



Tiina Hautamäki

Parallax-efekti digitaalisessa kuvituksessa

Metropolia Ammattikorkeakoulu

Medianomi (AMK)

Viestinnän tutkinto-ohjelma

Opinnäytetyö

5.5.2023

Tiivistelmä

Tekijä(t):	Tiina Hautamäki
Otsikko:	Parallax-efekti digitaalisessa kuvituksessa
Sivumäärä:	43 sivua + 1 liite
Aika:	05.05.2023
Tutkinto:	Medianomi (AMK)
Tutkinto-ohjelma:	Viestinnän tutkinto-ohjelma
Suuntautumisvaihtoehto:	Visuaalinen viestintä
Ohjaaja(t):	Lehtori Lauri Huikuri

Tämä opinnäytetyö käsittelee parallax-efektin käyttöä digitaalisessa kuvituksessa eri verkkoalustoilla. Työssä tarkasteltiin autoetnografisella tutkimusotteella efektin erilaisia käyttötapoja ja toteutuksen mahdollisuuksia verkkopohjaisen koodauksen avulla sekä valmiilla alustoilla ja applikaatioilla. Opinnäytetyö koostuu kahdesta osasta, teoriaosuudesta ja teososasta.

Teoriaosuudessa tutustuttiin kameran liikeratoihin parallax-efektin luomisessa elokuvateorian lähdekirjallisuutta hyödyntäen. Tämän lisäksi tarkasteltiin erilaisia visuaalisia esimerkkejä parallax-efektin käytöstä verkossa. Huomioitiin, että elokuvatekniset ratkaisut pätevät myös digitaalisessa ympäristössä. Parallax-efektillä voidaan myös edistää visuaalista tarinankerrontaa sivuston käyttötarkoitukselle sopivalla tavalla, esimerkiksi verkkoartikkelissa tai töitä esittelevässä portfolioissa.

Teososuudessa tutustuttiin erilaisiin teknisiin vaihtoehtoihin parallax-efektin luomiseksi verkossa. Tarkoituksena oli luoda interaktiivinen ja animoitu parallax-efektiä sisältävä kuvitus. Kokeiluja tehtiin verkkopohjaisella koodauksella, Facebookin 3D-generaattoria hyödyntäen sekä interaktiivisuuteen erikoistuneilla applikaatioilla. Viimeisenä testattiin vielä Adobe Animate -ohjelmiston VR360-työskentelytilaa.

Todettiin, että verkkopohjaisella koodilla luotu parallax-efekti toimii pääasiassa hyvin, mutta osa ominaisuuksista ei enää toiminut sille suunnitellulla tavalla. Koodi oli ilmeisesti vanhentunutta, eivätkä kaikki laitealustat olleet sen kanssa enää yhteensopivia. Facebookin automaatio toimi teknisesti loistavasti, mutta visuaalinen lopputulos ei ollut niin hiottu kuin olisi ollut toivottavaa. Erikoistuneemmat applikaatiot toimivat hyvin, mutta vaativat teokselle oman julkaisualustansa. Adobe Animate osoittautui hyväksi tavaksi luoda sekä toimiva että näyttävä kuvitus, kunhan kameran liikettä ensin hienosäädettiin. Opinnäytetyön tuloksista todettiin, että verkkoalustat muuttuvat alin-omaa, eikä yhtä varmaa ratkaisua kuvituksen luomiselle tässä vaiheessa ole. Tekniikasta kiinnostuneen tulee jatkossakin seurata kehityksen kulkua säännöllisesti.

Avainsanat: parallaksi, kuvitustaide, javascript, tilt parallax, scroll parallax

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author(s): Tiina Hautamäki
Title: Parallax Effect in Digital Illustration
Number of Pages: 43 pages + 1 appendix
Date: 5 May 2023

Degree: Bachelor of Culture and Arts
Degree Programme: Media
Specialisation option: Visual Communications
Instructor(s): Lauri Huikuri, Senior Lecturer

This final project explores the use of parallax effect in digital illustrations on different online platforms. An autoethnographic approach was used to study diverse ways and possibilities to use the parallax effect with the help of web-based coding and ready-made online platforms and applications. This thesis consists of two parts.

In the theoretical section, different types of camera movements were examined and how they are being used to create the parallax effect in films. Source literature for cinematography was essential to gain more understanding of the subject. In addition, various examples of the use of the parallax effect online were examined. It was noted that the solutions used in cinematography are just as essential in a digital environment.

In the practical section, different technical options were explored. The purpose of this project was to create a digital, animated illustration that also included the parallax effect. This was achieved with the help of web-based coding, using Facebook's automated 3D generator, and various applications that are specialised in using the parallax effect. Finally, the VR360 mode of Adobe Animate was tested.

It turned out that while web-based coding worked well in most parts, it was not completely trustworthy. Some of its features were not working the way they were meant to, as the code seemed to be somewhat outdated for modern systems and was no longer compatible with every device. Facebook's automated generator worked perfectly on a technical level, but the visual quality left much to be desired. The specific applications were satisfactory, but they often require a very specific type of platform to display the final product correctly. Adobe Animate turned out to be the best candidate to create working and visually pleasing projects after some tweaking. In conclusion, it could be said that online platforms are still evolving at a constant pace and there is yet to exist one, conclusive way to create an illustration with working parallax effect. Anyone who is interested in this technology should keep a close watch on its evolution on a regular basis.

Keywords: parallax, tilt parallax, scroll parallax, illustration

Sisällys

1	Johdanto	1
2	Parallax-efekti	2
2.1	Mikä on parallax	2
2.2	Kameran liike	5
2.3	Polttoväli	8
2.4	Point of interest	9
2.5	Syväterävyys	10
3	Parallax-efektin käyttötavat selaimessa	11
3.1	Tilt parallax	11
3.2	Scroll parallax	14
4	Tekninen toteutus ja eri mahdollisuudet	18
4.1	Julkaisutavat	18
4.2	Valmiit alustat	19
4.3	CSS ja JavaScript	21
5	Oma kuvitusprojektini	22
5.1	Suunnitteluprosessi	22
5.2	Testikuvitus	23
5.3	Koodaus verkkosivuille	25
5.4	Facebookin syvyyskartta	31
5.5	Applikaatiot ja editorit	34
5.5.1	Rive	34
5.5.2	Mental Canvas	35
5.5.3	Adobe Animate	37
6	Lopullinen kuvitus	38
7	Lopuksi	39
	Lähteet	40
	Liitteet	44
	Linkki valmiiseen parallax-kuvitukseen videomuodossa	44

1 Johdanto

Digitaalisuus ja taiteen ja kuvituksen siirtyminen tietokoneen ja puhelimen ruuduille paperiprinttien sijaan on yhä enemmän arkipäivää ja hyvin ajankohtainen haaste tällä alalla. Kuvituksella saadaan aikaan näyttäviä visuaalisia tehokeinoja verkkoartikkeleihin, mainoksiin ja applikaatioihin. Interaktiiviset taidenäyttelyt ovat jo mahdollisia verkosta käsin ja todennäköisesti yleistyvät entisestään, mikäli *kryptotaide* ja *NFT* tulevat jäädäkseen.

Tästä pääsen monimuotoisen opinnäytetyöni aiheeseen, parallaksiin. *Parallaksi* on monisyinen termi, joka ulottuu niin astrologiaan kuin taiteeseenkin. Tässä yhteydessä keskityn parallaksiin median visuaalisena tehokeinona, jolla on tarkoitus luoda syvyyden illuusiota kaksikulotteiseen kuvaan. Sitä käytetään niin animaatioissa, videopeleissä kuin verkkosivustoillakin, applikaatioista puhumattaakaan. Viime aikoina se on kuitenkin yleistynyt myös digikuvittamisessa, ja osaava tekijä voi luoda näyttävän portfolion parallax-efektiä käyttäen. Kuvitukset voivat olla kokonaisuudessaan 2D-grafiikkaa, mutta ne voivat sisältää myös 3D-mallinnuksia lisämausteena. Jarom Vogel on yksi kuvitustyylin tunnetuista edelläkävijöistä, ja hän on tehnyt aiheesta myös tutoriaaleja. Hänen kuvituksensa ovat inspiroineet myös minua tutustumaan tekniikkaan tarkemmin.

Vaikka Vogel käyttää töissään pitkälti koodaustaitojaan, on efekti mahdollista nykyään luoda myös mobiiliapplikaatioilla ja sosiaalisen median alustoilla. Opinnäytetyöni tarkoituksena onkin tutustua näihin eri vaihtoehtoihin ja projektiosassa lopulta löytää toimiva keino esittelemään *parallax-kuvitusta* esimerkiksi omalla verkkosivustolla. Käytän itse tekemääni digitaalista kuvitusta ja kokeilen sen käyttömahdollisuuksia eri alustoilla. Miten parallax toimii esimerkiksi *JavaScriptin* avulla? Miten kuvitukseen saa interaktiivisuutta, jotta se liikkuisi vaikka puhelimen näyttöä koskettamalla? Entä miten *parallax-efektiä* voi luoda erilaisissa applikaatioissa? Dokumentoin työvaiheitani ja havaintojani päiväkirjatyypisesti työn loppuosassa autoetnografista tutkimusmenetelmää hyödyntäen.

Rajaan kuitenkin käsittelyn ulkopuolelle joitakin yksityiskohtia eri teknisistä ratkaisuista. Erityisesti JavaScript-koodauksen toiminnallisuuksia selitän mahdollisimman yksinkertaisesti, sillä en ole vielä kovin kokenut koodaaja, eikä koodin perusteellinen avaus olisi kovin hyödyllistä tämän työn kokonaiskuvaan ajatellen. Sivuutan myös 3D-grafiikan käytön kuvituksissa, vaikka tämäkin on täysin mahdollista työssäni käsittelemillä työkaluilla. Aluksi toivoin perehtyväni siihenkin kevyesti, mutta olkoon se joskus oma aiheensa.

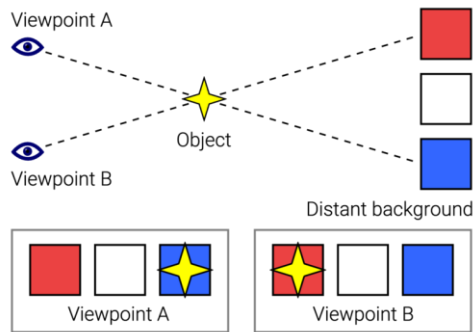
Teoriaan hyödynnän ensisijaisena lähteenä elokuvateorian kirjallisuutta ja artikkeleita. *Parallax-efektiä* hyödynnetään paljon juuri videokuvauksen keinoissa, ja on selvää, että myös digitaalisessa maailmassa pohja on lähtenyt elokuvien käytännöistä – virtuaalisen kameran liikkeistä lähtien.

2 Parallax-efekti

Tässä luvussa avaan aiheeni teorian osalta. Selvennän *parallax-efektin* käsitettä ja sen käyttötapoja lähihistoriassa. Selvitän tarkemmin myös videokuvauksessa käytettyjä keinoja ja kameran liikkeitä, jossa tekniikkaa on hyödynnetty.

2.1 Mikä on parallax

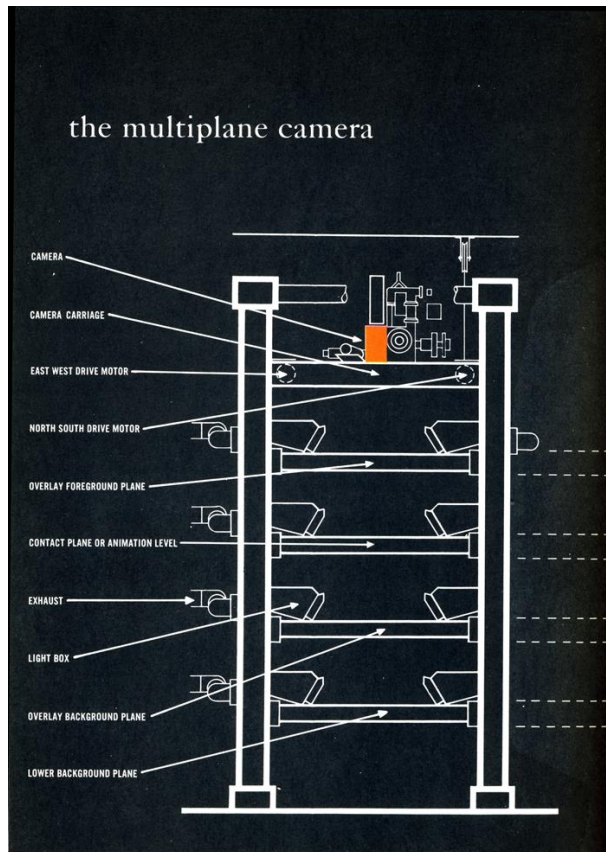
Parallaksin perimmäinen idea on, että objektien ollessa eri etäisyyksillä toisistaan luo niiden katsominen eri kuvakulmista illuusion paikallaan olevan esineen liikkeestä ja vuorovaikutuksesta taustaansa nähden. Lähempänä sijaitsevat objektit luovat isomman parallaksin kuin kauimmaisat objektit. (Wikipedia 2023a.) Kuvan 1 kaavio havainnollistaa, kuinka yksittäistä objektia tarkastellaan kahdesta eri kuvakulmasta ja kuinka efekti syntyy suhteessa kauempana oleviin taustoihin.



Kuva 1. Yksinkertaistettu kuvitus parallaksista, jossa tähtikuvio näyttää vaihtavan paikkaansa suhteessa taustalla oleviin objekteihin. (Wick 2004.)

