Matrices - From intuitive Understanding to application

Dheeraj

What is Matrix ?

Indian Institute of Technology Mandi

Scaling the Heights

Matrix

Expectation:

Reality:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrix Multiplication

$$\begin{pmatrix} 1 & 2 & 8 \\ 6 & 3 & 8 \\ 8 & 5 & 6 \end{pmatrix} \cdot \begin{pmatrix} 2 & 4 & 2 \\ 1 & 5 & 1 \\ 7 & 2 & 8 \end{pmatrix} =$$

Matrix Multiplication

$$\begin{pmatrix} 1 & 2 & 8 \\ 6 & 3 & 8 \\ 8 & 5 & 6 \end{pmatrix} \cdot \begin{pmatrix} 2 & 4 & 2 \\ 1 & 5 & 1 \\ 7 & 2 & 8 \end{pmatrix} = \begin{pmatrix} 60 & 30 & 68 \\ 71 & 55 & 79 \\ 63 & 69 & 69 \end{pmatrix}$$

Properties of Matrix

1. Associative

2. Commutative

3. Identity

4. Inverse

Associative Property

Let A and B be 2 Matrices -

Addition:

$(A + B) + C = A + (B+C)$

Multiplication:

$(AB)C = A(BC)$

Commutative Property

Let A and B be 2 Matrices -

Addition:

$A + B = B + A$

Multiplication (Not Commutative):

$AB \neq BA$

Identity Property

Let A be a matrix, I be Identity matrix and O be an unique matrix such that-

Addition:

$A + O = A$

Multiplication:

$AI = A$

$IA = A$

Inverse Property

Let A be a matrix, I be Identity matrix and O be an unique matrix such that-

Addition:

$A + (-A) = O$

Multiplication:

$AA^{-1} = I$

$A^{-1}A = I$

Problem Solving

If A is a 3 × 3 non-singular matrix such that $AA^T = A^TA$ and $B = A^{-1}A^T$, then BB^T equals

(a) $I+B$

(b) I

(c) B^{-1}

(d) $(B^{-1})^T$

Solution

Given that $AA^T = A^TA$ and $B = A^{-1}A^T$

$BB^T = (A^{-1}A^T)\,(A^{-1}A^T)^T$

$= A^{-1}A^T\,A(A^{-1})^T$ (since $(A^T)^T = A$)

$= A^{-1}A\,A^T(A^T)^{-1}$

$= I.I$

$= I$

Hence option b is the answer.

Understanding each element of Matrix

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 8 \\ 6 & 3 & 8 \\ 8 & 5 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 8 \\ 6 & 3 & 8 \\ 8 & 5 & 6 \end{pmatrix}$$

Basis

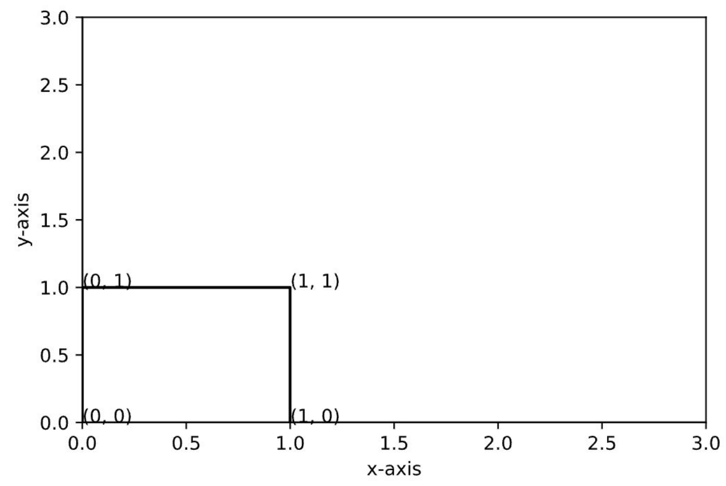
$$\begin{array}{cc} i & j \\ \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{array}$$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = x\,i + y j$$

Activity 1

1. Find Final coordinates of the square

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} a & 0 \\ 0 & 1 \end{pmatrix}$$

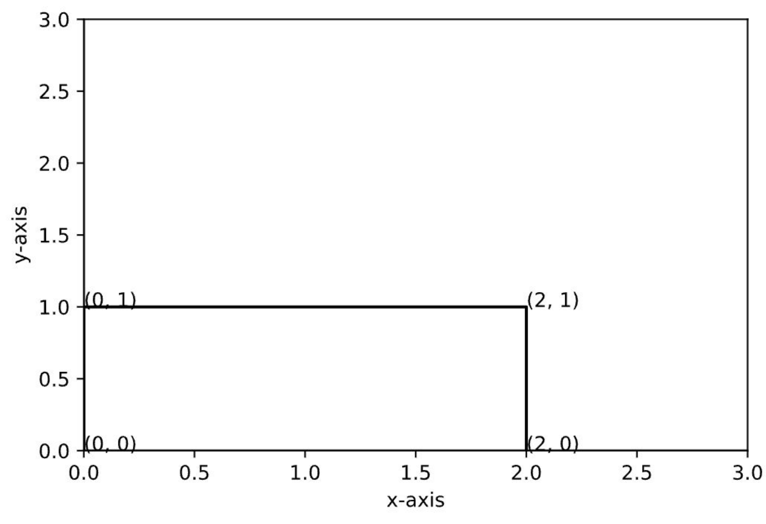


Scaling

1. Horizontal Expansion

$$\begin{pmatrix} a & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & a & a & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

