

Veli-Matti Korhonen

Fotogrammetria peliasettien luonnissa

Tradenomi (AMK)
Tietojenkäsittely

Kevät 2017



KAJAANIN
AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

TIIVISTELMÄ

Tekijä: Korhonen Veli-Matti

Työn nimi: Fotogrammetria peliassettien luonnissa

Tutkintonimike: Tradenomi (AMK), tietojenkäsittely

Asiasanat: fotogrammetria, 3D-mallinnus, peligrafiikka

Tämä opinnäytetyö käsittelee fotogrammetriaa osana pelien kolmiulotteista mallinnustyötä. Työn tavoite on kuvata ja mallintaa objekteja kirjastoon, sekä tutustua tätä kautta fotogrammetriaan peliassettien luonnissa.

Aluksi tässä opinnäytetyössä käsitellään fotogrammetrian ja sen historian sekä kolmiulotteisen mallintamisen periaatteet videopeleissä. Tässä opinnäytetyössä käytetään lähteinä alan kirjallisuutta ja Internetistä löytyviä opetuskäyttöön tarkoitettuja videoita, sekä muita dokumentteja.

Lopuksi tässä opinnäytetyössä on dokumentoitu mallin rakentaminen fotogrammetrisin menetelmin ja sen muokkaaminen pelimootorille soveltuvaan muotoon.

ABSTRACT

Author: Korhonen Veli-Matti

Title of the Publication: Creating game assets with photogrammetry.

Degree Title: Bachelor of Business Administration (UAS), Business information Technology

Keywords: photogrammetry, 3D modelling, game graphics

This thesis is about photogrammetry as a part of game asset pipeline. Goals of this thesis are to reconstruct assets for asset library as well as learn photogrammetric pipeline in game arts.

This thesis will introduce photogrammetry and its history and the basics of 3d modelling at video games. This thesis uses literature, internet sources and tutorial videos as sources as well as other documents.

This thesis documents the recreation of assets using photogrammetric methods and how those assets are processed for game engine.

SISÄLLYS

1 JOHDANTO.....	1
2 FOTOGRAMMETRIA	2
2.1 Fotogrammetrian toimintaperiaate: kameralla mittaaminen.....	2
2.2 Fotogrammetrian historia	4
2.3 Fotogrammetrian tyypit.....	5
2.4 Käyttökohteita.....	6
3 3D-MALLINNUS PELEISSÄ.....	8
3.1 3D-mallin rakenne	8
3.2 Renderöinti ja polygonimäärät.....	10
3.3 Tekstuurikartat.....	11
4 CASE: REKONSTRUKTIO	14
4.1 Tavoitteet	14
4.2 Suunnittelu	15
4.2.1 Kamera sekä asetukset.....	15
4.2.2 Käytettävä ohjelmisto	17
4.3 Toteutus	18
4.3.1 Kohteen kuvaaminen	18
4.3.2 Tarkan raakamallin rakentaminen PhotoScanilla	21
4.3.3 Mallin jälkikäsittely ja uusi topologia	23
4.3.4 UV-Kartoitus.....	26
4.3.5 Normaalikartan rakentaminen	26
4.3.6 Tekstuurin heijastaminen PhotoScanissa.....	26
4.3.7 Tekstuurin jälkikäsittely	27
4.3.8 Mallin asettaminen pelimoottoriin	28
5 POHDINTA.....	29
LÄHTEET	31

SYMBOLILUETTELO

Assetti	Mikä tahansa komponentti, malli, prosessi tai viitekehys, jolla on jonkinlaista arvoa ja jota voidaan käyttää uudelleen. Tässä opinnäytetyössä enimmäkseen peliin rakennettu kolmiulotteinen malli.
Data	Digitaalisessa muodossa oleva tieto.
Kärki	3D-avaruuden piste, joka määrittää 3D-mallin muotoa.
Meshi	Tahkoverkolla määritelty 3D-malli.
Renderöinti	3D-mallinnusohjelman tapa luoda kaksiulotteinen kuvatiedosto.
Särmä	Kahden verteksin välinen suora viiva tahkoverkossa.
Tahko	Vähintään kolmen verteksin välille muodostettu pinta tahkoverkossa.
Tahkoverkko	Muoto, joka on määritelty kärkien, särmien ja tahkojen avulla.
Tekstuuri	Tekstuurikartat kuvaavat mallin pintaa tai pinnan ominaisuuksia.

1 JOHDANTO

Videopelien varhaisessa grafiikassa muutamat pikselit kuvasivat pelien sankareita sekä maailmoja, mutta totta puhuen tuolloin piti käyttää paljon enemmän omaa mielikuvitusta löytääkseen pikselikasoista sotilaita, länkkäreitä tai avaruusaluksia. Nykyisin grafiikka on niin kehittynyttä että kykenemme melkein realistisiin ympäristöihin sekä hyvin ihmisen näköisiin hahmoihin. Mutta miten nykyinen realistinen grafiikka syntyy? Pitääkö jokaisen graafikon osata maalata sekä mallintaa realistisia objekteja käsin vai onko siihen olemassa muita työkaluja?

Idea tähän opinnäytetyöhön lähti eräästä seminaarista, jossa tutustuin tarkemmin fotogrammetrian käsitteeseen. Olin lukenut kyseisestä tekniikasta, mutta uskoin sen olevan varakkaiden elokuvayhtiöiden sekä isojen pelitalojen ison rahan sekä kalliin kaluston vaativa taidonnäyte. Kuitenkin tuolla muutaman tunnin seminaarissa sain rakennettua ja teksturoitua kännykkäkamerallani ottamistani kuvista mallin omista kasvoistani ja tämä avasi minulle uuden mahdollisuuksia täynnä olevan maailman.

2 FOTOGRAMMETRIA

Fotogrammetria on yleisen määritelmänsä mukaan se tieto, taito ja teknologia, jolla saadaan luotettavaa tietoa ympäristöstä sekä fyysisistä objekteista mittaamalla, tallentamalla sekä tulkitsemalla kuvia ja kuvioita sähkömagneettisesta säteilystä digitaaliseen kuvamateriaaliin. Parhaiten fotogrammetrian voisi kääntää sanaksi kuvamittaus. (Matthews 2008, Vinni 2003)

Useimmille kamera on laite, jolla otetaan kuvia henkilökohtaiseen, tekniseen tai taiteelliseen käyttöön. Fotogrammetrian asiantuntijoille jotka johtavat kolmiulotteista tietoa kuvista, kamera on mittalaite, jolla voidaan määrittää kohteen muoto sekä sijainti. Fotogrammetrialla saadut tulokset voivat olla hyvinkin tarkkoja varsinkin jos kamera, muu kuvauskalusto sekä ohjelmisto ovat laadultaan korkeita. (Matthews 2008, Vinni 2003, Luhmann ja Robson ja Kyle ja Harley 2011)

2.1 Fotogrammetrian toimintaperiaate: kameralla mittaaminen

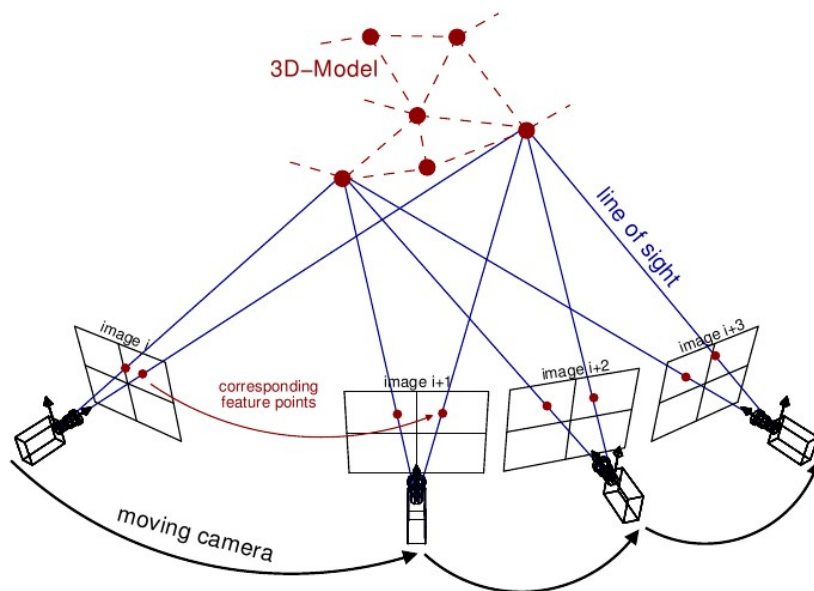
Valokuvaus sen laajimman määritelmänsä mukaan on prosessi siirtää reaali maailma litteiksi kaksiulotteisiksi kuviksi. Emme kykene tarkasti tallettamaan kolmiulotteisen maailman kaikkea tietoa kahteen ulottuvuuteen, jota seuraa vääjäämättä tiedon katoamista, pääasiallisesti syvyyssuunnasta. (Luhmann ja Robson ja Kyle ja Harley 2011)

Fotogrammetria pyrkii päinvastaiseen toimintaan kuin valokuvaaminen. Se kääntää tai kartoittaa kaksiulotteiset kuvat takaisin reaali maailman kolmiulotteisiksi kohteiksi. Koska valokuvauksessa kadotetaan tietoa, emme kykene uudelleenrakentamaan kolmiulotteista maailmaa pelkästään yhdestä valokuvasta. Tämän takia tarvitsemme vähintään kaksi kuvaa samasta kohteesta eri kuvakulmasta, jolloin saamme palautettua kadotetun tiedon. Mitä enemmän käytämme prosessissa kuvia, sitä enemmän saamme valokuvauksessa kadotettua tietoa kerättyä takaisin. Tätä kolmiulotteista tietoa, minkä saamme

prosessin lopputuotteena, kutsutaan fotogrammetriaksi. (Luhmann ja Robson ja Kyle ja Harley 2011)

Fotogrammetriassa käytetään kolmiomittauksen peruseriaatetta, jonka mukaan suorien leikkauspisteitä ja niiden muodostamia kulmia voidaan käyttää pisteen kaikkien kolmen ulottuvuuden mittaamiseen. Jotta pisteitä voidaan mitata tällä periaatteella, kameran sijainti sekä tähtäyskulma pitää olla tiedossa jokaiselle kuvalle ja lasketaan taaksepäinleikkauksella. Taaksepäinleikkaus poikkeaa kolmiomittauksesta siten, että sen mittaus perustuu kahden toisiinsa liittyvän kolmion varaan. Tällä mittauksella on kolme lähtöpistettä ja niistä saadaan johdettua lähtöpisteiden väliset kulmat sekä tähtäyssuunnat lähtöpisteille. (Luhmann ja Robson ja Kyle ja Harley 2011)

Jokaisesta kuvasta tunnistetaan samat kiintopisteet, joiden perusteella mittaus suoritetaan. Säde voidaan rakentaa kameran paikasta kohteen kiintopisteeseen. Näiden säteiden leikkauskohdista ja kulmista johdetaan kohteen kiintopisteiden kolmiulotteiset paikat kuvan 1 mukaisesti. (Luhmann ja Robson ja Kyle ja Harley 2011)



Kuva 1. Fotogrammetrian periaate.

