

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
PROGRAMŲ INŽINERIJOS KATEDRA

Liudas Drejeris

**Turiniu paremtas paveikslėlių atpažinimas  
panaudojant SURF (Speeded-Up Robust Features)  
algoritmą ir skaidymą spalvomis**

Magistro darbas

Darbo vadovas

prof. E. Bareiša

**Kaunas, 2012**

KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
PROGRAMŲ INŽINERIJOS KATEDRA

Liudas Drejeris

**Turiniu paremtas paveikslėlių atpažinimas  
panaudojant SURF (Speeded-Up Robust Features)  
algoritmą ir skaidymą spalvomis**

Magistro darbas

Recenzentas

lekt. dr. Kęstutis Jančauskas  
2012-05-28

Vadovas

prof. Eduardas Bareiša  
2012-05-28

Atliko

IFME-0/2 gr. stud.  
Liudas Drejeris  
2012-05-28

**Kaunas, 2012**

# **CONTENT-BASED IMAGE RECOGNITION USING SURF (SPEEDED-UP ROBUST FEATURES) ALGORITHM AND COLOUR DECOMPOSITION**

## ***SUMMARY***

Search for matching areas between the two images, displaying the same object or composition of objects, is a difficult task, which is widely used in computer vision. Image registration, camera calibration, object recognition and tracking are only few of possible application fields.

The aim of this work was to analyze and modify SURF algorithm in order to improve its recognition accuracy. SURF algorithm has difficulties in comparing differently colored images with similar features. This happens because the SURF algorithm only processes monochrome images, relying on regions of the image intensity. As observed in practice, different color objects' intensities in image can be the same, if image is converted to black and white. In this case SURF algorithm makes wrong image matching. This work investigates one possible solution to the problem using image decomposition into RGB color channels. "Universal video recognition system" was created during this study and now it is installed in two companies.

As noted at the end of the study, improved SURF algorithm is more accurate and responsive to changes in the colors of objects in texture. It also better compares images in lower lighting conditions.

# **TURINIŲ PAREMTAS PAVEIKSLĖLIŲ ATPAŽINIMAS PANAUDOJANT SURF (SPEEDED-UP ROBUST FEATURES) ALGORITMĄ IR SKAIDYMĄ SPALVOMIS**

## ***SANTRAUKA***

Sutampančių vietų paieška tarp dviejų paveikslėlių, kuriuose atvaizduotas tas pats objektas ar kompozicija, yra sudėtinga užduotis, kuri plačiai taikoma kompiuterinio matymo srityje. Paveikslėlių registracija, kameros suderinimas, objektų atpažinimas ir sekimas - tai tik keletas iš galimų pritaikymo būdų.

Šio darbo tikslas ištirti ir pagerinti SURF algoritmo atpažinimo tikslumą. SURF algoritmas sunkiai atskiria paveikslėlius, turinčius panašių požymių, ir besiskiriančius tik spalvomis. Tai vyksta todėl, kad SURF algoritmas apdoroja tik nespalvotus paveikslėlius, pasiremdamas paveikslėlio regionų intensyvumais. Kaip pastebėta praktikoje, skirtingų spalvų objektų paveikslėlyje, pavertus paveikslėlį į nespalvotą, intensyvumai gali sutapti. Tokiu atveju SURF algoritmas neteisingai palygina požymius. Šiame darbe tiriamas vienas iš galimų problemos sprendimo būdų, panaudojant paveikslėlių skaidymą į spalvų kanalus RGB. Atliekant tyrimą buvo sukurta „Universali vaizdo atpažinimo sistema“, kuri šiuo metu yra įdiegta dviejose įmonėse.

Kaip pastebėta tyrimo pabaigoje, pagerintas SURF algoritmas veikia tiksliau ir reaguoja į spalvų pakitimus objektų tekstūroje. Taip pat geriau palygina paveikslėlius mažesnio apšviestumo sąlygomis.

## TURINYS

<i>SUMMARY</i> .....	3
<i>SANTRAUKA</i> .....	4
PAVEIKSLĖLIŲ SĄRAŠAS .....	8
LENTELIŲ SĄRAŠAS .....	11
TERMINŲ IR SANTRUMPŲ ŽODYNAS .....	12
1. ĮVADAS .....	14
1.1. Dokumento paskirtis .....	14
1.2. Problema .....	14
1.3. Tikslas ir uždaviniai .....	15
1.4. Dokumento struktūra.....	15
2. PAVEIKSLĖLIO POŽYMIŲ ANALIZĖ .....	16
2.1. Globalūs požymiai .....	16
2.2. Paveikslėlių segmentai .....	16
2.3. Atrinkti požymiai .....	17
2.4. Lokalūs požymiai .....	18
2.5. Lokalių požymių kategorijos.....	18
2.6. Idealaus lokalaus požymio savybės .....	20
3. POŽYMIŲ APTIKIMO ALGORITMŲ PALYGINIMAS .....	22
4. LOKALIŲ POŽYMIŲ ALGORITMO SURF VEIKIMO ANALIZĖ .....	25
4.1. SURF algoritmo veikimas.....	25
4.1.1. Dominančių vietų aptikimas .....	25
4.1.1.1. Integraliniai paveikslėliai .....	25
4.1.1.2. Hessian matrica pagrįsti dominantys taškai .....	25
4.1.1.3. Dominančių vietų nustatymas .....	27
4.1.2. Dominančių vietų aprašymai ir palyginimas .....	27
4.1.2.1. Požymių aprašas pagrįstas Haar bangelių sumomis.....	27
4.1.2.2. Greitas indeksavimas skirtas palyginimui.....	28
5. SPALVŲ IR JŲ ERDVIŲ ANALIZĖ .....	29
5.1. Spalvų suvokimas ir aprašymas .....	29
5.2. HSL spalvų erdvė.....	29
5.3. RGB spalvų erdvė .....	30

5.4.	CMY(K) (Žydra Violetinė Geltona (Juoda)) spalvų erdvė .....	31
6.	SURF IR SPALVŲ DEKOMPOZICIJOS APJUNGIMO APRAŠYMAS .....	32
7.	OBJEKTŲ ATPAŽINIMO SISTEMOS ARCHITEKTŪROS APRAŠYMAS.....	36
7.1.	Serverio dalies aprašymas .....	36
7.2.	Kliento dalies aprašymas.....	37
7.3.	Serverio architektūra .....	37
7.3.1.	Serverio komponentų diagramos .....	37
7.3.2.	SocketService komponentas .....	38
7.3.3.	MySQLService komponentas .....	39
7.3.4.	Actions komponentas.....	40
7.3.5.	ObjectRecognition komponentas .....	41
7.3.6.	Main komponentas.....	42
7.4.	Kliento architektūra.....	42
7.4.1.	Kliento komponentų diagrama.....	42
7.4.2.	LiveImageProcessingPackage komponentas .....	43
7.4.3.	ClientObjectRecognitionPackage komponentas .....	45
7.4.4.	MainPackage komponentas .....	46
7.5.	Duomenų bazės struktūra .....	47
7.6.	Sistemos įdiegimas.....	47
8.	SURF IR SPALVŲ DEKOMPOZICIJOS EKSPERIMENTAI .....	48
8.1.	Atsparumo pasukimui eksperimentas .....	48
8.2.	Atsparumo apšvietimo kitimui eksperimentas .....	52
8.3.	Gebėjimo atskirti spalvas eksperimentas .....	56
9.	IŠVADOS .....	60
10.	LITERATŪRA .....	61
11.	PRIEDAI.....	64
1	priedas. Požymių aptikimo algoritmų apžvalgos lentelė .....	65
2	priedas. Kitų palyginimo bandymuose panaudotų daiktų paveikslėliai .....	66
3	priedas. Kitų paveikslėlių palyginimo rezultatų lentelė.....	67
4	priedas. UAB „Informacinių technologijų organizacija“ programų sistemos perdavimo ir aprobavimo akto kopija .....	68

5 priedas. Valstybinė įmonė „Registru centras“ programų sistemos perdavimo ir aprobavimo akto kopija.....	69
6 priedas. KTU jaunųjų mokslininkų darbų parodos “Technorama” padėkos už sėkmingą dalyvavimą parodoje kopija.....	70

## PAVEIKSLĖLIŲ SĄRAŠAS

1 pav. Paveikslėlio segmentų išskyrimo pavyzdys .....	17
2 pav. Kampų aptikimo algoritmų rasti lokalūs požymiai [2]. .....	18
3 pav. SURF algoritmo požymių išskyrimo pavyzdys .....	22
4 pav. Paveikslėlio dalinimo į regionas pavyzdys panaudojant MSER algoritimą .....	24
5 pav. Trys reikalingi veiksmai siekiant apskaičiuoti kvadratinės zonos intensyvumą.....	25
6 pav. Kairėje pusėje: originali (diskretizuota ir apkarpyta) Gauso antros eilės dalinė išvestinė $y - (L_{yy})$ ir $xy - (L_{xy})$ kryptimis; dešinėje pusėje: SURF algoritme naudojama Gauso antros eilės dalinė išvestinė $y - (D_{yy})$ ir $xy - (D_{xy})$ . Pilki regionai yra lygūs nuliui. ....	26
7 pav. Dominančios vietos aptiktos saulėgražų lauke. ....	27
8 pav. Kvadratinis regionas išskaidytas į smulkesnius 4x4 kvadratinis vidinius regionus.....	28
9 pav. Šviesios ir tamsios dėmių pavyzdžiai. ....	28
10 pav. HSL spalvų erdvės grafinis atvaizdavimas. ....	30
11 pav. Grafinis RGB spalvų erdvės atvaizdavimas.....	30
12 pav. CMY(K) spalvų erdvės grafinis atvaizdavimas. ....	31
13 pav. Dviejų spalvotų kortelių paveikslėliai, kurie aiškiai skiriasi vienas nuo kito. ....	32
14 pav. Pirmas spalvotas kortelės paveikslėlis atskirtas į spalvų kanalus RGB (R – raudonas, G – žalias, B - mėlynas) išsaugant tik kanalo ryškumą. ....	32
15 pav. Antras spalvotas kortelės paveikslėlis atskirtas į spalvų kanalus RGB (R – raudonas, G – žalias, B - mėlynas) išsaugant tik kanalo ryškumą. ....	33
16 pav. SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, raudonos spalvos kanale (sutapimas 65,52 %). ....	33
17 pav. SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, žalios spalvos kanale (sutapimas 32,65 %). ....	34
18 pav. SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, mėlynos spalvos kanale (sutapimas 33,33 %). ....	34
19 pav. Spalvoti kortelių paveikslėliai paversti nespalvotais prieš apdorojant su SURF algoritmu. ....	34
20 pav. SURF algoritmo aptikti ir sutampantys pirmos ir antros kortelių paveikslėlių lokalūs požymiai (sutapimas 96,97 %). ....	35
21 pav. Bendra sistemos paketų diagrama [22]. ....	37
22 pav. SocketService komponento klasių diagrama [22]. ....	38



23 pav. MySQLService komponento diagrama [22].	39
24 pav. Actions komponento paketų diagrama [22].	40
25 pav. ObjectRecognition komponento klasių diagrama [22].	41
26 pav. Main komponento klasių diagrama [22].	42
27 pav. Kliento pagrindinė komponentų diagrama [22].	42
28 pav. LiveImageProcessingPackage komponento diagrama [22].	43
29 pav. ClientObjectRecognitionPackage komponento diagrama [22].	45
30 pav. MainPackage komponento diagrama [22].	46
31 pav. Duomenų bazės struktūra [22].	47
32 pav. Etaloninis objektas (puodelis) sutapimo pasukant bandymui.	48
33 pav. Etaloninis objektas (arbatos pakelis) sutapimo pasukant bandymui.	48
34 pav. Pasukto nuo 5° iki 90° etaloninio objekto (puodelio) paveiksliai.	49
35 pav. Pasukto nuo 5° iki 50° etaloninio objekto (arbatos pakelio) paveiksliai.	49
36 pav. Sutapimas su etalonu pasukant objektą (puodelis).	50
37 pav. Palyginimo laikas pasukant objektą (puodelis).	50
38 pav. Sutapimas su etalonu pasukant objektą (arbatos pakelis).	51
39 pav. Palyginimo laikas pasukant objektą (arbatos pakelis).	51
40 pav. Etaloninis objektas (puodelis) sutapimo keičiant apšvietimą bandymui.	52
41 pav. Etaloninis objektas (arbatos pakelis) sutapimo keičiant apšvietimą bandymui.	52
42 pav. Skirtingai apšviesto etaloninio objekto (puodelio) paveiksliai.	53
43 pav. Skirtingai apšviesto etaloninio objekto (arbatos pakelio) paveiksliai.	53
44 pav. Sutapimas su etalonu keičiant objekto apšvietimą (puodelis).	54
45 pav. Palyginimo laikas keičiant objekto apšvietimą (puodelis).	54
46 pav. Sutapimas su etalonu keičiant objekto apšvietimą (arbatos pakelis).	55
47 pav. Palyginimo laikas keičiant objekto apšvietimą (arbatos pakelis).	55
48 pav. Spalvoti objektų paveikslėliai.	56
49 pav. SURF apdoroti objektai.	57
50 pav. SURF RGB apdoroti objektai (raudonos spalvos kanalas).	58
51 pav. SURF RGB apdoroti objektai (žalios spalvos kanalas).	58
52 pav. SURF RGB apdoroti objektai (mėlynos spalvos kanalas).	59
53 pav. Pirmas papildomas palyginimo objektas – puodelis.	66
54 pav. Antras papildomas palyginimo objektas – knyga.	66

55 pav. Trečias papildomas palyginimo objektas – puodelis. .... 66

## LENTELIŲ SĄRAŠAS

1 lentelė. SocketService komponento metodų aprašymas [22] .....	38
2 lentelė. MySQLService komponento metodų aprašymas [22] .....	39
3 lentelė. Actions paketo metodų aprašymas [22] .....	40
4 lentelė. ObjectRecognition komponento metodų aprašymas [22] .....	41
5 lentelė. Main komponento metodų aprašymas [22] .....	42
6 lentelė. LiveImageProcessingPackage komponento metodų aprašymas [22] .....	43
7 lentelė. ClientObjectRecognitionPackage komponento metodų aprašymas [22] .....	45
8 lentelė. MainPackage komponento metodų aprašymas [22] .....	46
9 lentelė. Sutapimas su etalonu pasukant objektą (puodelis) ir palyginimo laikas.....	49
10 lentelė. Sutapimas su etalonu pasukant objektą (arbatos pakelis) ir palyginimo laikas .....	50
11 lentelė. Sutapimas su etalonu keičiant objekto apšvietimą (puodelis) ir palyginimo laikas.....	53
12 lentelė. Sutapimas su etalonu keičiant objekto apšvietimą (arbatos pakelis) ir palyginimo laikas .....	54
13 lentelė. Požymių aptikimo algoritmų apžvalga.....	65
14 lentelė. Kitų paveikslėlių palyginimo rezultatai .....	67

## TERMINŲ IR SANTRUMPŲ ŽODYNAS

1. **RGB** (angl. Red Green Blue) – spalvų erdvė sudaryta iš trijų dedamųjų - raudonos, žalios ir mėlynos.
2. **SURF** (angl. Speeded-Up Robust Features) – lokalių paveikslėlio požymių aptikimo ir palyginimo algoritmas.
3. **DOG** (Difference of Gaussians) – algoritmas skirtas paveikslėlių požymių aptikimui, kai palyginami du etaloninio objekto paveikslėliai, vienas labiau sulietas, kitas mažiau.
4. **Lokalūs požymiai** (angl. local features) – paveikslėlio dalys kurios išsiskiria iš bendro paveikslėlio konteksto tam tikromis savybėmis (spalva, intensyvumu, forma).
5. **Objektas** (angl. object) – briaunų, linijų, kampų ir spalvų visuma.
6. **Etaloninis objektas** (angl. reference object) – objektas, kuris pasirinktas kaip pagrindinis ir su juo lyginami kiti objektai.
7. **Objektų atpažinimas** (angl. object recognition) – veiksmų seka, kurios metu atpažįstamas.
8. **Regionas** (angl. region) – paveikslėlio smulkesnė dalis.
9. **Ryškusumas** (angl. brightness) – žmogaus pojūtis, kuris nusako ar vieta skleidžia daugiau ar mažiau šviesos.
10. **Atspalvis** (angl. hue) – žmogaus pojūtis, kurio pagalba vieta atrodo kaip viena ar dviejų spalvų mišinys iš šių spalvų: raudonos, geltonos, žalios ir mėlynos.
11. **Spalvingumas** (angl. colourfulness) – žmogaus pojūtis, kuris parodo ar vieta atskleidžia daugiau ar mažiau savo atspalvio.
12. **Šviesumas** (angl. lightness) – pojūtis, kuris parodo vietos *ryškumą* priklausantį nuo toje kompozicijoje esančios baltos spalvos *ryškumo*.
13. **Chromatiškumas** (angl. chroma) – vietos spalvingumą priklausantį nuo toje kompozicijoje esančios baltos spalvos *ryškumo*.
14. **Prisotinimas** (angl. saturation) – vietos spalvingumą priklausantį nuo jos ryškumo.
15. **Pasikartojamumas** (angl. repeatability) – turint du vienodus objekto ar kompozicijos paveikslėlius, padarytus skirtingomis sąlygomis (pvz. skiriasi apšvietimas), dauguma Lokalių požymių rastų viename paveikslėlyje turi sutapti su požymiais rastais kitame.

16. **Skirtingumas/informatyvumas** (angl. distinctiveness/informativeness) – intensyvumo struktūros, kuriomis aprašomi rasti požymiai turi būti labai įvairios, siekiant didesnio požymių skirtumo ir galimybės juos teisingai palyginti.
17. **Lokalumas** (angl. locality) – požymiai turi būti lokalūs, siekiant sumažinti klaidos galimybę, dėl paveikslo dalių uždengimo.
18. **Kiekis** (angl. quantity) – aptiktų požymių kiekis turi būti pakankamai didelis, kad požymiai būtų aptikti net ir ant pakankamai mažų paveikslėlio objektų.
19. **Tikslumas** (angl. accuracy) – aptikti požymiai turėtų būti tiksliai lokalizuoti paveikslėlyje, nepriklausomai nuo dydžio ir galimai nuo formos pasikeitimo.
20. **Efektyvumas** (angl. efficiency) – pageidautina, kad požymių aptikimas paveikslėlyje būtų pakankamai greitas ir tiktų realaus laiko sistemose.
21. **Invariantiškumas** (angl. invariance) – kai tikimasi didelių deformacijų, pageidautinas būdas būtų, matematiškai sumodeliuoti šias deformacijas ir vėliau sukurti požymių išskyrimo metodus, kurie būtų nepaveikiami šių matematinių transformacijų.
22. **Stiprumas** (angl. robustness) – jei tikimasi santykinai mažų deformacijų, dažnai pakanka metodų mažiau reaguojančių į deformacijas, tikslumą galima dalinai sumažinti.

# 1. ĮVADAS

## 1.1. Dokumento paskirtis

Dokumentas yra „Programų sistemų inžinerijos“ magistrantūros baigiamasis darbas. Dokumento paskirtis - apibendrinti atliktą literatūros analizę, supažindinti su tiriamuoju objektu bei jo problemine sritimi, pateikti galimą problemos sprendimo būdą, aprašyti atliktus tyrimus ir jų rezultatus.

