

5CM507 Graphics

Lecture A02a Mathematics

Dr Youbing Zhao

October 2, 2025

Last Week

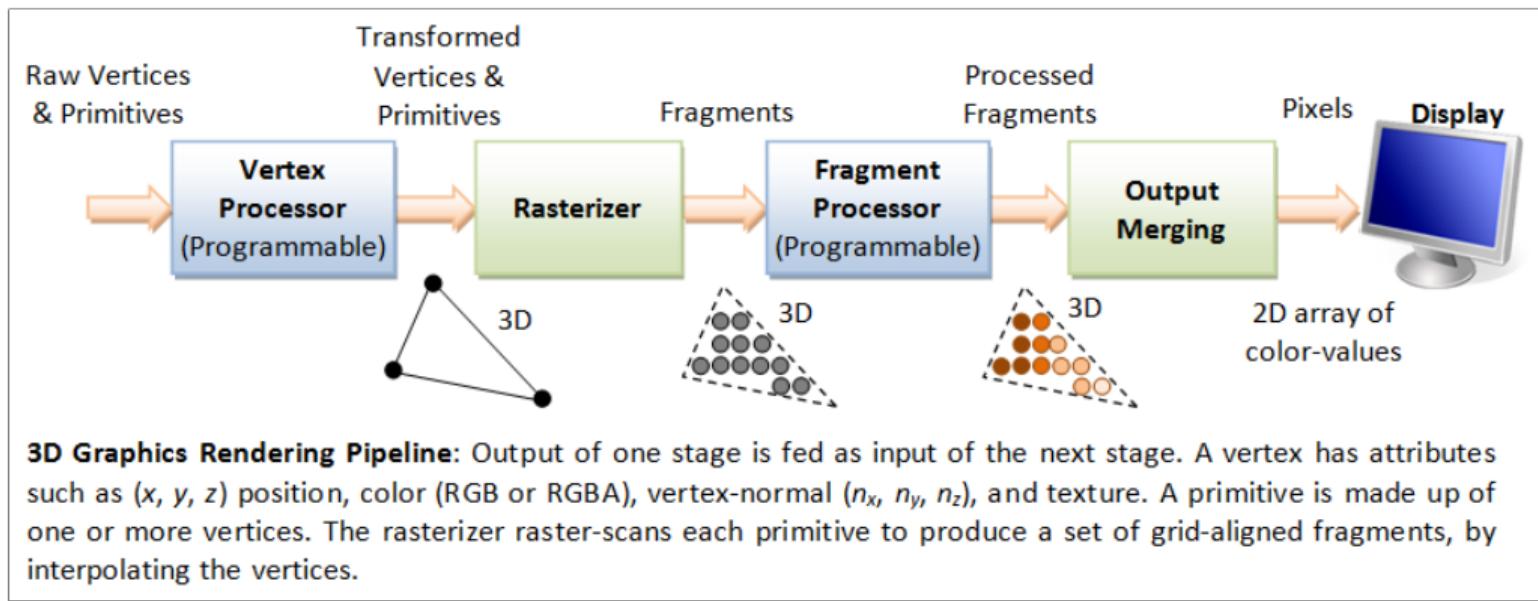


An Introduction to

- ▶ Computer Graphics (3D -> 2D)
- ▶ Graphics Pipeline (vertex shader and fragment shader)
- ▶ Module Logistics

Note: this module is not teaching how to use a 3d modelling software, but teach you the basic rendering, lighting, animation, and physics mechanisms of game engines. It requires programming.

The Classical Rasterisation Pipeline



[Prof. Chua Hock Chuan, NTU]

Overview



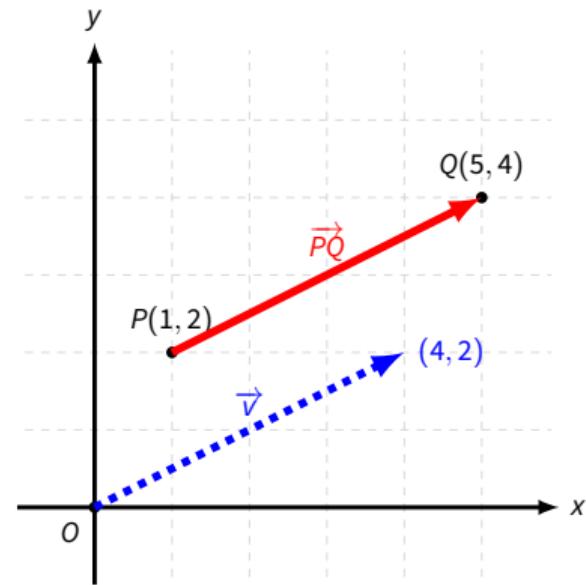
- ▶ **Vectors** and dot product
- ▶ Matrix and transforms

