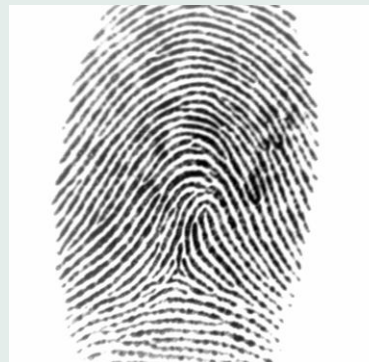


1. Pirštų atspaudų atpažinimas

1. Įvadas
2. Piršto atspaudų taikymai
3. Pirminis apdorojimas
4. Požymių išskyrimas
5. Požymių šablonų palyginimas
6. Praktinis darbas
7. Literatūra



Įvadas Piršto atspaudų taikymai

Piršto atspaudai yra taikomi:

- Kriminalistikoje identifikuojant asmenis
- Pasuose įrašant skaitmenine forma asmens piršto atspaudu duomenis bei juos lyginant pvz. oro uostuose
- Įmonėse organizuojant automatinę darbuotojų įleidimo į darbo vietą bei lankomumo registracijos sistemą

- Masinio aptarnavimo sistemose, pvz. Maximos tipo parduotuvėse greitam kortelės tapatybės patvirtinimui
- Bankinėse sistemose identifikuojant kortelės asmens tapatybę.
- Individualizuojant prieigą prie įrenginių, pvz. asmeninio kompiuterio, mobiliojo telefono.
- Atominėse elektrinėse, režiminėse įstaigose ir t.t.

Pirminis

piršto

atspaudo

vaizdo

apdorojimas

http://bias.csr.unibo.it/maltoni/handbook/extract_from_4.3.pdf

Požymių išskyrimas Alexej Kočetkov magistro darbas

Požymių šablonų palyginimas

Požymių šablonų palyginimą pasiaiškinsime FVC 2002 DB1 A bazės pirmųjų dviejų (1_1 ir 1_2) paveikslėlių pavyzdžiu.

Iš 2 paveikslėlio matyti, kad tai yra to paties asmens vieno piršto dviejų skenavimų rezultatai. Tokius pirštų porų atvejus vadinsime autentiškomis (angl. *genuine*). Jei paveikslėlių pora būtų skirtingų asmenų ar vieno asmens bet skirtingų pirštų, kad turime apsišaukėlių porą (angl. *impostors*).

Laikysime, kad abu atspaudai yra apdoroti ir juose išskirti požymiai. Vienas požymis turi tokius atributus: $\{t_i, x_i, y_i, \alpha_i\}$. Čia

Home Page

Title Page

◀

▶

◀

▶

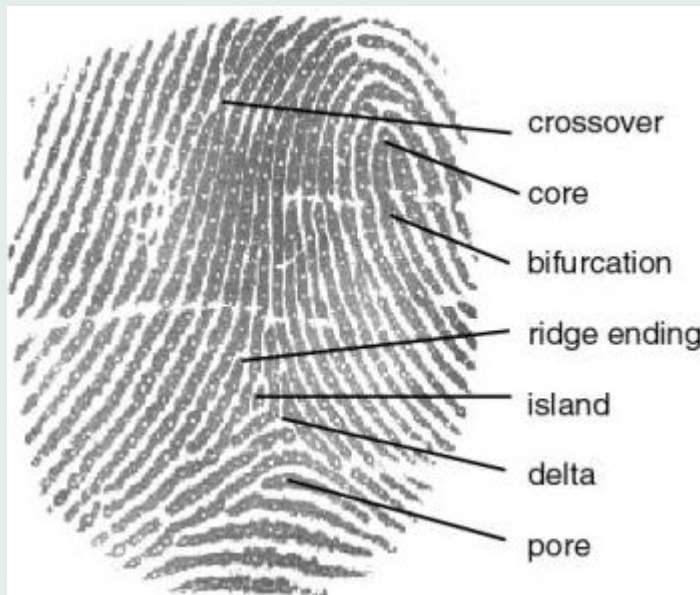
Page 3 of 21

Go Back

Full Screen

Close

Quit



1 paveikslėlis: Piršto atspaudų požymių angliški terminai

- i - i -asis požymis.
- t_i - i -ojo požymio tipas (0 - neapibrėžtas, 1 - linijos galas, 2 - linijos bifurkacijos (išsišakojimo į dvi dalis) taškas).
- (x_i, y_i) - požymio centrinio taško koordinatės. x matuojama taškeliais nuo kairės į dešinę, y -