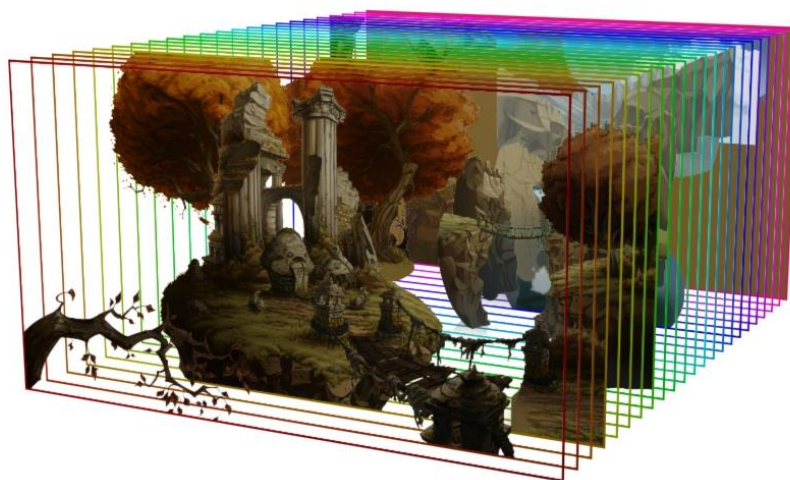
Parallax-efektin ymmärtää reaali maailmassa esimerkiksi liikkuvan auton kyydissä. Katsoessaan ikkunasta ulos voi huomata, kuinka lähimmäiset objektit, kuten katuvalot ja puut, näyttävät liikkuvan paljon nopeammin suhteessa kauempana siintäviin rakennuksiin, metsiin tai kukkuloihin. (Gharat i.a.)

Animaatiomaailmassa tekniikka on ollut käytössä jo pitkään – vuonna 1937 William Garity loi kuuluisan Walt Disneyn *Multiplane*-kameran parallax-efektin luomiseen. Hyllymäisessä laitteessa oli kerroksia seitsemällä lasilevyllä, kuten kuvan 2 piirros osoittaa. Kuvituksen eri kerrokset maalattiin näille levyille, ja jokaista levyä liikuteltiin erikseen kuva kuvalta, vaihtelevilla nopeuksilla luoden illuusion etäisyyksistä animaatioissa. Itse päällimmäisenä kohtauksen kuvaava kamera oli aina tiukasti paikallaan. (Kumar 2019.)



Kuva 2. Multiplane-kamerassa kuvituksen pystyi jakamaan useammalle lasilevyllä, jotka päällimmäinen kamera pystyi kuvaamaan. (Art of Animation 1958, 124)

Digitaalisessa mediassa tätä samaa efektiä imitoidaan tekniikalla nimeltään *parallax scrolling* eli parallax-vieritys. Sillä eri syvyyksiin asetettuja kaksiulotteisia kuvia liikutetaan virtuaalisen kameran edestä siten, että kameraa lähinnä sijaitsevat elementit liikkuvat nopeammin kuin kauempana sijaitsevat. Tämä luo vastaavanlaisen tunteen syvyydestä ja liikkeestä. Pelimaailmassa efekti tunnetaan myös ”2,5D-ulottuvuutena”. Kuvassa 3 on esimerkki Daedalic Entertainmentin vuoden 2009 pelistä *The Whispered World*, jossa pelimaailma rakennettiin vastaavalla tekniikalla (Wikipedia 2022a).



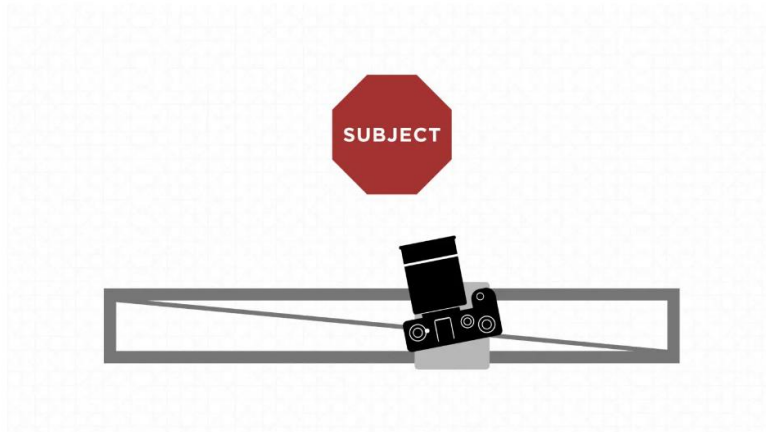
Kuva 3. 2D-peleissä hyödynnetään parallax-efektiä tehokkaasti. (Daedalic Entertainment 2009)

Käytännössä nykyaikaiset digitaaliset työkalut eivät siis näytä valtavasti erkaantuvan siitä, miten efekti toteutettiin aikaisemmin. Nykyään parallaksin tekemiseen ei tarvita enää onneksi isoja rakennelmia keskelle studiota, vaan voimme potentiaalisesti tehdä kaiken jopa mobiililaitteista käsin. Ei ole kuitenkaan liioiteltua sanoa, että *multiplane* oli selvästi suuri mullistus visuaaliselle alalle.

2.2 Kameran liike

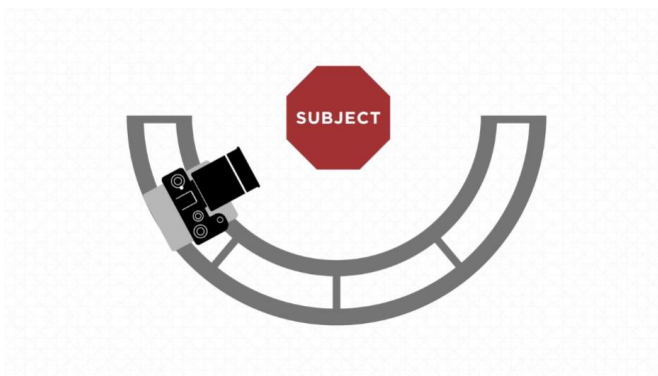
Kameran liikkeellä on oleellinen osa *parallax-efektin* luonnissa, ja tässä luvussa keskitynkkin sen toimintaan juuri elokuvamaailmassa. Samoja käytäntöjä hyödynnetään myös digitaalisesti.

Tapoja on useita. Janisch (2019) opastaa sivustonsa tutoriaaleissa näitä malleja. Ensimmäisessä metodissa kameraa ajetaan suoraa reittiä ylös tai sivuttain esimerkiksi *kameradollylla eli kamerakärryllä* tai liukukiskoja pitkin, samalla kun linssi on keskittynyt paikallaan olevaan kohteeseen (kuva 4).



Kuva 4. Kamera kuvaa kohdetta suoralla sivuttaisliikkeellä. (Fenchel & Janisch Film Production 2021)

Kamera-ajon voi Janischin (2019) mukaan toteuttaa myös siten, että kamera kiertää paikallaan olevaa kohdetta joko osittain tai kokonaan ympäri (kuva 5). Tärkeää on kuitenkin huolehtia, että liike on edelleen tasaista ja etäisyys kohteeseen pysyy aina samana, eli esimerkiksi kaarevat liukukiskot ovat oleelliset kuvaukseen.



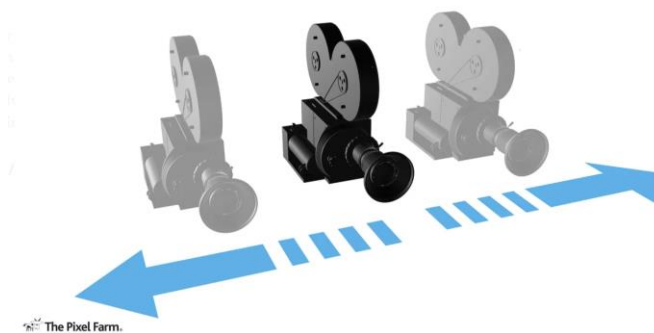
Kuva 5. Kohdetta voi myös kuvata ympäri efektin luomiseksi. (Fenchel & Janisch Film Production 2021)

Parallaksia voi luoda myös drone-kuvauksessa niin kutsutulla *Orbit*-tekniikalla. Drone-kameroiden verkkokauppa DJI-storen blogitutoriaali ohjeistaa, kuinka tekniikkaa voidaan käyttää näyttävien ilmakuvien ottamiseen (kuva 6). Käytännössä kamera kiertää kohdetta ympäri, aivan kuten kuvan 5 esimerkissä.

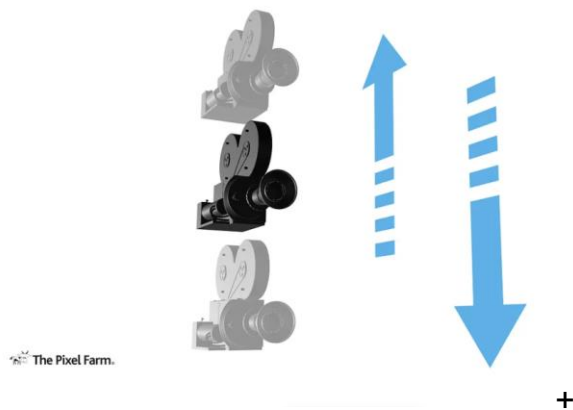


Kuva 6. Drone-kameralla voi myös luoda parallax-efektin. (DJI Store 2019)

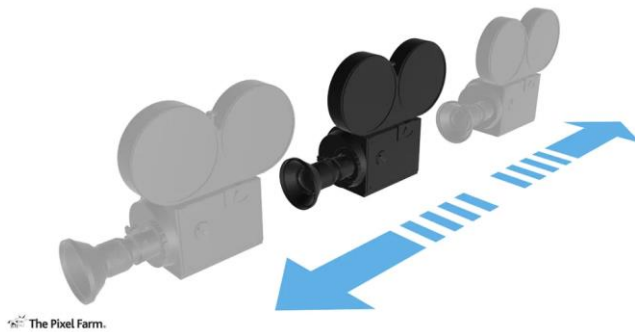
Yksi yleisimmistä tekniikoista, jota ei ehkä tule ajatelleeksi *parallax-efektiä* ajatellen, on myös kameran nosto, lasku tai ajo mihin tahansa suuntaan. Tämä tunnetaan elokuvatermeissä englanniksi *tracking shot*, *boom shot* tai *crane shot*. Siinä kamera on nimensä mukaisesti aseteltu esimerkiksi "nostokurjen" eli kraanan tai muun liikkuvan jalustan päälle, jossa kameraa sitten ajetaan kuvakulman muuten pysyessä paikallaan. Hawkes (2019) esittelee blogissaan kuvilla 7, 8 ja 9 nämä tekniikat mainiosti. Jokainen näistä on mainio suuriin kokokuvaan, yleiskuviin tai panoraamaotoksiin. Sivuttaisliike on yleensä tehokkain keino *parallax-efektiä* ajatellen. Sen avulla voidaan myös seurata esimerkiksi kävelevää kohdetta sivusta tai yläpuolelta, elokuvan miljöötä samalla esitellen. (Deguzman 2020.)



Kuva 7. Sivuttainen kameran ajo. (Medium 2019)



Kuva 8. Kameran nosto ja lasku. (Medium 2019)

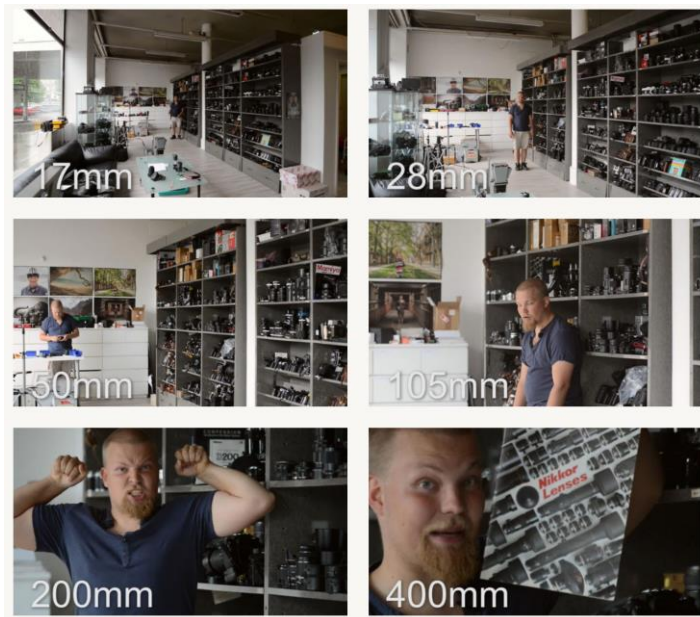


Kuva 9. Kameran ajo eteen- tai taaksepäin. (Medium 2019)

2.3 Polttoväli

Polttoväli on oleellinen valokuvauksessa. Se on kameraobjektiivin tärkein ominaisuus ja kertoo, miten laaja tai kapea objektiivin kuvakulma on. Mitä pidempi polttoväli on, sitä kapeampi on objektiivin tuottama kuvakulma. (Kamerakoulu i.a.)

Polttoväli on tärkeä huomioida myös *parallax-efektin* luomisessa, katsotaanhan kuvaa eräänlaisen virtuaalisen kameran lävitse. Mikäli kohteena on isompi maisema, voi pidempi polttoväli toimia paremmin kaukaisuuden tunnelman tuomiseksi. Lähikuvassa taas isompi polttoväli tuo intiimimmän tunnelman (kuva 10.)



Kuva 10. Polttovälillä on väliä. (Kamerakoulu i.a.)

Myös Trant (2020) mainitsee blogissaan, että esimerkiksi 50 mm:n polttoväli sopii hyvin ”katukuvaukseen, kokovartalomuotokuvaan ja arjen kuvaamiseen”. Todennäköisesti tämä on omiin tarpeisiin siis hyvä etäisyys.

2.4 Point of interest

Parallax-efektissä on oleellista myös *point of interest* eli kohdepiste, johon kuvassa keskitytään. Myös edellisen luvun kuvissa 4–6 kamera käytännössä kiertää valittua kohdepistettä. Kun itse kohde pysyy kuvassa paikallaan suhteessa elementteihin kameran liikussa, tapahtuu dynaaminen liike kohteen ja taustan välillä luoden voimakkaan kokemuksen tilasta. (Dji 2019.)

Digitaalisessa maailmassa *kohdepiste* on tärkeässä roolissa myös esimerkiksi After Effectsiä käytettäessä. Jos haluaa tehdä kolmiulotteista animaatiota, jossa kamera seuraa kohdetta tai pyörii sen ympärillä, tarvitaan käyttöön *two-node camera*, joka pystyy kiinnittämään itsensä kohdepisteeseen. (Thompson 2019.) Tätä luonnollisesti hyödynnetään myös parallax-kuvituksen luomisessa. Kun kamera seuraa kuvituksessa kohdepistettä, saadaan liikkeessä aikaiseksi dynaaminen liike ja syvyyssuhde kuvituksen eri elementtien välillä.

