izvedba števnik pa ima za posledico relativno velike napake nastavitve zadnjega koleščka, predvsem zato, ker se koračni motor obrača sunkovito in se kolešček odbija ter se znotraj praznega hoda zobnikov, ki ga poganjajo, nastavlja nepredvidljivo (slika 76).



**Slika 76: Sekvenca nastavljanja koleščka s pomočjo koračnega motorja**

Na sliki 76 vidimo zaporedje petih korakov, ki si kronološko sledijo z leve proti desni. Na drugi in zadnji sliki lahko opazimo, da se je kolešček premaknil nazaj glede na predhodni položaj. Na podlagi kvalitativnih opazovanj ocenjujemo, da največja netočnost nastavitve ne presega polovico razdelka, zato zaključujemo, da je izvedba pozicioniranja števnik s pomočjo koračnega motorja zadovoljiva za grobo oceno točnosti sistema.

Sistem smo testirali tako, da smo s tako izvedeno nastavitvijo števnik opravili nekaj milijonov odčitkov v zaporedju (kar je enakovredno cca. tisoč obratom zadnjega koleščka) ter rezultate primerjali s pričakovanimi vrednostmi. Primerjali smo celotni rezultat odčitka in ne samo vrednost zadnjega koleščka, ker smo se želeli prepričati tudi v ustreznost združevanja vrednosti posameznih števk v končni rezultat (poglavje 1). Nezvezno in nepredvidljivo nastavljanje koleščkov števnik je bilo kot

nalašč za ta namen, saj so sunki koračnega motorja nezvezno premikali več koleščkov hkrati in s tem ustvarili večje preskoke (poglavje 12.1), kot se tipično pojavljajo v praksi. Nobena meritev v sekvenci ni odstopala od pričakovane vrednosti za razdelek ali več, iz česar sklepamo, da sistem izpolnjuje zahtevo po točnosti, kot smo jo opredelili v poglavju 4.2.

### **13.1.2 Meritve s fantomsko obremenjenim števcem**

Večjo točnost nastavljanja števnikar smo skušali doseči z uporabo števca, ki je bil fantomsko obremenjen. Pri enakomerni obremenitvi števca se njegova kotva enakomerno vrti ter s tem počasi in v idealnih razmerah zvezno spreminja zasuk števnikar. Kvalitativna opazovanja so potrdila domnevo, da je nastavljanje koleščka s fantomsko obremenitvijo precej natančneje kot s koračnim motorjem, čeprav kaže manjše nezvezne preskoke zaradi trenja in prostega hoda. Kvalitativno ocenjujemo, da ne presegajo velikostnega reda petine razdelka.

Meritve smo izvajali tako, da smo nastavili zadnji kolešček števnikar na vrednost nič (položaj smo določili kvalitativno s pomočjo povečane slike koleščka), nato pa števec fantomsko obremenili in izvajali meritve, dokler se zadnji kolešček ni zasukal za celoten obrat (končni položaj smo zopet določili kvalitativno). Vmes smo predpostavili enakomerno kotno hitrost koleščka (kar ni čisto res, ker se hitrost spreminja zaradi nihanja omrežne napetosti ter segrevanja in ohlajanja pomožnih uporov, katere smo uporabljali v tokokrogu) ter z linearno interpolacijo določili pričakovano vrednost koleščka, katero smo primerjali z dobljenimi rezultati meritev.

Primerjali smo samo vrednost zadnje števkar, saj natančnost, s katero jo odčitamo, določa razred točnosti celotnega sistema (ob predpostavki, da se vrednosti ostalih števkar pravilno združujejo v končni rezultat, kar smo preverili tako, kot je opisano v predhodnem poglavju). Povprečen čas, ki ga porabimo za izvedbo posameznega odčitka smo ocenili tako, da smo čas celotnega zasuka koleščka delili s številom

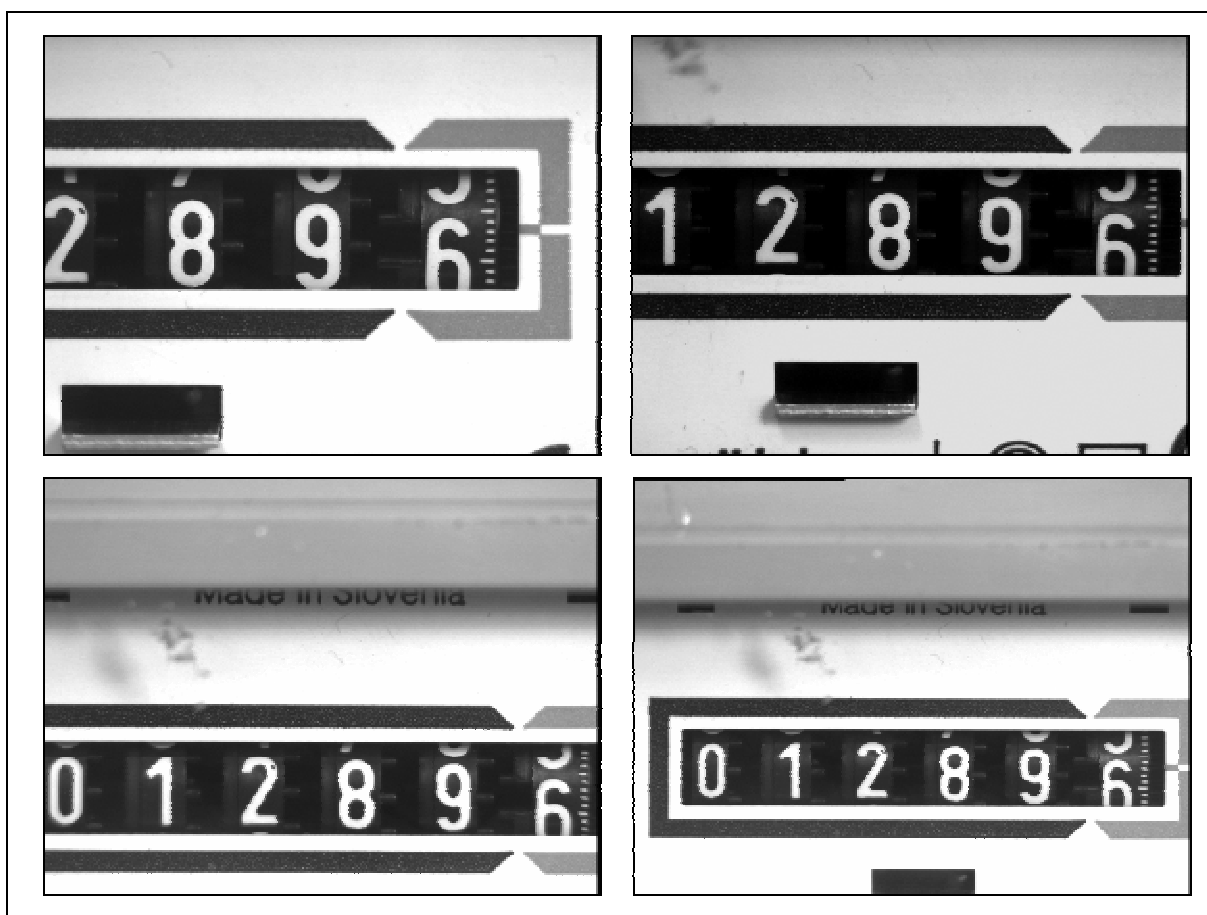
meritev, ki smo jih med tem izvedli. Meritve smo ponavljali pod različnimi pogoji zajema slike števca ter trakovi z različnimi višinami števk na njih, kot bo opisano v nadaljevanju.

## **13.2 Ocenjevanje karakteristik sistema**

Določanje merilnih karakteristik sistema je obsežen problem, saj je število parametrov, pod katerimi bi morali sistem testirati, veliko. Tako lahko spreminjamo povečavo števca, rotacijo števnika, velikost števk na traku in povečavo števnika, pri kateri zajemamo številke za izgradnjo traku. Natančnost sistema je v splošnem funkcija vseh štirih veličin, ki med seboj niso neodvisne. To pomeni, da bi morali izvesti veliko število meritev, če bi hoteli ugotoviti celotno karakteristiko sistema. Npr. pri štirih različnih povečavah, petih rotacijah, sedmih velikostih števk na traku ter štirih povečavah, pod katerimi zajemamo številke za izgradnjo traku, bi morali izvesti petstošestdeset različnih meritev.

Da zmanjšamo kompleksnost naloge, smo testirali samo manjšo podmnožico možnosti, za katere pričakujemo, da se bodo v praksi uporabljale najpogosteje. To so štiri različne povečave števca, pri katerih znaša višina števnika (poglavje 8.2) 170, 150, 130 in 110 slikovnih elementov (slika 77). Pri teh povečavah zajemamo slike za izgradnjo traku ter nato tudi odčitavamo vrednosti.

Zajemi pri teh velikostih so najbolj tipični in pokrivajo široko področje uporabe, saj že pri višini števnika 170 slikovnih elementov ne moremo zajeti več kot štiri v celoti vidne številke; višina števnika 110 slikovnih elementov pa nam omogoča številnik s šestimi dekadami zajeti v celoti. Potreba po delovanju sistema izven tega intervala povečav je malo verjetna.



**Slika 77: Testirane povečave števnik (višine: 170, 150, 130 in 110 slik. elem.)**

Nadalje naredimo pri vseh štirih povečavah trakove z velikostjo števk 50, 70, 90, 110, 130, 150 in 170 slikovnih elementov ter z njimi izvajamo meritve. Na ta način skušamo ugotoviti, kako zmanjšanje traku vpliva na točnost meritev in na hitrost odčitavanja. Manjši trakovi pospešijo meritve, ker je ugotavljanje vrednosti števk (poglavje 1) hitrejše zaradi manjših slik in s tem hitrejšega računanja razdalj med njimi. Pričakujemo pa manj natančne odčitke, ker manjši trakovi manj verodostojno reproducirajo sliko koleščka.

Na slikah od 108 do 111 v dodatku A si lahko ogledamo trakove vseh omenjenih velikosti, ki smo jih dobili s pomočjo števk, zajetih pri štirih testnih povečavah.

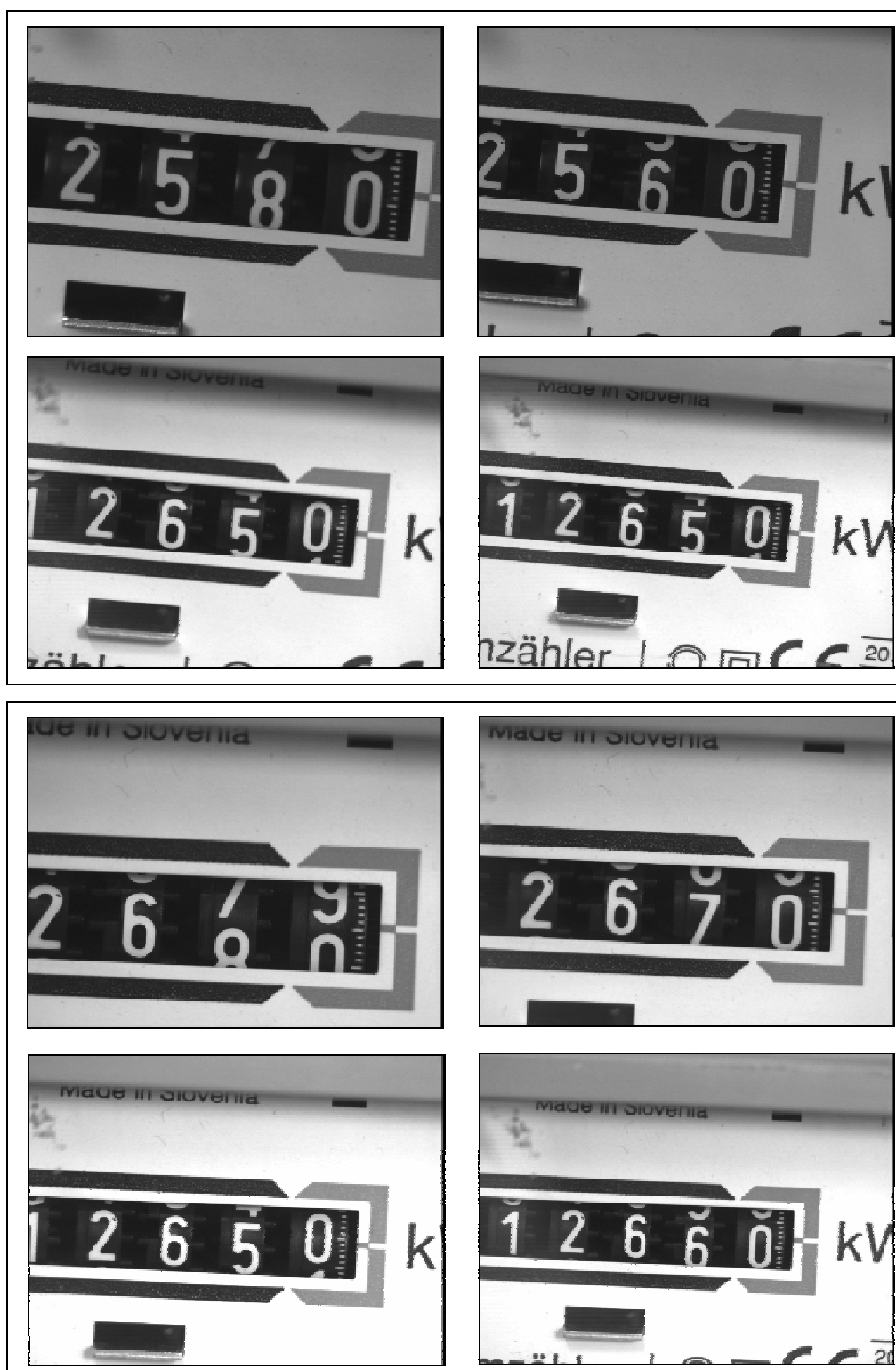
Opazimo, da kvaliteta trakov ne upada izrazito z manjšanjem velikosti števnik ob zajemu slik za izgradnjo. Počasno manjšanje kvalitete lahko zaznamo s podrobnim opazovanjem posameznih števk na traku. Tako je npr. številka "ena" prvega traku na sliki 108 zelo lepa in ima skoraj povsem gladek rob. Na slikah od 109 do 111 pa lahko opazujemo postopno upadanje kvalitete ter večanje netočnosti oblike in nazobčanosti robov.

Z manjšanjem višine števk na traku se količina podatkov, ki jih trak vsebuje, močno zmanjša in precej detajlov izpade. Kljub temu pa so vsi trakovi na slikah od 108 do 111 uporabni za odčitavanje vrednosti, kot bomo videli v nadaljevanju.

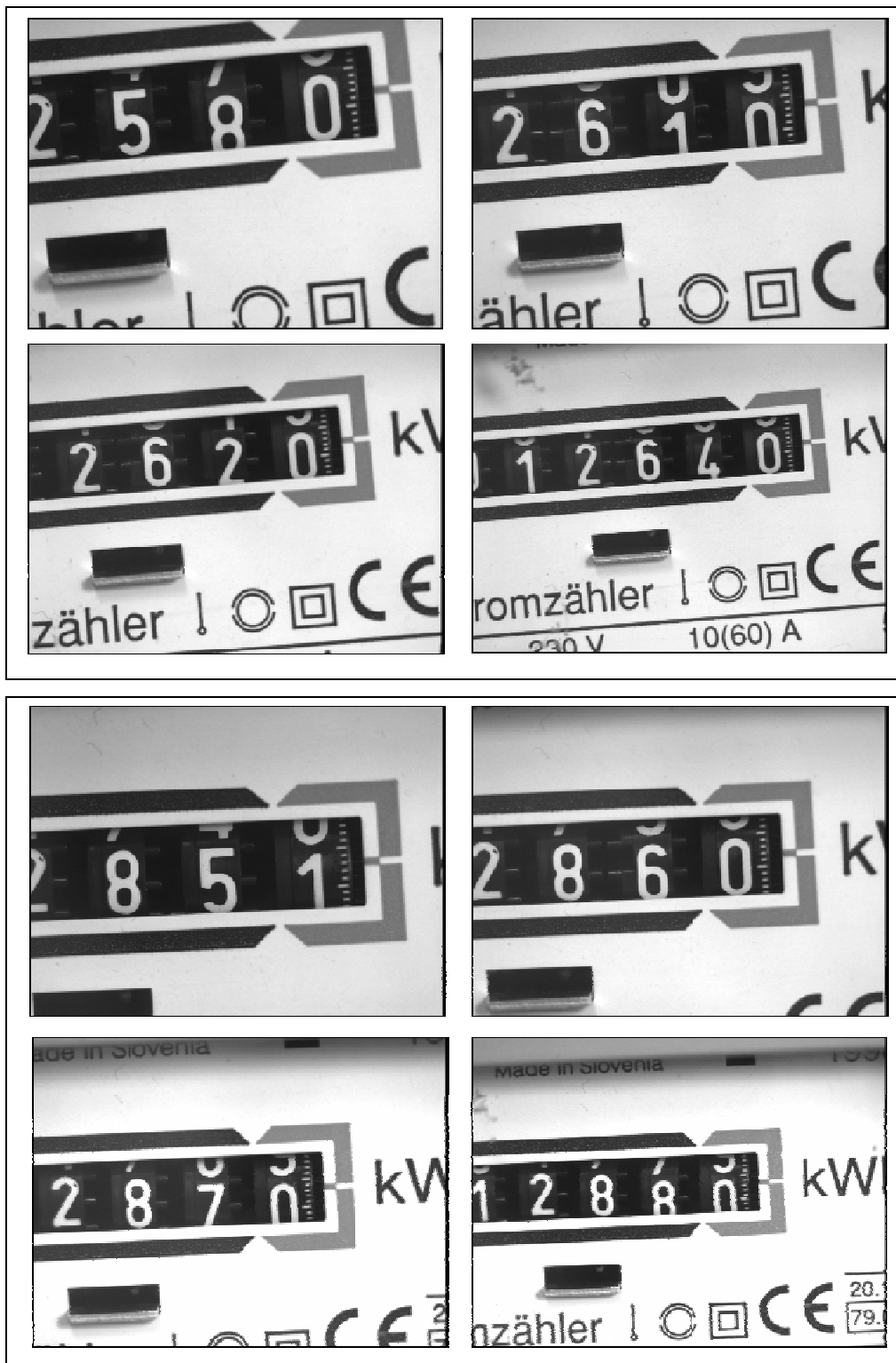
Slike od 112 do 115 prikazujejo iste trakove razporejene po posameznih slikah glede na višino števk na traku. Trakovi so neodvisno od njihove dejanske višine povečani tako, da zasedajo celotno stran. S teh slik lahko opazujemo zmanjšanje kvalitete trakov v odvisnosti od višine števk na njih. Vidimo, da so številke na traku čedalje manj natančne, kljub temu pa dobro ohranjajo topologijo koleščka (vidimo, da se meje posameznih števk na traku ne spreminjajo dosti; ta lastnost je zelo pomembna pri natančnem določanju vrednosti števk – poglavje 11.1). Posledica tega je, da so tudi trakovi z manjšo višino številčnice povsem uporabni za odčitavanje vrednosti, čeprav so odčitki nekoliko manj natančni (poglavje 13.4).

Meritve pri različnih velikostih števk na traku izvajamo samo pri zasuku števnik (poglavje 8.2) nič stopinj (slika 77). Sistem smo testirali tudi pri zasukih  $-5^\circ$ ,  $-2\cdot5^\circ$ ,  $+2\cdot5^\circ$  in  $+5^\circ$  (sliki 78 in 79), pri čemer pomeni negativni predznak zasuk sourni, pozitiven pa protiurni. Pri vsakem zasuku smo zajeli števec pri enakih testnih povečavah, kot smo izvajali predhodne meritve, pri tem pa je bila velikost števk na traku vedno enaka velikosti števk na zajeti sliki. S temi meritvami skušamo ugotoviti, koliko na odčitek števca vpliva rotacija števnik na sliki.





Slika 78: Zajemi števnik pri zasuku sournu 5° (zgoraj) in 2'5° (spodaj)



Slika 79: Zajemi števnika pri zasuku protiurno  $5^\circ$  (zgoraj) in  $2 \cdot 5^\circ$  (spodaj)

## 13.3 Rezultati meritev

Na tem mestu pojasnjujemo rezultate meritev, ki smo jih izvedli pod predhodno opisanimi pogoji in jih podajamo v dodatku B. Za vsako meritev tabelarično podajamo njene glavne parametre, nato pa še grafični potek treh karakteristik v odvisnosti od položaja kolesčka: odčitane vrednosti zadnje števk, potek napake ter odstotek ujemanja kalibrirane slike števk s trakom, vse v odvisnosti od pričakovane oz. interpolirane vrednosti zadnje števk (poglavje 13.1.2).

Podani parametri so naslednji (v enačbah, ki parametre pojasnjujejo, pomeni  $V_p$  pričakovano,  $V_i$  pa izmerjeno vrednost):

- **št. meritev (N):** število odčitkov, ki smo jih izvedli med odčitavanjem celotnega obrata zadnjega kolesčka števnika;
- **celotni čas (T):** čas, ki ga je kolesček potreboval, da se je v celoti zavrtel okoli svoje osi. Vrednost se nekoliko spreminja med posameznimi meritvami zaradi nestabilizirane fantomske obremenitve ter ne povsem natančnega določanja začetka in konca meritve s prostim očesom;
- **povpr. čas (P):** kvocient med celotnim časom meritve in številom odčitkov:  $P = T/N$ . Uporabljamo ga za oceno povprečnega časa trajanja posameznega odčitka;
- **maks. neg. odstop. (NO):** maksimalno odstopanje odčitane vrednosti od pričakovane v negativno smer:  $NO = -\min(V_p - V_i); \forall (V_p - V_i) < 0$ ;
- **maks. poz. odstop. (PO):** maksimalno odstopanje odčitane vrednosti od pričakovane v pozitivno smer:  $PO = \max(V_p - V_i); \forall (V_p - V_i) > 0$ ;
- **maks. napaka (AN):** največje odstopanje odčitane vrednosti od pričakovane (absolutna napaka odčitka):  $AN = \max(NO, PO)$ ;
- **min. % ujemanja (U):** odstotek točk, v katerih sta se števk in trak razlikovala v položaju najboljšega ujemanja.

## 13.4 Ocenjene karakteristike sistema

S pomočjo rezultatov meritev, ki jih podajamo v dodatku B, bomo sedaj ocenili merilne karakteristike našega sistema. Kot smo že omenili, se bomo omejili na natančnost in hitrost odčitavanja električnega števca, zajetega pod različnimi pogoji in z različno visokimi slikami števk na traku.

Povzetek karakteristik podajamo s pomočjo tabel in grafov, ki prikazujejo spreminjanje parametrov sistema v primeru zajema števca pri različnih povečavah in velikostih števk na traku ter rotaciji števnikar nič stopinj (poglavje 13.4.1). Vpliv rotacije na meritve podajamo v poglavju 13.4.2.

### 13.4.1 Merilne karakteristike sistema pri rotaciji števnikar nič stopinj

V tabelah od 6 do 9 so prikazane merilne karakteristike sistema pri zajemih števnikar z višinami 170, 150, 130 in 110 slikovnih elementov. Posamezni stolpci kažejo vrednosti parametrov v odvisnosti od višine števk na traku.

Pomen posameznih parametrov je naslednji:

- **maks. abs. napaka (MAN):** največja absolutna napaka pri celotnem obratu odčitavanja vrednosti koleščka:  $MAN = \max(AN)$ . Napaka, ki je enaka ali večja od 0,01 kWh, pomeni prekoračitev zahtev, specificiranih v poglavju 4.2;
- **rezerva do ene črtice % (RR):** odstotek, za kolikor bi napaka morala biti večja, da sistem ne bi izpolnjeval zahtev o natančnosti odčitavanja (poglavje 4.2):

$$RR = \frac{0,01 - MAN}{0,01} \cdot 100;$$

- **min % ujemanja števke s trakom (MU):** najmanjši odstotek ujemanja slike zadnje števke s sliko traku pri odčitavanju vrednosti preko celotnega obrata koleščka:  $MU = \min(U)$ ;

- **povprečni čas odčitka (P):** ocena časa, ki ga sistem potrebuje za posamezni odčitek. Izračunali smo ga kot kvocient med časom, ki ga je kolesček števnik potreboval za polni obrat, in številom meritev:  $P = T/N$ ;
- **povprečni čas normiran z VT (NP):** povprečni čas, ki smo ga normirali z višino števka na traku ter skalirali tako, da je enak 1 pri višini 50 slikovnih elementov:

$$NP = \frac{P}{P_{50}} \cdot \frac{50}{\check{S}},$$
 pri čemer pomeni  $P_{50}$  povprečni čas odčitka s trakom višine

števka 50 slikovnih elementov (najmanjši trak, ki ga uporabljamo),  $\check{S}$  pa je dejanska višina števka na traku.

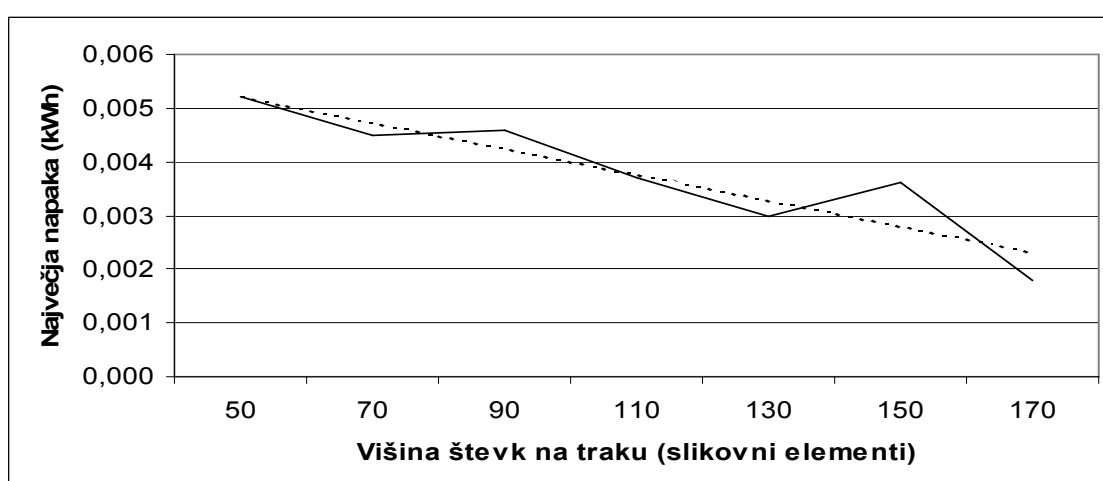
Iste podatke prikazujemo tudi grafično na slikah od 80 do 95. Grafom smo dodali funkcijo linerarnega trenda, s katero smo skušali nazorneje prikazati, ali določen parameter kaže koreliranost s spreminjanjem višine števka na traku.

Ocenjujemo, da sistem v vseh testiranih pogojih izpolnjuje zahteve po točnosti iz poglavja 4.2. Največja napaka, ki smo jo ugotovili, je znašala 0,0071 kWh oz. 71% razdelka. Pojavila se je pri višini števnik 150 slikovnih elementov in s trakom z višino števka 50 slikovnih elementov. Pri večjih in s tem preciznejših trakovih so bile napake manjše. Najmanjša napaka se je pojavila pri višini števnik in števka na traku 170 slikovnih elementov. Tu smo izmerili največjo napako 0,0018 kWh, kar je več kot petkrat bolje od zastavljenega cilja. Pri tem moramo imeti v mislih, da je povečava 170 slikovnih elementov zelo velika in z njo ne moremo odčitavati celotnega števca oz. zajeti vseh dekad. Zato je ta pogoj obratovanja primeren le za končni preizkus števec v proizvodnji, kjer je potrebno zajeti samo tri do štiri številke. Pri zajemu celotnega števca z manjšo povečavo, kjer je števnik visok 110 slikovnih elementov, lahko dosežemo zadovoljivo natančnost, če traku pretirano ne manjšamo. Tako lahko s trakovi od 70 do 170 slikovnih elementov dosežemo vsaj natančnost polovice razdelka (0,005 kWh). Iz prikazanih karakteristik vidimo, da je bolje uporabljati večje trakove, ki natančneje reproducirajo sliko raztegnjenega kolesčka, kar je pri

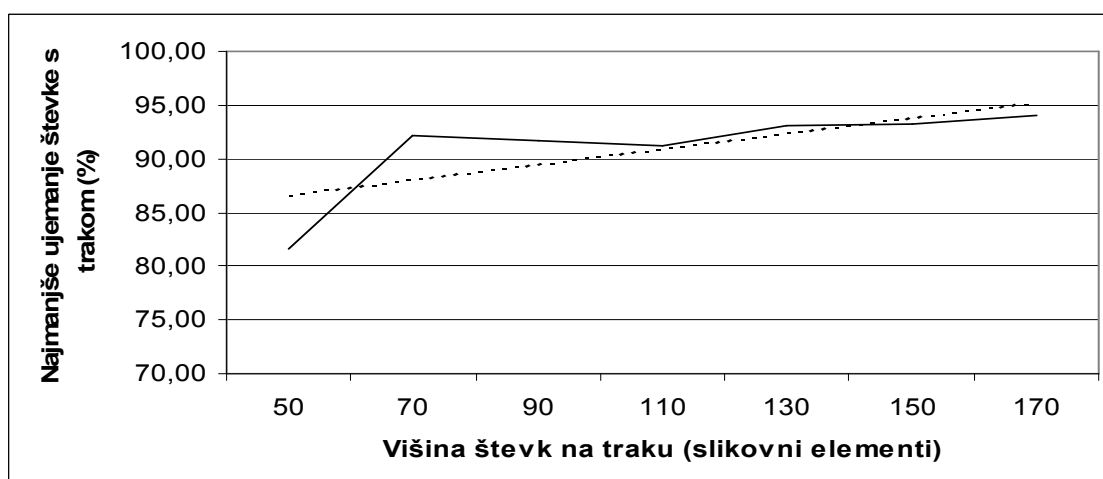
natančnem odčitavanju vrednosti zelo pomembno. S tem pa manjšamo hitrost odčitavanja, kar je z grafov ravno tako razvidno. Največjo hitrost odčitavanja smo dosegli pri zajemu števnik 110 slikovnih elementov in višini števk na traku 50 slikovnih elementov, torej pri kombinaciji parametrov, ki omogoča najmanj natančno odčitavanje. Dosegli smo čas odčitavanja manjši od 0,3 sekunde. Sistem je pri enaki povečavi zajema ter višini traku 170 slikovnih elementov potreboval največ časa 0,63 s, kar je več kot pri števniku in traku višine 170 slikovnih elementov. Razlog je v tem, da smo pri večji povečavi zajeli samo štiri števke, pri manjši pa vseh šest in proporcionalno temu se je moralo izvršiti tudi več primerjanj slik s trakom. Pri uporabi sistema lahko pričakujemo, da bo čas odčitavanja krajši od ene sekunde.

	Višina števk na traku (slikovni elementi)						
	50	70	90	110	130	150	170
Maks. abs. napaka	0,0052	0,0045	0,0046	0,0037	0,0030	0,0036	0,0018
Rezerva do ene črtice %	48,0	55,0	54,0	63,0	70,0	64,0	82,0
Min. % ujemanja števke s trakom	81,654	92,125	91,686	91,158	93,136	93,284	93,966
Povprečni čas odčitka (s)	0,322	0,349	0,367	0,413	0,447	0,504	0,567
Povprečni čas normiran z VT	1,000	0,774	0,633	0,583	0,534	0,522	0,518

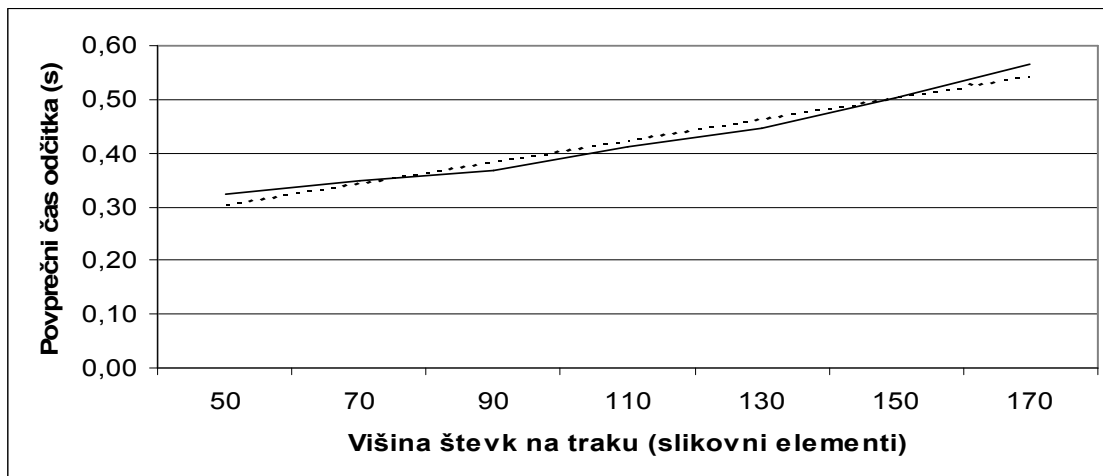
Tabela 6: Karakteristike sistema pri višini števnika 170 slikovnih elementov



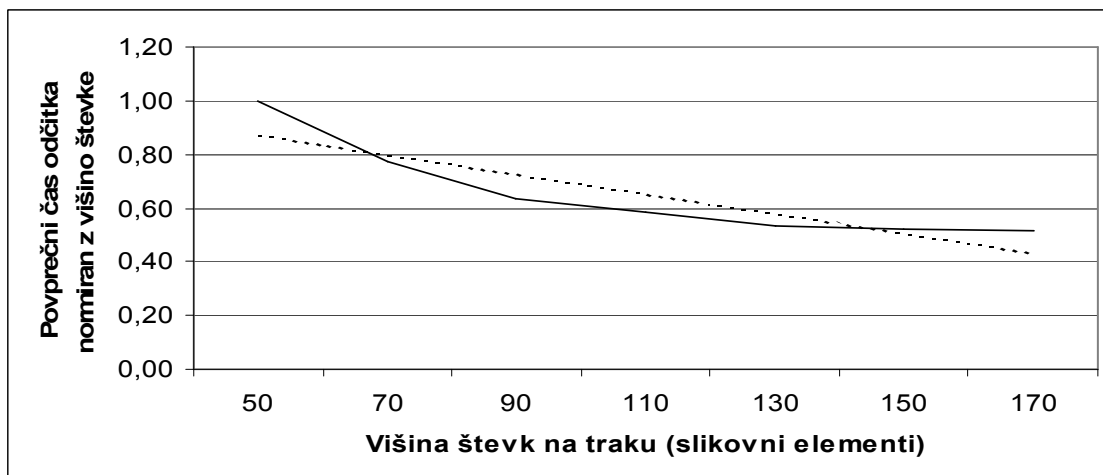
Slika 80: Največja napaka pri višini števnika 170 slikovnih elementov



Slika 81: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnika 170 slikovnih elementov



**Slika 82: Povprečno trajanje posameznega odčitka pri višini števnik 170 slikovnih elementov**

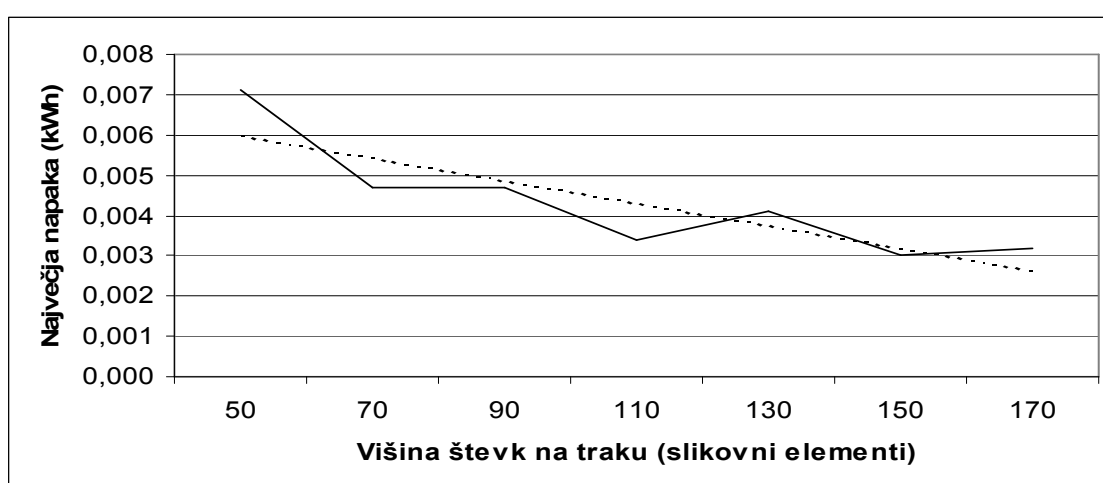


**Slika 83: Povprečno trajanje posameznega odčitka, normirano z višino števk na traku pri višini števnik 170 slikovnih elementov**

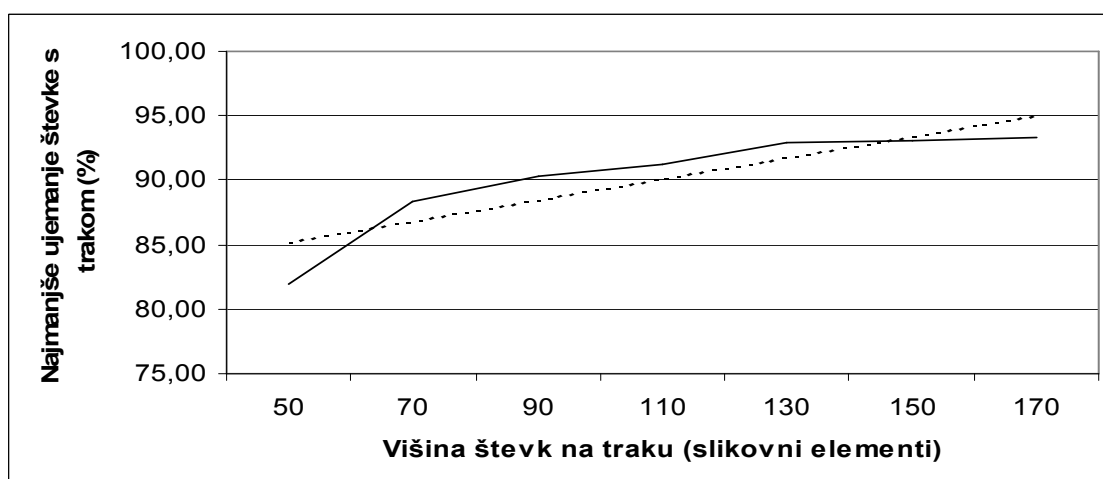


	Višina števk na traku (slikovni elementi)						
	50	70	90	110	130	150	170
Maks. abs. napaka	0,0071	0,0047	0,0047	0,0034	0,0041	0,0030	0,0032
Rezerva do ene črtice %	29,0	53,0	53,0	66,0	59,0	70,0	68,0
Min. % ujemanja števke s trakom	81,912	88,331	90,351	91,190	92,869	93,056	93,287
Povprečni čas odčitka (s)	0,296	0,320	0,329	0,349	0,369	0,409	0,429
Povprečni čas normiran z VT	1,000	0,772	0,617	0,536	0,479	0,461	0,426

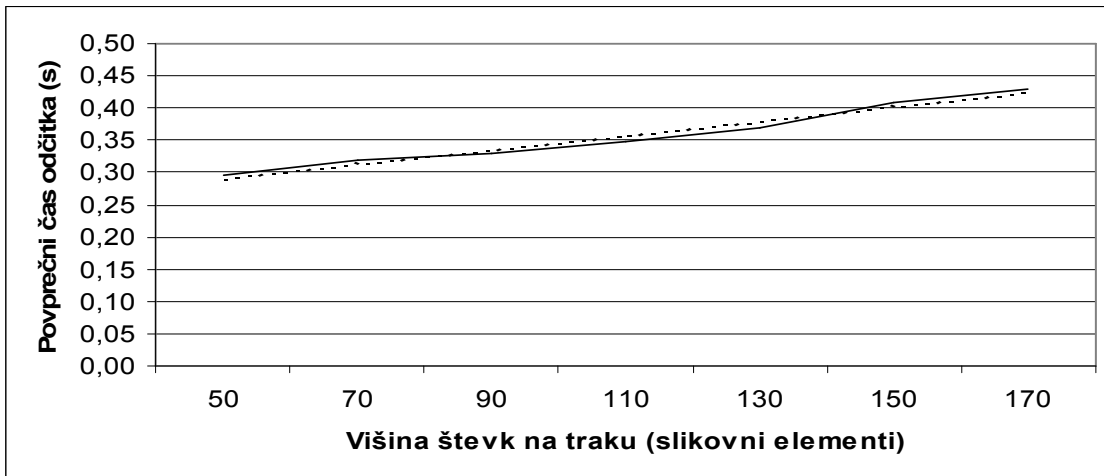
Tabela 7: Karakteristike sistema pri višini števnika 150 slikovnih elementov



