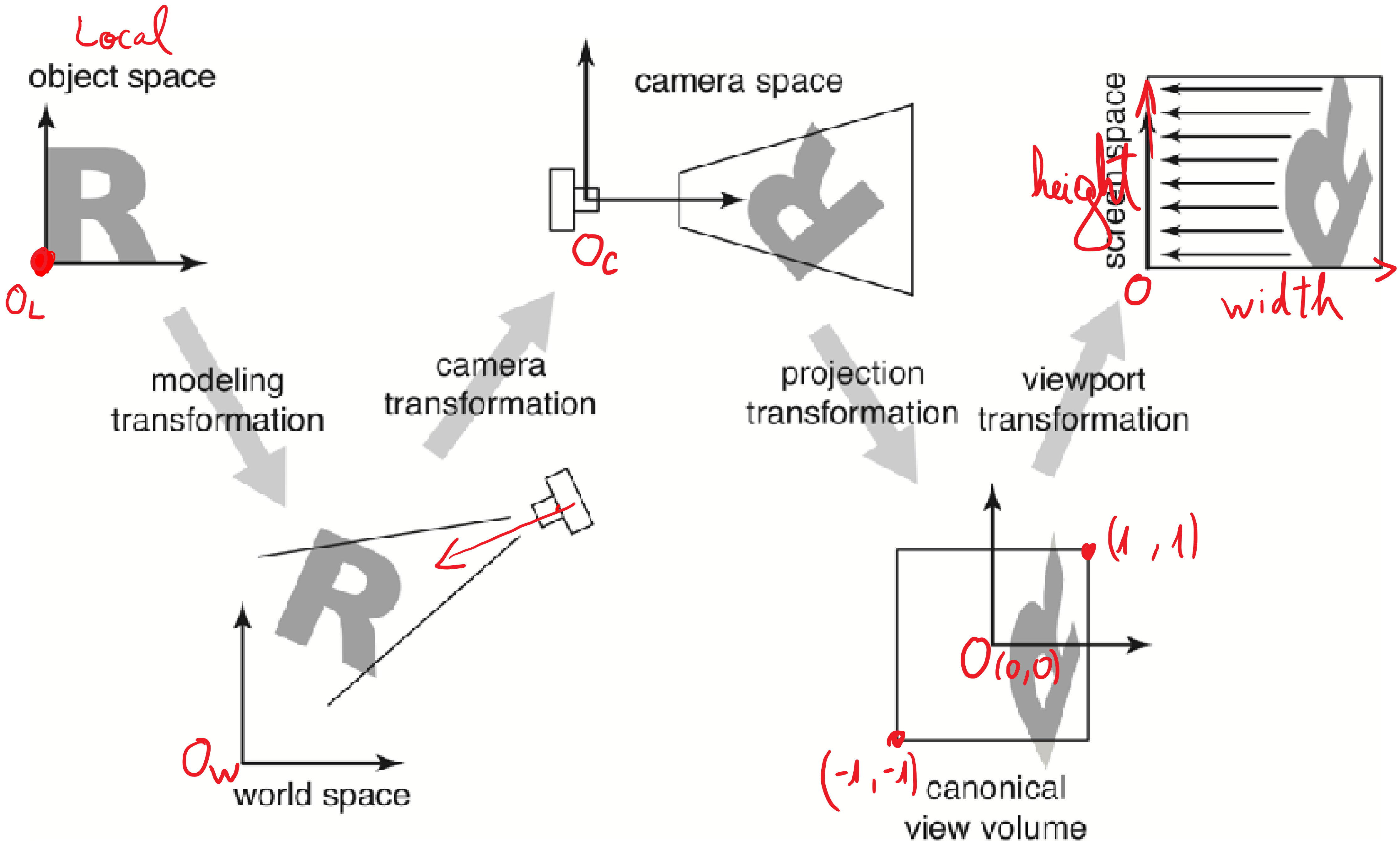