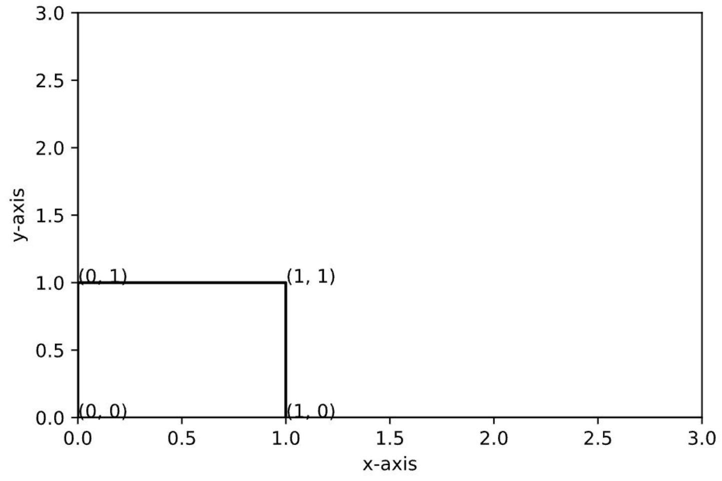
Putting a = 2



Activity 2

2. Find Final coordinates of the square

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & b \end{pmatrix}$$

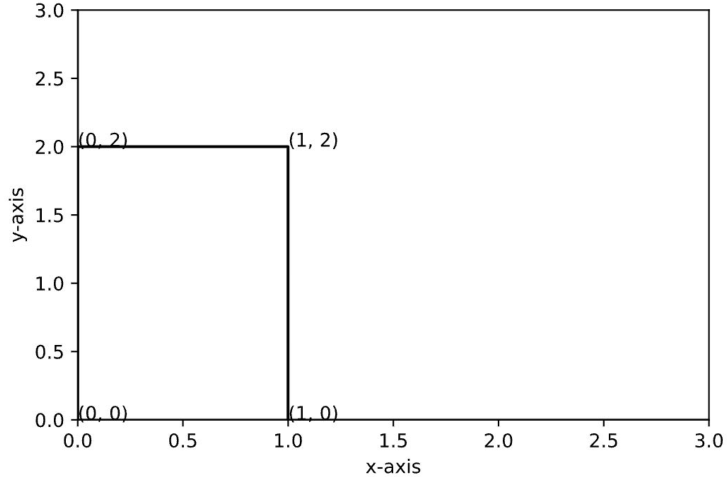


Scaling

2. Vertical Expansion

$$\begin{pmatrix} 1 & 0 \\ 0 & b \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & b & b \end{pmatrix}$$

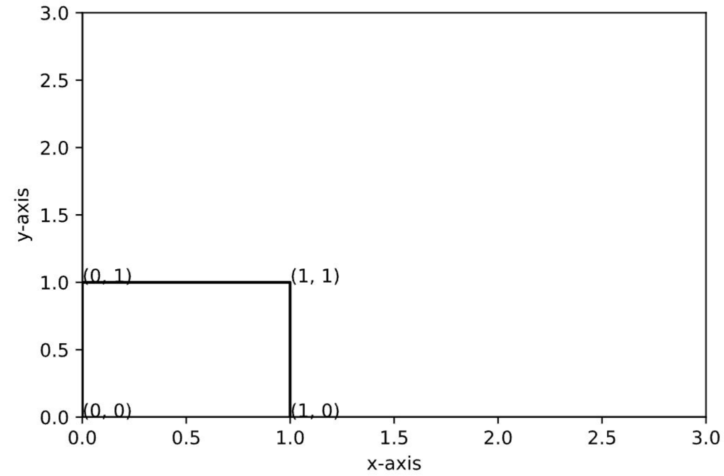
Putting b = 2



Activity 3

3. **Find Final coordinates of the square**

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

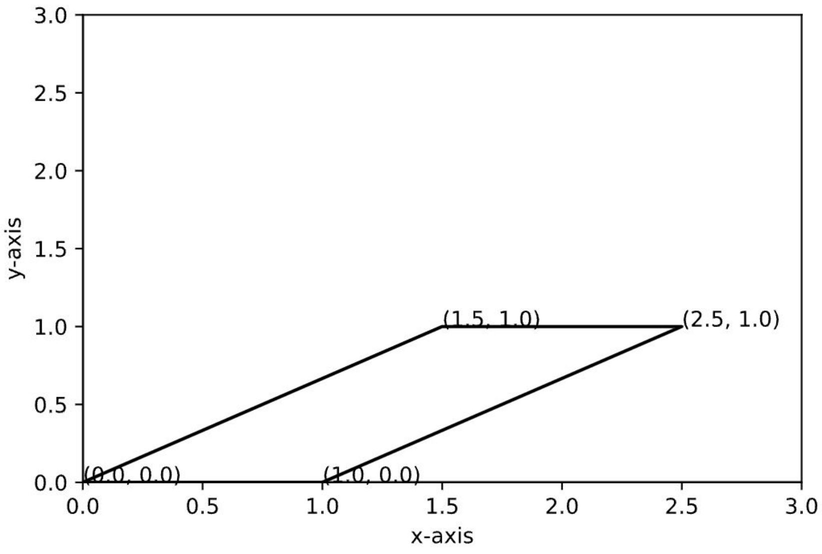


Shearing

3. Horizontal Shearing

$$\begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1+k & k \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

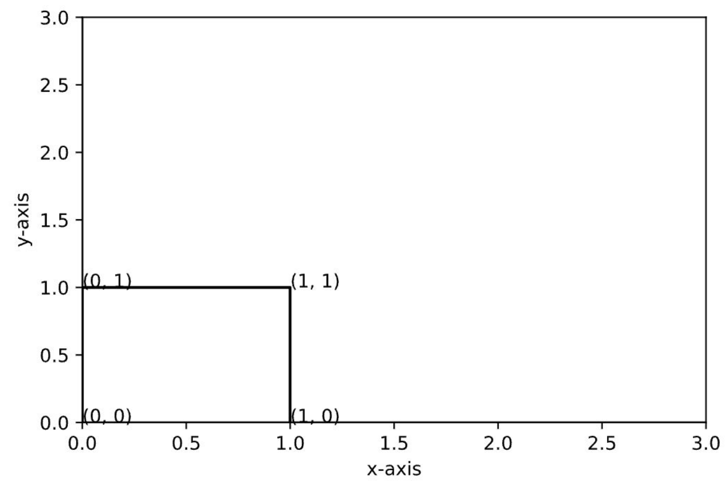
Putting k = 1.5, where k is the horizontal shear factor