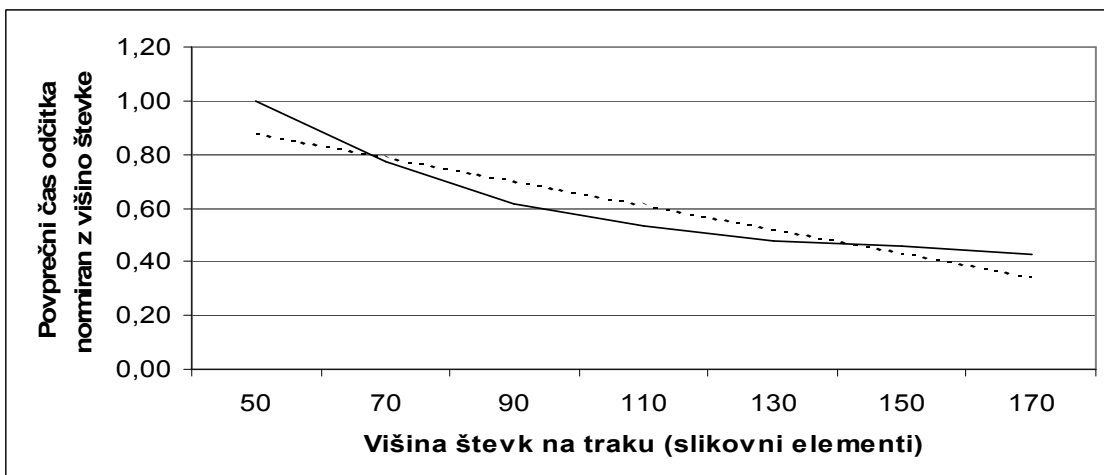
Slika 84: Največja napaka pri višini števnika 150 slikovnih elementov



Slika 85: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnika 150 slikovnih elementov



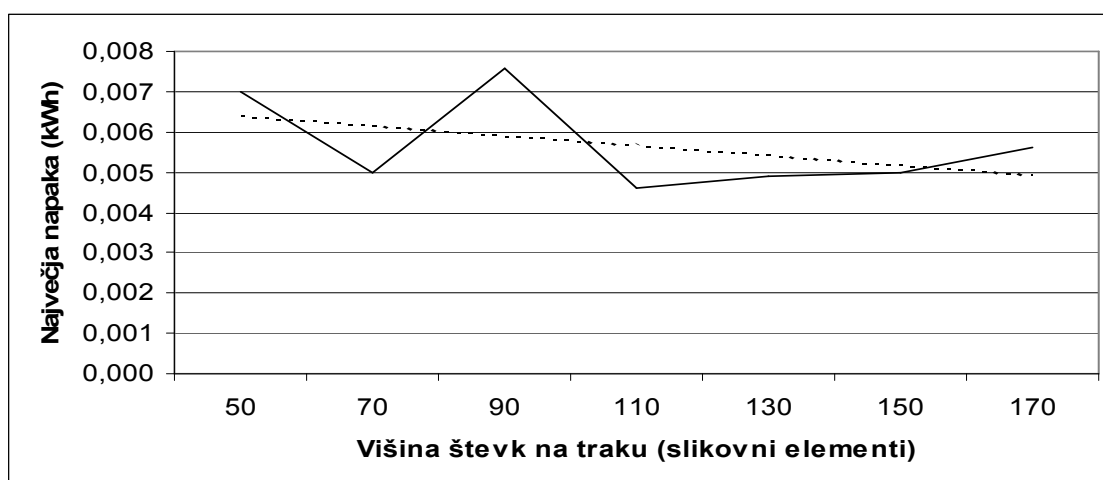
**Slika 86: Povprečno trajanje posameznega odčitka pri višini števnik 150 slikovnih elementov**



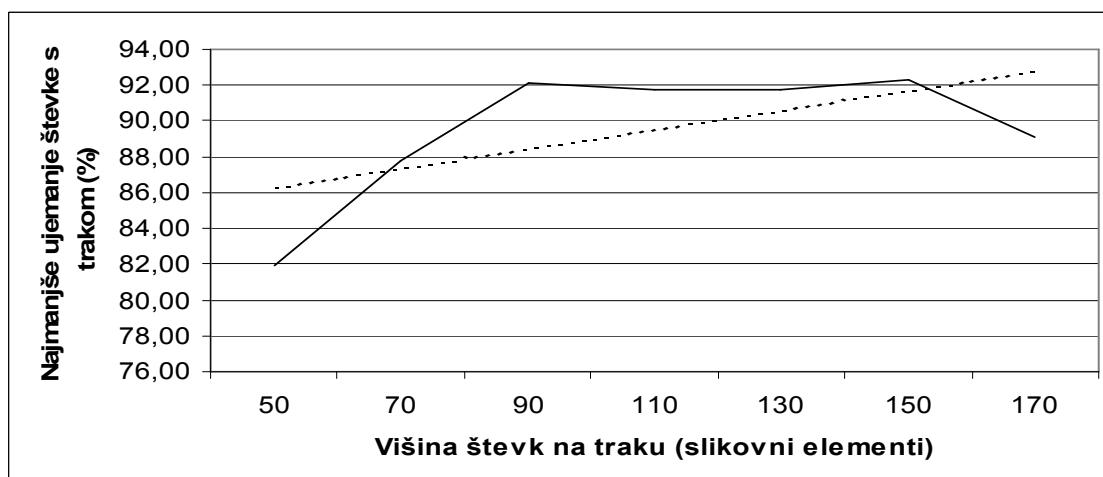
**Slika 87: Povprečno trajanje posameznega odčitka, normirano z višino števk na traku pri višini števnik 150 slikovnih elementov**

	Višina števk na traku (slikovni elementi)						
	50	70	90	110	130	150	170
Maks. abs. napaka	0,0070	0,0050	0,0076	0,0046	0,0049	0,0050	0,0056
Rezerva do ene črtice %	30,0	50,0	24,0	54,0	51,0	50,0	44,0
Min. % ujemanja števke s trakom	81,975	87,806	92,086	91,694	91,755	92,313	89,084
Povprečni čas odčitka (s)	0,281	0,323	0,356	0,376	0,460	0,538	0,478
Povprečni čas normiran z VT	1,000	0,821	0,704	0,608	0,630	0,638	0,500

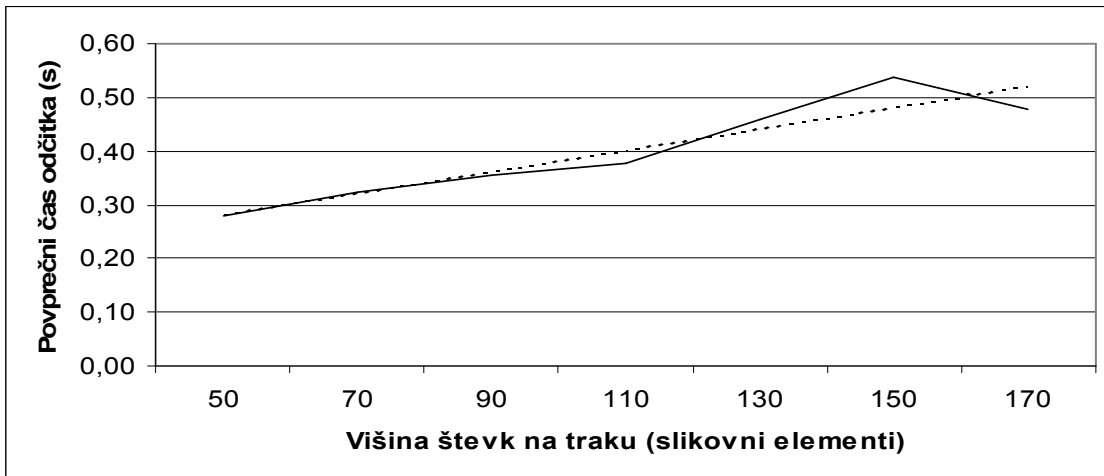
Tabela 8: Karakteristike sistema pri višini števnika 130 slikovnih elementov



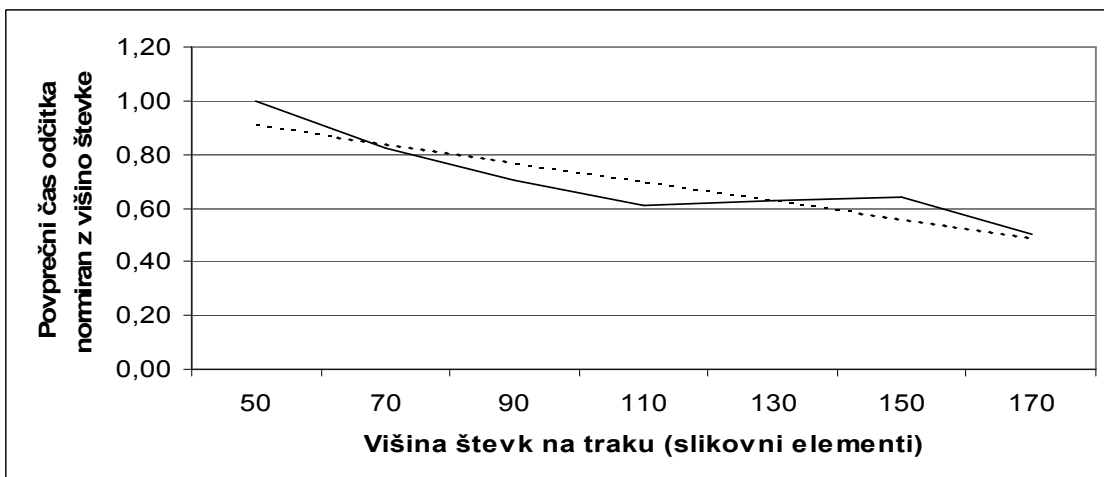
Slika 88: Največja napaka pri višini števnika 130 slikovnih elementov



Slika 89: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnika 130 slikovnih elementov



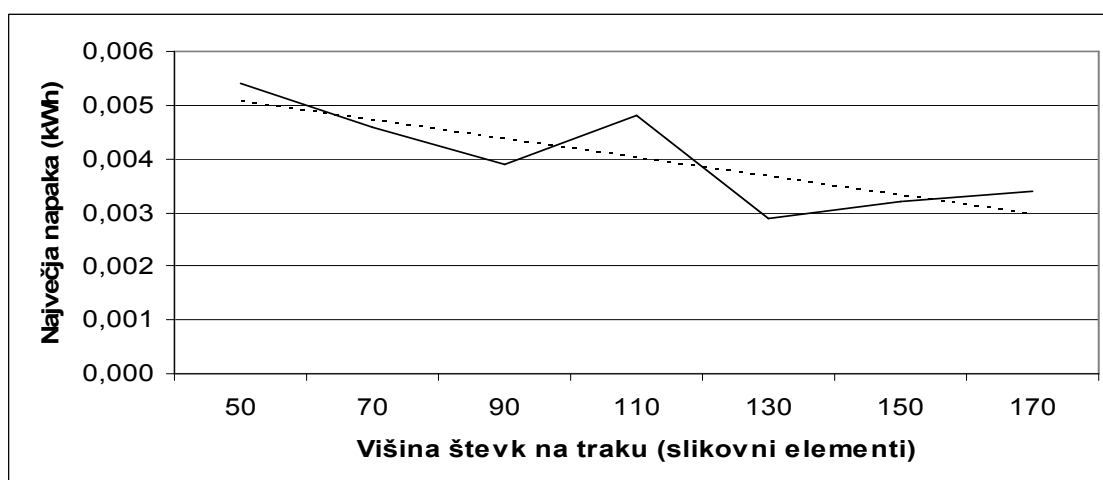
**Slika 90: Povprečno trajanje posameznega odčitka pri višini števnik 130 slikovnih elementov**



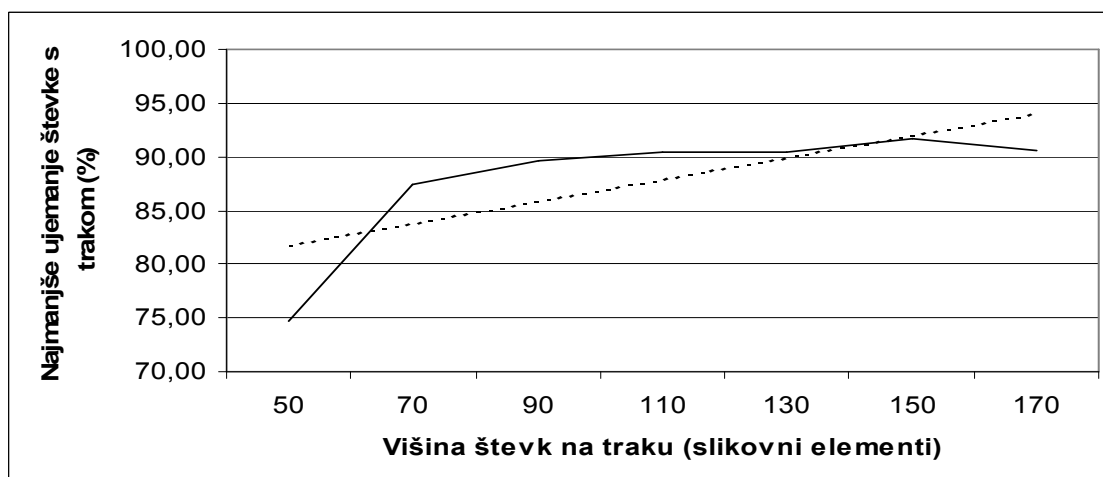
**Slika 91: Povprečno trajanje posameznega odčitka, normirano z višino števk na traku pri višini števnik 130 slikovnih elementov**

	Višina števk na traku (slikovni elementi)						
	50	70	90	110	130	150	170
Maks. abs. napaka	0,0054	0,0046	0,0039	0,0048	0,0029	0,0032	0,0034
Rezerva do ene črtice %	46,0	54,0	61,0	52,0	71,0	68,0	66,0
Min. % ujemanja števke s trakom	74,638	87,492	89,697	90,477	90,463	91,720	90,552
Povprečni čas odčitka (s)	0,255	0,276	0,319	0,370	0,466	0,501	0,629
Povprečni čas normiran z VT	1,000	0,773	0,695	0,660	0,703	0,655	0,725

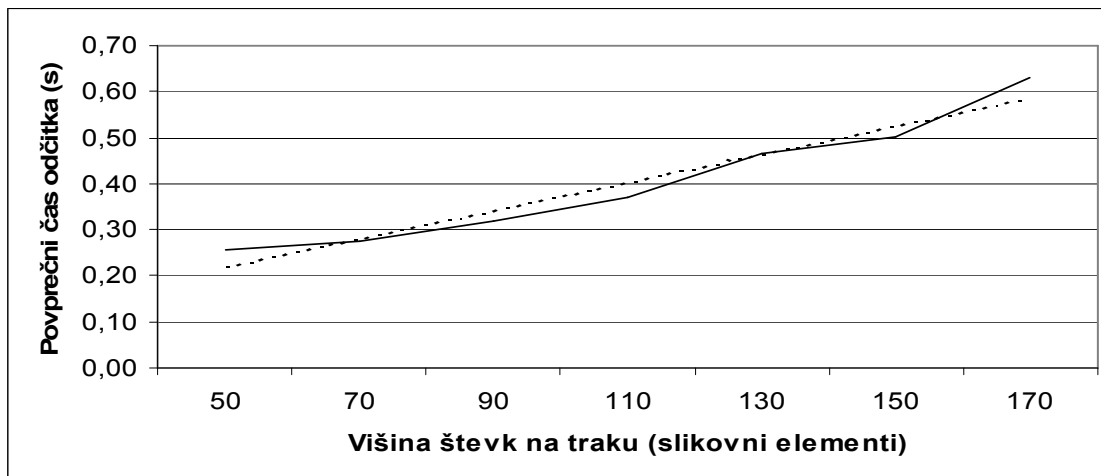
Tabela 9: Karakteristike sistema pri višini števnika 110 slikovnih elementov



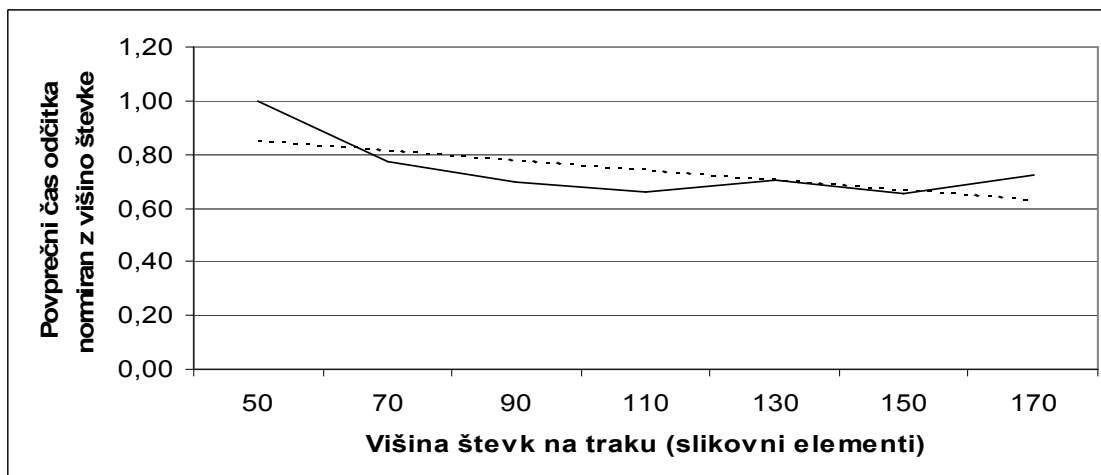
Slika 92: Največja napaka pri višini števnika 110 slikovnih elementov



Slika 93: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnika 110 slikovnih elementov



**Slika 94: Povprečno trajanje posameznega odčitka pri višini števnik 110 slikovnih elementov**



**Slika 95: Povprečno trajanje posameznega odčitka, normirano z višino števk na traku pri višini števnik 110 slikovnih elementov**

### 13.4.2 Merilne karakteristike sistema pri različnih rotacijah števnik

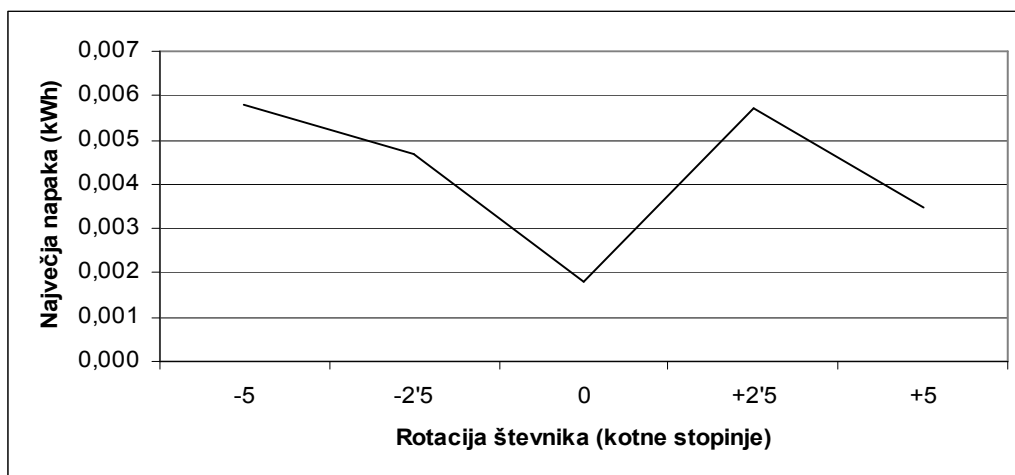
Podobno kot smo v prejšnjem poglavju prikazali karakteristike pri zajemih števca brez zasuka števnik, na tem mestu podajamo merilne lastnosti sistema v odvisnosti od rotacije števnik ob zajemu. Podatki so prikazani v tabelah od 10 do 13 ter grafično na slikah od 96 do 107.

Vidimo, da se ob prisotnosti zasuka števnik napake odčitkov v večini primerov nekoliko povečajo, kar smo pričakovali, saj to predstavlja dodatno spremembo slik števk, ki jo postopek poravnave (poglavje 1) ne more v celoti izničiti. V nekaterih primerih pa so bili rezultati celo boljši ob prisotnosti majhnih zasukov (višina števnik 130 slikovnih elementov in rotacija  $-2,5^\circ$ ). Sklepamo, da je to posledica neidealne izvedbe števnik, kjer kolesčki niso povsem pravokotni na vertikalne meje števca (ta pojav smo že omenili pri izgradnji traku v poglavju 1). Posledica tega je, da so številke že pri zajemu za izgradnjo traku nekoliko zasukane in ko zasukamo števnik pri odčitavanju vrednosti za podoben kot, se natančnost odčitavanja poveča. Iz tabel in grafov lahko vidimo, da zmanjšanje natančnosti odčitavanja ni usodno, saj nismo pri nobeni meritvi zaznali prekoračitve zahtevane natančnosti; smo pa pri višini števnik 130 slikovnih elementov in zasuku  $-5^\circ$  dopustno mejo skoraj dosegli. Zato pri preciznih odčitavanjih števnik priporočamo pogoje obratovanja, kjer je zasuk le-tega čim manjši.

Povprečni čas odčitka se ne spremeni bistveno v odvisnosti od rotacije. Čas, potreben za posamezni odčitek, kaže v večini primerov trend zmanjševanja za nekaj odstotkov ob prisotnosti rotacije. Razlog je v tem, da se z rotacijo pospeši detekcija vertikalnih mej števnik (poglavje 8.1), saj se ob prisotnosti zasuka zmanjšajo temna področja med števkami in s tem število točk, skozi katere računamo regresijsko premico. Sklepamo, da se ob prisotnosti rotacije hitrost odčitavanja poveča kvečjemu za nekaj odstotkov (višina števnik 150 slikovnih elementov, rotacija števnik  $+5^\circ$ ).

	Rotacija števnik (kotne stopinje)				
	-5	-2'5	0	+2'5	+5
Maks. abs. napaka	0,0058	0,0047	0,0018	0,0057	0,0035
Rezerva do ene črtice %	42,0	53,0	82,0	43,0	65,0
Min. % ujemanja števk s trakom	90,051	93,729	93,966	89,776	89,134
Povprečni čas odčitka (s)	0,426	0,396	0,567	0,379	0,401

Tabela 10: Karakteristike sistema pri višini števnik 170 slikovnih elementov

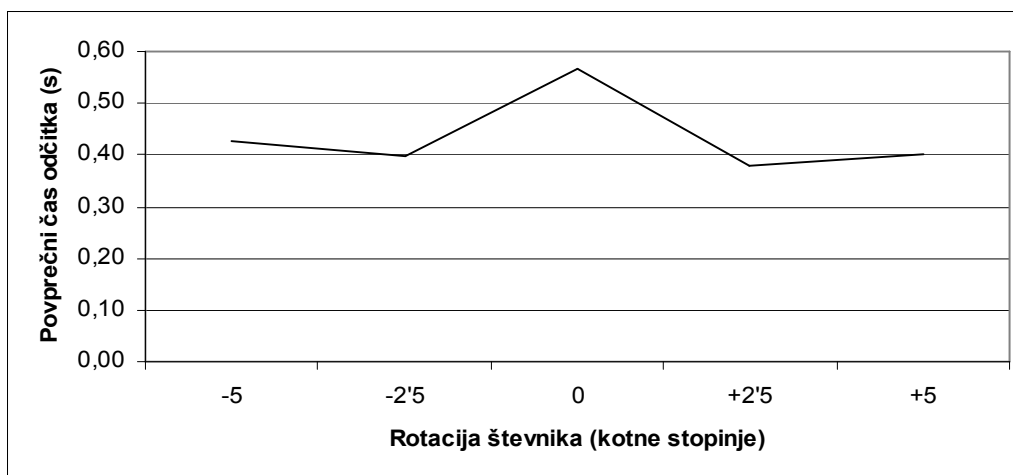


Slika 96: Največja napaka pri višini števnik 170 slikovnih elementov



Slika 97: Najmanjše ujemanje slike zadnje števk s trakom v odstotkih pri višini števnik 170 slikovnih elementov

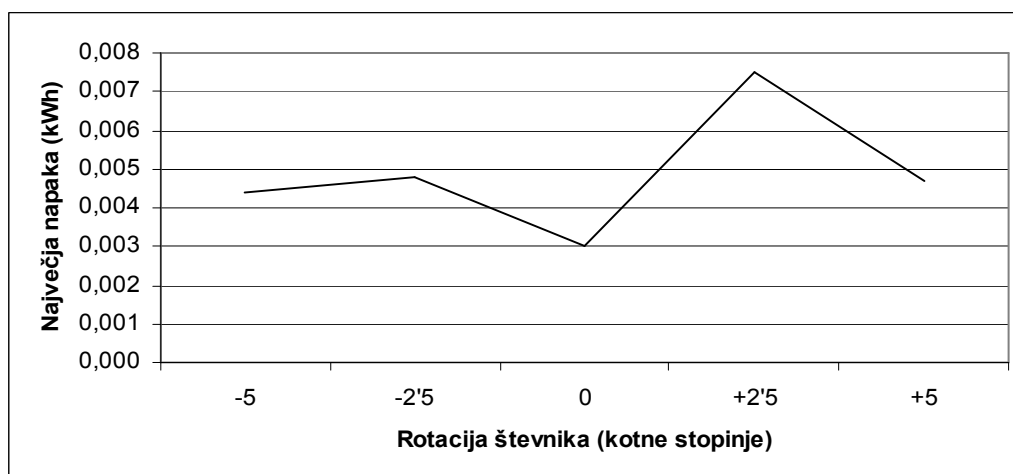




**Slika 98: Povprečno trajanje posameznega odčitka pri višini števnik 170 slikovnih elementov**

	Rotacija števnik (kotne stopinje)				
	-5	-2,5	0	+2,5	+5
<b>Maks. abs. napaka</b>	0,0044	0,0048	0,0030	0,0075	0,0047
<b>Rezerva do ene črtice %</b>	56,0	52,0	70,0	25,0	53,0
<b>Min. % ujemanja številke s trakom</b>	92,303	87,819	93,056	91,045	89,850
<b>Povprečni čas odčitka (s)</b>	0,322	0,366	0,409	0,334	0,469

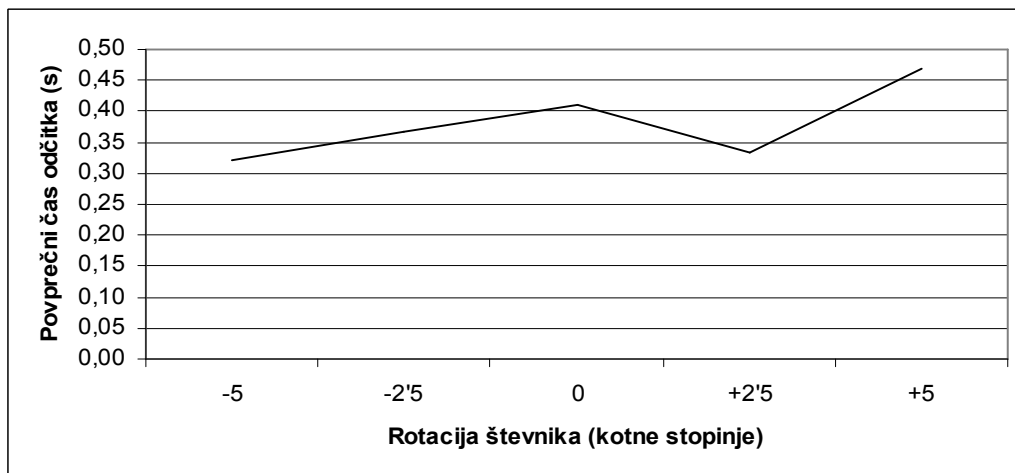
**Tabela 11: Karakteristike sistema pri višini števnik 150 slikovnih elementov**



**Slika 99: Največja napaka pri višini števnik 150 slikovnih elementov**



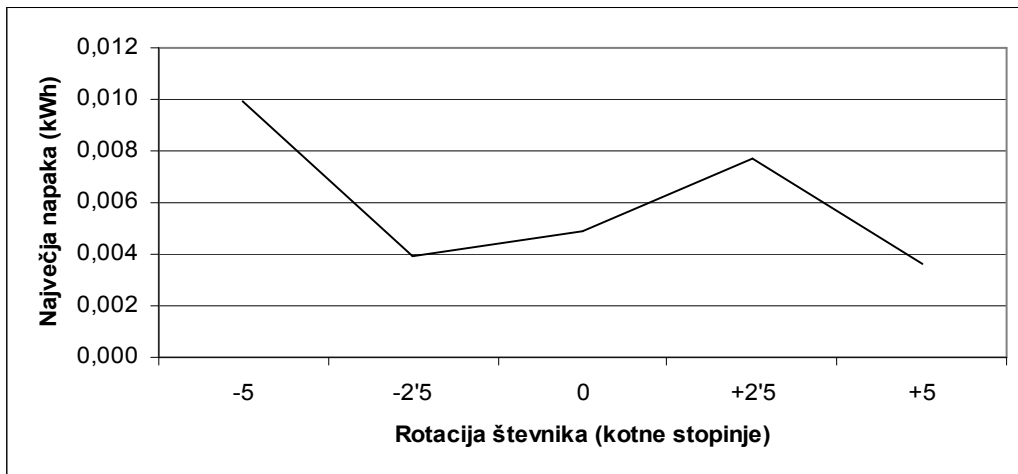
Slika 100: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnika 150 slikovnih elementov



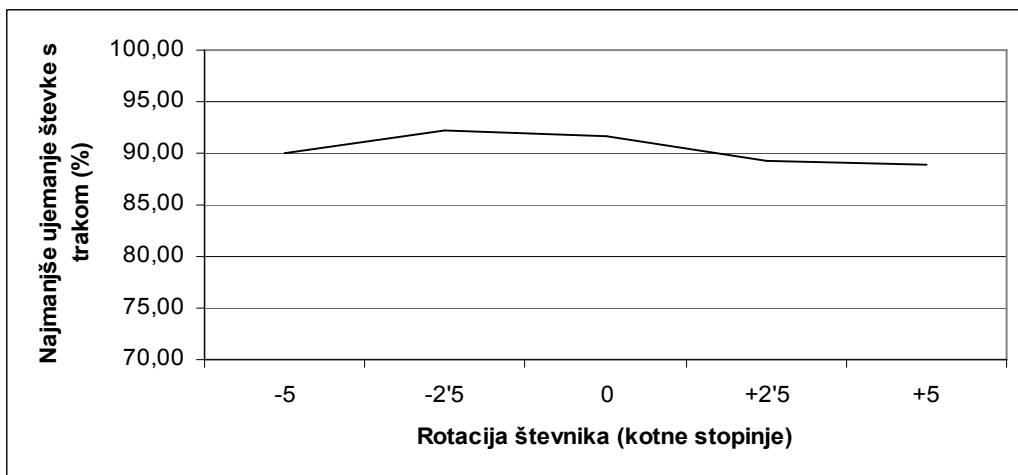
Slika 101: Povprečno trajanje posameznega odčitka pri višini števnika 150 slikovnih elementov

	Rotacija števnika (kotne stopinje)				
	-5	-2'5	0	+2'5	+5
Maks. abs. napaka	0,0099	0,0039	0,0049	0,0077	0,0036
Rezerva do ene črtice %	1,0	61,0	51,0	23,0	64,0
Min. % ujemanja števke s trakom	90,058	92,178	91,755	89,257	88,881
Povprečni čas odčitka (s)	0,315	0,324	0,460	0,308	0,318

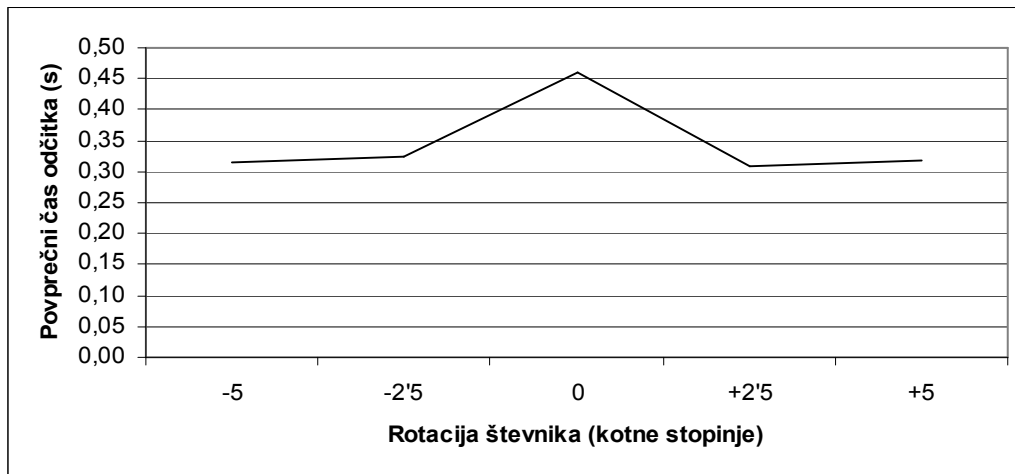
Tabela 12: Karakteristike sistema pri višini števnika 130 slikovnih elementov



**Slika 102: Največja napaka pri višini števnik 130 slikovnih elementov**



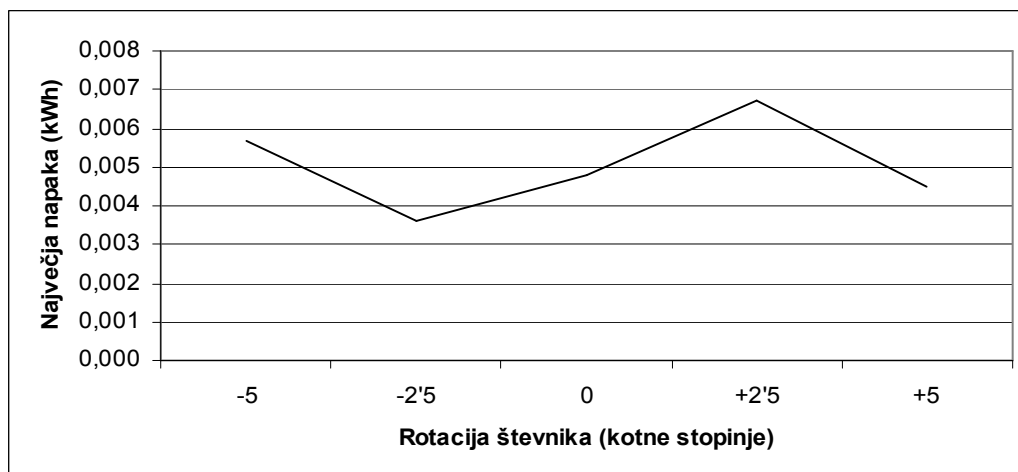
**Slika 103: Najmanjše ujemanje slike zadnje števke s trakom v odstotkih pri višini števnik 130 slikovnih elementov**



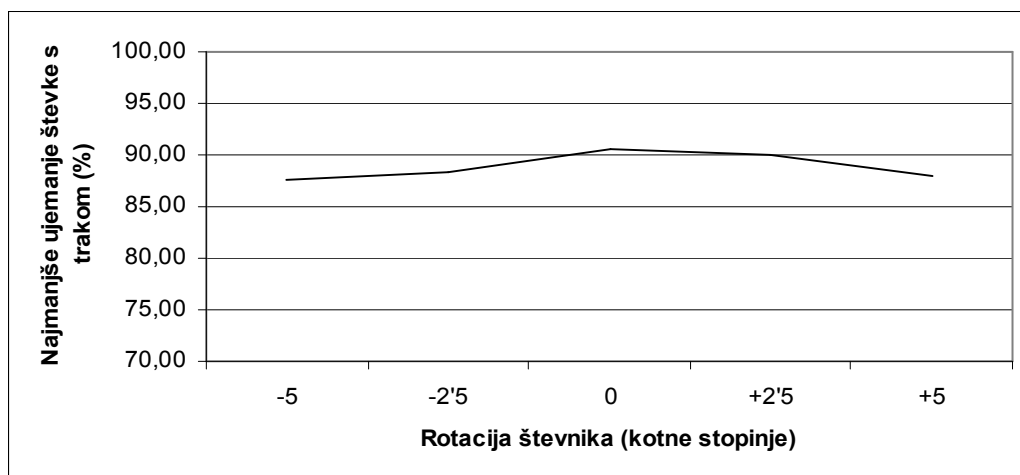
**Slika 104: Povprečno trajanje posameznega odčitka pri višini števnik 130 slikovnih elementov**

	Rotacija števnik (kotne stopinje)				
	-5	-2'5	0	+2'5	+5
<b>Maks. abs. napaka</b>	0,0057	0,0036	0,0048	0,0067	0,0045
<b>Rezerva do ene črtice %</b>	43,0	64,0	52,0	33,0	55,0
<b>Min. % ujemanja številke s trakom</b>	87,582	88,318	90,477	89,974	87,953
<b>Povprečni čas odčitka (s)</b>	0,322	0,349	0,367	0,413	0,374

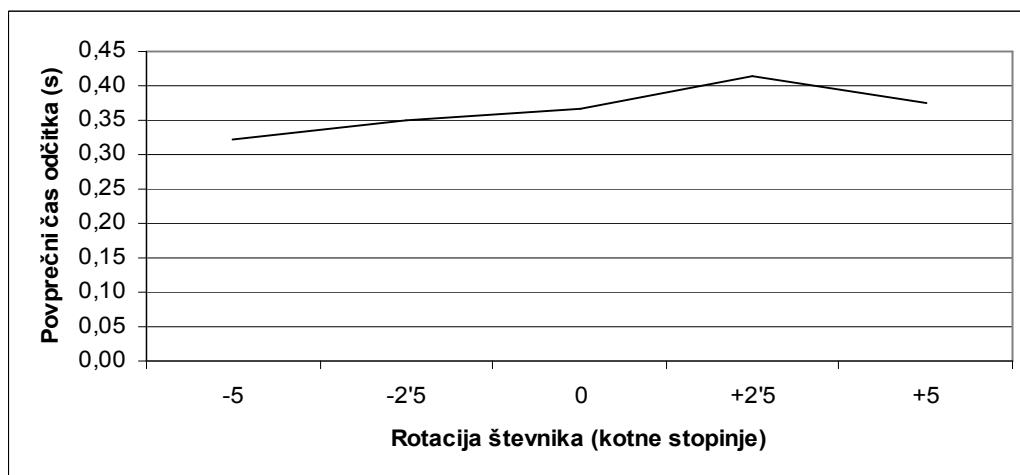
**Tabela 13: Karakteristike sistema pri višini števnik 110 slikovnih elementov**



**Slika 105: Največja napaka pri višini števnik 110 slikovnih elementov**



**Slika 106: Najmanjše ujemanje slike zadnje števk s trakom v odstotkih pri višini števika 110 slikovnih elementov**



**Slika 107: Povprečno trajanje posameznega odčitka pri višini števika 110 slikovnih elementov**



## 14 Zaključek

V tem poglavju podajamo strnjeno vsebino magistrske naloge ter sklepe, do katerih smo prišli na podlagi meritev in testiranj v laboratoriju.

Potreba po avtomatizaciji zajema vrednosti električnih števecv se je pojavila v proizvodnji teh naprav, kjer je v procesu končnega preizkusa potrebno njihove vrednosti odčitati dvakrat. To sedaj počnejo ljudje, ki morajo velikemu številu števecv (tipično se nahaja po štirideset števecv na enem nosilcu) relativno hitro odčitati vrednosti, pri čemer gledajo števnike s prostim očesom. To opravilo je naporno, utrudljivo, poleg tega so napake pri odčitavanju težko naknadno dokazljive. Z realizacijo sistema za avtomatski zajem vrednosti števecv se odpirajo možnosti za razbremenitev operaterjev v proizvodnji (poglavje 1).

Sistem avtomatskega odčitavanja električnih števecv je zasnovan od temeljev do končne realizacije s ciljem preproste integracije v industrijsko okolje. Tako je postopek odčitavanja v majhni meri odvisen od načina uporabe oz. implementacije zajema slik števecv ter izdaje rezultatov, s čimer ni potrebno spreminjati samega postopka odčitavanja med različnimi načini uporabe (poglavje 1).

