PELIKEHITYS ILMAN PIIRRETTYÄ GRAFIIKKAA

Lappeenrannan–Lahden teknillinen yliopisto LUT

Tietotekniikan kandidaatintyö

2023

Ida Kirveskoski

Tarkastaja: TkT Erno Vanhala

TIIVISTELMÄ

Lappeenrannan–Lahden teknillinen yliopisto LUT

LUT Teknis-luonnontieteellinen

Tietotekniikka

Ida Kirveskoski

Pelikehitys ilman piirrettyä grafiikkaa

Tietotekniikan kandidaatintyö

37 sivua, 12 kuvaa ja 2 taulukkoa

Tarkastaja: TkT Erno Vanhala

Avainsanat: grafiikka, visuaaliset tehosteet, ympäristötehosteet, partikkeliefekti, varjostaja, sensoritehosteet, valaistustehosteet

Nykypäivänä peleissä käytettävä grafiikka on korkealaatuista ja näyttävää. Pelaajat suosivat kaunista visuaalista ilmettä, ja tätä visuaalista ilmettä tuetaan visuaalisilla tehosteilla. Herää kuitenkin kysymys kuinka suuri rooli visuaalisilla tehosteilla on peligrafiikassa ja voidaanko piirretty grafiikka korvata kokonaan visuaalisilla tehosteilla.

Tämän kandidaatintutkielman tavoitteena on tutkia kuinka helppoa piirretyn grafiikan korvaaminen visuaalisilla tehosteilla on ja mitä etuja ja haittoja siihen liittyy. Työssä tutustutaan visuaalisiin tehosteisiin ja niiden mahdollisuuksiin. Piirretyn grafiikan korvaamista visuaalisilla tehosteilla tutkitaan käyttämällä suunnittelutieteellistä tutkimusmenetelmää ja työn tuloksia analysoidaan SWOT-taulukon avulla.

Työn artefaktina toimii itse suunniteltava ja toteutettava tasohyppelypeli Godot Engine versio 4.1. -pelimoottorilla. Pelin rajausehdot ovat seuraavat: Piirretty grafiikka muodostuu 16x16 pikselin kokoisesta neliöistä, tunnelma luodaan visuaalisilla tehosteilla ja peli on kaksiuloitteinen.

Tutkimuksen perusteella voidaan todeta, että visuaalisilla tehosteilla on potentiaalia korvata piirretty grafiikka ainakin osittain. Kuitenkin vaikuttavan kokonaisuuden luomiseen pelkillä visuaalisilla tehosteilla, pelin kehittäjältä vaaditaan hyvää ymmärrystä kunkin tehosteen toiminnasta ja mahdollisuuksista. Tutkimuksessa toteutetussa pelityypissä ei onnistuttu korvaamaan piirrettyä grafiikkaa täysin ja pelissä käytettiin vain pientä osaa mahdollisista visuaalisista tehosteista. Tulevaisuudessa lisätutkimusta voisi tehdä esimerkiksi toisen tyyppisen pelin näkökulmasta.

ABSTRACT

Lappeenranta–Lahti University of Technology LUT

School of Engineering Science

Software Engineering

Ida Kirveskoski

Game development without drawn graphics.

Bachelor's thesis

2023

37 pages, 12 figures and 2 tables

Examiner: D.Sc. (Tech.) Erno Vanhala

Keywords: graphics, visual effects, environmental effects, particle effect, shader, sensor effect, illumination effects

Graphics in games nowadays are high quality and spectacular. Gamers prefer beautiful visual appearance, and this visual appearance is supported with different visual effects. However, the question arises as to how big a role visual effects play in game graphics and whether drawn graphics can be completely replaced by visual effects.

The goal of this bachelor's thesis is to study how easily drawn graphics can be replaced with visual effects and what are the advantages and disadvantages associated with it. In Thesis introduces visual effects and their possibilities. The replacement of drawn graphics with visual effects is studied using the design science research method and the result of the work are analyzed using a SWOT analysis.

The artefact of this thesis is a game which will be designed and executed by myself and will be created with Godot Engine version 4.1. – game engine. The boundary conditions of the game are as follows: The drawn graphics consist of 16x16 pixel squares, the feeling is created with visual effects and the game is two-dimensional.

Based on this research, it can be concluded that visual effects have the potential to replace drawn graphics at least partially. However, to create an impressive whole with only using visual effects, a good understanding of the function and possibilities of each effect is required from the game developer. In the type of game created in study, it was not possible to completely replace the drawn graphics and only small amount of the possible visual effects were used in the game. In the future, further research could be done, for example, from the perspective of another type of game.

KIITOKSET

Kiitokset ohjaajalle hyvästä ohjaamisesta ja hyvistä vinkeistä prosessin aikana. Kiitos myös opiskelijatovereilleni, jotka ovat auttaneet tämän työn syntymisessä. Erityiskiitos myös kissalleni Hestialle hyvästä työnjohdosta, laadunvalvonnasta ja työskentelymoraalin ylläpitämisestä.

Sisällysluettelo

Tiivistelmä

Abstract

Kiitokset

1 Johdanto 7
1.1. Tausta 7
1.2. Tavoitteet ja rajoitukset 8
1.3. Sisältö 8
2 Tutkimusmenetelmä 10
2.1 Suunnittelutieteellinen tutkimusmenetelmä 10
2.1.1 Suunnittelutieteen tutkimusmenetelmän vaiheet 11
2.2. SWOT-analyysi 12
3 Peligrafiikka ja tehosteet 13
3.1 Aikaisempi tutkimus 13
3.2 Pelien ja grafiikan kehitys 14
3.3 Pelimoottori 15
3.4 Peligrafiikan suunnittelu 16
3.5 Visuaaliset tehosteet 17
3.5.1 Globaalit valaistustehosteet 17
3.5.2 Ympäristötehosteet 18
3.5.3 Sensoritehosteet 19
3.6 Mahdollisuudet 20
4 Pelin synty ja tulokset 21
4.1. Piirretty grafiikka 21
4.2. Tehosteet ja rakenne 21
5 Tulosten arviointi 29

5.1. SWOT-Analyysi 29
5.1.1 Vahvuudet ja mahdollisuudet 30
5.1.2 Heikkoudet ja uhat 31
6 Yhteenveto ja johtopäätökset 33
Lähteet 35

1 Johdanto

Tässä luvussa käydään tiiviisti läpi tutkimuksen taustaa, sen tavoitteita ja rajoituksia ja avataan hieman koko työn sisältöä.

1.1. Tausta

Luovuus ja mielikuvitus ovat osa ihmisen perusolemusta. Erilaiset tarinat ovatkin kiinnostaneet ja ruokkineet ihmisten mieliä alusta asti. Alun perin ihmiset kehittivät tarinoita ja seuraleikkejä, sitten alettiin luoda erilaisia fyysisiä pelejä (Harari, 2014, 2021) ja lopulta ihmiskunta keksi videopelit. Videopelien ero aikaisempiin peleihin on se, että harvassa pelissä pelaajan tarvitsee vaivautua itse kuvittelemaan tarinan miljöötä vaan tarinan miljöö luodaan erilaisten grafiikoiden ja tehosteiden avulla.

Ensimmäiset videopelit syntyivät jo 1950 luvulla ja pelihallipelien kulta-aika alkoi 1970-luvun lopulla. Nykyään videopelit ovat rantautuneet kotitalouksiin erilaisten siirrettävien pelikonsolien muodossa. (Cohen *et al.*, 2012.) Peligrafiikat näiden vuosien aikana ovat kehittyneet yksinkertaisesta muutaman pikselin peligrafiikasta lähes valokuvamaiseksi grafiikaksi. Nykyajan pelinkehittäjällä on paljon mahdollisuuksia pelinsä piirretyn grafiikan suunnittelussa. (Hantula, 2018.) Tärkeää on löytää grafiikkatyyli, joka sopii peliin ja sen aiottuun tunnelmaan.

Tunnelmaa peleihin luodaan myös erilaisilla visuaalisilla tehosteilla. Partikkeliefektillä, joka sai inspiraationsa Star Wars elokuvasta (Hasu, 2018), voidaan stimuloida useita luonnonilmiöitä. Varjostuksilla peleihin luodaan syvyyttä, mutta varjojen määrällä voidaan vaikuttaa myös siihen, miten synkkä tunnelma pelissä on. (Yang, Yip and Xu, 2009.) Myös valittu väripaletti vaikuttaa siihen, minkälainen tunnelma pelistä välittyy pelaajalle (Feldman, 2001).

Hyvin usein tehosteilla tuetaan piirretyn grafiikan vaikutusta. On kuitenkin pelejä, joissa piirretyn grafiikan osuus on jätetty vähemmälle ja sen sijaan käytetään enemmän tehosteita mutta on kuitenkin mahdollista luoda näyttävä peli myös ilman piirrettyä grafiikka. Herääkin

kysymys, miten helposti piirretty grafiikka voidaan korvata erilaisilla tehosteilla? Suuret grafiikka määrät vaativat paljon resursseja peliä pyörittävältä koneelta, minkä vuoksi pelikoneidenkin on kehtyttävä jatkuvasti (Dally, Keckler and Kirk, 2021). Visuaaliset tehosteet voisivat mahdollistaa vaikuttavan pelin luomisen huomattavasti pienemmällä resurssitarpeella, jolloin voidaan keskittyä itse tarinan rakentamiseen.

1.2. Tavoitteet ja rajoitukset

Tämän tutkimuksen tarkoituksena on selvittää, kuinka helposti piirretty grafiikka voidaan korvata erilaisilla tehosteilla. Tätä tutkimusta varten luodaan pieni peli, jossa ei käytetä juurikaan piirrettyä grafiikkaa ja tutkitaan asiaa käytännössä. Arviodaan samalla, mitä etuja ja haittoja tällaisessa pelissä on. Pelille on annettu seuraavat rajoitteet:

- Ainoa piirretty grafiikka pelissä on 16x16 pikselin kokoiset neliöt
- Tunnelma luodaan erilaisilla visuaalisilla tehosteilla, kuten partikkeliefektit, näytönliikkeet ja varjostukset
- Peli on kaksiulotteinen

Tutkimuksessa pyritään myös vastaamaa tutkimuskysymyksiin, mitä Godot Engine - pelimoottori tarjoaa näyttävän pelin luomiseen ilman piirrettyä grafiikkaa, ja miten visuaalisilla tehosteilla voidaan korvata grafiikan puutetta. Vastaukset näihin kysymyksiin voivat olla tärkeitä pelinkehittäjälle, joka haluaa säästää grafiikasta tai haluaa tietää, miten tehostaa pelin grafiikan vaikutusta entisestään tehosteiden avulla.