Activity 4

4. Find Final coordinates of the square

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ k & 1 \end{pmatrix}$$

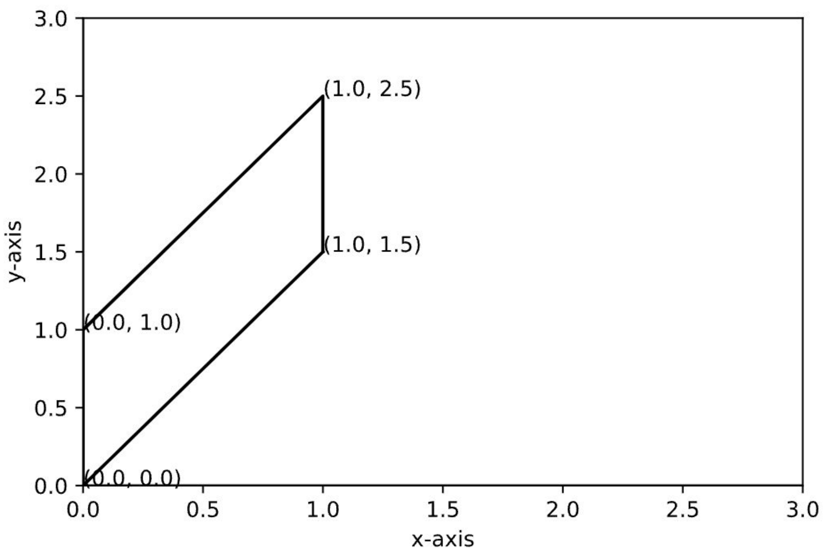


Shearing

4. Vertical Shearing

$$\begin{pmatrix} 1 & 0 \\ k & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & k & k+1 & 1 \end{pmatrix}$$

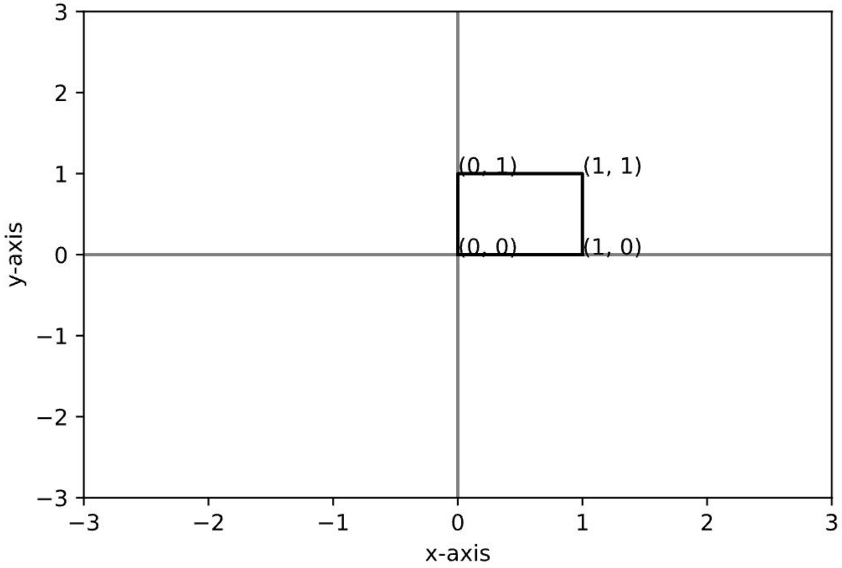
Putting k = 1.5, where k is the vertical shear factor



Reflections

5. Reflection along x-axis (y=0)

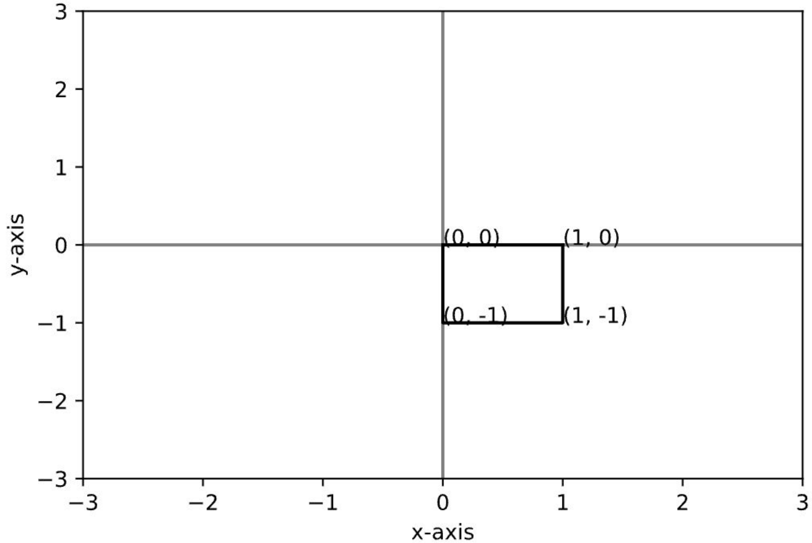
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

5. Reflection along x-axis (y=0)

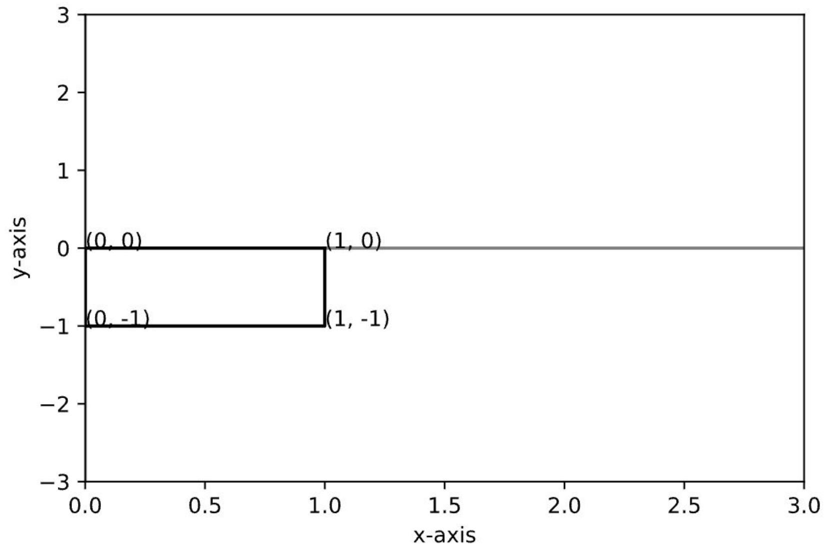
$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & -1 \end{pmatrix}$$



