



Aalto University
School of Science
and Technology

T-110.1100 Johdatus tietoliikenteeseen ja multimediatekniikkaan

Tietokonegrafiikka

Timo Tossavainen

Mediatekniikan laitos, Aalto-yliopiston perustieteiden korkeakoulu

Timo.Tossavainen@tkk.fi

25.3.2011

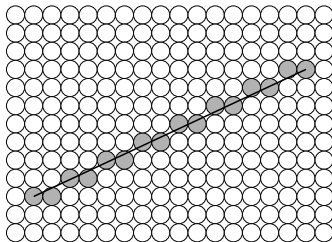
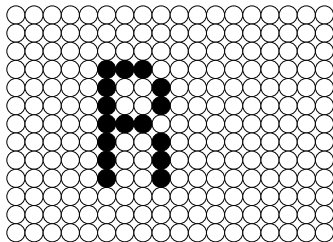
- ▶ Historiaa
- ▶ Rasterigrafiikkaa
- ▶ Vektorigrafiikkaa
- ▶ 3D-grafiikkaa

Historiaa

- 1940 Tietokoneet (Z3)
- 1950 CRT näyttö tietokoneessa (Whirlwind, SAGE)
- 1960 Vuorovaikutteinen grafiikka (Sketchpad)
- 1970 Grafiikan perusalgoritmit (mm. Utahin yliopisto)
- 1980 Fotorealistinen globaali valaistus
- 1990 Vuorovaikutteinen 3D-grafiikka yleistyy, OpenGL
- 2000 Ohjelmoitava kiihdytetty grafiikan liukuhihna

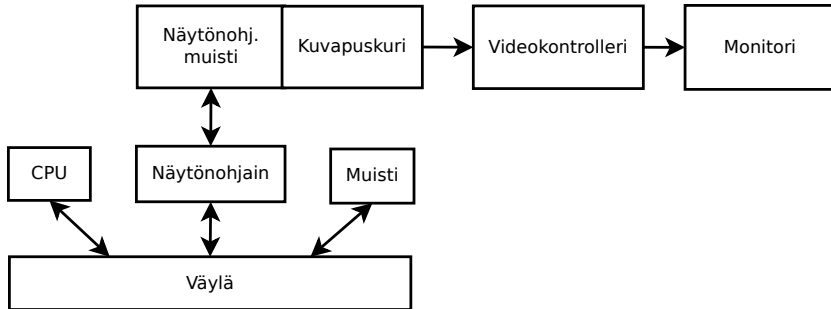
Rasterigrafiikkaa

- ▶ Digitaaliset kuvat koostuvat pikseleistä hilalla, rasterilla



- ▶ Pikselillä voi olla väri
- ▶ Tyypillinen esitys muistissa
 - ▶ Pikselit peräkkäin
 - ▶ Pikseleiden värikomponentit peräkkäin

Rasteriarkkitehtuuri



Rasteriesityksen ongelmia

- ▶ Rakenteeton eli kasa pikseleitä
- ▶ Tilantarve
- ▶ Heikko muokattavuus (zoom, kierto)
- ▶ Riippuu näytön tarkkuudesta

- ▶ Kuvataan piirrettävä kuva rakenteisesti “äärettömän” tarkkojen matemaattisten primitiivien avulla
- ▶ *Renderoidaan* eli muodostetaan kuva rasterille
- ▶ *Resoluutioriippumaton*: tarkkuus voi vaihdella
- ▶ Kuvan osat erikseen muokattavissa
- ▶ Tarvitaan sopivat *primitiivit* ja operaatiot
 - ▶ Muunnokset: siirto, skaalaus, kierto
 - ▶ Leikkaukset, etäisyydet
 - ▶ Rasterointi eli piirtäminen

Vektorigrafiikan muunnokset

- ▶ Siirtojen, kiertojen ja skaalausten yhdisteet esitettävissä matriiseilla, kun koordinaattivektoreiden loppuun lisätään 1.
- ▶ Esim. Siirto $(x, y) \mapsto (x + \Delta x, y + \Delta y)$ matriisimuodossa

$$\begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + \Delta x \\ y + \Delta y \\ 1 \end{bmatrix}$$

- ▶ Kuvausten yhdiste = matriisikertolasku.
- ▶ Käänteismatriisi = käänteiskuvauksen matriisi.
- ▶ Projektiiviset muunnokset esitettävissä homogeenisilla koordinaateilla operoivilla matriiseilla samaistamalla

$$(wx, wy, w), \quad w \neq 0,$$

pisteen (x, y) koordinaateiksi.