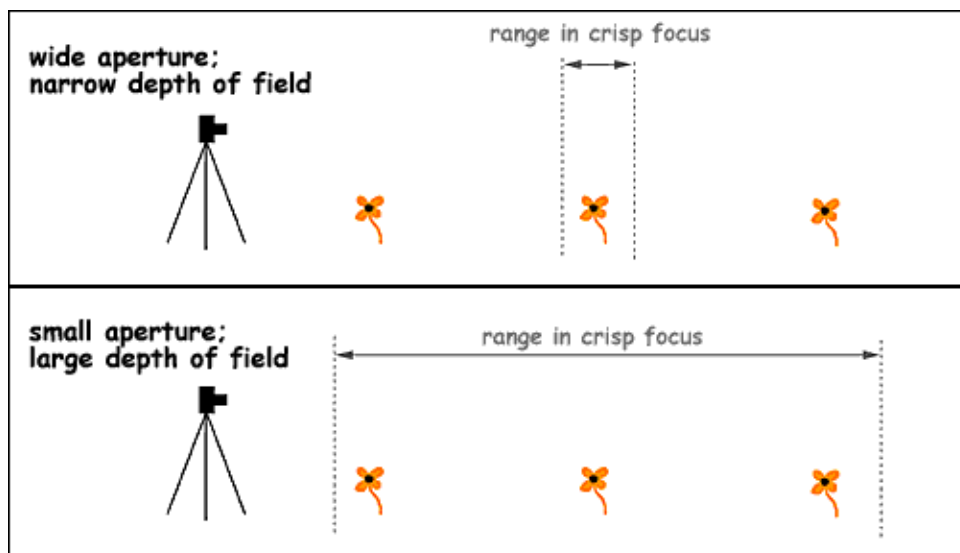
2.5 Syväterävyys

Syväterävyys eli *Depth of Field* tarkoittaa yksinkertaisimmillaan sitä, kuinka suuri valokuvan yhtenäinen terävä alue on syvyyssuunnassa katsottuna. Syväterävyyttä voidaan käyttää visuaalisena tehokeinona – esimerkiksi lyhyellä terävyyalueella kohde voidaan irrottaa ympäristöstään tarkempaan kuin muu tausta, tai sitten suurella alueella lähes koko kuva näkyy erityisen terävänä. Toisaalta esimerkiksi kameran etualan kohteet taas voivat olla sumeampia kuin kaukaisemmat kohteet. (Vaihtolammi 2021.)

Käytännössä kameran avulla syväterävyyttä säädetään polttoväleillä ja kameran aukoilla. Lyhyellä syväterävyydellä polttoväli on pidempi, kohde lähellä ja aukkoluku on isompi. Toisaalta pitkällä syväterävyyalueella käytetään lyhyttä polttoväliä ja pientä aukkoa ja kameran etäisyys kohteesta on kaukaisempi. (kuva 11). Yleisesti kuvan syväterävyys myös jakautuu niin, että kolmannes syväterävyyalueesta on tarkennuspisteen edessä ja kaksi kolmasosaa takana (Kamerakoulu i.a.). Tämä on hyvä huomioida kuvaa sommitellessa.

Esimerkiksi After Effectsiin virtuaalikameroilla syväterävyyttä voi hyvin testata *parallax-efektin* luomiseen. Efektin voi luoda kuvitukseen myöhemmin myös manuaalisesti sumentamalla eri kerroksia esimerkiksi kuvankäsittelyohjelmalla. Tämä voisi mahdollistaa vaikkapa keinotekoisien lyhyen terävyyalueen, jossa tarkasteltava yksittäinen kuvituksen elementti erottuu ympäristöstään terävämpanä kuin esimerkiksi kuvituksen sumennettu tausta tai kameran edessä sijaitsevat sumennetut elementit.

Tekniikkaa voisi parallax-kuvituksessa käyttää niin muotokuvamaalauksessa kuin pienten kohteiden yksityiskohtaisessa esittämisessäkin. Lyhyttä syväterävyyttä voi oivasti hyödyntää myös elokuvallisen ja unellisen tunnelman luomiseen, tai visuaalisesti enemmän henkilökohtaisen tarinan kertomiseen. Mikäli kuvituksessa on mukana yksinkertaista animaatiota, voisi alati vaihtuvaa syväterävyyttä käyttää korostamaan kuvassa tapahtuvaa toimintaa tai dynamiikkaa. (In Depth Cine 2022.)



Kuva 11. Kameran aukon koolla voi jo vaikuttaa syväterävyyteen. (Jyväskylän yliopiston informaatioteknologian tiedekunta 2006)

3 Parallax-efektin käyttötavat selaimessa

Vaikka nykyään *parallax-efektiä* on mahdollista luoda erilaisilla ohjelmilla ja ap- plikaatioilla, lähtevät sen juuret syvältä verkkosivukoodauksesta. *Hyper Text Markup Language*- (HTML), JavaScript- ja *Cascade Style Sheets* (CSS) -koodi- kielten kehittyminen mahdollistavat jatkuvasti uusia tapoja uusille tekniikoille ja mahdollisuuksille. Erityisesti HTML 5, CSS3 ja JQuery aloittivat ensitaipaleen kehitykselle, joka on vain jatkunut entisestään (Wang & Shyu 2014, 235–237). Näiden käyttöä käsitellään tarkemmin myöhemmässä luvussa.

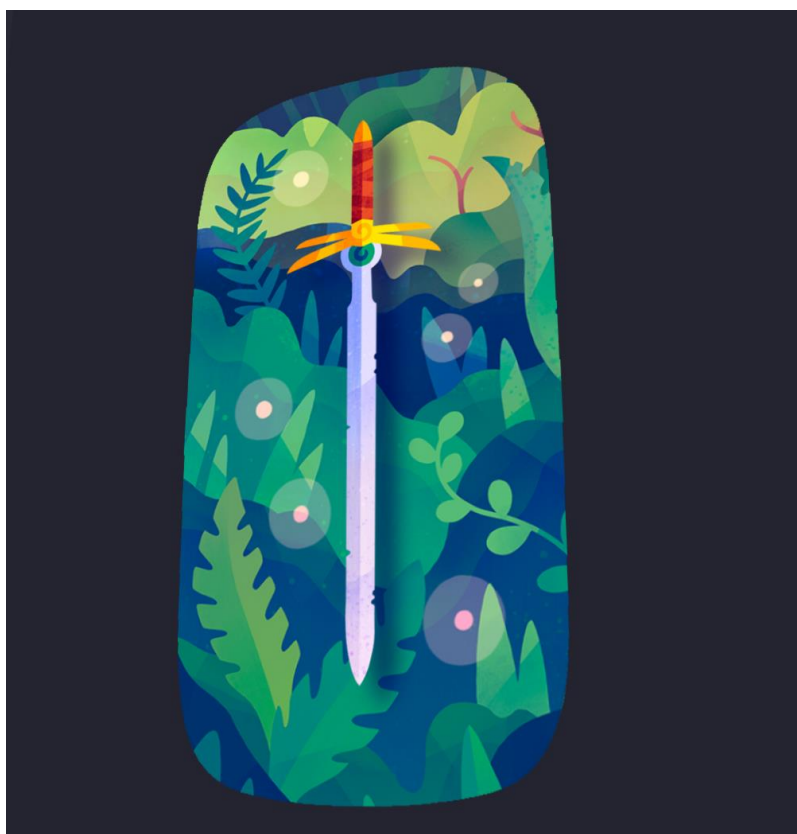
Käyttökohteita on monipuolisesti, ja esimerkiksi *scroll parallaxin* yhteydessä voi- daan luoda mitä innovatiivisempia visuaalisia esityksiä. Seuraavissa alaluvuissa käyn erilaisia esimerkkejä läpi.

3.1 Tilt parallax

Tilt parallax -efektissä elementit pysyvät paikallaan eri syvyyksissä, mutta ka- meraa voi virtuaalisesti kallistella eri kuvakulmiin samalla, kun kamera seuraa elementin kohdepistettä. Kallistusta voi tehdä hiirtä liikuttamalla, sormella vetä-

mällä tai vaikkapa puhelinta liikuttamalla (Scerbinina 2019). Termille ei ole vaikiintunutta suomennosta, joten viittaan efektiin tekstissäni sen englanninkielisellä termillä.

Tilt parallax on mielenkiintoinen esimerkiksi portfoliosivustolla töiden esittelemiseen, jolloin itse esiteltävä kohde liikkuu kevyesti hiiren liikkeillä keskellä sivustoa. Esimerkkinä voisin käyttää innoittajaani Jarom Vogelia, joka julkaisee monia *kuvituksiaan tilt parallaxin* avulla (kuva 12).



Kuva 12. Jarom Vogelin kuvituksia voi käännellä kulmista tilt parallaxilla. (Jarom Vogel 2019)

Toinen esimerkki *tilt parallaxista* on kevyempi sivuttaisliike, joka tapahtuu esimerkiksi hiirtä liikuttamalla tai puhelinta kallistamalla. Tähän ei tarvita käyttäjältä suoraa interaktiota sivuston kanssa, vaan liike tapahtuu luonnostaan. Raoul-Gaillardin portfoliosivustolla esimerkiksi kuvan 13 mies ja taustan palloelementit kallistuvat ja liikahtavat kevyesti hiiren liikkeisiin reagoiden.



Kuva 13. Tilt parallaxilla sivuston kuvat ja tekstit kääntyvät kevyesti eri kulmiin hiiren liikkeen mukana. (Raoul Gaillard 2016)

Tilt parallaxia voidaan käyttää myös 3D-mallinnusten kanssa, esimerkiksi tuotteiden esittelyyn. *WebGL:ään* erikoistuva verkkopohjainen applikaatio *Vectary* esittelee verkkosivuillaan mainioita esimerkkejä tähän liittyen. (kuva 12.) 3D-malli kääntyilee kevyesti hiirtä liikuttamalla. Vierittämällä kameraa voi zoomata kauemmas tai lähemmäs kohteeseen. Hiiren painiketta alas painamalla mallia voi myös vapaasti pyöritellä ja tarkastella mistä suunnasta tahansa, ja mallinnukseen voi upottaa myös erillisiä ”painikkeita” kohtiin, joita tuotteesta halutaan erityisesti esitellä. Tämä nappi kääntää kameran suoraan suotuisaan kohtaan yksityiskohdan esittämiseksi.



Kuva 12. 3D-mallinnusta voi pyöritellä ja tarkastella suoraan verkosta käsin.
(Vectary 2023)

3.2 Scroll parallax

Scroll parallax on varsin yleinen verkkosivuilla käytetty efekti. Siinä perinteisesti sivustoa ”rullataan” yhteen suuntaan, yleensä ylös tai alas, ja sivustolla olevat eri kerroksissa sijaitsevat taustaelementit näyttävät liikkuvan rullauksen mukana. Käytännössä kameraa siis ”ajetaan” alaspäin, luvun 2.2 kameran noston ja laskun periaatteiden mukaisesti. Kameran liike tuo visuaalisen efektin, jonka takia edessä sijaitsevat elementit näyttävät liikkuvan nopeammin ja kauempana sijaitsevat taas hitaammin, joten syvyyden illuusio syntyy. Tällekin termille ei ole vakiintunutta suomennosta, joten kerron efektistä tekstissäni vain sen englanninkielisellä nimellä.

Yksi mainio esimerkki *scroll parallaxista* on ukrainalaisen Candy Fish -nimisen mainostoimiston verkkosivusto. Värikkäät kuvitukset liikkuvat syvyytensä alas rullatessa, ja edessä olevat elementit liikkuvat nopeammin suhteessa kauempana siintävään maisemaan. (kuva 13).

Tämän lisäksi kuvituksissa on samanaikaisesti myös horisontaalisesti liikkuvia elementtejä, jotka luovat visuaaliselle ilmeelle näyttävää tarinankerrontaa.



Kuva 13. Sivuston kuvituksissa on alaspäin rullatessa myös sivuttain liikkuvia elementtejä, tässä tapauksessa oikealla näkyvä taksi. (Candy Fish 2022)

Kunniamaininnan mielestäni ansaitsee myös The New York Timesin artikkeli *Tomato Can Blues*, joka dokumentoi häkeissä tapahtunutta amatöörikamppailua ennen vapaaottelun kehittymistä. Kuvitukset ilmestyvät tekstin välissä ajoittain jopa toistensa päälle kuin sarjakuvapaneelit, ja taustoista irrotetut hahmot puhekuplineen liikkuvat kuvissa lähemmäs, ylhäältä alas tai vasemmalta oikealle lukijan rullatessa artikkelia alaspäin (Kuva 14). Tämä on mielestäni hurmaava tapa saada lukija sisälle haastateltavien tarinaan.

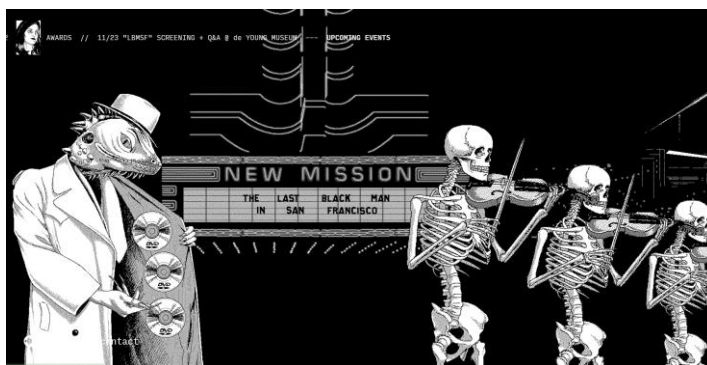


Even Rowan was shaken. "There was a lot of blood," he said. "Enough to water me. I'm a man used to seeing a lot of blood, but that was a lot of blood."

Rowan licked his girlfriend in the arm, hoping to make her seem like a second victim. He shoved eight handguns into his red and black duffel bag and then, on his way out, noticed Rabbitte's wallet sticking out of his pocket. He grabbed that, too, and tore off through the tunnel.

