

Osnove virtualnih okruženja ak.g. 2019./2020.

3.laboratorijska vježba

Ema Popović

0036506085

20.4.2020.

Cilj treće laboratorijske vježbe upoznavanje je sa iscrtavanjem. Iscrtavanje je skup podataka kojima se iz opisa virtualne scene proizvodi slika. Metoda praćenja zrake široko je upotrebljavana metoda iscrtavanja. Zahtjeva veliku količinu računskih operacija, pa se najčešće koristi u offline računalnoj grafici.

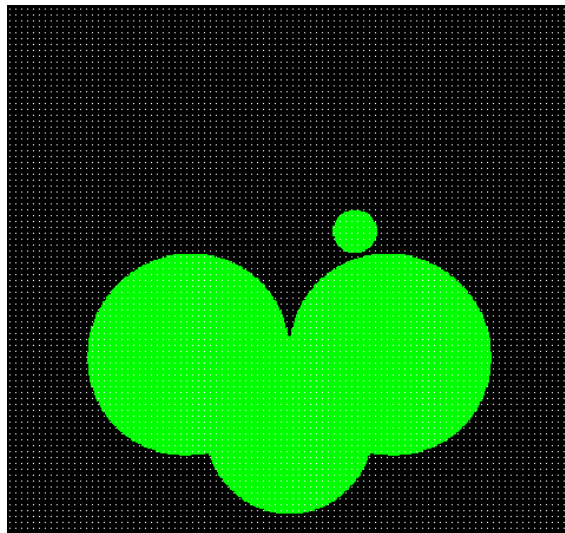
Postupak iscrtavanja

Praćenje zrake odlično prikazuje refleksije, oštre sjene i prozirnost. Njeni principi su jednostavni i dobro nam služe za uvod u iscrtavanje. Prati se zraka koje ulazi u scenu kroz svaku točku zaslona. Zraku promatramo kao pravac kroz dvije točke. Početnom tom označava se oko promatrača, a drugom točkom piksel na ekranu za iscrtavanje. Traži se presjek zrake s predmetima u sceni, te ako postoji računa se osvjetljenje u toj točki. U vježbi se to implementira tako da se ispita i odredi presjek zrake sa svakim objektom i odabere onaj s najmanjom udaljenosti od početka zrake. Svakom pikselu se onda mora pripisati intenzitet za što se u ovoj vježbi koristi Phongov model odbijanja svjetlosti. Prema tom modelu svjetlost dijelimo u tri komponente: ambijentalnu, difuznu i spekulativnu. Ambijentalna aproksimira globalno osvjetljenje okruženja, difuzna opisuje raspršivanje svjetla po predmetu, a spekulativna aproksimira odsjaj na predmetu. Sve komponente zapravo su vektori tri boje – crvene, zelene i plave. Udio svake se određuje vrijednosti između 0 i 1. Za realističan prikaz treba se uračunati i utjecaj sjene na okruženje. Za ovaj dio koriste se zrake za ispitivanje sjene. Idu od mjesta presjeka do pojedinog izvora svjetlosti i provjeravaju nalazi li se išta na putu. Ako naiđu na objekt, utjecaj lokalnog osvjetljenja ne uračunava difuznu i spekularnu komponentu. Nakon toga se rekurzivno računaju zrcaljane i lomljene zrake. Nakon cijelog navedenog procesa dobiva se osvjetljenje u točki presjeka zrake sa scenom. Naravno, različiti objekti različito kombiniraju doprinose različitih komponenti. Zbog toga se odbijeno i refraktirano osvjetljenje množe željenim koeficijentima od 0 do 1. Različitom kombinacijom predstavljaju se različite karakteristike materijala.

Faze iscrtavanja

Silueta scene

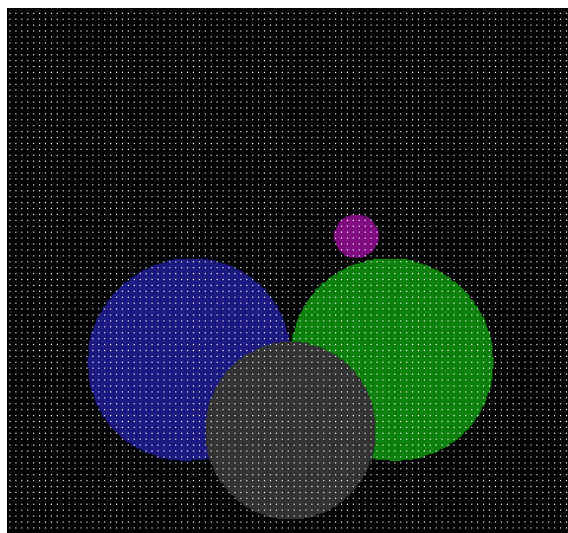
Rade se prve izmjene na kodu. U klasi `ColorVector` implementirana je metoda `correct` koja se nedopuštene vrijednosti boja mijenja u dopuštene. U klasi `Screen` implementirana je metoda `getPoint` koja za svaki piksel slike vraća točku na ekranu. U klasi `Sphere` implementirana je metoda `getNormal` koja vraća vektor normalu za zadanu točku na sferi i metoda `intersection` koja ispituje postojanje presjeka zadane zrake sa sferom. U klasi `Scene` implementirana je prva verzija metode `traceRay` koja računa boju u točki presjeka zrake. Oni pikseli čije zrake sijeku scenu obojani su zeleno i prikazuju siluetu scene.



