

**ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS FAKULTETAS
ELEKTRONIKOS INŽINERIJOS KATEDRA**

Sergejus Riadovikovas

**OBJEKTŲ JUDĖJIMO KRYPTIES IR SKAIČIAUS
NUSTATYMAS VAIZDO KADRUOSE**

Magistro darbas

Vadovas

dr. D. Dervinis

Konsultantas

V. Vyšniauskas

ŠIAULIAI, 2008

**ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS FAKULTETAS
ELEKTRONIKOS INŽINERIJOS KATEDRA**

TVIRTINU
Katedros vedėjas
doc. dr. G. Daunys

2008 06

**OBJEKTŲ JUDĖJIMO KRYPTIES IR SKAIČIAUS
NUSTATYMAS VAIZDO KADRUOSE**

Elektronikos inžinerijos magistro darbas

Vadovas

dr. D. Dervinis

2008 06

Recenzentas

doc. dr. G. Daunys

Atliko

RM-6 gr. stud.
S. Riadovikovas

2008 06

2008 06

ŠIAULIAI, 2008

Riadovikovas S. Determination of Number of Moving Objects and Their Movement Direction in Video. Master thesis of electronics. Engineering and research adviser dr. D. Dervinis; Electronics Department, Technology Faculty, Šiauliai University. – Šiauliai, 2008. – 55 p.

SUMMARY

In many computer vision systems it is important to classify parts of an image sequence as foreground or background. If it is possible to detect a foreground object further operations, such as recognition, identification or tracking, can be done on that object.

Background subtraction is a particularly popular method to segment foreground and background. With this method the current image is compared with reference image of the background, and then the decision is made what is background and what is not by looking for changes at each pixel.

In this thesis the adaptive background model calculation method is proposed. The key of the method is that the time of appearance of each pixel's value is stored in memory and recalled later to update the background image used in subtraction operation to compute foreground objects.

It is expected that this method will work well in ordinary image sequences where the foreground objects are the elements of urban scenery. The method probably will not work as well for objects which are of one color as the background because these pixels will be marked as background.

Riadovikovas S. Objektų judėjimo krypties ir skaičiaus nustatymas vaizdo kadruose. Elektronikos inžinerijos magistro darbas. Vadovas dr. D. Dervinis; Elektronikos inžinerijos katedra, Technologijos fakultetas, Šiaulių Universitetas. – Šiauliai, 2008. – 55 p.

SANTRAUKA

Daugumoje kompiuterinės regos sistemų yra svarbu identifikuoti kadro dalis kaip foną ar objektą. Jei yra įmanoma atskirti objektą nuo fono, tokios operacijos kaip atpažinimas, identifikavimas ir sekimas, gali būti taikomos šiam objektui.

Fono atėmimas yra gana populiarus metodas išskirti pirmame plane esančius objektus ir foną. Šiuo metodu esamas kadras palyginamas su atraminiu fono kadru, po ko priimamas sprendimas, kuris kadro taškas priklauso fonui, o kuris dominančiam objektui.

Šiame darbe siūlomas pritaikantis fono modelio apskaičiavimo metodas. Metodo esmė yra tame, kad kiekvieno taško reikšmės pasirodymo laikas yra išsaugomas atmintyje ir naudojamas fono vaizdo, naudojamo fono ir kadro skirtumo operacijoje, atnaujinimui.

Tikimasi, kad šis metodas gerai veiks vaizdo sekose, kuriose pavaizduoti miesto peizažo elementai. Šis metodas tikriausiai neduos gerų rezultatų kadruose, kur objektai yra tos pačios spalvos, kaip ir juos supantis fonas.

TURINYS

SAVOKŲ APIBRĖŽIMAS.....	8
ĮVADAS.....	9
1. TECHNOLOGIJŲ APŽVALGA.....	11
1.1. Aparatūrinė įranga.....	11
1.2. Duomenų apdorojimo procesas.....	14
1.2.1. Pradinė sistemos inicializacija.....	14
1.2.2. Raktinių sričių sekimas.....	15
1.2.3. Išsidėstymo ir formos atpažinimas.....	16
1.2.4. Veiksmo atpažinimas.....	17
2. JUDANČIŲ OBJEKTŲ APTIKIMO METODAI.....	18
2.1. Fono pašalinimo ir slenkstinės vertės taikymo metodai.....	18
2.2. Optinės slinkties metodai.....	23
2.2.1. Lucas-Kanade metodas.....	26
2.2.2. Horn ir Schunck metodas.....	28
3. ALGORITMO APRAŠYMAS.....	30
4. EKSPERIMENTO REZULTATAI.....	36
4.1. Slenkančio vidurkio, medianos, kadru skirtumo metodų tyrimas.....	36
4.2. Siūlomo metodo tyrimas.....	41
4.3. Programos sąsaja.....	51
IŠVADOS.....	53
LITERATŪRA.....	54

PAVEIKSLŲ SĄRAŠAS

1.1.1 Vienareikšmiškai atpažįstami žymekliai	12
1.1.2 Objekto aptikimas be žymeklių	13
2.1.1 Fono pašalinimo ir slenkstinės vertės taikymo metodo struktūrinė schema	20
2.2.1 Optinės slinkties metodo struktūrinė schema	26
3.1 Algoritmo struktūra	32
3.2 Algoritmo struktūrinė schema	33
3.3 Objekto judėjimo krypties nustatymas	34
3.4 Taško intensyvumo reikšmių pasirodymų skaičius	35
4.1.1 Slenkančio vidurkio metodu rastas fonas ir absoliutus skirtumas tarp rasto ir pasirinkto fono	37
4.1.2 Medianos metodu rastas fonas ir absoliutus skirtumas tarp rasto ir pasirinkto fono	39
4.1.3 Gretimų kadrų skirtumo metodas	40
4.2.1 Kadru sekos 250-asis kadras	41
4.2.2 Apskaičiuotas kadru sekos fonas	42
4.2.3 Invertuotas skirtuminis vaizdas	42
4.2.4 Invertuoto skirtuminio vaizdo eilučių vidurkio reikšmės	43
4.2.5 Siūlomu metodu apskaičiuotas fonas	43
4.2.6 Absoliutus skirtumas tarp surasto ir laisvai pasirinkto fono	44
4.2.7 Apskaičiuotas fonas	45
4.2.8 Vaizdo sekos 200-tojo kadro ir fono invertuotas skirtuminis vaizdas	47
4.2.9 Binarinis vaizdas po slenkstinės vertės taikymo	47
4.2.10 Fiksuojamų objektų skaičiaus kitimas prieš papildomą apdorojimą	48
4.2.11 Fiksuojamų objektų skaičiaus kitimas po apdorojimo	48
4.2.12 Kadras su nustatytais objektais prieš segmentų apjungimą	49
4.2.13 Kadras su nustatytais objektais po segmentų apjungimo	49
4.2.14 Kadras su „druska ir pipirai“ triukšmu	50
4.2.15 Apskaičiuotas kadru sekos su triukšmu fonas	50
4.2.16 Fono taškų klaidos kvadratas	51
4.3.1 Programos langas	52

SAVOKŲ APIBRĖŽIMAS

Objektas – bet kokio fizinio kūno, esančio kameros matomumo lauke, vaizdas.

Statinis objektas - bet kokio nejudančio fizinio kūno, esančio kameros matomumo lauke, vaizdas.

Judantis objektas - bet kokio fizinio kūno, esančio kameros matomumo lauke ir keičiančio savo padėtį erdvėje, vaizdas.

Fonas – kameros fiksuojamo vaizdo dalis, nepriklausanti judantiems objektams ir mažai arba visai nekintanti laikui bėgant.

Raktinė sritis – visumą objekto vaizdo detalių, pagal kurias galima pakankamai teisingai identifikuoti ar sekti objektą.

Žymeklis – sekamo objekto dalis arba papildomas objektas, pritvirtintas prie sekamo objekto tikslu palengvinti ieškomo objekto suradimą vaizdo kadre.

Objektų sekimo sistema – aparatūrinių ir/arba programinių priemonių, leidžiančių nustatyti norimo objekto padėtį ir/arba būseną bet kurio laiku, visuma.

Nekontroliuojama aplinka – eksperimento vykdymo sąlygų visuma, kurios dalis ar visi elementai negali būti paveikti/pakeisti jokiais būdais.

IVADAS

Automatinis judančių objektų aptikimas bei sekimas yra fundamentalūs daugelio vaizdo stebėjimo sistemų uždaviniai. Aukštesnio lygio apsaugos ir sprendimų priėmimo algoritmai remiasi šiais būtiniais vaizdo analizės uždaviniais. Judesio aptikimas vaizde leidžia greitai reaguoti į saugumo pažeidimus bei išsaugoti tai įrodančius vaizdo kadrus. Jis nėra apribotas veikimu viena kryptimi tiesia linija, kaip dauguma kitų technologijų, tokių kaip infraraudonųjų spindulių barjerai ar mikrobanginiai judesio jutikliai. Viena kamera gali saugoti didelį plotą, be to ji gali tai daryti selektyviai, tai yra reaguoti tik į pasirinktus įvykius. Naudojant „maskavimą“ (pasirenkant tik vaizdo scenos dalį ar dalis, kuriose bus įjungtas judesio aptikimas), galima sukurti atskiras zonas, kuriose bus fiksuojami skirtingi įvykiai. Dar vienas sistemos privalumas yra gebėjimas veikti toli nuo pavojingos zonos, statant kamerą saugioje vietoje. Sistema gali ženkliai sumažinti statybos kaštus. Pavyzdžiui, aptikimas gali būti atliekamas kitoje upės ar kelio pusėje be būtinybės tiesti ten kabelius ar naudoti bevielį ryšį. Efektyvus judesio aptikimas ir objekto sekimas leidžia atpažinti padidėjusį aplinkos aktyvumą, nesankcionuotą patekimą į teritoriją, judėjimą tikslu nustatyti judėjimo prigimtį, vykdyti paliktų daiktų registravimą.

Kompiuterinės regos užduotimi galima laikyti sukūrimą tokių algoritmų, kurių įėjimo duomenys yra vaizdo kadru seką, o rezultatas yra simbolinis objektų interpretavimas, pavyzdžiui, kokie objektai yra vaizdo kadre, koks jų tarpusavio išsidėstymas trimatėje erdvėje ir pan. Žmogaus regos sistema atlieka šias užduotis greitai ir nesunkiai. Žmogus gauna didžiąją dalį (apie 99%) informacijos apie jį supantį pasaulį per regos sistemą. Mes galime atpažinti objektus tūkstančiuose ar dešimtyse tūkstančių labai sudėtingų scenų. Šie protiniai gabumai yra unikalūs ir yra daug vaizdo apdorojimo sričių, kuriose gali būti pritaikytos tokios informacijos apdorojimo savybės.

Žmogus sukūrė daugybę prietaisų, padedančių gauti ir saugoti vaizdinę mus supančios aplinkos informaciją. Į kosmines misijas į kitas Saulės sistemos planetas visada imamos kameros ir apie šių misijų sėkmę mes sprendžiame iš padarytų nuotraukų. Paprasti optiniai prietaisai, tokie kaip mikroskopai ir teleskopai, leidžia mums pamatyti objektus, žymiai mažesnius ar didesnius, negu patys esame. Šiais laikais mes turime prietaisų, galinčių užfiksuoti vaizdą infraraudonajame, ultravioletiniame, rentgeno, radijo bangų diapazone. Visi šie vaizdų gavimo būdai daugiau ar mažiau reikalauja vaizdo apdorojimo proceso.

Vaizdo informacijos apdorojimas techninių požiūriu sudaro didelį iššūkį kūrėjui dėl didelio apdorojamų duomenų srauto ir naudotinių algoritmų sudėtingumo. Vaizdo apdorojimo uždaviniai yra sprendžiami įvairiais būdais, pradedant neuronų tinklais ir baigiant dirbtiniu intelektu. Tačiau kompiuterinių algoritmų šioms užduotims atlikti kūrimo tikslas dar nėra pasiektas.

Dar ankstyvaisiais 1940-taisiais metais Doug'as Engelbart'as, būdamas radarų inžinieriumi, radaro ekrane matė ne tik šviesių ir tamsių taškų visumą, bet ir plačią kompiuterių pritaikymo tokio vaizdo apdorojimui sritį, kurioje kompiuteris yra padidinto intelektualumo įrankis, įgalinantis atleisti žmogų nuo varginančio žiūrėjimo į ekraną. Tuo metu Engelbart'as dar pilnai nesuprato asmeninio kompiuterio svarbos, tačiau jo idėjos yra įkūnytos šiuolaikiniuose asmeniniuose kompiuteriuose.

Darbo tikslas:

Suprojektuoti programinę įrangą, gebančią iš vaizdo kadru sekos nustatyti objektų skaičių bei judėjimo kryptį.

Uždaviniai:

1. Išanalizuoti egzistuojančius judančių objektų aptikimo vaizdo kadruose metodus.
2. Sudaryti objektų judėjimo krypties ir skaičiaus nustatymo algoritmą.
3. Sukurti programą, naudojančią pasiūlytąjį algoritmą.

1. TECHNOLOGIJŲ APŽVALGA

1.1. Aparatūrinė įranga

Pirmos kartos vaizdo judesio aptikimo įtaisai buvo analoginiai. Jie turėjo ribotas galimybes analizuoti vaizdo kadrus ir eliminuoti melagingus suveikimus, tokius kaip medžių lapų judesys. Šių suveikimų skaičius buvo labai didelis, todėl reikėjo mažinti jautrumą iki tiek, kad reikiamo įvykio fiksavimas tapdavo abejotinas. Kai kurie įtaisai buvo tiek paprasti, kad susidėdavo iš fotoelemento, tvirtinamo prie vaizdo monitoriaus toje vietoje, kurioje buvo tikėtasi aptikti judesį. Pasikeitus tos vietos apšvietumui fotoelementas fiksuodavo aliarmą.

Keičiamo lango zonos analoginiai judesio detektoriai atlikdavo integravimo ar vidurkinimo operacijas visame lange, kad apskaičiuoti vieną intensyvumo reikšmę visam langui. Kai ši reikšmė tapdavo didesnė ar mažesnė už pasirinkto intervalo ribas, buvo fiksuojamas judesys. Ši analoginė technologija dažniausiai nenumato atsparumo vaizdo kameros vibracijoms bei natūraliam apšvietimo pasikeitimui dienos bėgyje ir turi būti naudojama griežtai kontroliuojamoje aplinkoje.

Skaitmeniniai vaizdo judesio detektoriai paprastai turi analoginius-skaitmeninius keitiklius vaizdo signalui diskretizuoti ir paversti į skaitmeninę formą. Vėliau mikroprocesorius atpažįsta judesį apdorodamas kadrą skaitmeniniame pavidale.

Didesnė skiriamoji geba arba aukštesnis vaizdo signalo diskretizavimo dažnis duoda didesnę atpažinimo tikslumą. Skaitmeninių judesio detektorių skiriamoji geba skiriasi nuo kelių iki kelių šimtų tūkstančių pikselių vienam kadru. Didesnė skiriamoji geba reiškia didesnę atstumą, platesnę apžvalgos kampą ir mažesnių bei lėčiau judančių objektų aptikimo galimybę.

Kuo didesnis reikalaujamas vaizdo judesio detektoriaus galingumas, tuo daugiau jam reikia atminties. Kuo daugiau atminties yra vaizdo kadrų saugojimui, tuo geresnė analizė tarp kadrų. Specializuotiems veikimo režimams, tokiems kaip objektų sekimas, reikalinga papildoma atmintis.

Kuo greitesnis procesorius, mikrovaldiklis ar skaitmeninis signalų procesorius, tuo daugiau pikselių gali būti apdorota per sekundę. Ši greitesnė analizė lemia didesnę skiriamąją gebą ir tikslumą. Apdorojimo greitis apibrėžiamas ne tik procesoriaus dažniu, bet ir tuo, kiek skaičiavimo operacijų yra faktiškai atliekama. Pavyzdžiui, asmeninis kompiuteris su 500 MHz procesoriumi atlieka 10 MIPS (milijonų instrukcijų per sekundę). Atskiras įrenginys gali turėti skaitmeninį signalų procesorių arba specializuotą mikroprocesorių, veikiantį 12 MHz dažniu, bet atliekanti 52 MIPS ar daugiau [1].

