

Introduktion till 3dsMax
TNM061/TNGD25 Lab 3:
Ljussättning och rendering

Contents

1	Ljuskällor	2
1.1	Enkla ljuskällor	2
1.2	Val av renderare	4
1.3	Arealjuskällor	4
1.4	Sekundärt ljus	5
1.5	Renderingsinställningar	5
1.6	Skylight	7
1.7	Exponeringskontroll	7
1.8	Daylight System	8
2	Reflexion och brytning	8
2.1	Testscen	8
2.2	Ett vinglas	9
2.3	Ett glasmaterial	11
2.4	Renderingsinställningar	12
2.5	Caustics	12
3	Uppgifter att redovisa	14

1 Ljuskällor

Ljussättning är en viktig del av datorgrafik, såväl estetiskt som tekniskt. En bra ljussättning är väldigt viktig för helheten, och en dålig ljussättning kan förstöra en i övrigt välgjord scen. Ljussättning i verkligheten för fotografi, film och scenbruk är en svår konst som det tar lång tid att lära sig, och för att göra det bra krävs det övning, talang, tålamod och ett kritiskt öga för resultatet. Detsamma gäller för ljussättning i datorgrafik, och det här är inte alls någon introduktion till estetiken i ljussättning. Det är däremot en introduktion till tekniken bakom ljusberäkningarna som sker i en modern renderare, och även om du inte blir så vidare bra på att göra en snygg ljussättning efter bara ett fyratimmars labpass så får du åtminstone erfarenhet av att hantera de grundläggande verktygen, och förhoppningsvis en insikt i vilka metoder som finns.

För att kunna börja behöver vi något att ljussätta. Skapa en enkel scen av boxar som ser ut ungefär som i figuren nedan. Skapa ett material med enbart diffus reflexion ($specular\ level = 0$), och sätt dess färg till ljusgrått. Applicera materialet på samtliga objekt i din scen.

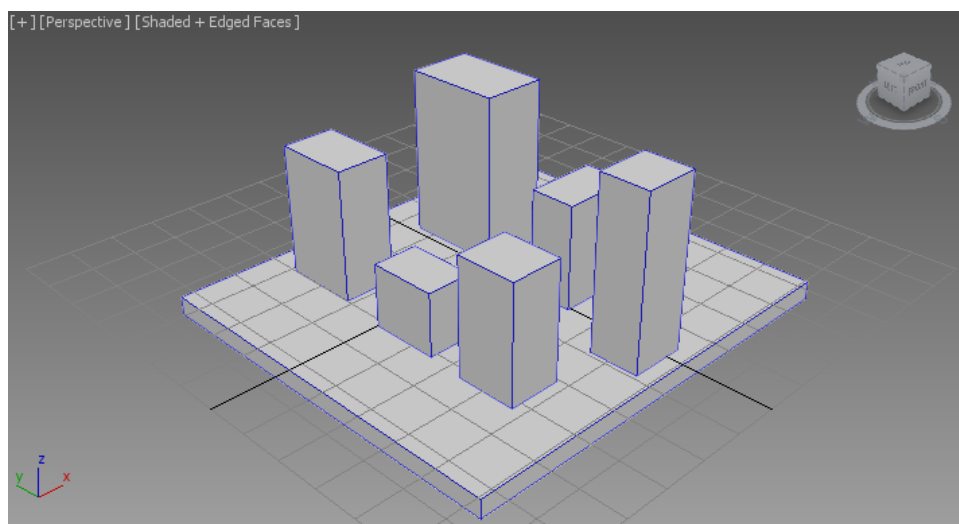


Figure 1: En enkel scen för experiment med ljussättning

Rendera. Resultatet ser platt ut, eller hur? Om du inte skapar någon ljuskälla i scenen så använder 3dsMax en standardljuskälla som är väldigt enkel. Den kastar till exempel inga skuggor. Dess syfte är bara att lysa upp scenen så att bilderna du renderar inte blir helt svarta innan du lagt in några egna ljuskällor. När du lägger in en egen ljuskälla i scenen så avaktiveras den här trista ljuskällan.

1.1 Enkla ljuskällor

Skapa nu en ljuskälla av typen *Target Direct*. "Direct" betyder i detta fall inte "direkt", utan det är en litet olämplig förkortning av ordet "directional", vilket avser en riktad ljuskälla med parallella strålar.

Rikta in ljuskällan så att den lyser mot din scen snett uppifrån, och ställ in storleken på "ljuskonen", det som kallas *hotspot* och *falloff* i *Directional parameters* under *Modify-*

fliken, så att alla objekt i din scen är belysta, eventuellt med undantag för golvet/marken. För att enkelt se vart ljuskällan är riktad och hur stort område den belyser så kan du högerklicka på titeltexten högst upp till vänster i en viewport, där det exempelvis står "right", "front" eller "perspective", och välja att i den viewporten titta i ljuskällans riktning från dess aktuella position. Kontrollerna längst ner till höger i programmets huvudfönster, de som används för att rotera och zooma i vyn, kommer då att flytta på ljuskällan och ändra dess spridning. Testa gärna! Det här är ofta ett väldigt smidigt sätt att rikta in en ljuskälla, även om det kan verka litet bakvänt.

När du riktat in ljuskällan, rendera scenen igen. Notera att inte heller den nya ljuskällan kastar skuggor. Du måste nämligen gå in i *Modify*-fliken och slå på dem. Under egenskaperna för ljuskällan finns en rubrik *Shadows*. Kryssa i rutan *On* för att slå på skuggorna, och rendera igen. Skuggorna är kanske litet väl hårda och mörka, och objekten kanske skuggar varandra på ett fullt sätt, men skuggorna finns där, och det är en väsentlig förbättring.

