



Osaamista  
ja oivallusta  
tulevaisuuden  
tekemiseen

Sami Björklund

# Trim-tekstuurit peliympäristöissä

Metropolia Ammattikorkeakoulu

Medianomi (AMK)

Viestintä

Opinnäytetyö

28.5.2020

Tekijä(t) Otsikko	Sami Björklund Trim-tekstuurit peliympäristöissä
Sivumäärä Aika	59 sivua 28.5.2020
Tutkinto	Medianomi (AMK)
Tutkinto-ohjelma	Viestintä
Suuntautumisvaihtoehto	3D-animointi ja -visualisointi
Ohjaaja(t)	Lehtori Jaro Lehtonen
<p>Opinnäytetyössä selvitettiin, kuinka trim-tekstuureilla pystytään optimoimaan peliympäristöjen teksturoimiseen käytettyjä resursseja. Opinnäytetyön toisena tarkoituksena on kerätä yhteen hajanaiset tiedot trim-tekstuureista ja antaa lukijalle kattava käsitys tekstuurin erilaista käyttömahdollisuuksista. Työssä trim-tekstuureja lähestytään sillä ajatuksella, että teksturoidut 3D-mallit viedään lopulta pelimoottoriin.</p> <p>Alkuosa opinnäytetyöstä keskittyy käsittelemään modulaarisuuta kuvien avulla trim-tekstuurien käyttöympäristön havainnollistamiseksi ja erilaisten trim-tekstuureiden rinnalla käytettävien tekniikoiden selkeyttämiseksi, jonka jälkeen esitellään kolme yleisintä peliympäristöjen teksturoimiseen käytettyä tekstuuria. Tämän jälkeen keskitytään trim-tekstuurien hyötyihin erilaisten esimerkkien kautta ja esitellään kaavioiden avulla erot normaalin 3D-mallin työnkulun ja trim-tekstuurityönkulun välillä, jonka jälkeen työssä käydään läpi trim-tekstuuritekniikkaan liittyviä tilanteita monipuolisten kuvaesimerkkien avulla. Aivan lopuksi työ keskittyy antamaan lukijalle hyödyllisiä neuvoja ja linkkejä trim-tekstuurien suunnitteluun.</p> <p>Opinnäytetyössä käsitellyiden aiheiden lukemisesta on hyötyä kaikille 3D-mallinnuksen parissa työskenteleville henkilöille, joille trim-tekstuurit eivät ole vielä ennestään tuttuja, mutta suurimman hyödyn työstä saavat peliympäristöjen luomisesta kiinnostuneet 3D-graafikot. Trim-tekstuurit ovat tietyissä tilanteissa erittäin tehokkaita työkaluja, jonka takia ne ovat pelialalla laajassa käytössä. Trim-tekstuurit antavat tekijälle vaihtoehtoisen tavan lähestyä 3D-mallintamiseen liittyviä ongelmia ja ne tulevat olemaan tulevaisuudessakin käytössä, kunnes keksitään uusi tapa teksturoida 3D-malleja ilman normal mappeja.</p>	
Avainsanat	3D, peli, teksturointi, optimointi, trim-teksturi, trim sheet

Author(s) Title	Sami Björklund Trim textures in game environments
Number of Pages Date	59 pages 28 May 2020
Degree	Bachelor of Culture and Arts
Degree Programme	Media
Specialisation option	3D Animation and Visualization
Instructor(s)	Jaro Lehtonen, Senior Lecturer
<p>The aim of this thesis is to study how trim textures can be used to optimize the available texture resources in game environments. As a secondary goal, the thesis pools scattered information about trim textures in one place and gives the reader a comprehensive overview of their affordance. In this thesis trim textures are approached with the idea, that the fully textured 3D models will eventually be implemented into a game engine.</p> <p>The beginning of this thesis will address modularity to demonstrate the standard use environment of trim textures, and to clarify certain techniques that are used in game environments side by side with the textures. Subsequently, the reader will be presented with the three most used texture types in game environments. After this, the thesis will focus on the advantages of trim textures, and on the differences between the normal 3D workflow and the trim texture workflow, after which the trim texture technique will be analyzed through diverse image examples. Lastly, the thesis will concentrate on giving the reader useful tips and links for designing their own trim textures.</p> <p>Reading about the subjects discussed in this thesis will be useful to everyone who works with 3D modeling and is not yet familiar with the trim texturing technique. It will be most valuable to 3D graphic artists who are interested in creating game environments. In certain situations, trim textures are extremely powerful texturing tools, for which they are in wide use in the games industry. Trim textures give the artist an alternative way to approach problems that arise when 3D objects are modeled and textured, and they will be in use until a new way to texture objects without a normal map is presented.</p>	
Keywords	3D, game, texturing, optimization, trim texture, trim sheet

## Sisällys

1	Johdanto	1
2	Kohderyhmä ja keskeisimmät käsitteet	3
2.1	Kohderyhmänä ympäristöjen luomisesta kiinnostuneet 3D-graafikot	3
2.2	Keskeisimmät käsitteet	3
3	Trim-tekstuurien käytöstä	4
3.1	Modulaarisuus	5
3.2	Modulaarisuuden rikkominen katsojaa huijaamalla	6
3.3	Modulaarisuuden rikkominen tasosuunnittelulla	10
3.4	Hero assettien luominen modulaarisista paloista	12
3.5	Texel density	13
3.6	Light map	15
3.7	Peliympäristön kolme tekstuuria	16
3.8	Kolme tapaa käyttää trim-tekstuuria	18
4	Trim-tekstuurin hyödyt	21
4.1	Huijari	27
4.2	Säästäjä	29
5	Käytännön sovellutukset ja esimerkit	32
6	Trim-tekstuurin suunnittelu	46
6.1	Lähestymistavat	46
6.2	UV-setit	49
6.3	Teksturoinnista	49
7	Yhteenveto ja pohdinta	51
	Lähteet	53

## 1 Johdanto

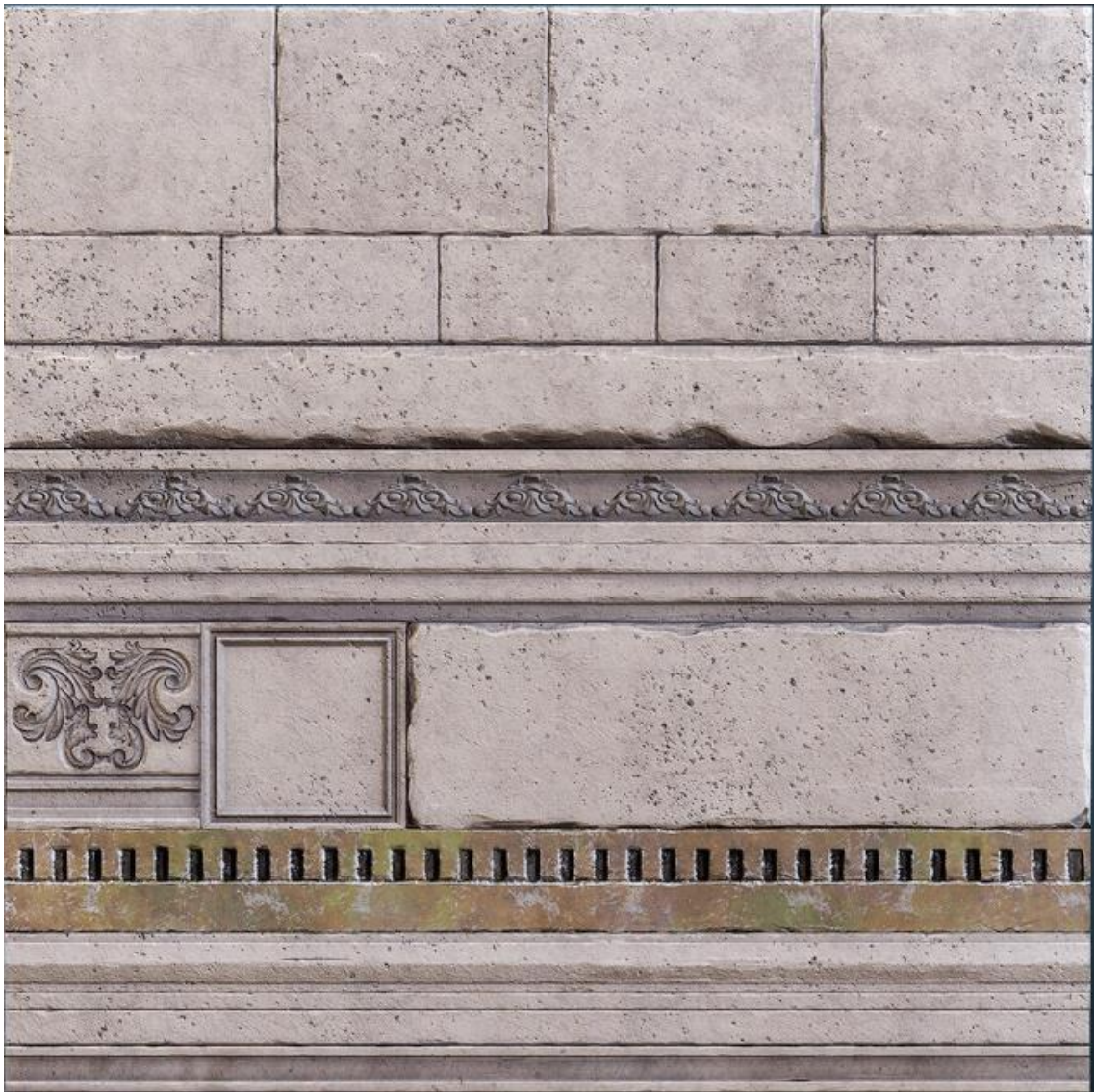
Tämän opinnäytetyön tavoitteena on esitellä lukijalle trim-tekstuuritekniikka, joka on erittäin käytännöllinen erilaisien ympäristöjen ja proppien teksturoimiseen. Trim-tekstureilla teksturointi poikkeaa yllättävillä tavoilla normaalista 3D-mallin teksturoinnista ja tarjoaa toisen tavan lähestyä teksturointiin ja 3D-mallintamiseen liittyviä ongelmia. Trim-tekstuurista käytetään myös nimitystä trim sheet tai trim material. Pelialan ammattilaiset ovat hyödyntäneet trim-tekstureja jo vuosia, mutta valitettavasti tietoa niiden käytöstä on tihkunut hyvin niukasti pelitalojen ulkopuolelle. Nyt oppi trim-tekstureista on kuitenkin alkanut vuotamaan erilaisille foorumeille ja videopalveluihin, minkä takia opinnäytetyöni toisena tavoitteena on kerätä hajanaiset tiedonjyvät yhtenäiseksi paketiksi ja innostaa lukijaa lähtemään tutkimaan trim-tekstuurien mahdollisuuksia omassa 3D-työnkulussa.

Tulen käymään tekniikan läpi useilla esimerkkikuvilla, mutta jätän tietoisesti scifi-aiheiset esimerkit vähemmälle, koska niitä on jo löydettävissä runsaasti. Trim-tekstuuritekniikka soveltuu erityisesti sciifiin, koska trim-tekstureilla voidaan helposti luoda erilaisia paneeleita, jotka mahdollistavat vaihtelevien seinä- ja lattiapintojen rakentamisen. Haluan kuitenkin antaa lukijalle ymmärryksen siitä, että tekniikka ei rajaudu pelkästään sciifiin, koska se soveltuu tehokkaasti kaikkiin ympäristöihin ja genreihin. Trim-tekstuuritekniikkaa voi käyttää vapaasti kaikessa 3D-mallinnuksessa, mutta se paljastaa suurimman potentiaalinsa peliasettien kanssa, koska yhdellä tekstuurilla voidaan teksturoida useita erilaisia 3D-malleja.

Oletuksena opinnäytetyössä on lukijan ymmärrys normaalista 3D-mallinnuksen työnkulusta ja valmiin 3D-mallin viemisestä pelimoottoriin, koska trim-tekstuurit mahdollistavat joustavamman, mutta aluksi vaikeaselkoisemman tavan lähestyä 3D-mallinnuksen eri työvaiheita. Käyn luvussa 2 läpi kohderyhmän ja keskeisimmät käsitteet. Luvussa 3 kerron aluksi lyhyesti peliympäristöjen modulaarisuudesta ja siitä, kuinka modulaarisuutta voidaan visuaalisesti rikkoa. Aihe on tärkeä, koska trim-tekstuurit ovat osa isompaa kokonaisuutta, jonka haluan lukijalle selventää. Modulaarisuusosion jälkeen kerron vielä muutamista trim-tekstuurin teknisistä rajoitteista ja lopuksi trim-tekstuurien erilaisista käyttötavoista.

Lukijan ei tarvitse pelästyä, jos osio trim-tekstureiden käytöstä ei heti aukea, koska lähdän avaamaan käsiteltyjä aiheita tarkemmin seuraavissa luvuissa. On myös normaalia, että lukija joutuu aluksi harjoittamaan mielen akrobatiaa tekniikan kanssa, koska trim-

tekstuureja lähestytään kulmasta, joka ei välttämättä ole lukijalle ennestään tuttu. Luvussa 4 käyn läpi trim-tekstuurin erilaisia hyötyjä ja tutustutan lukijalle, kuinka trim-tekstuurityönkulku poikkeaa normaalista 3D-mallin työnkulusta. Luvussa 5 näytän lukijalle erilaisia käytännön sovellutuksia ja esimerkkejä, kuinka trim-tekstuureja voidaan projekteissa hyödyntää. Luvussa 6 käyn vielä läpi trim-tekstuurin suunnitteluun yleisesti liittyviä asioita, mutta en anna lukijalle varsinaista step by step -ohjetta, koska tekstuurin voi tehdä niin monella eri tavalla. Luvun tarkoitus on antaa lukijalle eri työvaiheisiin liittyviä huomioita ja ideoita trim-tekstuurien mahdollisuuksista. Lopuksi luvussa 7 loppuyhteenveto ja pohdinta.



Kuvio 1. Kuva trim-tekstuurista. Trim-tekstuurilla pystytään optimoimaan pelissä näkyvien 3D-mallien pintojen tekstuureja tavalla, missä 3D-mallinnus, UV-työskentely ja tekstuurien maalaaminen yhdistyvät joustavasti yhteen. Yhdellä trim-tekstuurilla voidaan teksturoida useita 3D-malleja, minkä lisäksi saman layoutin mukaan tehtyjä trim-tekstuureja pystytään jopa vaihtamaan lennosta keskenään (Deschamps 2019).

## 2 Kohderyhmä ja keskeisimmät käsitteet

### 2.1 Kohderymänä ympäristöjen luomisesta kiinnostuneet 3D-graafikot

Tämä työ on suunnattu varsinkin peliympäristöjen luomisesta kiinnostuneille 3D-graafikoille, koska isoilla pelistudioilla ympäri maailmaa alkaa olemaan sanomaton oletus siitä, että työnhakija ymmärtää, kuinka trim-tekstuureja käytetään (Simpson 2019). Trim-teksturointitekniikan osaamista voisi suositella kaikille oman 3D-mallinnuksen optimoimisesta kiinnostuneille henkilöille, koska tietyissä tilanteissa tekniikka nopeuttaa mallinnustyöskentelyä tuntuvasti ja antaa vaihtoehdoisen tavan työskennellä 3D-mallinnuksen parissa. Lukijan täytyy hallita normaali 3D-mallin työnkulku voidakseen hyödyntää tekniikkaa.

### 2.2 Keskeisimmät käsitteet

**Assetti** – 3D-mallinnusprosessin ja teksturoinnin läpikäynyt valmis 3D-malli.

**3D-malli** – 3D-mallinnusohjelmassa työstettävänä oleva 3D-malli.

**Trim-tekstuuri / trim sheet / trim material** – On tekstuuri, joka on yleensä jaettu erikoisiin trim-liuskoihin. Liuskat muodostavat neliön, joka täyttää koko tekstuurikartan.

**Trim-liuska** – Trim-tekstuurit koostuvat yleensä vierekkäisistä tai päällekkäisistä trim-liuskoista, jotka voidaan mallintaa trim-tekstuuriin vertikaalisesti tai horisontaalisesti. Trim-liuskojen reunat bevelöidään, koska tieto bevelistä saadaan siirrettyä normal mappiin, jonka avulla 3D-mallin lowpoly-geometria saadaan näyttämään yksityiskohteisemmalta.

**Bevel** – On sana, joka on hyvin yleisessä käytössä 3D-piireissä. Sanalla tarkoitetaan toimenpidettä missä lowpoly-mallin terävät reunat pyöristetään, jonka seurauksena katsoja saa enemmän tietoa 3D-mallin pintojen muutoksista. Bevelöinti kasvattaa 3D-mallin polygonmäärää.

**Polygon** – 3D-mallin perusosa, joka koostuu kolmesta tai neljästä verteksistä.

**Verteksi** – Piste kolmiulotteisessa tilassa, jolla on sijainti mutta ei kokoa.

**Edge** – Viiva, joka yhdistää kaksi verteksiä kolmiulotteisessa tilassa.

**Edge loop** – Useasta edgestä muodostuva rengas, mikä kiertää koko 3D-mallin ympäri.

**Polygon loop** – Useasta polygonista muodostuva rengas, mikä kiertää koko 3D-mallin ympäri.

**Lowpoly** – 3D-malli, jossa on mahdollisimman vähän polygoneja.

**Highpoly** – 3D-malli, jossa on suuri määrä polygoneja.

**Beikkaus** – Käsite tulee englanninkielisestä sanasta baking ja sillä tarkoitetaan tiedon esilaskemista ja tallentamista staattisempaan muotoon, mikä säästää laskentaresursseja. Tässä opinnäytetyössä beikkaamisella tarkoitetaan highpoly-informaation siirtämistä normal mappiin.

**Normal map** – On RGB-kuva, johon tallennetaan jokaisen pikselin suunta. Näitä suuntia kutsutaan normaaleiksi. Kuvan Red, Green ja Blue-kanavilla kontrolloidaan arvojen muuntamista X-, Y-, Z-arvoiksi, joita käytetään simuloimaan pintojen yksityiskohtia. Normal mapilla vaikutetaan siihen miltä 3D-geometria näyttää valon osuessa siihen.

### 3 Trim-tekstuurien käytöstä

Mielestäni trim-tekstuureja voisi ajatella modulaarisina tekstuureina, koska tekstuurin voi jakaa paloiksi, joita monistamalla ja yhdistelemällä voidaan luoda täysin uusia kokonaisuksia. Trim-tekstuurit ovat omalla tavallaan yhdistelmä UV-atlasointia ja toistuvia saumattomia tekstuureja. Edellä mainittuja tekniikoita on käytetty 3D-mallinnuksessa jo vuosikymmeniä peliprojektien optimoimiseen, koska niiden tavoitteena on käyttää hyväkseen yhtä tekstuurikarttaa niin paljon kuin mahdollista.

Tekniikat auttavat supistamaan draw calleja ja uniikkien tekstuurien määriä, mikä johtaa optimaalisempaan lopputulokseen niin mallintajan ajankäytön niin-kuin pelin suorituskyvynkin kannalta. Draw call tarkoittaa vaihetta, jossa pelimoottori kutsuu objektin ja sen tekstuurit piirrettäväksi yksittäisen kuvaruudun aikana (Palmi 2013, 4.). Trim-tekstuureilla



on omat rajoitteensa, minkä takia 3D-mallien teksturointiin ei kannata yrittää käyttää pelkästään trim-tekstuureja. Tekniikalla pystytään teksturoimaan suuria alueita ja tietynlaiset 3D-mallit jopa kokonaan, mutta ongelmia ilmaantuu, jos tekniikalla pyritään teksturoimaan kaikki näkyvät pinnat, koska pakottaminen johtaa tilanteisiin, joissa visuaalinen lopputulos kärsii. Projekteissa trim-tekstuureja käytettäessä pitää tasapainoilla erilaisten teksturointitapojen ja projektin teknisten rajoitteiden kanssa. (Dyger 2019)

Seuraavaksi haluan kertoa lukijalle modulaarisuudesta, koska trim-tekstuurit soveltuvat erityisesti modulaaristen palojen teksturoimiseen, koska trim-teksturi joudutaan pitämään hyvin geneerisenä suuren toistuvuutensa takia. Modulaarisuudesta aiheutuu visuaalisia ongelmia, jotka lukijan on hyvä tiedostaa, koska kyseisiin ongelmiin on jo olemassa erilaisia ratkaisuja.

### 3.1 Modulaarisuus



Kuvio 2. Esimerkki modulaarisesta setistä, josta voidaan muodostaa useita erilaisia rakennuksia (Deschamps 2019).

Modulaarisuutta voi mielestäni helpoiten kuvata legopalikoiden kautta. Käytännössä tämä tarkoittaa sitä, että yksittäiset palaset yhdistyvät saumattomasti keskenään ja luovat yhdessä monimutkaisen kokonaisuuden (Kuvio 2). Kiinnostavat ja erilaiset lopputulokset saavutetaan, vaikka käytössä on vain kourallinen uniikkeja palikoita. Peliympäristöissä tämä johtaa kuitenkin ongelmiin, koska pelaaja voi helposti menettää mielenkiinnon tutkia ympäröivää pelimaailmaa, jos samat palat ja tekstuurit toistuvat liikaa. Tätä ilmiötä Bethesda Game Studios kutsuu art fatigueksi (Burgess 2019).

Art fatiguen seurauksena pelaaja menettää motivaation tutkia uusia löytämiään luolia, koska hän tietää, että uudet luolat tulevat näyttämään täsmälleen samalta kuin jo aikaisemmin löydetyt luolat. Pelaajaa voidaan kuitenkin huijata erilaisilla tavoilla ja saada toistuvat pinnat näyttämään monipuolisemmilta kuin ne todellisuudessa ovat.

### 3.2 Modulaarisuuden rikkominen katsojaa huijaamalla

Valolla ja varjolla voidaan saavuttaa paljon. Pelaajan käsitys tekstuurin toistuvuudesta häiriintyy, kun itseään toistavalle pinnalle lankeaa erilaisia valo- ja varjoalueita ympäristön objekteista (Kuvio 3).



Kuvio 3. Pelaaja ei kiinnitä huomiota suurien pintojen ja elementtien toistuvuuteen, kun pinnat ovat osittain valossa tai varjossa. Huomaa myös kuinka toistuva kaide ja portti näyttävät visuaalisesti kiinnostavammilta ollessaan osittain varjossa (Irrational Games 2013).

