April 4 (Monday).

Topics: 1.30 GE focused on Shadow Computation.

Example: Given

1. x_w-y_w-z_w world coordinate Right Hand System. $\vec{P_i} + \lambda(\vec{P_s} - \vec{P_i})$

$\vec{R}$

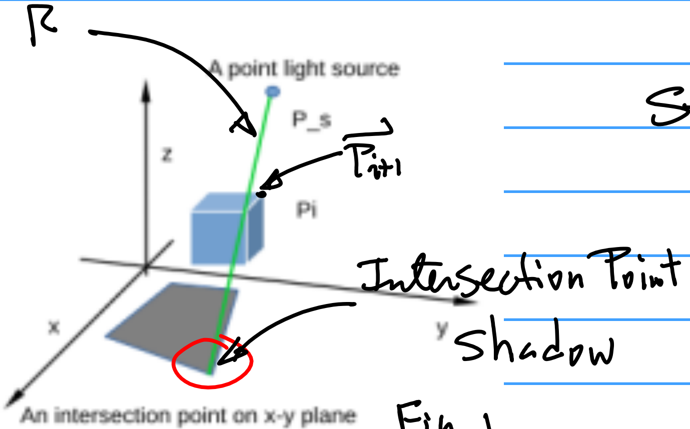


Fig. 1

2. A Given Cube $\{\vec{P_i}(x_i, y_i, z_i) \mid i = 1, 2, \cdots, N\}$

Counter Clockwise

3. A given Point light Source $\vec{P_s}(x_s, y_s, z_s)$

4. Generate Ray Equation / Ray Cast x_w-y_w plane, Produces Shadow if Blocked by the Cube.

References:

2021F-101b-notes-cmpe240-2021-12-1.pdf

Step 1. Generate A Ray Cast / Equation

$$\vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i}) \quad \dots (1)$$

Starting from $\vec{P_s}$, Passing through $\vec{P_i}$

$$\vec{d}(x, y) = \vec{P_s} - \vec{P_i} \quad \dots (1b)$$

Intersection Point: $\vec{P_i}$

Between Ray $\vec{R}(x, y)$, and x_w-y_w plane.

Step 2. To find the intersection Point on x_w-y_w plane

Define a plane equation.

a. Define A Normal vector

$\vec{n}(n_x, n_y, n_z)$

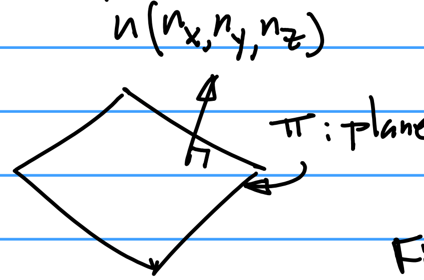


Fig. 2

use the normal vector to define the plane π.

b. Form a vector (on a plane) from $\vec{P_i}$ and $\vec{P_{i+1}}$, $(\vec{P_{i+1}} - \vec{P_i})$ a line

$$\vec{n} \cdot (\vec{P_{i+1}} - \vec{P_i}) = 0 \quad \dots (2)$$

Take

$\vec{v}(v_x, v_y, v_z)$, $\vec{a}(a_x, a_y, a_z)$

to Replace $\vec{P_i}$, $\vec{P_{i+1}}$

Assume $\vec{a}(a_x, a_y, a_z)$ is a Known Vector; And $\vec{v}(v_x, v_y, v_z)$ is unknown, But an arbitrary Point on the plane π.

Hence, Eqn(2) becomes

$$\vec{n} \cdot (\vec{v} - \vec{a}) = 0 \quad \ldots (2^*)$$

Now, find the intersection point defined By the Ray Eqn(1). In order to that, we will need to find λ.

Since, the intersection point $\vec{P_i}$ is the Common Point By the Ray and the plane π. we have

$$\begin{cases} \vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i}) & \ldots (3a) \\ \vec{n} \cdot (\vec{v} - \vec{a}) = 0 & \ldots (3b) \end{cases}$$

$\vec{n}(n_x, n_y, n_z)$, Normal Vector has to be Known,

$\vec{a}(a_x, a_y, a_z)$ is known on π.

Starting from the plane Eqn (3b).

$$\vec{n} \cdot (\vec{v} - \vec{a}) = 0$$

where $\vec{v} = \vec{R}$, e.g.

$$\vec{n} \cdot (\vec{v} - \vec{a}) \Big|_{\vec{v} = \vec{R}} = 0 \quad \ldots (4)$$

$$\vec{n} \cdot (\vec{R} - \vec{a}) \Big|_{\vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i})} = 0$$

$$\vec{n} \cdot (\vec{P_i} + \lambda(\vec{P_s} - \vec{P_i}) - \vec{a}) = 0$$

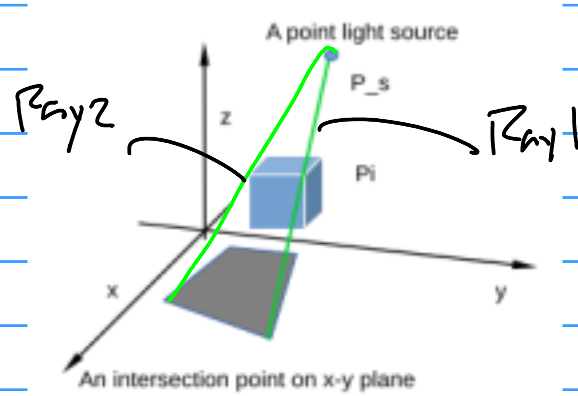
$$\vec{n} \cdot \vec{P_i} + \lambda \vec{n} \cdot (\vec{P_s} - \vec{P_i}) - \vec{n} \cdot \vec{a} = 0$$

$$\lambda \vec{n} \cdot (\vec{P_s} - \vec{P_i}) = \vec{n} \cdot \vec{a} - \vec{n} \cdot \vec{P_i}$$

$$\lambda = \frac{\vec{n} \cdot \vec{a} - \vec{n} \cdot \vec{P_i}}{\vec{n} \cdot (\vec{P_s} - \vec{P_i})}$$

$$= \frac{\vec{n} \cdot (\vec{a} - \vec{P_i})}{\vec{n} \cdot (\vec{P_s} - \vec{P_i})} \quad \ldots (5)$$

Note λ is NOT the intersection pt. it allows us to use Ray Eqn(1) to find the intersection.



use Eqn(5) to find more than one intersection

April 7th (Wed)

Example: Ref (see pp. 1. github LectureNotes, 2021F-101b-~)

Given $\vec{P_s}(-20, 110, 200)$, A vertex of a given Cube $\vec{P_i}(100, 100, 110)$. Find the intersection pt to Draw Shadow.

