



Proseduraalisen 3D-mallintamisen työnkulun kehittyminen ja sen vaikutus indie-pelikehitykseen

Eetu Pitkänen

Opinnäytetyö, AMK

Toukokuu 2023

Tradenomi

Tietojenkäsittelyn tutkinto-ohjelma

Pitkänen, Eetu

Proseduraalisen 3D-mallintamisen työnkulun kehittyminen ja sen vaikutus indie-pelikehitykseen

Jyväskylä: Jyväskylän ammattikorkeakoulu. **Toukokuu 2023**, 54 sivua

Tradenomi. Tietojenkäsittelyn tutkinto-ohjelma. Opinnäytetyö AMK.

Julkaisun kieli: suomi

Julkaisulupa avoimessa verkossa: kyllä

Tiivistelmä

Itsenäisillä eli indie-pelirytyksillä on usein rahoittajien puuttuessa rajalliset resurssit, minkä takia niiden on vaikeampaa tuottaa suuren mittakaavan pelejä. Budjetin ollessa pieni on myös saatavilla oleva työvoima suppeampi. Proseduraalinen 3D-mallintaminen pystyy vähentämään graafikon työtaakkaa automatisoimalla toistuvia työtehtäviä, esimerkiksi rakennuksia mallintaessa.

Tavoitteena oli lähteä selvittämään, millaisia proseduraalisen 3D-mallintamisen työnkulkua on ja miten ne ovat kehittyneet, sekä miten tämä istuu indie-pelikehityksen viitekehitykseen. Tutkimusta toteutettiin keräämällä tietoa aiemmista tutkimuksista sekä pitämällä teema- ja kyselyhaastatteluja pelialalla ja 3D-grafiikan parissa toimiville henkilöille. Kerättyä aineistoa analysoitiin ja peilattiin tutkittaviin aiheisiin.

Tutkimus osoitti proseduraaliselle mallintamiselle olevan useita erilaisia työnkulkua, jotka ovat kehittyneet vuosikymmenien aikaan huomattavasti. Tekoälyn kehittyminen koettiin tärkeimpänä tekijänä proseduraalisen mallintamisen tulevan kehittymisen osalta. Proseduraalisen mallintamisen todettiin vaikuttavan indie-pelikehitykseen pääasiassa positiivisesti. Sen havainnointiin mahdollistavan laajojen kokonaisuuksien, kuten maastojen ja kaupunkien, nopean ja helpon mallintamisen. Proseduraalinen 3D-mallintaminen ei kuitenkaan sovi kaikkiin indie-projekteihin ja se toimii ainoastaan avustavana työkaluna tiettyihin aikaa vieviin työtehtäviin.

Avainsanat (asiasanat)

proseduraalisuus, 3D-mallinnus, pelinkehitys, pelisuunnittelu, tekoäly, koneoppiminen, haastattelututkimus

Muut tiedot (salassa pidettävät liitteet)

Pitkänen, Eetu

The advancement of procedural 3D modeling workflow and its impact on indie game development

Jyväskylä: JAMK University of Applied Sciences, May 2023, 54 pages

Business Administration. Bachelor's Degree Programme in Business Information Technology. Bachelor's thesis

Permission for open access publication: Yes

Language of publication: Finnish

Abstract

Independent or indie game companies often have limited resources in the absence of investors, which makes it more difficult for them to produce large-scale games. When the budget is small, the available workforce is also more constricted. Procedural 3D modeling can reduce the graphic designer's workload by automating repetitive tasks, for example when modeling buildings.

The goal was to find out what procedural 3D modeling workflows are like and how they have developed, as well as how this fits into the framework of indie game development. The research was carried out by gathering information from previous studies and interviewing and surveying people working in the game industry and 3D graphics. The collected material was analyzed and reflected to the researched topics.

The research showed that there are several different workflows for procedural modeling, which have developed considerably over the decades. The development of artificial intelligence was perceived as the most important factor in the future development of procedural modeling. Procedural modeling was found to have a mainly positive effect on indie game development. It was found to enable fast and easy modeling of large entities such as terrains and cities. However, procedural 3D modeling is not suitable for all indie projects, and it only works as an auxiliary tool for certain time-consuming tasks.

Keywords/tags (subjects)

procedurality, Three-dimensional imaging, game design, artificial intelligence, machine learning, interview study

Miscellaneous (Confidential information)

Sisältö

1	Johdanto	3
1.1	Indie-pelikehitys ja proseduraalinen mallinnus	3
2	Tutkimusasetelma	4
2.1	Opinnäytetyön tavoitteet ja tutkimusongelma	4
2.2	Opinnäytetyön merkitys.....	5
2.3	Tutkimusmenetelmät.....	5
2.4	Aineistonkeruu ja -analyysi	6
2.4.1	Tiedonhaku	7
2.4.2	Teemahaastattelu.....	8
2.4.3	Kyselylomake	8
2.4.4	Analysointi	9
2.5	Luotettavuuden varmistaminen.....	10
3	Indie-pelikehitys	11
4	Proseduraalinen 3D-mallinnus	11
4.1	Taustaa	12
4.2	Hyödyt	13
5	Fraktaali ja L-systeemit	13
5.1	L-systeemit proseduraalisessa mallintamisessa	14
5.2	Parametrinen L-systeemi	15
5.3	Stokastinen L-systeemi.....	15
6	Muotokieliopit.....	16
6.1	CGA-muotokielen käyttö rakennusten mallintamisessa.....	17
6.2	Rakennusten mallintaminen kuvasyötteestä.....	18
6.3	Valintalauseilla toimiva SELEX-kieli	19
7	Kohinat.....	20
7.1	Perlin-kohina	20
7.2	Worley-kohina.....	21
8	Generatiiviset mallit ja koneoppiminen.....	22
8.1	Generatiivinen kilpaileva verkosto.....	22
8.1.1	GAN:n hyödyntäminen realistisemmän maaston luomisessa.....	23
8.1.2	2D-kuvilla tehostettu 3D-malligeneraattori	25
8.2	Autoenkooderi neuroverkosto.....	25
8.2.1	Autoenkooderin hyödyntäminen proseduraalisessa mallintamisessa.....	26

9	Aineiston analyysi.....	27
9.1	Teemahaastattelun tausta ja tulokset	27
9.1.1	Haastateltavan suhde aiheeseen.....	28
9.1.2	Indie-pelikehitys.....	28
9.1.3	Proseduraalinen 3D-mallintaminen.....	30
9.1.4	Proseduraalisen 3D-mallintamisen suhde pelikehitykseen.....	31
9.2	Kyselyhaastattelujen tausta ja tulokset	32
9.2.1	Indie-pelikehityksen rajoitteet, vapaudet ja kehittyminen	33
9.2.2	Proseduraalinen 3D-mallintaminen.....	34
9.2.3	Proseduraalisen mallintamisen vaikutus pelialaan	34
10	Johtopäätökset.....	35
11	Pohdinta.....	39
11.1	Ajatuksia tutkimuksen onnistumisesta	39
11.2	Eettisyys.....	41
11.3	Luotettavuus	42
11.4	Jatkotutkimus	42
	Lähteet	44
	Liitteet	48
	Liite 1. Haastattelurunko.....	48
	Liite 2. Kyselylomake	50
	Kuviot	
	Kuvio 1. Pythagoraan puu	14
	Kuvio 2. Edessä stokastisesti luodut rakennukset ja takana kolme deterministisesti luotua rakennusta (Hidalgo ym. 2008, 7).	16
	Kuvio 3. CGA-kielioppi vaatii paljon jakosääntöjä sopivan alueen muodostamiseen toisin kuin SELEX, jossa julkisivun ruudukosta valitaan haluttu alue kullekin elementille.....	20
	Kuvio 4. Blenderillä generoitua Perlin-kohinaa.	21
	Kuvio 5. Blenderillä generoitua Worley-kohinaa.	21
	Kuvio 6. Generatiivisen kilpailevan verkoston toimintaperiaate.....	23
	Kuvio 7. Autoenkooderi neuroverkoston toimintaperiaate.	26
	Kuvio 8. Proseduraalisen mallintamisen hyötysuhde tarvittavan sisällön määrään.	38

1 Johdanto

Useat oikean elämän rakenteista ovat monimutkaisia ja suuria, olivatpa ne sitten luonnollisia, kuten galaksit ja rantaviivat, tai ihmisten tekemiä rakennelmia, kuten kaupungit. Monimutkaisuutta löytyy myös pienemmästä mittakaavasta, kuten kasveista tai bakteerien kasvumalleista. Kun näitä asioita yritetään luoda virtuaalisesti, esimerkiksi videopeleihin, törmätään suorituskyvylisiin ongelmiin. Näiden rakenteiden mallintaminen manuaalisesti on aikaa vievää, työlästä ja kallista. Onneksi monien tällaisten objektien rakenteet ovat itsesimilaarisia eli ne syntyvät säännöllisistä iteraatioista ja toistuvista muodoista, joten niitä pystyy mallintamaan nopeammin ja helpommin proseduraalisesti.

Videopelien mittakaavojen suurentuessa vuosi vuodelta niihin tarvittava työmäärä kasvaa myös koko ajan suuremmaksi (Togelius, Shaker & Nelson 2016, 3). Togeliuksen ja muiden (mts. 3) mukaan on hyvin yleistä, että menestyvän pelin kehittämiseen tarvitaan vuosia aikaa ja satoja työnteekijöitä, joista moni työskentelee graafisella puolella. Manuaalisesti mallintamalla graafikko joutuu tekemään paljon toistuvia rutiinitehtäviä, jotka syövät pois aikaa luovilta ratkaisuilta. Proseduraalinen mallinnus ei siis poista tarvetta graafiselle suunnittelijalle vaan toimii avustavana työkaluna näissä tehtävissä. (Mts. 3.)

1.1 Indie-pelikehitys ja proseduraalinen mallinnus

Tämä opinnäytetyö keskittyy etenkin indie-pelikehittäjien eli itsenäisten (*engl. independent*) pelikehittäjien näkökulmaan. Tämän kaltaisilla peliyrityksillä tai työryhmillä on harvoin resursseja toteuttaa suuren mittakaavan pelejä vähäisen työvoiman ja budjetin vuoksi (Blatz & Korn 2017, 2; (Togelius ym. 2016, 3). Indie-peliryityksillä on sen sijaan etulyöntiasema luovissa ratkaisuissa, koska ne ovat riippumattomia ylemmän tahon vaikutusvallasta toisin kuin isot peliryitykset. AAA-pelien eli suuren luokan pelien tekeminen on kallista, joten ainoastaan suuret pelifirmat voivat kehittää niitä. (Togelius ym. 2016, 3.) Nämä yritykset eivät yleensä halua ottaa riskejä tekemällä pelattavuuteen liittyviä luovia ratkaisuja, koska sellainen peli saattaisi epäonnistua markkinoilla ja aiheuttaa yritykselle vakavia taloudellisia vaikeuksia. Tästä johtuen AAA-pelimarkkinoilla näkee yleensä vähemmän monipuolisuutta pelattavuudessa (mts. 3).

Proseduraalinen mallintaminen mahdollistaa pientenkin pelifirmojen kehittävän suuren luokan pelejä, sillä niiden ei tarvitse keskittyä työläisiin ja yksityiskohtaisiin rutiinitehtäviin (mts. 3). Lisäksi digitaaliset jakelualustat, kuten Steam, Itch.io ja Google Play, mahdollistavat helpon tavan levittää pelejä ilman erillistä julkaisijaa (Blatz & Korn 2017, 2). Blatz ja Korn (mts. 2) kuvaavat näistä syistä seuranneen pelialan renessanssin, jossa itsenäiset ryhmät ja yritykset kehittävät suuren määrän pelejä samalla synnyttäen pelattavuuden innovaatioita.

Yksinkertaisuudessaan proseduraalinen mallinnus hyödyntää algoritmeja monimutkaisten 3D-muotojen mallintamiseen. Erilaisia matemaattisia ja geometrisia rajoituksia säätelemällä pystytään kontrolloimaan tulosten muotoa. Näin saadaan haluttu lopputulos, joka näyttää luonnolliselta. Proseduraalisen mallinnuksen tekniikoita on useita. Ne generoivat monimutkaisia malleja joko automaattisesti tai puoliautomaattisesti ja hyödyntävät formaaleja kielioppeja, kuten muotokielioppeja (*engl. shape grammars*) tai L-systeemejä. (Merrell & Manocha 2010, 1.)

2 Tutkimusasetelma

2.1 Opinnäytetyön tavoitteet ja tutkimusongelma

Tämän opinnäytetyön tavoitteena oli selvittää erilaisia proseduraalisen 3D-mallintamisen viitekehysiksi ja niissä tapahtunutta sekä tulevaa kehittymistä. Lisäksi tutkimuksessa haluttiin perehtyä kyseisten työkulkujen vaikutuksia indie-pelikehitykseen. Näitä kokonaisuuksia tarkastellaan tietoperustassa sekä erillään että yhdessä ja tutkimustulosten avulla niistä muodostuu yhdistynyt pääkokonaisuus. Tutkimuksen tavoitteiden pohjalta opinnäytetyön tutkimusongelmaksi kehkeytyi:

Millainen vaikutus erilaisten proseduraalisen 3D-mallintamisen työkulkujen kehittymisellä on indie-pelikehitykseen?

Tutkimuskysymykset jaottelevat, mitä opinnäytetyössä tutkitaan ja mihin halutaan saada vastaus, sekä linjaavat tutkimusprosessin kulkua. Tutkimusongelmasta ja tutkittavista aiheista johdettiin kolme tutkimuskysymystä:

1. *Mitä erilaisia työkulkuja proseduraaliselle 3D-mallintamiselle on olemassa?*
2. *Miten proseduraaliset 3D-mallintamisen tavat ovat kehittyneet ja tulevat kehittymään?*

3. *Miten proseduraalisen 3D-mallintamisen työnkulun kehittyminen on vaikuttanut ja tulee vaikuttamaan indie-pelikehitykseen?*

Aiheita ja vaikutuksia tutkitaan pääosin sisällöntuottajan näkökulmasta ja yleisemmällä tasolla myös koko pelialan näkökulmasta.

2.2 Opinnäytetyön merkitys

Tutkimuksen kohde valittiin, koska proseduraalisen 3D-mallintamisen kehittyminen ja indie-pelikehityksen synergiat koettiin pelialalle tärkeäksi. Myös toisistaan erillään aiheet tuntuivat olevan kiinnostavia tutkimuksen kohteita. Opinnäytetyön aihe on ajankohtainen ja koko ajan kehittyvä. Lähitulevaisuudessa sen uskotaan olevan vielä oleellisempaa uusien ja monipuolisten työkalujen kehittyessä, kuten tekoälyn. Tutkimustuloksia voi käyttää aiheen ymmärtämisen kannalta kuka tahansa, mutta lähinnä pelialaan tutustuneet henkilöt, kuten pelikehittäjät sekä kyseisen alan opettajat ja opiskelijat.

2.3 Tutkimusmenetelmät

Tutkimusmenetelmäksi tähän opinnäytetyöhön valittiin kvalitatiivinen eli laadullinen tutkimus, koska se koettiin tutkimusongelman ratkaisemisen kannalta parhaimmaksi lähestymistavaksi. Laadulliselle tutkimukselle ominaista on tutkia jotakin ilmiötä ja yrittää ymmärtää sitä (Saaranen-Kauppinen & Puusniekka 2006). Tämän opinnäytetyön tutkimusongelmaa tutkitaan enemmän ilmiönä kuin tilastoilla testattavana ennako-oletuksena. Koska aihe on jatkuvasti ja osittain nopeastikin kehittyvä, on tutkimustuloksia hyödyllisempää tuottaa laadullisesti. Laadulliset tutkimustulokset eivät siis ole ajattomia ja paikattomia. Tutkimussuunnitelma monesti elää tutkimuksen rinnalla, jolloin saavutetaan ilmiöiden prosessiluonne. (Eskola & Suoranta 2003, 16.)

Laadullisessa tutkimuksessa tutkimusaineistoa kuvataan yleisenä esimerkkinä tutkittavasta aiheesta, mutta sitä ei voida suoraan yleistää kuvastamaan koko aihetta. Sen sijaan yleistyksiä tehdään aineiston pohjalta tehdyistä analyyseista. Tutkimusaineisto ei ole yleensä laaja ja jopa yksittäinenkin tapaus voi kelvata aineistoksi. (Saaranen-Kauppinen & Puusniekka 2006.) Sitä tarkastellaan kokonaisuutena, vaikka aineisto koostuisikin yksilöllisistä muuttujista (Alasuutari 2011, 38). Resurssien ollessa rajallisia tutkittavien kohteiden määrä jää opinnäytetyössä suppeaksi,

jolloin laadullinen tutkimus sopii paremmin menetelmäksi. Tämä ei tosin haittaa, koska työssä aineistoa kerätään haastattelulla ja kyselyillä. Laadullinen aineisto ilmaistaan tekstillä, joita ovat esimerkiksi litteroidut haastattelut (Eskola & Suoranta 2003, 15). Tällaisen aineiston koetaan olevan ilmiön tutkimisen kannalta tärkeämpää kuin tilastollinen tieto. Lisää opinnäytetyön aineistonkeruusta ja sen menetelmistä omassa alaluvussa.

Laadullisessa tutkimuksessa ei ole kiveen kirjoitettuja hypoteeseja eli tutkimuksessa ei kokeilla ja tavoitella minkään tietyn teorian toimivuutta (Eskola & Suoranta 2003, 15). Tutkijalla saattaa olla joitain esioletuksia aiheesta, mutta ne ovat oleellista tiedostaa, eikä niiden saa antaa häiritä tutkimuksen toteuttamista. (Saaranen-Kauppinen & Puusniekka 2006.) Ennen opinnäytetyön aloittamista tutkittavat aiheet olivat miltei tuntemattomia, joten vahvoja ennako-oletuksia ei ollut. Tämä myös mahdollisti tutkimusprosessin kautta oppimaan uutta ja saamaan uusia näkökulmia.

