



KAUNO TECHNOLOGIJOS UNIVERSITETAS

ELEKTROS IR ELEKTRONIKOS FAKULTETAS

Lina Jukonytė

**DVIKAMERINIO CHARAKTERINGŲ TAŠKŲ FILTRO
SUKŪRIMAS IR TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Doc. Dr. Renaldas Urniežius

KAUNAS, 2016

KAUNO TECHNOLOGIJOS UNIVERSITETAS

ELEKTROS IR ELEKTRONIKOS FAKULTETAS

AUTOMATIKOS KATEDRA

**DVIKAMERINIO CHARAKTERINGŲ TAŠKŲ FILTRO
SUKŪRIMAS IR TYRIMAS**

Baigiamasis magistro projektas

Valdymo Technologijos (kodas 621H66001)

Vadovas: Doc. dr. Renaldas Urniežius
(parašas)
(data) 2016-06-02

Recenzentas: Doc. Dr. Vidas Raudonis
(parašas)
(data)

Projektą atliko: Lina Jukonytė
(parašas)
(data) 2016-06-02

KAUNAS, 2016



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir elektronikos fakultetas

(Fakultetas)

Lina Jukonytė

(Studento vardas, pavardė)

Valdymo technologijos, 621H66001

(Studijų programos pavadinimas, kodas)

Dvikamerinio charakteringųjų taškų filtro sukūrimas ir tyrimas

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 16 m. gegužės 22 d.
Kaunas

Patvirtinu, kad mano **Linos Jukonytės** baigiamasis projektas tema „Dvikamerinio charakteringųjų taškų filtro sukūrimas ir tyrimas“ yra parašytas visiškai savarankiškai, o 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)

Jukonytė, L. Dvikamerinio charakteringųjų taškų filtro sukūrimas ir tyrimas. *Valdymo sistemų magistro baigiamasis projektas / vadovas* doc. dr. Renaldas Urniežius; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas, Automatikos katedra.

Kaunas, 2016. 79 psl.

SANTRAUKA

Jau kuris laikas kompiuterinė rega ir vaizdų analizė yra labai svarbi ir plačiai nagrinėjama dėl savo pritaikomumo įvairiose mokslo ir pramonės šakose. Įvairių uždavinių, naudojančių kompiuterinę regą, sprendimui reikalingi skirtingi algoritmai, vieniems reikalingas greitis, kitiems tikslumas. Dažniausiai sprendžiant tokius uždavinius, visi charakteringieji taškų filtrai yra orientuoti į greitį, todėl trūksta filtrų, kurie charakteringuosius taškus aptiktų kokybiškai.

Pagrindinis darbo tikslas yra sukurti ir iširti dvikamerinį charakteringųjų taškų filtrą, kuris būtų orientuotas į tikslumą. Darbo metu bus ieškoma tų pačių charakteringųjų taškų porų dviejuose nuotraukose, paieškos algoritme poros taip pat tikrinamos ir pagal trimatę padėtį remiantis dviejų kamerų vaizdais, taigi tikrinama ir lokalizacija. Charakteringųjų taškų filtras bus pritaikomas ir taškams spalvotame vaizdui rasti.

Tiriamasis darbas pradedamas literatūrine analize, kur apžvelgiami jau esami charakteringųjų taškų radimo metodai, aprašoma kameros, judančios aplink savo ašį sistemos kūrimas, naudojant paprastą ir pigų Arduino mikroprocesorių bei žingsninį variklį.

Ekspimentinėje dalyje analizuojamas filtro veikimas, tiriamas kampų ir jų porų dviejose nuotraukose atpažinimo tikslumas pilkame ir spalvotame fonuose. Atliekamas lokalizacijos eksperimentas, pagal atstumą ir kampą tarp kamerų, galiausiai atliekamas charakteringųjų taškų filtro palyginimas su jau esamais Harris ir FAST Features kampų atpažinimo filtrais.

Reikšminiai žodžiai (iki 8 žodžių):

Charakteringieji taškai nuotraukoje, Harris Corner metodas, FAST Features metodas, charakteringųjų taškų poros randamos remiantis dviejų kamerų vaizdais, dvikamerinė sistema.

Jukonytė, Lina. Development and Investigation of Features Descriptors Filter in Image Rectification. *Final project of Control Systems Master degree* / supervisor doc. dr. Renaldas Urniežius; Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, department of Automation.

Kaunas, 2016. 79 pages.

SUMMARY

For some time now computer vision and image analysis is very important and widely analyzed for their applicability in the various scientific and industrial applications. Solution of various tasks which using computer vision, requires different algorithms, one of them oriented on speed, others on accuracy. Dealing with such tasks filter for characteristic points of image, mostly, are focused on speed, so there is a lack of filters that can detect different algorithms qualitatively.

The main goal of this paper is to develop and explore the characteristic points (feature points) of the filter with pictures from two cameras, which would be focused on accuracy. During this research paper writing time there will be looking for the same characteristic points of the two pairs of photos, the search algorithm pairs also tested and the three-dimensional position of the two cameras, so checked and localization. Filter characteristic points will be adjusted and the points on the color images were found.

Research work begins literary analysis, which gives an overview of already existing characteristic point detection methods, described one chamber, moving around its axis system design, using a simple and cheap Arduino microprocessor and stepper motor.

The experimental part analyzes the functionality, studied the corners and two pairs of pictures recognition accuracy on gray and colored backgrounds. Localization experiment carried out in accordance with the distance and angle of the camera, then a filter is compared with the filter already which already exist Harris corners and FAST Features corner detections filters.

Keywords (up to 8 words):

Feature points on image, Harris corner detection, FAST Features corner deflection, pairs of feature points on image from two cameras, multi-camera system.

TURINYS

SANTRUMPOS IR TERMINAI	11
ĮVADAS.....	12
1. LITERARŪROS ANALIZĖ.....	13
1.1 Vaizdo suvokimas ir atpažinimas.....	13
1.2 Spalvos	14
1.2.1 RGB.....	15
1.2.2 HSL	15
1.3 Objekto charakteringųjų vaizde taškų nustatymo metodai.....	15
1.3.1 Moravec filtras	16
1.3.2 Harris – Stephens kampų aptikimo filtras	17
1.3.3 FAST Features vaizdo apdorojimo metodas	20
1.4 Sobelio kraštų išskyrimo metodas (filtravimas).....	22
1.4.1 Sobelio metodas 3x3	23
1.5 Gauso suliejimas	23
1.6 Gradientas	25
1.7 Objektų panašumas	26
1.7.1 Taksi geometrija.....	26
1.7.2 Manhattan'o atstumas	26
1.7.3 Euklido atstumas	27
1.8 Stereometrija	28
2. Naudojamų technologijų analizė.....	29
2.1 Jutikliai, vertinantys aplinką	30
2.2 Kamerų pasukimo mechanizmas.....	31
2.2.1 Servo pavara.....	31
2.2.2 Žingsninis variklis	32
2.3 Realizuojančios platformos	32
2.4 Programavimo kalbos.....	33
2.4.1. C# programavimo kalba	34
3. Eksperimentinė dalis	35
3.1 Kalibravimo eksperimentas.....	36
3.2 Charakteringųjų taškų filtras	38

3.3	Kampo aptikimas.....	40
3.4	Kampų porų radimas	42
3.5	Burbulo metodas	44
3.6	Paklaidos	44
5.	Ekspirimentiniai tyrimai	45
5.1	Rezultatai gauti tiriant nuotraukų poras (nespalvotas fonas)	48
5.2	Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, B sluoksnis).....	49
4.3	Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, G sluoksnis)	51
4.4	Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, R sluoksnis).....	52
4.5	Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, visi spalvų sluoksniai).....	54
4.6	Išmatuotų ir gautų atstumo įverčių radimo tyrimas.....	57
4.7	Sukurto charakteringųjų taškų filtro palyginimas su Harris – Stephens ir FAST Features kampų atpažinimo metodais.....	59
	IŠVADOS IR REZULTATAI.....	66
	Literatūros sąrašas	67
	PRIEDAI	71
	Priedas1. Ieškomų kampų ir atmetamų tiesių lygtys	71
	Priedas2. Arduino programinis kodas	72
	Priedas3. Kampų atpažinimo programinis kodas	76
	Priedas4. Kampų porų aptikimo programinis kodas	77

PAVEIKSLĖLIŲ SĄRAŠAS

1 pav. Tiriamojo lango funkcija Moravec filtre.....	16
2 pav. Pikselių regionai. Kairėje lygumos, per vidurį krašto ir dešinėje kampo.....	17
3 pav. Tiriamojo lango funkcija Harris – Stephens kampų atpažinimo filtre.....	18
4 pav. Kampų aptikimas pagal λ_1 ir λ_2	19
5 pav. Teritorijų atvaizdavimas x ir y ašyse.....	19
6 pav. FAST Features metodo charakteringųjų taškų veikimo principas.....	21
7 pav. Gauso suliejimas. Paveikslėlis prieš ir po.....	24
8 pav. Gradientas.....	25
9 pav. Manhatano atstumas.....	27
10 pav. Struktūrinė filtro tyrimo schema.....	29
11 pav. Supaprastinta duomenų perdavimo schema.....	30
12 pav. Logitech C270 internetine kamera.....	31
13 pav. Servo pavara.....	32
14 pav. Žingsninis variklis (kairėje) ir EsayDrive 4.4 valdiklis (dešinėje).....	32
15 pav. Arduino Uno mikrovaldiklis.....	33
16 pav. Sistema, sukurta pirmiesiems bandymams atlikti.....	35
17 pav. Kalibravimo eksperimentas.....	36
18 pav. Ūd skaičiavimas.....	37
19 pav. Statūs kampai, atpažįstami programoje.....	38
20 pav. Smailūs kampai, atpažįstami programoje.....	38
21 pav. Tiesės.....	39
22 pav. 45° pasukti statūs kampai.....	39
23 pav. Pirmieji rezultatai.....	46
24 pav. Dviejų nuotraukų pilkame fone eksperimentiniai tikrinimo rezultatai.....	47
25 pav. Charakteringųjų taškų porų radimas.....	48
26 pav. Atrastų porų dažnumo pilkame fone histograma.....	49
27 pav. Teisingai aptiktų porų pilkame fone pasiskirstymas.....	49
28 pav. Atrastų porų dažnumo spalvotame fone (mėlynos spalvos sluoksnyje) histograma.....	50
29 pav. Teisingai aptiktų porų spalvotame fone (mėlynos spalvos sluoksnyje) pasiskirstymas.....	51
30 pav. Atrastų porų dažnumo spalvotame fone (žalios spalvos sluoksnyje) histograma.....	52

31 pav. Teisingai aptiktų porų spalvotame fone (žalios spalvos sluoksnyje) pasiskirstymas.....	52
32 pav. . Atrastų porų dažnumo spalvotame fone (raudonos spalvos sluoksnyje) histograma.....	53
33 pav. Teisingai aptiktų porų spalvotame fone (raudonos spalvos sluoksnyje) pasiskirstymas.....	53
34 pav. Atrastų porų dažnumo spalvotame fone histograma.....	55
35 pav. Teisingai aptiktų porų spalvotame fone pasiskirstymas.....	55
36 pav. Atrastų porų dažnumas spalvotame fone, įvertinant atstumą.....	56
37 pav. Teisingai aptiktų porų spalvotame (įvertinant atstumą) fone pasiskirstymas.....	57
38 pav. Atstumo remiantis dviejų kamerų vaizdais nustatymo eksperimento langas.....	57
39 pav. Trimatės padėties nustatymo iš remiantis dviejų kamerų vaizdais paklaidos priklausomybė nuo atstumo.....	58
40 pav. Vienos nuotraukos kampų radimo greičio ir tikslumo palyginimo langas.....	59
41 pav. Teisingai rastų kampų tikslumo palyginimas tarp skirtingų metodų ir skirtingų rezoliucijų.	61
42 pav. Viso algoritmo greitaveikos ir tikslumo palyginimas įvairiais metodais.....	62
43 pav. Teisingai rastų 6 kampų porų dviejuose nuotraukose vidurkis.....	63

LENTELIŲ SĄRAŠAS

1 lentelė. Logitech C270 internetinės kameros techninės specifikacijos.....	31
2 lentelė. Pilkame fone teisingai atrastų porų skaičius.....	48
3 lentelė. Atrastų porų dažnumas pilkame fone.....	49
4 lentelė. Spalvotame fone (mėlynos spalvos sluoksnyje) teisingai atrastų porų skaičius.....	50
5 lentelė. Atrastų porų dažnumas spalvotame fone (mėlynos spalvos sluoksnyje).....	50
6 lentelė. Spalvotame fone (žalios spalvos sluoksnyje) teisingai atrastų porų skaičius.....	51
7 lentelė. Atrastų porų dažnumas spalvotame fone (žalios spalvos sluoksnyje).....	51
8 lentelė. Spalvotame fone (raudonos spalvos sluoksnyje) teisingai atrastų porų skaičius.....	52
9 lentelė. Atrastų porų dažnumas spalvotame fone (raudonos spalvos sluoksnyje)	53
10 lentelė. Spalvotame fone teisingai atrastų porų skaičius.....	54
11 lentelė. Atrastų porų dažnumas spalvotame fone.....	54
12 lentelė. Spalvotame fone teisingai atrastų porų skaičius, įvertinant atstumą.....	56
13 lentelė. Atrastų porų dažnumas spalvotame fone, įvertinant atstumą.....	56
14 lentelė. Išmatuota ir programiškai apskaičiuota trimatė padėtis remiantis dviejų kamerų vaizdais.....	58
15 lentelė. Greitaveikos ir tikslumo palyginimas tiriant vieną nuotrauką.....	60
16 lentelė. Greitaveikos vidurkiai tiriant nuotraukų poras aukštesne ir žemesne rezoliucijomis.....	61
17 lentelė. Greitaveikos ir tikslumo rezultatai pakeitus nuotraukos rezoliucija.....	62
18 lentelė. Viso algoritmo vidutinės greitaveikos ir tikslumo palyginimas.....	63

SANTRUMPOS IR TERMINAI

FAST Features (*angl. Features from accelerated segment test*) – charakteringųjų taškų nustatymo vaizde metodas.

HARRIS (*angl. harris*) – charakteringųjų taškų nustatymo vaizde metodas.

RGB (*angl. red green blue*) – trijų kanalų spalvų erdvė naudojama vaizdų kodavimui.

HSL – (*angl. hue, saturation, value*) – spalvų modelis sukurtas pagal pagrindinius tris parametrus: atspalvį, saturaciją ir apšviestumą.

USB (*angl. universal serial bus*) – standartizuota duomenų perdavimo sąsaja.

2D/3D (*angl. two/three dimensions*) – plokštumą arba trimatę erdvę nusakantis žymėjimas.

Visual Studio – atviro kodo C# programavimo kalbos kodo kompiliatorius.

Hardware – techninė įranga.

Software – programinė įranga.

IVADAS

Žmonės matydami aplinką gali lengvai išskirti ir atpažinti įvairias spalvas, formas ir objektus. Tai savaime suprantama užduotis, kuri žmogui išsprendžiama pasitelkus patirtį, įgytą laikui bėgant, tačiau mašininė rega ir objektų atpažinimas vis dar yra sudėtingas reiškinys, neturintis vieno universalaus gerai veikiančio įvairiomis aplinkos sąlygomis algoritmo. Vaizdų analizė yra labai svarbi sritis dėl savo pritaikomumo pramonės ir mokslo srityse. Dėl šios priežasties, tobulėjant naujoms technologijoms išaugo ir susidomėjimas paveiksle užkoduotos informacijos (charakteringųjų taškų), kuri leidžia atpažinti ir sekti vaizde esančių objektų padėtį, išskyrimu. Charakteringieji taškai realybėje naudojami objekto įterpimo koordinatėms nustatyti ir išlaikyti: objektas, įterptas į vaizdą, savo padėtį keičia nepriklausomai nuo vaizdo kameros padėties, jo padėtis priklauso tik nuo būdingųjų taškų, kuriuos naudojant objektas buvo įterptas, padėties. Charakteringieji taškai dažnai naudojami vaizdo glaudinimo algoritmuose bei pasirinktų objektų sekimui vaizdų sekoje.

Išaugus susidomėjimui būdingaisiais taškais, jų paieškos ir sutapdinimo algoritmų vis daugėja. Kiekvienam iš jų būdinga sava specifika: vieniems aktualiau atpažinti kampus, kitiems kraštines. Vieni orientuoti į didesnę tikslumą, kiti į aptikimo spartą ir t.t. Magistro tiriamojo darbo metu bus kuriamas charakteringųjų taškų filtras, kuris padės orientuotis erdvėje. Filtras bus kuriamas, norint išgauti kuo didesnę tikslumą, neatsižvelgiant į taškų aptikimo spartą.

Darbo tikslas ir uždaviniai:

Sukurti filtrą charakteringiesiems vaizdo taškams gauti, remiantis stereometrijos metodo suteikiama informacija apie aplinką.

Uždaviniai:

- Išnagrinėti esamus vaizdų charakteringųjų taškų radimo metodus ir stereometrijos būdą naudojamą erdvinėje lokalizacijoje;
- Realizuoti charakteringųjų vaizdo taškų radimą programinėje įrangoje;
- Iširti sukurto filtro veikimą eksperimentiškai (filtro tikslumą žinomos erdvės aplinkoje, uždaroje patalpoje) ir nustatyti įverčių nuokrypius.

1. LITERARŪROS ANALIZĖ

1.1 Vaizdo suvokimas ir atpažinimas

Skaitmeninis vaizdas – tai programiniu būdu atvaizduotas dvimatis (arba trimatis) skaičių masyvas, sudarytas iš mažiausių vaizdo elementų – taškų (pikselių). Vaizdas gali būti dviejų rūšių: statinis (paveikslas) arba dinaminis (judantis vaizdas). Skaitmeniniam vaizdų apdorojimui naudojamos įvairios kompiuterinės programos su įvairiais atpažinimo algoritmais [1]. Aplinkoje vaizdai būna dviejų tipų: atspindintys ir spinduliuojantys. Priklausomai nuo vaizdo rūšies jie turi skirtingus parametrus. Spinduliuojamą vaizdą aprašomas spalva (priklauso nuo bangos ilgio) ir apšviestumu (nulemia šviesos kiekis), o atspindintis vaizdas nulemiamas trijų parametru: atspalvio, spalvos saturacijos (sočio) ir apšviestumo. [2] [47]

Atspalviu vadinamas parametras, kai kalbama apie gryną spalvą (geltoną, žalią, mėlyną ar raudoną), spalva priklauso nuo bangos ilgio, kuris pasiekia stebėtoją. Pilka, juoda ir balta spalvos nelaikomos atspalviais, nes tai proporcingas kelių spalvų mišinys. Spalvos saturacija parodo spalvos grynumą ir jos artimumą baltai spalvai. Saturacijai didėjant spalva ryškėja, o mažėjant blanksta, kol tampa balta. Sočiaja spalva vadinama ryškus ir kontrastingas atspalvis, o nesochiaja – išblukęs. Baltos ir sočiosios spalvos diapazonas yra nuo 0 iki 100%: grynos spalvos saturacija yra 100%, o baltos 0%. Apšviestumas parodo, kiek spalva yra apšviesta, t.y. koks jos intensyvumas lyginant su juoda spalva: jei apšviestumas yra 100%, tai turima balta spalva, o jei 0%, tai turima juoda spalva. Keičiantis apšviestumui, spalva kinta pilkumo skalėje (nuo juodos iki baltos spalvos). Kontrastas yra santykinis dydis, parodantis šviesiausio ir tamsiausio paveikslo ploto santykį [3][2][45].

Paveikslams su dideliu kontrastingumu būdingi dideli šviesūs ir tamsūs plotai. Didėjant kontrastui, didėja paveiksle esančių detalių kiekis, nors paveiksle esančios informacijos kiekis išlieka toks pat. Tai paaiškinama dideliu žmogaus akių jautrumu šviesos kontrastui ir mažesniu jautrumu absoliučiam šviesos intensyvumui, kai žmogus nebeturi su kuo palyginti šviesos intensyvumo dydžio[4][47].

Vaizdo atpažinimas – tai svarbios informacijos išrinkimas iš vaizdo, jos įvertinimas bei priskyrimas tam tikrai kategorijai. Vaizdo atpažinimo uždavinius sprendžia kompiuteriai, lengvų užduočių sprendimui kompiuteriai yra nepakeičiami, nes jų darbo sparta žmogaus suvokimo spartą lenkia keletą kartų. Tačiau sunkėjant užduočiai, kompiuteriniai atpažinimo metodai tampa vis labiau riboti, nes žmogus šias užduotis gali išspręsti naudodamasis savo patirtimi, neskiriant daug papildomo

laiko ir resursų. Todėl daugelio sudėtingų uždavinių sprendimo būdai kuriami remiantis tais pačiais principais, kuriais naudojami žmogus, atpažįstantis aplinką [2] [47].

Dauguma teorijų, aiškinančių kaip žmogus atpažįsta aplinkoje esančius objektus, grindžiamos principu, kad aplinkos suvokimas vyksta dviem pagrindiniais etapais: jutiminės informacijos apdorojimu, kur siekiama kuo labiau susisteminti gaunamą informaciją (atpažinti formas, suteikti kontūrą ir pan.) ir gautos informacijos lyginimu su jau turima patirtimi ir gautos informacijos patvirtinimu. Tokiu principu paremtas ir kompiuterinis atpažinimas, kur procesas skaidomas į keturias dalis: vaizdo gavimą, apdorojimą, analizę ir supratimą [2] [45].

Vaizdo gavimo dalis naudoja metodus, kur gaunama informacija, kuri paverčiama į mums suprantamus vaizdus. Šiuo etapu iš kamerų gaunama informacija verčiama skaitmeniniu vaizdu, kur matomas kiekvienas paveikslas pikselis. Pikseliai aprašomi jo koordinatėmis paveiksle bei spalvine informacija tame taške. Praktikoje labiausiai paplitęs RGB spalvos kodavimo formatas, kurį sudaro trys spalvos įverčiai: raudona (*angl. red*), žalia (*angl. green*) ir mėlyna (*angl. blue*). Rečiau naudojamas pilkų atspalvių (*angl. greyscale*) formatas, pvz. HSL, kuris sudarytas iš vienos vertės, kuri nusako pikselio pilkumo lygį paveiksle [2][3].

Vaizdo apdorojimo procesas susideda iš svarbiausios paveikslas informacijos išskyrimo ir nereikalingos informacijos, pavyzdžiui, triukšmo, nufiltravimo. Informacijai išskirti naudojami metodai, kuriais klasifikuojami pikseliai, įvertinant jie atitinka užduotas sąlygas ar ne, jei atitinka, jie paliekami, jei neatitinka – pašalinami [4].

Analizės dalyje randama aukštesnio lygio informacija, pavyzdžiui, briaunos, kampai, kontūrai ir pan. Supratimo proceso metu, apdorojama aukščiausio lygio informacija, pagal ją objektai atpažįstami ir klasifikuojami.

