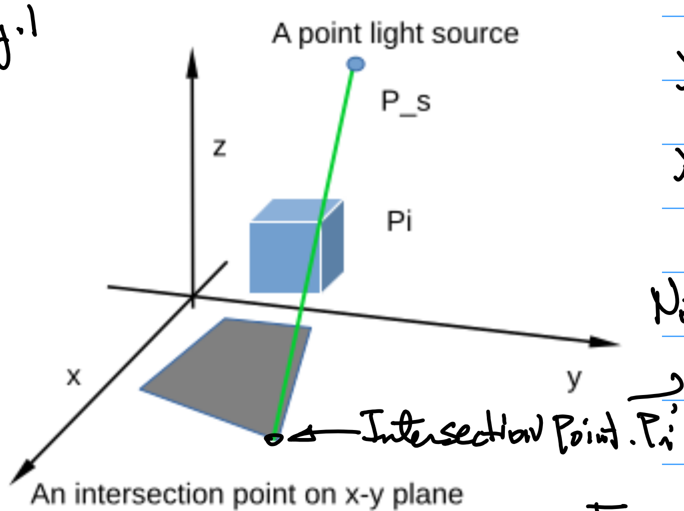
Nov. 15 (Monday).

Example: 3D Shadow Computation.

From Eqn (2) pp. 50.

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

Fig. 1



Ray Equation: pp. 46.

$$\vec{R}=\vec{P_i}+\lambda(\vec{P_s}-\vec{P_i}) \quad \cdots (3)$$

The Intersection Point on x_w-y_w plane, $\vec{P_i'}$ is a common pt shared by Ray Eqn (3) and plane eqn (2).

$$\begin{cases} \vec{R}=\vec{P_i}+\lambda(\vec{P_s}-\vec{P_i}) & \cdots (4\text{-}1) \\ \vec{n}\cdot(\vec{v}-\vec{a})=0 & \cdots (4\text{-}2) \end{cases}$$

From Eqn (4-2),

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

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

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

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

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

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

Note: λ together with the Ray Equation (3), will give the intersection point.

Example: Given a single point light source $\vec{P_s}(-20, 110, 200)$, $\vec{P_i}(100, 100, 110)$. Find intersection point to plot shadow.

Sol:

From Eqn (5), find λ.

Since

$\vec{n}=(0,0,1)$, $\vec{a}=(0,0,0)$

$\vec{P_i}=(100,100,110)$

Hence,

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

$$=\frac{(n_x,n_y,n_z)(a_x,a_y,a_z)-(n_x,n_y,n_z)\cdot(x_i,y_i,z_i)}{(n_x,n_y,n_z)(x_s-x_i,y_s-y_i,z_s-z_i)}$$

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

Note: Define Letter(s) for my initial for Linear Decoration. Latter with Furth Discussion.

```
typedef struct{
    float X[30], Y[30];
} letter;
```

$n_x = 0, n_y = 0, n_z = 1.$

Therefore, the above equation Becomes

$$= \frac{0+0+0 \quad -(0+0+1 \cdot z_i)}{0+0+1 \cdot (z_s - z_i)}$$

$$= -\frac{z_i}{z_s - z_i}$$ from the given condition

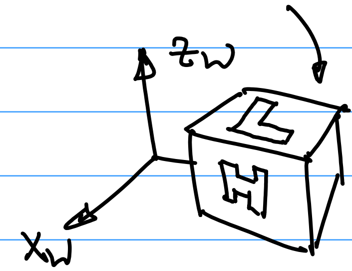
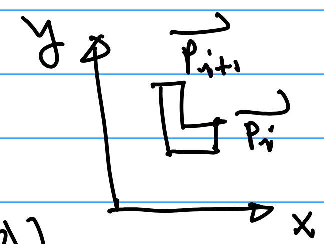


Fig. 2.

$z_i = 110, z_s = 200,$

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

Define 2D Patterns



2D Pattern to Decorate 3D plane.

Substitute λ into the Ray Equation (3)

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

$$= (100,100,110) + \lambda\left((-20,110,200) - (100,100,110)\right)$$

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

$$= \left(100 + \frac{11}{9} \times 130,\ 100 - \frac{11}{9} \times 10,\ 110 - \frac{11}{9} \times 90\right)$$

$$= \left(100 + \frac{11 \times 130}{9},\ 100 - \frac{11 \times 10}{9},\ 110 - 110\right)$$

$$= \left(100 + \frac{11 \times 130}{9},\ 100 - \frac{11 \times 10}{9},\ \underline{\underline{0}}\right)$$

On $x_w - y_w$ plane, so z must Be 0!

C/C++ Implementation.

2018F-116-11diffuse20181114.cpp

2018F-117-12dda.cpp

```
//define projection plane
world.X[4] = 60.0;      world.Y[4] = -50.0;    world.Z[4] = 0.0;//p4 of box
world.X[5] = 60.0;      world.Y[5] = 50.0;    world.Z[5] = 0.0; //p5 of box

world.X[6] = 60.0;      world.Y[6] = 50.0;    world.Z[6] = 100.0;//p6 of box
world.X[7] = 60.0;      world.Y[7] = -50.0;    world.Z[7] = 100.0;//p7 of box. Pi
```

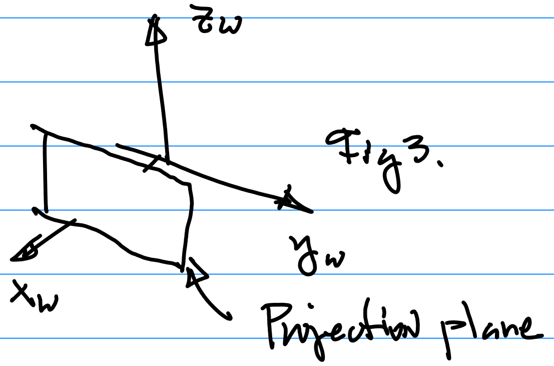


Fig 3.