Vastapainona laadulliselle tutkimukselle on kvantitatiivinen eli määrällinen tutkimus. Siinä tiedonkeruumenetelmät ovat yleisesti tilastollisia kyselyitä tai haastatteluja (Alasuutari 2011, 31). Argumentointi tapahtuu lukujen ja niiden systemaattisten, tilastolliset yhteyksien avulla. Kvantitatiivisessa tutkimuksessa keskeisiä piirteitä ovat hypoteettisuus ja deduktiivisuus. (Alasuutari 2011, 34; Eskola & Suoranta 2003, 14.) Vaikka tutkimusotteissa on eroja, ne eivät ole toisiaan pois sulkevia analyysimalleja. Molemmissa tutkimusotteissa on paljon samaa ja niitä voidaan soveltaa samassa tutkimuksessa esimerkiksi aineistonkeruussa ja -analyysissa. Periaatteena kummasakin on perustelu havaintoihin eikä omiin subjektiivisiin mieltymyksiin. (Alasuutari 2011, 32.) Tämä opinnäytetyö on pääasiassa laadullinen tutkimus, mutta sisältää muutamia määrällisen tutkimuksen piirteitä, kuten kyselyhaastatteluja ja teoriasidonnaista analysointia.

2.4 Aineistonkeruu ja -analyysi

Tutkimustyön aineistona toimii asiantuntijan teemahaastattelu, kyselyt sekä valmiit aineistot. Haastattelu kuvattiin ja nauhoitettiin osallistujan luvalla, minkä jälkeen se litteroitiin ja analysoitiin. Kysely julkaistiin verkossa sekä lähetettiin valituille henkilöille ja tuloksia analysoitiin laadullisesti. Lopuksi analyysista johdettiin johtopäätöksiä. Myös muiden tutkijoiden keräämiä sekundaariaineistoja eli pääasiassa aiheesta tehtyjä tutkimuksia ja kirjallisuutta hyödynnetään tutkimuskysymykseen vastaamisessa. Tutkimusaineiston hankkimiseen käytettiin siis menetel-

mättriangulaatiota. Se tarkoittaa useamman erilaisen tiedonhankintamenetelmän, kuten määrällisen ja laadullisen, käyttämistä tutkimusaineiston keräämiseen (Saaranen-Kauppinen & Puusniekka 2006). Haastateltavien henkilöiden vaikea saatavuus antoi motivaatiota kyselylomakkeen käyttöönotolle. Lisäksi useamman tutkimusmenetelmän käyttö lisää tutkittavien kohteiden määrää ja antaa erilaisia näkökulmia.

Opinnäytetyön analyysi nojaa enemmän aineistolähtöiseen tutkimukseen, jolloin tutkimuksen painoarvo on aineistossa eikä analyysi perustu johonkin valmiiseen teoriaan. Yksittäisistä havainnoista edetään yleisempiin tulkintoihin induktiivisesti. Koska opinnäytetyössä on tosin käytetty teoriapohjan sekundaariaineistoja tukemaan oman aineiston analyysia, voisi sitä tarkemmin sanottuna kutsua teoriasidonnaiseksi tutkimukseksi. (Mt.)

2.4.1 Tiedonhaku

Yksi osa aineistonkeruuta on sekundaariaineistojen etsiminen. Ne ovat aiempiin tutkimuksiin käytettyjä aineistoja, joita muut tutkijat ovat keränneet omia tutkimuksiaan varten. Se, että ne on alun perin kerätty toisiin tutkimuksiin, ei tee niistä huonompaa aineistoa. Näistä aineistoista tehtyjä analyysseja kutsutaan taas sekundaarianalyysseiksi. (Mt.)

Tässä opinnäytetyössä systemaattista tiedonhakua on toteutettu käyttämällä avainsanoja hakusanoina erilaisissa tieteellisiä julkaisuja julkaisevissa palveluissa. Päällimmäisinä hakusanoina toimivat oletettavasti *procedural modeling* ja *indie game development*, joihin yhdistettiin *advancement* (suom. kehitys). Kun kunkin aiheen osalta haluttiin löytää yksityiskohtaisempaa tietoa, haettiin erilaisia aiheisiin liittyviä käsitteitä. Näitä ovat proseduraalisen mallintamisen kohdalla esimerkiksi *L-systems*, *fractals*, *shape grammars*, *noise* ja *generative models*. Tiedonhakua tehtiin esimerkiksi Institute of Electrical and Electronics Engineers- (IEEE) ja ResearchGate-palveluihin sekä Google Scholar-hakupalveluun. IEEE on maailman suurin kansainvälinen tekniikan alan järjestö. ResearchGate on tutkijoille suunnattu julkaisupalvelu sekä keskustelufoorumi ja sopii hyvin erilaisten tutkimusten tiedonhakuun heidän semanttisen hakukoneensa ansiosta. Google Scholar on Googlen kehittämä laaja hakukone tieteellisiä julkaisuja varten. Lisäksi tiedonhakua on tehty syötämällä hakusanoja eri hakukoneisiin, kuten Google ja DuckDuckGo.

Tiedonhakuja on suoritettu ensin kattavasti, jota läpikäymällä on löydetty epärelevanttien hakutulojen joukosta opinnäytetyölle relevantteja lähteitä. Niiden pohjalta on suoritettu suppeampaa hakua, joka rajaa tuloksia edellistä tarkempiin hakutuloksiin ja vähentää epärelevantteja tuloksia.

2.4.2 Teemahaastattelu

Yksi perinteinen laadullisen tutkimuksen aineistonkeruumenetelmistä on teemahaastattelut ja sitä on hyödynnetty tässäkin tutkimuksessa. Saaranen-Kauppinen ja Puusniekka (2006) kuvailevat haastattelun olevan enemmän keskustelun tapaista ja etenee valmiiksi mietittyjen teemojen pohjalta. Kysymykset eivät ole välttämättä samoja kaikille ja niiden järjestystä voidaan muuttaa. Tätä ei kuitenkaan tule sekoittaa avoimeen haastatteluun, joka on kokonaan strukturoimaton aihetta lukuun ottamatta. Haastateltavaksi valitaan henkilöitä, joiden ajatellaan olevan oleellisen tutkimusaineiston tuottamisen kannalta hyödyllisiä. (Mt.) Opinnäytetyössä haastateltavana oli parikymmentä vuotta monella pelialan osa-alueilla toiminut asiantuntija, joten hänen koettiin olevan tutkimukseen sopiva henkilö. Teemahaastattelu valittiin menetelmäksi, koska se tehokas tapa kerätä aineistoa tämän tapaiseen tutkimukseen ja sopii hyvin ensimmäistä haastattelua tekeväälle henkilölle.

2.4.3 Kyselylomake

Toisena aineistonkeruumenetelmänä on käytetty kyselyhaastattelua, joka sijoittuu ennemmin puolistrukturoiduksi haastatteluksi. Siinä kaikille haastateltaville esitetään suunnilleen samat kysymykset samassa järjestyksessä (mt.). Strukturoidusta lomakehaastattelusta kyselyhaastattelu kuitenkin eroaa siten, että kyselyssä on valmiiden vastausvaihtoehtojen sijaan avoimet vastauskentät. Avoimet kysymykset antavat menetelmälle enemmän laadullisen luonteen, vaikka se onkin teemahaastattelua formaalisempi ja kvantitatiivisempi. Monivalintakysymysten puute tosin saattaa vähentää osallistujien halukkuutta vastata, mutta aineistoksi haluttiin kerätä laadullista aineistoa määrällisen sijaan. Saaranen-Kauppinen ja Puusniekan mukaan kysymysten sekä avoimien vastaus-ten määrä ja monipuolisuus vaikuttavat siihen, voiko aineistoa sisällyttää laadulliseen analyysiin.

Kysymysten vastauksia voidaan myös halutessaan muuttaa määrälliseksi aineistoksi esimerkiksi kvantifioimalla eli muuttamalla ne tilastolliseksi tiedoksi. Voidaan esimerkiksi laskea, kuinka monta kertaa tietty ilmiö esiintyy jossain yhteydessä. (Mt.) Kyselyhaastatteluja käytetään opinnäytetyössä

pikemmin lisäaineistona tukemaan muuta laadullista tutkimusta. Ne toteutettiin englanninkielisellä Google Forms-kyselylomakkeella (ks. liite 2), jossa oli 12 kysymystä. Kyselyihin vastasi kuusi henkilöä, jotka olivat indie-pelikehittäjiä ja 3D-graafikkoja. Osallistujia etsittiin lähettämällä kyselylinkki tutkimukseen sopiville henkilöille ja julkaisemalla linkki kyselyyn alaan liittyvillä Reddit-foorummeille. Tarkemmin tästä on kerrottu luvussa 9.2 *Kyselyhaastattelujen tausta ja tulokset*.

2.4.4 Analysointi

Tutkimusaineiston keräämisen jälkeen seuraava askel on sen analysointi, jonka päätarkoituksena on kertoa aineiston sisällöstä ja tulkita sitä. Analysointi alkaa aineiston huolellisesta lukemisesta ja sen sisäistämisestä, minkä jälkeen sen rakennetta järjestellään ja sisältöä tiivistetään. Näkökulmia otetaan tutkimustehtävän kannalta olennaisiin kohtiin. Analyysimenetelmiä on kuitenkin erilaisia ja niitä sovelletaan eri tutkimuksiin niihin sopivilla tavoilla. (Mt.)

Kuten jo aiemmin mainittiin, haastattelu videoitiin ja ääni nauhoitettiin, jonka jälkeen se litteroitiin eli muutettiin tekstimateriaaliksi. Tätä syntyi suuri määrä, minkä takia sen sisällön kiteyttäminen ja jäsentely oli oleellista. Apuna aineiston käsittelyssä voi käyttää muun muassa kvantifiointia tai koodausta, vaikkei se olekaan tutkimusprosessin kannalta pakollista (mt.). Aineiston koodausta voi Saaranen-Kauppinen ja Puusniekan mukaan tehdä esimerkiksi alleviivaamalla samalla värillä kaikki jotakin tiettyä aihetta käsittelevät kohdat, jotta ne löytyvät aineistosta nopeasti.

Kun opinnäytetyön haastattelu- ja kyselyaineistoja oli jäsennelty edellä mainituilla tekniikoilla, oli seuraava looginen tehtävä sen teemoittelu. Se sopii etenkin teemahaastattelujen analysoimiseen, koska teemoittelun tavoitteena on muodostaa keskeisiä yhdistäviä teemoja aineistosta. Jo haastattelurungosta voi olettaa, mitä kyseiset teemat ovat ja monesti ne ovatkin samoja, mutta joskus teemat voivat osittain erota tutkijan suunnitelluista aiheista ja/tai olla eri järjestyksessä. (Mt.) Analyysi ei kuitenkaan ole pelkkää aineiston jäsentämistä teemoihin. On oleellista haravoida aineistoa systemaattisesti läpi löytääkseen sisällöllisiä yhteyksiä ja eroja, mutta niitä täytyy myös osata peilata tutkimusongelmaan. Näiden synteesisistä muodostuu analyysi.

2.5 Luotettavuuden varmistaminen

Kvalitatiivisessa tutkimuksessa tutkija on tutkimuksen työväline. Hänen avoin subjektiivisuutensa ja sen myöntäminen nostaa tutkimuksen luotettavuutta sekä pätevyyttä. Tutkijan ja luotettavuuden arviointi koskee koko tutkimusprosessia. (Eskola & Suoranta 2003, 210.) Kvantitatiivisessa tutkimuksessa luotettavuus koskee nimenomaan mittauksen luotettavuutta (Eskola & Suoranta 2003, 211).

Olenaisia käsitteitä luotettavuuden arvioinnin osalta ovat validiteetti ja reliabiliteetti. Validiteetti kuvaa tutkimuksen pätevyyttä, eli kuinka asianmukaisesti ja sopivilla menetelmillä ilmiötä on tutkittu (Saaranen-Kauppinen & Puusniekka 2006). Myös uskottavuus on osa validiteettia ja se kertoo, vastaavatko tutkijan käsitteellisyys tutkittavien käsityksiä (Eskola & Suoranta 2003, 211–212). Opinnäytetyön osalta tutkimusmenetelmien valinta ja sen perustelut on koettu oikeutetuiksi. Esimerkiksi useamman tutkimusmenetelmän käyttö lisää luotettavuutta. Tutkimuksen kohderyhmä on tutkimuskysymysten ratkaisun kannalta otollinen. Aineistonkeruun ja -analyysin luotettavuuden sekä eettisyyden osalta enemmän tutkimuksen pohdintaosiossa.

Reliabiliteettia arvioidaan muun muassa tutkimustulosten siirrettävyydellä. Vahvistuvuus tarkoittaa reliabiliteetin kriteerinä sitä, että muut ilmiötä tarkastelevat tutkimukset tukevat saatuja tutkimustuloksia. (Eskola & Suoranta 2003, 211–212.) Havaintojen pysyvyyden arviointi on laadullisessa tutkimuksessa osittain ongelmallista, koska niissä yleensä tutkitaan muuttuvaa ilmiötä (Saaranen-Kauppinen & Puusniekka 2006). Tässäkin tutkimuksessa tehdyt havainnot tulevat todennäköisesti vanhentumaan osittain nopeastikin, mutta kertoo se silti tietyn ajan kuvausta ilmiöstä. Myöskään kaikki havainnot eivät välttämättä ole muuttuvia, kuten aiheen historia tai matemaattiset faktat. Toinen ongelma laadullisin keinoin saatujen tulosten kanssa on niiden uudelleen tuottaminen, koska havainnot eivät ole tarkkoja mittaustuloksia (Saaranen-Kauppinen & Puusniekka 2006). Tämä on oleellista ottaa huomioon laadullista tutkimusta tehdessä.

Lähteillä on suuri vaikutus tutkimuksen luotettavuuteen. Ne tulee merkitä huolellisesti ja niitä kohtaan tulee olla lähdekriittinen. Opinnäytetyössä on pidetty erityistä tarkkaavaisuutta lähdemerkinnöissä. Relevantteja lähteitä on etsitty laajasti aiemmin kuvatulla tavalla ja niiden luotettavuutta on pohdittu. Tätä on tehty esimerkiksi miettimällä julkaisijan/kirjoittajien motiiveja tai etsimällä tietoa siitä/heistä sekä vertaamalla lähteiden luotettavuutta keskenään.

Hyvä tieteellinen käytäntö on tiedeyhteisön yhteisesti tunnustama tutkimusohje. Sen pääpiirteitä ovat rehellisyys, huolellisuus ja tarkkuus. HTK ohjaa eettiseen sekä vaatimusten mukaiseen tutkimukseen aineiston hankkimis- ja analysointimenetelmien osalta. Muut tutkijat ja heidän tutkimustyönsä on huomioitava asiallisesti. Tiedevilppi eli sepittäminen, havaintojen vääristely, plagiointi ja anastaminen on ehdottomasti kielletty hyvän tieteellisen käytännön puitteissa. Hyvän tieteellisen käytännön noudattaminen lisää tutkimuksen luotettavuutta (Saaranen-Kauppinen & Puusniekka 2006.)

3 Indie-pelikehitys

Indie-pelien määritelmässä on paljon erimielisyyksiä, mutta yleisessä mielessä ne ovat itsenäisten peliyritysten (*eng. independent*) pelejä, joilla ei ole suuria ulkopuolisia rahoittajia ja julkaisijoita. Indie-pelin itsenäisyyttä kerrotaan myös olevan pelisuunnittelun tapahtuminen ilman ulkopuolisen ylemmän johdon sekaantumista. (Gril 2008.)

Monesti itsenäisillä peliyrityksillä on pienet tai olemattomat pääomatulot. Ihmistyö on kallista ja aikaa vievää, joten proseduraalinen mallinnus helpottaa pienten peliyritysten pelikehitystä huomattavasti. Proseduraalinen mallintaminen ei kuitenkaan korvaa 3D-mallintajaa vaan toimii enemmän työkaluna nopeuttamassa pitkäväteisiä työosuuksia. (Price 2018.)

Indie-pelikehittäjä hakee monesti vapautta suurempien pelifirmojen vaikutusvallasta, vaikka se ei olisi taloudellisesti yhtä kannattavaa. Indie-pelit saattavat kuitenkin uskaltaa olla enemmän korkeellisimpia joissain kohtaa verrattuna isojen firmojen peleihin, jotka ottavat vähemmän luovia riskejä. (YLE 2012.) Esimerkiksi kulttihiliksi muodostunut Minecraft-peli (2011) oli alun perin indie-peli ja on nykyään maailman eniten myydyin videopeli (Bailey 2020; Cheshire 2014).

4 Proseduraalinen 3D-mallinnus

Proseduraalinen mallinnus on laaja käsite, mutta sitä voisi havainnollistaa esimerkiksi tällä tavalla: Tehtävänä on mallintaa kaksikymmentä erilaista 3D-kerrostalomallia. Sen sijaan, että jokaista rakennusta lähdetään mallintamaan yksitellen, voidaan hyödyntää proseduraalista 3D-mallinnusta. Ensiksi mallinnetaan ainoastaan halutut muuttujat, kuten ovet, ikkunat, seinät ja katot, jonka jälkeen ohjelmiston algoritmi tekee näistä osista erilaisia rakennuksia proseduraalisesti. Näiden

muuttujien arvoja voidaan muunnella haluamalla tavalla, esimerkiksi määrittämällä ovien ja ikkunoiden lukumäärän tai rakennuksen leveyden ja korkeuden.

4.1 Taustaa

Lyhykäisyydessään proseduraalista mallinnusta voidaan kuvata rajoituksilla säädeltäväksi satunnaisuudeksi (Bubriski 2016). Nämä rajoitukset eli sääntösarjat voivat olla joko algoritmin sisäisiä ja parametreilla säädeltäviä tai olla kokonaan erillään algoritmin evaluaatiokoneistosta (Lorens 2019).

