



Tamara Munzner

### Math Review

Week 1, Wed Jan 6

<http://www.ugrad.cs.ubc.ca/~cs314/vjan2010>

### News

- no class this Friday (Jan 8)!
- UBC CS Dept news

#### Events this week

##### How to Prepare for the Tech Career Fair

Date: Wed. Jan 6  
Time: 5 – 6:30 pm  
Location: DMP 110

##### Resume Writing Workshop (for non-coop students)

Date: Thurs. Jan 7  
Time: 12:30 – 2 pm  
Location: DMP 201

##### CSSS Movie Night

Date: Thurs. Jan 7  
Time: 6 – 10 pm  
Location: DMP 310  
Movies: "Up" & "The Hangover"  
(Free Popcorn & Pop)

#### Drop-In Resume Edition Session

Date: Mon. Jan 11  
Time: 11 am – 2 pm  
Location: Rm 255, ICICS/CS Bldg

#### Industry Panel

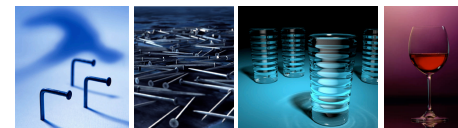
Speakers: Managers from Google, IBM, Microsoft, TELUS, etc.  
Date: Tues. Jan 12  
Time: Panel: 5:15 – 6:15 pm;  
Networking: 6:15 – 7:15 pm  
Location: Panel: DMP 110;  
Networking: X-wing Undergrad Lounge

#### Tech Career Fair

Date: Wed. Jan 13  
Time: 10 am – 4 pm  
Location: SUB Ballroom

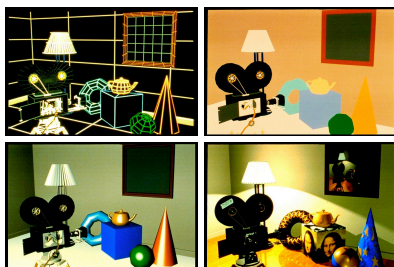
### Review: Computer Graphics Defined

- CG uses
  - movies, games, art/design, ads, VR, visualization
- CG state of the art
  - photorealism achievable (in some cases)



<http://www.alias.com/eng/etc/fakeorfoto/quiz.html>

### Review: Rendering Capabilities



[www.siggraph.org/education/materials/HyperGraph/shutbug.htm](http://www.siggraph.org/education/materials/HyperGraph/shutbug.htm)

### Today's Readings

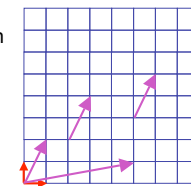
- FCG Chapter 2: Miscellaneous Math
  - except 2.7 (2.11 in 2nd edition)
- FCG Chapter 5: Linear Algebra
  - except 5.4 (not in 2nd edition)

### Notation: Scalars, Vectors, Matrices

- scalar  $a$ 
  - (lower case, italic)
- vector  $\mathbf{a} = [a_1 \ a_2 \ \dots \ a_n]$ 
  - (lower case, bold)
- matrix  $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$ 
  - (upper case, bold)

### Vectors

- arrow: length and direction
  - oriented segment in nD space
- offset / displacement
  - location if given origin



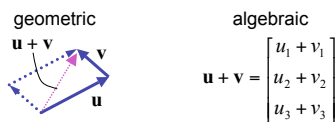
### Column vs. Row Vectors

- row vectors  $\mathbf{a}_{row} = [a_1 \ a_2 \ \dots \ a_n]$
- column vectors  $\mathbf{a}_{col} = \begin{bmatrix} a_1 \\ a_2 \\ \dots \\ a_n \end{bmatrix}$
- switch back and forth with transpose

$$\mathbf{a}_{col}^T = \mathbf{a}_{row}$$

### Vector-Vector Addition

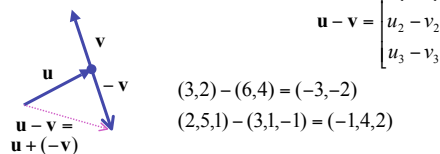
- add: vector + vector = vector
- parallelogram rule
  - tail to head, complete the triangle



examples:  $(3,2) + (6,4) = (9,6)$   
 $(2,5,1) + (3,1,-1) = (5,6,0)$

### Vector-Vector Subtraction

- subtract: vector - vector = vector

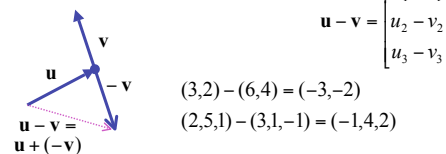


$$\mathbf{u} - \mathbf{v} = \begin{bmatrix} u_1 - v_1 \\ u_2 - v_2 \\ u_3 - v_3 \end{bmatrix}$$

