FILMUNIVERSITÄT
BABELSBERG
KONRAD WOLF

Mathematics for Audio & Graphics

Prof. Lena Gieseke

WS17 - Lecture 09

08.01.19

Today

- Summary Vectors
- Matrices
  - Operations
  - The Identity Matrix
  - The Inverse Matrix
  - Determinants

"Points, vectors, matrices and normals are to computer graphics what the alphabet is to literature…"

[1]

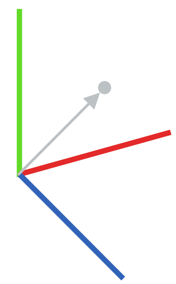
Summary Vectors

- Points vs. directions
- Operations
- Magnitude
- Normalization
- The Dot Product
- Projections
- The Cross Product
- Scalar Triple Product

Points vs. Directions

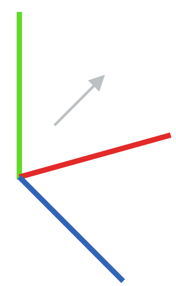
Summary Vectors

- Points in space
  - Locations of objects
  - Vertices of a triangle mesh
  - Usually there is some understood origin location from which all other locations are stored as offsets



$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

- Spatial directions
  - Orientation of the camera
  - An offset or a displacement
  - Velocity

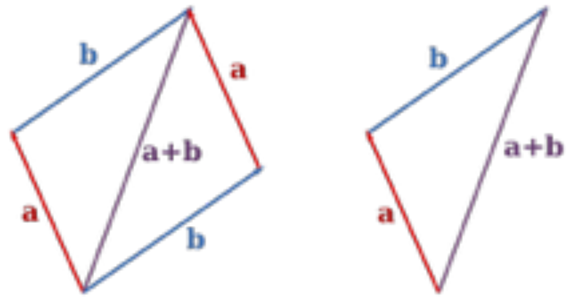


$$\mathbf{a} = \begin{bmatrix} x_a \\ y_a \end{bmatrix}$$

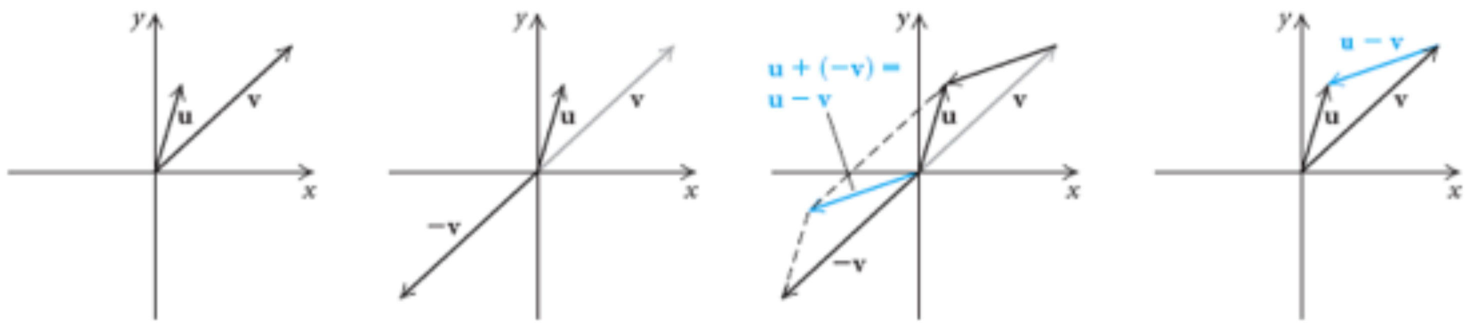
Operations

Summary Vectors

- Addition



- Subtraction



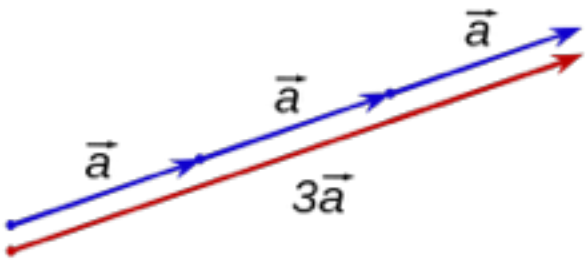
https://www.math10.com/en/geometry/vectors-operations/vectors-operations.html

https://en.wikipedia.org/wiki/Euclidean_vector#Addition_and_subtraction

Operations

Summary Vectors

- Scalar Multiplication



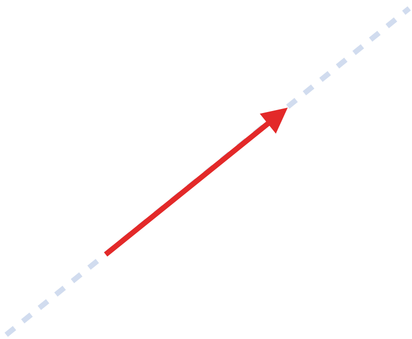
https://en.wikipedia.org/wiki/Euclidean_vector#Addition_and_subtraction

The Magnitude

Summary Vectors

- The **length** of a vector is also called the *magnitude*

$$\|\mathbf{a}\| = \sqrt{\sum_{i=1}^{n} a_i^2}$$



E.g. in 2D cartesian coordinates:

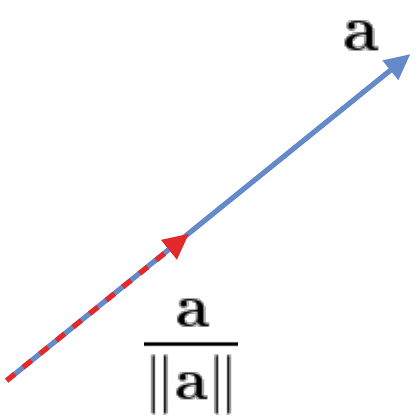
$$\|\mathbf{a}\| = \sqrt{x_a^2 + y_a^2}$$

Normalization

Summary Vectors

- *Normalization* is the resizing of a vector to length 1 (unit size)
- Done through multiplication by

$$\frac{1}{\|\mathbf{a}\|}$$

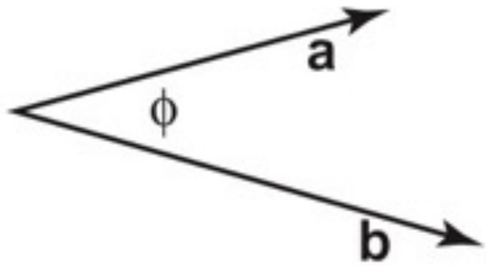


- This is essential for many graphics operations

Dot Product

Summary Vectors

$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \cos \phi$$



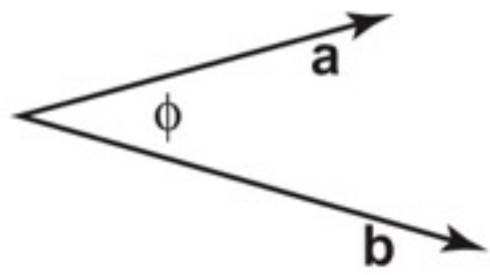
$$\mathbf{a} \cdot \mathbf{b} = x_a x_b + y_a y_b + z_a z_b$$

Dot Product

Summary Vectors

$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \cos \phi$$

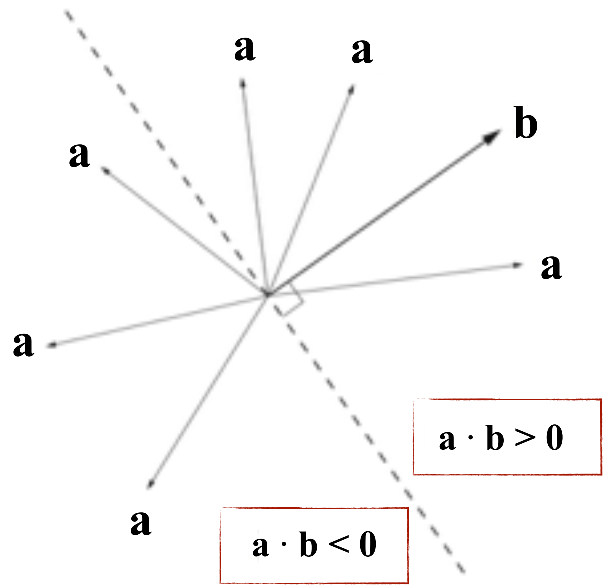
$$\cos \phi = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \, \|\mathbf{b}\|}$$



$$\mathbf{a} \cdot \mathbf{b} = x_a x_b + y_a y_b + z_a z_b$$

Dot Product

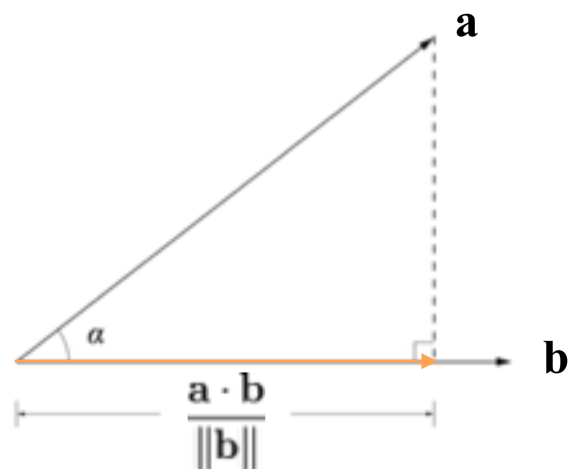
Summary Vectors



Projections

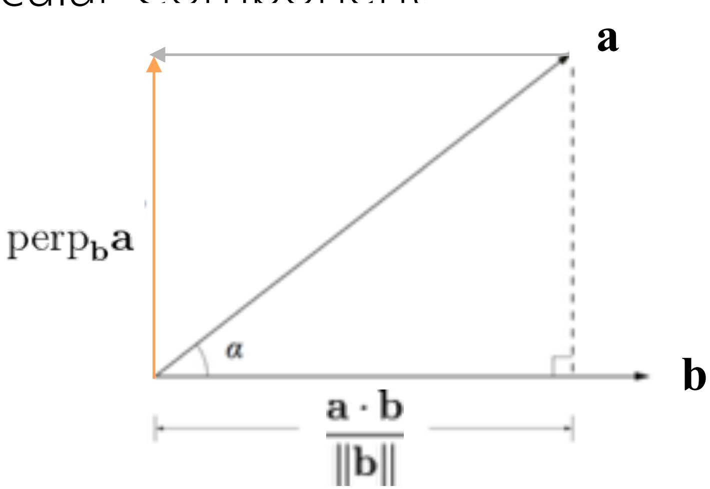
Summary Vectors

- Parallel Comp



$$\mathrm{proj}_{\mathbf{b}}\mathbf{a} = \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{b}\|^2}\mathbf{b}$$

