Computer Graphics

Ágoston Sipos
siposagoston@inf.elte.hu

Eötvös Loránd University
Faculty of Informatics

2025-2026. Fall semester

Table of Contents

Motivation

- The complex geometric entities of our scenes (house, tree, etc.) are made of smaller elements (door, window, leaves...) $\rightarrow$ the parts must be placed in space
- We have to place the shapes in the world, move them, etc.
- We also need to convert our virtual world into a two-dimensional image
- $\rightarrow$ For all the steps above we will need *geometric transformations* to change our shapes

Transformations

- Our expectations are that transformations
  - are defined for all points
  - map a point to a point, a line to a line, a plane to a plane
  - preserve incidence relation
  - should be unique and reversible

Remark

- We store our point in some coordinate system $\rightarrow$ transformations are operation on these coordinates
- From now on, let us associate the points of Euclidean space $\mathbb{E}^3$ (or plane $\mathbb{E}^2$) with the vectors of $\mathbb{R}^3$ (or $\mathbb{R}^2$)
- For this we set a point $\mathbf{o} \in \mathbb{E}^3$, the origin and for every point $\mathbf{q} \in \mathbb{E}^3$, we assign a (position) vector $\mathbf{p} = \mathbf{q} - \mathbf{o}$

Linear mapping

- *Linear mappings* are ϕ mappings for which $\forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ and $\lambda \in \mathbb{R}$
  - $\phi(\mathbf{a} + \mathbf{b}) = \phi(\mathbf{a}) + \phi(\mathbf{b})$ (additive)
  - $\phi(\lambda \mathbf{a}) = \lambda \phi(\mathbf{a})$ (homogeneous)
- Reminder: linear mappings $f : \mathbb{R}^n \to \mathbb{R}^m$ can be represented with an $\mathbf{A} \in \mathbb{R}^{m \times n}$ matrix; $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$, $\mathbf{x} \in \mathbb{R}^n$.

Projective and affine transformations – definitions

- In the Euclidean space extended with an ideal plane, the mappings that are bijective, preserve points, lines, planes, and incidences are called *collineations* or *projective transformations*.
- Affine transformations are a subset of projective transformations that map the "ordinary" Euclidean space onto itself, and also map the ideal plane onto itself

Properties

- Projective and affine transformations form an algebraic group with the operation of concatenation (composition of transformations) $\rightarrow$ what does this mean?
  - concatenation is associative (the operations can be grouped)
  - there exists an identity element (identity transformation)
  - if the transformation preserves the dimension, then it has an inverse (can be reversed)
- Attention: this group is not commutative! I.e. the order of operands matters.

Properties of affine transformations

- Every affine transformation can be written as a linear transformation followed by a translation, that is if $\varphi : \mathbb{R}^3 \to \mathbb{R}^3$ is an affine transformation, then there is an $\mathbf{A} \in \mathbb{R}^{3\times3}$ and $\mathbf{b} \in \mathbb{R}^3$, for $\forall \mathbf{x} \in \mathbb{R}^3$

$$\varphi(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$$

- The matrix-vector multiplication is performed in this order: the matrix is on the left, the vector is on the right

Properties of affine transformations

- $\varphi(\mathbf{x}) = \mathbf{Ax} + \mathbf{b}$ with homogeneous coordinates can be written with only one matrix-vector multiplication:

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ [0,0,0] & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4}$$

- Because in this case

$$\begin{bmatrix} \mathbf{A} & \mathbf{b} \\ [0,0,0] & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A}\cdot\mathbf{x} + \mathbf{b}\cdot 1 \\ \mathbf{0}\cdot\mathbf{x} + 1\cdot 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A}\cdot\mathbf{x} + \mathbf{b} \\ 1 \end{bmatrix}$$

