

Computer Graphics

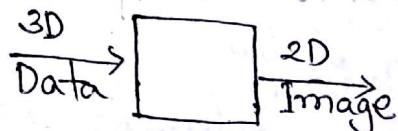
04.03.2014

(Haider Sir)

Lecture -1

Display Systems:

* Rendering Pipeline: Drawing



3D data \leftrightarrow 3D Model (Prototype of real world)

Simplest model \rightarrow Mathematical model

\rightarrow Geometric equation + variables which decide the shape

\rightarrow Controlling parameters x, y, z etc

Equation for sphere: $x^2 + y^2 = r^2$.

Real world mostly does not support such equations.

Random equation + complex mathematical model.

factors:

Where it would be displayed

How it would be displayed.

Environment

Surface রেশন, যত্ন নিষ্ঠা রেশন

How is the object \rightarrow glittering/rough.

Shading [Light/shape interaction + brightness difference]

Visibility etc.

animation etc.

Algo. 80.19

point-to-point Communication
(one to one)

Main target \rightarrow Efficiency / Speed.

Main PC \rightarrow 32 bit

1-bit

Display \rightarrow 128 bit [4 কুণ্ড যোকি]

Graphical so that general need is fulfilled. CPU does mathematical process. Graphical make it real time [Naturally স্বচ্ছতে যে time লাগে, computer এ দিতেও যেহেতু time লাগে]

Flicker free.



Second এ 20 বার flicker করাতে human eye gap থাকতে পারেনা (due to saccades)

Recent research shows 50 Hz max.

For example, for a image,

60 \times 1024 \times 768 \times 32 bit/sec লাগে
60 frames \times pixels Depth

for = $60 \times 1024 \times 768 \times 32 \times 32$ bit/sec

Output \rightarrow Integer coordinate system.

Floating point হাত ধরে নেয়ে আবেদন করে এবং উন্নের অংশে এর কাজ।

Types of display system

Display system

Vector Graphics

Raster

(Bit mapped)

bit mapping + soft rendering [quadtree] problem

o. utilities
o. optimization

Vector → Consisting of paths / lines, pdf
Raster → Consisting of dots / pixels, ps, eps

Vector ॲ अग्नि gap नेहे | smooth line draw करावे।

Example - Oscilloscope

Continuous lines.

Vector type 1986 मध्ये प्रयोग आहे। Research इथ्यात 50/60 ए.

At present mainly raster graphics आहे।

Raster graphics → Array of 2D pixels.

(Reality ते 1D बुलावे नाही तरी ते 2D वैष्णवी ही)

→ Buffer शायद.

→ 60Hz frame याने एक व्याप्त 60 pixel display

कऱ्यात पाठी monitor

→ ऊरु pixel नियंत्री buffer

→ frame buffer is adjustable.

frame ↔ no of bits.

Video Controller → pre defined speed ए frame buffer

→ read करावे monitor ते flush / send

log इले digital then digital आणि ana-

log इले convert करावे show करावे।

→ No connection with CPU.

→ Locally controlled by computer

→ हल्ले इले display running शायद।

→ continuous counter टाळत थायदे। Read

→ send करावे।

Dual access memory ↔ frame memory.

→ shading information to

→ R/W for CPU অর্থাৎ যখন read করে CPU
write করতে পারে না। But display তে GPU
করে যদ্বারা read/write করতে পারে।

Raster এ image বীরুত রয়ে frame buffer.

Vector এ each function pre-defined function.

Each frame is a set of functions / commands.

Repeatedly, function execute করে।

No. of lines হত যাতে, repetitiveness ও slow হয়।

Disadvantages of vector

→ Frame rate data dependent. Data বোঝে slow
হয়ে যায়।

→ Efficiently filling করতে পারে না। Boundary / line
draw করতে পারে। Filling slow isoz thousands
of line লাগে। Dot মুছে যাও।

Raster -এ পাশাপাশি dot আবলে human eye দ্বারা পারে না,
continuous, আলো পারে না। So almost real object /
filled object তৈরি করা যায়। That's why raster is
more used in today's display systems.

Advantages of vector

→ scalable, size modification very easy. এই
কারণে hardware vector system use করা হলেও
অনেক software system এ হয়।

scale dependent so vector graphics is a set
of commands.

PostScript (PS) language → set of commands नियंत्रण भाषा.

Raster graphics एवं नियंत्रण वेक्टर ग्राफिक्स एवं huge pixel भित्र show करते PS printeruring 80's.

All displays now use ps idea.

True type font → PS font.

Control point setup command द्वारा अंकित किया हुया.

Shape change हुए तथा scale change करना आसानी से easily by changing co-ordinates.

Scan conversion algorithm → algorithm used in Raster

Output → 2D integer co-ordinate system.

- allows fraction point निश्चयीकृत बिंदु.

- GPU to run on different systems.

Depth buffer algo / visible multiple algo.

→ Pixel level algo → scan conversion.

OpenGL one type of API.

Graphics Language

By using OpenGL → line draw

→ Polygon draw

→ Multiple polygon एवं उन particular अंकित visible.

→ Shading रेग्न पर्ट एवं करना.

OpenGL \rightarrow Pixel drawing \leftarrow OpenGL (Q) frame buffer

Reference book:

Computer Graphics by Foley et al.

- Foley et al. Computer Graphics II A

PS \rightarrow Post script \rightarrow Raster

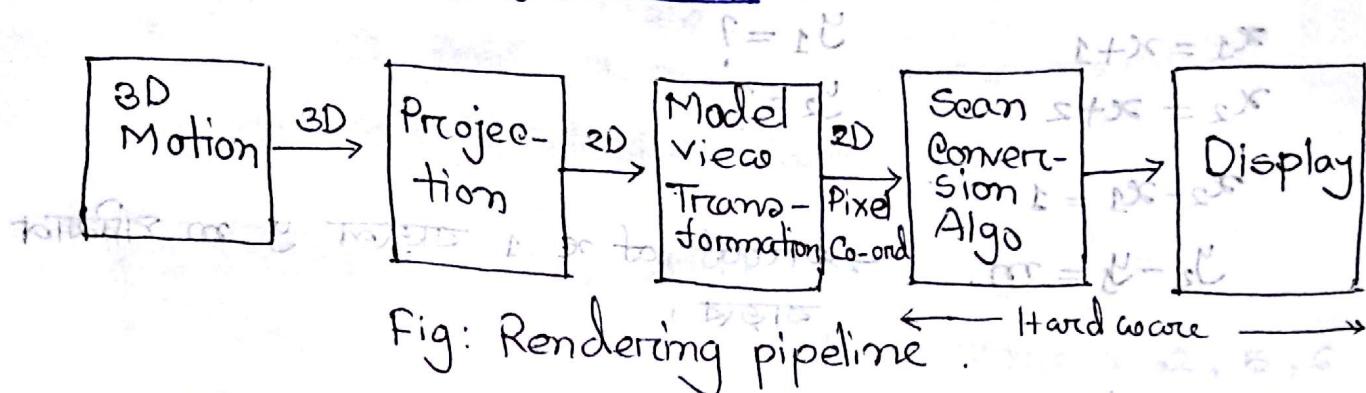
EPS

- vector, raster ~~are~~ diff.
- HSV, HLS ~~are~~ diff.
- Phong and Gouraud diff.
- co-ordinate, geometric এবং তুলনামূলক study
- prove that, hue ~~is~~ brightness independent of color. mathematically. same color এবং RGB values diff নিয়ে calculate করতে হবে।
- Rigid body motion
- Affine motion
- Math (3 marks) \rightarrow in at least 5 questions.

Lecture - 02

05.04.2014

Scan Conversion Algorithm:

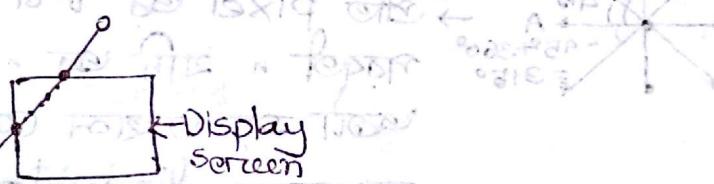


Line drawing:

Two aspects

(i) 2 endpoint and draw line.

(ii)



Display এর কিন্তু যেকোনো point আছে তাকে রেখার
consider করতে হবে। efficiency গড়ে।

$x = x_0, y = y_0$ \rightarrow (x_0, y_0) to (x_1, y_1) not ask

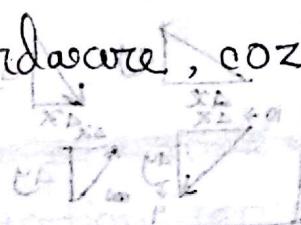
for ($x = x_0; x \leq x_1; x++$) { \leftarrow Hardware implemented }

$x++;$ // x এর পাশে y এর পরিবর্তন করবে

$y = y_0;$ // m এর value normally fractional হয়,

draw pixel(x, y); \rightarrow Hardware implemented

} not possible by hardware, coz floating point.



Incremental Algorithm

$$y = mx + c$$

$$x_1 = x + 1$$

$$y_1 = ?$$

$$x_2 = x + 2$$

$$y_2 = ?$$

polysid

$$x_2 - x_1 = 1$$

$$y_2 - y_1 = m$$

proof of $x \in \text{पास्तल } y \in \text{पास्तल}$

याहुए !

संस्कृत प्र०

Midpoint Line algorithm: Bresenham's Algo.

Floating point এবং অংশকে eliminate করুন।

and work base triangle ২ (i)

প্রতি pixel এর টেক্স প্রতিবন্ধী থাকে।
পথবর্ণ n যদি এবং n প্রতিবন্ধীয় যোগৈব
একটা জ্যোতিষ্ঠান ঘো একটা continuous line.
না হল discontinuous.

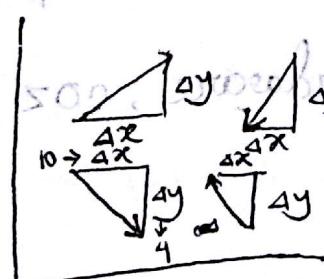
$45^\circ/2 = 22.5^\circ$ এবং যোগ হলে point B, বর্তমান point A.

Geometric point of view থাকে এ বিশ্লেষণ

Vector $\vec{v} = (dx, dy)$ এবং $(dx, dy) = 8dB = B \cdot e^{j\theta} = k$

লক্ষণস্বরূপ $\vec{v}^T(\vec{v} + \vec{v}) = \vec{v}^T \vec{v} \Rightarrow x^2 + y^2 = x^2 + 2xy + y^2$ সহ

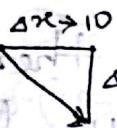
প্রতি উভয় দিকে মাঝে পার্শ্ব ক্ষেত্র থাকে।
অন্তর্ভুক্ত ক্ষেত্র প্রতি উভয় দিকে মাঝে পার্শ্ব ক্ষেত্র।



independent variable ক্ষেত্র
পরিবেচন হয়,

dependent variable ক্ষেত্র
পরিবেচন হয়,

থার্ড প্রবাচন্য ক্ষেত্র change হয়
আইন লুপ কন্ট্রোলের।



$\Delta x > 10$ এবং $\Delta y > 4$ বাজে 10 বাব
এটা ক্ষেত্রে y ক্ষেত্রে 4 নির্দিষ্ট করা হচ্ছে।

So, loop controller এক্ষেত্রে x .

for ($y = y_0 ; y \leq y ; y++$)

$$x + \frac{1}{m};$$

// For zone 1, 2, 5, 6

draw pixel (x, y)

Hence loop controller is y .

}

৪'th zone এর জন্য আলাদা পদ্ধতি চিহ্ন রয়েছে।

Line draw করার আগে কোন zone determine করতে হবে,

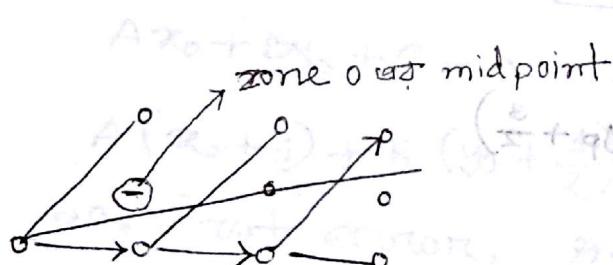
$x_1 > x_0, y_1 > y_0 \rightarrow$ zone 1

$x_1 < x_0, y_1 < y_0 \rightarrow$ zone 4.

$x_1 < x_0, y_1 > y_0 \rightarrow$ zone 3.

বই-ত একটি zone এর জন্য দেওয়া আছে। ৪'th ক্ষেত্রে

Equal হলে either \leq or \geq .

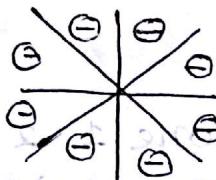


$$y = mx + c$$

$$\Rightarrow -y + mx + c = 0.$$

যদি x এবং y ক্ষেত্রে value নির্দিষ্ট করা হচ্ছে এবং এই value exactly on the line then equation \Rightarrow value 0.

x, y is not exactly in the line, then it determines which zone the distance is.



→ midpoints for 8 zone.

o → zone 1 to midpoint.

$$y = mx + c \quad \text{①} \rightarrow \text{implicit form}$$

$$mx - y + c = 0 \quad \text{②} \rightarrow \text{explicit form (क्षेत्रों का)}$$

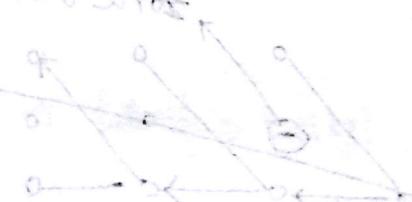
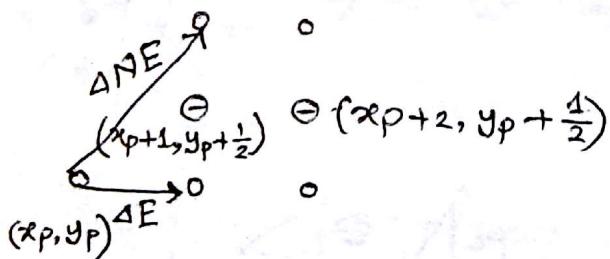
$$\Delta y - \Delta x y + \Delta x c = 0 \rightarrow \text{क्षेत्रों का}, m = \frac{\Delta y}{\Delta x}$$

$$Ax + By + C = 0 \quad \text{③}$$

Implicit form

$$A = \Delta Y, B = -\Delta X$$

$$\Theta(x_p + z, y_p + \frac{z}{2})$$



$$y + 25m = y$$

$$0 = 0 + 25m + y - y$$

Have solved for m or calculated y in this
o solve for m we need to calculate m .

eqn (1) \Rightarrow

$$f(x_p, y_p) = Ax_p + By_p + C \quad \downarrow \text{error}$$

একজুড়ু স্থান থেকে change হয়েছে (error) অথবা স্থান পরিবর্তন করতে হলে আগের যাদের কাছে থাকে।

initially একটি point এর error এবং deviation জানলে just এর value রে ~~new~~ old value এর আগে add করে নিলে new error পাওয়া যাবে।

deviation of,

$$\Delta F = f(x_{p+2}, y_p + \frac{1}{2}) - f(x_{p+1}, y_p + \frac{1}{2}).$$

→ change in error towards horizontal movement

$$\Delta NE = f(x_{p+2}, y_p + \frac{3}{2}) - f(x_{p+1}, y_p + \frac{1}{2}).$$

→ change in error towards vertical diagonal movement.

$$Ax_0 + By_0 + C = 0$$

$$A(x_0 + 1) + B(y_0 + \frac{1}{2}) + C = ?$$

so, first error, $A + \frac{B}{2} \rightarrow d_{\text{init}} = \frac{\Delta X}{2} - \frac{\Delta Y}{2}$.

$$\text{So, } \Delta F = A(x_{p+2}) + B(y_p + \frac{1}{2}) - A(x_{p+1}) - B(y_p + \frac{1}{2})$$
$$\therefore \Delta F = A = \Delta Y.$$

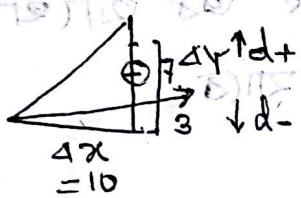
[2nd error].

$$\Delta E = \Delta Y.$$

$$\Delta NE = \Delta Y - \Delta X$$

initial error, $d_{init} = A + \frac{B}{2}$.

$$d_{init} = \Delta Y - \frac{\Delta X}{2} \quad [\text{2nd error}]$$



$$d = 3 - \frac{10}{2} = -2$$

$$\text{current } d_{init} = d_{init} + (\frac{b}{2} + qk) t = 3D$$

[for horizontal]

$$\text{Lateral error} = d_{init} + \Delta NE \quad [\text{for diagonal}]$$

$$\Delta NE = \Delta Y - \Delta X$$

$$(\frac{b}{2} + qk) t - (\frac{b}{2} + qk, start) t = 3AD$$

with lagging effect starts in opposite direction

$$0 = b + qS + qkA$$

$$qS = 0 + (\frac{b}{2} + qk) S + (b + qk) A$$

$$\frac{qS}{2} - qkA \leftarrow \frac{qS}{2} + qk \quad \text{starts front, 0S}$$

Diagonals keep

$$(\frac{b}{2} + qk) S - (\frac{b}{2} + qk) A - (\frac{b}{2} + qk) S + (b + qk) A = 3AD, 0E$$

$$YD = A = 3E$$

10.03.14

Scan conversion Algorithm:

Line drawing algorithm

$$Ax + By + C = 0$$

hence, $A = dy$, $B = -dx$

For zone 0 →

$$d = dy - \frac{dx}{2}$$

$$\Delta E = dy$$

$$\Delta NE = dy - dx$$

ক্ষেত্র ০। only negative/positive
ধৰণ কোটি যথে। so,
 $d = (dy - \frac{dx}{2}) * 2$, $\Delta E = dy * 2$,
 $\Delta NE = (dy - dx) * 2$

All slope < 0.5 এবং d negative. only ১ বাতুর

য রাখবে না।

For zone 0

void drawline (int x_0 , int y_0 , int x_1 , int y_1)

{
int $d, dE, dNE, x, y;$

$$d = (dy - \frac{dx}{2}) * 2;$$

$$dE = dy * 2;$$

$$dNE = (dy - dx) * 2,$$

$$x = x_0; y = y_0;$$

drawPixel (x, y);

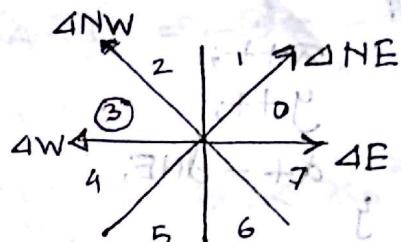
while ($x <= x_1$)

{
if ($d < 0$)

// ΔE

$x++;$

$d += dE;$



(x, y) box work

$$\text{slope} = 1 \\ \therefore dy = dx.$$

$$\text{slope} = \frac{5}{2} \\ \therefore dy = \frac{dx}{2}$$

$$|\Delta| < 1/2 | \quad (i)$$

$$|\Delta| > x \quad (ii)$$

// ২ দ্বারা multiply কৰা হয়

যাতে fractional value

না থাকে।

rough Drawing Exam

$$86 * 2 = 256 \quad dE = 50 \\ x = 6.3125y \quad dy = 25 \\ t = -(\frac{1}{2})t = dx = 80 \\ t < 7t = dinit = 25 - 40 = -15 \\ dNE = (25 - 80) = -55 \\ -15 - 55 = -70 \quad \frac{-290}{345} \\ -180 - 55 = \frac{235}{290} A = -125 - 55 = -180 \quad \frac{55}{345}$$

else

{

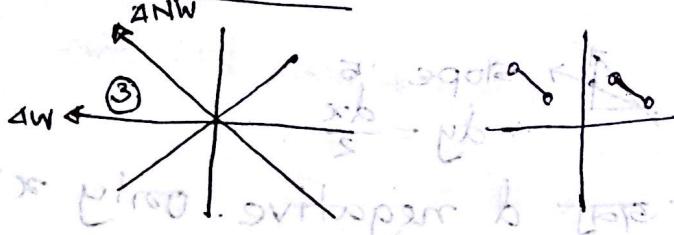
$x++;$
 $y++;$
 $d += d_{NE};$
 $y += 1;$
 $tracePixel(x, y);$

