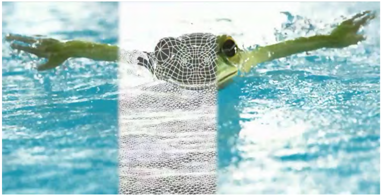
# Part 1: 2D/3D Geometry, Colour, Illumination

## Vectors, Matrices, Transformations

Patrice Delmas and Georgy Gimel'farb

COMPSCI 373 Computer Graphics and Image Processing



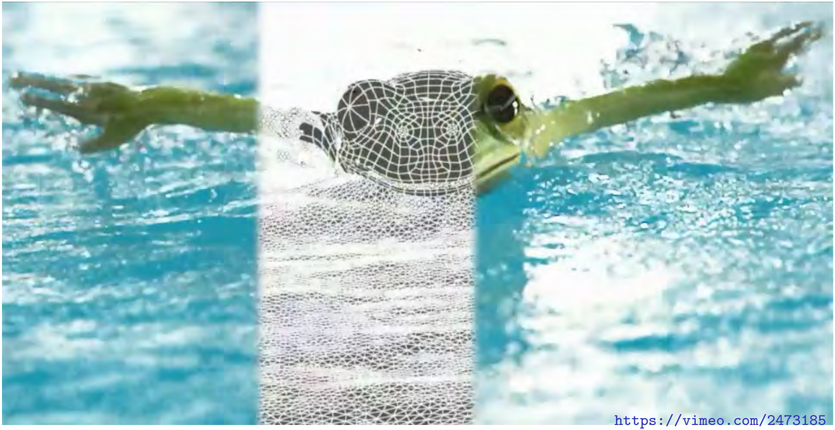
https://vimeo.com/2473185

- *Lecturer*: Georgy Gimel'farb (`ggim001@cs`; 86609)
- *Office hours*: whenever the door of 303S.389 is open...

**Part 1 overview:**

1. 2D/3D geometry:
    1. 2D/3D points; matrices; vectors; dot and cross products.
    2. Geometry of planes; 2D affine transformations.
    3. Homogeneous coordinates; 3D affine transformations.
2. Colour
    1. Colours: light-material interaction; human colour perception.
    2. SDF (spectral density function).
    3. SRF (spectral response function).
    4. Colour spaces. RGB, CIE XYZ, HLS; colour gamut.
3. Illumination
    1. Phong illumination model; shading; reflection; shadows.

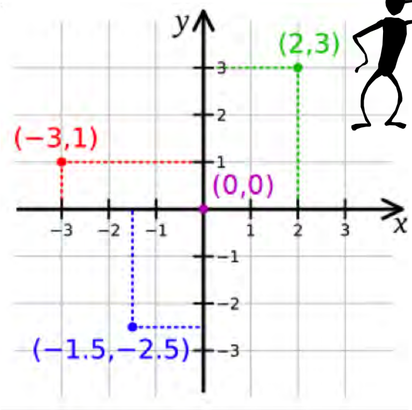
# Computer Graphics and Imaging Geometry



https://vimeo.com/2473185

Given illumination sources and optical cameras mimicking human eyes, model shapes and reflective properties of real-world surfaces to find an image or a video sequence that each particular eye will perceive. . .

# Points, Vectors, and Matrices




Two-dimensional (2D) points

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \end{bmatrix}$$

Vector - column
(or $2 \times 1$ matrix)

$$\begin{bmatrix} 0.5 & 3.0 & -1.7 \\ 3.8 & -0.3 & 0.7 \end{bmatrix}$$

$2 \times 3$ matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$2 \times 2$ matrix

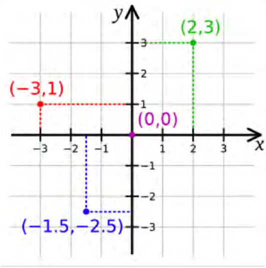
# Points

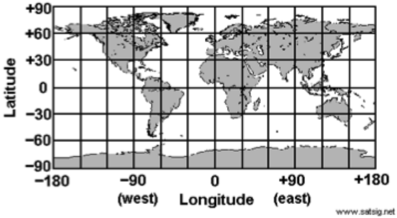
Cartesian **coordinate system**:

- Orthogonal axes of coordinates (numbers).
- **Origin**, or centre: all zero coordinates.

**Point** – a spatial position:

- 2D point – a pair $(x, y)$ of coordinate values.
- E.g., Auckland on a map: $y = -36°52'$ latitude (south) $x = 174°45'$ longitude (east)
- 3D point – a triple $(x, y, z)$.



# Points and Vectors

**Vector** – a displacement / difference between two points:

- Direction+length of displacing point $P_2$ relative to point $P_1$:


- Example: *Where is Hamilton*?
  - **Point**:
    $-39°43'$ latitude;
    $175°19'$ longitude.
  - **Vector**:
    120 km to the
    south-south-west of Auckland.

# Representing Points and Vectors

**Points** are represented by tuples:

2D: 2-tuples $(x, y)$ with $x$ and $y$ coordinates

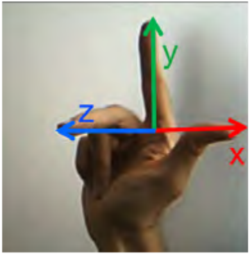
3D: 3-tuples $(x, y, z)$ with $x, y, z$ coordinates

**Vectors** are also represented as tuples,

but written usually as a column, rather than a row:

$$\mathbf{v} = \begin{bmatrix} x \\ y \end{bmatrix}$$

with $x$ and $y$ component
(in 3D also $z$ component)




Right-handed coordinate system

**Position vector** of a point: the vector from the origin to the point.

- Often convenient to use position vectors instead of points.

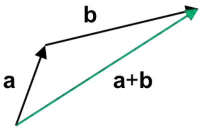
**Our notation**:

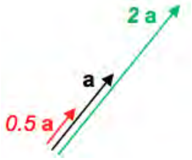
- Points are written in capital letters, e.g. P
- Vectors in small bold letters, e.g. position vector of P is $\mathbf{p}$

# Operations on Points and Vectors

**Vectors**

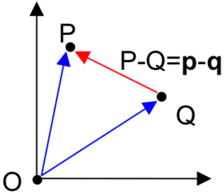
- Add, subtract
- Scale (change length)

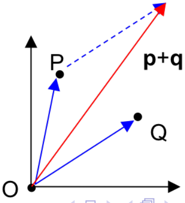



**Points**

- Subtracting one point from another gives a vector (displacement)
- **Cannot** add two points: Auckland + Hamilton = ???
- But can add and subtract their position vectors:



# Basic Operations on Vectors

**Addition**:

- Representing the combined displacement.
- Add the corresponding components.

$$\mathbf{u} \quad + \quad \mathbf{v} \quad =$$

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} u_1 + v_1 \\ u_2 + v_2 \end{bmatrix}$$

**Subtraction**:

- Same as adding a negated vector, i.e. one in the opposite direction.
- Subtract the corresponding components.

$$\mathbf{u} \quad - \quad \mathbf{v} \quad =$$

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

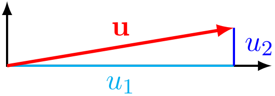
# Basic Operations on Vectors

**Scaling**:

- Changing the length (magnitude).
- Defined such that $\mathbf{v} + \mathbf{v} = 2\mathbf{v}$.
- Multiply all components by the scalar.

$$s\mathbf{u} = s\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} s \cdot u_1 \\ s \cdot u_2 \end{bmatrix}$$

**Magnitude** of a vector – its length or quadratic $(L_2)$ norm:




$$|\mathbf{u}| = \sqrt{u_1^2 + u_2^2}; \qquad |s\mathbf{u}| = |s||\mathbf{u}|$$

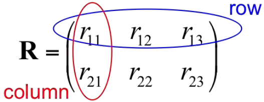
**Normalization**: $\widehat{\mathbf{u}} = \frac{\mathbf{u}}{|\mathbf{u}|}$, i.e., $|\widehat{\mathbf{u}}| = 1$

- Scaling a vector to make it of the length 1 (the unit vector).
- The scale by reciprocal of the magnitude.