2.2 Fotogrammetrian historia

Fotogrammetria on yhtä vanha keksintö kuin itse moderni valokuvaus. Se sai alkunsa hyvin pian valokuvauksen keksimisen jälkeen 1850-luvulla. Vuonna 1849 ranskalainen Aimé Laussedant oli ensimmäinen henkilö, joka käytti valokuvapareja topografisten karttojen luomiseen Pariisin kaupunkia kartoittaessaan. Häntä pidetään tästä syystä "fotogrammetrian isänä". Hän kokeili myös ilmasta tapahtuvaa fotogrammetriaa leijoilla vuonna 1858, mutta jätti tutkimukset kesken muutamaa vuotta myöhemmin. Laussedantin tapa käyttää valokuvia kartoitukseen hyväksyttiin virallisesti vuonna 1862 Madridin tiedeakatemiassa. Vuonna 1893 Albrecht Meydenbauer oli ensimmäinen henkilö, joka käytti termiä fotogrammetria. Meydenbauer tunnetaan fotogrammetrian käytöstä arkkitehtuurin tutkimiseen. Amerikan sisällissodan jälkeen 1863, Oliver Holmes käytti fotogrammetriaa sovittaessaan proteeseja raajansa menettäneille sotilaille. (The Center for Photogrammetric Training, Fryer ja Mitchell ja Chandler 2007)

Kaksi tärkeää innovaatiota, stereoskopian yleistyminen sekä Wrightin veljesten vuonna 1903 kehittämä lentokone veivät fotogrammetriaa reilusti eteenpäin. Mittamerkin periaatteen keksimisen jälkeen saksalainen fyysikko Carl Pulfrich valmisti 1900-luvun alussa ensimmäisen kolmiulotteisten mittausten tekemiseen sopivan laitteen. Ilmakuvia otettiin jo ennen lentokoneen käyttöönottoa, mutta ilmapallo ja leijat olivat tuulen armoilla toisin kuin lentokone, jota voitiin ohjata minne ja miten haluttiin. Tämä mahdollisti helpomman maastokartoituksen ja myös armeijat kiinnostuivat käyttämään ilmakuvia tiedustelutoiminnassaan. (Vinni 2003, The Center for Photogrammetric Training)

Toisen maailmansodan aikaan fotogrammetria hyppäsi jälleen eteenpäin teknologisten kehitysten kuten väri- ja infrapunafilmien ansiosta, joiden käyttö mahdollisti entistä tarkemman tiedustelun sekä kartoituksen. Sodan jälkeisenä aikana luotiin suurin osa nykyaikaisen fotogrammetrian matemaattisista menetelmistä. Tietokoneiden yleistyttyä ja teknologian kehittyessä fotogrammetria on siirtynyt lähes täysin digitaaliseen muotoon tietokoneiden laskentatehon merkittävän kasvun ansiosta. Myös laserskannerit ovat kehittyneet

merkittävästi viimeisen vuosikymmenen aikana jonka ansiosta saamme entistä helpommin ja tarkempaa tietoa, jota pystymme näyttämään nykyaikaisilla näyttölaitteilla kolmiulotteisessa muodossa. Fotogrammetristen menetelmien kehitys on viimeisen vuosikymmenen aikana siirtynyt kartoituksesta kuitenkin muille sovellusaloille. (Vinni 2003)

2.3 Fotogrammetrian tyypit

Fotogrammetria voidaan jakaa useaan kategoriaan, mutta yleisesti se on jaettu kameran etäisyydestä riippuen kahteen pääkategoriaan, perinteiseen ilmasta tapahtuvaan fotogrammetriaan (aerial photogrammetry) sekä uudempaan lähietäisyyden fotogrammetriaan (close-range photogrammetry). Teknologian kehityksen myötä tarkkojen kameroiden, tehokkaiden tietokoneiden sekä pienten lennokkien yleistyttyä jaottelun raja on kuitenkin hämärtynyt päällekkäisyyksien myötä.

Ilmafotogrammetria käyttää suuria kuvia sekä maastopisteiden koordinaatteja hyväkseen tarkkojen maakaistaleiden uudelleenluonnissa. Ilmasta kuvattaessa kamera kiinnitetään yleensä lennokkiin tai lentokoneeseen ja se laitetaan osoittamaan pystysuoraan maata kohti. Kamera ottaa usean hieman päällekkäisen kuvan maasta lennokin reitin varrelta. Päällekkäisten kuvien tulee kattaa noin 60% toistensa kuvatiedosta hyvän lopputuloksen saavuttamiseksi. (Matthews 2008)

Lähifotogrammetriassa käytetään samoja stereoskopian toimintaperiaatteita kuin perinteisessä ilmafotogrammetriassa. Lähifotogrammetriaksi luetaan kuvat jotka ovat otettu alle 300 metrin päästä kohteesta. Tämä etäisyys on sama kuin pienin sallittu ylilentokorkeus asutusalueilla Yhdysvaltojen ilmailuhallinnon (Federal Aviation Administration) mukaan. Koska samat toimintaperiaatteet pätevät läheltä käsikäyttöisellä kameralla kuvattaessa sekä pienten välimatkojen lennökkikuvaamisessa, molemmista käytetään nimitystä lähifotogrammetria. (Walford 2007, Matthews 2008)

Läheltä kuvattaessa voidaan käyttää erilaista kuvauskalustoa kännykkäkamerasta pieniin ilmapalloihin kiinnitettävään kameraan saakka. Koska lähifotogrammetria mahdollistaa monenlaisen kaluston käytön, on mahdollista päästä 0.25mm ja jopa 0.025mm tarkkuuteen rakenteiden mittaamisessa. (Matthews 2008)

Tässä opinnäytetyössä perehdytäänkin enemmän lähifotogrammetriaan sekä sen tekniikoihin ja sovelluksiin objekteja mallintaessa.

2.4 Käyttökohteita

Kuten jo aikaisemmin todettiin, fotogrammetrian alkuperäinen käyttötarkoitus oli pitkälti kartta- sekä maastotiedon tuottaminen, mutta sille on teoriassa loputtomasti käyttösovelluksia, loppujen lopuksi se on vain kuvista mittaamista. (Fryer ja Mitchell ja Chandler 2007)

Arkkitehtuurissa fotogrammetriaa käytetään usein historiallisten rakennusten tutkimiseen ja dokumentointiin, esimerkiksi entisöintiä sekä korjaustöitä varten. Kokonaisia rakennuksiakin voidaan myös tarpeeksi laajan kuvamateriaalin avulla mallintaa, joka avaa mahdollisuuden esimerkiksi historiallisesti merkittävien rakennuksien virtuaaliseen esittelyyn, katselemiseen ja tutkimiseen. Tarkkuutensa vuoksi osaksi raunioituneista monumenteista sekä rakennuksista otettuja malleja voidaan käyttää myös pohjana entisöintityöhön. Myös vedenalaisten kameroiden kehittyminen on mahdollistanut vedenalaisen fotogrammetrian, jonka käyttökohteita ovat esimerkiksi öljynporauslauttojen vedenalaisten rakenteiden rakennetarkastukset, merenpohjan kartoitus sekä syvyysmittaukset. (Agisoft 2015, Vinni 2003, Fryer ja Mitchell ja Chandler 2007)

Fotogrammetrian käyttö on myös yleistynyt oikeussaleissa jo 1950-luvulta lähtien. Sitä on käytetty esimerkiksi auto-, juna- sekä lento-onnettomuuksien rekonstruktiossa. Ennen tietokoneiden yleistymistä näitä malleja on käytetty havainnollistamaan onnettomuustilanteita. Tietokoneiden laskentatehon kasvun sekä valvontakameroiden yleistymisestä johtuen onnettomuustilanteiden uudelleenluonti ja havainnollistaminen ovat edelleen hyvin yleisiä

käyttötarkoituksia, mutta nykyään valokuvien sijaan kyetään tarjoamaan myös kokonaisia 3D-malleja sekä animaatioita tapahtuneesta. Myös rikoksista saadaan tarkempaa tietoa analysoimalla valvontakameroiden kuvia ja rakentamalla niistä tarkempia kokonaisuuksia rikoksesta epäillyistä. (Fryer ja Mitchell ja Chandler 2007)

Reaaliaikainen 3D-grafiikka on hypännyt todella isoja harppauksia eteenpäin viimeisten vuosien aikana. Tietokoneet kykenevät käsittelemään valtavia määriä verteksejä sekä pikseleitä työstäessään kolmestakymmenestä kuvasta kuuteenkymmeneen HD kuvaa sekunnissa. Todellisuus ja peleissä tehtävä reaaliaikainen laskenta ovat kuitenkin vieläkin kaukana toisistaan. Realistinen valaistus ja geometrian määrä ovat vielä rajoittavia tekijöitä jotka pitää ottaa aina huomioon 3d-malleja rakentaessa. Myös tekstuurien resoluutioiden tulee olla tarpeeksi pieniä, että kaikki tieto saadaan mahtumaan tietokoneen muistiin.

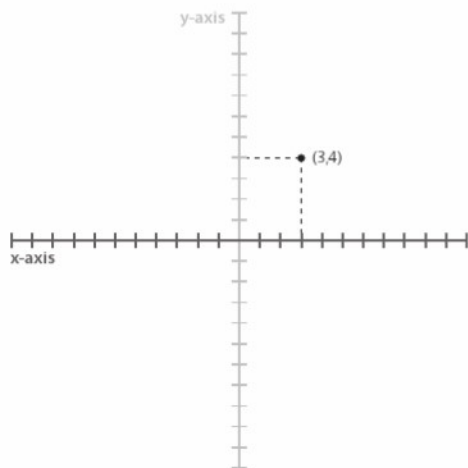
Videopelejä ajatellessa fotogrammetriaa käytetään erittäin tarkkojen ja realististen tekstuurien sekä mallien luomiseen. Fotogrammetristen ohjelmien tarjoamia malleja joudutaan kuitenkin vielä työstämään sekä jälkikäsittelemään, sillä usein pelimoottorien tekniset rajoitukset sekä muistin riittämättömyys tulevat vastaan. (Poznanski 2014, Graphine)

3 3D-MALLINNUS PELEISSÄ

Hyvin moni nykyaikainen peli käyttää kolmiulotteisia malleja graafisena ilmeenään. Perinteisemmät kaksiulotteiset pelit eivät ole kuolemassa tai katoamassa edelleenkään minnekään, mutta lähes kaikki alustat integroivat itseensä kolmiulotteisia elementtejä. Tässä kappaleessa käydään läpi kolmiulotteisten objektien teoriaa sekä tekstuurikarttoja ja niiden funktiota kolmiulotteisissa malleissa sekä pelimoottoreissa.