1.3. Sisältö

Alussa avataan työssä käytettyä tutkimusmenetelmää ja analysoinitmenetelmää. Sitten käydään läpi hieman aiempaa tutkimusta, minkä jälkeen siirrytään pelinkehityksen ja grafiikan kehityksen historiaan hyvin lyhyesti. Sitten esitellään lyhyesti, mikä on pelimoottori. Tämän jälkeen tutustutaan peligrafiikan suunnitteluun ja erilaisiin visuaalisiin tehosteisiin, joilla peleihin luodaan tunnelmaa. Kirjallisen katsauksen jälkeen siirrytään kertomaan tuloksista. Kerrotaan minkälainen peli luotiin, miten se syntyi ja millaisia

haasteita ja onnistumisia kohdattiin peliä tehdessä. Sitten heijastetaan tuloksia tutkimuskysymyksiin ja kerrotaan hieman rajoituksista. Lopuksi tiivistetään kokotutkimus lyhyeen yhteen vetoon ja tehdään johtopäätöksiä ja pohditaan, miten tämä tutkimus vaikuttaa tulevaisuudessa.

2 Tutkimusmenetelmä

Tässä luvussa avataan tutkimuksessa käytettyä tutkimusmenetelmää, ja miten tutkimusmenetelmän käyttö näkyy tässä tutkimuksessa. Luvussa esitellään myös tulosten arvioimisessa käytettävä analysointimenetelmä.

2.1 Suunnittelutieteellinen tutkimusmenetelmä

Tässä tutkimuksessa tarkoituksena on tutkia, miten helppoa on korvata piirretty grafiikka erilaisilla visuaalisilla tehosteilla. Tutkimuksessa hyödynnän suunniteltua ratkaisua, artefaktia, ongelman ratkaisemiseksi. Tutkimusmenetelmäksi tähän tutkimukseen sopiikin siis suunnittelutieteellinen tutkimusmenetelmä (design science research method).

Suunnittelutieteellinen tutkimus on menetelmä, joka mahdollistaa tutkimuksen, kun tavoite on artefakti. Tämän tutkimusmenetelmän tarkoituksena on tutkia jotakin artefaktia ja sen käyttäytymistä akateemisesta ja organisaation näkökulmasta. (Dresch *et al.*, 2015.) Tietojärjestelmien kohdalla suunnitteluprosessin vaiheet ovat rakentaminen ja arviointi, sekä artefaktit ovat usein rakenteita, malleja, menetelmiä tai ilmentymiä. Pohjimmiltaan tämä tutkimusmenetelmä on ongelmanratkaisuparadigma, joka pyrkii luomaan innovaatioita, jotka määrittelevät ideat ja käytännöt, joiden avulla ratkaistava ongelma voidaan suorittaa tehokkaasti. (Hevner *et al.*, 2004.)

Tietojärjestelmät ovat erilaisia järjestelmiä peliohjelmistoihin verrattuna, mutta ne ovat molemmat kokonaisuuksia, joiden taustalla on ohjelma. Siispä mielestäni suunnittelutieteen tutkimusmenetelmän metodit tietojärjestelmä tutkimuksessa toimivat myös tässä tutkimuksessa. Tässä tapauksessa artefaktina meillä on suunniteltava peli, jonka avulla voidaan selvittää piirretyn grafiikan korvattavuutta tehosteilla. Tähän työhön sopii myös hyvin tietojärjestelmien suunnitteluprosessin vaiheet: rakennetaan peli, ja arvioidaan siitä tuloksia.

Suunnittelutieteellinen tutkimusmenetelmä sisältää tiukan prosessin artefaktien suunnittelemiseksi jonkin ongelman ratkaisemiseksi, suunnitelmien arvioimiseksi ja

tulosten välitettäväksi eteenpäin. Tällainen artefakti voi olla mikä vain suunniteltu kohde, kunhan siihen sisältyy ratkaisu ymmärrettyyn tutkimusongelmaan. (Peffers *et al.*, 2007.)

2.1.1 Suunnittelutieteen tutkimusmenetelmän vaiheet

Suunnittelutieteen tutkimusmenetelmän perusperiaateenmukaan tietoa ja ymmärrystä suunnitteluongelmasta ja sen ratkaisusta hankitaan samanaikaisesti artefaktin rakentamisen ja soveltamisen yhteydessä (Hevner *et al.*, 2004). Tässä tutkimuksessa noudatellaan Peffersin ja muiden (Peffers *et al.*, 2007) tutkimuksessaan määrittelemää prosessia.

Taulukko 1: Peffersin ja muiden (2007) suunnittelutieteen tutkimuksen perusvaiheet.

Suunnittelutieteen tutkimuksen periaatteet tietojärjestelmä tutkimuksessa	
Perusvaiheet	Sisällön kuvaus
1. Ongelman tunnistaminen ja motivointi	Määritellään tutkimuksen ongelma ja perustellaan sen tärkeys
2. Ratkaisun tavoitteiden määrittely	Kuvataan ratkaisun tavoitteet ja mahdollisuudet
3. Suunnittelu ja kehittäminen	Luodaan ongelmalle artifakti
4. Demonstrointi	Demonstroidaan, miten artifaktilla voidaan ongelma ratkaista
5. Arviointi	Arvioidaan kuinka hyvin artifakti ratkaisee ongelman
6. Viestintä	Viestitään ratkaisu, eli julkaistaan tulokset

Tässä tutkimuksessa vaiheet 1-2 käydään läpi johdannossa ja tarkemmin seuraavassa luvussa. Tutkimuksen ongelma määritellään jo johdannossa ja sen tärkeyttä perustellaan johdannon lisäksi vielä seuraavassa luvussa. Ratkaisun tavoitteet kuvataan myös jo johdannossa ja visuaalisia mahdollisuuksia kuvataan tarkemmin ja laajemmin luvussa 3. Luvussa 3 esitellään muun muassa erilaisia visuaalisia tehosteita. Tämän työn artefakti, eli peli on esitelty johdannossa ja sen luónnista kerrotaan luvussa 4. Tämä peli luodaan

käyttämällä vain 256 pikselin grafiikkaa eli toisin sanoen 16x16 pikselin kokoisia neliöitä. Kaikki muut graafiset ominaisuudet toteutetaan visuaalisilla tehosteilla.

Vaiheet 4 ja 5 käsitellään luvuissa 4 ja 5 ja luvussa 4 avataan paremmin, miten peli, tehtiin ja mitä haasteita havaittiin. Arvioidaan myös, mikä oli helppoa, ja miltä tulos lopulta näyttää. Luvussa 5 peilataan tuloksia tutkimuskysymyksiä vastaan ja kerrotaan tutkimuksen rajoitteista ja arvioidaan samalla, kuinka hyvin rakennettu peli ratkaisee ongelman. Vaihe 6 käsittää koko tämä kandidaatin työn, missä kommunikoin ulos tämän tutkimuksen tulokset ja vaiheet.

2.2. SWOT-analyysi

SWOT analyysi on vanha työkalu ja sitä käytetään edelleen runsaasti eri järjestöjen keskuudessa (Sarsby, 2016). SWOT-analyysi on kehitetty liiketoimintastrategian työkaluksi, jonka avulla voidaan hahmottaa organisaation vahvuuksia (Strenght), heikkouksia (Weaknesses), mahdollisuuksia (Opportunities) ja uhkia (Threaths) suhteessa kilpailijoihin. Tämä työkalu on kehitetty arvioimaan organisaation tilannetta suhteessa kilpailijoihin ja auttamaan yrityksen strategiasuunnitelman kehityksessä. Vahvuudet ja mahdollisuudet auttavat näkemään ne puolet, jotka antavat etulyöntiaseman kilpailijoihin verrattuna. Uhat ja heikkoudet vastaavasti kertovat ominaisuudet, jotka johtavat epäedulliseen kilpailuasemaan. (Phadermrod, Crowder and Wills, 2019; Teoli, Sanvictores and An, 2019.) SWOT-analyysiä on helppo lukea, sillä se on yksinkertainen diagrammi, eikä sisällä matemaattisia ilmauksia. Lisäksi SWOT-analyysi on myös helposti sovellettavissa eri tasoille organisaation sisällä aina yksilöstä yrityksenstrategiaan. (Sarsby, 2016.)

Vaikka SWOT-analyysi on suunniteltu erityisesti organisaatioiden vertailuun, soveltuu SWOT-analyyi mielestäni hyvin myös tämän työn tulosten kartoittamiseen ja arvioimiseen. Tässä tapauksessa visuaaliset tehosteet ja piirretty grafiikka ovat kilpailijoita keskenään ja pyritään hahmottamaan visuaalisten tehosteiden heikkouksia ja vahvuusia piirrettyyn grafiikkaan nähden.

3 Peligrafiikka ja tehosteet

Tässä luvussa käsitellään lyhyesti aikaisempia tutkimuksia ja siirrytään pohjustamaan aihetta kertomalla pelinkehityksen sekä peligrafiikan historiaa. Sitten käsitellään lyhyesti myös pelimoottoreita sekä grafiikan suunnittelua. Lisäksi käsitellään tarkemmin peleissä käytettäviä visuaalisia tehostetia.

3.1 Aikaisempi tutkimus

Aivan vastaavia tutkimuksia pelien graafittomuudesta tai piirretyn grafiikan korvaamisesta ei ole aiemmin tehty, mutta eräässä tutkimusartikkelissa kuvataan tilanne, jossa luotiin peli siten, että peli oli pelattavissa niin sokeille kuin näkevillekin henkilöille. Östblad ja muut (2014) esittelivät projektin, jossa luotiin point-and-click seikkailupeli älypuhelimelle ja tabletille, mikä on yhtä toiminnallinen ja mielekäs niin näkeville kuin sokeillekin pelaajille. Tässä pelissä grafiikan rinnalle on lisätty äänimerkkejä ja muita auditiivisia keinoja, jotta sokeat pelaajat saisivat tarvittavat tiedot pelin ympäristöstä ja esineistä. (Östblad *et al.*, 2014.) Myös Enngström ja muut ovat suorittaneet tutkimukseen (2015), jossa testattiin saman pelin pelaamisen mielekkyyttä näkevien ja sokeiden välillä. Projektissa ryhmä oli luonut pelin, joka olisi pelattavissa niin sokeiden kuin näkevienkin näkökulmasta. Tässä artikkelissa he kuvasivat tutkimuksena tuloksia, jossa he arvioivat kolmen itsenäisen ryhmän pelikokemusta. Yksi ryhmä koostui näkevistä pelaajista, toinen sokeista ja kolmas näkevistä, joiden silmät oli peitetty. Tutkimuksessa ei havaittu suuria eroja kyseisen pelin pelattavuudessa ryhmien välillä. (Engström, Brusk and Östblad, 2015.) Jonkin verran tutkimuksia löytyy siitä, miten grafiikkaa voidaan korvata auditiivisilla tehosteilla varsinkin näkövammaisten näkökulmasta. Yleensä näissä peleissä on kuitenkin ollut myös grafiikkaa näkeviä pelaajia ajatellen.