Valolla huijaamiseen liittyy myös värien ja valon kimpoaminen, koska toistuvat pinnat näyttävät elävämmiltä, kun valoa kimpoaa varjossa oleviin asetteihin. Myös erilaiset suunnitellut pintojen heijastuksien vaihtelut saavat toistuvan materiaalin näyttämään visuaalisesti kiinnostavammalta. (Bernstein 2017, 48.)

Bethesdalla on myös huomattu, että pelaaja ei kiinnitä ensimmäisenä huomiota ympäristössä suurien elementtien toistuvuuteen, mikä antaa mahdollisuuksia erilaisten seinien ja lattioiden toistamiseen, mutta pienten esineiden toistuvuuteen pelaaja reagoi kuitenkin hyvin herkästi. Pelaajan huomio kiinnittyy nopeasti pöytään, jonka päällä on aina samat esineet ja samassa järjestyksessä. Tämä huomio on hyvä pitää mielessä ympäristöjä rakentaessa. (Burgess 2013)

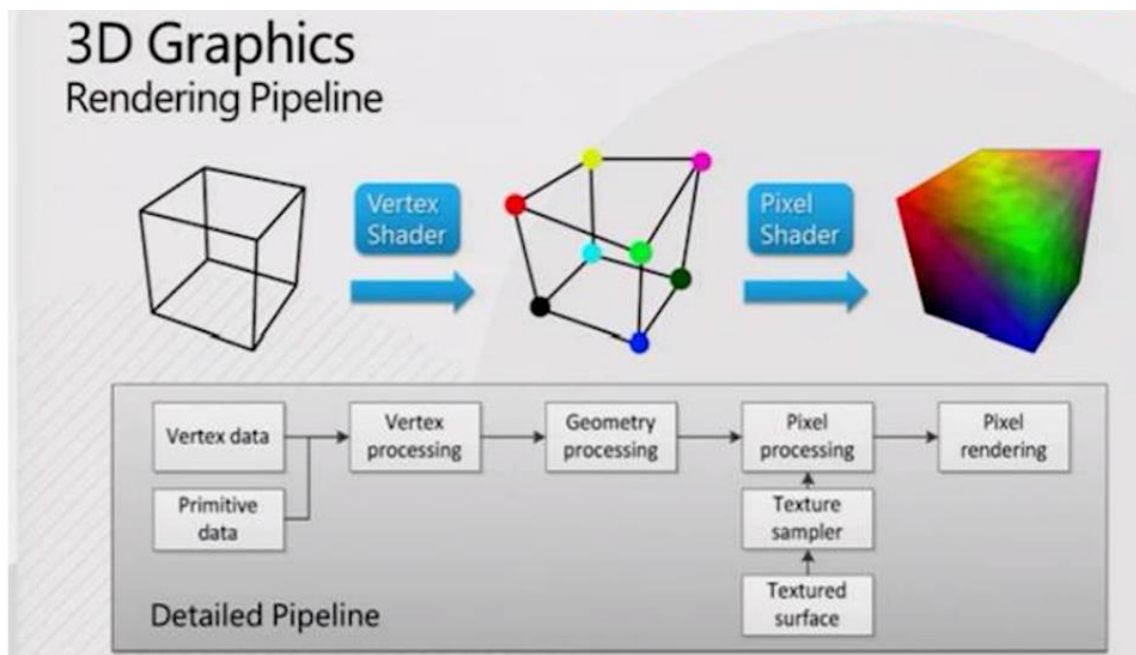


Kuvio 4. Kuvassa erilaisista ruostevaurioista koostuva decal-tekstuuri (TextureMax 2015).

Peleissä katsojaa huijataan myös decaleilla. Decalit ovat tekstuureja, joilla on läpinäkyvyyden mahdollistama alpha-kanava. Decaleita voi ajatella julisteina, jotka on teipattu kiinni seinään, koska pelimoottorin puolella decalit leijuvat hieman pintojen yläpuolella eivätkä ne ole oikeasti kiinni missään pinnassa tai tekstuurissa. Decal-tekstuurien läpinäkyvyys on itsessään raskasta, mutta koska yhdessä decal-tekstuurimapissa voi olla monta erilaista roiskekuviota, jotka voidaan rajata omiksi decaleiksi (Kuvio 4) ja kun yhtälöön lisätään pelimoottorin puolella mahdollisuus säätää decalien näkyvyyttä päälle ja pois, ovat decalit lopulta suhteellinen optimaalinen tapa luoda variaatioita toistuviin tekstuureihin. (Bernstein 2017, 49–51.)

Geneerinen toistuva saumaton tekstuuri on tavallaan muuttumattomana omalla layerillä ja decalit tämän layerin yläpuolella. Tämä mahdollistaa visuaalisesti erottuvien decalien lisäämisen toistuvan tekstuurin päälle, koska tekstuurit eivät ole samassa tekstuurissa, vaikka lopputulos siltä näyttääkin.

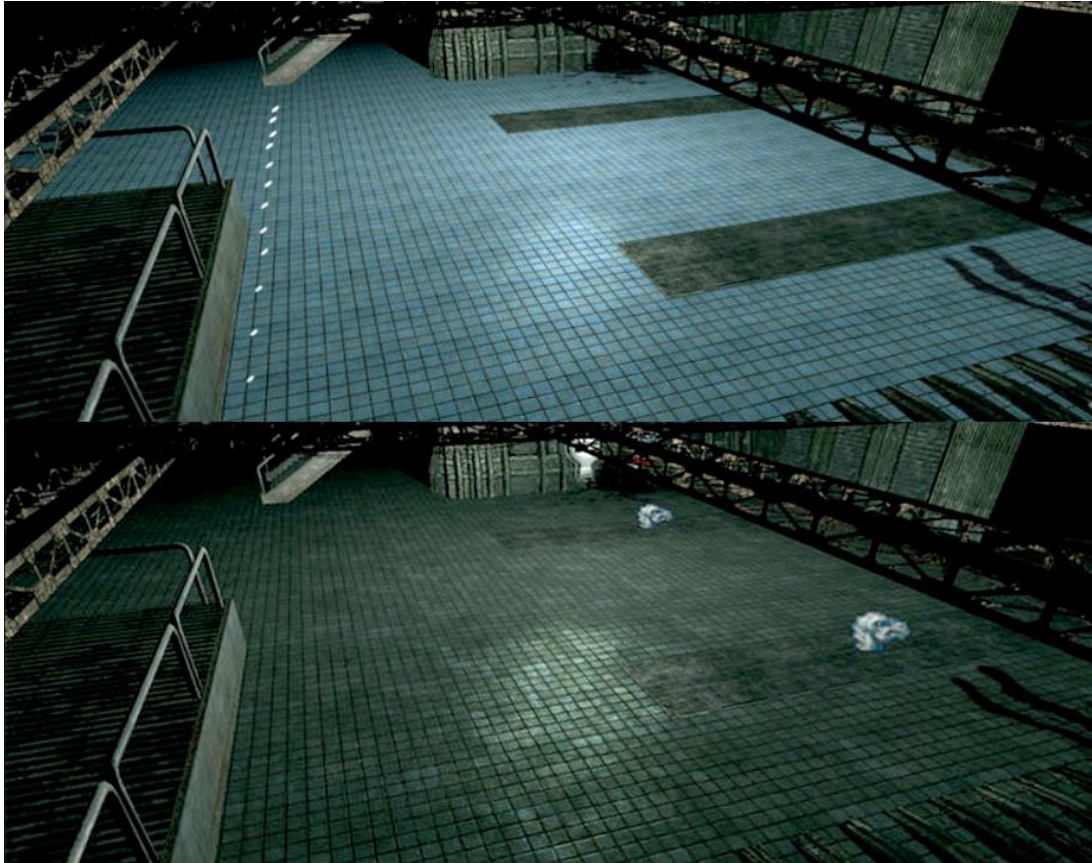
Pelimoottoreissa pystytään luomaan myös shadereita, jotka mahdollistavat erilaisten efektien lisäämisen objekteihin ja tekstuureihin ennen kuin ne piirretään kuvaruudulle. Vertex shadereita käytetään muuttamaan verteksin paikkaa, väriä ja tekstuurikoordinaatteja. Vertex shaderit voivat tehdä muutoksia jo olemassa oleviin vertekseihin, mutta ne eivät voi luoda uusia verteksejä. Pixel shadereita käytetään laskemaan yksittäisten pikselien efektejä. Yleisesti pikselin väriä, mutta laskenta ottaa myös huomioon valon, varjon, läpinäkyvyyden ja sumun. Esimerkiksi normal mapit ovat yksi suosituimmista tavoista hyödyntää pixel shaderia. Normal mapilla vaikutetaan siihen miltä 3D-geometria näyttää valon osuessa siihen. (Game Dev Academy 2015)



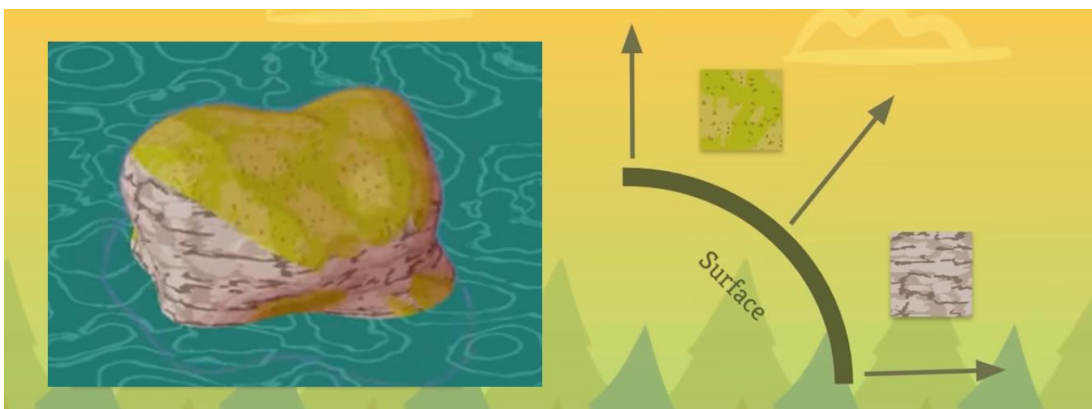
Kuvio 5. Shaderit manipuloivat kuvan lopputulosta ennen kuin se piirretään kuvaruudulle (Game Dev Academy 2015).

Yksinkertaistettuna shader on pala koodia, joka ajetaan tietokoneen grafiikkaprosessorissa. Pelimoottori lähettää tietoa 3D-mallin vertekseistä ja tekstuureista grafiikkaprosessoriin, josta tulee takaisin pikseleiden oikea väri, joka piirretään kuvaruudulle. (Shadecat 2016) Pelimoottoreissa shadereita voidaan käyttää yhdistämään erilaisia tekstuureja ja maskeja. Yleisimpiä käyttötapoja ovat esimerkiksi vertex painting, jossa saumattomia tekstuureja voidaan maalata ja sekoittaa keskenään ja erilaiset multiply shaderit, joilla

voidaan luoda värin ja tummuuden vaihteluita tekstuuriin päälle (Kuvio 6), mikä piilottaa tekstuuriin toistuvuutta (Albeluhn n.d.). Multiply shaderien lisäksi käytetään myös triplanar shadereita, joille voidaan määrittää erilaisia falloff-arvoja. Käytännössä tämä tarkoittaa sitä, että 3D-mallin pinnan kulma määrittää tekstuurien vaihtumisen (Kuvio 7). Esimerkiksi ylöspäin osoittaville pinnoille voidaan määrittää lunta, sammalta tai pölyä.



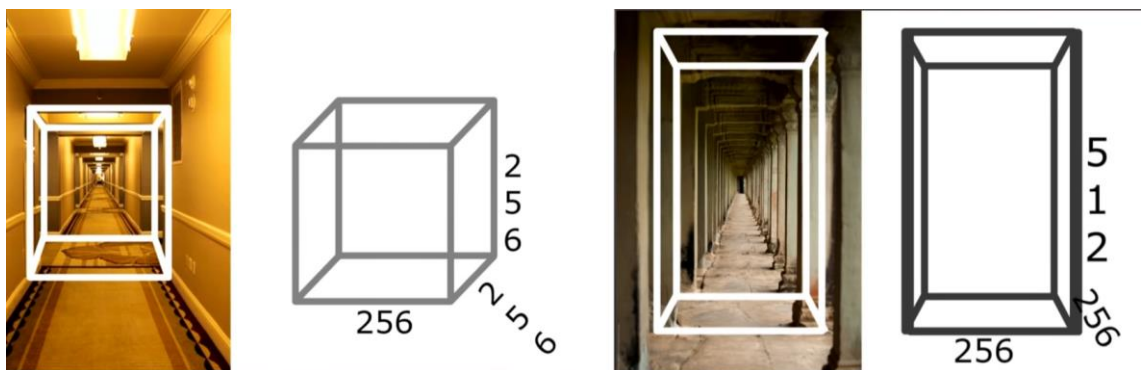
Kuvio 6. Ylhäällä voimakkaasti toistuva tekstuuri ilman multiply shaderia. Alakuvassa sama tekstuuri, jonka toistumista on paljon vaikeampi havaita shaderin ansiosta (Albeluhn n.d.).



Kuvio 7. Esimerkki triplanar shaderista, joka laskee automaattisesti tekstuurien sekoittamisen mallin pintakulman mukaan (Robinson-Yu 2020).

### 3.3 Modulaarisuuden rikkominen tasosuunnittelulla

Modulaariset assetit perustuvat gridiin. Gridin tarkoitus on määrittää assettien koko ja muoto. Gridillä myös varmistetaan, että kaikki saman setin palaset liittyvät yhteen saumattomasti. Visuaalisesti gridien ongelmana on se, että ne synnyttävät rakennelmia, jotka muodostavat voimakkaita 90 asteen kulmia. Peleissä näitä kulmia halutaan rikkoa ja luoda mahdollisimman paljon variaatiota erilaisilla kokosuhteilla, koska pelaajan immersio pelimaailmaan häiriintyy, jos se tuntuu gridin pohjalle rakennetulta. Modulaaristen palojen saumaton yhteenliittyminen varmistetaan käyttämällä lukuja 16, 32, 64, 128, 256 ja 512, koska ne voidaan aina kertoa tai jakaa kahdella. (Bethesda Games Studios 2016)

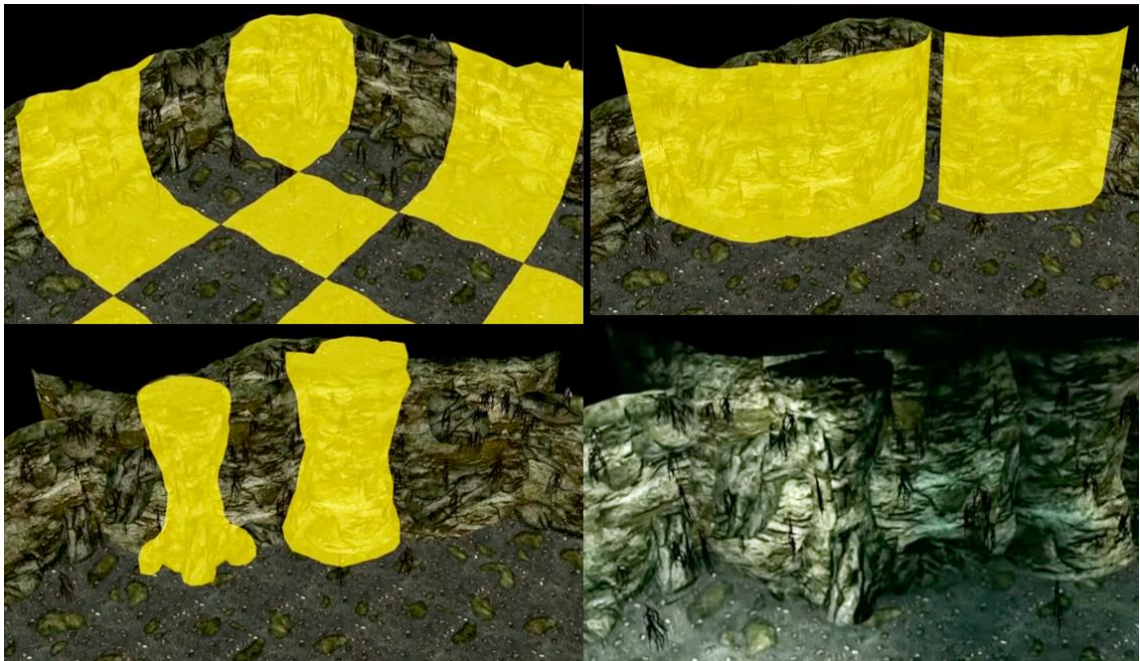


Kuvio 8. Modulaaristen palojen kokosuhteita muuttamalla luodaan nopeasti erilaista tunnelmaa (Bethesda Game Studios 2016).

Palojen ei tarvitse olla kokosuhteiltaan tylsiä neliöitä (256x256) kuten kaikissa vanhoissa peleissä. Neliöitä käytetään, koska muoto on helppo saada toistumaan ja linkittymään kaikissa suunnissa itsensä kanssa, mikä tekee tasojen suunnittelusta nopeaa, mutta visuaalisesti tylsän näköisiä. Nykyään neliöiden lisäksi käytetään erikokoisia paloja, jotka voivat olla esimerkiksi 512x256 tai 256x512 kokoisia. Modulaaristen palojen kokosuhteiden vaihtelu antaa peliympäristöön erilaisia tunnelmia (Kuvio 8), mutta huonona puolena on se, että palat linkittyvät yleensä vain yhdensuuntaisesti keskenään, mikä tarkoittaa suurempaa työmäärää ja useampien modulaaristen palojen luomista, että ongelma voidaan kiertää. (Bethesda Games Studios 2016)

Jätän aiheen pintaraapaisuksi, koska sen syvällisempi käsittely ei ole opinnäytetyön kannalta olennaista. Suosittelen kaikkia ympäristöjen luomisesta kiinnostuneiden henkilöiden tutkimaan lähteenä ollutta Bethesda Games Studiosin [luentoa](#), joka tarjoaa hyvin kattavan kuvan siitä, miten modulaarisuus on kehittynyt alalla viimeisten vuosien aikana.

Palaan hetkeksi takaisin 90 asteen kulmien ongelmallisuutteen, koska nurkkia syntyy väistämättä modulaaristen settien parissa työskennellessä. Nurkkia halutaan välttää, koska ne ovat visuaalisesti tylsiä ja muistuttavat pelaajaa gridistä. Luettelen nyt muutamia tapoja, joita käytetään modulaarisuuden piilottamiseen peliympäristöissä. Tekniikat ovat laajassa käytössä, koska suuri osa peleistä rakentuu edelleen modulaarisuuden varaan. Usein nurkat yksinkertaisesti täytetään romulla tai esineillä, jotka estävät pelaajan pääsyn kulmaan. Yleensä nurkassa olevat assetit asetellaan diagonaalisesti, koska diagonaalit tekevät tilasta visuaalisesti kiinnostavamman, ne luovat visuaalista liikettä ja ne mahdollistavat 90 asteen kulman rikkomisen, jonka jälkeen gridi ei enää ole yhtä helposti huomattavissa. (Perko 2017)



Kuvio 9. Gridin 90 asteen nurkka on häivytetty luonnollisemmaksi lisäämällä nurkkaan erillisiä assetteja (Bethesda Game Studios 2016).

Orgaanisissa ympäristöissä 90 asteen kulma voidaan piilottaa lisäämällä nurkkaan erillisiä orgaanisia assetteja (Kuvio 9). Seinistä läpi menevät kohdat täytetään saumojen piilottamiseen suunnitelluilla asseteilla. (Bethesda Games Studios 2016) Erilaiset siirtymät modulaaristen settien välillä hoidetaan esimerkiksi lisäämällä lankkuja tai romua siirtymäkohdan päälle, koska romut auttavat katsojan silmää punomaan eri setit visuaalisesti yhteen (Kuvio 10), jonka seurauksena pelaaja liikkuu liitoskohdan yli huomaamatta setin muutosta (Perko 2017).



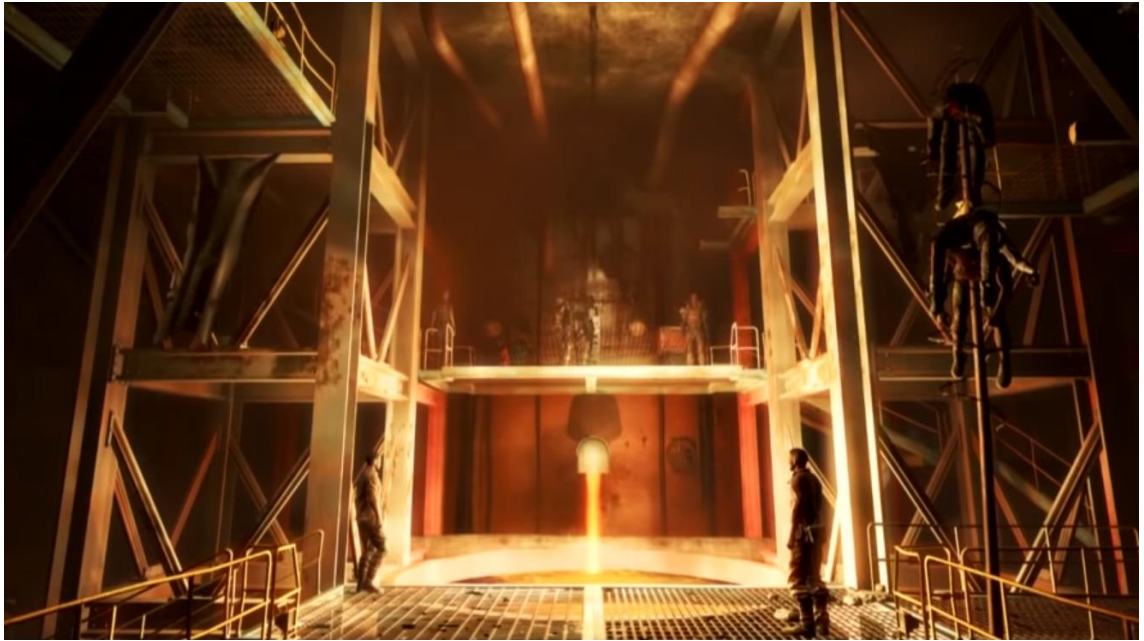
Kuvio 10. Romua käytetään piilottamaan modulaarisen setin vaihtuminen betonimateriaalista metalliin. Huomaa sauma, joka kulkee metallilaattojen alla (Platinum Games 2017).

