



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

Tomas Valatkevičius

**MOBILIŲJŲ ĮRENGINIŲ JUTIKLIŲ PARODYMŲ ĮTAKOS
VAIZDŲ ATPAŽINIMUI TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Lekt. dr. Kęstutis Jankauskas

KAUNAS, 2018

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

**MOBILIŲJŲ ĮRENGINIŲ JUTIKLIŲ PARODYMŲ ĮTAKOS
VAIZDŲ ATPAŽINIMUI TYRIMAS**

Baigiamasis magistro projektas
Informatika (kodas M4016O21)

Vadovas

(parašas) Lekt. dr. Kęstutis Jankauskas
(data)

Recenzentas

(parašas) Lekt. dr. Mikas Binkis
(data)

Projektą atliko

(parašas) Tomas Valatkevičius
(data)

KAUNAS, 2018



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Informatikos fakultetas

(Fakultetas)

Tomas Valatkevičius

(Studento vardas, pavardė)

Informatika (kodas M4016O21)

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Mobiliųjų įrenginių jutiklių parodymų įtakos vaizdų atpažinimui tyrimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

220 _____ m. _____ d.
Kaunas

Patvirtinu, kad mano, Tomo Valatkevičiaus, baigiamasis projektas tema „Mobiliųjų įrenginių jutiklių parodymų įtakos vaizdų atpažinimui tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Valatkevičius Tomas. Mobilųjų įrenginių jutiklių parodymų įtakos vaizdų atpažinimui tyrimas. Magistro baigiamasis projektas / vadovas Lekt. dr. Kęstutis Jankauskas; Kauno technologijos universitetas, Informatikos fakultetas.

Mokslo kryptis ir sritis:

Reikšminiai žodžiai: *vaizdų atpažinimas, charakteringi taškai, vaizdų apdorojimas*

Kaunas, 2018. 81 p.

SANTRAUKA

Vaizdų atpažinimas šiomis dienomis yra neatsiejamas nuo dirbtinio intelekto, robotų ir kitų kompiuterinės regos sprendimų. Nors pats vaizdų atpažinimas yra ne tokia didelė naujiena, nes kompiuteriai tampa našesni, bet augant mobiliųjų įrenginių rinkai ir techninėms specifikacijoms, visi vaizdų atpažinimo skaičiavimai bandomi perkelti į mobiliuosius įrenginius.

Šiame darbe analizuojama ir siekiama iširti mobiliųjų įrenginių jutiklių parodymų įtaką vaizdų atpažinimui, o tiksliau charakteringų taškų išskyrimui. Pateikiama mūsų sukurta sistema, skirta atlikti eksperimentus su skirtingais charakteringų taškų detektoriais, taškų aprašais ir palyginimo metodais. Taip pat pateikiama eksperimentų rezultatai, jų aptarimai. Eksperimentiniu būdu pateikiami siūlymai, kaip pagerinti charakteringų taškų aptikimą. Pateikiamos tyrimo išvados.

Valatkevičius, Tomas. *Mobile Device Sensors Data Influence to Image Recognition Precision: Master's thesis in* / supervisor assoc. Lekt. dr. Kęstutis Jankauskas. The Faculty of Informatics, Kaunas University of Technology.

Research area and field:

Key words: *Image recognition, key-points, image processing.*

Kaunas, 2018. 81 p.

SUMMARY

Image recognition nowadays is found in a lot of spaces. Robots, neural networking, artificial intelligence, all are using computer vision to make some actions. But what about mobile devices? Do they support image recognition or do they sensors can help to improve image recognition? In this paper we are discussing the influence of mobile devices sensors data to image recognition. During research part feature detection must be analyzed. Detectors and descriptors are implemented in our system which can also give as opportunity to calculate matches between two different images. The results of research showing feature detection dependency on mobile devices image sensors. The conclusions showing how digital noise can be affected to improve detection matches.

TURINYS

Santrauka	4
Summary.....	4
Turinys.....	5
Lentelių sąrašas	7
Paveikslų sąrašas	8
Terminų žodynas	10
Darbo tikslas ir uždaviniai	11
Darbo struktūra	11
1 Mobilųjų įrenginių jutiklių analizė.....	12
1.1 Jutiklių tipai ir jų parametrai	12
1.1.1 Akselerometras	13
1.1.2 Giroskopas	14
1.1.3 Aplinkos temperatūros jutiklis.....	14
1.1.4 Gravitacijos jutiklis.....	15
1.1.5 Aplinkos šviesos jutiklis (fotometras)	15
1.1.6 Magnetometras	15
1.1.7 Orientacijos jutiklis.....	16
1.1.8 Barometras	16
1.1.9 Artumo jutiklis.....	16
1.1.10 Drėgmės jutiklis	17
1.1.11 Pasisukimo krypties jutiklis	17
1.1.12 Vaizdo jutiklis.....	17
1.2 Kompiuterinės regos įrankiai	19
1.2.1 OpenCV	20
1.2.2 SimpleCV	20
1.2.3 Šablonų atpažinimas	20
1.2.4 Pikselių grupės aptikimo metodas	22
1.2.5 Požymių aptikimo metodas.....	22
1.2.6 Metodų analizės apibendrinimas	24
1.3 Charakteringų taškų aptikimas ir aprašymas	24
1.3.1 Detektoriai ir aprašai.....	24
1.3.2 AGAST	25
1.3.3 KAZE.....	25
1.3.4 BRISK.....	27
1.3.5 FAST.....	28
1.3.6 BRIEF.....	29
1.3.7 ORB	30
1.3.8 MSER	30
1.3.9 SIFT	31
1.3.10 SURF	34
1.3.11 GFTT	36
1.3.12 BLOB.....	37
1.4 Skaitmeninis vaizdų apdorojimas	37
1.4.1 Gauso suliejimas.....	38
1.4.2 Aštrinimas.....	38
1.4.3 Gama korekcija.....	39
2 Charakteringų taškų aptikimo ir aprašymo sistemos projektinė dalis.....	41
2.1 Sistemos funkciniai reikalavimai	41
2.2 Statinis sistemos vaizdas	44
2.3 Sistemos dinaminis vaizdas	47

3	Mobiliųjų jutiklių parodymų įtakos vaizdų atpažinimui Tyrimas.....	49
3.1	Sistema skirta eksperimentuoti su vaizdų atpažinimo algoritmais	49
3.1.1	Įvesties atvaizdo užkrovimo filtras.....	50
3.1.2	Atvaizdo apdorojimo filtras	50
3.1.3	Charakteringų taškų detektoriaus filtras	51
3.1.4	Charakteringų taškų aprašo filtras	51
3.1.5	Atvaizdų palyginimo filtras	52
3.1.6	Duomenų atvaizdavimo filtras.....	52
3.2	Tyrimo duomenų rinkinys.....	53
4	Mobiliųjų įrenginių jutiklių įtakos vaizdų atpažinimui eksperimentinė dalis.....	55
4.1	Priemonės eksperimentui atlikti.....	55
4.2	Charakteringų taškų aptikimo priklausomybės nuo įrenginio vaizdo jutiklio eksperimentas.....	56
4.2.1	Tikslas.....	56
4.2.2	Metodika.....	56
4.2.3	Eksperimento rezultatai	56
4.2.4	Rezultatų aptarimas	61
4.3	Charakteringų taškų aptikimo priklausomybės nuo įrenginio vaizdo jutiklio jautrumo parametrų eksperimentas.....	62
4.3.1	Tikslas.....	62
4.3.2	Metodika.....	62
4.3.3	Eksperimento rezultatai	62
4.3.4	Rezultatų aptarimas	67
4.4	Mobiliųjų įrenginių vaizdo jutiklio triukšmo įtakos, charakteringų taškų aptikimui, mažinimo eksperimentas.....	67
4.4.1	Tikslas.....	67
4.4.2	Metodika.....	68
4.4.3	Eksperimento rezultatai	68
4.4.4	Rezultatų aptarimas	70
5	IŠVADOS.....	71
6	LITERATŪROS SĄRAŠAS.....	73
7	Priedai.....	75
7.1	Straipsnis „The Mobile Application Based on Augmented Reality for Learning STEM Subjects“	75

LENTELIŲ SĄRAŠAS

lentelė 1.1 Įrenginių jutiklių apibendrinamas	19
lentelė 2.1 Panaudos atvejis Nr. 1	42
lentelė 2.2 Panaudos atvejis Nr. 2	42
lentelė 2.3 Panaudos atvejis Nr. 3	42
lentelė 2.4 Panaudos atvejis Nr. 4	42
lentelė 2.5 Panaudos atvejis Nr. 5	42
lentelė 2.6 Panaudos atvejis Nr. 6	43
lentelė 2.7 Panaudos atvejis Nr. 7	43
lentelė 2.8 Panaudos atvejis Nr. 8	43
lentelė 2.9 Panaudos atvejis Nr. 9	43
lentelė 2.10 Panaudos atvejis Nr. 10	43
lentelė 4.1 Eksperimento priemonės.....	55

PAVEIKSLŲ SĄRAŠAS

pav. 1.1 Akselerometro ir giroskopo dydžių palyginimas.....	14
pav.1.2 Mobilųjų įrenginių jutikliai; a) artumo, b) akselerometras, c) barometras, d) magnetometras, e) giroskopas.....	17
pav. 1.3 Vaizdo jutiklių dydžių palyginimas [12]	18
pav. 1.4 Šablonų aptikimo metodas [23]	21
pav. 1.5 Kampinių taškų išskyrimas [26]	23
pav. 1.6 <i>Gauso</i> ir netiesinio suliejimo skirtumas [5].....	26
pav. 1.7 <i>BRISK</i> charakteringų taškų paieškos lango reikšmių seka [15].....	27
pav. 1.8 detektorių palyginimas [4].....	28
pav. 1.9 <i>FAST</i> detektoriaus paieškos apskritimas su paieškos pikselių numeravimu	29
pav. 1.10 Duomenų sutraukimas <i>SIFT</i> detektoriuje [6]	32
pav. 1.11 Charakteringo taško radimas/fiksavimas <i>SIFT</i> apraše [6]	32
pav. 1.12 Charakteringo taško aprašas <i>SIFT</i> apraše [6]	34
pav. 1.13 Orientacijos nustatymas <i>SURF</i> [6]	35
pav. 1.14 Skirtingų dydžių paieškos langai su priskirta dominuojančia orientacija [6].....	36
pav. 1.15 Aprašas kvadratiname paieškos lange su 4x4 išskyrimu apie charakteringąjį tašką [6]	36
.....	36
pav. 1.16 Mokslinių tyrimų rezultatai palyginantys <i>GFTT</i> detektoriaus veikimą su kitais algoritmais [14].....	37
pav. 1.17 <i>Gauso</i> suliejimo realizacijos pavyzdys [24]	38
pav. 1.18 Kraštinių priklausomybė nuo skirtingų faktorių [17]	39
pav. 1.19 Gama korekcijos pavyzdys [2]	39
pav. 2.1 Sistemos panaudos atvejų diagrama	41
pav. 2.2 Sistemos paketų diagrama	44
pav. 2.3 Sistemos klasių įgyvendinančių Pipes and Filters projektavimo šabloną diagrama.....	46
pav. 2.4 Programos veiklos diagrama.....	47
pav. 2.5 Grandinės skaičiavimų veiklos diagrama	48
pav. 3.1 <i>Pipes and Filters</i> šablono orientacinė schema [27].....	49
pav. 3.2 Algoritmų kombinavimas sukurtoje sistemoje	50
pav. 3.3 Užrakto greičio reikšmės kitimo žingsnis.....	54
pav. 3.4 Jautrumo reikšmių įtakos atvaizdui padidintas vaizdas.....	54
pav. 4.1 <i>ORB</i> detektoriaus ir aprašo rezultatai.....	57
pav. 4.2 <i>SURF</i> detektoriaus ir aprašo rezultatai	57
pav. 4.3 <i>AKAZE</i> detektoriaus ir aprašo rezultatai	58
pav. 4.4 <i>BRISK</i> detektoriaus ir aprašo rezultatai	59
pav. 4.5 <i>SIFT</i> detektoriaus ir aprašo rezultatai	59
pav. 4.6 Detektoriaus <i>ORB</i> ir aprašo <i>BRIEF</i> rezultatai	60
pav. 4.7 Detektoriaus <i>ORB</i> ir aprašo <i>FREAK</i> rezultatai	61
pav. 4.8 Detektoriaus <i>ORB</i> ir aprašo <i>FREAK</i> rezultatai	62
pav. 4.9 Detektoriaus <i>ORB</i> ir aprašo <i>BRIEF</i> rezultatai	63
pav. 4.10 Detektoriaus ir aprašo <i>SIFT</i> rezultatai	64
pav. 4.11 Detektoriaus ir aprašo <i>BRISK</i> rezultatai	65
pav. 4.12 Detektoriaus ir aprašo <i>AKAZE</i> rezultatai	65
pav. 4.13 Detektoriaus ir aprašo <i>SURF</i> rezultatai	66
pav. 4.14 Detektoriaus ir aprašo <i>ORB</i> rezultatai	67
pav. 4.15 Mobiliojo įrenginio <i>Smasung Galaxy S II</i> vaizdo jutiklio keliamų triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai	68
pav. 4.16 Mobiliojo įrenginio <i>Smasung Galaxy Note</i> triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai	69

pav. 4.17 Mobiliojo įrenginio <i>Smasung Galaxy J5</i> vaizdo jutiklio keliamų triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai	70
--	----

TERMINŲ ŽODYNAS

Detektorius	charakteringų taškų aptikimo algoritmas;
Gauso funkcija	simetrinis varpo formos grafikas, kurio reikšmės greitai krenta žemyn, funkcijos argumento reikšmėms artėjant į teigiamą arba neigiamą begalybę;
Aprašas	(angl. <i>Descriptor</i>) charakteringų taškų aprašymo algoritmas;
Atitikimas (angl. <i>Matches</i>)	charakteringų taškų sutapimas, kai jie yra lyginami tarp dviejų skirtingų atvaizdų;
Haro bangelės	kvadrato formos sumažintų paveikslų ir jų funkcijų seka;
Exif	keičiamasis paveikslo failo formatas, leidžiantis užkoduoti visus vaizdo jutiklio papildomus ir pirminius duomenis. Tai yra duomenų apie vaizdo jutiklių parametrus kodavimo standartas;
Charakteringas taškas	atvaizdo vieta, išskirta detektoriaus, pasižyminti charakteringais parametrais;
Užrakto greitis	terminas sutinkamas fotografijoje, laikas per kurį šviesos srautas krenta į vaizdai jautrią plokštelę (vaizdo jutiklį);
ISO	(angl. <i>International Organization of Standardization</i>) standartizuotas vaizdo jutiklio jautrumo matas. Šis standartas vienu kintamuoju apibūdina visas vaizdo jutiklio jautrumo charakteristikas;
Diafragma	metalinų plokštelių sistema, skirta kontroliuoti šviesos srauto patekimą ant vaizdo jutiklio.

ĮVADAS

Šiomis dienomis technologijų rinka greitai plečiasi, o mobilieji įrenginiai tampa vis našesni dėl aparatinės įrangos dydžio mažėjimo neprarandant techninių parametrų, tačiau mobiliųjų įrenginių naudotojai tampa vis reiklesni ir siekia greito rezultato vaizdų atpažinimo srityje. Šiame darbe siekiama ištirti ar vadovaujantis mobiliųjų įrenginių jutikliais galima paspartinti charakteringų taškų aptikimą atvaizduose, naudojant mobilius įrenginius.

Darbo tikslas ir uždaviniai

Darbo tikslas – ištirti mobiliųjų jutiklių parametrų įtaką charakteringų taškų aptikimui ir pasiūlyti sprendimą, leidžiantį sumažinti vaizdo jutiklio keliamų triukšmų įtaką atvaizdų charakteringų taškų išskyrimui. Išskiriami darbo uždaviniai:

1. Išanalizuoti ir nustatyti mobiliųjų įrenginių jutiklių parodymų įtaką charakteringų taškų aptikimui;
2. Nustatyti charakteringų taškų aptikimo priklausomybę nuo mobiliojo įrenginio vaizdo jutiklio jautrumo;
3. Eksperimentiniais tyrimais nustatyti vaizdo triukšmų įtaką charakteringų taškų aptikimui ir pasiūlyti įtakos mažinimo sprendimą;
4. Išanalizuoti mobiliųjų įrenginių jutiklių tipus, siekiant išsiaiškinti, kuriuos galima naudoti vaizdų atpažinimui;
5. Išanalizuoti charakteringų taškų aptikimo ir aprašų algoritmus;
6. Suprojektuoti ir realizuoti sistemą, įgalinančią atlikti eksperimentinius tyrimus.

Darbo struktūra

Šis darbas sudarytas iš analizės, projektavimo, tyrimo ir eksperimentinės dalių. Analizėje apžvelgiami ir analizuojami mobiliųjų įrenginių jutikliai, taip pat pateikiama susisteminta jutiklių lentelė, pagal kurią bus atliekama eksperimentinė dalis. Taip pat analizuojami charakteringų taškų išskyrimo atvaizduose detektoriai ir aprašai, pateikiama jų veikimo principai ir trumpai aptariami tiesiniai vaizdų apdorojimo filtrai. Projektinėje dalyje pateikiamos ir aptariamos sukurtos sistemos klasių, panaudos atvejų, paketų ir veiklos diagramos. Tyrimo dalyje aprašoma sukurta sistema, pristatomas projektavimo šablonas ir sukurtos sistemos komponentai. Taip pat tyrimo dalyje pateikiama informacija apie eksperimentams skirtą duomenų rinkinį. Eksperimentų dalyje aprašomi tyrimo atlikti eksperimentai, pateikiami ir apžvelgiami jų rezultatai vėliau paskutinėje darbo dalyje pateikiamos išvados.

1 MOBILIŲJŲ ĮRENGINIŲ JUTIKLIŲ ANALIZĖ

Šioje dalyje analizuojama jutikliai, esantys mobiliuosiuose įrenginiuose. Apžvelgiama jutiklių kategorijos ir tipai. Aprašomi jutiklių parametrai ir jų pateikiama informacija. Kiekvienas jutiklis yra aprašomas atskirai, pateikiant pagrindines jo savybes ir paskirties tikslus. Toliau analizuojama vaizdų atpažinimo mobiliuosiuose įrenginiuose tema. Apžvelgiama kas jau yra sukurta, kokie yra išradimai. Analizuojama mobiliųjų įrenginių jutiklių parodymų įtaka vaizdų atpažinimo algoritmams. Paskutinėje analizės dalyje pateikiama jutiklių parodymų palyginimo metodikos analizė, apžvelgiami egzistuojantys palyginimo metodai. Pateikiama kita su jutiklių parodymų palyginimu susijusi informacija.

1.1 Jutiklių tipai ir jų parametrai

Dauguma mobiliųjų įrenginių turi įvairius jutiklius, kurie gali matuoti judėjimo kryptį, pasisukimo kampą, šviesos srauto kiekį ir kitus mobiliojo įrenginio parametrus (jutiklių parodymus), kuriuos interpretuojame, darydami prielaidas apie aplinką. Šie mobiliųjų įrenginių jutikliai pateikia tikslius duomenis operacinei įrenginio sistemai, kuri juos apdorojus atitinkamai panaudoja tolesniam operacinės sistemos funkcionavimui. Šie jutikliai leidžia stebėti įrenginio buvimo vietą erdvėje, taip pat galima stebėti pakeitimus įrenginio aplinkoje. Jutikliai yra skirstomi į 3 pagrindines kategorijas:

1. Judesio jutikliai

Pagrindinis šių jutiklių tikslas – matuoti prietaisą veikiančias pagreičio ir sukimosi jėgas erdvėje, naudojant tris (x, y, z) ašis. Šią kategoriją sudaro akselerometrai, gravitacijos jutikliai, giroskopai ir sukimosi krypties jutikliai. Jau pagal šios kategorijos pavadinimą galima suprasti, jog šie jutikliai naudojami prietaiso judėjimui nustatyti: judėjimui aukštyn, žemyn, kairėn, dešinėn, o taip pat pasisukimą ir kratymą. Šių judesių fiksavimas yra atvirkštinė funkcija to kaip yra judinamas įrenginys. Jeigu įrenginys yra judinamas erdvėje tai duomenys yra tokie patys. Jeigu įrenginys judinamas į kairę pusę, fiksuojama, kad jis pajudėjo į kairę, šiuo atveju duomenų stebėjimas yra santykinis prietaiso atžvilgiu. Tačiau duomenis fiksuoti galima ir globalios planetos koordinatų sistemos atžvilgiu.

2. Aplinkos parametrų jutikliai

Šie jutikliai matuoja įvairius aplinkos veiksnių parametrus, aplink prietaisą esančius aplinkos poveikio parametrus, tokius kaip temperatūra, drėgmė, slėgis ir apšvietimo intensyvumas. Šiai kategorijai priskiriami barometrai, fotometrai ir termometrai. Deja, šios kategorijos jutikliai mobiliuosiuose įrenginiuose sutinkami retai, išskyrus fotometrą, kurį galima rasti daugumoje mobiliųjų įrenginių, nes jo pagalba galima valdyti prietaiso ekrano apšvietimo intensyvumą. Priešingai negu pozicijos ir judesio jutikliai, aplinkos jutikliai gražina tik vieną reikšmę, pvz.: temperatūra 0C.

3. Pozicijos jutikliai

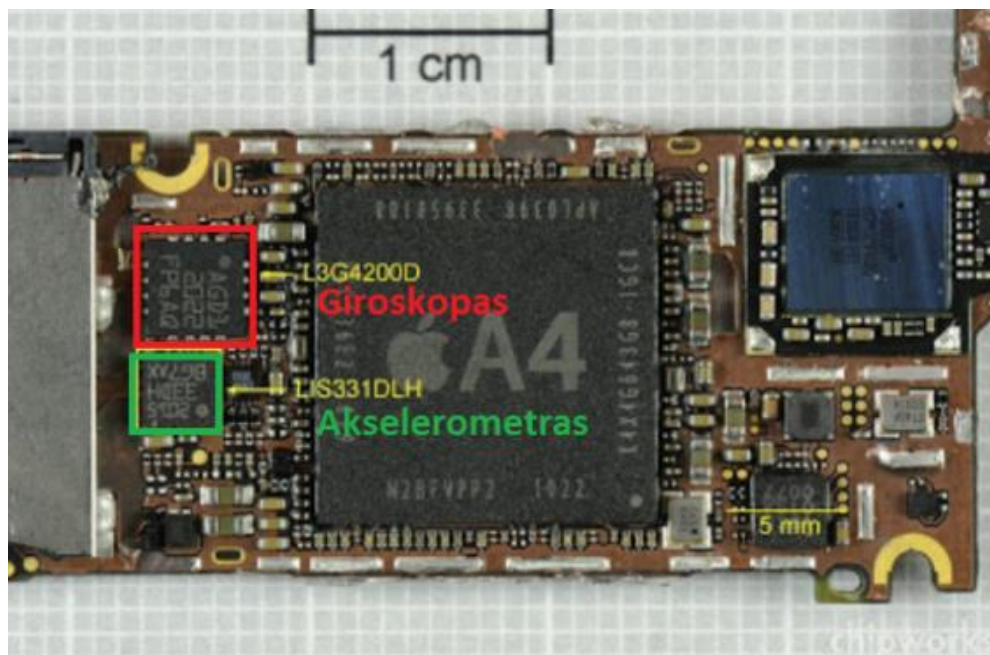
Šios kategorijos jutikliai atlieka svarbią funkciją – seka įrenginio poziciją globalios planetos koordinatų sistemos atžvilgiu. Šie jutikliai dažniausiai naudojami norint nustatyti įrenginio koordinates žemėlapyje arba, kitaip sakant, buvimo pasaulyje vietą. Pozicijos jutiklių kategorijai priskiriami orientacijos jutikliai ir magnetometrai. Reikėtų pažymėti, kad šios kategorijos jutikliams dažnai reikalinga pagalba iš judesio kategorijos jutiklių, pvz.: jeigu yra jungiami magnetometras ir akcelerometras į vientisą sistemą, galima nustatyti prietaiso vietą šiaurės ar pietų pusrutuliuose.

Taip pat jutikliai gali būti dvejų tipų: aparatiniai jutikliai ir programiniai jutikliai. Aparatiniai jutikliai yra materialūs prietaisai į įrenginį įstatyti prietaisai. Tokie jutikliai informaciją perduoda tiesiogiai matuojant specifinius aplinkos parametrus. Programiniai jutikliai yra nematerialūs, tai mažos operacinės sistemos programos. Tokie jutikliai informaciją pateikia, atlikę tam tikrus skaičiavimus su informacija gauta iš vieno ar daugiau aparatinių jutiklių.

Toliau pateikiama ir plačiai aprašoma informacija apie jutiklius, kurie yra sutinkami daugumoje mobiliųjų įrenginių. Tiesa, reikėtų paminėti, kad mobiliuosiuose įrenginiuose gali būti daugiau negu vienas tos pačios kategorijos jutiklis. Taip pat dažnu atveju ne visi mobilieji prietaisai gali turėti visus išvardintus jutiklius. Kiekviena mobiliojo įrenginio sistema yra sukomplektuojama taip, kad ji veiktų kaip galima našiau.

1.1.1 Akcelerometras

Šis prietaisas priskiriamas aparatinių jutiklių tipui. Mobiliuosiuose įrenginiuose tai yra mažas (žr. pav. 1.1) prietaisas, kurio veikimo principas paremtas tiesinio judėjimo išilgai trijų (x , y , z) statmenų ašių fiksavimu. Pagrindinė akcelerometro funkcija sekti ir matuoti įrenginio pagreitį išilgai minėtų ašių. Su šio jutiklio paramerais nesudėtinga nustatyti įrenginio orientaciją gravitacijos vektorius atžvilgiu. Judėjimo jutikliai sudarantys akcelerometrą taip pat gali būti naudojami aptikti žemės drebėjimus, prietaiso kritimą, mašinų avarijas ir kuriant dirbtines žmogaus galūnes medicinoje. Pagal akcelerometrų teikiamą informaciją, analizuojama kaip prietaisas juda erdvėje, nustatoma ar prietaisas juda aukštyn ar žemyn, kokia prietaiso padėtis erdvėje. Fizikiniu požiūriu akcelerometrai yra elektromechaniniai prietaisai matuojantys pagreitį. Tos jėgos gali būti pastovios (gravitacija) arba kintamos – įrenginio judėjimas ir vibracijos iš išorės. Iš pažiūros akcelerometras atrodo paprastas prietaisas, tačiau jį sudaro daug atskirų dalių, kurios gali veikti skirtingais būdais. Akcelerometro veikimo principas yra paremtas pjezoelektriniu efektu, duomenys gaunami iš elektros srovės, kuri yra generuojama veikiant mažus kristalus arba dirbtinius polimerus mechanine jėga [12], [19], [22].



pav. 1.1 Akselerometro ir giroskopo dydžių palyginimas

1.1.2 Giroskopas

Jutiklis priskiriamas aparatinių jutiklių tipui. Šis prietaisas labai panašus į akselerometrą, tačiau esminis skirtumas tarp šių dviejų jutiklių yra tas, kad giroskopas matuoja pasisukimo kampą laipsniais pagal kiekvieną ašį, o akselerometras matuoja jėgą veikiančią prietaisą x, y, z ašių atžvilgiu. Pagrindinė giroskopo funkcija yra sekti prietaiso pasisukimą visoms trimis ašims nuo prietaiso pozicijos fiksavimo pradžios (pradinio taško). Vien tik giroskopo generuojami duomenys nėra labai vertingi, tačiau sujungus giroskopo duomenis su kitų jutiklių duomenimis yra gaunama tiksli informacija apie momentinį prietaiso pasisukimo kampą lokaloje prietaiso koordinatinių sistemoje. Taip pat sujungiant giroskopo duomenis su kitų jutiklių duomenimis yra stabilizuojama ir patikslinama kitų jutiklių pateikiama informacija. Mobiliosiuose įrenginiuose esančio giroskopo veikimo principas yra paremtas Koriolio efektu. Koriolio efektas – matomas objektų nukrypimas nuo tiesaus judėjimo, stebint iš besisukančio atskaitos taško. Kai prietaisas yra sukamas Koriolio efektas generuoja vibracijos jėgą, kurios kryptis yra statmenas prieš tai buvusiai vibracijai, šių dviejų vibracijų amplitudė yra išmatuojama naudojant talpinius elektrodus, tokiu būdu galima nustatyti prietaiso kampinį pasisukimą [19].

1.1.3 Aplinkos temperatūros jutiklis

Šis jutiklis priskiriamas aparatinių jutiklių tipui. Pagrindinė jutiklio funkcija – matuoti temperatūrą ir jos pokyčius erdvėje esančioje aplink mobilųjį įrenginį. Jutikliai gali būti mechaniniai ir elektriniai. Elektrinių jutiklių veikimo principas paremtas jutiklio elektrinės varžos stiprumo kitimu. Jeigu temperatūra kyla (šyla), jutiklio varža mažėja ir atvirkščiai. Tačiau dėl netikslių matavimų šiuos jutiklius pradėta naudoti vidinei įrenginio temperatūrai stebėti [12], [19], [22].

1.1.4 Gravitacijos jutiklis

Šis jutiklis gali būti aparatinis montuojamas kartu su akcelerometru, giroskopu ar magnetometru. Taip pat šis jutiklis gali būti ir programinis, kai jutiklio duomenys gaunami iš vieno ar kelių aparatinių jutiklių duomenų. Pagrindinė šio jutiklio funkcija yra matuoti gravitacijos jėgą žemėje. Su šiuo jutikliu nustatoma mobiliojo įrenginio judėjimas. Aparatiniu gravitacijos jutikliu yra nustatoma prietaiso pasukimas, tam naudojama žemo dažnio filtro taikymo technika, tačiau dėl šio įrenginio prailgėja jutiklio atsako laikas. Dėl šių problemų dabar dažniausiai naudojama programinis gravitacijos jutiklis. Yra sukuriama sistemos programa veikianti pagal akcelerometro, giroskopo ar magnetometro generuojamus duomenis. Programinis gravitacijos jutiklis užtikrina greitesnį užklauso atsakymo greitį ir tikslesnius rezultatus [19], [25].

1.1.5 Aplinkos šviesos jutiklis (fotometras)

Šis jutiklis priskiriamas aparatinių jutiklių tipui. Pagrindinė jutiklio funkcija yra aptikti šviesą ir jos intensyvumą krentantį į jutiklį. Tokie jutikliai yra naudojami, norint pakeisti tam tikrus sistemos parametrus, kurie turėtų kisti pagal kintantį apšvietimą. Dažniausiai naudojama, norint pakeisti mobiliųjų telefonų ekrano apšvietimo intensyvumą atsižvelgiant į aplinkos apšvietimą. Šio jutiklio veikimo principas paremtas fotodiode veikimo principu. Tiesa tai yra specifinis fotodiode variantas, tačiau išlaiko pagrindinius savo principus. Šviesai jautrus paviršius priimančias šviesos srautą ir keičiantis šviesą į elektros įtampą. Aplinkos šviesos jutikliai atlieka funkciją panašią kaip žmogaus akys, tiesa, šviesos intensyvumo kitimo matavimą jutikliai fiksuoja tiksliau negu žmogaus akis [12], [19].

1.1.6 Magnetometras

Šis prietaisas yra priskiriamas aparatinių jutiklių tipui. Pagrindinė jutiklio funkcija yra matuoti žemės lauko magnetinę indukciją statmenai visoms trimis ašims. Pagrindinis magnetometro panaudojimas mobiliuosiuose įrenginiuose yra kompasos funkcijos. Pagal šio jutiklio duomenis galima nustatyti prietaiso orientaciją pasaulio šalių (šiaurė, rytai, pietūs, vakarai) atžvilgiu. Magnetometras mobiliuosiuose įrenginiuose veikia Hallo (Edwin Hall) efekto jutikliniu principu, gaminama įtampa, kuri yra proporcinga jėgai ir poliui, veikiančiam atitinkamą ašį. Įtampa yra pakeičiama į skaitmeninį signalą, kuriuo parodomas magnetinio lauko stipris. Šie prietaisai dažniausiai yra kombinuojami kartu su kitais mobiliųjų įrenginių jutikliais, magnetometrai sutinkami kartu su akcelerometrais, taip yra patikslinami magnetinio lauko matavimai, pasitelkiant akcelerometrą kaip pagalbinį jutiklį. Norint nustatyti prietaiso pasisukimo pasaulio atžvilgiu informaciją, magnetometro duomenys yra svarbiausi [19], [25].