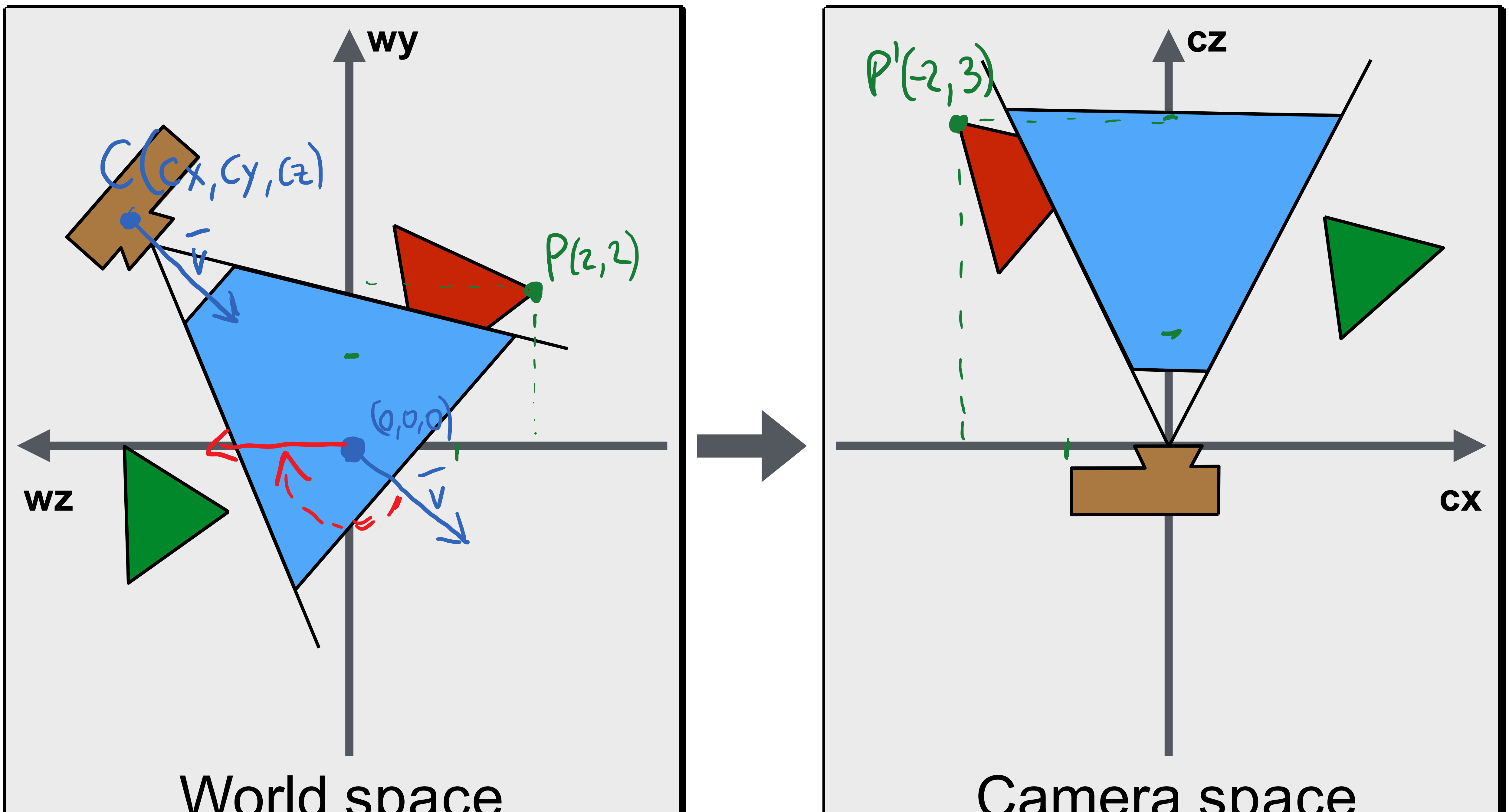
Transformarea de vizualizare