(2) Point Light Source $\vec{P_s}(x_s, y_s, z_s)$

```
//47 = normal vector, 46 = A, 45 = Ps, 7 = top left box vertex
world.X[45] = -200.0; world.Y[45] = 50.0; world.Z[45] = 200.0; // Ps (point source)
world.X[46] = 0; world.Y[46] = 0; world.Z[46] = 0; // arbitrary vector A on x-y plane
world.X[47] = 0; world.Y[47] = 0; world.Z[47] = 1; // normal vector for x-y plane
```

(1) normal vector $\vec{n}$ $(0,0,1)$;

λ Computation is implemented Below,

```
//----------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])));
```

$$\lambda = \frac{n_x \cdot a_x + n_y \cdot a_y + n_z \cdot a_z - (n_x x_i + n_y y_i + n_z z_i)}{n_x(x_s - x_i) + n_y(y_s - y_i) + n_z \cdot (z_s - z_i)} \quad \text{OR}$$

$$= \frac{n_x(a_x - x_i) + n_y(a_y - y_i) + n_z(a_z - z_i)}{n_x(x_s - x_i) + n\,(y_s - y_i) + n_z \cdot (z_s - z_i)}$$

CmpE240 (II)

Nov. 17 (Wed)

3D G.E. Design On Diffuse Reflection

Ref: from class github.

- 2018F-115-lab-DiffuseReflection-Rubrics.txt
- 2018F-116-11diffuse20181114.cpp
- 2018F-117-12dda.cpp
- 2018F-118-13diffuseInterpolation20181127....
- 2018S-17-Lab-report-rubrics.txt

- 2018S-22-lec7-Diffuse...
- 2018S-23-lec7-Diffuse...

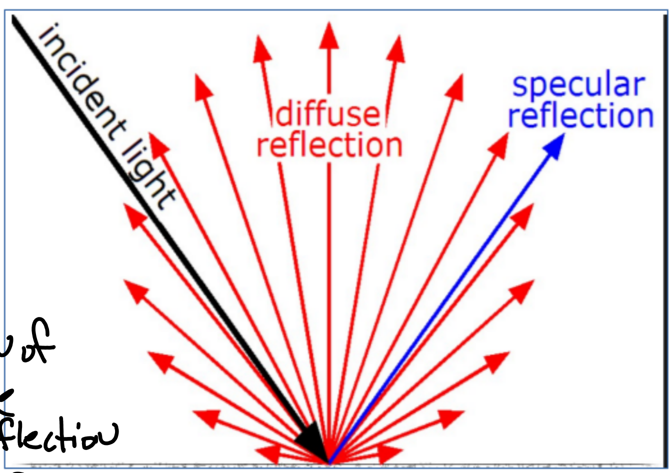


Illustration of Diffuse Reflection

(1) https://en.wikipedia.org/wiki/Diffuse_reflection

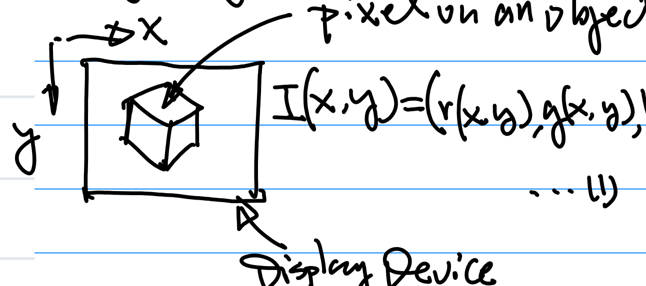
Chapter 12 (2) My Book-in-Progress

Lighting Models with Emphasis on Diffuse Reflection

In the lighting model formulation, very often you will see 3 different type of lighting models as shown in the following Figure:

54

Example: Generate Realistic Looking Graphics, Simulate/Formulate Lighting model.



$I(x,y) = (r(x,y), g(x,y), b(x,y))$ ... (1)

Color image defined By 3 Primitive colors, r — red, g — green, b — blue

Each primitive color is represented as 8 bit value.

$r(x,y) \in [0,255]$, $g(x,y) \in [0,255]$, and $b(x,y) \in [0,255]$

3 Type of Light Contributors to generate the color

$I(x,y) = I_1(x,y) + I_2(x,y) + I_3(x,y)$ ... (2)

Diffuse Reflection — Specular Reflection — Ambient Light

Note: Specular Reflection is the reflection which generates high light, it is a function of "Eye" e.g. Virtual Camera location.

Ambient Light (Reflection), are those Coming from indirect light Source(s).

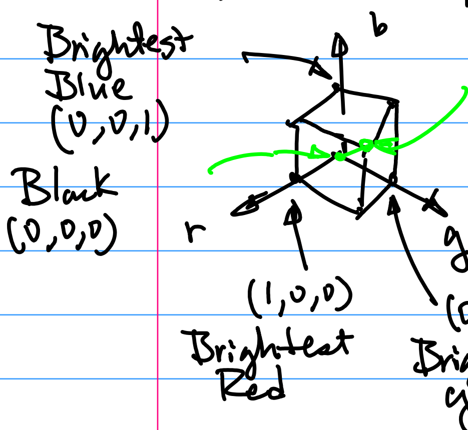
Example: Color Intensity generated by indirect Light when Viewing the object(s), for example, underneath a table.

