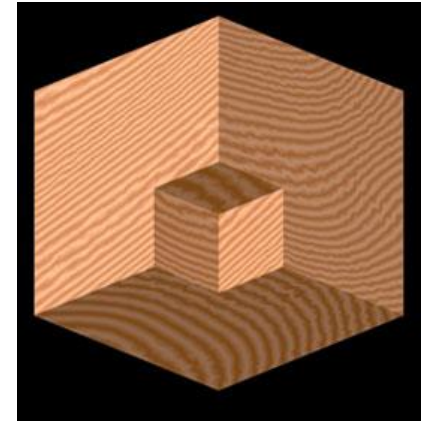


8. Modeli i postupci osvjetljavanja, sjenčanje, sjene

PODJELA MODELA OSVJETLJAVANJA I PRIPADNIH POSTUPAKA SJENČANJA



EMPIRIJSKI MODELI

- temelje se na iskustvu i estetskim aproksimacijama
- npr. jednostavni model osvjetljavanja (lokalni)
- u postupcima sjenčanja koji ga koriste Gouraud '71 i Phong '75.

PRELAZNI MODELI

- uključuju prozirnost, zrcaljenje i sjene, geometrijski egzaktan
- npr. Whittedov model '80.
(RTX- Nvidia Technology - OptiX API, DirectX Raytracing (DXR))
- koristi se u postupku sjenčanja pretrage zrake (engl. ray tracing)

ANALITIČKI MODELI

- temelji se na energetskej ravnoteži
- npr. model Cook-Torrancea '82 (PBR-Physically Based Rendering) i Goralov model '84.
- koristi se u postupku isijavanja (engl. radiosity method)

HIBRIDNI MODELI

- integriraju postupak pretrage zrake i postupak isijavanja
- npr. Wallace ' 87.

Prema modelu osvjetljavanja u pojedinoj točki računa se osvjetljenje

- interakcija svjetla i površine
- složena priroda svjetlosti i površine (različito reflektiranje raznih valnih duljina, ista valna duljina različito reflektirana ovisno o kutu)

Sjenčanje (engl. shading) dio je postupka izrade prikaza (engl. rendering).

Izvori svjetla

- točkasti, linijski, poligonalni, reflektori (usmjereni). <https://playcanv.as/p/zdkARz26/>

Materijali - površine

- hrapave (mat), glatke (sjajne), prozirne

Svjetlost

- stvaranje modela kako bi mogli ostvariti simulaciju (propagacija svjetlosti i interakcija s površinom)
 - geometrijska optika – skup svjetlosnih zraka – bez mase i dimenzije
refleksija, refrakcija (glatke plohe – zrcalna komp., hrapave - difuzna)
odsjaj jednog objekta u drugom ili samo boje objekta
 - fizikalna optika – elektromagnetsko zračenje – širenje vala
difrakcija (ogib svjetlosti), interferencija (npr. dugine boje na CD),
polarizacija svjetlosti (transverzalni val)
 - kvantna fizika – skup čestica tj. fotona – ‘paketići’ energije
sudar fotona s elektronom i prijelaz u višu energetska razinu – boja,
ispodpovršinsko raspršenje
- različiti modeli koji aproksimiraju ponašanje svjetla

8.1. EMPIRIJSKI MODEL OSVIJETLJENJA (Phongov model)

TRI KOMPONENTE

- ambijentna
- difuzna
- zrcalna

AMBIJENTNA KOMPONENTA

$$I = k_a I_a$$

- k_a koeficijent reflektirane ambijentne svjetlosti (ovisi o materijalu)
- I_a intenzitet ambijentne svjetlosti

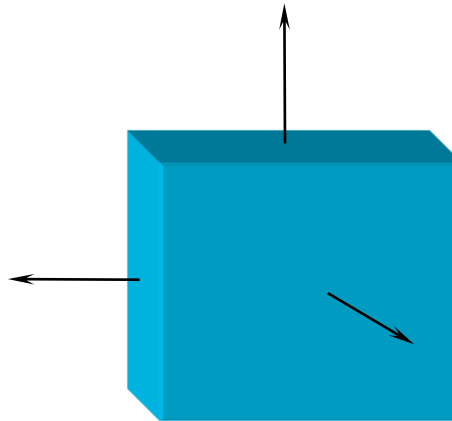
Reflektiranje svih poligona okolnog prostora. Aproksimacija globalnog osvjetljenja. Ukoliko ova komponenta nije prisutna stražnji poligoni obzirom na izvor svjetla bit će crni. <http://www.realtimerendering.com/udacity/?load=demo/unit3-lambert-demo.js>

DIFUZNA KOMPONENTA (engl. diffuse reflection)

LAMBERTOV ZAKON

Djelić površine dS rasvijetljen je proporcionalno kosinusu kuta pod kojim snop svjetlosti upada na površinu.

- <https://webglfundamentals.org/webgl/lessons/resources/directional-lighting.html>
- <https://webglfundamentals.org/webgl/lessons/resources/point-lighting.html>



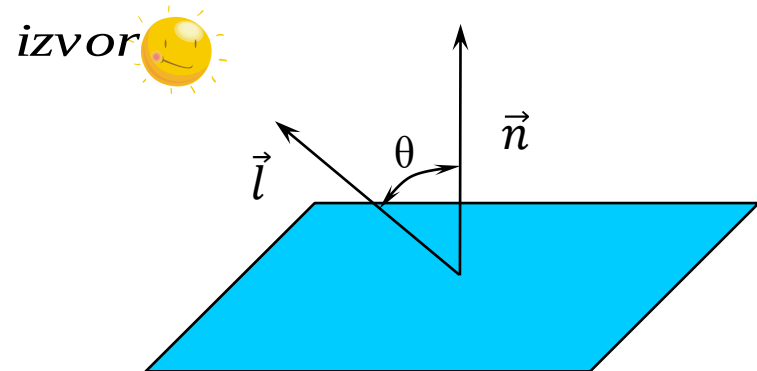
Količina svjetlosti koju vidi promatrač **neovisna** o položaju promatrača i proporcionalna je kosinusu kuta između normale na površinu i vektora prema izvoru.

LIGHTPOSITION

$$I_d = k_d I_P \cos \theta, \quad 0 \leq k_d \leq 1, \quad 0 \leq \theta \leq 90^\circ$$

- k_d koeficijent difuzne refleksije
- I_P intenzitet točkastog izvora
- θ kut između normale na površinu i vektora prema izvoru

$$I_d = k_d I_P \max(0, \vec{l} \cdot \vec{n})$$



- $\vec{l}$ normirani vektor prema izvoru svjetlosti
- $\vec{n}$ normirani vektor normale na površinu

- <https://webglfundamentals.org/webgl/lessons/resources/normals.html>
- <https://webglfundamentals.org/webgl/webgl-3d-lighting-point.html>

U praksi moramo uzeti u obzir i udaljenost izvora do površine d_L .

$$I = k_a I_a + k_d I_p \frac{\vec{l} \cdot \vec{n}}{d_q^2}$$

Bolje rezultate dobit ćemo formulom

$$I = k_a I_a + k_d I_p \frac{\vec{l} \cdot \vec{n}}{k_c + k_l d_L + k_q d_q^2}$$

- konstante k_c = konstanta (da ne dobijemo dijeljenje s nulom)
 k_l = podešavanje linearne komponente
 k_q = kvadratna komponenta

U slučaju potpunog modela boja, komponente RGB se posebno računaju:

$$I = k_a O_{dR} I_{aR} + k_d O_{dR} I_{pR} \frac{\vec{l} \cdot \vec{n}}{k_c + k_l d_L + k_q d_q^2}$$

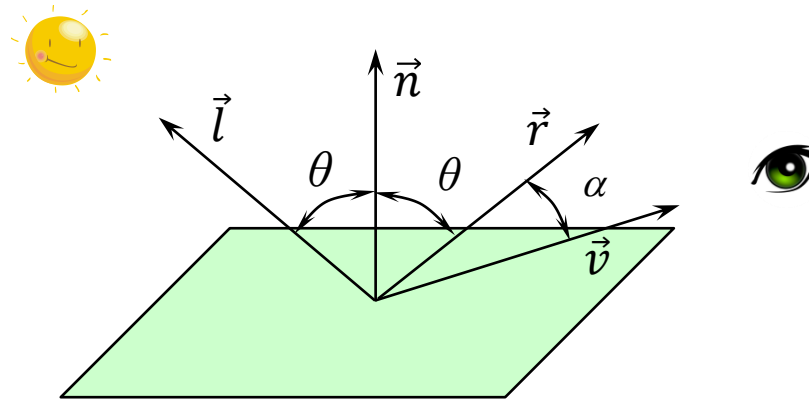
- O_{dR} definira komponentu boje objekta

<http://www.realtimerendering.com/udacity/?load=demo/unit3-lambert-demo.js>

ZRCALNA KOMPONENTA (engl. specular reflection)

Zrcalna refleksija nastaje na blještavoj površini.