Tietokannan vahvistaminen proseduraalisessa mallinnuksessa mahdollistaa laajojen alueiden generoimisen pienellä määrällä sääntöjä. Jos algoritmin tuottama tuloste on aina sama, tulostetta ei tarvitse säilöä. Sen sijaan käytetyistä säännöistä voidaan muodostaa siemen (*engl. seed*), jota algoritmi käyttää tulosteen saamiseksi. Käyttäjän syöte ei ole mallintamisen osalta välttämätöntä, koska mallit luodaan siementen pohjalta ja niitä voidaan muokata yksittäisten sääntöjen arvoja vaihtamalla. (Mt.)

Proseduraalista mallinnusta käytetään yleensä silloin, kun halutaan luoda paljon aikaa vieviä laajoja kokonaisuuksia, kuten arkkitehtuuria tai kasvistoa (mt). 3D-videopeleissä proseduraalista mallinnusta hyödynnetään monesti myös maaston, vihollisten, saaliin (*engl. loot*) ja muiden mallien ketterään luomiseen (Bubriski 2016). Beihangin yliopiston opiskelijat Siyuan Wang, Junjun Pan, Junxuan Bai ja Jinglei Wang (2020) loivat Flower Factory-nimisen (*suom. kukkatehdas*) viitekehysten ketterälle kukkien mallinnukselle. Kukkat koostuivat eri komponenteista, kuten terälehdistä, heiteistä, kukkapohjuksista ja lehdistä. Pisteillä ja splineilla muodostettiin näiden ulkomuodot, jonka jälkeen tekstuurit valmistuvat automaattisesti ottaen mallia oikeista kukista. Koko prosessia pystyy ohjaamaan parametreilla, jotka määrittelevät kukkien attribuutit. (Mt.)

Algoritmeja käyttävät proseduraaliset mallinnustekniikat hyödyntävät muun muassa fraktaaleja, L-systeemejä, muotokielioppeja, kohinoita ja generatiivisia malleja luodakseen proseduraalista sisältöä. Tätä sisältöä voi halutessaan muokata manuaalisesti. (Lorens 2019). Fraktaali on loputon kuvio, joka toistaa itseään ja näyttää samanlaiselta kaikissa kokoluokissa. Se on matemaattista geometriaa, jota esiintyy kaikkialla luonnossa, ja siksi sitä hyödynnetään proseduraalisessa generoinnissa. (Fractal Foundation 2009.) L-systeemi on kasvitieteilijä Astrid Lindenmayerin kehittämä matemaattinen malli, jolla pystyy simuloimaan kasvien kasvamisprosessia ja joitakin muita

orgaanisia rakenteita, kuten fraktaaleja (Ruemmler n.d.). Muotokieliopin komentosarjalla voidaan yksinkertaistettuna muokata valittu alkumuoto haluttuun loppumuotoon (Jesus 2016). Kohinaa luodaan määrittelemällä ristikkoon pisteille näennäissatunnaiset arvot, joiden väliin kohinafunktio laskee uusia arvoja tunnettujen arvojen perusteella (Perlin 2007). Generatiivinen malli on todennäköisyyttä laskeva tilastollinen malli (Karpathy, Abbeel, Brockman, Chen, Cheung, Duan, Goodfellow, Kingma, Ho, Houthoof, Salimans, Schulman, Sutskever & Zaremba 2016). Näitä käsitteitä avataan tarkemmin omissa luvuissaan.

4.2 Hyödyt

Proseduraalinen mallintaminen on epädestruktiivista (*engl. non-destructive*) eli siihen pystyy tekemään milloin tahansa muutoksia vaikuttamatta alkuperäiseen rakenteeseen ja palaamaan edellisiin versioihin rakenteesta. Kun mallia modifioidaan, muutokset kasaantuvat perustason päälle omille tasoilleen. Näin ollen perustasolla oleva rakenne ei muutu. Koska muutokset ovat omilla tasoillaan, niiden muokkaaminen ei vaikuta muihin tasoihin. Lisäksi proseduraalinen mallintaminen on aikaa säästävää ja helppokäyttöistä. (Lorens 2019.)

5 Fraktaali ja L-systeemit

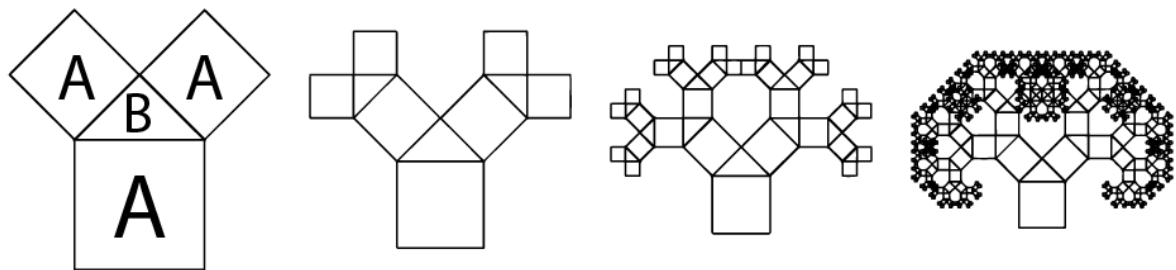
Fraktaali on itsesimilaarinen joukko eli se rakentuu keskenään yhdenmuotoisista pienemmistä osista. Fraktaali voi olla täysin tai satunnaisesti itsesimilaarinen. Yleensä luonnossa esiintyvät fraktaalit ovat satunnaisesti itsesimilaarisia. Matemaattisesti luotu fraktaali on ääretön, koska sen kuvat ovat rekursiivisia. Ne toistuvat loputtomasti missä tahansa mittakaavassa. Monimutkaisuutensa vuoksi sitä ei voi kuvata euklidisella geometrialla, jossa mitat ovat aina kokonaislukuja. Vaikka suora viiva on itsesimilaarinen, se ei ole fraktaali, koska sen voi kuvata euklidisella geometrialla. (Portillo 2022.)

Nykyään todella monessa videopelissä käytetään proseduraalista generointia kasviston luomiseen, koska kyseinen viitekehys soveltuu siihen mainiosti. Yksi yleisimmistä tekniikoista on aiemmin mainittu Lindenmayer-systeemi eli L-systeemi. Tällä tavalla pystyy luomaan luonnollisen näköisiä kasveja, jotka muistuttavat toisiaan, mutta eivät ole toistensa kopioita. (Togelius, Shaker & Dormans

2016, 73.) Kuten jo aiemmin mainittiin, fraktaalit ovat itsesimilaarisia. Yleisimmät esimerkit luonnossa esiintyvistä itsesimilaarisuudesta ovat romanesko-parsakaali ja saniaisen oksa. Koska L-systeemeillä on helppo luoda fraktaaleja, se sopii luonnollisten kasvien mallintamiseen. (Mts 74.)

Pythagoraan puu (ks. kuvio 1.) on hyvä tapa visualisoida ja selittää, miten L-systeemi toimii ja miten sillä pystyy rakentamaan orgaanista kasvua. Se on neliöistä ja niitä yhteen sitovista suorakulmaisista kolmioista rakentuva fraktaali (Weisstein 2022). Aksioma (neliö) kuvaa lähtötilannetta ja toimii kuvion merkkijonojen "siemenenä" (A) (Togelius ym. 2016, 76). Ensimmäinen rekursio lisää neliön päälle suorakulmaisen kolmion (B), jonka kateetteihin kiinnittyvät seuraavat neliöt. Jokaisella seuraavalla rekursiolla tapahtuu uusi iteraatio edellisen tapaan. (Weisstein 2022.) Esimerkki L-systeemin säännöistä:

1. $A \rightarrow B$
2. $B \rightarrow AA$



Kuvio 1. Pythagoraan puu.

L-systeemejä pystyy käyttämään lukuisiin muihin tarkoituksiin. Togelius ja muut (2016, 78–95) selittävät perinpohjaisesti, miten niillä pystyy luomaan proseduraalisesti esimerkiksi toimintaroolipeliä tehtäviä ja luolastoympäristöjä tai tasohyppelypelin pelitasoja. 3D-mallintamisen kannalta oleellisempi soveltaminen L-systeemeille on esimerkiksi Parishin ja Müllerin (2001) tutkimuksessa kuvattu proseduraalinen tieverkoston mallintaminen.

5.1 L-systeemit proseduraalisessa mallintamisessa

Edellisessä aluvussa perehdyttiin perinteisten L-systeemien perusteisiin. Tässä luvussa syvennytään edistyneempiin L-systeemeihin, joita hyödynnetään proseduraalisessa mallintamisessa.

Kun niitä yhdistellään 3D-mallinnustekniikoihin, pystyy generoimaan monimutkaistakin geometriaa (Schinko, Krispel & Ullrich 2015, 12).

Generatiivinen suunnittelu on mahdollista, kun suunnitteluprosessin vaiheet ovat toistuvia ja muistuttavat toisiaan (mts. 11). Vaikka Schinko ja muut (mts. 11) mainitsevat generatiivisen lähestymistavan olevan vastakohta luovalle prosessille, ja etteivät nämä vaiheet vaadi käyttäjän luovia ratkaisuja, on kuitenkin mainittavaa, että proseduraalinen mallinnus tarvitsee luovuutta muissa osa-alueissa. Dormans ja Leijnen (2013) perustelevat tätä sillä, että generoivan vaiheen jälkeen käyttäjän on osattava suodattaa haluamansa tulokset ja muuttaa ne järkeenkäyväksi sekä hyödylliseksi kokonaisuudeksi.

5.2 Parametrinen L-systeemi

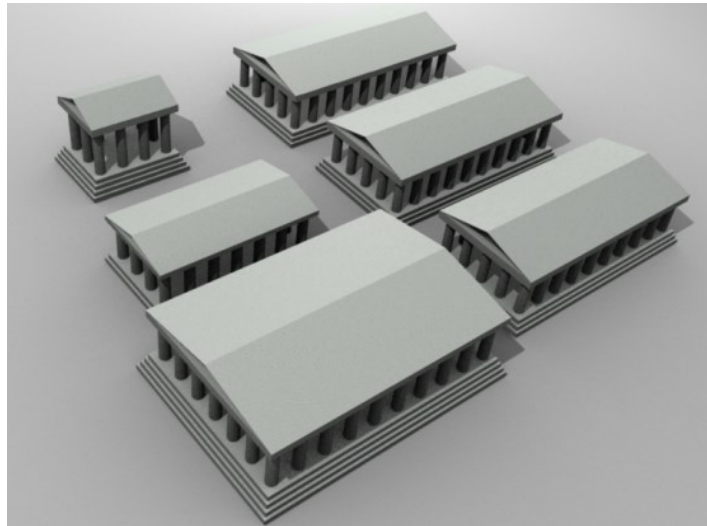
Aiemmassa alaluvussa perinteistä L-systeemiä selitettiin Pythagoraan puulla. Tarkemmin ottaen L-systeemi koostuu aksiomasta eli lähtöpisteestä, erilaisista symboleista ja sääntösarjoista, jotka ohjaavat prosessin etenemistä uudelleenkirjoittamalla aksiomaa. Tämän jälkeen niistä syntynyttä merkkijonoa tulkitaan (Hidalgo, Camahort, Abad & Vicent 2008, 2).

Siinä missä perinteisten L-systeemien geometriset esitystavat ovat rajoittuneita fraktaalien ja erilaisten kasvien kasvamisen mallintamiseen, Schinko ja muut (2015, 11) kertovat parametrinen L-systeemien pystyvän kuvaamaan monimutkaisempaa geometriaa. Parametriset L-systeemit käyttävät aritmetiikkaa ja totuusarvomuuuttujia (*engl. boolean*) parametreissaan uudelleenkirjoitusvaiheen aikana (Hidalgo ym. 2008, 2). Niissä numeeriset parametrit ovat L-systeemien symbolien tukena (Schinko ym. 2015, 11).

5.3 Stokastinen L-systeemi

Parametrisia L-systeemejä seurasi stokastiset L-systeemit, jotka tavallaan lukeutuvat myös parametrisiin L-systeemeihin (Hidalgo ym. 2008, 2, 5). Lyhyesti sanottuna stokastinen generointi on satunnaista eli se tuottaa aina eri tuloksen, vaikka samoja parametrejä käytettäisiin. Tämän vastakohtana on deterministinen generointi, joka generoi saman tuloksen joka kerta samoilla parametreilla. Tästä on hyötyä, mikäli halutaan pakata dataa ja säästää muistia. (Togelius, Yannakakis, Stanley & Browne 2010, 3.)

Stokastisissa L-systeemeissä sijoitetaan satunnaisia muuttujia parametreihin, jotta mallinnukseen saadaan luonnollista satunnaisuutta (Hidalgo ym. 2008, 2). Hidalgo ja muut (mts. 5) antavat stokastista L-systeemiä käyttävästä generoinnista esimerkin, jossa mallinnetaan satunnaisten muuttujien pohjalta Doorilaisia temppeleitä muistuttavia 3D-rakennuksia (ks. kuvio 2.). Tässä esimerkissä rakennus toimii L-systeemin symbolina ja parametreina lattia, pylväät sekä katto. Parametreille annetaan satunnaiset muuttujat, jotka kuvaavat pylväiden korkeutta ja niiden etäisyyttä toisistaan, katon korkeutta, rappusten määrää sekä itse rakennuksen leveyttä ja korkeutta. Tietyt rajoitukset antavat muuttujille minimi- ja maksimiarvot. Lisäksi rakennuksen mitat ja pylväiden etäisyydet vaikuttavat pylväiden määrään. (Mts. 5.)



Kuvio 2. Edessä stokastisesti luodut rakennukset ja takana kolme deterministisesti luotua rakennusta (Hidalgo ym. 2008, 7).

Näiden L-systeemien lisäksi Hidalgo ja muut (mts. 2–3) kertovat vielä kontekstista riippuvaisista L-systeemeistä, joissa generoidut objektit ovat vuorovaikutuksessa ympäristöönsä tai muihin objekteihin. Nämä edistyneemmät avoimet L-systeemit voivat esimerkiksi luoda kasveja, jotka kasvavat valoa kohti (mts. 8).

6 Muotokieliopit

Muotokieliopin lähtökohtana on jokin alkumuoto, jota muokataan komentosarjalla. Kielioppisäännön vasemmalla puolella on symboli eli tässä tilanteessa jokin muoto ja oikealla puolella komento, joka luo uusia muotoja jatkokäsittelyä varten. Komentojen vaikutusta geometriaan ohjaavat mo-

nosti niiden sisäiset parametrit. Geometriaa kehitetään siis iteratiivisesti kieliopin derivointiprosessilla korvaamalla muoto uudella muotojoukolla. Tätä jatketaan niin kauan, että muoto tulee päätepisteeseen. Päätepisteeseen se tulee, kun kieliopissa oleva komento ei enää tuota uutta muotoa eli symbolia sen oikealle puolelle. Geometrian lisäksi muodolla on myös materiaaleihin liittyvää tietoa ja skaala (*engl. scope*), joka määrittelee orientaation ja tilavuuden.

Yleisimmät komennot rakennusten proseduraalisessa mallintamisessa ovat ekstruusio (*extrusion*), jakaminen (*split*), toistuva jakaminen (*repeat split*), pinnan jakaminen (*face split*) ja sijoittaminen (*insert*). Ekstruusio tarkoittaa muodon ulospäin työntämistä, esimerkiksi tasaisen tason ylöspäin laajentamista luodakseen rakennukseen kerroksia. Jakamiskomento jakaa muodon useampaan muotoon akselia pitkin, esimerkiksi rakennuksen julkisivun jakamista x- ja y-akselilla ristikon muotoon luodakseen ikkunoille omat paikkansa. Toistuva jakaminen yksinkertaisesti suorittaa jakamiskomennon toistuvasti. Pinnan jakaminen mahdollistaa muodon pintojen muokkaamista, kuten rakennuksen julkisivun ja muiden sivujen erilaisen geometrian toteuttamista. Sijoittamiskomennolla pystyy lataamaan tiedostoista valmiiksi mallinnettuja objekteja, kuten ikkunoita tai ovia rakennukseen. (Jesus 2016.)

6.1 CGA-muotokielen käyttö rakennusten mallintamisessa

CGA (*engl. Computer Generated Architecture*) on muotokieli tietokoneella generoitavalle arkkitehtuurille, jota CityEngine-mallinnusohjelmisto käyttää proseduraaliseen mallintamiseen. Se perustuu jakamiskielioppiin, joka toimii jakamalla muodon pienempiin komponentteihin. Wonka, Wimmer, Sillion ja Ribarsky (2003, 3) esittelevät, miten tällä kieliopilla pystyy luomaan kolmiulotteisia rakennuksia johtamalla ne yksinkertaisista muodoista. Alkumuotona toimii rakennuksen kaksiulotteinen "tontti", josta johdetaan rakennuksen ulkoseinät eli julkisivut. Tämä on niin sanottu massamalli eli rakennuksen karkea tilavuus. Julkisivut jaetaan kerroksiin ja sarakkeisiin, joista muodostuu taas laattoja (*engl. tile*). Näitä laattoja voi jakaa entisestään ja niihin voidaan sijoittaa rakenteellisia elementtejä, kuten ikkunoita, koristeita, parvekkeita ja ovia. (Wonka ym. 2003, 3.)

Jakamiskielioppia kehittivät edelleen Müller, Wonka, Haegler, Ulmer & Van Gool (2006) parantamalla massamallinnusta ja rakennusten osien yhdistämistä. Heidän massamallinsa koostuvat yksinkertaisista toisiinsa yhdistyneistä tilavuudellisista muodoista, kuten suorakulmaisista särmiöistä ja sylintereistä sekä kattoja varten olevista muodoista, kuten pyramideista ja kartioista. Müller ja

muut selittävät julkisivujen suunnittelun pysyvän johdonmukaisena testaamalla alueellisia päällekkäisyyksiä rajapintakyselyllä sekä testaamalla toisiaan lähellä olevia pintoja ja linjoja niin kutsutuilla napsahduslinjoilla (*engl. snap lines*). Rajapintakysely tarkastaa massamallin kahden pinnan leikkauskohdan ja palauttaa tiedon siitä, onko kyseinen muoto kokonaan peittynyt, osittain peittynyt tai ei-ollenkaan peittynyt. (Müller ym. 2006, 4.) Ympäristön huomioon ottavat säännöt varmistavat, ettei ovia tai ikkunoita aseteta sivujen leikkauskohtiin eli rakennuksen kulmiin, ja että ovet avautuvat joko katutasolle tai parvekkeille (Müller ym. 2006, 1). Kun muotokielen komento muuttaa pinnan muotoa, laatta loksahdaa paikalleen lähimpään napsahduslinjaan (Müller ym. 2006, 5).