Den sorts skuggor som används om man inte ber om annat är av typen *Shadow map*. Det är ett litet gammaldags men snabbt och enkelt sätt att göra skuggor, och det kan i många fall vara tillräckligt.

En skuggmapp (*shadow map*) är en avståndsbild (en bild över z-djupet för varje pixel i xy-planet) som renderas från *ljuskällans* position i stället för från kameran, och som sedan används för att avgöra om en viss yta är belyst eller ligger i skugga. Den yta som är belyst är helt enkelt den som ligger närmast ljuskällan, vilket är den som syns när man renderar en bild från ljuskällans position. Ytor som inte syns från ljuskällan ligger däremot i skugga. (För detaljer om hur *shadow mapping* fungerar hänvisar vi till andra källor, t ex någon bok om datorgrafik. Wikipedia har också en bra artikel om ämnet.) Avståndsbilden som renderas har en begränsad upplösning, och det syns när du zoomar in på den renderade bilden. För att få ett bra resultat med *shadow maps* behöver du ställa in upplösningen på dina skuggmappar så att de har tillräckligt detaljerade för de vyer du vill rendera.

I den här laborationen ska vi detaljstudera ljus och skuggor, så vi vill använda ett modernare sätt att göra skuggor som också passar bäst för renderaren som vi skall använda senare (*mental ray*). Du kan göra litet olika saker med olika sorters skuggalgoritmer. Under *General Parameters*, *Shadows*, byt typ på skuggan till *Ray Traced Shadows* och rendera igen. Ser du någon skillnad? Om inte, zooma in litet på scenen så att du ser kanten på en skugga i närbild.

Fördelen med shadow maps att det är lättare att göra suddiga skuggor med den metoden. Raytraced skuggor blir skarpa, så skarpa att de lätt uppfattas som hårda och orealistiska. Vi går inte närmare in på hur man suddar till skuggor med shadow maps, utan vi beskriver i stället hur man med extra beräkningar kan använda raytracing för att beräkna även suddiga skuggor. Moderna datorer är snabba, och det är inte alltid klokt att välja den snabbaste renderingsmetoden. Raytracing av skuggor är mer generellt, och kan ge ett bättre och mer fysikaliskt korrekt resultat utan att man lägger ner en massa manuellt arbete. För rendering där man inte kan använda raytracing, till exempel vid realtidsrendering i datorspel, så är däremot skuggmappar fortfarande en vanlig och viktig metod.

En riktad ljuskälla simulerar ljus som kommer långt bortifrån, så långt bort att strålarna kan anses vara parallella. En ljuskälla som beter sig så är solen, som är så

långt bort att alla punkter på jordytan har i stort sett samma riktning på solljuset i varje givet ögonblick. (Ljuset har naturligtvis inte samma riktning relativt jordytan, men det beror på att jordytans riktning ändras.) Andra ljuskällor än solen ligger däremot ofta så pass nära det de belyser att de lyser i olika riktning i olika delar av scenen. En sådan ljuskälla är *Target Spot*. Skapa en sådan i stället för din *Target Direct* och studera skillnaden. Du behöver inte skapa en ny ljuskälla, du kan byta typ på den du redan har i drop-down menyn högst upp i Modify-panelen. Byt typ till *Spot*. Ljuset kommer nu att *divergera*, det utgår från en punkt och sprider sig i scenen. Skuggorna kommer att peka bort från ljuskällan åt litet olika håll. Effekten blir starkast om ljuskällan placeras nära objektet.

Egentligen ska ljuset också avta i intensitet med avståndet, men det finns en del problem med att räkna på intensiteten om man gör så. Testa med att slå på det som kallas *Decay* för din spotlight! Välj det som enligt fysikens lagar borde vara korrekt, *Inverse square*, alltså att intensiteten minskar med kvadraten på avståndet till ljuskällan. Du kommer förmodligen att se en mycket mörkare scen. Dra upp *Multiplier* för att öka intensiteten tills det ser OK ut. Som du ser blir det en stor skillnad mellan objekt som ligger nära ljuskällan och objekt som ligger längre bort, och det ser litet för dramatiskt ut. Det beror på att den virtuella kameran ger skarpere kontraster jämfört med hur riktiga kameror och mänskliga ögon fungerar. I stället kan man välja ”Inverse”, ljus som avtar proportionellt med avståndet i stället för avståndet i kvadrat. Det är egentligen helt orealistiskt, men kan se bättre ut i många fall. Testa! Du behöver förmodligen ändra *Multiplier* igen för att få lagom mycket ljus.

1.2 Val av renderare

Om du inte har bett om annat använder 3dsMax den renderare som kallas *Default Scanline Renderer*. Den är litet gammaldags, men väldigt snabb och tillräckligt bra för vissa ändamål. För återstoden av den här laborationen kommer vi däremot att använda den modernare renderaren *Mental Ray*. Den är något långsammare, men ger mycket mer realistiska bilder just när det gäller ljussättning och skuggor. Under *Render Setup* (knappen längst till vänster i *Toolbar* av dem som har tekannor i ikonerna), välj *Renderers*: till *NVIDIA mental ray*.