## 1.2. Problema

Sutampančių vietų paieška tarp dviejų paveikslėlių, kuriuose atvaizduotas tas pats objektas ar kompozicija, yra sudėtinga užduotis, kuri plačiai taikoma kompiuterinio matymo srityje. Paveikslėlių registracija, kameros suderinimas, objektų atpažinimas ir sekimas - tai tik keletas iš galimų pritaikymo būdų.

Išskirtinių paveikslėlio vietų paieška gali būti suskirstyta į tris pagrindinius žingsnius: visų pirma, dominančios vietos (*angl. interest points*) parenkamos išskirtinėse paveikslėlio pozicijose, tokiose kaip kampai, dėmės ir T-sandūros (*angl. T-junctions*). Šiame etape svarbiausia aptikimo algoritmo savybė - pakartojamumas (*angl. repeatability*). Ja nusakomas dominančių vietų aptikimo patikimumas, kai kinta paveikslėlio matymo sąlygos (apšvietimas, matymo kampas). Vėliau, kiekvienos parinktos dominančios vietos aplinka aprašoma požymių vektoriumi. Šie aprašymai turi būti skirtingi ir tuo pačiu metu būti atsparūs paveikslėlyje esantiems triukšmams, geometrinėms deformacijoms ir šviesos lygio pakitimams. Pabaigoje, aprašymų vektoriai iš skirtingų paveikslėlių yra sulyginami tarpusavyje. Palyginimas grindžiamas atstumu tarp vektorių, dažniausiai tai Euklido atstumas (*angl. Euclidean distance*). Aprašančio vektoriaus matmuo (*angl. dimension*) turi tiesioginę įtaką palyginimo laikui, pageidaujama naudoti mažų matmenų vektorius norint atlikti greitus palyginimus. Tačiau mažų matmenų, paveikslėlio dominančias vietas, aprašantys vektoriai paprastai būna neišskirtiniai. Siekiant tikslumo, verta naudoti didelių matmenų aprašančius vektorius.

Magistro studijų metu kurtoje „Universalioje vaizdų atpažinimo sistemoje“ buvo pastebėta, jog SURF algoritmas sunkiai atskiria paveikslėlius, turinčius panašių požymių, ir besiskiriančius tik spalvomis. Tai vyksta todėl, kad SURF algoritmas apdoroja tik nespaltvotus paveikslėlius. Šiame darbe aprašomas vienas iš galimų problemos sprendimo būdų, panaudojant paveikslėlių skaidymą į spalvų kanalus RGB.

### **1.3. Tikslas ir uždaviniai**

Šio darbo tikslas pagerinti SURF algoritmo, veikiančio pagal prieš tai išvardintus tris žingsnius, atpažinimo tikslumą. Siekiama, jog pagerintas algoritmas tiksliau atpažintų spalvotus objektus ir būtų atsparesnis šviesos pakitimams paveikslėliuose.

Pagrindiniai darbo uždaviniai būtų šie:

- Atlikti panašių į SURF algoritmų palyginimą;
- Išsiaiškinti kaip veikia SURF algoritmas;
- Pasirinkti tinkamiausią spalvų erdvę, kurioje paveikslėlis bus dalinamas į kanalus;
- Atlikti SURF algoritmo modifikaciją;
- Atlikti eksperimentus su modifikuotu SURF algoritmu ir išsiaiškinti ar modifikacijos pagerino algoritmo tikslumą.

### **1.4. Dokumento struktūra**

Šis dokumentas sudarytas iš 12 dalių.

- Darbo pradžioje pateikiama santrauka anglų ir lietuvių kalbomis.
- Pirmoje dalyje pateikiama šio dokumento paskirtis, tikslai ir keliami problema bei galimas jos sprendimo būdas.
- Antroje - penktoje dalyse atliekama literatūros analizė, kurioje aprašomas SURF algoritmo veikimas, lokalūs ir globalūs požymiai, pateikiamas atpažinimo algoritmų palyginimas bei aptariamos spalvų erdvės.
- Šeštoje dalyje pateikiamas SURF algoritmo atpažinimo tikslumo pagerinimo metodas, pritaikant skaidymą į spalvų kanalus RGB.
- Septintoje dalyje aprašoma „Universali vaizdo atpažinimo sistema“, kurioje šis algoritmo pagerinimas buvo pritaikytas.
- Aštuntoje dalyje aprašomi pagerinto SURF algoritmo bandymai ir gauti rezultatai.
- Devintoje dalyje pateikiamos šio darbo išvados.
- Dešimtoje dalyje pateikiami darbe naudoti literatūros šaltiniai.
- Vienuoliktoje dalyje paaiškinami darbe naudojami terminai ir santrumpos.
- Dvyliktoje dalyje pateikiami su šiuo darbu susiję priedai.

## 2. PAVEIKSLĖLIO POŽYMIŲ ANALIZĖ

### 2.1. Globalūs požymiai

Vaizdų paieškos srityje, buvo pasiūlyta daug globalių požymių, kurie nusako paveikslėlio turinį pasiremiant spalvų histogramomis ir jų variacijomis [17]. Šis būdas veikia palyginus gerai, bent jau su tais paveikslėliais, kurie turi labai ryškią ir išskirtinę spalvų gamą, taip pat jei vartotojas domisi visu paveikslėliu kaip kompozicija, o ne jo atskiomis dalimis. Globalūs požymiai nesugeba atskirti paveikslėlio fono nuo jo detalių, todėl visa informacija sumaišoma tarpusavyje.

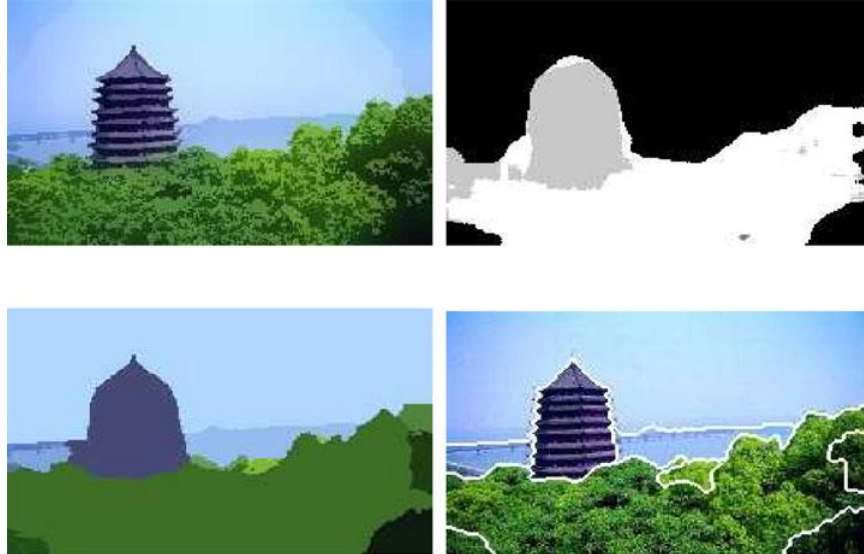
Globalūs požymiai taip pat buvo bandomi naudoti objektų atpažinimui, iš ko ir išsivystė pirmieji išvaizda pagrįsti (*angl. appearance-based*) metodai, kurie ir pabandė išspręsti sudėtingą vaizdų atpažinimo problemą. Turk ir Pentland [18] bei vėliau Murase ir Nayar [9] pasiūlė pirma atlikti pagrindinių paveikslėlių sudarančių komponentų analizę, ištiriant pasirinktą kiekį etaloninių paveikslėlių ir sutampančią informaciją panaudoti kaip tų paveikslėlių požymių aprašymus (*angl. descriptors*). Palyginus su ankstesniais, pagrįsti geometriniais elementais, bandymais jų pasiūlyto sprendimo rezultatas buvo labai novatoriškas ir efektyvus. Šio metodo pagalba buvo galima aptikti žymiai didesnę kiekį natūralių objektų. Tačiau, metodai pagrįsti globaliais požymiais yra labai jautrūs paveikslėlių turinio pasiskirstymui, iš ko vėl kyla didelė problema, apribojanti paveikslėlius apdorojančių sistemų naudingumą iki darbo su paveikslėliais turinčiais švarų foną ar galimybę aiškiai išskirti objektus, kaip pavyzdys sistemos tam panaudojančios paveikslėlyje aptiktą, objektų judesio informaciją.

### 2.2. Paveikslėlių segmentai

Bandymas apeiti globalių požymių trūkumus, tai paveikslėlio padalinimas į ribotą kiekį regionų ar segmentų, kurių kiekvienas atspindi tam tikrą paveikslėlio objektą ar jo dalį (1 pav.). Geriausiai žinomas pavyzdys yra dėmių pasaulio sistema (*angl. Blobworld system*), pasiūlyta [4] straipsnyje. Jos pagalba paveikslėlis pagal spalvas ir tekstūrą (geometrinių figūrų kontūrus) yra suskirstomas į segmentus, pagal kuriuos vėliau atliekamos panašių paveikslėlių paieškos duomenų bazėje. Pavyzdys, pagrįstas paveikslėlio padalinimu į segmentus pagal tekstūrą, pateikiamas [11] straipsnyje, aprašančiame plačių pagrindinių linijų sutapimo patikrinimą (*angl. wide baseline matching*).



Bet vėl gi, kyla problema, nes paveikslėlių padalinimas į segmentus yra labai sudėtingas uždavinys, kuris reikalauja gilesnio supratimo apie paveikslėlio turinį. Dažniausiai įprastiems, dažniausiai sutinkamiems objektams vien spalvos ir tekstūros informacijos neužtenka, norint gauti reikšmingą padalinimą į segmentus.



**1 pav.** Paveikslėlio segmentų išskyrimo pavyzdys

### 2.3. Atrinkti požymiai

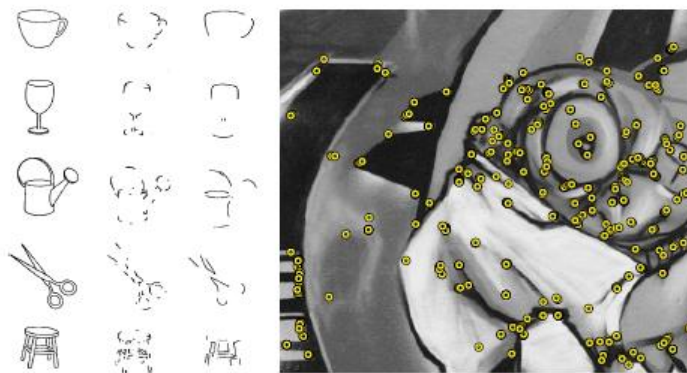
Vienas iš būdų susidoroti su globalių požymių problemomis, kylančiomis dalinant paveikslėlius į segmentus, yra daug laiko reikalaujantis, skirtingų paveikslėlio dalių, esančių kiekvienoje paveikslėlio pozicijoje ir kiekviename mastelyje, atrinkimo procesas. Tokiu atveju kiekvienai atrinktai paveikslėlio daliai galima paskaičiuoti globalius požymius. Šis metodas taip pat žinomas kaip „slenkančio lango“ (*angl. „Sliding Window“*) metodas. Jis buvo ypatingai populiarus veidų atpažinimo kontekste, bet taip pat buvo naudojamas ir atpažįstant specifinius objektus ar objektų grupes, tokias kaip pėstieji ar mašinos. Šie metodai, taikomi paveikslėlio dalims, gali nustatyti panašumus tarp užklausų ir etaloninių paveikslėlių nepriklausomai nuo besikeičiančio fono ar net jeigu objektas padengia sąlyginai mažą paveikslėlio dalį. Tokių metodų pagrindinis trūkumas, tai nesugebėjimas susidoroti su daliniu objekto uždengimu bei galimas aptinkamų formų įvairumas yra mažesnis, nei naudojantis lokaliais požymiais pagrįstais metodais. Nepaisant to, didžiausia problema yra labai neefektyvus panaudojimas. Kiekviena mažesnė paveikslėlio dalis turi būti patikrinta ir išanalizuota, dėl to tenka peržiūrėti tūkstančius ar

net milijonus paveikslėlio dalių. Tai reikalauja ypatingai efektyvių metodų, kurie stipriai apriboja galimų taikymų sritis.

Siekiant išvengti šių sudėtingumo problemų, daug retesnis fiksuoto tinklelio mėginių (*angl. fixed grid sampling*) metodas buvo naudojamas paveikslėlio sudalinimui (pvz. [3, 5, 19, 20]). Bet koku atveju, šiam metodui sunku pasiekti geometrinių deformacijų, susijusių su objekto požymiais, invariantiškumą.

## 2.4. Lokalūs požymiai

Lokalus požymis (*angl. local feature*), tai paveikslo ar nuotraukos raštas, kuris skiriasi nuo jį supančių gretimų raštų. Jis dažniausiai susijęs su paveikslėlio savybės ar keleto savybių tarpusavio pasikeitimu, tačiau lokalus požymis nebūtinai gali būti išskiriamas tiksliai pagal rašto savybių pasikeitimą. Paveikslėlių savybės dažniausiai būna spalva, intensyvumas ir tekstūra. Kairėje pusėje atvaizduojamas, lokalių požymių, esančių kontūriniam paveikslėlyje, pavyzdys, o dešinėje pusėje - esančių nespalvotame paveikslėlyje (2 pav.). Lokaliais požymiais gali būti ne tik nuotraukos taškai, bet ir briaunos, ar maži tekstūros lopinėliai. Dažniausiai atliekama keletas matavimų, sutelktų aplink lokalių požymių, o jų rezultatai paverčiami to požymio aprašymu (*angl. descriptors*). Tokie lokalių požymių aprašymai vėliau gali būti naudojami įvairiose paveikslėlių apdorojimo programose.



2 pav. Kampų aptikimo algoritmų rasti lokalūs požymiai [2].

## 2.5. Lokalių požymių kategorijos

Lokalūs (nekintantys) požymiai yra tikslus paveikslėlio aprašymo būdas, kuris sėkmingai ir labai plačiai taikomas programinėje įrangoje ir įvairiose sistemose.

Galima išskirti tris pagrindines lokalių požymių kategorijas, pagrįstas jų panaudojimu. Tai nėra vienintelis jų suskirstymo į kategorijas būdas, tačiau taip skirstant parodomi galimi jų panaudojimo būdai.

- Pirma kategorija, tai specifiniai lokalūs požymiai, kurie gali būti semantiškai interpretuojami pasiremiant jų panaudojimo sritimi. Kaip pavyzdžiui, kraštinės, aptiktos iš aukščio darytose nuotraukose, dažniausiai atspindi kelius. Dėmių aptikimas gali būti panaudotas defektų atpažinimui atliekant patikras ir panašiai. Tokie buvo pirmieji taikymai, kuriuose panaudoti lokalūs požymiai.
- Antra kategorija, tai lokalūs požymiai su aiškiai išskirta pozicija ir gali būti naudojami kaip lengvai atpažįstami kertiniai taškai. Šiuo atveju nelabai svarbu, ką atspindi vienas ar kitas lokalus požymis, svarbiausia, kad galima nustatyti jo tikslią vietą, kuri nesikeičia laiko atžvilgiu. Kaip pavyzdys, tai dažniausiai naudojama paveikslėlių palyginimuose ir objektų sekimo sistemose ir, ypatingai naudinga, kalibruojant kameras bei atliekant 3D rekonstrukcijas. Kitos sistemų sritys apima pozų numatymą, paveikslėlių sulygiavimą ar mozaikų sudarymą. Tipinis tokių lokalių požymių panaudojimo pavyzdys panaudojamas KLT sekime [16].
- Paskutinė kategorija, tai lokalūs požymiai, naudojami kaip greitas paveikslėlio atitikmuo, kuris leidžia atpažinti objektus ar kompozicijas nenaudojant paveikslėlio sudalinimo į segmentus. Šiuo atveju, kaip ir prieš tai buvusiu, nėra svarbu, ką požymiai atspindi. Taip pat nėra būtina žinoti ir jų tikslią vietą paveikslėlyje, nes pagrindinis tikslas ne šių požymių palyginimas vienas su kitu, bet jų statistinė analizė. Toks lokalių požymių panaudojimas pirmą kartą buvo aprašytas [13, 12] ir labai greit išpopuliarėjo, ypač objektų atpažinimo srityje (tiek specifinių objektų atpažinime, tiek ir objektų skirstyme į kategorijas). Kiti panaudojimo būdai apima klasifikaciją, tekstūros analizę ir filmuotos medžiagos tyrimus.

Iš prieš tai aprašytų kategorijų, aiškiai matosi, kad kiekvienas lokalių požymių panaudojimo būdas turi savo aiškias ribas, todėl geras požymis naudojamas vienoje sistemoje gali būti visiškai bevertis kitos sistemos kontekste. Šios kategorijos gali būti panaudotos ieškant tinkamo lokalių požymių išskyrimo pasirinktai sričiai. Šiame magistro darbe didžiausias dėmesys skirtas trečios kategorijos lokaliems požymiams.

Taip pat verta paminėti, kad lokalių požymių svarba buvo pademonstruota objektų atpažinime pagrįstame žmonių rega ir suvokimu [2]. Eksperimentai parodė, jog iš paveikslėlio panaikinus

kampus, žmonės žymiai sunkiau atpažįsta nupieštą objektą, nei pašalinus tiesias linijas. 1 pav. kairėje pusėje galima matyti šio bandymo iliustraciją.

## 2.6. Idealaus lokalaus požymio savybės

Lokalūs požymiai paprastai turi aplink juos esančią erdvę, kitaip tariant lokalių taškų erdvę (*angl. local neighborhood of pixels*). Kaip priešprieša paveikslėlių dalinimui į segmentus, tai gali būti bet kuri paveikslėlio dalis. Erdvės ribos neprivalo reaguoti į paveikslėlio išvaizdos pasikeitimus, tokius kaip spalva ar tekstūra. Taip pat, kelios erdvės gali persidengti.

Idealiu atveju, lokalūs taškai turėtų atspindėti semantiškai prasmingas paveikslėlio dalis. Tačiau praktikoje tai neįmanoma, nes tai reikalauja aukšto lygio turinio interpretavimo, kuris šiuo metu dar nėra pasiektas. Vietoje to, lokalūs požymiai yra parenkami tiesiogiai remiantis paveikslėlio intensyvumo struktūra.