}

$d = d_{SW} + d_{SE};$
 $y -= 1;$
 $x -= 1;$
 $tracePixel(x, y);$

}

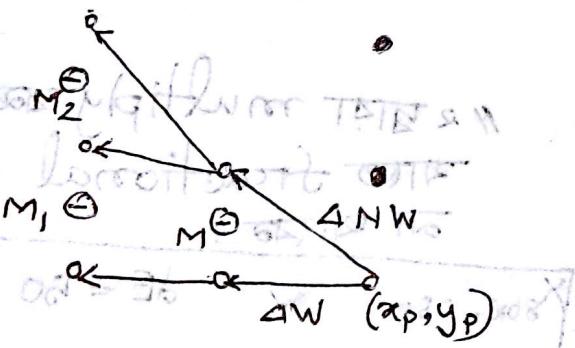
For zone 3



$$(i) |\Delta x| > |\Delta y|$$

$$(ii) \Delta y > y_p - \text{true}$$

$$(iii) x < x_p$$



$$\Delta w = f(M_1) - f(M)$$

$$\Delta w = f(M_2) - f(M)$$

$$\Delta w = A(x_p - \frac{1}{2}) + B(y_p + \frac{1}{2}) + C - A(x_p - \frac{1}{2}) - B(y_p + \frac{1}{2})$$
$$= A \Delta x + B \Delta y$$

$$\boxed{\Delta w = -\Delta y}$$

$$0 = D + PB + xA$$

$$x b - D - PB = A \rightarrow \text{correct}$$

$$-PD - PB = 5 \cos 12^\circ$$

$$\frac{x b}{2} - Pb = b$$

$$Pb = 3A$$

$$x b - Pb - 3A =$$

$$x b - Pb - 3A =$$

$$x b - Pb - 3A =$$

$$x b - Pb - 3A =$$

$$x b - Pb - 3A =$$

$$x b - Pb - 3A =$$

$$171.75 \text{ NTS}$$

$$0 \text{ NTS}$$

$$M = \left(x_p - \frac{1}{2}, y_p + \frac{1}{2} \right) = b$$

$$M_1 = \left(x_p - \frac{1}{2}, y_p + \frac{1}{2} \right) = b$$

$$M_2 = \left(x_p - \frac{3}{2}, y_p + \frac{3}{2} \right) = b$$

$$(C \approx) 1.5 \times 10^{-5} m^2$$

$$(B \approx) 3 \times 10^{-5} m^2$$

$$(A \approx) 1 \times 10^{-5} m^2$$

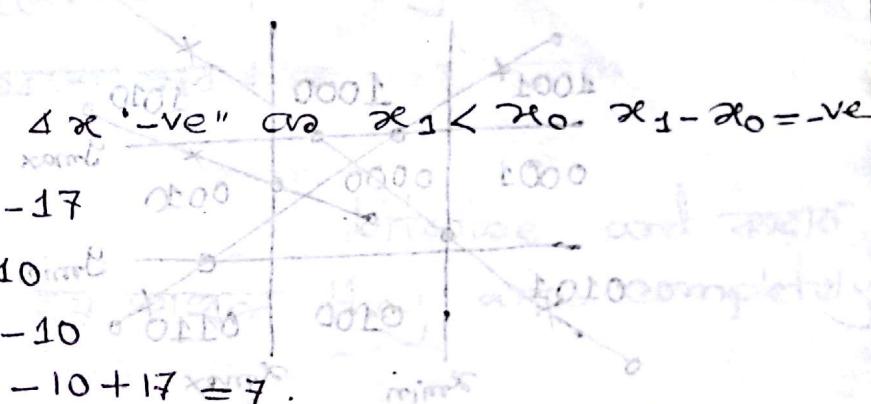
$$+ b$$

$$\Delta NW = -dy - dx \cdot A(x_p - 2) + B(y_p + \frac{3}{2}) + C - A(x_p - 1) - B(y_p + \frac{1}{2}) - C$$

$$= A(-1) + B(\frac{3}{2} - \frac{1}{2}) = -A + B = -dx - dy - da$$

$$\therefore \boxed{\Delta NW = -dy - dx}$$

$(-dy - dx)$ positive coz



(x_1, y_1)
 $(3, 5)$
 \downarrow
 (x_0, y_0)
 $(20, -5)$

$$\begin{aligned}\Delta x &= -17 \\ \Delta y &= 10 \\ 4E &= -10 \\ \Delta NE &= -10 + 17 = 7\end{aligned}$$

$(x_0 - 1), (y_0 + \frac{1}{2}) \rightarrow 1st \text{ mid point}$

$$A(x_0 - 1) + B(y_0 + \frac{1}{2}) + C = Ax_0 - By_0 + C$$

$$\Rightarrow -A + \frac{B}{2}$$

$$\Rightarrow \boxed{-dy - \frac{dx}{2}} \rightarrow d_{initial} \text{ (initial)}$$

For zone 3

$$d = -(dy + \frac{dx}{2}) * 2$$

$$\Delta W = -dy * 2$$

$$\Delta NE = -(dy + dx) * 2$$

Assignment on 11.04.14

Drawline for 8 zones.

Zone determine x_0, x_1 , only endpoints are given. Calculate midpoints and slope.

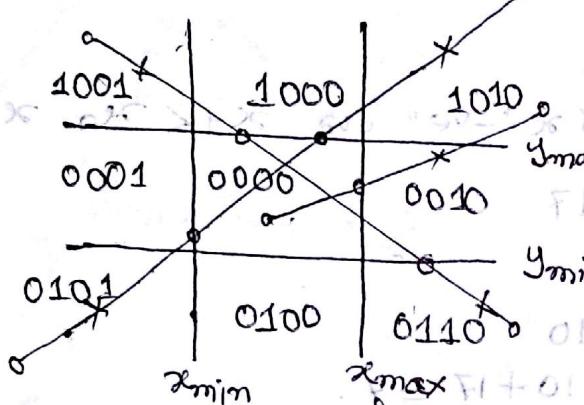
The ZF on infinite line

Lecture - 4

11.03.14

Scan conversion algorithm:

Line clipping algorithm



$$D = \left(\frac{b}{a} + q\right) A$$

$$\left(\frac{b}{a} + q\right) B + (1-q) A =$$

$$x_b - y_b - = WAD$$

$$500 \text{ clipping } (x_b - y_b -)$$

$$\begin{cases} (B/A) \\ (A/B) \end{cases}$$

$$\begin{cases} (B/A) \\ (A/B) \end{cases}$$

Draw করা আগেই Decision নেওয়া কার্যত visible
হবে।

Smart solution / efficiency বাঢ়ানোর জন্য algorithm
use করি।

general theory - if $x_{min} < y_{pixel} < x_{max}$
 $y_{min} < y_{pixel} < y_{max}$

Cohen - Sutherland Line clipping algo:

each end point is converted into 4-bit
binary value. অবস্থা হ'ল endpoint এর অবস্থানের
logical operation করে decision নিতে পারে Line কে কোথা হার

Most
probably
not in
Book.
Check
Net.

1st	bit এর অন্তর্যামীতি $\rightarrow x + (y + b) \rightarrow SNA$
2nd	bit এর অন্তর্যামীতি $\rightarrow y + (x + b) \rightarrow PNO$ ফর্মেটে $\rightarrow 1 \quad n \quad 0$
3rd	bit এর অন্তর্যামীতি $\rightarrow x + (y + b) \rightarrow PNO$ ফর্মেটে $\rightarrow 1 \quad n \quad 0$
4th	bit এর অন্তর্যামীতি $\rightarrow y + (x + b) \rightarrow PNO$ ফর্মেটে $\rightarrow 1 \quad n \quad 0$

3 decisions

(i) If endpoint of the code $\frac{region}{P-E}$ is zero then completely accepted.

Full line draw করতে হবে bit wise চান্দা

(ii)

পক্ষে যদি non zero হয় তাহলে they are completely rejected.

(iii)

যদি zero হলে তাকা partially accepted. অর্থাৎ যেগুন অংক draw হবে, কোন অংশ হবে না,

এই স্বাক্ষর আলোচনা proposal Cohen-Sutherland

$$P(+)=P_0 + (P_1 - P_0) \rightarrow \text{parametric equation of straight}$$

$$x = x_0 + (x_1 - x_0)$$

$$y = y_0 + (y_1 - y_0)$$

Geometric equation এর return value control করার
x, y, z এর variable.

Parametric eqn

+ এর return হয় x, y, z.

+ এর value change হলে return value change

হয়ে যাব। best adv decision ফিল নাই। geometric সাবে না ও infinite line দেয়।

$$x = x_0 + \frac{y - y_0}{y_1 - y_0} (x_1 - x_0) \rightarrow \text{equality not eqn. 8}$$

$$y = y_0 + \frac{x - x_0}{x_1 - x_0} (y_1 - y_0) \rightarrow x \text{ জানলে } y \text{ জানা যাবে।}$$

যাদের 1st width 1 bits ভাবত end point এর

নতুন $y = y_{\max}$

$$\begin{array}{ccccccc} & n & & 2nd & & 1 & \\ & " & & " & & " & \\ & " & \Rightarrow y = y_{\min} & & & & \end{array}$$

Now x, y পাওয়ার নথ আবাব proposal 3rd check
দিত হয়।

Efficiency কর নির্দ (আরে এই algo করে আজে আ।)

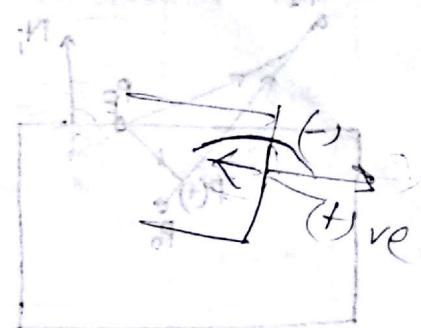
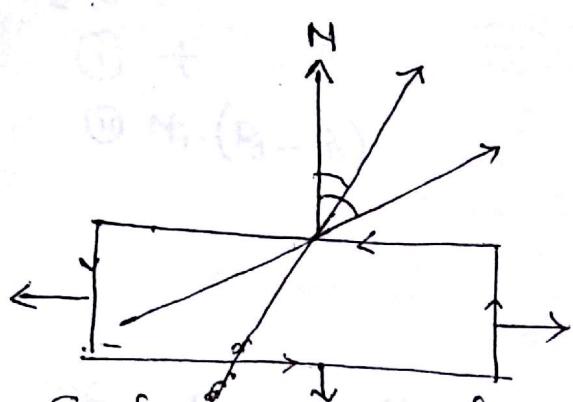
কারণ এ বাব করবার পথে আবাব check করার
প্রয়োজন হয়।

Normally maximum যতজুলা 1 হাঁকে at max
তত্ত্বালো step/check সতে final window পাওয়া
যায়। For example, 1010 এর জন্য 2 টি step লাগে,

Cutwo-Beck

যদি এর process করার পথে কোন কোন 8টি intersecting point বাইডে থাকে তাহলে তা solve
করা যায় এই algo দ্বারা।

Surface Normal (সৃষ্টি লম্ব)



Surface normal এর ধারণা
Dot product negative হলে line enters করবে

" positive " $= +1$ বেঁক হয়ে যাচ্ছে।

~~ক্ষেত্র পাই~~

Surface Normal হাব কিউ থাকে গহন্ত্রে দিকে
ইনে একটি point তান দিকে থাকলে positive and
অন্য দিকে হল -ve যদি তান দিকে কিউ থাকে,

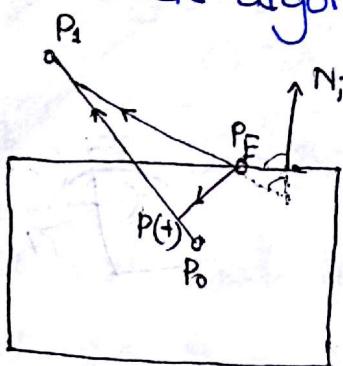
একটি point দেওয়া আছে $\theta = (1 - (1 - 1) + +1) / 4$

surface normal always কিউ থাকে গহন্ত্রে দিকে
surface এর ধারণা কোণ করে $\theta = (1 - 1) / 4$ +ve হল যাহায় -ve হল

$$\textcircled{iii} \rightarrow \frac{(1 - 1) / 4}{(1 - 1) / 4} = +1$$

the dot of all কাজের কাজের নীচে in

Cyrus-Beck algorithm for line drawing:



$$P(+)=P_0+t(P_1-P_0) \quad \text{--- (1)}$$

$t=0$ হলে, $P_+=P_0$ তাই লাইনের সমৰ্থন বিপরীত করা হয়।
 $t=1$ হলে, $P=P_1$.

$P(+)$ \rightarrow point on the line.

P_E \rightarrow " " " edge.

$N_i(P(+)-P_E)=0$, when $P(+)$ is exactly on the edge. --- (II)

$$N_i(P_0+t(P_1-P_0)-P_E)=0$$

$$-N_i(P_E-P_0)+N_i t(P_1-P_0)=0$$

$$\Rightarrow t = \frac{N_i(P_E-P_0)}{N_i(P_1-P_0)}$$

$$\therefore t = \frac{N_i(P_0-P_E)}{-N_i(P_1-P_0)} \quad \text{--- (III)}$$

N_i এর বক্টর হ্যাঁ অথবা parallel to x axis/ \perp

" " " y " "

Lecture - 6

Algorithm was दर्शाएँ आजे रोटी जिनिझ find out कराते हो।

① +

$$\frac{x_{min} - \alpha_L}{\alpha_L - \beta_L}$$

(x_{min}, ∞)

(t, 0)

q(t)

② N; (P₁ - P₀) .

$$\frac{x_{min} - \alpha_L}{\alpha_L - \beta_L} -$$

(min L)

(L, 0)

root(L)

$$\frac{x_{max} - \alpha_L}{\alpha_L - \beta_L} -$$

(L, max)

(0, t)

t(L)

$$\frac{x_{max} - \alpha_L}{\alpha_L - \beta_L} -$$

(L, max)

(0, t)

t(L)

$$\frac{x_{max} - \alpha_L}{\alpha_L - \beta_L} -$$

(L, max)

(0, t)

t(L)

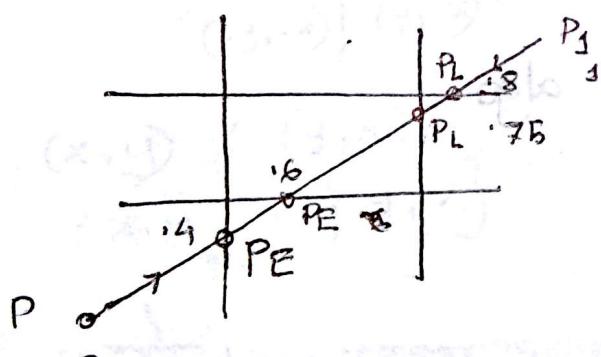
PE → Potentially entering, PL → potentially leaving.

यदि entering इय तात्काल intersecting अस वाई +_E
" " leaving " " intersecting अस वाई +_L.

यात्रा Line अस जवा +_E and +_L देव रुप्य हय,

Initially, +_E = 0 and +_L = 1 की हय।

यात्रा new +_E > old +_E then update +_E.



प्र॒ग्गिः ०.१५ x ३ र॒प्ताः ०.५

(c₁) = 9 ; (0, 0, 0), 9

P(c₁) f(c₁) P(c₂) f(c₂) = (9, 9)

यात्रा for example यात्रा ० प्र॒प्त वाई ० वाई ०

यात्रा ० लाभ अस असप्त वाई ० वाई ०

यात्रा यात्रा यात्रा यात्रा यात्रा यात्रा यात्रा यात्रा

यात्रा ० यात्रा ०

<u>Edge</u>	<u>N_i</u>	<u>P_E</u>	<u>t</u>
Top	(0, 1)	(x, y_{max})	$\frac{y_0 - y_{max}}{y_1 - y_0}$
Bottom	(0, -1)	(x, y_{min})	$-\frac{y_0 - y_{min}}{y_1 - y_0}$
Right	(1, 0)	(x_{max}, y)	$-\frac{x_0 - x_{max}}{x_1 - x_0}$
Left	(-1, 0)	(x_{min}, y)	$-\frac{x_0 - x_{min}}{x_1 - x_0}$

// underline & marked by ~~cont~~ ~~cont~~ ~~cont~~ ~~cont~~ ~~cont~~ ~~cont~~ calculation

9.

if $y_1 = y_0$ or $x_0 = x_1$ then $N(P_1 - P_0) = 0$.

coz parallel to axis ~~एवं याएँ~~

$N(P_1 - P_0) \rightarrow (+) \rightarrow$ Leaving

$\rightarrow (-) \rightarrow$ Entering

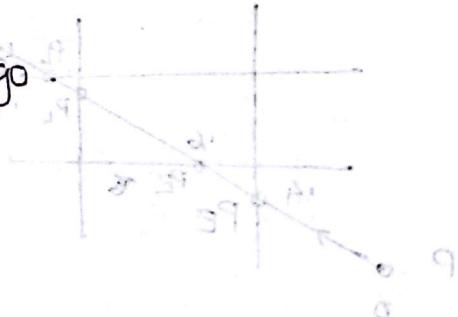
$\rightarrow 0 \rightarrow$ Discard.

2nd algo exp: Clipping algo.

$$P_0(0, 0, 0); P = (1, 50, 10)$$

$$N(P_1 - P_0) = \sqrt{(1-0)^2 + (50-0)^2 + (100-0)^2}$$

$$N \cdot (P_1 - P_0)$$



Lecture- 6

18.03.2014

Scan Conversion Algorithm:

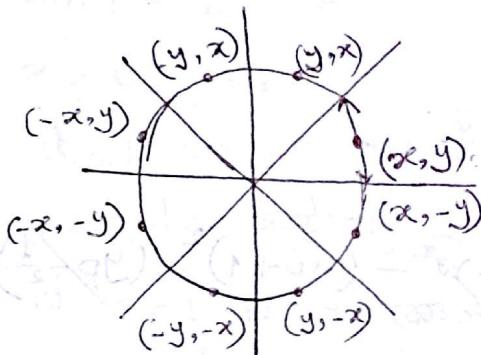
Circle drawing.

```
for (theta = 0; theta < 360.0; theta++)
```

```
    drawPixel(r cos theta, r sin theta);
```

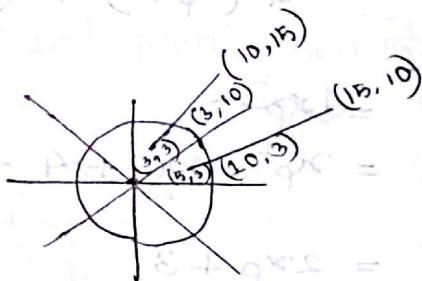
// complexity is very high.