2 paveikslėlis: FVC 2002 DB1 A bazės 1.1 (kairiau) ir 1.2 (dešiniau) pirštų atspaudų pavyzdžiai

taškeliais nuo viršaus į apačią. Viršutinio kairiojo paveiksluko kampo koordinatės yra $(0, 0)$.

- α_i kampas, kurį sudaro i -ojo požymio vektorius einantis linijos liestinės kryptimi iš požymio centrinio taško. Linijos galo atveju vektoriaus kryptis nukreipta link linijos, išsišakojimo atveju link išsišakojimo. Krypties vektorius pasirinkimas taip todėl, kad krypties požymis būtų stabi-



3 paveikslėlis: FVC 2002 DB1 A bazės 1.1 ir 1.2 pirštų atspaudai ir jų požymiai (detales). Detalės išskirtos panaudojant **verifinger** demonstracinį algoritmą

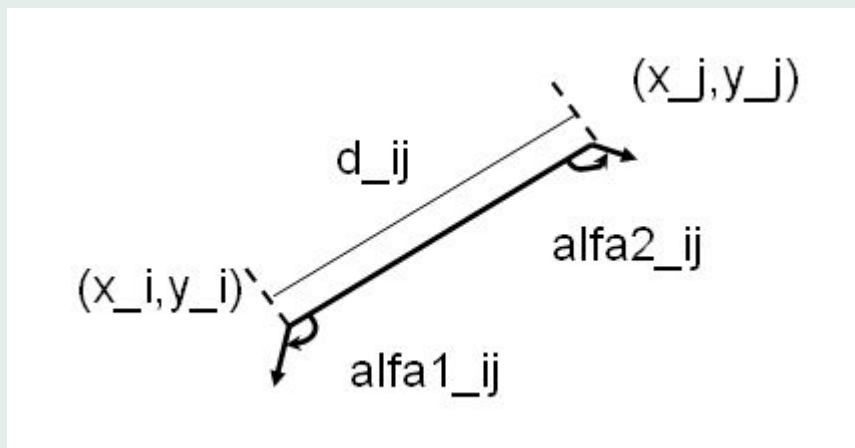
lesnis požymio tipo nustatymo klaidoms.

3 pav. grafiškai iliustruoja išskirtus požymius. Nesunku pastebėti, kad minučių požymių skaitinės reikšmės nėra invariantiškos standartinėms geometrinėms transformacijoms: poslinkiui ir posūkiui.

Kad išspręsti šią problemą pirminiais lyginamais atributais laikysime ne požymių taškus, o linijas sudarytas iš dviejų gretimų taškų. Linijai aprašyti naudosime jos ilgį $d_{i,j}$ ir du kampus $\alpha 1_{i,j}$ ir $\alpha 2_{i,j}$, kuriuos sudaro atkarpa su požymių krypčių vektoriais. Abu kampai matuojami nuo atkarpos link požymio taško vektoriaus skaičiuojant prieš laikrodžio rodyklę. Nesunku pastebėti, kad šie atkarpų požymiai bus invariantiniai (nekintantys) geometrinėms transformacijoms, kurios nekeičia atstumo tarp taškų. Todėl dviejų pirštų atkarpų požymius galima tiesiogiai lyginti ir jų artumas bus faktorius didinantis tikimybę, kad atkarpų linijų galai atitinka tuos pačius požymių centrų taškus.

Kyla klausimas kokias požymių taškų poras apjungti atkarpomis. Pasirinkimas sudaryti visas galimas taškų poras nėra geriausias, nes susidarytų labai daug linijų ir būtų sunku sukurti efektyvų (greitą) atkarpų palyginimo algoritmą bei gali kilti triukšmo problema dėl klaidingai išskirtų minučių. Galimi įvairūs sprendimai. Galima fiksuoti i -ojo požymio centrą ir surasti jam visas *detales* (angliškas požymio centro *minutia* terminas) nutolusias ne daugiau pasirinkto atstumo R . Kitas variantas rekomenduojamas JAV nacionalinio standartų instituto (NIST) suskaidyti i -ojo požymio atžvilgiu atspaudą į aštuonis sektorius ir kiekviename sektoriuje pasirinkti po vieną ar dvi artimiausias *detales*.