Odčitavanje števnik je specifičen problem, ki ga ne moremo rešiti s klasičnimi postopki OCR, ker le-ti ne delujejo ustrezno, če znaki, ki jih razpoznavamo, niso vidni v celoti. Pri odčitavanju števnik so posamezni koleščki največkrat postavljeni v pozicijo, kjer prikazujejo vmesne položaje med števki, kar pri uporabi metod OCR ni sprejemljivo. Poleg tega z njihovo uporabo ne bi dobili dovolj natančnih odčitkov,

ker ti postopki kot rezultat vračajo le oznako razpoznanega znaka. V primeru pozicij koleščka, ko se številka nahaja v vmesnem položaju, želimo razbrati analogno (v idealiziranem pomenu) vrednost zasuka. V ta namen smo razvili postopek določanja vrednosti številke s pomočjo traku, ki predstavlja sliko raztegnjenega koleščka. Slika neznanе številke se primerja s trakom preko celotne njegove dolžine ter ugotovi mesto najboljšega prileganja, ki je neposredno korelirano z vrednostjo številke na sliki (poglavje 1).

Da bi bil sistem uporaben v industrijskem okolju, mora biti čim manj občutljiv na parametre zajema, kot so sprememba osvetljenosti, natančen položaj, povečava in rotacija številnika na sliki. V poglavju 1 opisujemo, kako s pravilno postavitvijo praga binarizacije zmanjšamo občutljivost na spremembe osvetljenosti. Robustna detekcija številnika (poglavje 1) je ravno tako nujna za uspešno uporabo v industriji, saj brez nje sistem ne more razpoznati, ali se številnik na zajeti sliki sploh nahaja oz. je primeren za odčitavanje vrednosti, in javiti napake v primeru, da temu ni tako. Nadaljevanje tega koraka je pravilna detekcija prisotnosti vseh števk na sliki ter natančno določanje povečave in zasuka številnika (poglavje 1), ker te podatke potrebujemo pri poravnavi slik števk (poglavje 1), s čimer zmanjšamo vpliv povečave in rotacije na odčitavanje vrednosti posameznih števk (poglavje 1).

Za odčitavanje vrednosti števk potrebujemo trak, ki predstavlja sliko raztegnjenega koleščka. Enostavnost uporabe sistema je odvisna tudi od tega, kako preprosto lahko trakove za posamezne tipe števecv zgradimo. V poglavju 1 opisujemo postopek izgradnje traku, ki zahteva samo dvajset slik koleščka (deset celih števk ter deset vmesnih položajev med dvema števkama), za katere je potrebno, da so zajete pri precizno pozicioniranem koleščku. S tem se odpira možnost hitrega in enostavnega učenja (prilagajanja) sistema na posamezni tip številca brez posebnih aparatov za precizno nastavljanje koleščka v ustrezen položaj.

Nazadnje smo izvedli množico meritev pod različnimi pogoji zajema, s katerimi smo želeli pokazati ustreznost sistema za uporabo v industrijskem okolju ter določiti



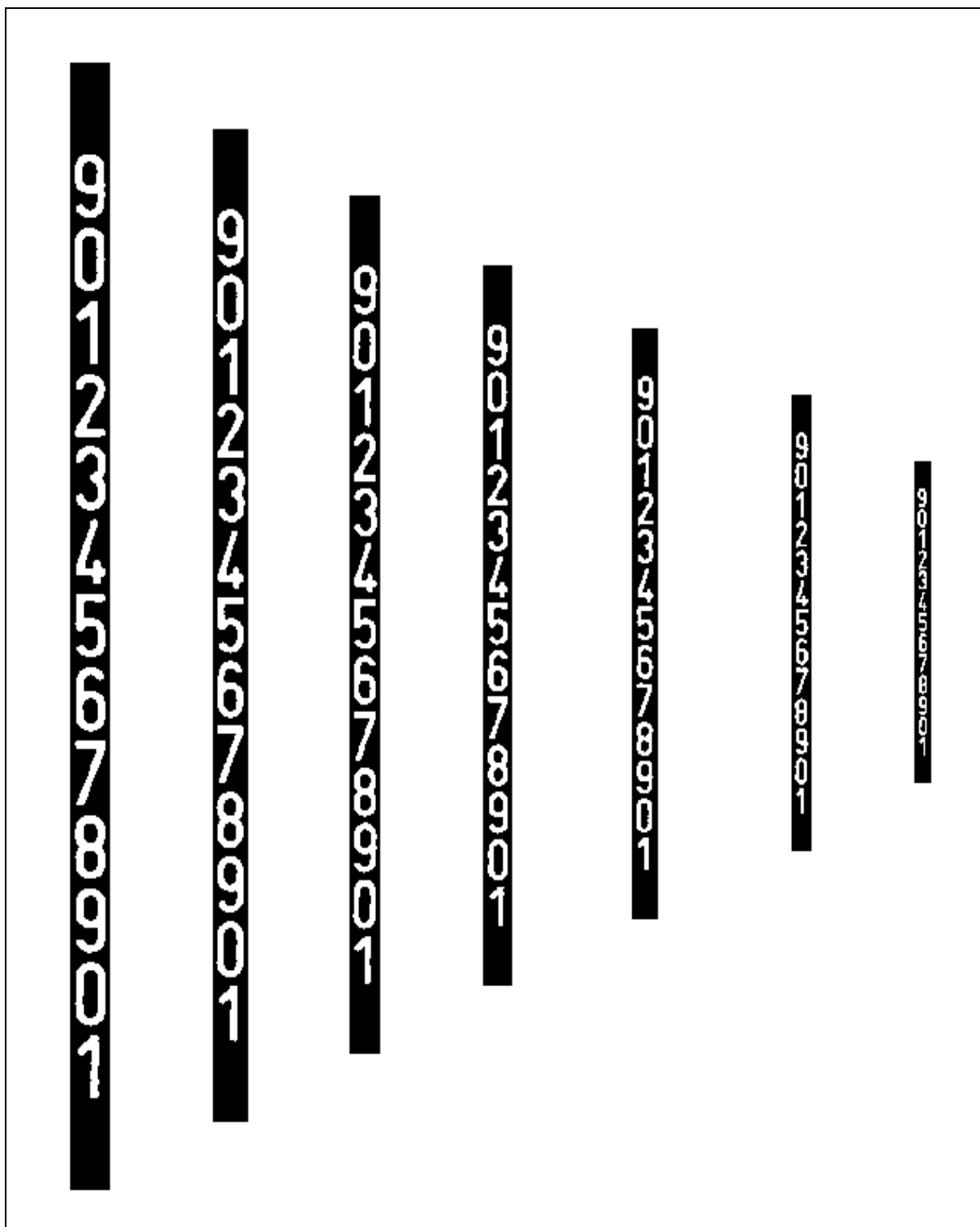
njegove merilne karakteristike (poglavje 1). Poskusi kažejo, da sistem izpolnjuje zahteve po natančnosti v relativno širokem območju pogojev zajema ter je tudi njegova časovna zahtevnost primerna.

Pod opisanimi pogoji lahko pričakujemo točnostni razred oz. največjo absolutno napako sistema manjšo od 0,01 kWh. Trajanje posameznega odčitka pa ne presega ene sekunde.

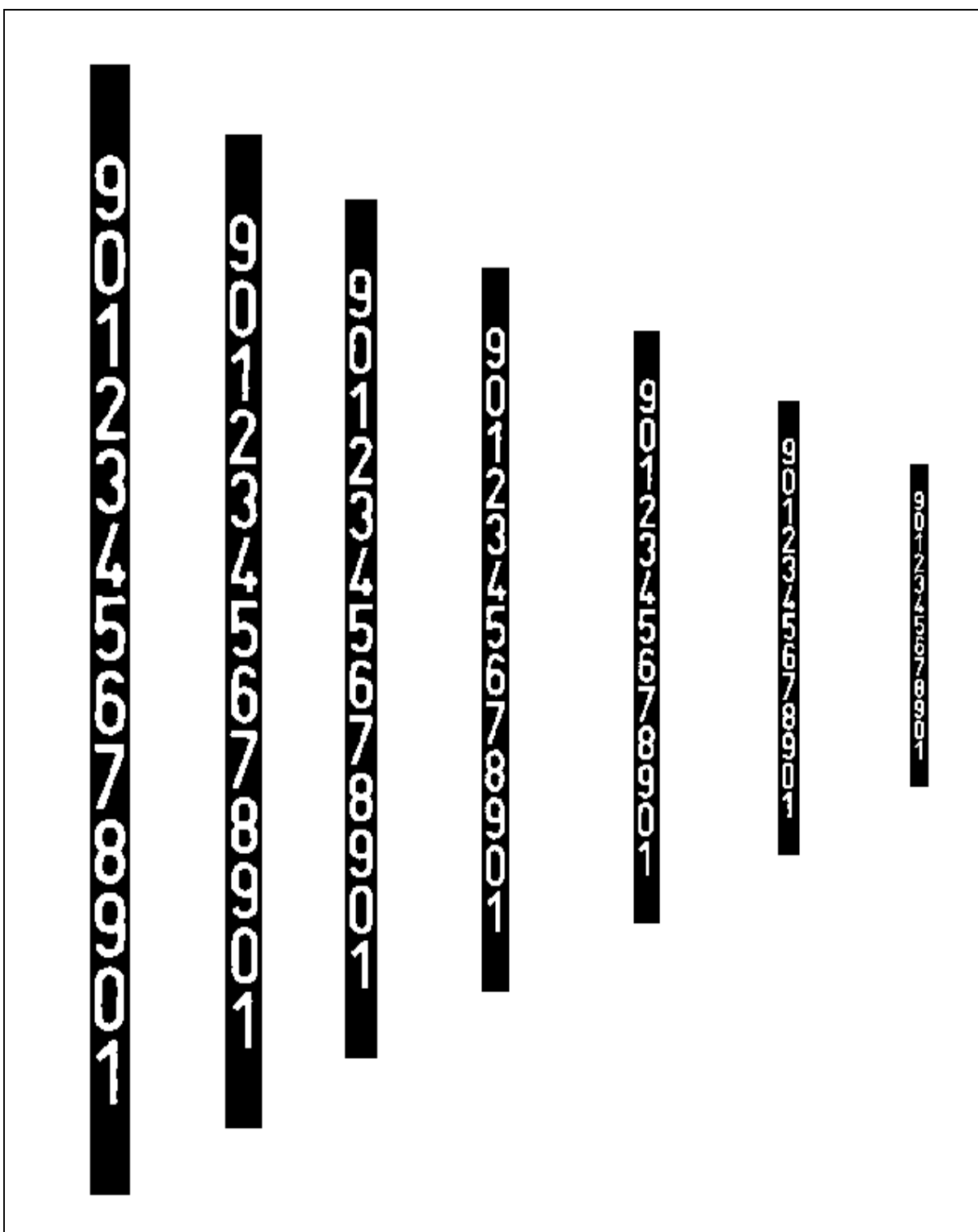


## **Dodatek A: Slike trakov**

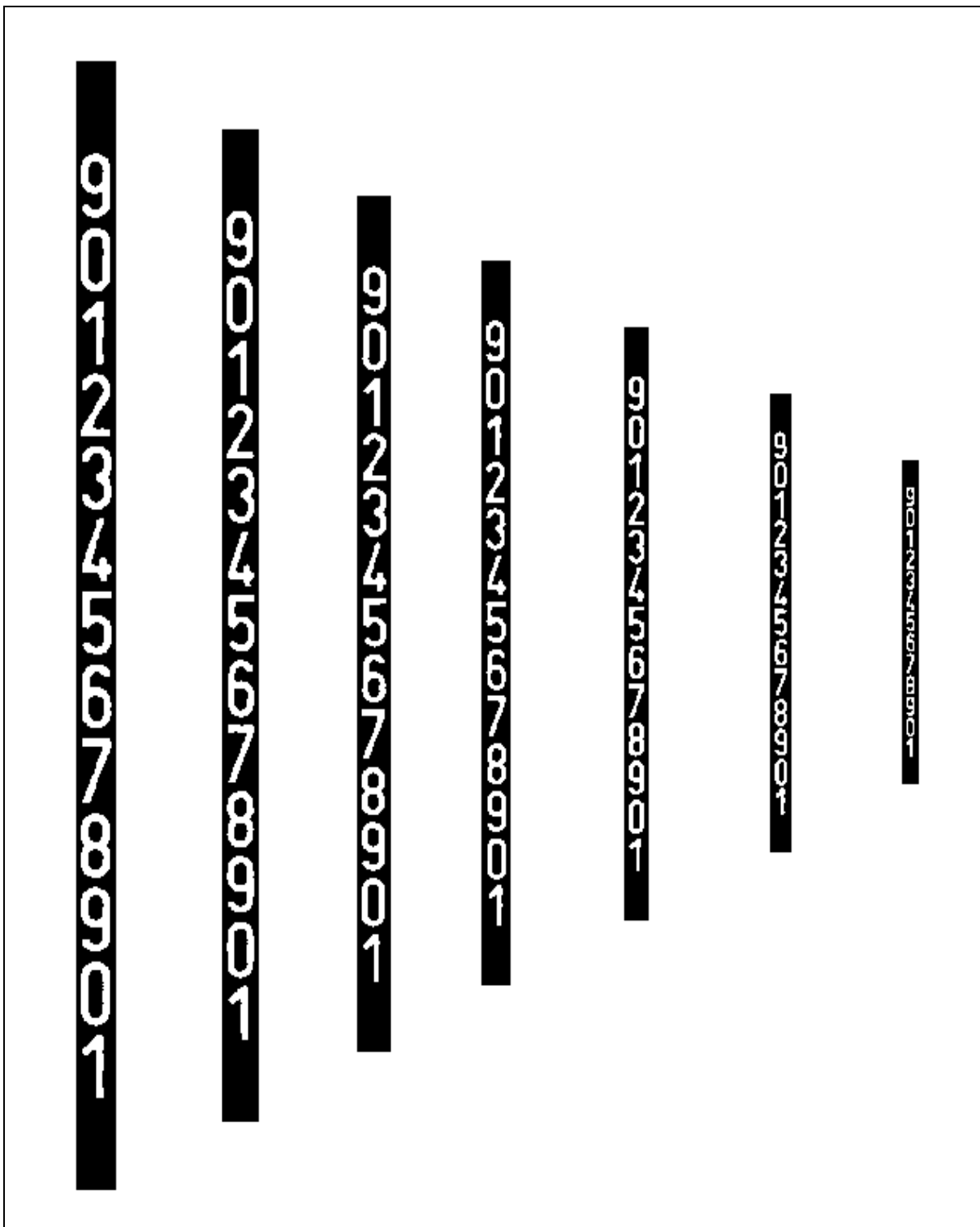
Na tem mestu prikazujemo slike trakov, ki smo jih uporabili pri izvajanju meritev za določanje merilnih karakteristik sistema (poglavje 1). Trakovi so prikazani glede na velikost števk, ki smo jih zajeli za njihovo izgradnjo (slike od 108 do 111) ter glede na velikost števk, s katerimi smo jih zgradili (slike od 112 do 115). Razlaga je opisana v poglavju 1.



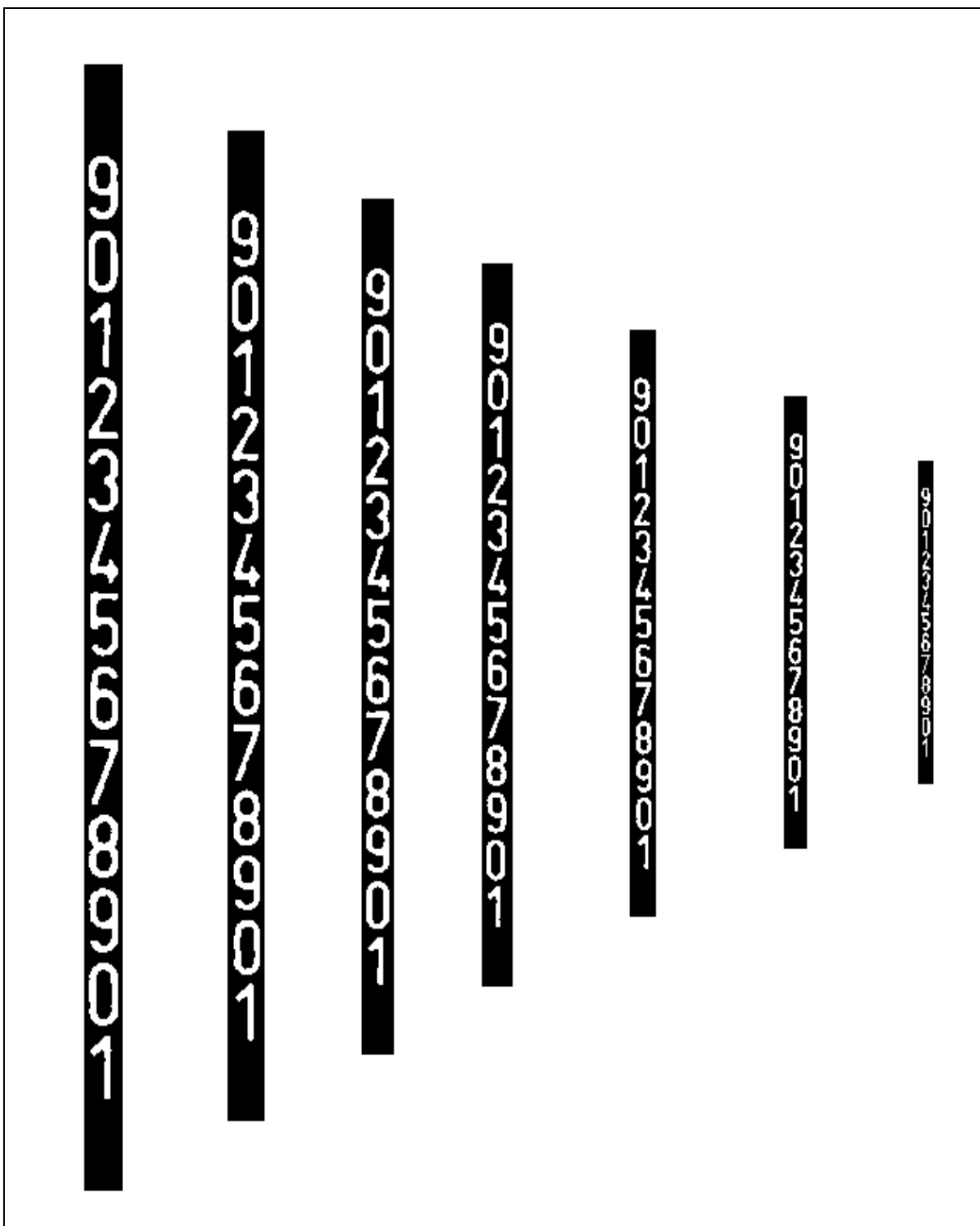
Slika 108: Trakovi s števki velikosti 170, 150, 130, 110, 90, 70 in 50 slikovnih elementov, zgrajeni iz števika, zajetega pri višini 170 slikovnih elementov



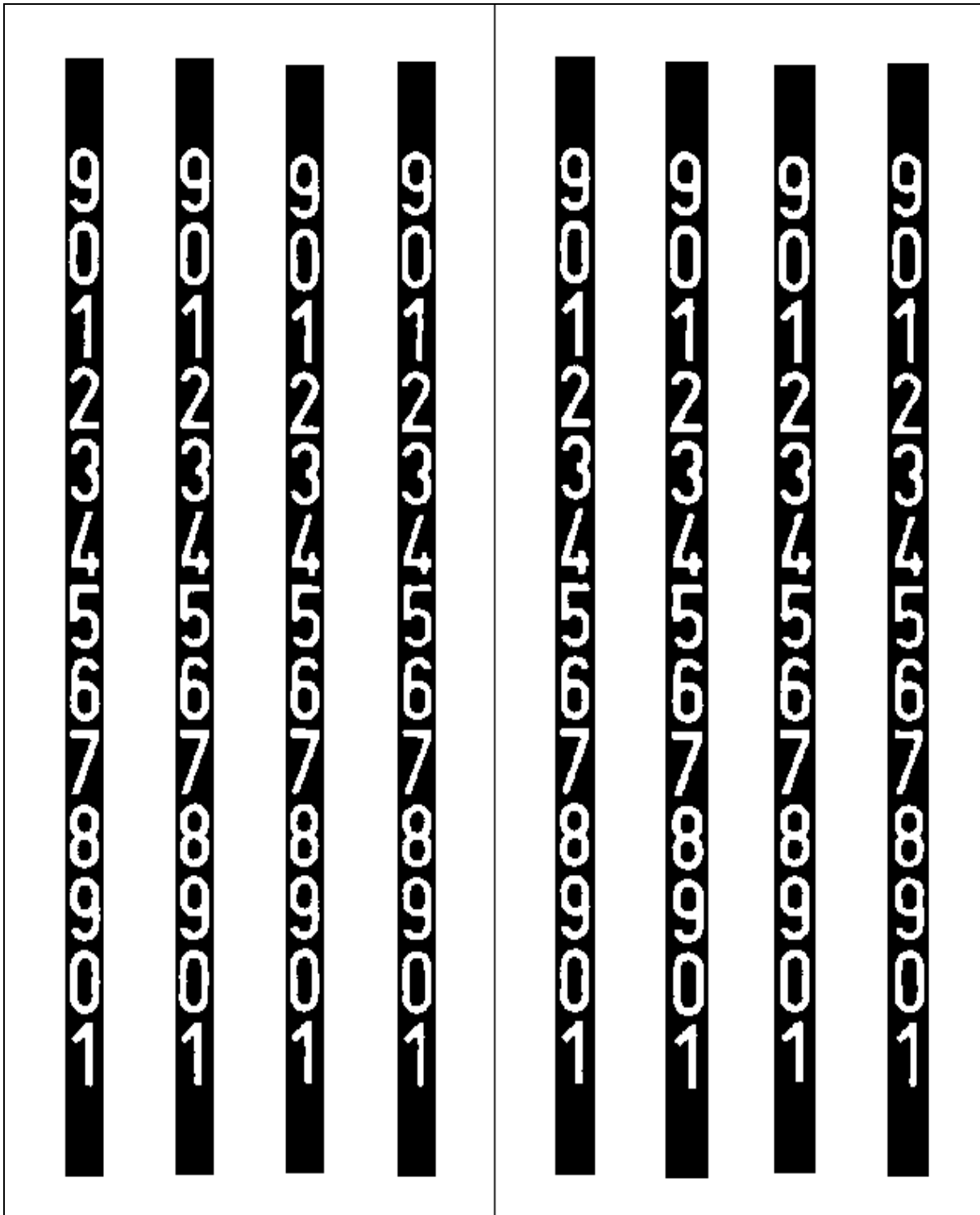
Slika 109: Trakovi s števkami velikosti 170, 150, 130, 110, 90, 70 in 50 slikovnih elementov, zgrajeni iz števnik, zajetega pri višini 150 slikovnih elementov



Slika 110: Trakovi s števki velikosti 170, 150, 130, 110, 90, 70 in 50 slikovnih elementov, zgrajeni iz števnik, zajetega pri višini 130 slikovnih elementov

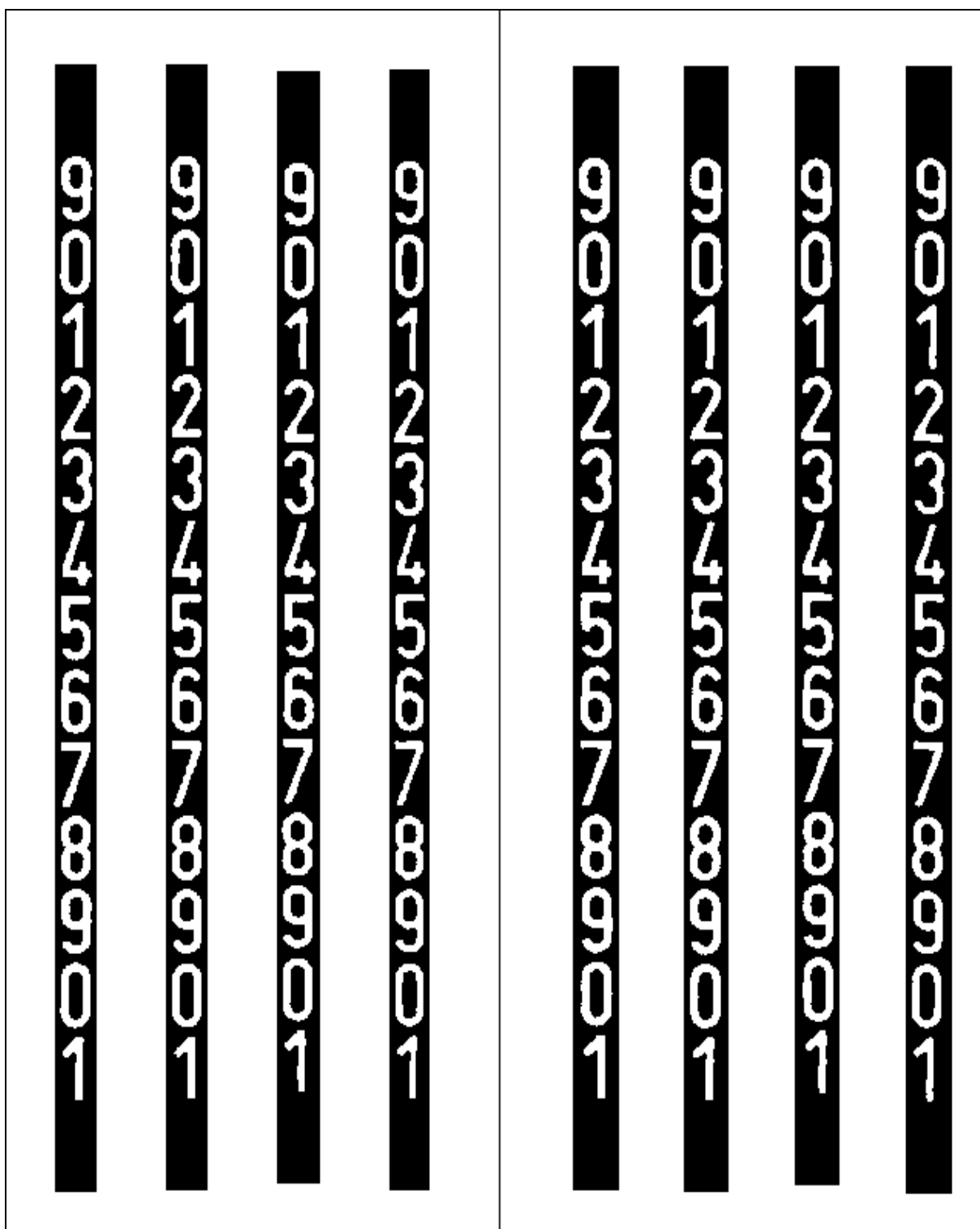


Slika 111: Trakovi s števki velikosti 170, 150, 130, 110, 90, 70 in 50 slikovnih elementov, zgrajeni iz števnik, zajetega pri višini 110 slikovnih elementov

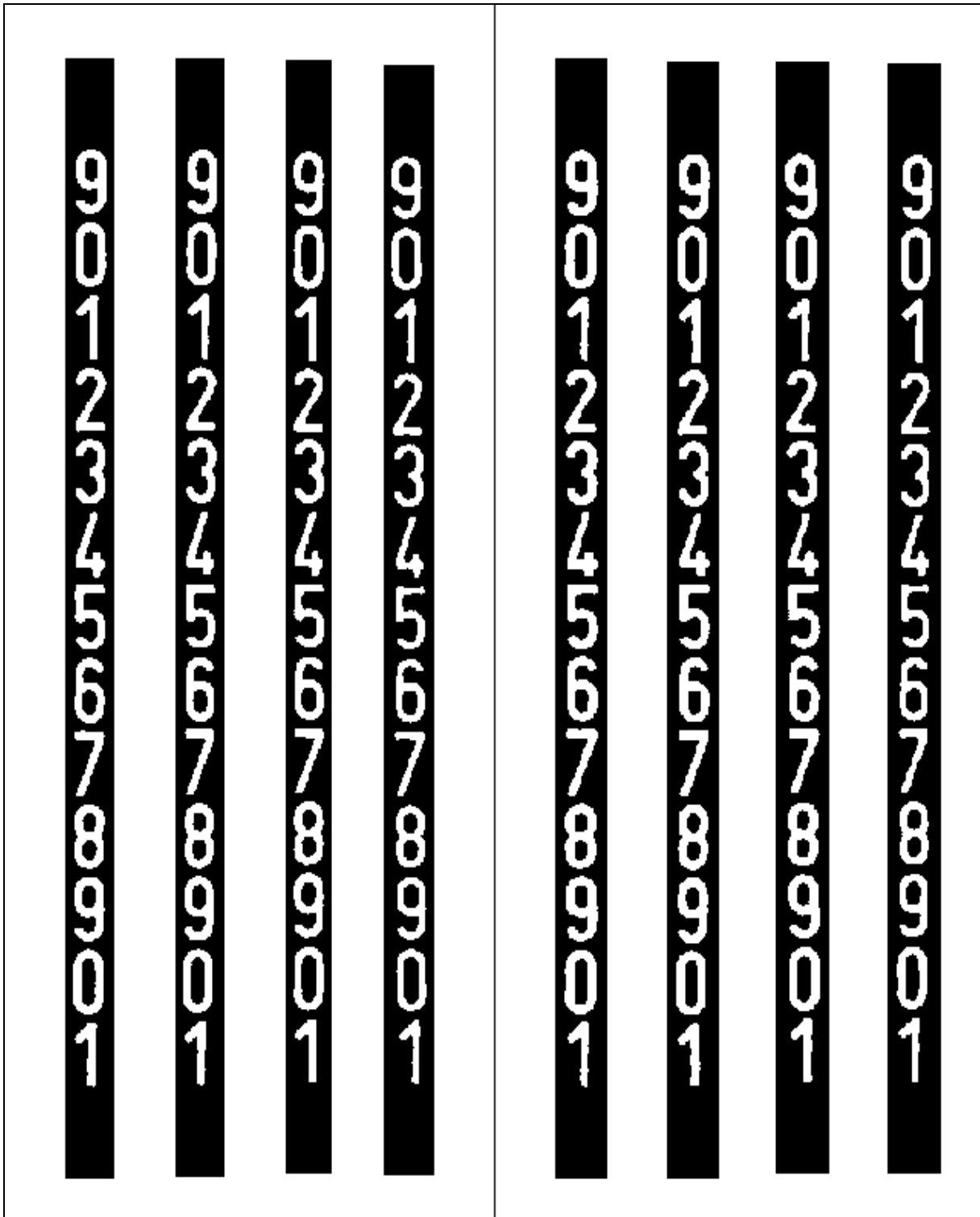


Slika 112: Trakovi s števki velikosti 170 (levo) in 150 (desno) slikovnih elementov, zgrajeni z zajemom števnik pri štirih testnih povečavah

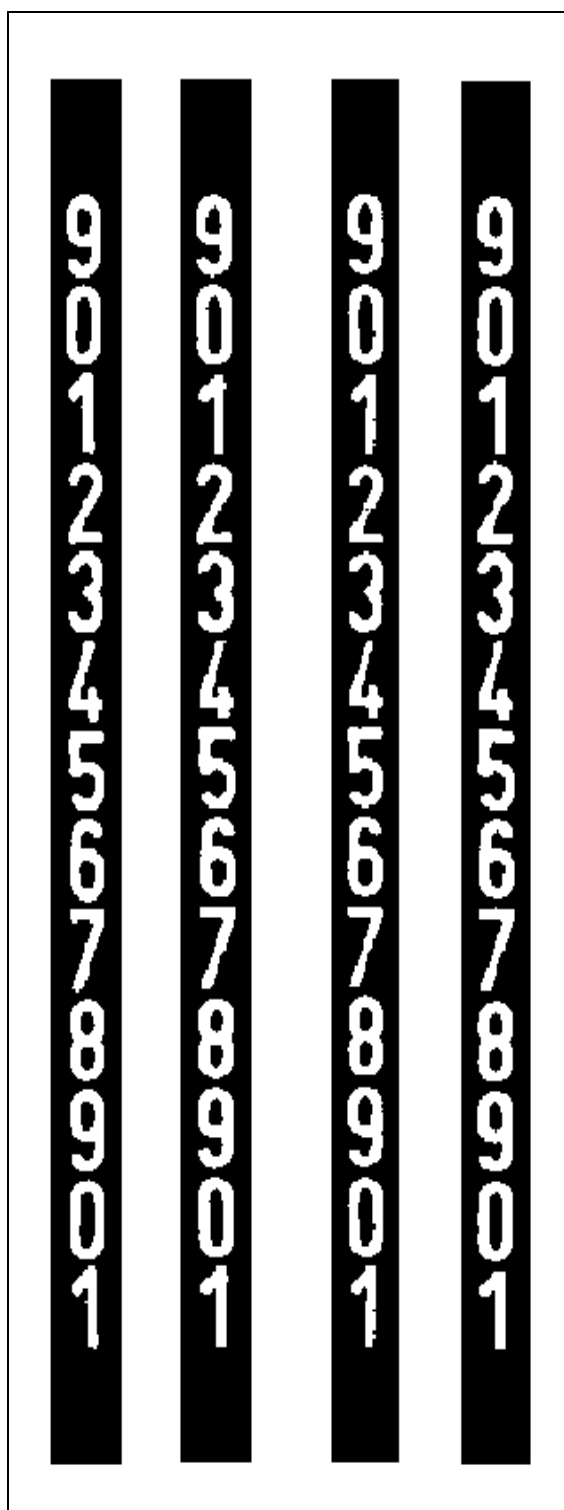




Slika 113: Trakovi s števki velikosti 130 (levo) in 110 (desno) slikovnih elementov, zgrajeni z zajemom števnik pri štirih testnih povečavah



Slika 114: Trakovi s števki velikosti 90 (levo) in 70 (desno) slikovnih elementov, zgrajeni z zajemom števnik pri štirih testnih povečavah



Slika 115: Trakovi s števkami velikosti 50 slikovnih elementov, zgrajeni z zajemom števniaka pri štirih testnih povečavah



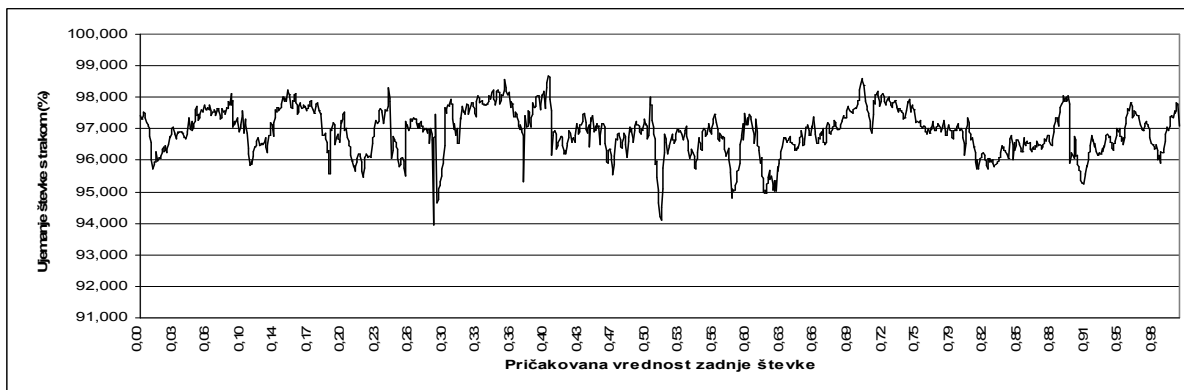
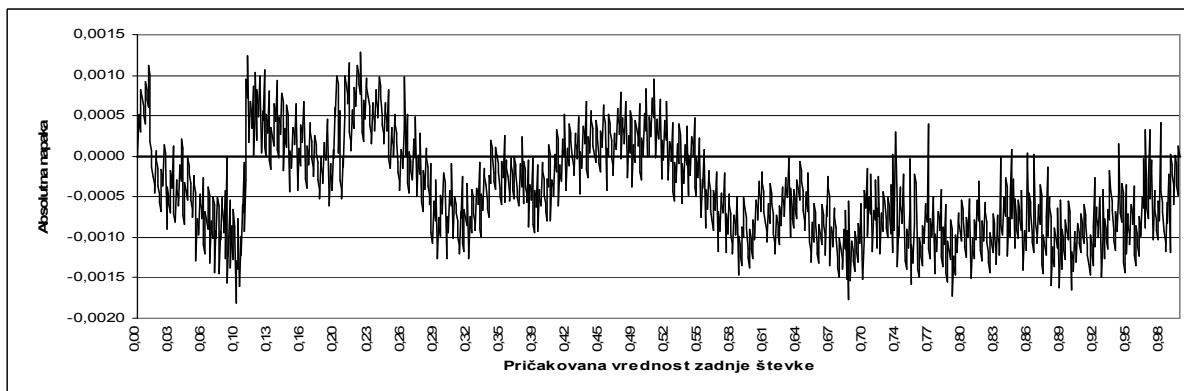
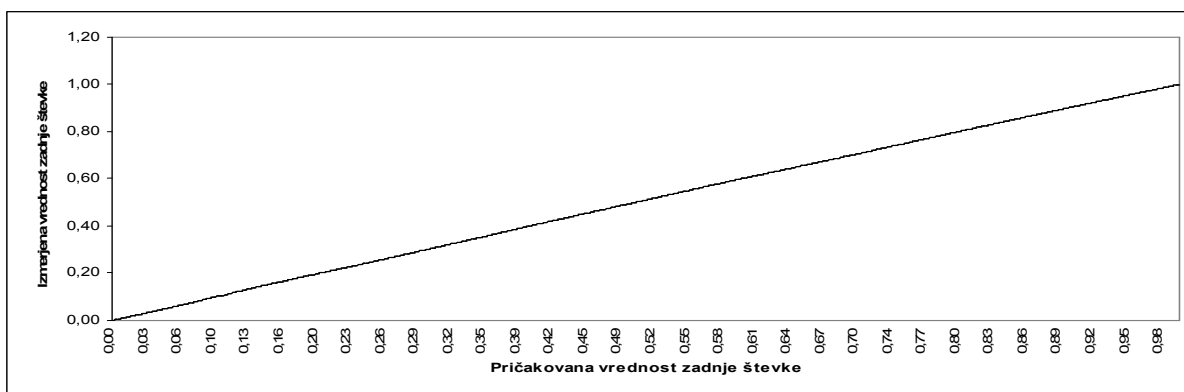
## **Dodatek B: Rezultati meritev**

V tem razdelku podajamo rezultate meritev, s pomočjo katerih smo določili merilne karakteristike sistema. Razvrščeni so glede na povečavo števnik ob izvajanju meritev, kot je opisano v poglavju 1 skupaj z analizo rezultatov.

