CSE452 Computer Graphics

Lecture 4: Points, Vectors and Shapes
Points and Vectors

• Same representation
  \[(x, y) \begin{bmatrix} (x, y, z) \text{ in 3D} \end{bmatrix}\]

• Different meaning:
  – Point: a fixed **location** (relative to \(0,0\) or \(0,0,0\))
    • Coordinates change as location changes
  – Vector: a **direction** and **length**
    • Coordinates do not change as location changes
Points and Vectors

- Same representation
  
  \((x, y)\) \[\begin{pmatrix} x \ y \ z \end{pmatrix}\text{ in 3D}\]

- Different meaning:
  
  - Point: a fixed location (relative to \(\{0,0\}\) or \(\{0,0,0\}\))
    - Coordinates change as location changes
  
  - Vector: a direction and length
    - Coordinates do not change as location changes
Point Operations

- **Subtraction**
  - Result is a vector

- **Addition with a vector**
  - Result is a point
Point Operations

• Subtraction
  – Result is a vector
    \[ \vec{P_2} - \vec{P_1} = \vec{v} = (P_{2x} - P_{1x}, P_{2y} - P_{1y}) \]

• Addition with a vector
  – Result is a point
    \[ \vec{P_1} + \vec{v} = \vec{P_2} = (P_{1x} + v_x, P_{1y} + v_y) \]
Point Operations

• Addition with a vector
  – Resulting location does not change with the origin

\[
\begin{align*}
  p + v &= (x_1 + x_2, y_1 + y_2) \\
  p &= (x_1, y_1) \\
  v &= (x_2, y_2)
\end{align*}
\]
Point Operations

• Addition with a vector
  – Resulting location does not change with the origin
Point Operations

• Can two points add?

\[ p_1 + p_2 = (x_1 + x_2, y_1 + y_2) \]
Point Operations

- Can two points add?
  - In general, **no**: result is dependent on where the origin is
    - But there are exceptions; will discuss later

\[
\begin{align*}
\text{p}_1 + \text{p}_2 & = (x_1 + x_2 + 2a, y_1 + y_2 + 2b) \\
\text{p}_1 & = (x_1 + a, y_1 + b) \\
\text{p}_2 & = (x_2, y_2)
\end{align*}
\]
Vector Operations

- **Addition/Subtraction**
  - Result is a **vector**
  \[ \mathbf{u}_1 + \mathbf{u}_2 \]

- **Scaling by a scalar**
  - Result is a **vector**
  \[ s \times \mathbf{u} \]

- **Magnitude**
  - Result is a **scalar**
  \[ |\mathbf{u}| = \sqrt{u_x^2 + u_y^2} \]
  - A **unit vector** \( |\mathbf{u}| = 1 \)
  - To make a **unit vector** \( \frac{\mathbf{u}}{|\mathbf{u}|} \) (normalization)
Vector Operations

• Dot product
  – Result is a scalar
    \[ \mathbf{v}_1 \cdot \mathbf{v}_2 = ||\mathbf{v}_1|| ||\mathbf{v}_2|| \cos \theta \]
  – In coordinates (simple!)
    • 2D: \[ \mathbf{v}_1 \cdot \mathbf{v}_2 = v_{1x}v_{2x} + v_{1y}v_{2y} \]
    • 3D: \[ \mathbf{v}_1 \cdot \mathbf{v}_2 = v_{1x}v_{2x} + v_{1y}v_{2y} + v_{1z}v_{2z} \]
  • Matrix product between a row and a column vector
Vector Operations

• Uses of dot products
  – **Angle** between vectors:
    • Orthogonal:
    – **Projected length** of \( \mathbf{v}_1 \) onto \( \mathbf{v}_2 \):
      
      \[
      h = \mathbf{v}_1 \times \mathbf{v}_2
      \]
      
      if \( \mathbf{v}_2 \) is a unit vector
Vector Operations

- **Uses of dot products**
  - **Angle** between vectors:
    \[
    a = \arccos \left( \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{|\mathbf{v}_1||\mathbf{v}_2|} \right)
    \]
  - **Orthogonal**:
    \[
    \mathbf{v}_1 \times \mathbf{v}_2 = 0
    \]
  - **Projected length** of \(\mathbf{v}_1\) onto \(\mathbf{v}_2\):
    if \(\mathbf{v}_2\) is a unit vector
    \[
    h = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{|\mathbf{v}_2|} = \mathbf{v}_1 \cdot \mathbf{v}_2
    \]
Vector Operations

