Tampereen yliopisto

Jesperi Tolvanen

KATSAUS PINNANMUODOSTUSALGO-RITMEIHIN

Kandidaatintutkielma
Informaatioteknologian ja viestinnän tiedekunta
Tarkastajat: Henri Lunnikivi
Joulukuu 2022

TIIVISTELMÄ

Jesperi Tolvanen: Katsaus pinnanmuodostusalgoritmeihin

Kandidaatintutkielma

Tampereen yliopisto

Tietotekniikan kandidaatin tutkinto-ohjelma

Joulukuu 2022

Kolmiulotteisia pintoja käytetään tietokonegrafiikassa esimerkiksi lääketieteessä ja komponenttien suunnittelussa. Pinnanmuodostusalgoritmien avulla on myös mahdollista tehdä reaaliaikaisesti muokattavia pintoja, joita hyödynnetään esimerkiksi simulaatioissa. Pintojen tietokoneellisessa esityksessä tärkeää on esittää pinnat mahdollisimman vähillä resursseilla, kuitenkin säilyttäen piirteet mahdollisimman tarkasti. Tässä työssä tutkitaan kirjallisuuskatsauksen avulla kahta pinnanmuodostusmenetelmää. Nämä ovat Marching Cubes ja Dual Contouring. Tavoitteena on selvittää näiden menetelmien vahvuuksia ja heikkouksia sekä löytää vastaus siihen, millaisiin tilanteisiin kukin menetelmä toimii parhaiten.

Marching Cubes on tutkituista menetelmistä varhaisempi. Sen toiminta perustuu siihen, että mallinnettava alue jaetaan samankokoisiin kuutioihin. Pinta muodostetaan ennalta määritetyt hakutaulukon pohjalta. Tämä hakutaulukko sisältää kaikki mahdolliset pintaelementtiyhdistelmät, jotka pinta voi saada yhden kuution sisäisellä alueella.

Dual Contouring on tutkituista menetelmistä uudempi. Menetelmä perustuu reunojen tarkasteluun, ja etumerkkien muutokseen. Dual Contouring menetelmä ei hyödynnä hakutaulukoita, vaan pintaelementtien asettelu määräytyy ajonaikaisesti. Tämä mahdollistaa sen, että menetelmä voi muodostaa vaihtelevan kokoisia pintaelementtejä.

Kummallekin menetelmälle löydettiin etuja sekä haittoja. Marching Cubes -menetelmä on vahvimmillaan, kun ajonaikaista laskentaa halutaan minimoida. Menetelmän heikkoutena on terävien reunojen tuotto sekä huonosti optimoidun lopputulokset. Dual Contouring -menetelmä soveltuu tilanteisiin, joissa muodostettava pinta sisältää teräviä reunoja. Menetelmällä tuotetut pinnat ovat myös paljon optimoidumpia. Heikkoutena menetelmässä on se, että tuotetut pinnat ovat aina täysin vesitiiviitä, mikä voi aiheuttaa vääristymiä pinnan muotoon ja tuottaa fyysisesti mahdottomia pintoja. Lisäksi menetelmä tarvitsee enemmän ajonaikaista laskentaa.

Avainsanat: pinnanmuodostus, 3D-grafiikat, Marching Cubes, Dual Contouring, algoritmit

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck –ohjelmalla.

SISÄLLYSLUETTELO

1. JOHDANTO 1

2. PINNOISTA 2

3. MARCHING CUBES -ALGORITMI 5

4. DUAL CONTOURING -MENETELMÄ 8

5. MENETELMIEN VERTAILUA 12

5.1 Pinnasta tarvittava tieto 12

5.2 Laskentateho 12

5.3 Teräväreunaiset pinnat 13

5.4 Reaalimaailmaan verrattavat pinnat 14

5.5 Pintaelementtien yhtenäisyys 14

5.6 Samapinnan optimointi 15

6. YHTEENVETO 17

LÄHTEET 19

.

1. JOHDANTO

Kolmiulotteisuus on merkittävä osa nykyisiä tietokoneohjelmistoja. Datan esittäminen kolmannessa ulottuvuudessa helpottaa ymmärtämistä, sillä näin havainnollistettava asia voidaan esittää ihmiselle luonnollisessa muodossa. Kolmiulotteisia tietokonemalleja hyödynnetään lisäksi myös simulaatiotarkoituksessa. Tietokonemallien käyttö vähentää esimerkiksi materiaalikustannuksia, minkä seurauksena simulaatioiden käyttö muun muassa tutkimuksissa ja elokuvateollisuudessa lisääntyy jatkuvasti.

Pintojen muodostus on ollut erityisen merkittävässä asemassa lääketieteessä. Pinnanmuodostusmenetelmien kehityttyä tietokonetomografian ja magneettikuvauksen avulla saadusta mittausdatasta voitiin muodostaa tarkkoja malleja esimerkiksi sisäelimistä. Suurena erona aiempiin kuvausmenetelmiin oli kyky tarkastella näitä malleja useasta eri suunnasta [1]. Pinnanmuodostusalgoritmeille on löytynyt tämän jälkeen käyttöä monella alalla. Viime aikoina yleiseksi noussut käyttökohde on reaaliaikaisesti muokattavien pintojen muodostus. Reaaliaikaisesti muokattavia pintoja käytetään esimerkiksi simulaation, tai videopelin maaston luomisessa. Tällaisen maaston tapauksessa käyttäjällä on mahdollisuus muokata maaston muotoa lähes rajoituksetta ja nähdä tekemänsä muutokset välittömästi. [2]

Tietokonegrafiikassa pinnat ovat aina äärellisiä yksinkertaistuksia, sillä tietokoneiden resurssit ovat rajalliset. Pinnanmuodostuksen kesto-ongelmiin kuuluvat muun muassa laskentateho, piirteiden säilyvyys ja muodostetun pinnan optimointi. Tavoitteena on tuottaa tarkkoja esityksiä pinnoista siten, että muodostukseen ja esitykseen tarvittavat resurssit pysyvät minimissä.

Tämä tutkielma on kirjallisuuskatsaus, jossa tutustutaan kahteen tunnettuun pinnanmuodostusmenetelmään. Tavoitteena on selvittää näiden menetelmien hyötyjä ja heikkouksia. Tarkoituksena ei ole tutustua yksityiskohtaisesti menetelmien matemaattisiin ominaisuuksiin, vaan löytää vastaus siihen, millaisiin tilanteisiin kutakin menetelmää on järkevää käyttää.

