

Infotehnoloogia teaduskond

Arvutisüsteemide instituut

Mihkel Abel

3D modelleerimine ja 3D mudelite tekstureerimine VR-rakenduste jaoks virtuaalse Raekoja platsi projekti näitel

Bakalaureusetöö

Juhendaja: Eduard Petlenkov, PhD Aleksei Tepljakov, PhD

Tallinn 2017

Autorideklaratsioon

Olen koostanud antud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud. Käsolevat tööd ei ole varem esitatud kaitsmisele kusagil mujal.

Autor: Mihkel Abel

[21. mai 2017. a.]

Annotatsioon

Käesolevas bakalaureusetöös kirjeldatakse virtuaalreaalsusrakenduse jaoks virtuaalsete 3D mudelite loomist ja nende tekstureerimist. Lugejale esitatakse ülevaade modelleerimise ja tekstureerimise põhitõdedest ning töövoogudest. Bakalaureusetöö aluseks on meeskonnatööna valminunud Virtuaalse Tallinna Raekoja platsi virtuaalreaalsusrakendus.

Töö lõpptulemusena anti ülevaade 3D mudelite loomisest ja nende tekstureerimisest ning valmis Virtuaalse Tallinna Raekoja platsi VR-rakendus.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 48 leheküljel, 4 peatükki ja 31 joonist.

Abstract

Virtual reality is rapidly growing filed, which has potential to benefit in various disciplines from medicine to design.

Given thesis titled "3D Modeling and Texturing for VR Applications on the example of Tallinn Town Hall Square Project" describes 3D modeling and texturing for VR applications. Thesis provides an overview of the basics and workflow for 3D modeling and texturing. This thesis is based on Virtual Tallinn Town Hall project, which was developed in Re:creation laboratory as a studient project. The main purpose of this thesis is to cover necessary knowledge of 3D modeling and texturing needed for creating a virtual reality application. The second goal is to apply this knowledge to create a fully immersive VR application.

The end result is a description of 3D modeling and texturing methods and workflows which consecutively resulted in working Virtual Tallinn Town Hall VR application. The VR application was created using CryEngine V game engine and is playable on HTC Vive.

The first section is introtuction. In the second section author describes the basic aspects of 3D modeling and gives a full workflow example of house 3D modeling. In the third section author gives an overview about texturing, which consists of UV-mapping, materials, texture creation and texture mapping. The last section is conclusion which includes outcomes of the thesis and further development possibilites.

The thesis is in Estonian and contains 48 pages of text, 4 chapters and 31 figures.

Lühendite ja mõistete sõnastik

VR	Virtuaalreaalsus
3D	Kolmedimensiooniline
2D	Kahedimensiooniline
Mesh	Punktivõrgustik
Vertex	Punkt või positsioon 3D ruumis
Edge	Serv, mis ühendab kahte punkti
Face	3D pinna tahk
Polygon	3D pinna tahk
Cryengine V	Mängumootor
GNU Üldine Avalik Litsents	Litsents vaba tarkvara jaoks
Mb	Megabait
FBX	3D geomeetria failiformaat
TIFF	pildi failiformaat
RGB	Red, Green, Blue liitvärvimudel
Google Street View	Google kaardirakendus
HTC Vive	Riistvaraline virtuaalreaalsuse peakomplekt

Sisukord

1	Sisse	ejuhatus	9
	1.1	Taust	9
	1.2	Eesmärk	10
	1.3	Bakalaureusetöö kirjeldus	10
2	3D r	nodelleerimine	12
	2.1	Mis on 3D mudel	12
		2.1.1 3D mudeli geomeetria piirangud	12
		2.1.2 Mosaiigistamine	13
	2.2	3D modelleerimine Blender tarkvaras	14
		2.2.1 Blenderi funktsioonid ja tööriistad	15
		2.2.2 Hoone modelleerimine	18
		2.2.3 3D mudeli osadeks eraldmine	29
		2.2.4 3D mudeli eksport Blender tarkvarast	31
3	3D r	nudeli tekstureerimine	31
	3.1	UV-kaardistamine	32
	3.2	Materjalid	33
	3.3	Tekstuurid	34
		3.3.1 Nõuded tekstuuridele	35
		3.3.2 Tekstuuride loomine	36
		3.3.3 Veebikeskkondadest allalaetavad tekstuurid	39
		3.3.4 Tekstuuride ülespildistamine	40
		3.3.5 Tekstuuri fotofailide töötlustöövood	40
	3.4	Tekstuuri kaardistamine Autodesk Mayas	45
		3.4.1 Realistlikuma tekstuuri kaardistamise võimalused	46
4	Kok	kuvõte	48
Ka	sutat	tud kirjandus	49
. .	4		
Li	sa 1		53
Li	sa 2		54
Li	sa 3		55
Li	sa 4		56

Lisa 5

Lisa 6

57

58

Jooniste loetelu

1	Diagramm projektis kasutatud leerimise töövoost	11
2	Diagramm projektis kasutatud tekstureerimise töövoost	11
3	Kuvatõmmis Blenderi töölauast enne modelleerima asumist	19
4	Kuvatõmmis Blender töölauast koos taustmudeliga	20
5	Modelleerimise abimaterjalina kasutatud foto	20
6	Kuvatõmmis silmuse funktsiooni rakendamisest	21
7	Kuvatõmmis peale peitmisfunktsiooni kasutamist	21
8	Kuvatõmmis hoone akende, ukse ja eri fassaadi osade asukoha määramisest	22
9	Kuvatõmmis aknaraamide modelleerimisest	23
10	Kuvatõmmis hoone karniisi modelleerimisest	23
11	Kuvatõmmis maja kõrguse suurendamisest	24
12	Kuvatõmmis servadevahelise hulknurga loomisest	24
13	Kuvatõmmis katuse korrigeerimisest	25
14	Kuvatõmmis objektide impordi tulemusest	25
15	Kuvatõmmis katuse kaldenurga korrigeerimisest	26
16	Kuvatõmmis karniisidest	26
17	Kuvatõmmis hulknurkade 1:2 suhtest	27
18	Kuvatõmmis modelleeritud akendest	28
19	Kuvatõmmis elektrikilbi modelleerimisest	28
20	Kuvatõmmis kumeruse lisamise funktsioooni kasutamisest ukse astmel	29
21	Kuvatõmmis 3D mudeli osadeks eraldamisest	30
22	Kuvatõmmis Adobe Photoshop tarkvaras uue faili loomisest	36
23	Kuvatõmmis Adobe Photoshop tarkvarast peale tausta lisamist	37
24	Kuvatõmmis Adobe Photoshop tarkvaras loodud algsest tekstuurist	38
25	Kuvatõmmis enne nihke-filtri rakendamist	38
26	Kuvatõmmis peale nihke-filtri rakendamist	39
27	Kuvatõmmis enne perspektiivkärpimist	41
28	Kuvatõmmis peale perspektiivkärpimist	42
29	Pilt peale perspektiivkärpimist	43
30	Pilt peale nihke-filtri rakendamist	44
31	Lõplik pilt peale fototöötlust	45

1. Sissejuhatus

1.1. Taust

Virtuaalreaalsus (edaspidi ka VR) on maailmas järjest enam populaarsust koguv valdkond. Virtuaalreaalsuslahendused on rakendatavad väga erinevates valdkondades meelelahutusest kuni meditsiinini. Virtuaalreaalsus annab võimaluse luua situatsioonimudeleid inimeste õpetamise ja treenimise eesmärgil, mille muudel viisidel loomine oleks kulukas, inimestele ohtlik või võimatu. Samuti annab VR võimaluse inimeste käitumise ja reaktsioonide uurimiseks eri olukordades. Virtuaalreaalsuslahendused on tänaseks leidnud kasutust lennunduses ja militaarvaldkonnas. Tulevikus nähakse spetsiifilistel virtuaalreaalsuslahendustel rakendust meditsiiniõppes, tootearenduses ja disainis [1, 2]. Igapäevakasutuses leiab VR rakendust kõige tõenäolisemalt üldhariduses, meelelahutuses arvutimängudena ja virtuaalturismi näol.

Eestis on antud valdkond veel vähe tuntud, samuti on sellega igapäevaselt tegelev arendajaskond on suhteliselt väikesearvuline. Selle põhjuseks võib lugeda asjaolu, et tegu on uue tehnoloogiaga ning virtuaalreaalsusseadmed on suuremale osale ühiskonnast oma kõrge hinna tõttu kättesaamatud. Valdkonna aktuaalsuse, uudsuse ja käesoleva bakalaureusetöö autori huvi tõttu antud temaatika vastu, osutus see ka bakalaureusetöö teemaks.

Käesoleva bakalaureusetöö aluseks on Re:creation laboris [3] loodud Virtuaalse Tallinna Raekoja platsi projekt (edaspidi ka Projekt). Projekt valmis meeskonnatööna kahe õppesemesti jooksul. Projekti eesmärgiks oli luua virtuaalreaalsusrakendus (edaspidi ka Rakendus), milles on virtuaalselt taasloodud Tallinna Raekoja plats.

Antud bakalaureusetöös kirjeldab autor virtuaalreaalsusrakenduse jaoks virtuaalsete 3D mudelite loomist ja nende tekstureerimist. Mõlemad aspektid on Rakenduse loomise seisukohast tähtsad. Töö maht ei võimaldnud kirjeldada terve Rakenduse loomise protsessi. Projekti planeerimisest saab lugeda kaastudengi Aleksander Lapini bakalaureusetööst "Virtuaalreaalsuse rakenduse loomise planeerimine ning projektiga alustamine Tallinna Raekoja platsi näitel". Tekstueeritud 3D mudelite ekspordist mängumootorisse ja tööst mängumootoriga saab lugeda kaastudengi Georg Haudi bakalaureusetööst "Virtuaalreaalsuse loomine CryEngine-s Tallina Raekoja platsi näitel".

1.2. Eesmärk

Antud bakalaureusetöö eesmärgid on :

- omandada teadmised VR-rakenduse loomiseks vajalikes aspektides,
- esitada 3D mudelite loomise metoodiliste põhitõdede ja töövoo kirjeldused,
- esitada 3D mudelite tekstureerimise metoodiliste põhitõdede ja töövoo kirjeldused.

Käesoleva bakalaurusetöö üheks eesmärgiks võib lugeda ka Virtuaalse Raekoja platsi Projekti eesmärgi :

• luua virtuaalreaalsusrakendus, milles on virtuaalselt taasloodud Tallinna Raekoja plats, kus kasutaja saab omal valikul määratud piirides ringi liikuda ning mis on vaadeldav VR-riistvaraga.

1.3. Bakalaureusetöö kirjeldus

Töö esimene osa on sissejuhatus.