# Matrices

**Matrix**: several vectors stuck together...

- $m \times n$ matrix has $m$ rows and $n$ columns.
- Like $m$ row vectors or $n$ column vectors.




**Operations**:

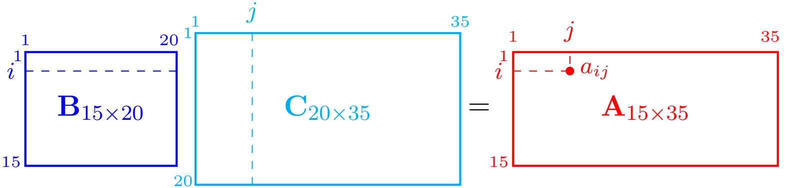
- **Addition / Subtraction –**
  like adding / subtracting several vectors at the same time:

$$\mathbf{M} \pm \mathbf{N} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \pm \begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} = \begin{bmatrix} m_{11} \pm n_{11} & m_{12} \pm n_{12} \\ m_{21} \pm n_{21} & m_{22} \pm n_{22} \end{bmatrix}$$

- **Scaling** – like scaling several vectors at the same time:

$$s\mathbf{M} = \begin{bmatrix} s \cdot m_{11} & s \cdot m_{12} \\ s \cdot m_{21} & s \cdot m_{22} \end{bmatrix}$$

# Matrix Multiplication: $\mathbf{BC} = \mathbf{A}$




Multiplying an $l \times m$ matrix $\mathbf{B}$ to an $m \times n$ matrix $\mathbf{C}$ to get an $l \times n$ matrix $\mathbf{A}$ with elements:

$$a_{ij} = b_{i1}c_{1j} + \ldots + b_{im}c_{mj} \equiv \sum_{k=1}^{m} b_{ik}c_{kj}$$

"Rows times columns" with the products summed up.

- Elements of $\mathbf{A}$ are **dot products** of the row vectors of $\mathbf{B}$ and column vectors of $\mathbf{C}$: $a_{ij} = [b_{i1} \ldots b_{im}] \begin{bmatrix} c_{1j} \\ \ldots \\ c_{mj} \end{bmatrix}$

# Matrix Multiplication: $\mathbf{BC} = \mathbf{A}$

Can be used to transform several vectors simultaneously:

$$\mathbf{B}\begin{bmatrix} c_{11} \\ c_{21} \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} \qquad \mathbf{B}\begin{bmatrix} c_{12} \\ c_{22} \end{bmatrix} = \begin{bmatrix} a_{12} \\ a_{22} \end{bmatrix}$$

Example: $l = m = n = 2$

$$\underbrace{\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}}_{\mathbf{B}} \underbrace{\begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}}_{\mathbf{C}} = \underbrace{\begin{bmatrix} b_{11}c_{11} + b_{12}c_{21} & b_{11}c_{12} + b_{12}c_{22} \\ b_{21}c_{11} + b_{22}c_{21} & b_{21}c_{12} + b_{22}c_{22} \end{bmatrix}}_{\mathbf{A}=\mathbf{BC}}$$

Numerical example:

$$\underbrace{\begin{bmatrix} 2 & -1 \\ 1 & 3 \end{bmatrix}}_{\mathbf{B}_{2\times 2}} \underbrace{\begin{bmatrix} 4 & 0 \\ -2 & 5 \end{bmatrix}}_{\mathbf{C}_{2\times 2}} = \underbrace{\begin{bmatrix} 2\cdot 4 + (-1)\cdot(-2) & 2\cdot 0 + (-1)\cdot 5 \\ 1\cdot 4 + 3\cdot(-2) & 1\cdot 0 + 3\cdot 5 \end{bmatrix} = \begin{bmatrix} 10 & -5 \\ -2 & 15 \end{bmatrix}}_{\mathbf{A}_{2\times 2}=\mathbf{B}_{2\times 2}\mathbf{C}_{2\times 2}}$$

# Identity Matrix and Inverse Matrix

**Identity matrix** $\mathbf{I}$ – the neutral element of matrix multiplication:

- For all square matrices $\mathbf{M}$: $\mathbf{IM} = \mathbf{MI} = \mathbf{M}$
- The $2 \times 2$ identity matrix $\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

**Inverse matrix** $\mathbf{M}^{-1}$ of a square matrix $\mathbf{M}$:

- It does not always exist.
- If it exists, then: $\mathbf{MM}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$ and $\left(\mathbf{M}^{-1}\right)^{-1} = \mathbf{M}$

Inverse of a $2 \times 2$ matrix:

$$\begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}^{-1} = \frac{1}{m_{11}m_{22} - m_{12}m_{21}} \begin{bmatrix} m_{22} & -m_{12} \\ -m_{21} & m_{11} \end{bmatrix}$$

exists only if the determinant $m_{11}m_{22} - m_{12}m_{21} \neq 0$

# Transpose Operation $^\mathsf{T}$ (Matrix/Vector Transposition)

Make rows out of columns (or vice versa).

- Transpose of a row vector is a column vector (and vice versa):

$$\mathbf{u} = [u_1\ u_2] \Longrightarrow \mathbf{u}^\mathsf{T} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

- For a matrix $\mathbf{M}$, swap $m_{ij}$ and $m_{ji}$ for all $i = 1..m$, $j = 1..n$:

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \end{bmatrix} \Longrightarrow \mathbf{M}^\mathsf{T} = \begin{bmatrix} m_{11} & m_{21} \\ m_{12} & m_{22} \\ m_{13} & m_{23} \end{bmatrix}$$

**Transpose rules**:

$$(\mathbf{M}^\mathsf{T})^\mathsf{T} = \mathbf{M} \qquad (s\mathbf{M})^\mathsf{T} = s\,(\mathbf{M}^\mathsf{T})$$

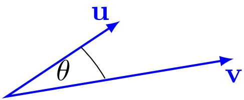
$$(\mathbf{M}+\mathbf{N})^\mathsf{T} = \mathbf{M}^\mathsf{T} + \mathbf{N}^\mathsf{T} \qquad (\mathbf{M}\mathbf{N})^\mathsf{T} = \mathbf{N}^\mathsf{T}\mathbf{M}^\mathsf{T}$$

# Dot, or Scalar Product $\mathbf{u}\bullet\mathbf{v}$

Produce a scalar (a single number) from two vectors $\mathbf{u}$ and $\mathbf{v}$:

$$\mathbf{u}\bullet\mathbf{v} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \bullet \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = u_1 v_1 + u_2 v_2$$

$$= \mathbf{u}^\mathsf{T}\mathbf{v} = |\mathbf{u}||\mathbf{v}|\cos(\theta)$$




$\theta$ – the angle between $\mathbf{u}$ and $\mathbf{v}$

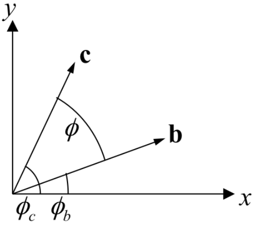
**Rules**:

| | |
|---|---|
| $\mathbf{a}\bullet\mathbf{b} = \mathbf{b}\bullet\mathbf{a}$ | Symmetry |
| $(\mathbf{a}+\mathbf{b})\bullet\mathbf{c} = \mathbf{a}\bullet\mathbf{c} + \mathbf{b}\bullet\mathbf{c}$ | Linearity |
| $(s\mathbf{a})\bullet\mathbf{b} = s(\mathbf{a}\bullet\mathbf{b})$ | Homogeneity |
| $\mathbf{b}\bullet\mathbf{b} = |\mathbf{b}|^2$ | |

Example: $|\mathbf{a}-\mathbf{b}|^2 = (\mathbf{a}-\mathbf{b})\bullet(\mathbf{a}-\mathbf{b}) = \mathbf{a}\bullet\mathbf{a} - 2\mathbf{a}\bullet\mathbf{b} + \mathbf{b}\bullet\mathbf{b}$