Blještavilo (engl. highlight) obojeno je bojom upadne svjetlosti.



$$I_s = k_s I_p \frac{\cos^n \alpha}{d_L + k} = k_s I_p \frac{(\vec{r} \cdot \vec{v})^n}{d_L + k}$$

- $\vec{r}$ vektor usmjeren u pravcu reflektirane zrake r - reflektirana zraka
- $\vec{v}$ vektor prema očistu
- α kut između reflektirane zrake i zrake prema očistu
- k_s koeficijent zrcalne refleksije
- n određuje prostornu razdiobu i vezan je uz materijal

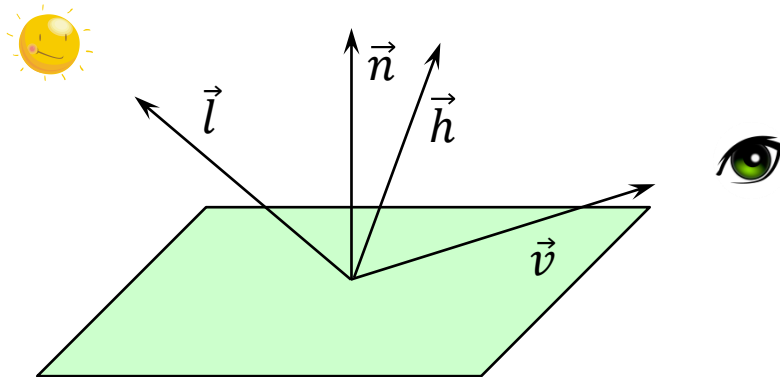
<https://webglfundamentals.org/webgl/webgl-3d-lighting-point-specular-power.html>

u praksi se često koristi vektor h , odnosno produkt hn umjesto vektora r , odnosno produkta rv

- jednostavno se računa

$$\vec{h} = \frac{\vec{l} + \vec{v}}{2}$$

- ukoliko pretpostavimo da su promatrač i izvor vrlo daleko tj. vektore možemo smatrati konstantnima, konačna jednažba ovisit će samo o normali na površinu



$$I_s = k_s I_p \frac{(\vec{h} \cdot \vec{n})^n}{d_L + k}$$

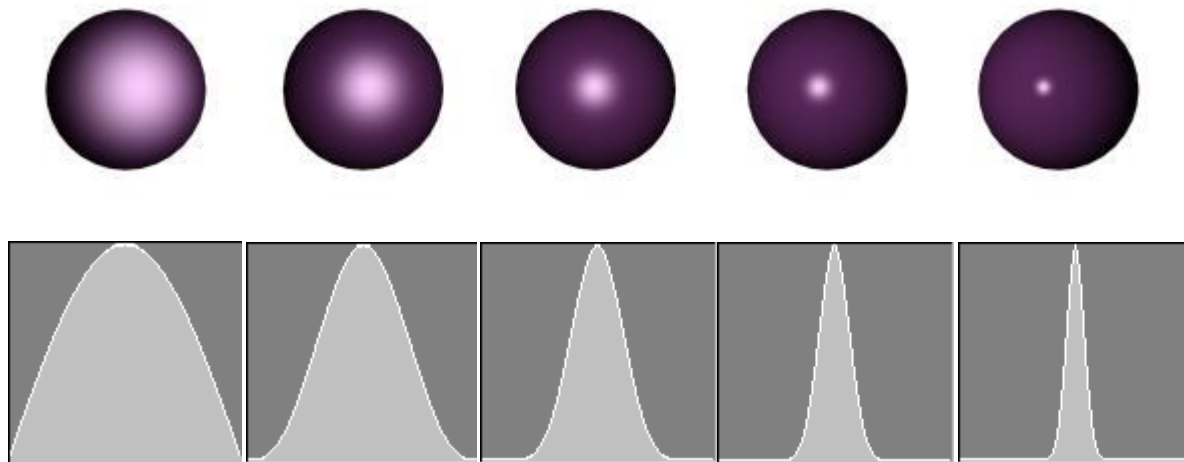
<https://webglfundamentals.org/webgl/lessons/resources/specular-lighting.html>

Za realne blještave površine imamo nepotpunu zrcalnu refleksiju.
(Phong, 1975)

$$I_s = k_s I_p \frac{\cos^n \alpha}{d_L + k} = k_s I_p \frac{(\vec{r} \cdot \vec{v})^n}{d_L + k}$$

Funkcija $\cos^n \alpha$:

LIGHTMATERIAL



$n = 1$

$n = 15$

$n = 30$

$n = 45$

$n = 60$

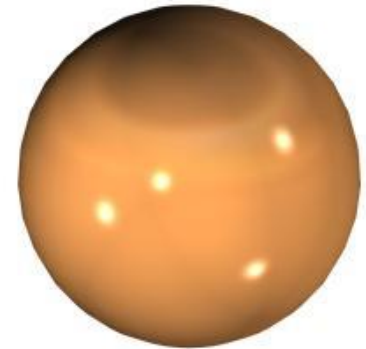
PLASTIKA



METAL

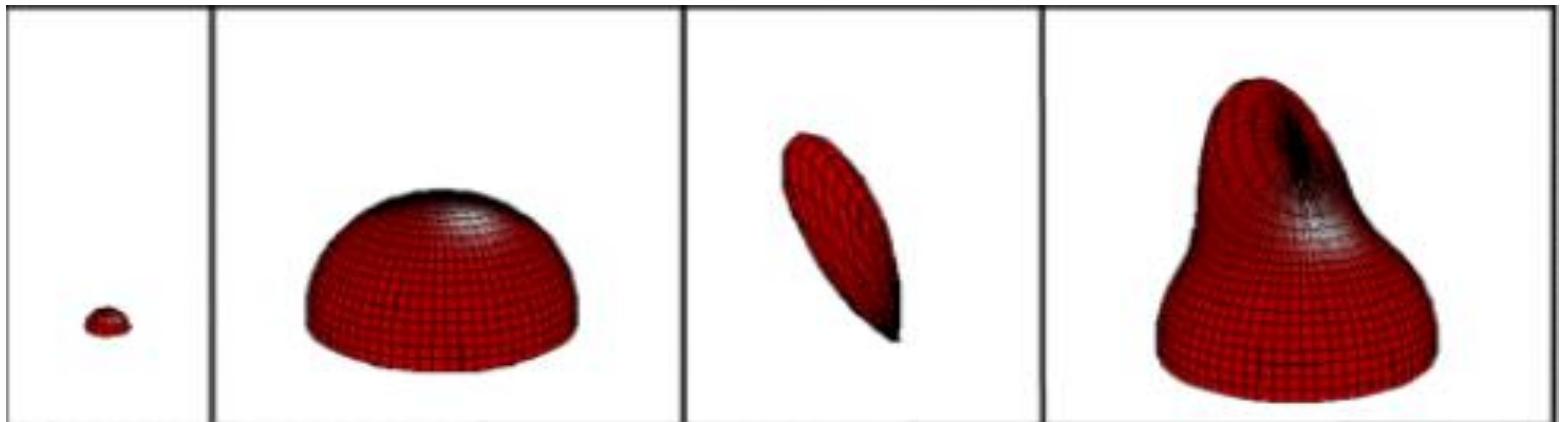
Opća jednađžba proračuna osvjetljenja za slučaj m izvora:

$$I = k_a I_a + k_d \sum_{j=1}^m I_{p,j} \frac{\vec{l}_j \cdot \vec{n}}{d_{L,j} + k} + k_s \sum_{j=1}^m I_{p,j} \frac{(\vec{r}_j \cdot \vec{v})^n}{d_{L,j} + k}$$



