Chapter II
Math: Bare Basics

Matrix and Vector

- This chapter provides an intuitive presentation of the basics of math, which are needed throughout this book.

- $m \times n$ matrix

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

- Matrix-matrix multiplication: If A's dimension is $l \times m$ and B's dimension is $m \times n$, AB is an $l \times n$ matrix.

$$\begin{aligned} AB &= \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{pmatrix} \\ &= \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} & a_{11}b_{13} + a_{12}b_{23} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} & a_{21}b_{13} + a_{22}b_{23} \\ a_{31}b_{11} + a_{32}b_{21} & a_{31}b_{12} + a_{32}b_{22} & a_{31}b_{13} + a_{32}b_{23} \end{pmatrix} \end{aligned}$$

Matrix and Vector (cont'd)

- The typical representation of a 2D vector, (x,y), or a 3D vector, (x,y,z), is called a *row vector*. Instead, we can use a *column vector*: $\begin{pmatrix} x \\ y \end{pmatrix}$ $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$
- Matrix-vector multiplication

$$Mv = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \\ a_{31}x + a_{32}y \end{pmatrix}$$

- *Transpose* denoted by M^T $\begin{pmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{pmatrix}$

- A different representation of matrix-vector multiplication

$$v^T M^T = \begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{pmatrix} = \begin{pmatrix} xa_{11} + ya_{12} & xa_{21} + ya_{22} & xa_{31} + ya_{32} \end{pmatrix}$$

- OpenGL uses the column vectors and the vector-on-the-right representation for matrix-vector multiplication, but Direct3D uses the row vectors and the vector-on-the-left representation.

Matrix and Vector (cont'd)

- Identity matrix denoted by I

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- For any matrix M, $MI = IM = M$.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

- When two 'square' matrices A and B are multiplied to make an identity matrix, i.e., $AB = I$, B is called the *inverse* of A and is denoted by A^{-1}. Equally, A is the inverse of B.
- Theorems
  - $(AB)^{-1} = B^{-1}A^{-1}$
  - $(AB)^T = B^TA^T$

Matrix and Vector (cont'd)

- The coordinates of a vector *a* in the *n*-dimensional space

$$(a_1, a_2, \ldots, a_n)$$

- Its length denoted by $||a||$

$$\sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}$$

- Normalization

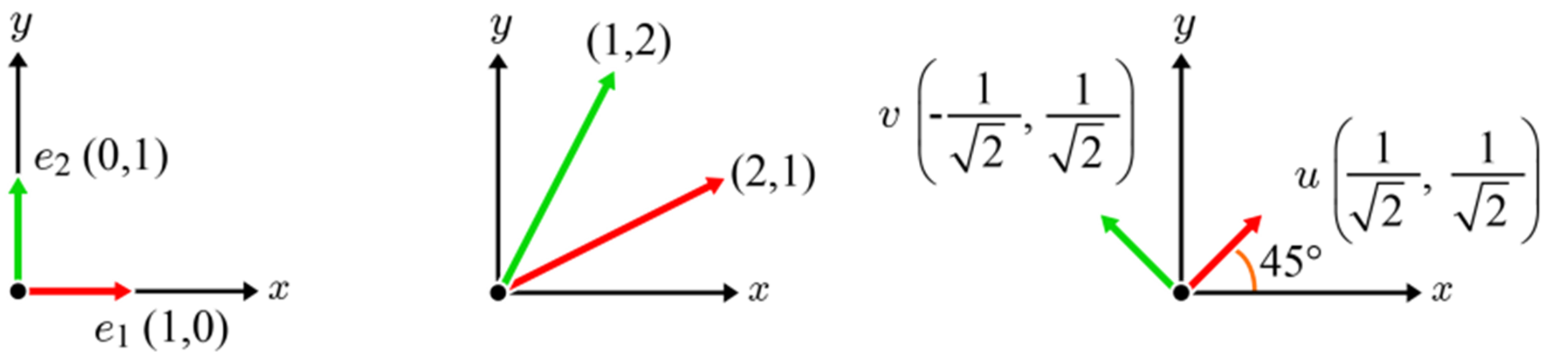
$$\frac{a}{||a||}$$

- Such a normalized vector is called a *unit vector* in that its length is 1.

