

Mi a pont és a vektor? Milyen műveleteket végezhetünk el pontokon és vektorokon?

Pont: Az Euklideszi sík/tér egy eleme, amelynek nincs kiterjedése.

Vektor: Geometriailag egy eltolás, aminek iránya és hossza van. A tér egy pontjához azt a másikat rendeli hozzá, ami a vektor irányában a vektor hosszának megfelelő távolságra van.

pont + vektor = pont

pont - pont = vektor

Értelmezve vannak rájuk további műveletek is: Összeadás, kivonás, skalárral szorzás, vektoriális szorzás --> Ezeknek az eredménye vektor

Skaláris szorzat --> eredménye skalár

Összeadás:

$a=[a_1, a_2]$ $b=[b_1, b_2]$ --> $a+b=[a_1+b_1, a_2+b_2]$

ábrázolva: felrajzoljuk a-t és b-t közös kezdőponttal, kiegészítjük egy 4. csúccsal paralelogrammává, és a kezdőcsúcsot és ezt a 4. csúcsot összekötő vektor lesz az eredmény.

Kivonás:

$a=[a_1, a_2]$ $b=[b_1, b_2]$ --> $a-b=[a_1-b_1, a_2-b_2]$

ábrázolva: felrajzoljuk az a és b vektorokat, és az eredmény az a vektor lesz, ami a b végpontjából az a végpontjába mutat.

Skalárral szorzás:

$a=[a_1, a_2]$, $x>0$ --> $x*a=[x*a_1, x*a_2]$

ábrázolva: felrajzoljuk az a vektort, és annyiszorozásra hosszabbítjuk, amekkora x.

ha $x<0$, akkor a vektor tükröződik az origóra.

Vektor hossza:

$$|a| = ||a||_2 = \sqrt{\sum_{i=0}^{n-1} a_i^2} = \sqrt{\langle a, a \rangle}$$

Skaláris szorzat:

$$\langle a, b \rangle = \sum_{i=0}^{n-1} a_i b_i$$

Mátrix szorzás:

Mátrixok:

Szorzás:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} * \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ae + bg & af + bh \\ ce + dg & cf + dh \end{bmatrix}$$

Az A négyzetes mátrixnak pontosan akkor létezik kétoldali inverze, ha $\det(A) \neq 0$

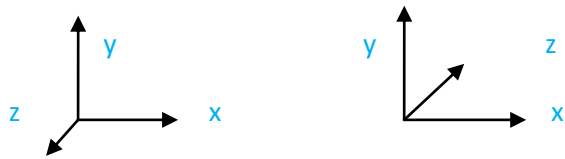
Mátrix szorzása vektorral:

Mátrix oszlopvektorral csak jobbról szorozható, ha a mátrix oszlopainak a száma megegyezik a vektor sorainak a számával. Ennek az eredménye egy oszlopvektor.

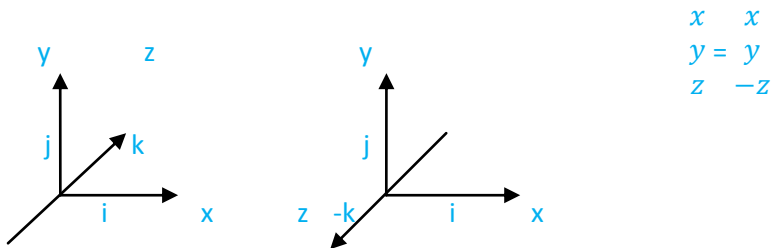
Mátrix sorvektorral csak balról szorozható, ha a sorvektor oszlopainak a száma megegyezik a mátrix sorainak a számával. Ennek az eredménye egy sorvektor lesz.

Mátrixok szorzatának transzponáltja megegyezik a transzponáltak ellenkező sorrendben vett szorzatával. $(A*B)^T = B^T * A^T$

Hogy néz ki egy térbeli jobbsodrású illetve balsodrású koordináta-rendszer?



Legyen adott egy balsodrású koordináta-rendszer i, j, k bázisvektorokkal. Ugyanebből az origóból az $i, j, -k$ vektorokkal képezzünk egy jobbsodrású rendszert (Rajzold le!) Mik lesznek a balsodrású rendszerben $[x, y, z]$ koordinátákkal azonosított pontok koordinátái a jobbsodrásúban?



Descartes féle derékszögű koordináta-rendszer:

Az Euklideszi tér minden pontjához egyértelműen hozzárendel egy rendezett valós számhármast. A Descartes féle koordináta-rendszert a térben egy kezdőponttal (origó) és egy három elemű ortonormált bázissal (3 páronként egymásra merőleges egységvektor) adjuk meg. A bázis vektorai a 3 koordinátatengely irányvektorait jelenti. A koordináta-rendszerben egy P pontot egy 3 komponensű vektorral adhatunk meg, aminek az első komponense azt adja meg, hogy a pont az x tengely irányában hány egységre van az origótól, a második azt, hogy az y tengely irányában hány egységre van, a 3. pedig azt, hogy a z tengely irányában hány egységre van.

Ekkor egy P pont x, y, z koordinátái sorban az origóból P -be mutató vektor i, j, k bázisvektorokra vett előjeles merőleges vetületével egyezik meg.

Előjeles, merőleges vetület:

Az a vektor előjeles merőleges vetülete a $[b]_0$ egységvektorra: $\langle a, [b]_0 \rangle = |a| \cos(\alpha)$ és $[b]_0$ által bezárt szög)

A koordináta-rendszer lehet jobbsodrású, vagy balsodrású.

Két pont távolsága: $d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$

Mi a síkbeli polárkoordináta-rendszer definíciója? Hogyan számíthatóak egy Descartes-féle derékszögű koordináta-rendszerben adott pont polárkoordinátái? Hogyan számíthatók egy polárkoordinátákkal adott pont Descartes-féle derékszögű koordináta-rendszerbeli koordinátái?

Egy síkbeli polárkoordináta-rendszert egy kezdőponttal (referenciapont) és egy a kezdőpontból kiinduló félegyenessel (polártengely) adhatunk meg. Egy P pont helyét a koordináta-rendszerben két adattal tudunk reprezentálni: $r \rightarrow$ vagyis a P távolsága a referenciaponttól, és ϕ , vagyis a referenciapontot és a P pontot összekötő egyenesnek a polártengellyel bezárt szöge.

Polár \rightarrow Descartes: $x = r \cos \phi$, $y = r \sin \phi$, Descartes \rightarrow Polár: $r = \sqrt{x^2 + y^2}$,

$$\varphi = \begin{cases} \arctg\left(\frac{y}{x}\right), & \text{ha } x > 0 \wedge y \geq 0 \\ \arctg\left(\frac{y}{x}\right) + 2\pi, & \text{ha } x > 0 \wedge y < 0 \\ \arctg\left(\frac{y}{x}\right) + \pi, & \text{ha } x < 0 \\ \frac{\pi}{2}, & \text{ha } x = 0 \wedge y > 0 \\ \frac{3\pi}{2}, & \text{ha } x = 0 \wedge y < 0 \end{cases}$$

Ezek akkor teljesülnek, ha az origó és a referenciapont egybe esik, illetve a descartes koordináta-rendszer x tengelye és a polártengely megegyezik.

Ha $x=0$ és $y=0$, akkor $r=0$ mellett tetszőleges szöggel visszajuk az origót.

Általában akkor használjuk, ha az ábrázolni kívánt dolgokhoz jól illeszkedik, pl körmozgás ábrázolása.

Hátránya, hogy egyik polárkoordináta-rendszerből áttáálni egy másikba nagyon költséges.

Mi a következő Descartes-féle derékszögű koordináta-rendszerben adott $[x,y]$ pontok (r,ϕ) polárkoordinátái?

$$[1,0] = (1,0) \quad [0,2] = (2,\frac{\pi}{2}) \quad [-3,0] = (3,\pi) \quad [0,4] = (4,\frac{3\pi}{2}) \quad [1,1] = (\sqrt{2},\frac{\pi}{4}) \quad [-2,2] = (\sqrt{8},\frac{3\pi}{4}) \quad [-3,-3] = (\sqrt{18},\frac{5\pi}{4}) \\ [4,-4] = (\sqrt{32},\frac{7\pi}{4}) \quad [1,\sqrt{3}] = (2,\frac{\pi}{3})$$

Mi a gömbi koordináta-rendszer definíciója? Hogyan számíthatóak egy Descartes-féle derékszögű koordináta-rendszerben adott pont gömbi koordinátái? Hogyan számíthatóak egy gömbi koordinátákkal adott pont Descartes féle derékszögű koordináta-rendszerbeli koordinátái?

A polárkoordináta-rendszer térbeli megfelelője. A koordináta-rendszert egy alapsíkkal és annak polárkoordináta-rendszerével, valamint egy arra merőleges Z tengellyel adhatunk meg. Egy térbeli P pontot 3 adattal tudunk reprezentálni: r : a P origótól vett távolsága, ϕ a P alapsíkra vett vetületének polárkoordinátái, θ : Az origót a P-vel összekötő egyenesnek a Z tengellyel bezárt szöge.

Gömbi \rightarrow Descartes: $x = r \cos \phi \sin \theta$, $y = r \sin \phi \sin \theta$, $z = r \cos \theta$

Descartes \rightarrow Gömbi: $r = \sqrt{x^2 + y^2 + z^2}$, $\phi = \arctan2(x, y)$, $\theta = \arccos \frac{z}{r}$, ha $r \neq 0$, $\theta = 0$, ha $x=y=0$ és $z>0$, $\theta = \pi$, ha $x=y=0$ és $z<0$

Hasznos pl. a földfelszín pontjainak azonosítására

Mik a következő Descartes féle derékszögű koordináta-rendszerben adott $[x,y,z]$ pontok (r,ϕ,θ) gömbi koordinátái? $[1,0,0] = (1,0,\frac{\pi}{2})$ $[0,1,0] = (1,\frac{\pi}{2},\frac{\pi}{2})$ $[0,0,1] = (1,0,0)$ $[1,1,1] = (\sqrt{3},\frac{\pi}{4},\frac{\pi}{4})$

Mik a következő gömbi koordinátákkal adott pontok $[x,y,z]$ koordinátái Descartes féle derékszögű koordináta-rendszerben? $(1,0,0) = [0,0,1]$ $(1,\pi,0) = [0,0,-1]$ $(1,0,\frac{\pi}{2}) = [1,0,0]$ $(1,\frac{\pi}{2},\frac{\pi}{2}) = [0,1,0]$

$$(1,0,\frac{3\pi}{4}) = [\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}] \quad (2,\frac{\pi}{2},\frac{3\pi}{4}) = [0,\sqrt{2}, -\sqrt{2}]$$

Baricentrikus koordináták:

A baricentrikus koordináta-rendszer egy olyan koordináta-rendszer, amelynek kezdőpontja a rendszer tömegközéppontjában van. Ha a síkban P_1, P_2, P_3 pontok kifeszítik a síkot (nem kollineárisak, nem esnek egy egyenesbe) és helyvektoruk v_1, v_2, v_3 , akkor a sík bármely S pontjának s helyvektora előállítható ezek súlyozott közepeként; azaz található hozzá olyan m_1, m_2, m_3 számhármassal, amellyel a S pont s helyvektora: $m_1 v_1 + m_2 v_2 + m_3 v_3 / m_1 + m_2 + m_3$.