# Angle between Two Vectors

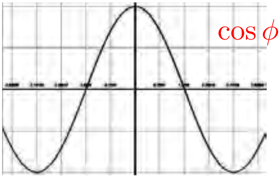
Most important dot product application: find the angle between two vectors (or two intersecting lines):




$$\mathbf{b} = \begin{bmatrix} |\mathbf{b}|\cos\phi_b \\ |\mathbf{b}|\sin\phi_b \end{bmatrix}; \quad \mathbf{c} = \begin{bmatrix} |\mathbf{c}|\cos\phi_c \\ |\mathbf{c}|\sin\phi_c \end{bmatrix}$$

hence

$$\begin{aligned} \mathbf{b}\bullet\mathbf{c} &= |\mathbf{b}||\mathbf{c}|\cos\phi_b\cos\phi_c + |\mathbf{b}||\mathbf{c}|\sin\phi_b\sin\phi_c \\ &= |\mathbf{b}||\mathbf{c}|\cos(\phi_b - \phi_c) = |\mathbf{b}||\mathbf{c}|\cos\phi \end{aligned}$$



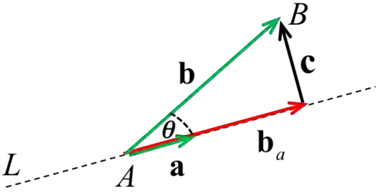

Two non-zero vectors $\mathbf{b}$ and $\mathbf{c}$ with common start point are:

| | | |
|---|---|---|
| less than | $90^\circ$ apart if $\mathbf{b}\bullet\mathbf{c} > 0$ | |
| exactly | $90^\circ$ apart if $\mathbf{b}\bullet\mathbf{c} = 0$ | [$\mathbf{b}$ and $\mathbf{c}$ are *orthogonal* (*perpendicular*)] |
| more than | $90^\circ$ apart if $\mathbf{b}\bullet\mathbf{c} < 0$ | |

# Orthogonal Projection of a Vector

Projecting a vector $\mathbf{b}$ onto a vector $\mathbf{a}$:

- $L$ – a line through $A$ in direction of $\mathbf{a}$
- $\mathbf{b}$ – the vector from $A$ to $B$




**Given**: $\mathbf{a}$ and $\mathbf{b}$

**Find**: $\mathbf{b_a}$ (the orthogonal projection of $\mathbf{b}$ onto $\mathbf{a}$)

**Solution**:

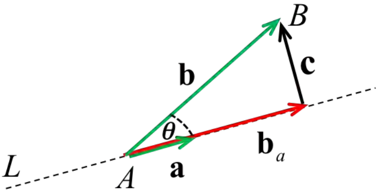
1. Length of $\mathbf{b_a}$: $|\mathbf{b_a}| = |\mathbf{b}| \cos\theta = \frac{\mathbf{a} \bullet \mathbf{b}}{|\mathbf{a}|}$ by definition of dot product: $\mathbf{a} \bullet \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos\theta$

2. Vector $\mathbf{b_a}$: $\mathbf{b_a} = \frac{\mathbf{a} \bullet \mathbf{b}}{|\mathbf{a}|} \frac{\mathbf{a}}{|\mathbf{a}|} = \frac{\mathbf{a} \bullet \mathbf{b}}{\mathbf{a} \bullet \mathbf{a}} \mathbf{a}$ because $\mathbf{a} \bullet \mathbf{a} = |\mathbf{a}|^2$

# Distance from a Line to a Point

Projecting a vector $\mathbf{b}$ onto a vector $\mathbf{a}$:

- $L$ – a line through $A$ in direction of $\mathbf{a}$
- $\mathbf{b}$ – the vector from $A$ to $B$




**Given**: $\mathbf{a}$ and $\mathbf{b}$

**Find**: $\mathbf{c}$ (the perpendicular from $L$ to $B$)

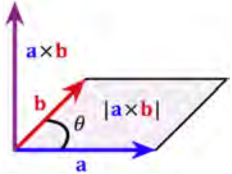
**Solution**:

$$|\mathbf{c}| = |\mathbf{b} - \mathbf{b_a}| = \left|\mathbf{b} - \frac{\mathbf{a} \bullet \mathbf{b}}{\mathbf{a} \bullet \mathbf{a}}\mathbf{a}\right|$$

# Cross, or Vector Product $\mathbf{u} \times \mathbf{v}$

Produce a 3D vector from two 3D vectors $\mathbf{u}$ and $\mathbf{v}$:

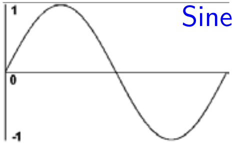
$$\mathbf{a}\times\mathbf{b} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix} = (|\mathbf{a}||\mathbf{b}|\sin\theta)\,\mathbf{n}$$




- $\theta$ – the angle between $\mathbf{a}$ and $\mathbf{b}$
- $\mathbf{n}$ – the unit normal vector ($|\mathbf{n}| = 1$) orthogonal to $\mathbf{a}$ and $\mathbf{b}$
- Hard to remember? Memorise its meaning, not formula!

**Rules**:

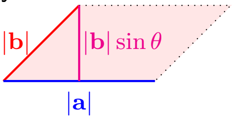
| | |
|---|---|
| $(\mathbf{a}+\mathbf{b})\times\mathbf{c} = \mathbf{a}\times\mathbf{c} + \mathbf{b}\times\mathbf{c}$ | Linearity |
| $(s\mathbf{a})\times\mathbf{b} = s(\mathbf{a}\times\mathbf{b})$ | Homogeneity |
| $\mathbf{a}\times\mathbf{b} = -(\mathbf{b}\times\mathbf{a})$ | Asymmetry |



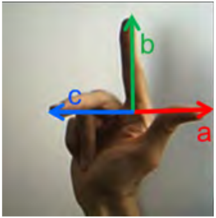
# Cross Product $\mathbf{a} \times \mathbf{b} = (|\mathbf{a}||\mathbf{b}| \sin\theta)\,\mathbf{n}$: Properties

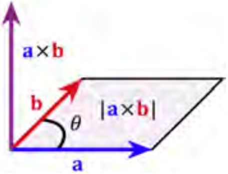
1. Vector $\mathbf{a} \times \mathbf{b}$ is perpendicular/orthogonal to both $\mathbf{a}$ and $\mathbf{b}$.
2. Direction of $\mathbf{a} \times \mathbf{b}$ is given by the "right-hand rule".
3. Asymmetry: $\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}$
4. Magnitude $|\mathbf{a} \times \mathbf{b}|$ – the area of parallelogram defined by $\mathbf{a}$ and $\mathbf{b}$:

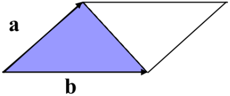
$$|\mathbf{a}\times\mathbf{b}| = |\mathbf{a}||\mathbf{b}| \sin\theta$$




5. $0.5|\mathbf{a} \times \mathbf{b}|$ – the area of triangle defined by $\mathbf{a}$ and $\mathbf{b}$



# Summary 1

1. **Vectors**:
   addition, subtraction, scaling, magnitude, normalisation.
2. **Matrices**:
   addition, subtraction, scaling, transposition, multiplication.
3. **Dot product**: $\mathbf{u} \bullet \mathbf{v} = \mathbf{u}^\mathsf{T}\mathbf{v} = |\mathbf{u}||\mathbf{v}|\cos\theta$
4. **Cross product**: $\mathbf{u} \times \mathbf{v} = \begin{bmatrix} u_2v_3 - u_3v_2 \\ u_3v_1 - u_1v_3 \\ u_1v_2 - u_2v_1 \end{bmatrix} = (|\mathbf{u}||\mathbf{v}|\sin\theta)\,\mathbf{n}$

**References**:

- Vectors, matrices: Hill, Chapter 4.2.
- Dot product: Hill, Chapter 4.3.
- Cross product: Hill, Chapter 4.4.

# QUIZ

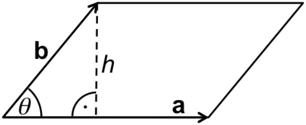
$$\mathbf{a} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} ; \quad \mathbf{b} = \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix} ; \quad \mathbf{M} = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 1 & 0 \\ 3 & 0 & 2 \end{bmatrix} ; \quad \mathbf{N} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 1 & 1 \\ -1 & 3 & -1 \end{bmatrix}$$

1. Calculate: $\mathbf{a} + \mathbf{b}$, $|\mathbf{b}|\mathbf{a}$, $\mathbf{Ma}$, $\mathbf{MN}$, $\mathbf{a}.\mathbf{b}$, $\mathbf{a} \times \mathbf{b}$.
2. What can you tell about the angle between $\mathbf{a}$ and $\mathbf{b}$?
3. What is the projection of $\mathbf{b}$ onto $\mathbf{a}$?
4. What is the distance between the point given by $\mathbf{b}$ and the line going through the origin along $\mathbf{a}$?