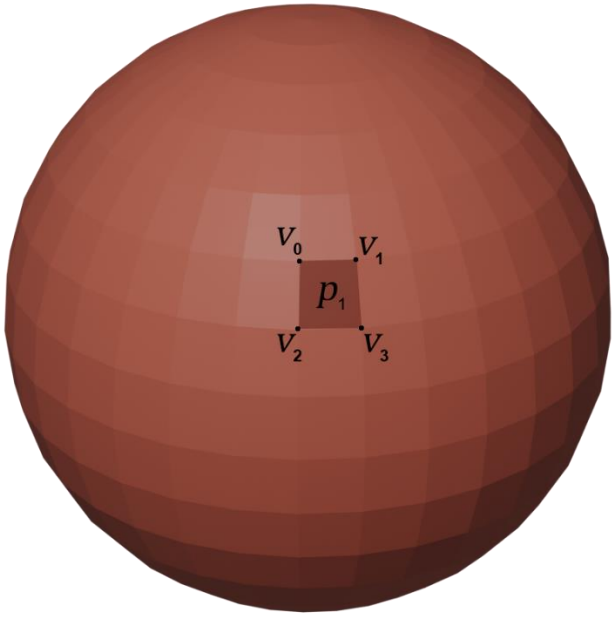
Luvussa 2 käsitellään yleisellä tasolla pintojen ominaisuuksia. Luvussa 3 käsitellään Marching Cubes -menetelmää. Luvussa 4 käsitellään Dual Contouring -menetelmää. Luvussa 5 vertaillaan menetelmien tehokkuutta sekä niiden muodostamien pintojen ominaisuuksia.

2. PINNOISTA

Tässä luvussa tutustutaan pintojen yleisiin ominaisuuksiin. Osiossa selvennetään matemaattisten jatkuvien pintojen sekä tietokonegrafiikassa esitettyjen äärellisten pintojen eroja. Luvussa perehdytään myös pinnanmuodostukselle tyypillisiin menetelmiin.

Pinta (engl. surface, implicit surface) on kolmannen ulottuvuuden pistejoukko. Yleisessä muodossa pinta voidaan esittää kolmen muuttujan jatkuvana funktiona $S(x, y, z) = 0$. Tietokonegrafiikassa käytetään termiä samapinta (engl. isosurface), jolla viitataan useimmiten äärelliseen esitykseen pinnasta S. Tämä tarkoittaa pinnan S approksimaatiota, joka on määritelty rajallisella määrällä näytepisteitä. [1] Nykytietokoneilla esitetyt kolmiulotteiset pinnat ovat aina rajallisella määrällä näytepisteitä esitettyjä samapintoja. Tässä työssä äärelliseen approksimaatioon viitataan termillä samapinta. Jatkuvasta funktiosta S käytetään termiä pinta.

Pinnanmuodostusalgoritmien lopputuloksena on verkko (engl. mesh), joka koostuu pintaelementeistä (engl. surface element). Tietokonegrafiikassa pintaelementit ovat usein kolmioita tai nelikulmioita. Kuvassa 1 esitetty pallo koostuu nelikulmaisista pintaelementeistä. Verkko koostuu kahdesta listasta: kärkilistasta ja pintaelementtilistasta. Kärkilistan jokainen alkio sisältää yksittäisen verkon kärjen (engl. vertex; vertices) koordinaatit kolmiulotteisessa avaruudessa. Pintaelementtilistan alkio sisältää viitteet niihin kärkilistan alkioihin, jotka määräävät kyseisen pintaelementin [1]. Esimerkiksi kuvaan 1 merkitty pintaelementti p_1 on muotoa (v_0, v_1, v_2, v_3), sillä se muodostuu näistä neljästä kärjestä.



Kuva 1. *Tietokone-esitys pallosta.*

Samapintojen muodostamiseen käytetään usein mallinnusohjelmistoja, jollaisia ovat esimerkiksi Blender ja Maya. Nämä ohjelmistot hyödyntävät monikulmiopohjaista mallinnusta (engl. polygonal modeling), mikä tarkoittaa, että verkkoa muokataan kärkiä siirtämällä. Menetelmä on intuitiivinen käyttäjälle, mutta rajoitteena on, että käyttäjän on aina aseteltava kärkien sijainti itse. Tässä työssä esitetyt menetelmät muodostavat samapinnan suoraan numeerisesta datajoukosta, esimerkiksi skalaarikentästä. Käyttäjä ei siis muokkaa verkkoa kärkien avulla, vaan muuttamalla datajoukon arvoja. Tällöin verkko muodostetaan uudestaan datajoukon pohjalta muokkauksen jälkeen. Menetelmää kutsutaan tilavuudelliseksi kuvanveistoksi (engl. volumetric sculpting). Vaihetta, jossa samapinta muodostetaan numeerisesta datasta, kutsutaan pinnanmuodostukseksi (engl. isosurface extraction). [2]

Pinnanmuodostukseen on monia menetelmiä. Yhteinen ominaisuus näille on, että pinnasta S muodostetaan diskreetti arvio näytteistämällä pinta soluihin, usein ruudukkoon. Karkeasti pinnanmuodostusmenetelmät voidaan jakaa ruudukolliseen (engl. grid-based) ja ruudukottomaan (engl. grid-less). Ruudukottomat menetelmät aloittavat jostain karkeasta pinnan S approksimaatiosta, jonka jälkeen laatua parannetaan iteratiivisesti viemällä kärkiä lähemmäs pintaa S. Ruudukolliset menetelmät jakavat pinnan osat soluihin, esimerkiksi yhdenkokoisiin kuutioihin. Muodostettava pinta näytteistetään sen perusteella, kuinka pinta S leikkaa kuution särmät. Pinnan näytteistetyt osat liitetään toisiinsa muodostamalla näytepisteiden väliin pintaelementti, yleensä kolmio tai nelikulmio. [3] Tässä työssä esitetyt menetelmät ovat ruudukollisia.

Tyypillinen esitysmuoto on merkitty etäisyyskenttäfunktio (engl. signed distance field function), joka määrittää jokaiselle pinnan pisteelle $[x, y, z] \in \mathbb{R}^3$ etäisyyden $f(x, y, z) = \mathrm{dist}([x, y, z], S)$. Etäisyysarvolle määritetään etumerkki pisteen sijainnin perusteella. Etumerkki on positiivinen, kun piste on pinnan S rajaaman alueen ulkopuolella, ja negatiivinen, kun piste on pinnan S rajaaman alueen sisäpuolella. Tämä esitystapa helpottaa esimerkiksi pisteiden sijainnin esittämistä sekä monien aritmeettisten operaatioiden suorittamista. [3] Tässä työssä esitetyt menetelmät hyödyntävät etumerkin muutosta pinnan muodostuksessa.

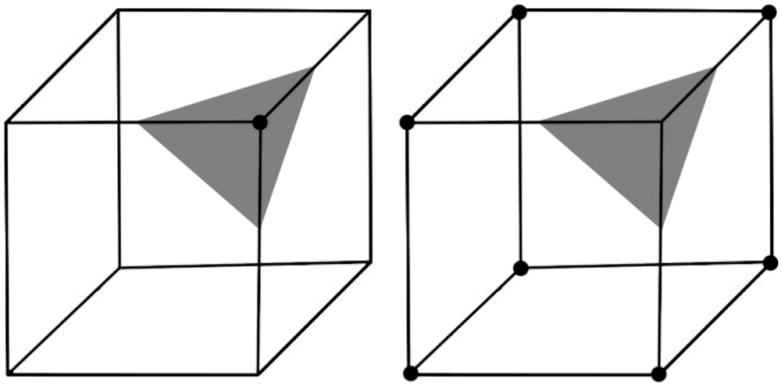
3. MARCHING CUBES -ALGORITMI

Tässä luvussa käydään läpi Marching Cubes -algoritmin toimintaperiaate. Marching Cubes on hajota ja hallitse -algoritmi, joka muodostaa kolmiopohjaisia malleja mittausdatan perusteella. Algoritmin ovat kehittäneet William E. Lorensen ja Harvey E. Cline vuonna 1987 ja se kehitettiin kolmiulotteiseen kuvantamiseen lääketieteessä [4].