3.1 3D-mallin rakenne

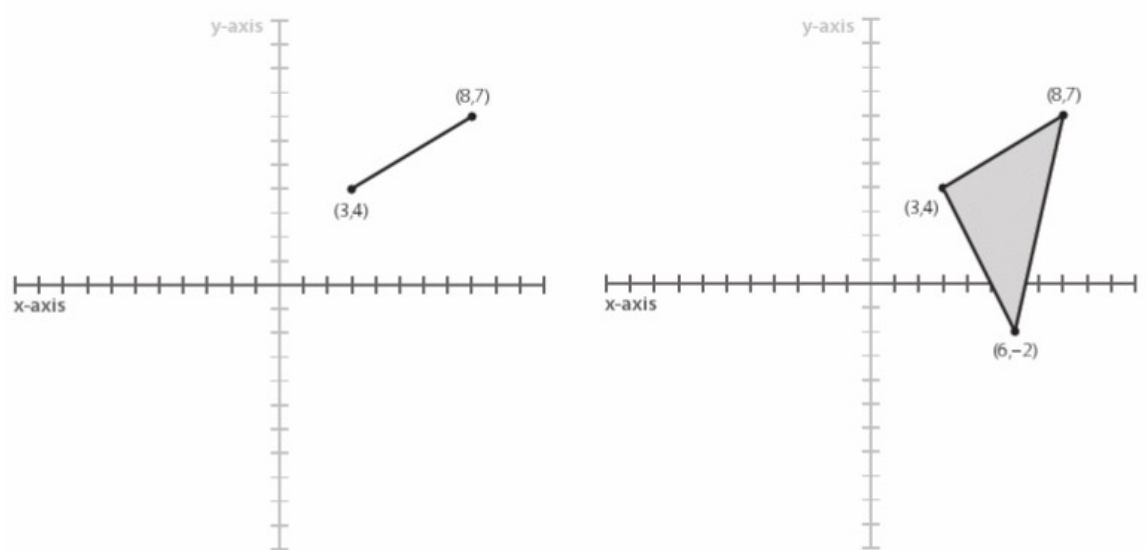
3D-mallin rakentuminen lähtee liikkeelle perinteisestä xy-koordinaatistosta, jossa kaksi akselia, vaakasuora x-akseli sekä pystysuora y-akseli muodostavat tason. Kun näille akseleille asetetaan piste, voidaan sille antaa numeerinen arvo kuvaamaan pisteen paikkaa tasolla kuvan 2 mukaisesti. Yksittäistä pistettä kutsutaan kärjeksi (vertex). (Totten 2012)



Kuva 2. Kärki ja sen paikka xy-koordinaatistolla.

Jos tasolle piirretään toinen piste ja se yhdistetään edelliseen pisteeseen, muodostuu kuvan 3 mukaisesti jana. Tällaista janaa kutsutaan särmäksi (edge). Kun tasolle luodaan useampi särmä ja ne järjestetään siten että ne muodostavat suljetun muodon, tuloksena tulevia kaksiulotteisia muotoja kutsutaan

polygoneiksi. Jos polygonilla on pinta, sitä kutsutaan tahkoksi (face). Kaikista yksinkertaisin muoto näistä on kolmio, joka koostuu kolmesta janasta kuvan 3 mukaisesti. (Totten 2012)



Kuva 3. Kahden kärjen väliin muodostuva särmä, sekä kolmen kärjen väliin muodostuva tahko.

Jos edellä mainittuun tasoon liitetään vielä kolmas akseli z, saadaan kaksiulotteisesta tasosta kolmiulotteinen. Kärjillä sekä särmillä on tällöin kolme koordinaattia, jotka ilmoittavat niiden sijainnin x-, y-, sekä z-akseleilla. Kuten kaksiulotteisessa tasossa, tahkot esiintyvät myös kolmiulotteisesti ja niitä yhdistelemällä voidaan luoda kolmiulotteisia muotoja, kuten lieriöitä, kuutioita sekä monia muita muotoja. (Totten 2012)

Tämän tyyliiset objektit ja mittausmenetelmät ovat tuntemamme nykyaikaisen 3D-tietokonegrafiikan perusta. 3D-ohjelmilla työskennellessä käsittelemme kohteiden kärkiä, särmiä sekä tahkoja muodostaaksemme monimutkaisempia muotoja kuten pelihahmoja. (Totten 2012)

3.2 Renderöinti ja polygonimäärät

3d-graafikot pyrkivät tekemään elokuvaan sekä animaatioihin mahdollisimman realistisia malleja. Myös piirroshahmoille, kuten esimerkiksi lasten ohjelmien hahmoille, pyritään luomaan mahdollisimman todenmukaisesti iho, silmät, karvat, vaatteet sekä taustaelementtien materiaalit. Pelit saavuttavat koko ajan enemmän realistista ulkomuotoa, mutta ne eivät vielä kykene täysin simuloimaan todellisuutta kuten 3D-elokuvat tai animaatiot lähes kykenevät. Tämä johtuu siitä, että videot ovat pääsääntöisesti sarja perättäisiä esirenderöityjä kuvia, joissa ei tarvita näyttöhetkellä enää reaaliaikaista laskentaa. (Totten 2012)

Elokuvateollisuuden 3D-animaattorit käyttävät usein samoja työkaluja, joita graafikot peliteollisuudessaakin, mutta he pyrkivät työssään näyttävyyteen. Usein tämä tarkoittaa enemmän kärkiä sekä särmiä pelimalleihin nähden, mikä tarkoittaa lisää tietoa jota tietokoneen pitää ottaa huomioon rakentaessaan malleja. (Totten 2012)

Kun malli on lopullisesti valmis ja se on aika saattaa loppuun, se renderöidään. Kun tietokone kokoaa kaiken geometrisen tiedon mallista sekä kaiken malliin vaikuttavan materiaalitiedon, se laskee, miltä lopputuote näyttäisi reaali maailmassa. Tätä laskutoimitusta sekä lopputuotetta kutsutaan renderöinniksi. Renderit ovat kaksiulotteisia kuvia, joissa ei ole ollenkaan kolmiulotteista tietoa, kuten valokuvat. Kolmiulotteisissa animaatioelokuviissa jokainen kuva on renderöity ja kun nämä kuvat järjestetään peräjälkeen, muodostuu elokuva. (Totten 2012)

Kuvien sekä elokuvien renderöinti voi viedä tunneista jopa viikkoihin. Nykyaikaisen animaatioelokuvan yksi kuva voi viedä jopa 38 tuntia aikaa renderöidä, ja jokainen sekunti sisältää vähintään 20 kuvaa. Tämän takia yleisesti isommissa renderöintitarpeissa käytetään renderöintifarmia tai parvea, isoa tietokonejoukkoa yhtäaikaiseen rinnakkaiseen laskemiseen, säästäten näin merkittävästi aikaa odottelulta. Kun laskenta on valmis, animaatio on helposti säilöttävässä, siirrettävässä ja katseltavassa muodossa. (Totten 2012)

Peligraafikon haasteena elokuvaan verrattuna on työn renderöinti reaaliajassa. Jokainen animaatio, 3D-malli ja niihin vaikuttavat valot, tekstuurit sekä interaktio muun maailman kanssa tulee tapahtua pelaajan silmien edessä välittömästi. Käytännössä peligraafikon tulee ottaa huomioon suorituskykyyn liittyviä asioita, jotta peli toimii sulavasti kaiken aikaa. Ensimmäisenä ja tärkeimpänä graafikon pitää valvoa tarkasti polygonien määrää pelissä. (Toten 2012)

3.3 Tekstuurikartat

Peligraafikot eivät voi käyttää valtavaa määrää polygoneja jokaisen yksityiskohdan saavuttamiseen pelimoottorien sekä tietokoneiden laskentatehon vuoksi. Näiden rajoitusten takia tekstuurikartat ovat peligraafikon tärkeimpiä työkaluja mallinnustyönsä täydentämiseksi. Tekstuureilla tarkoitetaan kaksiulotteisen kuvatiedon projisointia 3-ulotteisen kappaleen pinnalle antaen sille pinnoitteen. Tekstuurikartat ovat valmiiksi laskettua tietoa kuvamuodossa, esimerkiksi valon heijastuskulmat normaalikartoissa sekä väriarvot diffuusikartassa. (Masters 2013)

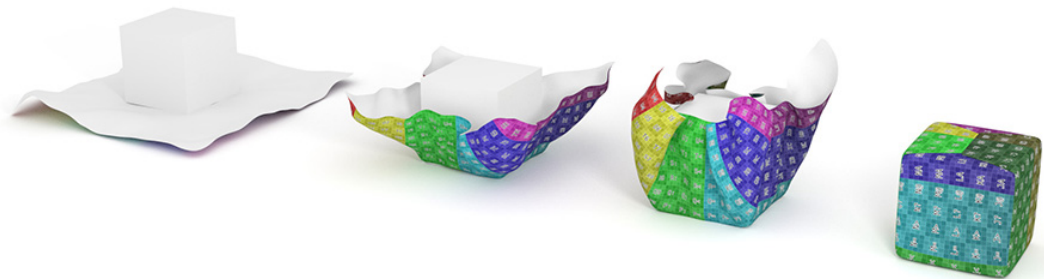
Tekstuurien koko on hyvä tehdä kahden potenssissa esimerkiksi 128x128 tai 256x1024. Mikäli kuvilla on jokin muu kuvasuhde, pelimoottori joutuu skaalaamaan kuvan koon uudelleen lähimpään potenssiin, jolloin kuluu lisää aikaa sekä laskentatehoa. Yleisesti ottaen graafikko voi käyttää mitä vain kokoa 4096x4096 saakka, tosin useille näytönohjaimille ja alustoille 2048x2048 on korkein kuvakoko. (Art Asset Best Practice Guide 2015)

Pelin aikana tietokone joutuu laskemaan sekä prosessoimaan monia asioita yhtäaikaaisesti jokaisella näytetyllä kuvaruudulla (frame). Näitä laskutoimituksia ovat esimerkiksi jokaisen vihollisen ja hahmon paikka maailmassa, erilaiset valaistukset ja äänenvoimakkuus hahmoihin nähden.