1.2 Spalvos

Spalvų paletės aprašo spalvas ir jų tarpusavio ryšius. Pagal poreikį praktikoje naudojamos kelios spalvų paletės, tačiau dauguma jų sudarytos iš trijų dedamųjų: RGB, HSL, HSV, SHI, CMY ir pan., toliau bus nagrinėjami du dažniausiai naudojami spalvų modeliai: RGB (naudojama televizoriuose, monitoriuose) ir HSL (naudojama sistemose, kur tiriama atspalvis, saturacija ir apšviestumas). Dauguma spalvų modelių gali būti suderinama tarpusavyje, t.y. atliekant matematinius skaičiavimus ar trigonometrines operacijas galima pereiti nuo vieno modelio prie kito. [4]

1.2.1 RGB

RGB spalvų paletė sudaryta iš trijų pirminių spalvų: raudonos (R), žalios (G) ir mėlynos (B). Maišant šias pagrindines spalvas galima išgauti bet kokią kitą spalvą. Dažnai RGB spalvų modelis vaizduojamas kaip kubo kraštinės: visoms dedamosioms esant 0, gaunama juoda, o esant maksimalioms reikšmėms gaunama balta spalva. Pilkos spalvos tonai gaunami tada, kai maišomos visos spalvos lygiais kiekiais.

RGB paletė yra labai patogi technikoje, didžiosios dalies vaizdinės technikos prietaisų veikla grindžiama, remiantis šiuo spalvų modeliu. Bet RGB paletė nėra universali, vaizdų apdorojimui ji netinka, nes RGB dedamosios koreliuoja tarpusavyje. Daugeliui vaizdų apdorojimo atvejų spalvoti vaizdai netinka, o dažniausiai naudojami baltos ir juodos spalvų vaizdai su pilkais atspalviais, tokie vaizdai vadinami nespalvotais vaizdais, tačiau tai nėra tokie patys vaizdai, kur vyrauja tik juoda ir balta spalvos. [6]

1.2.2 HSL

HSL spalvų modelis sukurtas pagal pagrindinius tris parametrus: atspalvį (*angl. hue*), saturaciją (*angl. saturation*) ir apšviestumą (*angl. lightness*). Norint nustatyti spalvą užtenka tik pasirinkti atspalvį, spalvos sodrumą ir apšviestumą (norimą spalvos tamsumą). Didžioji dalis skaitmeninių vaizdų apdorojimo algoritmų naudoja šį spalvų modelį, kai norima rasti objektą pagal spalvą ar vaizdo intensyvumą, ar atlikti sąsūką, kur svarbu tik objekto apšviestumas. Atspalvis (H) keičiasi, keičiantis kampui (intervalas nuo 0° iki 360°), saturacija (S) kinta tolygiai nuo cento (intervalas nuo 0 iki 1), o apšviestumas (L) kinta vertikaliai z ašies kryptimi nuo 0 iki 1 (0 – juoda, 1 – balta). Kaip ir RGB spalvų modelyje, keičiantis atspalviui nuo juodos iki baltos spalvos, gaunami pilki tonai. [5]

1.3 Objekto charakteringųjų vaizde taškų nustatymo metodai

Charakteringieji taškai apibūdina ribas, kurios yra labai svarbios atliekant vaizdo apdorojimą. Kampai paveikluose turi stipraus intensyvumo kontrastą – spalvos intensyvumas staigiai peršoka iš vieno pikselio į kitą. Pagrindinis charakteringųjų taškų aptikimo tikslas yra ženkliai sumažinti duomenų kiekį vaizde, tuo pačiu išsaugant struktūrines vaizdo savybes, kurios yra reikalingos tolesniam vaizdo apdorojimui.

Toliau bus apžvelgta šiuo metu egzistuojantys kraštų aptikimo filtrai: tai Moravec filtras, jo pagrindu sukurtas, Harris kampų aptikimo filtras ir FAST Features kampus identifikuojantis filtras.

1.3.1 Moravec filtras

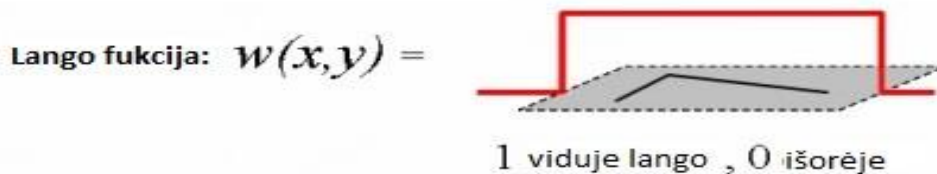
Moravec kampų atpažinimo filtras veikia atsižvelgiant į vietinį langą (*angl. local window*) tiriamame vaizde. Tiriant vietinį langą, stebimas pikselis ir jo kaimynų spalvų intensyvumo pokyčiai visomis kryptimis. Galimi trys aukščiau aprašyti atvejai:

- A. Lango vaizdas yra lygumos regionas (*angl. flat*), t.y. visi pikseliai yra maždaug panašaus intensyvumo. Visi atliekami poslinkiai sukelia nedidelį pokytį.
- B. Jei vaizdo langas kerta kraštą, tada krašto kryptimi spalvos intensyvumas keisis mažai, tačiau statmenai krašto intensyvumo pokytis bus didelis.
- C. Jei vaizdo langas yra kampe arba izoliuotoje vietoje, tai spalvų intensyvumo pokyčiai visomis kryptimis bus dideli. Kampas gali būti aptiktas ieškant minimalaus intensyvumo pokyčio tarp didelių intensyvumo skirtumų.

Viršuje pateiktas procesas matematiškai apsiraso taip. Intensyvumo pokytis, kaimyniniam (perstumtam) pikseliui $[u,v]$:

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (1)$$

kur $w(x, y)$ – tiriamo lango funkcija, $I(x, y)$ – intensyvumas, $I(x + u, y + v)$ – perstumto pikselio intensyvumas. $I(x + u, y + v) - I(x, y)$ – keliui, tarp artimų pikselių ši vertė bus artima 0, keliui, tarp tolimų pikselių, vertė bus didelė. Tiriamasis langas (1 pav) vaizduoja stačiakampę sritį (stačiakampio viduje reikšmė 1 ir 0 stačiakampio išorėje).



1 pav. Tiriamojo lango funkcija Moravec filtre

Moravec filtre atliekami 4 perstumimai: $(u,x) = (1,0), (1,1), (0,1), (-1,1)$. Kampas randamas ieškant lokalaus maksimumo $\min\{E\}$ aplinkoje.

Moravec filtro problemos:

- Rezultatas pasižymi triukšmais dėl dvejetainės tiriamojo lango funkcijos;

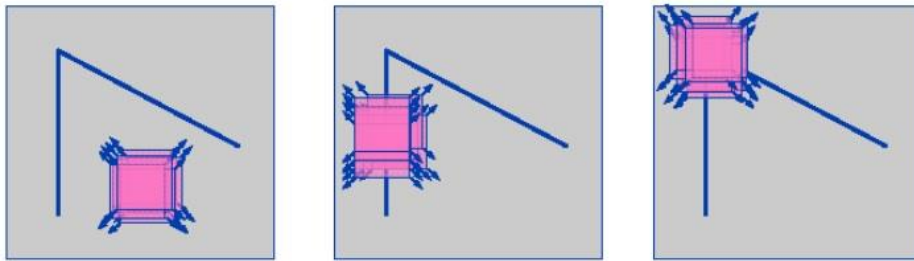
- Kampai ieškomi ir pikseliai tikrinami tik kas 45°;
- Atsižvelgiama tik į minimalias E reikšmes. [7][8]

Šios visos problemos yra bandomos spręsti Harris – Stephens kampų atpažinimo filtre.

1.3.2 Harris – Stephens kampų aptikimo filtras

Šis filtras taip pat žinomas kaip Plessey kampų radėjas (*Plessey corner finder*). Filtras yra plačiai naudojamas kampų ir kraštų atpažinimui vaizdų registravimo, judesio sekimo ir aptikimo, 3D modeliavimo ir objektų atpažinimo uždaviniuose [9]. Chris Harris ir Mike Stephens savo darbe [7] rašo, kad filtras remiasi autokoreliacijos skaičiavimu, tačiau matematikai teigia, kad filtras remiasi skirtumų tarp nagrinėjamų taškų kvadrato suma (*sum of squared differences*).

Pagrindinė kraštų patikimo filtrų esmė yra aptikti kampus ir kraštus, norint tai padaryti paprasčiausia paveikslų pikselių regionus suskirstyti į tris kategorijas (2 pav.), kuriuose vaizduojama lygumos (*angl. flat*), krašto (*angl. edge*) ir kampo (*angl. corner*) regionai.



2 pav. Pikselių regionai. Kairėje lygumos, per vidurį krašto ir dešinėje kampo

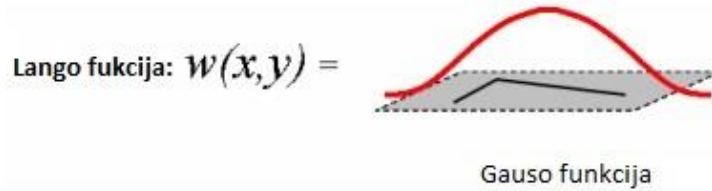
Iš paveikslo matyti, kad lygumos regiono pikseliai visomis kryptimis lieka nepakitę, t.y. nėra jokio ryškaus spalvos pokyčio visomis kryptimis, krašto regionui priklausantiems pikseliams nebūdingas joks pokytis palei krašto kryptį, pokytis vyksta horizontalia kryptimi, o pikseliams priklausantiems kampo regionui, būdingas pokytis visomis kryptimis.

Harris – Stephens filtras buvo sukurtas remiantis Moravec filtru ir norit jį patobulinti.

1. Moravec filtru gaunamas atsakas pasižymi vaizdo triukšmais, dėl tiriamojo langui naudojamos stačiakampės, dvejetainės funkcijos.

Harris – Stephens kampų atpažinimo filtre naudojama tolygiai apvalaus lango Gauso funkcija:

$$w(x, y) = \exp\left(-\frac{(x^2+y^2)}{2\sigma^2}\right) \quad (2)$$



3 pav. Tiriomojo lango funkcija Harris – Stephens kampų atpažinimo filtre

2. Moravec filtre kampai ieškomi ir pikseliai tikrinami tik kas 45° .

Harris ir Stephens šią problemą išsprendė pasinaudodami Teiloro eilute, visiems kaimyniniams pikseliams tikrinti, nesvarbu, koku kampu nuo pagrindinio pikselio jie būtų:

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 = \sum_{x,y} w(x, y) [I(x, y) + uI_x + vI_y - I(x, y)]^2 = \sum u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2 \quad (3)$$

3. Moravec filtre kampų aptikimo jautrumas yra labai mažas, nes ieškomas tik minimali E vertė. Harris ir Stephens performulavimo kampų radimo priemonę, jie pasinaudojo E vertės variacijomis ir poslinkio kryptimis.

Pokytis E, mažiems nuokrypiams (u,v) gali būti užrašytas:

$$E(u, v) \cong [u, v] M \begin{bmatrix} u \\ v \end{bmatrix} \quad (4)$$

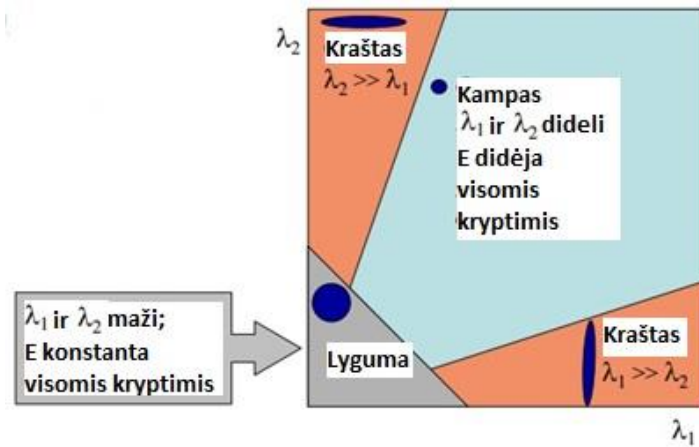
Kur M yra 2×2 dydžio matrica ir gaunama pasinaudojus 3 formulės galutine išraiška:

$$M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (5)$$

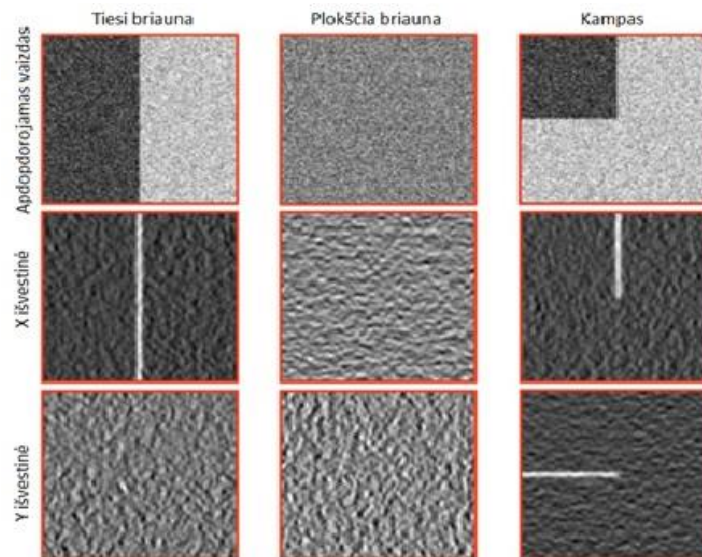
I_x ir I_y nesunkiai gali būti rastos naudojant Sobelio operatorių. Informacija apie Sobelio operatorių pateikta skyrelyje žemiau.

Tarkim, kad λ_1 ir λ_2 yra tikrosios M reikšmės (*angl. eigenvalues*). λ_1 ir λ_2 yra proporcingos pagrindiniams vietinės autokoreliacijos (pagal [8] arba matematiškai skirtumų tarp nagrinėjamų taškų kvadrato sumos) kreivių kreiviams (*angl. curvature*). Kaip ir anksčiau yra galimi trys kampų aptikimo atvejai, naudojantis λ_1 ir λ_2 :

- A. Jei λ_1 ir λ_2 yra mažos, tai vietinės autokoreliacijos funkcijos aptiktas rezultatas yra „lyguma“.
- B. Jei viena λ reikšmė didelė, o kita maža, tai tiriama vieta atitinka kraštą.
- C. Jei abi λ reikšmės yra didelės, tai tiriamoje vietoje, aptinkamas kampas.



4 pav. Kampų aptikimas pagal λ_1 ir λ_2



5 pav. Teritorijų atvaizdavimas x ir y ašyse

Kampų ir kraštų nustatymui reikia ne tik klasifikavimo regionų (kraštas, kampas, lyguma), bet taip pat reikia įvertinti kampo ir krašto kokybę ir atsaką. Atsako dydis naudojamas nustatyti izoliuotus kampų pikselius ir suploninti kraštų pikselius.

Atsakas R priklauso nuo λ_1 ir λ_2 reikšmių. Atsakui apskaičiuoti naudojamas M matricos determinantas ($\text{Det}(M)$) ir diagonalė matrica ($\text{Tr}(M)$), šios reikšmės taip pat gaunamos iš λ_1 ir λ_2 :

$$\text{Tr}(M) = \lambda_1 + \lambda_2 \quad (6)$$

$$\text{Det}(M) = \lambda_1 \lambda_2 \quad (7)$$

Kampo atsakas R gaunamas taip:

$$R = \text{Det}(M) - k(\text{Tr}(M))^2 \quad (8)$$

kur k – empiriškai nustatoma konstanta, $k = 0.04 - 0.06$.

Atsako R savybės:

- R priklauso tik nuo tikrųjų M reikšmių λ_1 ir λ_2 .
- Kai R reikšmė didelė, vadinasi turimas kampas.
- Kai R reikšmė neigiama ir didelė, turimas kraštas.
- Kai $|R|$ reikšmė maža, vadinasi turimas lygumos regionas.

Harris – Stephens kraštų aptikimo filtro algoritmas:

1. Apskaičiuoti x ir y krypčių intensyvumus:

$$I_x = G_{\sigma}^x \cdot I; \quad I_y = G_{\sigma}^y \cdot I \quad (9)$$

2. Apskaičiuoti kiekvieno pikselio intensyvumų produktus:

$$I_{x2} = I_x I_x; \quad I_{y2} = I_y I_y; \quad I_{xy} = I_x I_y \quad (10)$$

3. Apskaičiuoti kiekvieno pikselio intensyvumų produktų sumas:

$$S_{x2} = G_{\sigma1} \cdot I_{x2}; \quad S_{y2} = G_{\sigma1} \cdot I_{y2}; \quad S_{xy} = G_{\sigma1} \cdot I_{xy} \quad (11)$$

4. Kiekvieną pikselį (x,y) apibrėžti matrica:

$$H(x, y) = \begin{bmatrix} S_{x2}(x, y) & S_{xy}(x, y) \\ S_{xy}(x, y) & S_{y2}(x, y) \end{bmatrix} \quad (12)$$

5. Apskaičiuoti detektoriaus atsaką R kiekvienam pikseliui:

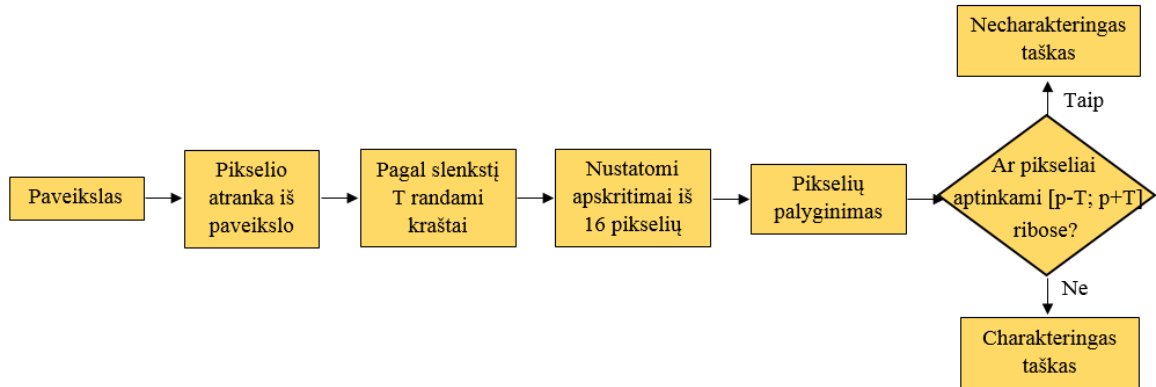
$$R = Det(M) - k(Tr(M))^2 \quad (13)$$

6. Patikrinti R su slenkstinėmis vertėmis. Apskaičiuoti nemaksimalų slopinimą.
[20][21][22][23][46]

1.3.3 FAST Features vaizdo apdorojimo metodas

FAST Features (*angl. Features from Accelerated Segment Test*) vaizdo apdorojimo metodas yra skirtas aptikti charakteringus vaizdo taškus, kurių padėtis yra apibrėžta ir kurie stabiliai randami tiriant skirtingus tos pačios vietos ar to pačio objekto vaizdus. Charakteringi taškai turi didelį kiekį informacijos apie vietą, kurioje jis yra, todėl šie taškai labai tinka naudojimui objektų aptikimui ir atpažinimui skirtinguose vaizduose bei vietos ar atstumo nustatymo algoritmuose [10]. Neretai vaizde esančių objektų kraštų ar kampų radimui naudojami ir kiti, jau anksčiau aprašyti metodai, tokie kaip Harris & Stephens, Moravec, Canny edge ar kiti, bet FAST Features iš visų šių algoritmų išsiskiria

tu, kad jis pasižymi ypač didele charakteringųjų taškų aptikimo sparta. 6 pav. pavaizduota FAST Features metodo charakteringųjų taškų veikimo principas [11]:



6 pav. FAST Features metodo charakteringųjų taškų veikimo principas

Iš 6 pav. matyti, kad FAST Features vaizdo apdorojimo metodas vykdomas keliais žingsniais. Pirmajame iš jų, gavus paveikslą, analizuojami jo pikseliai, gavus pirmąjį iš jų (pasirenkamas pirmas iš eilės arba jau po tam tikro atrinkimo likęs pirmasis pikselis) programa palygina jo pilkumo intensyvumą su jo kaimyniniais pikseliais. Šiame žingsnyje daroma prielaida, kad pikselio spalvos intensyvumas turi būti I_p , tam kad jis būtų priskirtas prie charakteringųjų taškų. Antrame žingsnyje nustatoma pikselio spalvos intensyvumo slenkstinė riba T ir trečiame žingsnyje atrinktas pikselis lyginamas su 16 kaimyninių pikselių, kurie sudaro apskritimą aplink tiriamąjį tašką. Taškas teikia informaciją tuo atveju, jei jo intensyvumas n kaimyninių pikselių (n yra intervale nuo 6 iki 12) yra mažesnis arba didesnis už kaimyninių pikselių spalvos intensyvumą I_p . Metodo autoriai nustatė, kad norint užtikrinti optimalų algoritmo įvykdymą pirmiausia lyginami 4 kaimynų iš apskritimo (I_1, I_5, I_9 ir I_{13}) intensyvumas su tikrinamo taško intensyvumu I_p , šiame tikrinime bent 3 iš 4 pikselių turi tenkinti sąlygą (būti mažesnio arba didesnio intensyvumo) nei tikrinamas taškas, tam kad tikrinimas būtų vykdomas toliau. Kiekvienas pikselis, kuris yra analizuojamas gali turėti 3 skirtingas būsenas, pagal kurias jis priimamas arba atmetamas iš tolesnio tyrimo:

- 1) $I_x \leq I_p - T$ – tokiu atveju priimami tamsesni pikseliai;
- 2) $I_p - T < I_x < I_p + T$ – tokiu atveju atmetami panašūs pikseliai;
- 3) $I_p + T \leq I_x$ – tokiu atveju priimami ryškesni pikseliai;

I_x – kaimyninis pikselis iš apskritimo, jis lyginamas su tikrinamu pikseliu I_p ir slenkstine verte T , pagal šiuos kriterijus tikrinamasis pikselis gali būti atrinktas arba atmetamas. Kai jau minėta anksčiau, jei bent 3 iš 4 pikselių (I_1, I_5, I_9 ir I_{13}) reikšmių nėra mažiau už $I_p - T$ arba daugiau už $I_p + T$, tai ir I_p yra necharakteringas taškas ir yra atmetamas iš tolesnio nagrinėjimo. Tokiu būdu yra tikrinami visi

paveikslo pikseliai, bet toks algoritmas turi ir trūkumų: ne visada galima aptikti tokį didelį kiekį pikselių, be to, toks didelis kaimyninių pikselių radimo ir nagrinėjimo skaičius mažina algoritmo vykdymo greitį, šis taip pat priklauso ir nuo pasirenkamos slenkstinės ribos T dydžio: kuo riba mažesnė, tuo daugiau požymių aptinkama ir tuo labiau lėtėja metodo greitis. Dėl to, kad metodas palaipsniui tikrina visus paveikslo pikselius, tame pačiame tikrinamo apskritimo plote gali atsirasti ne vienintelis charakteringas taškas, norint sumažinti aptinkamų charakteringųjų taškų skaičių, pikseliai dar karta perfiltruojami, naudojant nemaksimalaus slopinimo (*ang. Non-Maximal Suppression*) matematinę išraišką:

$$F = \max \begin{cases} \sum(I_x - I_p), \text{ jeigu } ((I_x - I_p) > T \\ (I_p - I_x), \text{ jeigu } ((p - I_x) > T \end{cases} \quad (14)$$

I_p – tikrinamas pikselis, esantis centre apskritimo, I_x – kaimyninis pikselis iš apskritimo. Kai F apskaičiuojamas kiekvienam pikseliui, daroma prielaida, kad jei šalia yra du charakteringi taškai (F reikšmės skiriasi nedaug), atmetamas tas taškas, kuris yra mažesnis [10][11][12].