Skaitmeninio vaizdo judesio aptikimo privalumai:

- Viena vaizdo kamera atstoja kelis ar keliolika judesio jutiklių
- Pavojingos teritorijos gali būti stebimos nuotoliniu būdu
- Objektų, į kuriuos reaguojama, dydis gali būti keičiamas

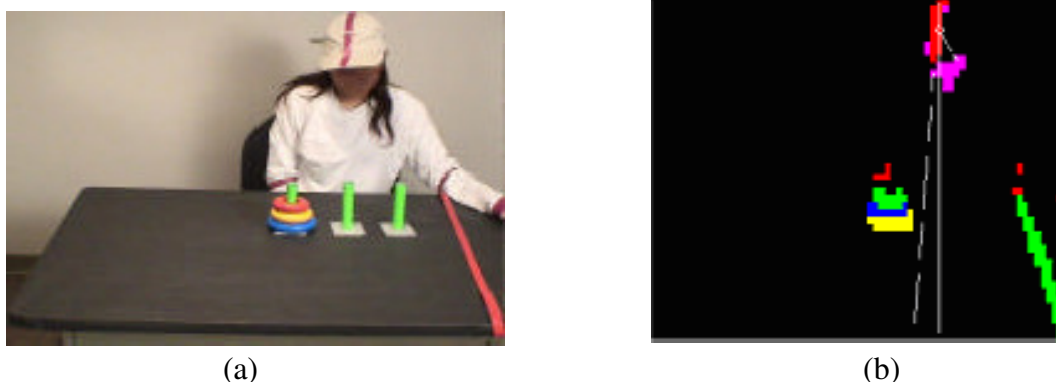
- Reguliuojamas jautrumas
- Objekto sekimo funkcija
- Objekto judėjimo krypties nustatymas
- Aptikimo zonos dydžio keitimas (pilnas vaizdas ar dalis kadro)
- Kelios aptikimo zonos vienai kamerai
- Gebėjimas adaptuotis prie apšvietimo pasikeitimų
- Kameros vibracijos aptikimas
- Gebėjimas atskirti melagingus suveikimus (vėjas, lietus, sniegas, lapai, paukščiai)
- Veikimas pagal tvarkaraštį

Šiais laikais teorinės ir praktinės galimybės leidžia realizuoti judančių objektų atpažinimo sistemas keliais būdais. Apžvelgsime jų klasifikavimą ir būdingas savybes. Pagal jutiklių panaudojimo būdą šias sistemas galima suskirstyti į aktyviasias ir pasyviasias.

- Aktyviosios sistemos naudoja jutiklius, kurie perduoda informaciją apie savo būseną. Pavyzdžiui, laivo stabilumo vandenyje analizei galima naudoti radijo jutiklį. Žmogaus judėjimo analizei gali būti naudojamas mechaninis kostiumas. Tokios sistemos išsiskiria savo paprastumu ir gautos informacijos apdorojimo tikslumu. Bet jos nevisada yra patogios naudoti, todėl kad aktyvieji jutikliai dažnai turi didelius matmenis ir gali apsunkinti pačio objekto judėjimą arba nesuteikti reikalingo informacijos apie judėjimą pilnumo. Nepaisant to, praktinės sistemų su aktyviais jutikliais realizacijos egzistuoja ir turi eilę privalumų prieš architektūrą su pasyviais jutikliais. Vienas iš didžiausių yra paprastas gaunamų duomenų apdorojimo algoritmas, ko pasekoje sistema gali pateikti mažiau apdorojimo klaidų turinčią informaciją apie objektą.
- Pasyviais vadinami jutikliai, kurie neformuoja duomenų apie tiriamo objekto judėjimą, o tik gauna kokią nors natūrinę informaciją apie jį, pavyzdžiui objekto vaizdą. Tolesnis apdorojimas vykdomas programinėmis priemonėmis. Pavyzdžiui, veido išraiškos atpažinimui būtų sunku naudoti mechaninius jutiklius, pritvirtintus ant žmogaus veido. Tokiu atveju naudoja vaizdą, gautą iš vaizdo arba infraraudonųjų kamerų. Tokio tipo sistemos dažniausiai paprastesnės aparatūrinės dalies atžvilgiu, bet matematinis duomenų apdorojimas pasidaro ženkliai sudėtingesnis, ypač tais atvejais, kai reikalingas didelis rezultatų tikslumas.

Pagal raktinių sričių atpažinimą judesio ir objektų aptikimo sistemos skirstomos į vienareikšmiško atpažinimo ir dirbtinio intelekto priemonės naudojančias sistemas. Raktinių sričių rinkiniu pavadinsime visumą objekto vaizdo detalių, pagal kurias galima pakankamai teisingai atstatyti judėjimo trajektoriją. Be to turi būti tenkinama vienareikšmiško raktinių sričių atpažinimo sąlyga. Pagal skirtingo tipo raktinių sričių naudojimą sistemas galima skirstyti į tris grupes:

- Sistemos su vienareikšmiškai atpažįstamais žymekliais. Tokiose sistemose raktinės sritys pažymimos žymekliais, kurie patys yra sritys, identifikuojamos pagal formą ar spalvą nenaudojant sudėtingų dirbtinio intelekto algoritmų (1.1.1 pav.) [2].



1.1.1 pav. Vienareikšmiškai atpažįstami žymekliai: a – objekto vaizdas, b – aptiktos raktinės sritys

- Sistemos su vienodo tipo žymekliais. Raktinės sritys, taip pat kaip aukščiau įvardintoje grupėje, pažymimos žymekliais. Bet šiuo atveju žymekliai niekuo nesiskiria tarpusavyje. Tokia architektūra leidžia ženkliai padidinti sistemos galimybes didinant žymeklių kiekį. Be to gali būti ženkliai supaprastinta aparatūrinė dalis – mažėja reikalavimai pasyviesiems jutikliams. Pavyzdžiui, galima naudoti monochromines vaizdo kameras vietoje spalvotų. Bet tokioje sistemoje atsiranda gana sudėtinga papildoma užduotis – žymeklių atpažinimas. Sistema kiekviename kadru rinkinyje turi nustatyti konkretaus žymeklio vietą, kitaip ji negalės spręsti apie objekto judėjimą netgi turint visus duomenis apie visų žymeklių išsidėstymą.
- Trečios grupės sistemos išviso nenaudoja žymeklių. Raktinių sričių atpažinimas atliekamas tiriant koreliacijų charakteristikas nuoseklyose kadruose. Analizuojamas vienaarūšių objekto vaizdo dalių tarpusavio išsidėstymas. Suprantama, tokia analizė reikalauja sudėtingesnių skaičiavimų naudojant statistikos dėsnius. Tokios sistemos privalumas yra tas, kad tampa prieinama visa informacija apie objekto struktūrą, nes nei viena jo sritis neužgožiama žymekliais. Be to bet kuri objekto sritis bet kuriuo metu gali tapti raktine, ir tam jos nereikia ženklinti žymekliais (1.1.2 pav.) [3].

Pagal laisvės laipsnių skaičių sistemos skirstomos:

- Dvimatės sistemos. Jos analizuoja objekto pasislinkimą plokštumoje. Taip realizuojamos stebėjimo sistemos, naudojančios aerofotonuotraukas. Tokiu atveju tiesiog neįmanoma susekti pakeitimų trečioje koordinatėje, todėl kad stebima plokštuma yra išsidėsčiusi statmenai regos spinduliui ir nutolusi dideliu atstumu.



1.1.2 pav. Objekto aptikimas be žymeklių

- Daugumoje atvejų naudojamos trimatės sistemos. Jos teikia detalesnę informaciją apie objektą, bet yra sudėtingesnės realizuoti.

Trečios grupės sistemos turi hibridinę architektūrą. Tokios sistemos neturi pakankamai duomenų, kad būtų įmanoma tiksliai nusakyti objektą erdvėje. Bet turint supratimą apie pačio objekto struktūrą galima atstatyti sričių tarpusavio išsidėstymą ir trečioje, nesekamoje, koordinatėje. Pavyzdžiui, galima pakankamai tiksliai atstatyti žmogaus eiseną trimačiame modelyje turint duomenis tik iš šoninės kameros, todėl kad turimi papildomi duomenys apie patį modelį.

1.2. Duomenų apdorojimo procesas

Duomenų apdorojimo procesą apibendrintoje judančių objektų sekimo sistemoje galima suskirstyti į keletą etapų:

- pradinė sistemos inicializacija;
- raktinių sričių sekimas (angl. tracking);
- išsidėstymo ir formos atpažinimas;
- veiksmo atpažinimas;
- sistemos reakcija.

1.2.1. Pradinė sistemos inicializacija

Inicializacijos etapas judėjimo analizės sistemose turi būti projektuojamas ypač kruopščiai, nes šiame etape vyksta duomenų, kuriuos vėliau naudos visos kitos posistemės, formavimas. Šis etapas apima: jutiklių derinimą ir kalibravimą, sisteminių duomenų apie jutiklių ir tiriamojo objekto pradinį išsidėstymą nustatymą, pradinės objekto būsenos nustatymą, apribojimų, pagal kuriuos sistema analizuos gautus duomenis baigiamajame etape, formavimą.

Pradinės būsenos formavimas gali vykti įvairiais būdais. Kai kuriose realizacijose egzistuoja tam tikra pradinė objekto būseną. Tokiu atveju sistema pradeda duomenų rinkimą tik po to, kai

inicializacijos posistemė praneša, kad objektas įgijo numatytą pradinę būseną. Kitu atveju sistema gali priimti bet kokią pradinę objekto būseną, bet tada būtina išanalizuoti keletą kadrų duomenų, kad įsitikinti, jog gauti duomenys apie objekto būseną yra teisingi. Kitaip galima situacija, kai sistema ilgą laiką rinks neteisingus duomenis ir negalės aptikti klaidos.

Apribojimai, kaip ir pradinė objekto būseną, gali arba būti sistemoje iš pat pradžių, arba formuoti iš transformacijų charakteristikos ir objekto judėjimo laiko intervale po pirmo sistemos paleidimo analizės.

Be aprašytų užduočių inicializacijos etape sistema gali vykdyti savo darbo testavimą. Testavimas paprastai atliekamas taip: tikrinamas jutiklių darbas, po to jie atjungiami ir į raktinių sričių aptikimo sistemos įėjimą paduodama iš anksto paruošta teisingų duomenų seka. Jei po apdorojimo gautas rezultatas atitinka turimą iš anksto žinomą teisingą rezultatą, sistema veikia teisingai. Testavimo etapas paprastai būna sistemose, kritiškose reakcijai, arba automatinėse sistemose, kuriose nėra operatoriaus, galinčio patikrinti sistemos darbą.

1.2.2. Raktinių sričių sekimas

Raktinių sričių sekimui gali būti panaudota keletas skirtingų algoritmų. Visi jie charakterizuojami sekančiu veiksmų realizavimo būdu: nuoseklus sričių išskyrimas, gautos informacijos minimizavimas ir pasikeitimų sekimo aprašymas. Du paskutiniai etapai liečia duomenų pateikimo klausimą ir yra skirtingi beveik visose realizacijose. Pirmam etapui egzistuoja keletas technologijų. Pati paprasčiausia yra jutiklių išskyrimas: jutiklis gali kartu su duomenimis perduoti savo adresą arba turėti savo dažnį, jeigu tai radijo švyturys. Pasyviose sistemose viskas yra sudėtingiau. Visų pirmą būtina išskirti analizuojamo objekto vaizdą nuo fono vaizdo. Sprendžiantys šį uždavinį algoritmai skirstomi į laikinius ir erdvinius.

Laikiniai algoritmai remiasi kelių kadrų palyginimu. Tokiu atveju objektą lengva išskirti pagal jo judėjimą jį supančios statinės aplinkos fone.

- Vaizdų atimtis – tai vienas iš laikinių algoritmų. Šis algoritmas sulygina kadrus ir pašalina iš jų vienodą informaciją, kuri su didele tikimybe yra fonas.
- Kitas laikinis sekimo algoritmas – koherentinių srautų analizė. Šiuo atveju iš vaizdo išskiriami vienas po kito segmentai ir palyginami su galimomis objekto formos detalėmis.

Erdviniai algoritmai analizuoja ne vaizdo pasikeitimus laike, bet struktūras, iš kurių jis susideda.

- Kontūro išskyrimo algoritmas remiasi objekto ir fono spalvų kontrastu. Egzistuoja du vaizdo analizės šiuo algoritmu metodai. Objektas vaizduojamas monotoniškai, paprastai mėlyna spalva bet kurios spalvos fone. Arba bet kokios spalvos objektas vaizduojamas monotoniškoje, paprastai juodoje aplinkoje.

- Į šią algoritmų grupę įeina įvairūs statistiniai objektų išskyrimo metodai. Objektas dažnai išskiriamas tam tikra seka ant jo išdėstomais spalvotais taškais arba tam tikromis formomis, būdingomis duotam objektui.

Tokiu būdu iš bendro informacinio srauto išskiriamos raktinės sritys. Jei sistema naudoja aktyviuosius jutiklius arba diferencijuojamus žymeklius, raktinių sričių sekimas šiame taške baigia darbą. Tačiau galimas vienaarūšių žymeklių naudojimas arba koreliacinis informacijos gavimas. Šiais atvejais atsiranda specifinis uždavinys – raktinių sričių palyginimas kadruose, gautuose skirtingu laiku. Šiai problemai spręsti būtina daugiau ar mažiau naudoti dirbtinį intelektą.

Aukštadažnių jutiklių panaudojimas supaprastina žymeklių lyginimo algoritmus, nes sekančiame kadre žymeklis visai nežymiai pasislenka nuo savo pirminės padėties. Tačiau ruošiami intelektualesni algoritmai, pasižymintys nepriklausomybe nuo aparatūros kokybės. Naudojamos apriorinės žinios apie objekto struktūrą. Tokiose sistemose paprastai vartojamas aktyvus objekto modelis. Įvedant į jį daugybę apribojimų galima pasiekti reikalaujamo duoto uždavinio sprendimui modelio intelektualumo.

1.2.3. Išsidėstymo ir formos atpažinimas

Ši sistemos darbo etapą galima laikyti pačiu sudėtingiausiu. Objekto formos ir transformacijos atpažinimas niekaip nepriklauso nuo jutiklių aparatūrinės realizacijos, todėl kad visi duomenys gaunami raktinių sričių išskyrimo ir sekimo etape. Jeigu objektas neturi laisvos transformacijos savybės, tai belieka sugretinti jo modelio raktinių sričių padėtį, ir bus gauti objekto forma ir jo padėtis. Objektų, galinčių keisti savo formą lokaliaje erdvėje, formos atpažinimas atliekamas šiek tiek sudėtingesnėmis priemonėmis. Egzistuoja trys metodai:

- Metodas be modelio naudojamas, kai iš sistemos nereikalaujama didelio tikslumo ir objekto transformacija turi griežtai apibrėžtas būsenas.
- Metodas su tiesioginiu modelio naudojimu yra nepakeičiamas itin sudėtingose situacijose, pavyzdžiui veido išraiškos atpažinimas, kai būtinas detalaus ir patogaus modelio sukūrimas.
- Metodas su netiesioginiu modelio naudojimu yra tarpinis variantas. Tokiu atveju sudaromas supaprastintas objekto modelis, leidžiantis atlikti jo formos analizę suvedant realias transformacijos būsenas į standartines, aprašytas modelyje.

Pats sudėtingiausias architektūros variantas – aktyvaus modelio panaudojimas. Visos tokių sistemų realizacijos skirstomos į automatines ir rankines. Automatinė sistema pati turi iširti paprasto sekimo duomenų taikymo modeliui rezultatą ir nuspręsti, ar galima tokia objekto būseną, ar buvo klaidų raktinių sričių atpažinime, ar tokia objekto būseną atitinka apskaičiuotą jo judėjimo trajektoriją. Jei sistema aptinka bet kokias klaidas ar neatitikimus, ji turi pati priimti sprendimą apie

tai, kaip galima teisingai ištaisyti duomenis. Rankinės sistemos atveju sprendimas apie gautų duomenų korektiškumą pateikiamas vartotojui ir jis pats turi įnešti atitinkamus pakeitimus matomo duomenų neatitikimo judėjimo pobūdžiui atveju.

Rankinėse sistemose pagrindine užduotimi tampa patogios ir efektyvios vartotojo sąsajos organizavimas. Turi būti apgalvota duomenų vizualizacijos procedūra, patogi interaktyvi pakeitimų įvedimo sistema.

Automatinėse sistemose gali būti realizuotos vartotojo sąsajos procedūros, tačiau pagrindinė užduotis yra intelektualios duomenų analizės posistemės projektavimas. Sistema turi būti suderinta taip, kad užtikrintų jos naudojimo galimybę sąlygomis, kurios visiškai nenumato vizualios vartotojo kontrolės.

1.2.4. Veiksmo atpažinimas

Tai baigiamoji sistemos, kaip analizatoriaus, darbo fazė. Po jos seka sistemos atsakas, jei toks numatytas. Veiksmo, kaip ir formos, atpažinimas gali būti realizuotas naudojant statistinius ar dinامينius duomenis. Sistemos, atpažįstančios objekto veiksmus pagal statistinius duomenis, palygina praeitame etape apskaičiuotą būseną su laikomais sistemoje šablonais. Tokios sistemos paprastai atpažįsta tik labai nesudėtingus veiksmus.

