Linear Algebra

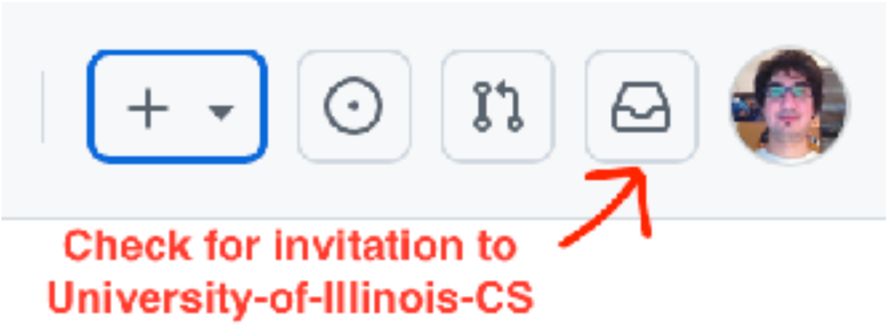
CS425: Computer Graphics I

Khairi Reda

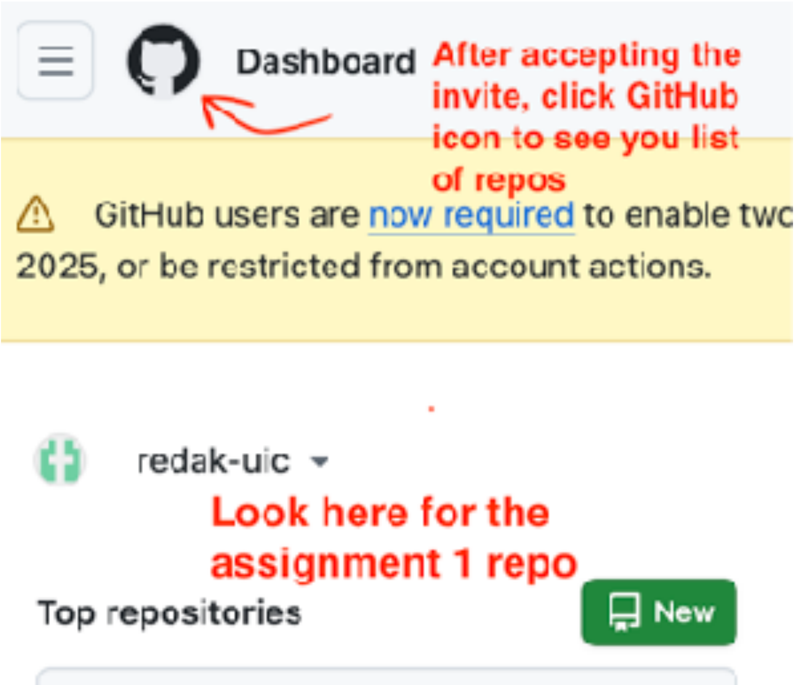
Administrativia

- Assignment 1 due **Friday the 19th (7pm sharp)**
- For those still having issues with GitHub Classroom
  - Login to your GitHub account
  - Check for an invitation from **University-of-Illinois-CS**
  - Accept invite to collaborate
  - Go to GitHub dashboard (click GitHub icon on the top-left), and check whether there's an assignment-1 repo with your NetID
  - **Test this by** cloning the repo and committing a change

1) Accept invite to Org



2) Check your repo list



Overview

- Basic linear algebra concepts
- Coordinate systems
- Coordinate frame

Euclidian space

- A n-dimensional real Euclidian space is denoted $\mathbb{R}^n$.
- A vector $\mathbf{v}$ in this space is an n-tuple (an ordered list of real numbers).

$$\mathbf{v} \in \mathbb{R}^n \iff \mathbf{v} = \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{pmatrix}, \text{ with } v_i \in \mathbb{R},\ i = 0, \ldots, n-1$$

- $v_0, \ldots, v_{n-1}$: elements, coefficients, or components of vector **v**.

Operations

$$\mathbf{u} + \mathbf{v} = \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{n-1} \end{pmatrix} + \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{pmatrix} = \begin{pmatrix} u_0 + v_0 \\ u_1 + v_1 \\ \vdots \\ u_{n-1} + v_{n-1} \end{pmatrix} \in \mathbb{R}$$

Addition

$$\alpha \mathbf{u} = \begin{pmatrix} \alpha u_0 \\ \alpha u_1 \\ \vdots \\ \alpha u_{n-1} \end{pmatrix} \in \mathbb{R}$$

Multiplication by scalar

Properties

i. $(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$ (associativity)

ii. $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ (commutativity)

iii. $\mathbf{0} + \mathbf{v} = \mathbf{v}$ (zero identity)

iv. $\mathbf{v} + (-\mathbf{v}) = \mathbf{0}$ (additive inverse)

v. $(ab)\mathbf{u} = a(b\mathbf{u})$

vi. $(a + b)\mathbf{u} = a\mathbf{u} + b\mathbf{u}$ (distributive law)

vii. $a(\mathbf{u} + \mathbf{v}) = a\mathbf{u} + a\mathbf{v}$ (distributive law)

viii. $1\mathbf{u} = \mathbf{u}$

Dot product

$$\mathbf{u} \bullet \mathbf{v} = \sum_{i=0}^{n-1} u_i v_i$$ (dot product)