Az m_1, m_2, m_3 számokat az S pont (-nak a P_1, P_2, P_3 rendszerre vonatkozó) *baricentrikus koordinátáinak* nevezzük.

Az affin transzformációk nem változtatják meg a Baricentrikus koordinátákat.

Baricentrikus \rightarrow Descartes: $x=ux_1+vx_2+wx_3, y=uy_1+vy_2+wy_3$

Descartes \rightarrow Baricentrikus: $u=(x; p_2; p_3)/(p_1; p_2; p_3), v=(p_1; x; p_3)/(p_1; p_2; p_3), w=(p_1; p_2; x)/(p_1; p_2; p_3)$

Hány pontot kell rögzítenünk a síkban, ha a teljes euklideszi síkot le akarjuk írni baricentrikus koordináták segítségével? Ebben az esetben mit jelent a definíció "nem egy n-1 dimenziós altérbe esnek" kitétele, milyen geometriai megkötést ad a rögzített pontokra?

3 pontot kell rögzíteni.

A három pontunk legyen egymástól lineárisan független a 2 dimenziós altérben

Legyenek adottak az $a=(-1,1), b=(2,4), c=(5,-2)$ pontok a síkban. Mik lesznek a következő a,b,c pontokra vonatkoztatott baricentrikus koordinátákkal meghatározott síkbeli pontok Descartes koordinátái? $(1,0,0)=[-1,1] (0,1,0)=[2,4] (0,0,1)=[5,-2] (-1,1,1)=[8,1] (1,-1,1)=[2,-5] (1,1,-1)=[-4,7]$
 $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})=[2,1] (-\frac{1}{4}, \frac{1}{2}, \frac{3}{4})=[5, \frac{1}{4}]$

Mik lesznek az $a=(0,0), b=(4,0), c=(2,4)$ pontokra vonatkoztatott baricentrikus koordinátái a következő pontoknak? $(0,0)=(1,0,0) (4,0)=(0,1,0) (2,4)=(0,0,1) (2,2)=(\frac{1}{4}, \frac{1}{4}, \frac{1}{2}) (0,4)=(\frac{1}{2}, -\frac{1}{2}, 1) (8,0)=(-1,2,0) (2,-4)=(1,1,-1) (-2,-2)=(\frac{7}{4}, -\frac{1}{4}, -\frac{1}{2})$

Hogyan bővítettük ki az euklideszi síkot és teret? Mik E2 és E3 projektív lezárásainak definíciói?

E2: tekintsük pontnak az egyenesek egyező állását, azaz minden egyenesen legyen még egy pont, az ideális pont. Párhuzamos egyenesek ideális pontja megegyezik. Egy sík ideális pontjai egy egyenesen vannak, ez az ideális egyenes.

E3: párhuzamos síkok ideális egyenese megegyezik. A tér ideális elemei egy síkban vannak, ez a tér ideális síkja

E2 projektív lezárása: 2 pont meghatároz egy egyenest, 2 egyenes meghatároz egy pontot

E3 projektív lezárása: 3 pont meghatároz egy síkot, 3 sík meghatároz egy pontot

Hogyan rendeltünk homogén koordinátákat az euklideszi tér pontjaihoz és vektoraihoz? Egy projektív síkbeli [térbeli] koordináta számhármass [számnégyes] mit ábrázol az euklideszi térben (az értékek függvényében)?

Az euklideszi tér minden pontjához hozzárendelünk egy valós számnégyest (homogén koordináták): $P(x, y, z) \rightarrow [x, y, z, 1] \approx h[x, y, z, 1] = [hx, hy, hz, h] (h \neq 0)$, az összes v vektorhoz pedig: $v = [x, y, z] \rightarrow [x, y, z, 0] \approx h[x, y, z, 0] = [hx, hy, hz, 0] (h \neq 0)$.

$[x_1, x_2, x_3, x_4]=?$

Ha $x_4 \neq 0$, akkor egy pontot ábrázol, $([x_1, x_2, x_3, x_4] \approx [\frac{x_1}{x_4}, \frac{x_2}{x_4}, \frac{x_3}{x_4}, 1] = p(\frac{x_1}{x_4}, \frac{x_2}{x_4}, \frac{x_3}{x_4}))$ ha $x_4=0$, de $x_1^2+x_2^2+x_3^2 \neq 0$, akkor egy ideális pontról van szó, amely az $[x_1, x_2, x_3]$ vektorral egyező állású egyenesek ideális pontja, ha x_1, x_2, x_3, x_4 mind nulla, akkor nincs értelmezve.

Mik lesznek az $[1,2,3]=[1,2,3,1]$, $[4,-2,-5]=[4,-2,-5,1]$ pontok homogén koordinátái? Mik lesznek az $[1,2,3]=[1,2,3,0]$, $[4,-2,-5]=[4,-2,-5,0]$ vektorok homogén koordinátái?

Mik az origó, az x tengely, az y tengely és a z tengely homogén koordinátás alakjai?

$[0;0;0; c]$ az origó, $[c;0;0;0]$ az x tengely ideális pontja, $[0; c;0;0]$ az y tengely ideális pontja, $[0;0; c;0]$ az z tengely ideális pontja

Mik lesznek az euklideszi térben a következő homogén koordinátákkal adott elemek? Mik lesznek a koordinátáik? $[6,15,9,3]=(2,5,3)$ pont $[0,0,0,1]=$ origó $[4,6,8,-2]=(-2,-3,-4)$ pont $[5,4,2,0]=(5,4,2)$ ideális pont/vektor $[1,0,0,0]=$ x tengely ideális pontja/vektora $[0,0,0,0]=$ nincs értelmezve

Milyen megadási módjait ismered az egyenesnek a síkban?

Explicit: $y=f(x)$ --> van-e inverz?

Parametrikus: $p(t)=[x(t), y(t)]$, $t \in R$

Implicit: $ax+by+c=0$

Normálvektoros: $p(p_x, p_y)$ és $n=[n_x, n_y]$ --> $\langle x-p, n \rangle = 0$

Adott egy egyenes a síkban a $p=[2,3]$ pontjával és $n=[1,1]$ normálvektorával. Mi lesz az egyenes egy homogén implicit egyenlete a síkban? Mi lesz a Hesse-féle normalizált alakja az egyenletnek?

Implicit: $1x+1y-(2*1+3*1)=0$ --> $x+y-5=0$ ($a=n_x$, $b=n_y$, $c=-(p_x n_x + p_y n_y)$)

Ha $a^2+b^2=1$, akkor Hesse féle normalizált alakról beszélünk.

Hesse féle normál alak: $\frac{Ax+By-C}{\sqrt{A^2+B^2}} = 0 \rightarrow \frac{x+y-5}{\sqrt{1^2+1^2}}=0 \rightarrow \frac{x+y-5}{\sqrt{2}}=0$

Milyen megadási módjait ismered az egyenesnek a térben?

Parametrikus: Az egyenes megadható egy $p(p_x, p_y, p_z)$ pontjával, és egy az egyenes irányával megegyező irányú $v=[v_x, v_y, v_z]^T \neq 0$ irányvektorral. $x(t)=p_x+tv_x...$

Parametrikus: Az egyenesnek ismerjük 2 pontját, $p-t$ és $q-t$, ha $v=q-p$, akkor visszajutunk az előbbihez: $x(t)=(1-t)p_x+tq_x$

Homogén koordinátás: $e=[e_1, e_2, e_3, e_4]$, $x=[x_1, x_2, x_3, x_4]^T$: $e x = e_1 x_1 + e_2 x_2 + e_3 x_3 + e_4 x_4 = 0$

Milyen megadási módjait ismered a síknak a térben?

Normálvektoros: $p(p_x, p_y, p_z)$, $n=(n_x, n_y, n_z)^T$ $\langle x-p, n \rangle = 0$

Homogén implicit: $ax+by+cz+d=0$ ($a=n_x$, $b=n_y$, $c=n_z$, $d=-n_x p_x - n_y p_y - n_z p_z$)

parametrikus: $p, q, r \rightarrow x(s, t) = p + s(q-p) + t(r-p)$ ($s, t \in R$)

Bázisvektorokkal: $x(s, t) = p + su + tv$

homogén koordináták: $[s_1, s_2, s_3, s_4]$, $[x_1, x_2, x_3, x_4] \rightarrow s x = s_1 x_1 + s_2 x_2 + s_3 x_3 + s_4 x_4 = 0$

Legyen adott egy sík $[3,3,2]$ $[6,4,6]$ $[2,9,5]$ pontjaival. Mi lesz a sík egy parametrikus egyenlete kifeszítő vektorokkal?

$$x(s, t) = p + s(q-p) + t(r-p) = \begin{bmatrix} 3 \\ 3 \\ 2 \end{bmatrix} + s \begin{bmatrix} 6-3 \\ 4-3 \\ 6-2 \end{bmatrix} + t \begin{bmatrix} 2-3 \\ 9-3 \\ 5-2 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 2 \end{bmatrix} + s \begin{bmatrix} 3 \\ 1 \\ 4 \end{bmatrix} + t \begin{bmatrix} -1 \\ 6 \\ 3 \end{bmatrix}$$

Kifeszítő vektorok: $(3,3,2)$, $(3,1,4)$, $(-1,6,3)$

Legyen adva a $3x+4y+5=0$ egyenletű egyenes a síkon. Mi az egyenes normálvektora? Add meg az egyenes egyik pontját!

$Ax+By+C=0$ esetén (A,B) az egyenes normálvektora, (x_0, y_0) pedig egy pontja $\rightarrow -C=Ax_0+By_0$

Normálvektor: $(3,4)$

Pontja: $(1, -2)$

Legyen adva az $x+5y+3z-9=0$ egyenletű sík a térben. Mi a sík normálvektora? Add meg a sík egyik pontját!

$Ax+By+Cz+D=0$ esetén (A,B,C) a sík normálvektora, (x_0, y_0, z_0) pedig egy pontja $\rightarrow -D=Ax_0+By_0+Cz_0$

Normálvektor: $(1,5,3)$

Pontja: $(1, 1, 1)$

Mi a lineáris leképezés definíciója?

ϕ lineáris leképezés, ha $\forall a, b \in \mathbb{R}^3$ és $\lambda \in \mathbb{R}$ esetén

$\phi(a+b) = \phi(a) + \phi(b)$ (additív)

$\phi(\lambda a) = \lambda \phi(a)$ (homogén)

Mik a projektív és affin transzformációk? Milyen algebrai struktúrát alkotnak a konkatenáció műveletével?

Projektív: Az ideális síkkal kibővített Euklideszi tér önmagára való, kölcsönösen egyértelmű, pont, egyenes, sík és illeszkedés tartó leképezései.

Affin: A projektív transzformációk azon alcsoportja, amelyek a kibővített tér "közönséges" Euklideszi részét önmagára képezi le, és az ideális síkot is önmagára képezi le.