Dinaminio veiksmų atpažinimo sistemos dirba su duomenų seka, gauta iš formos analizatoriaus, todėl gali pažinti sudėtingus veiksmus.

Ir statistinės, ir dinaminės sistemos baigiamajame darbo etape palygina apskaičiuotus objekto veiksmus su laikomais atmintyje šablonais ir daro išvadą apie panašumą su vienu iš jų.

Kai visi susiję su analize veiksmai užbaigti ir sistema turi objekto veiksmų šabloną, lieka tik išrinkti atitinkantį šį šabloną atsaką ir atlikti atitinkamus veiksmus. Sistemos reakcija visiškai priklauso nuo jos realizacijos.

2. JUDANČIŲ OBJEKTŲ APTIKIMO METODAI

Vaizdo judesio atpažinimas yra vaizdo kameros generuojamų kadrų skaitmenizavimas ir analizė. Judesys užfiksuojamas kaip pokytis vaizdo signale lyginant su atraminiu vaizdu ar jo dalimi. Judančių objektų aptikimas su stacionaria vaizdo kamera yra pagrindinis žingsnis intelektualaus vaizdo apdorojimo procese, kadangi stebėtojo (vaizdo kameros) judėjimas nėra įvertinamas ir vaizdo struktūra gali būti žinoma nuo pat pradžios. Tačiau visvien reikalingas vaizdo apdorojimas, liečiantis staigius ar lėtus apšvietimo pasikeitimus (dėl debesų), kameros vibracijas, greitai besikeičiančius foninius objektus (tokius kaip medžių šakos ar jūros bangos), eliminuojantis šiuos objektus ir pačios kameros generuojamus triukšmus. Yra pristatyti du požiūriai į šias triukšmo pašalinimo problemas. Pirmas požiūris yra fono nustatymas ir pašalinimas, po kurio atliekamas diferencijavimas [4]. Tam kad pašalinti triukšmo efektą, slenkstinės vertės taikymas ir paprasto vaizdo apdorojimo operacijos yra atliekamos po diferencijavimo žingsnio [5]. Antras požiūris yra optinės slinkties vektorių, vadinamų judėjimo vektorių kiekviename pikselyje, apskaičiavimas ir slenkstinės vertės apskaičiuotam optinės slinkties tarp nuosekliai einančių kadrų laukui taikymas.

Abu išvardinti metodai turi privalumų ir trūkumų tikslumo, greitimeikos, atsparumo triukšmams atžvilgiu. Fono pašalinimo metodas yra paprastas ir efektyvus, bet jis gali duoti klaidingus rezultatus apdorojant duomenis su triukšmu. Kita vertus optinės slinkties metodas yra tikslesnis, bet reikalaujantis daugiau apdorojimo.

2.1. Fono pašalinimo ir slenkstinės vertės taikymo metodai

Daugelyje vaizdo apdorojimo taikymo sričių stebėtoją domina ne visas kadras, bet tik vaizdo kadro dalis. Pavyzdžiui, norint sekti objekto trajektoriją, svarbi tik ta kadro dalis, kurioje tuo metu yra objektas. Pagrindinė kliūtis išskirti šią kadro dalį yra fono išskyrimas. Išskyrus vaizdo foną, judančio objekto sekimas tampa nesunkiai įgyvendinamas. Objekto išskyrimas atliekamas generuojant atraminį fono vaizdą ir lyginant esamą kadrą su šiuo atraminiu vaizdu. Vaizdo dalys, kurios nesutampa, priskiriamos judančiam objektui.

Dažnai daug paprasčiau yra atstatyti foną, negu išfiltruoti pirmame plane esančius objektus [12]. Kai kuriais atvejais yra reikalingi tam tikri specialūs metodai, pavyzdžiui kai kurios fono dalys visada yra nematomos (t.y. kai objektas visada uždengia foną kurioje nors kadro dalyje).

Jei vaizdo kamera nejuda, vaizdo fonas yra žinomas ir sąlyginai nekintantis, esamo kadro ir fono taškų intensyvumų atėmimas daugumoje atvejų yra pakankamas nustatyti pirmame plane esančius objektus (judančius objektus).

$$|VK_i - FK_i|,$$

kur VK - vaizdo kadras,

FK - fono kadras.

Tačiau dažnai fonas nėra iš anksto žinomas ir yra kintantis dėl apšvietimo ir aplinkos sąlygų. Todėl fono nustatymas bei triukšmo pašalinimas tampa pagrindine šio metodo užduotimi.

Paprasčiausias būdas pašalinti foną yra vartotojo įsikišimas. Nors tai atrodo primityvu, tačiau tai veiksminga situacijose, kur fonas ženkliai nesikeičia ir apdorojimo laikas yra kritinis veiksnys. Šis metodas yra labai jautrus pasirinktai slenkstinei vertei, kuri bus naudojama po kadro ir fono atimties. Kitas būdas yra ankstesnio kadro kaip fono naudojimas. Šis būdas yra labai jautrus ribos parinkimui, objektų greičiui ir kadro dažniui.

$$|VK_i - VK_{i-1}| > Thr,$$

kur VK_i - esamas vaizdo kadras,

VK_{i-1} - ankstesnis vaizdo kadras,

Thr - intensyvumo slenkstinė vertė.

Spartus būdas nustatyti fono intensyvumų reikšmes yra tam tikro skaičiaus prieš tai einančių kadro vidurkio apskaičiavimas arba slenkančio vidurkio metodo naudojimas atminties sąnaudoms sumažinti [6]. Paprasto vidurkio atveju atminties sąnaudos yra proporcingos kadro skaičiui ir kadro dydžiui, tuo tarpu slenkančio vidurkio atveju atminties sąnaudos proporcingos tik kadro dydžiui. Fono nustatymas naudojant slenkančio vidurkio metodą atliekamas taip [6]:

$$BG_{i+1} = \alpha \cdot FR_i + (1 - \alpha) \cdot BG_i,$$

kur α - svorio koeficientas,

BG_i - apskaičiuotas fonas i -tajam kadru,

FR_i - i -tasis kadras.

$$\alpha = \frac{2}{N+1},$$

kur N – kadro eilės numeris.

Slenkančio vidurkio metodas gali būti naudojamas selektyviai, pavyzdžiui jei pikseliai priskiriami objektui, jie negali būti priskirti fonui:

$$BG_{i+1}(x, y) = \begin{cases} \alpha \cdot FR_i(x, y) + (1 - \alpha) \cdot BG_i(x, y) & (a) \\ BG_i(x, y) & (b) \end{cases}$$

a – jei $FR_i(x, y)$ yra vaizdo fono taškas,

b – jei $FR_i(x, y)$ yra objekto taškas.

Paprastas fono vaizdo atėmimas turi didelį skaičiavimų atlikimo greitį, bet nesugeba pateikti gerų rezultatų nekontroliuojamose aplinkose [11]. Dažniausiai atsirandančios problemos yra dėl apšvietimo lygio pasikeitimo ir laikinų fono vaizdo triukšmų, kas dažnai atsiranda lauko scenose.

Šios dvi problemos yra dažniausiai sprendžiamos sukuriant prisitaikantį fono modelį, kad jo parametrai galėtų sekti apšvietimo pasikeitimus, o taip pat padarant modelį sudėtingesniu, kad jis geriau atspindėtų tikrąjį foną.

Slenkstinės intensyvumo reikšmės taikymas

Po to kai apskaičiuojamas fonas, objekto aptikimas atliekamas atimties ir slenkstinės reikšmės taikymo operacijomis [4].

$$FR_i(x, y) \text{ yra objektas, jeigu } |FR_i(x, y) - BG(x, y)| > Thr$$

$$FR_i(x, y) \text{ yra fonas, jeigu } |FR_i(x, y) - BG(x, y)| < Thr$$

Tai yra pagrindinis būdas paversti pilkumo lygiais išreikštą vaizdą į binarinį taip, kad mus dominantys objektai būtų atskirti nuo fono. Skirtuminiame vaizde pilkumo lygiai pikselių, priklausančių objektui, turėtų skirtis nuo pikselių, priklausančių fonui. Todėl tinkamos ribos suradimas išspręstų judančio objekto aptikimo problemą. Slenkstinės reikšmės taikymo operacijos rezultatas būtų binarinis vaizdas, kurio pilkumo lygis 0 (juodas) priskiriamas fono pikseliui, o pilkumo lygis 1 (baltas) priskiriamas objektui.

Apribojimo algoritmai gali būti skirstomi į šešias grupes [7] pagal:

- Histogramos entropiją
- Histogramos formą
- Vaizdo požymį
- Pilkumo lygių klasterizavimą
- Vietines charakteristikas
- Erdvinę informaciją

Entropija yra teorinis informacijos matas, apibūdinantis tikimybinę šaltinio elgseną. Entropija paremti algoritmai naudoja objekto-fono srities entropiją arba entropiją tarp pradinio ir binarinio vaizdo. Darant prielaidą, jog kadro histograma teikia informaciją apie šią tikimybinę elgseną, siekiama entropiją padidinti, nes apriboto kadro entropijos padidėjimas reiškia maksimalų informacijos perdavimą [7].

Histogramos forma paremti metodai analizuoja kadro histogramos ekstremumus, kreivumą ir pagal šiuos morfologinius parametrus nustato naudotiną slenkstinę reikšmę. Pavyzdžiui, jei objektas aiškiai išsiskiria fone, pilkumo lygių histograma turėtų turėti du maksimumus ir riba gali būti parinkta minimumo taške tarp dviejų histogramos maksimumų.

Požymių panašumu paremti metodai parenka slenkstinę reikšmę lygindami pradinį kadrą ir jo binarinį atvaizdą. Ieškoma kraštų, kreivių, objektų skaičiaus arba sudėtingesnių loginių

panašumų. Šio metodo pavyzdys gali būti iteracinis procesas ieškant slenksčio vertės, kuri maksimizuoja atitikimą tarp kraštų pilkumo lygių vaizde ir binariniame vaizde.

Klasterizavimu paremti algoritmai pradžioje sudalina pilkumo lygių duomenis į du segmentus ir tuomet atlieka analizę. Pavyzdžiui, pilkumo lygių pasiskirstymas priimamas kaip dviejų Gauso pasiskirstymų, atitinkančių objektą ir foną, mišinys, o slenksčio vertė iteraciškai tikslinama taip, kad ji maksimizuotų šių dviejų Gauso pasiskirstymų buvimo tikimybę.

Lokaliai prisitaikantys metodai vietoje globalios slenkstinės vertės nustato vertes kiekvienam pikseliui ar jų grupei. Vietinės pikselių ar jų grupių charakteristikos, tokios kaip vidurkis, kitimas, paviršiaus parametrai, naudojamos vertėms nustatyti.

Erdviniai metodai naudoja erdvinę objekto ir fono pikselių informaciją, tokią kaip koreliacijos funkcijos, tarpusavio pasirodymo tikimybės, lokalūs tiesinės priklausomybės pikselių modeliai ir t.t.

Fono pašalinimas ir apribojimas yra pirmi žingsniai judančio objekto aptikimo vaizdo sraute link. Efektyvus fono ir ribos reikšmės nustatymas gali pašalinti didžiąją dalį esančio vaizde triukšmo, bet vistiek toks algoritmas turi turėti triukšmo pašalinimo žingsnius, kad nustatytų tikslią objekto buvimo vietą. Po fono atimties ir slenksčio taikymo žingsnio yra siūlomi naudoti paprasti vaizdo apdorojimo įrankiai, tokie kaip filtrai ir morfologiniai operatoriai. Nors būdami gana paprasti, jie ženkliai pagerina triukšmo pašalinimą ir tikslų objekto buvimo vietos nustatymą. 2.1.1 paveiksle pateikta šio metodo struktūrinė schema.



2.1.1 pav. Fono pašalinimo ir slenkstinės vertės taikymo metodo struktūrinė schema

Plėtimasis ir erozija naudojami nuosekliai kaip morfologiniai operatoriai, kad pašalinti triukšmą ir atstatyti visas objekto detales. Morfologinių filtrų kaukė gali turėti 3x3, 4x4 ar kitokią struktūrą. Daugumoje atvejų po šių operacijų vistiek lieka sričių, kuriose yra triukšmas. Dažnai tokios sritys yra mažos, todėl naudojamas filtravimas pagal dydį, kad eliminuoti melagingus objekto aptikimus.

Apžvelgsime kelis labiausiai paplitusius metodo taikymus.

Heikkila ir Olli metodas

Vaizdo taškas yra žymimas kaip objektas, jei tenkinama sąlyga [13]:

$$|I_t - B_t| > \tau,$$

kur τ yra iš anksto nustatyta slenkstinė reikšmė. Po jos taikymo seka uždarymo operacija su 3×3 lango dydžiu ir mažų sričių atmetimas. Fono vaizdas atnaujinamas pagal formulę:

$$B_{t+1} = \alpha I_t + (1 - \alpha)B_t,$$

kur α nustatomas mažas, kad išvengti „uodegų“ už judančio objekto. Yra taikomos dvi fono korekcijos:

1) Jei taškas pažymimas kaip objektas daugiau nei m kartų iš M kadru, fonas atnaujinamas taip:

$$B_{t+1} = I_t.$$

Ši korekcija skirta kompensuoti apšviestumo pasikeitimą ir naujų statinių objektų atsiradimą.

2) Jei taškas dažnai keičia priklausomybę objektui ir fonui, jis nėra įtraukiamas į objektui priklausančių taškų grupę. Tai naudojama apšviestumo svyravimų, pavyzdžiui medžių šakų siūbavimo, kompensacijai.

Prisitaikantis Gauso metodas

Kiekvienas taškas modeliuojamas atskirai [14]:

$$P(I_t) = \sum_{i=1}^K \omega_{i,t} \eta(I_t; \mu_{i,t}, \Sigma_{i,t}).$$

Priimama, kad $\Sigma_{i,t} = \sigma_{i,t}^2 I$. Prieš objekto aptikimą fonas atnaujinamas sekančiai:

1) Jei I_t yra λ standartinių nuokrypių ribose, i -toji komponentė atnaujinama sekančiai:

$$\omega_{i,t} = \omega_{i,t-1},$$

$$\mu_{i,t} = (1 - \rho)\mu_{i,t-1} + \rho I_t,$$

$$\sigma_{i,t}^2 = (1 - \rho)\sigma_{i,t-1}^2 + \rho(I_t - \mu_{i,t})^T (I_t - \mu_{i,t}),$$

kur $\rho = \alpha \Pr(I_t | \mu_{i,t-1}, \Sigma_{i,t-1})$.

2) Komponentės, neatitinkančios I_t , atnaujinamos taip:

$$\omega_{i,t} = (1 - \alpha)\omega_{i,t-1},$$

$$\mu_{i,t} = \mu_{i,t-1},$$

$$\sigma_{i,t}^2 = \sigma_{i,t-1}^2,$$

3) Jei I_t neatitinka jokių komponentių, tada mažiausiai tikėtina komponentė pakeičiama nauja, kurios $\mu_{i,t} = I_t$, $\Sigma_{i,t}$ didelis, $\omega_{i,t}$ mažas.

Po atnaujinimo $\omega_{i,t}$ svoriai yra pernормuojami. Objektų aptikimas atliekamas taip: visos komponentės rūšiuojamos mažėjančio $\omega_{i,t} / \|\Sigma_{i,t}\|$ santykio tvarka. Taip didžiausią reikšmę turinčios komponentės priskiriamos fonui. Tarkim,

$$B = \arg \min_b \left(\frac{\sum_{i=1}^b \omega_{i,t}}{\sum_{i=1}^K \omega_{i,t}} > T \right),$$

kur T yra slenkstinė vertė. Tada komponentės 1... B yra laikomos fonu. Jei I_t neatitinka nei vienos šių komponentių, taškas laikomas objekto dalimi. Objekto taškai segmentuojami į sritis naudojant artimų komponentių ženklimą. Aptiktos sritys sekamos pagal jų svorio centrą.

Pfinder metodas

Šis metodas naudoja paprastą schemą, kur fono taškai modeliuojami viena reikšme, atnaujinama pagal formulę [15]:

$$B_t = (1 - \alpha)B_{t-1} + \alpha I_t$$

Objektų taškai modeliuojami naudojant vidurkį ir kovariaciją, kurie yra cikliškai atnaujinami. Tai reikalauja tuščios scenos vaizdo inicializacijos metu.