Algoritmi perustuu siihen, että data-alue, joka sisältää mallinnettavan pinnan, jaetaan keskenään yhtä suuriin kuutioihin. Hajota ja hallitse -periaatteen mukaisesti jokainen kuutio käsitellään yksitellen, riippumatta muista kuutioista. Kuution ja pinnan leikkauskohdat määritetään kuution kärkikoordinaattien perusteella: jos kuution kärkipisteen korkeuskoordinaatti on pienempi kuin pinnan korkeuskoordinaatti kärjen kohdalla, pinnan osa on kuution kärjen yläpuolella. Tällöin kuution kärjelle asetetaan arvo 0. Muussa tapauksessa pinnan osa on kuution kärjen alapuolella, ja kärjen arvoksi asetetaan 1. Pinta leikkaa ne kuution särmät, missä yksi kärki on pinnan ulkopuolella (arvolla 1) ja toinen kärki on pinnan sisäpuolella (arvolla 0). Kuution sisällä olevan pinnan osan topologia määritetään tällä perusteella. [4]

Jokaisessa kuutiossa on kahdeksan kärkeä, ja kaksi tilaa: sisäpuolella tai ulkopuolella. Näin ollen pinnan on mahdollista leikata kuutio $2^8 = 256$ eri tavalla. Nämä leikkauskombinaatiot voidaan sijoittaa hakutaulukkoon hyödyntäen kärjille asetettuja 0–1 arvoja [4]. Hakuindeksinä on siten kärkien arvojen muodostama kahdeksan merkin binääriluku.

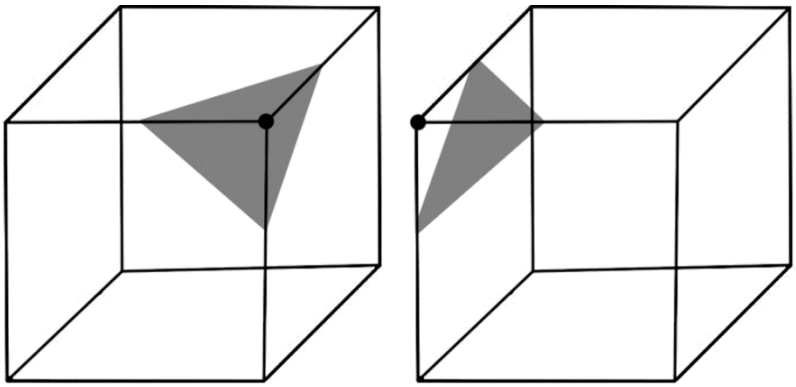
Kaikkien 256 kombinaation laskeminen on työlästä. Symmetria huomioon ottaen laskettavien tapausten määrää voidaan kuitenkin vähentää merkittävästi. Kun kuution kärkien arvot vaihdetaan vastakkaisiksi, muodostettavan kolmiopinnan rakenne pysyy samana. Tätä kutsutaan komplementtisymmetriaksi [4] ja se on esitetty kuvassa 2.



Kuva 2. *Komplementtisymmetria. Kun kuution kärkien arvot vaihdetaan vastakkaisiksi, kolmiopinta säilyy ennallaan.*

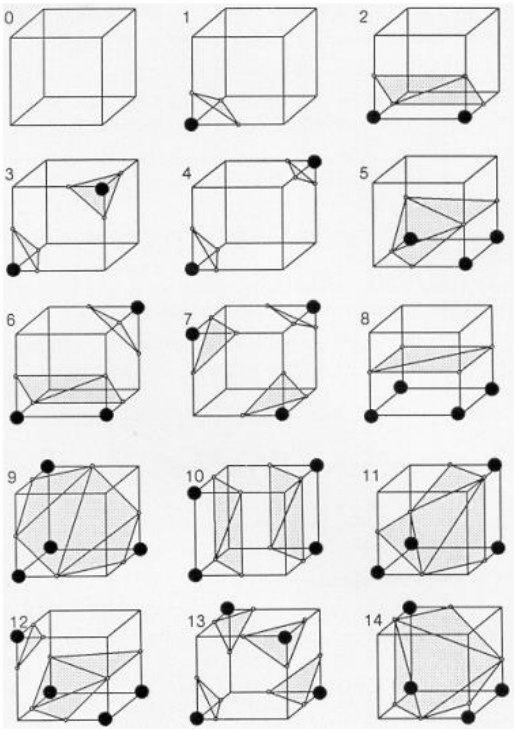
Komplementtisymmetria huomioon ottaen laskettavaksi jää vain ne kuutiot, joilla on nollasta neljään kärkeä arvolla 1. Laskettavien tapausten määrä on tällöin $256 \div 2 = 128$.

Kombinaatioiden määrää voidaan vähentää edelleen, kun huomioidaan kierteinen symmetria. Kierteisesti symmetrisiä ovat sellaiset tapaukset, jotka voidaan muodostaa jostain toisesta tapauksesta kääntämällä sitä jonkin koordinaattiakselin ympäri. Kuvassa 3 on esimerkki kierteisestä symmetriasta.



Kuva 3. *Kierteinen symmetria. Oikealla oleva tapaus saadaan kääntämällä vasemmalla olevaa kuutiota 90 astetta myötäpäivään.*

Kun huomioidaan komplementtisymmetrian lisäksi kierteinen symmetria, käsiteltäväksi jää 14 erilaista tapausta [4]. Nämä tapaukset on esitetty kuvassa 4.



Kuva 4. *Marching Cubes -algoritmin pintakombinaatiot. [4]*

Mahdolliset tapaukset tallennetaan hakutaulukkoon. Hakutaulukon perusteella jokainen kuutio voidaan käsitellä itsenäisesti. Taulukon indeksoinnissa käytetään kuution kärjille asetettuja arvoja. Jokaiselle kuutiolle määritetään kahdeksan merkkiä pitkä binäärilukuindeksi kärkien arvojen perusteella. Tällä indeksillä haetaan hakutaulukosta kaikki kuution särmien ja muodostettavan pinnan leikkauskohdat. Pinnan leikkaus voidaan tämän jälkeen interpoloida pitkin leikattua kuution särmää [4].

Lopuksi algoritmin on vielä määritettävä yksikkönormaalit jokaiselle muodostetun kolmion kärjelle. Yksikkönormaaleja käytetään käännösalgoritmeissa (engl. rendering) tuottamaan oikeanlainen varjostus muodostetulle kolmiopinnalle. Yksikkönormaalit määritetään laskemalla gradienttivektorit kuution kärjille. Gradientti leikkauspisteessä voidaan siten arvioida lineaarisen interpoloinnin avulla. Yksikkönormaalit voidaan laskea jakamalla saatu arvio kuution reunan pituudella. [4].