Esimerkiksi romun luomisessa trim-tekstuurit paljastavat hyödyllisyytensä, koska projektissa jo käytettyjä trim-tekstuureja voidaan käyttää uusien ja yllättävien asettien teksturoimiseen (Simpson 2019). Peliprojektiin ei täten tarvitse tuoda lisää tekstuuritiedostoja tai lähteä muokkaamaan projektissa jo olevia tiedostoja.

#### 3.4 Hero asettien luominen modulaarisista paloista

Bethesda games studios varoittaa modulaarisuusluennossaan myös hero asettien ansasta. Hero asetit ovat hyvin näyttäviä, mutta suuritöisiä asetteja, joiden ongelmana on se, että ne ovat helposti tunnistettavia ja niiden uudelleenkäyttö pelissä on haasteellista. Hero asettien suurena vaarana on niihin panostaminen väärään aikaan peliprojektia, koska projektin kannalta olisi parasta keskittyä useasti näkyviin isoihin pintoihin, kuten esimerkiksi seiniin, lattioihin, kattoihin ja oviin, koska nämä elementit tulevat toistumaan pelissä aina uudestaan ja uudestaan ja niillä rakennetaan suurin osa pelin maailmasta. Studiolla on myös huomattu, että monia kenttien ongelmakohtia saatiin ratkaistua käyttäen modulaarisia paloja, vaikka tiimin ensimmäisenä reaktiona ongelmiin oli refleksin omainen hero asettien luominen. Ylimääräisten hero asettien välttäminen säästää tietenkin aikaa ja resursseja, mutta kaikkein tärkeimpänä studio piti sitä, että he onnistuivat estämään pullonkaulojen syntymisen art design ja level design -tiimien välille, mitä hero asetit helposti aiheuttavat. (Bethesda Games Studios 2016)



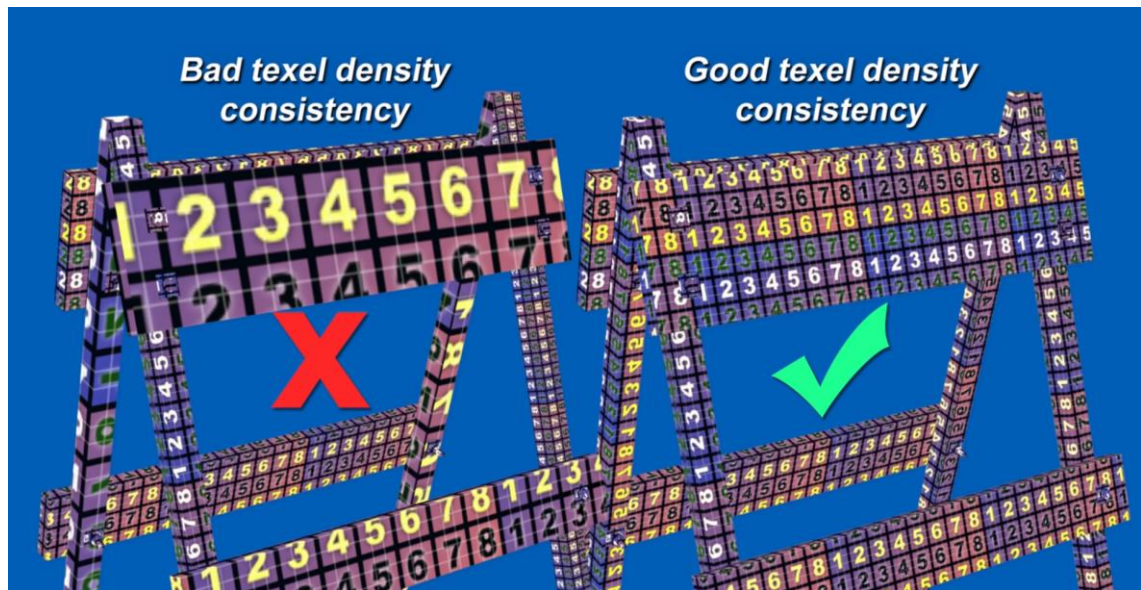


Kuvio 11. Modulaarisista paloista rakennettu hero asset. Valaistus ohjaa katseen keskelle kuvaa. Huoneen symmetrisyys viestittää voimaa ja valtaa. Visuaalista viestiä vahvistetaan asettamalla henkilöt korkeammalle tasolle kuin pelaaja (Bethesda Game Studios 2016).

Lukijalle tässä aluvussa tärkeää on huomata se, että trim-tekstuureja voidaan käyttää modulaaristen palojen teksturoimiseen, joilla voidaan luoda jopa hero asetteja muistuttavia rakennelmia. Tilan valaistus ja visuaalinen tarinankerronta nousevat kuitenkin korostettuun rooliin, koska modulaariset palat joudutaan pitämään visuaalisesti hyvin geneerisinä (Bethesda Games Studios 2016).

### 3.5 Texel density

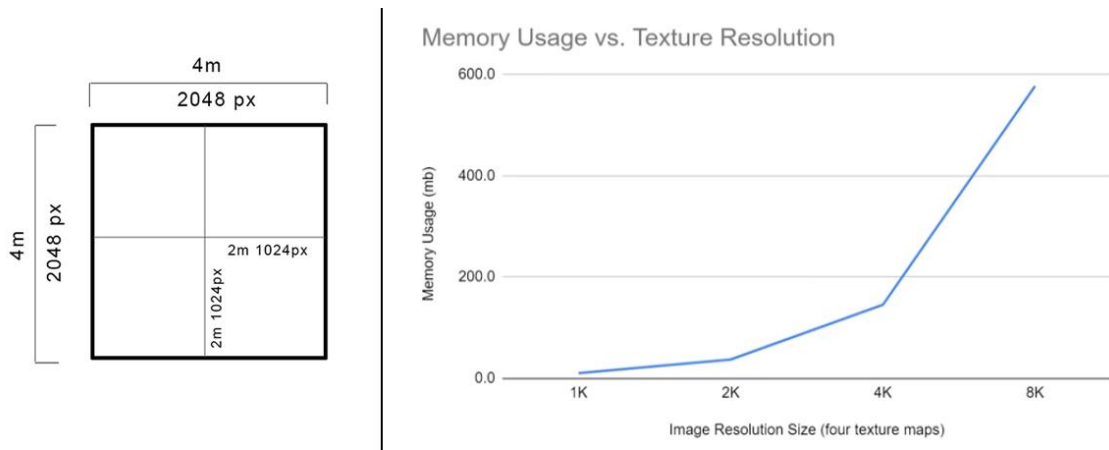
Texel on lyhenne sanoista texture element, joka on UV-tekstuuritilan perusyksikkö. 3D-mallien tekstuurit koostuvat texeleistä samalla tavalla kuin kuvat koostuvat pikseleistä. Texelit ja pikselit eivät kuitenkaan ole sama asia. 2D-kuvissa pienin yksikkö on pikseli, joka ei muutu, mutta 3D-mallien pinnalla texelin koko voi olla mitä tahansa. Tämä johtuu siitä, että 3D-mallin pinnalta auki käärittyjä UV-saaria voidaan tekstuuritilassa skaalata ja saada toistumaan loputtomasti. Joskus texel densityä kutsutaan myös pixel tai texture densityksi, mutta niillä tarkoitetaan samaa asiaa. (malcolm341 2018)



Kuvio 12. Kuvassa vasemmalla 3D-mallin eri osilla on ailahteleva tekstuuriresoluutio, koska mallin texel density vaihtelee voimakkaasti (malcolm341 2018).

Trim-tekstuurien kannalta olennaisin asia on se, että texel density pyritään pitämään mahdollisimman tasaisena aseteissa, jotka on teksturoitu trim-tekstuuritekniikalla. Yksi trim-tekstuurien ongelmista on se, että UV-saaria joudutaan joskus venyttelemään epäoptimaalisella tavalla. Tämä johtaa tilanteisiin, joissa texel density kärsii (Kuvio 12). Epäjohdonmukainen texel density näyttää 3D-mallin pinnalla hyvin epämiellyttävältä. Osa 3D-mallin tekstureista voi näyttää HD-laadulta ja toinen osa mallista aamupurolta. Tämä ongelma pitää ottaa huomioon trim-tekstuurin suunnittelussa. Tekstuurissa kannattaa olla korkeudeltaan erikokoisia liuskoja, koska se mahdollistaa tekstuurin monipuolisemman käyttämisen erilaisissa tilanteissa ilman, että optimaalista texel densityä rikotaan. Texel densityä voidaan laskea erilaisilla kaavoilla, mutta niiden osaaminen ei ole välttämätöntä trim-tekstuurin tekoon. (Simpson 2019)

Useissa nykypäivän PC- ja konsolipeleissä käytetään kaavaa missä yhdellä neliömetrin alueella halutaan saavuttaa 512 pikselin tekstuuriresoluutio. Hyvän trim-tekstuuriresoluution luomiseen riittää tieto siitä, että pelissä saavutetaan 2048x2048 pikselin tekstuuriresoluutio luomalla 3D-mallinnusohjelmassa 4x4 metrin kokoinen polygon plane. Tätä polygonia käytetään pohjana trim-tekstuurin tekoon. 2048x2048 kokoinen tekstuuri on laajasti käytössä, koska se tarjoaa hyvän resoluution, mutta se ei käytä liikaa grafiikka-suorittimen tekstuurimuistia, jota kutsutaan nimellä texture memory. (Simpson 2019)



Kuvio 13. Vasemmalla selventävä kuvio siitä, kuinka ison alueen voi teksturoida 1K ja 2K tekstuurikartalla. 3D-ohjelmassa luodaan mittojen mukainen polygon plane, jota käytetään trim-tekstuurin lowpoly-mallina, jonka päälle aletaan mallintamaan trim-tekstuurin highpoly-mallia. Oikealla kuvaaja tekstuurikarttojen eksponentiaalisesti nousevasta muistin käytöstä (Blender Guru 2020).

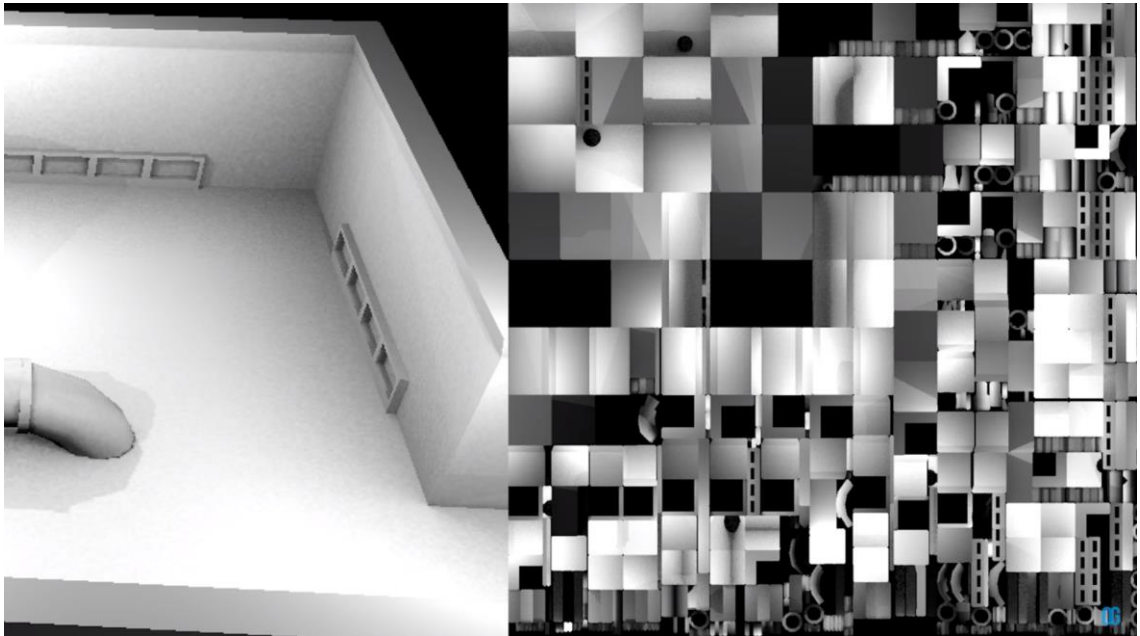
### 3.6 Light map

Toinen trim-tekstuurien teknisistä ongelmista ilmenee light mapin kanssa. Light map on erityinen tekstuuri, johon tallennetaan esilaskettua valoa peliympäristön staattisista objekteista (Kuvio 14), mikä vapauttaa tietokoneen laskentatehoa muihin tehtäviin (Overlord Gaming 2018). Esilaskettu valo ei kuitenkaan toimi oikein, jos 3D-mallien UV-saaret ovat päällekkäin. Tämä on ongelma, koska päällekkäisten UV-saarien hyödyntäminen on yksi trim-tekstuurien vahvuuksista. Ongelma pitää kiittää, koska esilaskettuvalo on useassa pelissä hyvin tärkeässä roolissa pelin suorituskyvyn optimoinnin kannalta.

Ratkaisuna toimii toisen UV-layoutin tekeminen samasta 3D-mallista, jossa ei ole UV-saarien päällekkäisyyksiä. Tämä uusi UV-layout määritetään pelimoottorissa light mapin laskemiseen. Esimerkiksi Unreal Engineissä pystytään automaattisesti generoimaan light mappi missä ei ole päällekkäisyyksiä, mutta tulokset eivät ole kovinkaan optimaalisia UV-saarien pakkaamisen kannalta mikä tarkoittaa hukkaan heitettyä light map -resoluutiota. UV-saarien pakkaaminen kannattaa tehdä käsin, koska pienempi light map -resoluutio on pelin optimoinnin kannalta parempi. (Simpson 2019)

Light mapin UV-saaret kannattaa kääriä auki tavalla, jossa saarien venymistä tapahtuu mahdollisimman vähän ja syntyvät UV-saumot minimoidaan, koska light bleedejä aiheuttaa sitä herkemmin, mitä enemmän UV-saumoja mallissa on. Light bleedissä UV-saarilla olevat valoalueet vuotavat reunojensa yli toisille UV-saarille, minkä seurauksena 3D-mallien pinnoille voi syntyä näkyviä artefakteja. UV-saarien venymistä halutaan välttää,

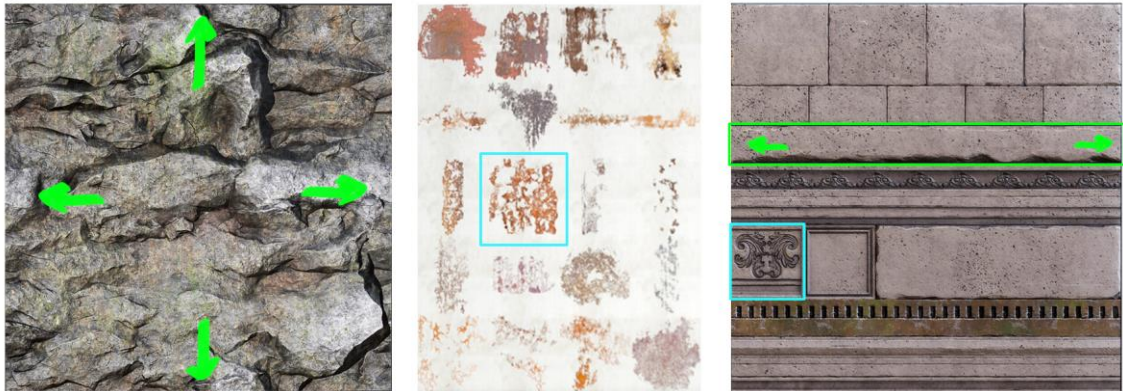
koska muuten valo alkaa taipumaan oudosti 3D-mallien pinnoilla. Light mapin auki käärityt UV-saaret aiheuttavat dilemman tuhlatus UV-tilan ja minimaalisten UV-saumojen kanssa, koska light map vuotaa sitä vähemmän, mitä harvemmin UV-saumoja on, mutta UV-tilan saa täytettyä tehokkaimmin lisäämällä malliin uusia UV-saumoja. (Simpson 2019)



Kuvio 14. Light map, johon on laskettu staattisen ympäristön valoja. (Overlord Gaming 2018)

### 3.7 Peliympäristön kolme tekstuuria

Peliympäristöjen luomiseen käytetään yleisesti kolmea erilaista tekstuuria. Saumattomia tekstuureja, jotka toistuvat loputtomasti niin horisontaalisesti kuin vertikaalisesti. Decal tekstuureja, jotka sisältävät läpinäkyvyyden mahdollistavan alpha kanavan, mikä tekee yksittäisten decalien rajaamisen isommasta tekstuurikartasta ja näiden rajattujen decalien saumattoman istuttamisen peliympäristöön mahdolliseksi. Ja lopuksi vielä molempia aikaisempia tekstuureja yhdisteleviä trim-tekstuureja, jotka yleensä koostuvat erikokoisista trim-liuskoista, jotka toistuvat saumattomasti vain yhdensuuntaisesti. Trim-teksturiin voidaan lisätä myös alueita, joita voidaan käyttää teksturoinnissa kuten decaleita tai UV-atlasointia. UV-atlasointi on teksturointitekniikka missä yhdellä isolla tekstuurikartalla on esimerkiksi kaikki kentän erilaiset tekstuurit, josta voidaan käydä rajaamassa haluttu teksturi tarkoin määritellylle 3D-mallin alueelle. Kaikkia kolmea tekstuuria tulisi käyttää yhdessä peliympäristöjen teksturoimiseen, koska ne tukevat toisiaan ja korjaavat toistensa heikkouksia. (Dsouza 2019)



Kuvio 15. Vasemmalla saumaton tekstuuri, joka toistuu horisontaalisesti ja vertikaalisesti. Keskellä decal tekstuuri, jota käytetään erilaisten erottuvien yksityiskohtien lisäämiseen. Oikealla trim-tekstuuri, joka yhdistää molemmat tekstuuri muodot (Piper 2016), (TextureMax 2015), (Deschamps 2019).

Saumattomia tekstuureja käytetään isojen pintojen teksturoimiseen. Saumattomilla tekstuureilla voidaan myös täydentää alueita, joita on hankala teksturoida trim-tekstuureja käyttämällä. Decal-tekstuureja käytetään erilaisten erottuvien yksityiskohtien lisäämiseen ja niiden tarkoitus on rikkoa toistuvien geneeristen tekstuurien monotonisuutta. Toistuvat saumattomat tekstuurit ja trim-tekstuurit joudutaan pitämään mahdollisimman geneerisinä, jotta niillä voidaan teksturoida suuria alueita, koska selvästi erottuvat yksityiskohdat itse tekstuurissa toistuisivat erittäin häiritsevästi peliympäristössä.

Trim-tekstuureja voidaan ajatella eräänlaisena tekstuurien sveitsiläisenä linkkuveitsenä, joka suunnitellaan ympäristön tarpeiden mukaan. Tekstuuriin voidaan lisätä decalmaisista alueita ja yksityiskohtia, joita voidaan käyttää propeissa ja ympäristöissä sellaisenaan. Tai esimerkiksi kuvassa (Kuvio 15) olevan trim-tekstuurin kivetyksiä voidaan monistaa ja saada toistumaan isolla alueilla UV-saaria manipuloimalla.

Trim-tekstuurien rajoituksena on se, että eri kokoiset liuskat ovat vain yhdensuuntaisesti saumattomia, eikä erilaisten decalien ja trim-liuskojen lisäämiseen ole paljoa tilaa. Trim-tekstuureja käytetään erityisesti hyväksi kaikenlaisissa reunapinnoissa, koska reunojen UV-saaret on helppo asetella trim-tekstuuriin, mikä mahdollistaa erilaisten reunavahinkojen luomisen tekstuurin avulla. Tietyissä tilanteissa toistuvia saumattomia tekstuureja käytetään yhdessä trim-tekstuurien kanssa, koska 3D-mallien epämääräisiä UV-saaria on vaikea saada aseteltua trim-tekstuurin päälle.