5 pav. iliustruoja požymio aplinkas išsirinktas dviem būdais. Pirmuoju atveju atkarpomis sujungiami nutolę nedidesniu pasirinktu atstumu R taškai. Kitu atveju pasirinktos i -osios *detales* atžvilgiu



4 paveikslėlis: Du požymio taškus jungiančių linijų invariantiniai geometrinės transformacijos požymiai

vaizdas padalinamas į keturis arba aštuonis sektorius ir apjungiamos kiekviename sektoriuje esančios artimiausios *detalės*. Sektorių skaidymas ir numeracija pradedama pasirinktos centrinės *detalės* krypties, t.y. nulinio sektoriaus vienas kraštų eina detalės kryptimi.

Praktinis darbas *Realizuokite piršto atspaudų palyginimo algoritmą.*

Pirštų atspaudų atpažinimo užduotis pasirenkama laisvai, kai atliktos privalomos užduotys. Darbas atliekamas auditorijoje pratybų metu arba savarankiškai namuose atsiskaitant auditorijoje pratyboms skirtu laiku už atskirus etapus.

Home Page

Title Page



Page 8 of 21

Go Back

Full Screen

Close

Quit



5 paveikslėlis: Požymių aplinka

Pirštų vaizdų pavyzdžius rasite [fvc2002db1a.zip](#) archyve. Pirštų atspaudų požymis rasite [FVC2002.DB1_A.info](#) faile.

1. Nuskaitykite piršto atspaudų požymius. Vieno piršto požymiai įrašyti vienoje teksto eilutėje tokiu formatu:

katalogoNr failoVardas id pirštoPožymiųSkaičius tipas xKoordinatė yKoordinatė kryptis . . . tipas
xKoordinatė yKoordinatė kryptis

Požymių failo pradžioje yra antraštė, kurioje pateikiama bendra visiems pirštų atspaudams informacija. Po "String katalogas[1]" eilutės parašytas katalogo vardas(ai), kuriame patalpinti pirštų atspaudų paveikslėliai. Šią eilutę galite pakeisti, kad atitiktų jūsų kompiuteryje patalpintų rainelių pavyzdžių katalogą(us).

Trumpai apibūdinsime parametrus.

- (a) *katalogoNr* numeris katalogo, kuriame patalpintas rainelės vaizdas. Jei antraštės eilutė yra *String katalogas[1]*, tai visi paveikslėliai yra viename kataloge ir tuomet visi katalogo numeriai lygūs 0.
- (b) *failoVardas* yra *tif* formato failas, kuriame yra nuskenuoto piršto atspaudų vaizdas. Failai patalpinti http://uosis.mif.vu.lt/atpazinimas/finger/FVC/2002/DB1_A/ kataloge.

- (c) *id* piršto atspaudų unikalūs identifikatoriai. Autentiškų pirštų pora turi tą patį *id*.
- (d) *pirštoPožymiųSkaičius* yra sveikasis skaičius, kurio reikšmė gali kisti nuo 0 iki 254.
- (e) *tipas* yra sveikasis skaičius, kurio reikšmė gali būti 0, 1 arba 2. 0 reiškia, kad požymio (detalės/minutia) tipas nežinomas, 1 - linijos galas, 2 - bifurkacija (linijos išsišakojimo į dvi dalis vieta).
- (f) *x koordinatė* yra sveikasis skaičius, kurio reikšmė gali kisti nuo 0 iki piršto atspaudų paveikslėlio *plotis* - 1. Laikoma, kad piršto atspaudų paveikslėlio viršutinio kairiojo taško koordinatės yra (0,0).
- (g) *y koordinatė* yra sveikasis skaičius, kurio reikšmė gali kisti nuo 0 iki piršto atspaudų paveikslėlio *aukštis* - 1.
- (h) *kryptis* yra sveikasis skaičius, kurio reikšmė gali kisti nuo 0 iki 255. 0 reiškia horizontalią kryptį nuo kairės į dešinę, $64 = 256/4$ yra vertikali kryptis nuo viršaus link apačios ir t.t. Krypčiai saugoti patogiau naudoti *byte* tipo kintamąjį, nes atliekant aritmetiką su byte skaičiais automatiškai gauname periodiškumą.