Aina kun graafikko voi vähentää tietokoneen tekemien laskutoimituksien määrää tekemällä ne jotenkin etukäteen, vapautetaan tietokoneen laskentatehoa muuhun käyttöön. Mikäli graafikko voi käsitellä tiedon määrää ja antaa valmiiksi laskettuja toimenpiteitä tietokoneelle, silloin sen ei tarvitse laskea niitä

reaaliajassa. Tätä ennalta laskettua tietoa kutsutaan nimellä leivottu tieto (baked data), ja toimenpidettä jolla se saadaan aikaiseksi, kutsutaan leipomiseksi (baking). (Thorn 2014)

Ennen karttojen tekemistä kolmiulotteiset mallit joudutaan purkamaan kuvatiedostoiksi, joista voimme heijastaa tarvitsemamme tiedon mallin päälle. Tätä työvaihetta kutsutaan UV-kartoitukseksi. UV-kartoitus käsittää prosessin muuttaa mallin tahkojen tieto kaksiulotteisiksi koordinaateiksi tasolle, jota voimme sen jälkeen käyttää lisätäksemme malliin tekstuurikarttoja. UV-koordinaattien luominen on kuin mallin pinnan leikkaaminen ja sen levittäminen tasaiselle pinnalle kuvan 4 mukaisesti. (Strong 2007, Totten 2012, Simonds 2012)

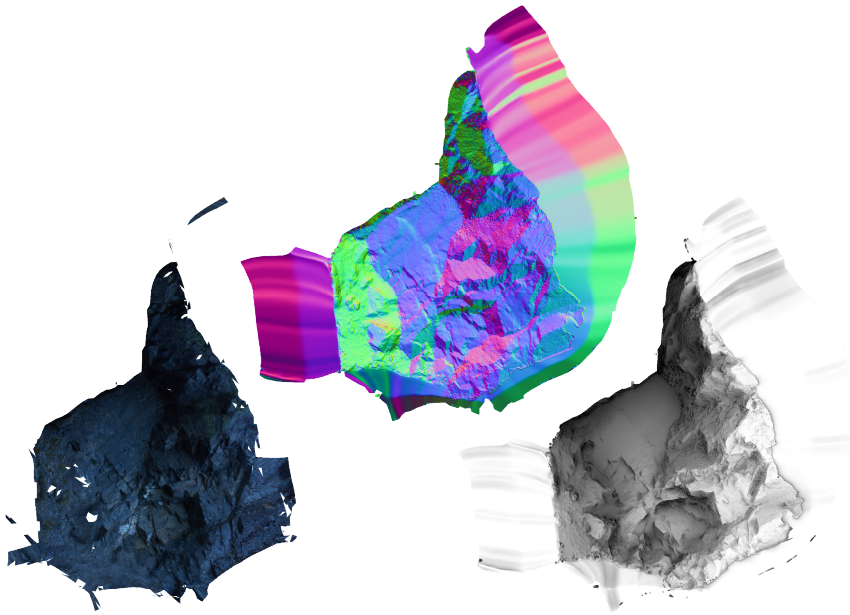


Kuva 4. UV-kartoituksen periaate.

Kaikista yleisin tekstuurikartta on diffuusikartta. Peleissä diffuusikartta tarkoittaa yleisesti väritekstuuria, mutta sitä voidaan kutsua myös albedo kartaksi. Tämä tekstuurikartta sisältää mallin perusväriyksen. (Stifter 2014)

Normaalikartta on tekstuuritiedosto, joka koostuu punaisesta, vihreästä sekä sinisestä (RGB) väristä. Nämä RGB arvot kuvastavat pinnan normaalivektorin komponentteja mahdollistaen täten 2D-kuvan sisältävän tietoa syvyydestä sekä kulmista. Tämän tiedon perusteella yksinkertainen 3D-malli saadaan näyttämään sisältävän paljon enemmän muotoja ja yksityiskohtia kuin se todellisuudessa sisältää. (Hajioannou 2013)

Valokartta sisältää valmiiksi lasketun tiedon valaistuksesta ennenaikaisesti. Peleissä käytetään useita tekniikoita joilla saadaan annettua valaistuksesta tietoa, mutta suurin osa tekniikoista käyttää tekstuureja ennalta lasketun tiedon tallentamiseen. Tämä käytäntö ennalta renderöi kaiken valaistuksen staattisesta objektista ja tallentaa tuloksen erikoiseksi tekstuuriksi, joita kutsutaan valokartoiksi(lightmap). Nämä valokartat yleensä sisältävät suoran ja epäsuoran valon, sekä varjojen tiedon kuvan 5 mukaisesti. (Thorn 2014)



Kuva 5. Mallin värin sisältävä diffuusikartta, heijastus suunnat sisältävä normaalikartta sekä valon määrän kertova valokartta.

4 CASE: REKONSTRUKTIO

Tämän opinnäytetyön tavoitteena on luoda asetteja käyttäen fotogrammetriaa työkaluna. Käyn tässä kappaleessa läpi tekemieni mallien työvaiheet suunnittelusta tekniseen toteutukseen.

4.1 Tavoitteet

Tavoitteenani on luoda asetteja asettikirjastooni sekä tutustua fotogrammetrian saloihin työkaluna. Markkinoilla on tarjolla useita työkaluja proseduaaliseen mallintamiseen, kuten puiden nopeaan tekemiseen tarkoitettu SpeedTree. Tavoitteeni on tehdä mahdollisimman realistisia asetteja kuten kiviä ja puita joita ei vielä voida tehdä proseduaalisesti realistiseksi. Esimerkiksi kiven luominen veistäen kuulostaa paljon helpommalta mitä se todellisuudessa onkaan.

Kuten Gnomon school of digital artsin perustaja Alex Alvarez sanoi fotogrammetria tarjoaa hyvän lähtökohdan muodoille, tekstuureille, materiaaleille, valoille sekä kompositiolle. Fotogrammetria mahdollistaa myös käyttökelpoisten teksturoitujen 3d-mallien lisäksi mahdollisuuden laajentaa omaa ZBrush alfojen kirjastoa. ZBrushin alfat ovat 16-bittisiä mustavalkoisia kuvatiedostoja, jotka sisältävät korkeustietoa, joita voidaan käyttää esimerkiksi seinämien, puun runkojen sekä ihon pinnan rakenteen tekemiseen. Alvarezin tekemän kuvan 6 kalliota ja niiden yksityiskohdat ovat tehty käyttämieni metodien mukaisesti. (The Gnomon Workshop 2016)



Kuva 6. Alvarezin tekemä maasto, jonka luonnissa on käytetty fotogrammetriaa.

4.2 Suunnittelu

Opinnäytetyön tavoite huomioon ottaen suunnittelutyö tämänkaltaiseen projektiin on hyvin suoraviivaista. Aluksi hankin ohjelmiston sekä kuvauskaluston, jonka jälkeen lähdin etsimään mielenkiintoisia kohteita kuvattavaksi. Minulla ei ollut mitään erikoista tarvetta juuri tietyn näköisille objekteille, joten valitsin mielenkiintoisimman näköiset kohteet ja kuvasin monta eri kohdetta, jotta kuvamateriaali riittäisi ainakin muutamaankorkealaatuiseen tuotokseen.

4.2.1 Kamera sekä asetukset

Käytössäni oli Canonin EOS 450D 12.2 megapikselin järjestelmäkamera. Mitä isompi pikselimäärä kamerassa on, sitä tarkempia kuvia ja sitä myötä enemmän yksityiskohtia saadaan mallinnettua lopputuotteeseen. Kännyköillekin on tarjolla fotogrammetrian sovelluksia, mutta megapikselien määrä, optiikan laatu sekä rajalliset säätömahdollisuudet eivät tarjoa aina halutun laatuista lopputulosta järjestelmäkameroihin verrattuna.

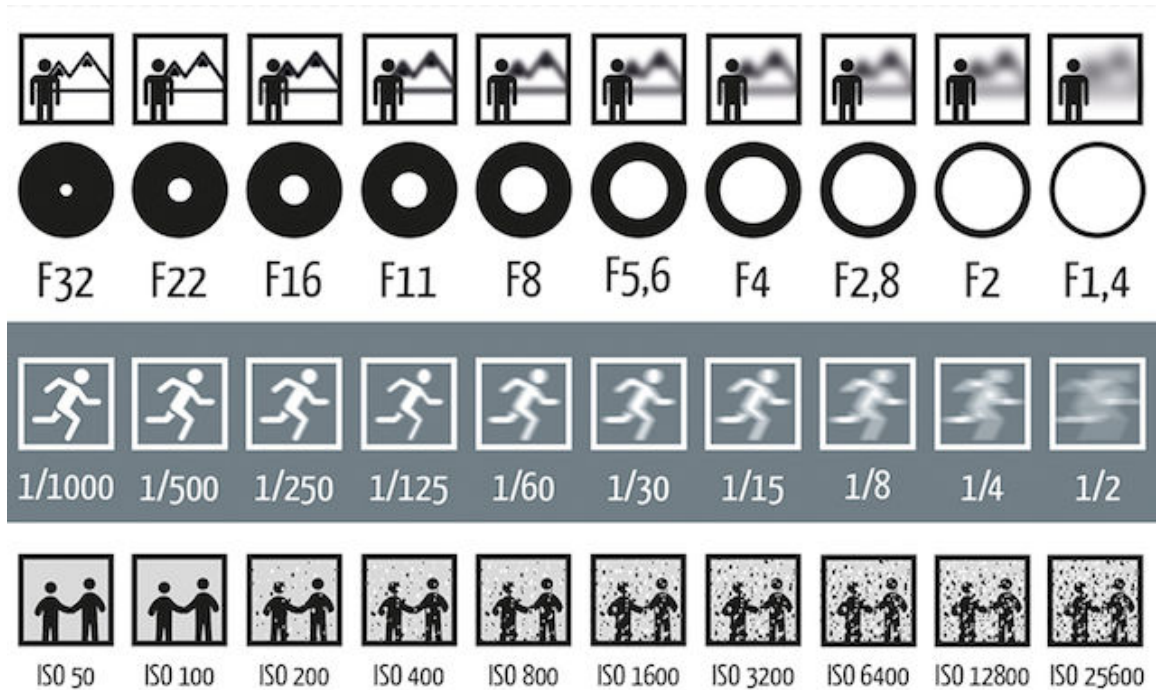
Alvarezin mukaan seuraavat asetukset ovat toimineet hänen tekemiin malleihin. Kiinteäpolttovälinen objektiivi takaa tasaisemman kuvanlaadun kuin zoomattavat objektiivit. Zoomattavien objektiivien kanssa kuvatessa zoomin määrä voi

vaihdella vahingossa tai tahallisesti, antaen epätasaisemman lopputuloksen kuvasarjalle. PhotoScan tuottaa myös tarkemman lopputuloksen kaikkien kuvien terävyysalueen ollessa sama.