Konkatenáció: Algebrai csoportot alkotnak (asszociatív, létezik egység elem, dimenziótartó transzformációknak létezik inverze)

Bizonyítsd be, hogy a baricentrikus koordináták affin invariánsak!

Az affin transzformációk érvényben hagyják a baricentrikus koordinátákat.

Legyenek a tetszőleges x baricentrikus koordinátái x_i -kre vonatkoztatva α_i , ekkor $\varphi(x) = \varphi(\sum_{i=0}^n \alpha_i x_i)$
 $= \sum_{i=0}^n \varphi(\alpha_i x_i) = \sum_{i=0}^n \alpha_i \varphi(x_i)$

Ismertesd az eltolást!

Minden pontot egy adott d vektorral eltolunk.: $x' = x + d$ ($T(dx, dy, dz)$)

Inverz: $T(a, b, c) \rightarrow T^{-1}(a, b, c) = T(-a, -b, -c)$

Mátrix:
$$\begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ismertesd a forgást!

Forgatás Z tengely körül az XY síkban θ szöggel:

$$x' = x \cos \theta - y \sin \theta, y' = x \sin \theta + y \cos \theta$$

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \rightarrow R_z(\theta) = \begin{bmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R_y(\theta) = \begin{bmatrix} c & 0 & -s & 0 \\ 0 & 1 & 0 & 0 \\ s & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & -s & 0 \\ 0 & s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ahol a $c = \cos \theta$, $s = \sin \theta$

Inverz: $R_T^{-1} \theta = R_T(-\theta)$, ahol t az egyik tengely

Ismertesd a méretezést! Milyen speciális esetei vannak, hogyan hatnak ezek a sodrásírányra, van-e minden esetben inverz?

Az x,y,z tengelyek mentén "széthúzzuk" vagy "összenyomjuk" az alakzatot, azaz más léptéket választunk tengelyenként egymástól függetlenül is akár.

$$\text{Mátrix: } S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ha s_x, s_y, s_z valamelyike negatív:

Ha 1 negatív, akkor tükrözés az irányra merőleges síkra

Ha 2 negatív, akkor tükrözés egy tengelyre

Ha 3 negatív, akkor középpontos tükrözés az origóra.

$$\text{Inverz: } S(a, b, c) \rightarrow S^{-1}(a, b, c) = S\left(\frac{1}{a}, \frac{1}{b}, \frac{1}{c}\right)$$

Ha páratlan számú negatív együtttható van, akkor változik a sodrásírány

Ha s_x, s_y, s_z valamelyike 0:

Ha 1 nulla, akkor az irányra merőleges síkra vetítünk

Ha 2 nulla, akkor egy tengelyre vetítünk

Ha 3 nulla, akkor az origóba vetítünk

Mi a merőleges vetítés transzformációs mátrixának inverze?

Nincs inverze, a determináns 0.

Az $R_x(\alpha)$, $R_y(\beta)$ és $R_z(\gamma)$, X, Y és Z tengelyek körüli elforgatások, és a $T(x, y, z)$ transzlációs mátrixok felhasználásával hogyan lehet felírni egy [5,0,4] középpontú XZ síkban fekvő 3 sugarú kör mentén történő forgatás transzformációs mátrixát?

$$T(3,0,0) * R_z(\gamma) * T(5,0,4) = M$$

Vezesd le az i,j,k ortonormált bázisból az u,v,w ortonormált bázisba való áttérés transzformációs mátrixát!

$$\begin{matrix} x & x' \\ y \rightarrow & y' : X \\ z & z' \end{matrix} \rightarrow \begin{bmatrix} ux & vx & wx \\ uy & vy & wy \\ uz & vz & wz \end{bmatrix} * x' = Bx' \rightarrow x' = B^{-1}x$$

$$\begin{bmatrix} ux & vx & wx \\ uy & vy & wy \\ uz & vz & wz \end{bmatrix}^{-1} = \begin{bmatrix} ux & vx & wx \\ uy & vy & wy \\ uz & vz & wz \end{bmatrix} \rightarrow \text{Transzformációs mátrix: } \begin{bmatrix} ux & uy & uz & 0 \\ vx & vy & vz & 0 \\ wx & wy & wz & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Hogyan kell felületi normálisokat transzformálni?

$n' = (M^{-1})^T n$, a transzformációs mátrix inverz transzponáltjával kell szorozni a normálvektorokat.

$$\langle x-p, n \rangle = 0$$

$$\langle M^{-1}M(x-p), n \rangle = 0$$

$$\langle M(x-p), (A^{-1})^T n \rangle = 0$$

Legyen adott egy felületi pont és a hozzá tartozó modell (világ) transzformációnk: $M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix} A$

felületi normális az adott pontban a modell saját koordinátarendszerében $n = [0, 1, 0]$. Mi lesz a transzformált felületi normális a világ-koordinátarendszerben?

$$n' = (M^{-1})^T n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{4} \end{bmatrix}^T * n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{4} \end{bmatrix} * \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{2} \\ 0 \end{bmatrix}$$

Mik lesznek a $[3, 4, 4]$ pont koordinátái a $[2, 2, 2]$ origójú, $i = [1, 0, 0]$, $j = [0, -1, 0]$, $k = [0, 0, 1]$ tengelyű bázisban?

Tegyük fel, hogy oszlopvektorokat használunk és a mátrixokhoz jobbról szorozzuk a vektorokat. Ekkor a következő 4×4 -es mátrixoknak mi a legszűkebb transzformációs osztálya az az lineáris, affin vagy projektív transzformációkat azonosítanak-e? Mik lesznek ezek a legszűkebb transzformációs osztályok, ha a vektoraink sorvektorok, és a mátrixokhoz a vektorokat balról szorozzuk?

$$A = \begin{bmatrix} 4 & 3 & 2 & 0 \\ 0 & 3 & 4 & 0 \\ 1 & 3 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} B = \begin{bmatrix} 4 & 3 & 2 & 0 \\ 0 & 3 & 4 & 0 \\ 1 & 3 & 2 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} D = \begin{bmatrix} 1 & 0 & 1 & 3 \\ 0 & 1 & 2 & 6 \\ 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} E = \begin{bmatrix} 1 & 2 & 3 & 5 \\ 1 & 1 & 6 & 0 \\ 0 & 0 & 3 & 2 \\ 2 & 5 & 0 & 1 \end{bmatrix}$$

Hogyan halad a fény a szemben? Milyen lépései függnek az útnak a beérkező fény hullámhosszától?

A fény a szaruhártyán megtörve jut a szembe. Ez lényegében egy fénygyűjtő lencse. A szivárványhártya csökkenti a szembe jutó fény mennyiségét, a pupilla pedig fényrekeszként funkcionál. A szemlencse a második gyűjtőlencse, a belépő fénysugarakat a retinára fókuszálja. A különböző hullámhosszú fény másképp törik. Ahhoz, hogy a szemlencse ezeket is a retinára tudja fókuszálni a sugárizomnak módosítania kell a szemlencse alakját. Ekkor ugyanaz történik, mint amikor közelre vagy távolra nézünk.

A retinában milyen fényérzékelést szolgáló idegsejtek találhatók?

A retinában kétféle fény érzékelésre szolgáló receptor található. A pálcikák az alacsonyabb intenzitású fényre érzékenyek, a sötét-világos megkülönböztetésére alkalmasak alacsonyabb

felbontásúak. A csapok erősebb fényingert igénylő idegsejtek, a színlátást és az éleslátást szolgálják, tizedannyira érzékenyek a fényre mint a pálcikák.

Az elektromágneses energia egy bizonyos sávjára érzékenyek csak a fenti sejtek. Amikor fény ér egy fotoreceptort, egy kémiai reakció indul el, aminek eredményeképp neurális jelet küldenek az agy felé, úgynevezett fotopigmentet. Az egyes fotoreceptorok más-más mértékben reagálnak a különböző hullámhosszú fényekre: A pálcikáknál hullámhossztól függően az ugyanolyan erős fényre adott reakció nagysága haranggörbe jellegű görbével írható le az ember éjszakai látásának nagyjából megfelelő módon. A csapokból három féle van (s, m, l), mindegyik különböző hullámhosszú fényre ad maximális reakciót, azoktól fokozatosan eltérőkre egyre kisebbet.

Csapok: S: 420 nm (kék), M: 530 nm (zöld), L: 560 nm (vörös) Ha egy csap az adott hullámhosszra x százalékban érzékeny, akkor 100-ból x-szer fogja abszorbeálni az olyan hullámhosszú fénykomponenst, és küld jelet az agy felé. A látóideg csatlakozása a szemgolyóhoz a vakfolt, itt nincsenek fotoreceptorok. A látógödör, ami a vakfolttól oldalra található az éleslátás helye, a közepe a foveola, ahol kizárólag csapok találhatóak. Innen kifelé haladva a csapok ritkulnak, pálcikák váltják fel őket.

Mi a CFF? Miért érzékeljük folyamatos képnek az ennél sűrűbben felvillanó képsorozatot?

Az egyetlen fotonra adott neurális jel néhány ms-ig tart, minden egyes beérkező foton hatása hozzáadódik az előzőhöz. A receptor által leadott jel lényegében egy időbeli átlag, egy alul-áteresztő szűrő, aminek vágási frekvenciája függ a megvilágítástól.

Egy lassan villogó fényt külön-külön felvillanásként érzékelünk, azonban a felvillanások között eltelt idő egyre kisebb, akkor a fotoreceptorok által leadott jelek összetorlódnak, (eléri a Critical flicker frequency-t) és folyamatos fénypontként érzékeljük a látottakat. A flicker rate több tényezőtől is függ, pl. a háttérmegvilágítástól, a megjelenített kép nagyságától, stb. Ideális körülmények között nagyjából 60Hz

Mi a tristimulus érték?

Bármely színérzet kódolható egy számhármassal, ezeket hívjuk tristimulus értékeknek.

Jellemezd az RGB színteret!

Egy additív színmodell. A 3 kiválasztott hullámhossz: $\lambda_{red}=700\text{ nm}$, $\lambda_{green}=561\text{ nm}$, $\lambda_{blue}=436\text{ nm}$ (λ egy monokromatikus fénynyaláb) Ekkor a hozzá tartozó RGB értékek megadására használjuk az $r(\lambda)$, $g(\lambda)$, $b(\lambda)$ színillesztő függvényeket.

Nem adható meg vele minden szín, amit az ember érzékelni tud!

Jellemezd a CYM(K) színteret!

Itt másik 3 hullámhosszt használunk, a ciánkék (C), a bíbor (M) és a sárga (S) színeket. Ez a három tengely ugyanúgy kifeszíti a színteret. Ezt használják a színes nyomtatáshoz (a feketével (K) kiegészítve, hogy annak előállításához ne kelljen mindhárom színt felhasználni.)

Jellemezd a HSL/HSV színmodelleket!

Egy olyan megoldás, ahol a színárnyalatokon (H) kívül a telítettséget (S) és a fényességet (L) vagy Világosságot (V) is meg tudjuk adni ezzel további színeket előállítva.