Kuva 14. Ilmestyvät sarjakuvapaneelit ja liikkuvat hahmot sopivat tekstin yhteyteen erinomaisesti. (New York Times 2013)

Mielenkiintoinen vaihtoehto on myös animoida liike kulkemaan sivuttaissuunnassa vasemmalle ja oikealle (Kuva 15). Tuotantostudio The Longshot Film käyttää horisontaalista *scroll parallaxia* sivuston navigoimiseen. Kohtaus rullautuu oikealle esitellen näyttävästi animoituja kuvituksia studion eri projekteista. Kuvituksia klikkaamalla pääsee katsomaan portfolioa tarkemmin. Kuvitukset ovat kerrostettuna toisistaan ja aiheuttavat liikkeessä myös *parallax-efektiä* – edessä olevat elementit liikkuvat nopeammin suhteessa niiden takana oleviin. Sivustolla on uniikki tarinankerronnan elementti.



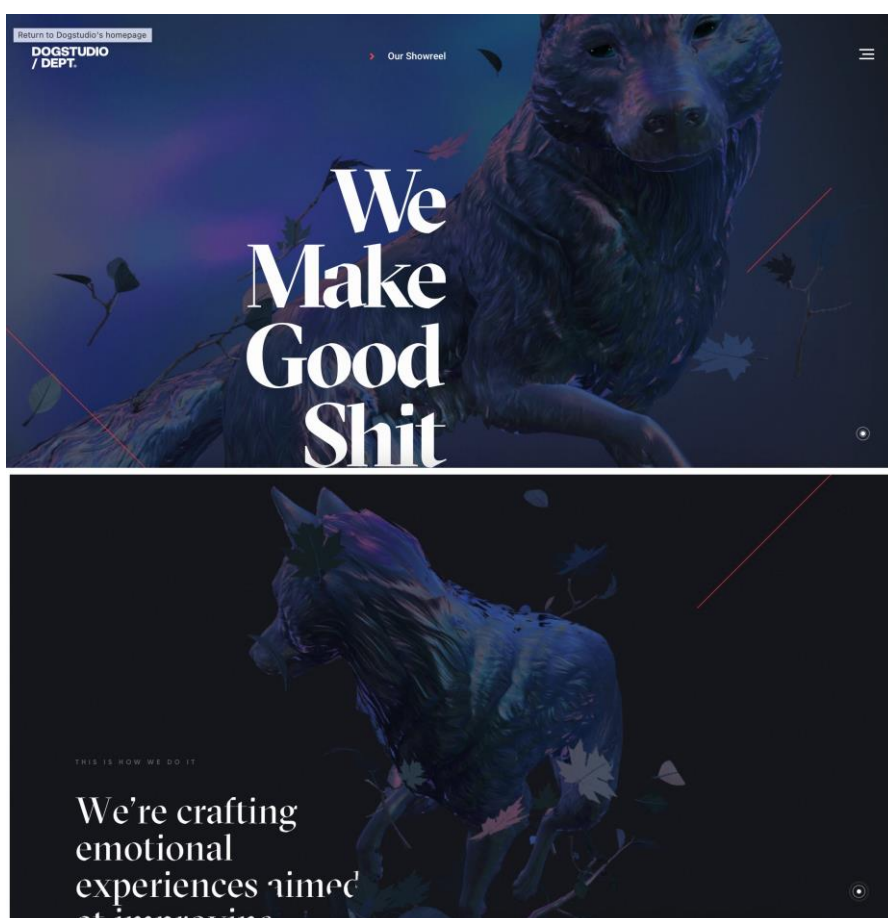
Kuva 15. Horisontaalisesti tapahtuva parallax-efekti luo näyttävää tarinankerrontaa. (longshotfeatures.com, 2022)

Kolmanneksi *scroll parallaxia* voi käyttää myös syvyys suunnassa eteenpäin liikkumiseen. Mainio esimerkki tästä on *The Goonies* -elokuvalle omistettu verkkosivusto, joka heti alussa rullatessa zoomautuu sisälle merimaisemaan, samalla kun ympärillä näkyvät puunlehdet liikkuvat pois tieltä. Tämä vastaa illuusiota eteenpäin tapahtuvasta kamera-ajosta (Kuva 16).



Kuva 16. Parallax-efektiä on mahdollista tehdä myös syvyys suunnassa. (The Goonies fansite 2022)

Scroll parallaxissa voidaan myös hyödyntää kolmiulotteisia malleja sekä hitaasti ilmestyvää tai häviävää tekstisisältöä (kuva 17). Dogstudion verkkosivusto on mielestäni erityisen näyttävä esimerkki. Kuvaa alas vierittäessä animoitu susimalli pyörii hitaasti ympäri ja siirtyy kauemmas kamerasta. Sivuston tekstielementit vierivät välillä myös mallin takana, haittaamatta silti tekstin näkyvyyttä. Verkkosivun hampurilaisvalikossa taas hyödynnetään *tilt parallaxia*, jossa susi kääntyy kevyesti eri kuvakulmiin hiirtä liikauttamalla. Portfoliossa uudet latautuvat kuvat ilmestyvät kevyesti animoituna näkyviin alas vierittäessä.



Kuva 17. Scroll parallaxissa voi käyttää myös pyöriviä elementtejä. (Dogstudio 2023)

Sivusto saattaa olla ehkä jopa ääriesimerkki visuaalisesti näyttävästä *parallax-efektistä*, joka ei kuitenkaan mene selkeyden tai käytettävyyden edelle. Visuaalisuuteen erikoistuneelle studiolle ulkoasu toimii mielestäni mainiosti sekä mahdollisesti vakuuttaa asiakkaan yrityksen ammattitaidosta luoda innovatiivisia ratkaisuja. Kuvittelisin kuitenkin, että sivuston optimointi erinäisillä laitteilla on

haastavaa, ja elementtejä olisi hyvä käyttää huolellisella harkinnalla. Osalle käyttäjistä liika visuaalisuus voi olla myös hämmentävää, mikäli ei ole siihen totunut.

4 Tekninen toteutus ja eri mahdollisuudet

Tapoja toteuttaa *parallax-efektiä* on nykyään useita, perinteisestä koodauksesta valmiisiin ohjelmistoihin, jotka osaavat visuaalisesti luoda kuvan ja laskea syvyyssparametrit. Ohessa syvennyn muutamaa esimerkkiin ja sivuan osaa kevyemmin.

4.1 Julkaisutavat

Perinteisin julkaisutapa verkossa lienee tekijän oma verkkosivusto. Oman tietämyksen ja taitojen puitteissa on mahdollista koodata sellainen alusta alkaen itse. Tämä myös tuo vapauksia luoda täysin omanlaista sisältöä niin toiminnallisuuden kuin visuaalisen ulkoasun suhteen. Kuitenkin verkkosivun voi myös rakentaa esimerkiksi WordPressin tai muun julkaisujärjestelmän avulla. Materiaali ”upotetaan” sivustolle koodin avulla tai suoraan järjestelmän editorin kautta minimaalisella kooditietämyksellä, mikäli palvelussa se on mahdollista. (Wikipedia 2022b.) Järjestelmiä on paljon erilaisia, ja sisältöä voi luoda jokaiselle ominaisuuksien ja rajoitteiden puitteissa. Vaihtoehtoja myös *parallax-efektin* luontiin on täten useita.

Kuitenkin myös sosiaalisen median kehittyminen ja esimerkiksi Facebookin nykyinen ominaisuus lähettää palveluun omia kolmiulotteisia kuvia, ja jopa 3D-malleja, tuo uusia mahdollisuuksia julkaisukanaviin. Optimointi toistaiseksi jää vielä tekijän omalle kontrolle, mutta töiden julkaisu on monella tapaa enää parin klikkauksen päässä.

Myös erinäiset applikaatiot verkossa ja mobiilialustoilla sisältävät omat julkaisukanavansa, joihin oman työnsä voi lisätä, näistä mainintoina vaikkapa Mental Canvas sekä Rive, joihin perehdyn seuraavassa luvussa. Monesti näihin liittyy

myös kyseisen applikaation oma erityinen tekniikka *parallax*-kuvituksen tai animaation luomiseen, minkä takia sen katselemiseen tarvitaan myös täysin oma alustansa.

4.2 Valmiit alustat

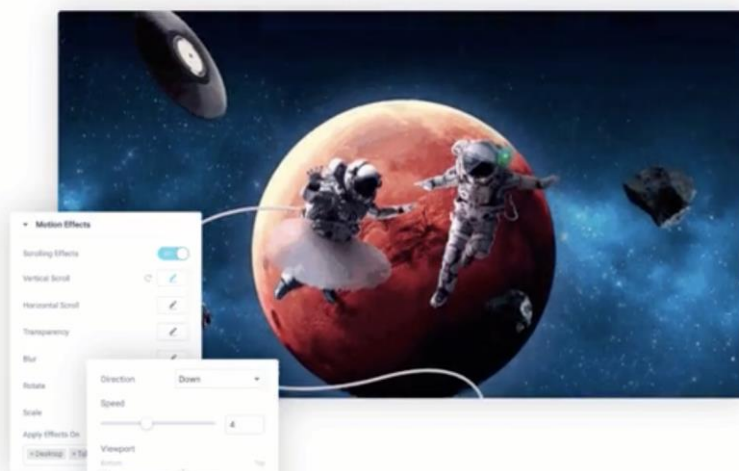
Mental Canvas on iOS- ja Windows-järjestelmille saatava rasteriin perustuva piirtämisohjelma. Työskentely on käytännössä samanlaista kuin perinteisissä ohjelmissa, joissa pohjaan on mahdollista luoda erillisiä *layereita eli* kerroksia. Eroavaisuus alkaakin siitä, että kerroksia on mahdollista upottaa eri syvyyksiin mahdollistaen täten kolmiulotteisen kokonaisuuden. Näkymää voi zoomata syvyyssuunnassa ja lisätä kauempana oleviin kerroksiin lisää yksityiskohtia. Alusta mainostaa, että applikaatio on ”digitaalisen ajan piirtämistä uudelleen kuviteltuna”.

Maksullisella versiolla kuvituksensa voi julkaista myös ohjelman omaan suljettuun verkkogalleriaan. Galleriassa käyttäjä voi käänellä kuvituksen katselukulmaa hiiren tai sormen avulla sekä rullata näkymää syvemmälle kuvan sisään. Tämän systeemin ulkopuolelle kuvitusta ei näyttäisi olevan mahdollista julkaista sellaisenaan. Kuvituksen interaktiivisuutta ei kuitenkaan näyttäisi pystyvän rajaamaan tiettyihin arvoihin, kuten kuvasta 18 näkyy. Sitä on mahdollista myös zoomata liian kauas tai sisään rajojensa yli alueille, mihin kuvan tekijä ei mahdollisesti ole tarkoittanut katsojan pääsevän. Voi olla, että kyseessä on bugi, joka korjataan myöhemmissä päivityksissä.



Kuva 18. Teoksia voi käänellä Mental Canvasissa myös parametrien ulkopuolelle. (Frostbite Studio 2022)

Kotisivukoneet kuten Elementor tarjoavat myös helpon tavan luoda Parallax-efektejä ilman koodaustarvetta. Elementor Pro -versiossa käyttäjä voi valita laajasta määrästä erilaisia vaihtoehtoja – eri *parallax-efektit* on yksinkertaisesti nimetty valikossa termillä *motion effect*. Valittavana on joko vertikaalinen tai horisontaalinen vieritys, läpinäkyvyyden säätö, sumennus, rotaatio ja skaalaus. Näiden nopeuksia ja muita ominaisuuksia voi liukuvalikosta säätää vieläkin tarkemmin (kuva 19).



Kuva 19. Elementor tarjoaa kattavasti vaihtoehtoja parallax-efektin luomiseen ilman kooditietämystä. (Elementor 2020)

Meta on luonut myös Facebookiin mahdollisuuden käyttäjälle luoda omiin valokuviinsa *parallax-efektin* automaattisen algoritmin avulla. Systeemi käyttää hyväkseen *depth map* -tekniikkaa eli syvyyskarttaa, jonka avulla se pystyy laskemaan elementtien etäisyydet toisistaan. Tekniikan tulisi toimia liki automaattisesti puhelimissa, joissa on käytössä useampi kamera, mutta syvyyskartan voi luoda myös itse omista valokuvistaan ja täten luoda efektin manuaalisesti. (Tabora 2019.) Syvennyn tähänkin ominaisuuteen myöhemmässä projektiosassa.

Metan *g/TF*-formaatti luo myös oman mahdollisuutensa. Tämän avulla voidaan Facebookiin siirtää omia 3D-mallejaan pyöriteltäväksi ja katseltavaksi. (Gönen 2018.) Ominaisuus on mielenkiintoinen, mutta jätän tämän tekniikan tarkastelun ulkopuolelle tekstissäni, vaikka seuraankin sen kehittymistä mielenkiinnolla.

4.3 CSS ja JavaScript

CSS eli *Cascade Style Sheets* on koodikieli, jonka tärkein tehtävä on muokata HTML-sivuston ulkoasua (w3schools.com i.a. a). *Parallax-efekti* onnistuu pohjimmiltaan pelkästään myös CSS:ää käyttämällä, mutta mahdollisuudet ovat toki rajallisemmat kuin *JavaScriptiä* hyödyntäessä. Tekniikoita on useita ja osa monimutkaisempiakin, mutta W3 Schools (i.a. b) opettaa perusteet. Tärkein ominaisuus on mahdollisuus kiinnittää tausta paikoilleen esimerkiksi *background-attachment:fixed* -komennolla. Tämän lisäksi taustan kokoa, minimikorkeutta ja -leveyttä, sekä sijoittelua voi vielä muokata. Tällä luodaan *parallax-efektin* yksinkertaisin muoto. Muut elementit sivustolla liikkuvat rullauksella alaspäin taustan pysyessä paikallaan, luoden efektin takana olevasta taustaelementistä. (w3schools i.a. b.)

JavaScript on suosittu, pääasiassa verkkoympäristössä käytettävä ohjelmointikieli (Wikipedia 2022c). Vaikka koodia voi luoda myös alusta asti käsin, on monelle ominaisuudelle olemassa myös valmiita JavaScript-kirjastoja, jotka voi vain liittää osaksi omaa html-sivustoaan. *Parallax-efektin* tapauksessa tällaisia on muun muassa *vanilla-tilt.js*, *pureParallax.js* ja *miniparallax.js*.

Jokainen näistä kirjastoista tuo oman tapansa lisätä *parallax-efektin* koodiin mukaan, osa yksinkertaisemmin kuin toiset. Esimerkiksi *tilt-js* luo *tilt parallaxin*, jonka liike toimii automaattisesti hiiren liikkeessa ruudulla. Tämän vahvuustasoa voi koodissa muuttaa käsin mieluiseseen.

