



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

Laurynas Bačys

MULTIMEDIJOS TURINIO PERDAVIMO INTERNETU
EFEKTYVUMO TYRIMAS

Baigiamasis magistro projektas

Vadovas

Lekt. dr. Kęstutis Jankauskas

KAUNAS, 2017

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

Multimedijos turinio perdavimo internetu efektyvumo tyrimas

Baigiamasis magistro projektas
Informatika (kodas 621I10003)

Vadovas

(parašas) Lekt. dr. Kęstutis Jankauskas
(data)

Recenzentas

(parašas) Doc. dr. Tomas Blažauskas
(data)

Projektą atliko

(parašas) Laurynas Bačys
(data)

KAUNAS, 2017



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Informatikos fakultetas

(Fakultetas)

Laurynas Bačys

(Studento vardas, pavardė)

Informatika (kodas 621I10003)

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Pavadinimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 ____ m. _____ d.
Kaunas

Patvirtinu, kad mano, **Lauryno Bačio**, baigiamasis projektas tema „Multimedijos turinio perdavimo internetu efektyvumo tyrimas“ yra parašytas visiškai savarankiškai ir 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 išaiškėjęs nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Bačys Laurynas. Multimedijos turinio perdavimo internetu efektyvumo tyrimas. Magistro darbo baigiamasis projektas / vadovas lekt. dr. Kęstutis Jankauskas; Kauno technologijos universitetas, Informatikos fakultetas.

Mokslo kryptis ir sritis: Fiziniai mokslai, informatika

Reikšminiai žodžiai: *interaktyvios interneto programos, duomenų perdavimo efektyvumas, 3D glaudinimas, paskirstymas tinkle*

Kaunas 2017. 36 p.

SANTRAUKA

3D turinio atvaizdavimo sistema yra sukurtas šablonas 3D turinio atvaizdavimo internete projektams kurti, tačiau sistema turi trūkumų, kurie stabdo tolesnį šio produkto pritaikymą. Sistema yra sukurta programinės įrangos, interaktyvių technologijų ir grafikos produktų vystymo įmonės UAB „Indeform“. Sistema turi du leidimus, pirmasis pritaikytas kompiuteriams, antrasis leidimas yra minimizuotas ir pritaikytas planšetiniams kompiuteriams. Pagrindinė iškilusi problema – ilgas multimedijos duomenų persiuntimo laikas į naudotojo kompiuterį. Ši sistema nėra pirma tokio pobūdžio produktams kurti, tačiau platus metodų ir algoritmų spektras trukdo pasirinkti efektyviausią sprendimą šiai problemai spręsti.

Šiame darbe atlikta 3D turinio atvaizdavimo sistemos analizė, kurios metu apibrėžti sistemos trūkumai. Atlikta multimedijos duomenų perdavimo ir duomenų glaudinimo analizė, parinktas 3D turinio glaudinimo algoritmas, pasiūlytas multimedijos duomenų perdavimo sprendimas. Sukomplektuotas sprendimas pritaikytas ir ištestuotas su šešiais skirtingais duomenų rinkiniais. Realizuotos sistemos rezultatai parodė, kad pritaikymo sprendimo sistemos įkrovimas naudojant stipresnę aparatinę įrangą sumažėjo beveik 26%, o minimizuotoje sistemoje sumažėjo net 40%, testavimas silpnesnės aparatinės įrangos kompiuteriu parodė, kad neminimizuotos sistemos įkrovimas sumažėjo 60%, o minimizuotos 37,2%.

Bačys Laurynas. Research of Multimedia Content transfer Efficiency Over Internet: Master's thesis / supervisor lekt. dr. Kęstutis Jankauskas. The Faculty of Informatics, Kaunas University of Technology.

Research area and field: Physical sciences, informatics

Key words: interactive internet application, data transfer efficiency, 3D compression, data transfer management

Kaunas 2017. 36 p.

SUMMARY

Interactive 3D content visualization system is a template for 3D visualization project development, but this system has flaws and it stops further usage of this system. This system is developed by software, interactive technologies and visual graphics development company „Indeform“ Ltd. This system has two editions, first one is designed to work with personal and desktop computers, second is designed to work with tablets. Main problem of this system is long data transmission and loading times. This system is not the first of this kind, but wide range of methods and algorithms disturbs discovery of most effective solution for this problem.

In this work we analyze 3D content visualization system, transmission and loading related issues. We analyze transmission and 3D content compression methods of multimedia content. Analyzed methods are gathered and offered as possible solution. Solution is adapted for old system and tested with six different contents on two different computers. Final result of this work confirmed that offered solution is more effective, than original system solution, results on laptop showed what non-minimized system loading times reduced by 26% and minimized system by 40%. Results on desktop computer shows that non-minimized system loading times reduced by 60% and minimized system by 37,2%.

TURINYS

ĮVADAS	10
Tikslas ir uždaviniai	10
Dokumento struktūra.....	10
1 3D TURINIO DUOMENŲ GLAUDINIMO IR MULTIMEDIJOS DUOMENŲ PERDAVIMO INTERNETO TINKLU METODŲ ANALIZĖ	11
1.1 Duomenų perdavimo internetu analizė	11
1.2 Multimedijos duomenų glaudinimas.....	13
1.2.1 3D daugiakampių tinklo glaudinimo algoritmai.....	14
1.2.2 Progresuojančio daugiakampių tinklo algoritmai.....	17
1.2.3 3D turinio glaudinimo algoritmai	18
1.3 Paveikslų glaudinimo analizė	20
1.4 Interaktyvios interneto aplikacijos sistemos analizė prieš atnaujinimą	20
2 MULTIMEDIJOS TURINIO PERDAVIMO INTERNETU METODŲ IR TINKLO ARCHITEKTŪROS PROJEKTAVIMAS.....	23
2.1 Sprendimo reikalavimai	23
2.2 Atnaujintos interaktyvios interneto aplikacijos sistemos sprendimas	23
3 MULTIMEDIJOS TURINIO GLAUDINIMO IR PERDAVIMO INTERNETU METODŲ REALIZACIJA IR EKSPERIMENTŲ REZULTATAI.....	26
3.1 3D turinio glaudinimo realizacija, eksperimentinis tyrimas ir analizė	26
3.2 Duomenų perdavimo internetu metodų realizacija ir eksperimentinė analizė.....	28
4 IŠVADOS.....	35
Literatūra.....	36

PAVEIKSLŲ SĄRAŠAS

1.1 pav. Duomenų perdavimo modeliai: (A) Klientas/serveris modelis; (B) P2P modelis.....	12
1.2 pav. Sinchroninio duomenų perdavimo schema.....	13
1.3 pav. Asinchroninio duomenų perdavimo schema	13
1.4 pav. Normalių žemėlapis ir jo pritaikymas [8]: (A) Modelio paviršius pritaikius normalių žemėlapi; (B) Normalių žemėlapis.....	14
1.5 pav. 3D daugiakampių tinklo supaprastinimas [9]: (A) Modelis su tankiu daugiakampių tinklu; (B) Supaprastintas modelio daugiakampių tinklas; (C) Supaprastintas modelio daugiakampių tinklas, panaudojus normalių žemėlapius	14
1.6 pav. 3D modelio daugiakampių tinklo elementai [10]: (A) Viršūnės; (B) Briaunos; (C) Paviršiai; (D) Objektai	15
1.7 pav. Dvejetainio failo turinio pavyzdys.....	15
1.8 pav. ASCII tipo fbx failo turinio pavyzdys	16
1.9 pav. Wavefront obj formato turinio pavyzdys.....	16
1.10 pav. Aštuntainio medžio progresyvaus daugiakampių tinklo atvaizdavimas [13]: (A) suglaudinto modelio pradinis atvaizdavimas, (B) atvaizdavimas įpusėjus modelio įkrovimą, (C) pilnas modelio atvaizdavimas.....	18
1.11 pav. 600×400px atvaizdai pritaikius skirtingą glaudinimą: (A) neglaudintas atvaizdas 4,55MB; (B) maksimaliai suglaudintas png atvaizdas 1,03MB; (C) maksimaliai suglaudintas jpeg atvaizdas naudojant nuostolingą metodą 49,8KB.....	20
1.12 pav. Duomenų apdorojimo schema	21
1.13 pav. Duomenų priėmimo ir įkrovimo į sistemą schema kliento dalyje.....	21
1.14 pav. 3D turinio atvaizdavimo sistema, pritaikyta kauno pilies apgulties projektui.....	22
2.1 pav. Atnaujinta duomenų apdorojimo schema serverio dalyje	24
2.2 pav. 3D daugiakampių tinklo ir jų parametrų eksportavimo schema.....	24
2.3 pav. Atnaujinta duomenų apdorojimo schema kliento dalyje	25
3.1 pav. 3D daugiakampių tinklo ir jų parametrų eksportavimo pasirinkimai Blender aplinkoje	26
3.2 pav. OpenCTM 3D turinio peržiūrėjimo ir eksportavimo įrankis.....	27

LENTELIŲ SĄRAŠAS

1.1 lentelė Progresuojančių daugiakampių tinklo kodavimo algoritmų palyginimas [10].....	17
1.2 lentelė Testuotų modelių dydis KB Glaudinimas atliekamas naudojant i7 CPU, 3.4 GHz.. Parinkti formatai turi integruotą papildomą GZIP glaudinimą perdavimo tinklu metu. Atvaizdai ir tekstūros neįtrauktos į skaičiavimus [12]	18
3.1 lentelė 3D turinio su objektais, kurių 3D daugiakampių tinklas turi 162684 viršūnės, glaudinimo rezultatai naudojant CTM formatą.....	27
3.2 lentelė Duomenų glaudinimo rezultatai testuojant su 1934676 viršūnės turinčiu 3D turiniu neskaidžius turinio, suskaidžius į penkias ir devynias dalis	28
3.3 lentelė Atvaizdų perdavimo laikai neatsižvelgiant į siuntimo laiko persidengimą, pateikti rezultatai yra trijų matavimų vidurkis.....	29
3.4 lentelė Minimizuoto turinio sistemos atvaizdų perdavimo laikai, neatsižvelgiant į siuntimo laiko persidengimą, pateikti rezultatai yra trijų matavimų vidurkis	31
3.5 lentelė 3D turinio perdavimo ir išskleidimo laikas perduodant neskaidytą ir skaidytą turinį skirtingų išteklių įrenginiuose, pateikti rezultatai yra trijų matavimų vidurkis	32
3.6 lentelė Duomenų perdavimo, išpakavimo ir įkrovimo į sistemą trukmės duomenys, naudojant 1934676 viršūnės turintį 3D turinį, pateikti rezultatai yra trijų matavimų vidurkis	33
3.7 lentelė 3D turinio atvaizdavimo sistemos turinio įkrovimo laiko palyginimas pradinėje ir galutinėje sistemos versijoje testuojant su i5-430M, 2.26 GHz, 4GB RAM, kai tinklo duomenų perdavimo sparta yra iki 60 Mbps, pateikti rezultatai yra trijų matavimų vidurkis	33
3.8 lentelė. 3D turinio atvaizdavimo sistemos turinio įkrovimo laiko palyginimas pradinėje ir galutinėje sistemos versijoje testuojant su i7-3930K, 3.20 GHz, 12GB RAM ,kai tinklo duomenų perdavimo sparta yra iki 94 Mbps, pateikti rezultatai yra trijų matavimų vidurkis	34

Terminų ir santrumpų žodynas

Terminas	(<i>angl. angliškas variantas</i>) termino paaiškinimas.
Duomenų paketo perdavimo laikas	(<i>angl. end-to-end transmission delay</i>) laikas, reikalingas perduoti vieną duomenų paketą iki vartotojo.
Duomenų perdavimo sparta	(<i>angl. bandwidth</i>) duomenų kiekis, kuris gali būti perduotas iš vieno taško į kitą per laiko vienetą (dažniausiai sekundę).
P2P	(<i>angl. peer-to-peer</i>) decentralizuotas duomenų komunikacijos modelis, kurio kiekvienas tinklo mazgas gali atlikti tiek kliento, tiek serverio rolę.
Klientas/serveris	(<i>angl. client/server</i>) duomenų komunikacijos modelis, kuriame procesai priklauso tik serveriui arba tik klientui.
Reljefo atvaizdas ASCII	(<i>angl. bump map</i>) paviršiaus šiurkštumo imitavimo technika 3D grafikoje (<i>angl. American Standard Code for Information Interchange</i>) simbolių koduotės standartas, naudojamas aprašyti informacijai, kuri bus perkeliama į kitą programinę įrangą tolimesniam naudojimui.
Dvejetainis failas	(<i>angl. Binary file</i>) failas, kurio turinys tam, kad būtų suprastas, turi būti interpretuotas programos arba aparatinės įrangos procesoriaus.
FBX	(<i>angl. Filmbox</i>) failų formatas, skirtas dalintis skaitmeniniu turiniu tarp programų.
bpv	(<i>angl. bits per vertex</i>) 3D daugiakampių tinklo suspaudimo santykis, kuris nuroto, kiek bitų skiriama vienai koduotai viršūnei.
Daugialypė geometrija	(<i>angl. non-manifold geometry</i>) 3D tinklelio struktūra, kai tinklo viršūnės ar kraštinės yra atskirtos nuo paties tinklo, egzistuoja paviršiai, esantys tinklelio viduje arba daugiakampių tinklas turi bent vieną sritį, kuri neturi storio.
LZMA	(<i>angl. Lempel-Ziv-Makov chain algorithm</i>) nenuostolingas domenų glaudinimo algoritmas.