$(3,2) - (6,4) = (-3,-2)$   
 $(2,5,1) - (3,1,-1) = (-1,4,2)$

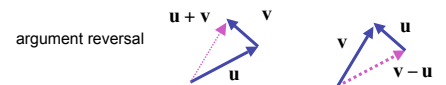
### Vector-Vector Subtraction

- subtract: vector - vector = vector



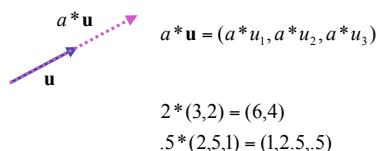
$$\mathbf{u} - \mathbf{v} = \begin{bmatrix} u_1 - v_1 \\ u_2 - v_2 \\ u_3 - v_3 \end{bmatrix}$$

$(3,2) - (6,4) = (-3,-2)$   
 $(2,5,1) - (3,1,-1) = (-1,4,2)$



### Scalar-Vector Multiplication

- multiply: scalar \* vector = vector
  - vector is scaled



$$a * \mathbf{u} = (a * u_1, a * u_2, a * u_3)$$

$2 * (3,2) = (6,4)$

$.5 * (2,5,1) = (1,2.5,.5)$

### Vector-Vector Multiplication

- multiply: vector \* vector = scalar
- dot product, aka inner product  $\mathbf{u} \bullet \mathbf{v}$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \bullet \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = (u_1 * v_1) + (u_2 * v_2) + (u_3 * v_3)$$

### Vector-Vector Multiplication

- multiply: vector \* vector = scalar
- dot product, aka inner product  $\mathbf{u} \bullet \mathbf{v}$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \bullet \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = (u_1 * v_1) + (u_2 * v_2) + (u_3 * v_3)$$

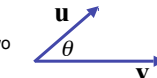
### Vector-Vector Multiplication

- multiply: vector \* vector = scalar
- dot product, aka inner product  $\mathbf{u} \bullet \mathbf{v}$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \bullet \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = (u_1 * v_1) + (u_2 * v_2) + (u_3 * v_3)$$

- geometric interpretation

- lengths, angles
- can find angle between two vectors



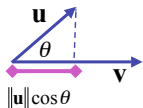
$$\mathbf{u} \bullet \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos \theta$$

## Dot Product Geometry

- can find length of projection of  $u$  onto  $v$

$$\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos \theta$$

$$\|\mathbf{u}\| \cos \theta = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{v}\|}$$



- as lines become perpendicular,  $\mathbf{u} \cdot \mathbf{v} \rightarrow 0$



18

## Dot Product Example

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = (u_1 * v_1) + (u_2 * v_2) + (u_3 * v_3)$$

$$\begin{bmatrix} 6 \\ 1 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 7 \\ 3 \end{bmatrix} = (6*1) + (1*7) + (2*3) = 6 + 7 + 6 = 19$$

18

## Vector-Vector Multiplication, The Sequel

- multiply: vector \* vector = vector
- cross product
  - algebraic

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \times \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{bmatrix}$$

19

## Vector-Vector Multiplication, The Sequel

- multiply: vector \* vector = vector
- cross product
  - algebraic

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \times \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{bmatrix}$$

20

## Vector-Vector Multiplication, The Sequel

- multiply: vector \* vector = vector
- cross product
  - algebraic

$$\begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} \times \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{bmatrix}$$

blah blah

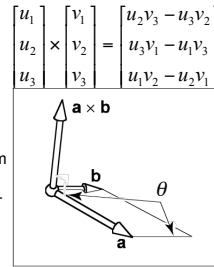
21

## Vector-Vector Multiplication, The Sequel

- multiply: vector \* vector = vector
- cross product
  - algebraic
- geometric

$$\|\mathbf{a} \times \mathbf{b}\| = \|\mathbf{a}\| \|\mathbf{b}\| \sin \theta$$

- $\|\mathbf{a} \times \mathbf{b}\|$  parallelogram area
- $\mathbf{a} \times \mathbf{b}$  perpendicular to parallelogram



22

## RHS vs. LHS Coordinate Systems

- right-handed coordinate system **convention**



right hand rule:  
index finger x, second finger y;  
right thumb points up  
 $\mathbf{z} = \mathbf{x} \times \mathbf{y}$