Žemiau pateiktas Java *Minutia.java* klasės pavyzdys, kuris apibrėžia duomenų struktūrą naudingą piršto atspaudų vienai *detalei* saugoti. Pagalbinė klasė *Tipas.java*

```
//package pirmstai;

public enum Tipas{NEZINOMAS, GALAS, BIFURKACIJA};

//package pirmstai;
import java.util.StringTokenizer;
import static java.lang.Math.*;
public class Minutia
{
    public static final int [] COLOR =
        { 0xff0000ff , 0xff00ff00 , 0xffffffff00 , 0xffff0000 };
    protected Tipas tipas;
    protected int x;
    protected int y;
    protected byte kryptis , kokybe;
    public Minutia(StringTokenizer st)
    {
        int t = Integer.parseInt(st.nextToken());
        tipas = ( t == 1 ) ? Tipas.GALAS :
```

```
( t == 2 ) ? Tipas.BIFURKACIJA : Tipas.NEZINOMAS;

x = Integer.parseInt(st.nextToken());
y = Integer.parseInt(st.nextToken());
kryptis = (byte)Integer.parseInt(st.nextToken());
kokybe = (byte)Integer.parseInt(st.nextToken());

}

public void mark(int[] data, int w, int h, int R)
{
    mark(data, w, h, COLOR[tipas.ordinal()], R);
}

public void mark(int[] data, int w, int h, int color, int R)
{
    int R1 = R*R-3*R, R2 = R*R+3*R;
    for (int i = max(-R,-x); i <= min(R,w-1-x); i++)
        for (int j = max(-R,-y); j <= min(R,h-1-y); j++)
            if (i*i+j*j>=R1 && i*i+j*j<=R2)
                data[x+i + w*(y+j)] = color;
```

```
double  c = cos(kryptis*PI/128),
        s = sin(kryptis*PI/128);
for (int l = 0; l != 2*R; l++)
{
    for (int ii=-1; ii!=2; ii++)
        for (int jj=-1; jj!=2; jj++) if (ii*ii+jj*jj!=2)
        {
            int i = (int)(0.5+max(0,min(w-1, x+l*c+ii))
            int j = (int)(0.5+max(0,min(h-1, y+l*s+jj))
            data[i + j*w] = COLOR[3];
        }
    }
}

public String toString()
{
    return "["+tipas+","+x+","+y+","+kryptis+","+kokybe+"]";
}
```

```
}
```

Klasėje yra metodas *mark* skirtas grafiškai pažymėti piršto atspaudų požymio parametrus *wxh* dydžio paveikslėlyje, kurio spalvos pateiktos `int[] data` masyve. Minutia klasės konstruktorius naudoja `StringTokenizer` klasės elementą iš kurio pakartotinai naudodamas `nextToken()` metodą gauna atskirų požymių tekstines reikšmes, kurios statiniu `Integer.parseInt()` metodu paverčiamos skaitinėmis reikšmėmis.

Nuskaičius požymius, sudarykite ir vizualizuokite piršto atspaudų detalių žvaigždines aplinkas. Žvaigždinės aplinkos vizualizacijos pavyzdys pateiktas 5 pav.

Atliekant užduotį rekomenduojama sukurti "Minutiae" - detalių aibės, "Line" - dvi minutes jungiančios linijos ir "Star" - iš vienos *detalės* išeinančių linijų aibės, struktūras. "Star" struktūrą sudaro centrinė detalė ir iš jos einančios linijos iki artimiausių kitų detalių pasirinktuose sektoriuose. Rekomenduojamas sektorių skaičius yra 4 arba 8, artimiausiųjų kaimynų sektoriuje - 1 arba 2. Pirmojo ir paskutiniojo sektorių jungimosi linija eina centrinės detalės kryptimi.