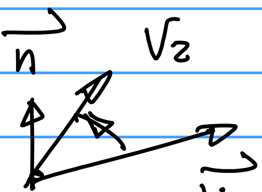
Sol First, Ray Equation, Eqn(1):

$$\vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i}) \quad \ldots (1)$$

then, plane Eqn for X_w-Y_w plane

$$\vec{n} \cdot (\vec{v} - \vec{a}) = 0$$

Find $\vec{n}$,



$$\vec{V_1} \times \vec{V_2} = \vec{n} \quad \ldots (2)$$

Let $\vec{V_1} = X_w \vec{i}$, $\vec{V_2} = Y_w \vec{j}$

$$\vec{n} = X_w \vec{i} \times Y_w \vec{j}$$
$$= (0, 0, 1)$$

Then, 2nd find λ for the Ray Equation, from Eqn.

$$\lambda = \frac{\vec{n} \cdot (\vec{a} - \vec{P_i})}{\vec{n} \cdot (\vec{P_s} - \vec{P_i})} \quad \ldots (5)$$

where $\vec{a} = (0, 0, 0)$, Hence,

$$\lambda = \frac{-\vec{n} \cdot \vec{P_i}}{\vec{n} \cdot (\vec{P_s} - \vec{P_i})}$$

Now, hand Calculation, also for Coding

$$\lambda = \frac{-\vec{n} \cdot \vec{P_i}}{\vec{n} \cdot (\vec{P_s} - \vec{P_i})}$$

$$= -\frac{n_x x_i + n_y y_i + n_z z_i}{n_x(x_s - x_i) + n_y(y_s - y_i) + n_z(z_s - z_i)}$$

From the given Condition $\vec{n}(0, 0, 1)$, Therefore

$$\lambda = \frac{-\cancel{n_z} \cdot z_i}{\cancel{n_z} \cdot (z_s - z_i)} = -\frac{z_i}{z_s - z_i}$$

$$= -\frac{110}{200 - 110} = -\frac{110}{90} = -11/9$$

Now, back to the Ray Equation

$$\vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i})$$

$$\vec{R} = (100,100,110) - \frac{11}{9}(-20-100,\ 110-100,\ 200-110)$$

$$= (100,100,110) - \frac{11}{9}(-120, 10, 90)$$

$$= \left(\frac{1100\times 120}{9},\ -\left(100 - \frac{110}{9}\right),\ 110-110\right)$$

$$= \left(\frac{1100\times 120}{9},\ \frac{110}{9} - 100,\ 0\right) = (246.7, 87.8, 0)$$

please finish this calculation.

Now, Coding Part. Same Code on github.

Note:

a. Define Normal vector $\vec{n}$ for X_w-Y_w plane

b. Note the "typedef struct" for Defining 3D points.

2018F-116-11diffuse20181114.cpp

```
world.X[47] = 0; world.Y[47] = 0; world.Z[47] = 1; // normal vector for x-y plane
```

```
typedef struct {
    float X[UpperBD], Y[UpperBD], Z[UpperBD];
} pworld;
```

```
pworld  world;
pviewer viewer;
```

$$\lambda = \frac{\vec{n}\cdot(\vec{a} - \vec{P_i})}{\vec{n}\cdot(\vec{P_s} - \vec{P_i})} \quad \cdots \quad (5)$$

$$= -\frac{n_x x_i + n_y y_i + n_z z_i}{n_x(x_s - x_i) + n_y(y_s - y_i) + n_z(z_s - z_i)}$$

Now, λ Calculation. Eqn (5)

```
//----------lambda for intersection pt on xw-yw plane----------*
float temp = (world.X[47]*(world.X[46]-world.X[45])
            +(world.Y[47]*(world.Y[46]-world.Y[45]))
            +(world.Z[47]*(world.Z[46]-world.Z[45]));
float lambda = temp / ((world.X[47]*(world.X[45]-world.X[7]))
                      +(world.Y[47]*(world.Y[45]-world.Y[7]))
                      +(world.Z[47]*(world.Z[45]-world.Z[7])));
float lambda_2 = temp / ((world.X[47]*(world.X[45]-world.X[6]))
                        +(world.Y[47]*(world.Y[45]-world.Y[6]))
                        +(world.Z[47]*(world.Z[45]-world.Z[6])));
```

Note, Substitute λ to Ray Equation to find the intersection point

```
//-----------ray equation to find intersection pts------------*
world.X[48] = world.X[45] + lambda*(world.X[45] - world.X[7]); // Intersection pt p7
world.Y[48] = world.Y[45] + lambda*(world.Y[45] - world.Y[7]); // Intersection pt p7
world.Z[48] = 0.0;

world.X[49] = world.X[45] + lambda_2*(world.X[45] - world.X[6]);  //intersection pt p6
world.Y[49] = world.Y[45] + lambda_2*(world.Y[45] - world.Y[6]);  //intersection pt p6
world.Z[49] = 0.0;
```

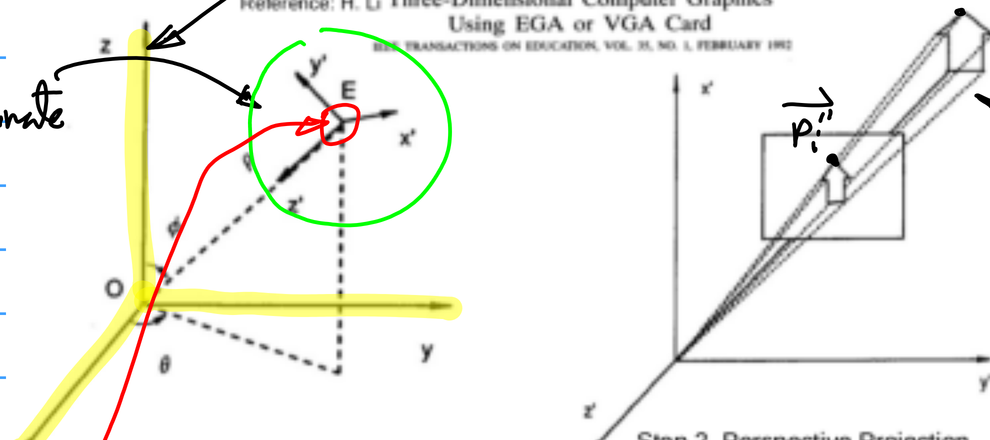
$$\vec{R} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i}) \rightarrow \begin{cases} x = x_i + \lambda(x_s - x_i) \\ y = y_i + \lambda(y_s - y_i) \\ z = z_i + \lambda(z_s - z_i) \end{cases}$$

Assignment in-Class Show & Tell. Implement Intersection Computation on LPC 1769, "Show+Tell" Demo in Class. On April 11 (Monday),

To Be Able to Display 3D Graphics on 2D Display Devices. Let's Define Transformation Pipeline. 1. Define World-Coordinate System; Right Hand System $x_w-y_w-z_w$

3D Transformation Pipeline Technique

Reference: H. Li Three-Dimensional Computer Graphics Using EGA or VGA Card



2. Viewer Coordinate System $x_v-y_v-z_v$

Left-Hand System

3. Virtual Camera is located $\vec{E}$ (e_x, e_y, e_z)

$\vec{P_i}$

$\vec{P_i''}$

Ray Cast

Step 1. World-to-viewer transform

Step 2. Perspective Projection

$$\mathbf{T} = \begin{bmatrix} -\sin\theta & \cos\theta & 0 & 0 \\ -\cos\phi\cos\theta & -\cos\phi\sin\theta & \sin\phi & 0 \\ -\sin\phi\cos\theta & -\sin\phi\cos\theta & -\cos\phi & \rho \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$x_p = x_e\left(\frac{D}{z_e}\right)$$

$$y_p = y_e\left(\frac{D}{z_e}\right)$$

Harry Li, Ph.D.

Example: Display Shadows on 2D Display Device.

Step 1. World To Viewer Transform.

$$T = \begin{bmatrix} -\sin\theta & \cos\theta & 0 & 0 \\ -\cos\phi\cos\theta & -\cos\phi\sin\theta & \sin\phi & 0 \\ -\sin\phi\cos\theta & -\sin\phi\cos\theta & -\cos\phi & \rho \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \cdots (1)$$

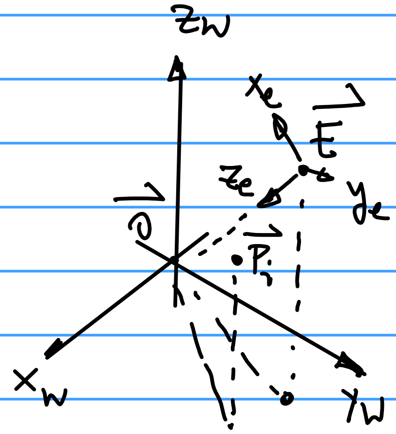


Fig. 1.