A fény-anyag kölcsönhatások tekintetében milyen típusú felületeket vettünk?

Emittáló felületek: saját fényt bocsátanak ki

Diffúz felületek: A diffúz vagy matt felületeket minden irányból nézve ugyanolyan színűnek látjuk. Ilyen pl. a frissen meszelt fal, vagy a homok. A diffúz felület a beérkező fénysugár energiáját minden irányban azonos intenzitással veri vissza.

Tükröző felületek: Az ideális fénytörés irányában verik vissza a beérkező fénysugár nagy részét.

Átlátszó felületek: Ezeken a felületeken áthalad a fény, a beérkező fénysugár energiájának nagy részét áteresztik.

Áttetsző felületek: Ezek a beérkező fény nagy részét magukba engedik, viszont ennek csak kis része lép ki az anyagból.

Anizotróp felületek: A felületet a tengelye körül forgatva a beeső és visszaverődő szögeket tartva is változik a színe.

Fényforrás modellek:

Pont: egyetlen pontból sugároz, az intenzitás a távolság négyzetének arányában csökken.

Irány: Egyetlen irányba sugároz, a sugarak párhuzamosak, az intenzitás független a pozíciótól.

Spot: Iránnyal és hatóterülettel rendelkeznek, pl.: zseblámpa.

Mi a BRDF? Miket használunk fel a megadásakor? (Rajz is!)

Kétirányú visszaverődéses eloszlási függvény:

L^{in} Egy adott irányból a felület egy pontjára beérkező fény intenzitása

L Az előbbi pontról visszavert fény intenzitása

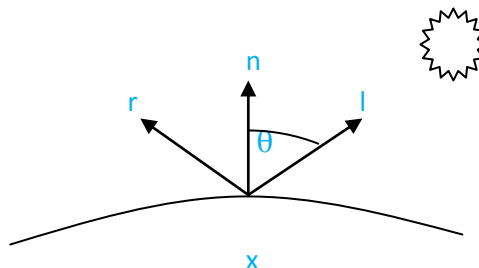
\mathbf{l} A fényforrás felé mutató egységvektor

\mathbf{v} a nézőpont felé mutató egységvektor

\mathbf{n} a felületi normális az adott pontban

θ' az \mathbf{n} és \mathbf{l} által bezárt szög.

$$f_r(\mathbf{l}, \mathbf{v}) = \frac{L}{L^{\text{in}} \cos \theta'}$$



Mi az ideális visszaverődés? Hogyan lehet kiszámítani az ideális visszaverődés irányát? Ábrával is szemléltesd a felhasznált mennyiségeket!

A beesési irány, a felületi normális és a kilépési irány egy síkban vannak, valamint a beesési szög megegyezik a visszaverődési szöggel.

Az ideális tükör csak az \mathbf{r} tükörirányba ver vissza. $f_r(\mathbf{x}, \mathbf{v}, \mathbf{l}) = k_r \frac{\delta(\mathbf{r} - \mathbf{v})}{\cos \theta'}$

A δ a Dirac-delta fv, ami egy általánosított fv, amely minden nem nulla paraméterre nullát ad, de a valós számok feletti integrálja 1. A k_r visszaverődési együttható a Fresnel-együttható, ez az anyag törésmutatójából és az elektromos vezetési képességéből származik. A Fresnel-együttható a visszavert és beeső energia hányadosát fejezi ki.

Vezesd le az ideális törési irány kiszámítását adott felületi normális, beérkező sugárirány és relatív törésmutató mellett!

$$\eta = \frac{\sin \alpha}{\sin \beta}$$

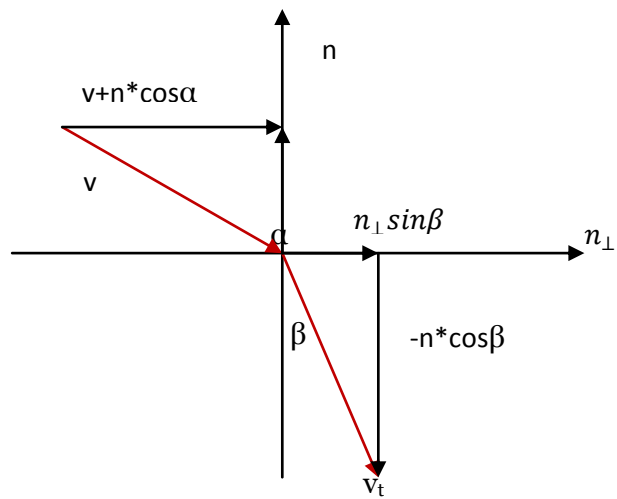
$$v_t = n_{\perp} \sin \beta - n \cos \beta$$

$$n_{\perp} = \frac{v + n \cos \alpha}{\sin \alpha}$$

$$v_t = \frac{v}{\eta} + n * \frac{\cos \alpha}{\eta} - \cos \beta$$

$$\cos \beta = \sqrt{1 - \sin^2 \beta} = \sqrt{1 - \frac{\sin^2 \alpha}{\eta^2}}$$

$$v_t = \frac{v}{\eta} + n * \left(\frac{\cos \alpha}{\eta} - \sqrt{1 - \frac{\cos^2 \alpha}{\eta^2}} \right)$$



Ismertesd a spekuláris visszaverődést és a Phong modelljét! Írd fel a BRDF-et, ismertesd a képletben szereplő együtthatókat!

A tükörirányban intenzíven visszaverő, de attól távolodva gyorsan elhaló csillanás adható meg vele.

Legyen ϕ az r tükörirány és a v nézeti irány által bezárt szög. Ekkor $\cos \phi = r \cdot v$. Olyan függvényt

keresünk, ami $\phi=0$ -ra nagy, de gyorsan elhal: $f_r(x, v, l) = k_s \frac{\cos^n \phi}{\cos \theta^l}$.

Nem szimmetrikus!

Csak ezt nézve: $L_{ref} = L_i k_s (\cos^+ \phi)^n$

Ismertesd a spekuláris visszaverődést és a Phong-Blinn modelljét! Írd fel a BRDF-et, ismertesd a képletben szereplő együtthatókat!

Legyen h a nézeti irány, és a megvilágító pont fele mutató vektor felezővektora. $h = \frac{v+l}{||v+l||}$ Legyen δ a

h és az n normálvektor által bezárt szög. Ekkor $\cos \delta = h \cdot n$

$f_r(x, v, l) = k_s \frac{\cos^n \delta}{\cos \theta^l}$. Csak ezt nézve: $L_{ref} = L_i k_s (\cos^+ \delta)^n$

Hogy néz ki az y tengelyű, $(0, p)$ fókuszpontú parabola implicit, explicit és parametrikus egyenlete?

Implicit: $x^2 = 4py$

Explicit: $y = \frac{x^2}{4p}$

Parametrikus: $p(t) = [t, \frac{x^2}{4p}]^T, t \in \mathbb{R}$

Hogy néz ki a $(4, 3)$ középpontú, $(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ és $(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$ tengelyű $a=4$, $b=3$ fél nagy- és kistengelyű ellipszis parametrikus egyenlete?

$$p(t) = a * \cos t * k + b * \sin t * l + c = 4 * \cos t * \frac{1}{\sqrt{2}} + 3 * \sin t * \frac{-1}{\sqrt{2}} + \frac{4}{3}$$

Legyen adott egy felület az $f(x, y, z) = x^2 + y^2 - z$ egyenlettel. Az alábbi pontok közül melyek vannak a felületen? $(0, 0, 0)$, $(4, 1, 2)$, $(-2, 2, 0)$, $(-2, 2, 8)$, $(0, 1, 1)$

$$f(0, 0, 0) = 0 + 0 - 0 = 0$$

$$f(4, 1, 2) = 16 + 1 - 2 = 15 \neq 0$$

$$f(-2, 2, 0) = 4 + 4 - 0 = 8 \neq 0$$

$$f(-2,2,8)=4+4-8=0$$

$$f(0,1,1)=0+0-1=0$$

Jellemezd a sugárkövetést! Mi a célja, hogyan működik, a fény útját hogyan követi, milyen komponensekre bontja, milyen sugártípusok vannak, stb.

Célja szimulálni a színtérben a fény terjedését, valamint a fénysugarak és a felületek ütközését. A képernyő pixeleire egymástól függetlenül oldja meg az árnyalási és takarási feladatokat. Minden pixelre indítunk egy sugarat a színtérbe. Minden objektumra a színtérben nézzük meg, hogy metszi-e a sugár az objektumot. A legközelebbi metszett objektum színével színezzük ki a pixelt. Mivel csak azokkal a fotonokkal érdemes foglalkozni, amik valóban a szembe jutnak, így a fény követését a szempozíciótól visszafelé végzi, így jut el a fényforráshoz. A fény útját két féle komponensre bontjuk: koherens és inkoherens komponensre. Koherens eset: Az optikának megfelelő ideális visszaverődés és tükrözés, ebben az esetben tovább követjük a fény útját. Inkoherens eset: Minden egyéb, ekkor csak az absztrakt fényforrás direkt megvilágítását vesszük figyelembe.

Ismertesd a sugárkövetés során megoldott egyszerűsített illuminációs egyenlet inkoherens visszaverődéseket összefoglaló tagját! Ismertesd az egyenlet tagjait, mit jelentenek, mik a paramétereik?

$$L(x,w)=L_e(x,w)+k_a * L_a + \sum_{I \in Lights} f_r(x, w_I, w) L_i(x, w_I) (-w_I * n) + k_r * L(x, w_r) + k_t * L(x, w_t)$$

$$\sum_{I \in Lights} f_r(x, w_I, w) L_i(x, w_I) (-w_I * n)$$

Az inkoherens visszaverődéseket foglalja össze a szummás tag. Itt csak a fényforrások direkt hatását vesszük figyelembe, és csak akkor, ha az az x felületi pontból látszik.

$w_I \rightarrow$ a fényforrásból a felületi pontba mutató egységvektor

$f_r(x, w_I, w) \rightarrow$ most csak a diffúz és spekuláris visszaverődést jellemző BRDF

$-w_I * n \rightarrow$ a felületi normális és a fényforrás fele mutató vektor által bezárt szög koszinusza

Írd fel a sugárkövetés során megoldott egyszerűsített illuminációs egyenletet! Mely tagokban találhatóak az emittált, ambiens fényforrásokból érkező, tükröző és törő fénymennyiségek?

$$L(x,w)=L_e(x,w)+k_a * L_a + \sum_{I \in Lights} f_r(x, w_I, w) L_i(x, w_I) (-w_I * n) + k_r * L(x, w_r) + k_t * L(x, w_t)$$

Emisszió: $L_e(x,w) \rightarrow$ Az x felületi pontból a w nézeti irányból érkező radiancia a felület saját sugárzása - emissziója miatt.

Ambiens: $k_a * L_a \rightarrow$ k_a a felület, L_a a környezet ambiens együtthatója. Az egyenlet ambiens tagja közelíti azt a fénymennyiséget, ami általánosan jelen van, minden felületet ér, azok helyzetétől és az absztrakt fényforrásoktól függetlenül.