Gerai požymiai turėtų turėti tokias savybes:

- *Pakartojamumas*: Turint du vienodus objekto ar kompozicijos paveikslėlius, padarytus skirtingomis sąlygomis (pvz. skiriasi apšvietimas), dauguma lokalių požymių rastų viename paveikslėlyje turi sutapti su požymiais rastais kitame.
- *Skirtingumas/informatyvumas*: Intensyvumo struktūros, kuriomis aprašomi rasti požymiai turi būti labai įvairios, siekiant didesnio požymių skirtumo ir galimybės juos teisingai palyginti.
- *Lokalumas*: Požymiai turi būti lokalūs, siekiant sumažinti klaidos galimybę, dėl paveikslo dalių uždengimo. Taip pat turi leisti nesudėtingus modeliavimus reikalingus dėl geometrinių ar fotografijos deformacijų, kurios kyla pakeitus matymo kampą.
- *Kiekis*: Aptiktų požymių kiekis turi būti pakankamai didelis, kad požymiai būtų aptikti net ir ant pakankamai mažų paveikslėlio objektų. Tačiau, požymių kiekis priklauso nuo taikymo srities. Požymių kiekis turėtų būti valdomas dideliame intervale panaudojant nesudėtingą ribą. Požymių tankumas turėtų nesunkiai atspindėti paveikslėlio turinį.
- *Tikslumas*: Aptikti požymiai turėtų būti tiksliai lokalizuoti paveikslėlyje, nepriklausomai nuo dydžio ir galimai nuo formos pasikeitimo.
- *Efektyvumas*: Pageidautina, kad požymių aptikimas paveikslėlyje būtų pakankamai greitas ir tiktų realaus laiko sistemose.

Nors dėl to galima nesutarti, tačiau pasikartojimas yra viena svarbiausių savybių, kuri gali būti pasiekama dviem būdais: per invariantiškumą arba per algoritmo atsparumą.

- *Invariantiškumas*: Kai tikimasi didelių deformacijų, pageidautinas būdas būtų, matematiškai sumodeliuoti šias deformacijas ir vėliau sukurti požymių išskyrimo metodus, kurie būtų nepaveikiami šių matematinių transformacijų.
- *Atsparumas*: Jei tikimasi santykinai mažų deformacijų, dažnai pakanka metodų mažiau reaguojančių į deformacijas, tikslumą galima dalinai sumažinti. Tipinės deformacijos, tai paveikslėlio triukšmai, suspaudimo netikslumai, suliejimas ir pan.

### 3. POŽYMIŲ APTIKIMO ALGORITMŲ PALYGINIMAS

Požymių aptikimo algoritmai yra suskirstyti į 4 grupes atsižvelgiant į jų invariantiškumą objekto pasukimui, dydžiui, uždengimui ir perspektyvai. Jų savybės palyginamos kiekvienoje grupėje. 1-ame priede pateikiame lentelę su požymių išskyrimo algoritmų svarbiausių charakteristikų palyginimais.

Pasukimo invariantiškumo požymių grupėje, didžiausią pasikartojimą ir tikslią lokalizaciją pasiekia Harris aptikimo algoritmas. Hessian aptikimo algoritmas suranda dėmes, kurios nėra taip gerai lokalizuotos ir reikalauja antrosios eilės išvestinių paskaičiavimo. SUSAN aptikimo algoritmas išvengia išvestinių skaičiavimo ir yra gerai žinomas dėl savo efektyvumo, tačiau neturėdamas paveikslėlio glotninimo jis tampa labiau jautrus triukšmams. Visi metodai turintys pasukimo invariantiškumą yra labiau tinkami taikyti ten, kur naudojama požymių vieta erdvėje ir nesitikima didelių dydžio pasikeitimų, pvz.: judesio struktūrose ar kamerų kalibravimuose.

Grupėje su dydžio invariantiškumu, Harris-Laplace algoritmas parodo didelį pasikartojimą ir lokalizacijos tikslumą paveldėtą iš Harris aptikimo algoritmo [14]. Tačiau jo mastelio įvertinimas mažiau tikslus, dėl įvairių mastelių kampų prigimties. Hessian-Laplace yra stipresnis, nei jo vieno mastelio versija [8]. Taip yra dėl to, kad dėminės struktūros geriau lokalizuojamos keičiantis masteliui, nei kampai. DoG ir SURF požymių aptikimo algoritmai buvo sukurti atsižvelgiant į našumą, todėl kitos savybės kiek nukentėjo. Vis dėl to daugumai taikymų jų daugiau nei pakanka. Taip pat rašoma, jog SURF algoritmas (3 pav.) veikia, daugiau nei penkis kartus, greičiau už DoG algoritmą.



3 pav. SURF algoritmo požymių išskyrimo pavyzdys

Požymių kiekis ir geras paveikslėlio padengimas yra kritinės vietos atpažinimo taikymuose, kuriuose lokalizavimo tikslumas mažiau svarbus. Taigi, Hessian-Laplace aptikimo algoritmai buvo sėkmingi įvairiose grupavimo srityse, nors yra ir algoritmų su geresniu pasikartojimo parametru. Atsitiktinė ir tanki atrankos, taip pat, duoda gerus rezultatus, kas parodo pakankamą padengimo reikalavimą, keliamą atpažinimo metodams [10], nors jų rezultatai, nėra pakankamai glausti.

Reiktų atsižvelgti į tai, jog metoduose nepriklausomuose nuo mastelio ir panašumų, skirtumas tarp kampų ir dėmių aptikimo tampa vis mažesnis, nes dauguma tokių metodų gali aptikti abiejų tipų požymius, tačiau vis tiek išlieka geresnis vieno ar kito požymių tipo aptikimas.

Perspektyvai invariantiški Harris ir Hessian algoritmai atitinka ankstesnės grupės pastabas. Pagrindiniai regionai reikalauja histogramų apskaičiavimo ir jų entropijų kiekvieno iš jų mastelio ar panašumų erdvėje, šios informacijos paskaičiavimas turi labai dideles sąnaudas [8]. Vienas iš tokių paveikslėlio regionų privalumų yra tai, kad jie gali būti reitinguojami pagal jų sudėtingumą ar aprašančios informacijos turinį. Tai pasitelkdami kai kurie taikymai išnaudoja tik mažą dalį pagrindinių regionų, taip išlaikydami didesnę invariantiškumą perspektyvai, paveikslėlių atpažinime [6]. Pradžioje, tokie regionai buvo nekintamo mastelio, bet vėliau praplėsti iki nekintamų perspektyvoje. Kraštinėmis pagrįsti regionai koncentruojasi į kampus, kuriuos formuoja kraštinių sandūros, tai suteikia gerą požymių lokalizavimo tikslumą ir algoritmo pasikartojimą, bet sumažina rastų požymių kiekį.

Regionų aptikimas pagrįstas idėja, kai paveikslėlis suskaldomas pagal vienodų regionų ribas. Intensyvumu besiskiriantys regionai naudoja euristinius metodus ir randa panašius regionus kaip MSER algoritmas. Superpikseliai (*angl. superpixels*) dažniausiai pagrįsti dalinimo į segmentus metodais, tokiais kaip normalizuotos paveikslėlio zonos, kas reikalauja didelių skaičiavimo sąnaudų. Superpikselių nekintamumas dažniausiai priklauso nuo naudojamo algoritmo skirto dalinimui į segmentus. Kaip priešprieša superpikseliams yra MSER metodas, kuris parenka tik labiausiai stabilius regionus (4 pav.), tai leidžia pasiekti didelį algoritmo pakartojimą. MSER metodas efektyvus dar ir dėl to, kad naudoja takoskyros (*angl. watershed*) algoritimą, paveikslėlio dalinimui į segmentus. Uždengimų kitimui atsparūs algoritmai labai naudingi tais atvejais, kai tikimasi didelių geometrinių deformacijų. Kitais atvejais jų, mastelio kitimui atsparios, versijos veikia geriau, ypač kategorijų atpažinime [7]. Tai išplaukia iš to, kad su matymo kampo kitimu iki 30 laipsnių, algoritmui susidoroti padeda jo tvirtumo savybė, vietoje nekintamumo savybės.

Uždengimo deformacijos yra kur kas dažnesnės, kai tie patys objektai stebimi iš skirtingų stebėjimo taškų.



**4 pav.** Paveikslėlio dalinimo į regionus pavyzdys panaudojant MSER algoritimą



## 4. LOKALIŲ POŽYMIŲ ALGORITMO SURF VEIKIMO ANALIZĖ

Veidų atpažinimo realiu laiku srityje, Viola ir Jones pasiūlė naudoti integralinius paveikslėlius (*angl. integral images*) [21], kurie leidžia labai sparčiai atlikti skaičiavimus su Haar bangelių (*angl. Haar wavelets*) ar bet koku kitu dėžės tipo (*angl. box-type*) sąsūkos filtru.

Toliau aprašysime kaip veikia integraliniai paveikslėliai, kaip jie naudojami SURF algoritme, norint išgauti Hessian matricas, bei kaip veikia pats SURF algoritmas [1].

### 4.1. SURF algoritmo veikimas

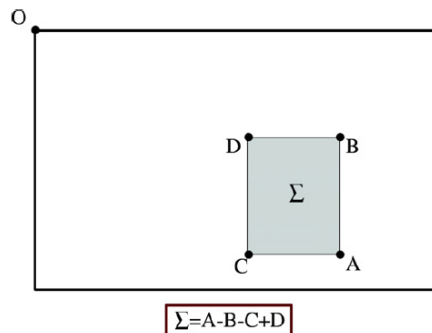
#### 4.1.1. Dominančių vietų aptikimas

##### 4.1.1.1. Integraliniai paveikslėliai

Integraliniai paveikslėliai leidžia greitus dėžės tipo sąsūkos filtrų paskaičiavimus. Integralinio paveikslėlio įvestis  $I_{\Sigma}(x)$  taške  $(x) = (x, y)^T$  yra lygi sumai visų pradinio paveikslėlio  $I$  taškų, esančių kvadratinėje zonoje suformuotoje aplink skaičiuojamą tašką.

$$I_{\Sigma}(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(j, j) \quad (1)$$

Kai integralinis paveikslėlis paskaičiuotas, reikia dar trijų papildomų veiksmų, siekiant apskaičiuoti zonų intensyvumo sumas (5 pav.). Tokios kvadratinės zonos intensyvumo paskaičiavimo laikas nepriklauso nuo jos dydžio.



5 pav. Trys reikalingi veiksmi siekiant apskaičiuoti kvadratinės zonos intensyvumą

##### 4.1.1.2. Hessian matrica pagrįsti dominantys taškai

SURF algoritmo lokalių požymių aptikimas grindžiamas Hessian matricomis, dėl jų didesnio tikslumo nei kitų aptikimo metodų. Aptikimui parenkamos į dėmes panašios struktūros,

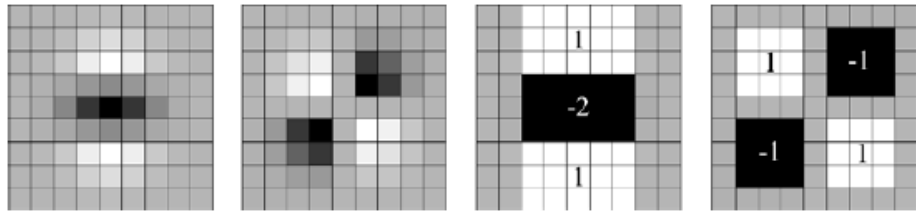
tose vietose, kuriose determinantas didžiausias. Palyginus su Hessian-Laplace aptikimo metodu, pasiūlytu Mikolajczyk ir Schmid [25], Hessian taip pat leidžia determinantą naudoti ir mastelio parinkimui, kaip rašo Lindeberg [24].

Turint tašką paveikslėlyje  $I$ , Hessian matrica -  $H(x, \sigma)$ , taške  $x$ , su masteliu  $\sigma$  yra aprašoma lygtimi:

$$H(x, \sigma) = \begin{pmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{pmatrix} \quad (2)$$

Kurioje  $L_{xx}(x, \sigma)$  yra Gauso antros eilės išvestinės sąsūka  $I$  paveikslėlio taške  $x$ , taip pat paskaičiuojami ir  $L_{xy}(x, \sigma)$  bei  $L_{yy}(x, \sigma)$ .

Apytikslės Gauso antros eilės išvestinės, gali būti nesunkiai paskaičiuotos, panaudojant integralinius paveikslėlius, kurie nereikalauja didelių skaičiavimų (6 pav.). Laikas per kurį atliekami skaičiavimai nepriklauso nuo filtro dydžio.



**6 pav.** Kairėje pusėje: originali (diskretizuota ir apkarpyta) Gauso antros eilės dalinė išvestinė  $y - (L_{yy})$  ir  $xy - (L_{xy})$  kryptimis; dešinėje pusėje: SURF algoritme naudojama Gauso antros eilės dalinė išvestinė  $y - (D_{yy})$  ir  $xy - (D_{xy})$ . Pilki regionai yra lygūs nuliui.

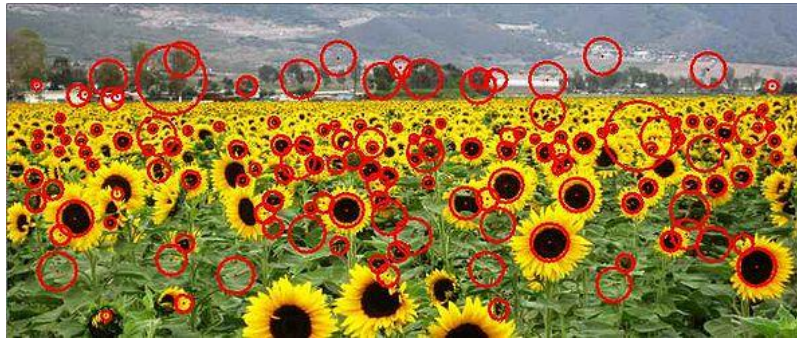
Regionams priskiriami svoriai yra nedideli, siekiant pagerinti skaičiavimo greitį. Apskaičiuojami Hessian determinantai:

$$\det(H_{approx}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (3)$$

Šie determinantai sudaro paveikslėlio dėmių atsako žemėlapi įvairiuose masteliuose. Vėliau šis žemėlapis panaudojamas skaičiuojant paveikslėlio lokalius maksimumus ir jų vietas.

### 4.1.1.3. Dominančių vietų nustatymas

Siekiant nustatyti dominančias paveikslėlio vietas įvairiuose jo masteliuose, panaudojamas  $3 \times 3 \times 3$  artimiausių kaimynų ne maksimumo slopinimas (*angl.* non-maximum suppression), pasiūlytas Neubeck ir Van Gool [26]. Tada Hessian matricos determinantų maksimumai interpoliuojami mastelyje su metodu pasiūlytu Brown ir Lowe [23]. Rastų dominančių vietų pavyzdys pateikiamas 7 pav.

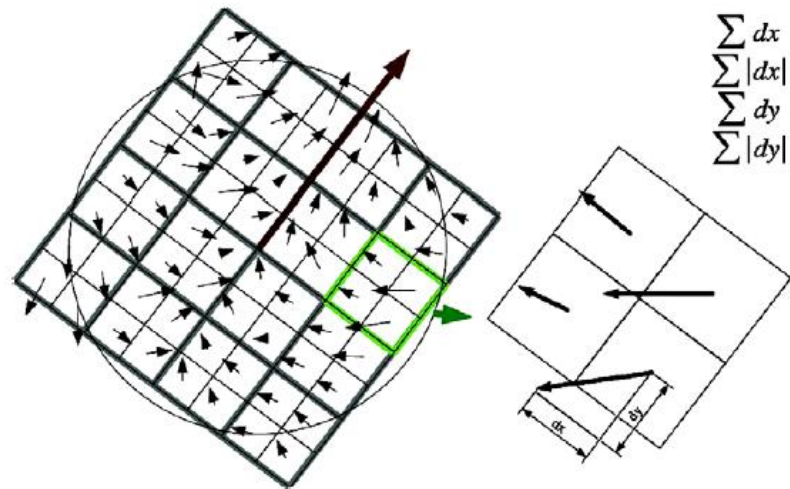


7 pav. Dominančios vietos aptiktos saulėgražų lauke.

## 4.1.2. Dominančių vietų aprašymai ir palyginimas

### 4.1.2.1. Požymių aprašas pagrįstas Haar bangelių sumomis

Dominančios vietos aprašymui išgauti, pirmiausia tos vietos centre sudaromas kvadratas atitinkantis jos dydį. Kvadratinis regionas vėliau išskaidomas į smulkesnius  $4 \times 4$  kvadratinis vidinius regionus (8 pav.). Taip išsaugoma svarbi erdvinė informacija. Tada kiekvienam mažesniai regionui paskaičiuojami Haar bangų atsakai, regionus padalinant į  $5 \times 5$  taškų kvadratus. Vertikalūs ir horizontalūs bangų atsakai susumuojami ir aprašymų vektoriuje sudaroma pirmoji sumų eilė. Kadangi dominanti vieta dalinama į  $4 \times 4$  mažesnius regionus, o mažesnius regionus aprašantis vektorius yra 4D tipo, tai gautas bendras aprašymo vektoriaus ilgis yra 64 skaičiai.



8 pav. Kvadratinis regionas išskaidytas į smulkesnius 4x4 kvadratinius vidinius regionus.

#### 4.1.2.2. Greitas indeksavimas skirtas palyginimui

Siekiant paspartinti paveikslėlių palyginimą panaudojamas indeksavimas pagal Laplacian ženklą. Paprastai dominančios paveikslėlio vietos randamos dėmių tipo struktūrose. Laplacian ženklas atskiria šviesias dėmes ant tamsaus paviršiaus nuo atvirkštinio varianto (9 pav.). Šis indeksavimas galimas be jokio papildomo skaičiavimo, nes ženklas paskaičiuojamas jau dominančios vietos aptikimo metu. Taip pat verta paminėti, jog tai padeda palyginimuose su labiau pažengusiomis indeksavimo technologijomis, kaip pavyzdžiui k-d tipo medžiuose.



9 pav. Šviesios ir tamsios dėmių pavyzdžiai.

## 5. SPALVŲ IR JŲ ERDVIŲ ANALIZĖ

### 5.1. Spalvų suvokimas ir aprašymas

Spalva yra smegenų reakcija į specifinę vaizdinę stimuliaciją. Nors ir galima tiksliai apibūdinti spalvą, išmatavus jos spektro galios pasiskirstymą (matomos elektromagnetinės spinduliuotės intensyvumas esantis atskiruose bandų ilgiuose), tačiau tai veda prie didelio pasikartojimo. Šio pasikartojimo priežastis yra ta, kad akių tinklainė atskiria spalvas naudodama tik tris plačiausias juostas, apytikriai atitinkančias raudoną, žalią ir mėlyną šviesas. Signalai iš spalvai jautrių ląstelių kartu su signalais, gautais iš intensyvumui jautrių ląstelių, sujungiami smegenys, taip sukurdami keletą skirtingų spalvų pojūčių [27]. Šiuos pojūčius galima apibrėžti kaip:

- *Ryškusumas (angl. brightness)*: žmogaus pojūtis, kuris nusako ar vieta skleidžia daugiau ar mažiau šviesos.
- *Atspalvis (angl. hue)*: žmogaus pojūtis, kurio pagalba vieta atrodo kaip viena ar dviejų spalvų mišinys iš šių spalvų: raudonos, geltonos, žalios ir mėlynos.
- *Spalvingumas (angl. colourfulness)*: žmogaus pojūtis, kuris parodo ar vieta atskleidžia daugiau ar mažiau savo atspalvio.
- *Šviesumas (angl. lightness)*: pojūtis, kuris parodo vietos *ryškumą* priklausantį nuo toje kompozicijoje esančios baltos spalvos *ryškumo*.
- *Chromatiškumas (angl. chroma)*: pojūtis, kuris parodo vietos spalvingumą priklausantį nuo toje kompozicijoje esančios baltos spalvos *ryškumo*.
- *Prisotinimas (angl. saturation)*: pojūtis, kuris parodo vietos spalvingumą priklausantį nuo jos ryškumo.