4. DUAL CONTOURING -MENETELMÄ

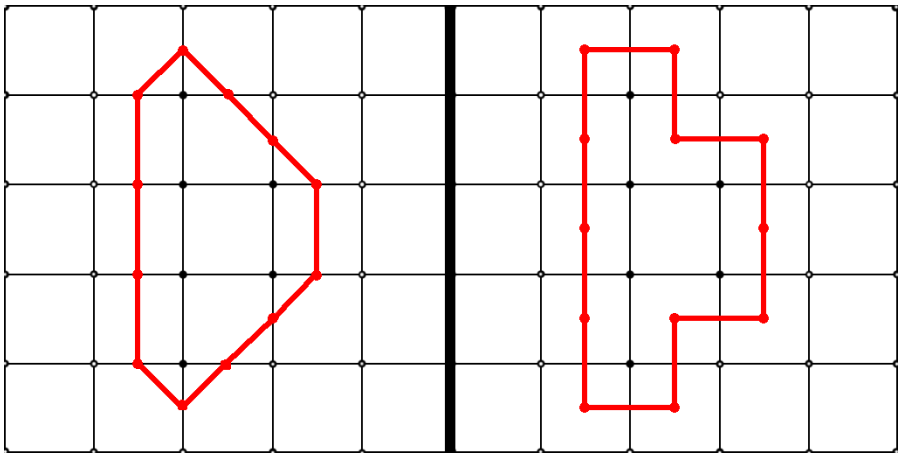
Dual Contouring on pinnanmuodostusmenetelmä, jonka tavoitteena on tuottaa piirteet säilyttäviä ja yksinkertaistettuja pintoja. Menetelmän ovat esittäneet Tao Ju, Frank Losasso, Scott Schaefer ja Joe Warren vuonna 2002. Menetelmän tavoitteena on tuottaa piirteet säilyttäviä pintoja, jotka ovat aiempien menetelmien tuloksiin verrattuna yksinkertaisempia. [5]

Dual Contouring menetelmän ensimmäinen vaihe on määrittää muodostettavan pinnan kärkien sijainnit. Kärki muodostetaan aina, kun kuution sisällä tapahtuu etumerkin muutos. Kärkien sijoittamisessa hyödynnetään näytepisteiden arvoja, sekä pintanormaaleja näissä pisteissä: jos pisteen ja pintanormaalin muodostama taso leikkaa jonkin toisen näytepisteen muodostaman tason, samapintaan on muodostettava kärki. Kärjen sijainti määritetään minimoimalla neliöllinen virhefunktio (engl. quadratic error function, QEF), joka lasketaan leikkaavien tasojen etäisyyksien neliöiden summana. Virhefunktio on siis muotoa

$$E[x] = \sum_i (n_i \cdot (x - p_i))^2,$$

missä p_i on pinnan ja kuution särmän leikkauspiste, n_i yksikkönormaali tässä pisteessä ja x muodostettavan kärjen sijainti [5]. Minimoimalla tämä funktio ja ratkaisemalla se muuttujan x suhteen saadaan koordinaattipiste, johon kärki tulee muodostaa. Menetelmä perustuu Extended Marching Cubes -algoritmiin (EMC), joka on laajennus Marching Cubes -algoritmista. EMC pyrkii parantamaan muodostetun pinnan laatua lisäämällä kärkiä kuutioruudukon reunojen ulkopuolelle [3]. Dual Contouring ja Extended Marching Cubes eroavat toisistaan siten, että Dual Contouring -menetelmässä samapinnan kaikkien kärkien sijainti määritetään virhefunktion avulla.

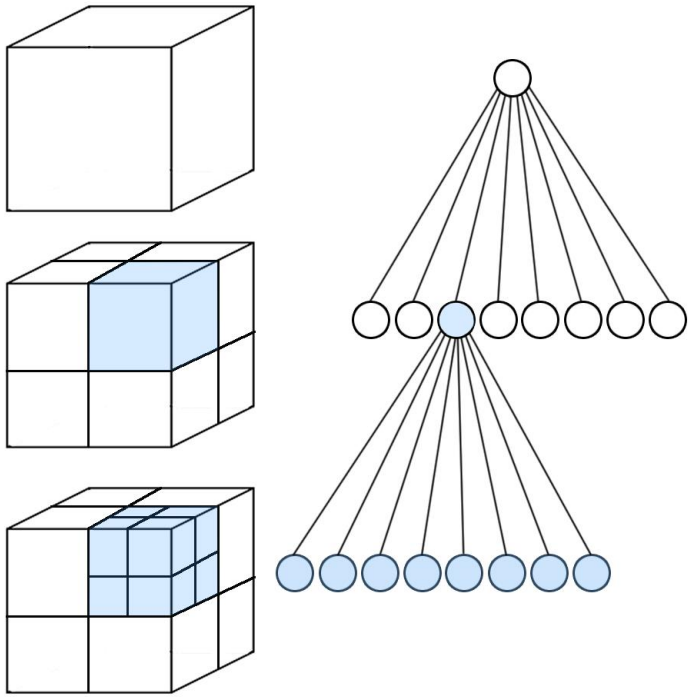
Dual Contouring menetelmässä kärkien sijainnit eivät rajoitu kuutioruudukon reunoihin. Koska kärjen sijainti määritetään virhefunktion avulla, kärki voi sijaita missä tahansa kohtaa kuutiota. Tämä on merkittävä ero verrattuna Marching Cubes -algoritmiin, jossa kärjet ovat aina kuutioruudukon särmillä. [2] Kuvassa 5 on esitetty näiden menetelmien ero kaksiulotteisessa tapauksessa.



Kuva 5. *Vasemmalla Marching Cubes -menetelmällä tuotettu ääriviiva, oikealla Dual Contouring -menetelmällä tuotettu ääriviiva.*

Toisin kuin Marching Cubesiin perustuvat menetelmät, Dual Contouring -menetelmä ei hyödynnä hakutaulukoita kuutioiden välisten osien yhdistämiseksi. Samapinnan muodostukseen hyödynnetään etumerkkien muutosta. Jokainen reuna, jossa ilmenee etumerkin muutos, luodaan nelikulmio yhdistämään minimoivat kärjet niiltä neljältä kuutiolta, jotka sisältävät tämän reunan [5].

Dual Contouring -menetelmä on adaptiivinen, mikä tarkoittaa, että se toimii myös vaihtelevan kokoisella ruudukolla. Adaptiivisuus saadaan aikaan vaihtamalla yhdenkokoinen kuutioruudukko kasipuuhun (engl. octree) [5]. Kasipuu on tietorakenne, jossa jokaisella puun solmulla on enintään kahdeksan lapsisolmua. Tietokonegrafiikassa kasipuuta käytetään tilan ositukseen, jota hyödynnetään esimerkiksi näkyvyyden tarkastuksessa ja pintojen tarkkuustason (engl. Level of Detail, LOD) määrityksessä. [6] Kuvassa 6 on esitetty malli tilan osituksesta käyttäen kasipuuta.