Vektorigrafiikan primitiivit

- ▶ Säilyttävä muunnoksissa. Projektiivisissa esim.
 - ▶ suora $\mapsto$ suora
 - ▶ kolmio $\mapsto$ kolmio
 - ▶ hypertaso $\mapsto$ hypertaso
 - ▶ kartioleikkaus $\mapsto$ kartioleikkaus
- ▶ Tehokas kuvata, leikata, piirtää
- ▶ Suora, puolisuora, jana, hypertaso, pallo, ellipsoidi, kolmio, monikulmio, kartioleikkaus
- ▶ Parametrinen esitys. Esim. Jana PQ

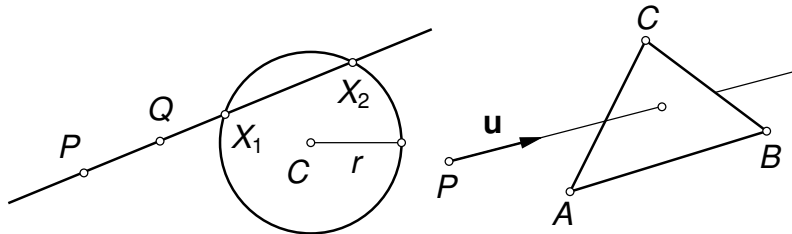
$$(1 - t)\mathbf{P} + t\mathbf{Q}, \quad 0 \leq t \leq 1$$

- ▶ Implisiittinen esitys. Esim. Ympyrä ja suora

$$x^2 + y^2 - r^2 = 0, \quad ax + by + c = 0.$$

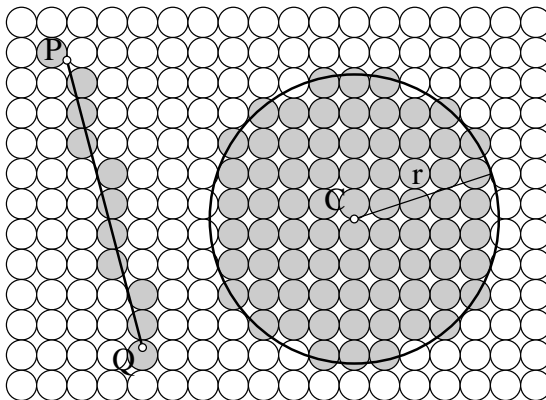
Geometrinen laskenta

- ▶ Piirtoalgoritmit rakentuvat primitiivien välisten geometristen laskelmien päälle
- ▶ Etäisyydet ja leikkaukset



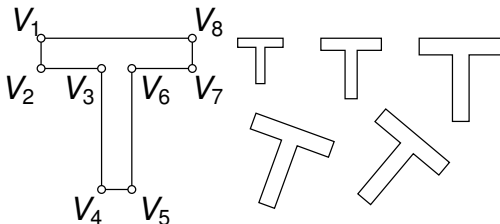
Rasterointi

- Piirretään approksimaatio geometrisesta primitiivistä.



Mallintaminen

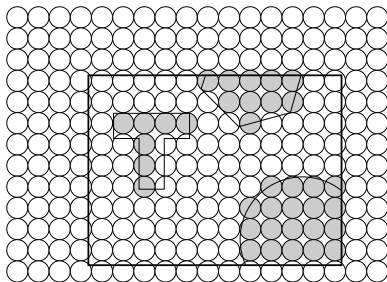
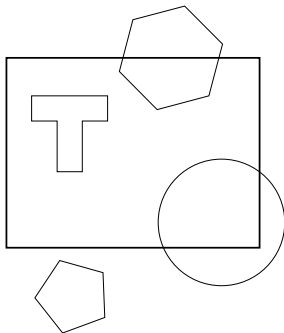
- ▶ Mallinnetaan piirros aiemmin käsitellyillä matemaattisilla primitiiveillä



- ▶ Muunnokset: Voidaan tuottaa kuva piirroksesta eri asennoissa, eri mittakaavoissa ja eri resoluutioilla.