IVADAS

Pasiekta pažanga technologijų srityje suteikia naujas galimybes vystant interneto technologijas. Technologijos vystosi taip greitai, kad net nespėjame suvokti, kokius didelius žingsnius darome tobulindami interneto galimybes. Nedaug kas prisimena pirmąjį kompiuterių tinklą, kurio greitaveika siekė vos 50 Kbps. Šiuolaikinis internetas yra pažengęs toli nuo pirmųjų interneto tinklų. Tobulėjant interneto technologijoms kartu vystėsi ir svetainių kūrimo tendencijos, o su jomis ir vartotojų poreikiai. Siekiant patenkinti besikeičiančius vartotojų poreikius bei suteikti naujas galimybes, vis labiau populiarėja interaktyvūs svetainių turinys. Interaktyviose svetainėse gali būti talpinamas įvairus multimedijos turinys, kurį sudaro atvaizdai, 3D modeliai, animacijos ir vaizdo bei garso įrašai. Tobulėjant interneto technologijoms didėja vartotojų poreikis matyti aukštos kokybės multimedijos turinį, tačiau tai lemia didelį duomenų kiekį, kurį reikia perduoti į vartotojo kompiuterį. Naudojamas turinys prieš persiunčiant jį į vartotojo yra glaudinamas, kad siunčiamų duomenų dydis būtų kuo mažesnis, o kokybė išliktų aukščiausia. Tačiau duomenų glaudinimas ne visada yra pakankamas sprendimas greitam duomenų perdavimui, o ilgas duomenų persiuntimas ir įkrovimas suteikia diskomfortą vartotojui.

Siekiant išvengti ilgo įkrovimo problemos, reikia pritaikyti sprendimus, kurie pagreitintų duomenų perdavimą iki vartotojo bei pagreitintų įkrovimą. Šiame darbe yra nagrinėjami metodai, kurie potencialiai gali pagreitinti multimedijos duomenų perdavimą internetu bei sutrumpinti 3D turinio atvaizdavimo sistemos įkrovimo laiką.

Tikslas ir uždaviniai

Šio tiriamojo darbo tikslas yra išanalizuoti 3D turinio atvaizdavimo sistemos multimedijos duomenų glaudinimo ir perdavimo efektyvumą ir pasiūlyti efektyvesnę sistemos modelį duomenis perduoti. Toliau pateikti tyrimo uždaviniai:

1. Išanalizuoti 3D turinio atvaizdavimo sistemos trūkumus bei galimus sprendimus šiems trūkumams spręsti;
2. Suprojektuoti efektyvų 3D turinio glaudinimo ir multimedijos turinio perdavimo internetu modelį;
3. Realizuoti ir ištestuoti suprojektuotą 3D turinio atvaizdavimo sistemos atnaujintą modelį.

Dokumento struktūra

1 Skyrius. Apžvalginė dalis. Šiame skyriuje atliekama duomenų perdavimo metodų apžvalga, siekiant išanalizuoti esamų metodų trūkumus ir privalumus, kuriuos vėliau galima būtų panaudoti projektuojant duomenų perdavimo modelį. Taip pat atliekama esamų metodų atvaizdams ir 3D turiniui glaudinti analizė ir palyginimas.

2 Skyrius. Sprendimo dalis. Šiame skyriuje projektuojamas sprendimas 3D turinio atvaizdavimo sistemos duomenų perdavimo efektyvumui padidinti. Pateikiamas siūlomas sprendimas duomenų perdavimui ir efektyvesni metodai duomenų glaudinimui.

3 Skyrius. Eksperimentinė dalis. Skyriuje pateikiami sistemos realizacijos žingsniai. Apžvelgiami atnaujintos ir neatnaujintos sistemos duomenų perdavimo rezultatai bei jų priežastys.

1 3D TURINIO DUOMENŲ GLAUDINIMO IR MULTIMEDIJOS DUOMENŲ PERDAVIMO INTERNETO TINKLU METODŲ ANALIZĖ

Pirmojo tinklapio idėja buvo sugalvota 1990 metais. Siekiant peržiūrėti dokumentą būdavo būtina prisijungti prie kompiuterio, kuriame saugomas ieškomas dokumentas, tačiau tai nepatogu, kai dokumentai gali būti saugomi skirtingose vietose esančiuose kompiuteriuose su skirtingomis programinėmis įrangomis. Tuo laikotarpiu internetas buvo tik grupė statinių dokumentų, kuriuos daugiausiai naudojo gynybos organizacijos ir akademinės institucijos. Britų informatikas Tim Berners-Lee [1] ieškodamas sprendimo šiai problemai sukūrė pirmąjį tinklapį [1]. Tinklapiai sukurti dokumentų atvaizdavimui ir peržiūrai leisdavo pasiekti ieškomus dokumentus nekreipiant dėmesio į naudojamą programinę įrangą ar dokumento formatą.

Vartotojų poreikiai nuo pirmojo tinklapio sukūrimo labai pasikeitė. Internetas tapo skirtas ne tik tekstinės informacijos, bet ir kito multimedijos turinio (vaizdo ir garso įrašų, 3D turinio, paveikslų) dalinimuisi. Neapdorotas multimedijos turinys užima didelę dalį atmintinės, todėl būtina taikyti sprendimus, kurie sumažintų jų dydį, taip sutrumpinant perdavimo internetu laiką. Vaizdo ir garso įrašai bei paveikslai yra visiškai apdorojami naudojant šio turinio redagavimo programas, todėl tolesnis šio turinio apdorojimas neduoda geresnių rezultatų. Kitaip nei paveikslai, vaizdo ar garso įrašai, 3D modeliavimo programose, gaunamas rezultatas pateikiamas nekoduotu formatu ir priklausomai nuo scenos sudėtingumo gali užimti didelę atmintinės dalį. Siekiant sumažinti 3D turinio apimtį, būtina atskirai jį apdoroti, taikant rankinius 3D daugiakampių tinklo paprastinimo metodus ir 3D turinio glaudinimo algoritmus.

UAB „Indeform“ sukurta 3D turinio atvaizdavimo sistema yra šablonas, skirtas pritaikyti 3D turinio atvaizdavimo projektams. Sistemos paskirtis atvaizduoti multimedijos turinį tinklapyje. Dabartinis sistemos sprendimas leidžia atvaizduoti 3D turinį su tekstūromis, objektų animacijomis, šviesomis bei 3D kameros perspektyvinės transformacijos reikšmės, pagal kurias sukuriama kameros perspektyvos imitacija. Sistema yra tobulinama ir pagrindinis sistemos trūkumas yra per ilgas duomenų įkrovimo laikas, kuris lemia diskomfortą vartotojams, besinaudojantiems šia sistema. Ilgas tinklapių įkrovimo laikas lemia sumažėjusį vartotojų kiekį, net pusę vartotojų, susidūrusių su efektyvumo problemomis, nutraukdavo naudojimąsi tinklapiu [2], todėl būtina pritaikyti sprendimus, kurie sumažintų duomenų persiuntimo laiką.

Kuriant interaktyvias interneto programas galima apibrėžti keturis pagrindinius parametrus, kurie lemia greitą duomenų perdavimą iki vartotojo:

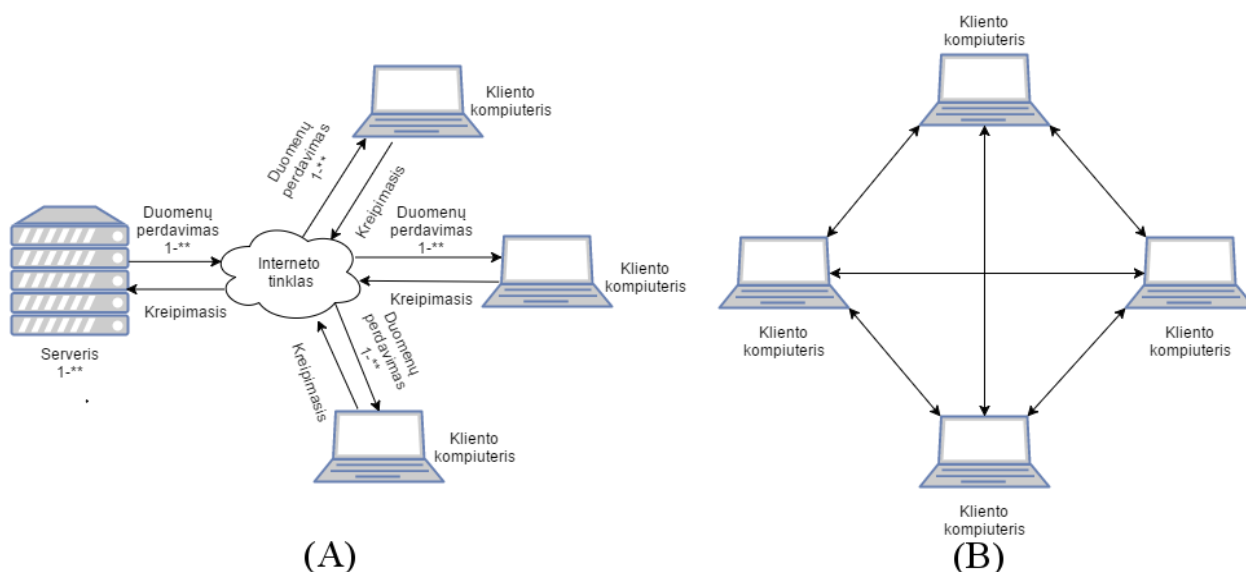
- Duomenų perdavimo sparta;
- Duomenų paketų dydis ir kiekis;
- Duomenų paketo išskleidimo laikas, jei paketas yra koduotas;
- Tinklo apkrova.

1.1 Duomenų perdavimo internetu analizė

Duomenų perdavimo efektyvumas yra neatsiejamas nuo duomenų perdavimo spartos. Šio parametro reikšmė priklauso tik nuo vartotojo interneto greitaveikos pasirinkimo ir interneto tiekėjo galimybių, todėl plėtotojas, įvertinęs vartotojų galimybes, turi realizuoti efektyviausią sprendimą ir visiškai išnaudoti turimas galimybes nekeičiant tinklo duomenų perdavimo spartos parametru. Siekiant išnaudoti tinklą pilnu pajėgumu yra būtina tinkamai paskirstyti duomenis tinkle. Duomenų persiuntimas lemia tinklo apkrovą, kuri gali atsirasti ne tik dėl plėtotėjo sistemos perduodamų duomenų, bet ir vartotojo atliekamų veiksmų tinkle.

Galime išskirti du pagrindinius duomenų perdavimo modelius [3] (1.1 pav.):

- Klientas/serveris modelis;
- P2P modelis.



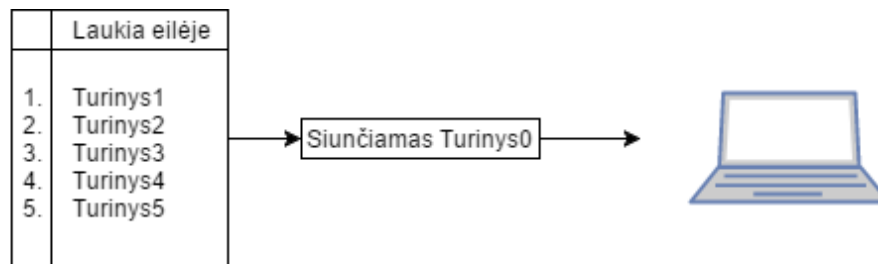
1.1 pav. Duomenų perdavimo modeliai: (A) Klientas/serveris modelis; (B) P2P modelis

P2P modelis (1.1 pav. B) yra dažnai naudojamas tarp vartotojų dalinantis turiniu. Šio modelio išskirtinumas yra tas, kad kiekvienas mazgas tinkle gali būti naudojamas kaip serveris arba klientas. Naudojant šią tinklo topologiją kiekvienas vartotojas gali kontroliuoti kitų mazgų prisijungimų kiekį, ieškoti kitų mazgų, pasirinkti, prie kurių sistemų jungtis ir kurių vengti. P2P modelis suteikia galimybę kontroliuoti tinklo išteklius sumažinant arba padidinant tinklo apkrovą. Naudojant P2P modelį failas nėra siunčiamas iš vieno mazgo, vietoje to failo mažos dalys siunčiamos iš tūkstančių vartotojų, kurie talpina siunčiamą turinį. Šio modelio trūkumas yra trumpalaikis duomenų išlaikymas. Vartotojai ištrindami turinį iš savo kompiuterių sumažina galimų prisijungimų kiekį, todėl kiti mazgai nebegali gauti duomenų. Dėl trumpalaikio duomenų pasiekiamumo P2P modelis nėra tinkamas interaktyvioms interneto programoms. Šis duomenų perdavimo modelis dažniausiai siejamas su turinio piratavimu ir autorių teisių pažeidimu. P2P modelio programos naudojamos apeiti ugniasienes ir platinti kenkimo programas, todėl sistemų administratoriai dažnai ragina nesunaudoti šiuo modeliu paremtomis programomis [4].

Klientas/serveris modelis (1.1 pav. A) yra dažniausiai naudojamas modelis kuriant internetines programas. Priešingai nei P2P modelis, klientas/serveris modelio mazgai gali atlikti tik vieną funkciją – kliento arba serverio. Šio modelio atveju kliento kompiuterio naršyklė kreipiasi į serverį siųsdama užklausa duomenims gauti. Serveris, gavęs užklausa, siunčia duomenis į kliento kompiuterį [5]. Klientas/serveris modelio atveju duomenys yra pasiekiami tol, kol plėtotojas nenusprendžia kitaip, todėl nekyla grėsmė, kad nesinaudojant programa turinys bus nepasiekiamas kaip P2P modelio atveju. Klientas/serveris modelio duomenų perdavimas gali funkcionuoti keliais skirtingais metodais:

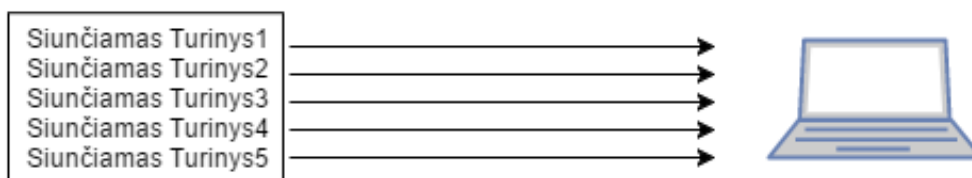
- Synchroninis duomenų perdavimas;
- Asinchroninis duomenų perdavimas.