- Perpendicular Component



$$\mathrm{perp}_{\mathbf{b}}\mathbf{a} = \mathbf{a} - \mathrm{proj}_{\mathbf{b}}\mathbf{a}$$

$$= \mathbf{a} - \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{b}\|^2}\mathbf{b}$$

Cross Product

$$\mathbf{a} \times \mathbf{b} = (y_a z_b - z_a y_b, z_a x_b - x_a z_b, x_a y_b - y_a x_b).$$

- Given two 3D vectors **a** and **b**, the cross product **a** × **b** satisfies the equation

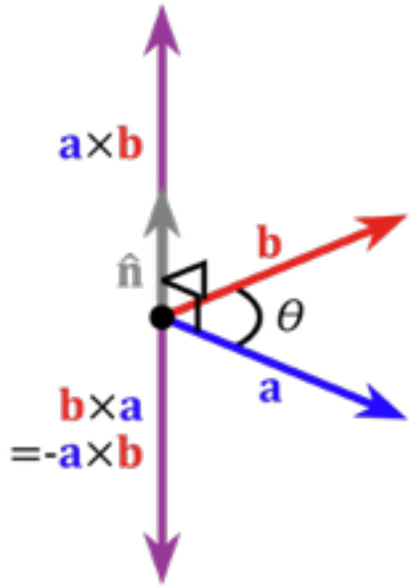
$$||\mathbf{a} \times \mathbf{b}|| = ||\mathbf{a}|| \, ||\mathbf{b}|| \sin \alpha$$

where α is the planar angle between **a** and **b.**

Cross Product

Summary Vectors

- Returns a new *vector* that is **perpendicular to both of the vectors** being multiplied together
  - One of its major uses in Computer Graphics is the calculation of a surface normal at a particular point given two distinct tangent vectors.



https://en.wikipedia.org/wiki/Euclidean_vector#Addition_and_subtraction

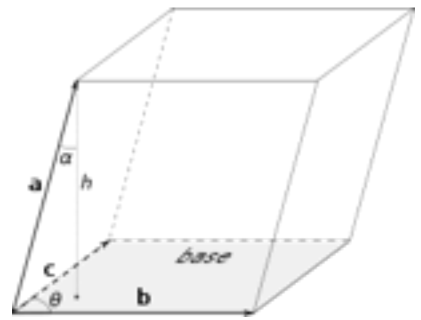
Scalar Triple Product

Scalar Triple Product

Vectors

- Defined as the *dot product* of one of the vectors with the *cross product* of the other two:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$$



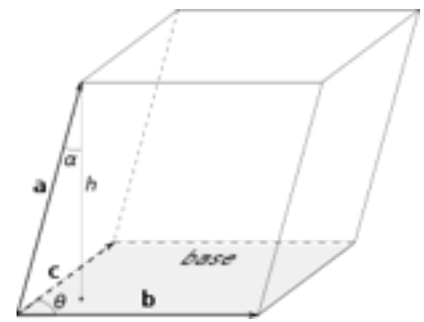
- The (signed) volume of the parallelepiped (3D parallelogram; a sheared 3D box) defined by the three vectors given

Scalar Triple Product

Vectors

- Defined as the *dot product* of one of the vectors with the *cross product* of the other two:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$$



https://en.wikipedia.org/wiki/Triple_product

- The (signed) volume of the parallelepiped (3D parallelogram; a sheared 3D box) defined by the three vectors given
  - Parentheses may be omitted without causing ambiguity, since the dot product cannot be evaluated first. If it were, it would leave the cross product of a scalar and a vector, which is not defined.

Properties

Vectors | Scalar Triple Product

- The scalar triple product is invariant under a circular shift of its three operands (**a**, **b**, **c**):

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{c} \times \mathbf{a}) = \mathbf{c} \cdot (\mathbf{a} \times \mathbf{b})$$

- Following from above and the commutative property of the dot product:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = (\mathbf{a} \times \mathbf{b}) \cdot \mathbf{c}$$

Matrices

Matrices

- Variety of purposes
  - E.g. representation of spatial transforms
  - Moving from one coordinate space to another

[3]

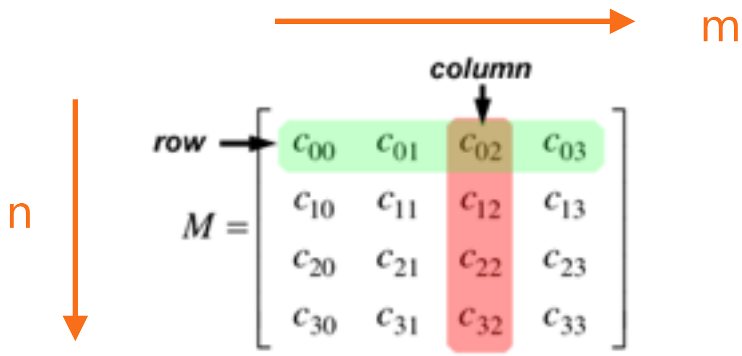
Properties

Matrices

$$
M = \begin{array}{cc}
 & \begin{array}{cccc} \phantom{c_{00}} & \phantom{c_{01}} & \text{column} & \phantom{c_{03}} \end{array} \\
\text{row} & \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} \\ c_{10} & c_{11} & c_{12} & c_{13} \\ c_{20} & c_{21} & c_{22} & c_{23} \\ c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix}
\end{array}
$$

Properties

Matrices

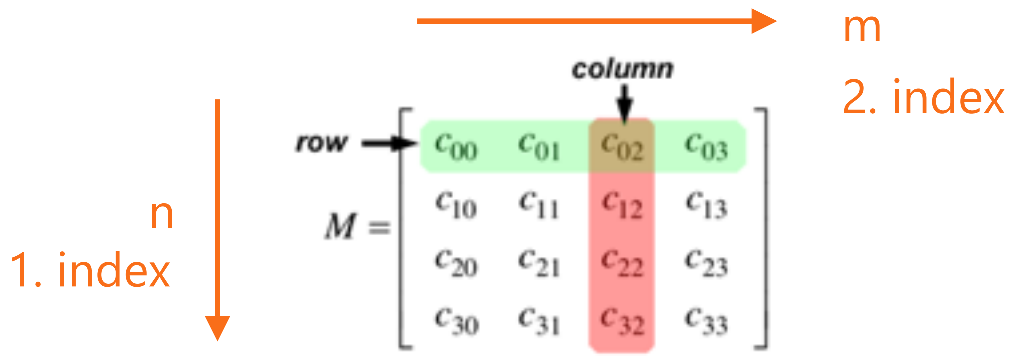


An $n \times m$ matrix $\mathbf{M}$ is an array of numbers having n rows and m columns.

If $n = m$, then the matrix $\mathbf{M}$ is *square*.

Properties

Matrices



An $n \times m$ matrix **M** is an array of numbers having n rows and m columns.

If $n = m$, then the matrix **M** is *square*.

M_{ij} (or as above c_{ij}) refers to the entry of **M** that resides at the i-th row of the j-th column.

Properties

Matrices

As an example, suppose that **F** is a 3×4 matrix

$$\mathbf{F} = \begin{bmatrix} F_{11} & F_{12} & F_{13} & F_{14} \\ F_{21} & F_{22} & F_{23} & F_{24} \\ F_{31} & F_{32} & F_{33} & F_{34} \end{bmatrix}$$

The entries for which $i = j$ are called the *main diagonal* entries of the matrix.

Properties

Matrices

A square matrix whose only nonzero entries appear on the main diagonal is called a *diagonal* matrix.

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & -3 & 0 \end{bmatrix}$$

Properties

Matrices

The *transpose* of an $n \times m$ matrix $\mathbf{M}$, denoted by $\mathbf{M}^T$, is an $m \times n$ matrix for which the *(i, j)* entry is equal to M_{ji}.

The transpose of the matrix **F:**

$$\mathbf{F} = \begin{bmatrix} F_{11} & F_{12} & F_{13} & F_{14} \\ F_{21} & F_{22} & F_{23} & F_{24} \\ F_{31} & F_{32} & F_{33} & F_{34} \end{bmatrix}$$

$$\mathbf{F}^T = \begin{bmatrix} F_{11} & F_{21} & F_{31} \\ F_{12} & F_{22} & F_{32} \\ F_{13} & F_{23} & F_{33} \\ F_{14} & F_{24} & F_{34} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}^T = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

Properties

Matrices

The *transpose* of an $n \times m$ matrix $\mathbf{M}$, denoted by $\mathbf{M}^T$, is an $m \times n$ matrix for which the (i, j) entry is equal to M_{ji}.

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ . \\ . \\ x_n \end{bmatrix} \quad \mathbf{x}^{\mathrm{T}} = [x_1 \ x_2 \ x_3 \ldots x_n] \quad (\mathbf{x}^{\mathrm{T}})^{\mathrm{T}} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ . \\ . \\ x_n \end{bmatrix}$$

(You will often times see transposed Version of a Vector in Text, as it is easier to write.)

Scalar Multiplication

As with vectors (which can be thought of as $n \times 1$ matrices), scalar multiplication is defined for matrices.