6.2 Rakennusten mallintaminen kuvasyötteestä

Myöhemmin on kehitetty tapoja muodostaa muotokielioppeja valokuvien perusteella. Tämä mahdollistaa nopean tavan visualisoida olemassa olevia rakennuksia tai luoda uusia erilaisia rakennuksia sellaisella tyylillä, joka mukailee alkuperäisiä rakennuksesta otettuja valokuvia. Kun syötteeksi antaa kuvan rakennuksen julkisivusta; Müllerin, Zengin, Wonkan ja Van Goolin (2007, 1) algoritmit muodostavat syötettä kuvaavan realistisen ja korkealaatuisen 3D-mallin rakennuksesta. Samantapaisen metodin esittävät tutkimuksessaan myös Aliaga, Rosen ja Bekins (2007, 1), mutta pelkän suoraan edestä päin otetun julkisivukuvan sijaan syötteenä voidaan käyttää kuvia koko rakennuksesta, jolloin rakennuksen tyyli ei muodostu pelkästään yhdestä julkisivusta.

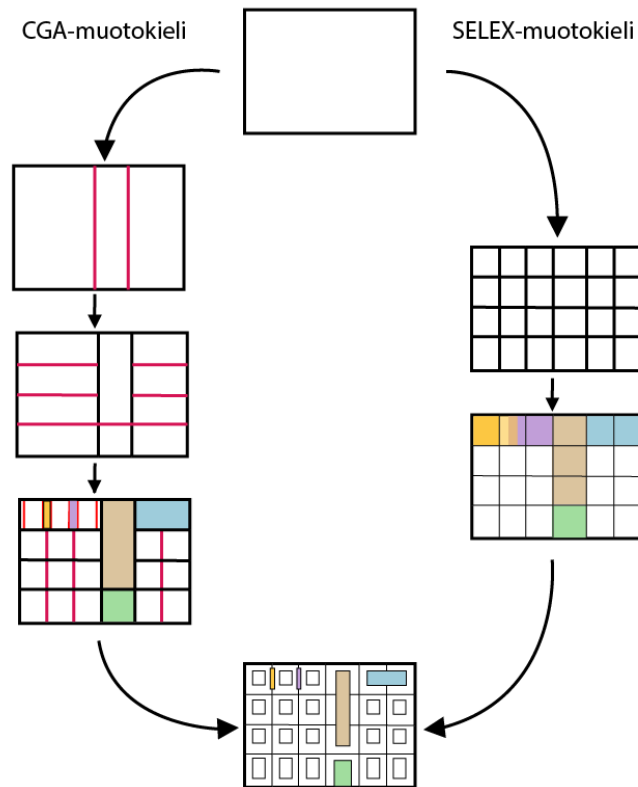
Müllerin ja muiden (2007) systeemi muuntaa yksittäisen syötekuvan teksturoiduksi 3D-malliksi, joka sisältää järkeenkäyvät rakenteet muotosarjana (*engl. shape tree*). Algoritmi pilkkoo julkisivukuvan kerroksiksi ja laatoiksi. Laatta voi sisältää esimerkiksi ikkunan ja sitä ympäröivää seinää. Seuraavaksi laatat jaotellaan pienempiin suorakulmioihin käyttäen jakokielioppeja. Näitä pieniä suorakulmioita verrataan 3D-objekteihin, joita haetaan arkkitehtuurillisten elementtien tietokannasta. Tietokantaan on luotu 2D-elementeistä automaattisesti 3D-objekteja. Tästä saadaan tulosteeksi julkisivun teksturoitu 3D-malli, joka sisältää muotosarjan. (Müller ym. 2007, 2.) Julkisivun elementtejä kuvaava muotosarja ei tosin sisällä syvyystietao, joten käyttäjän tulee muokata elementtien syvyyksiä toivottuihin arvoihin. Se on kuitenkin yksinkertaista, sillä käyttäjä pystyy samanaikaisesti muokkaamaan samankaltaisia elementtejä, kuten ikkunautoja. (Müller ym. 2007, 6.) Tuotettu muotosarja voidaan automaattisesti muuntaa muotokielen sääntösarjaksi, jolloin tyyliä tai osaa siitä voidaan kopioida ja liittää muihin 3D-malleihin (Müller ym. 2007, 7).

Aliagan ja muiden (2007) esimerkissä uusia rakennuksia voidaan nopeasti suunnitella käyttämällä mallirakennuksen dataa ja ne voidaan teksturoida joko fotorealistisesti tai tyylitellysti, mutta rakenteiden tyyli seuraa kuitenkin aina alkuperäistä mallia. Useampi valokuva voi auttaa täyttämään piiloon jääneitä kohtia sekä tasaamaan väri- ja valotusarvoja. Systemin työkalujen avulla käyttäjä pystyy rajaamaan valokuvista pääpiirteisen geometrisen mallin ja jakamaan rakennuksen perusominaisuuksiin, kuten kerroksiin, ikkunoihin ja oviin. Tämän jälkeen systeemi luo automaattisesti muotokielen, joka on sisäistänyt rakennuksessa toistuvat kuviot ja sen erikoispiirteet. Niin kuin Müllerin systeemissä, kyseisen tyylin voi siirtää muotokielen ansiosta taas toisiin jopa erimuotoisiin malleihin. (Aliaga 2007, 1.)

6.3 Valintalauseilla toimiva SELEX-kieli

CGA-kielioppia kehittyneemmän muotokieliopin loivat Jiang, Yan, Zhang ja Wonka (2018), joka perustuu jakamiskielioppien sijaan valintalausekkeisiin. Siinä missä CGA-kieliopin säännöt operoivat vain paikallisesti, SELEX-kieliopin säännöt toimivat globaalissa kontekstissa. Esimerkiksi CGA-kielioppia käyttäessä, rakennuksesta löytyvälle laatalle voitaisiin kutsua sääntö asettamaan siihen tietynlainen ikkuna. Sääntö päättää paikallisesti ikkunan koon ja asennon sekä sen, miten se sopii yhteen kaikkien muiden rakenteiden kanssa. Sen sijaan SELEX käyttäisi tässä tilanteessa globaalia sääntöä, joka kertoo mihin sijoittaa minkilainen ikkuna. (Mts. 1.)

Jakamiskieliopeilla rakennuksen julkisivu jaetaan hierarkkisesti usein kerroksiin, sarakkeisiin ja sitä kautta laattoihin. Tällainen sääntökirjoitus pakottaa rakennuksen jakamisen vain yhdenlaiseen hierarkiaan, joka taas vaikeuttaa useiden sarakkeiden välillä koordinoituvia mallinnusoperaatioita. (Mts. 1.) Koska solujen yhdistäminen ei ole mahdollista, monisoluisia alueita joudutaan muodostamaan suurella määrällä jakosääntöjä. Se vaatii sarjan vuorottelevia vaaka- ja pysty jakoja, mikä on monimutkaista ja vaikeasti ylläpidettävää. Jiangin ja muiden lähestymistavassa julkisivun ruudukosta voidaan valita mielivaltaisesti suorakulmion muotoisia pienempiä aliruudukkoja ja sijoittaa niihin rakennuksen elementtejä. Tästä on hyötyä, mikäli halutaan sijoittaa yksittäisiä elementtejä, jotka ulottuvat useamman ruudukon yli tai sijoittuvat ruudukkojen väliin. Julkisivun kokoa voi muuttaa vaikuttamatta elementtien sommitelmaan. Tämä metodi mahdollistaa siis luonnollisemman ja järkeenkäyvän tavan mallintaa rakennuksia. (Mts. 2.) Viitekehystä pystytään käyttämään myös muihin ihmisten tekemiin objekteihin, kuten työpöytiin, hyllyihin, tuoleihin ja sänkyihin tai vaikkapa parkkipaikkojen sekä puistojen generoimiseen (mts. 10).



Kuvio 3. CGA-kielioppi vaatii paljon jakosääntöjä sopivan alueen muodostamiseen toisin kuin SELEX, jossa julkisivun ruudukosta valitaan haluttu alue kullekin elementille.

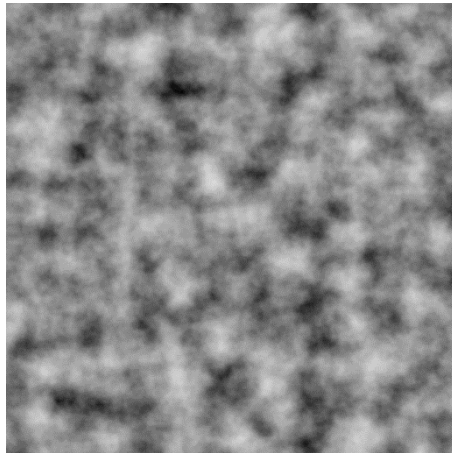
7 Kohinat

Kohina (*engl. noise*) kuvaa matematiikan avulla satunnaisilta vaikuttavista proseduraalisista kuvioista. Yksinkertaisuudessaan siinä kaksi- tai kolmiulotteisessa avaruudessa arvotaan pisteiden paikat ja näiden pisteiden välille muodostetaan uusia pisteitä. Tätä menetelmää kutsutaan interpolaatioksi. Kohinaa voidaan käyttää tekstuurien luomisesta 3D-maastojen generoimiseen. (Perlin 2007; Vivo & Lowe 2015.) Kohinatyyppejä on useita. Seuraavissa kappaleissa käydään nopeasti läpi kaksi kohinatyyppiä.

7.1 Perlin-kohina

Ken Perlin (2007) kertoo luoneensa vuonna 1983 kolmiulotteista avaruutta täyttävän signaalin, Perlin-kohinan, orgaanista teksturointia varten. Sen muodostamista varten hän selventää algoritmiaan, joka yksinkertaisuudessaan sisältää ruudukon luomisen kaksi- tai kolmiulotteiseen tilaan,

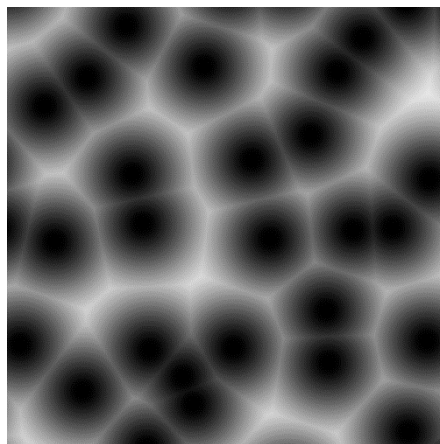
minkä jälkeen pisteitä hajautetaan siihen satunnaisesti. Ruudukon nurkissa olevien pisteiden ja ruudukon sisällä olevien pisteiden arvot interpoloidaan lineaarisesti, josta syntyy kohinaa. (Mt.)



Kuvio 4. Blenderillä generoitua Perlin-kohinaa.

7.2 Worley-kohina

Steven Worley esitteli oman nimikkokohinansa vuonna 1996. Worley-kohinaa kutsutaan myös solukohinaksi (*engl. cellural noise*), koska se luo solumaisia alueita. Kuten edellisessä esimerkissä, ensiksi tilaan hajautetaan satunnaisesti pisteitä. Tämän jälkeen pisteiden etäisyydet lasketaan ja niiden perusteella muodostetaan värialueet. (Vivo & Lowe 2015.) Tällä tekniikalla pystyy tekemään Perlin-kohinasta eroavia alueellisempia kuvioita (Cozzi & Riccio 2012).



Kuvio 5. Blenderillä generoitua Worley-kohinaa.

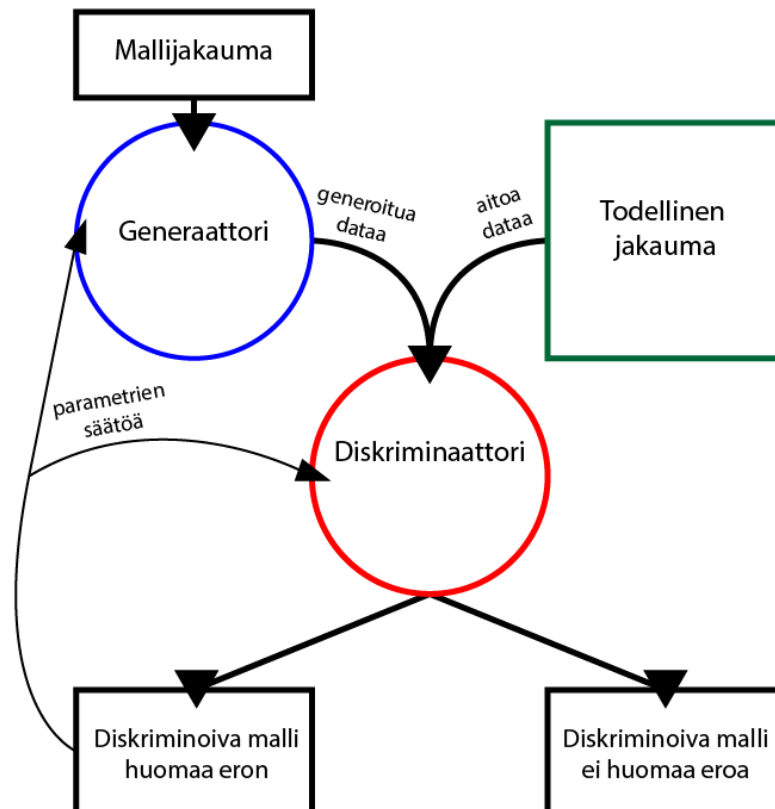
8 Generatiiviset mallit ja koneoppiminen

Aiemmissa luvuissa mainitut proseduraaliset tekniikat perustuvat matematiikkaan, statistiikkaan ja todennäköisyyksiin. Generatiiviset mallit hyödyntävät niiden lisäksi tekoälyn (*engl. artificial intelligence, AI*) koneoppimista (*engl. machine learning*) sisällön tuottamisessa (Generative modeling 2022). Suuri osa oikean maailman käsitteistä on meille itsestään selviä, kuten miltä tietyt eläimet näyttävät tai että elämme kolmiulotteisessa ympäristössä. On kuitenkin vaikeaa kehittää algoritmeja ja malleja, jotka pystyvät analysoimaan ja ymmärtämään suurta määrää maailmaamme liittyvää tietoa (Karpathy ym. 2016).

Generatiivisissa malleissa koneoppiminen käyttää ohjaamatonta oppimista, jonka tavoitteena on havainnoida oikean maailman käsitteitä analysoimalla datassa olevia ilmiöitä (Generative modeling 2022). Ohjaamattomassa koneoppimisessa generatiivista mallia koulutetaan suurella määrällä harjoitusdataa tuottamaan saman tyylistä dataa itse. Mallin neuroverkostoissa (*engl. neural network*) olevat parametrit prosessoivat ja sisäistävät suuresta harjoitusdatan määrästä keskeisimmät ja luonnollisimmat erityispiirteet. Näistä piirteistä neuroverkostot osaavat tuottaa dataa, joka on samanlaista kuin oikean maailman data. (Generative modeling 2022; Karpathy ym. 2016.)

8.1 Generatiivinen kilpaileva verkosto

Generatiivisen mallin vastakohtana on diskriminoiva eli erotteleva malli, joka osaa tunnistaa olemassa olevaa tietoa ja luokitella sitä tunnisteiden mukaan (Generative modeling 2022). Generatiivinen kilpaileva verkosto (*engl. Generative adversarial network, GAN*) on kahden erillisen mallin välinen koulutusprosessi. Siinä diskriminoiva malli yrittää lajitella datan olevan lähtöisin joko todellisesta jakaumasta tai mallijakaumasta. (Karpathy ym. 2016.) Generatiivinen malli sen sijaan yrittää koko ajan luoda aidompaa dataa huijatakseen diskriminoivaa mallia (Generative modeling 2022). Aina, kun diskriminoiva malli huomaa eron kahden jakauman välillä, generoiva malli säätää parametrejaan tuottamaan entistä aidompaa dataa, kunnes se (teoriassa) tuottaa täsmälleen todellista dataa. Kun diskriminoiva malli ei onnistu enää löytämään eroja jakaumien välillä, se päättyy arvailemaan niitä sattumanvaraisesti. (Karpathy ym. 2016.) Tämä koulutusprosessi tehostaa molempia malleja, sillä sen myötä niistä tulee entistä harjaantuneempia tehtävissään (Generative modeling 2022).



Kuvio 6. Generatiivisen kilpailevan verkoston toimintaperiaate.

8.1.1 GAN:n hyödyntäminen realistisemmän maaston luomisessa

Generatiivista kilpailevaa verkostoa voidaan käyttää proseduraalisessa mallintamisessa helpottamaan todenmukaisemman 3D-maaston luomista. Sitä koulutetaan esimerkiksi oikean maailman satelliittikuvilla tai digitaalisilla maastomalleilla (*engl. digital elevation map, DEM*), jotka ovat piste-mäistä geometriatietoa kolmiulotteisessa avaruudessa. Lopputuloksena syntyy näitä syötteitä mukailevia generoituja korkeuskarttoja (*engl. height map*), joissa jokaisen pikselin arvo kuvaa alueen korkeussuhdetta. Korkeuskartat pystytään sen jälkeen muuntamaan helposti 3D-muotoon. (Panagiotou & Charou 2020, 1; Wulff-Abramsson, Rant, Møller & Billeskov 2018, 1.)