1.3 Arealjuskällor

Verkliga ljuskällor ger inte helt skarpa skuggor. Det beror på att de inte är punktformiga, och därför blir det inte en abrupt övergång från ljus till skugga. I stället försvinner ljuskällan gradvis bakom ett objekt när man går från ljus till skugga. Den region där ljuskällan är delvis skymd går på engelska under det latinska namnet *penumbra*. På svenska säger man helt enkelt halvskugga, vilket är en direkt översättning. Alla ljuskällor har en viss utbredning, även solen, och ger åtminstone en liten region med halvskugga. Det kan vara viktigt att simulera det i renderingen, och det gör man med så kallade *arealjuskällor* (area light sources).

Skapa en ny ljuskälla av typen *mr Area Spot*, och ta bort eller stäng av din tidigare ljuskälla. I rollouten *Area Light Parameters*, ändra storleken på den yta som ljuskällan ska anses täcka så att den blir rimligt stor i förhållande till dina objekt i scenen. Sätt

Height och *Width* till ungefär samma värden som storleken på en av dina boxar. Du ser en representation av ljuskällans area i viewporten. Rendera. Se hur skuggorna blir skarpa närmast objekten, men mer och mer suddiga längre bort.

Renderingen av halvsugga baserar sig på att man inte bara tittar på en enda punkt mitt på ljuskällan när man beräknar om den syns eller inte, utan i stället tittar på flera punkter, tar flera *sampel*, och räknar hur stor andel av dem som syns. Antalet sampel kan styras med inställningen *Samples* under rollouten *Area Light Parameters*. Generellt sett behövs det fler sampelpunkter för stora ljuskällor och för ljuskällor som ligger väldigt nära objekt, det som brukar kallas ”mjukt ljus” (*soft light*).

1.4 Sekundärt ljus

Du har kanske märkt att de skuggade delarna av din scen ändå är litet svagt upplysta. Var kommer det ljuset ifrån egentligen? Svaret är att det är vad som kallas *sekundärt ljus* eller *indirekt ljus*, ljus som studsar mot omgivande objekt och sedan landar även på ytor som inte är direkt belysta. Det här är inte något som beskrivs särskilt noga i grundböcker om datorgrafik, men under de senaste åren har datorer blivit så snabba att man i dag rutinmässigt använder renderingsmetoder som ganska nyligen var omöjliga att använda i produktion. Sekundärt ljus i *mental ray* beräknas med två olika metoder som kallas *Final Gather* och *Photon mapping*, metoder som faktiskt inte har så värst många år på nacken. Final Gather går enkelt uttryckt ut på att man först beräknar en bild med endast direkt ljus och sedan använder den i ett andra renderingspass för att räkna ut det indirekta ljuset på ett ungefär. Photon mapping går enkelt uttryckt ut på att man först skjuter ut ”fotoner”, ljusstrålar, från alla ljuskällor och håller reda på var de landar i scenen. Vid rendering tittar man sedan dels på direkta ljuskällor, dels på indirekt ljus från fotoner som passerat genomskinliga objekt eller studsats mot blanka objekt. (En litet mer ingående teknisk beskrivning av såväl *Final Gather* som *Photon mapping* hittar du t ex på Wikipedia eller i programmets hjälpsystem.)

1.5 Renderingsinställningar

Nu ska vi studera vad som händer när man ändrar kvalitetsinställningarna för renderingen. Behåll renderingsfönstret öppet så att du ser de olika kontrollerna under bilden enligt figur 2. Alla inställningarna har det gemensamt att bilden blir sämre när du drar kontrollen åt vänster och bättre när du drar den åt höger. Det tar också längre tid att rendera bilden ju högre kvalitet du vill ha. Nästan inget är gratis vid rendering – det mesta är en kompromiss mellan renderingstid och kvalitet.

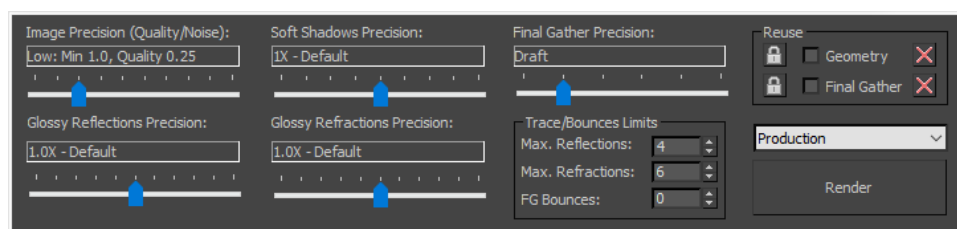


Figure 2: Inställningar i renderingsfönstret

Experimentera först med slidern *Image precision*. Längst åt vänster blir bilden lågupplöst och eventuellt brusig, i mitten blir vissa kanter eventuellt litet taggiga och bilden något brusig, och längst till höger blir bilderna väldigt snygga, men det tar lång tid att rendera. Scenen du har är väldigt enkel, och det skulle ta mycket längre tid att rendera en mer komplicerad scen med högsta kvalitet. Metoden som används här för att rendera pixels kallas *adaptiv stokastisk sampling* (adaptive stochastic sampling). ”Adaptiv” innebär att du ger renderaren en viss frihet att spara in på sampel där det inte händer så mycket i bilden, exempelvis i släta och mjukt belysta ytor, men lägga mer krut på regioner där det behövs, exempelvis längs kanter. ”Stokastisk” innebär att man samplar varje pixel inte bara med flera strålar, utan också på litet olika ställen för att bryta upp skarpa defekter och ersätta dem med brus, något som ögat oftast är mindre känsligt för. Kvaliteten anges som ett minsta antal sampel per pixel och en litet luddig siffra ”quality”. Om minsta antalet sampel är mindre än 1 kommer bilden inte garanterat att samplas i alla pixels utan interpoleras i vissa, vilket kan vara OK i många fall, men man riskerar att vissa regioner blir litet taggiga och fula. Om ”quality” är hög kommer de flesta pixels att räknas ut som ett medelvärde av många sampel. Vad som är ”tillräckligt bra” för den här slidern beror på vad bilden föreställer och vad den ska användas till. Du får helt enkelt pröva dig fram till ett bra värde.