Everything is defined in the World Coordinate System x_w-y_w-z_w including a Virtual Camera.

$\vec{E}(x_e, y_e, z_e)$, x_e-y_e-z_e "Viewer" Coordinate System.

Given $\vec{P_i}(x_i, y_i, z_i)$ in x_w-y_w-z_w World Coordinate, Represent this point in x_e-y_e-z_e Coordinate System.

$$\vec{P_i'} \quad \begin{bmatrix} x_i' \\ y_i' \\ z_i' \\ 1 \end{bmatrix} = \begin{bmatrix} -\sin\theta & \cos\theta & 0 & 0 \\ -\cos\phi\cos\theta & -\cos\phi\sin\theta & \sin\phi & 0 \\ -\sin\phi\cos\theta & -\sin\phi\cos\theta & -\cos\phi & \rho \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \vec{P_i} \quad \cdots (1\text{-}b)$$

Assume $\vec{E}(x_e, y_e, z_e) = (200, 200, 200)$

Physical meaning of Transformation Matrix T.

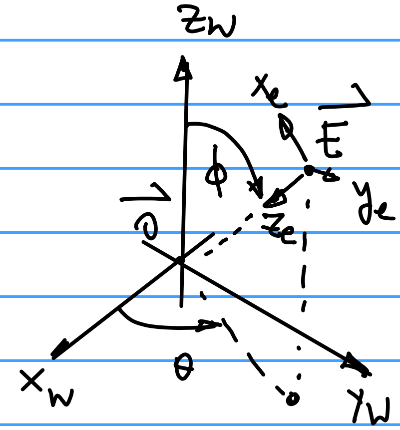


Fig. 2

θ: Angle from the dash line on x_w-y_w plane w.r.t positive x_w-axis

ϕ: Angle Between z_w & z_e.

ρ (rho): $\rho = \sqrt{x_e^2 + y_e^2 + z_e} \quad \cdots (2)$

distance from $\vec{E}$ to the origin $\vec{o}$ of x_w-y_w-z_w.

Suppose $\vec{E}(200, 200, 200)$ is given, Find $\cos\theta, \sin\theta, \cos\phi, \sin\phi$ for T-matrix.

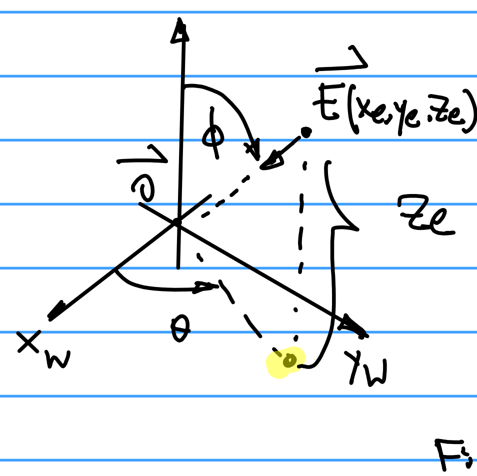
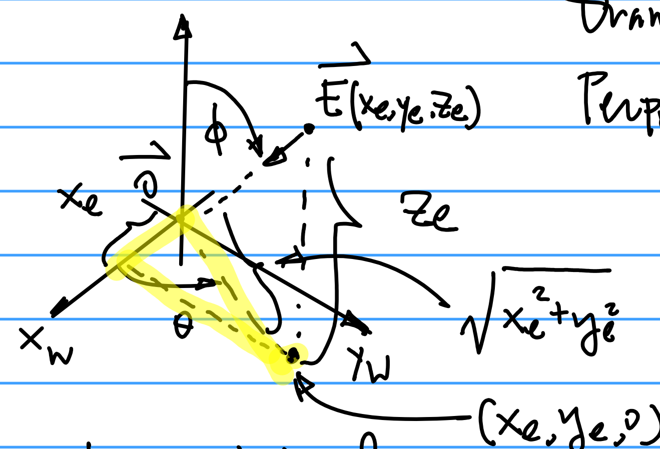


Fig. 3

Draw A Line passing through $\vec{E}$ Perpendicular to X_w-Y_w $\Rightarrow$ Form an intersection point.

Draw A Line Passing through the intersection point on X_w-Y_w plane on the plane and Perpendicular to X_w-axis.

Fig.4

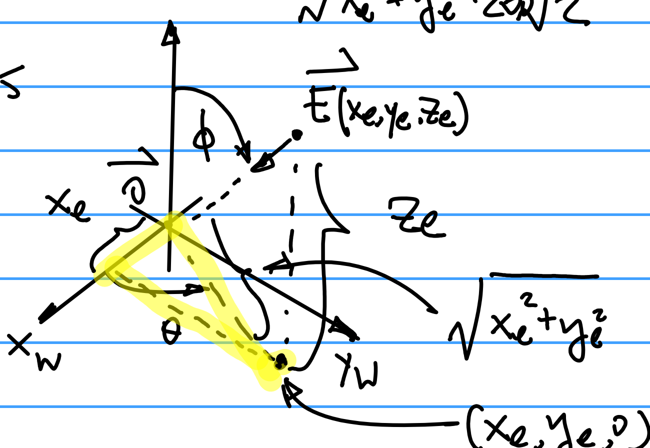


We can form a triangle on X_w-Y_w plane, as in Fig.4, hence

$$\cos\theta = \frac{x_e}{\sqrt{x_e^2+y_e^2}} = \frac{200}{200\sqrt{2}} = \frac{\sqrt{2}}{2}$$

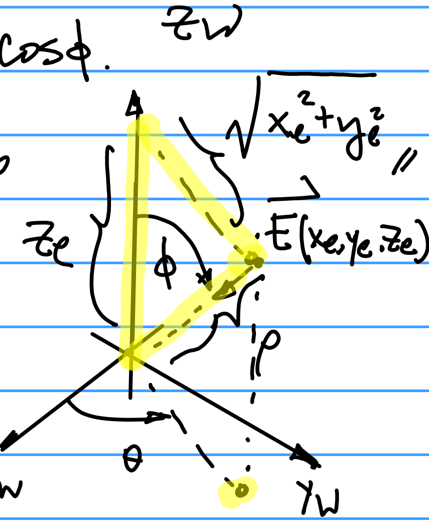
similarly, $$\sin\theta = \frac{y_e}{\sqrt{x_e^2+y_e^2}} = \frac{200}{200\sqrt{2}} = \frac{\sqrt{2}}{2}$$

Fig.5



Find $\cos\phi$.