• Cross product (in 3D)
  – Result is another 3D vector
    • Direction: Normal to the plane where both vectors lie (right-hand rule)
    • Magnitude: \[ |\mathbf{v}_1 \times \mathbf{v}_2| = |\mathbf{v}_1| |\mathbf{v}_2| \sin \theta \]
  – In coordinates:
    • Determinant of a matrix:
      \[
      \begin{vmatrix}
        \mathbf{i} & \mathbf{j} & \mathbf{k} \\
        \mathbf{v}_1 \\
        \mathbf{v}_2 \\
      \end{vmatrix}
      \]
      \[
      \mathbf{v}_1 \times \mathbf{v}_2 = 
      (v_{1y}v_{2z} - v_{1z}v_{2y}, v_{1z}v_{2x} - v_{1x}v_{2z}, v_{1x}v_{2y} - v_{1y}v_{2x})v_1
      \]
Vector Operations

• Uses of cross products
  – Getting the normal vector of the plane
    • E.g., the normal of a triangle formed by $\mathbf{v}_1 \times \mathbf{v}_2$
  – Computing area of the triangle formed by $\mathbf{v}_1 \times \mathbf{v}_2$
    \[
    \text{Area} = \frac{|\mathbf{v}_1 \times \mathbf{v}_2|}{2}
    \]
    • Testing if vectors are parallel:
    \[
    |\mathbf{v}_1 \times \mathbf{v}_2| = 0
    \]
# Vector Operations

<table>
<thead>
<tr>
<th></th>
<th>Dot Product</th>
<th>Cross Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Distributive?</strong></td>
<td>( \mathbf{v} \cdot (\mathbf{v}_1 + \mathbf{v}_2) \neq \mathbf{v}_1 \cdot \mathbf{v}_2 + \mathbf{v}_2 \cdot \mathbf{v}_1 )</td>
<td>( \mathbf{v} \times (\mathbf{v}_1 + \mathbf{v}_2) \neq \mathbf{v}_1 \times \mathbf{v}_1 + \mathbf{v}_2 \times \mathbf{v}_2 )</td>
</tr>
<tr>
<td><strong>Commutative?</strong></td>
<td>( \mathbf{v} \cdot \mathbf{v}_2 \neq \mathbf{v}_2 \cdot \mathbf{v}_1 )</td>
<td>( \mathbf{v}_1 \times \mathbf{v}_2 \neq \mathbf{v}_2 \times \mathbf{v}_1 )</td>
</tr>
<tr>
<td><strong>Associative?</strong></td>
<td>( \mathbf{v}_1 \cdot (\mathbf{v}_2 \cdot \mathbf{v}_3) \neq ? )</td>
<td>( \mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3) \neq (\mathbf{v}_1 \times \mathbf{v}_2) \times \mathbf{v}_3 )</td>
</tr>
</tbody>
</table>
## Vector Operations

<table>
<thead>
<tr>
<th></th>
<th>Dot Product</th>
<th>Cross Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Distributive?</strong></td>
<td>$\mathbf{v} \cdot (\mathbf{v}_1 + \mathbf{v}_2) \neq \mathbf{v}_1 \cdot \mathbf{v}_2 + \mathbf{v}_2 \cdot \mathbf{v}_2$</td>
<td>$\mathbf{v} \times (\mathbf{v}_1 + \mathbf{v}_2) \neq \mathbf{v}_1 \times \mathbf{v}_2 + \mathbf{v}_2 \times \mathbf{v}_2$</td>
</tr>
<tr>
<td><strong>Commutative?</strong></td>
<td>$\mathbf{v} \cdot \mathbf{v}_2 \neq \mathbf{v}_2 \cdot \mathbf{v}_1$</td>
<td>$\mathbf{v}_1 \times \mathbf{v}_2 \neq \mathbf{v}_2 \times \mathbf{v}_1$</td>
</tr>
<tr>
<td>$\mathbf{v}_1 \times \mathbf{v}_2 = -\mathbf{v}_2 \times \mathbf{v}_1$</td>
<td>$\mathbf{v}_1 \times \mathbf{v}_2 \neq \mathbf{v}_2 \times \mathbf{v}_1$</td>
<td></td>
</tr>
<tr>
<td><strong>Associative?</strong></td>
<td>$\mathbf{v}_1 \cdot (\mathbf{v}_2 \cdot \mathbf{v}_3)$ \textit{Not defined}</td>
<td>$\mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3) \neq \mathbf{v}_1 \times \mathbf{v}_2 + \mathbf{v}_1 \times \mathbf{v}_3$</td>
</tr>
<tr>
<td>$\mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3)$ \textit{Not defined}</td>
<td>$\mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3) \neq \mathbf{v}_1 \times \mathbf{v}_2 + \mathbf{v}_1 \times \mathbf{v}_3$</td>
<td></td>
</tr>
</tbody>
</table>
Shapes and Dimensions

• **0-dimensional shape:**

• **1-dimensional shape:**

• **2-dimensional shape:**
Shapes and Dimensions

- **0-dimensional shape: point**
  - No length or area

- **1-dimensional shape: curve**
  - Has non-zero “length”
  - Examples: line (segment), circle (arc), parabola, etc.

