Korištenje biometrijskih značajki dlana za identifikaciju osoba

Maraš, Manuel

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:195:002848

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-26**

Sveučilište u Rijeci
Fakultet informatike i digitalnih tehnologija

Repository / Repozitorij:

Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository

zir.nsk.hr

uniri DIGITALNA KNJIŽNICA

dabar
DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

Sveučilište u Rijeci – Odjel za informatiku

Informacijski i komunikacijski sustavi

Manuel Maraš

Korištenje biometrijskih značajki dlana za identifikaciju osoba

Diplomski rad

Mentor: Doc. dr. sc. Marina Ivašić-Kos

Rijeka, 12.7.2018

Sadržaj

Sažetak 3
1. Uvod 4
2. Biometrijski sustav 5
2.1. Arhitektura biometrijskog sustava za identifikaciju korisnika 5
2.2. Način korištenja biometrijskog sustava 5
2.3. Evaluacija performansi biometrijskog sustava 6
2.4. Metode uzimanja biometrijskih značajki 7
3. Biometrija dlana 8
3.1. Građa dlana 8
3.2. Značajke dlana 9
3.3. Prednosti otiska dlana nad ostalim biometrijskim tehnologijama 10
4. Biometrijski podatci 11
4.1. Prikupljanje otiska dlana 11
5. Izlučivanje značajki dlana 12
5.1. Registracija krajnjih točaka 14
5.2. Segmentacija dlana po značajkama glavnih točaka 18
5.2.1. Kvadratna segmentacija 18
5.2.2. Upisana kružna segmentacija 20
5.3. Tekstura dlana uz korištenje 2-D Gabor filtera 21
5.4. Značajke linija i njihova reprezentacija 22
5.4.1. Izlučivanje linija i nabora dlana 23
5.4.2. Povezivanje odgovarajućih linija 25
6. Eksperimenti na raznim tipovima slika dlana 28
7. Zaključak 42
8. Literatura 43
9. Reference 43
10. Slike 44
11. Prilog 45

Sažetak

U ovom diplomskom radu opisali smo biometrijski sustav te njegovu arhitekturu, njegove faze rada te načine prikupljanja značajki. Opisali smo dlan te njegove karakteristike. Dali smo uvid u načine prikupljanja biometrijskih podataka. Prikazali smo metode izlučivanja značajki dlana ovisno o tipu slike (offline ili online). Opisali smo nekoliko metoda za analizu linija dlana te njihovo uspoređivanje. Za kraj smo izvršili nekoliko eksperimenata u svrhu usporedbe raznih metoda za detekciju dlana.

Ključne riječi: biometrija, značajke, dlan, metode, izlučivanje, linije dlana, otisak, slika.

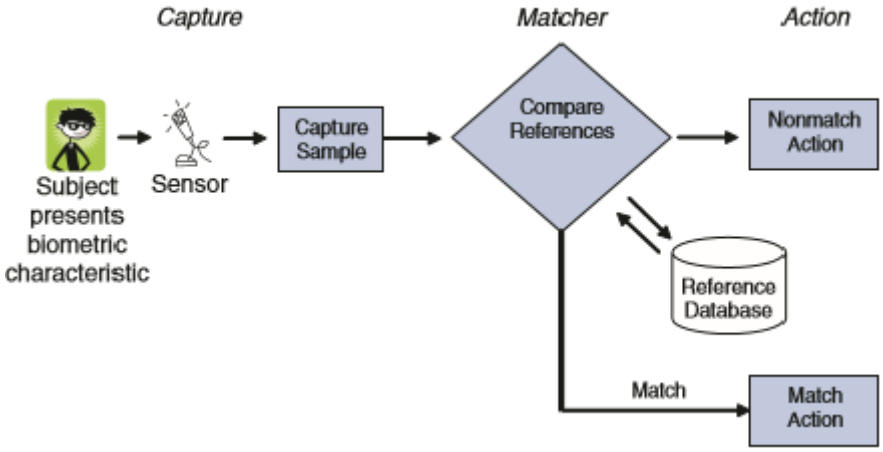
1. Uvod

Biometrija je automatizirani proces prepoznavanja osoba na temelju njihovih ponašajnih i bioloških karakteristika. Biološke karakteristike su otisak prsta, otisak dlana, geometrija ruke, sken oka, termogram lica i slično, a ponašajne karakteristike su govor pojedinca, potpis, način hoda, dinamika pisanja, itd. Služi nam kao instrument pomoću kojega možemo utvrditi dali je osoba koja koristi određeni sustav ta za koju se izdaje, te koje mogućnosti korištenja sustava ta osoba posjeduje.

Biometrijski sustavi se sve češće koriste za prepoznavanje osoba i reguliranje pristupa u prostor, reguliranje dostupnih informacija, usluga, prelazak granica, te drugih prava. Neke od prednosti korištenja takvih sustava su poboljšanje efikasnosti i pogodnosti rutinskih provjera identiteta, smanjenje mogućnosti prijevare, i poboljšanje nacionalne i javne sigurnosti.

Sustavi za prepoznavanje nisu savršeni što znači da se događaju pogreške. Šansa pogreške može biti jako niska no ne i nemoguća. Dizajneri takvih sustava moraju biti spremni i planirati tijek događaja u slučaju pogreške čak i ako su one rijetke. Znanstvena baza biometrije (razumijevanje biometrijskih obilježja dane populacije i način na koji se ljudi služe s biometrijskim sustavima) treba poboljšanje pošto se takvi sustavi sve češće koriste u tehnologijama i sustavima od nacionalne važnosti.

Biometrijski sustavi trebaju biti dizajnirani za njihovu specifičnu svrhu, a ne općenito. Razlog tome je što neki sustavi trebaju prihvaćati više slika, dok odbijaju manje ili suprotno, no to ćemo objasniti u sljedećem poglavlju. Njihova efikasnost ovisi o socijalnom kontekstu isto kao što ovisi o tehnologiji dizajna, sistemskom inženjeringu, operativnom okruženju i načinu testiranja [1]. Na Slici 1. dan je osnovni tok operacija biometrijskog sustava za autentifikaciju korisnika. U „capture“ fazi sustav preuzima i sprema biometrijske značajke, u „matcher“ fazi uspoređuje nove značajke sa značajkama pohranjenim u bazi te na osnovu toga u „action“ fazi odlučuje koja je sljedeća akcija, prihvatiti korisnika ili ne.



Slika 1 : Jednostavni biometrijski sustav [2]

2. Biometrijski sustav

2.1. Arhitektura biometrijskog sustava za identifikaciju korisnika

Biometrijski sustav je zapravo sustav prepoznavanja uzoraka odnosno prepoznavanja korisnika po nekim njegovim ponašajnim ili biološkim karakteristikama. Obično su to biološke karakteristike pojedinca, primjerice otisak prsta, dlana ili 3D geometrija ruke koje dobivamo preko senzora te šaljemo u biometrijski sustav koji analizira uzorak i uspoređuje ga sa uzorcima pohranjenim u bazi i dobivamo rezultate. Rezultat će potvrditi dali je pojedinac poznat sustavu te ima određene dozvole pristupa npr. nekim objektima ili podacima ili ne. Klasični biometrijski sustavi za identifikacije korisnika sadrže 4 koraka :

1) Prikupljanje podataka - Biometrijski podatci (slika ili signal) su dobiveni iz ulaznog uređaja. Kvaliteta dobivenih podataka je jako važna jer o njoj ovise sljedeći koraci.
2) Pred procesiranje slike ili signala – Poboljšanja na slici ili signalu Kao što je smanjenje pozadinske buke, segmentacija, normalizacija pomicanjem i rotacijom.
3) Izlučivanje značajki – Izlučene značajke imaju stabilne i jedinstvene karakteristike s malom razlikom unutar iste klase i velikom razlikom s drugim klasama. Te značajke koristimo da stvorimo glavni predložak ili model određene osobe koji je pohranjen u bazi podataka biometrijskog sustava.
4) Uspoređivanje značajki – Dobivamo rezultat podudarnosti uzorka kojeg identificiramo i glavnog predloška u bazi. Ako je razlika manja od danog praga korisnik je autentificiran

2.2. Način korištenja biometrijskog sustava

Biometrijski sustav općenito ima tri faze rada: upis, identifikacija i verifikacija. Neki sustavi pored upisa imaju još samo ili identifikaciju ili verifikaciju.

Upis – Prije nego korisnik može biti identificiran ili verificiran mora biti upisan u biometrijski sustav. Spremaju se biometrijski podatci korisnika koji su pred-procesirani i izlučene su im značajke. Korisnički predložak se tada sprema u bazu podataka.

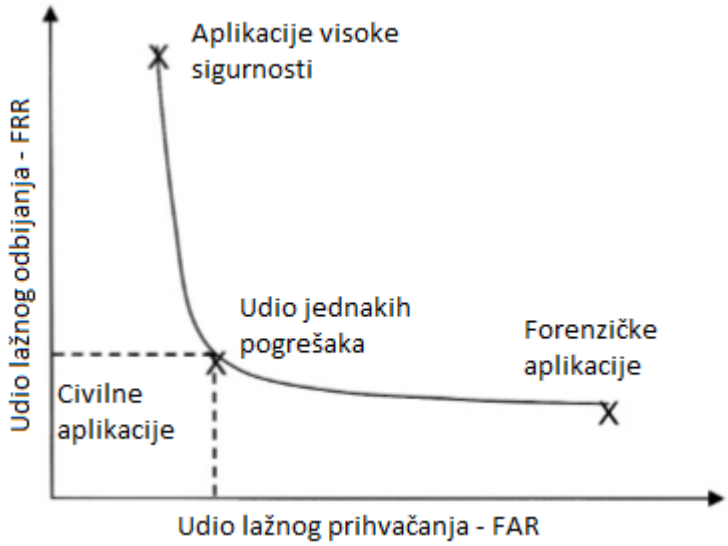
Identifikacija – Kod identifikacije misli se na identifikaciju korisnika samo prema njegovim biometrijskim značajkama, bez znanja o njegovom identitetu. Taj se proces nekada naziva povezivanje jedan na više ili prepoznavanje. Identifikacija koristi korake od 1 do 3 klasičnog biometrijskog sustava da bi napravili predložak korisnika. Sustav tada prelazi na 4. korak i uspoređuje dani predložak sa svim predloščima koje on sadrži. Rezultat je uspješna ili ne uspješna identifikacija. S povećanjem baze podataka predložaka pada i točnost identifikacije.

Verifikacija – Verifikacija zahtjeva da se identitet (ID kartica, pametna kartica, ID broj) pridruži odgovarajućoj osobi. Verifikacijski predložak se povezuje s glavnim predloškom za potvrdu osobnog identiteta. Proces verifikacije još se naziva povezivanje jedan na jedan ili ovjera.

2.3. Evaluacija performansi biometrijskog sustava

Pod evaluacijom performansi obično se misli na udio lažnog prihvaćanja (eng. False Acceptance Rate - FAR), udio lažnog odbijanja (eng. False Rejection Rate - FRR) i udio jednakih pogrešaka (eng. Equal Error Rate - EER). FAR označava situaciju u kojoj ne registrirani korisnik dobije pristup biometrijski zaštićenom sustavu, a FRR situaciju u kojoj registrirani korisnik ne dobije pristup biometrijski zaštićenom sustavu. EER je predstavljen postotkom u kojem su FAR i FRR jednaki, odnosno to je mjera koja se definira kao prag do kojeg su FAR i FRR isti. Što je taj prag manji sustav ima manje grešaka što znači veću točnost. Idealni sustav nema lažnog prihvaćanja i lažnog odbijanja no trenutno takav sustav ne postoji.

Slika 2. prikazuje hipotetsku ROC (eng. Reciver Operating Characteristics) krivulju s obzirom na FAR i FRR mjere. Istaknute su neke karakteristične točke koje se odnose na očekivanu pouzdanost biometrijskog sustava s obzirom na različite aplikacijama i različitu razinu sigurnosti.



Slika 2: Hipotetska ROC krivulja

Sigurnosni prag ovisi o kontekstu u kojem se biometrijski sustav koristi. Primjerice ako je to neki visoko čuvani objekt poželjnije je što manji FAR nego FRR jer je manji propust ako se netko tko ima dozvoljen ulaz u neki objekt dodatno provjeri nego da se propuste i oni koji zapravo nemaju dozvolu pristupa, dok je za forenzičke aplikacije poželjnije imati niži FRR i

provjeriti što je moguće više ljudi. Idealni rad biometrijskog sustava je kada su i FRR i FAR niski. Formule za izračun performansi su sljedeće:

$$FRR = \frac{FR}{GA + FR} \tag{1}$$

$$FAR = \frac{FA}{GR + FA} \tag{2}$$

FR - False Rejection (lažno odbijanje), FA – False Acceptance (lažno prihvačanje), GR – Genuine Rejection (ispravno odbijanje), GA – Genuine Acceptance (ispravno prihvačanje).

2.4. Metode uzimanja biometrijskih značajki

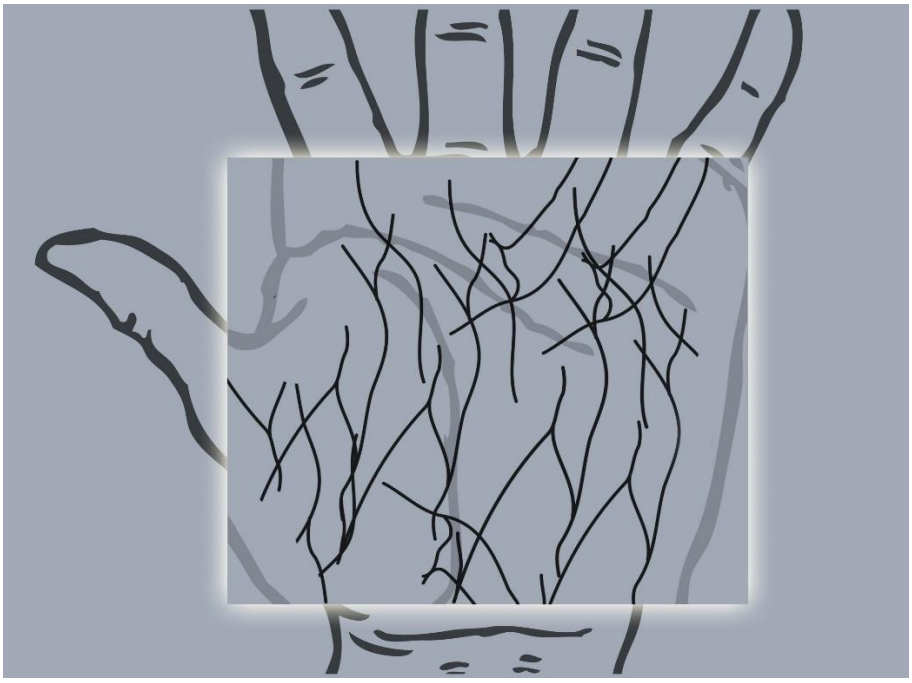
Postoji više metoda uzimanja biometrijskih značajki te svaka ima određene prednosti i mane. U ovom radu bazirat ćemo se na biometriju dlana no navesti ćemo i neke druge metode da bi bolje razumjeli prednosti koje nam biometrija dlana pruža. Jedna od tih metoda je prepoznavanje šarenice oka. Šarenica je biometrijska značajka koja osigurava veliku sigurnost i točnost jer je jedinstvena za svaku osobu, no takvi sustavi su skupi i korisnici osjećaju nelagodu kod skeniranja oka.