Fig.6



Draw A Line Passing $\vec{E}(x_e, y_e, z_e)$ Perpendicular to Z_w-axis

$$\cos\phi = \frac{z_e}{\rho} = \frac{z_e}{\sqrt{x_e^2+y_e^2+z_e}}$$

$$= \frac{200}{200\sqrt{3}} = \sqrt{3}/3$$

$$\sin\phi = \frac{\sqrt{x_e^2+y_e^2}}{\sqrt{x_e^2+y_e^2+z_e}} = \frac{200\sqrt{2}}{200\sqrt{3}}$$

$$= \frac{\sqrt{2}\cdot\sqrt{3}}{3} = \frac{\sqrt{6}}{3}$$

Homework: Due April 18th (Monday)

1. Draw A World Coordinate System X_w-Y_w-Z_w axis, with X_w Red, Y_w Green, Z_w Blue

2. Draw A cube, size Length = 100, floats 10 unit Above X_w-Y_w plane. in other word $\vec{P_i}$ (100, 100, 110);

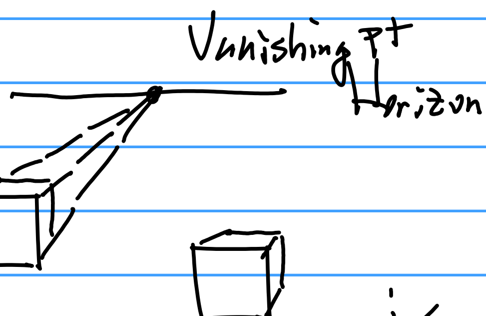
3. Draw a point light Source $\vec{P_s}(-20, 110, 200)$, And Ray Cast to Connect $\vec{P_s}$ to $\vec{P_i}$; use Green Color.

4. Compute the Shadow point $\vec{P_i'}$, Draw the intersection Point to link $\vec{P_s} - \vec{P_i} - \vec{P_i'}$

Note: You may want to Adjust the $\vec{P_s}$ position, So this $\vec{P_i'}$ (Intersection Point) Can be Visible on Your Display

Step 2. Perspective Projection

$$x_p = x_e\left(\frac{D}{z_e}\right)$$
$$y_p = y_e\left(\frac{D}{z_e}\right) \quad \ldots (3)$$



Eqn (3) project a point $\vec{P_i}(x_i, y_i, z_i)$ in X_e-Y_e-Z_e onto a 2D Display Device, like LCD.

April 13 (Wed)

Topics: 1° Perspective Projection

2° Diffuse Reflection

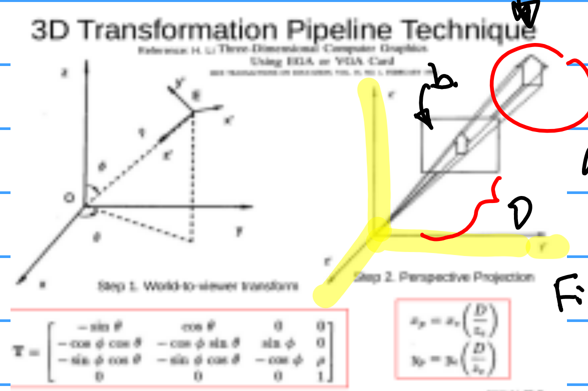


Fig.1

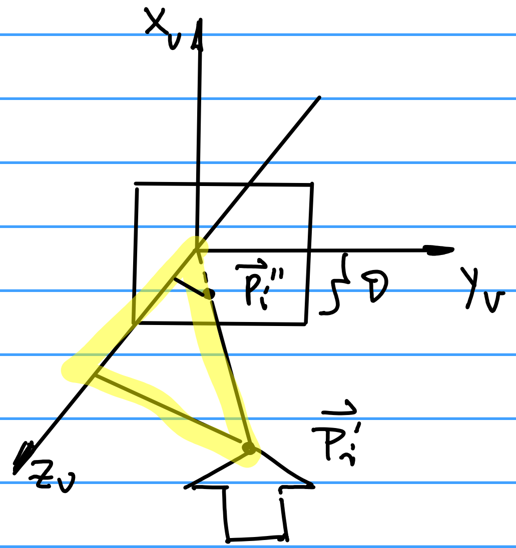
Note: X_v-Y_v-Z_v is Left Hand System.

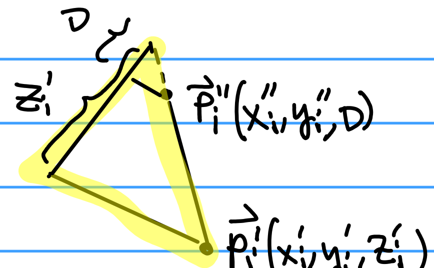
a. It is a 3D object

b. projection Plane, distance to the viewer Coordinate System $(0,0,0) \rightarrow (x_e, y_e, z_e)$ in $X_w Y_w Z_w$. D is the distance.

c. Origin of X_v-Y_v-Z_v. → Camera Location, Camera is modeled as "pin-Hole" model.

Projection is formulated By using Similar Triangles in Fig.1





$x_p = \frac{D}{z_e} x_e$ from Eqn(8).

Or,

$$x_i'' = \frac{D}{z_i'} x_i'$$

Similarly,

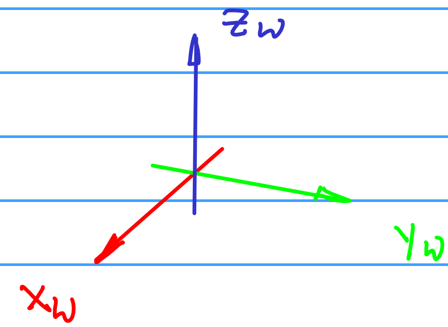
$$y_p = \frac{D}{z_e} y_e \quad \text{or}$$

$$y_i'' = \frac{D}{z_e} y_i'$$

Homework, Due A week from Today

April 20th

1. Draw a World Coordinate System, X_W: Red, Y_W: Green, Z_W: Blue, Design the size (X_W, Y_W, Z_W, 50 units)



2. Design By Defining Dimension of a Cube.

(Example: length = 100)

$\vec{P}_i(x_i, y_i, z_i) = (100, 100, 110)$

Elevate the cube By 10 units.

3. Draw the Cube on the LCD

4. Submission:

a. Screen Capture of your XPRESSO Screen, which Shows your Name (Folder Name) And your Program (Partial) — 0.5 pt

b. Take a photo of your display. With Entire Prototype System of your own — 0.5 pt.

5. Submission to CANVAS.

Note: You will need transform from a virtual coordinate System to physical coordinate

Consider Diffuse Reflection.

2018S-23-lec7-DiffuseReflection-v6-2018-4-25.pdf

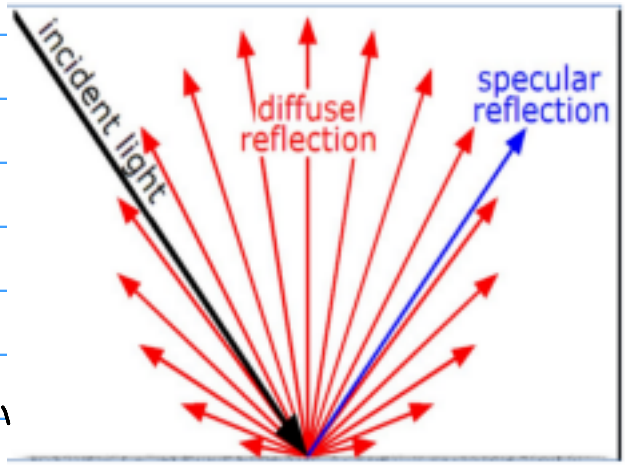


Fig. 1

https://en.wikipedia.org/wiki/Diffuse_reflectio

Definition: A Reflection from an object Surface uniformly in all directions.