Math Description: A constant.

Diffuse Reflection.

A Reflection Reflects in-coming light uniformally in all different directions

3D Vector Color Space



White (Brightest)

Green Line from (0,0,0) to (1,1,1)

Color is grey. Intensity Increases.

About color of An Object: Characteristic of the Object. It depends on reflectivity of the Object itself.

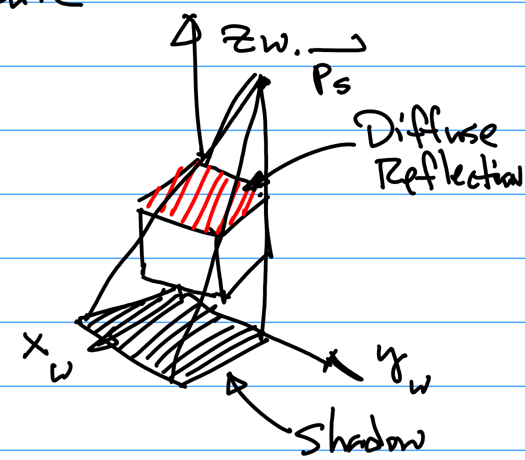
Nov. 22nd (Monday)

Note: Last Project Diffuse Reflection + Decoration Algorithm → 3D Graphics Processing Engine

2021F-109b-project-DiffuseReflection-Rubrics-...

1. Count 20 pts., Due Dec 8th (Wed) 11:59 pm. (No Late Submission) Submission on CANVAS.

2. (1) Solid Cube



(2) Diffuse Reflection on the top Surface, with 1 primitive color, Red

(3) Tree On One Surface

Self Design

2-Letter Font On the Surface

Your Initial

3. please follow the requirements Posted On-Line

(1) IEEE Style Report 5 pages.

(2) Exported Project; (3) photo of the Implementation; (4) 5 Second Video Clips

Diffuse Reflection Formulation.

2018S-23-lec7-DiffuseRefl...

1. The surface with reflectivity as K_d = (k_r, k_g, k_b), e.g., diffuse coefficients;

K coefficient of Reflectivity.

K_d : "d" for diffuse Reflection.

$\vec{K}_d(K_r, K_g, K_b)$

Reflectivity for Red

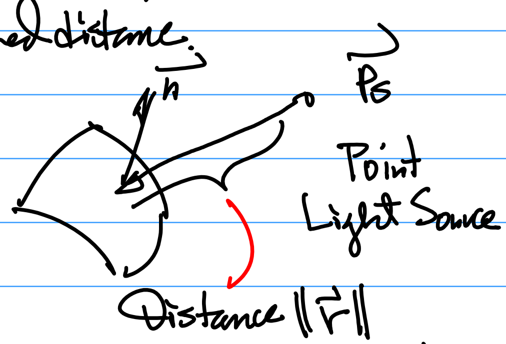
~ for green

~ for Blue

$$0 \le K_r \le 1, \quad 0 \le K_g \le 1, \quad 0 \le K_b \le 1 \quad \ldots (1)$$

If $K_r = K_g = 0$, $K_b = 1$, then we have "blue" color (Highest — Full Reflection of Blue).

2. Light (color) Intensity from $\vec{P}_s$ to a surface is formulated as a decay function to the Squared distance.

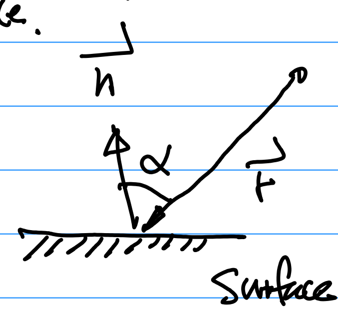


$\vec{r}$ Ray Equation from $\vec{P}_s$ to $\vec{P}_i$ on the surface.

Intensity (color)

$$\frac{1}{\left(\sqrt{x_r^2 + y_r^2 + z_r^2}\right)^2} = \frac{1}{\|\vec{r}\|^2}$$

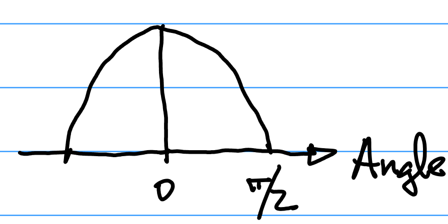
3. Intensity (Color) is a function of the angle of the incoming point light source.



$$\vec{I}_{diff} = (\text{Reflectivity})\left(\frac{1}{\|\vec{r}\|_2^2}\right)\left(\begin{array}{c}\text{Angle} \\ \text{of the} \\ \text{incoming} \\ \text{light}\end{array}\right) \quad \ldots (2)$$

at a particular point on a surface

Cos(Angle of the incoming Light):



Mathematically:

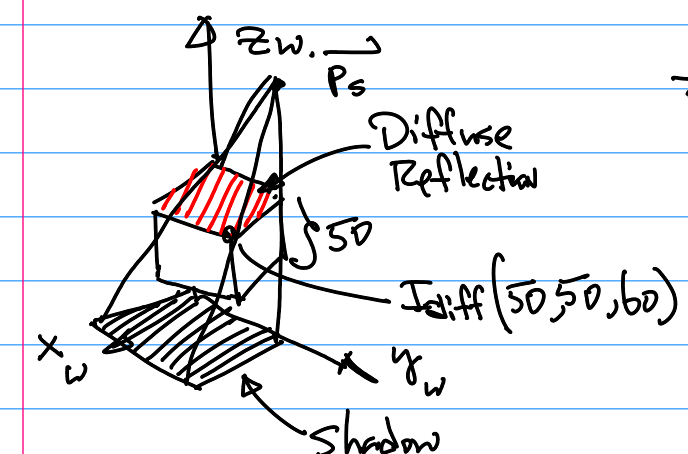
$$\vec{r} \cdot \vec{n} = \|\vec{r}\| \|\vec{n}\| \cos\alpha \quad \ldots (3)$$

$$\cos\alpha = \frac{\vec{r} \cdot \vec{n}}{\|\vec{r}\| \|\vec{n}\|} \quad \ldots (3\text{-}b)$$

Hence:

$$\boxed{\vec{I}_{diff} = K_d \cdot \frac{1}{\|\vec{r}\|^2} \cdot \cos\alpha = K_d \cdot \frac{1}{\|\vec{r}\|^2} \cdot \frac{\vec{n}\cdot\vec{r}}{\|\vec{n}\|\|\vec{r}\|}} \quad \ldots (4)$$

Example: Given Conditions



$\vec{P}_s(40,60,120)$. Cube is floating by $z=10$.

Find $\vec{I}_{diff}(50,50,60) = ?$

Sol: First, $K_d = (K_r, K_g, K_b) = (0.8, 0, 0)$

Then, the distance from P_s to $\vec{P}_i(50,50,60)$

$$\|\vec{r}\|^2 = (x_s - x_i)^2 + (y_s - y_i)^2 + (z_s - z_i)^2$$
$$= (40-50)^2 + (60-50)^2 + (120-60)^2$$
$$= 10^2 + 10^2 + 60^2$$

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

where $\vec{n} = (0,0,1)$,

$(50,50,60) - (40,60,120) = (10,-10,-60)$

$$\vec{r} = \vec{P}_i + \lambda(\vec{P}_s - \vec{P}_i)$$

! for this calculation

$$= (50,50,60) + \lambda\big((40,60,120) - (50,50,60)\big)$$
$$= (50,50,60) + \lambda(-10,10,60)$$

Let $\lambda = 1$. $\vec{r}\big|_{\lambda=1} = (50,50,60) + (-10,10,60)$

$$= (40,60,120)$$

$$\vec{n}\cdot\vec{r} = (0,0,1)\cdot(40,60,120) = \cancel{0\times40} + \cancel{0\times60} + 1\times120 = 120$$

$$\frac{\vec{n}\cdot\vec{r}}{\|\vec{n}\|\|\vec{r}\|} = \frac{120}{1\cdot\sqrt{40^2+60^2+120^2}} = \frac{120}{\sqrt{40^2+60^2+120^2}}$$

Therefore

$$\vec{I}_{diff}(50,50,60) = (0.8,0,0)\,\frac{1}{10^2+10^2+60^2}\cdot\frac{120}{\sqrt{40^2+60^2+120^2}}$$

red: $$I_{diff,r} = \frac{0.8\times120}{(10^2+10^2+60^2)\times\left(\sqrt{40^2+60^2+120^2}\right)}$$

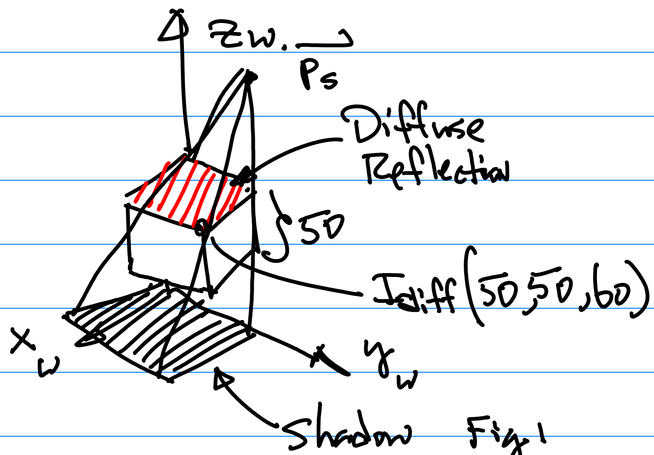


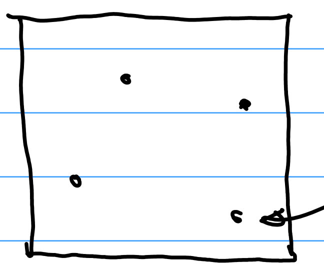
Fig 1

Note: The Computation is carried out in X_w-Y_w-Z_w.

