



TAMPEREEN
AMMATTIKORKEAKOULU

PROSEDURAALISEN TEKSTUURIN LUOMI- NEN SUBSTANCE DESIGNERILLA

Kimmo Lehtovaara

Opinnäytetyö
Huhtikuu 2017
Tietojenkäsittely
Pelituotanto



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely
Pelituotanto

LEHTOVAARA, KIMMO:

Proseduraalisen tekstuurin luominen Substance Designerilla

Opinnäytetyö 32 sivua, joista liitteitä 1 sivu
Huhtikuu 2017

Opinnäytetyön tavoitteena oli syventää osaamista Substance Designer -teksturointityökalun käytössä ja tuottaa suomenkielistä opetusmateriaalia Substance Designer -ohjelman käytön perusteista. Kirjoitushetkellä suurin osa kaikesta internetistä löytyvästä ohjelmaa koskevasta opetusmateriaalista oli englanninkielistä. Työ tehtiin toimeksiantona TAM-Kin tietojenkäsittelyn pelituotanto-linjalle.

Opinnäytetyön teoriaosuudessa käytiin läpi Substance Designerin käyttöön liittyviä tärkeitä asioita. Teoriaosuudella pyrittiin pohjustamaan tarvittavia tietoja ohjelman käytön aloittamiseen. Työn liitteeksi tuotettiin kolme opetusvideota Substance Designer -ohjelman käytöstä. Opetusvideot lisättiin Youtube-palveluun ja linkki videoihin lisättiin tämän opinnäytetyön liitteisiin.

Substance Designer on todella tehokas teksturointityökalu. Työhön toteutetussa opetusvideoiden sarjassa keskityttiin ohjelman perusteisiin. Aiheesta kiinnostunut voisi kehittää aihetta syventymällä Substance Designerin edistyneempiin toimintoihin.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Business Information Systems
Game Development

LEHTOVAARA, KIMMO:
Procedural Texture Creation with Substance Designer

Bachelor's thesis 32 pages, appendices 1 page
April 2017

The purpose of this thesis was to deepen the author's experience in the usage of the Substance Designer texturing tool and to produce educational material in Finnish about the basics of using Substance Designer. At the time this thesis was being written, almost all educational material found online was in English. This thesis was commissioned by the Game Development program at Tampere University of Applied Sciences.

The theoretical section of this thesis lays out the basics needed to better understand Substance Designer. Three educational videos about the basics of Substance Designer were uploaded to YouTube as a part of the thesis. The web address for the videos was added to the appendices of this thesis.

Substance Designer is a very powerful texturing tool. The series of videos created for this thesis covered the basics of Substance Designer. The topic can be expanded upon by going deeper into the program and its more advanced features.

Key words: substance designer, procedural, texturing

SISÄLLYS

1	JOHDANTO.....	5
2	TEKSTUROINTI	6
3	ERILAISET TEKSTUURIKARTAT	10
4	PROSEDURAALISET TEKSTUURIT	15
5	PBR – Physically-Based Rendering	17
6	SUBSTANCE DESIGNER	21
7	OPETUSVIDEOIDEN TOTEUTUS	25
8	POHDINTA.....	27
	LÄHTEET	29
	LIITTEET	32
	Liite 1. Linkki opetusvideoiden soittolistaan YouTube-palvelussa.	32

1 JOHDANTO

Tämän opinnäytetyön tarkoitus on tuottaa suomenkielistä opetusmateriaalia Substance Designer teksturointityökalun käytöstä. Ohjelman käytöstä löytyy pääasiassa vain englanninkielistä materiaalia. Opetusmateriaalit tehdään videomuodossa ja julkaistaan YouTube-palvelussa. Luomalla opetusmateriaalia Substance Designerin käyttöön, syvennän samalla omaa osaamistani ohjelmaan ja teksturointiin liittyen.

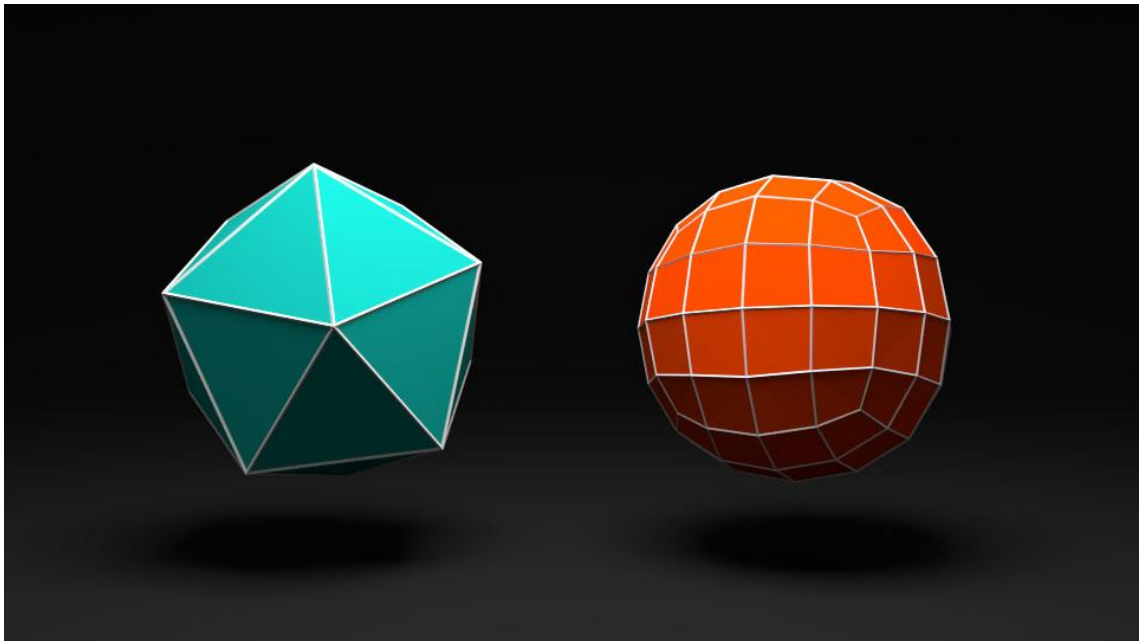
Työ on tehty toimeksiantona Tampereen Ammattikorkeakoulun Tietojenkäsittelyn Peli-tuotantolinjalle. Kirjoitushetkellä TAMK:ssa ei vielä opeteta Substance Designerin käyttöä, mutta ohjelman lisenssien hankinnasta on puhuttu. Ohjelma on tullut myös opiskelijoille ilmaiseksi syksyn 2016 aikana. Minulta itseltäni löytyy Substance Designer Indie lisenssi.

Työn kirjallinen osuus toimii pohjana opetusvideoille. Linkki opetusvideoiden YouTube soittolistaan löytyy opinnäytetyön liitteenä.

2 TEKSTUROIINTI

Teksturoidi tarkoittaa 2D-kuvien tekemistä käytettäväksi 3D-mallien kanssa. Näitä kuvia kutsutaan tekstuurikartoiksi. (Polycount 2016.) Tekstuureilla tuodaan lisää eloa 3D-malleille. Hyvillä tekstuureilla saadaan luotua paljon yksityiskohtia yksinkertaiselle 3D-mallille ja huonosti tehdyillä tekstuureilla voidaan myös pilata hyvä malli. Teksturoidi onkin tärkeä vaihe 3D-peliobjektien luomisessa.

Peleissä käytetyt 3D-mallit koostuvat polygoneista eli monikulmioista. Polygonit muodostuvat toisiinsa yhdistetyistä pisteistä 3D-avaruudessa. Yksinkertaisin mahdollinen polygoni on kolmio, joka luodaan yhdistämällä kolme pistettä toisiinsa. 3D-mallit koostuvat useista polygoneista, jotka muodostavat yhdessä polygoniverkon. (Munoz 2015a.) Tekstuurit täytyy kartoittaa, jotta ne saadaan näkymään 3D-mallin pinnalla.



KUVA 1. Polygoniverkko (Munoz 2015b)

Tekstuurit saadaan kartoitettua 3D-mallin pinnalle käyttämällä erilaisia projektiotapoja, kuten esimerkiksi laatikko, pallo tai sylinteri -projektiota. Näistä eri tavoista valitaan se, joka on lähimpänä 3D-mallin geometriaa. Projektio on tehokas tapa teksturoida yksinkertaisia muotoja. Yksityiskohtaisempaan malliin projektiolla ei saa kuvaa ilman tekstuurin venymistä. Tekstuurin saa kartoitettua monimutkaisempien mallien pinnalle käyttämällä UV kartoitusta. (Chopine 2011, 152–153.)

