

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ KATEDRA

Simas Jankūnas

**Panoramiškai apjungiamų nuotraukų metodų
tyrimas**

Magistro darbas

Darbo vadovas
dr. A. Ostreika

Kaunas
2011

Turinys

1. Įvadas	2
2. Panoramiškai apjungiamų nuotraukų metodų analizė	4
2.1. Vaizdų gavimas	4
2.1.1. <i>Vaizdų gavimas keičiant fotoaparato kryptį</i>	4
2.1.2. <i>Vaizdų gavimas slenkant fotoaparata</i>	5
2.1.3. <i>Vaizdų gavimas fotoaparata laikant rankose</i>	6
2.1.4. <i>Bendrosios vaizdų gavimo problemos</i>	7
2.1.5. <i>Gautų vaizdų klasifikavimas</i>	7
2.2. Vaizdų registravimas	8
2.2.1. <i>Požymių aptikimas naudojant Hario kampų metoda</i>	9
2.2.2. <i>Požymių sugretinimas naudojant koreliacijos metoda</i>	11
2.2.3. <i>Požymių sugretinimas naudojant sutampančių fazių metoda</i>	13
2.2.4. <i>Požymių aptikimo ir sugretinimo metoda – SIFT</i>	15
2.2.5. <i>Tinkamų taškų atrinkimo metoda – RANSAC</i>	20
2.3. Vaizdų apjungimas	26
2.3.1. <i>Homografinė ir afininė transformacijos</i>	26
2.3.2. <i>Santykis su kitomis geometrinėmis transformacijomis</i>	27
2.3.3. <i>Homografijos skaičiavimas</i>	29
3. Eksperimentų planavimas	32
4. Eksperimentinis tyrimas	33
4.1. Požymių aptikimo metodų palyginimas	33
4.2. Požymių sugretinimo metodų palyginimas	35
4.3. Transformacijų skaičiavimai	41
4.4. Vaizdo maišymas	47
4.5. Vaizdo kokybės tyrimai	47
5. Išvados	50
6. Literatūra	51
7. Summary	52
8. Terminai	53
9. Sutrumpinimai ir žymėjimai	53

1. Įvadas

Panoramiškas nuotraukų apjungimas, tai procesas dvi ar daugiau nuotraukų, su tarpusavio persidengiančiais plotais, apjungiantis į panoraminę ar vieną aukštos kokybės nuotrauką. Panoramiškai apjungiami vaizdai yra plačiai naudojami nuotoliniuose tyrimuose, mediciniuose vaizdiniuose tyrimuose, kompiuteriniuose matymuose ir kitose srityse. Kalbant apie programinę įrangą paprastam vartotojui, panoraminius vaizdus galima gauti naudojant vieną populiariausių grafikos redagavimo programų, Adobe Photoshop, taipogi egzistuoja keletas smulkių programinių paketų orientuotų į panoraminių vaizdų apjungimą – Autostitch, Hugin, Pgtui, Panorama Tools.



1 pav. Panoraminė nuotrauka

Panoramiškai apjungiamų vaizdų procesas yra sudarytas iš kelių pagrindinių žingsnių – būdingų požymių aptikimo, būdingų taškų sugretinimo persidengiančiuose plotuose, bei transformacijos skaičiavimo. Vien būdingų taškų aptikimo metodų yra daugybė – SIFT, SURF, GLOH, HOG, LESH, Hario kampų, SUSAN kampų ir daugybė kitų. Šiame darbe tirsime du būdingų taškų aptikimo metodus, tai SIFT, kurio veikimas pagrįstas šviesesnių ar tamsesnių, nei šalia esančių taškų aptikimu, bei Hario kampų metodą – aptinkantį kampus vaizduose. Būdingų taškų sugretinimo metodų nėra tiek daug, tačiau šiame darbe lyginsime tris skirtingus metodus: vienas jų jau minėtas SIFT, koreliacijos, bei sutampančių fazių. Transformacijos taškų rinkinio atlikimui naudosime RANSAC iteracinį metodą. Paskutinis etapas, kurį tirsime, tai transformacijos skaičiavimas ir vaizdo perrinkimas. Šiame etape naudosime dvi skirtingas transformacijas – afininę ir homografinę. Yra dar vienas etapas,

kurio šiuo atveju netirsime, kad geriau išryškėtų metodų privalumai ir trūkumai. Tai vaizdo maišymo algoritmo naudojimas, kad panaikintume atsiradusius susijungimų trikdžius.

Pagrindinės vaizdų apjungimo problemos yra tikslumas, bei apjungimo greitis, jos labiausiai išryškėja moksliniuose tyrimuose, kur dėl didelių apdorojamų vaizdų mastų ir tikslumo reikalavimų skaičiavimo laikas smarkiai padidėja. Pavyzdžiui: palydovu gautų paviršių vaizdų apjungimas, kur duomenų mastai yra milžiniški arba medicinoje, kur atliekami skaičiavimai turi būti nepriekaištingai tikslūs. Tokių sistemų skaičiavimai reikalauja galingų kompiuterių, bei atima daug laiko. Todėl šiame darbe daug dėmesio skirsime į tai, kai greitai metodas veikia. Nustatysime greičiausią metodą, tačiau nepamiršime ir kokybinio rodiklio, nes negaunant reikiamos kokybės, greitis tampa nebesvarbus. Atliekant tyrimus skirtingais metodais, tikimės rasti patobulinimus metodas, kad skaičiavimo greitis ar kokybė išaugtų.

2. Panoramiškai apjungiamų nuotraukų metodų analizė

Vaizdų apjungimas panoramiškai, yra skirstomas į du pagrindinius etapus: vaizdų registravimą¹ ir vaizdų apjungimą, tačiau prieš aptariant šituos etapus yra dar vienas etapas, be kurio neįvyktų vaizdų apjungimas, tai vaizdų gavimas, kurį dabar ir aptarsime.

2.1. Vaizdų gavimas

Kaip jau ir minėjome, pirmiausia aptarsime pagrindinius panoraminių vaizdų gavimo principus, bei pagrindines problemas, kas įtakoja tolimesnių veiksmų rezultatus. Teisingai pasirinktas vaizdų gavimo metodas, atsižvelgiant į tai, koku tikslu gauti vaizdai bus naudojami, vaizdų apjungimą gali supaprastinti, tokiu atveju naujai apjungtų vaizdų kokybė yra gaunama geresnė.

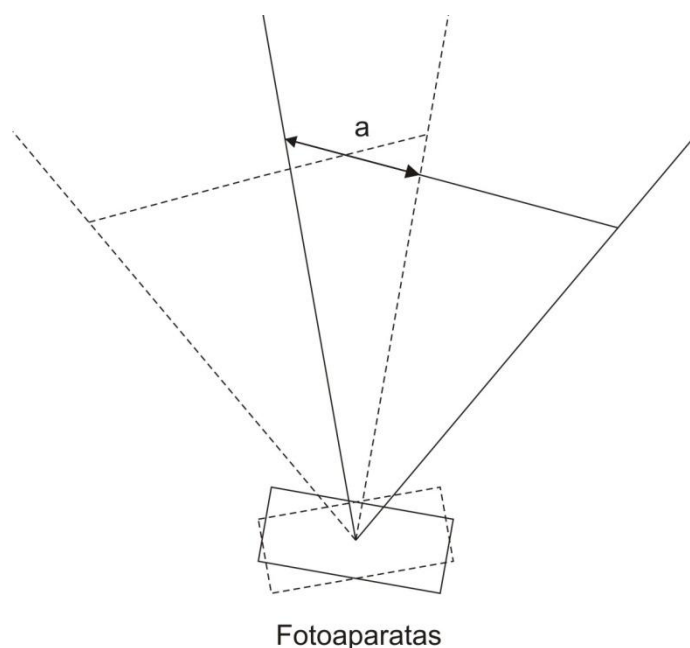
2.1.1. Vaizdų gavimas keičiant fotoaparato kryptį

Tai, labai retai naudojamas vaizdų gavimo metodas, kuris yra priskiriamas prie panoraminių vaizdų gavimo metodų [1]. Vaizdai gaunami, kai fotoaparatas yra pritvirtinamas prie specialaus stovo, fotoaparatas yra nukreipiamas į reikiamus objektus ir sukamas pasirinkta kryptimi aplink vertikaliąją ašį. Kiekvienas vaizdas turi turėti persidengiantį plotą su kitu iš savo šonų. Šiuo metodu gautų nuotraukų persidengiančių plotų dydis vaidina svarbų faktorių. Optimaliausias persidengiantis plotas turėtų būti bent 50% su kiekvienu kaimyniniu vaizdu [2]. 2 pav. vaizduoja dviejų persidengiančių nuotraukų geometriją iš viršaus:

čia,

a – *persidengiantis vaizdų plotas*.

¹ Vaizdų registravimas – procesas, skirtas pertvarkyti skirtingus duomenų rinkinius į vieną koordinatų sistemą.



2 pav. Persidengiančių vaizdų geometrija pasukant kamerą

Tačiau persidengiantis plotas turėtų būti didesnis, jei yra galimas kameros pakrypimas horizontalia kryptimi. Taipogi yra dar vienas faktorius, kad objektai realiame pasaulyje yra suprojektuoti 2D plokštumoje, vadinasi, kad gaunant vaizdus sukant fotoaparata, kiekviena nuotrauka bus ant skirtingų paviršių. Dėl to, tokie vaizdai turi būti projektuojami ant tokio pat paviršiaus, kaip cilindras ar sfera, kad būtų galima gauti kokybiškus galutinius rezultatus. Pagal daugumą prognozių, po projektavimo vaizdų kokybė krenta. Ši problema labiausiai išryškėja kai vaizdai yra gaunami naudojant fotokameras su trumpesniu židinio nuotoliu, ar kai vaizdo laukas yra platesnis.

Vienas privalumas, gaunant vaizdų serija fotoaparato sukimo metodu yra tai, kad fotoaparatas gali stovėti vienoje vietoje. Fotoaparato sukimas nereikalauja didelių matavimų, todėl įvykdyti pasukimus yra paprasta. Vis dėlto, gauti vaizdai yra ne vienodo paviršiaus ir turi būti suprojektuoti ant vienodo paviršiaus prieš atliekant apjungimą.

Nepaisant šito metodo paprastumo ir trūkumų, jis yra privilegijuotas kaip panoraminių nuotraukų apjungimo gavimo metodas.

2.1.2. Vaizdų gavimas slenkant fotoaparata

Tai vaizdų gavimo metodas, kai fotoaparatas yra perkeliamas lygiagrečiai gaunamo vaizdo plokštumai. Mažesniuose plotuose, fotoaparatas gali būti pritaistas prie plokštės ar

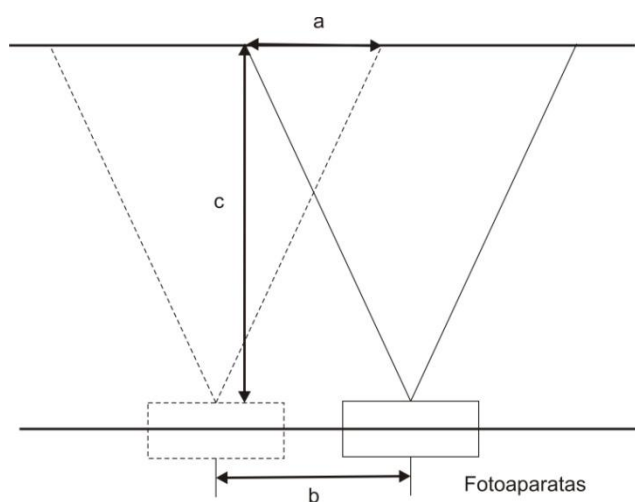
lentos, kad būtų galima tiksliai atlikti poslinkius. Fotoaparatas yra slenkamas pirmyn, tokiu poslinkiu, kad sekančio vaizdo dalis persidengtų su buvusiojo, tai puikiai matosi 3 pav..

čia,

a – persidengiantis plotas

b – kameros poslinkis

c – atstumas nuo kameros iki objektų



3 pav. Persidengiančių vaizdų geometrija paslenkant kamerą

Tikslus persidengiančio ploto dydis yra nustatomas atsižvelgiant į fotoaparato parametrus. Gaunamuose vaizduose slenkant fotoaparata yra svarbu, kad vaizdo projekcija būtų lygiagreči fotoaparato poslinkiams, nes objektų dydis vaizduose priklauso nuo kameros poslinkių.

Šio metodo trūkumas yra tai, jog didėjant fotoaparato poslinkiui nuo pirmojo taško, didėja ir atstumo c skirtumai tarp objektų skirtinguose vaizduose. Dėl to, kai mus dominantys objektai yra toli nuo fotoaparato yra sunku nustatyti tikslus poslinkius. Be to, šiuo metodu, vaizdai gaunami yra vienos projekcijos, todėl nesuteikia 3D aplinkos sąlygų, skirtingai nei vaizdai gauti naudojant fotoaparato pasukimą.

2.1.3. Vaizdų gavimas fotoaparata laikant rankose

Šis vaizdų gavimo metodas palyginus yra labai paprastas, vartotojas laiko fotoaparata pats ir fotografuoja jam reikalingus aplinkos vaizdus, keičiant tik fotoaparato kryptį ar kampą, apytiksliai lygiagrečiai vaizdo plokštumai. Tačiau taip gautus vaizdus, dėl bereikalingų fotoaparato pasukimų, netikslingų perkėlimų yra labai sunku apjungti į panoraminį vaizdą.

Taipogi pasunkėja apjungimo procesas, nes persidengiančių plotų dydžiai tarp nuotraukų gali būti labai skirtingi. Kad išvengtume kai kurių problemų, visada yra pageidaujama stengtis padaryti kuo didesnius vaizdų tarpusavio persidengiančius plotus. Didesni persidengiantys plotai reiškia, kad kameros pasukimai ar perkėlimai yra mažesni tarp kiekvieno vaizdo, kas sumažina nesuderinamus plotus tarp vaizdų.

Vis dėlto netgi atsižvelgiant į tai, kad šiuo metodu gautus vaizdus yra sunkiau apjungti, jis turi ir didelį privalumą - dėl paprastos įrangos jį galime naudoti daugelyje vietų.

2.1.4. Bendrosios vaizdų gavimo problemos

Viena iš dažniausiai pasitaikančių problemų gautuose vaizduose yra intensyvumo perėjimas tarp gretimų vaizdų. Idealiu atveju intensyvumo reikšmės tame pačiame plote ar objekte turi būti lygios. Tačiau dėl apšvietimo pokyčių, tarp kameros ir šviesos šaltinio, gretimų vaizdų intensyvumas tuose pačiuose plotuose arba objektuose skiriasi. Dabar ši problema mažiau opi, nes didžiausi nuokrypiai atsiradavo, kai vaizdai būdavo gauti ne skaitmeniniu formatu ir juos tekdavo perkelti į skaitmeninį formatą (pavyzdžiui, skenuojant).