W⁴ metodas

Taškas priskiriamas fonui, jei:

$$|M - I_t| > D \text{ arba } |N - I_t| > D,$$

kur taško parametrai M , N , D atitinka minimumą, maksimumą ir didžiausią absoliutų skirtumą tarp kadru, stebimą fono vaizde. Šie parametrai iš pradžių nustatomi iš kelių vaizdo sekos sekundžių, o vėliau periodiškai atnaujinami toms scenos dalims, kurios neturi objektų. Rezultate gaunamas pirmojo plano vaizdas yra veikiamas erozijos operacijos vieno taško triukšmui pašalinti, po to sujungtos sritys ženklina, mažos sritys atmetamos. Galiausiai likusioms sritims atliekamos plėtimosi ir erozijos operacijos [16].

Lehigh Omni-directional Tracking System (LOTS) metodas

Šio metodo bruožas yra tas, kad vienu metu yra laikomi trys fono modeliai: pirminis, antrinis ir senas fonas [17]. Jie atnaujinami taip:

1) Pirminis fonas atnaujinamas

$$B_{t+1} = \alpha I_t + (1 - \alpha)B_t,$$

jei taškas nepadarytas kaip pirmas planas, ir

$$B_{t+1} = \beta I_t + (1 - \beta)B_t,$$

jei taškas pažymėtas kaip pirmas planas. α buvo pasirinkta iš intervalo $[0,0000610351...0,25]$, su reikšme $\alpha = 0,0078125$ pagal nutylėjimą, ir $\beta = 0,25\alpha$.

2) Antrinis fonas atnaujinamas kaip

$$B_{t+1} = \alpha I_t + (1 - \alpha)B_t$$

taškuose, kur esamas vaizdas ženkliai nesiskiria nuo esamos antrinio fono reikšmės, kur α reikšmė tokia pati, kaip pirminiam fonui. Taškuose, kur yra didelis skirtumas, antrinis fonas atnaujinamas taip:

$$B_{t+1} = I_t.$$

3) Senas fonas yra vaizdo kopija iš intervalo nuo 9000 iki 18000 praėjusių kadru. Pirmo plano nustatymas remiasi prisitaikančios ribinės vertės su histereze principu. Atliekama keletas korekcijų:

- a. Maži pirmojo plano objektai pašalinami.
- b. Taškų virš ribinės vertės skaičius esamame kadre lyginamas su taškų skaičiumi ankstesniame kadre. Didelis skirtumas interpretuojamas kaip staigus apšvietimo pokytis. Reakcija į tai yra laikinas ribinės vertės padidėjimas.
- c. Taškų reikšmės pirmame plane palyginamos pirminiame ir antriniame fone. Tai pašalina artefaktus, atsirandančius dėl dalinių apšvietimo pasikeitimų ir nejudančių objektų.

Halevy metodas

Fono vaizdas atnaujinamas pagal formulę:

$$B_{t+1} = \alpha S(I_t) + (1 - \alpha)B_t,$$

kur $S(I_t)$ yra sugludinta I_t versija. Objektų taškai identifikuojami ieškant $S(I_t - B_t)$ maksimumų. Pažymima, kad $(1 - \alpha)^t < 0.1$ parodo kadru skaičių t , reikalingą fono vaizdai nusistovėti po inicializacijos [18].

Cutler metodas

Šis metodas naudoja spalvotus kadrus, nes teigiama, jog tai duoda geresnį segmentavimą, negu monochromatiniai vaizdai, ypač žemo kontrasto scenose (objektai šešėliuose). Fonas

apibrėžiamas kaip paskutinių N kadru mediana pagal laiką, pasirenkant N intervale nuo 50 iki 200. Taškai priskiriami objektui, jei

$$\sum_{C \in R, G, B} |I_t(C) - B_t(C)| > K\sigma,$$

kur σ yra nustatyta triukšmo standartinio nuokrypio vertė, K yra iš anksto parinkta konstanta [19].

Wallflower metodas

Naudojami du autoregresiniai fono modeliai:

$$B_t = -\sum_{k=1}^p a_k B_{t-k},$$

$$\hat{I}_t = -\sum_{k=1}^p a_k I_{t-k}.$$

Fono ribinė vertė:

$$\varepsilon(e_t^2) = \varepsilon(B_t^2) + \sum_{k=1}^p a_k \varepsilon(B_t B_{t-k}),$$

$$\tau = 4\sqrt{\varepsilon(e_t^2)}.$$

Taškai priskiriami fonui, jei

$$|I_t - B_t| < \tau \text{ ir } |I_t - \hat{I}_t| < \tau$$

Koeficientai a_k atnaujinami kiekvieną kadra iš stebimo fono reikšmių kovariacijų. Pritaikymui naudojamos paskutinės penkiasdešimt reikšmių trisdešimties parametru nustatymui. Jei daugiau nei 70% kadro klasifikuojama kaip pirmas planas, modelis atmetamas ir pakeičiamas atsarginiu [20].

2.2. Optinės slinkties metodai

Pagal tokį metodą apskaičiuojamas optinės slinkties laukas tarp nuosekliai einančių kadru. Savo paprasčiausioje formoje optinė slinktis seka kiekvieną pikselį iš vieno kadro į kitą. Rezultate gaunama vektorių, apibūdinančių kiekvieno taško judėjimą, seka. Judančio objekto suradimui šiam judėjimo vektorių dydžių laukui yra atliekama slenkstinės vertės taikymo operacija. Algoritmas paremtas prielaida, kad optinės slinkties lauką sukuria tik judantys objektai, kadangi netikėti aplinkos apšvietumo pokyčiai, galintys sukelti optinę slinktį, yra minimizuojami mažos laiko tarp gretimų kadru trukmės dėka.

Trys pagrindiniai optinės slinkties nustatymo algoritmai yra šie [8]:

- a) blokų atitikimo suradimas,
- b) koreliacijos dažnių srityje apskaičiavimas,

c) gradiento metodu paremtas algoritmas.

Blokų atitikimas tikrinamas taip: mažas A kadro blokas yra lyginamas su tokio pat dydžio bloku B kadre tol, kol surandamas vektorius, minimizuojantis tam tikrą paklaidos kriterijų (pavyzdžiui, blokų skirtumo absoliutinę vertę). Šio algoritmo atlikimo trukmė yra labai maža, bet dažni netikri atitikimai, todėl jis nėra tinkamas realizacijoms, kur priimami sprendimai.

Koreliacijos dažnių srityje apskaičiavimas atliekamas lyginant blokus dažnių srityje, kas duoda tikslesnį rezultatą. Būdamas labai sudėtingas, nes reikalauja daug greitų Furje transformacijų ir dauginimo operacijų, šis algoritmas nėra iteracinis ir tinka naudojimui skaitmeninių signalų procesoriuose. Furje transformacija atliekama sekančiu būdu:

$$I(k, \omega) = I(k, \omega) \exp(-i(d^T k + \Delta t \omega)),$$

čia I yra transformuotas atvaizdas, k ir ω yra dažninės komponentės. Supaprastintai galima užrašyti:

$$d^T k + \Delta t \omega = 0$$

Tai reiškia, kad kadras formuoja plokštumą dažnių srityje ir kad linija joje atitinka objekto judėjimą pastoviu greičiu.

Gradientu paremtas algoritmas iteraciškai naudoja gradiento informaciją apdorojant du duomenų blokus iš gretimų kadru. Dažnai algoritmas įvykdomas hierarchiniu būdu pradedant nuo vektoriaus dideliame pikselių bloke, palaipsniui baigiant vienu vektoriumi viename pikselyje. Kiekvienas blokas nepriklausomai nuo jo dydžio reikalauja tam tikro skaičiaus iteracijų, kas reiškia, jog tai gali būti nepriimtina laiko atžvilgiu. Tačiau tai duoda geriausius rezultatus.

Optinės slinkties metodas paremtas prielaida, jog vienas objektas scenoje gali atlikti tik vieną judesį, pavyzdžiui automobilis juda iš kairės į dešinę. Ši prielaida daro permatomų objektų ar objektų su judančiais saulės atspindžiais sekimo uždavinį sunkiai išsprendžiamu. Pavyzdžiui, scenoje, kur automobilis juda iš kairės į dešinę, o saulės atspindys ant automobilio kėbulo juda iš dešinės į kairę, vieno judesio vienam objektui prielaida žymiai palengvina apdorojimo procesą.

Labiausiai paplitęs būdas judėjimo vaizde analizei susideda iš dviejų dalių: optinės slinkties lauko apskaičiavimas ir šio lauko interpretavimas. Optinės slinkties lauko sąvoka apima ir greičio lauką, susidedantį iš vektorių, aprašančių vaizdo elementų momentinį greitį, ir pasislinkimo lauką, susidedantį iš vektorių, aprašančių vaizdo elementų judėjimą laiko tarpe tarp kadru. Toliau aptarsime optinės slinkties lauko skaičiavimus ir jo interpretavimą judančiam objektui nustatyti.

Optinės slinkties visuose kadro taškuose nustatymui naudojama intensyvumo pastovumo prielaida, kuri sako, jog bet kurio objekto bet kurio taško intensyvumas lieka pastovus laike, net tada, kai objektas juda, tai yra:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t).$$

Jei Δx ir Δy yra maži poslinkiai tarp gretimų kadru, lygtis gali būti išskleista pirmos eilės Teiloro eilute, ko pasekoje gaunama intensyvumų minimizavimo lygtis [9]:

$$I_x(x, y) \cdot u(x, y) + I_y(x, y) \cdot v(x, y) + I_t(x, y) = 0,$$

kur $I_x(x, y)$ - vaizdo intensyvumo išvestinė pagal x ašį,

$I_y(x, y)$ - vaizdo intensyvumo išvestinė pagal y ašį,

$I_t(x, y)$ - vaizdo intensyvumo išvestinė pagal laiką tarp gretimų kadru,

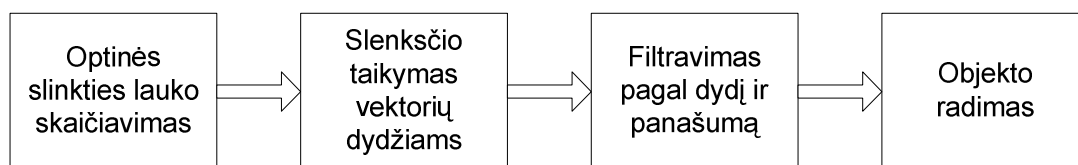
$u(x, y) = \frac{dx}{dt}$ - optinės slinkties lauko horizontali komponentė,

$v(x, y) = \frac{dy}{dt}$ - optinės slinkties lauko vertikali komponentė.

Tokie apribojimai, kaip kaimyninių pikselių vienodumas ir vieningas objekto dalių judėjimas, yra dažnai įvedami siekiant rasti lygties sprendinius, o taip pat atlikti tai greitai ir tiksliai. Pikselių vienodumo prielaida negali būti taikoma objekto kontūro pikseliams, kadangi šiuo atveju kaimyniniai pikseliai neturės vienodas šviesumo ir judesio charakteristikas.

Egzistuoja algoritmai, kurie bando lyginti kai kurias savybes, tokias kaip kampai, kontūrai, tekstūros, tarp dviejų ar daugiau kadru, kad nustatyti optinės slinkties lauką. Tačiau rezultate gaunamas laukas nėra pakankamai tankus objektų nustatymo tikslams.

Optinės slinkties lauką sudaro kiekvieno vaizdo taško judesio vektoriai. Šių vektorių dydžiai priklauso nuo judančio taško judėjimo charakteristikų. Jei taškas priklauso objektui, kiti taškai, priklausantys tam pačiam objektui, sukurs panašius optinės slinkties vektorius. To pasekoje judantys objektai sukurs panašaus judėjimo optinės slinkties vektorių blokus. Vektoriai, neįeinantys į blokus, priskiriami apšviestumo pokyčiams, kameros ar algoritmo triukšmams. Jų dydžiai turėtų būti maži ir turėtų pasireikšti nevienodumas. Judantis objektas nustatomas taikant slenksčių vektorių reikšmėms ir filtruojant pagal dydį ir panašumą. Algoritmo struktūrinė schema pateikta 2.2.1 paveiksle.



2.2.1 pav. Optinės slinkties metodo struktūrinė schema

Vektorių blokų panašumas gali būti išmatuotas panaudojus statistinius metodus, pavyzdžiui standartinį nuokrypį. Tarkime, N yra vektorių bloke skaičius. Kadangi judesio vektorius turi dvi

komponentes, horizontalią ir vertikalią, turėsime $2*N$ reikšmių. Horizontalios ir vertikalios komponentių vidurkiai ir standartiniai nuokrypiai apskaičiuojami atskirai. Panašumui nustatyti abiejų komponentių standartiniai nuokrypiai turi būti mažesni už slenkstinę vertę, kuri gali būti susieta su vektorių vidurkiu. Jei vienos iš komponentių vidurkis lygus nuliui, tuomet arba tos komponentės vektorių dydžiai artimi nuliui, arba jie yra atsitiktinai pasiskirstę. Jei jie artimi nuliui, jų standartinis nuokrypis irgi bus artimas nuliui, tai yra jie yra panašūs. Jei jų vidurkis lygus nuliui, bet turi didelį standartinį nuokrypį, tuomet komponentių blokas neturi panašumo.

Viena iš galimų lygybių, nustatančių panašumo matą tarp optinių vektorių blokų, atrodytų taip:

$$S = \frac{\sigma}{1 + \alpha * \|\mu\|},$$

kur S - panašumo matas,

σ - standartinis nuokrypis,

μ - vidurkis,

α - laisvės laipsnio koeficientas.

Iš lygties matyti, kad didžiausias panašumas pasiekiamas, kai standartinis nuokrypis lygus nuliui. Kai jo reikšmė didėja, panašumas mažėja.

Jei vaizde yra daug judančių objektų, šios reikšmės turėtų būti apskaičiuojamos kiekvienam objektui.

2.2.1. Lucas-Kanade metodas

Lucas-Kanade metodas yra vienas populiariausių judesio nustatymui. Bandoma apskaičiuoti judesį kiekviename pikselyje tarp dviejų vaizdo kadrų naudojant dalines išvestines pagal erdvines ir laikines koordinates. Tarkime, pikselis su pradinėmis koordinatėmis x, y, t ir intensyvumu $I(x, y, t)$ laiko tarpe tarp kadrų δt įgijo poslinkius $\delta x, \delta y$.

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t).$$

Išskleidžiant šią lygybę Teiloro eilute gauname:

$$I(x + \delta x, y + \delta y, t + \delta t) = I(x, y, t) + \frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t + \xi,$$

$$\xi \approx 0,$$

$$\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \delta t = 0,$$

arba

$$\frac{\partial I}{\partial x} \frac{\delta x}{\delta t} + \frac{\partial I}{\partial y} \frac{\delta y}{\delta t} + \frac{\partial I}{\partial t} = 0,$$

$$\frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} = 0,$$

kur V_x, V_y - greičio arba optinės slinkties x, y komponentės;

$\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}$ - vaizdo išvestinės atitinkamomis kryptimis.

$$I_x V_x + I_y V_y = -I_t$$

arba

$$\nabla I \cdot \vec{V} = -I_t.$$

Ši lygybė turi tris nežinomuosius ir negali būti išspręsta tokiu pavidalu. Reikia įvesti papildomus apribojimus, kad rasti lygybės sprendinius. Įvedamas pastovios slinkties apribojimas. Priimant kad slinktis (V_x, V_y) pastovi mažame lange, kurio dydis $m \times m, m > 1$, o pikselių skaičius $1 \dots n$, randamos papildomos lygtys:

$$I_{x1} V_x + I_{y1} V_y = -I_{t1}$$

$$I_{x2} V_x + I_{y2} V_y = -I_{t2}$$

⋮

$$I_{xn} V_x + I_{yn} V_y = -I_m$$

Taip gauname daugiau, negu tris lygtis trimis nežinomiesiems, todėl ši lygčių sistema turi sprendinius.

$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \vdots & \vdots \\ I_{xn} & I_{yn} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} -I_{t1} \\ -I_{t2} \\ \vdots \\ -I_m \end{bmatrix}$$

arba

$$A \vec{v} = -b.$$

Lygčių sistemos sprendimui naudojamas mažiausių kvadratų metodas:

$$A^T A \vec{v} = A^T (-b)$$

$$\vec{v} = (A^T A)^{-1} A^T (-b)$$

Išskleidus gauname:

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = \begin{bmatrix} \sum I_{xi}^2 & \sum I_{xi} I_{yi} \\ \sum I_{xi} I_{yi} & \sum I_{yi}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum I_{xi} I_{ti} \\ -\sum I_{yi} I_{ti} \end{bmatrix}$$

Tai reiškia, kad optinė slinktis gali būti rasta apskaičiuojant vaizdo išvestines visuose matmenyse. Svorio funkcija turėtų būti įterpta, kad padidinti centrinio lango pikselio reikšmę. Gauso funkcijos plačiai naudojamos šiam tikslui. Nors Lucas ir Kanade kūrė savo algoritmą

taikymui erdvinio vaizdo sekose, tačiau šiandien ši technika plačiai naudojama optinės slinkties skaičiavimuose.