Midpoint Circle drawing algorithm



$$(x, y) \neq (y, x)$$

$$\{x, y\} = \{y, x\}$$

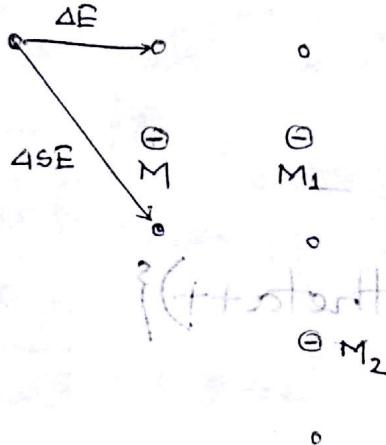


একটি Line দেওয়া আছে। এই Line এর যাপনক অন্য zone এ অবস্থিত
হলে for example zone 0 (থেকে zone 1 এ অঁকত হলে x_0, y_0 কে
 x_0, x_1 & y_0, y_1 কে y_1, x_1 এ swap করে) This way 8 separate
algo's are not needed for 8 zones. Pass $(x_0, y_0, 1)$ \rightarrow zone no.

$x=0, y=r \rightarrow$ start point

$x=y \rightarrow$ end

Disadvantage \rightarrow Always circle এর উপর নির্ভর হাব।

Force zone 1

$$x^2 + y^2 = r^2 \quad \text{①}$$

$$x^2 + y^2 - r^2 = 0 \quad \text{②}$$

motorway road

proposed scheme

$(f + \text{start}) ; 0.005 > \text{start} = \text{start}$) not
(center, 0.005) by 50m.

$$\Delta F \approx f(M)$$

$$M = (x_p + 1, y_p - \frac{1}{2})$$

$$M_1 = (x_p + 2, y_p - \frac{1}{2})$$

$$M_2 = (x_p + 2, y_p - \frac{3}{2})$$

$$\Delta F = f(M_1) - f(M)$$

$$= \cancel{x}$$

$$= \cancel{x} (x_p + 2)^2 + (y_p - \frac{1}{2})^2 \cancel{x} - (x_p + 1)^2 - (y_p - \frac{1}{2})^2 + \cancel{y}$$

$$= 4x_p^2$$

$$= x_p^2 + 4x_p + 4 - x_p^2 - 2x_p - \cancel{1}$$

$$= 2x_p + 3$$

$$= 2x + 3$$

$$\Delta SE = f(M_2) - f(M)$$

$$= (x_p + 2)^2 + (y_p - \frac{3}{2})^2 - r^2 - (x_p + 1)^2 - (y_p - \frac{1}{2})^2 + r^2$$

$$= x_p^2 + 4x_p + 4 + y_p^2 - 2y_p \cdot \frac{3}{2} + \frac{9}{4} - x_p^2 - 2x_p - 1$$

$$= y_p^2 + 2y_p \cdot \frac{1}{2} + \frac{1}{4} - \frac{1}{4}$$

$$= 2x_p - (\frac{3}{2} - \frac{1}{2}) y_p + 3$$

$$= 2x_p - (3-1)y_p + \left(4-1-\frac{1}{4}\right) + \frac{9}{4}$$

$$= 2x_p - 2y_p + 5. \quad ; (x+1, y+1) \text{ box corners}$$

$$= 2x - 2y + 5 \quad ; (x+1, y+1) \text{ box corners}$$

$d_{init} = ?$

$$(1, r - \frac{1}{2})$$

$$1^2 + \left(r - \frac{1}{2}\right)^2 - r^2 = 0$$

$$\Rightarrow 1 + r^2 - 2r + \frac{1}{4} - r^2 = 0$$

$$\Rightarrow 4 - \frac{5}{4} - r^2 = 0$$

$$d_{init} = \frac{5}{4} - r^2$$

$$x = 0, y = r.$$

$$d_{init} < 0 \text{ if } r > \frac{5}{4}$$

\therefore circle একটু বড় হলেই initial value negative.

if $r = 1$ pixel কেবল আহলেই 1st point নিচ্ছে দিয়ে যাবে,
সো, d +ve otherwise d -ve.

$$x = x+1, y = y$$

$$\leftarrow d = d + 2x + 3$$

else,

$$x = x+1, y = y-1$$

$$d = d + 2x - 2y + 5$$

d এর value initial value fractional. তাছে অভ্যন্তরীণ
4 দিয়ে শুন ক্ষয়ত হবে।

Derive a function যাতে x, y টিলে 8 টা draw ক্ষয়ত
circle হবে।

During Exam

$$x=0$$

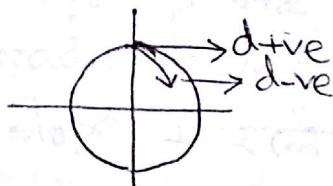
$$(x+0)^2 + (y+30)^2 = r^2$$

$$\therefore r = \sqrt{30^2} = 30$$

$$x^2 = 30y$$

$$y^2 = \frac{x^2}{30}$$

$$y = \pm \frac{x}{\sqrt{30}}$$



$$x^2 + y^2 = 1^2$$

$$x^2 + y^2 = 9$$

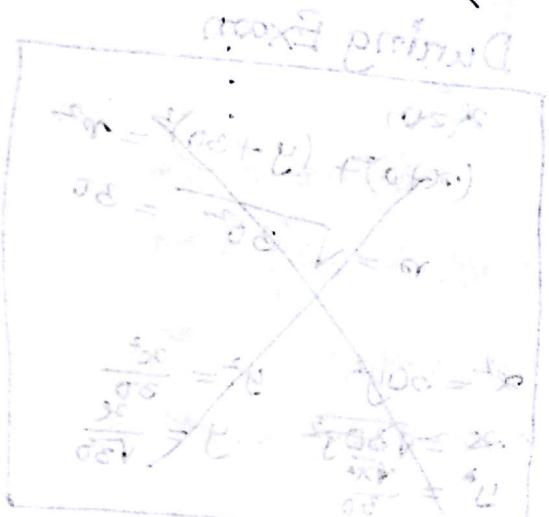
$$x^2 + 3^2 = 11$$

void draw8way(int xc, int yc, int x, int y){

drawPixel(x+xc, y+yc); \rightarrow for zone 0

drawPixel(y+xc, x+yc); \rightarrow " " 1

drawPixel(-y+xc, x+yc); \rightarrow " " 2



$$0 = ax - \left(\frac{b}{5} - 6\right) + 30$$

$$0 = ax - \frac{b}{5} + 6a - ax + b \leq 0$$

$$ax - \frac{b}{5} \leq 6a - b$$

$$a - \frac{b}{5} = 6a - b$$

$$a = b \Rightarrow a = 6$$

$$\frac{b}{5} \leq a \Rightarrow b \geq 30$$

anti-aliasing lattices have also a plus one

- so they are roughly $\sqrt{2}$ times larger than the original

(the original lattices have a plus one by $b = a + 1$)

- so they are $\sqrt{2} + 1$ times larger than the original



$$B = b + a = 10$$

$$a + b = 6$$

$$B - b = B - a = 4$$

antialiasing lattices have a $\sqrt{2} + 1$ times larger lattice than the original

1. does not take into account a square

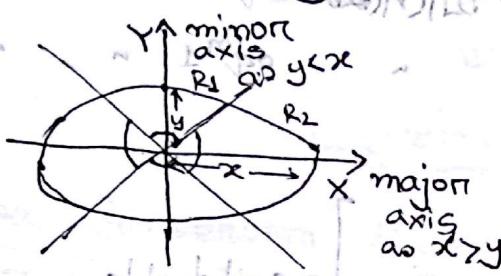
2. does not consider

Lecture - 7

19.03.2014

Scan conversion algorithm.

Ellipse drawing



$$f(x, y) = b^2x^2 + a^2y^2 - a^2b^2 = 0 \quad \text{--- (1)}$$

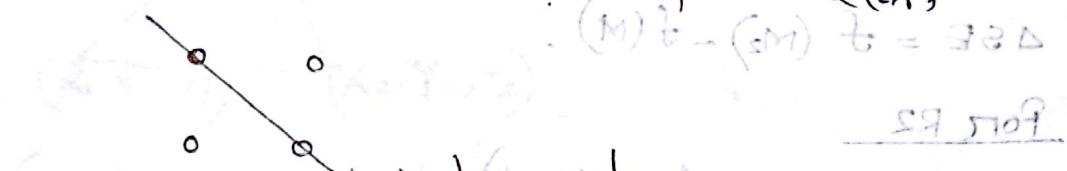
$a=b$ হলে circle

১/৪ অংশ draw কর্তৃত রয়ে

১. x অক্ষিক বাটু
 ২. $\Theta \rightarrow R_1$ এর mid point
 ৩. y অক্ষিক কম
 ৪. R_2 এর mid point

- tangent vector এর perpendicular কে gradient হল,
 Gradient is perpendicular to the tangent.
 Slope is the angle, gradient is the vector.

Gradient vector এর slope $\frac{\Delta y}{\Delta x} = M$



Gradient vector এর component = y component.
 $(\Delta x = \Delta y) = 1$.

Tangent হলো first derivative of surface.

$$\frac{\partial f}{\partial x} = M \quad \frac{\partial f}{\partial y} = N$$

Exercise - 3

$$\text{grad } f(x, y) = \frac{\partial f}{\partial x} i + \frac{\partial f}{\partial y} j = 2b^3x i + 2a^2y j \quad \text{(1)}$$

i component হলো x এর পার্য্যক্ষ eqn 1 এর derivative
j " " " " " " " "

R_1 এ $2a^2y j$ এর value এটি

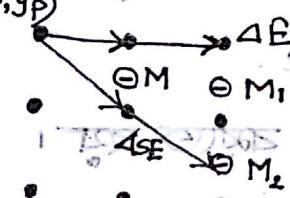
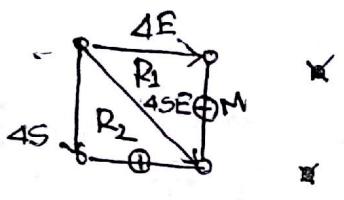
R_2 " $2b^3x i$ "

Gradient point

এ exactly $i = j$.

(x_p, y_p)

switchable condition



$a^2y > b^2x$ for R_1 .

if $\left(a^2(y_p - \frac{1}{2}) < b^2(x_{p+1} + 1)\right)$ then switch (to region 2).

Force R_1 $M = (x_{p+1}, y_p - \frac{1}{2})$

not freeible $\rightarrow M_1 = (x_{p+1}, y_p - \frac{1}{2})$ নেতৃত্ব ফ্ৰিবিলেবল

friction $M_2 = (x_{p+2}, y_p - \frac{3}{2})$ নেতৃত্ব ফ্ৰিবিলেবল অফ ফ্ৰিভন

not freeible \rightarrow স্থৰে নেতৃত্ব ফ্ৰিবিলেবল

$\Delta E = f(M_1) - f(M)$ \rightarrow নেতৃত্ব ফ্ৰিবিলেবল

$\Delta SE = f(M_2) - f(M)$.

Force R_2

$M = (x_p + \frac{1}{2}, y_p + 1)$

$M_1 = (x_p + \frac{1}{2}, y_p - 1)$

$M_2 = (x_p + \frac{3}{2}, y_p - 2)$ নেতৃত্ব ফ্ৰিবিলেবল

$(L = f(M_2) - f(M))$

$\Delta S = f(M_1) - f(M)$ \rightarrow নেতৃত্ব ফ্ৰিভন

$\Delta SE = f(M_2) - f(M)$.

R^1 final d is the initial d of R^2 .

$x=1, y=b-\frac{1}{2} \rightarrow \text{initial d init}$ এর কথা জন্য সুনি 1
এ ব্যাতে হার।

Lecture-8

24.03.2014

Scan Conversion Algorithms:

Polygon Filling Algorithm

or algorithm = [S - E]

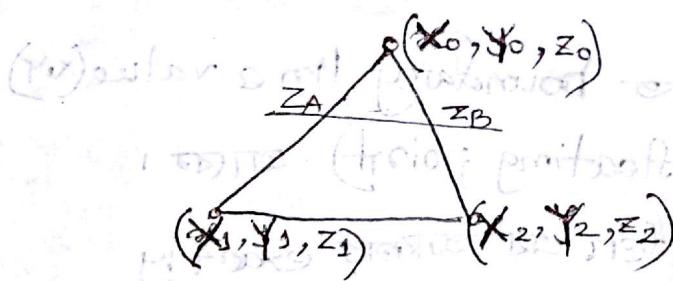
Scanline Algorithm

Concave, Convex, Twisted, Hexagon, Octagon.....
মুক্তিমান polygon দ্রব্যের ক্ষেত্র কী বা, এর visibility test করে।

Light position, Glittering object.

Central processor \rightarrow corner point শূলো দেয়।

Local \rightarrow মাঝখালে coordinate শূলো দেয়।



$(X, Y) \rightarrow$ projected value.

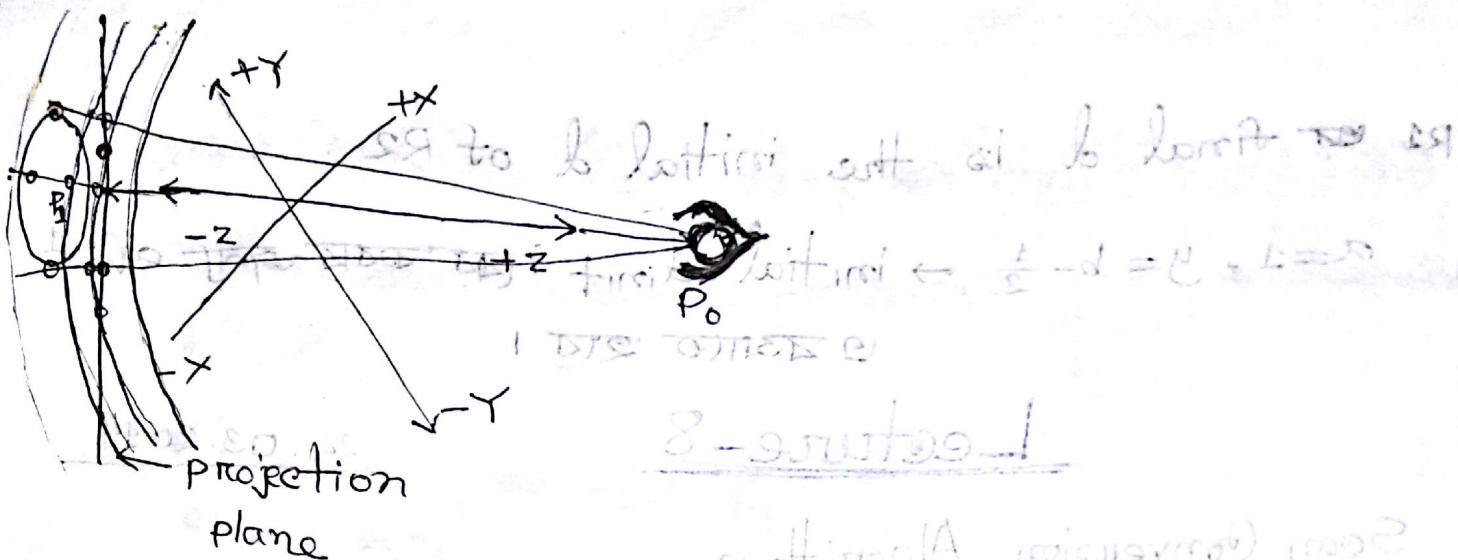
$(x, y, z) \rightarrow$ real value of original polygon.

viewer এর পুরু পুরু positive z axis থেকে যতুক্ত পোজিশন
রঙের উপরের দৃশ্য -> +z

- Norm

Right hand rule $\rightarrow +z$ axis থেকে viewer দৃশ্য,

Left " $\rightarrow -z$ "



$$|P_1 - P_0| = \text{radius } r$$

Visibility test algo \rightarrow buffering algo.

Filling এর আজে boundary draw করে নওয়া হয়।
Line drawing algo দিয়ে ও কাজ করা হয়।

Always line আগতে left to right fill করতে
করতে থাক।

$$z = z_0 + t(z_1 - z_0)$$

↓
depth
value

দূরটা array থাকে। একটা boundary line value(x,y)
এবং আগেক্ষণত z value (floating point) থাকে।

Color buffer আর z buffer পর আগত exactly
same.

z buffer প্রথম অনেক দূরের কোন point দিয়ে

initialize করা হয়। সব্বে z value এই আগের
চূড়া কম হয় অহল nearest z value দিয়ে

update করা হয়। — মুকোতি of multiple polygon
filling.

Lecture 19

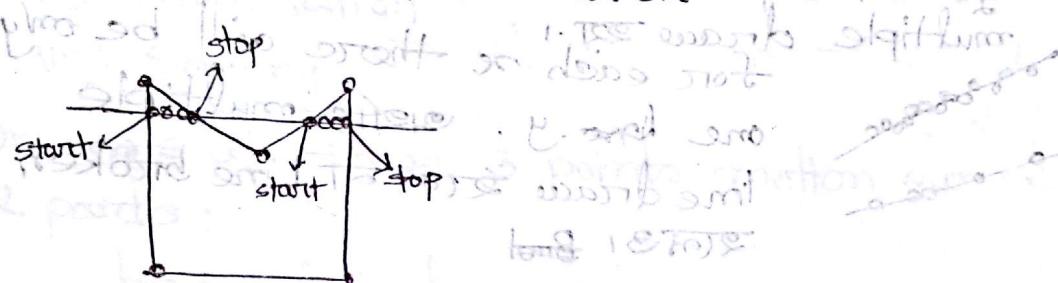
23.02.2021

Surface rendering: *ଏହିକିମ୍ବା ପାଇଁ କିମ୍ବା*

Internally draw *କଣ୍ଠ ରୂପ*, *ଅନ୍ତର୍ଗତ ବିନ୍ଦୁ* ଥାଏ
ଦେଖିଲେ ଯା ଦେଖା ଥାଏ ତା ହଜାର ସୁରଫ୍ଟ୍ କାମିକ୍ ଏବଂ
ଅଭିନାଶ କରିବାକୁ ହୁଏ ।

Depth buffering → ideal case.

କିମ୍ବା draw କରିବାକୁ ହୁଏ *z buffer* କିମ୍ବା *depth buffer*



Filling କରାଯାଇ ଥିଲୁ ପରିଚ୍ୟ କରିବାକୁ

odd parity → draw.

Even n → Don't draw.

Odd numbers

Odd numbers of intersection ଥାବଳୀ ଏବଂ ଏକ କଣ୍ଠ କିମ୍ବା ପରିଚ୍ୟ କରିବାକୁ
ହେବାକୁ ଏବଂ even parity କରିବାକୁ ହେବାକୁ

Recover → ରାତ୍ରି horizontal line ଥାଏବୋ ଏବଂ ବାଦ କିମ୍ବା ହେବାକୁ
କରିବାକୁ

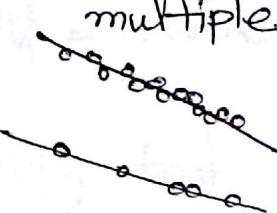


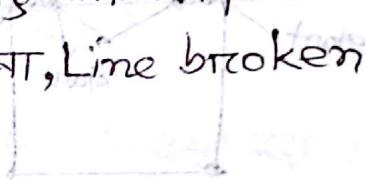
Exactly horizontal line ଗୁଣ୍ଠିତ ହୁଏ କିମ୍ବା draw
କରିବାକୁ ।

→ যদ্বারা multiple time intersect করে যোগান
multiple pixel যান করতে হবে।



এই line এর আকৃতি আছে।

→  and label \leftarrow pixelated line
যদি কোন slope এর line এর (কোন একটি) line
multiple trace হয়। i.e. for each \Rightarrow there will be only
one line.y. অর্থাৎ multiple
line trace হবে না, Line broken
হলেও। But



Line broken হলে history tape শুরু পরিষ্কা
র কাজ করা হবে।
সেখানে \leftarrow history tape
work \leftarrow history tape
work + nod \leftarrow work
history tape

যদি সম্ভব হয় তবে একটি
line to line transition করে রেখা রেখা
কে সাথে সাথে একটি স্লোপ মেনে নেওয়া
যাবে। এটি একটি পরিষ্কার রেখা।

যদি এই নির্দেশ গ্রাফ করে রেখা রেখা
কে সাথে সাথে একটি স্লোপ মেনে নেওয়া
যাবে। এটি একটি পরিষ্কার রেখা।

যদি এই নির্দেশ গ্রাফ করে রেখা রেখা
কে সাথে সাথে একটি স্লোপ মেনে নেওয়া
যাবে। এটি একটি পরিষ্কার রেখা।



Lecture - 9

25.03.2014

Motion:

Geometric transformation:

$$(x, \frac{x}{w}, \frac{y}{w}, \frac{z}{w}) = (x, y, z)$$

$$2D \rightarrow \boxed{\text{Motion}} \rightarrow 3D (x, y, z) = (x, y, z)$$

Motion is geometric transformation.