Slika 1: Silueta scene

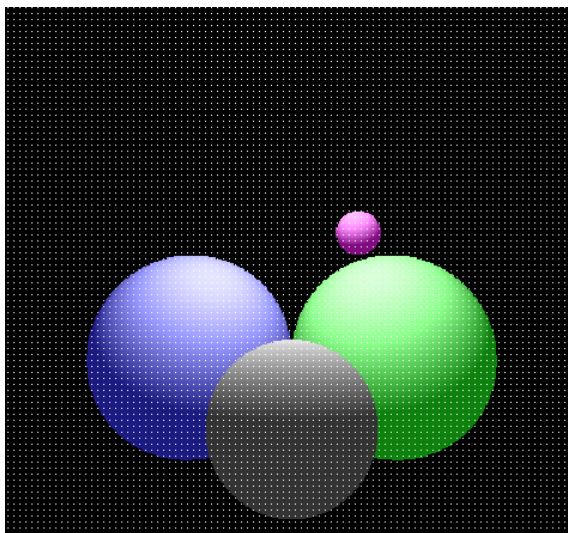
Lokalno osvjetljenje

Idući korak u razvoju okruženja je dodavanje lokalnog osvjetljenja u metodu `traceRay`, kako bi se vidjele boje objekata, a ne samo silueta. Lokalno osvjetljenje se po Phongovom modelu sastoji od tri komponente. Prvo se računa ambijentna komponenta, globalno osvjetljenje objekta. Pomoću zadanog ambijentalnog osvjetljenja i koeficijenta materijala određuje se reakcija materijala na ambijentnu svjetlost. Pomoću ovog procesa vide se boje objekata i njihov raspored u prostoru.



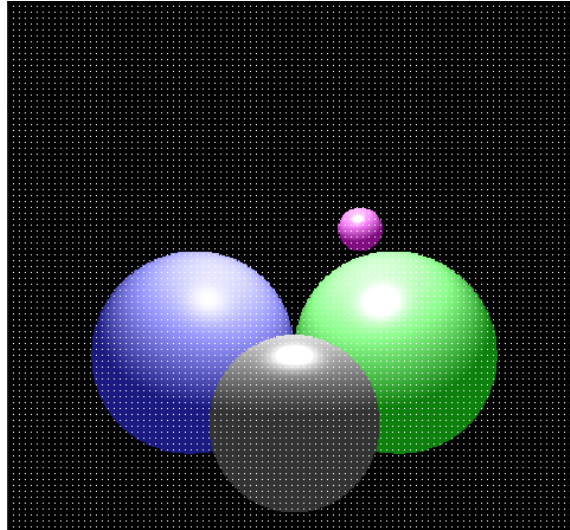
Slika 2: Scena s ambijentnom komponentnom

Nakon ambijentne računa se difuzna komponenta osvjetljenja. Ona opisuje raspršivanje svjetla na predmetu. Raspršivanje na svakom objektu ovisi o ulaznom kutu zrake. Ovakvo osvjetljenje pomaže pri prikazu oblika objekta i njegovog volumena.



Slika 3: Scena s ambijentnom i difuznom komponentnom

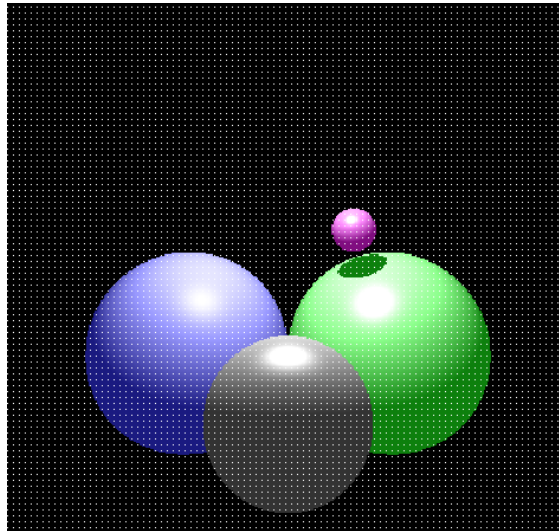
Zadnja komponenta je spekularna. Aproksimira spekularnu svjetlost, karakteriziranu oštrim odsjajem. Za uspješan rezultat dodatno se trebala implementirati metoda `getReflectedVector` iz klase `Vector`. Izvod potrebne jednadžbe nalazi se u uputama za vježbu. Pomoću ove komponente sjajniji materijali imaju oštriji odsjaj.