## B.1 Meritve pri velikosti števnik 170 slikovnih elementov

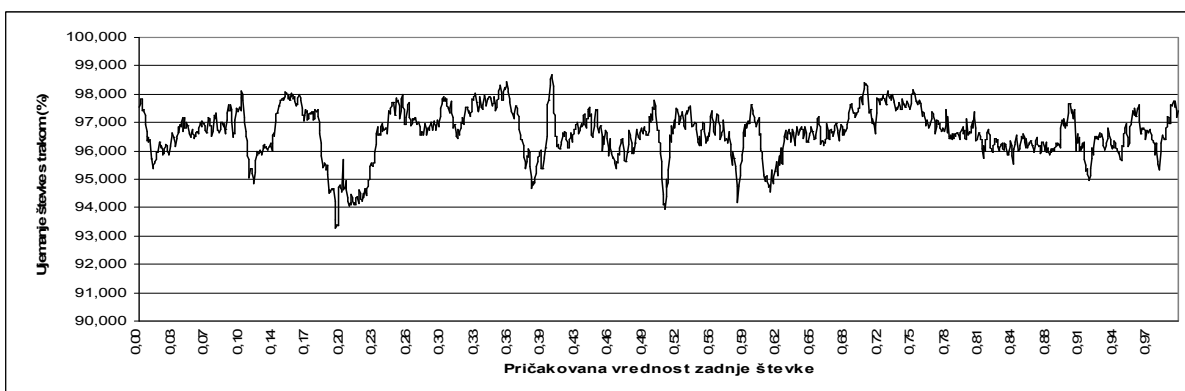
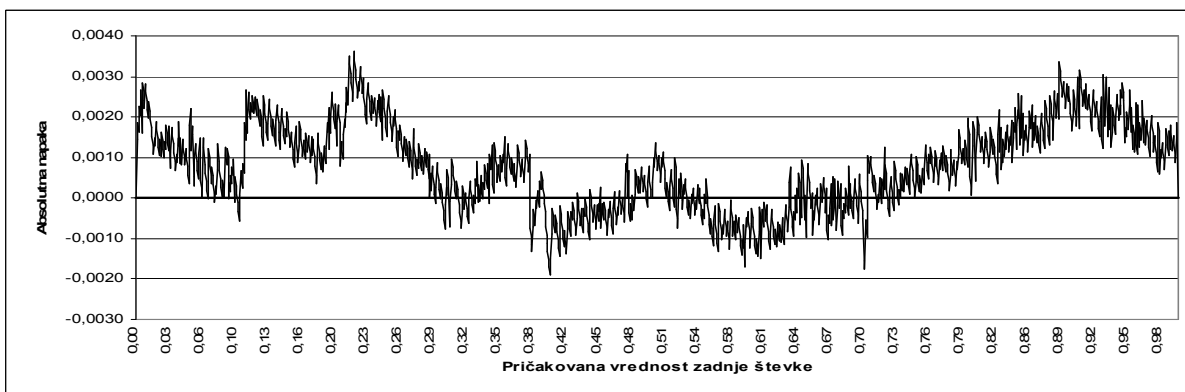
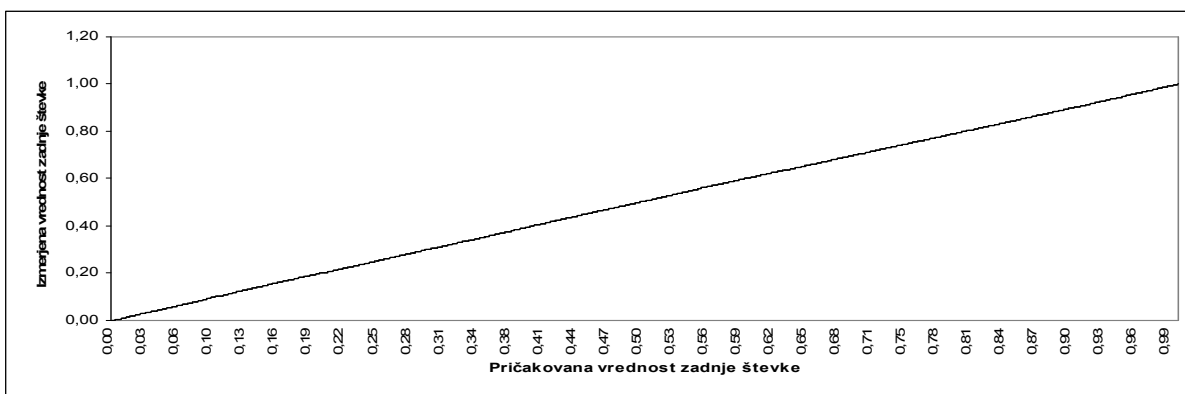
### B.1.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: 0°

Št. meritev	1231
Celotni čas (s)	697,89
Povpr. čas (s)	0,567
Maks. neg. odstop.	0,0018
Maks. poz. odstop.	0,0013
Maks. napaka	0,0018
Min. % ujemanja	93,966



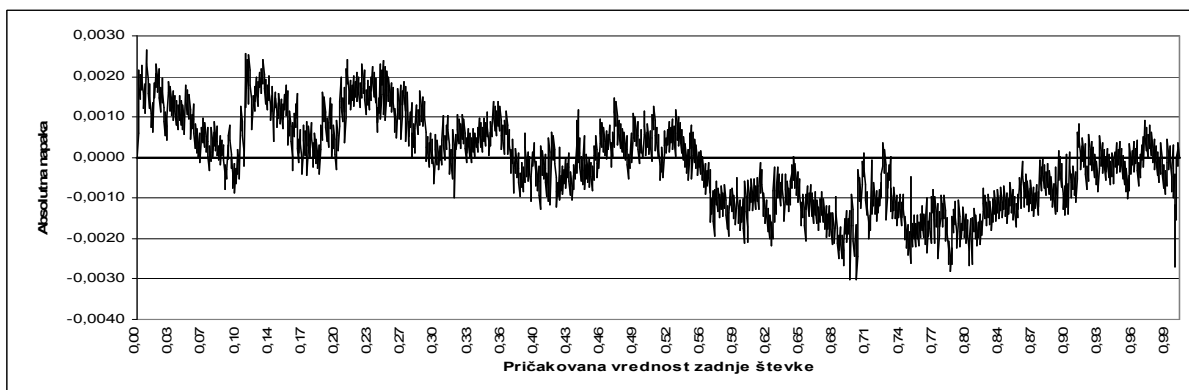
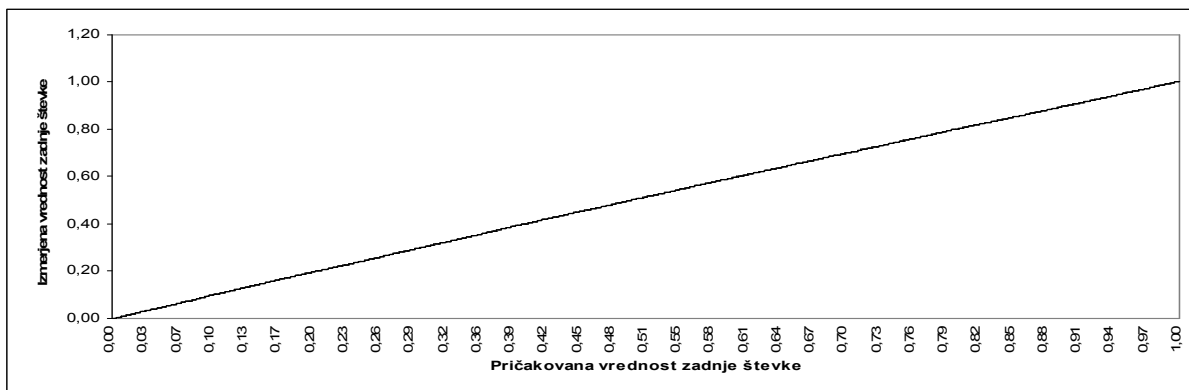
**B.1.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	1392
<b>Celotni čas (s)</b>	701,76
<b>Povpr. čas (s)</b>	0,504
<b>Maks. neg. odstop.</b>	0,0019
<b>Maks. poz. odstop.</b>	0,0036
<b>Maks. napaka</b>	0,0036
<b>Min. % ujemanja</b>	93,284



### B.1.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: 0°

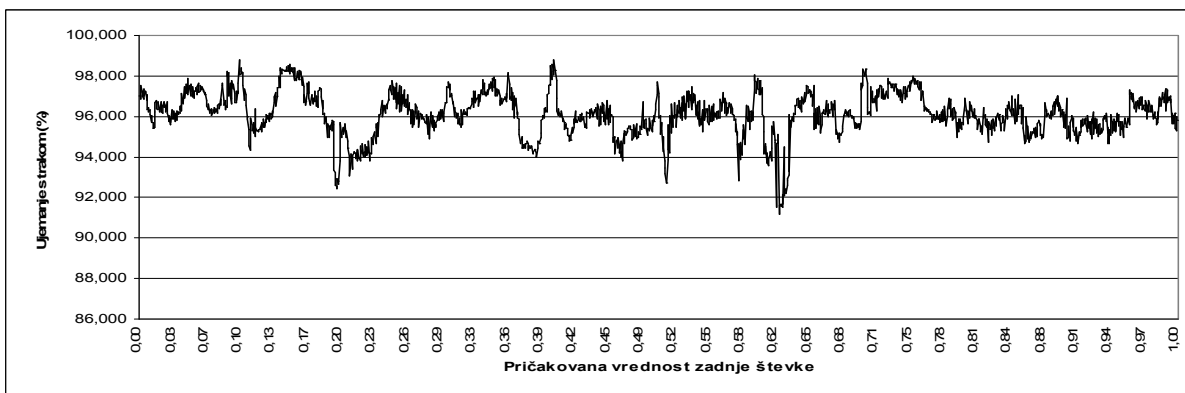
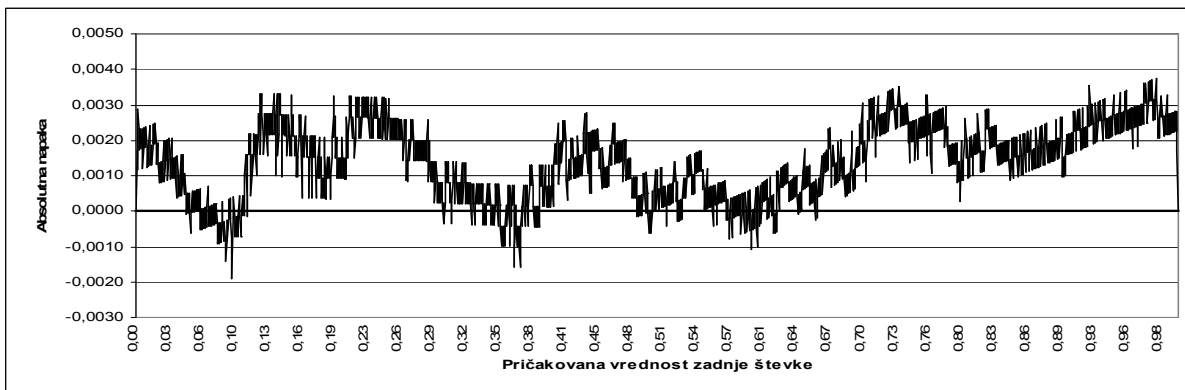
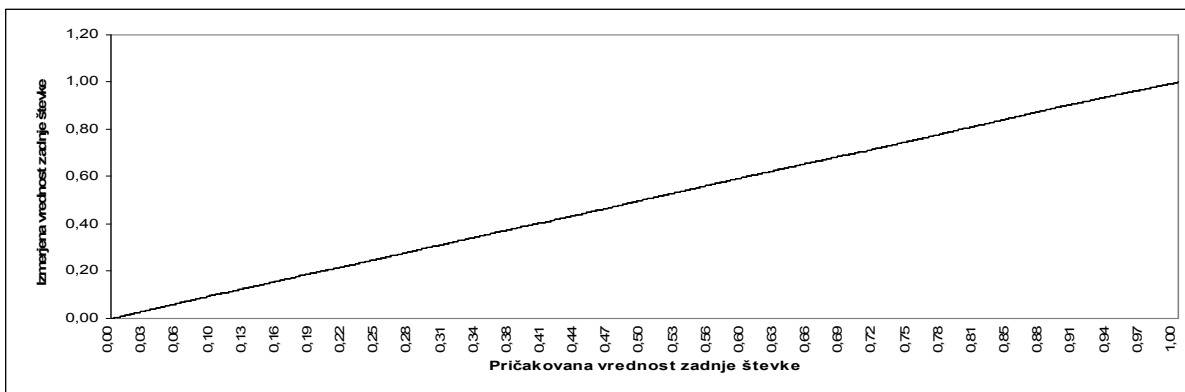
<b>Št. meritev</b>	1633
<b>Celotni čas (s)</b>	729,28
<b>Povpr. čas (s)</b>	0,447
<b>Maks. neg. odstop.</b>	0,0030
<b>Maks. poz. odstop.</b>	0,0027
<b>Maks. napaka</b>	0,0030
<b>Min. % ujemanja</b>	93,136





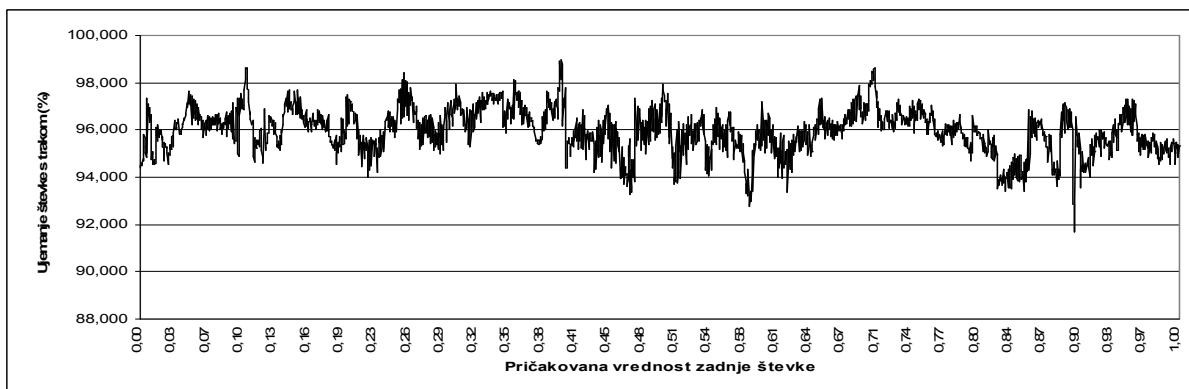
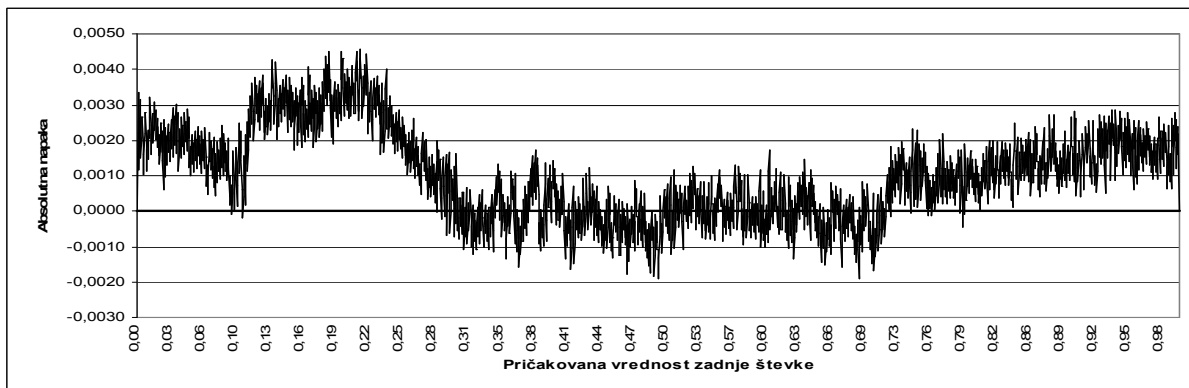
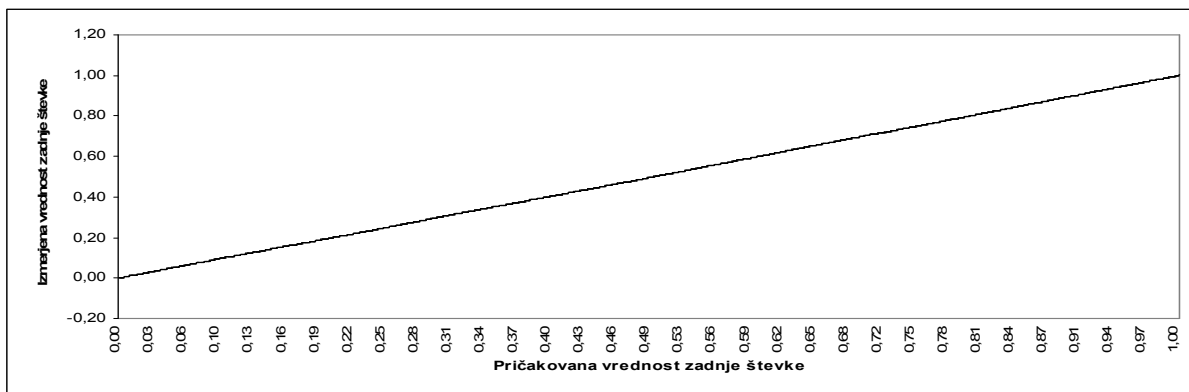
**B.1.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: 0°**

<b>Št.meritev</b>	1801
<b>Celotni čas (s)</b>	743,47
<b>Povpr. čas (s)</b>	0,413
<b>Maks. neg. odstop.</b>	0,0019
<b>Maks. poz. odstop.</b>	0,0037
<b>Maks. napaka</b>	0,0037
<b>Min. % ujemanja</b>	91,158



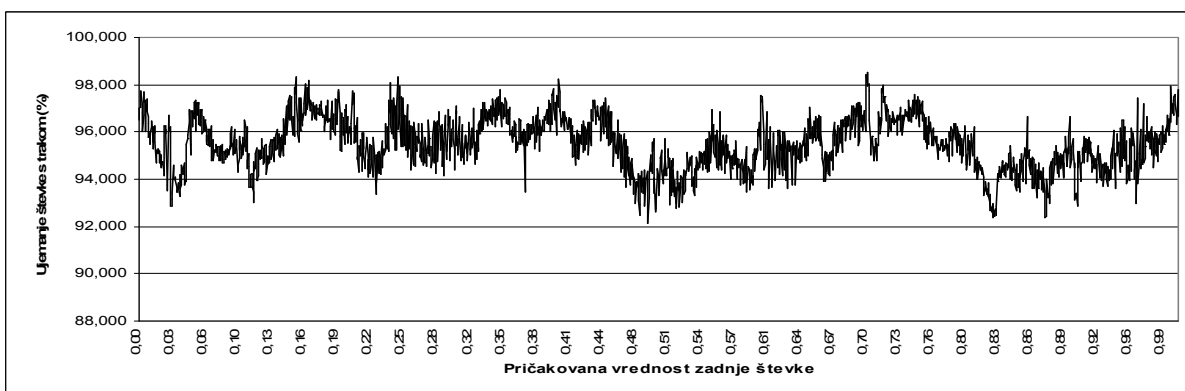
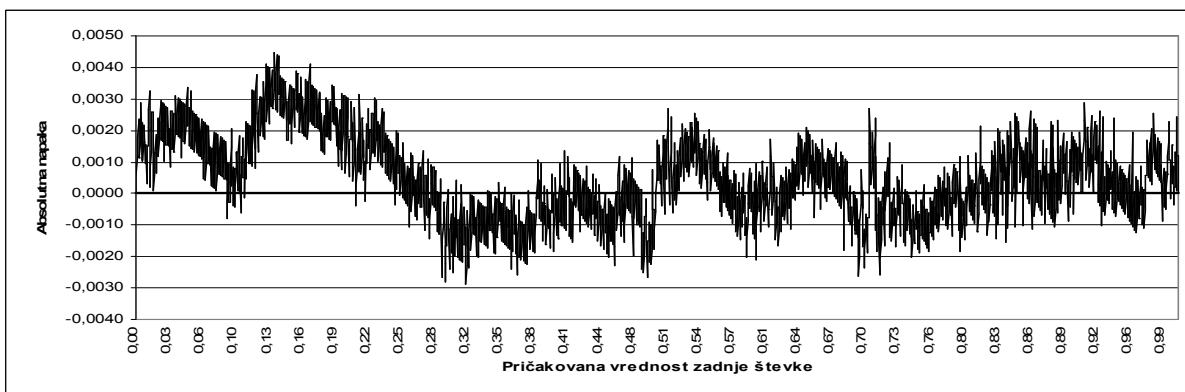
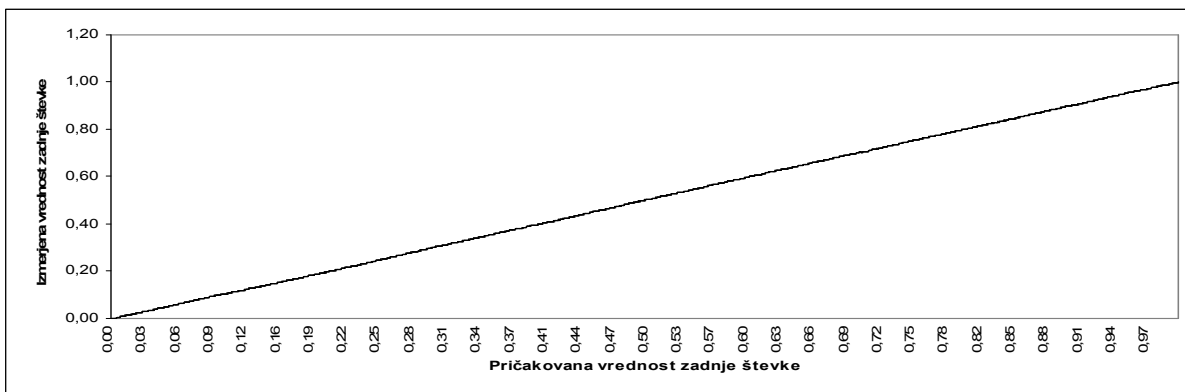
### B.1.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števniaka: 0°

Št. meritev	1989
Celotni čas (s)	729,71
Povpr. čas (s)	0,367
Maks. neg. odstop.	0,0019
Maks. poz. odstop.	0,0046
Maks. napaka	0,0046
Min. % ujemanja	91,686



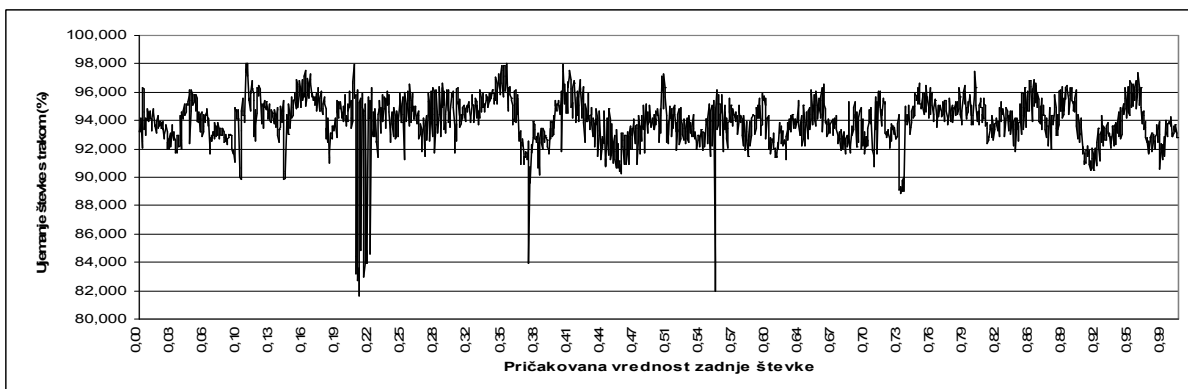
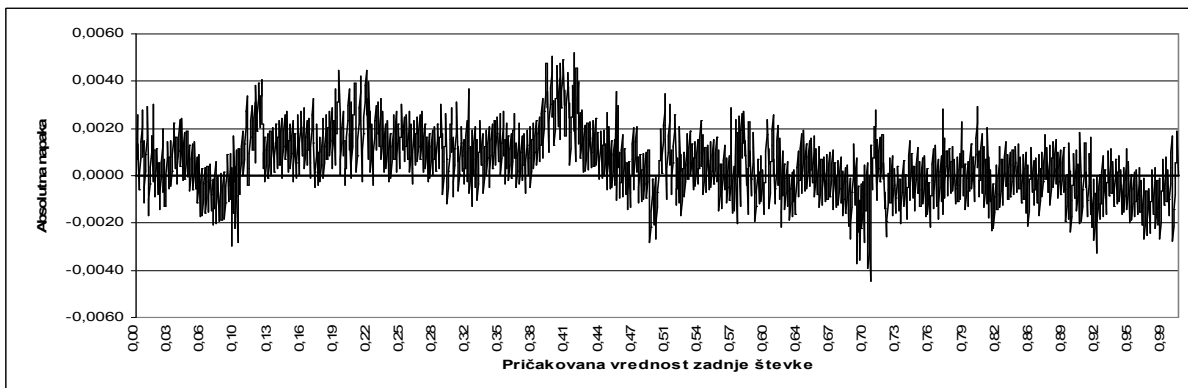
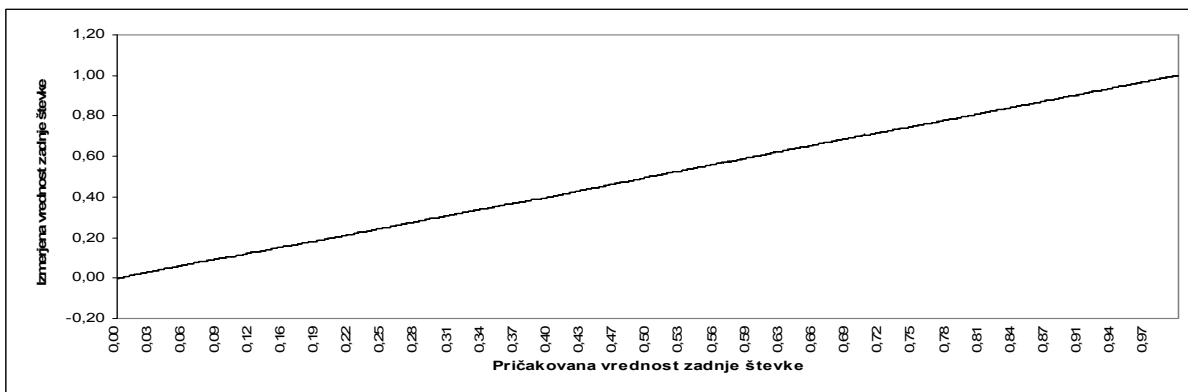
**B.1.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	2140
<b>Celotni čas (s)</b>	746,91
<b>Povpr. čas (s)</b>	0,349
<b>Maks. neg. odstop.</b>	0,0029
<b>Maks. poz. odstop.</b>	0,0045
<b>Maks. napaka</b>	0,0045
<b>Min. % ujemanja</b>	92,125



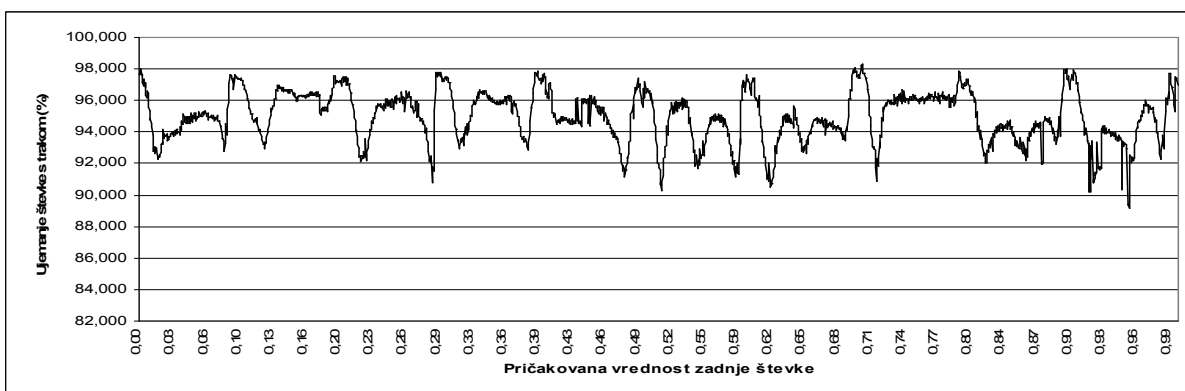
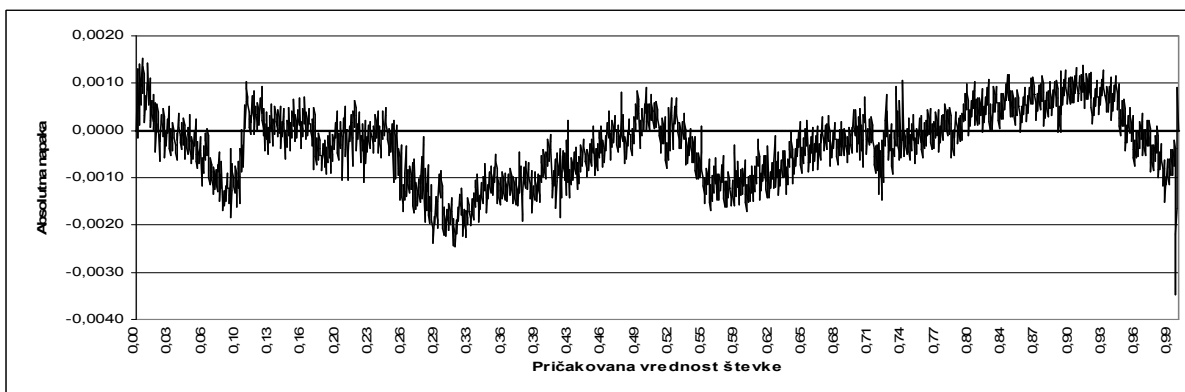
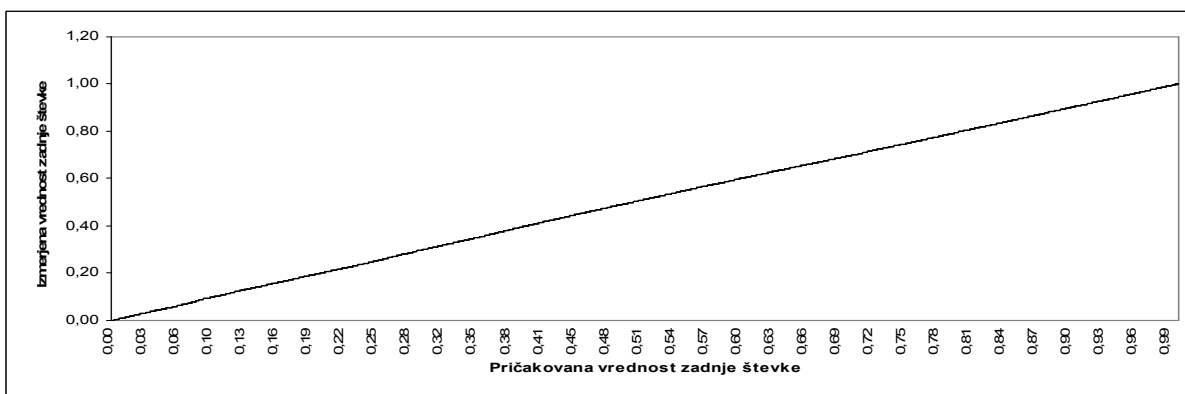
### B.1.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števniaka: 0°

Št. meritev	1983
Celotni čas (s)	638,98
Povpr. čas (s)	0,322
Maks. neg. odstop.	0,0045
Maks. poz. odstop.	0,0052
Maks. napaka	0,0052
Min. % ujemanja	81,654



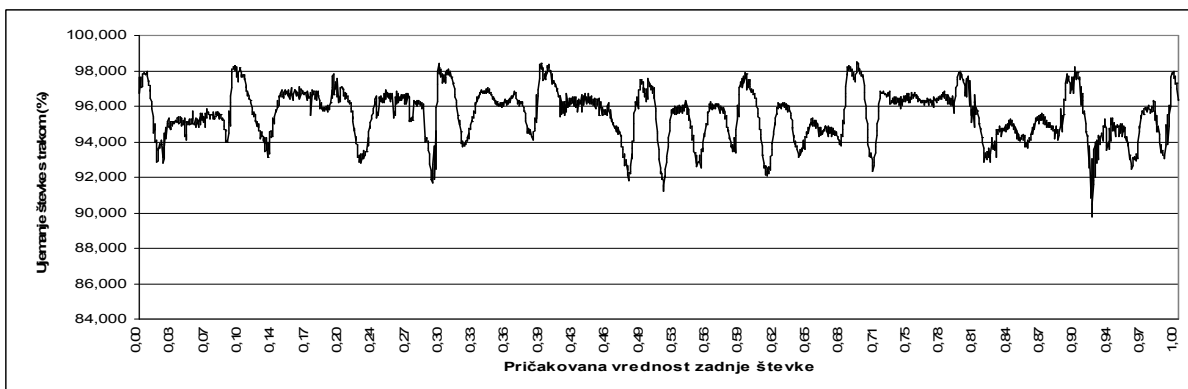
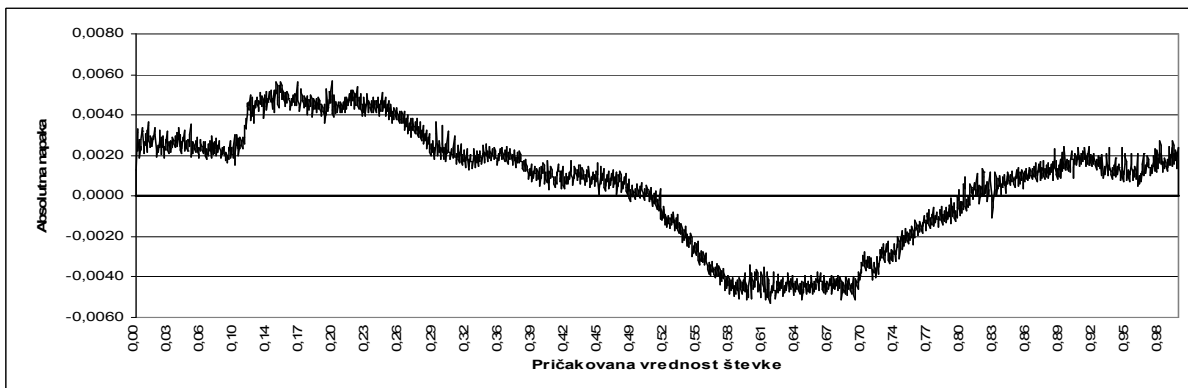
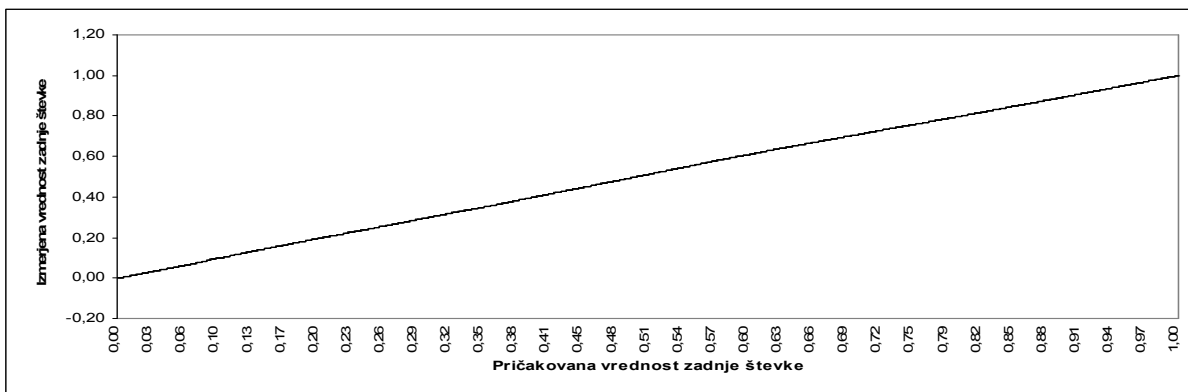
**B.1.8 Velikost števk na traku: 170 slikovnih elementov, rotacija števniaka: +5°**

<b>Št.meritev</b>	1908
<b>Celotni čas (s)</b>	764,97
<b>Povpr. čas (s)</b>	0,401
<b>Maks. neg. odstop.</b>	0,0035
<b>Maks. poz. odstop.</b>	0,0015
<b>Maks. napaka</b>	0,0035
<b>Min. % ujemanja</b>	89,134



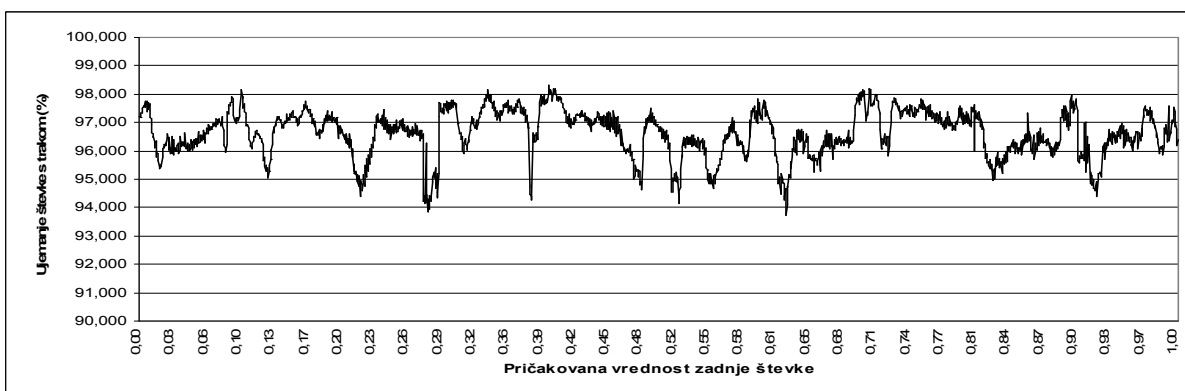
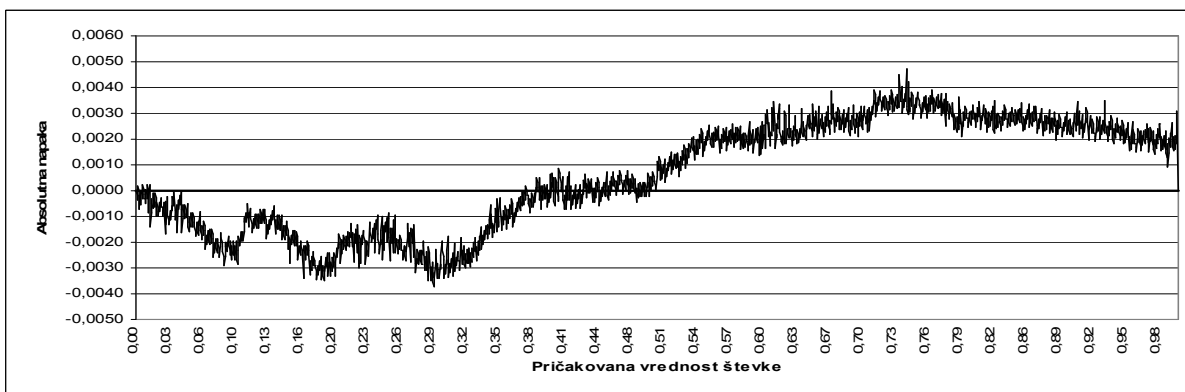
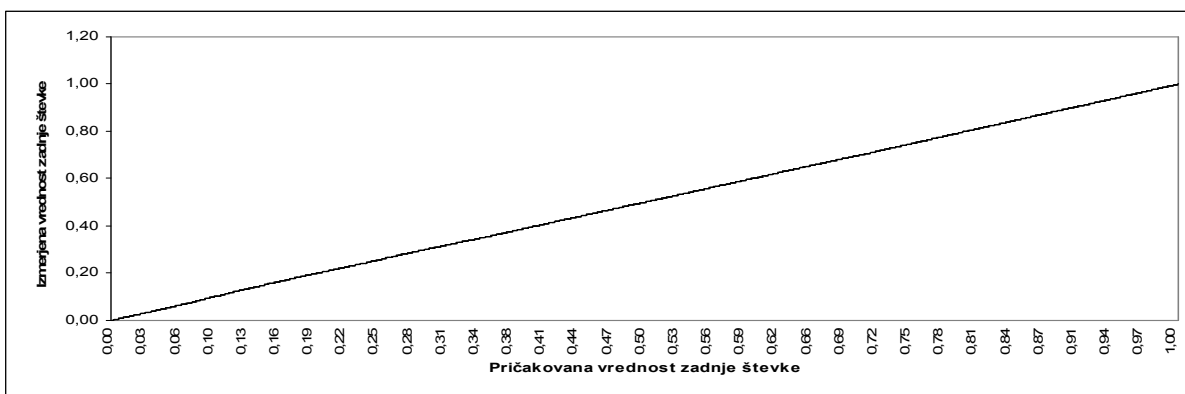
**B.1.9 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: +2°5'**

<b>Št. meritev</b>	2085
<b>Celotni čas (s)</b>	790,77
<b>Povpr. čas (s)</b>	0,379
<b>Maks. neg. odstop.</b>	0,0053
<b>Maks. poz. odstop.</b>	0,0057
<b>Maks. napaka</b>	0,0057
<b>Min. % ujemanja</b>	89,776