Basis

- Coordinate system = origin + basis

- The vectors $v_1, v_2, .. , v_n$ form a *basis* for the vector space V iff (1) $v_1, v_2, .. , v_n$ are linearly independent, and (2) $v_1, v_2, .. , v_n$ span V.
- Basis examples

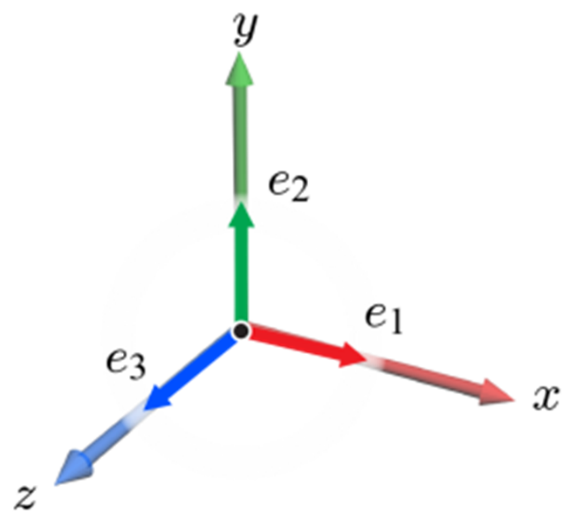


orthonormal standard | non-orthonormal non-standard | orthonormal non-standard

- An *orthonormal basis* is an orthogonal set of unit vectors.

Basis (cont'd)

- 3D standard basis, $\{e_1, e_2, e_3\}$, where e_1=(1,0,0), e_2=(0,1,0), and e_3=(0,0,1).

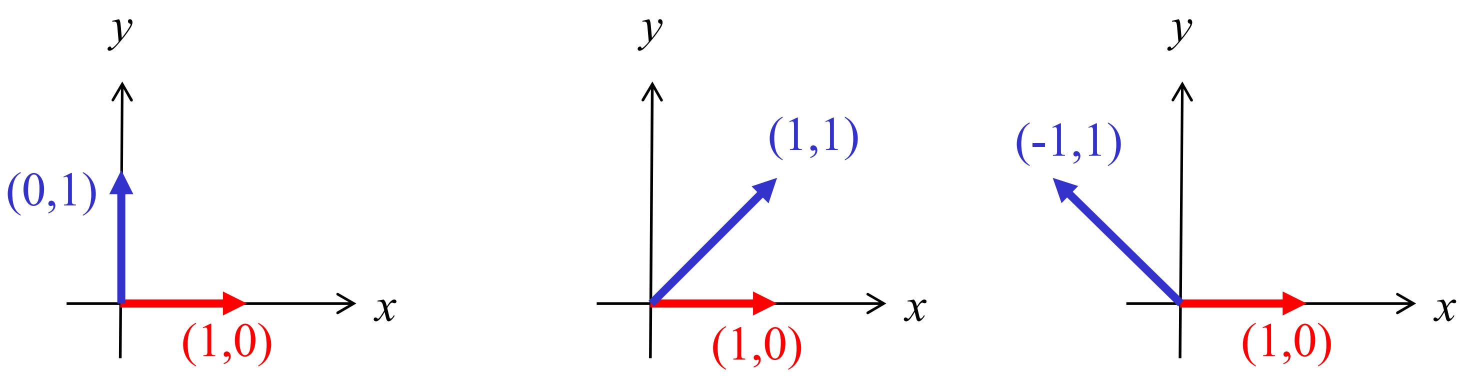


- It is also *orthonormal*.
- Of course, we can imagine non-standard orthonormal bases.

- From now on, we will handle only orthonormal bases.

Dot Product

- Given vectors, a and b, whose coordinates are $(a_1, a_2, .. , a_n)$ and $(b_1, b_2, .. , b_n)$, respectively, the dot product $a \cdot b$ is defined to be $a_1b_1+a_2b_2+ .. +a_nb_n$.
- In Euclidean geometry, $a \cdot b = \|a\|\|b\|\cos\theta$, where $\|a\|$ and $\|b\|$ denote the lengths of a and b, respectively, and θ is the angle between a and b.
  - If a and b are perpendicular to each other, $a \cdot b = 0$.
  - If θ is an acute angle, $a \cdot b > 0$.
  - If θ is an obtuse angle, $a \cdot b < 0$.