S druge strane, npr. 3D geometrija dlana ima malu količinu značajki kao što su dužina, širina, debljina, površina dlana i prsta i lako ju je prikupiti no nije identifikacijska značajka jer jedinstvo šake nije zajamčeno. Stoga se ova značajka slabo koriste u aplikacijama koje imaju pristup jedan na više, odnosno dobre su primjerice za identifikaciju zaposlenika neke firme gdje broj zaposlenika nije beskrajno velik, no korištenje za ulazak u avion nije moguć jer šaka pojedinca nije dovoljno jedinstvena da se osoba njome može identificirati. Otisak prsta je najčešća biometrijska značajka, no uz to može se zasebno koristiti za identifikaciju osoba. Prednosti su joj malena veličina, jednostavno prikupljanje i velika točnost što ju čini najpopularnijom biometrijskom tehnologijom. No ni ta metoda nije savršena. Moguća su oštećenja prstiju zbog radnog mjesta pogotovo kod ljudi koji u tvornicama rade sa strojevima i podložni su ozljedama prstiju ili kod fizičkih radnika, te zbog starenja kada se javlja zadebljala koža i nabori na prstima što otežava pravilnu usporedbu korisnika.

Jedna od najjeftinijih metoda za identifikaciju je skeniranje lica, zahtjeva samo postavljenu kameru na lokaciji ulaska u neki objekt ili slično [3]. Područja lica koja se skeniraju su očna šupljina, jagodice i područje oko usta. Skeniranje lica je najpogodnije u područjima gdje se skeniranje i nadzor vrši bez velike interakcije s prolaznicima. Neke države imaju postavljene kamere te automatski skeniraju sve prolaznike i uspoređuju ih s slikama traženih kriminalaca.

Nova metoda biometrijske identifikacije, nastala u Fujitsu laboratorijima 2002 godine [4], vrši prepoznavanje korisnika preko vena u dlanu, Slika 3. Osim veličine vene su različite od čovjeka do čovjeka po strukturi te se ne mijenjaju starenjem. Slike vena se dobivaju pomoću infracrvenog svjetla. Testiranjem 1400 slika dlana na 700 osoba sustav je imao pogrešku lažnog

odbijanja od 1% i pogrešku lažnog prihvaćanja od 0.5% i udio jednakih pogrešaka od 0.8% [5] problem je samo što je takav sustav vrlo skup.



Slika 3: Primjer obrasca vena

3. Biometrija dlana

Sve tehnologije imaju svoje prednosti i mane te ni jedna tehnologija nije savršena za svaku situaciju. U nastavku će detaljnije biti opisane značajke koje se izlučuju iz slike dlana.

3.1. Građa dlana

Dlanom se smatra unutarnji dio šake od zgloba šake do vršaka prstiju. Dlanovi imaju površinu puno veću nego vršci prstiju, ali je pokriven s istim tipom kože.

Kada kažemo građa dlana mislimo na tri stvari :

1) **Glavne linije** – Svaka osoba ima 4 glavne linije koje se formiraju na isti način, no bez obzira na to razlikuju se od osobe do osobe. Jedan od razloga za to je način na koji osoba hvata/uzima stvari. Još jedan razlog je različita veličina prstiju i debljina dlana. Također svaka šaka se savija drugačije što također oblikuje te 4 linije.

2) **Nabori** – Nabori nastaju po cijelom tijelu te pružaju koži određenu rastezljivost. Kako koža gubi na elastičnosti tako nabori postaju sve uočljiviji. Neki nabori su urođeni dok primjerice nabori na dlanu većinom nastaju zbog načina hvatanja stvari.

3) **Grebeni** – Grebeni su trajna zadebljanja na epidermu (vanjskom ćelijskom dijelu kože). Oni se uzdižu iznad normalne razine kože i nalaze se samo na dlanovima i stopalima. Njihova rasprostranjenost ovisi o područjima koja su pod naporom od nošenja velike težine ili hvatanja.

3.2. Značajke dlana

Značajke dlana su glavne linije, nabori, grebeni, singularne točke i detaljne točke. Mnoge karakteristike dlana možemo koristiti za identifikaciju osoba. Identifikacija pomoću dlana je metoda za identifikaciju osobe pomoću jedinstvenih značajki dlana koja mogu, ali i ne moraju biti vidljiva golom oku. To se postiže pomoću algoritama koji mogu razlučiti razlike između dvije osobe prema tim značajkama.

Značajke dlana možemo prikupiti uzimanjem otisaka dlana. Otisak je utisak na nekoj površini pritiskom. Otisak dlana čini obrazac kože na dlanu sastavljen od svojih jedinstvenih karakteristika kao sto su linije, točke i tekstura. Otiske dlana možemo naći na površinama raznih objekata zbog znojenja. Što je dlan suši to su otisci manje vidljivi.

Otisak dlana možemo prikazati sa i bez tinte. Četiri značajke možemo preuzeti iz oba načina prikaza dok dvije možemo samo s otiska korištenjem tinte. Te značajke su:

1) Geometrijske značajke – Ovisno o obliku dlana možemo izlučiti neke značajke kao sto su dužina, širina i debljina.

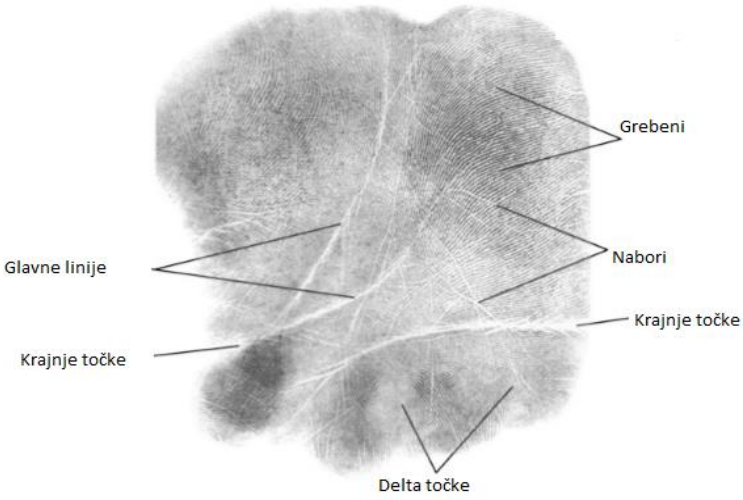
2) Značajke glavnih linija – Lokacija i forma glavnih linija dlana su važne značajke kod identifikacije zato što se jako malo mijenjaju tijekom godina.

3) Značajke nabora – Dlan ima mnoštvo nabora koji se razlikuju od glavnih linija po tome što su tanji i neregularni. Razlikujemo hrapave i glatke nabore tako da imamo više detalja za usporedbu.

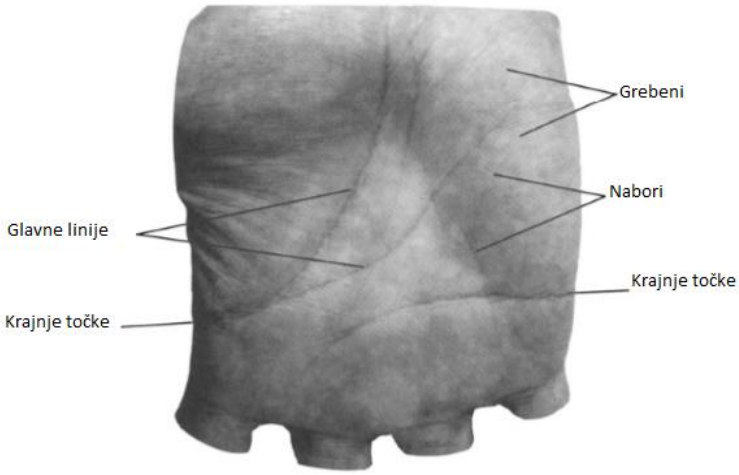
4) Krajnje točke – Dvije krajnje točke dobivamo pomoću glavnih linija (Slika 4.). Oni se nalaze na krajevima obje strane dlana i pružaju dobar način za registraciju dlana. Veličina dlana može biti izmjerena pomoću euklidske dužine između te dvije točke.

5) Značajke delta točaka – Delta točke su središnje točke koje se nalaze u području korijena prsta. One nam pružaju jedinstvenu mjeru za identifikaciju dlana.

6) Detaljne točke – Kako se dlan sastoji od grebena to nam dozvoljava da koristimo detaljne točke kao značajke u usporedbi dlanova.



Slika 4: Otisak dlana korištenjem tinte



Slika 5: Otisak dlana bez korištenja tinte

3.3. Prednosti otiska dlana nad ostalim biometrijskim tehnologijama

Osim mnoštva značajki otisci dlana imaju još prednosti nad ostalim biometrijskim tehnologijama šake.[6]

1) U odnosu na otisak prsta, dlan pruža veću površinu tako da se više značajki može izvući.
2) Pojedinac će puno prije oštetiti vrh prsta nego cijeli dlan. Također linije dlana se ne mijenjaju mnogo tijekom života.
3) Male količine prljavštine i masti mogu se naći na prstu pojedinca koji koristi biometrijski sustav te negativno utjecati na verifikaciju prsta. Takvi problemi

imaju manje značenje kod skeniranja dlana jer se koriste slike od 75-150 DPI dok se kod skeniranja vršaka prstiju koriste slike od 500 DPI-a.

4) U odnosu na tehnologiju 3D geometrije šake, dlan ima puno više jedinstvenih značajki koje možemo koristiti kod identifikacije te se stoga mogu očekivati i bolji rezultati.
5) Kako se za izlučivanje značajki dlana koristi senzor s manjom rezolucijom u odnosu na značajke prsta tako se štedi na vremenu jer su računanja puno brža kod predprocesiranja i kod izlučivanja značajki.

4. Biometrijski podatci

Svaki biometrijski sustav počinje sa prikupljanjem podataka. Biometrijski podatci se prikupljaju ili online ili offline metodama. Prije razvoja online metoda svi su se podatci prikupljali offline. Offline podatci se prikupljaju tako da se dlan uroni u tintu te se takav prisloni na papir, nakon toga se papir skenira i prebacuje u digitalni oblik. Online prikupljanje podataka je direktno skeniranje dlana putem kamere ili uređaja za skeniranje.

4.1. Prikupljanje otiska dlana

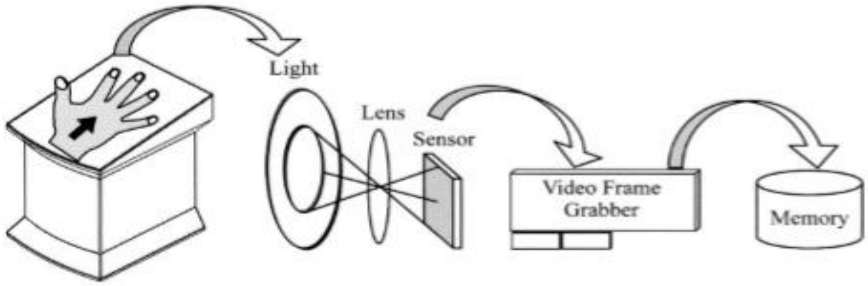
Otiske dlana možemo prikupljati offline ili online. Offline prikupljanje nam znači da otisak nije direktno unesen u računalo već je prvo otisnut pomoću tinte na papir te zatim digitaliziran pomoću skenera u računalo. Online prikupljanje otisaka dlana je direktan način prikupljanja podataka jer eliminira potrebu za tintom i papirom već direktno prenosi podatke u računalo. Može se vršiti sa skenerom za dlan ili pomoću video kamere.

Kod offline metode kvaliteta je upitna zbog razloga što tintu koju stavljamo na dlan ne možemo nanijeti u optimalnim količinama. I previše i premalo tinte daje ne zadovoljavajuće rezultate. Ova se metoda također ne može koristiti kod sustava koji trebaju identifikaciju u sadašnjem vremenu zbog toga što otisak tinte ne možemo instantno digitalizirati.

Prvi online skener nastao je 1999-te godine u Hong Kongu. To je prvi pokušaj prikupljanja otisaka dlana direktno u računalo. Uređaj je bio plastična kutija koja je sadržavala izvor svjetlosti, ogledalo, staklenu podlogu i CCD kameru. Nakon više testiranja pokazalo se da slika nastala kao refleksija sa ogledala nije dobra kao i direktno slikani dlan. Staklena podloga na kojoj bi stajao dlan iskrivila je površinu dlana te bi ekstrakcija značajki dlana bila loše kvalitete. Bio je potreban bolji uređaj za online prikupljanje otisaka dlana.

Online prikupljanje otisaka dlana zahtjeva skener koji može brzo prikupiti kvalitetne slike dlana. Također korisničko sučelje mora biti napravljeno da je korisniku

pojednostavljena interakcija sa takvim sustavom. Na slici 6. dan je uvid kako bi takav sustav trebao izgledati.



Slika 6: Struktura sustava za online prikupljanje otisaka dlana

Neke specifične specifikacije za korisničko sučelje i optički sustav su:

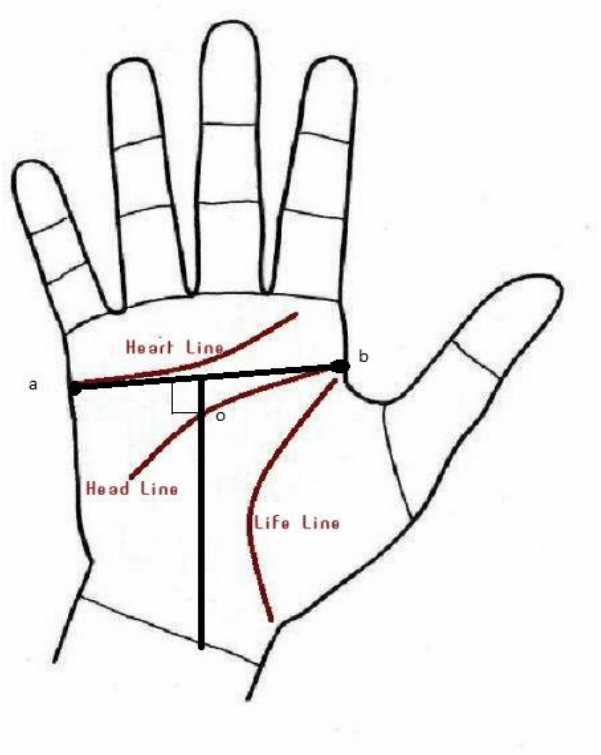
1. Korisničko sučelje:
    - Lagan i intuitivan za korištenje – Korisnik se mora osjećati ugodno dok koristi sustav jer je korisničko prihvaćanje sustava bitno za aplikacije koje su posvećene korisnicima.
    - Veličina – Ovisno o namjeni, ali treba se težiti što manjem sustavu.
2. Optički sustav:
    - Kvaliteta slike – Slike moraju biti dobre kvalitete tako da je moguće izvršavanje sljedećih koraka u procesiranju slika.
    - Procesiranje u sadašnjem vremenu – Odaziv sustava treba biti dovoljno brz da se može koristiti u praktičnim aplikacijama.

Sustav je strukturiran na način da korisnik stavlja dlan na skener nakon čega optičke komponente zajedno uzimaju značajke dlana. Tada je analogni signal transformiran u digitalni pomoću A/D konvertera i spremljen u memoriju sustava.

