3D Geometry

Reading: Angel Ch.4 + Appendix B&C

A Review of Geometry

How to represent and transform 3D shapes?

Primitive Objects: **Scalars** - real-numbers
Points - location in space
Vectors - directed line between 2 points

Representation independent of the coordinate frame

Mathematics used in computer graphics based on **'abstract spaces'**

- **Vector** space (vectors/scalars)
- **Affine** space (vectors/scalars + points)
- **Euclidean** space (vectors/scalars/points + distance)

Representation in a particular coordinate frame leads to

Scalars

Scalars are real numbers $a, b, c \in \Re$

Two fundamental operations:

Addition: $c = a + b$

Multiplication: $c = a.b = ab$

Operations are:

Associative: $a + b = b + a$

$$a.b = b.a$$

Commutative: $a + (b + c) = (a + b) + c$

$$a.(b.c) = (a.b).c$$

Distributive: $a.(b + c) = (a.b) + (a.c)$

Real numbers using addition/multiplication form a **scalar field**
other examples are complex numbers and rational functions

Points

Point is a location in space: $P, Q \in \Re^N$

- **independent coordinate system** (exists without it)
- **no size**
- specifying a coordinate system defines the relative location of a point to the origin
- addition and multiplication of points is not defined

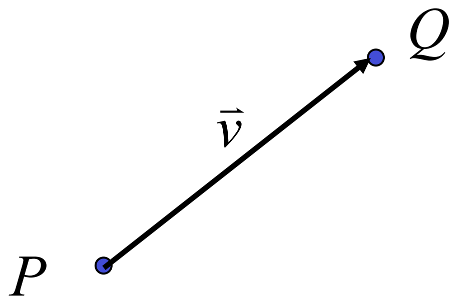
Vectors

Vector is the directed line between 2 points: $\vec{v} \in \Re^N$

- **no fixed location**
- **has direction & magnitude**
- vector-vector addition is defined

$$\vec{v} = Q - P$$

$$Q = \vec{v} + P$$



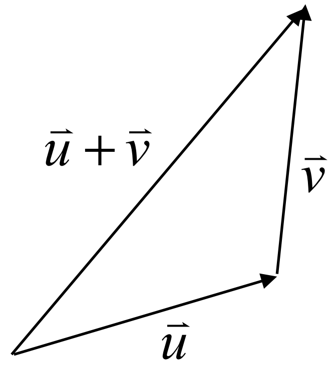
Vector Spaces

Vector spaces V contain scalars & vectors $\vec{u}, \vec{v} \in \Re^N$

Vectors have two operations:

vector-vector additions: $\vec{w} = \vec{u} + \vec{v}$

scalar-vector multiplication: $\vec{w} = a\vec{v}$



head-to-tail rule

Properties of operations:

vector addition is closed, $\vec{u} + \vec{v} \in \Re^N$

commutative $\vec{u} + \vec{v} = \vec{v} + \vec{u}$

& associative: $\vec{u} + (\vec{v} + \vec{w}) = (\vec{u} + \vec{v}) + \vec{w}$

Zero-vector: $\vec{0} \in \Re^N$

$$\vec{u} + \vec{0} = \vec{u}$$

$$\vec{u} + (-\vec{u}) = \vec{0}$$

Vector Space II

scalar-vector muliplication is distributive: $a(\vec{u}+\vec{v}) = a\vec{u} + a\vec{v}$

hence, $(a+b)\vec{u} = a\vec{u} + b\vec{u}$

- changes magnitude not direction

A vector can be expressed uniquely as a linear combination of a set of *N* basis vectors $(\vec{v}_1, \vec{v}_2, \vec{v}_3, \ldots, \vec{v}_N)$

$$\boxed{\vec{v} = a_1\vec{v}_1 + a_2\vec{v}_2 + a_3\vec{v}_3 + \ldots + a_N\vec{v}_N = \sum_{i=1}^{N} a_i\vec{v}_i = V[a_i]}$$

V is a matrix of basis vectors *[a_i] is a N-vector of coefficients*

where the **basis vectors are linearly independent**: $\vec{v}_j \neq \sum_{i=1, i\neq j}^{N} b_i\vec{v}_i$