Secvența de tranformare



[Peter Shirley et al. Fundamentals of Computer Graphics]

Transformarea de vizualizare



Translatie : (Cx, Cy, Cz) ⇒ (0,0,0)
Rotatie : aliniere $\bar{v}$ la axa Oz

} ⇒ Aplicate pe puncte. Ex: M = R · T

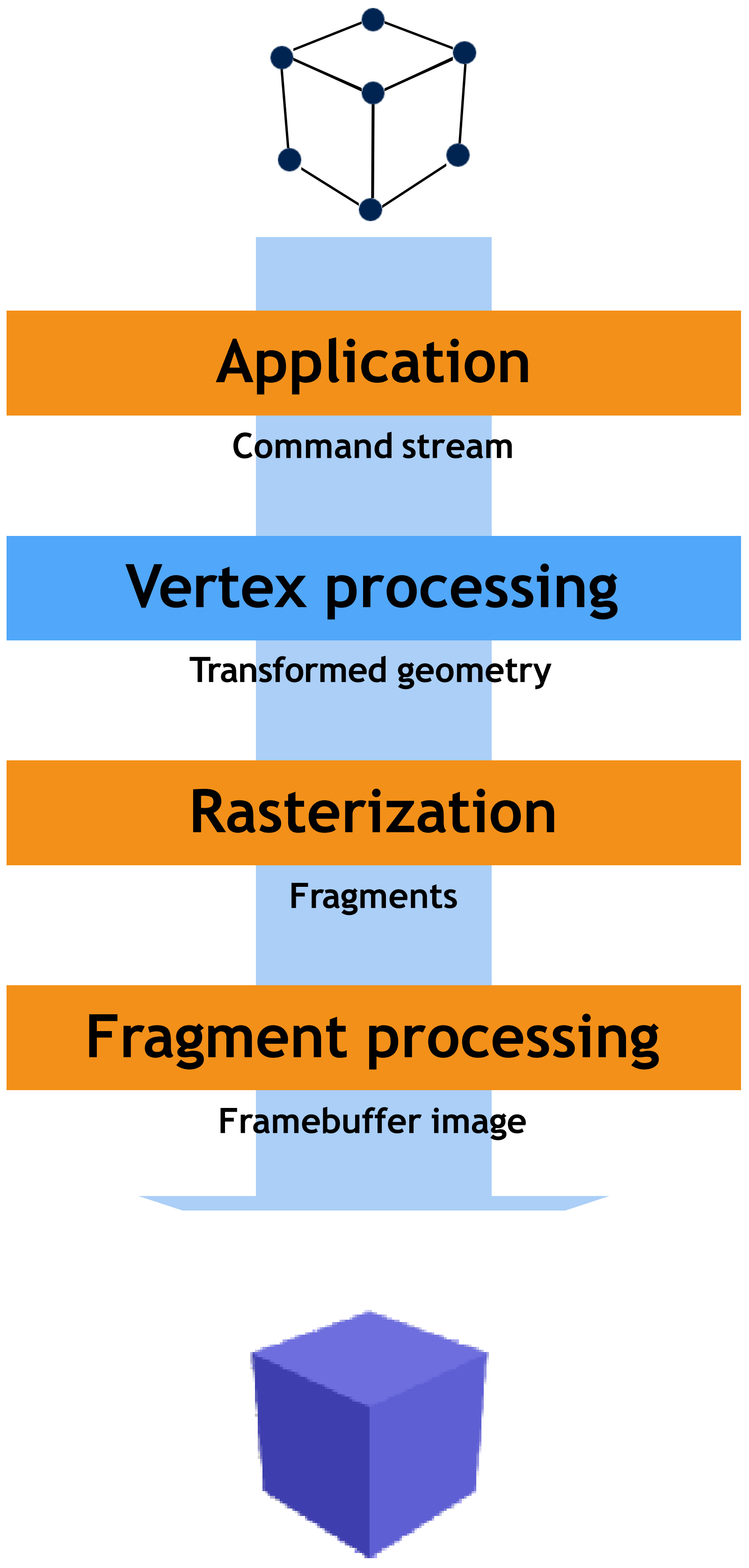
M · P = P'

Transformarea de vizualizare

Mapează puncte 3D (specificate prin coordonatele x, y și z) în puncte 2D în planul imaginii (specificate în pixeli)

Secvență de 3 transformări:

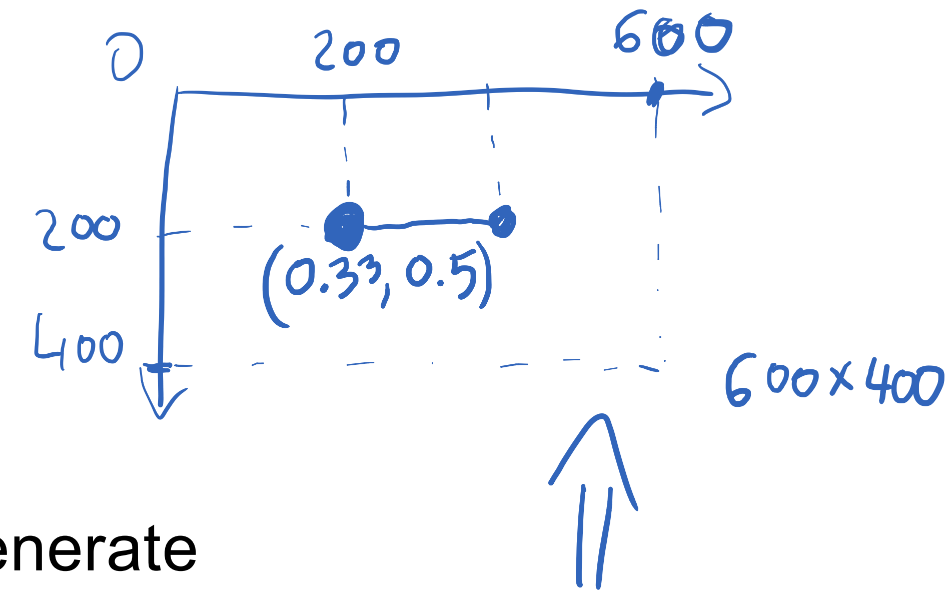
1. transformarea de cameră (camera/eye transformation)
2. transformarea de proiecție (projection transformation)
3. transformarea de viewport (viewport/windowing transformation)



Transformarea de vizualizare

1. **transformarea de cameră** (camera/eye transformation)
   - transformare de tip “rigid body”
   - plasează camera de vizualizare în origine
   - depinde doar de poziția și orientarea camerei de vizualizare
2. **transformarea de proiecție** (projection transformation)
   - proiectează punctele din spațiul cameră
   - coordonatele punctelor vizibile vor fi în intervalul $[-1, 1]$
   - depinde doar de tipul de proiecție
3. **transformarea de viewport** (viewport/windowing transformation)
   - mapează zona vizibilă la rezoluția dorită
   - depinde doar de dimensiunea și poziția în cadrul imaginii finale

Noțuni

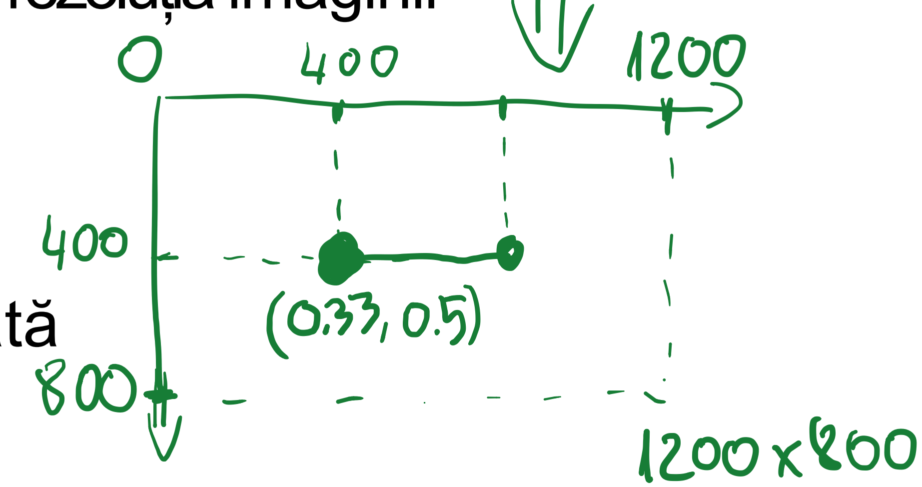


Coordonate dispozitiv (Device coordinates)

- Valori întregi dependente de rezoluția imaginii generate

Coordonate dispozitiv normalizate (Normalized device coordinates - NDC)