1.1.7 Orientacijos jutiklis

Tai programinio tipo jutiklis, pasitelkus akcelerometro ir magnetometro duomenis yra grąžinama informacija apie įrenginio pasisukimą žemės atžvilgiu. Orientacijos jutiklis apdorojus aparatinių jutiklių duomenis teikia informaciją visų trijų ašių atžvilgiu:

- prietaiso pasisukimo kampą aplink z ašį – tai kampas tarp magnetinio lauko šiaurės ir prietaiso y ašies. Pvz.: jeigu prietaiso y koordinatė yra atsukta tiesiai į magnetinio lauko šiaurę, prietaiso pasisukimo kampas lygus nuliui, o jeigu prietaisas nusuktas magnetinio lauko pietus, tai prietaiso pasisukimas aplink z ašį yra 1800 (azimutas).
- prietaiso pasisukimo kampą aplink x ašį – šio kampo reikšmės gali kisti nuo –1800 iki +1800. Kampas nustatomas pagal kitų ašių pasisukimą. Kampo reikšmė teigiama kai teigiama z ašies reikšmė yra sukama link teigiamos y ašies, kampo reikšmė tampa neigiama, kai teigiama z ašies reikšmė yra sukama link neigiamos y ašies. (mėtymas)
- Prietaiso pasisukimo kampą aplink y ašį – šio kampo reikšmės gali kisti nuo -900 iki 900. Kaip ir x ašies kampo nustatyme, šis pasisukimo kampas nustatomas pagal duomenis iš x ir z ašių. Kampo reikšmė yra teigiama, kai teigiama z ašis sukama link teigiamos x ašies, kampo reikšmė tampa neigiama, kai teigiama z ašis sukama link neigiamos x ašies (virtimas), [12], [22].

1.1.8 Barometras

Barometras matuoja aplink prietaisą esantį oro slėgį, taip pat pagal šio jutiklio duomenis yra nustatoma apytikslis aukštis pasaulio atžvilgiu (virš jūros lygio). Pagrindinė barometro funkcija stebėti oro slėgio pokyčius. Naudojant šį jutiklį mobiliuosiuose įrenginiuose yra tiksliau nustatoma įrenginio buvimo vieta, naudojant GPS funkciją. Šio jutiklio veikimo principas labai panašus į aplinkos temperatūros jutiklio. Jutiklio varža keičiasi, kai keičiasi aplinkos oro slėgis, tokie signalai perduodami į analoginį-skaitmeninį keitiklį, kur signalas paverčiamas skaitmeniniu ir yra atvaizduojamos oro slėgio pokyčio reikšmės bėgant laikui [22].

1.1.9 Artumo jutiklis

Šis jutiklis priskiriamas aparatinių jutiklių tipui. Pagrindinė šio jutiklio funkcija yra nustatyti, kaip toli objektas yra nuo mobiliojo įrenginio, be jokio fizinio kontakto su objektu. Šis jutiklis dažniausiai įtaisomas įrenginio priekyje, norint išmatuoti atstumą tarp įrenginio ir žmogaus besinaudojančio įrenginiu. Vienas iš pagrindinių šio jutiklio panaudojimų yra mobiliuosiuose telefonuose, kad būtų nustatyta telefono pozicija skambučio metu. Artumu jutikliu pateikiami duomenys gali būti išreikšti centimetrais arba dviem būsenomis: „objektas arti“ arba „objektas toli“ abi šios būsenos taip pat apibrėžiamos laike, t.y. kiek laiko objektas buvo arti arba toli. Artumo jutikliai veikia skleisdami elektromagnetinį lauką arba skleidžia elektromagnetinės radiacijos spindulį (pvz.: infraraudonosios bangos), toliau stebima aplinkos pakitimai arba analizuojamas grįžęs signalas. Šių

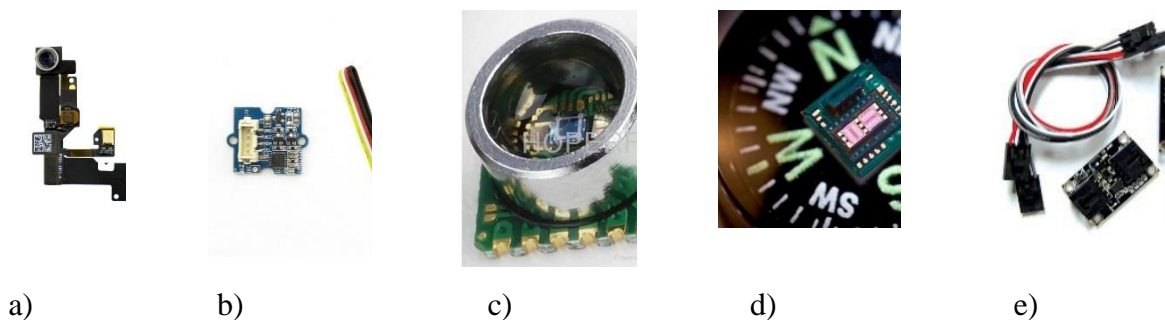
jutiklių patikimumas ir veikimo laikas yra didesnis negu kitų aparatinių jutiklių dėl to, kad nėra jokio fizinio kontakto su objektu.

1.1.10 Drėgmės jutiklis

Jutiklis priskiriamas aparatinių jutiklių tipui. Pagrindinė šio jutiklio funkcija yra stebėti oro esančio aplink įrenginį drėgmę. Šis jutiklis rezultatus atvaizduoja procentine išraiška. Drėgmės jutikliai veikia matuodami elektrinę varžą tarp jutiklyje esančios medžiagos ir aplinkoje esančios drėgmės. Deja, bet matuojant drėgmę varžos tipo jutikliai yra mažiau jautrūs, negu talpiniai jutikliai. Norint gauti tikslius duomenis, tokio tipo jutikliai reikalauja papildomų duomenų, tokiu atveju drėgmės jutiklio duomenys yra jungiami su aplinkos temperatūros jutiklio duomenimis, kad būtų užtikrinti tikslūs drėgmės matavimo rezultatai. Taip pat labai svarbu atkreipti dėmesį į tai, kokia medžiaga yra pasirinkta matuoti drėgmei, nes tai tiesiogiai daro įtaką matavimų tikslumui [12].

1.1.11 Pasisukimo krypties jutiklis

Šis jutiklis priskiriamas programinių jutiklių tipui. Pagrindinė šio jutiklio funkcija yra sekti prietaiso orientaciją erdvėje horizonto atžvilgiu, ar prietaisas yra pasuktas šonu, ar laikomas tiesiai, o galbūt prietaisas sukiojamas. Pagrindiniai šio jutiklio duomenų šaltiniai yra giroskopas ir akselerometras. Jutiklį duomenis aprūpina giroskopas, tačiau tai negarantuoja tikslų duomenų, norint sumažinti duomenų nuokrypį naudojama akselerometro duomenys. Kadangi tai yra programinis jutiklis jo atsakymo laikas priklauso nuo aparatinių jutiklių greičio [19][25].



pav.1.2 Mobilųjų įrenginių jutikliai; a) artumo, b) akselerometras, c) barometras, d) magnetometras, e) giroskopas

1.1.12 Vaizdo jutiklis

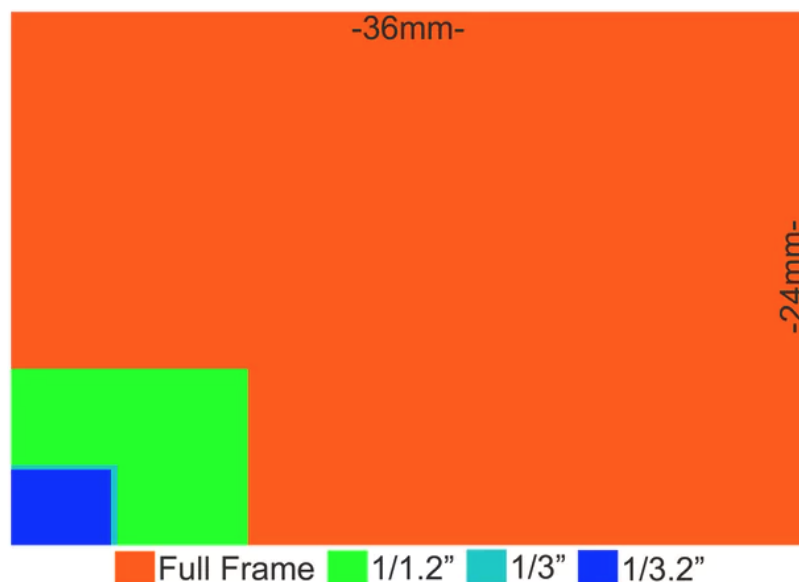
Vaizdo jutikliai ir jų teikiamos funkcijos mobiliuosiuose įrenginiuose šiomis dienomis yra vienas svarbiausių aspektų renkantis naują įrenginį. Socialinė medija vis dažniau tampa daugelio žmonių kasdienybe. Užuoat rašę tekstą žmonės dažniausiai pasidalina nuotrauka, kuri perduoda tą pačią informaciją, kurią galima būtų užrašyti tekstu, taip pat neretai kamera tampa žmogaus užrašų knygelė, užuoat užsirašius reikiamą informaciją, žmogus ją tiesiog nusifotografuoja ar nusifilmuoja.

Tuo tarpu mobiliųjų įrenginių kūrėjai nesnaudžia ir stengiasi pasiūlyti vartotojams įrenginį su naujomis vaizdo jutiklio funkcijomis; megapikselių kiekio didinimas, kameros objektyvo platumo didinimas, atviresnės diafragmos realizavimas, antro vaizdo jutiklio įdiegimas, dvigubų pikselių ir

fazinio autofokusavimo funkcijos, vaizdo stabilizavimas realiu laiku, vaizdo saugojimas raw formatu, ir 10 bitų vaizdo spaudimas.

Plačiausiai paplitę vaizdo jutikliai yra dviejų tipų: CMOS (angl. *Complementary Metal Oxide Semiconductor*) ir CCD (angl. *Charge-coupled device*). Šiuo metu mobiliuosiuose įrenginiuose yra realizuoti tik CMOS vaizdo jutikliai, nes CCD jutikliai naudoja daugiau energijos, kas trumpina mobiliojo įrenginio veikimo laiką, taip pat yra brangesni. CMOS jutikliai yra pikselių masyvas priimančias ir užfiksuojantis šviesą ir jos intensyvumą, toliau užfiksuota informacija apdorojama ir pateikiama vartotojui kaip nuotrauka ar nenutrūkstamas, nuoseklus vaizdas.

Vaizdo jutiklis dažniausiai sutinkamas su objektyvu, kuris valdo šviesos kiekio patekimą ant vaizdo jutiklio, todėl mobiliuosiuose įrenginiuose vaizdo jutiklis ir objektyvas yra vadinama kamera. Vis augant mobiliųjų įrenginių rinkai yra bandoma sukurti įrenginį, kuris bus patogus naudoti vartotojui, dėl to įrenginių dydis varijuoja. Siekiant optimizuoti įrenginių architektūrą yra naudojami maži (CMOS) jutikliai jų dydis: 4.54 x 3.42 mm, palyginimui jutikliai, kurie yra montuojami į pilno kadro fotoaparatus yra 36 x 24 mm dydžio. (žr. pav. 1.3)



pav. 1.3 Vaizdo jutiklių dydžių palyginimas [12]

Pagrindiniai jutiklio parametrai yra: užrakto greitis, jis nusako kiek laiko jutiklis fiksuoja šviesos srautą – matuojamas sekundėmis arba sekundės dalimis; jautrumas tai yra fiksuojamo vaizdo signalo stiprinimas, reikėtų pažymėti, kad kuo jautrumas didesnis, tuo vaizdo signale labiau pastebimi triukšmai; baltos spalvos balansas, vaizdo jutiklis turi galimybę pasirinkti kaip interpretuoti baltą spalvą, kadangi balta spalva skirtinguose apšvietimuose yra skirtinga, vartotojai gali pasirinkti baltos spalvos temperatūrą, matuojam kelvinais. Taip pat yra svarbu ir objektyvo techninės charakteristikos; matymo kampas, diafragmos reikšmė, optinių elementų skaičius. Fotografuojant visi šie parametrai yra saugomi nuotraukos *exif* faile, kuriame kaupiama visa meta informacija apie nuotrauką.

Atsižvelgiant į vartotojų poreikius gamintojai suteikia galimybę vartotojui kontroliuoti vaizdo jutiklio parametrus: jautrumą, užrakto greitį, fokusavimo režimą, baltos spalvos balansą (interpretatorių), spalvų profilį, diafragmos dydį. Reikėtų pažymėti, kad anksčiau tokios funkcijos buvo „paslėptos“ nuo vartotojų, tiesa ne visi ankstesni įrenginiai sugebėjo pasiūlyti tokias funkcijas. Pateikiama susisteminta mobiliųjų įrenginių lentelė. (žr. lentelė 1.1)

lentelė 1.1 Įrenginių jutiklių apibendrinamas

Jutiklis	Tipas	Trumpa informacija	Panaudos atvejai
Akselerometras	Aparatinis	Matuojama prietaiso pagreitį visomis ašimis	Prietaiso judėjimo nustatymas
Termometras	Aparatinis	Matuojama prietaiso vidinę ir lauko temperatūrą	Temperatūros pokyčio stebėjimas
Girokopas	Aparatinis	Matuojama prietaiso pasisukimo kampą pagal kiekvieną ašį	Įrenginio pasukimo aptikimas
Gravitacijos jutiklis	Aparatinis arba programinis	Matuojama prietaisą veikianti gravitacijos jėga pagal kiekvieną ašį	Prietaiso judėjimo nustatymas
Fotometras	Aparatinis	Matuojama šviesos srauto lygis	Pagal šią reikšmę yra galimybė parinkti tinkamą prietaiso ekrano apšvietimą
Magnetometras	Aparatinis	Matuojama atmosferos slėgio lygis	Atsiranda galimybė naudoti kompasą įrenginyje
Orientacijos jutiklis	Programinis	Matuojama prietaiso pasisukimo kampo laipsniai apie kiekvieną ašį	Įrenginio pozicijos nustatymas
Barometras	Aparatinis	Matuojamas aplink prietaisą esančio oro slėgis	Oro slėgio keitimosi stebėjimas
Artumo jutiklis	Aparatinis	Matuojamas atstumas tarp prietaiso (ekrano) ir objekto, centimetrų tikslumu	Dažniausiai naudojamas telefonuose, telefono pozicijos skambučio metu nustatymas
Drėgmės jutiklis	Aparatinis	Matuojama santykinė oro drėgmė procentine išraiška	Oro drėgmės stebėjimas
Pasisukimo krypties jutiklis	Programinis	Matuojama prietaiso orientacija erdvėje	Prietaiso judėjimo ir pasisukimo nustatymas
Vaizdo jutiklis	Aparatinis	Fiksuojamos nuotraukos, filmuota medžiaga	Nuotraukų fiksavimas, vaizdo įrašų filmavimas

1.2 Kompiuterinės regos įrankiai

Kompiuterinė rega – informacinių technologijų sritis, kurios pagalba kompiuteris gali atlikti paprastus sprendimus analizuodamas įvesties vaizdus. Šiam projektui realizuoti yra būtina naudoti kompiuterinės regos bibliotekas. Biblioteka skirta spręsti kompiuterinės regos uždavinius; metodų, funkcijų ir algoritmų rinkinys padedantis lengviau ir greičiau sukurti programinę įrangą ir atlikti reikiamus eksperimentus.

1.2.1 OpenCV

OpenCV – plačiausiai sutinkama kompiuterinės regos biblioteka, šioje bibliotekoje implementuota gausybė kompiuteriniai regai skirtų funkcijų, algoritmų, metodų. Dauguma šių funkcijų orientuotos į realaus laiko vaizdo apdorojimo programinės įrangos kūrimą. Ši biblioteka gali būti naudojama skirtingoms operacinėms sistemoms (*Windows, Linux, Mac OS, iOS* ir *Android*). Senesnės šios bibliotekos versijos buvo pritaikytos tik C++ programavimo aplinkai, tačiau naujausia *OpenCV* versija suteikia galimybę pasirinkti norimą programavimo kalbą: C++, C, *Java, Python*. Naudojant šios bibliotekas versiją C++ ar C programavimo kalboms, skaičiavimus galima atlikti, naudojant kelis kompiuterio procesoriaus branduolius. Biblioteka turi per 2500 optimizuotų algoritmų, skirtų vaizdų atpažinimui, veidų aptikimui, veidų atpažinimui, objektų identifikavimui, kameros judesio nustatymui, judančių objektų sekimui ir jų pozicijos nustatymui, žmonių veidų klasifikavimui ir kitiems kompiuterinės regos taikymams. Pagrindinis *OpenCV* bibliotekos tikslas – palengvinti programinės įrangos susijusios su kompiuterine rega kūrimą. Kompiuterinė rega labai dažnai siejama su mašininio mokymusi, dėl to *OpenCV* biblioteka taip pat turi gausų rinkinį metodų, skirtų mašininio mokymosi skaičiavimams (klasifikacijai ir rašto atpažinimui) atlikti.

1.2.2 SimpleCV

Tai paprastesnė *OpenCV* bibliotekos versija, skirta dar greičiau kurti programinę įrangą, skirtą kompiuterinės regos uždaviniams spręsti. *SimpleCV* yra orientuota darbui su vaizdu – statiniais vaizdais ir video sekomis. Kaip teigia patys autoriai, naudojant šią biblioteką, kompiuterinė programa ne tik gali analizuoti vaizdą, bet ir gali jį interpretuoti. Biblioteka sukurta, naudojant „*Python*“ programavimo kalbą, dėka šios kalbos paprastumo, ši biblioteka palengvina programuotojų darbą ir suteikia galimybę greitai integruoti biblioteką prie programinės įrangos kūrimo proceso. Biblioteka veikia *Windows, Mac* ir *Linux* operacinėse sistemose.

Šiame projekte yra tiriama mobiliųjų įrenginių papildomų duomenų, gautų iš jutiklių įtaka vaizdų atpažinimo tikslumui, todėl yra būtina išanalizuoti, kaip vyksta vaizdų atpažinimas ir kokie yra vaizdų atpažinimo metodai.

Vaizdų atpažinimas yra priskiriamas kompiuterinės regos, mašinės regos ir vaizdų apdorojimo uždavinių klasei. Vaizdų atpažinimo pagrindinis tikslas yra surasti atitinkamus objektus užduotuose paveiksluose. Ieškomi objektai, paveiksluose gali būti skirtingo dydžio ir pasukti tam tikru kampu pagal ašį, ne būtinai statmeną vaizdo plokštumai. Atpažinimo metodas susideda iš vieno ar kelių algoritmų, skirtų spręsti specifinį uždavinį ir pasirinkto metodo taikymo.

1.2.3 Šablonų atpažinimas

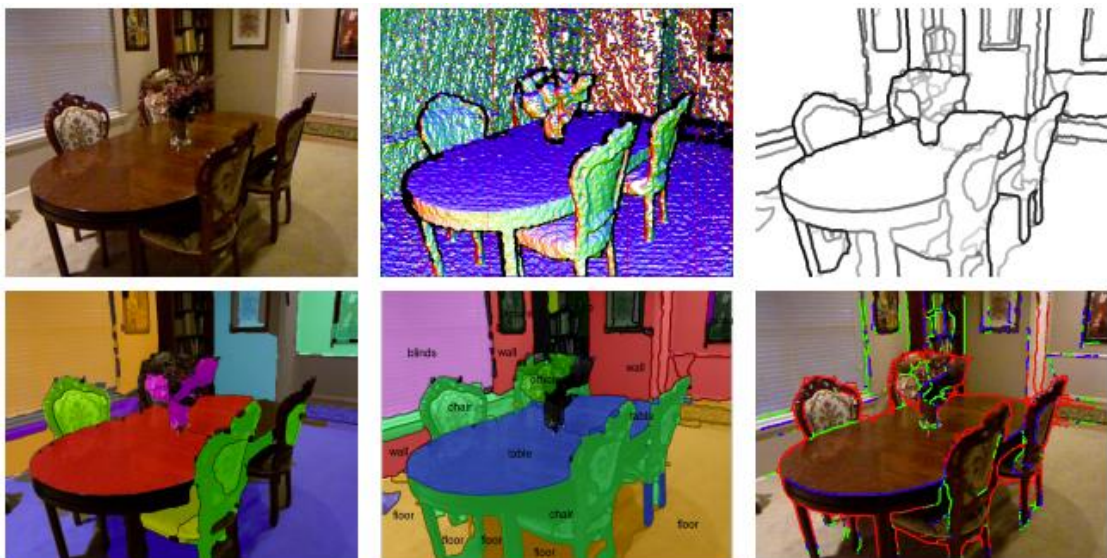
Šablonų atpažinimas – naudojamas atpažinti objekto vaizdą paveiksle, nustatant jų šablonų požymius. Šablonų požymiai yra panaši objekto forma ar kiti požymiai, kurie padeda susieti ir atpažinti objektą. Šablonų atpažinimas dažniausiai sutinkamas mašininio mokymosi aplinkoje. Naudojant šį

atpažinimo metodą minėtoje aplinkoje, yra pasitelkiamas dirbtinis neuroninis tinklas, kuris gali arba turi būti apmokytas su jau egzistuojančiais duomenimis, kurie yra išanalizuoti ir aprašyti. Taip pat šis metodas gali veikti kaip pats save apsimokantis iš turimų duomenų, tada atliekama objektų požymių analizė. Tai toks pat šablonų vaizdų atpažinimo metodas, kaip minėtas, tačiau jo specifika ir panaudojimas kiek kitokie. Pastarasis daugiau gali būti naudojamas duomenų bazių praplėtimams, duomenų gavimams ar tam pačiam mašininiam mokymuisi. Kaip atvaizde išskiriami šablonai pavaizduota pav. 1.4.

Šablonų vaizdų atpažinimo metodas numato galimą rezultatą visoms įėjimo reikšmių kombinacijoms, tai reiškia, kad kiekvienam šablono taškui, bandoma surasti atitikmenis. Metodas taip pat ieško tobulo įėjimo atitikmens esamoje šablonų duomenų bazėje. Šis metodas plačiai naudojamas žodžių apdorojime, sakinių ir žodžių sujungimo procese.

Algoritmai naudojami šablonų vaizdų atpažinimo metode [14]:

- Klasifikavimo algoritmai;
- Nuoseklaus ženklavimo algoritmai (angl. *Sequential labeling algorithm*);
- Regresijos algoritmai (angl. *Regression algorithm*);
- Bendrieji algoritmai šablono struktūros nuspėjimui (angl. *General algorithm for predicting arbitrarily structured*);
- Klasterizavimo algoritmai (angl. *Clustering algorithm*);
- Daugiagrafiškai sistemos mokymosi algoritmai (angl. *Multilinear subspace learning algorithm*);
- Bendrai apmokytas algoritmas (angl. *Ensemble learning algorithm*);
- Realių reikšmių rikiavimo algoritmai (angl. *Real valued sequential labeling algorithm*).



pav. 1.4 Šablonų aptikimo metodas [23]

Metodo taikymas:

- Šablonų vaizdų atpažinimo metodas naudojamas medicinoje padėti pastebėti žmogaus kūno sužalojimus;
- Naudojamas valstybės apsaugos sistemoje, atpažinti galimus priešo išsilaipinimus ar surasti potencialius taikinius;
- Naudojamas transporte, nustatyti automobilio markę ir atpažinti valstybinius automobilio numerius;
- Naudojamas autentifikavimo ar apsaugos sistemose, piršto anspaudu, veido šablonų atpažinimas;

1.2.4 Pikselių grupės aptikimo metodas

Šis metodas naudojamas surasti skirtingas sritis paveiksle. Metodo pagalba išskiriamos sritys, kurios turi skirtingas ypatybes – spalva ir šviesos intensyvumas paveiksle lyginant su kitomis paveikslo charakteristikomis. Įvesties paveiksle, gali būti, kad taškai yra tokie patys, skiriasi tik jų ypatybės, būtent tą ir aptinka pikselių grupės aptikimo metodas. Šis metodas yra klasifikuojamas į dvi rūšis:

- Diferencialinis metodas;
- Vietinių ekstremumų metodas.

Diferencialinis metodas remiasi/priklauso funkcijos išvestine, kuri priklauso nuo pikselių grupės pozicijos paveiksle. Tuo tarpu vietinių ekstremumų metodas remiasi/priklauso nuo pikselių grupės maksimalios ir minimalios funkcijos reikšmių [23].

Pikselių grupės aptikimo metodą naudoja šiuos algoritmus:

- Afinisis adaptuotas diferencialinis sričių detektorius;
- *Linderbergo* nespaltotų pikselių grupių aptikimas grįstas takoskyros algoritmu.

Šis metodas labai panašus į kraštinių/kontūrų bei kampų aptikimo metodą, tačiau skiriasi tuo, kad saugo informacija apie tiriamąją paveikslo sritį. Pikselių grupės aptikimo metodo pagalba visa informacija apie tirtas sritis yra perduodama tolimesniam vaizdų atpažinimo procesui. Srityse pateikta informacija apdorojama ir iš jos galima nustatyti objekto ar objekto dalių poziciją paveiksle. Šis metodas plačiai naudojamas objektų sekime ir objektų atpažinimo programinėje įrangoje.

1.2.5 Požymių aptikimo metodas

Požymių aptikimo metodo pagrindinis tikslas surasti/aptikti svarbius/charakteringus paveiksle esančių objektų taškus, pagal kuriuos būtų galima atpažinti objektą. Metodas apdoroja paveikslą skanuodamas kiekvieną pikselį ir saugodamas svarbius objektų taškus duomenų bazėje. Išsaugojus charakteringus/ svarbius objektų atpažinimo taškus, atliekama analizė, kurioje ieškoma ar paveiksle yra ieškomas objektas. Deja, skanuojant kiekvieną pikselį užtrunkama nemažai laiko, o tai sąlygoja prastesnius metodo greitaveikos rezultatus, tačiau galima apriboti metodo veikimo laiką, dėl to paveiksle aptinkama ne visi objektų charakteringi taškai, o tik dalis jų [20].

Kiekvienas paveiksle esantis objektas turi savo charakteristikas, šiame metode jos apibrėžiamos keturios:

- Viršūnės;
- Kampiniai taškai;
- Charakteringi taškai arba pikselių grupės;
- Briaunos.

Viršūnės – tiriamas vaizdas paverčiamas nespalvotu, išskiriama objekto simetrijos ašis, kurios galai ir yra ieškomos objekto viršūnės. Tirti objekto charakteringas viršūnes palyginus su kitomis charakteristikomis yra sudėtinga, dėl dažnos požymių metodo paklaidos. Objekto viršūnių charakteristikos dažnai naudojamos atpažinti objektams, kurie yra simetriški, važiuojamoji kelio dalis, simetriški namai ir kt.

Kampiniai taškai – ne kas kita, kaip objekto kraštinių/briaunų susikirtimo vietas, dar kitaip vadinami pažymimaisiais taškais (angl. *key-points*, *points of interest*). Prieš ieškant kampinių taškų yra būtina surasti objekto briaunas, toliau stebint briaunų pasisukimo kampo kitimą yra nustatomi objekto kampiniai taškai.



pav. 1.5 Kampinių taškų išskyrimas [26]

Charakteringi taškai arba pikselių grupės – apibrėžia objekto papildomą informaciją, lyginant su kampiniais taškais. Kampiniai taškai yra tik taškai, o charakteringi taškai apibrėžia objekto struktūrą, spalvą ir kitas charakteristikas. Taip pat naudojant charakteringus taškus norint atpažinti objektą yra pasiekiami tikslesni rezultatai, negu naudojant tik kampinių taškų detektorius. Nes kartais kampinių taškų detektorius gali klaidingai interpretuoti kampą arba išvis jo neinterpretuoti.

Briaunos – linijos, kurios išskiria objekto formą, atskiria objektą, nuo kitų paveiksle esančių objektų. Briaunos griežtai apibrėžia objekto formą, pagal kurią, galima nustatyti objekto kampinius charakteringus taškus, taip pat jeigu reikia galima išskirti objekto viršūnes. Dažniausiai briaunos įveda objekto parametrus, kurie yra saugomi objektų duomenų bazėje, kad būtų galima atlikti tolimesnius vaizdų analizės tyrimus.

1.2.6 Metodų analizės apibendrinimas

Norint pasiekti kokybiškus ir tikslius rezultatus yra būtina naudoti visus analizuotus metodus. Tačiau galima pažymėti, kad naudojant visus metodus, skaičiavimams atlikt bus reikalinga daugiau kompiuterio resursų. Norint paspartinti šį procesą reikia naudoti mažiau metodų, o tai sąlygoja mažesnę atpažinimo tikslumą, visgi tokiu atveju yra tikslinga pasirinkti požymių aptikimo ir išgavimo metodą, kurio pagalba objekto atpažinimui yra reikalinga ne visas paveikslas, o tik dalis jo. Toliau aprašomi vaizdų atpažinimo algoritmai, analizuojama jų veikimo principai.

1.3 Charakteringų taškų aptikimas ir aprašymas

Norint kompiuteryje atpažinti objektus, kurie paprastai yra trimačiai (3D) yra susiduriama su problema, nes užfiksuotoje nuotraukoje objektai tampa plokšti, dvimačiai (2D). Sprendžiant tokį atpažinimo uždavinį trūksta informacijos apie bandomą atpažinti objektą. Dėl šios priežasties tampa sunku greitai ir tiksliai vykdyti objekto atpažinimą be papildomų duomenų esančių atvaizde ir vaizdo jutiklio parodymų. Tokiu atveju reikia įvertinti objektų skaičių nuotraukoje, pasisukimo kampą, apšvietimą ir kitus parametrus. Norint išspręsti šią problemą yra pasiūlyti keli sprendimo būdai, erdvę keisti į 2,5D arba aprašyti taškais. Pirmasis sprendimas yra pasiūlytas Britų mokslininko D.Marr. Tačiau šis sprendimas yra tik teorinis, kadangi jį sudėtinga pritaikyti objektų atpažinimui kompiuteryje. Antras, kur kas efektyvesnis sprendimas yra atvaizduoti objekto charakteristikas taškais, tam yra sukurta daug algoritmų, kurie sudaryti iš detektorių ir aprašų apie charakteringus objekto taškus. Pvz.: SURF, SIFT, FAST, ORB, KAZE, BRISK, MSER, FREAK, BRIEF, HOG ir kt.

Vaizdų atpažinime yra išskiriami pagrindiniai etapai: taškų suradimas; taškų aprašymas; taškų palyginimas. Taškų suradimui naudojami algoritmai, kurie vadinami detektoriais, pagrindinis jų tikslas surasti charakteringus taškus užduotam vaizdui, detektorius gali būti kaip vaizdų atpažinimo algoritmo sudedamoji dalis, taip pat kaip vienas metodas, kurio rezultatai bus panaudoti tolimesniems skaičiavimams. Taškų aprašymui naudojami algoritmai vadinami aprašais. Aprašai atlieka tolimesnius skaičiavimus su taškais gautais iš detektorių, aprašai aprašo charakteringuosius taškus ir jų savybes, saugo jų koordinates. Aptikus ir aprašius charakteringuosius taškus galima juos lyginti su tiriamuoju atvaizdu. Kitaip tariant tiriamojo vaizdo aprašas lyginamas su šablono aprašu. Taškų palyginimas yra galutinis vaizdų atpažinimo žingsnis, kuriuo yra tikrinami ar taškai egzistuoja dvejuose atvaizduose toliau suskaičiuojama kiek tokių taškų yra, taip pat yra pateikiamas rezultatas su taškų, kurie yra tokie patys skaičiumi.

1.3.1 Detektoriai ir aprašai

Šioje dalyje plačiau aprašomi vaizdų atpažinimo detektoriai ir aprašai, gilinamasi į jų veikimo principus, išskiriamos stipriosios jų ypatybės, pateikiami panaudos atvejai. Kai kurie aprašai turi tokį patį pavadinimą kaip ir detektoriai, todėl šiame poskyryje jie gali būti vadinami detektoriais ir aprašo deriniu arba vaizdų atpažinimo algoritmu.