Kita problema, taipogi susijusi su apšvietimu, tai atspindžių atsiradimas plotuose, kur yra stiklai ar spindintys metalai. Dėl to ant gaunamų vaizdų tose vietose gaunamos „dėmės“.

2.1.5. Gautų vaizdų klasifikavimas

Gauti vaizdai prieš apjungimą gali būti klasifikuojami į keturias skirtingas grupes [3]: Skirtingų matymo taškų – tai vaizdai, gauti iš vienos vietos, tačiau iš skirtingų taškų. Šios grupės vaizdų gavimo tikslas yra, kad būtų galima padaryti didesnius dvimačius vietovės vaizdus arba reprezentuoti vietovę trimatėje erdvėje. Naudojimo pavyzdžiai: Nuotoliniuose tyrimuose² – tiriamos teritorijos nuotraukų apjungimuose.

Skirtingu metu gauti – tai vaizdai gauti iš tos pačios vietos, tačiau ne vienu metu, taigi neatmetama galimybė, kad vaizdai gali būti gauti skirtingomis sąlygomis. Tikslas yra surasti ir įvertinti vietovės pokyčius, kurie atsirado, tarp vaizdų gavimo laikotarpių. Naudojimo pavyzdžiai: Nuotoliniuose tyrimuose – bendruosiuose žemės plotų panaudojimo

² Nuotoliniai tyrimai – tyrimai atliekami analizuojant nutolusių objektų vaizdus įvairiuose elektromagnetinių bangų spektruose.

stebėjimuose, kraštovaizdžių planavimuose. Medicinos vaizdiniuose tyrimuose – gydymo terapijos stebėjimuose, navikų vystymosi stebėjimuose. Kompiuteriniuose matymuose³ – apsaugos ar judesio pokyčių stebėjimuose.

Skirtingų jutiklių – tai vaizdai gauti toje pačioje vietovėje, bet naudojant skirtingus jutiklius. To tikslas yra sujungti informaciją gautą skirtingais srautais, kad būtų galima gauti daug sudėtingesnius ir detalesnius vaizdus. Naudojimo pavyzdžiai: Medicinos vaizdiniuose tyrimuose – anatominei kūno struktūros užrašymui (magnetiniam rezonansui), kūno medžiagų apykaitos aktyvumo tyrimui (pozitronų emisijos tomografui), magnetinio rezonanso spektroskopijai.

Scenos registravimas modelyje - tai vaizdai gauti vietovėje, kai yra registruotas vietovės vaizdo modelis. Modelis gali būti kompiuteryje vaizduojama vieta, geografinėse informacijos sistemose, pavyzdžiui žemėlapiuose ar reljefiniame modelyje. Tikslas yra lokalizuoti gautus vaizdus vietovėje/modelyje ir palyginti juos. Naudojimo pavyzdžiai: Nuotoliniuose tyrimuose – žemėlapiuose ar geografinėse informacinėse sistemose duomenų registravimui iš oro ar palydovo.

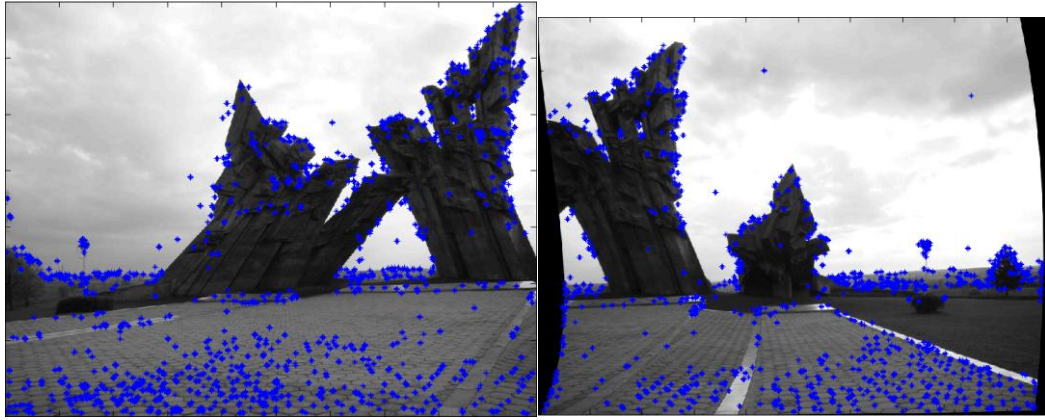
2.2. Vaizdų registravimas

Atsižvelgiant į anksčiau minėtus vaizdų gavimo klasifikavimus, akivaizdu, kad neįmanoma sukurti universalų metodą, kuris būtų taikomas visoms vaizdų registravimo užduotims atlikti. Kiekvienas metodas turėtų spręsti ne tik tariamus geometrinių transformacijų tipus tarp vaizdų, bet ir radiometrines transformacijas, trukdžių iškraipymus, kurie reikalauja ne tik registracijos tikslumo, bet ir požymių duomenų pritaikymo.

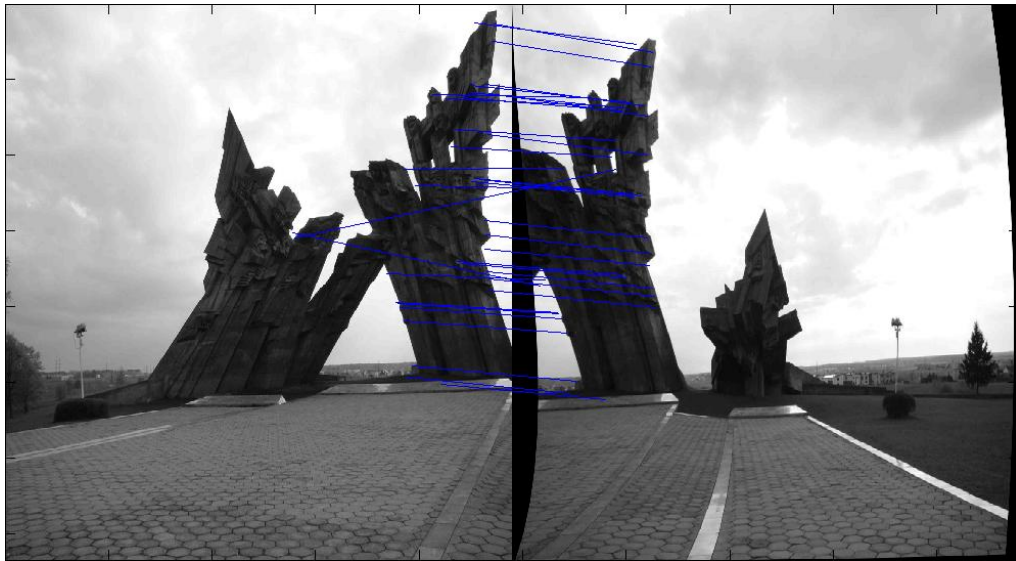
Nepaisant to, dauguma registracijos metodų apima du pagrindinius etapus *4 pav.*:

- *Požymių aptikimas* – tai automatiniu ar rankiniu būdu aptinkami ryškesni arba saviti objektai (briaunos, kontūrai, linijų susikirtimai, kampai ir t.t.). Visi šitie objektai yra reprezentuojami taškais (svorio centrais, linijų pabaigomis), kurie yra vadinami valdymo taškais (CP).
- *Požymių sugretinimas* – tai požymių panašumų ieškojimas tarp registruojamų vaizdų.

³ Kompiuterinis matymas – tai technologija ar mokslas apie vaizdus, su kuriais gali būti atlikti skaičiavimai kompiuteriuose.



4 pav. Požymių aptikimas



5 pav. Požymių sugretinimas

2.2.1. Požymių aptikimas naudojant Hario kampų metodą

Tai populiarus indikatorius, skirtas aptikti būdingus taškus, pasižymintis tuo, kad taškai nekinta nuo vaizdo pasukimų, dydžio, ryškumo ar kitų vaizdų trikdžių [5]. Hario indikatorius yra paremtas automatinė signalų koreliacijos funkcija, kur automatinė koreliacijos funkcija matuoja signalų pakitimus mažais žingsniais.

Tarkime, duotas postūmis $(\Delta x, \Delta y)$ ir taškas (x, y) , tai automatinės koreliacijos funkcija yra aprašoma taip:

$$c(x, y) = \sum_w [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (2.1)$$

čia $I(\cdot, \cdot)$ žymi vaizdo funkciją ir (x_i, y_i) yra and (x, y) centruoti Gausinės $e^{-(x^2+y^2)/(2\sigma^2)}$ funkcijos W taškai.

Pagal pirmosios eilės Teiloro plėtinį pastumtas vaizdas apytikriai lygus:

$$I(x_i + \Delta x, y_i + \Delta y) \approx I(x_i, y_i) + [I_x(x_i, y_i)I_y(x_i, y_i)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (2.2)$$

čia $I_x(\cdot, \cdot)$ ir $I_y(\cdot, \cdot)$ atitinkamai pažymi x ir y dalines išvestines.

Pakeičiant apytikres reikšmes iš aukščiau nurodytų formulių atitinkamai iš (2.2) į (2.1) žvelgiant nuo pradžios, gausime:

$$\begin{aligned} c(x, y) &= \sum_W [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \\ &= \sum_W \left((x_i, y_i) - I(x_i, y_i) - [I_x(x_i, y_i)I_y(x_i, y_i)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right)^2 \\ &= \sum_W \left(-[I_x(x_i, y_i)I_y(x_i, y_i)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right)^2 \\ &= \sum_W \left([I_x(x_i, y_i)I_y(x_i, y_i)] \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right)^2 \\ &= [\Delta x \quad \Delta y] \begin{bmatrix} \sum_W (I_x(x_i, y_i))^2 & \sum_W I_x(x_i, y_i)I_y(x_i, y_i) \\ \sum_W I_x(x_i, y_i)I_y(x_i, y_i) & \sum_W (I_y(x_i, y_i))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \\ &= [\Delta x \quad \Delta y] C(x, y) \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (2.3) \end{aligned}$$

čia, matrica $C(x, y)$ apimanti kaimyninę intensyvumo struktūrą. Tegul λ_1, λ_2 yra tikrinės matricos $C(x, y)$ reikšmės. Tuomet yra trys galimi atvejai:

- jei λ_1, λ_2 abu yra maži, tai automatinės koreliacijos funkcija yra lygi, tas vaizdo plotas yra apytikriai lygus intensyvumo konstantai.
- jei viena tikrinė reikšmė yra didelė, o kita maža, tuomet automatinės koreliacijos funkcija yra briaunos formos, tuomet tik vienas postūmis viena kryptimi įtakoja mažus pokyčius $C(x, y)$ ir ryškius pakitimus ortogonaliai kryptimi; tai pažymi kraštus.
- jei abi tikrinės reikšmės yra didelės, tuomet automatinės koreliacijos funkcija yra labai smaili, tokiu atveju postūmis bet kuria kryptimi duoda ryškaus padidėjimo rezultatą; tai pažymi kampus.

2.2.2. Požymių sugretinimas naudojant koreliacijos metodą

Kaip diferencialinis metodas, koreliacija yra sprendimas atsiradęs analizuojant pakitimus tarp dviejų nuoseklių vaizdų [6]. Kad rastumėme pirmojo vaizdo būdingus požymius antrajame - reikia paimti pirmojo vaizdo $g(t_1) = g_1$ ir palyginti su antrojo vaizdo ieškomos srities $g(t_2) = g_2$. Visoje srityje ieškome pozijos optimalios panašumui tarp dviejų vaizdų. Panašumo matavimo vienetas, turi būti atsparus ryškumo pokyčiams. Taigi, jei dviejuose erdviniuose modeliuose skiriasi tik konstanta α , kuri atspindi ryškumo skirtumus, šiuos modelius skaitome kaip lygius. Toks rezultatas vidinėse vektorių erdvėse reiškia, kad požymių vektoriai g_1 ir g_2 yra lygiagretūs. Tai gali būti tik vienu atveju, jei lygybė atsiranda Cauchy-Schwarz nelygybėje:

$$\left| \int_{-\infty}^{\infty} g_1(x)g_2(x-s)d^2x \right|^2 \leq \int_{-\infty}^{\infty} g_1^2(x)d^2x \int_{-\infty}^{\infty} g_2^2(x-s)d^2x. \quad (2.4)$$

Kitaip sakant, reikia maksimalizuoti kryžminės koreliacijos koeficientą

$$r(s) = \frac{\int_{-\infty}^{\infty} g_1(x)g_2(x-s)d^2x}{\left(\int_{-\infty}^{\infty} g_1^2(x)d^2x \int_{-\infty}^{\infty} g_2^2(x-s)d^2x \right)^{1/2}}. \quad (2.5)$$

Kryžminės koreliacijos koeficientas yra naudingas panašumų matavimuose. Jis yra nulinis, visiškuose nepanašumuose ir siekia maksimumą panašiuose požymiuose.

Panašiu būdu, kaip ir diferencialų metodas, koreliacijos metodas gali būti perdirbtas apjungiant susukimą ir taškų operacijas. Pirmasis žingsnis yra įvesti lango funkciją w į kryžminės koreliacijos koeficiento apibrėžimą. Langas yra sukamas aplink vaizdą, kad apskaičiuotumėme vietinį kryžminės koreliacijos koeficientą. Tuomet iš (2.5) lygties gausime:

$$r(x,s) = \frac{\int_{-\infty}^{\infty} w(x-x')g_1(x')g_2(x'-s)d^2x'}{\left(\int_{-\infty}^{\infty} w(x-x')g_1^2(x')d^2x' \int_{-\infty}^{\infty} w(x-x')g_2^2(x'-s)d^2x' \right)^{1/2}}. \quad (2.6)$$

Kryžminės koreliacijos koeficiento rezultatas yra keturmatė funkcija, priklausanti nuo vaizdo x ir poslinkio s pozicijos.

Greita iteracinė maksimumų paieška

Akivaizdu, kad aukščiau aptartas koreliacijos metodas reikalaujantis daug laiko. Tačiau, svarbias vietas apribojus ieškomos pozicijos maksimumu r , galima gauti žymų paieškos pagreitėjimą.