Properties:

i. $\mathbf{u} \bullet \mathbf{u} \geq 0$, with $\mathbf{u} \bullet \mathbf{u} = 0$ if and only if $\mathbf{u} = (0,0, \ldots, 0) = \mathbf{0}$

ii. $(\mathbf{u} + \mathbf{v}) \bullet \mathbf{w} = \mathbf{u} \bullet \mathbf{w} + \mathbf{v} \bullet \mathbf{w}$

iii. $(\alpha \mathbf{u}) \bullet \mathbf{v} = \alpha(\mathbf{u} \bullet \mathbf{v})$

iv. $\mathbf{u} \bullet \mathbf{v} = \mathbf{v} \bullet \mathbf{u}$ (commutative)

v. $\mathbf{u} \bullet \mathbf{v} = 0 \Leftrightarrow \mathbf{u} \perp \mathbf{v}$

Dot product: what does it mean for CG?

- **Dot product** allow us to combine two vectors and reduce them into one real number
- The number represents how “aligned” those vectors are
- This is very important when doing things like shading
  - We need to determine whether the light ray is aligned with the normal surface for a polygon
  - This determines how much the light will contribute to the appearance of the polygon

Norm of a vector (i.e., its length)

$$\|\mathbf{u}\| = \sqrt{\mathbf{u} \bullet \mathbf{u}} = \sqrt{\sum_{i=0}^{n-1} u_i^2}$$ (norm)

Properties:

i. $\|\mathbf{u}\| = 0 \Leftrightarrow \mathbf{u} = (0,0,\ldots,0) = \mathbf{0}$

ii. $\|\alpha\mathbf{u}\| = |\alpha|\,\|\mathbf{u}\|$

iii. $\|\mathbf{u} + \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$ (triangle inequality)

iv. $\|\mathbf{u} \bullet \mathbf{v}\| \leq \|\mathbf{u}\|\,\|\mathbf{v}\|$ (Cauchy-Schawrtz inequality)

Normalizing

- Length of a vector is denoted as $\|\mathbf{v}\|$
- A vector can be normalized, to change its length to 1, without affecting its direction: $\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|}$

Norm: why is it important for CG?

- If you want a vector that "points" to a particular direction, rather than specifying a specific location in the space, a normalized vector is a good representation.
- Many CG operations assume we're working with a unit length vector
  - For example, computing light contribution is done with normalized vectors
  - We often normalize before putting vectors into the GPU
-

Linear independence

- A set of vectors $\mathbf{v_0}$, $\mathbf{v_1}, \ldots,$ $\mathbf{v_{n-1}}$ is *linearly independent* if:

  $$\alpha_0 \mathbf{v}_0 + \alpha_1 \mathbf{v}_1 + \ldots + \alpha_{n-1} \mathbf{v}_{n-1} = \mathbf{0} \text{ if and only if } \alpha_0 = \alpha_1 = \ldots = \alpha_{n-1} = 0$$

- If a set of vectors is linearly independent, we cannot represent one in terms of the other.
- If a set of vectors is linearly dependent, at least one can be written in terms of the other.

Linear independence

- $\mathbf{v_0} = (4,3)$ and $\mathbf{v_1} = (8,6)$ Linearly independent?
- $\mathbf{v_0} = (4,3)$ and $\mathbf{v_1} = (2,6)$ Linearly independent?

Linear independence

- $\mathbf{v_0} = (4,3)$ and $\mathbf{v_1} = (8,6)$ are not linearly independent, since $\alpha_0 = 2$ and $\alpha_1 = -1$ satisfy.
- $\mathbf{v_0} = (4,3)$ and $\mathbf{v_1} = (2,6)$ are linearly independent, since the only scalars to satisfy previous equation are $\alpha_0 = 0$ and $\alpha_1 = 0$.

Span

Span of S of a set of vectors **v** is all finite linear combinations of vectors of S:

$$\mathbf{span(S)} = \sum_{i=0}^{n-1} u_i \mathbf{v}_i$$

Spanning set of $\mathbb{R}^3$:

- $(1,0,0), (0,1,0), (0,0,-1)$
- $(1,0,0), (0,1,0), (0,0,1)$
- $(1,0,0), (0,1,0), (0,0,0)$

Span

Span of S of a set of vectors **v** is all finite linear combinations of vectors of S:

$$\mathbf{span(S)} = \sum_{i=0}^{n-1} u_i \mathbf{v}_i$$

Spanning set of $\mathbb{R}^3$:

- $(1,0,0), (0,1,0), (0,0,-1)$: Yes
- $(1,0,0), (0,1,0), (0,0,1)$: Yes
- $(1,0,0), (0,1,0), (0,0,0)$: No

Basis vectors

- Basis vectors:
  1. Linear independence (as shown before).
  2. Spanning property (as shown before).

