

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

MULTIMEDIJOS INŽINERIJOS KATEDRA

Renatas Mandrijauskas

Interaktyvaus pateikčių valdymo metodas

Magistro darbas

Darbo vadovas:

dr. Sigitas Drąsutis

Kaunas, 2010

KAUNO TECHNOLOGIJOS UNIVERSITETAS

INFORMATIKOS FAKULTETAS

MULTIMEDIJOS INŽINERIJOS KATEDRA

Renatas Mandrijauskas

Interaktyvaus pateikčių valdymo metodas

Magistro darbas

Recenzentas

prof. dr. Lina Nemuraitė

2010-05-31

Vadovas

dr. Sigitas Drąsutis

2010-05-31

Atliko

IFN 8/2 gr. stud.

Renatas Mandrijauskas

2010-05-31

Kaunas, 2010

TURINYS

SUMMARY	7
TERMINŲ IR SANTRUMPŲ ŽODYNAS	8
ĮVADAS	9
1. ANALITINĖ DALIS	10
1.1. Problemos aprašymas	10
1.2. Metodai	10
1.3. Prototipai	10
1.4. Sistemos fizinė architektūra	12
1.4.1. Fizinės sistemos konfigūracijos	12
1.4.2. Fizinių sistemos konfigūracijų palyginimas	13
1.4.3. Sistemos įrenginių reikalavimai	13
1.5. Sistemų apžvalga	14
1.5.1. Carsten Kirstein, Heinrich Müller sistemos modelis	14
1.5.2. Jean-Francois Lapointe, Guy Godin sistemos modelis	15
1.6. Technologijų apžvalga	17
1.6.1. DirectX	17
1.6.2. GDI ir GDI+	18
1.6.3. AForge.NET Framework	18
1.6.4. Programavimo kalbos ir IDE alternatyvos	19
1.7. Išvados	20
2. PROJEKTINĖ DALIS	21
2.1. Projekto tikslas	21
2.2. Naudotas projektavimo metodas	21
2.3. Reikalavimų projektuojamai sistemai specifikacija	21
2.3.1. Bendras aprašymas	21
2.3.2. Specifiniai reikalavimai	22
2.4. Sistemos architektūra	28
2.4.1. Architektūros apribojimai	28
2.4.2. Projektuojamos sistemos architektūra	28
2.5. Duomenų struktūra	29
2.6. Programinių modulių specifikacijos	29
2.6.1. Vaizdo apdorojimo posistemis	30
2.6.2. Power Point pateikčių apdorojimo posistemis	31
2.6.3. Vartotojo sąsajos posistemis	31

2.6.4.	Nustatymų posistemis	33
2.6.5.	Grafinių vaizdų apdorojimo posistemis	34
2.7.	Sistemos elgsenos modelis	35
2.8.	Išvados	41
3.	EKSPERIMENTINIS TYRIMAS	42
3.1.	Skiriamųjų gebų ir filtrų tipo tyrimas.	42
3.2.	Slenksčio ir lazerinių žymeklių tipo tyrimas.....	44
3.3.	Minimalių ir maksimalių vaizdo objektų dydžio tyrimas	45
3.4.	Išvados	47
	IŠVADOS.....	48
	LITERATŪRA.....	49
	PRIEDAI.....	50

Lentelių sąrašas

1 lentelė. Prototipų palyginimas.....	11
2 lentelė. Fizinių sistemos konfigūracijų palyginimas.....	13
3 lentelė. „Keisti sistemos nustatymus“ panaudojimo atvejo specifikacija.....	23
4 lentelė. „Keisti skiriamąją gebą“ panaudojimo atvejo specifikacija.....	23
5 lentelė. „Ijungti kadro atvaizdavimą po filtrų pritaikymo“ panaudojimo atvejo specifikacija.....	24
6 lentelė. „Ijungti kamerą“ panaudojimo atvejo specifikacija.....	24
7 lentelė. „Išjungti kamerą“ panaudojimo atvejo specifikacija.....	24
8 lentelė. „Keisti filtro tipą“ panaudojimo atvejo specifikacija.....	24
9 lentelė. „Keisti slenksčio reikšmę“ panaudojimo atvejo specifikacija.....	25
10 lentelė. „Keisti minimalias vaizdo objekto reikšmes“ panaudojimo atvejo specifikacija.....	25
11 lentelė. „Keisti maksimalias vaizdo objekto reikšmes“ panaudojimo atvejo specifikacija.....	25
12 lentelė. „Keisti spalvos filtro reikšmę“ panaudojimo atvejo specifikacija.....	25
13 lentelė. „Keisti piešimo storį“ panaudojimo atvejo specifikacija.....	26
14 lentelė. „Keisti piešimo spalvą“ panaudojimo atvejo specifikacija.....	26
15 lentelė. „Pasirinkti Power Point pateikčių failą“ panaudojimo atvejo specifikacija.....	26
16 lentelė. „Pradėti pateikti“ panaudojimo atvejo specifikacija.....	26
17 lentelė. „Keisti skaidrę“ panaudojimo atvejo specifikacija.....	27
18 lentelė. „Piešti“ panaudojimo atvejo specifikacija.....	27
19 lentelė. „Baigti pateikti“ panaudojimo atvejo specifikacija.....	27
20 lentelė. Apdorojimo laiko priklausomybė nuo skiriamųjų gebų ir filtrų tipų.....	43
21 lentelė. Vaizdo objektu išskyrimo priklausomybė nuo slenksčio ir lazerinių žymeklių.....	44
22 lentelė. Vaizdo objektų išskyrimo priklausomybė nuo minimalių ir maksimalių vaizdo objektų dydžio.....	46

Paveikslėlių sąrašas

1 pav. UPoint nuotolinis pultas.	10
2 pav. Piešinys ant pastato sienos naudojant Laser Tag	11
3 pav. Fizinė sistemos konfigūracija. Įrenginiai prieš ekraną.	12
4 pav. Fizinė sistemos konfigūracija. Įrenginiai už ekrano.	12
5 pav. Carsten Kirstein ir Heinrich Muller sistemos modelio diagrama.	14
6 pav. Jean-Francois Lapointe ir Guy Godin lazerio šviesos radimo architektūra.	15
7 pav. Vaizdo konvertavimas į dvejetainį atvaizdą.	16
8 pav. Dvejetainio vaizdo segmentavimas.	16
9 pav. Panaudojimo atvejų diagrama.	23
10 pav. Posistemų tarpusavio ryšiai.	28
11 pav. Klasių diagrama.	29
12 pav. „Keisti skiriamąją gebą“ sekų diagrama.	36
13 pav. „Įjungti kadro atvaizdavimą po filtrų pritaikymo“ sekų diagrama.	36
14 pav. „Įjungti kamerą“ sekų diagrama.	37
15 pav. „Išjungti kamerą“ sekų diagrama.	37
16 pav. „Keisti filtro tipą“ sekų diagrama.	37
17 pav. Keisti filtro reikšmę (- es) sekų diagrama.	38
18 pav. „Keisti piešimo storį“ sekų diagrama.	38
19 pav. „Keisti piešimo spalvą“ sekų diagrama.	38
20 pav. „Pasirinkti Power Point pateikčių failą“ sekų diagrama.	39
21 pav. „Pradėti pateiktį“ sekų diagrama.	39
22 pav. „Keisti skaidrę“ sekų diagrama.	40
23 pav. „Piešti“ sekų diagrama.	40
24 pav. „Baigti pateiktį“ sekų diagrama.	41
25 pav. Laiko ir skiriamosios gebos priklausomybės diagrama.	43
26 pav. Vaizdo objektų kiekio priklausomybės nuo slenksčio diagrama.	45
27 pav. Vaizdo objektų kiekio priklausomybės nuo ribų diagrama.	46
28 pav. Vaizdo objektai aplink žalią (kairėje) ir raudoną lazerį.	47

SUMMARY

A method for interactive presentation control

Computer – based presentations are now the most common way to teach courses and make scientific or business presentations. To support this, the majority of university classrooms and corporate conference rooms are equipped with high – definition projectors and computers.

To control presentation, teachers or participants of the conference must use the computer keyboard and mouse. Operations such as changing slide, highlighting phrase or graphic, performed by sitting near computer instead of directly communicating with the audience.

Laser markers – a tool often used during the presentation. This tool provides more freedom, but presenter must go to the computer to make an action.

This document describes a method for interactive presentation controlling using laser pointer, web cam, projector and image processing. In project part you will find the main technical and design issues of system that was developed during master course. Document also contains developed system test results and reviews.

TERMINŲ IR SANTRUMPŲ ŽODYNAS

- IDE – Integrated Development Environment santrumpa. Integruota kūrimo aplinka.
- API – API – Application Programming Interface santrumpa. Aplikacijų programavimo sąsaja.
- JPEG – Joint Photographic Experts Group santrumpa. Jungtinė fotografijos ekspertų grupė.
- PNG – Portable Network Graphics santrumpa. Portatyvi tinklo grafika.
- GDI – Graphics Device Interface santrumpa. Grafinių įtaisų sąsaja.
- SDK – Software Development Kit. Programinės įrangos kūrimo įrankių rinkinys.
- XACT – Cross – platform Audio Creation Tool. Įvairių platformų garso kūrimo įrankis.
- CASE – Computer – Aided Software Engineering santrumpa. Kompiuterizuotas programinės įrangos projektavimas.
- UML – Unified Modeling Language santrumpa. Unifikuota modeliavimo kalba.
- 3D – Three – Dimensional space santrumpa. Trimatė erdvė

IVADAS

Skaitmeninė pateiktis šiandien yra labiausiai paplitęs būdas mokyti kursų ir rengti mokslo bei verslo pristatymus. Norint paremti šią koncepciją, daugumoje universitetų klasės kambarių ir verslo konferencijų salėse yra įrengtos didelės raiškos projektoriai ir kompiuteriai.

Siekiant kontroliuoti pateiktis, dėstytojai ar konferencijos dalyviai turi naudoti kompiuterio klaviatūrą ir pelę. Operacijos, tokios, kaip kitos skaidrės įjungimas, frazės paryškimas ar grafiko nubraižymas, atliekamos dirbant kompiuteriu užuot tiesiogiai bendraujant su auditorija. Bevielės kompiuterinės pelės davė pranešėjui tam tikrą laisvę, t.y. kontroliuoti pateiktį tam tikru atstumu, bet jiems vis dar reikalingas kietas paviršius ir ribotas atstumas norint naudotis pele.

Lazeriniai žymekliai – dažnai naudojamas įrankis pateikčių metu. Šis įrankis suteikia didesnę laisvę, tačiau atlikti pateikčių valdymo veiksams pranešėjas privalo nueiti prie kompiuterio.

Panaudojant turimą pateikčių įrangą ir įdiegus vaizdo apdorojimo sistemą, galima pateiktims suteikti interaktyvumo. Interaktyvios sistemos dėka, pranešėjas pateikčių metu galėtų atsiriboti nuo kompiuterio, sutelkti visą dėmesį auditorijai ir tuo pačiu metu atlikti pagrindinius pateikčių valdymo veiksmus. Tokiu būdu pranešėjas pateikiamą medžiagą auditorijai pateiktų efektyviau ir įtaigiau.

Atliekant šį magistrinį darbą, sukurta „Fenix“ sistema, kuri internetinės kameros fiksuojamam vaizde, aptinka lazerinį žymeklį, nustato žymeklio padėtį ekrane ir, priklausomai nuo padėties, atlieka pagrindinius pateikčių metu naudojamus veiksmus.

Darbo metu buvo iškeltas **darbo tikslas** – suprojektuoti ir realizuoti sistemą, interaktyviam pateikčių valdymui.

Šiam tikslui pasiekti buvo iškelti **darbo uždaviniai**:

- Atlikti analogiškų sistemų analizę ir parinkti tinkamiausius sprendimus sistemos realizacijai.
- Surinkti sistemai keliamus reikalavimus ir atlikti reikalavimų analizę.
- Suprojektuoti sistemą naudojant CASE priemones.
- Sukurti suprojektuotą sistemą.
- Atlikti testavimą sukurtai programinei įrangai ir apibendrinti testavimo rezultatus.

1. ANALITINĖ DALIS

1.1. Problemos aprašymas

Šiais laikais dažnai dėstytojai, skaitydami paskaitas, ar konferencijos dalyviai naudoja vaizdo projektorius vaizdinei medžiagai vaizduoti. Tačiau tai apriboja pranešėjo laisvę – jis privalo būti prie kompiuterio, kad pakeistų skaidrę, paryškintų norimą tekstą ar nubrėžtų brėžinį. Tada kyla poreikis turėti sistemą, leidžiančią veiksmus atlikti nutolus nuo kompiuterio. Šiame skyriuje pateikiama egzistuojančių prototipų analizė, sistemų ir technologijų apžvalga.

1.2. Metodai

Yra keli metodai, išspręsti kilusiai problemai:

- **Nuotolinio valdymo pultas.** Pulto mygtukų paspaudimais atliekamos operacijos, tokios kaip: skaidrės keitimas, pelės žymeklio pozicijos keitimas, pelės klavišo paspaudimo imitavimas. Šis metodas yra riboto funkcionalumo – galima atlikti tiek operacijų, kiek yra užprogramuotų mygtukų, o kai kurias operacijas (pvz. piešimas) atlikti labai sudėtinga.
- **Lazerinio žymeklio sekimas.** Sekant žymeklio, nukreipto į ekraną, judėjimo trajektoriją ir poziciją, galima imituoti įvairias operacijas, tokias, kaip piešimas, skaidrių keitimas, pelės žymeklio valdymas, pelės mygtuko paspaudimas ir kitas naudingas operacijas. Tačiau tam reikalinga papildoma įranga fiksuojanti lazerio žymeklio judesius ir padėtį.

Naudojant lazerinį žymeklį yra platesnė interaktyvumo realizacijos galimybė ir yra tinkamesnis metodas darbo tikslui pasiekti.

1.3. Prototipai

UPoint – komercinis produktas, apjungiantis kompiuterinės pelės, nuotolinio valdymo pulto ir lazerinio žymeklio įrankis.



1 pav. UPoint nuotolinis pultas.

Sistemoje naudojami papildomi komponentai pulto 3D orientacijai nustatyti ir pelės žymeklio pozicijai sukongretinti. Pagrindinė prototipo funkcija – imituoti kompiuterinės pelės veiksmus pateikčių metu. [1]

Laser Tag 2.0. Graffiti Research Lab atvirojo kodo projektas, skirtas lazerio ir projektoriaus pagalba piešti ant įvairaus dydžio pastatų sienų.



2 pav. Piešinys ant pastato sienos naudojant Laser Tag.

Ši graffiti entuziastų sukurta sistema naudoja galingus lazerinius žymeklius, projektorius ir vaizdo kameras. Pagrindinė sistemos funkcija – piešimas ant didelių paviršių. [2]

Žemiau pateikta prototipų palyginimų lentelė.

1 lentelė. Prototipų palyginimas.