Slidern *Soft shadows Precision* styr hur många skuggstrålar som ska skickas mot arealjuskällor för att beräkna hur stor del av ljuskällan som är skymd av andra objekt. Längs till vänster är funktionen för arealjuskällor avstängd, och arealjuskällor ger då lika skarpa skuggor som om de vore punktljuskällor. Längst till höger samplas väldigt många punkter på ljuskällans yta. Däremellan finns en glidande skala. Flytta slidern och studera resultatet. Vid låg kvalitet ser du brus och tydliga defekter i halvskuggorna i din scen. Mjuka skuggor är generellt sett tunga att beräkna på ett bra sätt, men det är svårt att klara sig utan dem om man vill simulera hur en typisk inomhusbelysning beter sig. Kvaliteten på mjuka skuggor kan också ställas in separat för varje ljuskälla genom att ändra antalet *Samples* i inställningarna för ljuskällan. Slidern i renderingsförnstret ställer sedan in precisionen relativt dessa inställningar, där 1x betyder att man gör som ljuskällans egna inställningar säger. Mindre faktorer än 1x sparar arbete på bekostnad av kvaliteten, och högre faktorer ger bättre kvalitet men en längre renderingstid.

Slidern *Glossy Refractions Precision* styr kvaliteten på halvdiffusa genomskinliga ytor. Vi ska inte göra några sådana i den här laborationen, men kom gärna ihåg att inställningen finns om du skulle behöva den någon gång.

Inställningen *Final gather precision* har att göra med sekundärt ljus. Dra den ända åt vänster och rendera om bilden, så ser du att skuggorna blir helt svarta. Det beror på att renderaren då inte beräknar något indirekt ljus alls. Ibland går det att komma undan med det, men ofta är det bra att ha åtminstone litet *Final Gather*. Det kostar faktiskt inte så otroligt mycket tid och beräkningskraft i moderna renderare, och det ger ofta mycket bättre bilder. När du i stället drar slidern längre åt höger så ser du en litet mer subtil skillnad i den renderade bilden. Bland annat får du ett ljusare stråk där två belysta objekt med ljus färg möts i ett inåtgående hörn. Om du inte ser det i din scen, gör materialet ljusare. Det ser litet konstigt ut när du ser det så här i isolering på helt släta och detaljlösa objekt utan andra effekter, men det är faktiskt så här som verkligt ljus beter sig. Det reflekteras mer ljus mot en yta när den ligger nära och inom synhåll för en annan ljus yta än när den ligger mer fritt eller nära en mörk yta.

Om du vill se effekten av *Final Gather* riktigt tydligt, sätt en kraftig, ljus färg som till exempel knallgrön eller skriande orange på någon av dina boxar och se vad som händer med neutralt färgade ytor i närheten av den. Du bör se en åtminstone svagt färgad diffus reflexion från den närliggande färgade ytan, något som kallas för *color bleeding*. Eventuellt kan du behöva experimentera litet med ljuskällans riktning för att se effekten tydligt. Det här kan vara förvånansvärt viktigt för realismen i vissa bilder, trots att det är en ganska subtil effekt.

Spinnern längst ner som heter *FG Bounces* har också med *Final gather* (FG) att göra. Vi återkommer till det litet senare i stycke 2.4 om reflexion och brytning. Då ska vi också titta närmare på inställningarna *Max Reflections* och *Max Refractions*.

De återstående inställningarna i rutan *Reuse* behandlas inte närmare i den här introduktionen, men vi vill nämna varför de finns. I vissa fall kan man spara arbete genom att inte beräkna om all ljussättning i varje frame i en animering. Det går också att minska det synliga bruset i rörliga bilder genom att beräkna ljuset en gång och sedan återanvända det i efterföljande rutor. Titta gärna i hjälpen för att se hur det är tänkt att fungera.

1.6 Skylight

Det är inte alla scener som har en stark riktad ljuskälla som dominerande belysning. Ofta har man ett stort inslag av ljus som kommer från många olika håll. Inomhus har man ofta ljusa tak och väggar som bidrar starkt till det indirekta ljuset i scenen, och det kan vara bättre att simulera det med en diffus ljuskälla än att lita på att *Final gather* tar hand om allt. Utomhus har man en hel del ljus från himlen, inte bara från solen. Om det är mulet ute så kommer ljuset mot en viss punkt från i stort sett alla riktningar där himlen är synlig från den punkten. De här fallen är svåra att hantera med vanliga ljuskällor, och de har därför en egen sorts ljuskälla, *Skylight*. Släck eller ta bort dina tidigare ljuskällor i scenen och skapa en *Skylight*. Ljuskällans position spelar ingen roll för resultatet – lägg den där den är lätt att komma åt att klicka på. Rendera.

1.7 Exponeringskontroll

Om scenen blir för mörk (eventuellt helt svart) eller för ljus (eventuellt helt vit) medan du experimenterar med olika ljuskällor kan du behöva ändra exponeringsinställningen i renderaren. I menyn *Rendering* → *Exposure Control...* kan du välja hur renderaren ska tolka ljusvärden (som kan variera från 0 till godtyckligt stora tal, beroende på hur många och hur starka ljuskällor du har i scenen) som pixelvärden för utbilden (som behöver ligga mellan 0 och 1 för att visas korrekt på skärmen). Alternativet *mr Photographic Exposure Control* är ett vettigt alternativ som försöker efterlikna hur en kamera skulle bete sig. Du ställer in parametern *EV* (förkortning för *Exposure Value*) för att variera den virtuella exponeringen. Notera att du kan testrendera din scen i låg upplösning i dialogrutan för *Exposure Control*, så att du ser om du träffar rätt när du ställer in exponeringen. Experimentera gärna!