Basis vectors

- In addition, if $u_0, \ldots, u_{n-1}$ are uniquely determined by $\mathbf{u}$ for all $\mathbf{u} \in \mathbb{R}^n$, then $\mathbf{v}_0, \ldots, \mathbf{v}_{n-1}$ is called a basis of $\mathbb{R}^n$.
- What this means: every vector can be described **uniquely** by *n* scalars $(u_0, \ldots, u_{n-1})$, and the basis vectors $\mathbf{v}_0, \ldots, \mathbf{v}_{n-1}$.

- $(1,0), (0,1)$
  - Linearly independent?
  - Spans $\mathbb{R}^2$?
  - Basis for $\mathbb{R}^2$
- $(1,0), (1,1), (0,2)$
  - Linearly independent?
  - Spans $\mathbb{R}^2$?
  - Basis for $\mathbb{R}^2$?
- $(1,1,0), (0,1,1)$
  - Linearly independent?
  - Spans $\mathbb{R}^3$?
  - Basis for $\mathbb{R}^3$?
- $(1,1), (1,-1)$
  - Linearly independent?
  - Spans $\mathbb{R}^2$?
  - Basis for $\mathbb{R}^2$?

Basis vectors

- In addition, if $u_0, \ldots, u_{n-1}$ are uniquely determined by $\mathbf{u}$ for all $\mathbf{u} \in \mathbb{R}^n$, then $\mathbf{v}_0, \ldots, \mathbf{v}_{n-1}$ is called a basis of $\mathbb{R}^n$.
- What this means: every vector can be described **uniquely** by *n* scalars $(u_0, \ldots, u_{n-1})$, and the basis vectors $\mathbf{v}_0, \ldots, \mathbf{v}_{n-1}$.

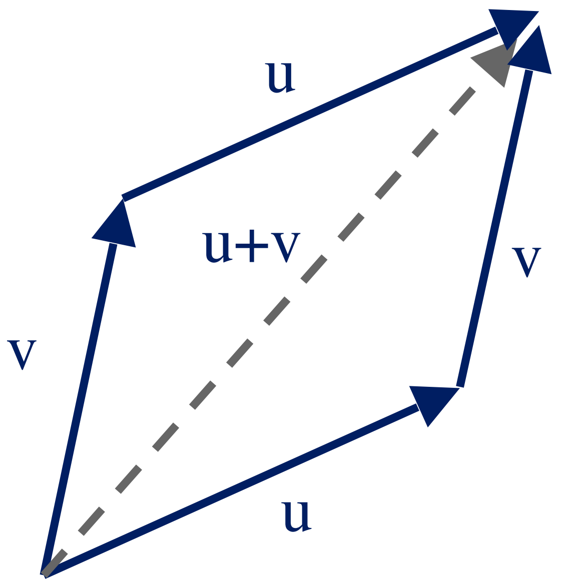
- $(1,0), (0,1)$
  - Linearly independent? Yes.
  - Spans $\mathbb{R}^2$? Yes.
  - Basis for $\mathbb{R}^2$? Yes.
- $(1,0), (1,1), \ (0,2)$
  - Linearly independent? No.
  - Spans $\mathbb{R}^2$? Yes.
  - Basis for $\mathbb{R}^2$? No.
- $(1,1,0), (0,1,1)$
  - Linearly independent? Yes.
  - Spans $\mathbb{R}^3$? No.
  - Basis for $\mathbb{R}^3$? No.
- $(1,1), (1,-1)$
  - Linearly independent? Yes.
  - Spans $\mathbb{R}^2$? Yes.
  - Basis for $\mathbb{R}^2$? Yes.

Coordinate systems

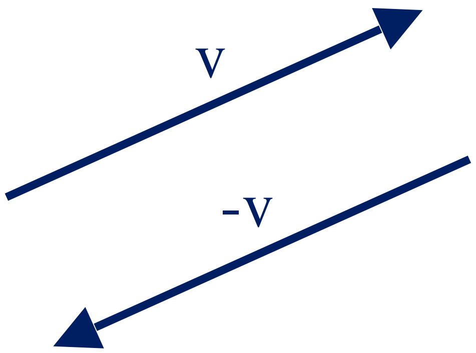
- Consider a basis $\mathbf{v}_0, \ldots, \mathbf{v}_{n-1}$
- A vector can be written as $\mathbf{v} = \alpha_0\mathbf{v}_0 + \alpha_1\mathbf{v}_1 + \ldots + \alpha_{n-1}\mathbf{v}_{n-1}$
- List of scalars is **sufficient** to represent $\mathbf{v}$ with respect to given basis.
- Therefore, we can write **v** as:

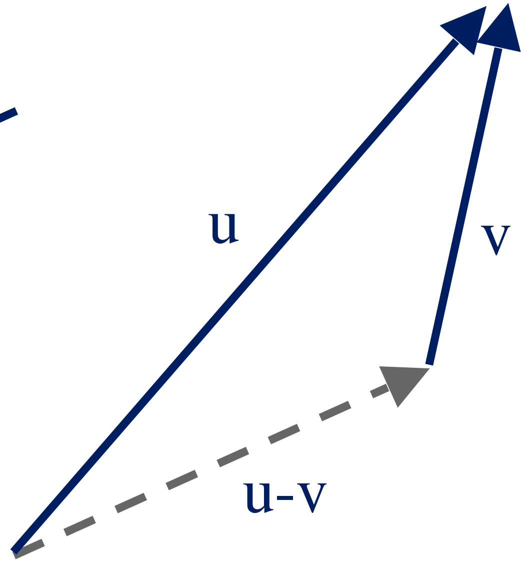
$$\mathbf{v} = \begin{pmatrix} \alpha_0 & \alpha_1 & \ldots & \alpha_{n-1} \end{pmatrix}^T = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \ldots \\ \alpha_{n-1} \end{pmatrix}$$

Operations on vectors



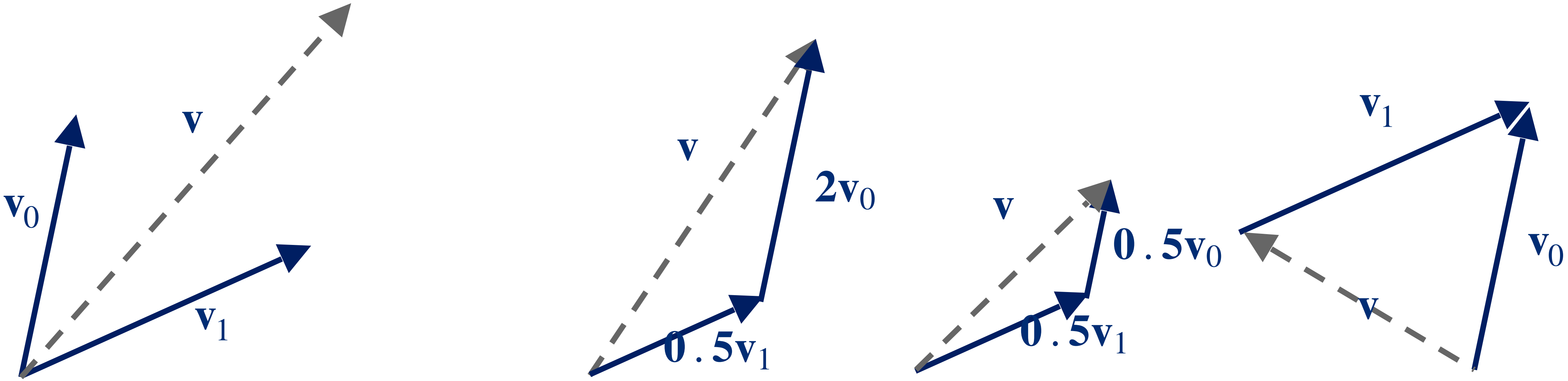
$\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ (commutativity)





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

Coordinates



$\mathbf{v} = \alpha_0\mathbf{v}_0 + \alpha_1\mathbf{v}_1$

$\mathbf{v} = 2\mathbf{v}_0 + 0.5\mathbf{v}_1$

$\mathbf{v} = 0.5\mathbf{v}_0 + 0.5\mathbf{v}_1$

$\mathbf{v} = \mathbf{v}_0 - \mathbf{v}_1$

$\mathbf{v}_0$ and $\mathbf{v}_1$ form a 2D basis

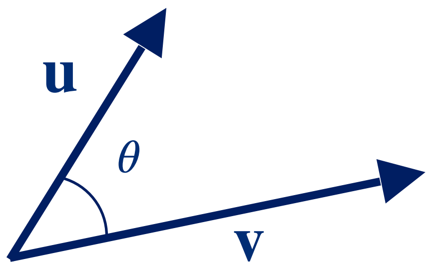
Length

- Length of a vector is denoted as $\|\mathbf{v}\|$
- A vector can be normalized, to change its length to 1, without affecting its direction: $\mathbf{u} = \dfrac{\mathbf{v}}{\|\mathbf{v}\|}$

Dot product

- Recall how dot product tells us about the **alignment** of two vectors
- The dot product is related to the length of two vectors and the angle between them.

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

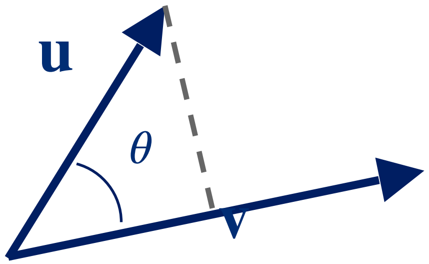


- If both are normalized, it is directly the cosine of the angle between them.