Koska oikean maailman ympäristöt sisältävät paljon monimutkaisia piirteitä, kuten kiviä, vuoria ja heinikkoja, realistisen sekä uskottavan maaston luominen digitaalisesti on haastavaa ja se usein tarvitsee ihmisen työpanosta (Voulgaris, Mademlis & Pitas 2021, 1). Edellisessä luvussa mainittuja kohinoita on perinteisesti käytetty generoimaan korkeuskarttoja proseduraalisia maastoja varten. Tosin niiden käytön yleisyys vähentää generoitujen maastojen monipuolisuutta ja voi täten tehdä

maastosta toistuvan ja epäuskottavan tuntuista (Wulff-Abramsson ym. 2018, 2). Lisäksi kohinoita hyödyntävissä maastogeneraattoreissa saattaa esiintyä suorituskyvyllisiä ongelmia, ja ne voivat vaatia työlästä jälkikäsitteilyä, kuten parametrien säätelyä sekä mainittujen ympäristön piirteiden lisäämistä manuaalisesti (Voulgaris ym. 2021, 1).

Wulff-Abramssonin ja muiden (2018, 2) käyttämä generatiivista kilpailevaa verkostoa hyödyntävä maastogeneraattori pystyy heidän mukaansa generoimaan erilaisia ominaisuuksia omaavia ympäristöjä, jotka sisältävät esimerkiksi tundroja, rotkoja ja vuoria. Generoidun maaston teksturointiin he (mts. 8) ehdottavat GAN-verkoston kouluttamista värillisillä satelliittikuvilla ja hyödyntämällä tästä saatua tulostetta. Myöhemmin tehdyssä tutkimuksessa Panagiotou ja Charou (2020, 1) käyttävätkin kahta erillistä GAN-verkostoa, joissa toinen luo digitaalisia maastomalleja 3D-esitystä varten ja toinen RGB-värimallin sisältäviä kuvia luonnolliseen teksturointiin. Värillisten satelliittikuvien pohjalta koulutetun GAN-verkoston generoimissa kuvissa esiintyy luonnollisia elementtejä, kuten joenrantoja ja lunta sekä eriväristä vettä perustuen oletettuun syvyyteen (mts. 7).

Sen sijaan Voulgaris ja muut (2021, 2) esittävät vielä kehittyneemmän GAN-terrain-nimisen systeemin proseduraalisen maaston luomiseen GAN-verkostoilla. Se ei tarvitse syötteeksi korkeuskarttoja, koska se käyttää maaston kohdepisteitä (engl. Point-of-Interest, POI), joita kuvataan geodeettisilla koordinaateilla ja korkeuskulmilla. GAN-verkostoa koulutetaan kuvasynteesiin, jotta se oppii kartoittamaan kohdepisteiden muodostamia korkeuskuvia maanpinnan yksityiskohtia sisältäviksi satelliittikuviksi. Uusia syötteitä koulutetulle neuroverkostolle voidaan luoda nopeasti ja helposti kuvankäsittelyohjelmilla. (Mts. 1–2.) Korkeusarvoja voidaan asettaa vastaamaan tiettyjä RGB-väriarvoja, jotta maanpinnanmuodot saadaan värillisesti näyttämään luonnollisilta ja spatiaalisesti intuitiivisilta (mts. 2–3). On mainittavaa, että tutkimuksessa käytetty koulutusdata on muodostettu hyödyntäen oikean maailman maastodataa. Voulgaris ja muut kertovat kuitenkin testanneen koulutettua GAN-terrainerä epämääräisillä syötekuvilla, kuten nopeasti käsin piirretyillä kuvilla. Jopa niitä se osaa tulkita onnistuneesti korkeuskuviksi. Sen sijaan muut GAN-verkostoa käyttävät maastogeneraattorit vaativat monimutkaisempia syötteitä. (Mts. 2–5.)

Kaikesta huolimatta manuaalisesti luodut maastot ovat silti yksityiskohtaisempia ja luonnollisempia kuin generatiivisilla malleilla luodut. Tarvittavan sisällön määrä ja käyttötarkoitus ratkaisevat,

kannattaako käyttää automatisoitua vai manuaalista lähestymistapaa. Korkea laatu maksaa työvoimaa ja aikaa sekä sitä kautta rahaa. Generatiiviset mallit ja koneoppiminen ylipäätään voivat tulevaisuudessa tuoda suuren edun pienemmän budjetin ja työtiimin omaaville peliyrityksille.

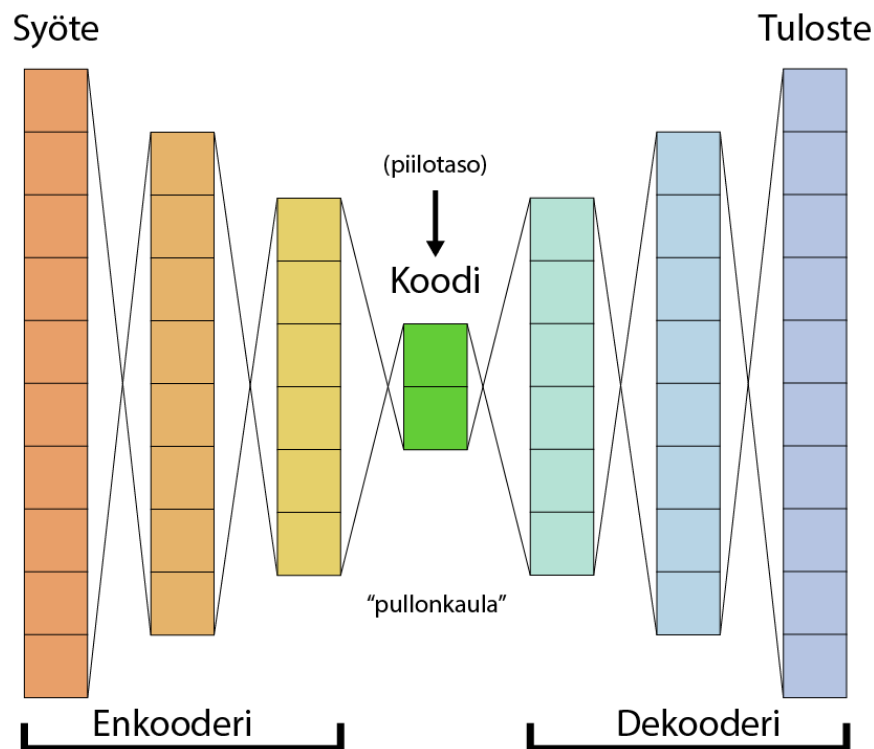
8.1.2 2D-kuvilla tehostettu 3D-malligeneraattori

Tutkimuksessaan Zhu, Xie ja Fang (2018, 1, 3) ehdottavat viitekehystä, jossa on kolme osaa: 3D-malligeneraattori ja diskriminoiva malli sekä tehostajaneuroverkosto. Tehostaja oppii kilpailevalla verkostolla löytämään syötteestä eli eri kuvakulmista otetuista 3D-mallien kuvista niiden olennaimpia piirteitä, jotka syötetään 3D-malligeneraattoriin koulutuksen parantamiseksi. Diskriminoivan mallin tarkoituksena on taas optimoida generaattoria. (Mts. 1–4.) 3D-malligeneraattoria koulutetaan diskriminoivan mallin kanssa yli 50 tuhannella 3D-mallilla 55 kategoriasta, joita ovat esimerkiksi lentokone, auto tai pöytä (mts. 4–5). 3D-malleilla koulutettavaa GAN-verkostoa koulutetaan samaan aikaan 2D-datasta oppivan tehostajaverkoston kanssa. Koulutuksen jälkeen viitekehukseen voi antaa syötteen, josta tehostaja ensin generoi vastaavat kuvaominaisuudet. Tämän jälkeen 3D-malligeneraattori pystyy syntetisoimaan 3D-mallin kuvan ominaisuuksia hyödyntämällä. (Mts. 1.)

8.2 Autoenkooderi neuroverkosto

Autoenkooderi (*engl. autoencoder*) on neuroverkoston tyyppi, jota käytetään ohjaamattomassa oppimisessa ja jonka tavoitteena on luoda syötteestä mahdollisimman identtinen tuloste (Bandyopadhyay 2023; Dertat 2017; Goodfellow, Bengio & Courville 2016, 499). Se ei kuitenkaan pysty eikä sen ole tarkoituskaan tuottaa täydellistä kopiota syötteestä, vaan tehdä tulosteesta arvioltaan samanlainen jälleenrakentamalla syötteestä saatua tietoa. Autoenkoodaria käytetään ulottuvuuden alentamiseen tai ominaisuuksien oppimiseen. (Goodfellow ym. 2016, 499.) Se on hyödyllinen, mikäli halutaan muuntaa dataa korkeaulotteisesta avaruudesta (*engl. high-dimensional space*) matalaulotteiseen avaruuteen (*engl. low-dimensional space*), koska korkeaulotteisessa avaruudessa saattaa olla joitakin ei-haluttuja piirteitä (Bandyopadhyay 2023; Dertat 2017). Koska autoenkooderi joutuu priorisoimaan syötteestä saatuja ominaisuuksia, matalaulotteinen muoto säilyttää alkuperäisestä datasta ainoastaan sen tärkeitä ominaisuuksia (Bandyopadhyay 2023; Goodfellow ym. 2016, 499).

Autoenkooderissa on kolme osaa: enkooderi (syötetaso), koodi eli "pullonkaula" (piilotaso) ja dekkooderi (tulostetaso) (Bandyopadhyay 2023; Dertat 2017; Goodfellow ym. 2016, 499). Syöte kulkee aluksi enkooderin läpi, jolloin se pakataan monta kertaa pienempään matalauloitteiseen muotoon muodostaen koodin (Bandyopadhyay 2023; Dertat 2017). Koodi on piilotasolla sijaitseva tiivistelmä syötteestä, joka muodostaa niin sanotun pullonkaulan kahden neuroverkoston (enkooderin ja dekkooderin) väliin (Bandyopadhyay 2023; Dertat 2017; Goodfellow ym. 2016, 499). Pullonkaula rajoittaa datan kulkemista enkooderista dekkooderiin, jolloin vain tärkein osa datasta pääsee läpi. Sen koko määrittää, kuinka paljon dataa tarvitsee pakata. (Bandyopadhyay 2023.) Dekkooderi sen sijaan purkaa pakatun koodin ja jälleenrakentaa siitä tulosteen sen piilevien ominaisuuksien perusteella (Bandyopadhyay 2023; Dertat 2017; Goodfellow ym. 2016, 499).



Kuvio 7. Autoenkooderi neuroverkoston toimintaperiaate.

8.2.1 Autoenkooderin hyödyntäminen proseduraalisessa mallintamisessa

Vaikka proseduraalinen mallintaminen nopeuttaa ja helpottaa 3D-mallinnuksen työkulkua, voi sitä helpottaa entisestään autoenkooderi neuroverkostoilla. Proseduraalinen malli on systeemi, jota käytetään tuottamaan tietynlaisia objekteja, kuten puita tai kirjahyllyjä, proseduraalisesti. Ne

sisältävät yleensä monia parametreja, joita säätämällä käyttäjä pystyy luomaan ja muokkaamaan generoituja objekteja. (Yumer, Asente, Mech & Kara 2015, 1.) Yumer ja muut (mts. 1) ehdottavat menetelmää, jolla tästä prosessista voisi tehdä intuitiivisempaa käyttäjälle hyödyntäen autoenkooderi neuroverkostoa.

Koska proseduraalisessa mallissa voi olla paljon parametreja ja niiden yhteisvaikutuksia, käyttäjän saattaa olla hankalaa ja työlästä generoida haluttuja muotoja. Käyttäjä ei välttämättä tiedä, millä parametrien arvojen yhdistelmällä saadaan aikaan haluttu visuaalinen lopputulos, ja se voi olla monen kokeilun takana. (Mts. 1.) Tämän sijaan Yumerin ja muiden (mts. 1) metodissa autoenkooderia käytetään alentamaan proseduraalisen mallin parametrien ulotteisuutta, jolloin käyttäjän ei tarvitse säätää yhtä montaa parametria. Autoenkooderia koulutetaan tekemällä ensin proseduraalisen mallin parametreja säätämällä sarja näyteobjekteja, jotka omaavat toivottuja visuaalisia kriteerejä. Näitä näyteobjekteja ja niiden sisältämiä parametriarvoja käytetään sen jälkeen autoenkooderin kouluttamiseen, minkä lopputuloksena saadaan pienempi määrä uusia parametreja. Niiden säätäminen pohjautuu ensisijaisesti visuaalisen ulkomuotoon ja ovat täten käyttäjälle mieleisempiä. (mts. 1–2.) On mahdollista, että lähitulevaisuudessa 3D-mallinnusohjelmiin integroidaan tällainen ominaisuus helpottamaan käyttäjäkokemusta ja ketterää mallintamista. Tämä mahdollistaisi keskivertokäyttäjän tuottamaan nopeasti suuria määriä sisältöä, mikä on tähän mennessä ollut kallista ja aikaa vievää.

9 Aineiston analyysi

Tässä luvussa kuvataan opinnäytetyön aineistonkeruuta ja -analyysia. Aineistoa kerättiin teemahaastattelulla ja kyselyhaastattelulomakkeella.

9.1 Teemahaastattelun tausta ja tulokset

Teemahaastattelua varten tutustuttiin ensin sekundaariaineistoon eli muiden tutkijoiden tekemiin tutkimuksiin aiheesta. Opittua teoriaa ja tutkimuskysymyksiä hyödyntämällä johdettiin teemahaastattelurunko (ks. liite 1). Pääteemana haastattelussa oli:

Proseduraalisen 3D-mallintamisen työnkulun kehittyminen ja sen vaikutus yleisesti pelikehitykseen sekä tarkemmin indie-pelikehitykseen.

Tämän luvun alaluvuissa käydään vuorotellen läpi haastattelun alateemoja:

1. *Haastateltavan suhde aiheeseen*
2. *Indie-pelikehitys*
3. *Proseduraalinen 3D-mallintaminen*
4. *Proseduraalisen 3D-mallintamisen suhde pelikehitykseen*

Jokaisen teeman alla on muutama syventävä apukysymys kyseistä teemasta. Tosin keskustelun annettiin kulkea luonnollisesti, joten osa kysymyksistä saattoi jäädä kysymättä, jos haastateltava vastasi jo kysymykseen. Myös muita teemaan sopivia kysymyksiä saattoi nousta pintaan keskustelun aikana ennalta suunniteltujen kysymysten lisäksi. Haastattelu toteutettiin videopuheluna ja se nauhoitettiin osallistujan luvalla sekä myöhemmin litteroitiin. Litteroitu aineisto lajiteltiin ottamalla talteen tärkeitä nostoja ja koodaamalla kuhunkin teemaan liittyviä yhtäläisyyksiä sekä eroja.

9.1.1 Haastateltavan suhde aiheeseen

Haastateltavaksi tähän opinnäytetyöhön osallistui kokenut pelialalla toiminut ammattilainen. Hän on vuonna 2003 aloittanut pelialalla toimimisen ja silloin kokeillut myös 3D-mallinusoohjelmia ensimmäistä kertaa. Toiminut useamassa firmassa muun muassa Art leadina, 3D-mallintajana ja animaattorina sekä näiden aiheiden opettajana useammassa koulussa. Osaamista löytyy kuitenkin kaikilta pelialan osa-alueilta, kuten johtotehtävistä, ohjelmoinnista, backend-kehittämisestä, grafiikasta sekä peli- ja äänisuunnittelusta.

9.1.2 Indie-pelikehitys

Indie-pelikehityksen suurimmiksi rajoitteiksi koettiin sen hinta, siihen käytetty aika ja työvoiman pienempi määrä. Resurssien puutteessa ei voida käyttää kaikkia työkaluja tai palkata lisää työntekijöitä. Suoritettavia asioita täytyy priorisoida ja työntekijät joutuvat monesti omaamaan useampia taitoja, mikä taas kehittää osaamista. Ilman rahoittajia on vaikea toimia. Rahoittajien puute koettiin toisaalta myös vapaudeksi.

rahoitus on sellainen, että rahasta on vapaus valita mistä ottaa sen.

Suuremmissa yrityksissä kerrotaan rahoittajien olevan enemmän sidoksissa kehitykseen ja työntekijät voivat olla tiukkojenkin salassapitosopimusten alaisuudessa. Näissä yrityksissä yleensä tietty hierarkia sanelee kehityksen kulkua. Indie-pelikehityksessä sen sijaan on enemmän päätösvaltaa kehityksestä, mutta samalla myös enemmän vastuuta sen toteuttamisessa. Vaikka julkaisemisai-kaa ei valvota indie-kehityksessä yhtä tarkkaan on tärkeää asettaa eräpäiviä ja miettiä, mikä on projektin osalta tärkeintä. Monella indie-kehittäjällä kerrotaan olevan vaikeuksia työrytmin asettamisessa, ja että tuotetta jäädään hiomaan loputtomiin.

Indie-kehittäjillä on lisäksi työkalujen valinnan vapaus. Työkalujen kerrottiin olevan ennen huom- noppia ja kalliita. Tämä oli aikanaan iso rajoite. Indie-kehittäjät joutuivat itse kehittämään peli- moottorinsa, mikä on todella vaikeaa ja työlästä sekä vaatii kaikkien pelikehityksen osa-alueiden osaamista. Nykyään on paljon erilaisia ilmaisia pelimoottoreita ja muita työkaluja, jotka ovat kehit- tyneempiä ja helpompia käyttää, vaikkei se silti olekaan yksinkertaista. Niihin löytyy tosin paljon ohjeita internetistä, kursseilta ja koulutuksista, joita ei ollut silloin kuin haastateltava aloitti pe- lialalla. Esimerkiksi haastateltavan paljon käyttämä Blender-mallinnusohjelma on kehittynyt huo- mattavasti ja saanut kehittäjäyhteisön tekemiä lisäosia (*engl. plugin/plug-in/add on*).