1.3.2 AGAST

Plačiai sutinkamas kompiuterinės regos charakteringų taškų/požymių detektorius. AGAST (angl. *Adaptive and Generic Accelerated Segment Test*) [3] veikimas pagrįstas kampų aptikimo charakteristikų išskyrimu. Kampų aptikimas kompiuterinės regos uždaviniuose apibūdinamas, kaip dviejų dimensijų aptikimas, dėl šios priežasties aptikimo algoritmai yra greiti, o juos naudojant sudaromi pasirinkimų medžiai leidžia greitai identifikuoti aptiktas charakteristikas. Reikėtų pažymėti, kad kampų aptikimo algoritmai yra labai jautrūs vaizdo pasisukimams, mastelio kitimams ir kitoms transformacijoms. AGAST algoritmo struktūra yra paremta paspartinto segmentų testo (AST angl. *Accelerated Segment Test*) veikimu, kuris yra papildytas prisitaikančiomis prie vaizdo kitimo funkcijomis. Šiuo atveju prisitaikymas prie vaizdo kitimo yra realizuojamas sukuriant du pasirinkimų medžius ir juos sujungiant. Vienas pasirinkimų medis saugo charakteringų taškų informaciją, kitas – pikselių informaciją, o juos sujungus realizuojamas detektorius AGAST.

1.3.3 KAZE

Tai algoritmas, kuriame yra įdiegtas tiek detektorius, tiek aprašas skirti aptikti ir aprašyti netiesinio dydžio erdves iš nuotraukų. *KAZE* algoritmą sudaro trys pagrindiniai žingsniai:

1. Netiesiško dydžio erdvės sudarymas;
2. Charakteristikų aptikimas;
3. Charakteristikų aprašymas.

Netiesiško dydžio erdvės sudarymo metodas yra panašus į *SIFT* metodą, mastelis yra išskaidomas į logaritminius žingsnius surūšiuotus į O oktavas ir S lygmenis. *KAZE* algoritmas kurdamas netiesiško dydžio erdvę nemažina skiriamosios gebos kiekvienoje naujoje oktavoje, kaip *SIFT* algoritmas. Kiekviena nauja oktava yra identifikuojama oktavų kintamuoju oktavų indeksu – o , savo ruožtu lygmenys kintamuoju – s . Oktavų ir lygmenų indeksai yra priskiriami sudaromai erdvei pagal formulę:

$$\sigma_i(o, s) = \sigma_0 2^{o+s/S}, \quad o \in [0 \dots O - 1], \quad s \in [0 \dots S - 1], \quad i \in [0 \dots N]$$

Čia σ_0 yra bazinis mastelio lygmuo ir N paveikslų skaičius. Toliau vykdoma duomenų transformacija iš σ_0 pikselių į laiko vienetus. Minėta transformaciją yra būtina atlikti dėl to, kad netiesiškas reikšmių sklaidos filtras (angl. *diffusion filtering*) yra aprašomas laiko atžvilgiu. *KAZE* algoritmas sudarydamas netiesiško dydžio erdvę transformuoja pikselius į laiko vienetus, o toliau priskiria juos mastelio erdvei.



pav. 1.6 *Gauso* ir netiesinio suliejimo skirtumas [5]

Naudojant netiesišką reikšmių sklaidos filtrą duotosios nuotraukos objektų briaunos išlieka ryškios ir aštrios. Tačiau jeigu yra naudojamas tiesiškas reikšmių sklaidos filtras nuotraukos pikselių reikšmės yra suliejamos (angl. *blur*), dėl to tampa sudėtinga sudaryti paveikslo erdvę ir ją užrašyti taškais.

Charakteringų taškų radimas realizuojamas skaičiuojant dydžio atžvilgiu normalizuotą Hesės (angl. *Hessian*) matricos determinantą. Skirtingų dydžių mastelių taškų požymių aptikimui reikalinga normalizuoti diferencijuotus operandus (angl. *operators*) mastelio atžvilgiu, nes bendru atveju erdvinių išvestinių atžvilgiu amplitudė mažėja.

$$L_{Hessian} = \sigma^2 (L_{xx}L_{yy} - L_{xy}^2)$$

Čia (L_{xx} , L_{yy}) yra antros eilės horizontalios ir vertikalios išvestinės ir L_{xy} – antros eilės kryžminė išvestinė. Duotas paveikslas transformuojamas į netiesinio dydžio erdvę L_i ir analizuojami detektoriaus duomenys taikant skirtingus mastelio dydžio parametrus. Ieškoma didžiausia imties (pikselių grupės) reikšmė dydžio ir erdvės atžvilgiu. Kiekviena didžiausia pikselių grupės reikšmė yra ieškoma sugrupavus gretimus pikselius į paieškos langą. Kad skaičiavimai būtų greitesni iš pradžių yra analizuojamos pikselių grupės (langai) 3×3 pikselių dydžio, tokiu būdu galima greičiau identifikuoti vietas, kuriose nėra galimybės surasti charakteringą tašką. Galiausiai charakteringas taškas yra priskiriamas jeigu paieškos lango pirmoji ir antroji išvestinės skiriasi vienu pikseliu.

Charakteringų taškų aprašymas. Iš pradžių yra nustatoma dominuojanti paveikslo orientacija (kryptis). Aplink detektoriaus surastus taškus suformuojamas 6 pikselių spindulio dydžio apskritas plotas ir slenkamas 1 pikselio žingsniu. Kiekvienam iš plotų yra suskaičiuojamos pirmos eilės išvestinės L_x ir L_y ir joms priskiriamas svorio koeficientas. Toliau išvestinių reikšmės pateikiamos kaip taškai vektorių erdvėje, o orientacija nustatoma sudedant išvestinių reikšmes stumiant paieškos apskritimą kampu $\pi/3$, taip suradus ilgiausią vektorių yra priskiriama orientacija. Surastiems charakteringiems taškams iš pirmos eilės L_x ir L_y išvestinių yra sudaromas keturkampis 24×24

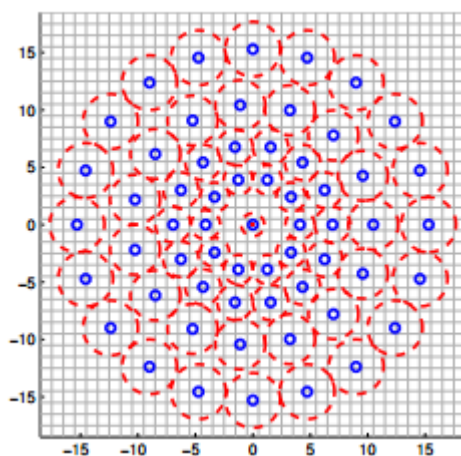
pikselių dydžio tinklelis. Toliau tinklelis yra skaidomas į 4x4 dydžio vidinius regionus, kurių dydis yra 9x9 pikseliai, dydis parenkamas didesnis su 2 pikselių persidengimu. Kiekvieno vidinio-regiono išvestinės rezultatas yra įvertinamas *Gauso* koeficientu ir sudedami į aprašo vektorių $dv = (Lx, Ly, |Lx|, |Ly|)$. Toliau kiekvienas vidinis-regionas yra įvertinamas *Gauso* reikšmėmis ir priskiriamas charakteringojo taško centrui. Toliau pagal dominuojančią orientaciją yra pasukami visi tinklelio elementai. Galiausiai aprašo vektorius normalizuojamas į vienetinį vektorių, kad orientacija išliktų atspari kontrasto kitimui [5].

1.3.4 BRISK

BRISK (angl. *Binary Robust Invariant Scalable keypoints*) - dvejetainis charakteringųjų taškų išskyrimo detektorius ir aprašas. Kaip detektorių *BRISK* galima aprašyti kartu su *SURF* ar *SIFT* aprašais. Algoritmas pasižymi greitu veikimu dėl nesudėtingų skaičiavimų, taip pat nėra skaičiuojamas gradientas.

BRISK algoritmas išskirstomas į kelis etapus:

1. Charakteringo taško išskyrimas į dvejetainę eilutę. Tai paprasčiausiai intensyviausių pikselių paieška. Taškai surandami pagal modelį pavaizduotą skaičius paveiksle. Taškai ieškomi naudojant apskritimo formos paieškos langą, mėlyni taškai nurodo paieškos vietą arba intensyviausio pikselio vietą, o raudoni apskritimai normalizuoja/glotnina galimą intensyvumo šuolį.



pav. 1.7 *BRISK* charakteringų taškų paieškos lango reikšmių seka [15]

2. Charakteringi taškai suskirstomi į poras, poros yra dviejų tipų: ilgosios ir trumposios, kad būtų išvengta persidengiančių porų išskyrimo yra įvedamos slenkstinės ribos. Tarp ilgųjų porų yra suskaičiuojamas gradientas, kurio pagalba yra nustatoma dominuojanti paveikslo orientacija. Pateikiama gradiento skaičiavimo formulė.

$$g(p_i, p_j) = (p_j - p_i) \cdot \frac{I(p_j, \sigma_j) - I(p_i, \sigma_i)}{\|p_j - p_i\|^2}$$

Kur g – gradientas, p_i, p_j – pikseliai, σ_j – slenkstinė riba.

3. Sudaromas aprašas, priskiriama dominuojanti paveikslo orientacija, lyginamos trumposios poros:

$$b = \begin{cases} 1, & \text{jeigu } I(p_j^\alpha, \sigma_j) > I(p_i^\alpha, \sigma_i) \\ 0, & \text{kitu atveju} \end{cases}$$

Trumpųjų porų palyginimui naudojamas Hamming atstumo skaičiavimo metodas, kuris skaičiuoja minimalų alternatyvių taškų porų skaičių. Taip pat šis metodas skaičiuoja minimalų klaidų skaičių, jeigu klaidos yra atsiradę transformuojant eilutę taškų aptikimo etape.

Kadangi *BRISK* yra dvejetainis charakteringų taškų detektorius ir aprašas pasižymintis greitu veikimu, algoritmą galima naudoti realaus laiko kompiuterinės regos uždaviniams spręsti [15].

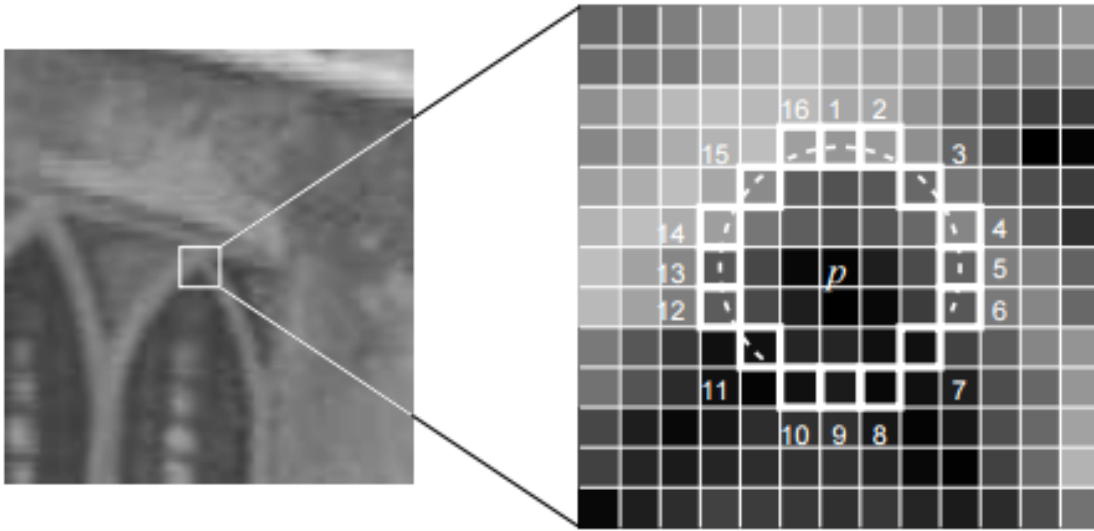
1.3.5 FAST

FAST (angl. *Features from Accelerated Segment Test*) – charakteringųjų taškų pagal kampus detektorius. Pagrindiniai šio detektoriaus privalumas yra greitaveika, palyginus su iki tol greitais detektoriais *SUSAN* ir *HARRIS*, *FAST* veikia greičiau (žr. pav. 1.8).

Detector	FAST	SUSAN	Harris
Time (ms)	2.6ms	11.8ms	44ms

pav. 1.8 detektorių palyginimas [4]

FAST detektorius charakteringų taškų ieško naudodamas 16 pikselių dydžio apskritimą, kuris dar gali būti apibūdinamas kaip požymių vektorius. Toliau vykdomas taškų aptikimas, paieškos apskritimas pasirinkęs nuotraukos dalį ją analizuoja. Įvedama charakteringo taško koeficiento riba, kuri padeda nuspręsti ar taškas charakteringas ar ne. Nustatoma centrinio pikselio intensyvumo reikšmė, dažniausiai procentinė išraiška. Aplink centrinį pikselį yra indeksuojami paieškos apskritimo pikseliai, kurie turi savo intensyvumo vertę, kuri gali būti tiek mažesnė, tiek didesnė negu centrinio pikselio. Siekiant greičiau atlikti skaičiavimus, visų pirma yra patikrinami keturių krypčių paieškos apskritimo pikseliai ties indeksais 1, 5, 9, 15, jeigu visi iš minėtų pikselių intensyvumo reikšmės yra mažesnės, negu charakteringo taško riba, tai nėra charakteringas taškas, tačiau, jeigu bent trys pikseliai yra mažesni už numatytą ribą, tada yra tikrinami visi 16 pikselių. Jeigu bent 12 pikselių atitinka numatytą ribą, tada taškas yra laikomas charakteringu. Taip pat toliau vykdoma paieška visame atvaizde.



pav. 1.9 FAST detektoriaus paieškos apskritimas su paieškos pikselių numeravimu

Kadangi FAST detektorius aptinka konkrečią reikšmę nuotraukoje, jis yra labai jautrus pikselių transformacijoms: nuotraukų pasisukimui, mastelio pakitimui, judesio sukeltam vaizdo suliejimui ir kitiems trukdžiams. Literatūroje rašoma, kad norint pagreitinti detektoriaus veikimo greitį, galima sumažinti paieškos lango apskritimo dydį iki 12 pikselių, tačiau tokiu atveju atsiranda labai daug charakteringų taškų, kurie ne visada yra patikimi.

1.3.6 BRIEF

BRIEF (angl. *Binary Robust Independent Elementary Features*) aprašas. Tai aprašas, kuriam yra būtina išskirti fiksuotą atminties dydį, kurio pagalba yra išskiriami aprašo branduoliai. Branduoliai naudojami nuotraukos dalių glotninimui dar prieš atliekant intensyvumų palyginimą. Aprašas yra skirstomas į 4 etapus:

- Sukuriamas aprašui skirtas langas, kuris visada yra centruotas charakteringo taško atžvilgiu;
- Aprašo langas apdorojamas *Gauso* filtru (žr. 14 formulę):

$$f(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right);$$

- Aprašo lange vyksta dviejų pikselių intensyvumo palyginimas, tikrinami visi pikseliai su centru;
- Sukuriama dvejetainė seka, pagal kurią yra atliekami pikselių intensyvumų palyginimai.

Sekos ilgis gali būti 128, 256 ar 512 bitų dydžio. Sekos sudarymui naudojama formulė:

$$f_{n_d}(p) := \sum_{1 \leq i \leq n_d} 2^{i-1} \tau(p; x_i, y_i);$$

Gautoji seka ir yra BRIEF aprašas. Kadangi aprašinėjant yra lyginami pikseliai, šis aprašas yra jautrus nuotraukos transformacijoms: pasisukimams, mastelio keitimui, apšvietimo kitimui. Tačiau atsparus judesio sukeltam susiliejimui. BRIEF aprašas skaičiuojant ir lyginant dviejų tarpusavio pikselių

intensyvumą, ilgį tarp taškų skaičiavimams naudoja Hemmingo ir Euklido atstumų skaičiavimų algoritmus. Sukildo atstumo skaičiavimo formulė:

$$d(a, b) = d(b, a) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_n - b_n)^2}.$$

Ši formulė gali būti naudojama tiek vienmatedei, tiek dvimatedei, tiek trimatedei erdvėms, jeigu naudojama dvimatede erdvė, formulėje atsiras koordinačių indeksai: $a = (a_1, a_2)$ ir $b = (b_1, b_2)$, žiūrėti 17 formulę. Ir taip toliau.

$$d(a, b) = d(b, a) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}.$$

Toks skaičiavimo algoritmas yra skirtas skaičiuoti atstumui tarp dviejų taškų.

Tuo tarpu Hemmingo algoritmas skirtas suskaičiuoti skirtumą tarp dviejų skaičių sekų. Pagrindinis algoritmo tikslas surasti minimalų klaidų skaičių, kai aprašas bando aptikti charakteringus taškus keliaujant iš vienos eilutės į kitą [4].

1.3.7 ORB

ORB – detektorius ir aprašas, veikimo principas yra pagrįstas detektoriaus *FAST* ir aprašai *BRIEF* veikimu. Pagrindinis skirtumas tarp šių algoritmų yra tai, kad *ORB* yra atsparus nuotraukų transformacijoms, pasisukimams, mastelio kitimui ir kt.

Mokslinėje literatūroje yra rašoma, kad *ORB* algoritmas yra alternatyva *SIFT* ir *SURF* algoritmams, kurie savo ruožtu jau yra patentuoti ir naudojami komerciniams tikslams. Taip pat yra atlikta mokslinių tyrimų, kad *ORB* našumo atžvilgiu lenkia *SURF* ir *SIFT* algoritmus [21].

1.3.8 MSER

MSER (angl. *Maximally Stable Extremal Regions*) – detektorius, kurio pagalba yra aptinkama charakteringi nuotraukos regionai. Algoritmas priskiriamas dėmių aptikimo algoritmams. *MSER* nuotraukoje aptinka regionus, kurie yra panašūs ir atitinka charakteringus kriterijų nustatytas ribas.

MSER detektoriaus specifikacijos:

- Nuotraukos paverčiamos į nespaltotas. Visi pikseliai nepatenkantys į aibę virš charakteringų kriterijų ribos yra balti, o patenkantys juodi.
- Nuotrauka laikoma charakteringa dėme iš pradžių visas algoritmo atsakas yra juoda spalva, tačiau pagal nustatytą kriterijų ribą, yra nuskaitoma vis daugiau regionų, kurie pamažu priskiriami kaip charakteringi. Iš pradžių surandamas intensyvumo minimumas, vėliau po detalesnio aptikimo dėmės ima didėti.
- Didėdamos baltos dėmės jungiasi viena su kita, kol visas paveikslas tampa baltas. Tuomet charakteringų regionų paieška baigta. Visų susijungusių baltų dėmių visuma yra regionų ekstremumų reikšmių seka.

Aprašyti *MSER* regionai gali turėti kelias būsenas: maksimaliai stabilūs regionai, per dideli regionai, per maži regionai, regionai nesuderinami. Norint nuspręsti ar regionas yra charakteringas yra surandamas mažiausias regionas įeinantis numatytų kriterijų ribą ir yra maksimaliai stabilios būsenos, toliau visi regionai, kurių pikselių intensyvumas yra didesnis už maksimaliai stabilaus regiono yra laikomi charakteringais [11], [18].

1.3.9 SIFT

SIFT (angl. *Scale Invariant Feature Transform*) – algoritmą sudaro charakteringų taškų detektorius ir jų aprašas. Vienas plačiausiai naudojamų vaizdų atpažinimo algoritmų, dėl savo greičio, atsparumo nuotraukų mastelio pokyčiams, išskiriamų charakteringų taškų kokybės, nepriklausomumo nuo vaizdo pasisukimo taip pat šiam algoritmui labai mažai įtakos turi apšvietimo pasikeitimas ir atpažinimas nepriklauso nuo stebėjimo taško. Kaip ir dauguma algoritmų *SIFT* sudaromas iš 4 pagrindinių žingsnių:

1. Paveikslo mastelio erdvės ekstremumų paieška;
2. Charakteringų taškų vietos išskyrimas;
3. Charakteringų taškų orientacijos nustatymas;
4. Charakteringų taškų aprašymas.

Toliau aptariamos kiekvieno iš žingsnių ypatybės.

Mastelio-erdvės ekstremumų paieška (angl. *Scale-space extrema detection*) – tai žingsnis, kurį atlieka SIFT detektoriaus dalis, užduotai nuotraukai ieškomi charakteringi taškai. Pagrindinis šio žingsnio tikslas yra nustatyti nuotraukos vietas ir dydį, kurie gali būti pakartotinai priskiriami tam pačiam objektui ar scenai žiūrint iš skirtingų stebėjimo taškų. Norint atlikti tokius skaičiavimus yra naudojama tolydžios Gauso funkcijos, Gauso branduolys (žr. formulę):

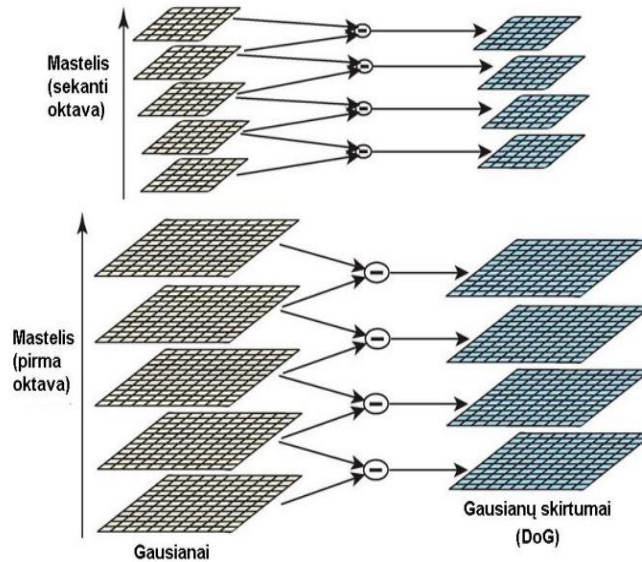
$$L(x, y, \sigma) = G(x, y, \sigma) \cdot I(x, y), G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}.$$

Čia L – mastelio erdvės funkcijos reikšmė pasirinktame taške, (x, y) – tiriamojo taško koordinatės, σ – Gauso glotninimo spindulys, G – Gauso funkcijos branduolys, I – vaizdo funkcija, \cdot – sąukos operacija. Tačiau ši funkcija ne visada yra efektyvi. Kur kas plačiau paplitusi mastelio erdvės ekstremumų paieškos funkcija yra Gauso funkcijų skirtumas (angl. *Difference of Gaussians*, *DoG*), ši funkcija yra efektyvesnė, kadangi skaičiuojant skirtumą galima naudoti skirtingus glotninimo spindulio koeficientus k .

$$D(x, y, \sigma) = G(x, y, k\sigma) - G(x, y, \sigma) \cdot I(x, y) = L(x, y, |k\sigma) - L(x, y, \sigma).$$

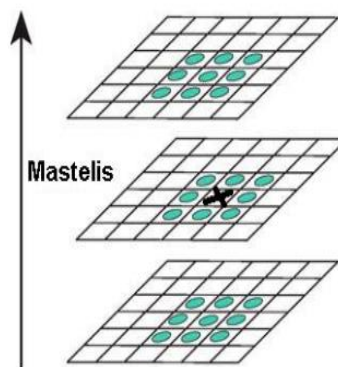
Čia k – glotninimo spindulio koeficientais, D – Gauso skirtumų funkcija, L – Laplaso funkcija. Skaičiuojant Gauso mastelio funkciją yra būtina ją normalizuoti naudojant Laplaso funkciją, kuri yra naudojama su σ^2 standartinio nuokrypio koeficientu. Toliau yra sudaroma Gauso piramidė, iš kurios yra generuojami skirtingo dydžio vaizdai. Visa sudarytos piramidės kintama erdvė suskaidoma į

atskiras dalis, kurios yra vadinamos oktavomis. Esminis oktavų bruožas, kad oktavos dydis, mažėjant atvaizdo dydžiui didės dvigubai. Atliekant charakteringų taškų paiešką yra skanuojamos suformuotos oktavos, pereinant nuo vienos oktavos į kitą, tačiau pereinant yra atliekamas duomenų pergrupavimas (žr. pav. 1.10), kurio rezultatas yra sumažėjęs duomenų kiekis piramidėje.



pav. 1.10 Duomenų sutraukimas *SIFT* detektoriuje [6]

Pagal glotninimo funkcijos žingsnį σ yra išrenkama N Gauso funkcijos reikšmių iš sudarytos piramidės. Kiekvienas Gauso skirtumų funkcijos atvaizdo taškas yra lyginamas su jo kaimynystėje esančiais aštuoniais taškais. Toliau visi taškai yra lyginami su kito mastelio piramidėse esančiais taškais. Taškas laikomas charakteringu jeigu jis priklauso Gauso skirtumų funkcijos ekstremumui. Charakteringo taško radimas iliustruojamas pav. 1.11, charakteringas taškas patikrinus jį tarp skirtingų mastelių yra žymimas (X).



pav. 1.11 Charakteringo taško radimas/fiksavimas *SIFT* apraše [6]

Kai charakteringi taškai yra surandami keičiant mastelį, juos reikia patikrinti ir užfiksuoti jų vietą atvaizde. Šis procesas vadinamas charakteringų taškų vietos išskyrimu.

Charakteringų taškų vietos išskyrimas (angl. *Keypoint localization*). Šiame žingsnyje išskiriamos tikslios taškų koordinatės, išskyrimui naudojama antros eilės *Teilorio* daugianaris, atliekamas *Gauso* skirtumų funkcijos aproksimavimas. **Pateikiama formulė:**

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x$$

Čia D – *Gauso* skirtumų funkcija, $x = (x, y, \sigma)^T$ – taško poslinkio vektorius. *Teilorio* daugianaro ekstremumas gaunamas skaičiuojant *Gauso* skirtumų funkcijos išvestinę. Ekstremumo taško poslinkis skaičiuojamas pagal **formulę:**

$$\hat{x} = \frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x}$$

Gautas ekstremumo rezultatas yra tikrinamas ar yra didesnis negu 0,5, jeigu ši riba yra peržengiama, vadinasi taško vietos išskyrimas yra klaidingas arba jis gali būti vienas iš kaimyninių taškų. Toliau tikrinami visi kaimyniniai taškai norint aptikti neatrastą tašką. Jeigu ieškant išeinama iš vienos iš oktavų ribų, ekstremumo taško tikrinamas yra nevykdomas. Funkcijos ekstremumo vertė $D(\hat{x})$ taikoma atmesti netinkamiems taškams:

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x}$$

Kai yra atliekamas charakteringų taškų vietos išskyrimas, toliau yra vykdomas charakteringų taškų orientacijos nustatymas.

Charakteringų taškų orientacijos nustatymas (angl. *Oreintation assignment*), atliekant šią operaciją algoritmas analizuoja lokalias užduoto atvaizdo savybes. Norint pasiekt tikslų orientacijos nustatymą ir priskyrimą nepriklausant nuo vaizdo pasisukimo išskiriami šie žingsniai:

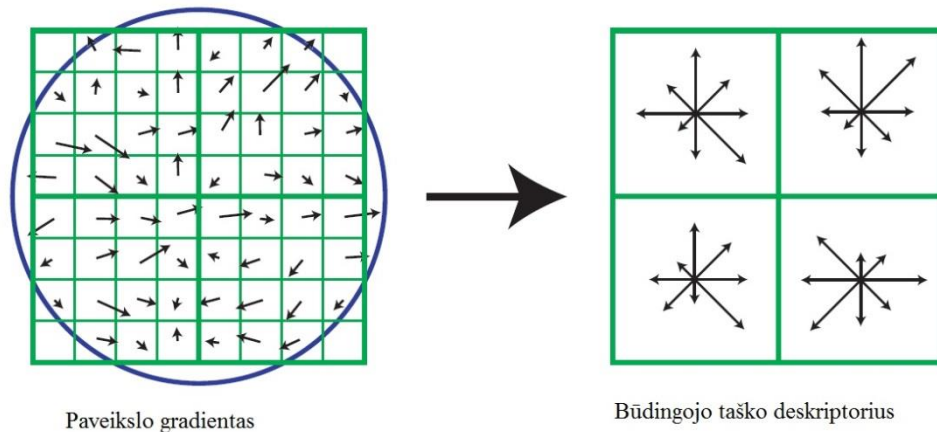
- Sukuriama histograma skirta krypties gradiento skaičiavimui su pasirinktu masteliu;
- Parenkama kanoninė orientacija aukščiausiam histogramos taške;
- Kiekvienam charakteringam taškui po parinktos orientacijos yra nustatomos 2D koordinatės (x, y) , dydis ir orientacija.

Vaizdas išlyginamas atliekamas naudojant *Gauso* funkciją, pagal gautą charakteringųjų taškų kiekį užduotam atvaizdui L . Toliau apskaičiuojamas gradiento dydis $m(x, y)$ ir nustatoma orientacija kiekvienam taškui $Q(x, y)$, kurio koordinatės yra (x, y) aplink jį esantį regioną. **Pateikiamos skaičiavimų formulės:**

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2},$$

$$Q(x, y) = \arctan(L(x, y + 1) - L(x, y - 1)) / (L(x + 1, y) - L(x - 1, y))$$

Charakteringų taškų aprašymas (angl. *keypoint descriptor*) – surastų taškų, kuriems jau nustatyta orientacija savybių aprašymas.



pav. 1.12 Charakteringo taško aprašas *SIFT* apraše [6]

Priskyrus taškams orientaciją, visas užduotas atvaizdas yra suskirstytas į 8 krypčių orientacijos histogramas. Kiekvienos histogramos rodyklės ilgis atitinka histogramos įėjimo gradiento stiprumą kiekvienoje iš 8 krypčių (žr. pav. 1.12). Pavyzdžiui sukuriamas aprašas 4x4 dydžio 16x16 regionams esantiems šalia charakteringo taško, vadinasi gaunasi $4 \cdot 4 \cdot 8 = 128$ elemento krypties vektoriai. [6], [10].

1.3.10 SURF

SURF (angl. *Speeded-up Robust Features*) – greitas stiprių požymių išskyrimas. Šis algoritmas apima abu ir detektorius ir aprašą. Algoritmas yra atsparus nuotraukos pasisukimams, dydžio kitimams, apšvietimo pakitimams ir mažiems žiūrėjimo kampo pasikeitimams.

Bendras SURF veikimo procesas aprašomas šiais žingsniais:

1. Greitas charakteringo taško aptikimas;
2. Orientacijos nustatymas;
3. Aprašas.

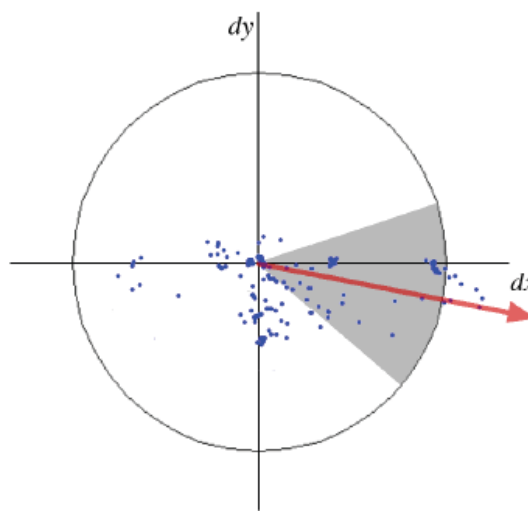
Greito charakteringo taško suradimui (angl. *Fast interest point detection*) yra naudojamas vientisas užduoto atvaizdo pikselių reikšmės atvaizdavimas, kurio pagalba galima naudoti greitus kvadrato tipo sąsukos filtrus. Vientiso vaizdo įėjimas $I_{\Sigma}(x)$ ir padėtis $x = (x,y)$ pateikia visų vaizdo elementų esančių įvesties vaizdo stačiakampio srityje sumą. Stačiakampis yra suformuojamas nuo atvaizdo pradžios iki taško x , pagal formulę:

$$I(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j).$$

Kai yra suskaičiuojamas vaizdo elementų kiekis, dar papildomai tris kartus yra skaičiuojami visų pikselių intensyvumai. Čia skaičiavimo laikas priklauso nuo užduoto atvaizdo dydžio. Toliau norint

išskirti tinkamo intensyvumo pikselius yra sudaroma Hesės matrica ir yra skaičiuojamas jos determinantas. Charakteringo taško aptikimui skaičiuojamas minėto determinanto maksimumas.