(i) Rotation

(ii) Translation

(iii) Scaling

On basis of these 3 points motion can be decided into 2 parts:

- Rigid body motion → motion এর আঙ্গে পরে ছাঁচ রয়েছে।
হলে যদি shape change না হয় তাহলে।
- Affine motion → shape change / বিকৃতি হলে।
→ soft body / plastic type.

Transformation → Geometric - center of rotation is the origin

Co-ordinate - নিজ center of rotation এর স্থান বর্জন
- rotation is done based on global origin

Homogenous co-ordinate system: based কর্তৃপক্ষ

$$(x, y, z) \leftrightarrow (x, y, z, w)$$

কার্ডিনেট
homogenous, w এর স্থান

E - Structure

In PG, instead of cartesian, homogenous is used.

$$(x, y, z) = \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}, 1 \right)$$

$$(3, 5, 7) = (6, 10, 14, 2)$$

$$= (3, 5, 7, 1)$$

Transformation matrices:

Rotation and scaling is multiplication-based
Translation is add/subtraction

In Rotation:

$$\begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix} = \begin{vmatrix} R_x & 0 & 0 & 0 \\ 0 & R_y & 0 & 0 \\ 0 & 0 & R_z & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

Translation:

$$x' = x + dx$$

$$y' = y + dy$$

$$z' = z + dz$$

$$\begin{vmatrix} x' \\ y' \\ z' \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

Imp

- * Difference between cartesian and homogenous.
- * Why homogenous system is used?

Rotation:

1 axis based rotation \rightarrow simple. ($\pm 90^\circ$)

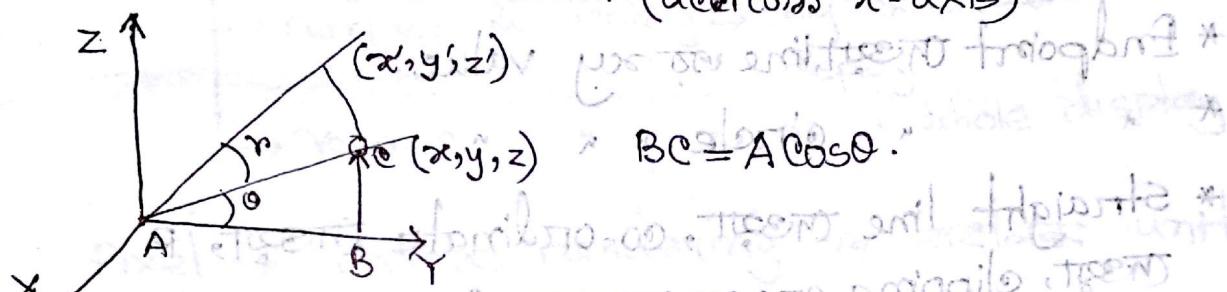
multiple axis \rightarrow composite.

pos x ଥାକେ pos origin ଯିଦିକେ ଅବଗାଳେ counter clockwise
rotate ହସ୍ତ ଦୟାଳୁ ଯେଉଁ positive rotation on x axis. Clockwise
ହଜେ negative rotation.

pos y ଥାକେ counter clockwise (+)ve on y. axis.

x axis କଣ୍ଠ କରି ଦୂରଳେ y, z ପରିବର୍ତ୍ତନ ହସ୍ତ କାହାରେ but x unchanged.
y axis ବାରେ ବାରେ କାହାରେ କାହାରେ x unchanged ଥାକେ।

$|R_{0x}|$ = Rotation matrix (across x-axis)



$$y = r \cos \theta_x$$

$$z = r \sin \theta_x$$

$$x = x$$

$$x' = x$$

$$y' = r \cos(\phi_x + \theta_x)$$

$$z' = r \sin(\phi_x + \theta_x)$$

from ① \Rightarrow

$$x' = x$$

$$y' = r \{ \cos \phi_x \cdot \cos \theta_x - \sin \phi_x \cdot \sin \theta_x \}$$

$$z' = r \{ \sin \phi_x \cdot \cos \theta_x + \cos \phi_x \cdot \sin \theta_x \}$$

$$x' = x$$

$$\therefore y' = y \cos \theta_x - z \sin \theta_x$$

$$z' = y \sin \theta_x + z \cos \theta_x$$

So, 4×4 matrix,

$$\begin{vmatrix} x' \\ y' \\ z' \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

In course

- * Endpoint दृश्यता, line के key value
- * " " circle का " " equation
- * straight line दृश्यता, co-ordinate दृश्यता, प्रकाश्यता, clipping area दृश्यता, + कोष्ठ?

$$① \rightarrow \begin{cases} (\omega_0 + \omega_1) \cos\theta = E \\ (\omega_0 + \omega_1) \sin\theta = F \end{cases}$$

$$② \rightarrow \begin{cases} \omega_0 \cos\theta = G \\ \omega_0 \sin\theta = H \end{cases}$$

$\Leftrightarrow ①$ most

$$x = X$$

$$\{\omega_0 \cos\theta - \omega_1 \cos\theta\} \theta = Y$$

$$\text{line का } \theta \text{ तथा } \omega_0 \text{ तथा } \omega_1 \text{ का } \theta = Z$$

$$③ \rightarrow \begin{cases} \omega_0 \cos\theta - \omega_1 \cos\theta = E \\ \omega_0 \sin\theta + \omega_1 \sin\theta = F \end{cases}$$

Lecture - 10

31.08.2014

Geometric Transformation:

$$\begin{array}{ccc}
 P & P' & P'' \\
 Qx - Qy & \rightarrow & Qz \\
 40^\circ & -30^\circ & 45^\circ \\
 -Qz & \rightarrow & -Qy \\
 -\frac{45^\circ}{P''} & \rightarrow & P' \\
 & \rightarrow & P'' \\
 & \rightarrow & -Qx \\
 & \rightarrow & -40^\circ \\
 & \text{(matrix)} &
 \end{array}
 \rightarrow \text{Opposite matrix}$$

যে পথের অংশটা যে ক্ষেত্রে ভাজ হবে।

Viewing in 3D:

Rendering

- Surface Rendering → only external surface drawing
- Volume → as a whole display.

Pixel → Surface rendering এর একটি unit, circle, square.

Voxel → Volume, sphere, volume

- supernatural
- medical side to use হয়, internal tumor detection easier হয়।
- transparent করে দেয়।

→ 6D MRI → real time 3D volume rendering করা

MRI → Magnetic Resonance Imaging.

→ (যাঁর দোষটা চাই) যাঁর দুষ্টা যাকে নিনিত স্বচ্ছ

করা হয়। semitransparent অব্যাক্ত transparency change করার shape কুণ্ডা যায়।

percentage change করার shape কুণ্ডা যায়।

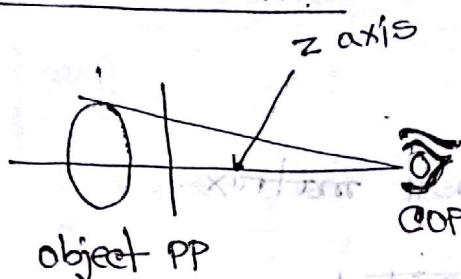
Rendering pipeline এর first stage হল 3D motion.

Projection:



Q1 - Projections

Classifications:



Classification based on COP

COP → independent coordinate system as

position of COP changes frequently.

Object, PP, COP → তাদের নিঃস্থ স্থিতিকে independent.

→ তিনির নিঃস্থ origin আছে।

এসের পথেকে একটি common coordinate system এ আনতে হবে, যাতে origin একটি হয়।

Based on the position of COP:

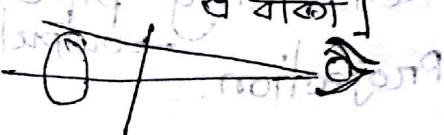
(i) Parallel projection. [Infinite distance]

(ii) Perspective " [Finite]

Based on the orientation of PP:

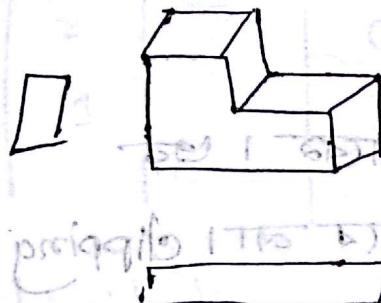
(i) Orthographic projection. [PP perpendicular to axis]

(ii) Oblique [axis এর respect অবস্থা]



ଏକେଟି Object ଦ୍ୱାରା ହୋଇଥିଲେ orthographic view.

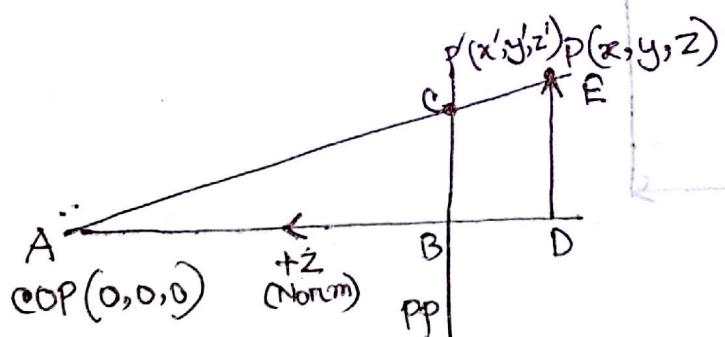
→ Engineering point of view



gl Ortho

gl Perspective

Derivation of projection matrix:



origin → COP

Top/bottom → y axis

left/right → x axis

near → +z

far → -z

All objects are on -z axis

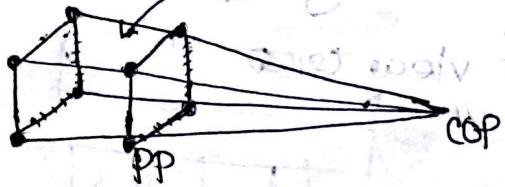
$$\frac{B}{DE} = \frac{B}{AB}$$

$$\frac{B}{DE} = \frac{B}{AB} = \frac{B}{BD}$$

$$\frac{B}{BD} = \frac{B}{B} \Leftrightarrow \frac{B}{B} = \frac{B}{B}$$

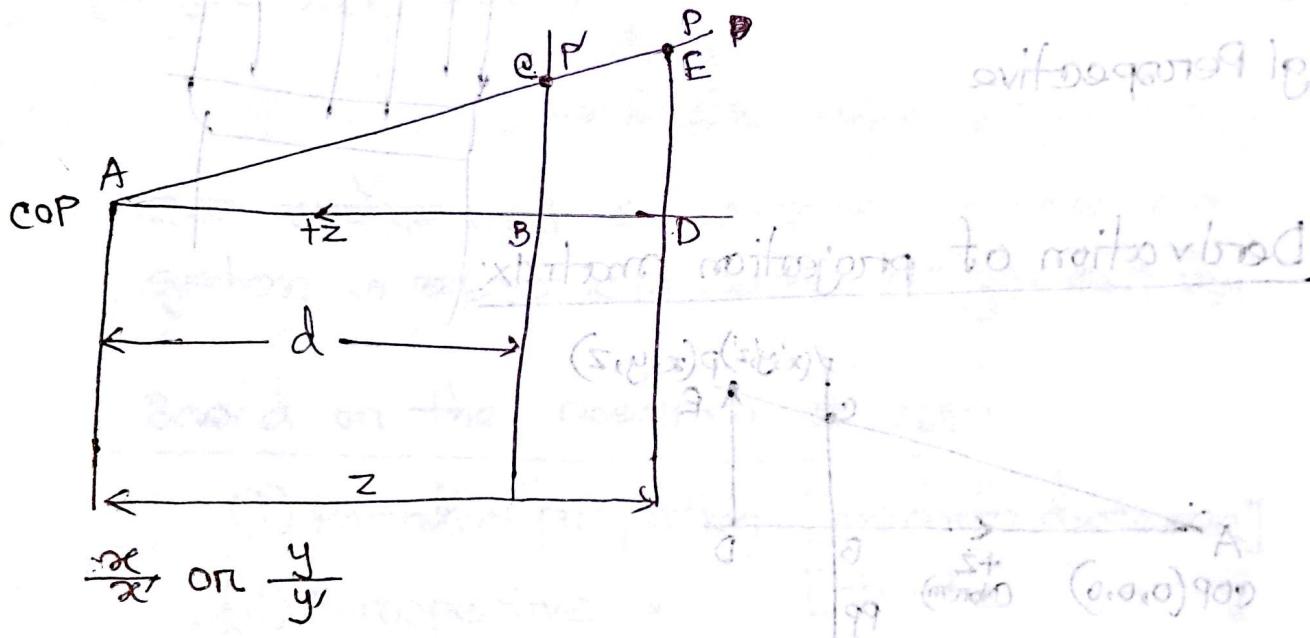
$$\frac{B}{BD} = \frac{B}{B} \Leftrightarrow \frac{B}{B} = \frac{B}{B}$$

$$\frac{B}{BD} = \frac{B}{B} = \frac{B}{B}$$



view volume এর মিলেই object থাকে। এর
বাইকে থাকলে object আর দেখা যাবে না। clipping
line এর মাত্র।

simpliest perspective projection.



$\triangle ABC$ and $\triangle ADE$

$$\frac{DE}{BE} = \frac{AD}{AB} \parallel \frac{z}{d}$$

$$\frac{x}{x'} = -\frac{z}{d} \Rightarrow x' = \frac{x}{z/d}$$

$$\frac{y}{y'} = -\frac{z}{d} \Rightarrow y' = \frac{y}{z/d}$$

$$x' = \frac{z}{z/d} = d$$

M-structure

$$\begin{vmatrix} x' \\ y' \\ z' \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

$\rightarrow 4 \times 4$ derivative projection matrix.

Limitation:

z एवं value कम्याना 0 होते नाही ता 100 एवं वर्गाच्याचि डोने, huge इडू याया।

अज्ञत origin at COP एवं equation मध्ये normally use करा शक्य ना।

→ set of relationships between variables of the camera
one relationship is the center of projection
box fit equation, value = 0 or 1
.0 = 0 or 1 relationship to distance of object
COP (x_c, y_c, z_c) → relationship of interest
 $b = 0.4, b_1$
0.4 = 0.4, 0.4

→ relationship between a and b
relationship between c and b
relationship between d and b

the coefficient value, $a = \frac{30}{30} = 1$

$$b = \frac{30}{30} = 1 \leftarrow$$

$$b_1 = \frac{30}{30} = 1 \leftarrow$$

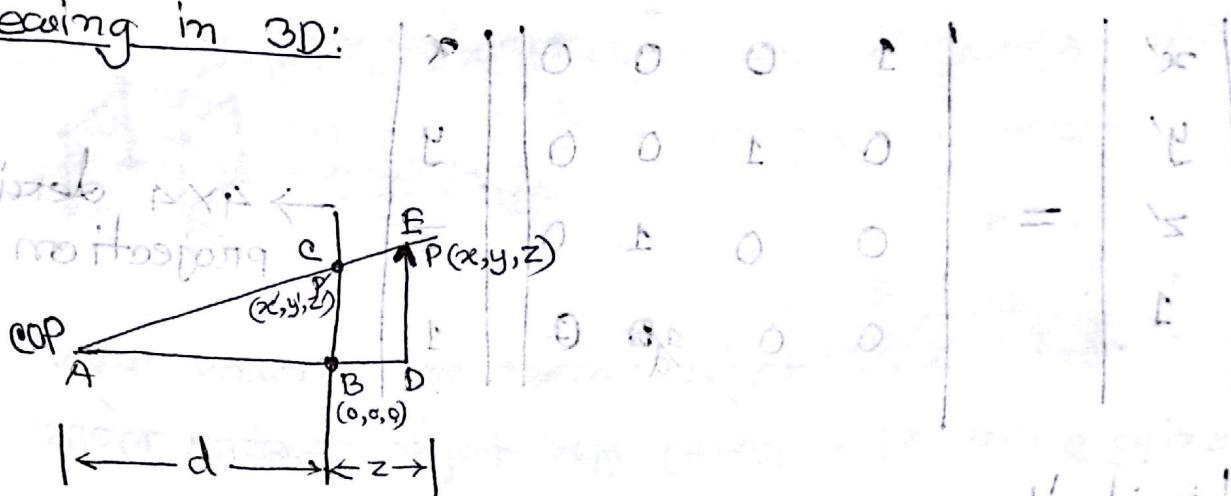
$$c = \frac{30}{30} = 1 \leftarrow$$

$$d = \frac{30}{30} = 1 \leftarrow$$

Lecture-11

01.04.2014

Viewing in 3D:



Computing projection plane, origin.

Since PP is a value variable, it's fixed at 2D.

Since

PP is always parallel to the z axis. So, z value always fixed.

Origin is center of projection plane so, $z=0$.

Orthographic projection \rightarrow

Computer graphics \rightarrow absolute scale/dimension

Modelling transformation \rightarrow real value

projection value \rightarrow output

$$\frac{DE}{BC} = \frac{AD}{AB}$$

$$\Rightarrow \frac{x'}{x} = \frac{z+d}{d}$$

$$\Rightarrow \frac{x'}{x} = \frac{d}{z+d}$$

$$\therefore x' = \frac{x}{1+z/d}$$

$$\text{Similarly, } y' = \frac{y}{1+z/d}$$

$$\text{and } z' = 0$$

} ①

More stable than previous day's eqⁿ. $\vec{H}^n = \vec{H}^q$

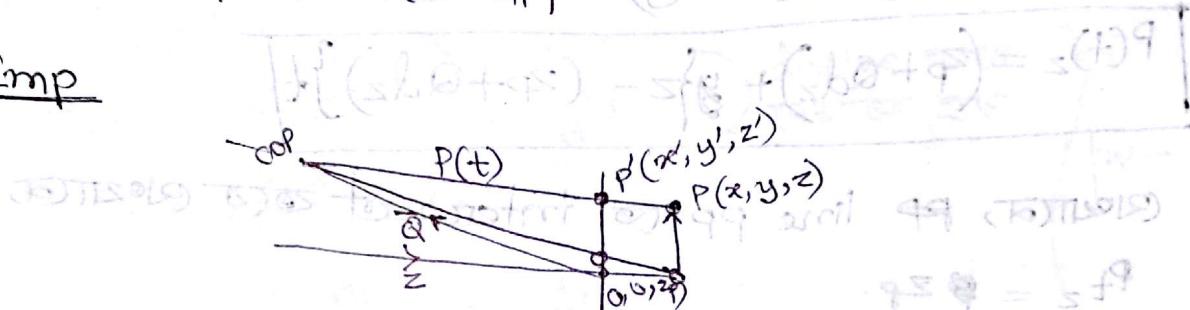
$$\begin{vmatrix} x' \\ y' \\ z' \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

$(\vec{H}^n - \vec{Q}) + (\vec{G}^n - \vec{Q}) = \vec{H}^q$

$w = \frac{z}{x}$

ইপরুষ হল standard কাম, কান্দন origin রক্ষণা fixed জায়গায় থাকে বা,

Imp



only one constraint \rightarrow PP' is perpendicular to the z axis
origin can be at any place.

$$COP = (0, 0, z_p) + \bar{Q} \rightarrow COP, PP' \text{ থেকে } Q \text{ দ্বারা আছে।}$$

$$COP - (0, 0, z_p) = \bar{Q}$$

$$Q_x = 0 + Q_i$$

$$Q_y = 0 + Q_j$$

$$Q_z = z_p + Q_k$$

+ অন্যেকটি special value আরে যাত্র কর $\Rightarrow P_z = z_p$.

$$P_t = P_0 + t(P_1 - P_0)$$

\downarrow \downarrow

COP $P(x, y, z)$

$t=0$ হলে COP

$t=1 \text{ n } P$.