UV kartoitus, englanniksi UV mapping, on tapa purkaa 3D-malli tasaiselle pinnalle. Hyvä esimerkki tästä on karkkipaperin suoristaminen. Tarkoitus tällä on luoda yhteys 3D-mallin ja 2D-tekstuurin välille, jotta tekstuuri voidaan projektoida 3D-mallin pinnalla. Ennen mallin purkamista täytyy polygoniverkkoon tehdä leikkauksia eli saumoja, jotta se saadaan avattua ja levitettyä tasaiseksi. (Thorn 2014, 89–90.) Hyvän UV kartan perustana on tehdä siitä helposti tunnistettava käyttämällä mahdollisimman vähän saumoja. Saumat kannattaa sijoittaa sellaisiin kohtiin, joista niitä ei ole helppo huomata.

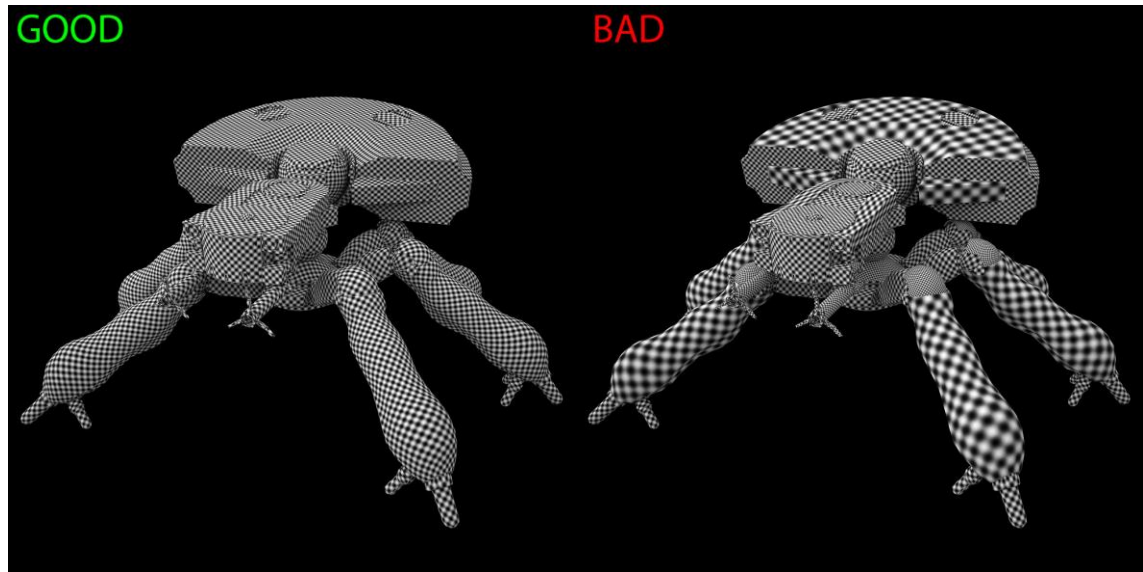
UV kartoituksen jälkeen 3D-mallin voi teksturoida, joko 3D-ohjelman sisällä tai tallentamalla UV kartan kuvana ja avaamalla sen kuvankäsittelyohjelmalla. Kuvassa 2 on 3D-malli, jonka tekstuurikartasta huomataan, että mallissa käytetään peilausta. Peilaamalla säästetään aikaa mallinnuksessa ja mikä tärkeintä, tilaa tekstuurikartassa. Ilman peilausta tekstuurikartan resoluutiota täytyisi nostaa, jos halutaan sama määrä yksityiskohtia, kuin kuvasta tällä hetkellä löytyy.



KUVA 2. 3D-malli tekstuurin kanssa ja ilman (Henriksen 2015)

UV karttaa luodessa kannattaa asettaa kartan taustalle shakkilauta tekstuuri. Tällä tavoin shakkilaudan ruutujen koko antaa hyvän indikaattorin polygoniverkon tekseli tiheydestä. Kaikkien ruutujen pitäisi olla neliskulmaisia ja samankokoisia joka puolella polygoniverkkoa. Tällä tavoin varmistetaan parhain mahdollinen laatu tekstuurille, sillä kaikki polygoniverkon osat saavat kokoonsa nähden tarpeeksi pikseleitä. (Thorn 2014, 95.) Tekseli tiheyttä voi toki myös nostaa tai laskea alueen tärkeydestä riippuen. Esimerkiksi ihmishahmon kasvoille voi antaa enemmän tilaa tekstuurikartassa, jotta kasvoihin

mahtuu tarkempia yksityiskohtia. Tiheyden ero ei saa kuitenkaan olla liian suuri, ettei vierekkäisten tekstuurien resoluution muutos ole huomiota herättävä.



KUVA 3. Esimerkki hyvästä ja huonosta tekseli tiheydestä (Quixel 2015)

Tekstuurien koon tulisi vastata objektien kokoa. Sopivaa resoluutiota päätettäessä täytyy ottaa huomioon se, miten näkyvä teksturoitava objekti on pelissä. Kaukana olevalle objektille ei kannata antaa liian suuren resoluution tekstuuria ja lähellä olevalle objektille ei kannata antaa liian pientä tekstuuria. Videopeleissä tekstuurien koko kannattaa pitää kahden potensseissa, eli 256x256, 512x512, 1024x1024 -pikseliä ja niin edelleen. Tällä tavalla optimoidaan tekstuurien renderöinnin suorituskykyä. Näin tekstuurien koot pysyvät pieninä ja renderöinti tehokkaana. Tämä voi aiheuttaa ongelmia pienen resoluution kanssa. Mikäli tekstuuri pikselöityy, niin täytyy resoluutiota nostaa. Tästä syystä saumattomat toistuvat tekstuurit ovat hyödyllisiä. Voit saada paljon resoluutiota toistamalla pientä kuvaa monta kertaa. (Chopine 2011, 163–164.)



KUVA 4. Esimerkki selvästä erosta resoluutioiden välillä (Nash 2009)

3 ERILAISET TEKSTUURIKARTAT

Artisteilla on nykypäivänä käytössä useita erilaisia tekstuurikarttoja. Tekstuureilla tuodaan lisää eloa harmaille elottoman näköisille 3D-malleille. Tekstuurikartoilla muokataan muun muassa 3D-mallin väriä, pinnan muotoja ja heijastuksia. Tähän lukuun on koottu erilaisia tekstuurikarttoja keskittyen siihen, miten osa kartoista toimii PBR-varjostimen kanssa. PBR:stä lisää opinnäytetyön viidennessä luvussa.

Perusvärikartta (basecolor map) sisältää ei-metallien heijastaman värin ja puhtaiden metallien heijastusarvot. Ei-metalleja eli eristeitä kuvaava väri on pinnasta heijastuva väri. Puhtaiden metallien heijastusarvot ovat esillä kartan niillä alueilla, jotka on merkitty metallikartassa puhtaaksi metalliksi. Näillä metalliksi määritetyillä alueilla heijastusarvot täytyy ottaa oikean maailman arvoista. Metalliksi määritettyjen alueiden väriarvo pitääkin olla välillä 180-255 sRGB. Perusvärikartta on usein väriskaalaltaan diffuusiokarttaa suppeampi. Eristeitä kuvaavien väriarvojen ei kannata olla liian kirkkaita tai liian tummia. Esimerkiksi valkoinen lumi ei ole täysin valkoista ja hiili ei ole täysin mustaa. Perusvärikartassa eristeiden väriarvojen kannattaa pysyä tietyn rajan sisällä. Tummat värit eivät saisi mennä arvoltaan alle 30 sRGB ja kirkkaat värit eivät saisi mennä 240 sRGB ylitse. Perusvärikartan tulisi olla täysin vailla valoinformaatiota, paitsi joissain tapauksissa voi lisätä mikro-sulkeutumaa, jos pelkkä ympäristön sulkeutuma -kartta ei riitä. (McDermott 2016b, 5–7.)

Diffuusiokartta (diffuse map) on yleisesti ottaen tekstuuri, joka sisältää vain väri-informaatiota. Alkujaan peliartisteilla oli käytössä vain diffuusiokartta. Tästä syystä värikarttaan maalattiin värin lisäksi varjoja ja heijastuksia. Nykyään on käytössä monia eri tekstuurikarttoja, joita yhdessä käyttämällä saadaan lopullinen materiaali. Tämän ansiosta värikarttaan ei tarvitse enää lisätä varjostuksia ja heijastuksia. (Ahearn 2008, 75–76.)