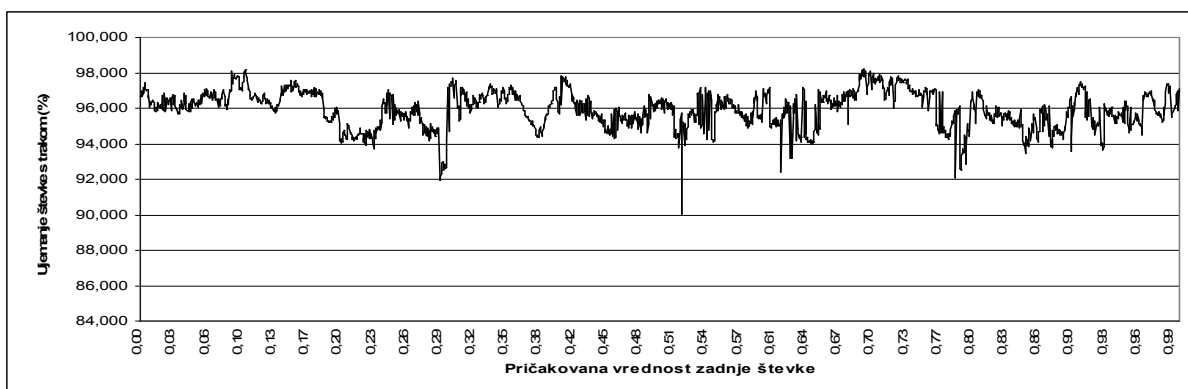
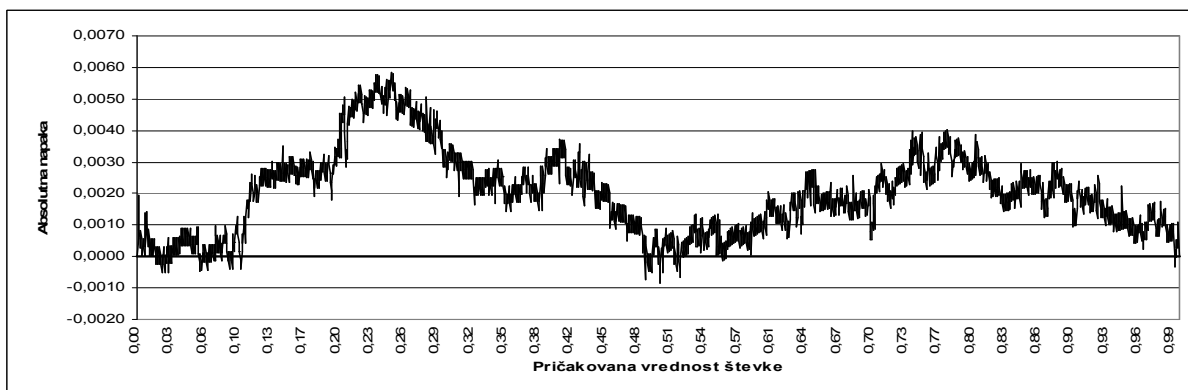
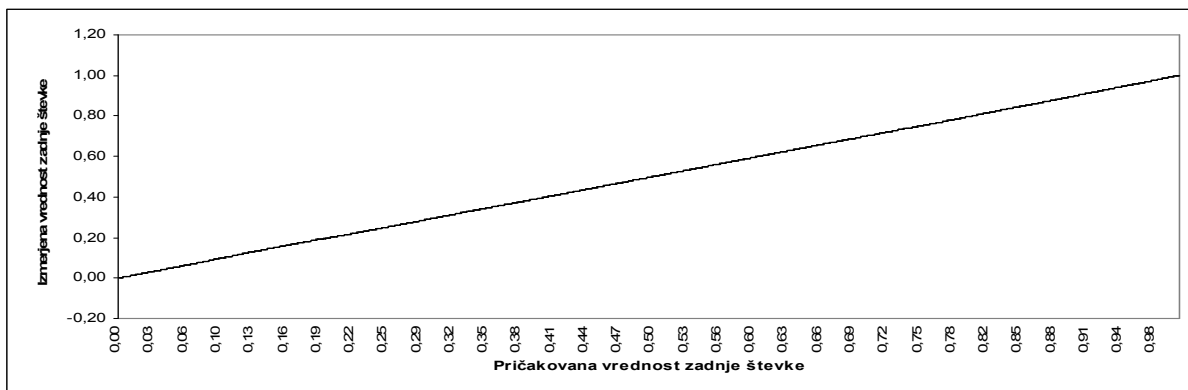
**B.1.10 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: -2'5°**

<b>Št.meritev</b>	1962
<b>Celotni čas (s)</b>	777,87
<b>Povpr. čas (s)</b>	0,396
<b>Maks. neg. odstop.</b>	0,0037
<b>Maks. poz. odstop.</b>	0,0047
<b>Maks. napaka</b>	0,0047
<b>Min. % ujemanja</b>	93,729



### B.1.11 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: -5°

Št. meritev	1812
Celotni čas (s)	772,71
Povpr. čas (s)	0,426
Maks. neg. odstop.	0,0008
Maks. poz. odstop.	0,0058
Maks. napaka	0,0058
Min. % ujemanja	90,051

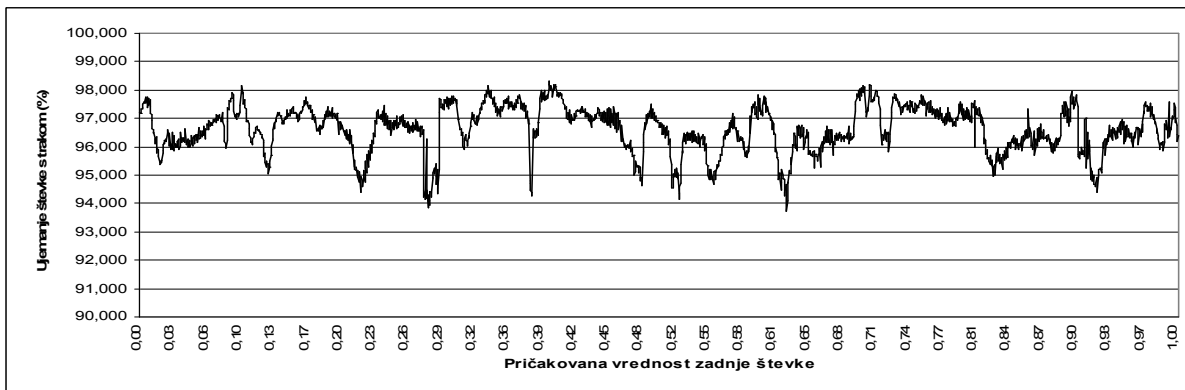
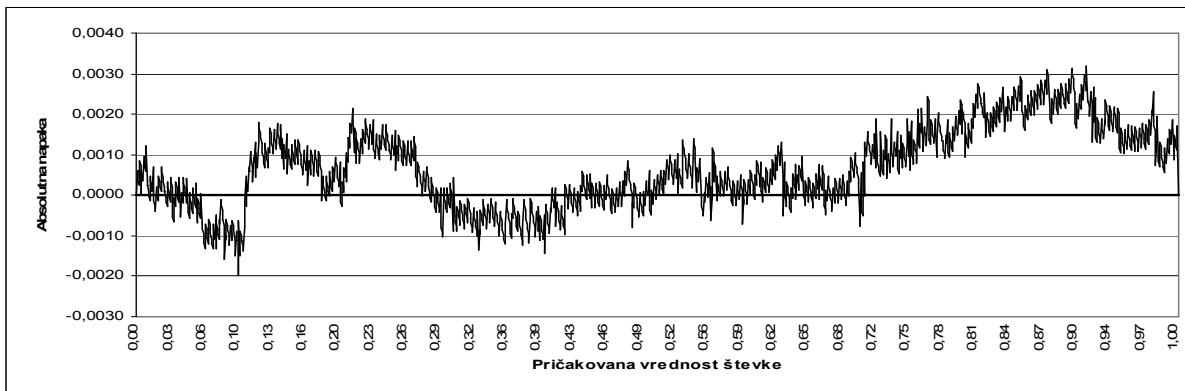
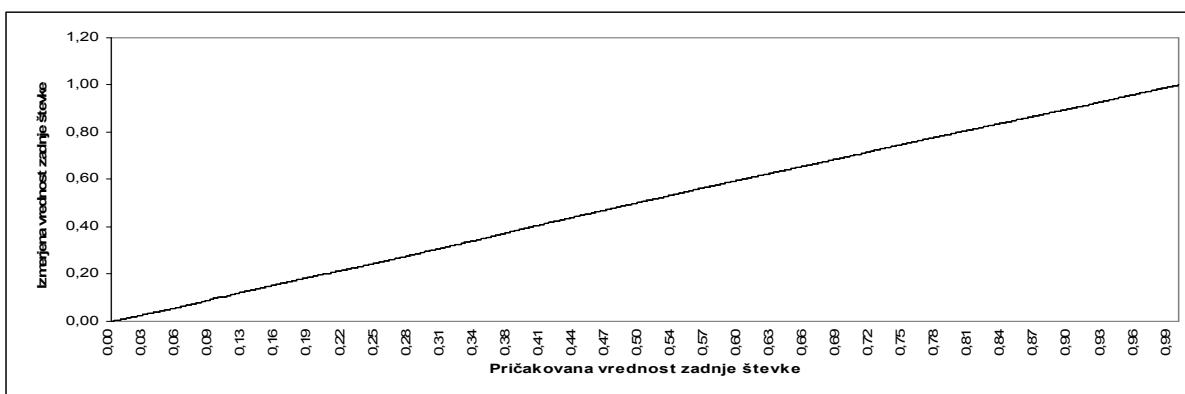




## B.2 Meritve pri velikosti števnika 150 slikovnih elementov

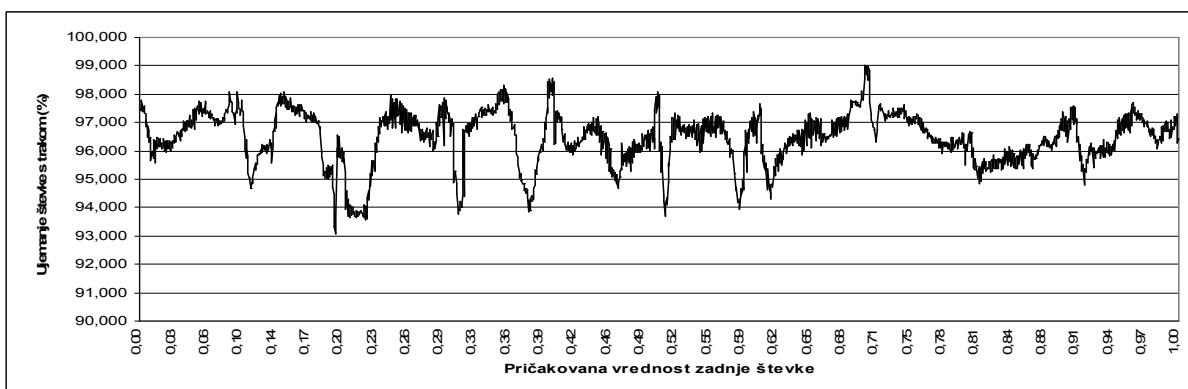
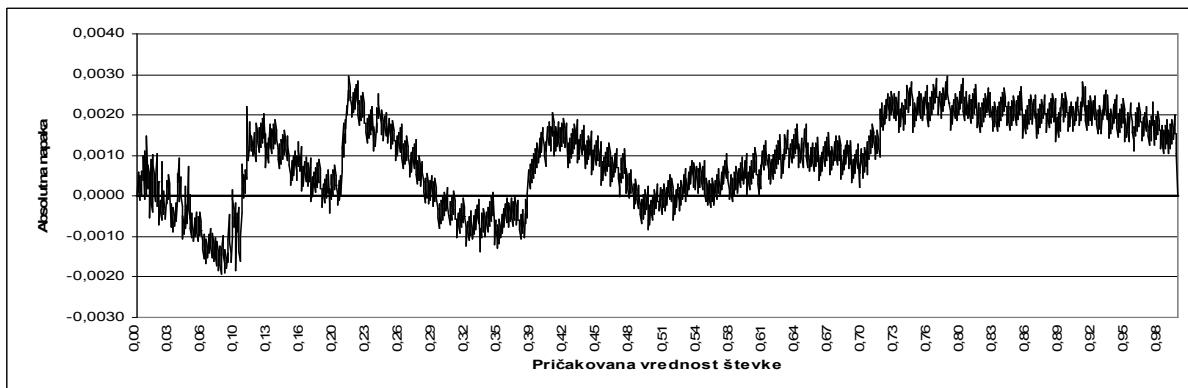
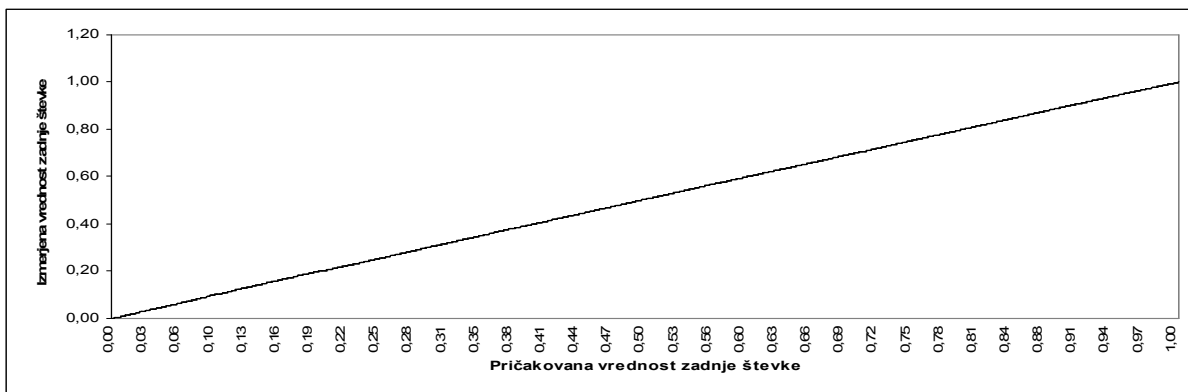
### B.2.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnika: 0°

Št.meritev	1648
Celotni čas (s)	707,35
Povpr. čas (s)	0,429
Maks. neg. odstop.	0,0020
Maks. poz. odstop.	0,0032
Maks. napaka	0,0032
Min. % ujemanja	93,287



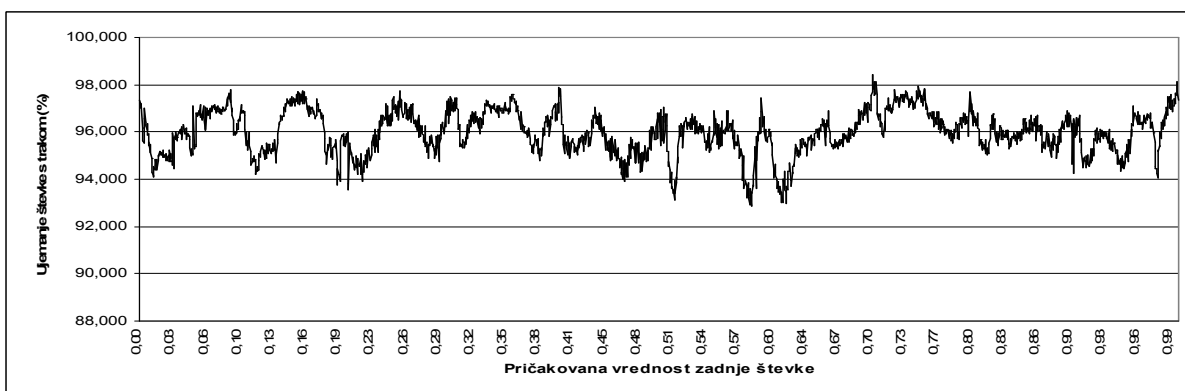
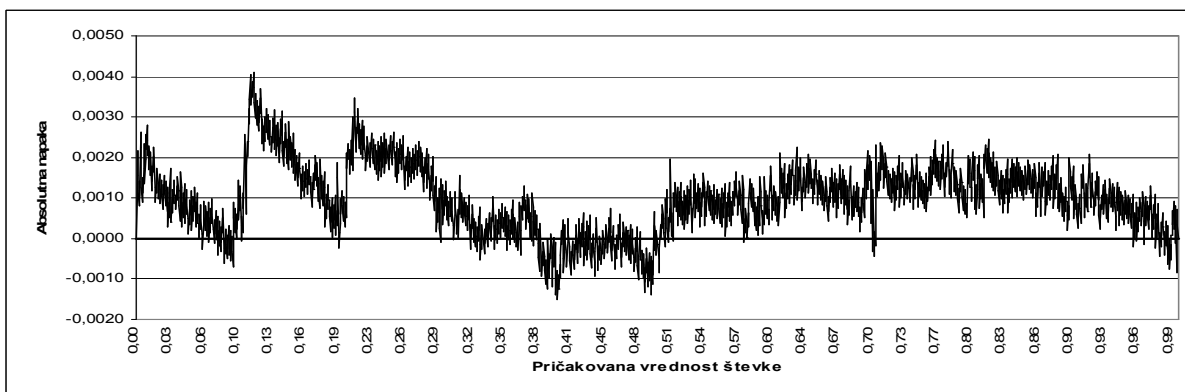
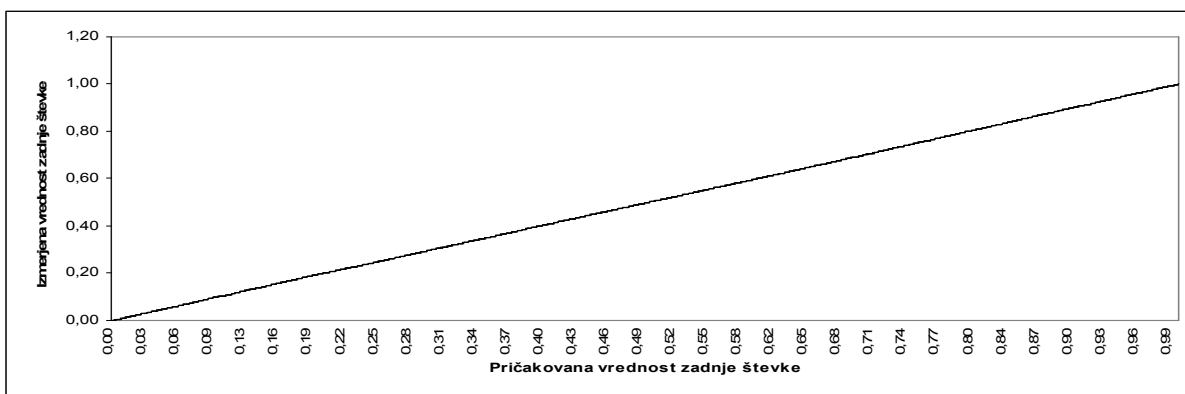
### B.2.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: 0°

<b>Št. meritev</b>	1769
<b>Celotni čas (s)</b>	723,26
<b>Povpr. čas (s)</b>	0,409
<b>Maks. neg. odstop.</b>	0,0019
<b>Maks. poz. odstop.</b>	0,0030
<b>Maks. napaka</b>	0,0030
<b>Min. % ujemanja</b>	93,056



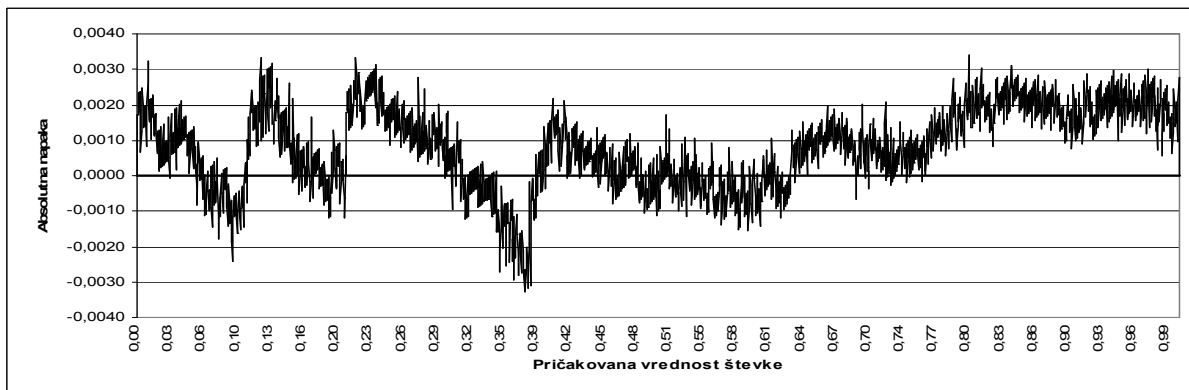
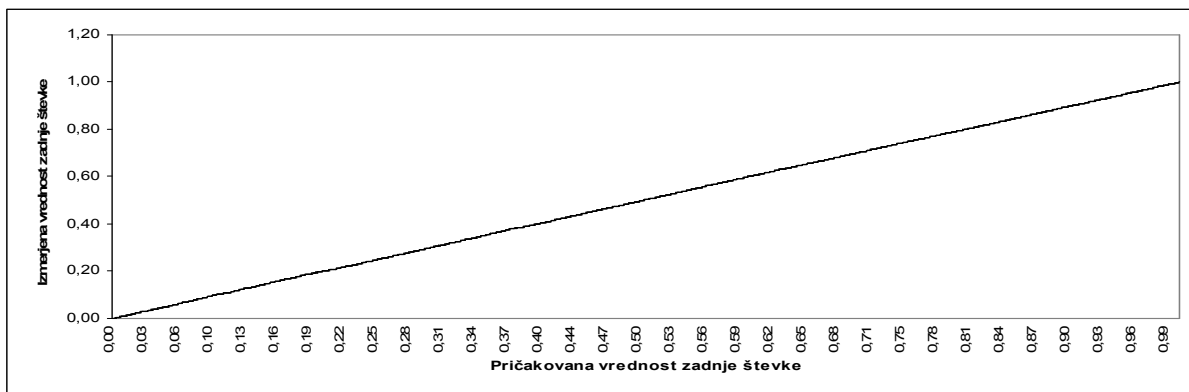
**B.2.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnika: 0°**

<b>Št.meritev</b>	1938
<b>Celotni čas (s)</b>	715,95
<b>Povpr. čas (s)</b>	0,369
<b>Maks. neg. odstop.</b>	0,0015
<b>Maks. poz. odstop.</b>	0,0041
<b>Maks. napaka</b>	0,0041
<b>Min. % ujemanja</b>	92,869



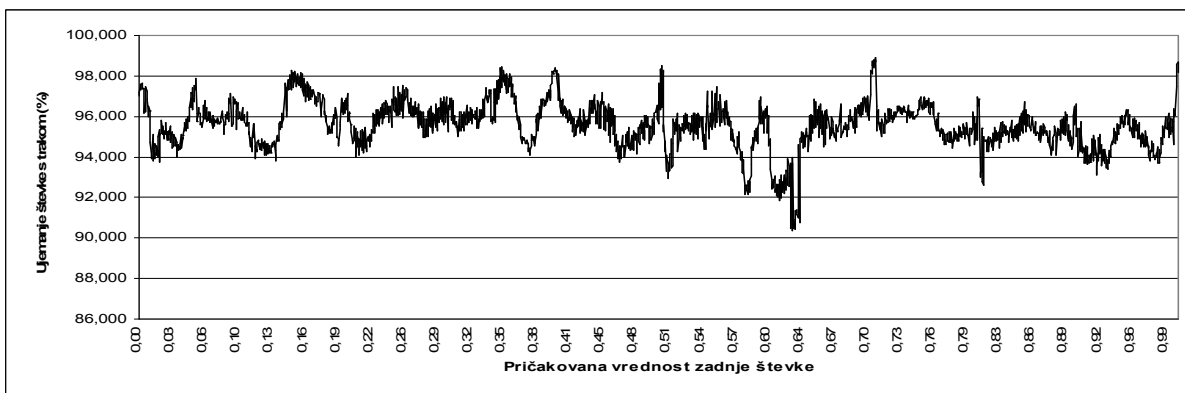
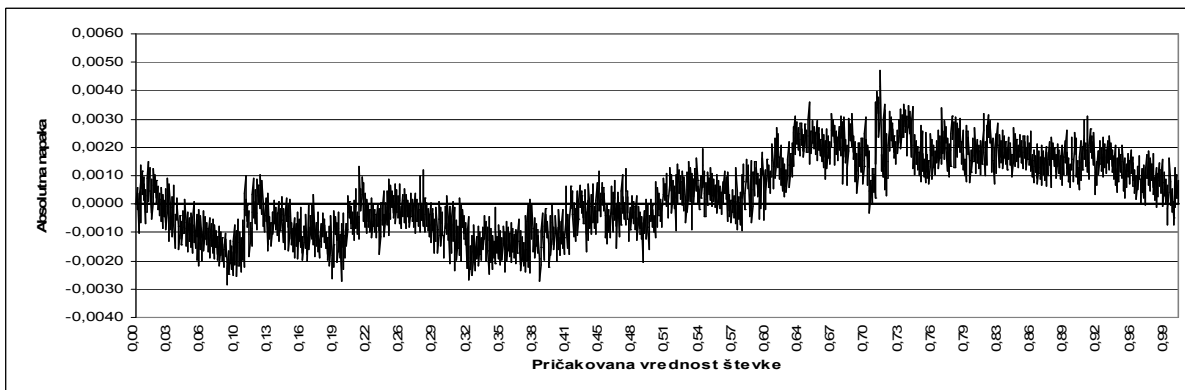
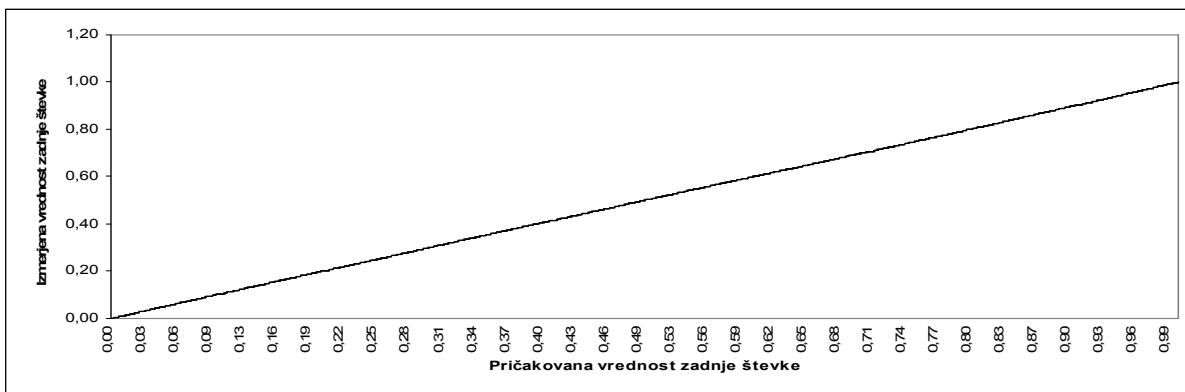
### B.2.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: 0°

<b>Št. meritev</b>	2039
<b>Celotni čas (s)</b>	711,65
<b>Povpr. čas (s)</b>	0,349
<b>Maks. neg. odstop.</b>	0,0033
<b>Maks. poz. odstop.</b>	0,0034
<b>Maks. napaka</b>	0,0034
<b>Min. % ujemanja</b>	91,190



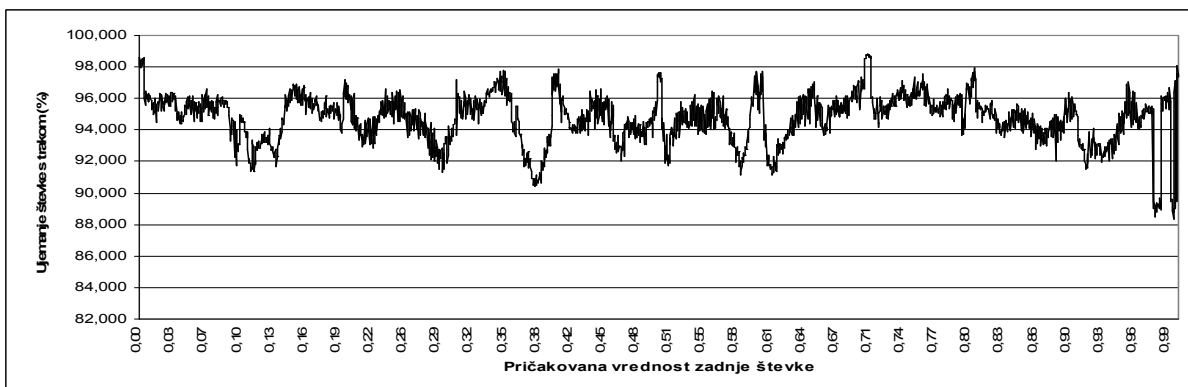
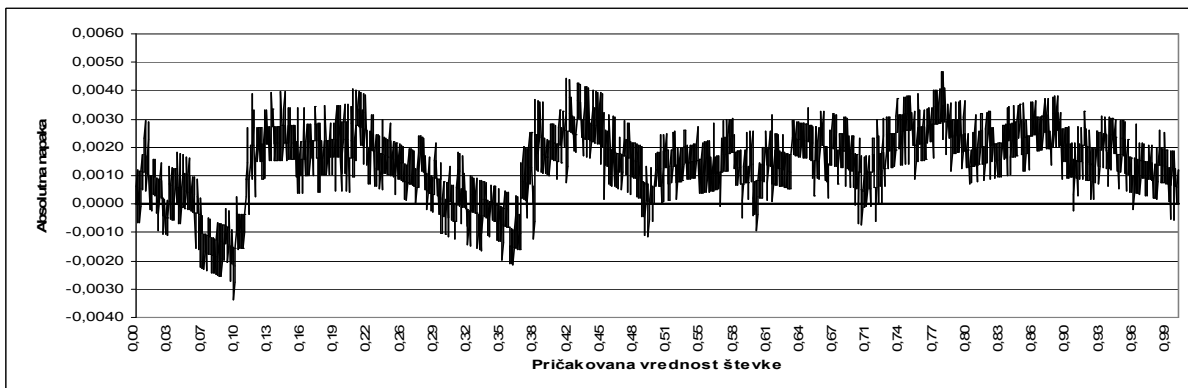
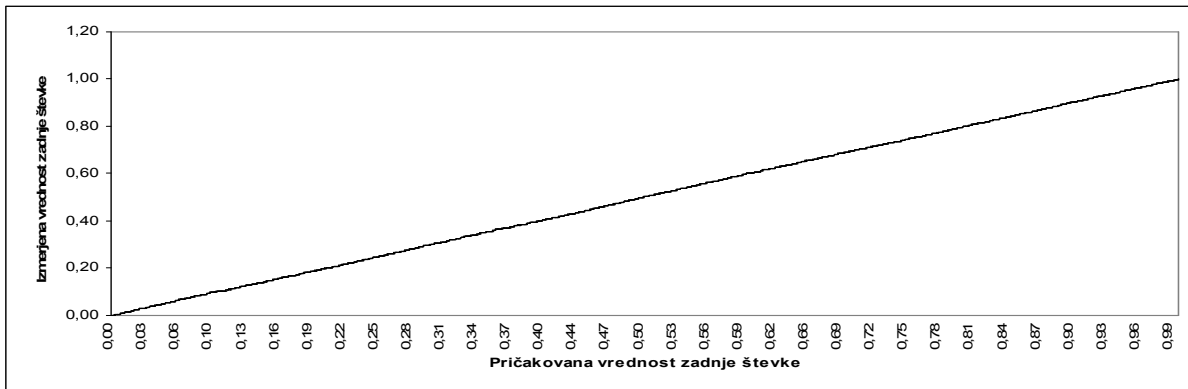
**B.2.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	2198
<b>Celotni čas (s)</b>	723,69
<b>Povpr. čas (s)</b>	0,329
<b>Maks. neg. odstop.</b>	0,0029
<b>Maks. poz. odstop.</b>	0,0047
<b>Maks. napaka</b>	0,0047
<b>Min. % ujemanja</b>	90,351



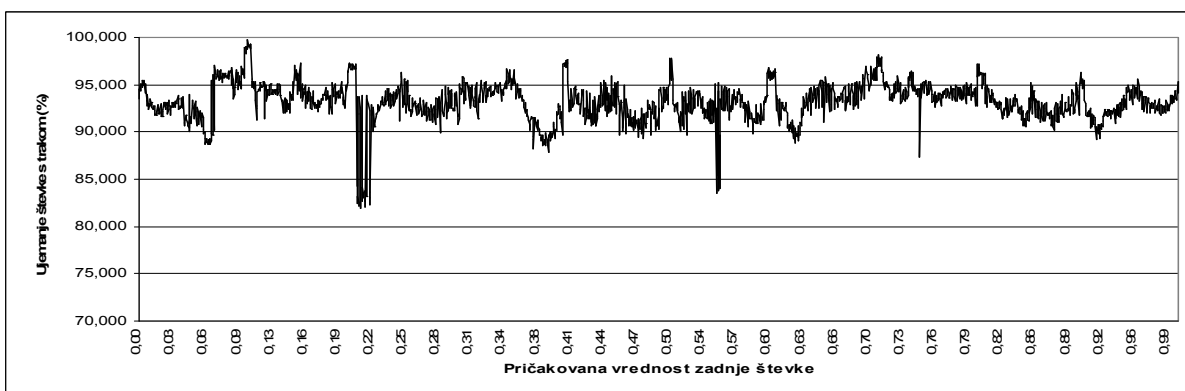
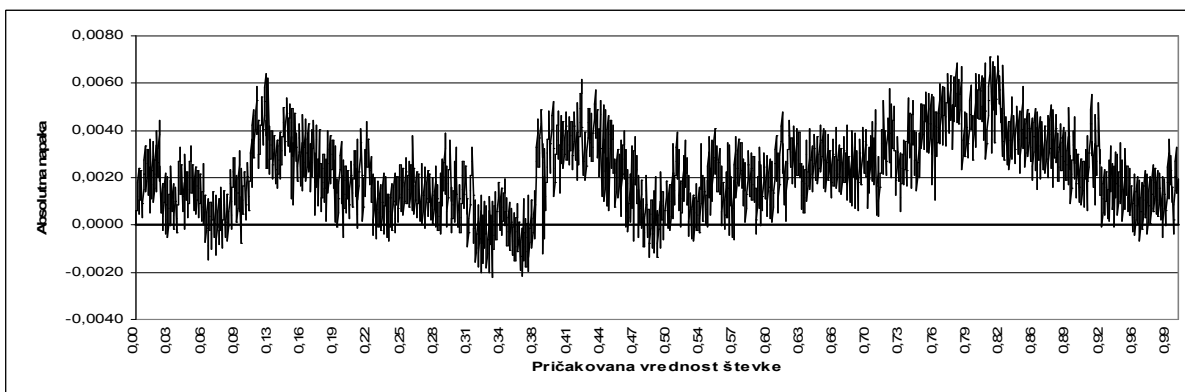
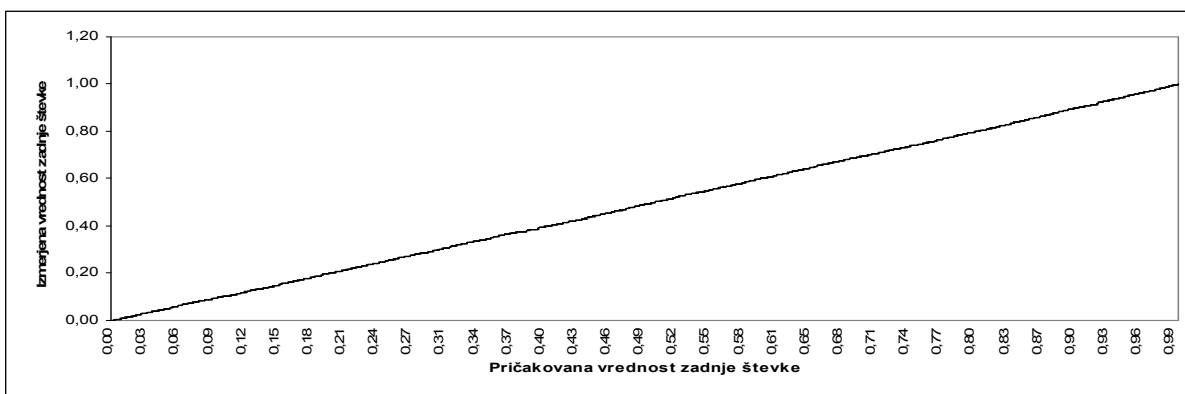
### B.2.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števniaka: 0°

<b>Št. meritev</b>	2225
<b>Celotni čas (s)</b>	711,22
<b>Povpr. čas (s)</b>	0,320
<b>Maks. neg. odstop.</b>	0,0034
<b>Maks. poz. odstop.</b>	0,0047
<b>Maks. napaka</b>	0,0047
<b>Min. % ujemanja</b>	88,331



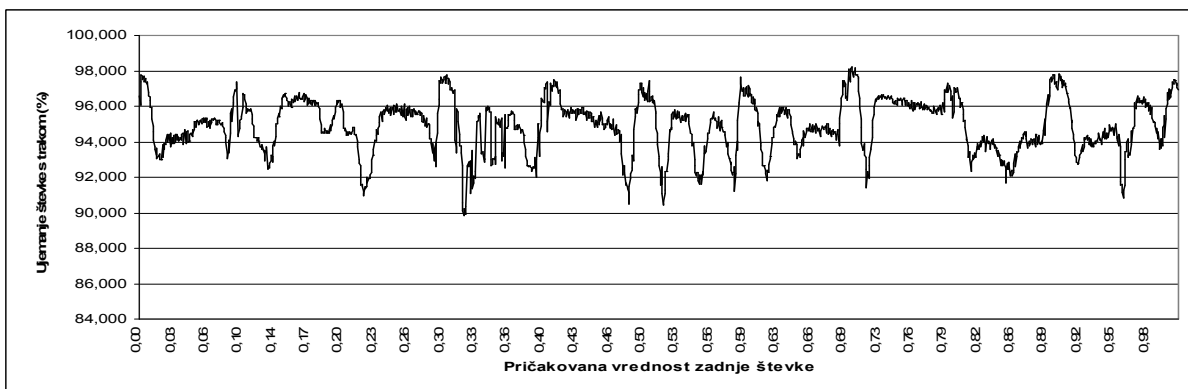
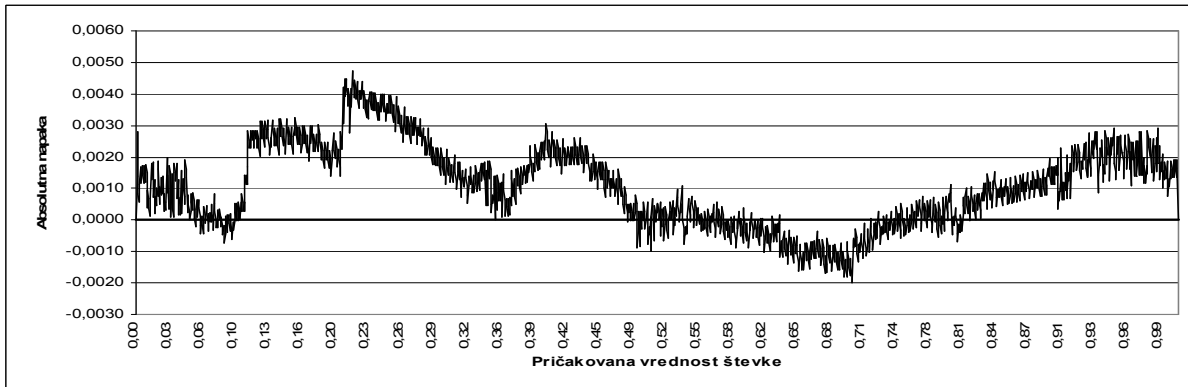
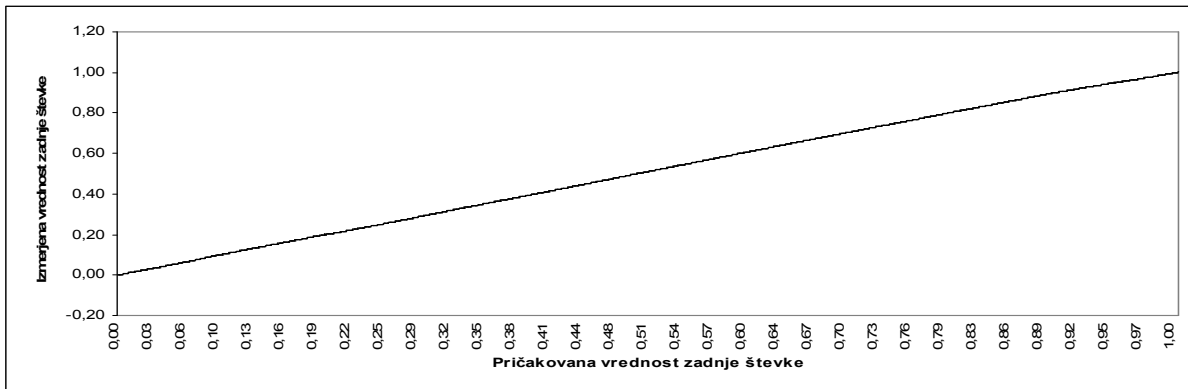
**B.2.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	2384
<b>Celotni čas (s)</b>	705,63
<b>Povpr. čas (s)</b>	0,296
<b>Maks. neg. odstop.</b>	0,0022
<b>Maks. poz. odstop.</b>	0,0071
<b>Maks. napaka</b>	0,0071
<b>Min. % ujemanja</b>	81,912