### 3.8 Kolme tapaa käyttää trim-tekstuuria

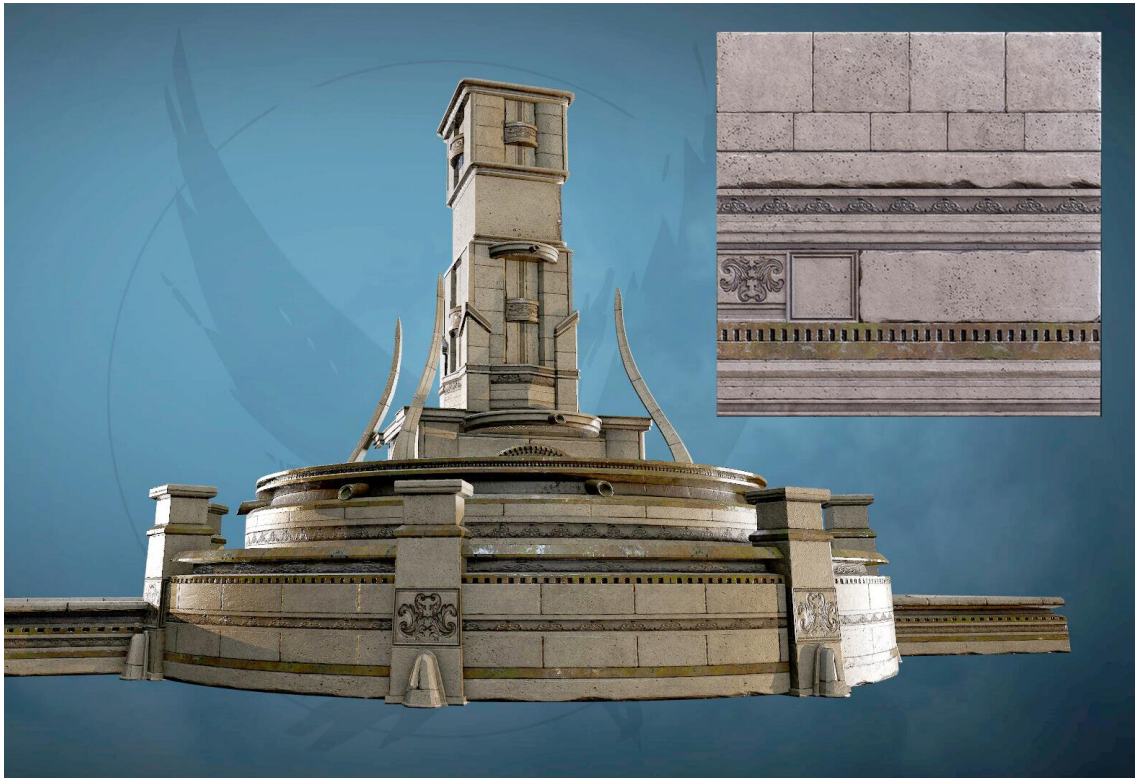
Tutkiessani trim-tekstuureja olen huomannut kolme tyypillistä tapaa, kuinka trim-tekstuureja hyödynnetään teksturoimisessa. Käyttötapa riippuu projektin teknisistä rajoitteista ja siitä mitä halutaan saavuttaa. Trim-tekstuureilla voidaan halutessa huijata bevelöidyt reunat lowpoly-geometriaan normal mapin avulla. Tekniikka säästää polygoneja ja visuaalisesti huijaa veitsenterävät lowpoly-reunat pois näkyvistä, mutta reunat ovat kuitenkin vielä löydettävissä tietyissä ääriasennoissa, koska normal map ei vaikuta lowpolyn geometriaan. Normal map on RGB-kuva, johon tallennetaan jokaisen pikselin suunta. Näitä suuntia kutsutaan normaaleiksi. Kuvan Red, Green ja Blue-kanavilla kontrolloidaan arvojen muuntamista X-, Y-, Z-arvoiksi, joita käytetään simuloimaan pintojen yksityiskohtia. (Get Learnt 2019)

Edellä mainitussa lowpoly-tekniikassa on erittäin tärkeää se, että trim-tekstuurin normal mapissa olevat trim-liuskojen bevelit ovat 45 asteen kulmassa, mikä mahdollistaa terävissä reunoissa pintojen yhdistämisen saumattomalla tavalla. Todennäköisesti siksi, koska kaksi 45 asteen beveliä muodostavat yhdessä 90 asteen kulman, joka asettuu 45 asteen kulmaan. En kuitenkaan löydä tälle väitteelle lähdettä, mutta tiedän että saumoja ilmestyy, jos tekstuurin trim-liuskojen bevelit eivät ole tasan 45 astetta. Kuviossa 16 esimerkki lowpoly-tekniikalla mallinnetuista 3D-malleista, jotka on teksturoitu trim-tekstuureilla.



Kuvio 16. Lähes kaikki pinnat voidaan jakaa niin, että ne voidaan teksturoida trim-tekstuurilla. Esimerkkejä pelistä Sunset Overdrive. Pelissä lowpoly-mallien reunat huijattiin näyttämään bevelöidyltä normal mapin avulla. Peliin kehitettiin standardisoitu trim-tekstuurilayout, joka teki assettien materiaalien vaihtamisesta vaivatonta (Olsen 2015).

Trim-tekstuureilla suositaan myös mallinnustapaa, jossa 3D-mallien reunat oikeasti bevelöidään. Tämä vapauttaa trim-tekstuurit täsmällisestä 45 asteen säännöstä, koska 3D-mallin teräviä reunoja ei enää yritetä piilottaa normal mapin avulla ja teksturoinnista syntyviä saumoja pystytään piilottamaan trim-tekstuurissa oleviin saumoihin. Tämä tekniikka keskittyy enemmän trim-tekstuurin maksimaaliseen hyödyntämiseen. Trim-tekstuurin normal mapin bevelit ovat edelleen tärkeitä, jotta tekstuurin käytettävyys säilyy, mutta nyt bevelöityihin saumoihin voidaan sculptata enemmän vaurioita. Sculptaamisella tarkoitan digitaalista kuvanveistoa esimerkiksi Z-brush ohjelmassa.



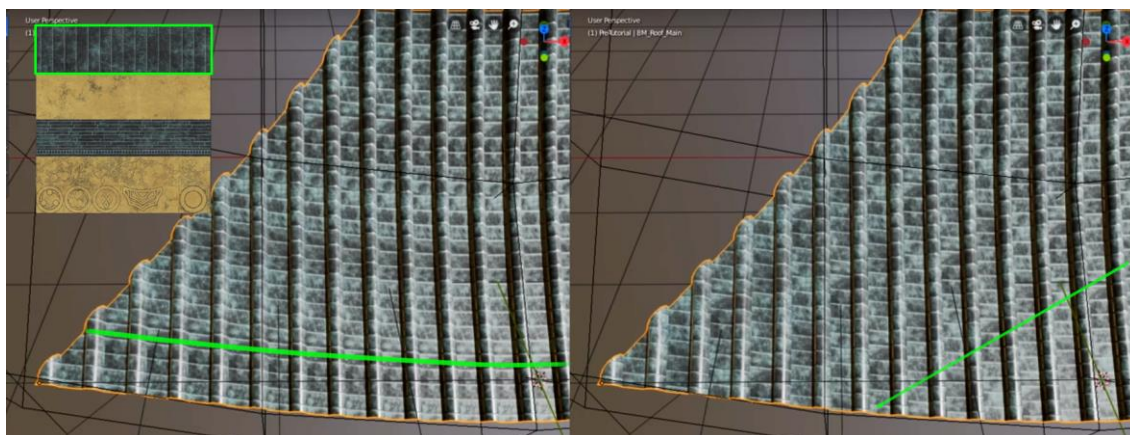
Kuvio 17. Asetti pelistä Haven. 3D-mallin teksturoimiseen on käytetty pelkästään oikeassa yläkulmassa näkyvää trim-tekstuuria (Deschamps 2019).

3D-malleihin lisätään topologiaa 3D-mallinnusohjelmassa tarpeen mukaan, eikä polycounteissa säästellä, jos trim-tekstuurilla halutaan saavuttaa jokin tietty visuaalinen lopputulos. Tämä korostuu erityisesti lieriömäisten muotojen kanssa. 3D-mallien geometriassa turhalta näyttävien edge looppien määrä lisääntyy, koska yhtenäisestä meshistä halutaan pystyä irrottamaan UV-saaria, jotka voidaan asetella eri kohtiin trim-tekstuuria. Tämän lisäksi tietyissä tilanteissa yritetään minimoida tekstuurin venymistä lisäämällä geometriaan ylimääräisiä looppeja. Esimerkiksi ympyrän muotoisissa elementeissä tekstuurit venyvät voimakkaasti keskipistettä kohti, koska niiden UV-saaret on pakotettu suorakulmion muotoon. Tässä vapaan mallinnuksen tekniikassa hyvää on se,

että 3D-mallintaminen ja teksturointi tuntuu huolettomalta ja nopealta. Tekijä saa myös reaaliaikaisen visualisoinnin valmiista teksturoidusta mallista, kun UV-saaret siirretään trim-tekstuurin päällä oikeille paikoilleen. Vapaasta mallinnuksesta aiheutuva suuri polycount ei enää nykypäivänä ole maailmanloppu PC- ja konsolipeleissä. Nykypeleissä polygoneja suurempi ongelma on tekstuurien viemä muisti, johon trim-tekstuurit tuovat helpotusta. Polygonmäärät vaihtelevat projektin ja assetin tärkeyden mukaan ja turhia polygoneja kannattaa tietenkin edelleen välttää. Lukija voi halutessaan tutkia tästä [linkistä](#) erilaisten peliasettien varmistettuja polygonmääriä. Valkoisia linkkejä klikkaamalla voi päästä tutkimaan hyvin mielenkiintoisia asetteja, kuvia tai artikkeleita.

Pelimoottoreissa polygonmääriä optimoidaan vielä erikseen LODien avulla. LOD on lyhenne sanoista level of detail. Käytännössä tämä tarkoittaa sitä, että jokaisella pelin objektilla on monta eri 3D-mallia, joiden geometriat eroavat toisistaan. Malleissa on tärkeää yrittää säilyttää objektin siluetti, mutta karsia niin paljon polygoneja kuin mahdollista. Pelimoottori vaihtaa malleja lennosta sitä mukaan, kun pelaaja liikkuu pelimaailmassa.

Pelaajan lähellä olevat mallit ovat lopullisia artistin luomia ja pelimoottoriin implementoituja malleja, kun taas kauempana olevissa objekteissa polygonien määrä vähenee asteittain sitä mukaan, mitä kauempana objekti pelaajasta on. Yhdellä artistin pelimaailmaan luomalla assetilla voi olla vaikka kolme polygoneiltaan karsittua mallia alkuperäisen mallin lisäksi. Pelimoottorissa näiden mallien automaattista vaihtumista kutsutaan level of detailiksi.



Kuvio 18. UV-offsetilla muutetaan tekstuuripalan toistuvuus suorassa linjassa. Tekstuurin toistuvuus on kuitenkin edelleen näkyvää ja siksi toistuvuuden piilottamiseen käytetään pelimoottorissa vielä shaderia, joka viimeistelee tekstuurin (GameTextures.com 2019).



Trim-tekstuurissa olevaa trim-liuskaa voidaan myös monistaa isoille pinnoille UV-offsettia hyödyntämällä, mikä on kolmanneksi yleisin tapa käyttää trim-tekstuureja. UV-offset on tapa manipuloida tekstuurin toistumista 3D-mallinnusohjelmissa, joka auttaa naamoimaan tekstuurin luonnottoman toistuvuuden (Kuvio 18). Tämä mahdollistaa suurien pinta-alojen teksturoimisen pienellä palalla tekstuuria. Palan on oltava hyvin geneerinen, koska se toistuu paljon, mutta siitä voidaan tehdä visuaalisesti kiinnostavampi pelimoottorin puolella shadereita ja decaleita käyttämällä.

#### 4 Trim-tekstuurin hyödyt

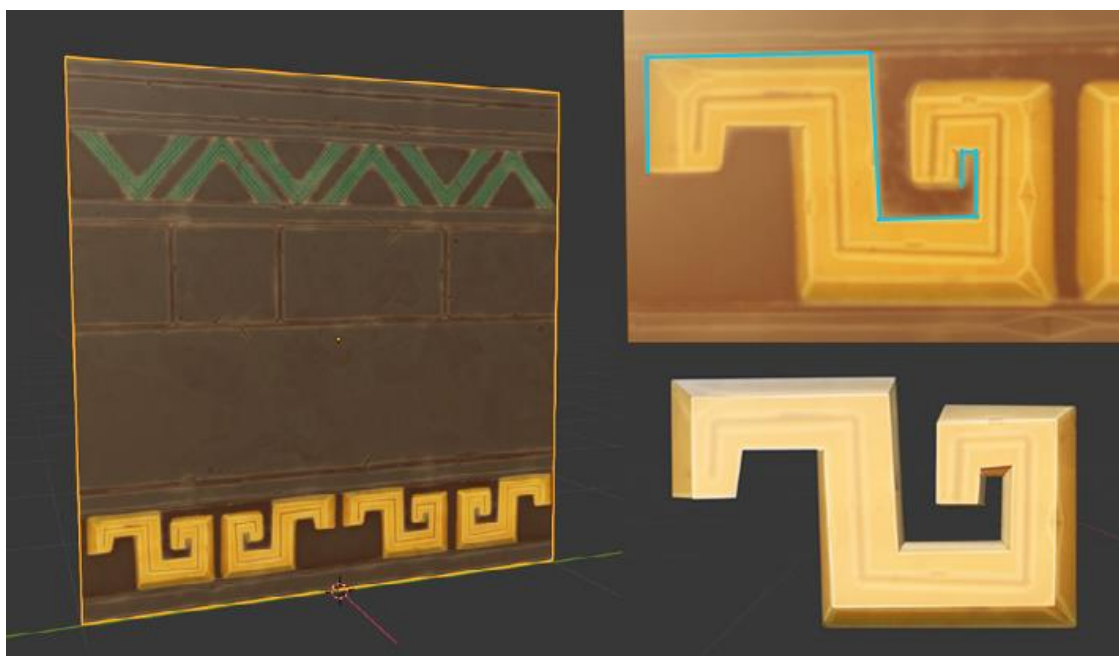
Trim-tekstuurien käyttö tuo mukanaan erilaisia etuja. Trim-tekstuureilla saavutetaan automaattisesti yhtenäinen vaikutelma useiden eri asettien välille, koska kaikki assetit voidaan teksturoida samalla tekstuurilla. Trim-tekstuureilla voidaan huijata pieniä reunavahinkoja vakuuttavasti, mikä vähentää vahinkojen sculptaamisen tarvetta ja highpoly-malleja. Pelimoottorin puolella saavutetaan graafisesti korkealaatuinen lopputulos isoilla alueilla, kun trim-tekstuurien modulaarisuutta hyödynnetään saumattomien materiaalien kanssa. (Simpson 2019)



Kuvio 19. Esimerkki trim-tekstuurista. Pallon keskisauma saadaan syntymään, kun tekstuurikartan ulkoreunat bevelöidään. Trim-liuska on saumaton, jos tekstuurin reunojen bevelöinnin jättää pois (Deschamps 2019).

Trim-tekstuureilla teksturointi poikkeaa normaalista 3D-mallin työnkulusta. Tekstuuri pitää luoda ensin, jotta sitä voi käyttää hyväksi 3D-mallien mallinnuksessa. Trim-tekstuurit

vaativat suuremman työpanoksen assetin tekovaiheen alussa, mutta tekstuurilla säästetään sitä enemmän aikaa, mitä useammassa assetissa tekstuuria käytetään. Aikaa trim-tekstuurin tekemisessä kuluu sen suunnittelemiseen, mallintamiseen ja teksturoimiseen. Valmiin trim-tekstuurilla teksturoidun assetin päivittäminen on helppoa, koska 3D-mallien UV-saarien paikkoja voidaan siirtää vapaasti tekstuurin pinnalla ilman pelkoa siitä, että mallin tekstuurit ja normal map pitäisi luoda uudestaan. Kaikki trim-tekstuurilla tehdyt assetit päivittyvät automaattisesti, jos itse tekstuuriin tehdään muutoksia. Trim-tekstuurilla tehdyissä 3D-malleissa riittää, että geometrian muutokset tehdään pelkästään low-poly-malliin ja UV-saaria siirretään tarvittaessa. Trim-tekstuureilla tehdyillä assetilla ei ole highpoly-mallia. Tämä säästää paljon aikaa ja highpoly-mallinnuksesta syntyviä tiedostoja. Ympäristöjen luomiseen tarvitaan vähemmän erilaisia tekstuureja, koska samoja tekstuureja voidaan käyttää monipuolisemmin.



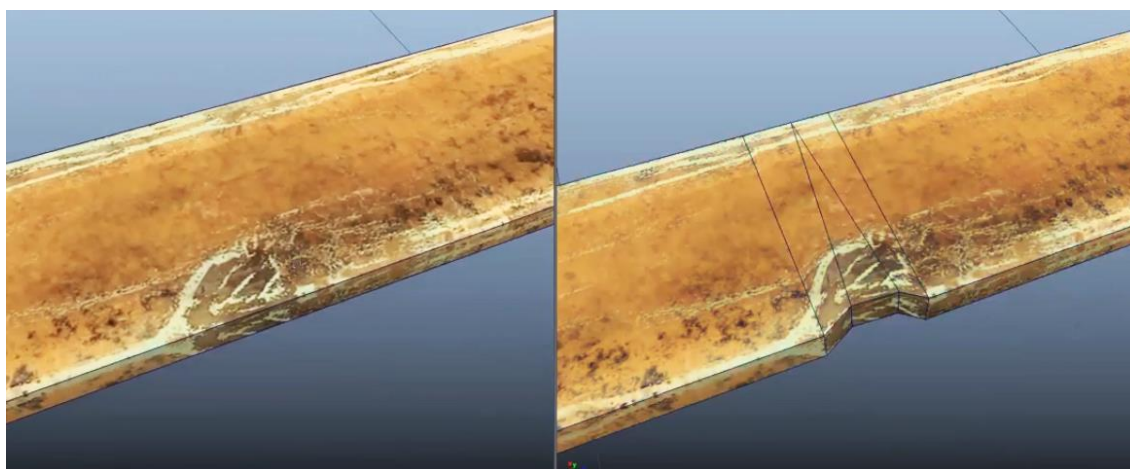
Kuvio 20. Trim-tekstuurista voidaan leikata irti paloja, joita voidaan yhdistellä modulaarisesti. Kuvassa trim-tekstuurista on irrotettu kuvio, jota voi käyttää luomaan erilaisia yksityiskohtia (McNally 2020).

Trim-tekstuurin pinta-ala on helppo jakaa puolittamalla standardisoituihin liuskakokoihin, koska trim-teksturi on muodoltaan neliö ja se täyttää koko tekstuurimapin. Yhdellä trim-tekstuurilla teksturoidut assetit on mahdollista korvata toisella trim-tekstuurilla, jos trim-tekstuurin liuskojen layout pysyy täsmälleen samana, koska UV-saaret on jo aseteltu oikeille paikoilleen ensimmäisen tekstuurin pinnalla. Käytännössä tämä luo lähes ilmaiseksi uuden version kaikista ensimmäisellä trim-tekstuurilla teksturoidusta asse-

teista. Samaa layouttia käyttäviä tekstuureja kutsutaan trim-tekstuurisetiksi. Trim-tekstuureja voidaan käyttää projektin loppupuolella yllättävien asettitarpeiden mallintamiseen ja teksturoimiseen nopealla aikataululla, koska projektin trim-tekstuurit on jo tehty. Trim-tekstuureissa on hyvää myös se, että 3D-mallintamisesta ja UV-tilassa työskentelestä tulee miellyttävällä tavalla aktiivisempaa, koska aikaa ei tarvitse käyttää UV-saarien pakkaamiseen ja mallinnuksen aikana saat reaaliaikaisen tiedon teksturoidusta lopputuloksesta.

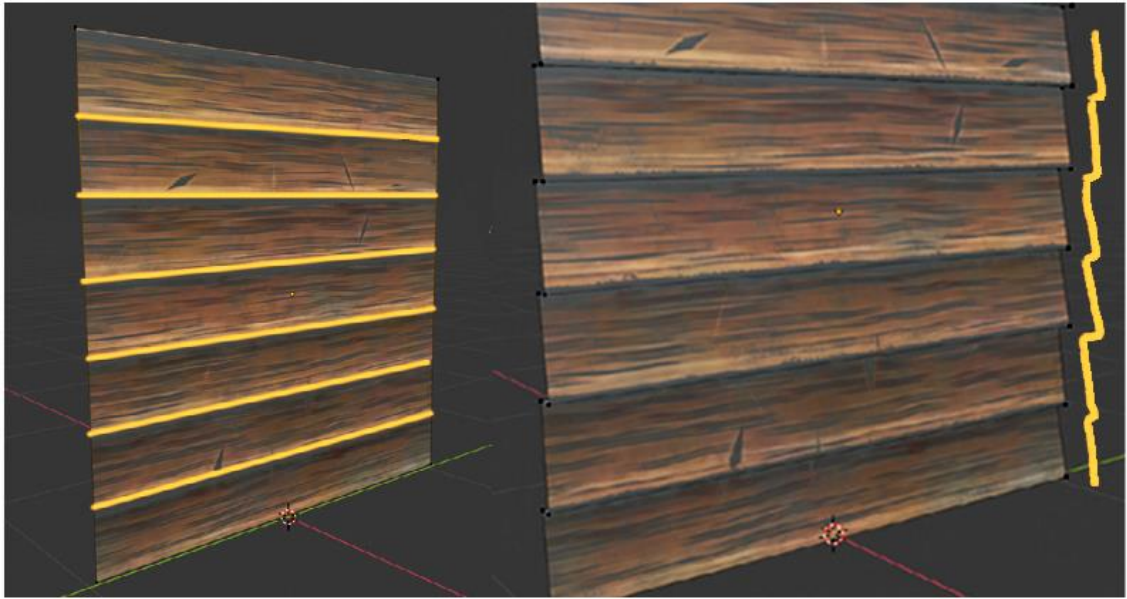
Trim-tekstuurissa olevista kuvioista voidaan myös helposti luoda kohokuvia (Kuvio 20). Kuvio leikataan irti työkalulla, jolla voidaan leikata polygonin geometriaa (esimerkiksi Mayan multi-cut toolilla), jonka jälkeen leikattu polygon valitaan ja extrudetaan. Tekstuurista irti leikattua 3D-mallia voi tämän jälkeen monistaa ympäristöön ja käyttää luomaan erilaisia kuvioita palan rotaatioita muuttamalla.

3D-mallin geometriaa muuttamalla voidaan luoda myös erilaisia vaurioita, joilla korostetaan trim-tekstuurissa olevia vauriokohtia. Kuvassa (Kuvio 21) on trim-tekstuurilla teksturoitu koristelista, missä koristelistan 3D-mallin geometriaan mallinnetaan vauriokohta, koska listaan halutaan saada lisää visuaalista vaihtelua.



Kuvio 21. Trim-tekstuurilla teksturoituun 3D-malliin on lisätty vauriokohta, joka tekee mallista visuaalisesti kiinnostavamman. Vaurioita mallintamalla voidaan korostaa trim-tekstuurissa olevia vauriokohtia (Dsouza 2019).

