

Virtuell verklighet

Grafik och Geometri

En föreläsning om
3D Computer Graphics

1999-11-01©Torbjörn Wiberg, UmU1

Virtuell verklighet

Geometrisk modellering och VR

- ✍ i en virtuell värld lever, rör sig och interagerar olika objekt
 - ✍ på egen hand - styrda av en simulator och
 - ✍ mer direkt - interaktivt - styrda av en användare som evbefinner sig i världen
- ✍ det behövs såväl geometriska som fysikaliska modeller av objekten
- ✍ en geometrisk modell beskriver objektens
 - ✍ form, utseende och placering i den virtuella världen
 - ✍ koppling till andra objekt i världen
 - ✍ möjliga geometriska rörelsemönster
 - ✍ övriga egenskaper av vikt för bl.a den fysiska modelleringen
- ✍ en fysisk modell baseras på kunskap om världens och objektens fysiska egenskaper som t.ex friktion, gravitation

1999-11-01©Torbjörn Wiberg, UmU2

Virtuell verklighet

3D Computer Graphics

- ✍ den här föreläsningen behandlar 3D-grafik - som ingår i avsnittet om geometrisk modellering
- ✍ kort sagt behandlar den konsekvensen för design och implementation av grafiken i VR av att **en eller flera observatörer** av VR-världen **rör sig** i den och **ser** den med **två ögon**
- ✍ kursen **datorgrafik o visualisering** är ett förkunskapskrav så föreläsningen ger endast ett VR-perspektiv av 3D-grafik
- ✍ se Foley, van Dam, Feiner, Hughes: Computer Graphics - Principles and Practices in C, second ed. eller, Foley, van Dam, Feiner, Hughes, Phillips: Introduction to Computer Graphics, second ed. för att fräscha upp kunskaperna i 3D-grafik

1999-11-01©Torbjörn Wiberg, UmU3

Virtuell verklighet

Den virtuella observatören - VO

- ✍ "De virtuella observatörerna vill se världen ur sitt perspektiv" - vad innebär det?
 - ✍ Det finns en absolut virtuell värld finns med objektens positioner och orientering uttryckta i absoluta världskoordinater.
 - ✍ Det finns ett antal VO som var och en från sin utgångspunkt och med sin orientering ser världen i sitt relativa perspektiv
 - ✍ Beskrivningen av de relativa perspektiven beräknas med information om den absoluta världens objekt, inklusive de virtuella observatörernas position och orientering
- ✍ VO har kanske två ögon och ser då världen i stereo - ögonavstånd, horisontella ögon
- ✍ VO ser bara delar av världen och olika delar i olika detalj - level of detail, field of view, siktdjup(near plane, projection plane o far plane)

1999-11-01©Torbjörn Wiberg, UmU4

Virtuell verklighet

Läge, referensram och djup

I den virtuella världen (VE) vill varje objekt se världen ur sitt perspektiv och t.ex. ange sina rörelser med lokal referensram, samtidigt som objektets läge i världen behöver kunna anges.

- ✍ flera koordinatsystem i VE
- ✍ tre fasta val dock:
 - ✍ högerorienterat - x:tumme, y:pek z:lång
 - ✍ positiv rotation = moturs, när man tittar från positiva delen av en axel mot origo
 - ✍ y uppåt, (VO tittar längs z-axeln mot 0
- ✍ transformationer mellan och rörelser i systemen kan uttryckas med
 - ✍ riktningsvinklar (direction cosines)
 - ✍ XYZ fasta vinklar
 - ✍ XYZ Euler-vinklar
 - ✍ quaternioner

1999-11-01©Torbjörn Wiberg, UmU5

Virtuell verklighet

Riktningvinklar - Direction cosines

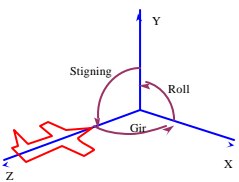
- ✍ observatören har i VEs referensram
 - ✍ en **position** (t_x, t_y, t_z) och
 - ✍ en **orientering** r_x, r_y och r_z av axlarna i VOs koordinatsystem r_x komponenter är t.ex $r_x = X_{v0}^T \cdot Z_{w0}$, dvs cosinus för vinklarna mellan VOs x-axel och VEs axlar
- ✍ transformationen från VEs koordinatsystem (w för world) till VOs bestäms av uttrycket till vänster, dvs först en translation och sedan en rotation
- ✍ Princip: VOs pos och orient. används omvänt för transformation till VOs koordinatsystem

$$\begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = \begin{bmatrix} r_{x1} & r_{x2} & r_{x3} & 0 \\ r_{y1} & r_{y2} & r_{y3} & 0 \\ r_{z1} & r_{z2} & r_{z3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & -t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

