

# Korszerű 3D algoritmusok AR/VR alkalmazásokhoz



Tóth Balázs György



NEMZETI  
KÖZSZOLGÁLATI EGYETEM  
BUDAPEST

SZÉCHENYI 2020



MAGYARORSZÁG  
KORMÁNYA

Európai Unió  
Európai Szociális  
Alap



BEFEKTETÉS A JÖVŐBE

OKOSVÁROS-TECHNOLÓGIÁK  
A technológia fejlődésének irányai és hatása  
XI. kötet

**Sorozatszerkesztő:**

Sallai Gyula

Tóth Balázs György

# KORSZERŰ 3D ALGORITMUSOK AR/VR ALKALMAZÁSOKHOZ



Nemzeti Közsolgálati Egyetem  
Közigazgatási Továbbképzési Intézet  
Budapest, 2020

A kötet a Nemzeti Közszerológálati Egyetem KÖFOP-2.1.2-VEKOP-15-2016-00001 „A jó kormányzást megalapozó közszológálat-fejlesztés” projektje keretében, a Budapesti Műszaki és Gazdaságtudományi Egyetem Villamosmérnöki és Informatikai Karán létesült „Okos város – okos közizgázgatás” kutatóműhelyben (2017/162/BME-VIK) készült.

**Szerző:**

Tóth Balázs György, egyetemi docens BME-VIK

**Szakmai lektor:**

Kiss Bálint, tszv. egyetemi docens, BME VIK

**A kézirat lezárásának dátuma:**

2018. szeptember 26.

Kiadja:

© NKE, 2019

Felelős kiadó:

Prof. Dr. Kis Norbert  
dékán

© Nemzeti Közszerológálati Egyetem  
Közizgázgatási Továbbképzési Intézet, 2020  
© Dr. Tóth Balázs György, 2020

A mű szerzői jogilag védett. Minden jog, így különösen a sokszorosítás, terjesztés és fordítás joga fenntartva. A mű a kiadó írásbeli hozzájárulása nélkül részeiben sem reprodukálható, elektronikus rendszerek felhasználásával nem dolgozható fel, azokban nem tárolható, azokkal nem sokszorosítható és nem terjeszthető.

# TARTALOM

<b>1. BEVEZETÉS</b>	<b>6</b>
<b>2. ÁRNYALÁS</b>	<b>8</b>
2.1 Lokális illumináció	10
2.2 Árnyékok	14
2.3 Kép alapú megvilágítás	17
2.4 Fizikai alapú árnyalás	21
<b>3. INVERZ KÉPSZINTÉZIS ÉS ÚJRAFÉNYELÉS</b>	<b>25</b>
3.1 Környezeti megvilágítás visszaállítása	25
3.2 HDR képalkotás	27
3.3 Geometria visszaállítása	28
3.4 Újrafényelés	30
<b>4. KÉPTÉRBELI HATÁSOK</b>	<b>31</b>
4.1 Tone mapping	31
4.2 Csillanások, ragyogás	34
4.3 Mélységélesség	37
4.4 Bemozdulásos életlenség	40
4.5 Árnyéksávok	42
4.6 Képtérbeli közvetett megvilágítás	45
<b>5. NEM-FOTOREALISZTIKUS MEGJELENÍTÉSI ALGORITMUSOK</b>	<b>50</b>
5.1 Cel Shading	50
5.2 Lit Sphere Mapping	52
5.3 Kontúrok	53
5.3.1 Képtérbeli kontúrozás	54
5.3.2 Objektumtérbeli kontúrozás	57
<b>6. KOMPOZITÁLÁS</b>	<b>61</b>
6.1 Szín alapú kompozitálás	61
6.2 Gradienstérbeli kompozitálás	63
6.3 Pontfelhő beillesztés	64
<b>IRODALOMJEGYZÉK</b>	<b>66</b>

# 1. BEVEZETÉS

Napjainkra a számítógép segítségével készített képek és képfolyamok az élet minden területén alapvető fontosságúak lettek. A legtöbb mozifilm ma már elképzelhetetlen lenne számítógéppel előállított effektusok nélkül, sőt nem ritkák a teljes egészében számítógéppel készített filmek sem. Természetesen a művészeti és szórakoztató célú felhasználás mellett ipari környezetben is nagyon gyakran elképzelhetetlen a munkavégzés számítógéppel készített képek nélkül. A mérnöki tervezésben a tervrajzok elkészítésén kívül gyakran szükség van látványtervek, háromdimenziós modellek készítésére is. Az orvosi diagnosztikában a betegekről készített felvételek kiértékelését vagy a modern műtéti beavatkozások megtervezését is számítógépes megjelenítéssel támogatják. A modern városokban a közlekedés tervezésében vagy a város arculatának kialakításában is alapvető szerep jut az áttekinthető és világos vizuális megjelenítésnek. A példákat hosszan lehetne még sorolni, de a bemutatott területek is világosan mutatják, hogy a számítógéppel támogatott képelőállítás lehetőségeit érdemes ismerni és szem előtt tartani a mindennapi munka során.

A virtuális valóság és kiterjesztett valóság eszközök egyre szélesebb körű elterjedésével a számítógépes grafika területe még hangsúlyosabb lett, hiszen ezekkel az eszközökkel még inkább bele tudjuk helyezni a szemlélőt a bemutatott virtuális világba. A virtuális valóság rendszerek mára képesek a teljes érzékelési spektrumot lefedni. Ez, a látvány mellett, akár térben elhelyezett hangforrásokat, a felhasználó mozgásának követését, illetve akár fizikai visszahatást is jelenthet, amelyek segítségével teljesen új távlatok nyílnak meg számunkra. A kiterjesztett valóság rendszerek ezzel szemben a felhasználó által a valóságban érzékelt környezetet bővítik ki, elsősorban a látvány tekintetében. A kiterjesztett környezetben egy sor olyan információt el lehet helyezni, amelyek segítségével hatékonyan támogatható a munkavégzés. A kiterjesztett valóság eszközöket előszeretettel alkalmazzák összetett szerelési műveletek támogatására, ahol a felhasználónak szüksége van mind a két kezére, így az utasítások vagy a tervrajz manuális követésére nincs lehetőség. Ennél is fontosabbak azok a felhasználási területek, ahol a munkavégzés során szabad szemmel nem látható körülményeket is figyelembe kell venni, például a föld alatt található csatornák és vezetékek elhelyezkedését. Ezen túlmenően elképzelhetők akár a levegő hőmérsékletét és az esetlegesen jelen levő egészséget károsító anyagokat megjelenítő rendszerek, amelyek nagy segítséget jelenthetnek karbantartási vagy mentési munkák esetén is. Ezen interaktív rendszerek esetén a kényelmes felhasználás szempontjából nagyon fontos, hogy a számítógéppel előállított képek pontosan kövessék a felhasználó mozgását, és az esetleges interakciók során ne térjenek el nagyon a fizikai világban megszokottaktól. Ellenkező esetben e rendszerek használata feleslegesen terheli meg a szervezetet, és szélsőséges esetben egy sor olyan mellékhatással járhat, amely megakadályozza a hosszantartó használatot.

Jelen kötetben elsődleges célunk a számítógépes grafika azon területének bemutatása, amelyet az interaktív rendszerek esetén sikerrel alkalmazhatunk. Ezekben a rendszerekben a virtuális világról készített képek előállítására igen szigorú időbeli korlátokat kell szabnunk, különben a mozgás darabos lesz, amely nagyban rontja a felhasználói élményt. A valós idejű képalkotás során egy modern rendszerben mindkét szem számára több millió képpontot kell kiszámítanunk, és másodpercenként legalább hatvanszor megjelenítenünk. A modern grafikus hardverek segítségével képesek vagyunk akár több millió geometriai primitívből álló színtereket is valóságként megjeleníteni, mindezt nagy felbontásban és nagy frissítési frekvenciával. Ahhoz, hogy a képelőállítási folyamatról teljes képet kapjunk, először áttekinthetjük azokat a modelleket, amelyekkel leírhatók

a virtuális világban található felületek és az azokat megvilágító fény közötti kölcsönhatások. Az *Árnyalás* című fejezetben bemutatjuk azokat az egyszerűsítéseket, amelyekre építve a modern grafikus hardver képes nagyszámú objektumot nagy felbontásban és valós időben feldolgozni és megjeleníteni. A valósághú képek előállításához az egyes felület elemek színének kiszámításán túl szükséges a tárgyak közötti takarás miatt kialakuló árnyékok közelítése is, amely nagyban hozzájárul a térérzet kialakulásához. A lokális illuminációs modell ezen árnyékok közvetlen megjelenítését nem teszi lehetővé, ezért azt ki kell egészítenünk valamilyen közelítő módszerrel az árnyalás során. Az egyszerűsített virtuális fényforrások mellett a megvilágítás követésére a valós világról készült felvételeket is felhasználhatunk, amely alapvető jelentőségű abban az esetben, ha számítógéppel készített képrészleteket szeretnénk elhelyezni valódi világról készült felvételeken.

A virtuális világról készült, a lehetőségekhez képest valósághú képek előállítása mellett gyakran szükség lehet a probléma megfordítására is, azaz a valódi világról készült képek alapján következtetni a felvételeken látható tárgyak geometriai felépítésére, azok egymáshoz képesti elhelyezkedésére, esetleg a felvételek alatti fényviszonyokra is. Az *Inverz képszintézis és újrafényelés* című fejezetben bemutatunk néhány alapvető módszert a nagyvonalú környezeti megvilágítás meghatározására, illetve a geometria sztereó képek alapján történő helyreállítására. A képekből meghatározott információk alapján megmutatjuk, hogy miként lehet megváltoztatni a képek megvilágítási viszonyait, azaz a képeket újrafényelni annak érdekében, hogy jobban illeszkedjenek a virtuális világról készített felvételeinkhez.

A képtérbeli hatásokról szóló fejezetben részletesen bemutatjuk azokat a módszereket, amelyek segítségével egy sor olyan jelenséget közelíthetünk hatékonyan, amelyeknek a fizikai világ működését pontosan lemásoló megoldása alkalmatlan lenne valós idejű felhasználáshoz. Bemutatjuk a nagy dinamika tartományú képelőállítás során alkalmazható leképzési eljárásokat, amelyek az emberi szem működését figyelembe véve képesek hozzáigazítani a látványt a megjelenítő eszközök képességeihez. A szemlélő által érzékelt látvány hihetőségéhez nagyban hozzájárulhat a valódi kamerák fizikai felépítéséből és azok tökéletlenségéből adódó jelenségek képekre gyakorolt hatása, ezért bemutatunk néhány egyszerű módszert, amelyekkel közelíteni lehet a kamera lencséinek becsillanását, a mélységélesség hatását, illetve a véges képrögzítési idő által okozott elmosódásokat. A levegőben található szennyeződések hatására létrejövő ragyogások és árnyéksávok is sikeresen közelíthetők képtérben, amelyek bemutatása is helyet kapott a fejezetben. Végül a másodlagos, azaz a felületekről visszavert fény hatására kialakuló, megvilágítás valós idejű közelítésére is mutatunk algoritmusokat. A bemutatott képtérbeli eljárásokon túl bemutatjuk a hatékony megvalósításhoz vezető elméleti háttérrel is.

A valósághú képek előállítása mellett bizonyos esetekben célravezetőbb lehet valamilyen stilizált megjelenítés alkalmazása, amely segít elkülöníteni és hangsúlyozni a számítógéppel készített elemeket. A *Nem-fotorealisztikus megjelenítési algoritmusok* című fejezetben bemutatjuk a stilizált megjelenítés alapjait. A stilizált árnyalási módok mellett a virtuális tárgyak külső és belső kontúrjainak hangsúlyozására is mutatunk hatékony módszereket.

Az utolsó fejezetben a számítógéppel készített képek és a valódi világról készített felvételek összeillesztésének alapjait tekintjük át, amely kiterjesztett valóság alkalmazások esetén elengedhetetlen. Bemutatjuk az egyszínű, általában zöld háttér előtt készült felvételek esetén alkalmazható háttér kilyukasztási módszert, a pontfelhő alapú illesztést és az illesztendő képek gradiensét felhasználó módszert is.

Bízunk benne, hogy e könyv elolvasásával az olvasó áttekintő képet kap a virtuális és kiterjesztett valóság rendszerekben alkalmazott valós idejű képelőállító algoritmusokról, és kedvet kap a további elmélyülésre a területen. Ehhez kiindulásként a magyar nyelven megjelent *Háromdimenziós grafika, animáció és játékfejlesztés*<sup>1</sup> című könyvet ajánljuk.<sup>2</sup>

<sup>1</sup> Szirmay-Kalos 2004.

<sup>2</sup> A szerző: Dr. Tóth Balázs György, a BME Villamosmérnöki és Informatikai Kara Irányítástechnika és Informatika Tanszékének egyetemi docense. PhD fokozatát a BME-en 2014-ben szerezte. Kutatási területe a valós idejű számítógépes grafika és a grafikus hardver általános célú számításokra történő alkalmazása.

## 2. ÁRNYALÁS

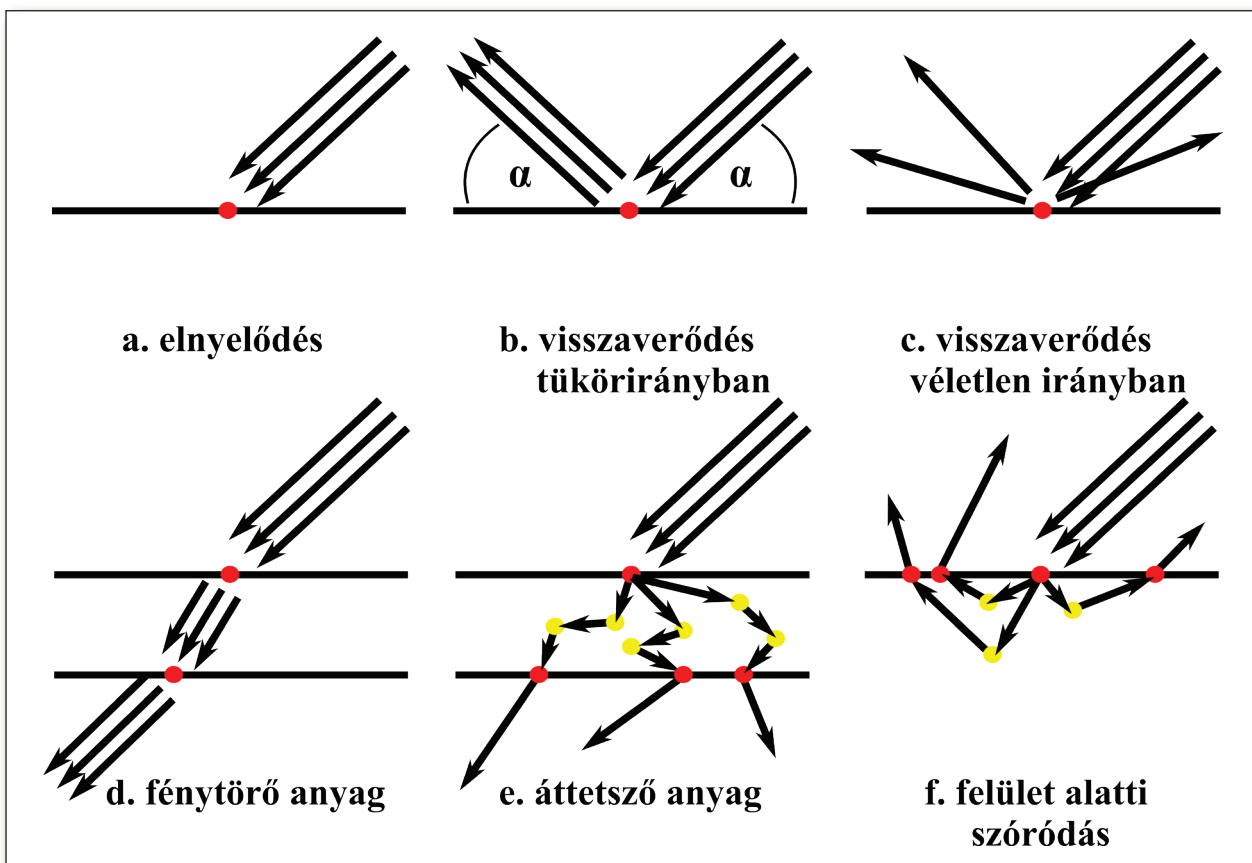
A virtuális világ valóság-hű megjelenítésének egyik kulcsfontosságú komponense az árnyalás. Az árnyalás során a fény terjedését próbáljuk meg utánozni. A valós világban a fényforrások fotonok milliárdjait bocsájtják ki magukból, melyek a felületekbe ütközve vagy elnyelődnek, vagy valamilyen irányba tovább pattannak és folytatják útjukat. A szemünk által látott kép valójában azon fotonok hatásából keletkezik, melyek végül a szemünkbe jutottak. Ha belegondolunk, ezek a szembe jutó fotonok csak egy igen kicsi százalékát teszik ki az összes fotonnak, a többi sok pattanás után végül elnyelődik, végső esetben valahol a világűrben. Ahhoz, hogy a szemünk érzékelni tudja a fényt, rengeteg fotonnak kell bejutnia a retinára, és a fotonok számával arányosan érzékeljük a látott fény világosságát (fényerejét).

A virtuális világ valóság-hűsége nagyban függ attól, hogy a valós világ fényjelenségeit milyen hűen tudjuk utánozni. A fotonok valós fényterjedése igen komplex folyamat. A komplexitása részint a fotonok nagy számában rejlik (egy fényforrás a másodperc töredéke alatt többmilliárd fotont is képes kibocsájtani), részint pedig a fotonok és a felületek kölcsönhatásának bonyolultságában. A felületre beérkező fotonok egy része a felület jellemzőitől függően elnyelődhet, vagy visszaverődhet. A visszaverődés történhet tükör irányban vagy tetszőleges irányban. Az is megeshet, hogy a foton behatol a felületbe, majd később kilép onnan, és folytatja útját. Ennek legegyszerűbb példája az üvegekben történő fénytörés, amikor is a foton behatol az üveg testbe, az anyag törésmutatójának megfelelően megváltoztatja irányát, majd a test túlsó oldalán kilép, újból megváltoztatva a haladási irányát. Mi a szemünkkel az üveg testen keresztülhatolt fotonokat érzékeljük és az üveg által torzított képet látjuk (a torzítás a fényirány megtörése miatt lép fel). Áttetsző anyagoknál, mint a tejüveg vagy a viasz, a testen belül több, szabálytalan irányú fénytörés is bekövetkezhet, ami elmossa a látott képet. Az is jellemző, hogy a felület alá behatoló fotonok a behatolás pontjának közelében lépnek ki a felületből és szabálytalan irányban folytatják útjukat. Ilyen anyag az emberi bőr is, ezért érezzük olyan lágynak, puhának a felületét szemben például a műanyagokkal.

A 2.1. ábra a fény és felület kölcsönhatásról mutat pár példát. Az ábrán tripla nyíllal jelöltük a felület egy pontjára egy irányból érkező fotonokat, piros ponttal pedig ezt a beérkezési pontot. Az a jelű kép mutatja a legegyszerűbb esetet, amikor minden foton azonnal elnyelődik az anyagban. A fotonok energiája ilyenkor általában hővé alakul. A valóságban ilyen anyagot nehéz találni, a fekete lyukak viselkednek így, illetve már létezik olyan szénszálászövetanyag (Vantablack), mely közel minden fotont elnyel. A b jelű képen az összes beérkező foton ugyanabba az irányba verődik vissza a felületről, a tökéletes tükörirányba (ahol a beesési szög és a kimenő szög egyforma). Ilyen anyag például egy tükör. A c jelű ábrán az egy irányból érkező fotonok véletlen visszaverődési irányban távoznak a felületről. A környezetünkben található legtöbb anyag így viselkedik, a felület látványát azonban nagyban befolyásolja, hogy az egyes véletlen irányokba a beérkező fotonok mekkora hányadát verik vissza. Erről a tulajdonságról még később részletesebben beszélni fogunk. A d jelű ábrán egy tökéletesen törő felületet látunk, minden, egy irányból jövő foton ugyanabba az irányba tör meg, mikor belép az anyagba és ugyanúgy tör ki, mikor kilép az anyagból. Az anyagon belül nem változtatják meg a haladási irányukat. Ilyen anyag például a tiszta üveg. Az e jelű ábrán az anyagba való belépés után tetszőleges irányt vehetnek a fotonok, az anyagon belül akár többször is irányt változtathatnak (ezt szóródásnak nevezzük), és kilépés után is véletlen irányba indulnak tovább. Ezek az áttetsző



anyagok, melyeken átnézve csak sejtjük mi lehet mögöttük, de részleteket nem láthatunk. Az f jelű kép pedig olyan anyagot mutat be, ahol a fény nem hatol át az anyagon, csak a beérkezési pont közelébe szóródik, és onnan indul tovább véletlen irányba. Ilyen anyag az emberi bőr is.



2.1. ábra Fény-felület kölcsönhatás fajtái.

A valós világban előforduló anyagok jelentős része a fent felsorolt tulajdonságok mindegyikével rendelkezik. Tehát a fotonok egy részét végleg elnyeli, egy részét visszaveri (akár tükör, akár véletlen irányban), az anyag belsejébe jutó fotonok egy része a törési irány szerint folytatja útját, más részük pedig véletlenül szóródik és vagy áthatol az anyagon, vagy a beesési pont közelében hagyja el a felületet. Az anyagok különbözőségét az adja, hogy milyen arányban jelentkeznek a felsorolt jelenségek.

Ismerve az anyagok ezen tulajdonságait, a valós világ fényjelenségei szimulálhatóak, csak a fotonok útját kell végigkövetnünk, míg egy virtuális fényforrásból a virtuális szembe (kamerába) jutnak. Bár létezik ez a megjelenítési módszer – fotonkövetés a neve – nagyon nagy a számításgénye, így a valós idejű virtuális valóság alkalmazások esetében használhatatlan. A valós idejű megjelenítési módszerek jelentős egyszerűsítésekkel élnek, hogy a probléma megfelelő sebességgel legyen megoldható. A valós idejű számításokat nem csak egyszerűbb algoritmusok, hanem az ezek futtatására képes speciális hardverek, a grafikus processzorok (GPU) teszik lehetővé.

A továbbiakban olyan megjelenítési módszerekről beszélünk, melyek valós idejűek, és a grafikus hardverek támogatják őket.

## 2.1 LOKÁLIS ILLUMINÁCIÓ

A valós világ fényjelenségeit leíró modellek pontos számításában problémát okoz az, hogy egy felületre, egy adott pontba beérkező fénysugár fényereje nem független magából a pontból távozó fotonoktól. Más szóval, mikor ki szeretnénk számítani, hogy egy felületi pontból egy adott irányba mekkora fény mennyiség távozik, számba kell vennünk a felületre érkező fénysugarakat, és rá kell jönnünk, hogy ezek nem függetlenek az éppen számolt értéktől. Ez azért van, mert a felületet elhagyó fotonok pár visszaverődés után akár vissza is juthatnak ugyanerre a felületre. Egy felületi pont fényessége tehát függ minden más felületi pont fényességétől és még önmagától is.

A hatékony megjelenítés érdekében egy drasztikus egyszerűsítéssel élhetünk: a felületek árnyalásának számításakor eltekintünk minden más felületről érkező fotonoktól és csak a fényforrásokból közvetlenül érkező fotonokkal törődünk. Ilyenkor lokális illuminációról beszélünk, mert csak a pont saját jellemzőit vesszük figyelembe a számításakor. Fontos megjegyezni, hogy még azt sem tudjuk figyelembe venni, hogy bármilyen más felület útját állja-e a fényforrásból a felület felé haladó fotonnak. Így sajnos az árnyékok megjelenítéséről is le kell mondanunk. Cserébe egy nagyon hatékony megjelenítést kapunk, mely a grafikus hardverek által maximálisan támogatott.

A lokális illumináció esetében minden a kamerából látott felületi pontot, más néven árnyalt pontot egymástól függetlenül értékelünk ki, így nagyfokú párhuzamosság érhető el. Minden felületi pont esetében végiglátogatjuk a virtuális világban található fényforrásokat, és a felület anyagjellemzőjének segítségével kiszámítjuk, hogy a fényforrásokból érkező fénysugarakból mekkora fény mennyiség távozna a virtuális kamerába. Az egyes fényforrások hatásait egyszerűen összegezzük, és ezt írjuk be a végső kép azon pixelébe, amelyben a felületi pont látszik.

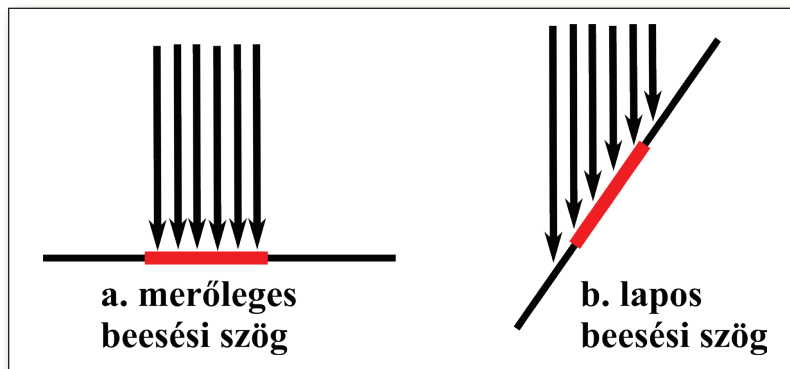
A fent vázolt algoritmusban minden fényforrásra egy számítást végzünk, minden fényforrásból csak egy bejövő fénysugárral számolunk. A valóságban a fényforrások valamekkora mérettel rendelkeznek, melynek következtében több pontjukból is érkehetnek az árnyalt pontba fotonok. A lokális illumináció ettől eltekint, és absztrakt fényforrásmodellekkel dolgozik. A fényforrást vagy végtelenül picinek, ezek a pontszerű fényforrások, vagy végtelen távolinak tekintjük, ezek pedig az irányfényforrások. Pontszerű fényforrással közelíthetünk egy villanykörtét, míg irányforrással a napot. Ez az egyszerűsítés nem teszi lehetővé a lágy megvilágítást adó nagy területű fényforrások szimulációját. Fontos még megjegyezni, hogy a figyelembe vett fényforrások száma is korlátozott. Ez nem jelent korlátozást a virtuális világ fényforrásainak a számára nézve, csupán nem az összes fényforrást fogjuk használni egy pont árnyalásakor, hanem egy virtuális objektum esetén összegyűjtjük a hozzá legközelebbi fényforrásokat, és ezek hatásait vesszük csak figyelembe. Mivel az irányfényforrások végtelen távoli fényforrásokat írnak le, ezeket mindig előtérbe helyezzük a pontszerű fényforrásokkal szemben.

Egy másik fontos kérdés, hogy miként határozzuk meg azt, hogy egy adott irányból bejövő fénysugárnak a teljesítményéből mennyi fog a kamera felé tovább haladni. A fény és felület közötti kölcsönhatásoknál láthattuk, hogy a felületek igencsak komplex viselkedést mutathatnak. A lokális illumináció esetében azonban csak lokális adatokból dolgozhatunk, így nem számíthatók olyan jelenségek, ahol a testbe történő beesési pont és a testet elhagyó kimenő pont nem ugyan az. Így csak elnyelődéssel és visszaverődéssel tudunk törődni, nem számíthatunk töréseket, anyagon belüli és felület alatti szóródásokat.

A fényelnyelés az anyag színével áll szoros összefüggésben. Ha egy felületet például zöldnek látunk, akkor az a felület a zöld hullámhossztartományba eső fotonokon kívül mindent elnyel, azaz csak a zöld színeket veri vissza. A felület színét pedig megadhatjuk objektumonként, vagy ha egy objektum felületén színmintázatot szeretnénk megjeleníteni (ami a legtöbb esetben igaz), akkor használhatunk textúrákat a felületi pontok színének (így fényelnyelő tulajdonságának) meghatározásához.