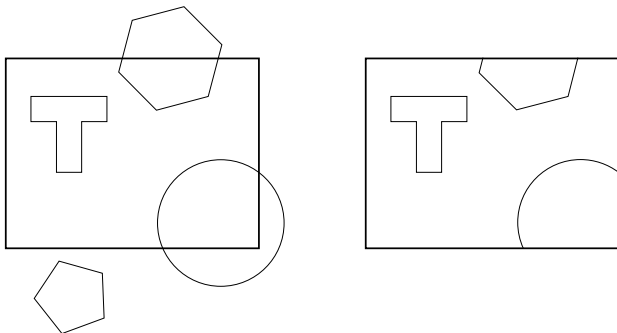
Katselu ja renderointi

- ▶ Piirrosta katsellaan ikkunan (window) läpi
- ▶ Ikkunasta näkyvä osuus piirretään näyttöalueeseen (viewport) esim. primitiivi kerrallaan

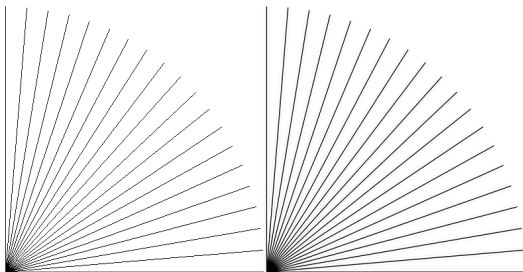


Leikkaaminen

- ▶ Näyttöalueen ulkopuolisia osia ei piirretä
- ▶ Hidasta testata pikseli kerrallaan (saksiminen)
- ▶ Yleensä primitiivit leikataan näyttöalueeseen ennen piirtoa



Antialiasointi

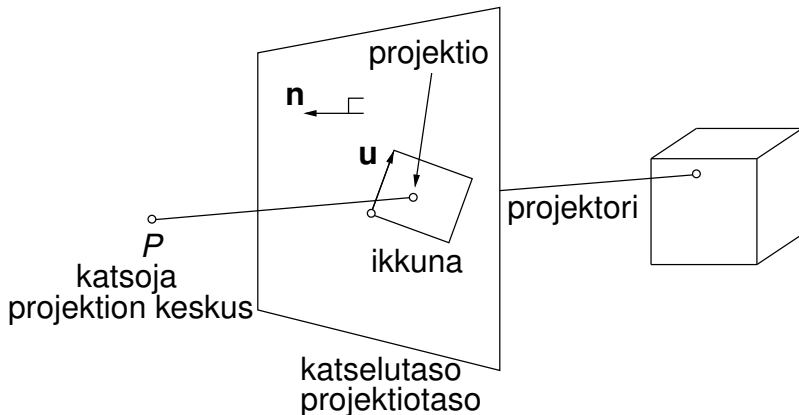


- ▶ Primitiivien määrittelemä ideaalikuva on äärettömän tarkka
- ▶ Pikselin värin poimiminen pistenäytteestä aiheuttaa laskostumista (aliasing)
- ▶ Optimaalinen: Alipäästösuodatetaan ideaalikuva
- ▶ Käytännössä: Piirretään korkeammalla resoluutiolla ja pienennetään tai käsitellään osittainen pikselin peitto heuristisesti

3D-vektorigrafiikka

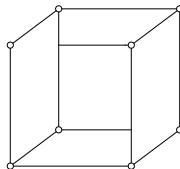
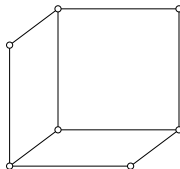
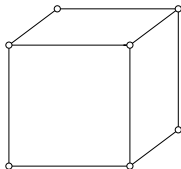
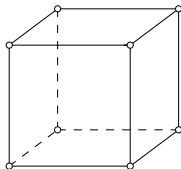
- ▶ Sama periaate kuin 2D-vektorigrafiikassa
- ▶ Primitiivit ja muunnokset kolmiulotteisia
- ▶ Tarvitaan projektio tasolle
 - ▶ Perspektiiviprojektio (realistinen)
 - ▶ Yhdensuuntaisprojektio (esim. tekniset piirrokset)
- ▶ Näkyvyysongelma
- ▶ Sävytys