3D-mallin geometriaa muuttamalla luodut vauriot kannattaa lisätä vasta aivan loppuksi, koska tekstuuri vaihtaa paikka 3D-mallin pinnalla, jos mallin UV-saaria siirtää vaurion mallintamisen jälkeen. Tämä johtaa tilanteeseen missä tekstuurissa oleva vaurio ja 3D-mallissa oleva vaurio eivät osu enää kohdakkain. (Dsouza 2019)

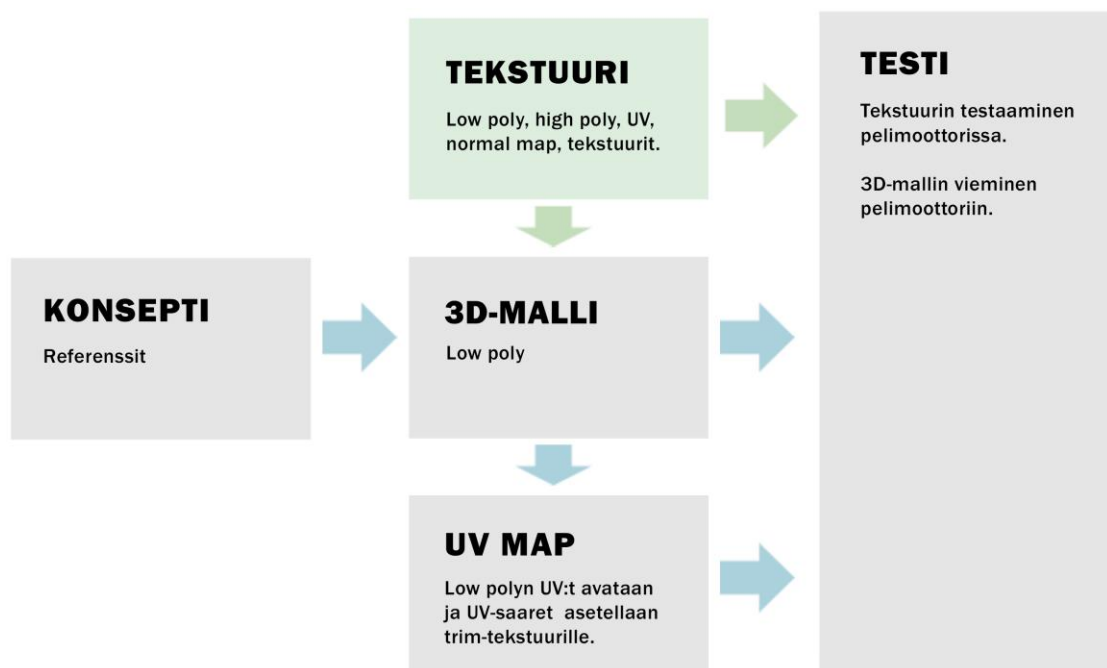


Kuvio 22. Trim-tekstuureja käytettäessä kannattaa pitää mielessä se, että 3D-mallin geometriaa muokkaamalla voidaan korostaa tekstuurissa normal mapilla luotua illuusiota. Kuvassa saumattomaan teksturiin on mallinnettu geometriaa, joka tekee tekstuurista vaikuttavamman näköisen (McNally 2020).



Kuvio 23. Kuva normaalista 3D-mallin työnkulusta, joka etenee jäykkänä linjastona.

Kuvaaja 3D-mallin normaalista työnkulusta (Kuvio 23), jossa malli ensin suunnitellaan, mallinnetaan, UV-mapataan, teksturoidaan ja testataan pelimoottorissa. Teksturoitua asettia päästään kokeilemaan peliympäristössä vasta aivan prosessin loppuksi. Normaali 3D-mallin työnkulku aiheuttaa ongelmia, jos teksturoitua 3D-mallia joudutaan muuttamaan teksturoinnin jälkeen, koska se toimii jäykkänä linjastona. Vähänkään suurempi 3D-mallin muuttaminen vaikuttaa automaattisesti myös UV-mappiin ja teksturointitiedostoon, jotka pitää käydä korjaamassa. Tähän prosessiin hukkuu helposti aikaa, koska artistin pitää hypätä ohjelmasta toiseen päivittämässä tiedostoja.

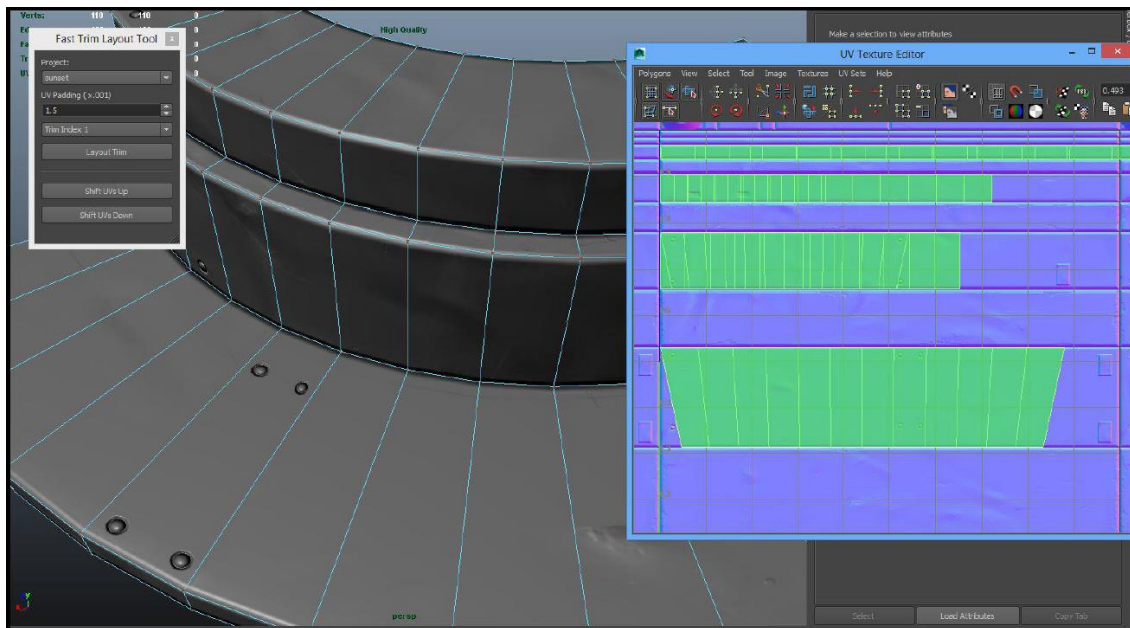


Kuvio 24. Kuva 3D-mallin työnkulusta trim-tekstuurin kanssa. Trim-tekstuuri mahdollistaa tekstuuritiedoston eristämisen 3D-mallista. Trim-tekstuuri pitää sisällään normaalin lowpoly, highpoly, UV, ja teksturointi työnkulun. Trim-tekstuuri voidaan asettaa 3D-mallin tekstuuriksi, jonka ansiosta 3D-mallin UV-saaria pystytään siirtämään vapaasti ilman huolta siitä, että tekstური vaurioituu tai että normal map pitäisi laskea uudestaan, minkä ansiosta korjausten tekeminen 3D-malliin on joustavampaa.

Trim-tekstuurityönkulku on paljon joustavampi, ja se tarjoaa mahdollisuuden hypätä eri työvaiheiden yli (Kuvio 24). Kaikkiin työvaiheisiin pystytään palaamaan myöhemmin uudestaan, koska trim-tekstuuritiedosto pitää sisällään normaalin highpoly, lowpoly, UV, normal map ja teksturoinnin työvaiheet. Tämä mahdollistaa nopean testauksen pelimoottorissa, koska testattavaan ongelmaan voidaan hypätä suoraan, tai priorisoida tarvittavat asiat, jolla ongelmaa voidaan testata. Ongelmien ratkomisen jälkeen voidaan palata viimeistelemään tekstuuria, mallin UV-mappia tai itse 3D-mallia. Kaikki assetit päivittyvät pelimoottorin puolella automaattisesti, jos nimeäminen ja pelimoottoriin vieminen on tehty oikein.

Olen esimerkiksi nähnyt trim-tekstureja käytettävällä tavalla, missä Photoshopissa on ensin hyvin karkeasti piirretty erilaisia kuvia 2048x2048 pikselin kokoiseen kuvaan. Tätä piirrettyä trim-tekstuuria on käytetty 3D-malleissa, jotka on viety pelimoottoriin. Pelikenttää on alettu suunnittelemaan pelimoottorin puolella 3D-malleilla, jotka käyttävät Photoshopilla piirrettyä trim-tekstuuria tekstuurinaan. Kun tekstuurin kanssa on ilmennyt ongelma missä jokin kohta esimerkiksi toistuu liikaa, tekijä voi palata Photoshopiin ja

muuttaa tekstuuria, jonka jälkeen vanha trim-tekstuuri voidaan korvata uudella ja havainnoida, kuinka korjattu tekstuuuri käyttäytyy pelimoottorissa. Trim-tekstuuri on mahdollista mallintaa ja teksturoida vasta sen jälkeen, kun trim-tekstuurin layout on jo todettu toimivaksi pelimaailmassa. Trim-tekstuureilla työskentely voi olla hyvin joustavaa kuten edellä mainitusta esimerkistä voi huomata. Tämän vuoksi tekniikka kannattaakin ottaa haltuun vasta sen jälkeen, kun normaali 3D-mallin työnkulku on tekijälle täysin selvä.



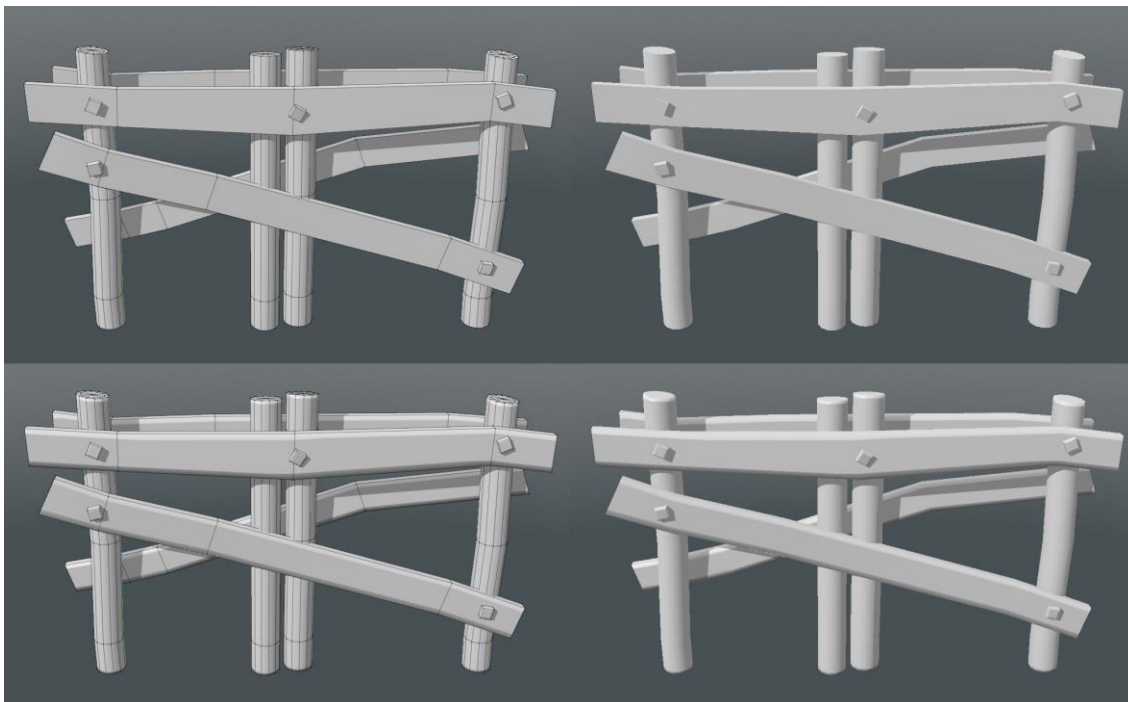
Kuvio 25. Kuvassa 3D-malli, jonka UV-saaret on kääritty auki ja pakotettu suorakulmion muotoon. Vihreät UV-saaret on aseteltu osumaan trim-tekstuurissa oleviin beveleihin, jotka normal mapin ansiosta aiheuttavat illuusion bevelöidyistä reunoista kuvan 3D-mallin terävissä lowpoly-reunoissa (Olsen 2015).

Trim-tekstuureilla teksturointi perustuu ideaan, missä 3D-mallin polygon looppeja kääritään auki ja pakotetaan UV-tilassa suorakulmion muotoon. Tämä mahdollistaa UV-saarien reunojen asettamisen tekstuurissa oleviin beveleihin (Kuvio 25). Saaria voidaan skaalata ja ne saavat mennä yli UV-mapin U1V1 mapista, koska tekstuuuri on reunoista saumaton ja se voi jatkua loputtomasti. Auki käärityt UV-saaret saavat mennä UV mapissa päällekkäin.

Koko 3D-malli puretaan osiin samaa tekniikkaa käyttämällä. UV-saarten asettelu on tarkkaa hommaa, mutta onneksi saumat voi kohdentaa myös silmämääräisesti, koska tekijä näkee tekstuurin muuttumisen reaaliaikaisesti 3D-mallin pinnalla, jos trim-tekstuuri on asetettu 3D-mallin tekstuuriksi. En käy aihetta tämän tarkemmin läpi, koska tekniikan ymmärtää paremmin, kun sen näkee käytännössä. Siksi suosittelenkin lukijaa katsomaan tämän Tim Simpsonin [videon](#) UV-saarien asettelusta trim-tekstuurin päälle.

## 4.1 Huijari

Trim-tekstuureilla ei aina kannata yrittää teksturoida koko 3D-mallia. Trim-tekstuuria kannattaa käyttää, jos malli on helposti jaettavissa poly looppeihin. Trim-tekstuureja suositellaan käytettäväksi saumattomien tekstuurien kanssa, jos mallissa on epämääräisen muotoisia alueita, joista ei muodostu selkeitä poly looppeja. Trim-tekstuureja käytetään erityisesti asettien reunoissa, koska reunat ovat helppoja UV-mapata ja niillä voidaan huijata reunavahinkoja ja beveleitä.

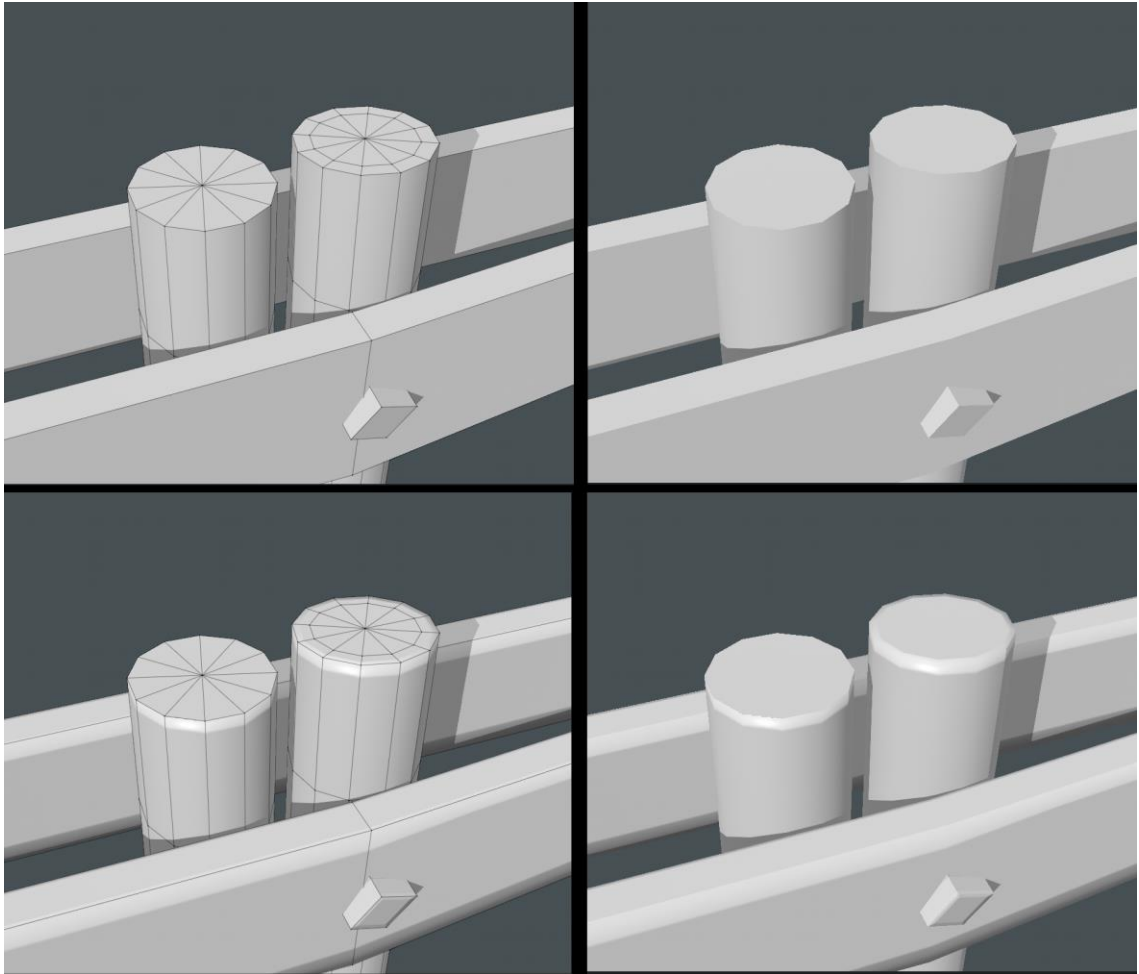


Kuvio 26. Kaikissa kuvissa sama lowpoly-malli. Alarivissä 3D-mallin tekstuuriksi on asetettu trim-tekstuuri, joka käyttää pelkästään normal mappia. Suosittelem lukijalle kuvan suurentamista zoomia käyttämällä paremman käsityksen saamiseksi tuloksista.

Esittelen lukijalle seuraavaksi tuloksia, joita voidaan saavuttaa hyödyntämällä 45 asteen bevel-tekniikkaa lowpoly-asetin terävissä reunoissa. Kuvassa (Kuvio 26) ylärivissä lowpoly-malli wireframen kanssa ja ilman. Alarivissä sama lowpoly-malli wireframen kanssa ja ilman, mutta sillä erolla, että 3D-mallin tekstuuriksi on asetettu trim-tekstuuri.

Olin hyvin positiivisesti yllätynyt siitä, kuinka uskottavan illuusion reunan kääntymisestä voi saavuttaa normal mapin avulla. Illuusio säilyy erittäin uskottavana, vaikka 3D-mallia tarkastellaan eri suunnista. Normal map ei kuitenkaan vaikuta mallin geometriaan ja sen luoma illuusio kärsii äärimmäisissä sivuprofiileissa, joissa mallin oikea laatikkomainen geometria tulee esiin. Normal mapin vaikutusta on myös hyvin vaikea huomata ja säätää

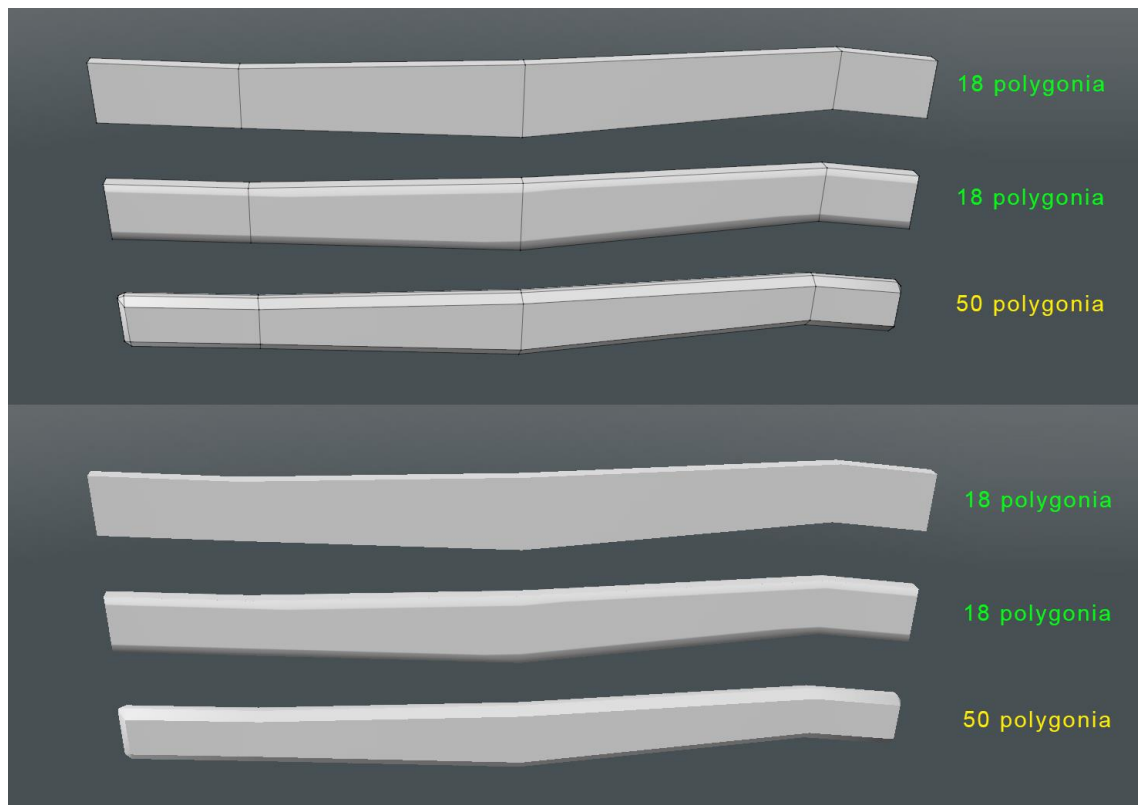
pienissä 3D-malleissa, kuten kuvan aidan nauloissa. Myös ympyrän muotoisissa end capeissa on ongelmia, jos reunaan ei lisätä ylimääräistä edge looppia, kuten kuvan (Kuvio 27) aidanseipäiden vertailusta voi huomata.



Kuvio 27. Kaikissa kuvissa sama lowpoly-malli. Alarivissä 3D-mallin tekstuuri on asetettu trim-tekstuuri, joka käyttää pelkästään normal mappia. Huomaa mallin wireframe trim-tekstuurilla teksturoidussa mallissa, joka paljastaa 3D-mallin oikean geometrian. Suosittelemme lukijalle kuvan suurentamista zoomia käyttämällä paremman käsityksen saamiseksi tuloksista.

Eroa oikeasti geometrialtaan bevelöityyn 3D-malliin on vaikea huomata, kuten kuvasta (Kuvio 28) voi nähdä. Tekniikalla pystytään säästämään mallien polygoneja merkittävästi, jos projektissa tätä halutaan tavoitella. Kuvassa ylhäällä lowpoly-malli, keskellä trim-tekstuurilla teksturoitu lowpoly-malli ja alhaalla reunoista oikeasti bevelöity 3D-malli. Tekniikassa hyödyllistä on se, että 3D-mallit voidaan pitää lowpolyna, joka helpottaa mallin jakamista selkeisiin osiin, joiden UV-saaret pystytään nopeasti avaamaan suorakaitteen muotoon, jonka jälkeen terävistä lowpoly-reunoista päästään eroon, kun UV-saaret asetellaan trim-tekstuurin päällä oikeille paikoilleen.