Linearly independent if $\sum_{i=1}^{N} a_i\vec{v}_i = \vec{0}$

only when $a_1 + a_2 + a_3 + \ldots + a_N = 0$

Vector Spaces III

Change of basis

given a different basis $(\vec{v}'_1, \vec{v}'_2, \vec{v}'_3, \ldots, \vec{v}'_N)$

$$\vec{v}' = a'_1 \vec{v}'_1 + a'_2 \vec{v}'_2 + a'_3 \vec{v}'_3 + \ldots + a'_N \vec{v}'_N = \sum_{i=1}^{N} a'_i \vec{v}'_i = V'[a'_i]$$

there exists an nxn matrix M such that: $\boxed{[v'_i]^T = M[v_i]^T}$

$$[v_i]^T = \begin{bmatrix} v_{11} & \cdots & v_{n1} \\ \vdots & \ddots & \vdots \\ v_{1n} & \cdots & v_{nn} \end{bmatrix} \quad [v'_i]^T = \begin{bmatrix} v'_{11} & \cdots & v'_{n1} \\ \vdots & \ddots & \vdots \\ v'_{1n} & \cdots & v'_{nn} \end{bmatrix} \quad M = \begin{bmatrix} \gamma_{11} & \cdots & \gamma_{1n} \\ \vdots & \ddots & \vdots \\ \gamma_{n1} & \cdots & \gamma_{nn} \end{bmatrix}$$

M enables us to change between different sets of basis vectors $\vec{v}, \vec{v}'$

Change of Basis

$$\boxed{[v'_i]^T = M[v_i]^T}$$

Can use M to transform a the representation of a vector from basis v_i to basis v'_i

In basis v_i $\quad w = a^T [v_i]^T$ $\quad$ and in v'_i $\quad w = a'^T [v_i']^T = a'^T M[v_i]^T$

where a = [a_1, a_2 …. a_n] is the coefficient representing the vector in basis v

Therefore:

$$\boxed{\begin{aligned} a^T &= a'^T M \\ a &= M^T a' \\ a' &= (M^T)^{-1} a \end{aligned}}$$

Using the above relationship M can be used to transform the representation of a vector between bases

Example: Change of Basis

Transform vector v=(1,1,1) from basis v to basis u

$$v_1 = (1,0,0) \quad u_1 = (0,1,0)$$
$$v_2 = (0,1,0) \quad u_2 = (-1,0,0)$$
$$v_3 = (0,0,1) \quad u_3 = (0,0,1)$$

$$[u_i]^T = M[v_i]^T$$

Gives 9 simultaneous equations

$$u_{11} = \gamma_{11}v_{11} + \gamma_{12}v_{12} + \gamma_{13}v_{13} = 0 = \gamma_{11}$$
$$u_{12} = \gamma_{21}v_{11} + \gamma_{22}v_{12} + \gamma_{23}v_{13} = 1 = \gamma_{21}$$
$$u_{13} = \gamma_{31}v_{11} + \gamma_{32}v_{12} + \gamma_{33}v_{13} = 0 = \gamma_{31}$$
$$\ldots$$

Solving for M gives

$$M = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} A = (M^T)^{-1} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Tranforming representation of vector v=(1,1,1) to basis u

$$a'^T = a^T A$$
$$u^T = v^T A = (1,-1,1)$$

Affine Spaces

Vector spaces lack geometric concept of location

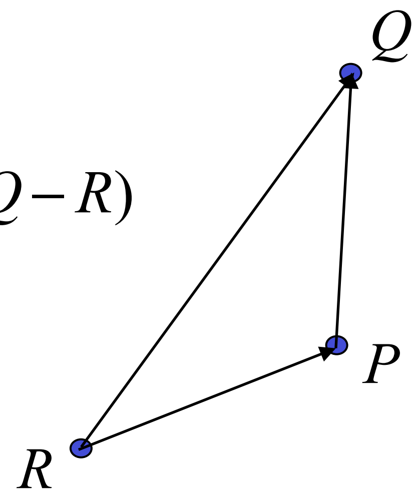
ie all vectors with the same magnitude and direction are identical