Käsisäätöinen tarkennus on erittäin tärkeä ominaisuus kuvattaessa. Kännykkäkamerat ovat tästäkin syystä usein huonompia kuvausvälineitä automaattisen tarkennuksensa takia. Jokaisessa kuvassa tulisi olla samat asetukset koko ajan jolloin aukkoarvo, pikasuljin ja ISO-herkkyys vaikuttavat lopputulokseen. Kuvista heijastetaan lopuksi tekstuuri valmiin mallin pintaan, joten kaikkien kuvien tulisi olla värimaailmaltaan sekä tarkkuudeltaan mahdollisimman saman kaltaisia. Alvarez suosittelee aukkoarvoksi 11+, nopeaa 1/125+ suljinaikaa sekä ISO-herkkyudeksi 800 tai vähempää parhaan lopputuloksen saavuttamiseksi. Kameran aukko päästää valoa sisään ja tätä säätelevää mekanismia kutsutaan aukkoarvoksi (F-Stop). Pieni aukkoarvo esimerkiksi F1,4 tarkoittaa että aukko on iso ja valoa pääsee sisään paljon ja iso aukkoarvo esimerkiksi F16, että valoa pääsee paljon vähemmän sisään kameraan. Tämä tarkoittaa että mitä isompi aukkoarvo on sitä suurempi syväterävyysalue. Koska haluamme jokaisen kuvan olevan mahdollisimman teräviä, pyrimme suhteellisen isoihin aukkoarvoihin kuten f11+. ISO tarkoittaa filmin tai kameran valoherkkyttä. Alvarez suosittelee ISO-herkkyudeksi 800 tai vähemmän, studiooloissa vielä vähemmän. ISO-herkkyuden ollessa esimerkiksi 1600 päiväaikaan tuottaa kuviin rakeisuutta ja tätä pyrimme aktiivisesti estämään. Suljinaika ilmaisee kuinka kauan kameran suljin on auki kuvattaessa. Käsivaralta kuvattaessa ulkona suljinajan tulisi olla hyvin nopea, jolloin mahdollisuudet liikkeestä johtuviin virheisiin vähenevät. Studio olosuhteissa suljinaika voi olla isompikin värinän ja liikkeen ollessa poistettuna.

Kuvamuodoksi Alvarez suosittelee .RAW muotoa kuvan korkeamman dynamiikan takia. Korkeampi dynamiikka antaa mahdollisuuden säätää kuvien valotuksen tasoa kuvankäsittelyohjelmissä, kuten Photoshop tai Lightroom. Fotogrammetrian yksi tarkoituksesta on myös saada kuvista luotua mallille automaattinen tekstuuri, jolloin valotuksen sekä tekstuurien tarkkuuden on hyvä olla niin hyviä kuin vain kuvauskalustolla on mahdollista saada.

Kohdetta kuvataan 10 asteen välein kohdentaen samaan kohtaan ja kierroksia tehdään yhdestä kolmeen, mahdollistaen päällyys- sekä pohjaosien kuvauksen. Täten kuvia kertyy keskimäärin 36-108 jokaisesta kohteesta.



Kuva 7. Aukkoarvo, valotusaika sekä ISO-herkkyys vaikuttavat kuvien lopputulokseen.

4.2.2 Käytettävä ohjelmisto

Kolmiulotteiseen rekonstruktioon käytän Agisoftin kehittämää PhotoScan ohjelmaa ja mallien jälkikäsittelyyn Blenderiä sekä pixologigin ZBrushia. Tekstuurien jälkikäsittelyyn käytän Adoben Photoshopia. Lopuksi luon alhaisen polygonimäärän malliin normaalikartat xNormalilla. Ohjelmat valikoituivat saatavuuden sekä hinnan mukaan. Koulullamme on lisenssit ZBrushille sekä sain koulun kautta käyttööni PhotoScaniin kuukauden Pro lisenssin. Markkinoilla on muitakin ohjelmia joilla voi toteuttaa saman, mutta kriteerini olivat helppo saatavuus sekä alhaiset kustannukset.

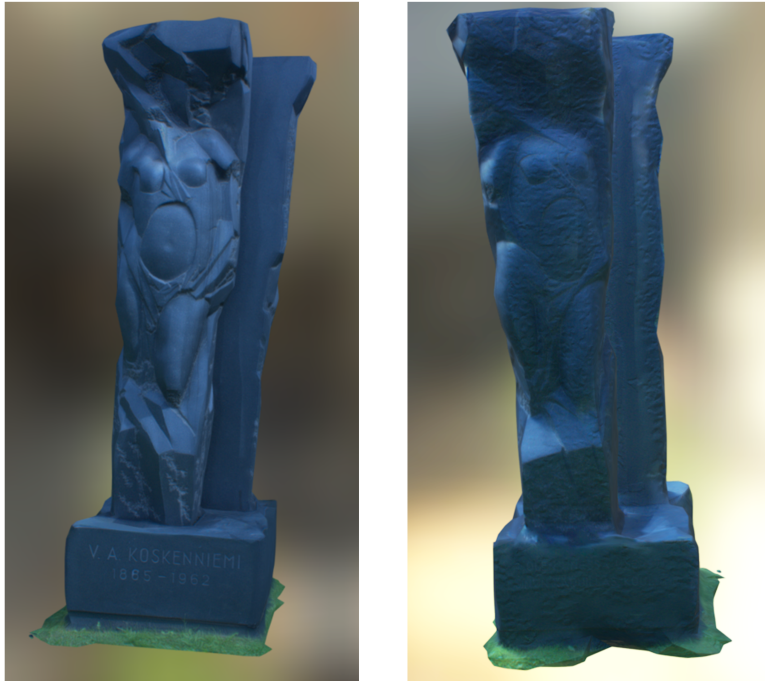
4.3 Toteutus

Tässä kappaleessa käyn läpi mallien toteutuksen. Työvaiheet voidaan jakaa karkeasti kahdeksaan osaan seuraavanlaisesti.

- 1 Kuvaaminen
- 2 Tarkan raakamallin rakentaminen PhotoScanilla
- 3 Mallin jälkikäsittely ja uusi topologia
- 4 UV-kartoitus
- 5 Normaalikartan rakentaminen
- 6 Tekstuurin heijastaminen PhotoScanissa
- 7 Tekstuurin jälkikäsittely
- 8 Mallin asettaminen pelimoottoriin

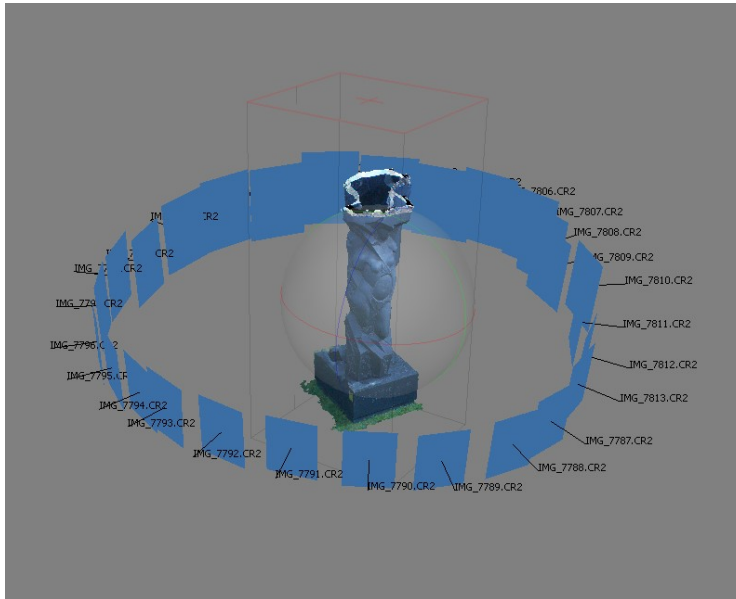
4.3.1 Kohteen kuvaaminen

Odotin kuvauspäiviksi pilvisiä päiviä, jolloin valaistus olisi mahdollisimman tasainen ja kuvat välttyisivät teräviltä valoilta sekä varjoilta. Kuvasin osan kohteista testimielessä myös aurinkoisena kesäpäivänä, jolloin näkisin miten vahvat valot sekä terävät varjot vaikuttaisivat lopputulokseen. Alla olevasta kuvasta 8 pystyy näkemään, että vasemmalla oleva ylivalottunut osuus tallentuu hyvin, mutta oikealla oleva varjoinen osuus sisältää enemmän epätasaisuutta. Terävät valot näkyvät myös selvästi patsaan pinnasta ja vaatii enemmän tekstuurin jälkikäsittelyä, sillä haluan valaistuksen tapahtuvan vasta pelimoottorin sisällä.



Kuva 8. Valaistuksen vaikutus lopputulokseen.

Kuvasin valituista kohteista noin 10 asteen välein yhden kuvan, kunnes olin kiertänyt kohteen ympäri ja kuvannut sen jokaisesta suunnasta. Päätelin kymmenen asteen olevan noin yhden askeleen kohteen ympäri, joka osoittautui toimivaksi arvioksi. Kuvasin kallion seiniä, pienehköjä kiviä, puita sekä patsaan, jolloin yksi kierros riitti usein. Kuten alla oleva kuva 9 osoittaa, patsaalle olisi pitänyt ottaa korkeammaltakin kuvia, jolloin sen yläpintakin olisi saatu mallinnettua, mutta en päässyt mitenkään ottamaan kuvia patsaan päältä. Tämä ei ole kuitenkaan ongelma, sillä patsaan päällys on luultavasti tasaista kiveä joten voin kopioida myöhemmin materiaalin esimerkiksi Photoshopilla patsaan päälle. Seinämiä kuvattaessa kuvia kertyi alle 20, mutta nekin riittivät pääsääntöisesti hyvään lopputulokseen.



Kuva 9. Patsaan ympäriltä otettu kuvasarja, jossa jokainen sininen neliö kuvastaa kohtaa, josta kuva on otettu.