Kuvio 28. Kuvassa ylhäällä lowpoly-malli, keskellä trim-tekstuurilla teksturoitu lowpoly-malli ja alhaalla reunoista oikeasti bevelöity 3D-malli.

#### 4.2 Säästäjä



Kuvio 29. Kuvassa näkyvät vaunut on teksturoitu yhdellä trim-tekstuurilla. Trim-tekstuuri luo automaattisesti yhtenäisen ilmeen eri assettien välille (Deschamps 2019).

Trim-tekstuurien käyttäminen projekteissa säästää aikaa ja tekstureja, koska yhtä trim-tekstuuria voidaan käyttää uudelleen ja uudelleen. Kuvassa (Kuvio 29) olevat vaunut on kaikki teksturoitu yhtä trim-tekstuuria käyttämällä. Aikaa säästyy, koska aseteille ei tarvitse tehdä erillistä highpoly-mallia tai teksturointitiedostoa. Pelkkä lowpolyn mallinnus ja UV-saarien suoristaminen riittää, koska kaikkien 3D-mallien UV-saaret voidaan asettaa trim-tekstuurin päälle erilaisilla variaatioilla.



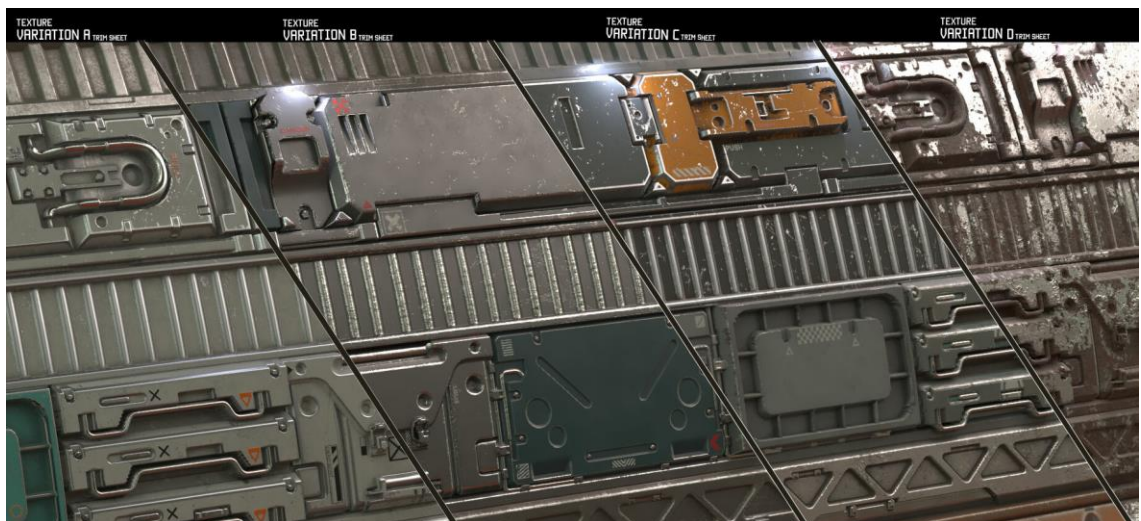
Kuvio 30. Kuva vaunujen teksturoimiseen käytetystä trim-tekstuurista (Deschamps 2019).

Tämä tarkoittaa sitä, että uusia asetteja voidaan teksturoida hyvin nopeasti, jos teksturointitapaa verrataan normaaliin 3D-mallin työnkulkuun. Trim-tekstuurit ovat erittäin hyödyllisiä tilanteissa, joissa pitää nopeasti tuottaa paljon samanhenkisiä asetteja. Trim-tekstuurit myös optimoivat peliprojektia, koska monet assetit voivat käyttää samaa tekstuuria. Trim-tekstuureilla teksturoiduista aseteista voidaan saada vieläkin enemmän

hyötyä irti, jos tekstuureista tehdään setti, jonka trim-liuskojen layout pysyy muuttumattomana. Tämä tekniikka mahdollistaa värien ja materiaalien vaihtamisen lennosta, koska assettien UV-saaret on jo aseteltu osumaan trim-tekstuurin beveleihin.



Kuvio 31. Kuvassa kolme erilaista trim-tekstuuria, joiden trim-liuskojen layout pysyy muuttumattomana. Nämä tekstuurit mahdollistavat assetin materiaalin vaihtamisen tiilen, maalattun metallin ja puun välillä (Olsen 2015).



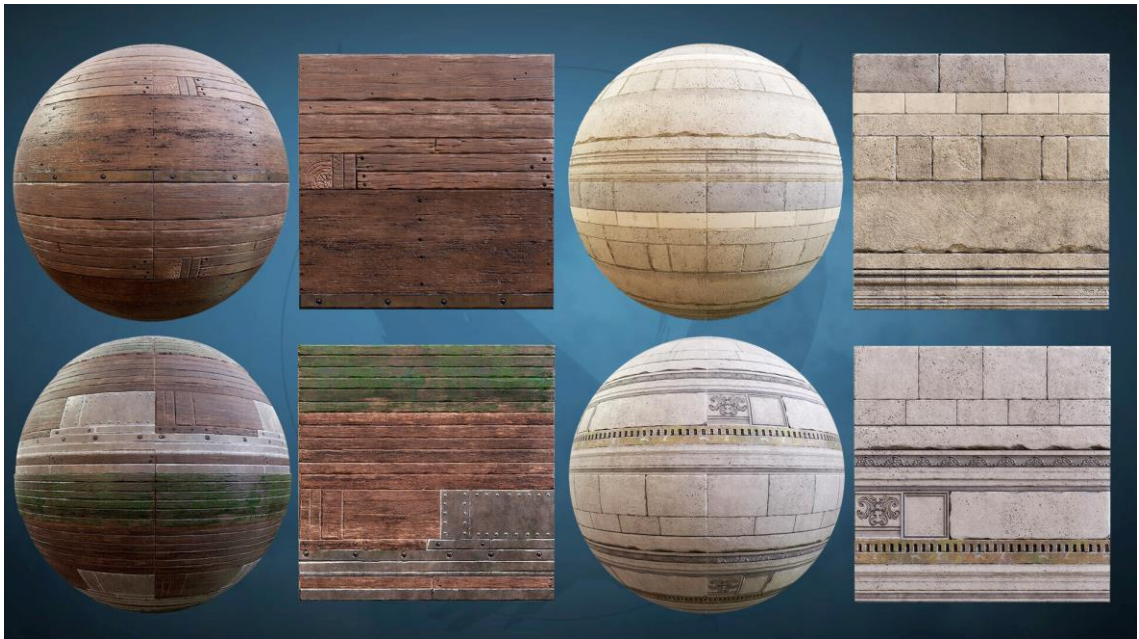
Kuvio 32. Esimerkki trim-tekstuurista, jolla on neljä erilaista tekstuurivariaatiota (Meinl 2019).



Kuvio 33. Kuvassa trim-tekstuurilla teksturoituja asetteja pelistä Sunset Overdrive. Alakerran trim-teksturi on korvattu toisella trim-tekstuurilla, joka muuttaa asetin ulkonäköä nopeasti (Olsen 2015).

## 5 Käytännön sovellutukset ja esimerkit

Olen koonnut tähän lukuun erilaisia esimerkkejä siitä, kuinka trim-tekstuureja on käytetty asettien teksturoimiseen. Pidän osion hyvin kuvapainoisena, koska itse teksturointiprosessi on aina hyvin samanlainen. Esimerkkien löytäminen tähän osioon osoittautui hankalaksi, koska teksturoimiseen käytettyä trim-tekstuuria ei yleensä esitellä erikseen.



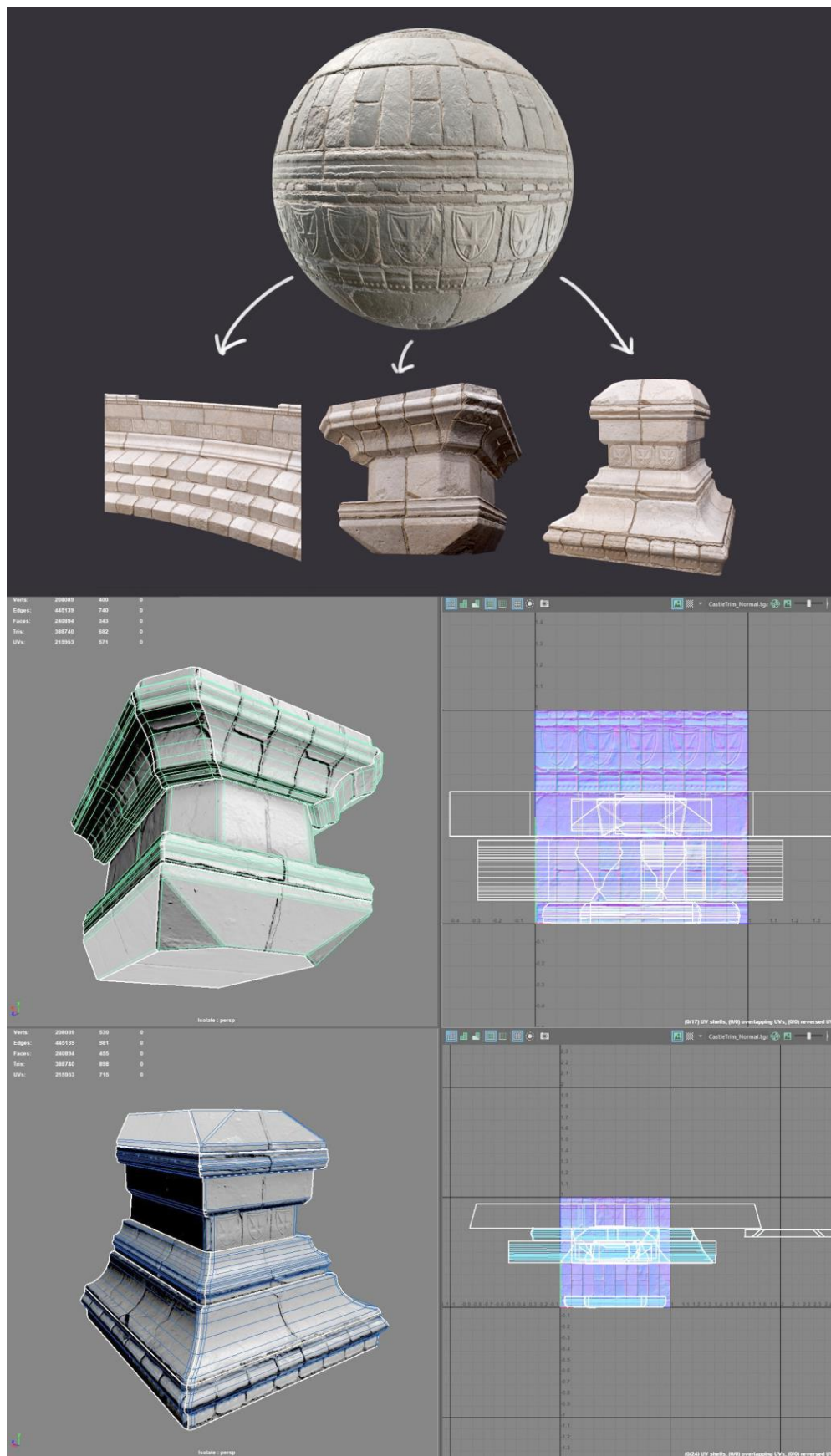
Kuvio 34. Kuvassa erilaisia trim-tekstuureja pelistä Haven (Deschamps 2019).



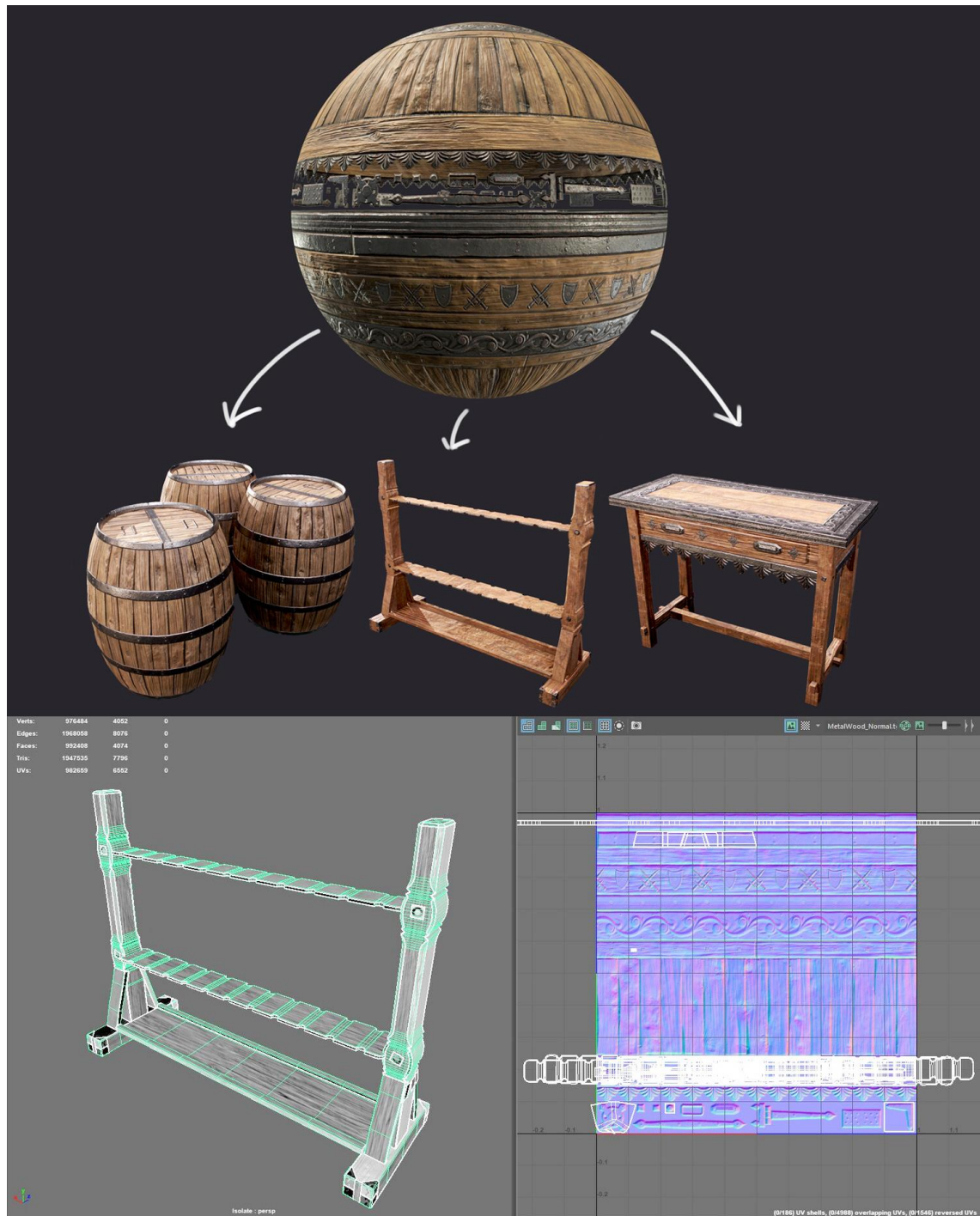
Kuvio 35. Trim-tekstuuri voidaan suunnitella erilaisten tarpeiden mukaan. Kuvan alareunassa on saumaton trim-tekstuuri, jota voidaan käyttää normaaliin tapaan, mutta loppuosaa tekstuurista on tarkoitus käyttää kuten decaleja. Tässä tapauksessa 3D-mallin geometrian pitää olla mallinnettu tavalla, josta voidaan luoda UV-saaria, jotka ovat muodoltaan trim-tekstuurissa olevien muotojen kaltaisia (GameTextures.com n.d.).



Kuvio 36. Kuvissa sama lowpoly-malli, mutta alakuvassa myös 3D-mallin normal map on näkyvissä. Normal map on saatu suoraan trim-tekstuurista, jonka trim-liuskojen välissä olevat bevelit ovat tasan 45 astetta (Olsen 2015).

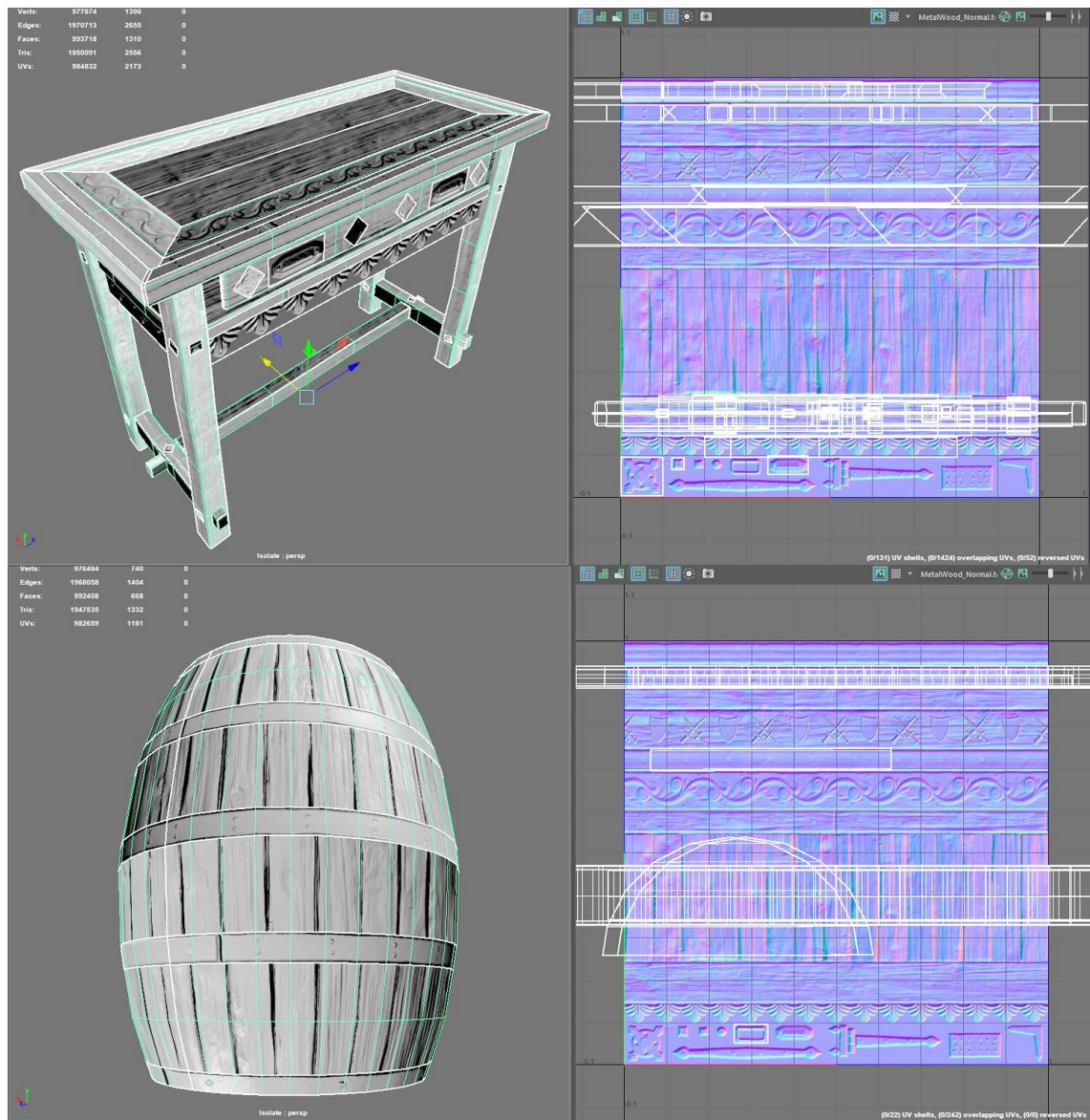


Kuvio 37. Kuvassa kolme yhdellä trim-tekstuurilla teksturoitua asetta ja kahden 3D-mallin UV-saaret saman trim-tekstuurin päällä (van Halteren 2019).



Kuvio 38. Kuvassa kolme yhdellä trim-tekstuurilla teksturoitua asetta ja yhden 3D-mallin UV-saaret trim-tekstuurin päällä (van Halteren 2019).





