



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS**

Simonas Naulickas

**AUTOMOBILIO KĖBULO GEOMETRIJOS PJAUSTYMO
PROGRAMINĖS ĮRANGOS KŪRIMAS IR GEOMETRIJOS
PJAUSTYMO ALGORITMŲ TYRIMAS**

Magistro projektas

Vadovas

Lekt. dr. Šarūnas Packevičius

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PROGRAMŲ INŽINERIJOS KATEDRA

TVIRTINU

Katedros vedėjas

(parašas) Doc. dr. Tomas Blažauskas

(data)

AUTOMOBILIO KĖBULO GEOMETRIJOS PJAUSTYMO
PROGRAMINĖS ĮRANGOS KŪRIMAS IR GEOMETRIJOS
PJAUSTYMO ALGORITMŲ TYRIMAS

Magistro projektas

Programų sistemų inžinerija (kodas 621E16001)

Vadovas

(parašas) Lekt. dr. Šarūnas Packevičius

(data)

Recenzentas

(parašas) Doc. dr. Rytis Maskeliūnas

(data)

Projektą atliko

(parašas) Simonas Naulickas

(data)



KAUNO TECHNOLOGIJOS UNIVERSITETAS
Informatikos Fakultetas

(Fakultetas)

Simonas Naulickas

(Studento vardas, pavardė)

Programų sistemų inžinerija, 621E16001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Pavadinimas“
AKADEMINIO SAŽINGUMO DEKLARACIJA

20 ____ m. ____ d.
Kaunas

Patvirtinu, kad mano **Simono Naulicko** baigiamasis projektas tema „Automobilio kėbulo geometrijos pjaustymo programinės įrangos kūrimas ir geometrijos pjaustymo algoritmų tyrimas“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

TURINYS

Lentelių sąrašas	6
Paveikslų sąrašas	7
1. Įžanga	9
1.1. Dokumento paskirtis.....	9
1.2. Santrauka	9
2. Analitinė dalis	11
2.1. Tikslas	11
2.2. Sistemos pagrindumas	11
2.3. Egzistuojantys sprendimai.....	11
2.3.1. Graphisoft Archicad.....	11
2.3.2. Autodesk Revit.....	11
2.3.3. Autodesk Alias	12
2.4. Egzistuojančios technologijos	12
2.4.1. Modelio pjaustymas	12
2.4.2. Pasirinktas geometrijos pjaustymo algoritmas.....	13
2.5. Užsakovai, pirkėjai ir kiti sistema suinteresuoti asmenys.....	13
2.5.1. Užsakovas	13
2.5.2. Pirkėjas.....	14
2.5.3. Vadovas.....	14
2.6. Vartotojai.....	14
2.7. Projekto apribojimai	14
2.7.1. Įpareigojantys apribojimai	14
2.8. Svarbūs faktai ir prielaidos.....	15
2.8.1. Svarbūs faktai.....	15
2.8.2. Prielaidos.....	15
3. Projektinė dalis.....	16
3.1. Reikalavimų specifikacija	16
3.1.1. Sistemos funkcijos	16
3.1.2. Panaudojimo atvejų sąrašas	17
3.2. Funkciniai reikalavimai ir reikalavimai duomenims.....	18
3.2.1. Funkciniai reikalavimai.....	18
3.2.2. Reikalavimai duomenims.....	21
3.3. Nefunkciniai reikalavimai	22
3.3.1. Reikalavimai sistemos išvaizdai	22
3.3.2. Reikalavimai panaudojamumui.....	22
3.3.3. Reikalavimai vykdymo charakteristikoms.....	22
3.3.4. Reikalavimai sistemos priežiūrai	23

3.3.5.	Reikalavimai saugumui.....	23
3.3.6.	Kultūriniai-politinai reikalavimai	23
3.4.	Statinis sistemos vaizdas	24
3.4.1.	Sistemos paketai.....	24
3.4.2.	Klasių diagramos.....	24
3.5.	Dinaminis sistemos vaizdas.....	26
3.5.1.	Sekos diagramos	26
3.5.2.	Būsenų diagramos	28
3.5.3.	Veiklos diagramos.....	31
4.	Tyrimo dalis	34
4.1.	Tikslas	34
4.2.	Sukurtos sistemos pjovimo algoritmo tyrimas	34
4.3.	Tyrimui naudojamų įrangos aprašas	34
4.4.	Nuoseklaus pjovimo algoritmo tyrimo rezultatai	35
5.	Eksperimentinė dalis	36
5.1.	Tikslas	36
5.2.	Eksperimentų aprašymas.....	36
5.3.	Eksperimentų rezultatai.....	37
5.3.1.	Nuoseklaus algoritmo palyginimas su Java 8 lygiagrečiu algoritmu.....	37
5.3.2.	Nuoseklaus algoritmo palyginimas su OpenCL lygiagrečiu algoritmu.....	38
6.	Išvados.....	39
7.	Literatūra	40
8.	Terminų ir santrumpų žodynas.....	41

LENTELIŲ SĄRAŠAS

Lentelė 1 Pjaustymo algoritmų palyginimas.....	13
Lentelė 2 Automobilių trimačių modelių modeliotojas	14

PAVEIKSLŲ SĄRAŠAS

1 pav. Sistemos panaudojimo atvejų diagrama	16
2 pav. Duomenų modelio diagrama	21
3 pav. Sistemos paketai	24
4 pav. Objektų užkrovimo ir išsaugojimo paketo klasių diagrama	24
5 pav. Atvaizdavimo paketo klasių diagrama	25
6 pav. Geometrijos pjaustymo paketo klasių diagrama	25
7 pav. eMMA sąsajos paketo klasių diagrama	25
8 pav. Pjaunančiosios plokštumos nurodymo sekų diagrama	26
9 pav. Pjaunančiosios plokštumos keitimo sekų diagrama	26
10 pav. Pjovimo sekų diagrama	27
11 pav. Nupjautos dalies pasirinkimo sekų diagrama	27
12 pav. Modelio užkrovimo sekų diagrama	28
13 pav. Pjaunančiosios plokštumos nurodymo būsenų diagrama	28
14 pav. Pjaunančiosios plokštumos keitimo būsenų diagrama	29
15 pav. Pjovimo būsenų diagrama	29
16 pav. Nupjautos dalies pasirinkimo būsenų diagrama	30
17 pav. Modelio užkrovimo būsenų diagrama	30
18 pav. Pjaunančiosios plokštumos nurodymo veiklos diagrama	31
19 pav. Pjaunančiosios plokštumos keitimo veiklos diagrama	31
20 pav. Pjovimo veiklos diagrama	32
21 pav. Nupjautos dalies pasirinkimo veiklos diagrama	32
22 pav. Modelio užkrovimo veiklos diagrama	33
23 pav. Nuoseklus algoritmo greitaveika	35
24 pav. Sudėtingiausias naudotas modelis	36
25 pav. Modelis po pjovimo	37
26 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (Java8 Parralel Stream)	37
27 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (OpenCL)	38

SUMMARY

Vehicle modelling is a process that is difficult and requires a lot of time. Therefore, software that makes inspecting the models less time consuming are very helpful. With the aid of such programs it is possible to find any discrepancies faster than using regular programs. Also, such software can help present the 3D model to customers and interested parties.

One possible way to make the inspection of 3D models faster is to cut it into smaller pieces that can later be viewed separately. Because the vehicle models are usually complex inspecting different cuts of a model allows the modeller to view the model from different angles.

The purpose of this work is to analyze existing 3D geometry cutting algorithms, examine the chosen algorithm's performance and implement improvements for the algorithm according to the algorithm's performance research results.

Keywords: 3D mesh cutting, 3D mesh segmentation, 3D mesh scissoring, plane triangle intersection, 3D mesh and plane intersection curve

1. ĮŽANGA

Šiuolaikinės technologijos atvėrė kelią trimačių kompiuterinių modelių panaudojimui ne tik filmų, žaidimų, reklamavimo, architektūros pramonei, bet ir tokioms pramonės šakoms kaip automobilių gamyba. Pasinaudodami įvairiais trimačio modeliavimo įrankiais, automobilių gamintojai gali tiksliai sumodeliuoti norimus automobilius bei paprastai atlikti pakeitimus jau sukurtiems modeliams. Nepaisant to, automobilių modelių kokybę vis tiek reikia įvertinti, o klaidos modelyje gali daryti įtakos automobilių kūrimo procesui. Dėl to, įrankiai padedantys įvertinti automobilio modelio kokybę, yra naudingi automobilių gamintojams. Vienas būdas, galintis palengvinti automobilio modelio kokybės įvertinimą – modelio padalinimas į kelias dalis naudojantis pjaunančiąja plokštuma.

Kadangi automobilių pramonėje naudojami modeliai turi tiksliai atitikti gaminamą automobilį, todėl jų geometrija yra gana sudėtinga. Programa, kuri atlieka tokių sudėtingų modelių peržiūra bei gali padalinti tokių modelių į kelias dalis, turi naudoti pakankamai greitą algoritmą, kad vartotojas nepajustų sulėtėjimo modelio sudalinimo metu.

Šio darbo tikslai:

- Išanalizuoti egzistuojančius sudėtingos geometrijos pjovimo algoritmus
- Iširti realizuotoje sistemoje naudojamo pjovimo algoritmo greitaveiką
- Atsižvelgus į tyrimo rezultatus, realizuoti algoritmo patobulinimus

1.1. Dokumento paskirtis

Šio dokumento paskirtis yra parodyti magistrinio darbo „Automobilio kėbulo geometrijos pjaustymo programinės įrangos kūrimas ir geometrijos pjaustymo algoritmų tyrimas“ rezultatus.

Šį dokumentą sudaro:

- Analitinė dalis – šioje dalyje aprašoma:
 - Kodėl sistema kuriama
 - Kokios panašios sistemos egzistuoja
 - Kokios egzistuoja technologijos sistemos realizavimui
 - Kokias technologijas pasirinkta naudoti
- Projektinė dalis – šioje dalyje pateikiama sukurtos sistemos techninė-projektinė dokumentacija:
 - Sistemos funkcijos
 - Architektūra
- Tyrimo dalis – šioje dalyje pateikiama sukurtos sistemos kokybės analizė bei aprašomos jos tobulinimo galimybės
- Eksperimentinė dalis – šioje dalyje pateikiami sukurtos programinės įrangos bei jos patobulinimų eksperimentinis tyrimas

1.2. Santrauka

Automobilio modeliavimas yra sudėtingas ir daug laiko reikalaujantis procesas. Dėl to, įvairūs įrankiai, paspartinantys modelio peržiūrą, yra ypač vertingi. Jų pagalba galima surasti neatitikimus greičiau nei įprastomis priemonėmis. Be to, šios priemonės taip pat naudingos sukurtu automobilio modelio pristatymui užsakovams ar suinteresuotiems asmenims.

Vienas iš būdų pagreitinti modelio peržiūrą – trimatės modelio supjaustymas į kelias dalis, kurias vėliau galima peržiūrėti atskirai. Kadangi automobilių modeliai būna gana sudėtingi, todėl jų peržiūrėjimas įvairiais pjūviais ar atskirų dalių peržiūrėjimas modeliuotojui leidžia pažvelgti į modelį iš įvairių pusių.

Šio darbo tikslas yra ištirti egzistuojančius trimatės geometrijos pjaustymo algoritmus, ištirti pasirinkto algoritmo greitaveiką ir, atsižvelgus į tyrimo rezultatus, realizuoti algoritmo patobulinimus.

2. ANALITINĖ DALIS

2.1. Tikslas

Šio darbo tikslas yra sukurti plėtinį programai „eMMA Analyst“, kuris leistų vartotojui nurodyti pjaunančiąją plokštumą, kuri padalintų trimatį modelį į dvi dalis ir pažymėtų susikirtimo kreivę. Vartotojas galės paslėpti kurią nors nupjautą dalį, kad būtų lengviau peržiūrėti kitą.

2.2. Sistemos pagrindumas

Tam, kad būtų užtikrinta sumodeliuoto automobilio kėbulo geometrijos kokybė, būtina jo modelį peržiūrėti iš visų pusių, taip pat ir skerspjūvius. Įprastos programos tokio funkcionaluma nesuteikia ir leidžia paslėpti tiksliai modelio dalis, kas reikalauja daugiau laiko. Galimybė padalinti sudėtingo automobilio modelį į dvi dalis per kelias akimirkas būtų labai naudinga jo peržiūrai. Be to, programa, gebanti surasti trimačio modelio ir plokštumas susikirtimo taškus, galėtų būti panaudojama automobilių trimačiam spausdinimui.

Sistema yra kuriama pagal užsakymą ir būtų integruojama į automobilių gaminimo rinkoje naudojamą programinę įrangą. Dėl to, kokybiško sistemos realizavimo atveju, ji bus naudojama realiomis gamybos sąlygomis ir turės išliekamąją vertę.

Beje, tokios sistemos bus reikalingos tol, kol nėra įmanoma automatizuoti automobilių modeliavimo bei modelio patikrinimo proceso.

