Mathematics for Computer Graphics
Outline – Linear Algebra topics

• Scalars
• Vector Space
  – Scalars and vectors
• Affine Space
  – Scalars, vectors, and points
• Euclidean Space
  – Scalars, vectors, points
  – Distance metric
• Projections
• Matrix representations and operations
Scalars

• Scalar – a number
  – Ex: the scale, weight, or magnitude of something

• Two Fundamental Operations
  – Addition and multiplication

\[ \forall \alpha, \beta \in S, \ \alpha + \beta \in S, \ \alpha \cdot \beta \in S \]

\[ \alpha + \beta = \beta + \alpha \]  
Commutative

\[ \alpha \cdot \beta = \beta \cdot \alpha \]  
Associative

\[ \alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma \]

\[ \alpha \cdot (\beta \cdot \gamma) = (\alpha \cdot \beta) \cdot \gamma \]

\[ \alpha \cdot (\beta + \gamma) = (\alpha \cdot \beta) + (\alpha \cdot \gamma) \]  
Distributive
Scalars

• Two Special Scalars
  
  – Additive identity: $0$, multiplicative identity: $1$

    \[ \alpha + 0 = 0 + \alpha = \alpha \]
    \[ \alpha \cdot 1 = 1 \cdot \alpha = \alpha \]

  – Additive inverse $\quad -\alpha \quad \alpha + (-\alpha) = 0$

  – Multiplicative inverse $\quad \alpha^{-1} \quad \alpha \cdot \alpha^{-1} = 1$
2D scalar field

2D array of scalar values

In C#, the data structure might look something like this:

```csharp
float[,] temps;
temps = new float[50,50];
// (allocate after declaring)
```

http://upload.wikimedia.org/wikipedia/commons/a/a8/Scalar_field.png
Vector space

- Two Entities: *Scalars* and *Vectors*
- Vectors
  - Directed line segments
  - $n$-tuples of scalars
  - Two operations
    - *vector-vector addition*
    - *scalar-vector multiplication*
- Zero Vector
  $$v + 0 = v$$
  $$v + (-v) = 0$$

Directed line segments
Vector space

• In vector space, two vectors are equal if they are represented with the same $n$-tuple
  – Ex: $\mathbf{v} = (1,2,3) = (1,2,3)$
    - $\mathbf{v} \neq (3,2,1)$

• A vector, by itself, does not have a starting position
  – Visually you can think of a vector starting at $(0,0,0)$
Vector space operations

- Vectors = n-tuples \( \mathbf{v} = (v_0, v_1, \ldots, v_{n-1}) \)

  - Vector-vector addition
    \[
    \mathbf{u} + \mathbf{v} = (u_0, u_1, \ldots, u_{n-1}) + (v_0, v_1, \ldots, v_{n-1}) = (u_0 + v_0, u_1 + v_1, \ldots, u_{n-1} + v_{n-1})
    \]

  - Scalar-vector multiplication
    \[
    \alpha \mathbf{v} = (\alpha v_0, \alpha v_1, \ldots, \alpha v_{n-1})
    \]

  - Vector decomposition
    \[
    \mathbf{u} = \alpha_0 \mathbf{u}_0 + \alpha_1 \mathbf{u}_1 + \cdots + \alpha_{n-1} \mathbf{u}_{n-1}
    \]
Vector space operations

• Scalar-Vector Multiplication
  \( u \) and \( v \): vectors, \( \alpha \) and \( \beta \): scalars
  \[ \alpha (u + v) = \alpha u + \alpha v \]
  \[ (\alpha + \beta)u = \alpha u + \beta u \]

• Vector-Vector Addition
  – Visualize using head-to-tail axiom
Vector space limitations

• Vector space has no geometric concept
  – Vectors indicate magnitude and direction, not position

• Coordinate System
  – Origin: a particular reference point

Basis vectors located at the origin

Arbitrary placement of basis vectors

Identical vectors
Coordinate system

Boardwork examples: 1D, 2D, 3D coordinate axes

Unity uses left-handed coordinate system with
+x being right and +y being up
Affine Spaces

- Scalars, vectors, and points
- A point is represented as an $n$-tuple

- Operations [P and Q are points, $\mathbf{v}$ is a vector]
  - Point-point subtraction operation
    \[ \mathbf{v} = P - Q \]
  - Vector-point addition operation
    \[ P = \mathbf{v} + Q \]
    \[ (P - Q) + (Q - R) = (P - R) \]