Grafiikan kehitystä vuosien varrella on tutkittu jonkin verran. Esimerkiksi Tienari on tutkinut kandidaatin työssään grafiikan kehitystä ja sen näkyvyyttä digitaalisissa peleissä (Tienari, 2017) ja Hantula tutki omassa kandidaatin työssään peligrafiikan kehittymistä (Hantula, 2018). Grafiikan suunnittelusta löytyy myös paljon erilaista aineistoa ja muun

muassa Feldmanien kirjassa (2001) esitellään yksityiskohtaisesti, miten suunnitellaan peligrafiikkaa arcade peleihin. Visuaalisia tehosteita käsitteleviä teoksia on myös saatavilla runsaasti. Esimerkiksi Yang ja muut (2009) ovat koonneet kirjaansa *Visual Effects in Computer Games* kattavasti visuaalisia tehosteita ja esittelevät niiden toiminnallisuutta. Tutkimuksia itse peligrafiikkan kehityksestä vuosien varrella on tehty ja on tutkittu myös minkälaiset asia ovat vaikuttaneet peligrafiikan kehitykseen. On myös tutkittu ja luotu erilaisia oppaita siitä, miten grafiikasta saa tehtyä vaikuttavan ja millaisilla tehosteilla grafiikan vaikutusta voidaan korostaa entisestään.

Grafiikan tyylejäkin on runsaasti aina yksinkertaisesta pikseligrafiikasta lähes fotorealsitiseen grafiikkaan ja näiden välillä on myös tehty vertailtua. Muun muassa Masuch ja Röber esittelevät artikkelissaan erialisia grafiikan tyylejä ja analysoivat tietokonepelien graafisia puolia (Masuch and Röber, 2004). Kuitenkaan tutkimuksia nimenomaan piirretyn grafiikan korvaamisesta visuaalisilla tehosteilla ei ole tehty. Olisi kuitenkin erittäin kiinnostavaa tietää, miten tehokkaasti visuaaliset tehosteet voivat korvata piirretyn grafiikan ja millaisia ongelmia siihen liittyy.

3.2 Pelien ja grafiikan kehitys

Peli on aktiviteetti, jossa on vähintään yksi pelaaja, säännöt sekä voitto- ja häviötila. Videopeli on peli, jota pelataan digitaaliselta näytöltä (Rogers, 2014). Ensimmäiset videopelit luotiin, kun tietojenkäsittely vielä perustui muistiin. Yliopiston tietokonelaboratoriossa, yksi tai kaksi ohjelmoijaa kehitti hauskan mutta helpon tavan viestiä tietokoneen ominaisuuksista ei teknisille henkilöille. Tietokoneiden yleistyessä kotitalouksiin teknikot alkoivat luoda omia pelejään, koodaamalla ja suunnittelemalla grafiikkaa, tasoja ja kerrontaa. Pelien kasvaessa monimutkaisemmiksi alettiin pelinkehitys jakaa osiin, jolloin ohjelmointi ja grafiikan suunnittelu eroteltiin toisistaan. (Cohen *et al.*, 2012.) Nykypäivän videopelit siis koostuvat itse koodista, grafiikasta ja äänistä ja varsinkin grafiikan arvo ja rooli peleissä on kasvanut ja sillä luodaan niin tunnelmaa kuin tarinaakin. Lisäksi modernit laitteet ovat mahdollistaneet yksityiskohtaisemman ja realistisemman grafiikan käytön.

Grafiikka on visuaalinen esitys, kuten kuva tai diagrammi, jota käytetään tiedottamiseen, havainnollistamiseen ja viihdyttämiseen. Tietokonegrafiikka taas on grafiikkaa, joka tuotetaan tietokoneella ja sen pienin yksikkö on pikseli. (Salomon, 2011.) Ensimmäisissä videopeleissä ei joko ollut grafiikkaa ollenkaan tai grafiikat olivat hyvin yksinkertaiset ja niitä pelattiin pienellä mustavalkoisella oskilloskooppinäytöllä. Tällaisia videopelejä ovat esimerkiksi OXO (1952) ja Spacewar! (1962). Ensimmäinen Arcade videopeli, Computer Space, tehtiin 1971 ja 70-luvun lopulla pelihallit alkoivat yleistyä. Ensimmäisissä arcade peleissä käytettiin joko vektorigrafiikkaa, kuten Battlezonessa 1980 tai rasterigrafiikkaa kuten Pac-Manissä 1980. (Rogers, 2014.) Nykypäivän laitteisto on kuitenkin kehittynyt niin paljon, että laitteistojen ominaisuudet mahdollistavat kehittäjän luoda kauniita pelejä tekniikoilla, jotka mahdollistavat jopa fotorealistisen pelimaailman vaikuttavine tehosteineen (Hasu, 2018).

3.3 Pelimoottori

Pelimoottori on ohjelmointikirjasto, joka mahdollistaa peliohjelmoijalle pääsyn hyödyllisiin toiminnallisuuksiin, joilla peli voidaan luoda. Näiden moottorien on oltava suorituskykyisiä ja vakaita sillä ohjelman on kyettävä kaatumatta suorittamaan kaikki ominaisuudet, joita pelin pyörittäminen vaatii. Tyypillisesti pelimoottorin tarjoaman rajapinnan kautta, voidaan muun muassa ohjelma saada reagoimaan hiiren tai näppäimistön toimintaan. Liäksi ohjelmaan voidaan luoda grafiikkaa sekä lisätä ääniä. (Kallio-Kujala, 2016.) Pelimoottori on siis niin ikään ohjelma, joka tekee pelien luonnista tänä päivänä helpompaa, sillä se tarjoaa erilaisia komponentteja ja valmiita matemaattisia malleja, joilla voi koota laajan kirjon erilaisia pelejä.

Pelimoottoreita on nykyään tarjolla paljon. Osa niistä on avoimen lähdekoodin ohjelmia, osa vaatii lisenssin, jotta niitä voi käyttää. Jokainen näistä pelimoottoreista on erilainen ja vastaa hiukan eri tarpeisiin: niissä on käytetty eri ohjelmointikieliä, ne on suunniteltu erilaisille alustoille, niiden design on erilainen ja niihin vaadittavat lisenssit ovat erilaiset. (Mustonen, 2023.) Godot on hyvä esimerkki avoimen lähdekoodin pelimoottorista. Avoimen lähdekoodin pelimoottorit tarjoavat usein hyvän paketin ominaisuuksia pelien luontiin. Suuret kaupalliset moottorit kuten Unity ja Unreal Engine tarjoavat uusimmat ja parhaat

tekniset ominaisuudet useimpiin käyttötarkoituksiin. (Hasu, 2018.) Tässä tutkimuksessa keskitymme Godot Engine -pelimoottoriin ja sen 4.1 versioon.

Godot Engine -pelimoottori on täysin ilmainen avoimen lähdekoodin ohjelma, joka tarjoaa renderöinti ominaisuuden sekä 2D- että 3D-grafiikalle, pelifysiikaa, tuen useille alustoille sekä kehitysympäristön. Godotilla tehdyistä peleistä ei peritä maksuja tai piilokuluja kehittäjältä. Lisäksi kehittäjän on myös mahdollista ja sallittua muokata moottorin koodia omiin tarpeisiinsa sopivaksi. (Bradfield, 2018.) Godot Engine tarjoaa kehittäjälle ilmaiseksi kattavan paketin ominaisuuksia ja mahdollisuuksia, joilla kehittäjä voi luoda juuri sellaisen peli kuin hän haluaa.

3.4 Peligrafiikan suunnittelu

Peleissä käytettävän grafiikan tyylejä on tänä päivänä lukuisia, ja pelinkehittäjät voivat itse valita minkälaista tyyliä lähtevät peleissä toteuttamaan. Peleissä käytettävään grafiikkaan vaikuttaa paljon se, minkälaista peliä ollaan luomassa ja millaista tunnelmaa pelaajalle halutaan välittää. ”Hyvä peligrafiikka ei pelasta huonoa peliä - oleellisinta on kuitenkin pelikokemus, mutta hyvin toteutettu peligrafiikka on merkittävässä osassa ja tuo suurta lisäarvoa” (Mansikka, 2015).

On myös mahdollista, ettei ei käytä grafiikkaa lainkaan. Mikäli oletetaan, että grafiikalla tarkoitetaan kaikenlaista visuaalista kuvaa, jolla peliin luodaan visuaalista ulkonäköä, grafiikattomuus voidaan tulkita parilla eri tavalla. Grafiikattomuus voi viitata siihen, ettei pelissä ole lainkaan kuvaa, jolloin peli perustuu ääniin ja tuntoon. Toisaalta grafiikattomuudella voidaan viitata peliin, joka perustuu pelkkään tekstiin. Sillä voidaan viitata myös piirretyn grafiikan puutteeseen tai sen vähäisyyteen, jolloin peli on voitu toteuttaa lähinnä kuvioita, merkkejä ja tehosteita käyttäen.

Kun suunnitellaan piirrettyä grafiikkaa tai grafiikkaa ylipäätänsä, tulee huomioida pelin väripaletti, jolla peliin luodaan tunnelmaa ja välitetään tunteita pelaajalle. Väripalettia valitessa tulee myös ottaa huomioon kohderyhmän ikä ja mahdollisesti myös sukupuoli. Lisäksi grafiikan suunnittelussa tulee ottaa huomioon varjostukset ja valaistus, sillä näillä luodaan volyymia. Varsinkin 2D-peleissä valoilla ja varjostuksilla tuodaan eloa ja kolmiulotteisuutta piirretylle grafiikalle. (Feldman, 2001.)

Grafiikan kehityksessä on huomioitava myös tehosteet, joilla piirretyn grafiikan vaikutusta voidaan tehostaa ja realisoida. Yhä useammin pelaajat suosivat visuaalista kokemusta ja erialaiset erikoistehosteet kuten varjostukset tai liikkeen animointi, voivat parantaa pelikokemusta ja herättää huomiota pelaajien keskuudessa. (Zhang and Hu, 2017.)

