The Group of Affine Maps $\mathcal{A}$

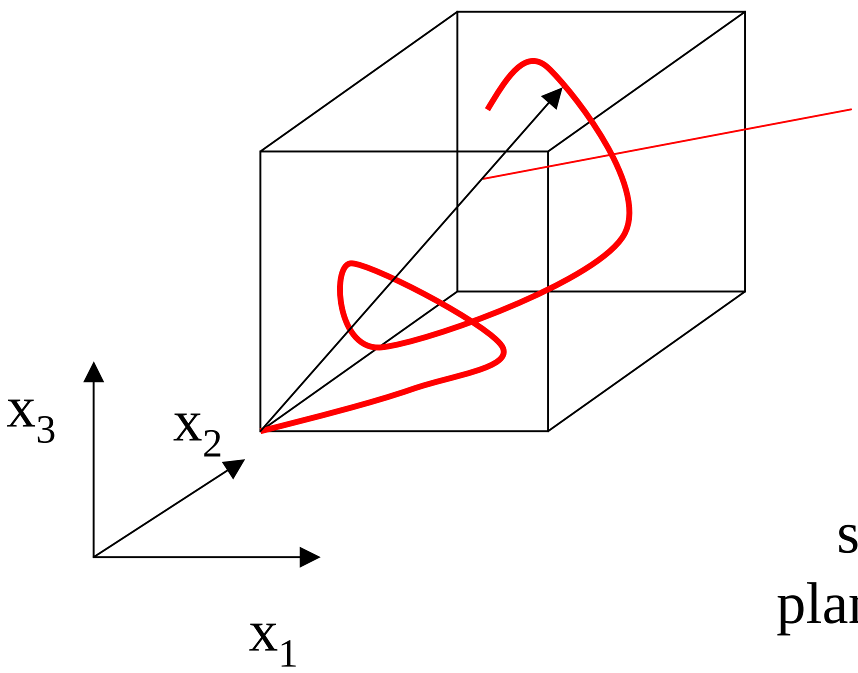
The group of affine maps results from choosing an arbitrary regular matrix **A** ($\det(\mathbf{A}) \neq 0$) and a translation **a**:

$$\mathbf{t}' = \mathbf{A}\mathbf{t} + \mathbf{a}$$

This map has 6 degrees of freedom (4 for the general matrix **A**, and two for the translation vector **a**).

The affine map describes the general steric motion of a planar master illustration followed by parallel projection on the camera plane, which is to be shown now.

Motion of an arbitrary curve in space with subsequent parallel projection on the camera plane



$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

special case:
plane steric curve: $x_3(t) \equiv 0$

The arc length calculates as follows:

$$s = \int_{t_0}^{t} \|\dot{\mathbf{x}}(t)\| \, dt = \int_{t_0}^{t} \sqrt{\dot{x}_1^2(t) + \dot{x}_2^2(t) + \dot{x}_3^2(t)} \, dt$$

with:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} dx_1/dt \\ dx_2/dt \\ dx_3/dt \end{bmatrix}$$

Euclidian Space Motion (Rotation and Translation):

$$\mathbf{x}'(t) = \mathbf{A}\,\mathbf{x}(t) + \mathbf{b}$$

With the orthogonal rotation matrix:

$$\mathbf{A} = \begin{bmatrix} A_{11} = c_2c_3 - c_1s_2s_3 & A_{12} = -\,c_2s_3 - c_1s_2c_3 & A_{13} = s_1s_2 \\ A_{21} = s_2c_3 + c_1c_2s_3 & A_{22} = -\,s_2s_3 + c_1c_2c_3 & A_{23} = -\,s_1c_2 \\ A_{31} = s_1s_3 & A_{32} = s_1c_3 & A_{33} = c_1 \end{bmatrix}$$

whereas:

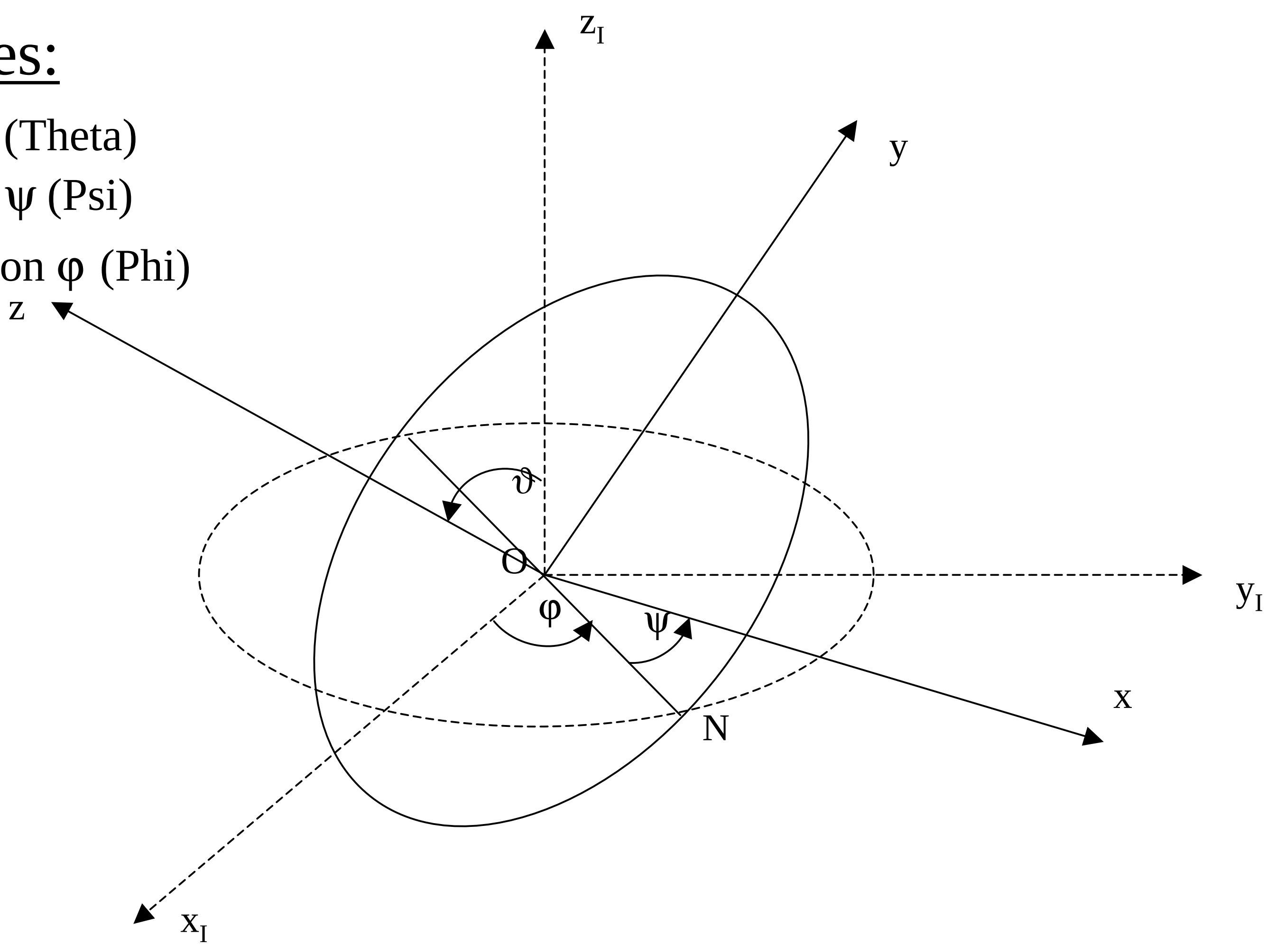
$$c_1 = \cos\vartheta, \quad c_2 = \cos\psi, \quad c_3 = \cos\varphi$$
$$s_1 = \sin\vartheta, \quad s_2 = \sin\psi, \quad s_3 = \sin\varphi$$

and the translation:

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Euler Angles:

1. nutation ϑ (Theta)
2. precession ψ (Psi)
3. mere rotation φ (Phi)



To exchange from the inertial system into the body coordinate system the following three rotations are needed:

1. rotation φ about the z_I - axis: the x-axis results/merges in the so-called line of nodes O-N.
2. rotation ϑ about the line of nodes O-N: the inertial z_I - axis merges in the body z-axis.
3. rotation ψ about the z - axis: The body (x,y,z)-coordinate system results.