2.3. Egzistuojantys sprendimai

Rinkoje egzistuoja keletas programų su modelio pjaustymo galimybėmis. Dauguma jų skirtos architektūriniam modeliavimui arba transporto priemonių modeliavimui.

2.3.1. Graphisoft Archicad

Tai architektūrinio modeliavimo programa pagaminta Vengrijos kompanijos „Graphisoft“. „Archicad“, yra pilnas 2D ir 3D projektavimo, vizualizavimo ir kitų funkcijų, skirtų architektams, projektuotojams ir planuotojams, rinkinys [1]. Šiame rinkinyje yra integruotas didelis programinės įrangos, skirtos architektūriniam projektavimui, pasirinkimas [2]:

- 2D kompiuterizuoto projektavimo programinė įranga
- 3D modeliavimo programinė įranga
- Architektūrinio atvaizdavimo ir vizualizavimo programinė įranga
- Dokumentų valdymo įrankis
- Pastatų informacijos modeliavimo programinė įranga

Šioje programiniame rinkinyje taip pat yra realizuotas interaktyvus trimačio modelio pjaustymas naudojantis pjaunančiąją plokštumą.

2.3.2. Autodesk Revit

„Autodesk Revit“ yra pastatų informacijos modeliavimo programinė įranga skirta architektams, statybų inžinieriams, inžinieriams, projektuotojams ir rangovams. Ji suteikia vartotojams galimybę suprojektuoti pastatą ir jo komponentus trimatėje erdvėje, anotuoti modelį naudojantis 2D projektavimo elementais ir pasiekti pastato informaciją iš pastatų modelių duomenų bazės [3].

Šioje programoje taip pat yra realizuotas modelio pjovimas naudojantis pjaunančiąją plokštumą.

2.3.3. Autodesk Alias

“Autodesk Alias” yra kompiuterizuoto pramoninio projektavimo produktų šeima. Eskizų piešimo, modeliavimo ir vizualizavimo įrankiai yra sujungti į vieną programą [4]. Ji yra dažniausiai naudojama automobilių, lėktuvų, laivų, sportinės įrangos, žaislų ir aksesuarų projektavime.

Be kitų funkcijų, šis produktas taip pat turi galimybę padalinti trimatį modelį panaudojant pjaunančiąją plokštumą.

2.4. Egzistuojančios technologijos

2.4.1. Modelio pjaustymas

Yra keletas trimačio modelio pjaustymo algoritmų, kuriuos būtų galima panaudoti pačius arba panaudoti juose naudojamas idėjas:

2.4.1.1. Interactive Mesh Cutting Using Constrained Random Walks

Tai algoritmas, sprendžiantis pjaunančiojo kontūro, naudojamo komponentų išgavimui iš trimačio modelio, suradimo problemą [5]. Šis algoritmas naudoja vartotojo įvestį, kad geriau nustatytų, kurią dalį vartotojas nori išgauti iš modelio. Yra naudojamos trys pagrindinės vartotojo įvestys bei jų kombinacija:

- Priekinio plano/fono brūkšniai, naudojami atrinkti, kuri dalis turi būti išpjauta
- Negriežti apribojimai, nurodantys sritis, šalia kurių turėtų būti atliktas pjovimas
- Griežti apribojimai, pažymintys sritis, pro kurias pjaunantis kontūras būtinai turi praeiti.

2.4.1.2. Intelligent Mesh Scissoring Using 3D Snakes

Tai sumanus modelio pjaustymo algoritmas, kuriuo galima atlikti modelio sudalinimą automatiškai arba rankiniu būdu [6]. Vietoj to, kad pjovimas būtų atliktas grupuojant, šis algoritmas suranda ir nusako pjaunantįjį kontūrą. Šis kontūras gali būti randamas arba automatiškai, arba rankiniu būdu, arba nupiešiant brūkšnį, kuri pradėtų pjaustymo procesą.

2.4.1.3. An Enhanced Slicing Algorithm Using Nearest Distance Analysis for Layer Manufacturing

Tai trimačio modelio skerspjuvių radimo algoritmas, greitai supjaustantis modelius ir surandantis pjūvio kontūrus efektyviai ir be klaidų [7]. Šis algoritmas yra labiau skirtas supjaustyti modelį į sluoksnius trimačiam spausdinimui, tačiau skerspjuvio suradimas yra aktuali problema kuriamoje programinėje įrangoje.

2.4.1.4. Basic Triangle Mesh Slicing

Šis algoritmas padalina algoritmą pagal tai, kokioje pozicijoje pjaunančios plokštumos atžvilgiu yra nagrinėjamas modelio trikampis. Yra 4 variantai, kokiose pozicijose gali būti trikampis [8]:

- 1) Visas trikampis yra už pjaunančios plokštumos.
- 2) Visas trikampis yra pjaunančios plokštumos priekyje.
- 3) Dvi trikampio viršūnės yra už pjaunančios plokštumos, viena – priekyje.
- 4) Viena trikampio viršūnė yra už pjaunančios plokštumos, dvi – priekyje.

Trikampio pozicija plokštumos atžvilgiu surandama naudojantis skaliarinę sandaugą [10]. Suradus trikampus, kurie patenka į 3 ar 4 variantą, kiekvienai briaunai, kertančiai plokštumą, reikia

surasti vietą, kurioje yra susikirtimas. Tai atliekama naudojantis atkarpų ir plokštumų susikirtimo radimu [9].

2.4.1.5. Algoritmų palyginimas

Lentelė 1 Pjaustymo algoritmų palyginimas

Algoritmas	Pjovimas nurodžius plokštumą	Pjovimo kontūro suradimas	Sudėtingų figūrų pjovimas (kelių sluoksnių)	Algoritmo greitis	Vartotojo įvestis
Interactive Mesh Cutting Using Constrained Random Walks	Nėra	Yra	Nėra	Pakankamai greitas, kad vykdyti interaktyviai	Fono ir priekinio plano linijos ir griežti bei negriežti apribojimai
Intelligent Mesh Scissoring Using 3D Snakes	Yra	Yra	Nėra	Pakankamai greitas, kad vykdyti interaktyviai	Plokštuma, linija arba vykdoma be įvesties (supjaustoma automatiškai)
An Enhanced Slicing Algorithm Using Nearest Distance Analysis for Layer Manufacturing	Yra	Yra	Yra	Nežinomas	Plokštuma (supjaustoma sluoksniais)
Basic Triangle Mesh Slicing	Yra	Yra	Yra	Gana lėtas, kadangi tikrinami visi trikampiai	Plokštuma (surandamas skerspjuvis)

2.4.2. Pasirinktas geometrijos pjaustymo algoritmas

Realizacijai pasirinktas „*Basic Triangle Mesh Slicing*“ algoritmas, kadangi jo vienintėlis trūkumas yra jo sparta, tačiau šią problemą galima būtų išspręsti pasinaudojant algoritmo vykdymo lygiagretingumą.

Būtų taip pat galima naudoti „*An Enhanced Slicing Algorithm Using Nearest Distance Analysis for Layer Manufacturing*“ algoritmą, tačiau trūksta duomenų apie jo vykdymo greitį.

2.5. Užsakovai, pirkėjai ir kiti sistema suinteresuoti asmenys

2.5.1. Užsakovas

Kronion GmbH

Adresas:

Max-von-Laue-Str. 2a

76829 Landau

Vokietija

El. paštas:

E-mail of customer: S.Heinz@kronion.de

2.5.2. Pirkėjas

Kronion GmbH

Adresas:

Max-von-Laue-Str. 2a

76829 Landau

Vokietija

El. paštas:

E-mail of customer: S.Heinz@kronion.de

2.5.3. Vadovas

Dr. Šarūnas Packevičius

Adresas:

Kauno Technologijos Universitetas,

Studentų g. 50,

LT-51368, Kaunas, Lietuva

El. paštas: sarunas@ieee.org

2.6. Vartotojai

Sistemos vartotojas bus viena vartotojų grupė – automobilių trimačių modelių modeliuotojai.

Modeliuotojai naudosis programa norėdami patikrinti ar automobilio modelis yra kokybiškas.

Lentelė 2 Automobilių trimačių modelių modeliuotojas

Vartotojo kategorija:	Modeliuotojas
Vartotojo sprendžiami uždaviniai:	Trimačių modelių patikrinimas
Patirtis dalykinėje srityje:	Patyręs
Patirtis informacinėse technologijose:	Patyręs
Papildomos vartotojo charakteristikos:	-
Vartotojo prioritetai:	Svarbiausias vartotojas

2.7. Projekto apribojimai

2.7.1. Įpareigojantys apribojimai

2.7.1.1. Apribojimai sprendimui

Kuriama programa turi būti sukurta naudojantis Java programavimo kalbą, OpenGL kompiuterinės grafikos biblioteką bei JOGL biblioteką, kuri įgalina naudotis OpenGL Java programose. Taip pat, programa turi veikti Windows operacinėje sistemoje. Be to, sukurta sistema turi būti integruota į užsakovo naudojamą programą „eMMA Analyst“.

2.7.1.2. Diegimo aplinka

Programa bus diegiama į kompiuterius su įdiegta „eMMA Analyst“ programa bei Windows operacine sistema.

2.7.1.3. Bendradarbiaujančios sistemos

Kuriama sistema bendradarbiaus su užsakovo naudojama programa „eMMA Analyst“. Kitokių bendradarbiavimo ryšių produktas neturės.

2.7.1.4. Komerciniai specializuoti programų paketai

Sistemos kūrimui bus naudojamos dvi atvirojo kodo bibliotekos: OpenGL bei JOGL.

OpenGL – kompiuterinės grafikos biblioteka, skirta dvimatės ir trimatės vektorinės grafikos atvaizdavimui.

JOGL – tai biblioteka, kuri suteikia galimybę naudotis OpenGL biblioteka Java programavimo kalboje.

2.7.1.5. Numatoma darbo vietos aplinka

Numatoma darbo vieta yra biuras, todėl tai neturi įtakos programinės įrangos realizavimui

2.8. Svarbūs faktai ir prielaidos

2.8.1. Svarbūs faktai

Kūrimo metu naudojamos bibliotekos bus nemokamos ir joms nereikės papildomų sąnaudų.

2.8.2. Prielaidos

Prielaidos, kuriomis bus remiamasi kūrimo metu:

- Produktas bus naudojamas kompiuteriuose su Windows operacine sistema bei integruojamas į „eMMA Analyst“ programą.
- „eMMA Analyst“ turės nesudėtingą išplėtimo sąsają.

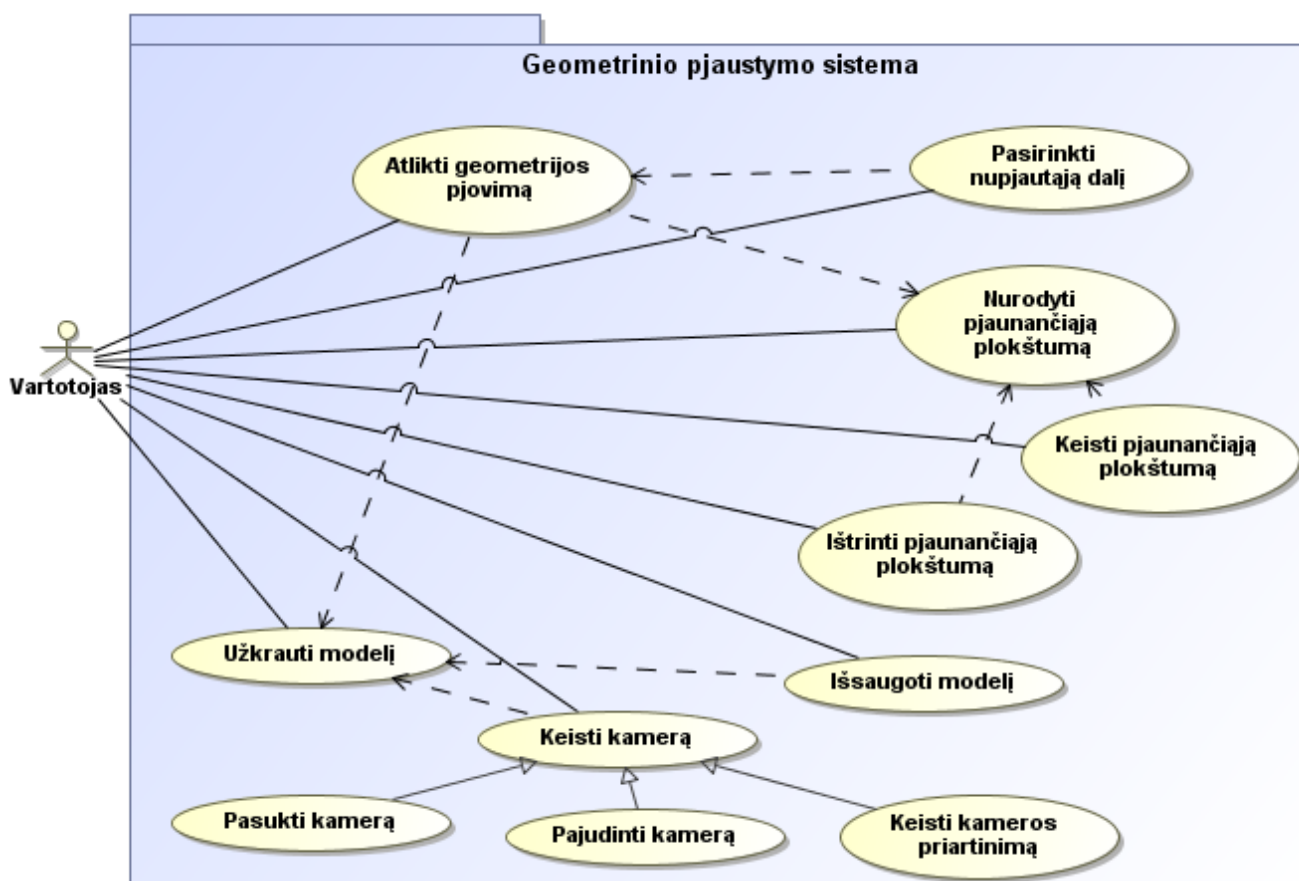
3. PROJEK TINĖ DALIS

3.1. Reikalavimų specifikacija

3.1.1. Sistemos funkcijos

- 1) Užkrauti ir peržiūrėti trimatį modelį iš įvairių pusių
- 2) Nurodyti pjaunančiąją plokštumą
- 3) Keisti pjaunančiąją plokštumą
- 4) Ištrinti pjaunančiąją plokštumą
- 5) Atlikti geometrijos pjūvį
- 6) Išsaugoti modelį
- 7) Pasukti kamerą
- 8) Pajudinti kamerą
- 9) Keisti kameros priartinimą

