



VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
KOMPIUTERIJOS KATEDRA



Baigiamasis magistro darbas

Veidų segmentacijos algoritmai

Atliko: 2 kurso, 9 gr. studentė
Ieva Zareckaitė

Darbo vadovas:
Docentas Dr. Algirdas Bastys

Vilnius
2011

Turinys

Turinys	2
Anotacija	3
Summary	5
Įvadas	7
1. Veidų segmentacijos teorinė analizė	10
1.1. Veidų segmentacijos uždavinys ir problematika	10
1.2. Veidų segmentacijos algoritmų apžvalga	11
1.3. Teorinės apžvalgos išvados	15
1.4. „Boosting“ metodika grindžiama segmentacija	16
1.5. Naiviaisiais Bajeso klasifikatoriais grindžiama segmentacija	21
2. Veidų segmentacijos praktinė analizė	24
2.1. Veidų duomenų bazės	24
2.2. „Boosting“ metodikos praktinis nagrinėjimas	25
2.3. „Boosting“ praktinio nagrinėjimo išvados	30
2.4. Binarizacijos įtaka veidų segmentacijai DAB metodika	30
3. Veidų segmentacijos sistemos įgyvendinimas ir tyrimas	32
3.1. Mokymo ir testavimo veidų duomenų bazės	32
3.2. Pagrindinės realizuotos sistemos komponentės	33
3.2.1. DAB kaskadinis klasifikatorius	34
3.2.2. Naivusis Bajeso klasifikatorius	35
3.3. Realizacijos segmentacijos rezultatai	41
3.4. Realizacijos tobulinimo gairės	48
Išvados ir pasiūlymai	50
Literatūros sąrašas	52
Priedas Nr. 1	57

Anotacija

Magistro studijų tiriamojo darbo esmė – susipažinti su veidų biometrijos pagrindais, išanalizuoti egzistuojančias ir plačiausiai taikomas šiuolaikines automatines veidų segmentacijos skaitmeniniuose vaizduose metodikas bei vėliau pritaikyti minėtos analizės rezultatus savojo segmentavimo algoritmo įgyvendinimui. Pastarajam svarbiausiam uždaviniui įgyvendinti magistro tiriamojo darbo eigą buvo nuspręsta suskaidyti į tris nuoseklius etapus, kiekvienam iš jų iškeliant atitinkamus žemiau išvardintus tikslus:

- susipažinti su populiariausiomis šiuolaikinėmis veidų segmentacijos metodikomis ir atlikti geriausių iš jų teorinę bei praktinę analizę (2 semestras);
- vystyti tolimesnę išrinktų geriausių segmentacijos algoritmų, jų galimų kombinacijų bei savų idėjų praktinę analizę, įgyvendinant naująją segmentavimo metodiką (3 semestras);
- testuoti prasmingų (su realizacija susijusių) veidų segmentacijos algoritmų apžvalgą analizę, siekiant įsitikinti sprendimo naujoviškumu, palyginti ankstesniame etape įgyvendintos sistemos rezultatus su kitais mokslinėje literatūroje aprašytais metodais bei galimai algoritmiškai ir / arba programiškai optimizuoti sistemą tikslumo ir / arba našumo prasme (4 semestras).

Antrojo semestro tikslams įgyvendinti buvo atlikta su veidų automatine segmentacija susijusios mokslinės literatūros apžvalga. Šios teorinės apžvalgos rezultate buvo išrinkti geriausi tikslumo ir našumo santykio prasme „boosting“ algoritmai, kurie buvo išbandyti praktiškai. Pastarųjų algoritmų teoriniams įverčiams praktiškai pagrįsti / paneigti buvo naudojamos jų egzistuojančios realizacijos viešai prieinamose kompiuterinės regos bibliotekose. Taip pat praktiškai patikrinta binarizuotų (angl. *binarized*) vaizdų segmentavimo įprastomis pilkiems (angl. *grayscale*) vaizdams skirtomis metodikomis hipotezė.

Trečiojo semestro tikslams įgyvendinti buvo panaudoti antrajame semestre atliktos mokslinės literatūros apžvalgos ir analizės rezultate išrinkti geriausi išvaizda pagrįstų (angl. *appearance-based*) klasei priklausantys „boosting“ metodai. Klasikinės šių metodų realizacijos buvo algoritmiškai patobulintos ir sujungtos su naujuoju naiviuoju Bajeso klasifikatoriumi grindžiamu veidų segmentavimo algoritmu. Tokiu būdu siekiama išgauti tikslesnius priešakinių veidų lokalizacijos rezultatus. Praktiškai įgyvendinta sistema buvo tobulinama ir testuojama taikant atitinkamai mokymo ir testavimo duomenis iš FERET ir FRGC veidų duomenų bazių, kurios dažniausiai minimos ir naudojamos veidų išskyrimo ir palyginimo algoritmų mokslinių straipsnių autorių.

Ketvirtajame semestre buvo atlikta išsamesnė mokslinėje literatūroje aprašytų veidų segmentacijos algoritmų ir jų rezultatų analizė, kuri įgalino įsitikinti realizuotos sistemos unikalumu ir palyginti gautus tikslumo įverčius su kitais metodais bei jų kombinacijomis. Įgyvendinti universalūs realizuotos sistemos testavimo protokolai, skirti segmentacijos tikslumo įverčių taisyklingam apskaičiavimui vaizdams su daugiau nei vienu veidu. Naujieji testai buvo atliekami naudojant žinomiausias, būtent priešakinių veidų segmentacijos testavimui skirtas, duomenų bazes – MIT/CMU ir BioID. Testavimo duomenų bazių praplėtimas leido išvelgti sistemos privalumus ir trūkumus, praktinių eksperimentų eigoje pagerinti jos tikslumą bei pateikti išsamesnėms tikslumo / našumo tobulinimo gaires.

Taigi, esminis šio magistro baigiamojo darbo rezultatas – priešakinių veidų segmentacijos skaitmeniniuose vaizduose algoritmo praktinė realizacija, grindžiama studijų metu ir papildomai atliktos mokslinės literatūros apžvalgos bei analizės pasekoje įgytomis žiniomis. Įgyvendinta metodika testuota naudojant viešai prieinamas veidų populiariausias atpažinimo (FERET, FRGC) ir segmentacijos (MIT/CMU, BioID) duomenų bazes. Testų rezultatai demonstruoja klasikinių „boosting“ tipo veidų suradimo metodų rezultatų pagerinimo galimybes kombinuojant jas su pasiūlytomis vaizdų glodinimo ir klasifikavimo naiviaisiais Bajeso klasifikatoriais metodikomis. Gautieji segmentacijos rezultatai palyginti su kitų mokslinėje literatūroje aprašytų veidų suradimo algoritmų ir jų kombinacijų rodikliais. Įvertinti sistemos privalumai ir trūkumai.

Taip pat pateiktos gairės perspektyvioms tolimesnio tyrinėjimo kryptims, kurios galėtų potencialiai dar labiau pagerinti veidų lokalizacijos tikslumą ir / arba našumą, kuris yra itin svarbus realaus laiko verifikavimo ir identifikavimo sistemoms.

Summary

Face detection algorithms

The main aim of the current master work – to make oneself familiar with the basics of the face biometry science, analyze the most widely used automatic face detection (a. k. a. face segmentation) within digital images techniques and to apply results of the analysis to the implementation of the own detection algorithm. The process of achieving the latter main goal has been further divided into three consequent stages. Each stage has been (respectively) assigned a task from the list below:

- make oneself familiar with nowadays widely used face detection techniques and analyze the best ones of those in both theory and practice (2nd term);
- proceed practical analysis of previously chosen best detection techniques and their combinations along with own ideas while implementing oneself face detection method (3rd term);
- extend general overview and theoretical analysis of scientific literature to prove the uniqueness of proposed solution as well as to compare implemented system with analogue face segmentation methods published by other researchers; attempt to algorithmically and / or programmatically optimize the algorithm for higher accuracy and / or performance rates (4th term).

The survey on different face segmentation related articles has been done to accomplish the tasks assigned to the 2nd term. Results of this survey enabled to choose the best algorithms in terms of general detection and speed ratio – “*boosting*” methods. These were further tested in practice. To approve / negotiate the estimates of the chosen algorithms that were claimed in theory (i. e. within articles), existing implementations from the currently publicly available computer vision libraries were mostly used. Also an additional practical work has been done to test some primary hypothesis regarding binarized image classification using the same grayscale images detection techniques.

The 2nd term research survey conclusions have been used to select the best methods to fulfill the tasks assigned to the 3rd term. Sophisticated versions of chosen classical appearance-based “*boosting*” algorithms were combined with own Naïve Bayesian classifiers based segmentation method to achieve more precise frontal faces localization results. Practically implemented system has been tuned up and tested on the training and testing sets (respectively) of the FERET and FRGC face databases, which are most usually referenced by face extraction and matching algorithms articles authors.

An extended overview and more in-depth analysis of state of the art face detection scientific papers has been accomplished within the recent (4th) term. The main aim of it was to convince oneself with the uniqueness of the approach proposed within the master work and compare achieved segmentation rates with other existing methods and their combinations. Secondly, implemented face segmentation framework has been evaluated on the most popular up to date MIT/CMU and BioID frontal face detection evaluation databases. Additional practical work has been done to implemented correct evaluation protocols for multi-face image cases, tune the accuracy and performance of the system also to propose possible further optimization guidelines in terms of speed and / or accuracy.

Thus, the main result of this master work – frontal face detection within digital images algorithm implementation, built on the basis of knowledge obtained while studies and additional scientific papers survey and analysis. Implemented system has been tested on the publicly available well-known face recognition (FERET, FRGC) and face segmentation (MIT/CMU, BioID) evaluation databases. Testing results demonstrate possibilities to improve accuracy of the classical “*boosting*” based face detection algorithms while combining the former with proposed fast image blurring and naïve Bayesian classification methods. Aforementioned results have been compared with other published face segmentation methods and their combination rates. Moreover, the guidelines for further perspective research directions are provided, since those are

believed to improve faces localization accuracy and / or performance. The latter is very important for a real-time face verification and identification systems.

Ivadas

Šiais laikais biometriniai atpažinimo algoritmai yra plačiai taikomi prieigos prie tam tikros informacijos saugumui užtikrinti. Trijų populiariausių šiems tikslams naudojamų subjektų biometrinių požymių tarpe yra veidai. Šiuolaikinių veidų atpažinimo sistemų tikslumas yra prastesnis nei kitų dviejų – pirštų atspaudų ar akies rainelių. Tačiau veidai įgyja vis didesnį populiarumą dėl savo natūralumo, t. y. itin nedidelio kiekio pastangų, kurias reikia sutelkti vartotojui, bendradarbiaujančiam su tokia sistema. Be to, skirtingai nuo minėtų dviejų biometrijų, veidų atpažinimas gali būti taikomas netgi įslaptintuose sistemose, pvz., oro uostuose identifikuojant sistemoje registruotus nusikaltėlius / teroristus. Be to, biometrinė veidų atpažinimo sistema nereikalauja tokių specifinių brangiai kainuojančių prietaisų, kaip pirštų atspaudų ar rainelių skeneriai. Veidų sistemai užtenka vos vienos ar kelių internetinių vaizdo kamerų (angl. *web camera*). Tačiau būtina paminėti, kad šiame darbe toliau išvardintos veidų segmentavimo skaitmeniniuose vaizduose problemos parodo, kad geresnės kokybės iš kameros gaunami vaizdai leidžia tikėtis geresnių veidų segmentavimo (angl. *face detection*), o kartų ir veidų atpažinimo (angl. *face recognition*) sistemos rezultatų.

Veidų atpažinimo biometrinės sistemos rezultatai priklauso nuo kelių esminių jos komponentų – požymių išskyrimo (angl. *feature extraction*) į biometrinius šablonus (angl. *biometric templates*) bei gautųjų šablonų palyginimo (angl. *matching*) algoritmu, kur antrojo rezultatai priklauso nuo pirmojo. Nesant surastiems veidams ir atitinkamai išskirtiems požymiams, sistema toliau tiesiog neturi ką lyginti, t. y. negali spręsti individų **verifikavimo** (1 : 1 palyginimo) arba **identifikavimo** (1 : N palyginimo) problemų. Taigi, bandant nuosekliai kurti veidų biometrinę atpažinimo sistemą, iš karto susiduriama su pirmuoju klausimu: „Kaip surasti (segmentuoti) veidą skaitmeniniame vaizde?“. Norint atsakyti į šį klausimą, reikia išspręsti labai sudėtingą problemą, kadangi veidų išskyrimas vaizduose itin priklauso nuo sąlygų, kuriomis šie vaizdai buvo gauti. Įtaka didelei intraklasinei veidų skaitmeninių atvaizdų variacijai daro net tik vadinamieji išoriniai veiksniai (angl. *extrinsic factors*)

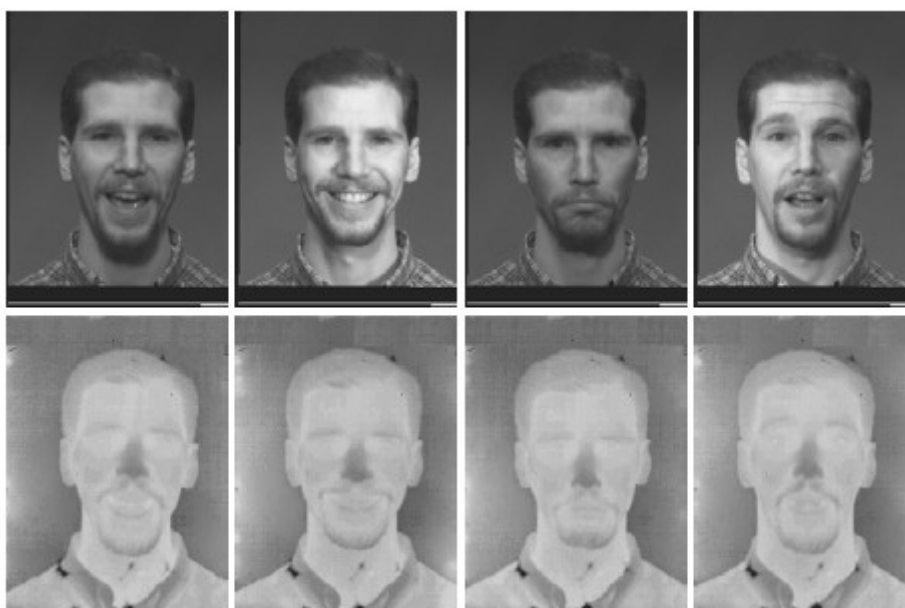
- filmavimo / fotografavimo aplinkos apšvietimo, kitimas,
- filmavimo / fotografavimo kampo kitimas,
- veidą užstojančių objektų atsiradimas,

tačiau ir vidiniai veiksniai (angl. *intrinsic factors*)

- filmuojamo subjekto šukuosenos ir galvos apdangalų (kepurė) kaita,
- veido išraiškos kitimas,
- amžiaus kaita,
- akiniai,
- barzda ir ūsai.

Taigi, esminė problema, su kuria susiduriama įgyvendinant automatizuotas veidų atpažinimo sistemas – itin sunkus mimikų, amžiaus kitimo (brendimo, senėjimo), posūkių kameros atžvilgiu (dvimatėje ir trimatėje erdvėje) tolerancijos autonomine programa uždavinys. Biometrinė veidų atpažinimo sistema, kuri būtų atspari visiems išvardintiems faktoriams praktiškai neegzistuoja. Ji egzistuoja tik kaip teorinis idealas, kurio reikia siekti, ir yra siekiama daug metų. Visos šiuolaikinės veidų atpažinimo biometrinės sistemos veikia tik tam tikromis mažiau arba labiau ribotomis sąlygomis, kurios deklaruojamos minėtų kintančių faktorių konstantiniais arba intervaliniais apribojimais (angl. *constraints*). Tačiau visą laiką stengiamasi tobulinti sistemas, siekiant sumažinti jų taikymo sąlygoms keliamus apribojimus. Dažniausiai tai daroma algoritmu, kartais – kitos techninės įrangos panaudojimo dėka. Pvz., apšvietimo sąlygų kaitos problema gali būti sprendžiama veidų gavimui naudojant infraraudonųjų spindulių kameras (žr. pav. 1). Tačiau šiuo metu jų taikymas realiose sistemose nėra itin paplitęs. Be to, visų anksčiau įvardintų problemų šios kameros neeliminuoja. Taigi, efektyvus veidų automatinio išskyrimo skaitmeniniuose vaizduose algoritmu kūrimas yra aktualus ir šiais laikais. Kadangi kalbama būtent apie automatinio segmentavimo algoritmo kūrimą, tai jo efektyvumui keliami

reikalavimai ne tik algoritmo tikslumo, tačiau ir algoritmo našumo atžvilgiu (t. y. algoritmas turi pakankamai tiksliai segmentuoti veidus realiu laiku).



Pav. 1. „Equinox IR“ duomenų bazės vaizdų pavyzdžiai. Viršuje pateikti įvairiomis apšvietimo sąlygomis gauti skirtingų veido išraiškų pilkųjų (angl. *grayscale*) vaizdų pavyzdžiai, apačioje – atitinkami ilgųjų bangų infraraudonieji vaizdai. [LJ04]

Magistro studijų tiriamojo darbo tikslas – priešakinių (angl. *frontal*) veidų segmentacijos skaitmeniniuose vaizduose algoritmo sukūrimas.

Šiam tikslui pasiekti, visų pirma buvo būtina apžvelgti šiai dienai plačiausiai taikomas veidų segmentavimo (angl. *face detection*) metodikas. Pažangiausia iš jų – „*boosting*“ technika grindžiamas *AdaBoost* mokymo algoritmas kartu su galimomis jo modifikacijomis (*Discrete AdaBoost*, *Gentle AdaBoost*, *Real AdaBoost*, *LogitBoost*, toliau atitinkamai – *DAB*, *GAB*, *RAB*, *LAB*) buvo išanalizuoti nuodugniau tiek teoriškai, tiek praktiškai. Praktinei analizei buvo pasitelkta *OpenCV* [OB09, OHT09] kompiuterines regos biblioteka bei asmenines realizuotos programos. Minėta analizė įgalino sudaryti gaires tolimesniems galimoms algoritmo modifikacijoms, patobulinimams bei kombinacijoms su kitomis segmentavimo technikomis. Visi teorinės analizės ir praktinių bandymų rezultatai pateikti žemiau išdėstytuose skyriuose.

Veidų segmentavimo algoritmo praktinio įgyvendinimo etape buvo pritaikytas analizės rezultate vienas geriausių išrinktas klasikinis *DAB* algoritmas, kuris yra plačiai taikomas įvairių klasių (t.y. ne tik veidų) objektams skaitmeniniuose vaizduose lokalizuoti. Kaip ir tikėtasi, *DAB* algoritmas (kuris buvo papildomai modifikuotas išplečiant mokymo fazėje naudojamu požymių įvairovę) iš esmės pateisino lūkesčius teisingai surastų ir nesurastų veidų santykio įverčio prasme. Tačiau, viena esminių *DAB* problemų – netiksli surastų veidų lokalizacija tikrosios pozicijos atžvilgiu, kas neigiamai įtakoja veidų verifikavimo / identifikavimo sistemos išskyrimo (angl. *extraction*), o vėliau – palyginimo (angl. *matching*) etapo rezultatus. Taip pat išlieka *DAB* neteisingai surastų netikrų veidų (angl. *false faces*) eliminavimo segmentacijos etape problematika (siekiant išvengti biometrinės sistemos resursų išnaudojimą jų apdorojimui kituose etapuose). Įvardintoms problemoms spręsti, *DAB* klasifikatorius buvo sujungtas su naujuoju segmentacijos algoritmu, kuris apdoroja *DAB* grąžintas veidų sritis ir jų aplinkas, pritaikydamas žemiau aprašytą greitąjį eksponentinį glodųjį filtrą ir naiviųjų Bajeso (angl. *Naive Bayesian*, toliau – *NB*) klasifikatorių metodologiją. Minėtų metodikų ir jų kombinacijos praktinės realizacijos aprašymas pateikti šio darbo tolimesniuose skyriuose. Taip pat pateikiama praktinės realizacijos eigoje atlikta papildoma *NB* klasifikatorių ir jų kombinacijų su kitais algoritmais taikymo šiuolaikinėse veidų biometrijos sistemose mokslinės literatūros apžvalga.

Igyvendintai veidų segmentacijos sistemai mokinti, o vėliau – tikslumui vertinti, pirma pasitelktos viešai prieinamų bene populiariausių veidų atpažinimo FERET ir FRGC duomenų bazių (žr. priedą Nr. 1) mokymo arba etalonų ir kandidatų vaizdų atitinkami poaibiai. Tačiau paskutiniame semestre atliktos išsamesnės analogiškų veidų segmentacijos algoritmų straipsnių analizės pasekoje išaiškėjo, kad dauguma autorių šias atpažinimo duomenų bazes naudoja skirtingais būdais ir proporcijomis segmentacijos algoritmams mokinti, derinti ir vertinti. Taigi, sistemos tikslumas buvo papildomai įvertintas ant dažniausiai minėtuose straipsniuose minimų kur kas mažesnių MIT/CMU ir BioID duomenų bazių (žr. priedą Nr. 1), kurios yra skirtos tik veidų suradimui vaizduose testuoti. Pastarieji testai įgalino realizuoto metodo palyginimą su kitų autorių darbais. Visų įvardintų mokymo ir testavimo duomenų bazių aprašai, įgyvendintos metodikos testavimo rezultatai, pastarųjų analizė ir palyginimas pateikti toliau.

1. Veidų segmentacijos teorinė analizė

Šiame skyriuje toliau pateikiami veidų segmentacijos uždavinio apibrėžimas ir jo problematika (žr. skyrius 1.1.). Trumpa labiausiai žinomų veidų segmentacijai dvimačiuose skaitmeniniuose vaizduose taikomų išvaizda grindžiamų klasės metodikų apžvalga ir jos išvados yra pateiktos darbo skyriuose 1.2 ir 1.3. Skyriuje 1.4 ir 1.5. aprašyta išsamesnė informacija apie su šiuo darbo praktine realizacija susijusius „boosting“ ir naiviojo Bajeso (NB) klasifikatorių metodikas.

1.1. Veidų segmentacijos uždavinys ir problematika

Veidų suradimas yra pirmasis uždavinys veido atpažinimo biometrinei sistemai atlikti. Sistemos gebėjimas sėkmingai ir greitai segmentuoti veidus tiesiogiai įtakoja visos sistemos patikimumą ir našumą. Tai reiškia, kad idealiu atveju biometrinės sistemos segmentavimo algoritmas turi surasti bet kokiame skaitmeniniame vaizde arba jų sekoje visus esančius veidus. Tai yra idealus, bet toli gražus ne realus atvejis, kadangi veidų segmentacija skaitmeniniuose vaizduose yra itin problematiškas uždavinys, įtakojamas tokių žemiau išdėstytų faktorių:

- subjekto pozos (galvos posūkių dvimatėje (angl. *in-plane rotations*) ir trimatėje erdvėje (angl. *out-of-plane rotations*));
- subjekto veido išraiškos (angl. *expression*);
- papildomos subjekto atributikos (pvz., kepurės, akinių) buvimo / nebuvimo,
- subjekto šukuosenos (taip pat plaukų, dengiančių veidą),
- subjekto barzdos ir ūsų,
- subjekto amžiaus kaitos,
- filmavimo aplinkos apšvietimo sąlygų,
- filmavimo kampo,
- persidengimo (angl. *occlusion*) su kitais objektais (t.y. matoma tik dalis veido) (žr. pav. 2 ir pav. 3).



Pav. 2. Intraklasinės vieno subjekto veidų variacijos pavyzdys (posūkių, išraiškos, apšvietimo kaita bei persidengimai su kitais objektais). [LJ04]

Visos šios problemos atsiranda dėl to, kad ieškant veidų skaitmeniniuose vaizduose, realiai bandoma išspręsti trimačio objekto, kuris yra įtakojamas apšvietimo, išraiškos ir visų kitų minėtų sąlygų, paieškos jo dvimačiame atvaizde (projekcijoje) uždavinį. Itin didelės veidų intraklasinės variacijos rezultate kartais skirtumas tarp tuo paties subjekto atvaizdų pasikeitus apšvietimui ir filmavimo kampui būna didesnis, nei pasikeitus pačiam subjektui.

Minėtai veidų intraklasei variacijai mažinti ir veidų bei neveidų interklasei variacijai didinti yra pasitelkiamos įvairios geometrinės ir fotometrinės normalizacijos. Taip pat veidų klasė būna skaidoma į poklasius, kiekvienam iš jų atskirai sprendžiant veidų segmentavimo problemą.



Pav. 3. Intraklasinės vieno subjekto veidų variacijos pavyzdys (veido išraiškos kaita).
Pavyzdžiai paimti iš „JAFFE“ veidų duomenų bazės (žr. priedą Nr. 1, [JAFFE]).

Pastarasis sprendimas yra gana populiarus dėl to, kad dažniausiai veidų suradimas yra traktuojamas kaip skaitmeninio vaizdo sričių klasifikavimas į dvi klases: veidus ir neveidus. Tokia strategija yra taikoma *išvaizda grindžiamuose algoritmuose* (angl. *Appearance-Based Algorithms*, toliau – *ABA*). Tokie algoritmai, tarp jų – šiame darbe toliau išnagrinėtas „*boosting*“: metodika grindžiamas diskretus *AdaBoost* – pasiteisino kaip pažangiausi veidų segmentacijos algoritmai. Tačiau egzistuoja ir kitų metodikų, kurios yra grindžiamos odos spalva (spalvotuose vaizduose), judėjimu (angl. *motion*), veido / galvos forma ir kontūrais ir t. t.. Trumpa veidų segmentacijai dvimačiuose skaitmeniniuose vaizduose taikomų metodikų apžvalga yra pateikta kitame skyriuje.

1.2. Veidų segmentacijos algoritmų apžvalga

Šiame skyriuje trumpai apžvelgtos šiai dienai populiariausios veidų segmentacijai taikomos technikos..

Dažniausiai veidų segmentacijai taikomi metodai naudoja tokius algoritmus:

- odos segmentavimo (taikomi spalvotiems vaizdams);
- principinių komponentų analizę (angl. *Principal Component Analysis*, toliau – *PCA*);
- neuroninius tinklus (angl. *Neural Networks*, toliau – *NN*);
- atraminių vektorių mašinas (angl. *Support Vector Machines*, toliau – *SVM*);
- Bajeso diskriminantinius požymius (angl. *Bayesian Discriminating Features*, toliau – *BDF*);
- „*Boosting*“ grindžiamas *AdaBoost* mokymo metodikas (su klasifikacijos ir regresijos medžiais (angl. *Classification and Regression Trees*, toliau – *CART*) arba kelmiais (angl. *stumps*)), tarp jų:
 1. *Discrete AdaBoost* (toliau – *DAB*),
 2. *Gentle AdaBoost* (toliau – *GAB*),
 3. *Real AdaBost* (toliau – *RAB*),
 4. *LogitBoost* (toliau – *LAB*).
 5. *FloatBoost* (dvigubo praėjimo „*boosting*“ algoritmas, toliau – *FAB*)

Šie algoritmai gali būti kombinuojami vienas su kitu (pvz, *PCA* su *BDF*). Tokių pavyzdžių bus matyti ir toliau išdėstytoje trumpoje šių algoritmų bei jų modifikacijų mokslinių straipsnių apžvalgoje.

Veidų segmentavimo algoritmai naudojantys **odos segmentaciją** taikomi tik spalvotiems vaizdams ar jų sekoms. Šie algoritmai yra grindžiami teiginiu, kad žmogaus oda turi savitą pasiskirstymą, skirtingą nuo daugumos neveidų tipo objektų. Dažniausiai odos spalva grindžiamą veidų segmentaciją sudaro tokie žingsniai:

1. veidų sričių kandidatų suradimas;
2. pirmajame žingsnyje gautų sričių apjungimas (angl. *merging*);

3. klaidingų segmentacijų eliminavimas.

Paskutinis žingsnis yra neprivalomas ir naudojamas tik labiau pažengusiuose tokio tipo algoritmuose.

Veido odos spalva grindžiamuose algoritmuose dažniausiai iš turimų apmokymo vaizdų apskaičiuojamas $p(\text{spalva} | \text{veidas})$ tipo tikimybinis modelis. Ši tikimybė apskaičiuojama tam tikroje spalvų erdvėje (pvz., normalizuotoje *Red-Green-Blue* (toliau – *RGB*), *Hue-Saturation-Value* (toliau – *HSV*), *YCbCr*) [Mar02, ZSQ99]. Žemiau patektame pav. 4 pateiktas segmentacijos pavyzdys, kai taškelis priskiriamas veido sričiai, jeigu *HSV* erdvėje $p(H | \text{veidas}) > 0.3$, o *S* ir *V* komponentų reikšmės pakliūna į tam tikrus leidžiamų reikšmių intervalų režius.