1.4 Sobelio kraštų išskyrimo metodas (filtravimas)

Sobelio operatorius aproksimuoja vaizdo nuolydžio gradientą. Jis gaunamas skaičiuojant suvidurkintas dalinių išvestinių horizontalia ir vertikalia kryptimis reikšmes. Operatorius naudoja dvi konvoliucines kaukes/filtrus (arba branduolius), viena iš jų skirta vertikaliam, o kita horizontaliam vaizdo nuolydžiui nustatyti (abi kaukės yra identiškos, tik viena iš jų pasukta 90° kampu, kitos kaukės atžvilgiu). Matuojant vaizdo nuolydį, paryškinami tos vaizdų vietos, kuriose nustatomas didžiausias erdvinis dažnis; tos vietos dažniausiai atitinka vaizdo kraštus. Abi kaukės yra 3×3 taškų dydžio.

Šie filtrai sukonstruoti taip, kad duotame vaizde leistų greitai ir teisingai aptikti horizontalius ir vertikalius kraštus. Naudojant šiuos filtrus po vieną lengvai galima apskaičiuoti kiekvienos krypties nuolydį atskirai, žinant abiejų kryptių rezultatus galima nesunkiai rasti kiekvieno pikselio nuolydžio reikšmę ir jo kryptį. Galimi du absoliutaus nuolydžio skaičiavimo būdai [14][13]:

$|G| = \sqrt{Gx^2 + Gy^2}$ – tikslus, tačiau lėtai skaičiuojamas;

$|G| = |Gx| + |Gy|$ – apytikslis, tačiau greičiau skaičiuojamas.

Krašto krypties kampas apskaičiuojamas: $\theta = \arctan \frac{Gy}{Gx}$.

1.4.1 Sobelio metodas 3x3

Dažnai absoliučios reikšmės būna vienintelis metodo rezultatas, kadangi patogų apjungti abu konvoliucinius filtrus į vieną ir vienu metu atlikti skaičiavimus pagal tokią pseudo-filtrą:

P1	P2	P3
P4	P5	P6
P7	P8	P9

Absoliutinis gradientas skaičiuojamas visomis kryptimis. G_x skaičiuojamas kryptimi P7 į P3, P9 į P1 ir P4 į P6, G_y skaičiuojamas kryptimis P9 į P1, P3 į P7 ir P2 į P8. Gradientas taip pat padaugintas iš atstumo tarp ieškomo taško iki centro:

$$Gx' = P3 + P9 - P7 - P1 + 2 \cdot (P6 - P4)$$

$$Gy' = P9 + P7 - P1 - P3 + 2 \cdot (P8 - P2)$$

Turint daugiklius prie kiekvienos pseudo-filtro P reikšmės gaunamas šio proceso rezultatas lygus intensyvumo vektoriams. Apačioje parodyta vertikali ir horizontali svorio funkcijos kryptys. Kairėje vertikalus (G_x), dešinėje horizontalus (G_y) svorio filtrai [11][13]:

-1	0	+1
-2	0	+2
-1	0	+1

-1	-2	-1
0	0	0
1	2	1

1.5 Gauso suliejimas

Gauso suliejimas (*angl. Gaussian smoothing*) – vaizdo apdorojimo operacija, kuria lengvinamas paveikslo triukšmas. Suliejimas remiasi konvoliucijos operatoriaus (*angl. convolution operator*) skaičiavimu su Gauso pasiskirstymo kauke [9] [17].



7 pav. Gauso suliejimas. Paveikslėlis prieš ir po [9]

Gauso suliejimas – vaizdo suliejimas, naudojant Gauso funkciją ($f(x) = a \cdot e^{-\frac{(x-b)^2}{2c^2}}$, kur a, b ir c yra realios konstantos). Tai plačiai paplitęs metodas, kurio rezultatas yra sumažintas vaizdo triukšmas. Vizualinis šio suliejimo efektas yra sulietas vaizdas, kuris primena žiūrėjimą į tą patį paveikslą per permatomą ekraną, tik pasinaudojus Gauso suliejimu nėra gaunamas natūralus šešėlis, kuris yra trikdis toliau apdorojant paveikslą. Gauso suliejimas taip pat naudojamas kaip pradinis žingsnis vykdant vaizdų atpažinimo algoritmą, siekiant išryškinti vaizdo struktūras.

Gauso suliejimas yra vaizdo suliejimas, kuris naudoja Gauso funkciją (kuri taip pat išreiškia įprastą duomenų pasiskirstymo statistiką) kiekvieno vaizdo pikselio transformacijos apskaičiavimui. Gauso funkcija 1D vaizdui atrodo taip:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{x^2}{2\sigma^2}} \quad (15)$$

Dviejų dimensijų (2D) vaizde formulė susideda iš dviejų tokių formulių, po vieną kiekvienai dimensijai:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (16)$$

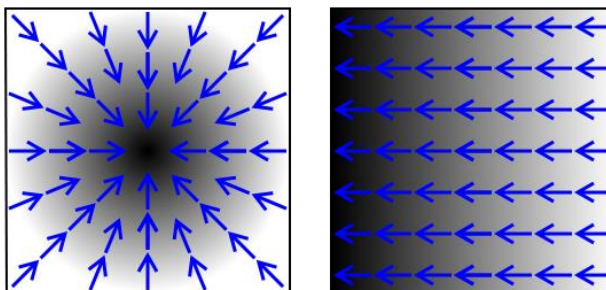
kur x yra atstumas nuo pradžios iki taško horizontalioje ašyje, o y – vertikalioje, σ – standartinis nuokrypis Gauso pasiskirstyme.

Kai Gauso suliejimas taikomas 2D vaizdui, formulė sukuria koncentrinio apskritimo kontūrą su Gausiniu pasiskirstymu nuo to apskritimo centro taško. Šio pasiskirstymo dydžiai yra naudojami kuriant konvoliucijos matricą, kuri taikoma turit pradinį vaizdą. Kiekvienas naujai gautas pikselis yra suvidurkinta šalia jo buvusių pikselių reikšmė. Originali pikselių reikšmė turi didžiausią svorį

(didžiausią Gauso skirstinio vertę), o jo kaimyninis pikselis turi mažesnes Gauso skirstinio vertes (kuo atstumas nuo didžiausią vertę turinčio pikselio didesnis, tuo kaimyninio pikselio vertė mažesnė). Būtent dėl tokio pasiskirstymo Gauso suliejimo rezultatas gaunamas vaizdas atrodo kaip per miglą, tokiu būdu geriau išsaugomos kraštų ribos ir kampai [18] [19].

1.6 Gradientas

Gradientas (8 pav.) – diferencialinis operatorius, skaliarinį lauką atvaizduojantis į vektorinį lauką taip, kad kiekvienas vektorinio lauko vektorius būtų nukreiptas skaliarinio lauko reikšmių didėjimo kryptimi, o jo modulis būtų lygus kryptinei išvestinei šiame taške. Vaizdo gradiento gavimas vienas iš svarbiausių vaizdo apdorojimo procesų. Jie dažnai naudojami žemėlapiuose, vaizdinių duomenų vaizdavime, kur reikalingas informacijos perdavimas.



8 pav. Gradientas

Dviejų ar daugiau kintamųjų funkcijų gradientas kiekviename taške yra dvimatis vektorius, išvestas vertikalia ir horizontalia kryptimis. Kiekviename pikselyje su didžiausia gradiento reikšme intensyvumas didėja ir jo ilgis atitinkamai kinta vektoriaus intensyvumo kryptimi [20]. Norint gauti dar tikslesnes gradiento reikšmes, naudojamas aukščiau aprašytas metodas su 3x3 ir 5x5 dydžio filtrais ir iš jo gautomis formulėmis kiekviename turimo vaizdo pikselyje. Teoriškai pastebima, kad spalvos nuolydis (gradientas) turėtų būti didelis objekto kraštuose, ir turėtų mažėti, artėjant į vidurį.

Matematiškai gradientas aiškinamas taip: Sakykime, kad funkcija yra diferencijuojama taške $M(x; y; z)$. Funkcijos gradientu taške M , vadinsime vektoriu, kurio koordinatės yra funkcijos dalinės išvestinės. Žymėsime gradu arba ∇u . Tuomet

$$\text{grad} u = \left(\frac{\partial u}{\partial x}; \frac{\partial u}{\partial y}; \frac{\partial u}{\partial z} \right) = \frac{\partial u}{\partial x} \vec{i} + \frac{\partial u}{\partial y} \vec{j} + \frac{\partial u}{\partial z} \vec{k}. \quad (17)$$

Funkcijos u gradientas ∇u apibūdina tos funkcijos greičiausio augimo (pakilimo) kryptį ir dydį duotame taške [21][22].

1.7 Objektų panašumas

Objektų požymių panašumo matai gana sunkiai formuluojamas matas, kadangi skirtingais atvejais panašumas gali būti vertinamas skirtingais požymiais. Panašumas priklauso nuo vertinamų požymių tipo, pasirinkto mato ar matavimo skalės. Panašiausi rezultatai gaunami, kai pasirenkamas kiekybinis matas (kai objektus apibūdinantys požymiai gaunami naudojant metrinius atstumo matus) [24]. Metriniai matai naudojami kai, panašumo požymiai ieškomi geometrinėje erdvėje. Šie matai gali būti vertinami kaip skirtingumo matai, nes objektai mažiau panašūs tada, kai gaunama reikšmė didesnė. Praktikoje dažnai naudojama taksi geometrija (dar gali būti vadinama Minkovskio atstumu), Euklido ar Manhatano atstumai [23][24].

1.7.1 Taksi geometrija

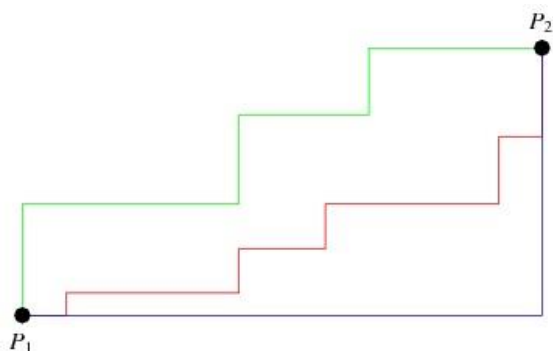
Taksi geometrija, pasiūlyta Hermann Minkovskio 19a., yra geometrijos forma, kurioje įprasta atstumo funkcija metrinėje arba Euklido geometrijoje pakeičiamas nauja metrika, kurioje atstumas tarp dviejų taškų yra absoliučių sumų skirtumas nuo jų sumų skirtumo Dekarto koordinatų sistemoje [25]. Taksi geometrijos atstumas tarp dviejų objektų $X_k = (X_{k1}, X_{k2}, \dots, X_{kn})$ ir $X_l = (X_{l1}, X_{l2}, \dots, X_{ln})$, apskaičiuojamas pagal formulę:

$$d_q(X_k, X_l) = \sqrt[q]{\sum_{j=1}^n |x_{kj} - x_{lj}|^q} \quad (18)$$

q – fiksuotas skaičius.

1.7.2 Manhattan'o atstumas

Taksi geometrijos metodas, kai $l = 1$ vadinamas ir Manhattan'o atstumu, yra atstumas, kuris paprastai yra kelias nuo taško P_1 iki taško P_2 (9 pav.):



9 pav. Manhatano atstumas

Manhattan'o atstumas Euklidinėje koordinačių sistemoje apibrėžiamas kaip:

$$g((\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2)) = |\mathbf{x}_1 - \mathbf{x}_2| + |\mathbf{y}_1 - \mathbf{y}_2| \quad (19)$$

Visiems taškams, esantiems tarp $P_1(x_1, y_1)$ ir $P_2(x_2, y_2)$. Šis skaičius yra lygus visų horizontalių ir vertikalinių takų (kelių), jungiančių P_1 ir P_2 taškus ilgiui. Naudojantis šiuo metodu niekada negalima keliauti atgal, reikia judėti taip, kaip juda automobilis „grotelių/bloku“ tipo gatvėje [26].

1.7.3 Euklido atstumas

Matematikoje Eulido atstumas yra trumpiausias atstumas tarp dviejų taškų Trimatėje erdvėje – tai taškus jungianti tiesė.

Euklido atstumas gali būti taikomas įvairių kiekių dimensijoms:

1D vaizdui atstumas tarp dviejų taškų yra reali linija, kurios absoliutinė vertė yra tų taškų skirtumas. Taigi, jei x ir y yra du realios linijos taškai, tai atstumas tarp jų apskaičiuojamas pagal formulę:

$$\sqrt{(x - y)^2} = |x - y| \quad (20)$$

2D vaizdui:

Jei $p = (p_1, p_2)$ ir $q = (q_1, q_2)$ tai atstumas gali būti apskaičiuojamas pagal:

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \quad (21)$$

N dimensijų (ND) vaizdui atstumas apskaičiuojamas taip:

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_i - p_i)^2 + \dots + (q_n - p_n)^2} \quad (22) [27]$$

Atstumas tarp objektų gali būti skaičiuojamas ir naudojantis stereometrijoje pateiktais metodais.

1.8 Stereometrija

Stereometrija – tai geometrijos šaka, tirianti erdvines (trimates) figūras. Stereometrijoje tiesėms erdvėje rasti neretai naudojamos tiesės parametrinės ir kanoninės lygtys. Rasti parametrinę tiesę, patogiu naudojant per du taškus einančio vektoriaus išraišką:

$$\vec{r}_k = \langle \vec{x}; \vec{y}; \vec{z} \rangle + t \langle x; y; z \rangle \quad (23)$$

čia \vec{r}_k – vektorius tarp dviejų taškų, \vec{x} – pirmojo taško koordinatė erdvėje, \vec{y} – pirmojo taško koordinatė erdvėje, \vec{z} – pirmojo taško koordinatė erdvėje, x, y ir z yra atitinkamai antrojo taško koordinatės erdvėje, t – laiko kintamasis.

Norint rasti tiesės lygtį, papildomai ją reikia apsisrašyti kaip tiesę, einančią per du taškus:

$$\frac{x-x_1}{x_2-x_1} = \frac{y-y_1}{y_2-y_1} = \frac{z-z_1}{z_2-z_1} \quad (24)$$

čia x, y ir z – kintančios koordinatės erdvėje, x_1, y_1, z_1 – pirmojo taško koordinatės erdvėje, x_2, y_2, z_2 – antrojo taško koordinatės erdvėje.

Tiesės padėtį erdvėje galima nustatyti, jei yra žinomas jos taškas $M_0(x_0, y_0, z_0)$ ir tam taškui lygiagretus tiesės krypties vektorius $\vec{s} = l \cdot \vec{i} + m \cdot \vec{j} + n \cdot \vec{k} = \{l; m; n\}$. Norint sudaryti tiesės lygtis, imamas bet koks taškas $M(x, y, z)$ esantis tiesėje.

Tiesės parametrinės lygtys:

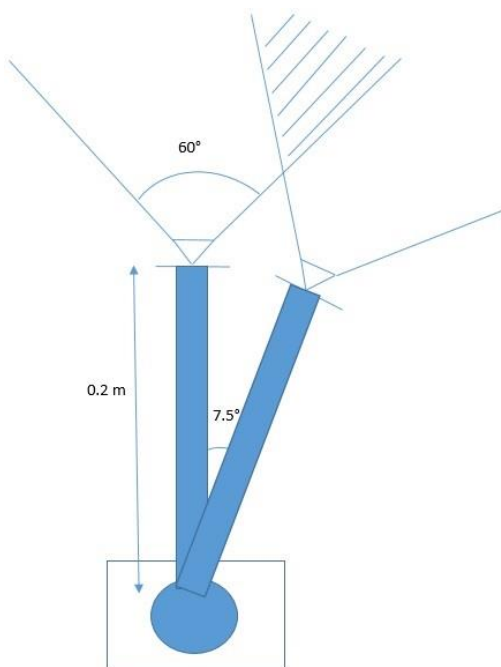
$$\begin{cases} x = lt + x_0 \\ y = mt + y_0 \\ z = nt + z_0 \end{cases} \rightarrow \vec{s} = (l, m, n) \quad M_0(x_0, y_0, z_0) \quad (25)$$

Tiesių tarpusavio lygiagretumą galima patikrinti iš savybės:

$$\vec{s}_1 \parallel \vec{s}_2: \frac{l_1}{l_2} = \frac{m_1}{m_2} = \frac{n_1}{n_2} \quad (26)$$

Objekto lokalizacija trijose dimensijose geriau žinoma kaip lokalizacija ir vietos žemėlapiu sudarymas tuo pat metu. Šio metodo esmė yra ta, kad vienu metu gali būti stebima aplinka ir gali būti žinoma sekamo objekto padėtis aplinkoje. Šioms užduotims įgyvendinti reikia apskaičiuoti parametrines tiesių lygtis. Jas išsprendus galima nustatyti objekto koordinates trijose dimensijose [28].

Tiriamajame darbe filtro tyrimui naudojama schema schematiškai pavaizduota 10 pav. Schemoje matyti žingsninis variklis, puse X formos iškyšos ant kurios primontuotos kameros ir kamerų matymo laukas (60°). Užbrūkšniuotas plotas yra vaizdas, kuris persidengia abiejuose vaizduose ir daiktai esantys tame plote bus matomi bendrai abiejuose nuotraukose, atstumai, randami remiantis stereometrija, iki jų taip pat bus matomi bendrai abiejuose nuotraukose.

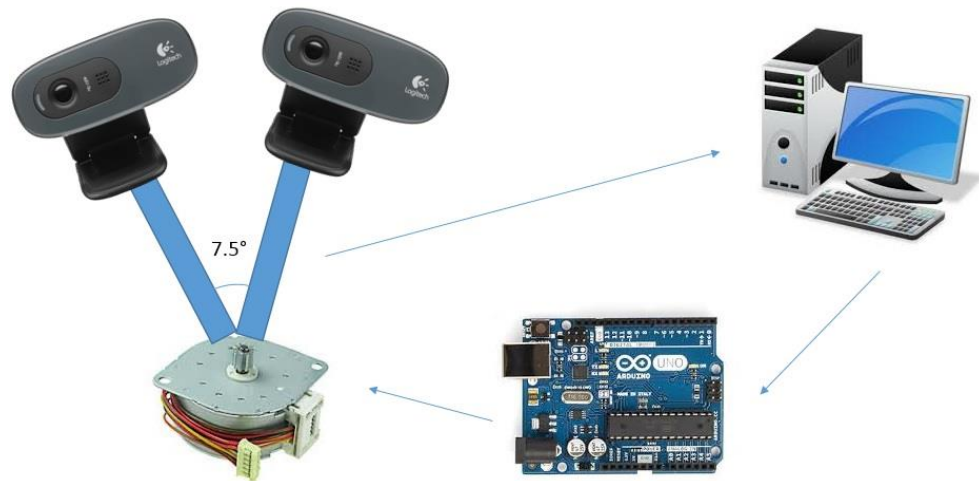


10 pav. Struktūrinė filtro tyrimo schema

Nufotografavus du vaizdus ir programiškai atlikus visus skaičiavimus sužinoma kiekvieno pikselio koordinatės, pasinaudojus jomis, bei formulėmis aukščiau ir informacija, gauta iš kalibravimo eksperimento (kalibravimo eksperimentas smulkiai aprašytas eksperimentinėje dalyje), galima atrasti kiekvieno norimo taško atstumą iki kameros bei atstumą vienas tarp kito.

2 Naudojamų technologijų analizė

Norint tinkamai ištestuoti sukurtą programinę įrangą, reikia nuspręsti, kaip tinkamai realizuoti techninę įrangą, tam kad būtų pasiektas geriausias rezultatas. Sukurta techninė schema (11 pav.) susideda iš dviejų internetinių kamerų, kurios yra iškeltos ant žingsninio variklio, 7.5° kampu viena nuo kitos, kuris yra valdomas Arduino Uno valdikliu, valdant žingsninį variklį galima sukoti kameras 360° kampu. Šiame skyriuje bus analizuojamos šios bei kitos esamos techninės realizacijos ir jų komponentai.



11 pav. Supaprastinta duomenų perdavimo schema

2.1 Jutikliai, vertinantys aplinką

Vaizdo kameros – vieni sudėtingiausių iš šiuo metu egzistuojančių jutiklių. Jas naudojant įmanoma gauti šimtus tūkstančių signalo reikšmių, didesniu nei 30 kadrų per sekundę greičiu. Šiuo metu pačios populiariausios RGB spektro vaizdo kameros. Pagal komponentų skaičių, esantį sistemoje išskiriamos vienos (*angl. monocular*), dviejų (*angl. stereo*) ar dar daugiau kamerų sistemų [30][29]. Tiriamam uždaviniui spęsti svarbiausi parametrai skiriamoji geba, kampas tarp kameros lęšių, fokusavimo atstumas. Norint sumažinti vaizdo apdorojimo spartą, neretai pasirenkama kuo mažesnė vaizdo raiška. Stereo kamerų privalumas yra tas, kad sistema papildomai leidžia ištirti vaizdo gylį (naudojant trianguliacijos metodą), tačiau norint pasiekti kuo didesnę tikslumą, naudojamos kameros turi būti kuo labiau identiškos.

Eksperimento metu buvo pasirinkta dviejų kamerų sistema, naudojanti internetines kameras (*angl. webcam*). Internetinė kamera (fotokamera ar vaizdo) dažniausiai jungiama per USB jungtį prie kompiuterio, interneto pagalba leidžianti perteikti vaizdą realiu laiku. Internetinę vaizdo kamerą paprastai sudaro lęšis, vaizdo jutiklis ir atitinkama schema. Pačios pirmosios internetinės kameros turėjo mažą skiriamąją gebą (352x288), tam kad naudotų kuo mažiau interneto resursų ir neapkrautų interneto greičio. Didėjant interneto spartai, paskutiniaisiais metais vis dažniau gaminamos didesnės (nuo 640x480) skiriamosios gebos kameros [32].

Išnagrinėjus daugelio internetinių kamerų technines specifikacijas buvo nuspręsta eksperimentą atlikti su Logitech C270 internetinėmis kameromis (12 pav). Kameros techninės specifikacijos pateiktos 1 lentelėje.