2.2.2. Horn ir Schunck metodas

Horn-Schunck optinės slinkties nustatymo metodas naudoja glotnumo apribojimą, tai yra šalia esantys objekto taškai juda panašiais x ir y greičiais, arba $\|\nabla u\|^2$, $\|\nabla v\|^2$ yra maži. Horn ir Schunck pasiūlė rasti u , v , kad minimizuoti sekančią energijos funkciją [10]:

$$E(u, v) = \int [(I_x u + I_y v + I_t)^2 + \lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2)] dx dy,$$

čia λ reiškia glotnumo apribojimo svorio koeficientą.

$$L(u(x, y), v(x, y)) = [(I_x u + I_y v + I_t)^2 + \lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2)],$$

$$\nabla_u E = \frac{\partial L}{\partial u} - \frac{\partial}{\partial x} \frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y} \frac{\partial L}{\partial u_y} = I_x^2 u + I_x I_y v - I_x I_t - \lambda \nabla^2 u,$$

$$\nabla_v E = I_x I_y u + I_y^2 v - I_y I_t - \lambda \nabla^2 v.$$

Būtina sąlyga E minimizavimui išreiškiama taip:

$$\nabla_u E = 0$$

$$\nabla_v E = 0$$

Tai yra vadinama Eulerio-Lagranžo lygybe, kurią pertvarkius gauname:

$$I_x^2 u + I_x I_y v = \lambda \nabla^2 u - I_x I_t$$

$$I_x I_y u + I_y^2 v = \lambda \nabla^2 v - I_y I_t$$

Kad išspręsti šias lygtis atskirų pikselių atžvilgiu, atliekama diskretinė aproksimacija:

$$\nabla^2 u(x, y) \approx 4(\bar{u}(x, y) - u(x, y)),$$

kur $\bar{u}(x, y) = \frac{1}{4}(u(x-1, y) + u(x+1, y) + u(x, y-1) + u(x, y+1))$. Apibrėžiant $\alpha = 4\lambda$,

gauname:

$$(I_x^2 + \alpha)u + I_x I_y v = \alpha \bar{u} - I_x I_t$$

$$I_x I_y u + (I_y^2 + \alpha)v = \alpha \bar{v} - I_y I_t$$

Tai gali būti išspręsta naudojant algoritmą, susidedantį iš trijų žingsnių:

- 1) pradinėje iteracijoje ($n = 0$) priimamos pradinės u, v reikšmės;
- 2) jos yra atnaujinamos:

$$u^{n+1} = \bar{u}^n - I_x \frac{(I_x \bar{u}^n + I_y \bar{v}^n + I_t)}{\alpha + I_x^2 + I_y^2},$$

$$v^{n+1} = \bar{v}^n - I_y \frac{(I_x \bar{u}^n + I_y \bar{v}^n + I_t)}{\alpha + I_x^2 + I_y^2};$$

3) sustoti, kai E nebemažėja arba $I_x \bar{u}^n + I_y \bar{v}^n + I_t$ yra mažas.

Horn-Schunck metodo privalumas yra tas, kad rezultate gaunamas tankus slinkties vektorių laukas.

3. ALGORITMO APRAŠYMAS

Sudarant algoritmą reikėjo įvertinti keletą aspektų, kurie galėjo paveikti rezultatus.

Apšvietimo pasikeitimo problema atsiranda dėl saulės šviesos srauto kitimo arba kito šviesos šaltinio įjungimo ar išjungimo. Tokie pasikeitimai gali paveikti fono modelį arba gali būti užregistruoti kaip judantys objektai, ko pasekoje tokio modelio naudojimas lemia visiškai klaidingus rezultatus. Lėtas apšvietimo pasikeitimas neturi jokio įtakos algoritmui, kadangi naudojamas prisitaikantis fono modelis, kuriam pilnai užtenka keleto sekundžių visiškam atsinaujinimui. Staigūs apšvietimo pokyčiai yra kontroliuojami remiantis prielaida, jog kadre esantys objektai negali vienu metu dingti iš kadro. Tai reikštų, kad arba vaizdo kamera buvo sabotuojama, arba staigiai pasikeitė scenos apšvietumo lygis. Tokiu būdu fono modelio atnaujinimas yra sustabdomas ir laukiama, kol vaizdas atsistatys.

Slenksčio taikymo problema išryškėja, kai esamas kadras lyginamas su atraminiu fono kadru. Atsakymas į klausimą, kiek panašūs turi būti taškų intensyvumai, kad juos būtų galima laikyti vienodais, sprendžiamas statistiniais metodais. Naudojamas fono vaizdo eilutės taškų vidurkis ir standartinis nuokrypis.

$$Thr = \bar{x}_{eil.} + \sigma_{eil.},$$

kur Thr - slenkstinė reikšmė;

\bar{x} - eilutės taškų intensyvumų vidurkis;

σ - eilutės taškų intensyvumų standartinis nuokrypis.

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i,$$

čia N - taškų skaičius eilutėje;

x_i - taško intensyvumo reikšmė.

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}.$$

Maskavimas yra reiškinys, kai objektas yra tos pačios spalvos ar tekstūros, kaip ir fonas. Visiškas maskavimas yra retai pasitaikantis dalykas. Tačiau blogos kokybės vaizdo sekoje gali pasitaikyti dalinis maskavimas, kuris neįneša didelių klaidų į objektų skaičiaus nustatymo rezultatus, todėl jo galima nepaisyti.

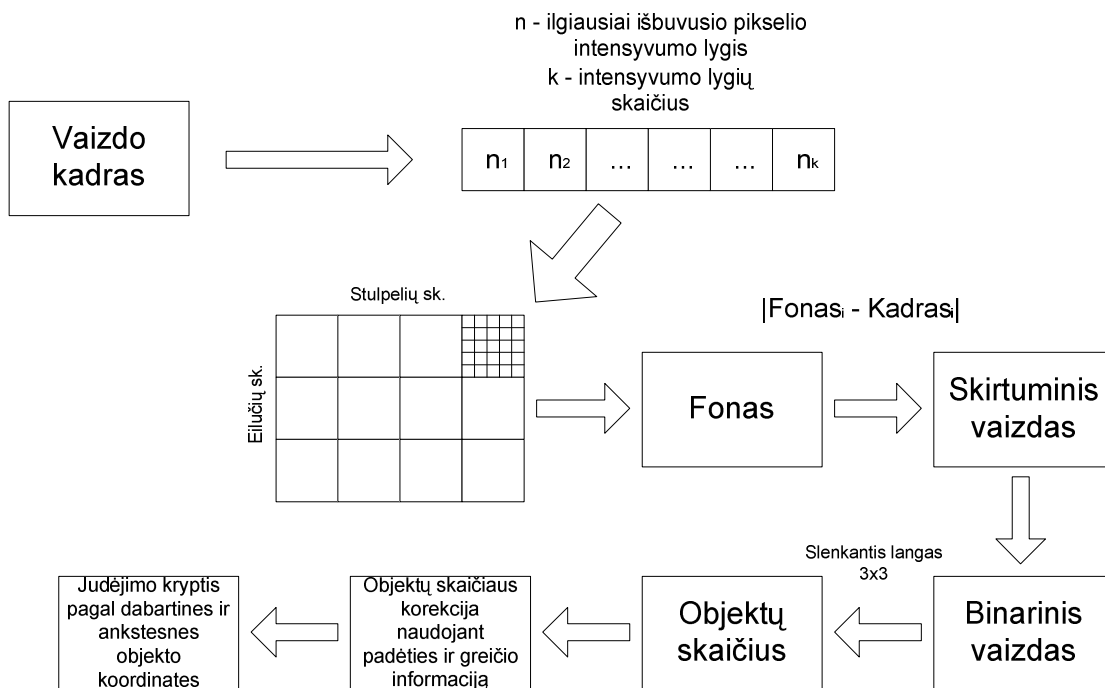
Atvejai, kai fonas turi nestatinių dalių, pavyzdžiui, judantys medžių lapai, gali būti eliminuoti paliekant aktyvias apdorojimui tik tas kadro dalis, kuriose yra stebėtoją dominantys objektai. Taip pat vaizde ignoruojami objektai, kurie periodiškai per trumpą laiką atsiranda ir dingsta. Tai atliekama fiksuojant kadru, kuriuose objektas buvo matomas, skaičių.

Iš fono vaizdo pasitraukiantys ar jame atsirandantys objektai (pavyzdžiui, ilgai stovėjęs automobilis nuvažiavo) laikui bėgant yra įtraukiami į fono modelį. Laikas, po kurio naujai atsiradęs objektas tampa fonu, priklauso nuo fono atnaujinimui naudojamų kadrų skaičiaus, kas savo ruožtu priklauso nuo kintamojo tipo. Pavyzdžiui, jei sveikojo tipo kintamajam naudojami 4 baitai atminties, maksimalus fono atnaujinimui naudojamų kadrų skaičius yra 2^{32} . Algoritmas taip pat turi galimybę neištraukti nejudančio objekto į fono vaizdą. Jei pasirenkama neištraukti jį į fono modelį, taškai, priklausantys objektui, nėra įkeliami į fono histogramą ir neturi jokios įtakos fono vaizdai. Įtraukiant šiuos taškus į modelį objektas pamažu tampa vertinamas kaip fono dalis.

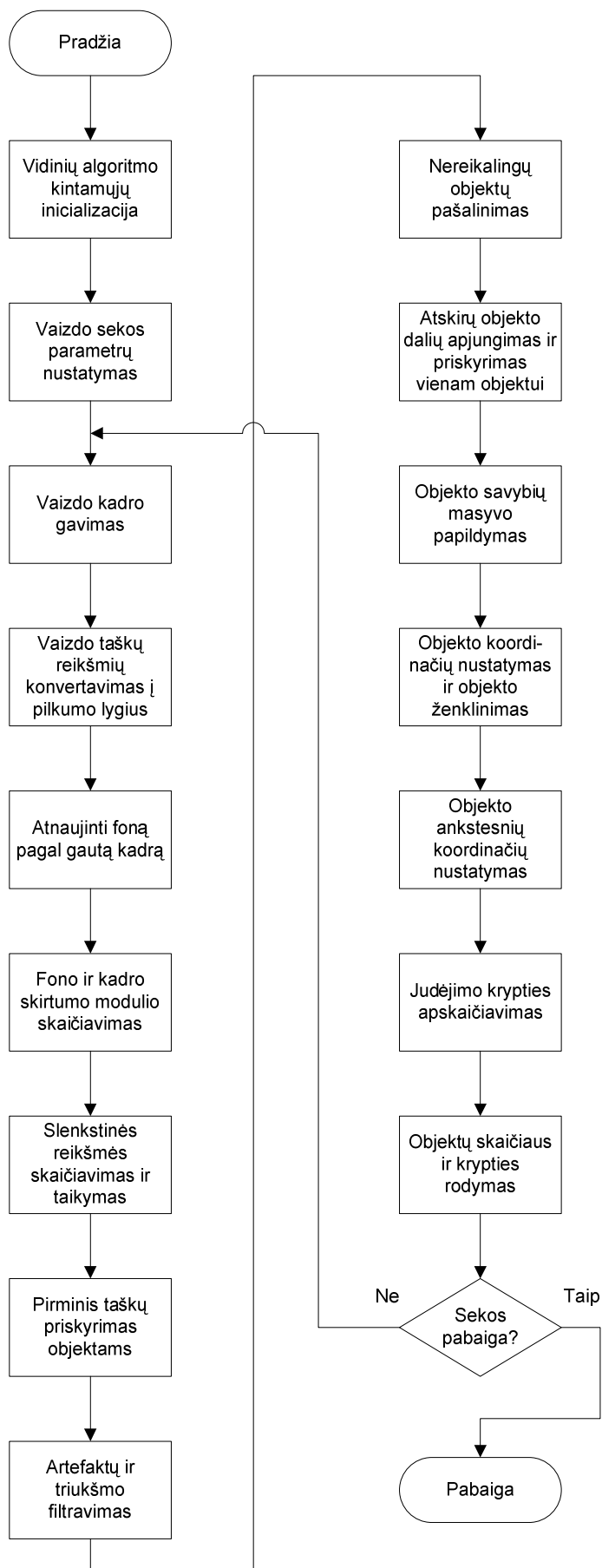
Preliminarus objekto/fono klasifikavimas atliekamas pikselių lygyje. Pagal juos sudaroma histograma, kurios kiekvienos koordinatės maksimumas yra ilgiausiai išbuvusio taško intensyvumo lygis, laikomas tos koordinatės fono tašku. Tokiu būdu atmintyje laikoma tam tikro laiko intervalo kadrų sekos istorija. Šio metodo privalumas yra tas, kad atminties sąnaudos yra proporcingos kadro eilučių bei stulpelių skaičiui ir intensyvumo lygių skaičiui, ir nepriklauso nuo kadrų dažnio ir jų skaičiaus. Tokiu būdu, 320x240 raiškos 256 intensyvumo lygių vaizdo sekos fono modeliui reikalinga atmintis skiriant po 4 baitus vienai reikšmei yra:

$$320 \times 240 \times 256 \times 4 = 78643200 = 75Mb$$

Supaprastinta blokinė schema pateikta 3.1 paveiksle. Algoritmo struktūra pateikta 3.2 paveiksle.



3.1 pav. Algoritmo struktūra



3.2 pav. Algoritmo struktūrinė schema

Apskaičiavus absoliutų fono ir esamo kadro skirtumą ir pritaikius slenksinį gaunamas kadro binarinis vaizdas, kuriame išsidėsčiusios kadro sritys, žyminčios judančius objektus. Šių sričių skaičius yra preliminarus užfiksuotų objektų skaičius. Atliekant triukšmo filtravimą pašalinami kameros generuojami triukšmai, mažų matmenų objektai (kurių projekcijos vaizde komponentės ne didesnės kaip 10 pikselių) ir objektai, kurių pasirodymo dažnis yra didesnis nei 1 s^{-1} . Binariniame vaizde lieka tik pakankamai didelės, stabilios laiko atžvilgiu sritys. Taip fiksuojamų objektų skaičius tikslinamas ir artėja iki tikrojo stebėtoją dominančių objektų skaičiaus.

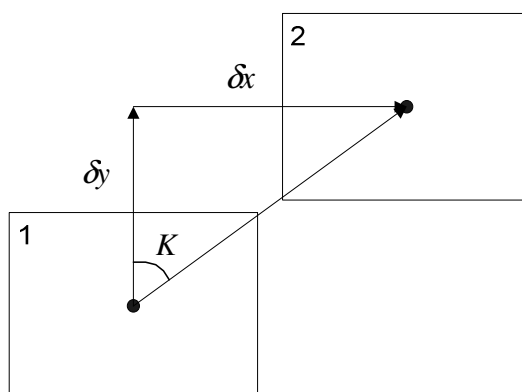
Objekto judėjimo kryptis nustatoma apskaičiuojant objekto centrinio taško koordinatinių poslinkių x ir y kryptimis santykio arktangentą (3.3 pav.):

$$K = \text{arctg}\left(\frac{\delta x}{\delta y}\right),$$

čia K – objekto kryptis laipsniais,

δx – objekto centro taško poslinkis x kryptimi,

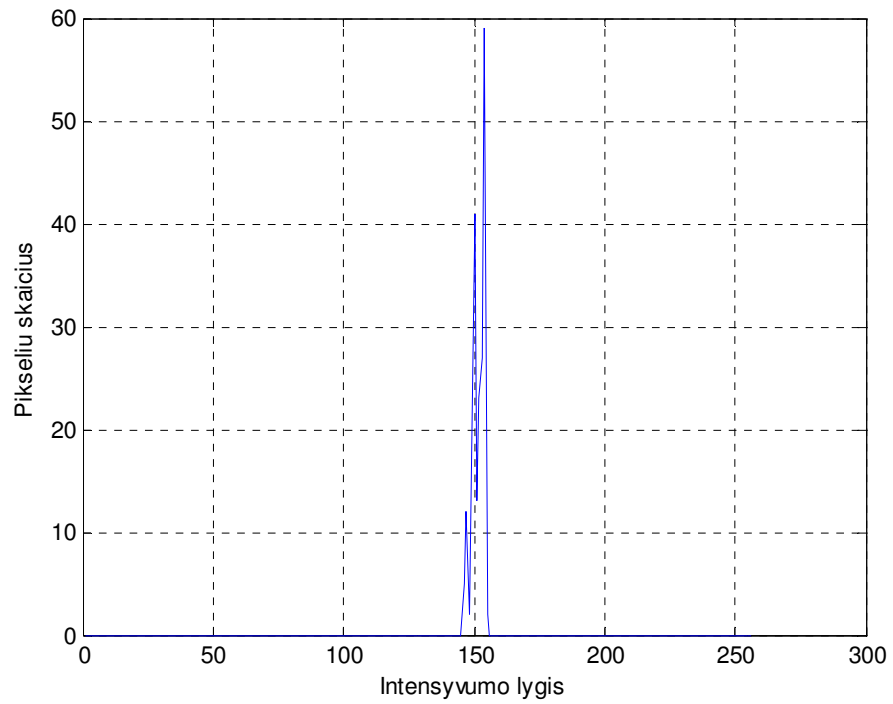
δy – objekto centro taško poslinkis y kryptimi.