Orientacijos nustatymas yra naudojamas tam, kad lyginant atvaizdą su kitu panašiu, tačiau šiek tiek transformuotu atvaizdu, būtų galima aptikti tą patį objektą ar kitas atvaizdo charakteristikas. Nustatyti charakteringų taškų orientacijas *SURF* algoritmas naudoja *Harr* bangelių (angl. *Harr wavelet*) metodą. Minėto metodo atsakas naudojamas x ir y kryptims einantis per paieškos apskritimo spindulius analizuoja kaimyninius charakteringo taško regionus. Kai yra apskaičiuojamas *Harr* bangelių atsakas yra skaičiuojamas charakteringo taško centro svoris pagal Gauso funkciją. Dominuojanti orientacija yra įvertinama apskaičiuojant visų *Harr* bangelių atsakų sumą, su vis skirtingu orientacijos dyžiu. (žr. pav. 1.13)



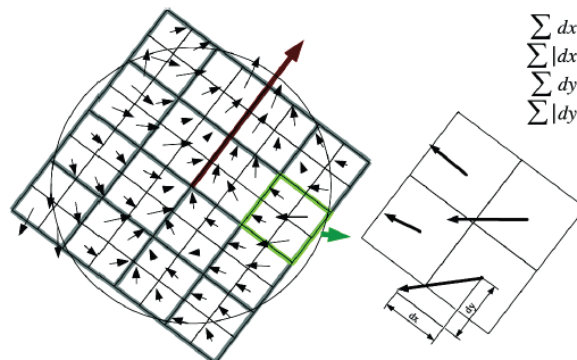
pav. 1.13 Orientacijos nustatymas *SURF* [6]

Toliau yra sumuojami horizontalūs ir vertikalūs atsakai su skirtingais paieškos lango dydžiais. Dviejų tokių atsakų suma yra lokalus orientacijos vektorius. Taip sudedami visi atsakai ir ilgiausias toks vektorius per skirtingus lango dydžius yra priskiriamas kaip dominuojantis ir laikomas charakteringo taško orientacijos vektoriumi. *Harr* bangelių atsako sumos yra gaunamos sudarant centralizuotą kvadratą apie charakteringą tašką (žr. pav. 1.14). Nustatinėjant paieškos lango dydį labai svarbu pats dydžio parinkimas. Jeigu lango dydis bus netinkamas, skaičiuojant atsakus gali atsirasti klaidingi vektoriai, kurie gali klaidingai užfiksuoti charakteringo taško orientaciją.



pav. 1.14 Skirtingų dydžių paieškos langai su priskirta dominuojančia orientacija [6]

Kai taškai yra surasti, juos reikalinga aprašyti. SURF algoritmas aprašui reikalingus duomenis išgauna dar aptikdamas taškus ir nustatydamas orientacija. Minėtieji Haro bangelių atsakų skaičiavimo kvadratai yra skaidomi į mažesnius 4x4 kvadratus, kurie saugo informaciją apie taško charakteristikas ir padėtį erdvėje. Kad užtikrinti charakteringų taškų tvirtumą Haro bangelių atsakas taip pat yra skaičiuojamas kiekviename iš mažesnių kvadratėlių su 5x5 pikselių dydžio tarpeliais. Pateikiamas pav. 1.15 kuriame vaizduojama charakteringo taško aprašymas [6], [9].



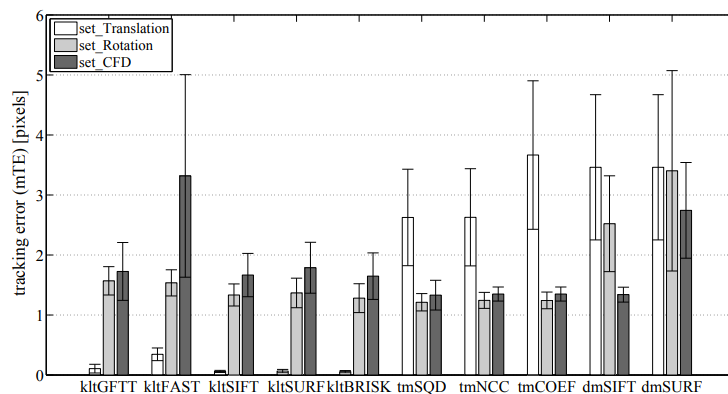
pav. 1.15 Aprašas kvadratiname paieškos lange su 4x4 išskyrimu apie charakteringąjį tašką [6]

Dėl iliustracinių sumetimų paimtas pavyzdys turintis 2x2 mažesnę kvadratą, kuriame skaičiuojamos reikšmės $|dx|$ ir $|dy|$ yra absoliučios dėl to, kad jas suskaičiavus galima sužinoti informaciją apie charakteringo taško intensyvumo pasikeitimus.

SURF veikimo metodika labai artima *SIFT* algoritmui, tačiau vienas papildomas žingsnis, kvadratų skaidymas į mažesnius padaro *SURF* algoritmą atsparesniu triukšmams.

1.3.11 GFTT

GFTT (angl. *Good Features To Track*) – taškų aptikimo detektorius skirtas naudoti su specifiniu vaizdo sekimo algoritmu *KLT-Tracker*. Pagrindinis *GFTT* detektoriaus uždavinys greitai ir optimaliai aptikti charakteringus taškus, kurie bus naudojami vaizdo sekimo algoritmui. *GFTT* yra jautrus nuotraukos transformacijoms taip pat nesaugo informacijos apie gretimais esančius taškus, pagal kuriuos galima spręsti apie nuotraukos orientaciją. Atsižvelgus į kitų mokslininkų gautus rezultatus, palyginus su kitais detektoriais *GFTT* yra lėtesnis tačiau mažiau klystantis detektorius. (žr. pav. 1.16)



pav. 1.16 Mokslinių tyrimų rezultatai palyginantys *GFTT* detektoriaus veikimą su kitais algoritmais [14]

1.3.12 BLOB

Dėmių arba pikselių grupės (angl. *Blob*) – detektorius, kurio pagrindinis tikslas yra aptikti pikselių, kurie yra greta vienas kito ir yra panašių charakteristikų grupės ir jas pažymėti. Norint aptikti dėmes nuotraukoje algoritmas skaidomas į etapus:

- Ribinių reikšmių nustatymas. Atvaizdas konvertuojamas į nespaltvotus atvaizdus ribojant pikselių reikšmes su numatyta riba, nurodamos ribos nuo mažiausios reikšmės iki didžiausios. Konvertuojama tiek atvaizdų, kol pasiekiamas numatyta didžiausia ribinė reikšmė;
- Grupavimas. Kiekviename nespaltvotame atvaizde šviesiausi pikseliai grupuojami į grupes, kurios dar kartais vadinamos – dvejetainėmis grupėmis.
- Suliejimas. Suskaičiuojamos dvejetainių grupių centro koordinatės. Grupės tarp kurių centrų atstumas mažiausias yra suliejamos į vieną.
- Centrų diametro skaičiavimas. Atliekami naujai sudarytų grupių centro koordinatžių ir spindulio aplink jas skaičiavimai.

Dėmės gali būti filtruojamos pagal spalvą, dydį ir formą. Dėmių detektorius dažniausiai naudojamas vaizdų sekose norint aptikti judesį [23].

Šioje dalyje apžvelgti plačiausiai naudojami vaizdų atpažinimo detektoriai ir aprašai. Pastebėta, kad didelė dalis detektorių veikia labai panašiu principu, taip pat išskyrus detektorius pagal tipus, galima teigti, kad vaizdų atpažinimas vykdomas pikselių lygmenyje. Atlikus algoritmų analizę tolimesniems projekto vykdymo etapams pasirinkta naudoti algoritmus, kurie kartu turi ir detektorių ir aprašą.

1.4 Skaitmeninis vaizdų apdorojimas

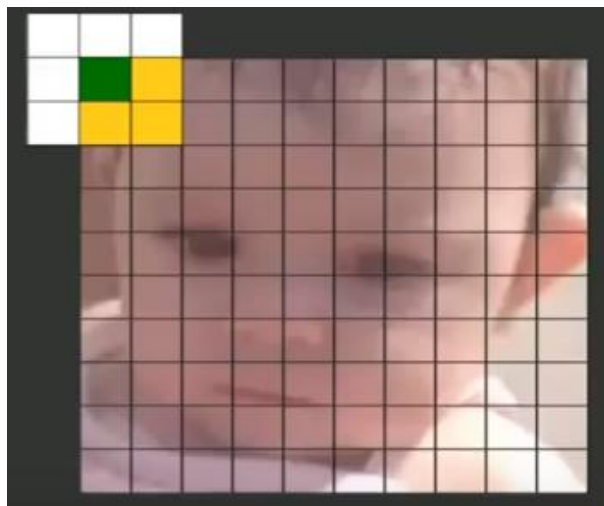
Šiame skyriuje trumpai analizuojami trys vaizdų apdorojimo metodai. Vaizdų apdorojimas tai yra įvairių filtrų pritaikymas atvaizdams, norint pakeisti jų charakteristikas. Šiame darbe naudojamais metodais siekiama iširti vaizdo apdorojimo metodų įtaką vaizdų atpažinimui, ar įmanoma pasiekti geresnių rezultatų su žemiau aprašomais metodais.

1.4.1 Gauso suliejimas

Gauso suliejimas (angl. *Gaussian blur*) – skaitmeninis vaizdų apdorojimo metodas, skirtas suglontinti atvaizdą. Gauso suliejimas realizuojamas naudojant dvimatę Gauso funkciją:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Pagrindinis Gauso suliejimo parametras yra Gauso branduolio dydis. Gauso branduolys tai dvimatė matrica, kurios dydis priklauso nuo vartotojo pasirinkimo. Pavyzdžiui yra pasirenkama 3x3 pikselių dydžio matrica (žr. pav. 1.17), kuri yra slenkama per užduotą atvaizdą, tai reiškia, kad Gauso branduolio viduje yra skaičiuojamos 9 pikselių reikšmės, pagal Gauso pasiskirstymo funkciją, tuomet tos reikšmės yra priskiriamos kiekvienam iš pikselių. Tiesa kiekvienas skaičiavimas atliekamas tris kartus, skirtingiems spalvos kanalams (raudona, žalia, mėlyna). Tokie skaičiavimai yra atliekami iki tol, kol yra išanalizuojamas visas atvaizdas.



pav. 1.17 Gauso suliejimo realizacijos pavyzdys [24]

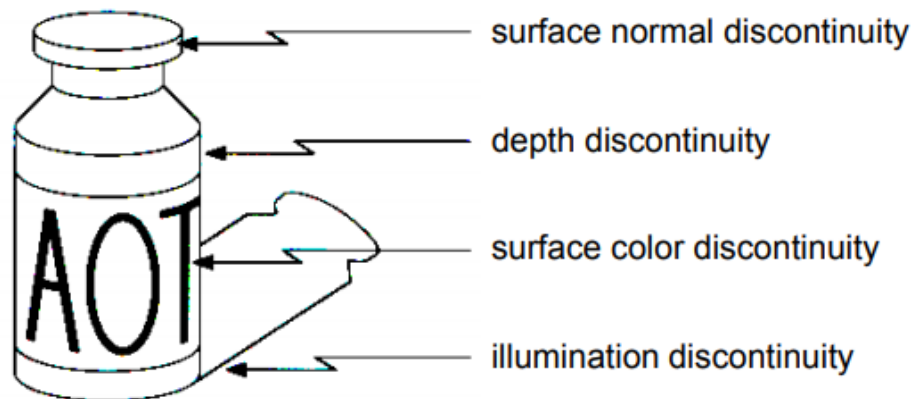
Gautas rezultatas yra užduoto paveikslo suliejimas, atitinkamai pagal Gauso funkcijos dydį, suliejimas gali būti mažesnis, gali būti didesnis, taip pat rezultate pastebimas užduoto atvaizdo detalių sumažėjimas. Gauso suliejimas gali būti naudojamas norint sumažinti atvaizde esančius triukšmus, tačiau neretai triukšmų mažinimas sumažina atvaizdo detalumą, dėl ko vaizdų atpažinimo algoritmams tampa sunkiau aptikti ir aprašyti charakteringuosius taškus [24].

1.4.2 Aštrinimas

Atvaizdo aštrinimas (angl. *Sharpening*) – metodas skirtas suteikti detalumo nuotraukoms, kartais metodas gali būti naudojamas atvaizdo judėjimo suliejimo mažinimui ar papildomų detalių atvaizde paryškinimui. Aštrinimui galima naudoti kelis skirtingus skaičiavimo metodus. Svarbiausias aštrinimo parametras yra atvaizdo aštrinimui yra išskiriami du žingsniai:

- Kraštinių ir kampų suradimas;
- Filtro pritaikymas.

Atvaizdas transformuojamas iš 2D erdvės į daug skirtingų kreivių, kreivės padeda išskirti ryškius atvaizde esančių objektų bruožus, kuriuos yra lengviau aprašyti ir saugoti negu pikselius. Yra išskiriamos kraštinės, kurios gali priklausyti nuo skirtingų faktorių (žr. pav. 1.18).



pav. 1.18 Kraštinių priklausomybė nuo skirtingų faktorių [17]

Kraštinės aptinkamos kai yra aptinkamas greitas kraštinės funkcijos kitimas. Skaiciuojamos kraštinių pirmos eilės išvestinės, kurių ekstremumų taškai apibrėžia funkciją kaip kraštinę. Taip aprašomos vienmatės linijos, tačiau atvaizdai yra dvimačiai, o aptikti ir aprašyti dvimates kraštines yra naudojama Gauso funkcija arba Gauso funkcijos išvestinė. Toliau kraštinės yra filtruojamos pagal nustatomą kraštinių intensyvumo ribą, jeigu aptiktos kraštinės pikselių intensyvumas yra aukščiau numatytos ribos, pikseliai yra paryškunami ir priskiriami aptiktai kraštinei, operacija kartojama, kol išryškintas atvaizde esantis kontūras [17], [26].

1.4.3 Gama korekcija

Gama – atvaizdo šviesos intensyvumo reikšmių pasiskirstymas. Paprastai atvaizdo gama siejama su atvaizdo histograma, tačiau histograma daugiau būtų trimatė gama, atspindinti visų trijų dedamųjų (raudona, žalia, mėlyna) pasiskirstymą, o šiuo atveju gama parodo pikselių pasiskirstymą pagal intensyvumą. Trumpai tariant gama korekcija yra atvaizdo šviesinimas arba tamsinimas, keičiant pikselių intensyvumo reikšmes. (žr. pav. 1.19)



pav. 1.19 Gama korekcijos pavyzdys [2]

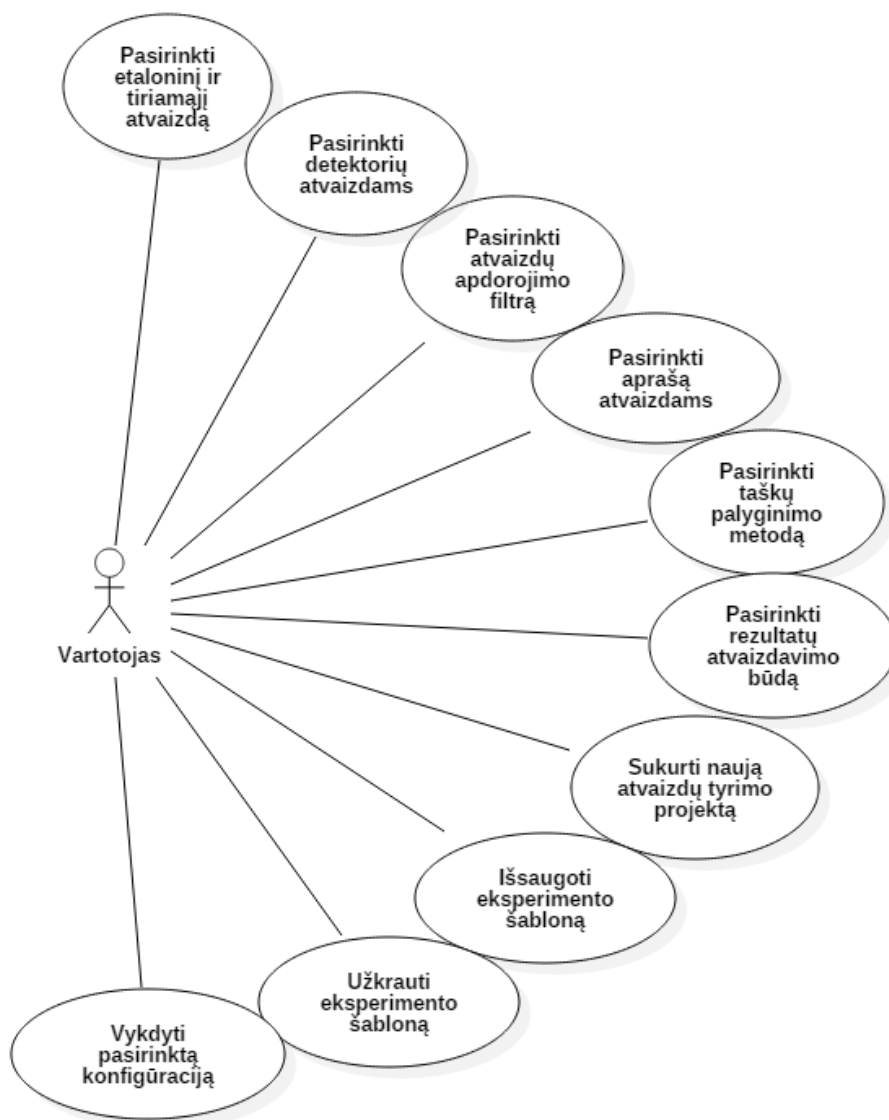
Gama korekcija atliekama paprasčiausiais skaičiavimais, pikselių intensyvumo reikšmės iš (0-255) pakeičiamos į koeficientus (0-1,0). Toliau viskas skaičiuojama pagal formulę $O = I^{1/\Gamma}$, čia I – įvesties atvaizdas, O – išvesties atvaizdas, Γ – gama korekcijos koeficientas. Kai reikšmės yra sudauginamos, pakeisti pikseliai transformuojami atgal į 0-255 skalę. Jeigu gama reikšmė kinta iki 1,0 atvaizdas yra tamsinamas, o jeigu reikšmė kinta virš 1,0 atvaizdas yra šviesinamas. Gama korekcija gali būti naudojama norint išskirti duomenis esančius atvaizdo tamsiuosiuose atspalviuose. Taip pat galima naudoti norint sumažinti atvaizdo šviesumą [2].

2 CHARAKTERINGŲ TAŠKŲ APTIKIMO IR APRAŠYMO SISTEMOS PROJEK TINĖ DALIS

Kuriama sistema skirta atlikti eksperimentus susijusius su charakteringų taškų aptikimu, jų aprašymu ir palyginimu atvaizduose.

2.1 Sistemos funkciniai reikalavimai

Charakteringų taškų aptikimo ir aprašymo sistema yra skirta atlikti eksperimentus susijusius su vaizdų apdorojimu. Kuriama sistema privalo turėti įdiegtus charakteringų taškų aptikimo, jų aprašymo ir palyginimo algoritmus. Taip pat sistemoje turi būti galimybė pritaikyti tiesinio vaizdų apdorojimo filtrus, atvaizdams. Sistemos funkcinis reikalavimus aprašo sistemos panaudos atvejų diagrama.



pav. 2.1 Sistemos panaudos atvejų diagrama

Diagrama nurodo sistemos vartotojo sąveiką su kuriama sistema, taip pat pavaizduoti sistemos funkcijų tarpusavio ryšiai, skirtingais naudojimo atvejais. Vartotojas gali atlikti 10 veiksmų.

lentelė 2.1 Panaudos atvejis Nr. 1

1.Panaudos atvejis	Pasirinkti etaloninį ir tiriamąjį atvaizdą
Aprašymas	Vartotojas gali pasirinkti atvaizdą, kuris bus etaloninis ir atvaizdą, kurį lygins su etaloniniu.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas turi paleisti sistemą.
Po – sąlyga	Pasirinkus atvaizdus galima vykdyti kitas sistemos funkcijas.

lentelė 2.2 Panaudos atvejis Nr. 2

2.Panaudos atvejis	Pasirinkti detektorių atvaizdams
Aprašymas	Vartotojas turi pasirinkti detektorių, kuriuo bus ieškomi charakteringi taškai.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas turi pasirinkti atvaizdą, kuriam bus taikomi pasirinktas detektorius.
Po – sąlyga	Pasirinkus detektorių, vartotojas gali pereiti prie kito žingsnio.

lentelė 2.3 Panaudos atvejis Nr. 3

3.Panaudos atvejis	Pasirinkti atvaizdų apdorojimo filtrą
Aprašymas	Vartotojas gali pasirinkti vieną iš realizuotų vaizdų apdorojimo filtrų.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas privalo būti pasirinkęs atvaizdą, kurį nori apdoroti.
Po – sąlyga	Pasirinkus apdorojimo filtrą, vartotojas privalo toliau tęsti programos vykdymą.

lentelė 2.4 Panaudos atvejis Nr. 4

4.Panaudos atvejis	Pasirinkti aprašą atvaizdams
Aprašymas	Vartotojas turi pasirinkti vieną iš realizuotų charakteringų taškų aprašų.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Prieš pasirenkant charakteringų taškų aprašą, vartotojas privalo būti pasirinkęs taškų detektorių.
Po – sąlyga	Pasirinkus aprašą vartotojas turi toliau tęsti programos vykdymą.

lentelė 2.5 Panaudos atvejis Nr. 5

5.Panaudos atvejis	Pasirinkti taškų palyginimo metodą
Aprašymas	Vartotojas turi pasirinkti dviejų atvaizdų aptiktų ir aprašytų charakteringų taškų palyginimo metodą.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Iki šio žingsnio vartotojas privalo būti pasirinkęs charakteringų taškų detektorių ir aprašą.

Po – sąlyga	Vartotojas turi pereiti prie kito sistemos konfigūravimo žingsnio.
-------------	--

lentelė 2.6 Panaudos atvejis Nr. 6

6.Panaudos atvejis	Pasirinkti rezultatų atvaizdavimo būdą
Aprašymas	Vartotojas gali pasirinkti kaip bus atvaizduojami gauti rezultatai.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas privalo būti pasirinkęs atvaizdą, detektorių, aprašą ir rezultatų atvaizdavimo būdą.
Po – sąlyga	Vartotojas gali peržiūrėti rezultatus ir baigti programos vykdymą.

lentelė 2.7 Panaudos atvejis Nr. 7

7.Panaudos atvejis	Sukurti naują atvaizdų tyrimo projektą
Aprašymas	Vartotojas gali sukurti naują atvaizdų tyrimo projektą, konfigūruodamas sistemą pagal savo poreikius.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas privalo paleisti sistemą.
Po – sąlyga	–

lentelė 2.8 Panaudos atvejis Nr. 8

8.Panaudos atvejis	Išsaugoti eksperimento šabloną
Aprašymas	Vartotojas sukongūruotą sistemos šabloną gali išsaugoti.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas privalo paleisti sistemą.
Po – sąlyga	–

lentelė 2.9 Panaudos atvejis Nr. 9

9.Panaudos atvejis	Užkrauti eksperimento šabloną
Aprašymas	Vartotojas gali užkrauti išsaugotą eksperimento šabloną.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas privalo paleisti sistemą.
Po – sąlyga	–

lentelė 2.10 Panaudos atvejis Nr. 10

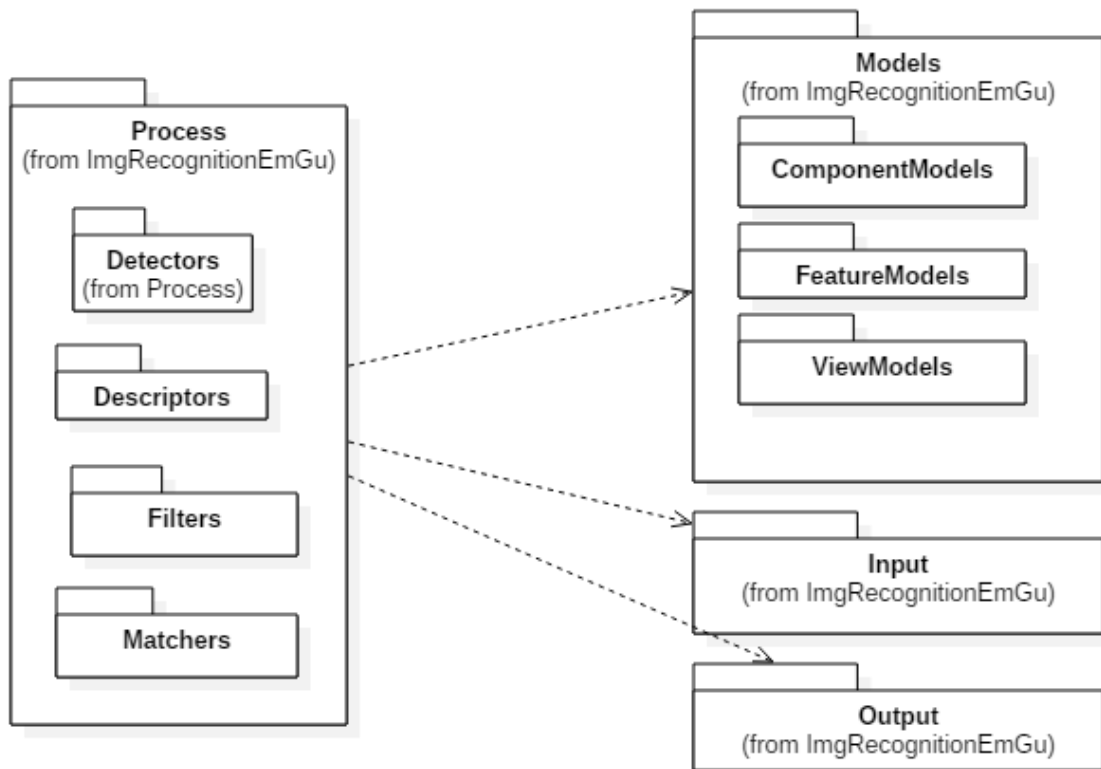
10.Panaudos atvejis	Vykdyti pasirinktą konfigūraciją
Aprašymas	Vartotojas gali įvykdyti pasirinktą konfigūraciją. Sistemos veikimo paleidimas.
Aktoriai	Paprastas sistemos vartotojas.
Prieš – sąlyga	Vartotojas turi būti pasirinkęs visus sistemos dedamuosius: atvaizdai, detektoriai, aprašai, taškų palyginimo metodas, duomenų

	atvaizdavimo metodas. Arba turi užsikrauti jau išsaugotą sistemos eksperimento šabloną.
Po – sąlyga	–

Panaudus atvejų diagrama apibūdina visus galimus vartotojo veiksmus sistemoje, toliau pateikiamas sistemos statinis vaizdas, paketų, veiklos diagramos.

2.2 Statinis sistemos vaizdas

Pateikiama sistemos paketų diagrama:



pav. 2.2 Sistemos paketų diagrama

Sistema sudaryta iš keturių paketų:

- *Process* paketas;
- *Input* paketas;
- *Output* paketas;
- *Models* paketas.

Toliau kiekvienas iš paketų apžvelgiamas plačiau.

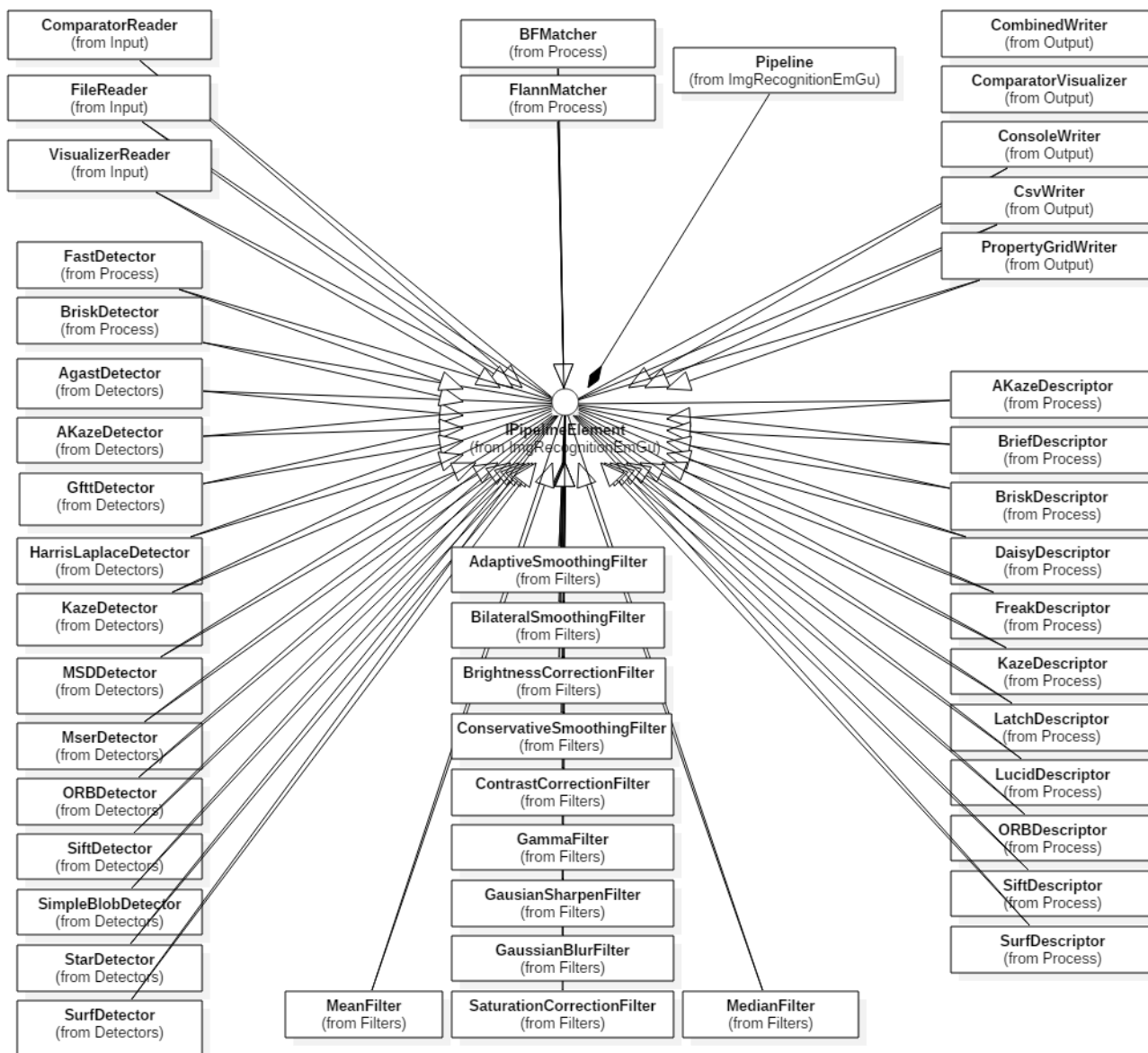
Process paketas yra sudedamoji *Pipe and Filters* projektavimo šablono dalis. Šį paketą sudaro keturios klasės skirtos darbui su vaizdų apdorojimu. Klasė *Detectors* apima visus sistemoje realizuotus charakteringų taškų detektorius. Klasė *Descriptors* apima visus sistemoje realizuotus charakteringų

taškų aprašus. Paketas *Filters* sudarytas iš visų sistemoje realizuotų atvaizdų apdorojimo filtrų. Klasė *Matchers* sudarytas iš sistemoje realizuotų charakteringų taškų palyginimo metodų.

Paketas *Models* nerealizuoja *IpipelineElement* interfeiso metodų. Šis paketas yra skirtas skirtingiems duomenų modeliams realizuoti. Į paketą *Models* įeina trys kiti paketai skirti realizuoti skirtingus duomenų modelius. Paketas *ComponentModels* sudarytas iš pagrindinių sistemos vykdymo funkcijų duomenų struktūrų. Paketas *FeatureModels* apima visus metodus susijusius su charakteringų taškų duomenų skaičiavimais. Paketas *ViewModels* sudarytas iš funkcijų skirtų gautų rezultatų atvaizdavimui.

