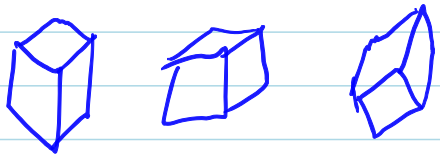
Viewing 2: projections

- 2nd part of viewing pipeline: Viewing
  - Planar geometric projections: map lines to lines
    - Parallel ← used often in CAD
    - Perspective ← common, ☺ very realistic

- Parallel; projection point is ∞
  - ① Orthographic: projectors ⊥ to projection plane
    - (XZ, YZ, XY) ← parallel to one plane of the world

plane at $z=0$: XY

  - ② Axonometric: like orthographic, but
    - Pplane has ANY orientation



  - ③ Oblique: projectors can make any angle to Pplane & any orientation

- Perspective: centre of projection / eye point
  - lines' intersection gives image
    - i.e vanishing point: (XY, YZ, XZ)
      - ① 1 point: 2 planes parallel

→ $x_p = \frac{d_x}{z}$, $y_p = \frac{d_y}{z}$ [similar △'s]

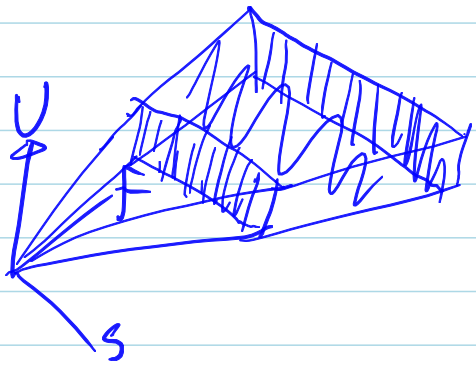
#Lab1 !!

- $\begin{bmatrix} dx/z \\ dy/z \\ d \\ 1 \end{bmatrix} = \left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & z/d & 0 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \right) \times \frac{d}{z}$ #Point projection

  THIS is used to give perspective effects

- ... but what can a camera see? OO
  - Objects in field of view, infront of it
  - FINITE distance: some objects become too small to see

- Viewing volumes: 3D shape, defined with 6 planes
  - parallel projection: cuboid
    - moves with camera
  - perspective: FRUSTUM



- Problems with perspective:
  - 2D image means → no depth info
    - ☹ Hidden surface removal impossible ... without

- ① Projection normalisation
  - Distorts Frustum into cube
  - Projects cube with default OpenGL projection
  - Distort cube, distort OBJECT inside of it too

#4D #Homogeneous coordinates

- Clipping: occurs in cube produced from projection normalisation

- Perspective division: FINAL stage after
  - Projection transformation
  - Clip to view volume