Vectors

Vectors

Representation

- ▶ Properties
 - ▶ Direction
 - ▶ Length(Magnitude)
 - ▶ No absolute starting point
- ▶ Examples include
 - ▶ Force
 - ▶ Velocity
 - ▶ Represented by the start and end points \vec{PQ}
 - ▶ Notation \vec{v} or in bold v : e.g. $v = Q - P = [4, 2]$



Vector Multiplication



- ▶ Scalar product (easy)
 - ▶ Every vector can be multiplied by a number
- ▶ Dot product (most important)
- ▶ Cross product (needed)

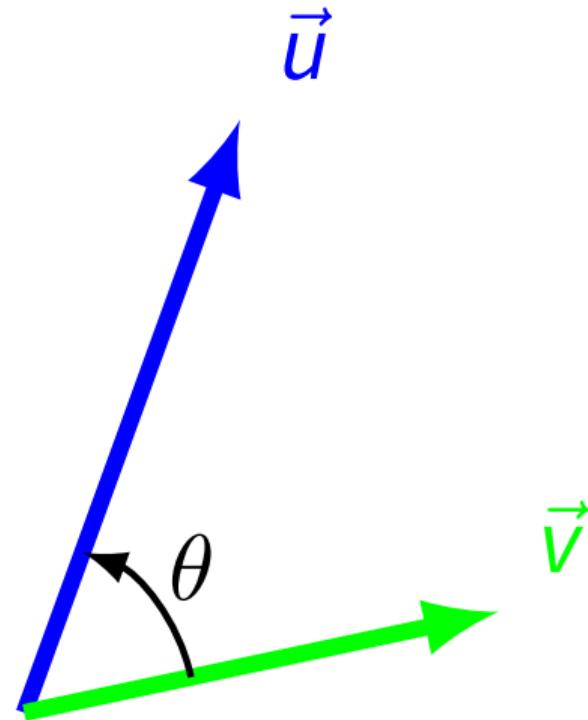
Dot Product



Definition

- ▶ 2D: $\mathbf{u} \cdot \mathbf{v} = u_x v_x + u_y v_y$
- ▶ 3D: $\mathbf{u} \cdot \mathbf{v} = u_x v_x + u_y v_y + u_z v_z$
- ▶ General: $\mathbf{u} \cdot \mathbf{v} = \sum_{i=1}^n u_i v_i$
- ▶ Vector length: $\mathbf{u} \cdot \mathbf{u} = \|\mathbf{u}\|^2$
- ▶ A dot product returns a scalar

```
float glm::dot(glm::vec u, glm::vec v)
```



Dot Product

Angle between vectors and vector projection

If two vectors have an angle of θ

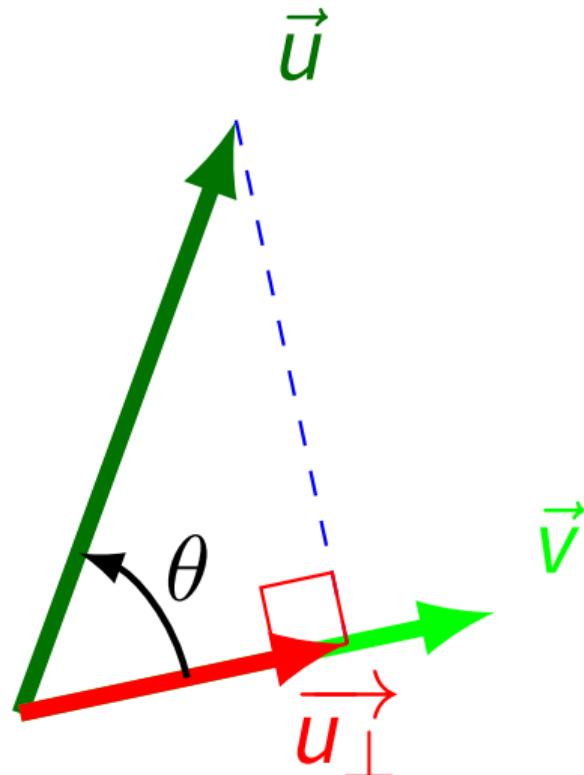
$$\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos(\theta)$$

The angle between two vectors

- ▶ $\cos(\theta) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$
- ▶ $\cos(0^\circ) = 1$: parallel
- ▶ $\cos(90^\circ) = 0$: perpendicular