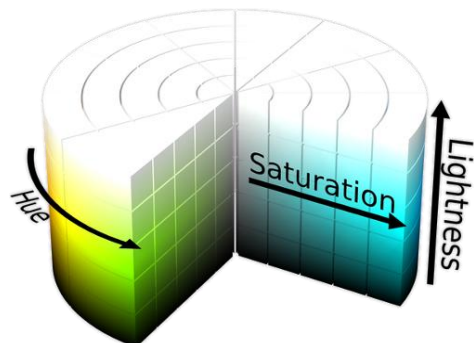
Trijų spalvų dedamųjų teorija (*angl. trichromatic theory*) aprašo kaip trys atskiros spalvos: raudona, žalia ir mėlyna, gali sudaryti bet kokią matomą spalvą. Ši teorija pagrįsta akies jautrumu trimis pagrindinėms spalvoms. Tai pagrindas, kuriuo remiasi fotografija ir daugelis kompiuterinių spalvų erdvių, tam panaudojant tris parametrus spalvai aprašyti.

### 5.2. HSL spalvų erdvė

Ši spalvų erdvė priklauso grupei panašių erdvių į kurias įeina: HSI (atspalvis, prisotinimas, intensyvumas), HSV (atspalvis, prisotinimas, reikšmė), HCI (atspalvis, prisotinimas,

spalvingumas), TSD (atspalvio sodrumas ir tamsumas) ir kitos. Dauguma šių spalvų erdvių yra RGB erdvės tiesinės transformacijos ir priklausomos nuo įrenginių. Jų pranašumas slypi nepaprastai intuityviame spalvų aprašyme. Labai nesunku pasirinkti norimą atspalvį ir vėliau pakoreguoti jo sodrumą ir intensyvumą (10 pav.).

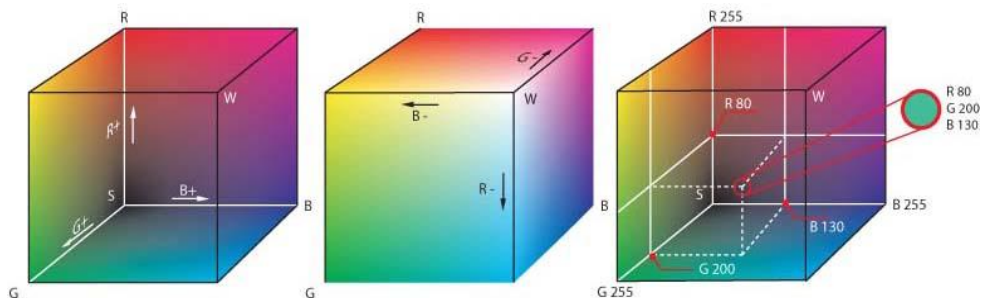
Tiriamas skaistumo ir spalvos informacijos atskyrimas, kaip pabrėžiama, turi pranašumų kai kuriuose paveikslėlių apdorojimo taikymuose. Tačiau tikslus RGB erdvės pervertimas į spalvos, sodrumo ir šviesumo informaciją pilnai priklauso nuo naudojamos įrangos parametrų.



10 pav. HSL spalvų erdvės grafinis atvaizdavimas.

### 5.3. RGB spalvų erdvė

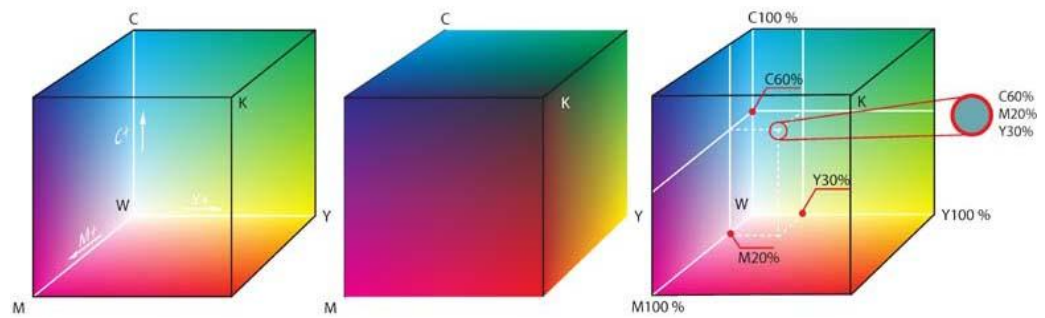
Tai suminė spalvų sistema pagrįsta trijų spalvų dedamųjų teorija (11 pav.). Dažnai naudojama sistemoje, kurios naudoja CRT monitorius paveikslėlių atvaizdavimui. RGB spalvų erdvę nesunku realizuoti, tačiau ji nėra tiesiškai priklausoma nuo vaizdų suvokimo. Ji yra priklausoma nuo įrangos, jos spalvų aprašymas pusiau intuityvus. RGB spalvų erdvė labai paplitusi ir naudojama beveik kiekvienoje kompiuterių sistemoje, televizijoje ir pan.



11 pav. Grafinis RGB spalvų erdvės atvaizdavimas.

## 5.4. CMY(K) (Žydra Violetinė Geltona (Juoda)) spalvų erdvė

Tai atimtimi pagrįsta spalvų erdvė (12 pav.). Ji dažniausiai naudojama spausdinime ir kopijavime. Pridedamas ketvirtas elementas, juoda spalva, padeda pagerinti tiek tankumą, tiek ir spalvų gamą. Teisingas pervertimas iš RGB spalvų erdvės į CMY(K) erdvę nėra lengvas. CMY(K) yra priklausoma nuo įrenginių, nėra tiesiškai priklausoma nuo vaizdų suvokimo ir labai neintuityvi.



12 pav. CMY(K) spalvų erdvės grafinis atvaizdavimas.

## 6. SURF IR SPALVŲ DEKOMPOZICIJOS APJUNGIMO APRAŠYMAS

Spalvų dekompozicija, tai spalvoto paveikslėlio padalinimas į atskirus nespalvotus paveikslėlius pagal jo spalvų kanalus. Spalvų kanalų gali būti įvairus kiekis, priklausomai nuo pasirinktos spalvų erdvės, į kuria perverčiamas paveikslėlis prieš kanalų atskyrimą. Kadangi populiariausia ir dažniausiai monitoriuose bei vaizdo apdorojime naudojama yra RGB spalvų erdvė (kartais BGR, nes skaičiavimo patogumui kanalai gali būti sukeisti vietomis), todėl ir šiame darbe pasirinktas paveikslėlio skaidymas į RGB kanalus.

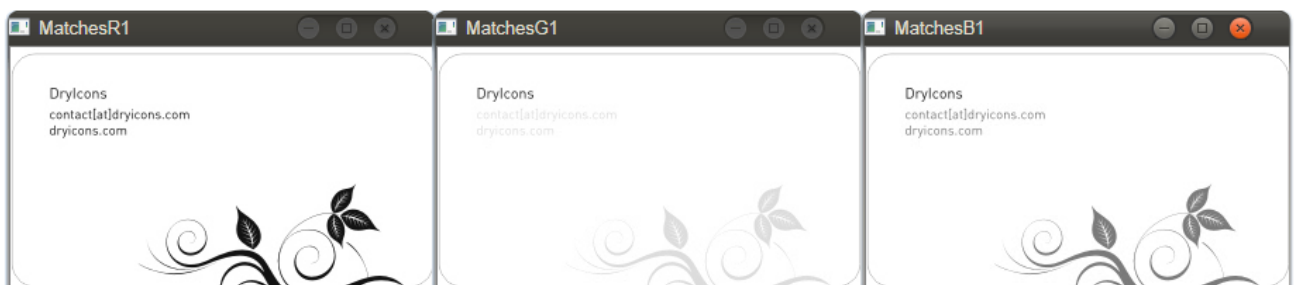
Papildytas SURF algoritmas su paveikslėlio išskaidymu į RGB spalvų kanalus (toliau vadinamas SURF RGB) turi penkis etapus:

1. Paveikslėlio nuskaitymas iš failo ar kameros (13 pav.).



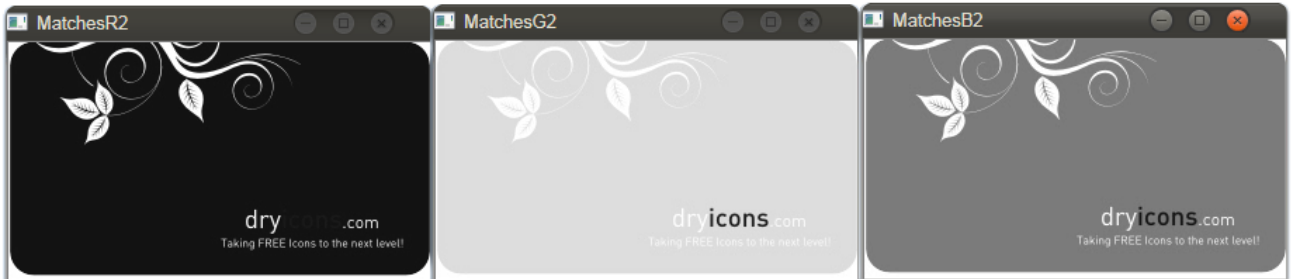
13 pav. Dviejų spalvotų kortelių paveikslėliai, kurie aiškiai skiriasi vienas nuo kito.

2. Iš paveikslėlio išskiriami trys spalvų kanalai, taip gaunami trys atskiri paveikslėliai, kuriuose atvaizduojama kiekvieno spalvos kanalo informacija. Išgauti iš spalvų kanalų paveikslėliai yra nespalvoti. Pirmame etape pateiktų pavyzdinių nuskaitytų paveikslėlių atitinkami skaidymo į kanalus pavyzdžiai pateikiami 14 pav. ir 15 pav..



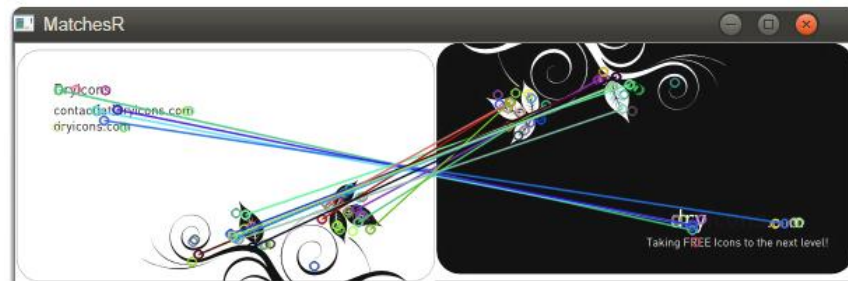
14 pav. Pirmas spalvotas kortelės paveikslėlis atskirtas į spalvų kanalus RGB (R – raudonas, G – žalias, B – mėlynas) išsaugant tik kanalo ryškumą.



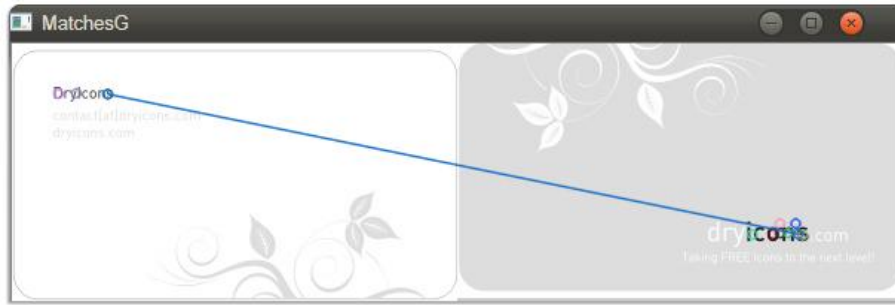


**15 pav.** Antras spalvotas kortelės paveikslėlis atskirtas į spalvų kanalus RGB (R – raudonas, G – žalias, B – mėlynas) išsaugant tik kanalo ryškumą.

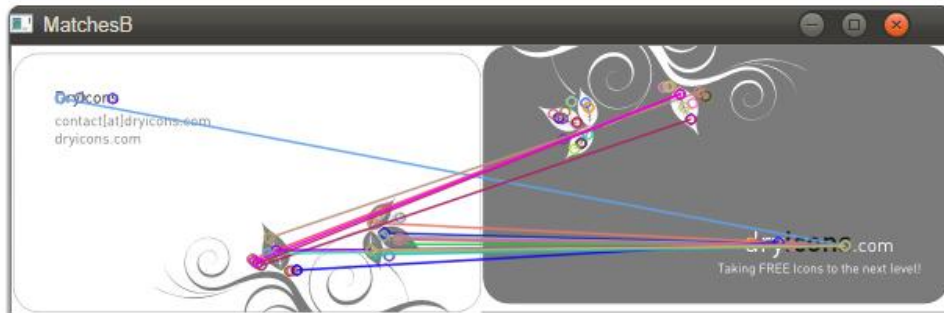
3. Kiekvieno paveikslėlio spalvų kanale aptinkami tam kanalui būdingi požymiai. Aptiktų požymių pavyzdžiai pateikiami 16 pav., 17 pav. ir 18 pav..
4. Kiekvienas požymis aprašomas išskirtiniu vektoriumi.
5. Dviejų paveikslėlių atitinkamų spalvų kanalų požymiai palyginami panaudojant iš jų išskirtų požymių aprašymo vektorius. Jei sutampa didesnis paveikslėlio kiekvienos spalvos kanalo požymių kiekis, išreikštas procentais, negu tikslumas nurodytas algoritmo parametruose (atpažinimo tikslumo ribą galima keisti, pagal konkrečiai sistemai keliamus reikalavimus), fiksuojama jog paveikslėliuose atvaizduota vienoda informacija. Sutampantys paveikslėlių kanalų požymių pavyzdžiai ir sutapimo procentinės vertės pateikiami 11-ame, 12-ame ir 13-ame paveikslėliuose.



**16 pav.** SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, raudonos spalvos kanale (sutapimas 65,52 %).



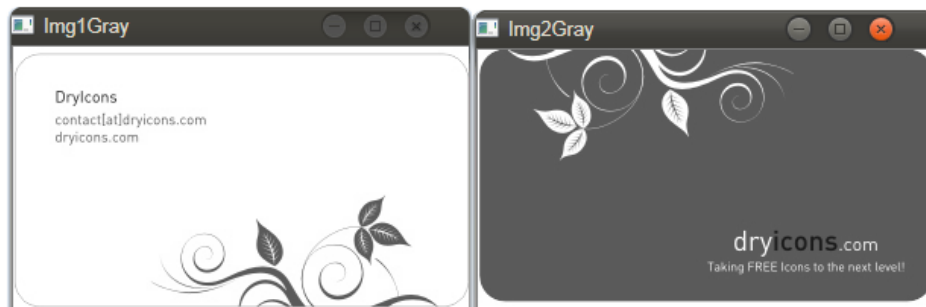
**17 pav.** SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, žalios spalvos kanale (sutapimas 32,65 %).



**18 pav.** SURF algoritmo aptikti ir sutampantys, pirmos ir antros kortelių paveikslėlių, lokalūs požymiai, mėlynos spalvos kanale (sutapimas 33,33 %).

Palygimui pateikiamas originalaus SURF algoritmo pavyzdys, turintis keturis etapus:

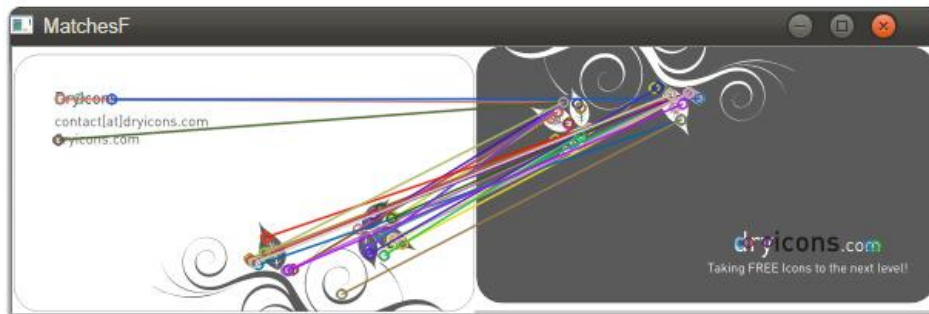
1. Paveikslėlių nuskaitymas iš failo ar kameros ir pervertimas į nespaltvotus. Dviejų nuskaitytų ir paverstų į nespaltvotus paveikslėlių pavyzdys pateikiamas 19 pav..



**19 pav.** Spalvoti kortelių paveikslėliai paversti nespaltvotais prieš apdorojant su SURF algoritmu.

2. Kiekviename nespaltvotame paveikslėlyje aptinkami jam būdingi lokalūs požymiai. Aptiktų požymių pavyzdžiai pateikiami 15-ame paveikslėlyje.

3. Kiekvienas požymis aprašomas išskirtiniu vektoriumi, panaudojant Haar bangų sumas.
4. Dviejų paveikslėlių požymiai palyginami panaudojant iš jų išskirtų požymių aprašymo vektorius. Jei sutampa didesnis paveikslėlio požymių kiekis, išreikštas procentais, negu tikslumas nurodytas algoritmo parametruose (atpažinimo tikslumo ribą galima keisti, pagal konkrečiai sistemai keliamus reikalavimus), fiksuojama jog paveikslėliuose atvaizduota vienoda informacija. Sutampantys kanalų paveikslėlių požymių pavyzdys ir sutapimo procentas pateikiamas 20 pav.



**20 pav.** SURF algoritmo aptikti ir sutampantys pirmos ir antros kortelių paveikslėlių lokalūs požymiai (sutapimas 96,97 %).

Kaip matome iš pavyzdžio, pateikiamo prie modifikuoto SURF algoritmo, dviejų skirtingų kortelių paveikslėlius, originalus SURF algoritmas fiksuoja kaip sutampančius, nes sutapimas remiantis jo skaičiavimais yra 96,97% ir tai viršija 95% nustatytą sutapimo ribą. Tuo tarpu, paveikslėlius apdorojant SURF RGB algoritmu, kai panaudojamas išskaidymas į spalvų kanalus, sutapimai lygūs: 65,52% - raudoname kanale, 32,65% - žaliame kanale ir 33,33% - mėlyname kanale. Kas rodo bendrą sutapimą tik 43,83 %. Kiekvieno RGB spalvų erdvės atitinkamo kanalo, tarp dviejų paveikslėlių, sutapimas turi siekti virš 95% nustatytos ribos, kad paveikslėliai sutaptų. Todėl galima fiksuoti paveikslėlių nesutapimą, jei bent vienas iš RGB spalvų erdvės kanalų neviršija nustatytos 95% atpažinimo tikslumo ribos. Algoritmo aprašymui ir pavyzdiniam palyginimui buvo pasirinktas beveik idealus paveikslėlis (formuotas kompiuteriu). Vaizdo kameromis ar fotoaparatais daromi paveikslėliai turi daugiau deformacijų ir triukšmų, kylančių nuo prasto apšvietimo, aparato virpesių ar kitų šaltinių, todėl palyginimo rezultatai ne visada siekia tokias aukštas ribas (95%).