Given a scalar a and an $n \times m$ matrix $\mathbf{M}$, the product $a\mathbf{M}$ is given by

$$a\mathbf{M} = \mathbf{M}a = \begin{bmatrix} aM_{11} & aM_{12} & \cdots & aM_{1m} \\ aM_{21} & aM_{22} & \cdots & aM_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ aM_{n1} & aM_{n2} & \cdots & aM_{nm} \end{bmatrix}$$

$$4 \begin{pmatrix} -1 & 0 & 2 \\ 3 & 2 & 1 \\ -2 & 3 & 0 \end{pmatrix} = \begin{pmatrix} -4 & 0 & 8 \\ 12 & 8 & 4 \\ -8 & 12 & 0 \end{pmatrix}$$

Addition

Matrices

Given two $n \times m$ matrices **F** and **G**, the sum **F**+**G** is given by

$$\mathbf{F}+\mathbf{G} = \begin{bmatrix} F_{11}+G_{11} & F_{12}+G_{12} & \cdots & F_{1m}+G_{1m} \\ F_{21}+G_{21} & F_{22}+G_{22} & \cdots & F_{2m}+G_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ F_{n1}+G_{n1} & F_{n2}+G_{n2} & \cdots & F_{nm}+G_{nm} \end{bmatrix}$$

$$\mathbf{A} = \begin{pmatrix} -1 & 2 & 4 \\ 4 & -5 & 3 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 3 & 0 & -1 \\ 1 & 2 & -4 \end{pmatrix}$$

$$\begin{aligned} \mathbf{C} &= \mathbf{A} + \mathbf{B} \\ &= \begin{pmatrix} -1+3 & 2+0 & 4+(-1) \\ 4+1 & -5+2 & 3+(-4) \end{pmatrix} \\ &= \begin{pmatrix} 2 & 2 & 3 \\ 5 & -3 & -1 \end{pmatrix} \end{aligned}$$

Properties

Matrices

Given any two scalars a and b and any three $n \times m$ matrices **F**, **G**, and **H**, the following properties hold.

Theorem 6

(a) $\mathbf{F} + \mathbf{G} = \mathbf{G} + \mathbf{F}$
(b) $(\mathbf{F} + \mathbf{G}) + \mathbf{H} = \mathbf{F} + (\mathbf{G} + \mathbf{H})$
(c) $a(b\mathbf{F}) = (ab)\mathbf{F}$
(d) $a(\mathbf{F} + \mathbf{G}) = a\mathbf{F} + a\mathbf{G}$
(e) $(a + b)\mathbf{F} = a\mathbf{F} + b\mathbf{F}$

Multiplication

Matrices

If **A** is an $n \times m$ matrix and **B** is an $m \times p$ matrix

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1p} \\ B_{21} & B_{22} & \cdots & B_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mp} \end{pmatrix}$$

the matrix product **AB** (denoted *without* multiplication signs or dots) is defined to be the $n \times p$ matrix

$$\mathbf{AB} = \begin{pmatrix} (\mathbf{AB})_{11} & (\mathbf{AB})_{12} & \cdots & (\mathbf{AB})_{1p} \\ (\mathbf{AB})_{21} & (\mathbf{AB})_{22} & \cdots & (\mathbf{AB})_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{AB})_{n1} & (\mathbf{AB})_{n2} & \cdots & (\mathbf{AB})_{np} \end{pmatrix}$$

where each i, j entry is given by multiplying the entries $\mathbf{A}_{ik}$ (across row i of **A**) by the entries $\mathbf{B}_{kj}$ (down column j of **B**), for $k = 1, 2, \ldots, m$, and summing the results over k:

$$(\mathbf{AB})_{ij} = \sum_{k=1}^{m} A_{ik} B_{kj} \, .$$

Multiplication

Matrices

For matrix multiplication, rows of the first matrix are multiplied with columns of the second matrix:

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

$$c_{11} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \end{pmatrix} \begin{pmatrix} b_{11} \\ b_{21} \\ b_{31} \end{pmatrix} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31}$$

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

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

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

http://www.mathematrix.de/matrixmultiplikation/

Multiplication

Matrices

If F is an $n \times m$ matrix and **G** is an $m \times p$ matrix, then the product **FG** is an $n \times p$ matrix whose *(i, j)* entry is given by

$$(\mathbf{FG})_{ij} = \sum_{k=1}^{m} F_{ik} G_{kj}$$

Another way of looking at this is that the *(i, j)* entry of **FG** is equal to the dot product of the *i*-th row of **F** and the *j*-th column of **G**.

Multiplication

Matrices

- Taking a product of two matrices is only possible if the **number of columns of the left** matrix is the **same** as the **number of rows of the right** matrix.

- Hence, Matrix multiplication is not commutative in most instances:

$$\mathbf{FG} \neq \mathbf{GF}$$

Multiplication

Matrices

Thus the product **AB** is defined only if the number of columns in **A** is equal to the number of rows in **B**, in this case *m*.

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1p} \\ B_{21} & B_{22} & \cdots & B_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mp} \end{pmatrix}$$

n x *m* *m* x *p* = *n* x *p*

Properties

Matrices | Multiplication

Given any scalar a, an $n \times m$ matrix **F**, an $m \times p$ matrix **G**, and a $p \times q$ matrix **H**, the following properties hold.

Theorem 7

(a) $(a\mathbf{F})\mathbf{G} = a(\mathbf{FG})$

(b) $(\mathbf{FG})\mathbf{H} = \mathbf{F}(\mathbf{GH})$

(c) $(\mathbf{FG})^{\mathrm{T}} = \mathbf{G}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}}$

Identity Matrix

Matrices

There is an $n \times n$ matrix called the *identity* matrix, denoted by $\mathbf{I}_n$, for which

$$\mathbf{M}\mathbf{I}_n = \mathbf{I}_n\mathbf{M} = \mathbf{M}$$

for any $n \times n$ matrix $\mathbf{M}$. The identity matrix has the form

$$\mathbf{I}_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

Inverse Matrix

An $n \times n$ matrix $\mathbf{M}$ is *invertible* if there exists a matrix, which we denote by $\mathbf{M}^{-1}$, such that

$$\mathbf{M}\mathbf{M}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}.$$

The matrix $\mathbf{M}^{-1}$ is called the *inverse* of $\mathbf{M}$.

Inverse Matrix

Matrices

An $n \times n$ matrix $\mathbf{M}$ is *invertible* if there exists a matrix, which we denote by $\mathbf{M}^{-1}$, such that

$$\mathbf{M}\mathbf{M}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}.$$

The matrix $\mathbf{M}^{-1}$ is called the *inverse* of $\mathbf{M}$.

Example: the inverse of a number $aa^{-1} = \frac{a}{a} = 1$

Inverse Matrix

An $n \times n$ matrix $\mathbf{M}$ is *invertible* if there exists a matrix, which we denote by $\mathbf{M}^{-1}$, such that

$$\mathbf{M}\mathbf{M}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}.$$

The matrix $\mathbf{M}^{-1}$ is called the *inverse* of $\mathbf{M}$.

$$A \cdot A^{-1} = \begin{pmatrix} 2 & 5 \\ 1 & 3 \end{pmatrix} \cdot \begin{pmatrix} 3 & -5 \\ -1 & 2 \end{pmatrix} = \begin{pmatrix} 6-5 & -10+10 \\ 3-3 & -5+6 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$$

Inverse Matrix

An $n \times n$ matrix $\mathbf{M}$ is *invertible* if there exists a matrix, which we denote by $\mathbf{M}^{-1}$, such that

$$\mathbf{M}\mathbf{M}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}.$$

The matrix $\mathbf{M}^{-1}$ is called the *inverse* of $\mathbf{M}$.

Not every matrix has an inverse, and those that do not are called *singular*.

Inverse Matrix

- There are numerous approaches to compute the inverse of a matrix.
  - Gauss-Jordan elimination
  - Newton's method
  - Cayley–Hamilton method
  - Eigendecomposition
  - Cholesky decomposition
  - Analytic solution
  - Blockwise inversion
  - By Neumann series
  - P-adic approximation

Why Do We Care?

- Because there is no matrix division!
  - But multiplying by an Inverse achieves the same thing

- *How do I share 10 apples with 2 people?*
  - Multiply 10 with the inverse (reciprocal) of 2
  - $10 \times 0.5 = 5$

Why Do We Care?

- Say we want to find Matrix **X**, and we know Matrix **A** and B:

$$\mathbf{XA} = \mathbf{B}$$

  - It would be nice to divide both sides by **A** (to get **X**=**B**/**A**), but remember we can't divide.
  - But what if we multiply both sides by $\mathbf{A}^{-1}$?

$$\mathbf{XA\,A}^{-1} = \mathbf{BA}^{-1}$$

  - And we know that $\mathbf{AA}^{-1} = \mathbf{I}$, so:

$$\mathbf{XI} = \mathbf{BA}^{-1}$$

  - We can remove **I** (for the same reason we can remove "1" from 1x = ab for numbers):

$$\mathbf{X = BA}^{-1}$$

  - And we have our answer (assuming we can calculate $\mathbf{A}^{-1}$)

Properties

Theorem 8

A matrix possessing a row or column consisting entirely of zeros is not invertible.

Theorem 9

A matrix $\mathbf{M}$ is invertible if and only if $\mathbf{M}^{\mathrm{T}}$ is invertible.

Theorem 10

If $\mathbf{F}$ and $\mathbf{G}$ are $n \times n$ invertible matrices, then the product $\mathbf{FG}$ is invertible, and $(\mathbf{FG})^{-1} = \mathbf{G}^{-1}\mathbf{F}^{-1}$.

Determinants

Determinants
Matrices

- The determinant of a *square* matrix is a **scalar** quantity derived from the entries of the matrix.