Kryžminė koreliacijos funkcija Teiloro serijoje, yra vienintelis tikslaus maksimumo pozicijos apskaičiavimo būdas. Išplečiame kryžminės koreliacijos koeficientą į antrosios eilės Teiloro plėtinį maksimalioje pozicijoje \check{s} ,

$$\begin{aligned} r(s) &\approx r(\check{s}) + \frac{1}{2} r_{xx}(\check{s})(s_1 - \check{s}_1)^2 + \frac{1}{2} r_{yy}(\check{s})(s_2 - \check{s}_2)^2 + r_{xy}(\check{s})(s_1 - \check{s}_1)(s_2 - \check{s}_2) \\ &= r(\check{s}) + \frac{1}{2} (s - \check{s})^T H(\check{s})(s - \check{s}), \quad (2.7) \end{aligned}$$

čia H yra Heeso matrica [6].

Mes nežinome maksimalaus koreliacijos koeficiento pozicijos. Todėl manome, kad antrosios eilės išvestinė yra konstanta, pakankamai artima maksimaliai pozicijai ir yra apskaičiuojama buvusios iteracijos $s^{(i)}$ pozicijoje. Jei neturime kitos informacijos, tai pradžinius skaičiavimus prilyginame nuliui: $s^{(0)} = 0$. Tol, kol nerandame koreliacijos koeficiento maksimumo pozicijos, $s^{(i)}$ turi turėti liekamąjį nuolydį, kuris gali būti apskaičiuojamas išvedant iš (2.7) lygties:

$$\nabla r(s^{(i)}) = H(s^{(i)})(s^{(i)} - \check{s}). \quad (2.8)$$

Su sąlyga, kad Hesso matrica yra invertuojama, galime išgauti sekančią iteraciją

$$s^{(i+1)} = s^{(i)} - H^{-1}(s^{(i)})\nabla r(s^{(i)}) \text{ su } s^{(0)} = 0. \quad (2.9)$$

Tokio tipo iteracija yra vadinama, Newton-Raphson iteracija. Kad apskaičiuotumėme postūmį, reikia skaičiuoti tik pirmosios ir antrosios eilės kryžminės koreliacijos koeficiento dalines išvestines.

2.2.3. Požymių sugretinimas naudojant sutampančių fazių metodą

Monogeninis signalas pristatomas, kaip apibrėžiantis vaizdo vietinį lopinėlių pagal

$$I(x_0 + x) = A_{x_0} \cos \varphi_{x_0}(x) \quad (2.10)$$

čia A_{x_0} yra centruota x_0 ir $\varphi_{x_0}(x)$ vietinė amplitudė, vadinama vietiniu aspektu. Nagrinėjamo vaizdo dalyje vietinė amplitudė lieka beveik pastovi, o vietinis aspektas aprašo lopinėlio detales. Kai filtruojamo vaizdo amplitudė yra išmatuota, atitinkamas aspektas atspindi struktūrinės lopinėlio reikšmes. Toks aspektas yra vadinamas monogeniniu aspektu.

Monogeninis signalas

Monogeninis signalas [8], tai sistema, aprašanti vaizdų orientaciją ir dydį aspektinėje išraiškoje. Jis gali būti laikomas kaip analitinis dvimačio apibendrinimo signalas. Riesz'o transformacija yra viena iš funkcijų, pagal kurią galima sudaryti monogeninį signalą. Jei apibrėšime $u = (u_1, u_2)^T$ kaip pasikartojančios srities koordinatas ir $x = (x_1, x_2)^T$ vaizdo sričiai, pasikartojančios srities branduolys yra gaunamas pagal

$$H(u) = -i \frac{u}{|u|}. \quad (2.11)$$

Vaizdo srities susukimo kaukė Riesz'o transformacijai yra gaunama pagal

$$h(x) = \frac{x}{2\pi|u|^3}. \quad (2.12)$$

Tegul $f(x)$ kaip įvedamas vaizdas. Tada monogeninis signalas $f_m(x)$ yra gaunamas pagal

$$f_m(x) = f(x) + h(x) * f(x). \quad (2.13)$$

Monogeninė mastelio imtis yra konstruojama iš pakopų kintamojo s , padalinant antrąją lygybės išraišką, pavyzdžiui, su Poisson'o kauke $P(x)$ ir jos poriniu $Q(x)$ kaip ir parodysimė:

$$p(x, s) = (f * P)(x), \quad (2.14)$$

čia

$$P(x) = \frac{s}{2\pi(|P(x)|^2 s^2)^{3/2}} \quad (2.15)$$

$$q(x, s) = (f * Q)(x), \quad (2.16)$$

čia

$$Q(x) = \frac{x}{2\pi(|P(x)|^2 s^2)^{3/2}} \quad (2.17)$$

Riesz'o susukimo kaukė yra specialus $s = 0$ atvejis. Kitu atveju, pakopa gali būti pateikta pagal Log-Gabor filtro apibrėžimą bangos ilgiui $1/f_0$,

$$G(r) = \exp\left(-\frac{\left(\log\left(\frac{r}{f_0}\right)\right)^2}{2\left(\log\left(\frac{r}{f_0}\right)\right)^2}\right). \quad (2.18)$$

Pasikartojančios srities Riesz'o branduolys gali būti pakeičiamas kaip

$$H(u) = -i \frac{u}{|u|} G(|u|). \quad (2.19)$$

Dėl atskirų $u = (u_1, u_2)^T$,

$$H(u_1) = i \frac{u_1}{\sqrt{u_1^2 + u_2^2}} G\left(\sqrt{u_1^2 + u_2^2}\right) = i \cos\theta G(r) = H_1(\theta, r) \quad (2.20)$$

$$H(u_2) = i \frac{u_2}{\sqrt{u_1^2 + u_2^2}} G\left(\sqrt{u_1^2 + u_2^2}\right) = i \sin\theta G(r) = H_2(\theta, r) \quad (2.21)$$

Vaizdo srityje, turime trimatį kintamąjį $\{p(x), q_1(x), q_2(x)\}$ monogeniniam signalui atvaizduoti.

$$\begin{aligned} p(x) &= (f * G_a)(x) \\ q_1(x) &= (f * h_1)(x) \\ q_2(x) &= (f * h_2)(x) \end{aligned} \quad (2.22)$$

čia, G_a yra vaizdo srities Log-Gabor filtras. Dydis $A(x)$, orientacija $\theta(x)$ tokiu atveju monogeninis aspektas $\varphi(x)$ yra gaunamas pagal,

$$A(x) = \sqrt{p(x)^2 + q_1(x)^2 + q_2(x)^2} \quad (2.23)$$

$$\theta(x) = \tan^{-1} \frac{q_2(x)}{q_1(x)} \quad (2.24)$$

$$\varphi(x) = \tan^{-1} \frac{p(x)}{\sqrt{q_1(x)^2 + q_2(x)^2}} \quad (2.25)$$

Būdingų taškų lyginimas

Lyginant pagal monogeninius aspektus yra sukuriami tariami atitinkantys taškai tarp dviejų vaizdų. Lyginame žiūrint į taškų minimalius tarpusavio monogeninių aspektų skirtumus per visą vaizdą. Monogeninių duomenų rinkinys $\{p(x), q_1(x), q_2(x)\}$ yra naudojamas vietoje analogo $\{A(x), \varphi(x), \theta(x)\}$.

2.2.4. Požymių aptikimo ir sugretinimo metodas – SIFT

SIFT - kompiuterinės regos algoritmas, skirtas aptikti vietinius požymius vaizduose. Algoritmą 1999 m. paskelbė David Lowe [9].

Mastelių imties ekstremumų nustatymas

Pirmasis būdingų taškų aptikimo etapas yra pasikartojančių vietovių ar pakopų atpažinimas tame pačiame objekte, tačiau tarp skirtingų vaizdų [11]. Pastovių vietovių pokyčių atpažinimas vaizduose gali būti randamas ieškant stabilių požymių. Ieškoma skersai visus galimas pakopas, naudojant funkciją, vadinamą pakopų imtimi. Koederink`as ir Lindeberg`as parodė, kad po daugybė priimtų prielaidų, kaip pakopų imties branduolys, vienintelė galima funkcija yra Gausinio suliejimo. Dėl to, vaizdų pakopų imtis yra apibrėžta,

kaip funkcija $L(x, y, \delta)$, kuri yra sukurta iš kintančių-pakopų Gausinio suliejimo $G(x, y, \delta)$ susisukimo su įvestu vaizdu $I(x, y)$:

$$L(x, y, \delta) = G(x, y, \delta) * I(x, y) \quad (2.26)$$

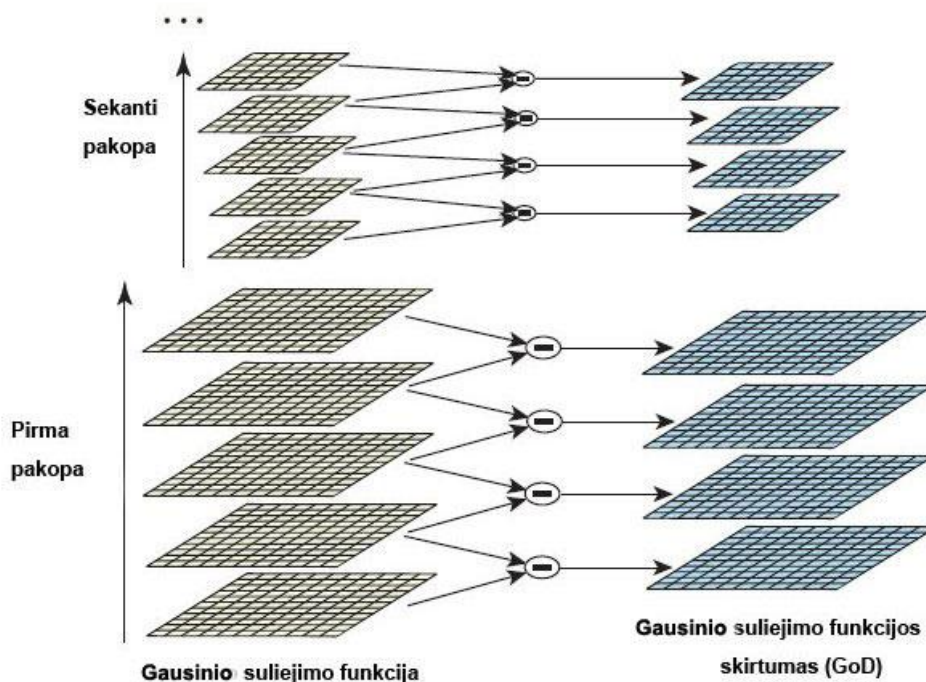
čia $*$ yra x ir y susisukimo operatorius, o

$$G(x, y, \delta) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (2.27)$$

Kad pavyktų efektyviai aptikti būdingų taškų padėtis pakopų imtyje, siūloma naudoti pakopų-imtys ekstremumus Gausinio suliejimo skirtumų (GoD) funkcijoje, kuri yra susukta su vaizdais $D(x, y, \delta)$. Vaizdai gali būti apskaičiuojami iš dviejų šalimų pakopų diferencialo atskirto pagal didinamosios konstantos faktorių k :

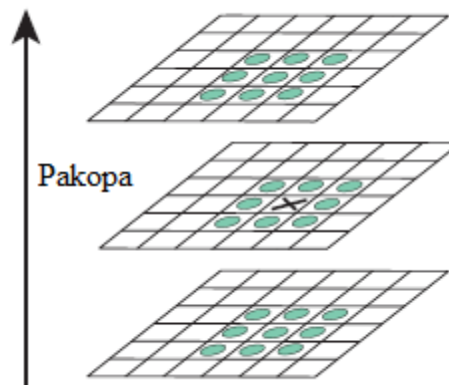
$$D(x, y, \delta) = (G(x, y, k\delta) - G(x, y, \delta)) * I(x, y) = L(x, y, k\delta) - L(x, y, \delta). \quad (2.28)$$

Yra keletas priešasčių rinktis šią funkciją. Visų pirma tai ypač efektyvi funkcija, kai sulyginami vaizdai L turi būti skaičiuojami bet kuriuo pakopų imties požymių apibūdinimo atveju ir dėl to D gali būti apskaičiuojamas kaip vaizdų skirtumas.



6 pav. Gausinio suliejimo funkcijos skirtumas

Be to, Lindeberg`as nustatė, kad Gausinio suliejimo skirtumo funkcija suteikia artimą aproksimaciją į pakopos-normalizuotą Laplaso operatoriaus Gausinę funkciją(LoG), $\sigma^2 \nabla^2 G$. Taipogi Lindeberg`as parodė, kad Laplaso operatoriaus su veiksnio σ^2 normalizavimas yra reikalaujamas teisingam pakopų pastovumui. Detaliuose palyginimų eksperimentuose Mikolajczyk`as atrado, kad $\sigma^2 \nabla^2 G$ maksimumas ir minimumas duoda stabiliausius vaizdų požymių palyginimus, pagal kitas galimas vaizdų apdorojimo funkcijas, tokias kaip Gradiento, Heseno ar Hario kampų.



6 pav. Gausinio suliejimo funkcijos skirtumo maksimumas ir minimumas yra aptinkami lyginant „x“ pažymėtą laukelį su šalia esančiais laukeliais 3x3 plotuose ir gretimomis pakopomis, kaip pažymėta paveikslėlyje.

Ryšys tarp D ir $\sigma^2 \nabla^2 G$ gali būti suprantamas iš šilumos difuzinos lygio:

$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G. \quad (2.29)$$

Iš to galime matyti, kad $\nabla^2 G$ gali būti apskaičiuotas naudojant $\partial G / \partial \sigma$ baigtinę skirtumų aproksimaciją. Naudojant šalimų pakopų skirtumus $k\sigma$ ir $n\sigma$ gauname:

$$\sigma \nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma} \quad (2.30)$$

dėl to,

$$G(x, y, k\sigma) - G(x, y, \sigma) = (k - 1)\sigma^2 \nabla^2 G. \quad (2.31)$$

Tai parodo, kad kai Gausinio suliejimo skirtumo funkcija turi pakopas, kurios skiriasi pastoviu veiksmu, tai į pakopų-pastovumo Laplaso operatoriui reikalinga σ^2 normalizacija. Veiksny $(k - 1)$, per visas pakopas yra pastovus lygtyje lygtyje, todėl neįtakoja ekstremumo taškų nustatymo. Aproximacijos klaida turėtų būti lygi 0, kai k yra lygus 1, bet praktikoje yra patikrinta, kad aproximacija neįtakoja ekstremumų taškų aptikimo ir nustatymo, netgi pastebimuose pakopų skirtumuose, tokiuose kaip $k = \sqrt{2}$.