Töö teise osa esimeses pooles kirjeldatakse põhitõdesid ja nõudeid 3D mudelitele, samuti antakse ülevaade projektis kasutatud modelleerimistarkvarst Blender [4]. Teise osa teises pooles kirjeldatakse detailselt modelleerimise töövoogu (Joonis 1) ühe hoone näitel.

Töö kolmanda osa esimeses pooles kirjeldatakse tekstureerimiseks vajalikke aspekte ja tegevusi. Kolmanda osa teises pooles esitatakse kirjeldus tekstuuride loomisest ja tekstureerimise töövoo osadest (Joonis 2).

Töö neljas osa on kokkuvõte.



Joonis 1. Diagramm projektis kasutatud leerimise töövoost



Joonis 2. Diagramm projektis kasutatud tekstureerimise töövoost

2. 3D modelleerimine

Virtuaalse Tallinna Raekoja platsi 3D modelleerimiseks kasutati kolme modelleerimise tarkvara: Blender [4], Autodesk Maya [5] ja Autodesk Mudbox [6]. Käesolevas peatükis esitatakse ülevaade 3D modelleerimisest ning eelpool nimetetud tarkvarade rakendamisest.

2.1. Mis on 3D mudel

3D mudeli näol on tegu punkti võrgustikuga ehk *mesh* inglise keeles. Võrgustikus paiknevaid punkte nimetatakse *vertex*-iteks, mis on oma olemuselt punktid või positsioonid 3D ruumis. Punktide omavahel ühendamisel tekivad servad ehk *edge*-d. Serv ühendab alati sirgjooneliselt kahte punkti. Ühe serva loomiseks peab eksisteerima vähemalt kaks punkti ning punktist väljuvate servade hulk ei ole piiratud. Servade ühendamisel tekitatakse tahud ehk hulknurgad ehk *face*-d teise nimega *polygon*-id, mida kasutatakse mudelites nähtavate 3D pindade loomiseks. Ala, mis asub kolme või rohkema hulknurga vahel ja on igast küljest ümbritsetud servaga, defineeritakse tahuna. Seega on ühe tahu loomiseks vajalik vähemalt kolme punkti ja kolme serva olemasolu. Hulknurgad võivad aga koosneda suvalisest hulgast punktidest ja servadest [7].

3D objektid koosnevad kolme liiki hulknurkadest. Lihtsam hulknurk on kolmnurk, see koosneb vaid kolmest servast ja punktist. Tähtsuselt teine kujund on nelinurk, mis on lihtsalt jaotatav kaheks kolmurgaks. Arvuti töötleb kõiki tahkusid jagades need esmalt kolnurkadeks. Sellist tegevust nimetatakse *tessellation*-iks ehk mosaiigistamiseks, millest räägitakse töö peatükis 2.1.2. Kolmas kõige keerukam hulknurk on *n*-nurkne, mida on arvutil raskem töödelda, sest selle kolmnurkadeks jagamine on keerukam. Seetõttu on mõistlik 3D objektides *n*-nurkseid hulknurki kasutada võimalikult vähe (*n* on positiivne neljast suurem täisarv).

2.1.1. 3D mudeli geomeetria piirangud

Modelleeritav 3D geomeetria peab olema eksportimiseks kehtiv. Kehtivus tuleb esile objekti eksportimisel Autodesk Mayas-st mängumootorisse. Halva geomeetria puhul lükkab tarkvara ekspordi tagasi ja annab veateate. Eksporti pole võimalik teostada seni, kuni vead on parandatud. Vigade vältimiseks tuleb järgida lihtsaid reegleid:

- iga punkt ja serv peab olema osa mõnest hulknurgast,
- ühe servaga saab ühenduses olla ülimalt kaks hulknurka,
- kehtiva geomeetria puhul näitavad kõik pinnanormaalid ühte suunda,
- hulknurk ei tohi murduda üle iseenda.

Ülalnimetatud reeglite loend ei ole lõplik, välja on toodud kõige olulisemad. Järgides reegeleid ja viies *n*-nurksete (*n* on positiivne neljast suurem täisarv) hulknurkade arvu miinimumini, kiirendatakse töövoogu ja vähendatakse vigade parandamisele kuluvat aega [7].

2.1.2. Mosaiigistamine

Kõik 3D objektid renderdatakse ehk formuleeritakse virtuaalses maailmas. See toimub vastavas tarkvaras, mida nimetatakse renderiks. Virtuaalse Raekoja Platsi puhul täidab renderi ülesannet Cryengine V [8] ehk mängumootor. Formuleerimine toimub mängumootoris reaal-ajas ja pidevalt, vastavalt kasutaja liikumisele. Parema ja kiirema formuleerimise jaoks on vajalik kõigi 3D objektide mosaiigistamine ehk *tessellation*. Selle käigus jagab render kõik hulknurgad kolnurkadeks. Kolmnurkade arv sõltub sellest, kui kaugel formuleeritavast objektist kasutaja 3D-ruumis viibib. Virtuaalruumis kaugelt vaadates võib render kasutada vaid väikest arvu kolmnurki, kolmnurkade hulk võib ulatuda paari sajani. Objektile lähemale liikudes suureneb pidevalt kolmnurkade arv ja jõuab maksimumini siis, kui kasutaja seisab täpselt selle ees. Olenevalt objekti detailsusest võib kolmnurkade arv ulatuda mitme tuhandeni [9].

Mosaiigistamisel on oluline osa ka tekstureerimisel. Tänu sellele on võimalik luua 3D pindadel pärismaailmale väga ligilähedasi tekstuure ja pinnastruktuure. Mosaiigistamist tekstureerimise seisukohast käsitletakse lähemalt peatükis 3.4.1.

2.2. 3D modelleerimine Blender tarkvaras

Blender on vabatarkvaraline 3D-graafika programm animatsioonide, visuaalsete efektide, 3D mudelite printimise, 3D kunsti ja arvutimängude loomiseks. Arendatud mittetulundusühingu Blender Foundation poolt GNU Üldise Avaliku Litsentsi all. Programm võimaldab luua kolmemõõtmelisi objekte ning neid renderdada ja animeerida. Tänu programmi laiale funktsionaalsusele võimaldab see muuta 3D modelleerimise töövoo peaaegu täielikult, kuigi teatavate piirangutega, vaid ühe programmi keskseks [4].

Blender osutus valituks kolmel põhjusel. Esiteks meie juhendajate soovituse põhjal, kes on ise seda kasutanud. Teiseks on varasemates projektides valdavalt kasutusel olnud Blender. Kolmandaks põhjuseks on Blender vabavaraline tarkvara, madalate jõudlusnõuetega ning installeeritav nii Windowsi kui ka MacOSX-i seadmetesse.

Projekti töö käigus ilmnesid Blenderil olulised puudused. Nimelt olime otsustanud mängumootorina kasutada Cryengine 5, mis ei toeta 3D objektide eksporti Blenderist. Internetist on allalaetavad eksporterid, mis on arendatud kolmandate osapoolte poolt, kuid mille kvaliteet ja efektiivsus ei olnud meie jaoks rahuldavad. 3D modellerimistarkvaradele on omane pikk õppeprotsess, mistõttu jäid projektis samaaegselt kasutusele Autodesk Maya ja Blender. Seetõttu võtsime kasutusele paralleelselt teise modelleerimistarkvara Autodesk Maya. Antud tarkvarale on loodud Cryengine-i arendajate poolt pistikprogramm, mis võimaldab objekte efektiivselt transportida mängumootorisse. Blenderit kasutasime Raekoja platsi hoonete ja kõrvalelementide (toolid, laternad, prügikastid ja sildid) loomiseks. Maya-t rakendati esmalt objektidele tekstuuride lisamiseks, objektide transportimiseks mängumootorisse ning projekti hilisemas staadiumis kõnniteede modelleerimiseks.

Blender ja Autodesk Maya sobivad eelkõige majade ja muude kandilise geomeetriaga mudelite loomiseks. Võimalikult tõetruu keskkonna saavutamiseks on vaja kasutada ka olemuselt kumeraid objekte.Virtuaalses maailmas on vajalik kasutaja liikumise piiramine alades, mis väljuvad rakenduse eesmärgilise kasutamise alast. Tallinna Raekoja platsi projektis tekkis vajadus selliste objektide järele, kui oli vaja piiritleda alad, kus kasutaja saab ringi liikuda ja alad, kuhu ei ole võimalik pääseda. Kasutajale parema kogemuse pakkumise huvidest lähtudes on mõistlik need alad piiritleda rakendusesiseste objektidega, millest ei ole võimalik edasi liikuda. Virtuaalse Raekoja platsi rakenduses kasutatakse nende objektidena betoontuve, mis eksisteerivad ka füüsilises maailmas (Lisa 1). Betoontuvide modellerimiseks tuli kasutusele võtta kolmas eraldiseisev 3D modelleerimistarkvara Autodesk Mudbox. Autodesk Mudbox on suunatud kumerustega 3D mudelite loomiseks.

Programmi paralleelina füüsilisest maailmast võib välja tuua savi voolimise, mille jäljendamisega tuleb Autodesk Mudbox hästi toime. Autodesk Mudbox ei ole siiski ainuke sellelaadne tarkvara. Antud juhul langes otsus Mudboxi kasuks, kuna see on arendatud ettevõtte Autodesk poolt ning kasutades paralleelselt nii Mudboxi kui ka Mayat, tekib nende vahel ühtne süsteem, mis võimaldab lihtsat objektide mõlemasuunalist transporti.

2.2.1. Blenderi funktsioonid ja tööriistad

Antud peatükis kirjeldan põhilisi Projektis kasutatud Blenderi funktsioone geomeetria lisamiseks ja selle manipuleerimiseks.

Baasfunktsioonid

Esmased kolm baasfunktsiooni objekti manipuleerimiseks 3D ruumis on *move*, *rotate* ja *scale*. *Move* ehk liigutamine on mõeldud ühe või mitme aktiveeritud punkti, serva, tahu või objekti vabaks liigutamiseks mistahes suunal (klahv G)¹. *Rotate* ehk pööramist kasutatakse ühe või mitme aktiveeritud punkti, serva, tahu või objekti pööramiseks X-,Y-, või Z-telje suunas (klahv R). *Scale* ehk mastaabi muutmist kasutatakse ühe või mitme aktiveeritud punkti, serva, tahu või objekti suurendamiseks või vähendamiseks valitud telje või korraga kõigi telgede suunal (klahv S) [10].

