

ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS FAKULTETAS
ELEKTRONIKOS KATEDRA

Donatas Šlaustas

Elektronikos magistrantūros studentas

ERDVINIŲ OBJEKTŲ GEOMETRIJOS NUSTATYMAS KOMPIUTERINE REGA

Magistro darbas

Mokslinis vadovas

Doc. dr. Gintautas Daunys

Šiauliai, 2011

Darbas originalus.....Donatas Šlaustas

SUMMARY

Šlaustas D. 3D: Master thesis of electrical engineer/research advisor Doc. Dr. G. Daunys; Siauliai University, Technological Faculty, Electrical Engineering Department. – Siauliai, 2011. – 60 psl.

This master thesis is about an investigation of the performance of the three-dimensional image reconstruction from two calibrated two-dimensional images. Objectives: to examine the existing three-dimensional imaging and reconstruction algorithms.

Scientists have carried out numerous studies of computer vision systems that receive accurate information about the observed 3D scenes or objects. Stereo-vision system is most often implemented using two video cameras, which are acquiring two-dimensional images. Three-dimensional image is obtained by processing two or more two-dimensional images. Despite that a lot of stereo vision algorithms were offered during last years still it is an issue accurate reconstruction of object 3D geometry.

The different stereo-vision system setups and camera calibration methods are analyzed in the first part of the thesis. The discussion of corner detectors is presented in next part. The points matching algorithms also are analyzed in the thesis. In the experimental part of the thesis, the used software and obtained reconstruction results are presented.

TURINYS

SUMMARY	2
TURINYS	3
PRIEDAI	4
LENTELĖS	5
PAVEIKSLAI	6
SANTRUMPOS IR PAGRINDINIAI ŽYMĖJIMAI	8
ĮVADAS	9
1. ALGORITMŲ ANALIZĖ	11
1.1. Idealaus stereovaizdo kameros modelis	11
1.2. Kalibravimo metodai	13
1.2.1. Fotogrametrinis kalibravimas	14
1.2.2. Autokalibravimo metodas	15
1.2.3. Tsai vaizdo kameros kalibravimo metodas	16
1.2.4. Zhang kameros kalibravimo metodas	19
1.2.4.1. Tikslus įvertinimas	20
1.2.4.2. Labiausiai tikėtinas įvertinimas	22
2. Kampų detektoriai	23
2.1. Harris ir KLT kampų detektoriai	23
2.1.1. Harris kampų radimo algoritmas	24
2.1.2. KLT algoritmas	25
2.2. Harris-Stephens ir Noble kampų detektorius	27
2.3. Ryškumo gradiento radimas	28
3. Tapačių taškų vaizduose radimas	31
3.1. Lucas-Kanade algoritmas	31
3.2. Kalibruotų stereo vaizdų išlyginimas Bouguet metodu	33
3.3. Tapačių taškų radimo algoritmų palyginimas	35
4. Trimačio vaizdo skeneriai	37
5. Eksperimentinis tyrimas	39
5.1. Kodas parašytas OpenCv bibliotekos funkcijomis	39
5.2. Kodas vykdomas MatLab programa	44
IŠVADOS	54
LITERATŪRA	55

PRIEDAI

Priedas nr.1. Paveikslų poros: Tsukuba, Venus, Teddy, Cones.....	58
Priedo nr.1 tęsinys. Paveikslų poros: Tsukuba, Venus, Teddy, Cones.....	59
Priedas nr.2. Kompaktinis diskas.....	60

LENTELĖS

Lentelė nr.1. Taškų atitikimo algoritmų palyginimas.....	36
Lentelė nr.2. Lazerinių skenerių charakteristikų palyginimas.....	38
Lentelė nr.3. Stereo paveikslų koordinatės ir jų skirtumai.....	51

PAVEIKSLAI

<p>1.1 pav. Vartotojiškos stereovaizdo kameros <i>Canon PowerShot S400</i> (kairėje) ir pramoninės gamybos stereovaizdo kameros <i>Sony HDV</i> (dešinėje), kurių vaizdo jutikliai yra vienoje plokštumoje bei optinės ašys lygiagrečios.....</p> <p>1.2 pav. Standartinis idealus stereovaizdo kameros modelis, kai abiejų kamerų optinės ašys yra lygiagrečios viena kitai.....</p> <p>1.3 pav. Iškraipytas vaizdas (kairėje), ištaisytas nuo iškraipymų vaizdas (dešinėje)</p> <p>1.4 pav. Radialinių iškraipymų pavyzdžiai.....</p> <p>1.5 pav. Kalibravimo šablono pavyzdys su neatitinkančiais taškais (kairėje) ir sukalibruotas vaizdas (dešinėje).....</p> <p>1.6 pav. Dviejų vaizdo kamerų epipolinė geometrija. O_L, O_R – kairioji ir dešinioji vaizdo kameros, P – stebimas taškas 3D erdvėje, $P_L E_L, P_R E_R$ – kairiosios ir dešinėsios vaizdo kamerų epipolinės linijos, esančios vaizdo projektavimo plokštumose</p> <p>1.7 pav. Tsai kameros reprojekcijos modelis su perspektyvia projekcija ir radialiniais iškraipymais.....</p> <p>2.1 pav. Kampo radimo algoritmo idėja. Pasirinkto dydžio langas slenkamas įvairiomis kryptimis. Pirmajame iš kairės vaizde nėra jokių esminių taškų intensyvumo pasikeitimų, todėl nustatoma, kad toje vietoje yra tolygi erdvė. Viduriniajame vaizde intensyvumas nekinta slenkant langą aukštyn ir žemyn – toje vietoje yra linija. Trečiajame vaizde taškų intensyvumas kinta slenkant langą visomis kryptimis, taigi, langas yra ant kampo.</p> <p>2.2 pav. Pirmoji eilutė vaizduoja pastatą, nufotografuotą skirtingu atstumu. Antroje eilutėje parodyta Laplaso atsakai pažymėtame taške imant skirtingo didinimo vaizdus. Pirmosios funkcijos maksimumas gaunamas ties didinimu 10,1, antrosios – 3,9; santykis tarp šių dydžių nurodo, kiek kartų pirmasis vaizdas turi būti sumažintas siekiant gauti antrąjį vaizdą. Geltoni apskritimai vaizduoja ypatingų taškų aplinką, kurios spindulys tris kartus didesnis už nustatytą charakteringą dydį.....</p> <p>2.3 pav. KLT kampų radimo algoritmo pavyzdys</p> <p>3.1 pav. Ideali stereo sistema (gali būti gaunama vaizdų išlyginimo metodu). Gylis Z gali būti rastas trianguliacijos metodu. Pagrindinės vaizdo kamerų ašys prasideda nuo projekcijų centrų O_l ir O_r.....</p> <p>4.1 pav. 3D lazerinis skeneris</p> <p>4.2 pav. 3D lazerinio skenerio veikimo principas.....</p> <p>5.1 pav. Kalibravimo šablonas: šachmatų lenta.....</p> <p>5.2 pav. Atliktas paveikslų išlyginimas ir randamos epipolinės linijos.....</p> <p>5.3 pav. Gauti pikselių neatitikimai</p> <p>5.4 pav. Gautas trimatis vaizdas.....</p> <p>5.5 pav. Išlyginti stereo paveikslai.....</p> <p>5.6 pav. Gauti pikselių neatitikimai</p> <p>5.7 pav. Gautas trimatis vaizdas.....</p> <p>5.8 pav. Rasti taškai iš kairės ir dešinės kameros paveikslų.....</p> <p>5.9 pav. Rasti charakteringi taškai iš kairės ir dešinės kameros paveikslų.....</p> <p>5.10 pav. Charakteringų taškų radimo kryptys</p> <p>5.11 pav. Dešinysis paveikslas su charakteringais taškais.....</p> <p>5.12 pav. Kairysis paveikslas su charakteringais taškais</p> <p>5.13 pav. Kalibravimo šablonas; Detektuoti kalibravimo taškai, pagal kuriuos atliekamas abiejų kamerų vidinių ir išorinių parametru kalibravimas</p>	<p>11</p> <p>12</p> <p>13</p> <p>13</p> <p>14</p> <p>15</p> <p>17</p> <p>24</p> <p>25</p> <p>26</p> <p>35</p> <p>37</p> <p>37</p> <p>41</p> <p>41</p> <p>42</p> <p>42</p> <p>43</p> <p>43</p> <p>44</p> <p>45</p> <p>45</p> <p>46</p> <p>46</p> <p>47</p> <p>48</p>
---	---

5.14 pav. Kairiosios ir dešinėsios kamerų židinių nuotolių priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera)	48
5.15 pav. Kairiosios ir dešinėsios kamerų židinių nuotolių santykio priklausomybė nuo vaizdo didinimo	49
5.16 pav. Kamerų vaizdo jutiklių matricų centrinių taškų x koordinatinių priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera).....	49
5.17 pav. Kamerų vaizdo jutiklių matricų centrinių taškų y koordinatinių priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera).....	50
5.18 pav. Tikrosios ir apskaičiuotos objekto koordinatės	50
5.19 pav. Centrinio taško priklausomybė nuo vidutinės kvadratinės paklaidos	52
5.20 pav. Židinio nuotolio priklausomybė nuo vidutinės kvadratinės paklaidos.....	53

SANTRUMPOS IR PAGRINDINIAI ŽYMĖJIMAI

Santrumpa	Angliškas terminas	Lietuviškas terminas
	Stereo Vision	stereorega
2D	Two-dimensional	dvimatis
3D	Three-dimensional	erdvinis, trimatis
CV	Computer Vision	kompiuterinė rega
R	Rotation Matrix	posūkio matrica
T	Translation Matrix	postūmio matrica
OpenCV	Open Computer Vision Library	atvirojo kodo kompiuterinės regos biblioteka

IVADAS

Mokslininkai atlieka daugybę tyrimų apie kompiuterines regos sistemas, galinčias gauti tikslią 3D informaciją apie stebimų scenų objektus ir išsidėstymą. Todėl sparčiai vystosi 3D vaizdų atkūrimo, priėmimo, skenavimo sistemos.

Dažniausiai stereorega įgyvendinama panaudojant dvi vaizdo kameras, kurios registruoja du dvimačius vaizdus. Trimatis vaizdas gaunamas apdorojant du ar daugiau dvimačių vaizdų. Pastaraisiais metais buvo pasiūlyta nemažai stereovaizdo gavimo algoritmų, kuriais galima rasti atitikimus tarp dešiniojo ir kairiojo vaizdo. Norint surasti atitikimus vaizduose, paprasčiausiuose metoduose yra matuojami absoliutiniai arba kvadratiniai skirtumai tarp pikselių ryškumų. Kituose metoduose yra panaudojami langai, kai funkcija tikrina pikselio sritį, ieškodama tinkamiausio lango. Šie metodai dažniausiai netinka, kai vaizde yra smulki ar pasikartojanti tekstūra, todėl funkcijos negali rasti atitikimų vaizduose. Todėl mokslininkai daug dirba su dinaminiu programavimu, kad patobulintų atitikimų suradimo algoritmus, kai yra skenuojama kiekviena linija, o kai kada ir tarp linijų. Gauti duomenys yra apdorojami tikslo funkcija ir įvertinami keliais kriterijais.

Temos aktualumas. Vaizduoti paveikslus ar filmus 3D formatu jau nėra labai didelė naujovė, tačiau kur kas sudėtingiau atkurti tikro daikto formas ir tekstūras kompiuterinėje trimatėje erdvėje. Trimačių vaizdų atkūrimas kompiuterinėje erdvėje gali būti labai plačiai naudojamas. Atsirastų galimybė per atstumą gamykloje užsisakyti gamybai atkurtą kompiuteryje detalę, protezuotojai galėtų gaminti tikslus protezus žmonėms be matavimų, o tik gavus tikslus vaizdus. Taip pat tai leistų kurti bei lengvai perkelti personažus į įvairius filmus ar reklamas. Panaudojimai gali būti ir daugybėje kitų sričių, kur gautus duomenis tereiktų nusiųsti, pavyzdžiui, elektroniniu paštu ir nereiktų atlikti tikslų matavimų ir braižyti ranka brėžinių, kas tikrai užima nemažai laiko.

Darbo tikslas: ištirti trimačio vaizdo atkūrimo iš dviejų kalibruotų dvimačių vaizdų galimybes.

Uždaviniai:

- Išnagrinėti vaizdo kamerų kalibravimo metodus.
- Išnagrinėti kampų taškų vaizduose suradimo metodus.
- Išnagrinėti tapačių taškų dviejuose vaizduose suradimo metodus.
- Ištirti erdvinių koordinačių nustatymo metodus.

Mokslinis naujumas: ši tema nors ir nagrinėjama daugybės mokslininkų, tačiau nėra iki galo išbaigta. Yra tiriami įvairūs algoritmai bei tobulinami. Pasaulyje yra sukurta panašių prototipų, tačiau

jie yra labai brangūs arba kol kas išvis neprieinami paprastam vartotojui. Todėl tobulinti ir tirti šia sritį dar yra kur.

1. ALGORITMŲ ANALIZĖ

1.1. Idealaus stereovaizdo kameros modelis

Paprasciausias geometrinis modelis gaunamas, kai abiejų vaizdo kamerų optinės ašys yra lygiagrečios, o kameros poslinkis atliekamas tik išilgai X ašies. Kairiosios kameros optinis centras nustatomas koordinatinių sistemos pradžioje. Vaizdo kameros optinis centras – taškas, esantis vaizdo kameros jutiklio plokštumoje, per kurį eina vaizdo kameros optinė ašis. Toks modelis vadinamas standartiniu idealiu stereovaizdo kameros modeliu (1.2 pav.). Šis modelis labiausiai paplitęs dėl savo paprastų ir patogių matematinių išraiškų bei iš to sekančio paprasto pritaikymo praktikoje. Remiantis šiuo modeliu dažnai konstruojamos tiek vartotojiškos tiek ir pramoninės, pavyzdžiui, gamintojų *Canon*, *Sony* stereovaizdo kameros (1.1 pav.).

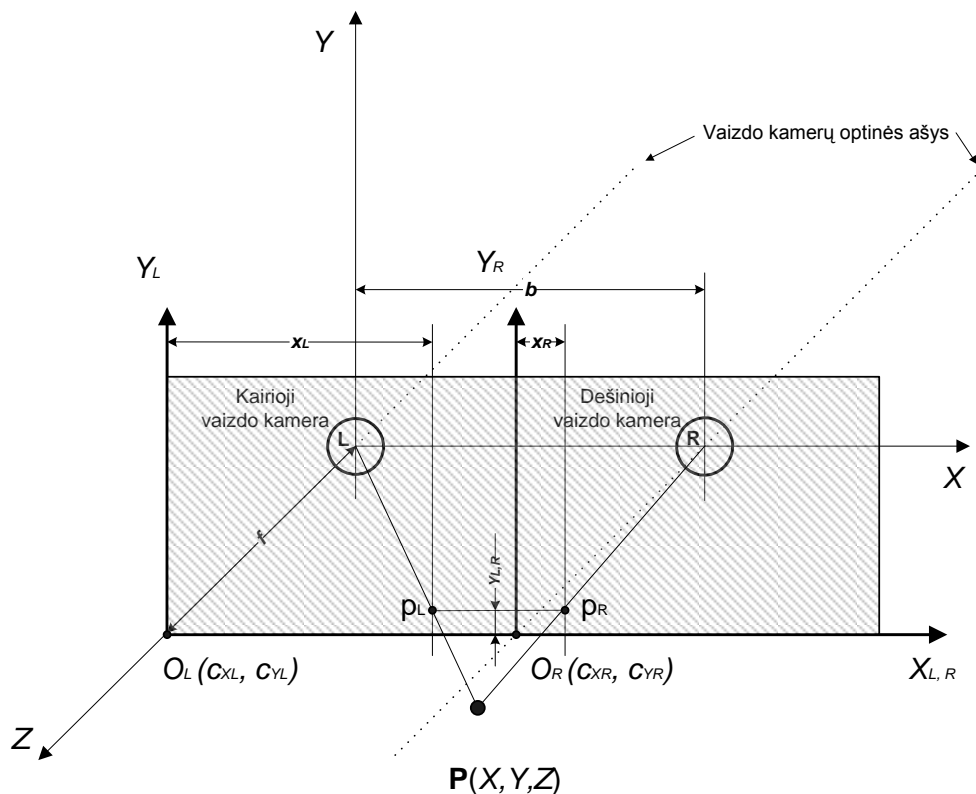


1.1 pav. Vartotojiškos stereovaizdo kameros *Canon PowerShot S400* (kairėje) ir pramoninės gamybos stereovaizdo kameros *Sony HDV* (dešinėje), kurių vaizdo jutikliai yra vienoje plokštumoje bei optinės ašys lygiagrečios

Apžvelkime idealų stereovaizdo kameros modelį detaliau [1] (1.2pav.):

- L ir R yra kairioji ir dešinioji vaizdo kameros, kurių optinės ašys yra lygiagrečios.

- Bazinė linija – tai tiesės atkarpa, jungianti du optinius centrus, kuri yra statmena optinėms ašims. Tegul b bus atstumas tarp abiejų kamerų optinių centrų globalioje 3D erdvėje.
- ZX plokštumoje yra optinės ašys. Ši plokštuma yra statmena abiejų vaizdo kamerų jutiklių plokštumoms.
- X ašis sutampa su bazine linija, ir globalios koordinatinių sistemos (X, Y, Z) pradžia yra kairiosios kameros optinis centras.
- $X_L O_L Y_L$ – kairiosios vaizdo kameros jutiklio koordinatinių sistema.
- $X_R O_R Y_R$ – dešinėsios vaizdo kameros jutiklio koordinatinių sistema.
- $O_L(c_{XL}, c_{YL}), O_R(c_{XR}, c_{YR})$ – kairiosios ir dešinėsios vaizdo kamerų jutiklių plokštumų optinių centrų koordinatės.
- p_L, p_R – globalioje 3D erdvėje esančio taško P koordinatės vaizdo kamerų jutiklių plokštumose.