Azok a fotonok, melyek nem nyelődtek el, visszaverődnek. A visszavert fény erőssége a különböző irányokban

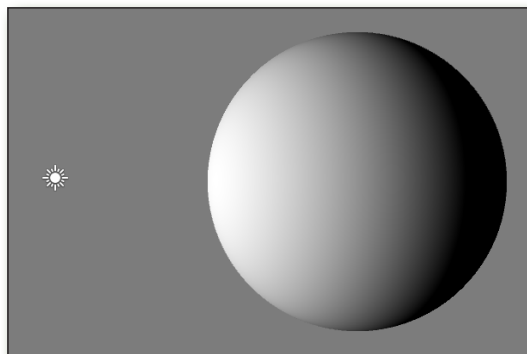
fontos felületi jellemző, melyet igyekszünk könnyen kiértékelhető matematikai formulákkal kezelni. Ilyen matematikai formulák a kétirányú visszaverődés eloszlás-függvények (Bi-directional Reflectance Distribution Function), röviden BRDF-ek. Ezen függvények bemenetei a beesési szög, a kimenő szög, a beeső fényerősség, és kimenetükön a kimenő fényerősséget adják. Számos ilyen BRDF modell létezik, melyek különböző anyagokat tudnak jól modellezni, és változó a számítási komplexitásuk, de két modellt nagyon gyakran használnak, így ezekről ejtünk pár szót.



2.2. ábra Lambert törvény.

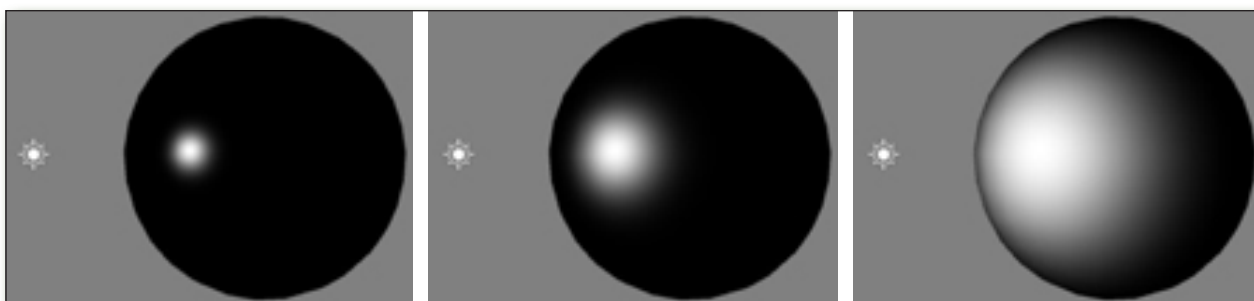
Az első BRDF modell a diffúz vagy Lambert modell. Ez olyan anyagokat modellez, melyek esetében a visszaverődés iránya teljesen véletlenszerű, nem függ a beesési szögtől és a kimenő szögtől. Ez azt jelenti, hogy minden irányban ugyanolyan fényerősséggel sugároz vissza a felület. A felületen azért látunk mégis tónuskülönbségeket, mert a beesési szög a felületre érkező fény mennyiségét befolyásolja. Ez a Lambert törvény miatt van: minél laposabb szögből érkezik a beeső fénysugár, annál nagyobb felületen oszlik el, így az egységnyi felületre annál kevesebb fényerősség jut. Ezt szemlélteti a 2.2. ábra, ahol a bal oldalon merőleges beesési szög mellett hat nyíllal jelöltük a vizsgált felületdarabra (piros szakasz) érkező fény mennyiségét. A jobb oldali ábrán lapos beesési szög mellett ugyanakkora felületelemre már csak négy nyíl mutat, tehát a ráeső fény mennyiség csökkent.

A diffúz felületek matt felületek, és a BRDF modellnek a felület színén kívül nem is kell más paramétert megadnunk. Ilyen felület például a papír, vagy a falfelületek, de sok szövet, például a pamut is diffúz felületű. Diffúz felületre mutat példát a 2.3. ábra.

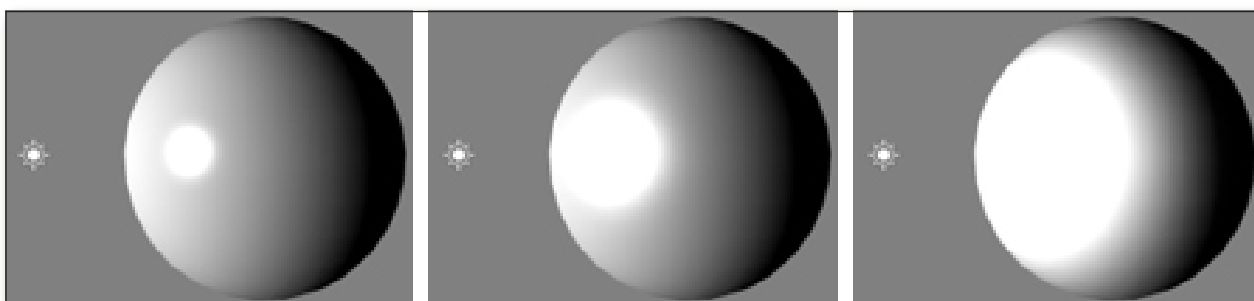


2.3. ábra Diffúz felület (a pontszerű fényforrást a stilizált napocska mutatja).

A másik, igen gyakran használt BRDF modell a Phong modell, mely csillogó felületű anyagok megjelenítésére használható. Ez a BRDF azt feltételezi, hogy a visszavert fénysugarak szeretnek a tökéletes visszaverődési irány közelében maradni. A Phong modellnél csillanás mértéke határozza meg azt, hogy mennyire térhetnek el a tüköriránytól a visszavert fénysugarak. A BRDF modellnek ez a fontos paramétere az úgynevezett shininess. A 2.4. ábra és a 2.5. ábra csillogó anyagú gömböket mutat be különböző shininess paraméterekkel (a paraméter értéke balról jobbra csökken). Fontos megjegyezni, hogy a legtöbb anyag esetében általában mindkét anyagmodellt, a Lambert és a Phong modellt is számítunk egyszerre (2.5. ábra), és hatásaikat összegezve jelenítjük meg. A 2.4. ábra esetében csak a Phong modell által számított hozzájárulásokat láthatjuk (ezek a tökéletes tüköriránynak környezetébe visszavert fénysugarak), míg a 2.5. ábra a diffúz modellel kombinált eredményt mutatja. Tehát a csillogó – más szóval spekuláris – modellek csak a csillanás fénykörét számítják, ehhez még általában hozzáadjuk a diffúz visszaverődéseket is.

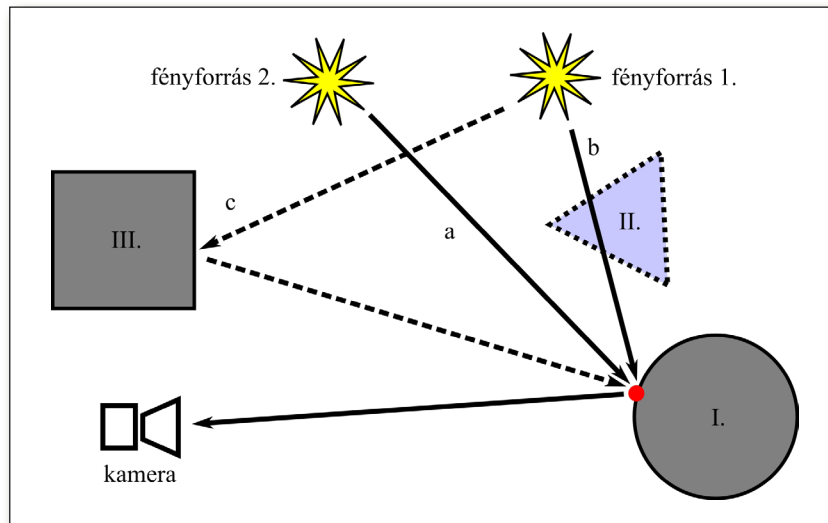


**2.4. ábra Phong BRDF hatása különböző csillogási paraméterrel rendelkező csillogó anyagok esetén.**



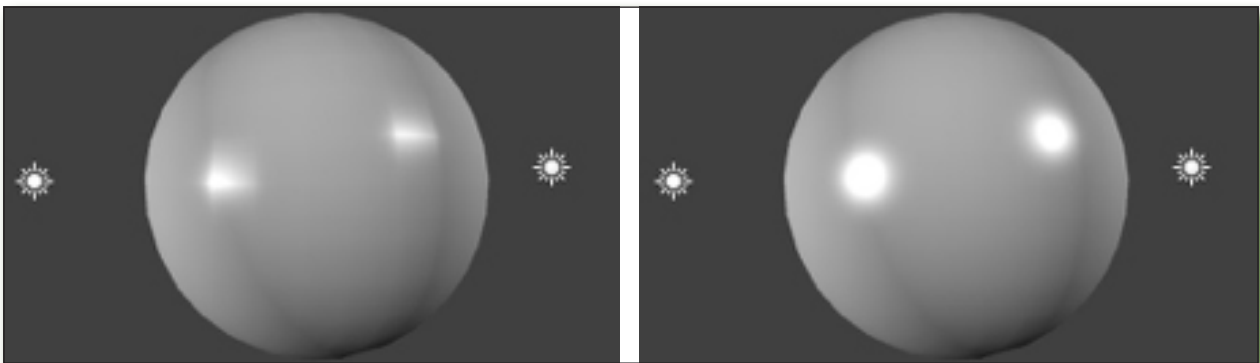
**2.5. ábra Phong és a diffúz BRDF kombinált alkalmazása különböző csillogási paraméterrel rendelkező csillogó anyagok esetén.**

Összefoglalásként, a 2.6. ábra jelöléseit használva elmondhatjuk, hogy a lokális illuminációs módszer során fix számú absztrakt fényforrás hozzájárulását összegezzük. A fényforrásokból minden árnyalt pontba csak a közvetlen fénysugarak (a. és b.) vehetők figyelembe más objektumok takarásától (II.) eltekintve. Az ábrán jelzett a. és b. fénysugarak tehát teljesen azonos módon kezeltek, függetlenül a II. objektum takarásától. Közvetett, több visszaverődés után a kamera felé jutó fénysugarak (c.) szintén nem számíthatók. Ezt az árnyalási modellt direkt árnyalási modellnek is nevezzük, szemben a többszörös visszaverődést is számító indirekt megvilágítási módszerekkel.



**2.6. ábra Lokális illumináció. A kamerából látott árnyalt pont (piros kör) fényességét az absztrakt fényforrásokból közvetlenül beérkező sugarak (a és b) fényességéből számítjuk. A fénysugár takarását más objektumok által nem tudjuk figyelembe venni (b sugár és II. objektum). Közvetett fénysugarak (c) számítása nem lehetséges.**

A lokális illuminációs modell minőségét nagyban befolyásolja az is, hogy a BRDF-eket milyen pontokon értékeljük ki. Eddigi leírásunkban azt feltételeztük, hogy minden árnyalt pontban kiértékeljük ezeket a függvényeket. Ezt a megoldást pixel alapú árnyalásnak nevezzük (per pixel shading, vagy Phong árnyalás, utóbbi nem keverendő össze a már bemutatott Phong BRDF-el). Bár ez vezet a jobb minőségű eredményhez, de költséges lehet. Megtehetjük ugyanakkor azt is, hogy a háromszög geometriánk csúcspontjaiban értékeljük csak ki a BRDF modelljeinket és az ott kapott értékeket interpoláljuk a háromszögek belső pontjaiban. Ezt csúcspont alapú árnyalásnak nevezzük (per vertex lighting, vagy Gouraud-árnyalás), mely csak közelítő eredményt ad, azonban jóval hatékonyabb lehet, így kisebb számítási kapacitású mobil platformon vagy a virtuális kamerától távol eső objektumokon érdemes megfontolni a használatát.



**2.7. ábra Csúcspont alapú (bal) és pixel alapú (jobb) árnyalás összehasonlítása.**

A 2.7. ábra a két árnyalás közötti különbséget mutatja be. Mindkét képen ugyanazt a geometriai modellt használtuk, ugyanolyan anyagjellemzőkkel. A bal oldali, csúcspont alapú BRDF kiértékelés nem képes visszaadni a csillanások fényköreit, míg a jobb oldali, csúcspont alapú árnyalás szép kerek csúcspontfényeket és sima árnyalást ad.

A 2.8. ábra egy összetettebb jelenetet mutat, egy fényforrással, lokális illuminációt használva.



**2.8. ábra** Összetett virtuális jelenet megjelenítése lokális illuminációval és pixel alapú árnyalással.

## 2.2 ÁRNYÉKOK

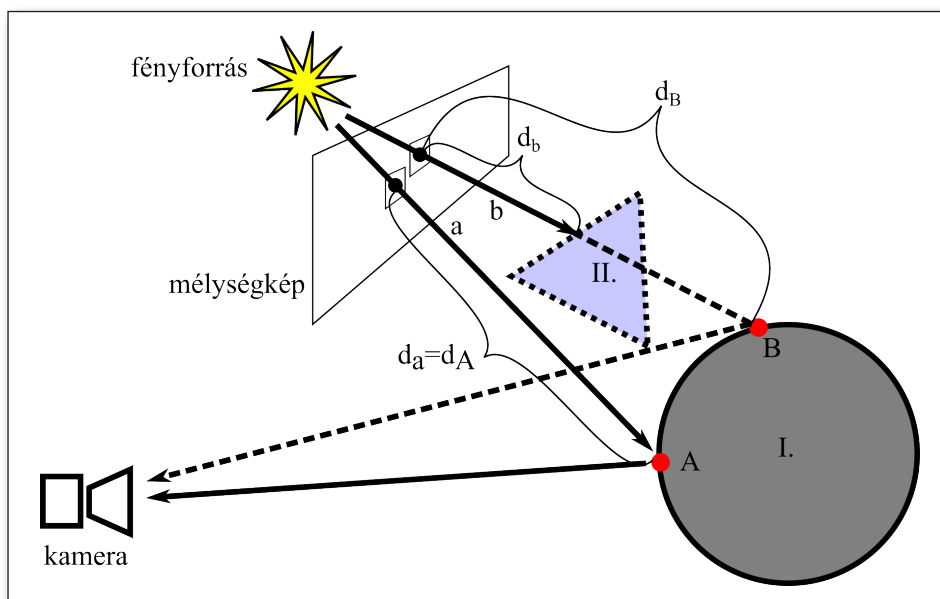
A lokális illuminációt kiegészíthetjük olyan módszerekkel, melyek célja, hogy legalább részben orvosolják annak hiányosságait. Az első hiányosság a takarás figyelmen kívül hagyása. A lokális illumináció során nincs információnk más felületekről, így a takarási információ hiánya miatt vetett árnyékok sincsenek. Árnyékok megjelenítésére több módszer is létezik, melyek közül mi a legelterjedtebb megoldást a mélységkép alapú árnyékolást mutatjuk be. (Depth shadow maps, shadow mapping, vagy depth shadows néven találkozhatunk vele.) Az árnyékok alkalmazását mutatja be a 2.9. ábra.



**2.9. ábra** Összetett jelenet lokális illuminációval és mélységkép alapú árnyékokkal.

Az a mélységkép alapú árnyékolás lényege, hogy egy olyan, úgynevezett mélységképet készítünk a fényforrás szemszögéből, mely színek helyett a látott pontok fényforrásoktól vett távolságát, azaz mélységét tartalmazza. A lokális illumináció számítását azzal egészítjük ki, hogy az árnyalt pont fényforrással vett távolságát összevetjük a mélységképben tárolt értékkel, és ha nagyobb, akkor biztosak lehetünk benne, hogy a fényforrás és az árnyalt pont között egy másik felület van. Ilyenkor az adott fényforrás hozzájárulását nem kell számítani, vagy jelentősen sötétítjük. Ez utóbbi megoldásnak nincs közvetlen fizikai megfelelője, egyszerűen csak sok esetben nem szeretnénk mély, teljesen sötét árnyékokat látni.

A mélységkép alapú árnyékszámítást mutatja be a 2.10. ábra. Első lépésként a fényforrás szemszögéből látható felületelemek fényforrástól vett távolságait eltároljuk egy képbe. Az ábrán ennek a képnek két pixelét jelöltük: ezek az a és b irányoknak megfelelő pixelek. Ezekben a pixelekben a távolságok rendre  $d_a$  és  $d_b$ , ez a két érték lesz a két pixelben tárolva. A végső megjelenítés során az árnyalt pontokra, melyek az ábrán A és B, kiszámítjuk a fényforrástól vett távolságukat,  $d_A$ -t és  $d_B$ -t. Ismerve a mélységkép készítésekor használt kamera beállításokat megkapjuk, hogy az árnyalt pontok a mélységkép mely pixeleiben látszódnak. Ezekben a pixelekben tárolt mélység információt összevetjük az árnyalt pont távolságértékét, és ha ez utóbbi a nagyobb, akkor felületi pont árnyékban van. Az ábránkon tehát A pontra igaz, hogy  $d_a = d_A$ , így a pont nincs árnyékban, míg B pontra  $d_b < d_B$ , így a B pont árnyékban van.

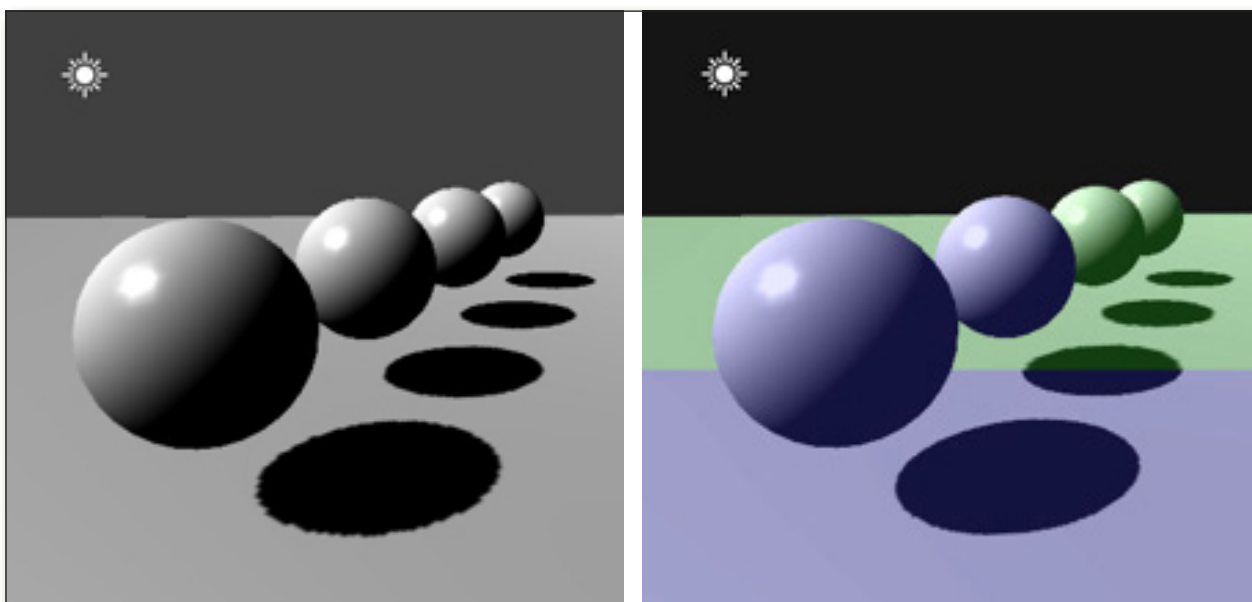


2.10. ábra Mélységkép alapú árnyékolás.

A mélységkép alapú árnyékolás előnye, hogy viszonylag egyszerűen implementálható, a lokális illuminációval jól kombinálható, és a másik népszerű technikával, az árnyéktestekkel ellentétben nem érzékeny a geometriai hibákra. Előnyein túl sajnos rendelkezik pár hátrányos tulajdonsággal is. Az árnyékok minőségét nagyban befolyásolja a mélységképek felbontása. Kis felbontás esetén láthatóvá válnak a mélységkép pixelhatárai. Ezt megfigyelhetjük a 2.11. ábra bal oldalán, ahol a közeli gömb árnyékán egyértelműen feltűnőek a mélységkép pixelhatárai. A távolabbi gömbök árnyékain ez nem annyira észrevehető. Ez azért van, mert az egész jelenetre egy mélységképet használunk, mely pixeleinek mérete az objektumok felületeire vetítve közel azonos. A végső megjelenítéskor használt perspektív torzítás miatt a közeli objektumok nagyítva jelennek meg a távolabbiakhoz képest, így a rájuk vetített mélységkép pixelei is nagyobbak látszanak. Így sajnos az történik, hogy pont azokon az objektumokon lesz rosszabb a minőség, melyek fontosabbak lennének, hiszen közelebb vannak.

Ezen úgy segíthetünk, hogy a közeli objektumokra nagyobb mélységkép felbontást próbálunk meg használni. Ez egy mélységképpel nehezen oldható meg, helyette minden fényforráshoz használhatunk több mélységképet. Ezt a módszert kaszkád árnyékolásnak nevezzük (cascade shadow mapping). A 2.11. ábra jobb oldalán látható kép két mélységképet használ egy fényforrásra: egyet a kamera közvetlen közelében látható objektumokra, egyet pedig a távolabbiakra. A két mélységkép által lefedett területet külön színnel jelöltük. A két mélységkép azonos felbontású, de a közeli objektumokhoz tartozó jóval kisebb területet fed le, így a végső képen nagyobb hasznos felbontást tud nyújtani. Az jobb oldali képen látható árnyékok minősége jelentősen javult a kaszkádok használatával. A jelenet méreteinek függvényében több kaszkádot is bevezethetünk. Ez persze azt jelenti, hogy az árnyalás során ki kell választanunk az árnyalt pontnak megfelelő kaszkádhoz tartozó mélységképet.





2.11. ábra Mélységkép alapú árnyékolás és kaszkád mélységkép használata (jobb).

### 2.3 KÉP ALAPÚ MEGVILÁGÍTÁS

Az mélységkép alapú árnyékolás segített könnyíteni a lokális illumináció egyik hiányosságán, a takarások problémáján. Másik nagy hiányosságként az indirekt fényutak kezelését emeltük ki, ezen segíthetnek a kép alapú megvilágítási technikák (image based lighting). Kép alapú megvilágításnál egy objektum megvilágításához nem absztrakt fényforrásokat határozzunk meg, hanem az objektum környezetének képét mentjük el egy célnak megfelelő struktúrába és ennek segítségével próbáljuk meg a különböző irányból bejövő fény mennyiségeket meghatározni. Ezt a képet környezeti térképnek (environment map) nevezzük.

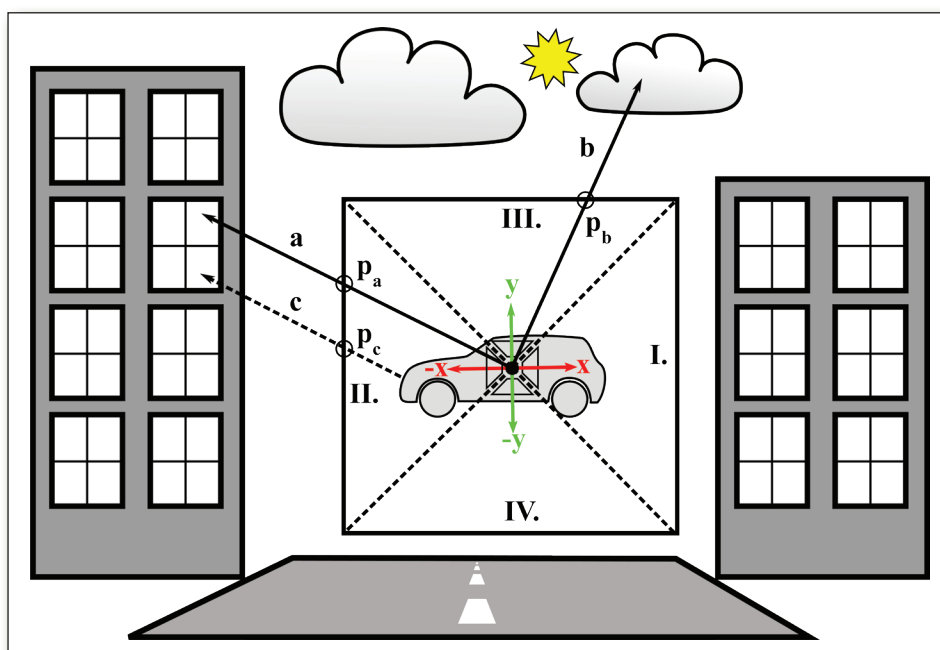
Képek tárolásával sokkal részletesebb megvilágítás érhető el, mint az absztrakt fényforrásokkal, ugyanis a környezetről készült képeken kiterjedéssel rendelkező fényforrások is megjelenhetnek, melyek már nem is feltétlenül egyszínűek, hanem akár változatos színmintázatúak is lehetnek. Sőt, ha ezek a fényforrások által megvilágított környezeti objektumokat is láthatunk a képen, akkor ezek is fényforrásokként fognak megjelenni, tehát indirekt megvilágítást is kezelni tudunk.

Az előzőekben tehát megjelent az objektumok egyfajta csoportosítása: vannak a környezethez tartozó objektumok, melyek a környezeti térképen megjelennek, és vannak azok az objektumok, melyeket ezekkel a képekkel szeretnénk megvilágítani. A gyakorlatban ezt úgy valósítjuk meg, hogy a nagy méretű, állandó objektumokat soroljuk a környezet objektumaihoz, és a mozgó, kis méretű objektumokat (dinamikus objektumok) világítjuk meg ezzel a technikával. A környezet képének előállítását elég költséges, ezért általában egyszer végezzük el, így azon akármilyen részletességgel számíthatjuk az árnyalást, akár egy offline előfeldolgozó lépésben, animációs filmeknél használt, indirekt megvilágítást is kezelni képes algoritmusokkal. Az is gyakran előfordul, hogy a környezeti képekben nem is a virtuális jelenetet tároljuk el, hanem egy ahhoz nagyon hasonló, valós környezetről készült fotót. Az előfeldolgozási jelleg és az objektumok szétválasztásának következtében a környezet esetleges megváltozása és más dinamikus objektumok nem lesznek hatással az árnyalásra.

A környezeti kép alapú megvilágításnál a környezeti kép minden pixele egy fényforrásnak felel meg. Ahhoz, hogy minden kameraállásra és minden felületi irányra képesek legyünk környezeti megvilágítást számolni, a dinamikus objektum környezetének minden irányához meg kell feleltetnünk egy pixelt a környezeti képben. Ezek a képek tehát nem egyszerűen egy virtuális ablakon keresztül látott képek, hanem speciális tárolással rendelkeznek. A leggyakrabban használt környezeti tárolási forma a dobozkép (cubemap). Ez gyakorlatilag hat különböző képet jelent, melyek a háromdimenziós tér hat fő irányába készített síkképeknek felelnek meg.

A 2.12. ábra a doboz környezeti képet szemlélteti két dimenzióban. Az ábrán a környezet egy utca, melyben egy autó halad. Ez esetben az aszfalt, az épületek és az égbolt változatlanak tekinthető, ezek kerülnek a környezeti képbe. Az autó pedig dinamikus objektum, amit kép alapú megvilágítással fogunk megjeleníteni. Az autó középpontjából a kétdimenziós tér négy fő irányába 90 fokos látószöggel 4 képet készítünk, és ezeket helyezük el képzetben egy doboz négy oldalára. Az ábrán ezek az oldalak római számmal jelöltek. Mivel ezek a képek átlapolás nélkül hézagmentesen illeszkednek, a dinamikus objektum középpontjából indított tetszőleges irányú sugárnak pontosan egy oldallal és annak egy pixele felel meg. Így az  $a$  jelű sugárnak a  $II.$  oldallal  $p_a$  pixele, míg a  $b$  jelű sugárnak a  $III.$  oldallal  $p_b$  pixele felel meg. Tehát ha meg szeretnénk tudni, hogy a dinamikus objektum középpontjába a irányból milyen fény mennyiség érkezik csak ki kell olvasnunk a  $II.$  oldallaphoz tartozó kép  $p_a$  pixelét.