Affine space adds point primitives to a vector space

One new operation $P, Q \in \Re^N$

point-point subtraction: $\vec{v} = Q - P$

From vector addition: $(Q-P)+(P-R)=(Q-R)$



Affine Spaces II

Frames:

A frame consists of point P_0 and a vector basis $(\vec{v}_1, \vec{v}_2, \vec{v}_3, \ldots, \vec{v}_N)$

an arbitrary vector $\vec{v} = a_1\vec{v}_1 + a_2\vec{v}_2 + a_3\vec{v}_3 + \ldots. + a_N\vec{v}_N$

& an arbitrary point $P = P_0 + b_1\vec{v}_1 + b_2\vec{v}_2 + b_3\vec{v}_3 + \ldots. + b_N\vec{v}_N$

the point and vector are represented by scalars $[a_i]$ and $[b_i]$

P_0 is the origin of the frame

Frames allow us to switch between changes in coordinate system where the origin changes: object frames/camera frame

ie. object is represented in a local frame but its position is in the camera frame

Euclidean Spaces

Affine spaces have no concept of how far points are

Euclidean space E is a vector space with a metric for distance

Define **dot (inner) product**:

$$\boxed{\vec{u}.\vec{v} = \vec{u}^T\vec{v} = u_1v_1 + u_2v_2 + u_3v_3 + \ldots + u_Nv_N = \sum_{i=1}^{N} u_iv_i}$$

The magnitude of a vector is defined as:

$$\boxed{|\vec{u}| = \sqrt{\vec{u}.\vec{u}} = \sqrt{\sum_{i=1}^{N} u_i^2}}$$

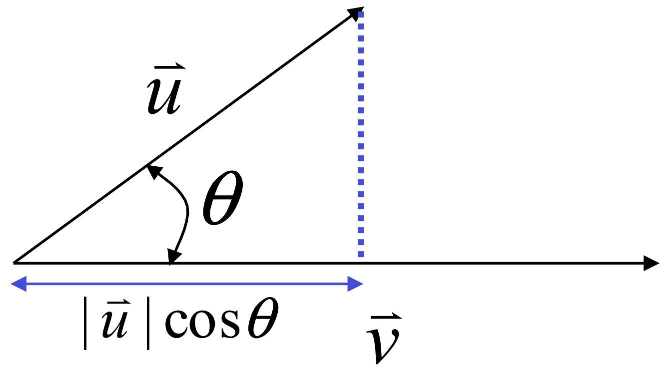
- square-root of the sum of square components

The distance between two points is defined as the magnitude of the vector between them

Dot (Inner) Product

Dot product is a scalar quantity:

$$\boxed{\vec{u}.\vec{v} = |\vec{u}||\vec{v}|\cos\theta \qquad \cos\theta = \frac{\vec{u}.\vec{v}}{|\vec{u}||\vec{v}|}}$$



Orthogonal projection of $\vec{u}$ onto $\vec{v}$ is:

$$\boxed{\frac{\vec{u}.\vec{v}}{|\vec{v}|} = |\vec{u}|\cos\theta}$$

Orthogonal vectors: $\vec{u}.\vec{v} = 0$

If 2 vectors are in the the same direction: $\vec{u}.\vec{v} > 0$

Properties:

Associative $\vec{u}.\vec{v} = \vec{v}.\vec{u}$

Distributive $(a\vec{u} + b\vec{v}).\vec{w} = a\vec{u}.\vec{w} + b\vec{v}.\vec{w}$

Gram-Schmidt Orthogonalisation

Given a set of basis vectors $u_1 \dots u_n$ create another basis $v_1 \dots v_n$ which is orthonormal: $\vec{v}_i . \vec{v}_j = 0$

Let $\vec{v}_1 = \dfrac{\vec{u}_1}{|\vec{u}_1|}$

find component of second basis vector orthogonal to $\vec{v}_1$

$$\vec{v}_2 = \vec{u}_2 + \alpha \vec{v}_1$$

$$\vec{v}_2 \cdot \vec{v}_1 = 0 = \vec{u}_2 \cdot \vec{v}_1 + \alpha \vec{v}_1 \cdot \vec{v}_1$$

$$\vec{v}_2 = \vec{u}_2 - \frac{\vec{u}_2 \cdot \vec{v}_1}{\vec{v}_1 \cdot \vec{v}_1} \vec{v}_1$$