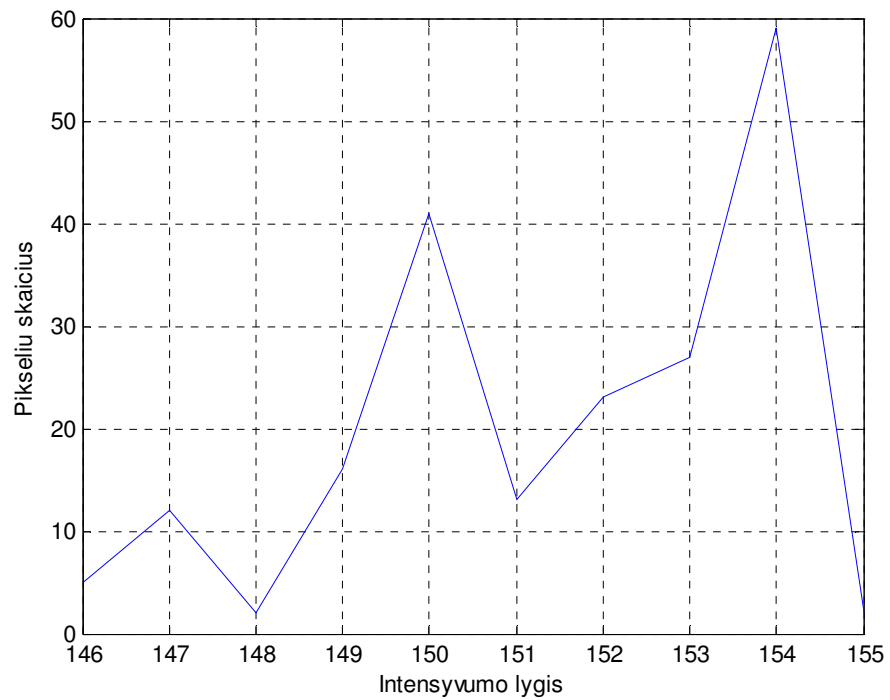
3.3 pav. Objekto judėjimo krypties nustatymas

Kad sumažinti naudojamos atminties dydį, siūloma naudoti dinaminį atminties skyrimą ir laikyti tik didžiausias reikšmes turinčią histogramos masyvo dalį. 3.4 paveiksle pavaizduotas vieno kadro taško intensyvumo lygių reikšmių pasirodymų skaičius po 200 vaizdo kadro. Iš paveikslo matyti, kad didelė atminties dalis yra švaistoma nulinėms reikšmėms, o naudingi duomenys yra sukonzentruoti intervale nuo 146 iki 155, kuriame apskaičiuota fono taško reikšmė yra 154 pilkumo lygis su 59 pasirodymais. Tokiu būdu dinamiškai skiriant tik 10 atminties ląstelių vietoje 256 sumažinama atminties vieta iki 2,93 Mb vietoje 75 Mb:

$$320 \times 240 \times 10 \times 4 = 3072000 = 2.93 \text{ Mb} .$$



a)



b)

3.4 pav. Taško intensyvumo reikšmių pasirodymų skaičius:

a – pilkumo lygiai nuo 0 iki 255,

b – pilkumo lygiai nuo 146 iki 155.

4. EKSPERIMENTO REZULTATAI

Visi eksperimentai buvo atliekami su iš anksto įrašyta vaizdo medžiaga. Vaizdo įrašas buvo atliktas su JVC TK-C676EB vaizdo kamera, kurios parametrai yra tokie:

- PAL standarto signalų sistema;
- ¼“ CCD vaizdo matrica;
- 440,000 (752H x 582V) efektyvių vaizdo elementų;
- -50Db signalas/triukšmas santykis (be automatinio signalo stiprinimo);
- 480 TV eilučių;
- 3,8-103 mm objektyvas;
- 0,025 Lux minimalus apšvietumas (f1.4).

Siekiant įvertinti algoritmo darbą realiu laiku, sukurta programa iš įrašyto vaizdo failo paima vieną kadrą per $\frac{1}{\text{kadrų / sekundę}}$ laiko intervalą, taip simuliuodama vaizdo kadro gavimą iš išorinio įrenginio.

Algoritmo kūrimui ir rezultatų analizei buvo panaudotas Matlab 7.0 programų paketas. Programos kūrimui bei optimizavimui panaudota Bloodshed wxDev-C++ 4.9.9.2 programavimo aplinka. Vaizdo medžiaga, išeities kodai ir sukompiluoti failai pateikti pridedamame kompaktiniame diske. Vaizdo įrašuose užfiksuoti įvairūs judantys objektai stacionariame fone. Vaizdo įrašai iš kamerų buvo išsaugoti MPEG2 formatu ir konvertuoti į AVI failus supaprastinant panaudojimą. Eksperimente taip pat buvo naudojami nesuspausti vaizdo įrašai, kad padidinti atitenkančius vaizdų apdorojimui procesoriaus darbo resursus, eliminavus dekompresijos žingsnį, kuris kitaip užimtų didesnę procesoriaus darbo laiko dalį.

4.1. Slenkančio vidurkio, medianos, kadrų skirtumo metodų tyrimas

Buvo išbandyti trys paplitę metodai: slenkančio vidurkio metodas, medianos metodas ir kadrų skirtumo metodas. Bandomoji kadrų seka sudaryta iš 160 kadrų su keliais judančiais objektais. Apskaičiuojamas fono vaizdas yra palyginamas su laisvai iš sekos pasirinktu kadru, kuriame nėra judančių objektų ir kuris laikomas fonu.

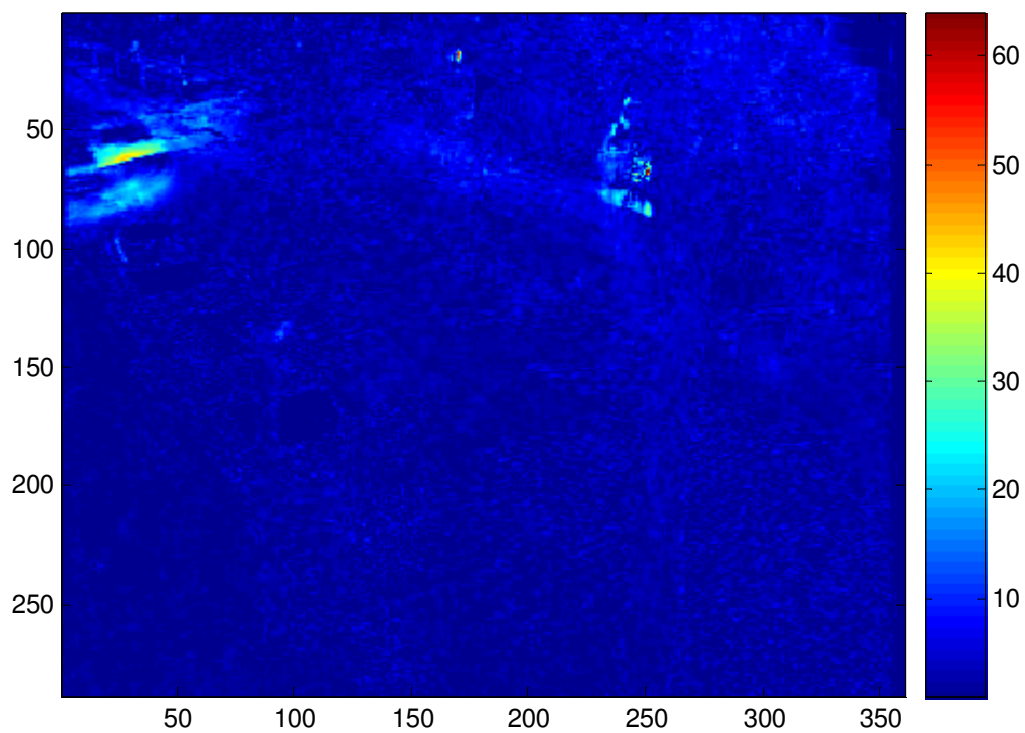
Slenkančio vidurkio metodas

Iš skirtumo tarp rasto ir pasirinkto fono (4.1.1 pav.) apskaičiuojamas klaidos vidurkis ir nuokrypių vidurkio kvadratinė šaknis:

$$KV = \frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)}{N} = 2.8591$$



a)



b)

4.1.1 pav. Slenkančio vidurkio metodu rastas fonas (a) ir absoliutus skirtumas tarp rasto ir pasirinkto fono (b)

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)^2}{N}} = 4.0155$$

Medianos metodas

Iš skirtumo tarp rasto ir pasirinkto fono (4.1.2 pav.) apskaičiuojamas klaidos vidurkis ir nuokrypių vidurkio kvadratinė šaknis:

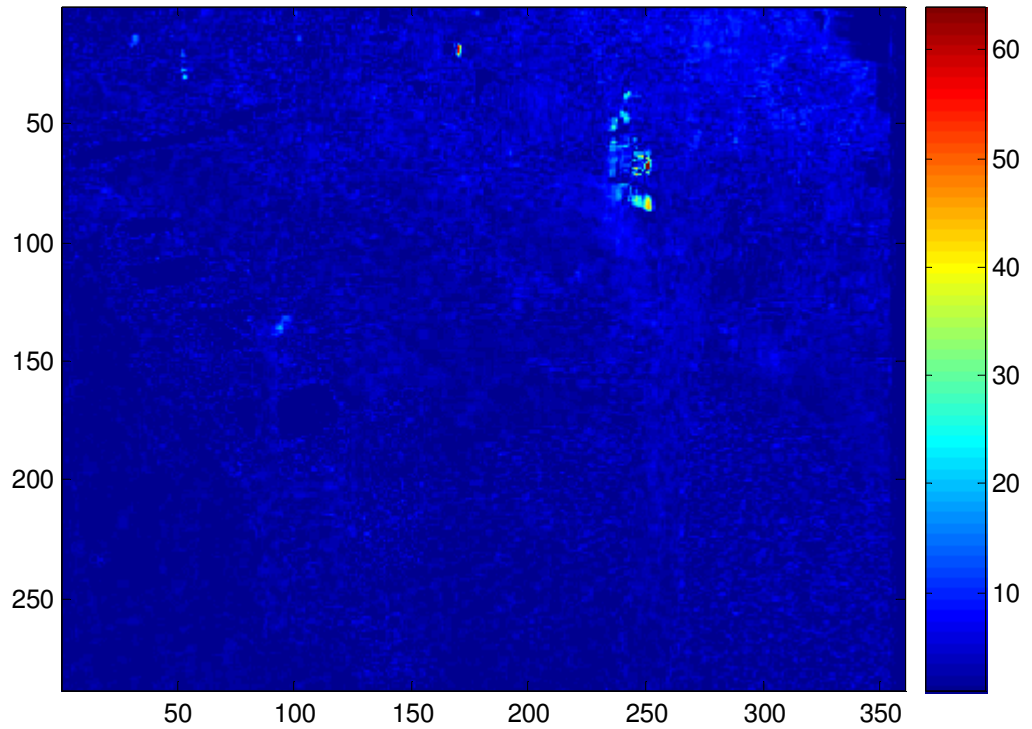
$$KV = \frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)}{N} = 2.5299$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)^2}{N}} = 3.5369$$

Medianos metodu gautas fonas turi mažesnę klaidos vidurkį ir nuokrypių vidurkio kvadratinę šaknį, negu slenkančio vidurkio metodu gautas fonas, todėl galima teigti, kad objektų nustatymo rezultatas bus geresnis naudojant šį metodą.



a)



b)

4.1.2 pav. Medianos metodu rastas fonas (a) ir absoliutus skirtumas tarp rasto ir pasirinkto fono (b)

Kadru skirtumo metodas

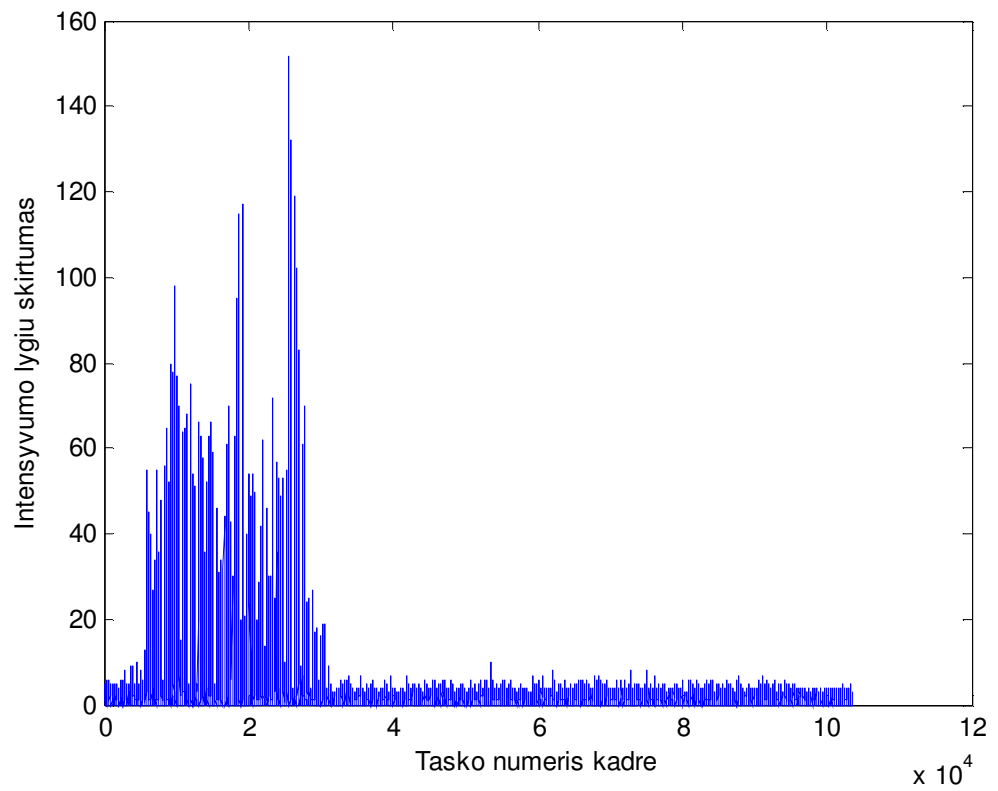
Kadru skirtumo metodas leidžia aptikti tik objekto kontūrus ir rezultate negarantuoja, kad gautas kontūras bus uždaras (4.1.3 pav.).



a – Kadru skirtumo metodui panaudoti du gretimi kadrai



b - Kadru skirtumo metodu rastas skirtuminis vaizdas



c - Skirtuminio vaizdo grafinis atvaizdavimas

4.1.3 pav. Gretimų kadru skirtumo metodas

4.2. Siūlomo metodo tyrimas

Naudojant pasiūlytą metodą efektyvus vaizdo fono atstatymas turi pagrindinę įtaką rezultatams. Pasiūlytas algoritmas pritaikomas vaizdo kadrų sekai, turinčiai 720x576 taškų skiriamąją gebą. Algoritmas gali aptikti judančius, o taip pat ir trumpai sustojusius objektus visoje vaizdo kadrų sekoje.

Kad įvertinti algoritmo atsparumą melagingiems suveikimams ir triukšmui, atliekamas bandymas su vaizdo kadrėmis, kuriuose nėra nepriklausomai judančių objektų. Paveiksle 4.2.1 pavaizduotas kadrų sekos paskutinis 250-asis kadras. Paveiksle 4.2.2 parodytas apskaičiuotas kadrų sekos fono vaizdas. Paveiksle 4.2.3 pavaizduotas invertuotas skirtuminis vaizdo kadro ir fono vaizdas. Invertuoto skirtuminio vaizdo eilučių taškų vidurkiai (4.2.4 pav.) rodo nedidelį triukšmą, kurio vidutinė invertuota reikšmė yra 254, tai yra triukšmo lygis skirtuminiame vaizde sudaro vos 0,3922 % taško maksimalios vertės. Iš paveikslų matyti, kad naudojant pasiūlytą algoritmą galima patikimai nustatyti sekos foną.