Värikartoista puhuttaessa käytetään joskus albedo nimikettä. Diffuusiokartassa eristeille annetaan väri ja puhtaat metallit merkitään mustalla. Diffuusiokartan puhtaat metallit merkitään täysin mustaksi, mutta eristeille toimii perusvärikartalle suositeltu raja 30 - 240 sRGB. (McDermott 2016b, 13.) Sébastien Lagarden mukaan diffuusiokartoista tehdään usein liian tummia. Liian tummat väriarvot voimistavat spekulari- ja diffuusiioheijastusten eroa ja tekevät spekulariheijastuksista liian voimakkaita. (Lagarde 2011.)

Metallikartta (metallic map) määrittää sen mitkä tekstuurin osat ovat puhdasta metallia. Se on mustavalkoinen kartta, jossa mustat alueet kuvaavat ei-metallia ja valkoiset puhdasta metallia. Metallikartta toimii maskin tavoin ja kertoo varjostimelle, miten sen pitäisi tulkita perusvärikärttaa. Varjostin vertaa metallikartan valkoisia alueita perusvärikärtan vastaaviin alueisiin ja määrittää metallin heijastusarvot. Maalattua tai ruostunutta metallia ei lasketa puhtaaksi metalliksi vaan eristeeksi, eli ne merkitään metallikarttaan mustalla. Puhdasta metallia kuvaavat väriarvot täytyy olla 235-255 sRGB välillä. Metallin pinnalla olevaa ohutta pölykerrosta tai likaa voidaan kuvata kartassa harmaansävyillä. (McDermott 2016b, 7–9.)

Spekulaarisuuskartta (specular map) määrittää metallien heijastusarvot sekä ei-metallien fresnel-heijastukset 0-asteen kulmassa. Spekulaarisuuskartta on RGB kartta, johon ei-metallit merkitään harmaansävyillä ja metallia kuvaavat alueet voidaan merkitä metallin heijastamalla värillä. Spekulaarisuuskartassa ei-metallien eli eristeiden heijastuksia voi säätää harmaansävyjä muokkaamalla. Heijastusarvojen täytyy pohjautua oikean maailman arvoihin. Useimpien eristeiden heijastusarvot ovat 40-75 sRGB välillä. Metallien heijastusarvot ovat 180-255 sRGB välillä. (McDermott 2016b, 14–16.)



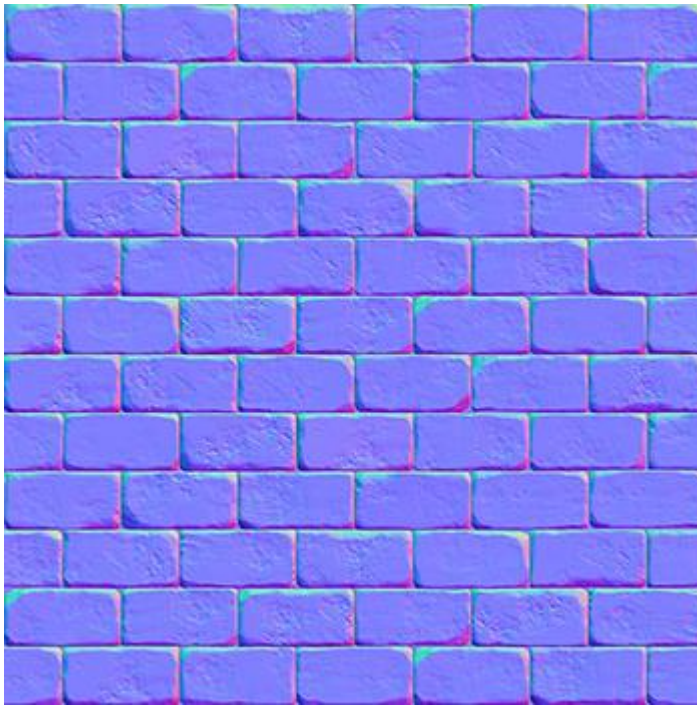
KUVA 5. Specular ja Metallic -työtavat (Wilson 2015a)

Karkeuskartta (roughness map) määrittää pinnan epäsäännöllisyyksiä, jotka aiheuttavat valon diffuusiota. Karkeilla pinnoilla on suuremmat ja himmeämmät heijastukset. Sileillä pinnoilla heijastukset näyttävät selkeämmiltä ja ne voivat näyttää kirkkaammilta, vaikka heijastetun valon määrä on sama, kuin karkealla pinnalla. Karkeuskartta on mustavalkoinen kartta, jossa musta kuvaa sileää pintaa ja valkoinen karkeaa pintaa. Karkeuskartta antaa artistille vapaat kädet luoda minkäläisen pinnan vain haluaa. (McDermott 2016b, 10.) Karkeuskartta kuuluu Metallic/Roughness -työskentelytapaan PBR-järjestelmässä.

Kiiltokartta (glossiness map) toimii karkeuskartan tavoin, mutta se on käänteinen siihen nähden. Musta väri kuvaa kiiltokartassa karkeaa pintaa ja valkoinen väri sileää pintaa. (McDermott 2016b, 16.) Kiiltokartta kuuluu Specular/Glossiness -työtapaan PBR-järjestelmässä.

Kuhmukartta (bump map) on mustavalkoinen kartta, jolla voidaan lisätä simuloitua syvyyttä tai korkeutta 3D-mallin pinnalle. Kartassa musta kuvaa matalinta tasoa ja valkoinen korkeinta. Kuhmukartta ei muuta objektin geometriaa tai silhuettia ja se toimii vain kameran suuntaisesti. Kuhmukartta on riittävä vaihtoehto mataliin muutoksiin 3D-mallin pinnassa. (Chopine 2011, 159.)

Normaalikartta (normal map) on uudempi, parempi versio kuhmukartasta. Normaalikartta käyttää RGB väri-informaatiota, joka vastaa 3D-mallin XYZ -koordinaatteja 3D-avaruudessa. RGB väri-informaatio kertoo 3D-ohjelmalle pinnan normaalien suunnan jokaisen polygonin kohdalla. (Pluralsight 2014.) Tällä tavoin pinnoille saadaan tarkempia kuhmuja. Normaalikartta simuloi pinnan geometriaa kuhmukartan tavoin, mutta pinnan muodot simuloidaan kaikkiin suuntiin pelkän kameran sijasta. Normaalikarttakaan ei muuta objektin geometriaa eli mallin silhuetti säilyy muuttumattomana. (Chopine 2011, 160.)



KUVA 6. Tiiliseinän normaalikartta (Vries 2017)

Korkeuskartta (height map) on normaalikartan kaltainen, mutta sen käyttö vie koneelta enemmän suoritustehoa. Korkeuskarttaa käytetään usein yhdessä normaalikartan kanssa. Käyttämällä karttoja yhdessä saadaan pinnan muotoihin lisää tarkkuutta. Korkeuskartta eroaa normaalikartasta siirtämällä tekstuurin pintaa saadakseen aikaiseksi sulkeuma efektiin tekstuurin pinnalle. Korkeuskartta on mustavalkoinen kartta, jossa valkoinen kuvaa korkeita alueita ja musta matalia alueita. (Unity 2016a.)

Siirtymäkartta (displacement map) muuttaa 3D-mallin geometriaa eli sen silhuettikin muuttuu. Siirtymäkartta on kuhmukartan tavoin mustavalkoinen.

Luodakseen yksityiskohtia siirtymäkartalla täytyy polygoniverkon sisältää tarpeeksi geometriaa. Kaiken tämän ylimääräisen geometrian luominen reaaliajassa on erittäin hankalaa ja rasittavaa tietokoneelle. Siirtymäkartan käyttö lisää renderöintiäikää huomattavasti. (Pluralsight 2014.)

Ympäristön sulkeutuma -kartalla ilmennetään alueet 3D-mallissa joiden pitäisi saada paljon tai vähän epäsuoraa valoa. Epäsuora valo saapuu ympäröivästä valosta ja heijastuksista, joten mallin syvät ja koverat kohdat, kuten raot ja taitokset eivät realistisesti saisi paljoa epäsuoraa valoa. Ympäristön sulkeutuma -kartta, englanniksi ambient occlusion map, lyhyemmin AO, on mustavalkoinen kartta, jossa valkoiset alueet saavat kaiken ympäröivän epäsuoran valon ja mustat alueet eivät saa valoa ollenkaan. (Unity 2016b.)



KUVA 7. Ympäristön sulkeutuma -kartta (Wilson 2015b)

Läpinäkyvyyskarttaa (opacity map) käyttämällä voi tehdä 3D-mallista osittain läpinäkyvän tai täysin näkymättömän. Läpinäkyvyyskartta on mustavalkoinen kartta, jossa valkoisella kuvataan täysin näkyvissä olevaa aluetta ja mustalla kuvataan täysin läpinäkyviä alueita. (Polycount 2015.)