	UPoint	Laser Tag 2.0
Naudojama daugeliui vartotojų prieinama įranga	–	–
Turi papildomų funkcijų	–	–
Veikimas priklausomas nuo aplinkos	–	+
Skirta naudoti patalpoje	+	–

Pagrindinis prototipų trūkumas – daugeliui vartotojų brangi sistemos įranga. Taip pat sistemos yra riboto funkcionalumo – kompiuterinės pelės imitavimas arba piešimas.

1.4. Sistemos fizinė architektūra

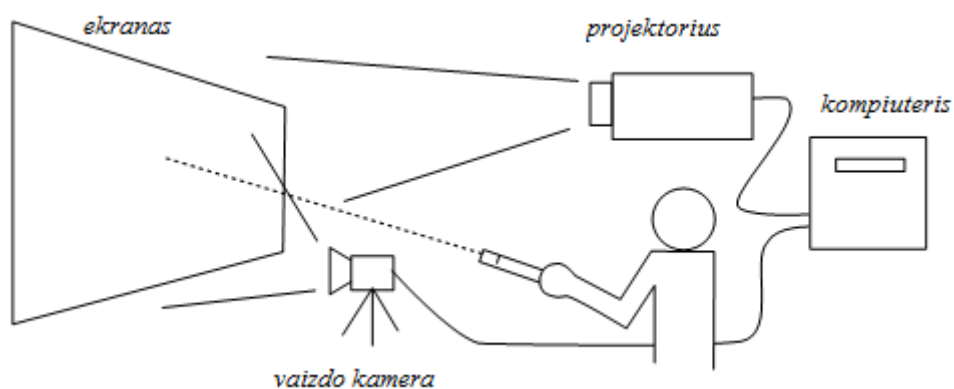
Lazerio šviesos suradimui ir interaktyvumui pateikčių valdyme suteikti naudojami tokie įrenginiai:

- Kompiuteris, vaizdo medžiagai analizuoti bei interaktyvios vartotojo sąsajos formavimui.
- Projektorius, atvaizduoti pateikčių medžiagą.
- Vaizdo kamera, vaizdo medžiagos fiksavimui.

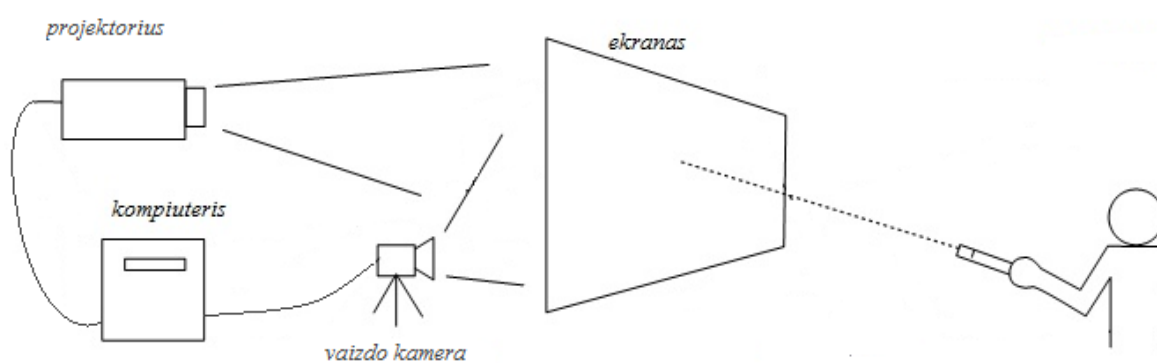
1.4.1. Fizinės sistemos konfigūracijos

Sistemoje naudojamus įrenginius galima išdėstyti dviem būdais:

- Įrenginiai išdėstyti prieš ekraną (3 pav.).
- Įrenginiai išdėstyti už ekrano (4 pav.).



3 pav. Fizinė sistemos konfigūracija. Įrenginiai prieš ekraną.



4 pav. Fizinė sistemos konfigūracija. Įrenginiai už ekrano.

1.4.2. Fizinų sistemos konfigūracijų palyginimas

2 lentelė. Fizinų sistemos konfigūracijų palyginimas.

	Įrenginiai prieš ekraną	Įrenginiai už ekrano
Nereikalingas specialus ekranas	+	–
Didesnė pranešėjo judėjimo aplinka	–	+
Pranešėjas neužstoja ekrano įrenginių atžvilgiu	–	+
Reikalinga mažesnė patalpa	+	–

Abi fizinės sistemų konfigūracijos turi privalumų ir trūkumų. Sistemos tikslesniam veikimui tinkamesnė fizinė sistemos konfigūracija, kai naudojami įrenginiai išdėstomi už ekrano. Taip sumažinama tikimybė atsirasti kliūtims analizuojamo vaizdo zonoje. Tačiau esant tokiam išdėstymui reikalingas specialus, dalinai skaidrus ekranas. Naudojant tokį ekraną sumažėja vaizdo kokybė ekrane ir apsunkina sistemai atlikti detalią vaizdo medžiagos analizę.

1.4.3. Sistemos įrenginių reikalavimai

Kompiuterio parametrų reikalavimai sistemoje nėra itin svarbūs. Svarbiausia, kad būtų galimybė prijungti išorinius įrenginius, t.y. projektorių ir internetinę vaizdo kamerą, bei būtų įdiegta įrenginių valdymo ir vaizdo apdorojimo programinė įranga.

Projektorius reikalingas pateikčių medžiagai atvaizduoti ekrane. Ekranas turėtų būti šviesus, kad atvaizduojama vaizdo medžiaga būtų aiškiai matoma ir suprantama, nors tai neįtakoja sistemos veikimo.

Lazerio žymeklis – dažnai naudojamas įrankis pateikčių metu, kuris fokusuoja intensyvų šviesos srautą į vieną tašką. Sistemoje galima naudoti žmogaus akims matomus arba nematomus (infraraudonusius) spindulius skleidžiantį lazerinį žymeklį. Jei naudojamas infraraudonųjų spindulių žymeklis, programinė įranga turi atvaizduoti jį ekrane, kitaip vartotojas nematys, kur fokusuojama pozicija.

Lazerio taškui ekrane aptikti reikalinga vaizdo kamera. Jos svarbus parametras yra vaizdo raiška. Esant mažai vaizdo raiškai vaizdo apdorojimo ir pozicijos nustatymo metu, žymeklio pozicija ekrane gali būti nustatyta netiksliai.

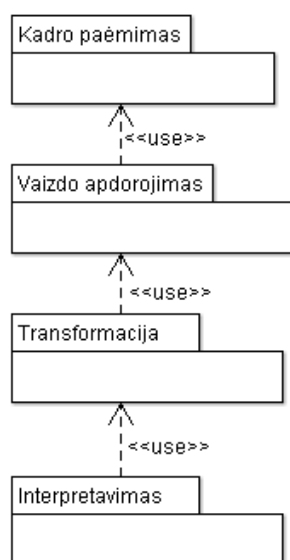
1.5. Sistemų apžvalga

Ryškaus lazerio taško atpažinimas vaizdo medžiagoje pagal spalvą atrodo nesudėtingas uždavinys, tačiau taip nėra. Lazerio šviesa yra per intensyvi kameros vaizdo jutikliams, todėl jie tampa prisotinti ir vaizdo medžiagoje taškas yra baltos spalvos, apsuptas blankiu lazerio spalvos apskritimu. Tai apsunkina taško paiešką pagal spalvą, ypač, kai lazerio žymeklis naudojamas ant šviesių paviršių, tokių, kaip baltas ekranas. [3]

Netiksli vaizdo medžiagos analizė, nustatant lazerio taško poziciją, tiesiogiai įtakoja interaktyvumų pateikčių valdymą. Todėl pagrindinė sistemos dalis yra pirminė vaizdo medžiagos analizė, išskiriant lazerio tašką iš kitų vaizdo medžiagoje esančių objektų.

1.5.1. Carsten Kirstein, Heinrich Müller sistemos modelis

Carsten Kirstein ir Heinrich Müller sistema suskirstyta į keturias dalis (5 pav.)



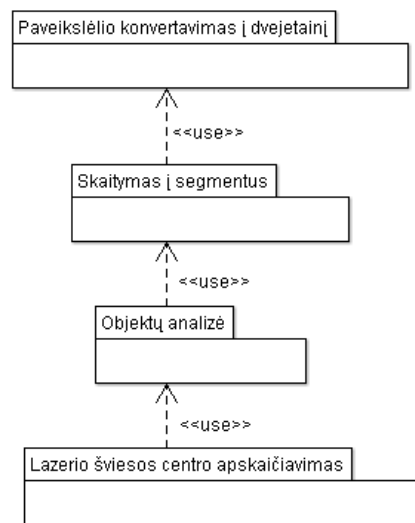
5 pav. Carsten Kirstein ir Heinrich Muller sistemos modelio diagrama.

- **Kadro paėmimas.** Pirmoje dalyje, naudojant vaizdo imtuvą, paimamas kadras analizei.
- **Vaizdo apdorojimo** dalyje judesio algoritmu aptinkamos kadre pakitusios vaizdo dalys, lyginant su pastaraisiais kadrais. Dalys, kuriose buvo aptikti pakitimai, analizuojamos vaizdui jose atpažinti ir išskirti ieškomus objektus. [4] Aptiktiems objektams vykdoma analizė pozicijai kadre nustatyti.
- **Transformacija.** Šioje dalyje apdorojamos objektų pozicijos kadre ir apskaičiuojama padėtis vaizduojamojoje dalyje (ekrane).
- **Interpretacijos** dalyje nustatytose pozicijose imituojami veiksmai.

Sistema vaizdui aptikti naudoja tik dalį vaizdo medžiagos, kurioje buvo pakitimai per pastaruosius keletą kadrų. Po judesio filtro pritaikymo, naudojami lazerio šviesos šablonai tose dalyse, kur buvo aptikti pakitimai ir nustatomi ieškomi objektai. Kadangi naudojami lazerio šviesos šablonai, dažnai besikeičiančiame fone gali būti neteisingai nustatytos lazerio šviesos pozicijos. Taip pat, esant skirtingai spalvos, dydžio ar intensyvumo lazerio šviesai, galima neaptikti ieškomo objekto.

1.5.2. Jean-Francois Lapointe, Guy Godin sistemos modelis

Kaip ir aukščiau apžvelgtoje sistemoje, ši sistema taip pat suskirstyta į keturias dalis, skiriasi tik vaizdo apdorojimo dalis.



6 pav. Jean-Francois Lapointe ir Guy Godin lazerio šviesos radimo architektūra.

Vaizdo apdorojimo dalis skirstoma į keturis etapus:

- **Vaizdo konvertavimas į dvejetainį skaitmeninį vaizdą** (*angl. binarize*). Lazerio šviesos radimui, pirmiausiai, gautasis kadras konvertuojamas į dvejetainį atvaizdą (7 pav.), taikant didžiausią ryškumo (raudonos, žalios ir mėlynos spalvų) ribą [5], kokią projektorius gali atvaizduoti ekrane. Prieš pradėdant naudoti sistema, atliekamas testas ribai nustatyti. Testo metu projektorius vaizduoja ryškiausių spalvų vaizdą, tuo metu vaizdo kamera fiksuojamas vaizdas ir, atlikus šviesų intensyvumo analizę vaizde, nustatomos didžiausio ryškumo ribos.



7 pav. Vaizdo konvertavimas į dvejetainį atvaizdą.

- **Skaidymas į segmentus.** Gautas dvejetainis atvaizdas skaidomas į segmentus, t.y. išskiriamos vaizdo dalys, kuriose aptikti didžiausi ryškumai.



8 pav. Dvejetainio vaizdo segmentavimas.

- **Dvejetainių objektų analizė.** Dvejetainių objektų analizės etape nagrinėjami išskirti vaizdo segmentai, taikant algoritmą. Remiantis prielaida, jog lazerio taškas dvejetiniame vaizde yra grupė taškų, esančių labai arti vienas kito, išrenkamas segmentas, kuriame yra ieškomas objektas.
- **Lazerio šviesos centro apskaičiavimas.** Paskutiniajame etape apskaičiuojamas nustatyto ieškomo objekto centras ir nustatomos centro koordinatės.

Sistema konvertuoja kadrą, gautą iš vaizdo kameros, į dvejetainį kadro atvaizdą, remiantis nustatyta ryškumo riba. Analizuojant kadro atvaizdą, nustatoma lazerio šviesos pozicija. Tačiau sistemoje nėra atsižvelgta į galimus aplinkos pokyčius (pvz. saulės šviesa) ir atspindžius. Atsiradus aplinkoje daugiau natūralios šviesos, lazerio šviesos ryškumas sumažėja ir, konvertuojant į dvejetainį atvaizdą, ryškumo gali neužtekti pasiekti nustatytai ribai. Lazerio šviesos atspindžio atveju, sistema gali nustatyti, jog atspindys yra ieškomasis taškas. Tokie pokyčiai gali įtakoti klaidingą objekto pozicijos nustatymą ir sistemos veikimo netikslumą.

1.6. Technologijų apžvalga

Interaktyviam pateikčių valdymui reikalinga grafinė vartotojo sąsaja. Kadangi Lietuvoje mokymo įstaigos dažniausiai naudoja Microsoft Windows operacinę sistemą, realizavimui numatomos naudoti technologijos šiai operacinei sistemai.

1.6.1. DirectX

DirectX – aplikacijų programavimo sąsajų (API) rinkinys, skirtas daugialypės terpės užduočių tvarkymui, ypač tų žaidimų kūrimui, kurie vėliau bus žaidžiami naudojant Microsoft produktus, tokius kaip Windows, Xbox ir Xbox 360. [6]

Nors DirectX yra bibliotekų kolekcija, skirta žaidimų programavimui Microsoft Windows operacinei sistemai, tačiau ji yra naudojama ne tik žaidimams, bet ir nemažoje dalyje vaizdą, garsą apdorojančių bei išvedančių programų.

DirectX platinamas dviem programų paketais: Runtime (skirtu vartotojams) bei SDK (skirtu programuotojams). Runtime yra visose naujesnėse Microsoft Windows versijose pridedamas kartu su operacine sistema, o senesnėms šios operacinės sistemos versijoms galima šį paketą įdiegti kaip atskirą programą.

DirectX biblioteka susideda iš keleto skirtingų komponentų:

- **DirectX Graphics.** Susideda iš kelių API:
 - **DirectDraw.** Skirtas atvaizduoti dvimatei grafikai.
 - **Direct3D (D3D).** Skirtas atvaizduoti trimatei grafikai.
- **DirectInput.** Skirtas apdoroti duomenis, gautus iš klaviatūros, pelės, vairasvirtės ar kito žaidimo valdiklio. Komponentas yra pamažu keičiamas į XBox kūrėjų komandos sukurtą XInput biblioteką.
- **DirectPlay.** Skirtas duomenų keitimuisi tinklu.
- **DirectSound.** Skirtas groti, bei įrašyti garsą (yra pamažu keičiamas į XACT biblioteką).
 - **DirectSound3D (DS3D).** Skirtas groti trimačius garsus.
- **DirectMusic.** Skirtas groti garso takelius.
- **DirectX Media.** Susideda iš kelių API:
 - **DirectAnimation.** Skirtas interneto animacijai kurti.
 - **DirectShow.** Skirtas daugialypės terpės atkūrimui ir transliacijai.
 - **DirectX Transform.** Skirtas internetinių puslapių interaktyvumui kurti.