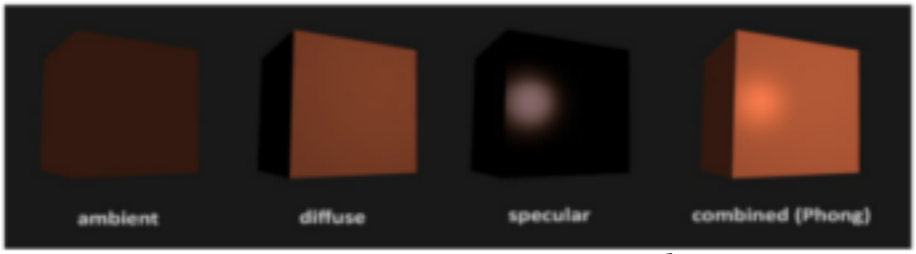


Fig2.

1. Define Reflectivity, A property of An Object surface.

$\vec{K_r} = (k_r, k_g, k_b) = (r, g, b)$ ....(1)

$0 \le r, g, b \le 1$

Note: for a Black object,

$r=0, g=0, b=0$

for a green leaf.

$r \approx 0;\ g \ne 0, 0 < g;\ b \approx 0$

2. Color Space. R, G, B

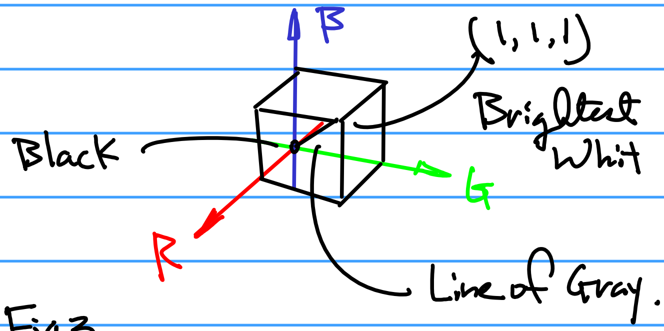
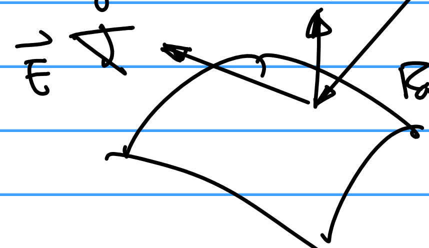


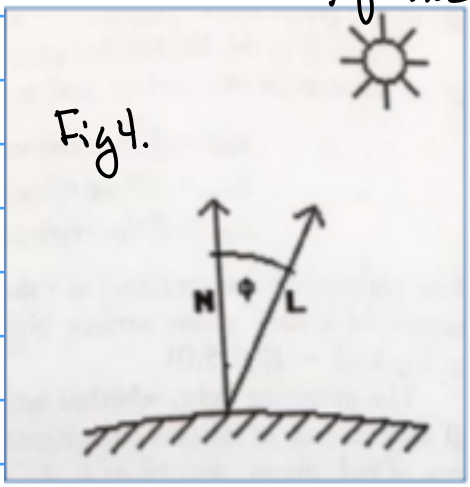
Fig3

3. Viewing Angle v.s. Incident Light.



a. Perceived color is independent of viewing Angle.

b. Normal vector $\vec{N}$ and incident Light $\vec{L}$ ($\vec{R}$ Ray Cast) form An Angle ϕ, the color Intensity follows



$\cos\phi$ function.

Color Intensity.

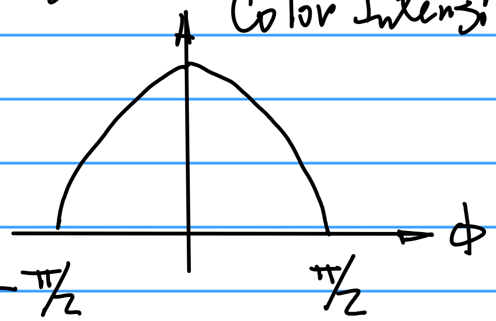


Fig 5.

$\phi = 0$, $\cos\phi = 1$, Strongest Reflection, highest Intensity.

$\phi = \pi/2$, $\cos\phi = 0$, No Reflection, hence, no color.

4. Distance.

April 18 (Monday)

Topics: Continue on Diffuse Reflection.

Light Intensity Satisfies the relationship Below.

$$\text{Intensity} \sim \frac{1}{\|\vec{r}\|^2} \quad \dots (1)$$

$\vec{r}$ Ray Equation.

$$\vec{r} = \vec{P_i} + \lambda(\vec{P_s} - \vec{P_i})$$

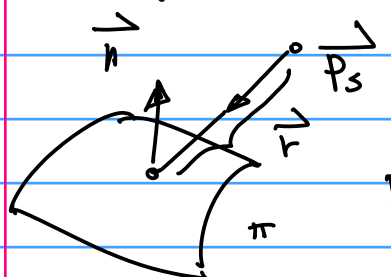


Fig. 1

Example: Given $\vec{P_s}(x_s, y_s, z_s)$ And A Cube.

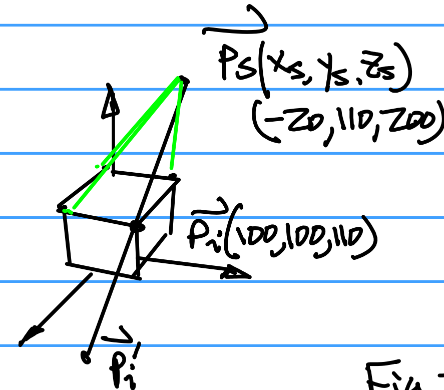


Fig. 2

$\vec{P_i}$ shadow Point.

Find Diffuse Reflection at $\vec{P_i}(x_i, y_i, z_i)$

Sol: Formulation.

$$\vec{I_{diff}}(x, y, z) = (I_{diff,r}(x,y), I_{diff,g}(x,y), I_{diff,B}(x,y))$$

$$\vec{K_d} = (K_r, K_g, K_b) \quad \dots (2)$$

$$0 \le K_r, K_g, K_b \le 1$$

Note: For Simplicity, we will focus on one type of Reflectivity for Now; "r" red

$$\boxed{I_r = K_{d_r} \frac{\vec{n} \cdot \vec{r}}{\|\vec{n}\| \|\vec{r}\|} \frac{1}{\|\vec{r}\|^2}} \quad \dots (3)$$

Reflectivity for "r"

$\cos\phi$. $\vec{n} \cdot \vec{r} = \|\vec{n}\| \|\vec{r}\| \cos\phi$

Compeato April 18, 22 12

$$\|\vec{r}\|_2 = \sqrt{(x_s - x_i)^2 + (y_s - y_i)^2 + (z_s - z_i)^2} \quad \cdots (4)$$

$$\|\vec{r}\|_2^2 = (x_s - x_i)^2 + (y_s - y_i)^2 + (z_s - z_i)^2$$

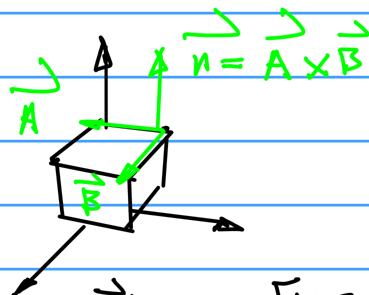
Suppos $\vec{P_s}(-20, 110, 200)$, $\vec{P_i}(100, 100, 110)$

Assum: $K_{dr} = 0.8$

Find Normal vector for the Cube Surface.

$\vec{n}$ Defined By Vector Cross Product

$$\vec{n} = \vec{A} \times \vec{B}$$



$\vec{p_i}$ Fig 3

But the Surface of the Cube is in Parallel with $x_w - y_w$ plane.