Sinchroninio duomenų perdavimo atveju (1.2 pav.) failai yra siunčiami paeiliui vienas po kito ir tik baigus persiųsti failą yra pradedamas siųsti kitas failas. Šis duomenų perdavimo modelis sukoncentruoja turimus aparatinės įrangos ir tinklo išteklius į vieną siunčiamą failą. Tai yra efektyvu, kai duomenų paketai yra pakankamai dideli apkrauti naudojamą aparatinę įrangą ir išnaudoti visą tinklo pajėgumą, jei turinys susideda iš mažo dydžio, bet didelio kiekio paketų, tai yra neefektyvu, nes bus išnaudojama daugiau laiko siunčiant užklausa į serverį.



1.2 pav. Sinchroninio duomenų perdavimo schema

Asinchroninio duomenų perdavimo atveju (*1.3 pav.*) failai yra siunčiami nepriklausomai nuo to, ar prieš tai siunčiami failai buvo baigti perduoti, ar ne. Šis modelis yra efektyviausias, kai siunčiamas turinys susideda iš nedidelio dydžio duomenų paketų, tokioje situacijoje vieno duomenų paketo siuntimas neišnaudoja visų turimų išteklių, todėl lygiagrečiai siunčiamas kelių paketų srautas leidžia išnaudoti visą turimų išteklių potencialą. Asinchroninis duomenų perdavimas yra neveiksmingas tuo atveju, kai duomenų paketai yra pakankamai dideli, kad apkrautų tinklą didesniam nei 1000 ms laiko tarpui, tuomet kelių duomenų paketų siuntimas užtrunka. Šiuo atveju tinklo duomenų perdavimo sparta visiškai išnaudojama vieno siunčiamo paketo ir trūkstantis išteklių kiekis neleidžia perduoti likusių duomenų, o išteklių eilėje laukiantys siuntimai arba siunčiami labai lėtai, arba tampa neveiksnūs ir neefektyviai apkrauna sistemą.



1.3 pav. Asinchroninio duomenų perdavimo schema

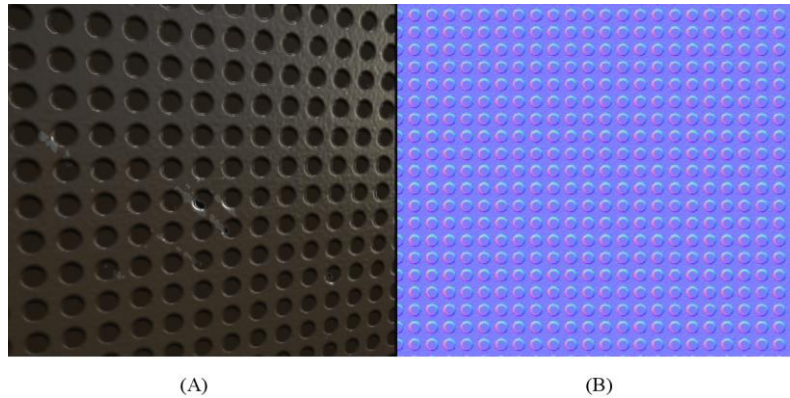
1.2 Multimedijos duomenų glaudinimas

Duomenų paketo dydis priklauso nuo sistemos kūrėjo perduodamo turinio dydžio ir paketų kiekio, tačiau esant poreikiui paketų kiekis ir dydis yra koreguojamas, didinant paketų skaičių ir mažinant kiekvieno paketo dydį arba mažinant paketų skaičių ir didinant kiekvieno paketo dydį. Perduodant didelį kiekį failų, kliento kompiuteris siunčia tiek užklausų į serverį, kiek reikalinga gauti failų, o tai yra efektyvu tik tais atvejais, kai failų persiuntimas ir išpakavimas užima daugiau laiko, nei failų perdavimas nesupakavus. Duomenų išskleidimo laikas priklauso nuo naudojamo suspaudimo algoritmo sudėtingumo, algoritmo realizacijos efektyvumo, aparatinės įrangos greičio bei glaudinamų duomenų kiekio [6]. Multimedijos duomenų glaudinimas interneto programoms yra būtinas dėl didelės nekoduotų duomenų apimties. Sumažinus multimedijos duomenų dydį galima greičiau juos perduoti į vartotojo kompiuterį.

Plečiantis interneto technologijų galimybėms vis labiau siekiama interaktyviose interneto programose naudoti didesnės kokybės grafinius elementus. Dėl platesnių realizavimo galimybių populiarėja 3D grafikos naudojimas interaktyviose programose. Turinio kokybė ir 3D objektų daugiakampių tinklo viršūnių skaičius yra labai svarbus – didelis viršūnių kiekis didina reikalingos atminties reikalavimus. Didelis duomenų kiekis lemia lėtesnį interaktyvios interneto programos darbą, o tai, savo ruožtu, sukelia nepatogumų vartotojui. Lėtas sistemos darbas gali sumažinti programos vartojimą, o tai yra nepalanku tokių sistemų plėtojui. Siekiant paspartinti interaktyvių internetinių programų paleidimo laiką yra atlikti tyrimai, siūlantys duomenis glaudinti ir taip sumažinti jų apimtį ir sutrumpinanti duomenų perdavimo iki vartotojo laiką. Yra išskiriami du duomenų glaudinimo tipai: nuostolingi (angl. *lossy*) ir nenuostolingi (angl. *lossless*). Nuostolingi metodai atrenka duomenis, kurių pašalinimas nesuteikia akivaizdžių originalaus failo pasikeitimų, tuo tarpu nenuostolingi veikia manipuliuojami duomenimis [7].

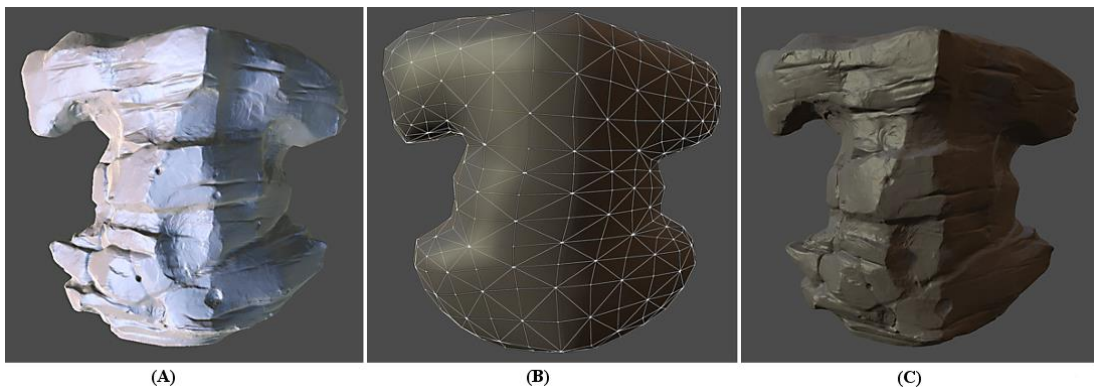
1.2.1 3D daugiakampių tinklo glaudinimo algoritmai

3D modelių naudojimas interaktyviose programose labai praplečia realizavimo galimybes, tačiau detalių modelių naudojimas didina reikalingų išteklių kiekį, padidėjus reikalingam resursų kiekiui, ilgėja ir duomenų perdavimo bei įkrovimo laikas į naudotojo kompiuterį. Siekiant sutaupyti išteklių kiekį ir sumažinti įkrovimo laiką, viršūnių skaičius yra sumažinamas iki minimalaus, išlaikant pagrindinę modelio formą (1.5 pav. B). 3D grafikoje kiekviena viršūnė ir paviršius, esantis daugiakampių tinkle, turi normalę, kuri yra vektorius, statmenas objekto paviršiui. Šis vektorius naudojamas skaičiuojant objekto reljefą, pritaikius normalių žemėlapi. Normalių žemėlapis yra 2D atvaizdas (1.4 pav. A), kuris saugo ribotą 3D informaciją. RGB spalvų kanalai atitinka X, Y ir Z paviršių normalių koordinates. [8]



1.4 pav. Normalių žemėlapis ir jo pritaikymas [8]: (A) Modelio paviršius pritaikius normalių žemėlapi; (B) Normalių žemėlapis

Normalių žemėlapiu saugoma informacija yra naudojama skaičiuojant apšvietimą ir taip sukuriama paviršiaus gylio imitacija (1.4 pav. A). Šis metodas nekeičia pačios daugiakampių tinklo geometrijos. Galutinis rezultatas turi tiek pat viršūnių kaip ir supaprastintas modelis, tačiau detalumas prilygsta detalų daugiakampių tinklą turinčiam modeliui (1.5 pav.). Toks modelių supaprastinimas yra atliekamas rankiniu būdu, naudojant 3D turinio apdorojimo programinę įrangą.



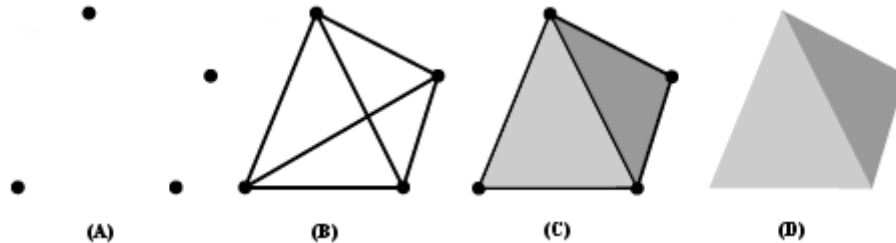
1.5 pav. 3D daugiakampių tinklo supaprastinimas [9]: (A) Modelis su tankiu daugiakampių tinklu; (B) Supaprastintas modelio daugiakampių tinklas; (C) Supaprastintas modelio daugiakampių tinklas, panaudojus normalių žemėlapius

Prieš perduodant 3D turinį internetu yra būtina jį suglaudinti. 3D daugiakampių tinklai yra skirstomi į dvi kategorijas – statinius ir dinامينius. Skirtingoms modelių kategorijoms taikomi skirtingi suspaudimo algoritmai. Statinių daugiakampių tinklų viršūnių skaičius nesikeičia kintant laikui. Tai gali būti kieti kūnai ar organiniai modeliai. Dinaminų daugiakampių tinklo paviršius nėra pastovus ir daugiakampių tinklas gali plėstis pridėdam naujas viršūnes. Dinaminiai daugiakampių tinklai dažniausiai atsiranda modeliuojant fizikinius reiškinius, pavyzdžiui, vandens modeliavimo metu daugiakampių tinklas gali būti dinamiškai praplečiamas suteikiant vandens tekėjimo efektą. Interaktyviose interneto programose naudojami modeliai yra statiniai.

3D modelių daugiakampių tinklas gali būti apibūdintas skirtingais hierarchiniais elementais (1.6 pav.). Viršūnės yra pagrindinis daugiakampių tinklo elementas. Briaunos yra segmentai, jungiantys dvi viršūnes, o paviršius sudaro uždara briaunų kelią. Objektai - tai uždaros ribos, suformuotos sujungtų paviršių.

Modelių daugiakampių tinklo informacija yra skirstoma į tris kategorijas [10]:

- Modelio geometrijos informacija yra nusakoma kiekvienos viršūnės pozicija modelio daugiakampių tinkle, trimatėje erdvėje.
- Jungiamumo informacija, dar kitaip vadinama topologija, kuri nusako kiekvieno daugiakampių tinklo elemento santykį su kitais elementais.
- Pasirinktini atributai, kurie nusako objekto savybes (spalva, normalių kryptis, UV koordinatės).



1.6 pav. 3D modelio daugiakampių tinklo elementai [10]: (A) Viršūnės; (B) Briaunos; (C) Paviršiai; (D) Objektai

Sukurtas 3D turinys 3D grafikos apdorojimo programose gali būti eksportuojamas įvairiais failų formatais, kurie skirtingai aprašo daugiakampių tinklo informaciją. 3D turinio informacija saugoma dviejų tipų formatuose:

- Dvejetainiuose failuose;
- Tekstiniuose failuose.

Dauguma formatų 3D turinio apdorojimo programose turi galimybę eksportuoti tuos pačius failus dvejetainiu formatu arba ASCII tipo formatu. ASCII tipo formatai yra saugomi žmogui suprantama kalba, todėl yra patogiu naudoti tokio tipo failus kuriant programas, kuriose reikia pasiekti ir panaudoti turinį. Tas pats formatas, saugomas dvejetainiuose ar ASCII failuose, saugo tą pačią informaciją, tačiau dėl skirtingo informacijos aprašymo failų dydis gali skirtis. Skirtingi formatai turi savo informacijos limitus ir gali reprezentuoti skirtingą kiekį 3D turinio informacijos, pavyzdžiui, wavefront .obj formatas neturi galimybės saugoti apšvietimo, perspektyvinės kameros bei kitų su 3D daugiakampių tinklu tiesiogiai nesusijusių parametrų.

Tam, kad suprastum dvejetainio failo turinį, turi būti interpretuotas programos arba aparatinės įrangos, kuri turi informacijos, kaip suformatuotas šis turinys. Tai nėra išoriškai atpažįstamas formatas, todėl programa ar aparatinė įranga nagrinėdama šį failą turi žinoti, kurioje tiksloje vietoje yra talpinama ieškoma informacija (1.7 pav.), todėl perduodant šį turinį joks įrenginys tinkle negali interpretuoti šio turinio [11].