# Applications of • and ×: Areas and Volumes

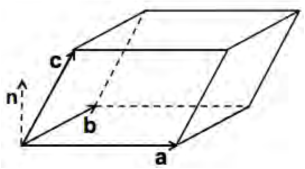
$|\mathbf{a} \times \mathbf{b}|$ – the area of a parallelogram, specified by $\mathbf{a}$ and $\mathbf{b}$:

$$\begin{aligned} |\mathbf{a} \times \mathbf{b}| &= |\mathbf{a}||\mathbf{b}|\sin(\theta)|n| \quad \Leftarrow |n| = 1 \\ &= |\mathbf{a}||\mathbf{b}|\sin(\theta) \quad \Leftarrow h = |\mathbf{b}|\sin\theta \\ &= |\mathbf{a}|h \end{aligned}$$




$(\mathbf{a} \times \mathbf{b}) \bullet \mathbf{c}$ – the volume of a parallelepiped specified by $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$:

$$\begin{aligned} (\mathbf{a} \times \mathbf{b}) \bullet \mathbf{c} &= (|\mathbf{a}||\mathbf{b}|\sin(\theta)\mathbf{n}) \bullet \mathbf{c} \\ &= (\underline{\textit{area of bottom}})\mathbf{n} \bullet \mathbf{c} \\ &= (\underline{\textit{area of bottom}})\ \underline{\textit{height}} \end{aligned}$$




Reminder: $\mathbf{n} \bullet \mathbf{c} = |\mathbf{n}||\mathbf{c}|\sin\theta$

# Coordinate Transformations

**Given:** A new coordinate system with location E and axis unit vectors $\mathbf{u}$, $\mathbf{v}$, $\mathbf{n}$

**Find:** Coordinates P′ of a point P in the new coordinate system.

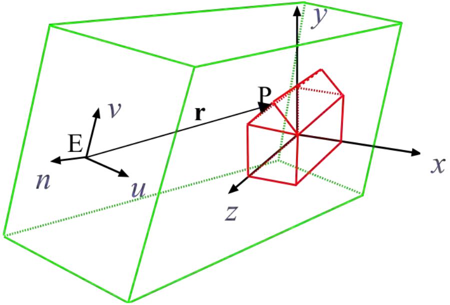
**Idea:**

1. Find position vector $\mathbf{r}$ expressing P relative to E:

$$\mathbf{r} = P - E$$

2. Project $\mathbf{r}$ onto each of the axis unit vectors to get the new coordinates:

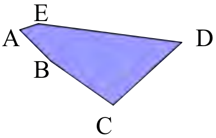
$$\mathrm{P}' = (\mathbf{r}\bullet\mathbf{u}, \mathbf{r}\bullet\mathbf{v}, \mathbf{r}\bullet\mathbf{n})$$



# Normal of a Polygon

In principle, the normal $\mathbf{n}$ can be obtained from the cross product, $\mathbf{a} \times \mathbf{b} = |\mathbf{a}||\mathbf{b}| \sin(\theta)\mathbf{n}$, of any two adjacent edge vectors, e.g.,

$$\mathbf{n} = (D - C) \times (B - C)$$



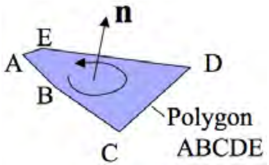

But this approach is **non-robust** – a non-representative or erroneous normal vector is computed when:

1. 3 vertices are co-linear (on a straight line).
2. 2 adjacent vertices are very close together.
3. Polygon is not coplanar (i.e., not all points are on a plane).

$\Rightarrow$ i.e., when the cross product's magnitude tends to zero and direction is sensitive to a slight movement of either vertex!

**Warning**: the above non-robustness conditions 1, 2 or 3 are not exceptional in computer graphics and occur all the time!

# Robust Normal Algorithm




**Note:** The orientation of the resulting normal is such that the vertices are listed in counterclockwise order around it.

---

Just sum together all the cross products, $\mathbf{a} \times \mathbf{b} = |\mathbf{a}||\mathbf{b}| \sin(\theta)\mathbf{n}$, of the adjacent edge vectors, i.e.,

$$\begin{aligned}(\mathrm{B}-\mathrm{A}) \times (\mathrm{E}-\mathrm{A}) + (\mathrm{C}-\mathrm{B}) \times (\mathrm{A}-\mathrm{B}) + (\mathrm{D}-\mathrm{C}) \times (\mathrm{B}-\mathrm{C}) \\ + (\mathrm{E}-\mathrm{D}) \times (\mathrm{C}-\mathrm{D}) + (\mathrm{A}-\mathrm{E}) \times (\mathrm{D}-\mathrm{E})\end{aligned}$$

and normalise the result.

---

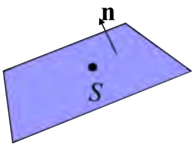
**Robustness**:

- Short edges or nearly co-linear vertex triples give negligible cross product contribution.
- Long nearly-perpendicular edges give the biggest contribution.

# Point-Normal Form of a Plane

Define plane by:

1. A **point** $S$ on the plane.
2. A **normal vector** $\mathbf{n}$ orthogonal to the plane (with $|\mathbf{n}| = 1$).




For any point $P$ on the plane, $(P - S)$ is orthogonal to $\mathbf{n}$:

$$\mathbf{n}\bullet(P - S) = 0 \quad (\text{“point-normal form” of the plane equation})$$

If $\mathbf{p}$ and $\mathbf{s}$ are the position vectors to $P$ and $S$:

$$\mathbf{n}\bullet(\mathbf{p} - \mathbf{s}) = 0 \;\Leftrightarrow\; \mathbf{n}\bullet\mathbf{p} = \mathbf{n}\bullet\mathbf{s} \;\Leftrightarrow\; \mathbf{n}\bullet\mathbf{p} = d \;\text{ where } d = \mathbf{n}\bullet\mathbf{s}$$

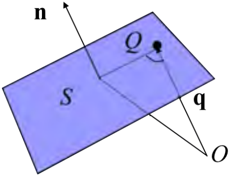
If $\mathbf{n} = [a, b, c]^\mathsf{T}$ and $\mathbf{p} = [x, y, z]^\mathsf{T}$, then this is the familiar 3D plane equation $ax + by + cz = d$

# Distance to a Plane from the Origin

- Let $Q$ be a point on the plane, such that the position vector $\mathbf{q}$ is parallel to the plane normal $\mathbf{n}$.
- Then $|\mathbf{q}|$ is the “shortest distance” to the plane from the origin.

The plane equation $\mathbf{n}\bullet\mathbf{p} = d$ is valid for every point $P$ on plane:

$$\begin{aligned} \mathbf{n}\bullet\mathbf{q} &= d && (Q \text{ is on the plane}) \\ \mathbf{n}\bullet\mathbf{q} &= |\mathbf{n}||\mathbf{q}|\cos 0^\circ && (\mathbf{n} \text{ is parallel to } \mathbf{q}) \\ &= |\mathbf{q}| && (|\mathbf{n}| = 1 \text{ and } \cos 0^\circ = 1) \\ &\Rightarrow |\mathbf{q}| = d \end{aligned}$$



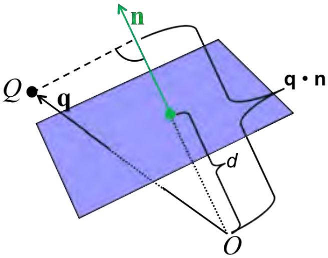

**Conclusion**:
Provided that $\mathbf{n} = [a, b, c]^\mathsf{T}$ is a unit vector, $d$ is the distance to the plane from the origin in the plane equation

$$\mathbf{n}\bullet\mathbf{p} = d \;\Leftrightarrow\; ax + by + cz = d.$$

# Distance of a Point from a Plane

How far is a point $Q$ from the plane $\mathbf{n}\bullet\mathbf{p} = d$ with the normal $\mathbf{n}$?