You can compute diffuse reflection for the rest 3 vertices

$\vec{P}(50,0,60)$, $\vec{P}(0,50,60)$, $\vec{P}(0,0,60)$

Note: To plot these 4 point with the color, use Transformation Pipeline,



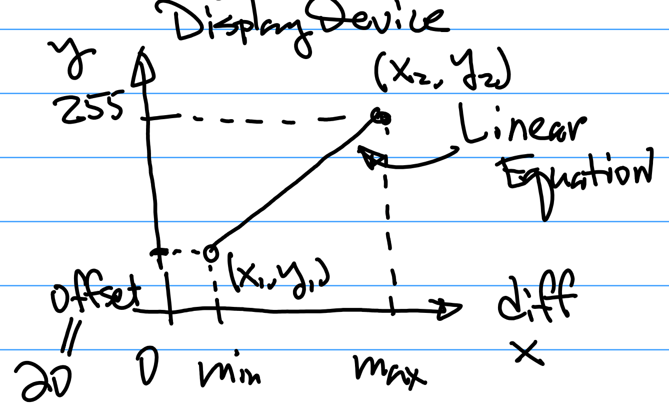
$\vec{P_i}''$ its diffuse reflection is Computed in X_w-Y_w-Z_w.

Fig.2

Their Locations are Computed by Transformation PipeLine !!!

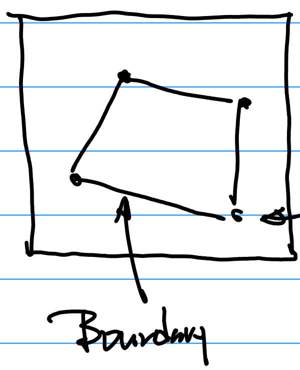
Important ! To make the Diff. Reflection Result Visible, use

Scaling as follows



$$\frac{x - x_2}{x_2 - x_1} = \frac{y - y_2}{y_2 - y_1} \rightarrow \text{find the Eqn.}$$

Example: Compute Boundary Color



$\vec{P_i}''$ its diffuse reflection is Computed in X_w-Y_w-Z_w.

Two Steps Process.

After Perspective Projection, we have to deal with Finite Resolution of the display device.

D.D.A

(Digital Differential Algorithm)

CMPE 240

2. Final Exam   Nov. 29 (Mon).

Group I Classes

Group I classes are those classes which meet M, W, F, MTW, MWR, MTWF, MWRF, MTWRF, MW, WF, MWF, MF, TW, WR, MT.

Regular Class Start Times	Final Examination Days	Final Examination Times
7:00 through 8:25 AM	Tuesday, December 14	7:15-9:30 AM
8:30 through 9:25 AM	Thursday, December 9	7:15-9:30 AM
9:30 through 10:25 AM	Monday, December 13	7:15-9:30 AM
10:30 through 11:25 AM	Wednesday, December 8	9:45 AM-12:00 PM
11:30 AM through 12:25 PM	Friday, December 10	9:45 AM-12:00 PM
12:30 through 1:25 PM	Tuesday, December 14	12:15-2:30 PM
1:30 through 2:25 PM	Thursday, December 9	12:15-2:30 PM
2:30 through 3:25 PM	Monday, December 13	12:15-2:30 PM
3:30 through 4:25 PM*	Wednesday, December 8	2:45-5:00 PM
4:30* through 5:25 PM*	Friday, December 10	2:45-5:00 PM

Due Dec 6 (1:30pm)   5a

1. Last Homework, take photos and submit them as one pdf document.
   a. Entire System (Host Laptop LPC1769); (one photo)
   b. LPC1769 w/ Display of Shadow Graphics (Trees). One photo.
   c. Screen Capture of NXP XPRESSO with Console Display of Name, ID, 3D Graphics. Submit to CANVAS.

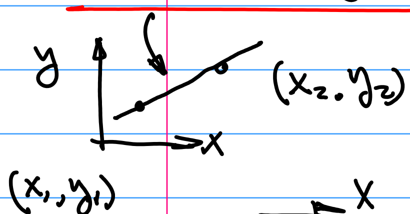
Need to use Prototype Board to Answer Questions.

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

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

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

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



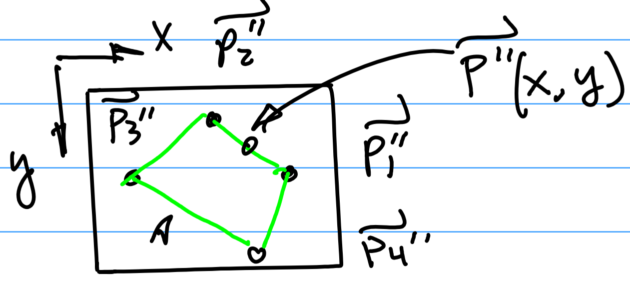


Fig. 1.

Note: 1° 4 vertices pts $\vec{P_1''}, \vec{P_2''}, \ldots, \vec{P_4''}$ have diffuse Reflection; 2° Extend the diffuse Reflection to Each Boundary Line (Green);