Žemiau pateikta sistemos panaudos atvejų diagrama (1 pav. Sistemos panaudojimo atvejų diagrama)



1 pav. Sistemos panaudojimo atvejų diagrama

3.1.2. Panaudojimo atvejų sąrašas

1. Pjaunančios plokštumos nurodymas	
Aktorius:	Vartotojas
Aprašas:	Nurodyti plokštumą, kuri padalins modelį
Prieš sąlygos:	Užkrautas trimatis modelis
Sužadinimo sąlygos:	Vartotojas nurodo pjaunančiąją plokštumą
Po sąlygos:	Sukurama pjaunančioji plokštuma

2. Pjaunančios plokštumos keitimas	
Aktorius:	Vartotojas
Aprašas:	Pakeisti pjaunančiąją plokštumą
Prieš sąlygos:	Sukurta pjaunančioji plokštuma
Sužadinimo sąlygos:	Vartotojas pakeičia pjaunančiąją plokštumą
Po sąlygos:	Pjaunančioji plokštuma pakeičiama

3. Pjaunančios plokštumos trynimasis	
Aktorius:	Vartotojas
Aprašas:	Ištrinti pjaunančiąją plokštumą
Prieš sąlygos:	Sukurta pjaunančioji plokštuma
Sužadinimo sąlygos:	Vartotojas ištrina pjaunančiąją plokštumą
Po sąlygos:	Pjaunančioji plokštuma ištrinama

4. Pjovimas	
Aktorius:	Vartotojas
Aprašas:	Atlikti trimačio modelio pjovimą su pjaunančiąją plokštuma
Prieš sąlygos:	Sukurta pjaunančioji plokštuma
Sužadinimo sąlygos:	Vartotojas paspaudžia pjovimo mygtuką
Po sąlygos:	Atliekamas geometrijos pjovimas

5. Nupjautos dalies pasirinkimas	
Aktorius:	Vartotojas
Aprašas:	Pasirinkti modelio dalį, kurios neturėtų rodyti po pjovimo
Prieš sąlygos:	Atliktas geometrijos pjovimas
Sužadinimo sąlygos:	Vartotojas pasirenka nupjautą dalį
Po sąlygos:	Pasirinktos dalis neberodoma arba rodoma permatomai

6. Modelio užkrovimas	
Aktorius:	Vartotojas
Aprašas:	Užkrauti trimatį modelį
Prieš sąlygos:	Užkrautas trimatis modelis
Sužadinimo sąlygos:	Pasirenkamas modelio užkrovimas
Po sąlygos:	Modelis užkraunamas

7. Modelio išsaugojimas	
Aktorius:	Vartotojas
Aprašas:	Išsaugoti trimatį modelį
Prieš sąlygos:	Užkrautas trimatis modelis
Sužadinimo sąlygos:	Pasirenkamas modelio išsaugojimas
Po sąlygos:	Modelis išsaugojamas

8. Pasukti kamerą	
Aktorius:	Vartotojas
Aprašas:	Pasukti kamerą
Prieš sąlygos:	Paleista programinė įranga
Sužadinimo sąlygos:	Vartotojas paspaudžia kameros pasukimo mygtuką
Po sąlygos:	Kamera pasukama

9. Pajudinti kamerą	
Aktorius:	Vartotojas
Aprašas:	Pajudinti kamerą
Prieš sąlygos:	Paleista programinė įranga
Sužadinimo sąlygos:	Vartotojas paspaudžia kameros pajudavimo mygtuką
Po sąlygos:	Kamera pajudinama

10. Keisti kameros priartinimą	
Aktorius:	Vartotojas
Aprašas:	Keisti kameros priartinimą
Prieš sąlygos:	Paleista programinė įranga
Sužadinimo sąlygos:	Vartotojas paspaudžia kameros priartinimo keitimo mygtuką
Po sąlygos:	Pakeičiamas kameros priartinimas

3.2. Funkciniai reikalavimai ir reikalavimai duomenims

3.2.1. Funkciniai reikalavimai

Reikalavimas #:	1	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	1
Aprašymas:	Sistema turi leisti nurodyti pjaunančiąją plokštumą erdvėje.				
Pagrindimas:	Be pjaunančios plokštumos negalima atlikti pjovimo.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Galima nurodyti pjaunančiąją plokštumą.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	7	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	2	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	2
Aprašymas:	Sistema turi leisti keisti pjaunančiąją plokštumą.				
Pagrindimas:	Koreguoti plokštumą greičiau nei sukurti naują.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Galima keisti pjaunančiąją plokštumą.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	1, 7	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	3	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	3
Aprašymas:	Sistema turi leisti ištrinti pjaunančiąją plokštumą.				
Pagrindimas:	Sukurta plokštuma gali neatitikti norų.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Galima ištrinti pjaunančiąją plokštumą.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	1, 7	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	4	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	4
Aprašymas:	Sistema turi leisti atlikti geometrijos pjovimą.				
Pagrindimas:	-				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Galima atlikti geometrijos pjovimą.				
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5		
Priklausomybės:	1, 7	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	5	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	5
Aprašymas:	Sistema turi leisti pasirinkti, kuri dalis yra nupjautoji ir kuri dalis turi būti vaizduojama.				
Pagrindimas:	Galimybė aiškiai matyti nupjautas dalis.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Leidžiama pasirinkti nupjautą ir vaizduojamą dalį.				
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	4		
Priklausomybės:	4, 7	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	6	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	6
Aprašymas:	Sistema turi leisti atidaryti trimačio modelio failą.				
Pagrindimas:	Pjaustoma trimačio modelio geometrija.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Failas atidaromas.				
Užsakovo tenkinimas:	3			Užsakovo netenkinimas:	5
Priklausomybės:	Nėra.			Konfliktai:	Nėra.
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	7	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	6
Aprašymas:	Sistema turi atvaizduoti atidarytą trimatį modelį.				
Pagrindimas:	Vartotojas turi matyti, ką pjausto.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Modelis atvaizduojamas.				
Užsakovo tenkinimas:	5			Užsakovo netenkinimas:	5
Priklausomybės:	6			Konfliktai:	Nėra.
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	8	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	8
Aprašymas:	Sistema turi leisti pasukti kamerą, atvaizduojančią trimatę erdvę				
Pagrindimas:	Modelį gali reikėti apžiūrėti iš visų pusių.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Kamerą galima pasukti visomis kryptimis.				
Užsakovo tenkinimas:	3			Užsakovo netenkinimas:	4
Priklausomybės:	7			Konfliktai:	Nėra.
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

Reikalavimas #:	9	Reikalavimo tipas:	9a	Įvykis/panaudojimo atvejis #:	9
Aprašymas:	Sistema turi leisti pajudinti kamerą, atvaizduojančią trimatę erdvę				
Pagrindimas:	Modelį gali reikėti apžiūrėti iš skirtingų vietų.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Kamerą galima pajudinti visomis kryptimis.				
Užsakovo tenkinimas:	3			Užsakovo netenkinimas:	4
Priklausomybės:	7			Konfliktai:	Nėra.
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 1d.				

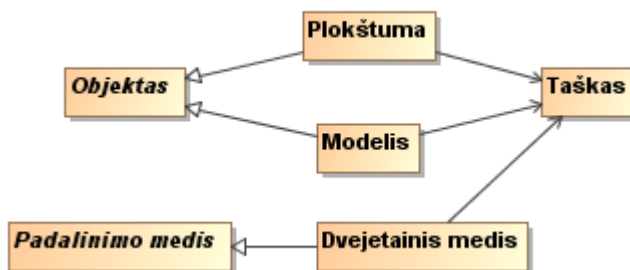
Reikalavimas #:	10	Reikalavimo tipas:	9a Įvykis/panaudojimo atvejis #:	10
Aprašymas:	Sistema turi leisti padidinti kameros rodomą vaizdą.			
Pagrindimas:	Modelį gali būti lengviau apžiūrėti iš arčiau/toliau.			
Šaltinis:	Užsakovas.			
Tikimo kriterijus:	Vaizdą galima padidinti.			
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3	
Priklausomybės:	7	Konfliktai:	Nėra.	
Papildoma medžiaga:	-			
Istorija:	Užregistruotas 2014 gegužės 1d.			

Reikalavimas #:	11	Reikalavimo tipas:	9a Įvykis/panaudojimo atvejis #:	7
Aprašymas:	Sistema turi leisti išsaugoti trimatį modelį.			
Pagrindimas:	Vėlesniam naudojimui gali būti reikalingas padalintas modelis.			
Šaltinis:	Užsakovas.			
Tikimo kriterijus:	Galima išsaugoti trimatį modelį.			
Užsakovo tenkinimas:	3	Užsakovo netenkinimas:	3	
Priklausomybės:	7	Konfliktai:	Nėra.	
Papildoma medžiaga:	-			
Istorija:	Užregistruotas 2014 gegužės 1d.			

Reikalavimas #:	12	Reikalavimo tipas:	9a Įvykis/panaudojimo atvejis #:	4
Aprašymas:	Pjovimas turi būti vykdomas lygiagrečiai.			
Pagrindimas:	Sistema veiks interaktyvioje aplinkoje.			
Šaltinis:	Užsakovas.			
Tikimo kriterijus:	Pjovimas atliekamas lygiagrečiai.			
Užsakovo tenkinimas:	5	Užsakovo netenkinimas:	5	
Priklausomybės:	4	Konfliktai:	Nėra.	
Papildoma medžiaga:	-			
Istorija:	Užregistruotas 2014 gegužės 1d.			

3.2.2. Reikalavimai duomenims

Duomenų modelio diagrama (2 pav. Duomenų modelio diagrama):



2 pav. Duomenų modelio diagrama

3.3. Nefunkciniai reikalavimai

3.3.1. Reikalavimai sistemos išvaizdai

Reikalavimas #:	11	Reikalavimo tipas:	10a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Vartotojo sąsaja turi būti paprasta ir lengvai skaitoma.				
Pagrindimas:	Vartotojai gali nenorėti naudotis nepatogia programa				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Trimačiai modeliai nenuitekinami į išorę.				
Užsakovo tenkinimas:	1	Užsakovo netenkinimas:	4		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.3.2. Reikalavimai panaudojamumui

Reikalavimas #:	12	Reikalavimo tipas:	11a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Paprastai panaudojamas asmens dirbančio su kompiuterine grafika.				
Pagrindimas:	Vartotojai dirbs su programa be papildomo apmokymo.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Žmogus išmoksta dirbti su programa per vieną valandą.				
Užsakovo tenkinimas:	2	Užsakovo netenkinimas:	5		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.3.3. Reikalavimai vykdymo charakteristikoms

Reikalavimas #:	13	Reikalavimo tipas:	12a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Sistema turi veikti greitai.				
Pagrindimas:	Sistema veiks interaktyvioje aplinkoje.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Sistema sklandžiai veikia interaktyvioje aplinkoje				
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	5		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

Reikalavimas #:	14	Reikalavimo tipas:	12a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Sistema turi veikti be trikdžių.				
Pagrindimas:	Trikdžiai mažina darbo našumą.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Sistema didžiąją dalį laiko veikia be sutrikimų.				
Užsakovo tenkinimas:	4	Užsakovo netenkinimas:	5		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.3.4. Reikalavimai sistemos priežiūrai

Reikalavimas #:	15	Reikalavimo tipas:	14a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Sistema turi veikti Windows operacinėje sistemoje.				
Pagrindimas:	Užsakovo įmonėje naudojama Windows operacinė sistema.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Sistema pritaikyta veikti Windows operacinėje sistemoje.				
Užsakovo tenkinimas:	1	Užsakovo netenkinimas:	5		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.3.5. Reikalavimai saugumui

Reikalavimas #:	16	Reikalavimo tipas:	15a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Sistemoje naudojami trimačiai modeliai neturi būti prieinami išorėje.				
Pagrindimas:	Tik atitinkami žmonės gali prieiti prie trimačių modelių.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Trimačiai modeliai nenuitekinami į išorę.				
Užsakovo tenkinimas:	2	Užsakovo netenkinimas:	5		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.3.6. Kultūriniai-politinai reikalavimai

Reikalavimas #:	17	Reikalavimo tipas:	16a	Įvykis/panaudojimo atvejis #:	1-8
Aprašymas:	Sistemos vartotojo sąsaja turi būti anglų kalba.				
Pagrindimas:	Užsakovo įmonės programose naudojama anglų kalba.				
Šaltinis:	Užsakovas.				
Tikimo kriterijus:	Sistemos vartotojo sąsaja realizuota anglų kalba.				
Užsakovo tenkinimas:	1	Užsakovo netenkinimas:	4		
Priklausomybės:	Nėra.	Konfliktai:	Nėra.		
Papildoma medžiaga:	-				
Istorija:	Užregistruotas 2014 gegužės 7d.				