Projection

- The length of the projection of $\mathbf{u}$ onto $\mathbf{v}$ can be computed using the dot product

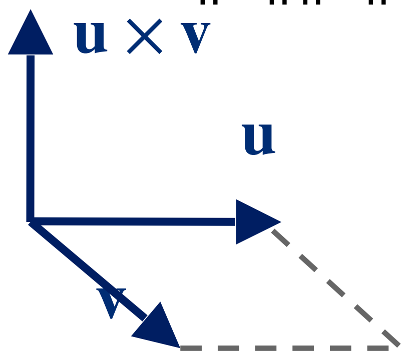
$$\mathbf{u} \rightarrow \mathbf{v} = \|\mathbf{u}\| cos\theta = \frac{\mathbf{u} \bullet \mathbf{v}}{\|\mathbf{v}\|}$$



Cross product

- Defined **only for 3D** vectors.
- The resulting vector is perpendicular to both $\mathbf{u}$ and $\mathbf{v}$, the direction depends on the right hand rule.
- The magnitude is equal to the area of the parallelogram formed by $\mathbf{u}$ and $\mathbf{v}$.

$$\mathbf{u} \times \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| sin\theta$$



Properties

i. $\mathbf{u} \times \mathbf{v} = -\mathbf{v} \times \mathbf{u}$ (anti-commutativity)

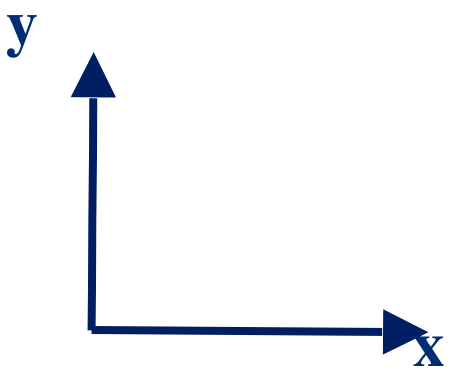
ii. $(a\mathbf{u} + b\mathbf{v}) \times \mathbf{w} = a(\mathbf{u} \times \mathbf{w}) + b(\mathbf{v} \times \mathbf{w})$ (linearity)

iii. $\mathbf{0} + \mathbf{v} = \mathbf{v}$ (zero identity)

iv. $\mathbf{u} \times (\mathbf{v} \times \mathbf{w}) = (\mathbf{u} \bullet \mathbf{w})\mathbf{v} - (\mathbf{u} \bullet \mathbf{v})\mathbf{w}$ (vector triple product)

Cartesian coordinates

- $\mathbf{x} = (1,0)$ and $\mathbf{y} = (0,1)$ form a canonical, Cartesian basis.



Coordinate systems

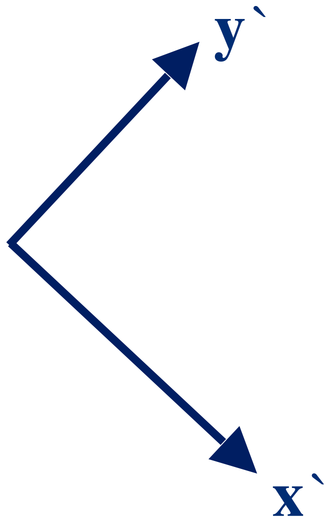
- We will always use orthonormal bases, which are formed by pairwise orthogonal unit vectors:

$$\mathbf{u} \bullet \mathbf{v} = \begin{cases} 0, i \neq j \\ 1, i = j \end{cases}$$

- This means that every basis vector must have a length of one, and also that each pair of basis vectors must be orthogonal.

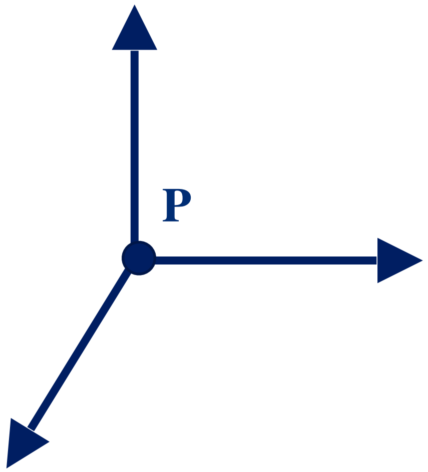
Cartesian coordinates

- Note that we can still other coordinate systems. Example
- $\mathbf{x} = (0.7, -0.7)$ and $\mathbf{y} = (0.7, 0.7)$ form a canonical, Cartesian basis.



Coordinate frames

- Note that a coordinate system is insufficient to represent points.
- We can add an origin to the basis vectors to form a frame.



Coordinate frames

- Frame determined by $(\mathbf{P}_0, \mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2)$.
- Within this frame, every vector can be written as:

$$\mathbf{v} = \alpha_0\mathbf{v}_0 + \alpha_1\mathbf{v}_1 + \ldots + \alpha_{n-1}\mathbf{v}_{n-1}$$

- Within this frame, every point can be written as:

$$\mathbf{P} = \mathbf{P}_0 + \beta_0\mathbf{v}_0 + \beta_1\mathbf{v}_1 + \ldots + \beta_{n-1}\mathbf{v}_{n-1}$$

Points and vectors

- Consider the point and the vector:

$$\mathbf{v} = \alpha_0 \mathbf{v}_0 + \alpha_1 \mathbf{v}_1 + \ldots + \alpha_{n-1} \mathbf{v}_{n-1}$$
$$\mathbf{P} = \mathbf{P}_0 + \beta_0 \mathbf{v}_0 + \beta_1 \mathbf{v}_1 + \ldots + \beta_{n-1} \mathbf{v}_{n-1}$$

- Often time, we assume that *P0*=(0,0,0,…)

- Similar representations:

$$\mathbf{v} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{n-1} \end{pmatrix} \qquad \mathbf{P} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n-1} \end{pmatrix}$$

- But: a vector has no position.