1.2 pav. Standartinis idealus stereovaizdo kameros modelis, kai abiejų kamerų optinės ašys yra lygiagrečios viena kitai

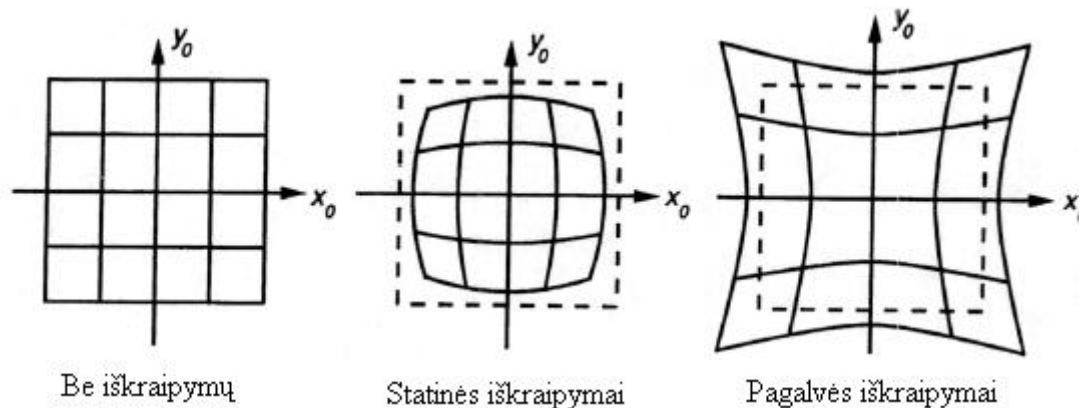
1.2. Kalibravimo metodai

Skirtingoms vaizdo kameroms turi būti nustatomi skirtingi parametrai. Dėl savo paprastumo labiausiai paplitusios mažos angos kameros. Tai tokia vaizdo kamera, kurioje nėra objektyvo, įnešančio geometrinius vaizdo iškraipymus. Objektyvo funkciją atlieka mažo diametro anga. Kamerų vaizduose pasireiškia *radialiniai* ir *tangentiniai* iškraipymai. Radialinius iškraipymus (1.3, 1.4 pav.) sukelia kameros objektyvo lęšis, kuris dažniausiai būna sferos formos. Tangentiniai iškraipymai pasireiškia dėl visos kameros konstrukcijos netobulumų arba nekokybiško jos surinkimo gamykloje. Visi kalibruojami vaizdo kameros parametrai gali būti:

- vidiniai parametrai – vaizdo kameros objektyvo ir vaizdo jutiklio matricos vidinės geometrinės ir optinės charakteristikos;
- išoriniai parametrai – vaizdo kameros padėtis erdvėje ir jos orientavimas realios aplinkos koordinačių sistemos atžvilgiu.



1.3 pav. Iškraipytas vaizdas (kairėje), ištaisytas nuo iškraipymų vaizdas (dešinėje)



1.4 pav. Radialinių iškraipymų pavyzdžiai

Dviejų vaizdo kamerų kalibravimas stereoregos atveju yra atliekamas tokia tvarka: pirmiausia kalibruojama kiekviena vaizdo kamera atskirai, po to atliekama išorinių parametru geometrinė transformacija ir surandamos stereovaizdo kamerų geometrinės charakteristikos.

Vaizdo kamerų kalibravimo metodai yra klasifikuojami pagal matematinio aparato pritaikymą:

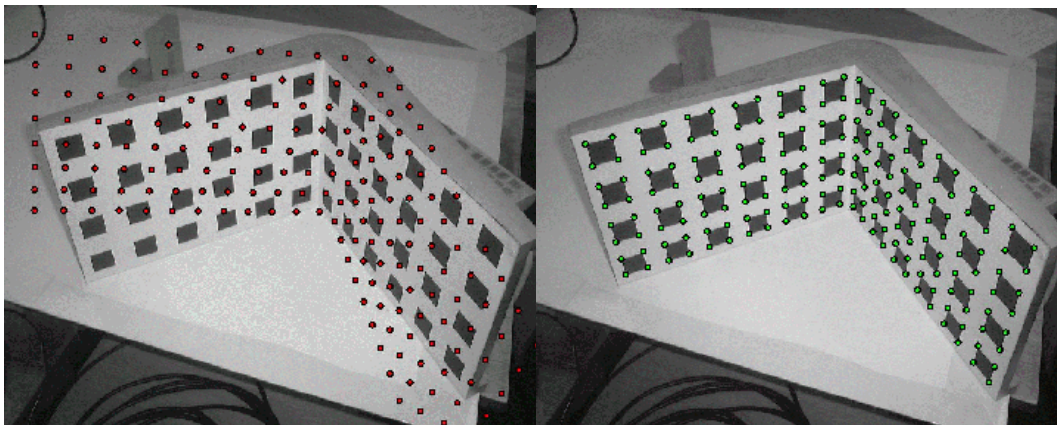
- baigtinės formos sprendinio radimas;
- netiesinis optimizavimas;
- dviejų etapų metodai.

Kitas kalibravimo metodų klasifikavimas yra skirstant jas pagal tai, ar būtinas kalibravimo šablonas:

- fotogrametrinis kalibravimas;
- autokalibravimas.

1.2.1. Fotogrametrinis kalibravimas

Fotogrametrinio kalibravimo [2] metu vaizdo kamera fotografuojamas precizinis trimatis kalibravimo šablonas, kurio geometrinė forma ir matmenys trimatėje erdvėje yra žinomi ir tikslūs. Kalibravimo šablonas sudaromas iš dviejų arba trijų ortogonalinių plokštumų (1.5 pav.) [3]. Šios plokštumos sumontuojamos ypač tiksliai. Plokštumose esančių kvadratų formos taip pat turi būti preciziški, nes jų briaunų vaizdo projekcijos naudojamos atraminiais 3D erdvės taškams įvertinti.

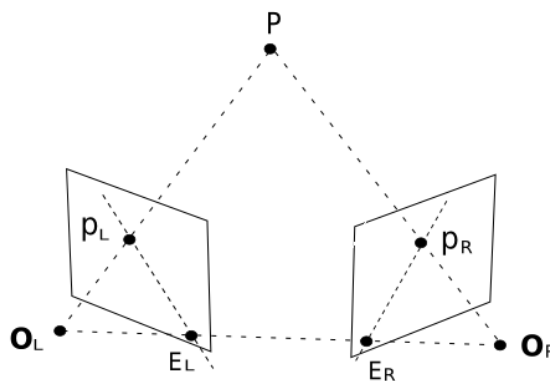


1.5 pav. Kalibravimo šablono pavyzdys su neatitinkančiais taškais (kairėje) ir sukalibruotas vaizdas (dešinėje)

Fotogrametrinis kalibravimas vykdomas šia tvarka:

- Vaizdo kameromis fotografuojamas/filmuojamas kalibravimo šablonas (1.5 pav.).
- Atliekama daugiakampė kontūrų aproksimacija, kurioje automatiškai išskiriami kampai ir apskaičiuojamos jų koordinatės vaizde.
- Apskaičiuojami vidiniai ir išoriniai vaizdo kameros parametrai.

Kalibravimo rezultatus galima kokybiškai įvertinti patikrinant, ar *epipolinė geometrija* (1.6 pav.) yra teisinga, t.y., ar *epipolinės linijos*, einančios per kiekvienos kalibruojamos kameros vaizdo projekcijos plokštumoje esančius kalibravimo taškus, sutampa. Prieš tai būtina atlikti abiejų kamerų vaizdo projekcijų išlyginimą, transformuojant juos į vieną plokštumą.



1.6 pav. Dviejų vaizdo kamerų epipolinė geometrija. O_L , O_R – kairioji ir dešinioji vaizdo kameros, P – stebimas taškas 3D erdvėje, $P_L E_L$, $P_R E_R$ – kairiosios ir dešinėsios vaizdo kamerų epipolinės linijos, esančios vaizdo projektavimo plokštumose

Fotogrametriniams kalibravimo metodams daugeliu atvejų būtina brangi kalibravimo įranga (precizinis kalibravimo šablonas), be to, techninės įrangos paruošimas darbui yra gana sudėtingas, paprastai reikalaujantis laboratorinių sąlygų.

1.2.2. Autokalibravimo metodas

Šiame metode [2] nenaudojami kalibravimo šablonai. Keičiant kameros padėtį bei esant statiniam vaizdai, bendruoju atveju, vaizdas gali būti apibrėžiamas kameros vidiniais parametrais, gaunamais iš vienos kameros poslinkio, pasinaudojant tik vaizdo informacija. Todėl, jei vaizdai yra paimti iš tos pačios kameros su fiksuotais vidiniais parametrais, tai surasti atitikmenys tarp trijų vaizdų yra pakankami, kad atkurti tiek vidinius, tiek išorinius parametrus, ir tokiu būdu galima tiksliai rekonstruoti erdvinę struktūrą. Šis metodas yra labai lankstus, bet jame egzistuoja daugybė apytikriai

įvertinamų parametrų, dėl kurių atsiranda paklaidos ir todėl jie ne visuomet sąlygoja patikimus vaizdo kamerų kalibravimo rezultatus.

Autokalibravimas – tai procesas, kurio metu yra tiesiogiai surandami vidiniai kameros parametrai pagal vaizdus, priimtus iš nekalibruotos kameros. Kai vidiniai parametrai jau surasti, galima atlikti vaizdų metrinę rekonstrukciją. Autokalibravimas leidžia išvengti sudėtingos vaizdo kamerų kalibravimo procedūros, naudojančios kalibravimo įvairius objektus (šablonus). Tai suteikia akivaizdžiai didesnę lankstumą, kai kamera gali būti kalibruojama pagal vaizdų seką, nepaisant nežinomų judesio ir kai kurių vidinių parametrų pasikeitimų.

1.2.3. Tsai vaizdo kameros kalibravimo metodas

Kameros kalibravimas ir pozicijos įvertinimas yra pagrindiniai klausimai kompiuterinėje regoje, nes jie yra susiję su daugeliu vaizdų apdorojimo problemų, tokių kaip stereorega, struktūra iš judesio, robotų navigacija ir pokyčių nustatymas.

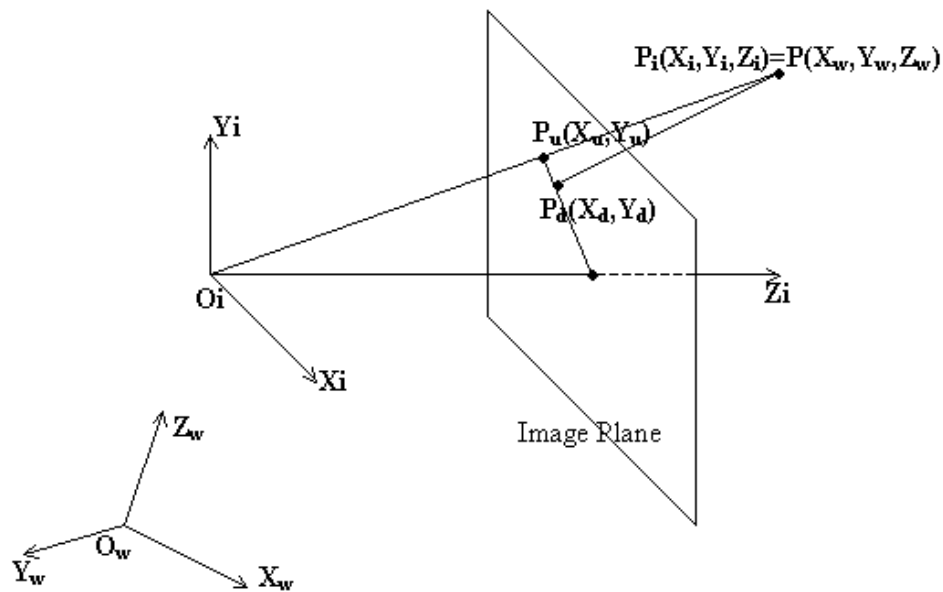
Kameros kalibravimas susidaro įvertinus tos pačios nekalibruotos kameros modelį. Tikslas yra rasti išorinius parametrus (padėties ir orientacijos palyginimas su pasaulio koordinačių sistema), ir vidiniai kameros parametrai (pagrindinis taškas arba paveikslo centras, židinio nuotolis ir iškraipymų koeficientai). Vienas iš labiausiai naudojamų kameros kalibravimo metodų yra tas, kurį siūlo Tsai [4]. Jo įgyvendinimui reikia charakteringų 3D taškų koordinačių ir 2D taškų paveiksle. Apskaičiavimui autorius naudoja du etapus: pirma, pozicijos ir krypties, ir, antra, vidinius fotoaparato parametrus.

Daugelyje kompiuterinės regos taikyme manoma, kad kamera jau kalibruota. Tai reiškia, kad vidiniai kameros parametrai yra turimi. Šis modelis gali būti numatytas gamintojo arba apskaičiuotas naudojant žinomus objektus (paprastai šachmatų lenta). Tokia situacija, iššaukusi pozicijos įvertinimą, tiesiog reikalauja atstatyti šešis parametrus lyginant poziciją ir kameros orientaciją. Kai kurie pozicijos įvertinimo metodai taipogi kaip ir jautrumo analizė gali būti rasti.

Vienas iš populiariausių kalibravimo metodų yra gerai žinomas Tsai kameros kalibravimo metodas, kuris turi plačią taikymo sritį, nes jis gali veikti tiek su plokštumos tiek ir ne plokštumos taškais. Šis metodas taip pat suteikia galimybę kalibruoti vidaus ir išorės parametrus atskirai. Ši galimybė yra ypač naudinga, nes ji suteikia galimybę pakoreguoti vidinius kameros parametrus, kai jie yra žinomi, ir atlikti tik plokštumos įvertinimą.

Tsai metodas yra pagrįstas taškiniu objektyvo kameros modeliu ir jame reikia įvertinti vienuolika parametrų:

f – kameros židinio nuotolis,
 k – lešio radialinių iškraipymų koeficientas,
 C_x, C_y – lešio radialinių iškraipymų centro koordinatės,
 S_x – perskaičiavimo faktorius, kuris atsižvelgia į bet kokius nepastovumus, dėl aparatūros skaičiavimo trūkumų skenuojant ir verčiant vaizdą į skaitmeninę informaciją,
 R_x, R_y, R_z – posūkio kampai transformacijai tarp tikrųjų (pasaulio) ir kameros koordinatčių,
 T_x, T_y, T_z – translacijos komponentai transformacijai tarp tikrųjų (pasaulio) ir kameros koordinatčių.



1.7 pav. Tsai kameros reprojekcijos modelis su perspektyvia projekcija ir radialiniais iškraipymais.

Transformacija iš tikrųjų koordinatčių (laboratorinės sistemos) (X_w, Y_w, Z_w) į paveikslo (X_i, Y_i, Z_i) koordinates, atsižvelgiant į išorinius kameros parametrus (translacijos vektorių \mathbf{T} ir posūkio matricą \mathbf{R}), išreiškiama lygtimi:

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \mathbf{T}. \quad (1)$$

Joje matrica \mathbf{R} aprašoma 9 komponentėmis, o vektorius \mathbf{T} - trimis komponentėmis:

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix}; \mathbf{T} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}. \quad (2)$$

Matricos \mathbf{R} komponentės aprašomos lygtimis:

$$\begin{aligned}
r_1 &= \cos R_y \cos R_z ; \\
r_2 &= \cos R_z \sin R_x \sin R_y - \cos R_x \sin R_z ; \\
r_3 &= \sin R_x \sin R_z + \cos R_x \cos R_z \sin R_y ; \\
r_4 &= \cos R_y \sin R_z ; \\
r_5 &= \sin R_x \sin R_y \sin R_z + \cos R_x \cos R_z ; \\
r_6 &= \cos R_x \sin R_y \sin R_z - \cos R_z \sin R_x ; \\
r_7 &= -\sin R_y ; \\
r_8 &= \cos R_y \sin R_x ; \\
r_9 &= \cos R_x \cos R_y .
\end{aligned} \tag{3}$$

čia R_x, R_y, R_z - posūkių aplink tris ašis Oilerio kampai.

3D translacijos parametrai (T_x, T_y, T_z) atitinka postūmius tarp laboratorinės sistemos ir vaizdo koordinatų sistemų. Transformacija iš 3D padėties (vaizdo koordinatų sistemoje) į vaizdo plokštumą yra apskaičiuojama naudojant šiuos veiksmus (žr. 1.7 paveikslą). Pirmiausia atliekama transformacija iš trimačių koordinatų (X_i, Y_i, Z_i) į neiškraipyto vaizdo plokštumos (X_u, Y_u) koordinatas:

$$X_u = f \frac{X_i}{Z_i}, \tag{4}$$

$$Y_u = f \frac{Y_i}{Z_i}. \tag{5}$$

Transformacija iš neiškraipyto vaizdo (X_u, Y_u) į iškraipyto (X_d, Y_d) vaizdo koordinatas:

$$X_u = X_d(1 + kr^2), \tag{6}$$

$$Y_u = Y_d(1 + kr^2), \tag{7}$$

čia $r = \sqrt{X_d^2 + Y_d^2}$, ir k yra lęšio radialinių iškreipimų koeficientas.

Transformacija iš iškraipyto vaizdo plokštumos koordinatų (X_d, Y_d) į galutinio vaizdo koordinatas (X_f, Y_f) yra:

$$X_f = \frac{S_x X_d}{d_x} + C_x, \tag{8}$$

$$Y_f = \frac{Y_d}{d_y} + C_y, \tag{9}$$

čia d_x ir d_y yra atstumai tarp gretimų jutiklio pikselių X ir Y kryptimis. d_x ir d_y yra pastovūs kameros parametrai. Jie priklauso tik nuo vaizdo jutiklio dydžio ir vaizdo skyros.