3.4. Statinis sistemos vaizdas

3.4.1. Sistemos paketai

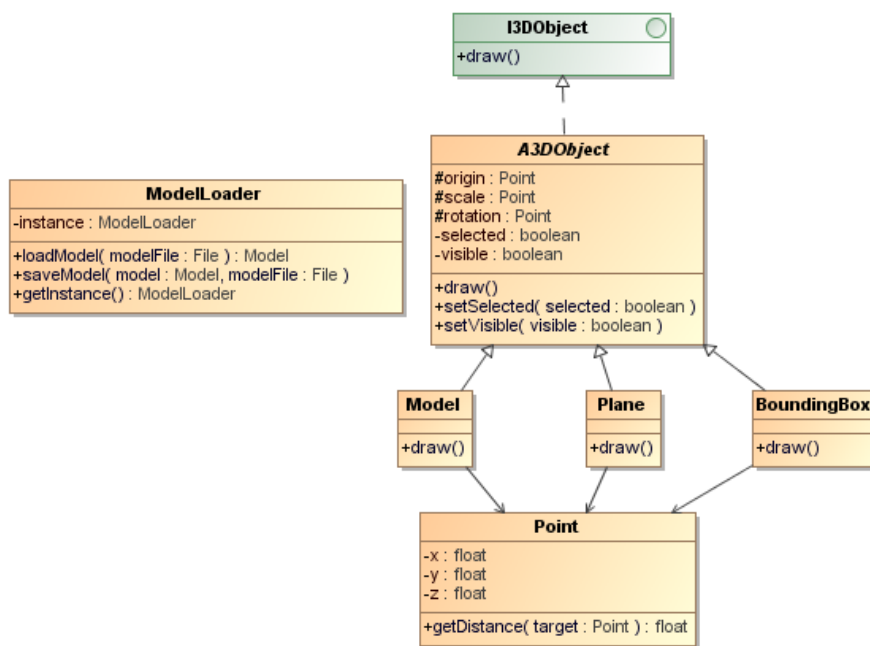
Sistemos išskaidymas į paketus parodomas paveikslėlyje 3 pav. Sistemos paketai.



3 pav. Sistemos paketai

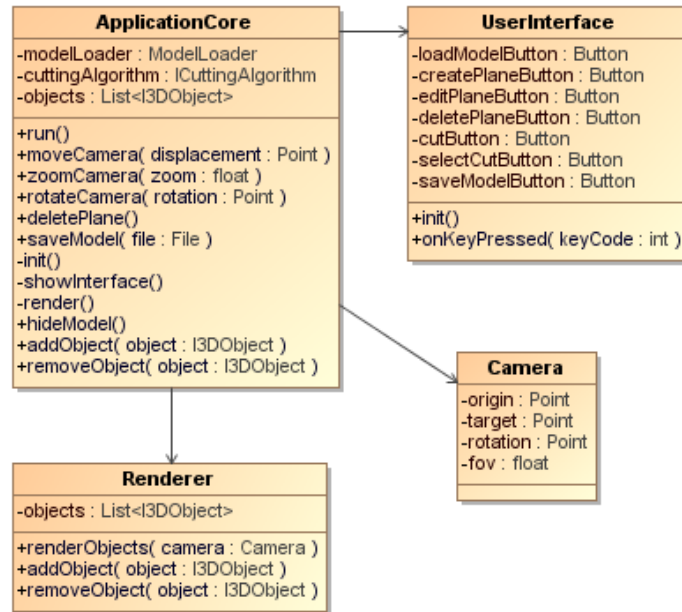
3.4.2. Klasių diagramos

3.4.2.1. Objektų užkrovimo ir išsaugojimo paketo klasių diagrama



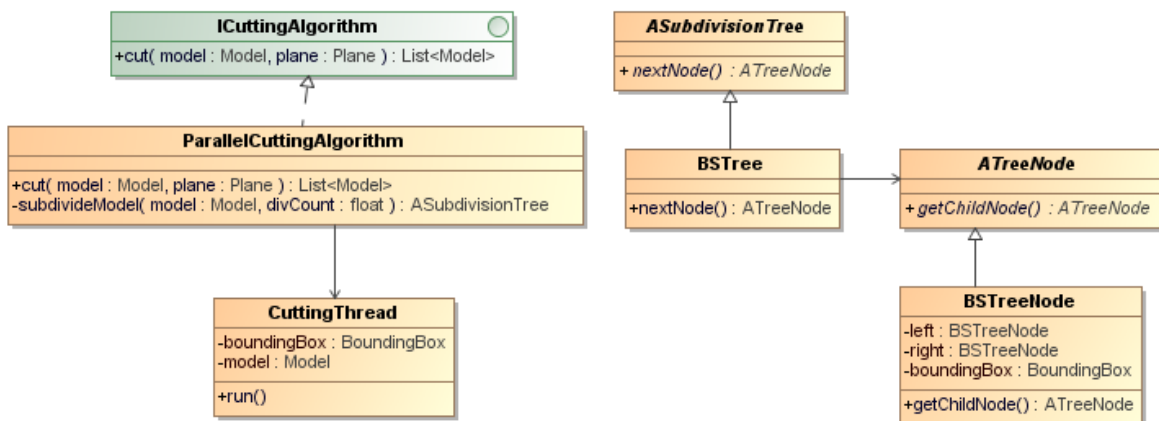
4 pav. Objektų užkrovimo ir išsaugojimo paketo klasių diagrama

3.4.2.2. Atvaizdavimo paketo klasių diagrama



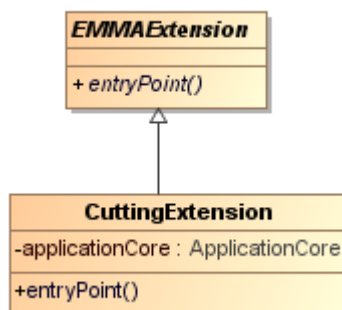
5 pav. Atvaizdavimo paketo klasių diagrama

3.4.2.3. Geometrijos pjaustymo paketo klasių diagrama



6 pav. Geometrijos pjaustymo paketo klasių diagrama

3.4.2.4. eMMA sąsajos paketo klasių diagrama



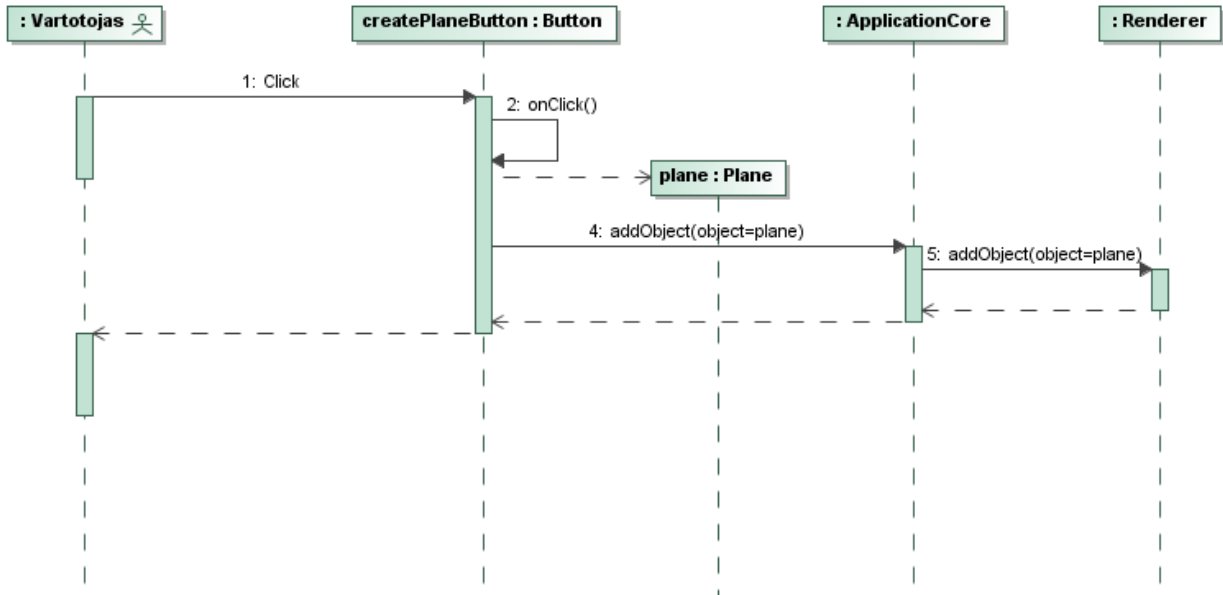
7 pav. eMMA sąsajos paketo klasių diagrama

3.5. Dinaminis sistemos vaizdas

3.5.1. Sekos diagramos

3.5.1.1. Pjaunančiosios plokštumos nurodymo sekų diagrama

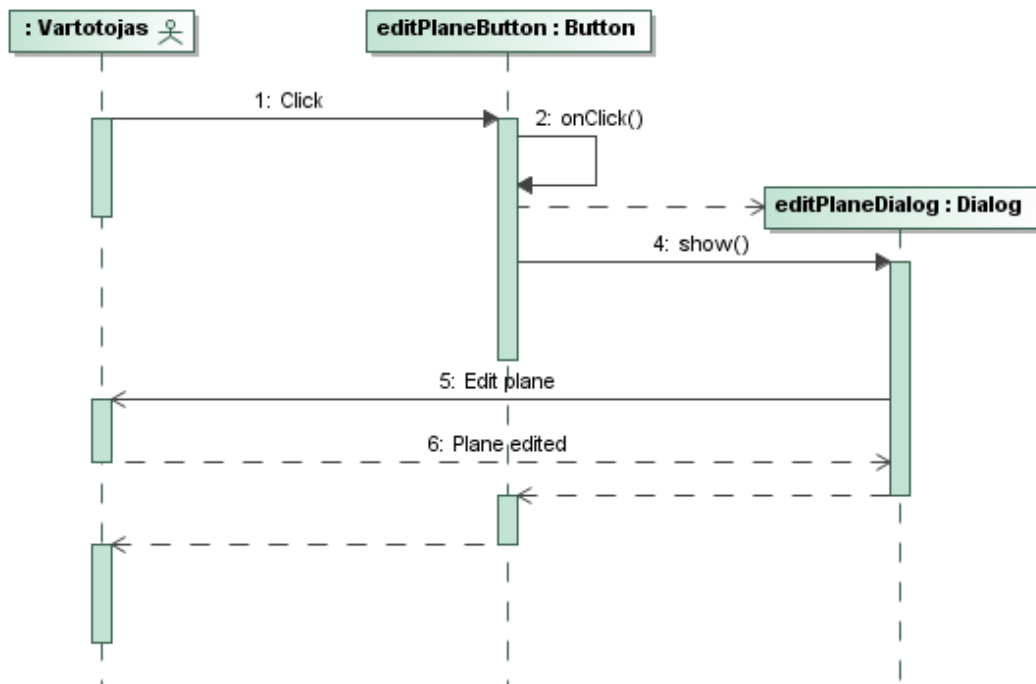
Pjaunančiosios plokštumos nurodymo sekų diagrama pateikiama paveikslėlyje 8 pav. Pjaunančiosios plokštumos nurodymo sekų diagrama.