3.5 Visuaaliset tehosteet

Videopeleissä käytettyjen tehosteiden inspiraatio on useimmiten saatu elokuvien edelläkävijöistä (Yang, Yip and Xu, 2009). Visuaalisilla tehosteilla pyritään simuloimaan tosielämän visuallisia kokemuksia. Yang ja muut (2009) jakavat kirjassaan visuaaliset tehosteet kolmeen ryhmään: Globaalit valaistustehosteet, ympäristötehosteet ja anturitehosteet. Valaistustehosteet viittaavat erialisiin valotehosteisiin ja varjostuksiin. Ympäristötehosteilla viitataan tehosteisiin, joilla imitoidaan esimerkiksi savua tai pilviä tunnelman luomiseksi. Anturitehosteilla viitataan kameran suunnitteluominaisuuksiin kuten liikkeeseen, kameran epätarkkuuteen liikkeessä ja yönäköön. (Yang, Yip and Xu, 2009.)

3.5.1 Globaalit valaistustehosteet

Globaalit valaistustehosteet luovat realistista tunnelmaa peleihin, stimuloimalla muun muassa heijastuksia ja pehmeitä varjoja. Vanhemmissa peleissä globaaleja valaistustehosteita ei juuri ole, mikä saa ne näyttämään epärealistisemmilta. Kun koneiden grafiikankäsittely-yksiköt (GPU, Graphics processing unit) kehittyivät, pelinkehittäjät alkoivat sisällyttää enemmän ja monimutkaisempia valaistustehosteita peleihin. Valaistustehosteita luodaan peleihin muun muassa valokarttatekniikoiden avulla. Valokarttoihin (Light map) tallennetaan ennalta laskettujen virtuaalipintojen kirkkaudet, ja näitä arvoja käytetään varjostamaan peliobjekteja reaaliajassa. (Yang, Yip and Xu, 2009.)

Valokartoilla voidaan luoda realistisen tuntuisia valoilmiöitä, mutta mitä tehokkaampi valokartta, sitä enemmän muistia peli tarvitsee toimiakseen. Valokartan valintaan vaikuttaa myös pitkälti, kuinka tarkkaan valaistustehosteiden tulee imitoida todellista elämää: riittääkö että valo pysyy staattisena vai pitääkö valaistuksen muuttua dynaamisesti kuin päivänkierto?

Ympäristön okkluusio (ambient occlusion) on taas globaali valaistusmenetelmä, jolla stimuloidaan ympäristön valaistuksen varjostuksia. Ne toimivat hyvin samalla tavoin kuin valokartat, mutta valopisteiden sijaan kerätään tummuusasteita. Nykyisten ympäristö okkluusio menetelmien etuna on se, että se on riippumaton kohtauksen monimutkaisuudesta ja grafiikkayksikön täydellisestä suorituksesta. (Yang, Yip ja Xu, 2009.) Varjostuksia voidaan luoda myös muilla keinoin.

Varjostajat (Shaders) ovat saaneet inspiraationsa Star Wars Episode IV elokuvasta 1977. Tämän jälkeen varjostustekniikoita alettiin kehittää videopeleihin sopivaksi. Varjostajat määrittävät miltä esine näyttää ja miten se reagoi valoon. Valolaskelmat yleensä koodataan varjostajaan ja näitä tietoja voidaan hyödyntää monin tavoin. Nykypäivän pelimoottorit myös tukevat moderneja varjostajia. Varjostaja toimii funktiona, joka vastaanottaa pikselin sijainnin ja palauttaa sille värin. Varjostaja käyttää prosessorin sijasta grafiikkasuoritinta toimiakseen. (Hasu, 2018.)

3.5.2 Ympäristötehosteet

Ympäristötehosteilla stimuloidaan luonnonilmiöitä, kuten savua, tulta ja sadetta. Näillä pyritään luomaan realistista tunnelmaa peliin. Partikkeliefektit (particle effects) ovat laajalti käytössä elokuvateollisuudessa, mutta se on omaksuttu myös osaksi videopelimaailmaa. Partikkeliefekti muodostuu hiukkasista, jossa kullakin hiukkasella on tietty aikajärjestys ja elinkaari. Partikkeliefekti mallintaa sumeita objekteja erillisillä hiukkasilla, mallintaa näiden objektien liikkeitä sekä renderöi tuloksen näytölle. Partikkelit voidaan renderöidä erilaisilla tekniikoilla taiteellisten ja järjestelmävaatimusten mukaan. Mitä tehokkaampi grafiikkaprosessori sen paremmat tekniikat partikkeliefektin luontiin on tarjolla. (Yang, Yip and Xu, 2009.)

Teknologian kehittyessä myöskin partikkeliefektien todentuntuisuutta on voitu lisätä. Nykyisten grafiikkaprosessorien ansioista on mahdollista luoda myös interaktiivinen, dynaaminen partikkeliefekti, jolloin muut objektit pelissä vaikuttavat hiukkasiin. (Yang, Yip and Xu, 2009.) On myös mahdollista luoda varjostustehoste partikkeleille, jolloin voidaan

korostaa hiukkaspilven tiheyttä ja muotoa. (Green, 2012). Nykyään voidaan myös hallita sitä, luovatko hiukkaset varjon ympäristöönsä tiettyjen tekniikoiden avulla (Raappana, 2018).

Partikkeliefektiä hyödyntävien ohjelman monimuotoisuus osoittaa tehosteen tärkeyden nykyaikaiselle interaktiiviselle medialle ja pelimaailmaan. Partikkeliefekteillä on lähes rajattomat mahdollisuudet mallintaa erilaisia luonnonilmiöitä ja jopa konkreettisempia esineitä kuten puita. Uusien ja ainutlaatuisten partikkeliefektien luominen vaatii kuitenkin huomattavaa matemaattista ja teknistä osaamista koodaajalta. Sillä saadakseen partikkelien liikkeestä realistisen, tulee pyrkiä stimuloimaan tosielämän fysiikkaa. (Hastings, Guha and Stanley, 2007.)

Partikkeliefektillä siis saadaan luotua luonnollisia ilmiöitä peleihin. Niiden tarkkuus ja hienovaraisuus riippuu pitkälti minkälaiseen peliin ja pelikoneeseen sitä käytetään ja mitä ilmiötä tehoste imitoi. Rosoinen partikkeliefekti voi riittää hyvin pienempi graafiseen peliin ja sillä voidaan imitoida vaikkapa veriroiskeita tai veden purkausta. Hienovaraisemmalla voidaan imitoida rauhallista savun liikettä ja siten luoda rauhallista tunnelmaa peliin. Partikkeliefektin tarkkuuteen vaikuttaa paljon käytössä olevan laitteen suorituskapasiteetti, sillä, mitä realistisemman tehosteen haluaa, sitä enemmän tehoja sen suoritus vaatii ja mitä enemmän tehoja vaaditaan, sitä kalliimpi järjestelmä tulee olemaan. Monet pelimoottorit tarjoavat keinoja luoda partikkelisysteemi peliin mutta aivan uusien partikkeliefektien kehittäminen vaatii hyvää ymmärrystä tosielämän fysiikasta ja matemaattisista malleista, jolla tämä ilmiö saadaan toteutettua peliin.

3.5.3 Sensoritehosteet

Tietokonegrafiikassa käytetään niin kutsuttua neulansilmä-kameramallia, jolla saadaan syntetisoitua teräviä ja täydellisiä kuvia. Tähän liittyy kuitenkin kohdistusvirhe, jonka vuoksi kuvat näyttävät pelaajan silmissä epätäydellisiltä, sillä kamera ei ota huomioon tosielämän optisia ja mekaanisia rajoituksia, joita ihmissilmään liittyy. Kuitenkin kolmella tekniikalla voidaan simuloida paremmin näköjärjestelmää ja luoda realistisempi vaikutelma

ja mahdollistaa myös taiteellinen näkökulma. Nämä tekniikat ovat syväterävyys, liike-epätarkkuus ja korkea dynaaminen alue. (Yang, Yip and Xu, 2009.)

Syväterävyydellä voidaan imitoida silmän tai linssin kykyä tarkentaa näköä tiettyyn pisteeseen siten, että tarkennetun kohteen ympärillä olevat asiat sumenevat. Liike epätarkkuudessa tekniikoilla näkökenttä saadaan sumenemaan reunoilta, jolloin syntyy realistisempi tuntemus liikkeellä olemisesta. Korkeadynaamisilla tekniikoilla korostetaan tummien ja kirkkaiden yksityiskohtien kontrastia, jolloin voidaan luoda mielikuva häikäisevästä valosta tai auringon valon heijastumasta lumihangesta.

3.6 Mahdollisuudet

Tarjolla onkin siis paljon mahdollisuuksia rakentaa vaikuttava ja näyttävä peli, mutta mitä realistisemmista tehosteista on kyse, sen enemmän tehoja suorittavalta koneelta vaaditaan. Kuitenkin luovat ja innovatiiviset tavat mahdollistavat upean pelin luomisen myös niille laitteille, joiden suoritustehot eivät ole yhtä tehokkaat (Hasu, 2018). On siis mahdollista, että pelit pienemmällä grafiikka määrällä voivat tehosteiden avulla olla vaikuttavia siinä missä erittäin realistisetkin pelit. Tällaisia pelejä on vuosien varrella ollut paljonkin, kuten Thomas Was Alone (2012) ja Geometry Dash (2013), jotka ovat olleet aikanaan pelaajien suosiossa. Tehdessä peliä ilman tai vähäisellä piirretyllä grafiikalla, tulee pohtia, miten erilaisilla tehosteilla saadaan tuotua haluttu tunnelma ja näyttävyys peliin, kun piirretty grafiikka itsessään ei riitä sitä tekemään.

4 Pelin synty ja tulokset

Tässä luvussa kerrotaan, miten peli syntyi ja millaisia visuaalisia tehostetita pelissä hyödynnettin. Kerrotaan myös vaikeuksista, joita työtä tehdessä kohdattiin.