### B.2.8 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: +5°

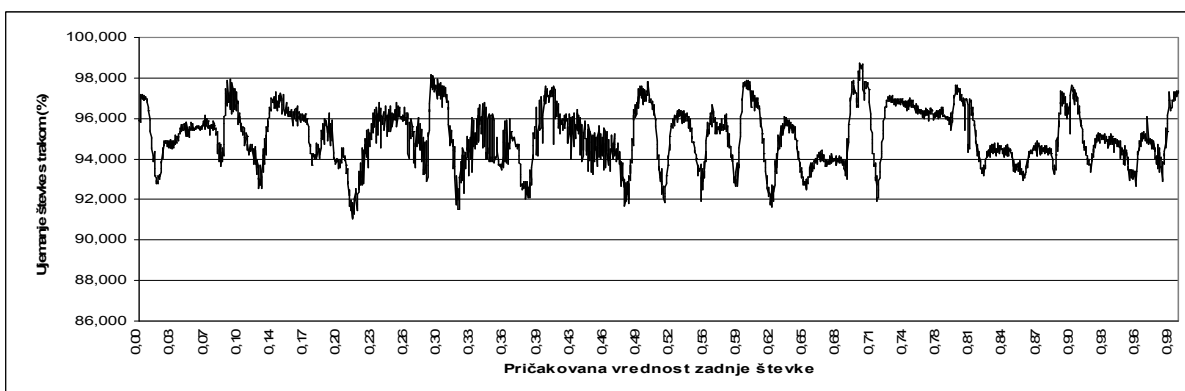
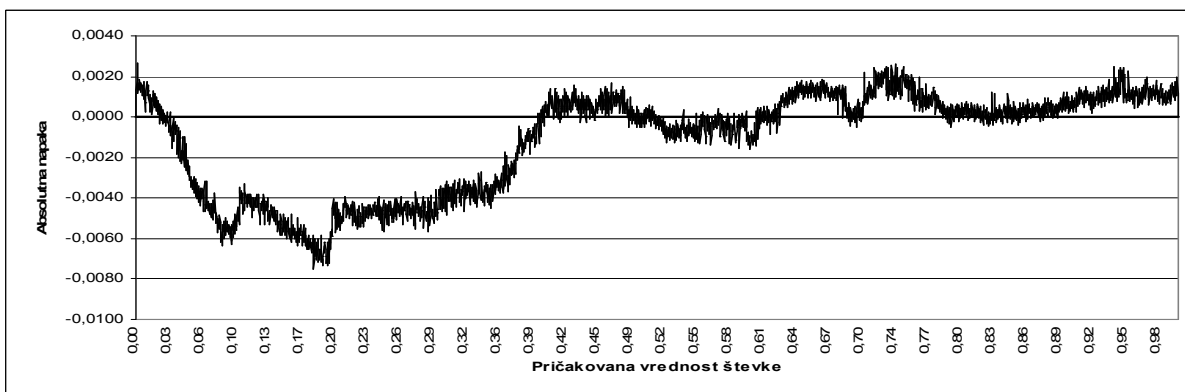
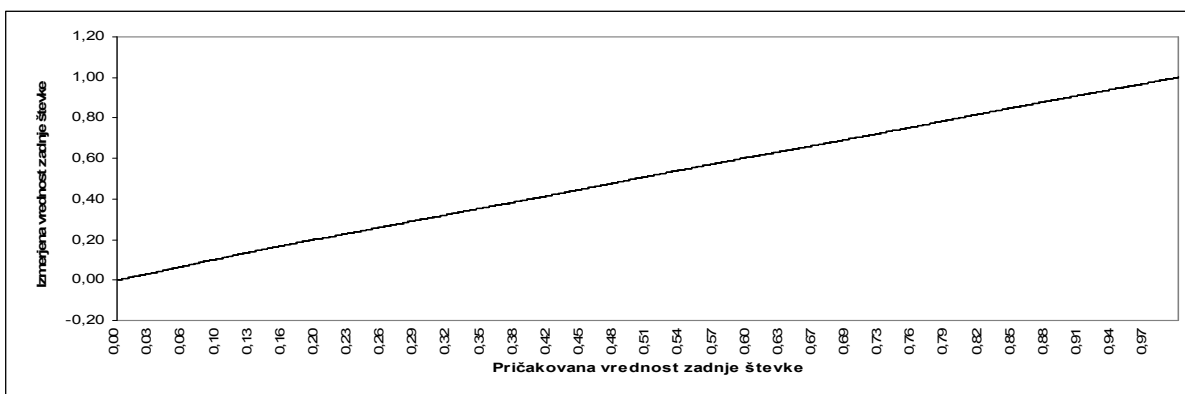
<b>Št. meritev</b>	1641
<b>Celotni čas (s)</b>	769,7
<b>Povpr. čas (s)</b>	0,469
<b>Maks. neg. odstop.</b>	0,0020
<b>Maks. poz. odstop.</b>	0,0047
<b>Maks. napaka</b>	0,0047
<b>Min. % ujemanja</b>	89,850





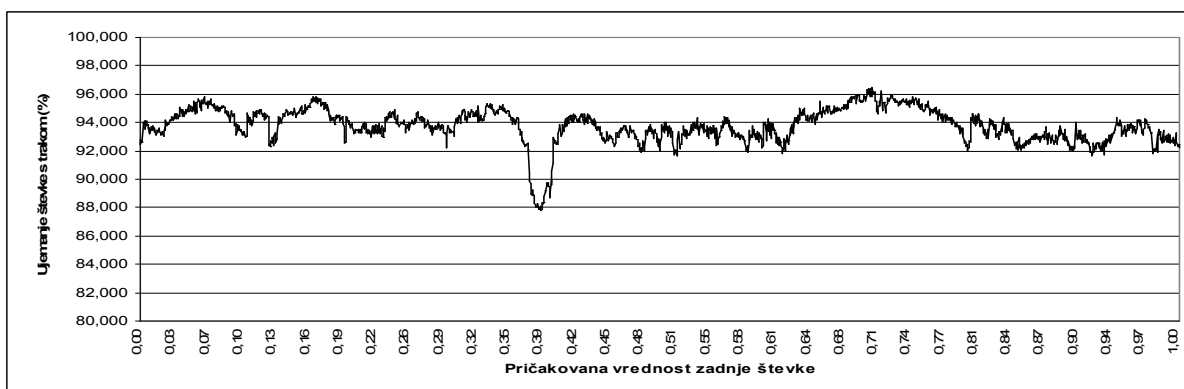
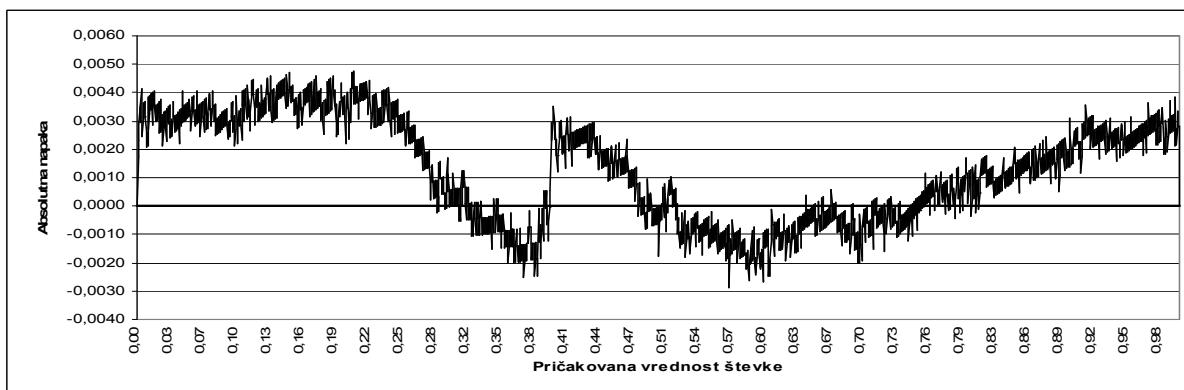
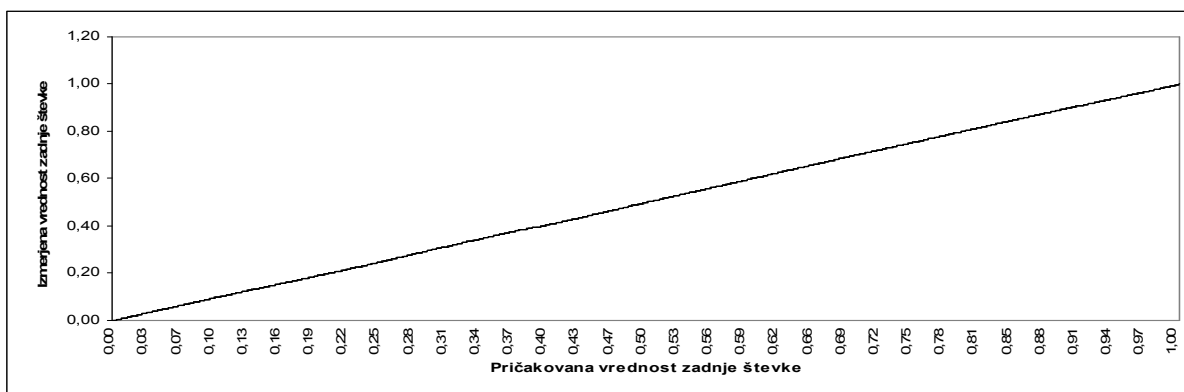
**B.2.9 Velikost števk na traku: 150 slikovnih elementov, rotacija števniaka: +2°5'**

Št.meritev	2625
Celotni čas (s)	877,2
Povpr. čas (s)	0,334
Maks. neg. odstop.	0,0075
Maks. poz. odstop.	0,0027
Maks. napaka	0,0075
Min. % ujemanja	91,045



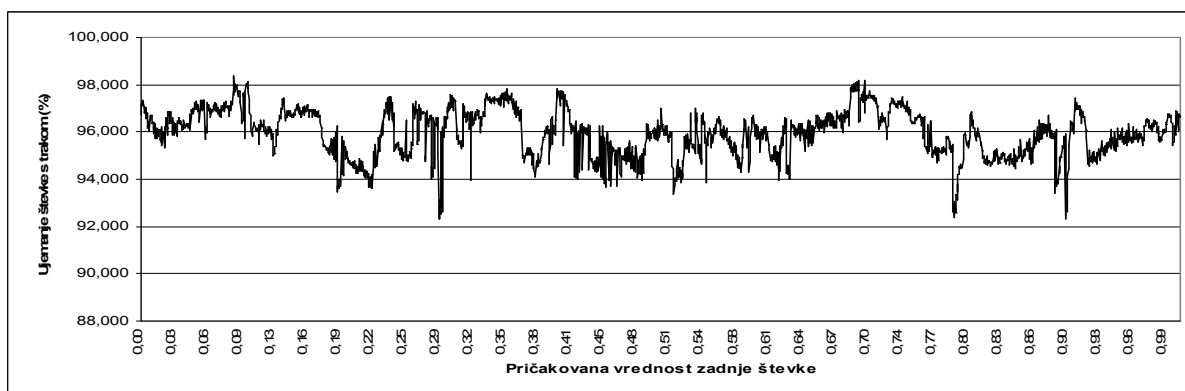
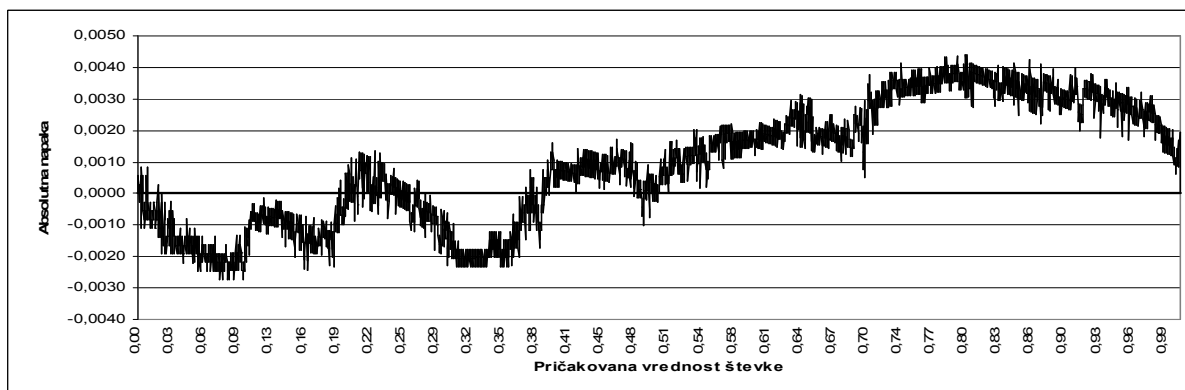
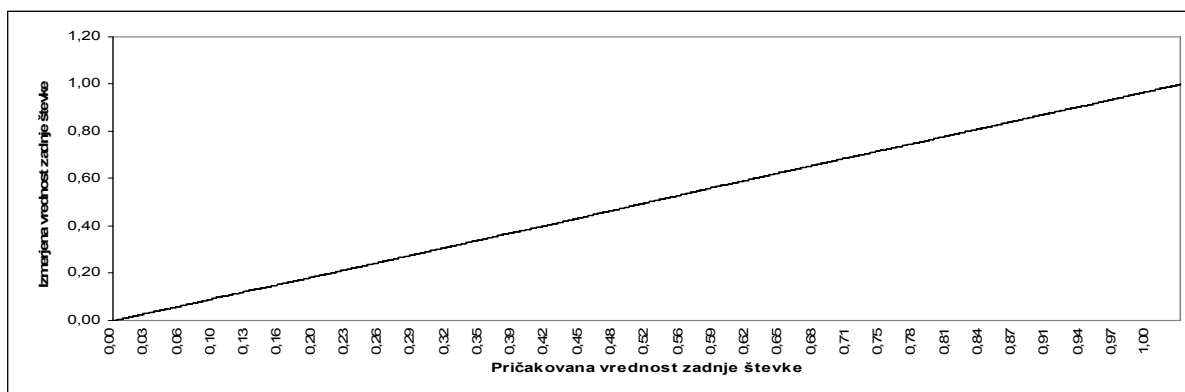
### B.2.10 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: -2'5°

Št. meritev	1992
Celotni čas (s)	729,71
Povpr. čas (s)	0,366
Maks. neg. odstop.	0,0029
Maks. poz. odstop.	0,0048
Maks. napaka	0,0048
Min. % ujemanja	87,819



**B.2.11 Velikost števk na traku: 150 slikovnih elementov, rotacija števniaka: -5°**

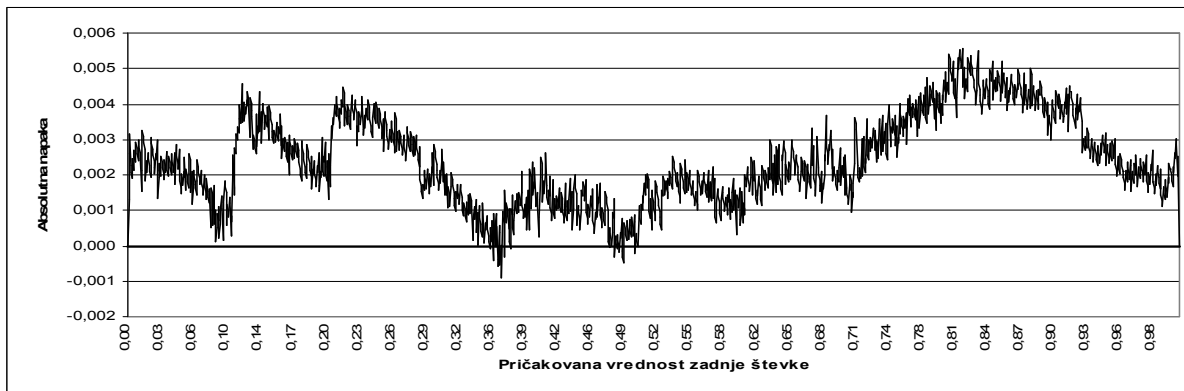
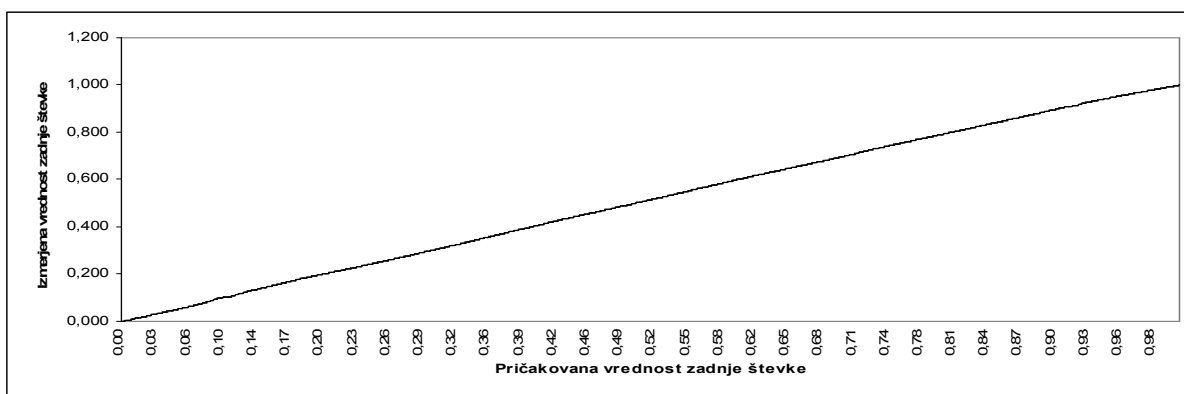
<b>Št.meritev</b>	2425
<b>Celotni čas (s)</b>	781,74
<b>Povpr. čas (s)</b>	0,322
<b>Maks. neg. odstop.</b>	0,0027
<b>Maks. poz. odstop.</b>	0,0044
<b>Maks. napaka</b>	0,0044
<b>Min. % ujemanja</b>	92,303



### B.3 Meritve pri velikosti števnik 130 slikovnih elementov

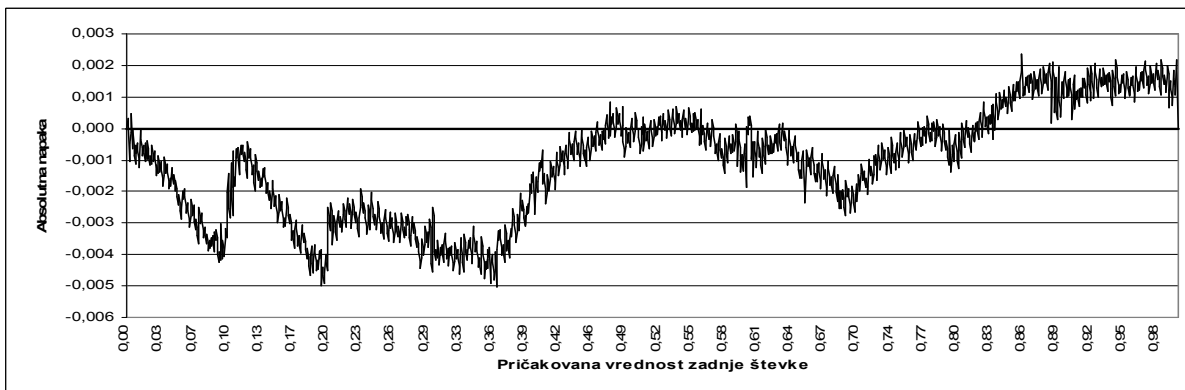
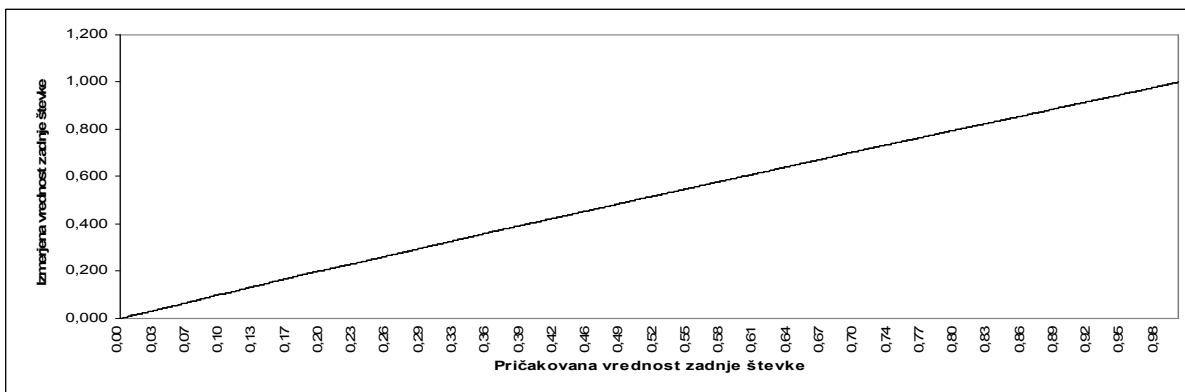
#### B.3.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: 0°

Št. meritev	1590
Celotni čas (s)	759,38
Povpr. čas (s)	0,478
Maks. neg. odstop.	0,0009
Maks. poz. odstop.	0,0056
Maks. napaka	0,0056
Min. % ujemanja	89,084



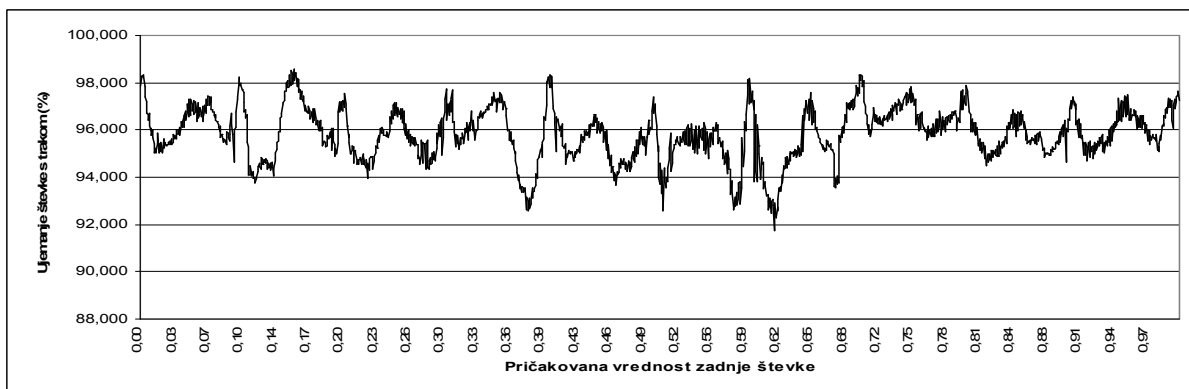
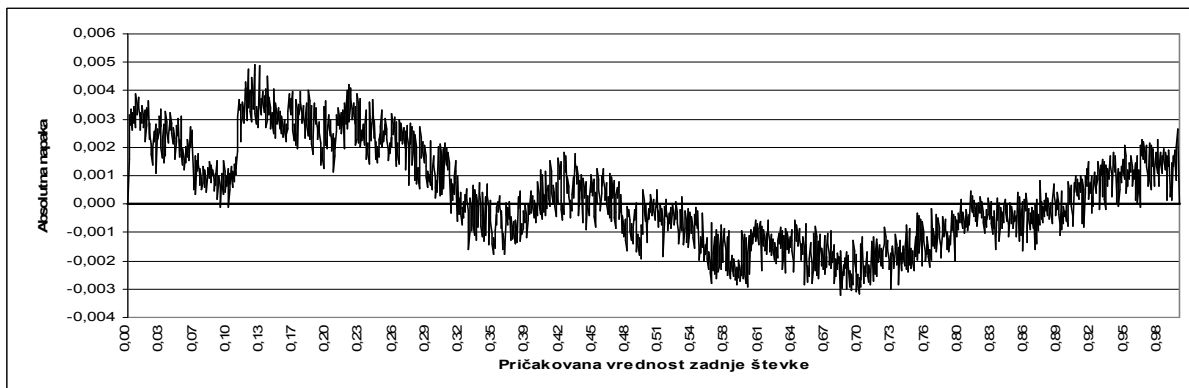
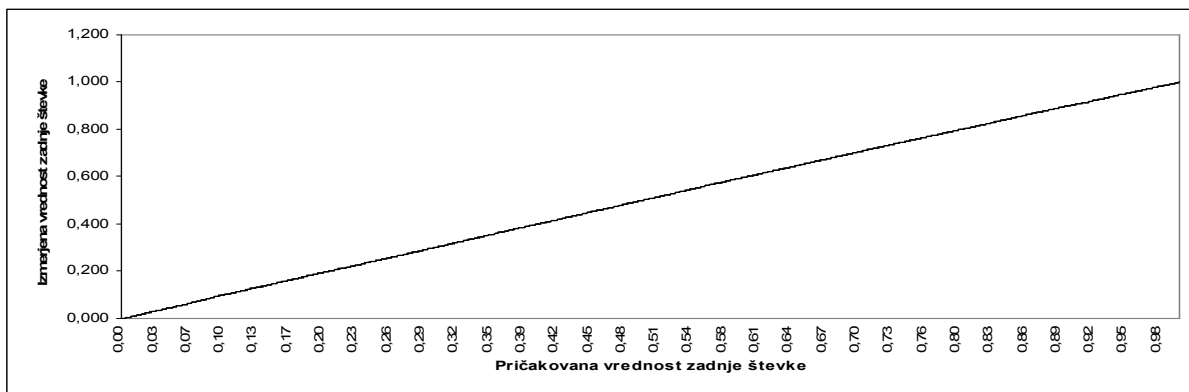
**B.3.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	1394
<b>Celotni čas (s)</b>	749,92
<b>Povpr. čas (s)</b>	0,538
<b>Maks. neg. odstop.</b>	0,0050
<b>Maks. poz. odstop.</b>	0,0024
<b>Maks. napaka</b>	0,0050
<b>Min. % ujemanja</b>	92,313



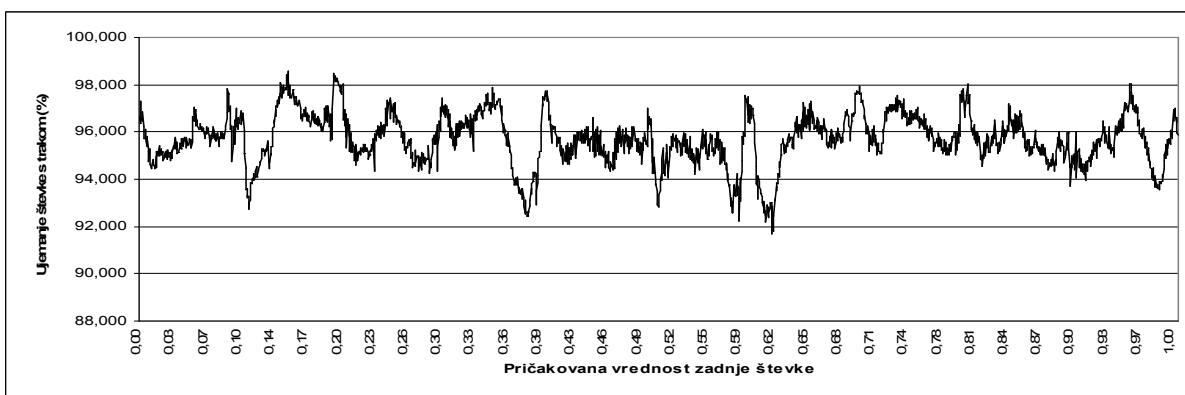
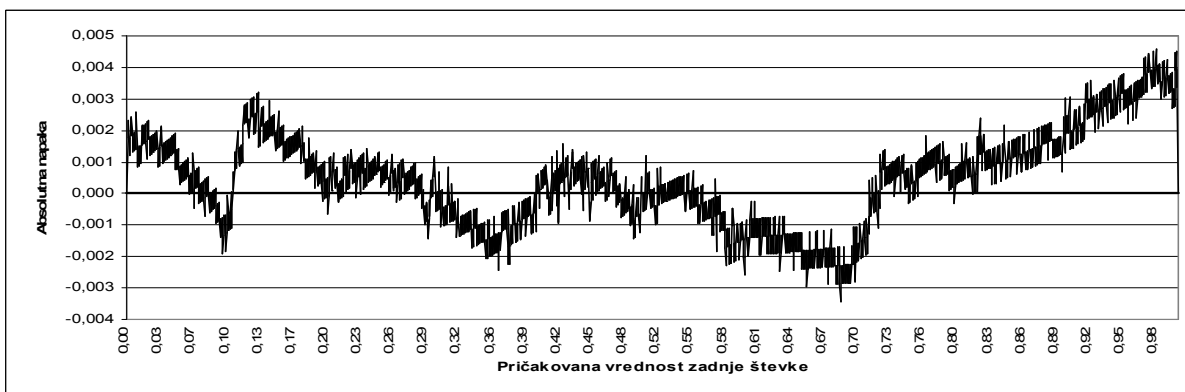
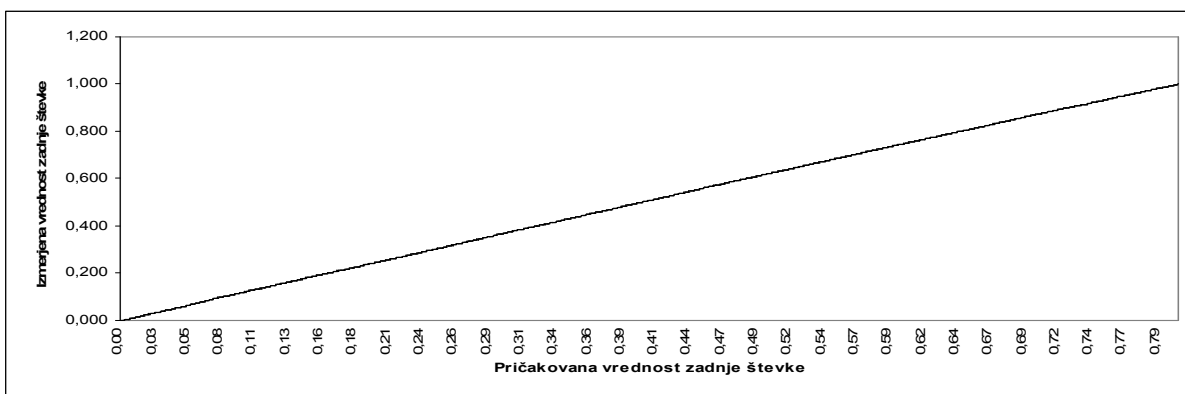
### B.3.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: 0°

<b>Št. meritev</b>	1550
<b>Celotni čas (s)</b>	712,51
<b>Povpr. čas (s)</b>	0,460
<b>Maks. neg. odstop.</b>	0,0032
<b>Maks. poz. odstop.</b>	0,0049
<b>Maks. napaka</b>	0,0049
<b>Min. % ujemanja</b>	91,755



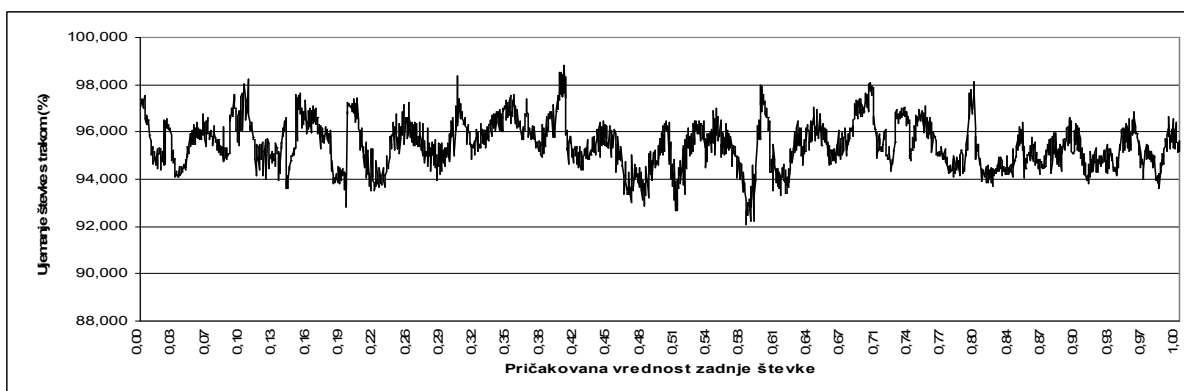
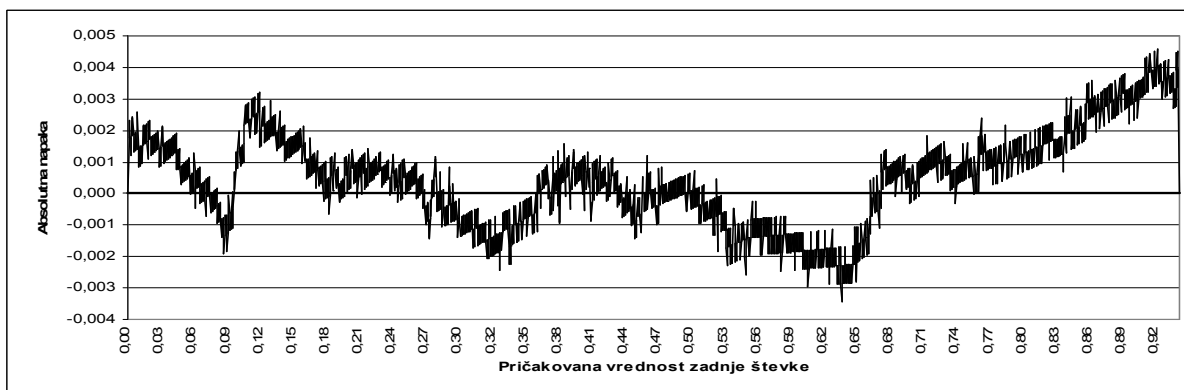
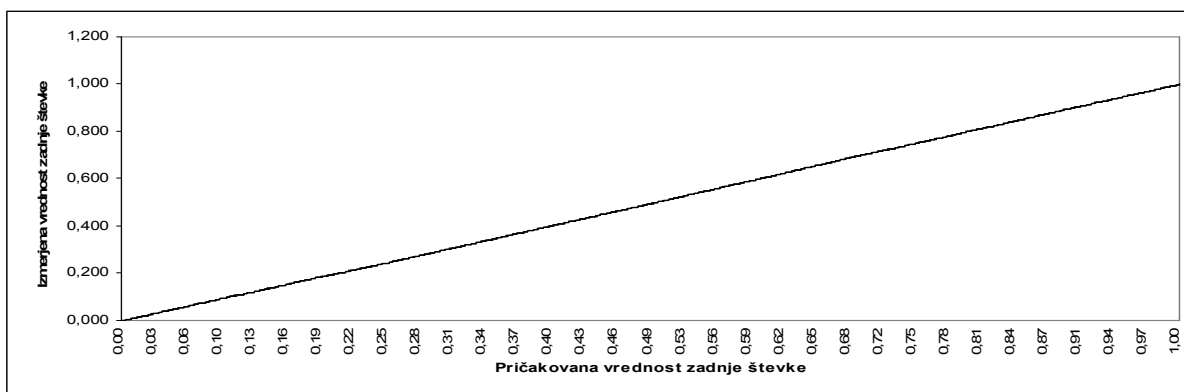
**B.3.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnika: 0°**

<b>Št.meritev</b>	1934
<b>Celotni čas (s)</b>	727,99
<b>Povpr. čas (s)</b>	0,376
<b>Maks. neg. odstop.</b>	0,0034
<b>Maks. poz. odstop.</b>	0,0046
<b>Maks. napaka</b>	0,0046
<b>Min. % ujemanja</b>	91,694



### B.3.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števniaka: 0°

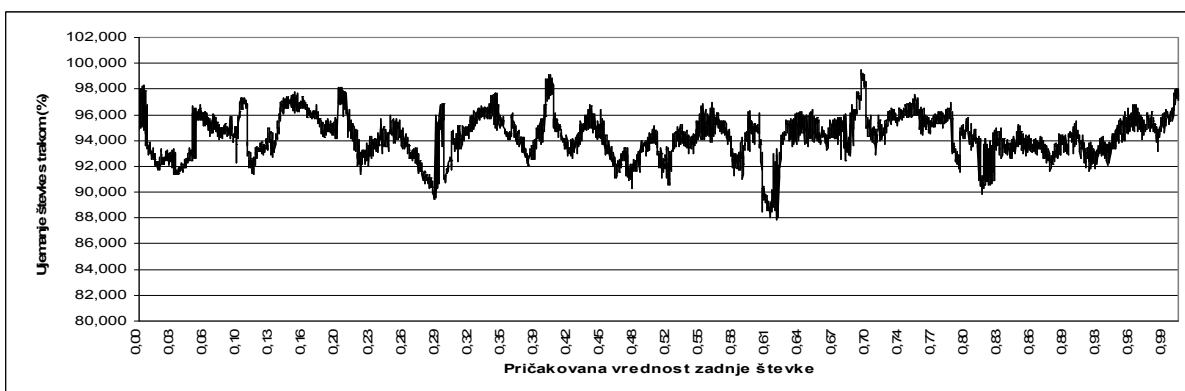
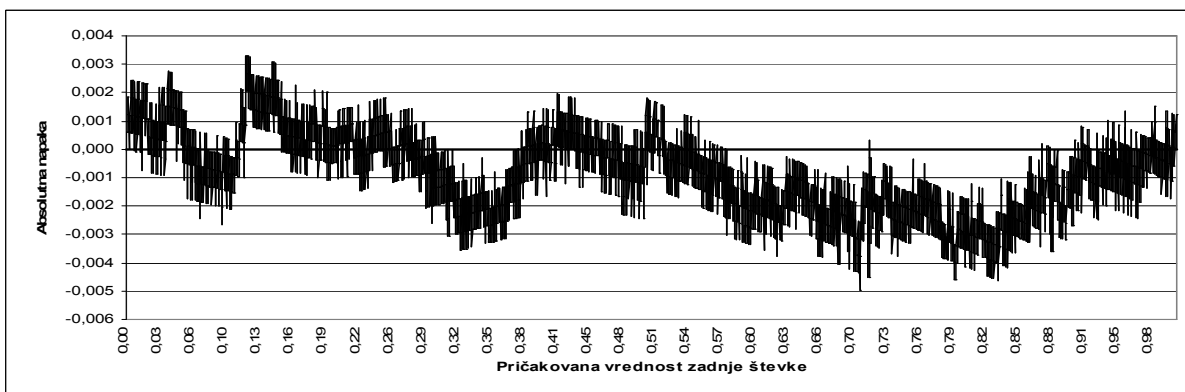
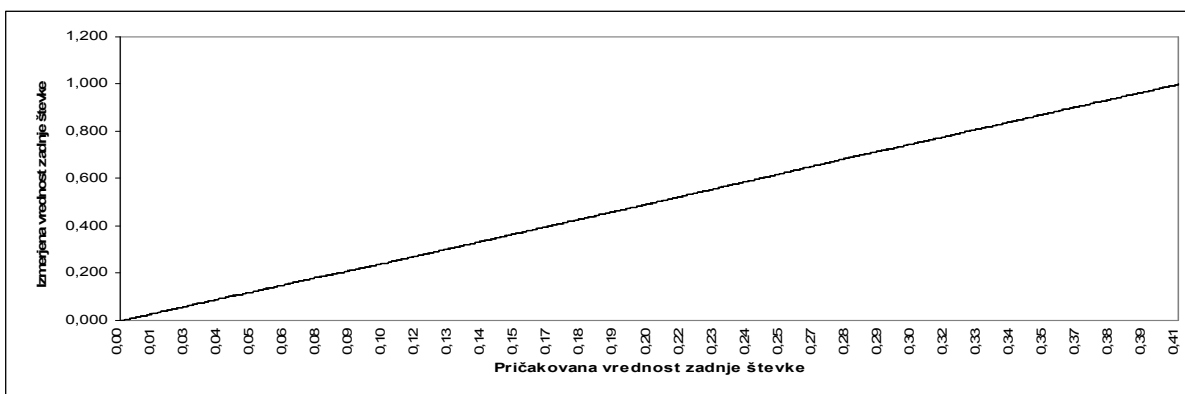
<b>Št. meritev</b>	2052
<b>Celotni čas (s)</b>	730,14
<b>Povpr. čas (s)</b>	0,356
<b>Maks. neg. odstop.</b>	0,0038
<b>Maks. poz. odstop.</b>	0,0076
<b>Maks. napaka</b>	0,0076
<b>Min. % ujemanja</b>	92,086