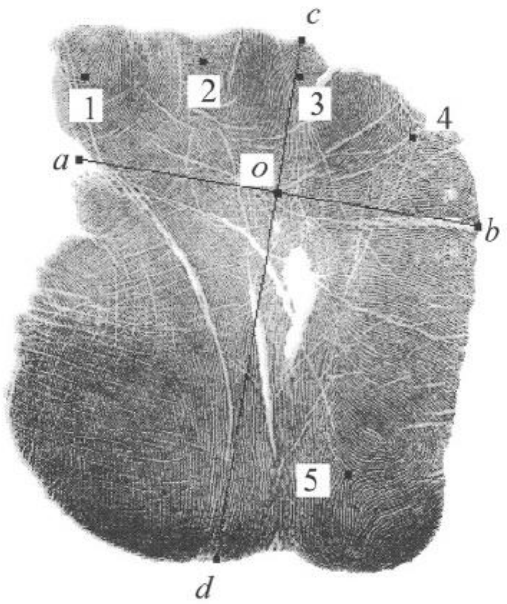
Korisničko sučelje je specijalno dizajnirana ploha za držanje dlana. Služi kao ulazni kanal za korisnika i sustav te za izlučivanje značajki dlana. Optičke komponente (Izvor svjetlosti, leče i CCD (charge-coupled device) senzori i A/D konverter su glavni dio sustava za izlučivanje značajki dlana. Svjetlost iz objekta (dlana) prolazi kroz otvor te je konvergiran pomoću leća u sliku. Optički sustav je proizveden u kontroliranom okruženju u svrhu smanjivanja vanjskog svjetla pošto ono može uzrokovati smanjenje kvalitete slike. CCD senzori konvertiraju signale fotona u električni signal. Hvatač okvira procesira analogni signal iz CCD senzora te ga se konvertira u digitalni signal. Tada se slika dlana sprema u memoriju sustava te je spremna za korištenje.

5. Izlučivanje značajki dlana

Značajke dlana mogu sadržavati neke distorzije jer značajke dlana mogu biti stečene u različitim vremenskim uvjetima (temperatura, vlažnost, svjetlost). Osim smanjenja vanjskih utjecaja i glađenja slika dlana, predprocesiranje slika dlana uključuje ispravljanje distorzije i stavljanje svih slika u isti koordinatni sustav tako da se isto područje svakog dlana uspoređuje u sljedećim usporedbama.



Slika 7: Definiranje dijelova dlana



Slika 8: Geometrijske značajke dlana

Uglavnom koristimo 3 glavne linije dlana (linija glave, života i srca). One su jedinstvene, ne mijenjaju se i ne mogu se krivotvoriti. Također imamo točke a i b koje su krajnje točke linije

glave i linije srca. Kako se te linije ne mijenjaju točke a i b i točka o koja je njihovo središte su stabilne i ne promjenjive. Neke bitne značajke možemo dobiti iz tih saznanja:

- Lokacija krajnjih točaka i njihove srednje točke su invarijantne po rotaciji.
- Dvodimenzionalni pravokutni koordinatni sustav možemo postaviti jedinstveno tako da je točka o središnja točka te da x-koordinata prolazi kroz krajnje točke.
- Veličina dlana može se jedinstveno odrediti pomoću Euklidske udaljenosti između 2 krajnje točke i pomoću simetrale na tu dužinu (Slika 8 c-d).
- Dlan možemo podijeliti na 3 dijela. Dio iznad x-koordinate dio lijevo od simetrale na tu dužinu i dio desno od simetrale.
- Neke značajke možemo dobiti iz delta točaka (Slika 8 točke od 1 do 5) koje su definirane kao središte dijela koje liči na grčko slovo delta odnosno frakcije koje dijele dva tipa linija. Dokazano je da uvijek postoje neki delta dijelovi kod korijena prstiju.

5.1. Registracija krajnjih točaka

Cilj određivanja krajnjih točaka je registracija istih u svrhu prikaza značajki i usporedbe. Kao glavni proces u identifikaciji dlana poželjno je da bude što jednostavniji i što efikasniji. Glavna ideja je određivanje kraja glavnih linija odnosno registracija krajnjih točaka. U odnosu na njihovu stabilnost, glavne linije i krajnje točke određuju se pomoću algoritma usmjerene projekcije.

Algoritam projekcije smjera: [7]

Ako je F crno bijela slika dužine M i širine N sa f(i,j) razinom sive boje piksela (i,j)(i = 0,1, … , M-1, j = 0,1, … , N-1). Uzevši u obzir da je projektni kut α mjeren u smjeru kazaljke sata u odnosu na i-os i da su pikseli (i`.j`) na slici F, možemo postaviti x-y pravokutni koordinatni sustav na kojem su točke (i`,j`) polazište i orijentacija x-osi je smjer projektivne linije (Slika 9). U x-y koordinatnom sustavu imamo pod sliku F_1 veličine (2m+1) x (2n+1) piksela, f(x,y) je razina sive boje piksela (x,y)(x = -m,-m+1, …, 0, m-1, m; y = -n,-n+1, …, 0, n-1, n). Stoga imamo vezu između para koordinatnih sustava:

$$i = i' + \cos(\alpha + \beta)\sqrt{x^2 + y^2} \tag{3}$$

$$j = j' + \sin(\alpha + \beta)\sqrt{x^2 + y^2} \tag{4}$$

gdje je $\beta = \tan^{-1}(y/x)$.

Na slici F_1 projekcija smjera ove slike je

$$p(y) = \sum_{i=-m}^{m} f(x, y). \tag{5}$$

Izglađeni skup q(y) računa se kao

$$q(y) = \frac{1}{2w+1} \sum_{k=-w}^{w} p(y+k). \tag{6}$$

Tada definiramo piksel $(0,y_0)$ kao

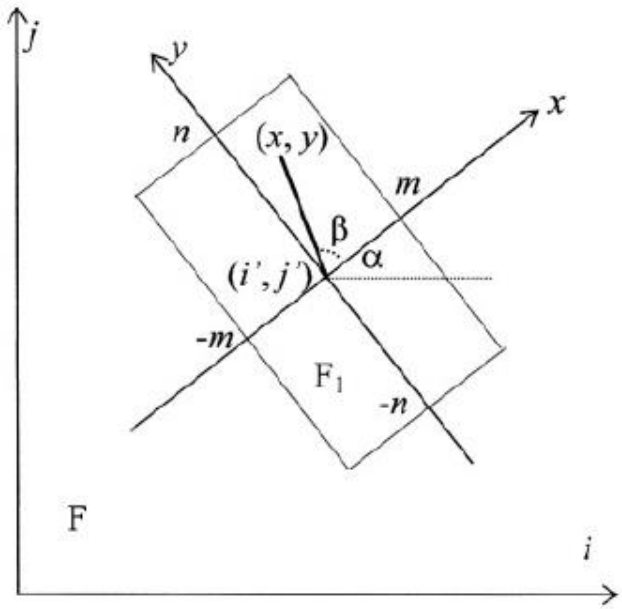
$$y_0 = \left\{ k \middle| q(k) = \max_y q(y) \right\}, \tag{7}$$

Te na kraju dolazimo do piksela (i_0,y_0) na slici F

$$i_0 = i' + y_0 \cos\left(\alpha + \frac{\pi}{2}\right) \tag{8}$$

$$j_0 = j' + y_0 \sin\left(\alpha + \frac{\pi}{2}\right). \tag{9}$$

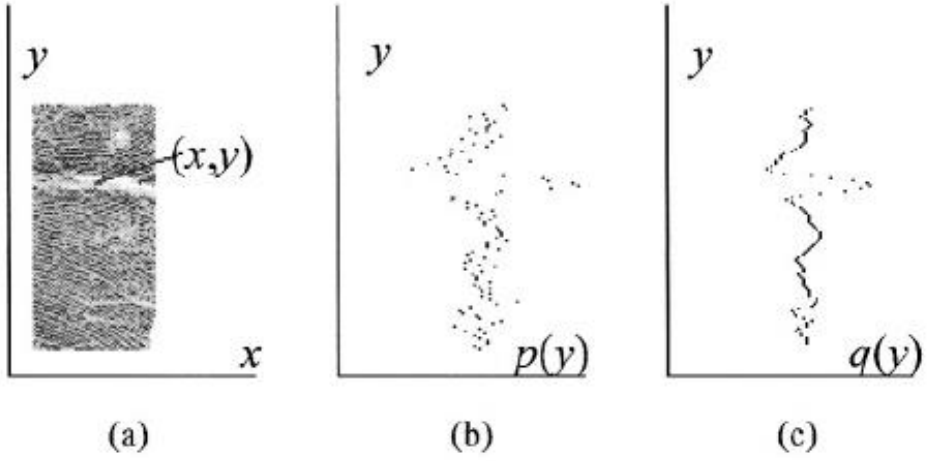
Algoritam je predstavljen u četiri forme različite po orijentaciji: horizontalna projekcija ($\alpha = 0^0$), projekcija s 45 stupnjeva ($\alpha = 45^0$), vertikalna projekcija ($\alpha = 90^0$) i projekcija s 135 stupnjeva ($\alpha = 135^0$).



Slika 9: Zavisnost dva koordinatna sustava u projekciji smjera

Svojstva glavnih linija: [8]

- Svaka glavna linija susreće se sa jednom stranom dlana približno pod pravim kutom u svojoj početnoj točci
- Životna linija nalazi se u unutarnjem dijelu dlana, odnosno postepeno se pomiče prema unutrašnjosti
- U većini slučajeva linija života i glave potječu iz iste točke
- Krajnje točke a i b bliže su prstima nego zglobu



Slika 10: a) originalna slika, b) projekcija smjera uz glavnu liniju,

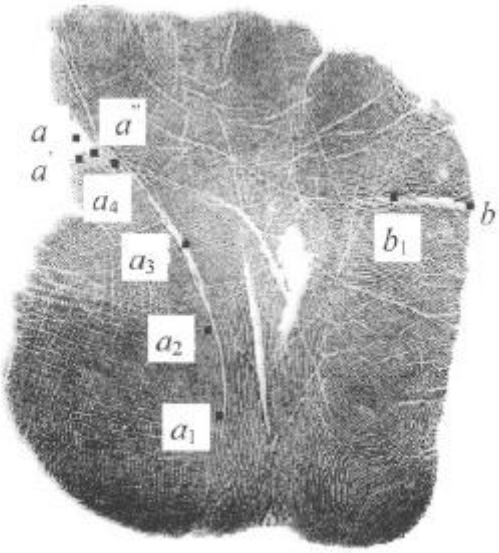
c) projekcija smjera nakon srednjeg(median)filtera

Na temelju projektivne sličnosti (Slika 10) jasno je da se piksel (x,y) izračunat pomoću osnovnog algoritma nalazi na glavnoj liniji ako orijentacija projekcije smjera prati glavnu liniju i ako je parametar w u (4) formuli jednak polovici širine glavne linije. Međutim pikseli na drugim linijama se ne mogu odrediti na ovaj način. Razlog tome je taj što bi parametar w bio manji kod drugih linija koje nisu paralelne sa smjerom projekcije ili su kraće nego glavna linija.

Određivanje krajnjih točaka:

Kod određivanja krajnjih točaka imamo dva procesa. Prvi proces je detektiranje glavne linije, a drugi praćenje glavne linije. Koristimo slike desnog dlana s prstima okrenutim prema gore.

Detektiranje glavne linije:



Slika 11: Detekcija krajnjih točaka

Cilj ovog koraka je dobivanje točaka na liniji srca ili liniji života. Liniju srca je lakše odrediti nego liniju života zato što je to jedina linija koja se nalazi na vanjskom dijelu dlana. Nakon što odredimo rubove gornjeg i vanjskog dijela dlana piksel koji ima prikladan istup od oba ruba uzima se kao početna točka. Pomoću algoritma horizontalne projekcije dobivamo točku b_1 koja pripada liniji srca. Međutim za detekciju životne linije ne možemo koristiti algoritam horizontalne projekcije u unutarnjem dijelu dlana. Razlog tome je što se u nekim slučajevima gdje linija glave i linija života ističu iz dlana na različitim mjestima dobiva linija glavne umjesto linije života. Stoga uzimamo još jednu pod sliku blizu zgloba pomoću algoritma vertikalne projekcije i dobivamo točku a_1 na liniji života.

Praćenje glavne linije: [9]

Glavni cilj praćenja glavnih linija je određivanje njihovih krajnjih točaka. Prema osobinama linije srca algoritam horizontalne projekcije može lako locirati krajnju točku b tamo gdje je piksel na vanjskom rubu s jednak pikselu b_1. Međutim detekcija krajnje točke a prati drugačije korake. Životna linija je krivulja od piksela a_1 do točke a. Način detekcije točke a je sljedeći:

1. Dobivamo točku a` tako da povučemo pravac kroz točke b i $b_1$ te lokacija gdje pravac siječe unutarnji rub dlana je lokacija točke a`.
2. Pikseli a_2 i a_3 koji dijele vertikalnu udaljenost između piksela a_1 i točke a` dobivaju se pomoću algoritma vertikalne projekcije.
3. Horizontalna udaljenost od piksela a_3 do unutarnjeg ruba dlana je izmjerena i nova točka a_4 može biti označena tako da je udaljena od točke a_3 horizontalno i vertikalno za pola udaljenosti koju smo izmjerili. Pod slika oko ove točke dobivena je pomoću algoritma projekcije smjera sa 135 stupnjeva.

4. Točka a`` je dobivena tamo gdje linija koja ide iz piksela a_4 pod kutom od 22.5 stupnjeva siječe unutarnji rub dlana.
5. Krajnja točka a dobivena je u pod slici sa točkom a`` pomoću algoritma horizontalne projekcije.

U odnosu na točke a i b jednostavno je odrediti središnju točku o. Za određivanje krajnjih točaka lijevog dlana koriste se isti koraci, jedino se kod trećeg koraka koristi kut od 45 stupnjeva umjesto kuta od 135 stupnjeva.

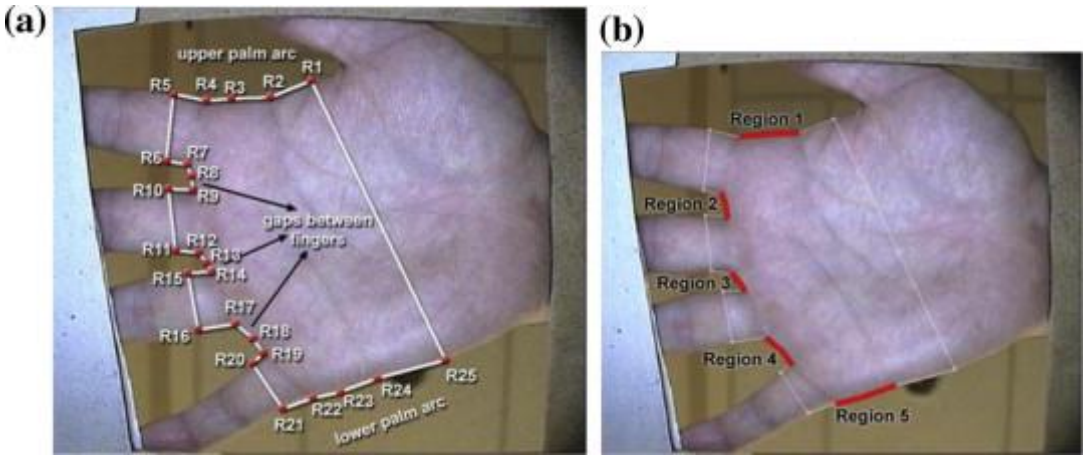
5.2. Segmentacija dlana po značajkama glavnih točaka

Segmentacija dlana je proces ispravljanja distorzija i stavljanje svih slika dlana u isti koordinatni sustav tako da uvijek dolazimo do očekivanog dijela dlana kod procesiranja slike. Metoda koju smo naveli za prikupljanje značajki kod slika dlana dobivenih pomoću tinte i papira nije pogodna za slike dlana dobivene kamerom i sličnim uređajima već se koriste metode za ekstrakciju regije interesa (ROI-region of interest) zvane centralni dio podslike dlana.