Tikslus būdingų taškų padėties nustatymas

Pirmąjį kartą svarbūs taškai randami lyginant vaizdo tašką su jo kaimynais, sekantis žingsnis yra nustatyti artimiausią duomenų tinkamumo vietą, pakopą ir nukrypimo santykį [11]. Tokia informacija padeda atmesti taškus su mažu kontrastu ar blogai nustatytus pagal kraštą.

Pradiniai šio metodo įgyvendinimai, lengvai nustatoma būdingų taškų vieta išsidėstyme ar pakopoje pagal centrinį modelio tašką. Vis dėlto Brown`as suformavo metodą, 3D kvadratinę funkciją, tinkančią vietinių taškų nustatymui įterpiančią maksimumą ir jo parodyti eksperimentai, kurie parodo patobulėjimą atpažinimui ir stabilumui. Jo sprendimas naudojo Teilorio plėtinį iš pakopų-imities funkcijos, $D(x, y, \sigma)$, pakeitė taip, kad modelio taško pradžia yra:

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \quad (2.32)$$

čia, D ir jo išvedimas yra nustatomas modelio taške ir $x = (x, y, \sigma)^T$ yra išvedamas nuo to taško. Ekstremumo vieta \hat{x} yra gaunama paimant šios funkcijos išvestinę pagal x ir prilyginam nuliui:

$$\hat{x} = \frac{\partial^2 D^T}{\partial x^2} \frac{\partial D}{\partial x}. \quad (2.33)$$

Kaip pasiūlė Brown ir Hessian, naudojant kaimyninių taškų skirtumus D išvestinė yra gaunama apytikrė. 3x3 linijinės sistemos rezultatas gali būti gaunamas atliekant mažai skaičiavimų. Jei \hat{x} atsvara yra didesnis nei 0.5 nepriklausomai nuo sistemos dydžio, tai reiškia, kad atsižvelgiant į modelio tašką, ekstremumai gali būti neteisingi. Dėl to, modelio

taškas yra pakeičiamas ir vietoje to yra vykdoma interpoliacija. Galutinė \hat{x} atsvara yra pridedama į modelio taško vietą, kad gauti interpoliacijos skaičiavimus vietos ekstremumo taškams.

Ekstremumų funkcijos reikšmė $D(\hat{x})$ yra gerai tinkanti nestabilių ekstremumo taškų atmetimui, kai kontrastas yra mažas. Tai gali būti gaunama, pakeičiant (2.33) lygtį pagal

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \quad (2.34)$$

gauname:

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x}. \quad (2.35)$$

Būdingų taškų su mažu kontrastu atmetimas

Kad atmesti mažo kontrasto taškus, skaičiuosime antrosios eilės Teiloro plėtinio $D(x)$ reikšmę, kuri skaičiuojama kompensuojant \hat{x} . Jeigu gauta reikšmė yra mažesnė už 0.03, galimi taškai atmetami. Kitu atveju paliekami su galutine vieta $y + \hat{x}$ ir pakopa, σ , kur y yra pradinė būdingų taškų vieta pakopoje σ .

Kraštų redagavimo pašalinimas

Dėl stabilumo, neužtenka vien atmesti būdingus mažo kontrasto taškus. Gauso suliejimo funkcijos skirtumas turi stiprų kraštų redagavimą, netgi jei vieta pagal kraštus yra sunkiai nustatoma ir dėl to nestabili mažiems trikdžiams.

Prastai apibrėžtas Gausinio suliejimo funkcijos skirtumų maksimalus taškas turi didelę svarbą netikslumui visuose kraštuose, bet minimalusis taškas tik statmenose kryptyse. Pagrindiniai netikslumai gali būti apskaičiuojami iš 2×2 Hessian matricos H , apskaičiuojami būdingų taškų vietovėje ar pakopoje:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}. \quad (2.36)$$

Išvestinė yra skaičiuojama paimant kaimyninių modelio taškų skirtumus.

Tikrinė H vertė yra proporcinga pagrindiniam D kreivumui. Pasiskolinant Hario ir Stephens'o siūlytą sprendimą, galime išvengti tikrinių reikšmių skaičiavimo, nes mus domina tik jų santykis. Tegul α bus didžiausia tikrinė reikšmė, o β - mažiausia. Tuomet, iš H galime apskaičiuoti tikrinių reikšmių sumą ir jų rezultatą iš determinanto:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta, \quad (2.37)$$

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta. \quad (2.38)$$

Mažai tikėtina, kad netikslumai turės skirtingus žymėjimus, o determinantas bus neigiamas, todėl taškas yra atmetamas kaip nepriklausantis ekstremumams. Tarkime r bus santykis tarp didžiausios ir mažiausios tikrinių netikslumo reikšmių ir yra žemiau ribos, tai, kad $\alpha = r\beta$. Tuomet,

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r + 1)^2}{r}, \quad (2.39)$$

kuris priklauso ne nuo savo individualių reikšmių, o nuo tikrinių reikšmių santykių. Kiekis $(r + 1)^2/r$ yra minimalus, kai dvi tikrinės reikšmės yra lygios ir jis didėja su r . Dėl to, tam, kad patikrintume svarbiausias nukrypimus, mažesnius nei kai kurios ribinės vertės r , mums tik reikia patikrinti

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r + 1)^2}{r}. \quad (2.40)$$

Šiuo efektyviu skaičiavimu, galime atlikti skaičiavimus su mažiau nei 20 nepastovių taško operacijų reikalingų patikrinti kiekvienam būdingam taškui.

2.2.5. Tinkamų taškų atrinkimo metodas – RANSAC

Atsitiktinio modelio ir vienodumo algoritmas (RANSAC) pirmiausia buvo pristatytas Fischler'io ir Bolles'o 1981 metais, kaip metodą, kuris skaičiuoja tam tikro modelio parametrus, pradėdamas nuo išsibarsčiusių duomenų rinkinio.

RANSAC yra algoritmas iš esmės sudarytas iš dviejų žingsnių, kurie yra kartojami pagal iteracijas.

- *Hipotezės iškėlimas.* Pirmiausia atsitiktiniu būdu yra parenkamas minimalus pavyzdinis rinkinys (MSSs) iš duomenų rinkinio, toliau, naudojant vien MSS yra apskaičiuojami modelio parametrai. MSS reikšmė yra pakankamai maža, lyginant su parametrais.
- *Testavimas.* Antrajame RANSAC etape, patikriname duomenų rinkinio elementai yra nuoseklūs iliustruojamam modeliui iš pirmojo žingsnio gautais parametrais. Tokių elementų rinkinys yra vadinamas vienodumo rinkiniu (CS).

RANSAC yra baigiamas skaičiuoti, kai tikimybė rasti geresnį vienodą rinkinį (CS) yra atmetama.

Modelio parametrai ir CS konstrukcija

Dabar aptarsime modelio parametrų ir CS konstrukcijos skaičiavimo žingsnius. Symbolis \hat{x} parodo x kiekio skaičiavimų reikšmę. Paduodamas N elementų duomenų rinkinys yra nurodomas pagal $D = \{d_1, \dots, d_N\}$ ir žymimas MSS su s raide. Tegul $\theta(\{d_1, \dots, d_h\})$ yra skaičiuojamo vektoriaus parametrai, naudojant duomenų rinkinį $\{d_1, \dots, d_h\}$, kur $h \geq k$ ir k yra MSS pagrindas. Modelio daugiklis M yra apibrėžiamas kaip:

$$M(\theta) \stackrel{\text{def}}{=} \{d \in \mathbb{R}^d : f_M(d; \theta) = 0\} \quad (2.41)$$

čia, θ yra vektoriaus parametras ir f_M yra lygumo funkcija, kurios nuliniam rinkinio lygiui priklauso visi taškai tinkami modeliui M su parametrų vektoriumi θ . Atsižvelgiant į įvairius $M(\theta)$ kaip atstumą nuo d į $M(\theta)$, apibrėžiame asocijuotą klaidą su d charakteristika:

$$e_M(d, \theta) \stackrel{\text{def}}{=} \min_{d' \in M(\theta)} \text{dist}(d, d') \quad (2.42)$$

čia, $\text{dist}(.,.)$ yra atitinkama atstumo funkcija. Naudojant tokį klaidų matavimą, apibrėžiame CS kaip:

$$S(\theta) \stackrel{\text{def}}{=} \{d \in D : d_M(d; \theta) \leq \delta\} \quad (2.43)$$

čia, δ yra slenkstis, kuris taipogi gali būti išvestas iš problemos bruožų ar pagal kažkokią automatinį būdą apskaičiuotą hipotezę. Ankstesniais skaičiavimais, mes norėjome nustatyti δ reikšmę trikdžių statistikoms, kurios paveikia duomenis ir atstumo funkciją, tai Euklidinė norma, kurią galime užrašyti:

$$e_M(d, \theta) = \min_{d' \in M(\theta)} \sqrt{\sum_{i=1}^n (d_i - d'_i)^2} = \sqrt{\sum_{i=1}^n (d_i - d_i^*)^2} \quad (2.44)$$

čia, d^* yra ortogonalė d projekcija, paveikta Gausinių triukšmų $\eta \sim N(0, \sigma_\eta I)$, tai $\eta = d - d^*$. Tikslas yra apskaičiuoti δ ribos reikšmes:

$$P[e_M(d, \theta) \leq \delta] = P_{inlier} \quad (2.45)$$

Atsižvelgiant į [7], 118 psl., galime išvesti lygtį:

$$P[e_M(d, \theta) \leq \delta] = P\left[\sum_{i=0}^n n_i^2 \leq \delta^2\right] = P\left[\sum_{i=0}^n \left(\frac{\eta_i}{\sigma_\eta}\right)^2 \leq \frac{\delta^2}{\sigma_\eta^2}\right] \quad (2.46)$$

ir nuo $\frac{\eta_i}{\sigma_\eta} \sim N(0,1)$, atsitiktinis kintamasis $\sum_{i=1}^n \left(\frac{\eta_i}{\sigma_\eta}\right)^2$ turi χ_n^2 išdalinimą.

Hence:

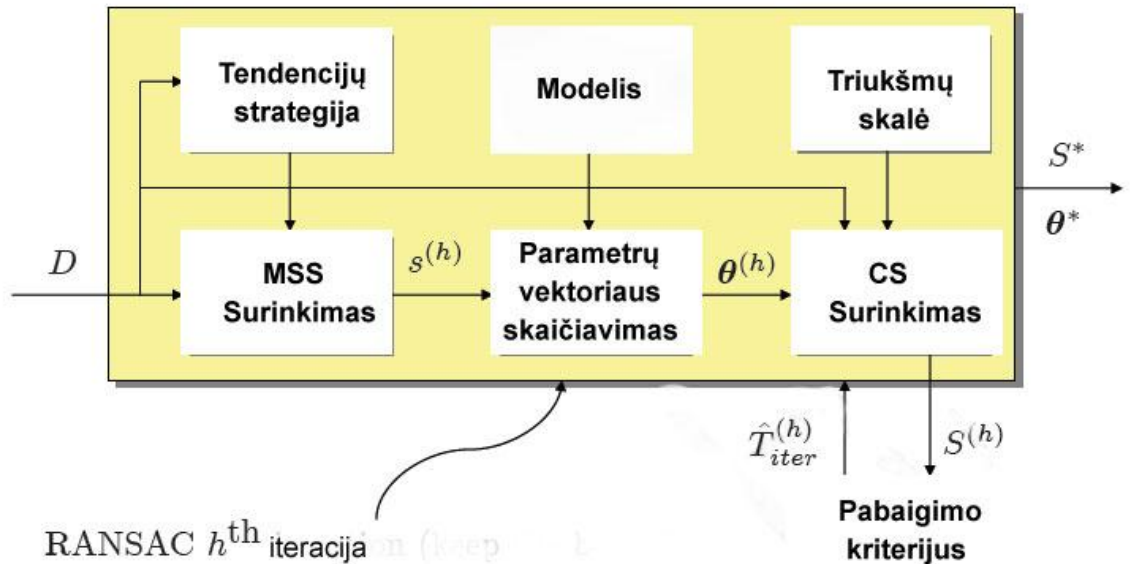
$$\delta = \sigma_n \sqrt{F_{\chi_n^2}^{-1}(P_{inlier})} \quad (2.47)$$

čia $F_{\chi_n^2}^{-1}$ didėja atvirkščiai išsibarstymo funkcijai sujungtai su χ_n^2 atsitiktiniu kintamuoju.

RANSAC esmė

Esminis RANSAC iteracijų sistemos atvaizdavimas matosi 8 pav.. Kai jau ir minėjome anksčiau, RANSAC algoritmas yra sudarytas iš dviejų žingsnių, kurie yra kartojami iteracinių stiliumi. Pirmiausia iš pradinio duomenų rinkinio yra parenkamas MSS $s^{(h)}$ ir naudojant tik MSS elementus skaičiuojami modelio, skaičiuojami modelio parametrai

$\theta^{(h)}$. Tuomet, RANSAC tikrina, kuris elementas duomenų rinkinyje D yra atitinkantis apskaičiuotus parametrus, jei sąlyga tenkina, tuomet yra atnaujinamas CSS S^* . Algoritmas yra baigiamas, kai tikimybė, kad bus rastas CS yra žemiau nustatytos slenkstinės ribos. Toliau, aptarsime, kaip nustatyti RANSAC atliekamų iteracijų skaičių.