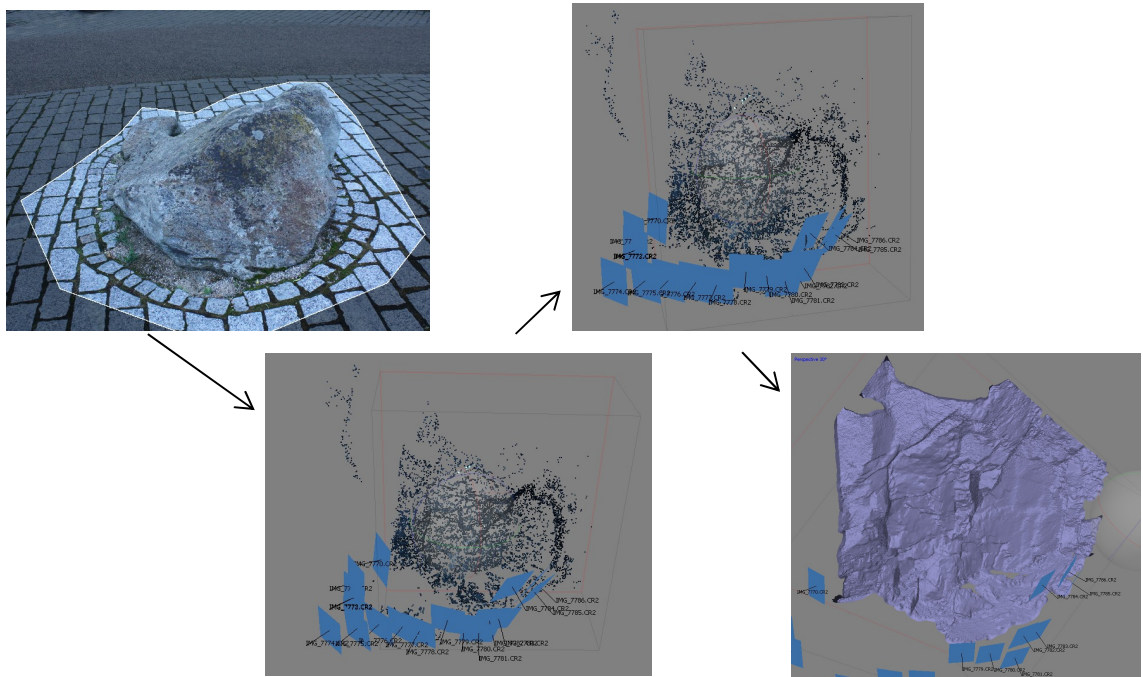
Tasaisimman mahdollisen lopputuloksen saisi studio-olosuhteissa, jossa kohde pyörisi kuvaajan sijaan tai se kuvattaisiin usealla kameralla jokaisesta siinnasta yhtäaikaaisesti. Tällöin valotus ja tarkkuus olisivat täydelliset jokaisessa kuvassa. Kuitenkaan kaikkia esineitä ei saada vietyä studioon, esimerkiksi kuvan 9 yli kolme metristä patsasta.

Parhaan lopputuloksen sekä tasaisesti valottuneen tekstuurin saamiseksi tarkastin kohteesta otetut raakakuvat tietokoneen ruudulta, jolloin sain todettua kameran esikatselussa hyvältä näyttäneet kuvat oikeasti onnistuneiksi kuviksi. Käsittelin myös kuvien valoarvoja Photoshopilla tasaisemmiksi, jonka jälkeen tallensin ne .TIFF muotoon. Kuvat voi tallentaa mihin tahansa PhotoScanin käyttämään kuvamuotoon. Valitsin .TIFF muodon koska myös Alvarez käyttää sitä työskennellessään. (ainakaan kirjoitus hetkellä PhotoScan ei osaa käsitellä RAW muotoisia kuvatiedostoja).

4.3.2 Tarkan raakamallin rakentaminen PhotoScanilla

Kohteen kuvaamisen jälkeen tuon kuvasarjan PhotoScan ohjelmaan. PhotoScan on Venäläisen Agisoftin kehittämä ohjelma, joka muodostaa fotogrammetrisin metodein kolmiulotteisia malleja digitaalisista kuvista. Kuvassa 10 on esitettynä PhotoScanin työvaiheet. Kallion seinämästä ei tarvitse luoda maskeja joten ylin kuva kivistä on toisesta mallista, loput kuvat ovat kallion seinämästä.

- Kuvien tuonti PhotoScaniin
- Maskien luominen
- Kuvien kohdentaminen
- Tiheän pistepilven rakentaminen
- Tahkoverkon rakentaminen
- Ylimääräisten palojen poistaminen
- .obj mallin vieminen



Kuva 10. PhotoScanin työvaiheet.

PhotoScanissa ensimmäisenä leikkasin kuvasarjan kuvista vähemmän tärkeät osat pois. Tässä työvaiheessa ohjelmalle kerrotaan maskin muodossa mikä kuvassa on tärkeää ja mikä tieto ohitetaan. Kuvat voi jättää myös ilman maskiakin, mutta tällöin ohjelma laskee jokaisen pikselin ja rinnastaa yhtäläisyyksiä muiden kuvien kanssa, mallintaa myös taustalla olevia turhia yksityiskohtia. Maskien tekeminen laskee täten merkittävästi seuraavan työvaiheen viemää aikaa sekä jälkisiivouksen määrää.

Maskien jälkeen ohjelman annetaan hetken laskea kuvissa olevia yhteisiä pisteitä ja rakentaa niistä karkea pistepilvi. Ohjelma laskee myös arvion kohdista, joista kuvat on otettu kohteeseen nähden. Tämä pistepilven ja kuvien orientaation luominen on nopea toimenpide ja vie nykyaikaisella tietokoneella minuutista kymmeneen minuuttia kuvien määrästä, maskeista sekä kuvien resoluutiosta riippuen. Tämän kohdan jälkeen voin osoittaa PhotoScanille haluamani raja-arvot mistä pistepilvistä haluan seuraavat työaskeleet tehtävän.

PhotoScan luo tiheän pistepilven antamieni rajojen mukaisesti. Tämä osio on kaikista aikaa vievin kokonaisuus. Minulla kesti aikaa jokaisen kohteen tiheän pistepilven luomiseen tunnista neljään tuntia.

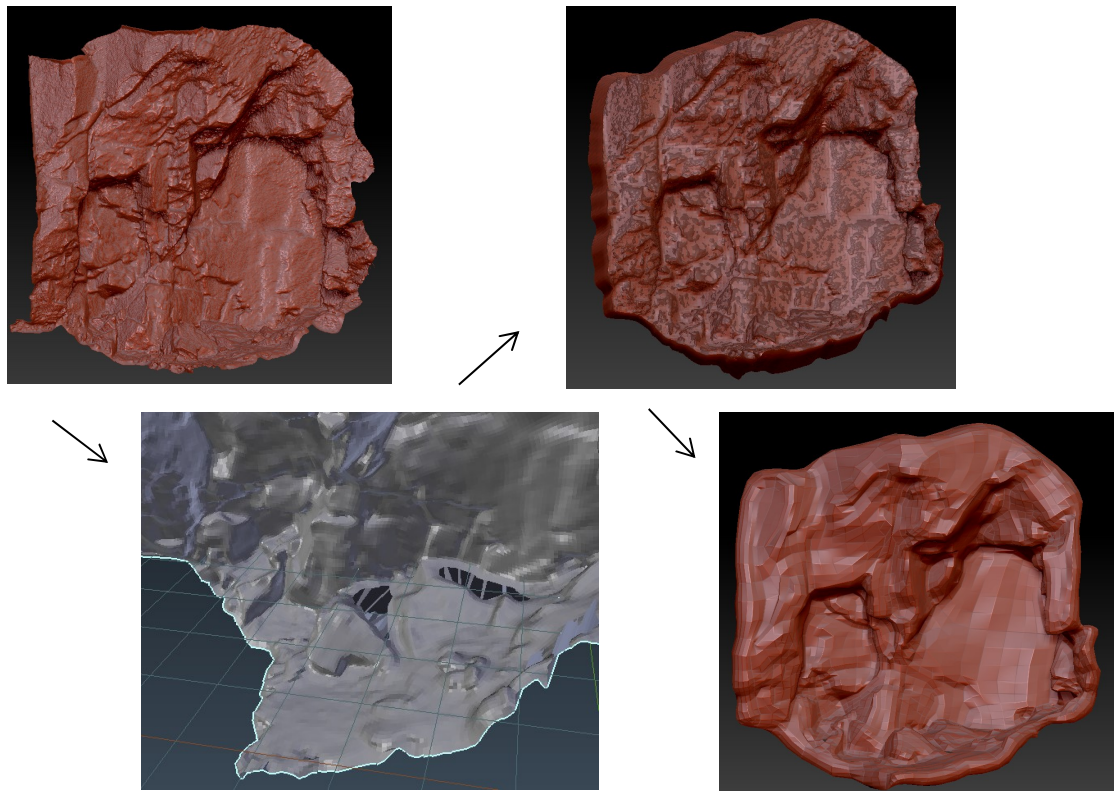
Tiheän pistepilven jälkeen PhotoScanille annetaan ohjeet rakentaa tiheästä pistepilvestä tahkoverkko. Tätä polygoniverkkoa voin lähteä työstämään ja muokkaamaan eteenpäin muilla kulmiulotteiseen mallintamiseen soveltuvilla työkaluilla. Tämä kohta vie aikaa minuuteista kymmeneen minuuttia. Tässä työvaiheessa poistan myös ylimääräiset polygoniryppäät, joita ohjelma muodostaa kuvien epäselvistä kohdista. Pääsääntöisesti näitä palasia ei tarvita, ja ohjelmassa onkin hyvä automaattinen työkalu niiden poistamiseen.

Seuraavaksi vien raskaan, lähes 1.5 miljoonan kolmion lopputuloksen ZBrushiin jälkikäsiteltäväksi. Tiedostomuodoksi käytin .obj joka osoittautui ongelmattomaksi valinnaksi kaikissa tekemissäni malleissa.

4.3.3 Mallin jälkikäsittely ja uusi topologia

PhotoScanin rakentama tarkka malli on erittäin yksityiskohtainen, mutta se on liian raskas käytettäväksi sellaisenaan pelissä. Tästä johtuen rakennan yksinkertaisemman mallin vanhan pohjalta ja heijastan tarkan mallin yksityiskohdat lopullisen päälle. Tässä työvaiheessa käytin pääsääntöisesti ZBrushia sekä Blenderiä.