For subsequent basis vectors find

$$\vec{v}_k = \vec{u}_k - \sum_{i=1}^{k-1} \frac{\vec{u}_k \cdot \vec{v}_i}{\vec{v}_i \cdot \vec{v}_i} \vec{v}_i$$

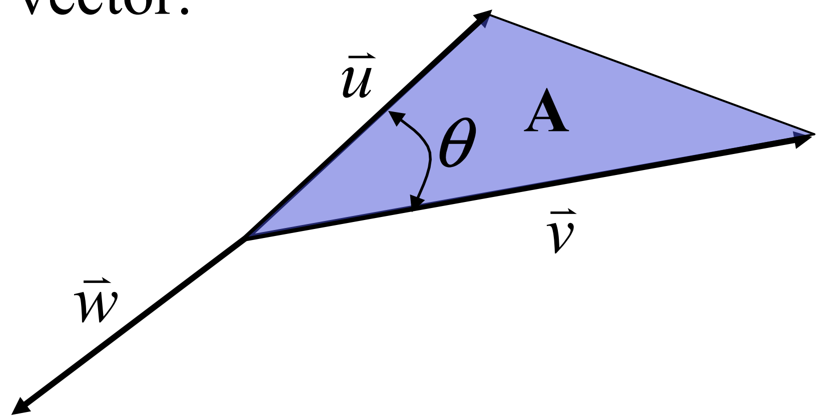
component of u orthogonal to all basis vectors v

Cross (Outer) Product

Cross-product of two linearly independent vectors produces a new orthogonal vector:

$$\boxed{\begin{array}{l}\vec{v}\times\vec{u}=\vec{w}\\ \vec{u}.\vec{w}=\vec{v}.\vec{w}=0\end{array}}$$

right-handed coordinate system



The cross-product in 3D space is defined as the vector:

$$\vec{w}=\vec{v}\times\vec{u}=[(v_2u_3-v_3u_2),(v_3u_1-v_1u_3),(v_1u_2-v_2u_1)]$$

$$\vec{v}=(v_1,v_2,v_3)\ \&\ \vec{u}=(u_1,u_2,u_3)$$

The magnitude of the cross-product is:

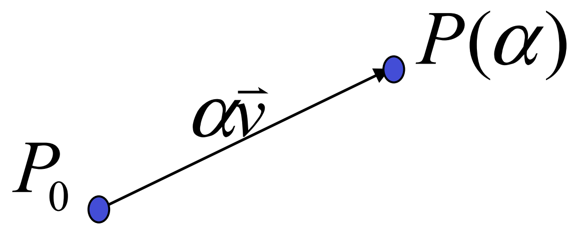
$$\boxed{|\vec{w}|=|\vec{v}\times\vec{u}|=|\vec{v}||\vec{u}||\sin\theta|=2A}$$

where A is the area of the triangle defined by vectors $\boxed{\vec{v},\vec{u}}$

Parametric Lines

We can define a line by an arbitrary point P_0 and vector $\vec{v}$ by the **parametric form**:

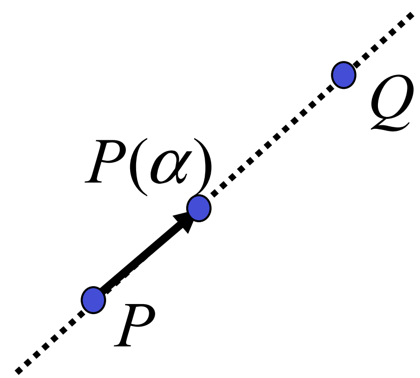
$$\boxed{P(\alpha) = P_0 + \alpha\vec{v}}$$