5.2.1. Kvadratna segmentacija

Glavna ideja kod kvadratne segmentacije je da se prvo definiraju tri glavne točke na dlanu. Nakon toga postavlja se ortogonalni koordinatni sustav koristeći te tri točke. Tijekom segmentacije kvadrat fiksne veličine je ekstrahiran iz predefinirane pozicije na koordinatnom sustavu. Veličina i pozicija kvadrata određuje se pomoću statističkih podataka uzetih s više dlanova. Najvažnije pravilo je da se do kvadrata kojeg ekstrahiramo može doći na svim slikama dlana koje koristimo.

Tri glavne točke na dlanu su R8, R13, R18 (Slika 12.). To su sve točke koje se nalaze između dva susjedna prsta. Povezivanjem točke R8 i R18 dobijemo y os koordinatnog sustava, nakon toga povučemo okomitu crtu kroz R13 i dobijemo x os koordinatnog sustava. Pretpostavlja se da prsti nisu stisnuti zajedno te da su svi prsti osim palca vidljivi na slici.



Slika 12: glavne točke segmentacije

Postupak dobivanja kvadratnog dijela dlana za ekstrakciju značajki je sljedeći:

1. Koristimo prag α za konvertiranje originalne sive slike (slika 13. (a)) u binarnu mapu (slika 13. (b)).

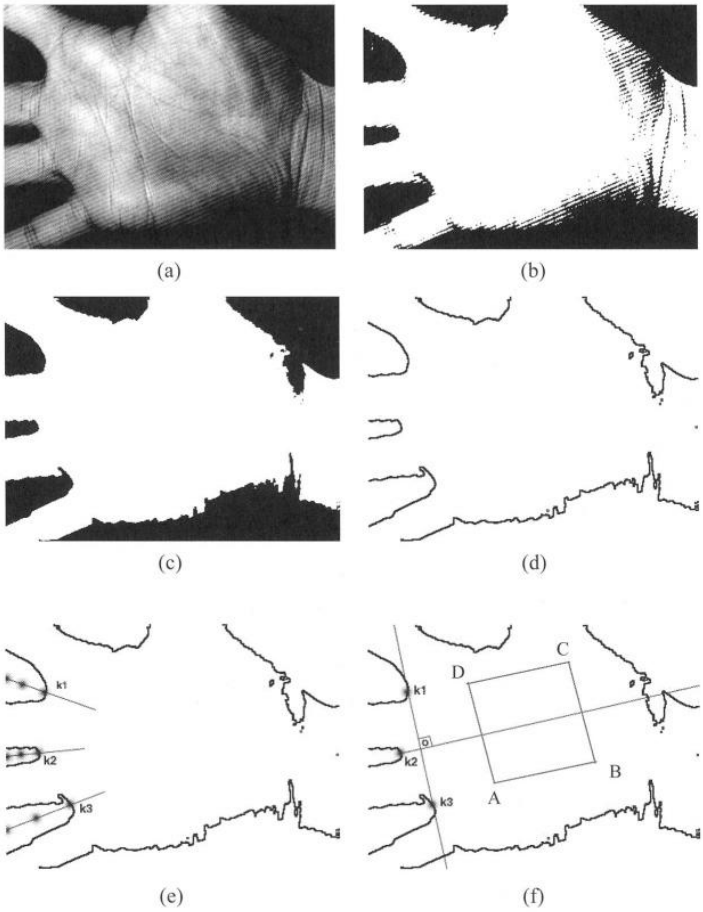
$$I_{binarna\ mapa}(i,j) = \begin{cases} 1 & I_{siva\ mapa}(i,j) \geq \alpha \\ 0 & I_{siva\ mapa}(i,j) < \alpha \end{cases} \quad (10)$$

2. Izgladimo binarnu mapu pomoću Guassovih filtera (slika 13. (c)).

$$I_{izglađena\ mapa} = I_{binarna\ mapa} * A \quad (11)$$

   A označava Guassov filter.

3. Nalazimo rubove između prstiju (slika 13. (d)).
4. Tražimo sredinu između prstiju i dobivamo ključne točke, k1,k2 i k3 (slika 13. (e)).
5. Povučemo liniju iz k1 do k3 da bi dobili y-os, te povlačimo okomitu liniju u odnosu na y-os kroz k2 da bi dobili ishodište tog koordinatnog sustava (slika 13. (f)).
6. Nakon određivanja koordinatnog sustava, fiksna podslika centralnog dijela dlana je ekstrahirana. Na slici 13 (f) to je kvadrat „ABCD“. Pozicija točke „A“ i veličina kvadrata „ABCD“ je fiksna na svim slikama.



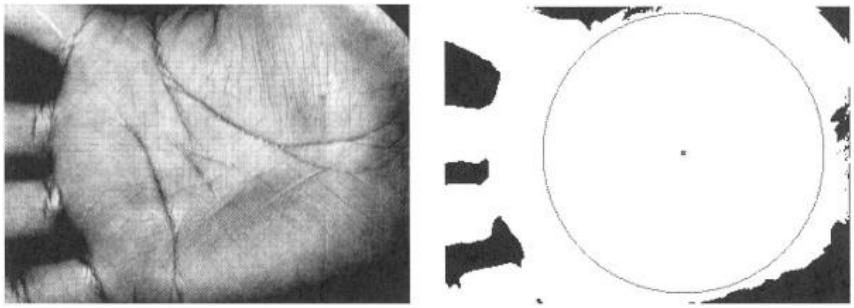
Slika 13: Postupak određivanja podslike na dlanu

Ova metoda segmentacije mora točno odrediti rupe između prstiju, jer inače postoji mogućnost od krivog određivanja centralnog dijela podslike što u konačnici umanjuje točnost cijelog sustava za identifikaciju.

5.2.2. Upisana kružna segmentacija

U odnosu na kvadratnu segmentaciju ovaj postupak baziran je na upisanoj kružnici za ekstrakciju centralnog dijela dlana.

Glavna ideja je naći rubove dlana te između njih upisati kružnicu tako da ona obuhvaća što veći prostor sredine dlana.



Slika 14:Kružna segmentacija

Postupak kružne segmentacije:

1. Za početak moramo odrediti rubove dlana. Prvo konvertiramo sivu sliku u binarnu mapu koristeći funkcije (10) i (11), nakon toga izgladimo rubove i dobijemo obris dlana.
2. Odredimo najveću upisanu kružnicu u odnosu na rubove dlana i odredimo njeno središte i radijus. Kako su dlanovi različiti tako je i radijus kružnice različit. Isti dlan imati će slične radijuse i središta kružnice. Kada uspoređujemo dlanove prvo usporedimo radijus. Ako radijusi odstupaju jedan od drugoga puno odmah možemo isključiti njihovu jednakost. U slučaju da imaju sličan radijus idemo na sjedeću usporedbu. Pretpostavljamo da su im centri blizu no ne i nužno u istoj točci. Stoga za usporedbu dlanova jedan dlan se rotira.
3. Nakon određivanja kružnice vršimo ekstrakciju svih piksela unutar nje.

Ako ćemo usporediti ove dvije metode mogli bi reći da je metoda kružnice točnija jer obuhvaća veći dio dlana te se može koristiti u više slučajeva, no kvadratna metoda je brža pošto analizira manji broj piksela. Ovisno o svrsi i potrebi sustava možemo birati metodu koja nam više odgovara[10].

5.3. Tekstura dlana uz korištenje 2-D Gabor filtera

Ova metoda koristi slike manje rezolucije za osobnu identifikaciju. Koristi se tekstura dlana tako da koristimo prilagođeni Gabor filter da bi dobili tu informacije o teksturi dlana.

Kao što smo ranije naveli postoje 3 glavne linije na dlanu. No one nisu adekvatno i dovoljno mjerilo za sličnost dlana, zbog postojanja sličnosti tih linija kod više dlanova. Zbog toga nabori imaju veliku ulogu kod identifikacije. Točno ekstrahiranje nabora je još uvijek zahtjevan zadatak.

Koristimo centralni dio dlana fiksne veličine, primjerice(100x100). Centralni dio dobijemo jednom od metoda koje smo ranije naveli. Korištenje tih metoda segmentacije smanjujemo potrebe rotacija slika istoga dlana.

Točku uzorka filtrirane slike stavljamo u 2 bita (b_r , b_i) prema sljedećim uvjetima:

$$b_r = 1 \text{ if } Re[G[x,y,\theta,u,\sigma]*l] \geq 0 \tag{12}$$

$$b_r = 0 \text{ if } Re[G[x,y,\theta,u,\sigma]*l] < 0 \tag{13}$$

$$b_i = 1 \text{ if } Im[G[x,y,\theta,u,\sigma]*l] \geq 0 \tag{14}$$

$$b_i = 0 \text{ if } Im[G[x,y,\theta,u,\sigma]*l] < 0 \tag{15}$$

$$G(x,y,\theta,u,\sigma) = \frac{1}{2\pi\sigma^2} exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\} exp\{2\pi i(ux\cos\theta + uy\sin\theta)\} \tag{16}$$

G[x,y,θ,u,σ] je diskretna Gaborova funkcija gdje je $i = \sqrt{-1}$, l centralni dio podslike dlana, θ kontrolira orijentaciju funkcije, u je frekvencija sinusoidnog vala, σ je standardna devijacija Gaussianovog omotača. Koristeći ovu metodu samo fazni podatci na slici dlana spremaju se u vektor značajki.

Značajke uspoređujemo na način da uzmemo 2 vektora, realni i imaginarni kao 2-D matrice značajki. Uspoređivanje dlana se bazira na normaliziranoj Hammingtovoj udaljenosti. Ako su P i Q dvije matrice značajki normaliziranu Hammingtovu udaljenost računamo prema sljedećoj formuli.

$$D_o = \frac{\sum_{i=l}^{N}\sum_{j=l}^{N}(P_r(i,j) \otimes Q_r(i,j) + P_i(i,j) \otimes Q_i(i,j))}{2N^2} \tag{17}$$

Gdje je $P_r(Q_r)$ i $P_i(Q_i)$ realan i imaginarni dio od P (Q). Boolean operator ⊗ jednak je nuli ako i samo ako su oba bita $P_{r(l)}(i,j)$ i $Q_{r(l)}(i,j)$ jednaka i ako je veličina matrice značajki NxN. D_o poprima vrijednosti od 0 do 1. Hammingtona udaljenost koja označava savršeno preklapanje je 0.

5.4. Značajke linija i njihova reprezentacija

Različita istraživanja dovodila su do različitih metoda za ekstrakciju značajki linija dlana. Neki istraživači koriste Sobelov filter i morfološke operacije za ekstrakciju značajki u obliku linija dlana dobivenih skeniranjem, dok drugi koriste usmjerene maske za ekstrakciju tih istih značajki dobivenih korištenjem digitalne kamere. No nijedna od metoda za online uspoređivanje dlanova ne koristi eksplicitno značajke linija dlana makar su te značajke robusne na razlike u osvjetljenju i šum na slici te mogu opisati dlan jasno da se može uspoređivati [11].

5.4.1. Izlučivanje linija i nabora dlana

Linije dlana, uključujući i glavne linije i nabore su sličnog tipa kao krovovi(kada pričamo o detekciji rubova)[12]. Pozicija rubnih točaka dobije se tako da se nađu točke u kojima se mijenja predznak (pozitivno u negativno ili suprotno) kada koristimo prvi derivat. Odnosno točke gdje dolazi do nepovezanosti prvog derivata na slici sivih tonova. Magnituda drugog derivata rubnih točaka može nam pokazati njihovu snagu(intenzitet). Ta svojstva koristimo u nalaženju linija dlana. Smjer linija na dlanu je proizvoljan te je vrlo teško direktno pronaći linije dlana na slikama sa šumom. Linije detektiramo ovisno u smjeru θ kojeg koristimo.

Pretpostavimo da I(x,y) predstavlja neku sliku. Konstruiramo horizontalni detektor linija tako da uzmemo θ vrijednost 0°. Da bi povećali povezanost i zaglađenost linija koristimo 1-D Gaussovu funkciju G_{σ_s} sa varijancom σ_s na smjeru linija koje želimo zagladiti. Operator * označava konvoluciju.

$$I_s = I * G_{\sigma_s} \tag{18}$$

Derivacije prvog i drugog reda vertikalnog smjera mogu biti izračunate korištenjem konvolucije na izglađenoj slici sa prvom (G'_{σ_s}) i drugom (G''_{σ_s}) derivacijom 1-D Gaussove funkcije G_{σ_d} sa varijancom σ_d:

$$I' = I_s * (G'_{\sigma_d})^T = \left(I * G_{\sigma_s}\right) * \left(G'_{\sigma_d}\right)^T = I * \left(G_{\sigma_s} * \left(G'_{\sigma_d}\right)^T\right) = I * H_1^0 \tag{19}$$

$$I'' = I_s * (G''_{\sigma_d})^T = \left(I * G_{\sigma_s}\right) * \left(G''_{\sigma_d}\right)^T = I * \left(G_{\sigma_s} * \left(G''_{\sigma_d}\right)^T\right) = I * H_2^0 \tag{20}$$

Gdje je $H_1^0 = G_{\sigma_s} * \left(G'_{\sigma_d}\right)^T$, $H_2^0 = G_{\sigma_s} * \left(G''_{\sigma_d}\right)^T$, T označava transponiranost.

H_1^0 i H_2^0 su detektori horizontalnih linija. Horizontalne linije možemo dobiti tako da tražimo nultočke od I' u vertikalnom smjeru, dok njihov intenzitet označuju odgovarajuće točke u I''.