Tiheyskartassa (thickness map) mustat alueet kuvaavat 3D-mallin ohuita kohtia ja valkoiset paksuja. Tiheyskartan voi luoda suoraan 3D-mallin pohjalta. Sitä voidaan käyttää alitason hajontaa eli Sub Surface Scattering efektiä hyödyntävässä varjostimessa tai efektiä voi huijata suoraan diffuse/albedo tekstuurissa. (Allegorithmic 2017b.) Alitason hajonta tarkoittaa sitä efektiä, jossa valo siroaa kulkiessaan läpikuultavan aineen lävitse. (Epic 2017.) Valon näkyminen korvan tai sormien läpi on hyvä esimerkki alitason hajonnasta.

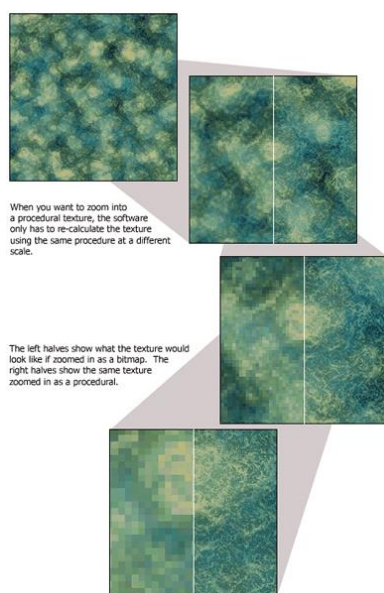


KUVA 8. Tiheyskartta 3D-mallissa (Allegorithmic 2017c)

4 PROSEDURAALISET TEKSTUURIT

Proseduraaliset tekniikat ovat tärkeitä työkaluja realististen grafiikoiden ja animaation luomisessa elokuvia ja videopelejä varten. Proseduraaliset tekniikat ovat koodi segmenttejä tai algoritmeja, jotka määrittävät millainen tietokoneella generoitu 3D-malli tai efekti on. Proseduraalinen marmori tekstuuri ei esimerkiksi määritä väriarvojaan käyttämällä skannattua kuvaa marmorista, vaan se määrittää värinsä matemaattisten funktioiden ja algoritmien avulla. Yksi tärkeimmistä ominaisuuksista proseduraalisissa tekniikoissa on abstraktio. Proseduraalisessa menettelytavassa yksityiskohtaisia tietoja ei ennalta määrätä vaan ne kiteytetään funktion tai algoritmin sisälle. Tällä tavalla säästetään tallennustilaa, sillä tietoa ei ole määrätty ennalta vaan se generoidaan sitä tarvittaessa. (Ebert ym. 2003, XX, 1–2).

Proseduraaliset tekstuurit vievät paljon vähemmän tilaa, kuin tavalliset tekstuurit. Ko-koero tekstuurien välillä voi olla jopa 100-kertainen. Proseduraalisille tekstuureille voi lisätä parametreja, joita helposti muokkaamalla saadaan aikaiseksi paljon erilaisia efektejä. Tavallisen tekstuurin muokkaus ei ole yhtä vaivatonta, kuin proseduraalisen tekstuurin muokkaaminen. Proseduraalisilla tekstuureilla ei ole kiinteää resoluutiota, kuten tavallisilla tekstuureilla. Algoritmien ansiosta proseduraalisen tekstuurin voi lisätä kaiken kokoisille objekteille. Tarkastelemalla tavallista tekstuuria lähemmin, voit nähdä yksityiskohtien pikselöitymistä, jota ei näy proseduraalista tekstuuria käytettäessä. (Kessenich, Sellers & Shreiner 2016.)



KUVA 9. Tavalliseen ja proseduraaliseen tekstuuriin zoomaus (McCombs 2017)

Proseduraalisten tekstuurien käyttö vaatii algoritmin koodattuna käytettävään ohjelmaan. Kaikilla ei ole tarvittavaa teknistä osaamista algoritmien ohjelmointiin. Tavallisten tekstuurien tekeminen ei vaadi yhtä paljoa teknistä osaamista. (Kessenich ym. 2016.) Onneksi proseduraalisten tekstuurien tekoon löytyy myös työkaluja, joiden käytössä ei tarvitse osata ohjelmointia tai tietää algoritmeista.

Proseduraalista tekstuuria luodessa kannattaa ottaa huomioon se, että ei tee tekstuuriin mitään liikaa huomiota herättäviä kohtia. Katsoja tietää heti, että samaa tekstuuria toistetaan, jos tekstuurista erottuu tietty kohta liikaa. Tekstuurin muotojen kannattaakin olla hienovaraisia eikä liian silmiinpistäviä. Ilmiselvä tekstuurin toistuminen voi häiritä pelaajaa niin paljon, että pelaajan immersio pelimaailmaan katkeaa.

5 PBR – Physically-Based Rendering

Physically-Based Rendering, PBR, on renderöinti ja varjostus tapa, jolla saadaan aikaiseksi tarkempi kuvaus siitä, miten valo käyttäytyy vuorovaikutuksessa eri aineiden kanssa. (McDermott 2016b, 3.) PBR perustuu valo-oppiin ja tässä luvussa käsitellään teoriaa valon käyttäytymisestä.

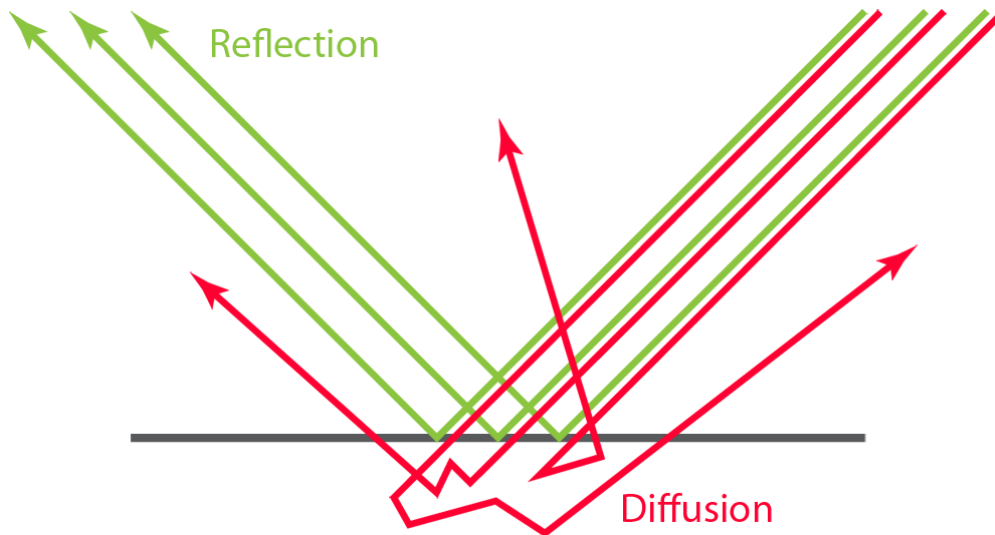
Valonsädemalli kuvaa valon ja aineen vuorovaikutusta keskenään. Valonsädemallin mukaan valonsäde etenee suoraviivaisesti, kun se liikkuu homogeenisessä aineessa, kuten ilmassa. Valonsäteen käyttäytyminen on mallin mukaan ennakoitavissa, kun säde törmää läpinäkymättömään esineeseen tai siirtyy aineesta toiseen, kuten esimerkiksi ilmasta veteen. Tämä mahdollistaa sen, että pystymme visualisoimaan säteen kulkeman reitin. (McDermott 2016a, 2.)

Valonsädettä joka törmää pintaan kutsutaan tulevaksi säteeksi ja sen törmäyskulmaa kutsutaan tulokulmaksi. Säteen törmätessä aineeseen, se joko heijastuu aineen pinnasta heijastuskulmassa, tai se siirtyy aineesta toiseen ja taittuu suoraviivaisesti, tai se heijastuu ja taittuu. Valon heijastuminen noudattaa heijastuslakia. Heijastuslain mukaan valonsäteen tulokulma on yhtä suuri kuin heijastuskulma. (McDermott 2016a, 2.)