Properties of affine transformations

- Barycentric coordinates are not affected by affine transformations (*barycentric coordinates are invariant under affine transformations*)
- Proof: α_i be the barycentric coordinates of $\mathbf{x}$ wrt. $\mathbf{x}_i$, then

$$
\begin{aligned}
\varphi(\mathbf{x}) &= \varphi\left(\sum_{i=0}^{n} \alpha_i \mathbf{x}_i\right) \\
&= \mathbf{A}\left(\sum_{i=0}^{n} \alpha_i \mathbf{x}_i\right) + \mathbf{b} \\
&= \mathbf{A}\sum_{i=0}^{n} \alpha_i \mathbf{x}_i + \sum_{i=0}^{n} \alpha_i \mathbf{b} \\
&= \sum_{i=0}^{n} \alpha_i (\mathbf{A}\mathbf{x}_i + \mathbf{b}) = \sum_{i=0}^{n} \alpha_i \varphi(\mathbf{x}_i)
\end{aligned}
$$

Specifying affine transformations

- An affine transformation in $\mathbb{E}^n$ is uniquely defined with $n+1$ affinely independent points and their image
- That is, for example, in a plane if there are three points

$$\mathbf{p}_i = \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix},\ i = 0, 1, 2$$

point and their images, in order

$$\mathbf{q}_i = \begin{bmatrix} x_i' \\ y_i' \\ 1 \end{bmatrix},\ i = 0, 1, 2$$

then for $\mathbf{R} \in \mathbb{R}^{3\times 3}$ transformation, transforming $\mathbf{p}_i$ into $\mathbf{q}_i$

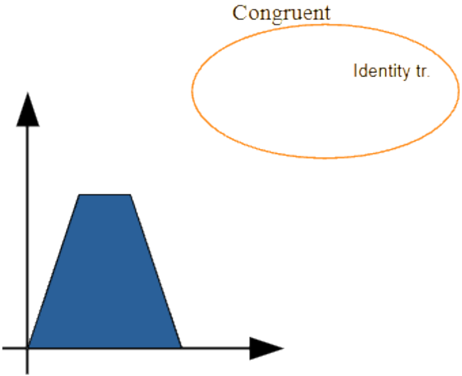
$$\mathbf{R} \cdot [\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2] = [\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2] \Rightarrow \mathbf{R} = [\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2] \cdot [\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2]^{-1}$$

Specifying affine transformations

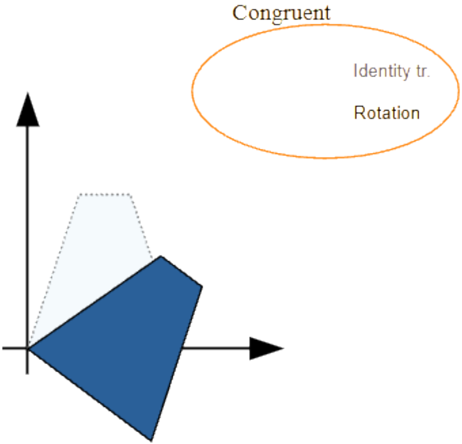
- Projective transformation in $\mathbb{E}^n$ is uniquely defined with $n+2$ affinely independent points and their image
- Then in a plane we need 4: let $\mathbf{P} \in \mathbb{R}^{3\times 3}$ and $\alpha_0, \alpha_1, \alpha_2, \alpha_3 \in \mathbb{R}$, then solve for $\mathbf{P}$

$$\mathbf{P} \cdot [\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3] = [\alpha_0 \mathbf{q}_0, \alpha_1 \mathbf{q}_1, \alpha_2 \mathbf{q}_2, \alpha_3 \mathbf{q}_3]$$