1.2.4. Zhang kameros kalibravimo metodas

Turime plokštumą, aprašomą lygtimi [2]:

$$\begin{bmatrix} \mathbf{r}_3 \\ \mathbf{r}_3^T \mathbf{t} \end{bmatrix}^T \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = 0 \quad (10)$$

Čia $w=0$, kai taškai yra begalybėje, ir $w=1$ visais kitais atvejais. Ši plokštuma kerta plokštumą begalybėje, todėl mes lengvai galime pamatyti, kad:

$$\begin{bmatrix} \mathbf{r}_1 \\ 0 \end{bmatrix} \quad (11)$$

ir

$$\begin{bmatrix} \mathbf{r}_2 \\ 0 \end{bmatrix} \quad (12)$$

yra du konkretūs taškai ant plokštumų susikirtimo linijos. Visi kiti taškai ant šios linijos yra išvedami per tuos pačius du taškus.

$$\mathbf{x}_\infty = a \begin{bmatrix} \mathbf{r}_1 \\ 0 \end{bmatrix} + b \begin{bmatrix} \mathbf{r}_2 \\ 0 \end{bmatrix} = \begin{bmatrix} a\mathbf{r}_1 + b\mathbf{r}_2 \\ 0 \end{bmatrix} \quad (13)$$

Dabar apskaičiuojame susikirtimą virš linijos. Pagal apibrėžimą taškas x_∞ , žinomas kaip apskritas taškas, tenkina: $x_\infty^T x_\infty = 0$, $\langle a\mathbf{r}_1 + b\mathbf{r}_2, \mathbf{x}_\infty \rangle = 0$, arba $a^2 + b^2 = 0$. Sprendimas yra $b = \pm ai$, kur $i^2 = -1$. Tai reiškia du susikertantys taškai yra:

$$\mathbf{x}_\infty = a \begin{bmatrix} \mathbf{r}_1 \pm i\mathbf{r}_2 \\ 0 \end{bmatrix}. \quad (14)$$

Taškai plokštumoje žymimi $\mathbf{m} = u, v^T$, o taškai trimatėje erdvėje - $\mathbf{M} = X, Y, Z, 1^T$. Ryšys tarp 3D taškų \mathbf{M} ir jų projekcijų vaizdo plokštumoje \mathbf{m} yra išreiškiamas homografijos matrica \mathbf{H} :

$$s\tilde{\mathbf{m}} = \mathbf{A} \mathbf{R} \mathbf{t} \tilde{\mathbf{M}} \quad (15)$$

$$\mathbf{A} = \begin{bmatrix} \alpha & \gamma & c_x \\ 0 & \beta & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{h}_1 \quad \mathbf{h}_2 \quad \mathbf{h}_3 = \lambda \mathbf{A} \mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{t} \quad (16)$$

$$s\tilde{\mathbf{m}} = \mathbf{H} \tilde{\mathbf{M}} \quad (17)$$

$$\mathbf{H} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \quad (18)$$

Negalima teigti, kad šios poros kompleksinių sujungtinių taškų reikšmės yra pastovios Euklido transformacijose. Jų projekcijos vaizdo plokštumoje yra duotos:

$$\tilde{\mathbf{m}}_{\infty} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 \pm i\mathbf{r}_2 \end{bmatrix} = \mathbf{h}_1 \pm i\mathbf{h}_2 \quad (19)$$

Taškas $\tilde{\mathbf{m}}_{\infty}$ yra paveikslas viršūnėje, apibūdinta: $\mathbf{A}^{-T} \mathbf{A}^{-1}$. Tai duoda:

$$\mathbf{h}_1 \pm i\mathbf{h}_2 \quad \mathbf{A}^{-T} \mathbf{A}^{-1} \quad \mathbf{h}_1 \pm i\mathbf{h}_2 = 0 \quad (20)$$

Tokiu atveju abi dalys: realioji ir menamoji – turi būti lygios nuliui:

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 = 0 \quad (21)$$

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 \quad (22)$$

Norint kuo efektyviau išspręsti kameros kalibravimo uždavinį, pradedama nuo analitinio sprendimo. Toliau vykdomas netiesinis optimizavimas pagrįstas maksimalios tikimybės kriterijumi. Galiausiai panaudojamas lęšio iškraipymų įvertinimas, duodantis abu analitinį ir netiesinį sprendinius.

1.2.4.1. Tikslus įvertinimas

Įvedama matrica \mathbf{B} lygtimis:

$$\mathbf{B} = \mathbf{A}^{-T} \mathbf{A}^{-1} \equiv \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{12} & B_{22} & B_{23} \\ B_{13} & B_{23} & B_{33} \end{bmatrix} = \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2 \beta} & \frac{c_y \gamma - c_x \beta}{\alpha^2 \beta} \\ -\frac{\gamma}{\alpha^2 \beta} & \frac{\gamma^2}{\alpha^2 \beta^2} + \frac{1}{\beta^2} & -\frac{\gamma}{\alpha^2 \beta^2} \frac{c_y \gamma - c_x \beta}{\beta^2} - \frac{c_y}{\beta^2} \\ \frac{c_y \gamma - c_x \beta}{\alpha^2 \beta} & -\frac{\gamma}{\alpha^2 \beta^2} \frac{c_y \gamma - c_x \beta}{\beta^2} - \frac{c_y}{\beta^2} & \frac{c_y \gamma - c_x \beta}{\alpha^2 \beta^2} + \frac{c_y^2}{\beta^2} + 1 \end{bmatrix} \quad (23)$$

Iš lygties (23) matome, kad \mathbf{B} yra simetrinė matrica, turinti šešias skirtingas komponentes. Surašydami šias skirtingas komponentes į vektorių, gauname vektorių \mathbf{b} :

$$\mathbf{b} = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T \quad (24)$$

Matricos \mathbf{H} i -tasis stulpelis yra vektorius $\mathbf{h}_i = [h_{i1}, h_{i2}, h_{i3}]^T$. Tada turime

$$\mathbf{h}_i^T \mathbf{B} \mathbf{h}_j = \mathbf{v}_{ij}^T \mathbf{b} \quad (25)$$

čia

$$\mathbf{v}_{ij} = \left[h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2}, +h_{i2}h_{j3}, h_{i3}h_{j3} \right]^T \quad (26)$$

Dėl to, du pagrindiniai apribojimai (21) ir (22), iš duotos homografijos, gali būti perrašyti kaip dvi homogeninės lygtys vektoriui \mathbf{b} :

$$\begin{bmatrix} \mathbf{v}_{12}^T \\ \mathbf{v}_{11} - \mathbf{v}_{22}^T \end{bmatrix} \mathbf{b} = 0 \quad (27)$$

Jei n paveikslų plokštumų modeliai yra nustatyti sukaupiant n kaip lygtyje (27), tada turime

$$\mathbf{V}\mathbf{b} = 0, \quad (28)$$

čia \mathbf{V} yra $2n \times 6$ matrica. Jei $n \geq 3$, turėsime unikalų sprendimą b apibūdinantį į padėties faktorių. Jei $n = 2$, galime priskirti nuožulnumo apribojimus $\gamma = 0$, t. y., $\begin{bmatrix} 1, 0, 0, 0, 0 \end{bmatrix} \underline{b} = 0$, kuris yra gautas iš lygties (28). Jei $n=1$, galima apskaičiuoti tik du vidinius parametrus, pavyzdžiui α ir β , jei u_0 ir v_0 yra žinomi ir $\gamma = 0$. Kaip minėta Tsai židinio ilgio radimas iš vienos plokštumos yra galimas, bet neteisingai sakoma, jog krypties koeficientas negalimas. Sprendimas lygčiai (28) gerai žinomas kaip tikrinis vektorius $V^T V$ susijęs su mažiausia tikrine verte (ekvivalenčiai kaip vektorius V susijęs su mažiausia paprasta verte).

Apskaičiavus \mathbf{b} , galima surasti visus vidinius kameros parametrus. Matrica \mathbf{B} yra apskaičiuota skalės daugikliu, t. y., $\mathbf{B} = \lambda \mathbf{A}^{-T} \mathbf{A}$ su λ pasirinkta skale. Be didelių sunkumų, galima gauti vidinius kameros parametrus iš \mathbf{B} matricos.

$$\begin{aligned} c_y &= B_{12}B_{13} - B_{11}B_{23} / B_{11}B_{22} - B_{12}^2 \\ \lambda &= B_{33} - \left[B_{13}^2 + v_0 B_{12}B_3 - B_{11}B_{23} \right] / B_{11} \\ \alpha &= \sqrt{\lambda / B_{11}} \\ \beta &= \sqrt{\lambda B_{11} / B_{11}B_{22} - B_{12}^2} \\ \gamma &= -B_{12}\alpha^2\beta / \lambda \\ c_x &= \gamma v_0 / \alpha - B_{13}\alpha^2 / \lambda \end{aligned}$$

Apskaičiavus \mathbf{A} , išoriniai parametrai kiekvienam vaizdui yra lengvai apskaičiuojami panaudojant (17), (18) turime: $\mathbf{r}_1 = \lambda \mathbf{A}^{-1} h_1$, $\mathbf{r}_2 = \lambda \mathbf{A}^{-1} h_2$, $\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$, $\mathbf{t} = \lambda \mathbf{A}^{-1} h_3$, su $\lambda = 1 / \| \mathbf{A}^{-1} h_1 \| = 1 / \| \mathbf{A}^{-1} h_2 \|$. Žinoma, dėl triukšmų duomenyse, taip apskaičiuota matrica $\mathbf{R} = \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ negali, iš esmės, atitikti posūkiu matricos ypatybių. Geriausiai posūkiu matrica gali būti gauta per pavyzdžiui paprastųjų verčių skaidymą.

1.2.4.2. Labiausiai tikėtinas įvertinimas

Ankščiau paminėtas sprendimas gautas minimizuojant algebrinį atstumą, kuris nėra fiziškai galimas. Galima tai tobulinti per didžiausios tikimybės įvertinimą.

Duoti n paveikslai iš plokštumos modelio ir m taškai ant plokštumos modelio. Tarkime, kad paveikslo taškai yra iškraipyti nepriklausomai arba tapačiai išskleistam triukšmui. Didžiausia tikimybė gali būti apskaičiuota minimizuojant funkciją:

$$\sum_{i=1}^n \sum_{j=1}^m \left\| m_{ij} - \hat{m}(A, R_i, t_i, M_j) \right\|^2, \quad (29)$$

kur $\hat{m}(A, R_i, t_i, M_j)$ yra taško M_j projekcija, vaizde i , atsižvelgiant į (17). Posūkis R yra apibūdinamas pagal trijų parametrų vektorių, pažymėtu r , kuris yra lygiagretus posūkio ašiai ir šis dydis yra lygus posūkio kampui. Matrica \mathbf{R} ir vektorius \mathbf{r} yra susiję Rodrigues formule [5]. Išraiškos (29) minimizavimas yra netiesinio minimizavimo uždavinys, kuris gali būti sprendžiamas Levenberg-Marquardt algoritmu. Jame reikalingas pradinis $\mathbf{A}(R_i, t_i | i = 1..n)$ pasirinkimas, kuris gali būti gautas naudojant metodą, aprašytą aukščiau. Kompiuterinės kameros dažniausiai turi žymius lęšio iškraipymus, ypač radialinius.

2. Kampų detektoriai

Kampų detektoriai turi tenkinti šiuos reikalavimus:

- Visi tikri kampai turi būti surasti.
- Neturi būti aptikta netikrų kampų.
- Kampų taškai turi būti gerai lokalizuoti.
- Kampų detektorius turi būti atsparus triukšmams.
- Kampų detektorius turi būti efektyvus.

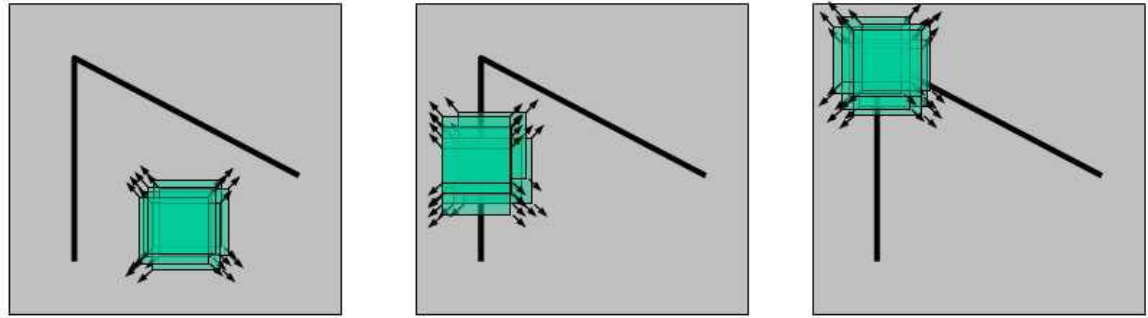
2.1. Harris ir KLT kampų detektoriai

Harris ir KLT (Kanade-Lucas-Tomasi) [7] kampų detektoriai remiasi ta pačia kampų radimo idėja, skiriasi kamputumo radimo funkcija. Pagrindinė jų idėja – esant mažiems lango pozicijos pasikeitimams, ryškus vaizdo pikselių intensyvumo pasikeitimas rodo, kad toje vietoje yra kampo taškas. Šios idėjos iliustracija pateikta 2.1 paveiksle. Pirmasis vaizdas simbolizuoja tolygią erdvę, t.y. stumdant langą visomis kryptimis pikselių intensyvumai praktiškai nesikeičia. Antrasis vaizdas rodo liniją, t.y. stumdant langą išilgai linijos pikselių intensyvumas nesikeičia. Trečiasis paveikslėlis vaizduoja kampą, nes lango judinimas bet kuria kryptimi sąlygoja ryškius pikselių intensyvumo pokyčius, kurie išreiškiami:

$$C(p) = \sum_{x \in W} \left\{ \begin{bmatrix} g_x^2(x) & g_x(x)g_y(x) \\ g_x(x)g_y(x) & g_y^2(x) \end{bmatrix} \times v(x) \right\} = \begin{bmatrix} G_{xx} & G_{xy} \\ G_{yx} & G_{yy} \end{bmatrix}, \quad (30)$$

čia $g_i = \partial_i(g \otimes I) = (\partial_i g) \otimes I$, $i \in (x, y)$, $g = G(x, y; \sigma)$.

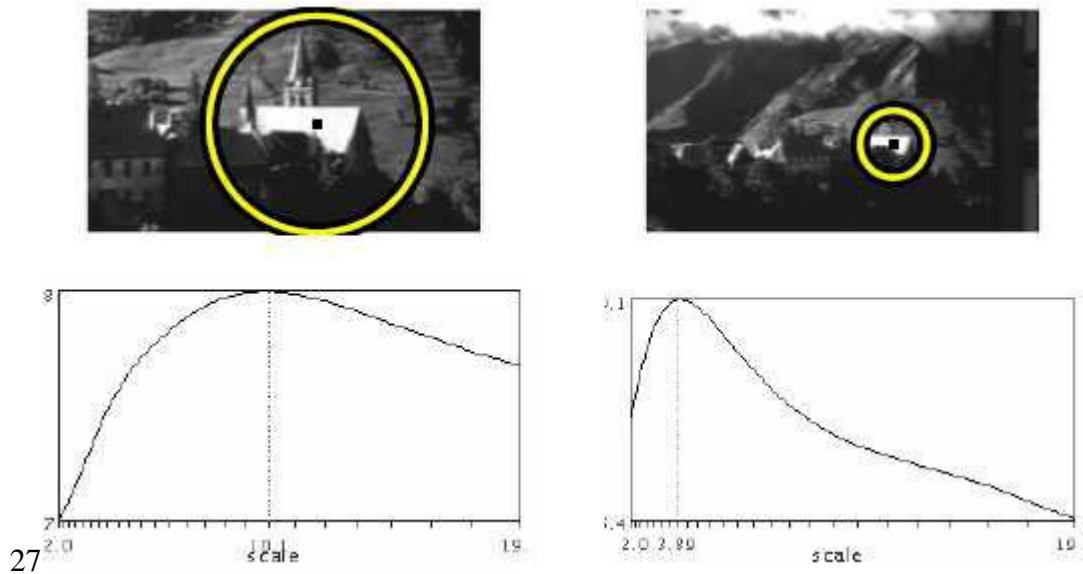
$v(x)$ dažniausiai yra Gauso funkcija, p yra centrinis pikselis, W yra langas, kurio centras yra pikselis p , I yra paveikslas, g yra dvimatė Gauso funkcija, g_x ir g_y yra paveikslo gradientai, randami skaičiuojant sąsūka tarp Gauso pirmos eilės dalinės išvestinės x ir y kryptimis operatoriaus su I .



2.1 pav. Kampo radimo algoritmo idėja. Pasirinkto dydžio langas slenkamas įvairiomis kryptimis. Pirmajame iš kairės vaizde nėra jokių esminių taškų intensyvumo pasikeitimų, todėl nustatoma, kad toje vietoje yra tolygi erdvė. Viduriniajame vaizde intensyvumas nekinta slenkant langą aukštyn ir žemyn – toje vietoje yra linija. Trečiajame vaizde taškų intensyvumas kinta slenkant langą visomis kryptimis, taigi, langas yra ant kampo.

2.1.1. *Harris kampų radimo algoritmas*

Šis algoritmas geba atpažinti pastumtus, pasuktus kampus, jis iš dalies yra atsparus apšvietimo (t.y. taškų intensyvumo) pokyčiams, tačiau labai jautrus objekto didinimui ir mažinimui. Siekdami pašalinti šį trūkumą autoriai naudoja T.Lindeberg pasiūlytą idėją: ieškomas požymis (sudarytas iš antro laipsnio išvestinių) įgis maksimumą, esant tokiam vaizdo didinimui, kuris bus charakteringas bei atspindės tikrąją objekto struktūrą. Vaizdas yra filtruojamas Laplaso (angl. *Laplacian*) filtru ir ieškoma, koks filtro dydis duos maksimalų atsaką ieškomoje padėtyje. Tokiu būdu išrenkamas charakteringas to požymio dydis. 2.2 paveiksle yra parodytas tas pats objektas, matomas skirtingu atstumu. Žemiau pavaizduota kiekvieno iš tų taškų „dydžio“ funkcija, kurios maksimumas padeda nustatyti, kokio dydžio aplinką aplink rastąjį tašką reikėtų naudoti siekiant sutapatinti šiuos du vaizdus.



2.2 pav. Pirmoji eilutė vaizduoja pastatą, nufotografuotą skirtingu atstumu. Antroje eilutėje parodyta Laplaso atsakai pažymėtame taške imant skirtingo didinimo vaizdus. Pirmosios funkcijos maksimumas gaunamas ties didinimu 10,1, antrosios – 3,9; santykis tarp šių dydžių nurodo, kiek kartų pirmasis vaizdas turi būti sumažintas siekiant gauti antrąjį vaizdą. Geltoni apskritimai vaizduoja ypatingų taškų aplinką, kurios spindulys tris kartus didesnis už nustatytą charakteringą dydį.

Kai jau rastas ypatingasis taškas ir jo charakteringas dydis, ieškoma charakteringa elipsės formos sritis aplink šį tašką, t.y. tiriama, koku kampu yra pasisukęs ypatingas taškas. Tada ši sritis normalizuojama į skritulio formos sritį.

Harris-Affine detektorius nustato ypatingo taško koordinatas, jo charakteringą dydį ir posūkį.