- The shortest line from $Q$ to the plane is parallel to $\mathbf{n}$.
- Project the position vector $\mathbf{q}$ of $Q$ onto $\mathbf{n}$:




$$\mathbf{q}\bullet\mathbf{n} = \text{the distance along } \mathbf{n} \text{ from } Q \text{ to the origin } O$$

To get only the distance of $Q$ from the plane, subtract the distance $d$ of the origin $O$ from the plane:

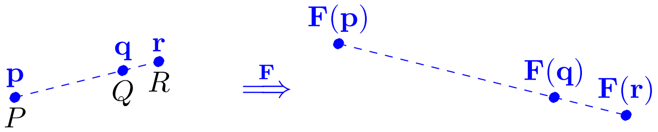
$$\mathbf{q}\bullet\mathbf{n} - d = \text{the distance along } \mathbf{n} \text{ from } Q \text{ to the plane}$$

(for the unit normal $|\mathbf{n}| = 1$).

# 2D Affine Transformations $\mathbf{F}(\mathbf{p}) = \mathbf{M}\mathbf{p} + \mathbf{t}$

Vector $\mathbf{F}(\mathbf{p})$ by linear transformation and translation of a vector $\mathbf{p}$:

- The linear transformation is a matrix multiplication: $\mathbf{M}\mathbf{p}$
- The translation is a vector addition: $\ldots + \mathbf{t}$




**Properties** of the affine transformation $\mathbf{F}$:

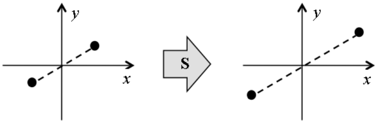
1. Preserves **collinearity**: if $P$, $Q$, $R$ are on a straight line, then also $\mathbf{F}(\mathbf{p})$, $\mathbf{F}(\mathbf{q})$, $\mathbf{F}(\mathbf{r})$.
2. Preserves **ratios of distances** along a line: if $P$, $Q$, $R$ are on a straight line, then

$$\frac{|Q-P|}{|R-Q|} \equiv \frac{|\mathbf{q}-\mathbf{p}|}{|\mathbf{r}-\mathbf{q}|} = \frac{|\mathbf{F}(\mathbf{q})-\mathbf{F}(\mathbf{p})|}{|\mathbf{F}(\mathbf{r})-\mathbf{F}(\mathbf{q})|}$$

# Scaling S and Translation T

**S**: squeezing and stretching along the $x$- and $y$-axis about the origin.

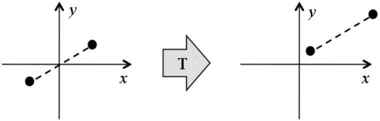
- Scaling factor $s_x$ / $s_y$ along the $x$- / $y$-axis.
- Scaling factor $< 1$ – squeezing.
- Scaling factor $> 1$ – stretching.




$$\mathbf{S}(\mathbf{p}) = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} s_x p_x \\ s_y p_y \end{bmatrix}$$

**T**: moving along the $x$- and $y$-axes.

- Distance (shift) $t_x$ / $t_y$ along the $x$- / $y$-axis.

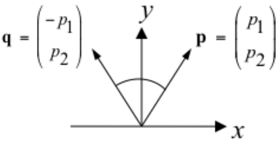



$$\mathbf{T}(\mathbf{p}) = \mathbf{I} \begin{bmatrix} p_x \\ p_y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} p_x + t_x \\ p_y + t_y \end{bmatrix}$$

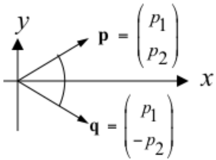
# Reflections at Axes and Origin

Special cases of scaling:

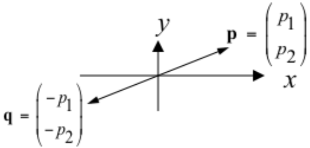
Reflection at the y-axis: $\mathbf{q} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{p}$




Reflection at the x-axis: $\mathbf{q} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \mathbf{p}$



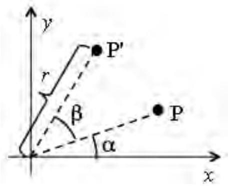

Reflection at the origin: $\mathbf{q} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \mathbf{p}$



# Rotation $\mathbf{R}$

About the origin anti-clockwise by angle $\beta$:

- $\alpha$ – an initial angle of point $\mathrm{P}$
- $\beta$ – the angle of rotation, so that $\mathrm{P}$ becomes $\mathrm{P}'$



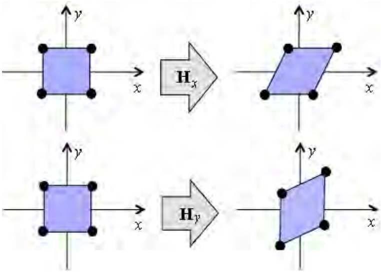

1. Coordinates of the point $\mathrm{P}$: $x = r\cos(\alpha)$; $y = r\sin(\alpha)$
2. Coordinates of the point $\mathrm{P}'$:

$$\begin{aligned} x' &= r\cos(\alpha+\beta) &= r\cos(\alpha)\cos(\beta) - r\sin(\alpha)\sin(\beta) \\ y' &= r\sin(\alpha+\beta) &= r\sin(\alpha)\cos(\beta) + r\cos(\alpha)\sin(\beta) \end{aligned}$$

3. Substitute formulae for $x$ and $y$ into $x'$ and $y'$:

$$\begin{aligned} x' &= x\cos(\beta) - y\sin(\beta) \\ y' &= y\cos(\beta) + x\sin(\beta) \end{aligned} \Longrightarrow \mathbf{R}(\mathbf{p}) = \begin{bmatrix} \cos(\beta) & -\sin(\beta) \\ \sin(\beta) & \cos(\beta) \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix}$$

# Shearing




**Horizontal shear $\mathbf{H}_x$:**

- Shifts points parallel to the $x$-axis proportionally to their $y$-coordinate.
- The further up a point, the more it is shifted to the right (or left).

Analogously: the **vertical shear $\mathbf{H}_y$**.

General shear $\mathbf{H} = \begin{bmatrix} 1 & s_x \\ s_y & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} p_x + s_x p_y \\ p_y + s_y p_x \end{bmatrix}$

- The greater the shearing factor $s_x$ or $s_y$, the stronger the horizontal or vertical shearing.
- $\mathbf{H}_x$: $s_x > 0$ and $s_y = 0$; $\mathbf{H}_y$: $s_x = 0$ and $s_y > 0$.
- Shearing preserves the area of a shape.

# Affine Transformation $\mathbf{T}(\mathbf{p}) = \mathbf{M}\mathbf{p} + \mathbf{t}$: Basic Properties

- Straight lines are preserved.
- Parallel lines remain parallel.
- Proportionality between the distances is preserved.
- Any arbitrary affine transformation can be represented as a sequence of shearing, scaling, rotation and translation.
- **Transformations generally do not commute**, i.e., $\mathbf{T}_1\mathbf{T}_2 \neq \mathbf{T}_2\mathbf{T}_1$:

$$\mathbf{T}_2(\mathbf{p}) = \mathbf{M}_2\mathbf{p} + \mathbf{t}_2 \;\Rightarrow\; \mathbf{T}_1(\mathbf{T}_2(\mathbf{p})) = \mathbf{M}_1\mathbf{M}_2\mathbf{p} + \mathbf{M}_1\mathbf{t}_2 + \mathbf{t}_1$$
$$\mathbf{T}_1(\mathbf{p}) = \mathbf{M}_1\mathbf{p} + \mathbf{t}_1 \;\Rightarrow\; \mathbf{T}_2(\mathbf{T}_1(\mathbf{p})) = \mathbf{M}_2\mathbf{M}_1\mathbf{p} + \mathbf{M}_2\mathbf{t}_1 + \mathbf{t}_2$$