## **7. OBJEKTŲ ATPAŽINIMO SISTEMOS ARCHITEKTŪROS APRAŠYMAS**

„Universali vaizdo atpažinimo sistema“, tai trijų magistrantų: Liudo Drejerio, Ernesto Kardzio ir Karolio Kriščiūno, praktinis darbo rezultatas. Sistema sukurta naudojant kliento - serverio architektūrą ir pritaikyta mobiliems įrenginiams (telefonai bei planšetiniai kompiuteriai veikiantys su Android OS 2.3).

Jos veikimo principas:

1. Vartotojas kliento pagalba nufotografuoja norimą atpažinti ar išsaugoti objektą.
2. Nufotografuoto objekto paveikslėlis yra apdorojamas SURF RGB algoritmo.
3. Išskirti iš paveikslėlio lokalūs požymiai ir jų aprašymai, JSON formatu, siunčiami į serverį (Ubuntu Linux 11.10).
4. Serveryje, gauti objekto požymių aprašymai, palyginami su MySQL 5.5 duomenų bazėje esančiais etaloninių objektų požymių aprašymais arba išsaugomi.
5. Operacijos rezultatas gražinamas vartotojui per kliento sąsają.

### **7.1. Serverio dalies aprašymas**

Serveryje naudojamos technologijos:

1. MySQL – duomenų bazių serveris.
2. Ubuntu Linux 11.10 – operacinė sistema.
3. OpenCV 2.1 – vaizdų apdorojimo biblioteka.
4. MySQL/C++ Connector – prisijungimo prie MySQL duomenų bazės, per C++ programavimo kalbą, biblioteka.
5. Boost – duomenų persiuntimo paspartinimo biblioteka.

Pagrindinės serverio atliekamos operacijos:

1. Prisijungiančių klientų valdymas.
2. Objektų atpažinimas (paieška duomenų bazėje).
3. Šalinimas.
4. Atnaujinimas.
5. Pridėjimas.

## 7.2. Kliento dalies aprašymas

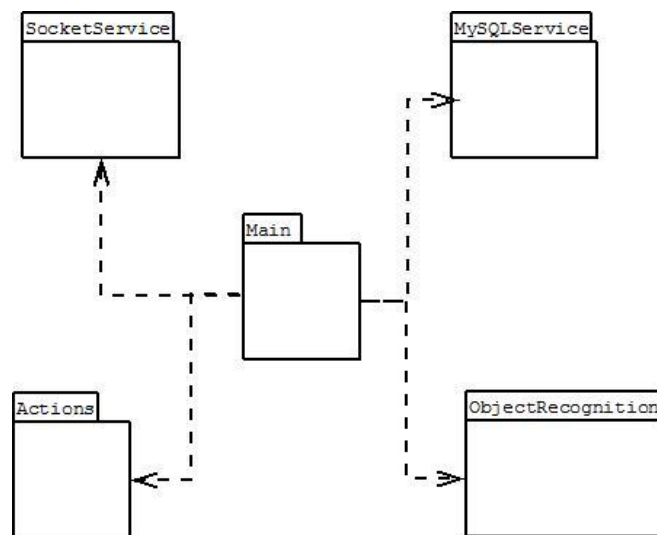
Kliente naudojamos technologijos:

1. OpenCV Java 2.3 – biblioteka skirta paveikslėlių apdorojimui Java programavimo kalba.
2. QT SDK – vartotojo aplinkos kūrimo įrankiai (Windows, Linux klientams).
3. Android SDK – Android programinės įrangos kūrimo įrankiai (Android OS klientui).

Pagrindinės kliento atliekamos operacijos yra: objekto pridėjimas, šalinimas, redagavimas bei paieška.

## 7.3. Serverio architektūra

### 7.3.1. Serverio komponentų diagramos



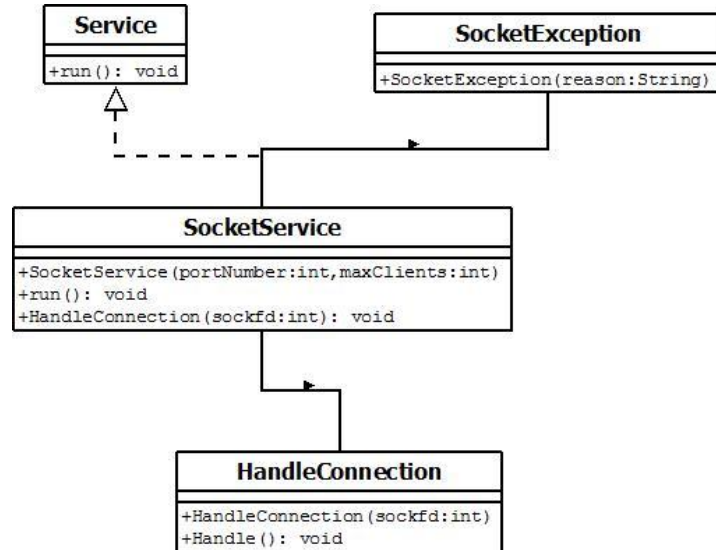
21 pav. Bendra sistemos paketų diagrama [22].

Serveris sudarytas iš kelių pagrindinių komponentų:

1. **SocketService.** Sistemos komponentas, atsakingas už darbą su tinklu.
2. **MySQLService.** Sistemos komponentas, atsakingas už darbą su duomenų baze.
3. **Actions.** Sistemos komponentas, atsakingas už kliento pasirinktų veiksmų apdorojimą.
4. **ObjectRecognition.** Sistemos komponentas, atsakingas už objektų atpažinimą.
5. **Main.** Pagrindinis sistemos komponentas. Jo paskirtis - susieti visus kitus komponentus.

### 7.3.2. SocketService komponentas

Komponento paskirtis yra apdoroti prisijungimus prie serverio, naudojant lizdus (*angl. socket*) (22 pav.).



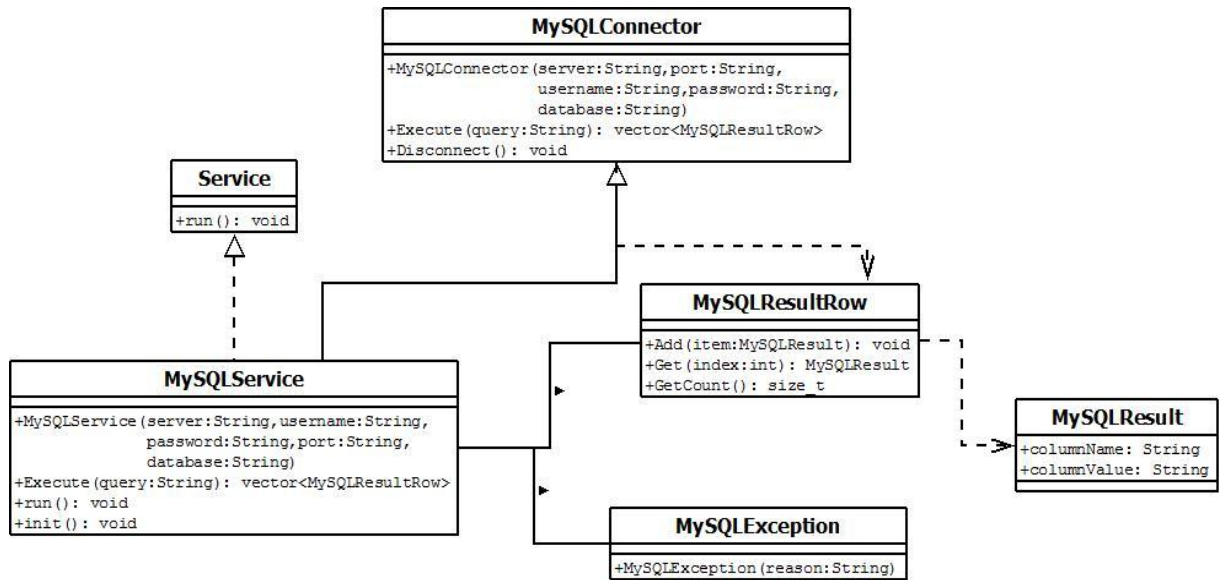
22 pav. SocketService komponento klasių diagrama [22].

1 lentelė. SocketService komponento metodų aprašymas [22]

Klasė	Metodas	Aprašymas
<i>Service</i>	run()	Atlikti serviso veiksmus
<i>SocketException</i>	SocketException(reason:String)	Soketų komponento išimtis
<i>SocketService</i>	SocketService(portNumber:int, maxClients:int)	Konstruktorius. Parametrai nurodo, kokią standartinę jungtį naudoti bei kiek gali būti daugiausiai klientų
	run()	Paleidžia soketų komponentą, laukia klientų prisijungimų
	HandleConnection(sockfd:int)	Prisijungus klientui iškviečiamas metodas, kuris sukuria kliento objektą
<i>HandleConnection</i>	HandleConnection(sockfd:int)	Kliento objektas
	Handle()	Apdoroja kliento prisijungimą

### 7.3.3. MySQLService komponentas

Komponento paskirtis yra dirbti su MySQL duomenų baze (prisijungimas prie duomenų bazės, atsijungimas, užklausų siuntimas ir gavimas, rezultatų apdorojimas ir pan.) ir suteikti API sąsają kitoms sistemos dalims (23 pav.).



23 pav. MySQLService komponento diagrama [22].

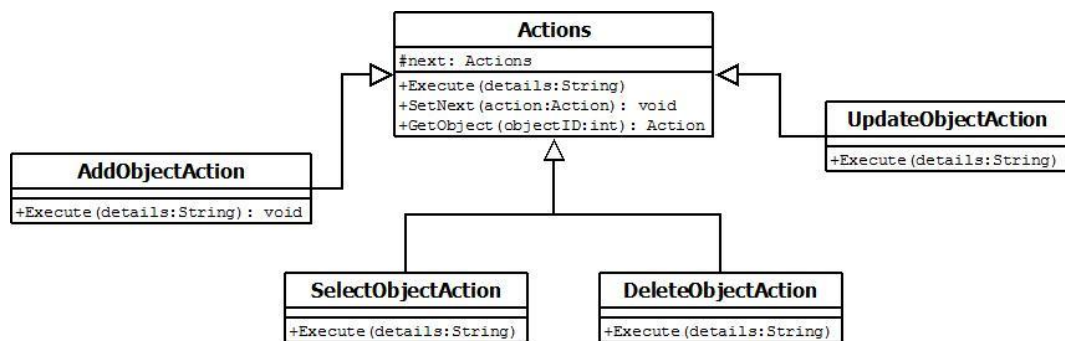
2 lentelė. MySQLService komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>Service</i>	run()	Atlikti serviso veiksmus
<i>MySQLException</i>	MySQLException(reason:String)	Duomenų bazės komponento išimtis
<i>MySQLService</i>	MySQLService(server:String, username:String, password:String, port:String, database:String)	Konstruktorius. Parametrai nurodo serverį, vartotojo prisijungimo duomenis ir duomenų bazę
	run()	Paleidžia operacijų su MySQL servisą
	Execute(query:String)	Įvykdo užklausą
	init()	Inicializuoja servisą - prisijungia prie duomenų bazių serverio, pasiruošia vykdyti užklausas
<i>MySQLConnector</i>	MySQLConnector(server:String, username:String, password:String, port:String, database:String)	Konstruktorius. Parametrai nurodo serverį, vartotojo prisijungimo duomenis ir duomenų bazę
	Execute(query:String): vector<MySQLResultRow>	Įvykdo užklausą
	Disconnect()	Atsijungia nuo duomenų bazių serverio
<i>MySQLResultRow</i>	Add(item: MySQLResult)	Prideda užklausos vykdymo rezultato eilutę į rezultatus
	Get(index: int) : MySQLResult	Grąžina užklausos vykdymo

Klasė	Metodas/kintamasis	Aprašymas
		rezultatą
	GetCount() : size_t	Grąžina užklauso vykdyto rezultatus skaičių
MySQLResult	columnName	Stulpelio reikšmė
	columnValue	Stulpelio pavadinimas

### 7.3.4. Actions komponentas

Komponento pagrindinė paskirtis yra atlikti veiksmus su objektų požymių aprašymais: pridėti, redaguoti ar pašalinti (24 pav.).



24 pav. Actions komponento paketų diagrama [22].

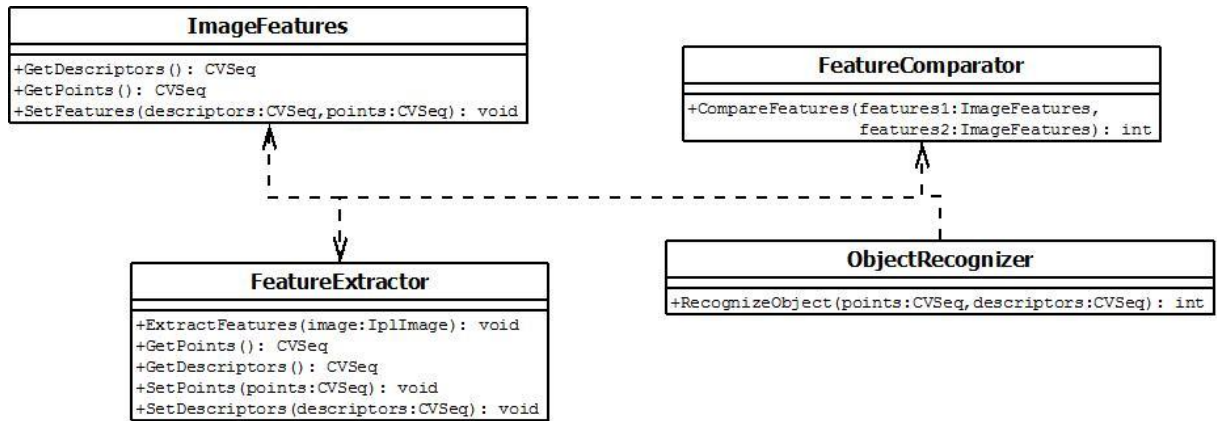
3 lentelė. Actions paketo metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
Actions	next: Actions	Elementų sąrašas
	Execute(details: String)	Įvykdo užklausą
	SetNext(action: Action)	Nustato sekantį elementą
	GetObject(objectID : int) : Action	Grąžina nurodytą sąrašo elementą
AddObjectAction	Execute(details: String)	Prideda objektą
SelectObjectAction	Execute(details: String)	Suranda objektą
DeleteObjectAction	Execute(details: String)	Ištrina objektą
UpdateObjectAction	Execute(details: String)	Atnaujina objektą



### 7.3.5. ObjectRecognition komponentas

Komponento paskirtis – išanalizuoti pateiktus serveriui paveikslėlių požymius ir jų aprašymus, vėliau juos panaudoti objektų palyginimui ir atpažinimui (25 pav.).



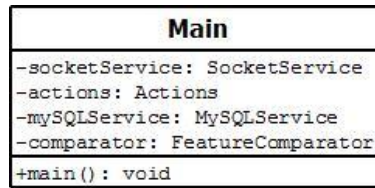
25 pav. ObjectRecognition komponento klasių diagrama [22].

4 lentelė. ObjectRecognition komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>ImageFeatures</i>	GetDescriptors() : CVSeq	Gražina deskriptorių sąrašą
	GetPoints() : CVSeq	Gražina taškų sąrašą
	SetFeatures(descriptors: CVSeq, points: CVSeq)	Nustato taškus ir taškų deskriptorius
<i>FeatureComparator</i>	CompareFeatures(features1: ImageFeatures, features2: ImageFeatures) : int	Palygina du paveikslėlius. Gražina jų panašumą
<i>FeatureExtractor</i>	ExtractFeatures(image: IplImage)	Iš paveikslėlio išskiria jį aprašančius taškus ir deskriptorius
	GetPoints() : CVSeq	Gražina taškus
	GetDescriptors() : CVSeq	Gražina deskriptorius
	SetPoints(points: CVSeq)	Nustato taškus
	SetDescriptors(descriptors: CVSeq)	Nustato deskriptorius
<i>ObjectRecognizer</i>	RecognizeObject(points: CVSeq, descriptors: CVSeq) : int	Atlieka objektų atpažinimą. Gražina objektų panašumą

### 7.3.6. Main komponentas

Šio komponento paskirtis yra sujungti visus komponentus į vieną visumą (26 pav.).



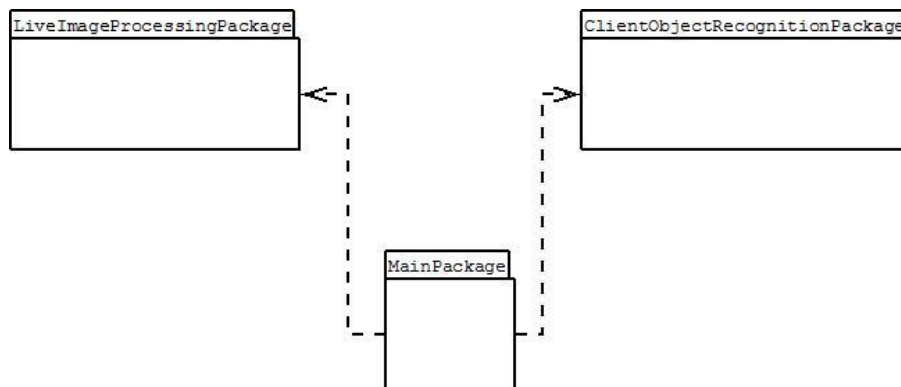
26 pav. Main komponento klasių diagrama [22].

5 lentelė. Main komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>Main</i>	socketService: SocketService	Soketų komponento objektas
	actions: Actions	Veiksmų komponento objektas
	mySQLService: MySQLService	Duomenų bazių komponento objektas
	comparator: FeatureComparator	Objektų palyginimo komponento objektas
	main()	Pagrindinis metodas, paleidžiantis visą sistemą

## 7.4. Kliento architektūra

### 7.4.1. Kliento komponentų diagrama



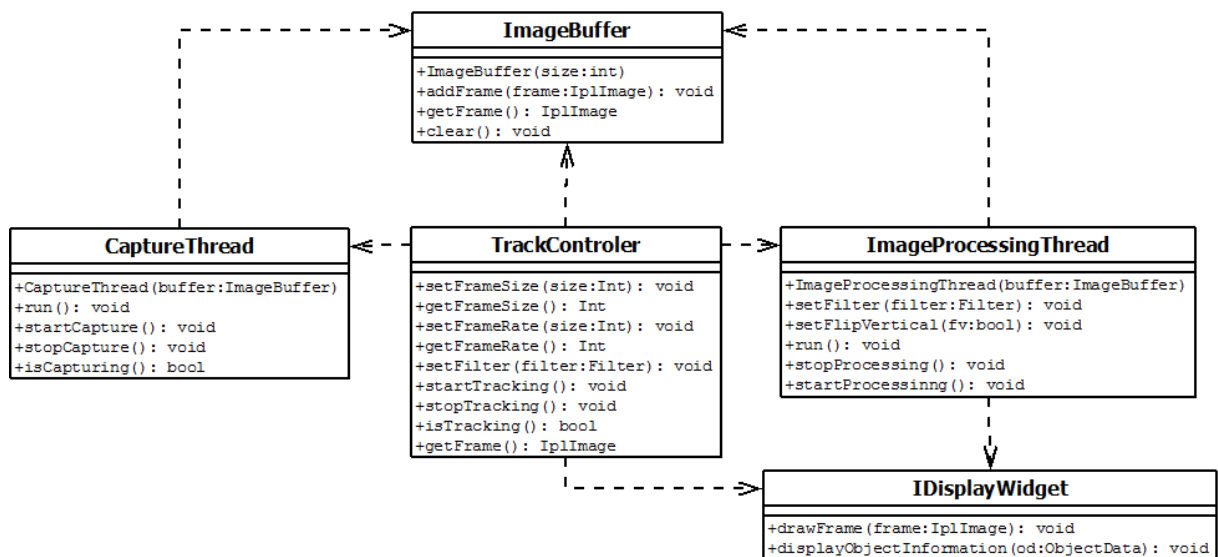
27 pav. Kliento pagrindinė komponentų diagrama [22].