Address	0	1	2	3	4
00000990	01101001	01100111	01101110	01010011	00000011
000009a0	01001001	01101110	01110100	01100101	01100111
000009b0	00000000	11110000	00001001	00000000	00000000
000009c0	00001110	00000000	00000000	00000000	01001111
000009e0	01101001	01110011	01010011	00000011	00000000

1.7 pav. Dvejetainio failo turinio pavyzdys

Dvejetainio formato failai užima mažesnę atminties kiekį, nes *ASCII* formato failai talpina daugiau skirtingų simbolių nei dvejetainiai (1.8 pav.). Dėl kompaktiško informacijos talpinimo dauguma atvaizdų, vaizdo ir garso įrašų yra saugomi dvejetainiuose formatuose. Kuriant programas, kuriose naudojamas individualus variklis, yra patogiau 3D turinį naudoti tekstiniuose nei dvejetainiuose formatuose. 3D turinys gali būti naudojamas dvejetainiu formatu, tačiau saugojimas tekstiniuose formatuose leidžia lengviau manipuluoti turiniu.

```
Model: "Model::Cube", "Mesh" {
Version: 232
Properties60: {
  Property: "QuaternionInterpolate", "bool", "", 0
  Property: "Visibility", "Visibility", "A+", 1
  Property: "Lcl Translation", "Lcl Translation",
"A+", 0.0000000000000000, 0.0000000000000000, 0.0000000000000000
  Property: "Lcl Rotation", "Lcl Rotation", "A+", -
90.000009334538021, 0.0000000000000000, 0.0000000000000000
  Property: "Lcl Scaling", "Lcl Scaling",
"A+", 1.0000000000000000, 1.0000000000000000, 1.0000000000000000
  Property: "RotationOffset", "Vector3D", "", 0, 0, 0
  Property: "RotationPivot", "Vector3D", "", 0, 0, 0
```

1.8 pav. *ASCII* tipo *fbx* failo turinio pavyzdys

Dauguma standartinių 3D turinio formatų talpina didelį kiekį informacijos, net jei ne visa turinio informacija yra naudojama, pavyzdžiui, *fbx* formatas gali saugoti apšvietimo, medžiagų bei kitus parametrus, kurie galutiniame produkte nėra naudojami. Efektyvus būdas yra saugoti 3D daugiakampių informaciją *wavefront obj* formatu, kuris talpina:

- Daugiakampių tinklo viršūnių, paviršių, kraštinių informaciją;
- Normalių ir UV koordinatijų reikšmes;
- Objektų grupių informaciją;
- NURBS kreives ir paviršius.

Šis formatas yra kompaktiškas, o failo struktūrą paprasta suprasti ir naudoti. Failo struktūra aprašoma konkrečia informacija, kuri nurodo atributo pavadinimą ir jo reikšmę (1.9 pav.). *Wavefront obj* yra standartizuotas formatas, todėl šį formatą palaiko dauguma įrankių.

```
v -120.049614 -12.059994 193.133820
v -124.494576 -12.059994 190.393097
v -120.002281 -12.277779 193.057068
v -124.447250 -12.277779 190.316345
v -119.887978 -12.367988 192.871674
v -124.332947 -12.367988 190.130936
v -119.773659 -12.277779 192.686310
v -124.218628 -12.277779 189.945572
```

1.9 pav. *Wavefront obj* formato turinio pavyzdys

Dvejetainiai ir tekstiniai failai yra neglaudinto formato, todėl naudojant 3D modelius su šiais formatais būtina suglaudinti duomenis. Glaudinimo algoritmas turi būti tinkamas glaudinti dvejetainę informaciją arba *ASCII* tipo failo formatus. Galime išskirti dviejų tipų glaudinimo algoritmus:

- Progresuojančio daugiakampių tinklo algoritmai;
- 3D turinio glaudinimo algoritmai.

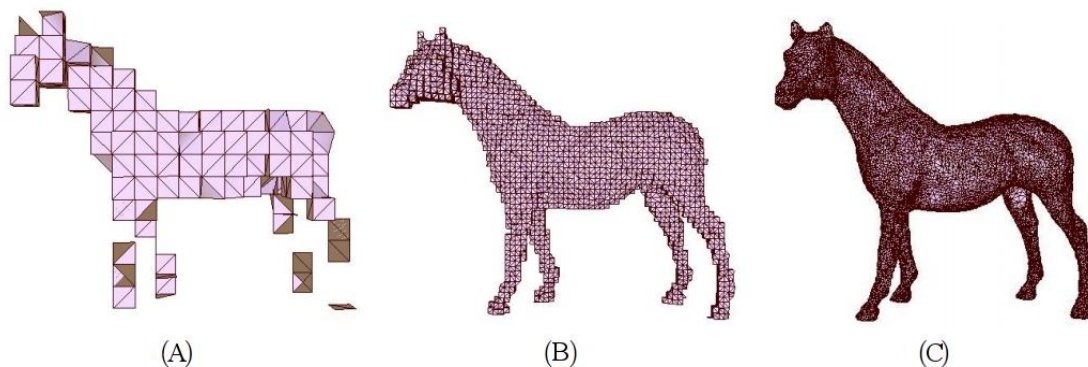
1.2.2 Progresuojančio daugiakampių tinklo algoritmai

Progresuojančio daugiakampių tinklo algoritmai yra naudingi interaktyvioms interneto programoms, kadangi prastos kokybės modelio daugiakampių tinklas įkraunamas vartotojui greičiau nei kitų algoritmų ir slenkant laikui vis labiau tobulinamas, kol galiausiai gaunamas galutinis aukštos kokybės daugiakampių tinklas. Pagrindinė šio algoritmo esmė yra kliento pusėje atkurti grubų daugiakampių tinklą ir laikui bėgant dalinti daugiakampių tinklą, kol pasiekiamas originalus aukštos kokybės detalumas [12]. Per pastaruosius dešimtmečius šio tipo glaudinimo algoritmai buvo vis labiau vystomi, buvo pasiekti aukšti glaudinimo rezultatai (1.1 lentelė.).

1.1 lentelė Progresuojančių daugiakampių tinklo kodavimo algoritmų palyginimas [10]

Algoritmas	Suspaudimo santykis (bpv)	Tinkamas daugialypiai geometrijai, koduoti	Trūkumas
Progresuojančių daugiakampių tinklo [Hoppe 1996]	37	ne	Nepalaiko standartizuotų formatų ir įrankių
Suspaustų progresuojančių daugiakampių tinklo [Pajarola and Rossignac 2000]	22	ne	
Valentinio kodavimo [Alliez and Desbrun 2001a]	21	ne	
Banguojančio daugiakampių tinklo [Valette and Prost 2004b]	19	ne	Mažas detalumo lygių kiekis
Spektrinis glaudinimo [Karni and Gotsman 2000]	19	ne	
Kd medžių kodavimo [Gandoin and Devillers 2002]	19	taip	Gali atsirasti iškraipymų
Aštuntainių medžių glaudinimo [Peng and Kuo 2005]	15	taip	Gali atsirasti iškraipymų
Pavienių parametrų glaudinimo [Valette et al. 2009]	15	ne	Dekodavimo metu originalus sujungimas gali neatsirasti
Geometrijos vaizdo [Gu et al. 2002]	-	ne	Gali atsirasti įtrūkimų, nuostolingas
Vilnelių glaudinimo [Khodakovsky et al. 2000]	8	ne	Nuostolingas
Normalaus daugiakampių tinklo [Guskov et al. 2000]	6	ne	Nuostolingas

Progresuojančio daugiakampių tinklo metodai yra sukurti apdoroti tvarkingus ir uždarus viršūnių tinklus, todėl apdorojant daugiakampių tinklą, kurie susideda iš kelių apjungtų daugiakampių tinklų, lemia prastesnius rezultatus [12]. Vienas iš sprendimo būdų yra išskirstyti daugiakampių tinklą į atskirus objektus, tačiau tai suteikia nepatogumų dirbant su didelės apimties scenomis. Dauguma iš progresuojančių daugiakampių tinklo algoritmų turi trūkumų, kurie lemia galutinio rezultato netikslumus [10]. Dauguma algoritmų yra sukonzentruoti į glaudinimo efektyvumą, o ne į atvaizdavimo kokybę, todėl grafinis šių algoritmų atvaizdavimas pirmose stadijose gali stipriai skirtis nuo originalaus modelio (1.10 pav. A ir C).



1.10 pav. Aštuntainio medžio progresyvaus daugiakampių tinklo atvaizdavimas [13]: (A) suglaudinto modelio pradinis atvaizdavimas, (B) atvaizdavimas įpusėjus modelio įkrovimą, (C) pilnas modelio atvaizdavimas

Progresuojančio daugiakampių tinklo algoritmai yra netinkami naudoti kuriant interaktyvias interneto programas. Dauguma šio tipo algoritmų turi trūkumų, kurie trukdo juos panaudoti komerciniuose projektuose, o standartizuotų formatų nepalaikymas sumažina panaudojimo galimybes. Algoritmai yra ribotų galimybių ir dauguma jų negali saugoti papildomų 3D daugiakampių tinklo atributų (spalvų, tekstūrų ir kiti susiję atributai) [14].

1.2.3 3D turinio glaudinimo algoritmai

3D turinio glaudinimo algoritmai, kitaip nei progresuojančio daugiakampių tinklo algoritmai, turinį atvaizduoja tik baigus iki galo perduoti visus reikalingus duomenis ir nuskaičius duomenų turinį. Nors šių algoritmų efektyvumas yra mažesnis, nei progresuojančių daugiakampių tinklo metodų, 3D turinio glaudinimo algoritmai yra universalesni ir plačiau pritaikomi komerciniams projektams. Naudojant šiuos algoritmus aparatinės įrangos išteklių kiekis, reikalingas turinio apdorojimui, yra mažesnis, nei progresuojančių tinklelių algoritmų. Toliau pateikta 3D turinio glaudinimo algoritmų efektyvumo palyginimo lentelė (1.2 lent.)

1.2 lentelė Testuotų modelių dydis KB Glaudinimas atliekamas naudojant i7 CPU, 3.4 GHz.. Parinkti formatai turi integruotą papildomą GZIP glaudinimą perdavimo tinklu metu. Atvaizdai ir tekstūros neįtrauktos į skaičiavimus [12]

	Viršūnių skaičius	X3D		BG		CTM-G		Chun		CTM	
		RAW	GZIP	RAW	GZIP	RAW	GZIP	RAW	GZIP	RAW	GZIP
Galinis kiemas	2,625	240	71	79	46	147	43	62	43	31	31
Faraonas	8437	618	227	185	151	495	116	149	111	81	81
Traktorius	27251	2296	617	646	431	1539	361	506	301	259	259
Paukštis	69948	7330	2454	1647	5465	1197	1197	1453	1020	947	948

OpenCTM yra atviro kodo dvejetainio tipo formatas 3D daugiakampių tinklo glaudinimui. Šis formatas siūlo aukštus duomenų glaudinimo rezultatus ir greitą duomenų išspaudimo greitį stacionariuose ir nešiojamuose kompiuteriuose. *OpenCTM* turi galimybę glaudinti tik 3D daugiakampių tinklo informaciją, todėl kita scenos informacija nėra glaudinama. Šis formatas turi tris glaudinimo lygius (*RAW*, *MG1* ir *MG2*), *MG2* režimas yra kompaktiškiausias [12]. *OpenCTM* formatas nėra apdorojamas pačios naršyklės, todėl JavaScript puseje reikalinga atlikti papildomus veiksmus duomenų dekodavimui [12].

OpentCTM daugiakampių tinklas turi šiuos parametrus [15] :

- Viršūnę sudaro atributų rinkinys, kuris nurodo unikalias savybes (koordinatės, normalė, UV koordinatės, svoriai ir kiti parametrai);
- Trikampis aprašomas nurodant trijų viršūnių indeksus.

Nuskaitant *OpenCTM* formato failą visa daugiakampių tinklo informacija yra laikoma masyve (sveikųjų skaičių masyvas indeksams ir slenkančio kabelio masyvai viršūnių informacijai). Kiekvienas masyvas privalo turėti vienodą skaičių elementų. Visa informacija (išskyrus trikampių indeksus ir viršūnių koordinates) yra pasirinktina, todėl esant poreikiui nereikalinga informacija gali būti nesaugoma [15]. Toliau pateiktas trikampių masyvas:

tri_0^0	tri_0^1	tri_0^2	tri_1^0	tri_1^1	tri_1^2	...	tri_M^1	tri_M^2	tri_M^3
-----------	-----------	-----------	-----------	-----------	-----------	-----	-----------	-----------	-----------

čia tri_k^j yra viršūnės indeksas k trikampio j kampui [15]. Toliau pateiktas viršūnių masyvo pavyzdys:

x_0	y_0	z_0	x_1	y_1	z_1	...	x_N	y_N	z_N
-------	-------	-------	-------	-------	-------	-----	-------	-------	-------

čia x_k , y_k ir z_k yra k viršūnės x , y ir z koordinatės, viršūnių koordinatės yra saugomos slenkančio kabelio skaičių formatu [15]. Analogiškai saugomas normalių masyvas:

x_0	y_0	z_0	x_1	y_1	z_1	...	x_N	y_N	z_N
-------	-------	-------	-------	-------	-------	-----	-------	-------	-------

čia x_k , y_k ir z_k yra k normalės x , y ir z komponentės [15]. Daugiakampių tinklas gali turėti daugiau nei vieną UV koordinačių planą, kuris tyru šiuos parametrus:

- UV koordinačių masyvas;
- Unikalus UV koordinačių plano pavadinimas;
- Failo pavadinimo nuoroda (pasirinktinas).

Kiekviena UV koordinatė yra aprašoma dviem slenkančio kabelio reikšmėmis u ir v . Toliau pateiktas UV koordinačių masyvo pavyzdys:

u_0	v_0	u_1	v_1	u_2	v_2	...	u_N	v_N
-------	-------	-------	-------	-------	-------	-----	-------	-------

čia u_k ir v_k yra k UV koordinatės u ir v komponentės [15]. Viršūnių atributai yra aprašyti analogiškame masyve:

a_0	b_0	c_0	d_0	a_1	b_1	c_1	d_1	...	a_N	b_N	c_N	d_N
-------	-------	-------	-------	-------	-------	-------	-------	-----	-------	-------	-------	-------

čia a_k , b_k , c_k ir d_k yra keturios k atributo reikšmės.

OpentCTM formatas palaiko tris skirtingus glaudinimo metodus. Metodai skiriasi naudojamais algoritmais ir glaudinimo efektyvumu [16]:

- Neapdorotas iki 51% originalaus failo dydžio;
- *MG1* metodas iki 15% originalaus failo dydžio;
- *MG2* metodas iki 7% originalaus failo dydžio.