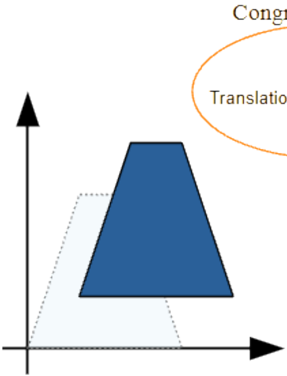
Classification



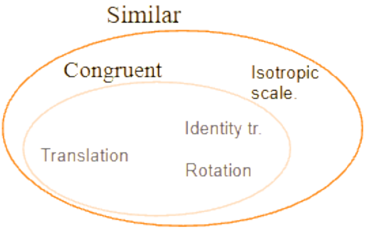
Classification

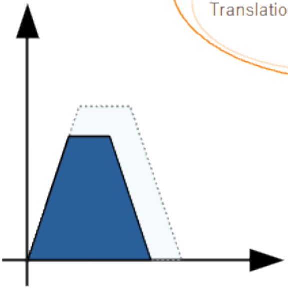


Classification

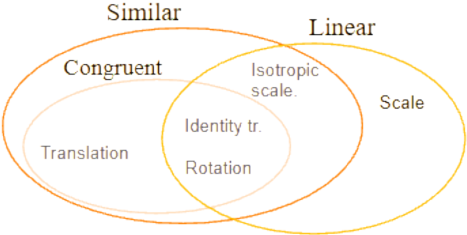


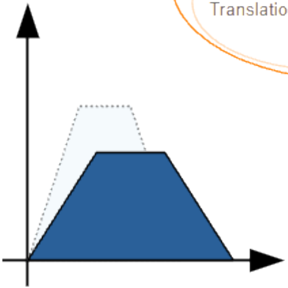
Classification



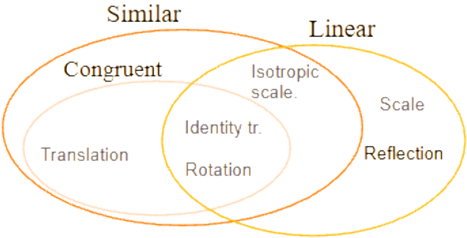


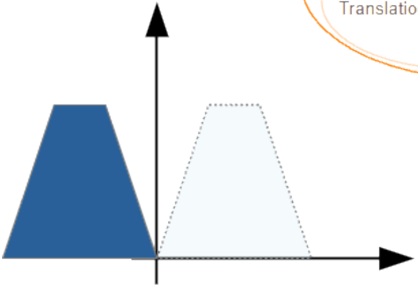
Classification



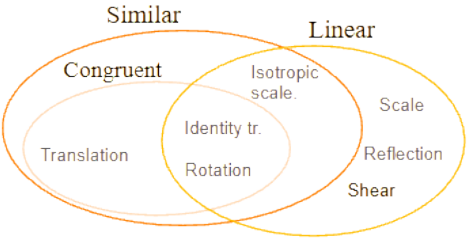


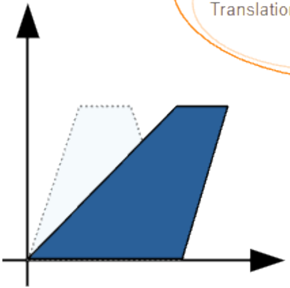
Classification



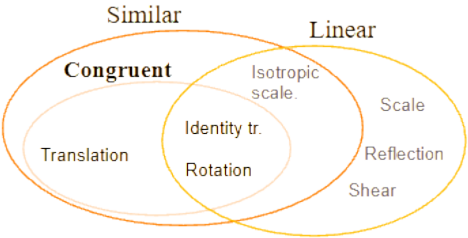


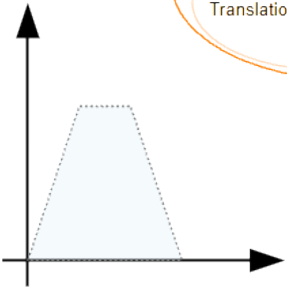
Classification



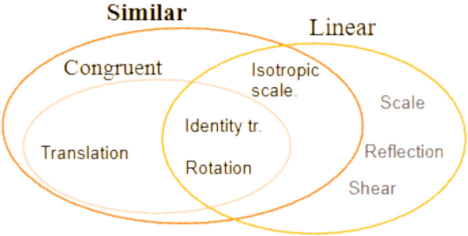


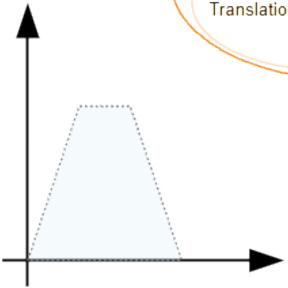
Classification



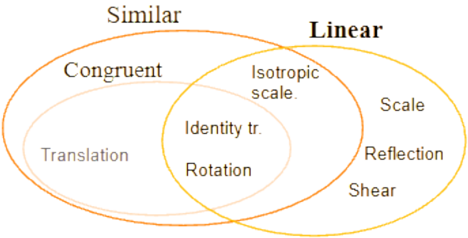


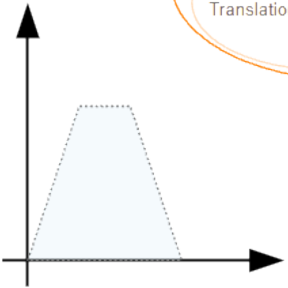
Classification



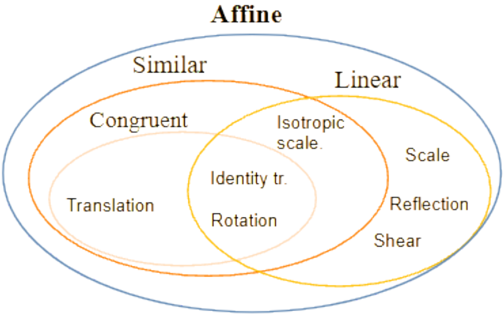


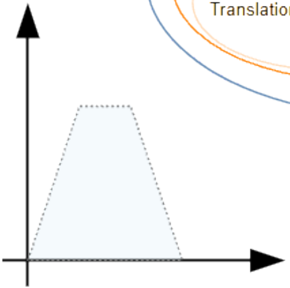
Classification



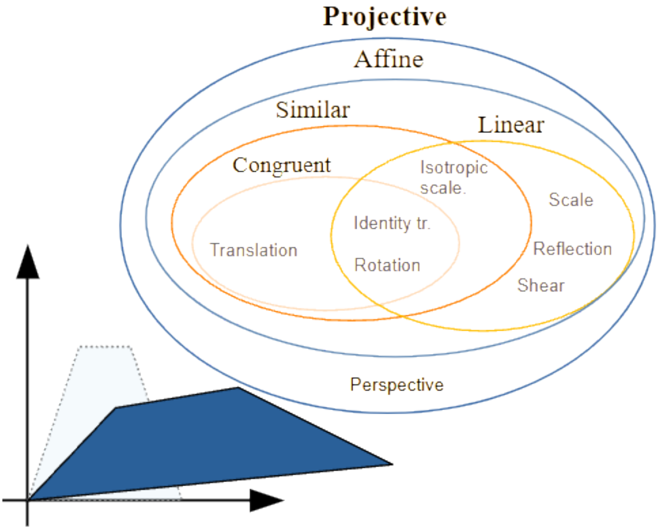


Classification

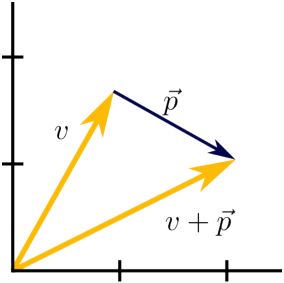




Classification



Translation



Translation

- We shift every point with a **d** vector:
$$\mathbf{x}' = \mathbf{x} + \mathbf{d}$$
- Usually denoted by $\mathbf{T}(d_x, d_y, d_z)$
- For the matrix form we need homogeneous coordinates, with $w = 1$, then it can be written as a 4×4 matrix:

$$\begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Translation

- After all, if we use the homogeneous coordinates of the point **x**, then

$$\begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \cdot x + 0 \cdot y + 0 \cdot z + 1 \cdot d_x \\ 0 \cdot x + 1 \cdot y + 0 \cdot z + 1 \cdot d_y \\ 0 \cdot x + 0 \cdot y + 1 \cdot z + 1 \cdot d_z \\ 1 \end{bmatrix}$$

Properties

- Translations are a commutative subset of the affine transformations
- The inverse of $\mathbf{T}(a, b, c)$ is $\mathbf{T}^{-1}(a, b, c) = \mathbf{T}(-a, -b, -c)$

Rotation

- Rotating in XY plane (in practice around Z axis) θ degrees:

$$
\begin{aligned}
x' &= x\cos\theta - y\sin\theta \\
y' &= x\sin\theta + y\cos\theta.
\end{aligned}
$$

- Matrix form:

$$
\begin{bmatrix} x' \\ y' \end{bmatrix} = x\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} + y\begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}
$$

- Similar on XZ and YZ plane.