4.1. Piirretty grafiikka

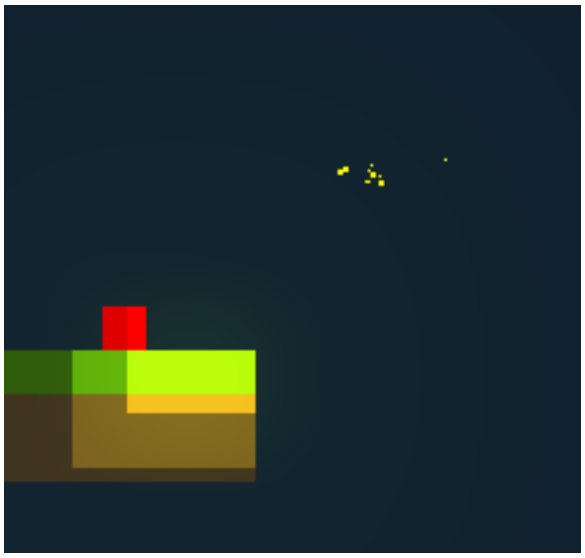
Peli syntyi pienin askelin vaihe vaiheelta. Pelille suunniteltiin pieni graafinen värikokoelma yhdeksästä erivärisestä neliöstä, jotka lopulta karsittiin seitsemään väriin selkeyden vuoksi. Nämä neliöt muodostavat pelin ainoan piirretyn grafiikan. Väreillä pyrityään luomaan pelin tunnelmaa ja odotuksia. Ympäritsö piirrettiin hyödyntäen vihreitä, harmaita, sinisiä ja oransseja laatikoista. Sininen ja vihreä luovat kuvan maasta ja taivaasta. Harmaa ja oranssi luovat kuvaa palavasta maan ytimestä. Punainen väri valittiin pelattavalle hahmolle, jotta se näkyisi paremmin taustasta. Pelin viholliset ovat mustia laatikoita, jotka sulautuvat hyvin iltahämärään maisemaan. Iltahämärän vaikutelma luotiin Godot Enginen tarjoaman kanvasmoduulin (CanvasModulate) avulla. Keltainen valittiin aarteen väriksi, jotta se muistuttaisi kultaa. Alunperin nämä neliöt olivat 1 pikselin kokoisia, mutta nämä olivat niin pieniä, että niitä oli vaikea käsitellä Godot ympäristössä, siispä laatikoiden kokoa kasvatettiin 16x16 pikselin kokoisiksi neliöiksi.

4.2. Tehosteet ja rakenne

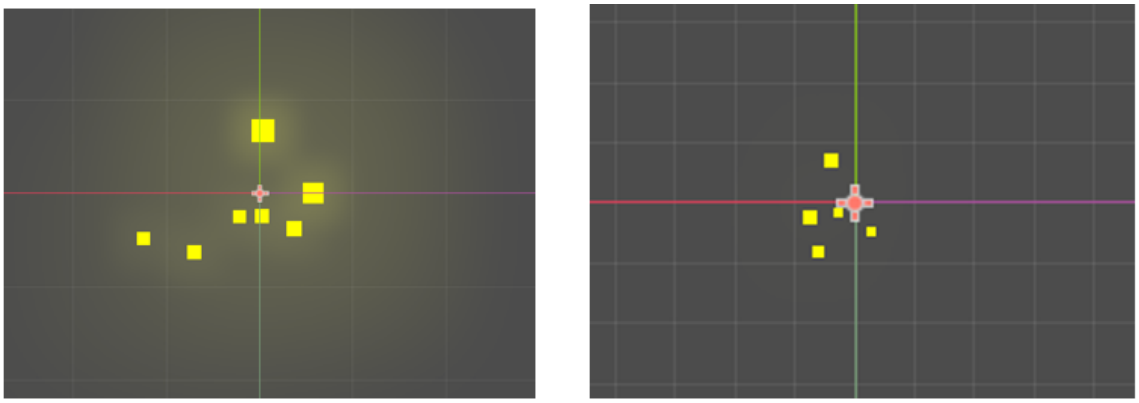
Pelin aloitusvalikko (kuva 1) on rakennettu siten, että väreillä pyritään kertomaan käyttäjälle, mistä napista aloitetaan peli ja mistä suljetaan. Godot Engine tarjoaa maailmaympäristö tehosteen (WorldEnvironment), jonka avulla napeille voitiin luoda hehku tehoste (Glow), jota korostetaan Godot Enginen tarjoamalla valaistustyökalulla (PointLight2D). Tällä tavoin saatiin luotua vaikutelma yötävasten hehkuvasta objekteista. Kursorin ollessa napin päällä, valo kirkastuu merkiksi siitä, että kohde on valittavissa.

Kuva 1. Valikko.

Hämärään maisemaan valonlähteeksi luotiin tulikärpäsparvia (kuva 2) Godot Enginen tarjoaman partikkeliefektin (GPUParticles2D) ja valaistustyökalun avulla. Pienet keltaiset pisteet pörräävät ilmassa valaisten porrastetusti samalla pienen alueen kenttästä. Hehku tehosteen avulla yritettiin luoda paremmin vaikutelmaa siitä, että hehku tulee tulikärpäsistä, mutta tulikärpästen pikselit ovat niin pienet, että hehku ei näy kaukaa (kuva 3).

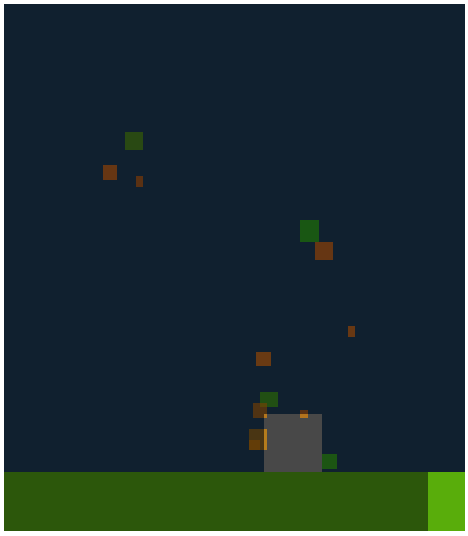


Kuva 2. Tulikärpäset.



Kuva 3. Tulikärpäset läheltä ja kaukaa.

Vihollisten mielentilan kuvaamiseen hyödynnettiin kahta partikkeliefektiä: Toinen kuvaa rauhallista vihollista (kuva 4) ja toinen vihaista (kuva 5). Kun vihollinen on rauhallinen, siitä irtoaa vihreän sävyisiä pikseleitä rauhalliseen tahtiin ylöspäin, mutta kun pelaaja astuu vihollisen hyökkäyetäisyydelle, muuttuvat pikselit punaisiksi ja partikkelien määrä kasvaa. Mikäli pelaaja poistuu hyökkäyetäisyydeltä vihollinen rauhoittuu ja pikselit palaavat vihreiksi. Viholliselle luotiin myös kuolema animaatio (Kuva 6) partikkeliefektin avulla. Jos pelaaja hyppää vihollisen päälle, syntyy pieni räjähdys aalto ja sitten vihollinen katoaa kokonaan.

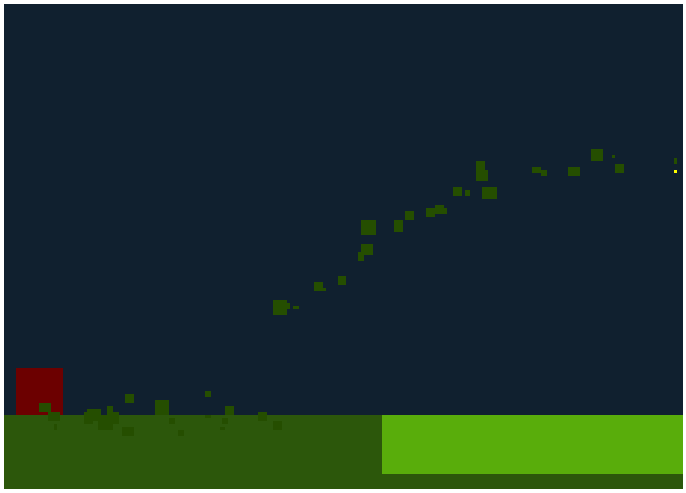


Kuva 4. Rauhallinen vihollinen.

Kuva 5. Jahtaava vihollinen.

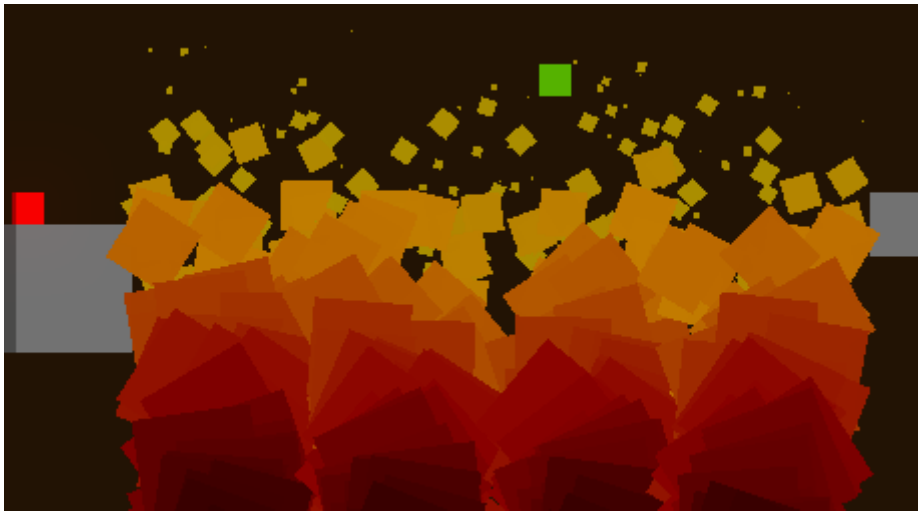
Kuva 6. Kuoleva vihollinen.

Punainen liikutettava neliö sinkoaa liikkuessaan ilmaan vihreitä partikkeleita (kuva 7). Tämän tehosteen avulla haluttiin luoda mielikuvaa liikkeestä ja vauhdista sekä lisätä käsitystä siitä, mihin suuntaan neliö katsoo. Lisäksi Godot Engine tarjoaa kamera tehosteen (Camera2D), jonka avulla peli keskitettiin pelattavaan hahmoon. Kuva seuraa punaista neliötä ja nopeassa liikkeessä, kuten pudotuksessa, kuva tulee hieman jäljessä. Tällä tavalla luotiin mielikuvaa liikkeestä ja näkökyvyn epätarkkuudesta. Kyettiin myös rajamaan kentän näkyvyyttä siten, että tiettyjen koordinaattien jälkeen kameran kuva ei enää liiku, jolloin pelaaja ei näe pelirajojen yli.