Change of basis

- Let $B = \{\mathbf{u}, \mathbf{w}\}$ and $B' = \{\mathbf{u}', \mathbf{w}'\}$ be two bases for $\mathbb{R}^2$. We would like to be able to express a vector $\mathbf{v}$ in B in terms of its coordinates in B'.
- Vector $\mathbf{v}$ has coordinates (x', y') in B'. This means that:

$$\mathbf{v} = x'\mathbf{u}' + y'\mathbf{w}'$$

Change of basis

- Let $B = \{\mathbf{u}, \mathbf{w}\}$ and $B' = \{\mathbf{u}', \mathbf{w}'\}$ be two bases for $\mathbb{R}^2$. We would like to be able to express a vector $\mathbf{v}$ in B in terms of its coordinates in B'.
- Vector $\mathbf{v}$ has coordinates (x', y') in B. This means that:

$$\mathbf{v} = x'\mathbf{u}' + y'\mathbf{w}'$$

- What about in B?
- Suppose the basis $\mathbf{u}'$ and $\mathbf{w}'$ have the following coordinates relative to the basis B:

$$\mathbf{u}' = a\mathbf{u} + b\mathbf{w}$$

$$\mathbf{w}' = c\mathbf{u} + d\mathbf{w}$$

Change of basis

$$\mathbf{v} = x'\mathbf{u}' + y'\mathbf{w}'$$

$$\mathbf{v} = x'(a\mathbf{u} + b\mathbf{w}) + y'(c\mathbf{u} + d\mathbf{w})$$

$$\mathbf{v} = (ax' + cy')\mathbf{u} + (bx' + dy')\,\mathbf{w}$$

$$= \begin{pmatrix} ax' + cy' \\ bx' + dy' \end{pmatrix}$$

$$= \begin{pmatrix} ac \\ bd \end{pmatrix} \mathbf{v}'$$

Switching to a matrix notation

Recall a, b, c, d are the expression of the B' {u', w'} bases in B

$$\mathbf{u}' = a\mathbf{u} + b\mathbf{w}$$

$$\mathbf{w}' = c\mathbf{u} + d\mathbf{w}$$

Change of basis

$$\mathbf{v} = x'\mathbf{u}' + y'\mathbf{w}'$$

$$\mathbf{v} = x'(a\mathbf{u} + b\mathbf{w}) + y'(c\mathbf{u} + d\mathbf{w})$$

$$\mathbf{v} = (ax' + cy')\mathbf{u} + (bx' + dy')\mathbf{w}$$

$$= \begin{pmatrix} ax' + cy' \\ bx' + dy' \end{pmatrix}$$

$$= \begin{pmatrix} a c \\ b d \end{pmatrix} \mathbf{v}'$$

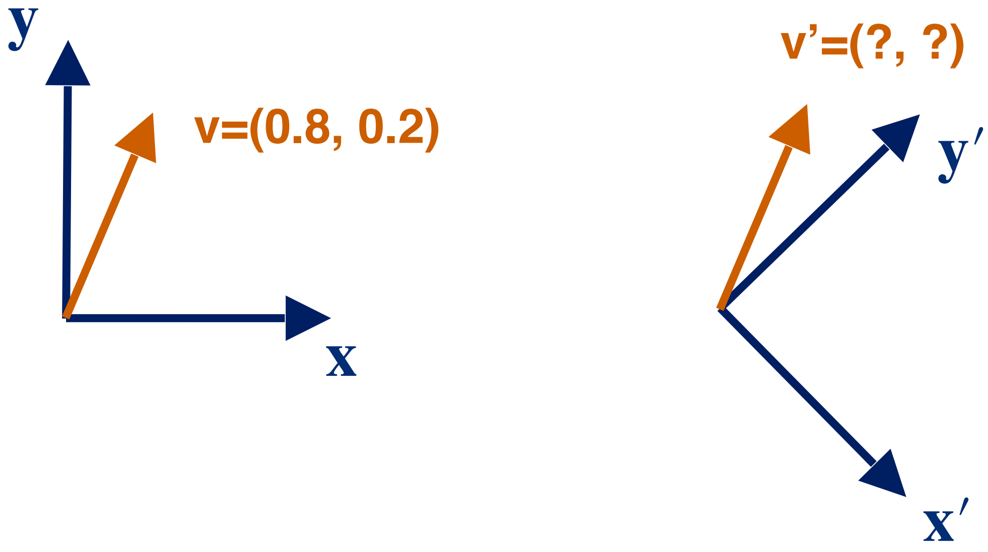
Switching to a matrix notation

Recall a, b, c, d are the expression of the B' {u', w'} bases in B

$$\mathbf{u}' = a\mathbf{u} + b\mathbf{w}$$

$$\mathbf{w}' = c\mathbf{u} + d\mathbf{w}$$

Change of basis (example)