Reflections

5. Reflection along x-axis (y=0)

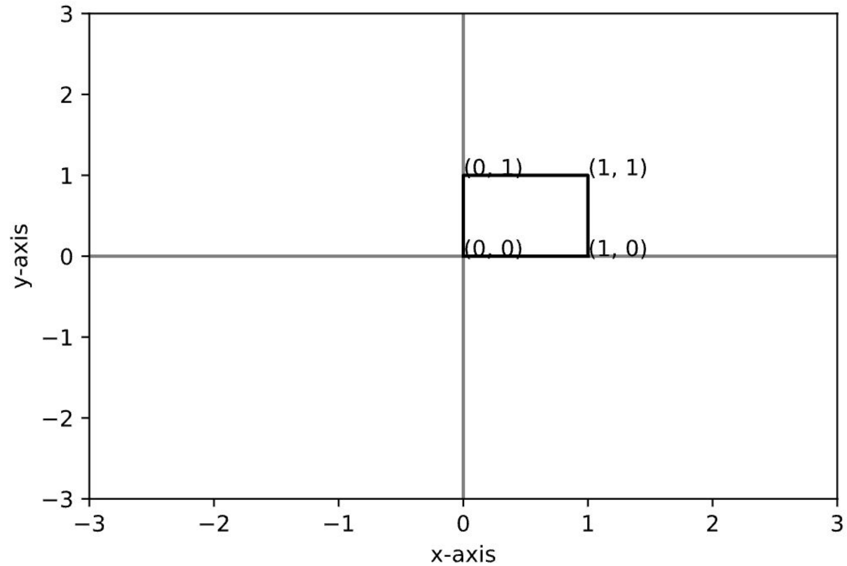
$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & -1 \end{pmatrix}$$



Reflections

6. Reflection along y-axis (x=0)

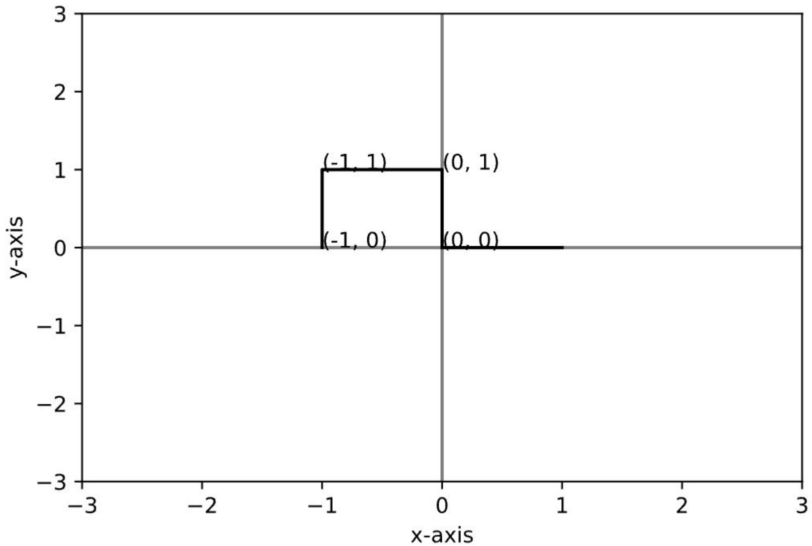
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

6. Reflection along y-axis (x=0)

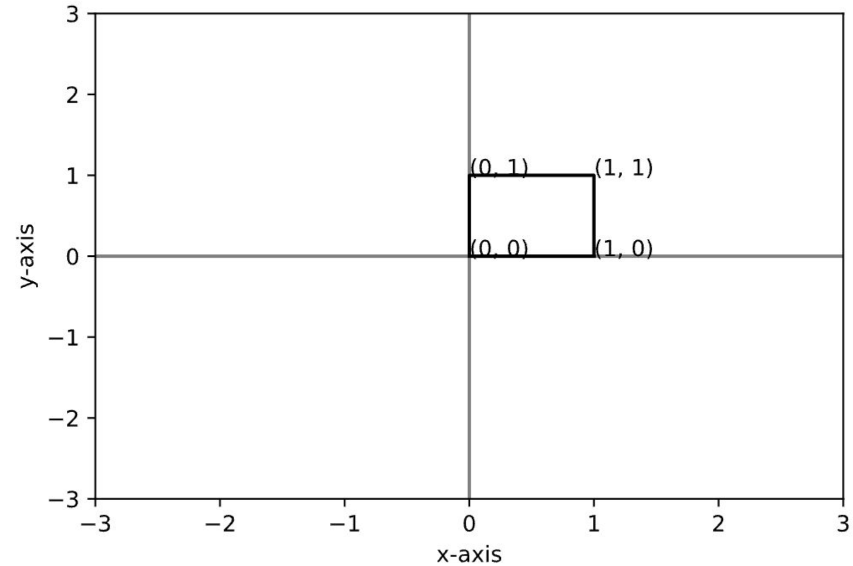
$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

7. Reflection along line y=x

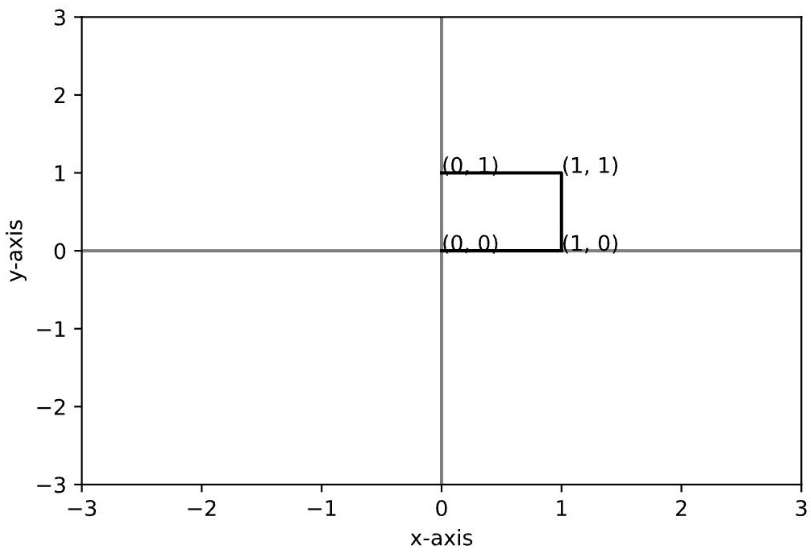
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