1999-11-01©Torbjörn Wiberg, UmU6

Virtuell verklighet

Rotationsvinklar



- Rotationsvinklarna har namn:
 - roll
 - pitch - stigning
 - yaw - gir
- Flyginspirerat: sett ur perspektivet att ett plan flyger längs positiva z-axeln mot origo, där x är den andra horisontal-axeln och y den vertikala, så rollar man runt z-axeln, stiger runt x-axeln och girar runt y-axeln
- Matematiskt säger man att en roll är en rotation i xy-planet
- Efter en rotation hur anges nästa? Sitter man i eller tittar man på?

1999-11-01 ©Torbjörn Wiberg, UmU 7

Virtuell verklighet

Fasta XYZ-vinklar - tittar på

- fasta XYZ-vinklar?
 - separata rotationer i en fast referensram
 - öst är öst och väst är väst ... - en observatör som ser på världen från en fast punkt
 - VOs orientering i VE anges som tre rotationer i ordningen roll, stigning och gir och dess position i VE anges som en translation
 - dvs [translate][yaw][pitch][roll]
- VOs värld ges av den inversa transformationen
 - $p_{vo} = [-roll][-pitch][-yaw][-translate](p_w)$
- när det gäller rotationer står -tecknet för vinkeln med motsatt tecken och för translationer translationsvektorn med motsatt tecken
- homogena koordinater används

1999-11-01 ©Torbjörn Wiberg, UmU 8

Virtuell verklighet

Euler-XYZ-vinklar - sitter i

- Euler-XYZ-vinklar?
 - rotationsvinklarna anges relativt en roterande XYZ-referensram
 - utgår från flygarens lokala orientering
- Teorem: Sekvens av roll, stigning och gir med fasta vinklar är ekvivalent med motsvarande sekvens i omvänd ordning av roll, stigning och gir med Euler-vinklar
 - Ex: fast roll 90 följt av fast gir 180 är ekvivalent med Euler-gir 180 följt av Euler-roll 90
- VO i VE anges med
 - [translate][yaw][pitch][roll] - OBS! i Euler-vinklar
- Transformationer till VOs referensram blir då:
 - $p_{vo} = [-yaw][-pitch][-roll][-translate](p_w)$

1999-11-01 ©Torbjörn Wiberg, UmU 9

Virtuell verklighet

Quaternions - Kvaternioner

- Hamilton 1800-talet
 - utvecklades för att beskriva rotationer kring en godt. axel
- en kvaternion är reell kvadrupel
 - $q = [s, v] = (s, x, y, z) = s + xi + yj + zk$
 - $i^2 = j^2 = k^2 = -1$
 - generalisering av komplexa tal
 - $q_1 = q_2$ om komponenterna lika
 - $q_1 * q_2$ def som \pm av komp multiplikation? $i^2 = j^2 = k^2 = -1$ betrakta ijikjk
 - för mult av två i den ordning de står så är resultatet komponenten närmast till höger, t.ex ksj
 - för mult av t.ex ik, dvs två i omvänd ordning så -komp till höger, dvs ik=-j
- värdet av $q_1 * q_2$ ges av dessa regler
 - inversen $q^{-1} = [s, -x, -y, -z] / |q|^2$
 - $|q|$ beräknas på vanligt sätt
- en punkt P(x,y,z) representeras i kvaternionform som $P = [0 + xi + yj + zk]$
- en rotation? kring axeln u, av längd ett, definieras av kvaternionen $q = [\cos(?/2), \sin(?/2) u]$
- rotationen av en punkt P beräknas som $P' = qPq^{-1}$
- Observera att boken använder u för att beteckna olika saker här!