12 pav. Logitech C270 internetine kamera

1 lentelė. Logitech C270 internetinės kameros techninės specifikacijos

Matmenys (IxPxA), cm	21.1 x 15.2 x 7.9
Sąsaja	USB
Vidaus/lauko naudojimo	Dažniausiai naudojama viduje, bet galima naudoti ir lauke
Darbo temperatūra	+5°C – +40°C
Raiška, pikseliai	iki 1280x720
Kadrų dažnumas	30 kadrų per sekundę (su rezoliucija 1280x720)
Raiška	720 pHD
Matymo laukas	60 laipsnių

2.2 Kamerų pasukimo mechanizmas

Norint paleisti kamerų padėti, naudojamas tam tikru kampu pasisukti galintis mechanizmas. Šiam tikslui puikiausiai tinka servo pavaros arba žingsniniai varikliai.

2.2.1 **Servo pavara** (13 pav.) – tai elektros pavara, valdoma neigiamo grįžtamojo ryšio pagalba, leidžiančio kontroliuoti pavaros judėjimo parametrus, tokius kaip greitis ar pozicija. Pagrindinis servo pavaros bruožas: jutikliai, reaguojantys į greičio, pozicijos, fizinio poveikio pokyčius, ir valdymo blokas, priimančias signalus iš jutiklių bei operatoriaus [33] Servo pavaros taip pat pasižymi dideliu kampiniu tikslumu ir yra valdomi impulsais. Kiekvienu impulsu servo pavara pasisuka tam tikru fiksuotu kampu, o priklausomai nuo rūšies gali atlaikyti 2.6–8.4 Kg•cm sukimo momentą prie 4.8 V [34].



13 pav. Servo pavara

2.2.2 **Žingsninis variklis** (14 pav. kairėje) – elektros variklis, kurio ašis apsisuka per daug vienodų žingsnių. Toks variklis gali atlikti nustatytą reikiamą kiekį žingsnių, rotoriumi užimant norimą padėtį. Žingsninis variklis keičia elektrinį signalą į reikiamą sukimą ar tiesinį judėjimą [35]. Žingsniniai varikliai pasižymi gana tikslu savo padėties nustatymu be papildomų padėties jutiklių, greičio ir padėties valdymu skaitmeniniais valdymo impulsais, dideliu sukimo momentu, esant mažam sukimosi greičiui, galimybe palaikyti rotoriumi fiksuojamą padėtyje be stabdymo [36]. Darbe naudojamas žingsninis variklis per vieną žingsnį pasisuka 7.5° , žingsninis variklis valdomas *EasyDriver* 4.4 žingsninių variklių valdikliu (14 pav., dešinėje).



14 pav. Žingsninis variklis (kairėje) ir EasyDrive 4.4 valdiklis (dešinėje)

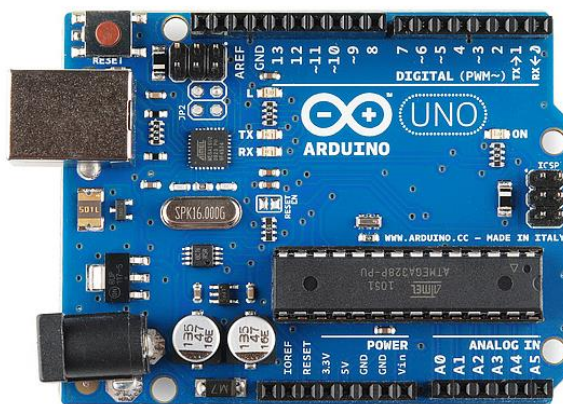
2.3 Realizuojančios platformos

Dažniausiai paprastų sistemų valdymui, įvairių skaičiavimų atlikimui naudojamos įvairios mikroprocesorinės sistemos ar mikrokompiuteriai. Šiuo metu tokių sistemų pasiūla yra be galo didelė, vartotojai gali rinktis tokias valdymo platformas kaip: Arduino, Raspberry Pi, BeagleBone ir kt. Visi jie yra tinkami valdymui ir puikiai atlieka savo funkciją, didžiausi skirtumai yra techninės specifikacijos ir kainos.

Žingsninio variklio valdymui buvo pasirinktas Arduino Uno valdiklis (15 pav.). Arduino – viena iš populiariausių dabartinių valdymo platformų. Tai atviro kodo platforma, kuri naudojama kuriant įvairius elektronikos projektus. Arduino savo sudėtyje turi tiek programuojamąją plokštę (vadinamą mikrovaldikliu), tiek programinę įrangą ar kitaip vadinamą IDE (integruotą programos

kūrimo aplinką), kuri veikia kompiuteryje ir yra naudojama rašyti ir įkelti programos kodą į Arduino valdiklį.

Arduino platforma yra labiausiai pamėgta eksperimentuotojų dėl savo paprastumo. Skirtingai nei daugumos ankstesnių programuojamų valdiklių, Arduino nereikia specialios įrangos, norint įkelti ar paremaguoti programinį kodą, užtenka naudoti paprastą USB kabelį. Be to, Arduino IDE naudoja paprastesnę programavimo kalbos C++ versiją, todėl vartotojui lengviau išmokti ir parašyti programą. Taip pat Arduino platformai yra sukurta daug įvairių modulių, naudojimo bibliotekų ir visos jos lengvai prieinamos [37].



15 pav. Arduino Uno mikrovaldiklis

2.4 Programavimo kalbos

Norint aprašyti programavimo kalbą reikia laikytis tam tikrų sintaksinių ir semantinių taisyklių, sintaksė nusako programos teksto standartą, o semantika – sintaksiškai teisingos programos prasmę. Didžiosios dalies programavimo kalbų semantinės ir sintaksinės taisyklės aprašomos jų specifikacijose, dalis iš jų oficialiai įgyvendintos, priimtais ISO standartais, o kitos – dominuojančiais sertifikatais [38][39].

Baigiamojo magistrinio projekto programavimui buvo pasirinkta C# kalba. Ją pasirinkau, nes pradėjusi darbą jau turėjau OpenCV C# apvalkalą, į kuri įsigilinau prieš pradėdama darbą ir tuo atveju man buvo patogiausia ši programavimo kalba, taigi dirbdama su C# programavimo kalva aš sutaupiau laiko, kurį būtų reikėję skirti mokantis kitų programavimo kalbų.

2.4.1. C# programavimo kalba

Pirmoji šios programavimo kalbos versija pristatyta 2000 metais, ją sukūrė du Microsoft darbuotojai: Anders Hejlsberg ir Scott Wiltamuth. Kuriant šią kalbą buvo remtasi C++ ir Java kalbomis, tam kad suderinti dvi pagrindines programavimo savybes: greitumą ir galingumą (C, C++ savybės) ir programos kodo rašymo paprastumą ir greitumą (Java savybės). C# greitaveika yra lėtesnė už C ar C++, bet C# turi aukštesnio lygio funkcijų, kurių neturi C ar C++ kalbos. C# – visiškai objektinė programavimo kalba, su ja galima sukurti įvairaus pobūdžio programas. C# programavimo kalba idealiai tinka kurti programas, kurios turi būti parašytos greitai, tačiau veikimo greitis nėra svarbus.

Kaip ir kiekviena programavimo kalba C# turi tiek privalumų, tiek trūkumų:

Privalumai:

- Ganėtinai greita ir galinga programavimo kalba, kadangi buvo sukurta kaip tarpinė kalba;
- Klasių ir funkcijų apibrėžimas gali būti atliktas bet kokia tvarka;
- Neegzistuoja žiedinė priklausomybė;
- Klasės gali būti apibrėžtos kitų klasių viduje;

Trūkumai:

- Lėtesnis paleidimas;
- Ne tokia lanksti kaip C ar C++ kalbos;
- Tinka naudoti tik Windows operacinėje sistemoje.

Apibendrinant galima pastebėti, kad projekto organizavimas ir programavimas lengvesnis su C# (palyginus su C++ ar C) [39][40][41].

Kurti programai buvo naudotas Microsoft Visual Studio C# kompiliatorius. Tai programa, naudojama norint sukurti kompiuterines programas, kurios skirtos Microsoft Windows šeimos operacinėms sistemoms. Kompiliatorius apjungia C# programavimo kalbą ir .NET Framework. Kompiliatorius yra dviejų variantų ir gali būti naudojamas kaip Microsoft Visual Studio arba Microsoft Visual C# 2010 Express narys [42].

3. Eksperimentinė dalis

Eksperimentinė dalis vyko tolygiai dviem etapais: tuo pat metu buvo kuriamas ir charakteringųjų taškų filtras (eksperimento *software*) ir eksperimentui atlikti reikalinga techninė dalis (*hardware*). Techninė įranga pirmiausia buvo kuriama vienai kamerai (16 pav.), norit patikrinti jos pritaikomumą, veikimo galimybes ir norint tiksliai nustatyti tikslų kampą tarp dviejų (kampas buvo pasirinktas pagal vieną žingsninio variklio žingsnio dydį).



16 pav. Sistema, sukurta pirmiesiems bandymams atlikti

Techninę įrangą sudaro žingsninis variklis, ant jo primontuojama lengvai nuimama ir uždedama 20 cm iškyša (arba kitaip lazda), kurią galima lengvai pasukti, valdant žingsninį variklį. Ant lazdos galo, ties 20–tu centimetru nuo žingsninio variklio ašies, primontuojama internetinė vaizdo kamera, tokiu būdu primontuota kamera visada žiūrės į priekį. Kameros padėtis keičiasi atitinkamai nuo žingsninio variklio paeitų žingsnių, vienas žingsnis atitinka $7,5^\circ$. Žingsninio variklio pasisukimo galiai pastiprinti buvo naudojamas 12 V DC išorinis maitinimo šaltinis. Pirmieji bandymai buvo atliekami žingsniam varikliui paėjus vieną žingsnį, tačiau Arduino valdiklis yra suprogramuotas taip, kad žingsnių skaičių ir pasisukimo kryptį galima nuolat keisti. Taip pat ekrane buvo galima stebėti, kokioje pozicijoje nuo pradinės ir nuo paskutinės padėties yra žingsninis variklis, vadinasi, kiekvieną kartą atlikus judesį aiškiai buvo galima nustatyti žingsninio variklio pasisukimą, o tuo pačiu ir kameros padėti buvusios padėties atžvilgiu.

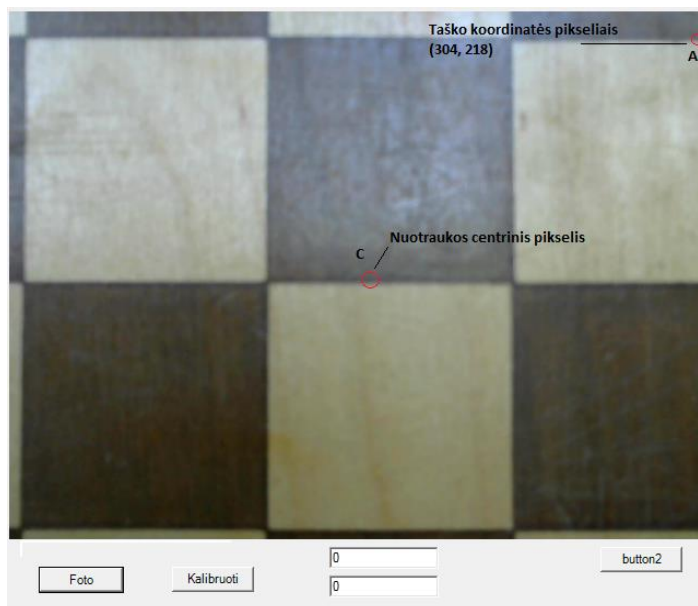
Įsitikinus, kad techninė įranga su viena kamera funkcionuoja gerai, ir atlikus tikslus kampo matavimus, buvo padaryta X formos iškyša (ant kurios galų buvo primontuotos internetinės kameros), kuri ties viduriu gali prisimontuoti prie žingsninio variklio. Kampo tikslumas matuotas tiek su matlankiu, tiek realiai tikrintas pagal vienos kameros sistemos pasisukimą. X formos iškyša ant

žingsninio variklio buvo montuojama tam, kad nereiktų kemerų sistemos judinti ir nešioti, norint pakeisti matymo kryptį. Visa kita įranga buvo lygiai tokia pati kaip ir pavaizduota 15 pav.

Atlikus programos pirmąjį žingsnį, t.y. nusifotografavus dvi tos pačios vietos nuotraukas, skirtingu kampu, atliekamas antras žingsnis, kurio metu atrandamas tam tikras skaičius kampų, esančių nuotraukose (surandamų kampų variantų skaičius randamas užsibrėžus spalvų kontrasto pokyčio (s – step) slenkstinę vertę), trečiuoju žingsniu atrenkamos 6 geriausiai atitinkančios kampų poros abiejuose nuotraukose.

3.1 Kalibravimo eksperimentas

Prieš pradėdant dirbti su internetine kamera atliekamas kalibravimo eksperimentas, tam kad sužinoti tikslų kameros lęšio matymo kampą. Kameros specifikacijose gamintojas pateikia, jog kameros matymo kampas yra 60°, tačiau magistrinio tiriamojo darbo metu tas kampas buvo patikrintas ir gautas eksperimentiškai. Kalibravimo eksperimentui buvo pasinaudota šachmatų lenta (17 pav.)



17 pav. Kalibravimo eksperimentas

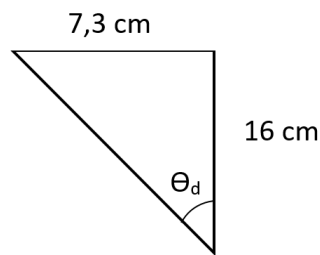
Pradžioje nusistatomas centrinis nuotraukos taškas C ir kampinis atskaitos taškas A (16 pav.) ir sužinomos jų koordinatės pikseliais: C (0; 0) ir A (304; 218). Taip pat išsimatuojamas atstumas nuo kameros lęšio iki šachmatų lentos centrinio taško, atstumas $l_C = 16$ cm, bei šachmatų lentoje išmatuojamas atstumas nuo vietos, kur nuotraukoje yra centrinis taškas iki nuotraukoje pažymėto

kampinio atskaitos taško, atstumas $l_A = 7,3$ cm. Žinant šias reikšmes ir nuotraukos rezoliuciją (640x480), apskaičiuojamos pagrindinės paveikslų įstrižainės, t.y. viso paveikslų įstrižainė ir įstrižainė arba atstumas nuo C iki A taško.

Nuotraukos įstrižainė: $\sqrt{640^2 + 480^2} = 800$ pikselių;

Atstumas tarp C ir A: $\sqrt{304^2 + 218^2} = 374$ pikselių;

Žinant l_C ir l_A bei pasinaudojus trigonometrijos formulėmis, iš stataus trikampio (18 pav.) galima apskaičiuoti dalinį kameros matymo kampą Θ_d , tame pikselėje, kur pasirinktas kampinis atskaitos taškas.



18 pav. Θ_d skaičiavimas

$$\tan \theta_d = \frac{7,3}{16} \rightarrow \theta_d = \arctan \frac{7,3}{16} = 24,53^\circ.$$

Apsiskaičiavus dalinį kameros matymo kampą Θ_d pikselėje bei žinant atstumą pikseliais tarp A ir C taškų, galima pasiskaičiuoti kameros koeficientą ks : $ks = \frac{24,53}{374} = 0,066$.

Pilnas kameros matymo kampas Θ apskaičiuojamas padauginus ks iš pilnos nuotraukos įstrižainės:

$$\theta = \frac{24,53}{374} \cdot 800 = 52,47^\circ.$$

Iš šios formulės matyti, kad pilnas kameros matymo kampas proporcingas daliniam kameros matymo kampui taip kaip pilna nuotraukos įstrižainė proporcinga pasirinktam kampiniam atskaitos taškui nuotraukoje:

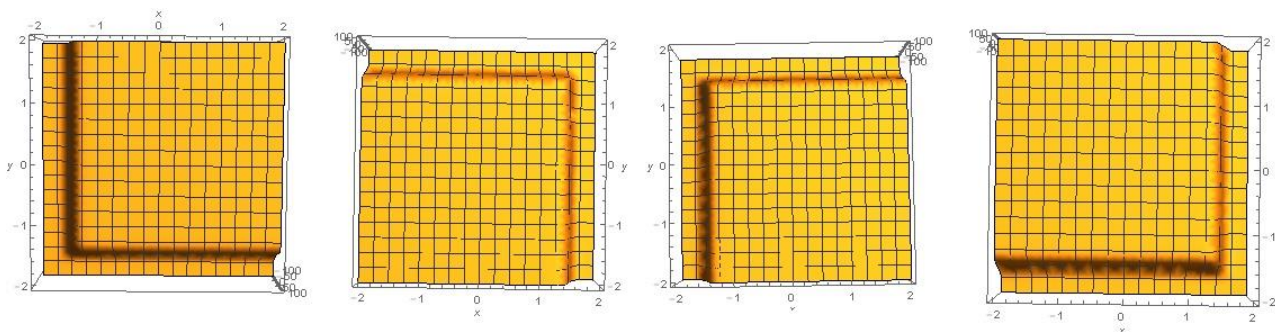
$$\left\{ \begin{array}{l} \theta - \theta_d \\ \text{Nuotraukos įstrižainė} - \text{Atstumas tarp C ir A} \end{array} \right. \quad (27)$$

Kaip jau minėta anksčiau, specifikacijose pateikta, jog kameros pilnas matymo kampas $\theta = 60^\circ$, apskaičiuotas kampas yra mažesnis, tad daroma prielaida, kad žmogiškasis faktorius skaičiuojant duomenis kalibravimo eksperimentui buvo lemiami (neteisingai išmatuoti atstumai tarp kameros ir šachmatų lentos, atstumai šachmatų lentoje pamatuoti neteisingai, taip pat gali būti, kad pati šachmatų

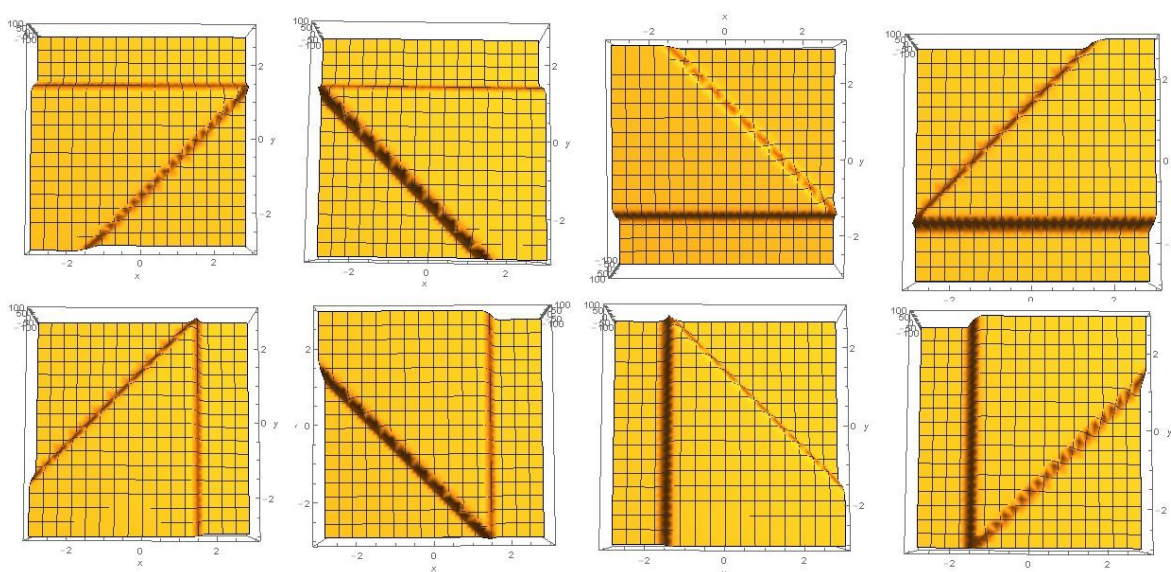
lenta stovėjo netiesiai ir pan.), todėl gauta tokia paklaida. Taigi norit sužinoti tikrą kameros koeficientą tiriamame taške, ks perskaičiuojamas iš naujo: $ks = \frac{60}{800} = 0.075$.

3.2 Charakteringųjų taškų filtras

Kampai paveiksle atpažįstami ir programoje apsirrašomi, kaip dviejų tiesių susikirtimo taškas. Atpažinimui aprašyta 12 kampų variantų: keturi staties kampams (19 pav.) ir aštuoni smailiems kampams (20 pav.). Pirminiame atpažinimo algoritme buvo naudojami ir buki kampai, tačiau vėliau nuspręsta jų atsisakyti, todėl, kad dažniausiai jie buvo atpažįstami ne kaip kampai, o kaip kraštai, esantys vaizde. 19-22 pav. gauti naudojant naudojant Mathematica paketu, pagal aprašytas tieses, tiesių formulės pateiktos Priede1.

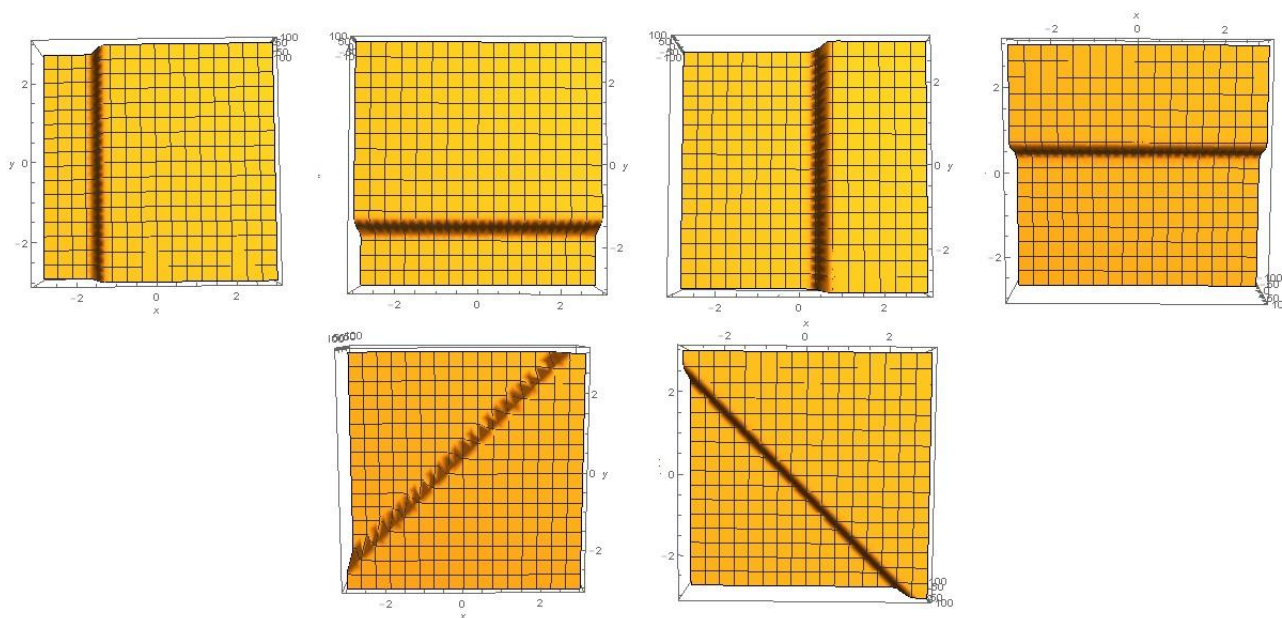


19 pav. Statūs kampai, atpažįstami programoje

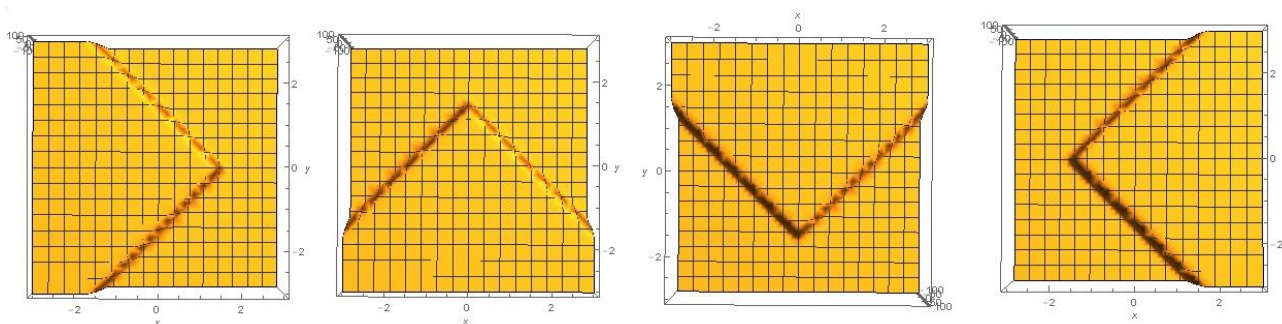


20 pav. Smailūs kampai, atpažįstami programoje

Eksperimentuojant buvo pastebėta ne tik tai, kad buki kampai yra matomi kaip kraštai, esantys paveiksle, taip pat pastebėta, kad kai kuriais atvejais ir smailūs bei statūs kampai yra atpažįstami ir priskiriami kraštams. Šiai problemai išspręsti buvo apsirašyti dar 10 variantų: 6 tiesės (21 pav.) ir 4 statūs 45° pasvirę kampai (22 pav.), konkuruojantys su tiesėmis. Kai vaizde atpažįstami tie 10 variantų, jie atpažinime priskiriami kraštams.