Slika 4: Slika s ambijentnom, difuznom i spekularnom komponentom osvjetljenja

Sjene

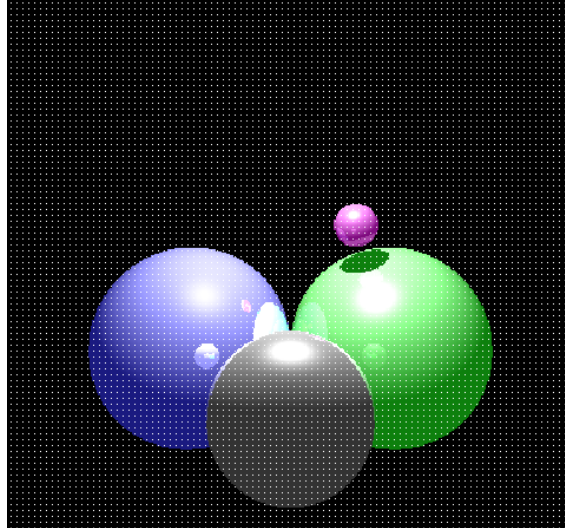
U sljedećem koraku dodaju se sjene. Za to je potrebno implementirati metodu `shadow` u klasi `Scene`. Pomoću nje provjerava se postoji li objekt između točke i izvora svjetlosti. Ako postoji, u toj točki računati će se samo ambijentna komponenta osvjetljenja. Opet se trebala se izmijentiti i metoda `traceRay`. Rezultat je sjena na zelenoj sferi koju uzrokuje ljubičasta sfera iznad nje. Dodavanjem sjene bolje se prikazuje gdje je izvor svjetlosti i u kojem odnosu su objekti.



Slika 5: Scena s lokalnim osvjetljenjem i sjenama

Reflektirane zrake

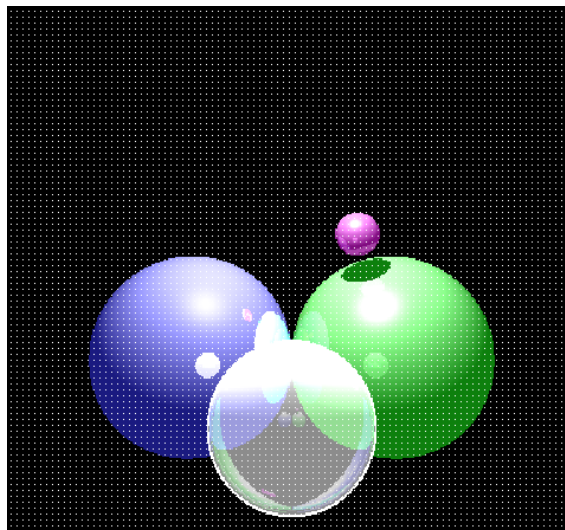
Pomoću implemetacije metode `getReflectedVector` klase `Vector` i rekurzije nad metodom `traceRay` računa se reflektirano osvjetljenje. Na sferama su vidljivi odsjaji drugih sfera.



Slika 6: Slika s lokalnim osvjetljenjem, sjenama i reflektiranim zrakama

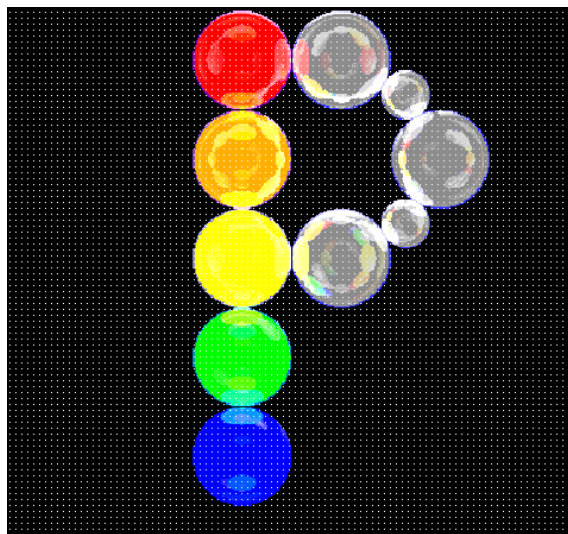
Refraktirane zrake

Potrebno je bilo implementirati metodu `getRefractedVector` uz pomoć Snellovog zakona. Koristeći ovu metodu i rekurziju nad najnovijom verzijom `traceRay` dobiva se konačna slika. Srednja sfera više nije puna i siva. Zbog velikog koeficijenta refrakcije postala je gotovo prozirna.



Slika 7: Potpuna slika

Prvo slovo prezimena (P)



Slika 8: Slika slova P