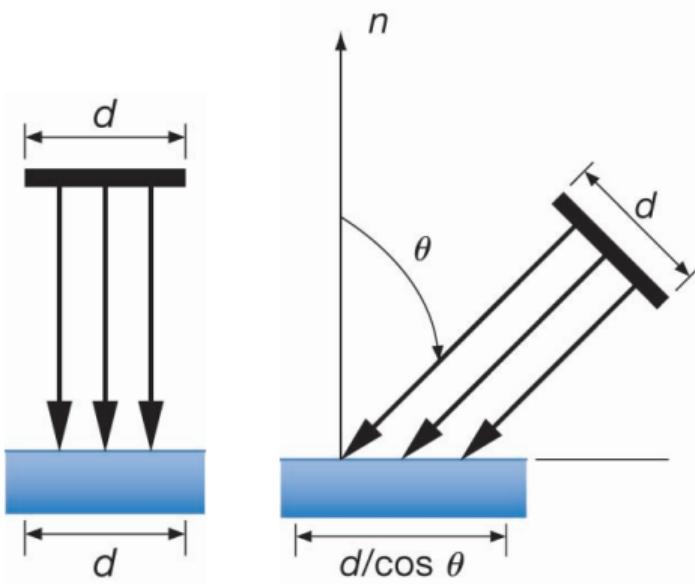
Projection of \mathbf{u} on \mathbf{v} :

$$\mathbf{u}_\perp = \|\mathbf{u}\| \cos(\theta) \frac{\mathbf{v}}{\|\mathbf{v}\|} = \frac{\mathbf{u} \cdot \mathbf{v}}{\mathbf{v} \cdot \mathbf{v}} \mathbf{v}$$

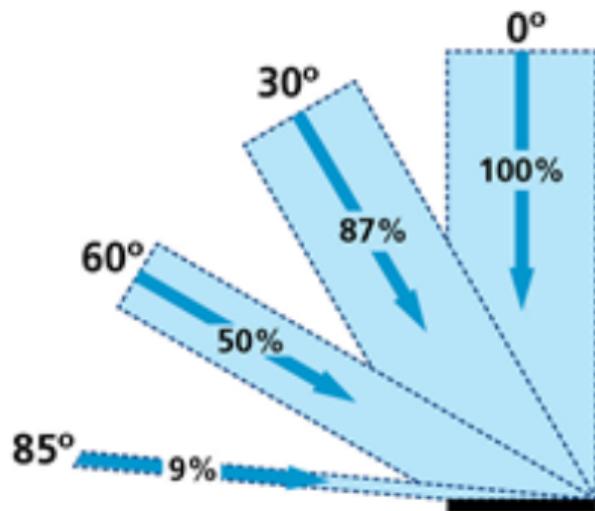


Dot Product in Lighting Models

$$E_\theta = E_0 \cos \theta = E_0 (\hat{l} \cdot \hat{n})$$



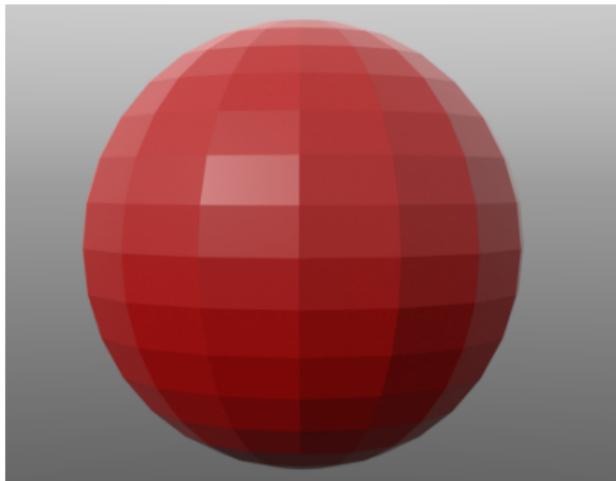
Cosine Law: $E_\theta = E * \cos(\theta)$



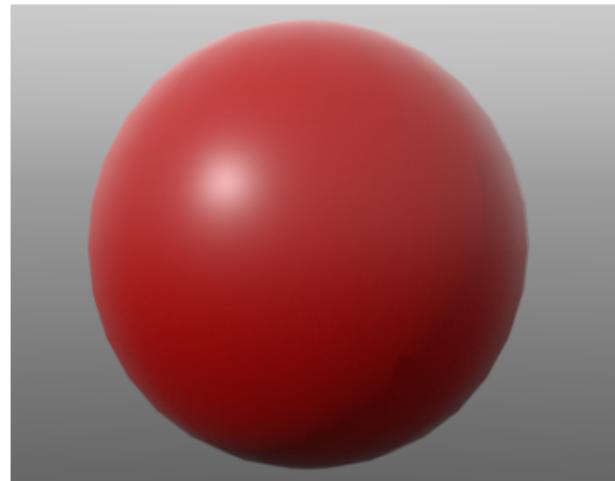
Light intensities Imply the Object Shape



Under a given directed light source, object surfaces with different normals will receive and reflect light differently, allowing us to perceive the shape of the 3D object.