Pav. 4. Veido odos spalvos segmentacija. Kairėje pateiktas originalus paveikslukas. Dešinėje – odos spalvos segmentacijos rezultate išgautos potencialios veidų sritys. [LJ04]

Veido spalva grindžiama segmentacija yra naši. Tačiau veidų segmentacijos prasme nėra stipri, kadangi segmentacijos kriterijus, kuris remiasi vien apribotais taškelių (angl. *pixel*) spalviniams atributams, yra nepakankamas geram veidų suradimo tikslumui pasiekti. Kriterijaus silpnumą įtakoja didelė veidų odos spalvinė variacija dėl filmavimo sąlygų skirtumo (apšvietimo, šešėlių) veiksnių bei etninių grupių skirtumo. Spalvinė veidų informacija kartais taikoma kaip papildomas kriterijus veidų sekimui vaizdų sekoje (angl. *tracking*). Be to, dauguma šiuolaikinių veidų segmentavimo algoritmų dirba tik su pilkais paveikslukas (angl. *grayscale*), t. y. narinėja tik vieno kanalo (pilkumo) lygį, o ne spalvinę kelių kanalų informaciją. Toliau aprašomi pilkus (angl. *grayscale*) paveikslus nagrinėjantys veidų segmentacijos algoritmai.

M. A. Turk ir A. P. Pentland [TP91] aprašė segmentavimo sistemą, kuri yra **grindžiama principinių komponentų analize (PCA)** poerdvio arba tikrinio veido (angl. *eigenface*) reprezentacija. Baziniame *PCA* metode yra naudojama tik *PCA* poerdvio tikimybė, tačiau B. Moghaddam ir A. Pentland [MP97] siūlo taip pat naudoti ortogonalus papildinio poerdvio tikimybę. Naudojant tokią sistemą, tikimybė skaitmeninio vaizdo erdvėje (t. y. dviejų poerdvių sąjungoje) yra modeliuojama kaip dviejų minėtų tikimybinių dydžių sandauga, kuri įgalina tikslesnę veidų segmentaciją. K. -K. Sung ir T. Poggio [SP98] visų pirma suskaido vaizdo erdvę į kelias veidų ir neveidų grupes (angl. *cluster*) ir vėliau atlieka kiekvienos iš jų dekompoziciją į *PCA* ir tuščius poerdvius. Toliau statistiniams požymiams išgauti yra taikomas *Bajeso* įvertis (angl. *Bayesian estimation*). H. A. Rowley ir kt. [RBK98] sukurta sistema naudoja sujungtus **neuroninius tinklus (NN)**. Įeities skaitmeninis vaizdas po ilgo išankstinio apdorojimo proceso (angl. *preprocessing*) yra tikrinamas naudojant slenkančio lango (angl. *sliding window*) strategiją (analogiškai *AdaBoost* daugiamastelinio segmentavimo strategijai). Tai vienas populiariausių ir geriausių tikslumo prasme veidų segmentavimo algoritmų. Jis atsilieka nuo P. Viola ir M. Jones [VJ01b] *AdaBoost* metodo tik savo žemais našumo rodikliais. E. Osuna ir kt. [OFG97] apmokino netiesinę **atraminių vektorių mašiną (SVM)** veidų ir neveidų šablonams klasifikuoti. M. -H. Yang ir kt. [YRA00] naudoja išsklaidyto vėtymo tinklo *SNoW* (angl. *Sparse Network of Winnows*, toliau – *SNoW*) apmokymo architektūrą veidams surasti. Visose išvardintose sistemose yra iteraciškai taikomas pakartotinių imčių (angl. *bootstrap*) statistinis algoritmas prasingiems neveidams iš veidų neturinčių vaizdų išrinkti (tolimesniam veidų segmentatoriaus (angl. *face detector*) mokymui).

H. Schneiderman ir T. Kanade [SK00] naudoja daugiarezoliucinio pobūdžio informaciją skirtingiems banginės (angl. *wavelet*) transformacijos lygiams. **AdaBoost** algoritmo pagalba iš veidų ir neveidų mokymo aibės vaizdų gautų histogramų sandaugų statistikų jie sukonstravo netiesinį klasifikatorių [Sch00]. Šis algoritmas yra efektyvus segmentacijos atžvilgiu, tačiau itin sudėtingas ir brangus skaičiavimo prasme. Nors algoritmas kelis kartus tobulintas, tačiau jo veikimo laikas išliko apie 5 sekundes net nedideliame 240 x 256 taškelių (pikselių) dydžio paveiksliui (naudojant kompiuterį su Pentium II 450 MHz procesoriumi).

P. Viola ir M. Jones [VJ01a, VJ01b] padarė tikrą proveržį veidų ir kitų objektų segmentacijos skaitmeniniuose vaizduose srityje. Naudojami **AdaBoost** apmokymo metodiką, jie sukūrė itin greitą ir tikslų netiesinį veidų kaskadinį klasifikatorių. Ankstesni ne tik veidų bet ir vaizdų klasifikacijai spręsti skirti **AdaBoost** algoritmo straipsniai pateikti [TV00] ir [Sch00]. **AdaBoost** bendru atveju sprendžia tokias tris fundamentalias problemas:

- godžia (angl. *greedy*) metodologija atrenka efektyvius požymius iš itin gausios požymių aibės;
- sukonstruoja silpnuosius klasifikatorius (angl. *weak classifiers*), kiekvienas iš kurių yra grindžiamas pasirinktu(-ais) požymiu(-iais);
- sukomponuoja silpnuosius klasifikatorius į vieną stiprųjį klasifikatorių (angl. *strong classifier*) (angliškai toks silpnųjų klasifikatorių sujungimas duodantis stipresnį rezultatą vadinamas „**boosting**“).

Silpnieji klasifikatoriai yra grindžiami įprastais Haar bangelės (angl. *wavelet*) tipo požymiais (žr. sk. 1.3). Vienas iš P. Viola ir M. Jones esminių nuopelnų yra sparti integralinio vaizdo schema, pasiūlyta efektyviam tokių požymių apskaičiavimui esant skirtingiems masteliams ir pozicijoms paveikslo viduje. Tai pirmasis sprendimas, padaręs labai didelę teigiamą įtaką našumui. Taip pat nuoseklus „nuo paprasčiausios iki sudėtingiausios“ problemų (angl. „*simple-to-complex*“) sprendžiančių atitinkamų klasifikatorių sekos apjungimas į kaskadą daro skaičiavimus dar spartesnius. Tai antrasis svarbus sprendimas našumui pagerinti. Šių autorių sistema buvo pirmoji, kuri veikė realiu laiku, t. y. 14 kadru (320 x 240 taškelių dydžio) per sekundę [VJ01b].

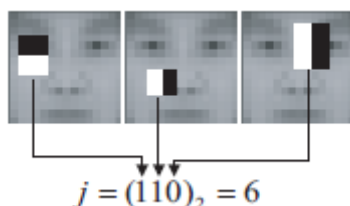
C. Liu [Liu03] savo straipsnyje pristatė **Bajeso diskriminantinių požymių metodą (BDF)**. Įeties skaitmeninis vaizdas, jo Haar bangelės (angl. *wavelet*) reprezentacija ir jo amplitudės projekcija yra sujungiami į išplėstinį 768 dimensijų įeties vektorių. Algoritme daroma prielaida, kad šie veidų vektoriai turi (vieną) daugiavariacinę normalų pasiskirstymą. Tuomet atliekamas tiesinis dimensijų mažinimas *PCA* modeliams išgauti. Tikimybinis įvertis yra išgaunamas iš *PCA* ir jo „liekanų“, taikant Bajeso technikas [MP97]. Neveidų klasifikatorius yra sudaromas analogiškai. Klasifikavimo sprendimas, ar tai veidas, ar neveidas yra daromas remiantis dviem tankio įverčiais. Autoriai teigia, kad šis algoritmas savo tikslumu artimas H. Schneiderman ir T. Kanade [SK00] metodikai. Įdomus ir tas faktas, kad tokie geri rezultatai yra pasiekiami taikant tik vieną Gauso pasiskirstymą veidams ir neveidams, bei apmokinat *BDF* ant santykinai mažos mokymo duomenų aibės: 600 FERET duomenų bazės (žr. priedą Nr. 1) veidų ir 9 natūralių neveidų vaizdų. Apmokintas klasifikatorius gerai veikia ant testavimo duomenų. Tačiau šiam metodui, kaip ir daugeliui kitų išvardintų anksčiau, suvokti, reikia nuodugnesnės analizės, kuri šioje apžvalgoje nėra pateikiama.

Daugelio aplikacijų atveju svarbi yra ir galimybė segmentuoti ne vien priešakinius, bet ir pasuktus veidus. Vienas iš protingų būdų ne itin sudėtingai išspręsti pasuktų veidų suradimo vaizduose problemą – ribotu regėjimo lauku (posūkio kampų) grindžiama strategija [PMS94], kurioje yra kuriami keli veidų klasifikavimo modeliai, t. y. po vieną kiekvienam tam tikram ribotam veido posūkio kampui ar intervalui. Tokiu būdu vėl gi išvengiama sudėtingo trimačio veido modelio sudarymo. J. Feraud ir kt. [FBC00] adaptavo šią strategiją pasuktų veidų segmentacijai ir naudoja penkių klasifikatorių masyvą kiekvienam iš analizuojamų vaizdų. Kai kurių vėlesnių šia strategija grindžiamų straipsnių autoriai naudojo papildomas medžių struktūras, leidžiančias išvengti visų masyvo klasifikatorių paleidimo, o vertinti tik dalį jų, priklausomai nuo veido posūkio prognozių paveiksle. S. Gong ir kt. [CMC96] studijavo veidų trajektorijas (kai šie sukasi) tiesinėse *PCA* požymių erdvėse ir naudojo branduolines **atramos**

vektorių metodikas (angl. *Kernel Support Vector Mahines*) daugiapoziciniam veidų segmentavimui ir pačios pozos įvertinimui. [LGL00, NG99]. J. Huang ir kt. [HSW98] naudoja *SVM* veido pozicijai (pozai) nustatyti. Jau ne kartą minėti H. Schneiderman ir T. Kanade [SK00] taip pat naudoja penkių veidų segmentatorių (klasifikatorių) masyvą skaidytų ribotais posūkio intervalais grindžiamoje sistemoje (angl. *view-based framework*).

S. Z. Li ir kt. [LZS+02, LZZ+02, LZ04] pristatė įvairius posūkius toleruojančią (angl. *multiview*) veidų segmentavimo sistemą, išplėsdamas kitų straipsnių darbus [SK00, VJ01a, VJ01b]. Vėliau šis autorius pasiūlė naują *AdaBoost* algoritmo modifikaciją – ***Floating Boost*** (arba ***FloatBoost***, toliau – *FAB*) algoritmą, kuris įdiegė slankios požymių paieškos (angl. *floating search*) principą į originalius *DAB* bei *RAB* algoritmus. Grįžtantis (angl. *backtrack*) naujojo algoritmo mechanizmas leidžia šalinti silpnuosius klasifikatorius (angl. *weak classifiers*) iš stipriųjų klasifikatorių (angl. *strong classifiers*), kai silpnieji yra neefektyvus klasifikavimo klaidos dažnio (angl. *error rate*) prasme. Tai dar labiau sumažina silpnųjų klasifikatorių skaičių stipriajame klasifikatoriuje. Taip pat buvo pasiūlyta išplėsta pačių Haar požymių aibė susidorojimui su veidų trimačiais posūkiiais (angl. *out-of-plane rotations*) į kairę ir į dešinę (tokio tipo požymiai pritaikyti ir šiame darbe). Kartu aprašyta nuo „paprasto prie sudėtingo“ arba nuo „prasto prie gero“ (angl. „*coarse-to-fine*“) tipo architektūros segmentacijos piramidė, skirta pasuktų veidų greitajam segmentavimui. Ji veikia apie 200 ms ant 320 x 240 taškelių vaizdų (naudojant kompiuterį su Pentium III CPU 700MHz procesoriumi). *FAB* apmokymo rezultate gautas klasifikatorius dažniausiai yra spartesnis ir tikslesnis nei *DAB* gautas klasifikatorius. Tačiau kaina už tokį pagerėjimą yra didelė – ir taip nemažas *AdaBoost* algoritmo mokymosi laikas padvigubėja. Tai apsunkina *FAB* algoritmo testavimą, tobulinimą ir, atitinkamai, jo taikymą.

R. Lienhart ir kt. [LKP02] savo darbe naudoja išplėstą aibę pasuktų Haar požymių dirbdamas su veidais, kurie gali būti pasukti plokštumoje (angl. *in-plane rotated*). Taip pat šis autorius vietoje *DAB* naudoja *GAB* [FS97] algoritmą su kelių viršūnių klasifikacijos ir regresijos *CART* medžias baziniams silpniesiems klasifikatoriams. Straipsnyje pateikti rezultatai parodo, kad ši kombinacija yra geresnė lyginant su įprasta *DAB* metodika, naudojančia tik vienos viršūnės *CART* medžių silpnuosius klasifikatorius, t.y. kelmus (ang. *stumps*). Buvo ir kitų silpnųjų klasifikatorių modifikacijų, pvz., T. Mita ir kt. [MKH05] savo straipsnyje siūlė panaudoti tam tikrų požymių koreliacijos, t.y. pasitaikymo veido vaizde kartu (angl. *co-occurrence*) statistiką ir naudoti jungtinius silpnuosius požymius (žr. pav. 5). Autoriai deklaruoja rezultatus, kuriuose kelių požymių jungtis į silpnuosius klasifikatorius duoda geresnį tikslumą nei naudojant silpnuosius pavienių požymių klasifikatorius. Požymių bendras skaičius būna apytiksliai lygus, tačiau tikslumas geresnis. Šio algoritmo apmokymo fazė taip pat gali būti ilgesnė nei bazinio *DAB* algoritmo, kadangi reikia naudoti nuoseklų išrinkimą į priekį (angl. *Sequential Forward Selection*, toliau – *SFS*) veidų vaizduose kartu pasitaikantiems požymiams atrinkti. Taip pat reikalingi papildomi testai optimaliam jungiamų požymių skaičiui F nustatyti, kadangi autoriai teigia, kad jungiant nuo 3 iki 6 požymių gaunamas bendros klaidos sumažėjimas net 37,4% (lyginant su pavieniais požymiais). Tačiau naudojant jau 9 požymius, klasifikatoriaus rezultatai ant testavimo duomenų prastėja., t.y. įvyksta persimokymas (angl. *overtraining*) – pernelyg didelė adaptacija prie apmokymo duomenų (angl. *overfitting*) neigiamai įtakojanti rezultatus ant duomenų, kurie nebuvo įtraukti į mokymo duomenų aibę. Koks būtų optimalus jungiamų požymių skaičius – vienas šio metodo klausimų, kurie dažniausiai sprendžiami empiriškai



Pav. 5. Jungtinis požymis sudarytas iš trijų dažniausiai kartų pasitaikančių veido vaizduose požymių. Čia j —požymio atsakas, apskaičiuojamas remiantis jungiamų požymių binariniais atsakais [MKH05].

Yra ir kitų mažiau žinomų algoritmų, kurie, pvz., yra grindžiami **aktyvaus pavidalo** (formos) (angl. *Active Shape*) bei **aktyvaus kontūro** (angl. *Active Contour*, dar vadinamas „Snake“) [BP02, GBA02] principais. Dauguma tokio tipo algoritmų yra naudojama vaizdų sekoje veidams sekti, nes geriausiai veikia, kai yra užduotas pirminis ne itin nutolusio vaizdo, kuriame yra dominantis veido objektas, kontūras. Tada kituose vaizduose metodas evoliucionuoja į veido kontūrą gerai. Tačiau bendru atveju algoritmai kenčia nuo nesėkmingo pradinio kontūro parinkimo problemos. Dažnai pasitaiko tokios segmentavimo klaidos: įtraukiamos perteklines kaklo ar plaukų zonos; arba atvirksčiai – nukerpama viena iš akių. Be to, daugelyje su aktyviais kontūrais veidams segmentuoti skirtų straipsnių yra paminėta, kad testavimo metu buvo eliminuojamos individų su akiniais, ūsais ar barzda nuotraukos. Taip pat dažniausiai testuotos tik vienos etninės grupės žmonių mažos duomenų bazės. Tai reiškia, kad tokių metodikų taikymas įgalintų sukurti sistemą su gana dideliais suvaržymais biometrinės sistemos taikymo aplinkai. O gali būti iškeltų papildomų reikalavimų ir su ja sąveikaujantiems vartotojams, kas eliminuotų esminį biometrinį veidų sistemų privalumą – lengva priimtinumą visuomenėje dėl itin paprasto, papildomų pastangų nereikalaujančio naudojimo. Tokių metodų taikymas taip pat riboja veidų biometrijos taikymą įslaptintose sistemose.

1.3. Teorinės apžvalgos išvados

Taigi, veidų segmentacijai skirtų algoritmų yra iš tiesų daug. (Ankstesniame skyriuje buvo išvardinta tik dalis jų. Plačiau galima pasiskaityti išsamiose apžvalgose [HL01, YKA02]). Kiekviena išvardintų algoritmų grupių turi savų privalumų ir trūkumų, esminiai iš kurių yra trumpai išvardinti toliau algoritmų palyginimui.

Veido odos spalva grindžiami segmentacijos algoritmai yra našūs, tačiau teikia mažą veidų segmentacijos kokybę (angl. *Detection Rate*, toliau – *DR*) ir daug klaidingų veidų sričių kandidatų (angl. *False Acceptions*, toliau – *FA*), kurias reikia apdoroti papildomame klaidingų sričių eliminavimo žingsnyje. Netgi patobulintų šių algoritmų versijų [SPJ04], kuriose kombinuojamos veidų odos spalvos pasiskirstymo charakteristikos įvairiose spalvinėse erdvėse, autoriai deklaruoja segmentavimo kokybę kintančią 71 – 94 % režiuose. Realiai tokio tipo algoritmų segmentavimo tikslumas yra gana silpnas ir siekia apie 80%. Be to, vis dar lieka opi didelio klaidingų segmentacijų skaičiaus (angl. *False Detection Rate*, toliau – *FAR*) problema.

PCA ir **tikrinių veidų** (angl. *eigenfaces*) grindžiamos technikos atveju veidų sėkmingo suradimo dažnis varijuoja nuo 64 iki 96 %, priklausomai nuo apšvietimo, orientacijos ir mastelio veiksmų [TP91]. Algoritmų, naudojančių **SVM** metodiką [OFG97] tikslumas varijuoja nuo 74 iki 97 % (ant labai mažo skaičiaus testavimo duomenų). **SNoW** metodika [YRA00] pagal straipsnių autorių pateiktus įverčius veikia apytiksliai 91 – 94 % tikslumu (tačiau vėl gi testuota ant nedidelio 150 paveikslų su 600 veidų skaičiaus). Jų metodo tikslumas artimas **neuroninius tinklus** taikančių veidų segmentacijos algoritmų tikslumui. **BDF** algoritmo straipsnio [Liu03] autorius teigė pasiekęs 97% sėkmingo veidų suradimo vaizduose statistiką. Tačiau tokius įverčius jis pateikia atlikęs testus ant 80 paveikslų su 227 veidais. Savo metodiką jis lygina tik su [SK00] straipsnio autorių metodais, kurie pirmieji taikė **AdaBoost** pagalba apmokintus klasifikatorius.

Pirmuosius AdaBoost taikymo rezultatus publikavę H. Schneiderman and T. Kanade [SK00] deklaravo apytiksliai 90 – 93 % sėkmingos veidų segmentacijos statistiką. Tačiau vėliau P. Viola ir M. Jones mokslininkų pagalba apmokinti ne vienos, o kelių pakopų (angl. *stages*) **AdaBoost** kaskadiniai veidų klasifikatoriai (angl. *cascade detectors*) teikė 93.7 % segmentavimo kokybę, kartu su nepalyginamu dešimtis kartų geresnis našumu [VJ01a, VJ01b]. Nors savo straipsnyje minėti mokslininkai nurodo rezultatus ant labai nedidelių duomenų aibių (pvz., MIT/CMU), šio kaskadinio algoritmo pranašumas (pvz., pusiau naivaus Bajeso (angl. *semi-naive Bayesian*) ir **SNoW** klasifikatorių atžvilgiu) buvo patvirtintas ir kitų vėlesnių straipsnių autorių [BAS+07], kurie atliko lyginamuosius testus ant naujos, kur kas didesnės FRGC duomenų bazės (žr. priedą Nr. 1). Taigi, lygi šių dienų nė vienas veidų segmentavimo / klasifikavimo algoritmas neteikia

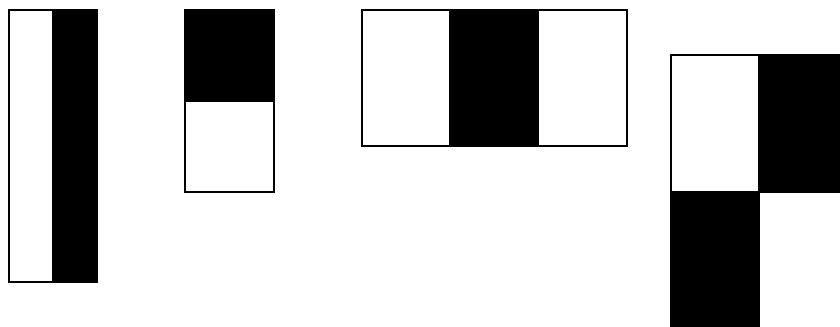
tokio aukšto sėkmingo veidų suradimo vaizduose ir greičio rodiklių santykio. Šis algoritmas įgalino taikyti veidų segmentaciją realaus laiko sistemose. Be to, Viola-Jones *AdaBoost* algoritmas leidžia balansuoti ir labiau nukrypti pageidaujamo našumo arba tikslumo prioriteto linkme. Itin nestipriai įtakojant kaskadinio klasifikatoriaus našumą, segmentavimo kokybę galima pagerinti, parenkant reprezentatyvesnę apmokymo duomenų bazę arba naudojant daugiau požymių. (žr. sk. 1.4), taip pat parenkant racionalesnius mokymosi parametrus (dažniausiai – eksperimentu būdu). Geriausiai (galimai dar geresnius) segmentacijos tikslumo ar našumo įverčius deklaruojantys naujausi straipsniai yra Viola-Jones metodikos ištobulintos modifikacijos, naudojančios platesnę požymių aibę arba kiek sudėtingesnę mokymą, kaskados žingsnių gardele (angl. *grid*). Kitame skyriuje bus išsamiau aprašytas *AdaBoost* kaskadinių klasifikatorių konstravimo principas.

1.4. „Boosting“ metodika grindžiama segmentacija

P. Viola ir M. Jones [VJ01a] savo darbe pristatė tris esminius dalykus:

1. integralinio paveikslo (angl. *integral image*) idėją greitam požymių atsakų apskaičiavimui;
2. efektyvų *AdaBoost* apmokymo algoritmą, kuris sukonstruoja gerais segmentavimo ir greičio rodikliams pasižyminčius stipriuosius klasifikatorius;
3. stipriųjų klasifikatorių (angl. *strong classifiers*) kombinaciją į kaskadą, sprendžiant veidų klasifikavimo problemą „nuo paprasčiausio prie sudėtingiausio“ (angl. „*simple-to-complex*“) principu.

Integralinio paveikslo idėjai aprašyti, iš pradžių būtina pristatyti požymius, kuriuos naudojo straipsnio autoriai veidų ir neveidų klasifikavimo problemai spręsti. Jie naudojo stačiakampius Haar tipo požymius, kuriuos galima matyti pav. 6. Bet kurio iš naudojamų požymių stačiakampiai visada yra tokios pačios formos ir dydžio bei vertikalčiai arba horizontalčiai gretimi (t.y. turi bendrą kraštinę). Požymio atsakas – įeties skaitmeniniame vaizde baltais stačiakampiais dengiamų taškelių sumos skirtumas su juodais stačiakampiais dengiamų taškelių suma.



Pav. 6. P. Viola ir M. Jones naudojami Haar tipo požymiai. Juodos taškelių sritys yra sumuojamos su neigiamu ženklu, baltos – su teigiamu.

Integralinis paveikslas I – tai greitam pav. 3 požymių apskaičiavimui skirta tarpinė segmentuojamo įeties vaizdo I reprezentacija, kurios reikšmė taške $I(x, y)$ yra lygi sumai visų vaizdo taškelių esančių virš ir po kairę nuo esamo taškelių $I(x, y)$ (įskaitant patį taškelį), t. y.

$$I(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y'). \quad (1)$$

Integralinis atvaizdas I gali būti apskaičiuojamas vienu praėjimu per įeties skaitmeninį vaizdą naudojant tokius rekursinius sąryšius (2) ir (3):

$$sum(x, y) = um(x, y -) + (x, y), \quad (2)$$

$$II(x, y) = I(x - , y) + um(x, y). \quad (3)$$

Čia $sum(x, y)$ – kaupiamoji (angl. *cumulative*) eilutės suma. Daroma prielaida, kad $sum(x, -1) = 0$ ir $II(-1, y) = 0$. Naudojant integralinę reprezentaciją bet kokio stačiakampio taškelių suma gali būti apskaičiuota per keturis taškus integraliniame atvaizde (žr. pav. 7).

A 1	B 2
C 3	D 4

Pav. 7. Integraliniame vaizde skaičiai nuo 1 iki 4 žymi atitinkamų stačiakampių nuo A iki D originaliame vaizde taškelių sumas. Bet kurio originalaus vaizdo stačiakampio D taškelių suma gali būti apskaičiuojama kaip tokia integralinio vaizdo taškų išraiška: $4 + 1 - (2 + 3)$.

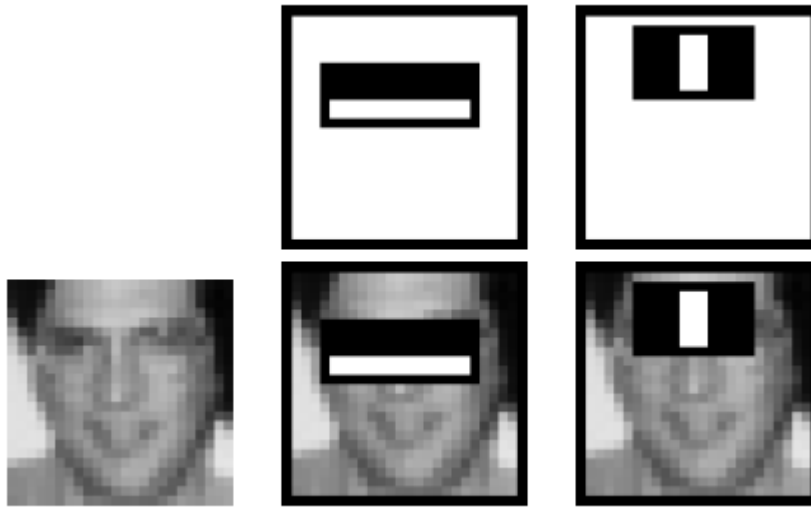
Minėti sparčiai apskaičiuojami požymiai yra naudojami silpniesiems klasifikatoriams $h_j(x)$ (angl. *weak classifiers*) atrinkti, kurie turi geriausiai atskirti pozityvių (veidų) ir negatyvių (neveidų) mokymo vaizdų priklausomybę atitinkamoms klasėms. Silpnąjį klasifikatorių $h_j(x)$ sudaro:

- požymis $f_j(x)$;
- jam taikomas optimalus slenkstis θ_j (angl. *threshold*), kuris minimizuoja klasifikatoriumi daromu klaidų skaičių;
- poliškumo p_j (angl. *polarity*) koeficientas, kuris nusako, kurioje slenksčio pusėje yra atitinkamai veidų ir neveidų klasių atsakai.

Taigi, silpnąjį klasifikatorių galima apibrėžti žemiau aprašyta lygtimi.

$$h_j(x) = \begin{cases} 1, & \text{kai } p_j f_j(x) < \nu_j \theta_j, \\ 0, & \text{kai } p_j f_j(x) \geq \nu_j \theta_j, \end{cases} \quad (4)$$

Čia x – originaliame vaizde nagrinėjamo stačiakampio langas, kurio dydis apmokinimo metu yra fiksuotas ir lygus apmokinimo paveikslukų dydžiui, pvz. 20×20 taškelių. Reiktų pažymėti, kad algoritmu atrenkami požymiai pasižymi klasifikavimo klaida e_j svyruojančia [30%, 50%) intervale. (Dėl to jie vadinami silpnaisiais.) Pirmuosiuose iteracijose atrenkami stipriausi požymiai su apytiksle 30% klaida. Keli iš jų yra pavaizduoti pav. 8.