Neapdorotas metodas nenaudoja glaudinimo algoritmo. Galutinis failas saugo originalią informaciją neglaudintoje formoje, tačiau kitokiu formatu. Šis metodas naudingas tik tokioje situacijoje, kai duomenų įrašymo greitis ir mažas resursų sunaudojimas yra svarbesnis, nei failų dydžio sumažinimas. Šis metodas naudingas ir tuo atveju, kai reikalingas greitas failo turinio interpretavimas programose, nesuglaudinti duomenys yra greičiau nuskaitomi [15].

MG1 glaudinimo metodas veikia efektyviau nei neapdorotas *OpenCTM* metodas. *MG1* metodas naudoja nenuostolingą *LZMA* glaudinimo algoritmą. Trikampių duomenys yra suglaudunami iki 17% originalaus failo dydžio, o viršūnių duomenys suglaudunami iki 75% originalaus failo dydžio [15]. *MG1* metodas yra lėtas glaudinant (lyginant su neapdorotu metodu), tačiau išskleidimo greitis yra labai aukštas dėl naudojamo *LZMA* dekodavimo algoritmo ir praprasto duomenų formato [15].

MG2 metodas, lyginant su kitais metodais, siūlo aukščiausią glaudinimo rezultatą. Šis metodas naudoja *LZMA* algoritmą duomenų glaudinimui, tačiau viršūnių informacija yra suglaudinama labiau nei *MG2* metode. Viršūnių informacija yra perkoduojama į fiksuoto kabelio reprezentaciją, o tai leidžia nenuostolingą, nuspėjimu pagrįstą duomenų glaudinimą [15]. *MG2* metodas yra greitesnė *MG1* metodo versija ir dėl mažesnio dydžio *MG2* metodas pasižymi greitesniu duomenų įkrovimu.

1.3 Paveikslų glaudinimo analizė

Apdorojus tekstūras tam skirtose programose rezultatas yra eksportuojamas *png* formatu jei reikalinga išsaugoti permatomumo reikšmes, o visais kitais atvejais *jpeg* formatu. *Jpeg* formato naudojimas glaudinimui sumažina paveikslų dydį iki 25 kartų, priklausomai nuo to ar naudojamas nuostolingas ar nenuostolingas glaudinimo metodas [17]. Tiek *png*, tiek *jpeg* formatai yra maksimaliai suspaudžiamas eksportuojant duomenis iš programinės įrangos, todėl papildomai apdoroti šių duomenų nereikia. Pavyzdys, pateiktas 1.11 pav., rodo skirtumą tarp atvaizdų pritaikius skirtingus glaudinimo metodus. Atvaizdų užimama atmintis ženkliai skiriasi, tačiau *jpeg* atvaizdas turi labiau susiliejusias spalvas dėl pritaikyto nuostolingos glaudinimo. Kadangi atvaizdų matmenys yra gana maži, skirtumas tarp atvaizdų spalvų pasiskirstymo yra nedidelis, būtent todėl *jpeg* maksimalus glaudinimas ypač tinkamas mažos raiškos atvaizdams suspausti. Atvaizdų formatai *jpeg* ir *png* yra standartiniai formatai naudojami grafikoje.



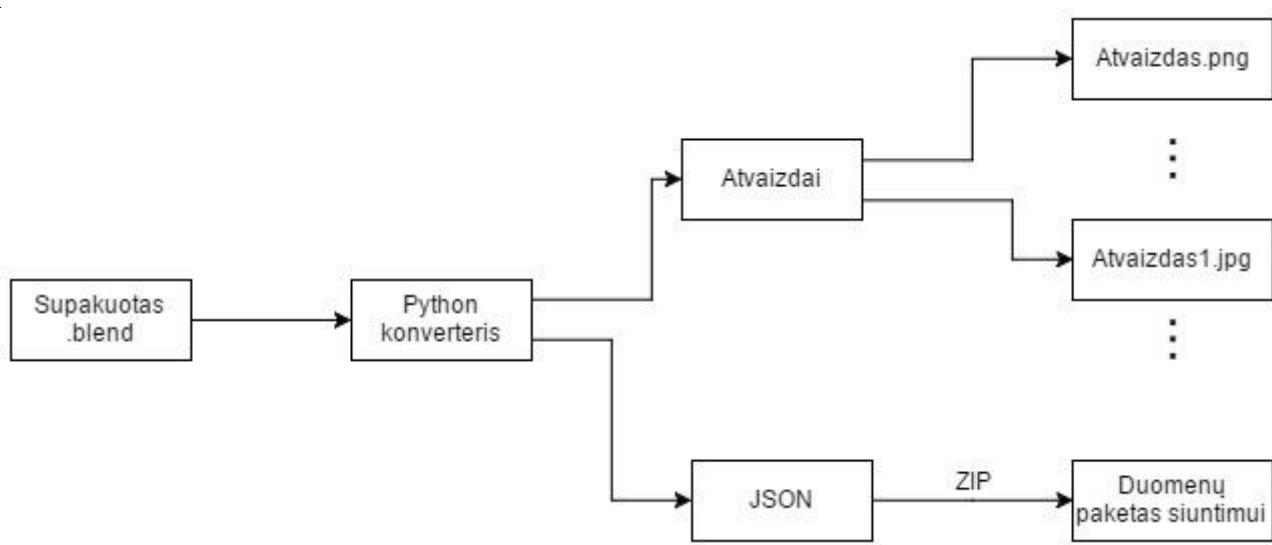
1.11 pav. 600×400px atvaizdai pritaikius skirtingą glaudinimą: (A) neglaudintas atvaizdas 4,55MB; (B) maksimaliai suglaudintas png atvaizdas 1,03MB; (C) maksimaliai suglaudintas jpeg atvaizdas naudojant nuostolingą metodą 49,8KB

1.4 Interaktyvios interneto aplikacijos sistemos analizė prieš atnaujinimą

Multimedijos duomenys, talpinami serveryje, yra suglaudintu formatu, todėl būtina duomenis paruošti prieš įkeliant į serverį. Multimedijos duomenys, baigus juos apdoroti 3D turinio kūrimo programinėje įrangoje *Blender*, yra saugomi supakuotame *blend* faile, kuris talpina ne tik 3D turinį, bet ir tekstūras, animacijas ir kitus sistemai reikalingus duomenis. Prieš duomenų naudojimą *blend* failas apdorojamas naudojant *Python* keitiklį, kuris yra įmonės UAB „Indeform“ sukurtas produktas. Pagal toliau pateiktą pseudo kodą duomenys yra suskirstomi į du paketus: atvaizdų ir scenos objektų ir jų duomenų:

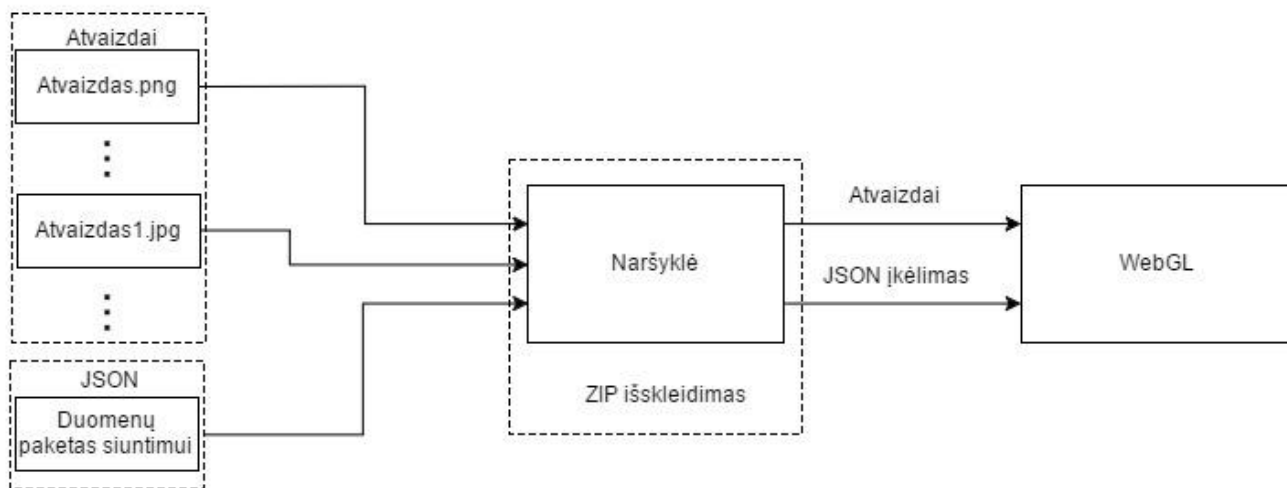
```
Scene data processing
  WHILE data
    IF scene object or scene data= true
      add to JSON
    ELSE IF image = true
      add to folder
  ENDWHILE
Compress JSON data to JSON.zip
```

Apdorojus duomenis galutinis rezultatas - *blend_data.json.zip* ir atskiras aplankas su tekstūromis *packed_files*. Keitiklis apdorojimo metu atvaizdus talpina į atskirą aplanką, o scenos objektų duomenys konvertuojami į *JSON* tipo failą. *JSON* duomenų paketas papildomai glaudinamas naudojant *ZIP* glaudinimo formatą. Pilna duomenų apdorojimo schema prieš talpinant serveryje yra pavaizduota 1.12 pav.



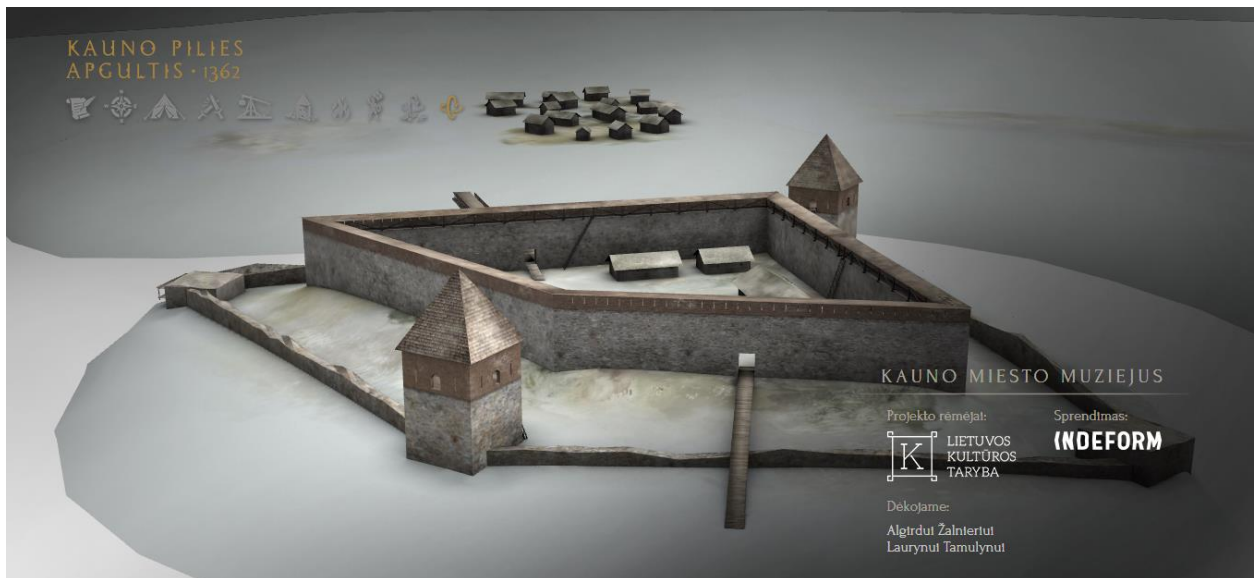
1.12 pav. Duomenų apdorojimo schema

Apdoroti duomenys yra talpinami serveryje. Paleidus interneto programą duomenys persiunčiami į naudotojo kompiuterį. *JSON* suglaudinti duomenys atsiunčiami pirmiausiai. Sistemai apdorojant gautą *JSON* failą asinchroniniu būdu, siunčiami atvaizdų failai (1.13 pav.).



1.13 pav. Duomenų priėmimo ir įkrovimo į sistemą schema kliento dalyje

Persiųsti multimedijos duomenys yra įkraunami į *WebGL* pagrindu sukurtą 3D turinio atvaizdavimo sistemą (1.14 pav.). Paskutiniame sistemos pagrindu sukurtame produkte 3D turinį sudaro 176 atskiri scenos objektai, kuriuos sudaro 162 684 viršūnės.



1.14 pav. 3D turinio atvaizdavimo sistema, pritaikyta kauno pilies apgulties projektui

Naudojant šį duomenų perdavimo modelį duomenų perdavimas yra per ilgas ir tai sukelia nepatogumų naudotojams. Priklausomai nuo scenos dydžio, detalumo ir atvaizdų rezoliucijos, duomenys gali užimti net iki 200Mb. Naudojamas duomenų glaudinimo ir perdavimo metodas nėra pakankamas norimam rezultatui.

2 MULTIMEDIJOS TURINIO PERDAVIMO INTERNETU METODŲ IR TINKLO ARCHITEKTŪROS PROJEKTAVIMAS

Šiame skyriuje nagrinėjamas neatnaujintas UAB „Indeform“ 3D turinio atvaizdavimo sistemos modelis ir pateikiamas duomenų glaudinimo ir perdavimo modelio atnaujinimas. Siūlomas sprendimas teoriškai lyginamas su neatnaujintu sistemos modeliu. Toliau pateiktos esamos sistemos savybės:

- 3D turinio atvaizdavimas;
- Tekstūrų atvaizdavimas;
- Apšvietimo atvaizdavimas;
- Animacijos ir kameros judesio atvaizdavimas.

2.1 Sprendimo reikalavimai

Toliau pateikti sprendimo reikalavimai:

- 3D turinio glaudinimo algoritmas turi efektyviau suglaudinti 3D turinį nei neatnaujintame sprendime;
- 3D turinio perdavimas internetu ir įkėlimas į sistema turi veikti efektyviau nei neatnaujintoje sistemos versijoje;
- 3D turinio glaudinimas turi būti nenuostolingas;
- Sprendimas turi būti pritaikytas naudoti standartizuotus formatus;
- Duomenų glaudinimas turi palaikyti kelių, sujungtų į vieną objektą, 3D daugiakampių tinklų glaudinimą.