Kustutamine

Kustutamist kasutatakse ühe või mitme aktiveeritud punkti, serva, tahu või objekti eemeldimiseks punktivõrgust. Kustutamise alla kuulub ka ühe või mitme aktiveeritud punkti, serva või tahu üheks liitmine. Samuti ühe või mitme aktiveeritud punkti eemaldamine ühelt servalt ilma serva kustutamata ning ka ühe või mitme aktiveeritud serva eemaldamine ühelt või mitmelt hulknurgalt ilma seda kustutamata. Sellist funktsiooni nimetetakse *dissolve* ehk likvideerimiseks. Klahv X [11].

Eraldamine

Punktivõrgustikust on võimalik eraldada punkte, servi, hulknurki või hulknurga kogumeid *separate* funktsiooni abil. Selle funktsiooni tulemusena eraldatakse aktiivne valik praegusest punktivõrgustikust ja muudetakse eraldiseisvaks objektiks. Eraldamine toimub vaid

¹Klaviatuuri kiirklahvikombinatsioon funktsiooni kasutamiseks

muutmise viisis ehk edit modes. Klahv P [12].

Geomeetria lisamine

Geomeetriale punktide, servade ja hulknurkade lisamine toimub vaid muutmise viisis ehk *edit modes* [13].

Extrude

Põhiline punktide juurde tekitamine toimub *extrude* ehk väljasurumisfunktsiooni abil, mis kopeerib ühe või mitme aktiveeritud punkti või serva, liigutab seda mööda soovitud telge ja ühendab duplikaadi servad ja punktid algsete servade ja punktidega. Blender visualiseerib funktsiooni välja sikutamisena või sisse lükkamisena. Tahu väljasurumisel sikutatakse tahku ümbritsevaid servi soovitud suunas ja algse tahu kuju jääb samaks. Kasutades *extrude* funktsiooni kuubi ühel tahul ühe telje suhtes ühe korra, saame algse 8 punkti asemel 12 punkti. *Extrude* funktsiooni on mõistlik kasutada vaid juhtudel, kui geomeetriale on vaja juurde tekitada punkte. Geomeetria suurendamiseks või vähendamiseks X-, Y-või Z-teljel on kohane kasutada vaid mastaabi vähendamise või suurendamise funktsiooni *scale*. Suurendades *extrude* abil, saame küll sama tulemuse, kuid suurema punktide arvu. Geomeetria mastaabi vähendamiseks *extrude* funktsiooni kasutada vid siste siste siste siste siste või mitut punkti või serva tahu või objekti sisse, jääb objekti pindala siiski samaks või suureneb. Klahv E [14].

Loop Cut

Loop cut ehk silmuse funktsioon lisab igale servale juurde ühe punkti ja ühendab need omavahel servadega. Antud funktiooni kasutatakse nii horisontaalselt kui ka vertikaalselt ning korraga on võimalik lisada suvaline arv silmuseid. Modelleerimisel kasutatakse silmuste abil loodavat geomeetriat akende ja uste asukohtade määramiseks ja hiljem nende detailsemaks loomiseks. Silmuse funktiooni tulemusena tekivad igale algse ristküliku küljele võrdsete suurustega hulknurgad, mis tavapäraselt on kas ruudud või ristkülikud. Iga ristküliku küljel olevate hulknurkade arv ja nende mõõtmed on võrdsed ainult vastasküljel olevatega (v-a juhul, kui tegu on kuubiga). *Loop cut* funktsiooni abil on võimalik paika panna esmane detailide raamistik ning erinevate fassaadi tekstuuride algus- ja lõpp- punktid. Klahvi kombinatsioon CTRL+R[15].

Inset

Inset ehk lisatüki lisamise funktsioon lisab aktiveeritud ühele või mitmele tahule vahetükkidena uued hulknurgad. Tükkide paksus ja sügavus on reguleeritav. Lisaks tuleb ära määrata, kas peale funktsiooni täitmist jäävad aktiveerituks juurdetekitatud vahetükid või esialgne mõõtudelt vähenenud hulknurk. Kahe või enam kõrvuti asetseva hulknurga puhul tuleb määrata, kas soovitakse säilitada algsed ühised servad või mitte. Klahv I [16].

Bridge Edge Loop

Bridge edge loop ehk servadest moodustunud silmuse ühendamine teise samalaadse silmusega. Mõlema silmuse kõik servad ühendatakse omavahel hulknurkadega. Funktsioon annab kõige parema tulemuse juhul, kui mõlemas silmuses on võrdne arv servi. Muudel juhtudel võib tulemus mitte vastata soovitule. Tööriista leidmiseks on vajalik liikuda: Mesh
► Edges ► Bridge Edge Loops [17].

Bevel

Bevel ehk kumeruse lisamise funktsiooni abil on võimalik lisada punktidele, servadele või hulknurkadele kumerust külgnevate hulknurkade suhtes. *Bevel* lisab punkti, serva või hulknurga asemele kindla hulga uusi punkte, mis on omavahel ühenduses hulknurkadega. Selle tulemusena tekib kumer- või kaldserva efekt. Klahv CTRL+B servale või hulknurgale rakendamisel ning CTRL+SHIFT+B punktile rakendamiseks [18].

Knife tool

Knife tool ehk lõike tööriist võimaldab kasutajal jagada hulknurga kaootiliselt talle vajalikeks osadeks või väiksemateks hulknurkadeks. Klahv K [19].

Subdivide

Subdivide ehk osadeks jaotamise tööriist poolitab aktiveeritud hulknurgad ja servad. Poolitamise käigus lisatakse hulknurgale kasutaja poolt määratud hulk servi ja vastav hulk punkte. Subdivide tööriista leidmiseks on vajalik liikuda: Tool Shelf ► Tools ► Mesh Tools ► Add: Subdivide [20].

Antud nimekiri Blenderi tarkvara tööriistadest ja funktsioonidest ei ole lõplik. Välja on toodud vaid need, mis leidsid Projektis kasutust.

2.2.2. Hoone modelleerimine

Virtuaalse Raekoja platsi näol on tegemist füüsilises maailmas eksisteeriva asukohaga, seetõttu oli esmatähtis silmnähtav sarnasus reaalsete hoonete ja kõrvalelementidega. Objektide loomisel VR-rakenduse joaks on vajalik detailsem modelleerimine ja suurem sarnasus tegeliku maailmaga, et rakenduse kasutajal tekiks reealne tunne nagu ta viibiki tõelises keskkonnas. Töö käigus tuli silmas pidada ajafaktorit ja mängumootori piiranguid.

Kõik modelleeritud hooned on loodud *high-polygon* põhimõttel ehk kõrge detailsuse astmega. Kõrge detailsuse saavutmaiseks on vajalik tavapärasest suurem punktide ja hulknurkade arv. Siin kehtib ka seos punktide arvu ja 3D mudeli faili mahu osas. Mida kõrgem on detailsus, seda suurem on punktide arv ning sellest tulenevalt objekti faili maht.

Detailsuse saavutamiseks modelleeritakse esmalt hoone raamistik ning seejärel eraldi hoonel olevad detailid. Hoone raamistiku alla kuulub: üldine hoone kuju, aknad, uksed, luugid ja katus. Selline töö käik võimaldas meil lisaks kõrgele detailsusele tagada ka parema kavaliteedi ja lihtsustas oluliselt tekstuuride lisamist hilisemas töö staadiumis.

Kõik erineva värvi, tekstuuri või materjaliga hoone pinnad eraldatakse modelleerimise käigus või lõpus eraldi detailideks. Suurema objekti väiksemateks tükkideks lammutamine lihtsustab tekstuuride muutmist, sest sellisel juhul puudub vajadus täpselt piiritletud tekstuuride järgi ning sisuliselt saab kogu UV-kaardistamise töö automatiseerida. UV-kaardistamist kirjeldan detailsemalt oma töö peatükis 3.1.

Parema selgitamise huvides kirjeldan kõrgearvulise hulknurkse hoone modelleerimist algusest lõpuni ühe hoone näitel. Modelleerimise töövoogu kirjeldan Kullassepa 9 hoone näitel, mis füüsiliselt ei asu Raekoja platsil, aga on siiski oluline virtuaalreaalsusrakenduse seisukohalt. Maja asub Kullassepa tänaval, mis algab Raekoja platsilt. Hoone on platsilt nähtav ning võimalikult tõetruu visuaalse kogemuse loomiseks on selle modelleerimine vajalik, olgugi et kasutajal ei ole võimalik rakenduses selle juurde liikuda.

Hoone modelleerimise alustamiseks tuleb avada Blenderi rakendus. Avakuvalt valitakse uus fail. Seejärel tekib 3D-ruumi kuup, mis on asetatud keskpunktiga X-, Y- ja Z-telje lõikepunkti (Joonis 3). Lisaks kuubile on uues failis valgusti ja kaamera, mis on meie projekti seisukohast ebaolulised – need võib kustutada. Pildil olev kuup kuvatakse kasutajale objekti viisis, kus seda on võimalik suurendada või vähendada kõigi kolme telje suunal. Samuti on objekti viisis võimalik lisada uusi punktikogumeid või olemasolevaid

duplikeerida.



Joonis 3. Kuvatõmmis Blenderi töölauast enne modelleerima asumist

Enne modelleerimise juurde asumist lisame esmalt faili tausta. Taustaks võib olla perspektiivi viidud pilt modelleritavast majast või mõni varem loodud, täpsete mõõtmetega 3D mudel antud hoonest. Selline võte lihtsustab oluliselt mõõtmete tegemist, eriti kõrguse osas. Maja ligilähedane pikkus ja laius on võimalik mõõta mõnelt veebi kaardirakenduselt. Täpse kõrguse ja selle suhtes paiknevate akende, uste ja muude detailide jaoks on vajalik teostada reaalseid mõõtmisi kohapeal. Tausta puudumisel saab hoone modelleerida niiöelda silmamõõdu järgi, kuid sellisel juhul on vajalik hilisem detailide suurendamine, vähendamine või asukoha muutmine.

Kullassepa 9 hoone modelleerimisel kasutasin taustaks 3D Tallinna rakenduses [21] kasutatud 3D mudelit (Joonis 4). Taustmudel imporditakse faili. Kõik 3D Tallinna rakenduses kasutatutud mudelite mõõdud koguti laser-skanneerimise teel - seega on need mõõtmetelt väga täpsed. Toon välja, et 3D Tallinna rakenduses olemasolevat 3D mudelit kasutan vaid hoone analoogse raamistiku modelleerimiseks, ning nendel olevaid hulknurki või objekte ei kopeerita ega kasutata Virtuaalse Raekoja platsi rakenduses nähtaval hoonel. 3D Tallinna rakenduses olemasoleva 3D mudelile kasutan lisaks Google Street View [22] rakendusest kättesaadavat fotomaterjali (Joonis 5). Eeskätt hoone detailsuse tagamiseks, kuid hilisemas töö faasis valminud mudeli osadeks jaotamiseks.