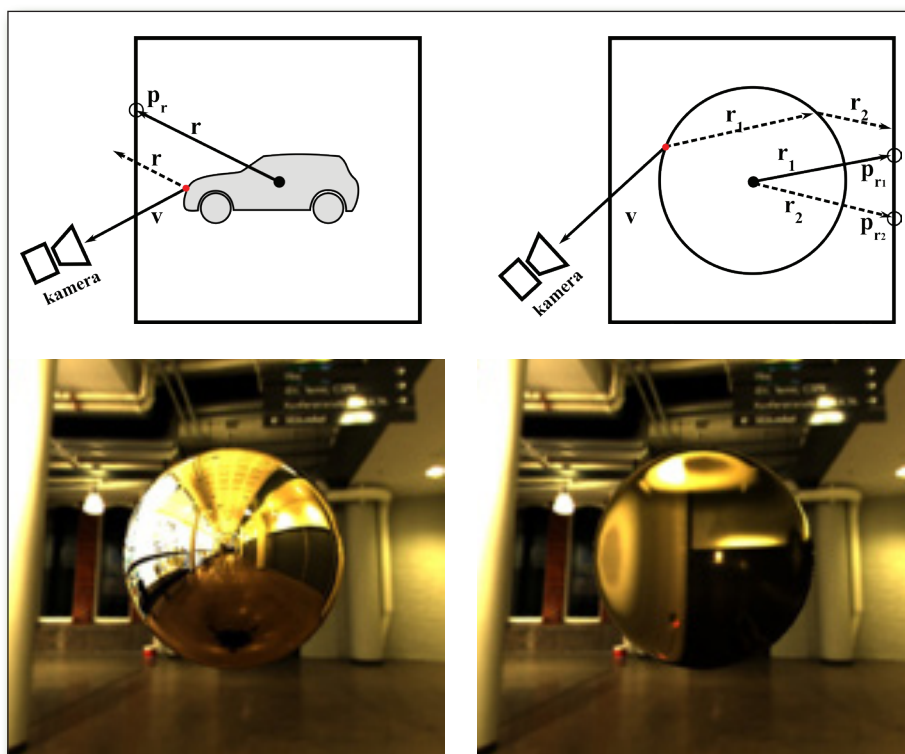
A környezeti térképekből tehát mindig egy térbeli irány szerint olvasunk ki. Szem előtt kell azonban tartanunk, hogy a  $c$  jelű sugár nem a középpontból indul, de  $a$  sugárral párhuzamos így hozzá is a  $p_a$  pixel rendelődik, míg valójában a  $p_c$  pixelből kellett volna olvasnunk. A környezeti képek egyik hátránya, hogy nem veszik figyelembe a sugár kezdőpontját, így csak akkor adnak pontos eredményt, ha a dinamikus objektumhoz képest a környezet végtelenül távolinak (végtelenül nagy) tekinthető. Ezt úgy is megfogalmazhatjuk, hogy a dinamikus objektum végtelenül kicsi, azaz pontszerű, így nincs kiterjedése. A környezet bizonyos elemeire a végtelen távolság teljesül is, például az égboltra, felhőkre, távoli hegyekre. Más elemekre, mint esetünkben a házak és az úttest, viszont nem, így nem kapunk pontos eredményt. Ez a pontatlanság azonban a legtöbb esetben nem feltűnő.



2.12. ábra Környezeti doboz térkép (environment cube map).

A környezeti képek legkézenfekvőbb felhasználása a tükröződések számítása. Tükröződések megjelenítéséhez minden árnyalt pontra kiszámítjuk a nézeti irány tükörirányát és ebből az irányból olvassunk ki a környezeti képből egy pixelt. Az innen visszaolvasott színértéket adjuk vissza árnyalt színeként. Ezt mutatja be a 2.13. ábra bal oldala. Itt is megfigyelhető, hogy az árnyalt pont helyét nem vesszük figyelembe, csak a  $v$  nézeti irány tükörirányát,  $r$ -et használjuk, melyhez  $p_r$  pixel tartozik. Ha az objektum nem kerül közel a környezetéhez, ez nem okoz feltűnő hibákat. A 2.13. ábra jobb oldalán a tökéletes átlátszó objektum fénytörését láthatjuk. Az itt elvégzett számítások sem sokkal bonyolultabbak, csupán a visszaverődési irány helyett a nézeti irány törési irányát számítjuk ki ( $r_1$ ), és az ehhez tartozó pixelt,  $p_{r_1}$  használjuk. Érdekes itt megjegyezni, hogy a módszer nem csak ott hibázik, hogy nem veszi figyelembe az árnyalt pont pozícióját, hanem ott is, hogy csak a testbe való belépésnél számítja törést. A valóságban egy olyan egyszerű objektumnál is, mint a gömb, a fénysugár kétszer is irányt változtat, másodszor a testből való kilépéskor, így az  $r_2$  irányhoz tartozó  $p_{r_2}$  pixel jobb közelítés lenne.

A dinamikus objektum geometriájának elhanyagolása a tükröződések során is lehetetlené teszi az öntükrözés számítását. Érdekes megjegyezni, hogy léteznek valós idejű módszerek az árnyalt pont pozíciójának figyelembe vételére, sőt többszörös tükröződés és törés számítására is. Ezek a módszerek azonban megnövekedett számítási igényeket támasztanak az itt bemutatott alap módszerekhez képest, így ritkán használjuk őket.<sup>3</sup>



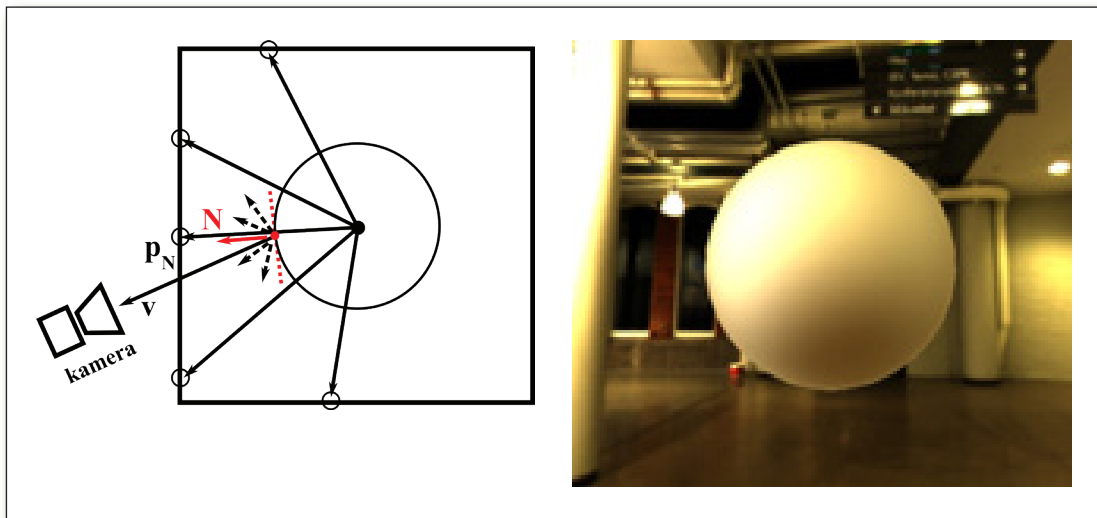
2.13. ábra Tükröződés és törés számítása környezeti képpel.

<sup>3</sup> Szirmay Kalos 2005, Umenhoffer 2007.

Ha nem tökéletesen tükröződő vagy törő objektumról van szó, a helyzet kicsit bonyolultabbá válik. Ezeknél az anyagoknál nem csak egy irány számít, hanem a környezeti kép minden pixele egy fényforrásként fog működni, és mindegyik pixel hatással lesz a kamera irányába haladó fényútra. Tehát az lenne a helyes megoldás, hogy a felületből látható minden pixelre számítsuk ki a felületre beeső fény mennyiségét, és az anyag BRDF modelljének segítségével számítsuk a kamera felé tovább haladó fény mennyiségét. Az összes ilyen pixel hatásának összege adja a végső árnyalt képet. Mivel a környezeti képek több millió pixelt is tartalmazhatnak, ezt túl sok számítást igényelne. Szerencsére a környezeti képeken előkalkulációt végezve egyetlen kiolvasásra szűkíthetjük a szükséges számításokat. Ezt azonban a különböző anyagokra (BRDF modellekre és azok paraméter beállításaira) külön-külön el kell végezni.

A diffúz anyag megjelenítését mutatja be a 2.14. ábra. Az árnyalt ponthoz tartozó „végtelen” számú visszaverődési irány helyett csak a felület normálvektorának irányában ( $N$ ) kell kiolvasnunk  $p_N$  pixelt egy, a diffúz BRDF-nek megfelelően előkészített (elő-konvolvált) környezeti képből.

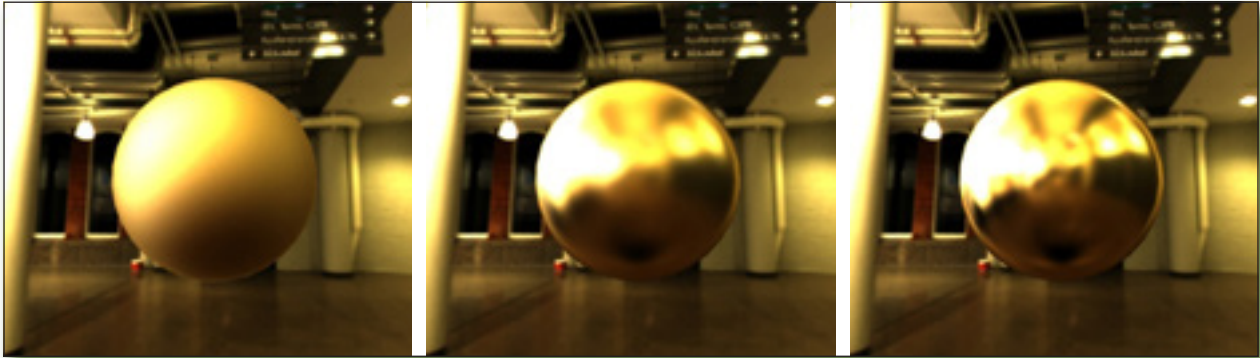
Más BRDF is használható, de annak megfelelően elő kell készíteni a környezeti képet. A gyakorlatban a két legnépszerűbb BRDF, a Lambert és a Phong BRDF-nek megfelelően készítenek elő-konvolvált környezeti kép változatokat. Mivel a Phong BRDF tartalmaz egy shininess paramétert, ezért annak pár jellemző értékére külön elvégzik a számításokat, és valós időben a paraméter konkrét értékének megfelelően lineárisan interpolálják a különböző környezeti képeket.



2.14. ábra Diffúz megvilágítás számítása környezeti képpel.

A 2.15. ábrán a Phong BRDF különböző shininess paraméter értékével megjelenített gömböket láthatunk. A diffúz és Phong anyagoknál szintén csak közelítő megoldást ad a dinamikus objektum pontszerűnek tekintése. A szakirodalomban található megoldásokat a pozíció helyes kezelésére ezen anyagok esetére is, de itt is igaz, hogy a nagy számításigény miatt csak indokolt esetben használjuk őket.<sup>4</sup>

<sup>4</sup> Lazányi 2006.

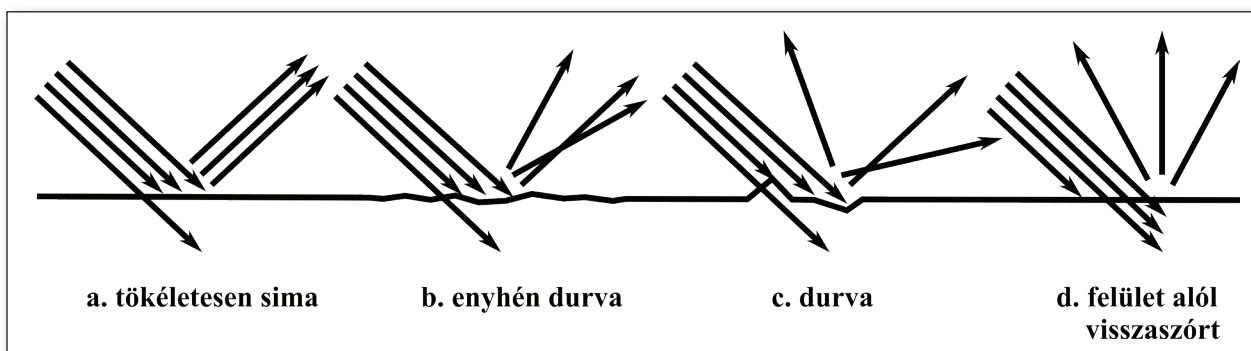


2.15. ábra Phong tükröződés.

## 2.4 FIZIKAI ALAPÚ ÁRNYALÁS

Korábban megmutattuk, hogy az anyagok legtöbbje nem sorolható a diffúz vagy a tükröződő anyagok sorába, mindkét tulajdonsággal bírnak, csak különböző arányban. A környezeti megvilágításnál is megtehetnénk azt, amit a lokális illuminációnál tettünk, azaz összeadhatjuk a diffúz és a Phong modellek hatását. Ez azonban nem felel meg pontosan a fizikai folyamatoknak. Ha visszaemlékszünk a fény-felület kölcsönhatásnál leírtakra, akkor a fotonok többféle utat is bejárhatnak, de egy foton egyszerre csak egy úton indulhat. Tehát visszaverődhet vagy elnyelődhet, esetleg szóródhat, de ezek közül csak egyet tehet meg. A diffúz és a Phong hatás összeadása azt jelentené, hogy egy foton kétszer is visszaverődik: egyszer a diffúz BRDF, egyszer pedig a Phong BRDF szerint. Ezzel energiát viszünk be a rendszerbe, ami túlzottan fényes területeket eredményezhet. Bár ez a probléma nem kötődik csupán a környezeti térképekhez, a lokális illuminációra is igaz volt, a környezeti képekkel történő megvilágítás elterjedésekor kezdtek el vele foglalkozni, így mi is itt említjük.

A problémára a megoldást a fizikai alapú árnyalás (physics based shading, PBS) módszere adja. Szerencsére nem kell teljesen újragondolnunk az árnyalásról alkotott nézeteinket. Összevetve a fény-felület kölcsönhatást korábban illusztráló ábránkkal, a fizikai alapú árnyalást bemutató 2.16. ábra nagy hasonlóságot mutat. A fizikai alapú árnyalás a beérkező fotonokat két csoportja osztja. Az első csoportot a felületről visszaverődő fotonok alkotják, ezeket ábrázoltuk az a, b és c jelű képeken. Azt, hogy milyen irányban verődik vissza egy fénysugár, a felület durvasága befolyásolja. A PBS a mikrofelületek elmélete szerint az objektumok felületét tökéletesen tükröződő apró felüldarabkák összességéként tekinti. Ha ezek a tükröcskék egy síkban vannak, akkor az árnyalt pont környezetében ugyanabba az irányba verik vissza a fénysugarakat, tökéletes tükröződést hozva létre (a). Ha ezeknek a tükröknek az orientációja véletlenszerűen eltér a makrofelület irányától, a fénysugarakat a tökéletes tükörirányhoz képest eltérítik (b és c). Azt, hogy mennyire térnek el a fénysugarak a tüköriránytól, a mikrofelületek irányának szórása, azaz a felület mikroszkópikus szintű rücskössége befolyásolja. A c jelű felületrészlet durvább, rücskösebb, mint a b jelű felület, így jobban szétszórja a fénysugarakat, ami elmosódottabb tükörképet eredményez. Az utolsó, d jelű felületrészlet azoknak a fénysugaraknak a sorsát mutatja, melyek nem verődtek vissza azonnal a felületről. A PBS szerint ezek egyenesen, diffúz jelleggel visszaszóródnak a felület alól, közvetlenül a becsapódás közelében. A PBS tehát nem kezeli a felület alatti szóródást, sem a törést, illetve az áttetsző, fényszóró anyagokat.



**2.16. ábra Fizikai alapú árnyalás alapjai.**

A fentebb leírtak alapján PBS anyagok két fő jellemzővel rendelkeznek: a felület visszaverő képessége és a felület durvasága. Az első azt adja meg, hogy a fénysugarak mekkora hányada verődik vissza a felületről, és nem hatol be, illetve szóródik vissza felület alól. A második pedig a visszavert sugarak visszaverődésének véletlenszerűségét befolyásolja. A visszaverő képesség értéke nulla és egy közé eshet, ahol az egy az összes fénysugár visszaverődését, nulla pedig ez összes fénysugár elnyelődését és visszaszóródását jelenti. A második paraméter, a simaság, szintén nulla és egy között vehet fel értéket. Egy érték mellett tökéletes tükörről beszélünk, nulla esetén pedig teljesen véletlenszerűen verődnek vissza a fénysugarak, ami diffúz visszaverődésnek felel meg. Bizonyos implementációkban a simaság helyett a durvaság paramétert használják, mely szintén nulla és egy értékű, csupán a jelentése fordított: nulla jelenti a tökéletes tükört.

Nem beszéltünk még az anyag színéről. A fotonoknak lehet egy harmadik sorsa is: bejutnak a felület alá és ott végleg elnyelődnek. Ez a fényelnyelési képesség pedig hullámhosszfüggő lehet. Bizonyos anyagoknál a fényvisszaverési képesség paramétert is érdemes a különböző hullámhosszokon függetlenül megadni. Tehát különböző színű fénysugaraknál más és más lehet a felület alá bejutó és az onnan visszaszóródó fotonok aránya.

A PBS tehát nem igényli az eddigi árnyalási számításaink újragondolását. Továbbra is használhatjuk a Phong modellt a csillanások, tükröződések számítására, és a Lambert modellt a felület alól történő visszaszóródásra. A leglényegesebb dolog az, hogy ezek összegzése helyett lineárisan interpoláljuk őket a fényvisszaverő képesség paraméternek megfelelően. PBS-t használhatjuk lokális illumináció BRDF kiértékeléseire, de környezeti kép alapú árnyalásra is, sőt a kettőt kombinálhatjuk is: a felületre beeső fényhatásokat részint környezeti megvilágítással, részint absztrakt fényforrásokkal és lokális illuminációval is számíthatjuk.



**2.17. ábra** PBS megjelenítés különböző paraméterekkel.

A 2.17. ábra PBS technikával megjelenített gömböket mutat különböző anyagtulajdonsággal. Az ábrán balról jobbra haladva a fényvisszaverő képesség (reflectivity) paramétert változtattuk, értékei rendre 0, 0,3, 0,6 és 1,0. Az egyes sorokban a felület durvasága állandó. Felről lefele haladva a simaság paraméter értékei a következők: 1,0, 0,85, 0,6. A bal oldalon lévő képek minden sorban egyformák, hiszen itt a reflectivity zérus, azaz nincs visszaverődés, csak a felület alól visszavert diffúz szóródás. Jól látszik, hogy a visszaverődést növelve a tükröződés egyre nagyobb szerephez jut. A felület durvaságát növelve pedig a tükrökép egyre homályosabb lesz.

A 2.18. ábrán egy összetett jelenetről készült képet láthatunk, amelyen a járművek és az épületek ablakai eltérő fényvisszaverési és simasági paraméterrel rendelkeznek. Jól látható, hogy a jobb oldali járművön a környezet lényegesen jobban visszatükröződik, mint a bal oldalin, köszönhetően a nagyobb fényvisszaverési paraméternek. A képen látható árnyékok a korábban bemutatott mélységkép alapú árnyékszámítással készültek.



**2.18. ábra** Összetett jelenet árnyékokkal és fizikai alapú árnyalással.



## 3. INVERZ KÉPSZINTÉZIS ÉS ÚJRAFÉNYELÉS

Az előző fejezetben képszintézis módszereket mutattunk be, melyek a feladata az volt, hogy a felület geometriai felületi és a virtuális megvilágítás jellemzőinek megfelelően valószerű képet állítsanak elő. Gyakran szükség lehet azonban a feladat megfordítására: adott a végső kép, és onnan szeretnénk visszakövetkeztetni valamely bemenet: geometria, anyag, fényviszonyok jellemzőire. Természetesen itt is a fényterjedés fizikáját használjuk fel és megpróbáljuk például visszakövetkeztetni, hogy milyen fényviszonyok eredményezhették a kiindulási képet. Ha az inverz képszintézis során számított megvilágítást lecseréljük egy másik virtuális megvilágításra, újrafényelésről beszélünk.

### 3.1 KÖRNYEZETI MEGVILÁGÍTÁS VISSZAÁLLÍTÁSA

Az inverz képszintézis során természetesen egyszerre nem tudjuk az összes hiányzó bemenetet visszakövetkeztetni, ezért annak egy részét mindig rögzítjük. Így vagy feltételezzük, hogy a geometria ismert, vagy egy adott BRDF modellből, vagy egyszínű felületekből indulunk ki, esetleg a virtuális fényforrásokat mi magunk adjuk meg.

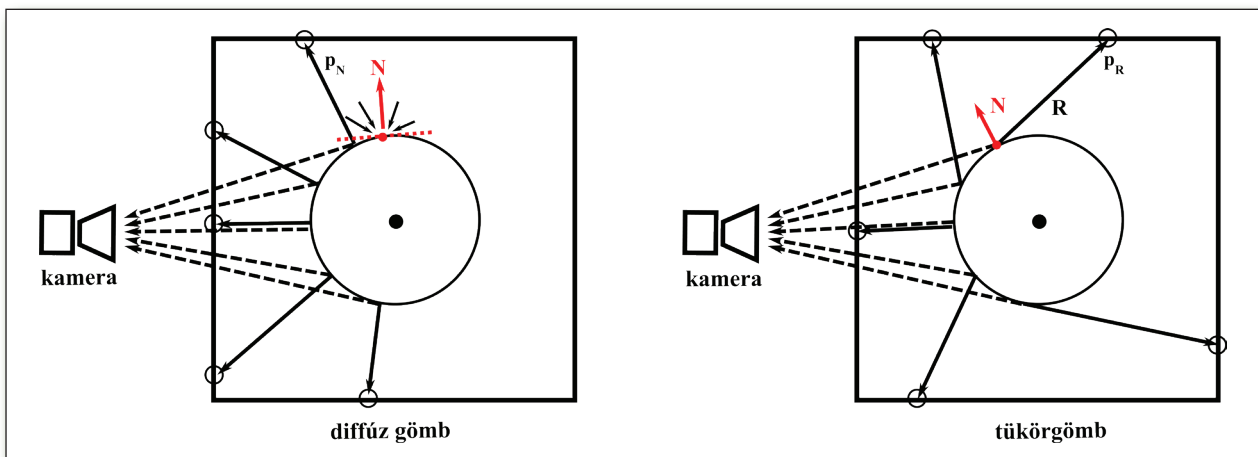
Egy gyakran használt inverz feladat a környezetből jövő megvilágítás számítása. Ez a kiterjesztett valóság alkalmazásoknál egy fontos probléma, ugyanis nem elég létrehozni egy virtuális háromdimenziós objektumot, azt úgy kell megjeleníteni a képernyőn, hogy az a valós világ részének tűnjön. Ehhez pedig elengedhetetlen, hogy megvilágítását a valós megvilágításhoz hangoljuk.

A valós környezet megvilágítását közelíthetjük absztrakt virtuális fényforrásokkal, és a megjelenítést végezzük lokális illuminációval, vagy megpróbálhatjuk a kép alapú árnyalás bemenetétől szolgáló környezeti kép előállítását is. Mindkét esetben kell valami kapaszkodó, hogy a feladatot meg tudjuk oldani, egy valós környezetről készült fotóból nem tudnánk eldönteni, hogy a környezet egyes irányaiban milyen fénysugarak érkezhettek egy virtuális objektumra, ugyanis a környezet geometriáját nem ismerjük. Ha vissza szeretnénk állítani a környezet geometriáját, körbe kell fényképeznünk a helyszínt, és gépi látás módszerekkel visszaállítható a háromdimenziós geometria, melyből egy környezeti kép már könnyen számítható.

A teljes környezet visszaállítása azonban hosszadalmas folyamat és megfelelő szoftvereket, illetve tapasztalatot igényel. Éppen ezért még a filmiparban is csak indokolt esetben használják. Ha élünk azonban a környezeti képek azon egyszerűsítésével, miszerint a környezet végtelenül nagynak tekinthető a később belé helyezett virtuális objektumhoz képest, akkor egyszerűbb megoldás is létezik. Az inverz feladat megoldásához a fényterjedés szimulációjában rögzíthetünk pár paramétert, név szerint egy objektum geometriáját, és az objektum felületi jellemzőit. Ezért a valós jelenetről úgy készítünk képet, hogy egy ismert egyszerű geometriájú, egyszínű, ismert BRDF-el rendelkező tárgyat helyezünk. Az ezen a tárgyon látható fényhatásokból már következtethetünk a környezeti megvilágításra.

A gyakorlatban ez az egyszerű tárgy egy gömb szokott lenni, és két BRDF-et szoktunk használni: egy diffúz semleges színű (fehér, vagy szürke) gömböt, és egy tökéletesen tükröződő (króm bevonatú) gömböt. A króm gömb

segítségével a részletes környezeti képet, a diffúz gömb segítségével gyakorlatilag a diffúz BRDF-el előkonvolválta képet fogjuk megkapni. A 3.1. ábra a környezetből érkező megvilágítás számítását mutatja. A bal oldalon a diffúz megvilágítást láthatjuk. A valós jelenetbe oda helyezük a diffúz gömböt, ahova a virtuális objektumainkat is helyezni fogjuk, és így készítünk képet a gömbről. A gömb egyes felületi pontjai más-más irányúak. A gömb éppen azért tökéletes számunkra, mert minden háromdimenziós irányhoz pontosan egy felületi pont tartozik, és ezek helyét pontosan ismerjük. A gömbről készített fotó minden olyan pixeléhez, amiben a gömb látható pontosan hozzárendelhetünk egy háromdimenziós irányt, ami a felület normálvektora. Az ábrán egy ilyen felületi pontot jelöltünk piros ponttal és a hozzá tartozó normálvektort  $N$ -nel. A gömb egy pixele azt az információt hordozza, hogy a pixelhez tartozó normálvektornak megfelelő irányultságú felületek milyen diffúz megvilágítást kapnak. Tehát a minden irányból érkező fénysugarak összesített hatását látjuk. Éppen azért fontos a semleges fehér vagy szürke szín, hogy ne módosítsa a környezeti megvilágítás színét. Egy gömbről egy nézőpontból készült képen az összes lehetséges irány fele reprezentálva van (melyek a kamera felé mutatnak), ezek értékeit be is írhatjuk a környezeti képbe ( $p_N$ ). A teljes környezeti kép elkészítéséhez az ellenkező irányból is le kell fotóznunk a gömböt. Az így kapott környezeti képet pedig már felhasználhatjuk egy diffúz virtuális objektum kép alapú megvilágításához, ahogyan azt az előző fejezetben is láttuk.



3.1. ábra Környezeti kép visszaszámítása diffúz és tükörgömből készített fotó segítségével.

A tükör gömb felhasználása hasonló, csupán az a különbség, hogy itt a gömb egy pixelében nem a minden irányból beérkező fénysugarak összességét, hanem csak a tükörirányból ( $R$ ) érkező fénymennyiséget láthatjuk. Itt is minden pixelhez egyértelműen egy irány tartozik, az irányok azonban nagyobb tartományt lefednek, szinte az egész teret, de azért érdemes itt is képet készíteni az ellenkező kamera irányból is. Az így kapott környezeti kép magát a környezet tiszta képét fogja tartalmazni, melyet felhasználhatunk tükröződések, törések megjelenítésére, vagy előkonvolúcióval matt fényű anyagokhoz is. Persze a diffúz környezeti képet is számíthatjuk ebből a képből, de ha diffúz virtuális objektumokkal fogunk dolgozni, akkor a diffúz gömb használata előnyösebb.

### 3.2 HDR KÉPALKOTÁS

A kép alapú megvilágítás bemutatásánál átsiklottunk egy igencsak fontos dolgon, a képek dinamika tartományán. Egy kép dinamika tartománya a legsötétebb és a legvilágosabb pixeléhez tartozó fényerősségek különbsége és megmutatja, hogy mekkora fényerősség tartományt rögzít a kép. A számítástechnikában mindent számokkal adunk meg és ezeknek a számoknak az ábrázolási pontossága véges, tehát rögzített számú számjegyük, tizedes jegyük van. Így az is fontos, hogy ebben a tartományban hány különböző fényesség értéket tudunk tárolni. Az ábrázolás pontossága az egyes számok tárolásához használt bitek száma, ezért ezt bitmélységnek is szoktuk hívni.

A mindennapjainkban használt képeken 8 bitet használunk egy szám rögzítésére. Ennek az a következménye, hogy egy szürkeskálás képen a legsötétebb fekete és a legvilágosabb fehér között 256 (2 a 8. hatványon) különböző világosságérték tárolható, ahol 0 jelöli a feketét, 255 pedig a fehéret. A legfehérebb fehér így 255-ször világosabb, mint a feketénél eggyel világosabb eltárolható szín. Ha a bitmélységet például 8-ról 9-re emelnénk, már 512 (2 a 9. hatványon) különböző értéket tudnánk tárolni, tehát a fehér 511-szer világosabb, mint a feketénél eggyel világosabb eltárolható szín.