Paketas *Input* šioje sistemoje priklauso *Pipe and Filters* šablonui, o jo klasės įgyvendina *IpipelineElement* elemento funkcijas. Šis paketas skirtas užtikrinti tinkamą duomenų nuskaitymą iš skirtingų šaltinių. Paketą realizuoja funkcijos skirtos nuskaityti duomenų palyginimo klasės duomenis, skirtingų failų duomenų struktūras ir rezultatų atvaizdavimo duomenų tipus.

Paketas *Output* taip pat įgyvendina *IpipelineElement* elemento funkcijas. Paketas skirtas užtikrinti tinkamą duomenų išvedimą ir pateikimą vartotojui. Šis paketas realizuoja funkcijas skirtas atvaizduoti kombinuotą duomenų išvedimą, kai išvedama tiek charakteringų taškų kiekis ir kiti skaičiavimai, tiek tiriamasis ir etaloninis atvaizdas su ant jų nupieštais charakteringais taškais ir jų atitikimais. Taip pat pakete yra funkcija, kuri atsakinga už duomenų išvedimą į komandinį langą ar programos langą.



pav. 2.3 Sistemos klasių įgyvendinančių Pipes and Filters projektavimo šablono diagrama

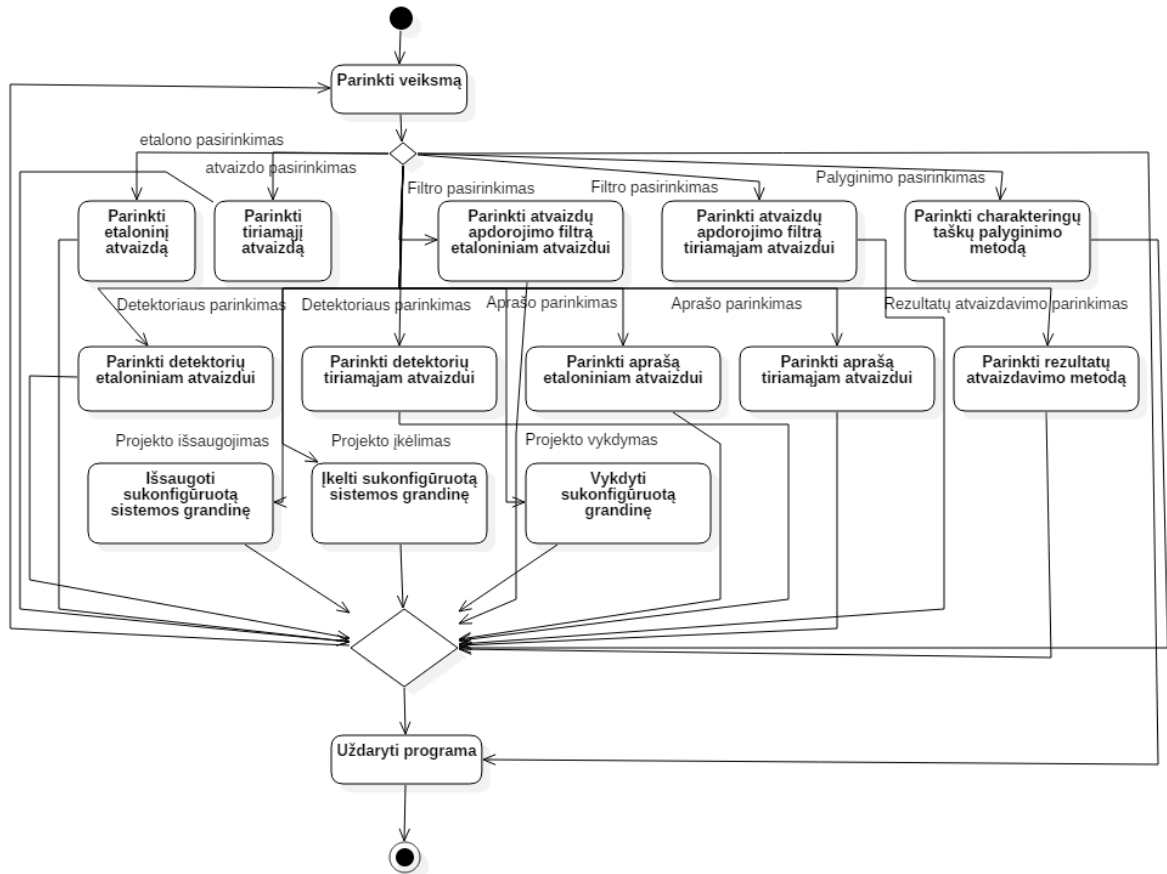
Sistemos klasių diagrama yra išskaidyta į skirtingas dalis, kaip ir vaizdų atpažinimo veiksmų seka. Visų pirma yra pasirenkami atvaizdai, kuriuos norima tirti, šiam tikslui yra realizuotas paketas *Input* su duomenų įvesties klasėmis, kurios skirtos failų, komparatorių ir vizualių duomenų nuskaitymui. Toliau yra pasirenkama charakteringų taškų detektorius, klasių diagramoje yra pateikiama 14 skirtingų klasių, kurios realizuoja atitinkamus detektorius, visa tai sudaro *Detectors* paketą. Trečias žingsnis vaizdų atpažinime yra aprašo pasirinkimas, kaip matyti klasių diagramoje šioje sistemoje yra realizuoti 11 charakteringų taškų aprašų, kurie atitinkamai yra suskirstyti į klases ir realizuoja *Descriptors* paketą. Kai charakteringi taškai yra aptikti ir aprašyti yra reikalinga juos palyginti norint nuspręsti ar jie atitinka tą pačią vietą atvaizduose. Šiam tikslui sistemoje yra paketas *Matchers*, kuris sudarytas iš dviejų skirtingų charakteringų taškų palyginimo metodų. Paskutinis žingsnis – rezultatų atvaizdavimas, suprojektuotoje sistemoje yra sukurtas paketas *Outputs*, kuris apima visas klases skirtas

duomenų išvedimui į ekraną, komandinę eilutę, charakteringų taškų pažymėjimą skirtinguose atvaizduose bei atitikimų užpiešimą ant dviejų atvaizdų vaizduojamų greta vienas kito.

Toliau pateikiama sistemos veiklos diagrama, kuri atvaizduoja sistemos veikimo žingsnius.

2.3 Sistemos dinaminis vaizdas

Šiame poskyryje vaizduojama programos vidiniai procesai. Pateikiama ir aptariama programos veiklos diagrama.

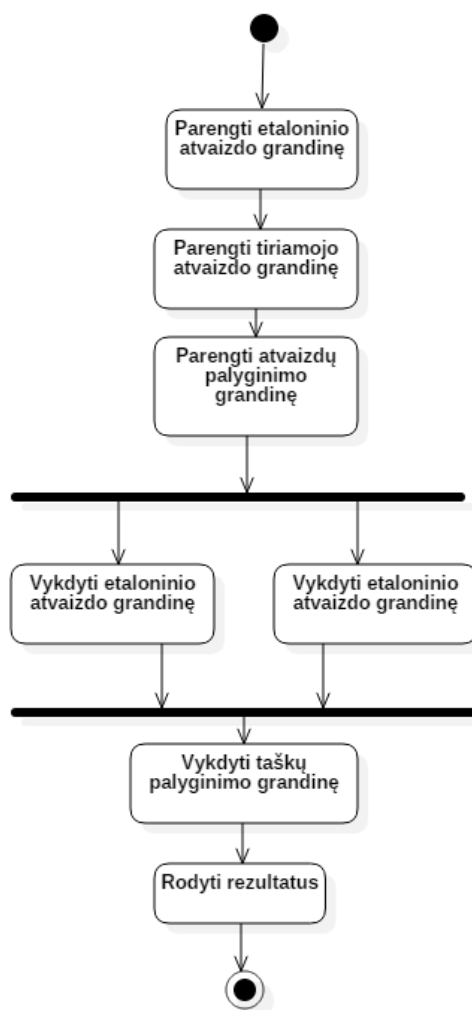


pav. 2.4 Programos veiklos diagrama

Šioje diagramoje pateikiamas pagrindinio programos lango funkcionalumas. Atsidarius programą, vartotojas gali:

- Parinkti skaičiavimų grandinės objektų parametrus;
- Užkrauti įrašytą projektą;
- Įrašyti projektą;
- Sukurti naują projektą;
- Vykdyti skaičiavimų grandinę.

Skaičiavimų grandinės vykdymo tvarka yra pateikta pav. 2.5.

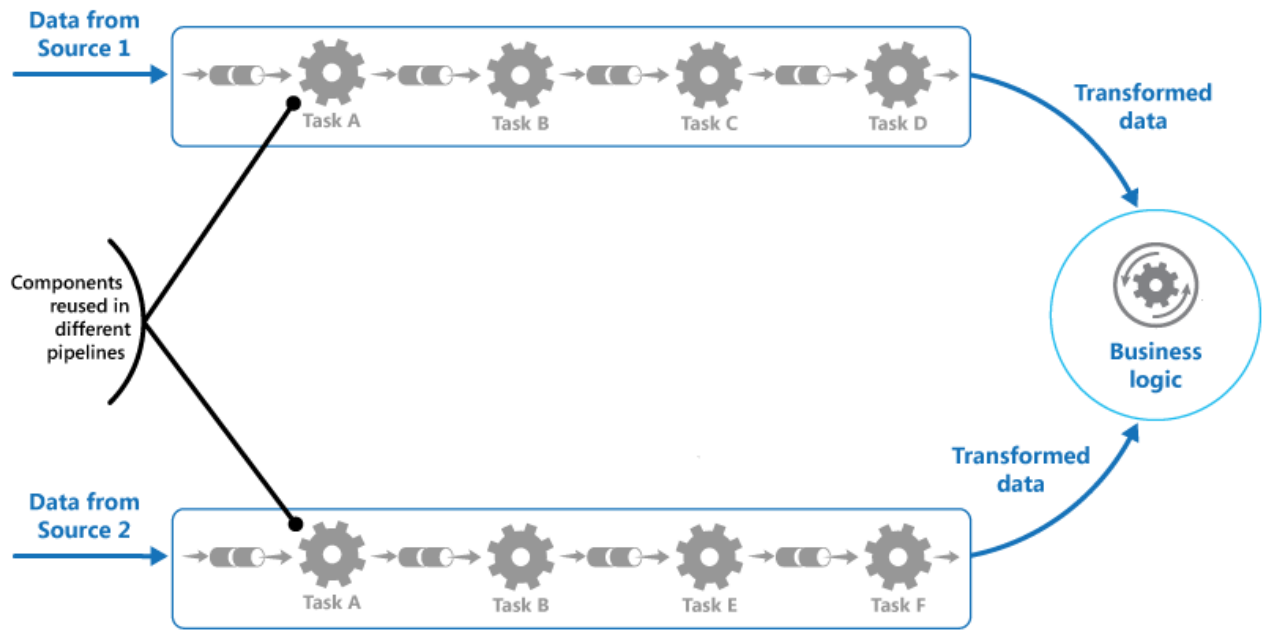


pav. 2.5 Grandinės skaičiavimų veiklos diagrama

Atvaizdų charakteringų taškų skaičiavimų grandinė susideda iš trijų dalių: etaloninio atvaizdo grandinės skaičiavimai, tiriamojo atvaizdo grandinės skaičiavimai ir atvaizdų palyginimo grandinės skaičiavimai. Į etaloninio ir tiriamojo atvaizdo grandinės skaičiavimus įeina atvaizdų nuskaitymas, atvaizdų apdorojimo filtro pritaikymas, charakteringų taškų detektoriaus pritaikymas ir charakteringų taškų aprašo pritaikymas. Į palyginimo grandinę įeina aptiktų ir aprašytų charakteringų taškų palyginimas. Į grandines sudėjus skaičiavimų objektus pirmiausia yra lygiagrečiai paleidžiamos etaloninio ir tiriamojo atvaizdo grandinės. Kai abi skaičiavimų grandinės yra įvykdomas – asinchroniškai paleidžiama palyginimo grandinė. Atlikus skaičiavimus valdymas grįžta į pagrindinę giją, kurioje pagal pasirinktą išvedimo objektą pavaizduojami objektai.

3 MOBILIŲJŲ JUTIKLIŲ PARODYMŲ ĮTAKOS VAIZDŲ ATPAŽINIMUI TYRIMAS

Tyrimo pagrindinis tikslas ištirti ar mobiliųjų įrenginių jutiklių parametrai turi įtakos charakteringų taškų aptikimo ir išskyrimo algoritams. Tam tikslui yra sukurta sistema, kurios pagalba galima analizuoti charakteringų taškų pateikiamus rezultatus. Taip pat tyrimui reikalingi duomenys skirti vaizdų apdorojimui. Paruoštas duomenų rinkinys su skirtingais mobiliųjų įrenginių jutiklių parametru parodymais.

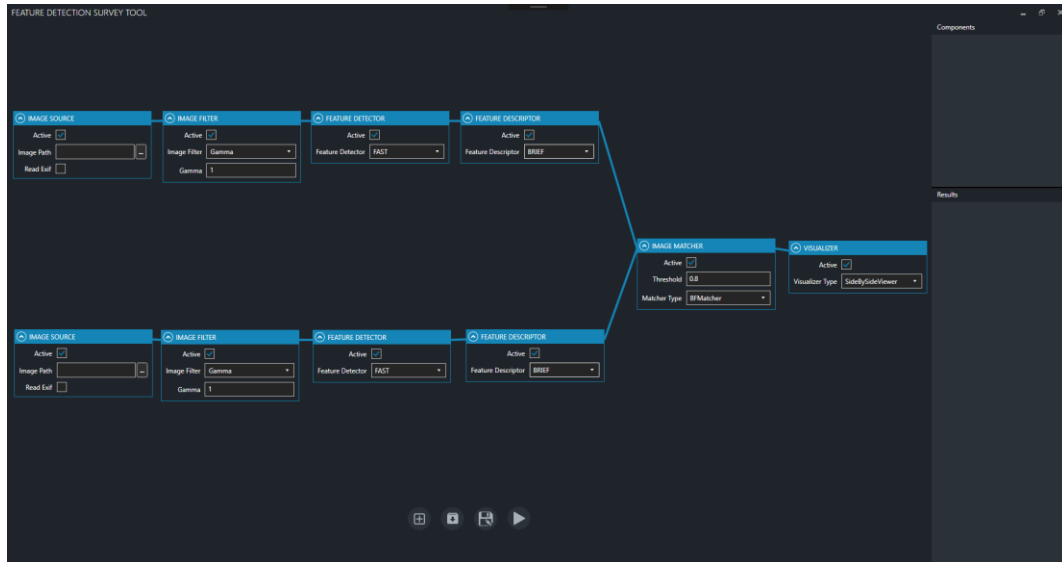


pav. 3.1 *Pipes and Filters* šablono orientacinė schema [27]

3.1 Sistema skirta eksperimentuoti su vaizdų atpažinimo algoritmais

Sukurta sistema yra realizuota naudojant C# programavimo kalbą ir vaizdų apdorojimui skirtas bibliotekas: *openCV*, *exif extractor*, *Aforge*. Kuriant sistemą pasirinktas *Pipes and Filters Pattern* projektavimo šablonas (žr. pav. 3.1), kuris suteikia galimybę programos vykdymo procesus išskaidyti į atskirus. Transformuojant duomenis į reikiamus tipus, kiekvienas atskiras procesas gali būti jungiamas tarpusavyje į vieną didesnę procesą. Tokiu būdu yra išvengiama kodo dubliavimo, galima lengviau prijungti ar pakeisti programos komponentus. Sukurta vaizdų atpažinimo sistema (žr. pav. 3.2), kurios veikimo principas paremtas vaizdų atpažinimo vykdomais etapais:

- Charakteringų taškų aptikimas;
- Charakteringų taškų aprašymas;
- Charakteringų taškų atitikmenų paieška.



pav. 3.2 Algoritmų kombinavimas sukurtoje sistemoje

Dėl tyrimo specifikos taip yra realizuota daugiau filtrų, kurie yra būtini vaizdų atpažinimui, tai filtrai skirti nurodyti įvesties atvaizdą, galimybė atlikti paprastą vaizdų apdorojimą ir atvaizduoti gautus rezultatus. Toliau detaliam aptariami sistemos filtrai.

3.1.1 Įvesties atvaizdo užkrovimo filtras

Šis skirtas pasirinkti atvaizdai, kuris bus analizuojamas. Šio filtro įėjimas yra *.jpg* formato atvaizdas. Filtras atvaizdą paruošia tolimesniam apdorojimui ir perduoda *Mat* tipo struktūrą sekančiam filtrui. *Mat* *OpenCV* bibliotekos duomenų tipas yra n-matė tanki matrica.

3.1.2 Atvaizdo apdorojimo filtras

Šiame filtre pasirinktinai galima skaitmeniniu būdu pritaikyti glotninimo ir tiesinio apdorojimo filtrus. Sistemoje realizuoti tokie glotninimo filtrai:

- Prisitaikantis glotninimas (*Adaptive smoothing*);
- Dvišalis glotninimas (*Bilateral smoothing*);
- Konservatyvus glotninimas (*Conservative Smoothing*);
- Gauso suliejimas (*Gaussian Blur*);
- Vidurkio reikšmės glotninimas (*Mean*);
- Medianos glotninimas (*Median*).

Taip pat realizuoti šie tiesinio apdorojimo filtrai:

- Kontrasto korekcija (*Contrast Correction*);
- Gauso aštrinimas (*Gaussian Sharpening*);
- Šviesumo korekcija (*Brightness Correction*);

- Sodrumo korekcija (*Saturation Correction*);
- Atvaizdo gama korekcija (*Gamma*).

Šio programos filtro įėjimas yra *Mat* duomenų struktūra, su kuria filtras atlieka skaičiavimus ir išėjime taip pat suformuoja tokią pat *Mat* duomenų struktūrą, kuri yra paruošta tolimesniems skaičiavimams.

3.1.3 Charakteringų taškų detektoriaus filtras

Šiame filtre galima pasirinkti skirtingus charakteringų taškų detektorius, kurie užfiksuoja tam atvaizdui būdingus taškus ir jų charakteristikas. Sukurtoje sistemoje realizuoti šie detektoriai:

- AGAST;
- AKAZE;
- BLOB;
- FAST;
- GFTT;
- HarrisLaplace;
- KAZE;
- MSD;
- MSER;
- ORB;
- SURF;
- SIFT;
- STAR.

Šiam filteriui yra pateikiama *Mat* duomenų struktūra, kuri pagal vartotojo pasirinkimą yra apdorojama vienu iš charakteringų taškų detektorių. Apdorojus struktūrą, filtras pateikia charakteringų taškų matricą tolimesniems skaičiavimams.

3.1.4 Charakteringų taškų aprašo filtras

Kad būtų galima išskirti atvaizdo taškus ir juos aprašyti yra būtina juos patikrinti. Šiame filtre surasti taškai yra patikrinami ir aprašomi vienu iš vartotojo pasirinktų aprašų. Vaizdų atpažinimo algoritmai sudaryti iš detektorių ir aprašų, tačiau ne visi detektoriai turi savo aprašus, dėl to aptikus taškus su vienu detektoriumi, juos aprašyti galima su kitu aprašu. Šioje sistemoje yra realizuoti šie aprašai:

- BRIEF;

- BRISK;
- DAISY;
- FREAK;
- KAZE;
- AKAZE;
- LATCH;
- LUCID;
- ORB;
- SIFT;
- SURF.

Šiam filtrui yra perduodama aptiktų charakteringų taškų matrica, kuri yra apdorojama pasirinktu aprašu ir toliau paruošiama atvaizdų charakteringų taškų palyginimui. Taigi šio filtro išėjime pateikiama aprašytų charakteringų taškų matrica.

3.1.5 Atvaizdų palyginimo filtras

Atvaizdų palyginimo filtras palygina dviejų atvaizdų aprašytus charakteringus taškus, apskaičiuoja jų kiekį ir paruošia atvaizdus atvaizdavimui. Šio filtro įėjime yra du atvaizdai aprašyti charakteringais taškais, kurie yra lyginami tarpusavyje. Filto išėjime yra pateikiamos atvaizdų charakteringų taškų koordinatės, skirtos atvaizduoti taškus ant atvaizdų, taip pat yra pateikiami skaičiavimų rezultatai apie charakteringų taškų atitikimo kiekį bei skaičiavimo trukmę.

Šioje sistemoje yra realizuoti du charakteringų taškų palyginimo metodai: *BFMatcher* ir *FlannMatcher*. Pirmasis atvaizdų palyginimo metodas tinkamas visų tipų vaizdų atpažinimo algoritmams, o antrasis gali lyginti tik ne dvejetainių algoritmų pateikiamus duomenis. Jeigu norima palyginti dvejetainio tipo algoritmo duomenis su *FlannMatcher* metodu yra būtina transformuoti algoritmo atsako reikšmes į kitas.

3.1.6 Duomenų atvaizdavimo filtras

Šis filtras skirtas atvaizduoti rezultatus gautus atliekant vaizdų atpažinimą. Šiame filtre yra keli pasirinkimai kaip atvaizduoti duomenis, kurie gali būti šio filtro išėjimo duomenys:

- Atvaizduoti duomenis komandinėje eilutėje (*Console Writer*);
- Užrašyti duomenis .CSV formatu (*CSV Writer*);
- Palyginamasis dviejų atvaizdų vaizdas (*Side by Side Viewer*);
- Parametrų išvedimas programos lange (*Property Grid View*);
- Kombinuotas atvaizdavimas (*Combined View*).

Filtro įėjime yra visų ankstesnių žingsnių rezultatai. Į rezultatų duomenų struktūrą įeina:

- Originalūs atvaizdai;
- Atvaizdai po vaizdų filtrų pritaikymo;
- Taškų aptikimo rezultatai;
- Taškų aprašymo rezultatai;
- Atvaizdų palyginimo rezultatai.

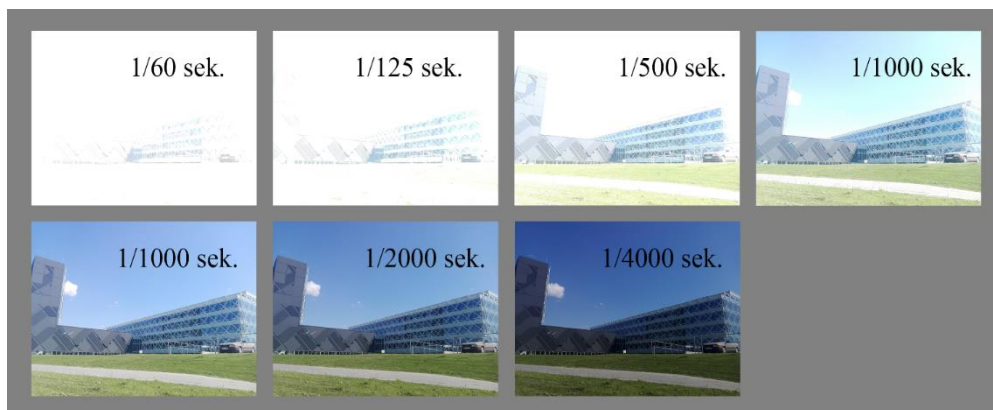
3.2 Tyrimo duomenų rinkinys

Tyrimui atlikti yra sudarytas duomenų rinkinys su skirtingais mobiliųjų įrenginių parametrais.

Rinkiniui sudaryti yra panaudoti 9 skirtingi mobilieji įrenginiai:

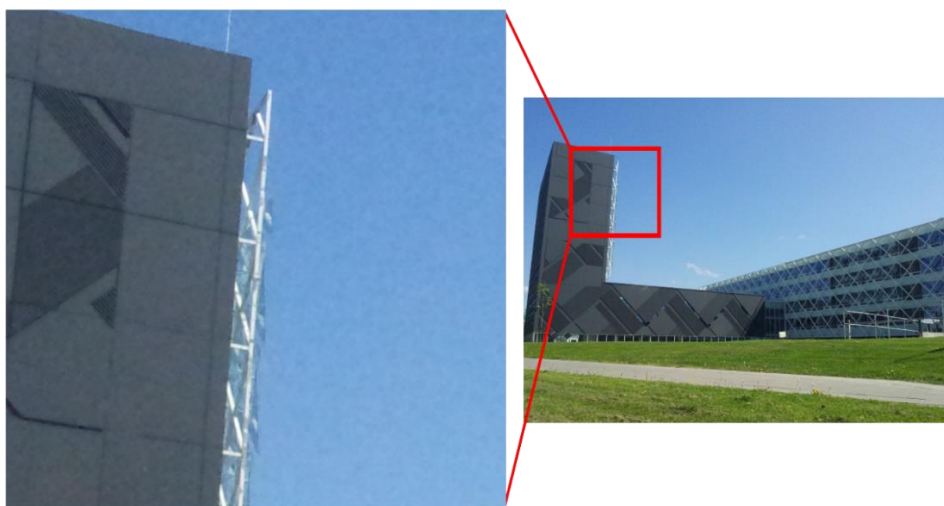
1. „Huawei P8 Lite“;
2. „Samsung Galaxy S II“;
3. „Samsung Galaxy Note“;
4. „Xiaomi Mi A1“;
5. „Motorola Moto G5 Plus“;
6. „Samsung Galaxy S7 edge“;
7. „Samsung Galaxy S8“;
8. „Samsung Galaxy S9+“;
9. „Samsung Galaxy J5“.

Duomenų rinkiniui buvo naudojami mobiliųjų įrenginių vaizdo jutiklio parametrai: užrakto greitis ir vaizdo jutiklio jautrumo parodymai, plačiau apie jutiklius (žr. 1.1.12). Renkant duomenis ne visi mobilieji įrenginiai suteikia galimybę valdyti užrakto greitį, todėl duomenų rinkinys išskaidytas į dvi grupes: atvaizdai su skirtingais užrakto greičio parodymais ir atvaizdai su skirtingomis vaizdo jutiklio jautrumo reikšmėmis. Sudarant duomenų rinkinį pasirinktas statinis objektas su neknitančia geometrija. Fiksuojant atvaizdą iš pradžių yra užfiksuojamas atvaizdas su automatiniais vaizdo jutiklio parametrais, vėliau atvaizdai fiksuojami keičiant parametrus rankiniu būdu.



pav. 3.3 Užrakto greičio reikšmės kitimo žingsnis

Renkant duomenis priklausančius duomenų grupei su skirtingais užrakto greičio parametrais buvo parinktas žingsnis per pus trumpinant užrakto greitį pagal prieš tai buvusią nuotrauką. Pateikiamas pav. 3.3 kuriame matyti atvaizdo užrakto greičio žingsnio kitimas ir gautas rezultatas. Kitoje duomenų grupėje fiksuojant atvaizdus buvo siekiama užfiksuoti skaitmeninius triukšmus sukeltus dėl mobiliojo įrenginio vaizdo jutiklio jautrumo reikšmių kitimo. Įrenginiai, kurie buvo naudoti atvaizdų fiksavimui leido naudoti keturias jautrumo reikšmes: 100, 200, 400 ir 800. Pateikiama pav. 3.4, kuriame 300% padidintas pasirinktas atvaizdo plotas iliustruoja skaitmeninius atvaizdo triukšmus. Atvaizde esančių objektų kraštinės ir raštas yra blyškūs ir neryškūs, neišlaiko realių savo charakteristikų, dėl to charakteringų taškų išskyrimas tampa komplikuoatas.



pav. 3.4 Jautrumo reikšmių įtakos atvaizdai padidintas vaizdas

Fiksuojant atvaizdus iš pradžių yra užfiksuojamas atvaizdas su automatiniais vaizdo jutiklio jautrumo parametrais, vėliau parametrai keičiami rankiniu būdu, kadangi kai kuriuose įrenginiuose kontroliuoti užrakto greičio programiškai negalime, ženklus pasikeitimo atvaizdo apšviestume negalime matyti, tačiau bandant aptikti ir išskirti charakteringus taškus, pastebimas skirtumas tarp skirtingų jautrumo parametrų reikšmių.

4 MOBILIŲJŲ ĮRENGINIŲ JUTIKLIŲ ĮTAKOS VAIZDŲ ATPAŽINIMUI EKSPERIMENTINĖ DALIS

Eksperimento tikslas – ištirti mobiliųjų įrenginių jutiklių parametų daromą įtaką vaizdų atpažinimui. Šiame skyriuje aprašomi eksperimentai skirti tyrimui atlikti, pateikiama eksperimentų metodika ir eksperimentų rezultatai taip pat kiekvieno eksperimento pabaigoje pateikiamas trumpas rezultatų aptarimas. Norint nustatyti, kokia yra vaizdo jutiklių parametų įtaka vaizdų atpažinimui, buvo atliekami šie eksperimentai:

1. Charakteringų taškų aptikimo priklausomybė nuo įrenginio vaizdo jutiklio;
2. Charakteringų taškų aptikimo priklausomybė nuo įrenginio vaizdo jutiklio jautrumo parametų;
3. Mobiliųjų įrenginių vaizdo jutiklio jautrumo keliamų/kuriamų triukšmų įtakos, charakteringų taškų aptikimui, mažinimas.

4.1 Priemonės eksperimentui atlikti

Eksperimento duomenys yra atvaizdai užfiksuoti su 9 skirtingo modelio mobiliaisiais įrenginiais. Ne visi eksperimente naudojami prietaisai gali keisti daugiau negu vieną vaizdo jutiklio parametą. Pagrindiniai vaizdo jutiklio parametrai mobiliuosiuose įrenginiuose yra vaizdo jutiklio jautrumas (ISO) ir užrakto greitis (*shutter speed*) patiekiami eksperimente naudojamų prietaisų lentelė.

lentelė 4.1 Eksperimento priemonės

NR.	Įrenginio modelis	Jutiklio jautrumo valdymas	Užrakto greičio valdymas	Vaizdo jutiklio parametrai
1.	„Huawei P8 Lite“	Taip	Taip	Vaizdo jutiklis CMOS, 13 Megapikselių.
2.	„Samsung Galaxy S II“	Taip	Ne	Vaizdo jutiklis CMOS, 8 Megapikselių.
3.	„Samsung Galaxy Note“	Taip	Ne	Vaizdo jutiklis CMOS, 8 Megapikselių.
4.	„Xiaomi Mi A1“	Taip	Taip	Vaizdo jutiklis CMOS, 12 Megapikselių.
5.	„Motorola Moto G5 Plus“	Taip	Taip	Vaizdo jutiklis CMOS, 12 Megapikselių.
6.	„Samsung Galaxy S7 edge“	Taip	Taip	Vaizdo jutiklis CMOS, 12 Megapikselių.

7.	„Samsung Galaxy S8“	Taip	Taip	Vaizdo jutiklis CMOS, 12 Megapikselių.
8.	„Samsung Galaxy S9+“	Taip	Taip	Vaizdo jutiklis CMOS, 12 Megapikselių.
9.	„Samsung Galaxy J5“	Taip	Ne	Vaizdo jutiklis CMOS, 13 Megapikselių.

Atvaizdai fiksuojami kiekvienu iš mobiliųjų įrenginių. Keičiami vaizdo jutiklio parametrai, jautrumas ir užrakto dydis. Galimos vaizdo jutiklio jautrumo reikšmės yra: 100, 200, 400, 800. Tuo tarpu vaizdo jutiklio užrakto greičio reikšmės kinta nuo didžiausios galimos vis mažinant per pusę: 1/4000sek., 1/2000sek., 1/1000sek., 1/500sek., 1/250sek., 1/125sek., 1/60sek., 1/30sek., 1/15sek. ir t.t. Taip pat užfiksuojamas atvaizdas su automatiniais vaizdo jutiklio parametrais, minėtas atvaizdas bus laikomas etaloniniu.