8 pav. Esminė RANSAC iteracija

Iteracijų kiekio parinkimas

Tegul q bandymų tikimybė iš duomenų rinkinio D , MSS s kuris duoda tikslius modelio parametrų skaičiavimus. Todėl, MSS atrinkimo tikimybė yra vienas MSS, kuris sukuria šališkus modelio parametrų vektoriaus skaičiavimus yra $1 - q$. Jei sukuriame h , skirtingą nuo MSSs, tuomet, kad visi jie yra šališki tikimybė bus lygi $(1 - q)^h$. Iteracijų skaičių h norime pasirinkti kuo didesnę, tuomet iš $(1 - q)^h$ matome, kad šališkimų tikimybė yra mažesnė arba lygi tikėtinais slenkščiai ε (dažnai vadinamu signalo rodikliu) $(1 - q)^h \leq \varepsilon$. Ankstenis ryšys gali būti invertuotas, todėl galime užrašyti: kur $\lceil x \rceil$ reiškia mažiausią sveikąjį skaičių po x . Dėl to galime nustatyti:

$$\hat{T}_{iter} = \left\lceil \frac{\log \varepsilon}{\log(1 - q)} \right\rceil. \quad (2.48)$$

MSSs sudarymas ir q skaičiavimas

Tarkime, kad duomenų rinkinyje D yra taisyklingi duomenys be triukšmų, tuomet MSS sudaryti iš taisyklingų duomenų sukuria teisingų reikšmių parametrų vektorių. Jei visi duomenų rinkinio elementai turi tą pačią tikimybę, kad bus išrinkti, tuomet tikimybė, gauti MSS vien iš taisyklingų reikšmių yra:

$$q = \frac{\binom{N_I}{k}}{\binom{N}{k}} = \frac{N_I! (N - k)!}{N! (N_I - k)!} = \prod_{i=0}^{k-1} \frac{N_I - i}{N - i} \quad (2.49)$$

čia, N_I yra bendras skaičius teisingų reikšmių. Atsiminkime, kad jei $N, N_I \gg k$, tuomet q yra apytiksliai lygus k kartų, teisingų reikšmių pasirinkimui iš duomenų rinkinio. Tokiu atveju:

$$q = \prod_{i=0}^{k-1} \frac{N_I - i}{N - i} \approx \left(\frac{N_I}{N}\right)^k \quad (2.50)$$

Deja, kad apskaičiuotumėme q , mes turime žinoti N_I , kurio prioritetas yra nežinomas. Tačiau yra lengva patikrinti, kiekvienam $\hat{N}_I \leq N_I$ turime, kad $q(\hat{N}_I) \leq q(N_I)$, todėl $(1 - q(N_I))^h \geq (1 - q(\hat{N}_I))^h$. Todėl galime apskaičiuoti didžiausio teisingų reikšmių rinkinio maksimalų iteracijų skaičių, kuris gali būti laikomas tradiciniu N_I skaičiavimu. Vadinasi skaičiavimų slenkstis gali būti nustatyta:

$$\hat{T}_{iter} = \left\lceil \frac{\log \varepsilon}{\log(1 - q(\hat{N}_I))} \right\rceil \quad (2.51)$$

Vienodumo rinkinio įvertinimas

Originaliose RANSAC formuluotėse, vienodumo rinkinio (CS) įvertinimas r buvo tik kiekinis skaitvardis:

$$r(CS) \stackrel{\text{def}}{=} |CS| \quad (2.52)$$

Kitai sakant, didžiausio CSs įvertinimas yra aukščiausias. Tokiu atveju RANSAC gali būti optimizacijos algoritmas, kuris sumažina funkcijos išlaidas:

$$C_M(D; \theta) = \sum_{i=1}^n \rho(d_i, M(\theta)) \quad (2.53)$$

čia,

$$\rho(d, M(\theta)) = \begin{cases} 0 & e_M(d, \theta) \leq \delta \\ 1 & \text{otherwise} \end{cases} \quad (2.54)$$

Toks pastebėtas priartėjimas prie M-įvertinimų, suteikia idėja, kad funkciją ρ galime pakeisti į praktiškesnę:

$$\rho(d, M(\theta)) = \begin{cases} e_M(d, \theta) & e_M(d, \theta) \leq \delta \\ \delta & \text{otherwise} \end{cases} \quad (2.55)$$

Šiuo atveju naudojant mažėjančius M-įvertinimus, teisingi duomenys yra atrenkami pagal jų tinkamumą modeliui, kol išsibarstę duomenys yra gaunami tokiu pat principu. Torr`as šią RANSAC modifikaciją vadino kaip M-įvertinimų modelis ir sutapimas (MSAC).

Skaičiavimo sudėtingumas

Kiekvienai pirmojo žingsnio iteracijai mums reikia apskaičiuoti pradinis modelio parametrus iš MSS, todėl sudėtingumą aprašysime operacija $C_{skaičiavimas}(k)$.

Sekančiame žingsnyje mums reikia įvertinti kiek tinkamų duomenų yra modelyje. Jei sudėtingumas apskaičiuoti vieną elementą yra $C_{tinkamas}$, tuomet visi skaičiavimai bus $NC_{tinkamas}$.

Sudedant kartu pirmąjį ir antrąjį skaičiavimo žingsnius, galutinis sudėtingumo algoritmas būtų:

$$Sudėtingumas = (T_{iteracija} (C_{skaičiavimas}(k) + NC_{tinkamas})) \quad (2.56)$$

2.3. Vaizdų apjungimas

Vaizdų apjungimas, taip pat kaip ir vaizdų registravimas, susideda iš dviejų etapų:

- *Transformacijos modelio nustatymas* – tai vaizdų lygiavimas, gautas naudojant taškinių funkcijų tipus ir parametrus. Taškinių funkcijų parametrai yra gaunami naudojant pagal požymių panašumo nustatymą.
- *Vaizdo perrinkimas ir transformavimas* – tai vaizdo transformavimas naudojant taškines funkcijas.



9 pav. Vaizdo perrinkimas ir transformavimas

2.3.1. Homografinė ir afininė transformacijos

2D taškas (x, y) vaizde gali būti atvaizduotas kaip 3D vektorius $x = (x_1, x_2, x_3)$, kur $x = \frac{x_1}{x_3}$ ir $y = \frac{x_2}{x_3}$. Tai yra vadiname taško homogeninis vaizdavimas ir jo atvaizdavimas ant projekcinės plokštumos P^2 . Homografija yra atvirkštinis taškų ir linijų žymėjimas projekcijos plokštumoje P^2 . Homografinę transformaciją sudaro projekcinės linijinės, projekcinės ir plokštuminės projekcinės transformacijos. Hartley ir Zisserman [7] suteikė sąvokos apibrėžimą, kad homografija yra atvirkštinis žymėjimas nuo P^2 į save, jei trys taškai yra ant tos pačios linijos. Taipogi jie suteikė algebrinį apibrėžimą, įrodant šią teoremą: Žymėjimas iš $P^2 \rightarrow P^2$ yra projekcinis, jei ir tik jei egzistuoja ne vienietinė 3×3 matrica H tokia, kad kiekvienam P^2 taškui pavaizduotam vektoriuje x jo priskirti taškai yra lygūs Hx . Tai parodo,

kad norint apskaičiuoti homografiją, kuri žymi kiekvieną tašką x_i atitinkama x'_i , tai užtenka apskaičiuoti 3x3 homografinę matricą H .

Reikia pažymėti, kad sudauginant su pasirinkta nenuline konstanta, H gali būti pakeista be projektinės transformacijos pokyčių. Tokiu atveju H galime laikyti homogenine matrica, kuri turi tik 8 laisvas padėtis, net jei sudaryta iš 9 elementų. Tai reiškia, kad yra 8 nežinomieji, kuriuos reikia išspręsti.

Tipiška homografija yra skaičiuojama tarp dviejų vaizdų, randant šių vaizdų atitikimus. Dažniausiai naudojami algoritmai atitikimams naudoja taškų požymius, tačiau kitais atvejais gali būti panaudoti ir linijos ar kokios figūros.

2.3.2. Santykis su kitomis geometrinėmis transformacijomis

Norint lengvai suprasti homografijas, reikia jas panaudoti geometrinėse transformacijose. Šiame skyriuje pristatysime pagrindines homografijos transformacijas ir parodysime, kaip homografijos gali būti panaudotos šiomis paprastomis transformacijomis.

Izometrija – tai transformacija, kuri išlaiko Euklidinius atstumus. Tai reiškia, kad atstumas tarp dviejų taškų tame pačiame vaizde bus vienodas kaip atstumas tarp atitinkamų taškų sužymėtame vaizde. Tas pats galioja ir su kampais tarp linijų ir plotų. Izometrija yra sudaryta tik iš 2D pasukimų ir 2D perkėlimų. Dėl to turi tris laisvas padėtis. Izometrija gali būti užrašyta:

$$x' = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} x \quad (2.57)$$

čia R yra 2x2 pasukimo matrica, t 2-vektorių perkėlimas ir 0^T yra 2 nulinė eilutės.

Panašumų transformacija – yra panaši į izometriją, tačiau papildomai apima izotropinę skalę. Izotropinė, reiškia, kad skalė yra nekintanti kryptiniu atžvilgiu. Šioje skalėje yra pridamas papildoma laisva padėtis, tai reiškia, kad panašumų transformaciją sudaro 4 laisvas padėtis. Kaip ir izometrinėje, šioje transformacijoje kampai yra pastovūs. Atstumas tarp taškų, nebėra nekintanti, bet atstumo santykis yra išlaikytas pagal panašumų transformaciją, iki kol kompensuojami skalės pakitimai. Panašumų transformacija gali būti užrašyta:

$$x' = \begin{pmatrix} sR & t \\ 0^T & 1 \end{pmatrix} x \quad (2.58)$$

čia s yra skaliaras ir reprezentuoja izotropinę skalę.

Afininė transformacija – yra kaip panašumų transformacija, tik vietoj vieno pasukimo ir izotropinės skalės yra du pasukimai ir dvi neizotropinės skalės. Apima dviemais laisvomis padėtimis daugiau, nei panašumų transformacija: vienas pasukimo apibrėžimui mastelio keitimo srityje, kitas mastelių keitimo parametrų santykiui. Skirtingai nei panašumų transformacija, afinine transformacija nesaugo atstumų santykių ir kampų tarp linijų. Yra ir keli panašumai, lygiagrečios linijos vaizduose ir išlieka lygiagrečios po žymėjimo, taipogi lygiagrečių linijų dalių ilgių dažnis ir sritys lieka nepakitę. Afinine transformacija gali būti užrašoma:

$$x' = \begin{pmatrix} A & t \\ 0^T & 1 \end{pmatrix} x \quad (2.59)$$

čia, A yra 2x2 nevienetinė matrica.

A gali būti suskaityta kaip:

$$A = R(\theta)R(-\phi)DR(\phi) \quad (2.60)$$

čia, $R(\theta)$ ir $R(\phi)$ atitinkamai yra pasukimo matricos ir D yra diagonali matrica:

$$D = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \quad (2.61)$$

čia, λ_1 ir λ_2 galime laikyti skaliavimo reikšmėmis.

Matrica A yra sujungiamą pagal x krypties pasukimą ϕ ir skaliavimą λ_1 , y krypties skaliavimą pagal λ_2 , pasukimą atgal pagal ϕ ir pagal dar vieną pasukimą θ .

Projektinė transformacija – tai homografijos. Projektinė transformacija yra nevienetinė linijinė vienodų koordinačių transformacija. Transformacija turėtų būti nevienetinė su suvienodintomis koordinatėmis, dėka to naudojimas yra vertingas. Projektinė transformacija apima dviemais laisvomis padėtimis daugiau nei afinine transformacija. Šiuo atveju, matrica turi 9 elementus su savo santykių reikšmėmis. Nei viena iš anksčiau minėtų

afininių transformacijų, neturi projekcinio tipo, tačiau, faktas, kad jei trys taškai tame pačiame vaizde yra vienoje linijoje, jie bus kolinearūs vis dar galioja. Projektinė transformacija gali būti užrašyta:

$$x' = \begin{pmatrix} A & t \\ v^T & v \end{pmatrix} x \quad (2.62)$$

čia, $v = (v_1, v_2)^T$.

Pagrindinis skirtumas tarp afininės ir projekcinės transformacijų yra vektorius v , kuris afininiu atveju yra nulis. Šis vektorius atsakingas už projekcinius nelineinius vektorius. Kol projekcinis mastelių pokytis keičiasi su vaizdo pozicija, tai jis kiekvienoje plokštumoje nuo A yra vienodas. Panašiai, kol originalios linijos projekcinė pozicija plokštumoje turi įtakos transformuotos linijos orientacijai, tai transformuotos linijos orientacija priklauso tik nuo originalios linijos orientacijos.

Projektinė transformacija gali būti suskaidyta į anksčiau minėtas transformacijas:

$$H = H_S H_A H_P = \begin{pmatrix} sR & t \\ 0^T & 1 \end{pmatrix} \begin{pmatrix} U & 0 \\ 0^T & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ v^T & v \end{pmatrix} = \begin{pmatrix} A & t \\ v^T & v \end{pmatrix} \quad (2.63)$$

čia, H_S simbolizuoja panašumų transformaciją, H_A simbolizuoja afininę transformaciją ir H_P simbolizuoja projekcinę. $A = sRU + tv^T$ ir U yra viršutinių trikampių matrica, normalizuota, kaip determinantas $U = 1$. Kad šis suskaidymas būtų teisingas, v negali būti ligus 0. Jei s yra pažymėtas kaip teigiamas, tuomet suskaidymas yra unikalus.

2.3.3. Homografijos skaičiavimas

Šioje dalyje aptarsime homografijos taškų atitikimo skaičiavimus naudojant RANSAC algoritmą. Homografinės transformacijos funkcinė forma Euklidėse koordinatėse nelinejine išraiška užrašome:

$$T_\theta(\mathbf{y}) = \begin{bmatrix} \theta_1 y_1 + \theta_4 y_2 + \theta_7 \\ \theta_3 y_1 + \theta_6 y_2 + \theta_9 \\ \theta_2 y_1 + \theta_5 y_2 + \theta_8 \\ \theta_3 y_1 + \theta_6 y_2 + \theta_9 \end{bmatrix} \quad (2.64)$$

Taipogi prisiminkime, kad homografija yra parametrizuojama 9 skaičiais, bet turi tik 8 laisvas padėtis, kol skaliavimas projektinėje erdvėje nėra svarbus.

Homografijos parametrai yra skaičiuojami naudojant normalizuotą DLT algoritmą, apie kurį aptarsime šiek tiek vėliau.

Pagrindinis DLT algoritmas

Tiesinės linijinės transformacijos (DLT) algoritmas yra paprastas algoritmas, skirtas spręsti homografinę matricą H , kai gautas pakankamas atitinkamų taškų rinkinio kiekis.

Prieš dirbant su homogeninėmis koordinatėmis, ryšys tarp dviejų atitinkančių taškų x ir x' gali būti užrašytas kai:

$$c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = H \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (2.65)$$

čia, c yra betkokia nenulinė konstanta, $(u \ v \ 1)^T$ vaizduoja x' , $(x \ y \ 1)^T$ vaizduoja x , ir

$$H = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix}.$$

Klasifikuojant pirmąją lygties (2.65) eilutę pagal trečiąją ir antrąją, gausime dvi lygtis:

$$-h_1x - h_2y - h_3 + (h_7x + h_8y + h_9)u = 0 \quad (2.66)$$

$$-h_4x - h_5y - h_6 + (h_7x + h_8y + h_9)u = 0 \quad (2.67)$$

Lygtis (2.66) ir (2.67) gali būti užrašytos matricine forma:

$$A_i h = 0 \quad (2.68)$$

$$\text{čia, } A_i = \begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & ux & uy & u \\ 0 & 0 & 0 & -x & -y & -1 & vx & vy & v \end{pmatrix}$$

$$\text{ir } (h_1 \ h_2 \ h_3 \ h_4 \ h_5 \ h_6 \ h_7 \ h_8 \ h_9)^T$$

Nuo kiekvieno taško panašaus taško priklauso 2 lygtys. 4 panašių taškų yra gana, kad išspręstumėme 8 laisvų padėčių H . Yra vienas apribojimas, kad 3 taškai negali būti vienoje

linijoje. Keturias 2×9 A_i matricas galima sudėti į vieną, tuomet gauname vieną 8×9 matricą A . Vienam tē nulinē A erdvē, yra h erdvēs sprendimas.