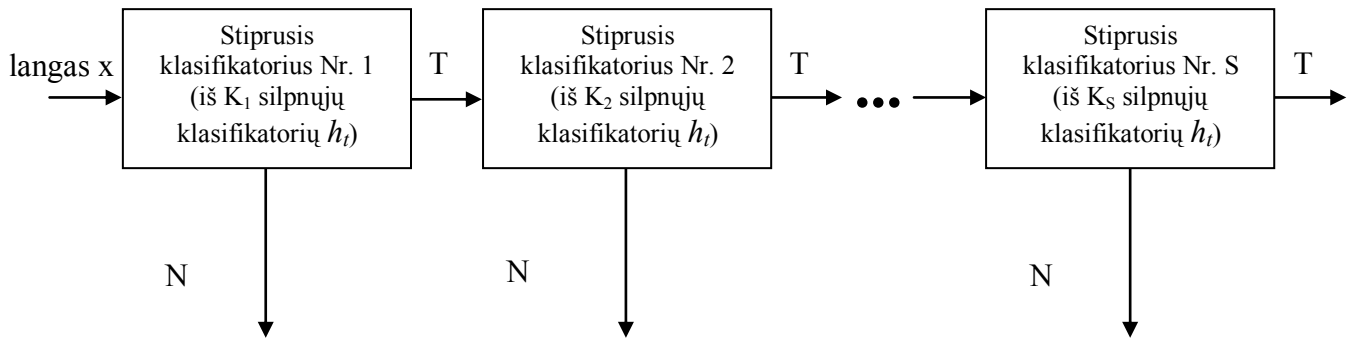
Pav. 8. P. Viola ir M. Jones algoritmu atrinkti stipriausi silpnieji klasifikatoriai [VJ01a].

Taigi paties algoritmo esmė yra naudojant sutarto dydžio, pvz., 20 x 20 taškelių normalizuotus veidų ir neveidų apmokymo vaizdus, atrinkti iš plačios galimų tokio dydžio vaizde stačiakampių požymių aibės h_j stipriausius h_t su klaida $e_t < 50\%$. Tada apjungti šiuos stipriausius h_t iš silpnųjų klasifikatorių h_j į stiprųjį klasifikatorių H , suteikiant silpniesiems svorinius balsavimo koeficientus α_t pagal atvirkščią proporciją jų daromai klaidai e_t . T.y. didžiausią balsą α_t stipriojo klasifikatoriaus H galutiniame „veidas / neveidas“ tipo atsake įgyja silpnieji klasifikatoriai h_t su mažiausia klasifikavimo klaida e_t . (angl. *misclassification error*). Būtent tokia „*boosting*“ kombinavimo technika įgalina iš „kažko“ silpniesnio išgauti „kažką“ stipresnio. Ir tokia strategija iš tiesų pasiteisino algoritmo tikslumo prasme. Pats algoritmas yra pateiktas schemoje 1. Esminis skirtumas palyginus su kitomis apmokymo technikomis – tai svorinis klasių apmokymo pavyzdžių pasiskirstymas: kiekvienas mokymo duomenų pavyzdys x_i (angl. *sample*) turi savo svorį $w_{i,t}$, kuris normalizuotas jo klasės y_i viduje. Po kiekvienos iteracijos, t.y. naujo silpniojo klasifikatoriaus h_t su požymiu f_t pridėjimo į stiprųjį klasifikatorių H , mokymo pavyzdžių svoriai $w_{i,t}$ yra perskaičiuojami tokiu būdu, kad didžiausius svorius įgauna klaidingai klasifikuoti elementai, mažiausius – teisingai. Taip susikoncentruojama ties neteisingai klasifikuotų elementų problema.

Kaip jau buvo minėta, straipsnio autoriai naudoja ne vieną klasifikatorių, o kombinuoja stipriuosius klasifikatorius H į vieną kaskadinį klasifikatorių (žr. pav. 9). Kai yra analizuojamas tam tikras modelio lango dydžio vaizdas, jis nuosekliai keliauja per kaskadoje esančius stipriuosius klasifikatorius. Tačiau, jeigu bent vienas stiprusis klasifikatorius grąžina neigiamą atsaką (t. y. klasifikuoja kaip neveidą), toliau vaizdas nebekeliauja klasifikatorių grandine – kaskados galutinis atsakas yra „neveidas“. Tai reiškia, kad tik veidai yra apdorojami visų kaskadoje esančių klasifikatorių. Kadangi veidų langų tikimybė yra maža (palyginus su neveidų langų statistika) – kaskados darbas paspartėja dešimtis kartų. Dažniausiai pirmieji klasifikatoriai yra sudaryti vos iš kelių požymių (pvz., 8 pav. pavaizduoti požymių) paprasčiausiems neveidams atmesti. Vėliau apmokyti kiti stiprieji klasifikatoriai būna didesni ir sudėtingesni skaičiavimo prasme, nes jiems tenka sudėtingesnis veidų ir neveidų klasifikavimo uždavinys. Tai įgalina greitą į neveidus ne itin panašaus fono tekstūros langų atmetimą ir itin greitą itin paprasto fono atmetimą.

Dėl anksčiau aprašytų požymių savybių, algoritmas leidžia segmentuoti bet kokio dydžio veidus bet kokuose vaizduose. Tai įgyvendinama kas tam tikrą žingsnį taikant apmokintą įvairiems masteliams apskaičiuotą kaskadinį klasifikatorių.

Straipsnio autoriai aprašė rezultate gautą 38 stipriųjų klasifikatorių H pakopų klasifikatorių su iš viso 6061 požymių f_t priešakiniais veidams segmentuoti. Kiekviena pakopa apmokinta naudojant 24 x 24 taškelių dydžio 9544 veidų 10000 neveidų paveiksliukus. Vėliau apmokintos pakopos sprendavo vis sudėtingesnes klasifikavimo problemas, kadangi klasifikavo labiau į veidus panašius neveidų egzempliorius, neatmestus ankstesniais klasifikatoriais.



Pav. 9. Kaskadinio P. Viola ir M. Jones klasifikatoriaus bendra schema.

Gautasis klasifikatorius yra 15 kartų greitesnis už populiarųjį H. A. Rowley ir kt. segmentatorių [RBK98]; taip pat 600 kartų greitesnis už H. Schneiderman and T. Kanade detektorius [SK00]. Vidutinis 384×288 taškelių dydžio paveiksluko apdorojimo laikas naudojant kaskadinį algoritmą ant kompiuterio su Pentium II 700 Mhz procesoriumi – 67 milisekundės. Algoritmo tikslumas gali siekti 94.8% ant MIT/CMU priešakinių veidų duomenų bazės (žr. priedą Nr. 1). Kaskadinio klasifikatoriaus algoritmo tikslumas yra kiek prastesnis, negu nekaskadinio su tokiu pačiu bendrų silpnųjų klasifikatorių skaičiumi, tačiau algoritmo našumas yra bent dešimt kartų geresnis [VJ01a]. Reikėtų paminėti, kad algoritmo tikslumas priklauso ne tik nuo apmokinto klasifikatoriaus, bet ir nuo žingsnio, kuriuo mes ieškome veidų skaitmeniniame vaizde; taip pat nuo žingsnio, kuriuo mes keičiame kaskados klasifikatorių mastelį.

P. Viola ir M. Jones pasiūlytas kaskadinio veidų klasifikatoriaus algoritmas yra iš tiesų geras, tačiau jis palieka daug klausimų teoriniam ir praktiniam nagrinėjimui, pvz.:

- Koks yra optimalus pakopų skaičius?
- Koks turi būti požymių skaičius pakopose?
- Koks turi būti pakopos slenkstis, kad mes neatmestume per daug veidų (neišaugtų *False Rejection Rate*, toliau – *FRR*), tačiau visą laiką nenagrinėtume ir per daug neveidų (pvz., fono) langų (neišaugtų *FAR*)?

Deja, daugelis šios strategijos naudojimo klausimų sprendžiami praktiškai nagrinėjant algoritmą ir tyrinėjant veidų ir neveidų klasių statistikas. Tokio tyrinėjimo rezultatai yra pristatyti šio darbo praktinės analizės dalyje. Taip pat buvo ištestuotos *AdaBoost* algoritmo modifikacijos, kurias kiti autoriai deklaravo kaip tikslesnius sprendimus veidų segmentacijai atlikti, t. y. *RAB*, *GAB* ir *LAB* (žr. lentelę 1) naudojantis *CART* medžius vietoje vieno požymio silpnųjų klasifikatorių. Ne visi minėtų autorių išdėstyti lyginamieji teiginiai pasitvirtino praktinių eksperimentų (aprašytų skyriuje 2) eigoje.

1. Sudaroma $(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)$ mokymo duomenų bazė. Čia x_i — skaitmeninis modelio dydžio vaizdas, $y_i = \{0, 1\}$ — klasės žymė.

$$y_i = \begin{cases} 1, & \text{kai } x_i \text{ — veidas} \\ 0, & \text{kai } x_i \text{ — neveidas} \end{cases}$$

2. Inicializuojami mokymo duomenų bazės egzempliorių svoriai.

$$w_{1,j} = \begin{cases} \frac{1}{2l}, & \text{kai } y_j = 1 \\ \frac{1}{2m}, & \text{kai } y_j = 0 \end{cases}$$

Čia m — skaičius neveidų vaizdų, l — skaičius veidų vaizdų.

3. Kol atrinktų požymių skaičius $t \neq T$, t. y. $t = 0 \dots T$

- 3.1. Normalizuoti svorius.

$$w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$$

- 3.2. Kiekvienam požymiui f_j apmokinti silpnąjį klasifikatorių h_j (žr. formulę (4)). Požymio klasifikavimo klaida e_j įvertinama pagal žemiau esančią formulę.

$$e_j = \sum_{i=1}^n v_i |h_j(x_i) - y_i|$$

- 3.3. Išsirinkti h_t , kurio klaida $h_t = \min(e_j)$

- 3.4. Atnaujinti svorius.

$$w_{t+1,i} = v_{t,i} \beta^{e_{t,i}}$$

$$\text{Čia } \beta = \frac{e_t}{1 - e_t}, \text{ o}$$

$$e_i = \begin{cases} 0, & \text{kai } h_t(x_i) \neq v_i \\ 1, & \text{kai } h_t(x_i) = v_i. \end{cases}$$

Rezultatas — stiprusis klasifikatorius:

$$H(x) = \begin{cases} 1, & \text{kai } \sum_{t=1}^T \alpha_t(x) > 0.5 \sum_{t=1}^T \alpha_t \\ 0, & \text{priešingu atveju.} \end{cases}$$

$$\text{Čia } \alpha = \log \frac{1}{\beta}.$$

Schema 1. P. Viola ir M. Jones *AdaBoost* algoritmas stipriajam klasifikatoriui apmokinti

<i>Viršūnių skaičius silpnajame CART klasifikatoriuje</i>	1	2	3	4
DAB	45.09	44.43	31.86	44.86
GAB	30.99	36.03	28.58	35.40
RAB	26.28	33.16	26.73	35.71

Lentelė 1. Vidutinis skaičius kaskados požymių atsakų, apskaičiuojamų klasifikuojant neveidų 20 x 20 taškelių dydžio langą. Kuo šis skaičius mažesnis, tuo mažesnis stipriųjų klasifikatorių klaidingų neveidų klasifikavimo (kaip veidų) klaidos dažnis. (*FAR*) ir geresnis kaskados našumas. Tačiau mažesnis *FAR* gali būti didesnio *FRR* pasekmė. Lentelė paimta iš [LKP02].

1.5. Naiviaisiais Bajeso klasifikatoriais grindžiama segmentacija

Naivusis Bajeso (NB) klasifikatorius yra grindžiamas Bajeso sąlyginių tikimybių teorema [Wik11], kuri pateikta žemiau esančioje formulėje (5). Šioje formulėje $P(Y|X)$ žymi ieškomą sąlyginę klasės Y tikimybę, kai žinomi duomenys X . $P(X|Y)$ vadinamas tikėtimumo įverčiu (angl. *likelihood*) – sąlygine duomenų X tikimybe, kai žinoma, kad šie priklauso klasei Y . $P(Y)$ yra nurodoma apriorinė klasės Y tikimybė (prieš bandymą). $P(X)$ žymi duomenų X tikimybę, kuri dažniausiai interpretuojama kaip klasifikavimo rezultatų neįtakojantis normalizacijos faktorius (nes šis nepriklauso nuo klasės Y , o tik nuo duomenų X).

$$P(Y | X) = \frac{P(X | Y) P(Y)}{P(X)} \quad (5)$$

$$P(Y | x_1, x_2, \dots, x_n) = \frac{P(x_1, x_2, \dots, x_n | Y) P(Y)}{P(x_1, x_2, \dots, x_n)} \quad (6)$$

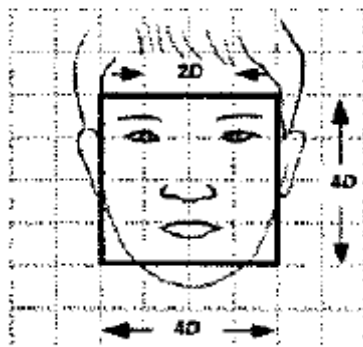
$$P(Y | x_1, x_2, \dots, x_n) = P(Y) \prod_{k=1}^n P(x_k | Y) \quad (7)$$

Reikėtų pastebėti, kad duomenys X dažniausiai nėra trivialūs ir yra sudaryti iš $n > 1$ atributų, t.y. $X = \{x_1, x_2, \dots, x_n\}$. Tokiu atveju formulėje (5) pateikta Bajeso teorema įgauna (6) pavidalą. NB metodika daro prielaidą apie atributų tarpusavio nepriklausomybę. Jei visi atributai x_1, x_2, \dots, x_n kinta nepriklausomai vienas nuo kito, duomenų X priklausomybės klasei Y tikėtimumas $P(X|Y)$ yra įvertinamas kaip minėtų atributų tikėtimumų $P(x_k|Y)$ sandauga (7). Nors minėta nepriklausomybės prielaida yra gana grubi, tačiau ji pasiteisina daugelių klasifikavimo užduočių atveju (pvz., paveikslų tapatybės identifikacijai [Ker02], rašytinių simbolių [Hua07], [Bas10b], gestų [ASM+03] ir transporto priemonių numerių [WZM+10] atpažinimui), netgi kai atributai galimai nėra visiškai nepriklausomi. Tai sąlygojama tuo, kad NB klasifikatoriaus sprendimai grindžiami maksimalios sąlyginės posteriorinės klasės tikimybės $P(Y|X=\{x_1, x_2, \dots, x_n\})$ išrinkimu (angl. *maximum a posteriori*, toliau – MAP). Ši taisyklė aprašyta formulėje (8). Reikia pripažinti, kad NB klasifikatorius atsilieka nuo įvairių sprendimų medžių algoritmų savo tikslumu, tačiau jis yra itin patrauklus daugiadimensinių duomenų klasifikavimo uždaviniams spręsti, kadangi konvertuoja pastaruosius į kiekvieno iš atributų paprastesnius vienos dimensijos klasifikavimo uždavinius. Taip pat daugelio NB klasifikatorių taikančių medicinos, biologijos ir kitų sričių straipsnių autorių pabrėžia, kad ši metodika pateikia žmonies intuityviai suvokiamus sprendimus, skirtingai nuo daugybės egzistuojančių sunkiau tobulinamų ir derinamų „juodos dėžės“ tipo klasifikavimo technikų. Intuityvus klasifikavimo modelio suvokimas itin svarbus įvardintose pirmosiose srityse, kur dažniausiai vykdomas pusiau automatinis klasifikavimas, t.y. galutinį sprendimą priima ekspertas. Ši savybė pasitarnavo ir šiame darbe realizuotos visiškai automatinės veidų segmentacijos sistemos derinimui bei jos tikslumo tobulinimui. Dar vienas vertas paminėjimo NB metodikos esminių patrauklumų – itin nedidelis duomenų kiekis, reikalingas klasifikatoriui sukonstruoti (apmokinti). Tai išryškėjo ir mūsų darbe palyginus DAB ir NB klasifikatoriams apmokinti reikalingų duomenų kiekį ir laiką.

$$Classifier_{NB}(x_1, x_2, \dots, x_n) = \max_c P(Y_c) \prod_{k=1}^n P(x_k | Y_c) \quad (8)$$

Būtina pabrėžti, kad *NB* klasifikatorius gali būti taikomas tiek diskretiems, tiek tolydiems atributams. Pastaruoju atveju atributų tikimybiniai modeliai sudaromi pasirenkant tam tikrą pasiskirstymą, pvz., Gauso normalųjį pasiskirstymą $x_k \sim N(\mu_k, \sigma_k^2)$ parametrizuotą vidurkio μ_k ir nuokrypio σ_k^2 reikšmėmis, kurios praktiškai apskaičiuojamos mokymo duomenims. Tolydiems duomenims gali būti taikoma ir diskretizacijos strategija – duomenų kvantavimas į intervalus (angl. *bins*). Praktikoje turint didesnę mokymo duomenų kiekį patariama taikyti pastarąją strategiją, t.y. diskrečius duomenis. Tačiau, jeigu diskrečių reikšmių kiekis kiekvienam iš atributų yra pernelyg didelis, *NB* klasifikatorius tampa labai griozdiškas ir ne itin tinkamas klasifikavimui. Todėl, šiame darbe sprendžiant fiksuoto modelio dydžio vaizdų sričių duomenų $X = \{x_1, x_2, \dots, x_n\}$ į vieną iš dviejų ($c = 2$) veidų (Y_1) arba neveidų (Y_2) klasių buvo naudojamos ne taškelių pilkumo reikšmės $x_k \in [0, 255]$, o, ieškoma kitų patikimų diskrečių požymių. *NB* klasifikatoriaus praktinis pritaikymas realizuotoje priešakinių veidų segmentacijos sistemoje aprašytas skyriuje 3.2.2.

NB klasifikatoriais grindžiama veidų segmentacija nėra itin populiarus sprendimas, todėl jis nebuvo pateiktas populiariausių veidų segmentacijos metodų apžvalgos skyriuje 1.2. Tačiau ketvirtajame semestre atliktos išsamesnės mokslinės literatūros (susijusios su „*boosting*“ ir naiviaisiais klasifikatoriais) apžvalgos rezultate išaiškėjo, kad *NB* klasifikatorių naudojimas veidų biometrijai nėra naujovė. Dažniausiai veidų suradimo etape *NB* klasifikavimo algoritmai nėra naudojami pavieniui, o yra kombinuojami su kitomis segmentacijos metodikomis geresniam našumui ir/ arba tikslumui išgauti. S. L. Phung ir kt. [PBC+03] naudojo veido odos, akių spalva ir veido geometriniu modeliu (žr. pav. 10) grindžiamą segmentaciją veidų kandidatiniams sritims surasti. Toliau normalizuotos kandidatiniės sritys buvo klasifikuojamos į veidus ir neveidus naudojant kombinaciją atsakų iš trijų *NB* klasifikatorių, apmokintų ant 64×64 dydžio kraštų (angl. *edge*) binarizuotų vaizdų, 16×16 taškelių dydžio kvantuotų intensyvumo reikšmių bei *PCA* tikrinių veidų. Jų algoritmas pasižymi veidų teisingo klasifikavimo dažniu $DR = 90.04\%$ (218 iš 231 veido) prie netesingai klasifikuotų neveidų absoliutaus skaičiaus $FA = 10$. Jų atliktų lyginamųjų eksperimentų rezultatuose deklaruojama, kad minėtas metodas teikia ekvivalentų tikslumą CMU detektoriumi [ST98], tačiau suranda daugiau nei 20 kartų mažesnę netikrų veidų skaičių.

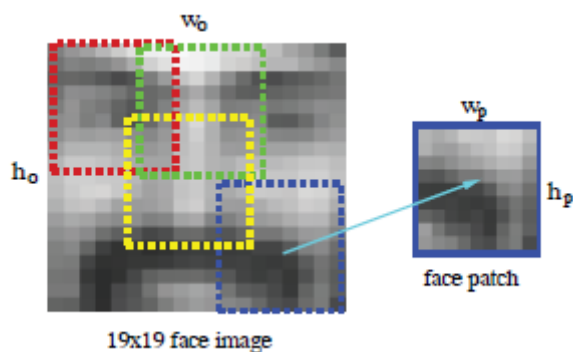


Pav. 10. Geometrinis veido modelis, kuriuo remiantis S.L. Phung ir kt. [PBC+03] sudarė atstumų tarp tam tikrų veido taškų (pvz, akių) euristikas pirminei kandidatinių veidų sričių segmentacijai vaizduose.

S. L. Phung su kitais bendraautoriais [PBD+04] taip pat naudojo veido spalvos tipo segmentatorių galimiems veidų kandidatams vaizde surasti. Toliau jie eksperimentavo su kandidatų klasifikavimo į veidus ir neveidus problematika mokindami *NB* klasifikatorius ant požymių, išgautų skirtingais būdais apdorojus 64×64 taškelių dydžio vaizdus (pvz., pritaikius vidurkio ar dispersijos normalizaciją, histogramos išlygiavimą (angl. *histogram equalization*)), Siekdami išsiaiškinti *NB* pritaikomumą veidų klasifikacijos uždaviniui ir įvairių normalizacijos technikų įtaką *NB* tikslumui, autoriai naudojo tokius požymius: kvantuotas pikselių intensyvumo reikšmes, veidų poerdvio projekcinis vektorius, binarizuotus kraštų (angl. *edge*) vaizdus, vertikalias ir horizontalias vaizdo matricos projekcijas. Eksperimentų išvadose jie nurodo, kas *NB* klasifikatorius veikia geriau nei koreliacinis šablonų lyginimo algoritmas. Jie deklaruoja geriausius rezultatus naudodami histogramos išlygiavimo tipo normalizuotų vaizdų kvantuotas

pikselių intensyvumo reikšmes arba kombinuodami visų keturių požymių *NB* klasifikatorių atsakus. Pirmuoju atveju jie pasiekė 6000 veidų iš įvairių veidų duomenų bazių (BioID, UMIST, AT&T, kt.) teisingo klasifikavimo dažnį $DR = 77.8\%$ ir $DR = 92.2\%$, antruoju – $DR = 96.2\%$ ir $DR = 99.2\%$ atitinkamuose netesingo 6000 neveidų klasifikavimo dažnio taškuose $FAR = 1\%$ ir $FAR = 5\%$. Kaip ir šio darbo eigoje atliktų praktinių bandymų ant FERET ir FRGC duomenų bazių rezultate (žr. skyrių 3.2.2), kraštų požymiai pasirodė esą netinkami tiksliai veidų klasifikacijai, nes jie yra ženkliai įtakojami apšvietimo ir intraklasinės veidų klasės variacijos.

B. S. Venkatesh ir S. Marcel [SM10] savo darbe kombinuoja *NB* klasifikatorių su Viola-Jones analogiška 5 stipriųjų klasifikatorių *DAB* kaskada. Kaskada yra sukonstruota ne iš Haar, bet iš kitokio tipo vadinamųjų *MCT* (angl. *Modified Census Transform*) požymių, kurie autoriais apibūdinami kaip atsparesni apšvietimo kaitai požymiai. *NB* klasifikatorius šiuo atveju naudojamas ne galutinėje kandidatų sričių verifikavimo fazėje, o kaskados taikymo etape – šios spartumui pagerinti maksimizuojant tikslumą. Tam jie naudojo veido persidengiančių dalių (angl. *patch*, žr. pav. 11) *NB* klasifikatorius tam tikrose kaskados pozicijų praretintos gardelės reliatyviuose vietose galimo veido gaubiančiajam stačiakampiui nustatyti. Šio gaubiančiojo stačiakampio aplinkoje jie taikė *DAB* kaskada. Patys *NB* klasifikatoriai naudoja paparčio tipo požymius (anglų kalba pavadintus μ -*ferns*), kurie grąžina binarinius atsakaus koduojančius atsitiktinių taškelių pozicijose esančias reikšmių palyginimą su vidutine pilkumo reikšme veido dalies modelio lange. Autoriai deklaruoja, kad veido dalių *NB* klasifikatorių pritaikymas pagerino greičio įverčius bei padidino DR 15 – 30% prie pakankamai išretintos skenavimo pozicijų gardelės, palyginus su įprastine skenavimo kaskada technologija,

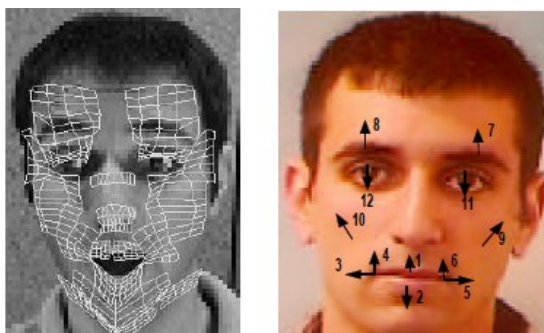


Pav. 11. *DAB MCT* požymių kaskados modelio išvestinės persidengiančios veido dalių sritys. Šias sritis B. S. Venkatesh ir S. Marcel [SM10] klasifikavo naudodami paparčio požymių apmokintus *NB* klasifikatorius gaubiančiajam veido stačiakampiui surasti kaskados veikimo fazėje.

Dauguma kitų mokslinių straipsnių autorių aprašo *NB* klasifikatorių ir jų kombinacijos su kitomis metodikomis taikymus veidų išraiškai ir orientacijai erdvėje nustatyti segmentacijos etape. S. M. Lajevardi ir Z. M. Hussain [LH09] (taip pat jų darbas [LNH09]) naudoja *AdaBoost* ir *NB* algoritmus veidų paieškai ir klasifikavimui į vieną iš šešių galimų emocijų klasių: pykčio, nepasitenkinimo, baimės, laimės, liūdesio, nuostabos. Visų pirma, *DAB* kaskada ieško veidų kandidatų normalizuotame (naudojant histogramos išlygiavimą) vaizde ir toliau priskiria juos tam tikrai emocijos klasei naudojant *NB* klasifikatorių, grindžiamą genetiniu algoritmu ištrinktais efektyviausiais *log-Gabor* filtro požymiais. N. Sebe ir kt. [SLC+02] pasiūlė veidų išraiškų klasifikavimo sprendimą remiantis iš sužymėtų metaduomenų sukonstruotu trimačiu vėliniu rėmeliu (angl. *3D wireframe*), Jie išrinko 12 judesio vektorių (angl. *motion units*) – nuotolio nuo neutralios emocijos požymių (žr. pav. 12). Vienai iš minėtų šešių arba neutraliai išraiškai nustatyti jie taikė *NB* klasifikatorius su prielaida, kad požymiai turi Koši pasiskirstymą. Savo praktiniais eksperimentų rezultatais jie parodė, kad Koši pasiskirstymo prielaida teikia geresnius emocijų veido išraiškų klasifikavimo rezultatus nei Gauso. S. Baluja [Bal99] nagrinėjo veido orientacijos erdvėje nustatymo iš tekstūros galimybes naudojant *NB*, Bajeso tinklą (angl. *Bayesian networks*) ir medžias papildytus *NB* (angl. *Tree Augmented Naive Bayes*, toliau – *TAN*) klasifikatorius. Klasifikatoriai buvo apmokami ant histogramos išlygiavimo filtru normalizuotų

ir binarizuotų skirtingai pasisukusių veidų, naudojant tiek sužymėtus (pagal posūkio klases), tiek nesužymėtus duomenis geresnėms tikslumo rezultatams pasiekti. Z. Zhang ir kt. [ZHL+07] siūlo kombinuoti *NB* klasifikatorius su *FAB* kaskadų ir paslėptojo Markovo modelio (angl. *Hidden Markov Model*, toliau – *HMM*) metodais. Jie naudojo penkias *FAB* kaskadas priešakiniais, dešinės bei kairės pusės pusiau pasuktiems, dešinės bei kairės pusės visiškai profilineiems veidams surasti vaizdų sekose, kurios gaunamos iš skirtingais fiksuotais kampais nustatytų kamerų. Toliau kiekvienos iš kameros *FAB* klasifikatorių atsakai (kampų informacija) yra paduodama *NB* klasifikatoriui, kuris apmokintas maksimizuoti teisingą kampo erdvėje nustatymą remiantis iš keturių kamerų gautais kaskadų numeriais (t.y. kampais filmuojančios kameros atžvilgiu). *HMM* taikomas pereinančios būsenos pozoms nusakyti.