$\therefore \vec{n}(0, 0, 1)$

Now, find

$$\cos\phi = \frac{\vec{n} \cdot \vec{r}}{\|\vec{n}\| \|\vec{r}\|}$$

Directional vector of $\vec{r}$

$$= \frac{(n_x, n_y, n_z) \cdot (x_i - x_s, y_i - y_s, z_i - z_s)}{\sqrt{n_x^2 + n_y^2 + n_z^2} \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2 + (z_i - z_s)^2}}$$

Substitute the given Condition, So

$$\cos\phi = \frac{z_i - z_s}{\sqrt{(100+20)^2 + 10^2 + 90^2}} = \frac{110 - 200}{\sqrt{\Delta}}$$

Therefore $\cos\phi = \left| \frac{110 - 200}{\sqrt{\Delta}} \right|$

$$= \left\| -\frac{90}{\sqrt{\Delta}} \right\| = \frac{90}{\sqrt{\Delta}} \qquad \frac{1}{\sqrt{\Delta}} \ll 1.$$

And the distance from $\vec{P_s}$ to $\vec{P_i}$

$$\|\vec{r}\|^2 = (x_s - x_i)^2 + (y_s - y_i)^2 + (z_s - z_i)^2$$

$$= 120^2 + 10^2 + 90^2$$

Therefore, we have

$$I_{diff} = 0.8 \times \frac{90}{\sqrt{\Delta}} \times \frac{1}{120^2 + 10^2 + 90^2}$$

Note: Very Small! Need Post Processing for Better Visualization.

$$= 0.8 \times \frac{90}{\sqrt{120^2 + 10^2 + 90^2}} \times \frac{1}{120^2 + 10^2 + 90^2} = 2.12 \times 10^{-5}$$

Since the result is very Small, we need to Perform Post Processing.

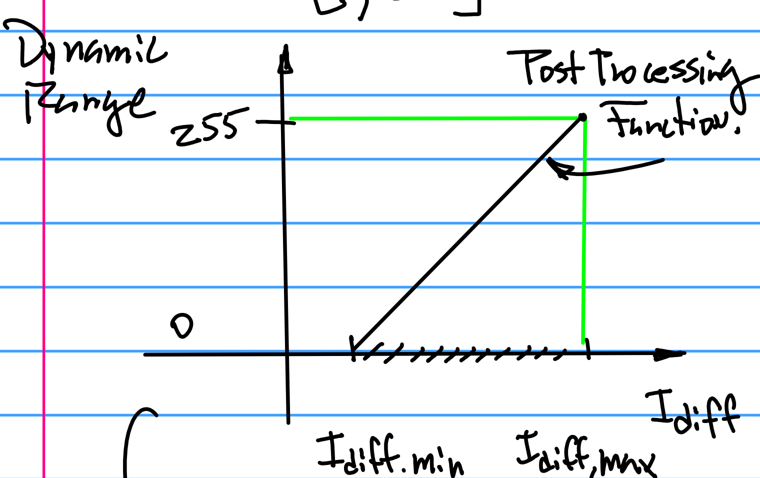
Note: Directional vector $\vec{P_i} - \vec{P_s}$ gives Negative Value, $\vec{P_s} - \vec{P_i}$ Positive

$\rightarrow$ take Absolute Value.

Objective: To Scale up the result in Eqn(3), to match the entire dynamic range of the display device.

1. Dynamic Range of the Display Device → 8 bits for Each Primitive color.

   $2^8 = 256$. Dynamic Range $[0, 255]$



By Eqn(3)

Fig(4-a)

Add An offset to this function

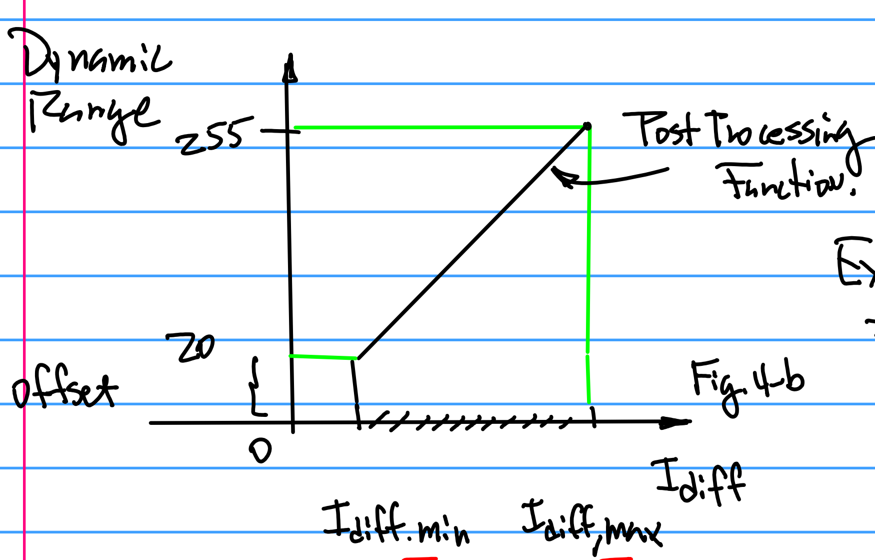


Fig. 4-b

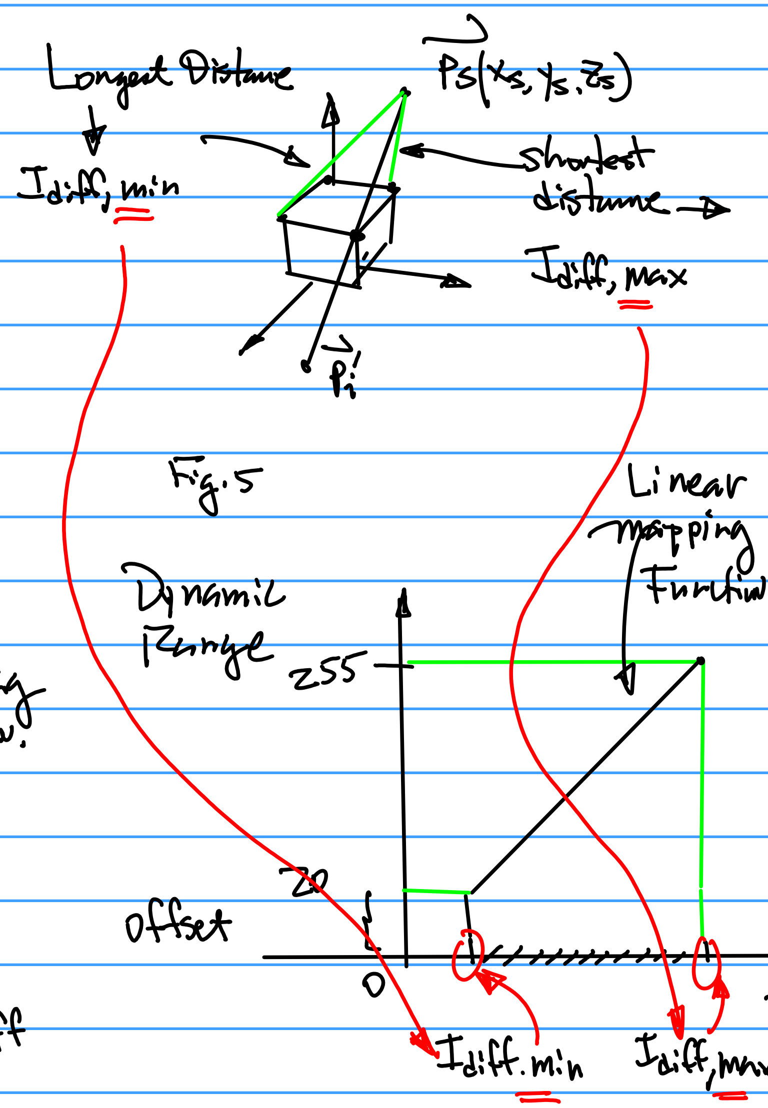


Fig. 5

Post processing function

$$\frac{x - x_2}{y - y_2} = \frac{x_1 - x_2}{y_1 - y_2} \quad \dots (5)$$

April 20 (Wed)

Example: Post Processing Technique From Fig 5, And Eqn (5).

