

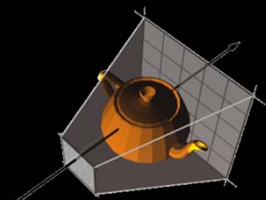


Računarska grafika

predavanja
v.prof.dr. Samir Lemeš
slemes@unze.ba

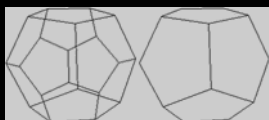
24. Uklanjanje nevidljivih površina

- Uklanjanje nevidljivih površina
- Odbacivanje pozadine
- Isijecanje iz vidokrug
- Algoritam slikara
- Warnockov algoritam
- Ray-casting algoritam
- z-Buffer algoritam



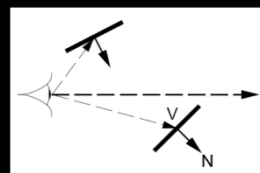
Uklanjanje nevidljivih površina

- "Hidden Surface Removal"
- U 3D grafici, to je proces kojim se određuje koje površine i dijelovi površina nisu vidljivi iz određene tačke posmatranja.
- Faze uklanjanja:
 - Odbacivanje pozadine (*Backface culling*)
 - Isijecanje iz vidokrug (*Viewing frustum culling*)
 - Odbacivanje zaklonjenih dijelova (*Occlusion culling*)
 - Odbacivanje predalekih objekata (*Contribution culling*)



Odbacivanje pozadine

- Eliminiraju se svi poligoni koji su okrenuti suprotno od oka posmatrača.
- Pozadinski poligon je onaj čija je normala usmjerena od posmatrača.
- Znači, ako je $N \cdot V > 0$, onda se taj poligon uklanja.
- V je vektor od oka do proizvoljne tačke na poligonu

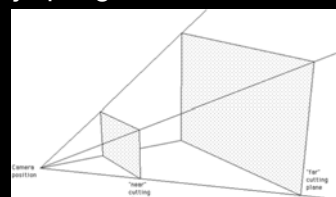


Odbacivanje pozadine

- Nedostaci:
 - Ako objekti nisu konveksni, potrebne su dodatne operacije za utvrđivanje vidljivosti.
 - Ako su poligoni dvostrani (odnosno ako nisu zatvoreni u volumen), onda je ova metoda neprimjenjiva.
- Prednosti:
 - Ako se može koristiti, postiže se znatno ubrzanje (min. 1/2 poligona se odbacuje).
 - Obično se kombinuje s drugim metodama.
 - Lako se integriše u hardver.

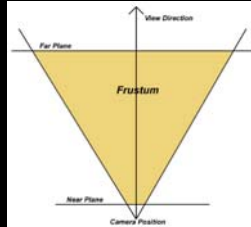
Odbacivanje pozadine

- Obično ga automatski vrši API za rasterizaciju (Direct3D, OpenGL) i očekuje se da se broj poligona smanji za oko 1/2 od broja poligona unutar vidokrug (frustum)



Isijecanje iz vidokruga

- Vidokrug je zarubljena piramida koja obuhvaća objekat koji se prikazuje.
- Ograničena sa 6 ravnina.
- Za svaku od 6 ravnina izvodi se jednačina:
 $A \cdot x + B \cdot y + C \cdot z + D = 0$