Lecture - 11

$$P(t) = COP + t(P - COP)$$

$$COP = (0, 0, z_p) + (Qdx, Qdy, Qdz)$$

$$COP_x = Qdx, COP_y = Qdy, COP_z = z_p + Qdz$$

$$P(t)_x = Qdx + (x - Qdx)t$$

$$P(t)_y = Qdy + (y - Qdy)t$$

$$\boxed{P(t)_z = (z_p + Qdz) + \{z - (z_p + Qdz)\}t}$$

যদ্বারা, PP line PP তি intersect করে যোগায়,

$$P_{tz} = z_p$$

$$x' = Qdx + (x - Qdx)t_p$$

$$y' = Qdy + (y - Qdy)t_p$$

$$z' = (z_p + Qdz) + \{z - (z_p + Qdz)\}t_p$$

এবং, $z' = z_p$, যদ্বারা PP is perpendicular to z axis.

$$t_p = \frac{z_p - (z_p + Qdz)}{z - (z_p + Qdz)}$$

$$= \frac{-Qdz}{z - (z_p + Qdz)}$$

$$(4 - 4) + + 4 = 4$$

$$(4 - 4) + + 4 = 4$$

$$900 - 0 = 900$$

$$900 - 0 = 900$$

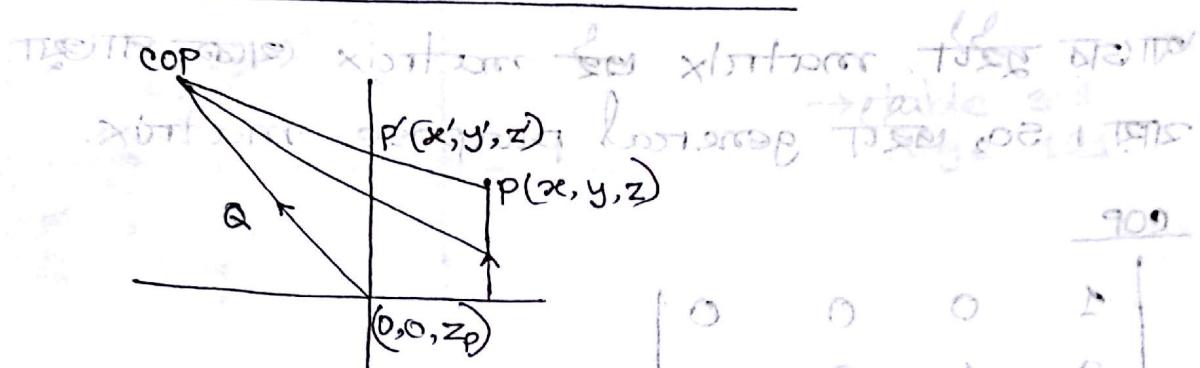
$$900 - 0 = 900$$

$$900 - 0 = 900$$

Lecture-12

02.04.14

Projection Matrix Derivation



Imp

$$x' = Qdx + (\alpha - Qdx) + p \quad \left(\begin{array}{cccc} 0 & 0 & 0 & \alpha \\ 0 & 0 & 0 & p \\ 0 & 0 & 0 & 0 \end{array} \right)$$

$$= \frac{x - z \cdot \frac{dx}{dz} + zp \cdot \frac{dx}{dz}}{\frac{zp - z}{Qdz} + 1} \quad \text{derive} \quad \left(\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right)$$

$$y' = \frac{y - z \cdot \frac{dy}{dz} + zp \cdot \frac{dy}{dz}}{\frac{zp - z}{Qdz} + 1}$$

$$z' = zp = zp \left\{ \frac{zp - z}{Qdz} + 1 \right\} / \left\{ \frac{zp - z}{Qdz} + 1 \right\}$$

N.V.I

$$\begin{vmatrix} x' \\ y' \\ z' \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & -\frac{dx}{dz} & \frac{zp - z}{Qdz} \\ 0 & 1 & -\frac{dy}{dz} & \frac{zp}{Qdz} \\ 0 & 0 & -\frac{z}{Qdz} & \frac{zp(zp/Qdz + 1)}{Qdz} \\ 0 & 0 & -\frac{1}{Qdz} & \frac{zp}{Qdz} + 1 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \\ 1 \end{vmatrix}$$

* Q, z_p and (x, y, z) are given. projection coordinate করা হবে।

General purpose matrix \rightarrow picovle ~~कर्तव्य~~ \rightarrow 9

आजेक दुखी matrix एवं matrix थार्ड सिट्या
याय। So, एही general purpose matrix.

COP

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1/d \end{vmatrix}$$

$$zp = d.$$

$$(v) \quad Qdz = -d.$$

PP

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/d & 1 \end{vmatrix}$$

$$zp = 0$$

$$Qdz = -d.$$

Projection/rotation related math.

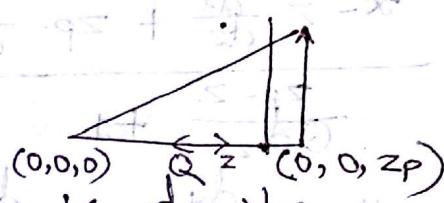
Line drawing एवं math

Circle



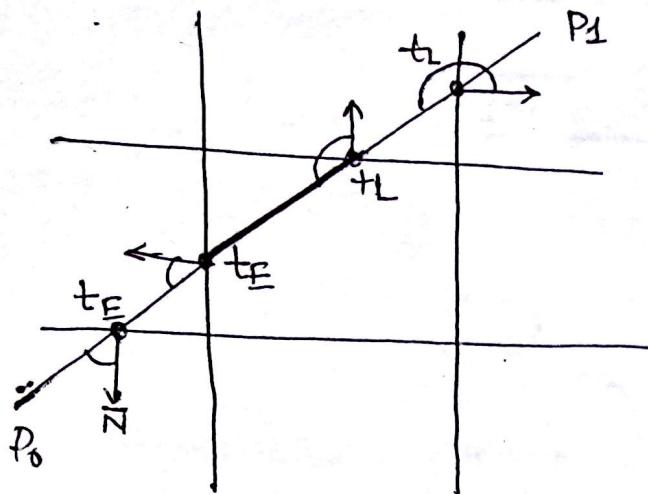
$$x' = \frac{x}{zd}$$

$$+ (y/d) + \frac{z}{d} = 1$$



$$x' = \frac{x}{zd}$$

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/d & 1 \end{vmatrix} \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$



→ table 3.1
page 121.

একটি line এর জন্য t_E & t_L রয়ে কাব।

ছুটি corner coordinate দিয়ে থাকা।

N রয়ে কাব নিয়ে হবে।

maximum t_E / t_L কত?

minimum

Max থেকে Min পর্যন্ত Line draw করে।

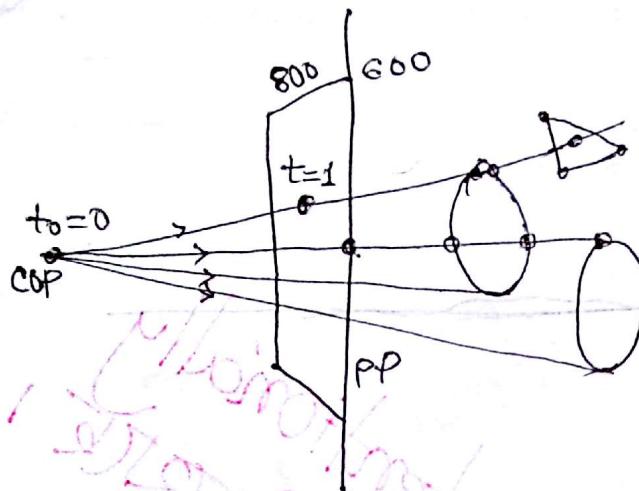
t_E

t_L

Lecture-13

• 16-04-2014

Visible Surface Ray Tracing:



- Light COP রে ডিয়ে বিলি হয়। \rightarrow conventional
- COP থেকে ray লেব হয়। \rightarrow ray tracing.
- অন্তর্ভুক্ত ray simultaneously অন্তর্ভুক্ত equation solve করে,
- আলাদা রূপ projection matrix নাই না, known ray use, করে intersection কুলো করে কষা হয়,
- Projected point determine হয়, ray রে object রে অন্তর্ভুক্ত intersect করে এটা থেকে object রে 2D-ordinate জানা যায় on 2D.
- একই ray রে multiple intersection হলে \Rightarrow nearest distance of COP রে visible করা যায়। যাকে কুলো hide হয় চায়।
- শুলভ raster graphics রে জন্ম, vectors রে জন্ম নয়, ফিল্ট রে output interestingly vectors পড় রে।

known value COP and pixel co-ordinate.

- यात्रा शर्तुले point थांत्रे शर्तुले ray trace झाले
एवं उपर्युक्त रूपाने overlap हय वा, तरी एवं output
analog तर यात्रा,
- Geometric equation द्यावा object शुल्कावा handle कर्या याया वा।
→ Weakness of ray tracing.
- मूलिक्षण equation आहे, तरी random object के handle कर्यात पाढू
ना। येत्या object एवं equation define कर्या केवल अद्वितीय मात्र।
Theoretically येती ray एको pixel के 1 यात्रा intersect
करते। In real life different case होते मात्रे।
- Ray Object के intersect कर्याव यात्रा यात्रा 2D डिस्प्ले
प्राप्त्या याया।

(i) Projected point.

(ii) + अना शायगले येत्याव थांत्रे Object एवं
co-ordinate.

(iii) Camera ray.

(iv) Ray buffer.

- 800×600 pixel एवं असलेला $1 \text{ pixel} \times 1 \text{ pixel}$ अनु अनेक वेळी
calculation कर्या लागे। तरी एवं अनेक वेळी time
consuming. तरी animation एवं कृती व्है कर्या याया
ना, still image एवं कृती व्है कर्या याया।
- Object semi-transparent होता refractive index
आवा शायगल magnified display कर्या याया।

Object এর mutual reflection করে calculate করে

shadow / semitransparent করা হায়।

Global line model

$$x = x_0 + t(x_1 - x_0)$$

$$y = y_0 + t(y_1 - y_0)$$

$$z = z_0 + t(z_1 - z_0)$$

n number of object এর অন্তরে যাই একটা

ray কে intersect করতে হয়।

Eqn of sphere front side এর বেলে রেখা

$$(x-a)^2 + (y-b)^2 + (z-c)^2 - r^2 = 0 \quad \text{⑪}$$

- Fraction algorithm \rightarrow (?)

Object এর mutual reflection calculate করে

Object এর সব আলো এমন কিছুটা আলো reflect করে। internal এবং আলোর এমন environment আলাদিত হয়

$t = 0 \rightarrow \text{cop}$

$t = 1 \rightarrow \text{PP}$

$$At^2 + Bt + C = 0 \quad \text{from ⑩ and ⑪}$$

Polynomial কাগজ এবং শাখা variable হচ্ছে

variable আরো geometric হচ্ছে।

$$\text{root} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

① Root = $\sqrt{-1} \Rightarrow A AC > B^2 \rightarrow \text{imaginary.}$

② ray intersect করে না। so pixel এর background।

③ $B^2 - 4AC = 0 \rightarrow \text{single root.}$

Exactly $\sqrt{B^2 - 4AC} > 0 \rightarrow$ real value.

(iii) $B^2 - 4AC > 0 \rightarrow$ real value.

যেটি smaller + near projection value.
যারিগুলো যথ যান।

১ নং এ t এর value ক্ষেত্রে object surface এ intersection হাত।

Lecture-14

21.04.2014

Chapter-16

Illumination and Shading:

Illumination \rightarrow আলো বঙ্গুর স্থান অতি সত্ত্ব হওয়া।

আলো রেখার চারে আয়তে দাঢ়ে।

(i) Source হিসেবে।

(ii) Reflection এর ক্ষেত্রে (Object).

চোখে দৃশ্যমান 2D (বেল্ট)।

2D একে 3D এর দৃশ্য হাত করে আয়ত এর ক্ষেত্রে,
+ knowledge গড়ে উন্নত করে।

Object shape shading এর ক্ষেত্রে জ্ঞান যান।

Shading \rightarrow বঙ্গুর surface orientation এর

বিপরীত দৈর্ঘ্য of reflected
light / brightness variation.

shadow → আলোর বাবি পাত্রে।



• shadow casting করা হয় + shadowing করা

যদি একটা cube এবং এমন এর অন্তরে একটা
আবে light reflect করে তাহলে একে hexa-
gon রেখায়।

Brightness variation হলেই তা এর 3D ল্যান্ড

কে cube এর মতো রাখ হয়।

Almost real world কে emit করা

→ purpose.

Light source:

2 types.

(i) Source থেকে যে light যায়, → light of emission

(ii) একটা object থেকে reflected হয়ে আসার,

Based on light source: → Reflected light.

① Global Light Model

② Local

Recursive Ray tracing; algorithm:

reflection to previous reflection
recursion - until hit point

~~Imp~~ Difference between global and local light model?

- mutual reflection तर੍ਹਾਂ global ਵਿਖੇ consider ਕਰਦੇ।
local ignore ਕਰਦੇ।
- in both model light comes from source.
- Local light model divides light in 2 divisions
 - (i) Ambient light → ਚਾਰੀਨਿਰੀ ਥਾਂ ਵਿਖੇ ਬਾਹਰ ਆਉਂਦਾ।
→ weak, 3D shape ਆਵਾਜ਼ ਨਾ।
 - (ii) Source light → strong.
→ ਧੜਾਵਿ ਆਈ
→ 3D shape ਆਵਾਜ਼ ਸਾਡੀ।
- * Local light & mutual reflection ਦੀ alternative ਹਿਆਂ ਵਿਖੇ ambient light ਨਾਲ ਵੱਡਾ ਹੈ।
- * Local & huge mutual reflection ਦੀ ਗੁਜ਼ਾਰੀ ਕਾਲ ਵਿਖੇ ਵੱਡਾ ਹੈ।
complexity ਰੱਖਾ। ਅਧਿਕ 3D shape ਆਵਾਜ਼ ਯੋਹੜੀ।
light ਵੱਡੀ ਅਧਿਕ ਅਤੇ ਅਨੇਕ ਆਵਾਜ਼ ਆਵਾਜ਼ ਵਿਖੇ।

$$I = I_{\text{ambient}} + I_{\text{local}}$$

$$\text{Ambient surface orientation} \times \text{ambient light} = I$$

2. Direct reflection ਹੈ।

(i) ਧੜਾਵਿ ਨਿਯਮ ਵਿੱਚ dominant ਹਾਂ।

(ii) Source light ਵਿੱਚ ਵੱਡਾ ਹੈ। $I = I_{\text{source}} + I_{\text{ambient}}$ → ਆਧੁਨਿਕ ਅਤੇ applicable

→ $I = I_{\text{source}} + I_{\text{ambient}}$ for (i) for bonding fibres or optical fibers
→ $I = I_{\text{source}} + I_{\text{ambient}}$ for (ii) for plastic boards etc.

Surface → Rough → রাষ্ট্ৰ, বেগপড়
 Surface → Glistening → metal, glass

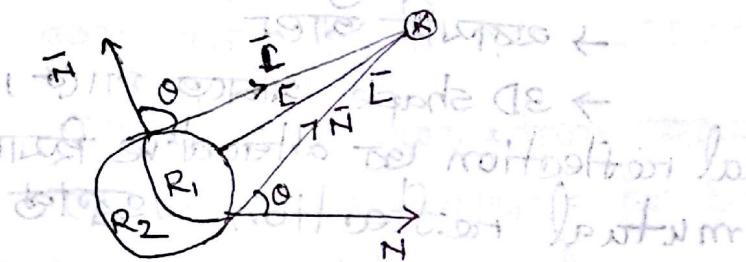
Source light এর deflection এর স্থান base কোথা:

- ① Diffuse reflection
- ② Specular reflection

অনুমিতি দেওয়া হচ্ছে অনুমিতি লেভেলে প্রতিফলন

Physics এর lambertian effect (?) হচ্ছে
 diffuse reflection.

অনুমিতি প্রতিফলন এবং অনুমিতি লেভেলে (ii)



Surface normal এবং কেন্দ্ৰীয়কে = sphere এৰ

L → light vector / vector towards light.

$$I = I_S k_d \hat{N} \cdot \hat{L} \rightarrow \text{real life এৰ diffuse}$$

— (i) reflection model.

$I =$ (i) হচ্ছে R_1 দেখা যায়, R_2 দেখা যায়,

$I =$ (ii) " R_1 দেখা যায়, R_2 দেখা যায়

$I =$ (i) + (ii) " R_1 " " , R_2 অনুমিতি হচ্ছে (i) দেখা যায়।

এই algo টে dot product (-) ve এৰ part কো
 হো discard কো যায়।

Lambert

L = Light Direction.

K_d = Diffuse absorption co-efficient
Point light

I_p = Source light intensity.

$$I = I_p K_d \cos \theta \quad (1)$$

Surface normal light এর কাছে হলে বেজি
ক্ষেত্র as $\cos 0^\circ = 1$, angle ক্ষেত্র পরিবর্তন হলে
brightness বর্ধমান ক্ষেত্র এক্ষেত্র ক্ষেত্র ক্ষেত্র full, dark হয়ে
যায়।

Normalized vector:

Normalized vector এর 3D component
square এর sum = 1.

$$\frac{\vec{N}}{|\vec{N}|} = \hat{N}$$

Normal এবং সূর্যের রেখা = H_o^9
Light এর সময়সূচি মধ্যে প্রতিক্রিয়া এবং উপরে
অভিক্ষেপণ ক্ষেত্র এবং উপরের উপরে
অভিক্ষেপণ এবং উপরের উপরে

($R = \text{Reflected light}$) $R = H_o^9 + I(R + G + B) \text{ min} = H_o^9$
Normal এর উপরে
অভিক্ষেপণ এবং উপরের উপরে

Lecture - 15

22.04.2014

Illumination and shading

Attenuation: \rightarrow कमी

(i) Light source attenuation.

(ii) Atmospheric attenuation.

Light source \rightarrow Light source intensity \rightarrow इसका अनुपात दूरी पर घटता है। यह अनुपात नियत नहीं होता, वह अनुपात लाइट स्रोत की दूरी पर घटता है।

$$Fatt \propto \frac{1}{d_L^2}$$

$$A = \frac{R}{H}$$

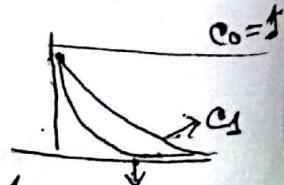
Fatt = Factor, value less than 1 because
extinguishing attenuation decreases the light
intensity.

$d_L \rightarrow$ Distance to light.

\rightarrow value less than 1 इसके लिए Fatt will
be more than 1, which is unrealistic.

Computer graphics में यूनिटी सेट स्केल नाहीं, so,
अधिकारी Fatt के reference में यह बहुत बड़ा होता है।

$$Fatt = \min \left(\frac{1}{c_0 + c_1 d_L + c_2 d_L^2}, 1 \right)$$



So, Fatt is never larger than 1.

$c_0, c_1, c_2 \rightarrow$ constant, realistic कीमत कीजिए
वे हैं, value may be 0/1.