Ez azért fontos, mert a valós világban ennél jóval nagyobb dinamikai tartományok vannak. Akár több tízezeres különbség is lehet egy árnyékban lévő pont, és egy fényforrás egy pontja között. Mivel a legtöbb kijelzőnk színcsatornánkként 8 bites színmélységgel dolgozik, ezért a digitális képeink esetében is elegendő 8 bit használata. A kép alapú árnyalás során azonban az árnyaláshoz használt kép pixeli a BRDF függvény bemenetei, és ez a függvény bizony jócskán súlyozhatja ezeket a pixelértékeket. Ez azért baj, mert a környezeti kép elkészítésekor döntéskényszerben vagyunk: el kell döntenünk, hogy a fix 256 fényességértéket a sötét területekhez vagy a világos területekhez használjuk fel. Ha a sötét területekhez használjuk fel, akkor a világos részek túlfutnak az ábrázolható számtartományon, úgy is mondjuk, hogy beégnek. A fotográfiában ezt túlexponálásnak hívjuk. Ilyenkor a sötét területeken kivehetők a részletek, de a világos területek egyszínű fehér területekké válnak. A BRDF-ben történő súlyozás (pixelértékek csökkentése) hatására minden világos terület egyszerre kezd sötétedni és ugyanazt a sötétebb árnyalatot veszi fel, míg az lenne a helyes, ha az igazán világos részek még mindig beégettek lennének. Ha úgy döntenénk, hogy a világos területekre helyezzük a dinamika tartományt, akkor pedig alulexponált képhez jutunk: bár a világos területeken kivehetők a részletek, a sötét területek egyszínű fekete foltokká válnak. Dönthetnénk még úgy is, hogy a rendelkezésünkre álló 256 értéket elosztjuk az egész tartományon, de ezzel csak azt érnének el, hogy sem a sötét, sem a világos területeken nem látnánk részleteket, mert a finom színekülönbségek nem, csak nagyobb ugrások tárolhatók. Ilyenkor a képen feltűnő sávok hibák látszanak. A legjobb megoldás az lenne, ha lefedjük az egész tartományt, de jóval több különböző színárnyalatot tudunk megkülönböztetni, tehát növeljük a bitmélységet.

A környezeti képek tárolására tehát nagyobb bitmélységet kell használnunk. Ilyen képek kezelése azonban nem megszokott, speciális szoftvereket igényel. Az is fontos, hogy a fényképezőgépeink is 8 bites képeket tárolnak. Itt megjegyezzük, hogy a professzionális fényképezőgépek képesek nagyobb bitmélységgel tárolni, de ekkor sem nő meg az ábrázolt dinamika tartomány, tehát továbbra is a fényképezőgép beállításával választunk a fényes vagy a sötét területek között, de ezeket finomabb lépésközzel tárolja. Nagy dinamikájú (High Dynamic Range, HDR) kép tárolásához több felvételt kell készítenünk, különböző fényesség tartományokra koncentráva és ezekből összerakunk egy végső képet.

Szerencsére léteznek szoftverek, melyek a különböző expozícióval készült képekből egy nagy dinamikájú képet tudnak előállítani. Ezeknek a képeknek speciális formátuma van, és a méretük is nagyobb az általánosan használt képekénél.

A környezeti kép elkészítéséhez tehát több felvételt is kell készítenünk a diffúz, vagy tükör gömbünkről minden kameraállásban, és ezeket HDR képpé kell alakítanunk. Így természetesen csak mozdulatlan környezetet tudunk rögzíteni, különben az egyes képek kicsit más környezetről készülnek, ami az összefésülésnél szellemképeket okoz. Ez azonban nem nagy megkötés számunkra, hiszen amúgy is legalább két irányból készítünk a gömből felvételt, így a mozdulatlan környezet a preferált.

A 3.2. ábra jól mutatja a különbséget az alacsony és a magas dinamikájú megvilágítás között. A bal oldali, alacsony dinamikájú megvilágítás esetén a sötét tüköröződő gömbön a felhők teljesen egyszínűek. Ez azért van, mert a környezet képeinek készítésekor túlexponáltuk a képet, hogy az árnyékos részek részletei jól látszódnak, viszont a felhők beégtek. A jobb oldali gömböt HDR képpel világítottuk meg, melyben a világos területeken is vannak részletek. Ezek a részletek szépen kivehetők a felhők tükröződésénél.



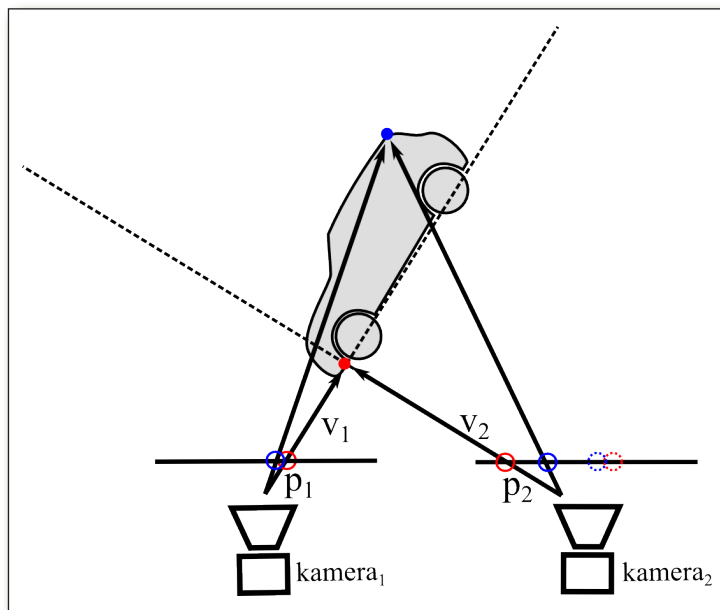
**3.2. ábra Alacsony (bal) és magas (jobb) dinamikájú környezeti megvilágítás hatása a tükröződésben.**

### 3.3 GEOMETRIA VISSZAÁLLÍTÁSA

Egy másik fontos inverz feladat a valós világ egyes objektumai geometriájának a visszaállítása. Ez egy igen nehéz feladat, egyetlen képből nem is lehet a geometriát pontosan visszafejteni, csak közelíteni. A geometria durva közelítésének is vannak hasznos alkalmazásai. Például a képen található főbb vonalak segítségével azonosíthatjuk a horizontot, esetleg a függőleges falakat, majd a környezetet egy nagy sík talajjal, egy félgömb égbolttal, és pár függőleges síklappal helyettesíthetjük. Erre a geometriára visszavetíthetjük a kameraképet, így egy textúrázott közelítő geometriát kapunk. Ezt a geometriát már felhasználhatjuk környezeti kép előállítására, ami akkor lehet hasznos, ha nem készült diffúz és tükörgömbös felvétel a környezetről.

Ha nem a tágabb környezetet szeretnénk rekonstruálni, hanem egy közelebbi, a képen központi szerepet játszó valós objektum geometriáját, akkor már valamilyen gépi látás algoritmust kell segítségül hívnunk.

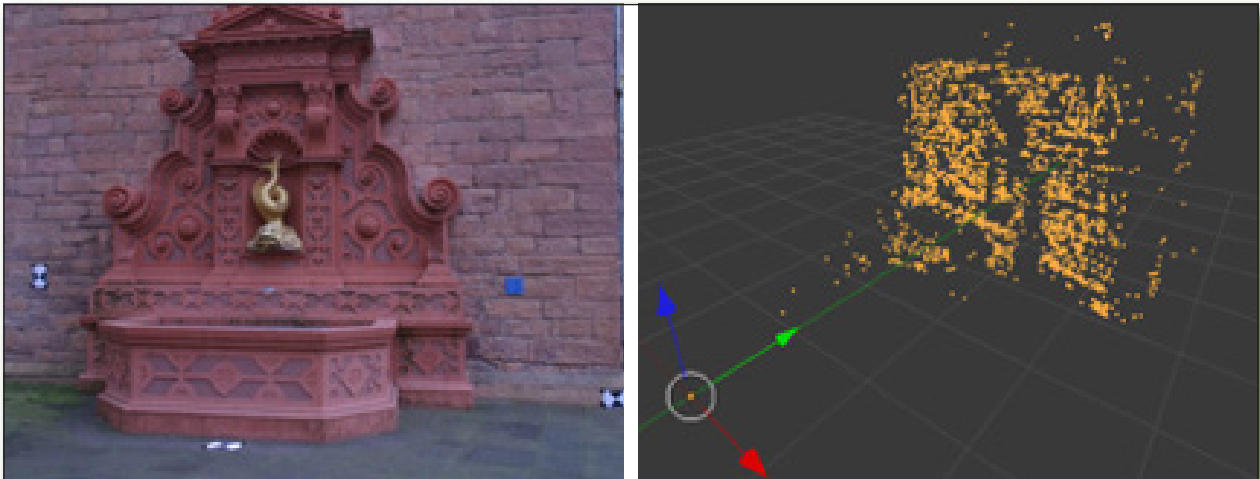
A legkézenfekvőbb megoldás ilyenkor a sztereó rekonstrukció lehet, amikor a jelentről két kép is készül egymáshoz képest kicsit más kamera szemszögből. Használhatunk két kamerát, és egyszerre készíthetünk képet, de felhasználhatunk két egymás utáni felvételt is, amik között a kamera pozíciót változtatott. Ez a két kép lehet egy videó két megfelelő képkockája is.



**3.3. ábra Sztereó rekonstrukció két kamerával.**

A rekonstrukció menete röviden a következő. Mindkét képen jól felismerhető, jellemző képrészleteket keresünk. Ezután a két képen megtalált jellemző pontokat megpróbáljuk megfeleltetni egymással, tehát olyan pontokat keresünk a felületen, melyek mindkét kamerából láthatók és jól azonosíthatók. A 3.3. ábra piros és kék pontokkal jelöl két ilyen felületi pontot. Ha ismerjük a két kamera egymáshoz képesti helyzetét és a kamerák látószögét, akkor egy egyenest indítva mindkét kamerából a jellemző pont pixelé felé a két egyenes (szaggatott vonal) egy pontban metszi egymást, ez lesz a felületi pont háromdimenziós koordinátája. Az ábrán a piros felületi ponthoz tartozó pixelt az első kamera képén  $p_1$ -gyel jelöltük, míg a második kamera képén  $p_2$ -vel. Ezekből a pixelekből indított sugarakat  $v_1$ -gyel és  $v_2$ -vel jelöltük. A kettes kamera képére pontozott körökkel bejelöltük a két felületi pont másik kamerához tartozó pixel koordinátákat is. Ezen jól látszik, hogy a pontok egészen más pixelekből látszódnak az egyik, mint a másik kamera képén. Azt is érdemes megfigyelni, hogy a kamerától távolabb eső kék pont két pixel koordinátája között már nincs akkora különbség, mint a közelebbi, piros pont koordinátái között. Ez a perspektív torzítás miatt van. Tehát, ha a két kamera csak egymáshoz képest vízszintesen van elcsúsztatva, akkor a felületi pontok kameráktól mért távolságát a hozzá tartozó két pixelkoordináta különbségéből (diszparitás) is kikövetkeztethetjük, anélkül, hogy a vonalmetszést elvégeznénk.

A jól azonosítható pontok megtalálása, összepárosítása és háromdimenziós rekonstrukciója után a köztes pontok koordinátái interpolációs technikákkal számíthatók. A végeredmény egy sűrű háromdimenziós ponthalmaz, melyből háromszögháló is előállítható. Ha sikerült a valós világ egy objektumát rekonstruálnunk, akkor azt felhasználhatjuk a virtuális világ objektumainak megjelenítésekor, például helyes takarás megvalósítására vagy fizikai interakció számítására.



**3.4. ábra Sztereó rekonstrukció eredménye.**

A geometria ismeretében megkísérelhetjük a geometria saját színének visszaállítását is. A probléma azért is bonyolult, mert a látott szín függ a geometriától, a felület saját színétől, az anyagmodelltől és a megvilágítástól is. A saját szín kinyerését általában a megvilágítás visszszámítása után tudjuk megtenni, és ekkor is rögzítenünk kell az anyagmodellt, mely a legtöbb módszernél csak diffúz BRDF lehet. A feladat gyakorlatilag a fényforrások hatásainak eltüntetése, így csak a felület saját színe marad.

### 3.4 ÚJRAFÉNYELÉS

Újrafényelés során a valós világ objektumait igazítjuk a virtuális világhoz. Erre szükség lehet, ha a virtuális világ fényviszonyai jelentősen eltérnek a rögzített valós környezet fényviszonyaitól, vagy esetleg a virtuális világ plusz fényforrásokat definiált. Mindkét esetben szükség van a felület saját színére. Az első esetben a rögzített kamera képen a rekonstruált objektum pixeleit teljesen újraszámoljuk vagy lokális illuminációval vagy környezeti megvilágítással, ugyanúgy, mint a virtuális környezet bármely más objektumát. Az utóbbi esetben az eredeti pixeleket megtartjuk, csupán az új, virtuális fényforrások hatásait hozzáadjuk a kameraképhez. Általában elmondható, hogy ha tudjuk, hogy a valós objektumok újra lesznek fényelve, akkor, amennyiben van rá módunk, a felvételkor érdemes őket a lehető legegyszerűsebben bevilágítani. Ez a saját szín számítását is megkönnyíti, sőt ez a lépés akár ki is hagyható, mert az egyenletesen bevilágított felület színei használhatók saját színeként.

Egy speciális esete a valós objektumok újrafényelésének a virtuális objektumok által keltett árnyékok megjelenítése. Mivel a praktikus alkalmazások során nagyon sokszor egy síklapra helyezünk el a virtuális objektumokat, így a valós objektum, amire az árnyékokat utólag vetni szeretnénk is közelíthető síkként. Ha sikerül meghatározunk egy fő virtuális fényforrást, akkor ennek az árnyékával csökkenthetjük az eredeti objektum pixeleinek fényességét. Ha a megvilágítás nem közelíthető egy-két virtuális fényforrással, akkor bonyolultabb árnyékolások is megjeleníthetők a differenciális képszintézis módszerének alkalmazásával.<sup>5</sup>

<sup>5</sup> Debevec 1998.

## 4. KÉPTÉRBELI HATÁSOK

A valódi kamerákkal készített felvételeken egy sor olyan jelenség megfigyelhető, amelyek a kamerák fizikai felépítéséből következnek. Ezen effektusok az inkrementális képszintézissel készült képeken hiányoznak, hiszen a nagyon egyszerű lyukkamera modell ezek megjelenését nem teszi lehetővé. A számítógéppel készített képek valószínűségének fokozása érdekében érdemes ezen jelenségek valamilyen közelítését alkalmazni. A kamera modellből fakadó hatások mellett vannak olyan további jelenségek is, amelyek a virtuális világ felépítésétől függetleníthetők és képtérben is hatékonyan közelíthetők.<sup>6</sup> Ennek a megközelítésnek az előnye abban rejlik, hogy a képtérben számítható hatások esetén a számítások bonyolultságát a kép felbontása határozza meg, amely lényegesen kisebb lehet a színtérbeli geometria bonyolultságánál. A képtérbeli módszerek hátránya viszont, hogy csak a geometria azon részéről rendelkezünk bármilyen információval, amelyek közvetlenül láthatóak a kamerából. Ebben a fejezetben áttekintjük a képtérben létrehozható jelentősebb hatásokat és bemutatjuk azok hatékony megvalósításait a valósidejűség jelentette korlátok figyelembevételével.

### 4.1 TONE MAPPING

Mint azt a HDR képalkotásról szóló 3.2 szakaszban láthattuk, a valósághű képek előállítására érdekében érdemes a felületi pontok árnyalását szélesebb dinamika tartományban végezni, mint amelyet a konvencionális kép megjelenítő képes kezelni. A széles dinamika tartományú képeket azonban a megjelenítő képességeihez kell alakítanunk. Ezt az eljárást nevezzük tone mappingnek avagy tónus leképezésnek. Az eljárás alkalmazása során a megjelenítendő kép értéktartományát valamilyen átviteli függvény segítségével leképezzük arra a véges diszkrét tartományra, amelyet végül meg tud jeleníteni a képernyő.

A dinamika tartomány leképezésére alkalmazható módszereket alapvetően két nagy csoportba sorolhatjuk. Az úgynevezett globális módszerek alkotják az egyik csoportot, amelyek alapvetően a teljes képen értelmezett dinamika tartományból indulnak ki. Ezzel szemben a másik csoportot alkotó, úgynevezett lokális módszerek a kép egyes régióit különválasztják és a leképezést az egyes régiókon egymástól függetlenül végzik el.

A globális leképezési eljárások esetén figyelembe tudjuk venni az emberi szem adaptációs képességét, azaz utánozni tudjuk azt a folyamatot, amely során az emberi látás igazodik az aktuális látvány fényerejéhez. A leképezés során először meg kell határoznunk a látott kép átlagos fényerejét, amely megadja azt a fényesség értéket, amelyhez a szem igazodni fog. A fényerő számítására az RGB komponensekkel adott színek nem alkalmasak közvetlenül, mivel az emberi szem az egyes hullámhosszokra eltérő mértékben érzékeny.

Ennek megfelelően első lépésként az RGB színtérben adott képet át kell számítani egy olyan színtérbe, amelyben a fényesség értelmezhetővé válik. Egy ilyen színtér a szabványos XYZ színtér, amely egy eszközfüggetlen rep-

<sup>6</sup> Szirmay-Kalos 2004.

rezentációja az emberi szem által érzékelhető színeknek. Az RGB színtér egy egyszerű lineáris leképzéssel átvihető az XYZ színtérbe az alábbi függvény segítségével:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0,176} \begin{bmatrix} 0,490 & 0,310 & 0,200 \\ 0,176 & 0,812 & 0,010 \\ 0,000 & 0,010 & 0,990 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}.$$

A XYZ színtérben ábrázolt színek Y komponense a fényesség érték, amelyeket a teljes képre kell átlagolnunk. Figyelembe véve, hogy az emberi szem sokkal érzékenyebb a fényerő értékek közti különbségre, mint a fényerő abszolút mértékére, az átlagos fényerőt geometriai átlagként számítjuk aritmetikai átlag helyett. A geometriai átlagot úgy számíthatjuk, hogy először vesszük a fényesség értékek (például 10-es alapú) logaritmusát, majd ezen logaritmusoknak számítjuk az aritmetikai átlagát, végül az aritmetikai átlagot hatványozással visszaemeljük az eredeti tartományba. Az így kapott átlagos fényesség érték ( $Y'$ ) lesz az, amelyhez a tone mapping eljárás során az egyes fényesség tartományokat viszonyítjuk.

Következő lépésként a pixelenként adott fényesség értékeket leképezzük az átlaghoz képesti viszonylagos fényesség ( $Y_r$ ) értékekre az  $Y_r = \frac{\alpha Y}{Y'}$  függvény segítségével, amelyben az  $\alpha$  paraméter egy szabadon választható konstans kulcs érték. A kulcs segítségével lehetőség nyílik annak a meghatározására, hogy az átlagos fényesség a megjeleníthető tartomány mely pontjára képződjön le. A viszonylagos fényesség értékeket ezek után leképezzük a megjeleníthető fényesség értékek tartományára a következő függvénnyel:

$$D = \frac{Y_r(1 + Y_r/Y_w^2)}{1 + Y_r},$$

Az  $Y_w$  egy szabadon választható konstans érték, amely azt viszonylagos fényesség értéket adja meg, amelyet a megjelenítő eszköz által megjeleníthető legnagyobb fényesség értékre szeretnénk leképezni. Az ennél az értéknél nagyobb viszonylagos fényesség értékek be fognak égni és nem lehet őket megkülönböztetni egymástól. Az utolsó lépés az eredeti RGB színtérbeli, nagy dinamika tartományú értékek leképzése a megjeleníthető tartományra, amelyet a következő függvény segítségével tehetünk meg:

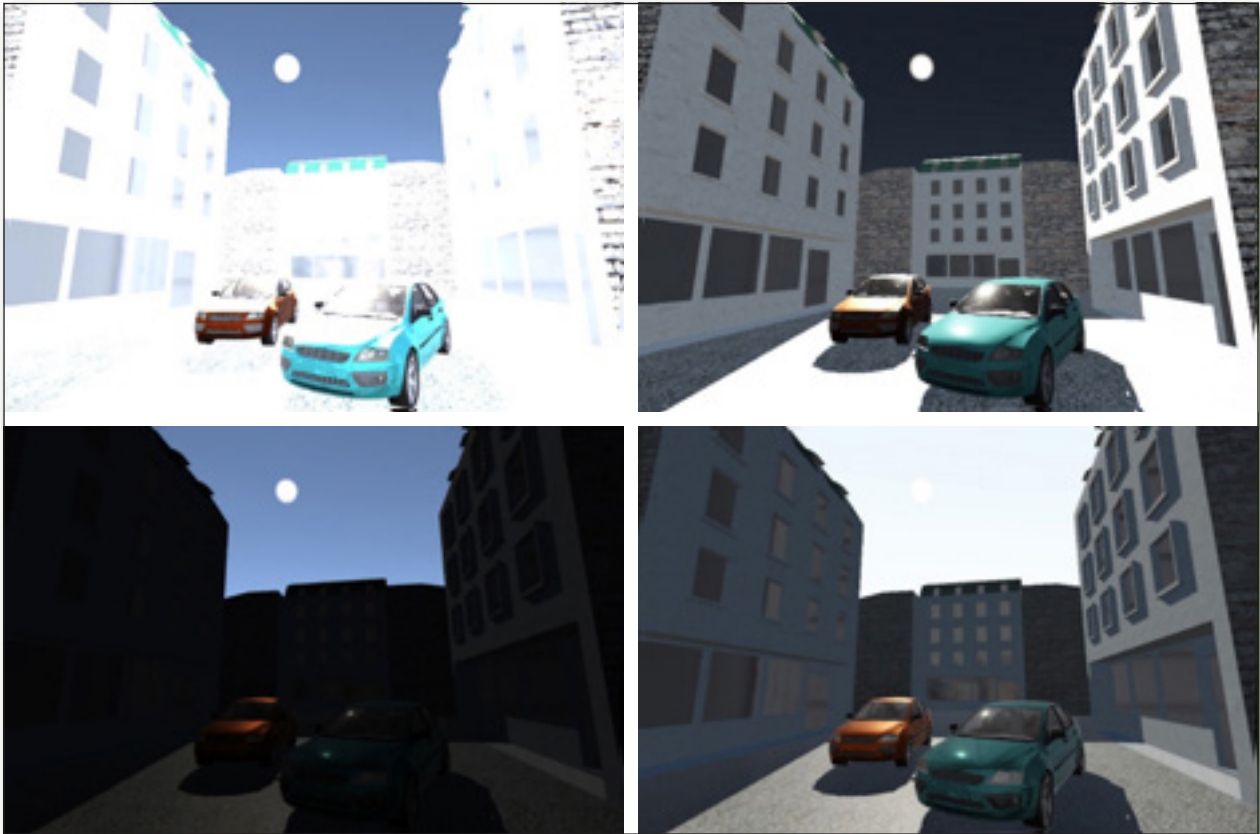
$$[r \quad g \quad b] = [R \quad G \quad B] \frac{D}{Y'}$$

A leképzés eredményeképpen előálló kép már csak olyan RGB értékeket tartalmaz, amelyek a igazodnak a képmegjelenítő eszköz képességeihez. Amennyiben az emberi szem időbeli adaptációját is közelíteni szeretnénk, akkor ezt az átlagos fényesség időbeli átlagolásával tehetjük meg a legegyszerűbben. Érdekes azonban figyelembe venni azt a tényt, hogy az emberi szem a kis fényerőtől a nagy irányába gyorsabban képes alkalmazkodni a változásokhoz, mint ellenkező irányban.



A fentebb vázolt globális módszerek gyakran alkalmazott kiegészítése, hogy az átlagos fényesség értéket nem a teljes képből számítjuk, hanem csak egy kisebb tartományból, például a kép közepéről. Ennek hatására a leképzés során azt a tartományt vesszük csak figyelembe, amely a kép fókuszában van, így a kép kevésbé érdekes régióiban esetlegesen megjelenő nagyon eltérő fényesség értékek nem fogják jelentősen rontani a kép közepének kontrasztját.

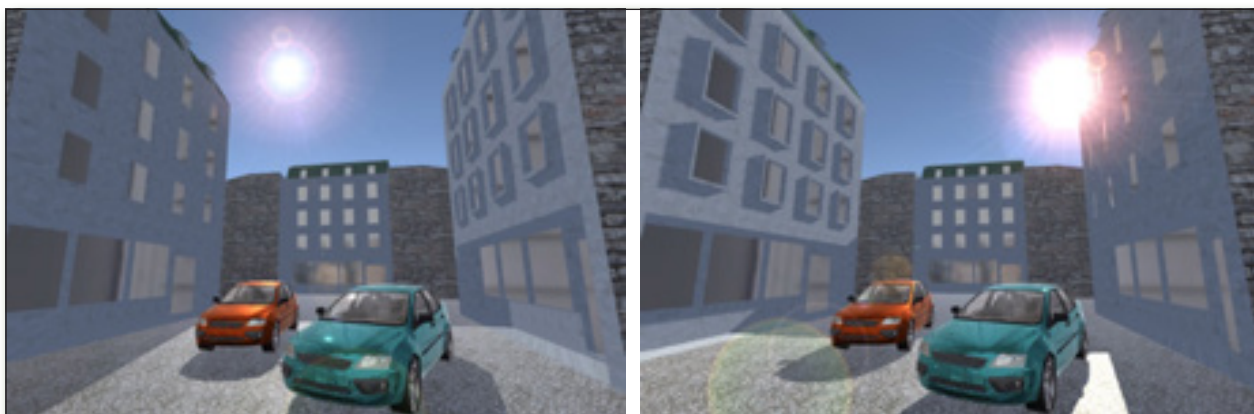
A lokális leképző módszerek ezt a gondolatot viszik tovább úgy, hogy a teljes dinamika tartományt több részre osztják és ott külön végzik el a leképzés folyamatát, így megőrizve a kontrasztot minden fényesség tartományban. Ennek a módszernek azonban hátránya, hogy a teljes képre nézve a kontraszt elveszik és a kép fakónak fog tűnni annak ellenére, hogy a részletek megvannak a képen. A 4.1. ábra a tone mapping művelet hatását mutatja egy nagy fényerejű és egy nagyon sötét jelenet esetén. Az ábra bal oldalán a nagy dinamika tartományú képeket egyszerű tartomány leképzéssel jelenítettük meg, míg a jobb oldalon az előbbieken bemutatott módszer segítségével.



4.1. ábra A tone mapping művelet hatása nagyon nagy fényerejű jelenet esetén (felső sor), illetve nagyon sötét jelenet esetén (alsó sor).

## 4.2 CSILLANÁSOK, RAGYOGÁS

A fizikai kamerákban található lencserendszer és a rekesz, amely a fényt beengedi az érzékelőre egy sor olyan másodlagos hatást képes létrehozni a képeken, amelyeket művészi eszközként is felhasználhatunk a kép egyes részeinek hangsúlyozására. Az egyik ilyen hatás a lencsecsillanás (lens flare), amelyet az optikai lencsék apróbb hibái okoznak. Ezek hatására a fény nem csak a kívánt irányba fókuszálódik, hanem a lencsén belül is szóródik. Ez a jelenség általában nagyon erős fényforrások környezetében jelenik meg, akár olyan esetben is, ha maga a fényforrás nem is a kamera látóterében található. Az elkészült képen kétféle módon jelenhet meg általában ez a hatás. Az egyik esetben csökkenti a kép kontrasztját és a színek telítettségét azáltal, hogy a sötét régiókba is juttat nem kívánt fényt, illetve a fényes régiókba is további fehér fényt. Ez a hatás általában rontja a kép minőségét, ezért ennek megvalósításával nem foglalkozunk részletesen. A képen megjelenő másik hatás a lencsék alakjának megjelenése, amely már sokkal érdekesebb művészeti szempontból is. Ezt a hatást a lencserendszer optikai szimulációjával igen pontosan lehet közelíteni, azonban ez a módszer valósidejű felhasználásra alkalmatlan. Amennyiben nem célunk valós kamerával készített felvételekhez illeszteni a számítógéppel alkotott képeinket, akkor a legegyszerűbb megoldás az egyes lencsékhez kapcsolódó alakzatok előre elkészítése és textúrában tárolása. A megjelenítés során ezen képeket utófeldolgozási lépésként tudjuk hozzá adni az elkészült képhez úgy, hogy a színtérben található fényforrást és a képsík közepét összekötő egyenes mentén plakátként (billboard) rávetítjük a képre, ahogy azt a 4.2. ábrán láthatjuk. Amennyiben valós felvételeket kell kiegészítenünk, akkor választhatunk valamilyen Monte-Carlo szimuláció alapú módszert, illetve léteznek modell alapú eljárások is, amelyekkel jól közelíthető meg ez a jelenség, azonban valós idejű felhasználásra csak korlátozottan alkalmazhatóak.