- **DirectX Media Objects.** Skirtas tinklu transliuojamiems objektams, tokiems kaip koduotojai, dekoderiai, bei efektai. [7]

DirectDraw komponentas sukurtas tam, kad išnaidotų vaizdo plokštės resursus ir GDI grafines galimybes bei paspartintų grafines sąsajos veikimą. [8, 9] Norint padidinti interaktyvią pateikčių valdymo sistemą, geriau naudoti šį komponentą.

1.6.2. GDI ir GDI+

Grafinių įtaisų sąsaja (*angl. Graphics Device Interface; GDI*) – viena Microsoft Windows operacinės sistemos branduolio dalių, naudojama atvaizduoti grafinius objektus (pvz. linijas, taškus, spalvas, tekstą) išvedimo įrenginiuose (pvz., monitoriuje, spausdintuve). Dažniausiai yra naudojamas piešti įvairius dvimačius objektus – langus, kitus paprastus vartotojo sąsajos elementus.

Nuo Windows XP versijos, su operacine sistema diegiamas naujos kartos GDI+ komponentas, papildantis senesnę GDI. GDI+ prie tradicinės GDI grafinių galimybių prideda galimybes naudoti alfa permatomumą, perėjimą, suliejimą, sudėtingesnę piešimo kreivių nurodymą, vidinį šiuolaikinių grafikos bibliotekų palaikymą (pvz. jpeg, png), giminingų (*angl. affine*) transformacijų palaikymą dvimačiams vaizdams. [10]

Windows operacinės sistemos komponentėje (Microsoft .NET) GDI+ galimybėms naudoti yra skirtos System.Drawing vardų srities klasės. [11]

Interaktyviai vartotojo sąsajai kurti, GDI bibliotekos yra tinkamos, jei sąsajos grafines detales nėra dažnai keičiamos. Interaktyviai pateikčių valdymo sistemai nereikalingas greitas grafinių dalių reagavimas į sistemos pokyčius. Todėl ši technologija yra dalinai tinkama darbo realizavimui.

1.6.3. AForge.NET Framework

AForge.NET yra bibliotekų bazė skirta C# programavimo kalbai. Bibliotekos skirtos programų kūrėjams ir mokslininkams kompiuterinio matymo ir dirbtinio intelekto srityse - vaizdo apdorojimas, neuroniniai tinklai, genetiniai algoritmai, mašinų apmokymas, robotams ir t.t. [12]

Bibliotekų bazė susideda iš tokio bibliotekų rinkinio:

- **AForge.Imaging** – biblioteka su vaizdo apdorojimo šablonais ir filtrai.
- **AForge.Vision** – kompiuterinio matymo biblioteka.
- **AForge.Neuro** – neuroninių tinklų skaičiavimo biblioteka.
- **AForge.Genetic** – vystymosi programavimo biblioteka.
- **AForge.Fuzzy** – migloto skaičiavimai biblioteka.

- **AForge.MachineLearning** – mašinų apmokymo biblioteka.
- **AForge.Robotics** – biblioteka palaikanti tam tikrus robotų rinkinius.
- **AForge.Video** – biblioteka vaizdo apdorojimas.

Bibliotekų bazė yra atvirojo kodo, todėl galima iš kūrėjų svetainės parsisiųsti išėjusius tekstus ir pritaikyti savom reikmėm. Šios bibliotekos yra nuolat naujinamos ir papildomos naujomis funkcijomis.

1.6.4. Programavimo kalbos ir IDE alternatyvos

Programavimo kalbos ir integruotos kūrimo aplinkos (*angl. Integrated Development Environment; IDE*) pasirinkimas lemia programavimo patogumą, galimybių gausą ir kuriamos sistemos realizacijos laiką. Buvo renkamos tarp šių programavimo kalbų ir jas palaikančių IDE:

- **Java** – programavimo sistemoje programuotojo parašytas kodas kompiliuojamas ne į procesoriui specifinę, o į tarpinę formą. Ši tarpinė forma nepriklauso nuo procesoriaus tipo ar operacinės sistemos, todėl tinkama vykdyti įvairiose aplinkose. Masyvo rėžių tikrinimas Java programuotojams padeda išvengti sunkiai aptinkamų programavimo klaidų. [13] Eclipse IDE turi gerai išvystytą tekstų redaktorių (jau renkant kodą automatiškai nurodomos trivalios klaidos), klasių, kintamųjų ir metodų automatinį pervardinimą. Inkrementinis kompiliavimas (po nedidelio pakeitimo kompiliuojama tik pakeista dalis) sudėtinguose projektuose pagreitina darbą. [14]
- **C#** – objektiškai orientuota programavimo kalba, sukurta Microsoft kompanijoje kaip dalis .NET iniciatyvos. Kalba paremta C++ bei Java kalbomis. Kalba kurta balansuojant tarp galingumo (C++ įtaka) bei greito programavimo (Java, Visual Basic įtaka). [15] Visual Studio .NET IDE galima pasirinkti šią programavimo kalbą ir su integruotais įrankiais patogiai ir greitai realizuoti projektą.

Programavimo kalbos panašios, tačiau Visual Studio .NET (IDE darbui su C#) siūlo labiau integruotus sprendimus Windows operacinės sistemos aplikacijų kūrimui.

1.7. Išvados

- Atlikus panašių sistemų apžvalgą, išsiaiškinta, jog pagrindinis sistemų uždavinys yra pirminis vaizdo medžiagos apdorojimas. Pirminio vaizdo apdorojimo rezultatas yra lazerio žymeklio pozicijos identifikavimas. Tam tikslui numatoma naudoti AForge.NET vaizdo apdorojimo bibliotekas.
- Microsoft Windows operacinėje sistemoje dažnai besikeičiančiai grafinei vartotojo sąsajai kurti, naudojama DirectX komponentai. Kadangi sistemos realizacijoje vartotojo sąsaja neturės sudėtingų grafinių elementų ir nebus reikalingas greitas grafinis reagavimas, numatoma naudoti GDI+ komponentą.
- Sistemos realizacijai pasirinkta C# programavimo kalba ir ją palaikanti Microsoft Visual Studio integruota kūrimo aplinka. Šie įrankiai pasirinkti dėl platesnių integruotų sprendimų Windows operacinės sistemos aplikacijų kūrimui.

2. PROJEKTINĖ DALIS

2.1. Projekto tikslas

Projekto tikslas – suprojektuoti ir realizuoti sistemą, leidžiančią pateikčių metu, naudojant lazerinį žymeklį, atlikti tam tikras pateikčių metu naudojamas funkcijas. Sistemos projektavimui naudoti CASE priemonės. Nustatyti, kaip turi būti realizuota vartotojo sąsaja, kokios reikalingos programinės klasės, realizuojančios tiek posistemų funkcijas, tiek vartotojo sąsają ir t.t. Toliau skyriuje pateikiamas sistemos projektas, reikalavimų specifikacija, duomenų struktūra, elgsenos medelis ir programinių modulių specifikacija.

2.2. Naudotas projektavimo metodas

Sistema projektuota laikantis šios schemos:

1. Probleminės srities apžvalga.
2. Reikalavimų analizė.
3. Panaudojimo atvejų išskyrimas.
4. Prototipo kūrimas.
5. Sistemos kūrimas.
6. Testavimas.
7. Klaidų taisymas.

Sistemos projektavimui naudotas atvirojo kodo ArgoUML v0.28 modeliavimo paketas, palaikantis visas standartines UML 1.4 diagramas. [16] Jo pagalba realizuoti visi kuriamos sistemos modeliai.

2.3. Reikalavimų projektuojamai sistemai specifikacija

2.3.1. Bendras aprašymas

2.3.1.1. Produkto perspektyva

„Fenix“ programa, panaudota pateikčių metu, suteikia pranešėjui daugiau laisvės ir nevaržomo bendravimo su auditorija. Sistemai nereikalinga sudėtinga ir brangi įranga, todėl programa galima naudotis be didelių išlaidų.

Programos funkcionalumas gali būti praplėstas pagal poreikius perprogramuojant už programos funkcionalumą atsakingus programinius modulius. Tobulinant programą, galima sukurti nustatymų modulį, leidžiantį pasirinkti norimas atlikti funkcijas iš pateikto funkcijų sąrašo.

2.3.1.2. Produkto funkcijos

Pagrindinės programos funkcijos yra:

- Vaizdo medžiagos apdorojimas, išskiriant lazerinio žymeklio poziciją ekrane.
- Žymeklio pozicijos projektavimas į vartotojo aplinką.
- Pagrindinių pateikčių funkcijų imitavimas.
- Vartotojo pasirinktų nustatymų išsaugojimas, atstatymas ir naudojimas programoje veikimo eigoje.

2.3.1.3. Vartotojo charakteristikos

Vartotojas turi turėti sistemoje naudojamą kompiuterinę įrangą t.y. kompiuterį, internetinę kamerą, projektorius ir lazerinį žymeklį. Kompiuteryje turi būti įdiegta .NET Framework – Microsoft Windows operacinės sistemos komponentas. Programos „Fenix“ diegimo vedlys, neaptikęs .NET Framework komponento vartotojo kompiuteryje, pasiūlys parsisiųsti ir įdiegti reikiamą komponento versiją.

2.3.1.4. Bendri apribojimai

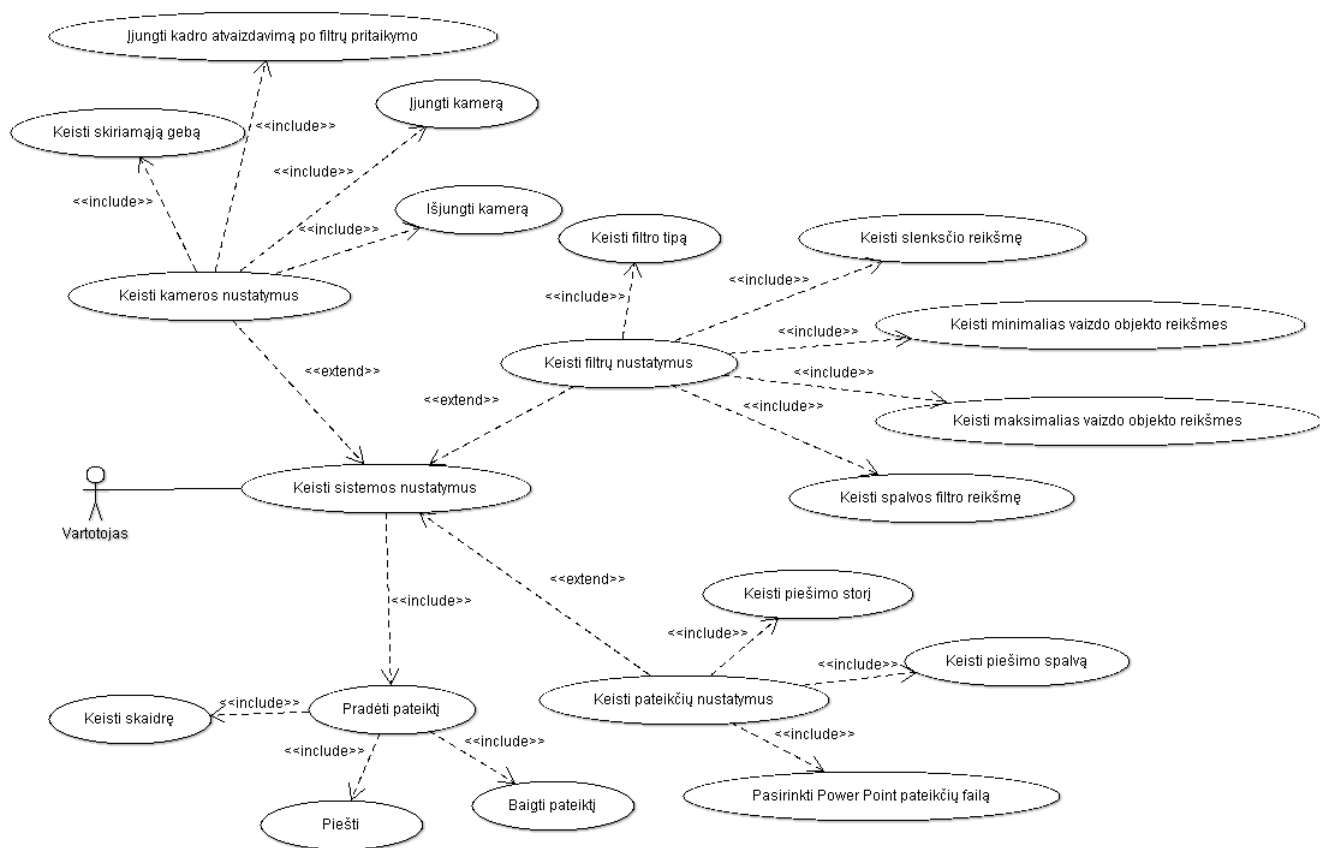
Vienu metu negalima naudoti kelių lazerinių žymeklių – sistemoje atliekama vieno lazerinio žymeklio paieška vaizdo kadre. Sistemoje nenumatyta vartotojų registracija, tad visi parinkti nustatymai yra bendri visiems vartotojams.

2.3.2. Specifiniai reikalavimai

2.3.2.1. Funkciniai reikalavimai

- **Pateikčių paleidimas** (*angl. Slide show*) – turi būti galimybė pradėti pateiktį iš programos.
- **Nustatymų keitimo galimybė** – pasirinkti norimą pateikčių failą, internetinės kameros, filtrų ir kitus sistemos parametrus.
- **Standartinės pateikčių funkcijos** – programa turi atlikti pagrindines pateikčių funkcijas, tokias kaip perėjimas prie sekancios skaidrės, grįžimas skaidre atgal ir kitas.

Žemiau pateikiama panaudojimo atvejų diagrama (USE CASE) ir specifikacijos (9 pav.).



9 pav. Panaudojimo atvejų diagrama.