Rotation matrices

Around Z axis, Around Y axis, Around X axis

$$\mathbf{R}_Z = \begin{bmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{R}_Y = \begin{bmatrix} c & 0 & s & 0 \\ 0 & 1 & 0 & 0 \\ -s & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{R}_X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c & -s & 0 \\ 0 & s & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $c = \cos\theta$ and $s = \sin\theta$.

Properties

- Rotations around the same axis are a commutative subset of the affine transformations
- Rotations in space can be written as a 3×3 matrix (linear transformation)
- Translation and rotation are not commutative!
- The inverse of the rotation is a rotation with equal magnitude, but in the opposite direction, e.g.: $\mathbf{R}_Z^{-1}(\theta) = \mathbf{R}_Z(-\theta)$

Arbitrary rotation

Any orientation can be produced by successively using the three rotations.

$$\mathbf{R}(\alpha, \beta, \gamma) =$$

$$\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}$$

Rotating around arbitrary axis

- Using what we already know:
  - shift our rotation axis into the origin ($\mathbf{T}$)
  - rotate it around an axis into the plane of the other two (e.g around $\mathbf{R}_Z$)
  - in this plane, rotate it with one the two axes into the other (e.g $\mathbf{R}_Y$)
  - do the rotation (e.g with $\mathbf{R}_X$, but: around the new (X") axis!)
  - we apply the inverses of the previous transformations
- For example $\mathbf{Mx} = (\mathbf{T}^{-1}\mathbf{R}_Z^{-1}\mathbf{R}_Y^{-1}\mathbf{R}_X\mathbf{R}_Y\mathbf{R}_Z\mathbf{T})\mathbf{x}$

*Rotating around arbitrary axis – Rodrigues formula

A rotation around an arbitrary *axis* can be given by a unit vector $\mathbf{z}$, which gives the axis of the rotation, and an angle θ.
This is described by the *Rodrigues formula*, using which:

$$\mathbf{v}' = \text{Rodrigues}(\theta, \mathbf{z})\mathbf{v}$$

$$\mathbf{v}' = \mathbf{v} \cdot \cos\theta + (\mathbf{z} \times \mathbf{v}) \cdot \sin\theta + \mathbf{z} \cdot \langle \mathbf{z}, \mathbf{v} \rangle \cdot (1 - \cos\theta)$$

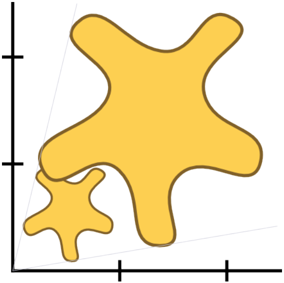
Yaw, pitch, roll

- The vertical (*yaw*), lateral (transverse) (*pitch*) and longitudinal (*roll*) rotations of an object are given at the same time.
- A commonly used method in aeronautics and robotics.
- It is practically the same as if we were rotating around three "ordinary" axis.
- It only works correctly if the axes of the object coincide with the axes of the coordinate system.
- Most API supports it.

Rigid body transformations

- Transformations that can be described as a combination of translations and rotations around an axis are called *rigid body transformations*
- The shape and size of objects are not changed

Scaling



Scaling

- Along the x, y, z axes, we "pull apart" or "compress" the shape, that is, we choose a different *scale* – even independently
- Matrix form:

$$\mathbf{S}(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- If $s_x = s_y = s_z$ it is an isotropic scaling

Special case: reflection

- If any of s_x, s_y, s_z is negative
  - if one is negative: reflection to the corresponding plane
  - if two are negative: reflection to the remaining axis
  - if all three are negative: central reflection
- Pay attention: if there is an odd number of negative coefficients, the winding also changes!

Winding?