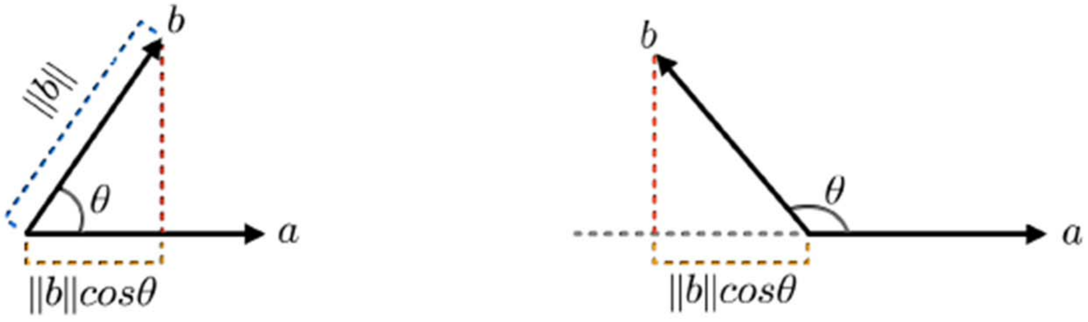


$(1,0) \cdot (0,1) = 0$ $(1,0) \cdot (1,1) = 1$ $(1,0) \cdot (-1,1) = -1$

- If a is a unit vector, $a \cdot a = 1$.

Dot Product (cont'd)

- Suppose that a is a unit vector, i.e., $\|a\| = 1$. Then, $a \cdot b = \|b\|\cos\theta$. It is the length of b projected onto a.

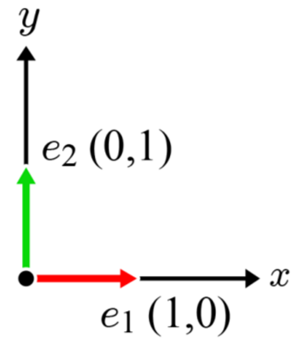


(a) $\|b\|cos\theta > 0$ (b) $\|b\|cos\theta < 0$

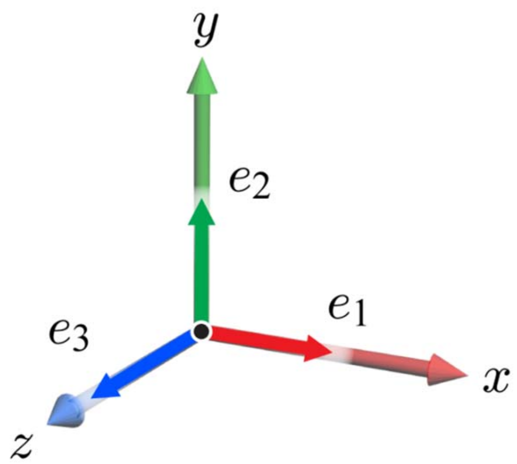
- The projected length is negative if θ is an obtuse angle. In this sense, $\|b\|\cos\theta$ is called the *signed length*.

Dot Product (cont'd)

- The 2D standard basis $\{e_1, e_2\}$ has the following feature: $e_1 \cdot e_1 = 1$, $e_2 \cdot e_2 = 1$, and $e_1 \cdot e_2 = 0$.

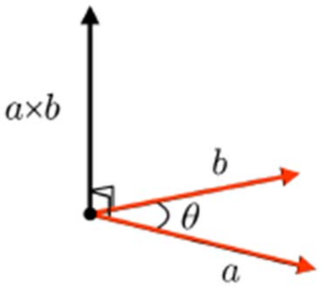


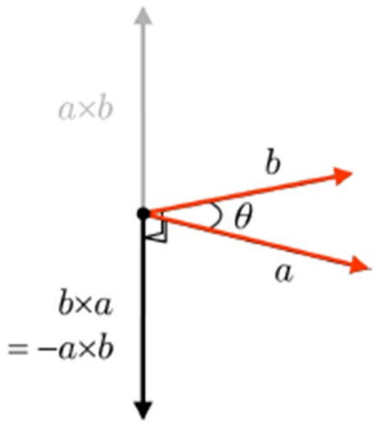
- Similar feature can be found in the 3D standard basis.
  - $e_1 \cdot e_1 = 1$, $e_2 \cdot e_2 = 1$, and $e_3 \cdot e_3 = 1$
  - $e_1 \cdot e_2 = 0$, $e_1 \cdot e_3 = 0$, and $e_2 \cdot e_3 = 0$



Cross Product

- The cross product takes as input two vectors, a and b, in 3D space and outputs another 3D vector which is perpendicular to a and b. It's denoted by $a \times b$ and is perpendicular to the plane spanned by a and b.
- The direction of $a \times b$ is defined by the right-hand rule. The length equals the area of a parallelogram that a and b span: $\|a\|\|b\|\sin\theta$.