Klientas sudarytas iš kelių pagrindinių komponentų:

1. **LiveImageProcessingPackage**. Sistemos komponentas, atsakingas už iš kameros paimto vaizdo atvaizdavimą ekrane.
2. **ClientObjectRecognitionPackage**. Sistemos komponentas, atsakingas už iš kameros paimtos nuotraukos apdorojimą.
3. **MainPackage**. Sistemos komponentas, atsakingas už kliento sistemos paleidimą ir kliento sistemos veikimą.

### 7.4.2. LiveImageProcessingPackage komponentas

Šiame pakete esančios klasės naudojamos paimti kameros vaizdą ir jį atvaizduoti ekrane (28 pav.).



28 pav. LiveImageProcessingPackage komponento diagrama [22].

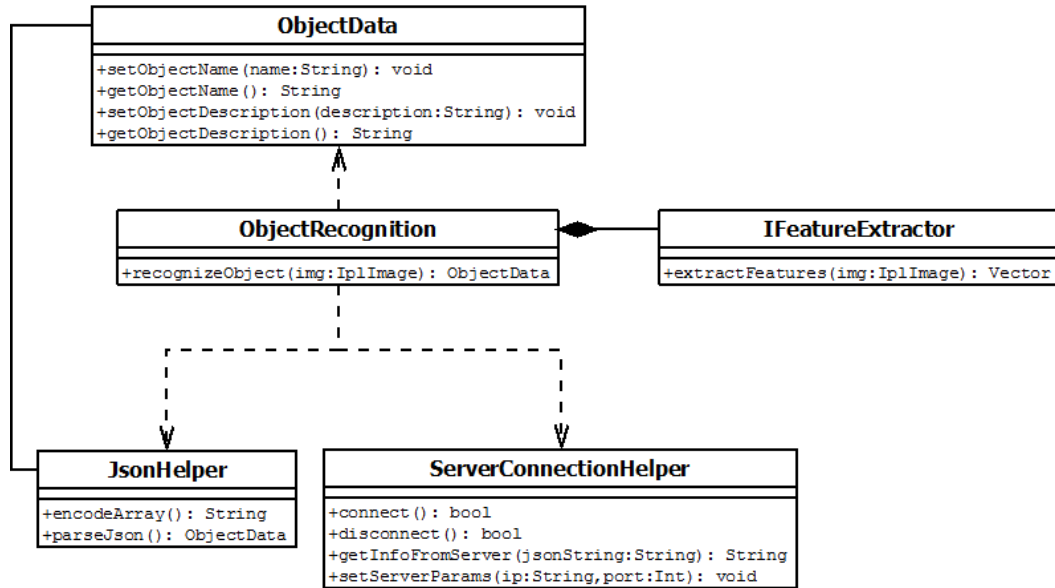
6 lentelė. LiveImageProcessingPackage komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>ImageBuffer</i>	ImageBuffer(size: int)	Klasės konstruktorius.
	addFrame(frame: IplImage)	Įdeda kadrą į buferį.
	getFrame() : IplImage	Grąžina naujausią kadrą iš buferio.
	clear()	Išvalo buferį.
<i>CaptureThread</i>	CaptureThread(buffer: ImageBuffer)	Klasės konstruktorius.
	run()	Metodas, kviečiamas gijos veikimo metu. Jei klasės būseną nustatyta į „CAPTURING“, ima iš kameros vieną

Klasė	Metodas/kintamasis	Aprašymas
		kadrą ir patalpina jį buferyje.
	startCapture()	Inicijuoja vaizdo ėmimą iš kameros.
	stopCapture()	Stabdo vaizdo ėmimą iš kameros.
	isCapturing()	Grąžina informaciją, ar gija ima kadrus iš kameros ir rašo į buferį.
<i>TrackController</i>	setFrameSize(size: int)	Nustato kadro dydį.
	getFrameSize() : int	Grąžina kadro dydį.
	setFrameRate(size: int)	Nustato kadro dažnį.
	getFrameRate() : int	Grąžina kadro dažnį.
	setFilter(filter: Filter)	Nustato filtrą
	startTracking()	Pradedama imti vaizdą iš kameros, jį atvaizduoti ekrane. Jei objektas atpažintas, aktyvuoja sekimo algoritmą.
	stopTracking()	Nustoja imti vaizdą iš kameros, jį atvaizduoti ekrane.
	isTracking() : bool	Grąžina, ar objektas yra sekamas.
	getFrame() : IplImage	Grąžina naujausią buferio kadrą.
<i>ImageProcessingThread</i>	ImageProcessingThread(buffer: ImageBuffer)	Klasės konstruktorius.
	setFilter(filter: Filter)	Nustato filtrą.
	setFlipVertical(fv: bool)	Apverčia paveikslėlį.
	run()	Sukūrus giją, ima kadrą iš buferio ir kviečia IDisplayWidget sąsajos metodu drawFrame().
	stopProcessing()	Pradedama paveikslėlio apdorojimą.
	startProcessing()	Stabdo paveikslėlio apdorojimą.
<i>IDisplayWidget</i>	drawFrame(frame: IplImage)	Atvaizduoti objektą ekrane.
	displayObjectInformation(od: ObjectData)	Pateikia informaciją apie objektą ekrane.

### 7.4.3. ClientObjectRecognitionPackage komponentas

Šis komponentas skirtas iš kameros paimto kadro apdorojimui, paveikslėlyje esančių objekto požymių išskyrimui bei kreipimuisi į serverį (29 pav.).



29 pav. ClientObjectRecognitionPackage komponento diagrama [22].

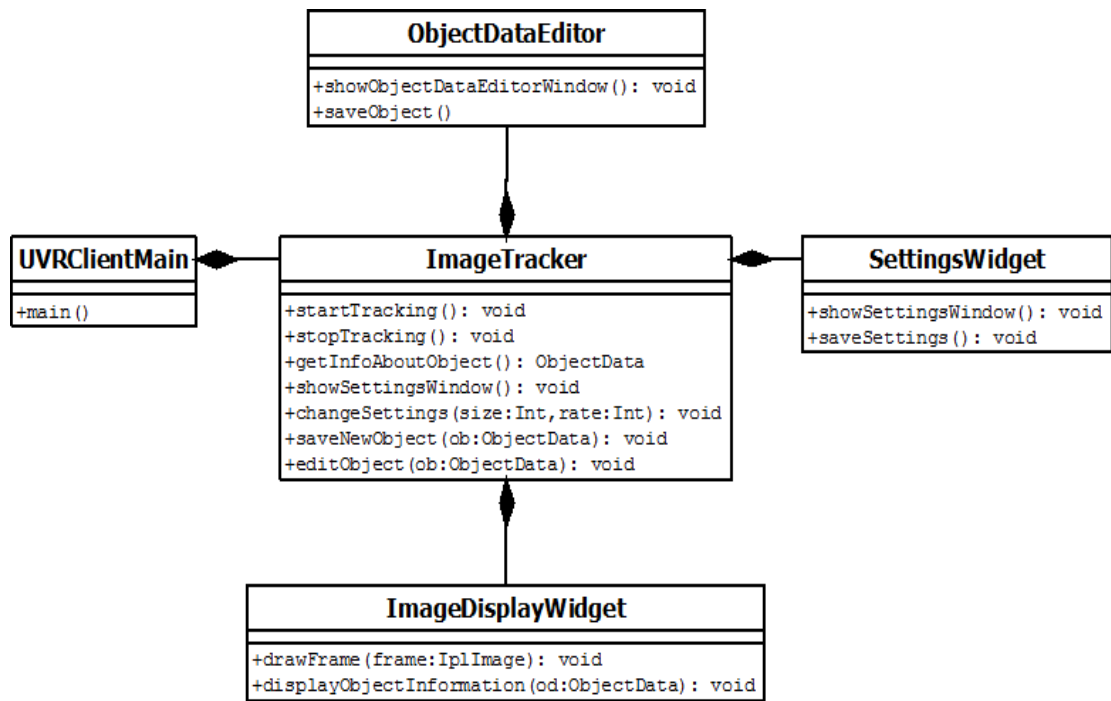
7 lentelė. ClientObjectRecognitionPackage komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>ObjectData</i>	setObjectName(name: String)	Nustato objekto vardą.
	getObjectName() : String	Gražina objekto vardą.
	setObjectDescription(description: String)	Nustato objekto aprašymą.
	getObjectDescription() : String	Gražina objekto aprašymą.
<i>ObjectRecognition</i>	recognizeObject(img: IplImage) : ObjectData	Išskiria norimo atpažinti objekto požymius, gauna reikiamą JSON bei asinchroniškai kreipiasi į serverį. Naudoja atitinkamai FeatureExtractor, JSONHelper ir ServerConnectionHelper klasių metodus.
<i>IFeatureExtractor</i>	extractFeatures(img: IplImage) : Vector	Išskiria objekto požymius.
<i>JsonHelper</i>	encodeArray() : String	Masyvą perkoduoja JSON formatu.
	parseJson() : ObjectData	JSON formato masyvą perkoduoja į masyvą.
<i>ServerConnectionHelper</i>	connect() : bool	Prisijungia prie serverio.
	disconnect() : bool	Atsijungia nuo serverio
	getInfoFromServer(jsonString : String) : String	Gauna atsakymą iš serverio.

Klasė	Metodas/kintamasis	Aprašymas
	setServerParams(ip: String, port : int)	Nustato prisijungimo prie serverio parametrus.

#### 7.4.4. MainPackage komponentas

Šiame pakete esančios klasės realizuoja kliento vartotojo sąsają bei koordinuoja kitus kliento dalies objektus (30 pav.).



30 pav. MainPackage komponento diagrama [22].

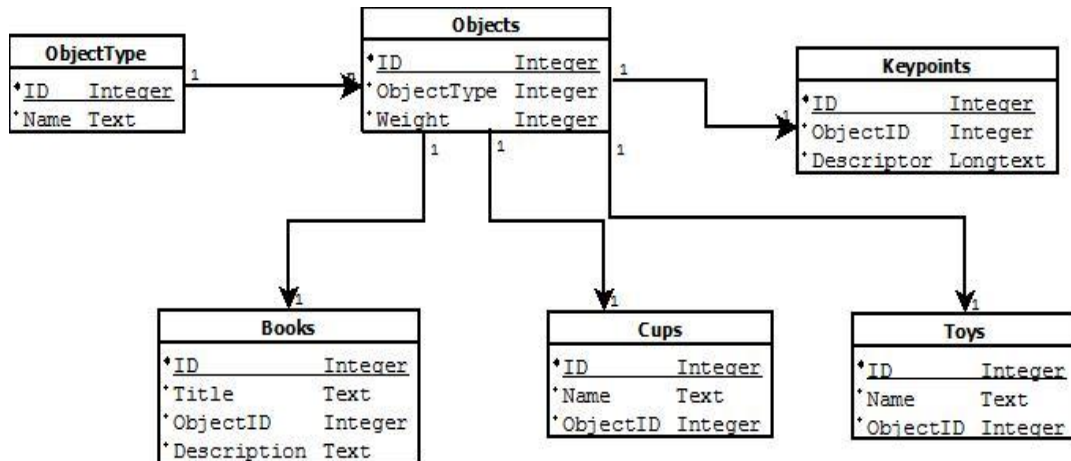
8 lentelė. MainPackage komponento metodų aprašymas [22]

Klasė	Metodas/kintamasis	Aprašymas
<i>ObjectDataEditor</i>	showObjectDataEditorWindow()	Parodo objekto informacijos redagavimo langą.
	saveObject()	Išsaugo naują informaciją.
<i>UVRClientMain</i>	main()	Paleidžia programą.
<i>SettingsWidget</i>	showSettingsWidget()	Parodo nustatymų langą.
	saveSettings()	Validuoja ir išsaugo nustatymus.
<i>ImageTracker</i>	startTrackgin()	Pradedą iš kameros imamą vaizdą vaizduoti ekrane.
	stopTracking()	Sustabdo iš kameros paimto vaizdo pateikimą ekrane (sustabdo kadra).
	getInfoAboutObject() : ObjectData	Apdoroja objekto atpažinimo mygtuko nuspaudimą.
	showSettingsWindow()	Parodo nustatymų koregavimo langą.

Klasė	Metodas/kintamasis	Aprašymas
	changeSettings(size : int, rate: int)	Pakeičia programos nustatymus.
	saveNewObject(ob: ObjectData)	Apdoroja naujo objekto išsaugojimo mygtuko nuspaudimą.
	editObject(ob: ObjectData)	Apdoroja objekto informacijos redagavimo išsaugojimo mygtuko nuspaudimą.
<i>ImageDisplayWidget</i>	drawFrame(frame: IplImage)	Piešia kadražą ekrane.
	displayObjectInformation(ob: ObjectData)	Piešia informaciją apie objektą.

## 7.5. Duomenų bazės struktūra

Sistemos duomenys yra saugomi reliacinėje duomenų bazėje, kurios apibendrinta duomenų bazės lentelių schema yra pateikta 31 pav.



31 pav. Duomenų bazės struktūra [22].

## 7.6. Sistemos įdiegimas

Sistema 2012 m. buvo pristatyta KTU jaunųjų mokslininkų darbų parodoje “Technorama” ir pelnė padėką už sėkmingą dalyvavimą. Padėkos kopija pateikiama 6–ame priede.

Sistema įdiegta įmonėse: UAB „Informacinių technologijų organizacija“ ir VĮ „Registru centras“. Sėkmingą diegimą patvirtinančios perdavimo ir aprobavimo aktų kopijos pateikiamos atitinkamai 4–ame ir 5–ame prieduose.

## 8. SURF IR SPALVŲ DEKOMPOZICIJOS EKSPERIMENTAI

### 8.1. Atsparumo pasukimui eksperimentas

Siekiant patikrinti ar patobulintas SURF (toliau SURF RGB) algoritmas veikia geriau už originalų, buvo atliekami trys eksperimentai. Pirmasis eksperimentas skirtas patikrinti kaip kinta patobulinto ir originalaus SURF algoritmo atpažinimo tikslumas, keičiantis objekto matymo kampui. Šis eksperimentas buvo atliekamas dviem bandymais: su erdviu paviršiumi (puodeliu) ir plokščiu paviršiumi (arbatos pakeliu). Abiejų bandymų etaloninių objektų paveikslėliai pateikiami atitinkamai 32 pav. ir 33 pav.



**32 pav.** Etaloninis objektas (puodelis) sutapimo pasukant bandymui.



**33 pav.** Etaloninis objektas (arbatos pakelis) sutapimo pasukant bandymui.

Pirmojo eksperimento metu etaloninių objektų pasukimo kampas buvo keičiamas nuo  $5^\circ$  iki  $90^\circ$  ir plokščias paviršius – iki  $50^\circ$ . Ties  $90^\circ$  riba dalis puodelio ornamento pasislėpė. Etaloniniai objektai buvo pasukami skirtingais kampais ir fotografuojami, vėliau gauti paveikslėliai lyginami su originaliu etaloninio objekto paveikslėliu, panaudojant SURF ir SURF RGB algoritmus. Nufotografuoti, įvairiais kampais pasukto, etaloninio objekto paveikslėliai pateikiami 34 pav. ir 35 pav.





34 pav. Pasukto nuo 5° iki 90° etaloninio objekto (puodelio) paveikslukai.



35 pav. Pasukto nuo 5° iki 50° etaloninio objekto (arbatos pakelio) paveikslukai.

Objektų palyginimo duomenys pateikiami 9-oje ir 10-oje lentelėse. 9-oje lentelėje pateikiamas erdvinio objekto (puodelio) sutapimas su etaloniniu objektu. 10-oje lentelėje pateikiamas plokščio objekto (arbatos pakelio) sutapimas su etaloniniu objektu. Pirmame lentelių stulpelyje pateikiamos SURF algoritmo sutapimo procentinės reikšmės, ir šalia pateikiami algoritmo veikimo laikai. Trečiame stulpelyje pateikiamos SURF RGB algoritmo sutapimo procentinės reikšmės, ir šalia algoritmo veikimo laikai. Likę trys lentelių stulpeliai parodo procentines sutapimo reikšmes atitinkamai raudoname, žaliame ir mėlyname kanaluose.

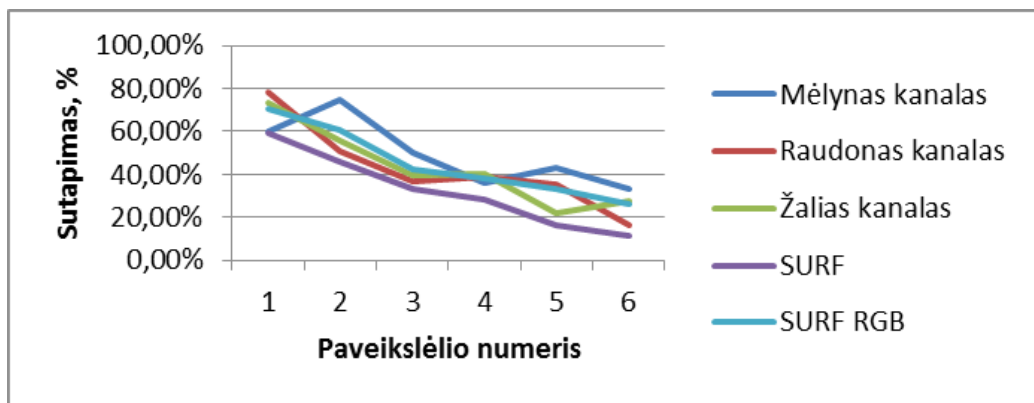
9 lentelė. Sutapimas su etalonu pasukant objektą (puodelis) ir palyginimo laikas

Pav. Nr.	SURF	Laikas	SURF RGB	Laikas	Raudonas kanalas	Žalias kanalas	Mėlynas kanalas
1	59,46%	0,30	70,75%	0,87	78,57%	73,68%	60,00%
2	45,95%	0,25	60,63%	0,82	51,11%	55,79%	75,00%
3	33,33%	0,27	42,12%	0,89	36,59%	39,77%	50,00%
4	28,57%	0,27	38,46%	0,84	39,02%	40,00%	36,36%
5	16,22%	0,26	33,48%	0,88	35,48%	22,09%	42,86%
6	11,76%	0,26	25,89%	0,82	16,67%	27,67%	33,33%

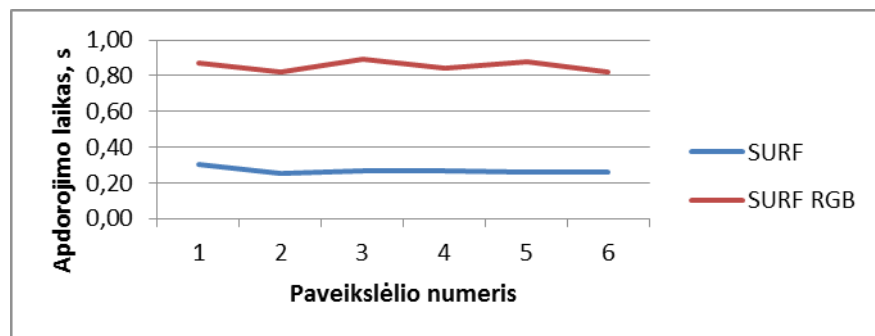
10 lentelė. Sutapimas su etalonu pasukant objektą (arbatos pakelis) ir palyginimo laikas

Pav. Nr.	SURF	Laikas	SURF RGB	Laikas	Raudonas kanalas	Žalias kanalas	Mėlynas kanalas
1	78,08%	0,74	79,94%	2,18	83,02%	76,73%	80,05%
2	83,94%	0,67	80,32%	2,02	84,81%	80,66%	75,49%
3	62,18%	0,65	70,88%	2,00	72,45%	63,07%	77,13%
4	44,72%	0,63	55,29%	1,95	54,48%	42,15%	69,24%
5	21,12%	0,55	29,70%	1,73	36,28%	19,50%	33,33%

Lentelėse pateikti duomenys taip pat atvaizduojami grafiškai. 36 pav. ir 37 pav. pavaizduoti grafikai parodo pasukto objekto procentinę sutapimo reikšmę su etaloniniu objektu, paveikslėlius apdorojant SURF ir SURF RGB algoritmais. Grafikų x ašyje pažymėti pasuktų objektų paveikslėlių numeriai, y ašyje – sutapimo su etaloniniu paveikslėliu procentinė reikšmė.