3 lentelė. „Keisti sistemos nustatymus“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti sistemos nustatymus“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – galimybė pakeisti sistemos nustatymus	
„Prieš“ sąlyga	-	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Paleidžiama programa	Atveriamas nustatymų langas	
„Po“ sąlyga	Pateikiama galimų sistemos nustatymų vartotojo sąsaja	
Veiklos taisyklės	-	
Ryšiai su kitais PA	Įeina „Keisti kameros nustatymus“, „Keisti filtrų nustatymus“, „Keisti pateikčių nustatymus“. Apima „Pradėti pateiktį“	

4 lentelė. „Keisti skiriamąją gebą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti skiriamąją gebą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti internetinės kameros skiriamąją gebą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Keisti skiriamąją gebą	Gaunamas skiriamųjų gebų sąrašas	
„Po“ sąlyga	Reikia pasirinkti vieną variantą iš sąrašo	
Ryšiai su kitais PA	Apima „Keisti kameros nustatymus“	

5 lentelė. „Įjungti kadro atvaizdavimą po filtrų pritaikymo“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Įjungti kadro atvaizdavimą po filtrų pritaikymo“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti video atvaizdavimą po filtrų pritaikymo	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Įjungti kadro atvaizdavimą po filtrų pritaikymo	Video atvaizdavimo komponentui perduodami kadrai po filtrų pritaikymo	
Ryšiai su kitais PA	Apima „Keisti kameros nustatymus“	

6 lentelė. „Įjungti kamerą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Įjungti kamerą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – įjungti kamerą ir pradėti video atvaizdavimą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Įjungiamą kamera	Perduodamas įjungimo signalas internetinei kamerai	
Ryšiai su kitais PA	Apima „Keisti kameros nustatymus“	

7 lentelė. „Išjungti kamerą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Išjungti kamerą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – išjungti kamerą ir sustabdyti video atvaizdavimą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Išjungiamą kamera	Perduodamas išjungimo signalas internetinei kamerai	
Ryšiai su kitais PA	Apima „Keisti kameros nustatymus“	

8 lentelė. „Keisti filtro tipą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti filtro tipą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti kadro filtravimo tipą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Keisti filtro tipą	Gaunamas filtrų sąrašas	
Ryšiai su kitais PA	Apima „Keisti filtrų nustatymus“	

9 lentelė. „Keisti slenksčio reikšmę“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti slenksčio reikšmę“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti slenksčio filtro reikšmę	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keisti slenksčio reikšmę		Pakeičiamas filtro parametras
Veiklos taisyklės	Keičiama reikšmė didesnė už nulį	
Ryšiai su kitais PA	Apima „Keisti filtrų nustatymus“	

10 lentelė. „Keisti minimalias vaizdo objekto reikšmes“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti minimalias vaizdo objekto reikšmes“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti minimalaus vaizdo objekto aukštį ir ilgį	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keičiamos minimalios vaizdo objekto reikšmės		Keičiami vidiniai parametrai
/Veiklos taisyklės	Keičiamos reikšmės didesnės už vienetą	
Ryšiai su kitais PA	Apima „Keisti filtrų nustatymus“	

11 lentelė. „Keisti maksimalias vaizdo objekto reikšmes“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti maksimalias vaizdo objekto reikšmes“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti maksimalaus vaizdo objekto aukštį ir ilgį	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keičiamos maksimalaus vaizdo objekto reikšmės		Keičiami vidiniai parametrai
Veiklos taisyklės	Keičiamos reikšmės didesnės už vienetą	
Ryšiai su kitais PA	Apima „Keisti filtrų nustatymus“	

12 lentelė. „Keisti spalvos filtro reikšmę“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti spalvos filtro reikšmę“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti filtruojamo spalvų kanalo spalvą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keisti spalvos filtro reikšmę		Gaunamas spalvų kanalų sąrašas
„Po“ sąlyga	Pasirinktas viena reikšmė iš pateikto sąrašo	
Ryšiai su kitais PA	Apima „Keisti filtrų nustatymus“	

13 lentelė. „Keisti piešimo storį“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti piešimo storį“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti piešimo storį	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keičiamas piešimo storis		Keičiamas vartotojo sąsajos pieštuko parametras
Veiklos taisyklės	Keičiama reikšmė turi būti didesnė už nulį	
Ryšiai su kitais PA	Apima „Keisti pateikčių nustatymus“	

14 lentelė. „Keisti piešimo spalvą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti piešimo spalvą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pakeisti piešimo spalvą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Keičiama piešimo spalva		Atveriamą spalvų paletę
Ryšiai su kitais PA	Apima „Keisti pateikčių nustatymus“	

15 lentelė. „Pasirinkti Power Point pateikčių failą“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Pasirinkti Power Point pateikčių failą“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pasirinkti norimą naudoti pateikčių failą	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Pasirinkti Power Point pateikčių failą		Atveriamas ppt failų pasirinkimo dialogas
Ryšiai su kitais PA	Apima „Keisti pateikčių nustatymus“	

16 lentelė. „Pradėti pateiktį“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Pradėti pateiktį“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – pradėti interaktyvia pateiktį	
„Prieš“ sąlyga	Aktyvuotas nustatymų langas	
Vartotojo veiksmas		Sistemos reakcija (jei ji turi būti)
Pradėti pateiktį		Konvertuojamos pateikčių skaidrės
Veiklos taisyklės	Power Point faile yra bent viena skaidrė	
Ryšiai su kitais PA	Apima „Keisti sistemos nustatymus“	

17 lentelė. „Keisti skaidrę“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Keisti skaidrę“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – perjungti skaidrę pirmyn arba atgal	
„Prieš“ sąlyga	Pradėta pateiktis	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Keičiama skaidrė	Atvaizduojama sekanti arba praėjusi skaidrė	
Veiklos taisyklės	Egzistuoja sekanti arba praėjusi skaidrė	
Ryšiai su kitais PA	Apima „Pradėti pateiktį“	

18 lentelė. „Piešti“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Piešti“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – įjungti piešimo režimą	
„Prieš“ sąlyga	Pradėta pateiktis	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Įjungiamas piešimo režimas	Pradedamas grafinių vaizdų kūrimas	
Ryšiai su kitais PA	Apima „Pradėti pateiktį“	

19 lentelė. „Baigti pateiktį“ panaudojimo atvejo specifikacija.

PA pavadinimas	„Baigti pateiktį“	
Aktorius	Vartotojas	
Aprašymas	Tikslas – baigti interaktyvią pateiktį	
„Prieš“ sąlyga	Pradėta pateiktis	
Vartotojo veiksmas	Sistemos reakcija (jei ji turi būti)	
Baigti pateiktį	Užveriamas vartotojo sąsajos langas	
Ryšiai su kitais PA	Apima „Pradėti pateiktį“	

2.3.2.2. Nefunkciniai reikalavimai

- **Vėlinimas** – pageidautina, kad laiko tarpas nuo kadro užfiksavimo internetinėje kameroje, žymeklio pozicijos nustatymo ir atvaizdavimo/funkcijos suvykdymo būtų kuo mažesnis. Vėlinimas turi būti minimalus, kad netrikdytų pateikčių eigos.
- **UML panaudojimas** – sistemos projektas ir architektūra turi būti specifikuota panaudojant UML diagramas.
- **.NET Framework panaudojimas** – programa turi būti realizuota naudojant .NET Framework. Vienas iš pagrindinių projekto uždavinių yra išsiaiškinti .NET Framework panaudojimo galimybes, apdorojant vaizdo medžiagą.

2.4. Sistemos architektūra

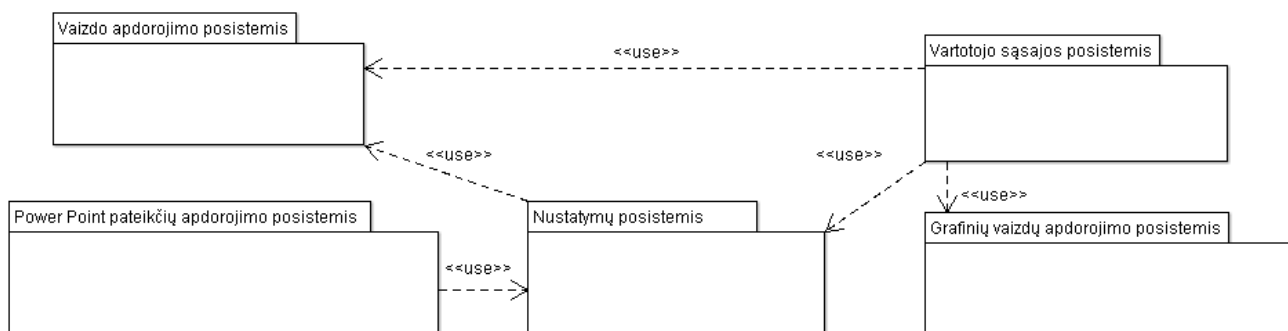
2.4.1. Architektūros apribojimai

Architektūros sprendimai buvo priimti remiantis šiais apribojimais:

- Sistema kuriama naudojant C# programavimo kalbą.
- Sistema veiks Windows 2000/XP ir naujesnėje operacinėje sistemoje.
- Sistema su internetine kamera bendraus per kameros valdiklį (*angl. driver*).
- Vartotojo sąsaja bus kuriama, naudojant .NET Framework komponentus.

2.4.2. Projektuojamos sistemos architektūra

Kuriama programa susideda iš kelių posistemių: vaizdo apdorojimo, Power Point pateikčių apdorojimo, nustatymų, vartotojo sąsajos ir grafinių vaizdų apdorojimo posistemių. Posistemių tarpusavio ryšiai pavaizduoti žemiau pateiktoje diagramoje (10 pav.).

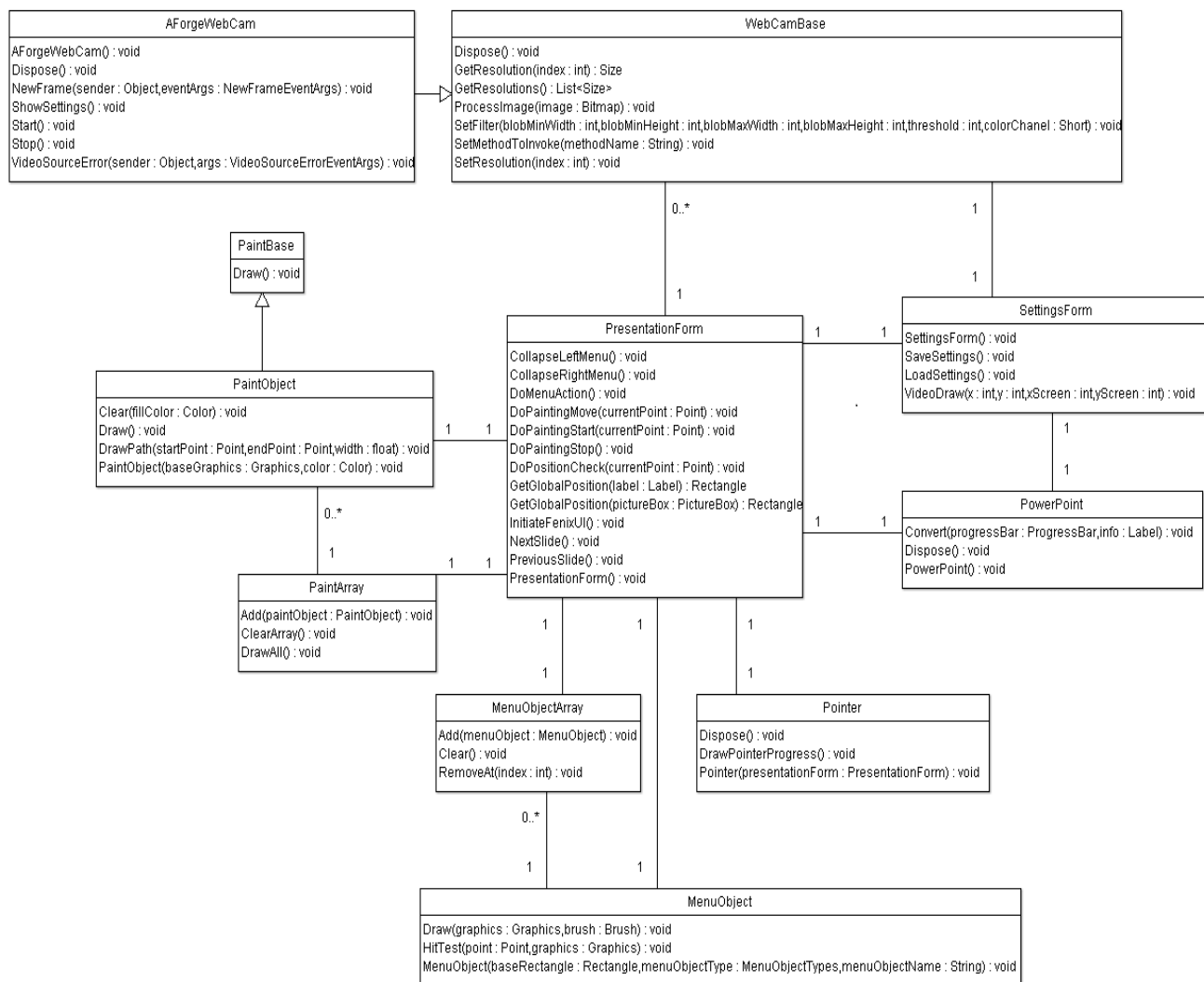


10 pav. Posistemių tarpusavio ryšiai.

- **Vaizdo apdorojimo posistemis** atsakingas už internetinės kameros valdymą, pirminį vaizdo kadro, gauto iš kameros, apdorojimą ir lazerinio žymeklio pozicijos aptikimą.
- **Power Point pateikčių apdorojimo posistemyje** apdorojamas Power Point pateikčių failas ir paruošiamos skaidrės atvaizdavimui vartotojo sąsajoje.
- **Nustatymų posistemyje** atliekami veiksmai susiję su nustatymų parinkimu, išsaugojimu ir atstatymu. Nustatymų posistemis naudoja vaizdo apdorojimo posistemo funkcijas.
- **Grafinių vaizdų apdorojimo posistemyje** atliekami veiksmai susiję su vartotojo sąsajoje grafinių vaizdų atvaizdavimu.
- **Vartotojo sąsajos posistemyje** atliekami visi veiksmai, skirti atvaizduoti pateikčių vartotojo sąsają. Vartotojo sąsaja naudoja vaizdo apdorojimo, nustatymų ir grafinių vaizdų posistemių modulius.

2.5. Duomenų struktūra

Programa nenaudoja jokios duomenų bazės. Duomenų šaltiniai yra du: nustatymų failas ir internetinė kamera. Nustatymų faile saugomi vartotojo pasirinkti nustatymai, kad paleidus programą nereikėtų iš naujo suvedinėti norimų parinkčių. Internetine kamera fiksuojami vaizdo kadrai, kurie yra apdorojami ir atliekami atitinkami sistemos veiksmai. Žemiau pateikta klasių diagrama (11 pav.) (detalesnę klasių diagramą žiūrėti 1 priede).



11 pav. Klasių diagrama.

































2.6. Programinių modulių specifikacijos

„Fenix“ sistemos architektūra sudaryta iš vaizdo apdorojimo, Power Point pateikčių apdorojimo, nustatymų, vartotojo sąsajos ir grafinių vaizdų apdorojimo posistemių. Žemiau pateiktose poskyriuose detalizuotos posistemių programinių modulių specifikacijos.











2.6.1. Vaizdo apdorojimo posistemis

Vaizdo apdorojimo posistemį sudaro klasės, reikalingos valdyti internetinę kamerą, jos nustatymus, vaizdo kadro filtrų parametrus. Šiame posistemyje atliekamas lazerio aptikimas vaizdo kadre, koordinacių nustatymas ir jų perdavimas nustatymų ir vartotojo sąsajos posistemiams.