4.2. ábra A lencsecsillanás hatása eltérő fényforrás pozíciók esetén.

Ide kapcsolódó jelenség a képen látható fényforrások körül megjelenő ragyogás jelensége is, amelyet részben a fényforrás és a kamera közötti térrészben, részben a lencserendszerben és a kamera fényérzékelőjében lejátszódó szóródási jelenség okoz. Ennek hatására a fényforrás környezete kivilágosodik, a távolabbi részek kontrasztja pedig csökken. Ahhoz, hogy ezt a jelenséget modellezzük, első lépésként meg kell határoznunk azokat a képpontokat, amelyben az fényesség értékek lényegesen magasabbak a kép átlagos fényességénél. Ehhez alkalmazhatjuk az előző szakaszban ismertetett fényesség transzformáció, illetve átlagos és viszonylagos fényesség számítási lépéseket.

Ezek után egy küszöbözési lépés segítségével kiválasztható azon a képpontok, amelyek lényegesen fényesebbek környezetüknél. A kiválasztott képpontok színét ezek után szétosztjuk a környező képpontok között, amelyhez például alkalmazhatjuk a Gauss függvénnyel való szűrést. Az így előállított ragyogás térképet ezek után egyszerűen hozzáadjuk az eredeti képhez a tone mapping lépés előtt. A 4.3. ábra a ragyogás hatását mutatja eltérő fényforrás pozíciók esetén.



**4.3. ábra Ragyogás effektus hatása különböző fényforrás helyzetek esetén.**

Az elmosáshoz alkalmazott Gauss-függvénnyel szűrés művelete gyakran előfordul a képtérbeli algoritmusok esetén, így érdemes megfontolni, hogy miként valósítható meg hatékonyan.<sup>7</sup> A képtérbeli szűrések egy kétdimenziós térbeli integrál kiértékelésének felelnek meg, amely a következő alakban írható le:

$$L'(X, Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L(X - x, Y - y) w(X, Y, x, y) dx dy,$$

ahol  $L'(X, Y)$  a szűrés eredményeként kapott érték az  $X, Y$  koordinátájú pixelben,  $L(X, Y)$  az eredeti kép, és  $w(X, Y, x, y)$  az adott pixelhez tartozó szűrő függvény. Abban az esetben, ha a szűrő függvény minden képpontban azonos, akkor a szűrő függvényt tér-invariánsnak nevezzük. Az elmosáshoz választott kétdimenziós  $\sigma^2$  szórású Gauss-függvény tér-invariáns alakja

$$w(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

<sup>7</sup> Tóth 2013.

A szűrő integrálokat közelíthetjük véges összegként, amely illeszkedik a szűrendő kép diszkrét természetéhez. Az  $N$  tagból álló véges összegként közelítés általános alakja

$$L'(X, Y) \approx \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}} L(X-i, Y-j)w(i, j).$$

A kifejezés kiértékeléséhez  $N^2$  műveletet kell végeznünk minden egyes képpontban. A szükséges számítások mennyisége csökkenthető abban az esetben, ha a szűrő függvény a dimenziók mentén szétválasztható. Ebben az esetben a szűrés a dimenziók mentén végzett szűrések szorzataként is felírható:  $w(x, y) = w_x(x) \cdot w_y(y)$ . Tér-invariáns szűrők esetén, a kétdimenziós integrált két lépésben, két egydimenziós integrál kiértékeléseként is számíthatjuk, amely a szükséges számítások mennyiségét az eredeti  $N^2$ -ről  $2N$ -re csökkenti. Az első lépésben a kiszámítandó kétdimenziós függvény alakja

$$L'(X, Y) \approx \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}} L(X-i, Y-j)w(i, j).$$

amely szűrés eredményeképpen előállt átmeneti képet újra megszűrjük a másik dimenzió mentén, amely általános alakja

$$L(X, Y) \approx \sum_{i=-\frac{N}{2}}^{\frac{N}{2}} L_t(X, Y-i)w_y(i).$$

További lehetőség a szűrés hatékonyságának növelésére, ha nem csak a szűrés eredményének számításához felhasznált minták számát csökkentjük, hanem a mintapontok elhelyezését is körültekintően választjuk meg. Eddig a szűrés során feltételeztük, hogy a mintapontok egyenletesen helyezkednek el a szűrendő pont körül, ha azonban ismerjük a szűrő függvény alakját, akkor oda tudjuk összpontosítani mintáinkat, ahol azoknak a legnagyobb hatása lesz a végeredményre. Ezt a fontosság szerinti mintavétel elvének nevezzük, és a következők szerint alkalmazhatjuk a konvolúciós szűrések esetében. Tekintsük a

$$L'(X) = \int_{-\infty}^{\infty} L(X-x)w(x)dx$$

egydimenziós konvolúciót! Határozzuk meg a szűrő függvény  $\tau(x)$  integrálját és  $x(\tau)$  inverzét úgy, hogy igaz legyen

$$\frac{d\tau}{dx} = w(x), \text{ azaz } \tau(x) = \int_{-\infty}^x w(t) dt.$$

Amennyiben a  $w(x)$  szűrő függvény egyben valószínűségi sűrűségfüggvény, azaz integrálja 1 és sehol sem negatív, akkor  $\tau(x)$  annak kumulatív sűrűségfüggvénye. Amennyiben ismerjük a szűrő függvényt, akkor  $x(\tau)$  kiszámolható és invertálható véges sok egyenletesen elosztott mintapontban. Helyettesítsük be  $x(\tau)$ -t a szűrő integrálba, amely eredményeképp a következő alakot kapjuk:

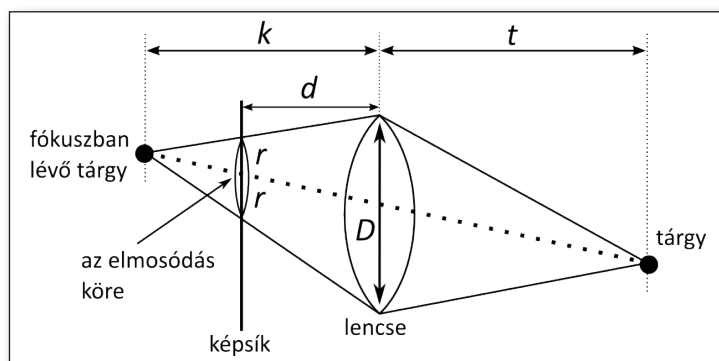
$$D = \frac{Y_r(1 + Y_r/Y_w^2)}{1 + Y_r},$$

Az így kapott integrál véges összegű közelítése egyenletesen választott mintapontokkal megegyezik az eredeti integrál véges összegű közelítésével, azonban a mintapontok elhelyezkedése már nem lesz egyenletes. A mintapontokhoz tartozó súlyok a transzformáció után egyenletesek lesznek. Az eredeti integrálhoz tartozó mintapontokat előfeldolgozási lépésként számíthatjuk ki, így a szűrés során nem jelent további költséget. A kapott mintapontok nem feltétlenül lesznek egész koordinátákon, így a hozzájuk tartozó színértéket interpolálva kell kiolvasni, amelyet a grafikus hardver lényeges költség nélkül támogat.

A fontosság szerinti mintavétel alkalmazásával azonos pontosság mellett csökkenteni tudjuk a szükséges minták számát, vagy azonos mintaszám esetén növelni tudjuk a közelítés pontosságát.

### 4.3 MÉLYSÉGÉLESSÉG

A valós idejű számítógépes grafikában alkalmazott lyukkamera modell csak egy végtelenül kicsiny lyukon keresztül engedti a virtuális világból érkező fénysugarakat a képsíkra. Ennek következményeként a tér minden pontjában éles képet kapunk. A fizikai kamerák lencséi véges kiterjedésűek, azaz lényegesen szélesebb iránytartományból engedik be a fényt az érzékelőre, így csak azok a régiók lesznek élesek a képen, amelyek a kamera aktuális fókusztávolsága körül találhatóak, a kép többi része elmosott lesz. Ezt a jelenséget mélységélességi hatásnak (depth of field) nevezzük. A valódi világról készült felvételeken ezt a hatást a téma hangsúlyozására használhatjuk általában, emiatt érdemes a számítógéppel készített képeken is valamilyen formában megvalósítani.



4.4. ábra A mélységélesség optikai háttere.

Annak érdekében, hogy a szemlélő számára hihető eredményt érjünk el, érdemes áttekinteni a jelenség optikai hátterét (4.4. ábra). Az általánosan alkalmazott geometria optika modellje<sup>8</sup> szerint egy  $f$  fókusz távolságú lencse esetén a kamerától  $t$  távolságra található pont képe akkor lesz éles a lencse mögött  $k$  távolságra található képsíkon, ha az  $f, t, k$  távolságokra igaz, hogy

$$\frac{1}{f} = \frac{1}{k} + \frac{1}{t},$$

azaz a lencse gyújtópont távolságának reciproka megegyezik a tárgytávolság és az érzékelő távolságának reciprokának összegével. Abban az esetben, ha a képsík nem  $k$  távolságban van, akkor a lencse által a képsíkra vetített pont képe elmosódik és egy sugarú körré válik:

$$r = \frac{|k - d| D}{k \cdot 2},$$

ahol  $D$  a lencse átmérője,  $d$  pedig a lencse és képsík közötti aktuális távolság. Az elmosódás hatására a képsíkra vetülő pont színe az  $r$  sugarú környezetben is hozzájárul a végső színérzethez. Tekintve, hogy a modern kamera rendszerek jellemzően összetett, változtatható elrendezésű lencserendszerekkel vannak felszerelve, ezért az egyes lencsék fókusz távolságára épülő modell használata nehézkes. A modellünk szempontjából érdemes inkább a teljes optikai elrendezést vizsgálni, és meghatározni azt a lencsétől vett  $P$  gyújtótávolságot, amely az élesen a képsíkra vetülő régiót jellemezi. A gyújtótávolságra, illetve a képsík lencsétől vett távolságára is igaz az

$$\frac{1}{f} = \frac{1}{d} + \frac{1}{P}$$

összefüggés. A fenti összefüggésekből már levezethető az elmosódás mértékét különböző tárgytávolságokra praktikus módon megadó képlet:

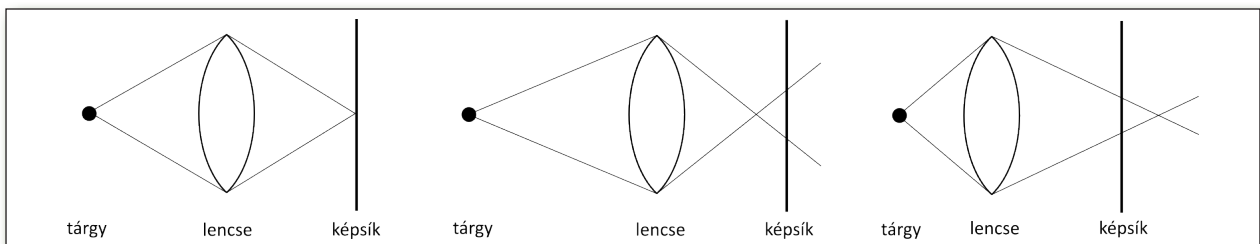
<sup>8</sup> Ábrahám 1997.

$$r = \left| \frac{1}{t} - \frac{1}{P} \right| \frac{Dd}{2}$$

Tehát az elmosódás mértéke arányos az objektumtávolság és a gyújtótávolság reciprokeinak különbségével. Ezt az összefüggést tovább egyszerűsíthetjük, ha felhasználjuk az inkrementális képalkotás során alkalmazott egyszerűsített kamera modellt, amely a kamera terében értelmezett  $z$  távolságokat a virtuális kamera első és hátsó vágósíkainak figyelembe vételével képezi le a képtérbeli  $Z$  távolságokra a  $Z=a+b/z$  összefüggés segítségével, ahol  $a$  és  $b$  a kamera vágósíkainak távolságát jelenti. Figyelembe véve a leképzést, a képtérben megjelenő elmosódás mértéke arányos lesz az egyes felületi pontok képtérbeli távolságának és a gyújtótávolság közötti különbséggel:

$$r = |Z - P'|S,$$

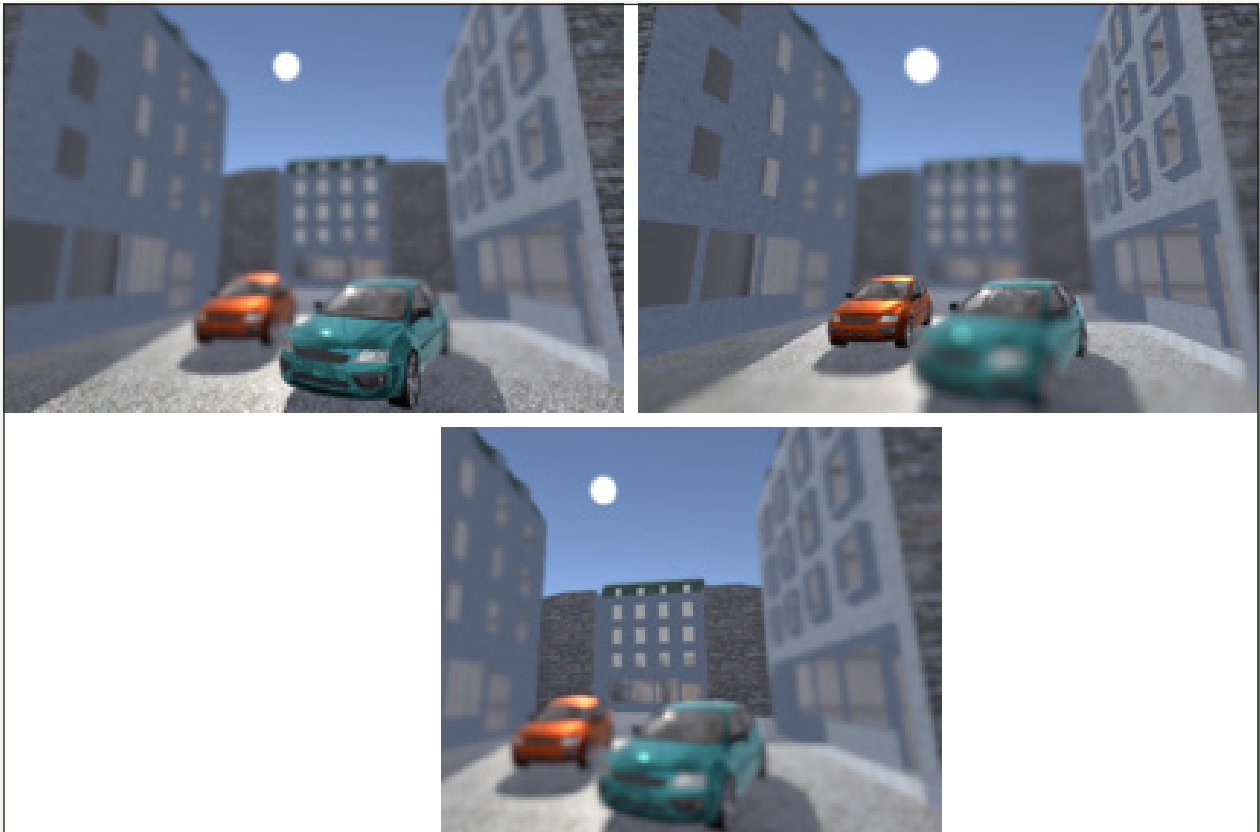
ahol  $S=Ddb/2$  a kamera skálázási paramétere, amely a lencse méretéből, a képsík távolságából, illetve az első és hátsó vágósík közötti távolságból áll. Ebből következően a színtérbeli felületi pontokról a kamera irányába szórt fény a képsíkon egy változó méretű körre válik (4.5. ábra) úgy, hogy a képsíkra jutó fény mennyisége nem egyenletesen oszlik el a körön belül, hanem a középponttól való távolsággal arányosan csökkenve. A kamera gyújtótávolsága közelében lévő pontok kisebb területre összpontosítódnak, a gyújtótávolságtól távolodva egyre nagyobb körre válnak. A felületi pontokról visszavert fény mennyiségét állandónak feltételezve, az egységnyi felületre eső fény mennyisége a kör átmérőjével arányosan csökken.



4.5. ábra A tárgy távolság hatása a kép élességére.

Az eddig vázolt elméleti háttér a mélységélesség hatását a színtérbeli tárgyak felületi pontjairól a kamera irányába szórt fény szempontjából közelíti meg, azonban a számítógéppel történő képelőállítás során hatékonyabb megközelítés a kép pixeleinek szempontjából vizsgálni a problémát. Kiindulásként feltételezzük, hogy a szomszédos képpontokban látható felületi pontok távolsága hasonló. Ebben az esetben, a szomszédos képpontokban található színértékeket a távolságukkal arányos módon bele kell mosni az aktuális képpontba (4.6. ábra). Ehhez egy olyan szűrő függvényt érdemes választani, amely a középpontban nagy értékű, majd a távolsággal arányosan csökken és egy határ után elhanyagolható válik. Az előző szakaszban bemutatott Gauss-függvény természetes választás, hogy a függvény  $\sigma$  szórás paraméterét az elmosás  $r$  átmérőjére választjuk. A mélységélesség hatás szimulációja során a színértékek mellett szükségünk lesz az egyes képpontokban látható felületi pontok képtérbeli távolságára is, így első lépésként ezeket kell előállítanunk. Amennyiben ezek rendelkezésünkre állnak, akkor a szűrést minden egyes pixelre el tudjuk végezni. Tekintve, hogy az alkalmazott szűrés nem tér-invariáns, hiszen az egyes képpontokban eltérő átmérőjű szűrőre van szükségünk, nem tudjuk alkalmazni a korábban bemutatott szűrő szeparálást. A fon-

tosság szerinti mintavétel elvét azonban érdemes ebben az esetben is kihasználni, hogy a számításgényt csökkenteni tudjuk. A bemutatott szűrő eljárás feltételezi, hogy az aktuális képpont környezetében a kamerától vett távolság hasonló, amely hibát okozhat az objektumok határainál azzal, hogy az előtér és háttér színét összemosza. Ezt el tudjuk kerülni úgy, hogy a szűrés során azoknak a képpontoknak a színét nem vesszük figyelembe, amelyek kamerától vett távolsága lényegesen eltér a középpontban mért távolságtól.



4.6. ábra Mélységélesség effektus hatása eltérő fókusztávolságok esetén.

#### 4.4 BEMOZDULÁSOS ÉLETLENSÉG

A számítógépes grafikában alkalmazott lyukkamera modell abban a tekintetben is eltér a valódi kamerától, hogy a képsíkra vetült fényt végtelenül rövid időtartam alatt rögzíti. A fizikai kamerák esetén a képrögzítés ideje véges, ráadásul bizonyos határok között szabadon megválasztható. Ez az úgynevezett expozíciós idő határozza meg azt, hogy a képek készítése során a valós világ mekkora időszelét szeretnénk megörökíteni. Az expozíciós idő megválasztásának elsősorban mozgó témák esetén van jelentősége. Gyorsan mozgó tárgyakat rövid expozíciós idővel tudunk élesen megörökíteni, ellenkező esetben a mozgás iránya mentén a tárgyak elmosódnak a képen. Természetesen a mozgásból fakadó elmosódás jelenségét művészi célra is felhasználhatjuk, ezzel érzékeltetve a téma időbeliségét a felvételen.





**4.7. ábra A bemozdulásos életlenség effektus hatása.**

A számítógéppel készített képeken is utánozhatjuk ezt a jelenséget, amely bizonyos esetekben jelentősen hozzájárulhat az elkészült kép valósághűségéhez. A nem valósídejú képalkotás során az egyik elterjedt megoldás az, hogy a kívánt expozíciós idő alapján meghatározzuk a virtuális világban található tárgyak mozgását és a számítások során a lehetséges elhelyezkedések közül véletlenszerűen választunk. Az ilyen sztochasztikus megoldások azonban nehézkesen alkalmazhatók abban az esetben, ha a képelőállítást valós időben szeretnénk végezni. Közelítő megoldásként azonban jó eredménnyel alkalmazható az a módszer, amely során a korábban elkészült képekből állítjuk elő a mozgásból fakadó elmosódást. A közelítő megoldás alapgondolata az, hogy az egyes képeken a mozgó objektumok kicsit máshol vannak, így ezen képeket átlagolva kirajzolódik a mozgás pályája. Ez a megoldás jól működik abban az esetben, ha az egyes képek között eltelt idő alatt a színtérben található mozgó tárgyak csak kicsit mozdulnak el, azaz a képfriessítési frekvencia arányos a mozgás sebességével. Amennyiben a képfriessítési frekvencia a mozgás sebességéhez képest alacsony, akkor az egyes képeken a mozgó tárgyak képe már élesen el tud különülni, ami zavaró lehet a néző számára. Ennek ellensúlyozására érdemes a képek átlagolása során egy elmosó szűrést is alkalmazni, amely folytonossá tudja tenni a mozgás képét. Az elmosásra ezen hatás esetén is jó választás lehet a Gauss-függvénnyel való szűrés. Nagyobb expozíciós idő esetén tovább javíthatunk a valósághűségen akkor, ha a szűrés során figyelembe vesszük a mozgás irányát és az elmosáshoz választott szűrő függvényt is ennek megfelelően módosítjuk. A 4.7. ábra a bemozdulásos életlenség hatását mutatja be, a bal oldalon látható jármű mozog, így körvonalai a mozgásának megfelelően elmosódnak.

## 4.5 ÁRNYÉKSÁVOK

A valós idejű számítógépes grafikában alkalmazott algoritmusok általában azt feltételezik, hogy a szintér felületi pontjai és a kamera közötti térrészben nincs semmi, ami megváltoztatná a felületi pontot elhagyó fénymennyiséget azelőtt, hogy azt regisztrálnánk a kamerába jutva. Alapvetően ez megfelelő közelítés, azonban vannak esetek, amikor a levegőben lejátszódó szóródási folyamatokat valamilyen módon modellezve nagyban javíthatunk az elkészült látványon. Ebben a szakaszban az úgynevezett árnyéksávok jelenségének közelítését mutatjuk be, amelyek megjelenhetnek akár a természetben a felhők árnyékaiként a hajnali, illetve esti égbolton, vagy akár zárt térbe beszűrődő fénypázmák képében is. Ezen térfogati szóródási jelenségek mögött az a felismerés áll, hogy a levegőben jelen levő parányi porszemek képesek elnyelni a rajtuk áthaladó fénysugarak egy részét, vagy megváltoztatni azok irányát. A szóró közeg hatását a rajtuk áthaladó fénysugarakra az úgynevezett  $\sigma$  írja le, amelynek szokásos alakja

$$\frac{dL(\vec{x}(s), \vec{\omega})}{ds} = -\sigma_t L(\vec{x}(s), \vec{\omega}) + \sigma_s \int_{\Omega'} L(\vec{x}(s), \vec{\omega}') P(\vec{\omega}', \vec{\omega}) d\vec{\omega}',$$

ahol  $L(\vec{x}(s), \vec{\omega})$  az  $\vec{x}$  felületi pontból az  $\vec{\omega}$  irányba visszavert fénysűrűséget,  $\sigma_t$  a közeg kioltási tényezőjét,  $\sigma_s$  a közeg szóródási tényezőjét jelenti. A változók feletti nyilak az úgynevezett vektor mennyiségeket jelölik, amelyek esetünkben három dimenzióval rendelkeznek. A  $P(\vec{\omega}', \vec{\omega})$  az úgynevezett fázisfüggvény, amely a szóródási irányok valószínűségét írja le. Bár az egyenlet első ránézésre ijesztő lehet, valójában azt fogalmazza meg, hogy egy felületi pontból a kamera irányába induló fényfotonok a közegben megtett  $S$  hosszú szakaszokon  $\sigma_t$  valószínűséggel elnyelődnek vagy szóródás következtében eltérülnek a kamerához vezető iránytól. Az egyenlet második tagja azon fotonok hozzájárulását írja le, amelyek eredetileg nem a kamera felé tartottak, azonban egy szóródási esemény hatására az irányuk úgy módosul, hogy útjuk végén a kamerában kötnek ki. Az első tag önmagában elegendő az olyan jelenségek nagyvonalú közelítésére, mint például a köd, amelyben a megtett út négyzetével arányosan csökken a fénysűrűség. A második tag megoldása nem ennyire egyszerű, mivel a térfogat minden pontja összefüggésbe hozható minden más ponttal. A megoldásra analitikus módszert nem ismerünk, általában valamilyen valószínűségi modellel vagy Monte-Carlo integrálással közelítjük. Sok esetben nincs szükség ilyen pontosságra, sokkal inkább a hatás jellegét szeretnénk megragadni. Ehhez általában elegendő csak az egyszeresen szóródott fotonokat figyelembe venni, amely lehetőséget nyújt az egyenlet lényeges egyszerűsítésére. Az egyszeres szóródásra szorítás miatt az egyenlet második tagja nem függ már a keresett  $L$  sugársűrűségtől, így jelöljük  $L_t(\vec{x}(s), \vec{\omega})$ -val. Az átalakítás után kapott

$$\frac{dL(\vec{x}(s), \vec{\omega})}{ds} = -\sigma_t L(\vec{x}(s), \vec{\omega}) + L_t(\vec{x}(s), \vec{\omega})$$

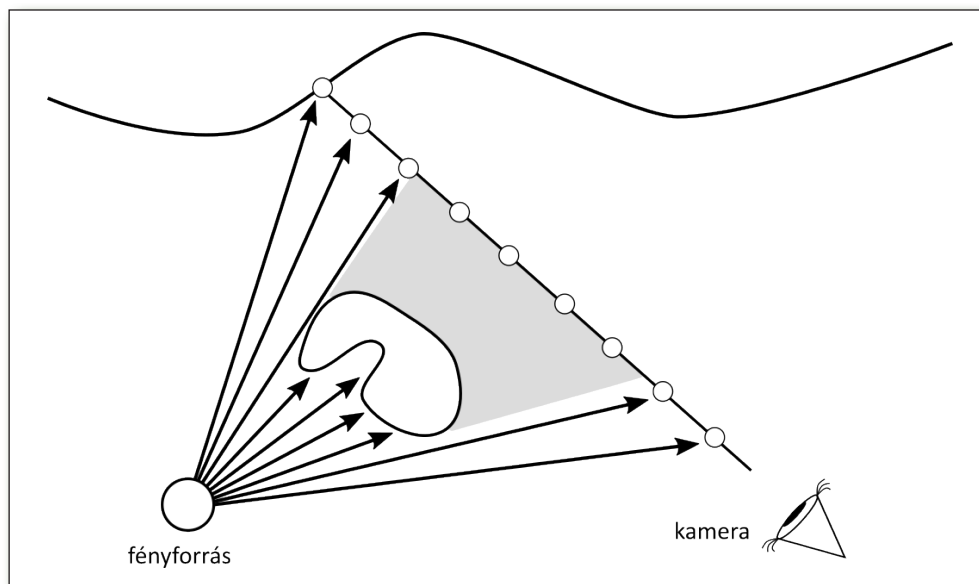
egyenlet már analitikusan megoldható lesz az

$$L(\vec{x}(s), \vec{\omega}) = e^{-\sigma_t s} L(\vec{x}_0, \vec{\omega}) + \int_0^s L_t(\vec{x}(l), \vec{\omega}) e^{-\sigma_t(s-l)} dl$$

alakban, amelyben a jobb oldalon található integrált közelíthetjük véges összegként:

$$L(\vec{x}(s), \vec{\omega}) \approx e^{-\sigma_t s} L(\vec{x}_0, \vec{\omega}) + \sum_{n=0}^N L_t(\vec{x}(l_n), \vec{\omega}) e^{-\sigma_t(s-l_n)} \Delta l,$$

ahol a  $\Delta l = s/N$  lépésköz arányos sugar hosszával és fordítottan arányos a közelítésre felhasznált mintapontok számával. Tovább egyszerűsíthetjük a problémát, ha a színtérben csak egy fényforrás található, hiszen ebben az esetben csak egy  $\vec{\omega}$  irányt kell figyelembe vennünk, amely az egyes mintapontok esetén egyszerű takarási feladattá válik. Ennek megfelelően a közelítő összeg számítása során minden egyes mintapontban el kell döntenünk, hogy az adott pont látszik-e a fényforrásból. Amennyiben látszik, akkor a fényforrásból érkező fotonok hozzájárulását a mintapont és a fényforrás, illetve a mintapont és a kamera közötti irányok alapján a fázisfüggvény segítségével számíthatjuk. Attól függően, hogy a szóródási irány valószínűsége egyenletes-e minden irányban vagy van valamilyen preferált irányultság, a fázisfüggvény lehet izotrópikus vagy anizotrópikus. A hatékony közelítés érdekében általában az egyenletes szóródási irány valószínűséget feltételezzük. A mintapont és a fényforrás közötti láthatóság vizsgálata az árnyéktérképekről szóló szakaszban (2.2. alfejezet) leírtakkal megegyező módon végezhető.