Verta paminėti ir vieną iš *NB* taikymo pavyzdžių veidų atpažinimui. D. Maturana ir kt. [MMS09] sugebėjo pritaikyti *NB* klasifikatorius ir veidų atpažinimui naudojant lokalius binarinius požymius (angl. *Local Binary Patterns*, toliau – *LBP*). Veidas buvo tam tikro dydžio gardele suskaidomas į regionus. Kiekvienam iš regionų apskaičiuojama *LPB* požymių histograma. Lyginant veidus buvo naudojamas artimiausio kaimyno *NB* klasifikatorius (angl. *Naive Bayes Nearest Neighbor* – *NBNN*), kuris kiekvienam kandidato veido regionui ieškojo artimiausio kaimyno tarp visų etaloninių veidų atitinkamų regionų. Artimiausi regionai buvo išrenkami naudojant *LBP* histogramų *NB* klasifikatorius. Tada mažiausią kvadratų metodu buvo išrenkamas veidas su artimiausias regionais.



Pav. 12. N. Sebe ir kt. [SLC+02] naudojamas veido vėlinis rėmelis bei juo grindžiami nuotolio nuo neutralios veido išraiškos 12 požymių. Veidai klasifikuoti į 7 emocijų klases remiantis šių požymių Koši *NB* klasifikatoriumi.

Visi aukščiau išvardinti mokslinėje literatūroje aprašyti *NB* klasifikatorių taikymai veidų segmentacijos bei kitiems veidų biometrinės sistemos uždaviniams spręsti įgalino manyti, kad ši metodika galėtų būti sėkmingai pritaikyta ir realizuotos veidų segmentacijos DAB kaskada tikslumui pagerinti. Didelę įtaką pasirinkimui padarė ir ne kartą minėtas metodikas paprastumas mokymo ir taikymo atžvilgiu. *NB* metodikos pritaikymas antrajame realizuotos priešakinių veidų segmentacijos sistemos žingsnyje kartu su darbo vadovu pasiūlytu sparčiuoju eksponentiniu filtru aprašyti šio darbo skyriuje 3.2.2.

2. Veidų segmentacijos praktinė analizė

2.1. Veidų duomenų bazės

Algoritmams praktiškai įvertinti ir testuoti šiame darbe buvo naudojami priešakinių veidų atvaizdai iš tokių žinomų veidų duomenų bazių:

- FERET,
- BANCA,
- XM2VTS,
- Notre Dame HumanID,
- BioID.

Taip pat buvo naudojama viena privati nedidelė duomenų bazė, skirta testuojamų veidų etniniai grupei išplėsti.

Daugiau informacijos apie šias ir kitas veidų duomenų bazės galima rasti remiantis priede Nr. 1 pateiktomis nuorodomis.

2.2. „Boosting“ metodikos praktinis nagrinėjimas

„Boosting“ *DAB, GAB, RAB, LAB* technikomis [LM02, LKP02] nagrinėti ir testuoti buvo naudojama viešai prieinama kompiuterinės regos biblioteka *OpenCV* [OCV09]. Šioje bibliotekoje kaip vienas iš automatinio apmokymo (angl. *machine learning*) algoritmų yra realizuotas P. Viola ir M. Jones pasiūlytas *AdaBoost stipriųjų klasifikatorių* (angl. *strong classifiers*) apmokymas ir jų apjungimas į klasifikavimo medį arba kaskadą.

OpenCV esanti „haartraining“ aplikacija suteikia galimybę apmokinti naujus kaskadinius klasifikatorius ant savų teigiamų (pvz., veidų) ir neigiamų (pvz., neveidų) apmokymo duomenų. T.y. šios aplikacijos pritaikymas neapsiriboja veidų segmentavimo sritimi – ji gali būti taikoma ir pėsčiųjų bei kitų subjektų / objektų segmentacijai skaitmeniniuose vaizduose. (Tai įrodo *OpenCV* pagal nutylėjimą duomenų direktorijoje esantys Lienhart ir kt. apmokinti klasifikatoriai, skirti priešakiniais bei pasuktiems veidams, taip pat visam kūnui arba atskirom jo dalim surasti vaizduose.) Minėta aplikacija pasižymi daugybe įvairių opcijų, tarp kurių svarbiausios leidžia nurodyti [OHT09, OB09]:

- pageidaujama pakopų, t.y. stipriųjų klasifikatorių H , skaičių kaskadoje;
- kiekvienam iš stipriųjų klasifikatorių minimalų priimtina teisingo veidų klasifikavimo dažnio įvertį (angl. *Detection Rate* arba *Hit Rate*, toliau – *DR*);
- kiekvienam iš stipriųjų klasifikatorių maksimalų priimtina neteisingo neveidų traktavimo veidais klaidos dažnį (angl. *False Acceptance Rate* arba *False Alarm Rate*, toliau – *FAR*);
- naudojamo lango modelio (kartu ir mokymo aibėse esančių vaizdų pavyzdžių dydį);
- „boosting“ algoritmo režimą: *DAB, GAB, RAB* arba *LAB* – kiekvienas iš kurių skiriasi svorių $w_{t,i}$ perskaičiavimo po kiekvienos iteracijos i strategija;
- silpnąjo klasifikatoriaus h_j požymių f_j optimalaus slenksčio θ_j parinkimo strategiją: minimizuoti bendrą klasifikavimo klaidų dažnį (angl. *misclassification error rate*), naudoti Gini indeksą, entropijos įvertį arba mažiausių kvadratų metodą;
- maksimalų viršūnių, t.y. požymių f_j silpnąjo klasifikatoriaus *CART* medyje h_j (nuo 1 iki 4) skaičių;
- mokymo aibės nagrinėjamų ir atnaujinamų svorių $w_{t,i}$ (po iteracijos i) maksimalią sumą (procentais), siekiant nukirpti (angl. *trim*) jau ankstesnėse iteracijose gerai klasifikuotus mokymo pavyzdžius, kurie įgijo itin mažus svorius $w_{t,i}$ ir didelės įtakos algoritmui esamoje iteracijoje neturi; algoritmui paspartinti perskaičiuojami tik didesnes klasifikavimo problemas turinčių mokymo elementų svoriai;
- Haar tipo požymių f_j aibės įvairovę ir dydį (P. Viola ir M. Jones [VJ01a] bazinę, praplėstą arba net naudojančią pasuktus (angl. *tilted*) požymius [LKP02]);
- klasifikuojamų objektų simetriškumo savybę, kuriai gali būti taikomas simetriškų požymių f_j (atspindžių vertikalią ašį atžvilgiu) kas antroje iteracijoje generavimo režimas, naudojant mažesnę bazinę tyrinjamų požymių aibę.

OpenCV taip pat realizuotos kelios papildomos aplikacijos automatiniam teigiamų (pvz., veidų) bei neigiamų (pvz., neveidų) mokymo pavyzdžių generavimui iš pirminių vaizdų (ar jų sąrašo) Tai įgyvendina „createsamples“ aplikacija, kuri taip pat leidžia generuoti vaizdus klasifikatoriaus testavimui, t.y. leidžia „įklijuoti“ transformuotus ir maskuotus veidų paveikslukus į neveidų atvaizdus, o transformacijos ir maskavimo parametrus nurodo pats vartotojas. Nors tai yra gana efektyvus dirbtinis testavimo duomenų sukūrimas, *FAR* klaidai tiksliau įvertinti klasifikatorių geriau testuoti ant natūralių veidų atvaizdų su atitinkamais natūraliais jų fonais. *OpenCV* „performance“ aplikacija išveda klasifikatoriaus testavimo ant tam tikro sąrašo vaizdų su

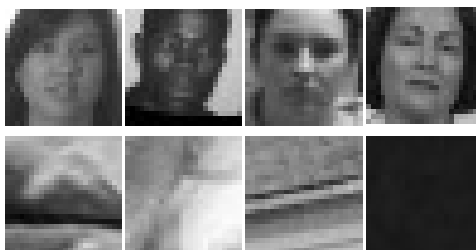
pažymėtai veidų stačiakampiais minėtų DR , FAR bei tikrų veidų nesuradimo (angl., *False Rejection Rate*, toliau – FRR) klaidų statistikas.

Šio darbo eigoje atliktų testų metų pastarosios dvi aplikacijos nebuvo naudojamos, t.y. mokymo duomenys (10000 veidų ir 10000 neveidų) buvo sugeneruoti naudojant savas aplikacijas (žr. pav. 13). Kiek pakeistas pačios „haartraining“ aplikacijos kodas, kad atsisakyti priverstinio mokymo duomenų aibių pakrovimo iš .vec failų, taikomo pagal nutylėjimą. Taip pat klasifikatorių patikimumui įvertinti buvo naudojamos ROC (angl. *Receiver Operating Charactrestics*) kreivės [WR09], kurios įgalina lengviau vizualiai įvertinti algoritmų tikslumą. Kaip jau buvo minėta, klasifikatoriai buvo mokinami ant 10000 veidų 10000 neveidų, o testuojami ant daugiau kaip 200000 modelio dydžio veidų ir tokio paties skaičiaus neveidų ROC kreivėms sugeneruoti.

Kadangi pats mokymo procesas trunka pakankamai ilgai, tai mokinti pilnas kaskadas būtų neefektyvu. Todėl buvo nuspręsta testavimą vykdyti ant atskirų stipriųjų klasifikatorių H , kurie teoriškai įeitų į galutinį kaskadinį klasifikatorių. Bendros skirtingų „boosting“ algoritmų variacijų efektyvumo tendencijos turėjo išryškėti ir ant ir atskirų pavienių stipriųjų klasifikatorių:

- I. iš pradžių apmokintų ant veidų bei labai paprastų neveidų (triukšmo) mokymo vaizdų egzempliorių;
- II. vėliau mokintų ant veidų ir kiek sudėtingesnių neveidų egzempliorių;
- III. dar vėliau ant veidų ir itin daug su veidais panašių savybių (pilkumo lygių prasme) turinčių neveidų.

Tampant sudėtingesnei veidų ir neveidų klasifikavimo problemai, proporcingai turėjo didėti ir požymių skaičius f_i stipriuosiuose klasifikatoriuose H . Tikslinga buvo išsiaiškinti, kurios „boosting“ metodikos generuoja daugiausiai požymių turinčius klasifikatorius, ir ar visada didesnis požymių skaičius teikia atitinkamai geresnę segmentavimo kokybę. Taip pat, ar galioja bendra tendencija, kad algoritmai, naudojantys GAB visada yra geresni už DAB ? Ar tas pats yra tiesa lyginant algoritmų su kelių viršūnių $CART$ medžiais bei $CART$ kelmais silpnuosiuose klasifikatoriuose?



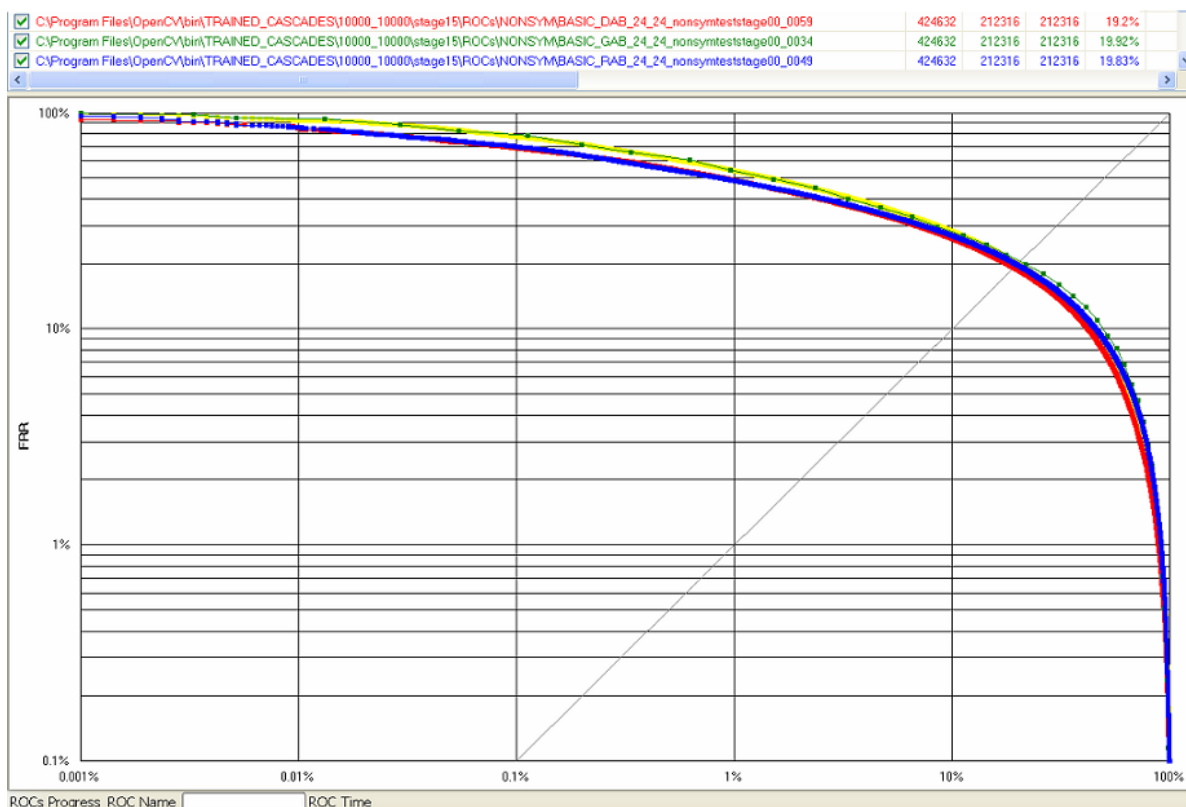
Pav. 13. *OpenCV* „haartraining“ aplikacijai paduodamų mokymo duomenų (veidų ir neveidų) pavyzdžiai.

Bendros visų atliktų testų išvados yra pateiktos žemiau:

1. LAB (*LogitBoost*) algoritmas buvo eliminuotas iš testuojamų „boosting“ algoritmų aibės po (I) testų atlikimo, kadangi visada atrinkdavo dvigubai daugiau požymių f_i palyginus DAB , GAB ir RAB strategijomis, tačiau tuo pačiu neteikė nė kiek geresnių segmentavimo rezultatų. LAB algoritmas yra literatūroje apibrėžiamas kaip algoritmas, tolerantiškas duomenų aibėms su išsišokėliais. Tačiau bendru atveju praktikoje su veidais jis veikė itin prastai.
2. DAB algoritmo versijos naudojančios entropijos ir Gini įverčius požymių h_j slenksčių parinkimui θ_j pasižymėjo kaip daugiausiai požymių f_i (palyginus su visomis kitomis „boosting“ algoritmo režimų ir požymių slenksčių parinkimo kombinacijomis, išskyrus jau atmetą LAB) atrinkančios metodikos. Atrinkdamos daugiau požymių šios strategijos nepasireiškė geresniu tikslumu. Be to, jas siejo gana keistas bendras bruožas: simetriiniu apmokinimo režimu (t.y. nurodžius „-sym“ opcija), šie algoritmai dažnai nuosekliai

atrinkdavo simetrinių požymių poras, o nesimetrinių režimu – tą patį požymį tik su kiek modifikuotais slenksčiais. Šių dviejų *DAB* versijų siūloma netaikyti veidų klasifikatoriaus apmokymui.

3. *DAB* algoritmas, kuris naudoja klasifikavimo klaidų (angl. *misclassification*) minimizavimo strategiją, silpnųjų klasifikatorių h_j požymių slenksčiams θ_j parinkti dažniausiai atrenka daugiau požymių f_i negu *GAB* ir *RAB* algoritmai bei jų modifikacijos. Silpnųjų kelmų, t. y. vienos viršūnės *CART* medžių atveju, didesnis požymių skaičius įtakoja geriausią *DAB* klasifikavimo tikslumą (palyginus su kelmus naudojančiomis *GAB* ir *RAB* versijomis). Tačiau *CART* su trimis arba keturiomis viršūnėmis atveju tai jau ne visada yra tiesa. Nors paskutiniųjų atveju *DAB* dažniausiai vėl gi atrenka daugiausiai požymių, tačiau šio algoritmo *ROC* kreivės jau nebūtinai būna geriausios. Vis dėlto *GAB* ir *RAB* geresnių *ROC* kreivių tendencija nėra stabili.
4. *GAB* algoritmas 85% testavimų rezultate pateikė stipriuosius klasifikatorius H , sudarytus iš 20% mažesnio požymių f_i skaičiaus negu analogiškos *DAB* ir *RAB* versijos. Tačiau toks didelis požymių skaičiaus sumažėjimas neigiamai atsiliepė klasifikatoriaus tikslumui (žr. pav. 14). *GAB* algoritmo atžvilgiu galimos tokios išvados:
 - šis algoritmas teikia mažesnę tikslumą, kai naudoja mažiau požymių f_i , negu kitų algoritmų išeities klasifikatoriai H ;
 - kai šio algoritmo rezultate išgautas klasifikatorius H naudoja tokį patį skaičių požymių f_i , kaip ir kitų algoritmų (*DAB* arba *RAB*) analogiškai apmokinti klasifikatoriai, *GAB* pasižymi geresniu klasifikavimo tikslumu.



Pav. 14. *DAB* (68 požymiai, raudona), *GAB* (34 požymiai, žalia) ir *RAB* (48 požymiai, mėlyna) algoritmais gautų atitinkamų stipriųjų klasifikatorių, naudojančių vienos viršūnės silpnuosius *CART* klasifikatorius (kelmus) segmentacijos *ROC* kreivės.

Šie testavimo rezultatai leidžia galvoti, kad norint turėti bent kiek mažesnę skaičių požymių f_i nei *DAB* generuojamame klasifikatoriuje H , bet apylygį tikslumą, reikia sugriežtinti pačius apmokymo parametrus, t.y. minimalųjį priimtina DR ir maksimalųjį priimtina FAR . Tai gali priversti kelmus h_i naudojančią *GAB* atrinkti kiek daugiau požymių

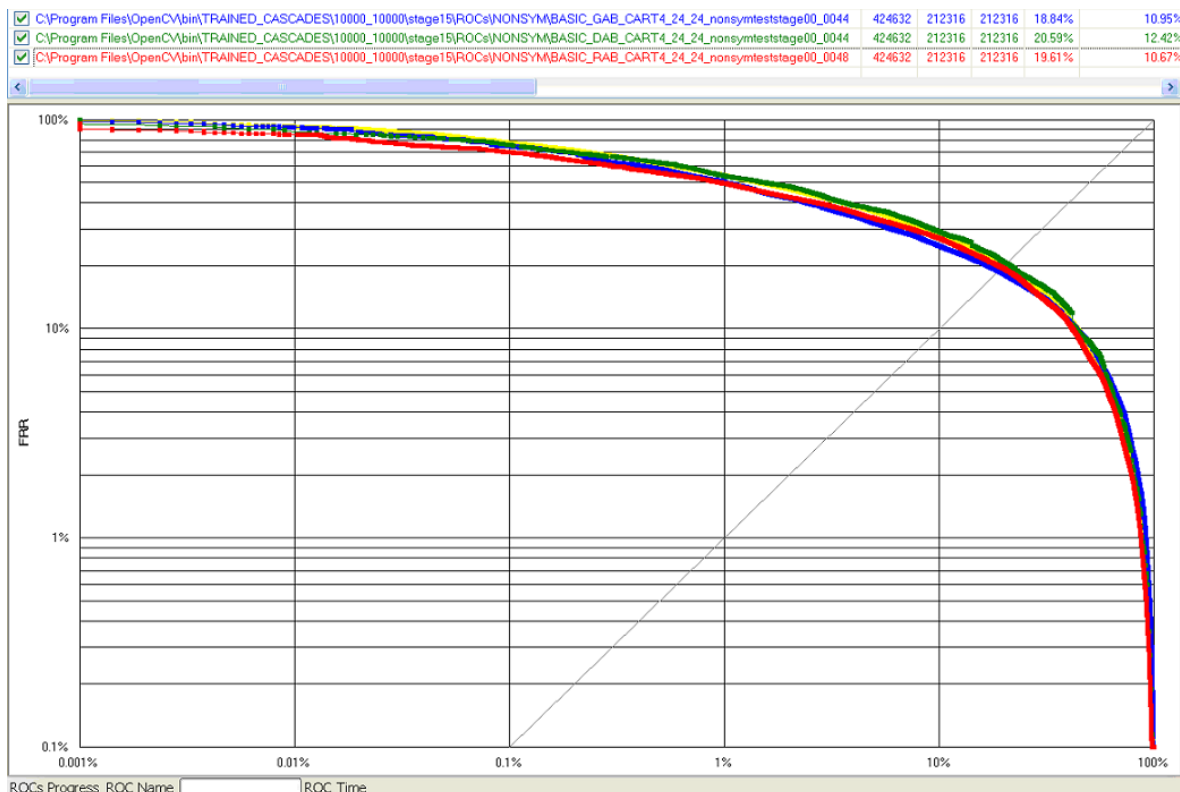
f_i ir išlaikyti klasifikavimo tikslumą pageidaujamuose režimuose. Tačiau čia iškyla nežinomų $d \geq 1$ ir $0 \leq f \leq 1$ koeficientų parinkimo problema ($d * DR_{DAB}$) ir ($f * FAR_{DAB}$) *GAB* mokymo parametrus siekiant išgauti *DAB* analogišką klasifikavimo klaidą.

Kitas būdas išnaudoti *GAB* gebėjimą atrinkti mažai požymių f_i (ir tokiu būdu paspartinti klasifikavimo laiką) – mokinti klasifikatorius, naudojančius daugiau nei vienos viršūnės *CART* silpnuosius klasifikatorius h_t . Nors tokiu atveju *GAB* klasifikatoriai dažniausiai atrenka kiek mažiau požymių negu *DAB* ir *RAB* algoritmų klasifikatoriai, tačiau vis tiek pasižymi gera klasifikavimo kokybe. Verta paminėti, kad *RAB* ir *GAB* stiprieji klasifikatoriai su trijų ir keturių viršūnių *CART* silpnaisiais klasifikatoriais gali veikti labai panašiai, o kartais *RAB* klasifikatorius gali veikti netgi kiek geriau. (žr. pav. 15). Bendru atveju kelių viršūnių *CART* silpnųjų klasifikatorių naudojimas lenkia vienos viršūnės *CART* kelmus naudojančius klasifikatorius tiek tikslumo, tiek našumo (mažesnio požymių skaičiaus) prasme.

5. *RAB* algoritmas su entropijos ir Gini įverčiais požymių f_j slenksčiams θ_j parinkti atrenka mažiausiai požymių iš visų testuojamų algoritmų. „Boosting“ algoritmus pagal atrenkamų požymių skaičių galima surikiuoti didėjimo tvarka taip:

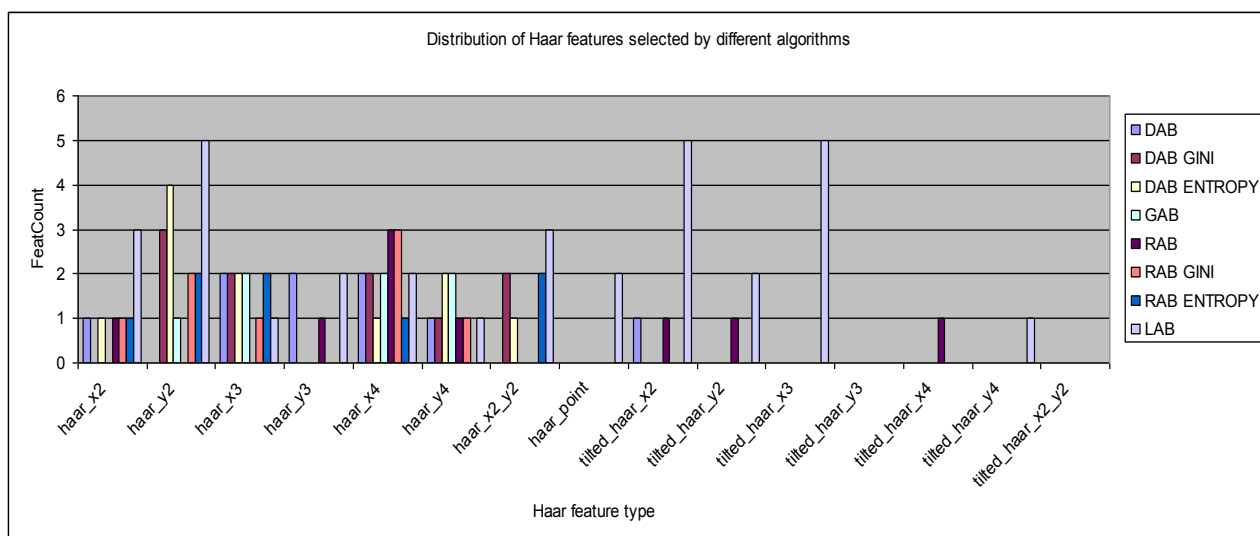
- *RAB* su entropijos / Gini įverčiu;
- *GAB*;
- *RAB* su neteisingų klasifikacijų absoliučios klaidos minimizavimo strategija;
- *DAB*;
- *DAB* su Gini įverčiu (nerekomenduojamas taikyti veidų klasifikavimui);
- *DAB* su entropijos įverčiu (nerekomenduojamas taikyti veidų klasifikavimui);
- *LAB* (nerekomenduojamas taikyti veidų klasifikavimui).

RAB algoritmui su silpnaisiais *CART* kelmis h_t galioja tie patys patarimai tikslumui išsaugoti kaip ir *GAB* atveju. *RAB* ir *GAB* algoritmai tiesiogiai konkuruoja kelių viršūnių *CART* silpnųjų klasifikatorių naudojimo atveju. Tačiau tuo atveju, kai požymių skaičius *RAB* ir *GAB* klasifikatoriuose sutampa, mažiau klaidingų klasifikavimo sprendimų daro *GAB* algoritmas (t.y. *GAB* ROC kreivė yra žemesnė).



Pav. 15. *DAB* (48 požymiai, žalia), *GAB* (44 požymiai, mėlyna) ir *RAB* (48 požymiai, raudona) algoritmais gautų atitinkamų stipriųjų klasifikatorių, naudojančių keturių viršūnių silpnuosius *CART* klasifikatorius ROC kreivės.

6. Visi minėti testai buvo atliekami naudojant tiek bazinę (*OpenCV* įvardijama kaip „BASIC“) požymių f_j aibę [VJ01a], tiek išplėstinę („CORE“), tiek pilną („ALL“), kur pastaroji įtraukia ir 45 laipsniais pasuktų požymių naudojimą, kurie gali praversti pasuktų veidų segmentacijai vaizduose [LKP02]. Tačiau pastariesiems apskaičiuoti reikia dar vienos tarpinės vaizdo reprezentacijos – pasukto integralinio paveiksluko (angl. *tilted integral image*) [LKP02]. Tai reikalauja papildomų atminties ir skaičiavimo mokymo bei segmentacijos metu resursų, todėl buvo įdomu išsiaiškinti, ar iš tikrųjų tokių požymių taikymas, ypač priešakiniais veidams, duoda kokį nors pagerėjimą segmentacijos atžvilgiu. Testai parodė, kad priešakiniu veidų segmentacijos atveju požymių aibės praplėtimas nuo bazinės iki išplėstinės įtakoja didesnį klasifikavimo rezultatų pagerėjimą negu perėjimas nuo išplėstinės prie pilnos požymių aibės. Tai reiškia, kad pasukti požymiai yra atrenkami labai retai, todėl dėl papildomai jiems palaikyti reikalingų skaičiavimų jų naudoti (bent jau priešakiniais veidams) neverta (žr. pav. 16).



Pav. 16. *DAB*, *GAB*, *RAB* ir *LAB* algoritmais išrenkamų požymių pasiskirstymas (kai silpnasis klasifikatorius sudarytas iš *CART* kelmo). Pasukti požymiai pažymėti prefiksu „tilted_“, nepasukti – be prefikso. Tokių požymių priešakinių veidų stipriuosiuose klasifikatoriuose būna labai mažai.

7. Praktinės analizės eigoje buvo atlikti modelio dydžio testai (nuo 18 x 18 iki 24 x 24 taškelių), bandant nustatyti optimalų veidų / neveidų mokymo aibės paveikslukų dydį. 22 x 22 dydžio langas gerai aproksimuoja 24 x 24 langą ir sumažina požymių aibę iki 30%, kas pagreitina apmokymo procesą. Naudojant mažesnius, pvz., 18 x 18 paveikslukus, patariama geriau naudoti didesnių paveikslukų 22 x 22 apkarpytą (angl. *cropping*) strategiją negu mastelio keitimą (angl. *scaling*). Svarbiausia, pernelyg stipriai neapkarpyti viršutinės kaktos ir antakių srities. Ji labai svarbi klasifikavimui. Smakro svarba yra mažesnė.
8. Taip pat patariama nenaudoti *OpenCV* „haartraining“ aplikacijos „-sym“ opcijos veidams. Taip išlaikant beveik tą patį tikslumą atrenkama žymiai mažiau (kartais dvigubai) požymių f_i . Be šios opcijos padidėja nagrinėjamų požymių f_j aibė, t. y. sulėtėja mokymas, tačiau rezultate turimas greitesnis klasifikatorius H .
9. Minėtos aplikacijos „-weighttrimming“ opcijos argumentu siūloma naudoti 0.95 reikšmę. Tada, pvz., *DAB* mokymosi algoritmas vidutiniškai naudoja 75 – 79% mokymo egzempliorių svorių ir pakankamai gerai aproksimuoja visus svorius naudojantį algoritimą (t. y. su argumentu lygiu 1.0). Su argumentu lygiu 0.90 pilno apmokymo aproksimacijos pasiekti jau nepavyksta – gauti stiprieji klasifikatoriai H silpnėja.