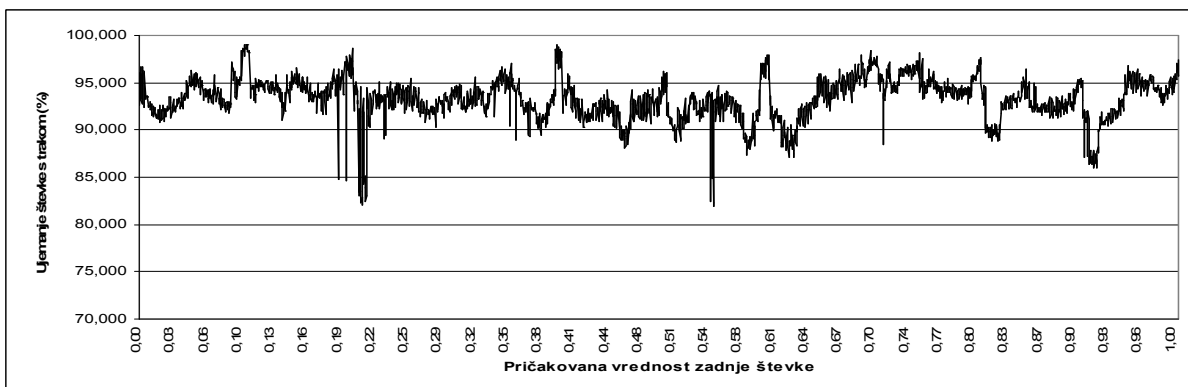
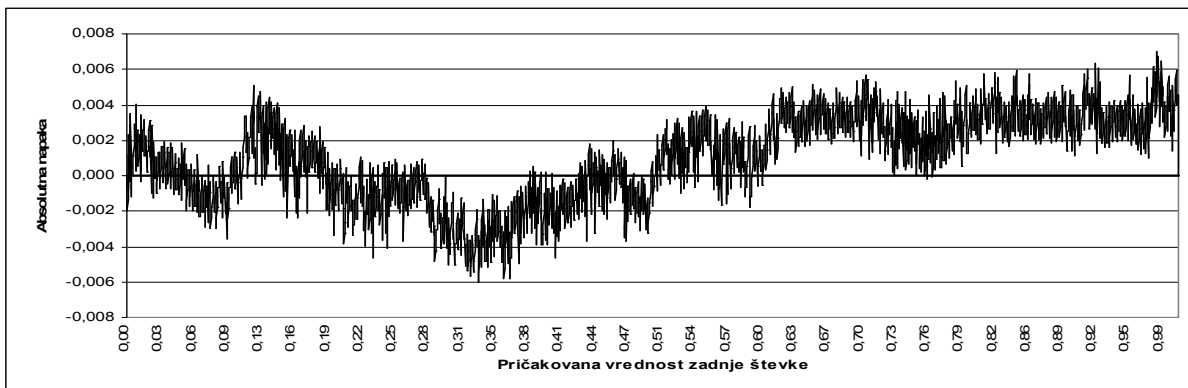
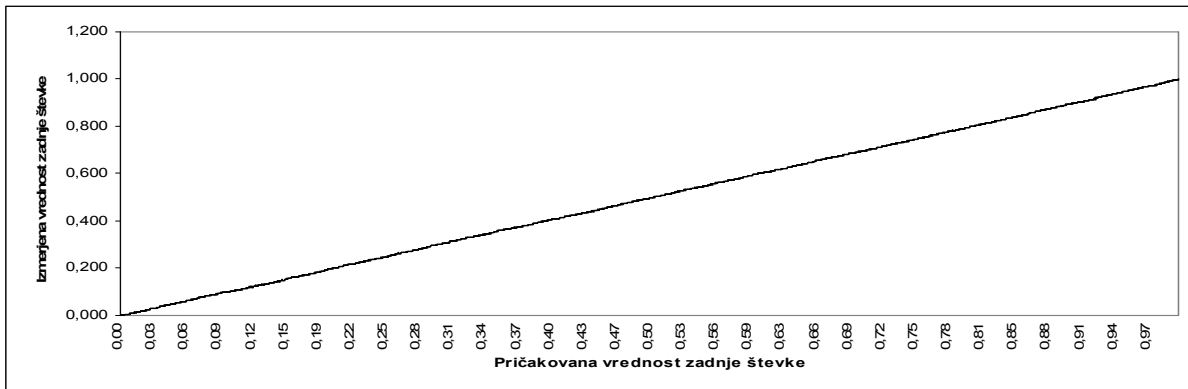
**B.3.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	5121
<b>Celotni čas (s)</b>	1653
<b>Povpr. čas (s)</b>	0,323
<b>Maks. neg. odstop.</b>	0,0050
<b>Maks. poz. odstop.</b>	0,0033
<b>Maks. napaka</b>	0,0050
<b>Min. % ujemanja</b>	87,806



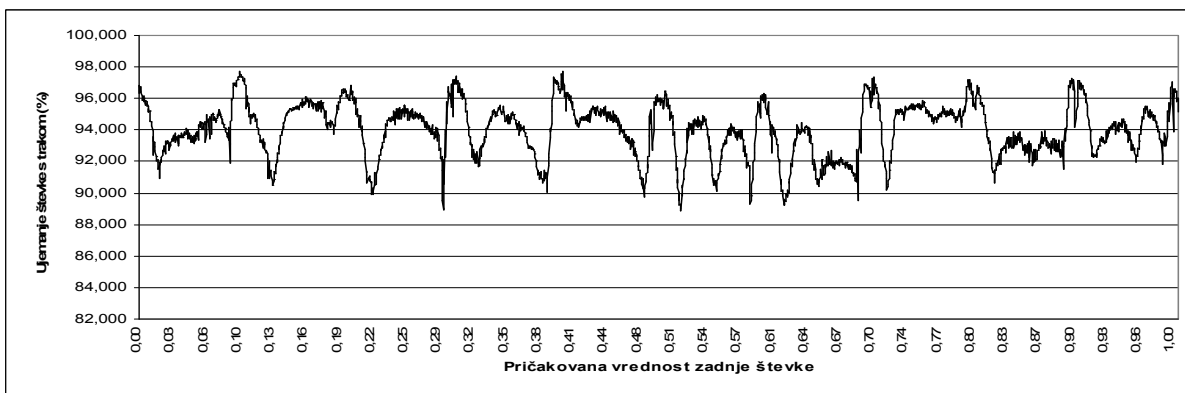
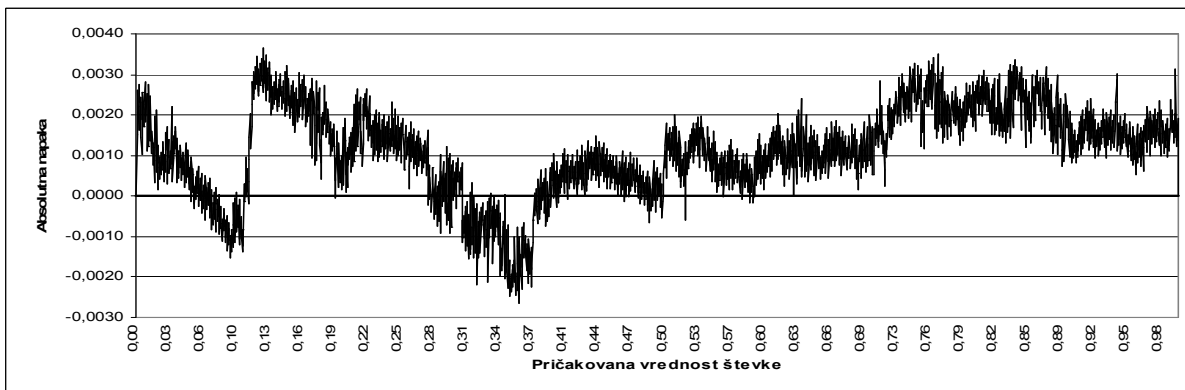
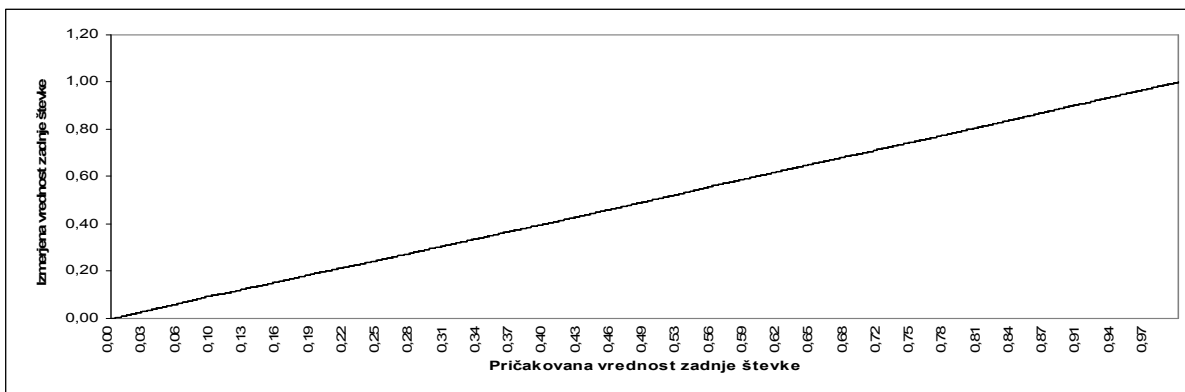
### B.3.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števniaka: 0°

<b>Št. meritev</b>	2649
<b>Celotni čas (s)</b>	744,76
<b>Povpr. čas (s)</b>	0,281
<b>Maks. neg. odstop.</b>	0,0060
<b>Maks. poz. odstop.</b>	0,0070
<b>Maks. napaka</b>	0,0070
<b>Min. % ujemanja</b>	81,975



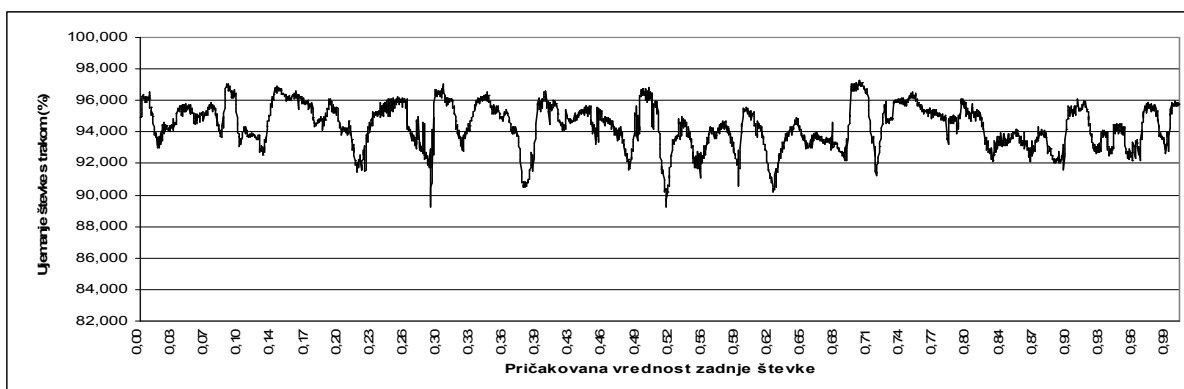
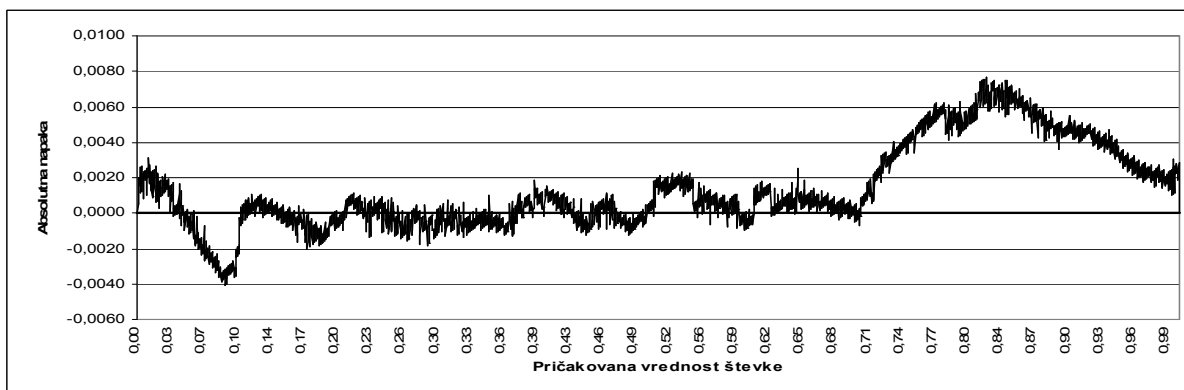
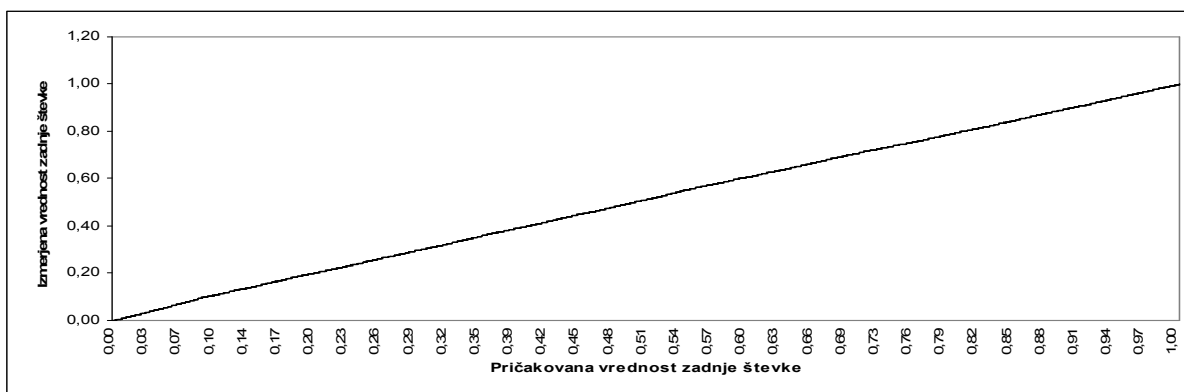
**B.3.8 Velikost števk na traku: 130 slikovnih elementov, rotacija števnika: +5°**

<b>Št.meritev</b>	2432
<b>Celotni čas (s)</b>	774
<b>Povpr. čas (s)</b>	0,318
<b>Maks. neg. odstop.</b>	0,0027
<b>Maks. poz. odstop.</b>	0,0036
<b>Maks. napaka</b>	0,0036
<b>Min. % ujemanja</b>	88,881



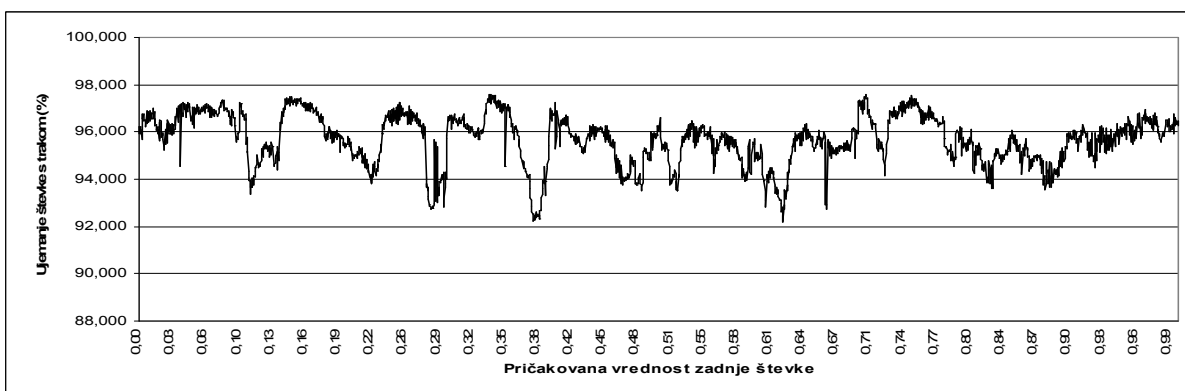
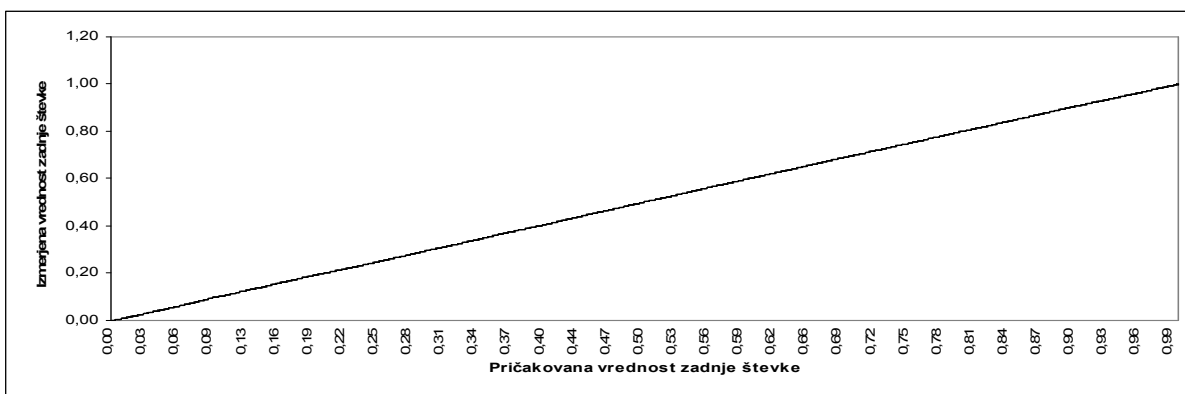
### B.3.9 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: +2°5'

<b>Št. meritev</b>	2701
<b>Celotni čas (s)</b>	831,19
<b>Povpr. čas (s)</b>	0,308
<b>Maks. neg. odstop.</b>	0,0041
<b>Maks. poz. odstop.</b>	0,0077
<b>Maks. napaka</b>	0,0077
<b>Min. % ujemanja</b>	89,257



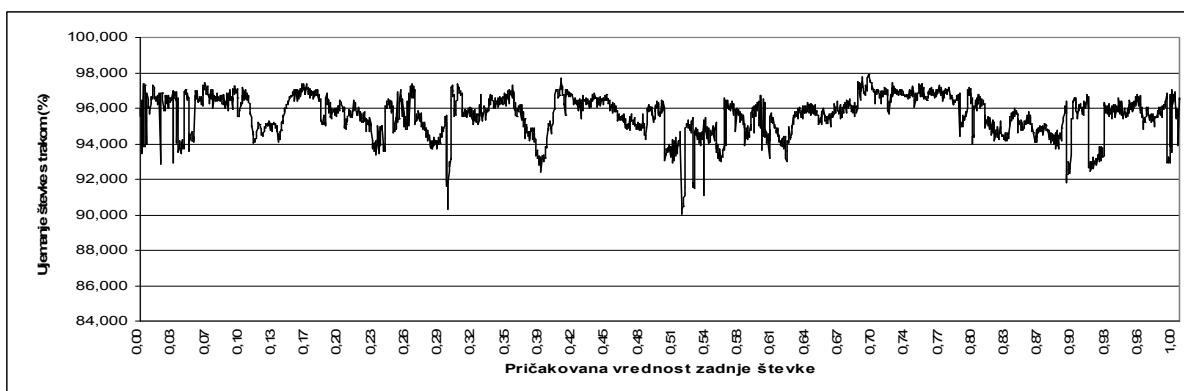
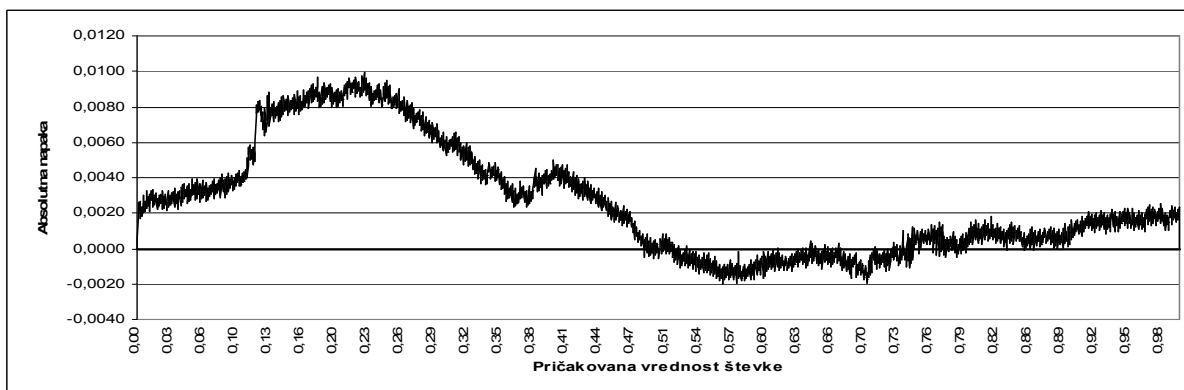
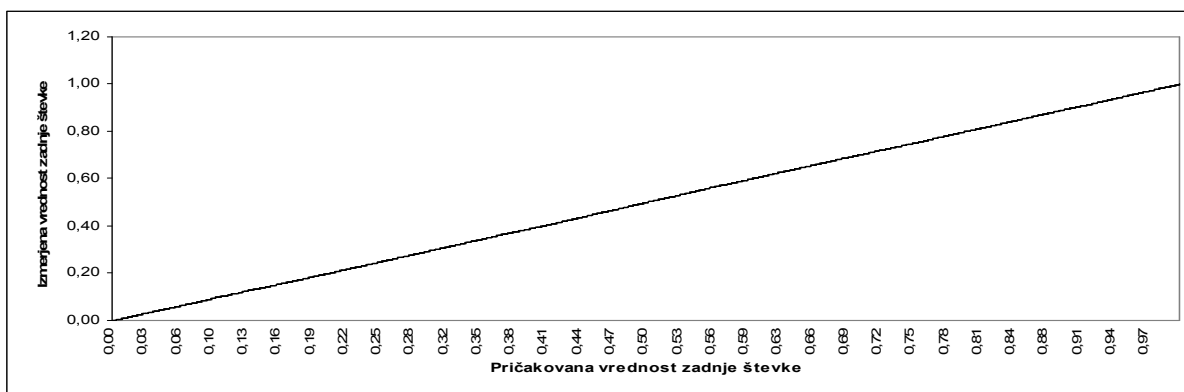
**B.3.10 Velikost števk na traku: 130 slikovnih elementov, rotacija števniaka: -2'5°**

<b>Št.meritev</b>	2191
<b>Celotni čas (s)</b>	709,07
<b>Povpr. čas (s)</b>	0,324
<b>Maks. neg. odstop.</b>	0,0029
<b>Maks. poz. odstop.</b>	0,0039
<b>Maks. napaka</b>	0,0039
<b>Min. % ujemanja</b>	92,178



### B.3.11 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: -5°

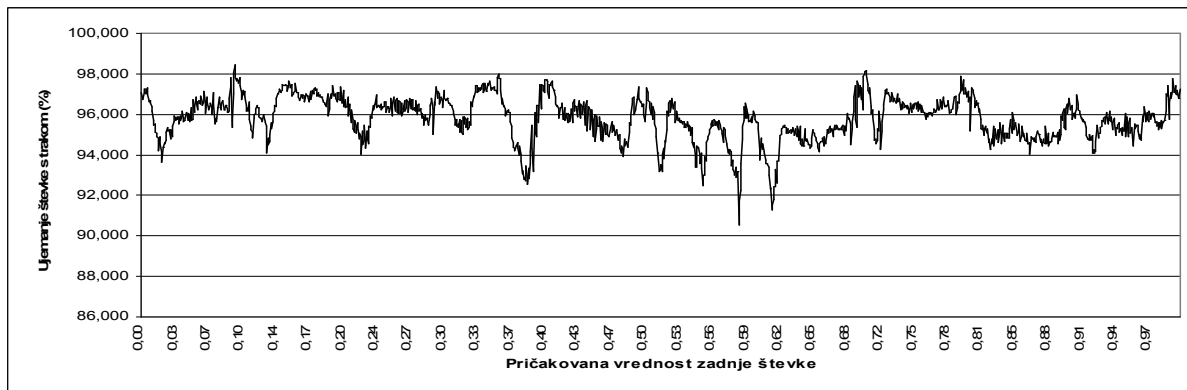
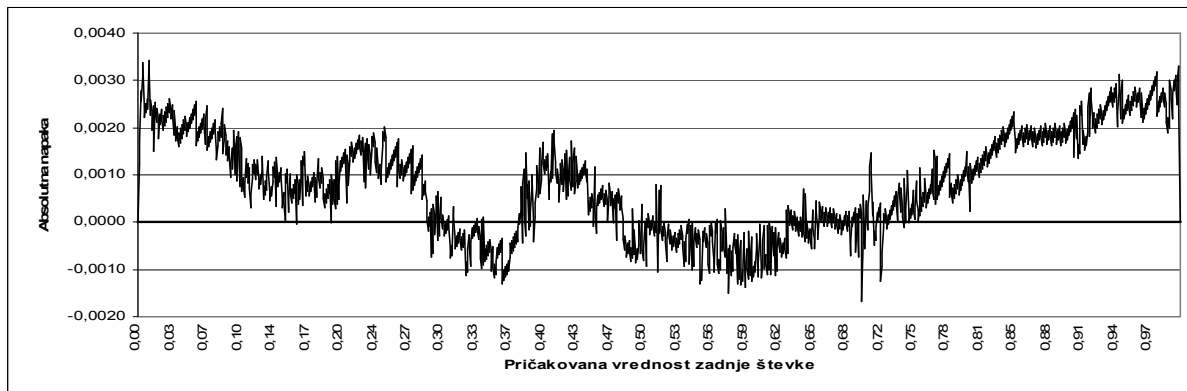
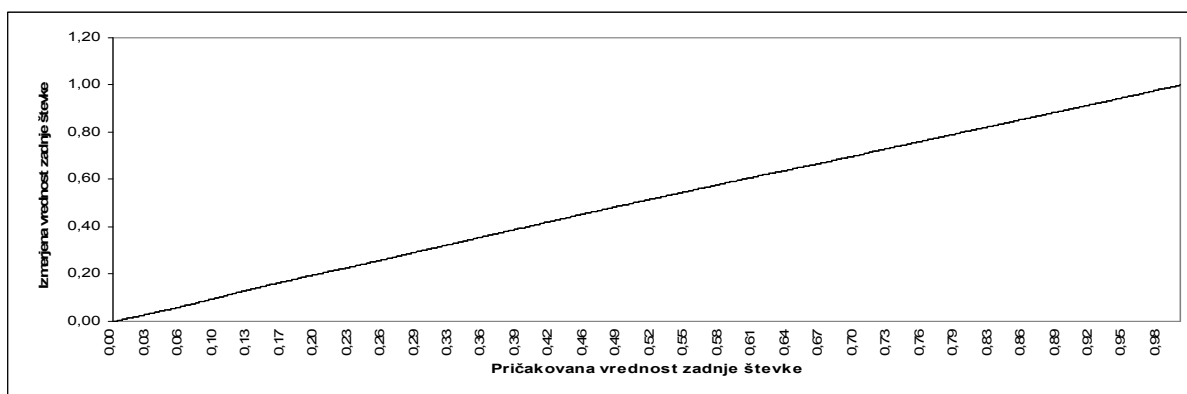
Št. meritev	2495
Celotni čas (s)	786,9
Povpr. čas (s)	0,315
Maks. neg. odstop.	0,0020
Maks. poz. odstop.	0,0099
Maks. napaka	0,0099
Min. % ujemanja	90,058



## B.4 Meritve pri velikosti števnik 110 slikovnih elementov

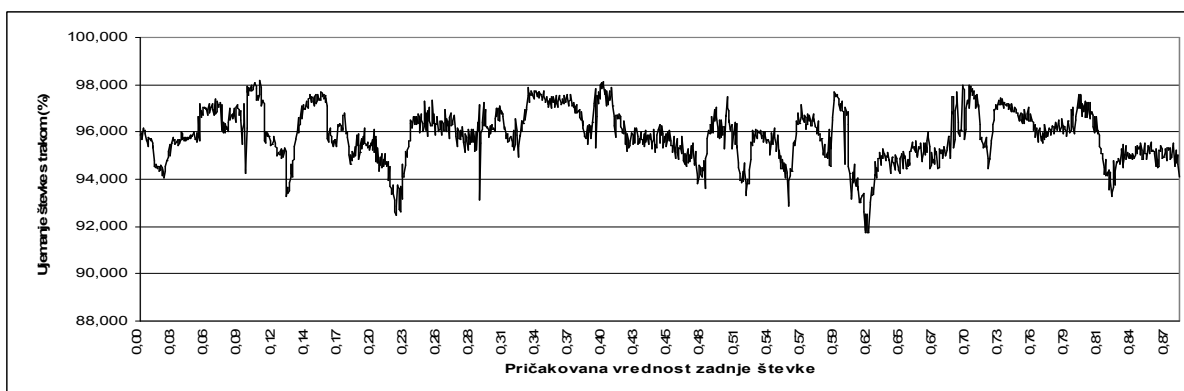
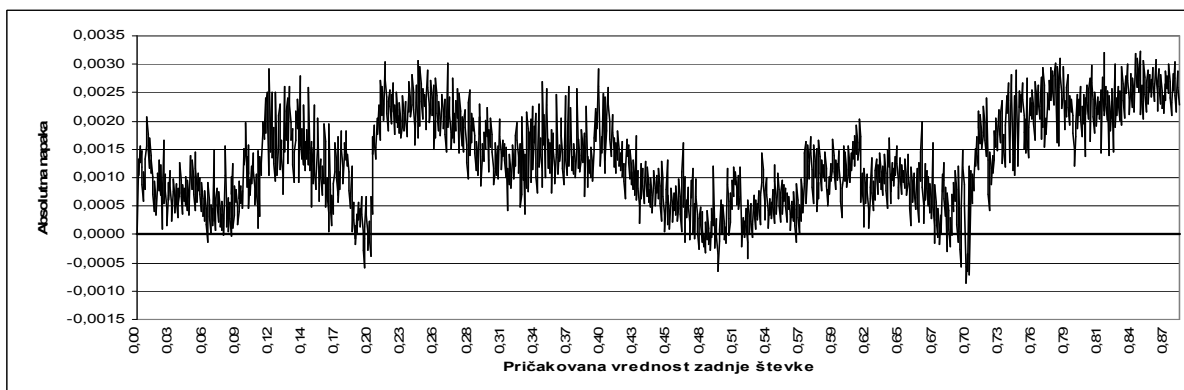
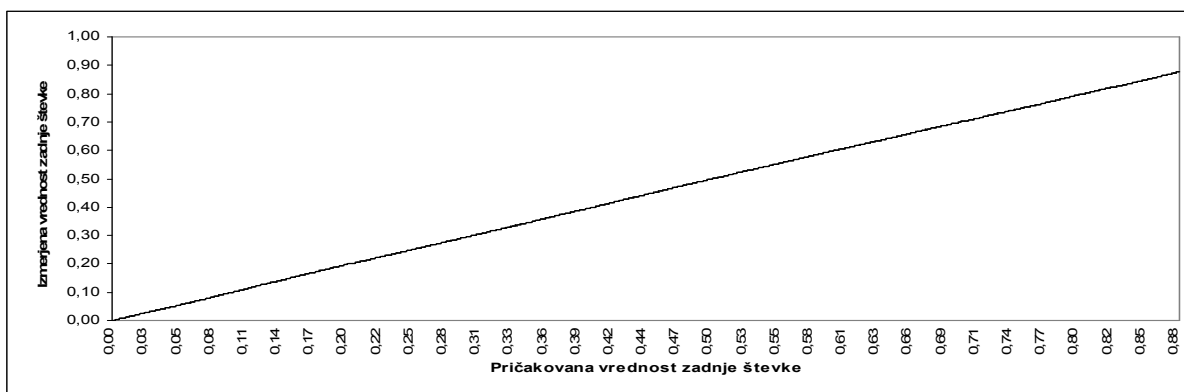
### B.4.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: 0°

Št.meritev	1330
Celotni čas (s)	836,35
Povpr. čas (s)	0,629
Maks. neg. odstop.	0,0017
Maks. poz. odstop.	0,0034
Maks. napaka	0,0034
Min. % ujemanja	90,552



### B.4.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: 0°

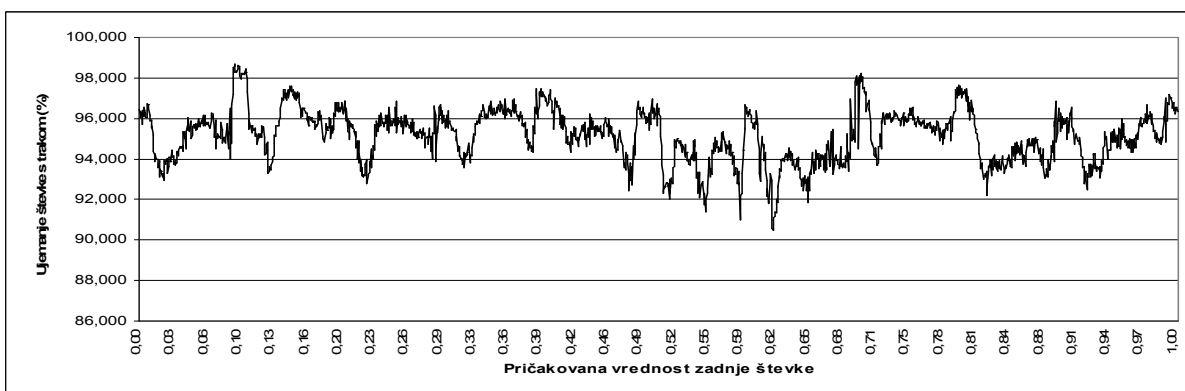
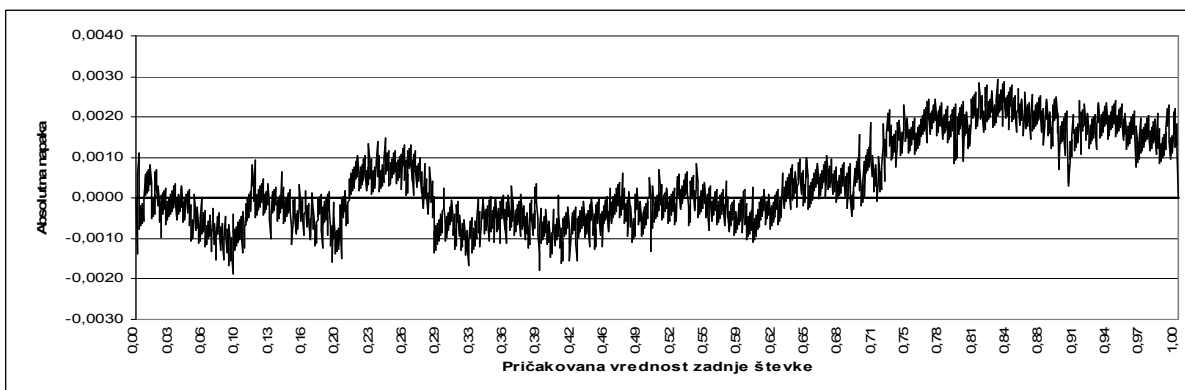
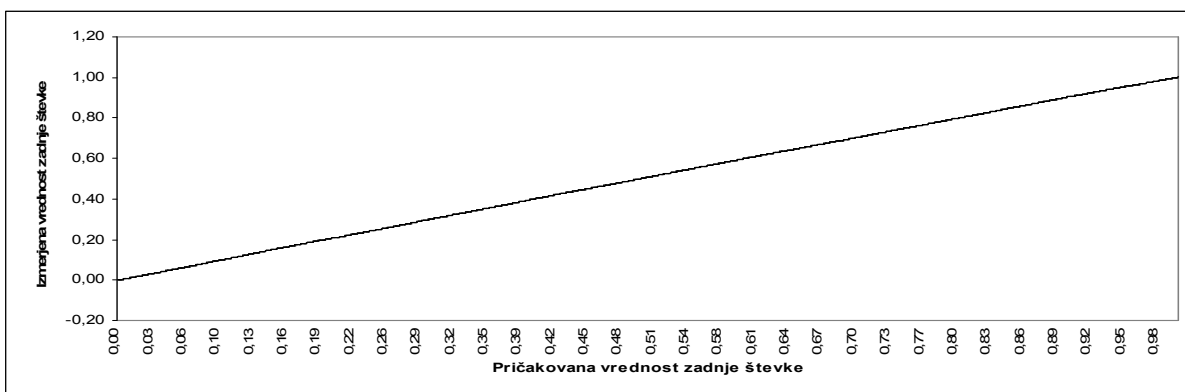
<b>Št. meritev</b>	1644
<b>Celotni čas (s)</b>	824,31
<b>Povpr. čas (s)</b>	0,501
<b>Maks. neg. odstop.</b>	0,0009
<b>Maks. poz. odstop.</b>	0,0032
<b>Maks. napaka</b>	0,0032
<b>Min. % ujemanja</b>	91,720





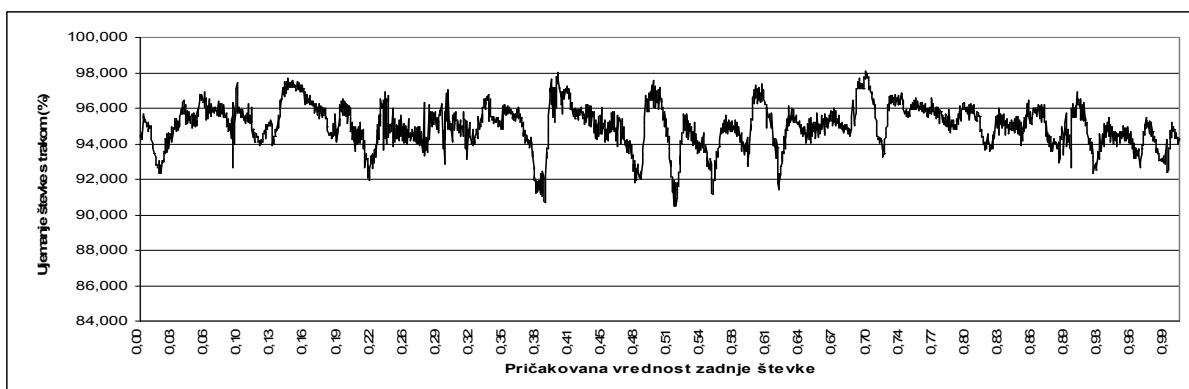
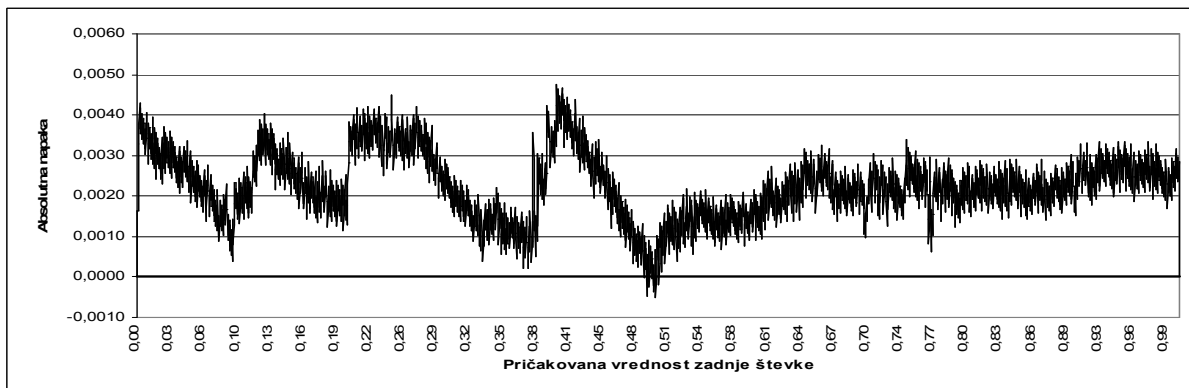
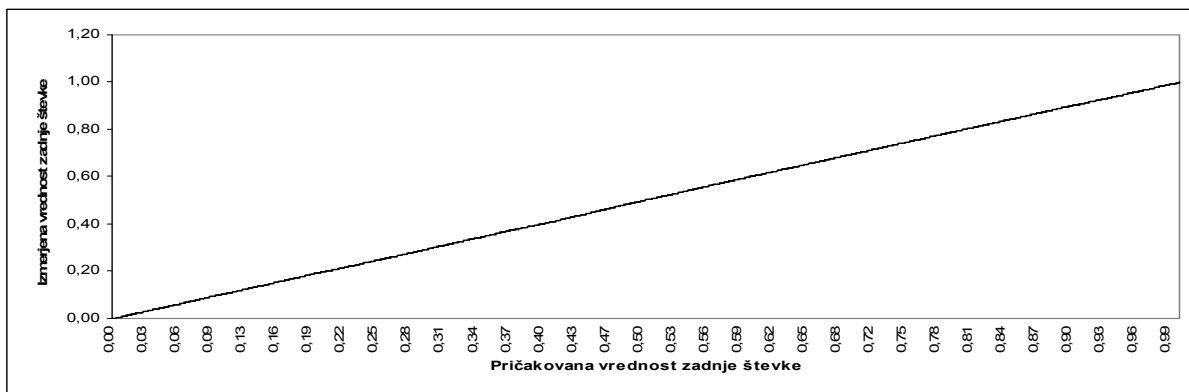
**B.4.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števniaka: 0°**