Valon liikkeessa epähomogeenisessa tai läpikuultavassa aineessa, se joko absorboituu eli imeytyy, tai hajaantuu eli siroaa. Valon imeytyessä sen intensiteetti laskee, mutta sen suunta ei muutu. Absorboituessaan valo muuttuu eri energiamuotoon, useimmiten lämmöksi. Energiamuodon muuttuessa valon aaltopituuden väri muuttuu. Sironnassa säteen suunta vaihtuu satunnaisesti riippuen siitä missä materiaalissa se liikkuu. Sironna muuttaa valon suuntaa, mutta ei sen intensiteettiä. (McDermott 2016a, 3.)

Heijastuslain mukaan tasaiseen pintaan osuva valonsäde törmää pintaan tulokulmassa ja heijastuu pinnasta yhtä suuressa heijastuskulmassa. Useimmat pinnat eivät ole tasaisia, joten pinnasta lähtevän heijastuksen suunta riippuu täysin pinnan karkeudesta. Pinnan epätasaisuus muuttaa valon suuntaa, mutta ei sen intensiteettiä. Karkeilla pinnoilla on suuremmat ja himmeämmät heijastukset. Sileämmillä pinnoilla spekulariheijastukset näyttävät kirkkaammilta, kun niitä katsoo tietystä kulmasta. (McDermott 2016a, 3.)

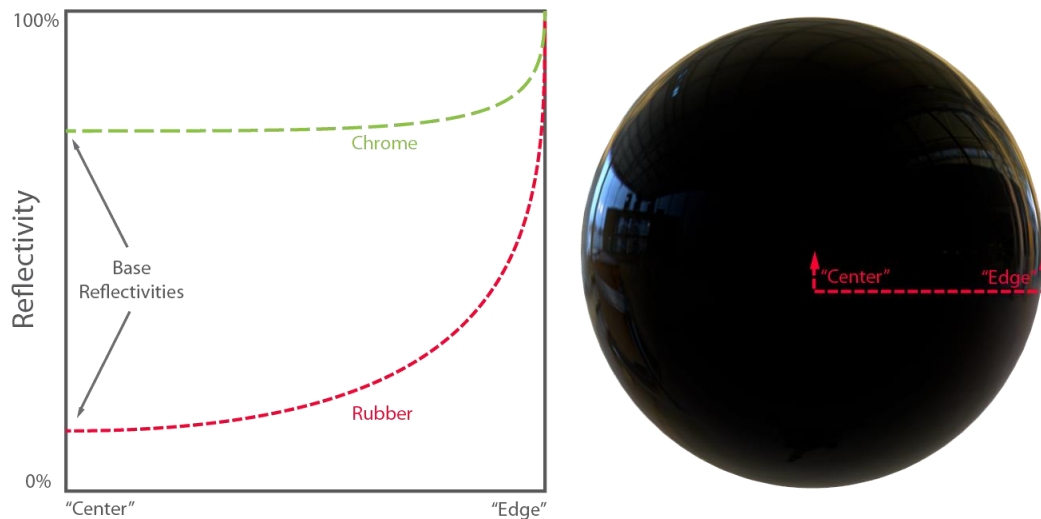
Diffuusioheijastus tarkoittaa taitettua valoa. Valonsäde siirtyy aineesta toiseen ja sirotaan moneen otteeseen, jonka jälkeen se taittuu uudestaan ja palaa takaisin alkuperäiseen aineeseen. Pinnan mikroskooppiset epäsäännöllisyydet aiheuttavat valon diffuusiota. Spekulariheijastus on suoraan pinnalta pelimäisesti heijastuvaa valoa. (McDermott 2016a, 3–5.)



KUVA 10. Diffuusio- ja spekulariheijastukset (Russell 2016b)

Objekti saa värinsä sen mukaan, mitä valonlähteen aallonpituuksia siihen imeytyy ja mitkä se heijastaa takaisin. Esimerkiksi punainen omena saa värinsä siitä, että omena diffuusioheijastaa valon punaista aallonpituutta ja muut värien aallonpituudet imeytyvät omena. (McDermott 2016a, 6.)

Kaksisuuntaisen heijastuskyvyn jakautumisen funktio eli Bidirectional Reflectance Distribution Function (BRDF) on yhtälö, joka kuvaa sitä miten paljon valoa heijastuu. Heijastetun valon määrään vaikuttaa katsojan ja valon sijainti suhteessa objektin pinnan sijaintiin. Objektin katsojan vaihtaessa paikkaa, vaihtuu myös valon heijastuksen paikka. (Wynn 2016.) Fresnel heijastuskerroin toimii BRDF:n kertoimena PBR:ssä (McDermott 2016a, 7).



KUVA 11. Fresnel heijastus (Russell 2016c)

Fresnel viittaa valon heijastumisen vaihtelun määrään, jota tapahtuu eri näkökulmista. Erityisesti sellaiseen valoon, joka osuu pintaan niin pienessä kulmassa, että se näyttää vain hipaisevan sitä. Hipaisevassa kulmassa osuva valo heijastuu pinnalta suuremmalla todennäköisyydellä, kuin valo joka osuu pintaan suoraan. Objektit joiden renderöinnissä on käytetty oikeanlaista Fresnel efektiä, näyttävät tästä syystä siltä, että heijastukset ovat kirkkaampia objektin reunoilla. (Russell, J 2016a.) Kuvassa 4 pallon reunoilla Fresnel-efekti erottuu hyvin.

Valon osuessa pintaan nollakulmassa, osa valosta heijastuu takaisin spekuläärisenä. Pinnan taitekertoimen avulla, voidaan selvittää heijastetun valon määrää. Tämä spekulääri-heijastettu valo on nimeltään F0 tai Fresnel-heijastus 0-asteen kulmassa. Yleisimmillä eristeillä F0-arvot ovat väliltä 0,02-0,05 ja johtimilla välillä 0,5-1,0. Puhtaat metallit ovat hyviä johtimia, mutta jos metalli on maalattua tai ruosteista, se luokitellaan eristeeksi. (McDermott 2016a, 8–9.)

Energian säilyvyyslaki on tärkeä osa PBR:ää. Säilyvyyslaki takaa sen, että heijastuvan valon intensiteetti ei ole koskaan suurempi, kuin mitä se oli ennen heijastumista. Varjostin hoitaa säilyvyyslain kontrolloinnin, joten graafikoiden ei tarvitse huolehtia fysiikasta. Perinteisimmillä varjostimilla Fresnel-efektiä piti itse kontrolloida, mutta PBR varjostin hoitaa myös Fresnel-efektin kontrolloinnin. (McDermott 2016a, 7.)

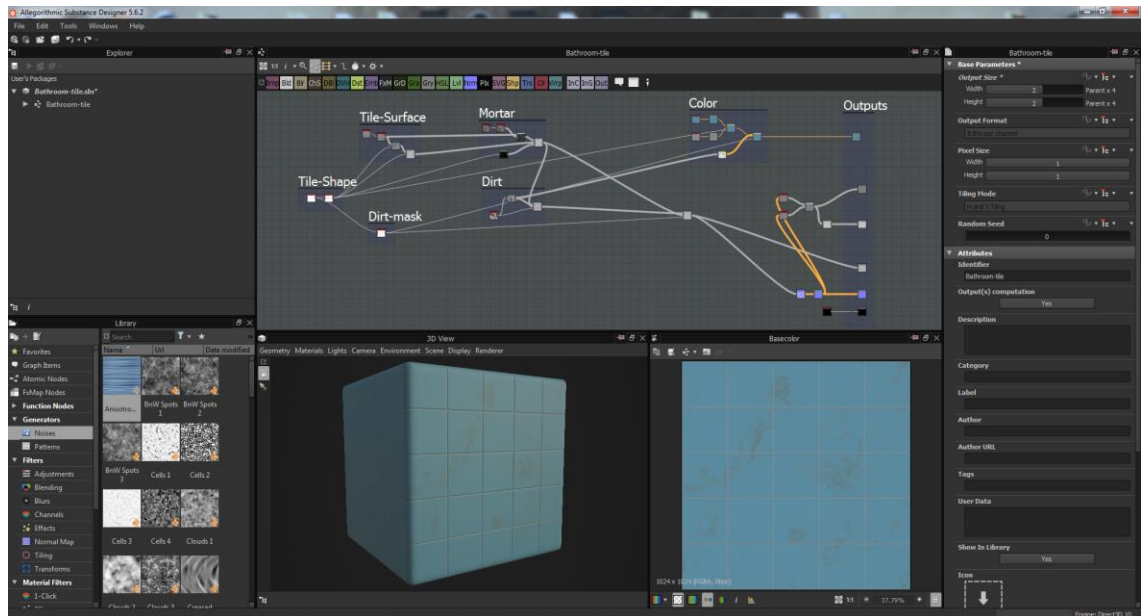


KUVA 12. Mikropinta kartoilla säädetään materiaalien heijastuksia (Russell 2015d)

Kuvasta 12 nähdään miten mikropintojen karkeuden säätäminen vaikuttaa materiaalin heijastuksiin. Reflectivity eli heijastuskyky on molemmilla kuvan materiaaleilla sama, mutta kiiltokarttaa säätämällä saadaan materiaalien heijastukset täysin erilaisiksi. PBR järjestelmässä kiiltokartalla ja karkeuskartalla säädetään mikropintojen karkeutta. (Russell 2015a.)