7. Reflection along line y=x

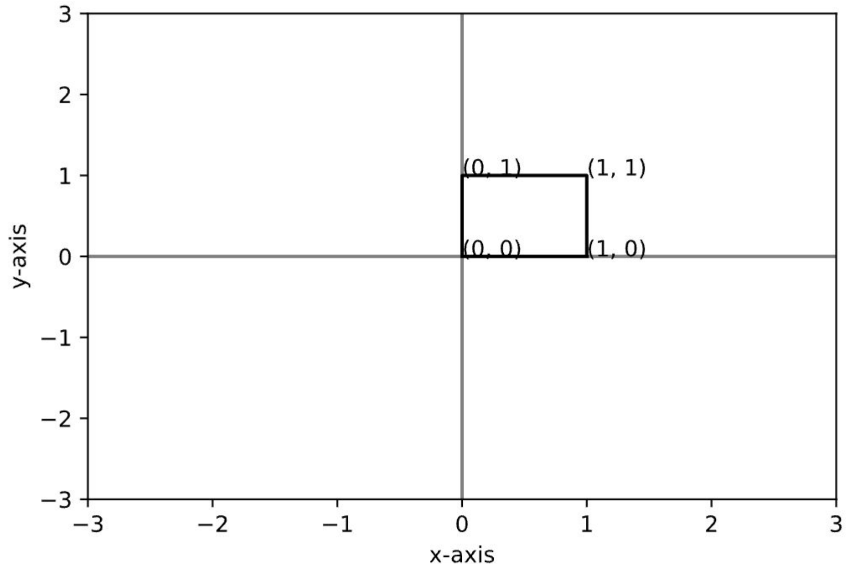
$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$



Reflections

8. Reflection along line y=-x

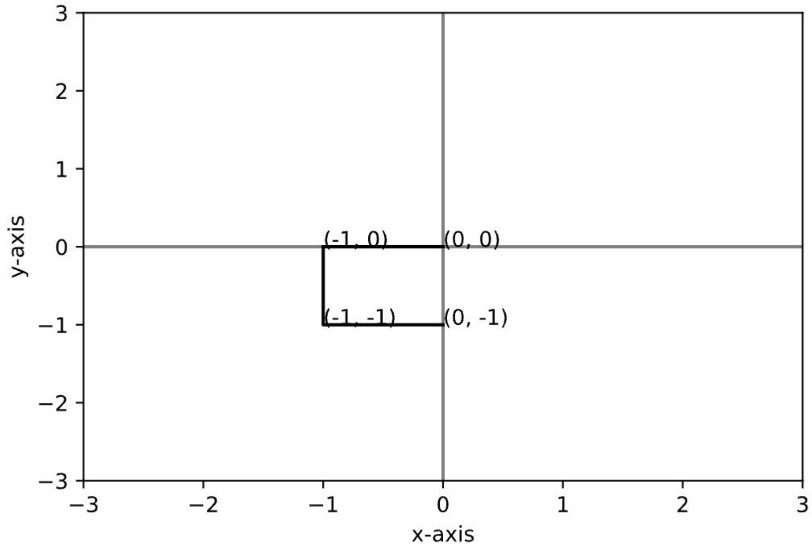
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

8. Reflection along line y=-x

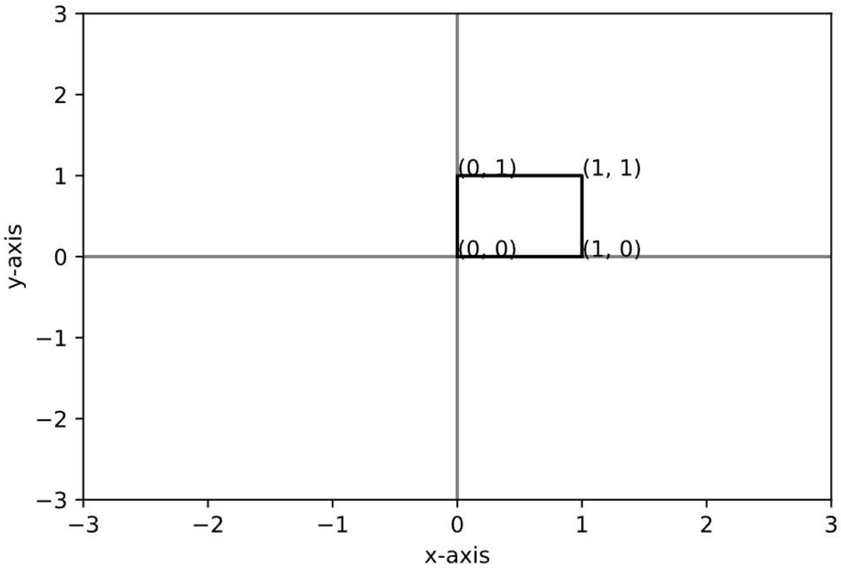
$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & -1 & -1 \end{pmatrix}$$



Reflections

9. Reflection along origin (0,0)

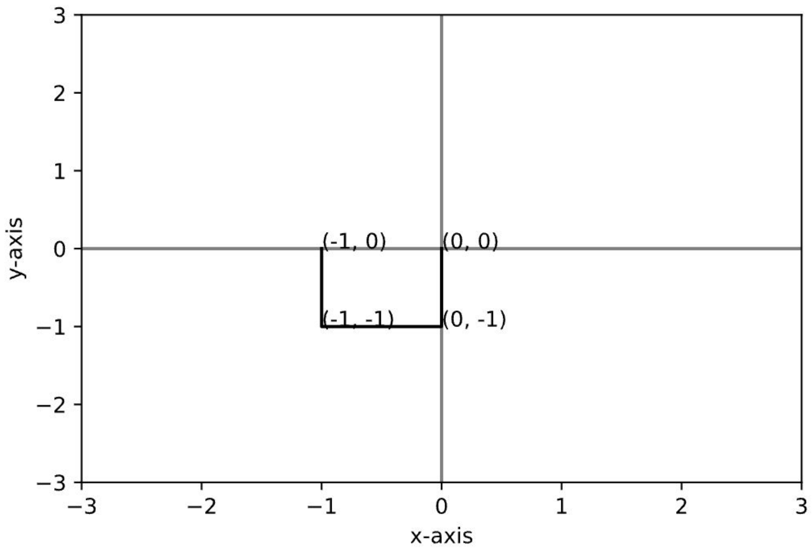
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$