4.2 Charakteringų taškų aptikimo priklausomybės nuo įrenginio vaizdo jutiklio eksperimentas

4.2.1 Tikslas

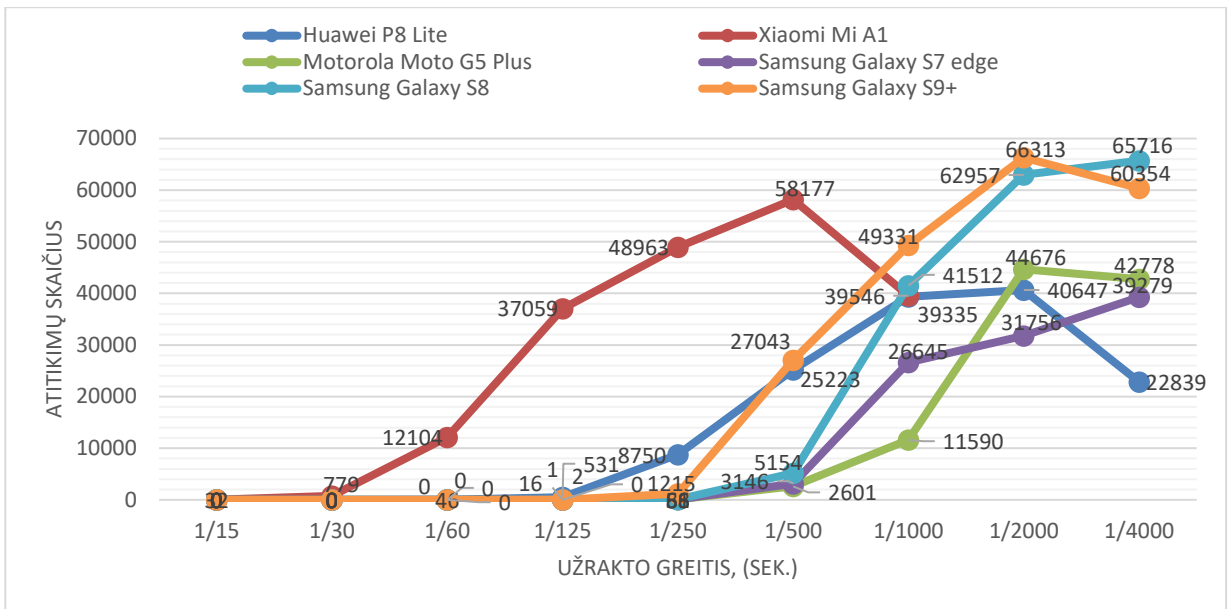
Šio eksperimento tikslas nustatyti vaizdų atpažinimo išskiriamų charakteringų taškų priklausomybę nuo mobiliųjų įrenginių vaizdo jutiklio.

4.2.2 Metodika

Norint nustatyti vaizdo jutiklių efektyvumą vaizdų atpažinimui yra lyginama atvaizdų atitikimai (*Matches*) tarp etaloninio (atvaizdas su automatiniais parametrais) atvaizdo ir atvaizdo su skirtingais jutiklio parametrais, daugiausiai atitikimų surinkęs vaizdo jutiklis laikomas efektyviausiu vaizdų atpažinimui.

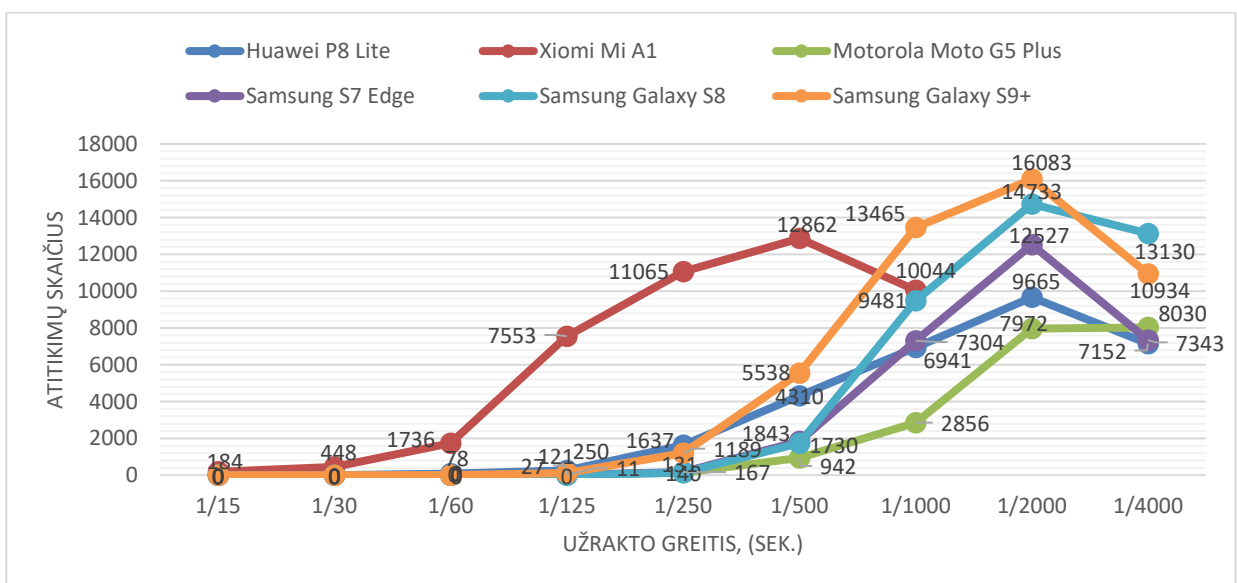
4.2.3 Eksperimento rezultatai

Pateikiami grafikai, kurie atvaizduoja charakteringų taškų priklausomybę nuo skirtingų mobiliųjų įrenginių vaizdo jutiklių. Eksperimente naudoti 5 algoritmai, kuriuos sudaro charakteringų taškų detektorius ir aprašas tai: *SURF*, *SIFT*, *ORB*, *AKAZE*, *BRISK*. Taip pat yra atliktos dvi iteracijos naudojant detektoriaus *ORB* derinį su aprašais *BRIEF* ir *FREAK*.



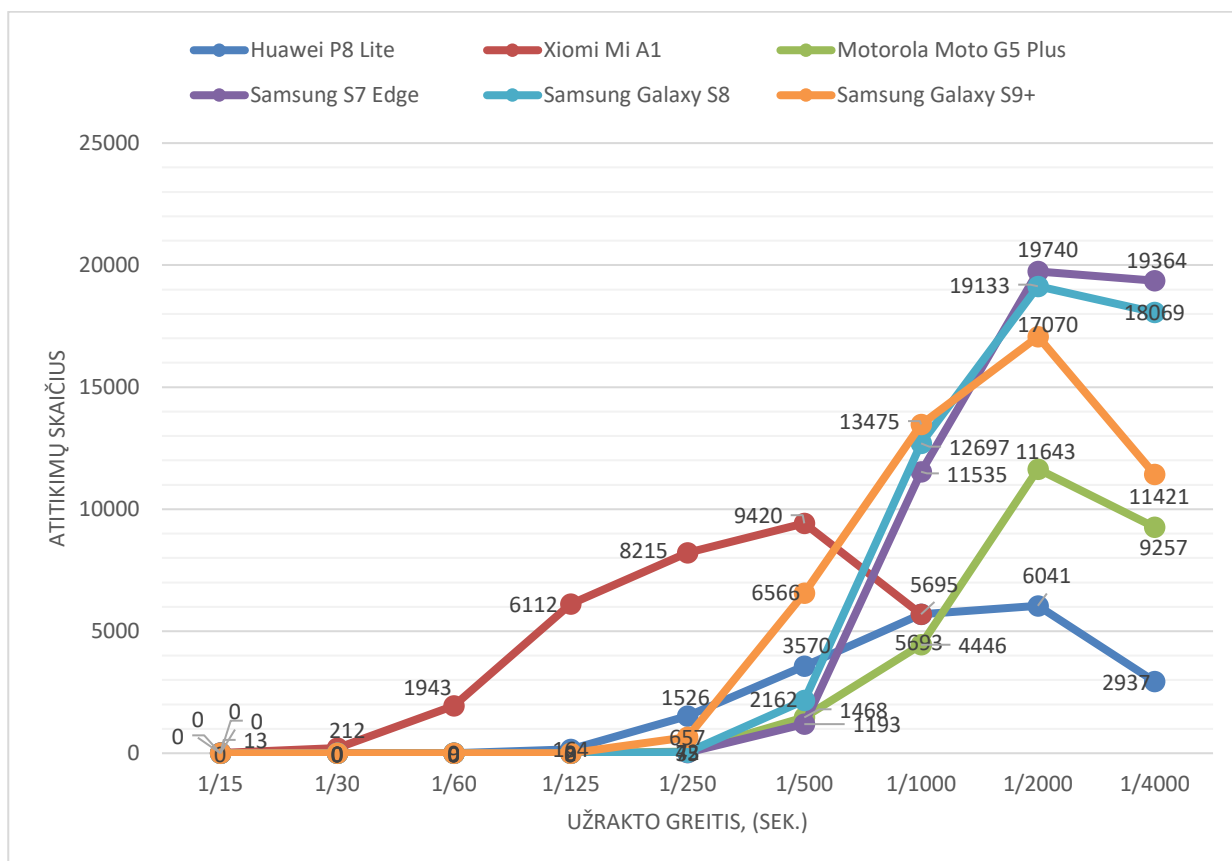
pav. 4.1 ORB detektoriaus ir aprašo rezultatai

Iš šio detektoriaus ir aprašo derinio matyti, kad daugiausiai atitikimų tarp etaloninio ir tiriamojo atvaizdo, aptinkama užrakto greičio kitimo režiuose tarp 1/500 sek. ir 1/2000 sek. Taip pat iš grafiko matyti, kad naudojant algoritmą ORB daugiausiai atitikimų surinko *Samsung Galaxy S9+* ir *Samsung Galaxy S8* mobiliųjų įrenginių vaizdo jutikliai. Jie surenka 32% daugiau charakteringų taškų negu kiti mobiliųjų įrenginių vaizdo jutikliai. Taip pat grafike ryškiai matoma *Xiaomi Mi A1* mobiliojo įrenginio vaizdo jutiklio atitikimų skaičiaus atitrūkimas nuo kitų mobiliųjų įrenginių. Iš grafiko galima spręsti, kad minėto įrenginio vaizdo jutikliui yra reikalingas ilgesnis laiko tarpas, per kurį šviesa yra fiksuojama, taip yra pasiekiami panašūs rezultatai, kaip kituose įrenginiuose. Galime daryti prielaidą, kad vaizdo jutiklis šviesą apdoroja kitaip negu kiti eksperimente naudoti įrenginiai.



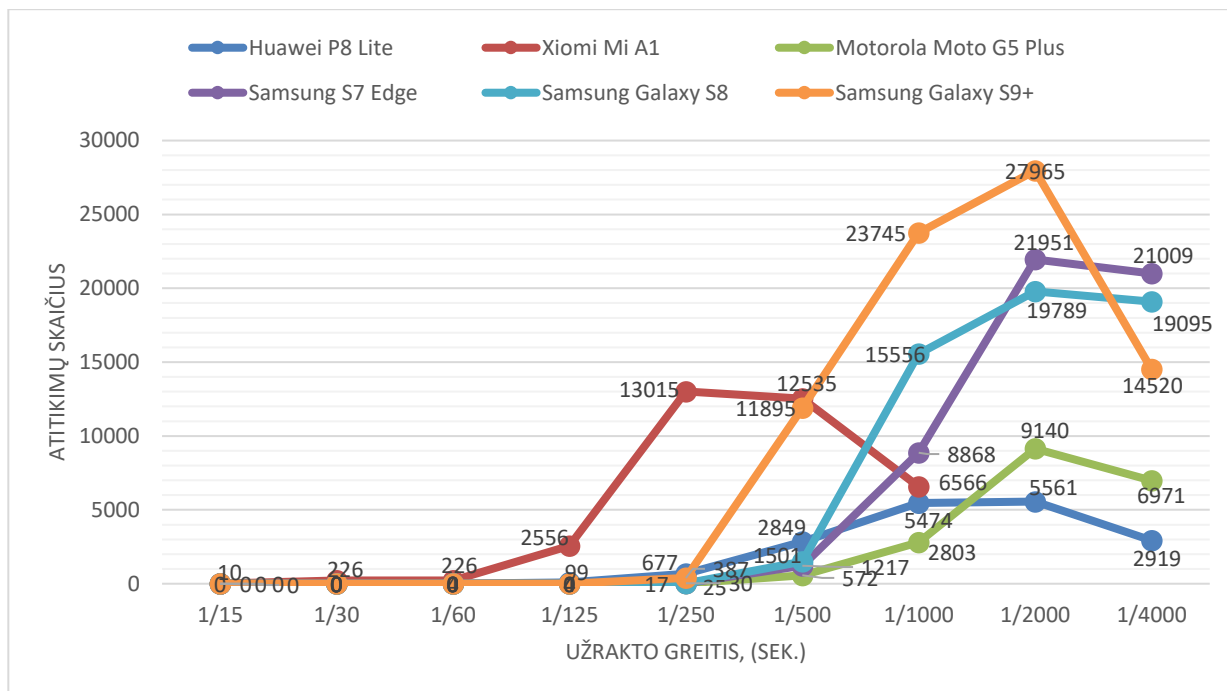
pav. 4.2 SURF detektoriaus ir aprašo rezultatai

SURF detektoriaus ir aprašo, rezultatų pasiskirstymas yra labai panašus į *ORB* detektoriaus ir aprašo kombinacijos. Tačiau verta pastebėti, kad charakteringų taškų sutapimų yra 75% mažiau negu *ORB* metode. Taip yra dėl kitokio detektoriaus ir aprašo veikimo principo, kad būtų surastas ir aprašytas taškas yra atliekama daugiau skaičiavimų, dėl to prailgėja ir sistemos atsako laikas. Didžiausią atitikimų skaičių pateikė *Samsung Galaxy S9+* ir *Samsung Galaxy S8* mobiliųjų įrenginių vaizdo jutikliai. Grafike taip pat matyti *Xiomi Mi A1* įrenginio duomenų reikšmių atsiplėšimas nuo kitų įrenginių.



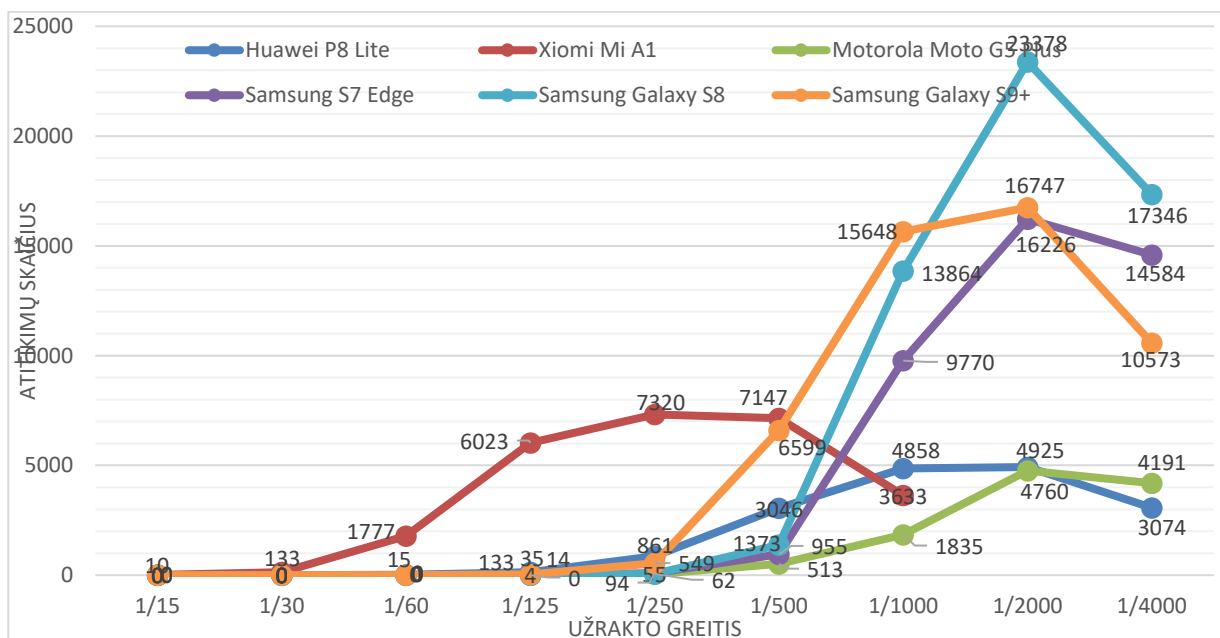
pav. 4.3 AKAZE detektoriaus ir aprašo rezultatai

AKAZE detektoriaus ir aprašo rezultatai pateikia kiek kitus dominuojančius mobiliųjų įrenginių jutiklius. Iš grafiko matyti, kad daugiausiai atitikimų pateikė *Samsung Galaxy S7 Edge* ir *Samsung Galaxy S8* įrenginių vaizdo jutikliai. Šiame grafike pastebimas *Samsung* įrenginių jutiklių pranašumas prieš kitus eksperimente dalyvavusius įrenginius. *Samsung* įrenginiai pateikia vidutiniškai 20% daugiau atitikimų negu kiti įrenginiai. Kaip ir kituose jau aptartuose bandymuose taip ir čia matyti kitokia *Xiomi Mi A1* įrenginio vaizdo jutiklio priklausomybė nuo užrakto greičio.



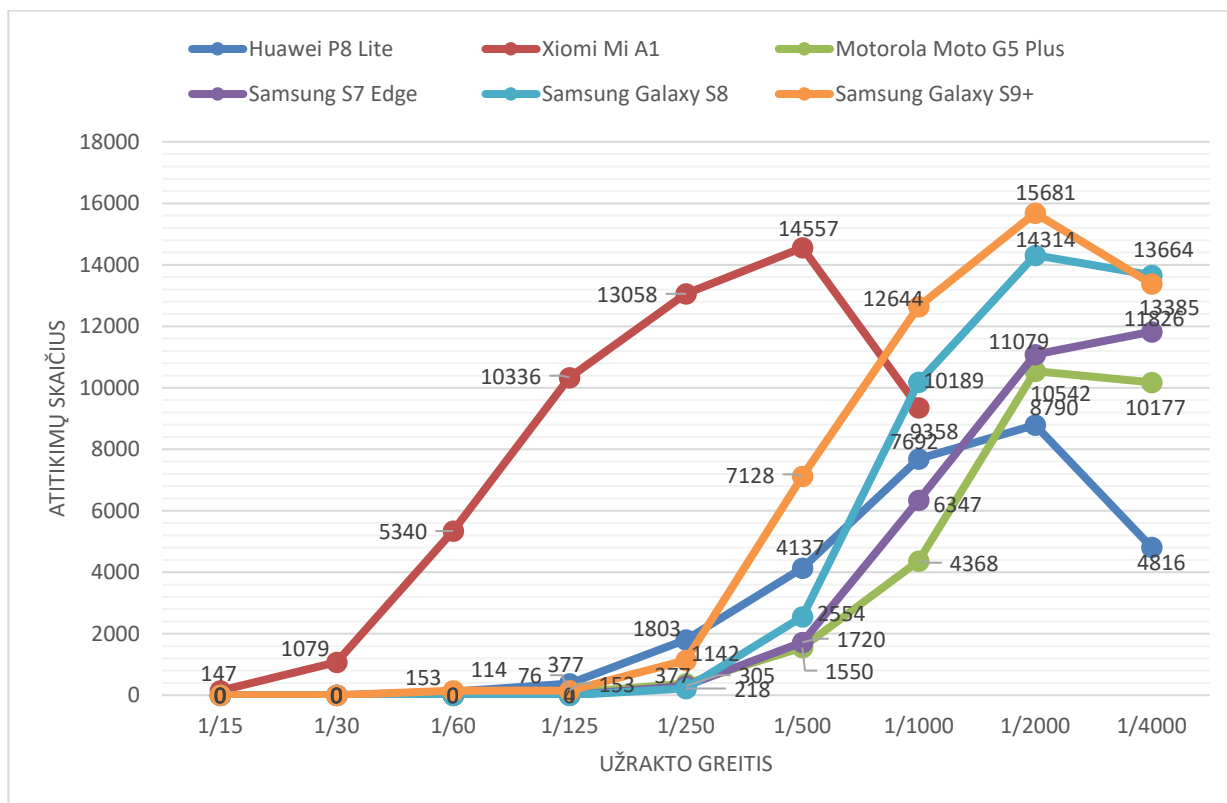
pav. 4.4 BRISK detektoriaus ir aprašo rezultatai

BRISK detektoriaus ir aprašo rezultatuose matyti, *Samsung Galaxy S9+* įrenginio pranašumas, lyginant su kitais jutiklių rezultatais minėtasis įrenginys pateikia 28% daugiau atitikimų negu kiti įrenginiai. Taip pat pagal gautus duomenis galime teigti, kad *BRISK* detektoriaus ir aprašo kombinacija veikia ne taip efektingai esant tamsesniems (1/4000 sek.) atvaizdams, nes pastebimas ženklus atitikimų sumažėjimas.



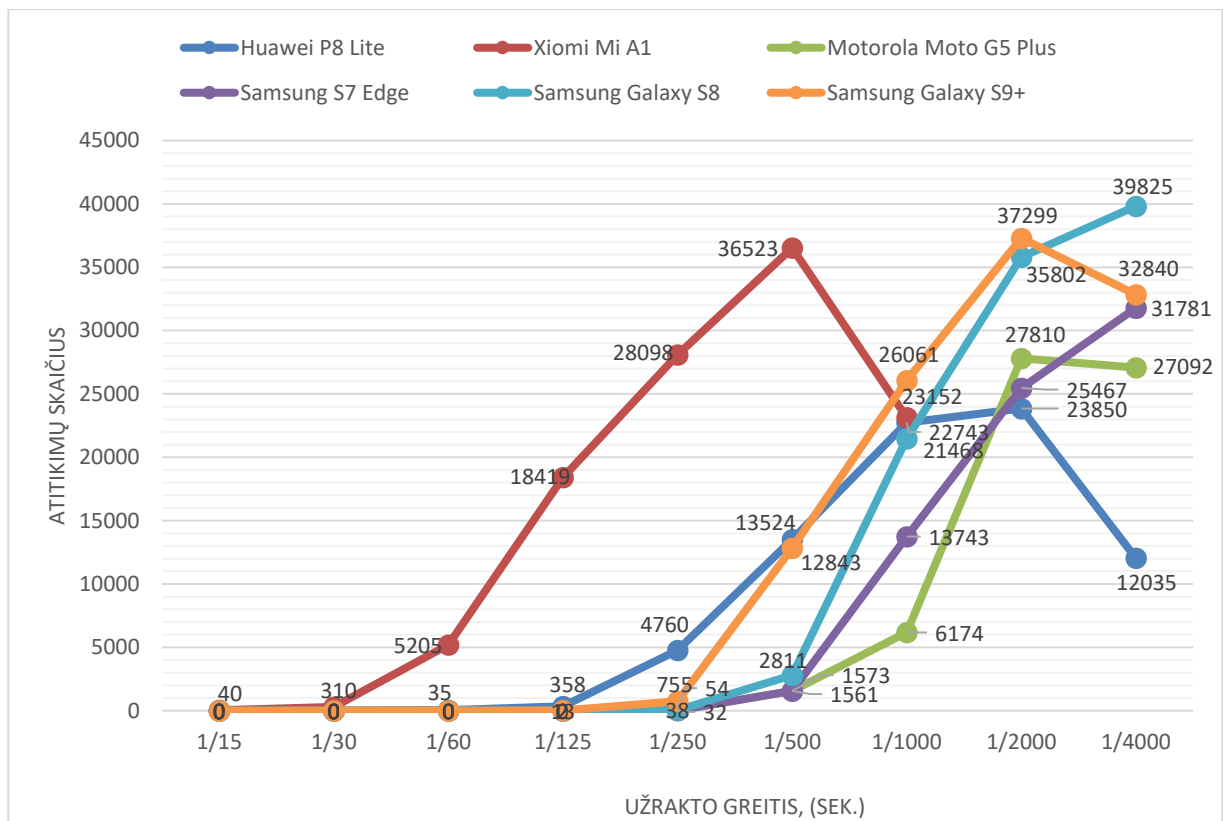
pav. 4.5 SIFT detektoriaus ir aprašo rezultatai

SIFT detektoriaus ir aprašo rezultatų grafike matyti aiškus *Samsung Galaxy S8* mobiliojo įrenginio jutiklio pranašumas prieš kitus įrenginius. Minėtas įrenginio jutiklis pateikia 27% daugiau atitikimų, negu artimiausi jam rezultatai. Kaip ir *AKAZE* detektoriaus ir aprašo rezultatuose, taip ir čia matomas *Samsung* įrenginių pranašumas prieš kitus įrenginius. *AKAZE* metodo pateikiamuose atitikimų rezultatuose visi *Samsung* modelio įrenginiai surinko daugiau negu 15000 atitikimų, kai tuo tarpu kiti įrenginiai neperkopė 10000 atitikimų.



pav. 4.6 Detektoriaus *ORB* ir aprašo *BRIEF* rezultatai

Detektoriaus *ORB* ir aprašo *BRIEF* pateikiamuose atitikimų rezultatuose matyti panašus duomenų pasiskirstymas tarp visų įrenginių, tačiau *Xiomi Mi A1* įrenginio rezultatai priklauso nuo užrakto greičio, efektyviausiai detektoriaus *ORB* ir aprašo *BRIEF* kombinacija su *Xiomi Mi A1* įrenginio jutikliu veikia 1/125 sek. ir 1/500 sek. užrakto greičio režimuose. Tačiau daugiausiai atitikimų pateikia *Samsung Galaxy S9+* įrenginio jutiklis.



pav. 4.7 Detektoriaus *ORB* ir aprašo *FREAK* rezultatai

Detektoriaus *ORB* ir aprašo *FREAK* rezultatų grafike matyti, kad *Samsung Galaxy S8* įrenginio vaizdo jutiklis pateikia didžiausią atitikimų skaičių. Visų įrenginių atitikimų rezultatai turi panašų pasiskirstymą, kaip ir ankstesniuose detektoriaus ir aprašo kombinacijų tyrimuose. Tačiau *Xiomi Mi A1* įrenginio vaizdo jutiklis nors ir ženkliai nenusileidžia kitiems įrenginiams atitikimų skaičiumi, bet išsiskiria kitokia priklausomybe nuo užrakto greičio.

4.2.4 Rezultatų aptarimas

Taikant skirtingus charakteringų taškų aptikimo ir išskyrimo algoritmus pastebėjome tendenciją, kad kiekvieno iš algoritmų rezultatų pasiskirstymas yra gana panašus. Iš gautų rezultatų grafikų matosi, kad tirtu atveju charakteringų taškų aptikimas priklauso nuo mobiliųjų įrenginių vaizdo jutiklių kokybės. *Samsung* modelio mobilieji įrenginiai, tirtu atveju 19% lenkia arčiausiai charakteringų taškų atitikimų skaičiumi esantį *Motorola Moto G5 Plus* mobiliųjų įrenginį. Tuo tarpu mažiau charakteringų taškų surinkę įrenginiai, tirtu atveju, imant 1/250 sek. užrakto greičio žingsnį išskiria 34% daugiau charakteringų taškų atitikimų (*Huawei P8 lite*). Taip reikia pažymėti, kad *Xiomi Mi A1* mobilusis įrenginys charakteringų taškų atitikimus pradeda išskirti esant 1/30 sek. – 1/500 sek. užrakto greičio režiuose, tai reiškia, kad šio prietaiso vaizdo jutikliui užfiksuoti atvaizdą tinkamą charakteringų taškų paieškai reikalinga daugiau laiko. Manome, kad tai atsitinka dėl to, kad šio prietaiso vaizdo jutiklis fiksuojant atvaizdą automatiniais parametrais vykdo papildomus vaizdo apdorojimus.

4.3 Charakteringų taškų aptikimo priklausomybės nuo įrenginio vaizdo jutiklio jautrumo parametrų eksperimentas

4.3.1 Tikslas

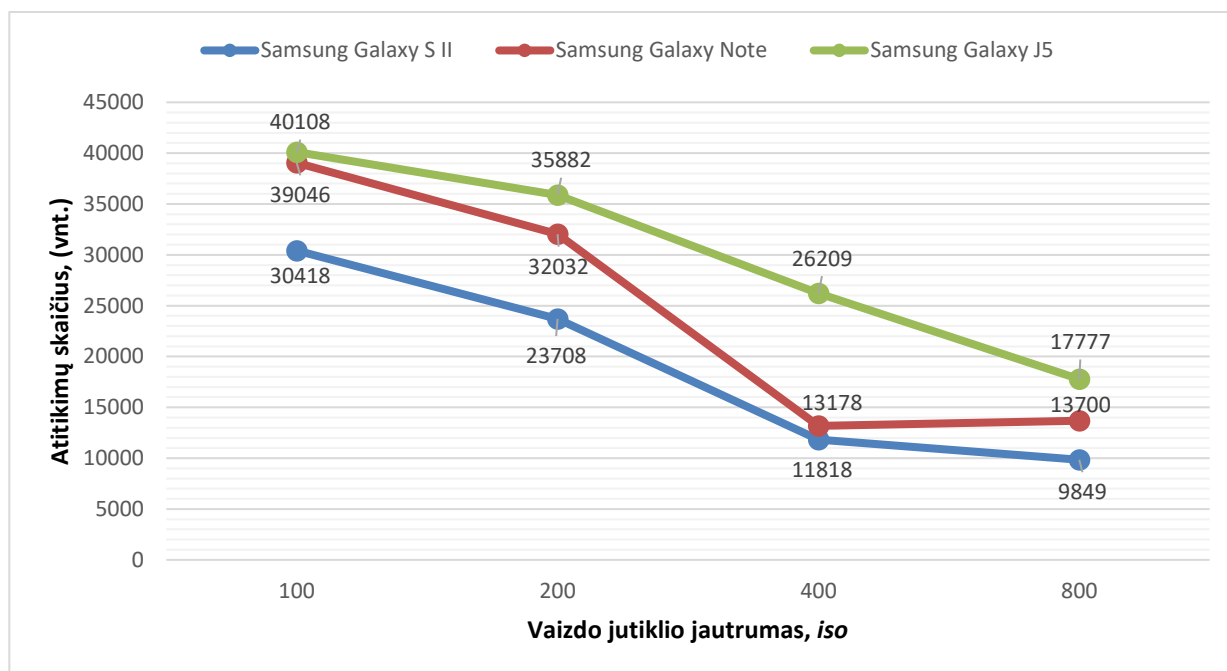
Šio eksperimento tikslas nustatyti vaizdų atpažinimo išskiriamų charakteringų taškų priklausomybę nuo mobiliųjų įrenginių vaizdo jutiklio jautrumo parametrų/parodymų.

4.3.2 Metodika

Tiriant charakteringų taškų priklausomybę nuo mobiliųjų įrenginių vaizdo jutiklio jautrumo parodymų yra užfiksuojama atvaizdas su automatiniais vaizdo jutiklio jautrumo parametrais, taip pat fiksuojami atvaizdai su skirtingais jutiklio jautrumo parametrais, šiuo atveju t.y. ISO vertės: 100, 200, 400 ir 800. Toliau atvaizdai yra analizuojami su sukurta sistema, aptinkami ir aprašomi charakteringi taškai atvaizde su automatiniais parametrais (etaloninis atvaizdas). Tada etaloninio atvaizdo charakteringi taškai yra lyginami atskirai su kiekvieno atvaizdo aptiktais ir aprašytais charakteringais taškais. Tokiu būdu gaunama charakteringų taškų priklausomybė nuo skirtingų vaizdo jutiklio jautrumo parametrų.