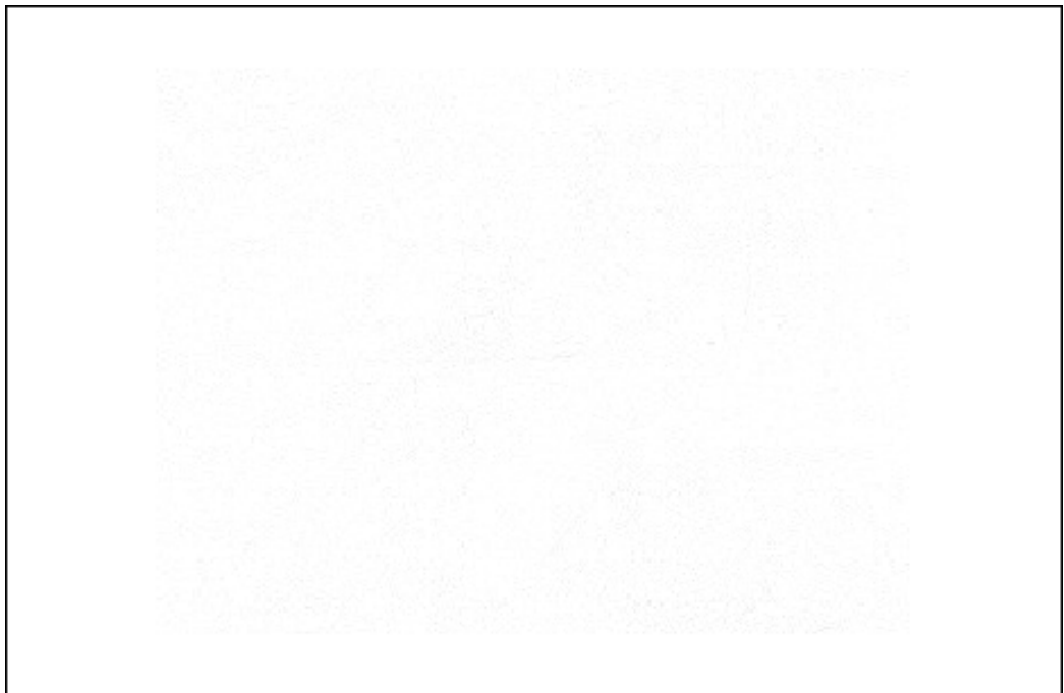
Slenksčio taikymo operacijos rezultate pašalinama didžiausia skirtuminio vaizdo triukšmo dalis, kadangi algoritmas patikimai nustato vaizdo foną. Yra pastebima apribojimo vertės didėjimo tendencija, kai kadre atsiranda judantys objektai. Tai susiję su didesnių intensyvumo lygių atsiradimu skirtuminiame vaizde, kai scenoje aptinkamas objektas, nepriklausantis fonui. Jei būtų pasirinkta nekintanti slenkstinė vertė, ji nereaguotų į tokius pasikeitimus, kaip scenos bendro apšviestumo pokyčiai.



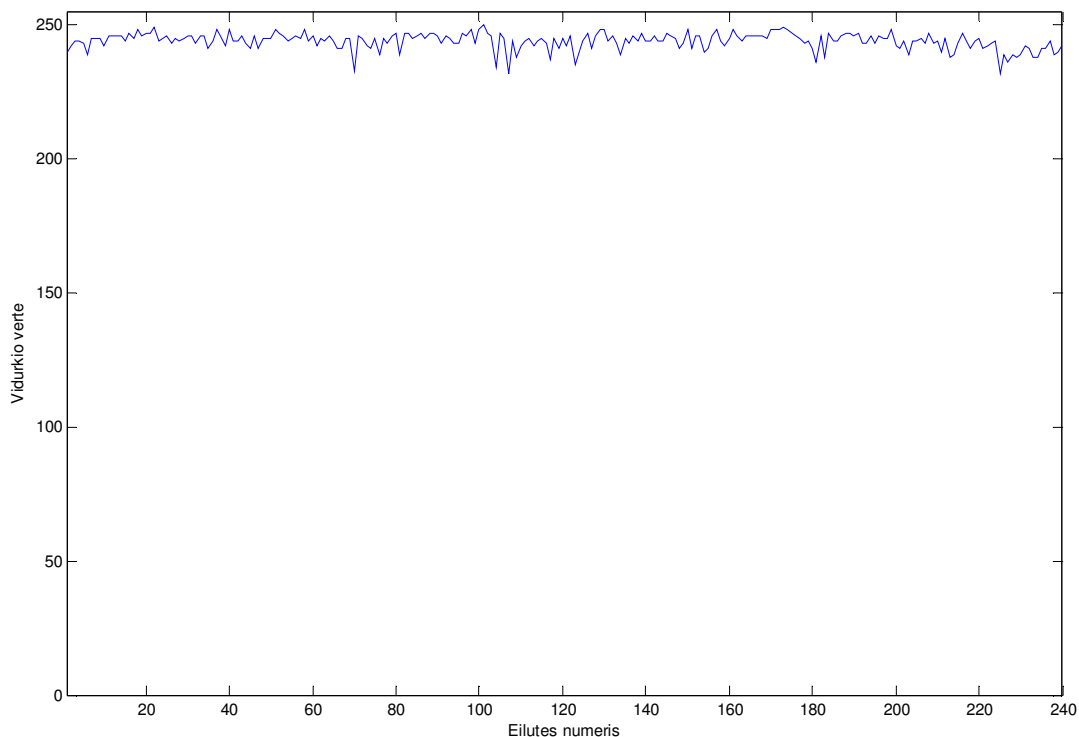
4.2.1 pav. Kadrų sekos 250-asis kadras



4.2.2 pav. Apskaičiuotas kadru sekos fonas



4.2.3 pav. Invertuotas skirtuminis vaizdas

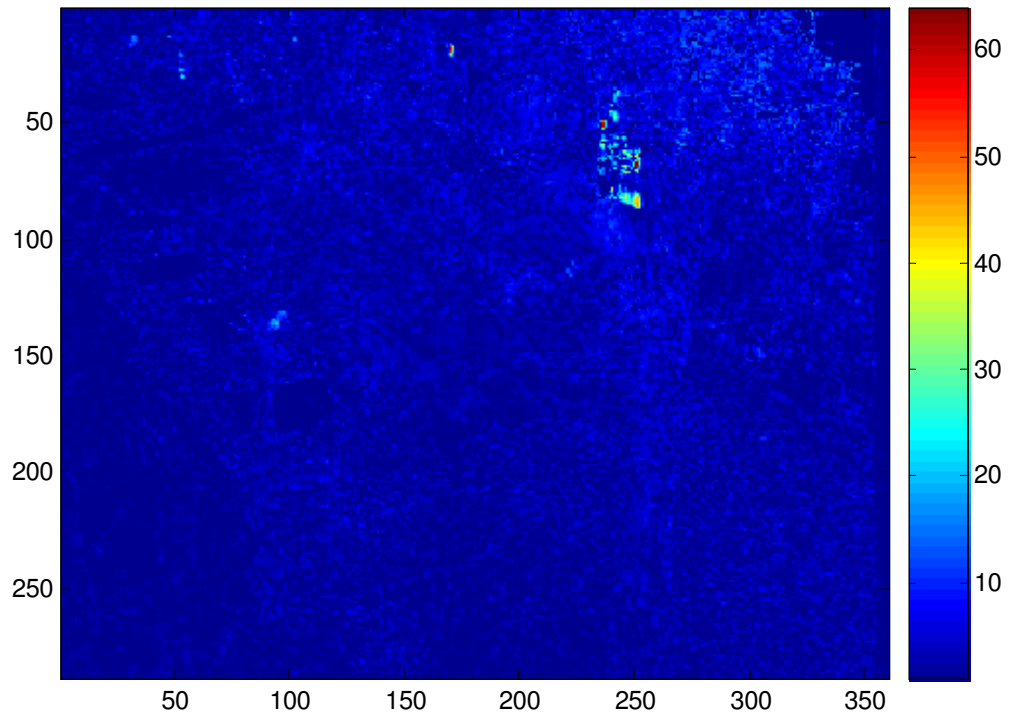


4.2.4 pav. Invertuoto skirtuminio vaizdo eilučių vidurkio reikšmės

4.2.5 paveiksle pavaizduotas apskaičiuoto fono vaizdas, 4.2.6 paveiksle pavaizduotas absoliutus skirtumas tarp surasto ir laisvai pasirinkto fono. Apskaičiavus klaidos vidurkį ir nuokrypių vidurkio kvadratinę šaknį matyti, kad šio metodo fono nustatymo rezultatai yra geresni, negu medianos ar slenkančio vidurkio metodų.



4.2.5 pav. Siūlomų metodu apskaičiuotas fonas



4.2.6 pav. Absoliutus skirtumas tarp surasto ir laisvai pasirinkto fono

$$KV = \frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)}{N} = 2.4139$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (Fonas_i - ApskFonas_i)^2}{N}} = 3.3615$$

Apskaičiuojamo vaizdo fono paklaida mažėja didėjant apdorotam vaizdo sekos kadrų skaičiui (4.2.7 pav.). Judančių objektų taškai pamažu dingsta iš fono vaizdo ir gautas fonas artėja prie tikrojo.



4.2.7 pav. a – apskaičiuotas fonas po 2 kadrų (po 0,17 sekundės)



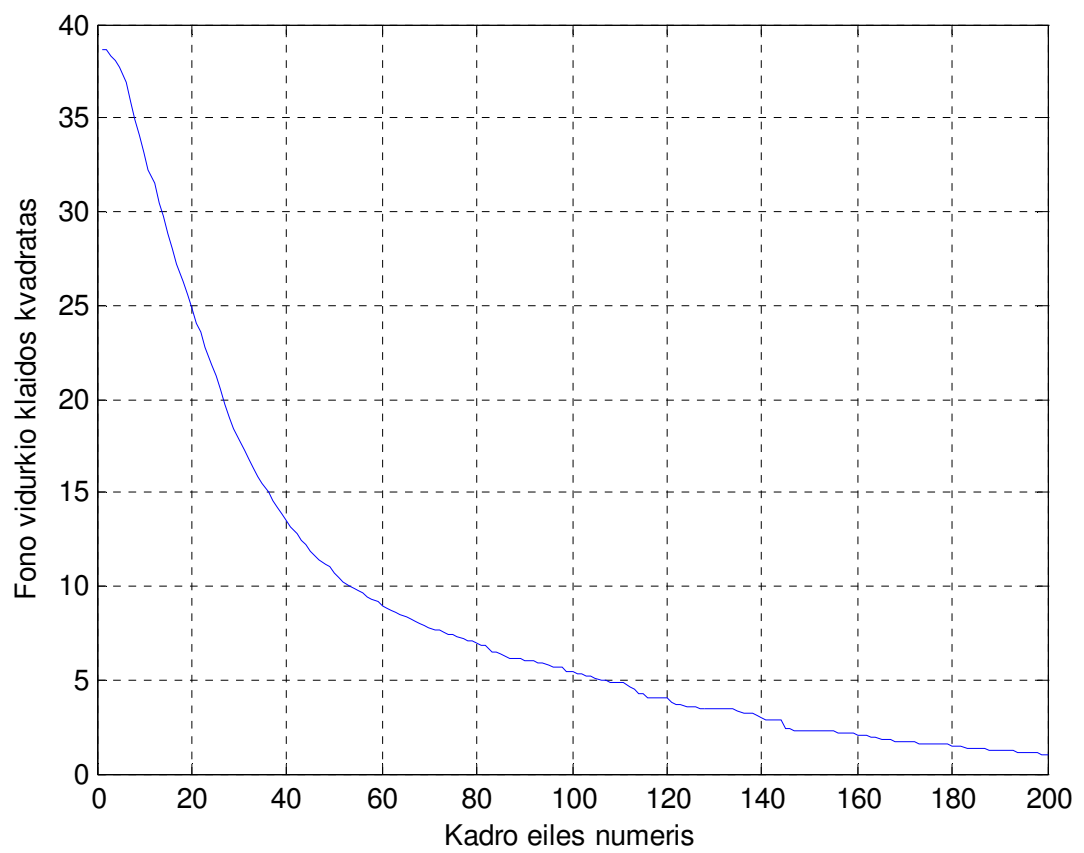
b – apskaičiuotas fonas po 10 kadrų (po 0,83 sekundės)



c – apskaičiuotas fonas po 20 kadrų (po 1,7 sekundės)



d – apskaičiuotas fonas po 30 kadro (po 2,5 sekundės)



e – fono vidurkio klaidos kvadrato kitimas

Eksperimento rezultatai rodo, kad fono vidurkio klaidos kvadratas didėjant fono atstatymui naudojamų kadrų skaičiui mažėja eksponentiškai (4.2.7 e pav.). Atlikus kadro ir fono atimties operaciją ir imant rezultato modulį gauname skirtuminį vaizdą, parodytą 4.2.8 paveiksle.



4.2.8 pav. Vaizdo sekos 200-tojo kadro ir fono invertuotas skirtuminis vaizdas

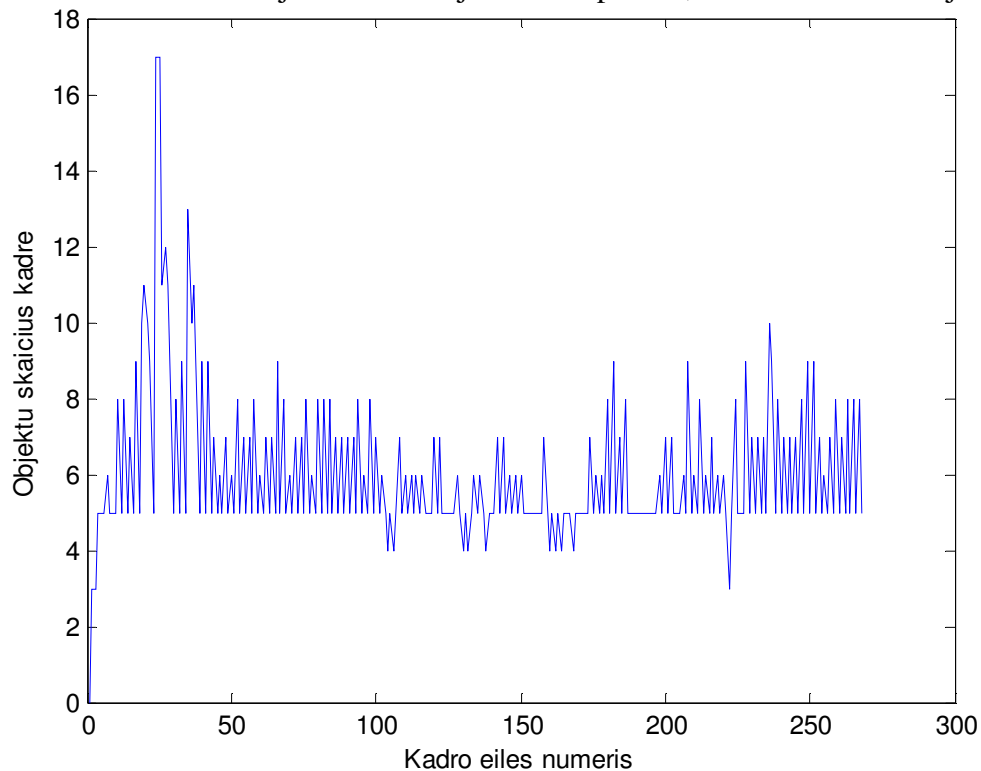
Paveiksle 4.2.9 parodytas binarinis vaizdas pritaikius apskaičiuotam skirtuminiam vaizdui dinamiškai kintančią slenkstinę vertę. Kaip matyti iš paveikslo, po šios operacijos dažnai pasireiškia



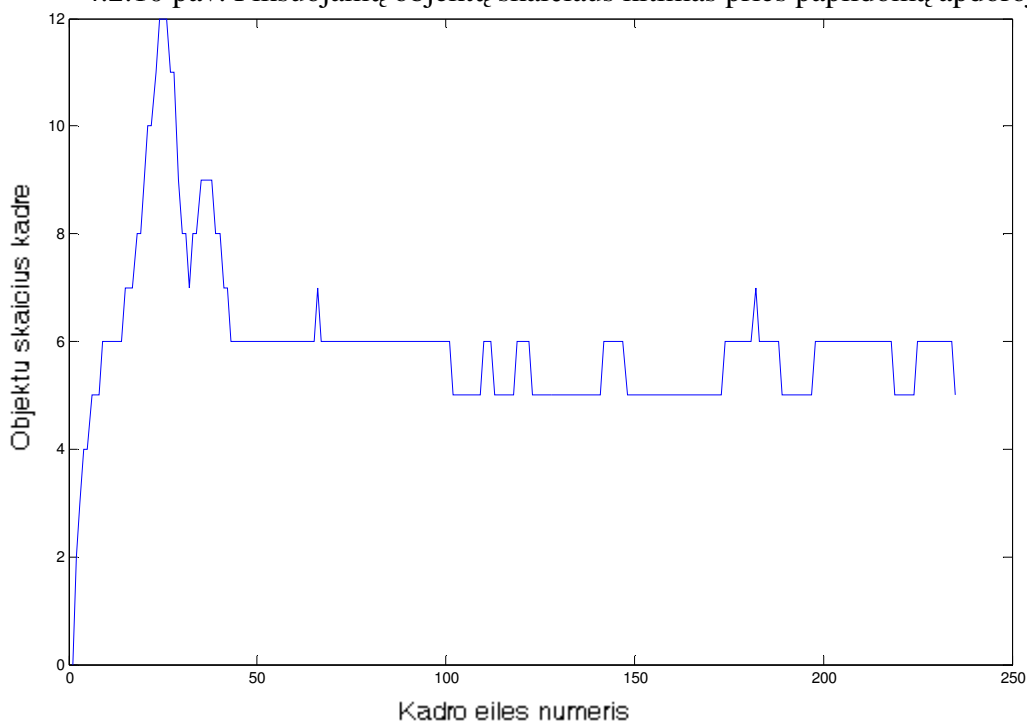
4.2.9 pav. Binarinis vaizdas po slenkstinės vertės taikymo

maskavimas, t.y. objekto taškai, turintys panašius į fono intensyvumo lygius, pažymimi kaip fonas ir objekto binarinis vaizdas įgyja juodas dėmes. Kad tai pašalinti, jų vidus užpildomas taškais, kurie priklauso atitinkamam objektui.