Factorization of the Rotation Matrix:

$$\mathbf{A} = \underbrace{\begin{bmatrix} c_2 & -s_2 & 0 \\ s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation about z-axis } (\psi)} \cdot \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & c_1 & -s_1 \\ 0 & s_1 & c_1 \end{bmatrix}}_{\text{rotation about x-axis } (\vartheta)} \cdot \underbrace{\begin{bmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation about z-axis } (\varphi)}$$

Properties of the Rotation Matrix **A**:

- The row and column vectors of **A** are orthogonal and normalized and thus
- A rotation matrix **A** is orthogonal, i.e.:
  $\mathbf{A}^{-1}=\mathbf{A}^{T}$ and thus also $\mathbf{A}^{T}\mathbf{A}=\mathbf{A}\mathbf{A}^{T}=\mathbf{I}$
- Furthermore: det(**A**)=1 (mere rotation, without reflection)
- Thus, for rotation matrices applies:

$$(\mathbf{ABC})^{-1}=(\mathbf{ABC})^{T}=\mathbf{C}^{T}\mathbf{B}^{T}\mathbf{A}^{T}$$

And thus:

$$\mathbf{A}^{-1} = \mathbf{A}^T = \begin{bmatrix} c_3 & s_3 & 0 \\ -s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_1 & s_1 \\ 0 & -s_1 & c_1 \end{bmatrix} \cdot \begin{bmatrix} c_2 & s_2 & 0 \\ -s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation backwards can be achieved by inversion of the order!

A more compact notation:

$$\mathbf{A} = \begin{bmatrix} \mathbf{R}_2 & & 0 \\ & & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & & \\ 0 & & \mathbf{R}_1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_3 & & 0 \\ & & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and: $\mathbf{R}_i^{-1} = \mathbf{R}_i^T$

Now: orthogonal projection of the sterical curve $\mathbf{x}$(t) into the ($\mathbf{e}_1$,$\mathbf{e}_2$)-plane with the projection operator $\mathbf{P}_{12}$:

$$\mathbf{x}_{12} = \begin{bmatrix} x_1(t) \\ x_2(t) \\ 0 \end{bmatrix} = \mathbf{P}_{12}\mathbf{x} \qquad \text{with:} \quad \mathbf{P}_{12} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The projection operator $\mathbf{P}_{12}$ is idempotent, i.e.:

$$\mathbf{P}_{12}^2 = \mathbf{P}_{12}$$

From the general motion in space with subsequent projection follows:

$$\mathbf{x}'' = \mathbf{P}_{12}\mathbf{x}' = \mathbf{P}_{12}(\mathbf{A}\mathbf{x} + \mathbf{b})$$

The projection has the following effect on **A**:

$$\mathbf{P}_{12}\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Since the curve is planar ($x_3 \equiv 0$). the 3rd column can be omitted:

$$\begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

And so the description can be reduced to 2 dimensions:

$$\mathbf{x}' = \mathbf{A}'\mathbf{x}'^{0} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1^0 \\ x_2^0 \end{bmatrix}$$

with:

$$\mathbf{A}' = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \underbrace{\begin{bmatrix} c_2 & -s_2 \\ s_2 & c_2 \end{bmatrix}}_{\text{rotate }(\psi)} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & c_1 \end{bmatrix}}_{\text{compress along y}} \underbrace{\begin{bmatrix} c_3 & -s_3 \\ s_3 & c_3 \end{bmatrix}}_{\text{rotate }(\varphi)} = \mathbf{R}_2 \begin{bmatrix} 1 & 0 \\ 0 & c_1 \end{bmatrix} \mathbf{R}_3$$

3 degrees of freedom : c_1, c_2, c_3 **resp.** ψ, c_1, φ

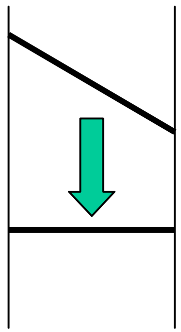
or expanded:

$$\mathbf{A}' = \begin{bmatrix} c_2c_3 - c_1s_2s_3 & -c_2s_3 - c_1s_2c_3 \\ s_2c_3 + c_1c_2s_3 & -s_2s_3 + c_1c_2c_3 \end{bmatrix}$$

Rotating a plane curve ($x_3 \equiv 0$) in a 3D space and projecting it afterwards in a ($\mathbf{e}_1$,$\mathbf{e}_2$)-plane is equivalent to describing it with a corresponding two-dimensional affine map!