WebCamBase klasėje yra baziniai parametrai reikalingi kameros valdymui, vaizdo kadro filtrai ir funkcija išskirianti vaizdo objektus, bei jų koordinates.






public class WebCamBase	
Modulio vardas	Funkcinis aprašymas
▷  Dispose	Naikina objektą
▷  GetResolution	Gražina skiriamąją gebą pagal indeksą
▷  GetResolutions	Gražina siriamųjų gebų sąrašą
▷  ProcessImage	Analizuoja kadra, išskiria lazerio poziciją ir koordinates
▷  SetFilter	Keičia filtrų parametrus
▷  SetMethodToInvoke	Nustato, kokį metodą iššaukti aptikus lazerį
▷  SetResolution	Nustato internetinės kameros skiriamąją gebą
▷  ShowSettings	Atveria internetinės kameros nustatymų langą
▷  Start	Ijungia internetinę kamerą
▷  Stop	Išjungia internetinę kamerą
 DisplayFilteredVideo	Rodyti kadrus po filtrų pritaikymo
 MaxBlobSize	Didžiausias vaizdo objekto dydis
 MinBlobSize	Mažiausias vaizdo objekto dydis
 Parent	Tėvinė klasė, kurioje iškviečiamas metodas
 VideoBox	Vaizdo panelė
 WebCamType	Klasės tipas
 _bitmapData	Kadro duomenys
 _blobCounter	AForge klasės objektas išskiriantis vaizdo objektus kadre
 _blobs	Vaizdo objektai kadre
 _coordinateYConst	Y koordinacių konstanta
 _coordinateXConst	X koordinacių konstanta
 _extractFilter	AForge spalvų filtro objektas
 _grayscale	Nurodo ar naudojama spalvų ar bespalvis filtras
 _grayscaleFilter	AForge bespalvis filtras
 _imageFiltered	Kadras po filtrų pritaikymo
 _imageOriginal	Kadras gautas iš internetinės kameros
 _imageTemp	Laikina kadro kopija
 _methodInf	Metodo informacija
 _resolutions	Skiriamųjų gebų sąrašas
 _thresholdFilter	AForge maksimumo filtro objektas
 _typExternal	Klasės tipas
 _videoDimension	Esamoji skiriamoji geba

AForgeWebCam klasėje AForge .NET Framework klasių objektais valdoma internetinė kamera.

public class AForgeWebCam : WebCamBase	
Modulio vardas	Funkcinis aprašymas
▷  AForgeWebCam	Konstruktorius
▷  Dispose	Naikinti objektą
▷  NewFrame	Naujo kadro gavimo įvykis
▷  ShowSettings	(Paveldėta iš WebCamBase) Atveria internetinės kameros nustatymus
▷  Start	(Paveldėta iš WebCamBase) Įjungia internetinę kamerą
▷  Stop	(Paveldėta iš WebCamBase) Išjungia internetinę kamerą
▷  VideoSourceError	Internetinės kameros klaidos įvykis
 _tempImage	Laikinas kadras
 _videoSource	Internetinės kameros šaltinis
 _videoSources	Internetinių kamerų šaltinių sąrašas

2.6.2. Power Point pateikčių apdorojimo posistemis




Power Point pateikčių apdorojimo posistemyje yra viena klasė *PowerPoint*, kurioje apdorojamas vartotojo pasirinktas pateikčių failas. Failo apdorojimo metu nustatomas skaidrių kiekis ir visos skaidrės, atsižvelgiant į kompiuterio ekrano skiriamąją gebą, konvertuojamos į jpeg tipo vaizdo failus.

public class PowerPoint	
Modulio vardas	Funkcinis aprašymas
▷  Convert	Konvertuoja skaidres į jpg failus
▷  Dispose	Naikinti objektą
▷  PowerPoint	Konstruktorius
 _count	Skaidrių kiekis
 _presentationFile	Kelias iki Power Point ppt failo

2.6.3. Vartotojo sąsajos posistemis

Šiame posistemyje yra klasės suteikiančios interaktyvumą pateikčių metu.

Menu juostų funkcionalumas, skaidrių atvaizdavimas, piešimo valdymas, veiksmo atlikimas pagal lazerio koordinatas yra *PresentationForm* klasėje.

public partial class PresentationForm : Form	
Modulio vardas	Funkcinis aprašymas
▷  CollapseLeftMenu	Paslepia kairįjį meniu
▷  CollapseRightMenu	Paslepia dešinįjį meniu
▷  DoMenuAction	Atlieka pasirinkto meniu punkto veiksmą
▷  DoPaintingMove	Piešia judant kursoriui
▷  DoPaintingStart	Pradedą piešimą
▷  DoPaintingStop	Baigia piešimą
▷  DoPositionCheck	Patikrina gautos koordinatės pozicija
▷  GetGlobalPosition	Grąžina kortelės koordinates
▷  GetGlobalPosition	Grąžina paveikslėlio koordinates
▷  InitiateFenixUI	Inicijuoja klasės objektą
▷  NextSlide	Perjungia sekančią skaidrę
▷  PaintMenuItem	Ijungia/išjungia piešimo funkcija
▷  PointerMove	Inicijuoja veiksmus kursoriui pakeitus koordinates
▷  PresentationForm	Konstruktorius
▷  PreviousSlide	Perjungia praėjusią skaidrę
 _currentMenuObject	Esamasis meniu pasirinkimas
 _currentSlide	Esamoji skaidrė
 _drawingPen	Piešimo pieštukas
 _graphics	Grafinis paviršius
 _invokePoint	Esamoji koordinatė
 _lastPoint	Paskutinioji piešimo koordinatė
 _menuObjectArray	Menu objektų sąrašas
 _paintArray	Piešimo objektų sąrašas
 _paintingEnabled	Ijuntas/išjungtas piešimas
 _paintingMoveDelegate	Delegatas inicijuojantis veiksmus kursoriui pakeitus koordinates
 _paintingStartDelegate	Delegatas inicijuojantis piešimo pradžia
 _paintingStarted	Piešimas pradėtas/baigtas
 _paintingStarting	Piešimas pradedamas
 _paintingStopDelegate	Delegatas inicijuojantis piešimo pabaiga
 _paintObject	Piešimo objektas
 _pointer	Kursorius
 _positionCheckDelegate	Delegatas inicijuojantis patikrinimą gautos kursoriaus pozicijos
 _powerPoint	Power Point klasės objektas
 _settingsForm	Nustatymų formos objektas
 _slideCount	Skaidrių kiekis
 _slideFile	Skaidrės failo kelias
 _webCam	WebCamBase klasės objektas

Pointer klasės funkcijos atlieka veiksmo progreso atvaizdavimą nustatytoje lazerio pozicijoje.

public class Pointer	
Modulio vardas	Funkcinis aprašymas
▷ Dispose	Naikinti objektą
▷ DrawPointerProgress	Piešia progresą
▷ Pointer	Konstruktorius
▷ Start	Pradėti progreso piešimą
▷ Stop	Baigti priogreso piešimą
▷ TimerTick	Laikrodžio žingsnio įvykis
Location	Kursoriaus pozicija
ProgressStarted	Progresas pradėtas/baigtas
_graphicPath	Grafinių elementų sąrašas
_presentationForm	Vartotojo sąsajos klasės objektas
_panel	Progreso panelė
_started	Progresas pradėtas
_step	Progreso žingsnis
_timer	Laikrodis

2.6.4. Nustatymų posistemis








Nustatymų posistemyje yra formos klasė *SettingsForm*, kurioje yra komponentai nustatymų keitimui, video medžiagos, gautos iš internetinės kameros, atvaizdavimui ir pateikčių vartotojo sąsajos paleidimui. Taip pat klasėje yra funkcijos išsaugančios/atstatančios nustatymus į/iš konfiguracionio failo, bei atvaizduojančios aptiktą lazerio poziciją video ekrane.

public partial class SettingsForm : Form	
Modulio vardas	Funkcinis aprašymas
▷ LoadSettings	Atstato vartotojo nustatymus
▷ SaveSettings	Išsaugo vartotojo nustatymus
▷ SettingsForm	Konstruktorius
▷ VideoDraw	Atvaizduoja lazerio pozicija video ekrane
_graphics	Grafinis paviršius
_presentationForm	PresentationForm klasės objektas
_pen	Piešimo pieštukas
_powerPoint	PowerPoint klasės objektas
_webCam	WebCamBase klasės objektas





2.6.5. Grafinių vaizdų apdorojimo posistemis

Grafinių vaizdų apdorojimo posistemis susideda iš *MenuObject*, *MenuObjectArray*, *PaintBase*, *PaintObject* ir *PaintArray* klasių, kurios naudojamos vartotojo sąsajos posistemyje grafiniams vaizdams apdoroti.

MenuObject klasėje yra funkcijos, kurios virtualiai apibrėžia kiekvieną meniu juostose esantį interaktyvų objektą. *HitTest* funkcijai perdavus koordinatinių parametrą, patikrina ar ši koordinatė patenka į meniu objekto ribas. Jei nurodytos koordinatės yra objekto ribose vartotojo sąsajos, posistemis atlieka atitinkamus veiksmus.

public class MenuObject	
Modulio vardas	Funkcinis aprašymas
▷  Draw	Nupiešia meniu objektą
▷  HitTest	Patikrina ar nurodyta koordinatė patenka į meniu objekto ribas
▷  MenuObject	Konstruktorius
 Enabled	Menu objektas aktyvus/neaktyvus
 Name	Menu objekto pavadinimas
 Rectangle	Menu objekto ribos
 Type	Menu objekto tipas

Menu objektų sąrašui valdyti skirta *MenuObjectArray* klasė.

public class MenuObjectArray	
Modulio vardas	Funkcinis aprašymas
▷  Add	Įtraukia meniu objektą į sąrašą
▷  Clear	Išvalo sąrašą
▷  RemoveAt	Naikina meniu objektą
 _list	Sąrašas

PaintBase yra bazinė piešimo objektų klasė, kurioje saugojami kiekvieno objekto parametrai.

public class PaintBase	
Modulio vardas	Funkcinis aprašymas
▷ ≡ Draw	Piešti grafinį objektą
BaseGraphics	Grafinio objekto grafinis paviršius
BaseGraphicsPath	Grafinių elementų sąrašas
BasePen	Piešimo pieštukas
Enabled	Grafinis elementas aktyvus/neaktyvus

Kiekvieną grafinį elementą aprašo *PaintObject* klasė. Joje yra funkcijos skirtos piešimo objekto atvaizdavimui ir išvalymui.

public class PaintObject : PaintBase	
Modulio vardas	Funkcinis aprašymas
▷ ≡ Clear	Panaikina grafinį objektą
▷ ≡ Draw	Nupiešia grafinį objektą
▷ ≡ DrawPath	Nupiešia grafinių objektų sąrašą
▷ ≡ PaintObject	Konstruktorius

Grafinių elementų sąrašui valdyti skirta *PaintArray* klasė.

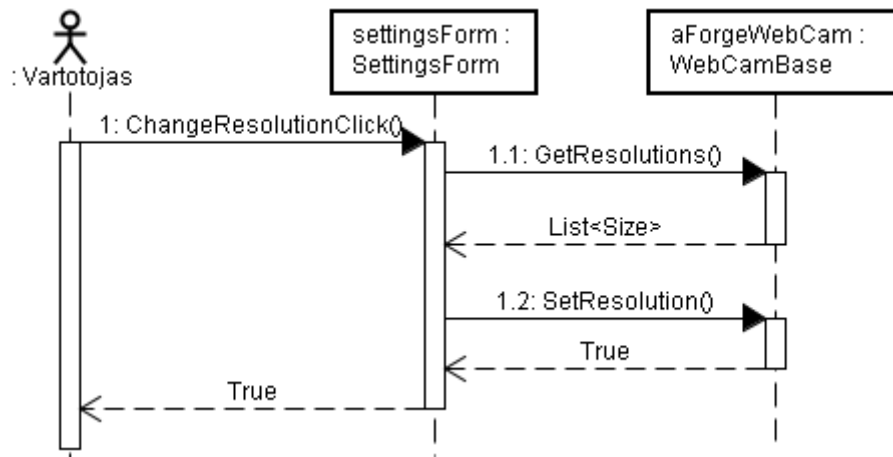
public class PaintArray	
Modulio vardas	Funkcinis aprašymas
▷ ≡ Add	Pridedamas grafinis objektas į sąrašą
▷ ≡ ClearArray	Išvalo sąrašą
▷ ≡ DrawAll	Nupiešia visus grafinius objektus
Ⓜ This	Grafinių objektų sąrašas

2.7. Sistemos elgsenos modelis

Vartotojas atlikdamas veiksmus iššaukia sistemos viduje veiksmų seką. Objektų sąveikos seką laike ištirti ir pavaizduoti naudojamos sekų diagramos. Dažniausiai, sekų diagrama vaizduoja seką objektų siunčiamų vienas kitam pranešimų, kurių siuntimo laikas ir tvarka yra griežtai apibrėžta.

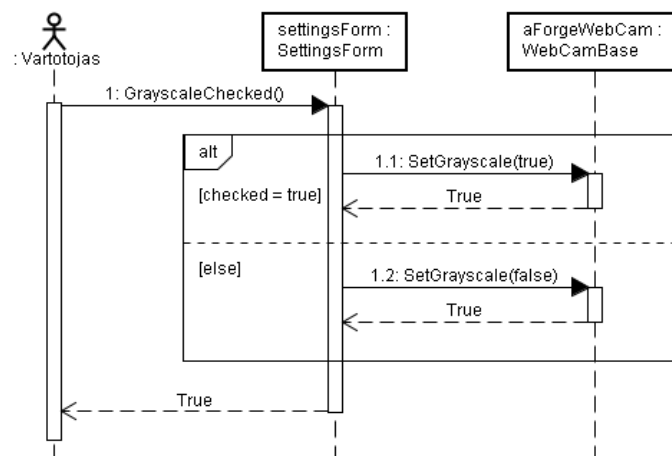
Žemiau pateikiamos sekų diagramos kiekvienam panaudos atvejui. Diagramose pavaizduota sistemos objektų sąveikos seką po vartotojo veiksmų.

„Keisti skiriamąją gebą“ panaudos atvejo sekų diagramoje (12 pav.), vartotojui iniciavus veiksmus, gaunamas skiriamųjų gebų sąrašas ir nustatoma pasirinkta geba.



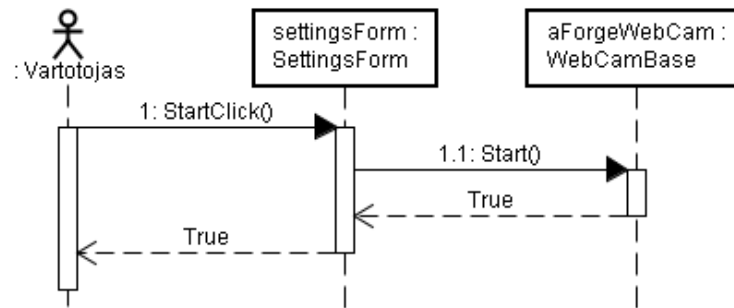
12 pav. „Keisti skiriamąją gebą“ sekų diagrama

Paveikslėlyje 13 pavaizduota „Ijungti kadro atvaizdavimą po filtrų pritaikymo“ panaudos atvejo sekų diagrama. Priklausomai nuo to ar vartotojas įjungė/išjungė nustatymą (*angl. checked*) atliekami atitinkami veiksmai.