CmpE240 April 20 (Wed), 22

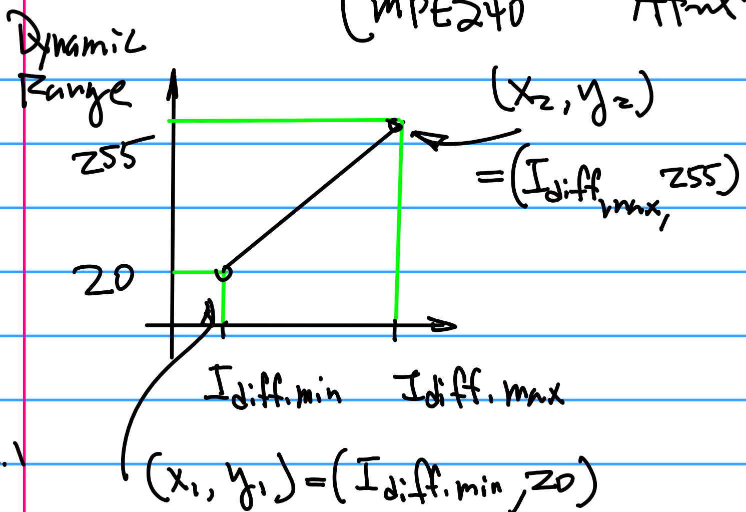


Fig. 1

From Eqn (5),

$$\frac{x - x_2}{y - y_2} = \frac{x_1 - x_2}{y_1 - y_2}$$

$$\frac{y_1 - y_2}{x_1 - x_2} = \frac{y - y_2}{x - x_2}$$

$$y = y_2 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_2)$$

$$\boxed{y = \frac{y_2 - y_1}{x_2 - x_1} x - \frac{y_2 - y_1}{x_2 - x_1} x_2 + y_2} \quad \cdots (1)$$

$$a\,y = b\,x + c, \quad y = \frac{b}{a} x + \frac{c}{a} \quad \cdots (2)$$

$$\frac{b}{a} = \frac{y_2 - y_1}{x_2 - x_1} \quad \cdots (2\text{-}b)$$

$$\frac{c}{a} = -\frac{y_2 - y_1}{x_2 - x_1} x_2 + y_2 \quad \cdots (2\text{-}c)$$

Substitute $(I_{diff,min}, 20)$, $(I_{diff,max}, 255)$

$$I_{dyn} = \frac{255 - 20}{I_{diff,max} - I_{diff,min}} x - \frac{255 - 20}{I_{diff,max} - I_{diff,min}} I_{diff,max} + 255$$

Project on Diffuse Reflection
Due May 8th (Sunday)
Compute Diffuse Reflection.
You can work the Diffuse Reflection on 4 corners of the Cube.
Make sure post processing is Implemented good.

General Guide for Diffuse Reflection

Point(s) →	Line(s) →	Surface/plane
Vertex ✓	Boundaries	planes

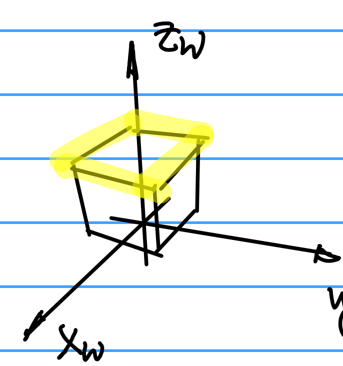


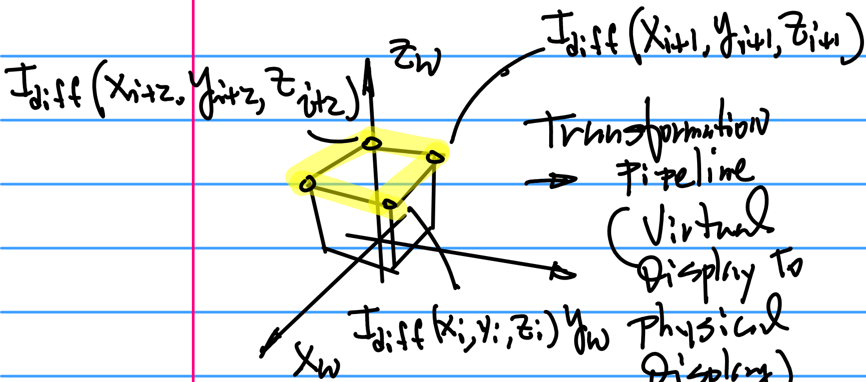
Fig 2

Two Phased Approach

- Phase I: Compute Diffuse Reflection After Transformation Pipeline
- Phase II: Map the Diffuse Reflection Result in Phase I to Actual Hardware Display Device

Example: Phase I Computation

In X_w-Y_w-Z_w



Display Device

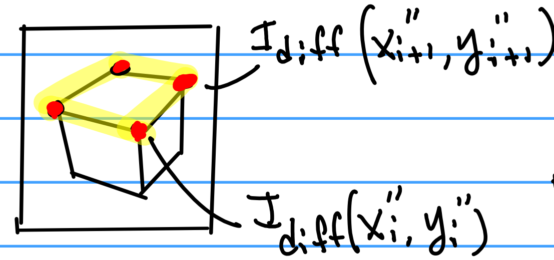


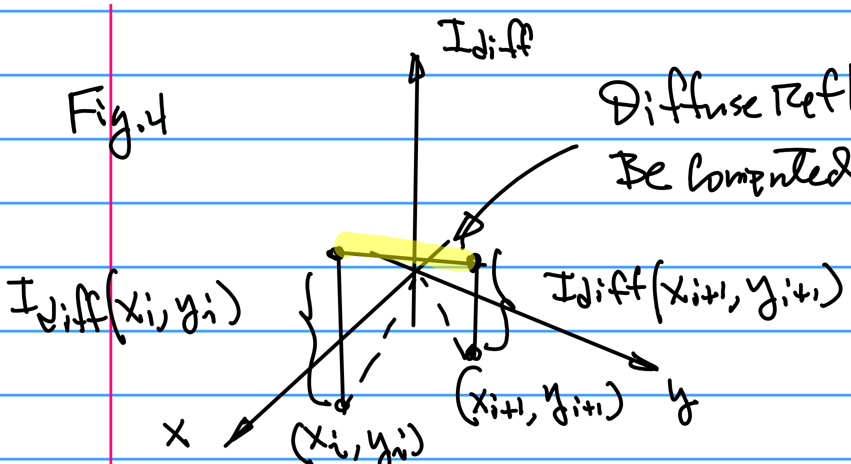
Fig 3.

Diffuse Reflections on 4 Corners Are Computed Already.

For Simplicity Purpose, with the understanding of working on the physical display, we use (x_i, y_i) not (x''_i, y''_i).