Discussing the Degrees of Freedom:

- We see only three degrees of freedom c_1,c_2,c_3, but the general affine map has 4!
- As described here, the objects are only compressed and not enlarged. Adding a general dilation (enlarging/compressing) with a factor k, 4 degrees of freedom can be obtained!



$$\underbrace{\begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}}_{\text{general compression}} = \underbrace{\delta_1}_{\text{dilation}} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & \delta_2 / \delta_1 \end{bmatrix}}_{\text{compressing along y}}$$

Invariants of geometric maps:

Congruent maps ($\mathbf{A}\mathbf{A}^T=\mathrm{I}$) are ***length preserving***:

$$\|\mathbf{x}'\|^2 = <\mathbf{x}', \mathbf{x}'> = <\mathbf{A}\mathbf{x}, \mathbf{A}\mathbf{x}> = <\mathbf{x}, \mathbf{A}^*\mathbf{A}\mathbf{x}> \overset{!}{=} <\mathbf{x}, \mathbf{x}> = \|\mathbf{x}\|^2$$

only applies for: $\mathbf{A}^*\mathbf{A} = \mathbf{I} \Rightarrow \mathbf{A}^{-1} = \mathbf{A}^*$ (orthogonal rotation matrix)

The group of similarities guarantees angle preserving (conform) maps:

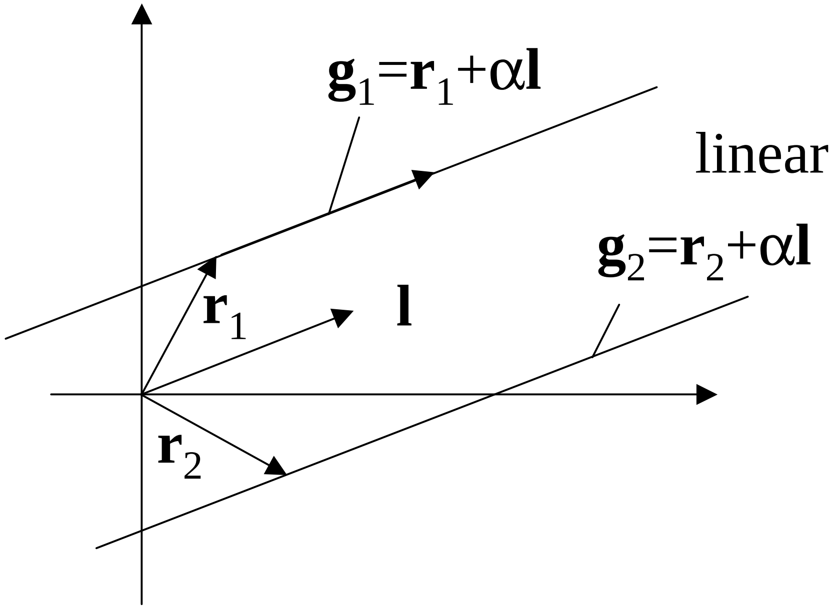
$$\cos\varphi = \frac{<\mathbf{x}',\mathbf{y}'>}{\|\mathbf{x}'\|\|\mathbf{y}'\|} = \frac{<\mathbf{Ax},\mathbf{Ay}>}{<\mathbf{Ax},\mathbf{Ax}>^{1/2}<\mathbf{Ay},\mathbf{Ay}>^{1/2}}$$

with: $\mathbf{A}^*\mathbf{A} = \mu\mathbf{I}$ follows:

$$\cos\varphi = \frac{<\mathbf{x},\mathbf{A}^*\mathbf{Ay}>}{<\mathbf{x},\mathbf{A}^*\mathbf{Ax}>^{1/2}<\mathbf{y},\mathbf{A}^*\mathbf{Ay}>^{1/2}} =$$

$$\frac{\mu<\mathbf{x},\mathbf{y}>}{\mu^{1/2}<\mathbf{x},\mathbf{x}>^{1/2}\ \mu^{1/2}<\mathbf{y},\mathbf{y}>^{1/2}} = \frac{<\mathbf{x},\mathbf{y}>}{\|\mathbf{x}\|\|\mathbf{y}\|}$$