$P(\alpha)$ is a point on the line for any value of α

If $\vec{v} = P - Q$ then:

$$P(\alpha) = P + \alpha(Q - P) = \alpha Q + (1-\alpha)P$$



$P(\alpha)$ is an **Affine sum** of two points P,Q

$0 \le \alpha \le 1$ is a point on the line segment between P and Q

Convexity

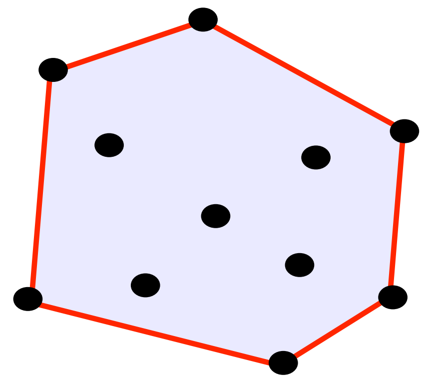
Definition: A convex object is one for which any point on the line segment connecting any 2 points in the object is inside the object

Affine sum can be used to represent all points inside a convex object, for an object defined by n-points:

$$P = \alpha_1 P_1 + \alpha_2 P_2 + \cdots + \alpha_n P_n = \sum_{i=1}^{n} \alpha_i P_i$$

$$0 \le \alpha_i \le 1$$

$$\sum_{i=1}^{n} \alpha_i = 1$$



The set of points formed by the affine sum of n-points is the **'convex hull'**

Convex hull is the smallest convex object which includes the set of all n points

Planes

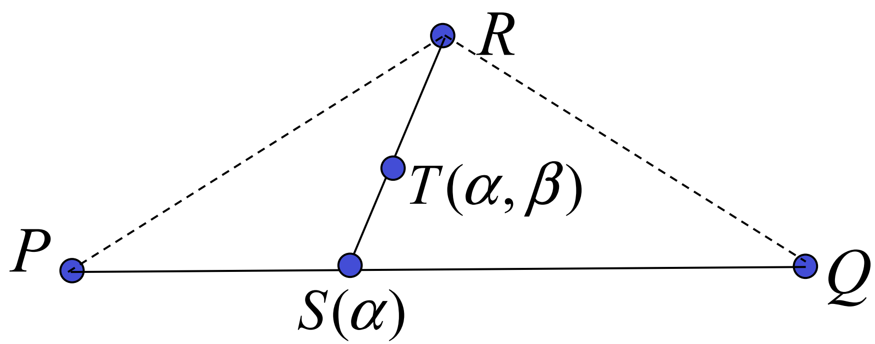
We can define a **parametric form** for a plane
from a set of 3-points P,Q,R which are not co-linear

$$S(\alpha) = \alpha P + (1-\alpha)Q = P + (1-\alpha)(Q-P)$$

is a point on the line from P to Q

$$\begin{aligned} T(\alpha,\beta) &= \beta S(\alpha) + (1-\beta)R \\ &= P + \beta(1-\alpha)(Q-P) + (1-\beta)(R-P) \\ &= P + a\vec{u} + b\vec{v} \end{aligned}$$

$0 \le a,b \le 1$ for all point inside the triangle (P,Q,R)



Three-Dimensional Coordinate Systems and Frames

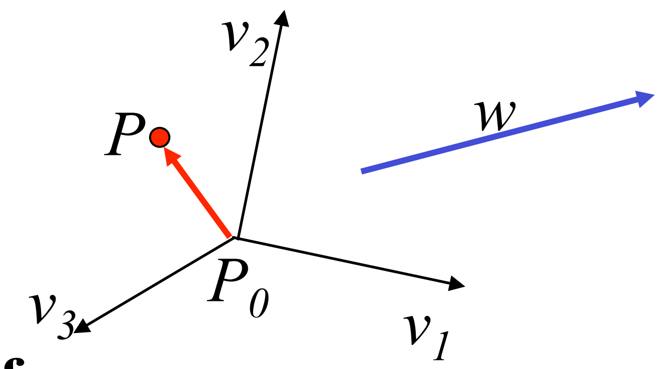
Have considered vectors and points abstract object without representing them in a specific coordinate system

Given a basis v_1, v_2, v_3 of linearly independent vectors we can represent any vector w with respect to this basis as:

$$w = a_1 v_1 + a_2 v_2 + a_3 v_3$$

Representation of w with respect to this basis is column matrix a:

$$a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \qquad w = a^T \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$