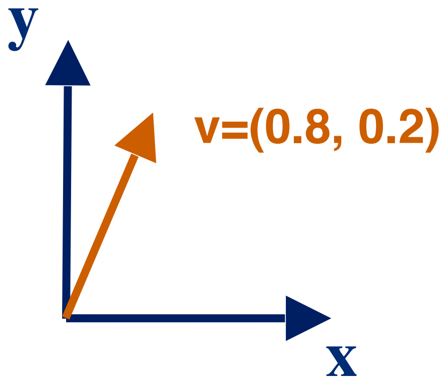


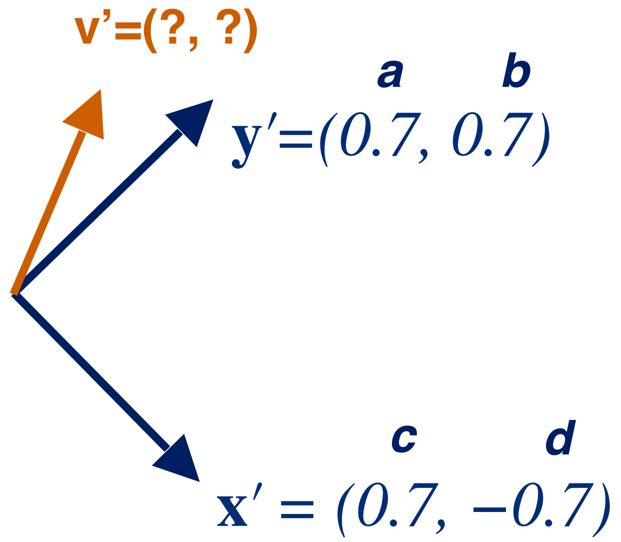
$$\mathbf{v}' = \begin{pmatrix} ac \\ bd \end{pmatrix} \mathbf{v}$$

$$x' = ax + by$$

$$y' = cx + y\mathbf{w}$$

Change of basis (example)





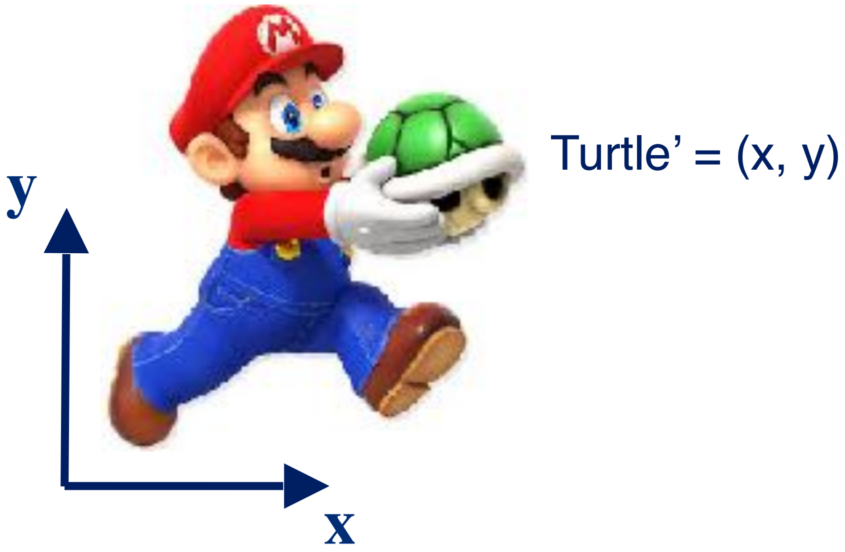
$$\mathbf{v}' = \begin{pmatrix} a\,c \\ b\,d \end{pmatrix} \mathbf{v}$$

$$x' = ax + by$$

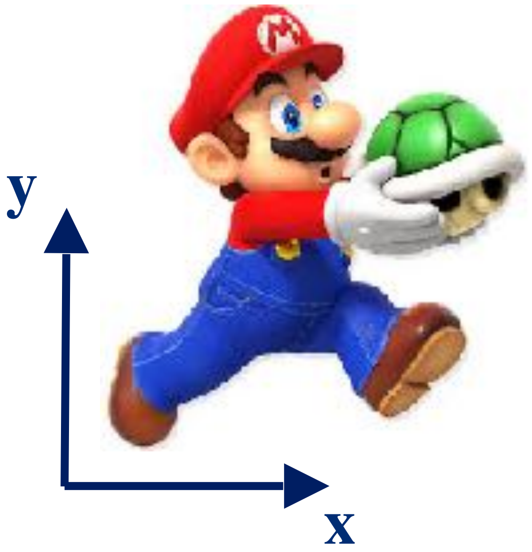
$$y' = cx + y\mathbf{w}$$

$$\mathbf{v}' = \begin{pmatrix} 0.7 * 0.7 \\ 0.7 * -0.7 \end{pmatrix} \mathbf{v}$$