4.3.3 Eksperimento rezultatai

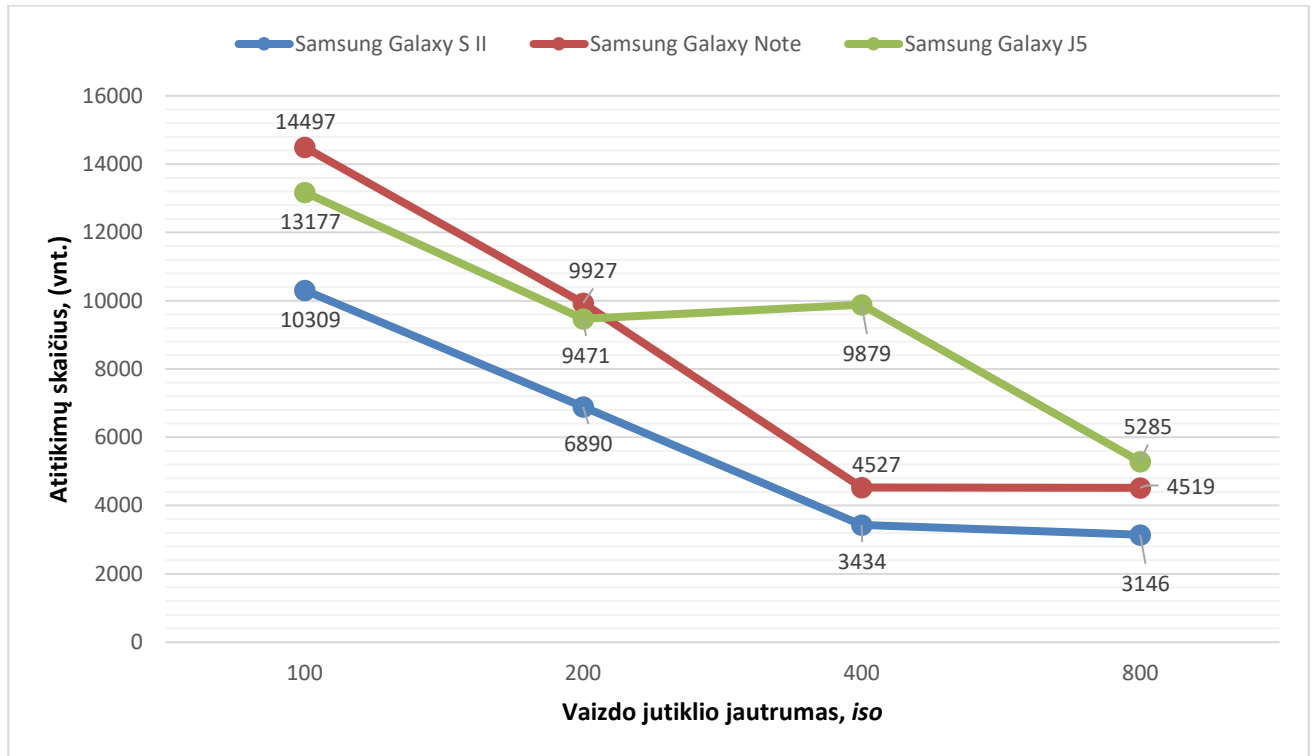
Pateikiami eksperimento rezultatai:



pav. 4.8 Detektoriaus ORB ir aprašo FREAK rezultatai

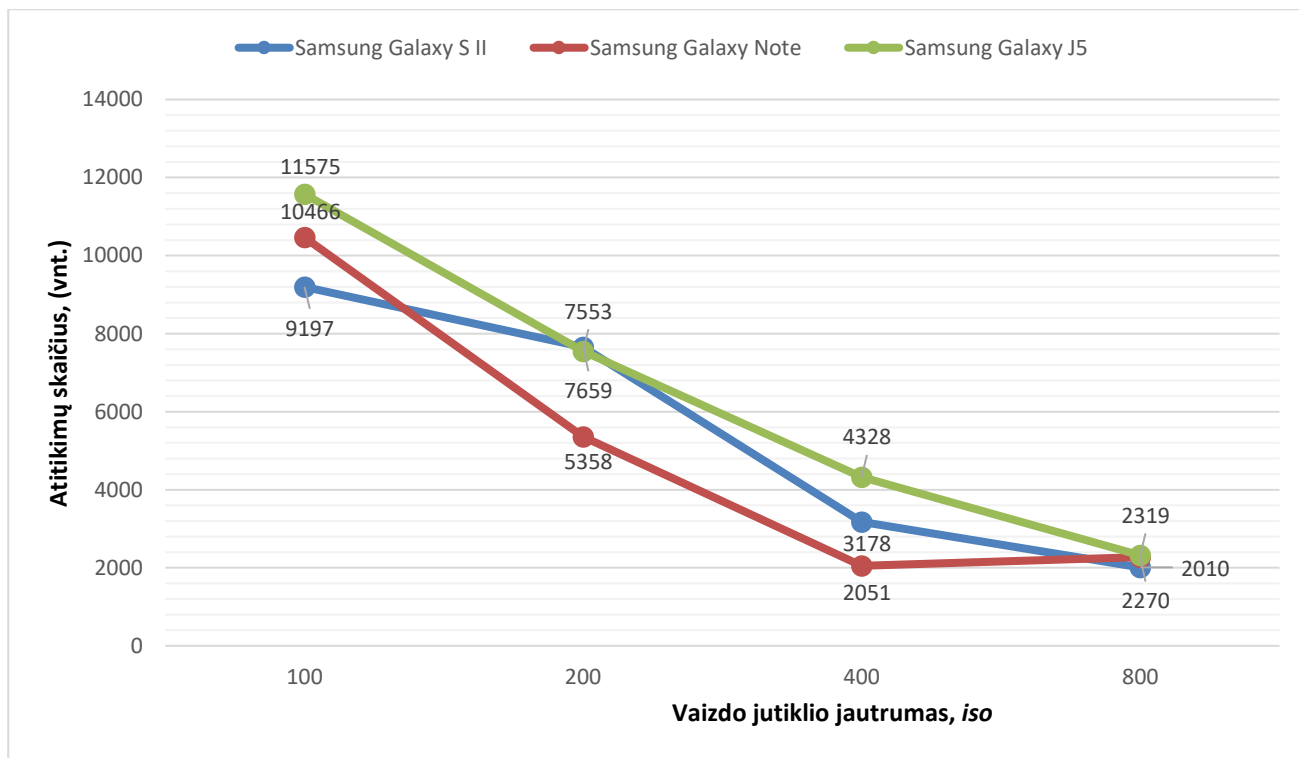
Pagal gautą grafiką matyti, kad didėjant įrenginio jutiklio jautrumui charakteringų taškų atitikimų skaičius mažėja. Daugiausiai charakteringų taškų atitikimų išskiria mobilusis įrenginys *Samsung Galaxy J5*. Tačiau nepaisant to, kad šio įrenginio jutiklio parametrų derinys išskyrė daugiausiai taškų

atitikimų, imant visą vaizdo jutiklio jautrumo reikšmių imtį nuo 100 iki 800, jutiklio išskiriamų atitikimų skaičius sumažėjo net 55%. Minėto įrenginio jutiklio išskiriamų atitikimų kiekis kitimas stabilesnis negu kitų dviejų įrenginių. Tuo tarpu likusių dviejų įrenginių *Samsung Galaxy Note* ir *Samsung Galaxy S II* jutiklių jautrumo parametrui esant 400, atitikimų reikšmės smarkiai krenta žemyn. Tiesa *Samsung Galaxy Note* įrenginys esant jautrumui 800 pateikia 4% daugiau taškų negu esant jautrumui 400.



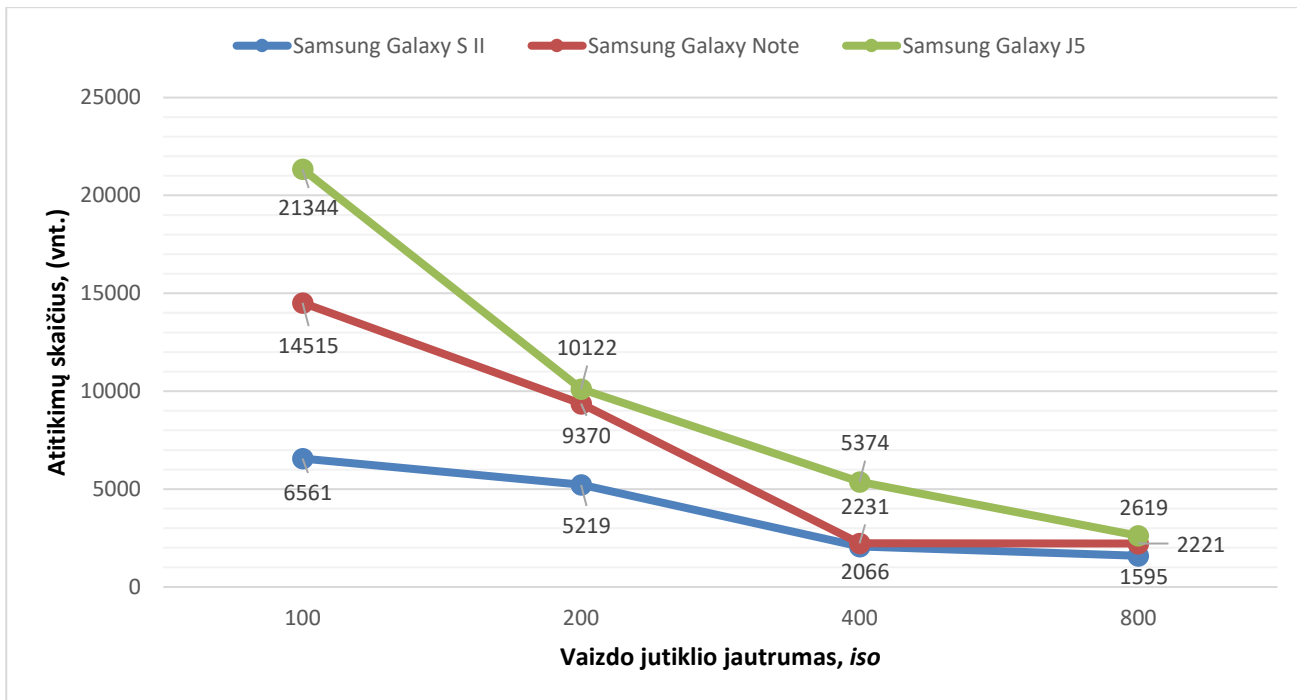
pav. 4.9 Detektoriaus *ORB* ir aprašo *BRIEF* rezultatai

Šiame grafike matyti, kad daugiausiai atitikimų mobiliųjų įrenginių jutikliai patiekia jutiklio jautrumo parametro reikšmei esant 100. *Samsung Galaxy Note* įrenginio jutiklis pateikia daugiausiai atitikimų jautrumo režiuose nuo 100 iki 200. Toliau grafike galima pastebėti staigų minėto įrenginio atitikimų skaičiaus kritimą žemyn esant jautrumo reikšmei 400, tuo tarpu *Samsung Galaxy J5* atitikimų skaičius yra 54% didesnis negu *Samsung Galaxy Note*. Tuo tarpu *Samsung Galaxy S II* įrenginio vaizdo jutiklio pateikiami rezultatai yra mažiausi šiame detektoriaus ir aprašo bandyme.



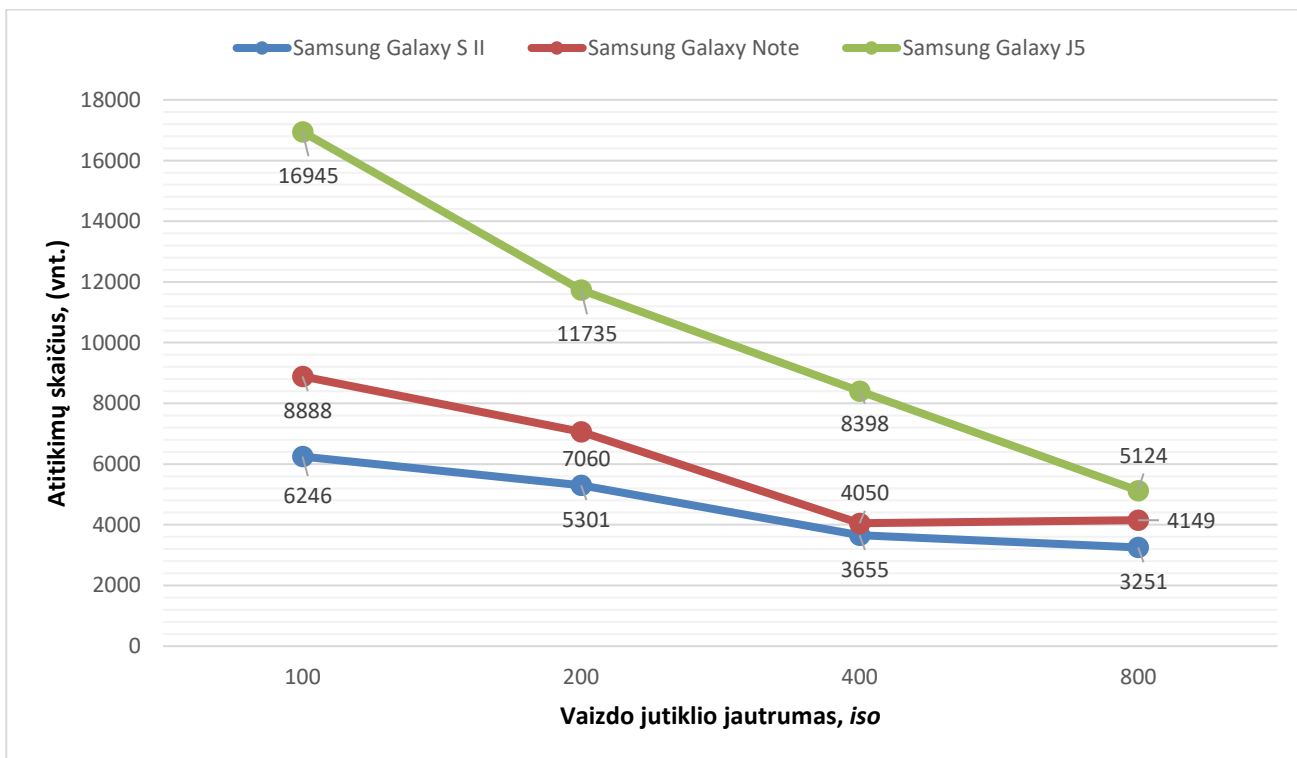
pav. 4.10 Detektoriaus ir aprašo *SIFT* rezultatai

Detektoriaus ir aprašo *SIFT* grafike matyti gana panašūs įrenginių vaizdo jutiklių charakteringų taškų atitikimų rezultatai. Jutiklio jautrumo reikšmei esant 800, visų įrenginių atitikimų skaičius yra panašus. Pagal grafike pateiktus duomenis galima teigti, kad daugiausiai charakteringų taškų atitikimų pateikia *Samsung Galaxy J5* įrenginio jutiklis, tačiau esant jautrumo reikšmei 200, *Samsung Galaxy S II* atitikimų pateikia 1,4% daugiau negu *Samsung Galaxy J5*. Taip pat galima pastebėti, kad *Samsung Galaxy Note* įrenginio charakteringų taškų atitikimų skaičius, esant jautrumo reikšmių nuo 400 iki 800 rėžyje, didėja. Kai tuo tarpu likusių dviejų įrenginių atitikimų skaičius mažėja.



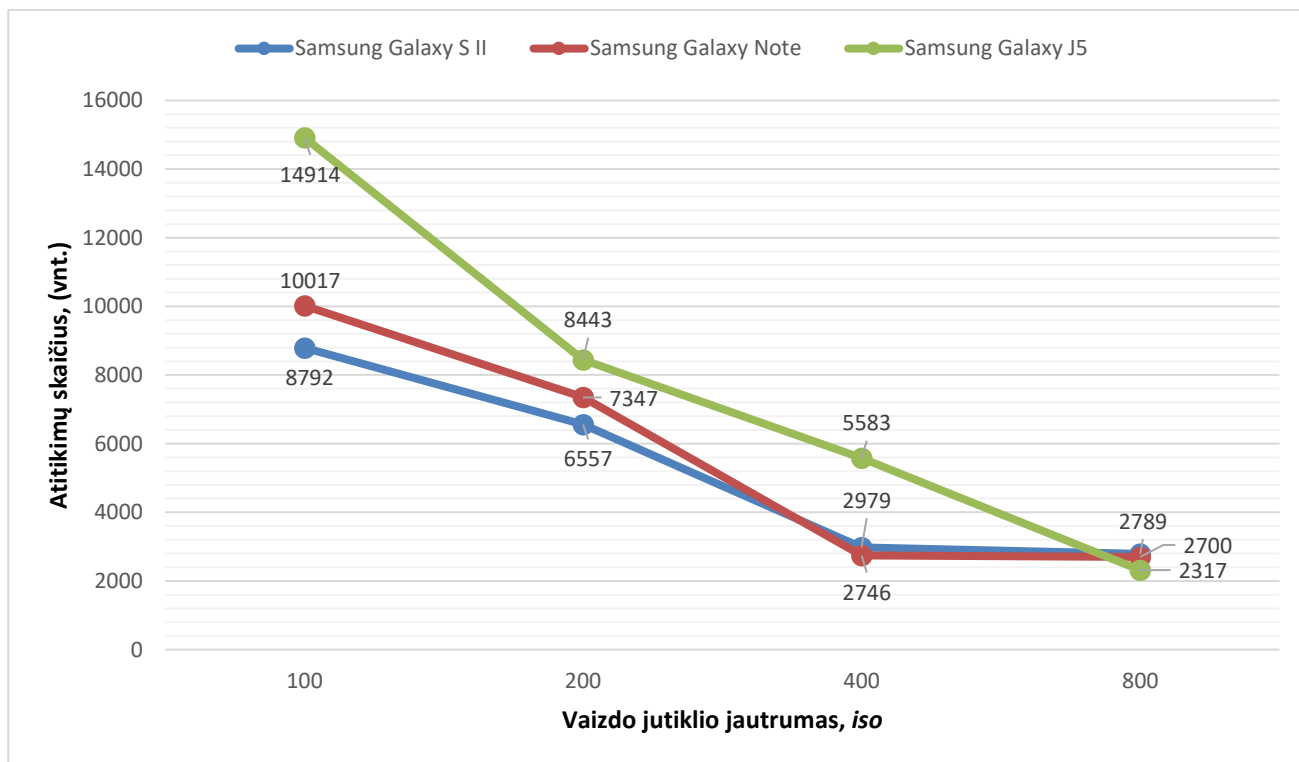
pav. 4.11 Detektoriaus ir aprašo *BRISK* rezultatai

Šio bandymo grafike matyti, kad įrenginio *Samsung Galaxy J5* rezultatai esantys jautrumo reikšmei 100, yra 32% didesni už artimiausio įrenginio esančio. Visų įrenginių pateikiami rezultatai yra panašūs esant jautrumo reikšmei 800. Taip pat galima pastebėti, kad įrenginio *Samsung Galaxy J5* atitikimų skaičiaus rezultatai esant vis kitai jutiklio jautrumo reikšmei mažėja per pus.



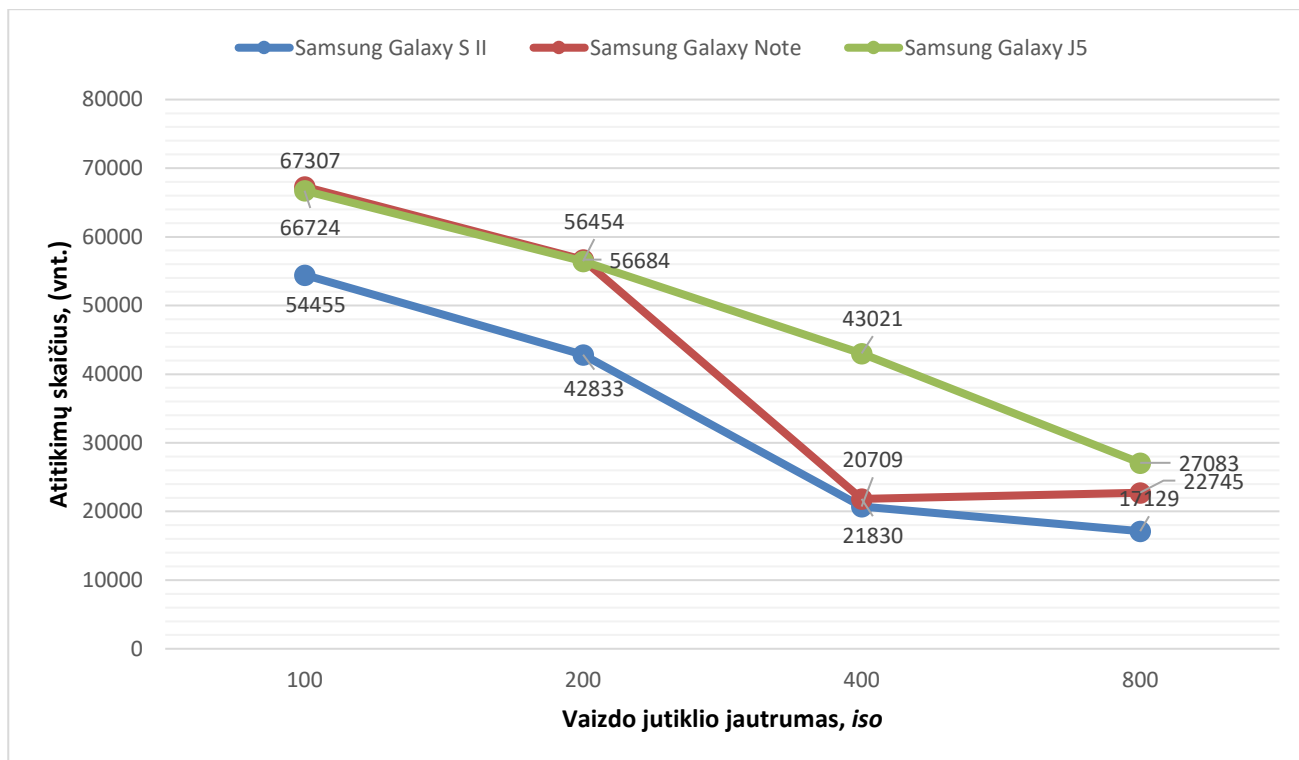
pav. 4.12 Detektoriaus ir aprašo *AKAZE* rezultatai

Iš šio grafiko galime matyti, kad *Samsung Galaxy J5* įrenginio vaizdo jutiklio pateikiamas charakteringų taškų atitikimų kiekis vidutiniškai 39,5% didesnis už *Samsung Galaxy Note* įrenginio. Tuo tarpu *Samsung Galaxy S II* įrenginio pateikiami charakteringų taškų atitikimai yra vidutiniškai 53,5 % mažesni už *Samsung Galaxy J5* įrenginio pateikiamus rezultatus.



pav. 4.13 Detektoriaus ir aprašo *SURF* rezultatai

Detektoriaus ir aprašo *SURF* pateikiami rezultatai parodo, kad esant jautrumo reikšmei 100, daugiausiai charakteringų taškų atitikimų pateikė *Samsung Galaxy J5* mobiliojo įrenginio vaizdo jutiklis. Tačiau kiek netikėtai esant jautrumo reikšmei 800, *Samsung Galaxy S II* įrenginio vaizdo jutiklis pateikė 17% daugiau charakteringų taškų atitikimų negu *Samsung Galaxy J5* įrenginys. Tuo tarpu *Samsung Galaxy Note* jutiklis jautrumo režimuose nuo 400 iki 800 pateikia 4% mažiau atitikimų negu *Samsung Galaxy S II* įrenginys.



pav. 4.14 Detektoriaus ir aprašo ORB rezultatai

Detektoriaus ir aprašo ORB bandyme jautrumo režiuose nuo 100 iki 200 mobiliųjų įrenginių *Samsung Galaxy Note* ir *Samsung Galaxy J5* jutiklių pateikiami charakteringų taškų atitikimai tirtu atveju, skiriasi tik 0,8%, tačiau jautrumo reikšmei esant 400, *Samsung Galaxy Note* įrenginio vaizdo jutiklio charakteringų taškų atitikimų skaičius, tirtu atveju yra 51% mažesnis už *Samsung Galaxy J5* įrenginio. Tuo tarpu *Samsung Galaxy S II* įrenginys viso bandymo jautrumo režiuose pateikia vidutiniškai 32% mažiau taškų atitikimų.

4.3.4 Rezultatų aptarimas

Tirtu atveju eksperimento rezultatai parodė, kad didėjant vaizdo jutiklio jautrumui, mažėja charakteringų taškų atitikimų skaičius. Galima daryti prielaidą, kad kai yra keičiamos vaizdo jutiklio jautrumo reikšmės, yra stiprinamas vaizdo signalas, o kartu ir visi triukšmai esantys signale. Atliekant eksperimentą, keičiant jautrumo reikšmes galima vizualiai matyti atvaizde esančius triukšmus, kurie apsunkina charakteringo taško išskyrimą, dėl to mažėja tiek jų, tiek atitikimų kiekis. Paėmus tirtu atvejo mažiausią ir didžiausią vaizdo jutiklio jautrumo reikšmes, charakteringų taškų atitikimų skaičius sumažėja ~72% esant didžiausiai jautrumo reikšmei.

4.4 Mobilųjų įrenginių vaizdo jutiklio triukšmo įtakos, charakteringų taškų aptikimui, mažinimo eksperimentas

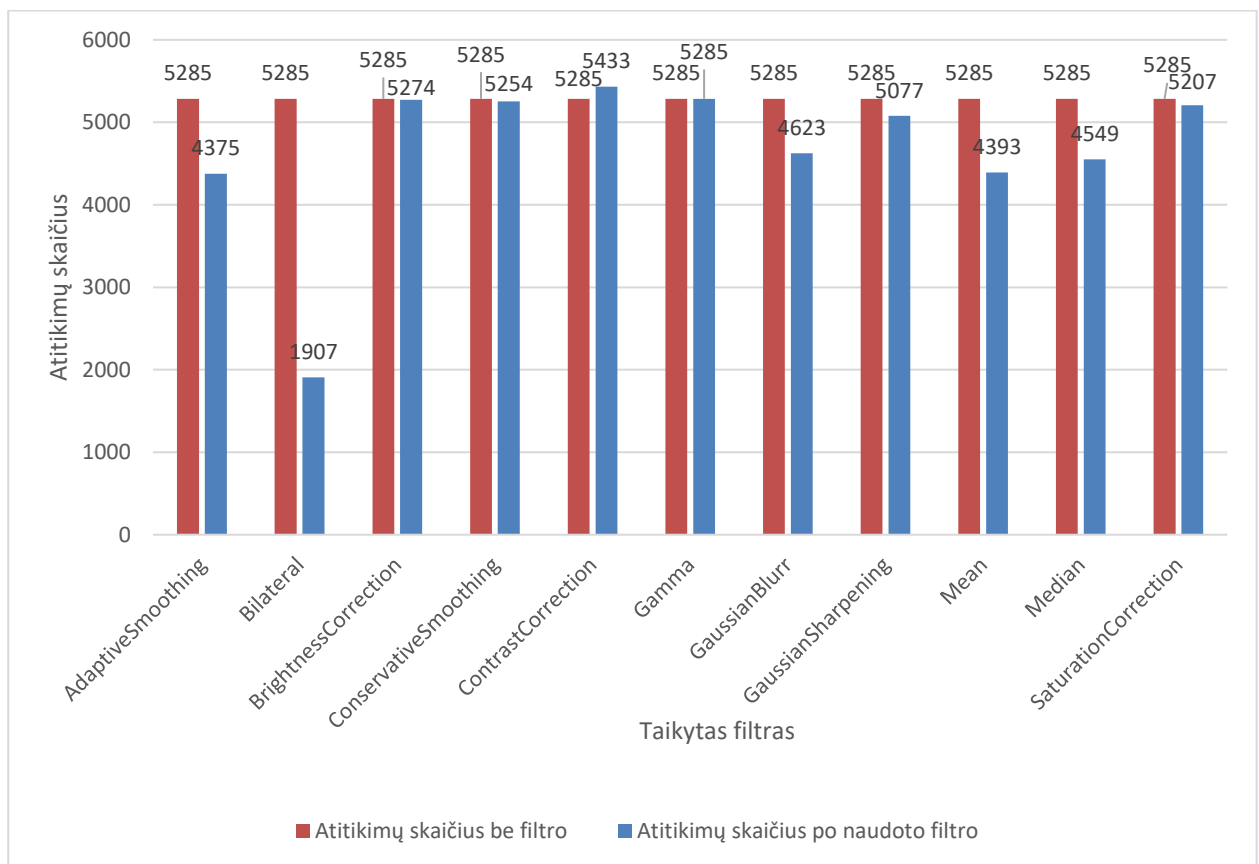
4.4.1 Tikslas

Šio eksperimento tikslas nustatyti ar naudojant atvaizdų apdorojimo filtrus galima sumažinti vaizdo jutiklio jautrumo sukiamų triukšmų įtaką charakteringų taškų aptikimui.

4.4.2 Metodika

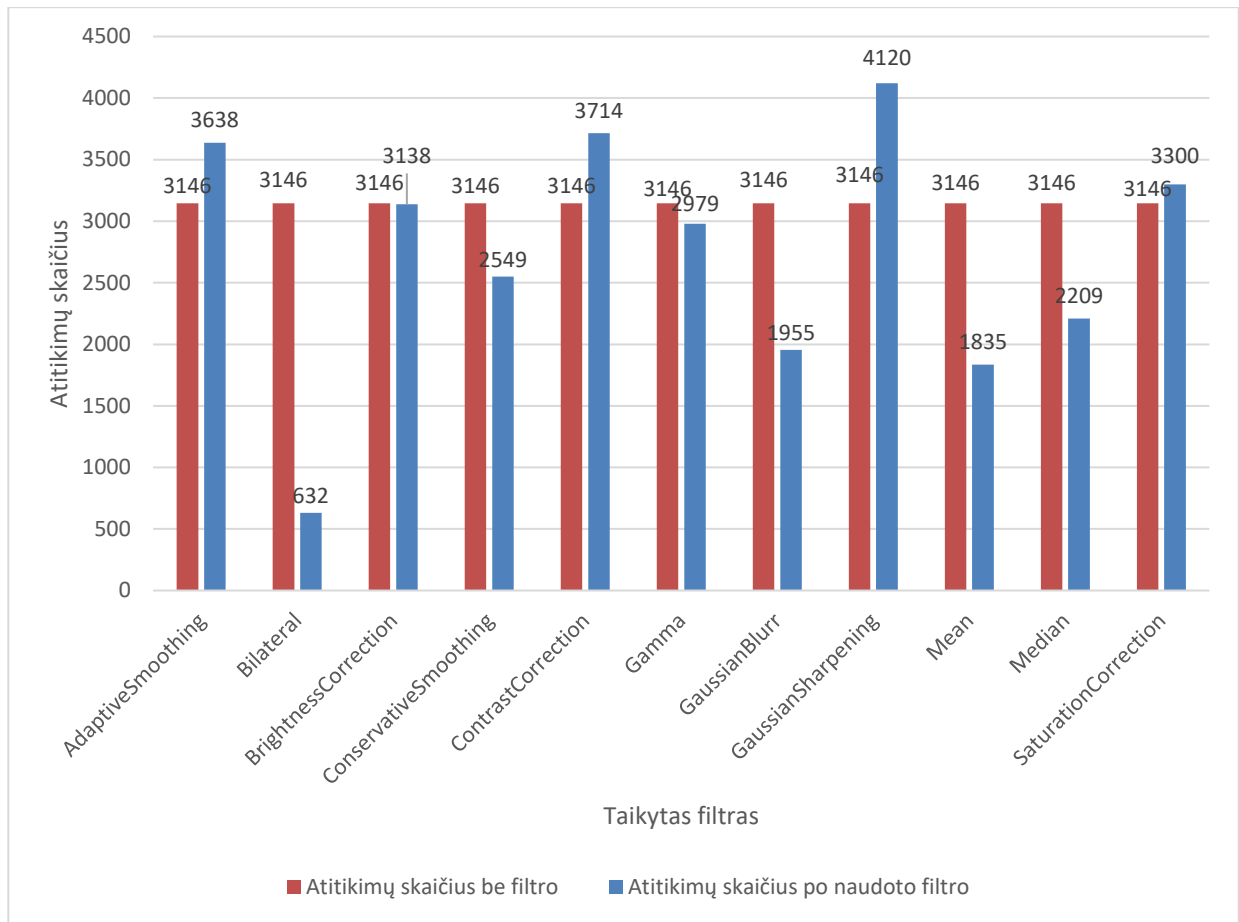
Atlikus charakteringų taškų priklausomybės nuo vaizdo jutiklio jautrumo eksperimentą (4.3), pastebėta, kad didėjant jutiklio jautrumui, mažėja charakteringų taškų skaičius, taip pastebimi skaitmeniniai triukšmai atvaizde. Pasirinkus atvaizdą su kuriuo etaloninis atvaizdas turi mažiausiai atitikimų, yra pritaikomi skirtingi vaizdų apdorojimo filtrai, kurie gali būti išskaidyti į dvi grupes: atvaizdų glotninimo filtrai ir paprastieji spalvų korekcijos filtrai. Pritaikant kiekvieną iš filtrų yra fiksuojama kaip kinta charakteringų taškų priklausomybė nuo mobiliojo įrenginio vaizdo jutiklio jautrumo parametrų po atvaizdo filtravimo.