Tükröződés: $k_r * L(x, w_r) \rightarrow$ A tükörirányból érkező fényt k_r arányban vesszük figyelembe. w_r az ideális tüköriránynak megfelelő beeső vektor

Fénytörés: $k_t * L(x, w_t) \rightarrow$ A törési irányból érkező fényt k_t arányban vesszük figyelembe. w_t a törésiránynak megfelelő beeső vektor.

Vezesd le a sugárkövetés során az (i,j) pixelből milyen sugár indul (kezdőpont, irányvektor)!

A sugár kezdőpontja a pixel középpontja, iránya pedig $\text{norm}(\text{center}(i,j) - \text{eye})$, vagyis a pixel középpontja - a szempozíció

Miért és hol van szükség a sugárkövetés során metszésvizsgálatokra?

A pixelek színének megállapításánál, fénytörés és árnyékok meghatározásánál

Ismertesd általánosan a parametrikus alakban megadott felület és a sugár metszését!

Legyen $r(u,v) = [r_x(u,v), r_y(u,v), r_z(u,v)]^T$ parametrikus felület. Kell találni egy olyan t sugárparamétert, amihez létezik $(u,v) : r(t) = r(u,v)$. Ez három ismeretlenes (t,u,v) , három egyenletes egyenletrendszer

$t > 0 \rightarrow$ metszi és a képernyő előtt van

$t = 0 \rightarrow$ a sugár a felületről indul

$t < 0 \rightarrow$ metszi és a képernyő mögött van

Ismertesd általánosan az implicit alakban megadott felület és a sugár metszését!

Legyen adva egy $f(x) = 0$ implicit egyenlet, ami meghatározza a metszeni kívánt felületünket ($x \in \mathbb{R}^3$). A sugarak egyenlete minden t eleme $[0, \infty)$ -re meghatároz egy pontot a térben \rightarrow tegyük be ezt a képletet az implicit egyenletbe, tehát a következő egyenletet kell megoldanunk t -re: $f(r(t)) = 0$

Ha $t > 0 \rightarrow$ van metszés, a képernyő előtt

Ha $t = 0$, a sugár a felületről indul

Ha $t < 0$, van metszés, a képernyő mögött

Vezesd le a q_0 pontjával és n normálisával adott sík és a p_0 pontjával és v irányvektorával adott sugár metszéspontját!

Egyenes: $p(t) = p_0 + tv$

Sík: $\langle n, q - q_0 \rangle = 0$

$\langle n, p_0 + tv - q_0 \rangle = 0$

$\langle n, p_0 \rangle + t \langle n, v \rangle - \langle n, q_0 \rangle = 0 \rightarrow t = \frac{\langle n, q_0 \rangle - \langle n, p_0 \rangle}{\langle n, v \rangle} \quad (\langle n, v \rangle \neq 0)$

Vezesd le a c középpontú, r sugarú gömb, és a (p_0, v) sugár metszéspontjának kiszámítását!

Sugár: $p(t) = p_0 + tv$

Gömb: $\langle p - c, p - c \rangle - r^2 = 0$

$\langle p_0 + tv - c, p_0 + tv - c \rangle - r^2 = 0 \rightarrow t^2 \langle v, v \rangle + 2t \langle v, p_0 - c \rangle + \langle p_0 - c, p_0 - c \rangle - r^2 = 0$

$D = (2 \langle v, p_0 - c \rangle)^2 - 4 \langle v, v \rangle (\langle p_0 - c, p_0 - c \rangle - r^2)$

Ha $D > 0 \rightarrow$ 2 megoldás van, a sugár dőfi a gömböt

Ha $D = 0 \rightarrow$ 1 megoldás van, a sugár érinti a gömböt

Ha $D < 0 \rightarrow$ nincs metszéspont

A felületünket metszse a $v = [0, -1, 0]$ irányvektorú sugarunk egy olyan p pontban, ahol a felületi normális $[1, 1, 0]$ iránnyal egyező irányú. Mi lesz az ideális visszaverődés irányába indított sugarunk V_r irányvektora?

$$\langle -v, n \rangle = |-v| |n| \cos \alpha \rightarrow \alpha = \arccos\left(\frac{1}{|v| |n|}\right)^{\frac{1}{\sqrt{2}}} = 45^\circ = \frac{\pi}{4}$$

$$|v| = 1, |n| = \sqrt{2}$$

$$\langle v_r, n \rangle = |v_r| * |n| * \cos\left(\frac{\pi}{4}\right) \rightarrow |v_r| = \frac{1}{|n| \cos \frac{\pi}{4}} = 1$$

$$v_r = (1, 0, 0)$$

Adott egy $c=[4,0,0]$ középpontú $r=2$ sugarú gömb. Mi lesz a $p_0=[0,1,0]$ -ból induló, $v=[1,0,0]$ irányvektorú sugár metszéspontja a gömbbel? Milyen távol van egymástól a sugár kiindulópontja és a metszéspont?

Adott egy $c=[4,0,0]$ középpontú $r=2$ sugarú gömb. Mi lesz a $p_0=[4,4,0]$ -ból induló $v=[0,1,0]$ irányvektorú sugár metszéspontja a gömbbel? Milyen távol van egymástól a sugár kiindulópontja és a metszéspont?

Adott egy sugár és egy objektum, amit M mátrixszal transzformáltunk. Hogyan számítható egy transzformált sugár és az eredeti, transzformálatlan objektum segítségével a transzformált objektum metszéspontja?

Mi az alias? A sugárkövetésnél miben jelentkezik?

Milyen módszereket tanultunk a metszésvizsgálat gyorsítására (befoglaló testek, térfelosztások, stb.)? Ismertesd őket röviden! Hogyan használjuk őket a metszéskeresés gyorsítására?

A metszésvizsgálatot úgy tudjuk gyorsítani, ha nem vizsgáljuk olyan objektumokra, amiket biztosan nem metsz a sugár, illetve amik biztosan távolabbi metszéspontot adnak, mint a már megtaláltak.

Befoglaló keret: Minden objektumot körbeveszünk egy olyan kerettel, amivel gyorsan lehet metszést számolni. (pl.: gömb, doboz). Ha egy sugár metszi az objektumot, az biztosan metszi a dobozt is, fordítva pedig legyen minél nagyobb a valószínűsége.

Hierarchikus befoglaló keretek: A kisebb kereteket nagyobb keretekbe foglaljuk, ezzel fa struktúrát kapunk. Egy részfat csak akkor kell megvizsgálni, ha a gyökérrel van metszés.

Szabályos felosztás: Egy szabályos 3D ráccsal lefedjük az egész színteret, és minden cellához feljegyezzük a beletartozó objektumokat. Csak azokra végezzük a metszésszámítást, amik benne vannak az adott cellában. Előnye, hogy a vizsgálandó cellák gyorsan számolhatók szakaszrajzoló algoritmussal, hátránya, hogy feleslegesen sok a cella, sok üres közülük.

Oktális fa: A fa gyökere a teljes színteret magába foglaló, tengelyekkel párhuzamos élű befoglaló doboz. Ezt 8 egyenlő részre osztjuk. minden új dobozra: ha elég sok objektum van benne, akkor tovább osztjuk újabb 8 egyenlő részre, különben pedig megállunk. Előny: Az üres részeket nem osztjuk tovább feleslegesen. Hátrány: bonyolultabb eljárás. Probléma: mindig középen, és minden sík mentén vág az objektumoktól függetlenül. Keresési idő \approx a fa magassága \rightarrow a fa kiegyensúlyozatlan

KD fa: minden lépésben egyetlen síkkal vágunk, ami egy tengelyre merőleges. Sorrend: x, y, z, x, y, z, \dots
Felezősíki elhelyezése: térbeli középvonal módszer, test középvonal módszer, költség modell alapú módszer

Mi az inkrementális képszintézis, milyen kapcsolódó fogalmakat vettünk? Hasonlítsd össze a sugárkövetést és az inkrementális képszintézist!

Az inkrementális képszintézissel objektumokat jeleníthetünk meg a monitoron, mégpedig a sugárkövetéshez képest sokkal kevesebb számítást igénylő módon. Ezt úgy tuja megvalósítani, hogy pixelek helyett primitívenként számol, ezt hívjuk koherenciának. Ez objektum tér pontosságát eredményez a sugárkövetésnél megjelenő pixelpontosság helyett. Ahol csak lehet kihasználja az inkrementális elv nyújtotta lehetőségeket, vagyis az árnyalási és takarási feladatoknál kihasználjuk a nagyobb egységenként szerzett információkat, így ezeket nem kell minden pixelre újra számolni. Minden alapfeladatot a hozzá legjobban illeszkedő koordináta rendszerben végez el, és azok között homogén lineáris geometriai transzformációkkal vált. Ezt akkor teheti meg könnyedén, ha a világban csak primitívekkel kell számolnia, így az inkrementális képszintézis során ami nem primitív, azt azzal kell közelíteni. Ezt hívjuk tesszellációnak. A felesleges számolások elkerülése érdekében a vágás során eltávolítja a képnek azon részeit, amik nem jelennek meg a monitoron. Ezek mellett a sugárkövetésnél jelenlévő tükröződés, fénytörés és árnyékolás megvalósításához az inkrementális képszintézis használatakor külön algoritmusra van szükségünk, és a sugárkövetésnél szintén triviális takarási feladatra is külön megoldást kell találni. Mivel az előbb felsoroltak miatt az inkrementális képszintézis sokkal gyorsabb mint a sugárkövetés, így valós idejű alkalmazásokban sokkal inkább ezt használják, többek között az OpenGL és a DirectX is ezeket az elveket alkalmazza.

Mi a grafikus szerelőszalag? Mi a bemenete és a kimenete? Mik a főbb lépései műveletek (transzformációk szerint)?

A színterünkéről készített kép elkészítésének műveletsorozatát nevezzük grafikus szerelőszalagnak. Ez a modellezési koordinátarendszertől a képernyőig tartó transzformáció sorozat, amiben több koordináta-rendszer váltás is történik, minden feladatot a hozzá legjobban illeszkedő koordináta-rendszerben próbálunk elvégezni.

A szerelőszalag bemeneti adatai: az ábrázolandó tárgyak geometriai modellje, a virtuális kamera adatai (nézőpont, és látógúla), a képkeret megadása (az a pixeltömb, amire a színterünk síkvetületét leképezzük), a színtérben található fényforrásokhoz és anyagokhoz tartozó megvilágítási adatok.

A szerelőszalag lépései a főbb műveletek szerint: Modellezési transzformáció, Nézeti transzformáció, Perspektív transzformáció, Vágás, Homogén osztás, Raszterizáció, Megjelenítés.

A Grafikus szerelőszalag eredménye egy kép, vagyis egy kétdimenziós pixeltömb, aminek minden elemében egy színérték található, ami megjeleníthető a monitoron.

Minden egyes primitív végigmegy az összes lépésen, az egyes lépések az eredményüket a következőnek továbbítják.

Milyen koordinátarendszereket vettünk a grafikus szerelőszalag kapcsán? Jellemezd őket röviden!