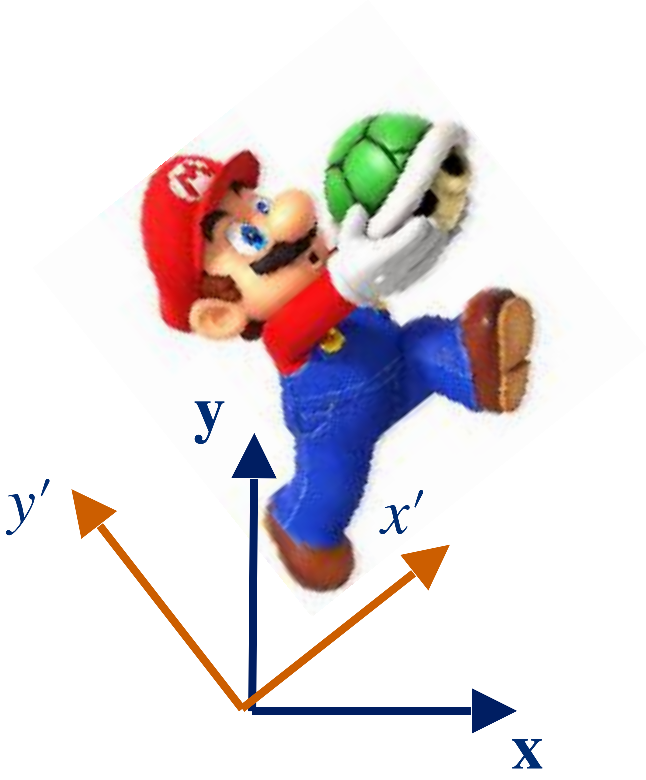
Why is this important for CG?



Why is this important for CG?

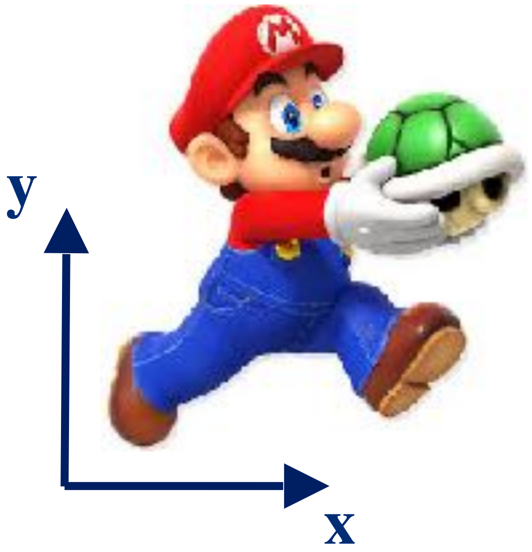


Turtle' = (2, 1)

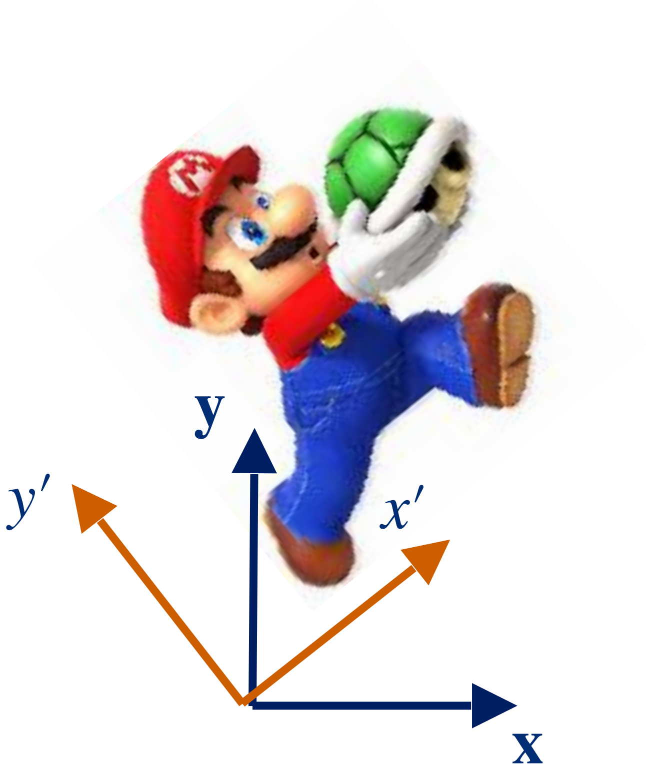


Turtle' = (?, ?)

Why is this important for CG?



Turtle' = (2, 1)



Turtle' = (?, ?)

$$T = \begin{pmatrix} a\,c \\ b\,d \end{pmatrix} T'$$

References

- Interactive Computer Graphics 7th Ed. by Ed Angel and Dave Shreiner (Chapter 3)
- Real-time Rendering, 3rd Ed. by Tomas Akenine-Möller, Eric Haines, and Naty Hoffman (Appendix A)