Kiekvienam binarinio vaizdo objektui priskiriamas identifikacinis numeris ir suskaičiuojamas bendras objektų skaičius. Paveiksle 4.2.10 parodytas objektų skaičiaus kitimas bėgant laikui. Dažnai vienas objektas fiksuojamas kaip keli, todėl stebimi objektų skaičiaus



4.2.10 pav. Fiksuojamų objektų skaičiaus kitimas prieš papildomą apdorojimą



4.2.11 pav. Fiksuojamų objektų skaičiaus kitimas po apdorojimo

svyravimai. Kad to išvengti, naudojamas papildomas apdorojimas, kurio metu atliekamas persidengiančių ir šalia vienas kito grupėje judančių objektų apjungimas. Paveiksle 4.2.12 ir 4.2.13 pavaizduotas vaizdo kadras atitinkamai prieš apjungimą ir po jo.



4.2.12 pav. Kadras su nustatytais objektais prieš segmentų apjungimą



4.2.13 pav. Kadras su nustatytais objektais po segmentų apjungimo

Matome, kad objektas, anksčiau pažymėtas kaip trys objektai, po segmentų apjungimo yra fiksuojamas kaip vienas.

Siūlomo metodo atsparumas triukšmui tikrinamas į vaizdo seką dirbtinai įnešant „druska ir pipirai“ triukšmą (4.2.14 pav.).



4.2.14 pav. Kadras su „druska ir pipirai“ triukšmu

Atlikus fono skaičiavimus fono taškų klaidos kvadrato (4.2.16 pav.) vidurkis yra

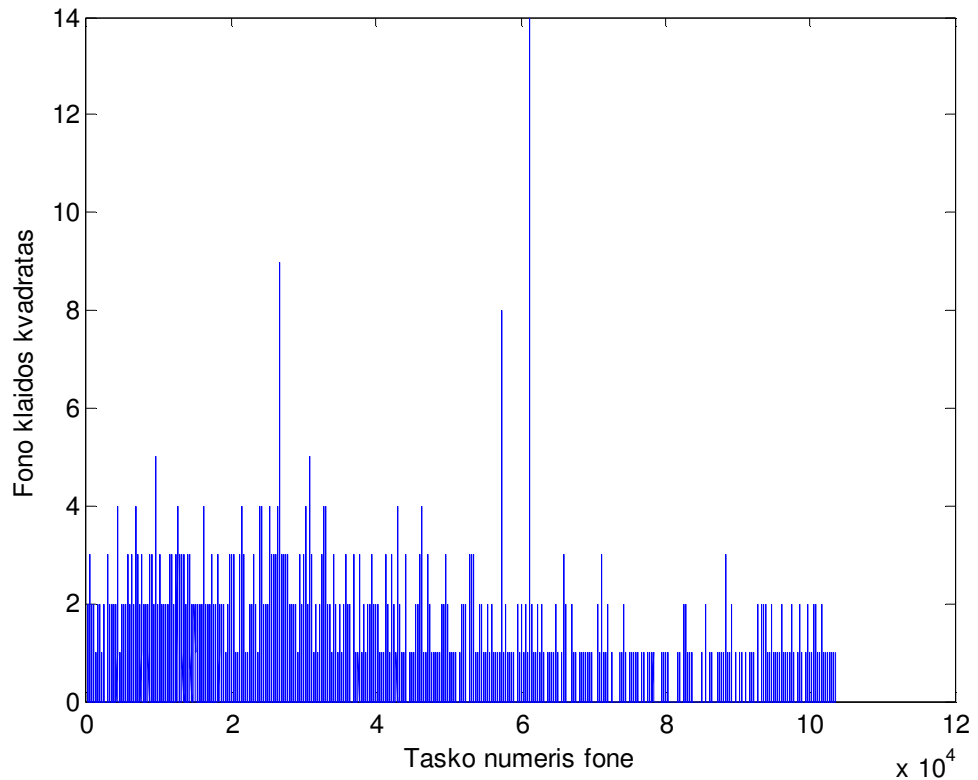
$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = 0.0166,$$

o standartinis nuokrypis:

$$s = \sqrt{\frac{\sum |\bar{x} - x|^2}{n-1}} = 0.1725 .$$



4.2.15 pav. Apskaičiuotas kadrų sekos su triukšmu fonas

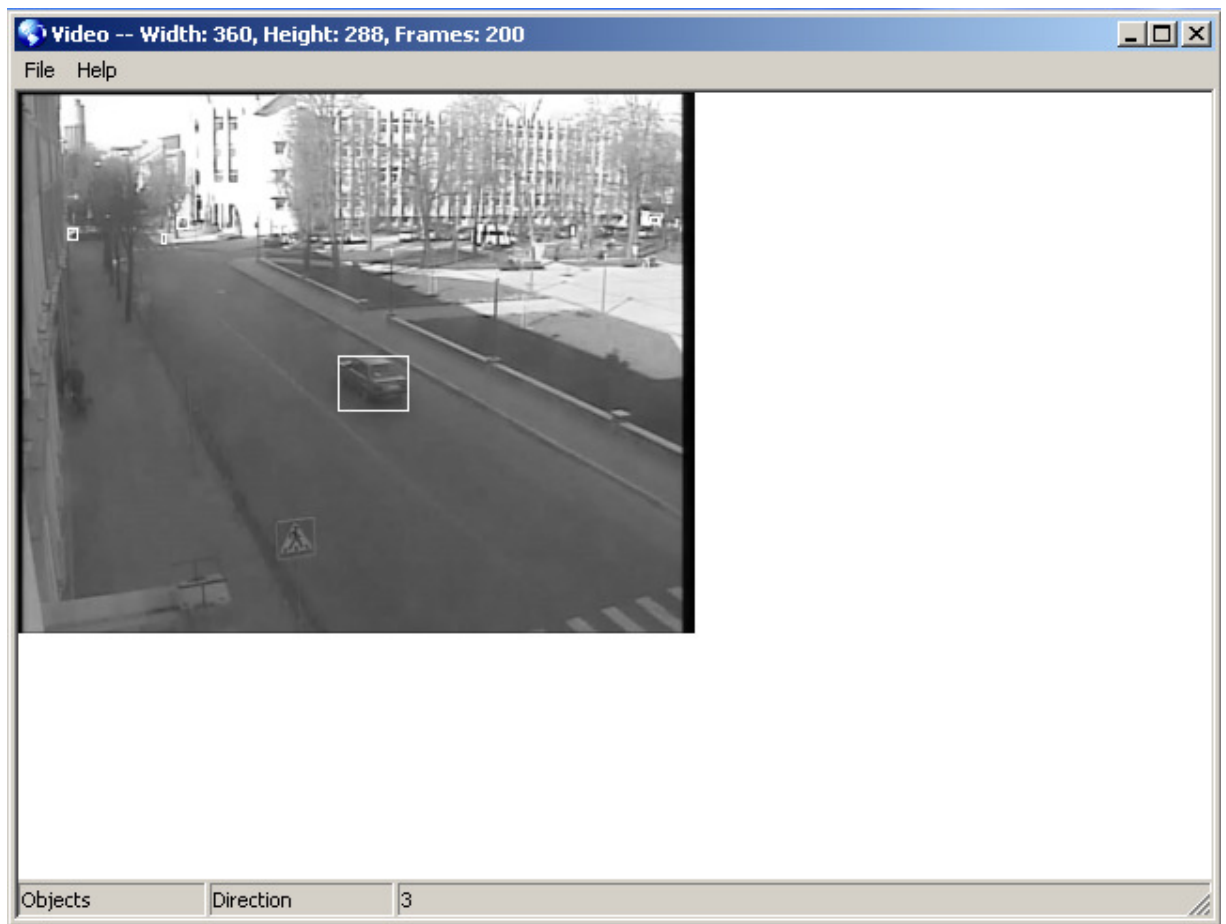


4.2.16 pav. Fono taškų klaidos kvadratas

Tokiu būdu triukšmas neįneša akiai pastebimų fono iškreipimų ir yra pašalinamas pradinėje algoritmo stadijoje.

4.3. Programos sąsaja

Programos langas parodytas 4.3.1 paveiksle. Vaizdo failo atidarymas atliekamas per meniu juostos punktą File->Open. Po failo atidarymo antraštės juostoje rodomas kadro stulpelių ir eilučių skaičius bei kadro skaičius. Pagrindinėje lango dalyje kadras po kadro rodoma vaizdo seka. Vaizdo kadre baltu rėmeliu pažymimas judantis objektas. Būsenos juostos pirmame langelyje rodomas kadre esančių objektų skaičius. Antrame langelyje rodoma objekto kryptis laipsniais, kur 0° yra kameros žiūrėjimo kryptis (\uparrow), 90° - stačiu kampu į kairę nuo nulinės pasukta kryptis (\rightarrow) ir t.t. Trečias būsenos juostos langelis rodo šiuo metu pasirinkto objekto numerį. Objektų pasirinkimas atliekamas pelės kursorių nustatant ant įrėminto objekto. Šis pasirinkimas yra išsaugomas atmintyje ir būsenos juostoje rodoma judėjimo kryptis yra paskutinio taip pažymėto objekto.



4.3.1 pav. Programos langas

IŠVADOS

Šiame darbe buvo sudarytas objektų judėjimo krypties ir skaičiaus nustatymo algoritmas ir suprojektuota programinė įranga, leidžianti iš vaizdo kadrų sekos nustatyti objektų skaičių bei judėjimo kryptį.

Algoritmo pagrindą sudaro fono pašalinimas iš vaizdo kadrų sekos, po kurio eina vaizdo taškų priskyrimas objektams, objektų ženklavimas ir numeravimas, objektų sekimas ir poslinkio krypties nustatymas. Tikslus fono modelio apskaičiavimas užtikrina gerus rezultatus nustatant judančių objektų skaičių.

Pasiūlytas algoritmas buvo taikomas RGB paletės kadrams. Algoritmo modifikavimas YUV paletę naudojančioms vaizdo sekoms galėtų užtikrinti patikimesnę fono atstatymą bei šešėlių priskyrimą fonui. YUV paletė naudoja Y kanalą skaisčiui ir du kanalus (U ir V) spalvoms aprašyti. Teoriškai tai turi pranašumą, nes keičiantis apšvietimui YUV kadre keičiasi tik skaitis, tuo tarpu RGB kadre keičiasi visos trys spalvos komponentės. YUV ir RGB sekų parametrų kitimas skiriasi. Y komponentės kitimas dažniausiai yra didesnis, negu U ir V komponentių, kadangi šešėlių atsiradimas paprastai veikia tik skaisčio komponentę.

LITERATŪRA

1. Video Motion Detection. URL:
<http://www.videomotiondetectors.com/manuals/VMDtalk.htm>
2. “Human Motion Segmentation and Object Recognition using Fuzzy Rules”, J. E. Hunter, Center for Intelligent Systems, Vanderbilt University, Nashville, 2005, p. 4. URL:
<http://eecs.vanderbilt.edu/CIS/pubs/2005-Hunter-ROMAN.pdf>
3. “Vehicle Segmentation and Tracking in the Presence of Occlusions”, Neeraj K. Kanhere, Wayne A. Sarasua, Clemson University, 2005, p. 15. URL:
http://www.ces.clemson.edu/~stb/publications/vehicle_tracking_trb2006.pdf
4. „Image difference threshold strategies and shadow detection“, Paul L. Rosin, Tim Ellis, p. 1-4. URL:
<http://users.cs.cf.ac.uk/Paul.Rosin/resources/papers/shadows.pdf>
5. “The Image Processing Handbook, 3rd edition“, John C. Russ, 1998, p. 530-533.
6. „Background subtraction techniques: a review“, Massimo Piccardi, University of Technology, Sydney, 2004. URL:
<http://www-staff.it.uts.edu.au/~massimo/BackgroundSubtractionReview-Piccardi.pdf>
7. “Survey over image thresholding techniques and quantitative performance evaluation”, M. Sezgin, B. Sankur, 2004. p. 2-13. URL:
http://www.busim.ee.boun.edu.tr/~sankur/SankurFolder/Threshold_survey.pdf
8. “Art of Optical Flow”. URL:
<http://www.fxguide.com/article333.html>
9. “Optical Flow”, Namrata Vaswani. URL:
<http://www.ece.iastate.edu/~namrata/EE527/OpticalFlow.pdf>
10. “Determining Optical Flow”, B. K. P. Horn, B. G. Schunck, Artificial Intelligence, vol. 17, 1981, p. 185-203. URL:
<http://www.caam.rice.edu/~zhang/caam699/opt-flow/horn81.pdf>
11. P. Wayne Power, Johann A. Schoonees, “Understanding Background Mixture Models for Foreground Segmentation”. Proceedings Image and Vision Computing, New Zealand, 2002, p. 267. URL:
<http://www.vision.irl.cri.nz/pubdoc/2002/JSP2002.pdf>
12. W. Long and Yee-Hong Yang, “Stationary background generation: an alternative to the difference of two images”. Pattern Recogn., 23(12), 1990.
13. J. Heikkila, O. Silven, “A real-time system for monitoring of cyclists and pedestrians”. Second IEEE Workshop on Visual Surveillance, Colorado, 1999, p. 82-90. URL:

http://www.ee.oulu.fi/mvg/files/pdf/pdf_156.pdf

14. C. Stauffer, W. E. L. Grimson, „Adaptive background mixture models for real-time tracking“. Computer Vision and Pattern Recognition, Colorado, 1999, p. 246-252. URL: http://www.ai.mit.edu/projects/vsam/Publications/stauffer_cvpr98_track.pdf
15. C. Wren, A. Azabajejani, T. Darrell, A. Pentland, “Pfinder: Real-time tracking of the human body”. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 1997, p. 780-785.
16. I. Haritaoglu, D. Harwood and L. S. Davis, “W⁴: Who? When? Where? What? A real time system for detecting and tracking people”. Third Face and Gesture Recognition Conference, 1998, p. 222-227. URL: <http://www.umiacs.umd.edu/~hismail/Publications/fg98W4.pdf>
17. T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, A. Erkan, “Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets”. Second IEEE Workshop on Visual Surveillance, Colorado, 1999, p.48-55.
18. G. Halevy, D. Weinshall, “Motion of disturbances: detection and tracking of multi-body non-rigid motion”. Machine Vision and Applications 11, 1999, p.122-137.
19. R. Cutler, L. Davis, “View-based detection and analysis of periodic motion”. International Conference on Pattern Recognition, Australia, 1998, p. 495-500.
20. K. Toyama, J. Krumm, B. Brumitt, B. Meyers, “Wallflower: Principles and practice of background maintenance”. International Conference on Computer Vision, 1999, p. 255-261. URL: <http://research.microsoft.com/~toyama/wallflower.pdf>