2.3. „Boosting“ praktinio nagrinėjimo išvados

Apibendrinant, galima tikėtis, kad *GAB* algoritmui sėkmingai parinkus griežtesnius nei tikimasi minimalaus *DR* ir maksimalaus *FAR* mokymo parametų įverčius, galima sukurti ne tik spartesnį, bet ir tikslesnį klasifikatorių *H* nei su *RAB* ir *DAB*. (Deja, neaišku, kaip tokius įverčius parinkti sėkmingai iš pirmo karto, išvengiant pakartotinio ilgo mokymo proceso.) *RAB* algoritmas yra kiek prastesnis, bet nelabai atitrūksta nuo *GAB* segmentacijos kokybe, ypač naudojant kelių viršūnių *CART* silpnuosius klasifikatorius h_t . Kelių viršūnių *CART* silpnųjų klasifikatorių naudojimas visada teikia geresnius rezultatus kokybės prasme nei kelmų taikymas. Tokia kaskada išliks sparti (palyginus su kelmų atveju), jeigu bendras atrenkamų požymių f_t stipriausioje klasifikatoriuje skaičius bus mažesnis, kadangi sprendimų medžio struktūros silpnajame klasifikatoriuje gali kiek lėtinti greitį segmentacijos metu (ypatingai didesnio, pvz., keturių, viršūnių skaičiaus atveju). *DAB* algoritmas daugelio silpnųjų klasifikatorių tipų (*CART* viršūnių skaičiaus) atveju atrenka daugiau požymių, tačiau atitinkamai pasižymi geresniu klasifikavimo tikslumu, kuris (remiantis atliktais testais) yra pastovus *DAB* bruožas fiksuotiems mokymo ir testavimo duomenims.

Reziumuojant, žemiau yra pateikiami patariamų naudoti algoritmų kombinacijų sąrašai.

- Silpnųjų *CART* medžių klasifikatorių h_t , kai šie sudaryti iš dviejų – keturių požymių f_t viršūnių, atveju siūloma naudoti tokius algoritmus (prioriteto mažėjimo tvarka):
 - *GAB*,
 - *RAB*,
 - *DAB*.
- Silpnųjų *CART* kelmų klasifikatorių h_t , kai šie sudaryti iš vieno požymio viršūnės f_t atveju siūloma naudoti tokius algoritmus (prioriteto mažėjimo tvarka):
 - *DAB*,
 - *GAB*,
 - *RAB*.

LAB naudojimas neatrodo esąs efektyvus algoritmas veidų segmentacijai spręsti dėl itin didelio požymių f_t skaičiaus klasifikatoriuje, kuris lemia tiek žymiai ilgesnį mokymo, tiek segmentacijos laiką neteikiant geresnės veidų suradimo kokybės.

Kaip jau buvo minėta, taip pat nepatariama taikyti mokymo bei paieškos laiką lėtinančius pasuktus požymius priešakiniais veidams nuo neveidų atskirti, nes tokie požymiai reikalauja pasukto integralinio atvaizdo apskaičiavimo. Kur kas prasmingiau yra naudoti nepasuktų požymių didesnę įvairovę, kuri turi bene didžiausią teigiamą įtaką segmentacijos kokybei kartu išlaikant gerą balansą su segmentacijos našumu. Deja, pastaruoju atveju mokymo laikas taip pat ilgėja.

Remiantis visomis aukščiau išvardintomis teorinio ir praktinio nagrinėjimo išvadomis, veidų segmentacijos algoritmo skaitmeniniuose vaizduose realizacijoje buvo pritaikytas *DAB* kaskadinio klasifikatoriaus algoritmas su stipriausiais klasifikatoriais H_t sudarytais iš *CART* kelmų (vienos viršūnės f_t) silpnųjų klasifikatorių h_t . Kelmų silpnieji klasifikatoriai buvo pasirinkti dėl savo paprastumo realizuojant klasifikatorių išvengiant segmentaciją lėtinančių papildomų vidinių medžių struktūrų. *DAB* pasirinktas dėl savo stabilumo gaunamos kaskados kokybės atžvilgiu, turint kiek lėtesnį už konkurencingos tikslumu *GAB* metodikos greitį. Be to, priešingai *GAB*, nereikia dirbtinai keisti mokymo parametų (mažinti *FAR* ir didinti *DR*) siekiant išgauti kokybės rezultatus, kurių tikimasi.

2.4. Binarizacijos įtaka veidų segmentacijai *DAB* metodika

Šio tiriamojo darbo eigoje taip pat buvo praktiškai patikrinta skaitmeninio vaizdo binarizacijos įtaką tolimesnės veidų segmentacijos rezultatams. Šiam tikslui buvo realizuotas Flojdo-Steinbergo binarizacijos (angl. *Floyd-Steinberg dithering*, toliau – *FS*) algoritmas.

FS binarizacijos algoritmas [WFS09] dažniausiai yra taikomas konvertuojant skaitmeninį vaizdą į tam tikrą formatą su ribotu spalvų arba pilkumo lygių skaičiumi. Algoritmas išsklaido

taškelių kvantavimo klaidą jo kaimynystėje esantiems taškeliams, remiantis žemiau pateiktu paskirstymu.

$$\frac{1}{16} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 7 \\ 3 & 5 & 1 \end{bmatrix}$$



Pav. 17. *FS* binarizavimo algoritmu apdorotas vaizdas (kairėje) ir originalusis vaizdas (dešinėje).

Algoritmas pavieniui kvantuoja taškelių reikšmes, skenuodamas vaizdą iš kairės į dešinę ir iš viršaus į apačią. Kaip jau buvo minėta, kiekvieno tokio apdorojimo metu kvantavimo paklaida paskirstoma kaimynams, nė kiek neveikiant tų taškelių reikšmių, kurios jau buvo kvantuotos. Taigi, jeigu ankstesnis taškelis buvo suapvalintas į mažesnę pusę, tai yra didelė tikimybė, kad kitas bus suapvalintas į didesnę. Tokiu būdu išgaunama vidutinė kvantavimo paklaida lygi 0. Šio algoritmo sklaidos koeficientai užtikrina, kad jeigu originalios taškelių reikšmės yra lygiai „pusiaukelėje“ nuo galimų spalvų, tai rezultate gaunamas šachmatų lentos šablonas. Algoritmo taisyklingam veikimui reikia naudoti kvantavimo klaidą tam tikru tikslumu, pakankamu apvalinimo klaidoms išvengti.

Reikia paminėti, kad šiame darbe įgyvendintoje *FS* realizacijoje yra naudojamas „vingiuojantis (gyvatės)“ skenavimas, kuriame lyginės ir nelyginės eilutės skenuojamas skirtingomis kryptimis. Šiuo algoritmu buvo apdorojami skaitmeniniai aukščiau minėtų veidų duomenų bazių vaizdai prieš taikant jiems „*boosting*“ metodika apmokintą kaskadinį klasifikatorių, skirtą veidams segmentuoti pilkuose (angl. *grayscale*) vaizduose. Toliau buvo lyginama to paties klasifikatoriaus atsakų į pilkuosius ir binarizuotus paveikslukus statistika.

Rezultatai gali būti apibendrinti taip: klasifikatorius be jokių jo modifikacijų veikia ant didesnių mastelių veidų (kurių atstumas tarp akių apytiksliai > 50 taškelių), tačiau prasčiau veikia ant mažo mastelio veidų. Jeigu apmokinto klasifikatoriaus vidutinis teisingo veidų segmentavimo pilkuose skaitmeniniuose vaizduose dažnis *DR* yra apie 99,5%, tai tuose pačiuose binarizuotose vaizduose *DR* bus lygus 99,15%. Kai apmokinto klasifikatoriaus vidutinis teisingo veidų segmentavimo pilkuose skaitmeniniuose vaizduose dažnis *DR* siekia 98,67%, tai atitinkamuose binarizuotose vaizduose *DR* lygus 96,9%. T.y. binarizuotų vaizdų atveju

segmentavimo DR gali sumažėti 0,35 – 1,77 %. Kadangi binarizuotų vaizdų segmentavimo atveju mažėja DR , t. y. didėja FRR , tai dažniausiai kiek sumažėja FAR . Tačiau, jeigu testuojamas klasifikatorius naudoja požymius, kurių atsakams naudojamas ne vienas slenkstis, o sudėtingesnė intervalų ar tikimybinė sistema, tuomet FAR gali kiek išaugti dėl dalies pilkumo informacijos praradimo, kuri gali būti svarbi sudėtingesnio klasifikatoriaus papildomiems apribojimams

Dėl tos pačios informacijos praradimo problemos šis algoritmas prastai veikia ant mažo mastelio veidų. Tai galima pamatyti iš klasifikatoriui mokinti naudotų veidų ir jų binarizuotų versijų vidurkio (žr. pav.18), kuris, palyginus su originaliuoju kenčia, nuo daugybės iškreipimų. Todėl mokymo algoritmo paleidimas ant binarizuotų paveiksliukų yra beprasmiškas (mokymosi algoritmas nesugeba surasti binariniais veidams požymių, tenkinančių duotus minimalaus DR ir maksimalaus FAR kriterijus).

Tačiau šią idėją galima pritaikyti itin dideliems skaitmeniniams vaizdams segmentuoti (siekiant išvengti integralinio paveikslo reikšmių perpildymo (angl. *overflow*) ir / arba klasifikatoriaus atsako skaičiavimams pagreitinti, perteikiant silpnųjų klasifikatorių h_t svorius α_t stipriausiam klasifikatoriuje naudojant bitinų manipuliacijų operacijas.



Pav. 18. Kairėje – *AdaBoost* mokymo metodui pateikiamų veidų klasės mokymo duomenų vidurkis. Dešinėje – tų pačių veidų, apdorotų *FS* binarizavimo metodu, vidurkis. *FS* labai iškreipia veidų reprezentatyvią aibę, tokiu būdu neleidamas sukurti efektyviai veidų ir neveidų klases atskiriančio klasifikatoriaus.

3. Veidų segmentacijos sistemos įgyvendinimas ir tyrimas

Šiame skyriuje yra aprašoma praktiškai įgyvendintą priešakinių veidų segmentacijos skaitmeniniuose vaizduose sistema bei jos sudedamųjų dalių algoritmai. Taip pat pateikta informacija apie sistemos mokymui, derinimui bei testavimui naudotas viešai prieinamų ir plačiai žinomų veidų duomenų bazių rinkinius. Kartu su segmentacijos rezultatais pateikiamos gairės tolimesniam tyrinėjimui realizacijos tikslumui ir / arba našumui pagerinti.

3.1. Mokymo ir testavimo veidų duomenų bazės

Pirminiams praktiniams tyrimams nuspręsta taikyti viešai prieinamas, dažniausiai su veidų biometrija susijusių algoritmų ir straipsnių autoriais naudojamas apažinimo tduomenų bazes ir joms taikomus testavimo protokolus. Šios duomenų bazės kartu su vaizdų duomenimis turi etaloninių vaizdų galerijos (angl. *referente set*, *gallery set*) bei tiriamų kandidatų aibės (angl. *candidate set*, *probe set*) metaduomenų aprašus, kurie įtraukiami į verifikacijos ir / arba identifikacijos sistemų minėtus patikimumo įvertinimo protokolus.

Galerijos rinkinys aprašo individų vaizdus, kurie yra dažniausiai saugomi (ang. *enrolled*) biometrinės veidų sistemos duomenų bazėje. Šie aprioriškai žinomi duomenys (jeigu jų sankirta su kandidatų rinkiniais yra tuščia) dar gali būti vadinami mokymo rinkiniais, kadangi gali būti naudojami papildomam sistemos automatiniam apsimokymui prieš priimant identifikavimo

(1:N) arba verifikavimo (1:1) užklausas. Kandidatų rinkiniai, priešingai – reprezentuoja nežinomų galimų užklausų duomenų aibes. Todėl jie dar vadinami užklausų rinkiniais (angl. *query set*).

Sistemos tobulinimo ir testavimo metu buvo naudojamos dvi biometrijos pasaulyje gerai žinomos FERET pilkųjų vaizdų (angl. *The Facial Recognition Technology (FERET) Database Grayscale*,) ir NIST FRGC (angl. *National Institute of Standards and Technology Facial Recognition Grand Challenge*) duomenų bazės. Jų rinkinių aprašus ir testavimo protokolus galima rasti vadovaujantis nuorodomis, kurios pateiktos šio darbo priede Nr. 1.

Žemiau pateikti šiame darbe naudojami FERET duomenų bazės kontroliuojamos sąlygomis gautų priešakinių veidų pilkųjų vaizdų rinkiniai bei jų sutrumpinti žymėjimai:

- *FERET_FA* – etaloninių subjektų galerijos rinkinys FA;
- *FERET_FB* – analogiškas *FERET_FA* kandidatų vaizdų rinkinys su kitokios veido išraiškos subjektų atvaizdais;
- *FERET_DUP1* – 3 metų laikotarpyje (nuo FA) fiksuotų kandidatų vaizdų rinkinys;
- *FERET_DUP2* – nuo 18 iki 36 mėnesių laikotarpyje (nuo FA) fiksuotų kandidatų vaizdų rinkinys (*FERET_DUP1* poaibis).

Žemiau pateikti šiame darbe naudojami FRGC duomenų bazės gautų priešakinių veidų pilkųjų vaizdų rinkiniai bei jų sutrumpinti žymėjimai:

- *FRGC_TRAINING* – FRGC mokymo vaizdų rinkinys;
- *FRGC_EXP1* – FRGC protokolo pirmajam eksperimentui priklausantis kontroliuojamomis sąlygomis fiksuotų subjektų galerijos ir kandidatų vaizdų rinkinys;
- *FRGC_EXP4* – FRGC protokolo ketvirtajam eksperimentui priklausantis nekontroliuojamomis sąlygomis fiksuotų subjektų galerijos ir kandidatų vaizdų rinkinys;

FERET_FA ir / arba *FRGC_TRAIN* rinkiniai naudoti segmentacijos algoritmo realizacijos mokymo ir derinimo fazėje. Kiti rinkiniai taikyti sistemos testavimui.

Siekiant atlikti įgyvendintos veidų sistemos lyginamąją analizę, testavimo duomenis buvo papildyti rinkiniais iš populiariausių, tik veidų segmentacijai skirtų duomenų bazių – MIT/CMU ir BioID. Pirmoji yra seniausia žinoma veidų segmentacijos testavimo duomenų bazė, kurios duomenų rinkiniai nėra taisyklingi realios veidų segmentacijos sistemos vertinimo prasme dėl įtrauktų nupaišytų ir nežemiškos prigimties filmų herojų veidų, itin menkos rezoliucijos skenuotų laikraščių paveikslukų, maksimalių veidų posūkių plokštumoje ir erdvėje neapibrėžtumo rinkiniams. Antroji duomenų bazė yra kur kas naujesnė ir realesnės kokybės. Ji yra sudaryta iš prieš kamera sėdinčių žmonių, kalbančių ir bekeičiančių veidų išraiškas vaizdų sekų. Abiejų duomenų bazių nuodugnesnius aprašus ir testavimo protokolus galima rasti vadovaujantis nuorodomis, kurios pateiktos šio darbo priede Nr. 1.

Žemiau pateikti šiame darbe naudojami MIT/CMU duomenų bazės priešakinių veidų pilkųjų vaizdų rinkiniai bei jų sutrumpinti žymėjimai:

- *MIT/CMU*– MIT/CMU priešakinių veidų rinkinys, sudarytas apjungus A, B ir C rinkinius;
- *MIT* – MIT/CMU priešakinių veidų rinkinys, sudarytas iš seniausio B rinkinio;
- *MIT80* – MIT/CMU aštuoniasdešimties vaizdų rinkinio poaibis, naudojamas [Liu03] priešakiniams nepasisukusiems veidams testuoti.

Žemiau pateikti šiame darbe naudojamas BioID duomenų bazės priešakinių veidų pilkųjų vaizdų rinkinis bei jo sutrumpintas žymėjimas:

- *BIOID* – BioID duomenų bazės vaizdų rinkinys.

3.2. Pagrindinės realizuotos sistemos komponentės

Realizuota sistema yra sudaryta iš dviejų pagrindinių komponentių:

- „*boosting*“ metodika (žr. sk. 1.4) grindžiamo pilkųjų paveiksliukų kaskadinio klasifikatoriaus, kuris naudojamas pirminiams veidų kandidatams surasti;
- naiviojo(-iųjų) Bajeso klasifikatoriaus(-ių), kuris(-ie) taikomi iš kaskados gautų kandidatų pozicijoms patikslinti ir / arba netikriems veidų kandidatams eliminuoti.

Komponentės sudaro algoritmo žingsnius, nuosekliai taikomus veidų segmentacijai vaizduose. Kiekviena iš komponentių ir joms įgyvendinti reikalingų algoritmų aprašymai pateikti žemiau.

3.2.1. DAB kaskadinis klasifikatorius

Remiantis ankstesniuose skyriuose pateikta veidų segmentacijos algoritmus aprašančios mokslinės literatūros apžvalga bei analize (žr. sk. 1), taip pat „*boosting*“ metodikos praktine analize (žr. sk. 2) buvo nuspręsta veidų segmentacijai naudoti *Discrete AdaBoost (DAB)* kaskadinio klasifikatoriaus realizaciją. „*Boosting*“ metodika grindžiama segmentacija įgalina itin sparčiai lokalizuoti vaizduose įvairius objektus (tarp jų - veidus), sprendžiant „objektų“ ir „neobjektų“ klasifikavimo problemą. Lygi šių dienų tai yra labiausiai paplitęs metodas būtent dėl nepranokstamos spartos tuo pačiu išlaikant daugelį sistemų tenkinančią teisingų objektų suradimo statistiką. Konkretaus *DAB* „*boosting*“ modifikacijos pasirinkimas yra pagrįstas ankstesniame skyriuje 2.3.

Mokymo metu buvo pritaikytos 3.1. skyriuje aprašytų duomenų bazių *FERET_FA* ir *FRGC_TRAINING* etaloniniai veidų vaizdų rinkiniai. Šie rinkiniai turi atitinkamai 1196 ir 12776 vaizdų. Siekiant išgauti geresnius rezultatus, buvo bandoma padaryti mokymo imtį didesnę bei reprezentatyvesnę priešakinių veidų klasės populiacijos atžvilgiu. Tuo tikslu kiekvienam iš 13972 mokymo vaizdų buvo pritaikytos atsitiktinės posūkio transformacijos, leidžiant absoliučiai veido posūkio plokštumoje reikšmei varijuoti uždarajame intervale [0; 10]. Taip buvo gauti 55888 veidų klasės mokymo vaizdai geometriškai normalizuoti į 24 x 24 dydžio mokymo vaizdus. Veidų mokymo aibės vaizdų pavyzdžiai pateikti žemiau pav. 19. Neveidų klasės reprezentacinei aibei naudoti įvairūs vaizdai (kuriuose nėra veidų) iš interneto šaltinių. Naudotas vadinamasis simetrinis mokymo duomenų atžvilgiu variantas, kai veidų ir neveidų klasių mokymo aibių elementų skaičiaus yra vienodas. Nors teoriškai skirtingas elementų skaičius yra kompensuojamas *AdaBoost* algoritmo viduje svorių $w_{t,i}$ dėka (žr. sk. 1.4, schemą 1), praktiniai tyrimai parodė, kad didelis disbalansas tarp veidų ir neveidų mokymo aibių dydžių (ypatingai kai šie yra nedideli) neigiamai atsiliepia mokymo rezultatams.



Pav. 19. *DAB* veidų klasifikatoriui apmokinti naudotų veidų klasės vaizdų pavyzdžiai. Mokymo bazė sugeneruota iš *FERET_FA* ir *FRGC_TRAINING* veidų rinkinių.

Remiantis praktinės analizės (žr. sk. 2.3) ir kitų praktinių tyrinėjimų patirtimi, veidų klasifikatoriui apmokinti buvo naudojamas patobulintas klasikinis mokymo algoritmas su paplėsta nepasuktų požymių f_j tipų aibe. Omenyje turimas ne *OpenCV* kompiuterinės regos mokymo programos „*EXTENDED*“ požymių aibės varianto analogas, kuris tik padidina generuojamų požymių dydžių įvairovę. Buvo pridėti iš pagrindų kitokio tipo Haar požymiai analogiški nesiliečiančių stačiakampių požymiams, aprašytiems apžvalgos skyriuje minėtuose S. Z. Li ir kt. [LZS+02, LZ04] straipsniuose. Praplėtimo rezultate silpniesiems *CART* kelmų klasifikatoriams h_j konstruoti mokymo procese buvo panaudota apie 900000 Haar požymių f_i . su neteisingo klasifikavimo (angl. *misclassification*) optimalaus slenksčio θ_j parinkimo strategija. Virš 50% naudojamų požymių priklausė paplėstųjų požymių aibe su nesiliečiančių sričių stačiakampiais.

Ilgo *DAB* trijų savaitių ant kelių gijų paleisto klasifikatoriaus mokymo proceso rezultate buvo gautas veidų kaskadinis klasifikatorius sudarytas iš 21 stipriųjų klasifikatorių H pakopų, kurios sumoje turi 590 silpnųjų *CART* kelmų klasifikatorių h_i , t.y. pavienių Haar požymių f_i . Kaip ir tikėtasi, silpnųjų klasifikatorių h_i skaičius stipriuosiuose klasifikatoriuose H tendencingai auga, t.y. didėjant pakopos numeriui. Pirmųjų dešimties pakopų stipriuosiuose klasifikatoriuose vidutinis požymių skaičius yra 9, tuo tarpu vienuolikoje paskutinių pakopų vidutinis skaičius yra 43. Chaotiško požymių skaičiaus pakopose svyravimo nebuvimas ir tendencijas jų didėjimas yra efektyvaus mokymosi požymis, kuris nurodo kad su kiekviena pakopa mokymo problema darėsi vis netrivialesnė – kaskados pakopos klasifikatorius mokėsi atskirti labiau į veidus panašius neveidus. Daugiau nei 21 pakopa nebuvo mokinama dėl itin ilgo mokymosi proceso (apie mėnesį) bei grėšiančio pernelyg didelio prisitaikymo prie mokymo duomenų (angl. *overfitting*) problemos

P. Viola ir M. Jones [VJ01b] praplėsta metodika apmokinto *DAB* kaskadinio klasifikatoriaus veidų segmentacijos vaizduose rezultatai pateikti toliau išdėstyta sistemos testavimo rezultatų skyriuje (žr. sk. 3.3).

3.2.2. Naivusis Bajeso klasifikatorius

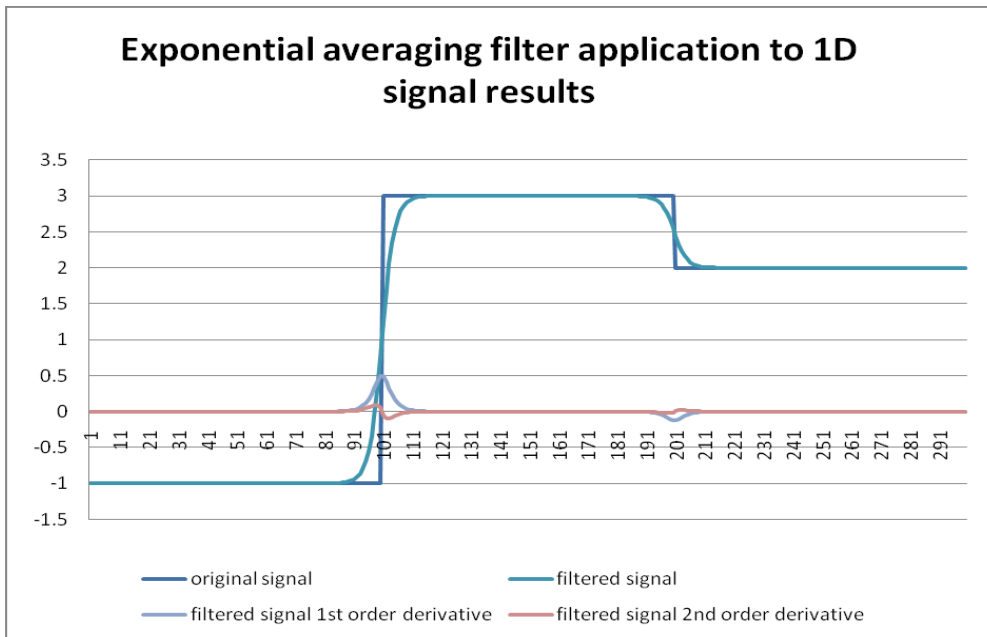
Iš toliau pateiktų testavimo rezultatų (žr. sk. 3.3, lentelės 2 ir 3) matyti, kad P. Viola ir M. Jones [VJ01b] kombinuota su S. Z. Li ir kt. [LZS+02, LZ04] analogiškais požymiais metodu pagrįstas kaskadinis klasifikatorius pasižymi gerais veidų suradimo patenkinamos rezoliucijos vaizduose rodikliais, tačiau ne itin tiksliai lokalizuoja surastus veidus tikrųjų veidų atžvilgiu. Taip pat egzistuoja netesingai surastų veidų, kurie iš tikrųjų nėra veidai problema. Šioms problemoms spręsti buvo pritaikytas papildomo žingsnis naudojantis intuityvų naiviojo Bajeso (*NB*) klasifikavimo algoritmą. Šio algoritmo bendras apibrėžimas kartu su sėkmingo taikymo rašytinių skaitmenų, gestų ir numerių atpažinimui bei veidų biometrijai (veidų segmentacijai, emocijų ir orientacijos nustatymui) pavyzdžiai yra pateikti skyriuje 1.5. *NB* klasifikatorius buvo apmokomas ir taikomas A. Basčio [Bas10a] pasiūlyto spartaus glodaus simetrinio eksponentinio filtro rezultate gautiems požymiams, pvz., glodinto veido vaizdo gradiento krypčių kvantuotoms reikšmėms klasifikuoti.

A. Basčio pasiūlytas glodus eksponentinis filtras (žr. formulę (9)) suglodina vaizdo pilkumo lygmenų reikšmes eliminuodamas lokalių išsišokėlių / ekstremumų triukšmingą informaciją. Šio filtro privalumai – eksponentinės prigimties dėka išgaunama efektyvi realizacija ir galimybė per du vienmačio filtruojamo signalo praėjimus vienu metu apskaičiuoti glodinto vaizdo bei jo pirmos ir antros eilės išvestinių reikšmės. Dvimačių atveju filtras yra vienmačių filtrų sandauga (žr. formules (10) ir (11)). T.y. norint atlikti vaizdo filtravimą minėtu dvimačiu eksponentiniu glodžiuoju filtru, reikia pritaikyti vienmatį filtrą eilutėms, o vėliau – gautojo tarpinio filmavimo rezultato stulpeliams (arba atvirkščiai). Filtro taikymo vienmačiams ir dvimačiams veidų vaizdų signalams pavyzdžiai pateikti pav. 20 ir pav. 21.

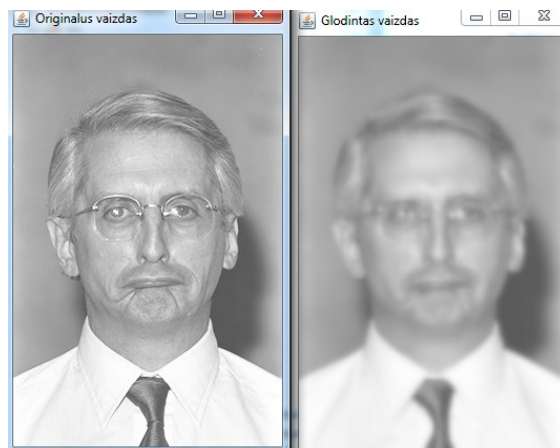
$$v_{\sigma}(x) = \left(\frac{|x|}{\sigma} \right) e^{-\frac{|x|}{\sigma}}, x \in -\infty \dots \infty \quad (9)$$

$$v_{\sigma}(x, y) = \left(\frac{|x|}{\sigma} + \frac{|y|}{\sigma} \right) e^{-\frac{|x|+|y|}{\sigma}}, x, y \in -\infty \dots \infty \quad (10)$$

$$v_{\sigma}(\xi, y) = v_{\sigma}(\xi) v_{\sigma}(y) \quad (11)$$



Pav. 20. (9) formulėje aprašyto vienmačio glodaus eksponentinio filtro taikymo signalui pavyzdys ($\sigma = 2$).