- left-handed coordinate system



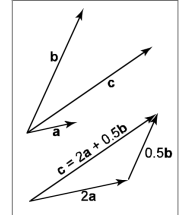
left hand rule:  
index finger x, second finger y;  
left thumb points down  
 $\mathbf{z} = \mathbf{x} \times \mathbf{y}$

23

## Basis Vectors

- take any two vectors that are **linearly independent** (nonzero and nonparallel)
  - can use linear combination of these to define any other vector:

$$\mathbf{c} = w_1 \mathbf{a} + w_2 \mathbf{b}$$



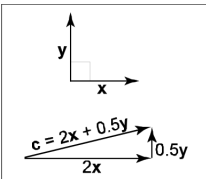
24

## Orthonormal Basis Vectors

- if basis vectors are **orthonormal** (orthogonal (mutually perpendicular) and unit length)
  - we have Cartesian coordinate system
  - familiar Pythagorean definition of distance

orthonormal algebraic properties

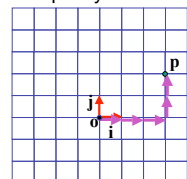
$$\|\mathbf{x}\| = \|\mathbf{y}\| = 1, \\ \mathbf{x} \cdot \mathbf{y} = 0$$



25

## Basis Vectors and Origins

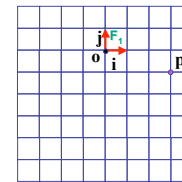
- coordinate system**: just basis vectors
  - can only specify offset: vectors
- coordinate frame**: basis vectors and origin
  - can specify location as well as offset: points



$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

26

## Working with Frames

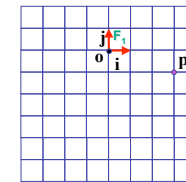


$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$

27

## Working with Frames

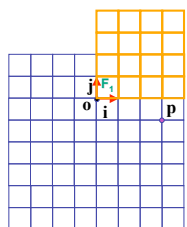


$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$   $\mathbf{p} = (3, -1)$

28

## Working with Frames

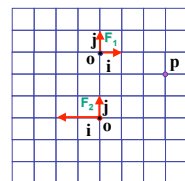


$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$   $\mathbf{p} = (3, -1)$

29

## Working with Frames



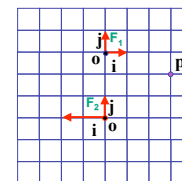
$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$   $\mathbf{p} = (3, -1)$

$\mathbf{F}_2$

30

## Working with Frames



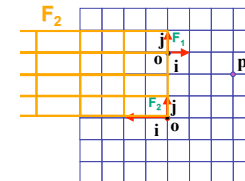
$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$   $\mathbf{p} = (3, -1)$

$\mathbf{F}_2$   $\mathbf{p} = (-1.5, 2)$

31

## Working with Frames



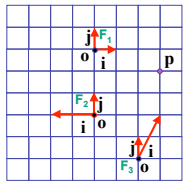
$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$\mathbf{F}_1$   $\mathbf{p} = (3, -1)$

$\mathbf{F}_2$   $\mathbf{p} = (-1.5, 2)$

32

## Working with Frames



$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

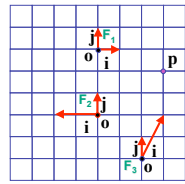
$$\mathbf{F}_1 \quad \mathbf{p} = (3, -1)$$

$$\mathbf{F}_2 \quad \mathbf{p} = (-1.5, 2)$$

$$\mathbf{F}_3$$

33

## Working with Frames



$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

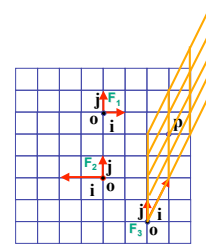
$$\mathbf{F}_1 \quad \mathbf{p} = (3, -1)$$

$$\mathbf{F}_2 \quad \mathbf{p} = (-1.5, 2)$$

$$\mathbf{F}_3 \quad \mathbf{p} = (1, 2)$$

34

## Working with Frames



$$\mathbf{p} = \mathbf{o} + x\mathbf{i} + y\mathbf{j}$$

$$\mathbf{F}_1 \quad \mathbf{p} = (3, -1)$$

$$\mathbf{F}_2 \quad \mathbf{p} = (-1.5, 2)$$

$$\mathbf{F}_3 \quad \mathbf{p} = (1, 2)$$

35

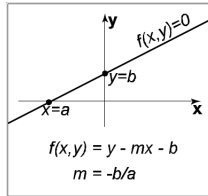
## Named Coordinate Frames

- origin and basis vectors  $\mathbf{p} = \mathbf{o} + ax + by + cz$
- pick canonical frame of reference
  - then don't have to store origin, basis vectors
  - just  $\mathbf{p} = (a, b, c)$
  - convention: Cartesian orthonormal one on previous slide
- handy to specify others as needed
  - airplane nose, looking over your shoulder, ...
  - really common ones given names in CG
    - object, world, camera, screen, ...