8 pav. Pjaunančiosios plokštumos nurodymo sekų diagrama

3.5.1.2. Pjaunančiosios plokštumos keitimo sekų diagrama

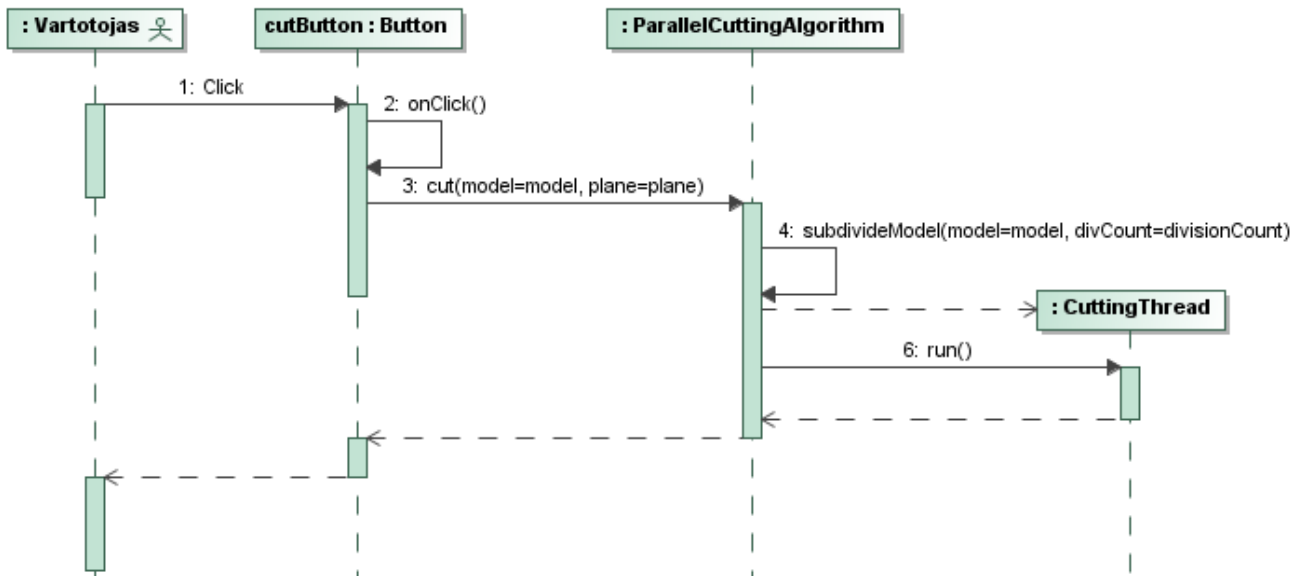
Pjaunančiosios plokštumos keitimo sekų diagrama pateikiama paveikslėlyje 9 pav. Pjaunančiosios plokštumos keitimo sekų diagrama.



9 pav. Pjaunančiosios plokštumos keitimo sekų diagrama

3.5.1.3. Pjovimo sekų diagrama

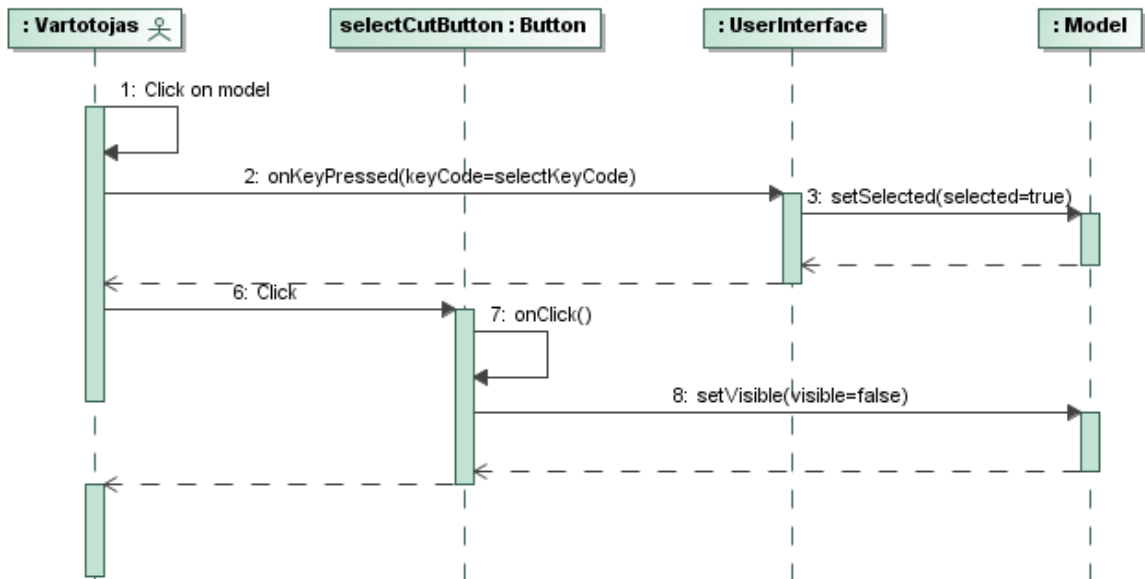
Pjovimo sekų diagrama pateikiama paveikslėlyje 10 pav. Pjovimo sekų diagrama.



10 pav. Pjovimo sekų diagrama

3.5.1.4. Nupjautos dalies pasirinkimo sekų diagrama

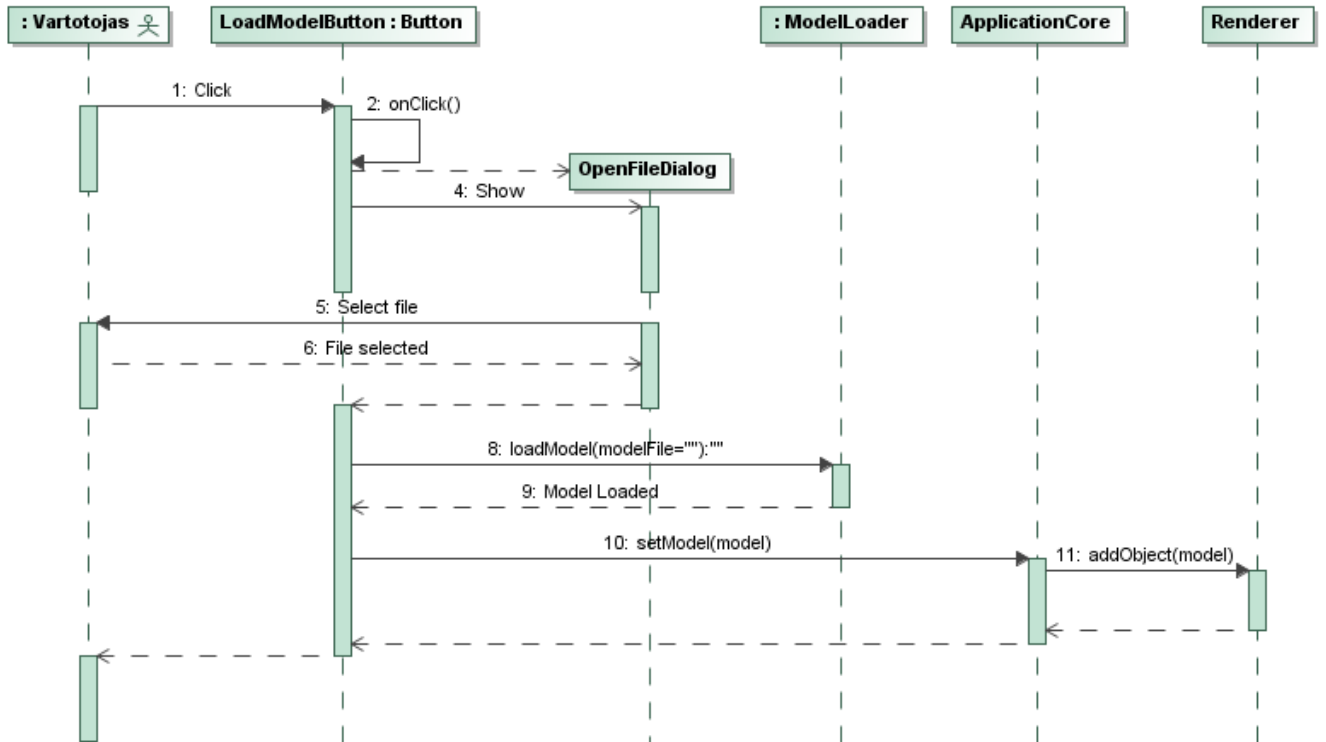
Nupjautos dalies pasirinkimo sekų diagrama pateikiama paveikslėlyje 11 pav. Nupjautos dalies pasirinkimo sekų diagrama.



11 pav. Nupjautos dalies pasirinkimo sekų diagrama

3.5.1.5. Modelio užkrovimo sekų diagrama

Modelio užkrovimo sekų diagrama pateikiama paveikslėlyje 12 pav. Modelio užkrovimo sekų diagrama.

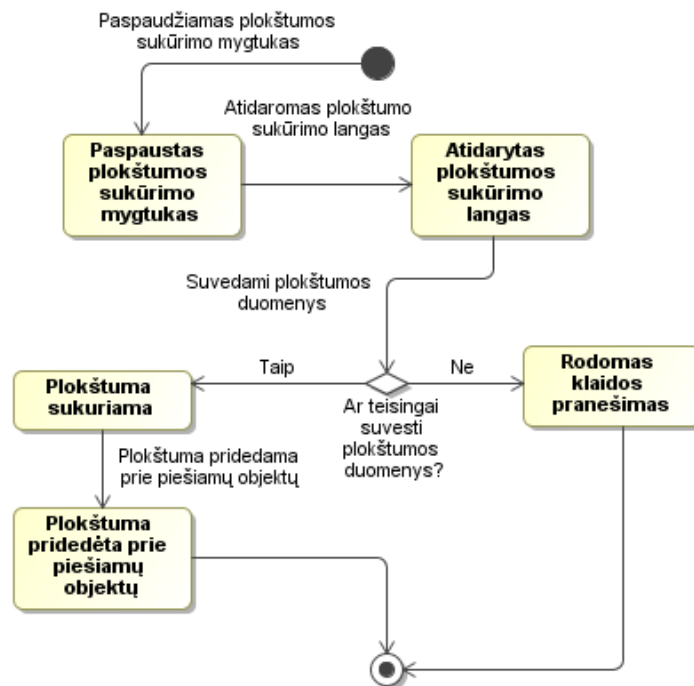


12 pav. Modelio užkrovimo sekų diagrama

3.5.2. Būsenų diagramos

3.5.2.1. Pjaunančiosios plokštumos nurodymo būsenų diagrama

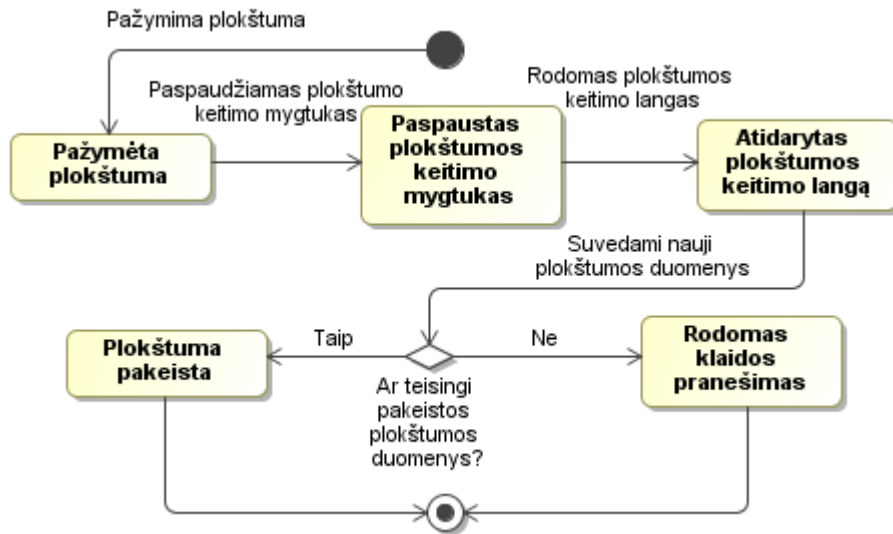
Pjaunančiosios plokštumos nurodymo būsenų diagrama pateikiama paveikslėlyje 13 pav. Pjaunančiosios plokštumos nurodymo būsenų diagrama.



13 pav. Pjaunančiosios plokštumos nurodymo būsenų diagrama

3.5.2.2. Pjaunančiosios plokštumos keitimo būsenų diagrama

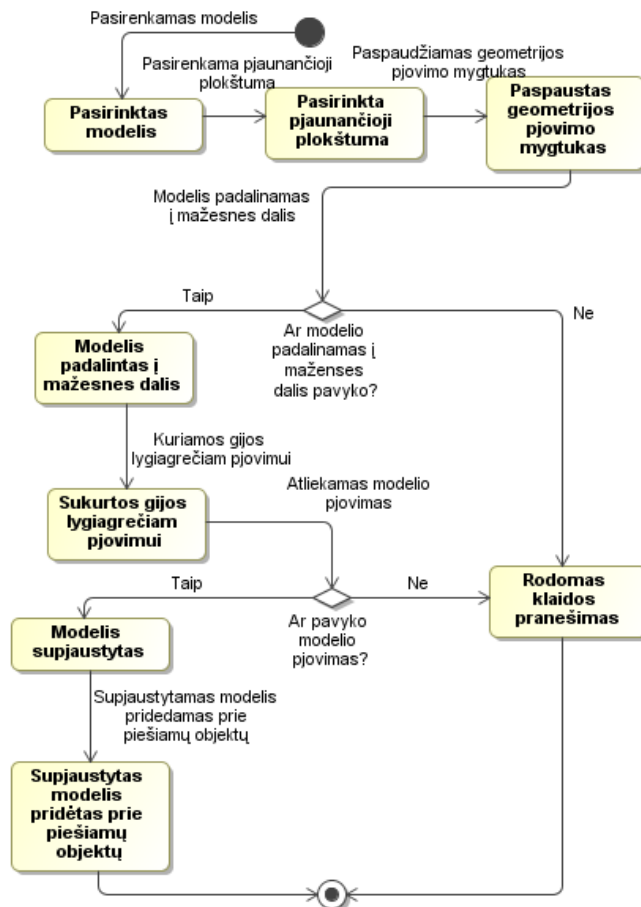
Pjaunančiosios plokštumos keitimo būsenų diagrama pateikiama paveikslėlyje 14 pav. Pjaunančiosios plokštumos keitimo būsenų diagrama.