- Using the basis vectors $\mathbf{i}, \mathbf{j}, \mathbf{k}$, if $\varphi : \mathbb{R}^3 \to \mathbb{R}^3$ is a linear transformation, then
$$\varphi(\mathbf{p}) = \varphi(x\mathbf{i} + y\mathbf{j} + z\mathbf{k}) = x\varphi(\mathbf{i}) + y\varphi(\mathbf{j}) + z\varphi(\mathbf{k})$$
- $\to$ if the determinant of a transformation matrix is negative, the winding (their direction in space) changes

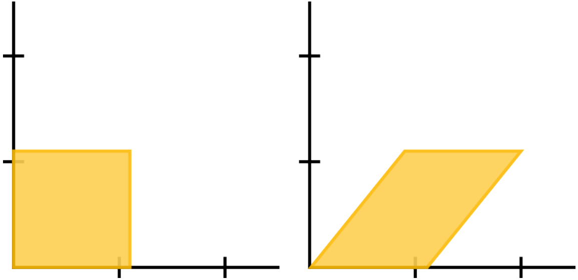
Special case: projection

- If any of s_x, s_y, s_z is zero
  - if one is zero: we project onto a plane perpendicular to the direction
  - if two are zero: we "project" onto an axis
  - if all three are zero: we project everything into the origin...
- Remark: determinant is zero! $\rightarrow$ there is no inverse!

Shearing

Example

Consider a deck of cards



Shearing

If, for example, we change the x, y values in each point proportionally to z:

$$\mathbf{N} = \begin{bmatrix} 1 & 0 & a & 0 \\ 0 & 1 & b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Shearing

In general:

$$\mathbf{N} = \begin{bmatrix} 1 & a & b & 0 \\ 0 & 1 & c & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Change of basis

- Let's assume that instead of orthonormal basis vectors of $\mathbf{i}, \mathbf{j}, \mathbf{k}$ we want to switch to $\mathbf{u}, \mathbf{v}, \mathbf{w}$ an orthonormal basis (the coordinates of the new basis vectors are known in the old basis).
- What will be the new $\mathbf{x}' = [x', y', z']^T$ coordinates (in the new base) of the point $\mathbf{x} = [x, y, z]^T$ (in the old base) That is, for what x', y', z' is it true that $\mathbf{x} = x'\mathbf{u} + y'\mathbf{v} + z'\mathbf{w}$?
- $\mathbf{x} = [\mathbf{u}, \mathbf{v}, \mathbf{w}]\mathbf{x}' = \mathbf{B}\mathbf{x}' \rightarrow \mathbf{x}' = \mathbf{B}^{-1}\mathbf{x}$
- The inverse of an orthonormal matrix is the transpose of the matrix, so our matrix $\mathbf{M} = \mathbf{B}^{-1}$ giving the new coordinates has the following form

$$\mathbf{M} = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

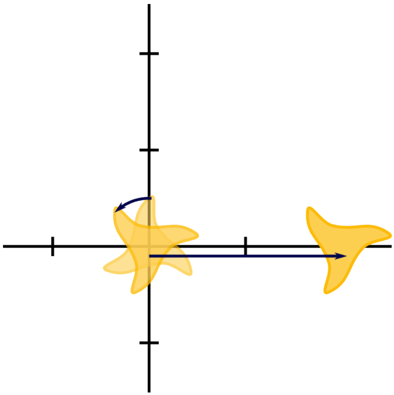
- If the coordinates of the new origin are $\mathbf{c}$, then $\mathbf{M} = \mathbf{B}^{-1}\mathbf{T}(-c_x, -c_y, -c_z)$

Commutativity

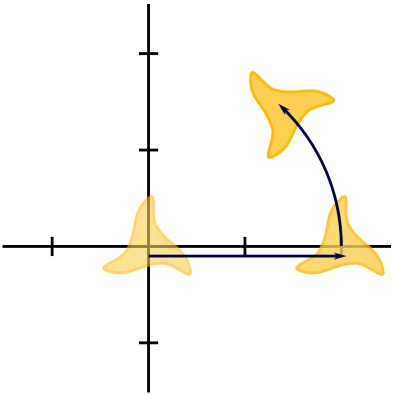
- Matrix multiplication is not commutative, so in general it is not true that
$$\mathbf{ABv} = \mathbf{BAv}$$
- This is good, since in general transformations are not commutative either