<b>Št.meritev</b>	1805
<b>Celotni čas (s)</b>	841,08
<b>Povpr. čas (s)</b>	0,466
<b>Maks. neg. odstop.</b>	0,0019
<b>Maks. poz. odstop.</b>	0,0029
<b>Maks. napaka</b>	0,0029
<b>Min. % ujemanja</b>	90,463



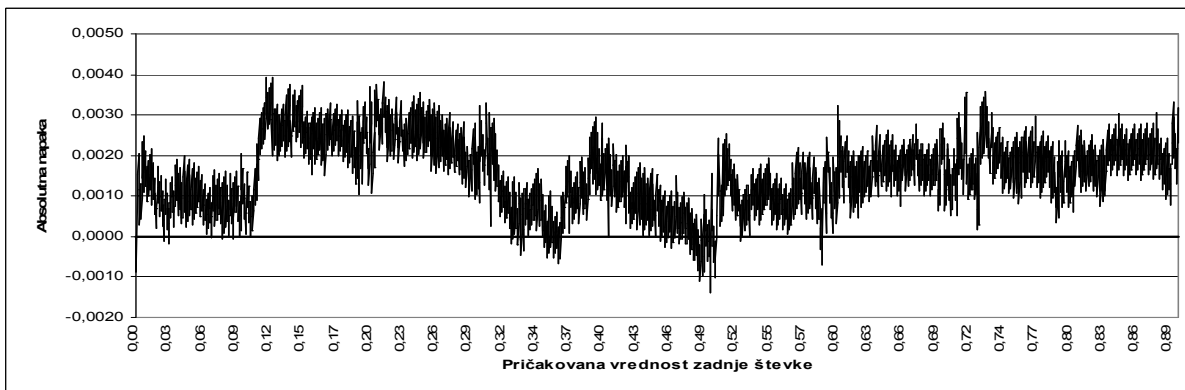
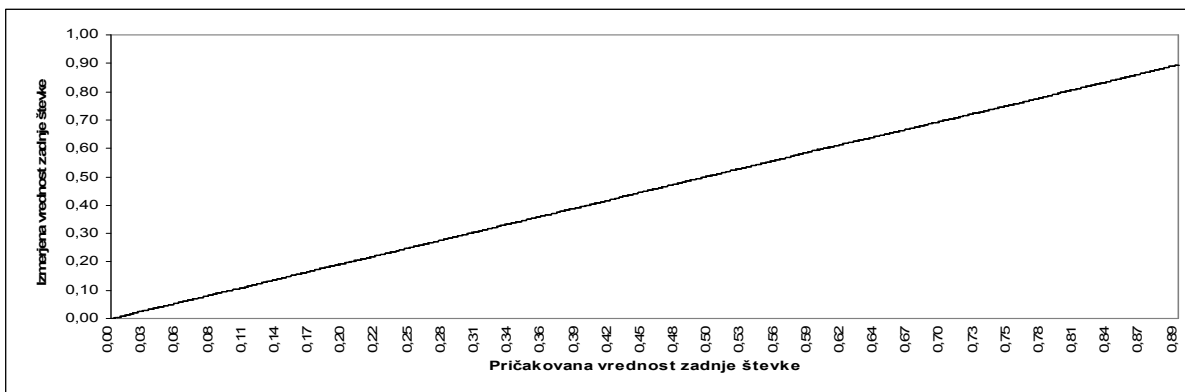
### B.4.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: 0°

<b>Št. meritev</b>	2263
<b>Celotni čas (s)</b>	837,64
<b>Povpr. čas (s)</b>	0,370
<b>Maks. neg. odstop.</b>	0,0005
<b>Maks. poz. odstop.</b>	0,0048
<b>Maks. napaka</b>	0,0048
<b>Min. % ujemanja</b>	90,477



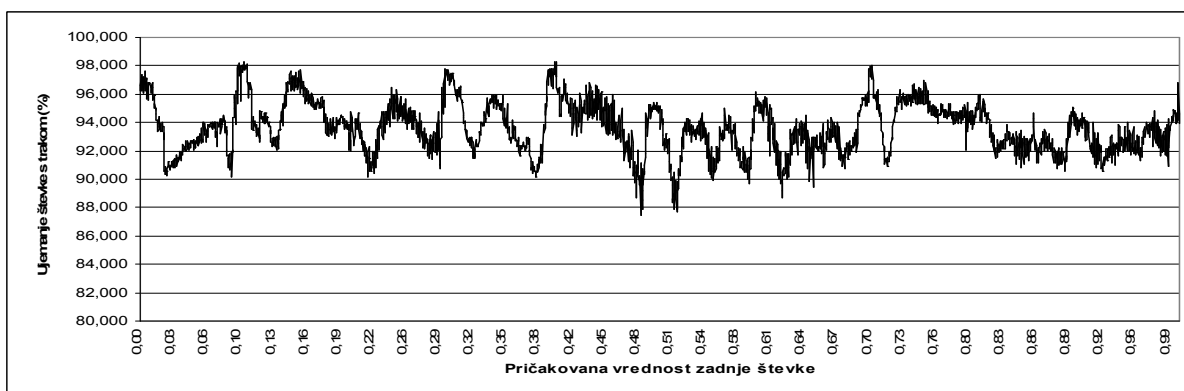
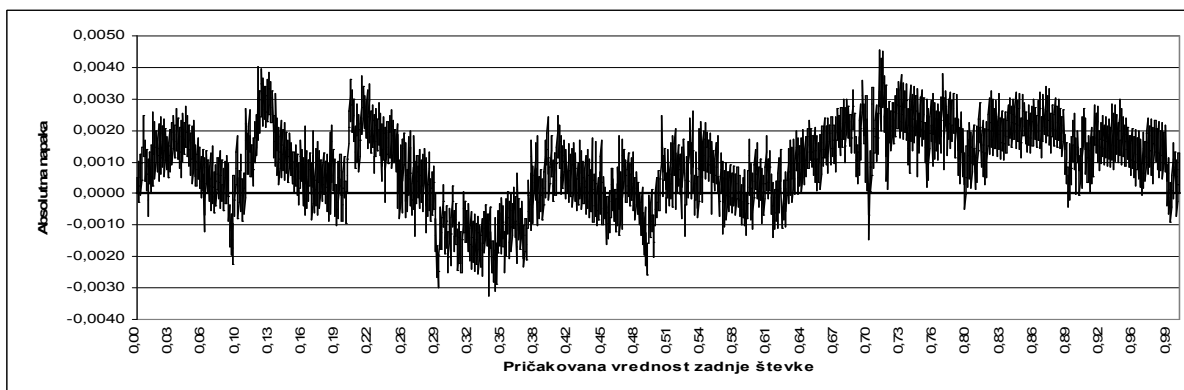
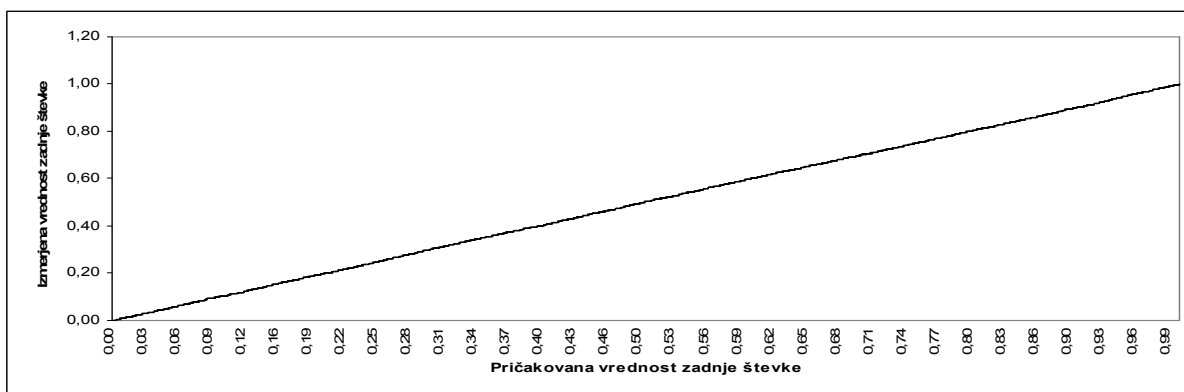
**B.4.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števnik: 0°**

<b>Št.meritev</b>	2651
<b>Celotni čas (s)</b>	846,24
<b>Povpr. čas (s)</b>	0,319
<b>Maks. neg. odstop.</b>	0,0014
<b>Maks. poz. odstop.</b>	0,0039
<b>Maks. napaka</b>	0,0039
<b>Min. % ujemanja</b>	89,697



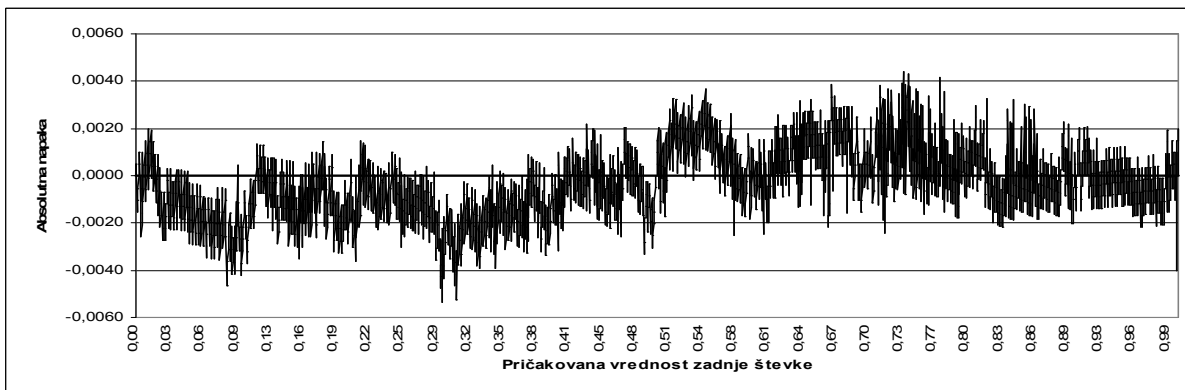
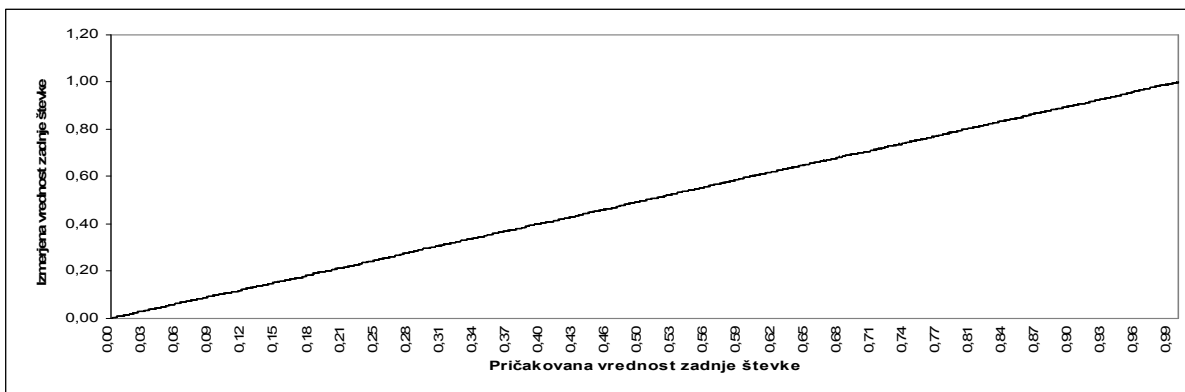
### B.4.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števniaka: 0°

Št. meritev	3073
Celotni čas (s)	847,1
Povpr. čas (s)	0,276
Maks. neg. odstop.	0,0033
Maks. poz. odstop.	0,0046
Maks. napaka	0,0046
Min. % ujemanja	87,492



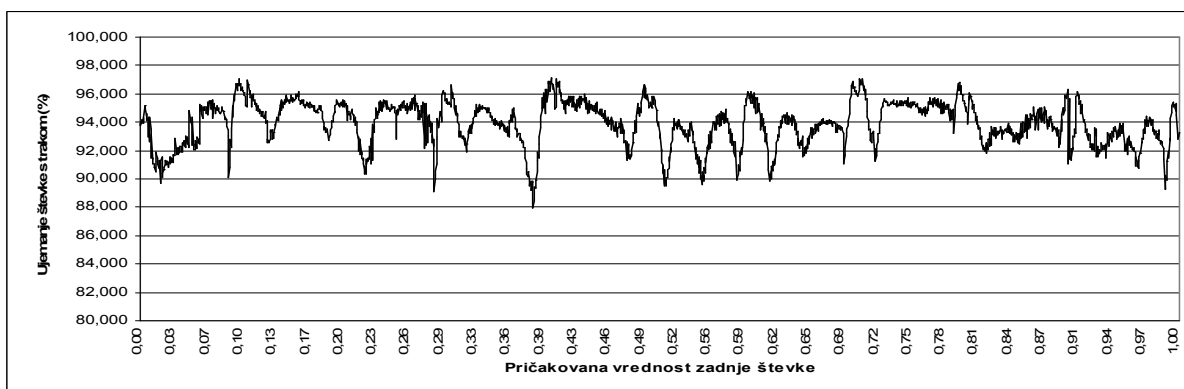
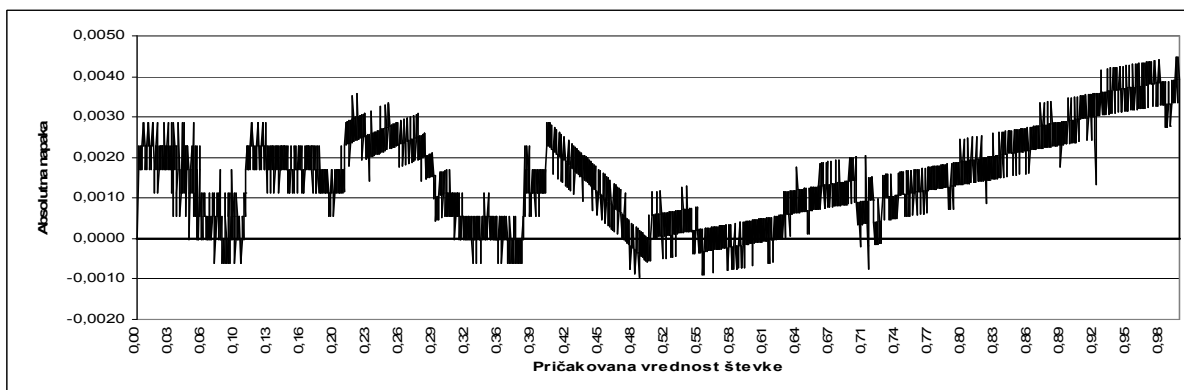
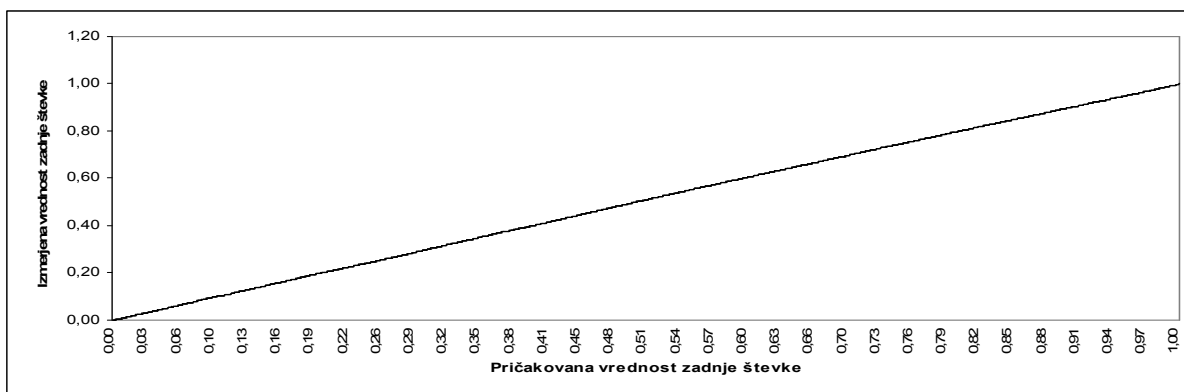
**B.4.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števnik: 0°**

<b>Št.meritev</b>	3324
<b>Celotni čas (s)</b>	847,96
<b>Povpr. čas (s)</b>	0,255
<b>Maks. neg. odstop.</b>	0,0054
<b>Maks. poz. odstop.</b>	0,0044
<b>Maks. napaka</b>	0,0054
<b>Min. % ujemanja</b>	74,638



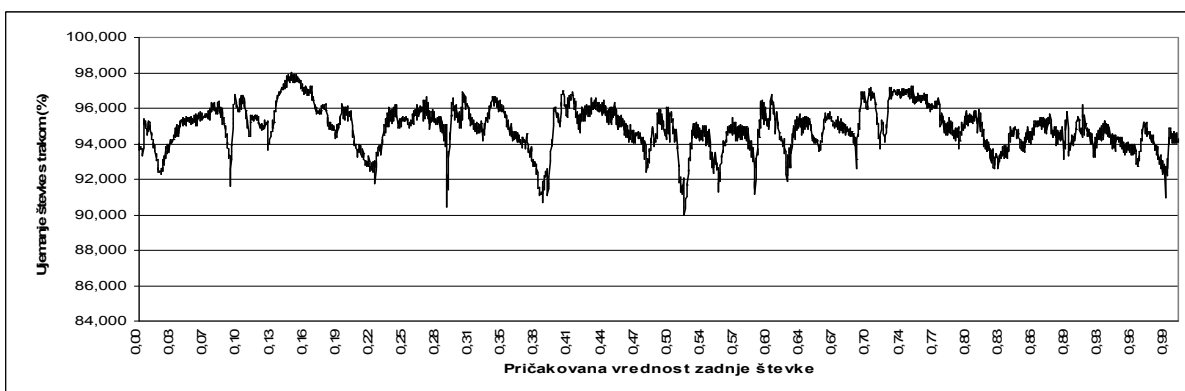
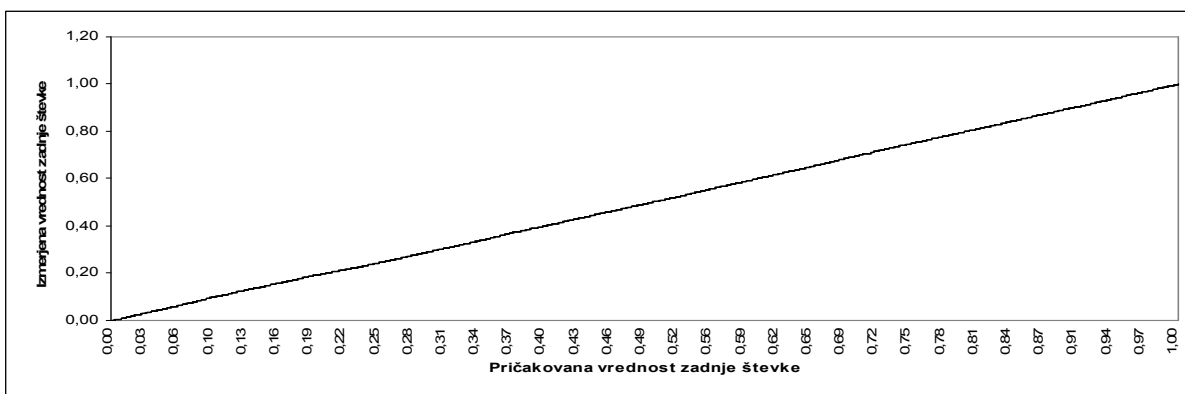
### B.4.8 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: +5°

<b>Št. meritev</b>	2023
<b>Celotni čas (s)</b>	763,25
<b>Povpr. čas (s)</b>	0,377
<b>Maks. neg. odstop.</b>	0,0010
<b>Maks. poz. odstop.</b>	0,0045
<b>Maks. napaka</b>	0,0045
<b>Min. % ujemanja</b>	87,953



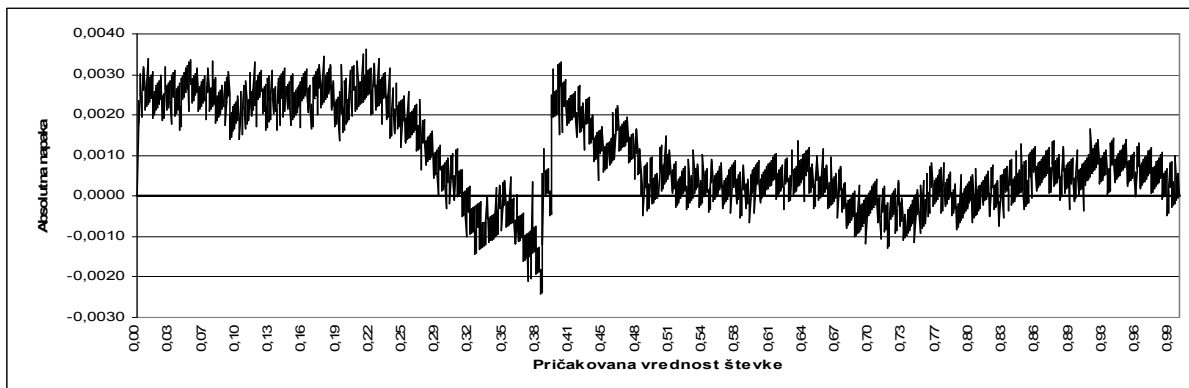
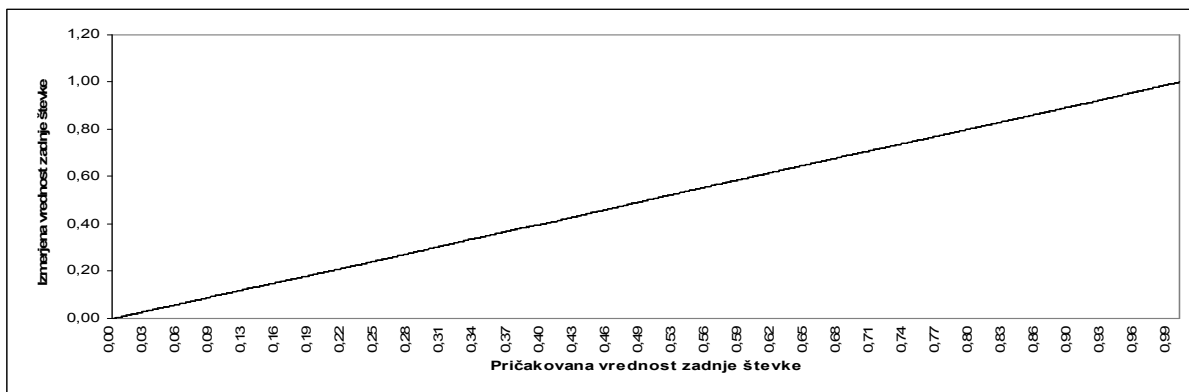
**B.4.9 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: +2'5°**

<b>Št. meritev</b>	2731
<b>Celotni čas (s)</b>	745,19
<b>Povpr. čas (s)</b>	0,273
<b>Maks. neg. odstop.</b>	0,0011
<b>Maks. poz. odstop.</b>	0,0067
<b>Maks. napaka</b>	0,0067
<b>Min. % ujemanja</b>	89,974



**B.4.10 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: -2'5°**

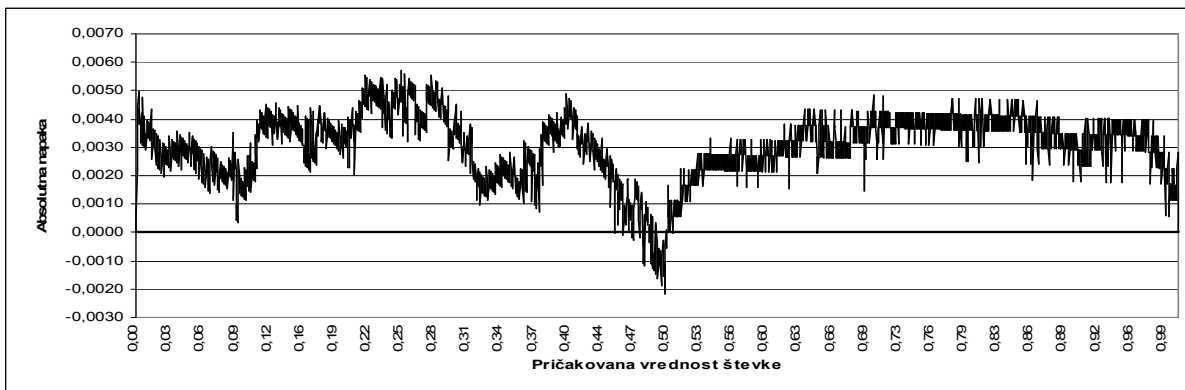
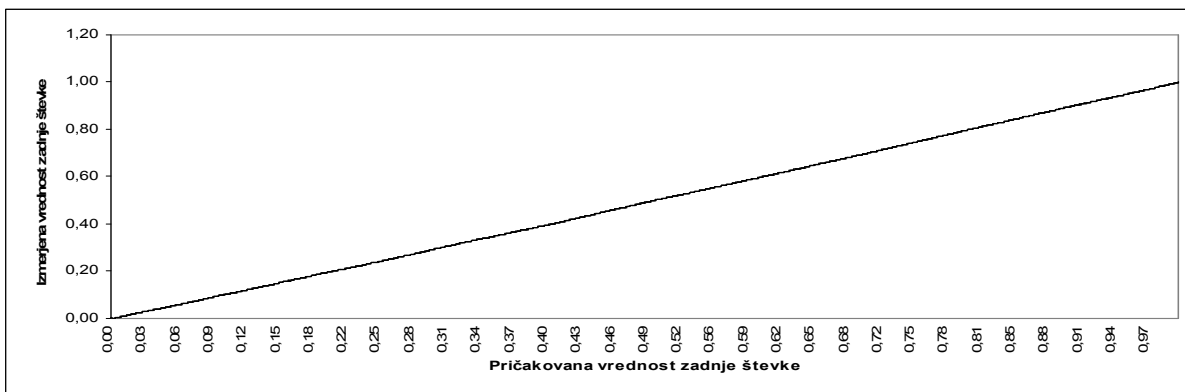
Št.meritev	1908
Celotni čas (s)	719,82
Povpr. čas (s)	0,377
Maks. neg. odstop.	0,0024
Maks. poz. odstop.	0,0036
Maks. napaka	0,0036
Min. % ujemanja	88,318





**B.4.11 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: -5°**

<b>Št.meritev</b>	2044
<b>Celotni čas (s)</b>	776,58
<b>Povpr. čas (s)</b>	0,380
<b>Maks. neg. odstop.</b>	0,0022
<b>Maks. poz. odstop.</b>	0,0057
<b>Maks. napaka</b>	0,0057
<b>Min. % ujemanja</b>	87,582





## Literatura

1. Peter Rogelj: Odčitavanje serijske številke električnih števecv – diplomska naloga, Ljubljana, 1998.
2. Shunji Mori, Ching Y. Suen, Kazuhiko Yamamoto: Historical Review of OCR Research and Development, IEEE, Vol. 80, No. 7, July 1992.
3. V. K. Govindan, A. P. Shivaprasad, Character Recognition – a review, Pattern Recognition, Vol. 23, No. 7, June 1989.
4. Nikola Pavešić: Razpoznavanje vzorcev, Ljubljana, 1992.
5. John C. Russ: The Image Processing Handbook – 2<sup>nd</sup> edition, CRC Press, 1994.
6. Rafael C. Gonzalez, Richard E. Woods: Digital Image Processing, Addison-Wesley, 1992.
7. C. A. Glasbey, G. W. Horgan: Image Analysis for the Biological Sciences, John Wiley & Sons Ltd., 1994.
8. Anton Pozne: Električna merilna tehnika I. del, Ljubljana, 1985.



## **Izjava**

Izjavljam, da sem magistrsko nalogo izdelal samostojno pod strokovnim vodstvom mentorja prof. dr. Stanislava Kovačiča, dipl. ing. Izkazano pomoč drugih sodelavcev sem v celoti navedel v zahvali.

Boštjan Murovec

V Ljubljani, aprila 1999.



## **Strokovni življenjepis avtorja**

Boštjan Murovec se je rodil 22.06.1971 v Šempetru pri Novi Gorici. Po končani osnovni šoli France Bevk v Tolminu se je vpisal na Pedagoško srednjo šolo v Tolminu, kjer si je pridobil poklic učitelj naravoslovno-matematične smeri. Na Fakulteto za elektrotehniko in računalništvo Univerze v Ljubljani se je vpisal leta 1990. Izbral je študijsko smer elektrotehnika-avtomatika-procesna informatika. Diplomiral je julija 1996 in se istega leta vpisal na podiplomski študij Fakultete za elektrotehniko v Ljubljani. Področje njegovega raziskovalnega dela v času magistrskega študija je obsegalo umetni vid ter digitalno obdelavo, analizo in razumevanje slikovnih vzorcev.

## Objavljena dela

1. MUROVEC, Boštjan, KNIEWALD, Matej, KOVAČIČ, Stanislav. Optično odčitavanje električnega števca. *Elektrotehniški vestnik*, 1997, let. 64, št. 2/3, str. 78-86.
2. MUROVEC, Boštjan, KNIEWALD, Matej, KOVAČIČ, Stanislav. Optical reading of electricity meters. V: GENGLER, Manfred (ur.), PRINZ, Michael (ur.), SCHUSTER, Ernst (ur.). *Pattern recognition and medical computer vision 1998 : 22<sup>nd</sup> Workshop of the Austrian Association for Pattern Recognition (ÖAGM / AAPR), Illmitz, May 14<sup>th</sup> – 15<sup>th</sup>, 1998*, (Schriftenreihe der Österreichischen Computer Gesellschaft, Bd. 106). Wien: Österreichische Computer Gesellschaft, cop. 1998, str. 255-263.
3. MUROVEC, Boštjan, KOVAČIČ, Stanislav. Optično odčitavanje električnih števecv. V: Zbornik sedme Elektrotehniške in računalniške konference ERK'98, 24. – 26. september 1998, Portorož, Slovenija / uredil Baldomir Zajc. – Ljubljana: IEEE Region 8, Slovenska sekcija IEEE, 1998. – Zv. B, str. 297-300.



# Kazalo

<b>ZAHVALA</b> .....	<b>1</b>
<b>POVZETEK</b> .....	<b>3</b>
<b>1 UVOD</b> .....	<b>5</b>
<b>1.1 ELEKTRIČNI ŠTEVCI IN PRIKAZ VREDNOSTI</b> .....	<b>6</b>
<b>1.2 ČELNA PLOŠČA ELEKTRIČNEGA ŠTEVCA</b> .....	<b>7</b>
<b>1.3 KONČNI PREIZKUS ELEKTRIČNIH ŠTEVCEV</b> .....	<b>9</b>
<b>1.4 SLABOSTI ROČNEGA ODČITAVANJA</b> .....	<b>10</b>
<b>1.5 PREDNOSTI AVTOMATSKEGA ODČITAVANJA</b> .....	<b>11</b>
<b>1.6 SPECIFIKACIJA ZAHTEV SISTEMA</b> .....	<b>11</b>
<b>2 SPLOŠNO O ZASNOVI SISTEMA</b> .....	<b>13</b>
<b>2.1 MAKROSKOPSKI OPIS SISTEMA</b> .....	<b>14</b>
<b>2.2 STROJNA OPREMA</b> .....	<b>16</b>
2.2.1 Osebni računalnik .....	16
2.2.2 Kamera za zajem slike.....	16
2.2.3 Kartica za zajem slike .....	17
<b>2.3 NEKAJ HITROSTNIH TESTOV</b> .....	<b>17</b>

---

2.4 UPORABA SISTEMA ZA KONČNI PREIZKUS .....	18
2.5 DRUGI NAČINI UPORABE .....	19
<b>3 IZBIRA METOD OBDELAVE SLIK.....</b>	<b>21</b>
3.1 PREDOBDELAVA DIGITALNIH SLIKOVNIH VZORCEV.....	21
3.1.1 Obnavljanje zajete slike.....	22
3.1.2 Izboljšanje kakovosti slike .....	22
3.2 SEGMENTACIJA SLIKE .....	23
3.3 RAZPOZNAVANJE ZNAKOV S POSTOPKI OCR .....	24
<b>4 ODČITAVANJE VREDNOSTI .....</b>	<b>27</b>
4.1 PROBLEMATIKA NATANČNEGA ODČITAVANJA .....	27
4.2 ZAHTEVANA NATANČNOST ODČITAVANJA .....	29
4.3 DOSEGANJE ŽELENE NATANČNOSTI .....	29
<b>5 POSTOPEK ODČITAVANJA VREDNOSTI .....</b>	<b>33</b>
5.1 ZASNOVA SISTEMA .....	33
5.2 OPIS POSTOPKA.....	34
<b>6 ZAČETNA ANALIZA SLIKE .....</b>	<b>39</b>
6.1 TVORJENJE PODVZORČENE SLIKE .....	39
6.2 DOLOČANJE PRAGA BINARIZACIJE .....	41
6.2.1 Postopek določanja praga binarizacije.....	43
<b>7 DETEKCIJA POLOŽAJA ŠTEVNIKA.....</b>	<b>53</b>
7.1 ŠTEVNIK IN NJEGOV POLOŽAJ NA SLIKI .....	53
7.2 SPECIFIKACIJA ZAHTEV ZA DETEKCIJO .....	57

---

<b>7.3 POSTOPEK DETEKTIRANJA ŠTEVNIKA.....</b>	<b>58</b>
<b>7.4 OMEJITVE OPISANEGA POSTOPKA .....</b>	<b>67</b>
<b>8 LOCIRANJE ŠTEVK.....</b>	<b>71</b>
<b>8.1 VERTIKALNI MEJI ŠTEVNIKA.....</b>	<b>71</b>
8.1.1 Izločanje neprimernih temnih področij za iskanje mej .....	72
8.1.2 Iskanje točk na vertikalnih mejah števnik.....	75
<b>8.2 PARAMETRI ŠTEVNIKA.....</b>	<b>79</b>
<b>8.3 DETEKCIJA PRVE IN ZADNJE ŠTEVKE.....</b>	<b>80</b>
<b>8.4 TVORJENJE SEZNAMA OBMOČIJ S ŠTEVKAMI .....</b>	<b>83</b>
<b>8.5 ODSTRANJEVANJE MOTEČIH ROBOV ŠTEVNIKA.....</b>	<b>85</b>
<b>9 PORAVNAVA ŠTEVK .....</b>	<b>89</b>
<b>9.1 PROBLEMATIKA PORAVNAVE ŠTEVK .....</b>	<b>90</b>
<b>9.2 KLASIČEN IN NAŠ PRISTOP K PORAVNAVI ŠTEVK.....</b>	<b>92</b>
9.2.1 Klasičen postopek transformacije slike.....	92
9.2.2 Naš pristop k transformaciji slike števke.....	94
<b>9.3 IMPLEMENTIRANI POSTOPEK PORAVNAVE.....</b>	<b>95</b>
<b>9.4 NAČIN UPORABE PORAVNAVE.....</b>	<b>101</b>
<b>10 IZGRADNJA TRAKU.....</b>	<b>103</b>
<b>10.1 ZAJEM SLIK ZA IZGRADNJO TRAKU.....</b>	<b>103</b>
<b>10.2 POSTOPEK IZGRADNJE TRAKU.....</b>	<b>106</b>
<b>11 DOLOČANJE VREDNOSTI ŠTEVKAM.....</b>	<b>117</b>
<b>11.1 PRIMERJAVA ŠTEVKE S TRAKOM.....</b>	<b>117</b>

<b>12 KONČNI ODČITEK ŠTEVCA.....</b>	<b>121</b>
<b>12.1 TEŽAVA PRI TVORJENJU REZULTATA .....</b>	<b>121</b>
<b>12.2 PREPREČEVANJE PRESKOKOV .....</b>	<b>122</b>
<b>13 MERILNE KARAKTERISTIKE.....</b>	<b>125</b>
<b>13.1 IZVAJANJE MERITEV .....</b>	<b>125</b>
13.1.1 Meritve s števníkom, nastavljanim s koračnim motorjem.....	125
13.1.2 Meritve s fantomsko obremenjenim števcem.....	127
<b>13.2 OCENJEVANJE KARAKTERISTIK SISTEMA.....</b>	<b>128</b>
<b>13.3 REZULTATI MERITEV .....</b>	<b>133</b>
<b>13.4 OCENJENE KARAKTERISTIKE SISTEMA.....</b>	<b>134</b>
13.4.1 Merilne karakteristike sistema pri rotaciji števnika nič stopinj.....	134
13.4.2 Merilne karakteristike sistema pri različnih rotacijah števnika .....	145
<b>14 ZAKLJUČEK .....</b>	<b>153</b>
<b>DODATEK A: SLIKE TRAKOV .....</b>	<b>157</b>
<b>DODATEK B: REZULTATI MERITEV .....</b>	<b>167</b>
<b>B.1 MERITVE PRI VELIKOSTI ŠTEVNIKA 170 SLIKOVNIH ELEMENTOV.....</b>	<b>168</b>
B.1.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnika: 0°.....	168
B.1.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnika: 0°.....	169
B.1.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnika: 0°.....	170
B.1.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnika: 0°.....	171
B.1.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števnika: 0°.....	172
B.1.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števnika: 0°.....	173
B.1.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števnika: 0°.....	174
B.1.8 Velikost števk na traku: 170 slikovnih elementov, rotacija števnika: +5°.....	175
B.1.9 Velikost števk na traku: 170 slikovnih elementov, rotacija števnika: +2'5°.....	176
B.1.10 Velikost števk na traku: 170 slikovnih elementov, rotacija števnika: -2'5°.....	177

---

B.1.11 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: $-5^{\circ}$ .....	178
<b>B.2 MERITVE PRI VELIKOSTI ŠTEVNIKA 150 SLIKOVNIH ELEMENTOV .....</b>	<b>179</b>
B.2.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	179
B.2.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	180
B.2.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	181
B.2.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	182
B.2.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	183
B.2.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	184
B.2.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	185
B.2.8 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $+5^{\circ}$ .....	186
B.2.9 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $+2^{\circ}5'$ .....	187
B.2.10 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $-2^{\circ}5'$ .....	188
B.2.11 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $-5^{\circ}$ .....	189
<b>B.3 MERITVE PRI VELIKOSTI ŠTEVNIKA 130 SLIKOVNIH ELEMENTOV .....</b>	<b>190</b>
B.3.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	190
B.3.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	191
B.3.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	192
B.3.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	193
B.3.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	194
B.3.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	195
B.3.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	196
B.3.8 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $+5^{\circ}$ .....	197
B.3.9 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $+2^{\circ}5'$ .....	198
B.3.10 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $-2^{\circ}5'$ .....	199
B.3.11 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $-5^{\circ}$ .....	200
<b>B.4 MERITVE PRI VELIKOSTI ŠTEVNIKA 110 SLIKOVNIH ELEMENTOV .....</b>	<b>201</b>
B.4.1 Velikost števk na traku: 170 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	201
B.4.2 Velikost števk na traku: 150 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	202
B.4.3 Velikost števk na traku: 130 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	203
B.4.4 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	204
B.4.5 Velikost števk na traku: 90 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	205
B.4.6 Velikost števk na traku: 70 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	206
B.4.7 Velikost števk na traku: 50 slikovnih elementov, rotacija števnik: $0^{\circ}$ .....	207
B.4.8 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: $+5^{\circ}$ .....	208

B.4.9 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: +2'5°.....	209
B.4.10 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: -2'5° .....	210
B.4.11 Velikost števk na traku: 110 slikovnih elementov, rotacija števnik: -5° .....	211
<b>LITERATURA .....</b>	<b>213</b>
<b>IZJAVA .....</b>	<b>215</b>
<b>STROKOVNI ŽIVLJENJEPIS AVTORJA .....</b>	<b>217</b>
<b>OBJAVLJENA DELA .....</b>	<b>218</b>