- Frame: a Point $P_0$ and a Set of Vectors $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{n-1}$
  - All vectors and points in the space are defined relative to these
    \[ \mathbf{v} = \alpha_0 \mathbf{v}_0 + \alpha_1 \mathbf{v}_1 + \cdots + \alpha_{n-1} \mathbf{v}_{n-1} \]
    \[ P = P_0 + \beta_0 \mathbf{v}_0 + \beta_1 \mathbf{v}_1 + \cdots + \beta_{n-1} \mathbf{v}_{n-1} \]
Euclidean Spaces

• Affine space does not include an operation or metric for distance between points

• Create a new operation: Inner (dot) Product
  – Input: two vectors   Output: scalar
  – $\alpha, \beta :$ scalars   $u, v, w :$ vectors

  – Properties we want:  $u \cdot v = v \cdot u$
    $(\alpha u + \beta v) \cdot w = \alpha u \cdot w + \beta v \cdot w$
    $v \cdot v > 0$ if $v \neq 0$
    $0 \cdot 0 = 0$
  For orthogonal vectors  $u \cdot v = 0$
Euclidean Spaces – dot product

• If we can multiply two n-tuples, this implies
  
  – Magnitude (length) of a vector  \( |v| = \sqrt{v \cdot v} \)

  – Distance between two points  \( |P - Q| = \sqrt{(P - Q) \cdot (P - Q)} \)

  – Measure of the angle between two vectors  \( u \cdot v = |u||v|\cos \theta \)
    
    • \( \cos \theta = 0 \)  \( \Rightarrow \) orthogonal
    • \( \cos \theta = 1 \)  \( \Rightarrow \) parallel

• Computation of dot product
  
  \[ u \cdot v = \sum_{i=0}^{n-1} u_i \cdot v_i = u_0 \cdot v_0 + u_1 \cdot v_1 + \ldots + u_{n-1} \cdot v_{n-1} \]
Projections

• We can determine if two points are “close” to each other, what about vectors?
• How much of \( w \) is in the same direction as \( v \)?
• Given vectors \( v \) and \( w \), decompose \( w \) into two parts, one parallel to \( v \) and one orthogonal to \( v \)

\[
w = \alpha v + u
\]

\[
w \cdot v = \alpha v \cdot v + u \cdot v = \alpha v \cdot v
\]

\[
\therefore \alpha = \frac{w \cdot v}{v \cdot v}
\]

\[
\therefore u = w - \alpha v = w - \frac{w \cdot v}{v \cdot v}v
\]
Matrices

• Definitions
• Matrix Operations
• Row and Column Matrices
• Change of Representation
• Relating matrices and vectors
What is a Matrix?

• A matrix is a set of elements, organized into rows and columns.

\[
\begin{bmatrix}
a & b \\
c & d
\end{bmatrix}
\]
Definitions

• \( n \times m \) Array of Scalars (\( n \) Rows and \( m \) Columns)
  – \( n \): row \textit{dimension} of a matrix, \( m \): column \textit{dimension}
  – \( m = n \) \( \Rightarrow \) \textit{square matrix} of dimension \( n \)
  – Element \( \{a_{ij}\}, \ i = 0, \ldots, n-1, \ j = 0, \ldots, m-1 \)
    \[
    A = \begin{bmatrix} a_{ij} \end{bmatrix}
    \]
  – \textit{Transpose}: interchanging the rows and columns of a matrix
    \[
    A^T = \begin{bmatrix} a_{ji} \end{bmatrix}
    \]

• Column Matrices and Row Matrices
  – \textit{Column matrix} (\( n \times 1 \) matrix): \( \mathbf{b} = [b_i] = \begin{bmatrix} b_0 \\ \vdots \\ b_n \end{bmatrix} \)
  – \textit{Row matrix} (\( 1 \times n \) matrix):
    \[
    \mathbf{b}^T = \begin{bmatrix} b_0 & b_1 & \ldots & b_{n-1} \end{bmatrix}
    \]
Matrix Operations

• Scalar- Matrix Multiplication \( \alpha A = [\alpha a_{ij}] \)
  – Multiply every element by the scalar

• Matrix-Matrix Addition \( C = A + B = [a_{ij} + b_{ij}] \)
  – Add elements with same index

• Matrix-Matrix Multiplication
  – A: \( n \times l \) matrix, B: \( l \times m \) \( \Rightarrow \) C: \( n \times m \) matrix

\[
C = AB = \begin{bmatrix} c_{ij} \end{bmatrix}
\]

\[
c_{ij} = \sum_{k=0}^{l-1} a_{ik} b_{kj}
\]