21 pav. Tiesės



22 pav. 45° pasukti statūs kampai

Taigi atpažinimui iš viso naudojami 22 variantai, nors kampams atpažinti yra tik 12 variantų, kiti 10 reikalingi tikrųjų kampų nufiltravimui, t.y. atskyrimui nuo kraštų.

Pridėjus atmetamus variantus, filtras tampa ne tik kampų atpažinimo filtru, jis gali identifikuoti ir kraštines, kurios atliekant šį magistrinį darbą buvo atmetamos ir netiriamos, bet jos yra atpažįstamos. Taigi tas pats filtras gali būti lengvai pakeistas tam, kad atpažintų ne tik kampus, bet ir kraštines, arba kraštines vietoj kampų.

3.3 Kampo aptikimas

Kampui aptikti sprendžiamas optimizavimo uždavinys mažiausių kvadratų metodu. Mažiausių kvadratų metodas – dažniausiai taikomas suglodinimo metodas. Metodą galima apibrėžti tokia formulute: visus koeficientus $\mathbf{a}_k (k = \overline{0, n})$ turi būti apskaičiuoti taip, kad $\mathbf{f}(\mathbf{x}_i)$ ir $\mathbf{F}(\mathbf{x}_i)$ skirtumų kvadratų suma būtų mažiausia, t.y. minimizuota.

$$z = \sum_{i=1}^m (F(\mathbf{x}_i, \mathbf{a}_0, \dots, \mathbf{a}_n) - y_i)^2 \quad (28)$$

(1) Formulė turi vieną ekstremumą, kuris yra apskaičiuojamas naudojantis dviejų lygčių sistema:

$$\frac{\partial z}{\partial \mathbf{a}_k} = \mathbf{0}, \quad k = \overline{0, n}. \quad (29)$$

(3) sistemos sprendimą galima palengvinti:

1. jei $F(\mathbf{x}_i, \mathbf{a}_0, \dots, \mathbf{a}_n)$ yra n -tojo laipsnio polinomas, tai lygčių sistema yra tiesinė ir jos sprendimas nesudaro sunkumų;
2. galima taikyti ištiesinimo metodą: atitinkamai parinktoje koordinačių sistemoje (\mathbf{X}, \mathbf{Y}) taškai $(x_i, y_i, \text{čia } i = \overline{1, m})$ apytiksliai tenkina tiesės. $\mathbf{Y} = \mathbf{kX} + \mathbf{b}$ lygtį, tada mažiausių kvadratų metodas toje sistemoje realizuojamas paprastai [43].

Mažiausio kvadrato metodo paaiškinimas charakteringųjų kampų aptikimui:

Uždavinys sprendžiamas vienam spalvos sluoksniui, pvz.: pilkos spalvos sluoksniui (n spalvotame paveikslėlyje). Pradėmis sąlygomis turime lokalią 5x5 dydžio matricą, kuri bet kuriuo laiko momentu x ir y atžvilgiu atrodo taip:

x	∈	-2	-1	0	+1	+2
---	---	----	----	---	----	----

y	∈	-2
		-1
		0
		+1
		+2

Pikslio įvertis, kurio funkcija aprašo paveikslą, atrodo taip:

$$P_{x,y \text{ įvertis}} = \mathbf{a} + \mathbf{d}_{x,y} \mathbf{s} \quad (30)$$

čia: $P_{x,y}$ įvertis – funkcija, aprašanti paveikslą, a - *angl. offset*, liet. foninis ryškumas, s - *angl. step*, liet. kontrasto pokytis), $d_{x,y}$ – daugiklis, priklausantis nuo koordinatų x ir y lokaliame paviršiuje, o x ir y visam paviršiui yra bendras, todėl daugiklis $d_{x,y}$ visada bus konstanta tai pačiai lokaliai matricai.

Įvertis (30) tuo geriau atitinka tiriamo paveikslo paviršių, kuo apskaičiuota (31) vertė mažesnė:

$$a + d_{x,y}s - grey(x, y) \quad (31)$$

čia $grey(x, y)$ – išmatuota pikselio vertė.

(31) formulę galima apibrėžti kaip vidutinę paklaidą, t.y. nuokrypį nuo tikrojo paveikslo. Jei visuose paveikslo taškuose vidutinė paklaida būtų 0 (paviršius būtų toks pat kaip ir originalus paveikslas), tai funkcija būtų tobula ir idealiai aproksimuotų tiriamą paveikslą. Tačiau deja, realiame gyvenime taip nebūna (į paveikslus įsiterpia įvairūs triukšmai, kurie didina paklaidas ir pan.), todėl toliau skaičiuojamas skirtumo kvadratas (tam, kad rezultatas būtų teigiamas skaičius):

$$\sum_{x=-2}^{+2} \sum_{y=-2}^{+2} (a + d_{x,y}s - grey(x, y))^2 \quad (32)$$

(32) funkcija skaičiuojama kiekvienam lokaliai matricos pikseliui. Šią, vidutinės kvadratinės paklaidos funkciją pavadinama minimalia deviacija. Skaičiuojant minimalią funkciją lokaliaje matricoje keisis tik $grey(x, y)$ vertė.

Norint surasti nežinomuosius dydžius a ir s , reikia skaičiuoti lygčių sistemą, kurią sudaro pirmosios minimalios deviacijos išvestinės a ir s atžvilgiu (33):

$$\left\{ \begin{array}{l} 25a + \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} d_{x,y} = \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} grey(x, y) \\ a + \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} d_{x,y} + s \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} d_{x,y}^2 = \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} grey(x, y) d_{x,y} \end{array} \right. \quad (33)$$

Vienai lokaliai matricai, t.y vienam „kernel‘iui“ kairiosios lygybės pusės visada bus konstantos, o dešinės bus skaičiuojamos kiekvienam paveiksliui atskirai, sudedant visų pikselių, toje lokaliaje matricoje, vertes.

Atskleidus ir pakėlus kvadratu (28) funkciją, gaunamas kriterijus (34), pagal kurį randami paveikslėlyje esantys kraštai:

$$\begin{aligned}
 & a^2 25 + s^2 \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} d_{x,y}^2 + 2as \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} d_{x,y} + \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} grey(x,y)^2 - 2a \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} grey(x,y) - \\
 & - 2s \sum_{x=-2}^{+2} \sum_{y=-2}^{+2} grey(x,y) d_{x,y}
 \end{aligned} \tag{34}$$

Mažiausių kvadratų metodas nėra sprendžiamas realiu laiku, nes jis šis uždavinys netiesinis, todėl jo sprendimas realiu laiku užtruktų labai ilgai, šiuo atveju iš mažiausių kvadratų metodo, gaunama įverčių skaičiavimo lygtis (34), pagal foninį ryškumą (a) ir kontrasto pokytį (s), ir pagal ją kiekvienam paveiklo pikseliui skaičiuojama 12 variantų (pagal aprašytus ieškomus kampus) ir pagal gautą kriterijaus vertę bei jos palyginimo su kaimyninių pikselių šiomis vertėmis yra atrenkamas labiausiai tinkamas kampo variantas. Šioje vietoje ir atsiranda metodo lėtumas, kadangi kiekvienam pikseliui yra skaičiuojama po 12 įverčių ir vėliau iš jų atrenkamas teisingas kampas. Haris ir FAST Features metodas to nedaro. Ir būtent iš šios vietos galima pasakyti, ne tik tai, kad atrenkamas taškas yra kampas, bet galima nustatyti jo orientaciją, taip pat pasakyti kampas ar kraštas yra aptiktas.

Šis kriterijus parodo, kaip gerai aproksimuojamas vietinis paveikslas. Naudojantis šia išraiška reikia suskaičiuoti kiekvieno paveiklo pikselio kriterijaus vertę ir pagal vertės dydį palikti tik lokalius minimumo taškus. Kiekvieno pikselio kriterijaus vertė lyginama su jo kaimyninių pikselių kriterijų vertėmis ir jei tikrinamo pikselio kriterijus yra mažesnis, vadinasi tas taškas buvo aproksimuotas geriau. Jei pikselio kriterijus yra mažiausias tarp jo kaimynų, tai tas pikselis yra vienareikšmiškai kampas. Taigi naudojantis minimalaus kvadrato metodu ir yra atrenkami kampai, bet sužinomos lokalių matricos a ir s koeficientai.

3.4 Kampų porų radimas

Atrinkus charakteringus taškus dviejuose nuotraukose pradedamas bendrų kampų porų radimas, t.y. išrenkami tie kampai, kurie abiejuose nuotraukose sutampa. Poros atrenkamos keliais žingsniais:

1. Tikrinamas kiekvienos rastos poros kampo indeksas abiejuose nuotraukose, indeksas priskiriamas kiekvienam kampui juos apsirašant, ir jei charakteringieji taškai, randami abiejuose

nuotraukose turi skirtingus indeksus, pora yra iškart atmetama (nes toje pačioje vietoje esantys kampai turi būti atpažinti, kaip tas pats kampas), jei indeksai sutampa, tada pora keliauja į antrą tikrinimo žingsnį.

2. Likusios poros yra atrenkamos pagal kriterijų, kurį aprašo (35) formulė:

$$kriterijus = (s_1 - s_2)^2 + (a_1 - a_2)^2 \quad (35)$$

3. Idealoje aplinkoje poros a ir s koeficientai turėtų sutapti ir poros kriterijus turėtų būti 0, tačiau realioje aplinkoje taip nebūna, todėl laikoma, kad pora yra pora, jei jos kriterijus yra kuo mažesnis, t.y. artėja į nulį.

4. Porų atrinkimui taip pat galima naudoti ir trimatę padėtį remiantis dviejų kamerų vaizdais. Šiame žingsnyje, žinant kameros koeficientą ks , atstumą tarp dviejų kamerų, kameros matymo kampą, nuotraukų rezoliuciją ir charakteringųjų taškų koordinatas pikseliais, galima apskaičiuoti mažiausią nuokrypį, o iš jo apskaičiuoti charakteringo taško atstumą iki kameros lęšio, taip pat kampo atstumą nuo centro (įstrižainę) bei x ir y pokytį metrais nuo centrinio taško nuotraukoje [31]. Atrinktos poros patikrinamos pagal atstumą iki kameros lęšio, jei to pačio taško atstumas iki kameros lęšio labai skiriasi (daugiau nei 0,05 – 0,1 m.) pora atmetama, jei atstumas iki lęšio sutampa arba skiriasi labai nedaug, tuomet tikrinama atstumo nuo taško iki kameros lęšio, priklausomai nuo patapos, kurioje atliekamas eksperimentas, pora laikoma gera, jei jos atstumas iki lęšio yra intervale nuo 0,2 – 4,0 m, jei į šį intervalą pora nepatenka, tuomet ji atmetama kaip netinkama.

5. Toliau vykdomas porų atrinkimas. Šiame žingsnyje tikrinamos arčiausiai esančias poros viena su kita, ir žiūrima kurios poros kriterijus yra mažiausias, porą su mažesniu kriterijumi paliekama, o su didesniu kriterijumi išmetama iš tolimesnio atrinkimo. Ir taip daroma iki tol, kol lieka tik tam tikras, pasirinktas skaičius porų.

6. Vietoje 4 žingsnio gali būti atliekamas ir penktas. Šiame žingsnyje galima daryti atrinktų charakteringųjų taškų porų rikiavimą. Tam labai tinka paprastas burbulo rikiavimo metodas (aprašytas skyriuje žemiau). Šiame žingsnyje atrinktos charakteringųjų taškų poros surikiuojamos pagal kriterijaus vertę nuo mažiausios reikšmės iki didžiausios. Tokiu būdu vėlgi pasiliekame tik tam tikrą, pasirinktą skaičių, porų pagal mažiausią kriterijų.

Kartu atlikti trečio ir ketvirto žingsnio nereikia, reikia pasirinkti kažkurį vieną iš jų. Kartu šia du žingsniai atliekami, kai tikrinamos nuotraukos pilnoje RGB spalvų paletėje. Tada trečiu žingsniu atliekamas porų išrinkimas iš kiekvienos spalvos sluoksnio, o ketvirtu žingsniu išrikiuojamos ir atrenkamos geriausios poros iš atskirų sluoksnių.

3.5 Burbulo metodas

Sprendžiant rūšiavimo uždavinį šiuo metodu, iš eilės lyginami du gretimus aibės A elementus (a_1 su a_2), (a_2 su a_3), ... , (a_N su a_{N-1}). Jei $a_i > a_{i+1}$, tai tokie elementai sukeičiami vietomis. Iš sekos matyti, kad po pirmo visų elementų patikrinimo didžiausias sekos elementas užims paskutinio elemento vietą: rikiuojama nuo mažiausią vertę turinčio elemento iki didžiausio. Todėl antrojo rikiavimo ciklo metu užtenka tikrinti duomenų seką su mažiau elementų, iki $(N - 1)$ -tojo elemento. Procesas kartojamas tol, kol vieno iš ciklų metu elementai nėra sukeičiami vietomis nei karto. Kadangi po kiekvieno rikiavimo ciklo, seka sumažėja vienu elementu, tai blogiausiu atveju rikiavimo algoritmas bus kartojamas $(N - 1)$ kartą.

Šis rūšiavimo algoritmas burbulo metodu (*angl. bubble sort*) vadinamas todėl, kad didžiausio elemento judėjimas primena burbulo kilimą į vandens paviršių. Ir šis rūšiavimo algoritmas yra stabilus [43].

3.6 Paklaidos

Beveik visuose eksperimentuose kažkas matuojama, tačiau bet kuris matavimas duoda ne tikslią matuojamo dydžio reikšmę, o tik apytikslę. Taip yra dėl mūsų turimų įrankių ir prietaisų riboto tikslumo, matavimo metu daromų įvairaus pobūdžio paklaidų.

Matavimo paklaida - tai matuojamojo dydžio matavimo rezultato ir tikrosios matuojamojo dydžio vertės skirtumas:

$$\Delta_t = |x - A_n| \quad (36)$$

čia x - matavimo rezultatas, o A_n - tikroji matuojamojo dydžio vertė.

Taip apibrėžta paklaida tikroji matavimų paklaida, tačiau ji praktiškai negali būti gaunama, nes tikroji matuojamojo dydžio vertė yra nežinoma. Praktikoje tenka tenkintis šios paklaidos įverčiu Δ , vietoje A_n įrašius sutartinę matuojamojo dydžio vertę A :

$$\Delta = |x - A| \quad (37)$$

Santykinė paklaida - absoliutinės paklaidos ir matavimo rezultato santykis, padaugintas iš 100, tokiu būdu ši paklaida išreiškiama procentais [44]:

$$\delta = \frac{|x - A|}{|x|} \cdot 100\% \quad (38)$$

Atlikus eksperimentus ir gavus tam tikras rezultatų skaitines reikšmes reikia jas matematiškai statistiškai apdoroti. Čia labai padeda tikimybių teorijos ir matematinės statistikos pagrindai. Turint eksperimento rezultatus, statistikoje vadinamą matematine imtimi, galima apskaičiuoti svarbiausius statistinius rodiklius:

1. Aritmetinį vidurkį :

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} \quad (39)$$

2. Dispersiją s^2 :

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1} \quad (40)$$

3. Vidutinį kvadratinį nuokrypį s :

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (41)$$

4. Atsitiktinė matavimo paklaidą:

$$\Delta = t_\beta \cdot \frac{s}{\sqrt{n}} \quad (42)$$

t_β – Stjudento kriterijus, priklausantis nuo užsiduotos pasiklovimo tikimybės α , moksle ir technikoje dažniausiai $\alpha = 0,95$ ir nuo laisvė laipsnių skaičiaus $\varphi = n - 1$, šis kriterijus randamas literatūroje, šiuo atveju $t_\beta = 1,936$.

5. Pasikliautinį intervalą:

$$I_\beta = (\bar{x} - \Delta; \bar{x} + \Delta) \quad (43)$$

Šis intervalas apibūdina rezultato (aritmetinio vidurkio) tikslumą. Kuo intervalas mažesnis, tuo geriau [44].

5. Eksperimentiniai tyrimai

Kaip jau minėta anksčiau eksperimentinės sistemos veikimui reikalinga tiek techninė (*angl. hardware*), tiek programinė įranga (*angl. software*). Techninę įrangą sudaro žingsninis variklis, ant jo primontuojama X formos iškyša ant kurios galų yra po internetinę kamerą. Norint pakeisti kamerų poziciją, galima jas sukti aplink žingsninio variklio ašį.

Dar kartelį primenami pagrindiniai techninės įrangos parametrai:

- Kameros matymo kampas: 60 °;
- Kampas tarp kamerų: 7,5 °;

- Žingsninio variklio vienas žingsnis: 7,5 °;
- Iškyšos ilgis: 0,2 m;
- Kameros rezoliucija: 640 x 480.

Taigi nufotografavus du vaizdus, abi nuotraukos tikrinamos to pačio algoritmo, ieškančio charakteringųjų taškų ir vėliau iš to sąrašo ieškomos bendros kampų poros. Norint, kad eksperimentas pavyktų reikia ne tik techninės, bet ir programinės įrangos, kurios kūrimas ir testavimas aptariamasis žemiau.

Eksperimentas buvo atliktas keliose elektros ir elektronikos fakulteto patalpose: robotikos laboratorijoje, bibliotekoje, FESTO laboratorijoje, koridoriuose, dėstytojo kabinete taip pat mano kambaryje.

Kuriant programinę įrangą nuolatos buvo atliekami tyrimai ir stebima kokie rezultatai prie kokių pirminių verčių yra gaunami. Pirmiausia, charakteringųjų taškų atpažinimo filtras buvo kuriamas ir tikrinamas su vienu vaizdu (23 pav.) ir tas vaizdas programiškai buvo konvertuotas į nespalvotą paveikslą (*angl. greyscale*). Kampai nuotraukoje apibrėžti raudonais apskritimais.



23 pav. Pirmieji rezultatai

Įsitikinęs, kad filtras veikia, jis buvo testuojamas su dviem paveikslais ir tikrinama ar atrandami tie patys charakteringi taškai (24 pav.)



24 pav. Dviejų nuotraukų pilkame fone eksperimentiniai tikrinimo rezultatai

Daugelio bandymų metu pastebėta, kad charakteringieji taškai atrandami geriau kai slenkstinės minimalios deviacijos, t.y. nuokrypio ir kontrasto pokyčio (s - step) vertės yra mažesnės, sumažinę minimalios deviacijos slenkstinę vertę išvengiame triukšmų ir kitų nuotraukoje esančių „šiukšlių“, o sumažinę kontrasto pokyčio slenkstinę vertę ieškomi ir pažymimi tik ryškūs kampai. Toliau tyrime naudojamos tokios pradinės sąlygos:

Pikselio tikrinimo lango dydis: 7x7;

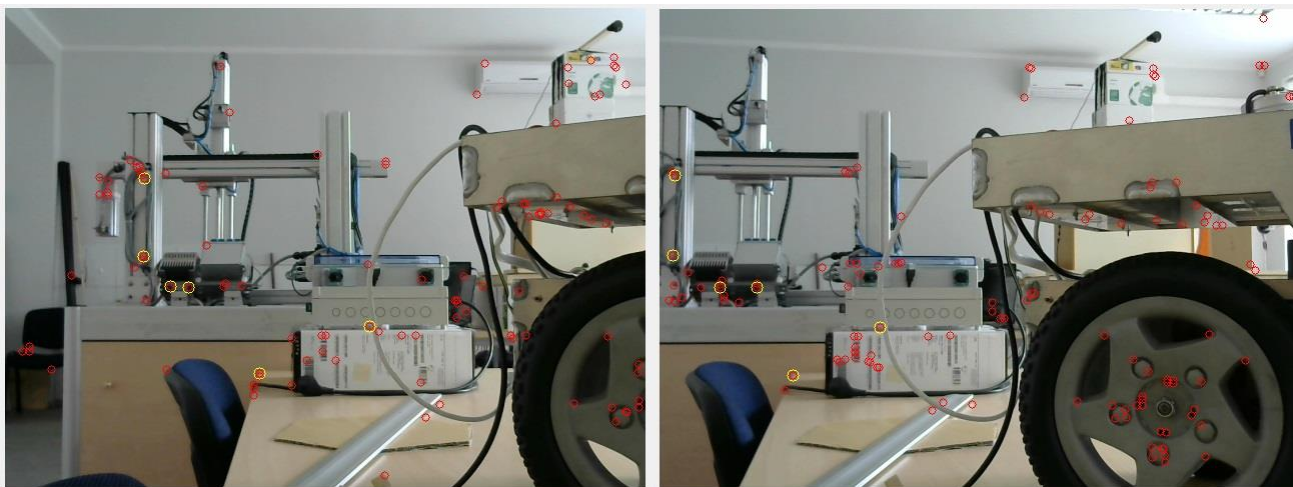
Minimalios deviacijos slenkstinė vertė: 20

Kontrasto pokyčio (s - step) slenkstinė vertė: 10

Įsitikinus, kad filtras veikia ir atpažįsta nuotraukoje esančius charakteringuosius taškus, internetinė kamera montuojama ant žingsninio variklio (21 pav.) ir toliau atliekami eksperimentiniai tyrimai, ieškant sutampančių charakteringųjų taškų porų.