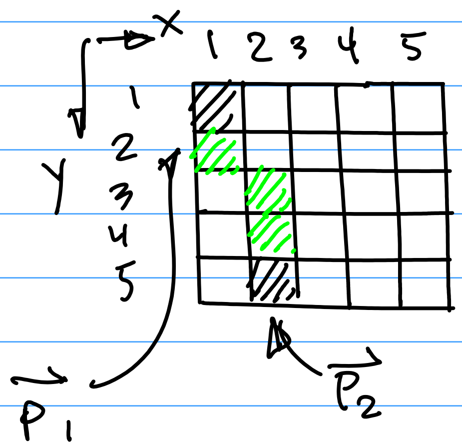
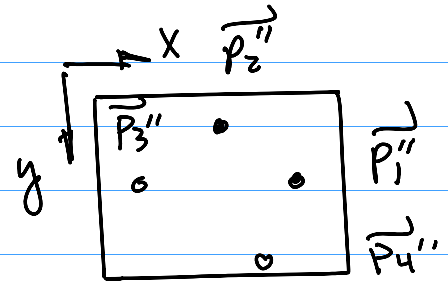
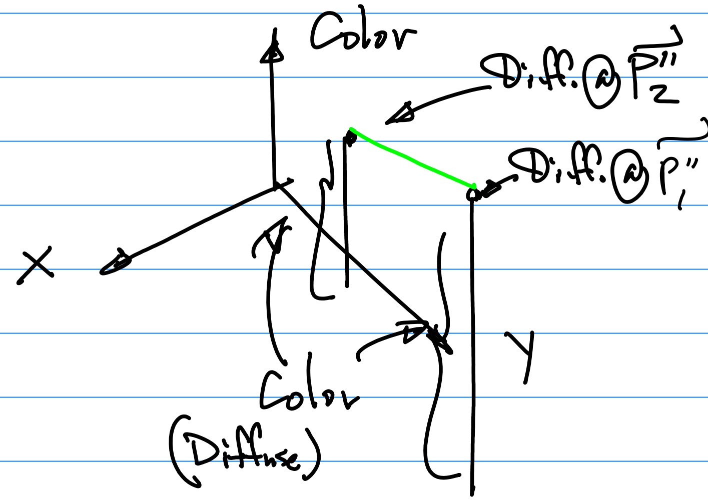


Fig. 2

"Plot" Diffuse Reflection in 3D Space



$\vec{P_1''}, \vec{P_2''}, \cdots, \vec{P_4''}$ Fig.3.



Note: Example for Interpolation. is to use Sin/Cosin Table

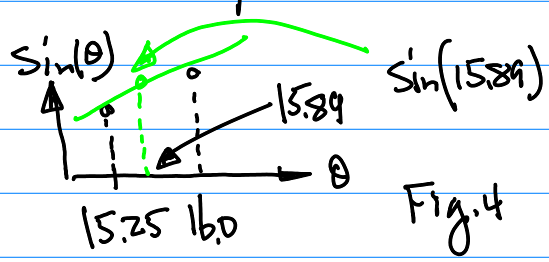


Fig.4

Note: Diffuse reflection on the green line in Fig.3 is a function of Both x, y. therefore we will treat interpolation Separately. One interpolation for x, one interpolation for y.

To visualize these 2 interpolations, we have the following illustration, 2021S-105b ~ Ref from github.

Project Fig.3 As function of one variable x, so, we have Fig.5

x — Independent variable

Diffuse Reflection — Function.

$$I_{diff} = ax+b \quad \ldots (2)$$

where a, b are determined by Eqn.(1b).

Similarly, in Fig.6. Diff. Reflection is a function of y.

$$I_{diff} = a'y+b' \quad \ldots (3)$$

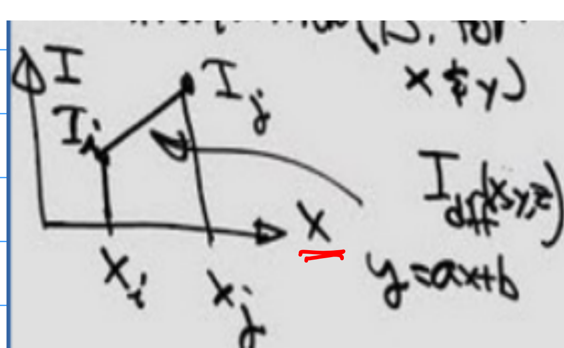


Fig.5

use eqn(1b) to find a', b'.

Then, Take Average.

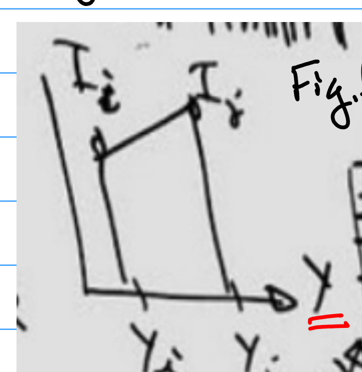


Fig.6

Example: Suppose, $I_{diff} = 0.78$ for red @ $\vec{P}_1''(x_1'', y_1'') = \vec{P}_1''(20,30)$

$I_{diff} = 0.5$ for red @ $\vec{P}_2''(x_2'', y_2'') = \vec{P}_2''(10,40)$ (Note: Arbitrary pt., Real value location, see Actual Calculation)

To find $\vec{I}_{diff}(15,35) = ?$

Step 1. Derive Interpolation formula for x Based Equation (1b)

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

Treat x as independent variable, y is diffuse Reflection Value from the given condition

$x_1 = 20$, $y_1 = I_{diff} = 0.78$

$x_2 = 10$, $y_2 = I_{diff} = 0.5$

Substitute x_1, y_1, x_2, y_2 into eqn (1b),

$$I_{diff} = \frac{0.78 - 0.5}{20 - 10} x - \frac{0.78 - 0.5}{20 - 10} \cdot 10 + 0.5$$

Let $x = 15$, find I_{diff}.

$$I_{diff,x} = \left. \frac{0.28}{10} x \right|_{x=15} - \frac{0.25}{10} \times 10 + 0.5$$

$$= 0.28 \times \frac{3}{2} - 0.25 + 0.5 = 0.14 \times 3 + 0.25$$

$$= 0.42 + 0.25 = 0.67$$

for y, we have

$I_{diff,y}$

$y_1 = 30$, $y_2 = 40$ (Indep)

