9/29/2020

From 9/24

Moving to CG10 and above
Theory
OGL Transformations

Geometry
Linear space
Affine space
Affine homogenous coordinates – in these coordinates all of OGL transformations can be implemented
Using matrix multiplication 4x4 by 4x4 or 4x4 by 4x1
Coordinate system
3-D coordinates
Volume

Linear Vector Space
Scalar Operations

1) Scalar Operations are real number operations (addition, subtraction, multiplication)

Vectors
Direction
Magnitude
No – location

Vector by Vector Operations

2) Addition / Subtraction
Cross product

Vector by Scalar Operations

3) Multiplication

Slide ch10-7

aV a is a scalar and V is a vector9multiplication)
When $a = -1$ $a * -V$ ➔ -V Type equation here.

$U - V == U + -V$ vector subtraction

Slide 8 Linear Vector Space

Scale and Rotate via matrix multiplication, not possible to implement translation via mutrix multiplication.

v=u+2w-3r

Slide 10

Points denote location
Point Vector operations.

Point + Vector yields a point
Point – Point yields a vector

Affine Space
- Vector Space
- Plus Points (abstract concept)
- Point vector operation (addition) ➔ point
- Point to point (subtraction) ➔ Vector

Slide 11

Ambiguity

Vector (1,1)
Vector(1, 1, 1)

Point (1,1) (1, 1, 1)

(1, 1) point or vector?

(2, 2, ,2, 1) – point
(2, ,2, 2, 0) Vector might denote light source at infinity

In 3D origin is at (0, 0, 0, 1) == P

Consider Q at (1, 2, 3, 1)

Moved to 4D the last component can be used to **distinguish** points and vectors
Enables to use matrix multiplication for all transformations

Affine space Slide 11

Definition of a line 12

Consider P_0 add aV to P_0

Assume $0 \le a \le \infty$
The collection of vectors $P_0 + aV \;\; 0 \le a \le \infty$ → ray
Vectors through P_0 in the direction of V

Assume -∞ $\leq a \leq$ ∞
The collection of vectors $P_0 + aV$ -∞ $\leq a \leq$ ∞ → Vectors through P_0 in the direction of $\pm V$

Consider P_0 = (x_0, y_0, z_0, 1) , P_1 = (x_1, y_1, z_1, 1)

$P_1 - P_0$ provides a vector call it U

$P_0 + aU$ = $P_0 + a \times (P_1 - P_0)$

When $0 \leq a \leq 1$ we get the line from P_0 to P_1

When $a = 0$ we get P_0 When a is 1 we get P_1

What is the explicit definition of the line from P_0 to P_1

The line is given by $y = mx + b$ where m and b are determined by the point coordinates

Parametric definition of the line.
Slide 13 provides the parametric equation

Every point P on the line is given by $\begin{bmatrix} \mathrm{x(a)} \\ \mathrm{y(a)} \\ \mathrm{z(a)} \end{bmatrix}$ $([x(a), y(a)]^T)$ = $\begin{bmatrix} \mathrm{a}x_0 + (1-a)x_1 \\ \mathrm{a}y_0 + (1-a)y_1 \\ \mathrm{a}z_0 + (1-a)z_1 \end{bmatrix}$

How would we draw this given P_0 = (x_0, y_0, z_0, 1) , P_1 = (x_1, y_1, z_1, 1)

```
glBegin(points)
loop on a from 0 to 1

glVertex(aX0 + (1-a)X1, aY0 + (1-a)Y1)


glend()
glFlush()
```

9/29

Explicit y=f(x) z=f(x, y) Draw using Loop on x [y]
Implicit f(x, y) = 0 , f(x, y, z) = 0 to draw convert to explicit or parametric
Parametric:

$$p(a) = \begin{bmatrix} \mathrm{x(a)} \\ \mathrm{y(a)} \end{bmatrix}$$

$$p(a,b) = \begin{bmatrix} \mathrm{x(a,b)} \\ \mathrm{y(a,b)} \\ \mathrm{z(a,b)} \end{bmatrix}$$

Loop on a, b (generated on the fly - iterations, recursion), comes from a user, comes from a data structure

None is available? Use parametric approximation (or interpolation) of lines and curves

Parametric equation of a line

$x(a) = ax_0 + (1-a)x_1$
$y(a) = ay_0 + (1-a)y_1$

$$0 \leq a \leq 1$$

Convexity (slide 16)

A polygon T is convex if

$\forall \{P \in T, Q \in T\}$ the line $\overline{PQ} \in T$
the line For every pair of point P, Q, which are in the polygon ➔ the entire line [P,Q]
Is on the polygon.

A polygon is Star convex
$\exists\, P \in T$ so that $\forall\, Q \in T$ the line $\overline{PQ} \in T$
If the exist at least one-point P inside so that for every other point inside the line PQ is inside

OpenGL requires that your polygons are convex
What if not? Undefined.

A triangle is convex, simple, planner ➔ preferred type of polygon.
Complex polygon – tiled with triangles manually automatically.

Convex Hull
Given $P_1, P_2, \ldots, P_n$ find the minimal convex polygon that contains all the points.

Back later

Slide 19 , curves, surfaces

A plan is defined by a point and two vectors or by three points.
Slide 20 Parametric equation of a plan

Slide 21 parametric equation of a triangle from scratch.

Consider an arbitrary point $S(a)$ between P and Q
Define $S(a)$ using the parametric equation of the line $\overline{PQ}$

The collection of lines R to $S(a)$ contains all the points
The point $R - S(a)$ obtained by the parametric equation $T(a, b)$
Back later.

Normals
The normal to a plan is important for lighting decisions

Three ways to find it:

1) Directly from the plan equation
2) Using dot product
3) Using Cross product

Slide 22.

Check Shoaff (WDS) under resources,

CG11 Representation

Representing Vector spaces, affine spaces,
Coordinate Systems, Volumes

Linear independence defines in slide 11-3.

The dimension of a linear (affine space) is the minimal number of
Linearly independent vectors in the space.

The intuitive 3D space where we "live"
The $(\hat{x}, \hat{y}, \hat{z})$ unit vectors → any other vector can be defined as a linear combination of $(\hat{x}, \hat{y}, \hat{z})$

Coordinate system
E.g., defined by $(\hat{x}, \hat{y}, \hat{z})$

A Frame
A Frame is a coordinate system with an origin

We add one more dimension to get homogeneous coordinates.

Here:

Consider:

v = $[a_1\ a_2\ a_3\ 0]^T$

p = $[b_1\ b_2\ b_3\ W]^T$

If W == 0 it is a vector

If W <> 0 it is point (vertex)

If W==1 it is a normalized point

Reducing Ambiguity

In Affine Homogeneous frame ever OGL transformation can be done using matrix multiplication.

GPU