14 pav. Pjaunančiosios plokštumos keitimo būsenų diagrama

3.5.2.3. Pjovimo būsenų diagrama

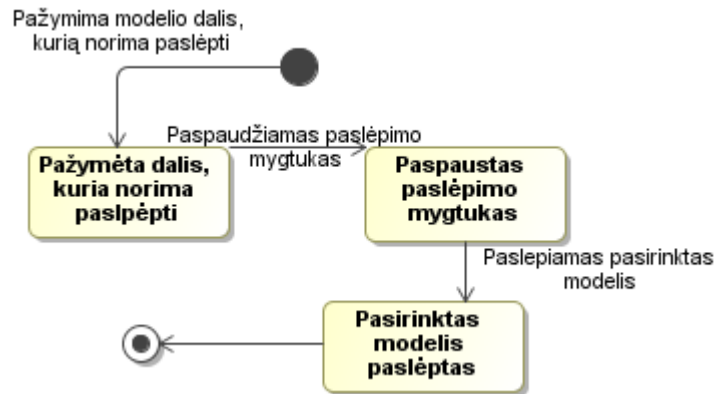
Pjovimo būsenų diagrama pateikiama paveikslėlyje 15 pav. Pjovimo būsenų diagrama.



15 pav. Pjovimo būsenų diagrama

3.5.2.4. Nupjautos dalies pasirinkimo būsenų diagrama

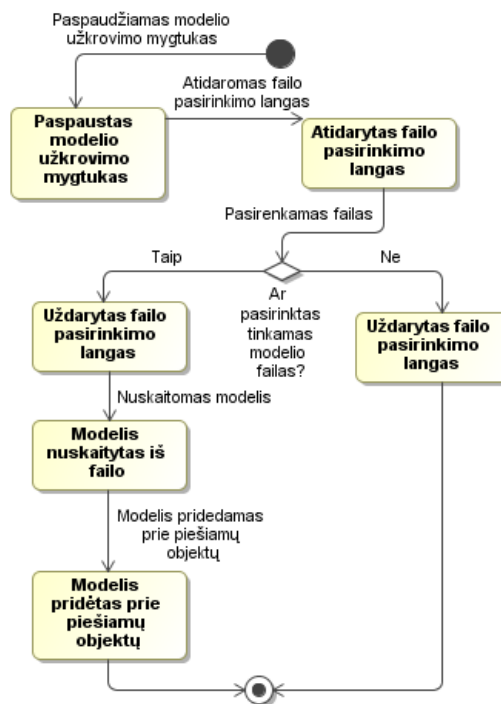
Nupjautos dalies pasirinkimo būsenų diagrama pateikiama paveikslėlyje 16 pav. Nupjautos dalies pasirinkimo būsenų diagrama.



16 pav. Nupjautos dalies pasirinkimo būsenų diagrama

3.5.2.5. Modelio užkrovimo būsenų diagrama

Modelio užkrovimo būsenų diagrama pateikiama paveikslėlyje 17 pav. Modelio užkrovimo būsenų diagrama.

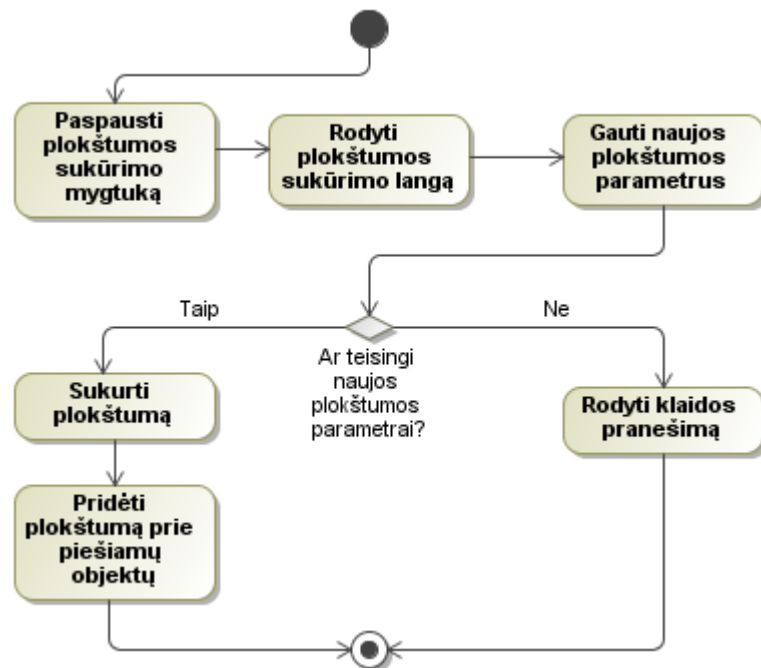


17 pav. Modelio užkrovimo būsenų diagrama

3.5.3. Veiklos diagramos

3.5.3.1. Pjaunančiosios plokštumos nurodymo veiklos diagrama

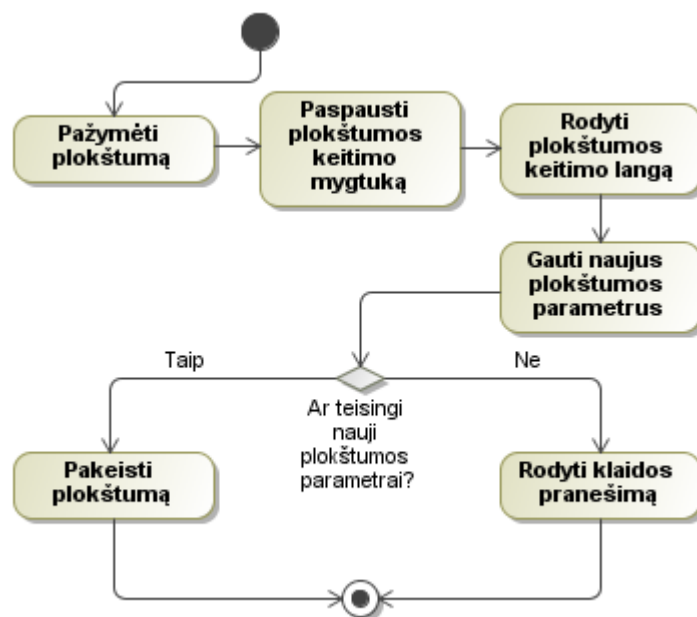
Pjaunančiosios plokštumos nurodymo veiklos diagrama pateikiama paveikslėlyje 18 pav. Pjaunančiosios plokštumos nurodymo veiklos diagrama.



18 pav. Pjaunančiosios plokštumos nurodymo veiklos diagrama

3.5.3.2. Pjaunančiosios plokštumos keitimo veiklos diagrama

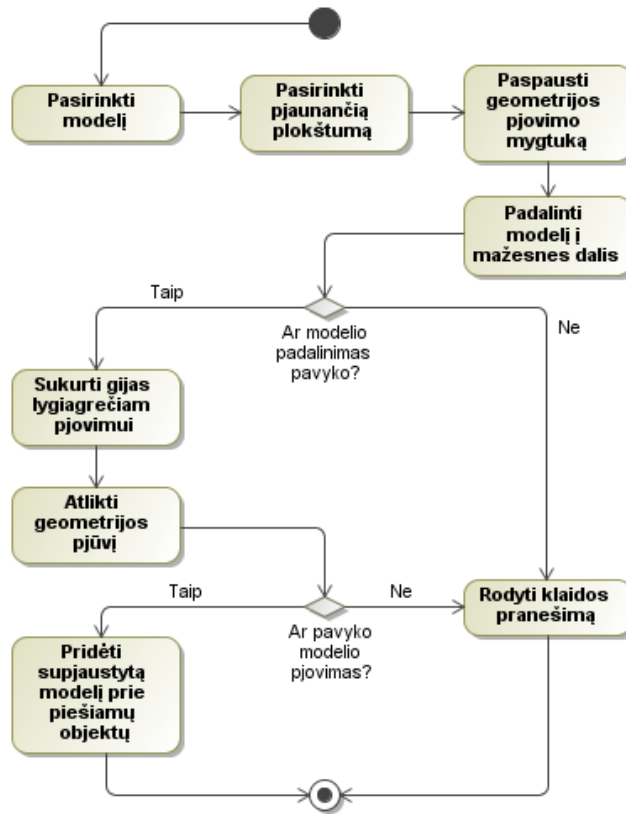
Pjaunančiosios plokštumos keitimo veiklos diagrama pateikiama paveikslėlyje 19 pav. Pjaunančiosios plokštumos keitimo veiklos diagrama **Error! Reference source not found.**



19 pav. Pjaunančiosios plokštumos keitimo veiklos diagrama

3.5.3.3. Pjovimo veiklos diagrama

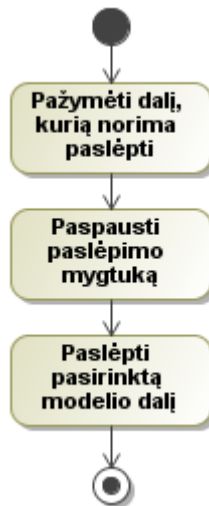
Pjovimo veiklos diagrama pateikiama paveikslėlyje 20 pav. Pjovimo veiklos diagrama.



20 pav. Pjovimo veiklos diagrama

3.5.3.4. Nupjautos dalies pasirinkimo veiklos diagrama

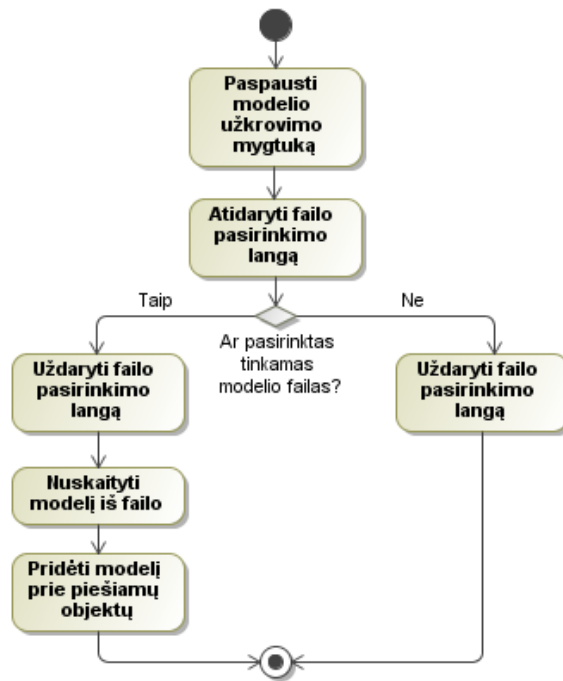
Nupjautos dalies pasirinkimo veiklos diagrama pateikiama paveikslėlyje 21 pav. Nupjautos dalies pasirinkimo veiklos diagrama.



21 pav. Nupjautos dalies pasirinkimo veiklos diagrama

3.5.3.5. Modelio užkrovimo veiklos diagrama

Modelio užkrovimo veiklos diagrama pateikiama paveikslėlyje 22 pav. Modelio užkrovimo veiklos diagrama.



22 pav. Modelio užkrovimo veiklos diagrama

4. TYRIMO DALIS

4.1. Tikslas

Tyrimo tikslas yra išsiaiškinti, kaip galima pagerinti sudėtingos geometrijos pjaustymo algoritmo greitaveiką tam, kad pjaustymą būtų galima atlikti realiu laiku (pjaustymo įvykdymo laikas nebūtų ilgesnis nei 16.6 milisekundės, kad vartotojas nematytų sulėtėjimo). Šiam tikslui pasiekti reikia ištirti pjaustymo algoritmo lygiagrečio galimybes.

Sukurta automobilio kėbulo geometrijos pjaustymo sistema šiuo metu naudoja vieną pjaustymo algoritmą, kuris yra vykdomas nuosekliai.

Tyrimo metu nebus nagrinėjama kitų, programoje nepanaudotų, nuosekliai vykdomų algoritmų greitaveika.

4.2. Sukurtos sistemos pjaustymo algoritmo tyrimas

Tam, kad įsitikinti ar nuoseklus algoritmo vykdymo laikas yra priimtinas atliekant sudėtingos geometrijos pjaustymą, buvo atliktas jo greitaveikos tyrimas.