Omissa kokeiluissani aion syventyä vain tähän kyseiseen kirjastoon, sillä se sopii kuvitustarpeisiini parhaiten.

5 Oma kuvitusprojektini

Tässä luvussa aloitan opinnäytetyöni projektiosuuden, joka sisältää aluksi paljon kokeiluja ennen varsinaisen työn aloitusta. Dokumentoin kaikki vaiheet suunnittelusta lopulliseen projektiin tulevissa alaluvuissa, edellisen luvun teoriaa samalla soveltaen.

5.1 Suunnitteluprosessi

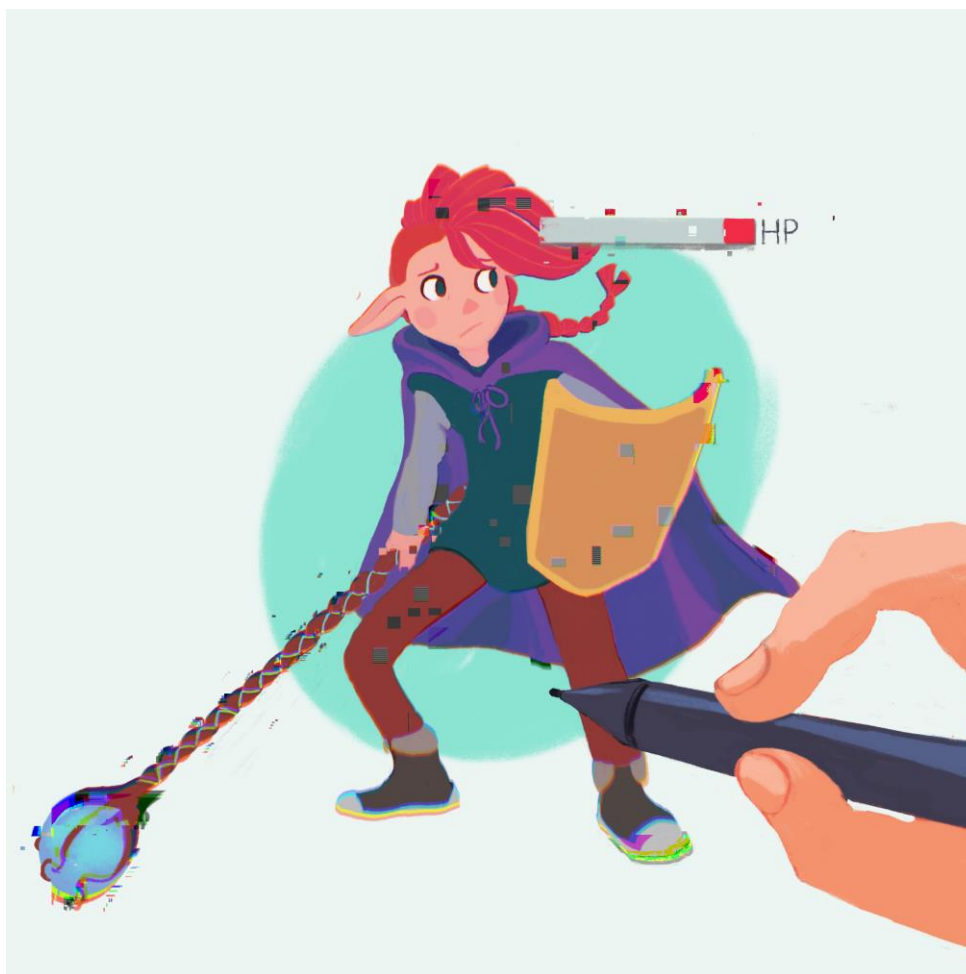
Lopputavoitteeni projektissani on tehdä *parallax-efektillä* toimiva kuvitus, joka pitäisi sisällään myös yksinkertaista animaatiota. Kuvitus toimisi selaimessa ja olisi upotettavissa esimerkiksi omalle verkkosivustolleni portfolioon. Lähdin työstämään ideaa usealla nopealla raakaluonnoksella ja muistiinpanolla, että saisin konseptin kasaan.

Lopputulemana oli idea neliönmuotoisesta kuvituksesta, jota voisi potentiaalisesti hyödyntää myöhemmin myös sosiaalisessa mediassa. Varsinkin Instagramissa yleistynyt kuvasuhde on usein 1:1, ja se tuntui luontevalta tähän projektiin (Myllyveräjä 2022). Päädyin myös keskittymään pääasiassa *tilt parallaxin* luontiin omassa kuvituksessani.

Aloitan prosessin ensin vanhemmalla kuvituksellani, jolla lähdän kokeilemaan erilaisia tekniikoita *parallax-efektin* toteutukseen. Valitsen näistä sitten yhden, jota käytän lopullisessa työssäni.

5.2 Testikuvitus

Testikuvitukseni on yksinkertainen *pikseleillä* ja *väriaberraatiolla* vilkkuva hahmo, jonka tein julkaisukurssin artikkelikirjoitustani varten (kuva 20). Kokeilu on sinänsä myös realistinen, sillä työelämässä verkkoartikkelikin voisi sisältää mielenkiintoisen *parallax-efektillä* toimivan kuvituksen. Voisipa sitä käyttää myös artikkelin markkinointiin sosiaalisessa mediassa, kunhan kuvasuhde on kohdillaan.



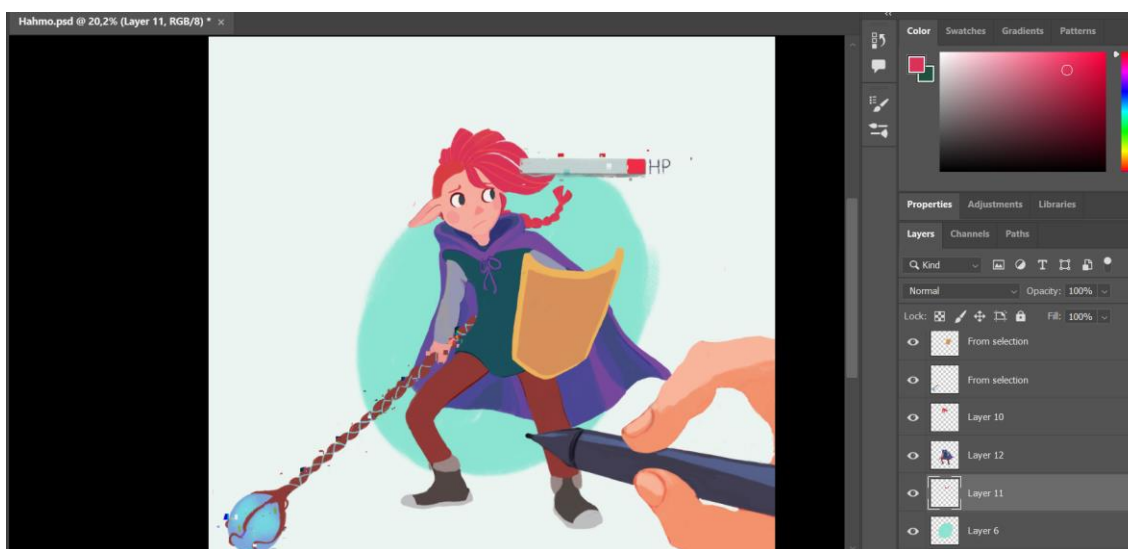
Kuva 20. Kuvitukseni hahmo sisältää pikseliefektiä ja väriaberraatiota, jotka animoin ilmestymään satunnaisesti. (Tiina Hautamäki 2023)

Aluksi avasin kuvituksen Photoshop-tiedoston ja erottelin kuvituksen elementtejä huolellisesti omiksi kerroksikseen eli *layereikseen*, kuten näkyy kuvassa 21. Termille *layer* ei ole toistaiseksi vakiintunutta suomennosta. Vaikka aiemmassa

luvussa suomensin tämän usein *kerroksena*, käytän projektiosassani jatkossa vain englanninkielistä termiä.

Päällimmäisenä on käsi joka pitelee kynää, sekä hahmon edessä leijuva energiapalkki. Hahmo sijaitsee kauempana kamerasta, mutta esimerkiksi pää ja kepin etuosa voisivat kevyesti tulla eteenpäin. Hahmon vasemmassa kädessä oleva kilpi tulee myös reilummin ulos etualalle. Pään takana sijaitseva hiusletti voisi olla aavistuksen syvemmällä hahmon takana, ja taustalla toimiva palloelementti on vielä reilummin muita elementtejä syvemmällä.

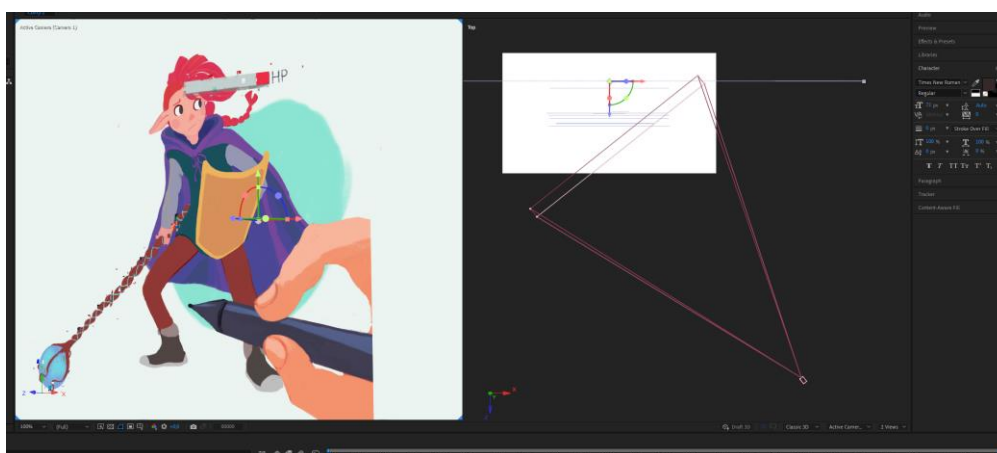
Maalasin uudelleen hahmon vartalosta huppua sekä toista käsivartta, sillä kuvaa käännellessä nämä osat tulevat varmasti näkyviin eri perspektiivissä. Maalaa myös hiusten lettiosaa pidemmälle pään taakse samasta syystä. Näin kuvitus näyttää luonnollisemmalta, eikä siihen ilmesty selkeästi havaittavia aukkoja.



Kuva 21. Kuvitusten eri syvyyksiin tulevat elementit tulee eritellä omiksi kerroksiksi. (Tiina Hautamäki 2023)

Tein seuraavaksi After Effectsissä testikomposition, jolla pääsin kokeilemaan, miltä käännettävä kuvitus näyttäisi käytännössä. Loin uuteen kompositioon *two node* -kameran eli kameran, jonka pystyy keskittämään kohdepisteeseen *parallax-efektin* luomiseksi. Käytän kamerassa 50 mm:n polttovälillä, kuten aiem-

massa luvussa olin suunnitellut. Lisäsin tämän jälkeen kuvituksen psd-tiedoston kerros kerrokselta mukaan kompositioon. *Two node* -kameran *point of interest* eli kohdepiste, johon kamera keskittyy, on tässä itse hahmon vartalo. Asettelin *layerit* eri etäisyyksiin toisistaan siten, että taustaelementit olivat kauimmaisena kamerasta, mutta käsi sekä HP-palkki ovat ”nollatasossa” eli lähimpänä kameraa. Hahmon hiukset ja kasvot ovat vartalon molemmin puolen hieman toisistaan erillään, mutta kuitenkin tarpeeksi lähellä toisiaan, jotta kuva ei näytä irralliselta. Kuvassa 22 näkyy lopputulos kameran näkökulmasta sekä kompositiion näkymä yläpuolelta.



Kuva 22. After Effectsissä voi työtään katsoa useammasta eri kuvakulmasta kerralla. (Tiina Hautamäki 2023)

Muutaman syvyyssäädön jälkeen kuvitus alkoi näyttää tässä vaiheessa jo ihan hyvältä. Joudun maalailemaan hahmon kaulaa ja hiuksia vähän pidemmälle Photoshopissa, sillä niiden rajat tulivat edelleen liian helposti näkyviin kameraa käännellessä.

5.3 Koodaus verkkosivuille

Käytän tähän vaiheeseen Jarom Vogelien videotutoriaalisarjaa *Art & Code: Create and Code Interactive Parallax Illustration* (i.a.). Tutoriaali on katsottavissa Skillshare.com -nimisen sivuston kautta jäsenyydellä.

Vogel käyttää tutoriaalissaan alusta loppuun asti iPadia. Itse siirryn tässä vaiheessa Photoshopin apuun kuvan editoinnissa ja Visual Studio Coden puoleen koodaamisessa. Tutoriaali on melko ystävällinen sellaisellekin, joka on täysin uusi koodausprosessin kanssa. Koen kuitenkin, että perusteiden tietäminen auttaa vaiheiden ymmärtämisessä jo paljon.

Ensiksi Vogel (i.a.) kertoo luoneensa jokaisesta kuvituksen *layerista* erilliset *png-tiedostot*. *Png* on tässä oleellinen tiedostomuoto juuri siksi, että se tukee läpinäkyvyyttä. Tiedostokoko kannattaa myös muuttaa pienemmäksi, kuin mitä se on Photoshop-versiossa, sillä kuvan ei tulisi olla kovin raskas selaimelle avattavaksi. Muutan kuvitukseni 1200 x 1200-pikselikokoon, ja tallennan kaikki hahmon seitsemän *layeria* erillisiksi kuviksi.

Seuraavaksi luodaan HTML-, CSS- ja JavaScript -tiedostot. Tallennan kaikki omaan yhteiseen *Parallax* -kansioon. Kansion sisällä on myös erillinen kuvat-kansio, jonne siirrän *png-tiedostoni*. Nimeän tiedostot sen mukaan, mitä kuvassa on, että muistan niiden nimet helposti Nämä ovat siis *pallo, vartalo, hiukset, pää, keppi, kilpi ja käsi*.