Kiekvienoje nuotraukų poroje nuotrauka viena nuo kitos pasisukusi $7,5^\circ$ kampu, t.y. toks yra vieno žingsninio variklio žingsnio dydis.

Nuotraukų tikrinimas visais atvejais toks pat: nufotografuojamos dvi nuotraukos (25 pav.) vėliau atrandamos visi charakteringieji taškai, atitinkantys sąlygas (pažymėta raudonais apskritimais), vėliau iš visų galimų variantų atrenkamos 6 geriausiai nustatytas sąlygas ir viena kitą atitinkančio poros (pažymėta geltonais apskritimais). Vėliau belieka suskaičiuoti kiek porų iš tiesų atitinka teisingai.



25 pav. Charakteringųjų taškų porų radimas

Iš 25 pav. matyti, kad šioje nuotraukoje teisingai atrasta ir pažymėta 5 poros iš 6. Toliau pateikiami rezultatai, gauti ištyrus 150 įvairių nuotraukų porų:

- nespaltotame fone;
- spalvotame fone, tiriant tik R spalvos sluoksnį;
- spalvotame fone, tiriant tik G spalvos sluoksnį;
- spalvotame fone, tiriant tik B spalvos sluoksnį;
- spalvotame fone, tiriant visus spalvų sluoksnius.

5.1 Rezultatai gauti tiriant nuotraukų poras (nespaltotas fonas)

Ištyrus 150 nuotraukų porų ir atlikus duomenų analizę, gauta nuotraukose mažiausiai aptikta 3 charakteringųjų taškų poros, daugiausia – 6, o teisingai aptiktų porų tikslumo vidurkis yra: 5,02 (2 lentelė)

2 lentelė. Pilkame fone teisingai atrastų porų skaičius

Minimaliai atrastas porų skaičius	3
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(4,885; 5,175)
Atrastų porų kiekio vidurkis	5,03

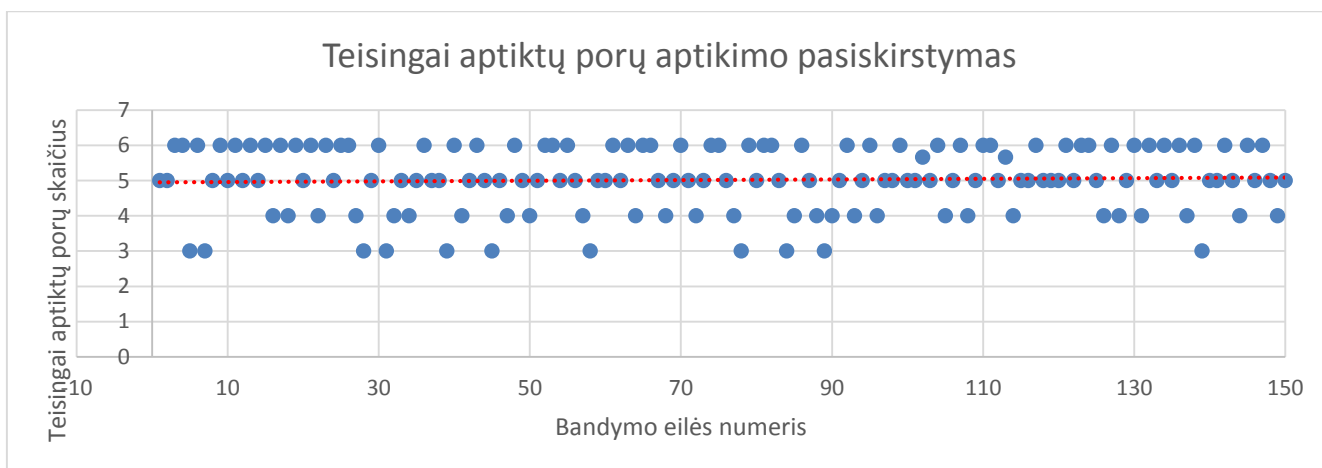
3 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 5 poros, o mažiausiai 3. Ši informacija taip pat pavaizduota ir 26 pav., 27 pav. pavaizduotas teisingai aptiktų porų pasiskirstymas.

3 lentelė. Atrastų porų dažnumas pilkame fone

Atrastų porų skaičius	Kiek kartų atrado
3	10
4	32
5	56
6	52



26 pav. Atrastų porų dažnumo pilkame fone histograma



27 pav. Teisingai aptiktų porų pilkame fone pasiskirstymas

Iš šios informacijos matyti, kad pilkame fone filtro veikimo vidurkis yra 83,67%, o atrastų porų kiekio vidurkio pasikliautinis intervalas (4,885; 5,175). Dokumentas su visa tikslia informacija yra pateikta diske.

5.2 Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, B spalvos sluoksnis)

Gavus priimtina tikslumą, buvo nutarta patikrinti charakteringųjų taškų paieškos filtro algoritmą spalvotam vaizdai, taigi antrasis bandymų etapas vyko su tomis pačiomis nuotraukomis, tačiau nesiverčiant nuotraukų į pilką foną, o tikrinant spalvoto vaizdo B (mėlynos) spalvos sluoksnį

Gauti rezultatai truputį kitokie, nei tyrus pilkas nuotraukas. Nuotraukose mažiausiai aptikta 1 charakteringųjų taškų pora, daugiausia – 6, o teisingai aptiktų porų tikslumo vidurkis yra: 4,50 (4 lentelė)

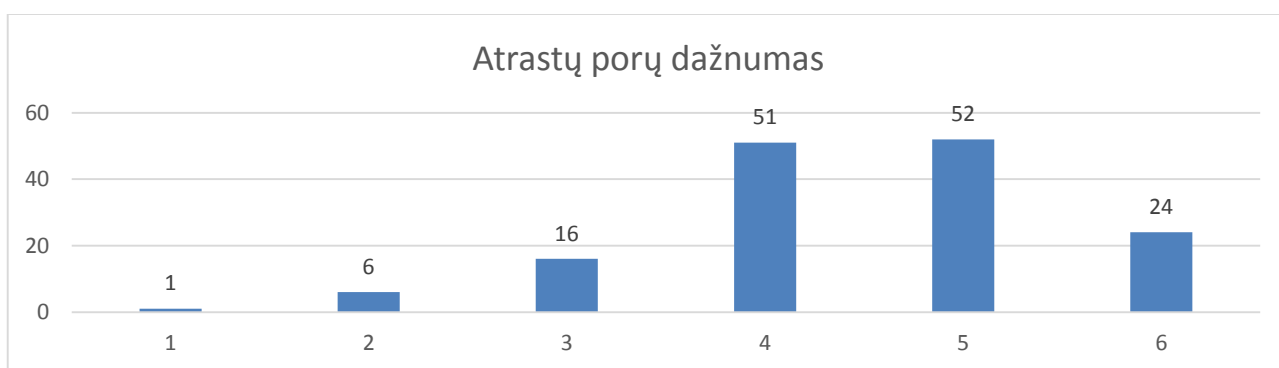
4 lentelė. Spalvotame fone (mėlynos spalvos sluoksnyje) teisingai atrastų porų skaičius

Minimaliai atrastas porų skaičius	1
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(4,333; 4,667)
Atrastų porų kiekio vidurkis	4,50

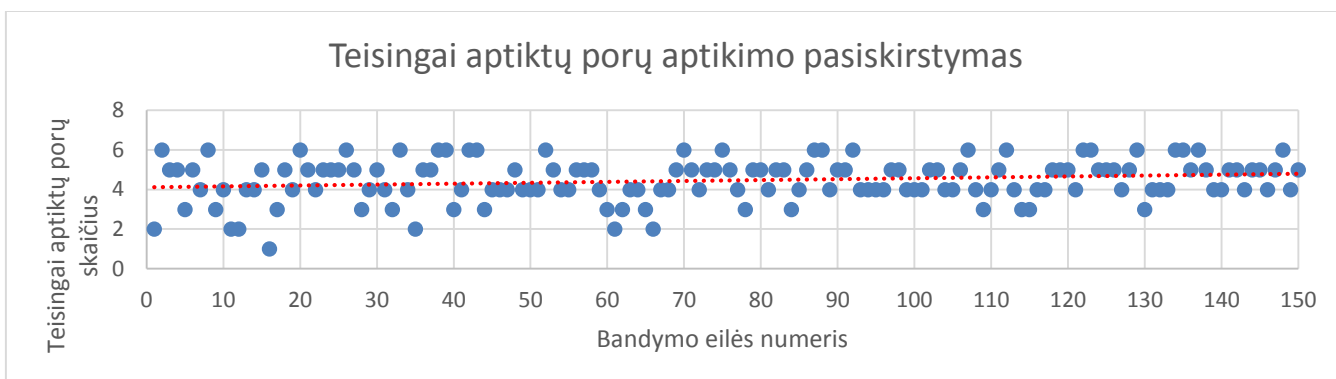
5 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 4 ir 5 poros, o mažiausiai 1 pora. Ši informacija taip pat pavaizduota ir 28 pav., 29 pav. pavaizduotas teisingai aptiktų porų pasiskirstymas.

5 lentelė. Atrastų porų dažnumas spalvotame fone (mėlynos spalvos sluoksnyje)

Atrastų porų skaičius	Kiek kartų atrado
1	1
2	6
3	16
4	51
5	52
6	24



28 pav. Atrastų porų dažnumo spalvotame fone (mėlynos spalvos sluoksnyje) histograma



29 pav. Teisingai aptiktų porų spalvotame fone (mėlynos spalvos sluoksnyje) pasiskirstymas

Iš šios informacijos matyti, kad spalvotame fone (mėlynos spalvos sluoksnyje) filtro veikimo vidurkis yra 75%, o atrastų porų kiekio vidurkio pasikliautinis intervalas (4,333; 4,667). Dokumentas su visa tikslią informacija yra pateikta diske.

4.3 Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, G spalvos sluoksniu)

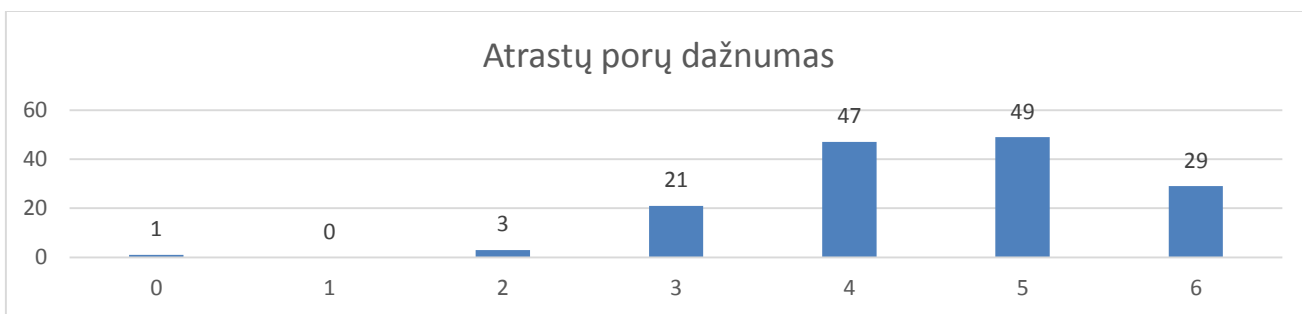
Toliau buvo tiriamas spalvoto vaizdo G (žalios) spalvos sluoksniu. Tiriant nuotraukos, pasitaikė atvejais, kai algoritmas neatpažino nei vienos bendros poros nuotraukose, tačiau daugiausia teisingai atpažintų porų buvo vėlgi 6, o teisingai aptiktų porų tikslumo vidurkis yra: 4,51 (6 lentelė) 6 lentelė. Spalvotame fone (žalios spalvos sluoksnyje) teisingai atrastų porų skaičius

Minimaliai atrastas porų skaičius	0
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(4,338; 4,682)
Atrastų porų kiekio vidurkis	4,51

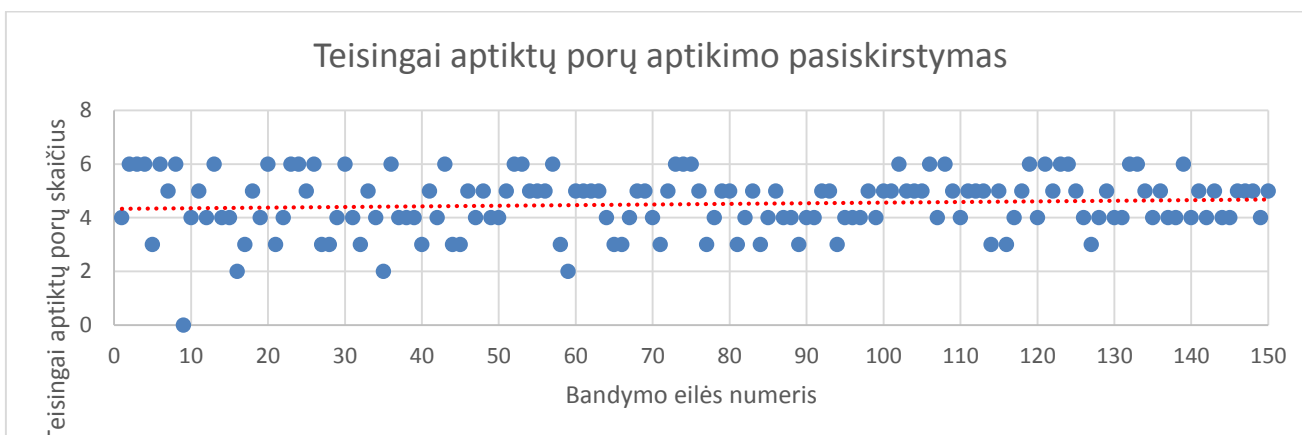
7 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 5 poros, o mažiausiai 0 porų. Ši informacija taip pat pavaizduota ir 30 pav., 31 pav. pavaizduotas teisingai aptiktų porų pasiskirstymas.

7 lentelė. Atrastų porų dažnumas spalvotame fone (žalios spalvos sluoksnyje)

Atrastų porų skaičius	Kiek kartų atrado
0	1
1	0
2	3
3	21
4	47
5	49
6	29



30 pav. Atrastų porų dažnumo spalvotame fone (žalios spalvos sluoksnyje) histograma



31 pav. Teisingai aptiktų porų spalvotame fone (žalios spalvos sluoksnyje) pasiskirstymas

Iš šios informacijos matyti, kad spalvotame fone (žalios spalvos sluoksnyje) filtro veikimo vidurkis yra 75,17%, o atrastų porų kiekio vidurkio pasikliautinis intervalas (4,338; 4,682). Dokumentas su visa tikslią informacija yra pateikta diske.

4.4 Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, R spalvos sluoksniu)

Galiausiai buvo tiriamas spalvoto vaizdo R (raudonos) spalvos sluoksniu. Tiriant nuotraukas, mažiausiai atpažintų charakteringųjų taškų porų buvo 1, o daugiausiai teisingai atpažintų porų buvo, kaip įprastai, 6, o teisingai aptiktų porų tikslumo vidurkis yra: 4,45 (8 lentelė)

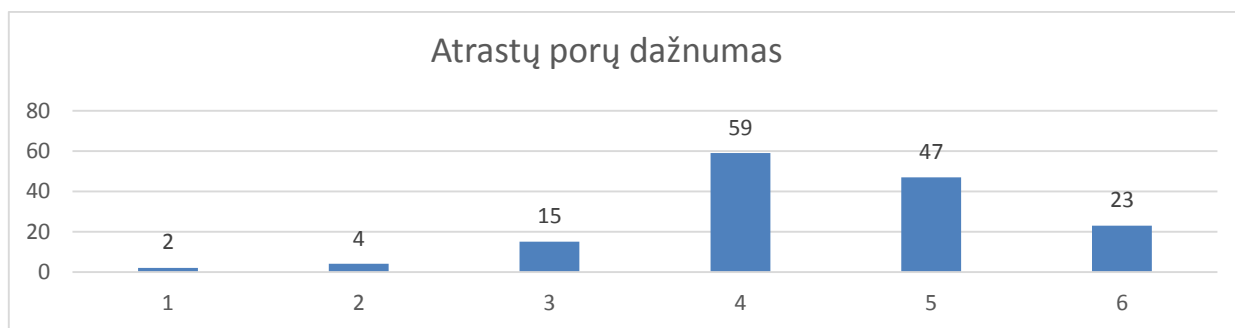
8 lentelė. Spalvotame fone (raudonos spalvos sluoksnyje) teisingai atrastų porų skaičius

Minimaliai atrastas porų skaičius	1
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(4,285; 4,614)
Atrastų porų kiekio vidurkis	4,45

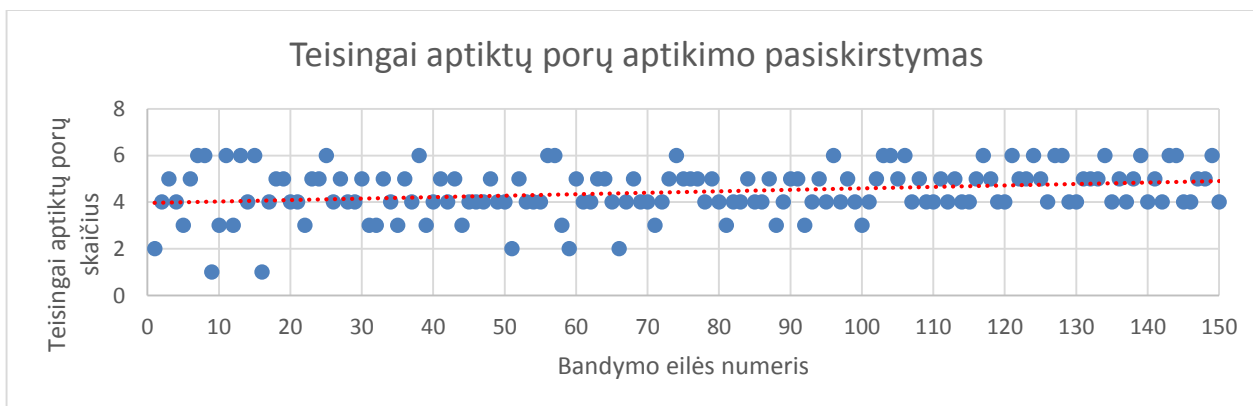
9 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 4 poros, o mažiausiai 1 pora. Ši informacija taip pat pavaizduota ir 32 pav., 33 pav. pavaizduotas teisingai aptiktų porų pasiskirstymas.

9 lentelė. Atrastų porų dažnumas spalvotame fone (raudonos spalvos sluoksnyje)

Atrastų porų skaičius	Kiek kartų atrado
1	2
2	4
3	15
4	59
5	47
6	23



32 pav. Atrastų porų dažnumo spalvotame fone (raudonos spalvos sluoksnyje) histograma



33 pav. Teisingai aptiktų porų spalvotame fone (raudonos spalvos sluoksnyje) pasiskirstymas

Iš šios informacijos matyti, kad spalvotame fone (raudonos spalvos sluoksnyje) filtro veikimo vidurkis yra 74,17%, o atrastų porų kiekio vidurkio pasikliautinis intervalas (4,285; 4,614). Dokumentas su visa tikslia informacija yra pateikta diske.

Ištyrus nuotraukas spalvotame fone, tik skirtinguose sluoksniuose, pastebėta, kad teisingai aptinkamų kampų porų tikslumas yra ~75%, o atrastų porų kiekio vidurkio pasikliautinis intervalas svyruoja tokiam intervalui (4,285; 4,682) ir tai yra natūralu, kadangi vienas sluoksnis neturi vienokio atspalvio, vietoje, kur yra charakteringasis taškas, kitas sluoksnis, kitoje vietoje, norint padidinti porų ir kampų aptikimo tikslumą atpažinimo algoritmas buvo perrašytas taip, kad charakteringųjų taškų aptikimas ir porų radimas būtų atliekamas spalvotame fone, neišskiriant nei vieno sluoksnio atskirai.

4.5 Rezultatai gauti tiriant nuotraukų poras (spalvotas fonas, visi spalvų sluoksniai)

Tiriant nuotraukos spalvotame fone, pastebėta, kad reikia pakeisti pradines sąlygas tam, kad atrandamų porų tikslumas būtų didesnis. Pradinės sąlygos, tiriant spalvotą vaizdą yra tokios:

- Pikselio tikrinimo lango dydis: 7x7;
- Minimalios deviacijos slenkstinė vertė: 20
- Kontrasto pokyčio (s – step) slenkstinė vertė: 8

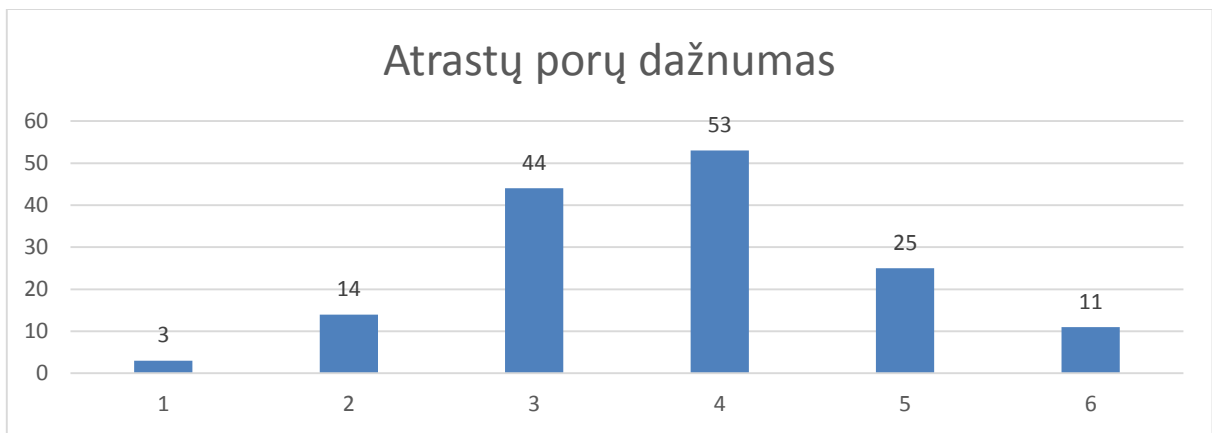
Atlikus eksperimentus, gauti tokie rezultatai: mažiausiai teisingų porų aptikimo nuotraukose skaičius buvo 1 pora, daugiausia teisingai atpažintų porų buvo 6, o teisingai aptiktų porų tikslumo vidurkis yra mažesnis, nei kad pilkame fone ar atskiruose spalvų sluoksniuose: 3,77 (10 lentelė) 10 lentelė. Spalvotame fone teisingai atrastų porų skaičius

Minimaliai atrastas porų skaičius	1
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(3,592; 3,948)
Atrastų porų kiekio vidurkis	3,77

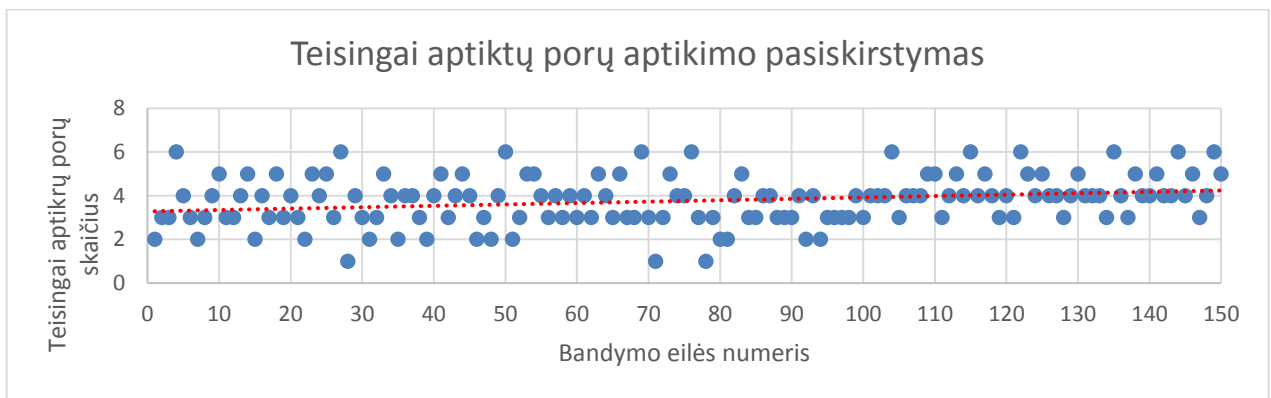
11 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 4 poros, o mažiausiai 1 porų. Ši informacija taip pat pavaizduota ir 34 pav. ir 35 pav.