Pav. 21. (10) formulėje aprašyto dvimačio glodaus eksponentinio filtro taikymo veidų vaizdams pavyzdys ($\sigma = 1.5$).

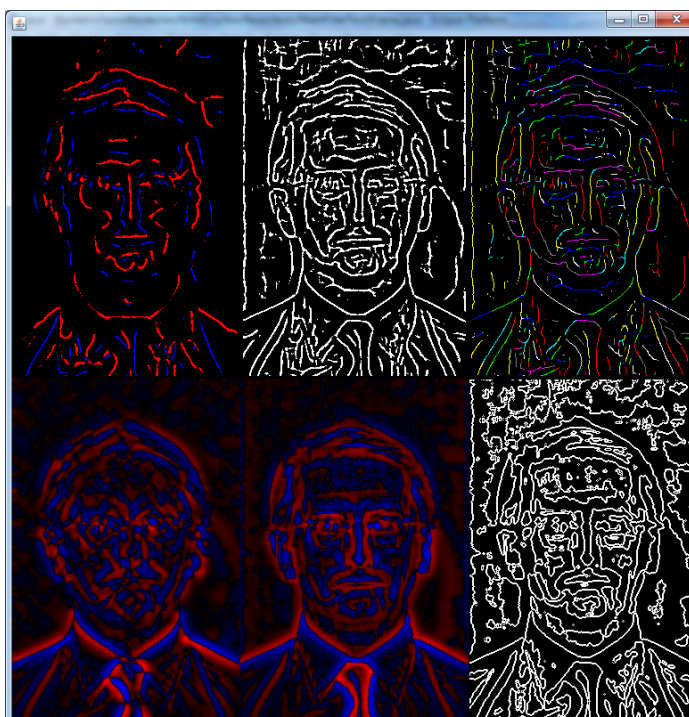
Filtro parametras σ apibrėžia filtro spindulį. Kuo šis parametras didesnis, tuo labiau suglodinimas vaizdas, ir atvirkščiai – kuo σ mažesnis, tuo mažiau vidurkinamas vaizdas. Filtro teisingas pritaikymas visam vaizdai be apriorinės informacijos apie galimą veido mastelį nėra galimas. Neteisingai parinkus pernelyg mažą σ , vaizdo filtravimas gali neturėti beveik jokio efekto. Parinkus neadekvačiai didelę segmentuojamų objektų atžvilgiu σ parametro reikšmę, dar blogiau – gali būti eliminuojami ne tik triukšmai, bet ir naudinga tiksliai segmentacijai būtina tekstūros informacija. Taigi, efektyviam triukšmų eliminavimui bei pirmos ir antros eilės išvestinėms apskaičiuoti spartusis filtras taikomas realizuotos veidų segmentacijos sistemos antrajame žingsnyje, panaudojant *DAB* kaskada gražintų kandidatinių sričių apytikslio mastelio informaciją. Tada veidai geometriškai normalizuojami į sukonstruoto *NB* klasifikatoriaus veido modelio dydžio langą, kuris įgalina taikyti fiksuotą filtravimo parametą (šiam darbe realizuoto ir toliau šiame skyriuje aprašyto *NB* klasifikatoriaus atveju optimalus $\sigma \approx 2$). Tokia strategija yra racionali ir sparti, skirtingai nuo kitų algoritmų autorių siūlymų taikyti tam tikrus normalizacijos būdus visam vaizdai prieš leidžiant kaskadą.

Kaip buvo minėta skyriuje 1.5, šiame darbe buvo siekiama klasifikuoti fiksuoto modelio dydžio vaizdų sričių (*DAB* kandidatų) duomenis X į vieną iš dviejų veidų (Y_1) ir neveidų (Y_2) klasių. Duomenys X yra dvimatis taškelių masyvas dydžio $d_x \times d_y = n$, kurį galima įsivaizduoti kaip vektorių $X = \{x_1, x_2, \dots, x_n\}$. *NB* klasifikatoriaus apibrėžimas yra pateiktas sk. 1.5. formulėje (8). Tačiau siekiant paspartinti klasifikavimo procesą buvo vengiama daugybės operacijos tikimybės (7) apskaičiavimo metu. Tam tikslui klasifikacijos metu buvo apskaičiuojama ne sąlyginė duomenų priklausomybės klasei Y_i tikimybė $P(Y_i|X)$, o jos natūralusis logaritmas $\ln(P(Y_i|X))$. Tikimybės logaritmo apskaičiavimas leido pakeisti taškelių sandaugos operaciją suma. Normuojančio $P(X)$ daliklio reikšmė nebuvo įtraukiama į skaičiavimus. Taip pat, $P(x_k|Y_i)$ nulinėms tikimybėms buvo priskiriama labai maža ε nenulinė reikšmė, kad išvengtų logaritminių įverčių apskaičiavimo problemų. Vienetinėms tikimybėms buvo pagal analogiją priskiriamos $(1 - \varepsilon)$ reikšmės. Galutinis praktinėje realizacijoje taikomas *NB* klasifikatoriaus apibrėžimas pateiktas formulėje (12).

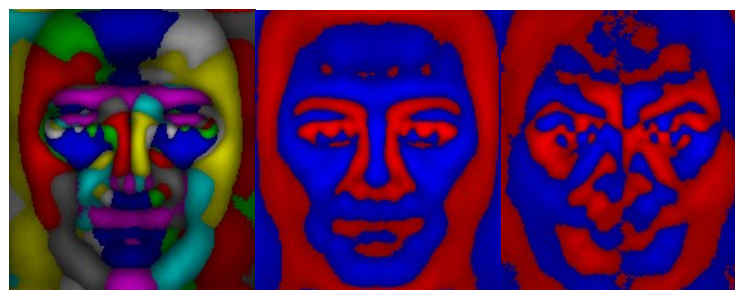
$$\text{Classifier}_{NB}(x_1, x_2, \dots, x_n) = \max_c \left(\ln \left(\prod_{k=1}^n P(x_k | Y_c) \right) + \ln P(Y_c) - \ln P(x_1, x_2, \dots, x_n) \right) \quad (12)$$

Skaitmenų ar kitų rašomųjų simbolių vaizdams klasifikuoti naudojami *NB* klasifikatoriai dažniausiai yra apmokinami and binarinių vaizdų, kuriuose kiekvienas taškelis atitinkamai gali įgyti 0 arba 1 reikšmes (žr. [Abb10]). Skaitmenų klasifikacijos uždavinys yra paprastesnis nei veidų vaizdų, kadangi simbolių vaizduose dažniausiai dominuoja kelios pagrindinės šviesioji (fono) ir tamsioji (simbolio kontūro) sritys, lengvai atskiriamos taikant pastovaus binarizacijos slenksčio metodą. Veidų atveju vaizdų tekstūros yra kur kas informatyvesnės ir sudėtingesnės. Pastovaus ar adaptyvaus slenksčio binarizacijos taikymas nėra tinkama strategija veidams, kadangi prarandama didelė dalis plačią intraklasinę variaciją turinčių veidų tekstūros informaciją.

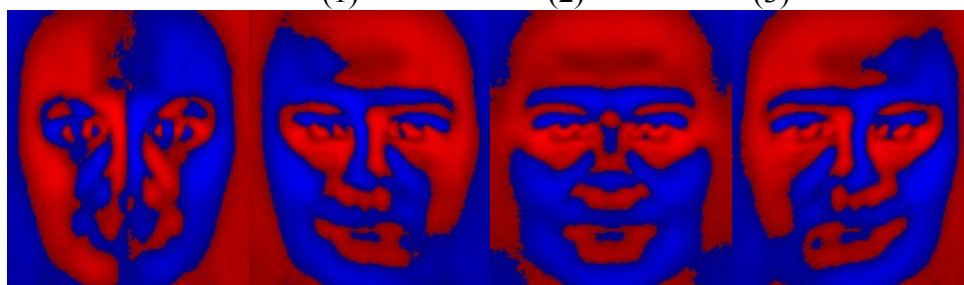
Tyrimų eigoje buvo analizuotas spartaus glodaus eksponentinio filtro taikymo rezultate gautos informacijos pritaikymas veidų suradimui naudojant *NB* klasifikatorius. Dalis pirminių hipotezių variantų yra pateikiama žemiau esančiame pav. 22. Požymių pritaikomumui veidų ir neveidų klasifikavimo uždaviniui spręsti ir vizualizuoti buvo vertinami vidutiniai vaizdai, vėliau – apmokintų *NB* klasifikatorių tikimybinių modelių atvaizdai, kuriuose tam tikroje pozicijoje esanti taškelių spalva nurodo labiausiai veidų klasei būdingos (maksimalios tikimybės) taškelių k reikšmės įvertį, o spalvos intensyvumas – proporcingas minėtos tikimybės $P(x_k|Y_i)$ dydžiui (žr. pav. 23). Praktinio tyrimo eigoje buvo ieškoma veidų požymių, kurių naiviųjų klasifikatorių vaizduose egzistavo sritys su sodriomis spalvomis. Tokie vaizdai charakterizuoja veidų *NB* klasifikatorius, kuriuose taškelių reikšmės gali gana tvirtai „nubalsuoti“ už nagrinėjamų duomenų X priklausomybę vienai iš Y_i klasių. Geriausiai minėtą kriterijų atitinkančių veidų vaizdų požymių *NB* klasifikatorių modelių tikimybiniai atvaizdai pateikti pav. 23.



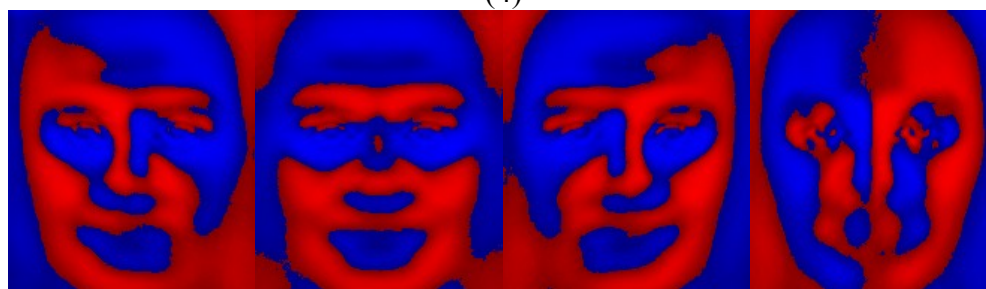
Pav. 22. *NB* klasifikatoriams tyrinėtų veidų vaizdų požymių (gautų taikant spartų glodų eksponentinį filtrą pavyzdžiai ($\sigma = 2.0$)). Iš kairės į dešinę, iš viršaus į apačią: 1) filtruoto vaizdo lokaliųjų ekstremumų taškai; 2) filtruoto vaizdo gradiento lokaliųjų maksimumų taškai; 3) filtruoto vaizdo gradiento skeletonizuotų kontūrų ir gradiento krypčių (kvantuotų į 8 intervalus) kontūruose taškai; 4) filtruoto vaizdo antros eilės dalinė išvestinė x ir y kryptimis (neigiamos reikšmės mėlynos, neneigiamos – raudonos); 5) Laplaso taškai (neigiamos reikšmės mėlynos, neneigiamos – raudonos); 6) Laplaso nulio kirtimų taškai.



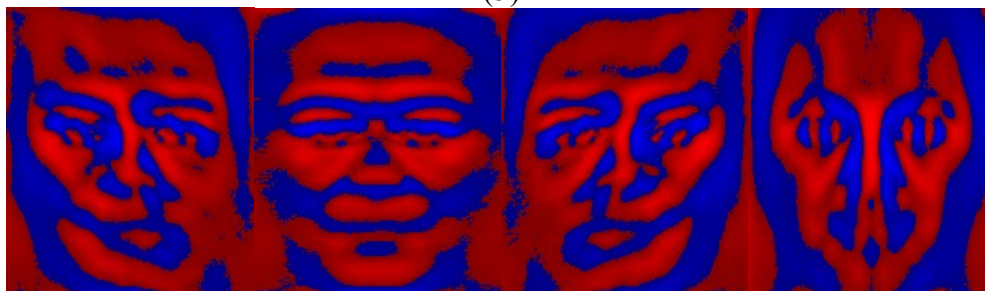
(1) (2) (3)



(4)



(5)

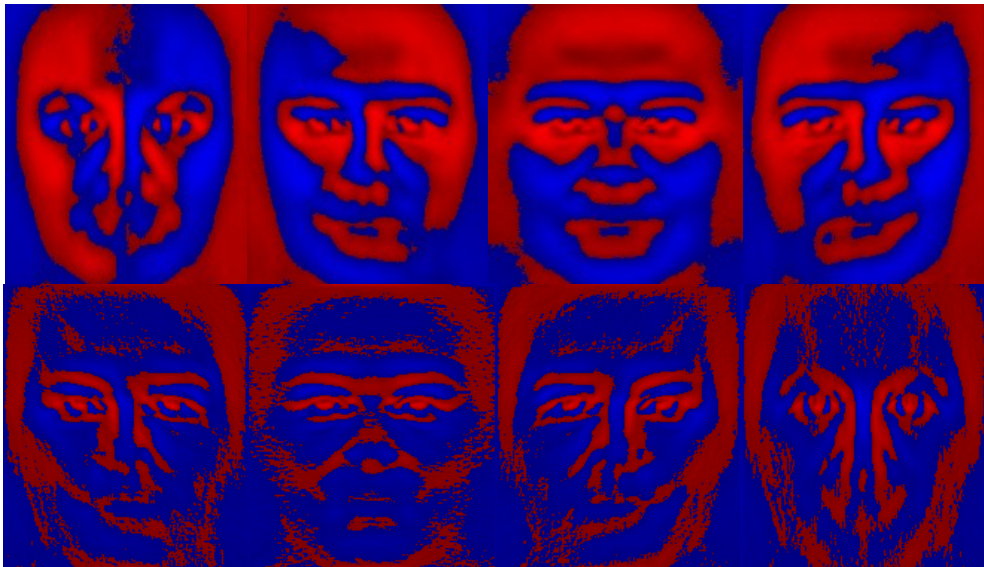


(6)

Pav. 23. Geriausių filtruotų veidų vaizdų požymių (aukštų tikimybių $P(x_k|Y_I)$ sričių egzistavimo kriterijaus atžvilgiu) *NB* klasifikatorių tikimybinių modelių atvaizdai. Sodresnės spalvos reprezentuoja veidų klasės reikšmių didesnes tikimybes. Išskyrus (1), mėlyna spalva nusako neigiamas reikšmes, raudona – neneigiamas. (1) – (3) eksponentinio filtro parametras $\sigma = 2.0$. 1) filtruotų veidų vaizdų gradiento krypčių (kvantuotų į 8 intervalus) reikšmių modelis; 2) filtruotų veidų Laplaso ženklų modelis; 3) filtruotų veidų išvestinės x ir y kryptimis ženklų modelis; 4) filtruotų veidų kryptinių (0, 45, 90, 135 laipsnių) pirmos eilės išvestinių ženklų modelis; 5) filtruotų skirtingais σ veidų skirtumų kryptinių (0, 45, 90, 135 laipsnių) pirmos eilės išvestinių ženklų modelis; 6) filtruotų skirtingais σ veidų skirtumų kryptinių (0, 45, 90, 135 laipsnių) antros eilės išvestinių ženklų modelis.

Praktinės realizacijos ir tyrimų eigoje išaiškėjo, kad reikia nagrinėti požymius platesnėse veido vaizdo srityse, o ne tik jo kontūruose (apie tai taip byloja ir [LH09] darbe aprašyti optimalių požymių parinkimo *NB* eksperimentai). Taip buvo nuspręsta įvertinus tik kontūruose apskaičiuotos informacijos požymių (žr. pav. 22, (1) – (3), (6)) *FERET_FA* mokymo rinkinio vidutinius veidus bei *NB* klasifikatoriaus modelius. Vidutiniai vaizdai buvo neryškūs, o naivuosiuose modeliuose visiškai nebuvo matyti vyraujančių tikimybių (t.y. dauguma tikimybinių modelių pozicijų k turėjo tikimybinis įverčius $P(x_k|Y_I) \approx 0.5$) arba tiesiog neturėjo informacijos (dominavo juoda spalva). Taip yra todėl, kad žmonių veidų formos kontūrai vaizduose labai varijuoja ne tik dėl natūralių žmonių veidų skirtumų, bet dėl tų pačių subjektų veido išraiškos, posūkių bei nuotraukų fiksavimo aplinkos apšvietimo ir kt. sąlygų. Taigi, toliau binariniai ir gradiento kvantuoti krypčių požymiai buvo nagrinėjami visose kandidatinio veido X pozicijose, o ne tik kontūruose.

Siekiant įsitikinti *NB* klasifikatorių veiksmingumu praktikoje, jiems buvo visų pirma iškeltas veidų ir neveidų klasifikavimo uždavinys ant mokymo duomenų. Jeigu klasifikatorius nepasiteisintų mokymo duomenis, galima nutraukti jo tolimesnį tyrimą, kadangi jis juo labiau neveiktų ant nežinomų testavimo duomenų. *FERET_FA* mokymo rinkinio veidai buvo geometriškai normalizuoti į *NB* klasifikatoriams pasirinktą 166 x 192 dydžio veido modelį. Neveidai buvo suformuoti iškerpant 166 x 192 dydžio sritis atsitiktinose x ir y pozicijose iš atsitiktinių (pakankamai dideliu / mažu išvengti veidams) masteliu pakeistu to paties *FERET_FA* rinkinio veidų vaizdų. Tai yra analogas situacijai, kurioje buvo planuojama taikyti naivųjį klasifikatorių: nagrinėjama *DAB* kaskada gražinto kandidatinio veido aplinka X ir, parenkant optimalius mastelio s , posūkio a , x ir y poslinkių t_x ir t_y įverčius, ieškoma didžiausios veido tikimybės $P(Y_{i=1} | X)$ pozicija. Atlikus minėtus klasifikavimo testus ant mokymo duomenų bei analogiškus testus ant testavimo *FERET_DUPI* duomenų, buvo įsitikinta tiesioginiu sąryšiu tarp *NB* klasifikatorių tikimybinių modelių vaizdų (žr. pav. 23) ir klasifikavimo rezultatų. Visi reprezentatyvus pagal tikiybinio modelio vaizdai parodė pakankamai gerus klasifikavimo rezultatus: mažiausiai 89.9% teisinga klasifikuotų veidų vaizdų ir daugiausiai 10% neteisingai klasifikuotų neveidų vaizdų. Tuo tarpu blankesnius naiviojo modelio atvaizdus turinčios filtruoto vaizdo antros eilės kryptinės išvestinių iš tikrųjų nepasiteisino ir atmetos kaip nevertos nagrinėti (žr. pav. 24).



Pav. 24. Testavimo fazėje atmesto filtruotų veidų kryptinių (0, 45, 90, 135 laipsnių) antros eilės išvestinių ženklo *NB* modelis (apačioje). Tikimybinio modelio spalvų sodrumo palyginimui pateikiamas perspektyvus filtruotų veidų kryptinių (0, 45, 90, 135 laipsnių) pirmos eilės išvestinių ženklo modelis (viršuje). ($\sigma = 2.0$).

Toliau, paprastumo dėlei, sistemos realizacijai pasirinktas naudoti turtingiausias informacijos prasme ir perspektyviausiu pasirodęs filtruoto vaizdo gradiento kvantuotų krypčių *NB* klasifikatorius (žr. pav. 25). Kitų atrastų perspektyvių požymių – filtruoto vaizdo Laplaso ženklo ir išvestinės x ir y kryptimis – tyrinėjimas ir galimas junginys su gradientais grindžiama segmentacijos patikslinimo metodika paliktas galimam tolimesniam sistemos algoritmo tobulinimui. Tas pats galioja ir kryptinėms pirmos bei antros eilės išvestinėms, pritaikytoms filtruotiems vaizdams ir filtruotų su skirtingais σ vaizdų skirtumams, kadangi šių požymių taikymas reikalauja papildomų laiko ir resursų segmentacijos metu. Papildomas laikas būtinas tiek galimiems papildomiems filtravimams, tiek papildomiems kryptinių išvestinių skaičiavimams po filtravimo atlikti.



Pav. 25. Veidų segmentacijos sistemos patikslinimo žingsniui pasirinkto kvantuotų gradiento krypčių filtruotame ($\sigma = 2.0$) veido vaizde *NB* klasifikatoriaus modelis ir jo didžiausių tikimybių sritys. Iš kairės į dešinę: 1) tikimybinio modelio vaizdas; 2) modelio tikimybės $P(x_k|Y_I) > 0.5$; 3) modelio tikimybės $P(x_k|Y_I) > 0.6$; 4) modelio tikimybės $P(x_k|Y_I) > 0.7$. Modelis apmokintas naudojant *FERET_FA* duomenų rinkinį.

Gradiento krypčių *NB* klasifikatorius buvo apmokintas ant 166 x 192 dydžio normalizuotų veidų modelių iš *FERET_FA*, o vėliau ir *FERET_FA* kartu su *FRGC_TRAINING* rinkinių. Apmokymas atitinkamai truko kelias minutes ir apie dešimtį minučių. Toks geras mokymo laikas ir sąlyginai nedidelis mokymo duomenų kiekis suteikia kur kas platesnes eksperimentavimo ir tobulinimo galimybes per trumpą laikotarpį nei *AdaBoost* godžioji metodika, Apmokintų naiviųjų Bajeso klasifikatorių pritaikymas *DAB* segmentacijos korekcijai rezultatai pateikti toliau išdėstyame skyriuje 3.3.

3.3. Realizacijos segmentacijos rezultatai

Realizuotojoje veidų segmentacijoje iš pradžių visam vaizdui yra pritaikomas *DAB* metodika apmokintas kaskadinis klasifikatorius (žr. sk. 3.2.1). Vėliau kaskados gražintos kandidatinių veidų vaizdo sritys (kartu su kaimynine aplinka) yra geometriškai normalizuojamos į *NB* modelio dydžio regionus ir perleidžiamos naiviajam Bajeso (*NB*) klasifikatoriui. Šis yra apmokintas klasifikuoti veidų ir neveidų vaizdų kvantuotas gradiento kryptis, kurios išgaunamos panaudojant vertikaliosios ir horizontaliosios krypties pirmos eilės išvestines, apskaičiuotas taikant anksčiau aprašytą spartųjų glodų eksponentinį filtrą (žr. sk. 3.2.2). (Šiame darbe filtro parametras σ buvo parinktas ant mokymo duomenų ir fiksuotas ($\sigma \approx 2$) visoms testavimo duomenų bazėms). *NB* klasifikatorius buvo iteratyviai kviečiamas iki trijų kartų ($1 \leq j \leq 3$) su skirtingais vis mažėjančiais mastelio $s^j \in [s^j_1, s^j_2]$, horizontalaus $t_x \in [t_{x,1}, t_{x,2}]$ ir vertikalaus $t_y \in [t_{y,1}, t_{y,2}]$ poslinkių, kampo $a^j \in [a^j_1, a^j_2]$ kitimo intervalų ir šių parametrų keitimo žingsniais $\{\Delta s^j, \Delta t_x, \Delta t_y, \Delta a^j\}$, ieškant *DAB* kandidato aplinkoje maksimalios tikimybės (tiksliau, tikimybės logaritmo $\ln(P(Y_{i=1}|X))$) veido. Jeigu po *NB* klasifikatoriaus kvietimo j patikslinimas neįvyko (t.y. gražinta priklausomybės veido klasei tikimybė nepasikeitė, $\ln^{j+1}(P(Y_i|X^j)) = \ln^j(Y_i|P(X^{j-1})^j)$), patikslinimo metodas daugiau nėra kviečiamas.

Idealiu atveju norėtuši perrinkti visas galimas keturių laisvų parametrų (mastelio s , horizontalaus poslinkio t_x , vertikalaus poslinkio t_y , posūkio a) kombinacijas, tačiau tai yra labai lėtas ir praktiškai nepritaikomas algoritmas, Todėl buvo pasirinkta minėta kelių ($1 \leq j \leq 3$) kvietimų su skirtingais intervalais ir žingsniai strategija. Kadangi segmentacijos sistema skirta priešakiniam veidams su posūkiu plokštumoje 10 laipsnių amplitudėje, daroma prielaida, kad, visų pirma, galima pakankamai efektyviai patikslinti mastelį s bei t_x ir t_y reikšmes, neatsižvelgiant į posūkį a . Filtravimą galima atlikti dviem būdais: kiekvienam tikrinamam masteliui s arba atlikti tik vieną kartą, prieš mažinant mastelį. Pastaruoju atveju 18% nukentčia naiviojo klasifikatoriaus korekcijos rezultatų tikslumas, o sparta pagerėja 16%. Todėl dabartinėje sistemoje tikslumui išlaikyti filtravimas atliekamas kiekvienam masteliui $s^j \in [s^j_1, s^j_2]$, o vėliau surasto mastelio sukamiems kandidatams ($a^j \in [a^j_1, a^j_2]$ prie fiksuoto s^j), Atliekamų filtravimų skaičius tiesiogiai įtakoja Java kalba realizuoto *NB* klasifikatoriaus spartą. Dabartinis greitis yra vidutiniškai 1.5 – 1.8 sekundės FERET duomenų bazei ir apie 2.1 – 2.3 sekundės *FRGC* duomenų bazei. C kalba realizuotas pirmoji sistemos dalis – *DAB* klasifikatorius – veikia itin sparčiai: mažiau nei 40 milisekundžių FERET duomenų bazei ir mažiau nei 550 milisekundžių arba 250 milisekundžių (parinkus skirtingus kaskados „žingsniavimo“ vaizdu gardedės parametrus) *FRGC* duomenų bazei. Šie greičio įverčiai parodo, kad *NB* klasifikatorių būtina optimizuoti ir galimai taip pat perrašyti C kalba, siekiant taikyti jį realaus laiko sistemose.

Veidų segmentacijos sistemos tikslumo ir jos pirmojo bei abiejų žingsnių taikymo rezultatai yra pateikiami žemiau esančiuose histogramose ir taškiniuose diagramose. Visų pirma yra pateikiami P. Viola ir M. Jones [VJ01b] surastų veidų tikslesnės lokalizacijos bandymų rezultatai pritaikant glodinto vaizdo gradiento kryptčių *NB* klasifikatorių. Veidų atstumai buvo vertinami pagal surastų veidų hipotetinių akių (numanomų remiantis *DAB* mokymo veidų modelio proporcijomis) atstumą iki tikrųjų akių. Šis atstumas yra vertinamas kaip maksimalaus iš hipotetinių ir tikrųjų dešinės arba kairės akių atstumų santykis su tikrųjų atstumų tarp tikrųjų akių (žr. formulę 12).

$$dist = \frac{\max \left(\text{dist} \left(\text{Eye}, l\text{FoundEye} \right), \text{dist} \left(\text{Eye}, r\text{FoundEye} \right) \right)}{\text{dist} \left(\text{Eye}, r\text{Eye} \right)} * 100\% \quad (13)$$

Šiame skyriuje įvertinant veidų lokalizacijos tikslumą daroma prielaida, kad veidas yra surastas (t.y. tinkamas tolimesniam požymių išskyrimui (angl. *extraction*) biometriniu sistema), jeigu jo akys yra nutolusios ne daugiau kaip 25 % nuo tikrųjų akių remiantis (13) formulėje apibrėžtu atstumo įverčiu (analogiškai BioID duomenų bazes apraše deklaruojamu protokolu ir [JKF01]).