En ala tekstissäni käymään kovin tarkasti läpi itse koodia, sillä teksti muuttuisi kohtuuttoman vaikeaksi myös lukijalle. Pyrin selittämään kuitenkin, miten prosessi rakentuu pääpiirteittäin. Aluksi luodaan itse verkkosivu *index.html*, johon linkitetään *styles.css*- sekä *main.js* -tiedostot. *CSS:n* puolella luodaan valkoinen, keskitetty tausta ja *display* eli esitystyyppi on *flexbox*. Tämä helpottaa sivuston *responsiivisuuden* luonnissa, jolloin ulkoasu pyrkii asettumaan erilaisilla näytöillä automaattisesti sopivimpaan kokoon.

Vogel (2020) hyödyntää tässä kohden myös HTML Canvas -elementtiä, joka mahdollistaa grafiikan piirtämisen verkkoselaimessa. Piirtäminen suoritetaan *JavaScriptin* komentojen avulla. (w3 Schools i.a. c) Vaikkakin Canvas toimii HTML5-pohjaisesti, se tukee myös *WebGL:ää*, mikä taas mahdollistaa myös *3D-renderöinnin* (Wikipedia 2023b). Tässä kohtaa mainittakoon, että molemmat

Canvas ja WebGL tunnetaan myös *API:na* eli *ohjelmointirajapintoina* (*Application programming interface*). Ne ovat *ohjelmointiliitäntöjä*, jotka sallivat useiden sovellusten kommunikoinnin toistensa kanssa (Visma i.a.).

Kuvituksen *layerit* linkitetään osaksi JavaScriptiä nimeämällä ne erikseen mukaan koodiin. Tämän jälkeen ne listataan ja jokaiselle annetaan koodissa oma sijainti eli *position*. Sijainti määritellään *z-akselilla* eli syvyysuunnassa. Esimerkiksi taustan palloelementin asetan Vogelin esimerkin mukaisesti *z-akselilla* luvulle -2.25 ja sen eteen tulevat elementit, kuten vaikkapa hahmon vartalo, asetetaan pienemmillä luvuilla esimerkiksi kohtaan -1,8. Päälimmäinen kerros on aivan syvyysakselin pinnalla kohdassa 0.

X- ja Y-akseleilla kuva pysyy aina nollassa. Läpinäkyvyys eli *opacity* määritellään vakioksi eli numerolla 1, mutta näitäkin voi halutessaan säätää. Yksi mielenkiintoinen mahdollisuus on myös jokaisen *layerin* yksittäinen sekoitustila eli *blend*-asetus: koodiin voi asettaa myös kuvankäsittelystä tuttuja vaiheita kuten *multiply* tai *overlay*, ja efektit toimivat sitten asetetussa *layerissa*. Itse laitoin jokaisen 'null'-muotoon, sillä tämä kuvitukseni ei käytä sekoitustiloja.

Koodia tulee paljon, mutta lopuksi Vogel (i.a.) asettaa vielä *JavaScriptin* "päivittämään" *layerit* *drawCanvas*- sekä *requestAnimationFrame* -funktioilla (kuva 23). Tärkeää on myös ilmoittaa, että Canvas "tyhjenee" jokaisella animaation toistokerralla, jotta päällekkäisyyksiä ei jäisi muistiin. Lopputulemana Canvas päivittyy tasaisesti 60 kuvaa sekunnissa. Se ei katsojalle näy selaimessa sen kummemmin, mutta on Vogelin (i.a.) mukaan olennaisen tärkeä ominaisuus myöhempää interaktiivisuutta ajatellen.

```

87
88 layer_list.forEach(function(layer, index) {
89     layer.image.onload = function () {
90         load_counter += 1;
91         if (load_counter >= layer_list.length) {
92             requestAnimationFrame(drawCanvas);
93         }
94     }
95     layer.image.src = layer.src;
96 });
97
98 function drawCanvas () {
99     // clear whatever is in the canvas
100    context.clearRect(0, 0, canvas.width, canvas.height);
101
102    //Loop through each layer and draw it to the canvas
103    layer_list.forEach(function(layer, index) {
104
105        if (layer.blend) {
106            context.globalCompositeOperation = layer.blend;
107        } else {
108            context.globalCompositeOperation = 'normal';
109        }
110
111        context.globalAlpha = layer.opacity;
112
113        context.drawImage(layer.image, layer.position.x, layer.position.y);
114    });
115    requestAnimationFrame (drawCanvas);
116 }

```

Kuva 23. JavaScript-koodia tulee Vogelin tutoriaalissa paljon. (Visual Studio Code 2023)

Seuraavaksi lisätäänkin kuvituksen tuo interaktiivisuus. Ensimmäiseksi Canvasin taustaväri muutetaan väliaikaisesti punaiseksi, jotta arvoja asettaessa voi nähdä, missä kohtaa kuvituksen ”rajat” menevät. Sitten Vogel (i.a.) ohjeistaa lisäämään JavaScriptiin rivin *var moving = false*. Tämä vaikuttaa siten, että kuvitus ei liiku ruudulla vasta kun sille antaa komennon esimerkiksi ruutua koskella. Seuraavaksi koodissa asetetaan ”nollakohta” sormelle tai hiirelle, mistä liike lähtee. Tärkeää on myös lisätä funktio, jossa hiiren tai sormen liike vaikuttaa myös *layerien z-indeksiin* eli *syvyysindeksiin*. Viimeiseksi määritellään myös *offset*-komennolla, että klikkauksen tai painalluksen päättyessä kaikki elementit palautuvat takaisin kuvituksen määritellyyn ”nollakohtaan” eli alkuperäiselle paikalleen.

Tässä kohtaa animaatio toimii, mutta liike on rajatonta ja turhan voimakas. Kuvituksesta näkyy käännellessä helposti myös alueita, joita ei ole tarkoitus nähdä (kuva 24).



Kuva 24. Kuvitus toimii tässä vaiheessa interaktiivisesti, mutta kääntyminen on turhan voimakasta. (Tiina Hautamäki 2023)

Korjatakseen tämän Vogel asettaa esimerkissään *touch_multiplier*- arvon luvulle 0.3. Omalla kohdallani tämä oli edelleen liian voimakas, joten pienennän lukua desimaalin verran kohtaan 0.05. Tämän jälkeen kokonaisuus alkaa näyttää jo paljon paremmalta. Keppiosa tosin yhä edelleen irtoaa liikaa paikaltaan, joten pienennän koodissa sen syvyyttä z-indeksissä lähemmäs vartaloa. Tässä vaiheessa vaihdan myös Canvasin taustaväriin punaisesta värittömäksi, sillä en tarvitse sitä enää indikoimaan minulle missä reunat menevät.

Seuraavassa vaiheessa tutoriaalissa lisätään myös *motion control* eli liikekontrolli. Tässä vaiheessa on tärkeää määritellä muuttujat, missä asennossa laite on sivuston latautuessa ja mihin asentoihin se menee. Liikkeen tehoa säädetään vastaavasti *motion_multiplier*- arvoilla, kuten kosketukselle ja hiiren painallukselle tehtiin aiemmin. Vogel (i.a.) lisää kuvaan myös lisäkomennolla enemmän rotaatiota liikkeen kanssa kääntymisefektin tehostamiseksi. Lisättyäni itse koodin kuvani muuttui lähes kokonaan kaksiulotteiseksi, joten lisäsin syvyyssarjoja

layereiden välillä vieläkin isommiksi luvuiksi saadakseni kolmiulotteisuuden takaisin. Tärkeänä lopetuksena koodiin lisätään vielä komento, jolla näytön orientaation muuttuessa kaikki elementit palautuvat aina takaisin alkuperäiseen nolakohtaansa.

Loppusilauksena lisätään latausruutu CSS:n ja JavaScriptin avulla, jossa käsketään koko ruudun peittävän elementin viipyä kuvan edessä pari sekuntia ja häivyttää sitten pois. Tällä varmistetaan, että kuvitus on ehtinyt latautua kokonaisuudessaan näkyviin ennen kuin käyttäjä voi sitä tarkastella.

Vogel (i.a.) ei kuitenkaan ollut vielä valmis. Hän ehdottaa viimeistelemään työn lisäämällä sivustolle mukaan JavaScript-kirjaston nimeltään *tween.js*. Tämä mahdollistaa sen, että kuvitus aina palautuessaan alkuperäiselle paikalleen suorittaa siirtymän kevyellä animaatiolla napakan napsahtamisen sijaan. Tutoriaalini iän vuoksi kirjasto oli jo päivittynyt nimellä *tween.umd.js*, mutta sain sen kopioitua ja tallennettua *gitHub*-sivuston kautta. Tämän jälkeen kirjasto vielä linkitetään osaksi koodia HTML:n ja *main.js:n* kautta.

Lopputulokset näyttävät mielestäni varsin hyvältä, mutta jostain syystä liikkeellä toimivat kontrollit eivät reagoi omassa Applen puhelimessani laisinkaan. Chrome-selaimen *developer*-moodi näyttää, että gyroskooppi toimisi koodissa kuten pitääkin, joten koitin lähettää työstäni linkin Androidin omistaville ystäväilleni testattavaksi. Vain yksi heistä sai kuvaa käänneltyä puhelimensa liikkeillä, tosin varsin kankeasti. Toisekseen en saanut kuvitukseen minkäänlaista animaatiota rikkomatta kokonaisuutta täysin käyttökelpottomaksi. Vika lienee vanhentuneessa koodissa, olihan tutoriaali kuitenkin jo kolmisen vuotta vanha, ja puhelinten omat ohjelmistot ovat päivittyneet tuossa ajassa jo jonkin verran.

Päätän kokeilla huomattavasti yksinkertaisempaa tekniikkaa *parallax-efektin* luomiselle samalla kuvituksella. *Tilt.js* on yllättävän helppo kirjasto käytettäväksi. Ylimääräistä koodausta ei paljoa tarvita lisäyksen jälkeen, ja arvojen muokkaus tapahtuu pitkälti CSS:n ja HTML:n kautta. Efekti toimii hienosti hiiren liikuttelulla, mutta valitettavasti kirjoituksen hetkellä tässä ei myöskään toiminut

puhelimien liiketunnistus eikä yllättäen edes sormen painallukset, vaikka molempiin oli asetettu arvot koodin sisällä. Sain kuitenkin lisättyä nyt koodiin simppelein animaation *setInterval*-nimisen skriptin avulla. Animoitavat kuvaruudut listataan koodiin järjestyksessä ja ilmoitetaan vain aikajana, jolla ne vaihtuvat.

(bhawnaatrish 2021.) Lopuksi teen näille ylimääräisille kuvaruuduille saman CSS-koodiin osiot kuin muillekin kuville, ja asetan ne syvyydessä samaan kohtaan missä niiden vastineet ovat. Täten animaatio ilmestyy vain alkuperäisen kuvituksen eteen sekunnin ajaksi.

5.4 Facebookin syvyyskartta

Seuraavaksi kokeilen Facebookin ominaisuutta luoda 3D-kuvaa *depth mapin* eli *syvyyskartan* avulla. Käytän apuna Taboran (2019) ohjetta, jossa syvyyskartta luodaan valokuvaan Photoshopissa.

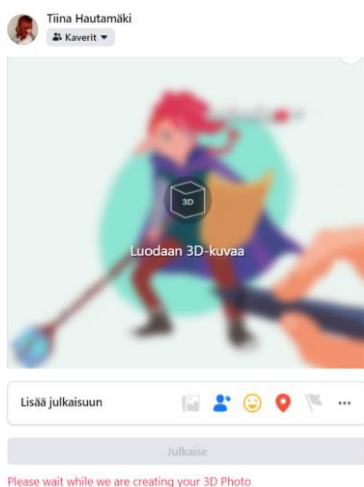
Aluksi avaan kuvituksen ja muunnan sen moodin harmaasävyihin. Ohje neuvoo irrottamaan eri syvyyksissä sijaitsevat elementit toisistaan tässä vaiheessa. Koska omalla kohdallani kuvituksen osat ovat jo omissa kerroksissaan, minun ei tarvitse prosessia enää tehdä. Seuraavaksi Tabora (2019) neuvoo määrittelemään syvyyden tummuusarvolla. Mitä kauempana kohde on, sitä tummempi se on sävyltään. Tämä toteutetaan säätämällä jokaisen *layerin* kontrastia ja valoa *brightness/contrast* säädöillä vaaleasta tummaan. Koen helpoimmaksi lisätä *alpha-lukituksen* jokaiseen *layeriini*, joka lukitsee läpinäkyvät pikselit maalattavan alueen ulkopuolelle. Tämän jälkeen maalaan *layereihin* suoraan valkoisen, harmaan ja mustan eri sävyillä. Väritän siis käden, energiapalkin ja kilven valkoisiksi kuvassa. Pää ja keppiosa ovat vain aavistuksen vaaleampaa harmaan sävyä vartalon kanssa, samaten hiusten takaosan maalaan hieman tummemmaksi. Näiden osien kuuluukin olla lähekkäin toisiaan, joten merkittävää sävyeroa ei tarvita. Viimeiseksi muutan taustapallon äärimmäisen tummaksi, jotta se olisi paljon kauempana *syvyyskartassa*. Tausta vielä mustaksi, koska kuva tallennetaan *jpg:nä*, joten ”läpinäkyvä” osa muuttuu valkoiseksi elementiksi. Tabora (2019) ohjeistaa sumentamaan kokonaisuuden *gaussian blurrilla* eli *Gauss-sumennuksella*. Tämä sumentaa kuvaa valittujen säätöjen mukaisesti.

(Adobe 2023.) Hänen esimerkissään säädetään valokuvaa, joka tarvitsee pehmennystä *syvyyskartassa* luonnollisuuden säilyttämiseksi. Koen, että tarkat rajat omaavan kuvituksen kanssa lopputulos olisi vain sotkuisempi, joten jätän tämän vaiheen pois. Lopullinen *syvyyskartta* on kuvassa 25.