Normalizacija

Hartley ir Zisserman [7] tvirtino, kad DLT algoritmo rezultatas yra priklausomas nuo vaizdo koordinačių sistemos pradžios ir skalės. Tai labai nepageidaujama ypatybė, kas algoritmą padaro ganėtinai nestabilų. Vienintelė priežastis, kodėl reikia normalizuoti, tai dėl DLT algoritmo naudojimo su vienetinės reikšmėmis A skilimo, kai perstatytos lygčių reikšmės $Ah = 0$. Daugiau yra paaiškinta [11], bet esmė yra tokia, kad esant tiksliesiems duomenims tikslus baigtinis rezultatui yra puikus, bet atsiradus trikdžiams, atsakymas dažniausiai neatitinka teisingo rezultato.

DLT taškų atitikimo variantui, Hartley ir Zisserman pasiūlė normalizaciją, kad gautas atsakymas būtų kuo teisingesnis. DLT normalizacijos algoritmas dirba pagal:

- skaičiuoja panašumo transformaciją T . Paima taškus x_i į naują taškų rinkinį \tilde{x}_i , iš taškų rinkinio \tilde{x}_i masių centrai yra pradžios koordinatės ir jų atstumų vidurkis nuo pradžios yra $\sqrt{2}$.
- apskaičiuoti panašius transformacijos T' pakeistus taškus x'_i ir \tilde{x}'_i .
- pritaikyti DLT algoritmą, naudojant \tilde{x}_i ir \tilde{x}'_i , kad gautumėme homografinę matricą H .
- nustatyti $H = (T')^{-1} \tilde{H} T$

3. Eksperimentų planavimas

Panoramiškas nuotraukų apjungimas yra kombinuojamas iš kelių metodų. Eksperimentinėje dalyje, visus apjungimo etapus tirsime atskirai. Požymių aptikimo, būdingų taškų sugretinimo, bei transformacijų skaičiavimo etapuose palyginsime skirtingus skaičiavimų metodus pagal jų greitį ir/ar kokybę. Tyrimuose naudosime dvejais būdais gautus vaizdus: slenkant fotoaparata, ir sukant apie savo ašį. Skirtingais būdais gautų vaizdų dėka galėsime tikslingiau iširti metodų veiksmingumą. Gautame galutiniame rezultate, vaizdus palyginsime vaizdų kokybės tikrinimo priemonėmis, kad patikrintumėme, kaip tiksliai veikia kombinuotas vaizdų apjungimo metodas.

Pagrindiniai tyrimų tikslai: iširti metodų privalumus ir trūkumus. Betiriant tikimasi pastebėti galimus atnaujinimus metodų kokybės ar greičio pagerinimui.

4. Eksperimentinis tyrimas

Nors jau ir minėjome metodų aprašymo srityje, tačiau pasikartosime, kad vaizdų apjungimas sudaromas iš keturių pagrindinių etapų – požymių aptikimo, požymių sugretinimo, transformacijos modelio nustatymo, bei vaizdo perrinkimo ir transformavimo.

Visi tyrimai bus atlikti naudojant *Matlab* programinį paketą.

4.1. Požymių aptikimo metodų palyginimas

Šiuo metu egzistuoja nemažai požymių aptikimo metodų, kurie yra pritaikomi skirtingose srityse. Šiuo atveju mes panaudosime du populiariausius metodus, kurių skaičiavimo principai skiriasi, tai Hario kampų, kurio skaičiavimai pagrįsti kampų aptikimu vaizduose ir SIFT metodą, pagrįstą dėmių aptikimo⁴ metodu. Kadangi SIFT algoritmas yra skirtas ne tik taškų aptikimui, bet ir palyginimui, tai šiuo atveju naudosime tik algoritmo dalį skirtą būdingų taškų aptikimui.

Abiejų metodų palyginimui naudosime tą pačią nuotrauką. Pirmiausia aptarsime Hario kampų metodą. Hario kampų metode naudojami keli parametrai, pagal kuriuos nustatome, būdingų taškų gavimo kriterijus:

sigma – Gausinio išlyginimo standartinis nuokrypis.

slenkstis - Hario kampų slenkstinis koeficientas. Kuo mažesnis koeficientas, tuo daugiau būdingų požymių aptinkame.



9 pav. Hario kampų metodu aptikti būdingi taškai. Kairėje naudojami parametrai:

sigma=1,*slenkstis*=1000, dešinėje - *sigma*=3,*slenkstis*=1000

⁴ Dėmių aptikimas – naudoja vaizdinius modelius, kurių dėka aptinkami šviesesni ar tamsesni regionai/ taškai, nei juos supantys.

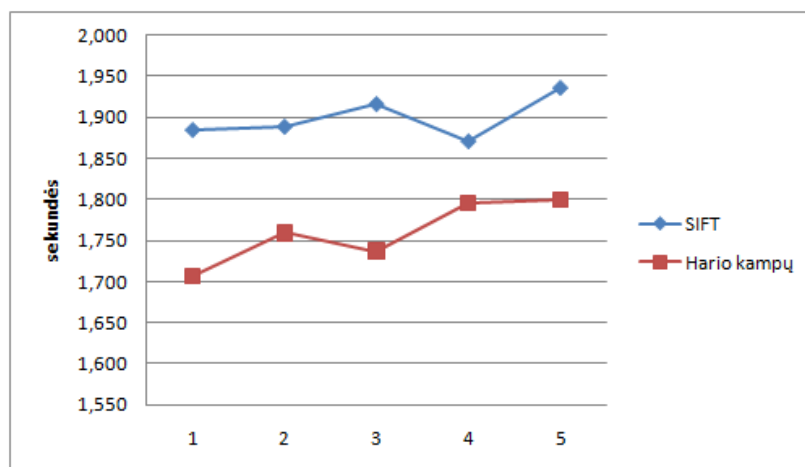
Kaip ir matosi pagal gautus rezultatus, sigma atradome toki, pagal kurį galime rasti daugiausiai būdingų taškų, kai slenkstis lieka pastovus. Kairiojoje 10 pav. dalyje rasta 1178, o dešiniajame – 467 būdingi požymiai. O mažinant sigma dar labiau, rastų būdingų taškų kiekis mažėja. Todėl tolesnius bandymus atliksime su $\sigma=1$, tik keisdami slenkstinį koeficientą.

SIFT požymių aptikimo metodo naudojime papildomi parametrai neįtraukti, todėl norėdami palyginti Hario kampų ir SIFT požymių aptikimo metodus – Hario kampų metodo papildomus parametrus nustatysime taip, kad abiem naudojamais metodais rastų būdingų taškų kiekis būtų vienodas. Tokiu atveju gauname:



11 pav. SIFT ir Hario kampų metodais aptikti būdingi taškai. Kairėje SIFT, dešinėje Hario kampų su $\sigma=1$, slenkstis=86

Parinkus tinkamą slenkstinį koeficientą, Hario metodu aptikome 2241 būdingą tašką, o SIFT metodu aptikome 2240 taškų. Kaip ir planavome, gauti būdingų taškų kiekiai apytikriai lygūs. Todėl dabar galime palyginti metodų skaičiavimo greičius, atvaizduotus 1 grafike:



1 grafikas. SIFT ir Hario kampų metodų skaičiavimo greičio palyginimas įvykdžius skaičiavimus 5 kartus

Kaip matome pagal 1 grafike pateiktus rezultatus, Hario kampų metodas dirba truputį sparčiau, nei SIFT. Nors greičių skirtumas ir nėra didelis, atliekant skaičiavimus didesnėse sistemose, greitis taptų aktualesnis.

Kiti būdingų taškų aptikimo metodų privalumai ir trūkumai išryškės, tuomet, kai vykdysime tolimesnius skaičiavimus, kur bus naudojami jau rasti būdingi taškai.

4.2. Požymių sugretinimo metodų palyginimas

Baigus požymių aptikimo etapą, sekantis etapas yra aptiktų požymių sugretinimas vaizduose turinčiuose panašumų. Šiame palyginime naudosime: SIFT (šiuo atveju pilną metodų rinkinį, kuriuo ieškosime požymių ir juos lyginsime), Hario kampų (požymių aptikimui), koreliacijos (požymių palyginimui), bei sutampančių fazių (požymių palyginimui) metodus.

Požymių sugretinimo jautrumas naudojant SIFT metodą yra apibūdinamas atstumo santykiu (nurodo maksimalų būdingų taškų vektorių kampų santykį, tarp artimiausių kaimyninių taškų), todėl prieš pradėdant palyginimus, nustatysime santykį, kuriuo aptikti požymiai bus tiksliausi.



12 pav. Požymių sugretinimas SIFT. Atstumo santykis=0.6



13 pav. Požymių sugretinimas SIFT. Atstumo santykis=0.7

Iš gautų rezultatų matosi, kad 12 pav. požymiai sugretinti tiksliau, nei 13 pav., tai atspindi linijų išsidėstymai. Paveiksluose linijos nurodo būdingų taškų aptikimą skirtingose vaizduose. Raudonais x pažymėtose linijos nuotraukose atvaizduoja klaidingai sugretintus taškus. Naudojant parametrus, kuriais išvengiama kuo daugiau neatitikimų, SIFT metodu pirmajame vaizde aptikome 2240 būdingus taškus, antrajame – 2109, tai pat aptikome 829 atitikimus, iš kurių buvo atrinkti 724 unikalūs.

Naudojant Hario kampų metodą, aptiktus būdingus taškus sugretinsime naudojant koreliacijos ir sutampančių fazių metodus parenkant parametrus, kad liktų kuo mažiau neatitikimų.



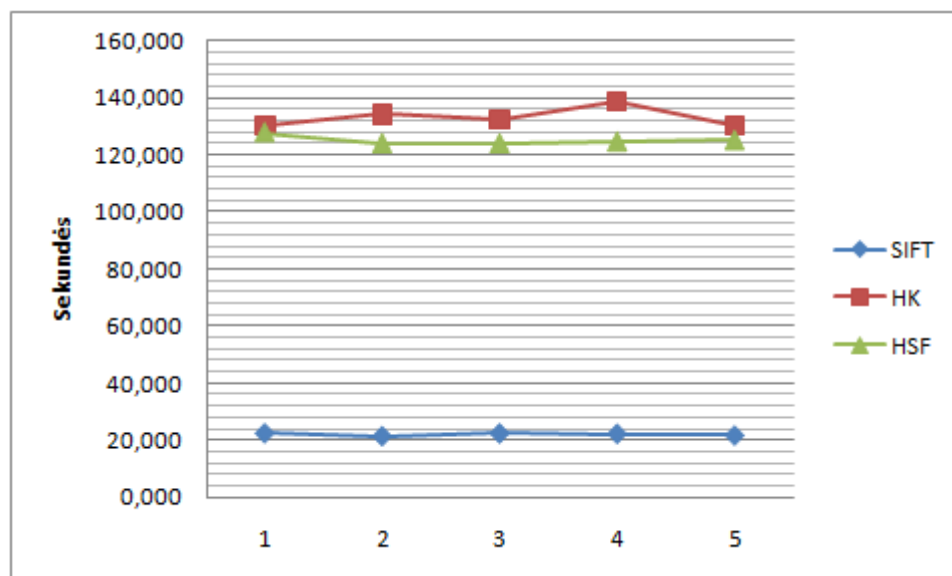
14 pav. Požymių sugretinimas naudojant koreliaciją



15 pav. Požymių sugretinimas naudojant sutampančias fazes

Naudojant Hario kampų metodą, su parametrais, kad išvengtume kuo daugiau neatitikimų, randame 467 būdingus taškus pirmajame vaizde ir 479 antrajame. Koreliacijos metodu aptinkame 266 būdingų požymių atitikimus (14 pav.), o sutampančių fazių – 259 (14 pav.). Bendri skaičiavimai užtruko 10.283 s ir 11.671 s atitinkamai.

Dabar palyginsime metodų skaičiavimo greičius. Kad galėtume tai palyginti, parinksime Hario kampų metodui parametrus kuriuos nustatėme ankstesniame palyginime, kad aptiktų būdingų taškų skaičius visais metodais būtų apytikriai lygus:



2 grafikas. SIFT, Hario kampų koreliacijos (HK) ir Hario kampų sutampančių fazių (HSF) metodų skaičiavimo greičio palyginimas sekundėmis, kai būdingų taškų kiekis visais metodais yra apytikriai lygus

Atlikus greičio matavimus matome, kad SIFT metodas smarkiai lenkia Hario kampų sukombinuoto su palyginimo koreliacijos ir sutampančių fazių metodais. SIFT metodu rasta 829, HK – 1001, o HSF – 965 būdingų taškų. Tačiau HK ir HSF metodų skaičiavimai buvo atlikti su labai daug klaidingai rastų būdingų taškų, todėl dabar ištirsime kokybinius metodų rodiklius, tam naudosime aptiktų būdingų taškų kiekio ir sutampančių taškų sandaugos santyki su skaičiavimo laiku milisekundėmis. Skaičiavimus atliksime keičiant metodų parametrus:

Metodas	Būdingi taškai	Sutampančys taškai	Skaičiavimo laikas (s)	Kokybinis rodiklis
SIFT 1	2240 ir 2109	467	21.329	46.28
SIFT 2	2240 ir 2109	829	22.003	79.56
SIFT 3	2240 ir 2109	1137	21.320	112.58
HK 1	2241 ir 1874	1001	122.911	15.28
HK 2	1352 ir 1300	708	58.654	15.72
HK 3	467 ir 479	266	10.483	12.20
HSF 1	2241 ir 1874	965	121.742	14.87
HSF 2	1352 ir 1300	679	55.254	16.00
HSF 3	467 ir 479	259	12.111	10.28

1 lentelė. SIFT, HK ir HSF metodais gauti duomenys kokybiniam metodų palyginimui

Apskaičiavus metodų kokybinius rodiklius, galime pastebėti, jog SIFT metodas smarkiai išsiskiria iš kitų dviejų metodų. Pagal gautus kokybinius rodiklius galime spręsti, kad SIFT metodo kokybinis rodiklis gaunamas tuo atveju, kai randama daugiausia sutampančių taškų, nes skaičiavimo laikas nuo to nekinta. Žiūrint į HK ir HSF metodus, matosi, kad didžiausias kokybinis rodiklis gaunamas su vidutiniu būdingų požymių, bei sutampančių taškų kiekiu. Gauti rezultatai gali kisti priklausomai nuo skaičiuojamų nuotraukų, tačiau pagal kokybinį rodiklį galime drąsiai teigti, kad SIFT metodas, nors kai kuriais atvejais yra lėtesnis, tačiau pritaikymo universalumu smarkiai pirmauja.