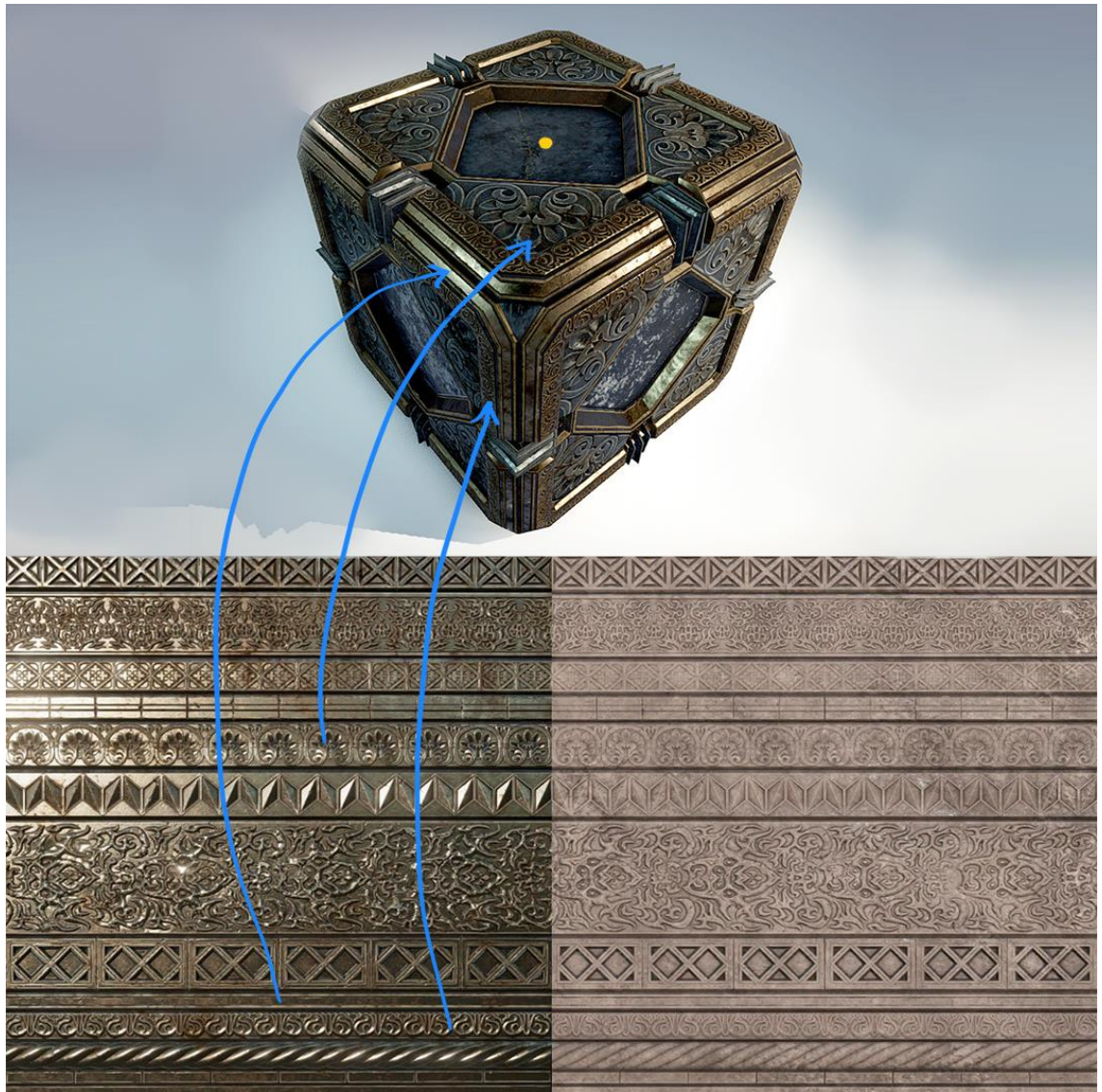
Kuvio 39. Kuvassa kaksi edellisen kuvan trim-tekstuurilla teksturoitua asetta ja 3D-mallien UV-saaret saman trim-tekstuurin päällä (van Halteren 2019).



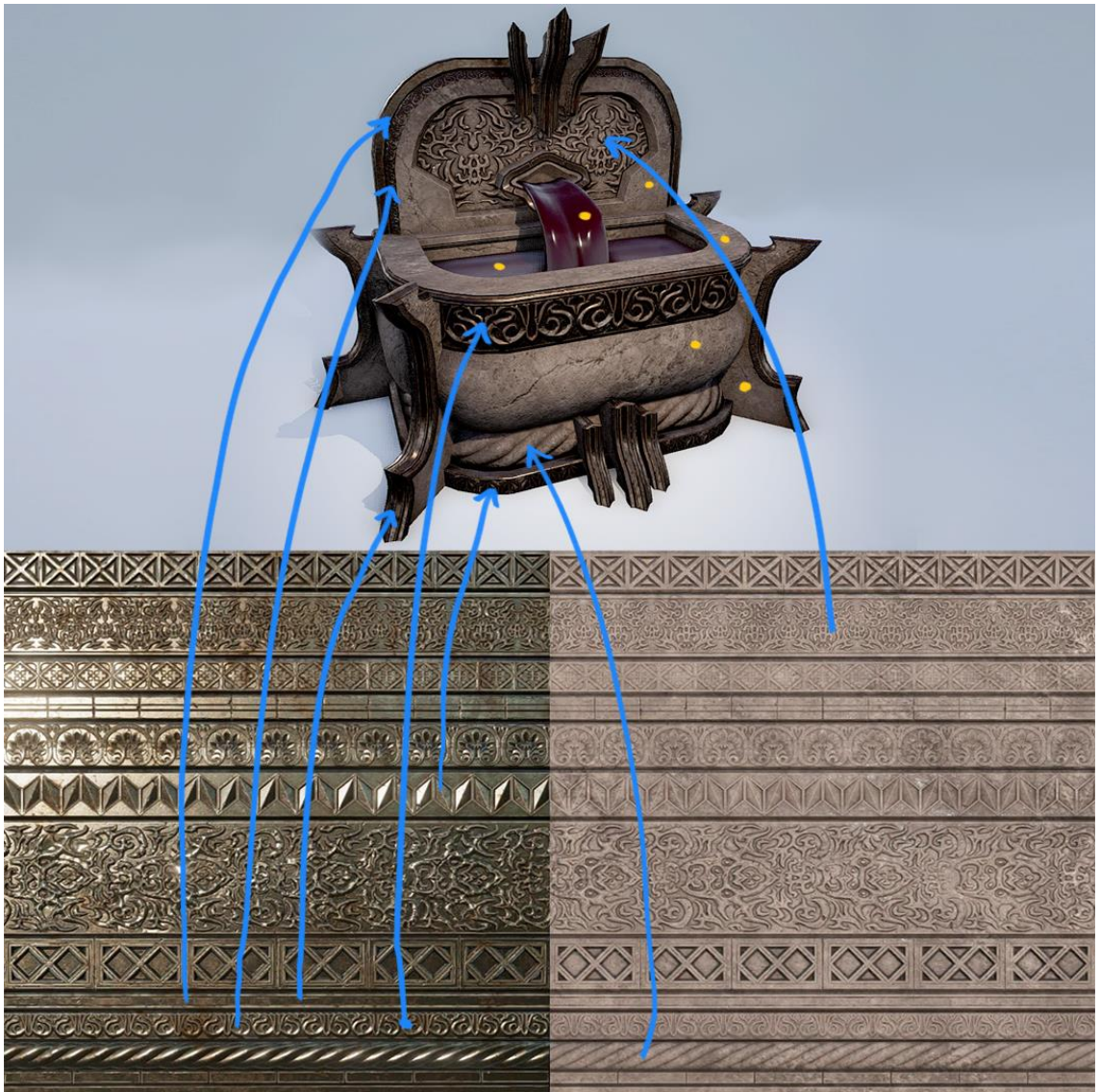
Kuvio 40. Punaisella maalattu alueet on teksturoitu kuvassa olevalla trim-tekstuurilla. Sinisellä maalattu alue toisesta trim-tekstuurista. Maalaamattomat alueet on teksturoitu saumattomalla tekstuurilla (Keene 2019).



Kuvio 41. Kaksi trim-tekstuuria, joilla voidaan vaihtaa 3D-mallin materiaali (Kang 2017).



Kuvio 42. Kuvassa siniset nuolet näyttävät teksturoimiseen käytety trim-liuskat ja oranssi piste merkitsee trim-tekstuurin ulkopuolista saumatonta tekstuuria. Trim-tekstuureissa on tärkeää se, että tekstuurissa on hyvä yhdistelmä yksinkertaisia ja koristeellisia liuskoja, koska teksturoitu lopputulos näyttää visuaalisesti tylsältä, jos tekstuuri on liian yksinkertainen ja visuaalisesti liian äänekkäältä, jos teksturoinnissa on käytetty pelkästään koristeellisia liuskoja (Kang 2017).



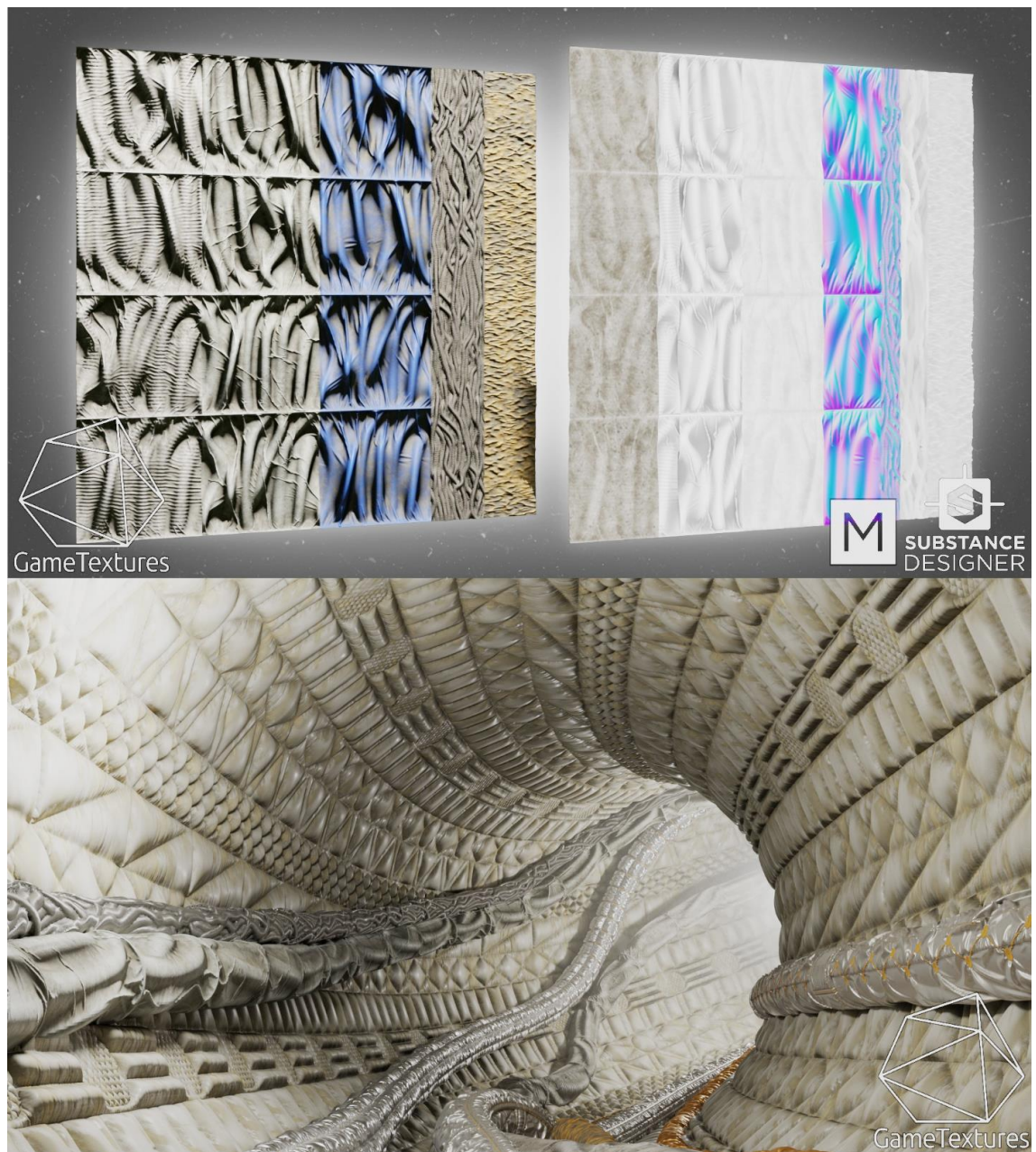
Kuvio 43. Kuvassa siniset nuolet näyttävät teksturoimiseen käytety trim-liuskat ja oranssit pisteet merkitsevät trim-tekstuurin ulkopuolista saumatonta tekstuuria (Kang 2017).



Kuvio 44. Trim-tekstuureilla voidaan teksturoida kätevästi myös erilaisilla kurvityökaluilla luotuja monimutkaisia muotoja, kuten kuvan lonkerokoristeet tai esimerkiksi erilaiset köydet (Keene 2018).



Kuvio 45. Punaisella maalattu alue on teksturoitu trim-tekstuurilla. Trim-teksturi on hyödyllinen suurien toistuvien pintojen teksturoimisessa, koska siihen voidaan lisätä liuskoja, joilla voidaan ratkaista useita eri ongelmia samassa ympäristössä käyttäen vain yhtä tekstuuria. Projektista saa lisätietoa kovalähteen kautta (Roodhuizen 2019).



Kuvio 46. Trim-tekstuurien luomista voidaan lähestyä erilaisilla ohjelmilla. Kuvassa kangasmaisia tekstuureja on valmistettu käyttämällä Marvelous Designeria, joka on tarkoitettu erilaisien kangassimulaatioiden laskemiseen (Keene 2018).





Kuvio 47. Kuvan assetti on teksturoitu pelkästään trim-tekstuureja käyttämällä. Lisää esimerkkejä kuvälähteen kautta (Scheu 2017).

## 6 Trim-tekstuurin suunnittelu

Yleisesti trim-tekstuurin pinta-alan jakamiseen ei ole yhtä oikeaa tapaa. Tilan jakaminen riippuu vahvasti tekijän omista mieltymyksistä ja siitä mitä tekstuurilla halutaan saavuttaa. Tekijän omat mieltymykset alkavat kehittymään sitä mukaan, kun trim-tekstuuritekniikkaa käyttää enemmän. Yhdellä trim-tekstuurilla ei tarvitse yrittää teksturoida kaikkia asetteja. Samassa ympäristössä voi olla vaikka kolmea erilaista trim-tekstuuria, joilla saadaan aikaan enemmän visuaalista variaatiota, mikä mahdollistaa sen, että yksittäisten trim-tekstuurien kanssa voidaan keskittyä ratkaisemaan jotain tiettyä ongelmaa peliympäristössä.

Tekstuureja voidaan yhdistellä useaa UV-settiä käyttämällä. UV-setit yhdistetään shade-reissa ja tekstuurien näkyvyys rajataan maskin avulla. Trim-tekstuurin luomiseen liittyy paljon erilaisia pieniä nippelitietoja, joiden tarkka läpikäyminen ei olisi opinnäytetyön kannalta järkevää, koska sanallisesti selitettynä ne aiheuttavat lukijassa sekaannusta. Siksi suosittelenkin trim-tekstuuritekniikasta kiinnostunutta lukijaa katsomaan [Tim Simpsonin](#) ja [Claudius Dsouzan](#) YouTube-soittolistat trim-tekstuurien luomisesta. Molemmissa lähteissä aihetta käsitellään hieman eri lähtökohdista ja ne antavat mainion pohjan, jonka päälle voi lähteä rakentamaan aiheen syvempää ymmärrystä.

### 6.1 Lähestymistavat

Trim-tekstuurien valmistamista voidaan lähestyä monella eri tavalla. Kaikkein yksinkertaisin vaihtoehto on lähteä rakentamaan tekstuurin highpoly-versiota 3D-mallinnusohjelmassa. Perusmuotojen mallintamisen jälkeen työskentely voidaan viedä loppuun samassa mallinnusohjelmassa tai 3D-malli voidaan siirtää Z-brushiin jatkotyöstettäväksi. Jos highpoly-malli viedään Z-brushiin, kannattaa kaikki trim-liuskat jakaa erillisiksi 3D-malleiksi, koska se tekee liuskojen erittelemisestä polygrouppeihin vaivatonta, minkä jälkeen niihin voidaan helposti veistää erilaisia vaurioita.

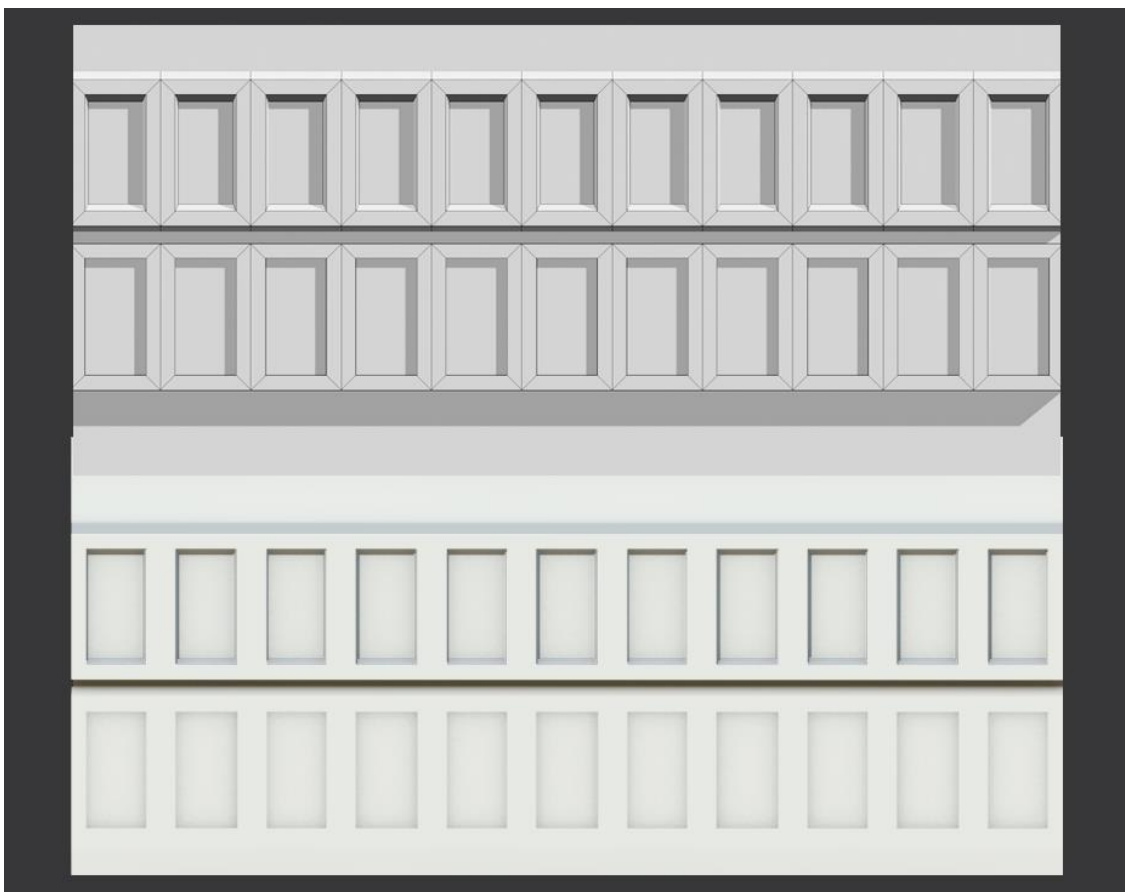
Trim-tekstuuri voidaan myös luoda kokonaan proseduraalisesti esimerkiksi Substance Designerissa ja tietyissä tilanteissa tekstuurin tekoon voidaan hyödyntää erikoisempiakin ohjelmia, kuten esimerkiksi Marvelous Designeria, jota käytetään erilaisten kangassimulaatioiden laskemiseen. Trim-tekstuurin tarpeet auttavat määrittämään mitä ohjelmaa tekstuurin luomiseen kannattaa käyttää, koska jokaisella ohjelmalla on omat vahvuudet.

tensa erilaisten pintojen luomisessa. Mutta pelkästään 3D-mallinnusohjelmissa mallinnetuilla trim-tekstuureilla päästään jo hyvin pitkälle. Tekstuurin tekeminen aloitetaan luomalla polygon plane, jonka koko on neljä kertaa neljä metriä. Tämä polygon plane toimii trim-tekstuurin lowpoly-mallina. Tämän jälkeen polygon planesta tehdään kopio, jota lähetetään jatkotyöstämään. Työstetty 3D-malli toimii trim-tekstuurin highpoly-mallina, jota käytetään normal mapin laskemiseen. Highpoly-mallin muodot kannattaa pitää suhteellisen litteinä, koska mallinnettu highpoly-geometria beikataan kaksiulotteiseen lowpoly-malliin. Normal bakesta tulee helposti oudon näköinen, jos litteällä pinnalla yritetään kuvata syvää tilan muutosta, kuten Tim Simpson on omissa trim-tekstuureja käsittelevissä YouTube-vidoissaan todennut. Simpson on työskennellyt AAA-pelien parissa yli 12 vuotta ja hän on ollut töissä senior 3D-artistina muun muassa Ubisoftilla, Relicillä ja Capcomilla. (Simpson 2018-2020)

Mallinnetun highpoly-mallin geometrian ei tarvitse olla toimivaa, koska se beikataan normal mapiksi. Tämä tarkoittaa sitä, että geometrian palat voivat olla irtonaisia ja mennä vapaasti läpi 3D-mallista, koska highpolyn ainoana kriteerinä on näyttää visuaalisesti hyvältä ja toimivalta, kuten Jeremy Estrellado mainitsee omissa trim-tekstuureja käsittelevissä YouTube-vidoissaan. Tällä hetkellä Estrellado työskentelee senior environment artistina Ubisoft Massivella. (DiNusty Empire 2016-2020)

Mallinnettu ja subdivoitu highpoly-geometria voidaan joutua jäädyttämään, koska subdivoidussa muodossa geometria voi aiheuttaa ongelmia normal map baken kanssa. Jäädyttämisen jälkeen 3D-mallin geometria muuttuu pysyvästi subdivoidun geometrian näköiseksi. Highpoly-malli kannattaa beikata 3D-mallinnusohjelmassa ajan säästämiseksi, kunhan tulokset ovat riittävän hyviä. Normal mapin laskeminen kannattaa suorittaa Substance Painterissa, Marmoset Toolbagissa tai xNormal ohjelmissa, jos 3D-mallinnusohjelmassa saadut tulokset jättävät toivomisen varaa.

Estrellado mainitsee videoissaan, että normal mapin laskemisen kanssa on tärkeää ottaa huomioon se, että 3D-mallissa olevat 90 asteen kulmat eivät välity normal mapin bakessa ortografisen kuvakulman vuoksi (Kuvio 48). Kulmista pitää mallintaa yli 90 asteisia, koska muuten geometrian reunoja ei voida hyödyntää valon tulokulman viestittämiseen katsojalle, koska kulmista ei saada taltioitua tietoa normal mappiin. (DiNusty Empire 2016-2020)



Kuvio 48. Ylhäällä 3D-mallinnusohjelmassa mallinnettu 3D-malli ja alhaalla mallista Substance Painterissa laskettu normal ja ambient occlusion map. Tiukoilla 90 asteen kulmilla mallinnetusta geometriasta ei välity tietoa normal mappiin. Ilman ambient occlusion mappia viimeisellä rivillä oleva korjaamaton versio olisi näkymätön.

Normaalin 3D-mallintamisen lisäksi Substance Designer on hyvin suosittu ohjelma trim-tekstuurien luomiseen. Ohjelman suosio perustuu siihen, että kaikki normal mapin muodot voidaan rakentaa proseduraalisesti ja non-destruktiivisesti erilaisia nodeja käyttämällä. Käytännössä tämä tarkoittaa sitä, että tekstuurista on mahdollista luoda kokonaan uusia variaatioita nodien arvoja muuttamalla ja kaikkea aikaisemmin tehtyä voidaan kiertää ja käyttää uudestaan.