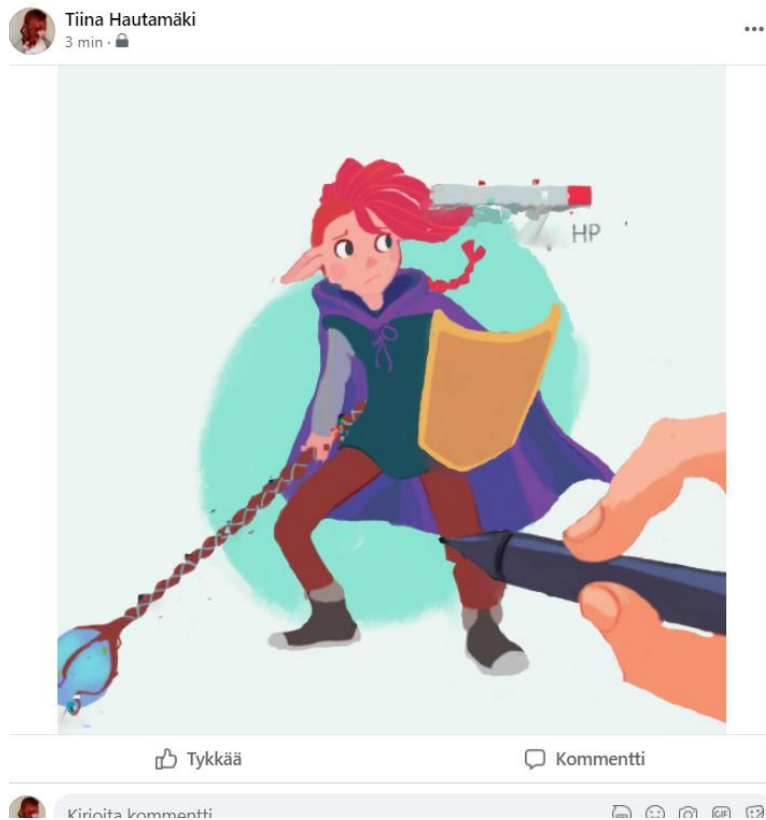
Kuva 25. Facebook tarvitsee parallax-efektin luomiseen *syvyyskartan* harmaasävyillä. (Tiina Hautamäki 2023)

Tämä kuva tallennetaan nimellä *image_depth.jpg*, kun taas varsinainen kuva on vain *image.jpg*. Kun molemmat lähetetään yhdessä Facebookin kuvan luontiin, se ymmärtää automaattisesti kyseessä olevan kolmiulotteinen kuva ja alkaa luomaan sitä annetun *syvyyskartan* perusteella (kuva 26).



Kuva 26. Facebook luo kolmiulotteisen kuvan automaattisesti. (Tiina Hautamäki 2023)

Kuvitus toimii käytännössä hyvin, mutta on silti kovin rosoinen reunoista ja tekee kummallisia virheitä muun muassa energiapalkin reunojen kanssa, kuten kuvassa 27 näkyy. Tämä johtunee siitä, että Facebookin 3D -efekti on luotu enemmän pehmennystä tarvitsevia valokuvia ajatellen.



Kuva 27. Automaattinen kuvitus toimii, mutta on reunoiltaan kovin rosoinen. (Tiina Hautamäki 2023)

Yksi hienous on kuitenkin, että kuva liikkuu valmiiksi Facebookin applikaatiossa myös puhelimen liikkeiden mukana, eikä tätä tarvitse säätää asetuksista erikseen.

5.5 Applikaatiot ja editorit

Seuraavaksi kokeilen muutamaa applikaatiota ja editoria, joilla on mahdollista luoda *parallax-efektiä* suoraan ohjelmasta käsin ja julkaista käyttövalmiina verkkoon. Näissä yhdistyvät koodauksen yksityiskohtaiset mahdollisuudet sekä Facebookin automaation helppous. Koodinkin käyttö on mahdollista jos tietämystä löytyy, mutta ei aina tarpeellista.

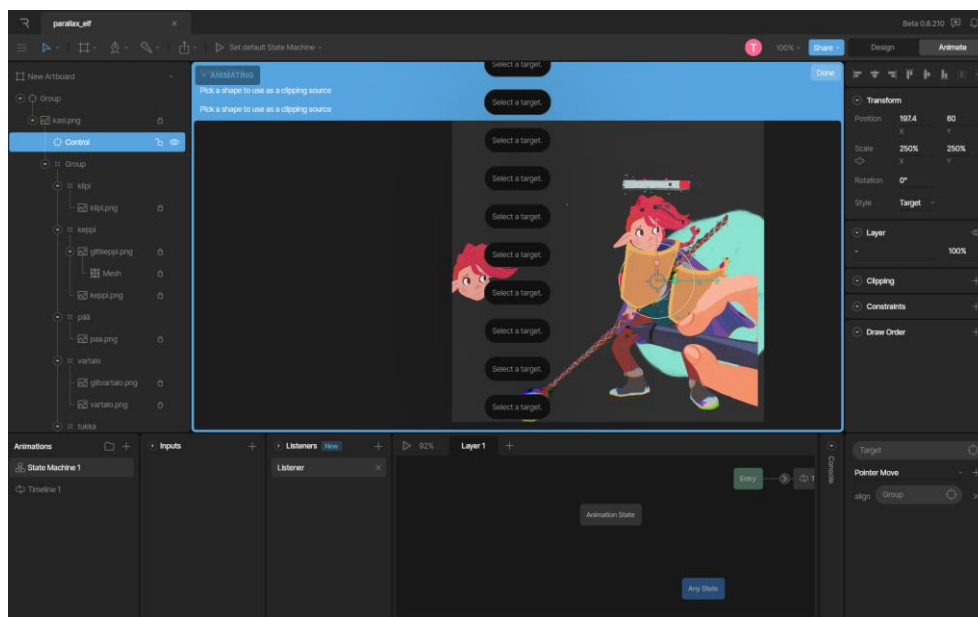
5.5.1 Rive

Rive on verkkopohjainen alusta, jossa käyttäjä luoda interaktiivista animaatiota kevyessä verkkoympäristössä. Tämän jälkeen tuotoksen voi julkaista Riven omalle verkkoalustalle, tai upottaa tiedoston omalle sivulleen eri *runtime*-kirjastojen kautta. Ehkäpä mainioin ominaisuus Riven omassa galleriassa on se, että muiden töitä voi esikatsella editorissa ja tarkastella, miten ne on valmistettu.

Käytännössä työskentely on yhtä helppoa kuin perinteisissä ohjelmissa. Kuva *layereineen* tuodaan alustalle ja asetellaan paikoilleen tyhjälle pohjalle. Animoointiprosessikin on varsin tuttu esimerkiksi Premierestä tai After Effectsistä - aikajanalla osia liikutellaan, koko- ja näkyvyysarvoja voi muunnella ja näistä muutoksista tallennetaan *keyframes* eli avainkehukset animaatiota varten. Animaation voi lopuksi laittaa "toistumaan" ikuisesti, jolloin se päätyttyään alkaa aina uudelleen ja uudelleen.

Riven galleriassa oli myös muutamia eri artistien esimerkkejä *parallax-efektin* käytöstä, jotka toimivat hiirtä liikuttamalla tai klikkaamalla. Käytännössä Rivessa ei kuitenkaan ole z-syvyysakselia, vaan illuusio luodaan animoimalla osat eri tahtiin *targettien* eli kohteiden, *constraints* eli rajoittimien ja polkujen avulla. Tämän jälkeen koodiin voi lisätä nappulasta *EventListener*-toiminnon eli "kuuntelijan", jonka kääsee kuuntelemaan esimerkiksi hiiren liikkeitä suhteessa tiettyyn kohteeseen, johon kuvan liike on yhdistetty.

Kokeilen tekniikkaa useaan otteeseen, mutta jostain syystä aina testini päättyä kuvituksen osat lennähtivät aina sinne tänne tai tuplaantuivat työskentelyalueen nurkkiin, vaikka uusia *layereita* niistä ei ollut syntynyt kuten kuvassa 28 näkyy. Rive on edelleen Beta-vaiheessa, mikä selvästi aiheuttaa näitä bugeja. Alustalla on kuitenkin valtavasti mahdollisuuksia. Uskon, että ihastun siihen itsekin, kunhan perehdyn ominaisuuksiin ajan kanssa ja sen kehitystyö etenee.



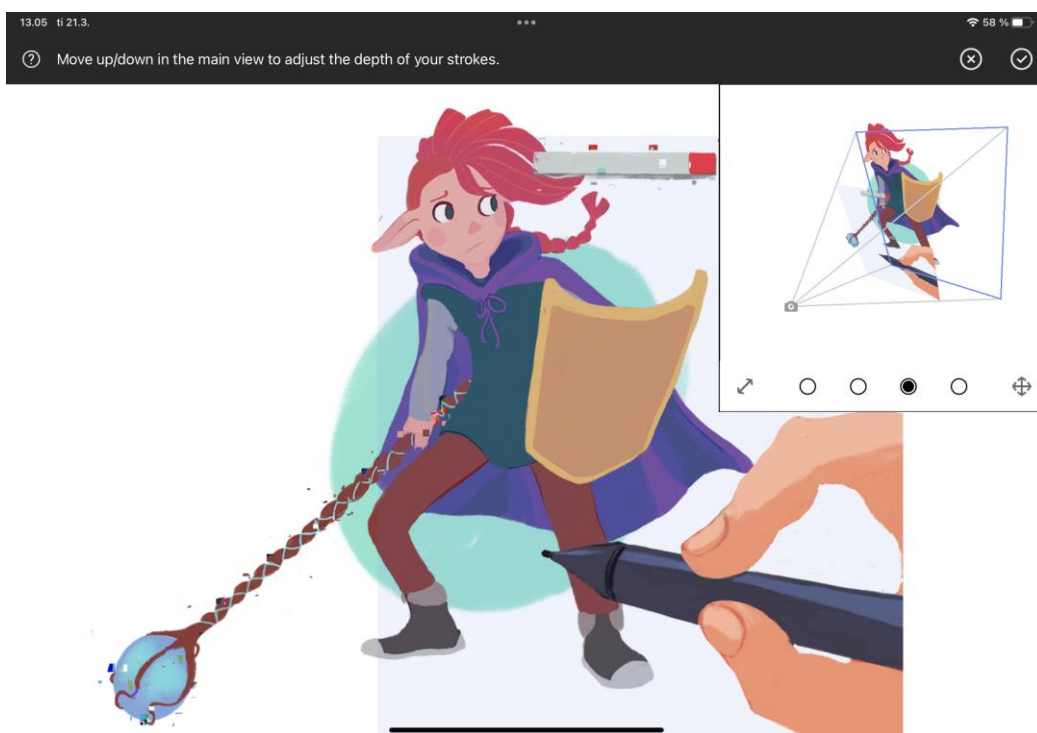
Kuva 28. Rive on näppärä, mutta ajoittain myös buginen ohjelma. (Tiina Hautamäki 2023)

5.5.2 Mental Canvas

Mental Canvas on saatavilla sekä Windowsille että iPadille, ja se on käyttöliittymältään yhtä lailla tuttu kuin aiemmatkin alustat – käytössä ovat *layerit* ja aikajana animaation tekoon kuten yleensäkin. Erityiseksi ohjelman tekee valmis syvyyden säätö, joka toimii varsin intuitiivisesti. Kameran asetuksissa voit valita joko panoroinnin, *tilttauksen* eli kallistuksen tai yhteen kohdepisteeseen kiinnittävän kuvakulman pyörittelyn. Kerroksia voi katsoa kahdella eri läpinäkyvyysasteella. Kerroksien syvyyttä säädettäessä yksitellen voi käyttäjä esikatsella työskentelyään useasta eri kuvakulmasta: suoraan ylhäältä, viistosti kulmasta sekä suoraan edestä (kuva 29). Valitsemaansa *layeria* siirrellään sormea tai Applen kynää raahaamalla eteen tai taakse. Windowsilla tämä varmasti sujuu hiirtä

käyttämällä. Työskentelypohjan kokoon on myös valittavana useita perinteisiä valmisasetuksia, mukaan lukien sosiaaliseen mediaan soveltuva neliö. Käytän tätä pohjaa kokeilussani.

Animaation luonti on kuitenkin keskittynyt enemmän kameran siirtelyyn eri kuvakulmissa. Kerroksia voi toki piilottaa ja tuoda taas esille kohtaus kerrallaan, mutta lyhin aika minkä sain kuvien vaihtumiseen oli 0,5 sekuntia. Tämä on aivan liian lyhyt aika toivomaani animaatiotavan luomiseen.



Kuva 29. Mental Canvasilla pystyy katsomaan helposti työskentelyänsä myös kameranäkymän ulkopuolelle. (Tiina Hautamäki 2023)

Valmiin tiedoston voi tallentaa sellaisenaan Mental Canvasin verkkoalustaan sopivalle formaatille, tai sitten tuoda ulos videotiedostona. Ilmaisversioon ilmestyy ohjelman oma vesileima. Mental Canvasin omalle alustalle en voi tätä kokeilua varten siirtää teostani, sillä ominaisuus on saatavilla vain kuukausittain maksaville käyttäjille. Siellä todennäköisesti kuvitusta olisi katsoja kuitenkin voinut pyöritellä vapaammin.

5.5.3 Adobe Animate

Adobe Animate oli pieni yllätys, jonka en ensin kuvitellut pystyvän tuottamaan toivomaani kuvitusta. Ohjelmassa on kuitenkin kirjoitushetkellä beta-versiona toimiva *VR360*-moodi. Nimensä mukaisesti tällä olisi lähtökohtaisesti määrä luoda virtuaaliympäristöön soveltuvaa kuvitusta, mutta luomuksia on mahdollista katsella myös hiirtä liikuttamalla *WebGL*-pohjan ansiosta. Ohjaava opettajani löysi keinon, jolla kameran liikettä pystyi *keyframeihin* eli avainkehyksiin liittävä koodin avulla rajoittamaan pienemmälle alueelle, joten kuvitusta oli mahdollista katsella perinteisen *parallax tiltin* tavoin. *Layereita* oli mahdollista myös kääntää vaakatasoon ja luoda kuvaan esimerkiksi kolmiulotteinen lattia, vaikkakin tämä toistaiseksi onnistui ainoastaan koodin avulla. Tämän kanssa tuli olla vain tarkka, sillä ohjelma ei toistaiseksi osaa leikata toisen elementin lävitse, joten kahden eri elementin olisi hyvä asettua limittäin sen sijaan, että ne kohtaisivat toisensa VR-ympäristössä (kuva 30).



Kuva 30. Adobe Animaten VR-moodissa on mahdollista myös kääntää kuvia vaakasuunnassa. (Tiina Hautamäki 2023)

Haasteeksi osoittautuu aluksi myös kuvituksen bittikarttapohja, jota ohjelma ei todennäköisesti Beta-version takia pystynyt lukemaan oikein VR-ympäristössä.