Example

Rotation then translation



Translation then rotation

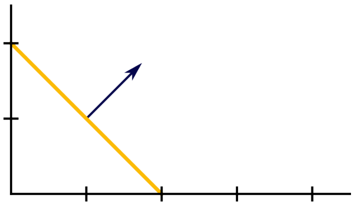


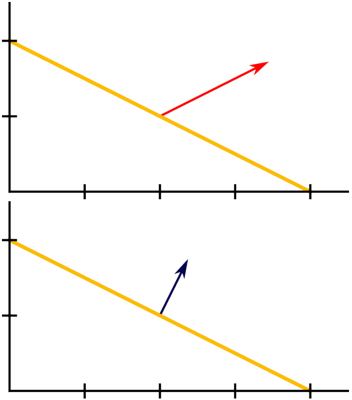
Determinants of transformation matrices

- When scaling, we saw that if one or three coefficients of the transformation are negative, it changes the winding direction.
- In general:
- If $\det(\mathbf{A}) > 0$, then the winding is unchanged
- If $\det(\mathbf{A}) < 0$, then the winding is reversed

Transformation of normals

- Let g a segment in plane, with $\mathbf{n}$ normal vector. Let $\mathbf{S}$ be a transformation describing a $2\times$ stretch along the x axis.
- Problem: g' can be obtained by transforming its two endpoints. What about the normal vector of g'? Will it be $\mathbf{n}' = \mathbf{Sn}$? NO!





Transformation of normals

- Let's examine the equation of the tangent plane given by the normal vector!
- Let $\mathbf{p}$ be a point on the tangent plane, then $\mathbf{x}$ is on the plane if and only if
$$\langle \mathbf{x} - \mathbf{p}, \mathbf{n} \rangle = 0$$
- Then with any arbitrary (invertible) $\mathbf{A}$ transformation:
$$\langle \mathbf{A}^{-1}\mathbf{A}(\mathbf{x} - \mathbf{p}), \mathbf{n} \rangle = 0$$
- Based on the rules of scalar product and matrix multiplication, we get that
$$\langle \mathbf{A}(\mathbf{x} - \mathbf{p}), \left(\mathbf{A}^{-1}\right)^T \mathbf{n} \rangle = 0$$
- That is, instead of the $\mathbf{A}$ matrix, the normal vectors must be multiplied by its inverse transpose!

Remark

- The affine transformations of the plane are uniquely defined by three independent points and their images
- The affine transformations of space are uniquely defined by four independent points and their images

Motivation

- We want to produce an image of our scene: project it onto a plane
- The image seen by humans cannot be reproduced using affine transformations. The parallel lines „moving away" from the observer appear to join
- This view can be reproduced with *central projection*. This transformation is linear in *homogeneous space*.
- The affine transformations did not „hurt" the ideal elements, but for the above, this is „necessary"

General case

If the last row of a *homogeneous* transformation matrix is not $[0, 0, 0, 1]$, then it is a *homogeneous linear transformation* that is a nonlinear transformation of the Euclidean space.

Parallel projection

- The matrix describing it is simple, for example the projection onto the XY plane

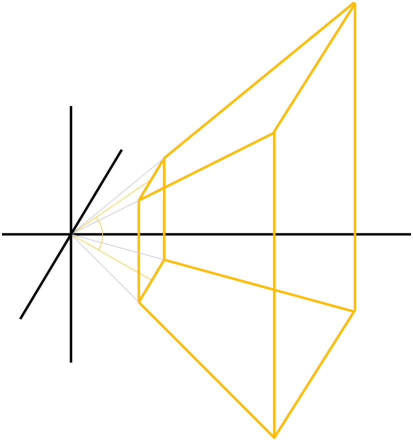
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- This is still an affine transformation (scaling)

Perspective transformation

- It implements a central projection.
- We "look" at the space from the origin along the z axis.
- A frustum corresponds to the visible space.
- The transformation makes parallel lines out of the projection lines that meet at the eye position.
- Transforms the frustum into a rectangular cuboid
- Its parameters:
  - vertical angle of the frustum,
  - the ratio of the sides of the base,
  - distance of the near clipping plane,
  - distance of the far clipping plane