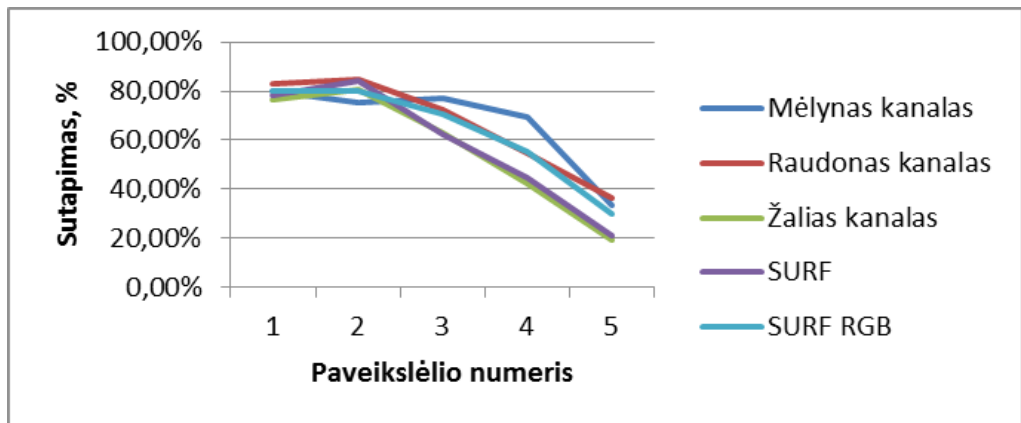
36 pav. Sutapimas su etalonu pasukant objektą (puodelis).



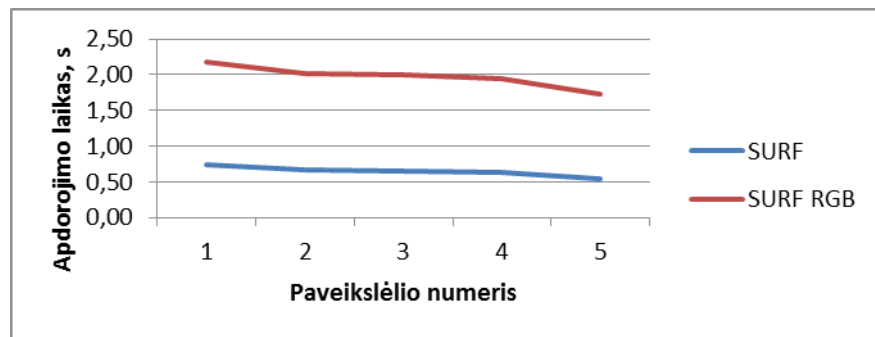
37 pav. Palyginimo laikas pasukant objektą (puodelis).

38 pav. ir 39 pav. pateikiami grafikai parodo SURF ir SURF RGB paveikslėlių palyginimo laikus. Grafikų x ašyje atvaizduojamas apdorojamo paveikslėlio numeris, y ašyje – apdorojimo trukmė sekundėmis.

Iš eksperimento rezultatų matome, jog trijuose kanaluose požymių aptinkama daugiau, nei viename, originalaus SURF algoritmo apdorojamame, paveikslėlyje, tai SURF RGB algoritmui leidžia tiksliau palyginti pasuktus objektus su etaloniniais.



38 pav. Sutapimas su etalonu pasukant objektą (arbatos pakelis).



39 pav. Palyginimo laikas pasukant objektą (arbatos pakelis).

## 8.2. Atsparumo apšvietimo kitimui eksperimentas

Antrasis eksperimentas skirtas patikrinti kaip kinta patobulinto ir originalaus SURF algoritmo atpažinimo tikslumas, keičiantis objekto apšvietumui. Šis eksperimentas buvo atliekamas dviem bandymais: su erdviu paviršiumi (puodeliu) ir plokščiu paviršiumi (arbatos pakeliu). Abiejų bandymų etaloninių objektų paveikslėliai pateikiami atitinkamai 40 pav. ir 41 pav.



**40 pav.** Etaloninis objektas (puodelis) sutapimo keičiant apšvietimą bandymui.



**41 pav.** Etaloninis objektas (arbatos pakelis) sutapimo keičiant apšvietimą bandymui.

Antrojo eksperimento metu etaloninių objektų apšvietimas buvo keičiamas, lempą artinant prie objekto. 60W lemputė buvo artinama nuo 2m ribos iki 10cm. Skirtingo apšvietimo etaloniniai objektai buvo fotografuojami, ir vėliau gauti paveikslėliai lyginami su originaliu etaloninio objekto paveikslėliu panaudojant SURF ir SURF RGB algoritmus. Nufotografuoti, įvairių apšvietimo lygių, etaloninio objekto paveikslėliai pateikiami 42 pav. ir 43 pav.



42 pav. Skirtingai apšviesto etaloninio objekto (puodelio) paveikslukai.



43 pav. Skirtingai apšviesto etaloninio objekto (arbatos pakelio) paveikslukai.

Objektų palyginimo duomenys pateikiami 11-oje ir 12-oje lentelėse. 11-oje lentelėje pateikiamas erdvinio objekto (puodelio) sutapimas su etaloniniu objektu. 12-oje lentelėje pateikiamas plokščio objekto (arbatos pakelio) sutapimas su etaloniniu objektu. Pirmame lentelių stulpelyje pateikiamos SURF algoritmo sutapimo procentinės reikšmės ir šalia pateikiami algoritmo veikimo laikai. Trečiame stulpelyje pateikiamos SURF RGB algoritmo sutapimo procentinės reikšmės ir šalia algoritmo veikimo laikai. Likę trys lentelių stulpeliai parodo procentines sutapimo reikšmes atitinkamai raudoname, žaliame ir mėlyname kanaluose.

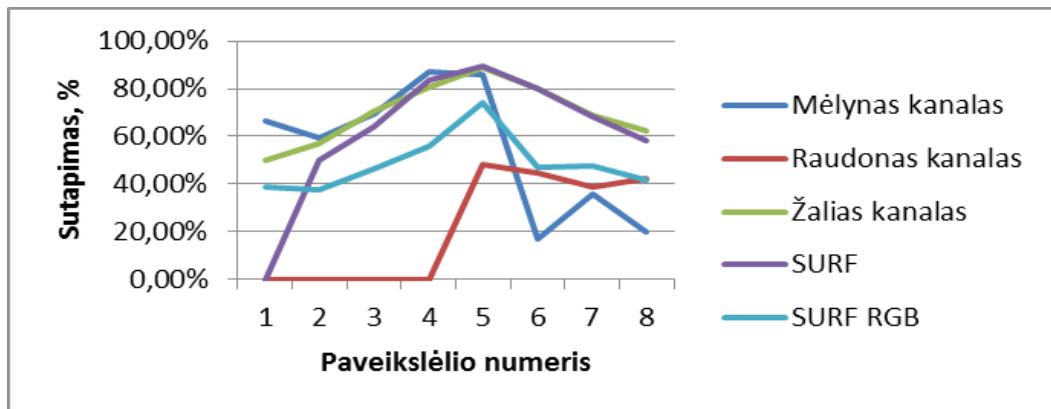
11 lentelė. Sutapimas su etalonu keičiant objekto apšvietimą (puodelis) ir palyginimo laikas

Pav. Nr.	Originalus SURF	Laikas	Pakeistas SURF	Laikas	Raudonas kanalas	Žalias kanalas	Mėlynas kanalas
1	0,00%	0,27	38,89%	0,79	0,00%	50,00%	66,67%
2	50,00%	0,26	37,74%	0,79	0,00%	57,14%	59,09%
3	64,29%	0,31	46,65%	0,82	0,00%	70,37%	69,57%
4	83,33%	0,27	55,93%	0,88	0,00%	80,70%	87,10%
5	89,19%	0,27	74,16%	0,83	48,00%	88,76%	85,71%
6	80,00%	0,26	47,13%	0,86	44,44%	80,28%	16,67%
7	68,29%	0,28	47,85%	0,94	38,78%	69,07%	35,71%
8	58,33%	0,29	41,58%	0,92	42,55%	62,20%	20,00%

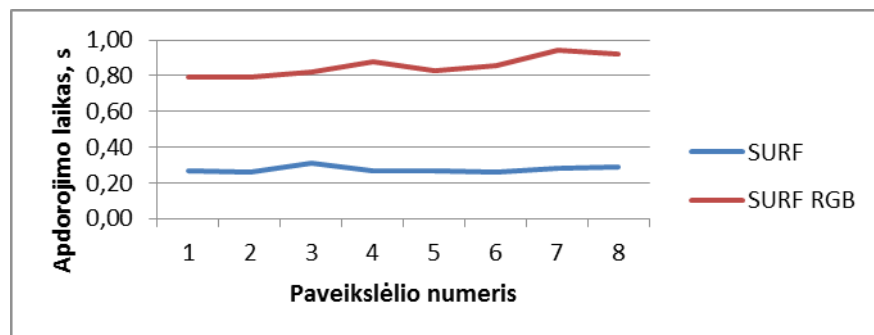
12 lentelė. Sutapimas su etalonu keičiant objekto apšvietimą (arbatos pakelis) ir palyginimo laikas

Pav. Nr.	Originalus SURF	Laikas	Pakeistas SURF	Laikas	Raudonas kanalas	Žalias kanalas	Mėlynas kanalas
1	67,09%	0,40	70,06%	1,21	73,53%	67,11%	69,53%
2	71,43%	0,40	77,26%	1,26	84,31%	77,42%	70,05%
3	71,77%	0,47	73,33%	1,50	72,00%	74,57%	73,44%
4	80,38%	0,53	80,64%	1,62	82,26%	80,61%	79,04%
5	81,79%	0,64	75,99%	2,07	69,95%	77,72%	80,29%
6	72,51%	0,76	68,96%	2,09	57,46%	68,46%	80,58%
7	71,38%	0,71	67,11%	2,06	54,23%	68,24%	78,86%
8	69,44%	0,70	64,30%	2,30	53,48%	65,79%	73,64%

Lentelėse pateikti duomenys taip pat atvaizduojami grafiškai. 44 pav. ir 45 pav. pavaizduoti grafikai parodo skirtingai apšviesto objekto procentinę sutapimo reikšmę su etaloniniu objektu, paveikslėlius apdorojant SURF ir SURF RGB algoritmais. Grafikų x ašyje pažymėti skirtingai apšviestų objektų paveikslėlių numeriai, y ašyje – sutapimo su etaloniniu paveikslėliu procentinė reikšmė.



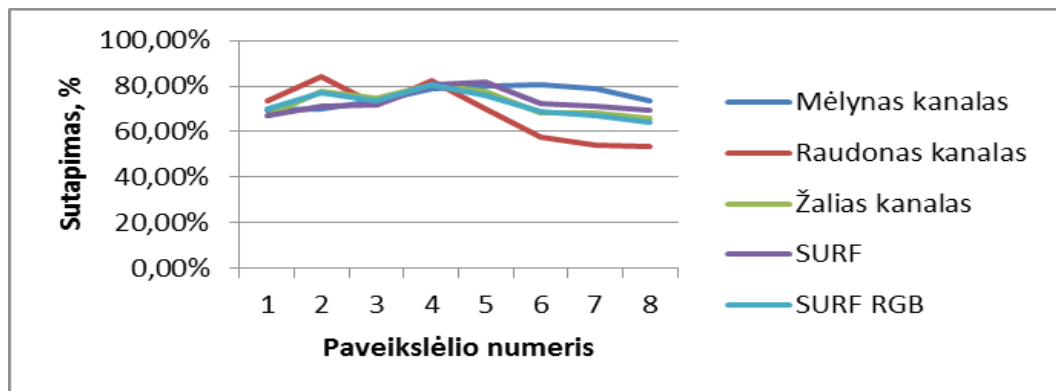
44 pav. Sutapimas su etalonu keičiant objekto apšvietimą (puodelis).



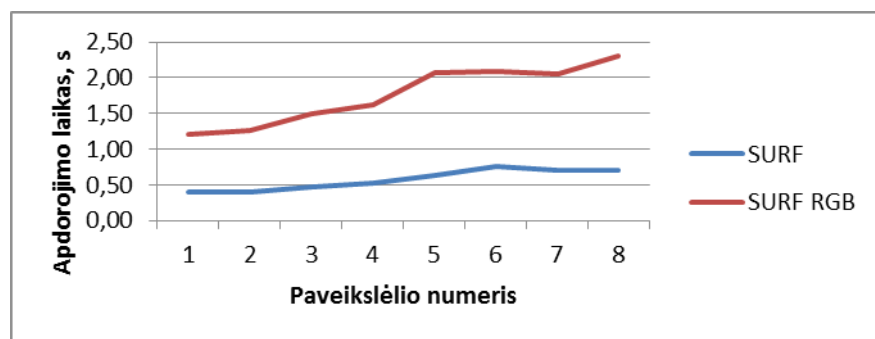
45 pav. Palyginimo laikas keičiant objekto apšvietimą (puodelis).

46 pav. ir 47 pav. pateikiami grafikai parodo SURF ir SURF RGB paveikslėlių palyginimo laikus. Grafikų x ašyje atvaizduojamas apdorojamo paveikslėlio numeris, y ašyje – apdorojimo trukmė sekundėmis.

Iš eksperimento rezultatų matome, jog SURF RGB algoritmas geriau susidoroja su mažiau apšviestais objektais ir atlieka tikslesnį lokalių požymių palyginimą. Tuo tarpu originalus SURF algoritmas tikslesnius rezultatus pateikia apdorojant paveikslėlius prie didesnio apšviestumo. Didesnis apšvietimas nei esantis ant etaloninio objekto pakeičia objekto spalvas, tai ir įtakoja SURF RGB algoritmo netikslumus ir leidžia originaliam SURF algoritmui, kuris apdoroja nespaltvotus paveikslėlius, silpniau reaguoti į spalvų pasikeitimus, taip išgaunant didesnę palyginimo tikslumą.



46 pav. Sutapimas su etalonu keičiant objekto apšvietimą (arbatos pakelis).



47 pav. Palyginimo laikas keičiant objekto apšvietimą (arbatos pakelis).

### 8.3. Gebėjimo atskirti spalvas eksperimentas

Trečiojo eksperimento metu buvo pasirinktos kelios objektų poros, kurios buvo panašios savo forma ir raštu, bet skyrėsi spalvomis. Šis eksperimentas atliktas siekiant patikrinti ar SURF RGB algoritmas sugeba tiksliau palyginti objektus, atsižvelgdamas į jų spalvinę informaciją, nei originalus SURF algoritmas, dirbantis su nespalvotais paveikslėliais. Pasirinkti objektai pateikiami 48 pav.



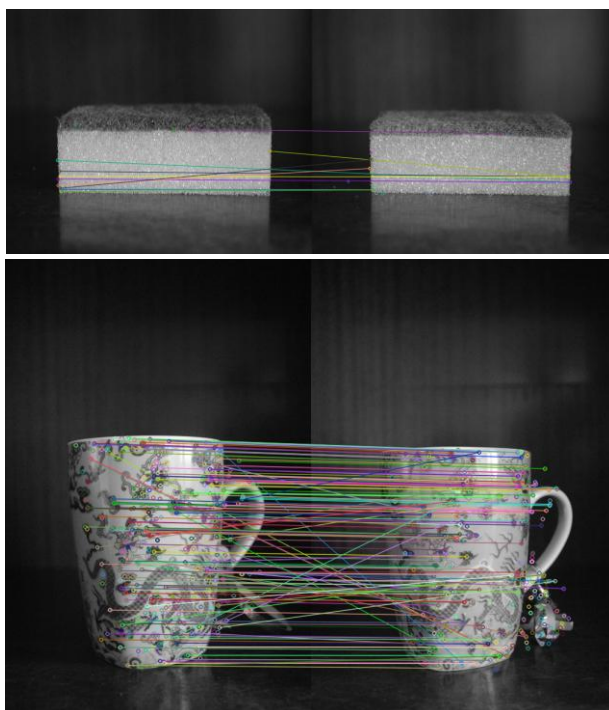
48 pav. Spalvoti objektų paveikslėliai.