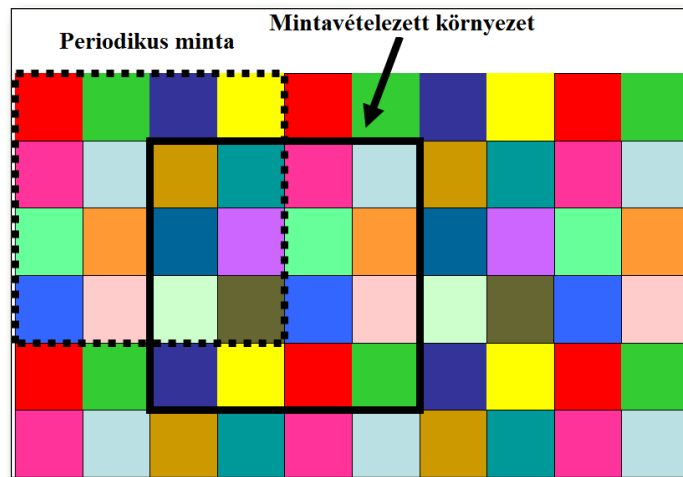


4.8. ábra Az egyszeres szóródás közelítése.

A fénypázmákat számító módszert utófeldolgozási lépésként építhetjük be a képelőállítás folyamatába. Első lépésként minden egyes képpontban meghatározzuk a látható felületi pontot és az onnan a kamera irányába visszavert sugársűrűséget. Ez praktikusán az árnyalás során előálló színértékeket és a felületi ponthoz tartozó mély-

séget jelenti, amelyek egyébként is rendelkezésre állnak. Ezt követően kis lépésekben végig haladunk a felületi pontot és a kamerát összekötő szakaszon a felületi pontból indulva. Minden egyes mintapontban kiszámítjuk az  $L_i(\vec{x}(l_n), \vec{\omega})$  beszóródási faktort a mintapont és a fényforrás között, illetve a  $e^{-\sigma(\vec{x}-l_n)}$  elnyelődési faktort a mintapont és a kamera között, majd ezek szorzatát hozzáadjuk a sugár mentén gyűjtött sugársűrűséghez. Amikor a sugár mentén elérjük a kamerát, akkor az összegyűjtött sugársűrűséget hozzáadjuk a képpontban található sugársűrűséghez. Amennyiben több fényforrás is található a színtérben, akkor ezeket a lépéseket minden egyes fényforrásra megismételjük. Ezt a folyamatot mutatja be a 4.8. ábra.

A megfelelő közelítés és így a jó képminőség elérése érdekében a sugarak mentén kiértékelt mintapontok számának helyes megválasztása elengedhetetlen. Túlságosan ritka mintavétel esetén a fénypátszámok határain tapasztalható finom átmenet hirtelen ugrásokká válik, amely jelentősen ront a képen. Növelve a sugarak mentén kiértékelt minták számát ez a hiba kiküszöbölhető, azonban a számításigény is jelentősen növekedhet. Az alacsony számítási igény biztosításának érdekében kihasználhatjuk azt a tényt, hogy a szomszédos képpontokban látható felületi pontok, illetve az egyszeres térfogati szóródásból származó hozzájárulás igen hasonló. Ebből következően a szomszédos képpontokban számított eredményeket felhasználhatjuk a számításaink pontosságának növelésére, az úgynevezett átlapolt mintavételezés módszerének segítségével.



4.9. ábra Periodikus mintakészlet az átlapolt mintavételezés számára.

Az átlapolt mintavételezés alkalmazásához felosztjuk a képsíkot  $M \times M$  képpontból álló blokkokra és periodikusan szétosztunk a sugarak mentén  $N$  mintapontot. Ezt minden további nélkül megtehetjük, hiszen a szóródási hozzájárulás közelítésére alkalmazott összeg tagjai egymástól függetlenek, azaz értékük nem függ az összeg többi tagjától. A 4.9. ábra egy lehetséges mintakészlet elrendezést mutat be  $4 \times 4$ -es blokkok esetén, amely elosztást a teljes képernyőre megismételve tetszőleges blokkot választva is biztosított az eltérő mintavételezés a szomszédos sugarak mentén. A mintapontok szétosztása után az egyes sugarak mentén a korábbi  $N$  minta kiértékelése helyett már csak  $N/M^2$  minta értékét kell kiszámítanunk. Feltételezve, hogy egy  $M \times M$ -es blokkban található képpontok nagyjából azonos nézeti irány mentén találhatóak, az egyes képpontokban számított összegek összeadhatók és eredményük hasonló lesz ahhoz, mintha az egyes sugarak mentén  $N$  mintát vettünk volna. Ez a feltételezés általában megállja a helyét, mivel a blokkhoz tartozó képpontokból kiinduló sugarak hasonló felületi pontokat érnek el a színtérben. A minták hozzájárulásának végső összegzését egy újabb feldolgozási lépésben tehetjük meg, amely során kihasználhatjuk a kétdimenziós összegképzés szeparálhatóságát és a 4.2. alfejezetben bemutatott módon

visszavezethetjük két egydimenziós számításra, amely nagyobb blokkméretek esetén sem növeli lényegesen a számításigényt. A 4.10. ábra az bemutatott módszerrel készült árnyéksávok hatását mutatja be.



4.10. ábra Összetett színtér árnyéksávok nélkül (balra), és árnyéksávok hozzáadásával (jobbra).

#### 4.6 KÉPTÉRBELI KÖZVETETT MEGVILÁGÍTÁS

Az interaktív számítógépes képelőállítás esetén általánosan alkalmazott lokális illuminációs módszer, amelyet a 2.1. alfejezetben részletesen bemutatunk, a lehető legnagyobb teljesítmény elérése érdekében igen erős egyszerűsítésekkel él. Annak következményeként, hogy az egyes felületi pontokat egymástól függetlenül számítjuk, elveszítjük a felületek egymásra hatásából fakadó jelenségek megjelenítését. Ilyen jelenségek a felületek által visszavert fénysűrűségből fakadó közvetett megvilágítás és a takarásból fakadó árnyékok, amelyek jelentősen növelik a képek valósághűségét. A közvetett megvilágítás globális közelítésére számos módszer létezik, azonban számításigényük általában nem megfelelő interaktív alkalmazások esetén. Azonban megfelelő kompromisszumok mellett lehetőségünk nyílik az elfogadható minőségű és költségű közelítésre. Lehetséges kompromisszum, ha nem a teljes színtérben lejátszódó fizikai folyamatokat próbáljuk meg utánogni, hanem csak az egyes árnyalt pontok valamely véges környezete alapján próbáljuk megragadni a legerősebb hatásokat.

A lokális illuminációs módszerek az indirekt megvilágítás elhanyagolásából fakadó energiahiányt az úgynevezett ambiens fényforrás hozzáadásával pótolják, amely a tér minden pontjában azonos sugársűrűség kibocsátással rendelkezik. Ennek hatására a közvetlenül nem megvilágított felületeket is éri valamekkora sugársűrűség, azonban ez a közelítés nem képes figyelembe venni, hogy az egyes felületi pontok mennyire vannak a környezetük által takarva. Ez a takartság egészen más például egy síkfelületen, mint egy henger belsejében. A síkfelület minden pontja a felületre illesztett félgömb tetszőleges irányából látszik, míg egy henger belsejében az irányok erősen korlátozottak. A takartsággal arányosan a felületre jutó sugársűrűség is csökken, azaz a jobban takart felületi pont be-sötétedik. Az árnyalási egyenletnek megfelelően az  $\vec{s}$  felületi pontban visszavert  $L$  sugársűrűség megkapható az

$$L_r(\vec{s}) = \int_{\Omega} L^{in}(\vec{s}, \vec{\omega}) \frac{\alpha(\vec{s})}{\pi} \cos^+ \theta d\omega$$

összefüggésnek megfelelően, ahol  $\Omega$  a felületi pontra illesztett gömbön felvehető irányokat,  $L^{in}(\vec{s}, \vec{\omega})$  az  $\vec{x}$  irányból érkező sugársűrűséget,  $\theta$  a beesési irány és a felületi normális közötti szöget, végül  $\alpha(\vec{s})$  a felület albedóját

jelenti, amely megadja beérkezett és visszavert sugársűrűség arányát. Amennyiben az  $\vec{x}$  irányból nem látható másik felület, akkor a beérkező sugársűrűség megegyezik az  $L^a$  ambiens sugársűrűséggel és az adott irányt nyílnak nevezzük. Ellenkező esetben az irányt zártnak nevezzük, és a beérkező sugársűrűség az adott irányban található  $\vec{o}$  felületi pont által visszavert sugársűrűség lesz. A fenti összefüggésből következően tetszőlegesen távoli felület is hatással lehet az ambiens megvilágítás mértékére, amely ellentmond a mindennapi megfigyeléseinknek. Ennek az ellentmondásnak a feloldására figyelembe vesszük, hogy a valóságban a felületi pontok közötti térrészben nem vákuum van, azaz a levegő szóró közegként viselkedik. Mint azt az előző szakaszban láthattuk, a szóró közeg a benne megtett úttal arányosan csökkenti a sugársűrűséget a kiszóródás és elnyelődés következtében. Ezt figyelembe véve bevezethetünk egy távolság alapú  $\mu(d)$  mértéket, amely megadja, hogy egy adott irány mennyire zárt a felületi pontok közötti távolság figyelembe vételével. A szóró közegből levezethető távolság függvénynek közvetlen alkalmazása azonban nem jó megoldás, mivel a függvény tetszőleges távolságban is nullánál nagyobb értéket vesz fel. Praktikus megfontolásból érdemes olyan távolság függvényt választani, amely monoton növekvő a  $[0,1]$  tartományon és szabadon választható  $R$  véges távolságon belül eléri maximumát, így takartság számítása során csak véges tartományt kell figyelembe vennünk. Ilyen függvény lehet például a következő egyszerű polinom függvény is:

$$\mu(d) = \left(\frac{d}{R}\right)^{\alpha},$$

ahol az  $\alpha$  paraméter segítségével a függvény felfutásának gyorsaságát adhatjuk meg. A  $\mu(d)$  mérték segítségével az árnyalt felületi pontból visszavert sugársűrűséget a

$$L_r(\vec{s}) = a(\vec{s})(L_a O(\vec{s}) + I(\vec{s}))$$

alakban kaphatjuk meg, ahol  $O(\vec{s})$  a felületi pont ambiens takartságának mértéke,  $I(\vec{s})$  pedig az árnyalt pontba a környező felületi pontokból visszavert sugársűrűség:

$$O(\vec{s}) = \frac{1}{\pi} \int_{\Omega} \mu(d) \cos^+ \theta d\omega,$$

$$I(\vec{x}) = \frac{1}{\pi} \int_{\Omega} (1 - \mu(d)) L_r(\vec{o}) \cos^+ \theta d\omega.$$

A felületi pont környezetből érkező fénysűrűség számítása esetén megtehetjük, hogy a környezet egyes  $\vec{o}$  pontjaiból visszavert sugársűrűség helyett az árnyalt pont környezetének  $L_r(\vec{s})$  átlagos sugársűrűségét alkalmazzuk, így az indirekt tag a következőre összefüggésre egyszerűsödik:

$$I(\vec{s}) = L_r(\vec{s})(1 - O(\vec{s})).$$

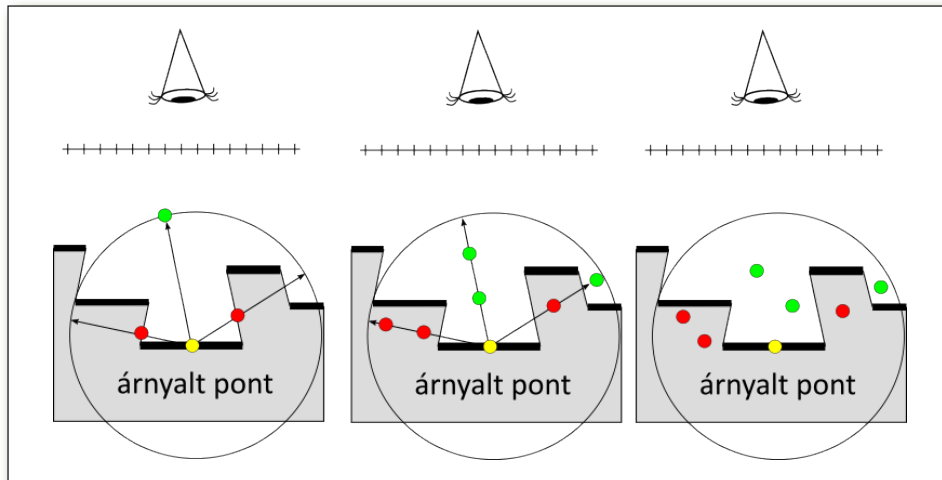
A környezeti takartság figyelembe vételéhez egy felületi pont árnyalása során ismernünk kell azt a véges környezetet, amelyben a takartsági és a szórt megvilágítási faktorokat számoljuk. Ehhez a legegyszerűbb a kép számítá-

sa során előálló mélységi térképet felhasználni, hiszen ezt úgyis előállítja a GPU az árnyalás során. A mélységtérképben az egyes képpontokban látható felületi pontok kamerától vett távolsága található, amely mintavételezett magasságmezőként is felfogható. Előnyös tulajdonsága ennek a megközelítésnek, hogy minden további nélkül alkalmazhatóak azok az algoritmusok, amelyeket a magasságmezők, illetve az implicit felületek megjelenítésére dolgoztak ki.

A magasságmezőt használó képtérbeli algoritmusok feltételezik, hogy a mélység térképben található információ kellő pontossággal írja le a képen látható felületi pontok geometriáját és a szín térkép pedig a felületi pontokból visszavert sugársűrűséget. További előnye a képtérben végzett műveleteknek, hogy a képpontokból induló vetítő sugarak párhuzamosak a  $z$  koordinátatengellyel, amely egyszerűsíti a számításokat. A kamera teréből a képtérbe átvivő transzformáció azonban nem szög és távolság megőrző, így ebben a képtérben ilyen jellegű számításokat nem tudunk végezni. A szintér koordináta-rendszeréből, azaz a világtérből a kamera terébe vivő transzformáció ezzel szemben mind szög, mind távolság megőrző, hiszen alapvetően egy eltolásból és egy elforgatásból áll. A kamera terének hátránya viszont, hogy a vetítő sugarak nem párhuzamosak, hanem a kamera középpontjában metszik egymást. Gyakran alkalmazott megoldás a kamera terének, illetve a képtérnek kedvező tulajdonságainak megőrzésére egy olyan közelítés, amelyben a mélységtérképben szereplő távolságokat a kamera teréből származtatjuk, azonban az egyes képpontokon végzett műveletek szempontjából feltételezzük a vetítő sugarak képtérbeli párhuzamosságát a  $z$  tengellyel. Ezt megközelítést a gépi látás területén gyenge perspektivikus közelítésnek nevezik, és megfelelő kompromisszumot biztosít a pontosság és a hatékonyság között.

A magasságmezőt felhasználva már közelítő megoldást tudunk adni az egyes árnyalt felületi pontok takartságára úgy, hogy a felületi pontra illesztett  $R$  sugarú gömbben irányokat jelölünk ki és ezen irányok mentén megvizsgáljuk, hogy az adott irányú sugárnak van-e metszéspontja a környező felületekkel (4.11. ábra). Ezt a vizsgálatot végezhetjük sugárkövetés segítségével, amelyre magasságmező esetén ismert analitikus megoldás, azonban számításigénye még így is túl magas lehet a valós idejű felhasználáshoz. Alkalmazhatunk a sugárkövetés helyett ray marching jellegű megoldást is, amely során a sugár mentén végighaladva meg tudjuk határozni a felülettel vett metszéspontot. Ezt úgy tudjuk megtenni, hogy először nagyobb lépésekben elindulunk a sugár mentén és minden mintapontban eldöntjük, hogy átléptünk-e a felületen. Amint sikerült megtalálni azt a két egymást követő mintapontot, amelyek közül az egyik a felület egyik oldalán, míg a másik a másik oldalon található, akkor a mintapontok által kijelölt régióban valamilyen kereső eljárással, például bináris kereséssel, meghatározzuk a metszéspont helyét. Ezen megközelítés hátránya, hogy változatos felület esetén a túl nagyra választott kezdeti lépésköz miatt átléphetünk úgy a felületen, hogy azt nem vesszük észre. Túlzottan kicsi lépésköz esetén azonban feleslegesen végzünk számításokat a felület előtti szakaszon. Amennyiben feltételezhetjük, hogy az árnyalt pontra illesztett  $R$  sugarú gömbön belül legfeljebb egy felülettel vett metszéspont található, akkor az egyes sugarak mentén történő keresést egy úgynevezett tartalmazási teszt kiértékelésére csökkenthetjük.<sup>9</sup> A tartalmazási teszt egy véletlen mintapontra eldönti, hogy az a kamerából látható felület előtt vagy mögött helyezkedik el azáltal, hogy a mintapont kamerától vett távolságát összehasonlítja annak a képpontnak a távolságával, amelyre a mintapont vetül.

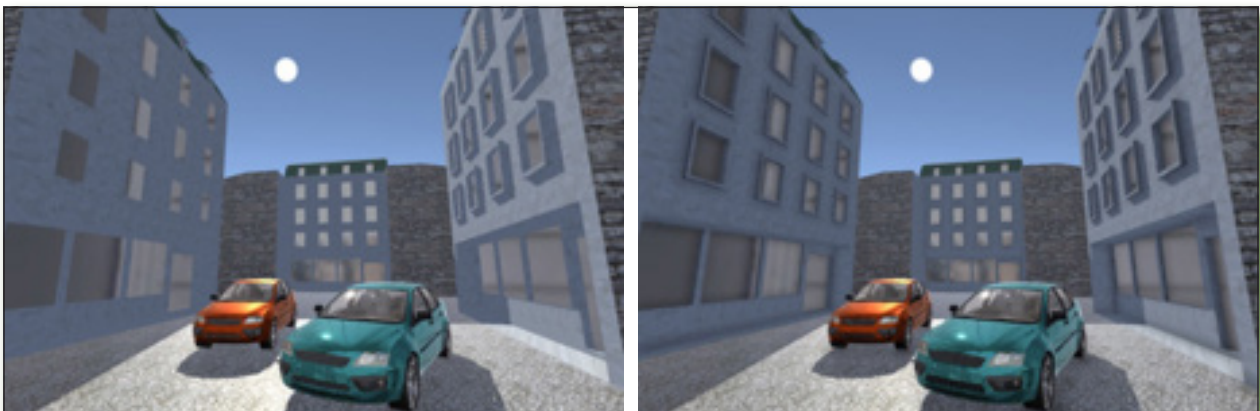
<sup>9</sup> Tóth 2011.



**4.11. ábra Takartság közelítése sugárkövetéssel (balra), sugár masírozással (középen), és tartalmazási teszt segítségével (jobbra).**

A környezeti takarás tartalmazási teszttel történő közelítésének a hibája a kiértékelt mintapontok számával arányosan csökken. A mintapontok meghatározásánál érdemes a fontosság szerinti mintavételezés elvét követni, azaz a mintapontokat a  $\mu(d)$  távolság függvény figyelembevételével felvenni. Amennyiben minden képponthez külön mintakészletet választunk, akkor a közelítés hibája pontszerű zajként fog megjelenni a képen, abban az esetben, ha minden képponthez azonos mintakészletet használunk, akkor a minták korrelációja miatt sávos hibákat kapunk. Tekintve, hogy mind a két hiba nagyon zavaró az emberi szem számára, és csak a minták számának növelésével orvosolható, érdemes ebben az esetben is a korábban bemutatott átlapolt mintavételezéssel élni.

A takartsági faktort kiértékelve az egyes képpontokban az ambiens megvilágítás hatását megfelelően tudjuk közelíteni, azaz a takart régiókban a felület besötétedik, amely megfelel a mindennapi intuícióinknak. A környezeti takarás hatására kialakuló árnyékokat mutatja be a 4.12. ábra.



**4.12. ábra Összetett színtér környezeti takarás nélkül (balra), és a környezeti takarás közelítésével.**



A környezeti takartság hatása általában nem annyira hangsúlyos a képen, mint a közvetlen megvilágítás hiányából fakadó árnyékok, amelyet finom részletekkel egészít ki. A 4.13. ábra ezeket a részleteket mutatja be külön.



**4.13. ábra A környezeti takarással számított árnyékok.**

Az árnyalt pont takartságából következő besötétedés mellett lehetőségünk van az ismertetett módszerrel a közvetlen környezetből visszavert fény hatását is közelítenünk. Ebben az esetben nem csak a takartságot számítjuk ki, hanem a mintapontok alapján az átlagos visszavert sugársűrűséget is. Ezt az átlagos visszavert sugársűrűséget közelíthetjük úgy, hogy a mintapontok kiértékelésénél azokban az esetekben mikor a tartalmazási teszt takartságot mutat, kiolvassuk a színeképet is. A színeképből vett mintákat ezek után a mintavételi ponthoz tartozó takartsági faktoral súlyozva átlagoljuk és hozzáadjuk az árnyalt felületi pontra számított ambiens megvilágításhoz. Amennyiben feltételezzük, hogy a mintapontokból visszavert sugársűrűség hasonló az árnyalt felületi pontból visszaverthez, akkor az átlagos sugársűrűséget közelíthetjük úgy is, hogy az árnyalt felületi pont sugársűrűségét a teljes takartsági faktoral súlyozzuk. Ez a megoldás a mintapontonkénti színekép mintavételezési költségét egy mintára csökkenti, és az esetek nagy részében nem növeli lényegesen a közelítés hibáját. Hátránya ennek a megközelítésnek, hogy az egymáshoz közel levő eltérő színű tárgyak között kialakuló színátfolyás jelenségét nem képes visszaadni.

## 5. NEM-FOTOREALISZTIKUS MEGJELENÍTÉSI ALGORITMUSOK

Az előző fejezetekben számos megoldást mutattunk be, melyek mind abban segítettek, hogy virtuális világunkat minél élethűbben tudjuk megjeleníteni. Előfordulhat azonban, hogy nem az élethűsége törekszünk, hanem valamilyen absztrakt vagy stilizált megjelenítés a cél. Az ilyen technikákat nem-fotorealistikus megjelenítési módszereknek nevezzük (Non-Photorealistic Rendering, NPR).

Alkalmazásuknak több oka is lehet. Leggyakrabban egy bizonyos hangulatot szeretnénk kifejezni, melyet a valóság-hű módszerekkel nem tudunk elérni. Ezek a stilizált megjelenítési módszerek gyakran egy hagyományos képkalkotási technikát próbálnak utánozni, például vízfestményt vagy ceruzarajzot. A stilizált megjelenítésnek az absztrakció is lehet a célja: nem szeretnénk, ha az árnyalás elvonná a figyelmünket a lényegről, vagy az árnyékok esetleg nehezen kivehetővé tennének lényeges elemeket. Az is előfordulhat, hogy mi magunk szeretnénk kiemelni bizonyos részleteket, melyeket árnyalással nem tudnánk (erre egy egyszerű példa a kontúrozás, ahol a geometriai határokat hangsúlyozzuk ki).

A következő alfejezetekben olyan nem-fotorealistikus technikákat mutatunk be, melyek valós időjű alkalmazásokban, így kiterjesztett vagy virtuális valóság szoftverekben is használhatók.

### 5.1 CEL SHADING

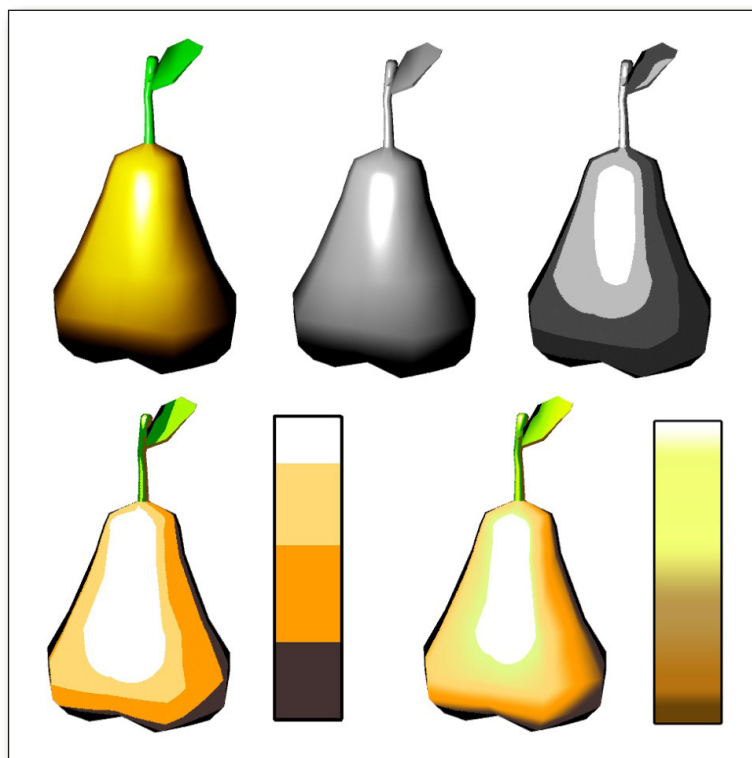
A Cel Shading nevű technika az egyik leggyakrabban használt NPR technika. Nevét a klasszikus rajzfilmeknél használt celluloid lapokról kapta. Ezeknél a rajzfilmeknél megfestették a jelenet háttérét, mely nem változik, így tetszőlegesen részletes technikát használhattak. Az animált karakterek pedig e háttér elé kerültek kirajzolásra egy átlátszó celluloid lapon, amit röviden cel-nek hívtak. Ezekre a lapokra nem lehetett finom árnyalatokat, színátmeneteket festeni, helyette korlátozott számú színárnyalatokat használtak. Így egy objektum színezése során pár tónustartományra osztották az árnyalatokat, egyszínű egymástól jól megkülönböztethető foltként jelentkeztek a sötét, a közép és a világos árnyalatok. A cel lapokat egymásra rétegesen el lehetett helyezni, így a különböző karakterek külön lapokon lettek megfestve, ami növelte a hatékonyságot, illetve az újrafelhasználás lehetőségét.

A Cel Shading során ezt a tónustartományokra osztott megjelenítést próbáljuk leutánozni. Az árnyalás során kiszámoljuk a felületről a kamera felé visszaverődő fény mennyiségét, de elhagyjuk a színi információkat, azaz csak a fényerősséget számítjuk, melyet a legkönnyebben úgy tehetünk meg, hogy a felületek színének a fehéret választjuk az árnyalás kiértékelésekor. Az árnyalást általában lokális illuminációval számítjuk, tükröződések, indirekt megvilágításokat elhanyagolunk. A vetett árnyékokat figyelembe vehetjük (például mélységkép alapú árnyékolással), de az is megesik, hogy ezeket külön kezeljük.