Sjenčanje u boji (primjer komponente crvene boje):

$$I = k_a O_{dR} I_a + k_d O_{dR} \sum_{j=1}^m I_{p,j} \frac{\vec{l}_j \cdot \vec{n}}{d_{L,j} + k} + k_s O_{dR} \sum_{j=1}^m I_{p,j} \frac{(\vec{r}_j \cdot \vec{v})^n}{d_{L,j} + k}$$



ambijentna

difuzna

zrcalna

ukupno

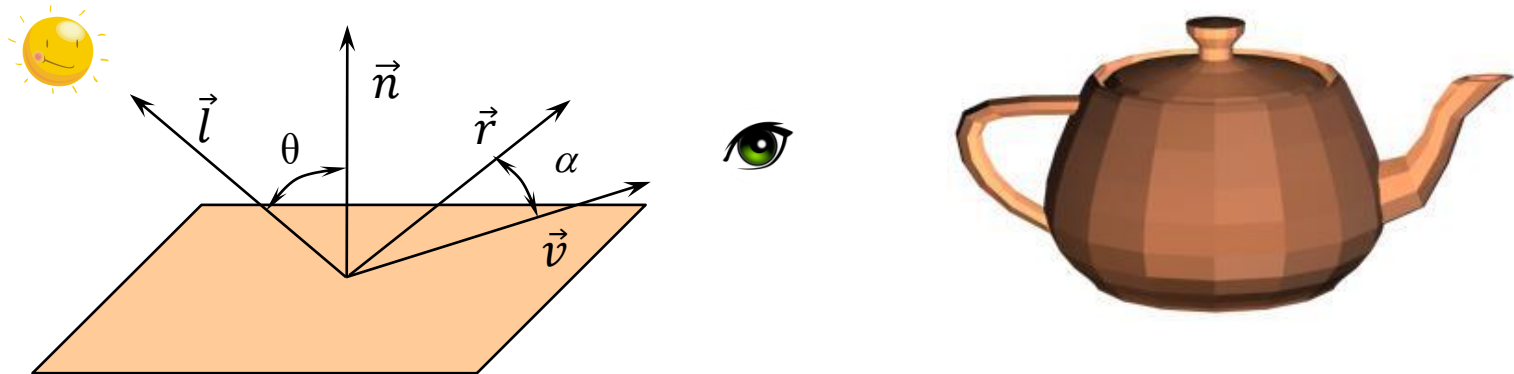
POSTUPCI SJENČANJA

Konstantno sjenčanje

Sve točke površine unutar jednog poligona imaju isti intenzitet.

To znači da je za neki poligon

$$\vec{l} \cdot \vec{n} = konst. \quad \vec{r} \cdot \vec{v} = konst.$$



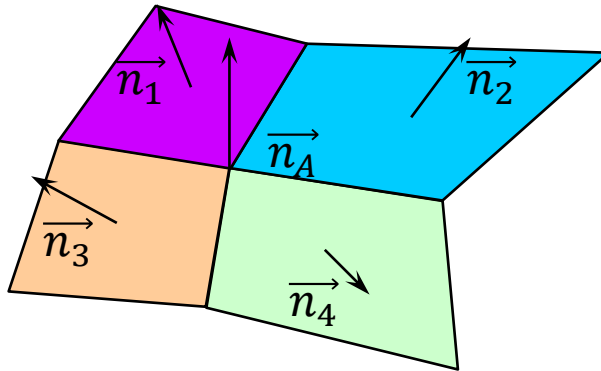
<http://neilwallis.com/projects/java/early3d>

http://www.realtimerendering.com/erich/udacity/exercises/unit3_specular_demo.html

acii sjenčanje: https://threejs.org/examples/#webgl_effects_ascii

određivanje normala u vrhovima:

- određivanje normala svih poligonskih površina
- aritmetička sredina normala poligona incidentnih s vrhom



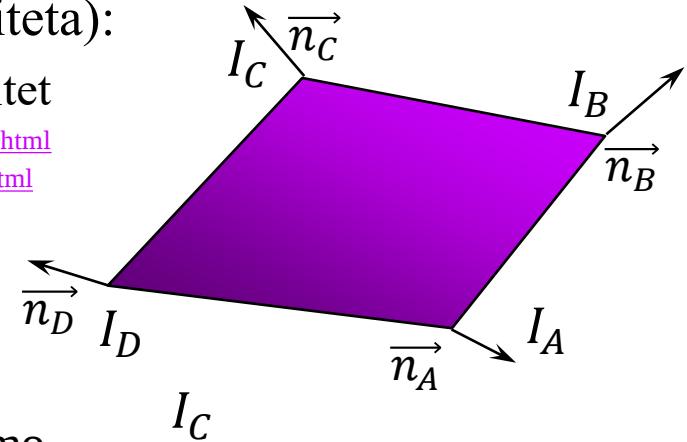
$$\vec{n}_A = \frac{\vec{n}_1 + \vec{n}_2 + \vec{n}_3 + \vec{n}_4}{4}$$

Gouraudovo sjenčanje (interpolacija intenziteta):

- u svakom vrhu poligona računa se intenzitet

<https://www.khronos.org/registry/webgl/sdk/demos/webkit/TeapotPerVertex.html>

<https://www.khronos.org/registry/webgl/sdk/demos/webkit/TeapotPerPixel.html>



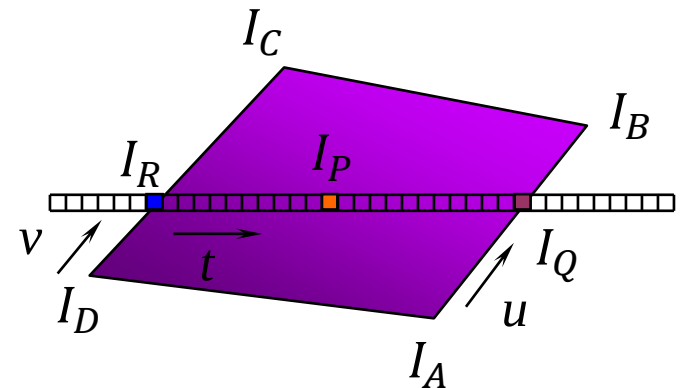
- dobivene intenzitete linearno interpoliramo
Sjecišta ispitne linije s bridovima neka su Q, R
interpoliranje duž bridova