Modell koordináta-rendszer: Minden objektum a saját koordináta-rendszerében adott

Világ koordináta-rendszer: Az objektumok egymáshoz viszonyított helyzete itt adott, ahogy a kamera/szem pozíció, és az absztrakt fényforrások pozíciója is.

Kamera koordináta-rendszer: A koordináták a kamera pozíciójához és orientációjához képest relatívan adottak.

Normalizált eszköz koordináta-rendszer: A hardverre jellemző $[-1, 1] \times [-1, 1] \times [0, 1]$ kiterjedésű koordináta-rendszer.

Képernyő koordináta rendszer:

A megjelenítendő képnek megfelelő Koordináta-rendszer (balkezes, bal felső sarok az origó)

Milyen transzformációkat vettünk a grafikus szerelőszalag kapcsán? Jellemezd őket!

Modellezési transzformáció: A modell koordináta-rendszerben megadott modelleket elhelyezi a világ koordináta-rendszerben. A transzformáció tipikusan minden modellre különböző, jellemzően affin transzformáció. A gyakorlatban ezt a modell mátrix ábrázolja a kódjainkban.

Nézeti transzformáció: A világ koordináta-rendszert a kamera koordináta-rendszerbe viszi át. A transzformáció a kamera tulajdonságaiból adódik. A gyakorlatban ezt a view mátrix ábrázolja.

Kamera transzformáció: Meg kell adni a szempozíciót (eye), a pontot, ahová a kamera néz (center), és a felfelé irányt megadó vektort a világban (up). Ezekből kapjuk a nézeti koordináta-rendszer tengelyeit: $w = \text{eye} - \text{center} / |\text{eye} - \text{center}|$, $u = \text{up} \times w / |\text{up} \times w|$, $v = w \times u$. Áttérés a -eye origójú u, v, w koordináta-rendszerbe: $T_{\text{View}} = [U_x U_y U_z 0, V_x V_y V_z 0, W_x W_y W_z 0, 0001] * [100 - \text{eyex}, 010 - \text{eyey}, 001 - \text{eyez}, 0001]$

Perspektív transzformáció: A perspektív transzformáció célja, hogy a modellezési és a nézeti transzformációval elhelyezett virtuális világot az ablak síkjára vetítse. Ehhez meg kell adnunk a függőleges és vízszintes nyílásszöget, vagy az alap oldalainak az arányát és a függőleges nyílásszöget, ezen kívül a közeli és a távoli vágósík távolságát.

Normalizált látógúla: A nézeti transzformáció után a képszíntézisben résztvevő pontok tartománya egy szimmetrikus csónakgúla. A további műveletekhez ezt a gúlát normalizáljuk olyan módon, hogy a csúcsában a nyílásszög 90° legyen. $T_{\text{norm}} = [(1/\tan(\text{fov}/2) * \text{aspect})000, 0(1/\tan(\text{fov}/2))00, 0010, 0001]$. Ez után már csak a közeli és távoli vágósík z koordinátáit kell a normalizálásnak megfelelően átképezni (-1, 1, vagy 0, 1-re). Ha a közeli vágósíkon elhelyezkedő pontot a $z=0$, a távoli vágósíkon elhelyezkedőt pedig a $z=1$ síkra akarjuk képezni, akkor enek a mátrixa: $[1000, 0100, 00(\text{far}/\text{far} - \text{near})(-\text{near} * \text{far}/\text{far} - \text{near}), 00-10]$

Mi az origó középpontú, attól a Z tengely mentén d egységre lévő XY síkkal párhuzamos vetítési síkra vetítő középpontos vetítés mátrixa? Vezesd le! (ábra+háromszögek, stb.)

Középpontos vetítés: $x' = x/z * d$, $y' = y/z * d$, $z' = z/z * d = d$. Az origó mint középpont, és egy attól a Z tengely mentén d távolságra található, XY síkkal párhuzamos vetítő síkra való vetítés mátrixa: $[1000, 0100, 0010, 001/d0]$

Mi a vágás, mi a feladata, milyen koordinátarendszerben kell végezni, miért?

Vágás: A vágás célja az összes olyan objektumrészlet eltávolítása, amely nem vetülhet az ablakra, vagy amely nem az első és hátsó vágósíkok között van. Az átfordulási probléma miatt a vágást a homogén osztás előtt kell végrehajtani. Vágás homogén koordinátákban: $(h > 0)$, $-1 < x < 1$, $-1 < y < 1$, $0 < z < 1 \rightarrow -h < xh < h$, $-h < yh < h$, $0 < zh < h$

Mi a raszterizáció, mi a tesszeláció?

Raszterizáció: Eddig minden primitív amiről beszéltünk folytonos objektum volt, azonban nekünk egy diszkrét térben a képernyő képpontjain kell dolgoznunk. A primitívek folytonos teréből át kell térni erre a diszkrét térre, ezt hívják raszterizációnak. Olyan geometriai primitíveket kell választanunk, amelyeket gyorsan tudunk raszterizálni, erre a legjobb a háromszög. Minden egyéb felületet ilyen primitívekkel közelítünk, ezt hívjuk tesszelációnak.

Milyen algoritmusokat néztünk a takarási feladat megoldására? Röviden ismertesd őket!

El kell dönteni, hogy a kép egyes részein milyen felületdarab látszik. Egy pont akkor takarja a másikat, ha az x és y koordinátái megegyeznek, és a z koordinátája kisebb a másik ponténál. Objektum tér algoritmusok: Logikai egységenként dolgozunk, nem függ a képernyő felbontásától --> nem fog menni. Képtér algoritmusok: Pixelenként döntjük el, hogy mi látszik, ilyen a sugárkövetés is.

Mi a triviális hátlapeldobás? Milyen feladat megoldásánál használtuk?

Tegyük fel, hogy az objektumok zártak, tehát a belsejüket nem látjuk, ha csak nem vagyunk benne az objektumban. Rögzítsük a körüljárási irányt (az óra mutató járásával megegyező, vagy ellentétes irány). Ha a transzformációk után a csúcsok sorrendje nem egyezik meg a megadott körüljárási iránnyal, akkor a lapot hátulról látjuk, így ezt nem kell kirajzolni, tehát eldobható.

Mi a festő algoritmus, milyen feladat megoldásánál használtuk?

Festő algoritmus: Rajzoljuk ki hátulról előre haladva a poligonokat. Ami közelebb van, azt később rajzoljuk ki, tehát ami takarva van, az valóban takarva lesz. A probléma az, hogy nem minden esetben lehet a poligonokat sorbarendezeni.

Ismertesd a Z-puffer eljárást!

A takarási feladatot pixelenként oldja meg. Minden pixelre megkeresi azt a poligont, amelynek a pixelen keresztül látható pontjának a Z koordinátája a legkisebb. A kereséshez minden pixelhez a feldolgozás adott pillanatának megfelelően tárolja az abban látható felületi pontok közül a legközelebbi Z pont koordinátáját, ezt a tömböt nevezzük Z -buffernek, vagy mélységi buffernek. A sokszögeket egyenként dolgozzuk fel, és meghatározzuk az összes olyan pixelt, ami a sokszög vetületén belül van. Amint egy pixelhez érünk, kiszámoljuk a felületi pont Z koordinátáját, és összehasonlítjuk a Z bufferben lévő, az adott pixelhez tartozó mélységértékkel. Ha az ott található mélységérték kisebb, akkor a már feldolgozott objektumok között van olyan, ami takarni fogja az aktuálist, tehát ezt a pontot nem kell kirajzolni, ha viszont a z -buffer beli érték nagyobb, akkor az aktuális sokszög az eddig feldolgozottakat takarja, ezért ennek a színét kell beírni az aktuális pixelbe, és egyúttal a Z értékét a z -bufferbe.

Milyen lokális illuminációs árnyalásokat vettünk? Ismertesd őket röviden!

Csak a fényforrásból érkező fényeket vesszük figyelembe az árnyalásnál, a visszaverődéseket és tükröződéseket nem. Ez nagymértékben leegyszerűsíti a fényviszonyok számítását.

Saját színnel árnyalás: Minden objektumhoz/primitívhez egy színt rendelünk, és kirajzoláskor ez lesz a pixelek értéke. Ez a leggyorsabb módszer, hiszen az illumináció gyakorlatilag egyetlen értékadás, viszont nem sok mindenre használható, nincs háromdimenziós hatása

Konstans árnyalás: A megvilágítást poligononként egyszer számítjuk ki, a szín homogén lesz a lapon belül. Gyors, mert a műveletek száma csak a poligonok számától függ, a pixelek számától független. Íves részeket nem tartalmazó, diffúz, egyszínű objektumoknál használható.

Mi a Goaraud árnyalás? Mi a Phong árnyalás? Hasonlítsd össze a kettőt!

Gouraud árnyalás: A poligonok csúcspontjaiban értékeli ki a fényvisszaverődést, a lapon pedig lineáris interpolációval számítjuk ki a színeket. Ez egy számításigényesebb módszer, N db megvilágítás számítás + minden pixelre interpoláció. Az előzőeknél szebb eredményt kapunk, jól használható abban az esetben, ha a visszaverődés a lapokon tényleg megközelítőleg lineárisan változik. Jó eredményt kapunk diffúz felületeknél, viszont tükröződő, vagy spekuláris felületek esetén nem használható.

Phong árnyalás: A Phong árnyalás az árnyalási egyenletben szereplő a fényforrás és a kamera irányába mutató egységvektorokat, illetve a normálvektort interpolálja a háromszög csúcspontjaiban érvényes adatokból, az árnyalási egyenletet pedig minden pixelre külön értékeli ki. Ez a leg-számításigényesebb árnyalási mód, és a legjobb minőségű is egyben. Az árnyalás nem függ a poligonok számától, a csillanás akár a poligon közepén is meg tud jelenni.

Pont egyenesre vágásakor mikor tartjuk meg a pontot, mitől függ?

Ha a pont az egyenes normálisával egy irányban fekszik, akkor megtartjuk a pontot, ha nem, akkor eldobjuk.

Pont egyenesre vágásakor az egyenes normálvektoros alakját használó módszer hány szorzást és összeadást/kivonást végez? A homogén alakot használó mennyit? Vektorműveleteket nézve mi a műveletigény?

$\langle p-p_0, n \rangle \geq 0 \iff (p_x-p_{0x})n_x + (p_y-p_{0y})n_y \geq 0 \rightarrow 3 \text{ db } +/-, 2 \text{ db } *, \text{ VM: } 1 \text{ db } +/-, 1 \text{ db } *$

$e=[e_1, e_2, e_3] \quad e_1^2+e_2^2+e_3^2=1, \quad p=[p_1, p_2, p_3] : e \cdot p \geq 0 \iff e_1p_1+e_2p_2+e_3p_3 \geq 0 \rightarrow 2 \text{ db } +/-, 3 \text{ db } *, \text{ VM: } 0 \text{ db } +/-, 1 \text{ db } *$

Térben megadható-e egy egyenes egy pontjával és a normálvektorával?

Nem, csak síkban

Pont síkra vágásakor mit mondhatunk a normálvektoros és homogén alakot felhasználó módszerek skalár műveletigényeiről? Vektorműveleteket nézve mi a műveletigény?