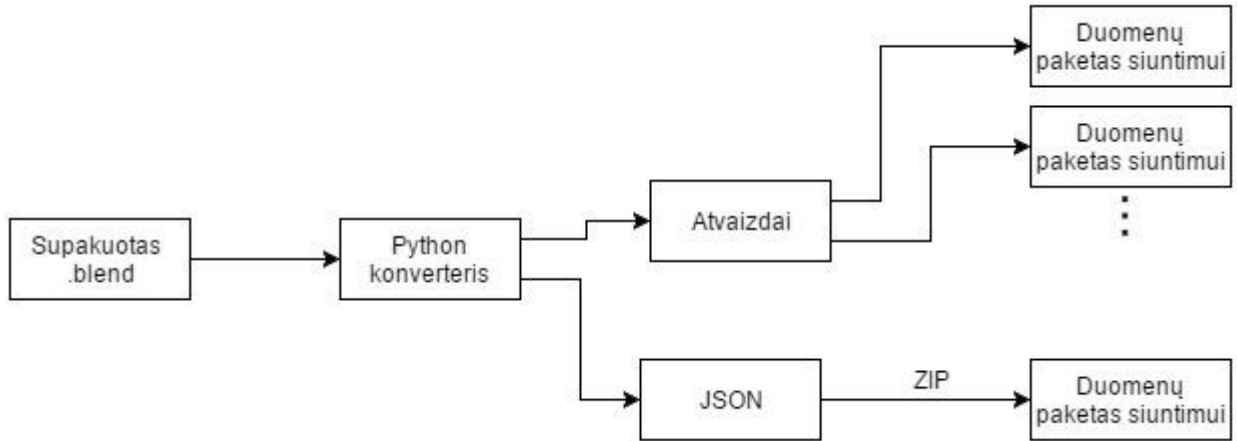
2.2 Atnaujintos interaktyvios interneto aplikacijos sistemos sprendimas

Duomenų perdavimo neefektyvumas slypi didelėje duomenų apimtyje, išskleidimo greityje ir netinkamame duomenų paskirstyme tinkle. Pirmas duomenų apdorojimo etapas yra duomenų apimties sumažinimas iki minimalaus galimo dydžio, išlaikant tenkinamą kokybės ir duomenų apimties santykį. UAB „Indeform“ produkte naudojamas multimedijos turinys, sukurtas naudojant *Blender* ir *Photoshop* programinę įrangą. Multimedijos turinys apima *blend*, *png*, *jpeg* formato failus.

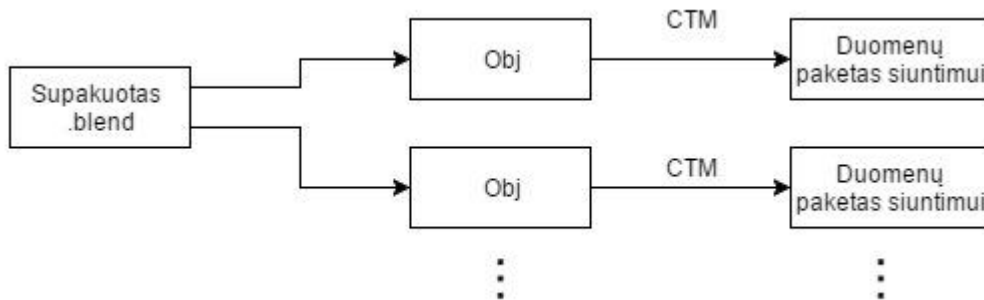
Trimatės scenos informacija, saugoma *blend* faile, yra visiškai nesuglaudinta, nagrinėdami šiuos duomenis darome prielaidą, kad trimačių objektų daugiakapiai tinklai yra maksimaliai optimizuoti pagal poreikius ir tolimesnis darbas su šiais objektais neduos geresnių rezultatų. Supakuotas *blend* failas turi būti išskirstytas ir paruoštas perdavimui tinklu. Pradinėje sistemos versijoje duomenys buvo skirstomi į du atskirus duomenų paketus. Pagal pateiktą atnaujintą pseudo kodą, duomenys skirstomi į tris duomenų paketus:

```
Scene data processing
  WHILE data
    IF mesh or mesh attribute = true
      add to obj
    ELSE IF image = true
      add to pack
    ELSE
      add to JSON
  ENDWHILE
  Obj data compress with CTM
  JSON data compress with ZIP
```

Duomenys skirstomi į atvaizdus, scenos objektų ir jų atributų informaciją bei objektus, turinčius daugiakampių tinklą, ir su jais siejamus atributus. Daugiakampių tinklą turintys objektai atskiriami tam, kad galima būtų pritaikyti atskirą glaudinimo algoritmą, kuris gali būti taikomas tik šio tipo elementams, šie duomenys saugomi *obj* formatu ir vėliau jiems pritaikomas *OpenCTM* glaudinimas. Likusi scenos informacija yra talpinama *JSON* formatu, tai yra patogu, nes šis formatas gali būti kontroliuojamas nesaugant informacijos, kuri yra nereikalinga. Atnaujinta duomenų konvertavimo schema pateikta 2.1 pav. 3D daugiakampių duomenys yra eksportuojami atskirai (2.2 pav.).

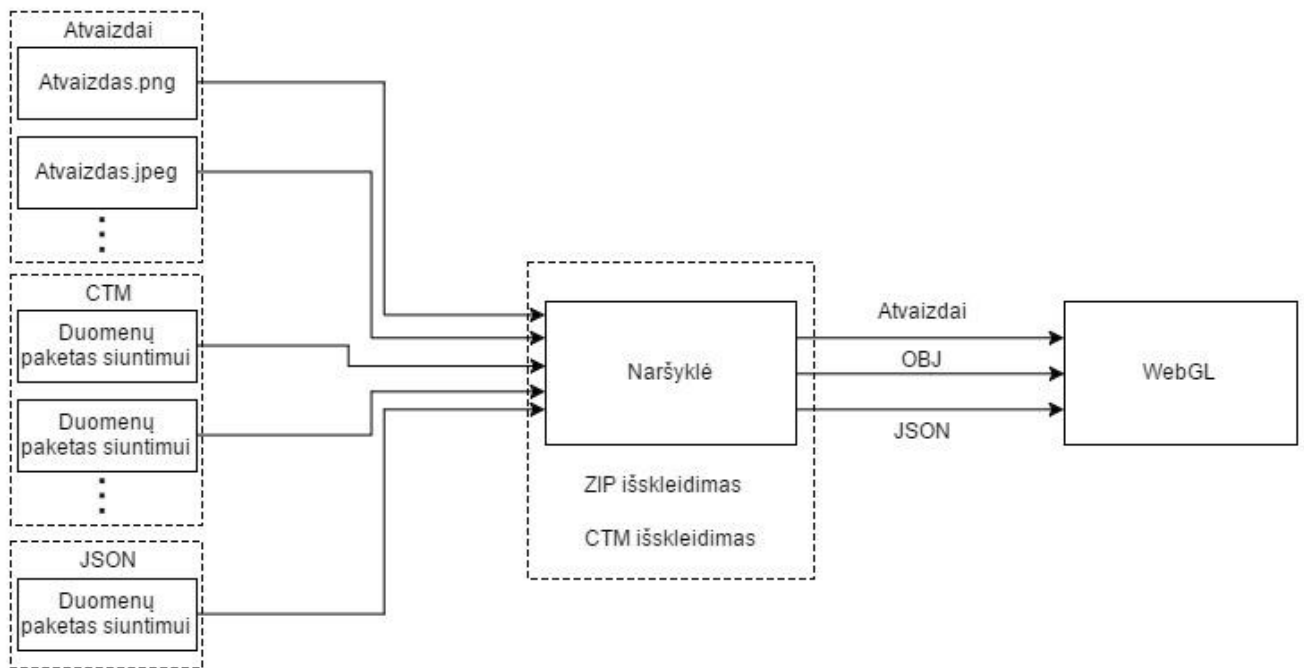


2.1 pav. Atnaujinta duomenų apdorojimo schema serverio dalyje



2.2 pav. 3D daugiakampių tinklo ir jų parametrų eksportavimo schema

Suskirstyti duomenys, kaip pavaizduota 2.1 pav., skirstomi į mažesnius paketus. Taip suskirsčius duomenis galima valdyti jų srautą į vartotojo kompiuterį, pasirenkant reikalingiausias duomenis siųsti pirmiau nei mažiau svarbius. Paskirstytas duomenų srautas (2.3 pav.) perduodamas į vartotojo kompiuterį ar kitą įrenginį.



2.3 pav. Atnaujinta duomenų apdorojimo schema kliento dalyje

3 MULTIMEDIJOS TURINIO GLAUDINIMO IR PERDAVIMO INTERNETU METODŲ REALIZACIJA IR EKSPERIMENTŲ REZULTATAI

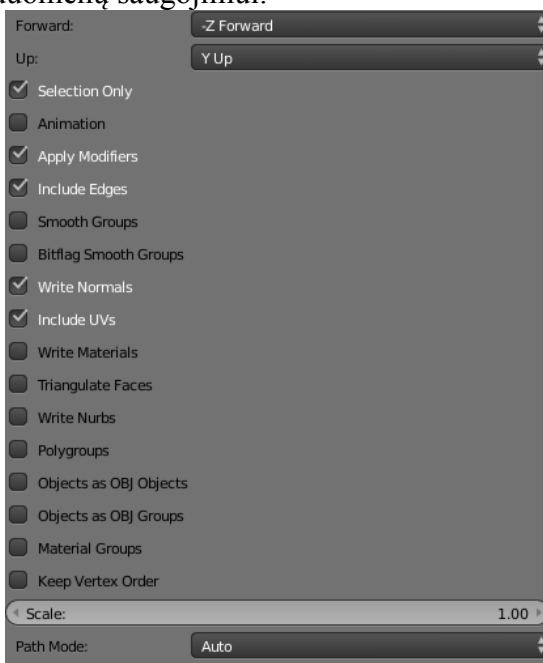
Šiame skyriuje pateikiami 3D turinio atvaizdavimo sistemos realizacijos ir eksperimento rezultatai. UAB „Indeform“ 3D turinio atvaizdavimo sistemos turinys yra 2 tipų:

- Neminimizuotas turinys;
- Minimizuotas turinys, kuris yra pritaikytas planšetiniams kompiuteriams.

Eksperimento rezultatai pateikiami po eksperimento su abiejų tipų turiniais.

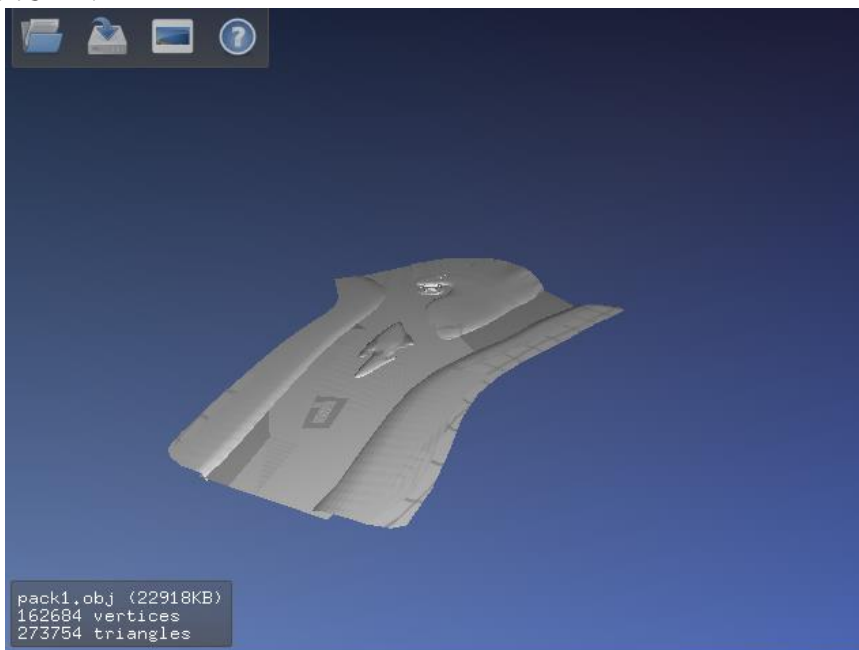
3.1 3D turinio glaudinimo realizacija, eksperimentinis tyrimas ir analizė

Multimedijos duomenys yra skirstomi naudojant *Python* keitiklį, kurio esmė yra atskirti *.blend* faile saugomą turinį ir išsaugoti jį atskirus duomenų paketus ir suglaudinti naudojant pasirinktus duomenų glaudinimo algoritmus. Atvaizdai nėra glaudinami, todėl *python* keitiklis juos atskirai išsaugo aplanke *packed_files*. 3D turinys yra saugomas *.obj* formatu naudojant standartinius eksportavimo įrankius *Blender* programinėje įrangoje (3.1 pav.). Eksportuojant duomenis pažymime išsaugoti kraštines, normales ir UV koordinates. *Wavefront .obj* formato struktūra yra paprasta ir neapkrauta nereikalinga informacija, šis formatas yra standartizuotas ir dažnai naudojamas 3D turinio apdorojimo programose, todėl puikiai tinka duomenų saugojimui.



3.1 pav. 3D daugiakampių tinklo ir jų parametrų eksportavimo pasirinkimai *Blender* aplinkoje

Išeksportuoti duomenys *wavefront .obj* formatu yra suglaudunami naudojant *OpenCTM 3D viewer* programinę įrangą. *OpenCTM 3D viewer* programinė įranga (3.2 pav.) glaudina duomenis naudojant *MG2* glaudinimo metodą, todėl galutinis rezultatas yra labiausiai suglaudintas, kokį galima gauti, naudojant *OpenCTM*.