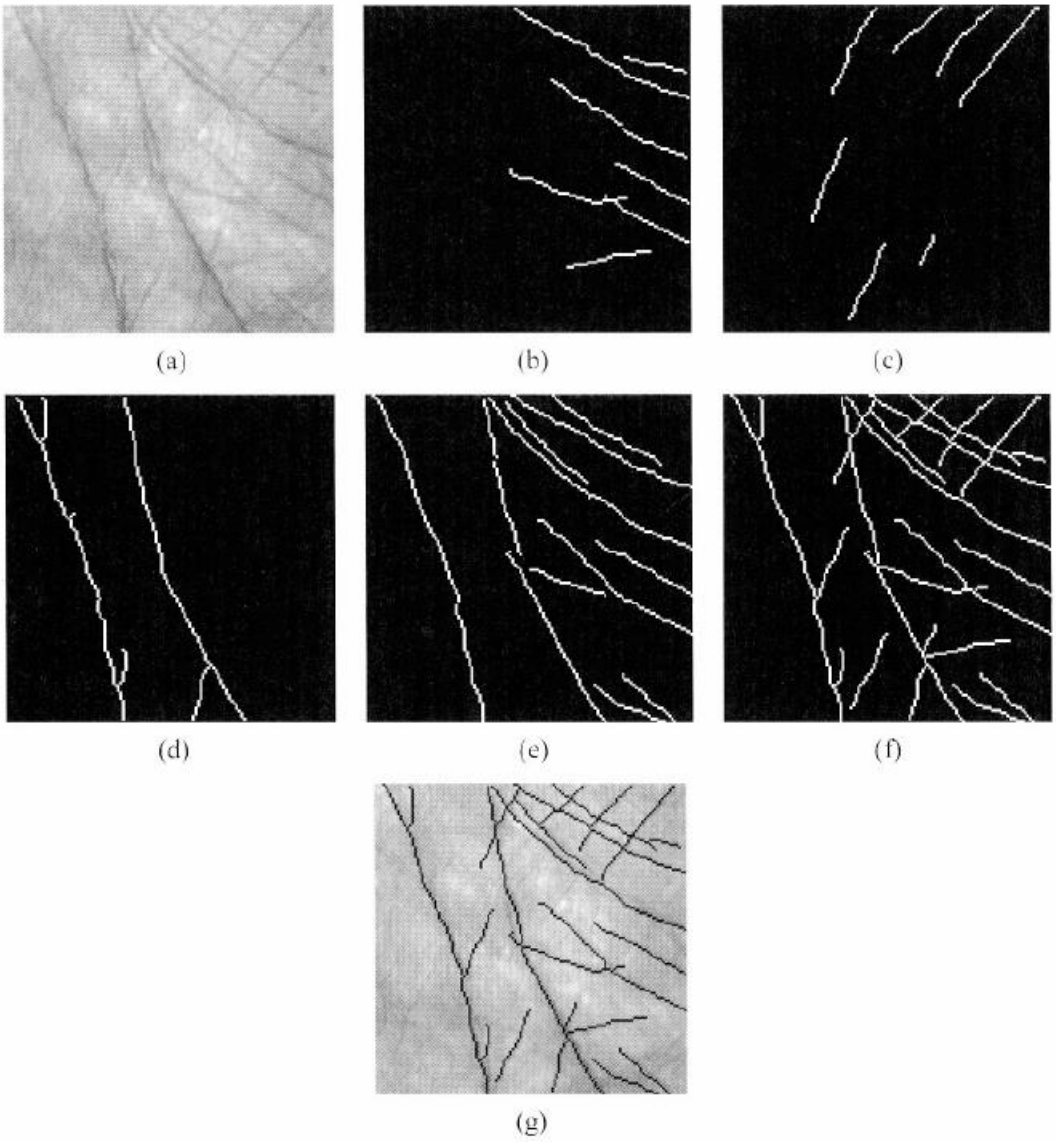
$$L_0^1(x,y) = \begin{cases} I''(x,y), ako\ I'(x,y) = 0\ ili\ I'(x,y) \times I'(x+1,y) < 0 \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad inače \end{cases} \tag{21}$$

Također možemo odrediti i tip linije, odnosno označava li ona udubljenje ili ispupčenje. Ako nam je vrijednost $L_0^1(x,y)$ pozitivna (+) ona nam označava udubljenje, dok nam negativna (-) označava ispupčenje. Kako su sve linije dlana udubljenja negativne vrijednosti možemo zanemariti.

$$L_0^2(x,y) = \begin{cases} L_0^1(x,y), \qquad\qquad akoL_0^1(x,y) > 0 \\ 0 \qquad\qquad\qquad\qquad inače \end{cases} \tag{22}$$

Linije dlana su puno gušče nego grebeni. Iz tog razloga možemo koristiti nekoliko pragova koji će nam pomoći u izbacivanju grebena iz L_0^2 u svrhu dobivanja binarne slike L_0 koju još nazivamo slika smjera linija u 0^0 smjeru. Detektore smjera linija H_1^θ i H_2^θ u θ smjeru možemo

dobiti rotacijom H_1^0 i H_2^0 za kut θ. Točke linije možemo dobiti tražeći nultočke u θ + 90^0 smjeru. Nakon odbacivanja najdubljih linija, grebena i ostalih točaka koje nam ne odgovaraju u odnosu na postavljeni prag možemo odrediti binarnu sliku linija L_θ u θ smjeru. Za kraj moramo povezati sve slike smjera linija L_θ u jednu sliku L koja prikazuje sve linije. Na slici 15. pokazan je taj proces. (a) je originalna slika, (b) – (e) nam prikazuje slike smjera linija na 0^0, 45^0, 90^0, 135^0 stupnjeva, (f) je konačna slika sa linijama iz svih prethodnih smjerova i (g) je preklapanje originalne slike s slikom svih ekstrahiranih linija.



Slika 15: Prikaz nalaženja linija dlana

Imamo dva parametra σ_s i σ_d u detektorima linija, σ_s kontrolira povezanost i uglađenost linija dok σ_d upravlja sa širinom linija koje možemo detektirati. Mali σ_s rezultira slabom zaglađenosti i slabom povezanosti detektiranih linija, dok veliki σ_s rezultira gubitkom nekih kratkih linija i gubitkom nekih segmenata linija gdje je velika zakrivljenost. Tanki rubovi linija ne mogu biti ekstrahirani ako je σ_d prevelik. Tako da te parametre treba odrediti u skladu

navedenim. Odnosno σ_s bi trebao biti velik, a σ_d mali zato što su linije dlana generalno dugačke, uske i uglavnom ravne. Prema nekim istraživanjima vrijednosti $\sigma_s = 1.8$ i $\sigma_d = 0.5$ su dobre za ekstrakciju linija dlana.

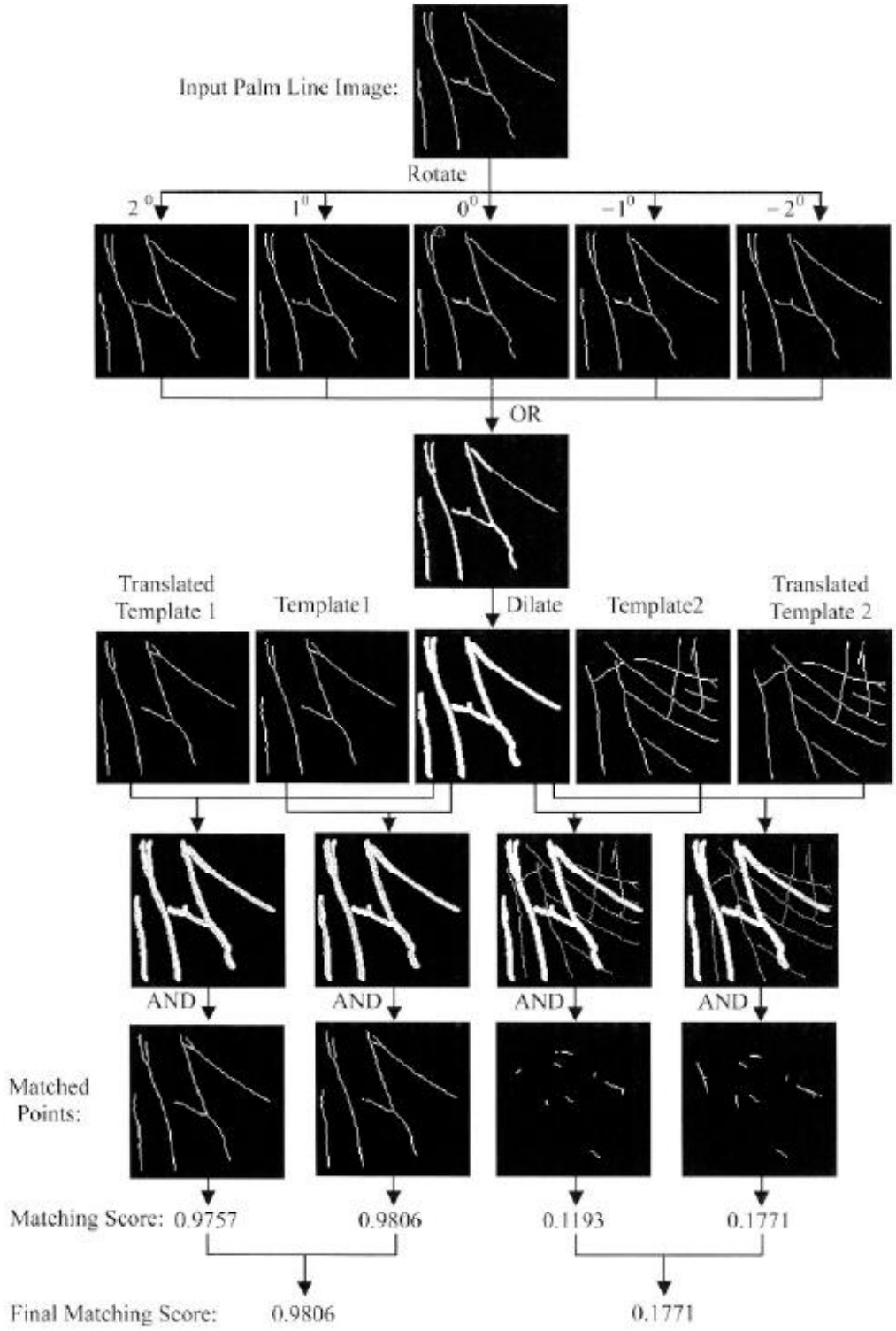
Pošto su linije dlana neregularni uzorci, odnosno nasumične linije teško ih je reprezentirati u matematičkom obliku. Efikasnija metoda reprezentacije je ulančani kod [13]. Kod smjera (0 ,1 , …, 7) je definiran za svaku točku na liniji u odnosu na njenog sljedbenika. To predstavlja ulančani kod svake linije. Svaku liniju možemo prikazati pomoću njezine početne točke i njenog koda smjera.

5.4.2. Povezivanje odgovarajućih linija

Neka L predstavlja sliku dlana, a C predstavlja predložak linija ulančanim kodom. Najjednostavniji način usporedbe je prebrojavanje svih točaka na koje su na istoj poziciji na slici L i na slici L_C koju dobijemo iz predloška C. Međutim zbog postojanja pogrešaka točke istoga dlana ne moraju biti na istim koordinatama na slikama istog dlana koje slikamo u različito vrijeme. Da bi riješili ovaj problem moramo proširiti L na L_D te izbrojati točke koje se preklapaju. Rezultat preklapanja računa se prema sljedećoj formuli:

$$S(L,C) = \frac{2}{M_L+M_C} * \sum_{i=1}^{M}\sum_{j=1}^{N}[L_D(i,j) \wedge L_C(i,j)] \tag{23}$$

Gdje je ∧ operator i, M x N je veličina slike L, M_C i M_L su točke linija na L_C i L slikama.



Slika 16: Proces uspoređivanja linija dlana

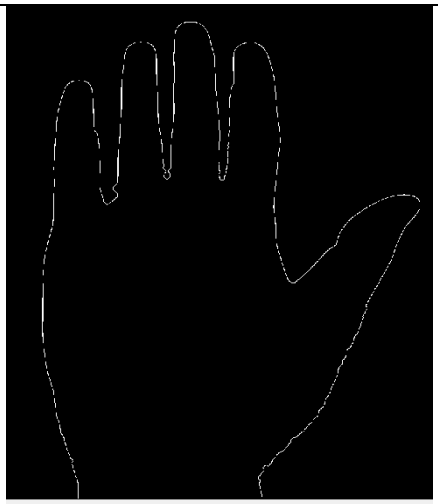
Za jednadžbu (23) moramo skenirati cijelu sliku, što zahtijeva dosta vremena. No ona je ista kao i sljedeća jednadžba.

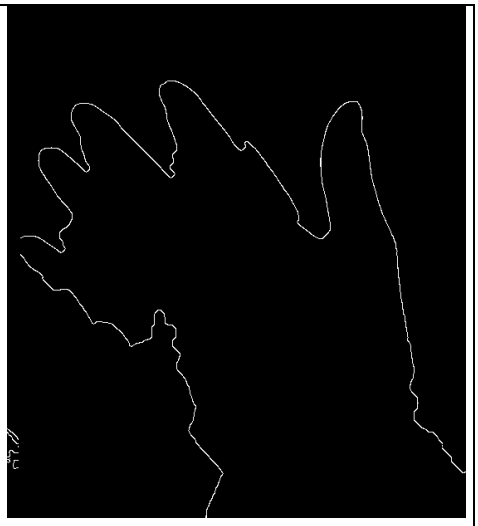
$$S(L,C) = \frac{2}{M_L+M_C} * \sum_{i=1}^{M_C} L_D(x_i, y_i) \tag{24}$$

L_D nam predstavlja proširenu sliku od L, M_C je zbroj svih točaka linija iz C i (x_i i y_i) (i=1 ,... ,M_C) su njihove koordinate, a M_L je broj točaka linija iz L. Očito je da je računalna složenost jednadžbe (24) manja od složenosti jednadžbe (23). Štoviše ako koristimo (24) jednadžbu za izračunati preklapanje, umjesto da vratimo sliku dlana L_C iz C, moramo samo vratiti koordinate točaka linija. Sličnost S(L,C) između L i C računa nalazi se između vrijednosti 0 i 1, što je ta vrijednost bliža broju 1 to su slike sličnije. Savršeno preklapanje je ako nam je vrijednost S(L,C) = 1 no zbog ne savršenosti predprocesiranja slike dlana moguća je mala rotacija i greška u prijenosu između dvije slike dlana slikanih u različito vrijeme. Da bi minimizirali taj efekt rotacije rotiramo L za nekoliko stupnjeva par puta u oba smjera te onda spojimo sve rotirane slike koristeći logički operator ILI i dobijemo spojenu binarnu sliku L_M. Nakon toga sliku L_M proširujemo da bi dobili L_{MD}. Koristimo L_{MD} da bi izračunali točnost preklapanja. Da bi riješili problem prijenosa, vertikalno i horizontalno prenosimo sliku C za nekoliko točaka te na svakoj prenesenoj poziciji računamo sličnost između danog prebačenog C i $L_{MD.}$ Na kraju uzmemo najveću vrijednost od svih testiranih te to smatramo kao finalnu ocjenu sličnosti. Slika 16 nam slikovito pokazuje ovaj navedeni proces. Ulazna slika linija dlana i Template 1 predstavljaju sliku jednog dlana, dok Template 2 predstavlja drugi dlan. Kada usporedimo ulaznu sliku s Template 1 dobijemo vrijednost 0.9806, a kod Templeta 2 ta vrijednost je svega 0.1771, što nam jasno govori o kojemu odnosno čijemu se dlanu radi.

6. Eksperimenti na raznim tipovima slika dlana

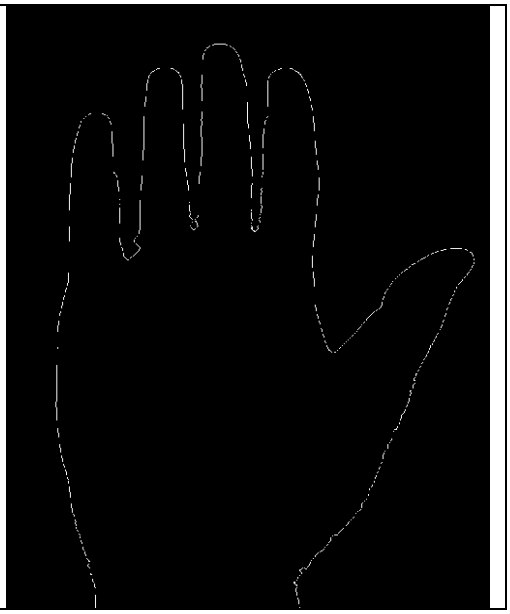
Prvi eksperiment koji ćemo probati je određivanje rubova dlana (eng. edge detection). To je osnovni korak za segmentaciju dlana. Pokušati ćemo detektirati rubove dlana na 3 slike različitih kvaliteta i na taj način doći do lokacije dlana na slici. Tipovi slika će biti: slika dlana s jednoličnom pozadinom, slika dlana s nejednoličnom, kompleksnom pozadinom i slika dlana sa sjenom na dlanu.