Joonis 4. Kuvatõmmis Blender töölauast koos taustmudeliga



Joonis 5. Modelleerimise abimaterjalina kasutatud foto

Kuubile geomeetria lisamiseks tuleb liikuda Tab klahviga muutmise viisi ehk *edit mode-*i. Esimese sammuna suurendatakse kuupi mööda X- (punane nool) ja Z-telgi (sinine nool), kuni saavutatakse soovitud esmane suurus ehk maja karp, mis on oma kujult ristkülik. Hoone sügavuse suurendamine mööda Y-telge (roheline nool) toimub tavaliselt modelleerimise viimases etapis. Detailide paiknemine kõrguse suhtes viiakase kooskõlla hiljem. Töö maja raamistiku modelleerimisel toimub põhiliselt *wireframe* ehk sõrestiku vaates, kus kuvatakse vaid punktid ja neid ühendavad servad. Sõrestiku vaatesse ja sealt välja saab liikuda Z klahviga. Järgmiseks jagatakse ristkülikul paiknevad hulknurad väiksemateks osadeks *edge cut* ehk silmuse funktsiooni abil (Joonis 6). See võimaldab korraga modellee-

rida kõik sama sümmeetriaga aknad, uksed ja muud olulised detailid. Esmane akende ja uste paiknemine kõrguse suhtes ei ole oluline. Algselt on tähtis hoida kõikide hulknurkade pindalad, kus hakkavad paiknema samade mõõtmetega detailid, võrdsed.



Joonis 6. Kuvatõmmis silmuse funktsiooni rakendamisest

Antud hoone eripäraks on asjaolu, et esimesel korrusel paikneb lisaks uksele ka kaks akent, aga ülejäänud kolmel korrusel vaid kaks akent. Seega tuleb lisada esimesi korruseid katvatele hulknurkadele lisasilmused, mis jagavad need omakorda pooleks. Antud juhul ei ole mõistlik lisada silmuseid tervele raamistikule, sest tekiksid ebavajalikud punktid. Sellisel puhul peidetakse *hide* ehk peitmisfunktisooniga kõik need hulknurgad, kus puudub vajadus lisaservade järele (Joonis 7). Peitmisfunktsioon: klahv H. Peidetud hulknurkade tagasikuvamiseks kasutatakse klahvikombinatsiooni ALT+H.



Joonis 7. Kuvatõmmis peale peitmisfunktsiooni kasutamist

Järgmiseks määratakse ära akende ja ukste suurused kasutades *inset* ehk lisatükkide lisamise funktsiooni. Korraga lisatakse lisatükid vaid nendele hulknurkadele, kus asuvad samade mõõtmetega aknad. Antud hoone puhul eksiteerib kolme tüüpi aknaid, seega tuleb funktsiooni kasutada iga tüübi puhul eraldi. Lisaks tuleb funktsiooni rakendada ka ukse puhul, kuid esmalt tuleb ühendada ukse asukohal olevad kaks hulknurka üheks. Selle jaoks eemaldatakse neid eraldav serv *dissolve edge* funktsiooniga, mis kuulub kustutamise funktsiooni alla. Vahetükkide lisamise funktsiooni puhul jäävad aktiivseks hulknurgad, mis on ümbritsetud juurde tekkinud hulknurkadest. Aktiivseid hulknurki tuleb suurendada mööda X- ja Z-telge kuni saavutatakse soovitud akna või ukse suurus. Tuleb silmas pidada, et hulknurk peab katma ka fassaadist väljaulatuva aknaraami osa, mis hiljem väljasurumise funktsiooni abil esile tõstetakse. Nüüd korrigeeritakse akende ja ukse paiknemist kõrguse suhtes nihutades neid Z-telge mööda üles või alla. Samuti kohendatakse esimese korruse mõttelist kõrgust fassaadis (Joonis 8), kuna selle värv erineb ülejäänud fassaadi värvist.



Joonis 8. Kuvatõmmis hoone akende, ukse ja eri fassaadi osade asukoha määramisest

Järgnevalt kasutatakse *extrude* ehk väljasurumisfunktsiooni akende või neid ümbritsevate raamide fassaadil esiletõstmiseks või fassaadi sisse surumiseks. Ukse ning esimese ja viimase korruse akende puhul tuleb funktsiooni kasutada vaid korra. Ülejäänud akende puhul ulatuvad fassaadist välja akanaid ümbritsevad detailsed fassaadi osad. Seetõttu on nende puhul vajalik kasutada lisaks väljasurumisfunktsioonile ka vahetükkide lisamise funktsiooni. Vahetükkide lisamine annab võimaluse igal korral muuta sissesurutav hulknurk mõnevõrra väiksemaks tekitades niinimetatud trepi efekti. Mõlemat funktsiooni tuleb rakendada korduvalt, et saavutada silmnähtav sarnasus reaalse hoonega (Joonis 9).



Joonis 9. Kuvatõmmis aknaraamide modelleerimisest

Sissesurumise funktsiooni kasutatakse ka hoone karniiside modelleerimiseks. Sedapuhku on vaja kasutada vaid *extrude* funktsiooni, kuid siingi korduvalt Z- ja Y-telje suunas. Nii tekitatakse hulknurk kõigepealt Z-telje suunas ning seejärel järgmine Y-telje suunas. Tegevust korratakse kuni saavutatakse soovitud tulemus. Katuse karniisi loomiseks peavad olema aktiivsed servad, millede külge uued hulknurgad lisatakse (Joonis 10).



Joonis 10. Kuvatõmmis hoone karniisi modelleerimisest

Seejärel aktiveeritakse kõik maja tagumisel küljel olevad hulknurgad ja liigutatkse Y-telje suunaliselt kaugemale. Seeläbi suureneb maja raamistiku sügavus.

Majale katuse lisamiseks on vajalik kasutada väljasurumise funktsiooni *extrude*, et suurendada maja kõrgust ja ühtlasi tekitada juurde vajalik hulk punkte. Väljasurutud osa kõrgus peab olema võrdne karniisi kõrgeima servaga (Joonis 11).



Joonis 11. Kuvatõmmis maja kõrguse suurendamisest

Tekkinud tasapinna karniisipoolsed servad ja ääred on vaja ühendada karniisi servadega. Selle jaoks kasutatakse hulknurga juurde tekitamise ehk *make face* funktsiooni. Esmalt tuleb aktiivseks muuta kõik servad, mis hakkavad juurdeloodavat hulknurka ümbritsema. Parima tulemuse saavutamiseks on mõistlik luua iga loodava hulknurga servad eraldi ehk luua soovitud hulknurgad ükshaaval (Joonis 12).



Joonis 12. Kuvatõmmis servadevahelise hulknurga loomisest

Tulemuseks saadakse neljast hulknurgast koosnev katus. Viilkatuse loomiseks on vaja sikutada kahte serva, mis eraldavad katuse fassaadi poolt vaadates pooleks ja mööda Z-telge üles. Katuse kõrgus võetakse taustmudelilt. Saavutamaks kaldkatust maja fassaadi poolses küljes, tuleb muuta pikem serv mitteaktiivseks ja sikutada lühemat mööda Y-telge katuse mõttelise keskpunkti suunas (Joonis 13). Katuse modelleerimise lõppemisega lõpeb ühtlasi ka vajadus taustmudeli järgi ning see kustutatakse.



Joonis 13. Kuvatõmmis katuse korrigeerimisest

Modelleerimise järgmises etapis lisatakse majale puuduolevad väiksemad detailid : aknaraamid, aknakarniisid, esimese korruse karniisid, ukse lävi, katuse vintskapp ja elektrikilp.

Aknakarniise ja vintskappi antud hoonel eraldi ei modelleerita, kuna need objektid esinevad mitmetel Projekti hoonetel. Sellisel juhul kasutatakse juba varem loodud sarnaseid objekte ja modifitseeritakse neid vastavalt hoonete eripäradele. See võimaldab modelleerimiseks kuluvat aega tunduvalt optimeerida. Kõik sellised objektid imporditakse hoone faili eraldi (Joonis 14).



Joonis 14. Kuvatõmmis objektide impordi tulemusest

Sobitamaks imporditud objekte olemasolevasse punktivõrku, tuleb kasutada Lõike tööriista

katusesse lisa hulknurkade tekitamiseks. Samuti on vajalik vintskapi katuse kaldenurga muutmine võrdseks majakatuse kaldenurgaga (Joonis 15). Järgmiseks imporditakse faili teise korruse akende kohal olevad karniisid, neid korrigeeritakse X-telje suunas laiemaks ning lisatakse Z-telje suunas kõrgust. Viimase objektina imporditakse esimese korruse kõrgem karniis, mis viiakse vastavusse maja laiusega, vähendades seda X-telje suunas (Joonis 16). Viimane puuduolev karniis esimesel korrusel modelleeritakse, sest tegu on mõõtmetelt ja detailsuselt väikese objektiga. Objekt lisatakse objekti viisis kohe eraldi objektina, et vähendada hilisemat tükkideks jagamisele kuluvat aega.



Joonis 15. Kuvatõmmis katuse kaldenurga korrigeerimisest



Joonis 16. Kuvatõmmis karniisidest

Nüüd asutakse modelleerima aknaraame. Tegu on objektidega, mida hakkab katma fassaadist erinev tekstuur, seetõttu eraldatakse esimese sammuna kõik aknaid katvad hulknurgad praegusest punktikogumist eraldi objektiks (kasutades klahvi P). Seejärel kasutatakse *subdivide* ehk osadeks jaotamise funktsiooni, mis jaotab kõik aktiivsed hulknurgad neljaks võrdseks osaks. Teise ja kolmanda korruse akende puhul nihutatakse horisontaalset poolitavat serva Z-telje suunas üles kuni servast ülespoole jäävad hulknurgad suhtuvad sellest allapoole jäävatesse servadesse nagu 1:2 (Joonis 17).



Joonis 17. Kuvatõmmis hulknurkade 1:2 suhtest

Aknaraamide tekitamine toimub analoogselt eelpoolkirjeldatud fassaadil olevate raamide modelleerimisele. Kasutatakse *extrude* ehk väljasurumise funktsiooni aktiivsete osade sisselükkamiseks ja *inset* ehk vahetükkide juurdetekitamise funktsiooni sissesurutava osa vähendamiseks. Eesmärgiks on taaskord tekitada trepi efekt. Jõudnud soovitud tulemuseni, jäävad aktiivseks hulknurgad, mida ümbritsevad mitmeastmelised akanaraamid. Aktiivsed hulknurgad eraldatakse taaskord olemasolevast objektist. Tekkinud uued objektid on akna-klaasid. Aknaraamide modelleerimine lõpetatakse horisontaalse vahepulga lisamisega teise ja kolmanda korruse akendele, tekitamaks efekti, et aknad koosnevad kuuest ruudukesest (Joonis 18).