3.2 pav. *OpenCTM 3D* turinio peržiūrėjimo ir eksportavimo įrankis

3D daugiakampių tinklo ir jų parametrų glaudinimas pritaikomas šešiams skirtingiems duomenų rinkiniams. Duomenų rinkinių glaudinimo efektyvumas testuojamas neskaidžius turinio, suskaidžius į penkias ir devynias dalis. Pirmojo 3D turinio, turinčio 162684 viršūnes, pirmas duomenų rinkinys susideda iš 1 suglaudinto failo, kuriame sutalpinta visa 3D daugiakampių informacija. Antrasis duomenų rinkinys yra sudarytas iš 5 suglaudintų failų, tai yra tas pats turinys kaip ir pirmojo rinkinio atveju tačiau padalintas į 5 dalis, o vėliau suglaudintas. Trečiasis duomenų rinkinys sudarytas iš 9 failų, kurie po išskaidymo yra suglaudunami. Duomenų glaudinimo rezultatų palyginimas yra pateikiamas 3.1 lentelėje.

3.1 lentelė 3D turinio su objektais, kurių 3D daugiakampių tinklas turi 162684 viršūnės, glaudinimo rezultatai naudojant CTM formatą

Pavadinimas	Pirmasis duomenų rinkinys			Antrasis duomenų rinkinys			Trečiasis duomenų rinkinys		
	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB
Pack1	162684	22918	1473	30939	4453	246	17770	2477	160
Pack2	-	-	-	33893	4704	305	16711	2353	167
Pack3	-	-	-	31072	4576	313	19835	2729	159
Pack4	-	-	-	35887	4809	297	20185	2785	156
Pack5	-	-	-	30893	4376	313	19823	2725	171
Pack6	-	-	-	-	-	-	16569	2385	160
Pack7	-	-	-	-	-	-	17252	2494	179
Pack8	-	-	-	-	-	-	17091	2487	175
Pack9	-	-	-	-	-	-	17448	2483	149
Viso:	162684	22918	1473	162684	22918	1474	162684	22918	1476

Pirmojo duomenų rinkinio atveju turinio dydis sumažintas 15,5 karto. Antrojo ir trečiojo duomenų rinkinio glaudinimo rezultatai skiriasi nuo pirmojo duomenų rinkinio vos 0,1-0,2%. Glaudinimo rezultatų skirtumas tarp skaidyto ir neskaidyto turinio yra nedidelis dėl duomenų turinio sandaros. Jei glaudinamas turinys turi didesni sutampančių duomenų kiekį, glaudinimo rezultatai rodo aukštesnius rezultatus. Glaudinant antrąjį 3D turinį, kurį sudaro 1934676 viršūnės, rezultatai rodo, kad

neskaidytų duomenų galutinis rezultatas yra 23 kartus mažesnis, nei pradinis failas. Glaudinant 3D turinį, kuris yra suskaidytas, gaunamas rezultatas yra 33% didesnis, nei glaudinant neskaidytą turinį.

3.2 lentelė Duomenų glaudinimo rezultatai testuojant su 1934676 viršūnes turinčiu 3D turiniu neskaidžius turinio, suskaidžius į penkias ir devynias dalis

Pavadinimas	Pirmasis duomenų rinkinys			Antrasis duomenų rinkinys			Trečiasis duomenų rinkinys		
	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB	Viršūnių skaičius	Pradinis failo dydis KB	Failo dydis po glaudinimo KB
Pack1	1934676	238347	10379	429928	52150	3440	214964	26178	1757
Pack2	-	-	-	429928	52846	3473	214964	26124	1755
Pack3	-	-	-	444812	54738	3363	214964	26052	1722
Pack4	-	-	-	384322	47072	3058	214964	26642	1714
Pack5	-	-	-	245686	31541	1996	214964	26715	1717
Pack6	-	-	-	-	-	-	214964	26829	1739
Pack7	-	-	-	-	-	-	214964	26801	1718
Pack8	-	-	-	-	-	-	214964	26646	1716
Pack9	-	-	-	-	-	-	214964	26360	1725
Viso:	1934676	238347	10379	1934676	238347	15330	1934676	238347	15563

Glaudinamo rezultatai (3.2 lent.) rodo, kad didesnės apimties glaudinamas turinys yra efektyviau suglaudinamas, nei mažesnės apimties. Palyginę išskaidyto ir neskaidyto turinio rezultatus matome, kad efektyviausiais glaudinimas yra nesuskaidžius turinio. Palyginus glaudinimo rezultatus matome, kad pirmojo 3D turinio glaudinimo efektyvumas yra 74 bpv, o antrojo 3D turinio efektyvumas yra 44 bpv, todėl darome prielaidą, kad glaudinimo efektyvumas tiesiogiai priklauso nuo glaudinamo turinio kiekio.

3.2 Duomenų perdavimo internetu metodų realizacija ir eksperimentinė analizė

Didžiausią dalį duomenų sudaro paveikslai, kadangi jų glaudinimui pritaikyti .png ir .jpeg formatai, tolimesnis duomenų glaudinimas naudojant nenuostolingus metodus neduos aukštesnių rezultatų. Šiai problemai spręsti pasiūlytas asinchroninis duomenų perdavimas atvaizdams ir sinchroninis perdavimas likusiems duomenis. Toks duomenų paskirstymas yra efektyvesnis, nes suglaudinti duomenys išpakuojami prieš siunčiant likusius duomenis. Sistemos veikimas apibūrinamas nuosekliu duomenų perdavimu. Siunčiamas „Pack1.ctm“ failas, pabaigus siūsti ir išpakuoti yra siunčiamas „blend_data.json.zip“ failas, kuris atsiuntus taip pat yra išskleidžiamas. Pabaigus siūsti 3D turinio duomenis yra siunčiamas atvaizdų srautas asinchroniniu būdu (visi atvaizdai siunčiami vienu metu). Toliau pateiktas pseudo kodas duomenų perdavimui ir įkrovimui:

```

Data load and parse processing
  Load CTM
  IF CTM loaded = TRUE
    Start CTM data parse
  IF CTM data parsed = TRUE
    Start JSON.ZIP load
    IF JSON.ZIP loaded = TRUE
      UNZIP JSON.ZIP
      IF JSON.ZIP unzipped = TRUE
        PARSE JSON
    IF JSON parsed = TRUE
      LOAD and PARSE images
  END

```

3.3 lentelėje pateikti duomenys rodo paveikslų perdavimo laiką, matuojant kiekvieno paveikslų perdavimą neminimizuotoje sistemoje, testuojant stacionariame ir nešiojamame kompiuteryje. Testuojama buvo tris kartus, kiekvieną kartą išvalant spartinčiąją atmintinę, lentelėje pateikti duomenys yra trijų matavimų vidurkis. Atskiri atvaizdavimų laikai matuojami neatsižvelgiant į galimą laiko persidengimą, šis matavimas atliktas siekiant patikrinti kaip efektyviai siunčiami atvaizdai.

3.3 lentelė Atvaizdų perdavimo laikai neminimizuotoje sistemoje, neatsižvelgiant į siuntimo laiko persidengimą, pateikti rezultatai yra trijų matavimų vidurkis

Pavadinimas	Dydis KB	i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps		i7-3930K, 3.20 GHz, 12GB RAM, 94 Mbps	
		Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms	Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms
bridge.jpg	635	68	65	48	38
bridge-burned.jpg	646	84	66	42	39
camp-1.jpg	152	123	84	71	53
camp-1_base.jpg	79	68	59	30	28
camp-2.jpg	180	128	98	71	72
camp-2_base.jpg	91	54	58	36	31
camp-3.jpg	205	123	124	74	72
camp-3_base.jpg	117	61	69	31	30
camp-4.jpg	70	76	59	36	40
camp-4_base.jpg	31	40	33	15	12
camp-5.jpg	162	287	136	74	71
camp-5_base.jpg	64	98	53	29	20
castle-houses.jpg	683	159	166	94	87
castle-houses-burned.jpg	519	230	91	47	39
castle-island.jpg	1516	558	514	305	293
castle-island_burned1.jpg	1460	625	564	311	316
castle-island_burned2.jpg	1560	698	557	333	335
castle-island_burned3.jpg	1786	666	597	371	362
castle-wall.jpg	1838	223	256	134	133
castle-wall-burned.jpg	1699	218	207	134	131
crusaders_defence.png	76	8	4	2	5
fosa.png	226	65	69	35	31
gallery.jpg	491	118	156	82	82
gallery-burned.jpg	291	63	85	34	35
ground-base.jpg	2	2	3	0	0
guidelines.png	21	84	40	27	28
icon_crusaders.png	6	6	4	3	3
icon_raigaine.png	6	6	20	3	3
icon_teutonic.png	8	235	120	17	18
infogr_fosa.png	4	8	10	2	2
infogr_griovys-su-pylimu.png	15	24	27	9	11
infogr_pilies-igula.png	7	9	12	6	6
infogr_pilies-siena.png	5	6	6	3	4
infogr_priespilio-bokstas.png	12	19	13	260	40
infogr_priespilis.png	6	7	8	4	3
map-big.jpg	1571	217	208	121	119
map-small.jpg	1036	156	142	88	64
rampart.png	120	46	69	20	17
river1.jpg	16	6	8	2	1

		i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps		i7-3930K, 3.20 GHz, 12GB RAM, 94 Mbps	
Pavadinimas	Dydis KB	Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms	Pavadinimas	Dydis KB
river2.jpg	11	5	4	4	2
river3.jpg	10	5	4	14	6
river-base.jpg	2	116	35	1	2
roof.jpg	769	168	164	90	90
roof-burned.jpg	512	71	68	39	39
ship.jpg	179	46	37	80	34
siege-tower.jpg	472	17	17	10	9
siege-weapons.png	5665	318	185	139	135
soldier-armor-base.png	4668	196	189	117	112
soldier-armor-cr.png	3969	201	189	118	111
tents_texture.png	1028	117	48	26	25
tower.jpg	1228	218	195	108	106
tower_top.jpg	978	177	193	135	105
village-grass_burned.jpg	79	36	47	19	17
village-grass_normal.jpg	82	37	34	18	21
village-houses.jpg	744	168	153	95	93
weapons.jpg	4809	405	257	130	124
gray-vignette.jpg	133	48	41	21	21
Viso:	42750	8021	6720	4168	3726

Rezultatai rodo, kad neminimizuotoje ir neatnaujintoje sistemoje, nešiojamame ir stacionariame kompiuteryje, paveikslai buvo perduodami ilgesnį laiką, nei atnaujintoje sistemoje. Tai rodo, kad atnaujintos sistemos sprendimas efektyviau išnaudoja turimus išteklius, neatnaujintoje sistemoje atvaizdai siunčiami, kai *JSON.zip* failas yra apdorojamas, todėl sistema yra labiau apkrauta, nei atnaujintos sistemos atveju. Stacionariame kompiuteryje skirtumas tarp atnaujintos ir neatnaujintos sistemos rezultatų yra mažesnis, nei nešiojamame kompiuteryje, tai rodo, kad siuntimas asinchroniniu būdu, sistemoje, kuri turi mažesnę tinklo pralaidumą ir sistemos išteklius, asinchroninis siuntimas maksimaliai išnaudoja turimus išteklius.

Atvaizdų persiuntimo testavimas su minimizuoto turinio sistema parodė mažesnę laiko skirtumą tarp atnaujintos ir neatnaujintos sistemos versijos (3.4 lent.). Mažesnis laiko pokytis egzistuoja dėl minimizuoto turinio dydžio. Minimizuoto turinio atvaizdų dydis yra beveik 40MB mažesnis, nei neminimizuoto turinio, todėl minimizuoto turinio perdavimas neapkrauna naudotojo kompiuterio taip, kaip neminimizuotas turinys.

3.4 lentelė Minimizuoto turinio sistemos atvaizdų perdavimo laikai, neatsižvelgiant į siuntimo laiko persidengimą, pateikti rezultatai yra trijų matavimų vidurkis

Pavadinimas	Dydis KB	i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps		i7-3930K, 3.20 GHZ, 12GB RAM , 94 Mbps	
		Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms	Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms
bridge.jpg	51	98	102	7	7
bridge-burned.jpg	38	45	43	18	15
camp-1.jpg	21	83	54	5	8
camp-1_base.jpg	19	57	36	2	9
camp-2.jpg	23	78	12	5	7
camp-2_base.jpg	21	76	55	98	88
camp-3.jpg	27	32	27	70	70
camp-3_base.jpg	25	84	26	29	23
camp-4.jpg	10	78	66	12	15
camp-4_base.jpg	8	15	34	8	45
camp-5.jpg	20	64	26	45	55
camp-5_base.jpg	18	88	63	8	7
castle-houses.jpg	31	36	62	7	5
castle-houses-burned.jpg	33	35	55	12	4
castle-island.jpg	38	60	29	12	12
castle-island_burned1.jpg	42	40	32	15	15
castle-island_burned2.jpg	43	28	85	74	52
castle-island_burned3.jpg	48	118	121	88	74
castle-wall.jpg	86	142	126	10	11
castle-wall-burned.jpg	71	102	37	57	48
crusaders_defence.png	6	27	14	6	9
fosa.png	59	31	45	15	22
gallery.jpg	28	24	21	11	27
gallery-burned.jpg	23	78	57	3	3
ground-base.jpg	2	10	4	71	27
guidelines.png	21	84	90	129	68
icon_crusaders.png	6	35	51	2	4
icon_raigaine.png	6	55	42	3	10
icon_teutonic.png	8	52	87	37	7
infoгр_fosa.png	4	20	15	31	6
infoгр_griovys-su- pylimu.png	15	78	80	6	5
infoгр_pilies-igula.png	7	53	38	24	25
infoгр_pilies-siena.png	5	66	58	2	9
infoгр_priespilio-bokstas.png	12	21	58	62	11
infoгр_priespilis.png	6	46	34	3	4
map-big.jpg	122	55	19	31	14
map-small.jpg	96	68	81	22	43
rampart.png	30	10	42	4	12
river1.jpg	16	85	93	5	22
river2.jpg	11	151	147	64	38
river3.jpg	10	8	6	2	20
river-base.jpg	2	106	107	35	12

		i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps		i7-3930K, 3.20 GHz, 12GB RAM , 94 Mbps	
Pavadinimas	Dydis KB	Neatnaujintos sistemos persiuntimo laikas ms	Atnaujintos sistemos persiuntimo laikas ms	Pavadinimas	Dydis KB
roof.jpg	43	44	39	53	24
roof-burned.jpg	46	92	63	21	36
ship.jpg	23	54	23	6	13
siege-tower.jpg	8	28	16	6	12
siege-weapons.png	127	50	27	65	29
soldier-armor-base.png	109	98	122	46	40
soldier-armor-cr.png	118	41	43	32	47
tents_texture.png	83	65	55	29	16
tower.jpg	49	51	30	7	12
tower_top.jpg	57	44	10	30	37
village-grass_burned.jpg	19	67	31	72	6
village-grass_normal.jpg	16	75	50	1	8
village-houses.jpg	34	28	60	3	6
weapons.jpg	8	7	3	4	4
Viso:	1908	3267	2851	1525	1259