- Valori exprimate procentual și independente de rezoluția imaginii generate



Volum de vizualizare (View volume)

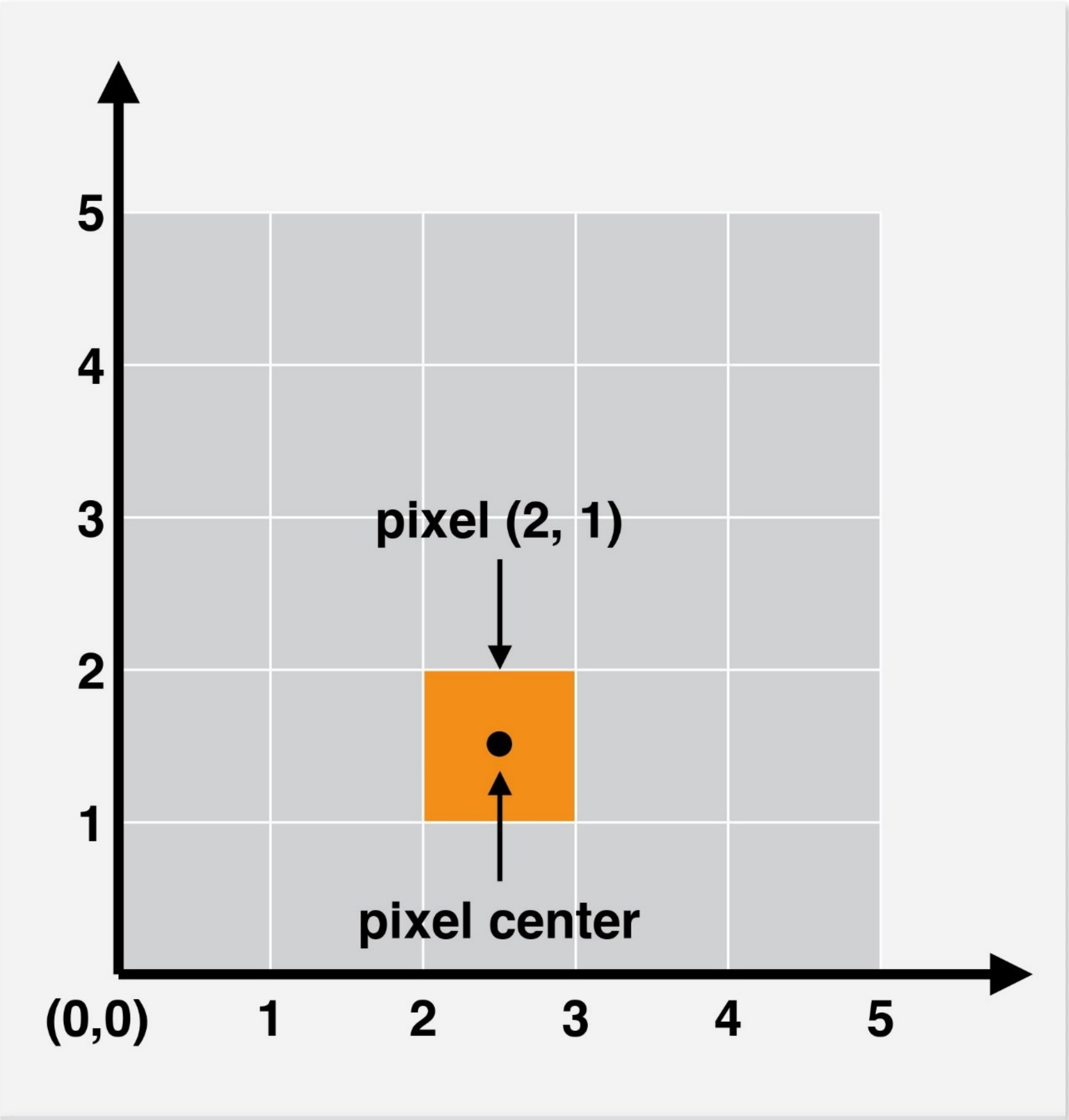
- Specifică porțiunea din scena 3D care va fi trasată
- Pentru proiecție paralelă - paralelipiped
- Pentru proiecția perspectivă - trunchi de piramidă