Harris kampų detektorius įvertina kiekvieno pikselio kampumą be tikslų tikrinių verčių dekompozicijos:

$$R = |\mathbf{C}| - k \times (\text{trace}(\mathbf{C}))^2. \quad (31)$$

Kol $|\mathbf{C}| = \lambda_1 \times \lambda_2$ ir $\text{trace}(\mathbf{C}) = \lambda_1 + \lambda_2$, čia λ_1 ir λ_2 yra \mathbf{C} tikrinės vertės, tokiu būdu Harris kampų detektorius nereikalauja tikrinių verčių aiškios dekompozicijos. k vertė yra parenkama tarp [0.04, 0.06]. Vaizdo pikselis yra kampas jei abi tikrinės vertės yra didelės, kaip to rezultatas yra gaunamas R pikas.

2.1.2. KLT algoritmas

KLT metodas aiškiai įvertina $\mathbf{C}(p)$ tikrines vertes. Jis pasirenka tuos taškus, kurių minimalios tikrinės vertės įvertintos kaip (32), yra didesnės nei duotas slenkstis:

$$P = \lambda_{\min} = \min(\lambda_1, \lambda_2), \quad (32)$$

$$\lambda_{\min} = \frac{1}{2}(G_{xx} + G_{yy} - \sqrt{(G_{xx} - G_{yy})^2 + 4 \times G_{xy}^2}). \quad (33)$$

Priimama kad $\lambda_1 \geq \lambda_2 \geq 0$. Idealaus pastovaus paveikslėlio: $C_{str} = 0$ ir $\lambda_1 = \lambda_2 = 0$.

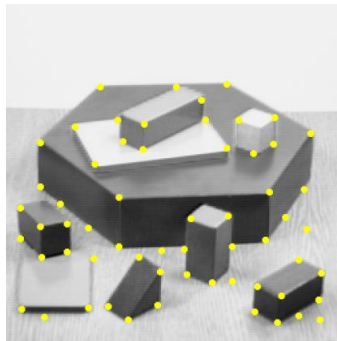
$$C_{str} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}.$$

Idealaus juodai balto vaizdo kraštas: $\lambda_1 > 0$, $\lambda_2 = 0$, kur tikrinis vektorius asocijuojasi su λ_1 ortogonaliam krašte. Juodo kampo kvadratas ant balto pagrindo: $\lambda_1 \geq \lambda_2 > 0$. Tikrinės vertės bus didesnės, taip pat didesnis kontrastas šia kryptimi. Kampas aptinkamas pagal du aiškius kraštus, tai yra kampo lokacija bus ten, kur mažesnės tikrinės vertės, λ_2 , yra pakankamai plačios. KLT kampų detektorius turi du parametrus: slenkstį λ_2 išreikštą per λ_{thr} , ir tiesinį dydį – kvadratinio lango dydį D .

KLT metodo algoritmas [8]:

1. Apskaičiuoti g_x ir g_y iš viso paveikslo $I(x, y)$.
2. Kiekvienam paveikslo taškui p reikia:
 - a) Iš matricos C_{str} per $D \times D$ rasti p kaimynus.
 - b) Apskaičiuoti λ_2 , mažiausias tikrinės C_{str} vertes.
 - c) Jei $\lambda_2 > \lambda_{thr}$, išsaugoti p į sąrašą L .
3. Surikiuoti L mažėjimo tvarka pagal λ_2 .
4. Nuskenuoti sąrašą nuo viršaus į apačią. Kiekvienam tinkamam taškui ištrinti visus arti esančius kaimyninius.

Slenkstis λ_{thr} gali būti apskaičiuotas iš λ_2 histogramos. Nėra paprasta įvertinti lango dydį D . Reikšmės tarp 2 ir 10 yra tinkamos daugumai praktinių pavyzdžių. Tačiau, jeigu keli kampai yra arti vienas kito, jie gali būti neaptikti dėl per didelio lango dydžio D .



2.3 pav. KLT kampų radimo algoritmo pavyzdys

Abu kampų radimo algoritmai gali būti suskirstyti į tris pagrindinius žingsnius: 1) randami gradientai g_x ir g_y skaičiuojant sąsūka tarp vaizdo I ir Gauso išvestinės operatoriaus; 2) kiekvienam vaizdo pikseliui surandama $C(p)$ matrica ir įvertinamas kamputumas P ; 3) naudojamas greitas klasifikavimas, kad būtų pašalinti mažesni atsaką duodantys taškai. Algoritmas gali būti supaprastintas dviem būdais. Pirma, integralinis vaizdas yra naudojamas sumažinti sąsūkos ir kamputumo įvertinimo rezultatų netikslumus. Antra, naudojamas supaprastintas taškų grupavimas.

2.2. Harris-Stephens ir Noble kampų detektorius

Autoriai [9] nusprendė, kad kampų detektoriuose geri kampai bus tie, kai jų srities pikselių skirtumai tarp kaimynų ir to paties kaimyno paslinkto varianto turi aiškų minimumą. Ši idėja remiasi skaičiavimo stabilumo samprata, kuris yra sąlygų teorijos [10] pagrindas. Harris ir Stephanas pasiūlė funkciją, su kuria gali būti išmatuotas kampo stiprumas (tikrumas):

$$M_H(q, \sigma_I, \sigma_D, L) = \det(A^T W A) - \alpha (\text{trace}(A^T W A))^2. \quad (34)$$

Didesnė α vertė reiškia mažesni detektoriaus jautrumą į suformuotą kampą. Ryškesnis šio kampų detektoriaus trūkumas yra parametro α parinkimas, ką reikia padaryti rankiniu būdu. Kad apeiti šią kliūtį, Noble [11] pasiūlė modifikuota Harris funkcijos versiją, kuriai nereikia naudoti nežinomų konstantų:

$$M_N(q, \sigma_I, \sigma_D, L) = \frac{\det(A^T W A)}{\text{trace}(A^T W A) + \varepsilon}. \quad (35)$$

Maža konstanta ε naudojama, norint išvengti dalybos iš nulio, jeigu autokoreliacijos matricos reikšmė būtų nulis.

Šiame algoritme dėmesys sutelktas posūkiams ir transliacijai, nuo to daugiau kompleksinių transformacijų reikalauja automatinų procedūrų, kad aptikti kaimynus, kurie kinta kovariaciškai dėl transformacijos. Rezultatai rodo, kad Harris, Stephens ir Nobel kampų detektorius, kaip ir savybių detektorius, pasiūlytas Shi ir Tomasi, abu gerai išrenka taškus tinkamus transliacijos skaičiavimams tarp paveikslų porų.

2.3. Ryškumo gradiento radimas

Aptarsime pikselių ryškumo gradiento radimo metodą [12]. Pasirenkame paveiksluko sritį aplink pikselį, kurioje bus skaičiuojamas gradientas. Sritis gali būti bet kokios formos, tačiau patogiausia pasirinkti kvadratinę sritį, kurioje analizuojamas pikselis būtų srities centre. Pavyzdžiui pasirenkame stačiakampę sritį 5x5 dydžio (2.4 pav.). Srityje turime 25 pikselius. Jų x ir y koordinatės kinta nuo -2 iki 2 žingsniu 1. Visų pikselių x koordinatės sudaro 5x5 dydžio matricą **X**, o y koordinatės – tokio pat dydžio matricą **Y**. Pikselių ryškumo vertės $I(x, y)$ surašomos į 5x5 dydžio **I** matricą. **I** matricos reikšmės gali būti aproksimuotos polinominėmis lygtimis naudojant **X** ir **Y** matricose surašytas x ir y koordinačių reikšmes. Gali būti naudojami pirmos, antros ir trečios eilės polinomai:

$$I(x, y) = c_0 + c_1x + c_2y, \quad (36)$$

$$I(x, y) = c_0 + c_1x + c_2y + c_3x^2 + c_4xy + c_5y^2, \quad (37)$$

$$I(x, y) = c_0 + c_1x + c_2y + c_3x^2 + c_4xy + c_5y^2 + c_6x^3 + c_7x^2y + c_8xy^2 + c_9y^3. \quad (38)$$

1 (-2, -2)	6 (-1, -2)	11 (0, -2)	16 (1, -2)	21 (2, -2)
2 (-2, -1)	7 (-1, -1)	12 (0, -1)	17 (1, -1)	22 (2, -1)
3 (-2, 0)	8 (-1, 0)	13 (0, 0)	18 (1, 0)	23 (2, 0)
4 (-2, 1)	9 (-1, 1)	14 (0, 1)	19 (1, 1)	24 (2, 1)
5 (-1, 2)	10 (-1, 2)	15 (0, 2)	20 (1, 2)	25 (2, 2)

2.4 pav. Pasirinkta 5x5 dydžio sritis. Numeracija pradedama nuo viršutinio kairio kampo, kurio koordinatės (-2, -2). Centrinis yra 13 pikselis, kurio koordinatės (0, 0)

Įvairių aproksimacijos lygčių koeficientai gali būti surasti mažiausių kvadratų metodu. Tuo tikslu matricas **X**, **Y** keičiame į **x** ir **y** vektorius. Kiekvienam pikseliui gali būti parinkti nuokrypių svoriai w_i . Iš jų sudarome svorių vektorių **w**. Mažiausių kvadratų metodo lygčių matricai sudaryti naudojamos svorių ir koordinačių vektorių skaliarinės sandaugos. Pavyzdžiui, aproksimuojant pirmo laipsnio polinomu sudaroma pagalbinė matrica

$$A_1 = \begin{pmatrix} \vec{1} * \vec{w} & \vec{x} * \vec{w} & \vec{y} * \vec{w} \end{pmatrix}, \quad (39)$$

čia $\vec{1}$ yra vienetų vektorius, * yra skaliarinė sandauga. Tiesinės lygčių sistemos, iš kurios surandame aproksimavimo koeficientus c_i , matrica A surandama:

$$A = A_1^T A_1. \quad (40)$$

Panašiai surandame lygčių sistemos laisvųjų narių vektorių:

$$\vec{B} = A_1^T (\vec{w}I). \quad (41)$$

Todėl aproksimavimo koeficientų skaičiavimui galime sudaryti tiesinę lygčių sistemą:

$$A \vec{c} = \vec{B}, \quad (42)$$

čia \vec{c} yra nežinomų koeficientų c_i vektorius. (42) lygtį galime išspręsti analitiškai. Spręsdami lygtis surandame C_0, C_1, C_2 matricas koeficientų c_0, c_1, c_2 apskaičiavimui iš pikselių ryškumo matricos I .

Jei visų pikselių nuokrypiams pasirenkami lygūs svoriai, tai gauname matricas:

$$C_0 = \frac{1}{25} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$C_1 = \frac{1}{30} \begin{pmatrix} -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \\ -2 & -1 & 0 & 1 & 2 \end{pmatrix}$$

$$C_2 = \frac{1}{30} \begin{pmatrix} -2 & -2 & -2 & -2 & -2 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 \end{pmatrix}$$

Tada c_i koeficientai gali būti suskaičiuoti lygtimi:

$$c_i = \sum_{j=1}^5 \sum_{k=1}^5 C_i(j, k) I(j, k) \quad (43)$$

Toks metodas leidžia pagrįsti įvairius kraštų radimo metodus. Jei paimtume sritį 3x3 ir svorius lygius 1, mes gautume Prewitt kraštų radimo metodą. Jei pasirinktume didesnius svorius taškuose (-1, 0), (1, 0), (0,-1), (0, 1) (didesni svoriai taškams, kurie yra horizontalioje ar vertikalioje ašyje) gautume Sobelio kraštų radimo operatorių.

Dalinė išvestinė pagal x taške $(0, 0)$:

$$\frac{\partial I}{\partial x} = c_1. \quad (44)$$

Dalinė išvestinė pagal y taške $(0, 0)$:

$$\frac{\partial I}{\partial y} = c_2. \quad (45)$$

Taigi gradientas gali būti išreikštas lygtimi:

$$\text{grad} I(x, y)_{(0,0)} = c_1 \vec{i} + c_2 \vec{j}. \quad (46)$$

Paskutinė lygtis gali būti naudojama su visų tipu polinomais taškui su koordinatėmis $(0, 0)$.

Gradianto kryptis (kampas φ) apskaičiuojama:

$$\varphi = \text{arctg} \frac{c_2}{c_1}. \quad (47)$$

3. Tapačių taškų vaizduose radimas

3.1. Lucas-Kanade algoritmas

Algoritmo [13] pagrindinė idėja yra tokia - viename vaizde pasirenkamas dvimatis šablonas $\mathbf{T}(\mathbf{u})$ ir surandama jo vieta kitame dvimačiame vaizde $\mathbf{I}(\mathbf{u})$, čia $\mathbf{u}=(u,v)^T$ yra dvikoordinatis vektorius, atitinkantis pikselio koordinatas. Panaudojama koordinačių transformacijos funkcija $\mathbf{W}(\mathbf{u};\mathbf{p})$, kurioje $\mathbf{p}=(p_1,\dots,p_n)^T$ yra parametrų vektorius. Funkcija $\mathbf{W}(\mathbf{u};\mathbf{p})$ paima pikselio koordinatas \mathbf{u} , kurios atitinka \mathbf{T} šabloną ir jas transformuoja subpikseliniu tikslumu į koordinatas $\mathbf{W}(\mathbf{u};\mathbf{p})$ paveiksle \mathbf{I} . Kai Lucas-Kanade algoritmas yra naudojamas apskaičiuoti optinį srautą, tada $\mathbf{W}(\mathbf{u};\mathbf{p})$ aprašo koordinačių transliaciją. Vektoriaus \mathbf{p} parametrų skaičius n gali būti įvairaus ilgio.

Matematinis Lucas-Kanade pagrindas yra minimizuoti paklaidų kvadratų sumą tarp šablono \mathbf{T} ir jį atitinkančių vaizdo \mathbf{I} pikselių šablono koordinačių ribose:

$$\sum_{\mathbf{u}} [\mathbf{I} \mathbf{W} \mathbf{u};\mathbf{p} - \mathbf{T} \mathbf{u}]^2, \quad (48)$$

vektoriaus \mathbf{p} atžvilgiu. Norint apskaičiuoti $\mathbf{I}(\mathbf{W}(\mathbf{u};\mathbf{p}))$, reikia interpoliuoti vaizdo \mathbf{I} pikselius subpikseliniu tikslumu koordinatėse $\mathbf{W}(\mathbf{u};\mathbf{p})$. Minimizavimas (48) lygtimi aprašytos išraiškos yra atliekamas \mathbf{p} vektoriaus atžvilgiu ir sumavimas yra atliekamas su visomis šablono $\mathbf{T}(\mathbf{u})$ pikselių koordinatėmis \mathbf{u} . Išraiškos minimizavimas yra netiesinis optimizavimo uždavinys net ir tuo atveju, jeigu transformacijos funkcija $\mathbf{W}(\mathbf{u};\mathbf{p})$ yra tiesinė išraiška vektoriaus \mathbf{p} atžvilgiu, nes pikselių vertės $\mathbf{I}(\mathbf{u})$ netiesiškai priklauso nuo vektoriaus \mathbf{u} . Iš tikrųjų, pikselių vertės $\mathbf{I}(\mathbf{u})$ yra nesusijusios su pikselio koordinatėmis \mathbf{u} . Optimizuojant (48) lygties išraišką, Lucas-Kanade algoritmas priima, kad yra žinomas apytikris \mathbf{p} vektoriaus įvertinimas \mathbf{p}_i ir tada iteraciniu būdu jis yra tikslinamas. Parametrų vektorių \mathbf{p} galima išreikšti per pokytį $\Delta\mathbf{p}$. ; Todėl minimizuojama ši išraiška:

$$\sum_{\mathbf{u}} [\mathbf{I} \mathbf{W} \mathbf{u};\mathbf{p}_i + \Delta\mathbf{p} - \mathbf{T} \mathbf{u}]^2, \quad (49)$$

atžvilgiu $\Delta\mathbf{p}$. Tuomet nauja \mathbf{p} vektoriaus vertė:

$$\mathbf{p}_{i+1} = \mathbf{p}_i + \Delta\mathbf{p}. \quad (50)$$

Šie iteraciniai žingsniai yra kartojami tol, kol apskaičiuojamas parametrų vektorius \mathbf{p} norimu tikslumu arba tenkinamos kitos iteracinio ciklo nutraukimo sąlygos.

Lucas-Kanade algoritme gali būti panaudotas Gauss-Newton gradientinio nusileidimo netiesinis optimizavimo algoritmas. Netiesinė (49) lygties išraiška yra padaroma tiesine, panaudojant $\mathbf{I}(\mathbf{W}(\mathbf{u};\mathbf{p}_i+\Delta\mathbf{p}))$ skleidimą pirmos eilės Teiloro eilute:

$$\sum_u \left[\mathbf{I} \mathbf{W} \mathbf{u};\mathbf{p}_i + \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - \mathbf{T} \mathbf{u} \right]^2, \quad (51)$$

šioje išraiškoje, $\nabla \mathbf{I} = \left(\frac{\partial I}{\partial u}, \frac{\partial I}{\partial v} \right)$ yra \mathbf{I} paveikslo gradientas įvertintas taške $\mathbf{W}(\mathbf{u};\mathbf{p}_i)$. Narys (matrica) $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ yra vadinamas Jakobianu.

Lygtimi (51) yra išreikštas mažiausių kvadratų uždavinys. Jį sprendžiant ieškomos dalinės išvestinės $\Delta \mathbf{p}$ atžvilgiu:

$$\sum_u \left[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[\mathbf{I} \mathbf{W} \mathbf{u};\mathbf{p} + \nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - \mathbf{T} \mathbf{u} \right], \quad (52)$$

Prilyginant šias išraiškas nuliui, galima išs্পęsti gautą lygčių sistemą $\Delta \mathbf{p}$ atžvilgiu. Gaunamas lygčių sistemos sprendinys turi formą:

$$\Delta \mathbf{p} = -\mathbf{H}_{\text{LK}}^{-1} \sum_u \left[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[\mathbf{I} \mathbf{W} \mathbf{u};\mathbf{p} - \mathbf{T} \mathbf{u} \right], \quad (53)$$

čia H_{LK} yra $n \times n$ dydžio Hessianės matricos aproksimacija:

$$\mathbf{H}_{\text{LK}} = \sum_u \left[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[\nabla \mathbf{I} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]. \quad (54)$$

Lucas-Kanade algoritmas yra sudarytas iš iteracijų, kartojant lygtis (53) ir (50). Dėl to, kad gradientas $\nabla \mathbf{I}$ turi būti įvertintas, panaudojant $\mathbf{W}(\mathbf{u};\mathbf{p})$, ir Jakobianas $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ skaičiuojamas, panaudojant \mathbf{p} , abi dedamosios priklauso nuo \mathbf{p} . Taipogi Hessianės matrica turi būti perskaičiuota kiekvienoje iteracijoje, nes parametras \mathbf{p} kinta po kiekvienos iteracijos. Iteraciją nulemia Hessianės matricos skaičiavimas pagal lygtį (54). Iteracijos trukmė proporcinga $O(n^2N)$, čia N yra pikseliu skaičius šablone.