Perspektiiviprojektio



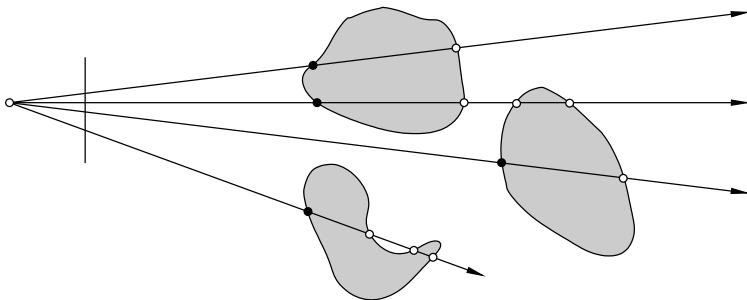
Näkyvyysongelma / Piilopintojen poisto

- ▶ Myöhemmin piirretyt peittävät aiemmin piirretyt.
- ▶ Kuution piirto sivu kerrallaan voi tuottaa ao. tulokset.



Säteenheitto

- ▶ Heitetään säde jokaisen kuvapisteen läpi
- ▶ Haetaan säteen leikkaukset primitiivien kanssa
- ▶ Lähin leikkauspiste on näkyvä pinnan piste

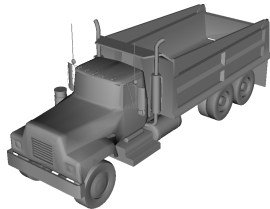
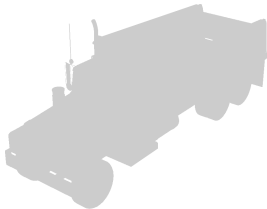
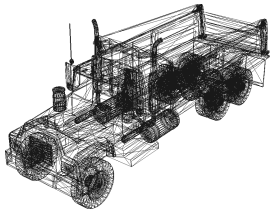


Syvyyspuskuri (Z-buffer / depth buffer)

- ▶ Pidetään kirjaa jokaiseen pikseliin piirretyn pinnan pisteen syvyydestä
- ▶ Piirretään pinnan piste, jos lähempänä kuin aiemmin piirretty
- ▶ Läpinäkymättömät primitiivit voidaan piirtää mielivaltaisessa järjestyksessä
- ▶ Nopea: Käytetyin tekniikka reaaliaikaisessa grafiikassa
- ▶ Tarkkuusongelmia, jos suuria mittakaavaeroja

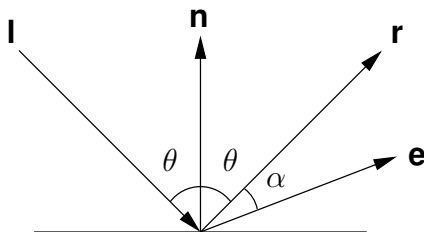
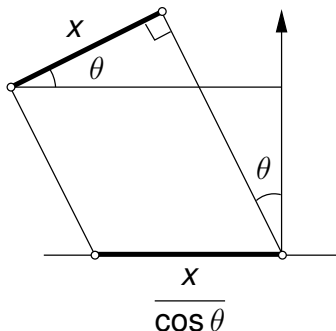
Sävytys (Shading)

- ▶ Pinnalta havaitun värin muodostaminen
- ▶ Valaistus, teksturointi, kuhmutus



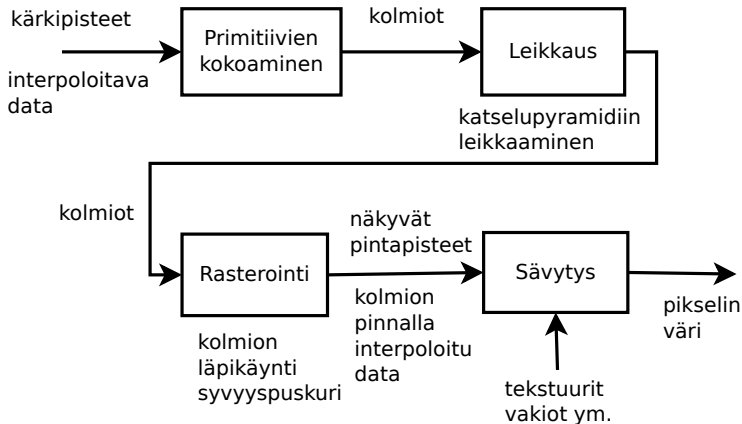
Paikallinen valaistus

- ▶ Phongin malli: taustavallo (ambient) + hajaheijastuva valo (diffuse, mattapinnat) + peiliheijastuva valo (specular, kiiltävät pinnat)
- ▶ Yksinkertaisia valonlähteitä: piste, suunta, spotti