/ 2018F / 2020S-APL29-BilinearDiff1.jpg

Fig.4



Diffuse Reflection to Be Computed, on the Bounding line with $\vec{P_i}(I_{diff,i})$ and $\vec{P}_{i+1}(I_{diff,i+1})$ as Starting & Ending point.

x, y are from the Display Device

/ 2018F / 2020S-APL27-Bilinear1.jpg

Project the "yellow Line" to x-z plane in Fig4. So it is just a function of x, then Compute Diffuse Reflection. → Similary, project the "yellow Line" to y-z plane, a function of y. then Compute Diffuse Reflection

↓

Combine both By Average Operation.

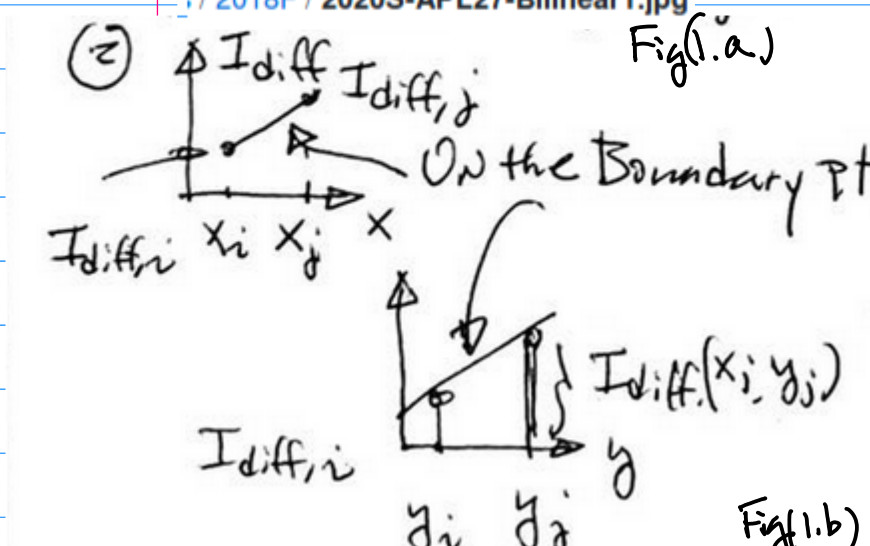
April 25 (Monday)

Topic: Boundary Line Computation for 3D G.E. Design.

Example: Formulation of Bilinear Interpolation to Compute Boundary Line (yellow) in Fig. 4.

Project the "yellow" Line onto X_w-Z(I_{diff}) axis

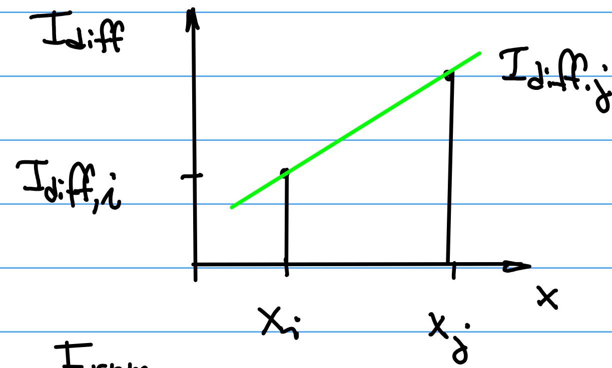
/ 2018F / 2020S-APL27-Bilinear1.jpg



Fig(1.a)

Fig(1.b)

Take care of the Diffuse Reflection on the Boundary Line w.r.t. x variable



Fig(1.c)

From,

$$\frac{x - x_2}{y - y_2} = \frac{x_1 - x_2}{y_1 - y_2} \quad \cdots (5)$$

then, derived the following Equations

$$\boxed{y = \frac{y_2 - y_1}{x_2 - x_1} x - \frac{y_2 - y_1}{x_2 - x_1} x_2 + y_2} \quad \cdots (1)$$

$$a y = b x + c, \quad y = \frac{b}{a} x + \frac{c}{a} \quad \cdots (2)$$

$$\frac{b}{a} = \frac{y_2 - y_1}{x_2 - x_1} \quad \cdots (2\text{-}b)$$

$$\frac{c}{a} = -\frac{y_2 - y_1}{x_2 - x_1} x_2 + y_2 \quad \cdots (2\text{-}c)$$

Once the diffuse Reflection(s) with respect to x independent variable is Computed, then the diffuse Reflection with respect to y independent Variable is Computed, then

From 1-c.

(x_1, y_1) :

$x_1 = x_i$

$y_1 = I_{diff,i}$

and

$x_2 = x_j$, $y_2 = I_{diff,j}$

$$\text{Actual Single pt. Diff. Ref} = \frac{1}{2}\left(I_{diff,x} + I_{diff,y}\right)$$

Therefore,

2020S-APL29-BilinearDiff2.jpg

$$I_{diff,x} = \frac{I_{diff,j} - I_{diff,i}}{x_j - x_i} x - \frac{I_{diff,j} - I_{diff,i}}{x_j - x_i} x_j + I_{diff,j} \quad \ldots (3)$$

For I_{diff} w.r.t y. we have (Symmetric)

$$I_{diff,y} = \frac{I_{diff,j} - I_{diff,i}}{y_j - y_i} y - \frac{I_{diff,j} - I_{diff,i}}{y_j - y_i} y_j + I_{diff,j} \quad \ldots (4)$$

Finally, Put them together,

$$I_{diff} = \frac{1}{2}\left(I_{diff,x} + I_{diff,y}\right) \quad \ldots (5)$$

Example:

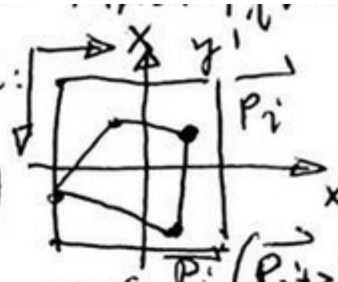


Fig. 2

x-y physical Display → x'-y' Virtual Display

Diffuse Refle.

$I_{diff}(x_i, y_i)$

$I_{diff}(x_{i+3}, y_{i+3}) = I_{diff}(x_j, y_j)$

$(x_i, y_i) = (10, 12)$, $(x_j, y_j) = (8, -14)$

Given $\vec{I}_{diff}(10, 12) = (100, 0, 0)$,

$\vec{I}_{diff}(8, -14) = (80, 0, 0)$.

Find $I_{diff}(x, y)$ on the Boundary Line

Now, DDA Algorithm.
(Digital Differential Algorithm)

Display Device has finite resolution. → "gaps" Problem

Example: For A finite Resolution Display Device Below,

Suppose $y=4x+1$ is given for $x=0$, $y=1$ starting point. Let draw a straight line for $x=0,1,2,\cdots,K$

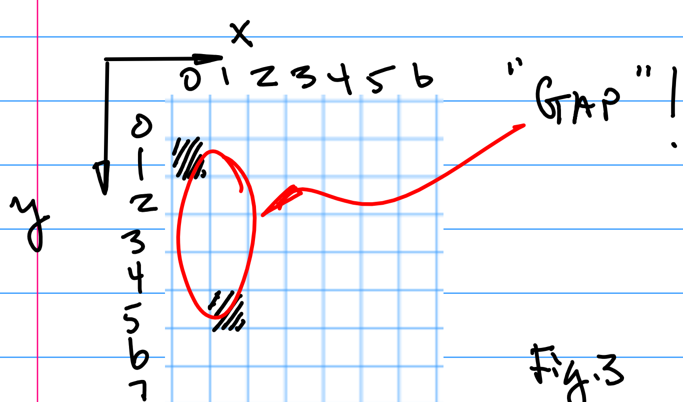


Fig.3

for $x=1$, $y=4x+1\Big|_{x=1}=5$