3.2. Kalibruotų stereo vaizdų išlyginimas Bouguet metodu

Atlikus erdvinės regos sistemos kamerų kalibravimą, gaunama kamerų tarpusavio posūkių matrica \mathbf{R} ir translacijos vektorius \mathbf{T} . Bouguet stereo išlyginimo algoritmas [14] paprasčiausiai stengiasi sumažinti pasikeitimus, gaunamus vaizdų reprojekcijos metu, kai tuo tarpu vaizdų plotą stengiamasi palikti kuo didesnę. Stengiantis minimizuoti vaizdų reprojekcijos iškraipymus, posūkių matrica \mathbf{R} , kuri aprašo dešinėsios kameros vaizdo plokštumos orientaciją kairiosios kameros vaizdo plokštumos atžvilgiu, yra suskaidoma į dvi lygių posūkių matricas \mathbf{r}_l ir \mathbf{r}_r . \mathbf{r}_l aprašo kairiosios kameros posūkį, o \mathbf{r}_r – dešinėsios kameros posūkį. Toks kamerų pasukimas į priešingas puses išdėsto jų vaizdų plokštumas lygiagrečiai, bet dar nėra atliekamas vaizdo išlyginimas. Surasime matricą \mathbf{R}_{rect} , kuri kairės kameros epipolių perkels į begalybę ir orientuos epipolines linijas horizontaliai. Tai padarysime pirmiausiai įvertindami epipoliaus kryptį \mathbf{e}_1 . Paimdami pagrindinį tašką su koordinatėmis (c_x, c_y) kaip kairio vaizdo pradžią, epipoliaus krypties vektorių nukreipsime išilgai translacijos vektoriaus \mathbf{T} tarp dviejų kamerų projekcijos centrų :

$$\mathbf{e}_1 = \frac{\mathbf{T}}{\|\mathbf{T}\|}. \quad (55)$$

Kitos kameros epipoliaus vektorius \mathbf{e}_2 turi būti ortogonalus \mathbf{e}_1 , bet neturi kitų apribojimų. Pasirinksime \mathbf{e}_2 kryptį ortogonaliai bazinei linijai (kuri bus lygiagreti vaizdo plokštumai). Tai bus užrašyta kaip vektoriaus \mathbf{e}_1 ir bazinės linijos vektorinė sandauga, normuojant gauto vektoriaus ilgį į vienetinį:

$$\mathbf{e}_2 = \frac{\begin{bmatrix} -T_y T_x & 0 \end{bmatrix}^T}{\sqrt{T_x^2 + T_y^2}}. \quad (56)$$

Trečias vektorius turi būti ortogonalus ir \mathbf{e}_1 , ir \mathbf{e}_2 . Jis gali būti randamas kaip pirmųjų dviejų vektorinė sandauga:

$$\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2. \quad (57)$$

Matrica, kuri kairiosios kameros epipolių perkels į begalybę, bus lygi:

$$\mathbf{R}_{rect} = \begin{bmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \mathbf{e}_3^T \end{bmatrix}.$$

Ši matrica atlieka menamą kairiosios kameros sukimą apie jos projekcijos centrą, kol epipolinės linijos tampa horizontaliomis ir begalinėmis. Abejomis kameromis gautų vaizdų linijų išlyginimas yra gaunamas panaudojant matricas:

$$\mathbf{R}_l = \mathbf{R}_{\text{rect}} \mathbf{\Gamma}_l; \quad (58)$$

$$\mathbf{R}_r = \mathbf{R}_{\text{rect}} \mathbf{r}_r. \quad (59)$$

Taip pat apskaičiuojame išlygintų kairiosios ir dešinėsios kamerų matricas $\mathbf{M}_{\text{rect}_l}$ ir $\mathbf{M}_{\text{rect}_r}$, ir sukombinuojame jas su projekcijų matricomis, gaudami \mathbf{P}_l ir \mathbf{P}_r :

$$\mathbf{P}_l = \mathbf{M}_{\text{rect}_l} \mathbf{P}'_l = \begin{bmatrix} f_{x_l} & \alpha_l & c_{x_l} \\ 0 & f_{y_l} & c_{y_l} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

ir

$$\mathbf{P}_r = \mathbf{M}_{\text{rect}_r} \mathbf{P}'_r = \begin{bmatrix} f_{x_r} & \alpha_r & c_{x_r} \\ 0 & f_{y_r} & c_{y_r} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

(čia α_l ir α_r yra pikselio išstrižumo koeficientas, kuris šiuolaikinėse kamerose dažniausiai yra 0). Projekcijos matricos erdviųjų taškų homogeninius vektorius paverčia į dvimačių taškų homogeninius vektorius:

$$\mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ w \end{bmatrix}.$$

Vaizdo pikselio koordinatės surandamos kaip $(x/w, y/w)$. Dvimačiai taškai taip pat gali būti suprojektuoti į trimatę erdvę, turint jų plokštumines koordinates ir kamerų vidinę matricą:

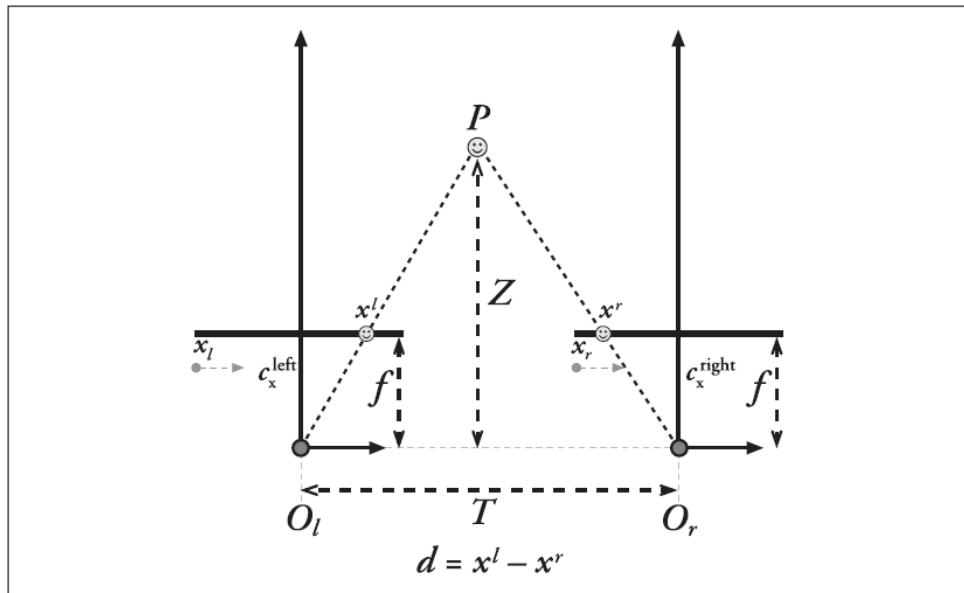
$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -1/T_x & (c_x - c'_x)/T_x \end{bmatrix}.$$

Joje yra kairiosios kameros parametrai, išskyrus c'_x , kuris yra pagrindinio taško x koordinatė dešiniajame vaizde. Jei bazinės linijos susikerta begalybėje, tada $c_x = c'_x$ ir narys apatiniame dešiniajame kampe yra 0. Iš dviejų duotų dvimačių homogeninių taškų ir jų neatitikimo (*angl. disparity*) d , galima taškus suprojektuoti į trimatę erdvę panaudojant lygtį:

$$\mathbf{Q} \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix}.$$

Erdvinio taško koordinatės tada yra $(X/W, Y/W, Z/W)$.

Bouguet išlyginimo metodas atkuria idealią stereo konfigūraciją, kaip tai pavaizduota 3.2.1 paveiksle. Naujų vaizdų centrai ir naujų vaizdų ribos yra parenkamos, kad būtų maksimizuojamos abiejų vaizdų sutampančios sritys. Praktiškai nustatomi vienodi kamerų centrai ir maksimalus dviejų vaizdų aukštis ir plotis kaip parametrai naujų stereo vaizdų.



3.1 pav. Ideali stereo sistema (gali būti gaunama vaizdų išlyginimo metodu). Gylis Z gali būti rastas trikampio metodu. Pagrindinės vaizdo kamerų ašys prasideda nuo projekcijų centrų O_l ir O_r .

3.3. Tapačių taškų radimo algoritmų palyginimas

Autoriai Daniel Scharstein ir Richard Szeliski [15] palygino 107 tapačių taškų paieškos algoritmus. Buvo pasirinktos keturios skirtingos paveikslų poros: Tsukuba, Venus, Teddy, Cones, pateiktos priede nr.1. Su šiomis paveikslų poromis buvo išbandyti įvairūs algoritmai. Gautus rezultatus palygino pagal rastų blogų pikselių tarp dviejų paveikslų kiekį išreikštą procentais. Dalis gautų rezultatų pateikta 1 lentelėje.

1 lentelė. Tapačių taškų radimo algoritmų palyginimas

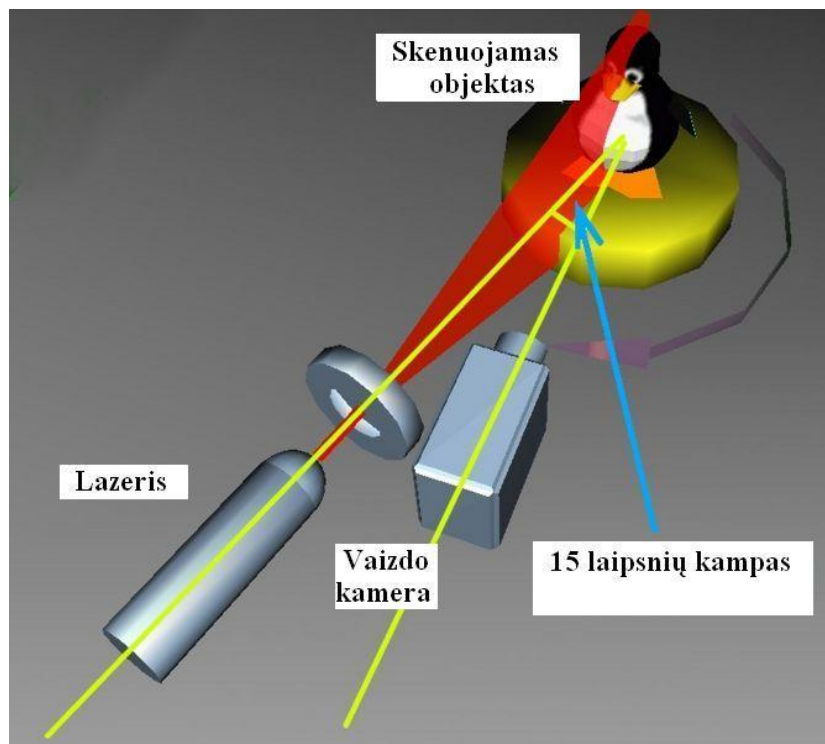
Algoritmas (Paklaidos slenkstis = 1)	Vidurkiai	Tsukuba paveikslų pora			Venus paveikslų pora			Teddy paveikslų pora			Cones paveikslų pora			Biųjų pikselių vidurkiai procentais
		nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonoc	all	disc	
ADCensus [20]	5.8	<u>1.07</u> 12	1.48 10	5.73 14	<u>0.09</u> 2	0.25 7	1.15 2	<u>4.10</u> 4	6.22 3	10.9 4	<u>2.42</u> 3	7.25 5	6.95 4	3.97
SurfaceStereo [21]	15.2	<u>1.28</u> 28	1.65 17	6.78 32	<u>0.19</u> 17	0.28 10	2.61 26	3.12 1	5.10 1	8.65 1	<u>2.89</u> 16	7.95 11	8.26 22	4.06
Undr+OvrSeg [22]	22.1	<u>1.89</u> 51	2.22 44	7.22 41	<u>0.11</u> 5	0.22 5	1.34 3	<u>6.51</u> 26	9.98 17	16.4 29	<u>2.92</u> 18	8.00 12	7.90 14	5.39
SO+borders [23]	32.0	<u>1.29</u> 30	1.71 21	6.83 34	<u>0.25</u> 27	0.53 23	2.26 18	<u>7.02</u> 39	12.2 35	16.3 27	<u>3.90</u> 42	9.85 44	10.2 44	6.03
eSGM [24]	44.5	<u>3.25</u> 76	4.22 70	14.0 77	<u>0.60</u> 48	1.31 56	8.22 65	<u>5.38</u> 15	13.0 38	15.6 19	<u>2.87</u> 15	8.95 28	8.57 27	7.16
CCH+SegAggr [25]	57.7	<u>1.74</u> 48	2.11 40	9.23 55	<u>0.41</u> 35	0.94 44	3.97 41	<u>8.08</u> 57	14.3 59	19.8 66	<u>7.07</u> 87	12.9 74	16.3 86	8.07
RealtimeVar [26]	64.8	<u>3.33</u> 78	5.48 85	16.8 91	<u>1.15</u> 68	2.35 74	12.8 78	<u>6.18</u> 22	13.1 40	17.3 46	<u>4.66</u> 59	11.7 61	13.7 75	9.05
<u>HRMBIL</u> [27]	73.1	<u>3.60</u> 80	4.79 78	16.8 90	<u>1.38</u> 74	2.72 78	17.3 87	<u>7.48</u> 47	15.8 75	20.8 75	<u>4.29</u> 53	13.7 78	12.0 62	10.0
<u>BP+MLH</u> [28]	82.7	<u>4.17</u> 89	6.34 95	14.6 81	<u>1.96</u> 85	3.31 86	16.8 85	<u>10.2</u> 79	18.9 88	24.0 87	<u>4.93</u> 64	15.5 87	12.3 66	11.1
SAD- IGMCT [29]	85.6	<u>5.81</u> 102	7.14 100	22.6 104	<u>2.61</u> 88	3.33 87	25.3 99	<u>9.79</u> 75	15.5 73	25.7 91	<u>5.08</u> 67	11.5 59	15.0 82	12.5
DP [30]	91.0	<u>4.12</u> 88	5.04 81	12.0 70	<u>10.1</u> 105	11.0 105	21.0 94	<u>14.0</u> 93	21.6 93	20.6 73	<u>10.5</u> 98	19.1 97	21.1 95	14.2
DPVI [31]	91.2	<u>4.76</u> 92	5.83 89	16.6 89	<u>4.89</u> 97	5.66 97	22.9 96	<u>11.0</u> 87	16.2 79	23.4 84	<u>9.64</u> 93	15.6 89	23.5 102	13.3
PhaseBased [32]	95.6	<u>4.26</u> 90	6.53 96	15.4 86	<u>6.71</u> 99	8.16 99	26.4 101	<u>14.5</u> 94	23.1 94	25.5 90	<u>10.8</u> 100	20.5 102	21.2 96	15.3
RegionalSup [33]	96.3	<u>3.99</u> 86	6.05 91	14.2 78	<u>8.14</u> 101	9.68 102	36.8 106	<u>18.3</u> 104	26.7 104	32.1 100	<u>9.16</u> 92	19.3 98	19.9 93	17.0
<u>SO</u> [34]	98.6	<u>5.08</u> 97	7.22 101	12.2 71	<u>9.44</u> 104	10.9 104	21.9 95	<u>19.9</u> 105	28.2 107	26.3 93	<u>13.0</u> 103	22.8 105	22.3 98	16.6
MI-nonpara [35]	100.8	<u>5.59</u> 101	7.54 103	18.8 96	<u>7.50</u> 100	8.99 100	35.0 104	<u>17.4</u> 101	25.7 103	36.9 104	<u>10.2</u> 97	19.9 100	22.6 100	18.0
LCDM+AdaptWgt [36]	102.1	<u>5.98</u> 104	7.84 104	22.2 103	<u>14.5</u> 107	15.4 107	35.9 105	<u>20.8</u> 107	27.3 105	38.3 106	<u>8.90</u> 91	17.2 92	20.0 94	19.5
Infection [37]	103.3	<u>7.95</u> 107	9.54 106	28.9 107	<u>4.41</u> 96	5.53 96	31.7 102	<u>17.7</u> 102	25.1 102	44.4 107	<u>14.3</u> 105	21.3 103	38.0 107	20.7

4. Trimačio vaizdo skeneriai

Vieni naujausių įrenginių, galintys nuskenuoti trimatį objektą, o jo erdvinį vaizdą perkelti į kompiuterį, yra 3D lazeriniai skeneriai (4.1 paveikslas). Jų veikimo principas [44] (4.2 paveikslas) yra paremtas tuo, kad lazerio spindulys eina per lęšį, kuris sugeneruoja vertikalią lazerio liniją. Objektas yra sukamas ratu, kol lazerio linija neapšviečia viso objekto nelygių paviršių. Gautas vaizdas atrodo kaip vingiuojanti lazerio linija ir yra analizuojama kompiuterio, kad apskaičiuoti kiekvieną tašką ant objekto nuo apsisukimo centro. Šis taškų „debesis“ suformuojamas kaip tinklas trimatėje erdvėje ir užpildomas tekstūra. Gautas vaizdas atvaizduojamas kompiuterinėse programose.



4.1 pav. 3D lazerinis skeneris



4.2 pav. 3D lazerinio skenerio veikimo principas

Pagrindinės šių lazerinių skenerių charakteristikos yra: objekto paviršiaus nustatymo netikslumų dydis, maksimalus skenavimo atstumas, mažiausias galimas nuskenuoti taškas, skenavimo plotas ir taškų skenavimo per sekundę skaičius. Kelis jų apžvelgsime 2 lentelėje.

2 lentelė. Lazerinių skenerių charakteristikų palyginimas

Skenerio pavadinimas	Skenavimo tikslumas, mikronais	Skenavimo greitis	Skenavimo greitis, taškų skaičius per sekundę	Maksimalus vieno taško gylis plokštumoje
ZScanner® 600	80		18,000	30 cm
ZScanner® 800	40		25,000	30 cm
LaserProbe 4500	30	100x100mm plotą, 6-10 minučių	4,500	10 cm
SCAN 1000 PRO	50			3 cm
Aktivity 101 (Protezų skeneris)	20	vieno danties protezui 1min, visam žandikaulio modeliui 3min		
Mephisto EX	50	0.3-1.5 sek.		
Mephisto CX	150	0.3 - 1.5 sek.		
Mephisto EOSScan	10-200	20 sek.		
PicoScan	100	20 sek.		
LPX-1200DS	50	Stalo sukimosi greitis: 9 apm, galvos sukimosi greitis: 4.48 apm, maksimalus galvos judėjimo greitis 7.58 mm/sek.		

