

T-111.300 Tietokonegrafiikka
Harjoitustyö
1. Kamera

Petri Kalsi 55347A
pkalsi@cc.hut.fi

Miro Lahdenmäki 55089K
mlahdenm@cc.hut.fi

23. tammikuuta 2005

Sisältö

1	Johdanto	2
2	Ohjelma	2
3	Ympäristö	3
4	Algoritmit	3
5	Maailman kuvaus	4
6	Koodi	5
7	Kommentteja kurssista	5
	7.1 Petri Kalsi	5
	7.2 Miro Lahdenmäki	6

1 Johdanto

Ohjelman tarkoituksena on luoda voimakas syvyysvaikutelma. Ohjelma pyrkii mallintamaan kameran ominaisuuksia, tärkeimpinä zoomaus ja tarkennus. Kameraa voi liikutella maastossa näppäimistön ja hiiren avulla. Syvyysvaikutelmaa lisätään sumun avulla.

2 Ohjelma

Ohjelma ajetaan komennolla `./camera`. Parametreja ei tarvita. Käyttöliitymä toimii näppäimistöllä ja hiirellä. Taulukossa 1 on listattu ohjelmassa käytettävät näppäinkomennot.

Näppäin	Toiminto
z	zoomaa sisään (fov)
x	zoomaa ulos (fov)
c	contra-zoom lähemmäs
v	contra-zoom kauemmas
+	tarkenna kauemmas (focus)
-	tarkenna lähemmäs (focus)
w	käännä kameraa ylös
s	käännä kameraa alas
a	käännä kameraa vasemmalle
d	käännä kameraa oikealle
↑	liikuta kameraa eteenpäin
↓	liikuta kameraa taaksepäin
←	liikuta kameraa vasemmalle
→	liikuta kameraa oikealle

Taulukko 1: Näppäinkomennot

Hiirellä on kaksi toimintoa: vasen nappi pohjassa vetämällä voidaan kääntää kameraa. Hiiren oikealla napilla avataan menu, josta voi vaihtaa renderöinnin asetuksia. Eri efektit voi kytkeä päälle tai pois ja näin säätää myös ohjelman nopeutta. Taulukossa 2 on listattu menusta valittavat toiminnot.

Koska useiden iteraatioiden käyttö depth of field -efektissä on hidasta, kannattaa kameraa liikutellessa käyttää pienempää asetusta (esim 2 iteraatiota) kunnes sopiva katselusuunta ja sijainti on löydetty. Lopuksi kuvaa tarkemmin katseltaessa voidaan maisema renderöidä useammalla iteraatiolla.

Koska contra zoomissa myös kamera liikkuu, on kävelymoodi hyvä kytkeä pois ennen manuaalista contra zoomin käyttöä, jotta se toimisi oikein.

Valinta	Selitys
Normal	Ei depth-of-field -efektiä (focus)
Depth of field	Asettaa depth-of-field -efektin päälle
Depth of field Contra	Asettaa depth-of-field-contra -efektin päälle
Depth of field passes	Valitaan dof-efektin iteraatioiden lukumäärä
Enable / Disable focus point	Focus-etäisyyden ilmaiseva pallo päälle / pois
Enable / Disable fog	Kytetään sumu-efekti päälle / pois
Enable / Disable walk mode	Kytetään kävelymoodi päälle / pois

Taulukko 2: Right-click menu

3 Ympäristö

Ohjelmointiympäristönä on Sgi O2 ja glut3.7. Ohjelma kääntyy myös muilla alustoilla, joille glut3.7 tai freeglut on asennettu. Ohjelma käyttää kurssin tarjoamaa readtex-kirjastoa, joka käännetään samalla muun lähdekoodin kanssa.