Žemiau pateiktas Java *Minutiae.java* klasės pavyzdys, apibūdinantis duomenų struktūrą naudingą piršto atspaudų visiems požymiams (detalėms, minutiae) saugoti ir vizualizuoti.

```
//package pirstai;
```

```
import java.util.StringTokenizer;

public class Minutiae
{
    protected int length, id, katalogoNr;
    protected String presentation;
    protected Minutia[] M;
    public Minutiae(String sData)
    {
        StringTokenizer st = new StringTokenizer(sData);
        katalogoNr = Integer.parseInt(st.nextToken());
        presentation = st.nextToken();
        length = Integer.parseInt(st.nextToken());
        M = new Minutia[length];
        for (int l = 0; l != length; l++)
            M[l] = new Minutia(st);
    }
    public void mark(int[] data, int w, int h)
```



```
{
    mark(data , w, h, 7);
}
public void mark(int [] data , int w, int h, int R)
{
    for (int l=0; l!=length; l++)
        M[l].mark(data , w, h, R);
}
}
```

Klasėje yra metodas *mark* skirtas grafiškai pažymėti piršto atspaudų požymius parametrus "w x h" dydžio paveikslėlyje. Minutiae klasės konstruktorius naudoja String elementą, kurio reikšmė yra viena "FVC2002_DB1_A,info" failo eilutė.

Vertinama iki 1.25 balo.

2. Sukurkite pirštų atspaudų linijų palyginimo metodą, pritaikykite jį "FVC2002_DB1_A.info" duomenų bazei ir įrašykite gautus rezultatus *.roc formatu. Reikiamą informaciją apie failų atitikmenis tiems patiems pirštams nustatysite pagal požymio failo *id* parametro reikšmes.

Home Page

Title Page

◀

▶

◀

▶

Page 17 of 21

Go Back

Full Screen

Close

Quit

Dviejų linijų panašumas (angl. *similarity*) įvertinamas neneigiamu sveikuoju skaičiu, gaunamu lyginant linijų ilgį (d) ir pradžios (α_1) bei galo (α_2) kampus (žiūr. pav.). Dviejų lyginamų linijų metrikos detalės pasakojamos paskaitos metu. Galutinė dviejų lyginamų pirštų atspaudų panašumo reikšmė yra gaunama susumuojant geriausias linijų panašumo vertes noruojant gautą sumą iš aritmetinio arba geometrinio vieno ir kito piršto atspaudų bendro linijų skaičiaus. Taupant skaičiavimo laiką yra lyginamos tik to paties sektoriaus linijos. Atliekant užduotį rekomenduojama sukurti "Finger" duomenų struktūrą, kuri būtų sudaryta iš "Star" struktūrų imant centrinėmis *detalemis* visas turimas piršto atspaudų detales saugomas "Minutiae" struktūroje.

Atlikite pirštų atspaudų palyginimo patobulinimą. Pirmu atveju sukaupkite geriausius linijų palyginimo taškus detalėse, kurias jungia linijos. Pavyzdžiui 1-ojo piršto 25-a linija, kurios pradžia yra 7-a, o galas 13-a detalė surinko daugiausiai panašumo taškų lyginant ją su 2-ojo piršto 20 linija, kurios pradžia yra 18-a, o galas 5-a detalė, ir atvirkščiai 2-ojo piršto 20 linija surinko daugiausia taškų, kai buvo lyginama su 1-ojo piršto 25-a linija. Tuomet pirmojo piršto 7-os ir antrojo piršto 18-os, bei pirmojo piršto 13-os ir antrojo piršto 5-os detalių porų kaupiamieji panašumo taškų skaitikliai yra padidinami linijų palyginimo taškais. Sukaupus pilnai taškų statistiką sudaromi atitikmenys tarp dviejų pirštų detalių. Atitikmenys sudaromi taikant maksimumo principą,

Home Page

Title Page

◀

▶

◀

▶

Page 18 of 21

Go Back

Full Screen

Close

Quit

pvz. laikoma, kad pirmojo piršto 7-a detalė atitinka antrojo piršto 18-ą detalę, jei (7,18) poros surinkti kaupiamieji panašumo taškai viršija visų likusių (7,y) ir (x,18) porų kaupiamuosius palyginimo taškus.