- The right-hand rule implies that the direction of $b \times a$ is opposite to that of $a \times b$, i.e., $b \times a = -a \times b$, but their lengths are the same. In this sense, the cross product operation is called *anti-commutative*.

Cross Product (cont'd)

- If $a=(a_x, a_y, a_z)$ and $b=(b_x, b_y, b_z)$, $a \times b=(a_y b_z - a_z b_y,\ a_z b_x - a_x b_z,\ a_x b_y - a_y b_x)$.

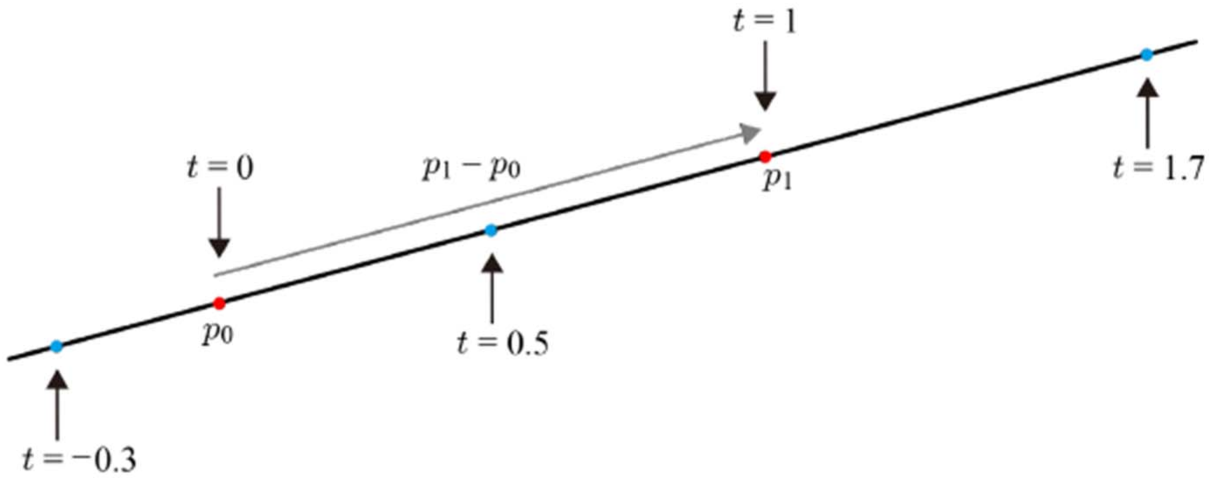
$$a=(a_x, a_y, a_z)$$
$$b=(b_x, b_y, b_z)$$

$$a=(a_x, a_y, a_z)\ a_x$$
$$b=(b_x, b_y, b_z)\ b_x$$

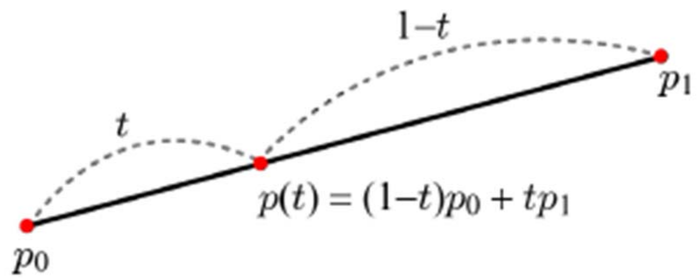
$$a=(a_x, a_y, a_z)$$
$$b=(b_x, b_y, b_z)$$

Line, Ray, and Linear Interpolation

- A line defined by two end points, p_0 and p_1: $p(t) = p_0 + t(p_1 - p_0)$



- When t is restricted to [0,1], $p(t)$ represents a line segment, which corresponds to linear interpolation of p_0 and p_1.



Line, Ray, and Linear Interpolation (cont'd)

- Linear interpolation in 3D space

$$p(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} (1-t)x_0 + tx_1 \\ (1-t)y_0 + ty_1 \\ (1-t)z_0 + tz_1 \end{pmatrix}$$

- Whatever attributes are associated with the end points, they can be linearly interpolated. Suppose that the endpoints are associated with colors c_0 and c_1, respectively, where $c_0 = (R_0, G_0, B_0)$ and $c_1 = (R_1, G_1, B_1)$. Then, the color $c(t)$ is defined as follows:

$$c(t) = (1-t)c_0 + tc_1 = \begin{pmatrix} (1-t)R_0 + tR_1 \\ (1-t)G_0 + tG_1 \\ (1-t)B_0 + tB_1 \end{pmatrix}$$