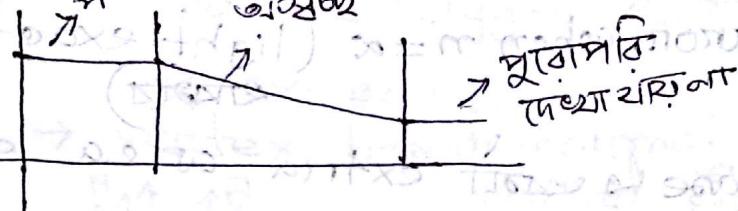
\rightarrow environment के बारे में determine करें।

Atmospheric Attenuation:

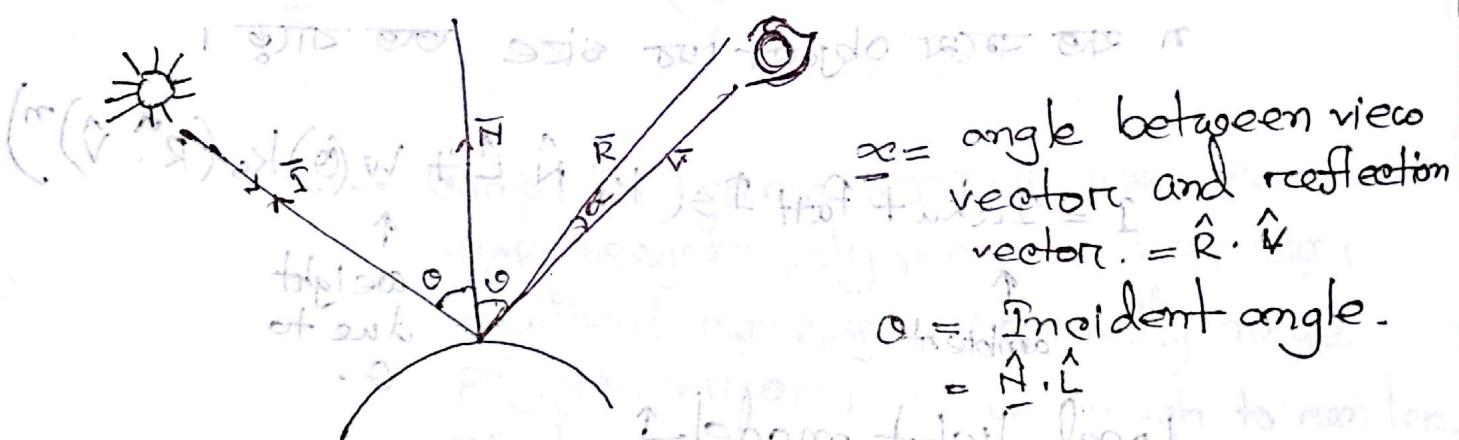
वातावर प्रकाश धूम्र की विनेशन environment
में होता है।

दूरी बढ़ावेर इलमग नीजे देखाया।

fogging effect → दूरी बढ़ावेर अस्ति / दूरी बढ़ाया वा
अप्पी बढ़ाया।



Specular Reflection: → चबूचे के बड़ुके पर प्रकाश देखाया।



$$\alpha = \text{angle between view vector and reflection vector.} = \hat{R} \cdot \hat{V}$$

$$\theta = \text{Incident angle.} = \hat{I} \cdot \hat{N}$$

$$I = I_s k_s (\hat{R} \cdot \hat{V})^n$$

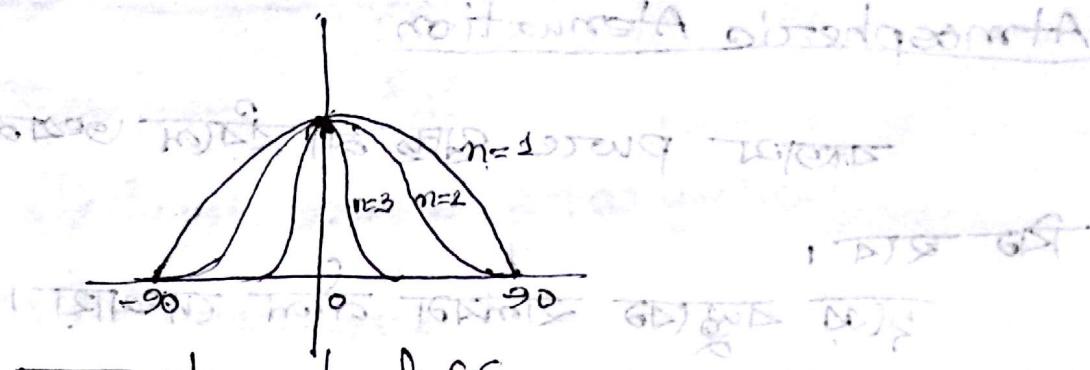
R = Reflected vector

V = View vector

n = Shiningness

k_s = co-efficient of specular surface

Normally α is very less.



$n=1$ হল almost diffuse যাতে হ্রস্ব।
 n এর value হত বাতে তত প্রভূত্বা উচ্চ (ধীর ?)

Pure mirror when $n=\infty$ (light exactly light এর মানে হ্রস্ব(১))

Other case \rightarrow extra area enlightened হয়,

Specular এবং ক্ষেত্রে dominant I_s and diffuse এবং ক্ষেত্রে dominant I_d .

n এত কমে object এর size ওত বাড়ে।

$$I = I_a + f_a I_s (k_d N \cdot L + w(\theta) k_s (R \cdot V)^n). \quad \text{--- (iv)}$$

↑
ambient

↑
weight
due to
 θ :

Local light model \rightarrow

Phong's elimination model \rightarrow এলে,

surface oriented equation.

L জন্ম থাবলে R জন্ম থাই

V \oplus একটি ক্ষেত্রে co-ordinate দ্বার্য থাবলে জন্ম থাই।

stone স্কেলে দ্বার্য থাবলে \oplus থাই

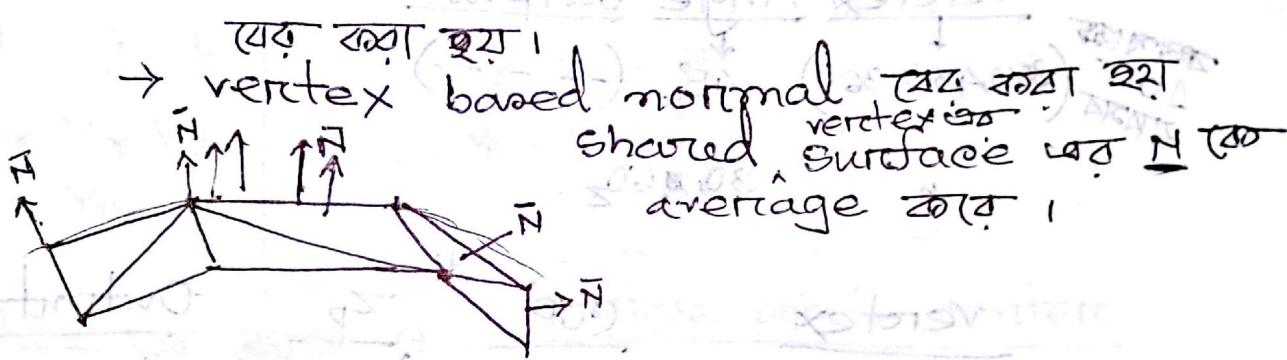
flat shading and smooth shading:

flat \rightarrow real world एवं shading / physics.

\rightarrow polygon के flat या वर्तक शद्दिंग
करा देय,

\rightarrow एकीं polygon एवं यह जायजाय in constant.

Smooth \rightarrow multiple polygon आरण्णन n तरं average



\rightarrow Computer game एवं जल्दी मै देय,
अल्पविविक्षण polygon use करा देय,
gradient एवं जल्दी normally huge
polygon लाई, it's tough to render.
That's why अल्प polygon कराना
we can give the effect of huge
no of polygon.

Lecture-16

29.04.2014

For end incomplete

4 topic এর মধ্যে 3টি

(i) 3D notation

(ii) 3D projection

(iii) Diffuse reflection

(iv) Ray tracing.

Vertex Angle output

বাস্তুত

প্রক্রিয়া

বিকল্প

($30^\circ, 45^\circ, 0, 360^\circ$)

(45°)

($30^\circ, 60^\circ$)

($30^\circ, 40^\circ, 0.0$)

($600, 40.0, -700.0$)

(-250.0)

($\dots \dots \dots$)

Q derive করতে হবে

$$Q_{dx} = 60$$

$$Q_{dy} = 40$$

$$Q_{dz} = -450$$

Vertex
on sphere

center
of sphere

Light

$k_d \quad i_d$

($\dots \dots$)

($\dots \dots$)

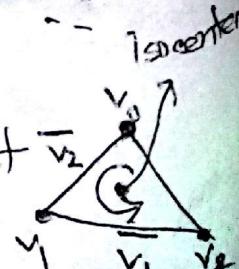
($\dots \dots$)

($\dots \dots$)

\rightarrow

Triangle

$(\frac{v_0}{\dots}), (\frac{v_1}{\dots}), (\frac{v_2}{\dots}) \rightarrow$ নথে অন্তর হবে একে product করে।



Isocenter এর কাছতে হবে। corner point দ্বারা average
করলে সোজ্য থায়। তাঁর আবগে তাই, $\frac{1}{N}$ হালে $\frac{1}{N}$ পর্যন্ত calculate
 $N = \sqrt{v_1} \times \sqrt{v_2} \rightarrow$ then normalize it $\rightarrow N = \frac{1}{INT}$

N এর L normalize করলে হবে, \rightarrow diffuse.

এটা কিন্তু দয়া আবগে আবগানের জন্য হ্রেফে করা হবে
 v_1 এর এবং আবগানের থেকে প্রথমে v_2 [যেখানে
ওটা দীর্ঘ হবে]

i	j	k
v_{1x}	v_{1y}	v_{1z}
v_{2x}	v_{2y}	v_{2z}

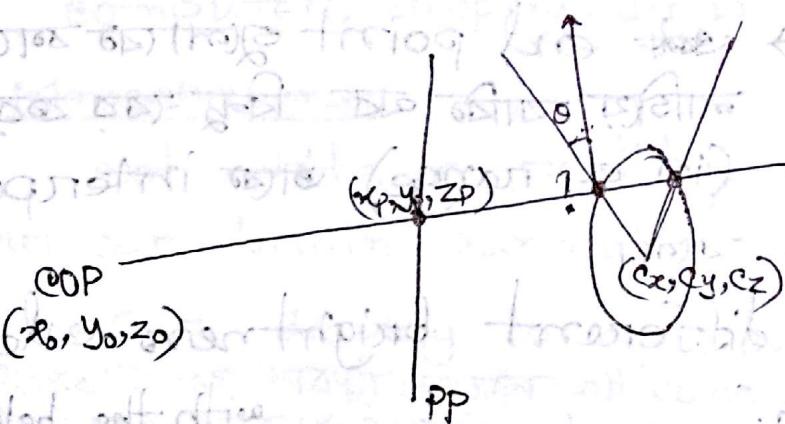
$$v_1 - v_2 = \overline{v_1}$$

$$v_0 - v_1 = \overline{v_2}$$

sphere এর জন্যে $N =$

surface point-center
 $\frac{1}{INT}$

sphere এর মধ্যে $r = |N|$.



$x_p, y_p, z_p \rightarrow$ দয়া আবগে

object এর একটি দয়া আবগে। I_s, K_d দয়া আবগে,

COP

point on the surface \rightarrow 2

$b^2 - 4ac$ calculate

করতে হবে।

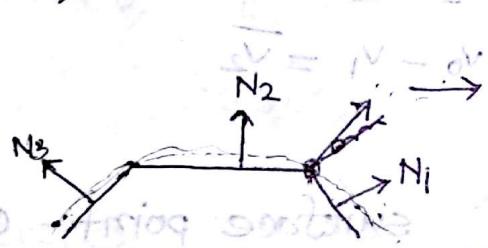
$$\vec{N} = \vec{N}_1 + t \vec{N}_2 \leftarrow \text{t is linear interpolation factor} \rightarrow \vec{N} \times \vec{N} = \vec{N}$$

Flat shading and smooth shading:

Smooth shading

Phong Shading
complexity high
reduces

Gouraud shading



মুক্তি এবং কোম্পনেটিভ নর্মাল
চিনা বা এবং অসম নর্মাল
নর্মাল চিনা করা হয়।

১০০% (১০০% [H] = ০ এবং ১০০%) এবং শেডিং

Interpolation \rightarrow এক এন্ড পয়েন্ট থেকে এক আরেক
লাইনে যাক এবং এই লাইনে এক এন্ড এবং অন্য এন্ড
(in a range), তাকে interpolation
করে।

এভিনি বিভিন্ন অন্তর্ভুক্ত different brightness calculate করে

Gouraud Shading:

মুক্তি এবং অন্তর্ভুক্ত interpolate করে। এবং কোম্পনেটিভ নর্মাল
with the help of n.

I এবং কোম্পনেটিভ interpolate করে। I এবং help নিয়ে,

এবং এবং কোম্পনেটিভ specular shading হলু অর্থাত

মুক্তি space এ shading হলু gouraud.

shading করে করে না। phong shading পাঠে

* Difference between Phong and Gouraud shading

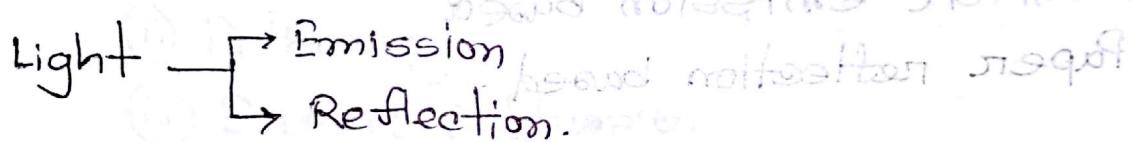
- ~~इसमें~~ same local light model \Rightarrow एक equation के लिए बनाये जाते हैं।
Gouraud प्रोसेस के लिए अलग calculate करते नहीं।

Lecture-17

05.05.2014

Monochrome and colored light:

- आजोवास्तु model किसे source based model.
अधिकतरामें light emission or reflection from other surface ~~is~~ based.



Not for real life example, only used for computer display. It is an illusion of our brain.

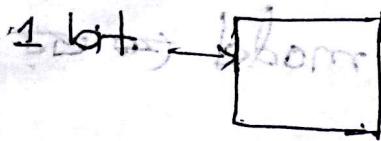
Monochrome:

not white in reality, every color is present equally but our brain cannot differentiate it.

- Our display is approximation based.
- विकास रखें अपना अध्यात्म all color component avg value
नियंत्रकों का जो बनाये हैं separate value बुला mostly same state \rightarrow monochrom
- Our retina has two components:
 - (i) One which senses colors
 - (ii) which senses brightness.

Environment आलोकित ना होने की वज़त त्रुटा थार्ड वा, प्रत्यक्ष परिवर्तन करते हैं अल्टर्नेटिव एक्स्प्रेसन

Bandwidth monitor → example of monochrome.



→ Black
1 → White.

In paper, 0 → White

1 → Black

Pixel based

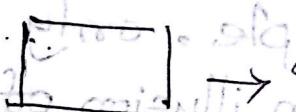
Monitor एवं paper fully opposite

यद्यपि monitor में एक pixel वाले intensity

तत गात्र, काजाजे यह बिंदी दिखा है
light intensity तत रूप।

Monitor emission based

Paper reflection based.



→ 8bit
00 → Black
FF → White.

आजे 0/1 होने वाला level हिल रहा,

So, all levels are grey level.

कठावे 256 से आलादा level आएँ,

Half-tone Approximation.

1 bit data निये प्रिये black/white
तैयार हो जाएगा, उसके लिए अन्य लेवल भी आवश्यक होते
पाएँ।

Print करना जब ये काली बुरबार करना

हो जाके toner होने।

half tone came from printing industry.

1800 DPF

Impact type printer → single bit. So, grey level
के represent करते नाही वा
possible doesn't slight
→ Example - प्रिंटर चल.

Half-toning → प्राकृतिक विभिन्न level के diameter
एवं dot पर illusion create करा।
→ It is a chemical process.

What is Digital halftoning?

3 types.

(i) Rasterizing.

(ii) Dithering.

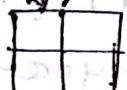
(iii) Error Diffusion.

Output bit single bit image but एकत्री grey

level image के help नियंत्रित single bit image

त्रिविक्षिप्त रूपांक, illusion के help नियंत्रित करा।

Ref.
busi
Image



जो कि इसकी reference image बनवाया करा;
reference image के pattern अद्वितीय

1 bit image त्रिविक्षिप्त करा या reference image एवं
grey image के सम्बन्ध बनाया,
याद 2x2 इसका आकार 4x4 pixel एवं 5x5 pixel.

combination

4 white, 4 black, 2 black 2 white, 3 black

1 white, 1 black 3 white.

Reference image and 1 bit image को normalize
করা হয়।

Reference image এর mask storage কীভূত
average করা হয় multiple thresholding

বক্তু, ref. image 1 bit image
255
5

Mask size এর either the matrix 3 or

the matrix এর size প্রাপ্তি effective reduction
করে থাব। vice versa.

$600 \times 400 \rightarrow 300 \times 200$

no of effective pixel per unit.

Assignment:

Lenna.bmp

512x512 8 bit Gray
2x2

3x3

3x3 window

reference image with enough gray level
image + 10 gray level + 10 gray level.

($544 \times 8 + 542 \times 512 \times 8$) bit // bmp image size
in windows.

OpenGL \rightarrow bitmap

bmp \rightarrow windows format file

\rightarrow OS-9 format

bmp \rightarrow un compressed; but multilayer.

→ single bit / 8 bit / 32 bit / 64 bit हिंदूर
 store करते पाएँ।

→ इन्हें header भाई, 14 byte + 40 byte

की संख्या अंक = 54 bytes.

• bmp info header, bmp file header.

14 → info

40 → file



→ normally single dimensional array.

मिलते हैं slow होते हैं। So, double
2D array हिंदूर किए हैं। img data

2D array के रूप में है।

→ unsigned characters 8 bit store करते
पाएँ।

for (y=0; y < *.Height; y++) {

 for (x=0; x < *.width; x++) {

 glColor3f(Data[y][x], .., ..);

 glVertex2i (x, y);

}

glOrtho (0, *.width, 0, *.height, -1, 1);

new float r, g, b, w; ← r-w ←

glColor3f (r, g, b); ← r-w ←

 r + g = b - w ← w ←

Lecture - 18

07.05.2014

Colored Light:

Emission এ নিজ light করে। So, যে রং ইচ্ছা দেওয়া থাক।

Reflection এ কোরে object Hiimit করতে পারে তখন color absorb করবে।

Reflection based object emit করতে পারে।
কোন রং করে absorb করতে পারবে।

একটি লাল রঙে লাল আলো যে R and B absorb করে।

Monitor লাল আলো যে লাল আলো emit করে।

External light dependency (বেহাল) Emission based.

$$W = R + G + B$$

$R \rightarrow R$. \rightarrow Emission

$R \rightarrow W - (G + B)$ \rightarrow Reflection ray.

আলোর default রং white.

Light color এর উপর depend করে Light কে
২ ভাবে আজ করা যায়।

(i) $R G B$ \rightarrow reflection এর main color

(ii) $C M Y$ \rightarrow emission
Magenta Yellow

Cyan \rightarrow অন্য color থাকে কেবল red color নয়।

Monitor এ Red and green পাঞ্চালিক মিল
yellow করাবে।

Cyan \rightarrow $W - R \rightarrow G + B$.

Magenta \rightarrow $W - G \rightarrow R + B$.

Yellow $\rightarrow W - B = R + G$.

\rightarrow Red + Green
overlap.

মুক্ত অসমর্পণ এবং প্রেরণ base 2024

(i) Light ~~or~~ colors Default environment
এবং color white.

(ii) চিত্র/paper by default color white.