Reflections

9. Reflection along origin (0,0)

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & -1 & -1 \end{pmatrix}$$



Basis

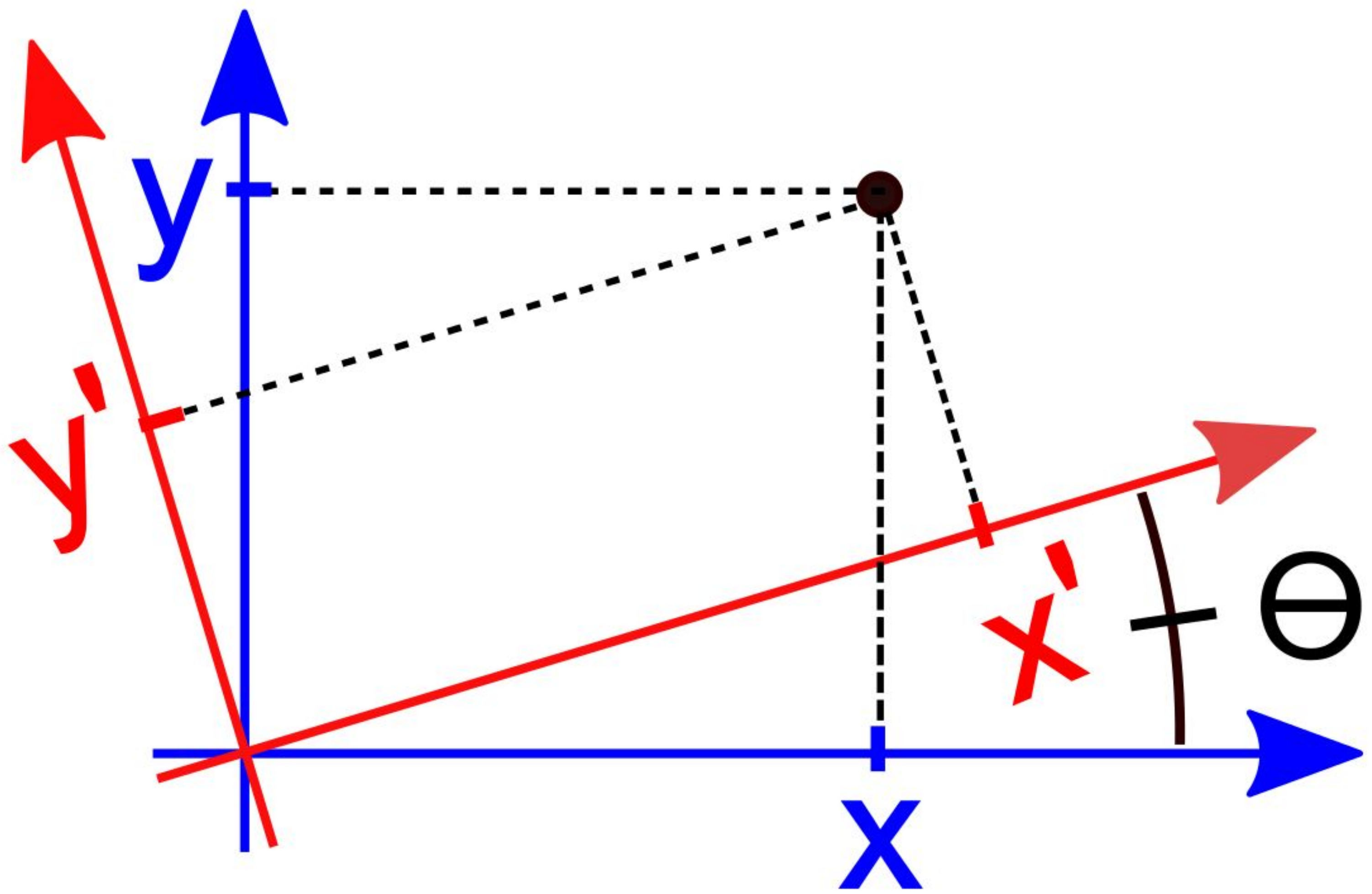
$$\begin{array}{cc} \boldsymbol{i} & \boldsymbol{j} \\ \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{array}$$

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = x\,i + y\,j$$

Basis

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix} = x \begin{pmatrix} a \\ c \end{pmatrix} + y \begin{pmatrix} b \\ d \end{pmatrix}$$

Find Rotation Matrix



Multiply without actually multiplying

$$\begin{pmatrix} 1 & 0 & 0 \\ -1/3 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 \\ 1/3 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Find the Inverse

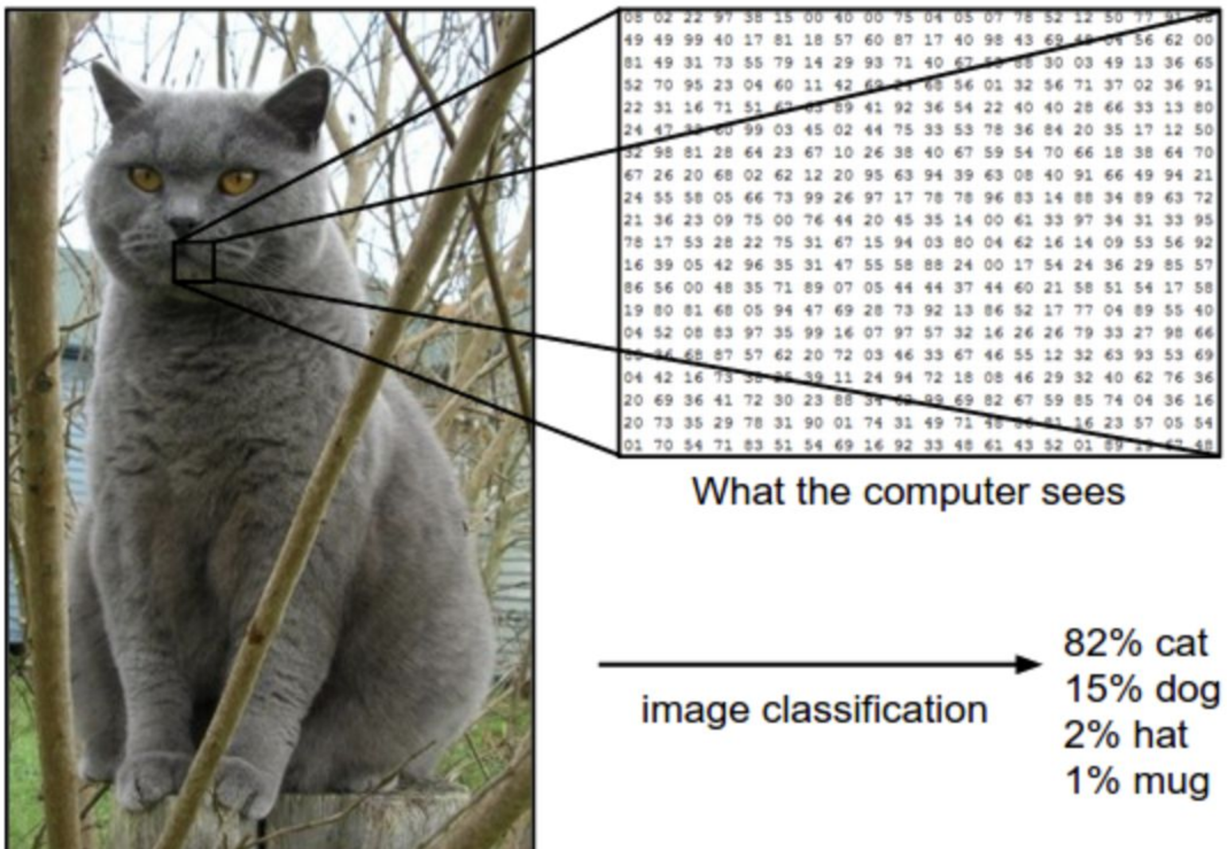
Without using:

1. **Gauss Jordan Elimination**
2. **Classical Adjoint**
3. **Partition method**

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 0 & -1 \\ 1 & 3 & 0 \end{pmatrix}$$

Applications

1. Image Classification



To a computer an image is represented as one large 2-dimensional array of numbers as shown in the figure on the left.

Applications

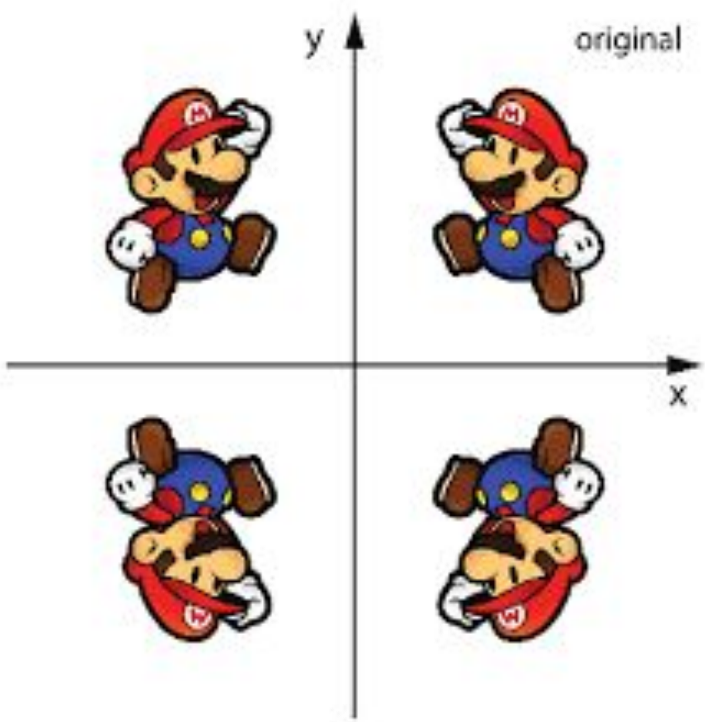
1. Image Classification

$$\left| \overset{\text{test image}}{\begin{bmatrix} 56 & 32 & 10 & 18 \\ 90 & 23 & 128 & 133 \\ 24 & 26 & 178 & 200 \\ 2 & 0 & 255 & 220 \end{bmatrix}} - \overset{\text{training image}}{\begin{bmatrix} 10 & 20 & 24 & 17 \\ 8 & 10 & 89 & 100 \\ 12 & 16 & 178 & 170 \\ 4 & 32 & 233 & 112 \end{bmatrix}} \right| = \overset{\text{pixel-wise absolute value differences}}{\begin{bmatrix} 46 & 12 & 14 & 1 \\ 82 & 13 & 39 & 33 \\ 12 & 10 & 0 & 30 \\ 2 & 32 & 22 & 108 \end{bmatrix}} \rightarrow 456$$

Using pixel-wise differences to compare two images by subtracting them elementwise and then all differences are added up to a single number.
If two images are identical the result will be zero.
But if the images are very different the result will be large.