Joonis 18. Kuvatõmmis modelleeritud akendest

Eelviimase detailina lisatakse esimese korruse vasaku akna alla elektrikilp. Selle jaoks kasutatakse taaskord lõike tööriista. Esmalt tekitatakse ristkülikukujuline hulknurk, kohendatakse seda vastavasse suurusesse ning siis eraldatakse kahemõõtmeline ristkülik eraldi objektiks. Seejärel liigutakse läbi objekti viisi elektrikilbi muutmise viisi, kus kasutatakse väljasurumisfunktsiooni, et tõsta see ümbritsevast seinast esile (Joonis 19).



Joonis 19. Kuvatõmmis elektrikilbi modelleerimisest

Viimase detailina lisatakse uksele aste. Astme lisamine toimub kohe uue objektina objekti viisist, kuid esmalt viiakse muutmisviisis 3D ruumi lähtepunkt ukse alumise serva keskele. See võimaldab tekitada uue objekti juba algselt õigesse asukohta. Lisatav objekt on kuup,

mille mõõtmeid manipuleeritaks muutmaks selle kõrgus, laius ning sügavus sobivaks. Seejärel muudetakse aktiivseks tekkinud risttahuka pealmine tahk ning kasutatakse esmalt väljasurumise funktsiooni, seejärel X- ja Y-telje suunas suurendamist ning siis veelkord väljasurumise funktsiooni, et tekitada algse ristküliku peale Y- ja X-telje suunas suuremate kuid kõrguselt väiksemate mõõtmetega risttahukas. Järgmiseks muudetakse aktiivseks pealmise risttahuka kõik nähtaval olevad horisontaalsed servad. Need servad muudetakse kumeraks *bevel* ehk kumeruse lisamise funktsiooni abil (Joonis 20).



Joonis 20. Kuvatõmmis kumeruse lisamise funktsioooni kasutamisest ukse astmel

Sellega lõpeb ka ühtlasi hoone ja selle detailide modelleerine ja asutakse loodud 3D mudeli väiksemateks eraldiseisvateks objektideks jaotamist.

2.2.3. 3D mudeli osadeks eraldmine

Valminud 3D mudeli jaotamine väiksemateks osadeks Blenderis kiirendab Autodesk Mayas tekstureerimist ja hoone eksporti mängumootorisse. Teise võimalusena saab Autodesk Maya-s tekstuurid lisada valides iga tekstuuri alla jääva hulknurga eraldi. Töövoo poolest on mõlemad võtted sarnased, kuid esimesel juhul tekib 3D ruumi suurem hulk eri objekte. Virtuaalse Raekoja platsi projektis langes valik esimese võtte kasuks kokkuleppeliselt, sest see võimaldas efektiivsemat meeskonnasisest tööjaotust. Nii tegutsedes jaotub töö hulk optimaalsemalt modelleerija, tekstureerija ning mängumootoris keskkonda loova liikme vahel.

Osadeks jagamiseks kasutatakse pildimaterjali füüsilisest maailamst (Joonis 3). Valminud 3D mudel koosneb kümnest eraldisesivast objektist, sest juba modelleerimise käigus jaotati võimalikult suur hulk detaile eraldiseisvateks. Samuti lisati kõik imporditud detailid uute

objektidena. Nüüd eraldatakse kõik hooneraamistiku hulknurgad, mida peab katma üks tekstuur. Antud maja puhul katus, esimese korruse fasaad ning teise kuni neljanda korruse fassaad. Siin lähtutakse põhimõttest: suurema pindalaga hulknurgad eraldatakse väiksema pindalaga hulknurkadest, sest nende aktiivseks muutmine on vähem aeganõudev. Kõik pindalalt väiksemad ja suurema detailsusega hulknurgad ning detailid jäävad algsesse objekti. Nende hulka kuuluvad katusekarniis ning fassaadist väljaulatuvad aknaraamid.

Eraldamiseks muudetakse aktiivseks kõik hulknurgad, mida peab katma esimese korruse puhul roosakas seina tekstuur. Samamoodi eraldatakse ka ülejäänud korruste ning katuse hulknurgad (Joonis 19). Nii jäävad algsesse objekti ainult need hulknurgad, mida hakkab katma valge fassaadi tekstuur.



Joonis 21. Kuvatõmmis 3D mudeli osadeks eraldamisest

Seejärel asutakse eraldama vintsakapi ning karniisi hulknurki. Vintsakapi puhul eraldatakse samuti katus, luuk ning vintskapist väljaulatuv post seintest. Seintel olevatel karniisidel eraldatakse neid katvad hulknurgad. Viimase detailina eraldatakse ukse ava kattev hulknurk. Järgmiseks liidetakse objektid, mida katab sama tekstuur, kuid olid algselt erinevad objektid, üheks objektiks. Antud hoone puhul on nendeks maja katus ning vintskapi katus ja omakorda karniisid ning seejärel hoone valge fassaadi tekstuuriga kaetavad detailid. See võimaldab taaskord kiirendada tekstureerimist. Osadeks jagamise tulemusena tekkis 16 eraldi objekti algse kümne asemel.

2.2.4. 3D mudeli eksport Blender tarkvarast

Viimase etapina modelleerimise töövoos eksporditakse 16 objektist koosnev hoone 3D mudel .fbx failiformaadina [23] Blenderist. Eksportimise näol on tegemist faili salvestamisel kausta .fbx failina. Eksportimine teise faili formaati on vajalik avamaks 3D mudelit Autodesk Maya tarkvaras, mis ei toeta Blenderi baasformaati .blend [24].

FBX failiformaadi kõrval eksisteerib ka teisi sarnast funktsionaalsust pakkuvaid formaate. FBX formaadil ei olnud projekti kontekstis mingeid eeliseid võrreldes teiste analoogsete formaatidega. Otsus kasutada FBX formaati tuli meeskonnasisese kokkuleppena. Antud formaati kasutati virtuaalse Raekoja platsi projektis kõigi 3D mudelite ja objektide eksportimiseks modelleerimistarkvarade vahel.

3. 3D mudeli tekstureerimine

Järgnevas peatükis kirjeldan tekstureerimise kõiki etappe, erinevaid tekstuuride ja materjalide variante ning tekstuuride loomist.

Virtuaalse maailma loomisel on lisaks detailsele modelleerimisele sama tähtsal kohal ka objektide tekstureerimine. 3D pinnad peavad võimalikult ligilähedaselt imiteerima füüsilises maailmas eksisteerivate materjalide pindasid. Pindade värv või värvide spekter ning struktuur peavad olema võimalikult ligilähedased reaalselt olemas olevatele.

Tekstureerimine koosneb neljast osast: UV-kaardistamisest, materjali valikust ja tekstuuri valikust ning selle kaardistamisest UV-tekstuuriruumis. Esmalt luuakse 3D pindade UV-kaardid, seejärel määratakse kõigile pindadele materjalid ning viimasena kaardistatakse 3D mudeli pindadele tekstuurid. Kõik tekstureerimise etapid, välja arvatud tekstuuri failide loomine, toimuvad Autodesk Maya tarkvaras, milles loodud UV-kaardid ning määratud materjalid ja tekstuurid kanduvad ekspordi käigus edasi mängumootorisse.

3.1. UV-kaardistamine

UV-d on kahedimensioonilised tekstuuri koordinaadid, mis vastavad kolmedimensioonilisele punkti geomeetriale. UV-kaardistamine defineerib seega kahedimensioonilise tekstuuriruumi, mida nimetatakse UV-tekstuuriruumiks. UV-kaardistamise tulemuseks on kindlate piiridega UV-kaart. UV-kaardid on hädavajalikud, sest need loovad ühenduse 3D objekti pindade ja nendele kaardistatava tekstuuri vahel [25]. Ruumiliste objektide puhul on UV-kaardistamise eelduseks, et kõik punktidevahelised pinnad 3D objektil on lamedad [7]. Kaardistamine toimub peale 3D mudeli täielikku valmimist ning enne materjali ja tekstuuri määramist. Vastasel juhul, viies mudelisse sisse geomeetrilisi muudatusi, ei kattu enam mudel UV-kaardiga [25].

UV-kaardistamine on kohustuslik osa 3D objektide eksportimiseks mängumootorisse. Kaardistatud peavad olema kõik eksporditava objekti punktid, vastasel juhul lükatakse eksport Autodesk Maya tarkvara poolt tagasi.

UV-kaardistamiseks kasutatakse UV-koordinaatsüsteemi. Kuna kaardistatav tekstuur on kahemõõtmeline, on ka UV-süsteem kahemõõtmeline, kus U tähistab üldjuhul horsontaaltelge ja V vertikaaltelge. Kui kolmemõõtmelises süsteemis kasutatakse telgedel X, Y, Z tähistust, siis UV-kaardistamiseks on kokkuleppeliselt võetud kasutusele erineva tähistusega süsteem. Kõigile hulknurkadest koosnevatele punktikogumitele tuleb määrata UV-koordinaadid. UV-kaardistamise käigus määratakse igale punktile ehk *vertex*-ile ühene UV-koordinaat. Iga 3D objekti iga punkti UV-koordinaat markeerib ära selle punkti asukoha tekstuuril või tekstuurikaardil. See võimaldab hiljem punkti koordinaatide asukohad viia vastavusse koordinaatidega tekstuuril või tekstuurikaardil [7].

Ruumilise objekti UV-koordinaatide kaardistamiseks tekstuurile või tekstuurikaardile on vajalik moodustada 3D objekti hulknurkade pinnalaotus. Pinnalaotuse moodustamiseks kasutatakse erinevaid viise vastavalt tekstureeritava objekti eripäradele. Kuna 3D objekt koosneb üldjuhul mitmest väiksemast objektist, on vajalik moodustada igale objektile eraldiseisev pinnalaotus ja seeläbi ka UV-kaart.

UV-kaartide valmistamist võimaldavad kõik Projektis kasutatud 3D modelleerimistarkvarad. Üldjuhul on Virtuaalse Raekoja platsi Projektis objektide UV-kaardistamiseks kasutatud Autodesk Maya tarkvara, sest mudelite mängumootorisse import toimus selle abil. UV-tekstuuriruumi kaardistamine on üldjuhul tarkvarapoolne, kus kasutaja peab vastavalt 3D mudeli eripäradele tegema kaardistamisviisi valikud. Kõige sagedamini Projektis kasutatud viis on lasta tarkvaral automaatselt genereeride UV-kaart, mis on ajaliselt ka kõige efektiivsem. Automaatsel UV-kaartide genereerimisel valib tarkvara ise hulknurkade paigutuse kaardil ja sellest tulenevalt ka selle, millised hulknurgad jäävad omavahel ühendusse ning millised muutuvad kaardi piirideks. Küll aga on võimalik, et kasutajal tuleb hiljem tekkinud UV-kaarti pöörata või selle mõõtmeid muuta. Antud kaardistamise viisi saab kasutada juhtudel, kui tekstureeritava 3D pinna tekstuur on terve pinna ulatuses samasugune ja nurkade ning äärte kattuvus pole oluline. See viis leidis virtuaalse Raekoja platsi Projektis enim kasutust, sest hoonete 3D mudelite pinnad vastasid nendele nõuetele [26].