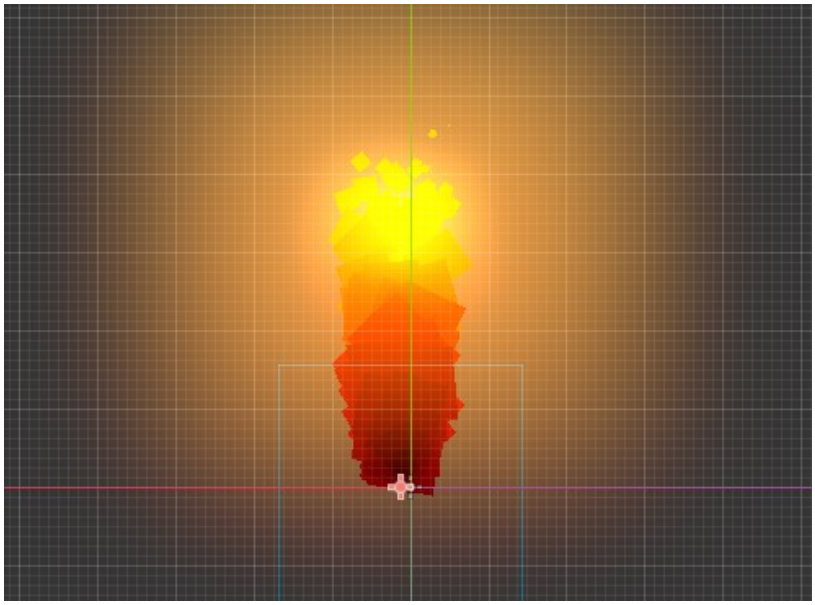


Kuva 7. Juokseva hahmo.

Maanalaiseen maailman luotiin tulia (kuva 8) partikkeliefektin avulla, mihin pudotessa pelaaja kuolee. Tuli on monimutkaisin partikkeliefekti pelissä ja se luotiin Godot Tutorials:in Fire Effect w/ Particles2D | Godot Basics Tutorial | ep 62 -videon (2020) avulla (*Fire Effect w/ Particles2D | Godot Basics Tutorial | Ep 62*, 2020). Näiden tulien oli tarkoitus valaista koko maanalainen maailma ympäristö tehosteen hehkuominaisuuden avulla (kuva 9), mutta tätä ei saatu toimimaan, sillä toinen ympäristötehoste kumoaa tulien ympäristötehosteen vaikutuksen eikä Godot Enigine 4 tue tätä tehostetta vielä täysin. Liekit valaisevat pelissä nyt tavallisen valotehosteen avulla.



Kuva 8. Liekit



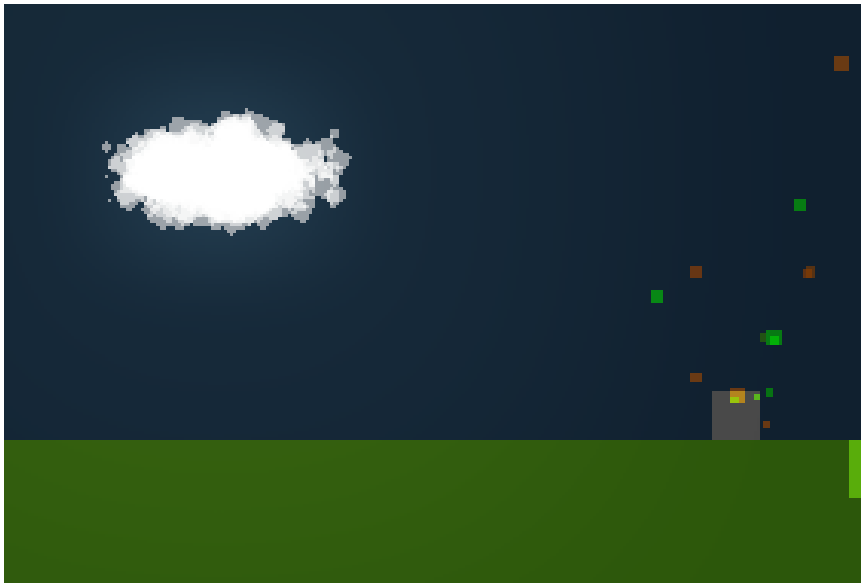
Kuva 9. Liekki

Aarteeseen (kuva 10), luotiin sekä valasitus että partikkeliefekti, jotka tuovat vaikutelmaa hehkuvasta ja loistavasta aarteesta. Lisä vaikutuksen luomiseksi lisättiin lyhyt animaatio ajastin ominaisuuden (Timer) avulla, jonka seurauksena aarre nousee ilmaan ennen kuin katoo.

Kuva 10. Aarre ennen ja jälkeen keräyksen

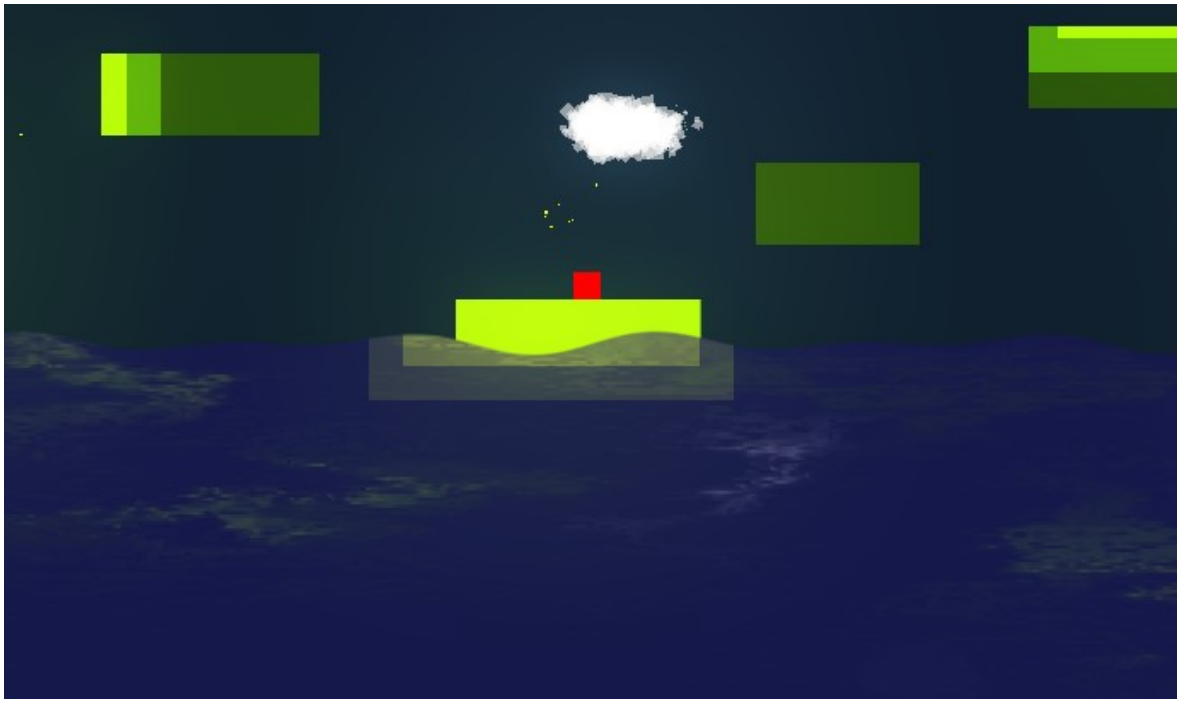
Peliin lisättiin myös pilviä (kuva 11) taivaalle partikkeliefektin avulla. Nämä pilvet sijaitsevat eri tasolla kuin muut pelin tehosteet. Tämän seurauksena aikaisempi kanvas moduuli tehoste, jolla luodaa hämärän vaikutelman peliin, ei vaikuta pilviin ja pilvet vaikuttavat hehkuvan taivaalla. Tämö pilvien hehku lisää dramaattisesti yön vaikutusta.

Pilvien liike luo myös tunnelmaa dynaamisesta ympäristöstä. Valitettavasti pilviä ei kuitenkaan saatu pysymään vain taivaalla, vaan ne seuraavat hahmoa myös maan alle.



Kuva 11. Pilvi

Peliin luotiin lopuksi vettä (kuva 12) varjostaja omianisuuden (shader) avulla. Varjostaja on muihin tehosteisiin verrattuna paljon koodipainoitteisempi ja sillä onkin Godot Enginessä oma editori. Käyttäjän tulee itse määritellä varjostimelleen parametreja, minkä vuoksi käyttäjän olisi hyvä tuntea tämä tehoste hyvin. Tehosteeseen luotiin meteli- (noise) sekä aaltotehosteet (wave) määrittelemällä itse koodin puolella tarvittavat parametrit, jonka jälkeen parametreja pystyi säätämään manuaalisesti. Tehostetta tehdessä, yritettiin seurata Nolkaloidin GODOT TUTORIAL: 2D Reflective water shader – videota (2020), mutta tämä ei onnistunut eri Godot Engine version vuoksi täysin. Videolta kuitenkin saatiin tärkeimmät pääpiirteet heijastavan vesi varjositimen rakentamiseen selville ja vesi elementti saatiin rakennettua peliin. (*GODOT TUTORIAL : 2D Reflective water shader*, 2020.)



Kuva 12. Vesi.

Pelistä muodostui lopulta lyhyt ja yksinkertainen tasohyppelypeli, jossa on jahtaavia vihollisia ja kerättävä aarre. Pelaajan on päästävä kentän loppuun kuolematta ja kun pelaaja kerää kentän lopussa olevan aarteen, peli päätty. Pelissä on käytetty enimmäkseen valaistustehosteita ja partikkeliefektejä, mutta myös yksi varjostaja saatiin tehtyä ja pieni kameran viive liikkeessä liitettiin peliin.

Peliä tehdessä havaittiin, että oli vaikea löytää tasapainoa visuaalisten tehosteiden määrän suhteen, jotta kokonaisuus olisi vaikuttava ja silmää miellyttävä. Havaittiin myös, että joissain tapauksissa visuaaliset tehosteet edesauttoivat pelattavuutta. Tulikärpäset ikään kuin valaisivat pelaajan reitin luolaan ja auttavat havaitsemaan viholliset. Myöskin visuaalisilla tehosteilla pystyttään ilmaisemaan kasvottoman pelihahmon katseen suunta ja vihollisten mielentila.

5 Tulosten arviointi

Tässä luvussa perehdytään tarkemmin SWOT-taulukon avulla tuloksiin ja huomioihin, joita tehtiin peliä tehdessä. Tässä luvussa myös arvioidaan visuaalisten tehosteiden uhkia ja mahdollisuuksia tämän taulukon avulla.