5. Eksperimentinis tyrimas

Bandymai buvo atliekami *DevCpp* bei *MatLab* programiniais paketais. Programos *DevCpp* kodas buvo kuriamas pasinaudojant standartinėmis *OpenCv* bibliotekos funkcijomis. *MatLab* programai buvo naudojamas kalibravimo įrankių rinkinys „*toolbox_calib*“, bei parašytas kodas apskaičiuoti 3D taškus.

Pirmiausia dviejų kamerų *Canon (EOS400D)* optinės ašys nustatytos lygiagrečiai, o atstumas tarp lęšių centrų turimoje sistemoje nustatytas pats mažiausias, t.y. 11,5 cm. Stereo sistemos kalibravimui buvo fotografuojama šachmatų lenta. Naudojant programą su *OpenCv* funkcijomis kalibravimui pakako po vieną kadrą iš kiekvienos kameros. Naudojant *MatLab* programą kokybiškesnis kalibravimas būdavo dar papildomai nufotografavus šachmatų lentą įvairiais kampais. Po kalibravimo buvo fotografuojami įvairių formų ir paviršių objektai. Buvo bandoma įvairiai nustatinėti fokusavimą, vyzdinę diafragma. Nustatyta jog jautresnis fokusavimui kodas buvo parašytas su *OpenCv* funkcijomis. Blogiau sufokusuotų vaizdų šis kodas nesugebėdavo sukalibruoti, kuomet su *MatLab* programa kalibravimas įvykdavo.

5.1. Kodas parašytas OpenCv bibliotekos funkcijomis

Vaizdo kamerų kalibravimas atliekamas naudojantis dviejų etapų fotogrametriniu kalibravimo metodu (žiūrėti 2.1 skyrių). Kadangi *OpenCV* [14] yra atvirojo kodo kompiuterinės regos biblioteka, tai tikslinga pasinaudoti jos funkcijomis, kuriant programos kodą.

Kalibravimo šablono („šachmatų lentos“) objekto vaizde (5.1 pav.) informaciniai taškai randami funkcijos *cvFindChessboardCorners* pagalba. Siekiant kuo mažesnės kalibravimo parametru neapibrėžties, šio kalibravimo metodo realizacijai autoriai rekomenduoja naudoti ne mažiau 15 kalibravimo šablono objekto vaizdų vidinių parametru apskaičiavimams. Kalibravimo šablonas turėtų būti sudarytas iš ne mažesnės nei 6×6 kalibravimo taškų matricos (5.1 pav.).

Abiejų vaizdo kamerų išoriniai parametrai (posūkio ir poslinkio vektoriai) randami funkcijos *cvStereoCalibrate* pagalba. Praktikoje neretai tenka atlikti vien tik išorinių parametru kalibravimą, pavyzdžiui, kai perstatomos jau sukalibruotos vaizdo kameros į kitą poziciją, pakeičiami atstumai tarp

jų, kryptys ir t.t. Po visų kalibravimo parametrų suradimo, atliekamas abiejų vaizdo kamerų radialinių iškraipymų ištaisymas.

cvStereoRectify funkcija apskaičiuoja posūkio matricas kiekvienai kamerai, kas virtualiai abiejų kamerų paveikslų plokštumas padaro kaip vieną plokštumą. Išlyginimas padaromas Bouguet metodu (žiūrėti 4.2 skyrių). Tada epipolinės linijos gaunamos lygiagrečios, kas supaprastina stereo atitikimo problemą. Funkcija paima matricas apskaičiuotas iš *cvStereoCalibrate*, o gaunami rezultatai yra dvi posūkio matricos ir dvi projekcijų matricos naujose koordinatėse (5.2 pav.).

cvInitUndistortRectifyMap funkcija apskaičiuoja iškraipymų ir išlyginimų transformacijas žinodama vaizdo kameros parametrus ir radialinių iškraipymų koeficientus. Ji apskaičiuoja iškraipymų žemėlapi visam vaizdo kadrai, kuris susieja kiekvieną vaizdo tašką su jo tikrąja (neiškraipyta) pozicija tame vaizde.

cvRemap funkcija, pagal sudarytą iškraipymų žemėlapi, atlieka charakteringų vaizdo taškų pozicijų korekcijas kiekvienam naujam kadrai. Šių funkcijų rezultatas – vaizdai be iškraipymų, gaunami realiuoju laiku, kuriuos toliau galima panaudoti siekiant nustatyti žmogaus rodymo kryptį 3D erdvėje. Baigus kalibravimą apskaičiuoti vidiniai ir išoriniai parametrai išsaugomi tekstinėje rinkmenoje vėlesniam panaudojimui.

Po to ieškoma taškų atitikimo blokų palyginimo metodu ir randami tapačių pikselių koordinatžių abiejuose vaizduose skirtumai. (5.3 pav.). Tai atliekama su *cvFindStereoCorrespondenceBM* funkcija.

Galiausiai turint visus duomenis, tokius kaip pikselių gylių žemėlapi ir reprojeccijos matricą su funkcija *cvReprojectImageTo3D* gaunamas trimatis vaizdas (5.4 pav.). Atitinkamai bandymai atlikti su daugiau kalibravimo šablonų (5.5, 5.6, 5.7 pav.).



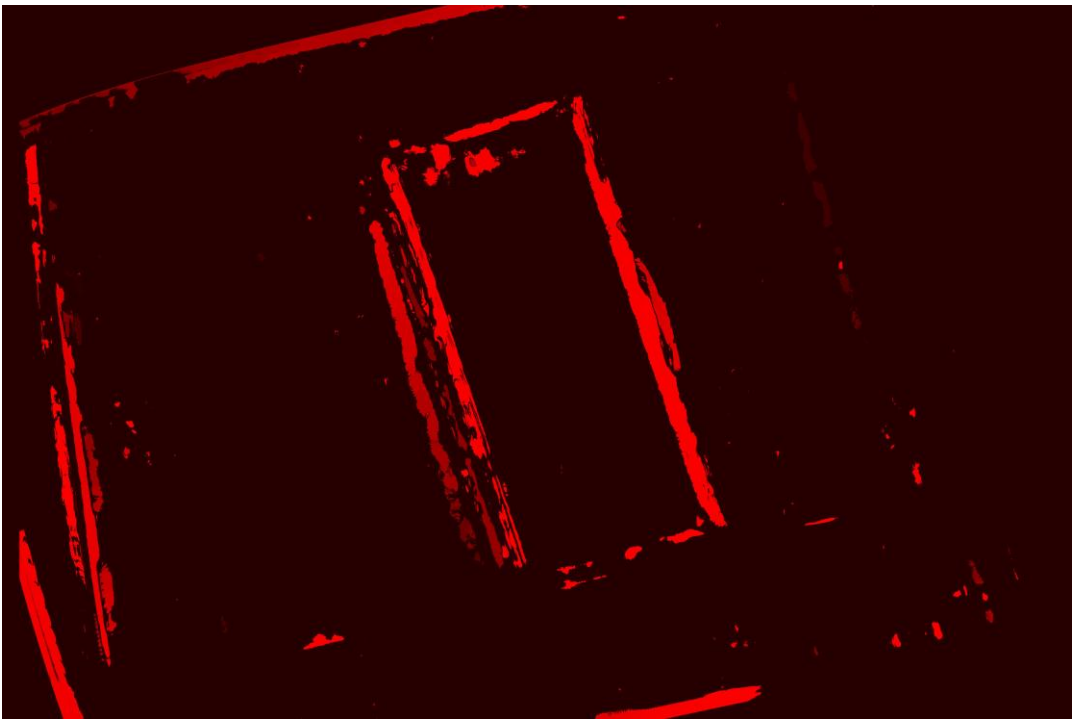
5.1 pav. Kalibravimo šablonas: šachmatų lenta.



5.2 pav. Atliktas paveikslų išlyginimas ir randamos epipolinės linijos



5.3 pav. Gauti pikselių neatitikimai



5.4 pav. Gautas trimatis vaizdas



5.5 pav. Išlyginti stereo paveikslai



5.6 pav. Gauti pikselių neatitikimai

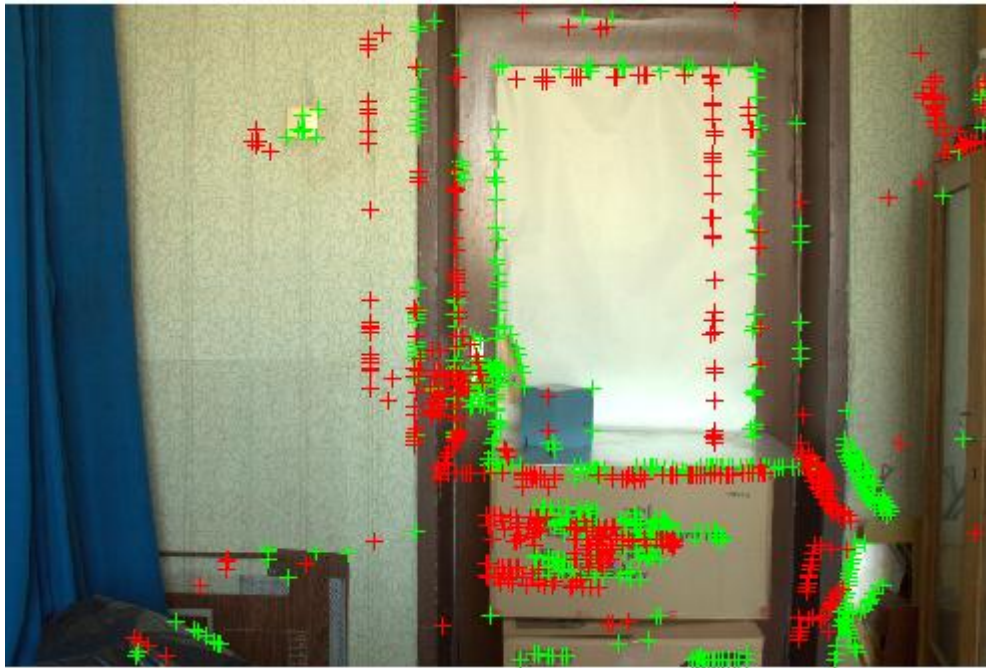


5.7 pav. Gautas trimatis vaizdas

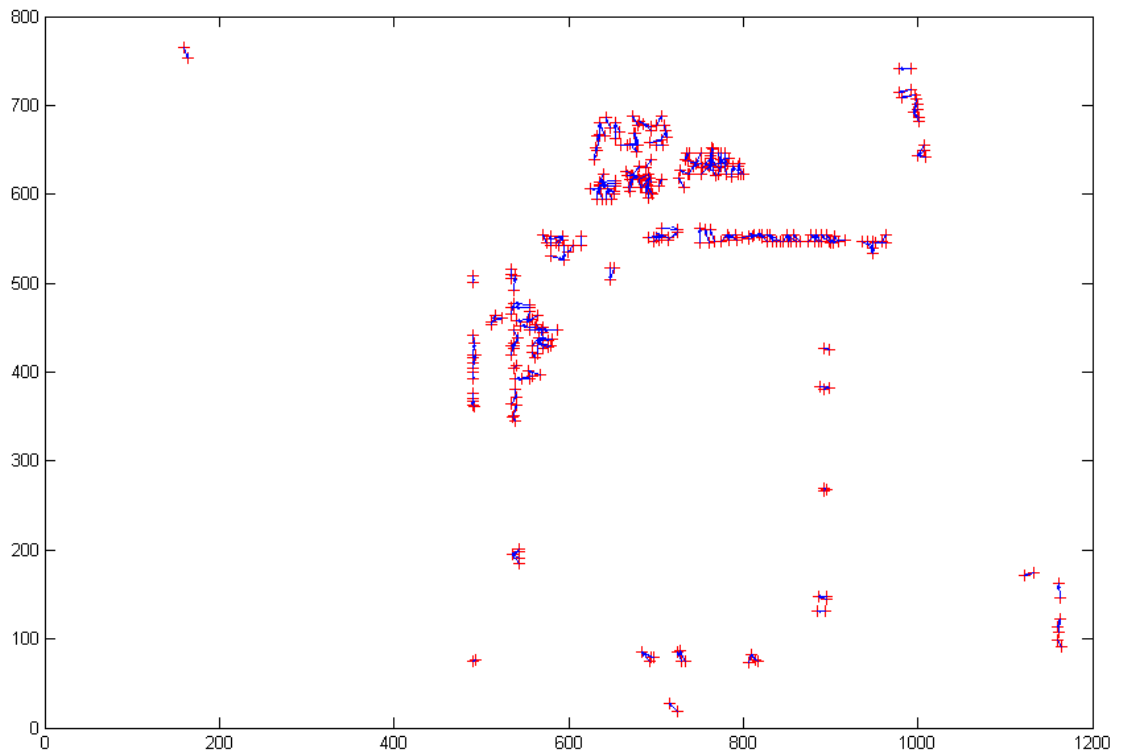
5.2. Kodas vykdomas MatLab programa

Stereo paveikslų kalibravimui ir išlyginimui buvo naudojamas programos MatLab kalibravimo įrankių rinkinys “toolbox_calib” [45]. Tai Jean-Yves Bouguet autoriaus iš autorių Reg Willson, Janne Heikkilä, Olli Silven, Zhang įvairių algoritmų ir metodų sudarytas įrankių rinkinys naudojamas kamerų kalibravimui ir paveikslų išlyginimui.

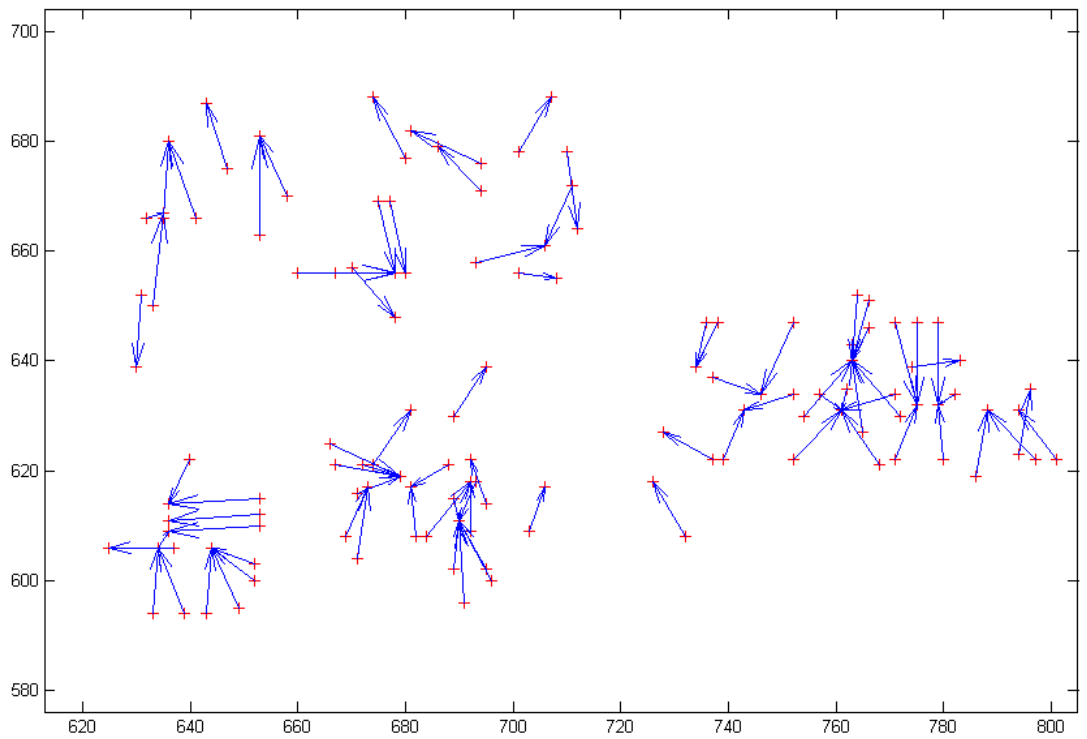
Pirmiausia charakteringiems taškams rasti buvo naudojama MatLab funkcija torr. Tačiau šios funkcijos gauti rezultatai netenkino, nes randami taškai buvo skirtingose vietose (5.8 pav.), o jų išsibarstymas buvo visomis kryptimis (5.9, 5.10 pav.). Panašūs rezultatai gauti ir su Harris kampu detektoriumi (5.11, 5.12 pav.). Čia taip pat aiškiai matyti radialinių iškreipymų išlyginimas.