Teise Projektis kasutatud lahendusena tuleb pinnalaotus moodustada kasutajapoolse abiga nii, et maksimaalne hulk hulknurki jääb omavahel ühendusse. Selle jaoks on vajalik 3D objektil käsitsi määrata äärised, mis ei jää ühendusse ühegi hulknurgaga. Need äärised muutuvad ühtlasi UV-kaardi piirideks [7]. Selline töö käik on aeganõudvam kui eelpoolmainitu ja võetakse kasutusele juhtudel, kui pindadele kantav tekstuur peab ületama pindadevahelisi servi ja nurki nii, et ei tekiks visuaalselt ebaloomulikku efekti. Antud töö käik leidis rakendust erandjuhtudel, sest objekte, mis nõudsid seda tüüpi lähenemist, esines minimaalselt.

3.2. Materjalid

Materjal esindab pinna tüüpi, kuhu hiljem kaardistatakse tekstuur. Materjal reguleerib 3D pinnal valguse käitumist, eeskätt materjali läikivust ning sellel valguse neelduvust ja sellelt valguse peegelduvust [27]. Virtuaalse Raekoja platsi Projektis kasutati kolme eri tüüpi Autodesk Maya materjale: Lambert, Blinn ja Phong.

Lambert

Lambert on materjal, mida kasutatakse matt või tuhmi pinna puhul, millel puudub läige. Lambert on Projektis kõige sagedamini kasutatav materjal. Seda kasutati kõikide kivi- ning puitpindade imiteerimiseks [28].

Blinn

Blinn on materjal, mida kasutatakse kergelt ja pehmelt läikivate pindade loomiseks. Blinn sobib kõige paremini metalse efekti loomiseks. Projektis kasutati antud materjali metall- ja klaaspindade puhul [29].

Phong

Phong on materjal, mida kasutatakse intensiivsema läike saavutamiseks. Phong sobib läikivate metallobjektide, klaasi ning keraamika imiteerimiseks. Projektis kasutati antud materjali harva, sest kõrgläikega objektide arv oli väike [30].

3.3. Tekstuurid

Tekstuur esindab visuaalset kahemõõtmelist pinnastruktuuri ja värvuse imitatsiooni füüsilisest keskkonnast. Tekstuuri näol on tegu üldjuhul kahemõõtmelise pildiga, mis kaardistatakse UV-kaardi abil 3D pinnale. Tekstuur tuleb pildistada, mõnest veebikeskkonnast alla laadida või graafikatarkavara abil luua. Lisaks pildile võib tekstuuriks olla ka lihtsalt värv.

Suurte 3D pindade, nagu näiteks seinad, trepid, katused ja kivisillutised, tekstureerimisel asetatakse ühte tekstuuri objektile korduvalt. See tähendab, et tekstuur asetatakse 3D pinnale malelaua põhimõttel ehk tekstuuri faili korratakse nii laiuse kui ka kõrguse suunaliselt. Kui malelaual eksisteerivad nii valged kui ka mustad ruudud, siis tektuureerimisel käitub tekstuur mõlema värvusena. Tekstuur tuleb pinnale kaardistada nii, et kõik ääred kattuksid omavahel, ei jääks vahesid ja tekstuur ei ulatuks üle iseenda. Suurtel 3D pindadel peab tekstuur katma pinda ühtlaselt, ilma et oleks näha liitekohti tekstuuri tükkide vahel või nende kordumist. See tähendab, et kaardistatav tekstuur peab olema töödeldud või loodud nii, et kõik tekstuuripildi ääred ühtiks omavahel veatult. Antud tulemuse saavutamiseks kasutati projektis kahte meetodit: loodi tekstuuri imitatsioon Adobe Photoshop [31] abiga või töödeldi olemasolev pilt sama programmiga nõuetele vastavaks.

Väikesemõõtmeliste või rakenduse kasutaja suhtes kõrgel asetsevate objektide, nagu näiteks käepidemed, katuseservad, akna- ja ukseraamid, tuulelipud, puhul on tekstuuri täpne ja ühtlane sobitamine mõnevõrra vähem tähtis, ning sageli kasutati nende tekstureerimiseks lihtsalt värvi.

Virtuaalse Raekoja platsi Projektis esinesid kõige sagedamini kivipinnad, mille alla kuuluvad erinevad fassaadid, katused, tänavasillutised ja trepid. Vähemal määral leidus metall-, puit- ja klaaspindu, kuid milledel oli siiski oluline koht keskkonna loomisel.

Kõigi virtuaalsete 3D pindade tekstuuride töötlemiseks ning samuti kivi-, puit- ja metallpindade imitatsioonide loomiseks kasutati Adobe Photoshop tarkvara. Lihtsamate graafiliste kujutiste, näiteks tänavate nimesiltide ja majanumbrite, loomiseks kasuati Adobe Illustartor tarkvara [32]. Lisa 2 on välja toodud Adobe Illustratoris projekti tarbeks loodud tänava nimesilt ja üks majanumber.

3.3.1. Nõuded tekstuuridele

Tekstuuridele esines projektis kahte liiki nõudeid: meeskonna poolt püstitatud ning tehnilised, meeskonnast mitte sõltuvad. Meeskond püstitas nõuded eelkõige pildi visuaalsuse ja pildifaili suuruse kohta. Eesmärk oli lihtsustada tekstureerimise töömahtu ning samuti, et virtuaalse keskkonna lõpptulemus oleks visuaalselt võimalikult meeldiv ja kasutajasõbralik.Tehniliseks nõudeks võib lugeda pildi failiformaati, mida toetab mängumootor. Kõigist nõuetest juhinduti projektis vajalike tekstuuride töötlemisel või loomisel.

Meeskonna poolt esitatud nõuded

1. Tekstuuri esindav pildifail peab olema töödeldud selliselt, et see on jätkatav igast pildi küljest sama pildifailiga ehk korrates pilti nii kõrguse kui laiuse suunaliselt, ei tohi pildi kordumine olla märgatav.

2. Pildil ei tohi olla selgelt nähtavaid ääri ega defkte.

3. Pildil ei tohi olla selgelt eristuvaid heledamaid ega tumedamaid kohti.

4. Pildil ei tohi esineda varje.

5. Pildi kõrgus peab võimalusel olema võrdne pildi laiusega või vastupidi. Mõõtühikuks on piksel ning soovitatav pildi suurus on kas 512x512 pikslit või 1024x1024 pikslit ning ühetoonilise värvi puhul 16x16 pikslit või 32x32 pikslit.

6. Pildifaili maht peab jääma alla 20MB, et tagada tervikrakenduse minimaalne maht.

Ülalnimetatud nõuete 1-5 jälgimine ei olnud oluline pildistatud tekstuuride puhul, mida

kasutati hoonete uste ja kunstiliste elementidena.

Tehnilised nõuded

1. Tekstuuriks olev pilt peab olema TIFF [33] failiformaadis.

3.3.2. Tekstuuride loomine

Uute tekstuuride loomise all mõeldakse tekstuuri joonistamist tühjalt lehelt. Selle eesmärgiks on luua füüsilises keskkonnas eksisteerivate tekstuuride imitatsioonid nii, et koheselt on täidetud kõik tekstuuridele esitatud nõuded. Täiesti uute tekstuuride joonistamise eeliseks on paindlikkus, mis võimaldab muuta tekstuuri värvi, struktuuri ja detailsust kiiresti vastavalt vajadusele. Projekti käigus loodi igale pinnatüübile mingi hulk baastekstuure, mida muudeti vastavalt 3D objekti eripäradele. See välistas vajaduse luua igale objektile täiesti uue tekstuuri. Baastekstuurid loodi kivi-, puit-, ja metallpindade jaoks, mida esines projektis kõige enam. Sellised pinnad erinesid teistest omalaadsetest tihtipeale vaid värvi tooni ning vähemal määral struktuuri poolest.

Kõige sagedamini kasutati tekstuuride joonistamiseks Adobe Photoshop tarkvara. Tekstuuri joonistamiseks Adobe Photoshopis tuleb luua uus fail. Uue faili loomisel määratakse ära pildi mõõtmed ja selle algne värv, mis antud juhul on 512x512 pikslit ning valge (Joonis 22).

lew		>
Name: U	Intitled-1	ОК
Preset: Custom	~	Cancel
Size:	~	Save Preset
Width:	12 Pixels ~	Delete Preset
Height:	12 Pixels ~	
Resolution:	2 Pixels/Inch ~	
Color Mode:	tGB Color 🗸 8 bit 🗸	
Background Contents:	Vhite ~	Image Size:
Advanced		768,0K
Color Profile:	RGB IEC61966-2.1 ~	
Pixel Aspect Ratio:	Gquare Pixels 🗸 🗸	

Joonis 22. Kuvatõmmis Adobe Photoshop tarkvaras uue faili loomisest

Värvuselt võimalikult sarnase tekstuuri loomiseks kasutatakse hoone pilti füüsilisest kesk-

konnast. Selleks kasutatakse pilti (Joonis 5), mis oli ka taustmaterjaliks modelleerimisel. Pilt kopeeritakse vastloodud Adobe Photoshop faili ning kasutatakse vaid värvitoonide defineerimiseks, mida tehakse Adobe Photoshopis oleva pipett - tööriista abil [34]. Pipett - tööriist võimaldab määrata pildi suvalise piksli värvi. Esmalt tuvastatakse kujutada soovitud fassaadi osalt võimalikult tume toon, mis määratakse tausta värviks (Joonis 23).



Joonis 23. Kuvatõmmis Adobe Photoshop tarkvarast peale tausta lisamist

Seejärel määratakse pipett - tööriista abil mõni heledam toon ning valitakse pintsel - tööriist [35]. Pintsel - tööriista puhul on oluline valida õige pintsli kuju ja määrata selle voo sagedus. Pintsliga kaetakse pilti võimalikult ühtlaselt nii, et taust jääb läbi kumama. Seejärel määratakse pildilt pipett - tööriista abil uus, eelmisest veidi erinev toon, ning tegevusi korratakse, kuni tumedam taustvärv on ühtlaselt kaetud. Loodud pildil on tausta toon väga õrnalt nähtav, sellel puuduvad selgelt eristatavad tumedamad ja heledamad kohad ning pilt meenutab fassaadi imitatsiooni (Joonis 24).