- The determinant of a matrix $\mathbf{M}$ is denoted det($\mathbf{M}$), det $\mathbf{M}$, or $|\mathbf{M}|$

- Determinants occur throughout mathematics and have many useful properties in linear algebra
  - The scaling factor of the transformation described by a matrix
  - Indicates if a linear system represented in matrix form is solvable

Determinant

Matrices

- When evaluating the determinant of a matrix, brackets are replaced with vertical bars:

$$\det \mathbf{M} = \begin{vmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{vmatrix}$$

Determinant

- Formula for the determinant of a 2×2 matrix:

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

Determinant

Matrices

- Formula for the determinant of a 2×2 matrix:

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

$$\begin{matrix} + & & - \end{matrix}$$

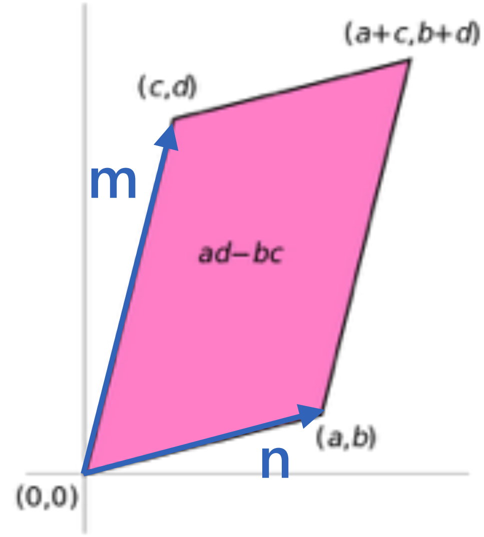
$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

$$B = \begin{bmatrix} 4 & 6 \\ 3 & 8 \end{bmatrix} \qquad |B| = 4\times8 - 6\times3 = 32-18 = 14$$

Geometric Interpretation

- For 2D vectors **n** and **m**, the determinant |**nm**| is the area of the parallelogram formed by **n** and **m**:

$$\begin{array}{c} \mathbf{n} \rightarrow \\ \mathbf{m} \rightarrow \end{array} \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$



- This is a signed area
  - If positive then **n** and **m** are right-handed, if negative they are left-handed.
  - |**nm**| = −|**mn**|

Determinant

Matrices

- Formula for the determinant of a 3×3 matrix:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}$$

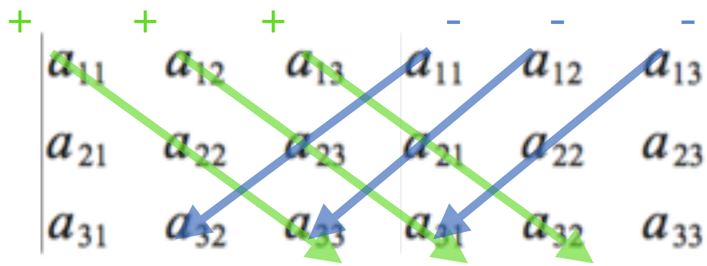
$$= a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31})$$

$$+ a_{13}(a_{21}a_{32} - a_{22}a_{31})$$

Determinant

Matrices

- To memorize the equation, write two copies of the matrix M side by side and multiply entries along the diagonals and back-diagonals, adding the diagonal terms and subtracting the back- diagonal terms:



$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} = a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31}) + a_{13}(a_{21}a_{32} - a_{22}a_{31})$$

$$= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{11}a_{23}a_{31} - a_{12}a_{21}a_{33} - a_{13}a_{22}a_{31}$$

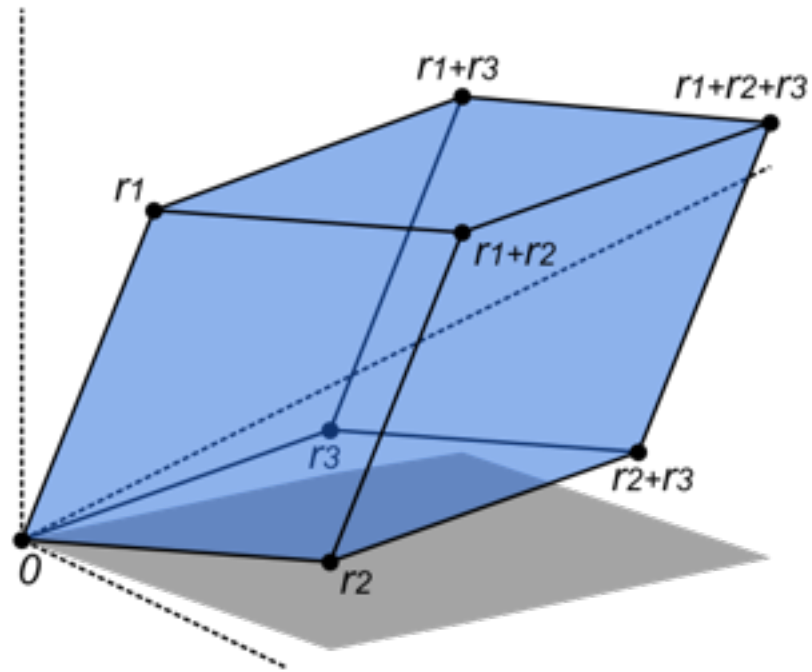
Geometric Interpretation

Matrices | Determinantes | 3 x 3

- For 3D vectors $\mathbf{r_1}$, $\mathbf{r_2}$ and $\mathbf{r_3}$ the determinant $|\mathbf{r_1}\,\mathbf{r_2}\,\mathbf{r_3}|$ is the signed volume of the parallelepiped formed by $\mathbf{r_1}$, $\mathbf{r_2}$ and $\mathbf{r_3}$:

$$\begin{array}{l} \mathbf{r_1} \rightarrow \\ \mathbf{r_2} \rightarrow \\ \mathbf{r_3} \rightarrow \end{array} \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}$$

$$= a_{11}(a_{22}a_{33} - a_{23}a_{32}) - a_{12}(a_{21}a_{33} - a_{23}a_{31}) + a_{13}(a_{21}a_{32} - a_{22}a_{31})$$



Geometric Interpretation

Matrices | Determinantes

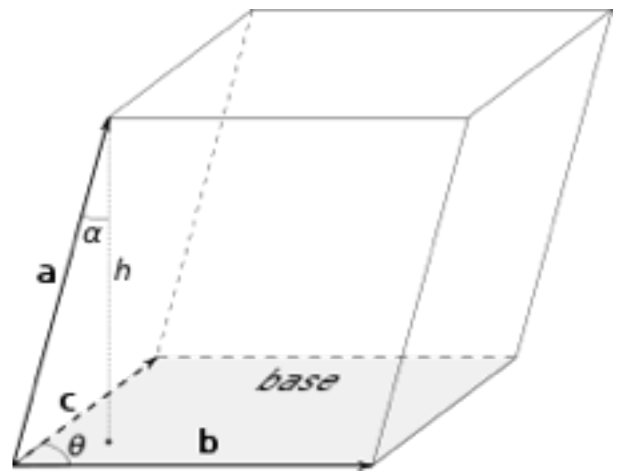
- To generalize:
  - The determinant takes n n-dimensional vectors and combines them to get a signed n-dimensional volume of the n-dimensional parallelepiped defined by the vectors.

Sounds Vaguely Familiar? 👬

Matrices | Determinantes

- Remember *scalar triple product*

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$$



- It can also be understood as the determinant of the 3×3 matrix (thus also its inverse) having the three vectors either as its rows or its columns.

Sounds Vaguely Familiar? 👬

- The *scalar triple product* is the same as the determinant of the 3×3 matrix:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \det \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} = \det(\mathbf{a}, \mathbf{b}, \mathbf{c}) \, .$$

Cross Product 👨‍👨‍👦

Matrices | Determinantes

- If we set **a** to unit vectors parallel to the **x**, **y**, and **z** axes (often written as **i**,**j**, and **k**

$$\boxed{\mathbf{a}} \cdot (\mathbf{b} \times \mathbf{c}) = \det \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} = \det(\boxed{\mathbf{a},} \mathbf{b}, \mathbf{c}) \,.$$

$$\boxed{\begin{aligned} \mathbf{i} &= \langle 1,0,0 \rangle \\ \mathbf{j} &= \langle 0,1,0 \rangle \\ \mathbf{k} &= \langle 0,0,1 \rangle \end{aligned}}$$

Cross Product 👨‍👨‍👦

- Then, we can compute the cross product by evaluating the *pseudo-determinant*

$$\mathbf{u} \times \mathbf{v} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{vmatrix}$$

$$= \begin{vmatrix} u_2 & u_3 \\ v_2 & v_3 \end{vmatrix} \mathbf{i} - \begin{vmatrix} u_1 & u_3 \\ v_1 & v_3 \end{vmatrix} \mathbf{j} + \begin{vmatrix} u_1 & u_2 \\ v_1 & v_2 \end{vmatrix} \mathbf{k}$$

Properties

Matrices | Determinantes

Theorem 11

For any two $n \times n$ matrices $\mathbf{F}$ and $\mathbf{G}$, $\det \mathbf{FG} = \det \mathbf{F} \det \mathbf{G}$.

Properties

Matrices | Determinantes

Theorem 12

An $n \times n$ matrix $\mathbf{M}$ is invertible if and only if $\det \mathbf{M} \neq 0$.

Inverses

- The inverse of a matrix can be computed by

$$\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{bmatrix}$$

$$\mathbf{B}^{-1} = \frac{1}{\det \mathbf{B}} \begin{bmatrix} B_{22}B_{33} - B_{23}B_{32} & B_{13}B_{32} - B_{12}B_{33} & B_{12}B_{23} - B_{13}B_{22} \\ B_{23}B_{31} - B_{21}B_{33} & B_{11}B_{33} - B_{13}B_{31} & B_{13}B_{21} - B_{11}B_{23} \\ B_{21}B_{32} - B_{22}B_{31} & B_{12}B_{31} - B_{11}B_{32} & B_{11}B_{22} - B_{12}B_{21} \end{bmatrix}$$

Adjoint Matrix

Matrices | Determinantes

$$\begin{bmatrix} B_{22}B_{33} - B_{23}B_{32} & B_{13}B_{32} - B_{12}B_{33} & B_{12}B_{23} - B_{13}B_{22} \\ B_{23}B_{31} - B_{21}B_{33} & B_{11}B_{33} - B_{13}B_{31} & B_{13}B_{21} - B_{11}B_{23} \\ B_{21}B_{32} - B_{22}B_{31} & B_{12}B_{31} - B_{11}B_{32} & B_{11}B_{22} - B_{12}B_{21} \end{bmatrix}$$

- This is the transpose of the matrix where elements of **B** are replaced by their respective cofactors multiplied by the leading constant (1 or -1). This matrix is called the *adjoint* of **B**.
- The adjoint is the transpose of the cofactor matrix of **B**.