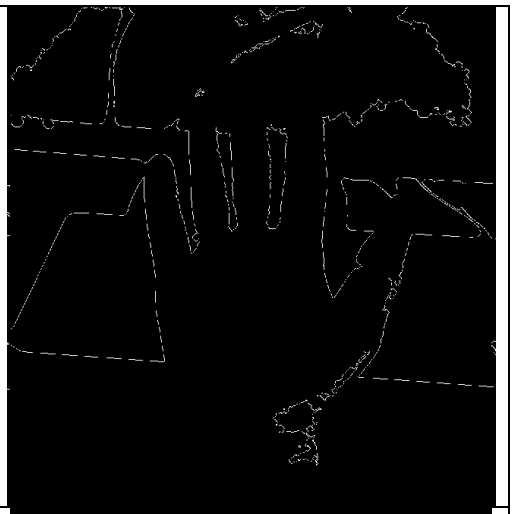
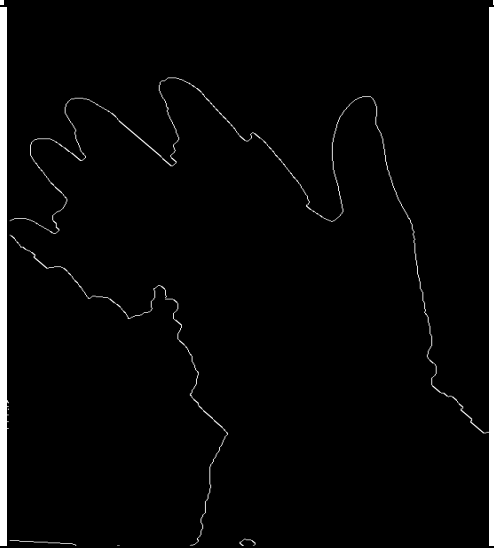
Dlan s jednoličnom pozadinom – Kao što se vidi u idealnim (laboratorijskim) uvjetima detektor rubova daje konture dlana, čak se i prsten na ruci može eliminirati jednostavnom naredbom imfill() koja popunjava rupice koje nastaju kod detekcije linija zbog prstena.		
Slika dlana sa nejednoličnom, kompleksnom pozadinom – Dlan je prepoznatljiv no sve daljnje radnje s takvom slikom bile bi jako teške te bi se svaka takva slika trebala posebno obraditi da bi došlo do nekog stanja u kojem se mogu uzeti značajke dlana.		

Slika dlana sa sjenom na dlanu – Dlan je neprepoznatljiv te ga se kao takvog ne može dalje procesirati.		

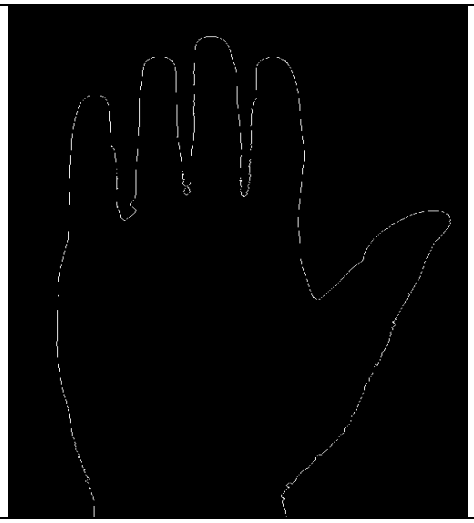
Na prijašnjem primjeru koristili smo Canny metodu detekcije rubova. Canny metoda je najmanje podložna šumovima i najbolje detektira slabije označene rubove. Dalje ćemo testirati i ostale metode detekcije te provjeriti dali ima kakvih značajnih razlika.

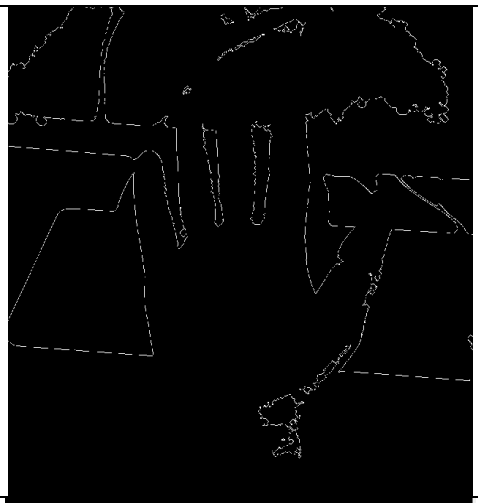
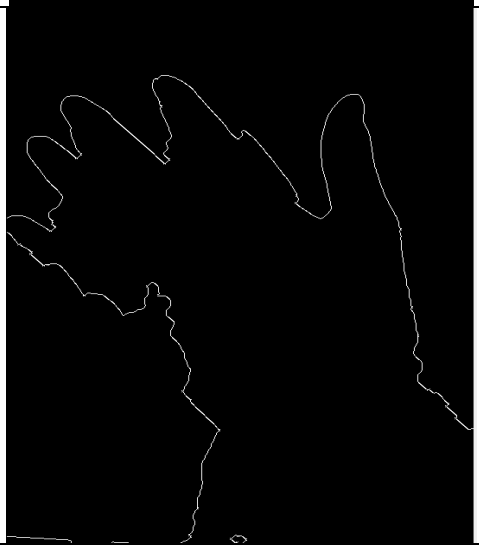
Sobel metoda :

Dlan s jednoličnom pozadinom		

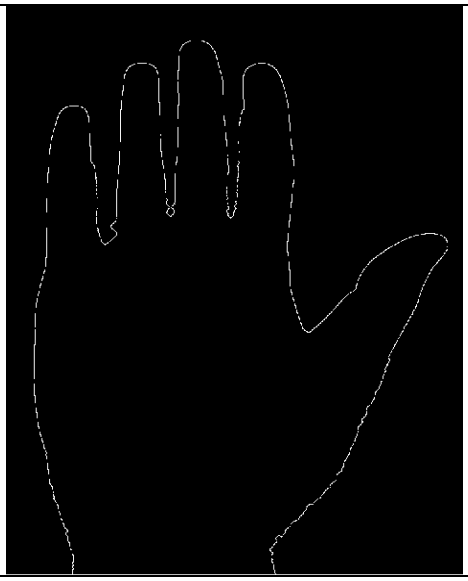
Dlan s nejednoličnom pozadinom		
Slika dlana sa sjenom na dlanu		

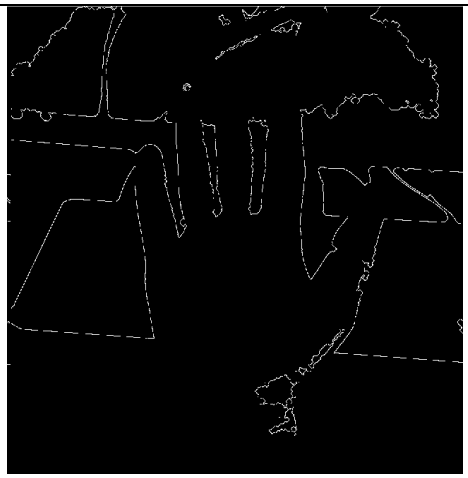
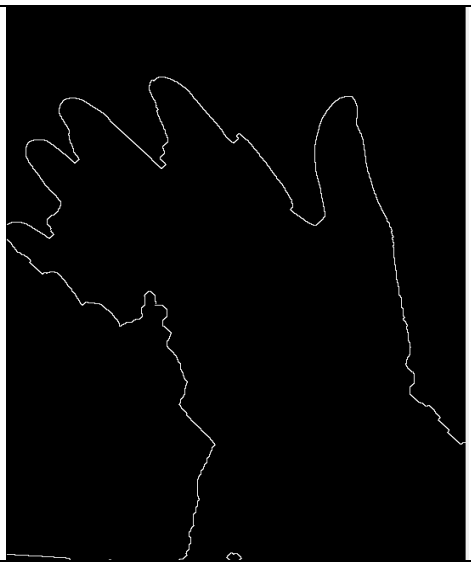
Prewitt metoda :

Dlan s jednoličnom pozadinom		

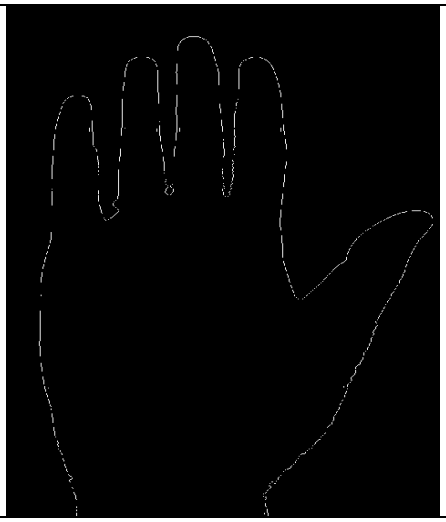
Dlan s nejednoličnom pozadinom		
Slika dlana sa sjenom na dlanu		

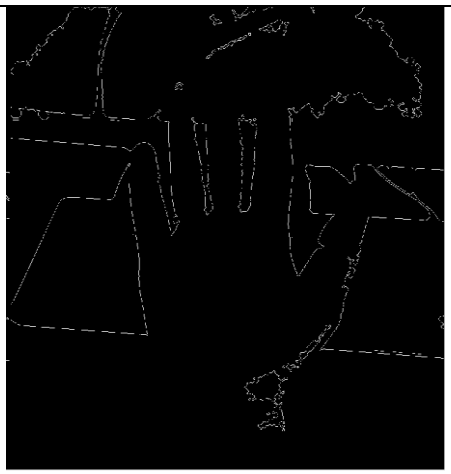
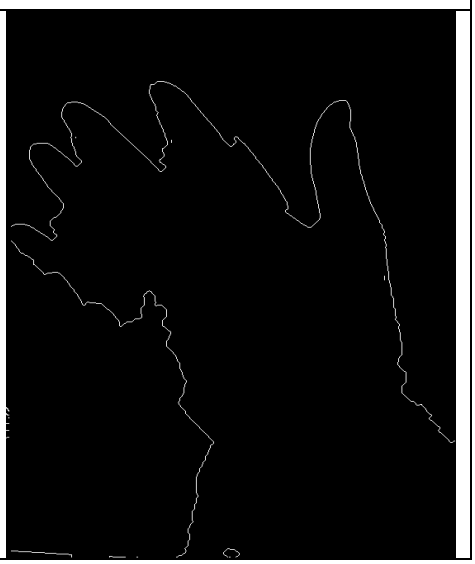
Roberts metoda :

Dlan s jednoličnom pozadinom		

Dlan s nejednoličnom pozadinom		
Slika dlana sa sjenom na dlanu		

Log metoda :

Dlan s jednoličnom pozadinom		

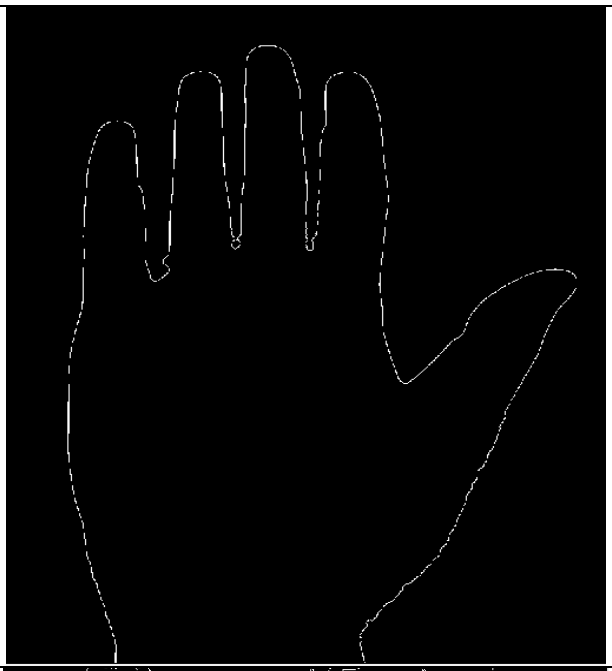
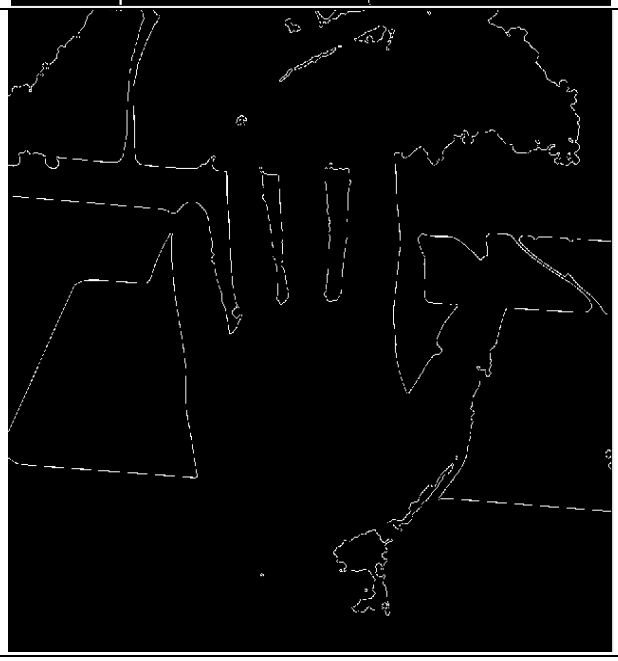
Dlan s nejednoličnom pozadinom		
Slika sa sjenom na dlanu		

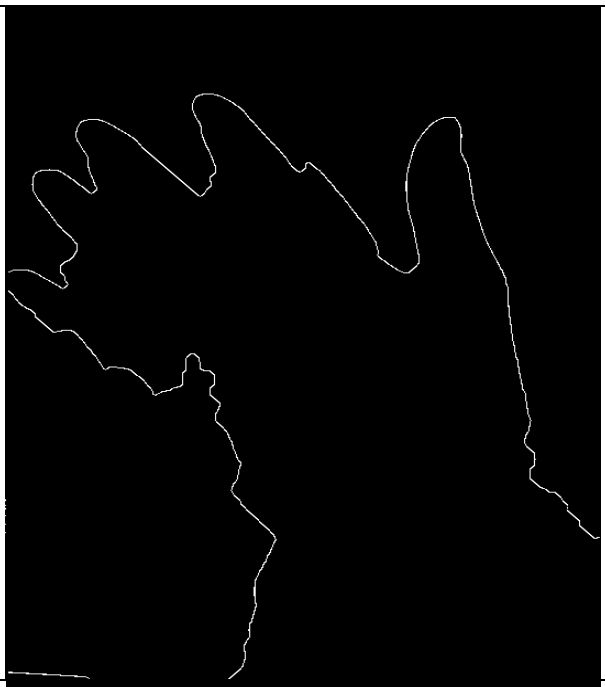
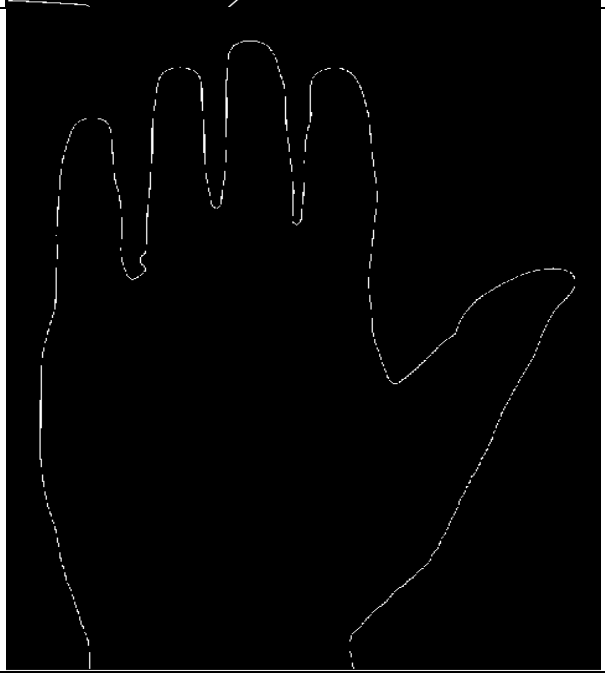
Metoda za detekciju rubova ima više. Za testiranje su korištene Canny, Prewitt, Roberts, Sobel i log metode za detekciju rubova te sve daju približno slične rezultate. Jedine uočljive razlike su dužine detektiranih linija i male razlike u detekciji kutova, no te razlike su minimalne. Kod koji smo koristili za detekciju dlana dan je u prilogu 1.