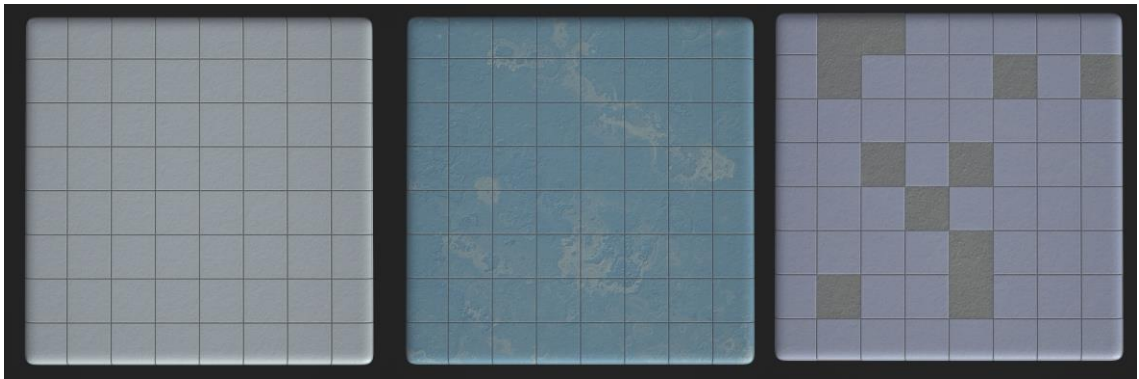
6 SUBSTANCE DESIGNER

Substance Designer on solmupohjainen teksturointityökalu. Solmut, englanniksi nodet, ovat yksinkertaisesti palikoita, jotka sisältävät dataa. Nodeja yhdistelemällä muodostetaan kuvaaja eli graafi, jossa data liikkuu hierarkian mukaisesti. Graafissa rakennetaan lopullinen tekstuuri liittämällä erilaisia nodeja toisiinsa. Nodejen sisältämää dataa säätämällä saadaan nopeita muutoksia lopulliseen materiaaliin. Graafi koostuu sisääntulo- ja ulostulo-nodeista. Sisääntulo-nodeilla rakennetaan tekstuurit ja ulostulo-nodet määrittävät sen, miltä lopullinen tekstuuri näyttää. Tekstuurikartat voi tuoda ulos Substance Designerista yksittäisinä 2D-tekstuureina tai Substance-materiaalina.



KUVA 13. Ruutukaappaus Substance Designerin käyttöliittymästä

Designerilla luodaan Substance-materiaaleja. Substancet ovat .sbsar-tiedostotyyppisiä materiaalipaketteja. Ne ovat dynaamisia, eli niitä voi muuttaa, esimerkiksi kokoa säätämällä. Materiaalin luoja voi myös paljastaa lisää parametreja näkyviin loppukäyttäjälle, jotta käyttäjä voi esimerkiksi muokata sitä, miten likainen materiaali on. Designerilla voi luoda joko täysin proseduraalisia tekstuureja tai käyttää hyödyksi bittikarttagrafiikkaa ja valokuvia. Substance-materiaalin etu on sen pieni koko. Täysin proseduraalinen materiaali voi olla kooltaan vain kymmeniä kilobittejä, kun taas perinteiset tekstuurit voivat olla kooltaan jopa 100-kertaisia. Substance-materiaaleja pystyy kaiken lisäksi animoida käymällä avoimiin parametreihin käsiksi koodin kautta. (Allegorithmic 2014.)



KUVA 14. Materiaalin parametrien säätelyllä luodut pikaiset muutokset

Vaikka Substance-työkaluja käytetään pääasiassa realististen PBR-materiaalien tekemiseen, pystyy niillä tekemään myös tyyliä grafiikkaa. Substance-tiedostojen pieni koko tekee Substance-materiaaleista hyödyllisiä myös mobiilipeihin. Mobiililaitteissa on usein aika vähän tallennustilaa, joten mobiilipeleissä on hyvin tärkeä, että pelin tiedostokoko pysyy mahdollisimman pienenä.



KUVA 15. Dragon Land -mobiilipelissä käytettiin Substance Designeria (Allegorithmic 2016d)

Substance Designer on osa ranskalaisen Allegorithmic-yhtiön Substance-työkalupakkia. Allegorithmic on kehittänyt Substance-työkalut, joihin kuuluu Substance Designerin lisäksi Substance Painter ja Substance Bitmap2Material -ohjelmistot. Substancen suosio kasvaa koko ajan, ja jopa yli 85 prosenttia isoista AAA-pelistudioista käyttää sitä. Sub-

stancesta on tullut pelialan uusi standardi PBR-materiaalien luomiseen seuraavan generaation peleihin. Substance-työkaluja on käytetty esimerkiksi Uncharted 4 ja Call of Duty Black Ops 3 -peleissä. Työkalujen käyttö on leviämässä myös muille aloille, kuten elokuvien visuaalisten tehosteiden luomiseen ja arkkitehtuurin 3D-visualisointiin. (Allegorithmic 2016a; Allegorithmic 2016b; Allegorithmic 2016c.)



KUVA 16. Black Ops 3 -pelissä käytettiin Substancea (Allegorithmic 2017a)

Substance Painter on 3D-maalausohjelma, jolla voi maalata suoraan 3D-mallin pinnalle. Painterissa käytetään Designerissa luotuja materiaaleja. Substance Painterin sisällä pystyy luomaan kaikki tarpeelliset tekstuurikartat suoraan 3D-mallista. (Allegorithmic 2016e.)

Substance Bitmap2Material tai lyhyemmin B2M on ohjelma, jolla pystyy generoimaan yhdestä valokuvasta kaikki tarvitsemasi PBR-tekstuurikartat. B2M:n avulla voi myös tehdä valokuvista saumattomia, jotta niitä pystyy käyttämään toistuvana tekstuurina. (Allegorithmic 2016f.)

Allegorithmic myy Substance-työkaluja omilla sivuillaan, joko paketissa tai yksittäin. Halvimmillaan ohjelmat saa käyttöön tilaamalla Substance Live Indie -pakettin. Substance Live on vuokrauspalvelu, jossa vuokraamalla maksat samalla tuotteen hintaa. Live maksaa 19,90 dollaria kuukaudessa ja ohjelmapaketin saa omakseen, kun kuukausimaksuista kertyy paketin kokonaishinta. Koko Substance-työkalupaketin Indie-lisenssi maksaa 299

dollaria. (Allegorithmic 2016g.) Työkalut voi ostaa myös Steam-palvelun kautta, josta Indie-lisenssin saa hankittua alennusmyyntien aikaan vähän halvemmalla.

Substance-työkalut ovat ilmaisia opiskelijoille ja opettajille (Allegorithmic 2016h). Tampereen ammattikorkeakoulua ei vielä kirjoitushetkellä löydy Allegorithmicin koululisältä, mutta Substance-työkalujen hankinnasta TAMKiin on puhuttu. Mikäli opiskelijalicenssiä ei vielä saa, niin Allegorithmicin sivuilta löytyy 30 päivän kokeiluversio kaikista työkaluista. Kokeiluaikajakson aikana ehtii hyvin oppia ohjelman perusteet katsomalla työn liitteeksi tehdyt videot ja Allegorithmicin omalta YouTube-kanavalta löytyvät aloittelijoille ja vähän kokeneemmille tarkoitetut videot.

Työkalujen lisäksi Allegorithmic myy Substance Source -materiaalikirjastoa, josta löytyy tällä hetkellä yli 800 materiaalia. Kirjasto on maksullinen, mutta sieltä löytyy myös joi-takin ilmaisia materiaaleja. Substance Sourcen ostaminen vuodeksi maksaa melkein 3000 dollaria. Halvimmillaan kirjastoon pääsee Substance Live -tilauksen ohella, sillä tilaajat saavat kuukausittain 30 Source -latausta. (Allegorithmic 2016i.)