Hence,

$I_{diff}(20,30) = 0.78$

$I_{diff}(10,40) = 0.5$

Therefore

$$I_{diff,y} = \frac{0.78 - 0.5}{30 - 40} \cdot y - \frac{0.78 - 0.5}{30 - 40} \cdot 40 + 0.5$$

$$= \left. \frac{0.28}{10} \cdot y - \frac{0.28}{10} \cdot 40 + 0.5 \right|_{y=35}$$

$$= \frac{0.28}{10} \times 35 - 0.28 \times 4 + 0.5$$

$$= 0.14 \times 7 - 1.04 + 0.5$$

$$= 0.91 + 0.5 - 1.04 = 1.41 - 1.04$$

$$= 0.37$$

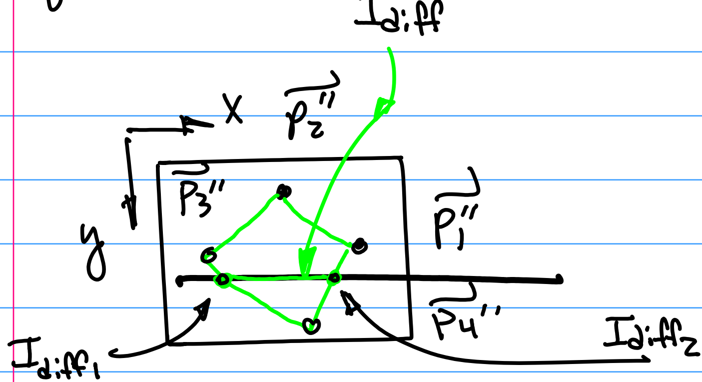
Finally, Average them.

$$I_{diff}(15,35) = \frac{1}{2}(I_{diff,x} + I_{diff,y})$$

$$= \frac{1}{2}(0.67 + 0.37)$$

Dec 1st (Wed)

Consider I_{diff} Computation of Interior Points.



$\vec{P_1''}, \vec{P_2''}, \cdots, \vec{P_4''}$ Fig 1.

Example: from the class github

2021S-105b-CMPE240-2021-05-17.pdf

P53

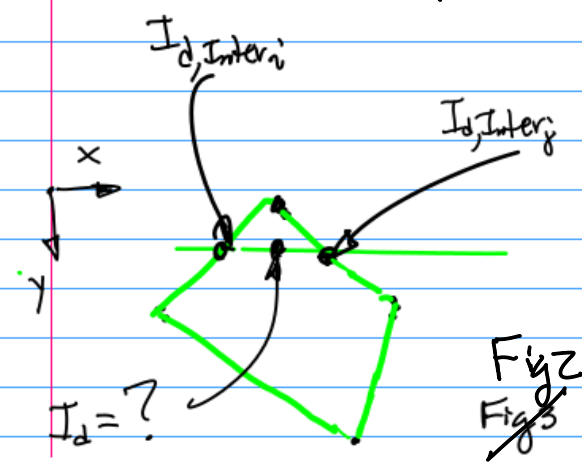


Fig 2

~~Fig 3~~

I_{diff_1}, I_{diff_2} Are Known,

Find I_{diff} By Linear interpolation

I_{diff} is a function of x given value y

Therefore, we have (from Eqn (16), pp. 61)

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

where y corresponds to diffuse reflection value, e.g.

$y_1 = I_{diff_1}, \quad y_2 = I_{diff_2}$

x_1 is the x value where I_{diff_1} is defined.

x_2 is the x value where I_{diff_2} is defined.

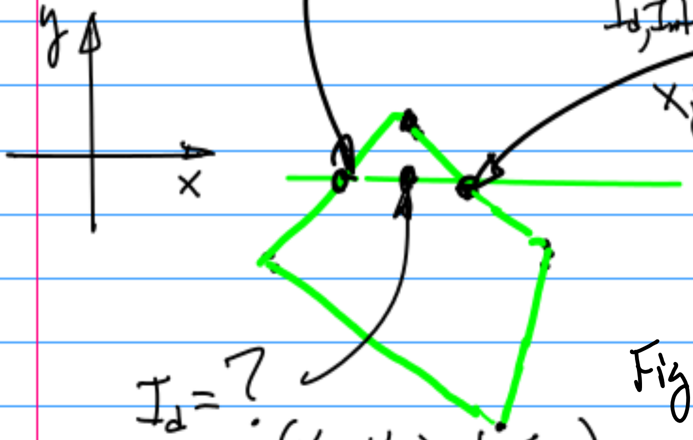
$I_{d,Interi}(x_i, y_i) = 80$, $x_i = 10$, $y_i = 10$

$I_{d,Interj}(x_j, y_j) = 90$ (closer to $\vec{P}_s$, pt. Light Source)

$x_j = 20$,

$y_j = 10$; Note: The variable x Contributes to the Change of Diffuse Refl.

Therefore, Linear Interpolation



$I_d = ?$

$(x_t, y_t) = (15, 10)$

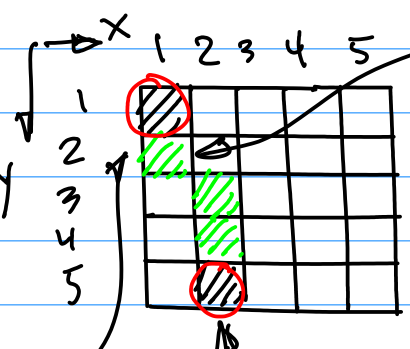
Fig.3

Therefore, Linear Interpolation with respect to X variable

$$I_d(15,10) = \frac{80+90}{2} = 85$$

Note: The point (Interior) is Any point Between the 1st Boundary point & the 2nd Boundary point. Use Eqn (1b), pp61.