The group of affine maps preserves ***parallelisms***:



linear equations

for all pairs of points on straight lines parallel to **l** applies: $\mathbf{x}_2 - \mathbf{x}_1 = \alpha\mathbf{l}$

affine transformed: $\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{t}$ and thus:

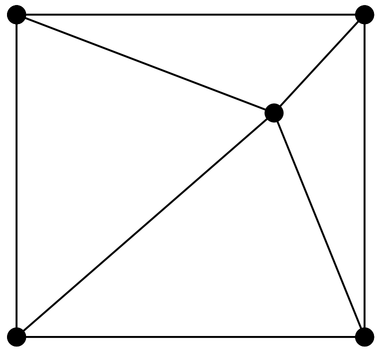
$$\mathbf{x}_2' - \mathbf{x}_1' = \mathbf{A}\mathbf{x}_2 + \mathbf{t} - \mathbf{A}\mathbf{x}_1 - \mathbf{t} = \mathbf{A}(\mathbf{x}_2 - \mathbf{x}_1) = \alpha\mathbf{A}\mathbf{l} = \alpha\mathbf{l}'$$

that are points on the parallel straight line with gradient **l´**

Interpolation

For rotation and translation about fractions of the sampling interval a grayscale interpolation of the images is required (non gridconforming maps).

In the easiest way this can be realized by
a) the Nearest-Neighbour-Rule:



Or better: with bilinear interpolation:

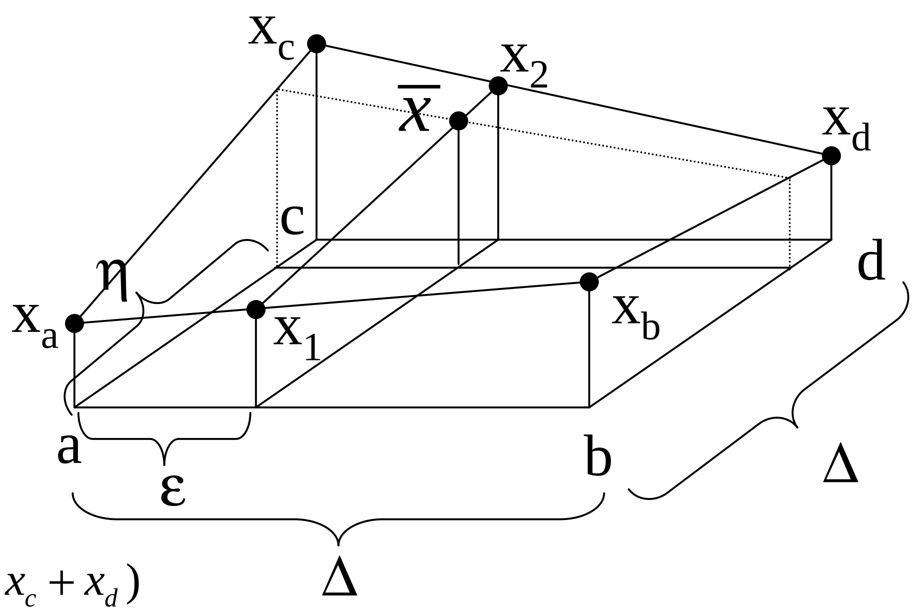
$$x_1 = x_a + \frac{\varepsilon}{\Delta}(x_b - x_a) = \overline{x}(x_a, x_b, \varepsilon)$$

$$x_2 = x_c + \frac{\varepsilon}{\Delta}(x_d - x_c) = \overline{x}(x_c, x_d, \varepsilon)$$

and so:

$$\overline{x} = x_1 + \frac{\eta}{\Delta}(x_2 - x_1) = \overline{x}(x_1, x_2, \eta)$$

$$= x_a + \frac{\varepsilon}{\Delta}(x_b - x_a) + \frac{\eta}{\Delta}(x_c - x_a) + \frac{\varepsilon\eta}{\Delta^2}(x_a - x_b - x_c + x_d)$$



By exchanging x_a for x_c and ε for η, one gets the same result, i.e. one could interpolate first with (x_a, x_a, η) and (x_b, x_d, η) and then interpolate the result linearly.