$\langle q-p, n \rangle \geq 0 \iff (q_x-p_x)n_x + (q_y-p_y)n_y + (q_z-p_z)n_z \geq 0 \rightarrow 5 \text{ db } +/-, 3 \text{ db } *, \text{ VM: } 1 \text{ db } +/-, 1 \text{ db } *$

$e \cdot q \geq 0 \iff e_1q_1+e_2q_2+e_3q_3 \geq 0 \rightarrow 3 \text{ db } +/-, 4 \text{ db } *, \text{ VM: } 0 \text{ db } +/-, 1 \text{ db } *$

Szakasz egyenesre (félsíkra) vágásakor a végpontok és a vágóegyenes szempontjából milyen eseteket különböztetünk meg? Mi van, ha az egyik végpontja a szakasznak a vágóegyenesre esik?

Ha a szakasz mindkét végpontja az egyenes normálisával egy irányban fekszik, akkor megtartjuk a pontot. Ha a szakasz egyik végpontja sem fekszik az egyenes normálisával egy irányban, akkor eldobjuk a szakaszt. Ha a szakasz A végpontja az egyenes normálisával egy irányban fekszik, a B végpontja viszont nem, akkor a vágóegyenes és a szakasz metszéspontjától az A végpontig megtartjuk a szakaszt, a metszésponttól a B végpontig pedig eldobjuk. Ha a szakasz egyik végpontja a vágóegyenesre esik, akkor a szakaszt megtartjuk

Legyen $P=(4,2)$, $q=(9,-5)$. Mik lesznek a vágott szakasz p' , q' végpontjai, ha az egyenes egyenlete amire a vágást végezzük $y=-1=Y_{\max}$ (azaz a tőle felfelé eső pontokat vágjuk ki)? Mi lesz a p'' , q'' , ha az előbbi után az $y=-4=Y_{\min}$ egyenessel is vágunk (az egyenes alatt található pontokat)?

A következő Cohen-Sutherland kódú pontpárok közül melyek azonosítanak olyan szakaszokat, amelyek biztosan a vágóablakon kívül lesznek? (0101,0110) (1001,0010) (0100,0001) (1000,0000) (0000,0000)

1001	1000	1010
0001	0000	0010
0101	0100	0110

Legyen adva egy háromszög a síkban a következő 3 pontjával: (4,4) (14,14) (-6,24). Az $x=0$, $y=0$, $x=10$, $y=10$ egyenesekre vágva sorra a hozzájuk tartozó $n=(1,0)$, $n=(-1,0)$, $n=(0,-1)$ normálisok által meghatározott félsíkban megtartva a pontokat mik lesznek a háromszög vágása után keletkező poligon csúcspontjainak koordinátái? (Sutherland-Hodgman)

$e1 \rightarrow p=(0,0)$, $n=(1,0)$
 $(x-0)*1+(y-0)*0 \geq 0 \rightarrow x \geq 0$ $c'_1=(0,21)$, $c'_2=(0,12)$
 $v_R=A-C=(4-(-6), 4-24)=(10, -20)=(1,-2)$
 $C'_2=(0, 24+6*(-2))=(0,12)$
 $v_1=B-C=(14-(-6), 14-24)=(20, -10)=(2,-1)$
 $c'_1=(0,24+3*(-1))=(0,21)$

 $(4,4), (10,10), (1,10)$

A Bresenham szakaszrajzoló algoritmusra teljesül-e, hogy a szakasz kezdő és végpontját is visszaadja (beszínezi)? Igen

Szükséges-e a Bresenham algoritmushoz lebegőpontos műveletek elvégzése?

Nem, egészszekkel számol

Az elsőre tárgyal Bresenham algoritmus csak egy bizonyos síkrészre működött. Melyik ez a síkrész? Hogyan valósítottuk meg, hogy lényegében ugyanezzel az algoritmusmaggal, de az egész síkrészre működjön az eljárás?

Naív: $|m| < 1$ -re

Nyolcadokra kéne bontani a síkot, és mindegyiket külön esetként kezelni

Jav: $|x_2-x_1| < |y_2-y_1| \rightarrow$ felcseréljük x -et és y -t (rajzolni is fordítva kell.). Ha $x_1 > x_2$, akkor felcseréljük $x_1 \leftrightarrow x_2$ és $y_1 \leftrightarrow y_2$

Az e hibtagot $|y_2-y_1|$ -gyel növeljük minden lépésben, y -nal y_2-y_1 előjele szerint haladunk.

Legyenek egy háromszög csúcspontjainak koordinátái (1,1) (9,5) (5,9), az óra járásával ellentétes irányban való csúcspont megadás mellett. Mik lesznek az oldalak irányvektorai, mik lesznek a belőlük kapott normálisok? Hogyan dönthetjük el egy p pixelről, hogy a háromszög belsejében van, vagy sem? ((x,y) +90 fokkal elforgatottja $(-y,x)$, -90 fokkal elforgatottja pedig $(y,-x)$).

Mi a textúra és a képtér alapú lekérdezés, mi a különbség köztük? Melyik illeszkedik jobban az inkrementális képszintézishez?

A textúrákban színinformációkat adunk meg.

textúratér --> képtér: Minden texelhez keressük a neki megfelelő pixelt (nem biztos, hogy minden pixel sorra kerül, de lehet, hogy van amelyik többször is)

Képtér --> textúratér: Minden pixelhez keressük a neki megfelelő texelt

Szükség van hozzá a paraméterezési és vetítési transzformációk inverzére

A képtér alapú leképezés illeszkedik jobban az inkrementális képszintézishez

Legyen $f(u,v)=[\cos(u)\sin(v), \sin(u)\sin(v), \cos(v)]$, az origó középpontú, 1 sugarú kör parametrikus egyenlete, $u \in [0, 2\pi]$, $v \in [0, \pi]$. A textúratérbeli u' , v' koordinátákat hogyan kapjuk az u, v -ből?

$$u', v' \in [0, 1] \times [0, 1] \quad u' = u \cdot \frac{1}{2\pi}, \quad v' = v \cdot \frac{1}{\pi}$$

Legyen adott egy háromszög a következő csúcsaival: $p_0=[0,0,0]$, $p_1=[0,2,0]$, $p_2=[4,0,0]$. Sorban a következő textúra koordinátákat rendeljük a fenti pontokhoz: $t_0=[0,0]$, $t_1=[0,1]$, $t_2=[1,0]$. Mi lesz a képtér alapú lineáris leképezés mátrixa? Milyen típusú affin transzformáció ez?

Legyen adott egy parametrikus felület, amelyet a sugárkövetés során egy sugárral metszünk. A metszéspont paramétertartománybeli helye $u=13\pi$, $v=4$. Mi lesz a metszésponthoz rendelt textúrankoordináta, ha a parametrikus felületem a $[0, 39\pi] \times [2, 6]$ -on van értelmezve?

$$u' = \frac{13\pi}{39\pi} = \frac{1}{3} \quad v' = \frac{v-2}{4} = \frac{1}{2} \quad \rightarrow (u', v') = \left(\frac{1}{3}, \frac{1}{2}\right)$$

Párhuzamos vetítés használata esetén helyes eredményt ad-e a textúrankoordináták lineáris interpolációja?

Hibás képet ad, ha nem csak affin transzformációkat használunk, tehát az esetek 98%-ában --> homogén osztással kell interpolálni

Képi textúrák használata esetén a nagyítás vagy a kicsinyítés problémája származik-e abból, hogy a képi textúránk egy mintavételezés eredménye?

Nagyítás: kicsi a pixel, tehát 1 texelre több pixel is jut

Kicsinyítés: a pixel nagy, tehát egy pixelre több texel is jut

Milyen előnyei és hátrányai vannak a procedurális textúrának? Melyik textúraszűrési probléma nem jelentkezik a használatuk esetén?

Sokkal kevesebb tárhely, tetszőleges felbontás (csak a numerikus pontosság a korlát), nem kell szűrés a nagyításhoz (a kicsinyítéshez igen!)

Nagy a számítási igénye, nem lehet vele bármit leírni, nem módosítható tetszőlegesen

Mi a vertex shader? Mi a feladata, milyen bemenő és kimenő adatokat vár, milyen elemeken dolgozik?

A vertex shader feladata a nézeti, világ és projektív transzformációk elvégzése, a modell koordináta-rendszerből NPKR-be visz. Bemeneti adatok: általában modell koordináta-rendszer belső pont, és egyéb attribútumok, kimenete NPKR belső pont, és egyéb attribútumok.

Mi a fragment shader? Mi a feladata, milyen bemenő és kimenő adatokat vár, milyen elemeken dolgozik?

Feladata a képernyőn megjelenő pixelek színének a meghatározása. Bemenete: a rasterizálás végén előálló fragmentnek, az össze attribútum elérhető, amit a vertex shaderben előállítottunk, textúra koordináták. Kimenet: egy színérték.

Az inkrementális képszintézis kapcsán tanult szerelőszalagunk melyik lépései programozhatóak a mai videokártyákon?

A mai videokártyákon programozható a Vertex shader, a Fragment shader, a Geometry shader és a tesszelláció

Jellemezd a DX9 grafikus szerelőszalagját, milyen lépései vannak, melyek konfigurálhatók, melyek programozhatóak?

Vertex data, Primitive data, Tesszelláció, Vertex processing, geometry processing, Textured surface, Texture sampler, Pixelprocessing, Pixel rendering. A Vertex és Pixel processing programozható, a többi konfigurálható.

Az absztrakt fényforrások során végzett számításoknál mire használjuk a felületi normális és a nézőpont felé mutató irányvektor által bezárt szöget?

Semmire! A felületi normális és a visszaverődési irány vektora által bezárt szöget a spekuláris visszaverődés kiszámításához, a visszaverődési irány és a nézőpont vektora által bezárt szöget a diffúz visszaverődéshez. A felületi normális és a nézőpont felé mutató vektor által bezárt szög az előző kettőnek az összege.

Hogyan számoltuk ki a Lambert törvény által meghatározott diffúz megvilágítást? Milyen shader parancsokat használtunk az implementáció során, azok mit csináltak?

BRDF: $L_{ref} = L_i k_d \cos \theta'$ --> L_i : aktuális fényforrás tul. (float4), k_d : diffúz szín (float4) az implementálás során float4 diffuseColor és float4 diffuseLight parancsokat használtuk. Ezek a diffúz komponens színét, és fényességét adják meg.

Kiszámítása: $\text{saturate}(\text{dot}(\text{normal}, \text{toLight}))$ --> ismerni kell hozzá a $\text{normal} = \mathbf{n} - t$ és a $\text{toLight} = \mathbf{l} - t$

A különböző absztrakt fényforrások esetén hogyan számoltuk ki a toLight nevű vektort? Mit jelképezett ez a vektor?

irány: a fény iránya, normalizált irányvektor (float3 lightDirection) $\text{toLight} = -\text{lightDirection}$

pont: A fény pozíciója, helyvektor (float3 lightPosition) $\text{toLight} = \text{normalize}(\text{lightPosition} - \text{worldPos})$

Spot: Fény iránya normalizált irányvektor (float3 lightDirection), fény pozíciója helyvektor (float3 lightPosition) : $\text{toLight} = \text{normalize}(\text{lightPosition} - \text{worldPos})$

Ambiens, diffúz és spekuláris modelleket használva milyen, a kirajzolt felülethez rendelt attribútumokat használunk a számításainkban?