Ohjelman lähdekoodi on saatavilla osoitteessa <http://iki.fi/baka/camera/>. Paketti puretaan ja ohjelma käännetään seuraavilla komennoilla:

```
% gunzip camera.tar.gz
% tar -xvf camera.tar
% cd camera
% gmake
```

4 Algoritmit

Tärkein ja näkyvin algoritmi ohjelmassa on depth-of-field -efektin tuottava algoritmi. Syvyystarkkuuden esittäminen polygonimallin kanssa on ongelmallista, sillä kuva koostuu oleellisesti terävistä kolmioista. Depth-of-field algoritmin tulisi säilyttää tietyllä etäisyydellä olevat objektit tarkkoina ja summentaa muilla etäisyyksillä olevia objekteja. Tämä voidaan toteuttaa yhdistämällä useampi kuva, joissa polttopisteen etäisyydellä olevat objektit ovat samassa kohdassa ruutua ja muilla etäisyyksillä olevia objekteja on poikkeutettu hieman. Perinteinen tapa tämän toteuttamiseen on liikuttaa katselukulmaa alikuvakohtaisesti.

Algoritmin pohja löytyi OpenGL Programming Guiden sivuilta 483-485, mutta joitakin optimointeja oli mahdollista tehdä. Algoritmista poistettiin accumulation bufferin nollaus alussa ja korvattiin ensimmäiseen `glAccum()`-kutsuun parametrin `GL_ACCUM` tilalle `GL_LOAD`, jolloin erillistä puskurin nollausta ei tarvita. Accumulation buffer on ohjelman hitain osa, joten yhdenkin

kutsun poistuminen auttaa paljon. Algoritmia muokattiin myös tukemaan iteraatioiden määrän vaihtoa ajon aikana.

Ohjelmassa kokeiltiin myös vaihtoehtoista depth-of-field -efektin toteutustapaa käyttämällä contra zoomia. Katselukulman poikkeuttamisen sijaan contra zoom -menetelmässä liikutetaan kameraa ja zoomataan samaan aikaan. Tämä tehdään siten, että tietyllä etäisyydellä oleva objekti pidetään ruudulla samankokoisena muiden objektien liikkua kohti keskustaa tai pois päin siitä. Alfred Hitchcock käytti menetelmää elokuvassaan Vertigo, ja sen myötä contra zoom tunnetaan myös Hitchcock-zoomina tai Vertigo-efektinä. Kaavan contra zoomille voi johtaa helposti perus-trigonometriasta. Depth-of-field:iä varten piirretään kuva muutamalla toisistaan vähän poikkeavalla contra zoom asetuksella ja kuvat lasketaan yhteen. Tällöin kuvan tarkennuspisteessä sijaitsevat objektit piirtyvät normaalisti muilla etäisyyksillä olevien objektien sumentuessa.

Puiden lajittelu ja blendaus tehtiin aiempien harjoituskierrosten perusteella. Puut piirretään aina samansuuntaisesti, ja niitä käännetään 180 astetta, mikäli kamera liikkuu puun toiselle puolelle. Tämän ansiosta voitiin ottaa display listat käyttöön myös puiden piirtoon.

Kumpuileva maasto tuotetaan yksinkertaisesti sin- ja cos-funktioiden yhdistelmänä. Tiedostossa land.c on funktio land_height(x,z), joka laskee yksittäisen pisteen korkeuden. Maastolle asetetaan ohjelman alussa halutut rajat leveydelle, pituudelle ja korkeudelle. Tämän jälkeen luodaan mesh, ottamalla sampleja maaston korkeudesta n*n ruudukolla (oletus 10*10 samplea). Myös maaston renderöinnissä käytetään display listaa. Tällä tavalla kameran liikkuminen maaston mukaan on yksinkertaista, funktio palauttaa tarkan arvon maan korkeudelle, eikä tarvitse selata itse meshiä.

Taivaan heikko sinisyys sumun takaa tehtiin piirtämällä kameran maksimikorkeuden yläpuolelle sininen polygoni, joka ulottuu leikkausetäisyyden ulkopuolelle.

5 Maailman kuvaus

Maalle ja puille ei käytetä valaistusta, vaan pelkkää teksturointia. Taivas piirretään myös ilman valaistusta, jotta väritys olisi tasaisen sininen. Kaikki muut objektit valaistaan yhdellä suuntaavalolla.