Flat shading



Smooth shading

3D Cross Product

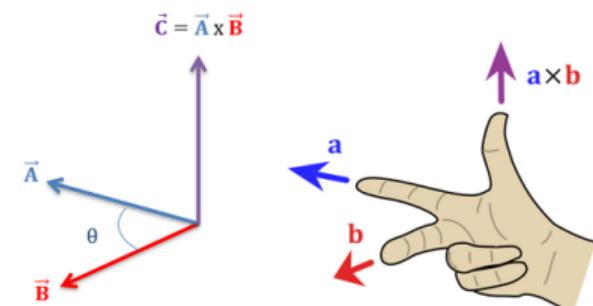
creates a vector

- ▶ Two non-parallel vectors A and B
- ▶ $C = A \times B$
 - ▶ a vector that is orthogonal to A and B
 - ▶ direction decided by the right hand rule
- ▶ $A \times B = (a_y b_z - a_z b_y, a_z b_x - a_x b_z, a_x b_y - a_y b_x)$

$$A \times B = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix} \quad \|A \times B\| = \|A\| \|B\| \sin \theta$$

Note: details not needed, what you need is only:

```
glm::vec3 glm::cross(glm::vec3 va, glm::vec3
vb)
```

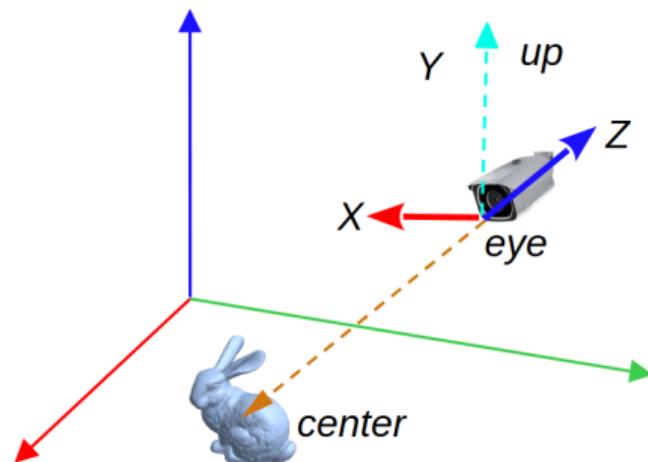


Right hand rule

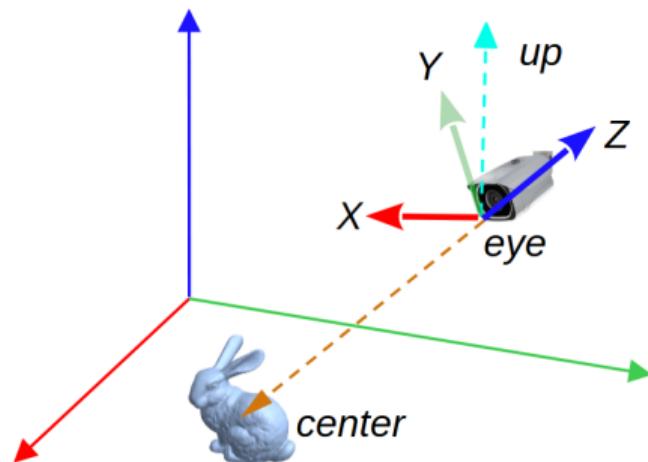
Using Cross Product to Set up the Camera Frame

```
mat4 glm::lookAt(vec3 eye, vec3 center, vec3 up)
```