1.8 Daylight System

Utomhus har man ofta en kombination av direkt solljus och diffust ljus från himlen. För att göra det enklare att skapa realistisk ljussättning i utomhusscener finns det ett färdigt *Daylight System* i 3dsMax. Du hittar det i menyn under *Create* → *Systems* → *Daylight System*, eller under *System*-fliken i *Create*-panelen. Det kan vara smidigt att använda om du vill rendera en utomhusscen, eftersom du ställer in konkreta parametrar som plats och tidpunkt snarare än i vilken vinkel solen ska lysa. Det är framför allt användbart för att visualisera hur ljus faller i en planerad verklig miljö eller inuti ett hus som inte byggts än. Då vill man nämligen testa hur det ser ut vid olika tidpunkter på dagen och under olika årstider. Vi går inte in på detaljer om hur *Daylight System* fungerar. Det är i princip bara en kombination av en riktad ljuskälla och en *Skylight*. Detaljer om hur man ställer in den finns i hjälpsystemet, och även i en separat tutorial.

2 Reflexion och brytning

Renderaren *mental ray* är en så kallad raytracer. Raytracing är en mycket lämplig renderingsmetod för att rendera blanka och genomskinliga objekt, och eftersom sådana objekt kan vara väldigt viktiga och centrala i många scener ska vi titta litet närmare på just speglingar och brytningar.

2.1 Testscen

Gör *Reset* på 3dsMax om du inte redan gjort det, för att ta bort eventuella kvarvarande inställningar från föregående uppgift. Glöm inte att ställa tillbaka *Render Setup* så att du använder *NVIDIA mental ray*.

För att visa reflexioner och brytningar behöver vi en testscen som kan reflekteras och brytas i objekten vi skapar. I stället för att göra en komplicerad scen med en massa objekt kan vi fuska och göra ett schackrutigt golv. Det är en klassisk bakgrund för att tydligt visa just reflexioner och brytningar, även om det naturligtvis inte är så realistiskt.

Skapa ett *Plane* med storlek 100 x 100 enheter och ganska många segment i båda ledderna så att det går att böja snyggt. Lägg på en *Bend* i X-led med *Angle* satt till -90, samt *Limit Effect* aktiverat med *Upper Limit* 60. Lägg på en *Bend* till i Y-led med *Angle* -75. Klicka ur rutan *Real-World Map Size* i ditt *Plane*, och lägg ett material på objektet som har en diffus map av typen *Checker*. Klicka ur rutan *Use Real World Scale* (om den är ikryssad) och sätt *Tile* till 12 i båda ledderna. Resultatet bör bli något i stil med figur 3.

För ett något tydligare resultat i följande uppgifter, byt gärna färg på rutorna från svart och vitt till något som har litet mindre extrem kontrast, till exempel mellangrått och ljusgrått.