Felület tul:

k_a : ambiens szín (float4),

k_d : diffúz szín (float4),

ks : spekuláris szín (float4),
n : spekuláris hatvány (float1),
x : világbeli pozíció (float3),
Fényforrás tul.:
Ld : diffúz fény (float4),
Ls : spekuláris fény (float4),
l : fényforrás felé mutató vektor (float3),
Színtér tul.:
La : ambiens fény (float4),
eye : kamera pozíció (float3),

A fragment shaderben számolva a megvilágítási modelleket, hogyan jutottunk világoordinátarendszerbeli pozíciókhoz és normálisokhoz? Miért kellett úgy megszerezni őket?

A felületi pont pozícióját és normálvektorát bemenetként kapja.
A kamera pozícióját globálisan adtuk át a szerelőszalagnak
A fényforrás felé mutató vektort a fényforrás típusától függően másként számoltuk

Adott egy spot fényforrás, ami a [4,8,0] pozícióból a [-1,0,0] irányban sugároz. A belső fénykört meghatározó kúp nyílásszöge 60 fok, míg a külső 90. A következő pontok körül melyek esnek a belső fénykörön belülre, melyek vannak a külső és a belső fénykör között, és melyek vannak a megvilágított tartományon kívül? [0,0,0] [0,4,0] [8,3,8] [4,10,2] [-4,4,0] [-4,5,0]

Egy felületi pont akkor van benne egy kúpban, ha a pontot a fényforrás pozíciójával összekötő egyenes a kúpon belül van.

Hasonlítsuk össze a geometria Brute-Force tárolását az indexbufferes megközelítéssel!

Brute force: A primitíveink poligonok, és ezeknek minden csúcsát tároljuk. Ha vannak közös csúcsaik, akkor ezeket többször is tároljuk feleslegesen. A közös csúcsokra annyiszor végezzük el a transzformációt, ahányszor szerepelnek. Nincsenek nyilvántartva a szomszédok, így a lekérdezések csak az összes csúcs bejárásával működnek. Egyetlen előnye, hogy egyszerű a tárolás.
Index pufferek: Minden csúcsot egyszer tárol egy nagy, közös tömbben. A poligonok csak hivatkoznak a csúcsok tömbjének elemeire. Tárolás hatékony, transzformálás is, lekérdezés még mindig nem igazán, bár ismerjük a közös csúcsokat.
PL: $N \times N$ négyzetből álló rács mérete puffer nélkül: $4N^2$, pufferrel: N^2+2N+1 ($+4N^2$ egész szám, index).
Ha egynél több négyzet van, már megéri.

Legyenek a kirajzolási primitívjeink háromszögek. Egy $N \times N$ négyzetből álló rács kirajzolásához indexpuffer nélkül illetve indexpufferrel hány darab csúcspontot kell tárolnunk? Mi történik, ha az indexek tárolását is bele vesszük a képletbe, feltételezve, hogy egy csúcsponthoz csak a három koordinátáját tároljuk, egy koordináta tárolásának helyigénye pedig megfelel az indexével?

$N \times N$ négyzetből álló rács mérete puffer nélkül: $4N^2$, pufferrel: N^2+2N+1 ($+4N^2$ egész szám, index). Ha egynél több négyzet van, már megéri.

A szárnyas él (Winged-edge) adatszerkezetnél milyen adatokat tárolunk egy élben?

kezdőcsúcs, végcsúcs, jobb lap, bal lap, balra és jobbra az előző és a következő él.

2*él	csúcs		lap		balra		jobbra	
	start	vég	Bal	Jobb	Előző	Következő	Előző	Következő
a	B	A	0	1	c	b	d	e

Winged-edge használata esetén írd algoritmust ami felsorolja az adott csúcspontba befutó összes élt!

Half-edge használata esetén hogyan tudjuk felsorolni az egy adott topológiai élre illeszkedő lapokat?

Half-edge használata esetén írd algoritmust, ami egy fél él törlését végzi el! A mutatókat is módosítsd!

Az $a=[5,1,4]$ és $b=[20,-5,10]$ pontok közötti lineáris interpolációval kapott szakasznak mik lesznek a $t=0, t=\frac{1}{3}, t=\frac{1}{2}, t=\frac{2}{3}, t=1$ paraméterértékekhez tartozó pontjai?

$(1-t)*a+t*b$

Legyenek adottak a $p_0=[0,0,0]$, $p_1=[4, 2, 1]$, $p_2=[10,10,10]$, $p_3=[-7,20,-4]$ pontok, továbbá $t_i=2^i$, $i=0,1,2,3$ paraméterértékek. Ezekkel képezzünk egy törött vonalat. Mi lesz a törött vonal paramétertartománya? Mik lesznek a $t=1, 2, 3, 4, 5, 6, 7, 8$ paraméterértékekhez rendelt pontok koordinátái?

$t \in [1,8]$

Igaz-e a törött vonalra, hogy minden kontrollponton (p_i) áthalad?

Egy n részből álló törött vonalnak $n+1$ kontrollpont kell. Az első és az n . kontrollpont a görbe végpontjai, az összes többin áthalad.

Lehet-e C^1 egy törött vonal egy csatlakozásánál (kontrollpontnál)?

Bizonyítsd be, hogy $\sum_{i=0}^n B_i^n(t) = 1$ teljesül a Bernstein polinomokra!

Mi volt Runge problémája?

Runge-problémája: az $f(x) = \frac{1}{1+25x^2}$ függvény közelítése során a magas fokú polinomok egyre nagyobb hibát generálnak

Mik a spline görbék? Hogyan segítenek Runge problémáján?

Hogyan működik Chaikin algoritmus?

Iterációs lépések sorozatával törött vonalból görbét csinál.

Az aktuális vezérlőpont halmazunk $\{p_i \in \mathbb{R}^3\} (i \in [0,n])$

Az iterációs lépés során az új vezérlőpont halmazunk: $\{q_i, r_i \in \mathbb{R}^3\} (i \in [0,n-1])$

$q_i = 3/4 * p_i + 1/4 * p_{i+1}$

$r_i = 1/4 * p_i + 3/4 * p_{i+1}$

Mi a szintérgráf, miket tárolunk az elemeivel?

A szintérgráf egy olyan adatszerkezet, amely a szintér különböző elemeit, tulajdonságait és az elemek alá-fölé rendeltségét tartalmazza. Egy irányított fa, ahol a csomópontok lehetnek: geometria, anyagjellemzők, fényviszonyok, kamera, transzformációk. Egy objektumhoz anyagjellemzők, és viselkedési minták is tartozhatnak.

Ha egy részfa gyökerén végrehajtunk egy transzformációt, az az egész részfára érvényes lesz.

Legyen adott egy origó középpontú 3 sugarú gömb (A) és egy origó középpontú 1 sugarú gömb (B). Mik lesznek a metszési szakaszai az A-B művelettel (halmazok különbsége) kapott lyukas gömbnek és az $r(t)=[0,0,10]+t[0,0,-1]$ sugárnak?

$(0,0,3) \rightarrow (0,0,1)$

$(0,0,-1) \rightarrow (0,0,-3)$

Mit kell tartalmaznia az idő függvényévé tett V "nézeti" transzformáció mátrixomnak, amennyiben a kamera látószögét is szeretnénk módosítani?

Ha V a View mátrix, akkor csak a kamera pozícióját, orientációját tartalmazza, ha V a View és a projection együtt, akkor tudjuk vele a látószöget is állítani.

Hogy néz ki egy valósidejű és egy nem valósidejű animációs program váza? Mik a főbb különbségek?

<p>Offline:</p> <p>$\Delta t = 1/FSP$</p> <p>For (t=t_start; t<t_end; t+=Δt):</p> <p> For o in objects:</p> <p> Mo=Mo(t)</p> <p> V=V(t)</p> <p> Render_scene_to_disk()</p> <p>Start_time = get_time()</p> <p>For (t=t_start; t<t_end; t+=Δt):</p> <p> Draw_frame(t)</p> <p> While (t-t_start+Δt > get_time()-start_time):</p> <p> Wait()</p> <p>Nem számít, hogy mennyi ideig tart kiszámolni egy képkockát, mert nem azonnal akarjuk megjeleníteni. Elkülönül a szintézis és a visszajátszás: először elmentjük a képkockát, aztán majd videóként lehet visszanézni. A felhasználó nem tud belenyúlni az animációba, olyan részletes színteret és animációt használunk, amit ki tudunk várni.</p>	<p>Valós idejű:</p> <p>def update () :</p> <p> # FrameUpdate () / UpdateScene ()</p> <p> t = get_time ()</p> <p> for o in objects :</p> <p> Mo = Mo(t)</p> <p> V = V(t)</p> <p>def render () :</p> <p> # Render () / DrawScene ()</p> <p> for o in objects :</p> <p> renderobject (o)</p> <p>A számítás után rögtön meg is jelenítjük a képkockákat, így a számításoknak elég gyorsnak kell lennie a folytonosság látszatához. A felhasználói eseményekre reagálni kell. Olyan részletgazdag lehet a szintér, amit még ezekkel a feltételekkel meg tudunk jeleníteni. Inkrementális képsintézist használunk.</p>
--	---

Hogyan érhetjük el, hogy valós idejű animáció során lassabb és gyorsabb gépeken is ugyanolyan sebességgel történjenek a szintérbeli események?

Ha nem azt tároljuk, hogy egy-egy frissítés alkalmával mennyit változzon, hanem azt hogy egységnyi idő alatt mennyit, majd minden frissítéskor lekérdezzük az előző frissítés óta eltelt időt és ezzel szorozzuk a tárolt mennyiséget, akkor egyforma mozgást kapunk a számítógép sebességétől függetlenül.

Mi a kulcskocka animáció?

Egy bonyolult mozgást nehéz lenne képlettel megadni. Inkább adjuk csak bizonyos időközönként, hogy mit szeretnénk látni. Ezek a kulcskockák. Egy tulajdonságot két kulcskocka között interpolációval számolunk ki.

Adott egy g tulajdonság, amire azt szeretném, hogy $g(5)=4$ és $g(9)=11$ teljesüljön. Lineáris interpolációval mi lesz a $g(t)$ interpolációja a g tulajdonságnak, $t \in [5,9]$?

Mikor nem tudjuk használni az előrehaladó kinematikát?

Az előrehaladó kinematika nem használható, ha a strukturális összefüggés erősen nem lineáris. Hiába interpolálunk egyenletesen az állapottérben, a végberendezés vadul kalimpálhat a kulcspontok között. (pl.: láb mozgása a talajon)

Mi az inverz kinematika?

Az inverz kinematika a kritikus végberendezés helyzetét interpolálja, majd az állapotváltozók értékét végberendezés interpolált helyzetéből számítja vissza.