Digital Differential Algorithm. DDA



$\vec{P}_1$ $\vec{P}_2$ Fig.2

See pp64.

Step 1. Compute Slop. $\vec{P}_i$ and $\vec{P}_{i+1}$

$$a = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \quad \ldots (1)$$

Check if $|a| < 1$

if yes

$$x_{k+1} = x_k + 1$$

$$y_{k+1} = y_k + a \quad \ldots (2)$$

$$y = ax + b \rightarrow y_k = ax_k + b$$

$$y_{k+1} = a(x_k + 1) + b$$

$$y_{k+1} = ax_k + a + b$$

$$= y_k + a$$

if $|a| > 1$
then, $y = ax + b$

$$\frac{1}{a}y = x + \frac{b}{a}$$

$$x = \frac{1}{a}y - \frac{b}{a} \quad \cdots (3)$$

$$x_{k+1} = x_k + \frac{1}{a} \quad \cdots (4)$$

Example: $\vec{P_i}(1,1)$, $\vec{P_{i+1}}(2,5)$
Sol: Compute slop

$$a = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} = \frac{5-1}{2-1} = 4$$

$|a| > 1$

Hence, use formula (4) to find Boundary point to link $\vec{P_i}$ to $\vec{P_{i+1}}$.

from Eqn (4), Let $y_k = 1$.

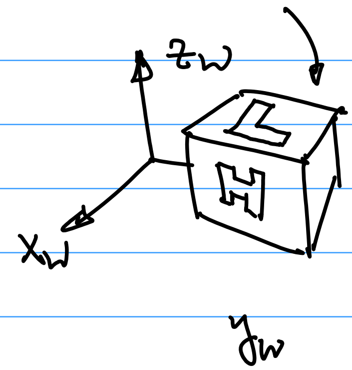
$$y_{k+1} = y_k + 1 = 1 + 1 = 2$$

$$x_{k+1} = x_k + \frac{1}{a}, \text{ from Eqn}(4)$$

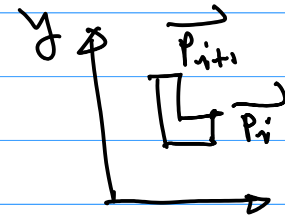
$$x_{k+1} = 1 + \frac{1}{a} = 1 + \frac{1}{4} = 1.25 \simeq 1$$

//

LINEAR Decoration Algorithm.



Define 2D Patterns



2D Pattern to Decorate 3D plane.

Step 1. Design your 2D Pattern

$\{\vec{P_i}(x_i, y_i) \mid i = 0, 1, 2, \cdots, K-1\}$, such as

$P_1(10, 10)$, $P_2(10, 15)$, ...

$x_1 = 10$, $y_1 = 10$, $x_2 = 10$, $y_2 = 15$

Step 2. Define 2D Pattern in 3D Space By Adding $z = 0$

$$\{\vec{P_i}(x_i, y_i)\} \xrightarrow{z=0} \{\vec{P_i'}(x_i', y_i', 0)\}$$

Such as

$P_1'(10, 10, 0)$, $P_2'(10, 15, 0)$, ...

Derive a formula,

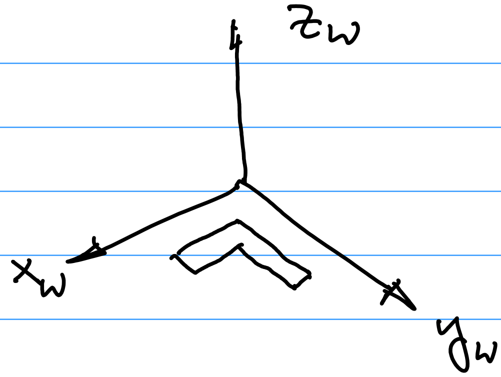
$$\begin{cases} x_i' = x_i \\ y_i' = y_i \\ z_i' = 0 \end{cases} \quad \ldots (3)$$

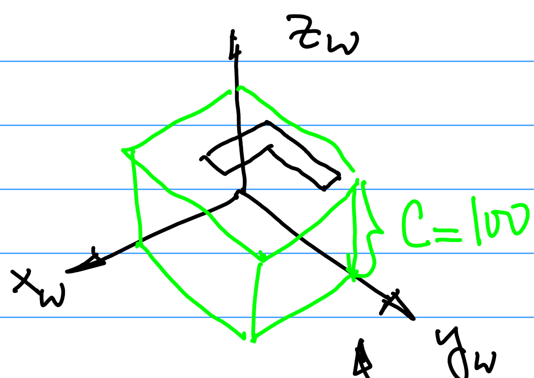
y_w-z_w plane, Right-Hand rule

y_w : Ind, z_w : Function

Therefore matching Ind → Ind, Function → Function

$$\begin{cases} y_i'(\text{Ind}) = x_i\,(\text{Ind}) \\ z_i'(\text{Func}) = y_i\,(\text{Func}) \\ x_i' = C \end{cases} \quad \ldots (4)$$





$$\begin{cases} x_i' = x_i \\ y_i' = y_i \\ z_i' = C \end{cases}$$