Transformations

Linear Transformations

- A *linear transformation* uses a 2 × 2 matrix to change, or transform, a 2D vector:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \end{bmatrix}$$

- This simple formula achieves a variety of useful transformations, depending on the entries of the matrix.

[2]

Linear Transformations

- A *linear transformation* uses a 2 × 2 matrix to change, or transform, a 2D vector:

$$\underbrace{\boxed{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}}_{\text{Transform}} \underbrace{\boxed{\begin{bmatrix} x \\ y \end{bmatrix}}}_{\text{Vector}} = \underbrace{\boxed{\begin{bmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \end{bmatrix}}}_{\text{Transformed Vector}}$$

- This simple formula achieves a variety of useful transformations, depending on the entries of the matrix.

[2]

Scaling

Linear Transformations

- The most basic transform is a scale along the coordinate axes.
  - This transform can change length and possibly direction:

$$\text{scale}(s_x, s_y) = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

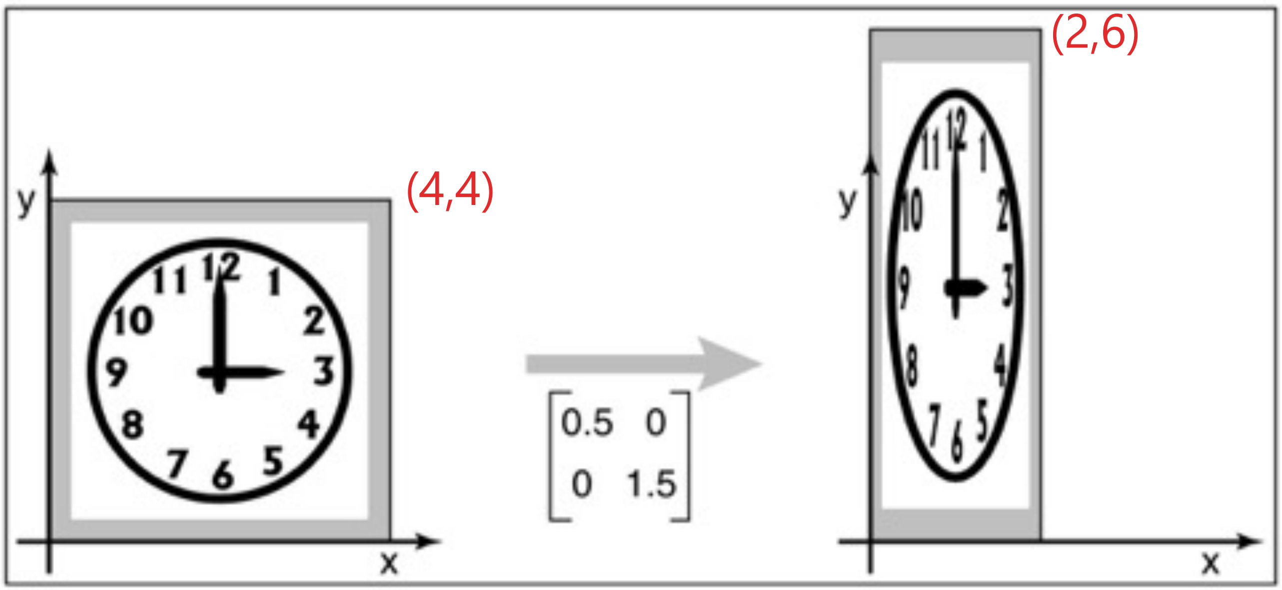
$$\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \end{bmatrix}$$

[2]

Scaling
Linear Transformations

$$\text{scale}(0.5, 1.5) = \begin{bmatrix} 0.5 & 0 \\ 0 & 1.5 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 0 \\ 0 & 1.5 \end{bmatrix} \begin{bmatrix} 4 \\ 4 \end{bmatrix} = \begin{bmatrix} 0.5 \times 4 + 0 \times 4 \\ 0 \times 4 + 1.5 \times 4 \end{bmatrix} = \begin{bmatrix} 2 \\ 6 \end{bmatrix}$$



[2]

Shearing

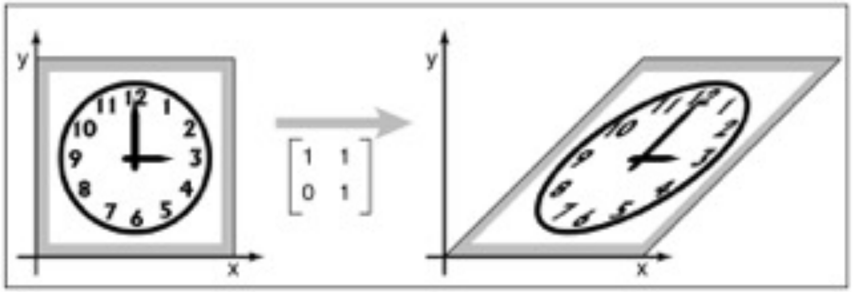
Linear Transformations

- The horizontal and vertical shear matrices are:

$$\text{shear-x}(s) = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}, \quad \text{shear-y}(s) = \begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix}$$

  - An x-shear matrix moves points to the right in proportion to their y-coordinate:

$$\text{shear-x}(1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

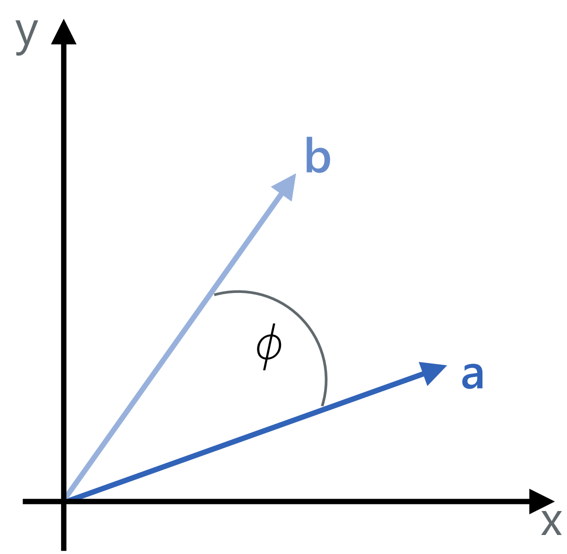


[2]

Rotation

Linear Transformations

- Suppose we want to rotate a vector **a** by an angle ϕ counterclockwise to get vector **b**.



[2]

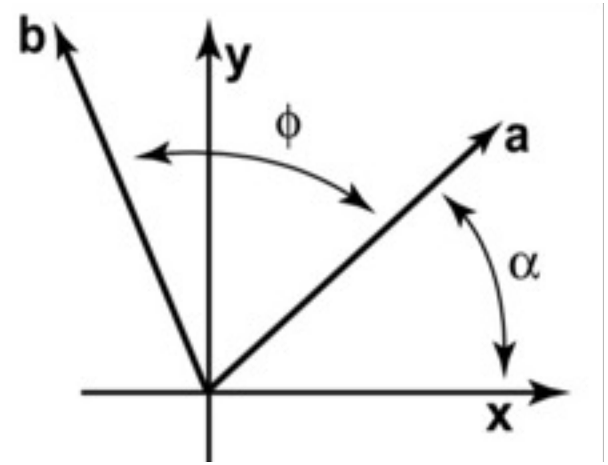
Rotation

Linear Transformations

- If **a** makes an angle α with the **x**-axis, and its length is $r = \sqrt{x_a^2 + y_a^2}$, then

$$x_a = r \cos \alpha$$

$$y_a = r \sin \alpha$$



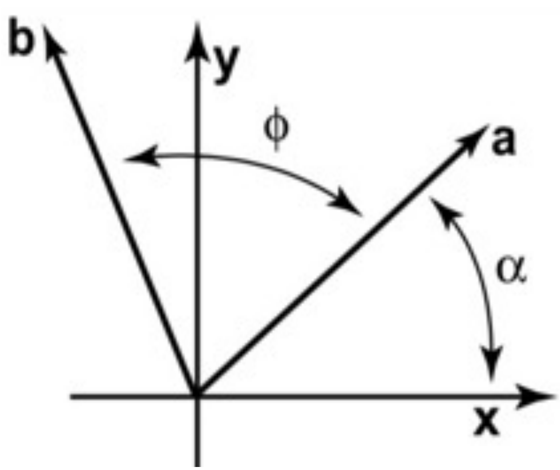
[2]

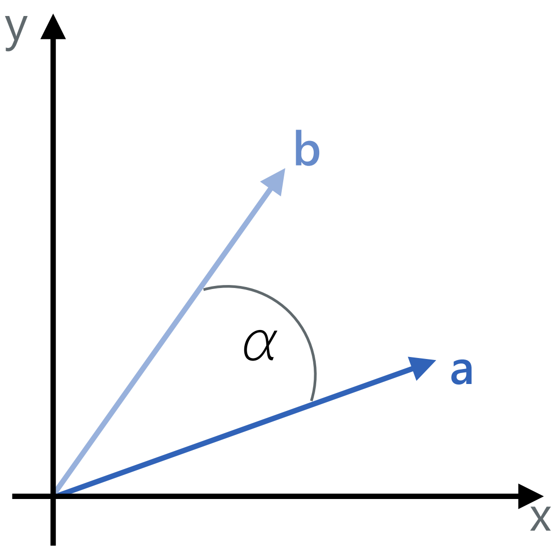
Rotation

Linear Transformations

- If **a** makes an angle α with the **x**-axis, and its length is $r = \sqrt{x_a^2 + y_a^2}$, then

$$x_a = r\cos\alpha$$
$$y_a = r\sin\alpha$$





[2]