- Transformations **are** associative, $\mathbf{T}_1(\mathbf{T}_2\mathbf{T}_3) = (\mathbf{T}_1\mathbf{T}_2)\mathbf{T}_3$:

$$\begin{aligned}\mathbf{T}_1\mathbf{T}_2\mathbf{T}_3(\mathbf{p}) &= \mathbf{M}_1\mathbf{M}_2\mathbf{M}_3\mathbf{p} + \mathbf{M}_1\mathbf{M}_2\mathbf{t}_3 + \mathbf{M}_1\mathbf{t}_2 + \mathbf{t}_1 \\ &= \mathbf{M}_1\left(\mathbf{M}_2\left(\mathbf{M}_3\mathbf{p} + \mathbf{t}_3\right) + \mathbf{t}_2\right) + \mathbf{t}_1\end{aligned}$$

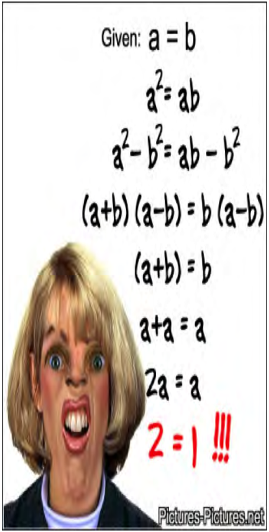
# Summary 2

1. Applications of the dot ($\bullet$) and cross ($\times$) products: areas and volumes, coordinate transformations, normals.
2. Planes
    1. Point-normal form: $\mathbf{n} \bullet \mathbf{p} = d$ with $d =$ *distance to the origin*
    2. Distance from a point $Q$ to plane: $\mathbf{q} \bullet \mathbf{n} - d$
3. 2D affine transformations: $\mathbf{F}(\mathbf{p}) = \mathbf{M}\mathbf{p} + \mathbf{t}$: scaling, translation, rotation, shearing.

**References**:

- Dot product: Hill, Chapter 4.3
- Cross product: Hill, Chapter 4.4
- Introduction to affine transformations: Hill, Chapter 5.2

# Quiz




1. Transform $P = (2, 2, -1)$ to the new coordinate system with the axis vectors $\mathbf{u} = [0, 1, 0]^\mathsf{T}$, $\mathbf{v} = [0, 0, -1]^\mathsf{T}$, $\mathbf{w} = [-1, 0, 0]^\mathsf{T}$ and origin $E = (0, 2, 0)$.
2. How far is the plane $3x + y - 2z = 5$ from the origin $(0, 0, 0)$?
3. How far is the point $Q = (3, 4, 2)$ from the plane $3x + y - 2z = 5$?
4. Transform the point $R = (1, 2)$: scale it along the $y$-axis with factor 0.5; move it up the $y$-axis by 4; then shear it vertically by 2.

# Homogeneous Coordinates

Cartesian 2D $(x, y)$-coordinates: $P = (x, y)$ $\Leftrightarrow$ Homogeneous 2D coordinates $P = (x, y, 1)$ or $(xw, yw, w)$; $w \neq 0$:

$$\begin{bmatrix} x \\ y \end{bmatrix} \longleftrightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \equiv \begin{bmatrix} xw \\ yw \\ w \end{bmatrix}$$

Cartesian 3D $(x, y, z)$-coordinates: $P = (x, y)$ $\Leftrightarrow$ Homogeneous 3D coordinates $P = (x, y, z, 1)$ or $(xw, yw, zw, w)$; $w \neq 0$:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \longleftrightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \equiv \begin{bmatrix} xw \\ yw \\ zw \\ w \end{bmatrix}$$

# Homogeneous Coordinates: Why?

- Affine transformation $\mathbf{F}$ consists of a linear (matrix) transformation and a translation: $\mathbf{F}(\mathbf{p}) = \mathbf{M}\mathbf{p} + \mathbf{t}$
- **Goal**: Represent translations with a matrix, too: $\mathbf{F}(\mathbf{p}) = \mathbf{M}\mathbf{p}$

**Solution** – Homogeneous coordinates:

- Add to every vector an additional coordinate $w$, which is initially set to 1: $\mathbf{p} = \begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$
- Also add another row and column to the matrices, specifying the transformations, e.g.,

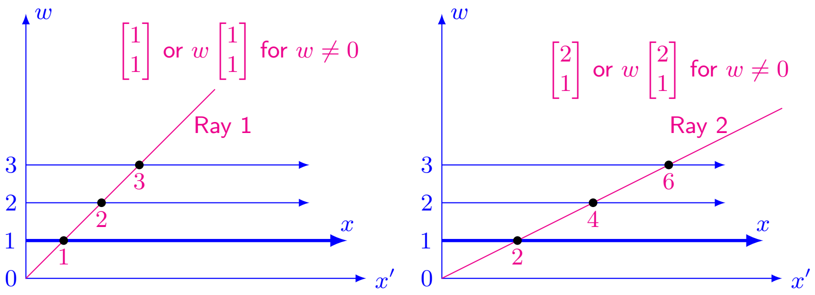
$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \longrightarrow \begin{bmatrix} m_{11} & m_{12} & 0 \\ m_{21} & m_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

# 1D Homogeneous Coordinates

Cartesian (inhomogeneous) 1D coordinate $x$:

- A point is represented by a single value, e.g., $x = 1$.

Homogeneous 1D coordinates represent the same 1D point by a 2D vector $[x', w]^\mathsf{T}$ or $\left[\frac{x'}{w}, 1\right]^\mathsf{T}$, which defines a 2D ray:

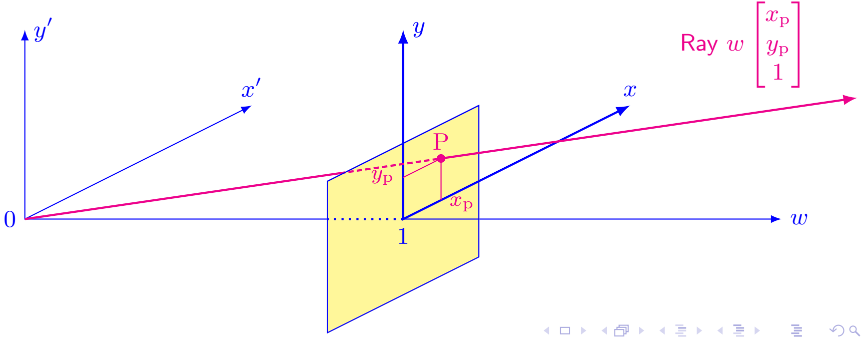


# 2D Homogeneous Coordinates

Cartesian (inhomogeneous) 2D coordinates:

- A point $\mathrm{P}$ is represented by a 2D vector, e.g., $[x_\mathrm{p}, y_\mathrm{p}]^\mathsf{T}$.

The same 2D point is represented by a homogeneous vector $[x', y', w]^\mathsf{T}$ or multiple of the vector $\left[\frac{x'}{w}, \frac{y'}{w}, 1\right]^\mathsf{T}$, defining a 3D ray:



# Using Homogeneous Coordinates

1. Every vector gets an additional coordinate with value 1.
2. Every matrix gets an additional row and column $(0, \ldots, 0, 1)$.

For affine transformations other than translations, no difference:

$$\underbrace{\begin{bmatrix} a & b \\ c & d \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\mathbf{p}} = \underbrace{\begin{bmatrix} ax+by \\ cx+dy \end{bmatrix}}_{\mathbf{Mp}} \Rightarrow \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} ax+by \\ cx+dy \\ 1 \end{bmatrix}$$

Converting translation vector $\mathbf{t}$ into translation matrix $\mathbf{T}$:

$$\overbrace{\begin{bmatrix} x \\ y \end{bmatrix}}^{\mathbf{p}} + \overbrace{\begin{bmatrix} t_x \\ t_y \end{bmatrix}}^{\mathbf{t}} = \overbrace{\begin{bmatrix} x+t_x \\ y+t_y \end{bmatrix}}^{\mathbf{p+t}} \Rightarrow \mathbf{T} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} x+t_x \\ y+t_y \\ 1 \end{bmatrix}$$

# Converting Coordinates

**Cartesian (ordinary) to homogeneous coordinates**:

- Just add another coordinate (often called $w$-coodinate): e.g., $[x, y, z]^\mathsf{T} \rightarrow [x, y, z, 1]^\mathsf{T}$.