Vector basis v_1, v_2, v_3 *and* P_0 define a **frame**

P_0 is the origin

$$P = P_0 + b_1 v_1 + b_2 v_2 + b_3 v_3$$

Changes in Coordinate System

Change of basis vectors from v_1, v_2, v_3 to u_1, u_2, u_3: $\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = M \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$

M is a 3x3 matrix

The representation of a vector *a* in *v* is transformed to *b* in *u* as

$$a = M^T b$$

$$b = (M^T)^{-1} a = Aa$$

Matrix *A* transforms the representation of vector in *v* to its representation in *u*.
Remember a,b are representations with respect to a particular basis

Change of basis *A* leaves the origin at P_0

Homogeneous Coordinates

Use a four-dimensional column matrix to represent both points and vectors in 3-space.

A point in frame v_1, v_2, v_3, P_0 is defined in Homogeneous coordinates as:

$$P = \begin{bmatrix} a_1 & a_2 & a_3 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ P_0 \end{bmatrix}$$

This gives a Homogeneous-coordinate representation: $p = \begin{bmatrix} a_1 & a_2 & a_3 & 1 \end{bmatrix}$

A vector can be written as:

$$w = \begin{bmatrix} c_1 & c_2 & c_3 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ P_0 \end{bmatrix}$$

Giving Homogeneous-coordinate representation: $c = \begin{bmatrix} c_1 & c_2 & c_3 & 0 \end{bmatrix}$

Advantage: Points and vectors have different representation

Change of Frame in Homogeneous Coordinates

Change of frame from v_1, v_2, v_3, P_0 to u_1, u_2, u_3, Q_0 :

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ Q_0 \end{bmatrix} = M \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ P_0 \end{bmatrix}$$

$$M = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} & 0 \\ \gamma_{21} & \gamma_{22} & \gamma_{23} & 0 \\ \gamma_{31} & \gamma_{32} & \gamma_{33} & 0 \\ \gamma_{41} & \gamma_{42} & \gamma_{43} & 1 \end{bmatrix}$$

For a point/vector with represent a in v_1, v_2, v_3, P_0 we can transform to its representation b in frame u_1, u_2, u_3, Q_0

$$a = M^T b$$

This transforms both the basis and the origin of the frame

Advantage of Homogeneous Coordinates

1. Common/distinct representation of points and vectors

2. Transformation between frames by 4x4 matrix
   (change of basis and origin)

3. All affine (linear) transformations can be represented by
   a single matrix multiplication in Homogeneous coordinates
   (Rotation, Translation, Shear, Projection)

4. Successive transformations given by concatenation of multiple
   transformation matricies T=ABC

5. Computationally efficient - multiplication/addition operations

Used to represent all transformations in OpenGL

Conversion of Homogeneous to Euclidean coordinates

Homogeneous representation (x,y,z,w)

is equivalent to (x/w,y/w,z/w) in 3-space

Warning: w=0 ie vector is a equivalent to a point at infinity

Affine Transformations

Transformation: takes a point (or vector) and maps it to another point (or vector)

$$Q = T(P)$$
$$v = R(u)$$

In Homogeneous coordinates we can use the same function for points or vectors

$$p = f(q)$$
$$u = f(v)$$

f() is a single-valued function representing a general mapping

Linear or Affine Transformations:

for all scalars α, β

$$f(\alpha p + \beta q) = f(\alpha p) + f(\beta q)$$

ie Linear transform of 2 points or vectors is the same as the sum of the linear transforms applied to each point separately

Affine Transformation of Lines

Affine transformation of a line results in a new line

$$p(\alpha) = \alpha p_1 + (1-\alpha) p_2$$

$$Ap(\alpha) = A(\alpha p_1 + (1-\alpha) p_2)$$

$$= \alpha A p_1 + (1-\alpha) A p_2$$

transformed line is an affine combination of the two transformed points
ie under affine transformation straight lines are preserved.