Keksimme, että rajaamalla kuvituksen tyhjän tilan vektorilla kaikki *layerit* skaalautuivat lopuksi oikein. Tärkeää oli myös muuttaa *layerit* ”elokuvaklapeiksi” *convert to symbol*-toiminnolla.

Animoinnin helppokäyttöisyyden takia tulin tulokseen, että teen lopullisen kuvitustyöni Adobe Animatea käyttämällä.

6 Lopullinen kuvitus

Lopullinen kuvitukseni on Adobe Animateella tehty aiemmin mainitun testikuvituksen pohjalta. Se sisältää yksinkertaista animaatiota kuten hahmon hännän liikettä. Siinä on myös testikuvituksen tavoin mahdollista kääntää kuvakulmaa ja tarkastella ympäristöä hiiren avulla *tilt parallaxin* periaattein. Animaatio toistuu ja palautuu aina ensimmäiseen kuvaruutuun loppuessaan.

Animaation tein *frame by frame* -periaatteella, jossa liike perustuu joka *framella* eli ruudulla vaihtuvaan uuteen kuvaan (Adobe 2021). Omassa tapauksessani käytän tekniikkaa yksinkertaisemmin ja vaihdan kuvitusta harvemmillä ruutuväleillä. Vesitason pohja on asetettu vaakasuuntaan aiemman luvun toimivaksi testatun koodin avulla. Keltainen taustaelementti on kauimmaisena syvyydessä ja asetettu *texture wrapping* -tilaan, eli kuva on ”kääriytynyt” pallomaisesti koko kuva-alueen reunalle. Valkoinen ”maski” aivan kameran edessä peittää kuvituksen reunat, mikä saa kokonaisuuden näyttämään siltä, että kuvaa katsoisi pienestä aukosta sisään.

Publish-toiminnolla ohjelma luo HTML-tiedoston ja tekee myös automaattisesti kansiorakenteen, johon se sisällyttää itse kuvituksen ja kaikki tarvittavat JavaScript-kirjastot verkkoon julkaisua varten. Käyttäjän tarvitsee vain lisätä HTML-tiedosto kansioineen valitsemaansa *webhotelliin*, missä verkkosivusto sijaitsee vuokratussa palvelintilassa (Väisänen 2020). Vaihtoehtoisesti sivustoa voi myös isännöidä omalta koneeltaan, mutta tämä vaatii enemmän tietämystä ja aikaa (One i.a.).

Kuvassa 31 on ruutukaappaus kuvituksesta. Se on tässä vielä luonnosvaiheessa, mutta toiminnallisuus on kohdillaan ja animaatio toimii. Valmista projektia voi tarkastella liitteen 1 linkin kautta.



Kuva 31. Lopullinen parallax-kuvitus, tosin visuaalisesti vielä karkeassa vaiheessa. (Tiina Hautamäki 2023)

7 Lopuksi

Parallax-efektin luominen verkkoympäristöön on täysin mahdollista jo nykyisillä työkaluilla, ja VR-tekniikan harppaukset mahdollistavat sen, että kehitys ei ole pysähtymässä lähiaikoina. Mahdollisuuksia tekemiseen on niin paljon kuin on tekijöitäkin. Tekniikka kehittyy kuitenkin nopeaa vauhtia, ja vaarana on, että tällä hetkellä toimivat alustat sekä JavaScript-kirjastot eivät taas toimi uusissa ympäristöissä muutaman vuoden päästä. Yhtä täysin yhteensopivaa koodia kaikille alustoille tai laitteille ei toistaiseksi myöskään näyttäisi olevan. Se mikä toimii toisaalla, voi olla aivan rikkinäinen muualla. Onkin tärkeää seurata ja pysyä ajan tasalla verkkoalustojen kehityksestä, mikäli aikoo erikoistua virtuaalisten kuvitusten luomiseen.

Tällä hetkellä koen, että turvallisinta olisikin pysyä ehkä valmiissa ohjelmissa ja alustoissa ellei ole täysin sinut koodaamisen kanssa. Koodia kun voi joutua säännöllisesti päivittämään yrityksen ja erehdyksen kautta. Henkilökohtaisesti haaveilen myös 3D-mallien yhdistämisestä kaksiulotteisten kuvien sekaan, ja

Adobe Animateella sekin onnistuu. Tiedän myös, että esimerkiksi Jarom Vogel on näitä portfolioissaan esitellyt. Tarkoitukseni onkin opiskella kyseistä tekniikkaa tarkemmin läheisessä tulevaisuudessa. Tämän opinnäytetyön prosessi oli mielenkiintoinen matka, ja seuran kehityksen kulkua innostuksella tulevaisuudessakin.

Lähteet

Adobe 2021. Create frame-by-frame animations. Verkkotutoriaali 27.4.2021. <https://helpx.adobe.com/animate/using/frame-by-frame-animation.html> (viitattu 14.4.2021).

Adobe 2023. Sumennusgallerian käyttö. Verkkotutoriaali 6.1.2023. <https://helpx.adobe.com/fi/photoshop/using/blur-gallery.html> (viitattu 3.4.2023.)

Candy Fish i.a. Verkkosivu. <http://candy.fish/> (viitattu 20.3.2023).

Deguzman, Kyle 2020. What is a Boom Shot? Camera Shot Definition and Examples. Blogi 20.9.2020. Studio Binder. <https://www.studio-binder.com/blog/what-is-a-boom-shot-definition-film/> (viitattu 20.3.2023).

DJI 2019. Film Like a Pro: How to Create Cinematic Orbit Shots. Verkkotutoriaali 28.6.2019. DJI Store. <https://store.dji.com/guides/film-like-a-pro-how-to-create-cinematic-orbit-shots/> (viitattu 20.3.2023).

DOGSTUDIO i.a. Verkkosivu. <https://dogstudio.co/> (viitattu 20.2.2023)

Frostbite Studio i.a. Viridian. Mental Canvas. Verkkokuvitus. <https://mentalcanvas.com/vs/00/m-yh2ftr9/2022-05-04T21-13-01.767Z/scene/index.html> (viitattu 13.3.2023).

Gaillard, Raoul 2021. Lightswan. Verkkosivu. <https://www.raoul-gaillard.com/work> (viitattu 29.2.2023)

Geeks for Geeks 2021. How to create Frame by Frame Animation using CSS and JavaScript? Verkkotutoriaali 28.7.2021. Geeks for Geeks. <https://www.geeksforgeeks.org/how-to-create-frame-by-frame-animation-using-css-and-javascript/> (viitattu 28.3.2023).

Gharat, Prajakta i.a. 5 Examples of Motion Parallax. Verkkosivu. Lambda Geeks. <https://lambdageeks.com/motion-parallax-example/> (viitattu 11.4.2023).

The Goonies 2022. The story of the Goonies. Verkkosivu. <https://the-goonies.webflow.io/> (viitattu 20.2.2023)

Gönen, Aykud 2018. Richer 3D Posts on Facebook and New Ways to Share. News for Developers. Blogi 20.2.2018. Meta. <https://developers.facebook.com/blog/post/2018/02/20/3d-posts-facebook/> (viitattu 11.4.2023).

Hawkes, Adam 2019. Knowing Your Track from Your Truck... Blogi 7.5.2019. Medium. <https://medium.com/pftrack-blog/knowning-your-track-from-your-truck-d7613aa42ab1> (viitattu 20.3.2023).

In Depth Cine 2022. The Pros and Cons of Shallow Depth of Field. Verkkootikkeli 31.11.2022. In Depth Cine. <https://www.indepthcine.com/videos/depth-of-field> (Viitattu 14.4.2023).

Janisch, Moritz 2021. Parallax Effect – filmmaking tutorial. Blogi 9.6.2021. Fenchel & Janisch Film Production. <https://www.fenchel-janisch.com/parallax-effect-filmmaking-tutorial/> (viitattu 16.3.2023).

Jyväskylän yliopiston informaatioteknologian tiedekunta 2006. Video- ja valokuvauksen suunnittelu ja tekniikka. Verkkoluento. <https://ap-pro.mit.jyu.fi/2006/syksy/ope/luennot/luento2/#TOC5> (viitattu 20.3.2023).

Kamerakoulu i.a. Objektiivit ja polttovälin merkitys valokuvauksessa. Verkkosivu. <https://kamerakoulu.fi/valokuvauksen-perusteet-objektiivit#polttovalien-vertailu> (viitattu 20.2.2023).

Keshtcher, Yuval 2020. The Complete Guide to Parallax Effects. Blogi 7.6.2020. Elementor. <https://elementor.com/blog/parallax-effects/> (viitattu 20.2.2023).

Kumar, Gaurav 2019. Walt Disney's Multiplane Camera: Illusion King. Verkkootikkeli 9.5.2019. Animaders. <https://animaders.com/walt-disneys-multiplane-camera/> (viitattu 14.3.2023).

LONGSHOT i.a. Verkkosivu. <https://longshotfeatures.com/> (viitattu 20.3.2023).

Moos, Caleb 2019. The Multiplane Camera. Animation Appreciation. Blogi 4.6.2019. Medium. <https://medium.com/animation-appreciation/the-multiplane-camera-a5581d7ea926> (viitattu 24.3.2023).

Myllyveräjä, Eetu 2020. Sosiaalisen median kuvakoot vuonna 2022 – tarkista oikea kuvakoko jokaiseen somekanavaan. Verkkootikkeli 27.5.2022. Duunitori. <https://duunitori.fi/tyoelama/sosiaalisen-median-kuvakoot> (viitattu 3.3.2023).

One i.a. Voinko isännöidä omaa nettisivua (nettisivu hosting)? Verkkosivu. <https://www.one.com/fi/webhotelli/nettisivu-hosting> (viitattu 17.4.2023.)

Pilon, Mary 2013. Tomato Can Blues. Verkkootikkeli. The New York Times. <https://www.nytimes.com/projects/2013/tomato-can-blues/index.html> (viitattu 14.3.2023).

Prendergast, Chris 2012. Parallax scrolling: Examples and history. Blogi 30.10.2012. ClickRain. <https://www.clickrain.com/blog/parallax-scrolling-examples-and-history> (viitattu 20.2.2023)

Rive i.a. Rive. Verkkosivu. <https://rive.app/> (viitattu 6.3.2023).

Tabora, Vincent 2019. Creating 3D Images on Facebook Using Depth Maps. High-Definition Pro. Blogi 21.5.2019. Medium. <https://medium.com/hd-pro/creating-3d-images-on-facebook-using-depth-maps-f078cf344975> (viitattu 11.4.2023).

Thomas, Bob 1958. Walt Disney: The Art of Animation. **1. painos**. New York: Simon & Schuster. 124.

Thompson, Ben 2019. Understanding One-Node Vs. Two-Node Cameras in After Effects. Blogi 5.12.2019. Action VFX. <https://www.actionvfx.com/blog/understanding-one-node-vs-two-node-cameras-in-after-effects> (viitattu 20.3.2023).

Trant, Christian 2020. Nämä 6 polttoväliä on syytä tuntea. Verkkootikkeli 15.11.2020. DigiKuva. <https://digi-kuva.fi/valokuvauslaitteet/objektiivit/nama-5-polttovalia-sinun-tulisi-tuntea> (viitattu 19.3.2023).

Visma i.a. API – Mikä on API? Verkkosivu. Visma. <https://www.visma.fi/epaseli/kirjanpidon-sanakirja/a/api/> (viitattu 19.3.2023).

Vogel, Jarom i.a. Art & Code: Create and Code an Interactive Parallax Illustration. Verkkovideosarja. Skillshare. <https://www.skillshare.com/en/classes/Art-Code-Create-and-Code-an-Interactive-Parallax-Illustration/1862124549> (viitattu 14.3.2023).

Vogel, Jarom 2021. Sword. Verkkokuvitus. <https://jaromvogel.com/illustration/sword2> (viitattu 20.2.2023).

Wang, SN. & Shyu, FM. 2014. Study on Parallax Scrolling Web Page Conversion Module. Intelligent Data analysis and its Applications, Volume I. Advances

in Intelligent Systems and Computing, vol 297. 235–243. https://doi-org.ezproxy.metropolia.fi/10.1007/978-3-319-07776-5_25 (viitattu 14.3.2023).

Väisänen, Viivi 2020. Mikä on webhotelli, mitä tarkoittaa hosting? Verkkootik-
keli 19.2.2020. Markkinointi Muikea. <https://markkinointimuikea.fi/mika-on-webhotelli-mita-tarkoittaa-hosting/> (viitattu 17.4.2023.)

Wikipedia 2023b. Canvas element. Verkkosivu 26.2.2023. https://en.wikipedia.org/w/index.php?title=Canvas_element&oldid=1141758448 (Viitattu 16.3.2023).

Wikipedia 2022c. JavaScript. Verkkosivu 29.10.2022. <https://fi.wikipedia.org/w/index.php?title=JavaScript&direction=prev&oldid=21402394> (Viitattu 10.3.2023)

Wikipedia 2023a. Parallax. Verkkosivu 23.2.2023. <https://en.wikipedia.org/w/index.php?title=Parallax&direction=prev&oldid=1148526079> (viitattu 10.3.2023)

Wikipedia 2022a. Parallax scrolling. Verkkosivu 15.9.2022. https://en.wikipedia.org/w/index.php?title=Parallax_scrolling&direction=prev&oldid=1139822729 (viitattu 18.2.2023.)

Wikipedia 2022b. Verkkosivu. Verkkosivu 6.7.2022. <https://fi.wikipedia.org/w/index.php?title=Verkkosivu&oldid=20578288> (viitattu 10.3.2023).

W3 Schools i.a. a. CSS Introduction. Verkkosivu. W3 Schools. https://www.w3schools.com/css/css_intro.asp (viitattu 14.3.2023).

W3 Schools i.a. b. How TO – Parallax Scrolling. Verkkosivu. W3 Schools. https://www.w3schools.com/howto/howto_css_parallax.asp (viitattu 10.3.2023).

W3 Schools i.a. c. HTML Canvas Graphics. Verkkosivu. W3 Schools. https://www.w3schools.com/html/html5_canvas.asp (viitattu 14.4.2023).

Liitteet

Linkki valmiiseen parallax-kuvitukseen videomuodossa

https://youtu.be/IOb0zsO_gHQ