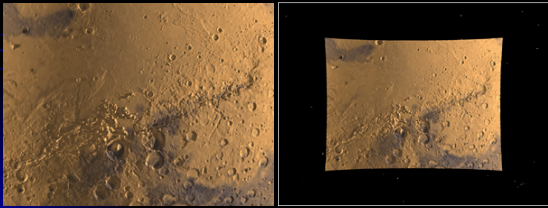


Isijecanje iz vidokruga

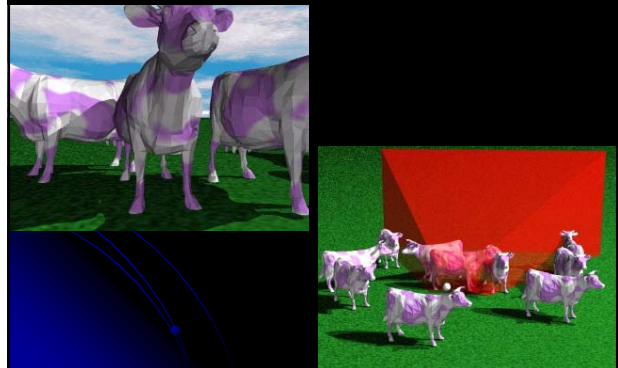
- Udaljenost proizvoljne tačke (px,py,pz) od granične ravni vidokruga:
 $d = A \cdot px + B \cdot py + C \cdot pz + D$
- Za svaku tačku (ili verteks) objekta koji se posmatra može se utvrditi udaljenost do svake od 6 ravnina.
- Primjer (sfera): Ako se centar sfere uvrsti u prethodnu jednačinu, poredi se radijus sfere sa svih 6 udaljenosti.
- Ako je jedna od tih udaljenosti veća od radijusa, znači da sfera ne može cijela stati u vidokrug i potrebno je isijecanje.

Isijecanje iz vidokruga

- Pomjeranje kamere unazad pokazuje da je geometrija (cijele planete) isječena da bi se uklonili svi poligoni koji su izvan vidokruga.

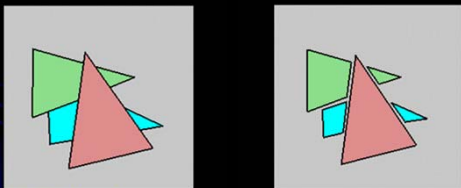


Isijecanje iz vidokruga



Algoritam slikara

- Kad se neki poligoni preklapaju, treba odrediti koji dio kojeg poligona je vidljiv.



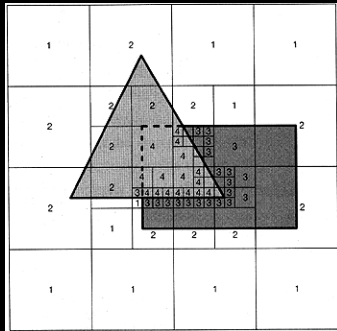
Algoritam slikara

- *Painter's algorithm*, sortiranje po dubini, dobilo je ime po načinu na koji slikari nanose boje na platno.
- Prvo se slikaju nebo i zemlja, zatim najdalji objekti, itd. Na taj način slojevi boje prekrivaju jedan drugog.
- U 3D grafici, prvo se objekti sortiraju po udaljenosti od tačke posmatranja, a zatim se crtaju tim redoslijedom.



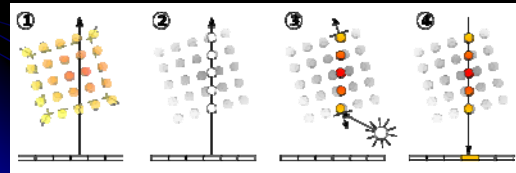
Warnockov algoritam

- Rekurzivni algoritam za podjelu površine.
- Rekurzivne podjele površine sve do veličine jednog piksela.



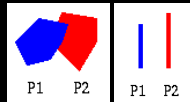
Ray-casting algoritam

- Samo konceptualna upotreba, bez praktične primjene: propušta zraku od tačke posmatranja kroz svaki piksel da bi se našla površina najbliža posmatraču.

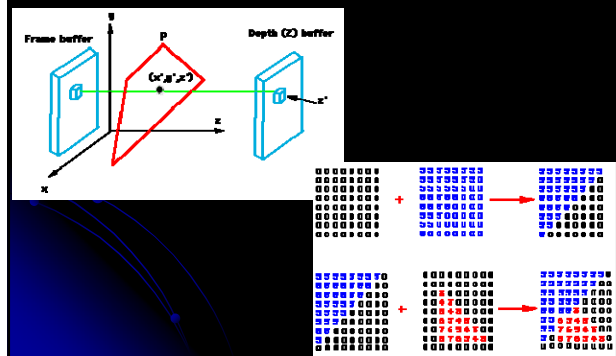


z-Buffer algoritam

- z-Buffer algoritam je najčešće korištena programska rutina za uklanjanje nevidljivih površina. Jednostavan je, lako se implementira i često se nalazi u hardveru.
- Ideja: dodijeli z-vrijednost svakom poligonu, a zatim prikaži samo one piksele koji imaju najmanju vrijednost.



z-Buffer algoritam



Zagonetka

- Na koliko načina se torus može presjeći 1 ravinom da se dobiju dva kruga?



Je li ovo isijecanje OK?