Suradus detalių atitikmenis iš geriausių linijų, iš ankstesnės užduoties rastos bendros linijų panašumo taškų sumos atmeskite panašumo taškus tų linijų, kurių bent vienas arba abu galai turi neatitinkančias detales. Lygindami "*" .roc" kreives suraskite koks atmetimo principas geresnis. Gal iš vis geriau neatmetinėti griežtai, o įvesti linijų panašumo taškams daugiklius, pvz. 1, kai abu linijų galai turi "teisingus" detalių atitikmenis, 1/2, kai tik vieno galo atitikmuo geras ir 1/4 likusiu atveju.

Vertinama iki 1.25 balo.

3. Pirštų atspaudų atpažinimo konkursas. Paruoškite pirštų atspaudų verifikacijos algoritmą, kuris apskaičiuotų duotų pirštų atspaudų duomenų (pirshtai.info) pirshtai.roc failą. Skaičiavimams turite perduoti dėstytojui algoritmus paruoštus tokiu būdu.

- Jūsų vykdomasis "*" .exe" arba java "*" .jar" failas (Matlab, Python ir kitos programavimo terpės neturi būti naudojamos) turi būti paleidžiamas komandine eilute, kurioje papildomai nurodomas vienas parametras: [FVC2002_DB1_A.info](#).

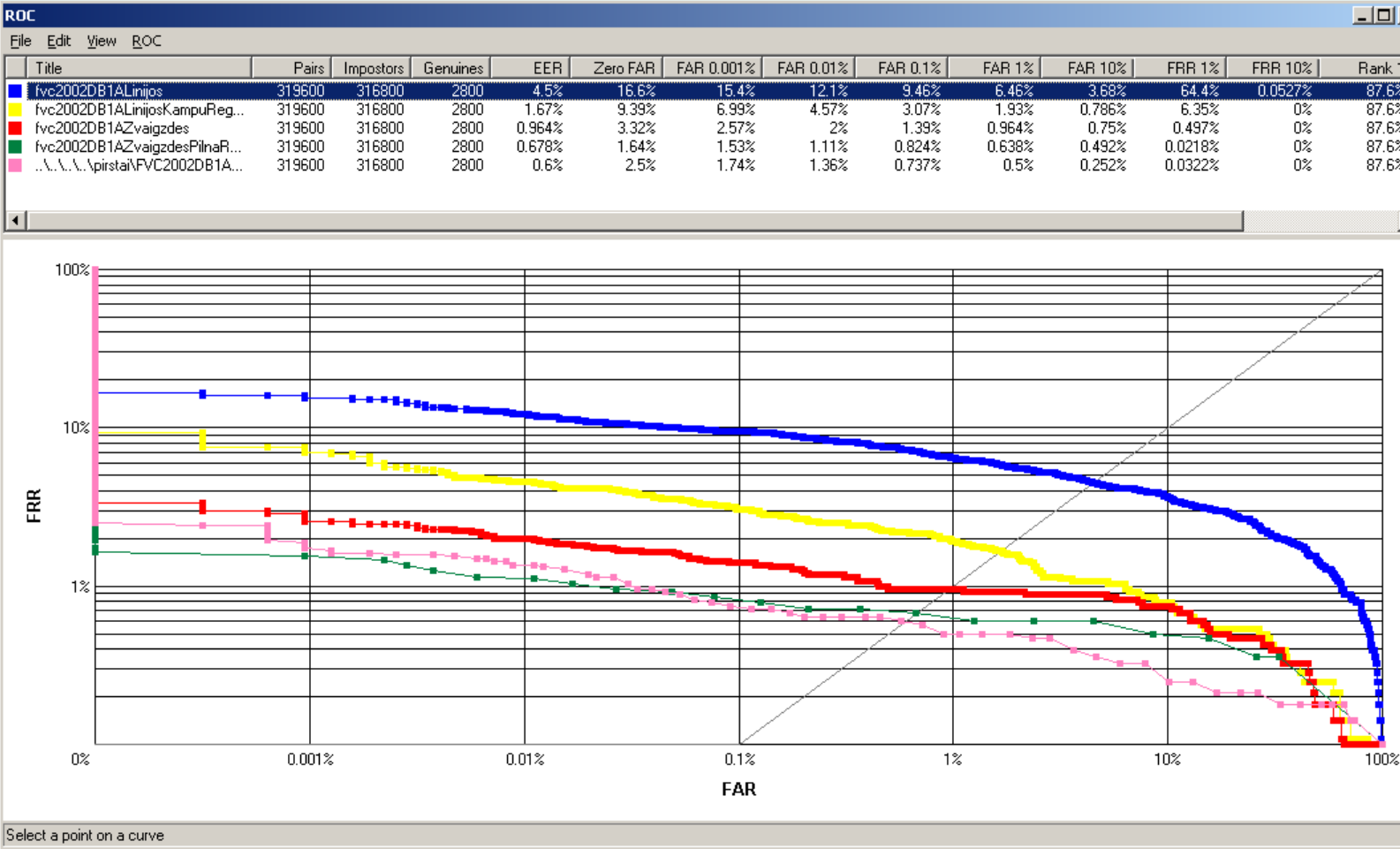
Tarkime, jei jūsų programos vardas ManoPirshtai.jar, naudojimo pavyzdys būtų