\( c_{ij} \) = the sum of multiplying elements in row \( i \) of matrix \( a \) times elements in column \( j \) of matrix \( b \)
Matrix Operation Examples

\[
\begin{bmatrix}
a & b \\
c & d \\
\end{bmatrix} + \begin{bmatrix}
e & f \\
g & h \\
\end{bmatrix} = \begin{bmatrix}
a+e & b+f \\
c+g & d+h \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
a & b \\
c & d \\
\end{bmatrix} - \begin{bmatrix}
e & f \\
g & h \\
\end{bmatrix} = \begin{bmatrix}
a-e & b-f \\
c-g & d-h \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
a & b \\
c & d \\
\end{bmatrix} \begin{bmatrix}
e & f \\
g & h \\
\end{bmatrix} = \begin{bmatrix}
ae+bg & af+bh \\
ce+dg & cf+dh \\
\end{bmatrix}
\]
Matrix Operations

• Properties of Scalar-Matrix Multiplication
  \[ \alpha (\beta A) = (\alpha \beta) A \]
  \[ \alpha \beta A = \beta \alpha A \]

• Properties of Matrix-Matrix Addition
  – Commutative: \( A + B = B + A \)
  – Associative: \( A + (B + C) = (A + B) + C \)

• Properties of Matrix-Matrix Multiplication
  \[ A (BC) = (AB) C \]
  \[ AB \neq BA \]

• Identity Matrix \( I \) (Square Matrix)
  \[
  I = \begin{bmatrix}
  1 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
  \end{bmatrix}
  \]
  \[ a_{ij} = \begin{cases} 
  1 & \text{if } i = j \\
  0 & \text{otherwise}
  \end{cases} \]
  \[ AI = A \]
  \[ IB = B \]
Matrix Multiplication Order

• Is $AB = BA$? Maybe, but maybe not!

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} ae + bg & \ldots \\ \ldots & \ldots \end{bmatrix} \quad \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} ea + fc & \ldots \\ \ldots & \ldots \end{bmatrix}$$

• In general, matrix multiplication is NOT commutative!

• The order of matrix multiplications is important!
Row and Column Matrices + points

• Column Matrix
  
  \[
  \mathbf{p} = \begin{bmatrix}
  x \\
  y \\
  z
  \end{bmatrix}
  \]

  By convention we will use \( \mathbf{p} \) column matrices for points

  - Row matrix
    
    \[
    \mathbf{p}^T = \begin{bmatrix}
    x \\
    y \\
    z
    \end{bmatrix}
    \]

• Concatenations
  
  \[
  \mathbf{p}' = \mathbf{A}\mathbf{p}
  \]

  - Associative
    
    \[
    \mathbf{p}' = \mathbf{A}\mathbf{B}\mathbf{C}\mathbf{p}
    \]

• By Row Matrix
  
  \[
  (\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T
  \]

  \[
  \mathbf{p}'^T = \mathbf{p}^T \mathbf{C}^T \mathbf{B}^T \mathbf{A}^T
  \]
Vector Operations

• Vector: 1 x N matrix
• Interpretation: a line in N dimensional space
• Dot Product and Magnitude operations

\[ \vec{v} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \]
Vectors: Dot Product

\[ a \cdot b = ab^T = \begin{bmatrix} d \\ e \\ f \end{bmatrix} = ad + be + cf \]

Think of the dot product as a matrix multiplication

The magnitude is the dot product of a vector with itself

If \( a \) and \( b \) are both length one, the angle between them is the \( \cos^{-1} \) of their dot product

\[ \|a\|^2 = aa^T = \sqrt{aa + bb + cc} \]
Inverse of a Matrix

- Identity matrix:
  \[ AI = A \]
- Some matrices have an inverse, such that:
  \[ AA^{-1} = I \]

\[ I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \]

\[ A = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \]

\[ A^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \]
Inverse of a Matrix

• Some matrices do not have an inverse

\[
A = \begin{bmatrix}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix} \quad A^{-1} = \begin{bmatrix}
a & b & c \\
d & e & f \\
g & h & i
\end{bmatrix}
\]

\[
AA^{-1}_{00} = 0 \cdot a + 0 \cdot d + 0 \cdot g = 0 \neq 1
\]
Inverse of Matrix Concatenation

• Inversion of concatenations

\[(ABC)^{-1} = C^{-1}B^{-1}A^{-1}\]

\[A * B * C * X = I\]
\[A * B * C * C^{-1} = A * B\]
\[A * B * B^{-1} = A\]
\[A * A^{-1} = I\]

Order is important, so \[X = C^{-1}B^{-1}A^{-1}\]
Summary

• Primitives: scalars, vectors, points

• Operations: addition and multiplication

• Matrix representation and operations