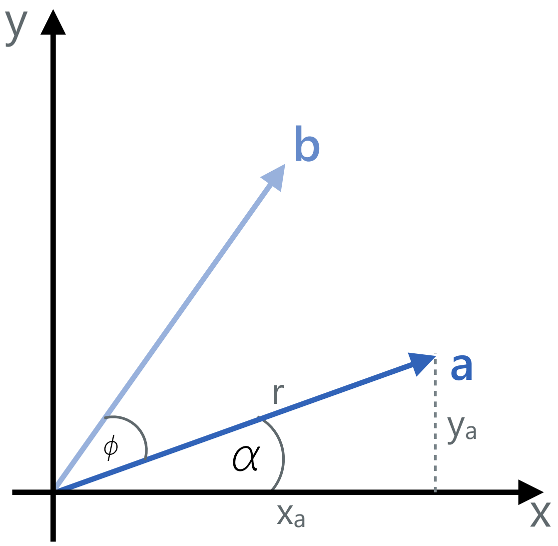
Rotation

Linear Transformations

- If **a** makes an angle α with the **x**-axis, and its length is $r = \sqrt{x_a^2 + y_a^2}$, then

$$x_a = r\cos\alpha$$

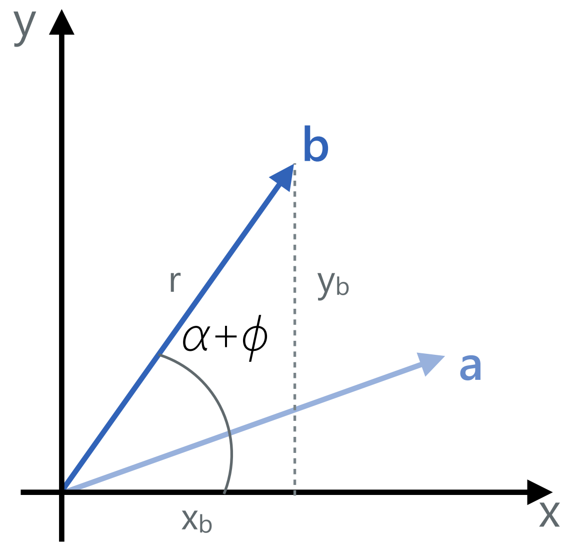
$$y_a = r\sin\alpha$$



[2]

- As **b** is a rotation of **a**, it also has length *r*
- It is rotated an angle ϕ from **a**, hence **b** makes an angle $(\alpha + \phi)$ with the x-axis

$$x_b = r\cos(\alpha + \phi)$$
$$y_b = r\sin(\alpha + \phi)$$

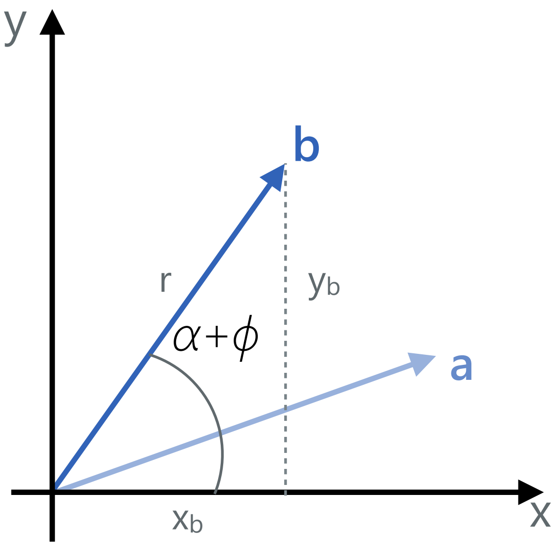


[2]

Rotation

Linear Transformations

- As **b** is a rotation of **a**, it also has length *r*
- It is rotated an angle ϕ from **a**, hence **b** makes an angle $(\alpha + \phi)$ with the x-axis



$$x_b = r\cos(\alpha+\phi)$$
$$y_b = r\sin(\alpha+\phi)$$

- Using the trigonometric addition identities:

$$\cos(\beta+\alpha) = \cos(\beta)\cdot\cos(\alpha) - \sin(\beta)\cdot\sin(\alpha)$$
$$\sin(\beta+\alpha) = \sin(\beta)\cdot\cos(\alpha) + \cos(\beta)\cdot\sin(\alpha)$$

$$x_b = r\cos(\alpha+\phi) = r\cos\alpha\cos\phi - r\sin\alpha\sin\phi,$$
$$y_b = r\sin(\alpha+\phi) = r\sin\alpha\cos\phi + r\cos\alpha\sin\phi.$$

[2]

Rotation

Linear Transformations

- Substituting $x_a = r\cos\alpha$ and $y_a = r\sin\alpha$ gives

$$x_b = x_a \cos\phi - y_a \sin\phi,$$
$$y_b = x_a \sin\phi + y_a \cos\phi.$$

- In matrix form, the transformation that takes **a** to **b** is then

$$\text{rotate}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}$$

[2]

Rotation

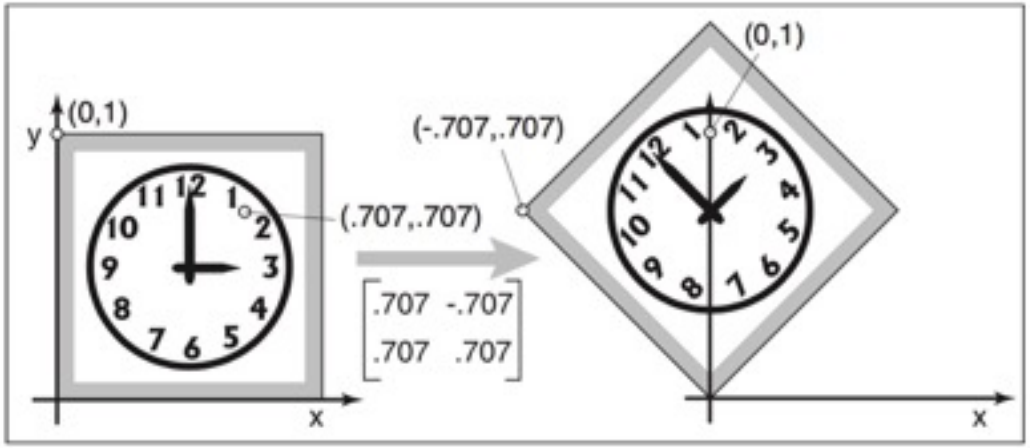
Linear Transformations

- A matrix that rotates vectors by π/4 radians (45 degrees) is

$$\begin{bmatrix} \cos\frac{\pi}{4} & -\sin\frac{\pi}{4} \\ \sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix} = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}$$

$$\begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} \begin{bmatrix} 0.707 \\ 0.707 \end{bmatrix} = \begin{bmatrix} 0.707 \times 0.707 - 0.707 \times 0.707 \\ 0.707 \times 0.707 + 0.707 \times 0.707 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \times 0.707 - 1 \times 0.707 \\ 0 \times 0.707 + 1 \times 0.707 \end{bmatrix} = \begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix}$$



[2]

Rotation

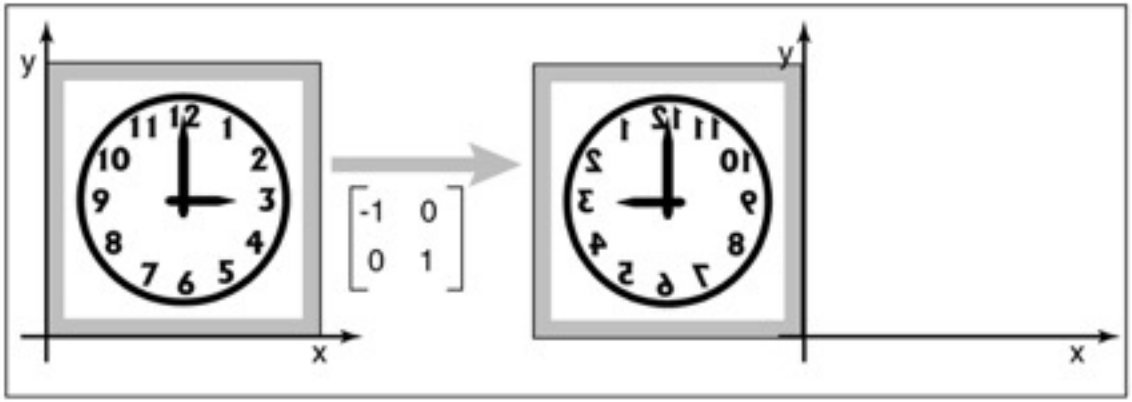
Linear Transformations

- These formulae define the rotation around the origin!
  - A positive angle rotates counter-clockwise, a negative one clockwise
- These formulae assume that the x axis points right and the y axis points up
  - Different coordinate systems work with the same principle, but the coordinate assignments need to be adjusted

[2]

- To reflect a vector across either of the coordinate axes, a scale with one negative scale factor is used:

$$\text{reflect-y} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{reflect-x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



[2]

Composition

- The effects of transforming a vector by two matrices in sequence (e.g. scale **S**, rotation **R**) can be done multiplying the two transformation matrices to a single matrix of the same size:

$$\mathbf{M} = \mathbf{R}\mathbf{S}$$

[2]

Composition

Linear Transformations

- The effects of transforming a vector by two matrices in sequence (e.g. scale **S**, rotation **R**) can be done multiplying the two transformation matrices to a single matrix of the same size:

$$\mathbf{M} = \mathbf{RS}$$

- It is very important to remember that **these transforms are applied from the right side first**.
  - The matrix **M** = **RS** first applies **S** and then **R**

[2]

Composition

Linear Transformations

- The effects of transforming a vector by two matrices in sequence (e.g. scale **S**, rotation **R**) can be done multiplying the two transformation matrices to a single matrix of the same size:

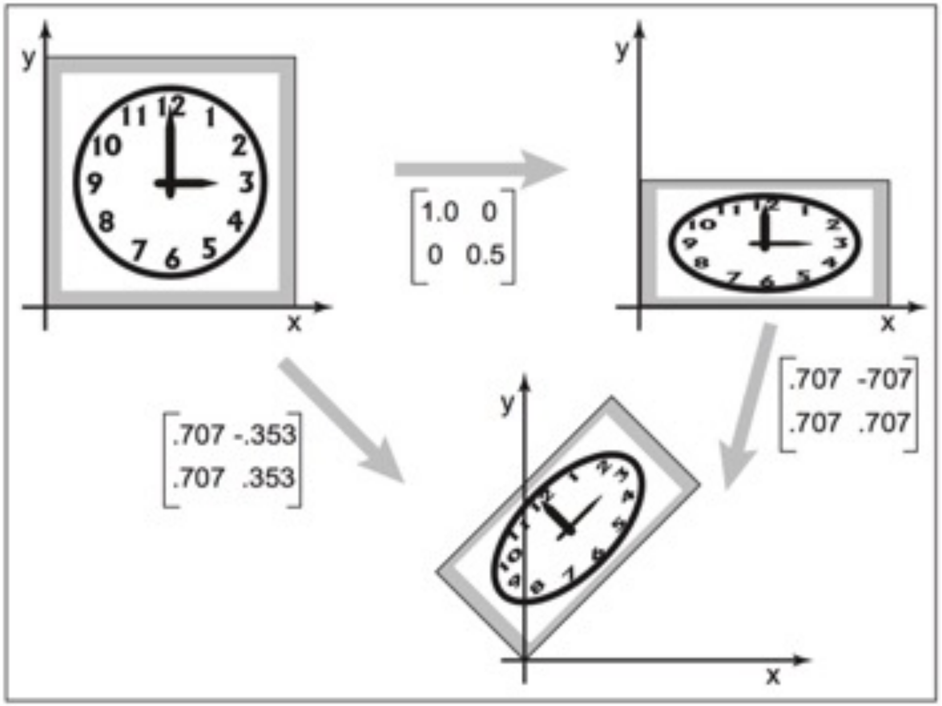
$$\mathbf{M} = \mathbf{RS}$$

- It is very important to remember that **these transforms are applied from the right side first**.
  - The matrix **M** = **RS** first applies **S** and then **R**
- Matrix multiplication is not commutative. So the order of transforms does matter!

[2]

Composition

Linear Transformations



$$\begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} = \begin{bmatrix} 0.707 & -0.353 \\ 0.707 & 0.353 \end{bmatrix}$$

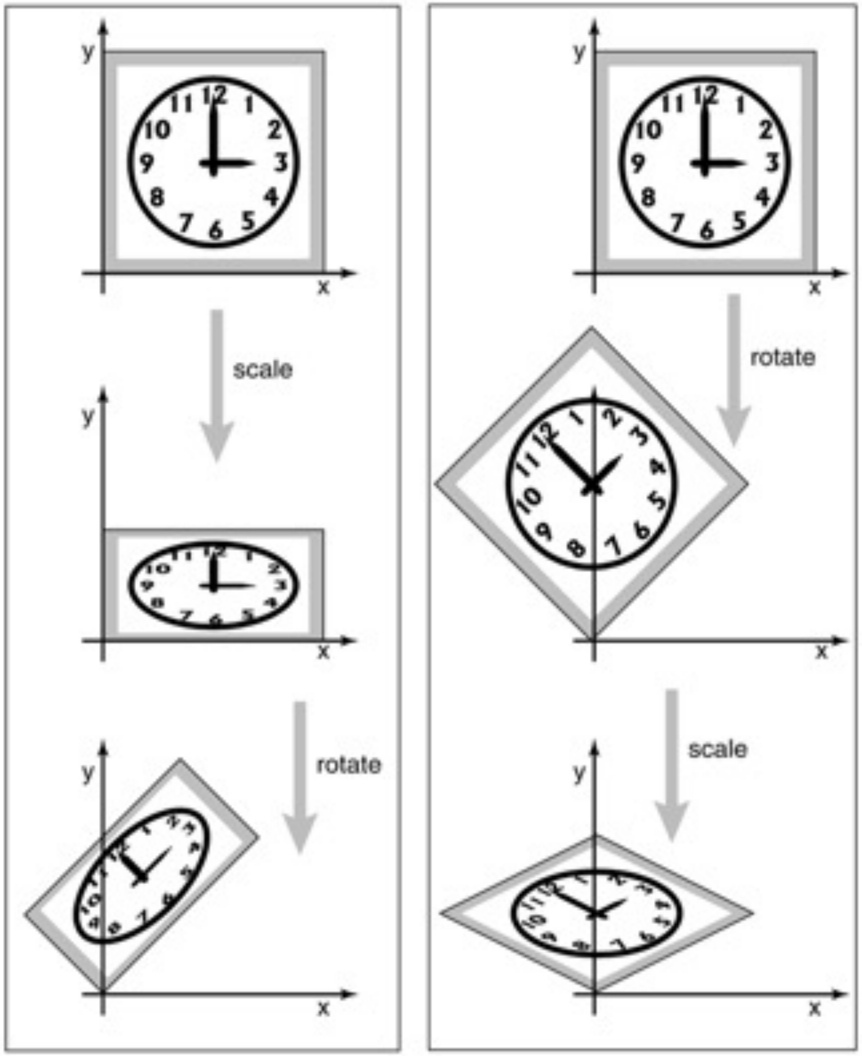
!=

$$\begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} = \begin{bmatrix} 0.707 & -0.707 \\ 0.353 & 0.353 \end{bmatrix}$$

[2]

Composition

Linear Transformations



[2]

- Transformations so far work similarly in 3D

$$\text{scale}(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} \qquad \text{shear-x}(d_y, d_z) = \begin{bmatrix} 1 & d_y & d_z \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

[2]

- Rotation is considerably more complicated in 3D than in 2D, because there are more possible axes of rotation.
- For now we simply want to rotate about one specific axis
  - This will only change the other two coordinates and we can use the 2D rotation matrix with no operation on the rotation axis:

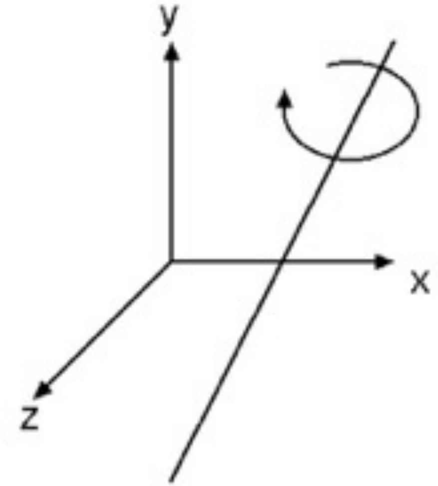
$$\textbf{rotate-x}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \quad \textbf{rotate-y}(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \quad \textbf{rotate-z}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

[2]

3D Rotation With An Arbitrary Axis

Linear Transformations

- Rotating around one arbitrary axis can be shown to work the following:
- Given a unit vector $\mathbf{u} = (u_x, u_y, u_z)$, the matrix for a rotation by an angle of θ about an axis in the direction of **u** is



$$R = \begin{bmatrix} \cos\theta + u_x^2\left(1-\cos\theta\right) & u_x u_y\left(1-\cos\theta\right) - u_z\sin\theta & u_x u_z\left(1-\cos\theta\right) + u_y\sin\theta \\ u_y u_x\left(1-\cos\theta\right) + u_z\sin\theta & \cos\theta + u_y^2\left(1-\cos\theta\right) & u_y u_z\left(1-\cos\theta\right) - u_x\sin\theta \\ u_z u_x\left(1-\cos\theta\right) - u_y\sin\theta & u_z u_y\left(1-\cos\theta\right) + u_x\sin\theta & \cos\theta + u_z^2\left(1-\cos\theta\right) \end{bmatrix}$$

[https://en.wikipedia.org/wiki/Rotation_matrix]

Affine Transformations

- So far all transforms have the form $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} m_{11} & m_{21} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$

$$\begin{aligned} x' &= m_{11}x + m_{12}y, \\ y' &= m_{21}x + m_{22}y. \end{aligned}$$

- With this the origin (0, 0) always remains fixed (under a *linear* transformation for scale and rotate).

Affine Transformations

- To move, or *translate*, an object by shifting all its points the same amount, we need a transform of the form,

$$\begin{aligned} x' &= x + x_t, \\ y' &= y + y_t. \end{aligned}$$

- There is no way to do that by multiplying (x, y) by a 2×2 matrix.

2D Translation

Affine Transformations

$$
\begin{aligned}
x' &= x + x_t, \\
y' &= y + y_t.
\end{aligned}
$$

- We could associate a separate translation vector with each transformation matrix, letting the matrix take care of scaling and rotation and the vector take care of translation.
  - This is perfectly feasible, but the bookkeeping is awkward and the rule for composing two transformations is not as simple and clean as with linear transformations.

2D Translation

Affine Transformations

- Instead, we can use a clever trick to get a single matrix multiplication to do both operations together.
  The idea is simple:
  - Represent the point (x, y) by a 3D vector $[x\ y\ 1]^{\mathrm{T}}$ and use 3 × 3 matrices of the form

$$\begin{bmatrix} m_{11} & m_{12} & x_t \\ m_{21} & m_{22} & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D Translation

Affine Transformations

- Adding an extra dimension is called *homogeneous* coordinates.
- This kind of coordinates are commonly used in graphics.

Homogeneous Coordinates

Affine Transformations

$$\begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- The following are all the same point

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \longleftrightarrow \begin{bmatrix} 3x \\ 3y \\ 3 \end{bmatrix} \longleftrightarrow \begin{bmatrix} \frac{x}{2} \\ \frac{y}{2} \\ \frac{1}{2} \end{bmatrix}$$

- "Dehomogenisation"

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \longrightarrow \begin{bmatrix} \frac{x}{w} \\ \frac{y}{w} \end{bmatrix}$$

Translation

Affine Transformation

$$\operatorname{translate}(x, y) = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix}$$

Affine Transformation

- Now, a single matrix can implement a linear transformation followed by a translation.