$$I_Q = (1-u)I_A + uI_B, \quad u = \frac{\overline{AQ}}{\overline{AB}}$$

$$I_R = (1-v)I_D + vI_C, \quad v = \frac{\overline{DR}}{\overline{DC}}$$

interpoliranje unutar poligona

$$I_P = (1-t)I_R + tI_Q, \quad t = \frac{\overline{RP}}{\overline{RQ}}$$



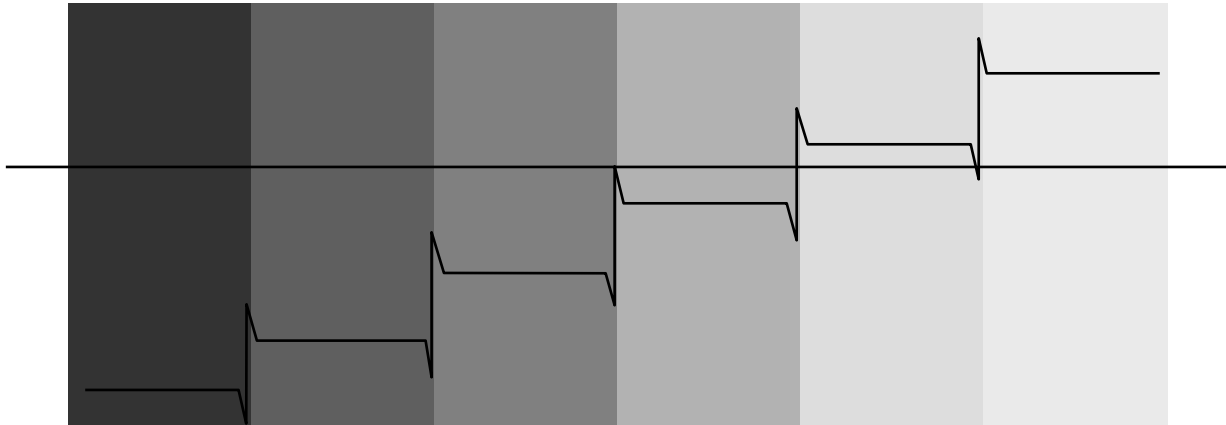
- inkrementalni oblik

$$I_{P1} = (1 - t_1)I_R + t_1 I_Q, \quad (1)$$

$$I_{P2} = (1 - t_2)I_R + t_2 I_Q, \quad (2)$$

$$I_{P2} = I_{P1} + (I_Q - I_R)(t_2 - t_1) = I_{P1} + \Delta I \Delta t \quad (2) - (1)$$

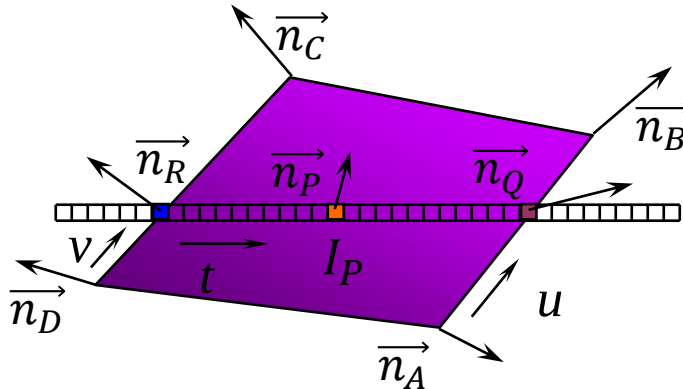
- nedostatak postupka je Machov vizualni učinak (svijetle i tamne pruge)



Phongovo sjenčanje (interpolacija normala):

- izračunamo normale u vrhovima
- interpoliramo normale duž bridova
- interpoliramo dobivene normale duž ispitne linije
- intenzitet izračunamo na kraju

<http://www.realtimerendering.com/teapot/>



$$\overrightarrow{n_Q} = (1 - u)\overrightarrow{n_A} + u \overrightarrow{n_B}, \quad u = \frac{\overline{AQ}}{\overline{AB}}$$

$$\overrightarrow{n_R} = (1 - v)\overrightarrow{n_D} + v\overrightarrow{n_C}, \quad v = \frac{\overline{DR}}{\overline{DC}}$$

$$\overrightarrow{n_P} = (1 - t)\overrightarrow{n_R} + t\overrightarrow{n_Q}, \quad t = \frac{\overline{PR}}{\overline{QR}}$$

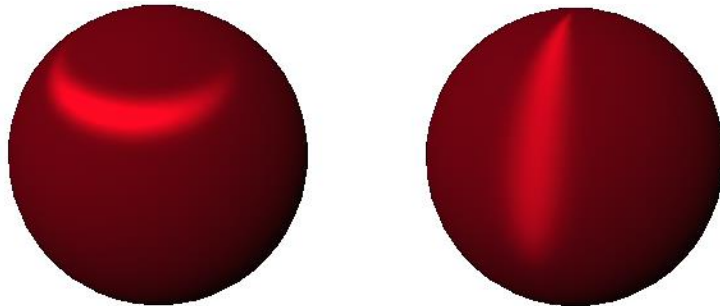
Inkrementalni oblik:

$$\overrightarrow{n_{P2}} = \overrightarrow{n_{P1}} + (\overrightarrow{n_Q} - \overrightarrow{n_R})(t_2 - t_1)$$

- Phongovo sjenčanje je bolje od sjenčanja Gouraud
no sporije je (za svaki slikovni element treba odrediti v , l , r , I)
i umanjuje Machov vizualni učinak



anizotropni učinak



Nisu obuhvaćene specifičnosti nekih površina

- anizotropnost: http://www.realtimerendering.com/erich/udacity/exercises/unit3_anisotropy_solution.html
https://threejs.org/examples/#webgl_loader_gltf_anisotropy

- prozirnost <https://tympanus.net/Tutorials/GlassEffect>

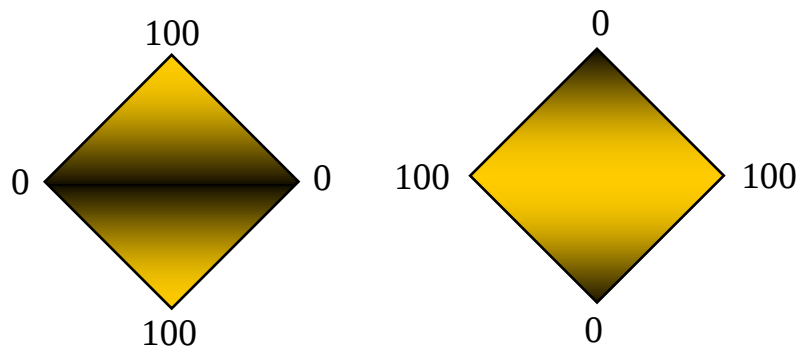
Problemi u postupku sjenčanja

- silueta objekta - broj poligona

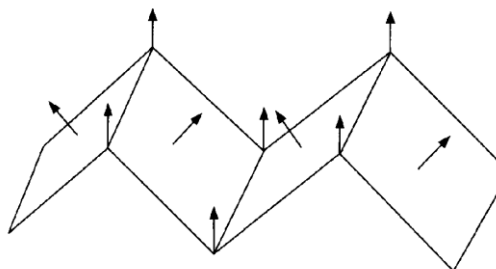
<http://www.realtimerendering.com/udacity/?load=demo/unit3-tessellation-demo.js>