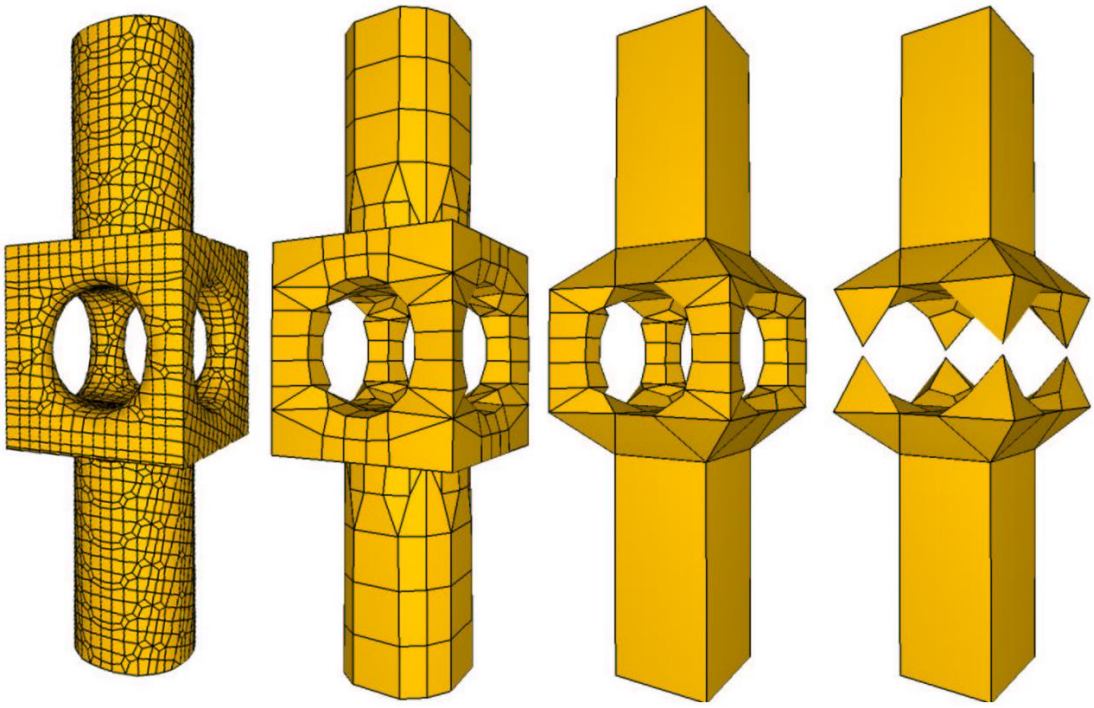


Kuva 6. *Tilan ositus kasipuun avulla.*

Adaptiivinen Dual Contouring -menetelmä hyödyntää kasipuuta vähentääkseen samapinnan kärkien ja monikulmioiden määrää. Esimerkiksi suuri osa pinnan ääriviivan sisäpuolelle jäävistä kuutioista ovat samanarvoisia (engl. homogenous), eli jokaisella kuution kärjellä on sama arvo. Tällaiset kuutiot voidaan yhdistää yhdeksi isommaksi kuutioksi tilan säästämiseksi. [5] Kasipuurakenteessa menetelmää kutsutaan puun tiivistämiseksi (engl. collapsing).

Adaptiivisessa Dual Contouring -menetelmässä on kolme vaihetta:

- Muodostetaan etumerkitty kasipuu, jonka samanarvoiset lehdet ovat täysin tiivistetyt. Tiivistäminen voidaan toteuttaa esimerkiksi aloittamalla yhdenkokoisesta ruudukosta ja yhdistämällä samanarvoisia kuutioita isommiksi kuutioiksi.
- Määritetään neliöllinen virhefunktio jokaiselle eriarvoiselle (engl. heterogenous) lehdelle, eli sellaiselle kuutiolle, joka leikkaa pinnan ääriviivan. Tämän jälkeen yksinkertaistetaan kasipuu käyttämällä määritettyä virhefunktiota. Tavoitteena on yhdistää puuelementin lapsialkioiden kärjet yhdellä, korkeammalla puussa olevalla kärjellä.
- Muodostetaan monikulmiot rekursiivisesti yksinkertaistetulle puulle. [5]



Kuva 7. *Vasemmalla ei-adaptiivisesti tuotettu pinta, oikealla yksinkertaistettuja pintoja [5].*

Kuvassa 7 on esitetty pintoja, joista ensimmäinen on muodostettu vakiokokoisella ruudukolla ja jälkimmäiset vaihtelevalla ruudukon koolla. Ei-adaptiivisessa menetelmässä pinnat muodostuvat nelikulmioista. Kun hyödynnetään adaptiivista menetelmää, muodostetussa pinnassa voi esiintyä myös kolmioita.

5. MENETELMIEN VERTAILUA

Tässä luvussa vertaillaan Marching Cubes -menetelmää ja Dual Contouring -menetelmää sekä näiden muodostamien samapintojen laatua.

5.1 Pinnasta tarvittava tieto

Marching Cubes -algoritmia käytetään tuottamaan pintoja skalaarikenttädatasta. Muodostettava pinta määritetään ainoastaan skalaarikentän arvojen etumerkin perusteella. Pinnan muodostukseen ei tarvita muuta tietoa datajoukosta. [2] Näin ollen muodostettavan pinnan matemaattiset ominaisuudet eivät vaikuta algoritmin toimintaan, mikä tekee menetelmästä helppokäyttöisen.

Dual Contouring -menetelmä tarvitsee myös muodostettavan pinnan arvon jokaisessa pisteessä. Tämän lisäksi menetelmä tarvitsee myös derivaatan (gradientin) arvon jokaisessa pisteessä, sillä kärkien sijoittamiseen käytetään pinnan normaaleja. Pinnasta tarvitaan siis tarkat sijainnit ruudukon ja pinnan leikkauspisteille sekä pinnan normaalit näissä pisteessä [5]. Joissain tapauksissa pinnan derivaattaa ei voida määrittää, tai sen määritys on työlästä. Tällöin gradientin arvo joudutaan arvioimaan esimerkiksi numeerisin menetelmin. [3] Tämä voi tehdä menetelmästä monimutkaisemman ja vaikeammin ymmärrettävän.