Affine Transforms in Homogeneous Coordinates

For 4D Homogeneous coordinates all linear transforms can be represented as a matrix multiplication:

$$v = Au$$

A is a 4x4 matrix

Linear transform can be viewed as

(1) a change in frame

or (2) transformation of points within a frame

For Homogeneous coordinates A is given by:

$$A = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & \alpha_{14} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & \alpha_{24} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & \alpha_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

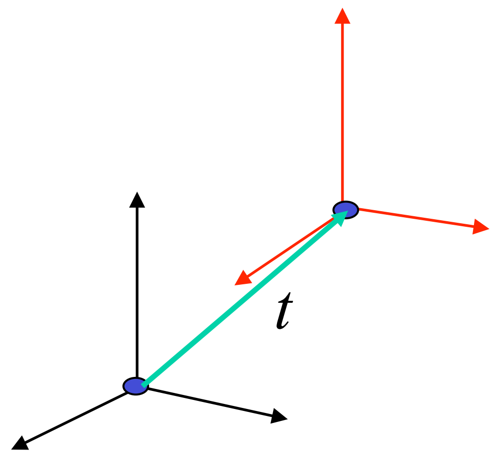
For a point $p=(x,y,z,1)$ the transformation has 12 degrees-of-freedom

for a vector $v=(u,v,w,0)$ the transformation has 9 degrees of freedom

Homogeneous Affine Transformations

Translation

$$A_t = \begin{bmatrix} I & t \\ \vec{0}^T & 1 \end{bmatrix}$$



I is a 3x3 identity matrix

$t=(t_1, t_2, t_3)$ is a 3x1 translation vector

$\vec{0}^T=(0,0,0)$ is a 3x1 zero vector

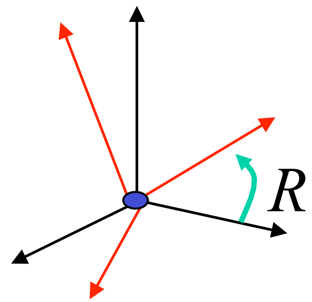
$$\begin{bmatrix} x' \\ 1 \end{bmatrix} = A_t \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} x+t \\ 1 \end{bmatrix}$$

Equivalently in Euclidean coordinates $x' = x + t$

Homogeneous Affine Transformations

Rotation

$$A_R = \begin{bmatrix} R & \vec{0} \\ \vec{0} & 1 \end{bmatrix}$$



R is a 3x3 rotation matrix

$0 = (000)^T$ is a 1x3 zero vector

$$\begin{bmatrix} x' \\ 1 \end{bmatrix} = A_R \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Rx \\ 1 \end{bmatrix}$$

Equivalently in Euclidean coordinates $x' = Rx$

Properties of Rotation matrix R

R is an orthonormal matrix

Columns of R are independent vectors $r_1 \cdot r_2 = r_1 \cdot r_3 = r_2 \cdot r_3 = 0$

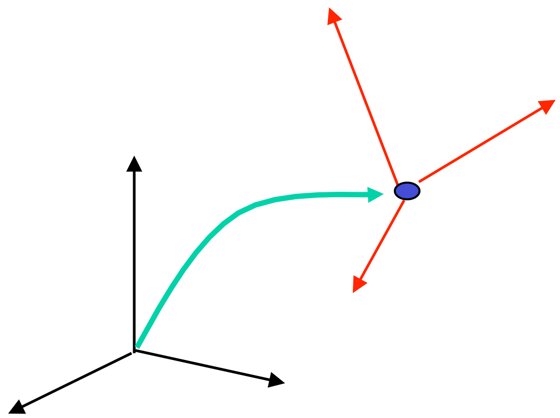
$$R^{-1} = R^T$$

$$RR^T = I$$

Homogeneous Affine Transformations

Composition of Rotation+Translation

$$A_{Rt} = \begin{bmatrix} I & t \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} R & \vec{0} \\ \vec{0}^T & 1 \end{bmatrix} = \begin{bmatrix} R & t \\ \vec{0}^T & 1 \end{bmatrix}$$



$$\begin{bmatrix} x' \\ 1 \end{bmatrix} = A_{Rt} \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Rx + t \\ 1 \end{bmatrix}$$