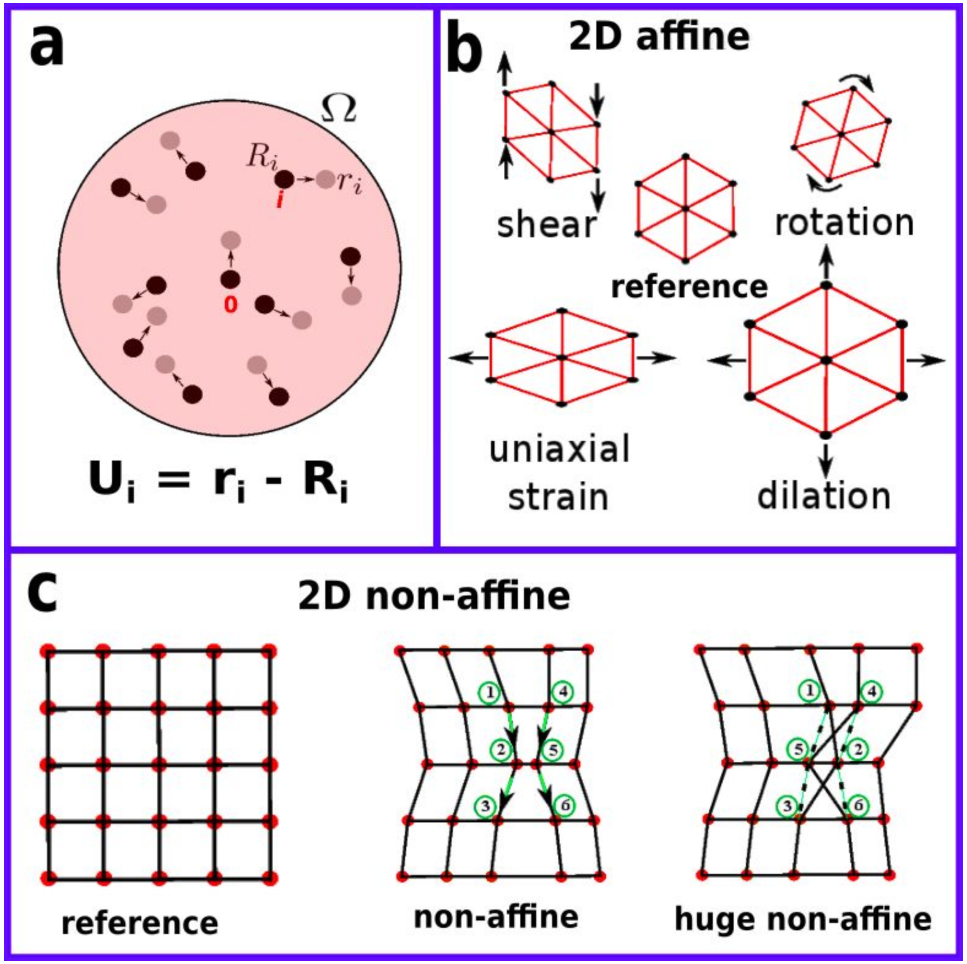
Applications

2. Graphics and Gaming

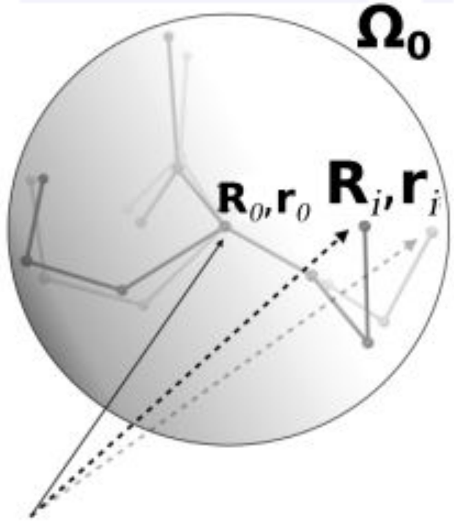


A 3D object in a game that needs to be translated, rotated, and scaled can be done using matrices.
A transformation matrix is used to transform the position, orientation, and size of an object in a 3D space.
Matrices are also used to perform lighting and shading operations in graphics and gaming.

Difference between Affine and Non-affine Displacements



Formulation



$u_i = r_i - R_i$

$u_i = D\, R_i$ (Complete Affineness)

$u_i = D\, R_i + u_{i,non\text{-}affine}$ (general case)

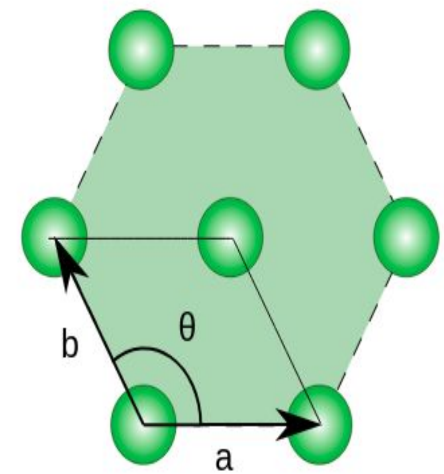
For reduced displacements

$\Delta_i = u_i - u_0$

$u_0 = D\, R_0 + u_{0,non\text{-}affine}$

$\Delta_i = D\,(R_i - R_0) + \Delta_{i,non\text{-}affine}$

$\mathrm{NAP} = \Sigma_i\, |\Delta_{i,\mathrm{nonaffine}}|^2 = \min_D[\Sigma_i\, |\Delta_i - D$



$$NAP = \sum_i |\Delta_{i,non-affine}|^2 = \min_{\mathcal{D}} [\sum_i |\Delta_i - D(R_i - R_0)|^2]$$

https://github.com/dheeraj08dube/GNAP_CALCULATION.git

PloS Comput Biol 15(1):e1006665.

Alternate Interpretation of Minimization

$$NAP = \sum_i |\Delta_{i,non-affine}|^2 = \min_{\mathcal{D}}\left[\sum_i |\Delta_i - D(R_i - R_0)|^2\right] \Longleftrightarrow NAP = S^T S$$

$$S = \underset{\Delta}{\begin{pmatrix} \Delta_{1x} \\ \Delta_{1y} \\ \Delta_{1z} \\ . \\ . \\ . \\ \Delta_{Nx} \\ \Delta_{Ny} \\ \Delta_{Nz} \end{pmatrix}} - \underset{\mathcal{R}}{\begin{pmatrix} (R_1-R_0)_x & (R_1-R_0)_y & (R_1-R_0)_z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (R_1-R_0)_x & (R_1-R_0)_y & (R_1-R_0)_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (R_1-R_0)_x & (R_1-R_0)_y & (R_1-R_0)_z \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ (R_N-R_0)_x & (R_N-R_0)_y & (R_N-R_0)_z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (R_N-R_0)_x & (R_N-R_0)_y & (R_N-R_0)_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (R_N-R_0)_x & (R_N-R_0)_y & (R_N-R_0)_z \end{pmatrix}} \underset{e}{\begin{pmatrix} D_{xx} \\ D_{xy} \\ D_{xz} \\ D_{yx} \\ D_{yy} \\ D_{yz} \\ D_{zx} \\ D_{zy} \\ D_{zz} \end{pmatrix}}$$

$$\frac{d(S^T S)}{de} = 0 \Rightarrow NAP = \Delta^T P \Delta$$

$$\text{where } P = I - (R(R^T R)^{-1} R^T)$$

PLoSComputBiol 15(1):e1006665.https://doi.org/10.1371/journal.pcbi.1006665

Applications

3. Cryptography

Multiply the column vector with E to generate a matrix D (decrpytion matrix) given to our receiver.

$$EC = D$$
$$C = E^{-1}D$$

To decrypt and find the message, the receiver will have to multiply the Decryption matrix with inverse of E.

This technique is known as **“Matrix Cypher”**

Now you all make a (one word) secret message and see if the person sitting next to you can decode it or not.

Applications

3. Cryptography

Let E be our encryption matrix and our message be “MATRIX”, each letter can be uniquely mapped to its rank in alphabets. E.g. A=1, B=2 etc.

Take 2 letter at a time to make 2x1 column vector.
E.g. “MA” = $[13\ 1]^T$

$$E = \begin{pmatrix} 3 & 2 \\ 1 & 4 \end{pmatrix}$$

Taking a 2x2 matrix will make computation of inverse easy for us.