- **2-dimensional shape: surface**
  - Has non-zero “area”
  - Examples: filled triangle or quad, filled circle, surface of a cylinder, surface of a sphere, etc.
Tessellation

• Graphics cards are good at drawing tessellated elements
  – E.g., line segments, triangles, etc.
1D Tessellation

• Approximate a 1D curve shape by line segments
  – Define the curve as a function of one parameter
  – Generate samples on the curve at fixed intervals of the parameter
  – Connect successive samples by line segments

\[
\begin{align*}
  f(0) & \quad \text{f(t)} & \quad f(0.1) & \quad f(0.2) \\
  f(0) & \quad f(1) & \quad f(0)
\end{align*}
\]
A line segment:

\[ \mathbf{P}(t) = (1 - t) \mathbf{P}_1 + t \mathbf{P}_2 \quad 0 \leq t \leq 1 \]
Can Points Add? Sometimes.

- Linear interpolation (for two points)
  \[ P(t) = (1-t)P_1 + tP_2 \quad 0 \leq t \leq 1 \]
  - For any \( t \), location of \( p \) is invariant to origin change
- It is basically a point-and-vector addition:
  \[ P(t) = P_1 + t(P_2 - P_1) \]
Can Points Add? Sometimes.

- **Affine combinations (for multiple points)**

  \[ P = \sum_{i=1}^{n} t_i P_i \quad \text{where} \quad \sum_{i=1}^{n} t_i = 1 \]

  - For any \( t_i \), location of \( P \) is invariant to origin change

- Again, a point-and-vector addition:

  \[ P = P_i + \sum_{i=1}^{n} t_i (P_i - P_i) \]
Parameterizing 1D Shapes

- A line segment:

\[ P(t) = (1 - t) P_1 + t P_2 \quad 0 \leq t \leq 1 \]
Parameterizing 1D Shapes

- Circle
  - Centered at origin with radius $r$

$$P(\alpha) = (r \cos \alpha, r \sin \alpha) \quad 0 \leq \alpha < 2\pi$$
Parameterizing 1D Shapes

- **Ellipse**
  - Centered at origin with axes $a$, $b$

  $$p(d) = (a \cos d, b \sin d) \quad 0 \leq d < 2\pi$$
2D Tessellation

• Approximate a 2D surface shape by \textit{triangles}
  – Define the surface as a function of \textit{two parameters}
  – Generate samples at fixed intervals of both parameters
  – Connect samples by triangles
Parameterizing 2D Shapes

- Filled disk
  - Centered at origin with radius r
Parameterizing 2D Shapes

- Filled disk
  - Centered at origin with radius $r$

$$P(d, \alpha) = (d \cos \alpha, d \sin \alpha)$$

$0 \leq d \leq r$, $0 \leq \alpha < 2\pi$
Parameterizing 2D Shapes

- Filled quad
Parameterizing 2D Shapes

- Filled quad

\[ q(u) = (1-u) p_1 + u p_2 \]
\[ r(u) = (1-u) p_3 + u p_4 \]
\[ p(u,v) = (1-v) q(u) + v r(u) \]

\[ 0 \leq u \leq 1, \ 0 \leq v \leq 1 \]
Parameterizing 2D Shapes

- Filled triangle
Parameterizing 2D Shapes

- Filled triangle

\[ Q(u) = (1-u)P_1 + uP_2 \]

\[ P(u,v) = (1-v)Q(u) + vP_3 \]

\[ 0 \leq u \leq 1, \quad 0 \leq v \leq 1 \]
Parameterizing 2D Shapes

- Outer surface of a cylinder
  - Base centered at origin
  - Radius $r$, height $h$
Parameterizing 2D Shapes

• Outer surface of a cylinder
  – Base centered at origin
  – Radius r, height h

\[ p(d, \alpha) = \left( r \cos \alpha, r \sin \alpha, d \right) \]

\[ 0 \leq d \leq h, \quad 0 \leq \alpha < 2\pi \]
Parameterizing 2D Shapes

- **Cone surface**
  - Base centered at origin
  - Radius $r$, height $h$
Parameterizing 2D Shapes

- Cone surface
  - Base centered at origin
  - Radius $r$, height $h$

$$p(d, \alpha) = (g \cos \alpha, g \sin \alpha, d)$$

$$g = \frac{r(h - d)}{h}$$

$$0 \leq d \leq h, \quad 0 \leq \alpha < 2\pi$$
Parameterizing 2D Shapes

- Sphere surface
  - Centered at origin with radius $r$
  - Not the best parameterization...
Parameterizing 2D Shapes

- Sphere surface
  - Centered at origin with radius $r$

  $$P(\alpha, \beta) = (r \cos(\beta) \cos(\alpha), r \cos(\beta) \sin(\alpha), r \sin(\beta))$$

  - Not the best parameterization…