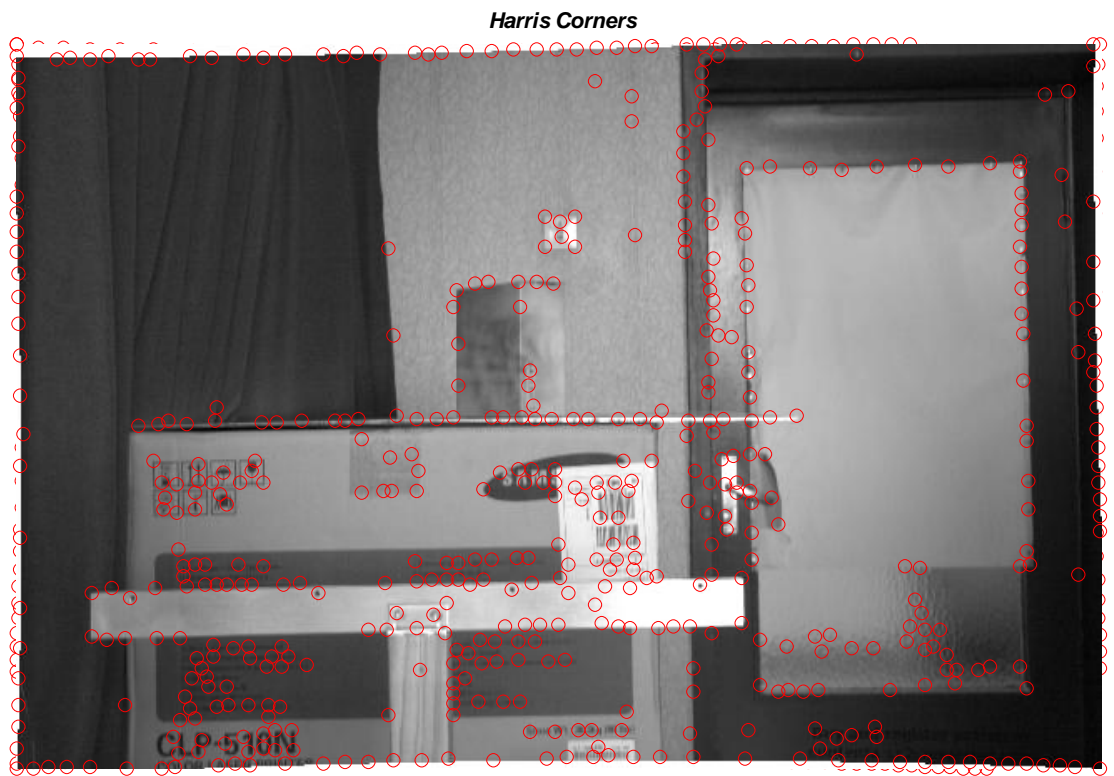
5.8 pav. Rasti taškai iš kairės ir dešinės kameros paveikslų



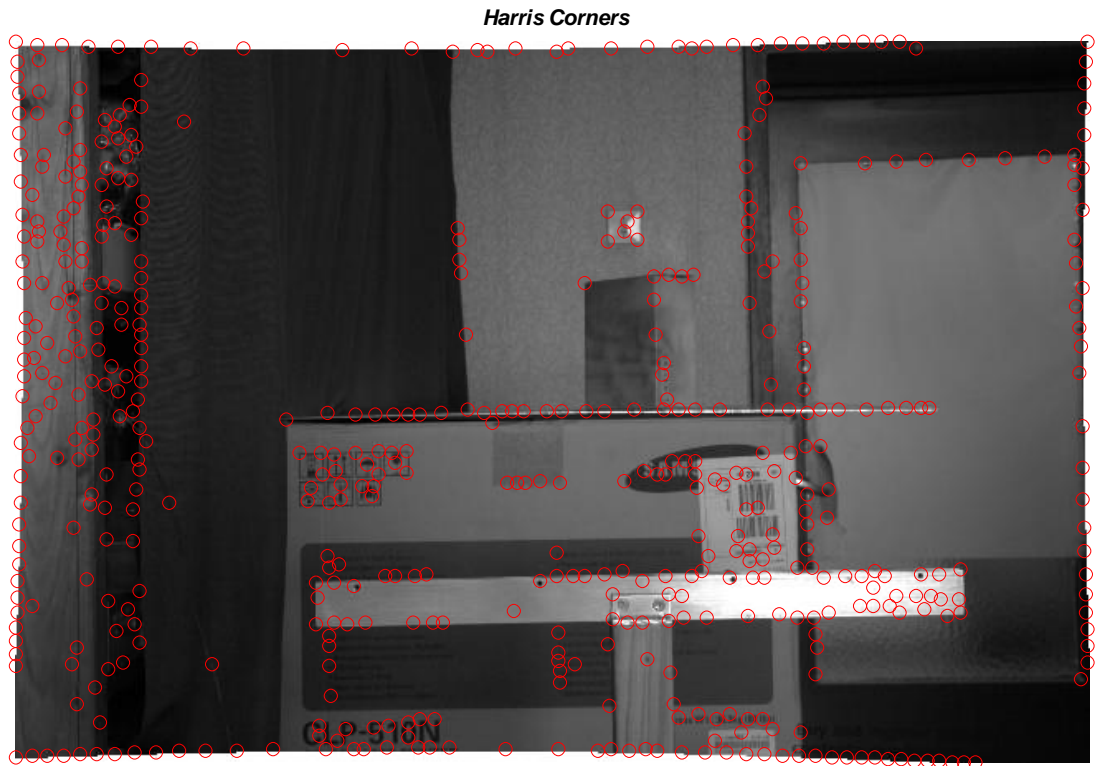
5.9 pav. Rasti charakteringi taškai iš kairės ir dešinės kameros paveikslų



5.10 pav. Charakteringų taškų radimo kryptys

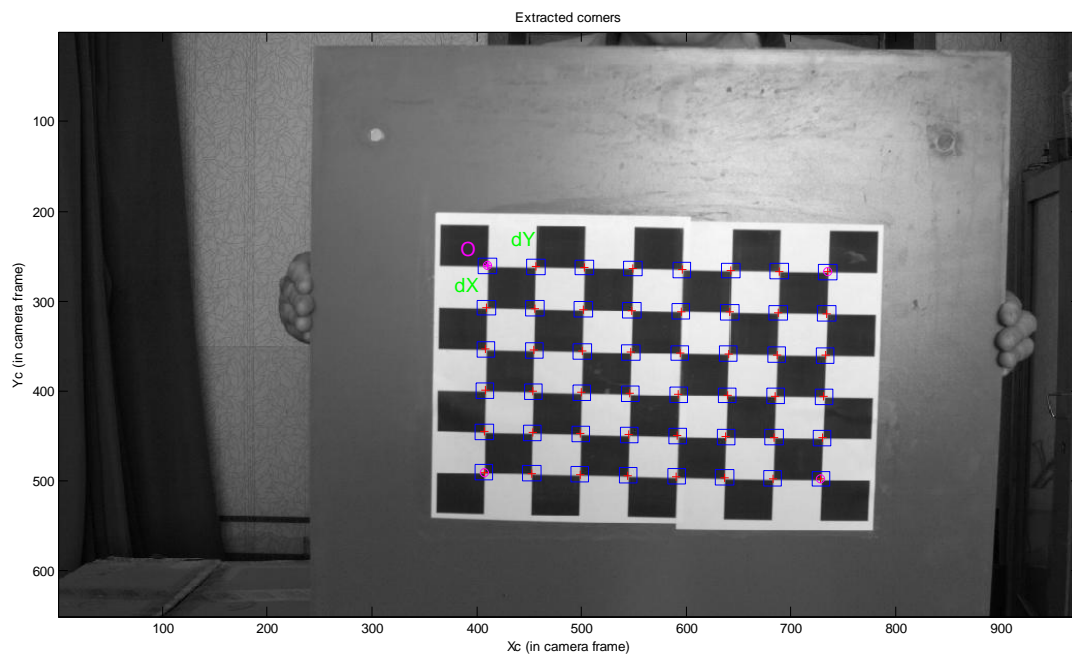


5.11 pav. Dešinysis paveikslas su charakteringais taškais

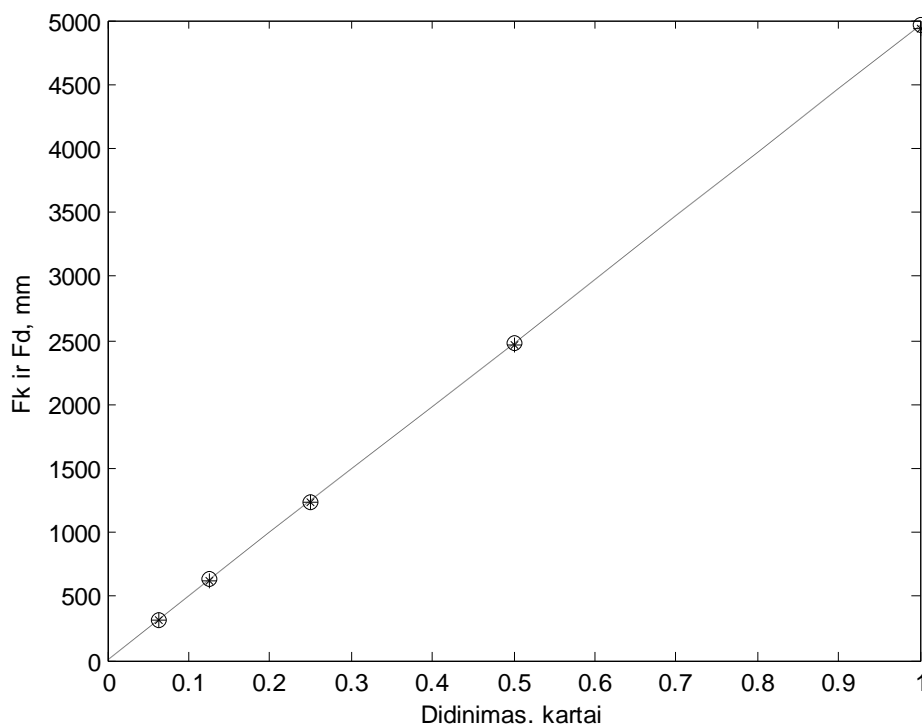


5.12 pav. Kairysis paveikslas su charakteringais taškais

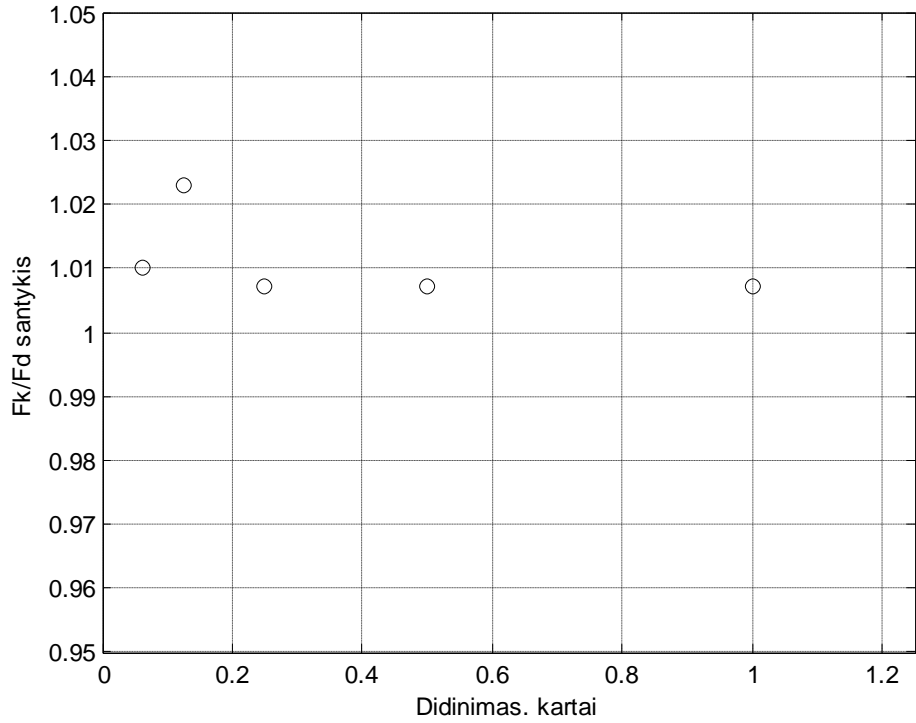
Po to buvo vykdomi bandymai su įrankių rinkiniu *“toolbox_calib”*. Pirmiausia vykdomas kalibravimas su skirtingais šablonais (vienas iš kalibravimo šablonų pavaizduotas 5.13 paveiksle). Buvo nustatinėjama kaip židinio nuotolį įtakojo paveikslų dydis. Buvo sulyginami $1/16$, $1/8$, $1/4$, $1/2$, karto sumažinti ir originalaus dydžio paveiksiai. Nustatyta jog nuo paveikslo mažinimas/didinimas ir apskaičiuojamas židinio nuotolis yra tiesiškai priklausomi (5.14 pav.). Taip pat nustatyta jog mažinant originalų paveikslą $1/2$ ir $1/4$ karto jų židinio nuotolių santykis kito 1% ribose (5.15 pav.). Taip pat nustatyta kaip kinta jutiklių matricių centrinių taškų x ir y koordinatų priklausomybės nuo vaizdo didinimo (5.16 ir 5.17 paveiksiai).



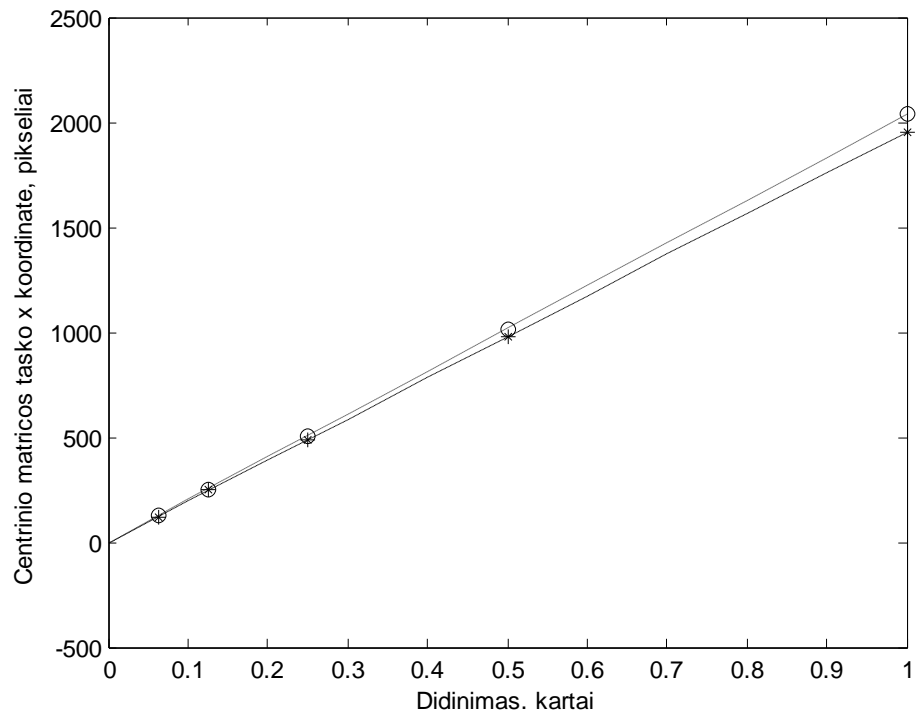
5.13 pav. Kalibravimo šablonas; Detektuoti kalibravimo taškai, pagal kuriuos atliekamas abiejų kamerų vidinių ir išorinių parametru kalibravimas



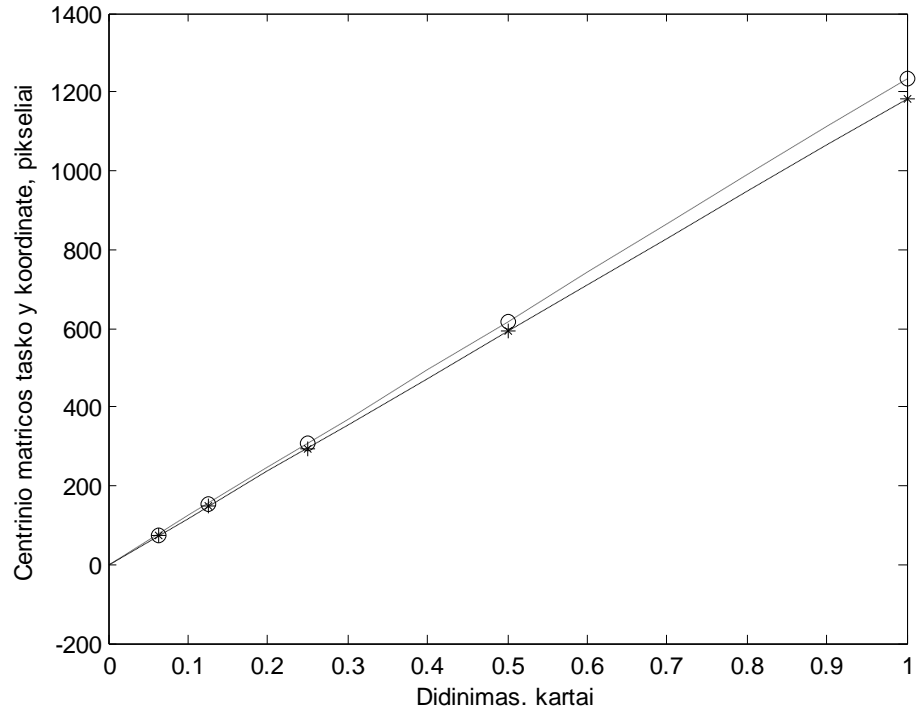
5.14 pav. Kairiosios ir dešinėsios kamerų židinių nuotolių priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera)



5.15 pav. Kairiosios ir dešinėsios kamerų židinių nuotolių santykio priklausomybė nuo vaizdo didinimo

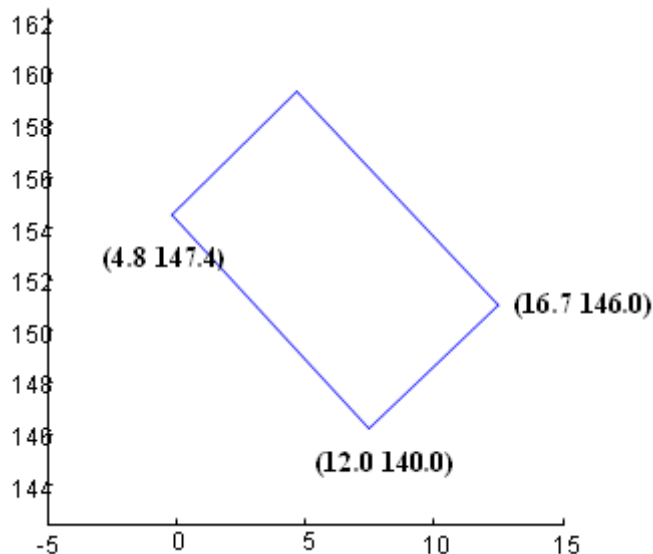


5.16 pav. Kamerų vaizdo jutiklių matricių centrinių taškų x koordinatė priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera)



5.17 pav. Kamerų vaizdo jutiklių matricių centrinių taškų y koordinatė priklausomybės nuo vaizdo didinimo (o - kairioji kamera, * - dešinioji kamera)

Gauti objekto kontūrai (5.18 pav.), pažymėjus ranka (*ginput* funkcija MatLab'e) charakteringus taškus, su tikromis (ant *x* ir *y* ašių) ir apskaičiuotomis koordinatėmis (skliausteliuose). Ketvirtas taškas (viršuje) liko nematomas, todėl nubrėžtas kaip ketvirtas stačiakampio kampas, žinant jog daiktas stačiakampis.