Grafiikan liukuhihna (yksinkertaistettu)

- Reaaliaikaisessa grafiikassa käytetty laskennan malli

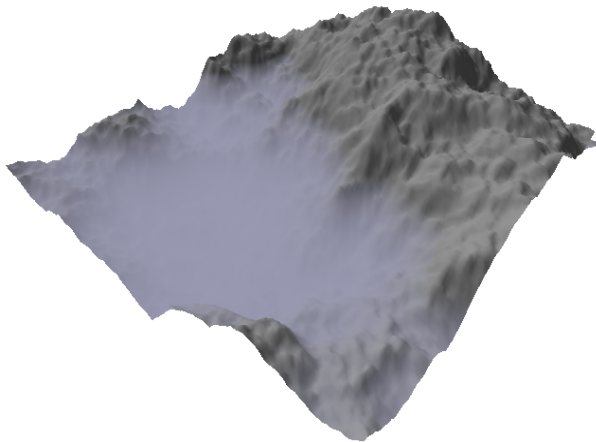


Ohjelmoitava sävytys

- ▶ Liitetään mallin kulmapisteisiin tietoa, joka interpoloidaan kolmioiden yli.
- ▶ Sävytysfunktio (pixel/fragment shader) laskee pinnan värin vakioista ja interpoloidusta tiedosta, mm.
 - ▶ Valonlähteiden paikat ja tyypit
 - ▶ Materiaalin valaistusominaisuudet
 - ▶ Tekstuurit ja tekstuurikoordinaatit
 - ▶ Pinnan hienorakenne (kuhmut)
- ▶ Joissakin tekniikoissa useita piirtokertoja (esim. varjot).

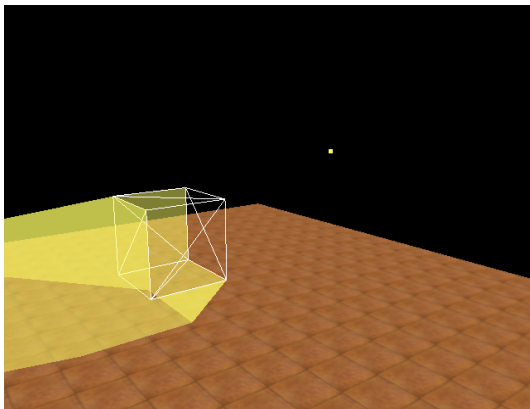
Esimerkki: Volumetrinen sumu

- ▶ Väri muuttuu valonsäteiden sumun sisällä kulkeman matkan mukaan

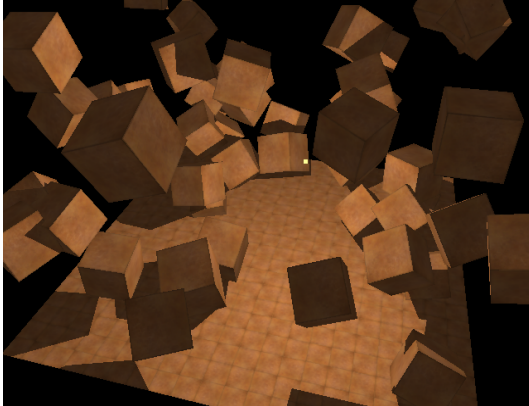


Esimerkki: Varjot

- Piste varjossa -testi piirtämällä varjotilavuudet



Esimerkki: Varjot (2)