Joonis 24. Kuvatõmmis Adobe Photoshop tarkvaras loodud algsest tekstuurist

Äärte kattuvuse nõude täitmiseks kasutatakse nihke - filtrit ehk Offset filtrit [36]. Nihke - filter töötab samal põhimõttel mis tekstuurid 3D pinnal ehk kordab sama pilti kõrguse ja laiuse suunaliselt. Filtri töö põhimõtte parema selgitamise huvides lisatakse pildile nüüd punane ja sinine raam. Pilt (Joonis 25) on toodud enne filtri rakendamist ja pilt (Joonis 26) peale filtri rakendamist.



Joonis 25. Kuvatõmmis enne nihke-filtri rakendamist



Joonis 26. Kuvatõmmis peale nihke-filtri rakendamist

Need pildi alad, kus kohtuvad punane ja sinine värv, on tekstuuri liitekohtadeks 3D pindadel. Liitekohad on vaja töödelda selliselt, et neid ei ole võimalik eristada ülejäänud pildist. Selle jaoks kasutatakse tempel-kloonimise tööriista, mis kopeerib pildilt kasutaja poolt valitud pikslid ning asendab need liitekohtades olevate pikslitega [37]. Kopeeritavate pikslitega hõlmatav ala ei tohi olla tekstuurist mingil moel eristuv. Vastasel juhul ei saavutata soovitud efekti. Nihke-filtrit ja tempel-kloonimist tuleb rakendada korduvalt, kuni pole võimalik eristada liitekohti. Saadud tulemus kattub pildiga (Joonis 24), millele on rakendatud nihke - filtrit, kuid liitekohti eristada pole võimalik. Viimase sammuna salvestatakse pilt valitud nimega TIFF formaadis mängumootori poolt nõutud kausta.

3.3.3. Veebikeskkondadest allalaetavad tekstuurid

Allalaetavad tekstuurid jagunevad laias laastus kaheks: pildi failid, mis on juba töödeldud ja pildid, mis vajavad eelnevat töötlust. Mõlemal juhul on vajalik jälgida pildi litsentsi - mille alusel ja mis eesmärgiks võib allalaetavat faili kasutada ning kas on nõutud viitamine autorile.

Virtuaalse Raekoja platsi Projektis leidsid kasutust pildid, mis olid vabaks kasutamiseks nii era- kui ka kommertsprojektides ning milledel puudus nõue viidata autorile. Projekti käigus võeti eesmärgiks kasutada allalaetavaid tekstuure võimalikult vähe, sest üldjuhul ei andnud need soovitud tulemust. Samuti piiras allalaetavate tekstuuride kasutamist asjaolu,

et paljud veebikeskkonnad, kust võis leida sobivaid faile, olid tasulised.

3.3.4. Tekstuuride ülespildistamine

Pildistamise puhul tuleb pinda pildistada lähedalt, et tabada selle struktuur. 3D pinnale tuleb selline tekstuur asetada väiksemõõdulisena ning seega korrates seda suuremal arvul kordadel, et katta kogu pind. Üldjuhul ei anna pildistamine aga head tulemust, sest välitingimustes pinnale langev valgus ei ole ühtlane ning valguse tekitatud varjud jäävad virtuaalses keskkonnas ebaloomulikud. Samuti vajab pildistatud pilt, enne kui seda on võimalik tekstuurina kasutada, töötlust. Pilti tuleb töödelda nii, et jätkatavuse nõue oleks täidetud ning pildil olevad heledamad ja tumedamad osad ühtlustatud. Vastasel juhul on 3D pinnal näha tekstuuri tüki kordumist.

Pildistatud tekstuure rakendati kõige sagedamini hoonete uste (Joonis 28) ja keerukamate kunstiliste elementide puhul (Lisa 3), sest nende modelleerimine või graafiliselt joonistamine on ajaliselt ebaefektiivne. Mõningatel juhtudel kasutati ülespildistatud tekstuuri ka 3D hoonete fassaadidel, näiteks Tallinna Raekoja hoonel (Lisa 4), mille jäljendamine muul moel osutus keerukaks.

3.3.5. Tekstuuri fotofailide töötlustöövood

Pildistatud või allalaetud tekstuuride juures esines kahte sorti töötluse töövoogu. Esiteks perspektiivkärpimine, mis koosneb perspektiivi muutmisest ja pildi äärte väljalõikamisest. Teiseks aga keerukam töötlustöövoog, mis hõlmab endas lisaks perspektiivkärpimisele ka fotomanipulatsiooni. Seda kasutati vaid juhul, kui tekstuur pidi ühel pinnal korduma. Oma olemuselt nõudis see sarnast lähenemist tekstuuride joonistamisele. Kõik pildifaili töötluse etapid toimusid Adobe Photoshop tarkvaras.

Töövoog 1

Perspektiivi muutmine on oluline, sest üldjuhul ei ole võimalik objekti pildistada seistes selle mõttelise keskkohaga täielikult paralleelselt. Äärte väljalõikamine pole töö seisukohalt oluline, kuid ilma ääri lõikamata ei ole perspektiivi parandamine võimalik. Esmalt avatakse Adobe Photoshopis lähtefail (Joonis 27).



Joonis 27. Kuvatõmmis enne perspektiivkärpimist

Perspektiivi parandamiseks ja äärte lõikamiseks kasutati tarkvaras perspektiivkärpimise tööriista [38]. Tulemuseks peale perspektiivi parandamist on pilt (Joonis 28).



Joonis 28. Kuvatõmmis peale perspektiivkärpimist

Selline töötlusmeetod eeldab, et tekstuuri kasutatakse ühel 3D pinnal vaid korra ehk et see ei pea korduma. Seda rakendati Projektis kõigi uste ja kunstiliste elementide tekstuuride sobivale kujule viimiseks.

Töövoog 2

Esmalt avatakse lähtefoto (Joonis 29), mis on pildistatud Raekoja hoone sammastest, kus esines projekti seisukohast oluline tekstuur – paetellised. Seejärel rakendatakse ka selle pildi puhul perspektiivkärpimist, et eemaldada fotolt ebaoluline info.



Joonis 29. Pilt peale perspektiivkärpimist

Säilitamaks pildil olevate unikaalsete telliste arvu, ei kärbita pildi mõõtmeid enam väiksemaks. Vastasel juhul oleks lihtsasti nähtav tekstuuri kordumine 3D pinnal. Tulenevalt tekstuuri kordumise vajadusest 3D pinnal, tuli ka selle pildi ääred töödelda nii, et liitekohad ei ole muust pildist eristatavad. Selle saavutamiseks kasutati samuti nihke - filtrit (Joonis 30).



Joonis 30. Pilt peale nihke-filtri rakendamist

Nihke - filtrit rakendades on näha, et liitekohad ei kattu. Nende ühtlustamiseks kasutatakse tempel-kloonimise tööriista. Nii filtrit kui ka tööriista rakendatakse korduvalt. Seekord on vajalik täpne kopeeritave pikslite lähtekoha valik, sest oluline on korrigeerida ka telliste ja vuukide kõrgust. Mõningates kohtades on vajalik terve vuugi kopeerimine. Järgmiseks ühtlustatakse sama tööriistaga vuugi ja telliste värvusi. Viimase sammuna muudetakse kogu pildi värvide tasakaalu nii, et vuugid tuleksid selgelt esile ja eristuksid kividest. Tulemuseks on pilt (Joonis 31).



Joonis 31. Lõplik pilt peale fototöötlust

3.4. Tekstuuri kaardistamine Autodesk Mayas

Tekstuuri kaardistamine on tekstureerimise töövoos kõige viimane osa. Tekstuur kaardistatakse UV-tekstuuriruumi ja selle eelduseks on eelnevalt loodud UV-kaart, 3D pinnale on juba määratud materjal ning tekstuuri esindav pildifail või värv.

Tekstuuri kaardistamine toimub tekstuuri UV-kaardi abil. Muudetakse UV-kaardi paiknemist, nurka ja suurust tekstuuri suhtes. Paiknemist ja nurka muudetakse või korrigeeritakse juhtudel, kui tarkvara poolt loodud ja paigutatud UV-kaardid ei paikne UV-tekstuuriruumis soovitult. UV-kaardi suuruse muutmise abil määratakse ära tekstuuri kordumiste arv 3D pinnal. Suurendades UV-kaarti kasvab tekstuuri kordumiste arv.

Igale eraldi seisvale UV-kaardile tuleb tekstuur kaardistada eraldi. Kui kõigile 3D mudeli pindadele on tekstuur kaardistatud, saab objekti eksportida mängumootorisse.

3.4.1. Realistlikuma tekstuuri kaardistamise võimalused

Eelnevas alapeatükis kirjeldatud tekstuuri kaardistamisel jääb tekstuur 3D pinnal kahemõõtmeline, mis paljudel juhtudel ei anna füüsilise maailmaga võrreldavat sarnasust. Põhjuseks on asjaolu, et 3D mudeli hulknurkadest koosnevad pinnad on täiesti siledad, kuid päris maailmas eksisteerivad pinnad on üldjuhul krobelised, ebaühtlsed ning reljeefsed. Siledust on virtuaalsetel pindadel võimalik mingil määral peita ka kahemõõtmelise tekstuuriga, kuid on olemas tekstureerimise meetodid reaalsete või illusoorsete ebaühtluste, krobelisuse ja reljeefsuse loomiseks. Projektis kasutatud meetodeid on kolm ning kõigil juhtudel on nende rakendamiseks vajalik luua tekstuuri failist kindlate nõuetega tekstuurikaardid.

- 1. **Bump map** ehk kõrguse kaart. Kõrguse kaart on halltoonidest tekstuuri kaart tekstuurist, mis kaardistatakse 3D pinnale. Kaardil esindab iga piksel seda, kui palju antud punkti peab esile tooma. Antud juhul luuakse illusioon pinna reljeefsusest, see tähendab, et pinna geomeetriat ei muudeta [39].
- 2. Normal map ehk pinnanormaali kaart. Pinnanormaali kaart on erilist tüüpi kõrguse kaart, mis võimaldab sarnaselt tavalisele kõrguse kaardile lisada pinnale illusoorseid kõrgendusi, madaldusi ja kriimustusi nii, et valgus peegeldub neilt nagu oleks tegu modelleeritud geomeetriaga. Antud juhul on tegu RGB tekstuuri kaardiga tekstuurist, kus iga piksel esindab suunda, kuhu poole pind peab olema suunatud [40].
- 3. **Displacement map** ehk nihke kaart. Nihke kaart on samuti halltoonidest tekstuuri kaart, mis kaardistatakse 3D pinnale. Antud kaardi abil muudetakse 3D pinna geomeetriat tõeliste ebaühtluste ja krobelisuse loomiseks. Reljeefsuse loomine nihke kaartide abil toimub mosaiigistamise meetodil ning mängumootor formuleerib detailsust ja mosaiigistamise käigus tekkivate kolmnurkade arvu reaalajas[41].