**Homogeneous to ordinary coordinates**:

- Divide all other coordinates by $w$-coordinate (if $w \neq 0$): e.g.,

$$[x, y, z, w]^\mathsf{T} \rightarrow \left[\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right]^\mathsf{T}$$

- All homogenous 2D coordinate points $[wp_1, wp_2, w]^\mathsf{T}$ with $w \neq 0$ represent the same ordinary 2D point $[p_1, p_2]^\mathsf{T}$.
- Usually (e.g., for affine transformations) $w = 1$, so the conversion means just omitting the $w$-coordinate.

# Conversion Examples

The ordinary 3D point $[5, 3, 2]^\top$ has the homogeneous representation $[5w, 3w, 2w, w]^\top$ with an arbitrary factor $w \neq 0$, e.g.,

$$\begin{bmatrix} 5 \\ 3 \\ 2 \\ 1 \end{bmatrix}, \text{ or } \begin{bmatrix} 15 \\ 9 \\ 6 \\ 3 \end{bmatrix}, \text{ or } \begin{bmatrix} -55 \\ -33 \\ -22 \\ -11 \end{bmatrix}, \text{ or } \begin{bmatrix} 0.05 \\ 0.03 \\ 0.02 \\ 0.01 \end{bmatrix}, \text{ and so on.}$$

Conversely, the homogeneous vector $[900, 300, 450, 150]^\top$ and all other vectors of the form $[6\alpha, 2\alpha, 3\alpha, \alpha]^\top$ with $\alpha \neq 0$ represent the same 3D point $[6, 2, 3]^\top$; i.e., $\frac{900}{150} = 6$; $\frac{300}{150} = 2$; and $\frac{450}{150} = 3$.

- In homogeneous coordinates **projective transformations** as well as **affine transformations** (e.g. translations, rotations, scaling) are specified by linear equations.

# 3D Affine Transformations

Mostly analogous to 2D and represented by a left-multiplied matrix $\mathbf{M}$ in homogeneous coordinates, too: $\mathbf{Mv}$.

**Translation $\mathbf{T}$ by a vector $\mathbf{t} = [t_x, t_y, t_z]^\mathsf{T}$:**

- Similar to identity matrix.
- The rightmost column contains $\mathbf{t}$.

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**Scaling $\mathbf{S}$ about the origin with scaling factors $s_x$, $s_y$, $s_z$:**

- Similar to identity matrix.
- Scaling factors at main diagonal.
- Negative $s_x$, $s_y$, or $s_z$ reflect on the $x = 0$, $y = 0$, or $z = 0$ plane.

$$\mathbf{S} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

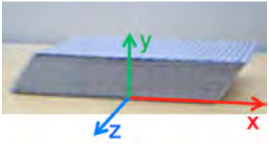
# 3D Affine Transformations

## General **shearing** $\mathbf{H}$:

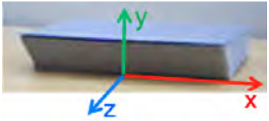
- Any coordinate $(x/y/z)$ can linearly influence any other coordinate.
- $h_{yx}$ expresses how much $y$ influences $x$.

$$\mathbf{H} = \begin{bmatrix} 1 & h_{yx} & h_{zx} & 0 \\ h_{xy} & 1 & h_{zy} & 0 \\ h_{xz} & h_{yz} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

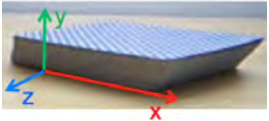
**Examples:**




$h_{yx} > 0$;
all others $= 0$




$h_{yz} > 0$;
all others $= 0$

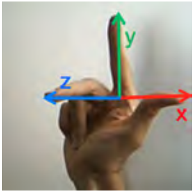


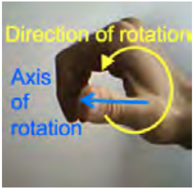

$h_{yx} > 0$; $h_{yz} > 0$;
all others $= 0$

# 3D Affine Transformations: Rotation

Rotations are the most difficult transformations.

- We will consider three rotation situations:
  1. Rotation around the three coordinate axes $(x, y, z)$.
  2. Rotation to align an object with a new coordinate system.
  3. Rotation around an arbitrary axis.
- We use a right-handed coordinate system.
- We use positive (right-handed) rotation, i.e. counterclockwise when looking into an axis.



# 1. Rotating Around Coordinate Axes $(x, y, z)$

Three matrices for positive (right-handed) rotation
($C$ and $S$ stand for $\cos\theta$ and $\sin\theta$, respectively).

**Rotation about $x$-axis:**

$$\mathbf{R}_x = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & C & -S & 0 \\ 0 & S & C & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

**Rotation about $y$-axis:**

$$\mathbf{R}_y = \left[\begin{array}{ccc|c} C & 0 & S & 0 \\ 0 & 1 & 0 & 0 \\ -S & 0 & C & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

**Rotation about $z$-axis:**

$$\mathbf{R}_z = \left[\begin{array}{ccc|c} C & -S & 0 & 0 \\ S & C & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

**Notes on $3 \times 3$ rotation matrices**:

Row and column corresponding to rotation axis are as for the identity $\mathbf{I}$.

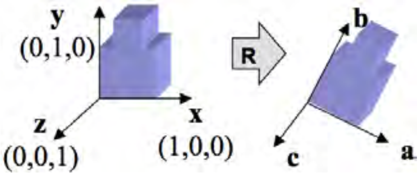
Other elements are $C$ on and $\pm S$ off diagonal, so that $\mathbf{R} = \mathbf{I}$ if $\theta = 0$.

Sign of $S$ can be inferred from the fact that rotation around $x$, $y$, $z$ by $\theta = 90^\circ$ transforms $y \to z$, $z \to x$, $x \to y$, respectively.

# 2. Rotating to Align with New Coordinate Axes

**Find**: the matrix $\mathbf{R}$ that rotates the coordinate system to align with a new coordinate system $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ with the same origin.

- $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ – unit vectors along the axes of the old system.
- $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ – unit vectors along the axes of the new system.




**Solution**:

$\mathbf{R}_{3\times3}$ should do the following:

$$\mathbf{R}_{3\times3}[1\ \ 0\ \ 0]^\mathsf{T} = \mathbf{a}$$
$$\mathbf{R}_{3\times3}[0\ \ 1\ \ 0]^\mathsf{T} = \mathbf{b}$$
$$\mathbf{R}_{3\times3}[0\ \ 0\ \ 1]^\mathsf{T} = \mathbf{c}$$

Using homogeneous coordinates:

$$\underbrace{\left[\begin{array}{c|c} \mathbf{R}_{3\times3} & \mathbf{0} \\ \hline \mathbf{0}^\mathsf{T} & 1 \end{array}\right]}_{\mathbf{R}} \left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \left[\begin{array}{ccc|c} a_x & b_x & c_x & 0 \\ a_y & b_y & c_y & 0 \\ a_z & b_z & c_z & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right] = \mathbf{R}$$

# 3. Rotating About an Arbitrary Axis

- Often need to rotate an object about some arbitrary axis through a reference point on it.
- E.g., forearm of robot rotating around an axis through the elbow.

Involves three steps:

1. Translate the reference point to the origin.
2. Do the rotation.
3. Translate the reference point back again.

- Translation is easy (steps 1 and 3).

*We know how to rotate about coordinate axes, but how about an arbitrary axis through the origin?*

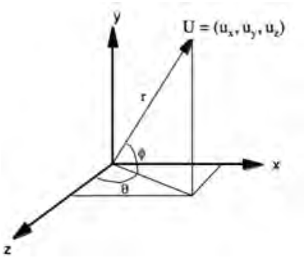
1. **Textbook method**: Decompose the rotation into primitive rotations about $x$, $y$, and $z$ axes.
2. **Coordinate system alignment method**.

# (3.1) Arbitrary Axis Rotation: Textbook

1. Rotate the object so that the required axis of rotation $\mathbf{r}$ lies along the $z$-axis ($\mathbf{R}_{\text{align}Z}$)
2. Do the rotation about $z$-axis
3. Undo original rotation ($\mathbf{R}^{-1}_{\text{align}Z}$)

How to get $\mathbf{R}_{\text{align}Z}$?