5.1. SWOT-Analyysi

Pelin luonnin yhteydesä havaittiin paljon mahdollisuuksia sekä ongelmia, kun piirrettyä grafiikka korvattiin visuaalisilla tehosteilla. Alla olevaan SWOT-taulukkoon on koottu pelin luonnin yhteydessä havaitut vahvuudet, heikkoudet, mahdollisuudet ja uhat, kun piirrettyä grafiikkaa korvattiin visuaalisilla tehosteilla.

Taulukko 2: SWOT-analyysi grafiikan korvaamisesta visuaalisilla tehosteilla.

Vahvuudet - Helpottaa varjojen ja valojen luontia - Mielikuvitus rajana partikkeliefektien käytössä - Monipuolisia tehosteita ympäristön muokkaamiseen ja tehostamiseen - Realismin lisäys ja tunnelman luonti	**Heikkoudet** - Kentän luominen haastavaa - Toisinaan tehosteet kumoavat toisensa - Vaatii joidenkin tehosteiden osalta hyvää asiantuntemusta
Mahdollisuudet - Voidaan luoda vaikuttava dynaaminen kokonaisuus vähillä resursseilla - Realismin lisäys yksinkertaisin keinoin - Pelattavuuden paraneminen	**Uhat** - Monimutkaiset yhdistelmät vaikea saada toimimaan - Peli voi kadota massaan - Liiallinen tai liian vähäinen käyttö voi johtaa kömpelöön lopputulokseen - Heikko koodaus johtaa heikkoon pelattavuuteen

5.1.1 Vahvuudet ja mahdollisuudet

Visuaalisilla tehosteilla on hyviä vahvuuksia piirretyn grafiikan korvaamisessa. Valo- ja varjotehosteilla voidaan helpottaa varjojen ja valojen luontia peliin ja näitä ei tarvitse erikseen huomioida grafiikan piirtämis- ja suunnitteluvaiheessa. Valoilla ja varjoilla voidaan luoda nopeasti ja tehokkaasti dramaattista vaikutusta pelin tunnelmaan ja valojen ja varjojen määrällä voidaan vaikuttaa tunnelman raskauteen kätevästi (Feldman, 2001).

Godot Engine -pelimoottori tarjoaa useita erilaisa keinoja luoda valoja ja varjoja peleihin. Tässä työssä käytettiin vain yhtä valotehostetta, mutta tämän pistevalon lisäksi Godot Engine tarjoaa 2D-pelejä varten suuntavalon (DirectionalLight2D) ja varjostimen (LightOcclucer2D). 3D-pelejä varten Godot engine tarjoaa hieman monimutkaisempia valojen ja varjojen lähtietä.

Peliä tehdessä havaittiin, että partikkeliefektit mahdollistavat kehittäjälle lukemattomia mahdollisuuksia stimuloida luonnonilmiöitä monipuolisesti ja todentunutisesti. Godot Enginen tarjoamassa partikkeliefektissä on runsaasti ominaisuuksia, joita säätää ja muokata, jotta kehittäjä saa luotua juuri sellaisen tehosteen kuin haluaa. Godot pelimoottori tarjoaa 2D-peleille kaksi vaihtoehtoista tapaa luoda partikkeliefektejä. Näiden ero on resursseissa, joita ne käyttävät. Godotin omilla dokumentti sivuilla (2014) avataan tarkemmin näiden kahden eroja: Grafiikkaprosessointiyksikköä, GPU:ta, käyttävä partikkelitehoste on näistä kahdesta kehittyneempi ja keskusprosessoriyksikköä, CPU:ta käyttävä partikkelitehoste on näistä kahdesta prosessoriohjatumpi vaihtoehto. CPU:ta käyttävällä partikkeliefektillä on lähes samat ominaiusudet kuin GPU:ta käyttävällä tehosteella, mutta sen suorituskyky on alhaisempi, mitä enemmän hiukkaisa käytetään, mutta voi olla resurssien kannalta resurssitehokkaampi vaihtoehto. (Godot Community, 2014.)

Peliin on myös mahdollista luoda realistista tunnelmaa yksinkertaisin keinoin visuaalisten tehosteiden avulla. Valoilla ja varjoilla voidaan lisätä pelin realistista ulkonäköä luomalla valonlähtietä, heijastuksia ja varjoja (Yang, Yip and Xu, 2009). Godot engine tarjoaa myös erilaisia ymprätistötehosteita ja varjostajan realismin lisäämiseksi. Godot Engine tarjoaa muun muassa maailmaympäristö tehosteen (WorldEnvironment), jonka avulla kehittäjän on mahdollista luoda ympäristöön muun muassa usvaa ja hehkuvia objekteja. Kanvasmoduulin (CanvasModulate) avulla kehittäjä voi vaikuttaa koko kohtauksen väriin. Tähän työhön tehtyä peliä varten, iltahämärä luotiin nimenomaan kanvasmoduulin avulla. Varjostaja

(Shader) mahdollistaa näkyvyyden epätarkkuuden ja valaistusharhojen stimuloimisen. Varjostajalla voidaan luoda esimerkiksi liikkuvaa, heijastavaa vettä, niin kuin luodussa pelissä, mutta voidaan myös muokata koko pelin värimaailmaa tai luoda esimerkiksi mielikuva paahtavasta ympäristöstä luomalla sumeaa ja epätarkkaa näkyvyyttä. Visuaalisten tehosteiden käytössä vain mielikuvitus on rajana (Yang, Yip and Xu, 2009).

Visuaalisilla tehosteilla on mahdollista luoda yksittäisiä kokonaisuuksia, joita voidaan liittää sellaisenaan peliin. Godot engine mahdollistaa tämän erilaisten tehosteiden yhdistämisen, jolloin voidaan luoda vaikuttavia kokonaisuuksia. Työhön tehdyssä pelissä yhdisteltiin paljon partikeliefektejä ja valaistutehosteita keskenään ja luotiin siten ilmiöitä loistavista objekteista. Godot engine tarjoaman kameraominaisuuden (Camera2D) avulla voidaan vaikuttaa siihen, miten kuva seuraa pelaajaa. Kameran liikkeellä ja partikkeliefektillä rakennetuista pilvistä, voidaan luoda vaikutelma tuulesta ja dynaamisesta ympäristöstä.

Visuaalisilla tehosteilla on myös mahdollista parantaa pelin pelattavuutta. Visuaaliset tehosteet voivat muun muassa valaistuksen avulla helpottaa pelaajan näkökykyä pelissä tai visuaalisilla tehosteilla voidaan viestiä pelaajan kanssa muun muassa antamalla varoituksia tai vihjata oikeasta suunnasta. Kokonaisuudessaan kaikilla esitellyillä visuaalisilla tehosteilla voidaan tehokkaasti ja yksinkertaisesti lisätä pelin realistista ilmettä ja näyttävyyttä.

5.1.2 Heikkoudet ja uhat

Visuaalisilla tehosteilla on myös jonkin verran heikkouksia ja uhkia liittyen grafiikan korvaamiseen. Työssä toteutetussa tasohyppelipelissä itse kentän grafiikkaa oli mahdoton korvata pelkillä visuaalisilla tehosteilla ilman, että kokonaisuudesta olisi syntynyt sotkuinen ja empäköytännöllinen. Tämän vuoksi kentän luonnissa käytettiinkin 16x16 pikselin kokoisia neliöitä eikä esimerkiksi luotu tasoja partikkeliefekteillä. Visuaalisilla tehosteilla voi siis olla vaikea korvata itse pelin runkoa. Voi myös käydä niin, että visuaaliset tehosteet kumoavat toisensa, jolloin ne eivät toimi oikein. Näin kävi esimekrkiksi tähän peliin tehtyjen liekkien hehkun suhteen. Godot Engineä käyttäessä, jotkin ominaisuudet siis kumoavat toisensa, jos esiintyvät väärässä järjestyksessä tai väärässä paikassa. Visuaalisisten

tehosteiden käyttö vaatii selkeää ymmärrystä siitä, miten tehosteet toimivat, jotta ne saadaan toimimaan yhdessä.

Uhkiakin piirretyn grafiikan korvaamiseen liittyy. Liiasta visuaalisten tehosteiden käytöstä voi syntyä kaaoottinen ja sotkuinen kokonaisuus, joka ei enää miellytä pelaajien silmää. Toisaalta liian vähäisestä käytöstä voi seurata tylsä ja mitään sanomaton kokonaisuus. Piirretyllä grafiikalla kehittäjät voivat helposti luoda pelilleen uniikkia imagoa ja tämän imagon luominen pelkillä visuaalisilla tehosteilla on erittäin vaikeaa. Luotu peli voikin siis helposti kadota muiden piirretyltä grafiikaltaan uniikimpien ja näyttävämpien pelien sekaan.

Godot pelimoottori tarjoaa runsaasti visuaalisia tehostekeinoja, mutta niiden hydyntämisen onnistuminen on kiinni kehittäjän taidoista yhdistellä osia toisiinsa. Jotta visuaalisilla tehosteilla voidaan luoda vaikuttava kokonaisuus, vaaditaan kehittäjältä hyvää tehosteiden mahdollisuuksien ja toiminnan tuntemusta. Aloittelijan voi olla vaikea luoda vaikuttava peli pelkkien tehosteiden nojalla. Toisaalta pelimoottorin uusimmat versiot eivät välttämättä tue kaikkia niitä ominaisuuksia, joita ovat aikaisemmin tukeneet, joten kehittäjän tulee opetella uusien versioiden käyttötavat, mikäli mielii käyttää päivitettyä versiota moottorista.

Tehosteilla luotuun peliin voi liittyä paljon ongelmia myös persutasolla, jos itse koodaus tehosteiden takana on heikkoa. Heikosti koodatut tehosteet voivat syödä paljon resursseja ja silloin pelistä tulee todennäköisesti erittäin hidas ja pelin pyörittäminen voi olla jopa mahdotonta. Jos kehittäjä ei osaa koodata tehosteita kunnolla, voi myös seurata tilanne, jossa peli toimii hyvin yhdellä koneella, mutta ei juuri ollenkaan toisella koneella. Kehittäjän on siis osattava suunnitella myös koodi näiden tehosteiden taakse, jotta tehosteet eivät kuluttaisi turhaan resursseja ja pelin pyörisi kunnolla laitteella kuin laitteella.

6 Yhteenveto ja johtopäätökset

Tässä kandidaatintyössä tutkittiin, miten helppoa piirretyn grafiikan korvaaminen visuaalisilla tehosteilla on. Tutkittiin myös, mitä etuja ja haittoja siihen liittyy ja vastattiin tutkimuskysymyksiin: Mitä Godot Engine -pelimoottori tarjoaa vaikuttavan pelin luomiseen ilman piirrettyä grafiikkaa, ja miten visuaalisilla tehosteilla voidaan korvata grafiikan puutetta.