Sutampančių taškų rinkinių sudarymas transformacijoms

Toliau, gautus sugretintus taškus skaičiuosime iteraciniu metodu RANSAC. Kurio dėka sudarysime tinkamiausių taškų rinkinius tarp dviejų skirtingų vaizdų. Taškus atrinksime

dviem skirtingoms transformacijoms atskirai, vienas vienodų taškų rinkinys bus skaičiuosi afininei transformacijai, kitas homografini. Afininei transformacijai minimaliai užtenka trijų sutampančių taškų per abu vaizdus, o homografini reikia bent keturių. Taipogi visi trys taškai negali būti toje pačioje linijoje.



16 pav. CS taškų rinkinys afininei transformacijai



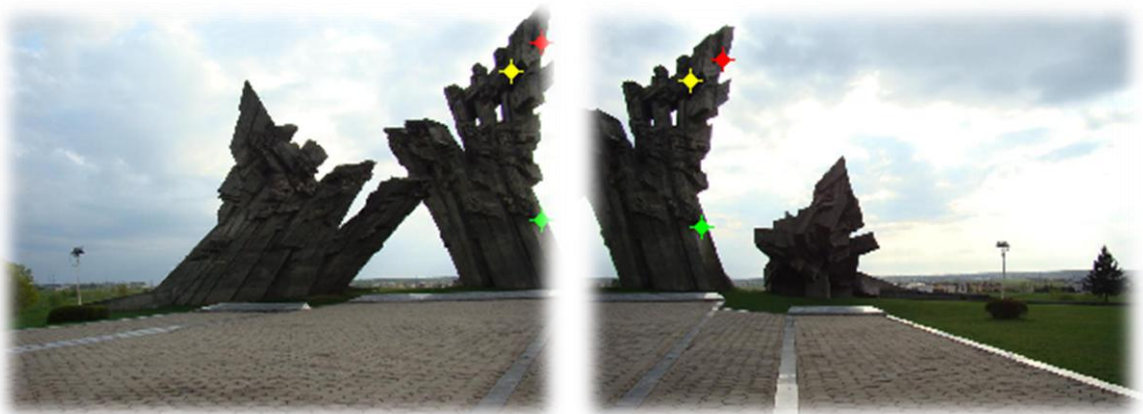
17 pav. CS taškų rinkinys homografini transformacijai

Patikrinsime kaip suvienodinto rinkinio (CS) taškų aptikimo, bei aptiktų teisingų taškų kiekis priklauso nuo iteracijų kiekio. Bandysime įvairius iteracijų kiekius, kad iširtumėme, kaip nuo to keičiasi laikas, bei teisingų reikšmių kiekis. Pirmiausia naudosime nuotraukas, gautas slenkant fotoaparata.

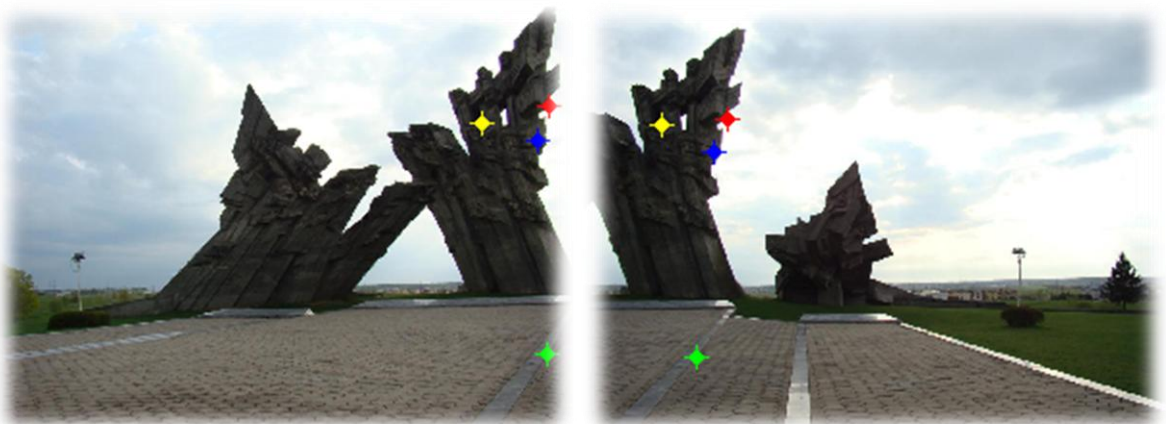
Iteracijos	Skaičiavimo laikas (s)			Viso rasta teisingų reikšmių		
	SIFT	HK	HFS	SIFT	HK	HFS
3	0.176	0.224	0.167	716	139	139
4	0.240	0.177	0.335	716	139	139
80	0.458	0.358	0.336	716	139	139

2 lentelė. Teisingų vienodų taškų skaičiavimų palyginimai, naudojant nuotraukas gautas slenkant fotoaparata

Toliau naudosime vaizdus gautus sukant fotoaparata:



18 pav. CS taškų rinkinys afininei transformacijai



19 pav. CS taškų rinkinys homografiniai transformacijai

Iteracijos	Skaiciavimo laikas (s)			Viso rasta teisingu reikšmiu		
	SIFT	HK	HSF	SIFT	HK	HSF
3	0.170	0.171	0.165	31	4	8
4	0.177	0.188	0.170	31	1	10
80	0.518	0.367	0.425	41	9	15
1600	3.271	3.135	3.508	43	12	17

3 lentelė. Teisingu vienu tašku skaičiavimų palyginimai, naudojant nuotraukas gautas sukant fotoaparata

Atlikę skaičiavimus, matome, kad kai vaizdas yra gautas slenkant fotoaparata, tai yra gautas ant tos pačios projekcijos, stabiliam CS skaičiavimui visuose metoduose užtenka vos kelių iteracijų, tačiau kai vaizdas gautas sukant fotoaparata – ant skirtingų projekcijų, visi metodai aptinka proporcingai daugiau teisingų suvienodinimo taškų. Dar pastebėjome, kad tai pirmasis atvejis, kai išryškėja HSF metodo privalumai, lyginant su HK metodu.

4.3. Transformacijų skaičiavimai

Taigi, transformacijų skaičiavimai, paskutinis tyrimo etapas, kaip jau ir minėjome anksčiau, jis susideda iš dviejų atskirų etapų: transformacijos modelio nustatymas, bei vaizdo perrinkimas ir transformavimas. Šiuo atveju transformacijos modelio nenustatinėsime, o visais skirtingais vaizdais naudosime dvi skirtingas transformacijas: afininę ir homografinę (projektinę).

Šiame tyrime, kaip ir prieš jį buvusiame, naudosime dvejais būdais gautus vaizdus, slenkant fotoaparata, bei sukant aplink jo ašį, kiekvienu būdu jungsime panoraminį vaizdą iš trijų nuoseklių vaizdų. Abejais atvejais naudosime visus anksčiau paminėtus metodus, kad vėliau galėtumėme patikrinti tikslumo skirtumus.

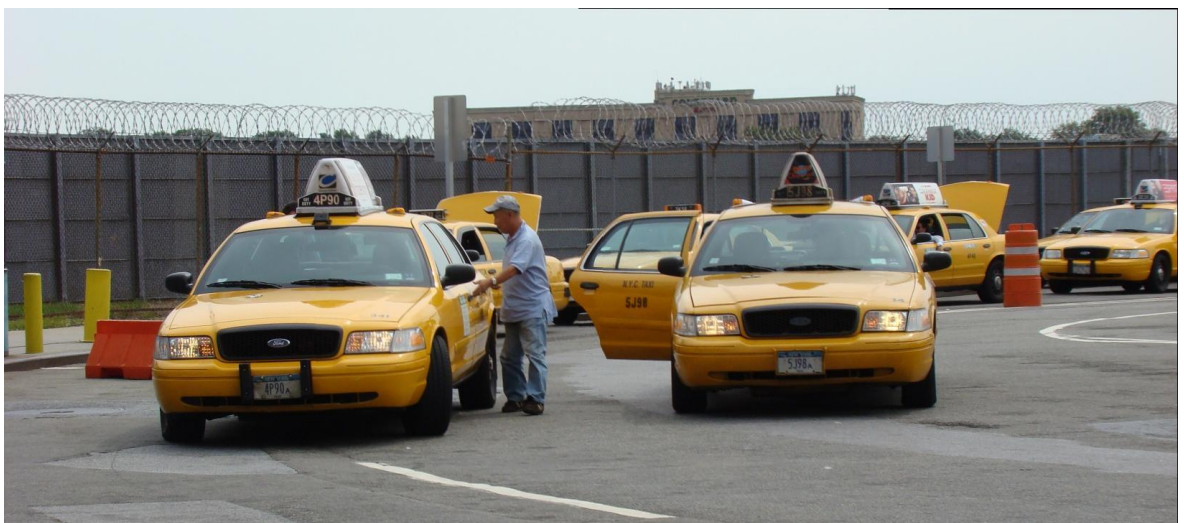
Pirmiausia nuotraukų apjungimus atliksime naudodami vaizdus gautus iš slenkant fotoaparata:



20 pav. Apjungtos nuotraukos naudojant SIFT ir afininę transformaciją



21 pav. Apjungtos nuotraukos naudojant SIFT ir homografinę transformaciją

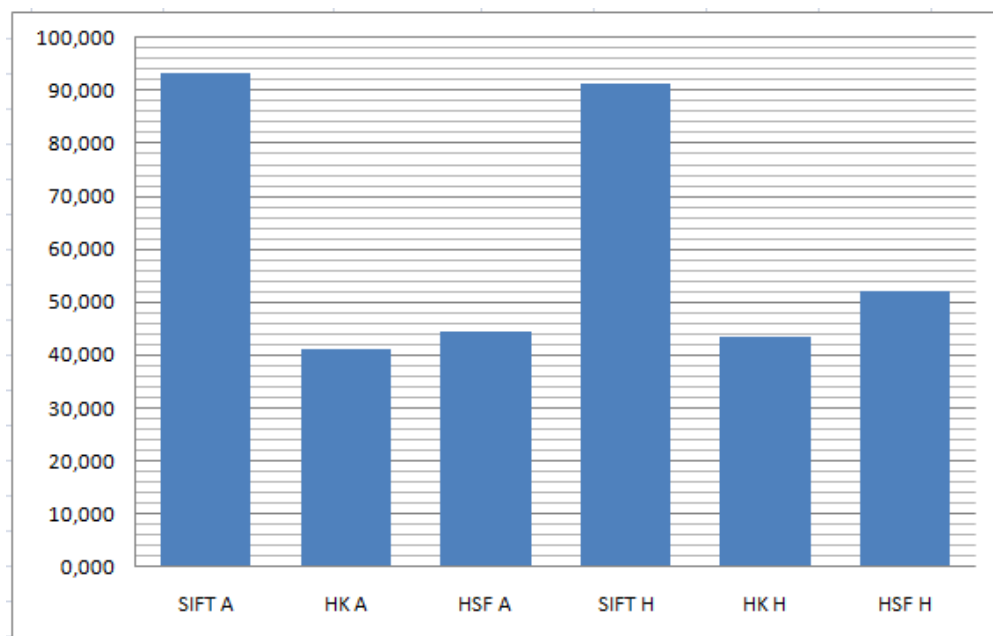


22 pav. Apjungtos nuotraukos naudojant HK-HSF ir afininę transformaciją



23 pav. Apjungtos nuotraukos naudojant HK-HSF ir homografinę transformaciją

Kaip matome iš aukščiau gautų rezultatų, kad vaizdų apjungimas, su vaizdais gautais slenkant fotokamerą yra gana paprastas. Naudojant HF ir HSF būdingų taškų aptikimo metodus, vaizdai gavosi labai panašūs, tačiau naudojant SIFT atsirado šiek tiek iškraipymų. Toliau ištirsime metodų skaičiavimo greičius. Greitį tikrinsime bendro apjungimo kombinuoto keliomis skirtingomis kombinacijomis, taip bandysime nustatyti greičiausią metodą gautų vaizdų slenkant fotokamerą skaičiavimams.



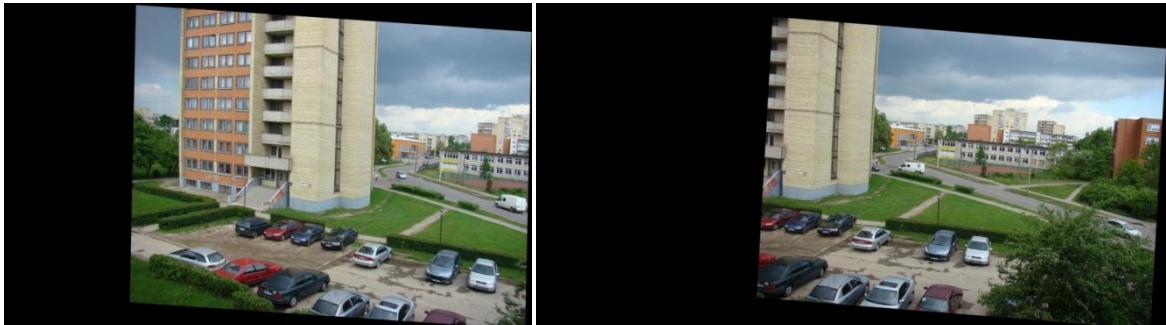
3 grafikas. Kombinuotų metodų greičiai sekundėmis. Afininė transformacija žymima (A), homografinė (H)

Apskaičiavus metodų skaičiavimo greičius aiškiai matosi, kad slenkant fotoaparata gautiems vaizdams apjungti, parankesni metodai yra HK ir HSF, kuriuos naudojant skaičiavimai atliekami apytiksliai dvigubai greičiau nei SIFT metodas.