36

## Lines

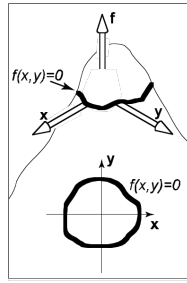
- slope-intercept form
  - $y = mx + b$
- implicit form
  - $y - mx - b = 0$
  - $Ax + By + C = 0$
  - $f(x, y) = 0$



37

## Implicit Functions

- find where function is 0
  - plug in (x, y), check if
    - 0: on line
    - < 0: inside
    - > 0: outside
- analogy: terrain
  - sea level:  $f=0$
  - altitude: function value
  - topo map: equal-value contours (level sets)



38

## Implicit Circles

- $f(x, y) = (x - x_c)^2 + (y - y_c)^2 - r^2$ 
  - circle is points (x, y) where  $f(x, y) = 0$
- $\mathbf{p} = (x, y), \mathbf{c} = (x_c, y_c) : (\mathbf{p} - \mathbf{c}) \cdot (\mathbf{p} - \mathbf{c}) - r^2 = 0$ 
  - points  $\mathbf{p}$  on circle have property that vector from  $\mathbf{c}$  to  $\mathbf{p}$  dotted with itself has value  $r^2$
- $\|\mathbf{p} - \mathbf{c}\|^2 - r^2 = 0$ 
  - points points  $\mathbf{p}$  on the circle have property that squared distance from  $\mathbf{c}$  to  $\mathbf{p}$  is  $r^2$
- $\|\mathbf{p} - \mathbf{c}\| - r = 0$ 
  - points  $\mathbf{p}$  on circle are those a distance  $r$  from center point  $\mathbf{c}$

39

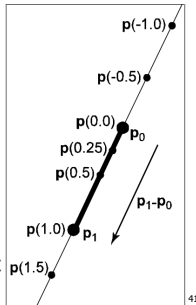
## Parametric Curves

- parameter: index that changes continuously
  - (x, y): point on curve  $\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} g(t) \\ h(t) \end{bmatrix}$
  - t: parameter
- vector form
  - $\mathbf{p} = f(t)$

40

## 2D Parametric Lines

- $\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 + t(x_1 - x_0) \\ y_0 + t(y_1 - y_0) \end{bmatrix}$
- $\mathbf{p}(t) = \mathbf{p}_0 + t(\mathbf{p}_1 - \mathbf{p}_0)$
- $\mathbf{p}(t) = \mathbf{o} + t(\mathbf{d})$
- start at point  $\mathbf{p}_0$ , go towards  $\mathbf{p}_1$ , according to parameter t
  - $\mathbf{p}(0) = \mathbf{p}_0, \mathbf{p}(1) = \mathbf{p}_1$



41

## Linear Interpolation

- parametric line is example of general concept
  - $\mathbf{p}(t) = \mathbf{p}_0 + t(\mathbf{p}_1 - \mathbf{p}_0)$
  - interpolation
    - $\mathbf{p}$  goes through  $\mathbf{a}$  at  $t = 0$
    - $\mathbf{p}$  goes through  $\mathbf{b}$  at  $t = 1$
  - linear
    - weights  $t, (1-t)$  are linear polynomials in  $t$

42

## Matrix-Matrix Addition

- add: matrix + matrix = matrix

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} + \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} n_{11} + m_{11} & n_{12} + m_{12} \\ n_{21} + m_{21} & n_{22} + m_{22} \end{bmatrix}$$

- example

$$\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} + \begin{bmatrix} -2 & 5 \\ 7 & 1 \end{bmatrix} = \begin{bmatrix} 1 + (-2) & 3 + 5 \\ 2 + 7 & 4 + 1 \end{bmatrix} = \begin{bmatrix} -1 & 8 \\ 9 & 5 \end{bmatrix}$$