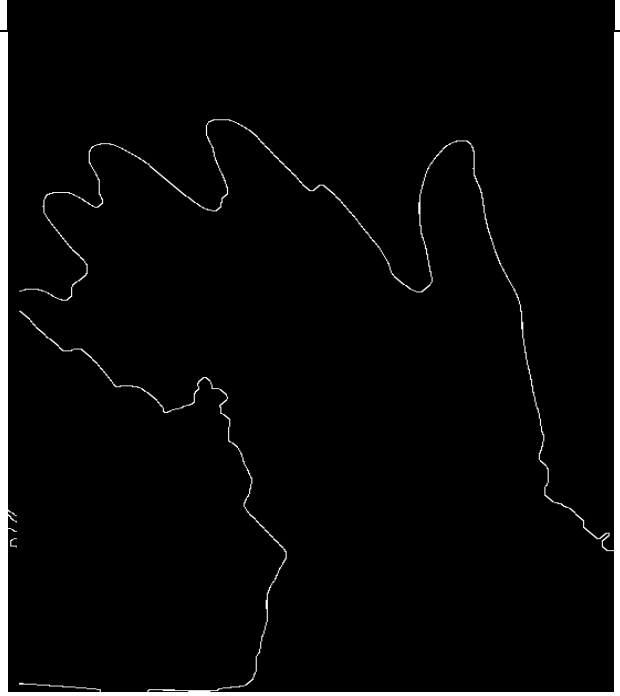
Također možemo pokušati poboljšati rezultate korištenjem raznih filtera za izoštravanje rubova. Najpopularniji takav filter je Gaussian filter. Matlab funkcija koju koristimo je

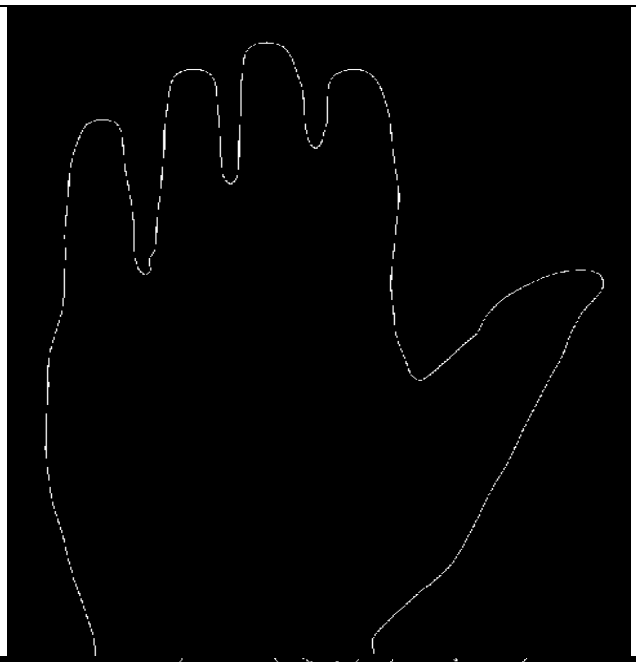
```
i = imgaussfilt(i, σ);
```

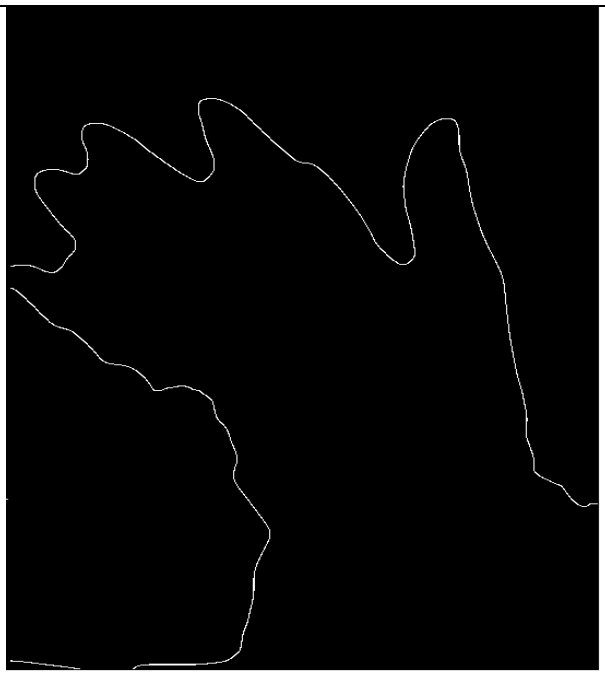
Testirati ćemo ga s 3 različite vrijednosti parametra sigma te prokomentirati rezultate.

σ = 0.5		
		

			
σ = 2			

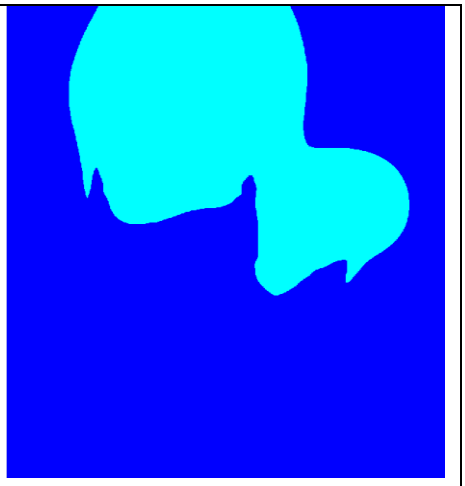
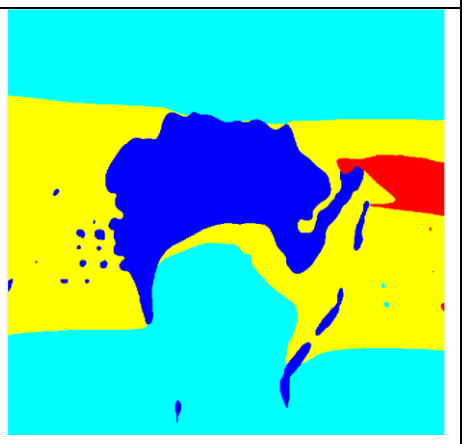
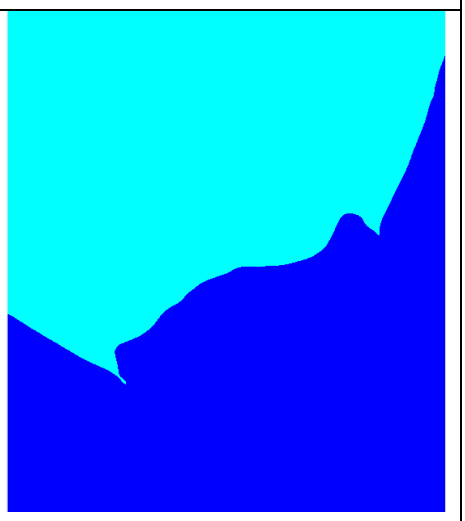
		

σ = 6		

Korištenje Gaussian doprinijelo je manjoj isprekidanost linija na nekim dijelovima slike. Dlan s jednoličnom pozadinom nam gubi na jasnoći rubova što više povećavamo sigmu dok primjerice na slici s ne jednoličnom pozadinom jasnije možemo odrediti pojedine elemente slike kako povećavamo sigmu. Slika sa sjenom gubi na konturama prstiju što više povećavamo sigmu. Iz ovih rezultata možemo zaključiti da nam Gaussian filter najviše odgovara kod slika s raznim objektima u pozadini jer jasnije možemo odrediti rubove tih objekata.

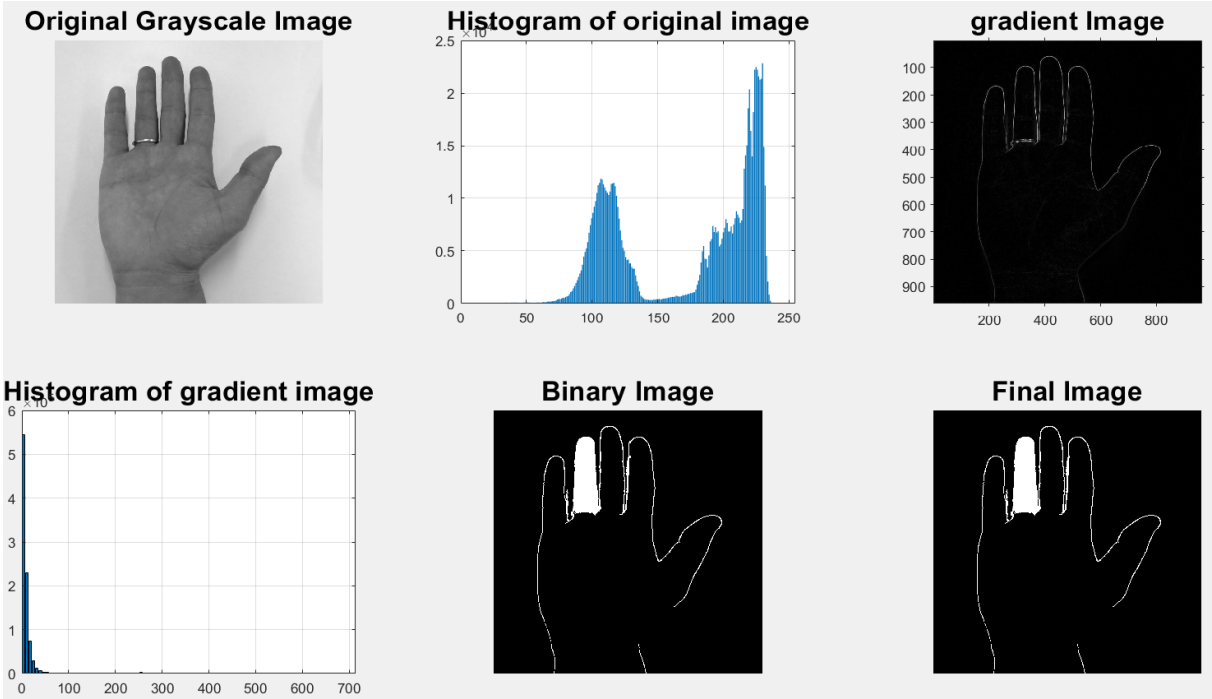
Sljedeći eksperiment vršit ćemo na istom skupu slika. Sada ćemo pokušati koristiti Gabor filtere da bi dobili informaciji o teksturi koja se nalazi na slikama te na taj način ekstrahirali samo dlan. Ovisno o slici moramo u kodu promijeniti podatke o koliko se različitih tekstura na slici nalazi.

Dlan s jednoličnom pozadinom – Makar je slika relativno kvalitetna, rezultati koje dobijemo nisu zadovoljavajući. Dlan je ne prepoznatljiv.		
Dlan s ne jednoličnom pozadinom – Kao i na prvoj slici ne možemo prepoznati dlan, unijeli smo da ima 4 razne teksture, to su dlan, stol, plava pozadina monitora i crna tipkovnica i dio monitora.		
Slika sa sjenom na dlanu – Dlan je kao i na prijašnjim slikama ne prepoznatljiv.		

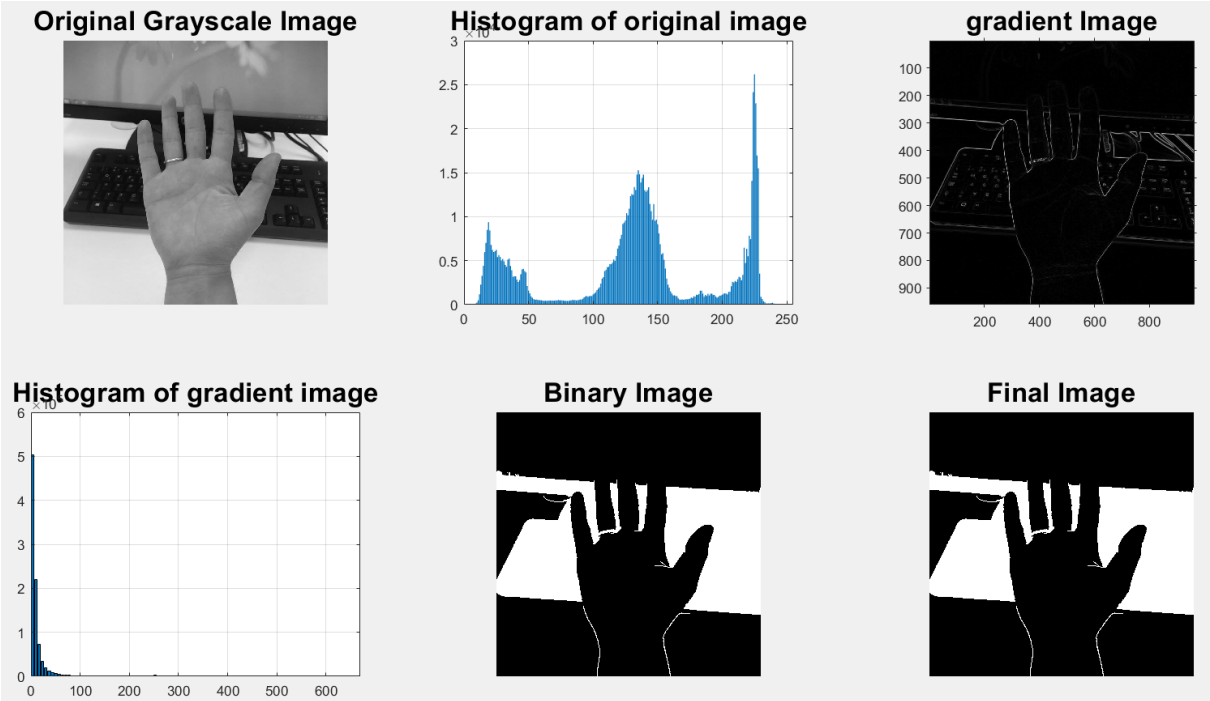
Kod koji smo koristili za detekciju dlana dan je u prilogu 2.

Iz eksperimenta se vidi da je detekcija rubova puno bolji način za detektirati dlan na slici od segmentacije pomoću Gaborovih filtera.