Kun indie-kehityksen kehittymisestä kysyttiin, vastattiin rahoitusmallien olevan yksi kehityksistä. Nykyään on joukkorahoitusta eli peliä haluava yhteisö otetaan kehitykseen mukaan heti alussa. Yhteisölle se on tapa saada tiettyyn markkinarakoon sijoittuvia niche-pelejä, joita suuremmat peli- firmat harvemmin tuottavat. Indie-pelikehityksestä on siis kehittynyt myös tapa tuottaa tuotteita, joita ei aiemmin ole ollut markkinoilla. Ennen indie-kehittäjillä ei ollut samanlaisia toimitiloja, pal- velimia tai ilmaisia palveluja, kuten GitHub versionhallintaa varten. Nykyään indie-kehittäjillä on parempia toimitiloja ja pelifirman tietorakenteen järjestäminen on helpompaa.

Sun ei tarvitse set upata sitä koko struktuuria alusta loppuun itse tai pienellä ryh- mällä, saati maksaa jostain omasta dedikoidusta serveristä ja muusta, vaan sä voit ostaa palveluja, voit hankkia halpoja työkaluja.

9.1.3 Proseduraalinen 3D-mallintaminen

Kysyttäessä suhdetta proseduraaliseen 3D-mallintamiseen haastateltava kertoi käyttävänsä päätyökalunaan Blender-mallinnusohjelmaa, johon proseduraalinen mallinnus on tullut liitännäisten tai lisäosien kautta. Ennen tällaisia lisäosia haastateltava kertoo tehneensä proseduraalisia malleja Blenderillä yhdistämällä taulukko- (*engl. array*) ja kaari-modifikaattoreita (*engl. curve modifier*), jolloin pystytään tekemään monimutkaisia muotoja yksinkertaisista muodoista. Lisäosien tultua mainitaan varsinaisten 3D-mallien luomisesta esimerkkinä lisäosia, joilla voi luoda portaikkoja, rakennuksia ja muuta arkkitehtuuria.

Niin paljon kuin koodaajia löytyy ja niin paljon kuin niillä löytyy aina mielenkiintoa tehdä tiettyä asiaa, niin sitten siihen yleensä tuli, ilmestyi jonkun näköinen add-on.

Alun perin Blenderin lisäosanakin ollut *Make Human* mahdollistaa ihmishahmojen luomisen parametrisoidusti, mikä on aloittelijalle intuitiivisempaa ja nopeuttaa luomisprosessia. Haastateltava kertoo nykyään käyttävänsä Unity-pelimoottorin työkaluja ja lisäosia, joilla voi proseduraalisesti luoda kokonaisuuksia, kuten maastoja ja kenttiä. Näistä esimerkeistä voi nähdä yhtäläisyyksiä teoriapohjassa mainittuihin esimerkkeihin, joista monet olivatkin arkkitehtuurin ja maaston luomiseen suunnattuja. Lisäksi haastateltava toteaa proseduraalisen mallintamisen helpottavan ja nopeuttavan 3D-mallien generoimisprosessia, kuten teoriapohjan perusteella on havaittu.

-- artistina kuvittelet mitä sä haluat. Sitten löydät työkalut sen tekemiseen nopeasti. Totta kai kaiken voi [myös] tehdä alusta asti manuaalisesti hyvin hitaasti.

Vaikka haastateltava on opiskellut Blenderin käyttöä jo pitkään, kertoo hän siihen tulevan koko ajan uusia ominaisuuksia ja lisäosia. Mutta jos osaa perustyökalun hyvin, on muiden lisäosien opetteleminen helpompaa. Ei pelkästään riitä, että osaa mallintaa, vaan täytyy osata korjata mahdollisia työnkulun aikana syntyneitä virheitä tai lisäosan käytöstä syntyneitä vikoja. Tekoälyn kehittymisen ennustettiin helpottavan proseduraalista mallintamista entisestään, mutta siinäkin oleellista on ymmärtää, mistä järjestelmä rakentuu.

Jos sä tiedät ne kaikki osat siitä prosessista, niin se lopputuloksen aikaansaaminen ja muokkaaminen, vaikka sulla olisi automaattiset työkalut, niin helpottuu.

Tekoälyn kehittymisen vaikutuksena proseduraaliseen mallintamiseen mainittiin verbalisointi. Nykyään voidaan ensin mallintaa esimerkiksi ihmismalli joko manuaalisesti tai proseduraalisesti ja lisätä sille vaatteet joko manuaalisesti tai lisäosalla mallintamalla tai ostamalla valmiita vaatteita. Tekoäly saattaa pystyä automatisoimaan ja nopeuttamaan näitä vaiheita. Antamalla tekoälylle kuvailevan syötteen se toteuttaa haluaman tuloksen hoitamalla manuaaliset vaiheet käyttäjän puolesta.

9.1.4 Proseduraalisen 3D-mallintamisen suhde pelikehitykseen

Viimeisenä teemana haastattelussa puhuttiin kahden aiemman teeman keskinäisestä suhteesta. Proseduraalisen mallintamisen vaikutus pelikehitykseen nähtiin parhaiten tällä hetkellä asettien osalta eli tässä yhteydessä 3D-malleissa ja niiden tekstuureissa. Kuten jo aiemmin opinnäytetyössä on huomioitu, itseään toistavan luonteen takia proseduraalisen mallintamisen on sen havaittu tuottavan helposti geneerisyyttä. Haastateltava henkilö kuitenkin kertoo hyvin tehdyn proseduraalisen mallintamisen tuntuvan vähemmän yksipuoliselta.

Jos on ammattitaitoinen osaava henkilö, joka käyttää proseduraalisia työkaluja, niin sä [käyttäjä] et näe sitä, että sen [saumattoman toistuvuuden] saa piilotettua sinne.

Proseduraalisen mallintamisen nähtiin antavan etua indie-kehittäjille, mikäli he ovat valmiita panostamaan työkaluihin. Vaikka työkalut ja lisäosat olisivatkin maksullisia, niistä voi saada rahallista etua, koska ne nopeuttavat ja helpottavat työprosessia. Haastateltava kertoo, että proseduraalisten työkalujen käyttämisestä saattaa seurata joillekin huono omatunto, koska tuntee sen olevan huijaamista. Hänen mielestään ne kuitenkin ovat työkaluja siinä missä muutkin ja niiden opetteleminen on tärkeää, jos haluaa pysyä kehityksen mukana.

Työkalut on aikojen kuluessa millä tahansa alalla kehittyneet, niin sama tässä. Pelialan työkalut kehittyvät ja niitä pitää oppia käyttämään. Ne, jotka ei käytä niitä, niin on sitten alakynnessä. He jäävät jälkeen niin sanotusti.

Proseduraalisen mallintamisen haittapuolena pelikehitykseen kuvattiin aiemmin mainittu geneerisyys, mutta kaikki pelaajat eivät välttämättä edes kiinnitä huomiota siihen ja on olemassa tapoja minimoida sitä. Suuremmaksi ongelmaksi koettiin sen sijaan tekoäly ja tekijänoikeudet. Jos tekoälyä käytävä työkalu haravoi informaatiota internetistä ja ei huomioi generoimiseen käytettävän

materiaalin tekijänoikeuksia, se vie pois muilta tuottajilta eikä tuotokselle voi antaa omia tekijänoikeuksia. Haastateltava kuitenkin ajattelee tämän olevan vain alkuaikojen ongelma ja on oikeastaan mielissään siitä, että valtioiden hallitukset joutuvat ottamaan kantaa asiaan.

Kun se ratkaistaan, niin sitten ne työkalut tulee käyttöön.

9.2 Kyselyhaastattelujen tausta ja tulokset

Toisena aineistonkeruumenetelmänä opinnäytetyössä käytettiin kyselyhaastatteluja tukemaan teemahaastattelusta saatavaa aineistoa. Tämä toteutettiin Google Forms-kyselylomakkeen (ks. liite 2) muodossa, jossa on 12 aiheeseen liittyvää kysymystä. Kysymykset johdettiin haastattelu-
rungon pohjalta ja siinä on valmiiden vastausvaihtoehtojen sijaan avoimet vastauskentät. Kyselyn otsikkona toimi opinnäytetyön englanninkielinen nimi:

Advancement of procedural 3D modelling workflow and its impact on indie game development

Kyselyn alussa on ohje vastaajille, jossa kerrotaan kyselyn tavoite ja anonymiteettisyyden turvaaminen. Lisäksi vastaajia on ohjeistettu lukemaan kysymykset huolellisesti ja vastaamaan muutamalla lauseella. Kaikkiin kysymyksiin ei tosin ollut pakko vastata, mikäli vastaaja ei halunnut tai kyennyt siihen vastaamaan. Osa vastaajista jättikin vastauskenttiä tyhjiksi, mikä on ymmärrettävää, kun kyseessä on avoimet vastauskentät ja aiheita, joihin kaikilla ei ole kokemusta.

Kysely toteutettiin englanniksi ja vastaajiksi saatiin kuusi peli- ja grafiikka-alan henkilöä. Vastaajia etsittiin joko lähettämällä kyselylinkki tutkimukselle oleellisiksi koetuille henkilöille tai julkaisemalla linkki Reddit-verkkosivuston aliredditeillä (*engl. subreddit*), jotka ovat yleensä tiettyyn aiheeseen liittyviä foorumeita. Näitä 3D-mallintamiseen, proseduraaliseen generointiin sekä indie-peleihin ja -kehitykseen liittyviä foorumeita olivat *r/3Dmodeling*, *r/proceduralgeneration*, *r/IndieGaming* ja *r/IndieDev*. Tutkimuksen luotettavuuden kannalta ongelmana on sivuston vahva anonymiteettisyys, jolloin vastaajien luotettavuutta on vaikea selvittää. Tämä tosin auttaa eettisyyden kannalta turvaamaan vastaajien henkilöllisyyksiä paremmin. Näistä aiheista keskustellaan enemmän pohdintaluvussa.

Samoin kuin teemahaastattelussa, kun vastaukset oli kerätty, oli aika jäsenellä, teemoitella ja tiivistää aineistoa. Kyselyvastausten keskinäisiä eroja ja yhtäläisyyksiä havainnoitiin sekä lajiteltiin. Alaluvut on jaettu käsittelemään näitä asioita kunkin teeman kohdalta. Niiden välisiä eroja ja yhtäläisyyksiä analysoidaan vahvasti myös teemahaastatteluun ja teoriaan viitaten.

9.2.1 Indie-pelikehityksen rajoitteet, vapaudet ja kehittyminen

Suurimmaksi rajoitteeksi mainittiin rahallisten resurssien puute, mikä olikin ennalta arvattavaa. Rahoituksen puutteellisuuden nähtiin vaikeuttavan työvoiman palkkaamista, kalliiden työkalujen hankkimista ja pelien markkinointia. Vastaajat kertoivat rahoittavansa projektejaan joko oma- tai joukkorahoituksella. Toiseksi yleisin rajoite oli pelin tekemiseen käytettävä aika. Osa indie-kehittäjistä kertoi tekevänsä pelejä vapaa-ajalla ja käyvänsä normaalissa päivätyössä tai koulussa. Yksi henkilö kertoi haasteeksi kilpailun isompia peliyrityksiä vastaan, joihin aiemmin mainitut rajoitteet ovat kytköksissä. Samoja havaintoja tehtiin aiheen teoriaan tutustuessa ja teemahaastattelussa.

Indie-kehityksen vapauksista kysyessä vastaajat ilmoittivat uniikkien ja innovatiivisten pelien kehittämisen ilman ylemmän tahon ja julkaisijan asettamia rajoitteita. Yksi vastaajista tarkensi tätä tietyn markkinaraon täyttämiseksi ja lähempänä suhteena kohderyhmäänsä esimerkiksi sosiaalisen median tai joukkorahoituksen kautta. Näitä käytiin läpi myös teemahaastattelussa niche-pelien ja rahoitusmallien osalta. Yhtä vastaajaa ei haitannut, vaikka hänen pelinsä eivät tuottaisi mitään, kunhan saa tehdä pelejä, joista itse pitää ja päättää samalla työtahtinsa nopeuden. Tällöin voi päätellä kyseessä olevan henkilö, joka tekee projekteja enemmän harrastusmielessä. Teemahaastattelussa puhuttiin työrytmin tärkeydestä, mikäli haluaa julkaista pelinsä tietyssä ajassa isommalle yleisölle.

Indie-pelikehitys koettiin parantuneen työkalujen, jakelualustojen ja joukkorahoituksen saatavuuden puolesta. Vastauksissa lueteltiin samoja ilmaisia pelimoottoreita ja ohjelmistoja sekä jakelualustoja kuin tietoperustassa ja teemahaastattelussa. Näistä tärkeimmät olivat Unity- ja Unreal Engine-pelimoottorit, Blender- ja Houdini-mallinnusohjelmat, Steam- ja Itch.io-jakelualustat sekä Kickstarter- ja Indiegogo-rahoituspalvelut. Vastauksissa toistettiin myös sama väite siitä, että indiepelit olivat ennen huonompilaatuisia ja niiden kehittämiseen vaadittiin laajempaa osaamista. Mainittujen kehityskohteiden voidaan siis ajatella helpottaneen ja parantaneen indie-pelikehitystä.

9.2.2 Proseduraalinen 3D-mallintaminen

Suurin osa vastaajista ei ollut aktiivisesti käyttänyt proseduraalista 3D-mallintamista, mutta tiesivät sen olemassaolosta. Yksi vastaajista kertoi käyttäneensä Blenderin proseduraalista mallinnusta rakennusten sekä kasvistojen luontiin, ja toinen generoinut Unreal Enginellä maaston ja tekstuuria. Nämä ovatkin huomattu opinnäytetyön aikana yleisimmiksi kohteiksi käyttää proseduraalista mallintamista.

Proseduraalisen mallintamisen vaikutusta 3D-graafikon työtehtäviin mietittiin manuaalisen mallintamisen vähenemisenä. Aikaa jää enemmän muihin graafikon tehtäviin, kun yksinkertaisemmat työvaiheet automatisoituvat. Yksi vastaajista huomioi graafikoiden joutuvan opettelemaan uutta, vaikka se kuuluukin työnkuvaan.

Graphic designers must learn how to use new workflows and tools which can be challenging for a beginner but designers have to learn new things all the time anyways when technology gets better and better.

Proseduraalisen mallintamisen nähtiin kehittyneen vähän samoilta osa-alueilta kuin indie-kehityksenkin. On enemmän siihen erikoistuneita työkaluja, kuten Houdini, ja niitä on helpommin saatavilla toisin kuin ennen. Työkaluja on myös integroitu eri pelimoottoreihin. Tietokoneiden kasvanut laskentateho helpottaa suorituskyvylisesti vaativimmissa tehtävissä eli pystyy esimerkiksi renderöimään suuria rakennelmia nopeasti tai tekemään realistisempia ja yksityiskohtaisempia malleja. Yksi kehityskohdista oli myös koneoppimisen lisääntynyt käyttö ja sen sisällyttäminen eri ohjelmistoihin. Tämän mainittiin vaikuttavan prosessin automatisointiin työvaiheiden osalta, mikä taas mainittiin teemahaastattelussa verbalisoinnilla ja tietoperustassa useilla esimerkeillä. Yksi vastaajista kuvaili, että esimerkiksi ihmismallin voisi luoda verbaalisesti.

Users may be able to create models by writing a descriptive input. AI can help in modeling complex things e.g people.

9.2.3 Proseduraalisen mallintamisen vaikutus pelialaan

Proseduraalisen mallintamisen kerrotaan vaikuttavan pelikehitykseen siten, että voidaan luoda laajoja alueita nopeasti, kuten maastoa tai kaupunkeja. Tämä on etenkin indie-ryhmälle hyödyksi

pienillä resursseilla toimiessa. Toisaalta, vaikka yksi vastaajista huomioikin proseduraalisen mallintamisen vähentävän kuluja ja sen mahdollistavan joidenkin asioiden nopeampaa mallintamista, hän ei usko sen yksinään poistavan indie-kehityksen rajoitteita tai antavan indie-kehittäjille suurta etulyöntiasemaa pelimarkkinoilla.

Just as indie developers use it, so do bigger game companies.

Proseduraalisen 3D-mallintamisen haittapuolia pelikehitykseen nähtiin geneerisyys myös kyselyvastaajien kesken. Manuaalisen mallintamisen todettiin tuottavan aina uniikimpaa jälkeä kuin proseduraalinen ja joidenkin tiettyjen asioiden mallintaminen saattaa olla hankalaa, jos ei löydy niiden generoimiseen tarkoitettua työkalua. Vaikka proseduraalisia työkaluja on paljon, ne on yleensä kehitetty luomaan jotain tiettyä asiaa. Yksi vastaajista mainitsee myös uusien proseduraalisten työkalujen ja -prosessien harjoittelun haittapuolena, koska se voi hidastaa projektin etenemistä. Tosin kuten aiemmin todettiin, kuuluu se työkuvaan. Sen sijaan tekoälyn haittapuolia ei huomioitu kyselyvastaajien kesken ollenkaan toisin kuin teemahaastattelun henkilö teki.

10 Johtopäätökset

Tässä opinnäytetyössä lähdettiin tutkimaan proseduraalisen 3D-mallintamisen työnkulun kehittymistä ja sen vaikutus indie-pelikehitykseen. Näiden tavoitteiden kautta johdettuun tutkimusongelmaan ja tarkemmin tutkimuskysymyksiin pyrittiin löytämään vastauksia kerätyn aineiston pohjalta. Tutkimuskysymykset ovat vahvasti kytköksissä toisiinsa, jolloin niitä on vaikea analysoida erillään. Luvun kappaleet kulkevat kuitenkin suurin piirtein kronologisessa järjestyksessä tutkimuskysymyksestä seuraavaan. Tietoperustan sekundaariaineistosta etsittiin vastausta ensimmäiseen ja osittain toiseenkin tutkimuskysymykseen:

1. *Mitä erilaisia työnkulkuja proseduraaliselle 3D-mallintamiselle on olemassa?*
2. *Miten proseduraaliset 3D-mallintamisen tavat ovat kehittyneet ja tulevat kehittymään?*

Proseduraaliselle 3D-mallintamiselle on olemassa lukuisia työnkulkuja ja tekniikoita. Tässä opinnäytetyössä niitä on kuvailtu aiemman tutkimustiedon avulla vain osa. Yhdeksi yleisimmäksi mallintamistavaksi havainnoitiin formaalit kieliopit, joihin lukeutuvat muotokieliopit ja L-systeemit.