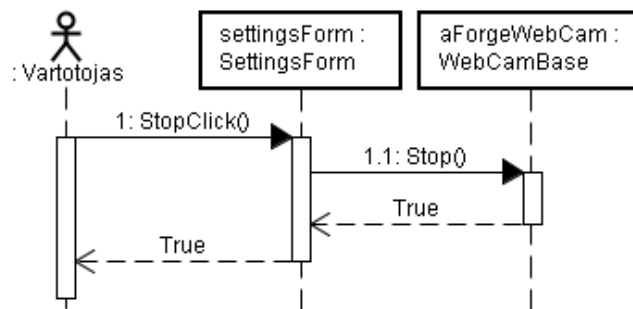


13 pav. „Ijungti kadro atvaizdavimą po filtrų pritaikymo“ sekų diagrama

Ijungti/išjungti kamerą sekų diagramose (14, 15 pav.) pavaizduotas po vartotojo veiksmo internetinės kameros įjungimas/išjungimas.

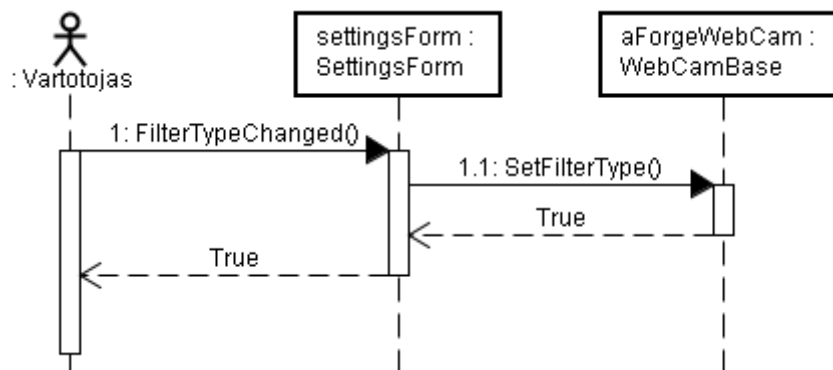


14 pav. „Ijungti kamera“ sekų diagrama



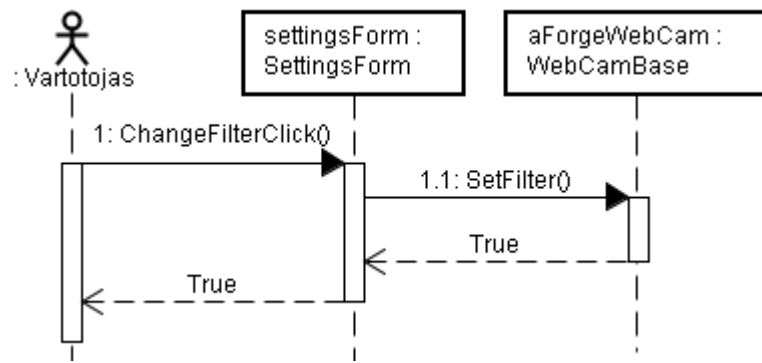
15 pav. „Išjungti kamera“ sekų diagrama

„Keisti filtro tipą“ panaudojimo atvejo sekų diagramoje pavaizduoti, filtro keitimo veiksmi.



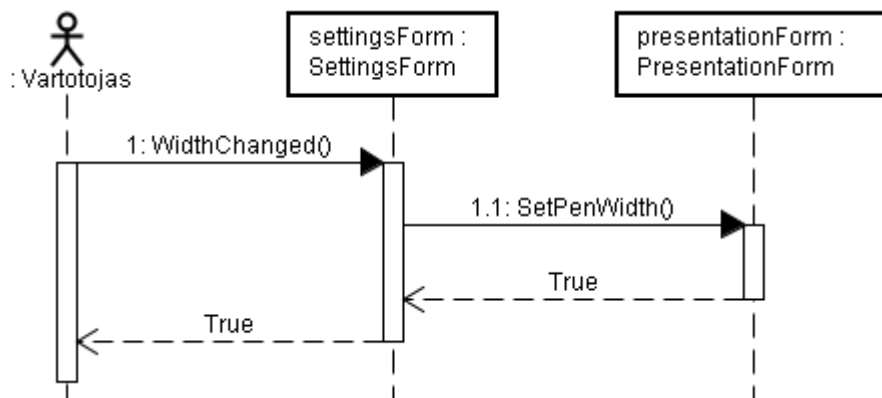
16 pav. „Keisti filtro tipą“ sekų diagrama

Keisti slenksčio, minimalaus ir maksimalaus vaizdo objekto, spalvos filtro reikšmių panaudojimo atvejų sekų diagrama yra vienoda (17 pav).



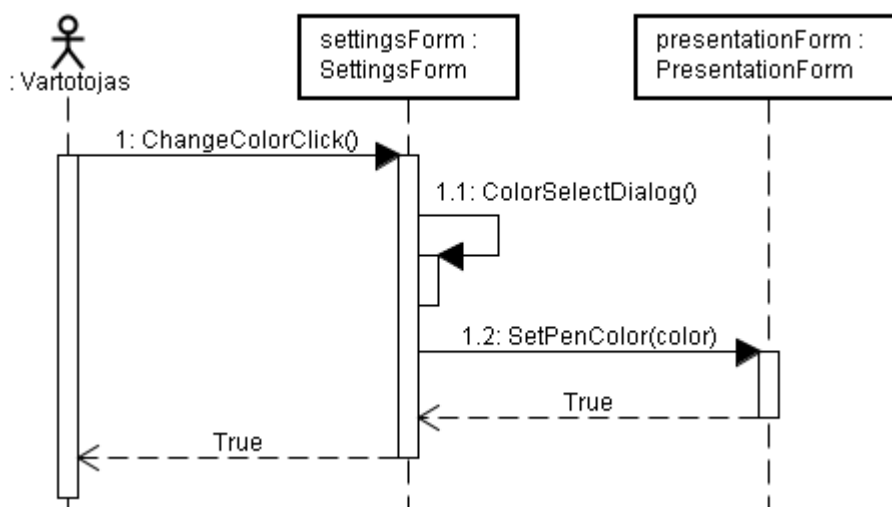
17 pav. Keisti filtro reikšmę (- es) sekų diagrama

„Keisti piešimo storį“ panaudos atvejo sekų diagramoje (18 pav.) pavaizduota, kaip yra keičiamas piešimo pieštuko storis.



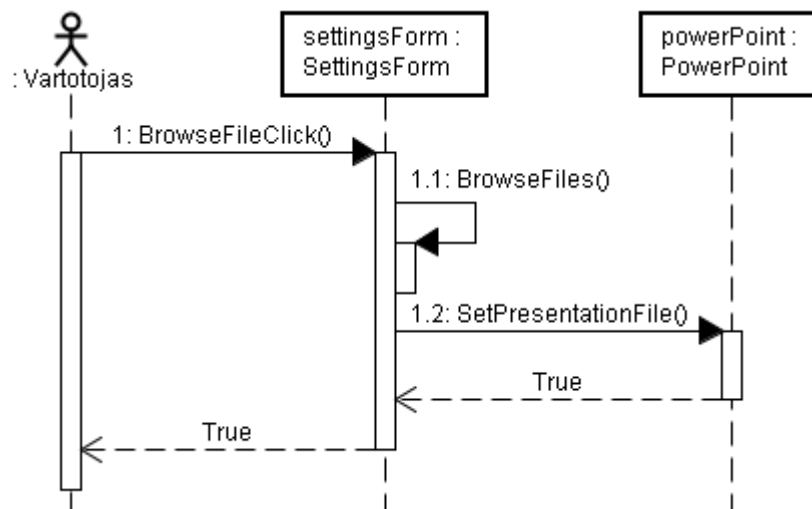
18 pav. „Keisti piešimo storį“ sekų diagrama

„Keisti piešimo spalvą“ panaudojimo atvejo sekų diagramoje (19 pav.) pavaizduotas spalvos pasirinkimo dialogo iškvietimas ir spalvos pakeitimas.



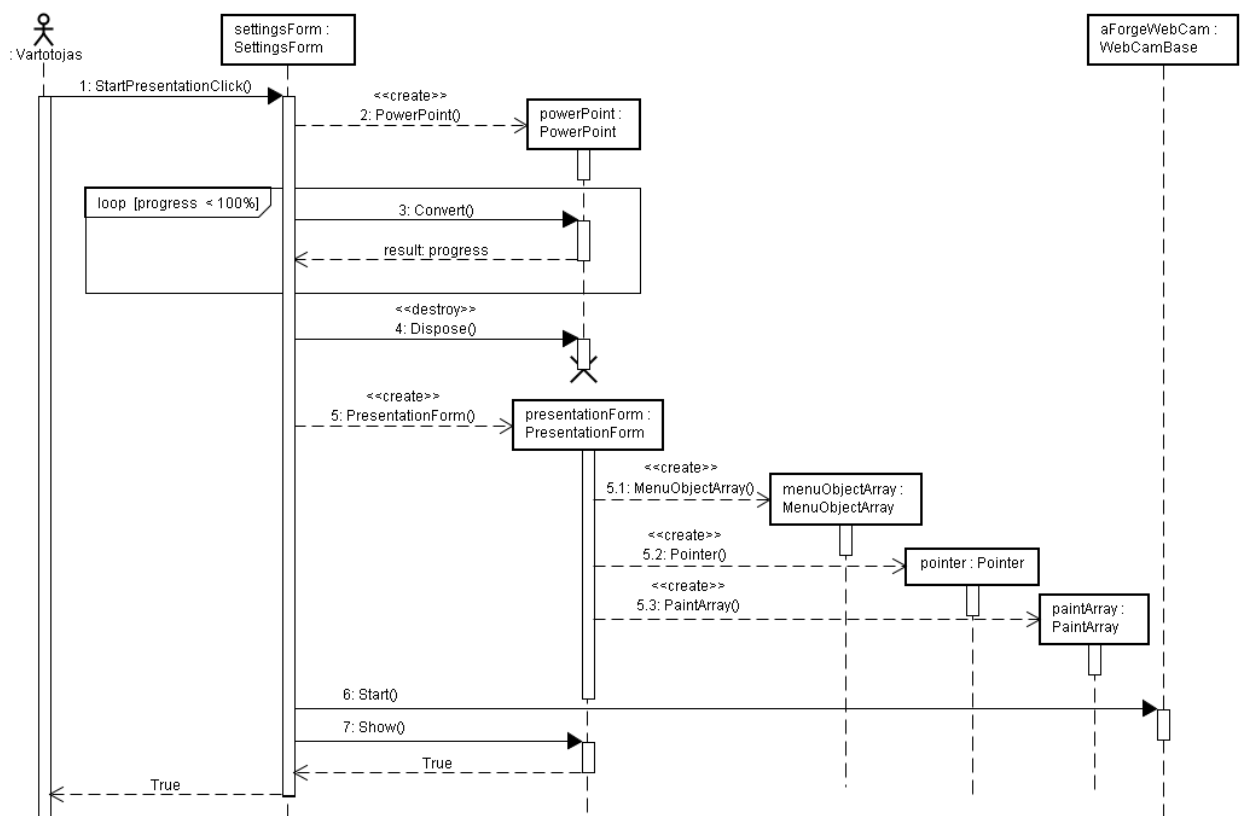
19 pav. „Keisti piešimo spalvą“ sekų diagrama

Panaudos atvejui „Pasirinkti Power Point pateikčių failą“ sekų diagrama pateikta 20 paveikslėlyje. Jame pavaizduotas failo pasirinkimo dialogo iškvietimas ir pasirinkto pateikčių failo pakeitimas.



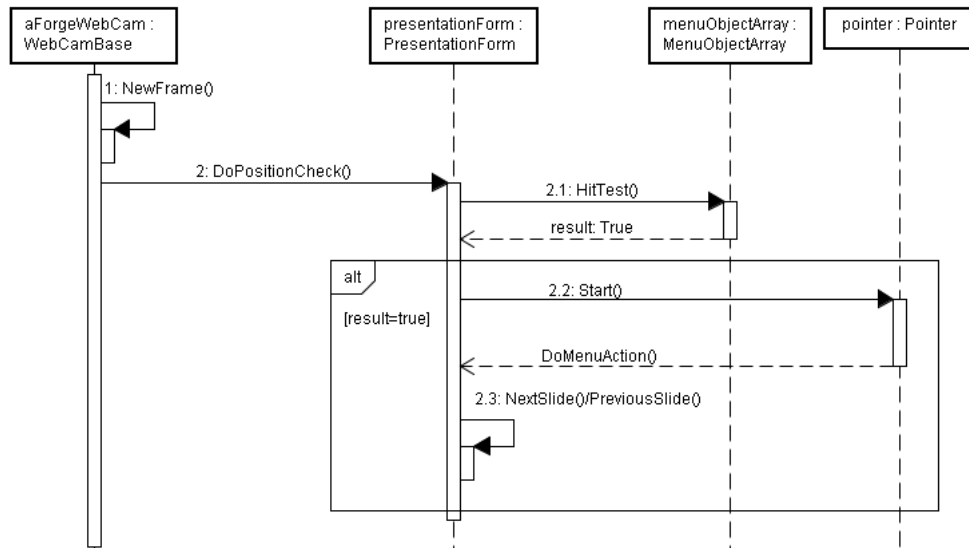
20 pav. „Pasirinkti Power Point pateikčių failą“ sekų diagrama

„Pradėti pateikti“ sekų diagramoje (21 pav.) vartotojui pasirinkus pradėti pateiktį pirmiausia konvertuojamos skaidrės į paveikslėlius. Toliau sukuriami vartotojo sąsajos ir kiti naudojami objektai bei atveriamas vartotojo sąsaja ekrane.



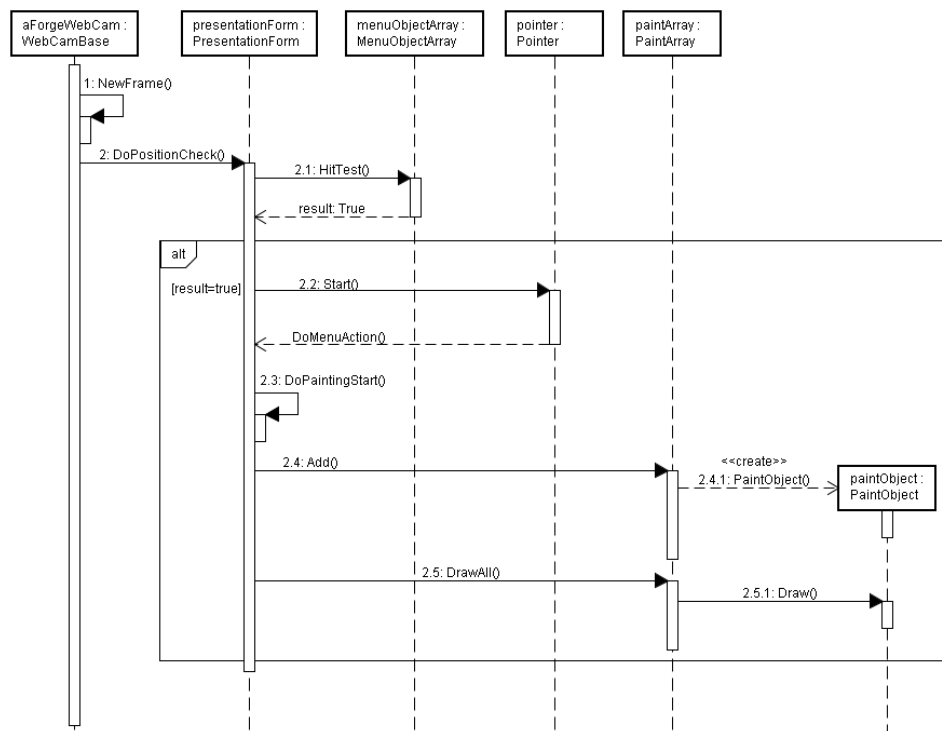
21 pav. „Pradėti pateiktį“ sekų diagrama

„Keisti skaidrę“ panaudos atvejo sekų diagramoje (22 pav.) pavaizduoti veiksmai reikalingi skaidrės pakeitimui. Gavus naują kadra išskiriamos koordinatės, toliau atliekama analizė ar šios koordinatės patenka į meniu punkto ribas. Jei rezultatas teigiamas, atvaizduojamas progresas iki veiksmo ir atliekamas skaidrės keitimo veiksmas.