Buvo pasirinkti devyni trimačiai automobilių modeliai su skirtingu trikampių kiekiu. Visų bandymų metu plokštuma buvo tokia pati (išilgai automobilio modelio). Kiekvienas modelis buvo padalinamas penkis kartus, kiekvieną kartą registruojant vykdymo laiką. Galiausiai paimamas kiekvieno modelio pjaustymo vykdymo laikų vidurkis.

4.3. Tyrimui naudojamos įrangos aprašas

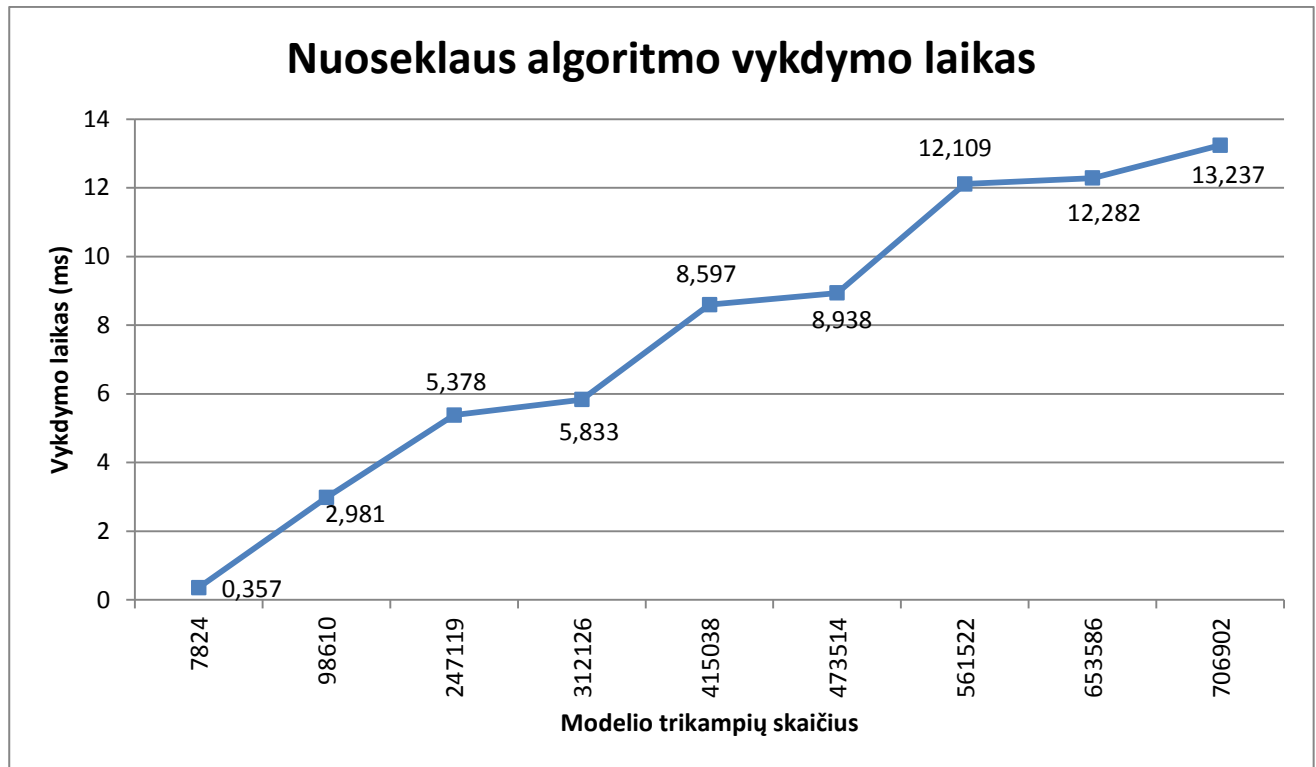
Tyrimui naudojamas kompiuteris, turintis tokius parametrus:

- Centrinis procesorius: Intel(R) Core(TM) i7-3770K CPU @ 3.50GHz 3.90GHz
- Operatyvioji atmintis: 24GB DDR3
- Grafinis procesorius: AMD Radeon HD7950

Tyrimų rezultatai yra priklausomi nuo aukščiau aprašytos įrangos. Naudojantis kita įranga, gali būti gauti skirtingi arba visiškai priešingi rezultatai.

4.4. Nuoseklaus pjovimo algoritmo tyrimo rezultatai

Bandymų rezultatai surikiuoti modelio sudėtingumo didėjimo tvarka (kuo daugiau trikampių, tuo modelis sudėtingesnis) atvaizduoti diagramoje 23 pav. Nuoseklaus algoritmo greitimeika.



23 pav. Nuoseklaus algoritmo greitimeika

Iš diagramos matyti, kad algoritmo vykdymo laikas yra tiesiogiai proporcingas modelio trikampių skaičiui. Remiantis rezultatais, galima apskaičiuoti, kada algoritmo vykdymas nebus priimtinas ir viršys 16,6 ms.

$$\text{Trikampių skaičius} = \frac{16,6\text{ms}}{\left(\frac{13,237\text{ms} - 0,357\text{ms}}{706902 - 7824}\right)} = \sim 900986$$

Taigi, jei modelį sudaro daugiau nei 900986 trikapiai, tada algoritmas pradeda nebetenkinti greitimeikos reikalavimų. Įvertinus tai, kad sudėtingi modeliai, kurie yra naudojami automobilių pramonėje, gali turėti kelis milijonus trikampių, galima teigti, kad šis algoritmas nėra pakankamai spartus.

5. EKSPERIMENTINĖ DALIS

5.1. Tikslas

Atliktų eksperimentinių tyrimų tikslas – ištirti, kokie algoritmo lygiagretinimo būdai gali padėti pasiekti tinkamą algoritmo spartą sudėtingų automobilių modelių pjaustymui.

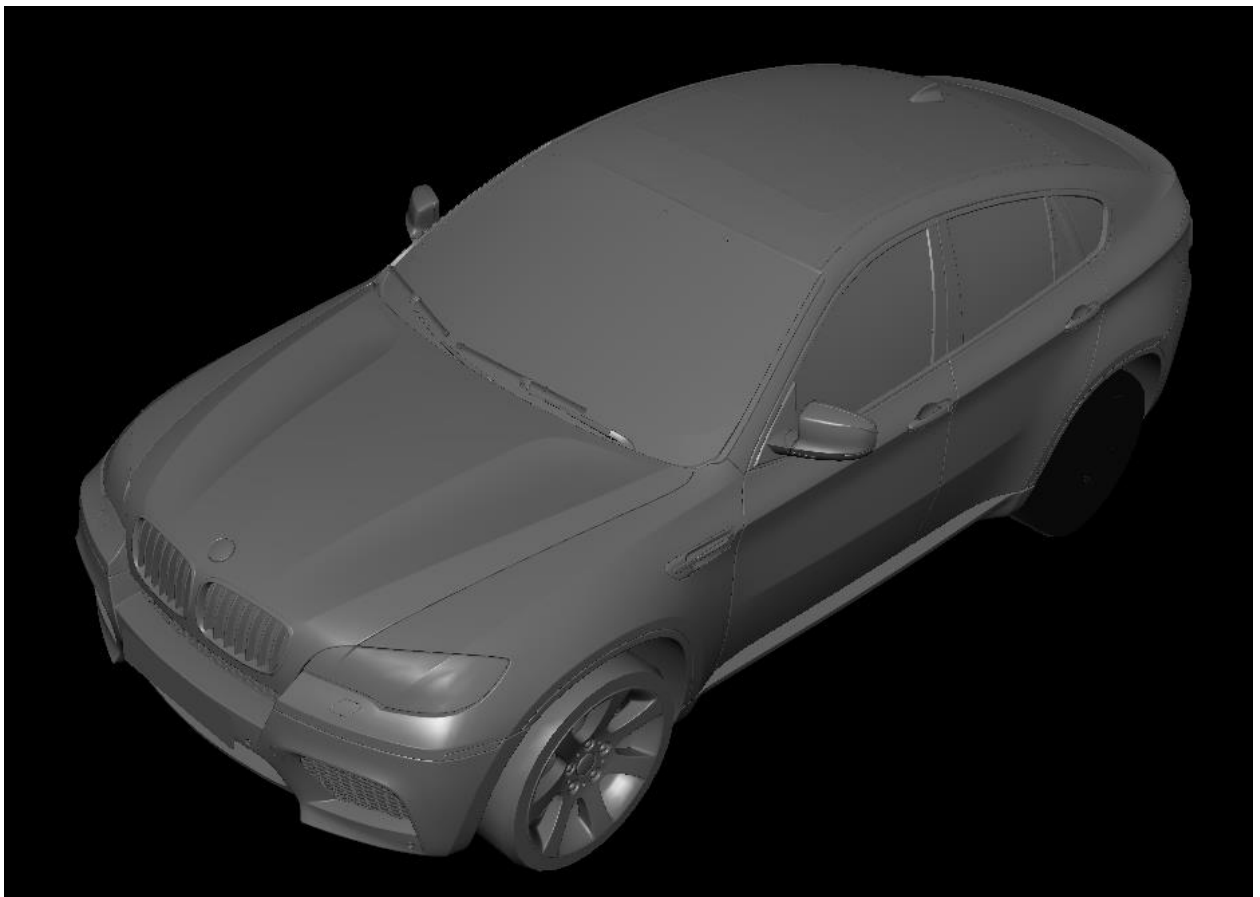
5.2. Eksperimentų aprašymas

Buvo atliekami tokie algoritmo lygiagretinimo eksperimentai:

- Lygiagretinimas naudojantis Java 8 lygiagrečius srautus (angl. *parallel stream*).
- Lygiagretinimas naudojantis OpenCL.

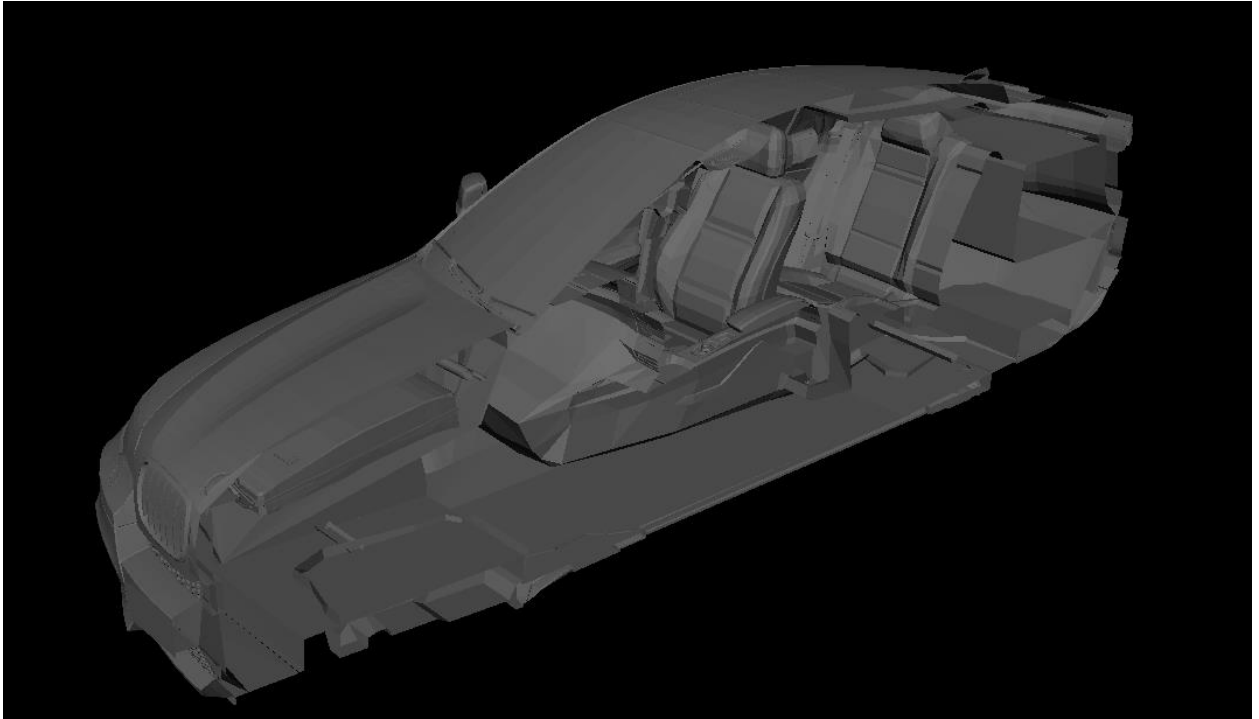
Visi eksperimentai buvo atlikti naudojantis technine įranga aprašyta skyriuje 4.3.

Eksperimentui naudotas sudėtingiausias modelis pavaizduotas 24 pav. Sudėtingiausias naudotas modelis.



24 pav. Sudėtingiausias naudotas modelis

Kaip modelis atrodė po pjovimo atvaizduota paveikslėlyje 25 pav. Modelis po pjovimo.