43

## Scalar-Matrix Multiplication

- multiply: scalar \* matrix = matrix

$$a \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} = \begin{bmatrix} a * m_{11} & a * m_{12} \\ a * m_{21} & a * m_{22} \end{bmatrix}$$

- example

$$3 \begin{bmatrix} 2 & 4 \\ 1 & 5 \end{bmatrix} = \begin{bmatrix} 3 * 2 & 3 * 4 \\ 3 * 1 & 3 * 5 \end{bmatrix} = \begin{bmatrix} 6 & 12 \\ 3 & 15 \end{bmatrix}$$

44

## Matrix-Matrix Multiplication

- can only multiply (n, k) by (k, m): number of left cols = number of right rows

• legal

$$\begin{bmatrix} a & b & c \\ e & f & g \\ o & p & q \end{bmatrix} \begin{bmatrix} h & i \\ j & k \\ l & m \end{bmatrix}$$

- undefined

$$\begin{bmatrix} a & b & c \\ e & f & g \\ o & p & q \end{bmatrix} \begin{bmatrix} h & i \\ j & k \end{bmatrix}$$

45

## Matrix-Matrix Multiplication

- row by column

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$p_{11} = m_{11}n_{11} + m_{12}n_{21}$$

46

## Matrix-Matrix Multiplication

- row by column

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$p_{11} = m_{11}n_{11} + m_{12}n_{21}$$

$$p_{21} = m_{21}n_{11} + m_{22}n_{21}$$

47

## Matrix-Matrix Multiplication

- row by column

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$p_{11} = m_{11}n_{11} + m_{12}n_{21}$$

$$p_{21} = m_{21}n_{11} + m_{22}n_{21}$$

$$p_{12} = m_{11}n_{12} + m_{12}n_{22}$$

48

### Matrix-Matrix Multiplication

- row by column

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$\begin{aligned} p_{11} &= m_{11}n_{11} + m_{12}n_{21} \\ p_{21} &= m_{21}n_{11} + m_{22}n_{21} \\ p_{12} &= m_{11}n_{12} + m_{12}n_{22} \\ p_{22} &= m_{21}n_{12} + m_{22}n_{22} \end{aligned}$$

49

### Matrix-Matrix Multiplication

- row by column

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$\begin{aligned} p_{11} &= m_{11}n_{11} + m_{12}n_{21} \\ p_{21} &= m_{21}n_{11} + m_{22}n_{21} \\ p_{12} &= m_{11}n_{12} + m_{12}n_{22} \\ p_{22} &= m_{21}n_{12} + m_{22}n_{22} \end{aligned}$$

- noncommutative: **AB != BA**

50

### Matrix-Vector Multiplication

- points as column vectors: postmultiply

$$\begin{bmatrix} x' \\ y' \\ z' \\ h' \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ h \end{bmatrix} \quad \mathbf{p}' = \mathbf{M}\mathbf{p}$$

- points as row vectors: premultiply

$$\begin{bmatrix} x' & y' & z' & h' \end{bmatrix} = \begin{bmatrix} x & y & z & h \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \quad \mathbf{p}'^T = \mathbf{p}^T \mathbf{M}^T$$

51

### Matrices

- transpose  $\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix}^T = \begin{bmatrix} m_{11} & m_{21} & m_{31} & m_{41} \\ m_{12} & m_{22} & m_{32} & m_{42} \\ m_{13} & m_{23} & m_{33} & m_{43} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{bmatrix}$
- identity  $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- inverse  $\mathbf{A}\mathbf{A}^{-1} = \mathbf{I}$ 
  - not all matrices are invertible

52

### Matrices and Linear Systems

- linear system of n equations, n unknowns

$$\begin{aligned} 3x + 7y + 2z &= 4 \\ 2x - 4y - 3z &= -1 \\ 5x + 2y + z &= 1 \end{aligned}$$

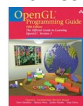
- matrix form **Ax=b**

$$\begin{bmatrix} 3 & 7 & 2 \\ 2 & -4 & -3 \\ 5 & 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ -1 \\ 1 \end{bmatrix}$$

53

### Readings for Next Time

- Mon Jan 11
  - RB Chap Introduction to OpenGL
  - RB Chap State Management and Drawing Geometric Objects
  - RB App Basics of GLUT (Aux in v 1.1)
- RB = Red Book = OpenGL Programming Guide
- <http://fly.cc.fer.hr/~unreal/theredbook/>



54

### Re-Reminder

- no class this Friday (Jan 8)!

55