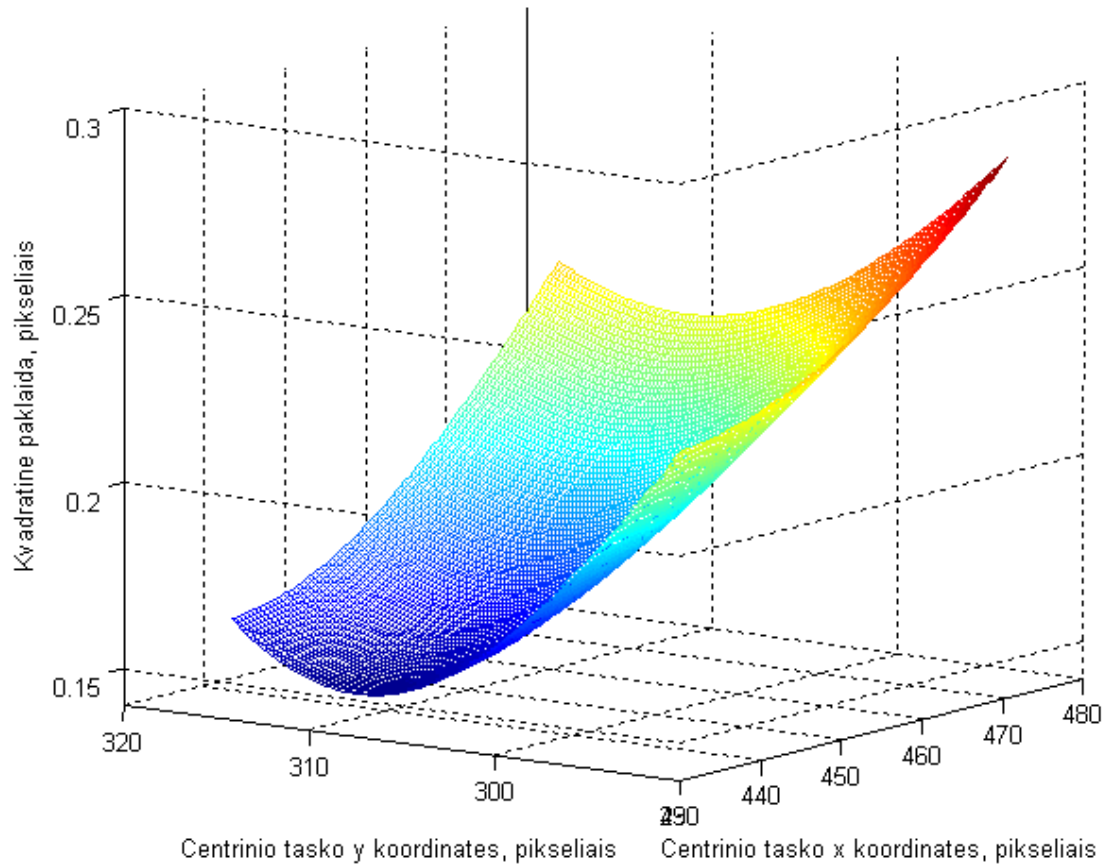
5.18 pav. Tikrosios ir apskaičiuotos objekto koordinatės

Toliau tirta taip pat pažymint objekto kampus ranka. Po kalibravimo ir išlyginimo MatLab programa stereo paveiksluose pažymėta nuo 14 iki 17 charakteringų taškų, turint jų koordinates buvo skaičiuojami poslinkiai (skirtumai) tarp kairiojo paveikslo x_k ir y_k koordinačių ir dešiniojo paveikslo x_d ir y_d koordinačių (3 lentelė). Taip pat apskaičiuotos 3D koordinatės

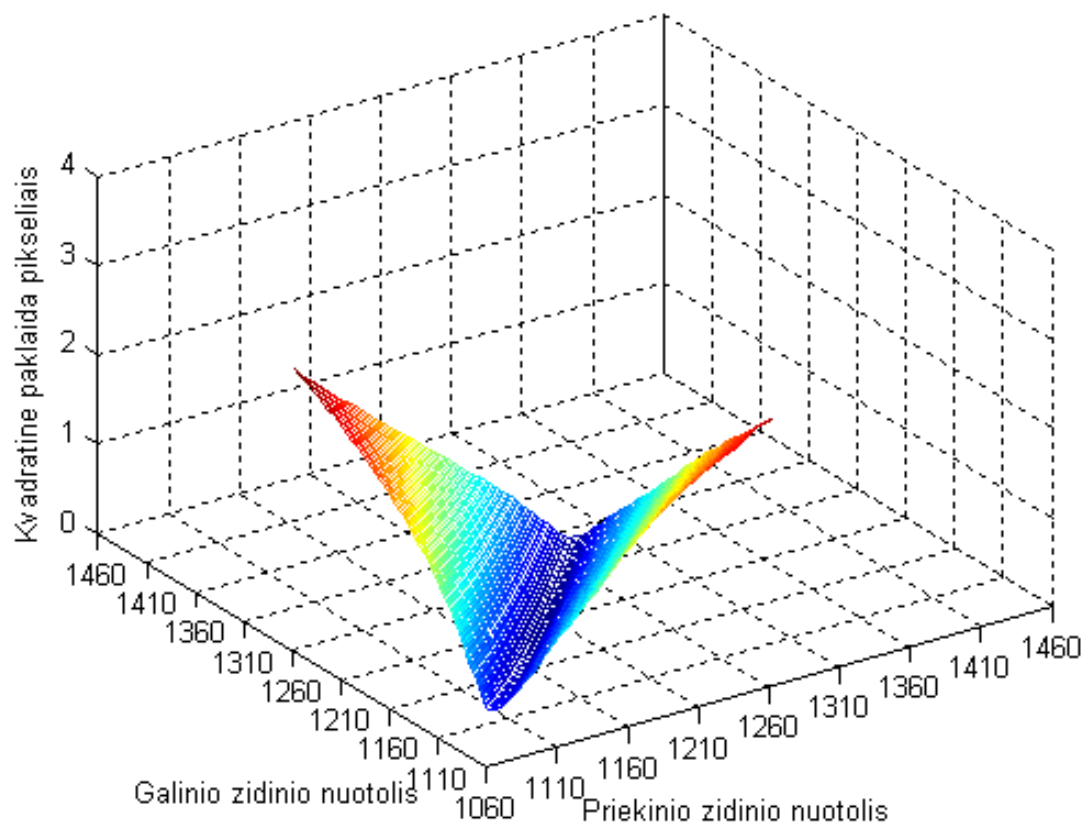
3 lentelė. Stereo paveikslų koordinatės ir jų skirtumai

Kairiojo K1 paveikslo x_k koordinatės	Kairiojo K1 paveikslo y_k koordinatės	Skirtumai $y_d - y_k$	Skirtumai $x_k - x_d$	X	Y	Z
279.9080	114.9453	0.9280	38.7733	-0.3912	-0.4381	3.3099
308.1533	114.9453	1.0133	39.1146	-0.3229	-0.4430	3.3554
306.9587	143.0200	0.7573	38.0054	-0.3238	-0.3661	3.3353
279.9933	143.0200	1.0133	38.8586	-0.3893	-0.3625	3.2927
502.0307	386.1240	1.1840	42.7840	0.1778	0.2771	3.2895
551.8653	386.1240	1.3973	41.2053	0.3071	0.2742	3.2421
549.9880	438.5187	-0.8213	38.9013	0.2982	0.4046	3.1973
501.0067	438.0067	-0.0533	41.9734	0.1738	0.3997	3.1867
475.9400	455.2333	-0.9067	45.3333	0.1018	0.4059	2.9247
750.2867	458.6467	-0.0533	47.0400	0.7194	0.4078	2.8367
750.7133	603.7133	0.3733	49.1733	0.7018	0.7137	2.7381
476.3667	599.0200	-1.3333	45.3334	0.0992	0.7208	2.8318
490.5000	620.3000	1.1420	49.7067	0.1270	0.7339	2.7015
742.2333	624.0330	1.4620	49.2800	0.6613	0.7344	2.6531
Kairiojo K6 paveikslo x_k koordinatės	Kairiojo K6 paveikslo y_k koordinatės					
279.8440	114.8813	0.9066	38.7413	-0.3912	-0.4381	3.3099
308.0040	115.0520	1.0773	38.6560	-0.3229	-0.4430	3.3554
308.0040	143.1267	1.1627	38.9973	-0.3213	-0.3662	3.3368
279.8440	143.0413	0.9920	38.8267	-0.3893	-0.3625	3.2927
504.8467	378.9667	-0.4266	39.7867	0.1930	0.2636	3.3481
537.7000	383.8733	0.2133	42.3467	0.2582	0.2559	3.0871
573.3267	380.4600	-0.6400	40.4267	0.3679	0.2642	3.3042
572.6867	434.6467	-0.8533	39.5734	0.3641	0.4051	3.2688
537.9133	438.2733	-1.0667	41.4933	0.2646	0.4004	3.1662
504.4200	433.3667	0	39.1467	0.1898	0.3907	3.2797
476.5267	454.1667	-2.4440	44.4694	0.1042	0.4037	2.9257
750.4467	458.8600	0.2013	47.3707	0.7194	0.4078	2.8367
747.8867	603.9267	1.2254	46.8587	0.6955	0.7140	2.7398
476.5267	599.2333	-0.7374	45.8347	0.1014	0.7209	2.8321
489.8067	618.1847	-3.6313	48.1920	0.1265	0.7529	2.7346
742.3933	622.0240	-0.4747	48.8746	0.6615	0.7304	2.6544

Iširta jog nuo centrinio taško padėties kitimo vidutinė kvadratinė paklaida mažai kinta, todėl negalime tiksliai įvertinti centrinio taško padėties (5.19 pav.). Taip pat ir židinio nuotolis (pagal kalibravimo taškus) mažai priklauso nuo vidutinės kvadratinės paklaidos, todėl taip pat negalime tiksliai vertinti židinio nuotolio (5.20 pav.).



5.19 pav. Centrinio taško priklausomybė nuo vidutinės kvadratinės paklaidos



5.20 pav. Židinio nuotolio priklausomybė nuo vidutinės kvadratinės paklaidos

IŠVADOS

Vaizdų apdorojimo kokybė priklauso nuo daugybės veiksnių tokių kaip: suspaudimas (t.y. paveikslo formato parinkimas), kamerų nustatymų (fokusavimas, vyzdinė diafragma) ir jų pozicijos tarpusavyje (atstumas tarp lęšių centrų, pasukimas viena kitos atžvilgiu), programinė įranga ir algoritmai naudojami apdorojant vaizdus. Taip pat ir nuo pačių vaizdų dydžio, fono, tekstūros.

Naudojant *OpenCV* bibliotekos funkcijas gautos erdvinės objektų koordinatės netenkinamos, todėl buvo bandoma vaizdus registruoti ne įprastiniu fotoaparatu nuotraukų formatu *JPEG*, bet formatu *RAW*, vėliau apdorojimui juos konvertuojant į *BMP* formatą. Tokiu būdu stengtasi išvengti vaizdo glaudinimo sukurto vaizdo triukšmo. Vaizdų apdorojimui panaudota *MatLab* programinis paketas ir šiam paketui laisvai platininami kompiuterinės regos įrankiai. Buvo bandomos įvairios funkcijos pavyzdžiui kampų radimo funkcija *torr* ir *Harris* metodas, tačiau jos gerų rezultatų nedavė. Charakteringi taškai randami skirtingose vietose, o epipolinės linijos netiesios, tai reiškia išlyginti paveikslų neįmanoma. Tikslesni tyrimai buvo pasiekti pasinaudojus kalibravimo įrankių rinkiniu "*toolbox_calib*", su kuriuo buvo toliau atliekami bandymai. Buvo iširta jog židinio nuotolis tiesiškai priklausomas nuo paveikslo didinimo/mažinimo, ko galima tikėtis iš teorinės analizės Taip pat palygintos apskaičiuotos koordinatės su tikrosiomis. Ranka pažymint įvairių objektų kampus rezultatai gauti gana tikslūs, maksimalus skirtumas tarp to paties kampo iš stereo paveikslų buvo 4 pikseliai. O kur matyti tikslus kampai, skirtumų išvis nėra arba su 1 pikselio neapibrėžtimi. Apskaičiuotos ir išmatuotos *Z* koordinatės (atstumas nuo kameros iki objekto) netikslumas buvo apie 4% - 5%. Nustatyta, kad kameros kalibravimo metu kalibravimo šablono kampų vidutinė kvadratinė paklaida mažai kinta nuo centrinio taško padėties, tiek ir nuo židinio nuotolio, todėl tiksliai negalima įvertinti nei židinio nuotolio atstumo nei centrinio taško padėties.

LITERATŪRA

1. **Serafinavičius P., Sajauskas S., Daunys G.** Evaluation of Hand Pointing System Based on 3–D Computer Vision // *Electronics and Electrical Engineering*. – Kaunas: Technologija, ISSN 1392–1215 – 2008. – No. 8(88). – p. 95–98. [ISI Web of Sciences, INSPEC, VINITI, EBSCO].
2. **Zhang.** „A Flexible New Technique for Camera Calibration“, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, November 2000.
3. **Huang-Faugeras.** Camera self-calibration. University of Cambridge [interaktyvus], [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <<http://mi.eng.cam.ac.uk/~prdsm2/Research/SelfCalibration/>>
4. **Tsai R.Y.** „An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision“, CVPR’86 Proceeding, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, June 22-26, pp. 364-373, 1986. [interaktyvus] [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/DIAS1/>
5. **Faugeras O.** Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press, 1993.
6. **More J.** „The Levenberg-Marquardt Algorithm, Implementation, and Theory,“ Numerical Analysis, G.A. Watson, ed., Springer-Verlag, 1977.
7. **Mainali P., Yang Q., Lafruit G., Van Gool L., Lauwereins R.** Robust Low Complexity Corner Detector. *IEEE Transactions on Circuits and Systems for Video Technology*, April 2011, vol. 21, no. 4, p. 435-445.
8. **Wyman.** Group Meeting: Extension of M-VOTE: Improving Feature Detection, 24 October 2005.
9. **Zuliani M., Kenney C., Manjunath B. S.** A Mathematical Comparison of Point Detectors. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2004 (CVPRW’04) 1063-6919/04.
10. **Kenney C., Manjunath B., Zuliani M., Hower G., Van Nevel A.** „A condition number for point matching with application to registration and post-registration error estimation,“ *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1437–1454, November 2003.
11. **Noble A.** „Descriptions of image surfaces,“ Ph.D. dissertation, Department of Engineering Science, Oxford University, 1989.
12. **Daunys G., Vysniauskas V.** Face image segmentation for gaze tracking // Department of Electronics, Siauliai University, Siauliai, Lithuania.

13. **Baker S., Patil R., Cheung G., Matthews I.** Lucas-Kanade 20 Years On: Part 5, CMU-RI-TR-04-64.
14. **Bradski G., Kaehler A.** Learning OpenCV Computer Vision with the OpenCV Library, September 2008.
15. **Scharstein D., Szeliski R.** A taxonomy and evaluation of dense two-frame stereo correspondence algorithms [interaktyvus]. [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <<http://vision.middlebury.edu/stereo/eval/>>.
16. **Anonymous.** Accurate and efficient stereo matching with AD-census measure, cross-based regions and multi-step refinement. CVPR 2011 submission 1039.
17. **Bleyer M., Rother C., Kohli P.** Surface stereo with soft segmentation. CVPR 2010.
18. **Anonymous.** Stereo matching based on under- and over-segmentation with occlusion handling. Submitted to PAMI 2009.
19. **Mattocchia S., Tombari F., Di Stefano L.** Stereo vision enabling precise border localization within a scanline optimization framework. ACCV 2007.
20. **Anonymous.** Memory efficient semi-global matching. ICCV 2011 submission 641.
21. **Liu T., Zhang P., Luo L.** Dense stereo correspondence with contrast context histogram, segmentation-based two-pass aggregation and occlusion handling. PSIVT 2009.
22. **Kosov S., Thormählen T., Seidel H.-P.** Accurate real-time disparity estimation with variational methods. ISVC 2009.
23. **Röhl S., Speidel S., Gonzalez D.-Aguirre, Suwelack S., Asfour T., Dillmann R.** From stereo image sequences to noise-free surface models. Submitted to ICRA 2011
24. **Stankiewicz O., Wegner K.** Depth map estimation software version 2. ISO/IEC MPEG meeting M15338, 2008.
25. **Ambrosch K., Kubinger W.** Accurate hardware-based stereo vision. CVIU 2010.
36. **Scharstein D., Szeliski R.** A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 2002. Dynamic programming, similar to Bobick and Intille (IJCV 1999)
37. **González V., Cabezas I.** Estimación de puntos correspondientes mediante programación dinámica. Congreso Multimedia 2009.
38. **El-Etriby S., Al-Hamadi A., Michaelis B.** Dense stereo correspondence with slanted surface using phase-based algorithm. IEEE ISIE 2007.
39. **Stivanello M., Leal E., Palluat N.** Stereo vision based on local correspondence with regional support. Submitted to Image and Vision Computing.

40. **Scharstein D., Szeliski R.** A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 2002. Scanline optimization (1D optimization using horizontal smoothness terms)
41. **Lai X., Zhu S.** A mutual information based non-parametric transform stereo matching algorithm. Submitted to ACTA OPTICA SINICA 2010.
42. **Nalpantidis L., Gasteratos A.** Stereo vision for robotic applications in the presence of non-ideal lighting conditions. Image and Vision Computing 2010.
43. **Olague G., Fernández F., Pérez C., Lutton E.** The infection algorithm: an artificial epidemic approach for dense stereo correspondence. Artificial Life, 2006.
44. A Simple 3D Scanner [interaktyvus], [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <http://www.sjbaker.org/wiki/index.php?title=A_Simple_3D_Scanner>
45. **Bouguet J.-Y.** Camera Calibration Toolbox for Matlab, Last updated July 9th, 2010 [interaktyvus]. [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <http://www.vision.caltech.edu/bouguetj/calib_doc/>
46. **Dervinis D.** Head Orientation Estimation using Characteristic Points of Face // Electronics and Electrical Engineering. Kaunas: Technologija, 2006. – No. 8(72). – P. 61–64.
47. **Palme E.** Stereo Vision Library for Obstacle Avoidance Applications, October 3, 2007.
48. **Calin G., Roda V. O.** Real-time disparity map extraction in a dual head stereo vision system [interaktyvus]. Fribourg, 2007, vol. 37, no.1, [žiūrėta 2011 m. gegužės 23 d.]. Prieiga per internetą: <http://www.scielo.org.ar/scielo.php?pid=S0327-07932007000100005&script=sci_arttext>. ISSN 0327-0793



Tsukuba



Venus



Teddy



Cones