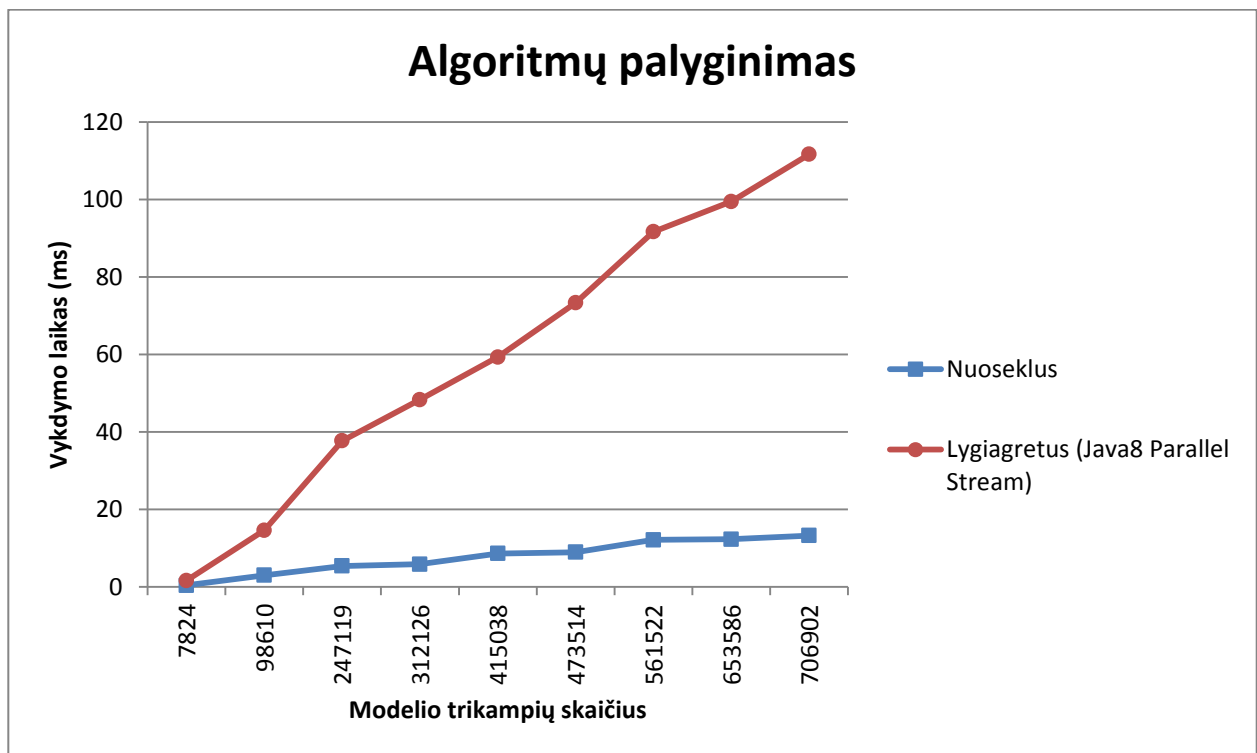


25 pav. Modelis po pjovimo

5.3. Eksperimentų rezultatai

5.3.1. Nuoseklaus algoritmo palyginimas su Java 8 lygiagrečiu algoritmu

Nuoseklaus algoritmo ir lygiagretaus algoritmo, naudojant Java 8 lygiagrečius srautus, palyginimas atvaizduotas diagramoje 26 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (Java8 Parallel Stream).



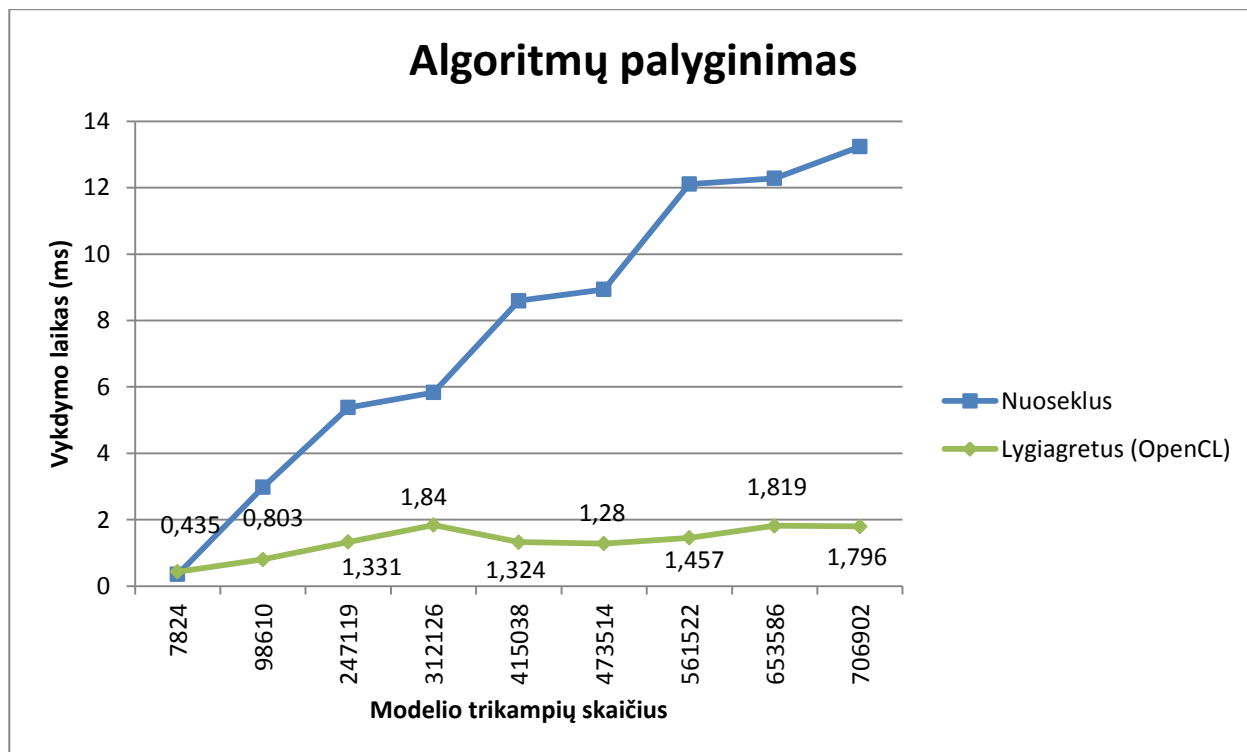
26 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (Java8 Parallel Stream)

Iš diagramos matyti, kad naudojant Java 8 lygiagrečius srautus, algoritmo vykdymo laikas gerokai padidėjo. Taip galėjo atsitikti dėl to, kad pats algoritmas nėra sudėtingas ir jo vieną iteraciją yra vykdoma gana greitai, tačiau jį reikia vykdyti daug kartų. Naudojant Java 8 lygiagrečius srautus, yra vykdomas papildomas darbas: gijų išskyrimas, rezultatų sujungimas. Jei šie darbai yra vykdomi ilgiau nei pats algoritmas, tuomet gaunami tokie rezultatai, kaip šio tyrimo atveju.

Taigi, šis lygiagretinimo būdas nedavė tokių rezultatų, kurių buvo tikimasi. Algoritmo vykdymo laikas buvo net iki 8 kartų ilgesnis, o sudėtingėjant modeliui, vykdymo laiku skirtumas tik didėtų.

5.3.2. Nuoseklaus algoritmo palyginimas su OpenCL lygiagrečiu algoritmu

Nuoseklaus algoritmo ir lygiagretaus algoritmo, naudojant OpenCL, palyginimas atvaizduotas diagramoje 27 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (OpenCL).



27 pav. Algoritmų palyginimas: Nuoseklus prieš Lygiagretus (OpenCL)

Iš diagramos matyti, kad algoritmo pritaikymas naudojimui su OpenCL davė svarių teigiamų rezultatų. Naudojantis nuosekliu algoritmu, didėjant modelio sudėtingumui, vykdymo laikas gerokai sparčiau kilo nei naudojantis lygiagrečiu algoritmu realizuotu OpenCL. Kuomet modelio trikampių skaičius siekė 706902, lygiagretus algoritmas buvo apie 7 kartus spartesnis nei nuoseklus algoritmas.

Lygiagretaus algoritmo vykdymo laikas taip pat yra tiesiškai priklausomas nuo modelio trikampių skaičiaus. Tuo remiantis, galima apskaičiuoti, kiek modelis gali turėti trikampių, kad vykdymo laikas neviršytų 16,6ms.

$$\text{Trikampių skaičius} = \frac{16,6\text{ms}}{\left(\frac{1,796\text{ms} - 0,435\text{ms}}{706902 - 7824}\right)} = \sim 8526594$$

Pagal gautą rezultatą galima teigti, kad šis algoritmas yra pakankamai spartus atlikti sudėtingu trimačių automobilių modelių pjaustymą.

6. IŠVADOS

1. Renkantis pjovimo algoritmą realizacijai kuriamoje sistemoje, buvo atsižvelgta į egzistuojančių algoritmų analizę. Kai kurie algoritmai veikė greitai, tačiau neturėjo galimybės padalinti modelį naudojantis pjaunančiąją plokštumą. Dėl to, buvo nuspręsta naudotis „*Basic Triangle Mesh Slicing*“ algoritmu.
2. Pasirinktas algoritmas leido padalinti modelį naudojantis pjaunančiąją plokštumą, tačiau atlikus jo greಿತaveikos matavimus buvo pastebėta, kad jis nėra pakankamai greitas norint jį panaudoti sudėtingesnės geometrijos pjovimui. Dėl to, buvo atliekamas šio algoritmo lygiagretinimo tyrimas.
3. Tiriant pjovimo algoritmo lygiagretinimo galimybes naudojantis Java 8 lygiagrečiais srautais, buvo pastebėta, kad toks lygiagretinimas smarkiai sulėtino pjovimo algoritmo veikimą. Atliekant pjovimą naudojantis taip lygiagretintu algoritmu, vykdymo laikas buvo iki aštuonis kartus ilgesnis nei atliekant pjovimą naudojantis nuosekliu algoritmu.
4. Ištyrus pjovimo algoritmo lygiagretinimo galimybes naudojantis OpenCL programavimo kalba, buvo pastebėtas žymus algoritmo vykdymo laiko sumažėjimas lyginant su nuosekliai vykdomu algoritmu. Lygiagrečiai atliekamo algoritmo vykdymo laikas sumažėjo apie septynis kartus sudėtingiausio turėto modelio pjovimo atveju. Be to, tokiu būdu paspartintas algoritmas galėtų pakankamai greitai apdoroti ir iki dvylikos kartų daugiau geometrijos turinčius modelius.

7. LITERATŪRA

- [1] ArchiCAD 17 brochure [žiūrēta 2013-12-09], prieiga internete: http://download.graphisoft.com/ftp/marketing/ac17/pdf/ArchiCAD17_Brochure.pdf
- [2] Graphisoft Archicad FAQ [žiūrēta 2013-12-09], prieiga internete: http://www.graphisoft.com/archicad/archicad_17/faq/
- [3] Revit for building design and construction [žiūrēta 2013-12-09], prieiga internete: <http://www.autodesk.com/products/autodesk-revit-family/overview>
- [4] Alias – Industrial design and class-A surfacing software [žiūrēta 2013-12-09], prieiga internete: <http://www.autodesk.com/products/autodesk-alias-products/overview>
- [5] Juyong Zhang, Jianmin Zheng, and Jianfei Cai, Senior Member, IEEE, Interactive Mesh Cutting Using Constrained Random Walks, Nanyang Technological University, Singapore, May 2011
- [6] Yunjin Lee, Seungyong Lee, Ariel Shamir, Daniel Cohen-Or, Hans-Peter Seidel, Intelligent Mesh Scissoring Using 3D Snakes, '04 Proceedings of the Computer Graphics and Applications, 12th Pacific Conference, 2004
- [7] M. Vatani, A. R. Rahimi, F. Brazandeh, A. Sanati, An Enhanced Slicing Algorithm Using Nearest Distance Analysis for Layer Manufacturing, World Academy of Science, Engineering and Technology, 2009
- [8] Stereolithography (3D printing) Algorithms and Thoughts [žiūrēta 2013-11-26], prieiga internete: <http://ravehgonen.wordpress.com/tag/plane-triangle-intersection/>
- [9] Intersections of Rays and Triangles [žiūrēta 2013-11-26], prieiga internete: http://geomalgorithms.com/a06-_intersect-2.html
- [10] Dot product [žiūrēta 2013-12-10], prieiga internete: <http://www.mathsisfun.com/algebra/vectors-dot-product.html>

8. TERMINŲ IR SANTRUMPŲ ŽODYNAS

UML (angl. *unified modeling language*) – vieninga modeliavimo kalba.

OpenGL (angl. *Open Graphics Library*) – atvira grafikos biblioteka.

JOGL (angl. *Java OpenGL*) – Java biblioteka, sudaranti galimybę naudoti OpenGL Java programavimo kalboje.

Primitive – paprasčiausias sistemos apdorojamas objektas.

OpenCL (angl. *Open Computing Language*) – programavimo kalba, skirta rašyti programas, kurios bus vykdomos skirtinguose procesoriuose.