RGB → monitor এর ধরনের color দ্বারা কৃত হয়।
→ additive color

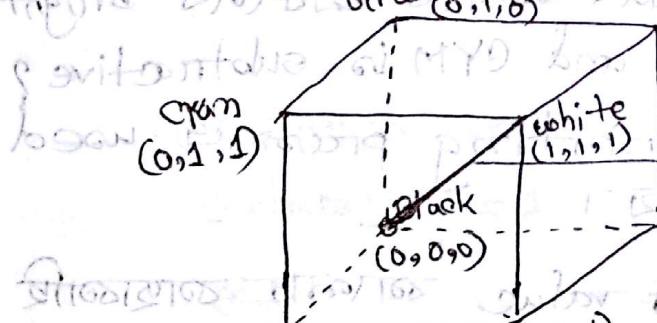
★ even

CMY → subtractive color

emission based কৃত color যোগাযোগ এর environment

External light helps reflection light

blocks emission



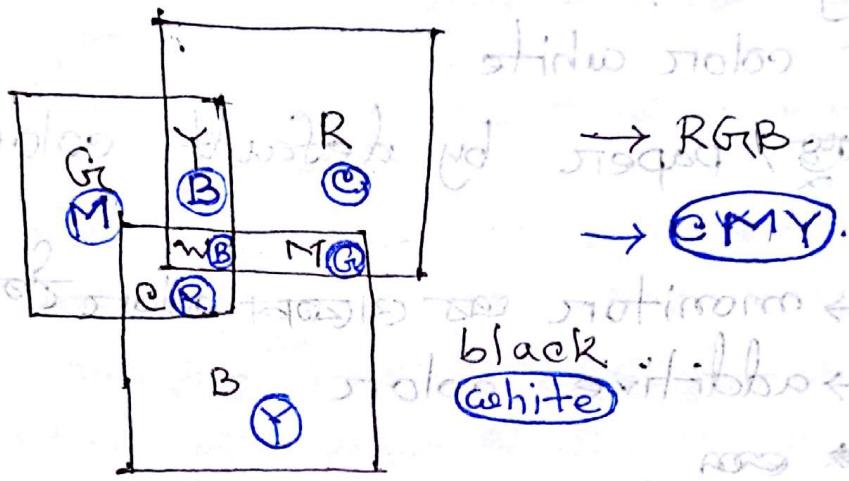
Monochrome Space, special axis, pure black/white.

RGB color cube → representing emission based color model.

Just opposite → CMY → $(1,1,1) \rightarrow$ black → $(0,0,0) \rightarrow$ white.

the pixels are triploid, which means three channels

CMY নিয়ে - RGB এর basic colors এর মতোই
ভাবতে at least রেখা রেখা করা হবে।



color component কে বলি নিয়ে শুধু bright-
ness নিয়ে চিন্তা করলে white, এবং brightness
কে প্রাপ্তি করে আবেগ হয়। additives color অথবা RGB অথবা
এতে যাতে এটি brighter হয়।

~~CMY → color বাটালো effective brightness
করে যায়। that's why subtractive
mode একে white করতে হয়।~~

~~* Why RGB is additive and CMY is subtractive?~~

~~bioque. ১০০% CMYB/CMYK → printing process used
sticker should print colors~~

~~Black~~

~~৫০%~~

~~→ যাতের value কমান্ত রাখতে হবে।~~

~~প্রযুক্তি CMY এর overlapping
করার বদলে black dot মের
হয়।~~

~~→ thresholding মের হয়।~~

~~যাতে ৭৫ থাকলে ১০০ হিসাবে দেখা~~

~~↳ low cost (1.00)~~ → commercial side হয়ে = cost

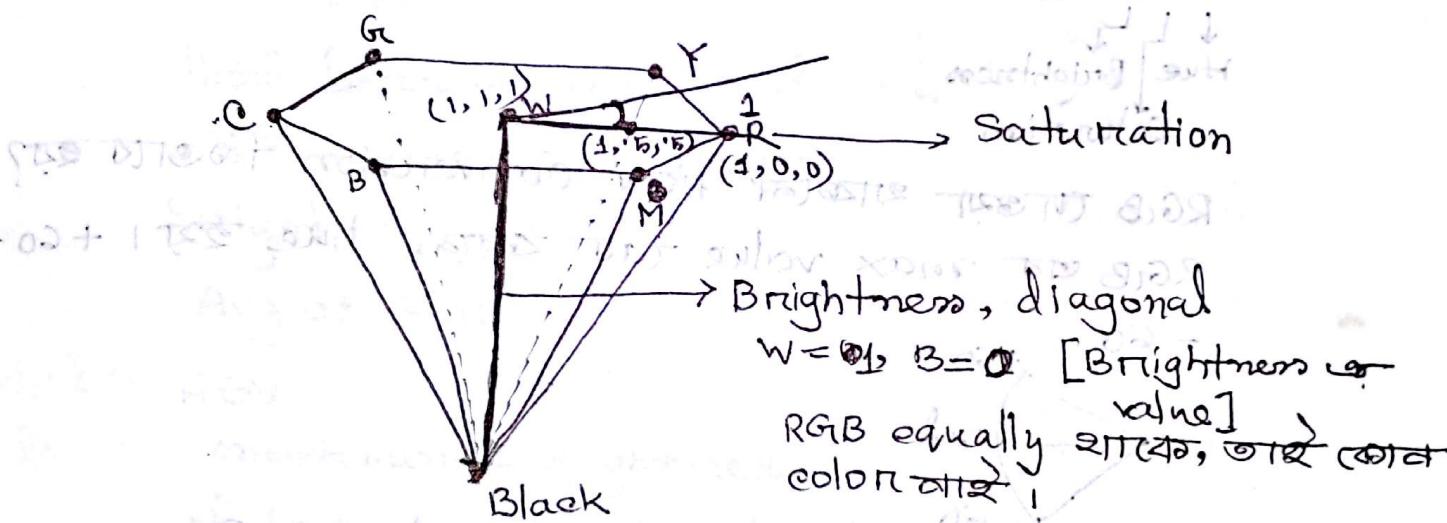
~~sticker < 0.00~~

~~বাটালো এর color এর দাব তৈরি।~~

~~→ কালোর ঘোষণা করলে জ্বাল, bright হয়, quality এ~~

ଯେତୁସେବ ଟୋର୍ମ RGB, ଯାନ୍ତୁ କେବଳ ଏକାକୀକ୍ରମରେ sufficient
 RGBA → RGB alpha. RGB ଲେଖାତ୍ତ ପାଇଁ, sufficient light.
 → software ଦ୍ୱାରା manage ହେଲୁ ହେଲୁ ।
 mat. soft. → transparency ଏକାକୀକ୍ରମରେ ଉପରେ କଥାବିର୍ତ୍ତ କରିବା
 ହେଲୁ ।

HSV/HSB → emission-based system ଏ - ନେ କଥା
 ↓
 Hue Saturation Brightness.



ଏକାକୀ ଥିଲେ ଯତ୍ନରେ ଏହା saturation ଅତିକରି ।

saturation = 1 ହଲେ color ଅତିକରି

କେବଳ ଏହା କଥା ।
 red କେବଳ reference କିମ୍ବା ଏହା angle ଯେତେ hue .

$$\theta = 120^\circ \rightarrow G$$

$$\theta = 240^\circ \rightarrow B$$

$$\theta = 0^\circ \rightarrow R$$

Hue brightness and saturation free .

ତେଣୁ କଥାବିର୍ତ୍ତ କରିବାକୁ କଥାବିର୍ତ୍ତ କରିବାକୁ
 କେବଳ କଥାବିର୍ତ୍ତ କରିବାକୁ ।

So, color based search এর ফল দেখি মে
হ্যাঁ। \rightarrow effective.

It is a ratio based color.

Real life ক্ষেত্রে hue is better than RGB.

Colored Light:

HSV color model

Hue
Brightness
Saturation

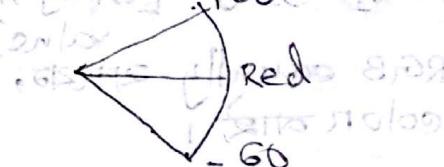
Lecture 19

12.05.2014

RGB থেকে HSV conversion কিভাবে হ্যাঁ?

RGB এর max value টাই করলে hue হ্যাঁ। +60 to

-60.



$G \uparrow \rightarrow$ Green value একটি হলো উপরের দিকে যাতে
 $R \uparrow \rightarrow$ Red value একটি হলো উপরের দিকে যাতে
 $B \downarrow \rightarrow$ Blue value একটি হলো উপরের দিকে যাতে

max value red হলে red dominant, Hue এর

180 range +60 \rightarrow -60, $G=0$

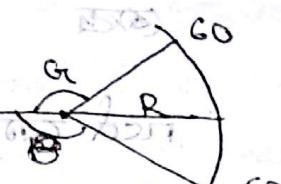
$B=1$ হলে -60° , $B=0$.

$$\text{Hue} = \left\{ \frac{(G-R)}{\text{Max-Min}} \right\} * 60 \quad // 4 \text{ red dominant.}$$

(0.7, 0.3, 0.3)

(0.7, 0.7, 0.3) $\rightarrow 60^\circ$ if red dominant
if green

(0.7, 0.3, 0.7) $\rightarrow -60^\circ$



Green \rightarrow 120 $\rightarrow \pm 60$

Red \rightarrow 0 $\rightarrow \pm 60$

Hue $\rightarrow = \left\{ 2 + \frac{B-R}{Max-Min} \right\} * 60$ // Green dominant.
dominant $\leftarrow 1$

Q কণার 120° আজ্ঞা এক্ষে হয়।

10.7, 0.7, 0.3

Hue $= \left\{ 4 + \frac{R-G}{Max-Min} \right\} * 60$ // Blue dominant.

Saturation:

Diff between max and min

max = min \Rightarrow যাইকোনো

Brightness:

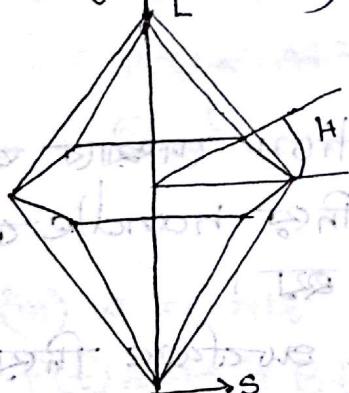
Avg of RGB.

HSV:

to one maximum = brightness

If (0.7, 0.7, 0.7) then brightness = 0.7 as max 0.7

(0.7, 0, 0)



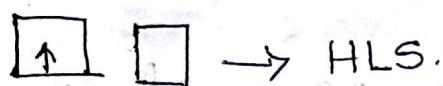
HLS is more realistic

L = 1 \rightarrow যাদ

L = 0 \rightarrow কালো

HSV তে এক্ষেত্রে কিছু গুরুত্ব নাই।



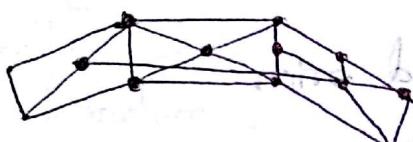


$L \rightarrow$ Luminance.

Curves and surfaces:

Computer Graphics surface rendering করে।

Surface \rightarrow External boundary of any object.



Polygonal surface হাঁ হোক ফ্লেট অসমিষ্ট।

Polygonal mesh:

Shape is very accurate, as corner point defined.

বিন্দু surface smoothing করতে আলোচনা করে এবং এটি বিন্দু পরিসরে প্রযোজন করে।

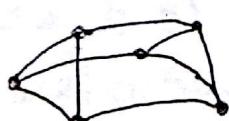
বিন্দু পরিসর কে একটি ক্ষেত্র হিসেবে ধরে তার মধ্যে একটি ক্ষেত্র হিসেবে ধরে।

Curve surface কর্তৃ। solution. দোষার অনেকগুলো।

বিন্দু omit করে কর্তৃপৃষ্ঠী বিন্দু handle করা হয়।

Curve surface পুরু হয়।

অনেক surface কর্তৃ curve surface দিয়ে handle করে।



Cubic curve \rightarrow bicubic surface

2 way curve { surface generate করে, $f(t) = g(t)$

Adv:

($S_A + f_{x_0} + f_{x_1} + f_{x_2} + f_{x_3} = s(t)$)

($S_A + f_{y_0} + f_{y_1} + f_{y_2} + f_{y_3} = t(t)$)

• Polygonal surface নিতে storage আবেক্ষণিক লাভত

• Curve surface এর shape change করা আবেক্ষণিক সহজ

• So, animation এর যুক্তিশয়ু হয়।

• জীবন্ধু অন্যথা করা লাভে।

Curves:

$$|ex A| \cdot |pxz| = |exA|$$

(i). Two points $\rightarrow A \cdot I = f^q$

$$\text{④} \rightarrow D \cdot M \cdot T = f^q$$



$$x = x_0 + t(x_1 - x_0)$$

$$\Rightarrow P(t) = P_0 + t(P_1 - P_0)$$

$$P(t) = At + B \quad \text{①}$$

(ii). Three points \rightarrow $|pxz|$

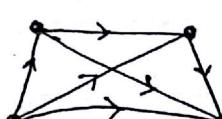
$$P(t) = At^2 + Bt + C \quad \text{②}$$

Quadratic polynomial \rightarrow P_0, P_1, P_2
 In between P_0P_1 and P_1P_2 \rightarrow straight line
 In between P_0P_2 \rightarrow curve.

(iii) 4 points

$$P(t) = At^3 + Bt^2 + Ct + D = 0 \quad \text{③}$$

Cubic polynomial



Viva
 Polynomial single variable
 n multi variable

$$\left. \begin{array}{l} P(t)_x = A_x t^3 + B_x t^2 + C_x t + D_x \\ P(t)_y = A_y t^3 + B_y t^2 + C_y t + D_y \\ P(t)_z = A_z t^3 + B_z t^2 + C_z t + D_z \end{array} \right\} \quad \begin{array}{l} \text{प्राप्ति दिया} \\ \rightarrow \text{प्राप्ति} \\ \text{वही} \end{array}$$

($P_t_x \ P_t_y \ P_t_z$) = $P(t) = \begin{vmatrix} t^3 & t^2 & t & 1 \\ A_x & B_x & C_x & D_x \\ A_y & B_y & C_y & D_y \\ A_z & B_z & C_z & D_z \end{vmatrix}$

$$|4 \times 3| = |1 \times 4| \cdot |4 \times 3|$$

$$P_t = I \cdot A \quad \textcircled{v} \quad // \text{A } 4 \times 12 \text{-t component जिते } 1 \text{ point, } 3D \text{ co-ordinate value. (i)}$$

$$P_t = T \cdot M \cdot G_c \quad \textcircled{vi}$$

$$|1 \times 4| \cdot |4 \times 4| \cdot |4 \times 3|$$

Given Unknown Co-ordinate values.

M = Basis matrix.

G_c = Geometric constraint.

$|4 \times 4|$ के unknown value अनुसार विविध विकल्पों के बारे में विवरण हैं।

(i) Interpolated

(ii) Approximated

(iii) Hybrid

प्राप्ति



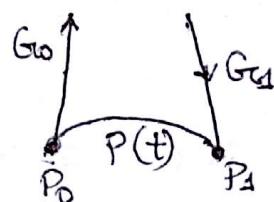
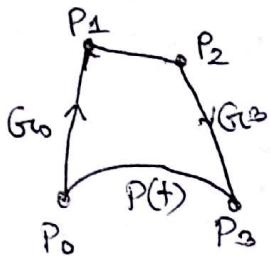
$$0 = t_1 + t_2 + t_3 + t_4 = 4t \quad \text{बहुभागी}$$

प्राप्ति

Lecture-20

14.05.2014

Hermite Curve:



$$\begin{aligned} P_0 &\Rightarrow P_0 \\ P_3 &\Rightarrow P_1 \\ G_0 &\Rightarrow G_0 \\ G_{t3} &\Rightarrow G_1 \end{aligned}$$

$G_0 / G_{t3} / G_1 \rightarrow$ Tangent vector.

$$\frac{d}{dt} P(t) \quad // \text{at zero} \rightarrow G_0 = \frac{d}{dt} P(t)$$

$$// \text{at } 1 \rightarrow G_{t3}/G_1 = \frac{d}{dt} P(t).$$

$$P_t = T \cdot A$$

$$= T \cdot M \cdot G$$

$= TM_H G_H$ // specially 4 hermit curve.

Curve ~~এবং~~ shape maintain ~~এবং~~ geometric constraint.

Bezier $\rightarrow P_0 P_1 P_2 P_3 (?)$

Hermite $\rightarrow P_0 P_3 G_0 G_1$

$$G_H = \begin{vmatrix} P_0 \\ P_1 \\ G_0 \\ G_1 \end{vmatrix} \quad // \text{for hermite}$$

Right side pic

$$G_H = \begin{vmatrix} P_0 \\ P_3 \\ G_0 \\ G_{t3} \end{vmatrix}$$

left side pic

$$P_t = T \cdot M_H \cdot G_H$$

$M_H \rightarrow$ Hermite Basis Matrix

$G_H \rightarrow$ Geometric constraint // $T = 1 t^3 t^2 t 1$
 $// GT = 3t^2 + 2t + 1$

$$P(t)_{t=0} = P_0 = \begin{vmatrix} 1 & 0 & 0 & 1 \end{vmatrix} \cdot M_H \cdot G_{H^*}$$

$$M_H \cdot M \cdot T = (4)^9$$

$$P(t)_{t=1} = P_3 = \begin{vmatrix} 1 & 1 & 1 & 1 \end{vmatrix} \cdot M_H \cdot G_{H^*}$$

$$M_H \cdot M =$$

$$\frac{\delta P(t)}{\delta t_{t=0}} = G_0 = \begin{vmatrix} 0 & 0 & 1 & 0 \end{vmatrix} \cdot M_H \cdot G_{H^*}$$

(initial position)

$$\frac{\delta P(t)}{\delta t_{t=1}} = G_3 = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \cdot M_H \cdot G_{H^*}$$

$$\begin{vmatrix} P_0 \\ P_3 \\ G_0 \\ G_3 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ -1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{vmatrix} \cdot M_H \cdot G_{H^*}$$

Since, $G_{H^*} = \begin{vmatrix} P_0 \\ P_3 \\ G_0 \\ G_3 \end{vmatrix}$

Find direction vector along the path of motion of CHT.

$$M_H = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{vmatrix}^{-1}$$

from page 30 find 10 more terms for

→ Show the inversion

→ Derive the basis matrix of Hermite curve.

$$M_H = \begin{vmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \end{vmatrix}$$

first row is still lost.

→ Basis matrix of Hermite

$$\begin{vmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix}$$

curve contains 7 terms.

still loss 3 terms starting from 4th term on both sides

example 7.17.2

$$P(t) = T \cdot M_H \cdot G_H \quad \text{and} \quad M \cdot \begin{vmatrix} 1 & 0 & 0 & 0 \end{vmatrix} = 0^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$= B_H \cdot G_H \quad \text{and} \quad M \cdot \begin{vmatrix} 1 & 1 & 1 & 1 \end{vmatrix} = 0^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$B_H \rightarrow 1 \times 4$ matrix

$$\text{Blending functions.} \quad \text{and} \quad M \cdot \begin{vmatrix} 0 & 1 & 0 & 0 \end{vmatrix} = 0^T = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$B_H = \begin{vmatrix} t^3 & t^2 & t & 1 \end{vmatrix} \cdot \begin{vmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} = \frac{(t^3)^4}{4!} \cdot \begin{vmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{vmatrix} = \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}$$

$$B_{0H} = 2t^3 - 3t^2 + 1 \quad // 1\text{st component of } B_H \quad P_0$$

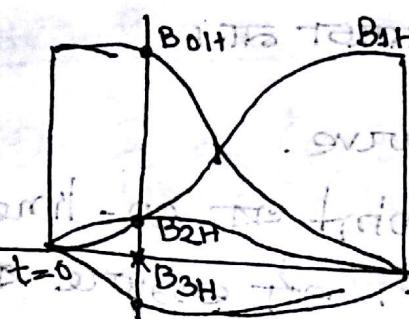
$$B_{1H} = -2t^3 + 3t^2 \quad // 2\text{nd} \quad P_1$$

$$B_{2H} = t^3 - 2t^2 + t$$

$$B_{3H} = t^3 - t^2$$

- This formula is applied here for four points but formula for more than 4 points is not applicable. 4 point curve segments. If 4 points are connected then many points of curve segment. 4 point curve.