Equivalently in Euclidean coordinates $x' = Rx + t$

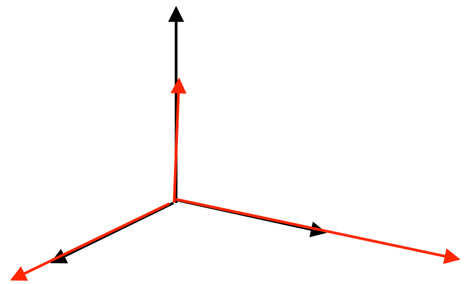
Rigid-body transform (no change in object shape/size)

Homogeneous Affine Transformations

Scale

- Non-rigid transformation

$$A_S = \begin{bmatrix} S & \vec{0} \\ \vec{0}^T & 1 \end{bmatrix}$$



$$S = \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & s_3 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ 1 \end{bmatrix} = A_S \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Sx \\ 1 \end{bmatrix}$$

Equivalently in Euclidean coordinates $x' = Sx$

Elementary transforms: Translation, Rotation, Scale

All other linear transformations formed
by composition of elementary transforms

Concatenation of Transforms

Given a point x we want to apply a series of transforms $T_1 \ldots T_n$
- order of composition is critical

$$
\begin{aligned}
T &= T_1 \cdots T_n \\
x' = Tx &= T_1 \cdots T_n x \\
&= T_1 \cdots T_{n-1}(T_n x) \\
&= T_1 \cdots T_{n-2}(T_{n-1}(T_n x))
\end{aligned}
$$

The last transformation T_n is applied first

Example compositon of rotation and translation transforms:

$$\begin{bmatrix} x' \\ 1 \end{bmatrix}^{tR} = A_t A_R \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} I & t \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} R & \vec{0} \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Rx + t \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ 1 \end{bmatrix}^{Rt} = A_R A_t \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} R & \vec{0} \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} I & t \\ \vec{0}^T & 1 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} R(x + t) \\ 1 \end{bmatrix}$$

Inverse Rigid-body Transform

$$\begin{bmatrix} x' \\ 1 \end{bmatrix} = A_{Rt} \begin{bmatrix} x \\ 1 \end{bmatrix} = \begin{bmatrix} Rx + t \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ 1 \end{bmatrix} = A_{Rt}^{-1} \begin{bmatrix} x' \\ 1 \end{bmatrix}$$

$$= A_{Rt}^{-1} \begin{bmatrix} Rx + t \\ 1 \end{bmatrix}$$

$$A_{Rt}^{-1} = \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix}$$

Rotation

(1) Euler Angles

3 parameters = 3 degrees of freedom

'Gymbal lock' when axis align dof are reduced ie 90degree rotation

$$R = R_z(\theta_z)R_y(\theta_y)R_x(\theta_x)$$

$$R_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix}$$

$$R_y(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix}$$

$$R_z(\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation

(2) Axis Angle

Represent rotation by angle θ about a unit axis w

Rotation = exponential map of w

3 degrees of freedom

Avoids gymbal lock

Singularity at $\theta = 0$

$$R = \exp(-W) = I + W + \frac{W^2}{2!} + \frac{W^3}{3!} \cdots$$

$$W = \theta \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix}$$

No 3-parameter representation of rotation can avoid singularities

(3) Quarternions

4 parameter representation $q = q_o + q_1 i + q_2 j + q_3 k = (q_o, v)$

uses complex basis *i,j,k* $i^2 = j^2 = k^2 = -1$

Rotation by θ *about w*

$$q_o = \cos\frac{\theta}{2} \qquad v = w\sin\frac{\theta}{2}$$

No singularities

Simple operations:

$$q_A + q_B = (q_{0A} + q_{0B}, v_A + v_B)$$
$$q_A q_B = (q_{0A} q_{0B} - v_A \cdot v_B, q_{0A} v_B + q_{0B} v_A + v_A \times v_B)$$
$$|q|^2 = q^2{}_0 + v \cdot v$$

Efficient composition of rotations

Summary - 3D Geometry

(1) Spaces - Vector/Affine/Euclidean
- operations on points/vectors

(2) Affine representation of points/lines
- convexity/convex hull

(3) Coordinate system and frames
- basis transformations

(4) Homogeneous Coordinates
- affine transformations rotation/translation/scale