4.4.3 Eksperimento rezultatai



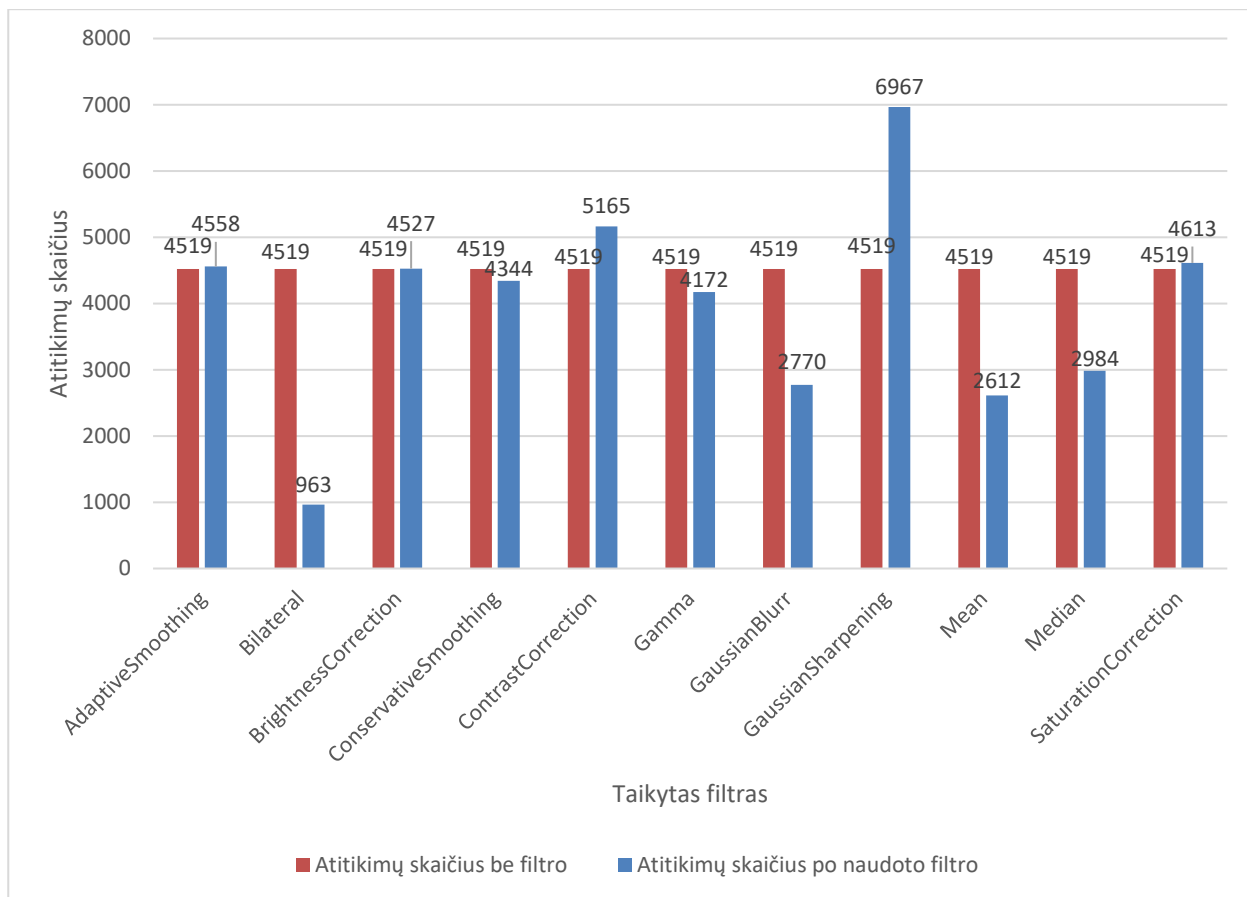
pav. 4.15 Mobiliojo įrenginio *Smasing Galaxy S II* vaizdo jutiklio keliamų triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai

Šiame grafike atvaizduojami charakteringų taškų atitikimai prieš pritaikant atvaizdų apdorojimo filtrą (raudoni stulpeliai) ir po atvaizdų apdorojimo filtro pritaikymo (mėlyni stulpeliai). Iš grafiko matyti, kad taikant kontrasto korekcijos (angl. *contrast correction*) filtrą yra gaunama daugiau charakteringų taškų atitikimų. Galima daryti prielaidą, kad pritaikius šį filtrą yra aptinkama daugiau charakteringų taškų, nes yra paryškinamos kraštinės, kurios iki tol nebuvo tokios ryškios. Taip pat galime matyti, kad taikant dvišalio, vidurkio, prisitaikantį ir medianos glotninimo filtrus charakteringų taškų atitikimų rezultatas skiriasi vidutiniškai 29,5%, tirtu atveju.



pav. 4.16 Mobiliojo įrenginio *Smasung Galaxy Note* triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai

Šioje diagramoje galime aiškiai pastebėti, kad naudojant prisitaikantį glotninimo filtrą charakteringų taškų atitikimų išskiriama daugiau. Taip pat verta pažymėti, kad kontrasto ir spalvų sodrumo korekcijos taip pat pateikia aukštesnius rezultatus, kontrasto korekcijos filtras 15% daugiau charakteringų taškų atitikimų ir 5% daugiau atitikimų naudojant spalvų sodrumo filtrą. Tačiau didžiausią atitikimų padidėjimą pateikia *Gauso* aštrinimo filtras, pateikiama 23% daugiau atitikimų, tirtu atveju. Tuo tarpu didžiausias atitikimų sumažėjimas matomas taikant dvišalį glotninimo filtrą, atitikimų po filtro panaudojimo, tirtu atveju sumažėja 80%.



pav. 4.17 Mobiliojo įrenginio *Smasung Galaxy J5* vaizdo jutiklio keliamų triukšmų filtravimo atvaizdų apdorojimo filtrais rezultatai

Šioje diagramoje matoma aiškia *Gauso* aštravimo filtro įtaka charakteringų taškų atitikimų skaičiui, pritaikius šį filtrą atitikimų išskiriama 35% daugiau. Taip pat verta pažymėti, kad taikant šviesumo, kontrasto ir spalvų sodrumo korekcijos filtras atitikimų skaičius taip pat padidėja vidutiniškai apie 4% daugiau charakteringų taškų atitikimų. Mažiausiai atitikimų pateikia dvišalio, vidurkio ir medianos filtrų naudojimas, tirtu atveju vidutiniškai atitikimų skaičius sumažėja 56%.

4.4.4 Rezultatų aptarimas

Iš pateiktų ir aptartų rezultatų diagramų galime teigti, kad naudojant tiesinius vaizdų apdorojimo filtras: *Gauso* aštrinimas; šviesumo, kontrasto ir spalvų sodrumo korekcija galima aptikti daugiau atitikmenų. Galima daryti prielaidą, kad po filtrų panaudojimo atvaizdų charakteringų taškų aptikimas tampa efektyvesnis. Pagal rezultatus, galime matyti, kad visi eksperimente naudoti mobilieji įrenginiai pritaikius vieną iš filtrų pateikia daugiau charakteringų taškų atitikimų.

Taip pat iš diagramų galime teigti, kad naudojant filtras, skirtus atvaizdų glotninimui gauti, rezultatai buvo ženkliai prastesni už etaloninį atvaizdą. Didžiausias charakteringų taškų atitikimų skaičiaus sumažėjimas pastebimas naudojant dvišalį glotninimo filtrą, po kurio panaudojimo atitikimų skaičius, tirtu atveju sumažėjo vidutiniškai 79%.

5 IŠVADOS

Atlikus eksperimentus galima teigti, kad:

1. Atlikus charakteringų taškų priklausomybės nuo mobiliojo įrenginio eksperimentą pastebėta, kad naudojant *Samsung Galaxy S9+*, *Samsung Galaxy S8* ir *Samsung Galaxy S7 edge* modelio mobiliuosius įrenginius, tirtu atveju charakteringų taškų atitikimų išskiriama vidutiniškai 19% daugiau negu *Huawei P8 lite*, *Xiaomi Mi A1*, *Motorola Moto G5 Plus*. Nors pastebėta tendencija, kad charakteringų taškų aptikimas priklauso nuo vaizdo jutiklio naujumo, tačiau yra atvejų kai daugiau rezultatų pateikia ir senesnių mobiliųjų įrenginių vaizdo jutikliai. Iš pav. 4.3, galime matyti, kad bandymo metu 13 % daugiau rezultatų pateikia *Samsung Galaxy S7 edge* mobilusis įrenginys;
2. Atlikus charakteringų taškų priklausomybės nuo vaizdo jutiklio jautrumo parametru eksperimentą, pastebime aiškų charakteringų taškų mažėjimą, didėjant vaizdo jutiklio jautrumo reikšmėms. Galima manyti, kad tirtu atveju, didinant jutiklio jautrumo reikšmes išryškėja triukšmas esantis atvaizde, dėl to tampa sudėtingiau aptikti charakteringus taškus. Tirtu atveju daugiausiai atitikimų jutiklio jautrumo reikšmei esant 100, pateikė *Samsung Galaxy J5* mobiliojo įrenginio vaizdo jutiklis. Minėto įrenginio jutiklis pateikė vidutiniškai 18,79% daugiau charakteringų taškų atitikimų negu *Samsung Galaxy note* mobilusis įrenginys, taip pat 36,92% daugiau atitikimų negu *Samsung Galaxy S II* įrenginys. Nors bandymų metu pastebėta, kad didėjant vaizdo jutiklio jautrumo reikšmėms, charakteringų taškų atitikimų kiekis mažėja, tačiau pasitaiko atvejų, kai naudojant *Samsung Galaxy note* įrenginį, vaizdo jutiklio jautrumo režyje tarp 400 ir 800, taškų atitikimų padaugėja 6,3%. (žr. pav. 4.8, pav. 4.10, pav. 4.12, pav. 4.14);
3. Įvertinus antro eksperimento rezultatus pastebėta, kad atitikimų skaičius, didinant jutiklių jautrumo reikšmes nuo 100 iki 800, sumažėja vidutiniškai 59%. Manoma, kad toks mažėjimas pastebimas dėl skaitmeninių atvaizdo triukšmų, todėl yra taikomi tiesinio vaizdų apdorojimo filtrai siekiant sumažinti triukšmų įtaką charakteringų taškų aptikimui. Bandymų metu pastebėta, kad naudojant kontrasto korekcijos filtrą, atvaizdas su jutiklio jautrumo reikšme 800, pateikia 35,1% daugiau atitikimų su etaloniniu atvaizdu žr. pav. 4.17. Tačiau naudojant dvišalio suliejimo (angl. *Bilateral*) filtrą, tirtu atveju atitikimų skaičius vidutiniškai sumažėja 74,23% (žr. pav. 4.17). Atlikus eksperimentus ir įvertinus gautus rezultatus, galime teigti, kad tirtu atveju charakteringų taškų pasiskirstymas priklauso nuo vaizdo jutiklių parametru. O vaizdo jutiklių jautrumo parametro keliami skaitmeniniai triukšmai daro įtaką charakteringų taškų aptikimui. Norint sumažinti daromą įtaką yra reikalingas papildomas atvaizdų apdorojimas;

4. Išanalizavus mobiliųjų įrenginių jutiklius ir jų charakteristikas, buvo nuspręsta didžiausią dėmesį skirti vaizdo jutiklių parametrų įtakos vaizdų atpažinimui tyrimams;
5. Išanalizavus charakteringų taškų detektorius, aprašų kūrimo ir atitikmenų paieškos algoritmus, buvo nuspręsta įgyvendinti *SURF*, *SIFT*, *ORB*, *BRISK* ir kitus populiarius metodus ir eksperimentiškai ištirti jų našumą skirtingiems vaizdo jutikliams;
6. Suprojektavus ir realizavus charakteringų taškų aptikimo, aprašymo ir palyginimo sistemą buvo galima atlikti eksperimentus nuskaitant atvaizdus, šiems atvaizdams pritaikant atvaizdų apdorojimo filtrus, pritaikant pasirinktą detektorius, aprašą ir charakteringų taškų palyginimo metodą. Sistemai pritaikytas *pipes and filters* projektavimo šablonas leido efektyviai atlikti skaičiavimus paleidžiant lygiagrečias skaičiavimo sakas.

6 LITERATŪROS SĄRAŠAS

- [1] Abdel-Hakim, A.E. And Farag, A.A. Csift: A Sift Descriptor With Color Invariant Characteristics. *Ieee*, 2006.
- [2] Adrian Rosebrock. *Opencv Gamma Correction*. October 5, 2015, October 5, 2015 Available From: <https://www.pyimagesearch.com/2015/10/05/opencv-gamma-correction/>.
- [3] Agast Corner Detector. <http://www.i6.in.tum.de/main/researchagast>. Available From: <http://www.i6.in.tum.de/main/researchagast>.
- [4] Alcantarilla, P.F. And Solutions, T. Fast Explicit Diffusion For Accelerated Features In Nonlinear Scale Spaces. *Ieee Trans.Patt.Anal.Mach.Intell*, 2011, Vol. 34, No. 7. Pp. 1281-1298.
- [5] Alcantarilla, P.F., Bartoli, A. And Davison, A.J. *Kaze Features*. Springer, 2012.
- [6] Bay, H., Ess, A., Tuytelaars, T. And Van Gool, L. Speeded-Up Robust Features (Surf). *Computer Vision And Image Understanding*, 2008, Vol. 110, No. 3. Pp. 346-359.
- [7] Calonder, M., Et Al. Brief: Computing A Local Binary Descriptor Very Fast. *Ieee Transactions On Pattern Analysis And Machine Intelligence*, 2012, Vol. 34, No. 7. Pp. 1281-1298.
- [8] Calonder, M., Lepetit, V., Strecha, C. And Fua, P. *Brief: Binary Robust Independent Elementary Features*. Springer, 2010.
- [9] Cho, K., Iketani, N., Setoguchi, H. And Hattori, M. Human Activity Recognizer For Mobile Devices With Multiple Sensors. *Ieee*, 2009.
- [10] David Lowe. The Sift (Scale Invariant Feature Transform) Detector And Descriptor, *Newer Journal Paper Ijcv* 2004.
- [11] Donoser, M. And Bischof, H. Efficient Maximally Stable Extremal Region (Mser) Tracking. *Ieee*, 2006.
- [12] Engineer Live. Measure Temperature And Humidity With Mobile, 12th April 2013. Available From: <http://www.engineerlive.com/content/measure-temperature-and-humidity-mobile>, <https://newatlas.com/camera-sensor-size-guide/26684/>
- [13] Hinckley, K., Pierce, J., Sinclair, M. And Horvitz, E. *Sensing Techniques For Mobile Interaction*. *Acm*, 2000.
- [14] Klüppel, M., Et Al. Bildverarbeitung Für Die Medizin 2014springer, 2014 *On Feature Tracking In X-Ray Images*, Pp. 132-137.
- [15] Leutenegger, S., Chli, M. And Siegwart, R.Y. *Brisk: Binary Robust Invariant Scalable Keypoints*. *Ieee*, 2011.

- [16] Mair, E., Et Al. Adaptive And Generic Corner Detection Based On The Accelerated Segment Test. Springer, 2010.
- [17] Noah Snavely. *Cs6670: Computer Vision*. Available
From: https://www.cs.cornell.edu/courses/cs6670/2011sp/lectures/lec02_filter.pdf.
- [18] [P. Forss´En, D. Lowe, S-H Wang. *Multi-Resolution Mser.* , 2011.
- [19] Philip Daubmeier. Determining Body-Orientation From Sensors In Mobile Devices. Master Ed. , July 16, 2012.
- [20] Rosten, E. And Drummond, T. Fusing Points And Lines For High Performance Tracking. Ieee, 2005.
- [21] Rublee, E., Rabaud, V., Konolige, K. And Bradski, G. *Orb: An Efficient Alternative To Sift Or Surf*. Ieee, 2011.
- [22] Ryan Goodrich. *Accelerometers: What They Are & How They Work*. October 1, 2013
Available From: <http://www.livescience.com/40102-accelerometers.html>.
- [23] Satya Mallick. *Blob Detection Using Opencv*. February 17, 2015, February 17, 2015
Available From: <https://www.learnopencv.com/blob-detection-using-opencv-python-c/>.
- [24] University Of Auckland. *Gaussian Filtering*. 5/25/2010 Available
From: https://www.cs.auckland.ac.nz/courses/compsci373s1c/patriceslectures/gaussian%20filtering_1up.pdf, <http://easylearntutorial.com/series/algorithms-data-structures/gaussian-blur-image-processing-algorithm/>.
- [25] Web.Stanford.Edu *Sensors And Cellphones*. Available
From: <http://web.stanford.edu/class/cs75n/sensors.pdf>.
- [26] Yao Wang. *Image Filtering: Noise Removal, Sharpening, Deblurring*. 2006, 2011
Available From: http://eeweb.poly.edu/~yao/ee3414/image_filtering.pdf,
<https://robotronblog.com/2016/04/20/computer-vision/>.
- [27] Will Decker, Masashi Narumoto. *Pipes and Filters pattern*. 06/23/2017 Available
From: <https://docs.microsoft.com/en-us/azure/architecture/patterns/pipes-and-filters>.

7 PRIEDAI

7.1 Straipsnis „The Mobile Application Based on Augmented Reality for Learning STEM Subjects“

Paduotas recenzuoti 2018.04.27 ICIST 2018 konferencijoje.

The Mobile Application Based on Augmented Reality for Learning STEM Subjects

Tomas Valatkevičius¹, Andrius Paulauskas¹, Tomas Blažauskas¹, Reda Bartkutė¹

¹ Kaunas University of Technology
tomas.valatkevicius@ktu.lt
andrius.paulauskas@ktu.lt
tomas.blazauskas@ktu.lt
reda.bartkute@ktu.lt

Abstract. App store are full of programs, which are based on augmented reality. A lot of studies showed that augmented reality has a lot of benefits for user's ability to learn new things or their increase in motivation. However, the majority of these programs are dedicated for entertainment and just several of them are designed for learning processes. The authors have developed an app based on an augmented reality, which aims to provide scientific formulas for math, physics and chemistry subjects and by this to ease the exercise solving. The target group is K12 learners in school. The app was uploaded into app store for both, iOS and Android operating systems. Further researches have to be done on finding the impact of this type of programs. If the impact would be positive, the authors suggest to improve an app by adding more specific topics.

Keywords: Augmented Reality; Mobile Application; STEM; Learning.

1 Introduction

In Lithuania, education system is quite static and lacks of different learning methods and styles. Students are not involved in learning process and it affects the way that they will generate information in the future. The Stanford Teaching Commons indicate that active learning is meaningful not only for a child but for a student also. It indicates that when students are engaged in active learning, they are more fascinated about learning subject and learn more, whether it is a practice, discussion or some application [1].

Not new term in the field of technology is augmented reality. It incorporates virtual reality into the real world and is interactive [2]. Apart from its advantages, AR is not widely used in educational process, in Lithuania. Another issue here is that a lot of teachers do not imagine how VR and AR technologies can be incorporated into learning processes. One of the simplest ways to do it is a mobile application for learning [3].

The use of mobile applications for studying in Lithuanian schools and universities is still not so popular. The old methods used, without any teaching interactions and the advantages of the new emerging technologies remain untouched. Another problem of the usage of mobile learning applications related with the fact that there are quite small amount of educational content applications in the market, especially when talking about STEM subjects.

2 Methodology

First of all, the literature on the augmented reality-based mobile applications models used for education in the world was overviewed. In the second part of the article, the mobile application “KTU Decode” was introduced and briefly presented. Later on, basic principles of operation of the application were presented. All in all, the authors have made the conclusions with some recommendations and future perspectives.

3 Literature review

When people hear an augmented reality, it mostly is associated with the use for games and fun experiences. However, in the whole world, there are a thousand of good practices of augmented reality usage in educational process. The cheapest way for all students to have interactive and emerging new technologies in their everyday learning process is mobile applications based on augmented reality [4]. These applications are usually free and requires just a smartphone and the camera [5].

In 2014, two IT developers from Malaysia have created an augmented reality-based mobile application for learning science subject in primary schools. Their application called *iSains* and had two topics: moon phases, day, and night. The application let students to see different moon phases in 3D through their textbooks. The authors had claimed that further researches have to be done to know the exact effect on pupils learning process and more topics should be involves making this augmented reality-based application more useful [6].

Research on an augmented reality based mobile application for learning made in 2014, in Taiwan. The research took the place at natural science course in elementary school. The authors have constructed an augmented reality-based learning system, which was created by using these tools: JAVA, Oracle, Xcode. The operating principle of the system based on the GPS location of the students and it guided them to the ecology areas where they could do the special tasks. Pupils could use the camera to capture the images of real environment and later on students can edit images, according to the tasks and share them with their community through the WIFI. In the test presented feedbacks from 57 pupils. Results have revealed that students showed higher motivation in learning, lesson attendance and confidence [7].

In 2015, Bernadette Perry from Canada created and implemented a prototype of an “*Explorez*”, which is an augmented reality based mobile learning-tool to help students to learn French language. This prototype implemented into a first year University French-language class. The author asked students about their experience using this learning-tool and all of them were positive. Majority of the students have identified that it was a great experience and it was very nice to get out of a class, even just through the mobile phones. They liked the concept that everything is going on in a real environment but with some added augmented reality elements. Perry has identified that there should be further researches on the feedbacks after a longer time but for this day increased students motivation to learn is clearly seen [8].

The benefits of augmented reality are obvious and easy acceptable. Without the benefits listed above, there are much more. For example, in engineering graphics lessons, augmented reality book used for students to see different objects from 3D perspectives

and to solve complementary problems. The questions and problems defined in that way, that students would be able to train their mental rotation abilities. After the test, researchers claimed that augmented reality based learning has a positive effect on students spatial abilities improvement [9].

Other researches about augmented reality based mobile learning applications revealed that this type of tools increase collaborative learning [10], motivation to perform learning activities [11], improve learning attitude and learning outcome [12], enrich ideas [13] and engage concurrence and satisfaction [14].

4 Overview of “KTU DECODE”

“KTU Decode” is an augmented reality-based mobile application, which purpose is to compliment the presents of Kaunas University of Technology: agendas for math physics and chemistry. With the help of this app, in the agendas users can see virtual formulas. In every different agenda users can see formulas for specific wanted theme. The application reflects virtual formulas for specific themes and with the help of it, it is easier to solve exercises.

The target group of the application is higher classes learners at school. The aim of this app is to ease the solving of exercises by using formulas for math, physics and chemistry.

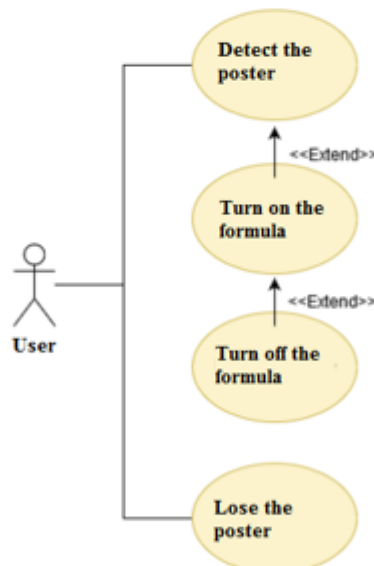


Fig. 1. Use case diagram.

In the figure above (Fig.1.) you can see the use case diagram. In the program one user engaged, that can do several tasks. The entire task are listed in the figure and described below.

Poster detection. To detect the poster it is necessary to turn on the app and to allow it to use your device camera. After the camera directed to the poster, the app detects what kind of poster it is and formulas appears in the screen.

Turning on the formulas. Each of the posters have different themes and formulas that are complimenting one specific poster. User is able to choose themes and by pressing on them to turn on the layer of scientific formulas, that can be switched.

Turning off the formulas. User, which already turned on the layer of formulas, by pressing on the same theme twice, can turn off an additional layer of formulas.

Lose of the poster. When the poster is lost, user cannot turn on or turn off the scientific formulas.

5 The model of “KTU Decode”

The app based on two main components: “Unity 3D” game engine and “Vuforia” the library of an augmented reality platform. The logic of an app was written in C# language. In the figure below (Fig.2.) is the model of the app.

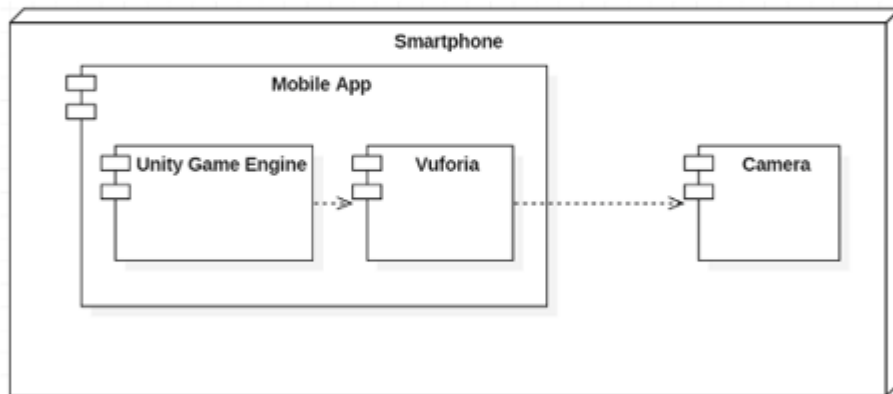


Fig.2. Layout diagram of the app

The process is starting with the “Vuforia” which uses a camera of the smartphone to locate points of the interests. The points transferred to Unity game engine in which a content of an augmented reality is drawn.

For the use of “Vuforia”, the unique app licensee identification is must. This identification generated in “Vuforia” management interface by registering the app in progress. The license identification must be transferred to “Vuforia” configurable file in “Unity 3D” game engine (Fig.3.).



Fig. 3. License identification in “Unity 3D” game engine

The app of “KTU Decode” does not use in “Vuforia” platform provided, points of interests, based on cloud computing. As points for an app, posters were used (Fig.4.). The worksheets of the posters were uploaded to the “Vuforia” database through it management interface.



Fig.4. Posters used as points of interests in the app

The system process these posters and give the quality rate as a feedback. The quality rate is the explanation of how good the points of interests will be followed. Processed databases of images were downloaded and implemented into the project.

The app was uploaded into an “App store” and “Google Play” stores and is free to download. When uploading the app to “iOS” operational system, “Unity 3D” tools have been used for generating “XCode” environment.

6 The experiment on the mobile application effectiveness

To evaluate the effectiveness of the mobile application taken two aspects: existing applications for educational needs and learning results.

There do exist many applications for learning, but there are not so many related with VR and AR. For the experiment there were taken two types of applications 1st model (not related with AR or VR) and the 2nd model related with AR and VR (see fig. 5).

To evaluate KTU DUCODE were invited an experts, who provided their opinion about the models integration into practice.

For experiment there were taken two different models to evaluate the effectiveness of the KTU DUCODE application. There was created questionnaire for the experts to compare that two methods implementation. The 1st group used 1 METHOD, the 2nd group used 2 METHOD.

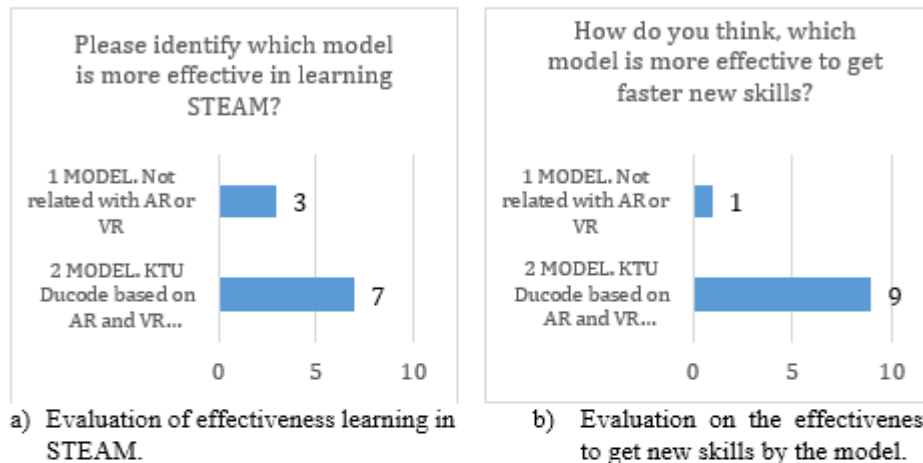


Fig. 5. Evaluation results.

Finally, we can declare that the 2nd model is more useful in learning STEM subjects. Totally 7 respondents declared the functionality of the 2 model is more effective in learning process and totally 9 experts declared that this model is more effective to get new skills.

7 Conclusions

As the great number of studies showed that an augmented reality can help students to remember everything better and to engage in learning process, the authors have decided to try to create simple mobile application, which would be based on an augmented reality. The augmented reality-based mobile application created to help students to solve exercises for math, physics and chemistry faster and easier. The authors have uploaded the app into the app stores for both, iOS and Android operating systems. "KTU DUCODE" is free to access and use by students.

However, no studies done to investigate the benefits of the app: if it is useful or how to improve it; if the app help to solve exercises faster or not. The future research can be done

by investigating the impact of these kind of programs for students. For the app improvement, the authors suggest to add more topics/themes.

8 References

1. Teaching Commons Homepage, <https://teachingcommons.stanford.edu/>, last accessed 2018/03/30.
2. Barfield, W. (Ed.). (2015). *Fundamentals of wearable computers and augmented reality*. CRC Press.
3. Ledford, C. J., Canzona, M. R., Cafferty, L. A., & Hodge, J. A. (2016). Mobile application as a prenatal education and engagement tool: a randomized controlled pilot. *Patient education and counseling*, 99(4), 578-582.
4. Rauschnabel, P. A., Brem, A., & Ro, Y. (2015). *Augmented reality smart glasses: definition, conceptual insights, and managerial importance*. Unpublished Working Paper, The University of Michigan-Dearborn, College of Business.
5. tom Dieck, M. C., & Jung, T. (2018). A theoretical model of mobile augmented reality acceptance in urban heritage tourism. *Current Issues in Tourism*, 21(2), 154-174.
6. MAJID, Nazatul Aini Abd; HUSAIN, Nooraidah Kamarudin. Mobile learning application based on augmented reality for science subject: *Isains*. *ARN Journal of Engineering and Applied Sciences*, 2014, 9.9: 1455-1460.
7. Chiang, T. H., Yang, S. J., & Hwang, G. J. (2014). An augmented reality-based mobile learning system to improve students' learning achievements and motivations in natural science inquiry activities. *Journal of Educational Technology & Society*, 17(4), 352.
8. Perry, B. (2015). Gamifying French Language Learning: a case study examining a quest-based, augmented reality mobile learning-tool. *Procedia-Social and Behavioral Sciences*, 174, 2308-2315.
9. Hung, Y. H., Chen, C. H., & Huang, S. W. (2017). Applying augmented reality to enhance learning: a study of different teaching materials. *Journal of Computer Assisted Learning*, 33(3), 252-266.
10. Ke, F., & Hsu, Y. C. (2015). Mobile augmented-reality artifact creation as a component of mobile computer-supported collaborative learning. *The Internet and Higher Education*, 26, 33-41.
11. Ibáñez, M. B., Di Serio, Á., Villarán, D., & Delgado-Kloos, C. (2016, July). The acceptance of learning augmented reality environments: A case study. In *Advanced Learning Technologies (ICALT), 2016 IEEE 16th International Conference on* (pp. 307-311). IEEE.
12. Cai, S., Chiang, F. K., Sun, Y., Lin, C., & Lee, J. J. (2017). Applications of augmented reality-based natural interactive learning in magnetic field instruction. *Interactive Learning Environments*, 25(6), 778-791.
13. Wang, Y. H. (2017). Exploring the effectiveness of integrating augmented reality-based materials to support writing activities. *Computers & Education*, 113, 162-176.
14. Wu, P. H., Hwang, G. J., Yang, M. L., & Chen, C. H. (2018). Impacts of integrating the repertory grid into an augmented reality-based learning design on students' learning achievements, cognitive load and degree of satisfaction. *Interactive Learning Environments*, 26(2), 221-234.