Originalus SURF algoritmas sutampančiais laikė 75% (49 pav. viršutinė dalis) lokalių požymių, rastų kempinių paveikslėliuose ir palyginimui užtruko 0,23 s. Tuo tarpu SURF RGB algoritmas parodė tik 34,16% sutampančių lokalių požymių ir užtruko 0,81 s. Požymių sutapimas atitinkamuose RGB kanaluose buvo 36,84% – raudoname (50 pav. viršutinė dalis), 51,35% – žaliame (51 pav. viršutinė dalis), 14,26% – mėlyname (52 pav. viršutinė dalis). Antro bandymo metu buvo panaudoti puodeliai. Originalus SURF algoritmas 40,23% (49 pav. apatinė dalis) jų lokalių požymių laikė sutampančiais, kuriuos palygino per 0,51 s. SURF RGB algoritmas sutampančiais laikė 34,09% požymių, jis apdorojimui užtruko 1,86 s. Požymių sutapimas

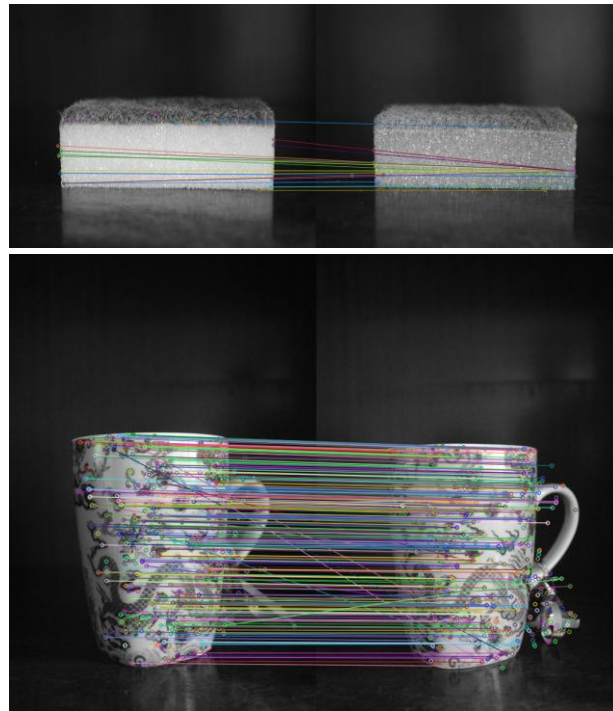


atitinkamuose RGB kanaluose buvo 35,16% – raudoname (50 pav. apatinė dalis), 50,64% – žaliame (51 pav. apatinė dalis), 16,54% – mėlyname (52 pav. apatinė dalis).

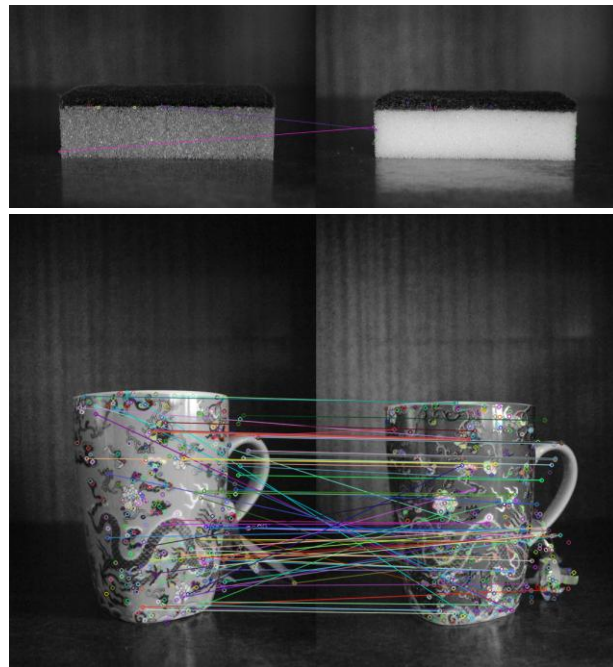
Iš šio eksperimento galime spręsti, jog SURF RGB algoritmas tiksliau palygina objektus, kai sutampa jų tekstūros ir forma, bet skiriasi spalvos. Skirtingų spalvų kanalų požymių palyginimas SURF RGB algoritmui leidžia apytikriai įvertinti objektų spalvų skirtumus.



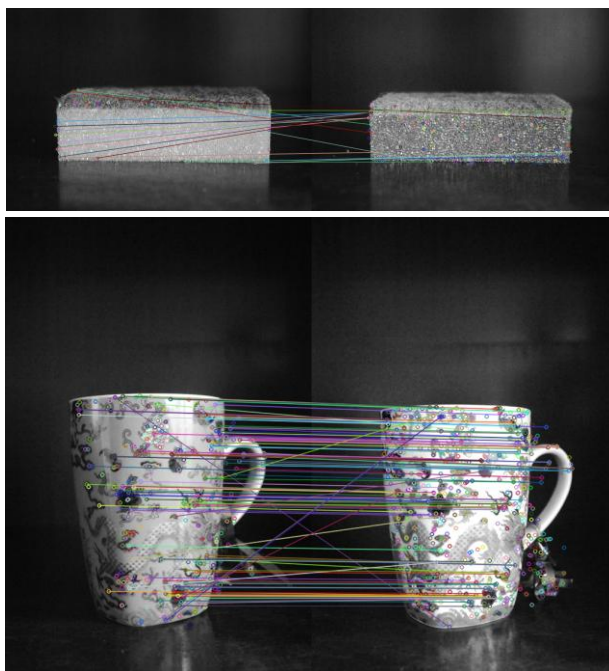
**49 pav.** SURF apdoroti objektai.



**50 pav.** SURF RGB apdoroti objektai (raudonos spalvos kanalas).



**51 pav.** SURF RGB apdoroti objektai (žalios spalvos kanalas).



**52 pav.** SURF RGB apdoroti objektai (mėlynos spalvos kanalas).

Kitų vienodų daiktų palyginimo paveikslėliai pateikiami 2–ame priede, o palyginimo rezultatai 3–iame priede.

## 9. IŠVADOS

Šiandieninėje visuomenėje SURF algoritmas plačiai taikomas ne tik mokslinėje veikloje, bet ir kasdieninėje aplinkoje, tokioje kaip kelių žemėlapių palyginimas pagal palydovines nuotraukas, defektų atpažinimas, kamerų kalibravimui ir kitose panašiose srityse.

Šio darbo tikslas buvo pagerinti plačiai naudojamo SURF algoritmo paveikslėlių palyginimo tikslumą, mažesnę dėmesį kreipiant į palyginimo laiką.

Teorinėje dalyje buvo atlikta algoritmų palyginimo analizė, iš kurios paaiškėjo, jog SURF algoritmas veikia bent penkis kartus greičiau už tos pačios kategorijos DOG algoritmą, kuris kaip ir SURF buvo pritaikytas greičiau apdoroti paveikslėlius su mažesniu tikslumu.

SURF algoritmas, palyginus su kitais lokalių požymių palyginimo algoritmais, turi didžiausią spartą ir palyginimo pasikartojimą, be to, yra nekintamas objekto pasukimo ir dydžio keitimo atžvilgiu.

Šiame darbe aprašytas SURF algoritmo patobulinimas suteikia galimybę tiksliau palyginti objektus, kurių tekstūra ir forma sutampa, bet skiriasi spalvos. Iš tyrimo rezultatų matyti, kad modifikuotas algoritmas atlieka tikslesnę, mažiau apšviestų objektų palyginimą, tačiau sunkiai susidoroja su didesniu objektų apšvietimu.

Objektų pasukimo atveju, SURF algoritmas praranda didelę dalį palyginimo požymių, kai pasukimo kampas būna didesnis nei  $40^\circ$ . Modifikuoto SURF algoritmo požymių kiekis taip stipriai nesumažėja, nes trys spalvų kanalai kompensuoja vienas kito prarastus požymius.

Paveikslėliai turi labai skirtingus kiekius taškų ir deformacijas, kilusias dėl apšvietimo ir kameros trikdžių, todėl sunku tiksliai nustatyti modifikuoto algoritmo palyginimo tikslumo pagerėjimą, originalaus SURF algoritmo atžvilgiu, tačiau, apibendrinus tyrimų rezultatus, modifikuotas SURF algoritmas veikia iki 15% tiksliau už originalą.

Šiame darbe atliktos SURF algoritmo modifikacijos pagerino atpažinimo tikslumą, spartą palikdamos antroje vietoje, dėl to algoritmas veikia iki trijų kartų ilgiau.

Modifikuoto SURF algoritmo tolesnio tobulinimo ir tyrimo veiklos galėtų apimti: paveikslėlių apdorojimą prieš skaidant į kanalus (apšviestumo ir spalvų normalizavimą), po skaidymo į kanalus (kanalų intensyvumų normalizavimą, papildomą triukšmų filtravimą) ir lokalių požymių išrinkimo bei jų palyginimo spartos didinimą, apjungiant spalvų kanalų informaciją.

## 10. LITERATŪRA

1. Bay H.; Tuytelaars T. ir Van Gool L. SURF: Speeded up robust features// Proceedings of the European Conference on Computer Vision. 2006 p. 404–417.
2. Biederman I. Recognition-by-components: A theory of human image understanding// Psychological Review. T. 2, Nr. 94 (1987), p. 115–147.
3. Carbonetto P.; de Freitas N. ir Barnard K. A statistical model for general contextual object recognition// Proceedings of the European Conference on Computer Vision. Pirma dalis, 2004, p. 350–362.
4. Carson C.; Belongie S.; Greenspan S. ir Malik J. Blobworld: Image segmentation using expectation-maximization and its application to image querying// IEEE Transactions on Pattern Analysis and Machine Intelligence. T. 24 (2002), p. 1026–1038.
5. FeiFei L.; Perona P. A Bayesian hierarchical model for learning natural scene categories// Proceedings of the Conference on Computer Vision and Pattern Recognition. T. 2 (2005), p. 524–531.
6. Fergus R.; Perona P. ir Zisserman A. Object class recognition by unsupervised scale-invariant learning// Proceedings of the Conference on Computer Vision and Pattern Recognition. 2003, p. 264–271.
7. Mikolajczyk K.; Leibe B. ir Schiele B. Local features for object class recognition// Proceedings of the International Conference on Computer Vision. 2005, p. 525–531.
8. Mikolajczyk K.; Tuytelaars T. ; Schmid C.; Zisserman A. ir kt. A comparison of affine region detectors// International Journal of Computer Vision, T. 65, Nr. 1/2 (2005), p. 43–72.
9. Murase H.; Nayar S. Visual learning and recognition of 3D objects from appearance// International Journal on Computer Vision. T. 14 (1995), p. 5–24.
10. Nowak E.; Jurie F. ir Triggs B. Sampling strategies for bag-of-features image classification// Proceedings of the European Conference on Computer Vision. 2006, p. 490–503.

11. Schaffalitzky F.; Zisserman A. Viewpoint invariant texture matching and wide baseline stereo// Proceedings of the International Conference on Computer Vision. 2001, p. 636–643.
12. Schiele B.; Crowley J. L. Probabilistic object recognition using multidimensional receptive field histograms// Proceedings of the International Conference on Pattern Recognition. T. 2 (1996), p. 50–54.
13. Schmid C.; Mohr R. Local gray-value invariants for image retrieval// IEEE Transactions on Pattern Analysis and Machine Intelligence. T. 19, Nr. 5 (1997), p. 530–534.
14. Schmid C.; Mohr R. ir Bauckhage C. Evaluation of interest point detectors// International Journal of Computer Vision. T. 37, Nr. 2 (2000), p. 151–172.
15. Sharon E.; Brandt A. ir Basri R. Segmentation and boundary detection using multiscale intensity measurements// Proceedings of the Conference on Computer Vision and Pattern Recognition. 2001, p. 469–476.
16. Shi J.; Tomasi C. Good features to track// Proceedings of the Conference on Computer Vision and Pattern Recognition. 1994, p. 593–600.
17. Swain, M.; D. Ballard, D. Color indexing// International Journal in Computer Vision. T. 7 (1991), p. 11–32.
18. Turk, M. A.; Pentland A. P. Eigenfaces for face recognition// Proceedings of the Conference on Computer Vision and Pattern Recognition. 1991, p. 586–591.
19. Tuytelaars T.; Schmid C. Vector quantizing feature space with a regular lattice// Proceedings of the International Conference on Computer Vision. 2007.
20. Winn J.; Criminisi A.; Minka T. Object categorization by learned universal visual dictionary// Proceedings of the International Conference on Computer Vision. T. 2 (2005), p. 1800–1807.
21. Viola P.; Jones M. Rapid object detection using a boosted cascade of simple features// Proceedings of the Conference on Computer Vision and Pattern Recognition. T. 1 (2001), p. 511–518.

22. Drejeris L.; Kardzys E.; Kriščiūnas K. „Universalios vaizdo atpažinimo sistemos kūrimas ir tyrimas// Magistrinio projekto pilna dokumentacija. 2012, Kaunas.
23. Brown M.; Lowe D. Invariant features from interest point groups// BMVC. 2002.
24. Lindeberg T. Feature detection with automatic scale selection// IJCV. 1998, 79–116.
25. Mikolajczyk K.; Schmid C. Indexing based on scale invariant interest points// ICCV. T. 1 (2001), p. 525–531.
26. Neubeck A.; Van Gool L. Efficient non-maximum suppression// ICPR. 2006.
27. Ford A.; Roberts A. Colour Space Conversions. 1998, p. 3-17.

## **11. PRIEDAI**

Su magistro darbu pateikiami šie priedai:

1. Požymių aptikimo algoritmų apžvalgos lentelė.
2. Kiti palyginimo bandymuose panaudoti daiktų paveikslėliai.
3. Kitų paveikslėlių palyginimo rezultatų lentelė.
4. UAB „Informacinių technologijų organizacija“ programų sistemos perdavimo ir aprobavimo akto kopija.
5. Valstybinė įmonė „Registru centras“ programų sistemos perdavimo ir aprobavimo akto kopija.
6. KTU jaunųjų mokslininkų darbų parodos “Technorama” padėkos už sėkmingą dalyvavimą parodoje kopija.



## 1 priedas. Požymių aptikimo algoritmų apžvalgos lentelė

13 lentelė. Požymių aptikimo algoritmų apžvalga

Požymių aptikimo algoritmas	Kampai	Dėmės	Regionai	Pasukimo invariantiškumas	Dydžio invariantiškumas	Panašumų invariantiškumas	Pasikartojimas	Lokalizacijos tikslumas	Tvirtumas	Našumas
Harris	*			*			+++	+++	+++	++
Hessian		*		*			++	++	++	+
SUSAN	*			*			++	++	++	+++
Harris-Laplace	*	(*)		*	*		+++	+++	++	+
Hessian-Laplace	(*)	*		*	*		+++	+++	+++	+
DoG	(*)	*		*	*		++	++	++	++
SURF	(*)	*		*	*		++	++	++	+++
Harris-Affine	*	(*)		*	*	*	+++	+++	++	++
Hessian-Affine	(*)	*		*	*	*	+++	+++	+++	++
Salient Regions	(*)	*		*	*	(*)	+	+	++	+
Edge-based	*			*	*	*	+++	+++	+	+
MSER			*	*	*	*	+++	+++	++	+++
Intensity-based			*	*	*	*	++	++	++	++
Superpixels			*	*	(*)	(*)	+	+	+	+

## 2 priedas. Kitų palyginimo bandymuose panaudotų daiktų paveikslėliai



53 pav. Pirmas papildomas palyginimo objektas – puodelis.



54 pav. Antras papildomas palyginimo objektas – knyga.



55 pav. Trečias papildomas palyginimo objektas – puodelis.

### 3 priedas. Kitų paveikslėlių palyginimo rezultatų lentelė

14 lentelė. Kitų paveikslėlių palyginimo rezultatai

<b>Pav. Nr.</b>	<b>Originalus SURF</b>	<b>Laikas</b>	<b>Pakeistas SURF</b>	<b>Laikas</b>	<b>Raudonas kanalas</b>	<b>Žalias kanalas</b>	<b>Mėlynas kanalas</b>
41 pav.	13,69%	0,38	19,03%	1,22	17,39%	24,27%	15,42%
42 pav.	41,29%	0,47	46,45%	1,50	40,96%	51,45%	46,45%
43 pav.	29,50%	0,39	40,46%	1,22	34,89%	54,37%	32,11%

## 4 priedas. UAB „Informacinių technologijų organizacija“ programų sistemos perdavimo ir aprobavimo akto kopija



### UAB „INFORMACINIŲ TECHNOLOGIJŲ ORGANIZACIJA“

Savanorių pr. 184, 44150 Kaunas

Tel.: 8-671 1 39 58

Faks.: 8-37 300 489

El.paštas: [info@ito.lt](mailto:info@ito.lt)

### PROGRAMŲ SISTEMOS PERDAVIMO IR APROBAVIMO AKTAS

20 11 m. gruodžio 12 d.

Programų sistemos pavadinimas „Universali vaizdo atpažinimo sistema“

Kūrinio tipas programinė įranga

Programų sistemos sukūrimo data 20 11 m. gruodžio 12 d.

Kūrinio įteikimo UŽSAKOVUI data 20 11 m. gruodžio 12 d.

Užsakovo arba trečiojo asmens Kūrinio aprobavimo rezultatas:

Programinė įranga sukurta ir įdiegta sėkmingai. Darbas priimtas ir laikomas tinkamu.

Kūrinio aprobavimo data 20 11 m. gruodžio 12 d.

Kūrinio originalo saugotojas - UAB „Informacinių technologijų organizacija“

AUTORIUS

Liudas Drejeris

(vardas, pavardė)

(parašas)

UŽSAKOVAS

Paulius Lazauskas

(vardas, pavardė)



Paulius Lazauskas

Direktorius

## 5 priedas. Valstybinė įmonė „Registru centras“ programų sistemos perdavimo ir aprobavimo akto kopija



REGISTRŲ CENTRAS

### VALSTYBĖS ĮMONĖ REGISTRŲ CENTRAS

Vinco Kudirkos g. 18-3, 03105 Vilnius,  
tel. (8 5) 268 8202, faks. (8 5) 268 8311, el. p. [info@registrucentras.lt](mailto:info@registrucentras.lt).  
Duomenys kaupiami ir saugomi Juridinių asmenų registre, kodas 124110246

Kauno Technologijos universitetas  
Informatikos fakultetas  
Studentų 50-411, LT 3031 Kaunas  
Tel. (8 37) 300350  
Faksas (3 37) 300352

### PROGRAMŲ SISTEMOS PERDAVIMO IR APROBAVIMO AKTAS

2011 m. Gruodžio 19 d.

Programų sistemos pavadinimas „Universali vaizdo atpažinimo sistema“

Kūrinio tipas Programinė įranga

Programų sistemos sukūrimo data 2011 m. Gruodžio 15 d.

Kūrinio įteikimo UŽSAKOVUI data 2011 m. Gruodžio 19 d.

Užsakovo arba trečiojo asmens Kūrinio aprobavimo rezultatas:

Sistema įdiegta sėkmingai ir atitinka iškeltus reikalavimus.

Kūrinio aprobavimo data 2011 m. Gruodžio 19 d.

Kūrinio originalo saugotojas Ernestas Kardzys

AUTORIUS

Ernestas Kardzys  
(vardas, pavardė)

*E. Kardzys*  
(parašas)

UŽSAKOVAS

Algirdas Remeikis  
(vardas, pavardė)

*Remeikis*  
(parašas)

Valstybės įmonės Registrų centro  
Informacinių technologijų centro  
IS konstravimo departamento viršininkas  
Asmens pareigų pavadinimas

*Antanas Krikščiūnas*  
(Parašas)

Antanas Krikščiūnas  
(Vardas ir pavardė)

Paraga  
Algirdas Remeikis  
2011 12 19

*Remeikis*

VĮ Registrų centras, Vinco Kudirkos g. 18-3, 03105 Vilnius, tel. (8 5) 268 8202, faks. (8 5) 268 8311, el. p. [info@registrucentras.lt](mailto:info@registrucentras.lt).

**6 priedas. KTU jaunųjų mokslininkų darbų parodos “Technorama” padėkos už sėkmingą dalyvavimą parodoje kopija**