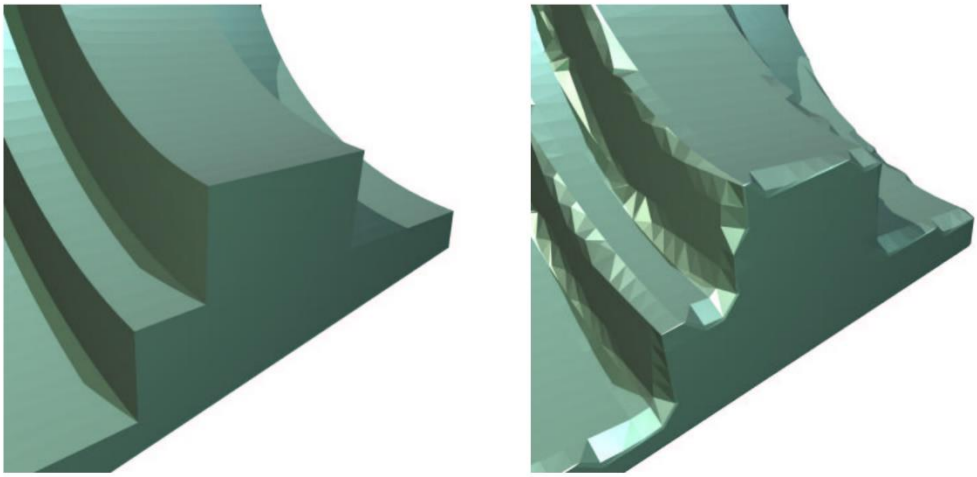
5.2 Laskentateho

Rakenteeltaan Marching Cubes -algoritmi on yksinkertainen. Pinta muodostetaan aina hakutaulukon perusteella vakiokokoisilla kuutioilla, mikä minimoi reaaliaikaisen laskentalogiikan tarvetta [2]. Algoritmi suoritetaan yksi kuutio kerrallaan, eikä yhden kuution käsittelyssä tarvita tietoa muista kuutioista, joten algoritmin tehokkuus on lineaarinen $O(n)$ [1]. Koska jokainen kuutio käsitellään samalla tavalla, algoritmin suorittaminen rinnakkaisesti on helppoa. Suoritusnopeutta saavutetaan etenkin, kun toteutuksessa käytetään erillisten grafiikkaprosessorien laskentatehoa. [7]

Dual Contouring -menetelmä ei hyödynnä hakutaulukoita, vaan jokaisen kärjen sijainnin määrittämiseksi on laskettava neliöllinen virhefunktio. Tämän seurauksena tarvittava laskentateho on suurempi ja algoritmin suoritus vaatii enemmän aikaa [2].

5.3 Teräväreunaiset pinnat

Marching Cubes -algoritmilla ei saada aikaan tarkkoja esityksiä teräville pinnoille, sillä algoritmi tuottaa laskostumisartefakteja terävien reunojen kohdalle. Kuvassa 8 on esitetty Marching Cubes -algoritmilla muodostettu teräväreunainen kappale.



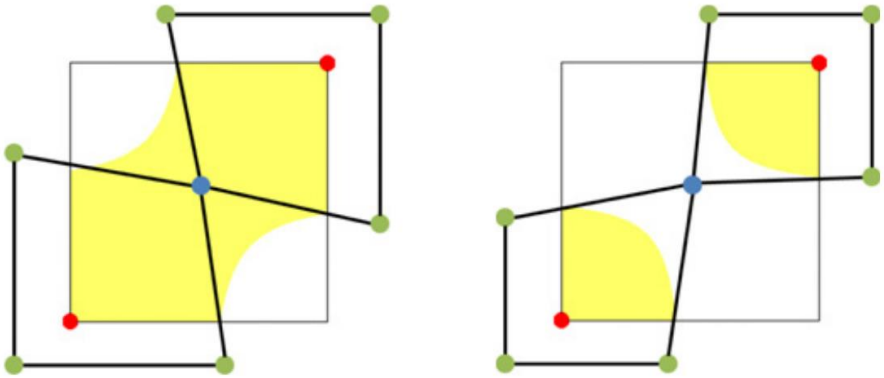
Kuva 8. *Alkuperäinen teräväreunainen pinta ja Marching Cubes -algoritmilla tuotettu samapinta [3].*

Marching Cubes -algoritmi hyödyntää lineaarista interpolointia kolmiopinnan muodostamiseen. Näytteistyspisteet määritetään interpoloimalla etäisyysarvot kahden ruudukkopisteen välillä [4]. Menetelmä on yksinkertainen, mutta terävien reunojen tapauksessa näytepisteet eivät välttämättä mukaile todellista pintaa. Tällöin lineaarinen interpolointi ei anna hyvää arviota pinnan muodosta. Tuloksena on ”pehmennetty” reuna terävän reunan sijasta [3]. Pienentämällä kuutioruudukon kokoa artefaktien kokoa voidaan pienentää, mutta perusongelma säilyy samana. Lisäksi kuutioiden koon pienentäminen lisää samapinnan monikulmioiden määrää, mikä lisää algoritmin suoritusvaatimuksia.

Dual Contouring -menetelmä on piirteet säilyttävä. Reunojen olemassaolo tarkastetaan jokaisen kuution kohdalla, ja ylimääräisiä kärkiä lisätään malliin reunojen perusteella [5]. Kärkien sijainti ei ole rajoittunut vain kuutioiden särmille, kuten Marching Cubes -algoritmissa. Näin ollen Dual Contouring menetelmällä muodostetuissa samapinnoissa voi esiintyä teräviä reunoja, ilman laskostumisartefaktien syntymistä.

5.4 Reaalimaailmaan verrattavat pinnat

Dual Contouring -menetelmällä tuotetut pinnat ovat aina vesitiiviitä, eli niissä ei ilmene aukkoja. Tämä aiheuttaa ongelmia tilanteissa, joissa yhden solun (kuution) sisällä positiiviset ruudukon pisteet ovat vastakkaisilla puolilla. Dual Contouring -menetelmä muodostaa näissä tapauksissa niin sanotun tiimalasityyppisen pinnan. [8] Kaksiulotteinen kuvaus ongelmasta on esitetty kuvassa 9.



Kuva 9. *Alkuperäinen pinta on kuvassa värjätty keltaisella, mustat ääriviivat kuvaavat Dual Contouring -menetelmällä tuotettua samapintaa [8].*

Ongelma johtuu siitä, että Dual Contouring -menetelmä sallii ainoastaan yhden minimoijan jokaisessa solussa [8]. Kuvassa 9 minimoija on esitetty sinisellä.

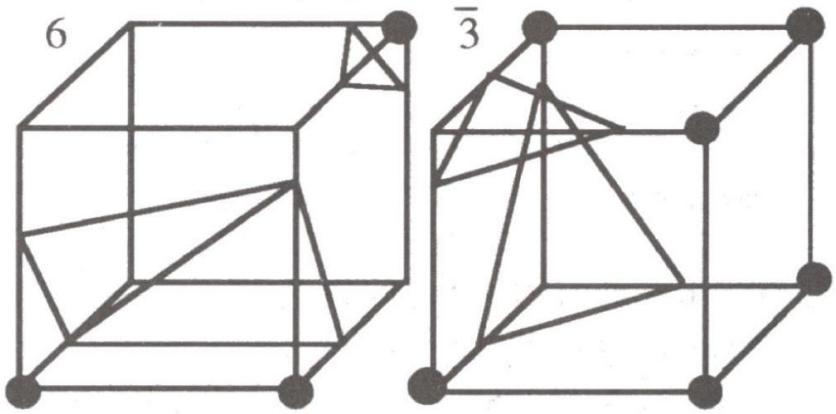
Tällainen samapinta on niin sanotusti ei-monisto (engl. non-manifold) ja se tarkoittaa, että useampi monikulmio jakaa keskenään saman reunan [9]. Tällaiset pinnat aiheuttavat ongelmia pinnan jatkokäsittelyssä, kuten pehmennyksessä ja parametrisoinnissa [8]. Ei-monisto on reaalimaailmassa mahdoton tilanne, mikä voi näin ollen myös rajoittaa Dual Contouring -menetelmällä tuotettujen pintojen käyttökohteita.