Mainitsin opinnäytetyössäni aikaisemmin Sunset Overdrive pelin, jonka teksturoimiseen kehitettiin Ultimate Trim-tekniikka. Tekniikan innoittamana Justen Lazzaro rakensi Substance Designerissa samalla idealla toimivan Ultimate Trim-generaattorin, joka on ilmaiseksi [ladattavissa](#) ei kaupalliseen käyttöön. Generaattorissa trim-tekstuurin bevelit muodostavat automaattisesti 45 asteen kulmaan ja erilaisia trim-liuska-layoutteja pystytään luomaan vaivatta. Generaattorilla pääsee nopeasti kokeilemaan normal mapilla luotua illuusiota bevelöidyistä reunoista lowpoly-geometrian kanssa, jos lukijalla on aikaisempaa kokemusta Substance Designer ohjelmasta.

## 6.2 UV-setit

3D-malleissa on mahdollista yhdistellä useaa tekstuuria UV-settien avulla. Tekstuureja sekoitetaan shadereita käyttämällä, jossa määritetään tekstuurikoordinaatteja hyödyntämällä mitä UV-mappia pelimoottori katsoo. Useat UV-setit luovat enemmän draw calleja, joita halutaan välttää, mutta esimerkiksi peliympäristöjen kanssa ylimääräisten settien käyttäminen tuo mukanaan mahdollisuuksia, jotka kannattaa ottaa huomioon (Tor-QueMoD 2018).

Useissa aseteissa voidaan käyttää yhtä geneeristä saumatonta tekstuuria, johon voidaan sekoittaa toisella UV-setillä olevaa trim-tekstuuria alpha-maskin avulla, jolla rajataan toisen tekstuurin näkyvyyttä tavalla, mikä mahdollistaa esimerkiksi saumattoman tekstuurin ja trim-tekstuurin normal mappien yhdistämisen. Trim-tekstuurien normal mappia hyödyntämällä voidaan visuaalisesti geneerisiin asetteihin lisätä paljon uusia yksityiskohtia. Linkkinä aiheita selventävä [video](#) usean UV-setin käyttämisestä ja [video](#) siitä kuinka tietoa voidaan soveltaa käytäntöön. (Estrellado 2017)

Yleisesti pelien tekstuurien sekoittamisesta kiinnostuneille lukijoille vielä linkki [Benjamin Lindquistin](#) opinnäytetyöhön. Lindquist työskentelee tällä hetkellä Remedy Entertainmentilla senior environment artistina.

## 6.3 Teksturoinnista

Erityisesti koristeellisten liuskojen saumattomuus voi aiheuttaa tekijälle päänvaivaa, koska toistuva kuvio pitää saada mahtumaan trim-liuskalle tietyssä koossa. Jos koristeellinen geometria on luotu 3D-mallinnusohjelmassa, voidaan käyttää hyväksi tekniikka, missä trim-tekstuurin reunalle asetellaan geometriaa, joka peittää reunan näkyvyyden. Tämän jälkeen geometriasta tehdään kopio ja trim-tekstuurin yläkulman verteksiä käytetään pivottina, jonka avulla geometria snäpätään vastakkaiseen yläkulmaan. Trim-tekstuurin reunat ovat tämän jälkeen saumattomia ja trim-lisukan keskelle jäävä tyhjä tila voidaan täyttää vapaasti. Linkkinä prosessia selventävä [video](#). (Estrellado 2017)

Z-brushissa saumattomuuden saavuttamiseksi pitää käyttää warp modifieria. Tulokset eivät kuitenkaan aina ole optimaalisia ja mappoja voidaan joutua korjaamaan Photoshopissa, jossa saumattomuus saavutetaan käyttämällä offset työkalua, joka löytyy filter ja other valikosta.

On hyvin todennäköistä, että max frontal distance -arvoa joudutaan nostamaan, kun normal map beikataan Substance Painterissa lowpoly-malliin. Tämä johtuu siitä, että ohjelman lähettämät säteet eivät yllä taltioimaan kaikkea tietoa paljon paksummasta highpoly-mallista, koska lowpoly-mallina on kaksikulotteinen polygon plane. (Simpson 2019)

Korkearesoluutioinen normal map ja maskit luovat illuusion siitä, että myös tekstuurin muut mapit olisivat korkearesoluutioisia. Usean yksittäisen tekstuuri-mapin resoluutiota kannattaakin kokeilla laskea matalammaksi ja katsoa säilyykö illuusio korkeammasta resoluutiosta pelimoottorin puolella, vaikka osassa mappeja käytetäänkin pienempi resoluutioisia tekstuureja. Projektin optimoinnin kannalta on parempi mitä pienempiä tekstuurikarttoja pystytään käyttämään, koska ne kuluttavat vähemmän muistia. (Estrellado 2017)

Trim-tekstuurit jättävät jälkeensä paljon UV-saumoja. Niitä syntyy, kun 3D-mallin geometriaa kääritään auki ja UV-saaret pakotetaan suorakaiteen muotoon. Ongelma on siinä, että trim-liuska jolle UV-saaret asetellaan, on kyllä saumaton, mutta UV-saaret eivät jatku tekstuurin reunasta reunaan, vaan ne jäävät kesken, jolloin UV-saumot syntyvät suorakaiteen muotoisten UV-saarien päihin. Tämän vuoksi trim-teksturiin on hyödyllistä lisätä visuaalisia saumoja esimerkiksi lautojen tai kivien väliin, koska UV-saumot voidaan asetella osumaan trim-tekstuurissa oleviin saumoihin, mikä tekee UV-saumoista vähemmän huomiota herättäviä 3D-mallin pinnalla.



Kuvio 49. Kuvan trim-tekstureissa rytmitystä luodaan lisäämällä toistuvia elementtejä kuten saumoja ja pultteja. Kuvälähteenä kiinnostava projekti, joka kannattaa vilkaista läpi (Norris 2014).

Jos tehdään useamman trim-tekstuurin settejä, missä tarkoituksena on tekstuuria vaihtamalla saada 3D-mallille mahdollisimman erilainen visuaalinen ilme, kannattaa trim-liuskojen rytmitykseen kiinnittää erityistä huomiota, koska sitä muuttamalla voidaan saavuttaa enemmän visuaalista variaatiota vaihdettavien tekstuurien välille (Kuvio 49).

Kerron lopuksi Tim Simpsonin antamia hyödyllisiä huomioita, jotka liittyvät trim-tekstuurien teksturointiin. On hyvin käytännöllistä, että trim-tekstuuuri sisältää yhden pienen ja yhden suuremman trim-liuskan, jotka ovat visuaalisesti hyvin geneerisiä, koska ne eivät kiinnitä katsojan huomiota itseensä, jonka vuoksi niitä voidaan toistaa paljon. (Simpson 2018-2020)

Ympyrän muotoisten alueiden keskikohdat aiheuttavat haasteita trim-tekstuureille, koska tekstuurit venyvät voimakkaasti keskikohtaa kohti. Ongelma voidaan kiertää, jos trim-tekstuurissa on iso tasainen alue, jonka päälle ympyrän muotoisen geometrian keskikohdasta irrotettu UV-saari voidaan asettaa. (Simpson 2018-2020)

Tämä ei kuitenkaan ole optimaalista, koska trim-tekstuureissa ei ole tilaa tuhlatavaksi. Vaihtoehtoisesti keskikohdan päälle voidaan mallintaa uutta geometriaa, jonka teksturoimiseen käytetään jotain toista tekstuuria.

Tekstuurin värit kannattaa pitää hyvin lähellä keskisävyjä, vaikka esimerkiksi vaaleiden korostusvärien lisääminen teräviin reunoihin tuntuisikin hyvin houkuttelevalta, koska teksturoinnilla tehdyt vaaleat reunat heikentävät valon vaihteluiden havaitsemista 3D-mallin pinnalla pelimoottorin puolella. (Simpson 2018-2020)

## **7 Yhteenveto ja pohdinta**

Opinnäytetyön tarkoituksena on ollut tiivistää trim-tekstuurit ja niiden yleiset käyttötavat helppolukuisiksi ja mahdollisimman tiiviiksi paketuksi, joka herättäisi lukijan kiinnostuksen tekniikkaa kohtaan ja motivoisi kokeilemaan sitä omassa työnkulussa. Mielestäni aiheen rajaus onnistui hyvin ja tärkeimmät tiedot tekniikasta ja sen käyttöympäristöstä tulevat työssä selvästi ilmi. Trim-tekstuurit ovat kiinnostaneet minua siitä hetkestä asti, kun näin tekniikan ensimmäisen kerran toiminnassa ja opinnäytetyö tarjosi loistavan tilaisuuden keskittyä aiheeseen syvemmin. Toivon että opinnäytetyö avautuu kuvien avulla mahdollisimman monelle lukijalle, koska olen huomannut, että trim-tekstuurien idean selittäminen suullisesti on hyvin hankalaa, jos kuulijalla ei ole vahvaa aikaisempaa

kokemusta tekstuurien optimoisesta. Tekniikan oppiminen on mielestäni hyvin tärkeää, koska sen avulla voidaan lähestyä projektien erilaisia mallintamiseen ja teksturointiin liittyviä haasteita erilaisesta kulmasta, mikä saa oman 3D-osaamisen tuntumaan tasapainoisemmalta ja valmiilta kohtaamaan uusia haasteita.

Opinnäytetyön tekeminen selkeytti trim-tekstuureihin liittyviä teknisiä asioita ja täytti puuttuvat palat trim-tekstuurien käyttötavoissa. Erilaisten ympäristöjen toteuttaminen tulee olemaan jatkossa paljon sujuvampaa trim-tekstuurien avulla, koska en enää ole lukittuna toistuviin saumattomiin tekstuureihin ja uniikkeihin asetteihin. Koen että opinnäytetyö on hyvin ajankohtainen, koska trim-tekstuurit ovat vasta nousussa oleva trendi, jonka voi huomata esimerkiksi siitä, että hyviä lähteitä aiheesta on edelleen suhteellisen niukasti löydettävissä ja trim-tekstuureista mainitseminen aiheuttaa kuulijassa yleensä reaktion pelkkää ihmetystä.

Uskon että trim-tekstuurit nousevat yleiseen tietoisuuteen muutamassa vuodessa, jos 3D-alalla ei tapahdu isoja merkittäviä muutoksia normal mappien käytön suhteen. Yksi hyvä esimerkki mahdollisista isoista tulevista muutoksista on Unreal Engine 5. On hyvin todennäköistä, että monet polygon määriin ja tekstuureihin liittyvät rajoitteet saattavat poistua lähes kokonaan, jos pelimoottorin kehittäjät pystyvät pitämään lupauksensa. Lupauksen toteutuessa koko reaaliaikaisen 3D-grafiikan scene tulee muuttumaan ja uusia tehokkaampia työskentelytapoja ilmaantuu, mutta lupaukset eivät aina käy toteen ja ainakin vielä toistaiseksi trim-tekstuurit ovat erittäin tehokas tapa teksturoida ja optimoida 3D-malleja.



## Lähteet

Albeluhn, Christopher n.d. Adding variation to a repeating texture pattern. [Verkkodokumentti] <<https://www.chrisalbeluhn.com/ut3-adding-variation-to-a-repeating-texture-pattern>> (luettu 26.5.2020)

Bernstein, Mike 2017. Modularity In Creating 3D Game Environments. [Opinnäytetyö] <<http://urn.fi/URN:NBN:fi:amk-201805168579>> (luettu 26.5.2020)

Bethesda Game Studios 2016. Fallout 4's Modular Level Design. [Youtube] <<https://www.youtube.com/watch?v=QBAM27YbKZg>> (katsottu 26.5.)

Burgess, Joel 2013. Skyrim's Modular Level Design - GDC 2013 Transcript. [Verkkodokumentti] <<http://blog.joelburgess.com/2013/04/skyrims-modular-level-design-gdc-2013.html>> (luettu 26.5.2020)

Burgess, Joel 2019. Building Huge Open Worlds: Modularity, Kits & Art Fatigue. [Verkkodokumentti] <<https://80.lv/articles/building-huge-open-worlds-modularity-kits-art-fatigue/>> (luettu 25.5.2020)

DiNusty Empire 2016-2020. [Youtube-kanava 2016-2020] <<https://www.youtube.com/channel/UCKfTBvcsO9Kw0b5EVuf4CJQ/videos>> (katsottu 8.5.2020)

Dsouza, Claudius 2019. 01- Planning a Trim Texture Explained. [Youtube] <<https://www.youtube.com/watch?v=2gpBPKRDLJs#t=0m22s>> (katsottu 26.5.2020)

Dsouza, Claudius 2019. 09- Using Trims In Maya Explained. [Youtube] <<https://www.youtube.com/watch?v=66yvJMCMfTA#t=19m33s>> (katsottu 26.5.2020)

Dygert, Mark 2019. [Polycount-foorumi] <<https://polycount.com/discussion/212378/trim-sheets-how-to-use-them>> (luettu 25.5.2020)

Estrellado, Jeremy 2017. Building A Trim Sheet Pt.6. [Youtube] <<https://www.youtube.com/watch?v=ee8Bivel7dA>> (katsottu 27.5.2020)

Game Dev Academy 2015. What is a Shader? | Pixel and Vertex Shaders. [Youtube] <<https://www.youtube.com/watch?v=TDZMSozKZ20>> (katsottu 26.5.2020)

Get Learnt 2019. What is a Normal Map? [Youtube] <<https://www.youtube.com/watch?v=jK11WjGuJal>> (katsottu 26.5.2020)

malcolm341 2018. What is Texel Density and How to Master it. [Youtube] <<https://www.youtube.com/watch?v=5e6zvJqVqlA>> (katsottu 26.5.2020)

Overlord Gaming 2018. What is Baked Lighting? (Lightmap). [Youtube] <<https://www.youtube.com/watch?v=eGmXDOGjKPA>> (luettu 26.5.2020)

Palmi, Daniel 2013. Mobiilipeligrifiikan optimointi : Case: Trials Frontier. [Opinnäytetyö] <<http://urn.fi/URN:NBN:fi:amk-2013101116034>> (luettu 25.5.2020)

Perko, Craig 2017. Level Design: Modular Levels in Nier. [Youtube] <<https://www.youtube.com/watch?v=JKn7u09mR8M>> (katsottu 26.5.2020)

Shadercat 2016. What is a Shader? [Verkkodokumentti] <<https://www.shader-cat.com/what-is-a-shader/>> (luettu 26.5.2020)

Simpson, Tim 2018-2020. [Youtube-kanava 2018-2020] <[https://www.youtube.com/channel/UCGXr6E\\_g91ue1rfhA9j4TLA/videos](https://www.youtube.com/channel/UCGXr6E_g91ue1rfhA9j4TLA/videos)> (katsottu 25.5.2020)

Simpson, Tim 2019. Quick and Easy Trim Sheet UV Mapping Tips. [Youtube] <<https://www.youtube.com/watch?v=VEHsZniXquY>> (katsottu 25.5.2020)

Simpson, Tim 2019. Trim Sheets in Substance Painter: Normal Map Baking + Smart Materials - Trim Texture Tutorial Part 3. [Youtube] <<https://www.youtube.com/watch?v=CarefswACgs>> (katsottu 27.5.2020)

Simpson, Tim 2019. Trim Texture Tutorial - Environment Art 101. [Youtube] <<https://www.youtube.com/watch?v=lzilY674NAw>> (katsottu 25.5.2020)

TorQueMoD 2018. [Unreal Engine -foorumi] <[forums.unrealengine.com/development-discussion/rendering/1505806-how-many-uv-sets-for-a-character](https://forums.unrealengine.com/development-discussion/rendering/1505806-how-many-uv-sets-for-a-character)> (luettu 27.5.2020)

## Kuvalähteet

Kuvio 1. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 2. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/oOKPkw>

Kuvio 3. Irrational Games 2013

<https://www.youtube.com/watch?v=tWrelx5goh0&t=16m20s>

Kuvio 4. TextureMax 2015

<https://www.unrealengine.com/marketplace/en-US/product/200-grunge-decals>

Kuvio 5. Game Dev Academy 2015

<https://www.youtube.com/watch?v=TDZMSozKZ20#t=0m24s>

Kuvio 6. Albeluhn, Christopher n.d.

<https://www.chrisalbeluhn.com/ut3-adding-variation-to-a-repeating-texture-pattern>

Kuvio 7. Robinson-Yu, Adam 2020

<https://www.youtube.com/watch?v=ZW8gWgpptl8#t=12m17s>

Kuvio 8. Bethesda Game Studios 2016

<https://www.youtube.com/watch?v=QBAM27YbKZg#t=5m23s>

Kuvio 9. Bethesda Game Studios 2016

<https://www.youtube.com/watch?v=QBAM27YbKZg#t=8m58s>

Kuvio 10. Platinum Games 2017

<https://www.youtube.com/watch?v=JKn7u09mR8M#t=7m12s>

Kuvio 11. Bethesda Game Studios 2016

<https://www.youtube.com/watch?v=QBAM27YbKZg#t=24m17s>

Kuvio 12. malcolm341 2018

<https://www.youtube.com/watch?v=5e6zvJqVqlA#t=1m27s>

Kuvio 13. Blender Guru 2020

<https://www.youtube.com/watch?v=-VgtSL5ZpYc#t=2m26s>

Kuvio 14. Overlord Gaming 2018

<https://www.youtube.com/watch?v=eGmXDOGjKPA#t=0m10s>

Kuvio 15. Piper, Richard 2016

<https://www.artstation.com/artwork/vZ9I6>

Kuvio 15. TextureMax 2015

<https://www.unrealengine.com/marketplace/en-US/product/200-grunge-decals>

Kuvio 15. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 16. Olsen, Morten 2015

<https://www.gdcvault.com/play/1022324/The-Ultimate-Trim-Texturing-Techniques>

Kuvio 17. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 18. GameTextures.com 2019

[https://www.youtube.com/watch?v=qODD-79i5\\_4#t=38m49s](https://www.youtube.com/watch?v=qODD-79i5_4#t=38m49s)

Kuvio 19. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 20. McNally, Christopher 2020

<https://spark.adobe.com/page/dLqfx8R4vgj8u/>

Kuvio 21. Dsouza, Claudius 2019

<https://www.youtube.com/watch?v=66yvJMCMfTA#t=19m33s>

Kuvio 22. McNally, Christopher 2020

<https://spark.adobe.com/page/dLqfx8R4vgj8u/>

Kuvio 25. Olsen, Morten 2015

<https://www.gdcvault.com/play/1022324/The-Ultimate-Trim-Texturing-Techniques>

Kuvio 29. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 30. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 31. Olsen, Morten 2015

<https://www.gdcvault.com/play/1022324/The-Ultimate-Trim-Texturing-Techniques>

Kuvio 32. Meini, Frank 2019

<https://www.artstation.com/artwork/4bBv61>

Kuvio 33. Olsen, Morten 2015

<https://www.gdcvault.com/play/1022324/The-Ultimate-Trim-Texturing-Techniques>

Kuvio 34. Deschamps, Sophie 2019

<https://www.artstation.com/artwork/xzrXqW>

Kuvio 35. GameTextures.com n.d.

<https://gametextures.com/shop/706683>

Kuvio 36. Olsen, Morten 2015

<https://www.gdcvault.com/play/1022324/The-Ultimate-Trim-Texturing-Techniques>

Kuvio 37. van Halteren, Frank 2019

<https://www.exp-points.com/king-arthur-grandiose-kings-chamber>

Kuvio 38. van Halteren, Frank 2019

<https://www.exp-points.com/king-arthur-grandiose-kings-chamber>

Kuvio 39. van Halteren, Frank 2019

<https://www.exp-points.com/king-arthur-grandiose-kings-chamber>

Kuvio 40. Keene, Keegan 2019

<https://www.artstation.com/artwork/nQIQR4>

Kuvio 41. Kang, Weidi 2017

<https://www.artstation.com/artwork/Redmr>

Kuvio 42. Kang, Weidi 2017

<https://www.artstation.com/artwork/Redmr>

Kuvio 43. Kang, Weidi 2017

<https://www.artstation.com/artwork/Redmr>

Kuvio 44. Keene, Keegan 2018

<https://www.artstation.com/artwork/gA9gRD>

Kuvio 45. Roodhuizen, Math 2019

[https://mathsgame.art/projects/gJ04L8?album\\_id=77490](https://mathsgame.art/projects/gJ04L8?album_id=77490)

Kuvio 46. Keene, Keegan 2018

<https://www.artstation.com/artwork/N5wADD>

Kuvio 47. Scheu, Ethan 2017

<https://www.artstation.com/artwork/b4mZE>

Kuvio 49. Norris, Jacob 2014

<https://polycount.com/discussion/144838/ue4-modular-building-set-breakdown/p1>

### **Tekstiin upotetut linkit**

<https://www.youtube.com/watch?v=QBAM27YbKZg>

<https://polycount.com/discussion/141061/polycounts-in-next-gen-games-thread>

<https://www.youtube.com/watch?v=VEHsZniXquY>

[https://www.youtube.com/watch?v=DipfrjCgYW8&list=PLBi3xvwwY3dk09fNSd7jrnEs-cXS\\_6\\_1p2](https://www.youtube.com/watch?v=DipfrjCgYW8&list=PLBi3xvwwY3dk09fNSd7jrnEs-cXS_6_1p2)

[https://www.youtube.com/watch?v=2gpBpkRDLJs&list=PLU74dDTYexO5ap\\_Ko9stC-RZs5Pw1cJOi&index=2#t=0m22s](https://www.youtube.com/watch?v=2gpBpkRDLJs&list=PLU74dDTYexO5ap_Ko9stC-RZs5Pw1cJOi&index=2#t=0m22s)

<https://gumroad.com/l/ByFF>

<https://www.youtube.com/watch?v=6g5YKzKxX3U>

<https://www.youtube.com/watch?v=ee8Bivel7dA#t=1h30m10s>

<http://urn.fi/URN:NBN:fi:amk-2015052510183>

<https://www.youtube.com/watch?v=ee8Bivel7dA#t=20m35s>