Žemiau esančioje lentelėje 2 pateikiami duomenys skyriuje 3.1. aprašytiems visiems duomenų rinkiniams, naudojant realizuotos sistemos: a) *DAB* kaskadą; b) *DAB* ir *NB* klasifikatorių, apmokintą tik *FERET_FA* duomenų rinkiniu. Lentelėje pateikti veidų suradimo statistiniai įverčiai parodo, kad realizuotos veidų segmentavimo sistemos pirmasis *DAB* žingsnis leido surasti vidutiniškai 64.44% iš visų 511 sužymėtų veidų itin prastos kokybės mažiems MIT/CMU duomenų rinkiniams, dalis kurių įtraukia nemažai veidų, kurie netenkina pirminių sistemos minimalaus veido dydžio ir maksimalių posūkių plokštumoje ir erdvėje hipotezių. Ant visų kitų šias hipotezes apytiksliai (daugumoje atveju) tenkinančių ir normalios raiškos vaizdų duomenų bazių vidutinis *DAB* veidų sėkmingos segmentacijos dažnis 98.03% 41820 sužymėtų veidų. Vidutinis teisingai segmentuotų veidų dažnis *DR* (angl dar vadinamas *Recall*) visiems rinkiniams yra 87.95% (iš viso 42331 veidų). *NB* klasifikatoriaus pritaikymas pagerina minėtus tris *DR* įverčius iki 67.94%, 98.26% ir 89.16% atitinkamai. Nors *NB* pritaikymas pagerina minėtus vidutinius *DAB* segmentacijos rezultatus prie $dist \leq 25\%$ procentaliai 1.3% – 5.4%, remiantis lentelėje 3 pateiktas rezultatai galima matyti, kad *NB* klasifikatorius ženkliai pagerina *DAB* gražintų kandidatų lokalizacijos rezultatus nuotolio nuo tikrų akių 5%, 10% taškuose. Antrasis žingsnis vidutiniškai net 20 kartų padidina mažiau nei 5% nutolusių nuo tikrųjų veidų skaičių ir daugiau nei trigubai padidina mažiau nei 10% nutolusių surastų veidų regionų veidų skaičių. Svarbus faktas, kad *NB* klasifikatoriaus pajungimas antrajame etape duoda tendencinę pagerėjimą tiek mokymo, tiek testavimo duomenims, t.y. nepasižymi persimokymu. Remiantis toliau pateiktomis verifikacijos *DET* kreivėmis, natūralu tikėtis, kad tai turės ženklios įtakos veidų palyginimo tikslumo pagerėjimui. Lokalizacijos tikslumui prieš *NB* patikslinimą ir po jo pademonstruoti pav.26 pateikiamos surastų veidų regionų atstumų iki tikrų akių *dist* pasiskirstymo histogramos ir kaupiamosios histogramos (simbolizuojančios teisingai klasifikuotų sričių iš visų gražintų kandidatų sričių įvertį *TPR* (angl. *precision*)).

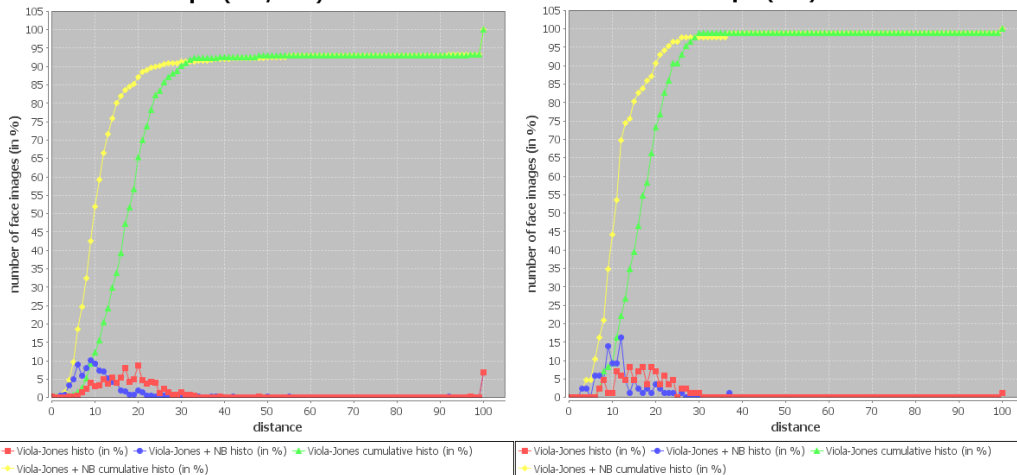
Vaizdų rinkinys	Vaizdų skaičius	Veidų skaičius	Teisingai surasta veidų (<i>DR</i> , <i>Recall</i>), %		Nesurasta veidų (<i>FRR</i>) %	
			<i>DAB</i>	<i>DAB</i> + <i>NB</i>	<i>DAB</i>	<i>DAB</i> + <i>NB</i>
MIT/CMU	130	511	60.67	64.97	39.33	35.03
MIT	23	157	50.32	53.50	49.68	46.50
MIT80	80	232	82.33	85.34	17.67	14.66
BioID	1521	1521	96.38	95.66	3.62	4.34
FERET_FA	1196	1196	98.75	99.16	1.25	0.84
FERET_FB	1195	1195	98.16	99.50	1.84	0.50
FERET_DUP1	722	722	99.03	99.72	0.97	0.28
FRGC_TRAIN	12776	12881	98.87	99.04	1.13	0.96
FRGC_EXP1	16028	16052	99.73	99.83	0.27	0.17
FRGC_EXP4	8037	8253	95.30	94.91	4.70	5.09

Lentelė 2. Realizuotos veidų segmentacijos sistemos veidų teisingos segmentacijos *DR* ($dist \leq 25\%$ režiuose) rezultatai, kai taikoma tik pirmoji (*DAB* kaskados) ir abi komponentės (*DAB* + glodintų gradiento krypčių *NB* klasifikatorius).

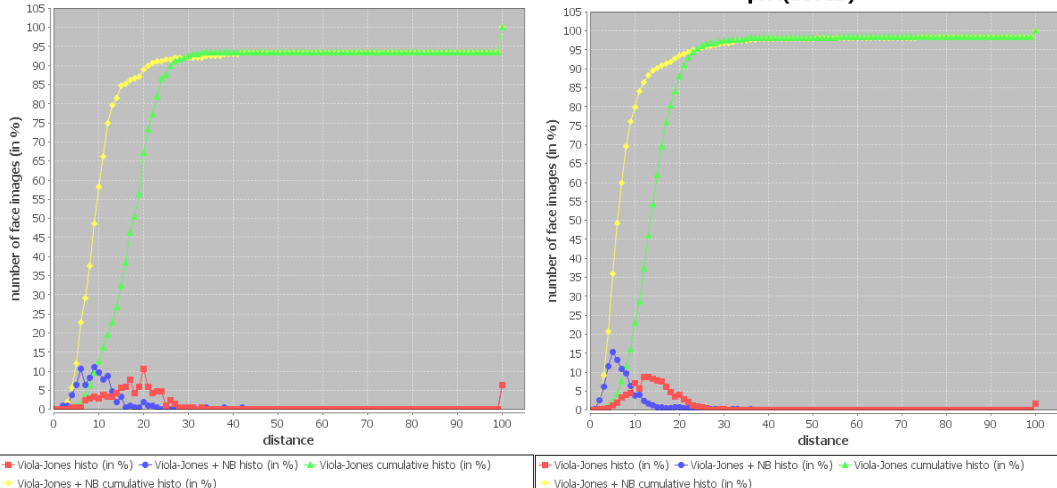
Vaizdų rinkinys	Teisingai surastų veidų regionų skaičius su $dist \leq 40\%$ (TPR, Precision), %		Teisingai surastų veidų regionų skaičius su $dist \leq 25\%$ (TPR, Precision), %		Teisingai surastų veidų regionų skaičius su $dist \leq 10\%$ (TPR, Precision), %		Teisingai surastų veidų regionų skaičius su $dist \leq 5\%$ (TPR, Precision), %	
	DAB	DAB + NB	DAB	DAB + NB	DAB	DAB + NB	DAB	DAB + NB
MIT/CMU	92.62	92.08	84.70	90.71	12.30	51.91	0.82	9.56
MIT	98.84	98.84	91.86	97.67	9.30	44.19	0.00	4.65
MIT80	93.52	93.06	88.43	91.67	12.50	58.33	0.93	12.04
BioID	98.03	97.83	96.38	95.66	22.95	79.88	2.24	35.96
FERET_FA	99.16	99.16	98.75	99.16	16.30	94.73	0.42	57.11
FERET_FB	99.41	99.50	98.16	99.50	14.73	95.31	0.67	57.41
FERET_DUP1	99.72	99.72	99.03	99.72	21.61	95.43	0.55	51.66
FRGC_TRAIN	99.43	99.38	98.87	99.04	36.41	86.75	2.66	36.22
FRGC_EXP1	99.82	99.83	99.73	99.83	49.41	92.09	4.01	43.43
FRGC_EXP4	98.36	98.15	97.86	97.46	40.49	73.40	3.53	21.77

Lentelė 3. Realizuotos veidų segmentacijos sistemos teisingai surastų veidų iš visų rastų regionų dažnis TPR įvairiuose atstumo nuo tikrojo veido slenksčio $dist$ režiuose, kai taikoma tik pirmoji (DAB kaskados) ir abi komponentės ($DAB +$ glodintų gradiento krypčių NB klasifikatorius).

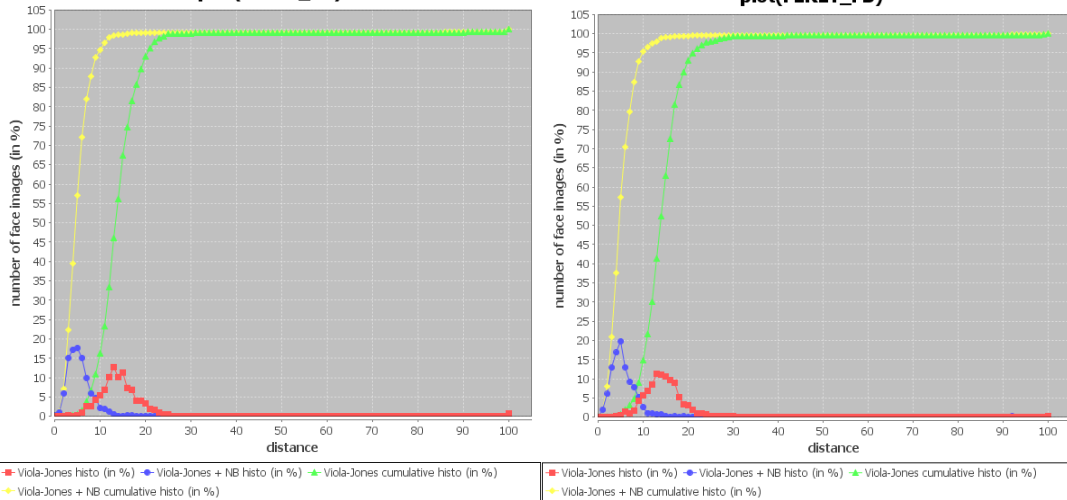
Distance(found_eyes, true_eyes) (cumulative) histograms plot(MIT/CMU) Distance(found_eyes, true_eyes) (cumulative) histograms plot(MIT)



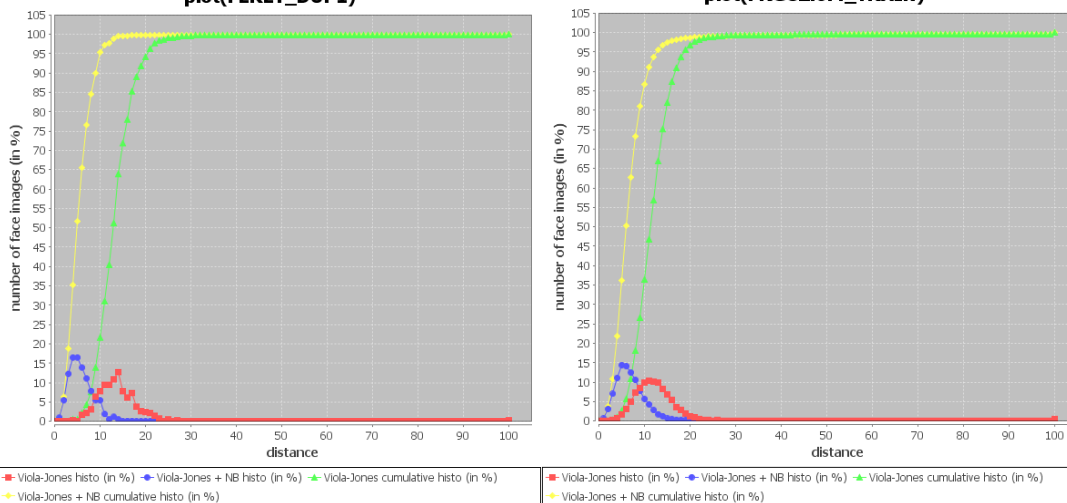
Distance(found_eyes, true_eyes) (cumulative) histograms plot(MIT/CMU 80) Distance(found_eyes, true_eyes) (cumulative) histograms plot(BIOID)



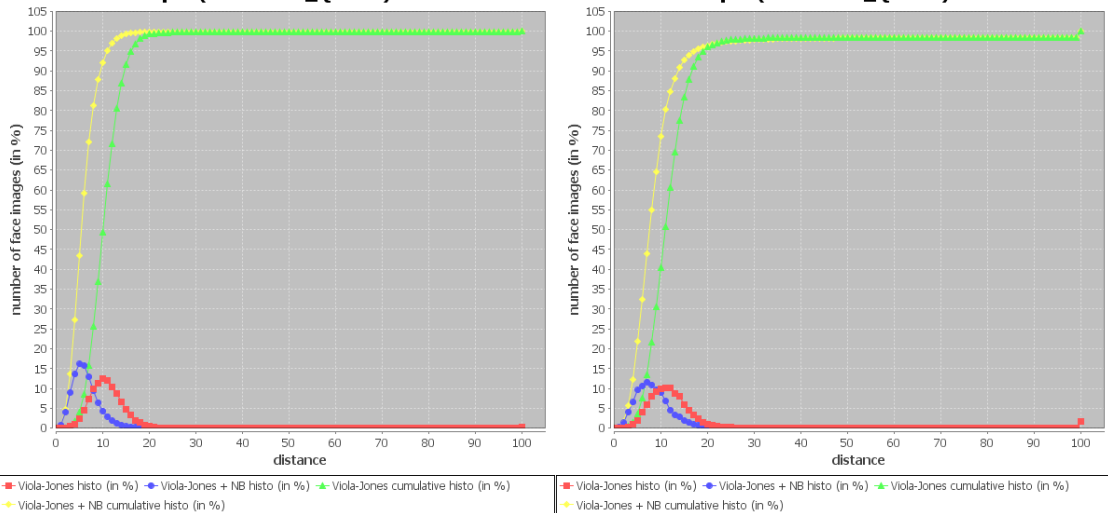
Distance(found_eyes, true_eyes) (cumulative) histograms plot(FERET_FA) **Distance(found_eyes, true_eyes) (cumulative) histograms plot(FERET_FB)**



Distance(found_eyes, true_eyes) (cumulative) histograms plot(FERET_DUP1) **Distance(found_eyes, true_eyes) (cumulative) histograms plot(FRGC2.0.4_TRAIN)**

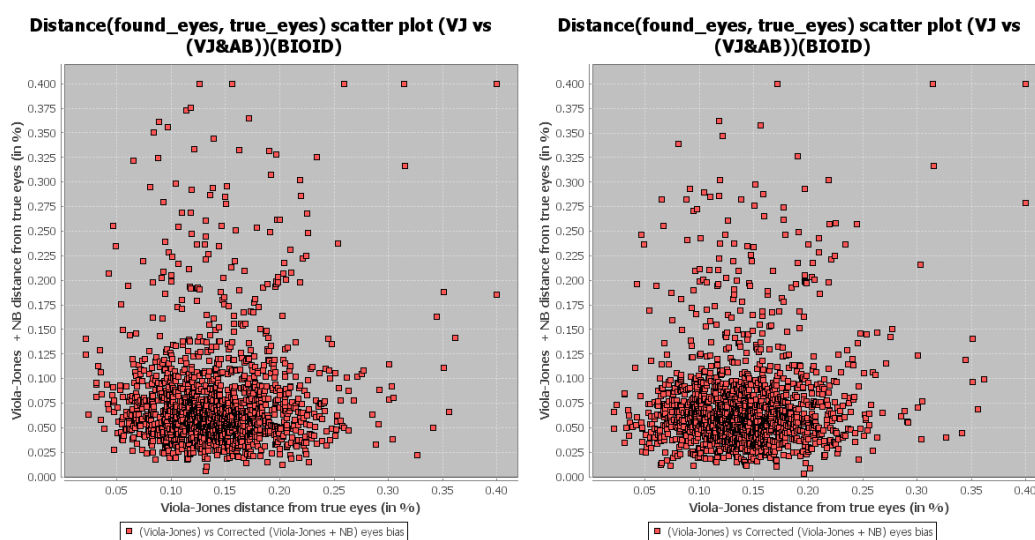


Distance(found_eyes, true_eyes) (cumulative) histograms plot(FRGC2.0.1_QUERY) **Distance(found_eyes, true_eyes) (cumulative) histograms plot(FRGC2.0.4_QUERY)**



Pav 26. Realizuotos veidų segmentacijos sistemos surastų veidų atstumo iki numanomo tikrojo veido procentinės histogramos ir kaupiamosios (angl. *cumulative*) procentinės histogramos, kai taikoma tik pirmoji (*DAB* kaskados) ir abi komponentės (*DAB* + glodintų gradiento krypčių *NB* klasifikatorius). Horizontaliojoje ašyje žymimas atstumas iki tikrųjų akių *dist* (13), vertikaliajoje – surastų veidų regionų skaičius procentais nuo visų grąžintų regionų *TPR* (angl. *Precision*).

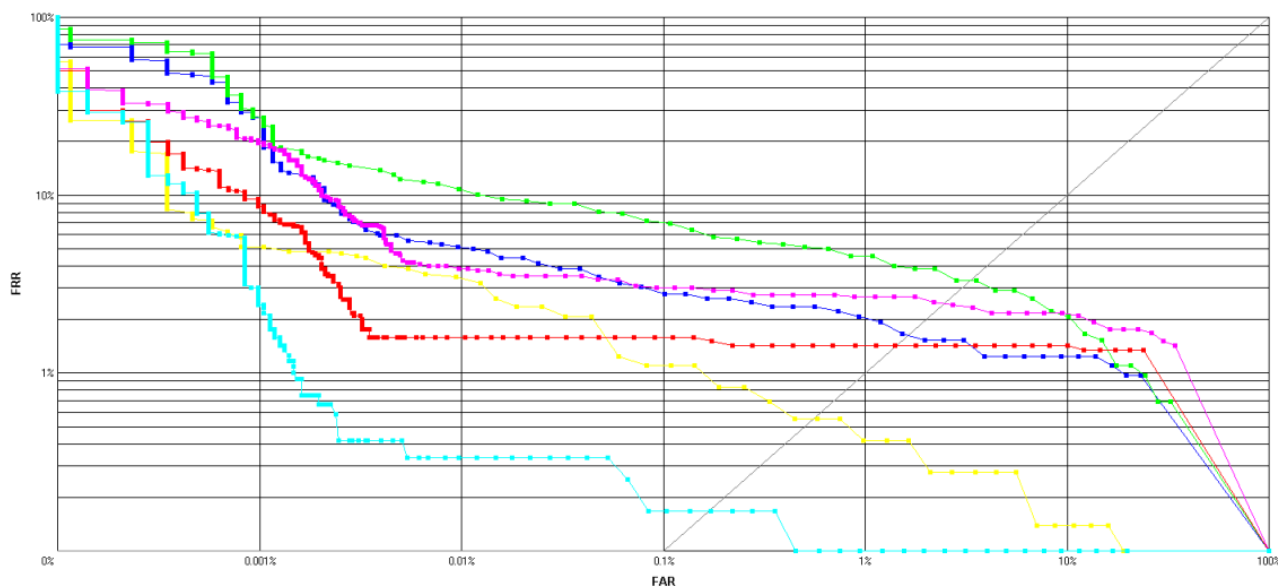
Galima pastebėti, kad šiame darbe naudojamo veidų teisingo klasifikavimo slenksčio taške $dist \leq 25\%$ DR (*Recall*) ir TPR (*Precision*) pagerėja po NB $FERET_FA$ mokinto pritaikymo ne visoms duomenų bazėms (pvz., BioID). Iš praktinės patirties ir mokslinės literatūros šaltinių analizės gerai žinoma, kad DAB rezultatai stipriai priklauso nuo mokymo imties ir jos dydžio. Šio darbo eigoje buvo nuspręsta ištirti, kaip pasikeičia rezultatai, apmokinus NB klasifikatorių ant visų mokymo duomenų, t.y. $FERET_FA$ ir $FRGC_TRAIN$ rinkinių. Žemiau yra pateikiami DAB ir NB $FERET_FA$ atstumų bei DAB ir NB apmokinto ant $FERET_FA$ bei $FRGC_TRAIN$ kartu pasiskirstymo grafikai BioID testavimo rinkiniui. BioID įgavo kiek geresnį nei aukščiau patektose lentelėse $TPR = 96.15\%$ ir $DR = 96.25\%$. Panašios tendencijos buvo pastebimos ir ant kitų duomenų bazių, kai šioms antrajame etape taikomas ant dviejų rinkinių apmokintas NB klasifikatorius. Nors (kaip matoma iš pav. 27) įverčių skirtumas yra nedidelis, tačiau verta bandyti NB pateikti dar daugiau mokymo duomenų, kadangi NB mokymas yra greitas procesas, ir segmentacijos fazės našumas yra faktiškai konstantinis. Iš kitos pusės, parktiniai eksperimentai netgi vos su ant kelių dešimčių veidų apmokintais NB klasifikatoriais, veikė stebėtinai gerai ant kokybiškos FERET veidų duomenų bazių. Todėl gal labiau prasminga kurti klasių poklases ir kombinuoti jų NB tiesiškai taikant svorinius koeficientus.



Pav. 27. Surastų veidų regionų atstumų pirmajame etape ir antrajame etape NB pakoreguotų atstumų taškinis pasiskirstymo grafikas BioID duomenų rinkiniui, X koordinatė žymi pirmajame (DAB kaskados) etape gautus atstumo įverčius. Y koordinatė žymi koreguotus atstumo įverčius gautus papildomai pritaikius antrąjį žingsnį (NB $FERET_FA$ klasifikatorių kairėje, NB $FERET_FA$ + $FRGC_TRAIN$ dešinėje).

Visi tolimesni testavimo rezultatai pateikiami vėl gi naudojant NB $FERET_FA$ klasifikatorių. Siekiant įvertinti sistemos patikimumą bei įsitikinti antrojo žingsnio (filtruoto vaizdo gradiento krypčių naiviojo Bajeso klasifikatoriaus) daroma teigiama įtaka DAB rezultatams, buvo taip pat įvertintas veidų verifikacijos patikimumas naudojant naudojamų segmentacijos algoritmų gautas akis. Žemiau pav. 28 yra pateiktos atpažinimo tikslumą vizualizuojančios DET kreivės (angl. *Detection Error Tradeoff Curves*). Veidų palyginimui buvo naudojama Neurotechnology veidų atpažinimo programinė įranga „VeriLook” [Neu11]. Nors NB patikslinimas pagerina DET kreivės lygių klaidų $FRR = FAR = EER$ (angl. *Equal Error Rate*) taške nuo apytiksliai 1.7 iki 2 kartų, bendri sistemos rezultatai yra prastesni už minėtos komercinės programinės įrangos rezultatus.

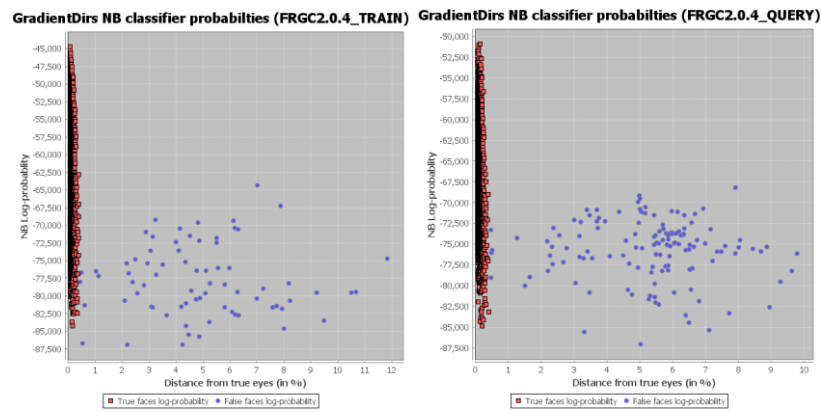
Title	Pairs	Impostors	Genuines	EER	Zero FAR	FAR 0.001%	FAR 0.01%	FAR 0.1%	FAR 1%	FAR 10%	FAR 100%	Zero FRR	FRR 1%	FRR 10%	FRR 100%	Rank 1	Rank 10	Rank 100
feret_fa_vs_dupl_Eeyes	863512	862790	722	1.63%	70.2%	27.3%	5.1%	2.76%	2.04%	1.9%	0%	100%	1%	0.000348%	0%	5%	97.5%	99.8%
feret_fa_vs_dupl_VIeyes	863512	862790	722	3.32%	88.3%	27.3%	10.7%	7.01%	4.57%	2.09%	0%	100%	23.6%	0.0128%	0%	32.2%	96%	97.9%
feret_fa_vs_dupl_LIeyes	863512	862790	722	0.554%	58.5%	5.12%	3.41%	1.11%	0.416%	0.139%	0%	59.6%	0.16%	0.000348%	0%	98.1%	99.4%	99.9%
feret_fa_vs_fb_Eeyes	1429220	1428025	1195	1.42%	66.9%	8.83%	1.93%	1.93%	1.42%	1.42%	0%	100%	43.1%	0.00384%	0%	98.4%	99.6%	99.7%
feret_fa_vs_fb_VIeyes	1429220	1428025	1195	2.47%	63.2%	19.7%	3.85%	3.01%	2.68%	2.15%	0%	100%	53.8%	0.00203%	0%	96.9%	97.3%	98%
feret_fa_vs_fb_LIeyes	1429220	1428025	1195	0.167%	73.3%	2.43%	0.395%	0.167%	0.0837%	0%	0%	59.9%	0.00147%	0.00049%	0%	99.7%	99.9%	100%



Select a point on a curve

Pav. 28. Veidų segmentacijos sistemos pirmajame ir abiejų etapų rezultate gautų akių pritaikymo įtaka veidų verifikacijai *FERET_FB* ir *FERET_DUPI* rinkinių su *FERET_FA* galerija. Raudona ir mėlyna kreivės žymi atitinkamus minėtų kandidatinių rinkinių palyginimo su galerija rezultatus, kai naudojamos akys, gautos po abiejų realizuotų segmentacijos sistemos žingsnių (*DAB + NB*); žalia ir rausva – palyginimo rezultatus, kai naudojamos pirmojo žingsnio rezultate (*DAB*) gautos akys; žydra ir geltona – palyginimo rezultatus, kai naudojamos „VerLook“ SDK segmentuotos akys.

Taip pat buvo išanalizuota anksčiau minėta antroji *DAB* kaskada grąžinamų rezultatų problematika – neteisingai surastų veidų eliminavimas tolimesniame segmentacijos etape. Tuo tikslu buvo apskaičiuotos *NB* klasifikatoriaus gaunamų log-tikimybių atsakų $\ln(P(Y_i|X))$ pasiskirstymas teisingiems ir tikrai neteisingiems veidams su $dist \geq 40\%$ (neveidams). Toks pasiskirstymas (žr. pav. 29) leidžia įvertinti tikrų ir netikrų *DAB* grąžinamų kandidatų atskyrimo galimybes, t.y. klaidingus veidus atmesti leidžiančio, bet tuo pačiu teisingus veidus išsaugančio tam tikro slenksčio Bajeso atsakams parinkimo galimybes. Žemiau esantis pasiskirstymo grafikas didesniu neteisingų veidų skaičiumi pasižymintiems *FRGC* vaizdams parodo, kad nepavyks parinkti slenksčio, kuris vieningai leistų atmesti visus *DAB* klasifikatoriumi grąžinamus klaidingus veidus neatmetant ne vieno teisingo veido, patingai testavimo (ne mokymo) rinkiniams. Galima parinkti tokį slenksčių, kuris leistų atmesti „užtikrintų neveidų“ dalį, šiems įvertinant naiviojo Bajeso klasifikatoriaus atsaką vienintelį kartą (tik esamojoje pozicijoje) ir nutraukiant jų tolimesnį apdorojimą. Taip pat galima išskirti „užtikrintų veidų“ bei „abejotinų veidų“ aibes, kiekvienai iš jų atitinkamai parenkant skirtingus siauresnius bei platesnius mastelių, poslinkių ir posūkio parametrus lokalizacijos korekcijos etape.



Pav. 29. Veidų segmentacijos sistemos *NB* klasifikatoriaus tikimybių logaritmo $\ln(P(Y_i|X))$ atsakų pasiskirstymas *FRGC_TRAIN* mokymo ir *FRGC_EXP4* testavimo veidų rinkiniams vaizduose *DAB* surastiems teisingiems (raudona) ir klaidingiems veidų (neveidų) regionams (mėlyna).