java -Xmx1024m -jar ManoPirshtai.jar FVC2002_DB1_A.info

- Jūsų programa turi kiekvieną fiksuotą pirshtai.info sąrašo elementą palyginti su pirshtai.info sąrašo elementais įrašytais didesnio numerio eilutėse. Gautus rezultatus įrašyti į *.roc formato failą. *FVC2002_DB1_A.info* pavyzdyje yra 800 įrašų. Vadinasi paskaičiuoto pavyzdžio *.roc faile turės būti $800 * 759/2$ palyginimų įrašų.
- Pilnai paruošta programa vykdymui kartu su programų pradiniais kodais turi būti pateikta iki š. m. gruodžio 15 d. (imtina).
- Vertinama 1 balo už atitinkančius reikalavimus algoritmo pateikimą.

Papildomai, priklausomai nuo algoritmu gautų rezultatų, skiriama iki penkių konkursinių egzamino balų, kurie atskirai prisideda prie jūsų pratybų ir teorijos balų.

6 paveikslėlis iliustruoja FVC2002DB1A pirštų duomenų skirtingų algoritmų palyginimo kreives. Kuo kreivė žemiau, tuo geresnė algoritmo kokybė. Mėlyna viršutinė kreivė iliustruoja palyginimo algoritmo, kuris remiasi tik linijų panašumais ir jų atiktimis, kokybę. Geltona kreivė iliustruoja tą patį algoritmą, kuris papildomai naudoja vaizdo registracijos kampo reikšmę. Raudona kreivė iliustruoja palyginimo algoritmą, kur atitikmenys ieškomi detalių centrų lygmenyje. Tamsiai žalia kreivė iliustruoja raudonos kreivės palyginimo algoritmą,



6 paveikslėlis: FVC2002DB1A pirštų duomenų bazės palyginimo algoritmų kokybės DET kreivės. Skirtingomis spalvomis pažymėti skirtingų algoritmų DET kreivės.

Home Page

Title Page



Page 21 of 21

Go Back

Full Screen

Close

Quit

kuriam papildomai pritaikyta pilna registracija pagal vaizdų registracijos posūkio kampą ir poslinkį. Paskutinė fuksinės spalvos kreivė gauta panaudojant profesionalią **Neurotechnology Megamatcher** pirštų atspaudų požymių išskyrimo ir palyginimo algoritmą. Ši kreivė gauta panaudojant programos versiją skirtą produkto įvertinimui (angl. *evaluation*).

Jūsų pateikto algoritmo kokybė bus laikoma tinkama, jei gauta DET kreivė neviršys "gel-tonos" DET kreivės.

Literatūra

- [1] **Handbook of Fingerprint Recognition**, D.Maltoni, D.Maio, A.K. Jain, S. Prabhakar
- [2] **Apžvalga**
- [3] **Jean Francois puslapis**
- [4] **Finger verification competition 2002**, <http://bias.csr.unibo.it/fvc2002/>
- [5] **Alexej Kočetkov, Pirštų atspaudų atpažinimo problemos**, magistro darbas, 2005 m., Vilnius