- 16 point bi-cubic surface.
- Real life every 16 point cube overlap each other. The patches are joined to form a surface.
- Surface is patchwork. Cubic curve to integrate in real life surface.

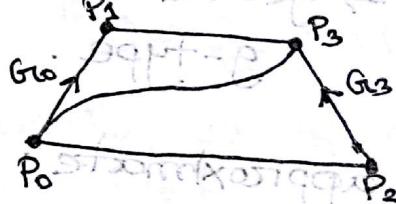


$$B_{1H} = 1 - B_{0H}$$

→ check graph from book.

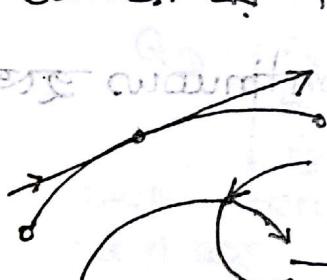
Curve এর special বৈশিষ্ট্য হলো blending function always 1 হয় কখন এবং certain value এতে জন্মে।

function গুরুত্বের সাথে আনন্দের ফলে বিনারি।



Convex Hull

Hull can be both convex or concave. Hermite blending function এর অভিযন্তা always 1 হয় এবং convex এবং প্রিয় পৰিবহন থাকে।



continuous as overlap করে। But 1st derivative different.

জড়া লাগে + 1st derivative different → zero order cont.

" " + 1st " equal → 1st order "

" " + 2nd " → 2nd " curve

Zero order curve কোন কাজ করে নাই

Hermite \rightarrow 1st order cont. curve.

Good work করেছে এটি point এর co-linearity
maintain হতে হব্বে, এটি ensure করব
জন্য এই point overlap হতে হব্বে।

স্পুল্স without problem এর ফলিষ্ঠ লেজে তা সম্ভব

Continuity is of 2 types:

• Control point continuous \rightarrow geometric continuity/
g-type.

• Control point touch করব বা, approximate
 $t=0, t=1$ এ জোড়া লাগে, Given point এ জোড়া কাজ
করব, parametric এ জোড়া লাগে! \rightarrow c-type / parametric

Order same থাকে, এর মধ্যে continuity.

Cubic curve এর বেশি 2nd order continuous হতে পারে।

বেজ করে একটি কোণ করে একটি কোণ করে একটি কোণ করে

বেজ করে একটি কোণ করে একটি কোণ করে একটি কোণ করে

বেজ করে একটি কোণ করে একটি কোণ করে একটি কোণ করে

বেজ করে একটি কোণ করে একটি কোণ করে একটি কোণ করে

বেজ করে একটি কোণ করে একটি কোণ করে একটি কোণ করে

Lecture-21

19.05.2014

Bezier Curve:

$$P(t) = \sum_{i=0}^n B_{in}(t) \cdot P_i \quad (\text{for cubic})$$

curve n = 3, no of control points = 4

Blending functions / control / geometric constraints

For cubic curve,

$$P(t) = \sum_{i=0}^3 B_{i3}(t) \cdot P_i$$

Hermite এবং geometric constraint

$$G_H = \begin{vmatrix} P_0 \\ P_3 \\ G_0 \\ G_3 \end{vmatrix} \rightarrow \text{অন্যান্য } 3D \text{ component}$$

Bezier এবং geometric constraint,

$$G_B = \begin{vmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{vmatrix}$$

যথেন Hermite এবং যারে relate করা হলু ও স্থায়ী গুণ
করা হয় : $G_0 = 3(P_1 - P_0)$, $G_3 = 3(P_3 - P_2)$.

$$P(t) = T \cdot M \cdot G_B$$

$$= T \cdot M_H \cdot G_H \rightarrow \text{For Hermite}$$

$$= T \cdot M_H \cdot G_B \rightarrow \text{Deriving Bezier curve using the help of Hermite curve.}$$

$$G_H = \begin{vmatrix} P_0 \\ P_3 \\ 3(P_1 - P_0) \\ 3(P_3 - P_2) \end{vmatrix}$$

অবস্থা কিছু অভিহৃত

Terminology

$$G_B = \begin{vmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{vmatrix} \quad G_B = G_{CH}$$

$M_{HB} \rightarrow$ makes a relation between
Hermite basis matrix and Bezier.

$$P(t) = T \cdot M_H \cdot M_{HB} \cdot G_B$$

$$= T M_{HB} \cdot G_B$$

Hermite basis matrix $\times M_{HB} =$ Bezier basis matrix.

$$T \cdot \begin{vmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} \rightarrow \text{basis matrix of Bezier, } M_B.$$

\rightarrow blending function एवं 1×4 matrix प्राप्त हो था, या ये

$$= [t^3 \ t^2 \ t \ 1] \begin{vmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix}$$

$$B_{03} = -t^3 + 3t^2 - 3t + 1 = (1-t)^3 \cdot T \rightarrow C_{30}$$

$$B_{13} = 3t^2 - 6t^2 + 3t = 3t(1-t)^2 \cdot T \rightarrow C_{31}$$

$$B_{23} = -3t^3 + 3t^2 - 3t^2 (1-t) \cdot T \rightarrow C_{32}$$

$$B_{33} = t^3 \cdot T \rightarrow C_{33}$$

$$t = A, 1-t = B. \text{ इसे binomial expansion करेंगे!}$$

এখন তাঁ point দ্বারা blending function হিসেবে

$$n \cdot \frac{t^n}{n!} = t^n \cdot \frac{1}{n!}$$

১ Beizer নি general purpose.

২ point \rightarrow straight line.

৩ " \rightarrow quadratic.

৪ " \rightarrow cubic.

৫ " \rightarrow curve.

$$P(t) = \sum_{i=0}^n B_m(t) \cdot P_i \quad \text{geometric constraint}$$

\rightarrow standard equation of Beizer curve
where, n is the degree of the polynomial.

Foley এর মাছে স্বীকৃত হৈ

for $n+1$ points general purpose equation

for beizer curve is, written below where,

$$B = \quad , P = \quad .$$

Blending function can be written as follows,

$$B_m(t) = C_m \cdot t^m \cdot (1-t)^{n-m} \quad \text{①} = \frac{n!}{m!(n-m)!}$$

where, C_m = binomial coefficient,

অঙ্গুলো দ্বারা হৈ $B_{30}, B_{31}, B_{32}, B_{33}$. আংশ
basis matrix দ্বারা নামকৰণ কৰা হৈ by back calculation.

V.V.I
From this two way derivation is very important.

// Bernstein polynomials..

$$Q(t) = (1-t)^3 P_1 + 3t(1-t)^2 P_2 + 3t^2(1-t) P_3 + t^3 P_4$$

$$(1-t)^3 \rightarrow t^3(1-t)$$

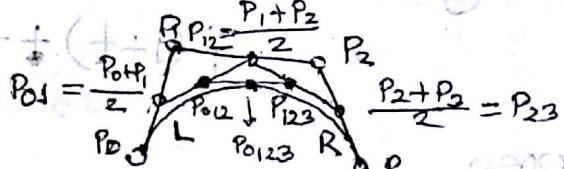
Lecture - 22

Subdivision of curves

$$\text{Here, } P_{012} = P_0 + P_1 + P_2 / 2$$

$$P_{123} = P_1 + P_2 + P_3 / 2$$

$$P_{0123} = P_{012} + P_{123} / 2$$



$$t = 0.5$$

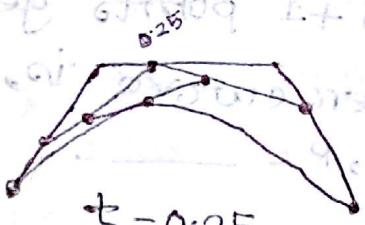
2nd a interpolated but other points approximated. End point given.

Left and right segment derived.

L and R Q1 type continuous as geometric

point connect each other

first derivative equals $P_{012}, P_{0123}, P_{123}$, co-linear.



$t = 0.25$

$$P_{0123} = P_0 + t(P_1 - P_0)$$

$$P_{12} = P_1 + t(P_2 - P_1)$$

$$P_{012} = P_0 + t(P_{12} - P_0)$$

$$= P_0 + t(P_3 - P_0) + t[P_1 + t(P_2 - P_1) - (P_0 + t(P_1 - P_0))]$$

Diagram shows a smooth curve passing through points P0, P1, P2, and P3.

Ex-curves

$$P(t)_L = \frac{1}{8}$$

G_{BL}

$$\left| \begin{array}{cccc} 8 & 0 & 0 & 0 \\ 4 & 4 & 0 & 0 \\ 2 & 4 & 2 & 0 \\ 1 & 3 & 3 & 1 \end{array} \right| \cdot |G_B| \rightarrow \text{at } t=0.5$$

(bottom left) rotation

bottom right rotation & ellipse

$$P(t)_R = \frac{1}{8}$$

G_{BR}

$$\left| \begin{array}{cccc} 1 & 3 & 3 & 1 \\ 0 & 2 & 4 & 2 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 8 \end{array} \right| \cdot |G_B|$$

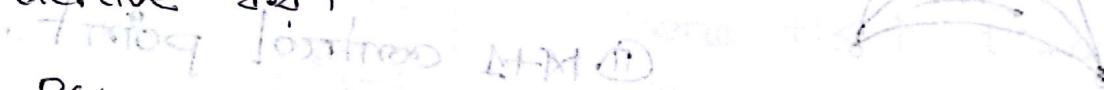
(bottom right) smiley ←

bridge planned

break door, bridge

Assignment:

* t based curve, अर्थात् left and right curve derive कर।



$$\therefore P(t) = t \cdot M(G)$$

∴ P(t) = t $\begin{pmatrix} 1 & 3 & 3 & 1 \\ 0 & 2 & 4 & 2 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix}$

∴ P(t) = $t \begin{pmatrix} 1 & 3 & 3 & 1 \\ 0 & 2 & 4 & 2 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix}$



∴ P(t) = $t \begin{pmatrix} 1 & 3 & 3 & 1 \\ 0 & 2 & 4 & 2 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix}$

∴ P(t) = $t \begin{pmatrix} 1 & 3 & 3 & 1 \\ 0 & 2 & 4 & 2 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 8 \end{pmatrix}$

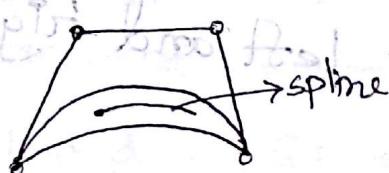
Lecture-23

Spline:

→ Natural (Interpolated)

→ Spline (Approximated)

Generally spline হলো approximated. control point কে touch করতে বা,



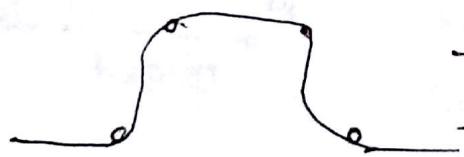
① C2 continuous. ২য় ও ৩য় derivative যান্ত হবে।

② M+1 control point.

M-2 curve + segments.

M > 3

4 point এর spline হলো minimum. এর মিচ হচ্ছে
পার্ট বা। যদি এখন এর C2 continuous. double deri-
vation করলে zero হওয়া যেতে পারে। simplest spline



→ Spline এর example.

→ যান এবেটা material যাকে আজ
কষ্ট যায় না, bend / গুঁগানো
থায় তাকে সেবকে দিকে লাজাই।
Like - spring.

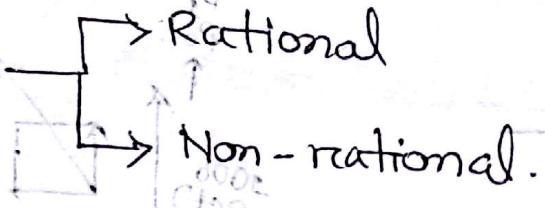
Approximated → Spline

→ B-spline (Basis matrix আছে)

Spline → Uniform (+ এর step uniform)

→ Non-uniform (+ non-uniform)

uniform \rightarrow Rational



Non-uniform \rightarrow Non-uniform rational B-spline

NURBS \rightarrow Non uniform rational B-spline
 \rightarrow most popular

B-spline: bring lot of advantages

$$P(t)_i = T \cdot M \cdot G$$

$$G_{BS} = \begin{cases} P_{i-2,3} & i \geq M+3 \\ P_{i-2} & \\ P_{i-1} & \\ P_i & \end{cases}$$

$i \geq M+3$.

Here $t_i \leq t \leq t_{i+1}$.

3rd overlap point হাতে।

10 segments point এর segment হাতে
total 13 points হিয়ার
treat হয়।

Blending function 1 এর মধ্যে উচ্চ বা, always
convex hull এর ভিত্তি থাকে,

~~Definition of spline~~] Details, জানাব দেবে আর,
Classification " "

* Spline \rightarrow রেখার নাম, সৃষ্টি হবে,

* B-spline \rightarrow How to draw?

cubic spline এর basis matrix এর derivation

পুরোগত নাম
প্রাচীন পদের স্বরূপ

due to given-id গুরুত নিত বা form 0.8

চুক্তি করা হবে কুর্ব এর অঙ্কন ও

কুর্ব করণ এর একটি সহজ এবং সুলভ

সুস্থিত পদ্ধতি। একটি কুর্ব এবং একটি সুস্থিত

Bi-cubic surfaces:

$$P(s, t) =$$

→ surface & location mapping
2 stage: 2x cubic surface

So, total 16 control points

Surface orientation → surface normal

normal जाना पर्योजन
बाहर का shading करना चाहिए

Complexity $1000 \times 1000 \rightarrow 1M$ point के deal

complex problematics
2 ways to reduce complexity:

Complexity avoid करना चाहिए $10 \times 10 = 100$ point

दिये गए calculate करना है। फिर उसे difference करके आवर्त अर्थात् छुलों के

polygon fill करके draw करना है। इसे दो part के divide करके patch करें।



100वां ब्रेकपॉइन्ट वाले draw करने के लिए यहाँ का patch छुलों के smooth shading करके approximate version पाउँगा याहूँ।

So, most of the cases bi-cubic के sub-cubic patch ए तरफ करके draw करें।

spline base surface ए 16 point दिये cubic curve आवर्त चाहिए। ये थार्ड surface पाउँगा याहूँ।

Some notes for viva:

(i) G^0 geometric continuity \rightarrow

If two curve segments join together.

(ii) G^1 geometric continuity \rightarrow

• If the directions (but not necessarily the magnitudes) of the two ^{curve} segments' tangent vectors are equal at join point.

• For two tangent vectors TV_1 and TV_2 to have same direction it is necessary that one be a scalar multiple of the other; that is,

$$TV_1 = k \cdot TV_2 \text{ with } k > 0.$$

• Required in computer aided design of objects.

(iii) C^1 continuous \rightarrow

• If the tangent vectors of two cubic curve segments are equal (i.e. their directions and magnitudes are equal) at the segments' joint point the curve has a first degree continuity in the parameter t .

3 major type of curves: ~~can't~~ can't custom shape #

(i) Heronite

(ii) Bezier

(iii) Spline

Heronite: Two endpoints and two endpoint tangent vectors [↔] (i)

Bezier: Two endpoints and two other points that control the endpoint tangent vectors [↔] (ii)

Spline: Four control points [↔] (iii)

Diff between global and local light model:

Deriving the basis matrix of Hermite curve:

A parametric cubic curves is defined by,

$$Q(t) = T \cdot M \cdot G_t$$

Here, $T =$ for cubic curves,
 $\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix}$
 $M = 4 \times 4$ basis matrix
 $\text{and } ST = 3t^2 - 2t + 1$

$G_t =$ four element column vector
 of geometric constraints,
 called geometry vectors.

For Hermite,

$$Q(t) = TM_H G_{tH}$$

where, $G_{tH} = \begin{bmatrix} P_1 \\ P_4 \\ G_1 \\ G_4 \end{bmatrix}$

Restricting t in interval $[0, 1]$ we get,

$$Q(t)_{t=0} \Rightarrow P_1 = \begin{vmatrix} 1 & 0 & 0 & 1 \end{vmatrix} \cdot M_H \cdot G_{tH}$$

$$Q(t)_{t=1} \Rightarrow P_4 = \begin{vmatrix} 1 & 1 & 1 & 1 \end{vmatrix} \cdot M_H \cdot G_{tH}$$

$$\frac{\delta Q(t)}{\delta t} \Big|_{t=0} \Rightarrow G_1 = \begin{vmatrix} 0 & 0 & 1 & 0 \end{vmatrix} M_H G_H$$

$$\frac{\delta Q(t)}{\delta t} \Big|_{t=1} \Rightarrow G_4 = \begin{vmatrix} 3 & 2 & 1 & 0 \end{vmatrix} M_H G_H$$

So, we can write,

$$\begin{bmatrix} P_1 \\ P_4 \\ G_1 \\ G_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \cdot M_H \cdot G_H$$

Since, $G_H = \begin{bmatrix} P_1 \\ P_4 \\ G_1 \\ G_4 \end{bmatrix}$

$$M_H = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}^{-1}$$

$$|M_H| = -1 \begin{vmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 3 & 2 & 1 \end{vmatrix}$$

$$= - \{ 1(0-2) - 1(0-3) + 1(0-0) \}$$

$$= - \{ -2 + 3 \} = -1.$$

$$A_{11} = \begin{vmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 2 & 1 & 0 \end{vmatrix} = -(-2) = -2.$$

$$A_{12} = (-1) \begin{vmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 3 & 1 & 0 \end{vmatrix} = 3$$

$$A_{13} = \begin{vmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 3 & 2 & 0 \end{vmatrix} = 0$$

$$A_{14} = (-1) \begin{vmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 3 & 2 & 1 \end{vmatrix} = -1$$

$$A_{21} = (-1) \begin{vmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 2 & 1 & 0 \end{vmatrix} = -2$$

$$A_{22} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 3 & 1 & 0 \end{vmatrix} = -3$$

$$A_{23} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 3 & 2 & 0 \end{vmatrix} \stackrel{(-1)}{=} 0.$$

$$A_{24} = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 3 & 2 & 1 \end{vmatrix} = 0$$

$$A_{31} = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 2 & 1 & 0 \end{vmatrix} = 1(1-2) = -1$$

$$A_{32} = (-1) \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 3 & 1 & 0 \end{vmatrix} = -(1-3) = 2$$

$$A_{33} = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 3 & 2 & 0 \end{vmatrix} = 2-3 = -1$$

$$A_{34} = (-1) \begin{vmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 3 & 2 & 1 \end{vmatrix} = 0.$$

$$A_{41} = (-1) \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{vmatrix} = -1(1-0) = -1$$

$$A_{42} = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{vmatrix} = 1$$

$$A_{43} = (-1) \begin{vmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 3 & 2 & 0 \end{vmatrix} = 0.$$

$$A_{44} = 0.$$

$$\text{Adj. matrix} = \begin{bmatrix} -2 & 3 & 0 & -1 \\ 2 & -3 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & 1 & 0 & 0 \end{bmatrix}^T$$

$$= \begin{bmatrix} -2 & 2 & -1 & -1 \\ 3 & -3 & 2 & 1 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$\therefore M_H = \frac{1}{-1} \begin{bmatrix} -2 & 2 & -1 & -1 \\ 3 & -3 & 2 & 1 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$