3D turinio glaudinimo rezultatai parodė, kad, glaudinant neskaidytą turinį, gaunamas geresnis rezultatas. Panaši situacija matoma testuojant neskaidytų ir skaidytų duomenų perdavimą ir įkrovimą (3.5 lent.). Lentelėje pateikti rezultatai rodo, kad geriausi rezultatai yra pasiekiami siunčiant pirmąjį duomenų rinkinį. Tokie rezultatai rodo, kad algoritmas efektyviausiai veikia apdorodamas vieną didelį failą, nei kelis mažus. Ilgiausias laikas reikalingas duomenų išskleidimui ir įkrovimui yra naudojant antrąjį duomenų rinkinį, taip yra dėl to, kad duomenų perdavimas ir išskleidimas nėra toks efektyvus kaip naudojant pirmąjį duomenų rinkinį, o failų dydis yra gana didelis lyginant su trečiuoju duomenų rinkiniu, todėl laikas prarandamas ir siunčiant ir išskleidžiant duomenis.

3.5 lentelė 3D turinio perdavimo ir išskleidimo laikas perduodant neskaidytą ir skaidytą turinį skirtingų išteklių įrenginiuose, pateikti rezultatai yra trijų matavimų vidurkis

	i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps			i7-3930K, 3.20 GHz, 12GB RAM , 94 Mbps		
Failo pavadinimas	Pirmo duomenų rinkinio perdavimo laikas ms	Antro duomenų rinkinio perdavimo laikas ms	Trečio duomenų rinkinio perdavimo laikas ms	Pirmo duomenų rinkinio perdavimo laikas ms	Antro duomenų rinkinio perdavimo laikas ms	Trečio duomenų rinkinio perdavimo laikas ms
Pack1.ctm	1398	957	350	1081	915	235
Pack2.ctm	-	1049	343	-	920	248
Pack3.ctm	-	1044	342	-	923	244
Pack4.ctm	-	981	342	-	949	241
Pack5.ctm	-	1041	362	-	929	254
Pack6.ctm	-	-	357	-	-	245
Pack7.ctm	-	-	369	-	-	262
Pack8.ctm	-	-	383	-	-	260
Pack9.ctm	-	-	339	-	-	248
Viso:	1398	5073	3186	1081	4635	2238

Didesnės apimties 3D turinio perdavimo rezultatai (3.6 lent.) taip pat parodė, kad neskaidytų duomenų perdavimas ir įkrovimas į sistemą yra efektyvesnis, nei skaidytų duomenų. Priešingai, nei apdorojant pirmąjį 3D turinį, antrasis duomenų paketo perdavimas parodė vidutinius rezultatus, o tai rodo, kad glaudinimo algoritmas veikia pakankamai efektyviai su tokio dydžio turiniu. Duomenų glaudinimas proporcingai veikia blogiau jei apdorojamas turinys yra suskaidomas.

3.6 lentelė Duomenų perdavimo, išpakavimo ir įkrovimo į sistemą trukmės duomenys, naudojant 1934676 viršūnes turintį 3D turinį, pateikti rezultatai yra trijų matavimų vidurkis

	i5-430M, 2.26 GHz, 4GB RAM, 60 Mbps			i7-3930K, 3.20 GHz, 12GB RAM , 94 Mbps		
Failo pavadinimas	Pirmo duomenų rinkinio perdavimo laikas ms	Antro duomenų rinkinio perdavimo laikas ms	Trečio duomenų rinkinio perdavimo laikas ms	Pirmo duomenų rinkinio perdavimo laikas ms	Antro duomenų rinkinio perdavimo laikas ms	Trečio duomenų rinkinio perdavimo laikas ms
Pack1.ctm	12621	5168	4074	8095	3448	3253
Pack2.ctm	-	4837	3541	-	3224	3145
Pack3.ctm	-	4801	3421	-	3510	3011
Pack4.ctm	-	5079	3510	-	3371	3014
Pack5.ctm	-	2547	3754	-	1523	3029
Pack6.ctm	-	-	4001	-	-	3046
Pack7.ctm	-	-	3547	-	-	3019
Pack8.ctm	-	-	3744	-	-	3021
Pack9.ctm	-	-	3711	-	-	3071
Viso:	12621	22432	33303	8095	15076	27609

Pritaikyto sprendimo bendras perduodamų duomenų dydis sumažėjo vos 6,3% neminimizuotų duomenų sistemoje, tačiau pritaikytas duomenų glaudinimo metodas pagreitino duomenų įkrovimo laiką (3.7 lent.) nešiojamame kompiuteryje 2,5 karto. Minimizuotos sistemos įkrovimo laikas sumažėjo 1,6 karto. Tai rodo, kad sprendimo taikymas sistemai, kurios naudotojai turi silpnus kompiuterius yra tinkamas.

3.7 lentelė 3D turinio atvaizdavimo sistemos turinio įkrovimo laiko palyginimas pradinėje ir galutinėje sistemos versijoje testuojant su i5-430M, 2.26 GHz, 4GB RAM, kai tinklo duomenų perdavimo sparta yra iki 60 Mbps, pateikti rezultatai yra trijų matavimų vidurkis

Failo pavadinimas	Neminimizuotos sistemos įkrovimas		Minimizuotos sistemos įkrovimas		Pritaikyto sprendimo neminimizuotos sistemos įkrovimas		Pritaikyto sprendimo minimizuotos sistemos įkrovimas	
	Dydis KB	Laikas ms	Dydis KB	Laikas ms	Dydis KB	Laikas ms	Dydis KB	Laikas ms
Pack1.ctm	-	-	-	-	1473	1398	1473	1398
blend_data.JSON.zip	6800	37415	6800	12960	2214	12220	2214	7556
Atvaizdai	42750	6044	2005	2482	42750	4957	2005	2147
Viso:	49550	43459	8805	15442	46437	17177	4219	9703

Testuojant pritaikyto sprendimo sistemą su galingesne aparatine įranga, rezultatai (3.8 lent.) parodė, kad neminimizuotos 3D turinio atvaizdavimo sistemos duomenų įkrovimo laikas sumažėjo 1,35 karto, o minimizuotoje sistemoje sumažėjo 1,67 karto. Pritaikyto sprendimo rezultatas neminimizuotoje sistemoje yra mažesnis, nei gautas rezultatas su nešiojamu kompiuteriu. Tokie rezultatai gauti dėl to, kad didelis kompiuterio išteklių kiekis efektyviau apdorodavo neatnaujinto ir atnaujinto sprendimo sistemas. Minimizuotos sistemos rezultatas parodė, kad pasiekti sprendimo rezultatai yra geresni, nei ant nešiojamo kompiuterio, tai yra todėl, kad didelis išteklių kiekis leido labai greitai apdoroti perduodamą turinį.

3.8 lentelė. 3D turinio atvaizdavimo sistemos turinio įkrovimo laiko palyginimas pradinėje ir galutinėje sistemos versijoje testuojant su i7-3930K, 3.20 GHz, 12GB RAM ,kai tinklo duomenų perdavimo sparta yra iki 94 Mbps, pateikti rezultatai yra trijų matavimų vidurkis

Failo pavadinimas	Neminimizuotos sistemos įkrovimas		Minimizuotos sistemos įkrovimas		Pritaikyto sprendimo neminimizuotos sistemos įkrovimas		Pritaikyto sprendimo minimizuotos sistemos įkrovimas	
	Dydis KB	Laikas ms	Dydis KB	Laikas ms	Dydis KB	Laikas ms	Dydis KB	Laikas ms
Pack1.ctm	-	-	-	-	1473	1081	1473	1081
blend_data.JSON.zip	6800	12719	6800	8947	2214	7645	2214	4278
Atvaizdai	42750	3511	2005	1118	42750	3214	2005	652
Viso:	49550	16230	8805	10065	49550	11940	4219	6011

4 IŠVADOS

1. Išanalizavus 3D turinio atvaizdavimo sistemos trūkumus paaiškėjo, kad duomenų perdavimo ir glaudinimo sprendimas yra nepakankamai efektyvus. Išanalizavus duomenų perdavimo modelius pastebėta, kad didelės apimties duomenims yra tinkamiausias sinchroninis duomenų perdavimas, o nedidelės apimties duomenims – sinchroninis. Analizės metu taip pat nustatyta, kad 3D turinio glaudinimo formatas *OpenCTM* yra 1,3 karto efektyvesnis, nei kiti glaudinimo metodai.
2. Suprojektuotas duomenų perdavimo ir glaudinimo modelis parodė, kad sistemos efektyvumui padidinti gali būti taikomi paprasti, tačiau efektyvūs duomenų perdavimo ir glaudinimo metodai.
3. Realizavus sprendimą efektyviam multimedijos duomenų perdavimui ir 3D turinio glaudinimui, rezultatai parodė, kad sinchroninio duomenų perdavimo taikymas 3D turiniui perduoti ir asinchroninio duomenų perdavimo taikymas perduoti paveikslams sunaudoja mažiau aparatinės įrangos išteklių, nei sinchroninis duomenų perdavimas, taikomas visiems duomenims perduoti. Rezultatai parodė, kad pritaikius apjungtą *OpenCTM* ir *ZIP* 3D turinio glaudinimo metodą duomenų perdavimo ir įkrovimo greitis sumažėjo daugiau, nei 38% lyginant su *ZIP* glaudinimu, kuris taikomas visiems 3D turinio duomenims glaudinti. Visos sistemos duomenų perdavimo ir įkrovimo laikas neminimizuoto turinio sistemoje sumažėjo 43,45%, o minimizuotoje sistemoje 38,72%.

LITERATŪRA

1. MURGIA M. *The world's first website went online 25 years ago today*. [interaktyvus]. 2015 [žiūrėta 2017 05 11]. Prieiga per: <http://www.telegraph.co.uk>.
2. BUTKIEWICZ M., MADHYASTHA H.V., SEKAR V. *Understanding Website Complexity: Measurements, Metrics, and Implications*. Berlin, 2011.
3. LIN C.W., LEE J.S., KUO L.C., YEH Y.J., WEN J.H. *Hybrid P2P Client-Server Data Transmission Using Dynamic Peer Grouping and Switching*. Taichung, 2012.
4. ROUSE M, WOLFF D. *peer-to-peer (P2P)*. [interaktyvus]. 2012 [žiūrėta 2017 05 04]. Prieiga per <http://searchnetworking.techtarget.com>
5. ROUSE M, SULLIVAN J. *client/server (client/server model, client/server architecture)*. [interaktyvus]. 2008. Prieiga per: <http://searchnetworking.techtarget.com>.
6. DHEEMANTH H.N. *LZW Data Compression*. Karnataka, 2014, vol 03, no. 02, pp-22-26, ISSN : 2320-0936
7. SUARJAYA I.M.A.D. *A New Algorithm for Data Compression Optimization*. 2012, Bali, Vol. 3, No.8,.
8. PETTIT N. *Understanding Normal Maps*. [interaktyvus]. 2015 [žiūrėta 2017 05 05]. Prieiga per: <http://blog.teamtreehouse.com/understanding-normal-maps>.
9. Q. E. *DX11 Displacement & Tessellation*. [interaktyvus]. 2013 [žiūrėta 2015 07 14]. Prieiga per: <https://www.marmoset.co/>.
10. MAGLO A., LAVOUE G., DUPONT F., HUDELLOT C. *3D mesh compression: survey, comparisons and emerging trends*. 2013; vol 9, no 4. Prieiga per: doi: <http://dx.doi.org/10.1145/0000000.0000000>
11. ROUSE M., SPEIRS R. *binary file*. [interaktyvus]. 2005 [žiūrėta 2017 05 05]. Prieiga per: <http://whatis.techtarget.com/definition/binary-file>.
12. LIMPER M, WAGNER S, STEIN C, JUN Y, STORK A. *Fast Delivery of 3D Web Content: A Case Study*. 2013.
13. PENG J., C. C. KUO J. *Octree-Based Progressive Geometry Encoder*. Los Angeles, 2003.
14. LAVOUE G, CHEVALIER L., DUPONT F. *Streaming Compressed 3D Data on the Web using JavaScript and WebGL*. 2013.
15. GEELNARD M. *OpenCTM Developers Manual*. 2010.
16. GEELNARD M. *OpenCTM performance*. [interaktyvus]. 2010 [žiūrėta 2017 05 05]. Prieiga per: <http://openctm.sourceforge.net>.
17. AL-ANI M. S., AWAD FH. *the jpeg image compression algorithm*. 2013.vol. 3, no 3, pp. 1055-1062, ISSN: 22311963
18. J. S, SONS K, SLUSALLEK P. *Blast A Binary Large Structured Transmission Format for the Web*. 2014..
19. MURGIA M. *The world's first website went online 25 years ago today*. [interactive]. 2015 [žiūrėta 2017 04 17]. Available from: <http://www.telegraph.co.uk/technology/internet/12061803/The-worlds-first-website-went-online-25-years-ago-today.html>