- rotaciona ovisnost

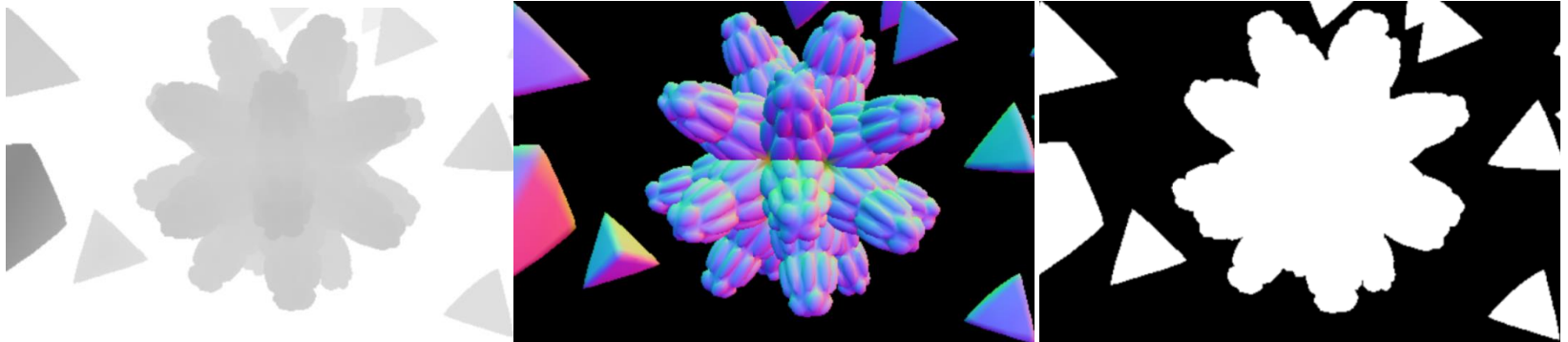


- normale u vrhovima



Odgođeno sjenčanje (engl. *deferred shading*)

- krajnje izračune želimo provoditi samo za elemente koji se u konačnici vide zato provodimo nekoliko prolaza
- u prvom prolazu sakupljaju se podaci koji su potrebni za prikaz tako da se puni G-spremnik (*G-buffer*, spremnik geometrije) – pozicija, normala, materijal u koordinatnom sustavu ekrana <http://marcinignac.com/blog/deferred-rendering-explained/demo/>



- za prikaz u G-spremnik se koristi prikaz u tekstu primjenom tehnike MRT (engl. *Multiple Render Targets*)
- u daljnjem prolazu se radi izračun konačnog osvjetljenja
- + izračun osvjetljenja se provodi samo za slikovne elemente na koje izvor utječe (moguć veliki broj izvora) <https://toji.github.io/webgpu-clustered-shading/>
- potrebna je dodatna memorija, problem prikaza prozirnih objekata

8.2. PRELAZNI MODELI

WHITTEDOV MODEL OSVJETLJAVANJA

Intenzitet koji dolazi do promatrača iz točke na površini funkcija je intenziteta

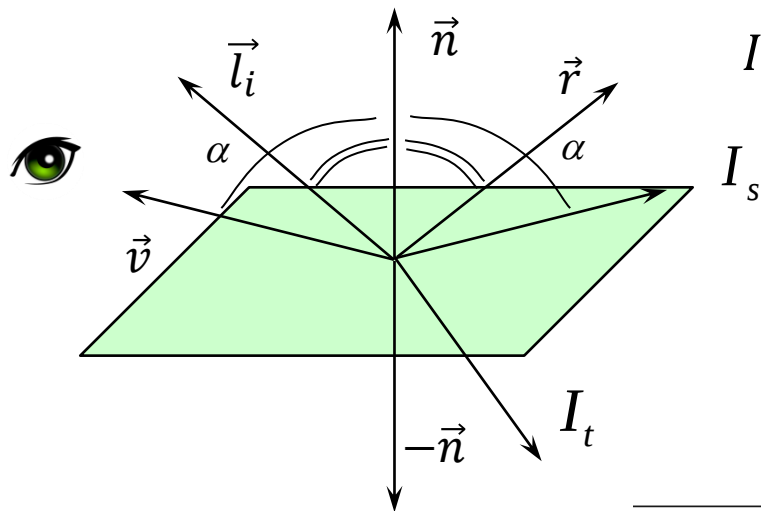
- ambijentne svjetlosti
- difuzno odbijene svjetlosti
- zrcalno odbijene svjetlosti (intenzitet sa susjednih površina kao posebnih izvora)
- lomljena svjetlost https://threejs.org/examples/#webgl_refraction

$$I = k_a I_a + k_d \sum_{j=1}^m I_{p,j} (\vec{l}_j \cdot \vec{n}) + k_s I_s + k_t I_t$$

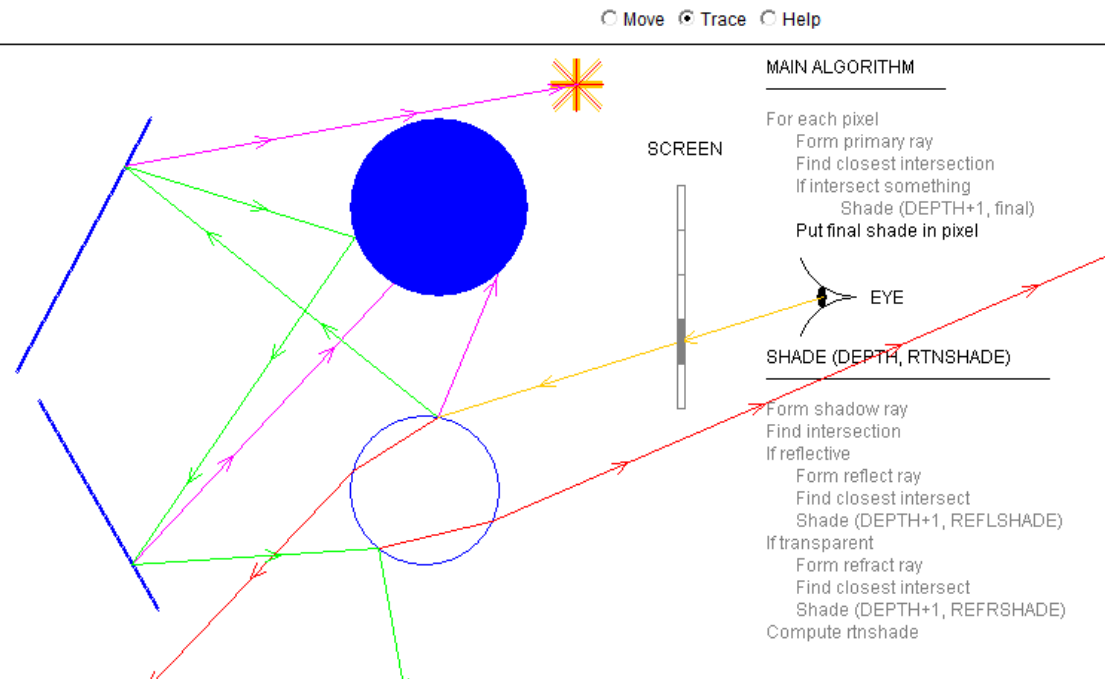
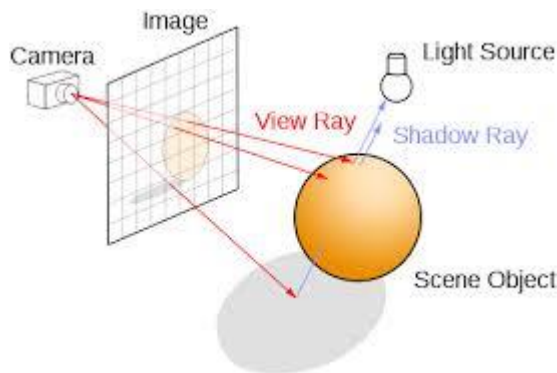
- I_{pi} intenzitet i-tog točkastog izvora
- k_s koeficijent zrcalnog odbijanja
- I_s intenzitet zrcalno odbijene svjetlosti
- k_t koeficijent lomljene svjetlosti
- I_t intenzitet lomljene svjetlosti
- $\vec{l}_j \cdot \vec{n}$ skalarni produkt vektora prema izvoru j i normale

- https://phet.colorado.edu/sims/html/bending-light/latest/bending-light_en.html
- <https://stemkoski.github.io/Three.js/Bubble.html>