Az 5.1. ábra a módszer alapjait szemlélteti. A bal felső képen egy körtét láthatunk Phong és diffúz BRDF-et használva, lokális illuminációval megjelenítve. A Cel shading első lépéseként az árnyalt szín tónusértékeit számít-

juk ki, ez látható a mellette lévő középső képen. Ha a tónusértékeket négy fix tartományra osztjuk és ezekhez egy értéket rendelünk, megkapjuk a jobb felső képet. A végső lépés, hogy ezekhez a tartományokhoz színértékeket rendeljünk. A bal alsó ábrán az egyes tartományokhoz a sötétől a világosig rendre barna, sötét sárga, világossárga és fehér színeket rendeltünk.

Az egyes színértékeket átadhatjuk paraméterként is, de hatékonyabb, ha egy egyszerű egydimenziós (1 pixel-sorból vagy egy pixeloszlopból áll) textúrát festünk meg, és az egyetlen textúra koordinátát az árnyalási tónus fogja adni. Az így megfestett textúra tehát egy lookup map, mely tónusértékhez színt rendel. Így a tónustartományok számát és az ezekhez tartozó színértékeket egyszerűen és jól értelmezhetően tudjuk tárolni egyetlen képben. A bal alsó képen látható körte mellett a hozzá tartozó tónus textúrát láthatjuk. A tónus textúra viszont azt is megengedi, hogy a tartományok között finomabb átmenetek legyenek, amit ki tudunk használni különböző megjelenítési stílusok megvalósításához. A jobb alsó képen egy finomabb átmenetű árnyalást láthatunk, mely azonban megőrizte rajzfilmszerű jellegét. A tónus textúrák nagy előnye, hogy a különböző stílusok (tónusok száma és azok átmenetei) azonos módon kezelhetők, csak a textúrát kell változtatnunk. Gyakran elő szokott fordulni, hogy a sötétebb területek más színezetet – akár egymás komplementerei – kapnak, mint a világosak, például kék árnyékokhoz narancsos színű fényes területek tartoznak. Ez a színhasználat gyakran megfigyelhető a művészi ábrázolásban a tradicionális technikáknál.



5.1. ábra Objektum stilizált megjelenítése Cel Shading módszerrel.

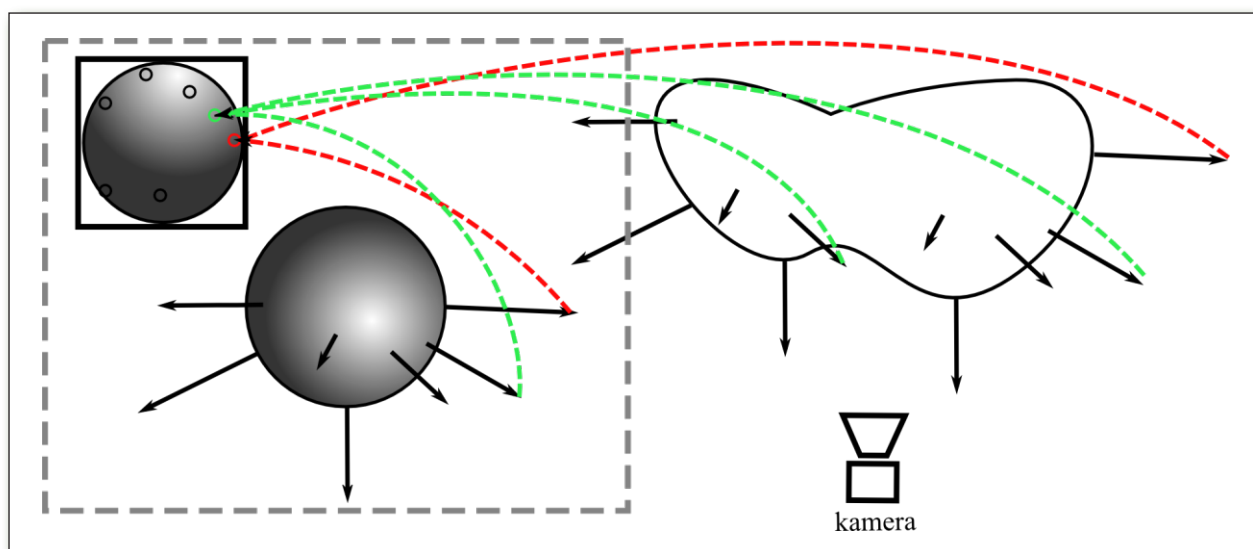
## 5.2 LIT SPHERE MAPPING

A Cel Shading nem képes felületi színváltozásokat kezelni, mivel csak a tónusváltozásokhoz rendel színeket. Vannak azonban olyan technikák, melyek képesek a felületekhez textúrát rendelni, de továbbra is stílusos megjelenítést adnak. Ilyen módszer a Lit Sphere Mapping is.<sup>10</sup>

Ennél a módszernél is egy lookup textúrát használunk, de ebben az esetben már nem egydimenziós, hanem kétdimenziós textúrát. Nagy különbség a Cel Shadinghez képest, hogy ezt a képet nem az árnyalás alapján címezzük, sőt nincs is szükség árnyalásra sem, hanem a felületek irányvektoraival. A módszer azon alapszik, hogy egy adott stílusú felület ábrázolásához egy művész előre elkészít egy képet egy tárgyról az adott stílusban. Ennek a tárgynak a lehető legtöbb felületi irányt tartalmaznia kell, amire a legjobb modell egy gömb.

Ha tehát egy gömböt ábrázolunk, akkor a kapott képen minden kamera irányába néző felületi irány képviselve van, sőt pontosan meg tudjuk mondani, hogy egy pixelhez milyen irány tartozik, és fordítva is: egy irányhoz mely pixel tartozik. Az 5.2. ábra a módszer alapjait szemlélteti. A bal oldalon látható feketén keretezett kép a stílus textúra, amit a felhasználó készít el. A szaggatott vonallal keretezett részben ábrázoltuk azt a virtuális gömböt, amit a felhasználó megfest a stílus textúra megalkotásakor. A gömb egy irányának egy pixel felel meg a stílus textúrában. Két ilyen összekapcsolást jeleztünk is piros, illetve zöld nyilakkal.

Amikor az árnyalni kívánt objektumot jelenítjük meg, annak felületének irányvektorai megfeleltethetők a virtuális gömb egy irányvektorával, így a stílus textúra egy pixelével. Nincs más dolgunk tehát, mint ennek a pixelnek az értékét kiolvasni. Fontos látnunk azonban, hogy ha a virtuális objektum bonyolultabb geometriával rendelkezik, több felületi pontjához is ugyanaz az irány fog tartozni, így a stílus textúra egyes területei többször is megjelenhetnek a végső képen. Ilyen irány például az ábrán a zöld nyíllal jelzett irány is.



5.2. ábra A Lit Sphere Mapping technika alapjai.

<sup>10</sup> Sloan 2001.

Az 5.3. ábra a bemutatott technikával készült képeket mutatja különböző stílus textúrákkal. A bal oldalon a virtuális objektum (teáskancsó) árnyalt képe látható, alatta színekkel láthatjuk egy gömb és a teáskancsó felületi irányait. Az azonos színű felületi pontok tehát azonos irányt jelentenek. Érdekes megfigyelni, hogy a teáskancsó jobb oldalán, csőrén, és fogantyújának jobb oldalán is megjelennek ugyanazok a narancssárga színek, tehát ezek ugyanazt a területet fogják címezni a stílus textúrából. A többi képen alul a felhasznált stílus textúrát, felül pedig a hozzá tartozó végső megjelenítést látjuk.



**5.3. ábra Lit Sphere Mapping technikával megjelenített teáskancsó.**

A Lit Sphere Mapping technika előnye, hogy könnyen implementálható és rendkívül gyors. Kényelmessé teszi a használatát az is, hogy nem igényli textúra koordináták megadását az objektumokhoz, illetve virtuális fényforrások megadását sem. Nagy hátránya azonban, hogy kameratérbeli irányokkal dolgozik, melyek az objektum forgása, vagy a kamera mozgása következtében megváltoznak, így a mintázat elcsúszni látszik a felületen. A stílus textúrákba belefestett árnyalás is a kamerához lesz rögzítve, olyan, mintha a fényforrások (amolyan fejlámpaként) mozognának a kamerával együtt. Ezen lehet enyhíteni, ha objektum térbeli irányokat használunk, de ilyenkor kezelniük kell azt is, hogy nem elég egy félgömbről képet készítenünk, a gömb másik felének kinézetét is meg kell adnunk. Ilyenkor a technika gyakorlatilag egyenértékű egy automatikus textúra koordináta generálási algoritmussal. A módszer másik nagy hiányossága, hogy sík felületeket nem tud kezelni, hiszen azokon nincs irányváltozás, így egyszínű felületeket kapunk.

### 5.3 KONTÚROK

Sok hagyományos művészi technikának szerves részét képezik a kontúrok. Ez részint azért is van, mert a kép elkészítésének első lépése a kontúrok megrajzolása, ami alapot jelent a végső technikának. Ugyanakkor sok technikánál a kontúrokat nem csak a megrajzolandó formák támpontjaiként használják, hanem újból kiemelik. Fontos szerepet játszanak a kontúrok a ceruzarajzoknál, szén- és tusrajzoknál, filcrajzoknál. Például a képregények és a rajzfilmek grafikai világa elképzelhetetlen kontúrok nélkül. A már korábban bemutatott celluloid technikánál is erős kontúrokat rajzoltak.

A számítógépes grafikában a kontúrok rajzolására számos módszert fejlesztettek ki melyek különböző minőségi jellemzőkkel és számításgigénnel rendelkeznek. A módszereket alapvetően két fő csoportra oszthatjuk: képtérbeli és objektumtérbeli. Ezen módszerek közül tárgyaljuk a legfontosabbakat a következő fejezetekben.

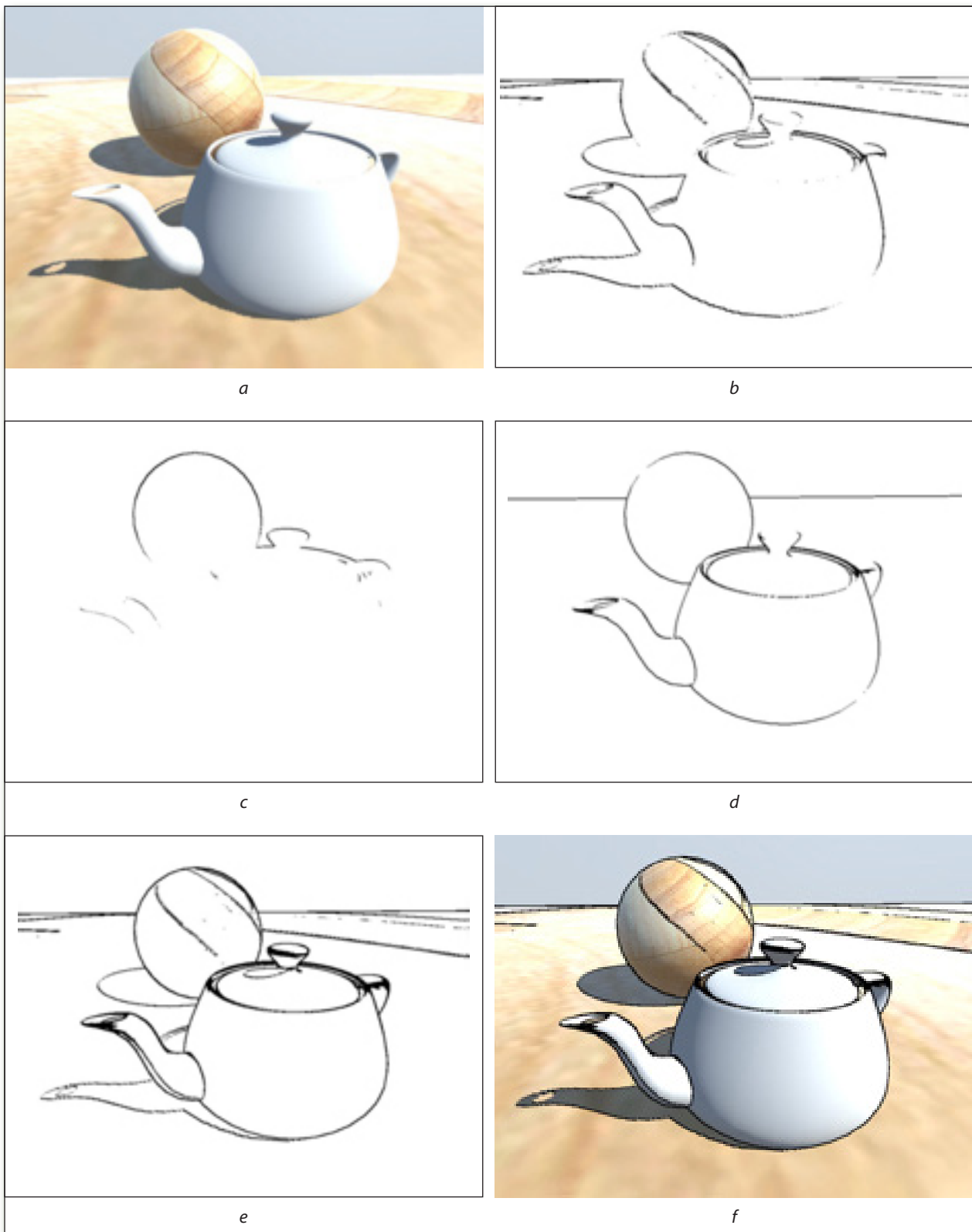
### 5.3.1 Képtérbeli kontúrozás

Valós idejű alkalmazásoknál gyakran használt módszer az éldetektált kontúrozás, mely tisztán képi információkból dolgozik. A módszer lényege, hogy a megjelenített képen képfeldolgozási módszerekkel éleket keres. Erre a feladatra általában egy konvolúciós szűrőt szoktak használni, melyből több, széleskörűen elterjedt verzió is van. A legismertebb talán a Sobel szűrő, mely egy egyszerű  $3 \times 3$ -as konvolúciós szűrő, ami a két fő képtérbeli irány szerint vett deriváltakat összegzi, és kombinálja egy zajcsökkentő szűréssel.

Az 5.4. ábra az éldetektált kontúrok különböző verzióit mutatja be. A bal felső képen a virtuális jelenetet látjuk valószerű árnyalással. A mellette lévő kép az ezen a képen alkalmazott Sobel szűrő eredményét mutatja. Valójában ez nem a szűrő közvetlen kimenete, ugyanis a szűrő eredménye egy gradiens kép, ami sötét ott, ahol nincs változás és világos ott, ahol nagy változás van. Az itt bemutatott élkép két Sobel szűrőt kombinál, egy  $x$  és egy  $y$  irányút, az eredményül kapott gradiens képek abszolút értékeit összeadja, és ha az összeg egy határ alatt van, akkor fehéret, egyébként feketét adunk vissza. Az abszolút értékre azért van szükség, hogy mind a pozitív, mind a negatív, tehát mindkét irányú, gradiens megjelenjenek. Fontos még, hogy húztunk egy határt, ami alatti gradienseket elhagyjuk. Ennek a határküszöbnek a nagysága jelenetfüggő, gondos finomhangolást igényel.

Az árnyalt színek alapján számolt élkép jó kontúrképet eredményez, de vannak hiányosságai. Azokon a területeken, ahol a háttér és az előtér nem különül el megfelelően, nem is jelennek meg kontúrok. Ilyen például a teáskancsó füle és a fedelének tetején megjelenő világos foltok, melyek egybeolvadnak a háttér világos részeivel. A gömb kontúrja is elveszik az árnyékos részeken. A határküszöb csökkentésével esetleg előcsalogathatók lennének ezek az élek is, de ez a többi él megvastagodását és rövid zaj élek megjelenését okozhatja. A színek alapján számolt élkép egyébként is hajlamos a zajosodásra a textúrák bonyolultsága miatt. A mi példánkon ezt a kancsó felett balra, illetve a gömbön lehet megfigyelni, ahol megjelentek a rövid vonalak.

A textúrázott, árnyalt képet éppen ezért nem is igazán szokták élképre használni, helyette más információkat tartalmazó képeken számítanak éleket. A háromdimenziós grafikában nem jelent nagy problémát, hogy olyan ideiglenes képeket készítsünk a jelenetről, mely nem lesz kirajzolva a képernyőre, de fontos információval szolgálhat. Ennek két leggyakoribb formája a mélységkép és a normálvektorokat tartalmazó kép. Ezekre szintén lehet élképet számítani a Sobel szűrő segítségével. A c. jelű ábrán a mélységképből számított élképet láthatjuk. Csak ott jelennek meg kontúrok, ahol két felület kamerától vett távolsága nagyban eltér. Ez az objektumok határainál szokott jelentkezni, kivéve, ha közel kerülnek egymáshoz, amire egy tipikus példa a síklapon álló objektumok. A mi esetünkben is ezt láthatjuk: a teáskancsó és a távoli háttér között jól detektálható a kontúr, ugyanez igaz a gömbre. A gömb és a teáskancsó azonban már túl közel vannak egymáshoz, ugyanez igaz a talajra és a rajta álló objektumokra. A küszöbérték változtatásával javíthatók ezek a hiányosságok, de a lapos szögben látott felületeknél sávosság és hamis kontúrok jelennek meg.



5.4. ábra Éldetektált kontúrok. a. eredeti kép, b. színek alapján számolt élek, c. mélységkép alapján számolt élek, d. normálvektorok alapján számolt élek, e. Összes élkép együttes használata, f. összes élkép kontúrjainak berajzolása az eredeti képbe.

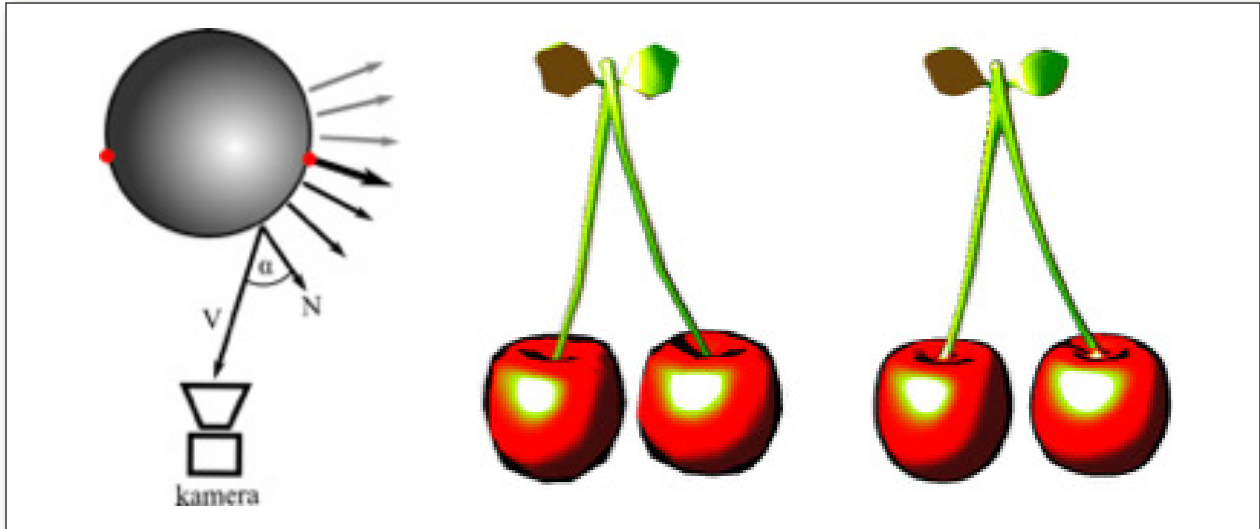
Az utolsó kép, amelyen éleket keresünk a felületek normálvektorait tartalmazó kép. Az ebből számított élképet mutatja a d. jelű ábra. A normálvektorokból kapott élképek ott találnak kontúrt, ahol a felületek iránya hirtelen megváltozik. Ez gyakran igaz az objektumok külső határainál, hiszen kicsi az esély, hogy a mögöttük lévő felület is pont abba az irányba néz. Ha sík lap a háttér, akkor viszont biztosan elő fog fordulni ez az eset is. A teáskancsó felfelé néző felületelemeinél jól látszik a kontúr hiánya. Ezeken a részeken viszont a mélységkép alapú kontúrok teljesítenek jól. A megoldás tehát az lesz, hogy a két élképet együtt használjuk. Az e. jelű képen mind a három élképet kombináltuk, így minden lényeges kontúrt megtaláltunk. A normálvektor alapú élek például jól megtalálják az objektumok határain belüli kontúrokat, a geometriai töréseket, melyek a mélységképen egyáltalán nem, de a színekben is bizonytalanul jelennek meg. Erre példa a kancsó csőrének és testének találkozási vonala. Az f. jelű képen a kontúrokat és az árnyalt képet kombináltuk, ez adja a végső stilizált megjelenítést.

Az éldetektált kontúrok egy képtérbeli technika, mert csak képeken dolgozik, közvetlenül a geometriát nem dolgozza fel. Igaz a geometriai információk megjelennek a mélység és normál képeken, de ettől ezt még képtérbeli technikának tekintjük. Ugyanígy képtérbeli technikának tekinthetjük a következő egyszerű módszert is, mely a sziluettek megjelenítésére alkalmas. A sziluettek azok a körvonalak, ahol a kamera szemszögéből nézve, az objektum határaihoz érünk. Úgy is elképzelhetjük, hogy ha elkezdjük egy test felületén egy pontot követni és haladunk a test széle felé, akkor a sziluett az a pont, ahol a haladási irányunk megfordul, a követett pont az eddig követett felület alá bukik (persze csak a kamera szemszögéből nézve). A felület irányát nézve pedig azt vesszük észre, hogy a sziluettnél a felület iránya elfordul a kamerától: eddig a kamera felé nézett, innentől pedig az ellenkező irányba néz, a hátát mutatja a kamerának.

Az 5.5. ábra bal oldalán egy gömbön illusztráljuk a módszert. A gömb árnyalt pontjaihoz tartozó normálvektorok ( $N$ ) és a kamera irány ( $V$ ) lapos szöget zár be, ha a felületelem látható a kamerából. Ahogy haladunk a test széle fele a bezárt szög egyre nagyobb, eléri a derékszöget, majd tompaszögbe megy át. A derékszögnél található pontok a sziluett pontjai (piros pontok), minden utánuk következő pont már a kamerából láthatatlan. Az objektum árnyalása során tehát megvizsgálhatjuk  $N$  és  $V$  által bezárt szöget ( $\alpha$ ), és ha az elég közel van a derékszöghöz, akkor az árnyalt pont a sziluetten fekszik, így sötétre színezzük. A sziluett kontúr vastagságát az a küszöbérték befolyásolja, amekkora eltérést megengedünk a sziluett pontoknak a derékszögtől. Fontos tudni, hogy az ábrán bemutatott tökéletesen görbült felületeket nem tudunk a grafikus hardveren megjeleníteni, a felület mindig háromszögekkel lesz közelítve. Egy háromszögön belül azonban a normálvektor állandó, vagy legjobb esetben is lineárisan interpolált, ami csak egy közelítést adja az eredeti felület irányvektorainak. Minél durvább a háromszögesítés, annál pontatlanabb a közelítés. Ez nagyban befolyásolja a sziluettek alakját, minőségét.

Az 5.5. ábra középső képén egy cseresznye modellt láthatunk durva háromszögesítéssel, míg tőle jobbra egy finomabb felbontású változatot. Mindkét esetben a felületi irány alapú sziluett detektálást használtuk. Jól látható a durvább felületen, hogy a kontúrok vastagsága változik, a háromszögek határai is felismerhetők. A finomabb felbontású modellnél a kontúrok vastagsága már közel állandó, minőségük sokkal jobb.

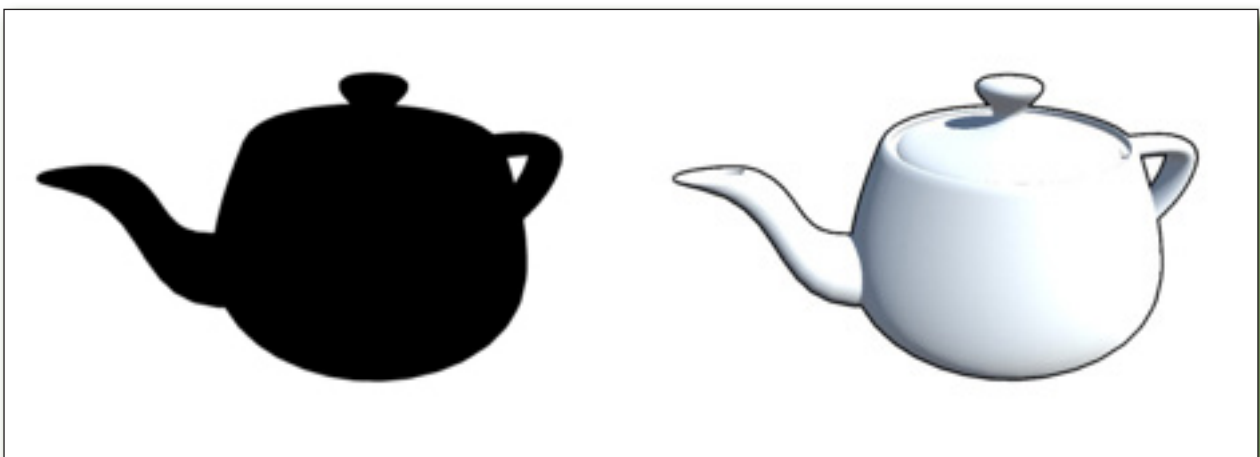




5.5. ábra Felületirány alapú kontúrozás.

### 5.3.2 Objektumtérbeli kontúrozás

Amennyiben a kontúrvonalak létrehozásához közvetlenül az objektumok háromszög geometriáját dolgozzuk fel, objektumtérbeli módszerről beszélünk. Ilyenkor tipikusan új geometria jön létre a kontúrvonalakhoz. Legegyszerűbb formája a sziluett megjelenítés dupla kirajzolással. A módszer lényege, hogy az objektumot először kicsit megvastagítva rajzoljuk ki a sziluett színével és csak erre rajzoljuk rá újból a végleges árnyalással. A két rajzolás lépés eredményeit mutatja az 5.6. ábra.



5.6. ábra Kétmenetes kirajzolással történő sziluett megjelenítés. A bal oldalon az első, sötét színnel történő rajzolás, a jobb oldalon az e fölé történt végső színnel való árnyalás látható.

A módszer megvalósítása egyszerű. A sötét alakzat kirajzolásához a skálázás nem megfelelő művelet, az objektumot kicsit „fel kell fűjni”. Ez annyit jelent, hogy minden csúcspontot a saját normálvektorának irányában kifelé eltolunk. Ennél a kirajzolásnál a takarásért felelős z-buffer algoritmus beállításaira kell még figyelni. Ha más objektumok is vannak a jelenetben, magát a z-tesztet érdemes bekapcsolva hagyni, hogy a közelebbi objektumok ki tudják takarni a sziluettet. A z-bufferbe azonban nem szabad beírni a sziluett mélység értékeit, különben az ki fogja takarni a tárgy végső geometriáját.

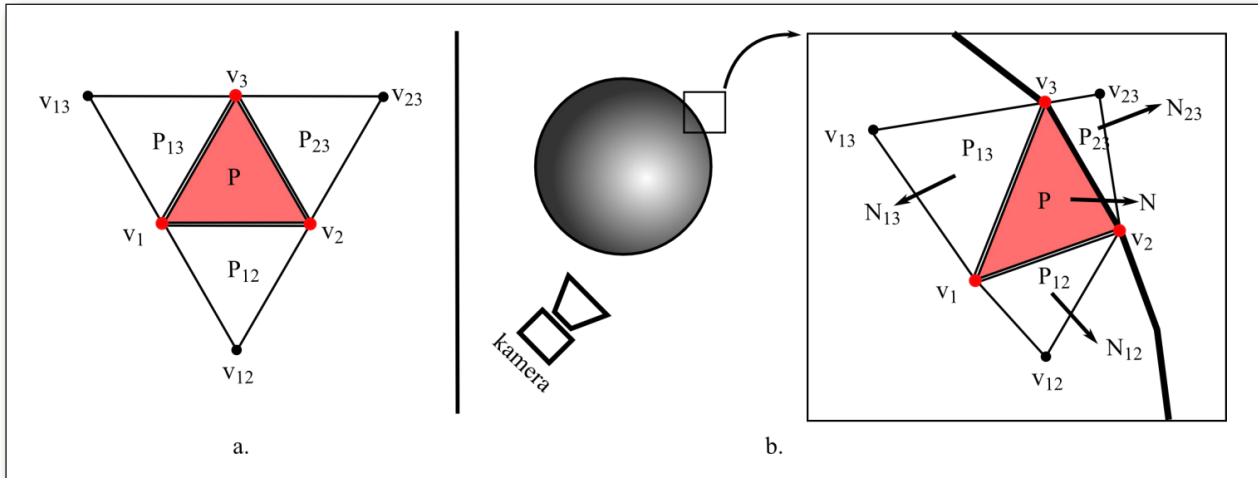
A végső kirajzolás már a szokásos árnyalással történik. Mivel a z-bufferbe nincsenek beírva a sziluett mélység értékei, a végső árnyalt objektum rárajzolódik a sziluett geometriára, mely csak a tárgy szélein fog kilátszódni. Ezt a módszert nem csak stílusos megjelenítésnél használják, hanem annak jelzésére is, hogy egy objektum kitüntetett szerephez jutott. Például egy virtuális valóság alkalmazásban először ki kell választanunk a manipulálni kívánt objektumot. A kiválasztott tárgyat szokás piros vagy sárga sziluettel jelölni.