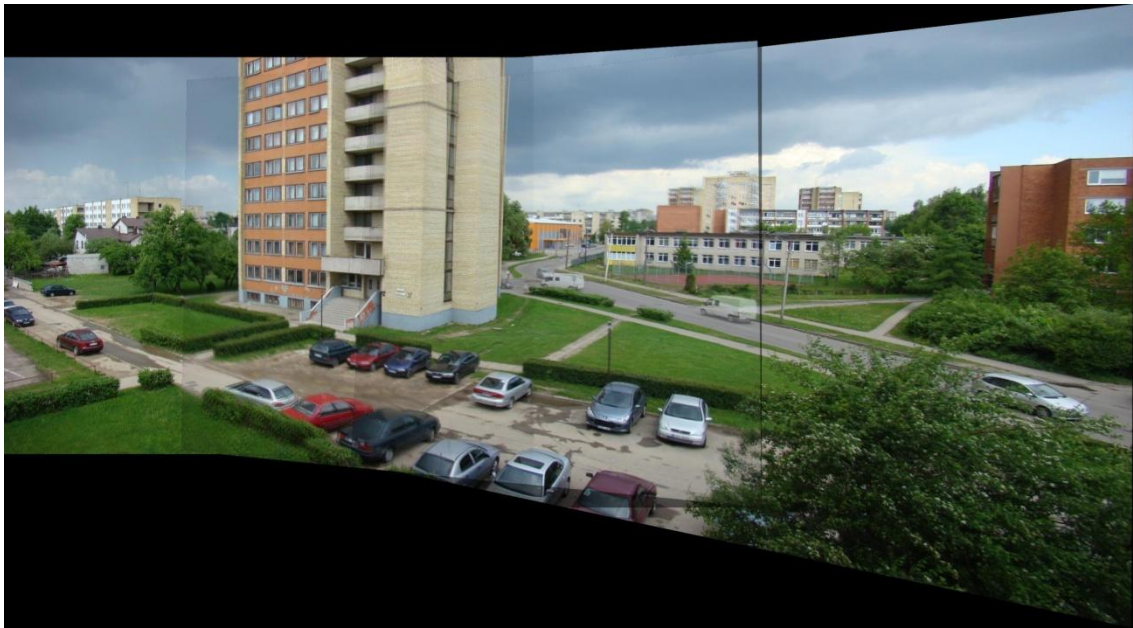
Tuos pačius tyrimus atliksime ir su vaizdais gautais sukant fotoaparata apie savo ašį:



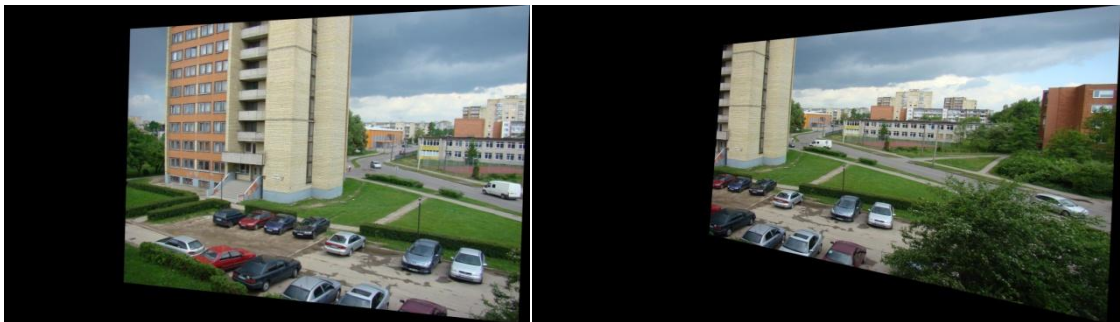
24 pav. Apjungtos nuotraukos naudojant afininę transformaciją



25 pav. Nuotraukos transformuotos afinine transformacija



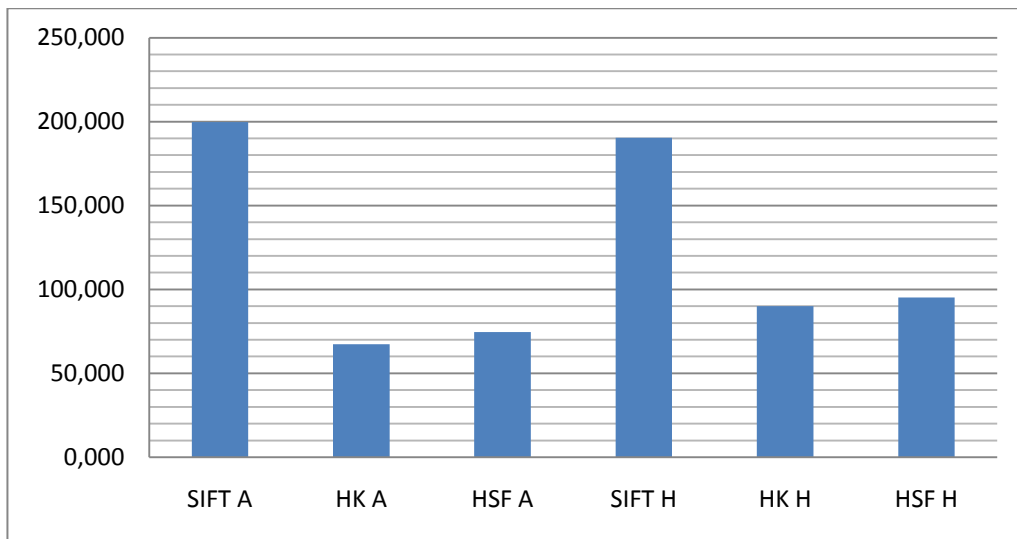
26 pav. Apjungtos nuotraukos naudojant homografinę transformaciją



27 pav. Nuotraukos transformuotos homografinė transformacija

Pagal gautus rezultatus matome, kad homografinė transformacija žymiai geriau atlieka skaičiavimus su vaizdais gautais sukant fotoaparataus apie savo ašį. Kaip ir matome iš gautų rezultatų, homografinės transformacijos pliusas tas, kad ji išlaiko atstumus iki objektų.

Kadangi gaunant pradinis vaizdus, buvo atlikti gana dideli kaimyninių nuotraukų persidengimai, vaizdų apjungimas nebuvo labai sudėtingas. Visi požymių aptikimo ir sugretinimo metodai sugebėjo rasti pakankamai būdingų taškų tarp vaizdų, todėl apjungti vaizdai gavo vienodos kokybės. Todėl, norint įvertinti metodų efektyvumą, išmatuosime metodų skaičiavimo greičius:



4 grafikas. Kombinuotų metodų greičiai sekundėmis. Afininė transformacija žymima (A), homografinė (H)

Pagal greičio skaičiavimus matome, kad skaičiavimo greičiu ir vėl nusileidžia SIFT metodas. Todėl dabar bandysime palyginti Metodų tikslumą, tam naudosime vaizdus gautus sukant fotoaparataž apie savo ašį, tačiau vaizdų persidengiantis plotas bus mažesnis nei 50 % viso vaizdo. Apjungimui naudosime tik homografinę transformaciją:



28 pav. Apjungtos nuotraukos naudojant homografinę transformaciją

Apjungimas pavyko tik SIFT metodu, kiti du metodai HK ir HSF nesugebėjo rasti pakankamai būdingų taškų, kad įvyktų vaizdų apjungimas. Tokiu atveju matome, kad SIFT metodas dirba žymiai lėčiau, tačiau stabiliau.

4.4. Vaizdo maišymas

Beveik ant visų apjungtų vaizdų matosi po persidengimo atsiradę netikslumai, kurie atsiranda dėl skirtingo ryškumo vaizduose ar dėl vaizdų transformacijų. Todėl norint juos panaikinti reikia panaudoti vaizdo maišymo algoritmą, atlikus skaičiavimus neryškūs nelygumai vaizde bus pašalinti. Jungiant du ir daugiau vaizdų, po kiekvienos poros apjungimo yra tikslinga naudoti vaizdo maišymo funkcija. Nes esant didesniai persidengimo plotui tarp vaizdų, ant ankstesnio apjungimo metu gautų vaizdų netikslumų gali atsirasti nauji netikslumai, kas gali pasunkinti netikslumų panaikinimą.



29 pav. Panoramiškai apjungtas vaizdas, po vaizdų maišymo algoritmo skaičiavimų

4.5. Vaizdo kokybės tyrimai

Gautų nuotraukų kokybės įvertinimus atliksime su vaizdais gautais slenkant fotoaparata ir sukant apie savo ašį.

Skaičiuosime šiuos parametrus: *vidutinę kvadratinę paklaidą* (MSE), *struktūrinio turinio santyki* (SC):

$$MSE = \frac{1}{MN} \sum_{j=1}^M \sum_{k=1}^N (x_{j,k} - x'_{j,k})^2 \quad (4.1)$$

$$SC = \frac{\sum_{j=1}^M \sum_{k=1}^N x_{j,k}^2}{\sum_{j=1}^M \sum_{k=1}^N x'_{j,k}^2} \quad (4.2)$$

čia, M ir N atitinkamai pradinis ir gauti vaizdai, o x ir x' jų taškai.

Abiems vaizdams parinksime pradinių vaizdų dalis:



31 pav. Nuotraukų fragmentai, pagal kuriuos bus paskaičiuoti kokybiniai vaizdų parametrai, gautų skirtingais apjungimo metodais.

Su vaizdu gautu slenkant fotokamera, gauname tokius rezultatus:

	SIFT A	SIFT H	HK A	HK H	HSF A	HSF H
MSE	167.2219	278.0300	207.3244	211.0600	296.8304	207.3244
SC	0.9993	0.9989	0.9962	0.9967	0.9963	0.9962

4 lentelė. Apjungtų vaizdų kokybiniai rodikliai slenkant fotoaparata

	SIFT A	SIFT H	HK A	HK H	HSF A	HSF H
MSE	1344.2982	1425.6923	1478.0534	1349.7718	1203.8619	1425.7185
SC	1.0271	1.0345	1.0507	1.0470	1.0528	1.0356

5 lentelė. Apjungtų vaizdų kokybiniai rodikliai sukant fotoaparata apie savo ašį

Pagal kokybinius rodiklius matome, kad pirmuoju metodu gauti vaizdai yra apjungiami apytikriai apie į kartus mažesniu nuokrypių, o struktūra pakinta labai mažai. Mažiausias nuokrypis gautas naudojant SIFT ir afininę transformaciją, didžiausias HSF ir afininę transformaciją. Antruoju atveju taipogi naudojant SIFT metodą, skaičiavimai gauti kokybiškesni. Tačiau atsimerame iš ankstesnių tyrimo rezultatų, kad SIFT metodas veikia lėčiausiai.

5. Išvados

- Atlikus tyrimus pastebėjome, kad vaizdų apjungimo procesas ir galutiniai rezultatai, labai priklauso nuo to, kaip taisyklingai buvo gauti pradiniai vaizdai.
- Po galutinių tyrimo rezultatų paaiškėjo, kad Hario kampų požymių aptikimo metodas atlieka skaičiavimu greičiau, nei SIFT metodas.
- Įvertinus būdingų taškų suliginimo kokybinius rodiklius pastebėjome, kad SIFT metodas smarkiai lekia koreliacijos ir sutampančių fazių metodą, tačiau tai pasireiškia tik mažesniuose persidengiančiuose plotuose.
- Atlikus vaizdų apjungimus, kai pradiniai vaizdai gaunami fotoaparato sukimo metodu, pastebėjome, kad skirtingai nei afininė transformacija, homografinė išlaiko atstumų santykius.
- Galutiniuose rezultatuose pastebėjome, kad nors SIFT metodas skaičiuoja žymiai lėčiau, tačiau SIFT metodu gautų vaizdų standartinis kvadratinis nuokrypis buvo mažiausias, o struktūra tiksliausia atitiko pradinį vaizdą.

6. Literatūra

1. C. Y. Chen, Image Stitching – Comparisons and New Techniques, Computer Science Department of The University of Auckland CITR at Tamaki Campus, 1998
2. R. N. Bracewell, Two dimensional imaging, Prentice-Hall, New Jersey, 1995.
3. B. Zitova, J. Flusser, Image registration methods: a survey, Department of Image Processing, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, 2003
4. B.K. Ghaffary, A.A. Sawchuk, A survey of new techniques for image registration and mapping, 1983
5. C. Harris and M.J. Stephens, A combined corner and edge detector. In Alvey Vision Conference, 1988.
6. B. Jahne, Digital Image Processing 5th ed., 2002
7. R. Hartley and A. Zisserman, Multiple view geometry in computer vision, second ed., Cambridge University Press, 2003.
8. M. Felsberg and Sommer, G. The monogenic signal. Tech. Rep. <http://citeseer.ist.psu.edu/felsberg01monogenic.html> [žiūrėta 2010 11 05]
9. David G. Lowe, Object recognition from local scale-invariant features, Proceedings of the International Conference on Computer Vision. 2, 1999
10. David G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, Computer Science Department, University of British Columbia, Vancouver, B.C., Canada, 2004
11. R. Hartley, In defense of the eight-point algorithm. In IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997
12. <http://www.mathworks.com/products/matlab/> [programinio paketo Matlab aprašymas]

7. Summary

Analysis of image stitching methods

Currently image stitching methods are widely using in in scientific researches, such as remote sensing, medical imaging, computer vision or others. So we know, that in scientific could be only two chooses, we will need methods, which can compute images with perfect quality or work fast with giant data. So we choose to compare two feature detection methods: SIFT and Harris corners. Three feature matching methods: SIFT correlation and monogenic phase. For image transformation we will use affine and homography transformations.

So we found out, that SIFT is most stable feature finding method by quality measurements. But Harris corner method works two times faster. In feature matching same as in feature detection, SIFT works better, but slower than correlation and monogenic phase. In general results we found, that SIFT method by structural content (SC) and mean square error (MSE) is a little bit more accurate than other methods.

8. Terminai

Izometrija – paviršiaus atvaizdas kitame paviršiuje, kai kreivių ilgiai nekinta.

Transformacija – formos, išvaizdos ar kitų savybių pakeitimas, pertvarkymas.

Izotropinė – vienoda visomis kryptimis.

Diagonali matrica – kvadratinė matrica, kurios visi elementai, nepriklausantys pagrindinei įsitižainei (\backslash) yra lygūs nuliui, pagrindinės elementai negali būti lygūs nuliui.

Kolinearus – esantis vienoje tiesėje arba lygiagrečiose tiesėse.

9. Sutrumpinimai ir žymėjimai

[1] – nuoroda į literatūros sąrašė esantį šaltinį

RANSAC – tinkamų taškų atrinkimo metodas

SIFT – požymių aptikimo ir sugretinimo metodas

CS – RANSAC metodu gautas taškų vienodumo rinkinys

HK – kombinuotas, Hario kampų ir koreliacijos metodas

HSF – kombinuotas, Hario kampų ir sutampančių fazių metodas