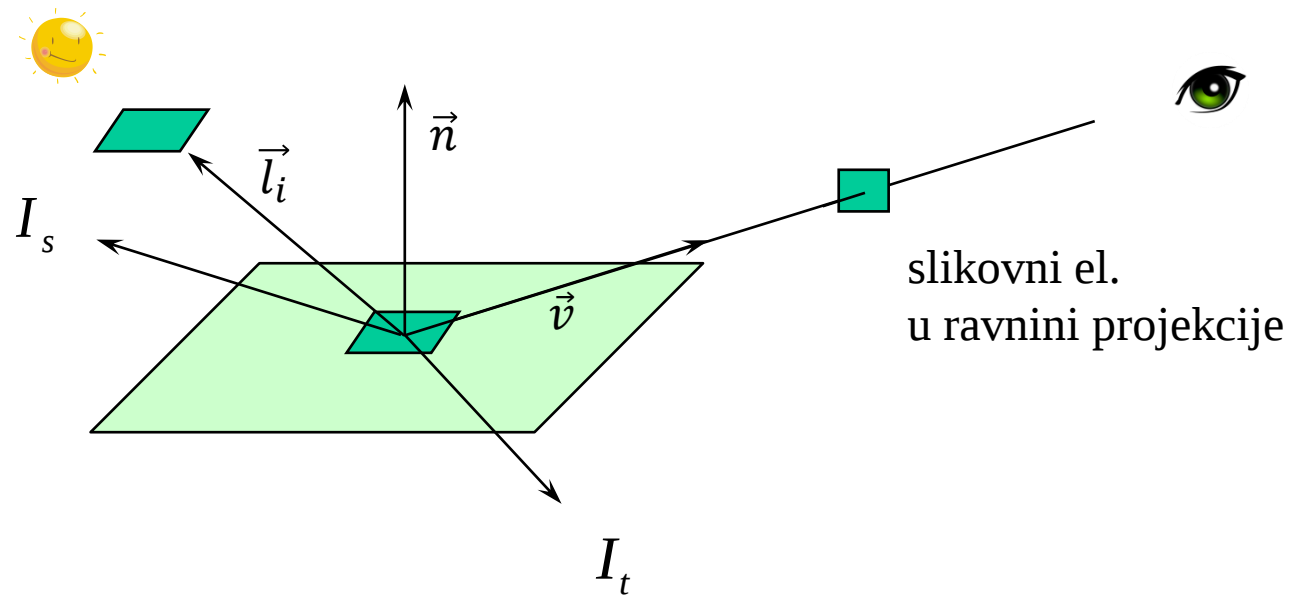
doprinosi zrcaljenja i loma svjetlosti



$$I = k_a I_a + k_d \sum_{j=1}^m I_{p,j} (\vec{l}_j \cdot \vec{n}) + k_s I_s + k_t I_t$$



- postupak se sastoji od određivanja slikovnog elementa pogođenog zrakom od promatrača (kroz ravninu projekcije) i praćenja zrake dalje u scenu
- kod svakog pogotka objekta u sceni postupak se rekurzivno nastavlja za reflektiranu I_s (primarnu zraku) i lomljenu I_t (sekundarnu) zraku
- postupak se završava kada zraka izađe iz scene bez pogotka ili na unaprijed zadanoj dubini rekurzije (3-10)
- ukoliko postoji prepreka između izvora i točke u kojoj računamo osvjetljenje tada neki drugi objekt baca sjenu na promatranu točku



http://alteredqualia.com/three/examples/materials_shaders_fresnel.html

http://threejs.org/examples/webgl_materials_cubemap_refraction

<https://playground.babylonjs.com/#FEEK7G#116>

Ž. Mihajlović, ZEMRIS, FER

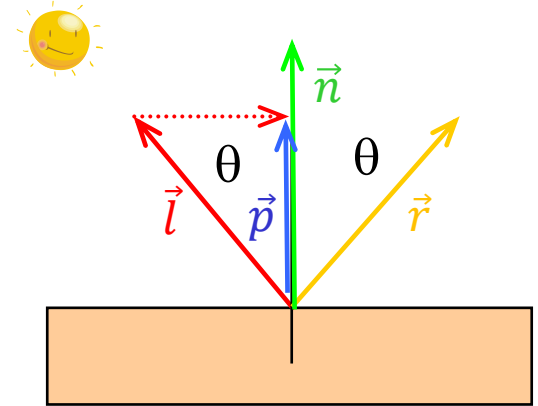
Određivanje reflektirane i lomljene zrake

- određivanje reflektirane zrake – slično kao kod zrcalne komponente
normiramo vektore (duljina je jedan)
projekcija vektora $\vec{l}$ na $\vec{n}$ je vektor $\vec{p}$

$$\vec{p} = (\vec{l} \cdot \vec{n})\vec{n}$$

$$\vec{r} = 2(\vec{p} - \vec{l}) + \vec{l} = 2\vec{p} - \vec{l}$$

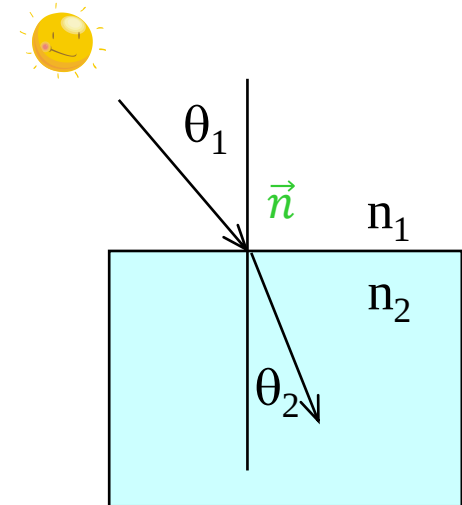
http://immersivemath.com/ila/ch03_dotproduct/ch03.html#fig_dp_reflection_vector



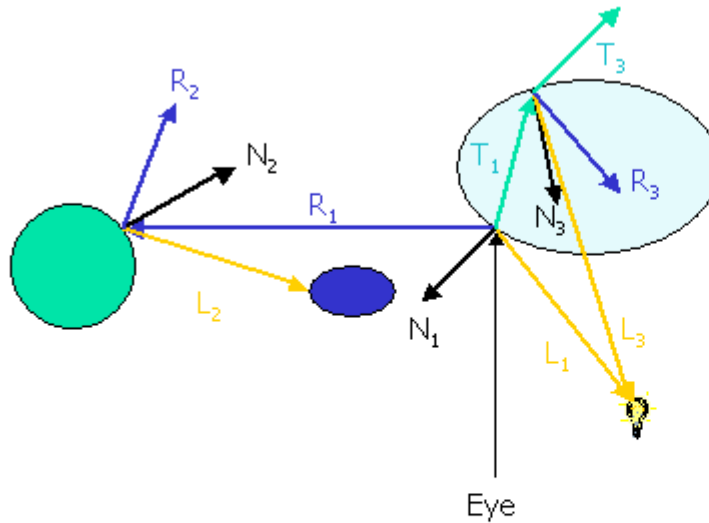
- određivanje lomljene zrake
– Snell-ov zakon loma

$$\frac{\sin \theta_1}{\sin \theta_2} = \frac{n_2}{n_1}$$

<https://tympanus.net/Tutorials/GlassEffect/>
<http://micro.magnet.fsu.edu/primer/java/refraction/criticalangle/index.html>
<http://www.babylonjs.com/Demos/Fresnel/>