Marching Cubes -algoritmissa ei ilmene vastaavaa ongelmaa, sillä samapinnan kärjet muodostetaan aina solun reunoille hakutaulukon määräämällä tavalla. Menetelmä ei käytä kärkien sijoittamiseen minimoijafunktiota. Marching Cubes -algoritmilla tuotetut pinnat ovat aina monistoja (engl. manifold) [8].

5.5 Pintaelementtien yhtenäisyys

Pinnan yhtenäisyydellä tarkoitetaan sitä, että muodostettu monikulmiopinta on täysin vesitiivis. Ei-yhtenäinen pinta voi esimerkiksi tarkoittaa virheellisesti tuotettua halkeamaa pinnassa.

Alkuperäinen Marching Cubes -algoritmi voi tuottaa virheellisiä halkeamia muodostettuihin samapintoihin. Kyseessä on niin sanottu moniselitteisyysongelma (engl. ambiguity problem) ja se johtuu hakutaulukon rakenteesta. Lorensenin ja Clinen esittämä 14 tapauksen hakutaulukko ei ole riittävä kaikenlaisten pintojen muodostamiseen. [10] Yksi virheellinen kombinaatio on esitetty kuvassa 10.



Kuva 10. *Esimerkki tapauksista, jotka muodostavat virheellisen pinnan [10].*

Alkuperäisessä hakutaulukossa tapaukset 3, 6, 7, 10, 12 ja 13 (Kuva 4, s. 7) ovat tapauksia, joissa vastaava moniselitteisyysongelma voi ilmentyä [10]. Ongelman ratkaisuksi on esitetty vaihtoehtoisia hakutaulukoita, jotka sisältävät useampia tapauksia. Yleisesti käytetty ratkaisu on jättää pois komplementtisymmetria [1].

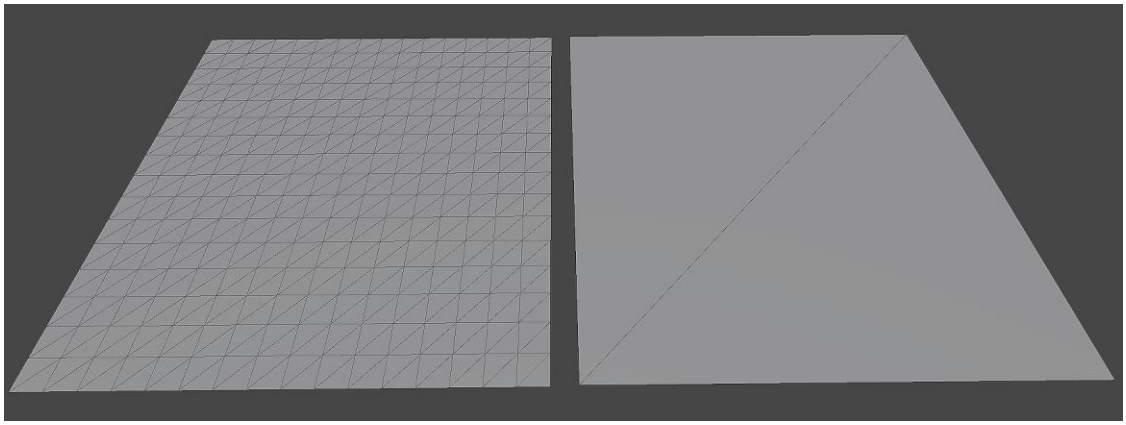
Luvussa 5.4 selvitettiin, kuinka Dual Contouring -menetelmä tuottaa aina vesitiiviitä pintoja, joissain tapauksessa virheellisesti. Menetelmän etuna on kuitenkin, että vastaavaa moniselitteisyysongelmaa ei ilmene Dual Contouring -menetelmällä tuotetuissa pinnoissa. Menetelmällä tuotetut pinnat ovat aina vesitiiviitä.

5.6 Samapinnan optimointi

Samapinnan optimoinnilla tarkoitetaan kärkien ja pintaelementtien määrän vähentämistä. Pintaelementtien määrän kasvu lisää muistinkulutusta sekä lisää pinnan visualisointiin tarvittavaa laskentaa. Toisaalta liian vähän monikulmioita sisältävä samapinta voi näyttää yksinkertaistetulta ja palikkamaiselta. Optimaalinen pinta on sellainen, jossa kärkiä ja monikulmioita on mahdollisimman vähän, mutta lopputulos näyttää virherajojen sisällä samalta, kuin alkuperäinen. [11]

Marching Cubes -algoritmin tuottaman samapinnan resoluutio määräytyy ainoastaan käytettävän kuutioruudukon perusteella. Pienentämällä kuutioiden kokoa samapinnan kärkien ja pintaelementtien määrä kasvaa. Jos samapinnan koko kasvaa suureksi tai kuutioiden koko on hyvin pieni, tuotettu samapinta voi sisältää liian monta monikulmiota. Tämä voi tehdä samapinnan visualisoinnista, tai jatkokäsittelystä mahdotonta [9].

Tasaisten pintojen esittämiseen ei tarvita paljon pintaelementtejä. Esimerkiksi neliönmuotoinen taso (engl. plane) voidaan tuottaa yhdellä, tai kahdella pintaelementillä. Tasaisten pintojen tapauksessa Marching Cubes tuottaa siis huomattavan määrän turhia pintaelementtejä, kuten kuvasta 11 voi huomata [12].



Kuva 11. *Marching Cubes -menetelmän tuottama tasainen pinta verrattuna optimaaliseen tasaiseen pintaan.*

Yhdenkokoista ruudukkoa on pidetty yhtenä Marching Cubes -algoritmin suurimpana heikkoutena. Pintojen optimointia varten on kehitetty menetelmiä, joilla pintaelementtien lukumäärää voidaan vähentää. Esimerkiksi Shekhar et al. [13] ovat esittäneet kasipuurakenteeseen perustuvan variaation, jonka tuottamat samapinnat sisältävät vähemmän monikulmioita. Nämä variaatiot kuitenkin sisältävät usein erillisiä reiänpaikkausmenetelmiä, mikä lisää algoritmin monimutkaisuutta.

Dual Contouring -menetelmä on adaptiivinen, joten se toimii yhdenkokoisen ruudukon lisäksi myös vaihtelevan kokoisessa ruudukossa. Muodostettujen pintojen monikulmioiden määrä ei siten ole yksinomaan riippuvainen kuutioruudukon koosta. Kasipuurakennetta hyödyntämällä ja neliöllistä virhefunktiota muokkaamalla tuotetut pinnat sisältävät huomattavasti vähemmän monikulmioita [5]. Tuotettujen monikulmioiden määrää voidaan pienentää edelleen kasvattamalla yksinkertaistuksen kynnysarvoa, kuten Kuva 7 (s. 11) on esitetty.