$$\vec{Z} : \text{eye} - \text{center}$$



X (left) Axis of the camera frame: $\vec{X} : \vec{up} \times \vec{Z}$



Y (up) Axis of the camera frame: $\vec{Y} = \vec{Z} \times \vec{X}$

Normals of Triangles and Vertices in a Triangle Mesh for Lighting and Shading

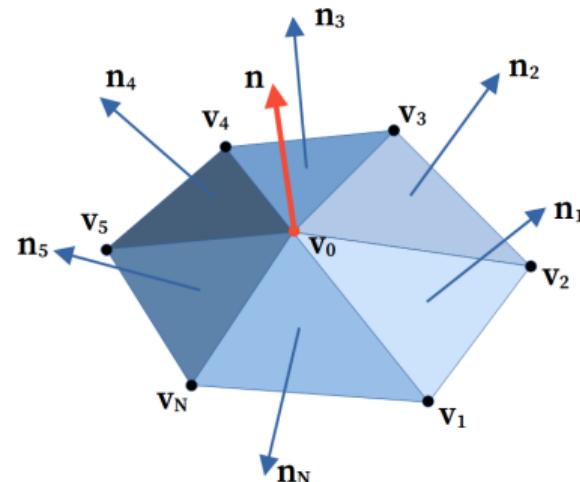
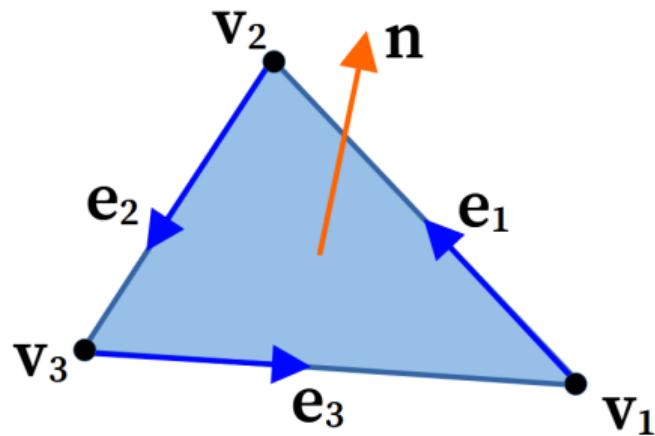
$$\vec{e}_1 = \frac{\vec{v}_2 - \vec{v}_1}{\|\vec{v}_2 - \vec{v}_1\|}, \vec{e}_2 = \frac{\vec{v}_3 - \vec{v}_2}{\|\vec{v}_3 - \vec{v}_2\|}, \vec{e}_3 = \frac{\vec{v}_1 - \vec{v}_3}{\|\vec{v}_1 - \vec{v}_3\|}$$

Triangle normal: $\vec{n} = \vec{e}_1 \times \vec{e}_2$

Vertex Normal: The angle or area-weighted average of the normals of neighboring triangles.

$$\vec{n} = \frac{\sum_{i=1}^N w_i \vec{n}_i}{\left\| \sum_{i=1}^N w_i \vec{n}_i \right\|}$$

Notice : $\text{Area}_{123} = \frac{1}{2} |\vec{v}_1 \vec{v}_2 \times \vec{v}_1 \vec{v}_3|$



Matrices

Matrix Multiplication

- ▶ $[C] = [A][B]$
- ▶ Sum over rows & columns
- ▶ Recall: matrix multiplication is
 - ▶ Associative $ABC = A(BC)$
 - ▶ Not commutative $AB \neq BA$

$$c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} + a_{14}b_{41}$$
$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{bmatrix}$$

2×4 4×3 2×3

$$c_{22} = a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32} + a_{24}b_{42}$$
$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{bmatrix}$$

Linear Transformation as Matrices

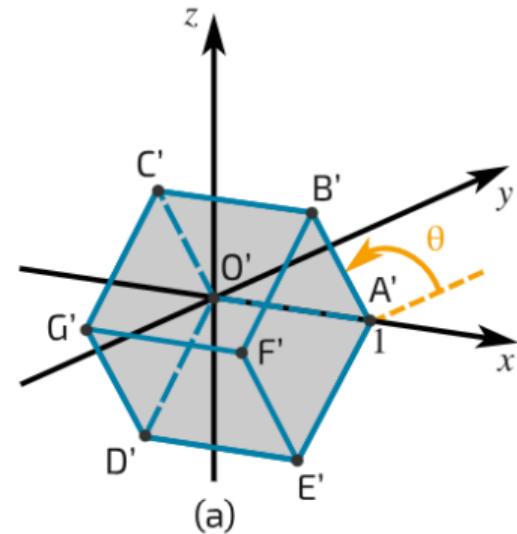
$$x' = x$$

$$y' = \cos(\theta)y - \sin(\theta)z$$

$$z' = \sin(\theta)y + \cos(\theta)z$$

$$M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix}$$

$$P' = MP = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$



Rotation by an angle θ around the x axis

To be continued by **Transformation**