1. Measure azimuth $\theta$ as a right handed rotation about the $y$-axis, starting at the $z$-axis.
2. Measure elevation (or "latitude") $\phi$ as the angle above the plane $y = 0$.
3. $\mathbf{R}_{\text{align}Z} = \mathbf{R}_x(\phi)\mathbf{R}_y(\theta)$




$$\phi = \tan^{-1}\left(\frac{u_y}{\sqrt{u_x^2+u_z^2}}\right)$$

$$\theta = \text{atan2}(u_x, u_z)$$

i.e. a 4-quadrant $\tan^{-1}\left(\frac{u_x}{u_z}\right)$

# (3.2) Arbitrary Axis Rotation: Alignment

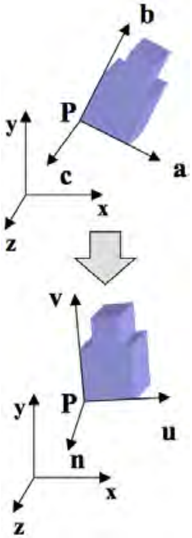
**Given**:

- Coordinate system $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ attached to the object to be rotated.
- Position $P$ of the object's coordinate system.
- New system $(\mathbf{u}, \mathbf{v}, \mathbf{n})$ to rotate the object to.

**Solution**:

1. Translate the object to the origin $(\mathbf{T}_P^{-1})$.
2. Rotate $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ to align with the world coordinate axes (inverse of the "rotate to align" case: $\mathbf{R}_{\mathbf{abc}}^{-1}$).
3. Rotate the coordinate axes to align with $(\mathbf{u}, \mathbf{v}, \mathbf{n})$ $(\mathbf{R}_{\mathbf{uvn}})$.
4. Translate the object back to the original position $(\mathbf{T}_P)$.

The full matrix: $\mathbf{T}_P \mathbf{R}_{\mathbf{uvn}} \mathbf{R}_{\mathbf{abc}}^{-1} \mathbf{T}_P^{-1}$



# The Inverse of a Rotation Matrix

Columns of a rotation matrix are unit vectors along the rotated coordinate axis directions.

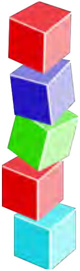
- So columns are orthogonal, i.e., their dot products $= 0$:

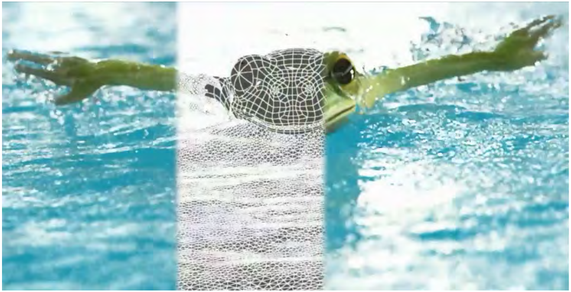
$$\underbrace{\begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix}}_{\mathbf{R}^{\mathsf{T}}_{3\times 3}} \underbrace{\begin{bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{bmatrix}}_{\mathbf{R}_{3\times 3}} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{I}_{3\times 3}}$$

$$\mathbf{R}^{\mathsf{T}}_{3\times 3}\mathbf{R}_{3\times 3} = \mathbf{I}_{3\times 3} \qquad \text{therefore, } \mathbf{R}^{-1}_{3\times 3} = \mathbf{R}^{\mathsf{T}}_{3\times 3}$$

- The inverse of a rotation matrix is its transpose.
- Matrices with this property are called **orthogonal**.

# Examples





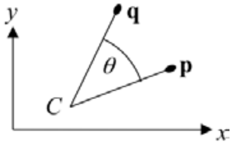
https://vimeo.com/2473185

# Composition of Transformations

- All transformations that can be represented in the matrix form.
- Combine several transformations into a single matrix by multiplying all transformation matrixes: $\mathbf{M}_n\mathbf{M}_{n-1}\cdots\mathbf{M}_1 = \mathbf{M}$
- Transformation of the rightmost matrix is applied first (i.e., $\mathbf{M}_1$).

**Example** – Rotating an object about its centre point $C$:

1. Translate the object so that its centre is at the origin ($\mathbf{M}_1$: $C \to 0$).
2. Rotate about the origin ($\mathbf{M}_2$: by angle $\theta$).
3. Translate object back to its original position ($\mathbf{M}_3$: $0 \to C$).




$$\begin{bmatrix} q_1 \\ q_2 \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & c_1 \\ 0 & 1 & c_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -c_1 \\ 0 & 1 & -c_2 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{M}=\mathbf{M}_3\mathbf{M}_2\mathbf{M}_1} \begin{bmatrix} p_1 \\ p_2 \\ 1 \end{bmatrix}$$

# Order of Transformations Does Matter!

In general, affine transformations do not commute, i.e., $\mathbf{KL} \neq \mathbf{LK}$.

(a) First scale by $(1, 2)$, then rotate $90°$:

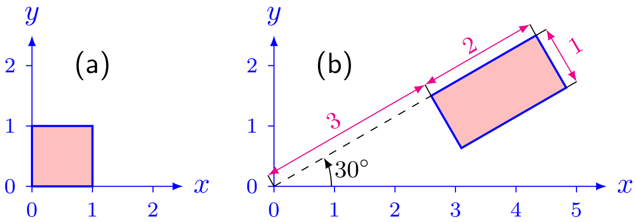
$$\mathbf{M} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -2 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(b) First rotate $90°$, then scale by $(1, 2)$:

$$\mathbf{N} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 2 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

# Question 1 [1996 exam]

Which homogeneous 2D matrix $\mathbf{M}$ transforms (a) to (b)?




- Sometimes it is easier to do this backwards, then take inverse, i.e.,

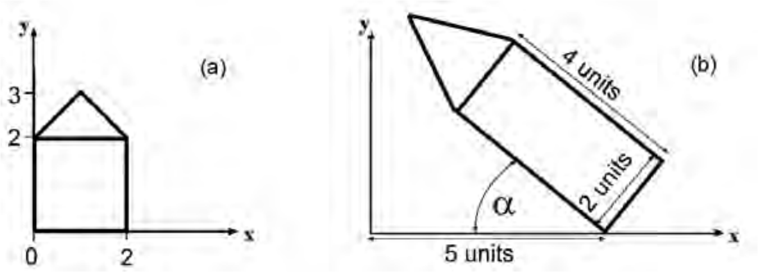
  starting with (b): Rotate $-30°$; Shift by $(-3, 1)$; Scale by $(0.5, 1)$.
- Hence the required transformation is: $\mathbf{M} = \mathbf{R}(30°)\mathbf{T}(3, -1)\mathbf{S}(2, 1)$

  (first scaling, then translation, finally rotation).
- Do not forget to use **homogeneous** matrices.

# Question 2 [2003 exam]

Which homogeneous 2D matrix $\mathbf{M}$ transforms (a) to (b)?

You are allowed to write $\mathbf{M}$ as a product of simpler matrices (i.e., you need not multiply the matrices).



# Summary 3

1. Homogeneous coordinates make it possible to represent translation as a matrix.
2. 3D affine transformations similar to 2D: translation, scaling, shearing, and rotation.
   - Column vectors of a rotation matrix $\mathbf{R}$ are axis unit vectors of a new coordinate system to align the current unit vectors $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ with.
   - $\mathbf{R}^{-1} = \mathbf{R}^{\mathsf{T}}$
3. Transformations are applied from right to left.

**References**:

- Homogeneous coordinates: Hill, Section 4.5.1
- 3D affine transformations: Hill, Section 5.3

# Quiz

1. An object has a local coordinate system
$$\mathbf{a} = (1,0,0), \quad \mathbf{b} = (0,0,-1), \quad \mathbf{c} = (0,1,0)$$
at position $(-10, 2, 5)$. Which homogeneous matrix rotates the object into the new coordinate system
$$\mathbf{u} = (0,-1,0), \quad \mathbf{v} = (0,0,-1), \quad \mathbf{n} = (1,0,0)?$$

2. Solve Questions 1 and 2 (Slides 59 and 60).
3. Create your own variant of these questions and solve it.

Count the black dots! $\Longrightarrow$