6. YHTEENVETO

Tässä kirjallisuuskatsauksessa tutkittiin Marching Cubes -menetelmää sekä Dual Contouring -menetelmää, ja näiden muodostamia samapintoja. Menetelmiä vertailtiin monelta kannalta, minkä jälkeen vastattiin siihen, millaisiin tilanteisiin kukin menetelmä on soveltuva. Taulukossa 1 on esitetty lista vaatimuksia ja niihin parhaiten vastaavat menetelmät.

Oikean menetelmän valinta riippuu paljon käyttötilanteesta. Reaaliaikaiseen mallinnukseen, esimerkiksi lääketieteessä, tai digitaalisessa kuvanveistossa, tarvitaan nopeasti toimivia menetelmiä. Suurien pintojen visualisoinnissa taas tietokoneen laskentakyky ja muisti ovat rajoittavia tekijöitä, ja samapinnan optimointi on tärkeässä asemassa. Vertailluista menetelmistä Dual Contouring on varmempi valinta vaihteleviin tapauksiin. Se soveltuu Marching Cubesin kanssa yhtäpitäviin tilanteisiin ja tuottaa yleensä paremman lopputuloksen.

Marching Cubes -menetelmä soveltuu parhaiten tilanteisiin, joissa reaaliaikainen laskentateho on kriittisessä asemassa, sillä ajonaikaiselta laskennalta vältytään hakutaulukon ansiosta. Lisäksi Marching Cubes -menetelmällä tuotetut pinnat ovat aina mahdollisia muodostaa myös fyysisesti. Menetelmä ei sovellu tilanteisiin, joissa tarkoituksena on tuottaa teräväreunaisia pintoja. Vakiokokoisen ruudukon seurauksena isot pinnat voivat myös sisältää liian paljon monikulmioita tietokoneen käsiteltäväksi. Marching Cubes -algoritmia käytettäessä on myös huomioitava alkuperäisen hakutaulukon puutteet ja tarvittaessa soveltaa päivitettyjä hakutaulukoita.

Dual Contouring -menetelmä soveltuu tilanteisiin, joissa pinnan piirteet on säilytettävä mahdollisimman tarkasti. Tällaisia ovat esimerkiksi teräväreunaiset kappaleet. Dual Contouring -menetelmällä tuotetut pinnat sisältävät usein myös vähemmän pintaelementtejä, joten ne käyttävät vähemmän tietokoneen resursseja. Dual Contouring -menetelmä saattaa tuottaa fyysisesti mahdottomia pintoja, mikä voi rajoittaa menetelmän käyttöä esimerkiksi 3D-tulostuksessa. Lisäksi menetelmä vaatii enemmän ajonaikaista laskentaa, mikä rajoittaa sen käyttöä nopeuskriittisissä tilanteissa, esimerkiksi tilavuudellisessa kuvanveistossa.

Taulukko 1: *Yhteenveto käyttökohteista ja niihin suositelluista menetelmistä.*

Vaatimus/käyttökohde	**Suositeltu menetelmä**
Nopeus	Marching Cubes
Piirteiden säilyvyys	Dual Contouring
Suuret pinnat	Dual Contouring
Vähän resursseja vievä lopputulos	Dual Contouring
Helppokäyttöisyys	Marching Cubes
Aina fyysisesti mahdolliset pinnat	Marching Cubes

Tässä työssä vertailtiin kummankin menetelmän alkuperäisiä versioita. On hyvä huomata, että kummastakin menetelmästä on olemassa uudempia variaatioita, jotka ratkaisevat alkuperäisissä ilmeneviä ongelmia. Esimerkiksi Marching Cubes -algoritmista johdettu laajennus, Extended Marching Cubes, kykenee tuottamaan teräväreunaisia pintoja. Vastaavasti Dual Contouring -menetelmästä on kehitetty variaatioita, jotka tuottavat aina fyysisesti mahdollisia pintoja. Variaatioiden haittapuolina voi taas olla algoritmien monimutkaistuminen, sekä lisääntynyt laskentatehon tarve.

LÄHTEET

[1] Wenger. (2013). Isosurfaces : geometry, topology, and algorithms. CRC Press. https://doi.org/10.1201/b15025

[2] Chen, C.-W., Hu, M.-C., Chu, W.-T., & Chen, J.-C. (2021). A Real-Time Sculpting and Terrain Generation System for Interactive Content Creation. IEEE Access, 9, 114914–114928. https://doi.org/10.1109/ACCESS.2021.3105417

[3] Kobbelt, Botsch, M., Schwanecke, U., & Seidel, H.-P. (2001). Feature sensitive surface extraction from volume data. International Conference on Computer Graphics and Interactive Techniques: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, 57–66. https://doi.org/10.1145/383259.383265

[4] Lorensen, & Cline, H. (1987). Marching cubes: A high resolution 3D surface construction algorithm. Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, 163–169. https://doi.org/10.1145/37401.37422

[5] Ju, Losasso, F., Schaefer, S., & Warren, J. (2002). Dual contouring of hermite data. ACM Transactions on Graphics, 21(3), 339–346. https://doi.org/10.1145/566654.566586

[6] Kelleghan, Michael (1997) Octree Partitioning Techniques, URL: https://www.gamedeveloper.com/programming/octree-partitioning-techniques (Haettu 24.11.2022)

[7] Schmitz, Dietrich, C. ., & Comba, J. L. . (2009). Efficient and High Quality Contouring of Isosurfaces on Uniform Grids. 2009 XXII Brazilian Symposium on Computer Graphics and Image Processing, 64–71. https://doi.org/10.1109/SIBGRAPI.2009.19

[8] Zhang, & Qian, J. (2012). Dual Contouring for domains with topology ambiguity. Computer Methods in Applied Mechanics and Engineering, 217-220, 34–45. https://doi.org/10.1016/j.cma.2012.01.004

[9] Schaefer, Ju, T., & Warren, J. (2007). Manifold Dual Contouring. IEEE Transactions on Visualization and Computer Graphics, 13(3), 610–619. https://doi.org/10.1109/TVCG.2007.1012

[10] Nielson, G. M., & Hamann, B. (1991). The Asymptotic Decider: Rosolving the Ambiguity in Marching Cubes. URL: https://escholarship.org/uc/item/17p025zk

[11] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., & Stuetzle, W. (1993). Mesh optimization. International Conference on Computer Graphics and Interactive Techniques: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, 19–26. https://doi.org/10.1145/166117.166119

[12] Rajon, & Bolch, W. E. (2003). Marching cube algorithm: review and trilinear interpolation adaptation for image-based dosimetric models. Computerized Medical Imaging and Graphics, 27(5), 411–435. https://doi.org/10.1016/S0895-6111(03)00032-6

[13] Shekhar, R., Fayyad, E., Yagel, R., & Cornhill, J. (1996). Octree-based decimation of marching cubes surfaces. Proceedings of Seventh Annual IEEE Visualization '96, 335–ff. https://doi.org/10.1109/VISUAL.1996.568127