Grafiikan merkitys peleissä on valtava ja pelaajat suosivat visuaalista ilmettä. Piirretyn grafiikan rinnalla on usein erilaisia visuaalisia tehosteita, joilla tehostetaan ja syvennetään piirretyn grafiikan vaikutusta. Grafiikan korvaamista ei ole aiemmin tutkittu visuaalisten tehosteiden näkökulmasta. Kuitenkin on kiinnostavaa tietää, minkälainen merkitys visuaalisilla tehosteilla on piirretyssä grafiikassa ja onko mahdollista korvata piirretty grafiikka kokonaan tehosteilla.

Tutkimusta varten luotiin yksinkertainen tasohyppelypeli Godot Engine -pelimoottorilla. Työssä käytettiin versiota 4.1, joka on yksi Godot Enginen uusimmista versioista. Kentän luomisessa käytettiin mahdollisimman yksinkertaista piirrettyä grafiikkaa, joka muodostui 16x16 pikselin kokoisista neliöistä. Loput ominaisuudet ja tunnelma luotiin erilaisilla visuaalisilla tehosteilla, kuten valo- ja partikkeliefekteillä sekä varjostajalla.

Lopputuloksena saatiin selville, että monipuolisilla visuaalisilla tehosteilla on mahdollista korvata piirrettyä grafiikkaa, mutta visuaalisten tehosteiden runsaaseen käyttöön liittyy riskejä. Visuaalisilla tehosteilla voidaan korvata grafiikan puutetta ja säästää resursseissa lukuisilla tavoin: Voidaan luoda valoja ja varjoja, voidaan stimuloida monenlaisia luonnonilmiöitä, voidaan luoda dynaaminen vaikutelma ja voidaan luoda monipuolisia ja monimutkaisia ympäristöjä ympäristötehosteiden avulla. Visuaalisten tehosteiden liiallinen tai liian vähäinen käyttö voi kuitenkin saada aikaan sotkuisen ja kömpelön lopputuloksen. Visuaalisten tehosteiden potentiaalin optimaaliseen käyttöön vaaditaan ymmärrystä kunkin visuaalisen tehosteen mahdollisuuksista ja toiminnasta.

Saatiin myös selville erilaisia tehosteita, joita Godot Engine -pelimoottori tarjoaa piirretyn grafiikan korvaamiseen. Godot Enginellä on erilaisia valaistus- ja varjostuselementtejä, partikkeliefektejä, ympäristötehosteita sekä varjostaja, joiden käyttämisessä vain

mielikuvitus on rajana pelin luonnissa. Myöskin pelikameran liikkeeseen voidaan vaikuttaa ja siten luoda syvyyttä peliin.

Johtopäätöksenä voidaan kuitenkin todeta, että visuaaliset tehosteet ovat vaikuttava keino, jolla piirrettyä grafiikkaa voidaan korvata. Visuaalisten tehosteiden käyttö vaatii kuitenkin tasapainottelua, hyvää suunnittelua ja taitavaa tehosteiden ymmärrystä ja hyödyntämistä, jotta pelistä tulisi vaikuttava ja pelaajien silmissä houkutteleva. Johtopäätöksenä voidaan myös todeta, että tämän tyyppisessä tasohyppelypelissä, piirretty grafiikka on vaikea korvata täysin. Lisäksi tässä työssä käytettiin vain murto-osaa niistä mahdollisuuksista, joita visuaalisilla tehosteilla voidaan toteuttaa niin yleisesti kuin myös Godot Enginen tasolla. Potentiaalia lisätutkimuksielle löytyy siis paljon. Asiaa voisi tutkia lisää muun muassa toisen tyyppisen pelin näkökulmasta tai hyödyntäen toista pelimoottoria. Yhtä kaikki tämän tutkimuksen tuloksista voivat hyötyä ne pelinkehittäjät, jotka suunnittelevat pelin luomista minimaalisella piirretyllä grafiikalla, tai jotka pohtivat visuaalisten tehosteiden mahdollisuuksia korvata piirrettyä grafiikkaa.

Lähteet

Bradfield, C. (2018) *Godot Engine Game Development Projects: Build Five Cross-platform 2D and 3D Games with Godot 3.0*. Packt Publishing Ltd. ISBN 1-78883-642-1

Cohen, D. *et al.* (2012) *Producing games: from business and budgets to creativity and design*. Routledge. ISBN 0-08-092804-8

Dally, W.J., Keckler, S.W. and Kirk, D.B. (2021) 'Evolution of the graphics processing unit (GPU)', *IEEE Micro*, 41(6), pp. 42–51. Available at: https://ieeexplore.ieee.org/abstract/document/9623445 (Accessed: 23 November 2023).

Dresch, A. *et al.* (2015) *Design science research*. Springer. ISBN 3-319-07373-7

Engström, H., Brusk, J. and Östblad, P.A. (2015) 'A comparison of immersion between players playing the same game with and without graphics', in. *International Conference on Game and Entertainment Technologies*, IADIS Press, pp. 84–92. ISBN 978-989-8533-38-8

Feldman, A. (2001) *Designing arcade computer game graphics*. Wordware Pub. ISBN 1-55622-755-8

Fire Effect w/ Particles2D | Godot Basics Tutorial | Ep 62 (2020). Available at: https://www.youtube.com/watch?v=5RhIXpOuYKo (Accessed: 11 December 2023).

Godot Community (2014) *2D particle systems*, *Godot Engine documentation*. Available at: https://docs.godotengine.org/en/stable/tutorials/2d/tutorials/2d/particle_systems_2d.html (Accessed: 12 December 2023).

GODOT TUTORIAL : 2D Reflective water shader (2020). Available at: https://www.youtube.com/watch?v=32jdNLTJ3zY (Accessed: 24 November 2023).

Green, S. (2012) 'Volumetric particle shadows', *NVIDIA Developer Zone* [Preprint]. Available at: https://developer.download.nvidia.com/compute/DevZone/C/html_x64/5_Simulations/smokeParticles/doc/smokeParticles.pdf (Accessed: 5 October 2023).

Hantula, J. (2018) 'Peligrafiikan kehittyminen'. Available at: http://urn.fi/URN:NBN:fi:jyu-201805252772 (Accessed: 18 October 2023).

Harari, Y.N. (2014) *Sapiens: a brief history of humankind by Yuval Noah Harari*. New York : Harper. ISBN 978-0-06-231609-7

Harari, Y.N. (2021) *Sapiens A Graphic History, Volume 2: The Pillars of Civilization*. Random House. ISBN 1-4735-9814-1

Hastings, E., Guha, R. and Stanley, K.O. (2007) ‘Neat particles: Design, representation, and animation of particle system effects’, in. *2007 IEEE Symposium on Computational Intelligence and Games*, IEEE, pp. 154–160. ISBN 1-4244-0709-5

Hasu, J. (2018) ‘Fundamentals of shaders with modern game engines’. Available at: https://urn.fi/URN:NBN:fi-fe2018110547290 (Accessed: 30 September 2023).

Hevner, A.R. *et al.* (2004) ‘Design Science in Information Systems Research 1’, *MIS Quarterly*, 28(1), pp. 75–105. Available at: https://www.proquest.com/docview/218119584/abstract/2797A34B5CD34C72PQ/1 (Accessed: 17 October 2023).

Kallio-Kujala, J. (2016) ‘Pelimoottori opetuskäyttöön’. Available at: https://urn.fi/URN:NBN:fi:amk-2016120419018 (Accessed: 12 October 2023).

Mansikka, H. (2015) ‘2D-ja 3D-peligafiikan ominaisuudet’. Available at: https://urn.fi/URN:NBN:fi:amk-201504073979 (Accessed: 22 September 2023).

Masuch, M. and Röber, N. (2004) ‘Game graphics beyond realism: Then, now and tomorrow’, in. *Level UP: digital games research conference. DIGRA, Faculty of Arts, University of Utrecht*. Available at: http://www.digra.org/wp-content/uploads/digital-library/05150.48223.pdf (Accessed: 18 October 2023).

Mustonen, M. (2023) *WEB BASED GAME ENGINE DESIGN*. Lappeenranna-Lahden Teknillinen Yliopisto. Available at: https://lutpub.lut.fi/handle/10024/166230 (Accessed: 28 November 2023).

Östblad, P.A. *et al.* (2014) ‘Inclusive game design: audio interface in a graphical adventure game’, in. *Proceedings of the 9th audio mostly: A conference on interaction with sound*, pp. 1–8. Available at: https://dl.acm.org/doi/abs/10.1145/2636879.2636909 (Accessed: 13 October 2023).

Peffers, K. *et al.* (2007) ‘A design science research methodology for information systems research’, *Journal of management information systems*, 24(3), pp. 45–77. Available at: https://www.tandfonline.com/doi/abs/10.2753/MIS0742-1222240302 (Accessed: 16 October 2023).

Phadermrod, B., Crowder, R.M. and Wills, G.B. (2019) ‘Importance-performance analysis based SWOT analysis’, *International journal of information management*, 44, pp. 194–203. Available at: https://www.sciencedirect.com/science/article/abs/pii/S0268401216301694 (Accessed: 29 November 2023).

Raappana, J. (2018) ‘Great Particles and How to Make Them: Development of guidelines for the Unity Particle System’. Available at: https://urn.fi/URN:NBN:fi:amk-2018120319610 (Accessed: 9 October 2023).

Rogers, S. (2014) *Level Up! The guide to great video game design*. John Wiley & Sons. ISBN 1-118-87719-5

Salomon, D. (2011) *The computer graphics manual*. Springer Science & Business Media. ISBN 0-85729-886-0

Sarsby, A. (2016) *SWOT analysis*. Lulu. com. ISBN 0-9932504-2-4

Teoli, D., Sanvictores, T. and An, J. (2019) 'SWOT analysis'. Available at: https://www.ncbi.nlm.nih.gov/books/NBK537302 (Accessed: 29 November 2023).

Tienari, A. (2017) 'Grafiikan kehittyminen ja sen näkyvyys digitaalisissa peleissä'. Available at: https://jyx.jyu.fi/handle/123456789/54772 (Accessed: 18 October 2023).

Yang, X., Yip, M. and Xu, X. (2009) 'Visual effects in computer games', *Computer*, 42(7), pp. 48–56. Available at: https://doi.org/10.1109/MC.2009.240.

Zhang, B. and Hu, W. (2017) 'Game special effect simulation based on particle system of Unity3D', in. *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, IEEE, pp. 595–598. ISBN 1-5090-5507-X