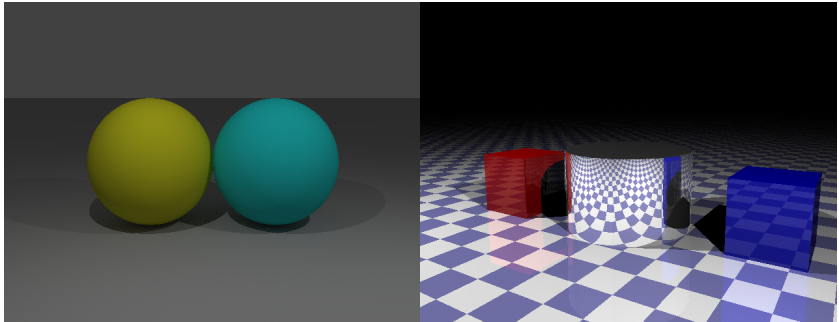


Grafiikkaliukuhinnan ominaisuuksia

- ▶ Piirtää primitiivin kerrallaan
- ▶ Ei tarvitse tietoa koko näkymästä
- ▶ Läpinäkymättömien primitiivien piirtojärjestys vapaa
- ▶ Laitteistokiihdytetty
- ▶ Suurelta osin ohjelmoitavissa
- ▶ Maiseman geometriaa tarvitsevien tekniikoiden toteutus hankalaa (heijastukset, varjot, globaali valaistus).

Fotorealistinen kuvanmuodostus

- ▶ Simuloidaan valon heijastumista maisemassa
- ▶ Tarvitsee tietoa koko maisemasta
- ▶ Laskennallisesti raskasta

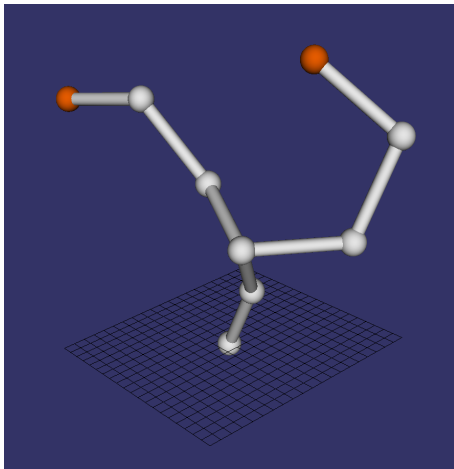


Tietokoneanimaatio

- ▶ Muutetaan piirrettävän maailman tilaa suhteessa aikaan
- ▶ Piirretään kuvia tietyillä ajanhetkillä
- ▶ Pieni muutos kuvien välillä saa aikaan liikehavainnon
- ▶ Reaaliaikainen 20+ kuvaa/s.
- ▶ Kaksoispuuskurointi: Näytetään yksi kuva, piirretään toinen, vaihdetaan kun piirto valmis
- ▶ Ongelman monimutkaisuuden takia tarvitaan työkaluja helpottamaan animaation määrittelyä.
- ▶ Liikkeet voivat olla suunniteltuja tai pohjautua fysiikkaan tai interaktioon.

Esimerkki: Käänteiskinematiikka

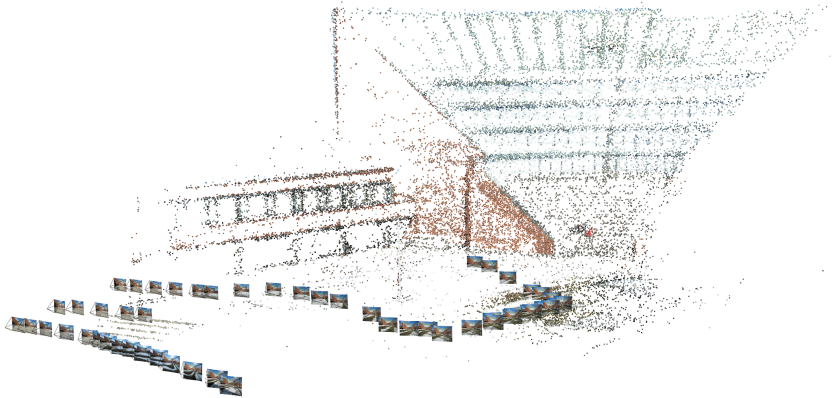
- Nivelrakenteen liikkeen laskeminen tavoitteiden perusteella



- Törmäykset, fysiikka, interaktio



Tutkimus: Kuvapohjainen mallinnus



Grafiikan kurssit

- ▶ Vuorovaikutteisen tietokonegrafiikan perusteet
- ▶ Vuorovaikutteisen tietokonegrafiikan jatkokurssi
- ▶ 3D-tuotanto
- ▶ Tietokoneanimaatio
- ▶ Tietokonegrafiikan seminaari
- ▶ Keinotodellisuus
- ▶ Erikoiskurssit/tutkimusseminaarit

A?