Muotokielioppeista opinnäytetyössä käytiin läpi CGA- ja SELEX-muotokieliä, joista myöhempi on kehittyneempi vaihtoehto. Niitä selitettiin pääasiassa arkkitehtuurin luomista helpottavina työkaluina, mutta kielioppeja voi käyttää myös moniin muihin käyttötarkoituksiin. Perinteinen L-systeemi kuvaa hyvin kasviston luonnollista kasvua, mitä varten se alun perin kehitettiin.

Fraktaalien ja kasvien lisäksi opinnäytetyössä annettiin esimerkki L-systeemillä tehdystä tieverkostosta, joka haarautuu alkupisteestä laajalle alalle kasvin tavoin. L-systeemeistä mainittiin myös myöhemmin tulleet parametriset ja stokastiset systeemit, joilla voi mallintaa monimutkaisempaa geometriaa, kuten rakennuksia.

Kohinoista opinnäytetyössä puhuttiin Perlin- ja Worley-kohinoiden osalta, joilla voidaan pääasiassa tehdä proseduraalista maastoa ja tekstuureja. Maaston luomisen osalta tämä koettiin tosin vanhanaikaiseksi ja vähemmän realistiseksi vaihtoehdoksi generatiivisella mallilla tuotettuun maastoon verrattuna. Koneoppia hyödyntäviä generatiivisia malleja työssä mainittiin generatiivinen kilpaileva verkosto ja autoenkooderi neuroverkostot. Näistä ensimmäinen soveltuu kohinoita tarkemman sekä realistisemmän maaston luontiin ja vaatii vähemmän manuaalista työpanosta. Maaston lisäksi GAN:lla pystyy luomaan 2D-kuvasyöttestä 3D-malleja. Autoenkooderia käyttävä esimerkki kuvaili sen mahdollista hyödyntämistä proseduraalisten työkalujen käytön helpottamiseen. Pelkästään jo näistä esimerkeistä voi nähdä, miten proseduraaliset työnkulut ovat kehittyneet, sekä ennustaa niiden tulevaa kehittymistä tekoälyn kohdalla. Nämä esimerkit ovat vain pieni osa siitä, mihin generatiivisia malleja voidaan käyttää ja uusia parannuksia tehdään koko ajan.

Haastatteluja analysoidessa proseduraalisen mallintamisen käyttökohteet vastasivat hyvin tietopuoleissa kuvattuja esimerkkejä. Haastateltavat kertoivat käyttäneensä proseduraalista mallintamista nimenomaan maastojen, arkkitehtuurin ja muiden objektien mallintamiseen. Lisäksi teema-haastattelussa tuli puheeksi ihmisten mallintaminen proseduraalisesti, mitä teoriapohjassa ei olla huomioitu. Tarkkojen teoriakäsitteiden ja työnkulkujen sijaan haastateltavat puhuivat proseduraalisesta mallintamisesta yleisemmällä tasolla. Käyttötavat olivat ensi sijassa kytköksissä eri mallinussohjelmistojen ja pelimoottorien lisäosiin, joiden toimintatapoja ei avattu syvemmin.

Haastatteluista kävi ilmi proseduraalisten työkalujen kehittyneen ja halventuneen huomattavasti viimeisten vuosikymmenten aikana. Niiden kehittyessä myös käyttäjän täytyy oppia uusia työkaluja ja -menetelmiä, jos ei halua jäädä jälkeen. Haastateltavien kesken proseduraalisten työkalujen

oletettiin kehittyvän tekoälyn rinnalla. Tekoälyllä ennustettiin pystyvän mallintamaan objekteja antamalla kirjallisia kuvailevia käskyjä, mikä vähentäisi manuaalista mallinnusta huomattavasti. Teoriapohjan esimerkkien ja haastateltavien yhteisen mielipiteen perusteella voi päätellä tekoälyn olevan suurin vaikuttaja kehityksessä proseduraalisen mallintamisen osalta, ainakin tällä hetkellä. Siitä huolimatta tekoäly on proseduraalisen mallintamisen osalta vielä harvinaista, joten nämä päätelmät ovat osittain spekulatioita. Joissakin tapauksissa tekoälyn käyttämisessä todettiin olevan myös laillisia ongelmakohtia. Tästä päästäänkin opinnäytetyön kolmanteen tutkimuskysymykseen:

3. Miten proseduraalisen 3D-mallintamisen työnkulun kehittyminen on vaikuttanut ja tulee vaikuttamaan indie-pelikehitykseen?

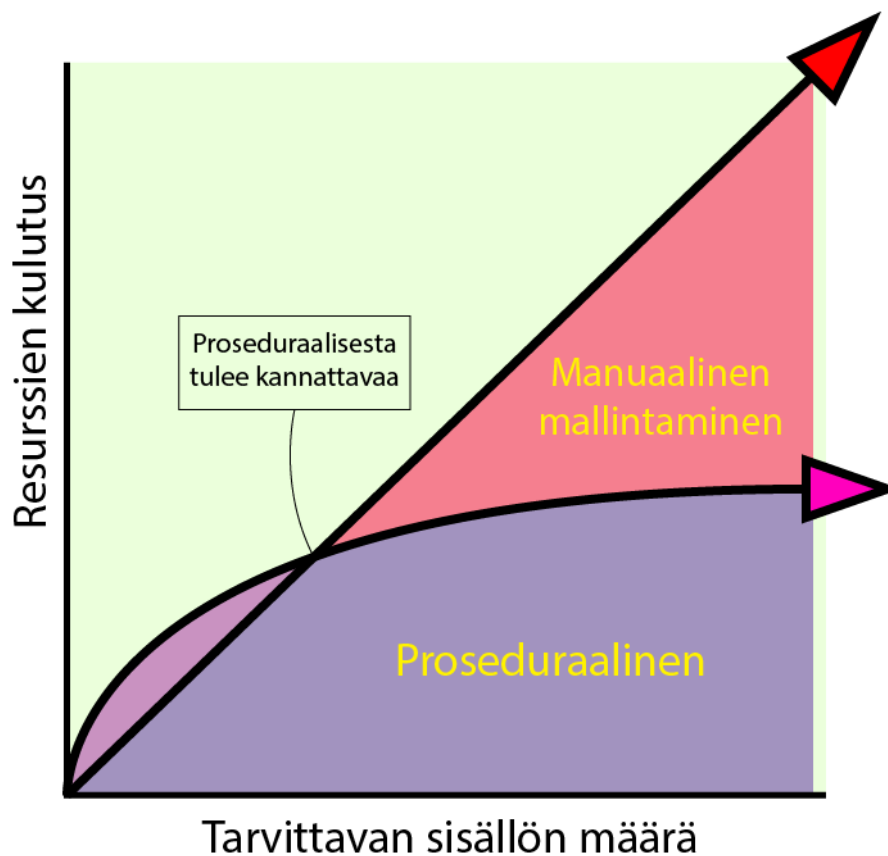
Ennen kuin proseduraalisen mallintamisen kehittymisen vaikutuksia indie-pelikehitykseen voi tulkita, tulee indie-kehityksen rajoitteita ja kehittymistä ensin avata. Tietoperustassa ja haastattelussa indie-kehityksen suurimmaksi rajoitteeksi selostetaan resurssien puute, mikä tarkoittaa toisin sanoen rahoituksen puutetta. Rahallisten resurssien puute hankaloittaa muun muassa työvoiman palkkaamista, työkalujen hankkimista ja markkinointia. Ajan käyttäminen mainittiin haastatteluissa toiseksi suurimmaksi rajoitteeksi.

Kehittymistä nähtiin parempien ja halvempien työkalujen saatavuutena sekä joukkorahoitusmallien lisääntymisellä. Tämä mahdollistaa indie-pelikehittäjien pääsevän helpommin käsiksi proseduraalisiin työkaluihin. Kun proseduraalisia työkaluja hyödynnetään indie-kehityksessä, säästetään aikaa ja työvoimaa. Nämä resurssit todettiin olevan osa suurimpia rajoitteita indie-kehityksessä. Proseduraalisella mallintamisella pystyy luomaan asioita, jotka eivät olleet aikaisemmin mahdollisia itsenäisille pelikehittäjille eli suuria kokonaisuuksia ohittaen työläitä itseään toistavia vaiheita. Kun tämä yhdistetään siihen, että indie-pelit suuntautuvat erilaisiin markkinarakoihin verrattuna suurempiin peliyrityksiin, voi olettaa niiden markkinamahdollisuuksien nousevan entisestään.

Tulee kuitenkin jälleen huomioida, ettei proseduraalinen mallintaminen ratkaise kaikkia indie-kehityksen rajoitteita eikä se tule ongelmitta. Suurin ongelma todettiin haastatteluissa geneerisyyden lisääntymisen ja yksityiskohtaisten objektien mallintamisen kohdalla. Toiseksi suurin ongelma-kohta oli tekoäly, mikä toisaalta mainittiin myös suurimmaksi kehityksen ajajaksi. Ongelmana teko-

älyssä tuli vastaan sen tekijänoikeuksiin liittyvät ongelmat. Byrokratia liikkuu hitaammin kuin teknologia kehittyä, jolloin huimaa vauhtia kehittyvän teknologian hyviä ja huonoja puolia ei valvota lailla tarpeeksi nopeasti.

Manuaalinen mallintaminen on tehokasta, mutta se on aikaa vievää ja vaatii teknisiä taitoja. Lisäksi työnkulussa on paljon vaiheita, jotka ovat avoinna virheille. Proseduraalisen mallintamisen ja tekoälyn hyödyntäminen vähentää tällaisten kalliiden inhimillisten erehdysten tapahtumista sekä säästää aikaa, jolloin se mahdollistaa riskialttiimpien luovien ratkaisujen tekemistä. Loppujen lopuksi proseduraalinen mallintaminen ei poista tarvetta ihmiselle. Se koostuu monista eri metodeista ja niitä yhdistämällä käyttäjä pystyy hyödyntämään sitä työkaluna ketterässä sisällöntuotannossa. Proseduraalisen mallintamisen vaikutus indie-pelikehitykseen riippuu pienemmällä mittakaavalla projektin tarpeista sekä graafikkojen taidoista. Pelikehittäjän on siis oleellista pohtia, hyötykö projekti proseduraalisen mallintamisen käyttämisestä.



Kuvio 8. Proseduraalisen mallintamisen hyötysuhde tarvittavan sisällön määrään.

11 Pohdinta

Tässä luvussa tarkastellaan opinnäytetyön onnistumista kokonaisuudessaan. Tutkimusprosessin aikana kertynyttä tietoa yhdistetään tuloksiin ja niihin vaikuttaneita tekijöitä tarkastellaan. Lisäksi käsitellään tutkimuksen eettisyyttä ja luotettavuutta sekä pohditaan mahdollisia jatkotoimenpiteitä.

11.1 Ajatuksia tutkimuksen onnistumisesta

Opinnäytetyön tavoitteena oli selvittää, mitä proseduraalinen 3D-mallintaminen on ja miten se on kehittynyt, sekä vaikuttaako sen kehittyminen itsenäisten pelifirmojen tuotekehitykseen ja toimintatapoihin. Taustatietoa proseduraalisesta mallintamisesta ja indie-pelikehityksestä on kerätty useista englanninkielisistä tieteellisistä julkaisuista, mutta osittain myös aiheeseen liittyvistä blogeista, artikkeleista ja ohjekirjoista. Tutkimusasetelman kuvaamisessa hyödynnettiin suomenkielistä tutkimuskirjallisuutta. Luotettavuuden kannalta on pyritty käyttämään relevantteja ja luotettavia lähteitä, minkä takia suurin osa lähteistä on tutkimuspapereita. Blogijulkaisujen ja artikkelien suhteen on käytetty tiukempaa lähdekriittisyyttä ja niitä on pyritty käyttämään mahdollisimman vähän. Tarkemmin tiedonhausta (2.4.1 *Tiedonhaku*) ja lähdekriittisyydestä (2.5 *Luotettavuuden varmistaminen*) kerrottiin tutkimusasetelmassa.

Sekundaariaineiston pohjalta saatiin näkökulmia proseduraalisen mallintamisen erilaisiin tekniikoihin sekä niiden kehittymiseen ja käyttötarkoituksiin 3D-mallintamisessa. Vahvistusta näihin näkökulmiin sekä uusia näkökulmia saatiin haastatteluaineistoa keräämällä ja analysoimalla. Tietoperustassa proseduraalista mallintamista käytiin läpi syvällisesti tarkkojen teoriakäsitteiden kautta, mutta haastatteluissa siihen otettiin käytännönläheisempi asenne kokemusten pohjalta. Proseduraaliseen mallintamiseen ei tietoperustassa olisi välttämättä tarvinnut syventyä niin yksityiskohtaisesti vaan antaa enemmän esimerkkejä erilaisista proseduraalisista mallinnusohjelmista ja lisäosista sekä viitata niiden osalta tutkimuskysymyksiin. Tietoperustassa olisi tullut puhua enemmän indie-pelikehityksestä, jotta proseduraalisen mallintamisen vaikutuksia olisi ollut selkeämpää peilata siihen. Erilaisia esimerkkejä indie-peleistä, jotka käyttävät proseduraalisia 3D-malleja, olisi myös ollut hyödyllistä sisällyttää.

Tutkimusta varten haastateltiin pelialan asiantuntijaa, kellä on vuosikymmenten kokemus alalta sekä aiheen opettamisesta. Teemahaastatteluun valmistautuminen ja sen toteutus koettiin onnistuneeksi. Teoreettiseen viitekehykseen oli paneuduttu huolella ennen haastattelua ja keskustelun aikana osattiin olla näyttämättä tutkijan omia mielipiteitä. Toisaalta jotkin haastattelurungon kysymyksistä saattoivat olla hieman johdattelevia, koska niihin oletettiin saatavan tietynlainen vastaus. Proseduraalisen mallintamisen kehittymisen ja sen vaikutukset indie-pelikehitykseen osattiin odottaa olevan pääpiirteiltään positiivista, vaikka positiivisia vaikutuksia ei erikseen kysytty. Kaikesta huolimatta haastattelu oli menestys ottaen huomioon haastattelijan kokemuksen puutteen. Se oli opinnäytön kannalta hyödyllinen sekä oleellinen osa koko tutkimusta. Vaikka haastateltava henkilö on kokenut ja sopi hyvin tutkimuksen kohderyhmäksi, olisi haastateltavia kaivattu enemmän. Olipa haastateltava kuinka hyvä, se on vain yhden ihmisen näkökulma. Haastateltavien vähyyteen vaikutti sopivien henkilöiden vaikea saatavuus, koska haastateltavalla tuli olla kokemusta tutkittavista aiheista. Haastattelupyyntöihin ei usein vastattu tai vastattiin kielteisesti. Toisaalta kohderyhmää olisi voinut etsiä aktiivisemmin ja sisällyttää siihen vähemmän kokeneita henkilöitä.

Aiemmin mainituista syistä aineistoa päätettiin kerätä myös tekemällä kyselyhaastatteluja. Kyselylomakkeeseen vastasi kuusi henkilöä, jotka toimivat joko indie-pelialalla tai 3D-grafiikan parissa. Henkilöiden anonymiteettisyys tosin vähentää tutkimuksen luotettavuutta, mutta lisää henkilöiden tietoturva. Teema- ja kyselyhaastattelujen välillä oli kuitenkin paljon samankaltaisia vastauksia, joten ne vahvistivat hyvin toisiaan. Haastatteluilla saatiin yleinen käsitys siitä, miten proseduraalinen mallintaminen ja indie-pelikehitys ovat kehittyneet alkuaikojen ongelmista, ja mitä vaikutuksia niiden synergialla on. Pääsääntöisesti proseduraalinen mallintaminen ja indie-kehitys ovat helpottuneet ja parantuneet huomattavasti. Proseduraalisen mallintamisen nähtiin helpottavan niin itsenäistä kuin suuremman luokan pelikehitystä.

Tutkimuskysymyksiin ja -ongelmaan löydettiin kuitenkin vastauksia tietoperustaa ja tutkimusaineistoa hyödyntäen. Opinnäytetyön lopputuloksena syntyi laadullinen tutkimus, joka auttaa hahmottamaan proseduraalisen 3D-mallinnuksen työnkulun vaikutuksia etenkin indie-kehittäjien näkökulmasta sekä arvioi proseduraalisen mallinnuksen mahdollisuuksia ja hyötyjä tulevaisuudessa. Proseduraaliseen 3D-mallintamiseen löytyy useita tekniikoita vaihdellen formaaleista kieliopista tekoälyä hyödyntäviin generatiivisiin malleihin. Ammattilaiset ovat kehitelleet erilaisia lähestymis-

tapoja proseduraaliselle mallintamiselle, minkä pohjalta toiset henkilöt ovat jatkaneet niiden kehittämistä. Myös kokonaan uusia mallintamistaktiikoita on syntynyt hyödyntämällä olemassa olevia teknologioita, kuten neuroverkostoja. Olemassa olevia proseduraalisia työkulkuja parannelaan koko ajan ja uusia työkulkuja tullaan näkemään tulevaisuudessa lisää teknologian kehittyessä.

Indie-pelikehityksessä havainnoitiin olevan tiettyjä rajoitteita, joita proseduraalisella mallintamisella pystyttäisiin helpottamaan. Proseduraalisen mallintamisen hyödyt vähentävät graafikkojen työtaakkaa ja vapauttavat täten ajallisia sekä rahallisia resursseja muihin pelikehityksen osa-alueisiin. Proseduraalisen mallintamisen suurimpina ongelmakohtina nähtiin kuitenkin sen mahdollinen geneerisyys ja tekoälyllä tehtyjen objektien tekijänoikeudet sekä se, mihin sitä voi ylipäättään käyttää. Proseduraaliset työkalut rajoittuvat yleensä vain tietynlaisten objektien mallintamiseen ja riippuvat hyvin paljon projektin tarpeista, jolloin manuaalinen mallintaminen voisi joissain tilanteissa olla parempi vaihtoehto. Vaikka proseduraalinen mallintaminen ei sovi kaikkiin indie-projekteihin eikä poista kaikkia rajoitteita pelikehityksessä, voi siitä olla suuri hyöty tietynlaisissa tilanteissa.