Perspective transformation



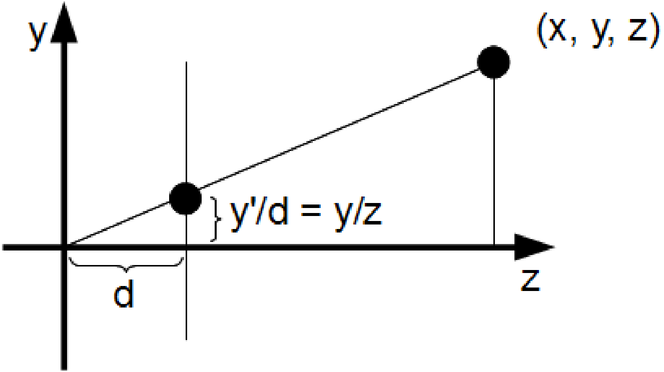
Homogeneous division

- Since the last row of $\mathbf{M}$ „real” projective transformation is not $[0, 0, 0, 1]^T$, therefore

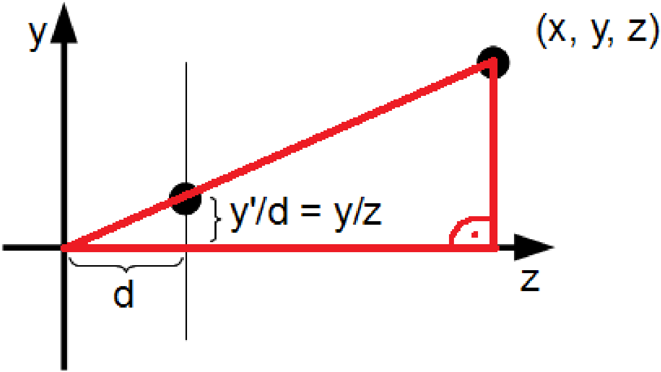
$$[x, y, z, w]^T = \mathbf{Mv}$$

  after transformation, $w \neq 1$ in general.
- If we want to transfer this point to the Euclidean space (because we want to display it, for example), then we have to divide by w.
- (Only if $w \neq 0$, of course)
- This is called homogeneous division.

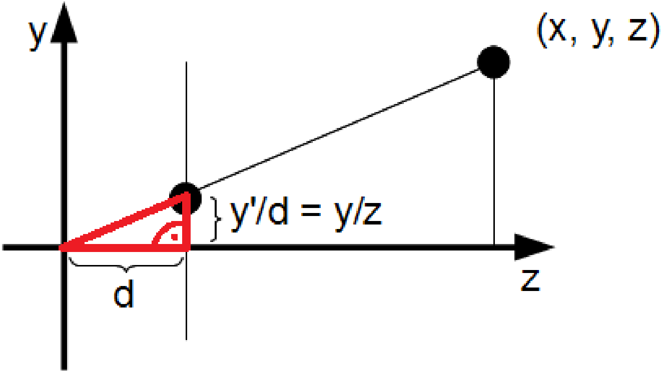
Central projection



Central projection



Central projection



Central projection

- That is:

$$x' = \frac{x}{z}d, \qquad y' = \frac{y}{z}d, \qquad z' = \frac{z}{z}d = d$$

- Matrix form of projecting to a plane, which is parallel to XY plane and located d units along Z axis with origin as the projection center:

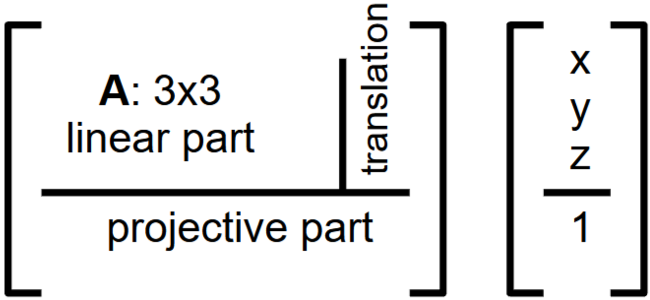
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{d} & 0 \end{bmatrix}$$

- After homogeneous division (with $\frac{z}{d}$) we get the above

Remark

- The projective transformations of the plane are uniquely defined by four independent points and their images
- The projective transformations of the space are uniquely defined by five independent points and their images

Transformation matrices



Transformation matrices

- What happens if the fourth coordinate of our vector is zero (i.e, if the four number represents a vector)?
- The translation part does not affect it!
- Pay attention: some use a convention to multiply from the left with the vector, in that case the transpose of our matrices should be used.

Transformation matrices