A dupla kirajzolással történő sziluett megjelenítés a körvonalakat jól megtalálja, de a belső kontúrokat nem képes kezelni. További hátránya még, hogy az objektumot kétszer kell kirajzolnunk. Ez utóbbi igazából nem nagy hátrány, mert az egyszínű sziluett geometria kirajzolása általában elég gyors, hacsak a test geometriai felbontása nem nagyon magas. Bár az egész objektum dupla kirajzolása feleslegesnek tűnik, főleg, hogy annak nagy részére úgyis rárajzolunk, látni fogjuk, hogy más objektumtérbeli módszerek geometria feldolgozási igénye sem alacsonyabb.

A kontúrvonalakat közvetlenül a geometriából úgy is kinyerhetjük, hogy azonosítjuk a háromszögek azon oldaléleit, melyek kontúréleket jelenthetnek, és ezeket az éleket külön, megvastagítva kirajzoljuk. A legfontosabb felmerülő kérdés, hogy hogyan döntjük el egy oldalról, hogy kontúr él vagy sem. Ha a háromszögeket önmagukban dolgozzuk fel, akkor természetesen nem tudjuk ezt eldönteni, ha viszont egy háromszög feldolgozásakor ismerjük annak szomszédos háromszögeit is, akkor már tudunk következtetéseket levonni.

Az 5.7. ábra a. jelű képén láthatunk egy háromszöget (P), melynek feldolgozásakor a szomszédos háromszögeit is ismerjük (P12,P23,P13). Erre minden mai grafikus processzor képes, ha a háromszögeket nem három, hanem hat csúcsponttal adjuk meg: a három csúcspontjával melyet a három szomszédos csúcspont követ (az ábra jelöléseivel: v1,v2,v3,v12,v23,v13). Ismerve a lokális geometriát, már eldönthetjük egy élről, hogy sziluett él-e. A b. jelű képen egy gömböt és annak geometriájának részletét látjuk nagyítva, a sziluett közelében. A háromszögek csúcspontjainak ismeretében meg tudjuk határozni a háromszögek normálvektorait (N,N12,N23,N13). A feldolgozott háromszög három élére ellenőrizzük, hogy a háromszög és az élnek megfelelő szomszédos háromszög normálvektora a kamera irányába néz-e. Ha ebben különböznek, tehát az egyik a kamera felé nézett, a másik viszont nem, akkor az él a kontúr része. Az ábrán a kontúrt vastag vonallal jelöltük, v2 és v3 között futó háromszöggel tehát kontúrél.

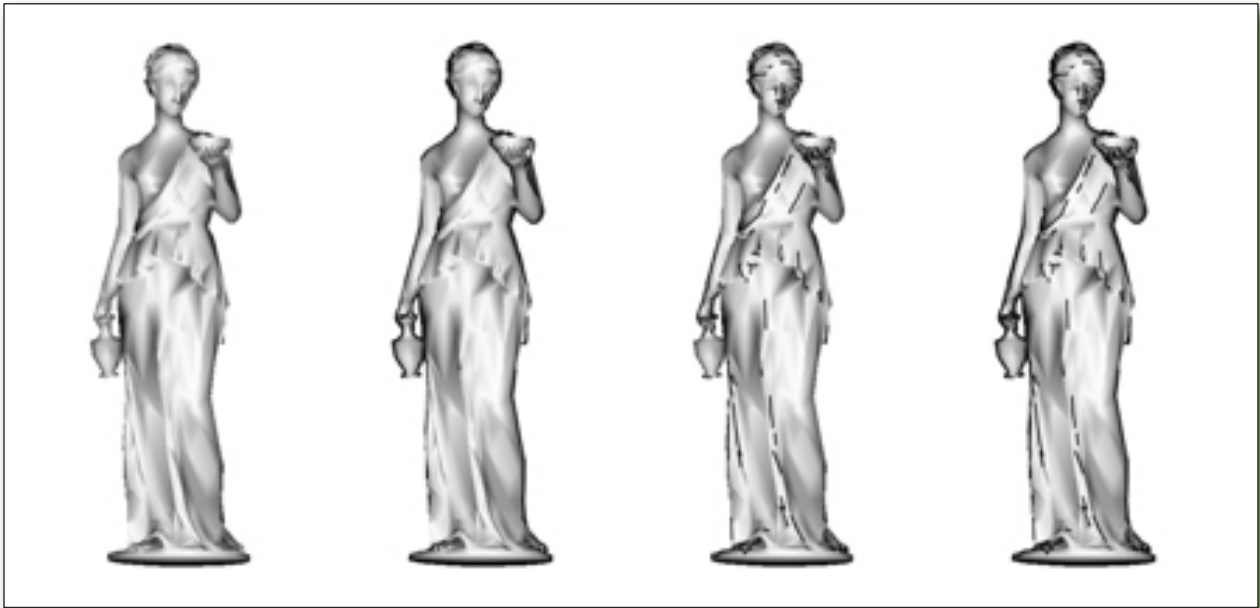
Hasonló módon az objektumok belső éleit is tudjuk azonosítani. Mivel a belső élek ott futnak, ahol a felület megtörik, a szomszédos háromszögek normálvektorai által bezárt szöget kell összevetnünk. Ha ez a szög elegendően nagy, az azt jelenti, hogy a felületben az adott élnél törés található, így az élet kontúrélként azonosíthatjuk.



5.7. ábra Kontúrélek detektálása a háromszög geometriából.

Az itt vázolt megoldásokat a grafikus hardverek teljes mértékben támogatják, a geometria árnyaló által lehetőségünk van minden háromszög feldolgozásakor annak három élére három vonal primitívet előállítani, vagy ezek közül bármelyiket eldobni. A kontúrok ilyen megjelenítéséhez tehát szintén kétmenetes megjelenítés szükséges: első menetben a szokásos árnyalással jelenítjük meg az objektumot, második menetben pedig a háromszögek valamely élére vonal primitívek kerülnek és ezeket sötét színnel rajzoljuk a felületre. A z-buffer algoritmus speciális beállítása itt nem szükséges.

Az 5.8. ábra a háromszög geometriából számolt kontúrokat mutatja. Baloldalon az árnyalt modell látható kontúrozás nélkül. Mellette jobbra a külső kontúréleket jelenítettük meg, ezek azok az élek, ahol a szomszédos háromszögek a kamerához képest átfordulnak. A harmadik kép csak a belső kontúrokat mutatja. Ezek azok az élek, ahol a szomszédos háromszögek normálvektorai nagyon eltérnek, így geometriai törésvonalnak számítanak. Az utolsó képen mindkét típusú kontúréleket berajzoltuk.



**5.8. ábra Geometriából számolt kontúrok. Balról jobbra haladva: árnyalt modell, külső kontúrok, belső kontúrok, külső és belső kontúrok.**

A geometriából számolt kontúrok minősége nagyban függ a geometria felbontásától és minőségétől. Ha a geometria háromszögei nem jól közelítik az eredeti görbült felületeket, tehát zajos felületeket adnak, akkor sok hamis belső kontúr keletkezhet, illetve a külső kontúrok sem mutatnak egyenletes íveket. A nagy felbontású geometria kezelése költséges lehet mind memória, mint számítási kapacitás szempontjából. A memóriaigényt növeli az is, hogy háromszögenként hat csúcspont adatot kell tárolnunk a három helyett, hogy a szomszédossági információkhoz hozzáférjünk.

A módszer előnye viszont, hogy a vonalakat nemcsak vonal primitívként, hanem háromszögszalagként is megjeleníthetjük, melyeket textúrázni is tudunk, amivel a kontúrok stílusának változatossága növelhető. Ez azonban igényli a kontúrélek összefűzését, és a textúrázás időbeli koherenciájának kezelését. Ezen problémák tárgyalása túlmutat a jelen dokumentum keretein, és ezek a módszerek jelenleg még inkább az animációs filmek területén használatosak.

## 6. KOMPOZITÁLÁS

A számítógéppel készített képek vagy képrészletek elhelyezése egy valódi felvételen képezi az alapját a kiterjesztett valóság rendszereknek. Ahhoz, hogy a képeket a szemlélő számára hihető módon tudjuk illeszteni, a lehető legpontosabban reprodukálnunk kell a valódi felvétel környezetében tapasztalható megvilágítási és takarási viszonyokat a virtuális világunkban is. Ez azonban csak a probléma megoldásának egyik szelete, hiszen az elkészült képrészletek beillesztését is úgy kell elvégeznünk, hogy az eredményt már ne lehessen különválasztani az eredeti képrészletektől. A korábbi fejezetekben bemutattunk néhány alapvető algoritmust, amelyek segítségével valósághű képeket állíthatunk elő, ebben a fejezetben pedig felvázolunk néhány módszert a képek hatékony keverésére.

### 6.1 SZÍN ALAPÚ KOMPOZITÁLÁS

A szín alapú montírozást régóta alkalmazzák a stúdió technikában az élő szereplővel készült felvételek esetében a háttér leválasztására és kicserélésére. A montírozás során az előtér felvétel azon régióit, amelyek egy előre kiválasztott szín környezetét tartalmazzák, lecseréljük a háttér felvétel azonos elhelyezkedésű képpontjaival. A hatékony háttér leválasztás érdekében az előtér felvételeket általában valamilyen egyszínű háttér előtt készítik. A háttér színét úgy érdemes megválasztani, hogy lehetőség szerint az előtérben lévő téma elhanyagolható mértékben tartalmazza. Az élő szereplős felvételeknél korábban kék színű háttérrel, míg napjainkban jellemzően a zöld valamilyen árnyalatát alkalmazzák, amely oka az, hogy ezek a színek vannak a lehető legtávolabb az emberi bőr árnyalataitól.

Történelmi érdekesség, hogy a kék háttérszín elsősorban az analóg rendszerekben volt elterjedt, különösen a mozifilmek esetén. Ennek alapvető oka a széles körben elterjedt filmeknek a fény különböző hullámhosszaira való érzékenységében keresendő. A zöld háttér ezzel szemben a digitális műsorkészítés körében terjedt el. A digitális rendszerekben alkalmazott tömörítési eljárások erősen építenek az emberi szem működésére, így például arra, hogy a zöld színekben megjelenő apró különbségekre az emberi szem lényegesen érzékenyebb, mint a kék csatornában tapasztalható változásokra. Ennek megfelelően a kék csatornát erőteljesebben lehet tömöríteni, mint a zöldet, hiszen az ebből fakadó információvesztés nem okoz zavaró hibákat a felvétel megtekintése során. A tömörítés hatása azonban erősen ronthatja a háttér leválasztásának minőségét, így érdemes olyan csatornában tárolni, amely a lehető legkevésbé van tömörítve. Emellett a zöld háttérszín elterjedésében erőteljesen közrejátszott az a tény is, hogy a színes televíziózás kezdetén a kék színű ruhák elterjedtebbek voltak, különösen a férfiak körében, így zöld színű háttér esetén kevésbé kellett odafigyelni a viselt ruha színére.

A napjainkban elterjedt digitális rendszerek esetén sokkal szélesebb eszköztár áll rendelkezésünkre a háttér leválasztására, az alapvető módszerek azonban megmaradtak, így érdemes nagy vonalakban áttekinteni két képfolyam szín alapú montírozásának lépéseit.

A montírozás első lépése azon képpontok meghatározása az előtér tartalmazó képen, amelyek színét ki kell cserélni a háttér megfelelő képpontjának színével. Ez alapvetően egy olyan leképzésként írható le, amely egy adott

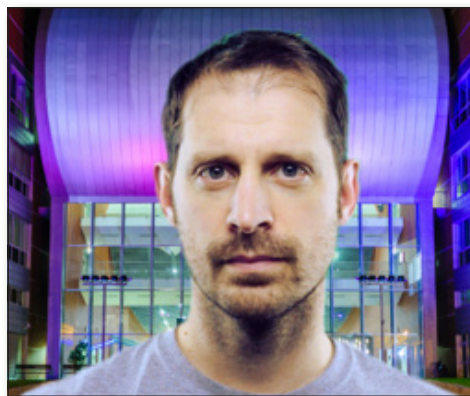
színtérben definiált csatorna értékekhez egy átlátszósági értéket rendel ( $f(c_r, c_g, c_b) = a$ ). Ahhoz, hogy a lehető legegyszerűbben tudjuk meghatározni a szükséges leképző függvényt, érdemes olyan színteret keresnünk, amely a lehető legnagyobb mértékben toleráns a felvételeken jelentkező színhibákra. Ilyen színtér lehet az úgynevezett  $YC_b C_r$ , ahol az  $Y$  csatorna az adott képpont fényességét, a  $C_b$  és  $C_r$  csatornák pedig a kék és vörös irányban értelmezett színességet írják le. A zöld csatornát nem szükséges külön módon tárolni, mivel a színtér kialakítása miatt kiszámítható a kék és vörös csatornák értékeiből. A kilyukasztandó színeket ezek után a  $C_b, C_r$  síkon adhatjuk meg, amely az adott színtartományt a fényességtől függetlenül tudja kiválasztani.



**6.1. ábra Zöld háttér előtt készített felvétel (balra), és a szín alapján készített maszk.**

A leképzés eredményeképpen a képpontokhoz rendelt átlátszósági értékeken ezek után egy küszöbölés jellegű műveletet hajtunk végre, amely a kvázi folytonos átlátszósági értékeket két csoportba rendezi. Az egyik csoport az előtérhez tartozó képpontokat fogja tartalmazni, míg a másik a háttér régiókat. A küszöb megfelelő megválasztásával tudjuk pontosan meghatározni az előtér és a háttér határát, így ez akár képkockánként is eltérhet szélsőséges megvilágítási viszonyok esetén. Az így kialakított maszkok segítségével már össze tudjuk montírozni az előteret és a háttérrel. A 6.1. ábra bal oldalán egy zöld háttér előtt készített felvételt láthatunk, a jobb oldalon a szín alapján készített előtér maszkot. A maszk fehér képpontjai jelölik az előteret, míg a fekete képpontok a háttér régiót. A 6.2. ábra a háttérkicserélés eredményét mutatja.

A szín alapú montírozás, mint az a fentiekből jól látszik, akkor tud igazán hatékony lenni, ha az előtér és a háttér színe élesen elkülönül, valamint a leválasztandó háttérszint egy szűk tartományra tudjuk korlátozni.

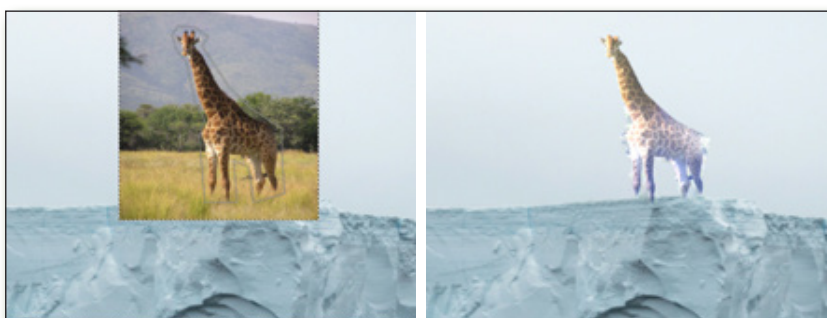


**6.2. ábra Szín alapú kompozitálással kicserélt háttér.**

## 6.2 GRADIENSTÉRBELI KOMPOZITÁLÁS

A maszk alapú kompozitálási módszerek, mint például az előző szakaszban bemutatott szín alapú lyukasztás alkalmazásával járó kötöttségek nem minden alkalmazási területen elfogadhatók. Gyakran nincs lehetőségünk a háttér szabad megválasztására, illetve a megvilágítási viszonyok összehangolására az összeillesztendő képek között. Ennek következményeként a maszkot nem tudjuk automatikusan előállítani, annak szélein zavaró hibák jelenhetnek meg. Ezen problémák feloldására alkalmazhatjuk a gradienstérbeli montírozás módszerét,<sup>11</sup> amely az egyszerű színmásolás helyett egy lényegesen összetettebb eljárást alkalmaz a képek illesztésére. A módszer alapja az, hogy a képeken általában nem maguk a színek, hanem azok változása hordozza a legfontosabb közvetíteni kívánt információt. Ugrásszerű változást tapasztalhatunk az objektumok határainál, a megvilágítás és árnyékok határainál, és természetesen az anyagjellemzőket is a változások írják le a leginkább. Ezeket a változásokat a kép deriváltjával, azaz az egyes képpontok és azok környezete alapján számított gradienssel írhatjuk le. A képek ilyen reprezentációját nevezzük gradienstérnek. A gradienstérbeli reprezentációból annak integrálásával állíthatjuk elő az eredeti képet.

A gradienstérben egy sor képjavító eljárást is alkalmazhatunk, de számunkra most a legfontosabb a képrészletek másolása. Ehhez ki kell választanunk a másolandó régiót az előtér képen, amelyet általában valamilyen kontúrkövető eljárással tehetünk meg. A régió határait nem szükséges teljesen pontosan kijelölni, a legjobb eredmény elérése érdekében azonban érdemes elkerülni a nagyobb változásokat, ha azok nem tartoznak az átmásolni kívánt régióhoz. Következő lépésként előállítjuk a háttér és a kijelölt előtér régió gradienstérbeli reprezentációját, majd átmásoljuk az előteret a háttér képre. Az átmásolt régió gradiense nem feltétlenül kompatibilis a háttérrel, így mielőtt visszaintegráljuk a képet a színtérbe, valamilyen módon ki kell javítanunk a hibákat. Erre több módszert is találhatunk az irodalomban, amelyek közül a legelterjedtebb egy iteratív optimalizáló algoritmus, amely egy olyan képet állít elő, amelynek gradienstérbeli reprezentációja a lehető legközelebb van az általunk előállított gradiensképhez.<sup>12</sup> A közelítő eljárás alapja a diszkrét Poisson egyenlet megoldására alkalmazható lineáris egyenletrendszer. Ennek a megoldásnak a hátránya, hogy a megoldandó egyenletek száma arányos lesz a képpontok számával, amely nagy felbontás esetén nehezen kezelhető lehet.



**6.3. ábra** Gradienstérbeli kompozitálás. Balra a kijelölt előtér régió, jobbra az illesztés eredménye.

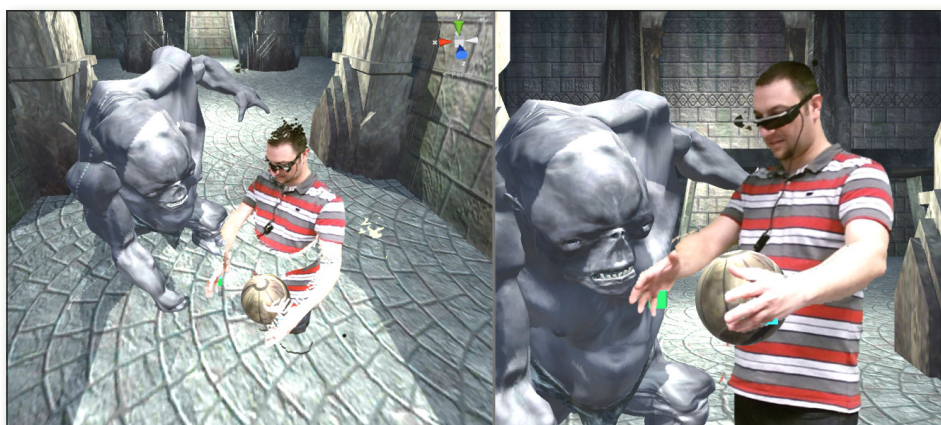
<sup>11</sup> Pérez 2003.

<sup>12</sup> Agarwala 2007.

Példaként a 6.3. ábra bal oldalán az összeillesztendő előtér és háttérét láthatjuk. A másolandó területet a zsiráf körüli folytonos vonal jelöli ki. Az ábra jobb oldalán a kompozitálás eredményét láthatjuk. Azon a területen, ahol az eredeti előtér képen a fák is belelógtak a maszkba, látható némi hiba, a kisebb gradienssel rendelkező területeken ez nem jelentkezik. Jól látható a zsiráf testén az átmenet az előtér és háttér domináns színe között, amelyet a szín-csatornákon egyenként végzett kompozitálás okoz.

### 6.3 PONTFELHŐ BEILLESZTÉS

Az eddig bemutatott kompozitáló módszerek csak az előtér és háttér képek színe alapján végezték az összeillesztést. Ezek a módszerek abban az esetben nehézkesen működnek, ha nem csak egyszerű háttér kicserélést szeretnénk, hanem ténylegesen be szeretnénk illeszteni az új képrészleteket a virtuális világba. Ebben az esetben már nem beszélhetünk előtéréről és háttéről, hiszen az szeretnénk, hogy a beillesztett képrészlet is a virtuális világ szerves része legyen. Ez azt jelenti, hogy a beillesztett kép egyes részleteit a kamerához közelebb eső objektumok kitakarhassák, és akár az árnyalás számításában is részt vehessen. Önmagában a beillesztendő képrészletek színe ehhez nem tartalmaz elegendő információt, szükségünk az egyes képpontok térbeli elhelyezkedésére is. Ennek meghatározására felhasználhatjuk az *Inverz képszintézis* fejezetben bemutatott sztereó rekonstrukciós módszert, amely két párhuzamosan felvett kép alapján képes kiszámítani az egyes képpontok egymáshoz viszonyított elhelyezkedését. Alkalmazhatunk közvetlen távolságkép előállítására is alkalmas képfelvevő eszközöket (például Microsoft Kinect), amelyek általában infravörös vetített minta vagy a kibocsátott fényfotonok által megtett út alapján számítják ki az egyes képpontokban látható felületi pontok távolságát. A szín- és távolságképek alapján már rekonstruálni tudjuk a képen látható objektumok felületi pontjainak térbeli elhelyezkedését és az onnan a kamerába érkező sugársűrűséget. Az így kialakult struktúrát nevezzük *pontfelhőnek*, hiszen valódi geometria és objektum címkék helyett sok egymástól független pontot tartalmaznak, összességükben azonban jól jellemzik a kamerából látható tér elrendezését.



**6.4. ábra** A virtuális világba illesztett pontfelhő, balra a mélységkamerához képest lényegesen eltérő szögből nézve, jobbra a mélységkamera nézőpontjából.



A pontfelhőket közvetlenül beilleszthetjük a virtuális világunkba egy egyszerű 3D transzformáció segítségével, amely a mélységkamera teréből átviszi a pontokat a világtérbe. Ezek után a pontfelhő minden pontja egy elemi felület darabot fog jelképezni és teljes mértékben azonosan kezelhető, mint bármely más felület a megjelenítés során. Amennyiben a mélység kamera nézőpontja közel azonos, mint ahonnan a virtuális világunk képét előállítjuk, akkor a pontfelhő nagyjából folytonos felület képét mutatja, mint az a 6.4. ábra jobb oldali képén láthatjuk. Lényegesen eltérő nézőpontból készítve a képet előfordulhat azonban, hogy a pontfelhő képe kilyukad vagy akár teljesen szét is esik, hiszen ilyenkor olyan felületi pontok megjelenítésére lenne szükségünk, amely nem láthatóak a mélységkamera által készített felvételen. A 6.4. ábra bal oldali képén egy ilyen eltérő nézőpontból készült felvételt láthatunk, a kéz környéki területeken jól látszik, hogy a pontfelhő nem tartalmaz semmilyen információt a jobb kézfejről és a bal tenyéréről. Ennek a problémának kiküszöbölésére általában valamilyen mélységfúziós eljárás alkalmazható, amely több nézőpontból készített távolságképek alapján építi fel a pontfelhőt. Ezen eljárások részletesebb bemutatása azonban már túlmutat ezen mű keretein.

# IRODALOMJEGYZÉK

- ÁBRAHÁM GYÖRGY (1997): *Optika*, Panem Kft., 660. old.
- AGARWALA, ASEEM (2007): *Efficient Gradient-Domain Compositing Using Quadtrees*. *ACM Transactions on Graphics* (Proceedings of SIGGRAPH 2007).
- DEBEVEC, PAUL (1998): *Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography*. In Proceedings of the 25th annual conference on Computer graphics and interactive techniques (SIGGRAPH,98). ACM, New York, NY, USA, 189–198. old. DOI: <http://dx.doi.org/10.1145/280814.280864>
- SLOAN, PETER-PIKE J. – MARTIN, WILLIAM – GOOCH, AMY – GOOCH, BRUCE (2001). *The lit sphere: a model for capturing NPR shading from art*. In Proceedings of Graphics Interface 2001 (GI,01). Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 143–150. old.
- SZIRMAY-KALOS LÁSZLÓ – ANTAL GYÖRGY – CSONKA Ferenc (2004): *Háromdimenziós grafika, animáció és játékfejlesztés*. Budapest, ComputerBooks.
- SZIRMAY KALOS, L. – ASZÓDI, B. – LAZÁNYI, I. – PREMECZ, M. (2005): *Approximate Ray Tracing on the GPU with Distance Impostors*. *Computer Graphics Forum*, 24: 695–704. old. doi:10.1111/j.1467-8659.2005.0m894.x
- SZIRMAY-KALOS, LÁSZLÓ – LAZÁNYI, ISTVÁN (2006): *Indirect diffuse and glossy illumination on the GPU*. In Proceedings of the 22nd Spring Conference on Computer Graphics (SCCG,06). ACM, New York, NY, USA, 25–32. old. PAGE@7. DOI=<http://dx.doi.org/10.1145/2602161.2602164>
- TÓTH, BALÁZS (2013): *Efficient sampling methods for photorealistic rendering on the GPU*. PhD thesis. Budapest.
- TÓTH, BALÁZS – SZIRMAY-KALOS, LÁSZLÓ – UMENHOFFER, TAMÁS (2011): *Efficient Post-Processing with Importance Sampling*. In: Wolfgang Engel (ed) *ShaderX 7*. Boston: Charles River Media, 259–276. old.
- UMENHOFFER, TAMÁS – SZÉCSI, LÁSZLÓ –, SZIRMAY-KALOS, LÁSZLÓ (2011): *Hatching for Motion Picture Production*. *Comput. Graph. Forum* 30 (2011): 533–542. old.
- UMENHOFFER, TAMÁS – SZIRMAY-KALOS, LÁSZLÓ – SZÉCSI, LÁSZLÓ – LENGYEL ZOLTÁN – MARINOV, GÁBOR (2018): *An image-based method for animated stroke rendering*. *The Visual Computer*, June 2018, Volume 34, Issue 6–8, 817–827. old.
- UMENHOFFER, TAMÁS – PATOW, GUSTAVO – SZIRMAY-KALOS, LÁSZLÓ (2007): *Robust Multiple Specular Reflections and Refractions*. In: Hubert Nguyen (editor) *GPU Gems III*. Boston: Addison-Wesley Publishing Company, 2007. 387-408. old.
- PÉREZ, P. – GANGNET, M. – BLAKE, A. (2003): *Poisson image editing*. *ACM Transactions on Graphics* 22, 3 (July), 313–318. old.



# A Nemzeti Közszerológati Egyetem kiadványa



Kiadó:

Nemzeti Közszerológati Egyetem;  
Államtudományi és Közigazgatósi Kar  
[www.uni-nke.hu](http://www.uni-nke.hu)

Felelős kiadó:

Prof. Dr. Kis Norbert dékán

Címe: 1083 Budapest, Üllői út 82.

Olvasó szerkesztő: Kiss Eszter

Tördelőszerkesztő:

Mikes Vivien

ISBN (pdf) 978-963-498-118-3