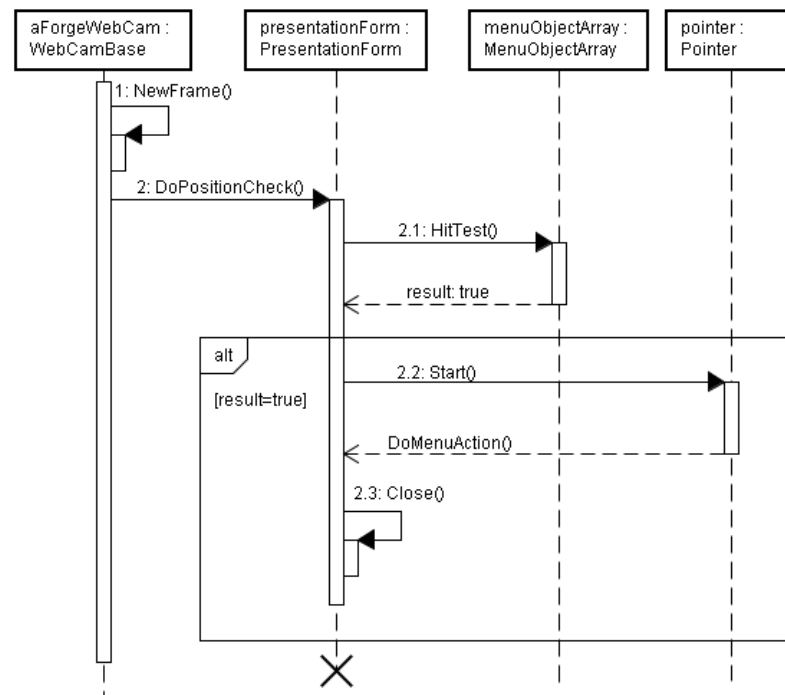
22 pav. „Keisti skaidrę“ sekų diagrama

„Piešti“ panaudos atvejo sekų diagramoje (23 pav.) matoma, kad pradžioje atliekama veiksmų seka tokia pati kaip ir 22 pav. Nustačius, kad pasirinktas piešimo meniu punktas sukuriami piešimui reikiami objektai ir iškviečiamos piešimo atvaizdavimo ekrane funkcijos.



23 pav. „Piešti“ sekų diagrama

„Baigti pateikti“ panaudojimo atvejo sekų diagramoje pavaizduota vartotojo sąsajos išjungimo (formos uždarymo) veiksmai, po to, kai buvo nustatyta jog pasirinktas pateikčių nutraukimo meniu punktas.



24 pav. „Baigti pateikti“ sekų diagrama

2.8. Išvados

- Šiame skyriuje apžvelgti suprojektuotos sistemos projektinės dokumentacijos aspektai. Dokumentavimui panaudota Argo UML programinė įranga, kurios pagalba CASE priemonėmis specifikuotos diagramos.
- Nustatyti nefunkciniai ir funkciniai sistemos reikalavimai bei specifikuota panaudojimo atvejų diagrama – taip nustatytos pagrindinės vartotojo parinkčių galimybės.
- Sistemos viena iš pagrindinių užduočių yra pirminis vaizdo apdorojimas ir lazerio žymeklio koordinatų išskyrimas. Suprojektuota sistema atlieka paminėtas pagrindines funkcijas.

3. EKSPERIMENTINIS TYRIMAS

Testavimo metu stengiamasi nustatyti, kokią įtaką sistemos darbui (tiksliau pagrindiniam sistemos uždaviniui – pirminio kadro analizei) daro skirtingi periferiniai įrenginiai, lazerinio žymeklio tipai ir skirtingi vartotojo nustatymai.

Eksperimentiniam tyrimui buvo panaudotos tokios įrangos:

- Kompiuteris: Dell Inspiron 6400 (Intel Duo Core 2GHz procesorius, 2 Gb operatyvinė atmintis, operacinė sistema Windows 7).
- Periferiniai įrenginiai:
 - Internetinės kameros: Creative Netbook Live Web Cam ir Prestigio PWC 213.
 - Projektorius Nixon.
- Lazeriniai žymekliai: raudonas (bangos ilgis 680nm, galia 1mW) ir žalias (bangos ilgis 532nm, galia 30mW).

Pagrindinis duomenų įvesties įrenginys – internetinė kamera – turi bazinius įrenginio nustatymus, tokius, kaip ryškumas, kontrastas, dažnis, išotinimas ir kiti. Testavimui visoms internetinėms kameroms buvo nustatyti šie parametrai:

- Šviesumas (*angl. brightness*) – 0%.
- Kontrastas (*angl. contrast*) – 0%.
- Gama koeficientas – 100%.
- Išotinimas (*angl. saturation*) – 100%.

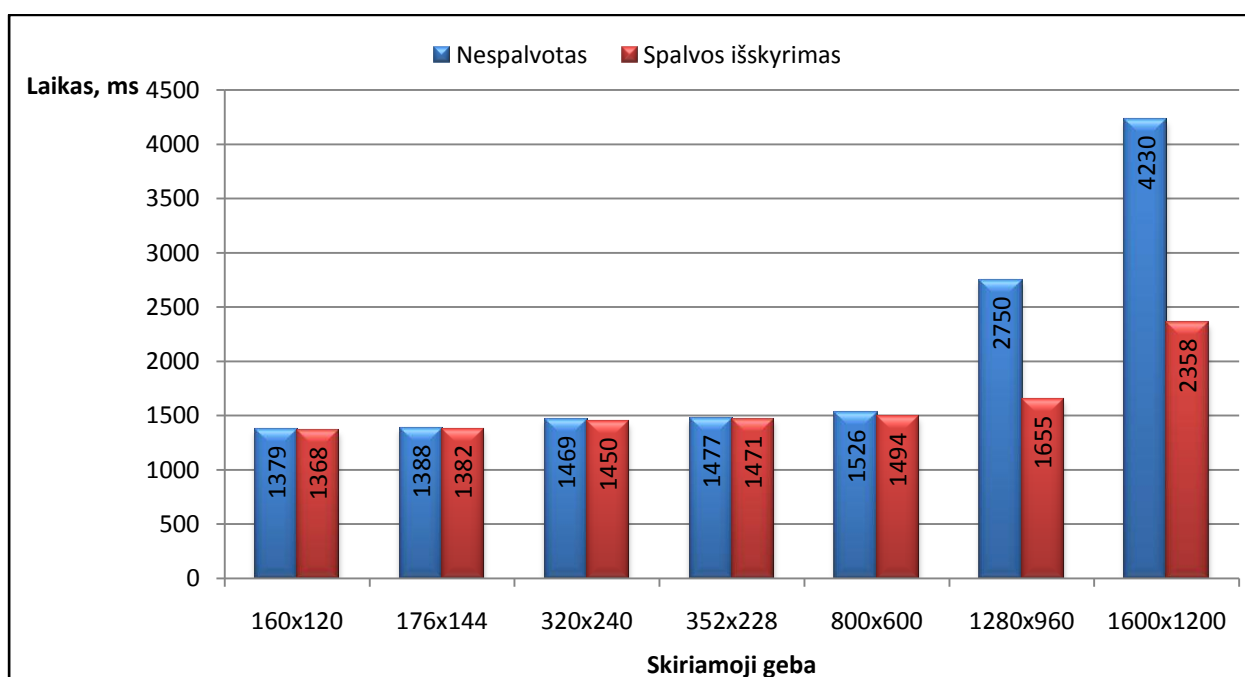
Nustačius šiuos parametrus, internetinėje kameroje fiksuojamame vaizdo kadre, nebus dalies natūralių šviesos atvaizdų. Tokiu būdu sumažinamas programinės įrangos analizės laikas, tačiau analizės laikas neapsiriboja vien tik nuo šių parametrų. Žemiau pateikiamos vaizdo analizės priklausomybės nuo vartotojo pasirinktų nustatymų.

3.1. Skiriamųjų gebų ir filtrų tipo tyrimas.

Sistemoje vartotojui galima pasirinkti dviejų tipų vaizdo filtrus ir skirtingas skiriamąsias gebas. Eksperimentiniu tyrimu buvo bandoma nustatyti, kokią įtaką daro šie nustatymai vaizdo apdorojimo greičiui. 19 lentelėje ir 11 pav. pateikti tyrimo rezultatai.

20 lentelė. Apdorojimo laiko priklausomybė nuo skiriamųjų gebų ir filtrų tipų.

Skiriamosios gebos	Apdorojimo laikas pagal filtrą, ms	
	Nespalvotas	Spalvos išskyrimas
160 x 120	1379	1368
176 x 144	1388	1382
320 x 240	1469	1450
352 x 228	1477	1471
800 x 600	1526	1494
1280 x 960	2750	1655
1600 x 1200	4230	2358



25 pav. Laiko ir skiriamosios gebos priklausomybės diagrama.

Iš eksperimento rezultatų galima daryti prielaidą, jog sistemos optimaliam veiklos greičiui (vėlinimui sumažinti) geriausia naudoti spalvos išskyrimo filtrą ir mažesnes skiriamąsias gebas: 160x120, 176x144, 320x240 arba 352x228.

Mažesnė skiriamoji geba tiesiogiai įtakoja lazerinio žymeklio pozicijos atvaizdavimą vartotojo sąsajoje. Pagal žemiau pateiktas formules apskaičiuojamos koordinatžių paklaidos:

$$\Delta X = \frac{X_v}{X_k}, \Delta Y = \frac{Y_v}{Y_k}$$

Čia:

ΔX – x koordinatės paklaida;

X_v – vartotojo sąsajos skiriamosios gebos vertikalinių taškų kiekis;

X_k – kadro skiriamosios gebos vertikalinių taškų kiekis;

ΔY – y koordinatės paklaida;

Y_v – vartotojo sąsajos skiriamosios gebos horizontalinių taškų kiekis;

Y_k – kadro skiriamosios gebos horizontalinių taškų kiekis;

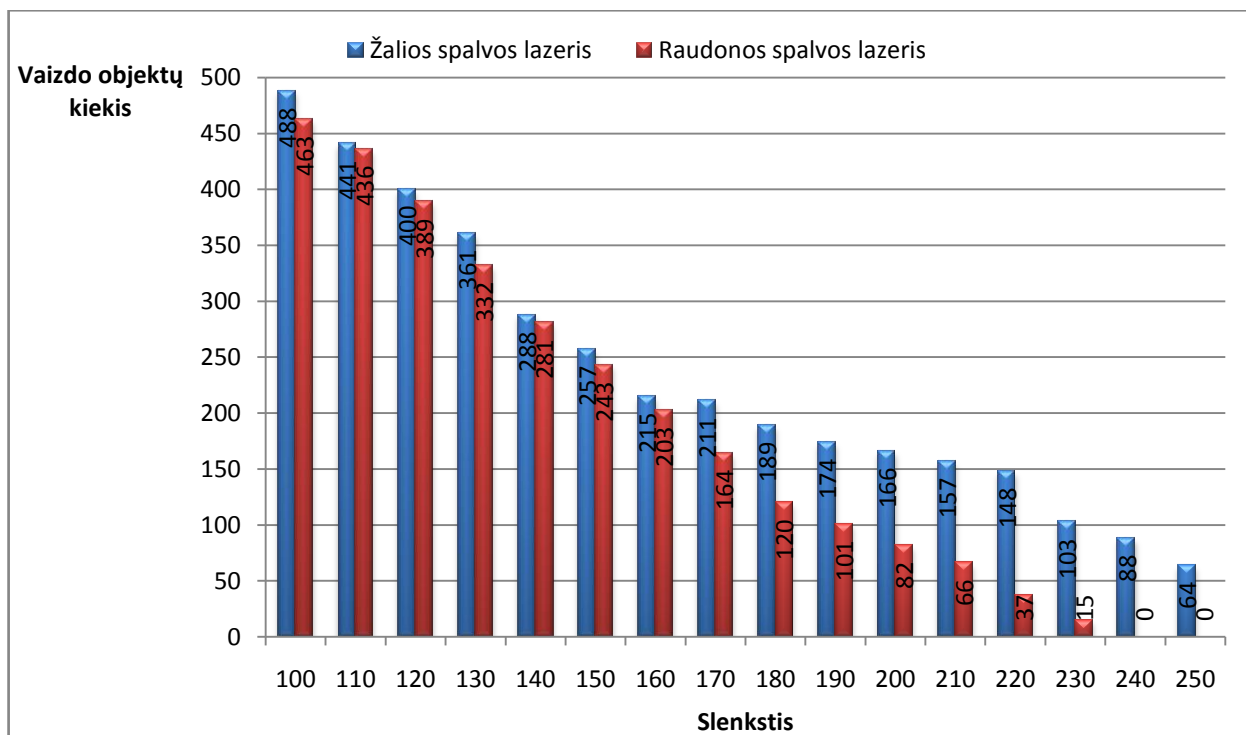
Vartotojui pasirinkus 1024x768 vartotojo sąsajos ir 160x120 kadro skiriamąsias gebas, gaunamas 6,4 X ir Y koordinatės paklaidos. Tokios paklaidos reiškia, jog pakitus aptiktoms lazerio šviesos centro koordinatėms vienu punktu, vartotojo sąsajoje atvaizduojamas lazerio taškas pasislenka 6,4 punktais. Sistemos tikslesniam ir spartesniam vaizdo apdorojimui pasiekti, reikėtų pasirinkti maksimalią (iš optimalių) skiriamąją gebą – 352x228 arba 320x240.

3.2. Slenksčio ir lazerinių žymeklių tipo tyrimas

Slenkstis – tai minimalus vaizdo objekto intensyvumas. Vartotojui nustačius mažą slenksčio ribą sistema kadre išskiria daugiau vaizdo objektų, tarp kurių yra lazerinio žymeklio šviesa. 20 lentelėje ir 12 paveikslėlio diagramoje pateiktos vaizdo objektų priklausomybės nuo pasirinktų slenksčio nustatymų bei lazerinio žymeklio tipo.

21 lentelė. Vaizdo objektų išskyrimo priklausomybė nuo slenksčio ir lazerinių žymeklių.