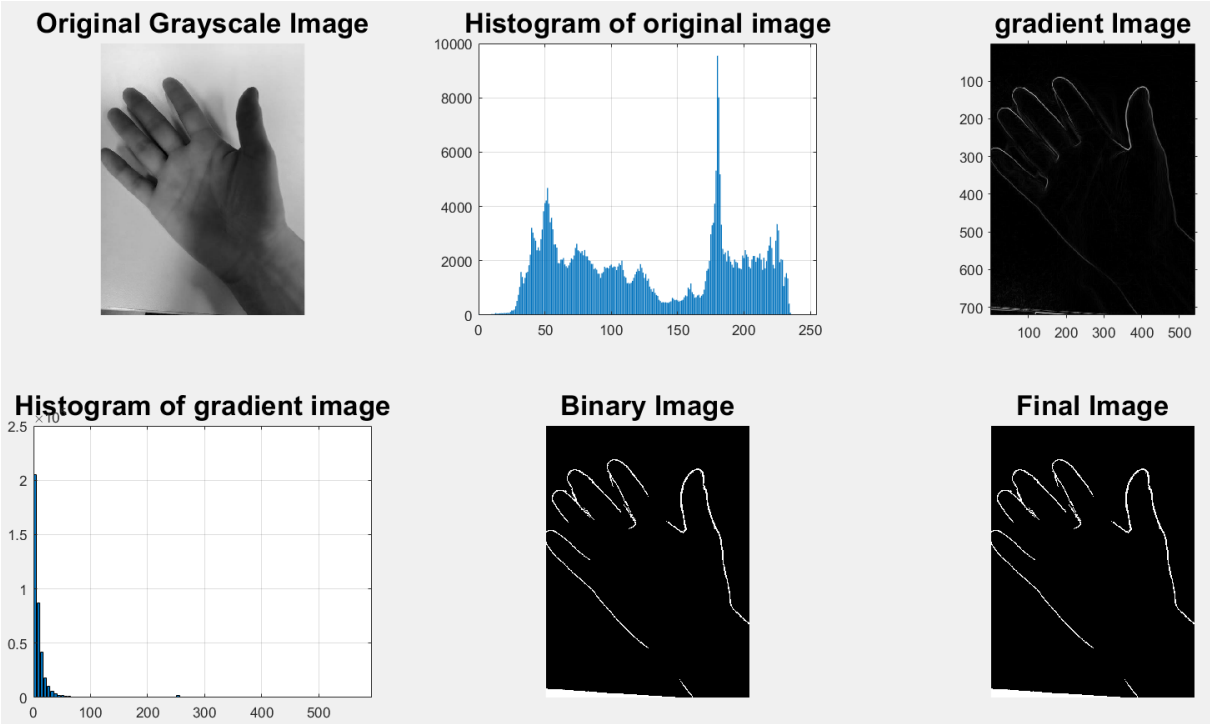
Još jedna metoda za pronalazak dlana je detekcija grumena (područja piksela sličnih svojstva). Testirat ćemo tu metodu na 3 slike dlana kao i u prošlim eksperimentima.



Na prvoj slici koja ima najmanje oscilacija u boji pozadine i koja ima najbolju predodžbu dlana dobivamo kvalitetan otisak uz jedan problem. Detektor grumena detektira prstenjak kao zasebni element zbog prstena. Tako da bi jedna od prije navedenih metoda bila jednostavnija jer se ne bi morala korigirati greška nastala zbog prstena.



Druga slika sa ne jednoličnom pozadinom daje dobre rezultate ako imamo u umu da nam je ROI(region of intrest) sredina dlana koju bi s ovako dobivenim rezultatima mogli ekstrahirati.



Zadnja slika koja je na prošlim eksperimentima davala loše rezultate sada daje dosta kvalitetnu predodžbu dlana te njegovu lokaciju. Na takvoj slici bi se dalo ekstrahirati ROI nakon sitnih preinaka i spajanja linija dlana.

Kod koji smo koristili za detekciju grumena dan je u prilogu 3.

7. Zaključak

U ovom radu opisan je biometrijski sustav za identifikaciju osoba korištenjem dlana i način na koji on funkcionira. Dan je pregled biometrijskih značajki dlana kao što su geometrija dlana, glavne linije, nabori, grebeni i singularne točke te način njihovog prikupljanja i daljnje obrade kako bi se koristili za identifikaciju osoba. Pobliže je opisano više načina izlučivanja značajki dlana kao što je registracija krajnjih točaka dlana i segmentacija dlana za određivanje geometrijskih karakteristika dlana, korištenje Gaborovih filter a za određivanje teksture dlana te postupci izlučivanja linija i nabora na dlanu i njihovo povezivanje. Iz toga smo saznali da postoje biometrijski sustavi za određivanje i verificiranje osoba različitih složenosti već na razini prikupljanja informacija, a posebno na razini obrade informacija da bi postale korisni podaci za usporedbu uzoraka i identifikaciju osoba.

Radi se o sustavima koji su u današnjem društvu postali neophodni za kontrolu ulaska u različitim objektima i pristupa različitim aplikacijama tako da će se njihov razvoj i pouzdanost neprestano usavršavati. Sustavi od kojih se ne očekuje visoka razina točnosti postali su sveprisutni i koriste se već npr. na mobilnim telefonima i laptopima za autentifikaciju korisnika. U budućnosti očekujem sve veću primjenu biometrijskih sustava u mnogim aspektima ljudskog života, neki mogući primjeri su upravljanje razno raznim kućanskim uređajima te kontroliranje njihovih parametara ovisno o osobi koja ih želi koristiti, paljenje motornih vozila preko biometrijskih značajki i postavljanje granica brzine ovisno o upravljaču vozila, automatizirana narudžba raznih namirnica ovisno o biometrijskom potpisu osobe.

8. Literatura

[1] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

[2] Ivan Fratrić, Biometrijska verifikacija osoba temeljena na značajkama dlana i lica dobivena iz video sekvenci, Zagreb 2011

[3] Joseph N. Pato and Lynette I. Millett, Biometric Recognition challanges and opportunities , The National Academic Press 2010

[4] Dakshina Ranjan Kisku, Phalguni Gupta, Jamuna Kanta Sing,Advances in Biometrics for Secure Human Authentication and Recognition, Tayler & Francis Group LLC 2014

[5] Anil K. Jain, Patrick Flynn, Arun A. Ross, Handbook of Biometrics, 2008 Springer Science

9. Reference

[1][2] Joseph N. Pato and Lynette I. Millett, Biometric Recognition challanges and opportunities , The National Academic Press 2010, str. 2

[3] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 7

[4][5] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 11

[6] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 15

[7] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 39

[8] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 40

[9] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 42

[10] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 80

[11] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 99

[12] David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers, str 99

[13] http://www.mind.ilstu.edu/curriculum/chain_codes_intro/chain_codes_intro.php

10. Slike

Slika 1. Joseph N. Pato and Lynette I. Millett, Biometric Recognition challanges and opportunities , The National Academic Press 2010

Slika 2. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers,

Slika 3. http://www.m2sys.com/blog/scanning-and-efficiency/iris-recognition-vs-palm-vein-biometrics-how-do-they-compare/

Slika 4. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 5. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 6. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 7. https://www.pinterest.com/pin/256001560038564343/

Slika 8. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 9. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 10. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 11. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 12. Murat Aykut, Murat Ekinci, Pattern Recognition Letters, 2013

Slika 13. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 14. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 15. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

Slika 16. David D. Zhang, Palmprint Authentification, 2004 Kluwer Academic Publishers

11. Prilog

Prilog 1.

```
clear
i = imread('dlan_X.jpg')

level = graythresh(i);

BW = im2bw(i,level); % pretvaramo RGB sliku u binarnu
figure;
imshowpair(i,BW,'montage')

dilated_BW = imdilate(BW,strel('diamond',5)) % prosirujemo, odnosno zagladujemo
rubove
dilated_BW = imfill(~dilated_BW,'holes') % ispunjavamo rupe nastale zbog lose
kvalitete slike

BW2 = edge(BW,'canny'); % algoritam za detekciju rubova
BW22 = edge(dilated_BW,'canny');

figure;
imshowpair(BW2,BW22,'montage')
title('Canny Filter                    Canny Filter dilated');
```

Prilog 2.

```
clear
A = imread('dlan_X.jpg');


Agray = rgb2gray(A);
figure
imshow(A)

imageSize = size(A);
numRows = imageSize(1);
numCols = imageSize(2);
```

```
wavelengthMin = 4/sqrt(2);
wavelengthMax = hypot(numRows,numCols);
n = floor(log2(wavelengthMax/wavelengthMin));
wavelength = 2.^(0:(n-2)) * wavelengthMin;

deltaTheta = 45;
orientation = 0:deltaTheta:(180-deltaTheta);

g = gabor(wavelength,orientation);

gabormag = imgaborfilt(Agray,g);

for i = 1:length(g)
    sigma = 0.5*g(i).Wavelength;
    K = 3;
    gabormag(:,:,i) =
imgaussfilt(gabormag(:,:,i),K*sigma);
end

X = 1:numCols;
Y = 1:numRows;
[X,Y] = meshgrid(X,Y);
featureSet = cat(3,gabormag,X);
featureSet = cat(3,featureSet,Y);

numPoints = numRows*numCols;
X = reshape(featureSet,numRows*numCols,[]);

X = bsxfun(@minus, X, mean(X));
X = bsxfun(@rdivide,X,std(X));

coeff = pca(X);
feature2DImage = reshape(X*coeff(:,1),numRows,numCols);
figure
imshow(feature2DImage,[])

L = kmeans(X,2,'Replicates',5); % kod modificiramo ovisno
koliko raznih tekstura ima na slici

L = reshape(L,[numRows numCols]);
figure
imshow(label2rgb(L))

Aseg1 = zeros(size(A),'like',A);
Aseg2 = zeros(size(A),'like',A);
BW = L == 2;
BW = repmat(BW,[1 1 3]);
Aseg1(BW) = A(BW);
```

```
Aseg2(~BW) = A(~BW);
figure
imshowpair(Aseg1,Aseg2,'montage');
```

Prilog 3.

```
function test1()
close all;  % zatvaramo otvorene slike
clear;  % brisemo postojuce varijable
format long g;
format compact;
fontSize = 20;

folder = 'C:\Users\nicma\Documents\MATLAB';
baseFileName = 'dlan_X.jpg';
% uzimamo puno ime filea
fullFileName = fullfile(folder, baseFileName);
% provjeravamo dali file postoji
if ~exist(fullFileName, 'file')
    % file ne postoji - nismo ga pronasli,  trazimo u
searchu
    fullFileNameOnSearchPath = baseFileName; % nema
putanje do filea
    if ~exist(fullFileNameOnSearchPath, 'file')
        % jos uvijek nema filea
        errorMessage = sprintf('Error: %s ne postoji.',
fullFileName);
        uiwait(warndlg(errorMessage));
        return;
    end
end
grayImage = imread(fullFileName);
% dobivamo dimenziju slike
% numberOfColorBands treba biti = 1.
[rows, columns, numberOfColorBands] = size(grayImage);
if numberOfColorBands > 1
    % ako nije grayscale - slika je u boji
    % pretvaramo je u grayscale uzimajuci samo zeleni
kanal
    grayImage = grayImage(:, :, 2); % uzmemo zeleni kanal
end
% prikazemo grayscale sliku
subplot(2, 3, 1);
```

```
imshow(grayImage, []);
title('Original Grayscale Image', 'FontSize', fontSize);
% povecavamo sliku preko cijelog ekrana
set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1
1]);
% imenujemo naslov
set(gcf, 'Name', 'Demo by ImageAnalyst', 'NumberTitle',
'Off')
% racunamo i prikazujemo histogram
[pixelCount, grayLevels] = imhist(grayImage);
subplot(2, 3, 2);
bar(grayLevels, pixelCount);
grid on;
title('Histogram of original image', 'FontSize',
fontSize);
xlim([0 grayLevels(end)]);
% trazimo gradient slike da bi nasli rubove dlana
gradientImage = imgradient(grayImage);
% stavljamo bijelu liniju na dnu,
% desnom i lijevom kraju da bi zatvorili sliku dlana i
ruke
gradientImage(:,1) = 255; % cinimo prvi stupac bijelim
gradientImage(end,:) = 255; % cinimo zadnji red bijelim
gradientImage(:,end) = 255; % cinimo zzadnji stupac
bijelim
% prikazemo sliku
subplot(2, 3, 3);
imshow(gradientImage, []);
axis on;
title('gradient Image', 'FontSize', fontSize);
% racunamo i prikazujemo histogram
[pixelCount, grayLevels] = hist(gradientImage(:), 100);
subplot(2, 3, 4);
bar(grayLevels, pixelCount);
grid on;
title('Histogram of gradient image', 'FontSize',
fontSize);
xlim([0 grayLevels(end)]);
binaryImage = gradientImage > 60;
% ispunjavamo rupe
binaryImage = imfill(binaryImage, 'holes');
% micemo sitne dijelove
binaryImage = bwareaopen(binaryImage, 1000);
% prikazemo sliku
subplot(2, 3, 5);
imshow(binaryImage, []);
title('Binary Image', 'FontSize', fontSize);
```

```
%-------------------------------------------------------------
------------------
% ekstrahiramo najvece dijelove koristeci funkciju
ExtractNLargestBlobs().
biggestBlob = ExtractNLargestBlobs(binaryImage, 3);
% prikazemo sliku
subplot(2, 3, 6);
imshow(biggestBlob, []);
title('Final Image', 'FontSize', fontSize);
%-------------------------------------------------------------
------------------
%=============================================================
=====================================
% funkcija vraca specifican broj najvecih ili najmanjih
grumena u binarnoj slici
% ako numberToExtract > 0 vraca numberToExtract najvecih
grumena
% ako numberToExtract < 0 vraca numberToExtract najmanjih
grumena
% primjer: vraca binarnu sliku s najvecim grumenom:
%   binaryImage = ExtractNLargestBlobs(binaryImage, 1)
% primjer: vraca binarnu sliku s 3 najmanja grumena:
%   binaryImage = ExtractNLargestBlobs(binaryImage, -3)
function binaryImage = ExtractNLargestBlobs(binaryImage,
numberToExtract)
try
    % dobivamo svojstva grumena
    [labeledImage, numberOfBlobs] = bwlabel(binaryImage);
    blobMeasurements = regionprops(labeledImage, 'area');
    % dobivamo sva podrucja
    allAreas = [blobMeasurements.Area];
    if numberToExtract > 0
        % za pozitivne brojeve, sortiramo od najveceg
prema namjanjem
        [sortedAreas, sortIndexes] = sort(allAreas,
'descend');
    elseif numberToExtract < 0
        % za negativne brojeve, sortiramo od najmanjeg do
najveceg
        [sortedAreas, sortIndexes] = sort(allAreas,
'ascend');
        % negiramo numberToExtract da bi ga koristili u
sortIndexes poslije
        numberToExtract = -numberToExtract;
    else
        % numberToExtract = 0.  ne smije se desit, ne
vraca grumen
```

```
        binaryImage = false(size(binaryImage));
        return;
    end
    % ekstrahiramo "numberToExtract" najveci grumen
koristeci ismember().
    biggestBlob = ismember(labeledImage,
sortIndexes(1:numberToExtract));
    % konvergiramo iz integer oznacene slike u binarnu
(logicku) sliku
    binaryImage = biggestBlob > 0;
catch ME
    errorMessage = sprintf('Error u funkciji
ExtractNLargestBlobs().\n\nError Message:\n%s',
ME.message);
    fprintf(1, '%s\n', errorMessage);
    uiwait(warndlg(errorMessage));
end
```