Allegorithmicilta löytyy myös ilmainen Substance Share -palvelu, johon Substance -ohjelmien käyttäjät voivat jakaa tiedostoja ilmaiseksi muille ohjelmien käyttäjille. Sharesta löytyy muun muassa Designerilla luotuja materiaaleja, 3D-malleja sekä erilaisia pensseleitä käytettäväksi Substance Painterissa. (Allegorithmic 2016j.) Sharesta löytyy monenlaisia materiaaleja, joista voi ottaa esimerkkiä omien materiaalien luomiseen.

7 OPETUSVIDEOIDEN TOTEUTUS

Opetusmateriaalin olisi voinut toteuttaa videon sijasta tekstin ja kuvien avulla. Päätin tehdä opetusmateriaalin videomuodossa, koska Substance Designerin perusteisiin liittyy niin paljon asioita, että tekstimuotoisesta oppaasta tulisi todella pitkä. Halusin välttyä enemmältä kirjoittamiselta ja videoiden tekeminen oli minulle uusi mielenkiintoinen haaste.

Videoiden nauhoitukseen valitsin ruudunkaappausohjelma OBS Studion. Ohjelma on ilmainen ja siitä ei jää nauhoitettuun videoon vesileimaa, kuten monissa muissa samankaltaisissa ilmaisissa ohjelmissa. Ruudunkaappauksen yhteydessä voi nauhoittaa myös ääntä mikrofoniin avulla. OBS Studio on helppokäyttöinen ja internetistä löytyy ohjeita videoasetusten optimaaliseen säätöön.

Ennen videoiden nauhoittamista, täytyi suunnitella mitä videoissa käydään läpi. Suunnittelua varten katsoin esimerkkinä muiden tekemiä englanninkielisiä opetusvideoita, joista otin mallia omia videoitani varten. Monet internetin opetusvideot vaikuttavat täysin improvisoiduilta. Mielestäni kaikki parhaimmat opetusvideot seuraavat edellä määrättyä rakennetta ja opettaja käy asiat läpi luontevasti. Suunnittelin videoille rakenteen ja tein itselleni muistilistan läpikäytävistä asioista, jotta pysyisin paremmin asiassa videoita nauhoittaessa.

Päätin nauhoittaa kolme videota, koska olen käynyt läpi kymmeniä opetusvideoita aiheeseen liittyen ja uskoin, että saan Substance Designerin perusteet mahtumaan kolmeen videoon. Halusin pitää videot kohtuullisen lyhyinä. Olisin voinut tehdä vain yhden lähes kahden tunnin mittaisen videon, mutta pitkän videon nauhoitus ja editointi olisi ollut lyhyitä videoita vaativampaa.

Yritin pitää videot noin puolen tunnin mittaisina, jotta ne on helpompi sisäistää pieninä annoksina. Videoiden sisältö piti käydä läpi moneen kertaan, jotta sain nauhoitettua mielestäni tarpeeksi hyvää videota. Halusin myös välttää mahdollisimman paljon videoiden editointia, joten nauhoitus piti aloittaa moneen otteeseen alusta. Kaikista videoista tuli lopulta vähän yli puolen tunnin mittaisia. Videot ovat ehkä liian pitkiä, jos miettii optimaalista pituutta opetusvideolle.

Luodut opetusvideot ovat tarkoitettu aloittelijoille, joilla ei ole aiempaa kokemusta Substance Designer -ohjelman käytöstä. Ensimmäisessä videossa tutustutaan ohjelman käyttöliittymään ja perusteisiin rakentamalla yksinkertainen betoniseinä materiaali. Tällä tavoin opimme navigoimaan ohjelman sisällä ja miten ohjelmalla rakennetaan tekstuureja. Toisessa videossa rakennetaan kaakeli-materiaali, jolle asetetaan avoimet parametrit, joilla loppukäyttäjä voi säätää muun muassa kaakelien väriä. Kolmannessa videossa tutustutaan uusiin nodeihin, joita ei aiemmissa videoissa käytetty. Käyn läpi myös sen, miten Designerilla luodun materiaalin saa käyttöön Unity-pelimoottorissa ja miten videoiden jälkeen kannattaa jatkaa Substance Designerin harjoittelua.

Latasin videot YouTube-palveluun. Linkki videoihin löytyy opinnäytetyön liitteistä. YouTubeen käyttöön tarvitsee vain ilmaisen Google Gmail-tilin. Videoiden editoinnissa käytin YouTubeen omaa videonmuokkausohjelmaa, joka toimii suoraan selaimesta käsin. Videoiden lisääminen YouTubeen on todella helppoa. YouTubeen ladatut videot tulevat näkyviin omalle Creator Studio -sivulle YouTubeessa, jossa näkee tietoja omista videoista. Creator Studion kautta pääsee käyttämään YouTubeen omaa videonmuokkausohjelmaa. Videoiden muokkaaminen suoraan YouTubeessa on kätevää, sillä käyttäjän ei tarvitse asentaa erillistä ohjelmaa muokkausta varten. YouTubeen oma videonmuokkausohjelma on riittävä työssä tehtyjen videoiden muokkaamiseen.

8 POHDINTA

Tämän opinnäytetyön tuotoksena syntyi suomenkielinen kolmen opetusvideon sarja Substance Designer -teksturointityökalun perusteista. Ohjelman käytöstä löytyy pääasiassa vain englanninkielistä opetusmateriaalia. Englanninkielinen materiaali internetissä on riittävää ja sillä pääsee hyvin ohjelmaan sisälle. Ammattikorkeakoulussa kaikki osaavat englantia, mutta opiskelijat voivat olla hyvinkin eri tasoilla kielen ymmärtämisessä. Mielestäni suomenkielinen opetusmateriaali on hyvä lisä internetin muuhun materiaaliin ja luomani videot antavat hyvän pohjan, joka toivottavasti helpottaa siirtymistä englanninkielisiin opetusvideoihin.

Opetusvideoiden nauhoittaminen oli hankalaa. En ole aikaisemmin nauhoittanut tällaisia videoita ja videolle puhuminen oli suurempi haaste, kuin aluksi kuvittelin. Ennen videoiden nauhoitusta pidin itsestänselvyytenä, että osaan puhua videolle. Minun täytyi totutella videolle puhumiseen, ettei kieli mene solmuun kesken puheen. Lisävaikeuksia tuotti myös se, että yritin käyttää videoissa enimmäkseen suomenkielisiä termejä. Alalla käytetään pääasiassa englanninkielistä termistöä ja tuntuu oudolta puhua asioista suomeksi.

Sain käytyä videoissa mielestäni kaiken tarpeellisen perustiedon Substance Designerin käytön aloittamiseen. Videoiden katsomisen jälkeen voi siirtyä Allegorithmicin omiin englanninkielisiin opetusvideoihin. Vaikka kävin videoissani läpi ohjelman perusteet, koen silti aloittelijalle hyödylliseksi katsoa Allegorithmicin aloittelijoille tarkoitetut videot, sillä mitä enemmän ohjelmaan tutustuu sitä parempi. Allegorithmicin videoissa rakennetaan metallinen materiaali, tästä syystä rakensin omissa videoissani ei-metallisen materiaalin. Olisin voinut mahduttaa metallisen materiaalin rakentamisen omiin videoihini, mutta mielestäni pelkän ei-metallisen materiaalin rakentaminen on riittävä pohjustus ohjelman perusteisiin. Allegorithmicilta löytyy myös edistyneemmille käyttäjille tarkoitettuja videoita ja internetistä löytyy myös maksullisia ammattilaisten tekemiä opetusvideoita.

Kaikista videoista tuli vähän yli puolen tunnin mittaisia. Videot venyivät vähän liian pitkiksi, mutta YouTuben videosoittimessa voi onneksi säätää katsottavan videon nopeutta. Videon asetuksista nopeuden saa säädettyä normaalia jopa kaksi kertaa nopeammaksi. Asetuksista nopeus 1.25 nopeuttaa videota 25 prosentin verran normaaliin verrattuna. Tällä nopeudella videossa puheeni on vielä selkeää ja kaikki videot saa katsottua yhteensä

melkein puoli tuntia tavallista lyhyemmässä ajassa. Lisäsin videoihin aikaleimoja läpikäytävistä asioista, jotta videoita olisi helpompi seurata. Aikaleimat mahdollistavat myös sen, että käyttäjä löytää tärkeän kohdan uudelleen ilman ylimääräistä videon kelaamista.