11.2 Eettisyys

Opinnäytetyön aiheen ja tutkimusmenetelmien ei koettu aiheuttavan eettisiä ristiriitoja. Ennen tutkimuksen aloittamista aihe-ehdotus hyväksyttiin Jyväskylän ammattikorkeakoulun toimesta. Haastateltavien henkilöiden anonymiteettisyys turvattiin eikä niitä voi selvittää työn pohjalta. Tämä on etenkin kyselyhaastatteluun vastanneiden kesken varmaa, koska tutkija ei itsekään tiedä heidän henkilöllisyyksiänsä. Teemahaastattelun video- ja äänimateriaali sekä haastattelun litterointi säilöttiin tietoturvallisesti Jyväskylän ammattikorkeakoulun palvelimelle ja Jamkin tarjoamalle Microsoft OneDrive pilvipalveluun. Aineistoa ei luovutettu eteenpäin ja ne poistuvat säilytyksestä, kun käyttäjä poistuu Jamkista. Kyselyhaastattelujen tuloksia säilöttiin kirjoittajan Google-tilillä, josta ne poistetaan opinnäytetyön julkaisun yhteydessä. Google-tili ei ole yhtä tietoturvallinen kuin muut mainitut säilytystilat, mutta kyselyjen vastaukset eivät sisällä henkilötietoja tai arkaluontoista materiaalia.

11.3 Luotettavuus

Luotettavuuden varmistamista käytiin läpi jo samannimisessä luvussa 2.5. Siinä puhuttiin muun muassa lähteiden huolellisesta merkkauksesta ja käytettyjen julkaisujen lähdekriittisyydestä. Koko tutkimuksen osalta on tärkeää tarkastella tutkijan subjektiivisuutta. Tutkija pyrki olemaan objektiivinen, mutta inhimillisistä syistä se ei ole täysin mahdollista ja sen myöntäminen on oleellinen osa luotettavuutta. Vaikka ennen tutkimuksen aloittamista tutkittavat aiheet eivät olleet tutkijalle tuttuja ja ennako-oletuksia ei ollut, syntyi niitä joltain osin tutkimusprosessin aikana. Esimerkiksi proseduraalisesta mallintamisesta oletettiin olevan hyötyä indie-pelikehityksessä. Tutkimusprosessin kautta saatiin kuitenkin ennakkoluuloihin uusia sekä erilaisia näkökulmia ja opittiin paljon uutta aiheesta.

Validiteetin kannalta tutkimusmenetelmien valinnat koettiin opinnäytetyön kannalta sopiviksi ja niiden valintaa perusteltiin. Tutkimuksen luotettavuus kärsi tutkijan ajanhallintaongelmien vuoksi, koska tutkimuksen pituus venyi odotettua pidemmäksi. Vaikka aikaa käytettiin paljon, jäi tutkimuksen kohderyhmä pieneksi. Aineisto antaa vain rajallisen määrän näkökulmia, joista yleistyksiä tehtiin. Tutkimukseen valitut henkilöt olivat kuitenkin tutkimuksen kannalta sopivia, vaikka otos olikin pieni.

Tulosten toistettavuutta eli reliabiliteettia pohtiessa voidaan todeta joidenkin tutkimustulosten olevan toistettavissa ja osan ei. Muut samaa ilmiötä tarkastelevat tutkimukset voivat saada samoja tuloksia ainakin proseduraalisen mallintamisen tavoista ja niiden kehittymisestä. Tulokset proseduraalisen mallintamisen vaikutuksista indie-pelikehitykseen voivat sen sijaan vaihdella. Laadullisten tulosten uudelleen tuottaminen on hankalaa, koska ne ovat osittain ihmisten henkilökohtaisia näkemyksiä. Samoin pysyvyyden määrittäminen on monimutkaista, sillä tutkitaan muuttuvaa ilmiötä. Osa havainnoista tulee vanhentumaan ja ennustukset tekoälystä eivät saata pitää paikkaansa, mutta saadaan tutkimuksesta kaikesta huolimatta tietyn aikakauden kuvaus tutkittavasta aiheesta.

11.4 Jatkotutkimus

Tutkimuksen kohteen valintaa ja sen hyödyntämistä perusteltiin luvussa 2.2 *Opinnäytetyön merkitys*. Proseduraalisen mallintamisen mahdolliset hyödyt havainnoitiin olevan joissakin tilanteissa

indie-pelialaa edistäviä ja kerättyä tietoa voivat hyödyntää kyseisellä alalla toimivat henkilöt. Myös aiheen ajankohtaisuus ja sen jatkuva kehittyminen mainittiin perusteluiksi. Mikäli samasta aiheesta halutaan tehdä jatkotutkimuksia, on oleellista toteuttaa laajempaa aineistonkeruuta- ja analyysia. Kuten aiemmassa alaluvussa mainittiin, opinnäytetyö antaa vain tietynlaisia näkökulmia tämänhetkisestä ajasta. Jatkotutkimuksen voi nähdä hyödylliseksi, kun uusia proseduraalisia työkaluja kehitetään ja ne ovat saatavilla indie-pelikehittäjille.

Lähteet

Alasuutari, P. 2011. Laadullinen tutkimus 2.0. uud. p. Tampere: Vastapaino.

Aliaga, D. G., Rosen, P. A. & Bekins, D. R. 2007. Style Grammars for Interactive Visualization of Architecture. Viitattu 17.4.2023. https://www.researchgate.net/publication/6337303_Style_Grammars_for_Interactive_Visualization_of_Architecture.

Bai, J., Pan, J., Wang, J. & Wang, S. 2020. Flower Factory: A Component-based Approach for Rapid Flower Modeling. IEEE International Symposium on Mixed and Augmented Reality (ISMAR). Viitattu 26.4.2022. <https://ieeexplore.ieee.org/document/9284720>.

Bailey, D. 2020. How many Minecraft players are there? PCGamesN. Viitattu 27.4.2022. <https://www.pcgamesn.com/minecraft/minecraft-player-count>.

Bandyopadhyay, H. 2023. Autoencoders in Deep Learning: Tutorial & Use Cases [2023]. V7labs. Viitattu 16.3.2023. <https://www.v7labs.com/blog/autoencoders-guide>.

Blatz, M. & Korn, O. 2017. A Very Short History of Dynamic and Procedural Content Generation. Game Dynamics. Viitattu 16.6.2022. https://www.researchgate.net/publication/315863952_A_Very_Short_History_of_Dynamic_and_Procedural_Content_Generation.

Bubriski, J. 2016. Procedural Generation 101 (for games). Viitattu 26.4.2022. <https://johnnycode.com/2016/11/08/procedural-generation-101-for-games/>.

Cheshire, T. 2014. Changing the game: how Notch made Minecraft a cult hit. Wired. Viitattu 27.4.2022. <https://www.wired.co.uk/article/changing-the-game>.

Cozzi, P. & Riccio, C. 2012. OpenGL Insights. CRC Press. Viitattu 8.6.2022. https://books.google.fi/books?id=CCVenzOGjpcC&pg=PA113&redir_esc=y#v=onepage&q&f=false.

Dertat, A. 2017. Applied Deep Learning - Part 3: Autoencoders. Towards Data Science. Viitattu 16.3.2023. <https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798>.

Dormans, J. & Leijnen, S. 2013. Combinatorial and exploratory creativity in procedural content generation. Viitattu 12.6.2022. http://www.fdg2013.org/program/workshops/papers/PCG2013/pcg2013_1.pdf.

Eskola, J. & Suoranta, J. 2003. Johdatus laadulliseen tutkimukseen. uud. p. Tampere: Vastapaino.

Generative modeling. 2022. TechTarget. Viitattu 13.3.2023. <https://www.techtarget.com/searchenterpriseai/definition/generative-modeling>.

Goodfellow, I., Bengio, Y. & Courville, A. 2016. Deep Learning. MIT Press. Viitattu 16.3.2023. <https://www.deeplearningbook.org/contents/autoencoders.html>.

- Gril, J. 2008. The State of Indie Gaming. Gamasutra. Viitattu 27.07.2022. <https://www.gamedeveloper.com/business/the-state-of-indie-gaming>.
- Hidalgo, J., Camahort, E., Abad, F. & Vicent, M. 2008. Procedural Graphics Model and Behavior Generation. Viitattu 12.6.2022. https://www.researchgate.net/publication/220857164_Procedural_Graphics_Model_and_Behavior_Generation.
- Indie-pelit ovat suomalaisen peliteollisuuden aluskasvillisuutta. 2012. YLE. Viitattu 27.4.2022. <https://yle.fi/uutiset/3-5345152>.
- Jesus, D. A. R., 2016. Procedural Modeling of Buildings with Shape Grammars. GameDev.net. Viitattu 6.4.2023. <https://www.gamedev.net/tutorials/programming/engines-and-middleware/procedural-modeling-of-buildings-with-shape-grammars-r4596/>.
- Jiang, H., Yan, D., Zhang, X. & Wonka, P. 2018. Selection Expressions for Procedural Modeling. IEEE. Viitattu 17.4.2023. https://www.researchgate.net/publication/328484066_Selection_Expressions_for_Procedural_Modeling.
- Karpathy, A., Abbeel, P., Brockman, G., Chen, P., Cheung, V., Duan, Y., Goodfellow, I., Kingma, D., Ho, J., Houthoof, R., Salimans, T., Schulman, J., Sutskever, I. & Zaremba, W. 2016. Generative models. OpenAI. Viitattu 27.4.2022. <https://openai.com/research/generative-models>.
- Lorens, A. 2019. Everything you wanted to know about procedural modeling. Viitattu 26.4.2022. <https://professional3dservices.com/blog/procedural-modeling.html>.
- Merrell, P. & Manocha, D. 2010. Model Synthesis: A General Procedural Modeling Algorithm. University of North Carolina at Chapel Hill. Viitattu 16.6.2022. <https://paulmerrell.org//tvcg.pdf>.
- Müller, P., Wonka, P., Haegler, S., Ulmer, A. & Van Gool, L. 2006. Procedural Modeling of Buildings. Viitattu 16.4.2023. <http://peterwonka.net/Publications/pdfs/2006.SG.Mueller.ProceduralModelingOfBuildings.final.pdf>.
- Müller, P., Zeng, G., Wonka, P. & Van Gool, L. 2007. Image-based procedural modeling of facades. ACM Transactions on Graphics. Viitattu 17.4.2023. https://www.researchgate.net/publication/220183899_Image-based_procedural_modeling_of_facades.
- Panagiotou, E. & Charou, E. 2020. Procedural 3D Terrain Generation using Generative Adversarial Networks. Viitattu 18.3.2023. <https://arxiv.org/pdf/2010.06411.pdf>.
- Parish, Y. & Müller, P. 2001. Procedural Modeling of Cities. Viitattu 8.6.2022. https://cgl.ethz.ch/Downloads/Publications/Papers/2001/p_Par01.pdf.
- Perlin, K. 2007. Making Noise. Noise Machine. Viitattu 8.6.2022. <https://web.archive.org/web/20071008162042/http://www.noisemachine.com/talk1/index.html>.

Portillo, G. 2022. Fraktaalit. Meteorología en Red. Viitattu 8.6.2022. <https://www.meteorologiaen-red.com/fi/fraktaalit.html>.

Price, A. 2018. The Next Leap: How A.I. will change the 3D industry. Blender. Viitattu 27.4.2022. <https://www.youtube.com/watch?v=FlgLxSLsYWQ>.

Ruemmler, A. N.d. Lindenmayer Systems. Viitattu 27.4.2022. <https://rue-a.github.io/L-Systems/>.

Saaranen-Kauppinen, A. & Puusniekka, A. 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto. Tampere: Yhteiskuntatieteellinen tietoarkisto. Viitattu 25.4.2023. <https://www.fsd.tuni.fi/menetelmaopetus/kvali/>.

Schinko, C., Krispel, U. & Ullrich, T. 2015. Know the Rules - Tutorial on Procedural Modeling. Tutorial Notes. Viitattu 10.6.2022. https://www.researchgate.net/publication/288668420_Know_the_Rules_-_Tutorial_on_Procedural_Modeling.

Togelius, J., Shaker, N. & Dormans, J. 2016. Procedural Content Generation in Games: A Textbook and an Overview of Current Research. Springer. Viitattu 8.6.2022. <http://pcgbook.com/wp-content/uploads/chapter05.pdf>.

Togelius, J., Yannakakis, G., Stanley, K. & Browne, C. 2010. Search-based Procedural Content Generation. Viitattu 11.6.2022. <http://julian.togelius.com/Togelius2010Searchbased.pdf>.

Vivo, P. & Lowe, J. 2015. The Book of Shaders: Noise. Viitattu 8.6.2022. <https://thebookofshaders.com/>.

Voulgaris, G., Mademlis, I. & Pitas, I. 2021. Procedural Terrain Generation Using Generative Adversarial Networks. Viitattu 18.3.2023. <https://aiia.csd.auth.gr/wp-content/uploads/2021/11/ProceduralTerrainGenerationUsingGenerativeAdversarialNetworks.pdf>.

Weisstein, E. 2022. Pythagoras Tree. MathWorld--A Wolfram Web Resource. Viitattu 8.6.2022. <https://mathworld.wolfram.com/PythagorasTree.html>.

What are Fractals? 2009. Fractal Foundation. Viitattu 27.4.2022. <https://fractalfoundation.org/resources/what-are-fractals/>.

Wonka, P., Wimmer, M., Sillion, F. & Ribarsky, M. 2003. Instant Architecture. ACM Transaction on Graphics. Viitattu 16.4.2023. https://www.researchgate.net/publication/47504041_Instant_Architecture.

Wulff-Abramsson, A., Rant, N., Møller, T. & Billeskov, J. 2018. Deep Convolutional Generative Adversarial Network for Procedural 3D Landscape Generation Based on DEM. Viitattu 18.3.2023. https://www.researchgate.net/publication/323579734_Deep_Convolutional_Generative_Adversarial_Network_for_Procedural_3D_Landscape_Generation_Based_on_DEM.

Yumer, M., Asente, P., Mech, R. & Kara, L. 2015. Procedural Modeling Using Autoencoder Networks. Viitattu 15.3.2023. <https://patentimages.storage.googleapis.com/5e/c1/8a/eb92e78d13f6e9/US20170004397A1.pdf>.

Zhu, J., Xie, J. & Fang, Y. 2018. Learning Adversarial 3D Model Generation with 2D Image Enhancer. Viitattu 26.3.2023. https://www.researchgate.net/publication/326464571_Learning_Adversarial_3D_Model_Generation_with_2D_Image_Enhancer.

Liitteet

Liite 1. Haastattelurunko

Pääteema:

Proseduraalisen 3D-mallintamisen työnkulun kehittyminen ja sen vaikutus yleisesti pelikehitykseen sekä tarkemmin indie-pelikehitykseen

Kysymysten teemat:**1. Haastateltavan suhde aiheeseen**

- Kerro kokemuksestasi pelialalla ja 3D-grafiikkaan liittyen
- Kuvaile suhdettasi indie-pelikehitykseen

2. Indie-pelikehitys

- Mitä rajoitteita koet indie-pelikehityksessä olevan? Mitä vapauksia?
- Miten indie-pelit ja niiden kehittäminen on kehittynyt sinun aikanasi?

3. Proseduraalinen 3D-mallintaminen

- Kuvaile suhdettasi proseduraaliseen 3D-mallintamiseen
- Miten luulisit proseduraalinen 3D-mallintaminen vaikuttavan 3D-graafikkojen työtehtäviin?
- Oletko kokenut proseduraalisen mallintamisen kehittyneen aikanasi? Millä tavoilla?
- Miten luulisit tekoälyn kehittymisen vaikuttavan proseduraaliseen 3D-mallintamiseen?

4. Proseduraalisen 3D-mallintamisen suhde pelikehitykseen

- Oletko huomannut proseduraalisen mallintamisen vaikuttaneen pelialaa? Entä indie-pelikehitystä?

- Uskotko proseduraalisen 3D-mallintamisen kehittymisen antavan etua indie-kehittäjille pelimarkkinoilla?
- Näetkö proseduraalisessa 3D-mallintamisessa haittapuolia pelikehitykseen joko AAA- tai indie-puolella?

Liite 2. Kyselylomake

The advancement of procedural 3D modeling workflow and its impact on indie game development

The goal of this survey is to collect data for a **thesis** about the topic mentioned. Please read the questions **carefully** and then try to answer in a few sentences. If you cannot come up with an answer you can move on to the next question. The **anonymity** of the people who responded to the survey is protected, their identities are not revealed and cannot be based on the thesis.

Thank you for answering this survey!

1. Tell us about your experience in the gaming industry. How long have you worked in the field in question and in which areas do you specialize? (Do not answer if you have no experience)

2. Tell us about your experience with 3D graphics. How long have you worked in the field in question and in which areas do you specialize? (Do not answer if you have no experience)

3. What limitations do you feel there are in indie game development?

4. What liberties do you feel there are in indie game development?

5. How has the development of indie games developed during your time?

6. Have you used procedural 3D modeling? How and for what purposes?

7. How do you think procedural 3D modeling will affect the work tasks of 3D graphic designers?

8. Has procedural modeling developed during your time? In what ways?

9. How do you think the development of artificial intelligence will affect procedural 3D modeling?

10. Have you noticed that procedural modeling has influenced the game industry in general or indie game development?

11. Do you think the development of procedural 3D modeling will give an advantage to indie developers in the game market?

12. Do you see any disadvantages of procedural 3D modeling for game development, either on the triple A or indie side?

Google ei ole luonut tai hyväksynyt tätä sisältöä.

Google Forms