Rotation Translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & x_t \\ m_{21} & m_{22} & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Scale

Affine Transformation

- Now, a single matrix can implement a linear transformation followed by a translation.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & x_t \\ m_{21} & m_{22} & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- This kind of transformation is called an *affine* transformation.

Example

Affine Transformations

- With being able to translate we can now rotate around any point!
  - Translate point to the origin
  - Rotate
  - Translate point back to original position

Example

Affine Transformations

- Rotation (in 2D) around point $\mathbf{c} = (c_x, c_y)$ with angle ϕ
- Steps: $\mathbf{T}(-\mathbf{c}) \rightarrow \mathbf{R_z}\, \phi \rightarrow \mathbf{T}(\mathbf{c})$

$$\begin{pmatrix} 1 & 0 & c_y \\ 0 & 1 & c_x \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & -c_y \\ 0 & 1 & -c_x \\ 0 & 0 & 1 \end{pmatrix}$$

Example

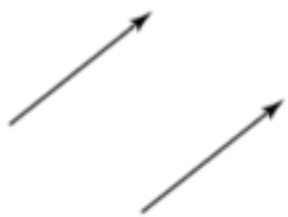
Affine Transformations

- Rotation (in 2D) around point $\mathbf{c} = (c_x, c_y)$ with angle ϕ
- Steps: $\mathbf{T}(-\mathbf{c}) \rightarrow \mathbf{R_z}\,\phi \rightarrow \mathbf{T}(\mathbf{c})$

$$\begin{pmatrix} 1 & 0 & c_y \\ 0 & 1 & c_x \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & -c_y \\ 0 & 1 & -c_x \\ 0 & 0 & 1 \end{pmatrix}$$

Affine Transformation

- If a vector represents a direction, it should not change through translation



Affine Transformation

- For this we simply set the third coordinate to zero:

$$\begin{bmatrix} 1 & 0 & x_t \\ 0 & 1 & y_t \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$$

- If there is a scaling/rotation transformation in the upper-left 2×2 entries of the matrix, it will apply to the vector, but the translation still multiplies with the zero and is ignored.

Affine Transformation

- The extra (third) coordinate will be either 1 or 0 depending on whether we are encoding a position or a direction. Now we can distinguish between locations and other vectors.

$\begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix}$ is a location and

$\begin{bmatrix} 3 \\ 2 \\ 0 \end{bmatrix}$ is a direction or displacement

Affine Transformation

- The extra (third) coordinate will be either 1 or 0 depending on whether we are encoding a position or a direction. Now we can distinguish between locations and other vectors.

$$\begin{bmatrix} 3 \\ 2 \\ \boxed{1} \end{bmatrix}$$ is a location and

$$\begin{bmatrix} 3 \\ 2 \\ \boxed{0} \end{bmatrix}$$ is a direction or displacement

3D Translation

Affine Transformation

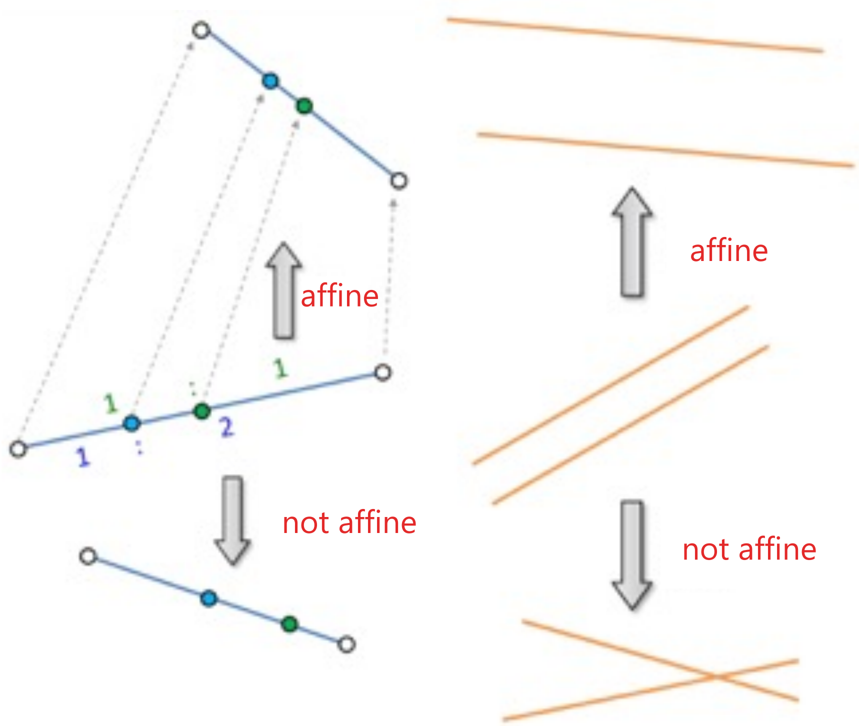
- In 3D a fourth coordinate is added accordingly:

$$\begin{bmatrix} 1 & 0 & 0 & x_t \\ 0 & 1 & 0 & y_t \\ 0 & 0 & 1 & z_t \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + x_t \\ y + y_t \\ z + z_t \\ 1 \end{bmatrix}$$

Properties

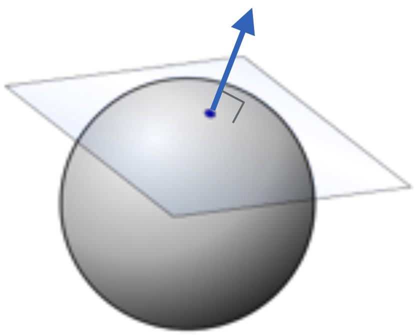
Affine Tranformations

- Lines are projected onto lines
- Parallel lines remain parallel
- Ratio of distances remain
- **Not angle-preserving**
- Examples: rotation, translation, scaling, shearing



Transforming Normal Vectors

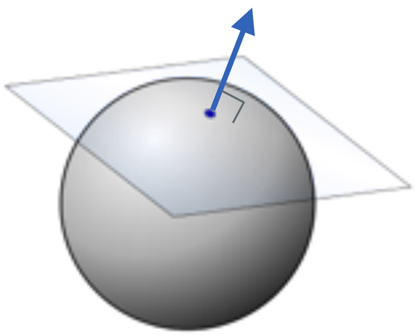
- Normals do not transform the same way as its underlying surface.



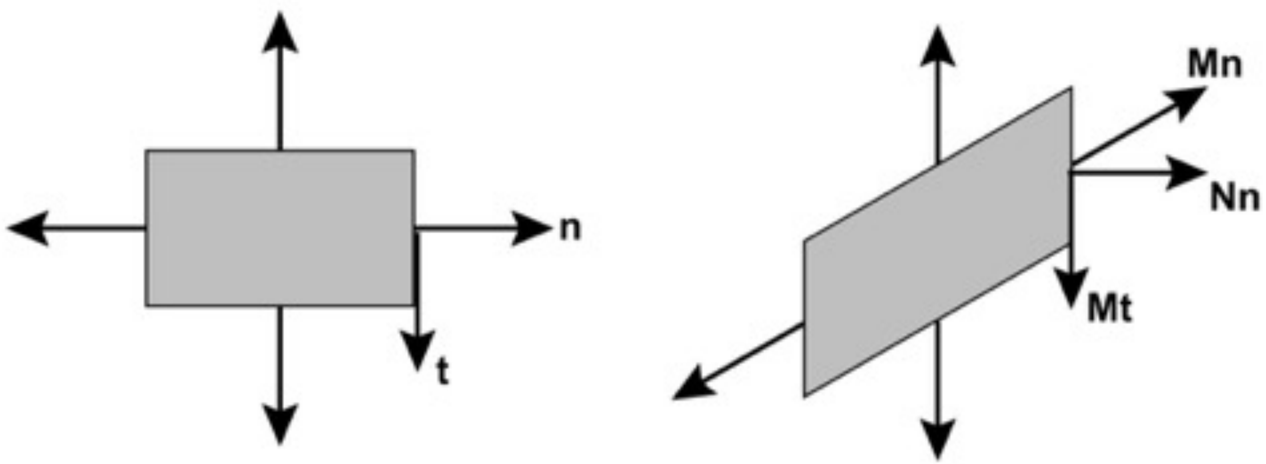
https://en.wikipedia.org/wiki/Tangent
https://en.wikipedia.org/wiki/Normal_(geometry)

Transforming Normal Vectors

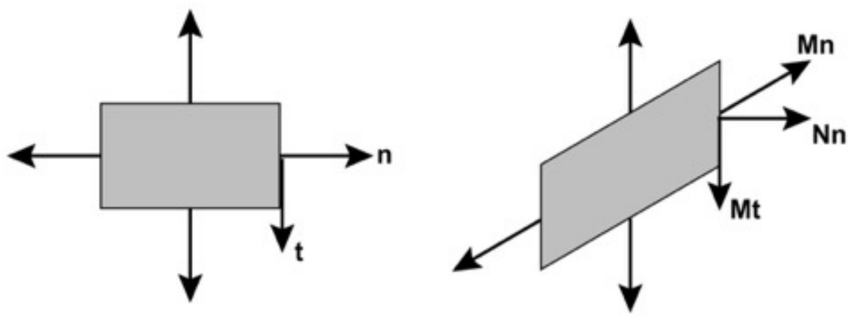
- Normals do not transform the same way as its underlying surface.
  - A surface normal vector **n** that is transformed by **M** may not be normal to the transformed surface
- Example:



https://en.wikipedia.org/wiki/Tangent
https://en.wikipedia.org/wiki/Normal_(geometry)



Transforming Normal Vectors



- The matrix **N** which correctly transforms normal vectors so they remain normal is the transpose of the inverse matrix.

$$\mathbf{N} = (\mathbf{M}^{-1})^{\mathrm{T}}$$

Literature

1. https://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/geometry/points-vectors-and-normals
2. Shirley, P. and Marschner, S. 2012. Fundamentals of computer graphics. CRC, Boca Raton [u.a.].
3. https://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/geometry/matrices