Ülaltoodud kaarte võib rakendada kas koos või eraldi (Lisa 5 on toodud näited kõigist kaartidest). Antud kaartide loomine toimub Adobe Photoshop tarkvaras. Virtuaalse Raekoja platsi Projektis kasutati neid meetodeid tänavasillutistel tõetruuma tekstuuri saavutamiseks.

Hoonete fassaadidel neid võimalusi ei kasutatud, sest antud juhul oleks kõigi pindade mängumootori poolne formuleerimine hõivanud liialt palju arvuti põhimälu.

4. Kokkuvõte

Antud bakalaureusetöö eesmärkideks olid omandada teadmised ja oskused VR-rakenduste loomiseks, anda ülevaade 3D modelleerimisest ja 3D mudelite tekstureerimisest ning luua VR-rakendus.

Käesolevas bakalaureusetöös kirjeldatud meetodite ja põhimõtete kasutamise tulemusena valmis Virtuaalse Tallinna Raekoja platsi VR-rakendus, mis on kasutatav HTC Vive riistavaral [42]. Lisa 6 on toodud kuvatõmmised rakendusest.

Bakalaureusetöös anti põhjalik ülevaade Projektis kasutatud 3D mudelite loomisest ühe hoone näitel. Samuti on antud üksikasjalik kirjeldus tekstureerimise töövoost. Valminud rakenduse põhjal võib järeldada, et autor on omandanud vajalikud teadmised ja oskused töös kirjeldatud aspektide kohta VR-rakenduse loomiseks ning seega on täidetud kõik sissejuhatuses püstitatud eesmärgid.

Valminud VR-rakendust on võimalik kasutada Tallinna vanalinna, mis kuulub UNESCO maailmapärandi nimistusse [43], tutvustamiseks. Rakenduses on võimalik kasutajal ringi liikuda ja keskkonda vaadelda, kuid see ei ole pikalt kaasahaarav. Atraktiivsuse tõstmiseks on vajalik lisada rakendusele funktsionaalsust. Projekti on võimalik edasi arendada mitmel suunal. Esiteks on võimalik juurde luua ajastutes rändamise võimalus, kus saab näha, milline nägi Raekoja plats välja erinevatel ajalooperioodidel. Teiseks, laiendades virtuaalset keskkonda Raekoja platsist ja vanalinnast kaugemale, on võimalik seda rakendust kasutada linnaplaneerimisel. Kolmandaks, kuna antud VR-rakendus on loodud CryEngin V mängumootoris, on see lihtsalt edasi arendatav arvutimänguks. Arvutimängud on hõlpsasti levitatavad interneti vahendusel ja asjaolu, et tegevus toimub reaalselt eksisteerivas keskkonnas, võib tõsta turismi huvi Tallinna vastu.

Kasutatud kirjandus

- R. Wang, J. Yao, L. Wang, X. Liu, H. Wang, and L. Zheng, "A surgical training system for four medical punctures based on virtual reality and haptic feedback," in 2017 IEEE Symposium on 3D User Interfaces (3DUI), March 2017, pp. 215–216.
- [2] N. S. S. Hamid, F. A. Aziz, and A. Azizi, "Virtual reality applications in manufacturing system," in 2014 Science and Information Conference, Aug 2014, pp. 1034–1037.
- [3] Re:creation. [Last access: 08.05.2017]. [Online]. Available: http://recreation.ee/
- [4] Blender (software). [Last access: 07.05.2017]. [Online]. Available: https: //en.wikipedia.org/wiki/Blender_(software)
- [5] Autodesk maya. [Last access: 07.05.2017]. [Online]. Available: https://en.wikipedia. org/wiki/Autodesk_Maya
- [6] Autodesk mudbox. [Last access: 07.05.2017]. [Online]. Available: https: //en.wikipedia.org/wiki/Autodesk_Mudbox
- [7] A. Chopine, *3D Art Essentials The fundamentals of 3D Modeling, Texturing and Animation.* Focal Press, Elsevier, 2011, pp 21-23, 36, 153-154.
- [8] Cryengine v manual. [Last access: 10.05.2017]. [Online]. Available: http: //docs.cryengine.com/display/CEMANUAL/CRYENGINE+V+Manual
- [9] Directx 11 tessellation. [Last access: 07.05.2017]. [Online]. Available: http://www.nvidia.com/object/tessellation.html
- [10] Translation, rotation, scale. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/manual/en/dev/modeling/meshes/editing/basics/ translation_rotation_scale.html#transform-panel
- [11] Deleting and dissolving. [Last access: 07.05.2017]. [Online]. Available: https://docs. blender.org/manual/en/dev/modeling/meshes/editing/basics/deleting.html#delete
- [12] Vertex tools, separate. [Last access: 07.05.2017]. [Online]. Available: https: //docs.blender.org/manual/en/dev/modeling/meshes/editing/vertices.html#separating

- [13] Adding geometry. [Last access: 07.05.2017]. [Online]. Available: https://docs. blender.org/manual/en/dev/modeling/meshes/editing/basics/adding.html
- [14] Extrude. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/ manual/en/dev/modeling/meshes/editing/duplicating/extrude.html
- [15] Loop subdivide. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender. org/manual/en/dev/modeling/meshes/editing/subdividing/loop_subdivide.html
- [16] Inset faces. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/manual/en/dev/modeling/meshes/editing/duplicating/inset. html?highlight=inset%20face
- [17] Bridge edge loops. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/manual/en/dev/modeling/meshes/editing/edges.html# modeling-meshes-editing-bridge-edge-loops
- [18] Bevel. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/ manual/en/dev/modeling/meshes/editing/subdividing/bevel.html
- [19] Knife tool. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/ manual/en/dev/modeling/meshes/editing/subdividing/knife_subdivide.html
- [20] Subdivide. [Last access: 07.05.2017]. [Online]. Available: https://docs.blender.org/ manual/en/dev/modeling/meshes/editing/subdividing/subdivide.html
- [21] 3d Tallinn. [Last access: 07.05.2017]. [Online]. Available: http://3d.tallinn.ee/
- [22] Google street view. [Last access: 10.05.2017]. [Online]. Available: https: //www.google.com/streetview/
- [23] Fbx assets file format. [Last access: 07.05.2017]. [Online]. Available: http: //images.autodesk.com/adsk/files/fbxassets.pdf
- [24] The blend-file-format. [Last access: 07.05.2017]. [Online]. Available: https: //wiki.blender.org/index.php/Dev:Source/Architecture/File_Format
- [25] Mapping UVs. [Last access: 07.05.2017]. [Online]. Available: https://knowledge. autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/ Maya/files/GUID-FDCD0C68-2496-4405-A785-3AA93E9A3B25-htm.html
- [26] Automatic UV mapping. [Last access: 07.05.2017]. [Online]. Available: https: //knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/

2016/ENU/Maya/files/GUID-CD17C2C5-A442-4960-91DB-A2E5099EBF61-htm. html

- [27] Maya materials. [Last access: 07.05.2017]. [Online]. Available: https://knowledge. autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/ Maya/files/GUID-E985864A-133C-47E6-B989-890EDB920D9F-htm.html
- [28] Lambert. [Last access: 07.05.2017]. [Online]. Available: https://knowledge.autodesk. com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/ GUID-97B781A0-7F21-42D4-B040-8EB967CB8563-htm.html
- [29] Blinn. [Last access: 07.05.2017]. [Online]. Available: https://knowledge.autodesk. com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/ GUID-13C10DF8-5CB4-49D2-9456-DCC1B7F1653C-htm.html
- [30] Phong. [Last access: 07.05.2017]. [Online]. Available: https://knowledge.autodesk. com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/ GUID-3EDEB1B3-4E48-485A-9714-9998F6E4944D-htm.html
- [31] Adobe Photoshop. [Last access: 07.05.2017]. [Online]. Available: https://en. wikipedia.org/wiki/Adobe_Photoshop
- [32] Adobe Illustrator. [Last access: 07.05.2017]. [Online]. Available: https://en.wikipedia. org/wiki/Adobe_Illustrator
- [33] Tiff revison 6.0. [Last access: 07.05.2017]. [Online]. Available: https://www.itu.int/ itudoc/itu-t/com16/tiff-fx/docs/tiff6.pdf
- [34] Choose colors with the eyedropper tool. [Last access: 07.05.2017]. [Online]. Available: https://helpx.adobe.com/photoshop/using/choosing-colors.html
- [35] Painting tools. [Last access: 07.05.2017]. [Online]. Available: https://helpx.adobe. com/photoshop/using/painting-tools.html
- [36] Other filters. [Last access: 07.05.2017]. [Online]. Available: https://helpx.adobe.com/ photoshop-elements/using/filters-1.html
- [37] Retouch and repair photos. [Last access: 07.05.2017]. [Online]. Available: https://helpx.adobe.com/photoshop/using/retouching-repairing-images.html
- [38] Crop and straighten photos, transform perspective while cropping. [Last access: 07.05.2017]. [Online]. Available: https://helpx.adobe.com/photoshop/using/ crop-straighten-photos.html

- [39] Bump maps. [Last access: 07.05.2017]. [Online]. Available: https://knowledge. autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/ Maya/files/GUID-792F3CD8-FD78-41C6-B967-14A0FC13AF46-htm.html
- [40] Normal map (bump mapping). [Last access: 07.05.2017]. [Online]. Available: https://docs.unity3d.com/Manual/StandardShaderMaterialParameterNormalMap.html
- [41] Displacement maps. [Last access: 07.05.2017]. [Online]. Available: https: //knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/ 2016/ENU/Maya/files/GUID-E7A03E4E-8C9A-4B62-8218-28FC063A82AB-htm. html
- [42] Vive. [Last access: 14.05.2017]. [Online]. Available: https://www.vive.com/eu/ product/
- [43] World Heritage List. [Last access: 14.05.2017]. [Online]. Available: http: //whc.unesco.org/en/list/

Lisa 1 - [Kuvatõmmised Autodesk Mudboxis modelleeritud betoontuvist]





Lisa 2 - [Adobe Illustrator tarkvaras loodud sildid]





Lisa 3 - [Näide keerukamast kunstilisest elemendist Tallinna raekoja hoonel]



Lisa 4 - [Näide keerukamast fassaadi tekstuurist Tallinna raekoja hoonel]



Lisa 5 - [Näide algsest tekstuurist ning Bump, Normal ja Displacement kaardist]



Lisa 6 - [Kuvatõmmised loodud VR rakendusest]