Sumun, taivaan ja maaston väritys valittiin kokeilemalla useita kirkkausvaihtoehtoja, jotta maisemasta saataisiin mahdollisimman realistinen.

Objektit ovat display listoja tai glutin perusobjekteja.

6 Koodi

Taulukossa 3 on listattu ohjelman lähdekooditiedostot ja niiden kuvaukset. Muut palautuspakettiin kuuluvat tiedostot on listattu taulukossa 4.

Tiedosto	Kuvaus
src/main.c	Pääohjelma, sisältää renderöinnin ja käyttöliittymän
src/const.h	Vakioita
src/focus.c	Frustum- ja perspektiiviprojektoiden asettaminen
src/focus.h	Otsikkotiedosto (jitter-arvot)
src/land.c	Maaston luominen ja renderöinti
src/land.h	Otsikkotiedosto
src/utils.c	Lineaarialgebraa ja muita yleisiä funktioita
src/utils.h	Otsikkotiedosto (makroja)
src/readtex.c	Kurssin tarjoama kirjasto tekstuurien käsittelylle
src/readtex.h	Otsikkotiedosto

Taulukko 3: Lähdekooditiedostot

Tiedosto	Kuvaus
README.TXT	Tiedostolistaus ja ohjeiden tiivistelmä
Makefile	Projektin makefile, käynnä ajamalla <code>gmake</code>
data/grass_tex.rgb	Nurmikon tekstuuuri (rgb)
data/tree_tex.rgb	Puiden tekstuuuri (rgba)
docs/raportti.tex	Raportti (LaTeX)
docs/raportti.ps	Raportti (Postscript)
docs/raportti.pdf	Raportti (PDF)

Taulukko 4: Lähdekooditiedostot

7 Kommentteja kurssista

7.1 Petri Kalsi

Harjoitustyön aihevalikoima oli riittävän laaja, kiinnostavia aiheita oli useita. Ryhmän koko on mielestäni sopiva, useamman jäsenen ryhmässä yhteydenpito tuottaa enemmän työtä, ja ylimääräisestä jäsenestä saatava etu on lähes mitätöntä.

Pikkuharjoitusten aihepiirit oli hyvin valittu, eikä yhtään kierrosta joutunut vääntämään ennalta-arvaamattoman pitkään. Ensimmäiset kierrokset tulivat

valmiiksi alle kymmenessä tunnissa, viimeiseen meni hieman yli kymmenen tuntia. Harjoitukset olisivat toisaalta voineet olla paremmin toisiinsa sidottu, niin että jokaisella kierroksella lisättäisiin selvemmin aiemmin toteutettuun koodiin uusia ominaisuuksia.

Suurin negatiivinen asia kurssilla oli käsittämätön hitaus tehtävien arvostelussa; on täysin mahdotonta oppia omista virheistään, jos harjoitusten palaute tulee vasta viimeisen kierroksen deadlineen jälkeen.

7.2 Miro Lahdenmäki

Kiinnostavia aiheita oli harjoitustyöhön tarjolla hyvin. Harjoitustyötä pystyi tekemään kätevästi myös kotikoneella, mikä on aina positiivista. Olisi ollut kiinnostavampi tehdä harjoituksia suoraan uudemmalle raudalle. Nyt ohjelman joutui tekemään käyttäen menetelmiä, jotka eivät ole kiihdytettyjä nykyraudalla. Täten työ tuntuu menevän osittain hukkaan.

Ilman aiempaa kokemusta grafiikkaohjelmoinnista ensimmäisiin harkkoihin meni huomattavan paljon aikaa. Loppuihinkin harjoituksiin kului runsaasti aikaa, eikä aika meinannut riittää muiden kurssien töiden rinnalla. Sinänsä harjoitukset olivat kyllä hyviä ja kattoivat laajan alueen.

Olen samaa mieltä siitä, että tehtävien arvostelun hitaus tekee omista virheistään oppimisen hyvin vaikeaksi.