11 lentelė. Atrastų porų dažnumas spalvotame fone

Atrastų porų skaičius	Kiek kartų atrado
1	3
2	14
3	44
4	53
5	25
6	11



34 pav. Atrastų porų dažnumo spalvotame fone histograma



35 pav. Teisingai aptiktų porų spalvotame fone pasiskirstymas

Iš šios informacijos matyti, kad spalvotame fone filtro veikimo vidurkis yra 62,83%, o atrastų porų kiekio vidurkio pasikliautinis intervalas (3,592; 3,948). Dokumentas su visa tikslia informacija yra pateikta diske.

Prieš tiriant nuotraukas ir ieškant geriausiai atitinkančių kampų porų spalvotame vaizde (su visais sluoksniais) buvo tikėtasi gauti daug didesnę aptikimo tikslumą, t.y. tikėtasi gauti bent 80% tikslumą, tačiau jis galėjo būti negautas dėl didelės atskirų spalvų sluoksnių koreliacijos, kaip jau minėta anksčiau. Tačiau, tyrimo buvo iškelta hipotezė, kad į geriausių porų atradimo algoritmą įvedus papildomą paieškos žingsnį pagal atstumo erdvėje įverčio radimą remiantis dviejų kamerų vaizdais tikslumas turėtų pagerėti. Eksperimentas buvo atliktas naudojantis doc. Dr. Renaldo Urniežiaus matematiniu modeliu ir programiniu kodu.

Rezultatai: mažiausiai teisingų porų aptikimo nuotraukose skaičius buvo 0 porų, daugiausia teisingai atpažintų porų buvo 6, o teisingai aptiktų porų tikslumo vidurkis yra mažesnis, nei kad spalvotame fone: 3,43 (12 lentelė)

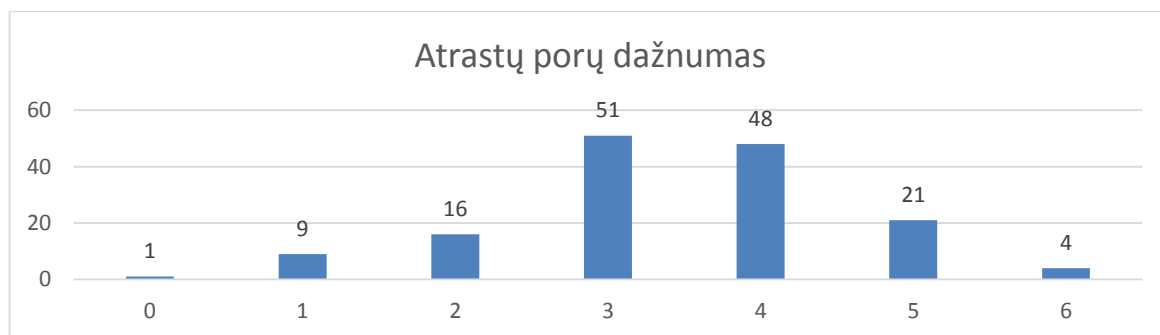
12 lentelė. Spalvotame fone teisingai atrastų porų skaičius, įvertinant atstumą

Minimaliai atrastas porų skaičius	1
Maksimaliai atrastas porų skaičius	6
Atrastų porų kiekio vidurkio pasikliautinis intervalas	(3,246; 3,614)
Atrastų porų kiekio vidurkis	3,77

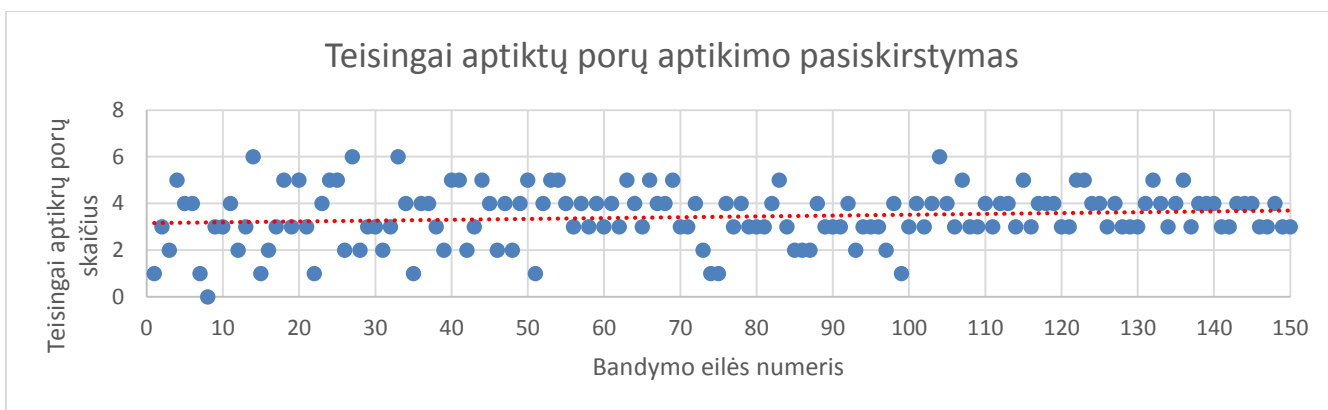
13 lentelėje pateikta, kaip dažnai buvo atpažįstamas tam tikras skaičius charakteringųjų taškų porų. Pastebėta, kad daugiausia kartų buvo aptikta 3 poros, o mažiausiai 0 porų. Ši informacija taip pat pavaizduota ir 36 pav. ir 37 pav.

13 lentelė. Atrastų porų dažnumas spalvotame fone, įvertinant atstumą

Atrastų porų skaičius	Kiek kartų atrado
0	1
1	9
2	16
3	51
4	48
5	21
6	4



36 pav. Atrastų porų dažnumas spalvotame fone, įvertinant atstumą



37 pav. Teisingai aptiktų porų spalvotame (įvertinant atstumą) fone pasiskirstymas

Iš šios informacijos matyti, kad spalvotame fone, naudojant algoritmą su filtro veikimo vidurkis yra 57,17%. o atrastų porų kiekio vidurkio pasikliautinis intervalas (3,246; 3,614). Dokumentas su visa tikslia informacija yra pateikta diske.

Norint išsiaiškinti, kodėl šis metodas nepadėjo tikslumui pagerinti, buvo atliktas dar vienas eksperimentas, naudojantis tuo pačiu atstumo erdvėje įverčio radimo remiantis dviejų kamerų vaizdais programiniu kodu, kur buvo tiriama išmatuoto atstumo skaičiavimo paklaida.

4.6 Išmatuotų ir gautų atstumo įverčių radimo tyrimas

Tyrimas atliekamas nusifotografavus dvi nuotraukas. Gautose nuotraukose patikrinamas apskaičiuotas to paties taško atstumas iki kameros lęšio (38 pav.). Žinant tikrą atstumą iki daikto, kokio atstumą tikriname nuotraukose galima apskaičiuoti gaunamas skaičiavimo paklaidas (12 lentelė).

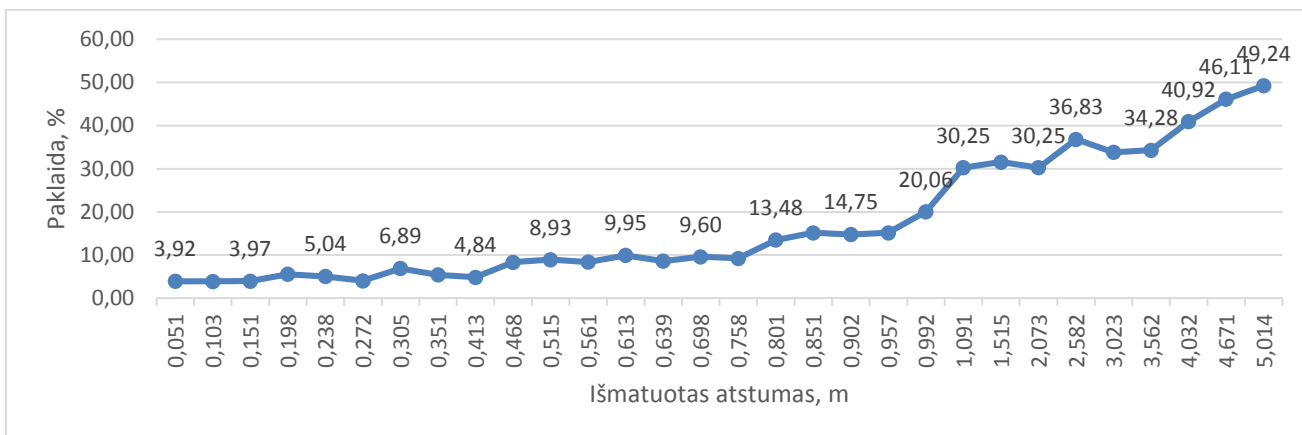


38 pav. Atstumo, remiantis dviejų kamerų vaizdais, nustatymo eksperimento langas

38 pav. Atstumas tikrinamas tame taške, kuris apibrauktas žaliu apskritimu, lange taip pat matoma z1 ir z2 reikšmės – programiškai apskaičiuotas atstumas nuo kameros lęšio iki pasirinkto taško. 14 lentelėje parodytos vidutinės atliktų bandymų išmatuotų ir apskaičiuotų atstumų vertės, bei gautos paklaidos. Detali lentelė su rezultatais pateikta diske.

14 lentelė. Išmatuota ir programiškai apskaičiuota trimatė padėtis remiantis dviejų kamerų vaizdais.

Atstumas	Vidutinė paklaida, m	Paklaida, %	Dispersija	Pasikliautinimo intervalas
> 0.5 m	0,015	5,19	0,0721	(-0,175; 0,205)
0.5 m - 0.8 m	0,058	9,12	0,2708	(-0,345; 0,461)
0.8 m - 1 m	0,143	15,72	0,3551	(-0,342; 0,628)
< 1 m	0,478	30,68	0,0967	(0,165; 0,791)
< 2.5 m	1,065	34,97	0,1149	(0,734; 1,396)
< 4 m.	2,091	45,43	0,1451	(1,708; 2,474)



39 pav. Trimatės padėties nustatymo remiantis dviejų kamerų vaizdais paklaidos priklausomybė nuo atstumo

Iš 14 lentelės matyti, kad kamerasi esant nedideliu atstumu nuo tiriamo daikto, apskaičiuojami atstumai, yra priimtino tikslumo, jų paklaida nesiekia 5%, atstumui didėjant, didėja ir paklaida, tačiau nuo 0,5 m iki 0,8 m, ji taip pat nėra labai didelė, nesiekia 10%, nuo 0,8 m iki 1 m paklaida dar neviršija 20%, peržengus 1 m ribą, paklaida didėja žymiai sparčiau ir pasiekia 30%. Peržengus 2,5 m ribą paklaida didėja dar labiau, prie 5 m atstumo paklaida yra apie 50%. Iš 14 lentelės ir 39 pav. matyti, kad geriausiai atstumas randamas iki 1m.

Tai turėtų reikšti, kad trimatės padėties nustatymas remiantis dviejų kamerų vaizdais turėtų padėti tiksliau rasti charakteringųjų taškų poras, tačiau atliekant šį eksperimentą, buvo pastebėta, kad

naudojamos Logitech C270 internetinės kameros turi automatinį fokusavimą ir daiktų esančių labai arti nemato, nes jie tampa išsilieję. Kuo toliau nuo kameros yra vaizdas, tuo jis ryškiau matomas, tačiau, tuo didesnė paklaida gaunama skaičiuojant atstumą.

4.7 Sukurto charakteringųjų taškų filtro palyginimas su Harris – Stephens ir FAST Features kampų atpažinimo metodais

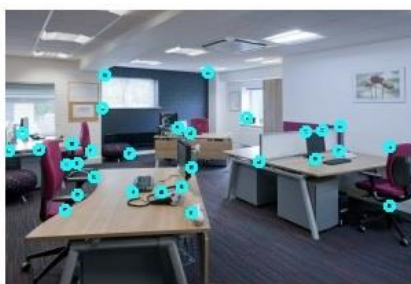
Atlikus visus numatytus eksperimentus, tikrinančius filtro veikimą, nuspręsta palyginti tiriamąjį charakteringųjų taškų filtrą su jau esamais Harris-Stephens ir FAST Features kampų atpažinimo filtrais. Šie du filtrai lyginimui buvo pasirinkti neatsitiktinai: Harris-Stephens filtras yra originalus, vienas pirmųjų charakteringųjų taškų atpažinimo filtrų, būtent šis filtras davė pradmenis šiai mokslo sričiai, o FAST Features – labai naujoviškas, įvairiuose uždavinių sprendimuose populiarėjantis filtras bei šis metodas yra vienas iš paskutinių naujai sukurtų charakteringųjų taškų atpažinimo filtrų.

Filtrai palyginami dviem aspektais: greitimeikos klausimu ir kampų atpažinimo paveiksle kiekiu, t.y. atpažinimo tikslumu. Palyginimas vyksta lyginant atskirus filtro žingsnius: tikrinant kampų atpažinimo greitį ir tikslumą vienoje nuotraukoje, dviejuose nuotraukose ir atliekant visą filtro veikimo algoritmą: t.y. nuo nuotraukų nufotografavimo iki 6 tų pačių kampų porų radimo dviejuose kadruose. Metodų tikrinimui naudojamos tos pačios nuotraukos ir po kiekvieno metodo suveikimo patikrinama, kaip tiksliai buvo aptikti charakteringieji taškai tose pačiose nuotraukose, bet naudojant skirtingus metodus.

Pirmiausia buvo lyginami kampai, atrandami vienoje nuotraukoje. Tikrinimo langas pavaizduotas 40 pav.



Rasta kampų: 30; Laikas kampams rasti: 2659



Harris:
Laikas: 102.7422; Rasta kampų: 30



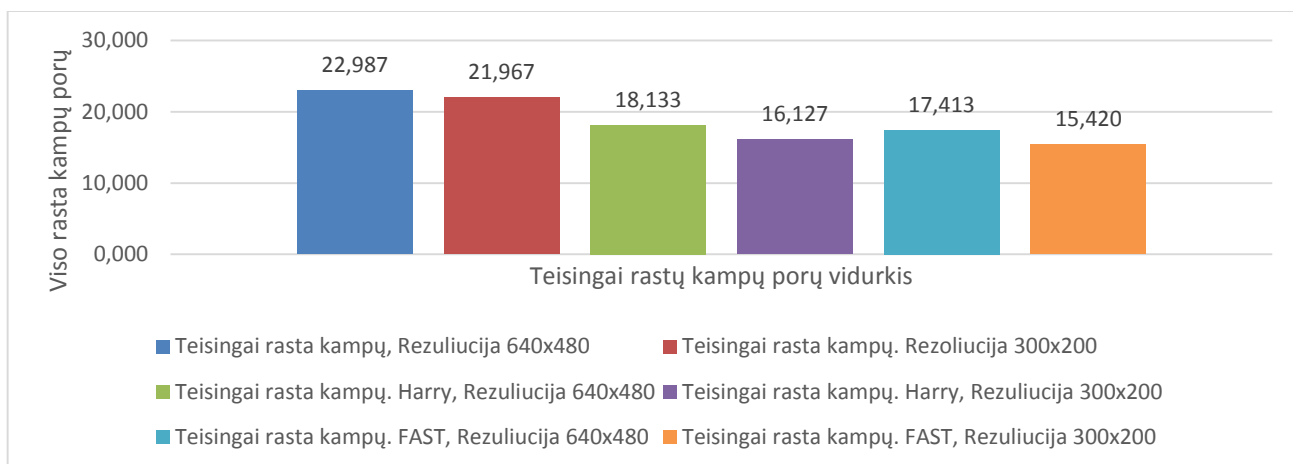
FAST:
Laikas: 89.9274; Rasta kampų: 30

40 pav. Vienos nuotraukos kampų radimo greičio ir tikslumo palyginimo langas

Atlikus 150 bandymų nustatyta, kad greičiausiai vienoje kampus atranda FAST FEATURES filtras, bet jo tikslumas mažiausias, vidutinis kampų atradimo tikslumas yra 17,41 kampo iš 30 rastų. Lėčiausiai kampus atranda tiriamasis filtras, bet jo tikslumas yra didžiausias: 22,99 taškai iš 30 surastų. Norint padidinti greitaveiką buvo pakeista nuotraukų rezoliucija iš 640x480 į 300x200. Gauti greitaveikos ir tikslumo rezultatai nepasikeitė: greičiausias FAST Features, tiksliausias – tiriamasis filtras. Tačiau sumažinus rezoliuciją tiriamojo filtro greitis sumažėjo 7,2 karto, o tikslumas sumažėjo vidutiniškai tik vienu kampu. FAST Features filtro greitaveika sumažėjo tik 1,16 karto, o tikslumas sumažėjo 1,13 karto (2 kampais). Vidurkiniai duomenys pateikti 15 lentelėje ir 41 pav. O visi tikslūs duomenys pateikti diske.

15 lentelė. Greitaveikos ir tikslumo palyginimas tiriant vieną nuotrauką.

Bandymo nr,	Iš viso rasta kampų	Rezultatai, tiriant vieną nuotrauką					
		Tiriamasis filtras					
		Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų	Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų		
Vidurkis:		20060,560	22,987	2786,168	21,967		
Rezultatai, tiriant vieną nuotrauką							
Harris filtras				FAST Features filtras			
Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų	Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų	Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų	Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų
110,167	18,133	61,891	16,127	70,764	17,413	61,004	15,420



41 pav. Teisingai rastų kampų tikslumo palyginimas tarp skirtingų metodų ir skirtingų rezoliucijų

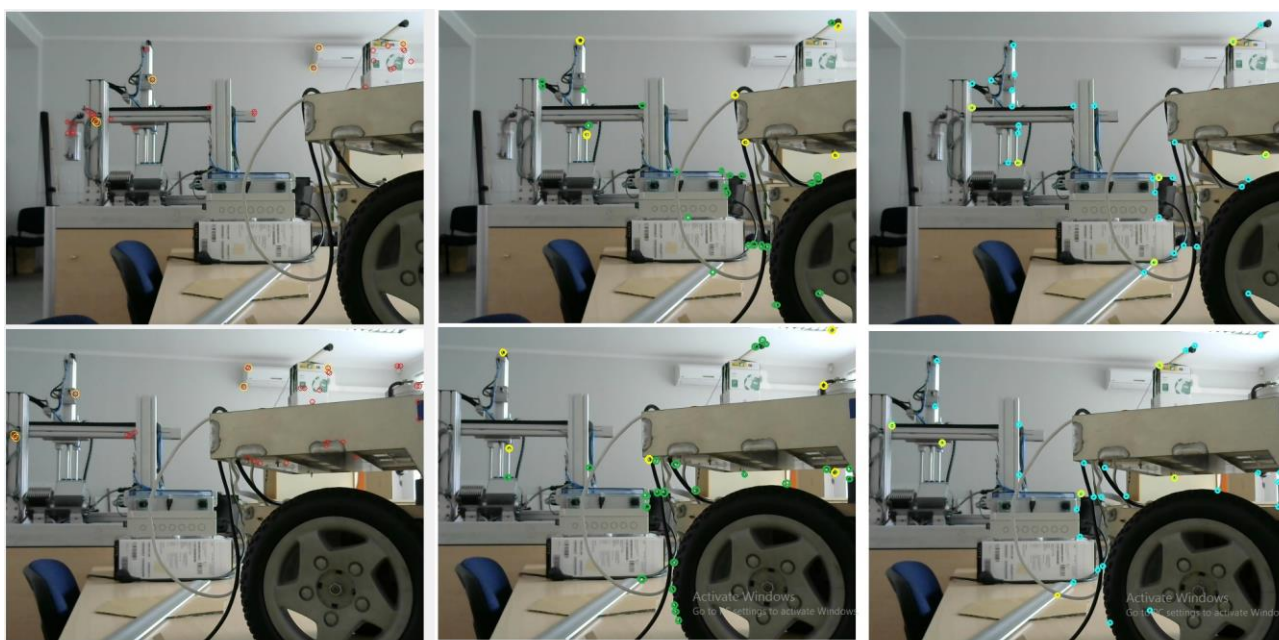
Taigi iš palyginimo matyti, kad sumažinus paveikslo rezoliuciją, tiriamojo filtro greیتaveika labai sumažėja, o tikslumas nukenčia nelabai stipriai.

Toliau buvo tiriama kaip greitai atpažįstami kampai tiriant dvi nuotraukas. Rezultatai gauti tiriant dvi nuotraukas, taip pat yra tokie patys. Greitaveikos vidurkiai, gauti ištyrus 150 nuotraukų porų pavaizduoti 16 lentelėje, o visa detali lentelė pateikta diske.

16 lentelė. Greitaveikos vidurkiai tiriant nuotraukų poras aukštesne ir žemesne rezoliucijomis.

Bandymo nr,	Iš viso rasta kampų	Rezultatai, tiriant nuotraukų porą					
		Tiriamasis filtras		Harris filtras		FAST Features filtras	
		Greitaveika, ms. Rezoliucija 640x480	Greitaveika, ms. Rezoliucija 300x200	Greitaveika, ms. Rezoliucija 640x480	Greitaveika, ms. Rezoliucija 300x200	Greitaveika, ms. Rezoliucija 640x480	Greitaveika, ms. Rezoliucija 300x200
Vidurkis:		32005,77	4445,25	220,33	123,78	141,53	122,01

Galiausiai buvo palyginta visa algoritmo įvykdymo greیتaveika ir 6 porų dviejose nuotraukose aptikimo tikslumas. 42 pav. pavaizduotas rezultatų langas. Skirtingai nei tiriamasis filtras, FAST Features ir Harris metodai patys savyje neturi charakteringųjų taškų porų radimo funkcijos, todėl norit surasti taškų poras šiais dviem metodais buvo pasinaudota Ransack metodu, šešioms charakteringųjų taškų poroms rasti.