1999-11-01 ©Torbjörn Wiberg, UmU 10

Virtuell verklighet

Kvaternioner - forts

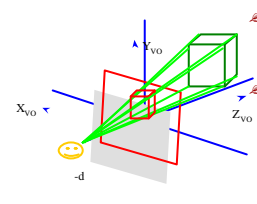
- Hur hanterar vi då en VO med hjälp av kvaternioner?
 - roll, stigning och gir
 - $q_{roll} = [\cos roll/2, \sin roll/2 e_3]$
 - osv
 - den sammantagna effekten av en roll, stigning och gir i fasta vinklar ges av
 - $q = q_{yaw} q_{pitch} q_{roll}$ där
 - $s = c y/2 * c p/2 * c r/2 + s y/2 * s p/2 * s r/2$
 - $x = c y/2 * s p/2 * c r/2 - s y/2 * c p/2 * s r/2$, osv
 - mönstret ges av att x,y och z hör ihop med respektive gir, stigning och roll
 - i Euler-vinklar är kvaternionen som representerar den sammantagna effekten av en roll, stigning och gir
 - $q = q_{roll} q_{pitch} q_{yaw}$
 - en kvaternion kan representeras av en matris

$$M = \begin{bmatrix} 1-2(y^2+z^2) & 2(xy-sz) & 2(xz+sy) \\ 2(xy+sz) & 1-2(x^2+z^2) & 2(yz-sx) \\ 2(xz-sy) & 2(yz+sx) & 1-2(x^2+y^2) \end{bmatrix}$$
 - givet att VOs orientering anges av en "kvaternion" M så orienterar M^T om VE till VO
 - om VOs pos ej origo så krävs även en translation som tidigare

1999-11-01 ©Torbjörn Wiberg, UmU 11

Virtuell verklighet

Perspektivprojektion

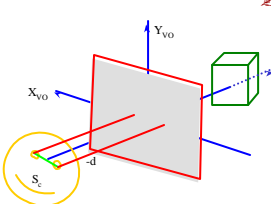


- VO ser världen ur sitt perspektiv
 - VO står i origo
 - tittar ut längs positiva z-axeln
 - Inför ett projektningsplan genom att placera VO en bit -d - ut på negativa z
 - att låta xy-planet vara projektningsplan
 - perspektivet beror av avståndet från VO till objektet
 - storleken på den projicerade bilden beror på d - stort d ger stor bild
 - d styr field of view
 - för realistiska bilder bör fov i bilden överensstämma med fov i presentationsutrustningen

1999-11-01 ©Torbjörn Wiberg, UmU 12

Virtuell verklighet

Stereoperspektiv



- ✂ VO ser världen ur två perspektiv
- ✂ större avstånd mellan ögonen ger större skillnad mellan bilderna, förstärkning av djupet i bilden
- ✂ stereoeffekten förutsätter att vi kan få ögonen att konvergera med samma vinkel som i projektionen, annars 2 bilder
- ✂ enklast att anta konvergensvinkel 0 och förskjuta bilderna så att man upplever en bild
- ✂ egentligen bör presentationen känna av ögonens konvinkel och räkna om persp.proj.
- ✂ ögonen parallella med xaxeln

1999-11-01 ©Torbjörn Wiberg, UmU 13

Virtuell verklighet

Enkel 3D-modellering

- ✂ En 3D-modell är konstruerad av primitiva komponenter
 - ✂ solida byggstenar som rätblock cylindrar ... eller
 - ✂ polygonytor i rummet, t.ex trianglar eller
 - ✂ svepta profiler t.ex
- ✂ Ljus, och texturering mm baseras på att attribut ges till dessa komponenter och att man kan göra beräkningar utifrån dessa attribut och t.ex deras ytnormaler för att realisera olika ljus- och skuggmodeller
- ✂ Eliminering av dolda ytor med t.ex
 - ✂ målralgoritmen
 - ✂ z-bufferalgoritmen

1999-11-01 ©Torbjörn Wiberg, UmU 14

Virtuell verklighet

Realism

- ✂ Vi vill ha en VR-applikation med bildväxlingar i realtid och med en realism som inte förhindrar en upplevelse av en virtuell värld som vi interagerar med och rör oss i.
- ✂ Fusk med visst för att ge ökad realism i annat
- ✂ Man kan öka realismen i bilden genom att
 - ✂ ta med inslag av verkliga världar i bilderna (fototexturapetsering)
 - ✂ begränsa störningar och fel som introduceras vid renderingen (anti-aliasing)
- ✂ Man kan fuska genom att
 - ✂ begränsa världens utsträckning (sikt djup o fov)
 - ✂ variera precisionen i renderingen (detaljeringsgrad, ljus o skuggor)
 - ✂ använda modellschabloner (texturapetsering, skugg- och reflexschabloner, bump mapping, ljus o skuggor)

1999-11-01 ©Torbjörn Wiberg, UmU 15