- STABLO ZRAKE

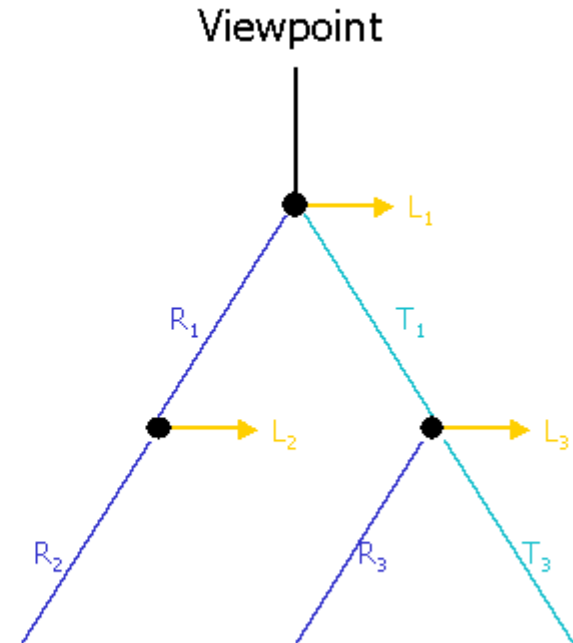


N normala na površinu

R reflektirana zraka

L zraka sjene

T lomljena zraka



rekurzivna formula za računanje intenziteta

- koeficijenti s oznakom $k+1$ koeficijenti su konkretne površine na dubini $k+1$

$$I^{(k)} = k_a^{(k+1)} I_a^{(k+1)} + k_d^{(k+1)} \sum_{j=1}^m I_{p,j} \left(\vec{l}_j^{(k+1)} \cdot \vec{n}^{(k+1)} \right) \\ + k_s^{(k+1)} I_s^{(k+1)} + k_t^{(k+1)} I_t^{(k+1)}$$

nedostaci postupka

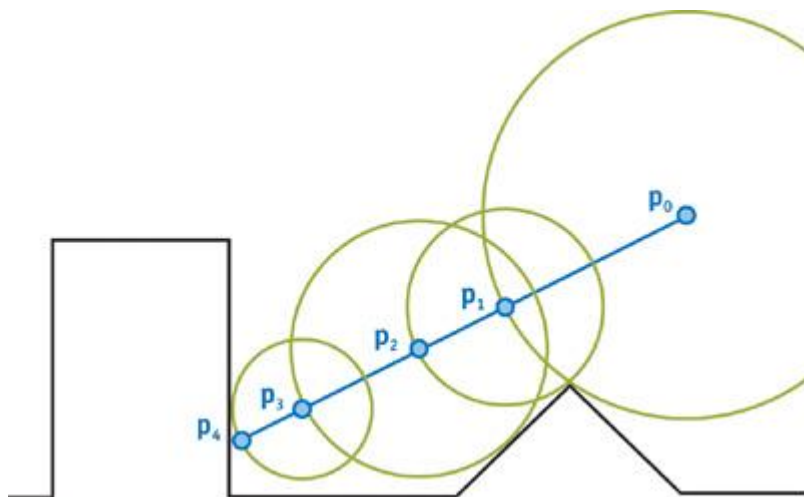
- 75-95 % vremena troši se na određivanje sjecišta zrake i poligona
 $\Rightarrow$ koriste se minimaks provjere s kvadrima ili kuglama, oktalna stabla, BSP stabla kako bi se smanjio utrošak vremena na računanje sjecišta
 $\Rightarrow$ koraćanje duž zrake (engl. ray marching)
- postupak svejedno ostaje vremenski zahtjevan

<https://aadebdeb.github.io/webgl-raytracing/brdfs.html>

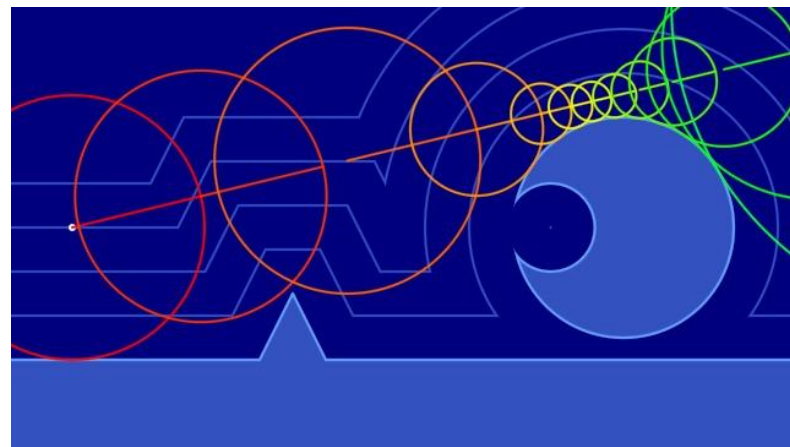
Koraćanje duž zrake (engl. ray marching)

- potrebna je predznačena funkcija udaljenosti (SDF engl. Signed distance functions) za svaku točku scene (izračuna se unaprijed ili ju za implicitne površine znamo odrediti)
- npr. za scenu s jednom kuglom SDF je pozitivna ako je udaljenost $d > R$, općenito kod CSG (Constructive solid geometry) objekata je lako odrediti SDF
- duž zrake se rade koraci a veličinu koraka određuje vrijednost SDF na promatranom mjestu dok ne dođemo dovoljno blizu površine

<https://aadebdeb.github.io/webgl-raytracing/ray-tracing-based-on-ray-tracing-in-one-weekend.html> samples
<https://stephaneinier.com/archive/editSDF/>



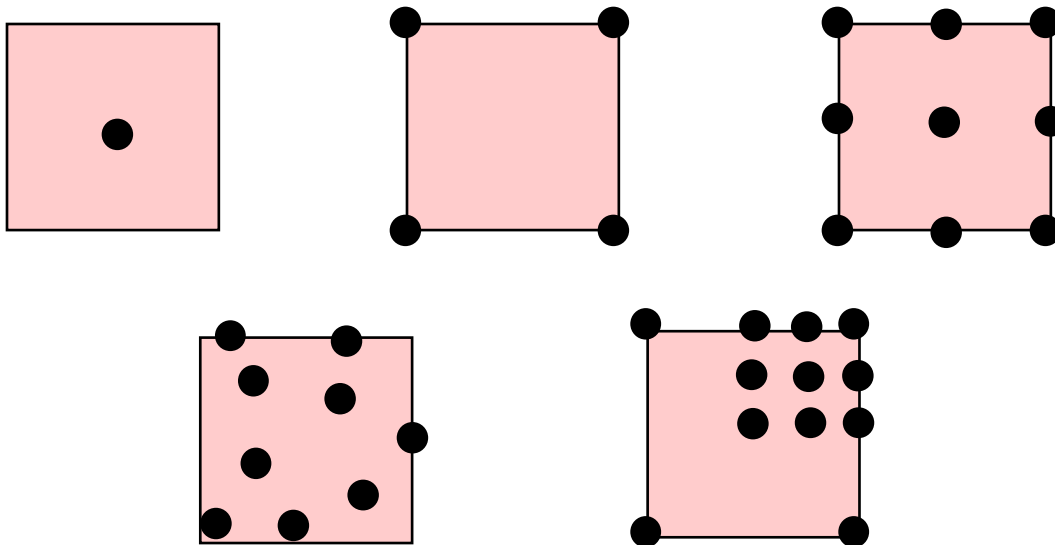
<https://www.shadertoy.com/view/4dKyRz>



nedostaci

- pojava aliasa

⇒ povećano uzorkovanje – 4, 9, po razdiobi, adaptivno)



<http://www.povray.org/>

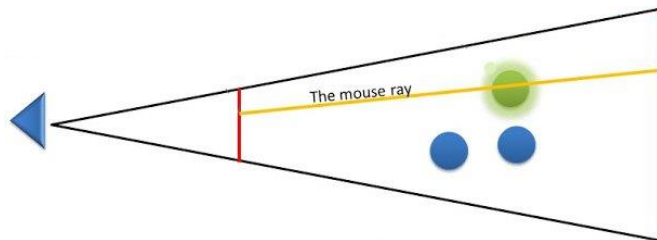
https://threejs.org/examples/webgl_raymarching_reflect.html