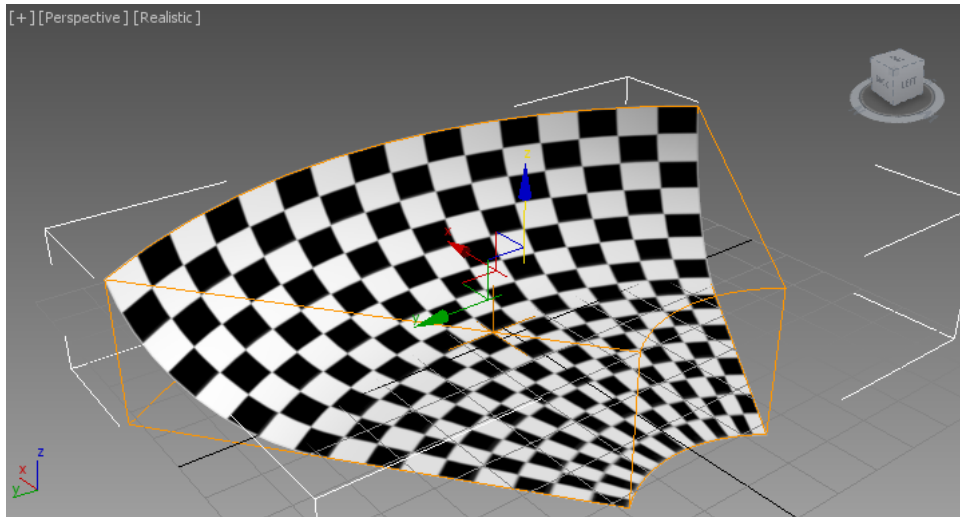


Figure 3: Bakgrund för experiment med raytracing

2.2 Ett vinglas

Nu ska vi göra ett objekt som har både reflexion och transparens. Vi kunde välja en enkel sfär. Många klassiska vetenskapliga artiklar om raytracing har illustrerats i stort sett uteslutande med bilder på sfärer i glas och krom på schackrutiga golv, men för att göra det litet intressantare ska vi göra ett vinglas med hjälp av modifiern *Lathe*. Ordet "lathe" är engelska för "svarv", och modifiern används för att skapa rotationssymmetriska objekt. Rita först en Line ser ut ungefär som figur 4. Rita i Top-vyn så att du jobbar i xy-planet. Börja med att placera ut hörnpunkterna. Högerklicka när du satt ut den sista punkten. Linjera sedan upp de två ändpunkterna så att de ligger på samma x-koordinat. Det kan du göra på flera sätt, men *Align* fungerar bra. i *Modify*-panelen för din *Line*, under rollouten *Selection*, välj *Vertex* (prickarna) för att manipulera enstaka punkter längs linjen. Markera en av ändpunkterna, välj *Align* (alt-A) och klicka på linjen självt som *Target*-objekt. Linjera i X-led mot *Maximum*. Upprepa för den andra hörnpunkten. Nu ligger båda på exakt samma X-koordinat. Det är viktigt för att få ett snyggt resultat i nästa steg. (Om du vill kan du också klicka på hörnpunkterna du vill flytta och knappa in deras exakta koordinater direkt i textrutorna längst ner i programfönstret. Huvudsaken är att de två ändpunkterna har samma x-koordinat innan du går vidare.)

För att göra linjen mjukt krökt, högerklicka på hörnpunkterna en efter en så att du får upp en meny. Ändra punkternas typ till *Béziér*. För punkten på ovansidan av foten där du ska ha ett skarpt hörn väljer du i stället *Béziér Corner*. Pilla med handtagen på tangenterna en stund och flytta hörnpunkterna tills du är nöjd med resultatet. Det lönar sig att vara noggrann här, men du kan naturligtvis gå tillbaka och ändra senare om du inte blir nöjd.

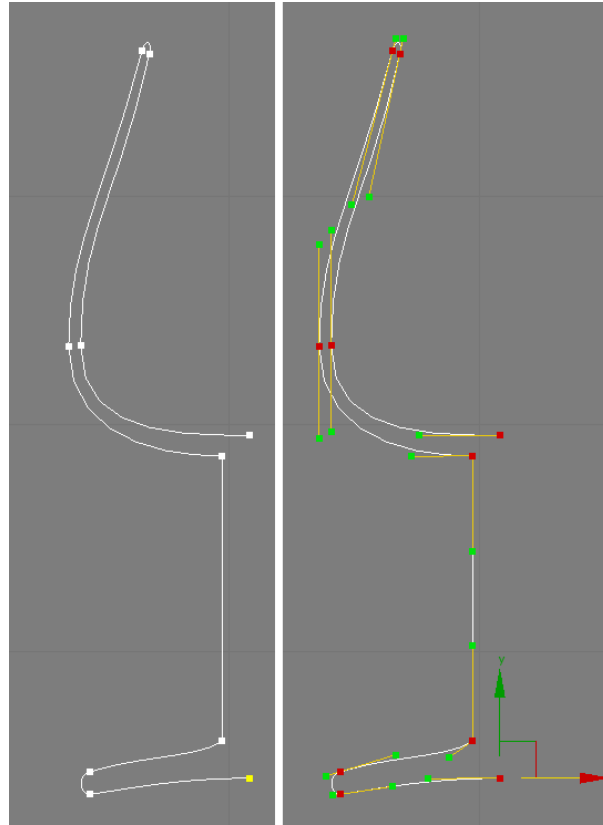


Figure 4: Profilen för ett vinglas

Lägg sedan modifiern *Lathe* på din *Line*. För *Direction*, välj *Y*, och för *Align*, välj *Max*. Välj *Mesh* under *Output* för att skapa en vanlig polygonmodell. Resultatet bör bli något i stil med vänstra halvan av figur 5. Om det ser fult ut, försök gärna att fixa till proportionerna, men det estetiska är inte huvudsaken här. Om det ser ut som till höger i figuren har du råkat få normalerna åt fel håll. Det beror på åt vilket håll du ritade din *Line* tidigare, och det är svårt att från början veta vad som är "rätt" håll. Fixa problemet om det uppstår genom att kryssa i rutan *Flip Normals* i *Lathe*. Felvända normaler är ett vanligt problem medan man modellerar, speciellt när man skapar olika slags svepta ytor. Kryssa även i rutan *Weld Core*, så blir glaset snyggare precis kring rotationsaxeln.