Opinnäytetyön teoriaosuuteen olen sisällyttänyt tärkeitä Substance Designerin käyttöön liittyviä asioita. Teoriaosuus toimii pohjatietona videoiden katsomiseen. Teoriaosuutta ei ole pakko lukea ennen videoiden katsomista, mutta ohjelmaan voi päästä paremmin sisälle, jos teoriaosuus on luettu. Etenkin PBR teoriaa käsittelevä luku on tärkeää tietoa Substance Designerin käyttöön PBR-materiaalien tekemisessä. Teoriaosuudessa käyn asiat läpi yleisesti. Kaikissa luvuissa olisi voinut syventyä paremmin läpikäytävään aiheeseen ja rakentaa perusteellisempi opas aiheeseen. En ole koskaan ollut hyvä kirjoittamaan ja halusin keskittyä videoihin, joten tyydyin asioiden läpikäyntiin yleisellä tasolla.

Käytän elämässäni pääasiassa englantia kaikkeen tekemiseen puhumista lukuun ottamatta. Tarkoitukseni oli aluksi kirjoittaa opinnäytetyö englanniksi, sillä käytän melkein kaikessa englantia. Alalla käytetään englanninkielistä termistöä ja suomenkielisten termien käyttö sekoittaa asioita. Päätin myöhemmin aiheen muodostuessa, että haluan kirjoittaa työn suomeksi, vaikka se tuntuu itselle englantia vaikeammalta. Suomenkielinen teoriaosuus liittyy myös paremmin suomenkielisiin opetusvideoihin.

Substance Designer on todella tehokas teksturointityökalu ja suosittelisin kaikkia peilialalle haluavia graafikkoja opettelemaan ohjelman käyttöä. Designer on maksullinen ohjelma, mikä on iso kynnys opiskelijalle ohjelman hankkimiseen. Tästä syystä toivoisinkin, että TAMK saisi hankittua ohjelman lisenssit mahdollisimman pian.

Opetusvideoissa kävin läpi vain Substance Designerin käytön perusteita. Jatkokehitys aiheita löytyy paljon. Designer sisältää monia edistyneempiä toimintoja, joita en käynyt videoissa läpi millään tavalla. Hyvä esimerkki jatkokehitykseen aiheesta kiinnostuneelle voisi olla monimutkaisen materiaalin rakentaminen tai FX-map -nodeen ja funktio graafeihin syventyminen. Substance Designeria päivitetään koko ajan. Jos jokin tuleva päivitys muuttaa käyttöliittymää tai työskentelytapaa merkittävästi, voi tekemistäni opetusvideoista tarvita uudet versiot.

LÄHTEET

- Ahearn, L. 2008. 3D Game Environments: Create Professional 3D Game Worlds. Burlington, MA: Focal Press.
- Allegorithmic. 2014. Substance Designer #1 – Overview of a Substance. Katsottu 13.10.2016. <https://www.youtube.com/watch?v=OwLWishhLPM>
- Allegorithmic. 2016a. Luettu 13.10.2016. <https://www.allegorithmic.com/>
- Allegorithmic. 2016b. Substance in Games. Luettu 13.10.2016. <https://www.allegorithmic.com/games>
- Allegorithmic. 2016c. Substance. Luettu 13.10.2016. <https://www.allegorithmic.com/substance>
- Allegorithmic. 2016d. https://www.allegorithmic.com/sites/default/files/Luis-Hostos-Dragonland-Substance-User-story-1_1200.jpg
- Allegorithmic. 2016e. Substance Painter 3D Painting Software. Luettu 20.10.2016. <https://www.allegorithmic.com/products/substance-painter>
- Allegorithmic. 2016f. Substance B2M Image To-Material Generator. Luettu 20.10.2016. <https://www.allegorithmic.com/products/bitmap2material>
- Allegorithmic. 2016g. Luettu 12.01.2017. <https://www.allegorithmic.com/buy/indie>
- Allegorithmic. 2016h. Luettu 12.01.2017. <https://www.allegorithmic.com/buy/education>
- Allegorithmic. 2016i. Luettu 12.01.2017. <https://www.allegorithmic.com/buy/studio>
- Allegorithmic. 2016j. Luettu 12.01.2017. <https://share.allegorithmic.com/>
- Allegorithmic. 2017a. https://www.allegorithmic.com/sites/default/files/call_of_duty_black_ops_3_18%5B1%5D_1.jpg
- Allegorithmic. 2017b. Luettu 07.03.2017. <https://support.allegorithmic.com/documentation/display/SD5/Thickness+Map+from+Mesh>
- Allegorithmic. 2017c. https://support.allegorithmic.com/documentation/download/attachments/102400039/alien_hunter_thickness_from_mesh.jpg?version=1&modificationDate=1423476116000&api=v2
- Chopine, A. 2011. 3D Art Essentials. The Fundamentals of 3D Modeling, Texturing, and Animation. Burlington, MA: Focal Press.
- Ebert, D. S., Musgrave F. K., Peachey, D., Perlin, K., Worley, S. 2003. Texturing & Modeling, A Procedural Approach, Third Edition. San Francisco: Morgan Kaufmann Publishers.

- Epic. 2017. Luettu 07.03.2017. https://docs.unrealengine.com/latest/INT/Engine/Rending/Materials/HowTo/Subsurface_Scattering/index.html
- Henriksen, M. 2015. <https://www.artstation.com/artwork/180G8>
- Kessenich, J., Sellers, G., Shreiner, D. 2016. OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V, Ninth Edition. Addison-Wesley Professional.
- Lagarde, S. 2011. Feeding a physically based shading model. Luettu 20.02.2017. <https://seblagarde.wordpress.com/2011/08/17/feeding-a-physical-based-lighting-mode/>
- McCombs, S. 2017. Intro to Procedural Textures. http://www.upvector.com/pages/Tutorials/Intro%20to%20Procedural%20Textures/image05/procedural_zoom.gif
- McDermott, W. 2016a. The Comprehensive PBR Guide by Allegorithmic - vol 1. Luettu 15.10.2016. PDF-dokumentti. https://www.allegorithmic.com/system/files/software/download/build/PBR_Guide_Vol.1.pdf
- McDermott, W. 2016b. The Comprehensive PBR Guide by Allegorithmic - vol 2. Luettu 15.10.2016. https://www.allegorithmic.com/system/files/software/download/build/PBR_volume_02_rev05.pdf
- Munoz, P. 2015a. 3D Lingo. Luettu 10.02.2017. <http://www.zbrushguides.com/3d-lingo/>
- Munoz, P. 2015b. <http://www.zbrushguides.com/wp-content/uploads/2015/05/Poly-mesh.jpg>
- Nash, S. 2009. <http://sheilanash.com/images/telexample011.jpg>
- Pluralsight. 2014. Eliminate Texture Confusion: Bump, Normal and Displacement Maps. Luettu 25.01.2017. <https://www.pluralsight.com/blog/film-games/bump-normal-and-displacement-maps>
- Polycount 2015. Transparency map. Luettu 02.03.2017. http://wiki.polycount.com/wiki/Transparency_map
- Polycount 2016. Texturing. Luettu 03.03.2017. <http://wiki.polycount.com/wiki/Texturing>
- Quixel. 2015. <http://quixel.se/hub/wp-content/uploads/2015/11/Texel-Density.jpg>
- Russell, J. 2015a. Basic theory of physically-based rendering. Luettu 15.10.2016. <https://www.marmoset.co/posts/basic-theory-of-physically-based-rendering/>
- Russell, J. 2015b. https://www.marmoset.co/wp-content/uploads/2016/11/pbr_theory_fresnel.png
- Russell, J. 2015c. https://www.marmoset.co/wp-content/uploads/2016/11/pbr_theory_refl_diff.png

Russell, J. 2015d. https://www.marmoset.co/wp-content/uploads/2014/01/pbr_theory_watermud2.png

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

Unity 2016a. Height Map. Luettu 20.01.2017. <https://docs.unity3d.com/Manual/StandardShaderMaterialParameterHeightMap.html>

Unity. 2016b. Occlusion Map. Luettu 20.01.2017. <https://docs.unity3d.com/Manual/StandardShaderMaterialParameterOcclusionMap.html>

Vries, J. 2017. https://learnopengl.com/img/advanced-lighting/normal_mapping_normal_map.png

Wilson, J. 2015a. <https://www.marmoset.co/wp-content/uploads/2014/01/metalness02.jpg>

Wilson, J. 2015b. <https://www.marmoset.co/wp-content/uploads/2014/01/ao01.jpg>

Wynn, C. 2016. An Introduction to BRDF-Based Lighting. Luettu 16.10.2016. <http://www.cs.princeton.edu/courses/archive/fall06/cos526/tmp/wynn.pdf>

LIITTEET

Liite 1. Linkki opetusvideoiden soittolistaan YouTube-palvelussa.

<https://www.youtube.com/playlist?list=PL7RXLGEyddUsDqJZ290oq-nq1n7b4FGhq>