Informatica Instituut, Faculteit Wiskunde en Informatica, UU.
In elektronische vorm beschikbaar gemaakt door de TEBC van A−Eskwadraat.
Het college INFOGR werd in 2003/2004 gegeven door René van Oostrum.

Graphics (INFOGR)
11 november 2003

Opgave 1. Scan-line Rasterization

a) In hoofdstuk 12 van het bij het college gebruikte boek hebben we gezien hoe we geprojecteerde polygonen efficiënt kunnen tekenen met behulp van het *scan-line* algoritme. Wat is in dit verband *scan-line coherence*, en hoe maakt het scan-line algoritme hier gebruik van?

b) Het scan-line algoritme maakt gebruik van de *Edge Table* en de *Active Edge Table*. Neem aan dat we een polygon projecteren waarvan één van de edges als eindpunten de punten $(2, 10)$ en $(5, 4)$ heeft.

1. Welke informatie (d.w.z. welke getallen, en wat stellen ze voor) wordt er van bovenstaande edge opgeslagen in de Edge Table?
2. In welke entry van de Edge Table komt bovenstaande edge terecht?
3. Hoe ziet de entry voor bovenstaande edge in de Active Edge Table eruit om het moment dat we de scanline $y = 8$ aan het tekenen zijn?

Opgave 2. Het z-buffer Algoritme

a) Leg de principes achter het z-buffer algoritme uit (daarbij hoef je niet in te gaan op een efficiënte implementatie m.b.v. een variant op het scan-line algoritme), en noem het grote voordeel van het z-buffer algoritme ten opzichte van de andere projectiemethode(n) die we eerder in het college hebben besproken.

b) Wanneer we een variant op het scan-line algoritme gebruiken om het z-buffer algoritme te implementeren, in welk(e) coördinatenstelsel(s)/ruimte(n) kunnen we dan werken?

1. In *World Coordinates*
2. In *Camera Coordinates*
3. In het *Canonical Perspective Frame*
4. In *Canonical Projection Space*
5. In alle bovenstaande ruimtes
6. In geen van de bovenstaande ruimtes

Opgave 3. Texture Mapping

a) Bij texture mapping zal het dikwijls voorkomen dat een een pixel wordt gemapt naar een gebied in de texture dat meerdere texels bevat. De kleur van de pixel moet dan worden bepaald aan de hand van de kleur van de texels in dat gebied; dit wordt *filtering* genoemd. Beschrijf twee filtering-methoden die in het bij het college gebruikte boek zijn beschreven.

b) Wat is *mipmapping*?

c) Stel dat we een cilinder hebben gemodelleerd als een getrianguleerd polyhedron, waarbij alle driehoeken min of meer hetzelfde formaat hebben. We willen het zijvlak (d.w.z.: het gekromde oppervlak) van de cilinder texturen met een tegelpatroon van witte en zwarte

vierkantjes. We kunnen daarbij texture mapping uitvoeren voor elke driehoek afzonderlijk, maar dat geeft niet al te fraaie resultaten. Wat is hier het probleem, en hoe kunnen we dit oplossen?

Opgave 4. Shadows

Eén van de methoden om bij real-time projectietechnieken alsnog schaduwen te introduceren is de zogenaamde *fake-shadow* methode. Daarbij projecteren we, na de scene te hebben gerenderd, alle objecten (of een selectie van belangrijke objecten) op een relevant vlak. Wanneer we bijvoorbeeld schaduwen op een tafel willen hebben, projecteren we op het vlak waarin het tafelblad ligt. Met gebruikmaking van de z-buffer kunnen we de geprojecteerde objecten vervolgens in donkergrijs tekenen, maar dan verliezen we de texture van het onderliggende tafelblad. Fraaier is het om te *blenden*, door bijvoorbeeld de kleurwaarden van de pixels waarop de schaduw-polygonen worden geprojecteerd 50% omlaag te brengen.

a) In het college hebben we twee problemen met deze methode besproken (ze kwamen ook aan de orde in het *stencil buffer tutorial* van NVIDIA). Noem deze twee problemen.

b) Leg uit hoe de problemen waarnaar bij opgave a. wordt gevraagd, kunnen worden opgelost met gebruikmaking van de stencil buffer.

Opgave 5. Radiosity

a) Vergelijk *radiosity* met *recursive ray tracing*, en ga daarbij in op:

   1. lokaal vs. globaal
   2. view-dependant vs. view-independant
   3. de soorten reflecties waar beide methoden rekening mee houden.

b) Leg uit wat *progressive refinement* is.