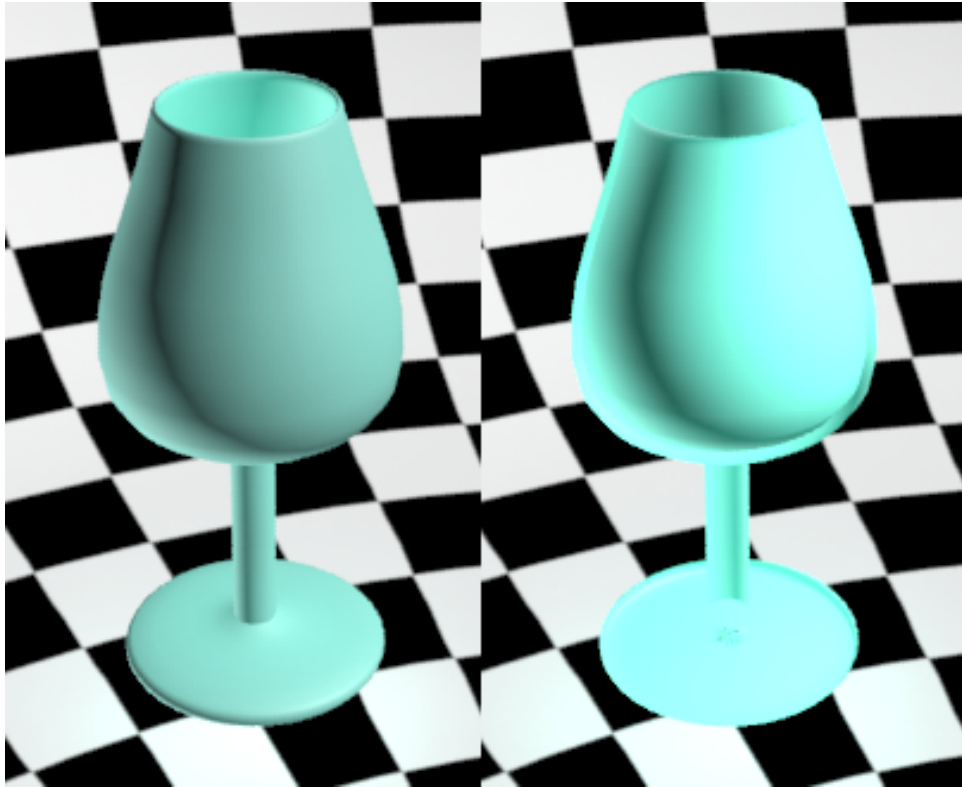


Figure 5: Det färdiga vinglasen, rättvänt och felvänt

2.3 Ett glasmaterial

Nu ska vi göra ett glasmaterial till vinglasen, och vi ska först göra det med ett *Standard*-material för att förstå vad det är som behövs. Det går att hitta färdiga glasmaterial i materialeditorn, men då lär man sig inte lika mycket.

Lägg ett material på glasen. Sätt typen till *Standard*. Välj en diffus färg som är helt svart. Blankt och rent glas har ingen diffus reflexion. Sätt däremot *Specular*-färgen till vitt. Den färgen påverkar av någon outgrundlig anledning även reflexioner som görs med en *Reflection map*.

Glas är genomskinligt, men det reflekterar också en del i ytan. Vi börjar med att göra reflexionen, inte för att det är den viktigaste egenskapen hos glas, utan för att den är enklast att göra och enklast att förstå sig på. Under *Maps* i materialet, välj *Reflection* och sätt typen till *Raytrace*. Rendera. Ditt glas ser ut att vara gjort av blankpolerad metall, men det är i alla fall en korrekt reflexion.

Glas är genomskinligt också. Man kan göra enkel genomskinlighet via *Opacity* för materialet. Testa det. Det blir inte snyggt. Glas har som framträdande egenskap att det bryter ljus som passerar igenom det, och *Opacity* fixar inte att simulera den effekten. Sätt tillbaka *Opacity* till 100 och välj i stället under *Maps* en *Refraction* av typen *Raytrace*. Stäng dessutom av *Reflection* genom att bocka ur kryssrutan. Vi kommer strax till varför. Rendera. Nu tar det väsentligt längre tid att beräkna bilden, men på en modern dator bör du ändå ha rimligt korta väntetider om du har valt rimliga kvalitetsinställningar för renderingen.

Om du tittar noga på ditt renderade glas ser du att det faktiskt sker både en reflexion i

ytan och en brytning genom ytan. Anledningen till det ligger i inställningarna i rollouten för din *Raytrace*. Längst ner under *Refractive Material Extensions* finns en kryssruta *Treat Refractions as Glass (Fresnel Effect)*. Bocka ur den och rendera igen. Nu bör det gå mycket fortare att rendera, men resultatet ser inte längre riktigt ut som glas. Det beror på att reflexionerna är borta. Du kan få tillbaka dem med en separat *Reflection* map, men det finns ett problem med det: glas reflekterar olika mycket i olika vinklar, och hur mycket som bryts beror på hur mycket som reflekteras. Reflexion och transmission i ett material måste tillsammans vara mindre än 1 för att energi inte ska skapas ur intet, och det är litet trixigt att få en *refraction* map och en *reflection* map att variera sina intensiteter tillsammans på det sätt som behövs. Det är inte omöjligt, men det blir en del pill. Därför finns den här praktiska kryssrutan som gör det enklare att skapa just glasliknande material (vilket inkluderar även klara vätskor, till exempel vatten).

2.4 Renderingsinställningar

I renderingsinställningarna under renderingsfönstret finns inställningar som heter *Trace/Bounces limits*. Minska *Max Reflections* och *Max Refractions*. Om du minskar *Max Reflections* till 0 så kommer reflexionen i glasets yta att försvinna. Om du minskar *Max Refractions* till 0 så kommer glaset att bli svart eftersom det då inte skickas någon bruten stråle igenom ytan för att ta reda på vad som finns bakom den.

Vilket är det minsta värdet på *Max Refractions* som ger en någorlunda korrekt rendering av ditt vinglas? Varför just detta värde?

2.5 Caustics

Det finns fortfarande ett fel med vår bild: Glaset kastar en kompakt skugga. Det går att ange att skuggan som kastas inte ska vara helt kompakt, men det är inte så som glas beter sig. Ljus bryts genom glaset och kan landa delvis i den region som skulle ha varit i skugga om glaset hade varit genomskinligt, men omfördelat på grund av ljusets brytning. På liknande sätt kan ett speglande objekt med en krökt yta kasta kraftigt deformerade reflexer som lyser upp dess omgivning. Sådana effekter kallas *Caustics* på engelska, och de är tyvärr krångliga att rendera även med raytracing, eftersom ljus först går från ljuskällan, studsar i ett speglande eller transparent objekt och sedan landar på en diffus yta. Raytracing i sin grundläggande form (det som kallas *invers raytracing*) klarar bara av det omvända fallet: ljus landar på en diffus yta, och den diffusa ytan betraktas sedan genom en eller flera reflexioner och brytningar.