<http://hoxxep.github.io/webgl-ray-tracing-demo/>

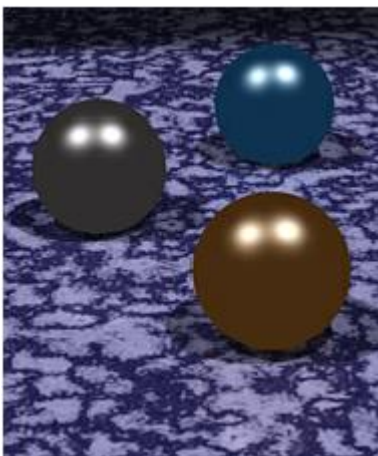
Ostale primjene

Odabir mišem (engl. picking)

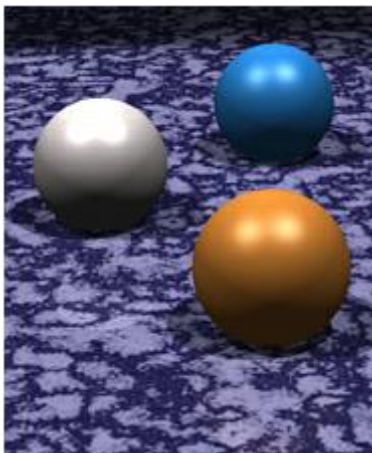
- postupak bacanja zrake primjenjuje se i kod odabira objekata mišem: kada imamo veći broj objekata u sceni iz 2D pozicije miša određuju se koordinate u sustavu prikaza – baca se jedna zraka u scenu i određuje najbliže probodište (taj objekt/ploha je odabran)
Za veliki broj objekata ovaj postupak je spor pa treba koristiti omeđujuće volumene.



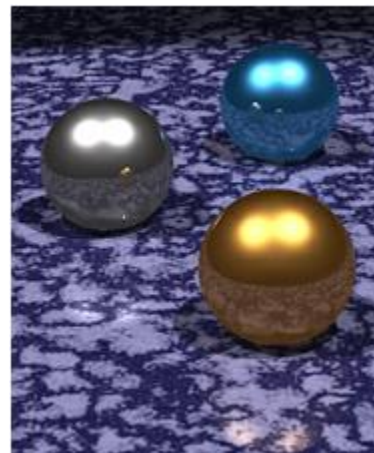
- spremnik za odabir (glSelectBuffer) prikaz se radi samo za mali prozorčić oko pozicije miša. Potreban prikaz cijele scene za pojedini odabir.
- kodiranje bojom
Načinimo prikaz objekata (obično u teksturu) tako da boju koristimo za indeksiranje. Npr. Objekt prikazujemo u boji određenoj s identifikatorom objekta (ili ID poligona). Kada mišem dođemo iznad nekog slikovnog elementa boja nam daje koji je to element.



nema difuzne komp.



difuzna komponenta

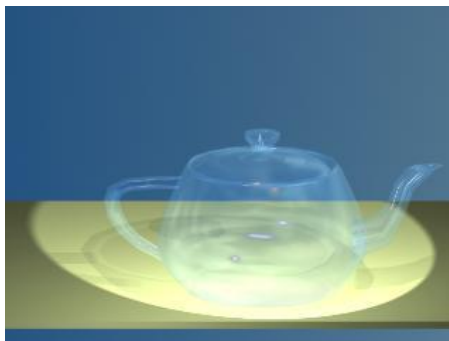


zrcaljenje okoliša



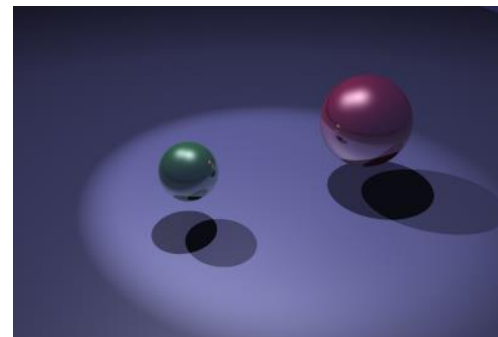
proziran objekt

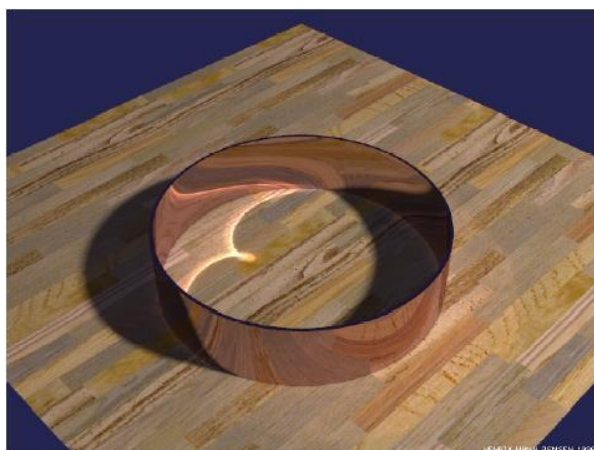
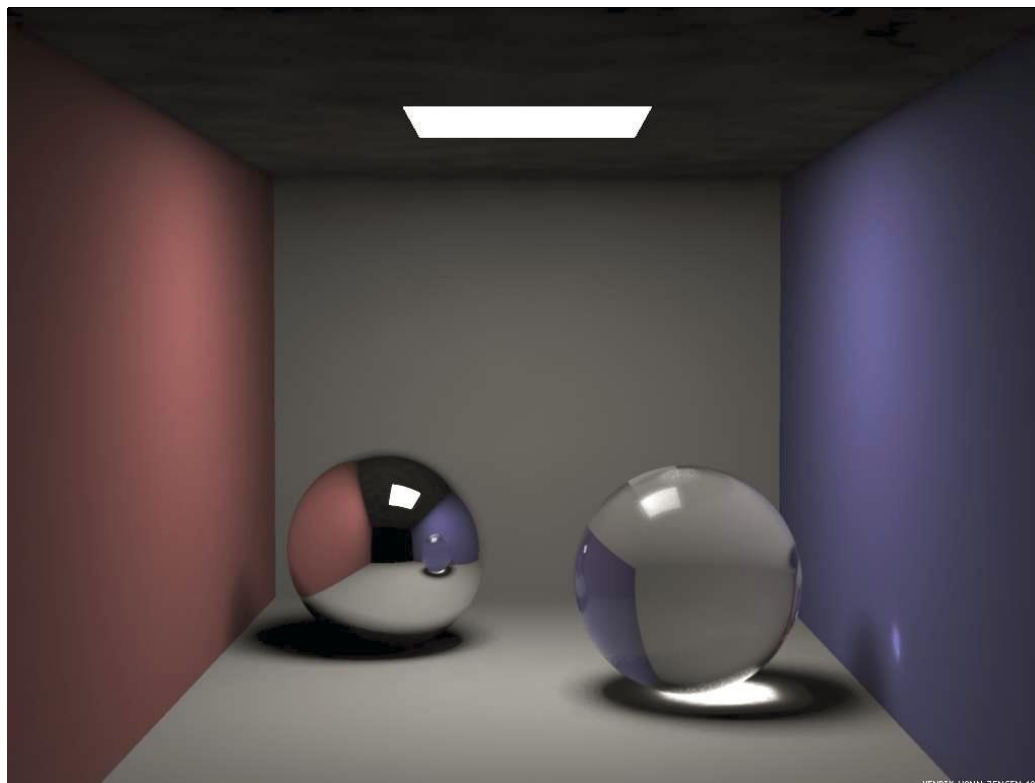
<http://www.vill.ee/eye/>



bačene sjene, reflektor

https://threejs.org/examples/webgl_lights_spotlight





Vizualni učinci koji nisu obuhvaćeni:

Kaustika

<http://madebyevan.com/webgl-water/>

https://erichlof.github.io/THREE.js-PathTracing-Renderer/Switching_Materials.html

<https://martinrenou.github.io/threejs-caustics/>

Podpovršinsko raspršenje (*Subsurface Scattering*)

https://alteredqualia.com/three/examples/webgl_materials_skin.html