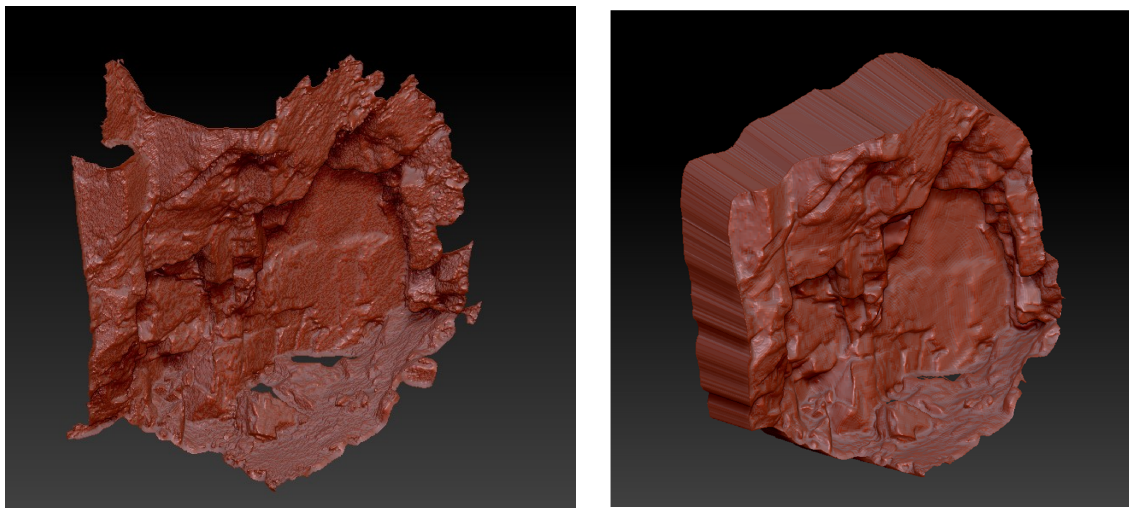
- Mallin tuonti ZBrushiin
- Kopio alkuperäisestä mallista
- Dynamesh reikien täyttämiseen
- Zremesh uuden perusmallin tekemiseen
- Yksityiskohtien heijastaminen uuteen malliin
- Reikien paikoille lisää yksityiskohtia



Kuva 11. Mallin jälkikäsittelyn työvaiheet.

Avaan PhotoScanin mallin ZBrushissa ja luon siitä ensimmäisenä kopion, sillä tarvitsen alkuperäistä mallia myöhemmin yksityiskohtien heijastamiseen. Käytän malliin aluksi ZRemesh työkalua, joka laskee ja rakentaa mallin uudelleen sulavaviivaisemmalla topologialla. Tämän kohdan tavoite ei niinkään ole vielä lopullinen topologia, vaan paremmat suuntaviivat Dynamesh työkalulle mallissa olevien reikien sekä takasivun täyttämiseen. Alkuperäisessä mallissa on lähes aina pieniä kohtia joita kuviin ei tallentunut, esimerkiksi raot, halkeamat tai liian korkealla tai matalalla olevat kohdat. PhotoScan jättää nämä kohdat mallintamatta, jättäen tahkoverkkoon reikiä.

Avaan mallin seuraavaksi Blenderillä, jossa luon sille paksuutta sekä siltaan reikiä edellä mainittua Dynamesh työkalua varten kuvan 12 mukaisesti. Dynamesh tekee mallista aina umpinaisen, joten ilman paksunnettuja seinämiä sekä ohjattuja siltauksia, lopputulos voi olla epämieluisasti itsensä päälle luhistunut möykky tahkoja.

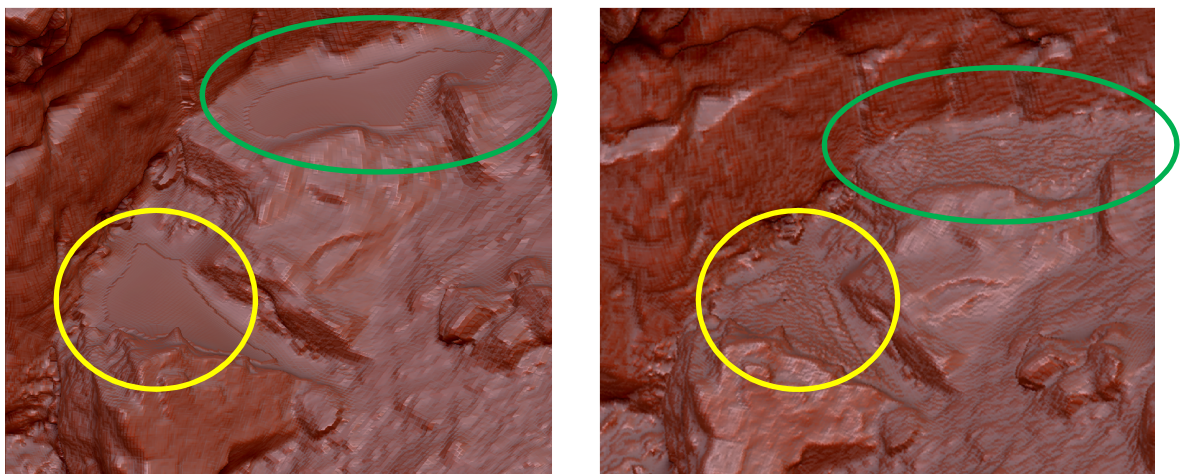


Kuva 12. Paksunnettu seinämä sekä reikien siltaus.

Tuon mallin takaisin ZBrushiin jossa viimein pääsen käyttämään Dynameshiä reikien sulkemiseen ja takaseinän luomiseen. Dynamesh täyttää kaikki reiät antamiani siltauksia apuna käyttäen ja lopputulos on tämän ansiosta fiksun näköinen. Dynamesh jakaa tahkoverkon myös tasaiseksi, joten joudun käyttämään vielä ZRemesh työkalua malliin uudelleen paremman topologian rakentamiseen. Tässä vaiheessa voin poistaa mallista Dynameshin luoman takaseinän sekä osat sivuista, mitä en enää usko tarvitsevani.

Jaan seuraavaksi mallin tahkojen määrän alkuperäisen PhotoScanin mallin lukemiin, eli tässä tapauksessa noin 1,5 miljoonaan. Tämän hetkiksessä mallissa minulla on erittäin hyvä pohja sekä kunnollinen topologia, joten haluan heijastaa vanhan mallin yksityiskohdat uuden aihion päälle.

Tässä vaiheessa malli näyttää lähes samalta miltä alkuperäinen, mutta siinä ei ole enää reikiä ja sillä on sulava topologia. Kaikki edelliset kohdat ovat pitkälti pohjatyötä tähän pisteeseen. Nyt voin tasoittaa kaikki jälkikäsitteilyä vaativat kohdat sekä reiät mitkä Dynamesh täytti. Tässä vaiheessa minulla on kaksi vaihtoehtoa, joko veistän itse tasaisiin kohtiin yksityiskohtia tai teen alpha-maskin, eräänlaisen sapluunan kallon toisisita kohdista. Alpha-maski on 16-bittinen mustavalkoinen kuvatiedosto, joka kuvastaa korkeusarvoja. Voin käyttää olemassa olevaa korkeustietoa muusta seinämästä ja kopioida sitä tasaisiin kohtiin alla olevan kuvan 13 mukaisesti.



Kuva 13. Reikäkohtia sekä niiden päälle veistettyjä yksityiskohtia.

Tässä vaiheessa kopioin mallin ja käytän vielä kerran ZRemesh työkalua tiputtaakseni lopullisen, pelimootorissa käytettävän objektin tahkomäärän halutulle tasolle. Tälle kallon kielekkeelle asetin tavoitteeksi 3500 kolmiota, mallin ollessa lopuksi 3722 kolmiota.

4.3.4 UV-Kartoitus

Mallin ollessa valmis seuraava työvaihe on sama kuin perinteisin menetelmin mallinnetuissa objekteissakin eli UV-kartoitus. UV-kartoitus määrittää kartoitetun pinta-alan käyttöasteen, joten hyvin tehty kartoitus takaa enemmän ja paremmin heijastettuja yksityiskohtia sekä pintatekstuureja. Käytin malleissani ZBrushin UV Master työkalua, sen ollessa helppokäyttöinen sekä tarjoavan riittävän hyvän lopputuloksen.

4.3.5 Normaalikartan rakentaminen

Tässä vaiheessa minulla on lopullinen pelissä käytettävä UV-kartoitettu 3722 tahkon malli sekä yksityiskohtainen 1.5 miljoonan tahkon malli. Normaalikarttoja tehdessäni käytän pääsääntöisesti ohjelmaa xNormal. xNormal on tehokas ja helppokäyttöinen sekä ilmainen ohjelma. Kokemukseni mukaan se tarjoaa hyvän näköisiä normaalikarttoja. xNormaliin ladataan matala- sekä korkeatahkoinen malli ja se laskee ja tallentaa tekstuurikarttana tarkan mallin heijastuskulmat epätarkan mallin UV-karttaan.

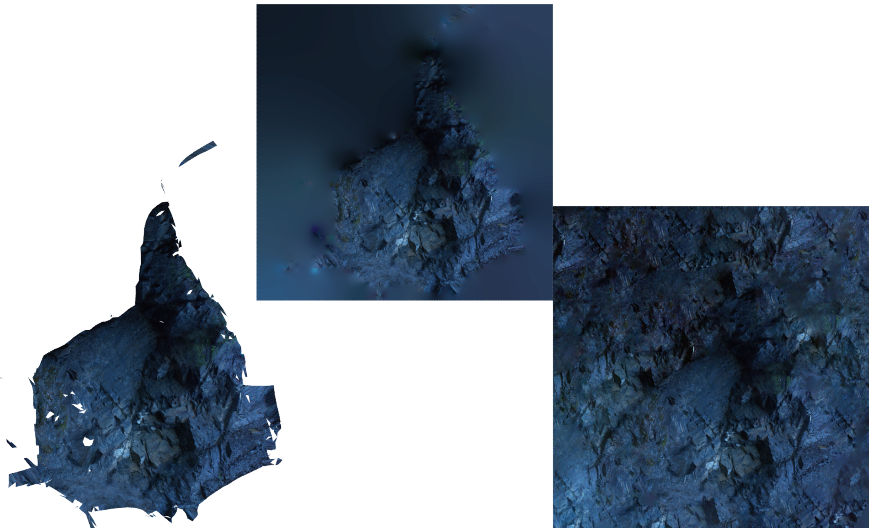
4.3.6 Tekstuurin heijastaminen PhotoScanissa

Diffuusitekstuurin tekemiseksi vien matalan tahkomäärän mallin PhotoScaniin ja korvaan alkuperäisen mallin. Tämän jälkeen minun pitää enää valita minkä kokoisena tiedostona haluan tekstuurini tallennettavan. Valitsen tähän tarkoitukseen 2048x2048 pikseliä, vaikka vähemmänkin voisi riittää tähän objektiin. Tässä kuten kaikessa muussakin ajattelen, että voin aina myöhemmin

pienentää kuvakokoa, mutta jos haluan suuremman niin minun pitää tehdä kaikki jälleen myöhemmin uudelleen.

4.3.7 Tekstuurin jälkikäsittely

Tämä työvaihe on hyvin tyypillistä tekstuurin jälkikäsittelyä, joka ei paljoa eroa muusta teksturoinnista. Käytän jälkikäsittelyyn Photoshopia sekä XnViewiä. Avaan PhotoScanin rakentaman tekstuurikartan ensin XnViewillä, sillä photoshop ei jostain syystä löydä ylivuotanutta väriä tekstuuritiedostosta, mutta XnView löytää. Tallennan tiedoston uudelleen, jonka jälkeen avaan sen Photoshopissa. Työstän kaikki reikäkohdat sekä ylivuotaneet kohdat hyvin nopeasti kloonaustyökalulla, jonka pohjana käytän jo olevaa tekstuuria hieman alpha-maskien tavoin. Lopputuotteet osoittavat että tämän työvaiheen kanssa ei tarvitse olla täydellisen tarkka, sillä yleiskuva pysyy hyvin samanlaisena, vaikka kuvaa kloonaa pieniin paikkoihin joissa ei tekstuuria vielä ole. Lopuksi tasoitan ylivalottuneet sekä epäselvät kohdat tekstuurista kuvan 14 mukaisesti.