För att rendera *Caustics* i *mental ray* behöver vi använda en renderingsmetod som kallas *Photon mapping*. Tyvärr går det inte att göra det med ett *Standard*-material, så för att fixa till det sista felet byter vi till ett fördefinierat glasmaterial, nu när du vet hur det fungerar.

Byt material på ditt vinglas till *Autodesk Solid Glass*. Grundinställningarna i materialet fungerar bra som exempel. Rendera gärna och titta på resultatet. Skuggan är inte längre svart, och den kanske kan funka på håll och vid en snabb anblick, men den ser ändå konstig ut. Det är fortfarande bara ett fusk med *Opacity* på materialet för skuggberäkningen. Nu ska vi slå på *Caustics*.

Högerklicka på vinglasets, och välj *Object Properties*, eller markera vinglasets och välj *Edit* → *Object Properties* i menyn. Ta fram fliken *mental ray*, och under *Caustics and Global Illumination (GI)*, kryssa i *Generate Caustics* (se till vänster i figur 6). Gör samma sak även för din ljuskälla. Nu är ljuskällan och glaset inblandade i simuleringen av caustics.

Dessutom måste du slå på *Caustics* i renderaren. Det kräver ett extra förberäkningspass som tar tid att rendera och är därför inte påslaget som standard. Välj *Rendering* → *Render Setup* (F10) och under fliken *Global Illumination*, scrolla ner till rollouten *Caustics and Photon Mapping (GI)*, och under *Caustics*, kryssa i rutan *Enable* (se till höger i figur 6).

Rendera. Nu bör sin skugga se betydligt bättre ut, men den kan fortfarande förbättras, eller åtminstone förändras. Bläddra litet längre ner i *Render Setup* tills du hittar *Light Properties*, och ändra *Decay* till 1.5 i stället för 2.0. Rendera igen. Nu klingar inte sekundära effekter som caustics av lika fort, och effekten blir mer framträdande. Däremot blir den inte nödvändigtvis mer realistisk. Caustics simuleras i *mental ray* på ett sätt som strängt taget inte är fysikaliskt korrekt, men det blir oftast tillräckligt bra.

För att få en skarpere och mer detaljerad caustics-effekt behöver du också öka antalet fotoner från ljuskällan. Det gör du några steg nedanför *Caustics* under *Light Properties*, *Average Caustic Photons per Light*. I rutan står ett (i det här sammanhanget) ganska lågt tal, 20 000. Öka det till minst tio gånger högre och se vad som händer med bilden, och med renderingstiden. När caustics är så framträdande som i den här ganska udda och svårrenderade scenen så behöver man ägna mer kraft åt effekten än annars för att få ett bra resultat. Det är inte orimligt att låta förberäkningen av fotoner ta en mer tid än själva slutrenderingen. Experimentera med antalet fotoner och se vilken effekt det har!

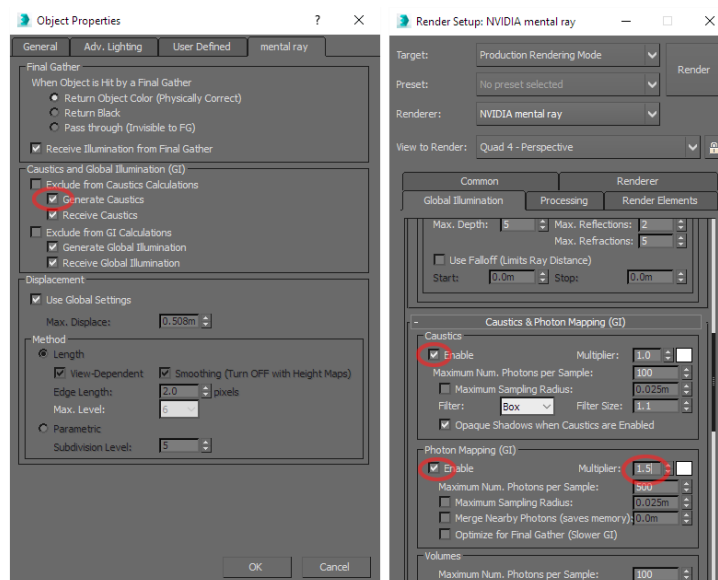


Figure 6: Inställningar för att rendera Caustics

3 Uppgifter att redovisa

- Studera avsnittet om ljus och skuggor, och rendera en bild av din testscen med diffust reflekterande objekt så att den får mjuka skuggor från en arealjuskälla. Rendera också med en Skylight, gärna i kombination med en riktad ljuskälla, om du så vill med simulerat dagsljus från ett *Daylight System*. Rendera i så fall några olika bilder för att visa hur det ser ut vid olika tider på dygnet.
- Rendera bilder som du är nöjd med av glaset du skapade, dels med ditt egna Standard-material och dels med det färdiga glasmaterialet och caustics. Bilden ska vara rimligt realistisk både vad gäller reflexion och brytning, men den ska inte behöva ta en timme att rendera. Använd en enda riktad ljuskälla, förslagsvis en av typen *mr Area Spot*. Kom ihåg att kryssa i rutan *Generate Caustics* i *Object Properties* för både ljuskällan och glasmaterialet: för ljuskällan för att Photons ska skapas överhuvudtaget, och för glasmaterialet så att det sprider dem vidare i scenen.

Bilderna redovisar du under laborationen till labassistenterna på labtillfället. Lycka till, och fråga gärna om du vill veta mer, men glöm inte heller bort att du kan läsa mer på nätet och i programmets omfattande hjälpfiler och tutorials. Och inte minst: var inte rädd för att experimentera på egen hand!