Segmentacijos sistemos rezultatai taip pat buvo palyginti su kitų mokslinėje literatūroje aprašytų metodikų rezultatais. Reikia pripažinti, kad realizuotos sistemos tikslumas ant MIT/CMU duomenų bazės rinkinių yra ženkliai prastesnis nei ant visų kitų. Praktinių eksperimentų eigoje išgautas geriausias MIT/CMU rezultatas buvo $DR = 71.43\%$ ir $FP = 43$ ($FPR = 10.54\%$). Tuo tarpu P. Viola ir M. Jones [VJ01a, VJ01b] deklaruoja $DR = 88.4\%$ prie $FP = 31$, o H. A. Rowley, ir kt. [RBK98] -- $DR = 83.2\%$ prie to paties FP . Būtina pabrėžti, kad skirtumas gali būti įtakojamas ne tik prastesniu tikslumu, bet ir skirtingu teisingo / neteisingo veido traktavimu (šio darbo atveju žr. formulę (13)). Lyginant su P. Viola ir M. Jones, verta paminėti, kad šiame darbe naudojama kaskada sudaryta iš dvigubai mažiau pakopų (stirpųjų klasifikatorių H) ir daugiau kaip 10 kartų mažiau silpnųjų klasifikatorių h_i . Tiesinio atotrūkio tarp naudojamų požymių skaičiaus ir Viola-Jones *DAB* bei šiame darbe realizuoto klasifikatoriaus nebuvimas įrodo paskirtų stačiakampių požymių efektyvumą. Nors rezultatai ant itin prastos kokybės nevienareikšmiškos apibrėžtų teisingų veidų prasme MIT/CMU duomenų bazės nėra itin geri, aukščiau pateiktuose lentelėse matomas *NB* pritaikymo efektyvumas kombinacijoje su *DAB* tikslesnei lokalizacijai iš gauti.

Juo labiau įgyvendinta sistema BioID duomenų bazės segmentacijos kokybės prasme yra konkurencinga ir netgi geresnė nei daugelis moksliniuose straipsniuose aprašytų metodikų (žr. [RF05] ir lentelę 4 žemiau).

Autoriai algoritmo, įvertinto ant BioID duomenų bazės	Teisingai surasta veidų (DR, Recall), %	Surasta netikrų veidų (FP), vnt.
Jesorsky et al. [JKF01]	91.80	Nepateikta
Kirchberg et al. [RF05]	92.80	Nepateikta
Hamouz et al. [RF05]	91.30	Nepateikta
Fr̄oba and Ernst [RF05]	97.75	25
G. A. Ram´irez et. al [RF05] test 3	95.00	5240
G. A. Ram´irez et. al [RF05] test 5	93.23	2236
<i>DAB</i>	96.38	55
<i>DAB + NB</i>	95.66	66

Lentelė 4. Realizuotos sistemos ir kitų mokslinėje literatūroje aprašytų veidų segmentacijos metodikų teisingos segmentacijos *DR* ir neteisingai rastų veidų regionų *FPR* įverčiai BioID duomenų bazei.

3.4. Realizacijos tobulinimo gairės

Esminis dabartinės realizacijos trūkumas – antrojo sistemos žingsnio, t.y. glodintų kvantuotų gradiento krypčių naiviojo Bajeso klasifikatoriaus lėtas veikimas. Dideli algoritmo veikimo laiko įverčiai (žr. skyrių 3.3) ne tik trikdo realizuotos segmentacijos taikymą realaus laiko veikimo biometrinėse veidų verifikavimo identifikavimo sistemose, tačiau ir ženkliai lėtina praktinių tyrimų darbą. Todėl žemiau siūlomos antrojo žingsnio optimizavimo greičio prasme gairės:

1. Kiek įmanoma labiau sumažinti veido ir jo aplinkos nagrinėjamų sričių pakartotinių filtravimų skaičių,
2. Bandyti mastelio keitimą taikyti pačiam modelio klasifikatoriui. Tai leistų iš anksto apskaičiuojant N nagrinėjamų mastelių tikimybinį modelius ir netaikyti mastelio keitimo transformacijos nagrinėjamam vaizdui.
3. Realizavus punktą 2, reikėtų filtruoti vaizdą tik vieną kartą (darant prielaidą, kad DAB gražinamų veidų mastelis varijuoja nedideliuose režiuose). Remiantis 3.2, ši optimizacija įgalintų apie 16% pagreitėjimą.
4. Išlygiagretinti patikslinimo užduotis, panaudojant gijas.
5. Eiti tik didėjančios tikimybės x ir y poslinkių kryptimis.
6. Dar labiau sumažinti naudojamo tikimybinio Bajeso modelio dydį.
7. Kiek įmanoma daugiau skaičiavimų transformuoti iš slankaus kablelio (angl. *floating point*) į fiksuoto kablelio (angl. *fixed point*) aritmetiką.
8. Perrašyti Java realizuotus esminius algoritmo metodus į C kalbos funkcijas.

Rezultatų skyriaus 3.3 pav. 28 pavaizduotos DET kreivės nusako, kad norint konkuruoti su komercinėmis sistemos, realizuotą priešakinių veidų segmentacijos vaizduose sistemą būtina tobulinti ir algoritmo tikslumo prasme. Šiam tikslui įgyvendinti gali būti pasitelktos žemiau išdėstytos gairės realizacijos algoritminiam tobulinimui:

1. Naudojamą gradiento kvantuotų reikšmių Bajeso klasifikatorių sujungti su kitų perspektyvių (žr. sk. 3.2.2. ir pav. 23, (2) – (3)) binarinių požymių, kurie nereikalauja reikšmingų papildomų skaičiavimo resursų, NB klasifikatoriais.
2. Naudojamą gradiento kvantuotų reikšmių Bajeso klasifikatorių sujungti su kitų perspektyvių (žr. sk. 3.2.2. ir pav. 23, (4) – (6)) binarinių požymių, kurie galimai reikalauja reikšmingų papildomų skaičiavimo resursų, NB klasifikatoriais. (Ypatingai aktualu, jei pavyks paspartinti esamą realizaciją).
3. Ieškoti universalesnių ir patikimesnių mastelio $s^j \in [s^j_1, s^j_2]$, horizontalaus $t^j_x \in [t^j_{x,1}, t^j_{x,2}]$ ir vertikalų $t^j_y \in [t^j_{y,1}, t^j_{y,2}]$ poslinkių, kampo $a^j \in [a^j_1, a^j_2]$ kitimo intervalų ir jų kitimo žingsnių $\{\Delta s^j, \Delta t^j_x, \Delta t^j_y, \Delta a^j\}$ reikšmių patikslinimui NB klasifikatoriumi. Šiam darbui jie buvo intuityviai atiderinti and $FERET_FA$ duomenų bazės.
4. Paeksperimentuoti su glodaus filtro σ parametrais bei NB klasifikatoriaus tikimybinio modelio dydžio parametrais. Mažesnis modelis teigiamai įtakotų našumą.
5. Paeksperimentuoti su NB klasifikatoriui naudojamo veido modelio turiniu. T. y. potencialiai naudoti mažesnę veidui priklausančią sritį (pvz., sritį aplink akis ir nosį), kur dažniausiai sutinkamos didžiausios priklausomybės veidų klasei tikimybės (žr. sk. 3.2.2. ir pav. 25). Tokiu būdu klasifikavimui bus naudojama tik didžiausiu „balsus“ turinčių taškelių pozicijos.
6. Taikyti NB klasifikatorių DAB algoritmo viduje, pakeičiant daugelio rastų vieno veido kandidatų sąląją maksimalios Bajeso tikimybės stačiakampio išrinkimu. Sėkmės atveju, tai galėtų teigiamas paveikti ir našumą (pvz., sumažėtų reikalingų patikslinimo metodo kvietimų skaičius).
7. Ieškoti kitų, potencialiai naudingų požymių papildomiems informacijos bitams išgauti.

Reikėtų paminėti, kad greičio ir tikslumo optimizacijos dažnai neigiamai įtakoja vieną kitą. Todėl dažniausiai biometrinėse sistemose pasirenkamas priimtinausias šių dviejų sistemos parametų variantas(-ai), atsižvelgiant į potencialaus pritaikymo galutinio vartojimo produktuose (angl. *end-user product*) sritis. Pakankamiems tikslumo ir / arba našumo režiams pasirinkti

galima atlikti išsamesne palyginimo analize su kitomis viešais prieinamomis algoritmų realizacijomis ar komerciniais produktais.

Išvados ir pasiūlymai

Šio darbo eigoje buvo susipažinta su veidų biometrijos pagrindais, atlikta pagrindinės su veidų segmentacija skaitmeniniuose vaizduose susijusios literatūros apžvalga. Remiantis minėtos teorinės apžvalgos rezultatais, buvo išrinkta optimaliausia algoritmo veidų suradimo tikslumo ir našumo santykio prasme metodika – P. Viola ir M. Jones *AdaBoost* veidų klasifikavimo metodas, kuris kombinacijoje su naiviojo Bajeso klasifikatoriaus technika sudarė įgyvendintos segmentavimo sistemos algoritminį pagrindą. Atlikta išsamesnė su praktine realizacija susijusių metodikų literatūros apžvalga, siekiant pagrįsti įgyvendintos sistemos unikalumą ir atlikti lyginamąją analizę.

AdaBoost metodika buvo darbo eigoje nuodugniau išnagrinėta tiek teoriškai (žr. sk.1.2), tiek praktiškai (žr. sk. 2.2). Praktiniam tyrimui naudojant *OpenCV* kompiuterinės regos bibliotekoje prieinamas klasikinio kaskados mokymo realizaciją, buvo praktiškai pagrįsta / paneigta dalis mokslinėje literatūroje skelbiamų teiginių apie *GAB*, *RAB*, *LAB* pranašumą segmentavimo ir našumo kriterijais palyginus su *DAB* algoritmu. Tokia tendencija pasitvirtino *GAB* ir *RAB* algoritmams, kai yra naudojami kelių viršūnių (nuo dviejų iki keturių) silpnieji *CART* medžių klasifikatoriai. Tačiau ši tendencija nėra akivaizdi *CART* kelmų taikymo atveju. *LAB* algoritmo pranašumas pasirodė nepagrįstas priešakinių veidų segmentacijos atveju, kadangi praktinių bandymų metu jis buvo prastesnis už visus tris konkurentus (*DAB*, *GAB* ir *RAB*) abiejų išvardintų kriterijų atžvilgiu. Tačiau visoms minėtoms „boosting“ modifikacijoms pasiteisino teorija, kad kelių viršūnių *CART* medžių naudojimas vietoje P. Viola ir M. Jones įprastų vieno požymio kelmų rezultate duoda mažiau požymių turintį našesnę klasifikatorių, kuris kartu dažnai pasižymi geresniais segmentavimo rezultatais. Išsamesnės teorinės ir praktinės analizės išvados pateikiamos atitinkamai skyriuose 1.3 ir 2.3.

Įgyvendintos segmentacijos sistemos realizacijos paprastumo dėlei buvo panaudoti *CART* kelmų silpnieji klasifikatoriai ir tokiu atveju geriausia tikslumo atžvilgiu *DAB* kaskada, kuriai mokinti pritaikyta praplėsta požymių tipų aibė (žr. sk. 3.2.1). Nors *DAB* surandamų veidų skaičiaus įverčiai iš esmės pateisino lūkesčius, išryškėjo ne itin tikslios veido lokalizacijos ir klaidingų veidų atmetimo segmentacijos fazėje problemos. Šioms problemoms spręsti sistema buvo papildyta antruoju surastų veidų pozicijų korekcijos žingsniu – naiviuoju Bajeso klasifikatoriumi (žr. sk. 1.5 ir 3.2.2), kuris naudoja glodintų vaizdų kvantuotų gradientų krypties informaciją. Ši išgaunama pritaikius A. Basčio pasiūlytą greitąjį simetrinį glodinantį eksponentinį filtrą (žr. sk. 3.2.2). Antrasis žingsnis sėkmingai susitvarkė su veidų lokalizacijos koregavimo užduotimi ir ženkliai (dvidešimčia kartų) pagerino veidų pozicionavimo tikslumo rezultatus. Tačiau kitos netesingų veidų eliminavimo problemos šiuo metu vienareikšmiškai išspręsti nepavyko. Išlaikant visus teisingus veidus, neteisingus veidus galima atmesti tik dalinai. Nors pastarajai iš išvardintų problemai spręsti įvertintas naiviojo Bajeso klasifikatoriaus tikimybinis atsakų pasiskirstymas gali būti pritaikytas tolimesnei skirtingai „abejotinų“ ir „užtikrintų“ veidų koregavimo strategijai realizuoti. (žr. sk. 3.3). Tai kartu su kitomis realizuotos segmentacijos metodikos tobulinimo idėjomis galėtų pagerinti sistemos patikimumą. Taip pat pateikiamos greičio optimizavimo gairės (žr. sk. 3.4).

Siekiant įsitikinti sistemos unikalumu ir produktyvumu buvo atlikta papildoma mokslinės literatūros analizė, ieškant kitų „boosting“ ir / arba naiviųjų Bajeso klasifikatorių taikymo veidų segmentacijos uždaviniui spręsti pavyzdžių. Verta paminėti, kad tiek *DAB*, tiek *NB* metodikų pritaikymas veidų segmentacijai nėra naujas (žr. sk. 1.5). Pavieniui kiekvienos iš jų taikymo atveju, *DAB* yra itin populiarus, *NB* – visiškai ne, nes, kaip ir dauguma kitų algoritmų, negali konkuruoti savo našumu su kaskada. Nėra populiarus ir *DAB* junginio su *NB* atvejis. Pastaruoju atveju *DAB* tiesiogiai įkomponuoja *NB* silpnuosius klasifikatorius stipriojo kaskados klasifikatoriaus mokymo procese (t.y. sąlaja atliekama *DAB* struktūros viduje mokymo metu). Darbe pateikta paprasta nuoseklus kombinavimo strategija tikslesnei lokalizacijai spręsti atrodo neturinti analogų. Dauguma publikacijų kombinuoja *DAB* ir *NB* ne tarpusavyje, o su kitomis įvairiomis metodikos, kur pirmoji naudoja net ne Haar o kitokių tipų požymius.

Papildoma mokslinių straipsnių apžvalga buvo reikalinga ir praktinei palyginamajai analizei atlikti (žr. sk. 3.2). Buvo nagrinėjami dažniausi išskirtinai priešakinių veidų segmentacijos (o ne atpažinimo) tikslumui įvertinti naudojami protokolai bei duomenų bazės, siekiant palyginti įgyvendintos sistemos rezultatus su kitų straipsnių pasiekimais. Dauguma straipsnių pateikia segmentacijos rezultatus įvairioms atpažinimo duomenų bazėms naudoja skirtingus vaizdų rinkinius ar jų poaibius tiek mokymo, tiek testavimo fazėje. Paaiškėjo, kad priešakinių veidų segmentacijai skirtų nedidelių standartinių duomenų bazių yra kelios: MIT/CMU ir naujesnė BioID. Buvo atlikti papildomi praktiniai realizuotos DAB ir NB sistemos testai ant šių bazių. Esminė praktinės lyginamosios analizės išvada – realizuota priešakinių veidų automatinės segmentacijos sistema teikia konkurencingą vidutiniškai $DR = 98.26\%$ veidų suradimo dažnį ir $TPR = 98.62\%$ tikrų veidų iš visų sistema gražintų veidų dažnį ant normalios raiškos veidų duomenų bazių 25% nuotolio nuo tikrų akių (13) rėžiuose, bei $DR = 67.93667\%$ ir $TPR = 93.35\%$ ant prastos kokybės skenuotų ir kt. MIT/CMU duomenų bazės poaibių to paties nuotolio rėžiuose. Rezultatai gauti remiantis 42331 veidų segmentacijos testais.

Pagrindinė šio darbo išvada – magistro studijų tiriamojo darbo užsibrėžtas tikslas, t.y. priešakinių (angl. *frontal*) veidų segmentacijos skaitmeniniuose vaizduose algoritmo sukūrimas, sėkmingai pasiektas. Pagrindiniam tikslui pasiekti buvo atliktas nuoseklus mokslinis tiriamasis darbas, susidedantis iš tokių žingsnių:

1. bendra automatinei veidų segmentacijai spręsti mokslinėje literatūroje aprašytų metodikų teorinė apžvalga ir analizė;
2. teorinės analizės rezultate geriausios metodikos išrinkimas ir jos praktinis tyrinėjimas;
3. teorinės analizės rezultate geriausios metodikos praktinis įgyvendinimas ir tolimesnis tyrinėjimas
4. realizuotos metodikos rezultatų tobulinimas kombinuojant su kitomis vaizdų apdorojimo (spartus simetrinis glodus eksponentinis filtras) bei klasifikavimo metodikomis (*NB* klasifikatorius);
5. papildoma mokslinės literatūros analizė, siekiant pagrįsti įgyvendintos sistemos unikalumą;
6. papildoma mokslinės literatūros analizė, siekiant atlikti įgyvendintos sistemos lyginamąją analizę;
7. praktinių testų palyginamajai analizei ir jų rezultatams pateikti įgyvendinimas.

Šiuo metu realizuota veidų segmentacija teikia labai gerus segmentacijos tikslumo įverčius ant normalios raiškos veidų duomenų bazių. Tačiau tai nėra visiškai išbaigta sistema pagrįdė dėl jos našumo įverčių. Sistemos antrąjį žingsnį siūloma tobulinti siekiant įgalinti jos taikymą realaus laiko biometrinėse verifikacijos / identifikacijos sistemose. Šiam tikslui įgyvendinti gali būti vadovaujama darbe pateiktomis algoritminio ir programinio pobūdžio optimizavimo gairėmis.

Literatūros sąrašas

- [TP91] M. A. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
- [MP97] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:696–710, 1997.
- [SP98] K.-K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.
- [RBK98] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–28, 1998.
- [OFG97] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. In *CVPR*, pages 130–136, 1997.
- [YRA00] M.-H. Yang, D. Roth, and N. Ahuja. A SNoW-based face detector. In *Proceedings of Neural Information Processing Systems*, pages 855–861, 2000.
- [SK00] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2000.
- [Sch00] H. Schneiderman. *A Statistical Approach to 3D Object Detection Applied to Faces and Cars (CMURI-TR-00-06)*. PhD thesis, RI, 2000.
- [VJ01a] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, 12-14 2001.
- [VJ01b] P. Viola and M. Jones. Robust real time object detection. In *IEEE ICCV Workshop on Statistical and Computational Theories of Vision*, Vancouver, 2001.
- [TV00] K. Tieu and P. Viola. Boosting image retrieval. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 228–235, 2000.
- [Liu03] C. Liu. A Bayesian discriminating features method for face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):725–740, 2003.
- [PMS94] A. P. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 84–91, 1994.
- [FBC00] J. Feraud, O. Bernier, and M. Collobert. A fast and accurate face detector for indexation of face images. In *Proc. Fourth IEEE Int. Conf on Automatic Face and Gesture Recognition*, Grenoble 2000.

- [CMC96] S. Gong, S. McKenna, and J. Collins. An investigation into face pose distribution. In Proc. IEEE International Conference on Face and Gesture Recognition, Vermont, 1996.
- [LGL00] Y. M. Li, S. G. Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In IEEE Int. Conf. On Face & Gesture Recognition, pages 300–305, Grenoble, 2000
- [NG99] J. Ng and S. Gong. Performing multi-view face detection and pose estimation using a composite support vector machine across the view sphere. In Proc. IEEE International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, pages 14–21, Corfu, 1999
- [HSW98] J. Huang, X. Shao, and H. Wechsler. Face pose discrimination using support vector machines (SVM). In Proceedings of International Conference Pattern Recognition, Brisbane, Queensland, Australia, 1998.
- [LZ04] S. Z. Li and Z. Zhang. FloatBoost learning and statistical face detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(9):1112–1123, 2004.
- [LZS+02] S. Z. Li, Z. Q. Zhang, H.-Y. Shum, and H. Zhang. FloatBoost learning for classification. In Proceedings of Neural Information Processing Systems, Vancouver, 2002.
- [LZZ+02] S. Z. Li, L. Zhu, Z. Q. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical learning of multi-view face detection. In Proceedings of the European Conference on Computer Vision, volume 4, pages 67–81, Copenhagen, Denmark, May 28 - June 2 2002
- [LKP02] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. MRL Technical Report, Intel Labs, December 2002.
- [FS97] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119–139, August 1997.
- [MKH05] T. Mita, T. Kaneko and O. Hori, Joint Haar-like Features for Face Detection, in Proc. of the International Conference on Computer Vision, p.1619-1626, 2005.
- [HL01] E. Hjelmås and B. K. Low. Face detection: A survey. Computer Vision and Image Understanding, 3(3):236–274, September 2001
- [YKA02] M.-H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: a survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1):34–58, 2002.
- [BP02] P.K. Biswas and M. Pandit, Opti-GVF Snake Model for Face Segmentation from Video Sequences, in Proc. ICVGIP, 2002.

- [GBA02] M. Gastaud, M. Barlaud, and G. Aubert. Combining shape prior and statistical features for active contour segmentation, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 5, pp. 726–734, May 2004
- [SPJ04] T. Sawangsri, V.Patanavijit and S. S. Jitapunkul. Face segmentation using novel skin-color map and morphological technique. In *Transactions on engineering, computing and technology V2*, 2004
- [LJ04] Stan Z. Li, Anil K. Jain, *Handbook of Face Recognition*, Springer, New York, 2004.
- [LM02] R. Lienhart and J. Maydt, An Extended Set of Haar-Like Features for Rapid Object Detection, *Proc. IEEE Int'l Conf. Image Processing*, vol. 1, pp. 900-903, 2002.
- [BAS+07] J. R. Beveridge, A. Alvarez, J. Saraf, W. Fisher, P. J. Flynn, J. Gentile. Face Detection Algorithm and Feature Performance on FRGC 2.0 Imagery, *First IEEE International Conference on Biometrics: Theory, Applications, and Systems*, pp. 1 – 7, 2007
- [PBD+04] S. L. Phung, A. Bouzerdoum, D. Chni, A.Watson. Naïve Bayes Face/Nonface Classifier: A Study of Preprocessing and Feature Extraction Techniques, *International Conference on Image Processing*, vol.2, pp. 1385 – 1388, 2004
- [ST98] H. Schneidennan, T. Kanade. Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition, in *Proc. of Conference on Computer Vision and Pattern Recognition*, pp. 45-51, 1998
- [PBC+03] S. L. Phung, A. Bouzerdoum, D. Chai, W. Kuczborski. A color-based approach to automatic face detection. *Proceedings of ISSPIT2003*, pp. 531 – 534, 2003
- [SM10] V.B. Subburaman, S. Marcel. An Alternative Scanning Strategy to Detect Faces, *IEEE International Conference on Acoustics Speech and Signal Processing*, pp. 2122 – 2125, 2010
- [Ker02] D. Keren. Painter Identification Using Local Features and Naïve Bayes. In *Proceedings of the 16th International Conference on Pattern Recognition*, vol. 2, pp. 474 – 477, 2002
- [ASM+03] H. H. Aviles-Arriaga, L. E. Sucar, C. E. Mendoza, C.E. B. Vargas. Visual Recognition of Gestures using Dynamic Naïve Bayesian Classifiers. *Proceedings of the 12th IEEE International Workshop on Robot and Human Interactive Communication*, pp. 133 – 138, 2003
- [WZM+10] F. Wang, D. Zhang, L. Man, J. Yu. A Naïve Bayesian Approach for Color Recognition of License Plates. *International Conference on Intelligent Control and Information Processing (ICICIP)*, pp. 154 – 159, 2010
- [LH09] S. M. Lajevardi, Z.M. Hussain. Feature Selection For Facial Expression Recognition Based on Optimization Algorithm. *INDS '09. 2nd International Workshop on Nonlinear Dynamics and Synchronization*, pp. 182 – 185, 2009

- [LNH09] S. M. Lajvardi, K. L. Neville, Z. M. Hussain. Facial Expression Recognition Over FFT-OFDM. ATC '09. International Conference on Advanced Technologies for Communications, pp. 35 – 38, 2009
- [SLC+02] N. Sebe, M. S. Lew. Emotion Recognition Using a Cauchy Naive Bayes Classifier. Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02), vol. 1, 2002
- [MMS09] D. Maturana, D. Mery and A. Soto. Face Recognition with Local Binary Patterns, Spatial Pyramid Histograms and Naïve Bayes Nearest Neighbor classification, International Conference of the Chilean Computer Science Society (SCCC), p. 125, 2009
- [Bal99] S. Baluja. Probabilistic Modeling for Face Orientation Discrimination: Learning from Labeled and Unlabeled Data. Proceedings of the 1998 conference on Advances in neural information processing systems II, 1999
- [ZHL+07] Z. Zhang, Y. Hu, M. Liu, T. Huang. Head Pose Estimation in Seminar Room using Multi View Face Detectors. Lecture Notes in Computer Science, vol. 4122, pp. 299 – 304, 2007.
- [RF05] G. A. Ramirez, O. Fuentes. Face Detection Using Combinations of Classifiers. The 2nd Canadian Conference on Computer and Robot Vision, pp. 610 – 615, 2005.
- [JKF01] O. Jesorsky, K. Kirchberg, R. W. Frischholz. Robust Face Detection Using the Hausdorff Distance. Third Inter-national Conference on Audio- and Video- based Biometric Person Authentication, Lecture Notes in Computer Science, pp. 90 – 95, Springer, 2001
- [OCV09] Open Source Computer Vision library (v2.1). URL: <http://opencv.willowgarage.com/wiki/Welcome>
- [OB09] OpenCV Boosting documentation. URL: <http://opencv.willowgarage.com/wiki/MachineLearning#Boosting>
- [OHT09] OpenCV Haartraining tutorial. URL: <http://note.sonots.com/SciSoftware/haartraining.html>.
- [WR09] Wikipedia. Receiver Operating Characteristics. URL: http://en.wikipedia.org/wiki/Receiver_operating_characteristic
- [WFS09] Wikipedia. Floyd-Steinberg Dithering. URL: http://en.wikipedia.org/wiki/Floyd-Steinberg_dithering
- [Bas10a] A. Bastys. „Vaizdų vidurkinimas ir požymių išskyrimas“, VU MIF „Biometrikos sistemos“ kurso konspektai, 2010. URL: <http://www.mif.vu.lt/~bastys/academic/ATE/biometrika/Vidur.pdf#Apibendrinime%20vienmačiu>

- [Bas10b] A.Bastys, „Klasifikavimo metodai“, VU MIF „Biometrikos sistemos“ kurso konspektai, 2010. URL: <http://www.mif.vu.lt/~bastys/academic/ATE/biometrika/Klasifikacija.pdf>
- [Wik11] Wikipedia. Bayes' theorem. URL: http://en.wikipedia.org/wiki/Bayes'_theorem
- [Abb10] Pieter Abbeel, Artificial Intelligence Course, University of California, 2010. <http://inst.eecs.berkeley.edu/~cs188/sp08/projects/project8/webpage/index.html>
- [Hua07] Jonathan Chung-Kuan Huang, „Naïve Bayes Classification“ presentation, 2007. URL: <http://www.cs.cmu.edu/~jch1/research/presentation/nbslides.ppt>
- [Neu11] Neurotechnology „VeriLook“ SDK veidų identifikavimo programinė įranga. URL: <http://www.neurotechnology.com/verilook.html>

Priedas Nr. 1

Veidų duomenų bazės:

- [FERET] „The Facial Recognition Technology (FERET) Database Grayscale“. URL: http://www.itl.nist.gov/iad/humanid/feret/feret_master.html
- [FRGC] „National Institute Of Standards and Technology Face Recognition Grand Challenge database (NIST FRGC)“. URL: <http://face.nist.gov/frgc/>
- [MIT/CMU] „The Combined MIT / CMU Test Set with Ground Truth for Frontal Face Detection.“ Carnegie Mellon University. URL: http://www.vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html
- [BioID] „The BioID Face Database“. URL: <http://www.bioid.com/downloads/facedb/index.php>
- [JAFFE] „The Japanese Female Facial Expression (JAFFE) Database“. URL: <http://www.kasrl.org/jaffe.html>.
- [BANCA] „The BANCA database“. URL: <http://www.ee.surrey.ac.uk/CVSSP/banca/>
- [XM2VTS] „The Extended M2VTS database“. URL: <http://www.ee.surrey.ac.uk/CVSSP/xm2vtsdb/>
- [EQUINOX] „Equinox Infrared Face Database“. URL: <http://www.equinoxsensors.com/products/HID.html>
- [DB09] Bendra informacija apie duomenų bazes:
- Veidų segmentacijos algoritmų testavimo duomenų bazių sąrašas. URL: <http://www.facedetection.com/facedetection/datasets.htm>
 - Veidų atpažinimo algoritmų testavimo duomenų bazių sąrašas: URL: <http://www.face-rec.org/databases/>.
 - Stan Z. Li, Anil K. Jain, Handbook of Face Recognition, Springer, New York, 2004, skyrius 13
Pastaba: nurodytoje knygoje plačiai apžvelgiamos visos egzistuojančios veidų duomenų bazės, jų sandara (skirtingų išraiškų, posūkių ir pan. reprezentacijų buvimas) bei pritaikymas konkretiems testams (pvz., tam tikros duomenų bazės naudojamos veidų segmentacijai, kitos – verifikacijai / identifikacijai testuoti), siekiant gebėti palyginti su kitų straipsniuose pateiktų metodikoms rezultatais.