Kuva 14. Alkuperäinen tekstuuri, ylivuodon sisältävä tekstuuri sekä lopullinen kloonaustyökalulla täytetty tekstuuri.

4.3.8 Mallin asettaminen pelimoottoriin

Lopuksi katson miltä malli näyttää pelimoottorissa. Valitsin pelimoottoriksi Unrealin Unreal Engine 4, sillä se on minulle kaikista tutuin moottori ja osaan työskennellä sen kanssa parhaiten.

Rakensin pelimoottorissa objektille oman materiaalin ja annoin tekemäni tekstuurin sekä normaalikartan materiaalin arvoiksi. Lisäsin myös specular sekä roughness arvoiksi tasaiset lukemat, jotka näyttivät hyvältä materiaalin esikatseluikkunassa. Alla kuvassa 15 lopullinen kallion seinämä Unreal Engine 4 pelimoottorissa.



Kuva 15. Lopullinen objekti pelimoottorissa.

5 POHDINTA

Tavoitteena minulla oli luoda pelimoottorissa toimivia objekteja asettikirjastooni, sekä tutustua fotogrammetriaan peligrafiikan työkaluna. Fotogrammetria on vielä suhteellisen uusi tuttavuus peligraafikon arkipäiväisenä työkaluna, joten lähdekirjallisuuden määrä sekä laatu vaihtelevat fotogrammetriasta pelikäytössä. Kuten monessa muussakin grafiikan osiossa myös fotogrammetriassa graafikoiden työtavat eroavat tekijänsä mukaan, tarjoten erilaisia lähestymistapoja samankaltaisen lopputuloksen saavuttamiseksi. Tästä tiedon määrästä pitääkin rakentaa itselle sopivin kokonaisuus.

Koen saavuttaneeni taroitteeni mallejen rakentamisesta fotogrammetrisin menetelmin samalla tutustuen sekä oppien paljon uusia asioita niin mallintamisesta kuin ohjelmistosta kuten Zbrushista sekä PhotoScanista.

Olen tyytyväinen tekemiini asetteihin, sekä niiden yleiseen laatuun. Automaattisten työkalujen käytön takia asetteihin jäi vielä mahdollisuuksia optimointiin esimerkiksi tahkomäärissä sekä UV-kartoituksessa, mutta ne ovat siltikin käyttökelpoisia peliin asetettavaksi.

Kyseisten asettien täysin omin käsin tekemiseen menisi huomattavasti enemmän aikaa ja niiden tekeminen vaatisi myös merkittävää taitotasoa, jota minulla ei vielä ole. Yhden asettin tekemiseen menee kahdesta kahdeksaan tuntia ja siitäkin tiheän pistepilven laskemiseen noin puolet, joten useammalla tietokoneella työskennellessä tämä on nopea tapa luoda korkealaatuisia asetteja pelieihin.

Suurin virheen lähde työskennellessäni olin minä itse, sekä kamerakalustoni. Työskennellessä kuvien laatuun tulee mielestäni kiinnittää eniten huomiota. Riittävä määrä tarpeeksi tarkkoja kuvia vähentää jälkikäsitteilyn määrää huomattavasti. Myös korkeamman megapikselimäärän kamera mahdollistaa korkeamman kiintopisteiden lukumäärän, näin tehden malleista tarkempia ja tekstuureista yksityiskohtaisempia.

Tarkoittaako fotogrammetrian kehittyminen nyt sitä, että perinteisiltä 3d-mallintajilta tulee loppumaan työt? Ei todellakaan, fotogrammetria on tarkoitettu jo olemassa olevien esineiden ja objektien tuomiseen tietokoneen ruudulle. Joudumme edelleen mallintamaan sekä teksturoimaan keksityt asiat, kuten hirviöt tai tulevaisuuden laser aseet vanhoilla mallintamisen menetelmillä. Fotogrammetrisin menetelmin tehdyt mallit tulevat myös vielä nykyisessä muodossaan vaatimaan ihmisen tekemää jälkikäsitteilyä. Uskon fotogrammetrian käytön kuitenkin yleistyvän peligrafiikassa. Jatkuva realismin tavoittelu saadaan kuitenkin parhaiten tallennettua itse reaali maailmasta.

LÄHTEET

Agisoft LLC. 2015. PhotoScan – Fully automated professional photogrammetric kit. http://www.agisoft.com/pdf/photoscan_presentation.pdf Luettu 9.6.2016.

Art Asset Best Practice Guide. (2015). Haettu 5.11.2015 sivustolta Unity Documentation <http://docs.unity3d.com/Manual/HOWTO-ArtAssetBestPracticeGuide.html>

Blender. <https://www.blender.org/>

Blizard, B. 19.2.2014. The Art of Photogrammetry: How To Take Your Photos. Tested. Haettu osoitteesta. <http://www.tested.com/art/makers/460142-art-photogrammetry-how-take-your-photos/>. Luettu 21.6.2016.

Center for Photogrammetric Training. History of Photogrammetry. Luettu 7.6.2016 <https://spatial.curtin.edu.au/local/docs/HistoryOfPhotogrammetry.pdf>

Fryer, J., Mitchell, H & Chandler, J. 2007. Applications of 3D Measurement from Images. Whittles Publishing.

Graphine. Photogrammetry in games. <http://graphinesoftware.com/our-technology/photogrammetry-games>. Luettu 9.6.2016.

Hajioannou, Y. (2015, 25. tammikuuta). Gamedev Glossary: What Is a “Normal Map”? Haettu 11.11.2015, sivustolta Tuts+ <http://gamedevelopment.tutsplus.com/articles/gamedev-glossary-what-is-a-normal-map--gamedev-3893>

Luhmann, T., Robson, S., Kyle, S & Harley, I. 2011. Close Range Photogrammetry: Principles, Techniques and Applications. Whittles Publishing.

Masters, M. Start Mastering Important 3D Texturing Terminology. (2013) Haettu 11.11.2015 sivustolta: Digitaltutors <http://blog.digitaltutors.com/cover-bases-common-3d-texturing-terminology/>

Matthews, N. A. 2008. Aerial and Close-Range Photogrammetric Technology: Providing Resource Documentation, Interpretation, and Preservation. Technical Note 428. U.S. Department of the Interior, Bureau of Land Management, National Operations Center, Denver, Colorado. 42 pp. Luettu 7.6.2016. <http://www.blm.gov/nstc/library/pdf/TN428.pdf>

PhotoScan. <http://www.agisoft.com/>

Photoshop. <http://www.adobe.com/products/photoshop.html>

Poznanski, A. 2014. Visual Revolution of The Vanishing of Ethan Carter. <http://www.theastronauts.com/2014/03/visual-revolution-vanishing-ethan-carter/> Luettu 7.6.2016.

Simonds, B. 2012. Blender Master Class: A Hands-On Guide to Modeling, Sculpting, Materials, and Rendering (1). No Starch Press, US.

Stifter, A. (2014, 23, syyskuuta) Diffuse Maps. Haettu 6.7.2016 sivustolta CRYENGINE MANUAL: <http://docs.cryengine.com/display/SDKDOC2/Diffuse+Maps>

Strong, B. 2007. Creating Game Art for 3D Engines. Course Technology.

The Gnomon Workshop. (24.5.2016). An Evening with Alex Alvarez: Creating Natural 3D Environments. Haettu osoitteesta <https://www.thegnomonworkshop.com/blog/an-evening-with-alex-alvarez-creating-natural-3d-environments>

Thorn, A. 2013. Game Development Principles. Cengage Learning PTR.

Thorn, A. 2014. Practical Game Development with Unity and Blender. Cengage Learning PTR.

Totten, C. 2012. Game Character Creation with Blender and Unity (1). Sybex.

Vinni, P. 2003. Kuva kertoo enemmän kuin tuhat sanaa – Mitä on fotogrammetria? <http://www.kotikone.fi/faryan/Teksteja/JatkokurssiB.htm> Luettu 6.6.2016.

Walford, A. (2007). What is Photogrammetry? Haettu 6.6.2016 sivustolta Photogrammetry: <http://www.photogrammetry.com/index.htm>

xNormal. <http://www.xnormal.net/>

XnView. <http://www.xnview.com/en/>

Zbrush. <http://pixologic.com>

Kuvat

Kuva 1. Fotogrammetrian periaate. Haettu sivustolta http://www.clemson.edu/restoration/wlcc/equipment_services/equipment/photogrammetry.html

Kuva 2. Kärki ja sen paikka xy-koordinaatistolla. Totten, C. 2012. Game Character Creation with Blender and Unity (1). Sybex.

Kuva 3. Kahden kärjen väliin muodostuva särmä, sekä kolmen kärjen väliin muodostuva tahko. Totten, C. 2012. Game Character Creation with Blender and Unity (1). Sybex.

Kuva 4. UV-kartoituksen periaate.

Kuva 5. Mallin värin sisältävä diffuusikartta, heijastus suunnat sisältävä normaalikartta sekä valon määrän kertova valokartta.

Kuva 6. Alvarezin tekemä maasto, jonka luonnissa on käytetty fotogrammetriaa. Haettu sivustolta <https://www.thegnomonworkshop.com/blog/an-evening-with-alex-alvarez-creating-natural-3d-environments>

Kuva 7. Aukkoarvo, valotusaika sekä ISO-herkkyys vaikuttavat kuvien lopputulokseen. Haettu sivustolta <http://lifelhacker.com/how-aperture-shutter-speed-and-iso-affect-pictures-sh-1699204484>

Kuva 8. Valaistuksen vaikutus lopputulokseen.

Kuva 9. Patsaan ympäriltä otettu kuvasarja, jossa jokainen sininen neliö kuvastaa kohtaa josta kuva on otettu.

Kuva 10. PhotoScanin työvaiheet.

Kuva 11. Mallin jälkikäsittelyn työvaiheet.

Kuva 12. Paksunnettu seinämä sekä reikien siltaus.

Kuva 13. Reikäkohtia sekä niiden päälle veistettyjä yksityiskohtia.

Kuva 14. Alkuperäinen tekstuuri, ylivuodon sisältävä tekstuuri sekä lopullinen kloonaustryäkalulla täytetty tekstuuri.

Kuva 15. Lopullinen objekti pelimoottorissa.