Atrasta kampų: 30; Trukmė: 30558

FAST:
Atrasta kampų: 30; Trukmė: 3122,717

Harry
Atrasta kampų: 30; Trukmė: 3439,321

42 pav. Viso algoritmo greitimeikos ir tikslumo palyginimas įvairiais metodais.

Matant rezultatų langą, nesunku pastebėti, kad FAST Features kampų atpažinimas ir porų suradimas vyksta greičiausiai, bet ne taip tiksliai kaip tiriamojo filtro, tačiau pamažinus nuotraukos rezoliucijas visų metodų greitimeika padidėja, o teisingai surastų porų skaičius lieka nepakitęs, priklausomai nuo nuotraukų poros surastų kampų poros pakeitus rezoliucija kartais skiriasi, bet tikslumas išlieka toks pats. Tai pačiai nuotraukai, kaip pavaizduota 43 pav. pakeitus rezoliuciją gaunami rezultatai, parodyti 17 lentelėje.

17 lentelė. Greitimeikos ir tikslumo rezultatai pakeitus nuotraukos rezoliucija.

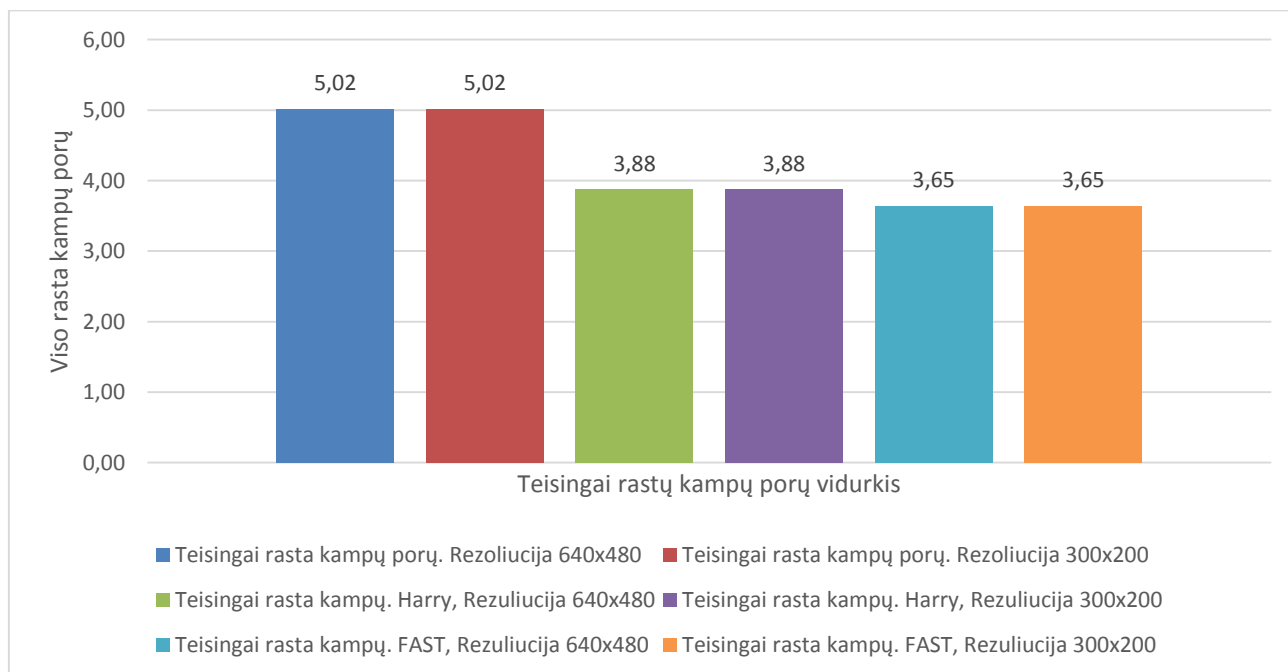
Rezultatai, tiriant nuotraukas ir ieškant kampų porų					
Tiriamasis filtras				Harris filtras	
Greitimeika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų	Greitimeika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų	Greitimeika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų
30558	5	4244,167	5	3439,321	4
Rezultatai, tiriant nuotraukas ir ieškant kampų porų					
Harris filtras		FAST Features filtras			
Greitimeika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų	Greitimeika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų	Greitimeika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų
1932,203	4	3122,717	3	2691,997	3

Iš 17 lentelės rezultatų matyti, kad sumažinus greitaveiką, tiriamasis filtras savo greitaveika (palyginus su didesnės rezoliucijos nuotraukomis) ne tiek daug atsilieka nuo FAST Features ir Harris filtrų, tačiau juos aplenkia savo tikslumu.

18 lentelėje pateikta vidutinės viso algoritmo veikimo greitaveikos ir tikslumo vertės. Visa detali eksperimentų rezultatų lentelė pateikta diske.

18 lentelė. Viso algoritmo vidutinės greitaveikos ir tikslumo palyginimas

Rezultatai, tiriant nuotraukas ir ieškant kampų porų						
Tiriamasis filtras				Harris filtras		
	Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų	Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų	Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų
Vidurkis:	32308,77	5,02	4487,33	5,02	3474,33	3,88
Rezultatai, tiriant nuotraukas ir ieškant kampų porų						
Harris filtras		FAST Features filtras				
Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų	Greitaveika, ms. Rezoliucija 640x480	Teisingai rasta kampų porų	Greitaveika, ms. Rezoliucija 300x200	Teisingai rasta kampų porų	
1951,87	3,88	3135,53	3,65	2703,04	3,65	



43 pav. Teisingai rastų 6 kampų porų dviejuose nuotraukose vidurkis

Taigi, palyginus metodus buvo pastebėta, kad FAST Features metodas visais atvejais yra greitesnis už Harris ir tiriamąjį metodą, tačiau dėl metodo greito veikimo, jis buvo pats netikslingiausias.

Tiriant vidutinės rezoliucijos (640x480) nuotraukas pastebėta, kad tiriamasis metodas yra lėtas, tiriant vieną nuotrauką ir ieškant 30 kampų jo vidutinis radimo greitis buvo 20,06 s, kaip tuo tarpu Harris metodas su tą pačią užduotį įvykdė per 0,1 s, o FAST Features – per 0,07 s, tačiau tiriamasis filtras teisingai atpažino 22,99 kampus iš 30, kai tuo tarpu Harris metodo rezultatas buvo 18,13 kampų iš 30, o FAST Features – 17,41 kampo iš 30 atrastų.

Atliekant tą patį eksperimentą su tomis pačiomis nuotraukomis, tik su mažesne rezoliucija (300x200) rezultatų išvada išliko tokia pati (FAST FEATURES Features metodas greičiausias, bet netikslingiausias, o tiriamasis metodas lėčiausias, bet tiksliausias), tačiau patys rezultatai pasikeitė. 30-ties kampų atpažinimo laikas: tiriamasis filtras 30 kampų rado per vidutiniškai 2,78617 s, FAST Features per 0,061 s, o Harris per – 0,076, pasikeitė ir kampų atpažinimo tikslumas: tiriamasis filtras vidutiniškai aptiko 21,97 kampus iš 30, FAST Features rado 15,42 iš 30, o Harris teisingai aptiko 16,13 kampų iš 30 rastų charakteringųjų taškų. Taigi, iš šio eksperimento matyti, kad dirbant su mažesnės rezoliucijos nuotraukomis metodo greitaveika padidėjo (~7,2 karto), bet sumažėja tikslumas, nors ir mažai (sumažėja vidutiniškai 4-5 proc.). Harris ir FAST Features metodų greitaveika taip pat didėja, apie du kartus, o jų tikslumas krenta apie 10 proc.

Atliekant tą patį eksperimentą su dviem nuotraukomis gaunamas analogiškas rezultatas, kaip ir dirbant su viena nuotrauka, tik kampų atradimo laikas kiekvienam metodui išauga apie 1,5-2 kartus, tai yra natūralu, kadangi tikrinama ne viena, o jau dvi nuotraukos.

Galiausiai buvo atliekamas viso algoritmo greitaveikos ir tikslumo palyginimas naudojant dvi nuotraukas (eksperimentas vėlgi buvo daromas su aukštesne ir žemesne rezoliucijomis), kur iš pradžių buvo atliekamas 30 kampų radimas, o vėliau iš jų atrenkamos 6 geriausiai viena kitą atitinkančios kampų poros. Su vidutinės rezoliucijos nuotraukomis tiriamojo filtro greitaveika buvo 30,3914 s, Harris 3,4743, o FAST Features užduotį atliko per 3,1355 s. Tikslumas ieškant 6 atitinkančių porų su tiriamuoju filtru buvo 5,02 (83,67%), su FAST Features – 3,65 (60,83%), o su Harris – 3,88 (64,67%).

Pakeitus rezoliuciją teisingai atrastų porų skaičius nepasikeitė, priklausomai nuo nuotraukų poros kai kur pasitaikė, kad atrado nebe tokias pačias kampų poras kaip su didesnės rezoliucijos

nuotraukomis, bet teisingai atrastų porų skaičius ir tikslumas nepakito. Pakito greitaveika: tiriamojo filtro greitaveika buvo 4,221 s, Harris 2,703s, o FAST Features užduotį atliko per 1,9518 s.

Taigi su mažesnės rezoliucijos nuotraukomis tiriamasis filtras greitaveika nuo Harris atsilieka 1,56 karto, o nuo FAST Features 2,17 karto, bet tiriamasis filtras net 23% tiksliau nei FAST Features ir 19% tiksliau nei Harris randa vienodas kampų poras. Tiriamasis metodas greitaveika atsilieka nuo šių filtrų, nes kampą atpažįsta ne tik pagal spalvos intensyvumo pokytį lyginant su kaimyniniais pikseliais, bet ir pagal kampo dydį, kampo kryptį, kampo orientaciją, fono kontrasto pasikeitimą. Toks filtro tikslumas kainuoja laiko.

IŠVADOS IR REZULTATAI

1. Tiriant filtro tikslumą buvo atlikta po 150 bandymų, ieškant 6 geriausių charakteringųjų taškų porų iš dviejų nuotraukų. Pilkame fone teisingai aptiktų porų vidurkis yra 83,67% (pasikliautinumo intervalas (4,885; 5,175)), spalvoto fono mėlynos spalvos (B) sluoksnyje - 75% (pasikliautinumo intervalas (4,333; 4,667)), spalvoto fono žalios spalvos (G) sluoksnyje 75,17% (pasikliautinumo intervalas (3,338; 4,682)), spalvoto fono raudonos spalvos (R) sluoksnyje 74,17% (pasikliautinumo intervalas (4,285; 4,614)), spalvotame fone su algoritmu be atstumo įvertinimo – 62,83% (pasikliautinumo intervalas (3,592; 3,948)), spalvotame fone su algoritmu įvertinančiu atstumą – 57,17% (pasikliautinumo intervalas (3,246; 3,614)).

2. Ištirtas atstumo erdvėje įverčio radimas, remiantis dviejų kamerų vaizdais. Atlikus bandymus pastebėta, kad kamerai esant nedideliu atstumu nuo tiriamo daikto, apskaičiuojami atstumai yra priimtino tikslumo, jų paklaida nesiekia 5 %. Nuo 0,5 m iki 0,8 m, paklaida taip pat nėra labai didelė, nesiekia 10%, nuo 0,8 m iki 1 m paklaida dar neviršija 20%, peržengus 1 m ribą, paklaida didėja sparčiau ir gali siekti 30%. Rekomendacija: norint tikslinti tolimų objektų atstumo skaičiavimo tikslumą, reikėtų didinti atstumą tarp kamerų.

3. Tiriant vidutinės rezoliucijos (640x480) nuotrauką ir ieškant 30 kampų pastebėta, kad tiriamasis metodas yra lėtas, tiriant vieną nuotrauką ir ieškant 30 kampų jo vidutinis radimo greitis buvo 20,06 s, su Harris - 0,1 s, o FAST Features – per 0,07 s, tačiau tiriamasis filtras kokybiškiau atpažįsta kampus: 22,99 kampus iš 30. Harris ir FAST Features metodams buvo pritaikytas ekspertinis porų atpažinimo vertinimas Harris atrado - 18,13 kampų iš 30, o FAST Features – 17,41 kampo iš 30 atrastų.

4. Tiriant tokias pat, tik mažesnes rezoliucijos nuotraukas (300x200) metodo greitaveika didėja (~7,2 karto). Harris ir FAST Features metodų greitaveika taip pat didėja, tik apie du kartus, o jų tikslumas krenta apie 10 proc.

5. Tiriamasis metodas greitaveika atsilieka nuo šių filtrų, nes kokybiškiau įvertina, kur yra kampai paveiksle, taip pat kampą atpažįsta ne tik pagal spalvos intensyvumo pokytį lyginant su kaimyniniais pikseliais, bet ir pagal kampo dydį, kampo kryptį, kampo orientaciją, fono kontrasto pasikeitimą. Tiriamasis filtras lengvai atranda kampų poras dviejuose paveiksluose, o šios savybės neturi nei Harris nei FAST Features charakteringųjų taškų atpažinimo filtrai.

Literatūros sąrašas

- [1] – Skaitmeninis vaizdas. [žiūrėta 2016 04 24 d.]. Prieiga per internetą: <<https://sites.google.com/site/learnimagej/image-processing/what-is-a-digital-image>>;
- [2] – Dervinis D. (2012), „Vaizdų apdorojimas“, TEV, ISBN 978-609-433-107-7;
- [3] – Christopher M. Bishop (2006), „Pattern recognition and machine learning“, ISBN 978-0-387-31073-2;
- [4] – Mubarak Shah (1997), „Fundamentals of computer vision“, Orlando. Prieiga per internetą: <<http://www.cse.unr.edu/~bebis/CS485/Handouts/ShahBook.pdf>>;
- [5] – Torbjorn Ersfjord Bjork (2006), “Gray scale image analysis”, Prieiga per internetą: <http://folk.uio.no/torbjoeb/image_analysis/gray_scale_image_analysis_2006.pdf>;
- [6] – R.V.V.Krishna, S.Srinivas Kumar (2015), “Color image segmentation using pixel wise support vector machine classification”. Prieiga per internetą: <<http://airconline.com/sipij/V6N5/6515sipij04.pdf>>;
- [7] – C. Harris and M. Stephens straipsnis (1988) "A combined corner and edge detector".
- [8] – Yung-Yu Chuang paskaitos “Digital Visual Effects” skaidrės, [žiūrėta 2015 m. 06 15 d.]. Prieiga per internetą: <http://www.csie.ntu.edu.tw/~cyy/courses/vfx/07spring/lectures/handouts/lec04_feature_4up.pdf>.
- [9] – Robert B. Fisher, Toby P. Breckon, Kenneth Dawson-Howe, Andrew Fitzgibbon, Craig Robertson, Emanuele Trucco, Christopher K. I. Williams (2014), Dictionary of Computer Vision and Image Processing, 2nd edition, USA, Wiley, ISBN 978-1-119-94186-6
- [10] – Edward Rosten, Reid Porter, and Tom Drummond, (2008), “FAST Featureser and better: a machine learning approach to corner detection”. , [žiūrėta 2016 m. 05 15 d.]. Prieiga per internetą: <<http://arxiv.org/pdf/0810.2434.pdf>>;
- [11] – Kompiuterinė rega. [žiūtėta 2015 m. 05 15 d.]. Prieiga per internetą: <<http://homepages.inf.ed.ac.uk/rbf/online/>>
- [12] – FAST Features kampų atpažinimo metodas. [žiūrėta 2016 m. 04 17 d.]. Prieiga per internetą: <http://web.eecs.umich.edu/~silvio/teaching/EECS598_2010/slides/11_16_Hao.pdf>;

- [13] – Sobelio operatoriaus aprašymas. [žiūrėta 2015 04 24 d.]. Prieiga per internetą: <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm>>.
- [14] – Sobelio operatorius. [žiūrėta 2015 04 18 d.] Prieiga per internetą: <https://en.wikipedia.org/wiki/Sobel_operator#cite_note-1>.
- [15] – Irwin Sobel. Istorija ir Sobelio operatoriaus aprašymas. Prieiga per internetą: http://www.researchgate.net/publication/239398674_An_Isotropic_3_3_Image_Gradient_Operator
- [16] – Elif Aybar straipsnis “Sobel edge method for MATLAB”. Prieiga per internetą: <http://figes.com.tr/matlab/teknik-makaleler/eaybar_tam_metin.pdf>.
- [17] – Emanuele Trucco, Alessandro Verri (1998), Introductory techniques for 3-D computer vision, USA, Prentice Hall ISBN 978-0132611084
- [18] – Gauso suliejimas. [žiūrėta 2015 05 04 d.]. Prieiga per internetą: <http://en.wikipedia.org/wiki/Gaussian_blur>.
- [19] – Gauso funkcija. [žiūrėta 2015 05 04 d.]. Prieiga per internetą: <http://en.wikipedia.org/wiki/Gaussian_function>.
- [20] – Korn, Theresa M.; Korn, Granino Arthur (2000), Mathematical Handbook for Scientists and Engineers: Definitions, Theorems, and Formulas for Reference and Review, New York: Dover Publications, ISBN 0-486-41147-8.
- [21] – Gradiento skaičiavimas ir apibrėžimas. [žiūrėta 2015 06 12 d.]. Prieiga per internetą: <<http://mathworld.wolfram.com/Gradient.html>>.
- [22] – Gradiento skaičiavimas ir apibrėžimas. [žiūrėta 2015 06 11 d.]. Prieiga per internetą: <<http://lt.wikibooks.org/wiki/Matematika/Gradientas>>
- [23] – Luis Otavio Alvares, Vania Bogorny , Bart Kuijpers , Bart Moelans , Jose Antonio Fernandes de Macedo³ , Andrey Tietbohl Palma, “Towards Semantic Trajectory Knowledge Discovery”. Prieiga per internetą: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.9912&rep=rep1&type=pdf>>;

[24] – Prof. Rimvydas Simutis (2015), “Roboto ranka neįgaliesiems: QUADRIBOT” Prieina per internetą:

<http://www.mita.lt/uploads/documents/eureka/quadribot_2015_ktu_ataskaita_koreguota.pdf>;

[25] – Taksi geometrija. [žiūrėta 2015 m. 05 18 d.]. Prieiga per internetą:

<<http://mathworld.wolfram.com/TaxicabMetric.html>>.

[26] – Manhattan'o atstumas. [žiūrėta 2015 m. 06 10 d.]. Prieiga per internetą:

<<http://xlinux.nist.gov/dads//HTML/manhattanDistance.html>>.

[27] – Euklido atstumas. [žiūrėta 2015 m. 06 14 d.]. Prieiga per internetą:

<https://en.wikipedia.org/wiki/Euclidean_distance>.

[28] – Dominyko Rentelio bakalauro baigiamasis darbas „Stereometrijos metodo trimatei kamuoliuko loklizacijai sukūrimas ir eksperimentinis tyrimas su ABB IRM 120 robotu“, 2015, darbo vadovas doc. dr. Renaldas Urniežius.

[29] – M. S. Guzel, Mobile Robot Navigation using a Vision Based Approach, 2009

<http://www.ncl.ac.uk/mech/students/conference/documents/Guzel.pdf>

[30] – C. Lin, E. Setiawan, Object Orientation Recognition Based on SI6 and SVM by Using Stereo Camera, International Conference on Robotics and Biomimetics, 2009

[31] – R. Urniežiaus trimatės padėties nustatymo remiantis dviejų kamerų vaizdais metodas

[32] – Solomon Negash, Michael E. Whitman. Editors: Solomon Negash, Michael E. Whitman, Amy B. Woszczyński, Ken Hoganson, Herbert Mattord. Handbook of Distance Learning for Real-Time and Asynchronous Information Technology Education, Idea Group Inc (IGI), 2008, p. 17, ISBN 1-59904-964-3, ISBN 978-1-59904-964-9. Note costing: "students had the option to install a webcam on their end (a basic webcam costs about \$40.00) to view the class in session."

[34] – Servo pavara. [žiūrėta 2015 m. 12 14 d.]. Prieiga per internetą:

<<http://www.rcdude.com/product-p/hp-ds13tcb.htm>>;

[35] – Žingsninio variklio veikimo principo paaiškinimas. [žiūrėta 2015 m. 01 28 d.]. Prieiga per internetą:

<<http://www.zaber.com/wiki/Tutorials/Microstepping>>;

- [36] – Žingsninis variklis. [žiūrėta 2015 m. 01 28 d.]. Prieiga per internetą: <<https://learn.adafruit.com/all-about-stepper-motors/what-is-a-stepper-motor>>;
- [37] – Arduino aprašymas. [žiūrėta 2015 m. 02 02 d.]. Prieiga per internetą: <<https://learn.sparkfun.com/tutorials/what-is-an-arduino>>;
- [38] – Programavimo kalbos. [žiūrėta 2015 m. 01 28 d.]. Prieiga per internetą: <<http://ims.mii.lt/valentina/publ/Progrkalb.pdf>>;
- [39] – 10 populiariausių programavimo kalbų. [žiūrėta 2016 m. 05 03 d.]. Prieiga per internetą: <<https://www.english4it.com/unit/9/reading>>;
- [40] – C# programavimo kalba. [žiūrėta 2016 m. 05 03 d.]. Prieiga per internetą: <http://home.hit.no/~hansha/programming_languages.htm#CSharp>;
- [42] – Visual Studio. [žiūrėta 2016 m. 05 03 d.]. Prieiga per internetą: <<https://camhang.files.wordpress.com/2012/09/beginning-visual-c-2010.pdf>>;
- [43] – Burbulo rūšiavimo algoritmas. [žiūrėta 2016 m. 05 14 d.]. Prieiga per internetą: <http://www.techmat.vgtu.lt/konspektai/Algoritmu%20teorija/Paskaita7_NEW.pdf>;
- [44] – Pekarskas V., (2010) „Trumpas matematikos kursas“, „Technologija“, ISBN 9955-09-858-9
- [45] - Kristen Grauman, Bastian Leibe (2011), Visual Object Recognition. Synthesis Lectures on Artificial Intelligence & Machine Learning, USA, Morgan & Claypool, ISBN 978-1598299687
- [46] – Robert Collins paskaitos “Harris Corner Detector” skaidrės, [žiūtėta 2015 m. 06 15 d.]. Prieiga per internetą: <<http://www.cse.psu.edu/~rtc12/CSE486/lecture06.pdf>>.
- [47] – Mark S. Nixon, Alberto S. Aguado (2012), Feature extraction & image processing for computer vision, USA, Academic Press, ISBN 978-0123965493