Slenkstis	Vaizdo objektų kiekis	
	Žalios spalvos lazeris	Raudonos spalvos lazeris
100	488	463
110	441	436
120	400	389
130	361	332
140	288	281
150	257	243
160	215	203
170	211	164
180	189	120
190	174	101
200	166	82
210	157	66
220	148	37
230	103	15
240	88	0
250	64	0



26 pav. Vaizdo objektų kiekio priklausomybės nuo slenksčio diagrama.

Priklausomai nuo naudojamo lazerinio žymeklio galingumo, pasirinktos slenksčio ribos sumažina analizuojamų vaizdo objektų kiekį kadre. Sistemoje slenksčio ribos nustatymas yra svarbus norint sumažinti paieškos spektrą, atsižvelgiant į tai, jog kadre gali būti daug nemažą intensyvumą turinčių objektų (pvz. saulės šviesos pluoštas ar atspindys).

3.3. Minimalių ir maksimalių vaizdo objektų dydžio tyrimas

Kadre aptiktų vaizdo objektų kiekio priklausomybė nuo minimalių ir maksimalių aptiktų vaizdo objektų dydžio yra tiesiogiai susijusi su atstumu tarp internetinės kameros ir ekrano bei lazerio tipo. Pasirinkus mažesnę atstumą ir galingesnę lazerį, kadre gaunamas didesnis lazerio šviesos vaizdo objektas. Nustačius nedideles analizuojamų objektų dydžio ribas, platus lazerio taškas bus ignoruojamas.

Priklausomybei nuo minimalaus ir maksimalaus objekto dydžio nustatymų analizei buvo pasirinktas 7 metrų atstumas tarp internetinės kameros ir naudojami skirtingo galingumo lazeriai. Siaurinant objekto dydžio ribos reikšmes, stebimas surastų objektų kiekis kadre. Žemiau pateiktoje lentelėje ir diagramoje pavaizduoti tyrimo rezultatai.

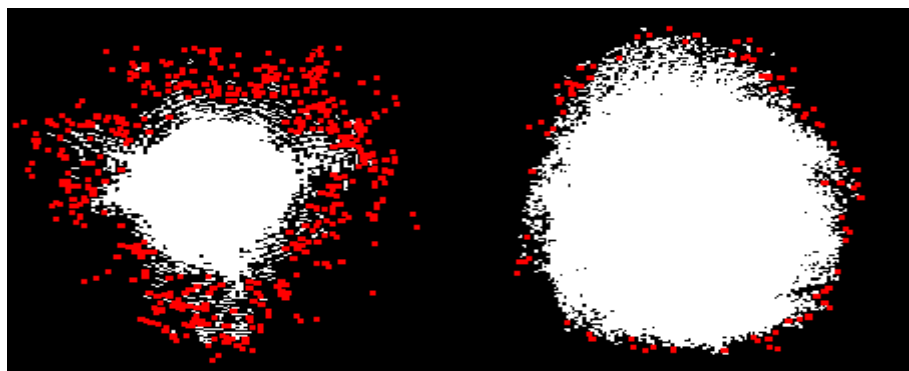
22 lentelė. Vaizdo objektų išskyrimo priklausomybė nuo minimalių ir maksimalių vaizdo objektų dydžio

Objekto ribos(minimali x maksimali)	Vaizdo objektų kiekis	
	Žalios spalvos lazeris	Raudonos spalvos lazeris
1 x 140	265	249
10 x 130	237	223
20 x 120	191	176
30 x 110	166	157
40 x 100	138	121
50 x 90	67	48
60 x 80	25	2
70 x 70	0	0



27 pav. Vaizdo objektų kiekio priklausomybės nuo ribų diagrama.

Sistemoje naudojant didesnės galios lazerinį žymeklį (šiuo atveju žalia), kadre gaunamas ryškus lazerio taškas ir aplink jį susidaro daug įvairaus dydžio ryškių vaizdo objektų (28 pav.). Šie objektai ir lemia didesnę aptiktų objektų kiekį rezultatuose, kai naudojamas žalias lazeris.



28 pav. Vaizdo objektai aplink žalią (kairėje) ir raudoną lazerį.

Iš tyrimo rezultatų pastebima, jog siaurinant dydžių ribas sumažėja vaizdo objektų kiekis. Žinant apytiksles lazerinio žymeklio kadre dydžio ribas, galima susiaurinti analizuojamų vaizdo objektų spektrą. Todėl vaizdo objektų dydžių ribos nustatymai yra svarbūs tikslesniam sistemos darbui.

3.4. Išvados

- Eksperimentinio tyrimo metu nustatyta, jog pirminio kadro apdorojimo laikas yra priklausomas nuo vartotojo pasirinktų skiriamųjų gebų ir kadro filtravimo tipo. Optimaliam sistemos veikimui patartina naudoti 352x228 arba 320x240 skiriamąsias gebas ir spalvos išskyrimo filtrą.
- Atlikus vaizdo objektų išskyrimo kiekio priklausomybės nuo slenksčio ribos ir lazerinio žymeklio tipo analizę, nustatyta, jog slenksčio ribos nustatymas sistemoje yra svarbus tikslesnei lazerio paieškai kadre. Atsižvelgiant į aplinkos apšvietumą, pasirinktą slenksčio ribą ir lazerinio žymeklio tipą, galima susiaurinti (tarp kitų intensyvią šviesą skleidžiančių objektų) ieškomo lazerio šviesos pluošto paiešką.
- Norint padidinti sistemoje analizuojamo vaizdo kadro tikslumą, galima naudoti minimalaus ir maksimalaus vaizdo objektų dydžio nustatymus. Šių parametru dėka sumažinamas analizuojamų vaizdo objektų kiekis.

IŠVADOS

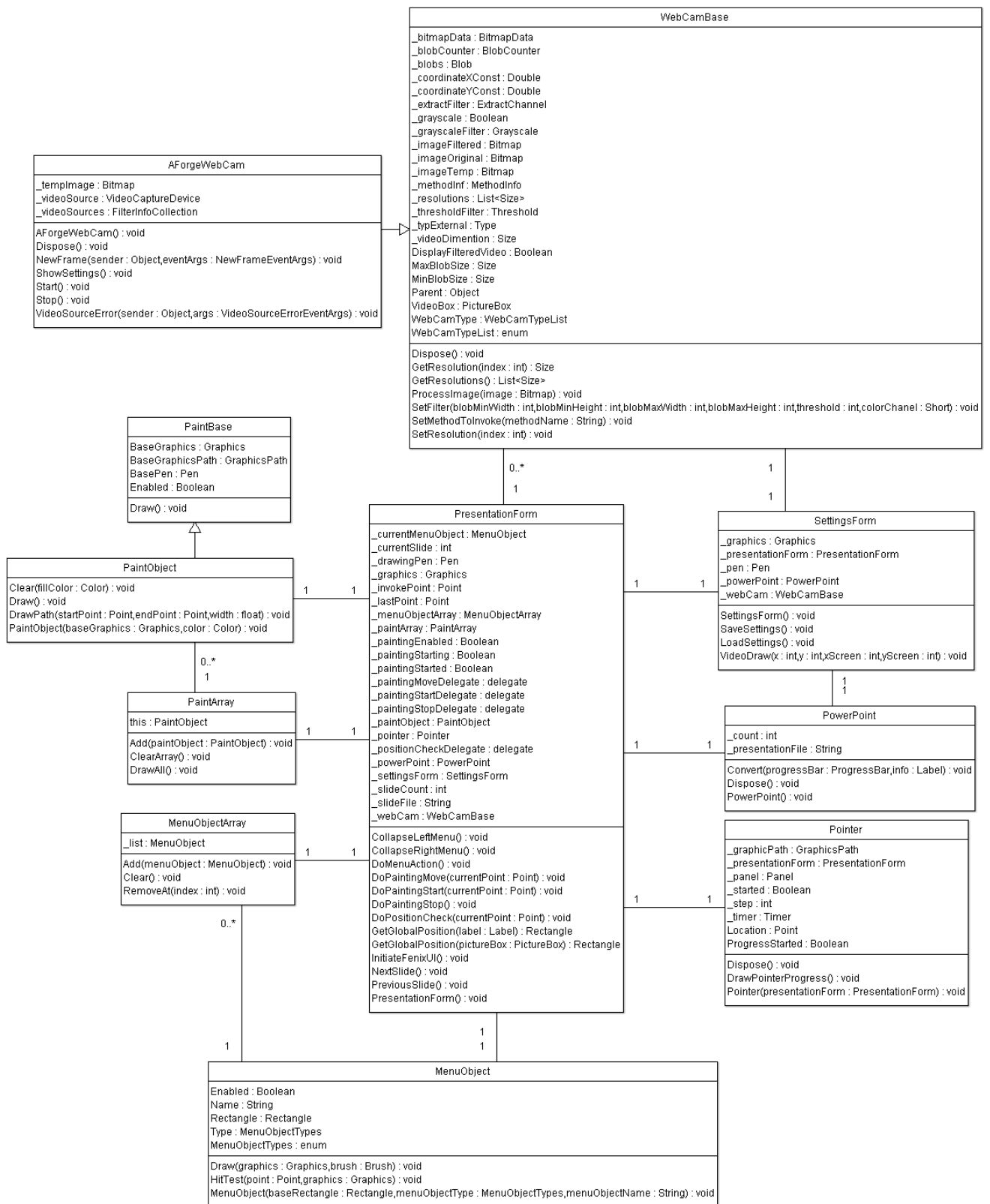
- Šiame darbe apžvelgtos panašios sistemos ir nustatyta, jog vienas iš pagrindinių darbo užduočių yra pirminis vaizdo medžiagos apdorojimas sistemoje. Pirminio vaizdo apdorojimo rezultatas yra lazerio žymeklio pozicijos identifikavimas. Trumpai aptarti Carsten Kirstein ir Heinrich Müller bei Jean-Francois Lapointe ir Guy Godin sistemų modeliai ir veikimo principai.
- Aprašytos technologijos t.y. GDI ir GDI+, DirecX, AForge.NET Framework, programavimo kalbos ir IDE, kuriomis galima realizuoti sistemą. Technologijų tinkamumą sistemos realizavimui lėmė, tai, jog sistemoje nėra greitai besikeičiančios grafinė aplinkos ir sistema pritaikyta Windows operacinei sistemai. Visual Studio .NET, C#, GDI+ ir AForge .NET Framework pasirinktos kaip tinkamiausio technologijos realizacijai.
- Surinkus sistemai keliamus reikalavimus išsiaiškinta, jog vienas iš pagrindinių reikalavimų yra vėlinimas. Realizuotos sistemos vėlinimui įvertinti atliktas vaizdo apdorojimo laiko priklausomybės nuo skiriamosios gebos tyrimas.
- Naudojant CASE priemones, suprojektuota sistema. Projektuojant išsiginčinta į sistemos posistemų architektūrą, posistemius sudarančias klases ir objektų sąveikos sekas.
- Realizuojant sistemą, sužinota apie įvairius apdorojimo ir ryškiausio taško paieškos metodus pirminiame vaizdo kadre, apdorojimui naudojant .NET Framework ir C# programavimo kalbą. Įgyta nauja programavimo patirtis ir atlikta analizė, kokią įtaką programos veikimui daro sistemos įrangos nustatymai bei išoriniai veiksniai.
- Eksperimentinio tyrimo metu atlikti testavimai, kurių metu nustatyta, jog sistemos optimaliam veikimui (mažiausiam vėlinimui ir didžiausiam tikslumui) pasiekti geriausia naudoti 320x240 arba 352x228 internetinės kameros skiriamąsias gebas. Taip pat atlikti tyrimai parodė, kad vartotojas, sistemoje keisdamas filtrų nustatymus, gali išvengti netikslaus programos veikimo, kurį sukelia aplinkos (saulės šviesa, atspindžiai ir kiti) veiksniai.

LITERATŪRA

1. UPoint – the Ultimate Pointer. [interaktyvus]. [Žiūrėta 2010 02 25]. Prieiga per internetą: <<http://www.ultimatepointer.com/Upoint.htm> >
2. Graffiti Research Lab. Laset tag. [interaktyvus]. [Žiūrėta 2010 02 25]. Prieiga per internetą: <<http://graffitiresearchlab.com/projects/laser-tag/> >
3. Anders Marklun. Building a mobile robot with optical tracking and SLAM. Magistro darbas. Lulea University of Technology, 2009.
4. Carsten Kirstein ir Heinrich Müller. On-Screen Laser Spot Detection for Large Display Interaction. Multimedia Modeling konferencijos pranešimo medžiaga. 1998, p. 191-192.
5. Jean-Francois Lapointe ir Guy Godin. On-Screen Laser Spot Detection for Large Display Interaction. HAVE konferencijos pranešimo medžiaga. 2005, p. 5.
6. Binod Kumar. DirectX Explained. [interaktyvus]. 2009 liepa. [Žiūrėta 2010 02 26]. Prieiga per internetą: <<http://www.viaarena.com/directx-explained.aspx?ID=167&MCatID=1> >
7. K. Hawkins; D. Astle. Open Game Programming. Roseville, California, 2001. 15 p.
8. Victor A. Debelov, Yuri A. Tkachov. One approach to C++ look at DirectDraw and Direct3D. Graphicon konferencijos pranešimo medžiaga. Maskva, 1999, p. 205-211.
9. Andrew Lewycky ir Gavriel State. DirectX and Wine. [interaktyvus]. 2001 liepa. [Žiūrėta 2010 02 26]. Prieiga per internetą: <<http://www.linuxsymposium.org/archives/OLS/Reprints-2001/state.pdf> >
10. Windows GDI+. [interaktyvus]. [Žiūrėta 2010 02 23]. Prieiga per internetą: <<http://msdn.microsoft.com/en-us/library/ms533798%28VS.85%29.aspx> >
11. .NET Framework library. [interaktyvus]. [Žiūrėta 2010 02 23]. Prieiga per internetą: <<http://msdn.microsoft.com/en-us/library/system.drawing.aspx> >
12. AForge.NET Framework. [interaktyvus]. [Žiūrėta 2010 01 10]. Prieiga per internetą: <<http://code.google.com/p/aforge/> >
13. Programavimo kalba: Java. [interaktyvus]. [Žiūrėta 2010 02 23]. Prieiga per internetą: <<http://www.phpfusion-lt.com/programavimo-kalba-java,s200> >
14. Object Technology International, Inc. Eclipse Platform Technical Overview. [interaktyvus]. 2003. [Žiūrėta 2010 02 23]. Prieiga per internetą: <www.eclipse.org/whitepapers/eclipse-overview.pdf >
15. Ben Albahari. A Comparative overview of C#. [interaktyvus]. 2000 liepa. [Žiūrėta 2010 02 23]. Prieiga per internetą: <http://genamics.com/developer/csharp_comparative.htm >
16. ArgoUML. Preface. [interaktyvus]. [Žiūrėta 2010 03 15]. Prieiga per internetą: <<http://argouml-stats.tigris.org/documentation/manual-0.30/pr01.html> >

PRIEDAI

1. PRIEDAS. Detalizuota klasių diagrama.



2. PRIEDAS. CD.