Volum de vizualizare canonic (Canonical view volume)

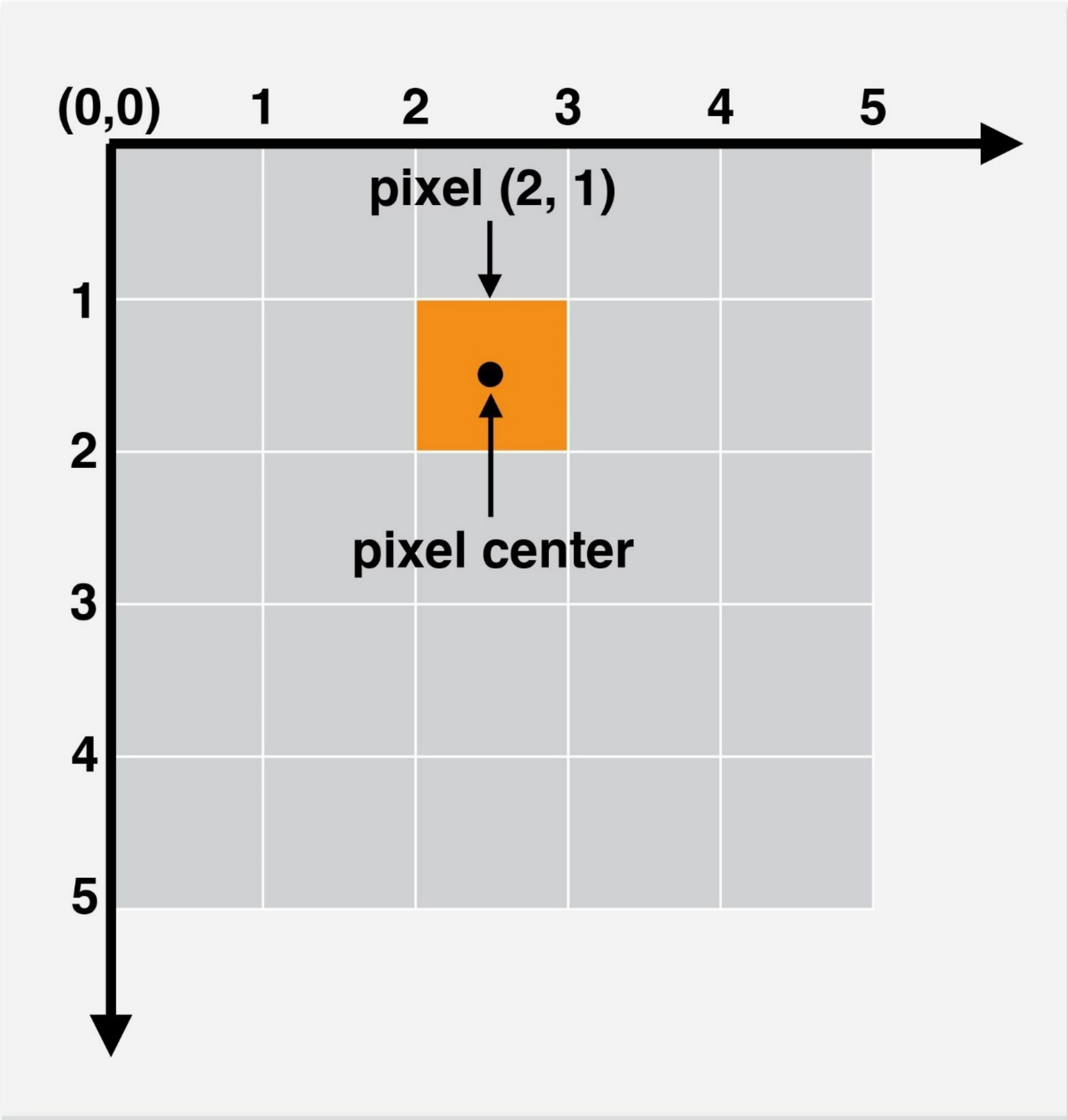
- Cub centrat în origine, laturile aliniate cu axele sistemului de coordonate

(-1, -1, -1) to (1, 1, 1)

Definirea unui pixel

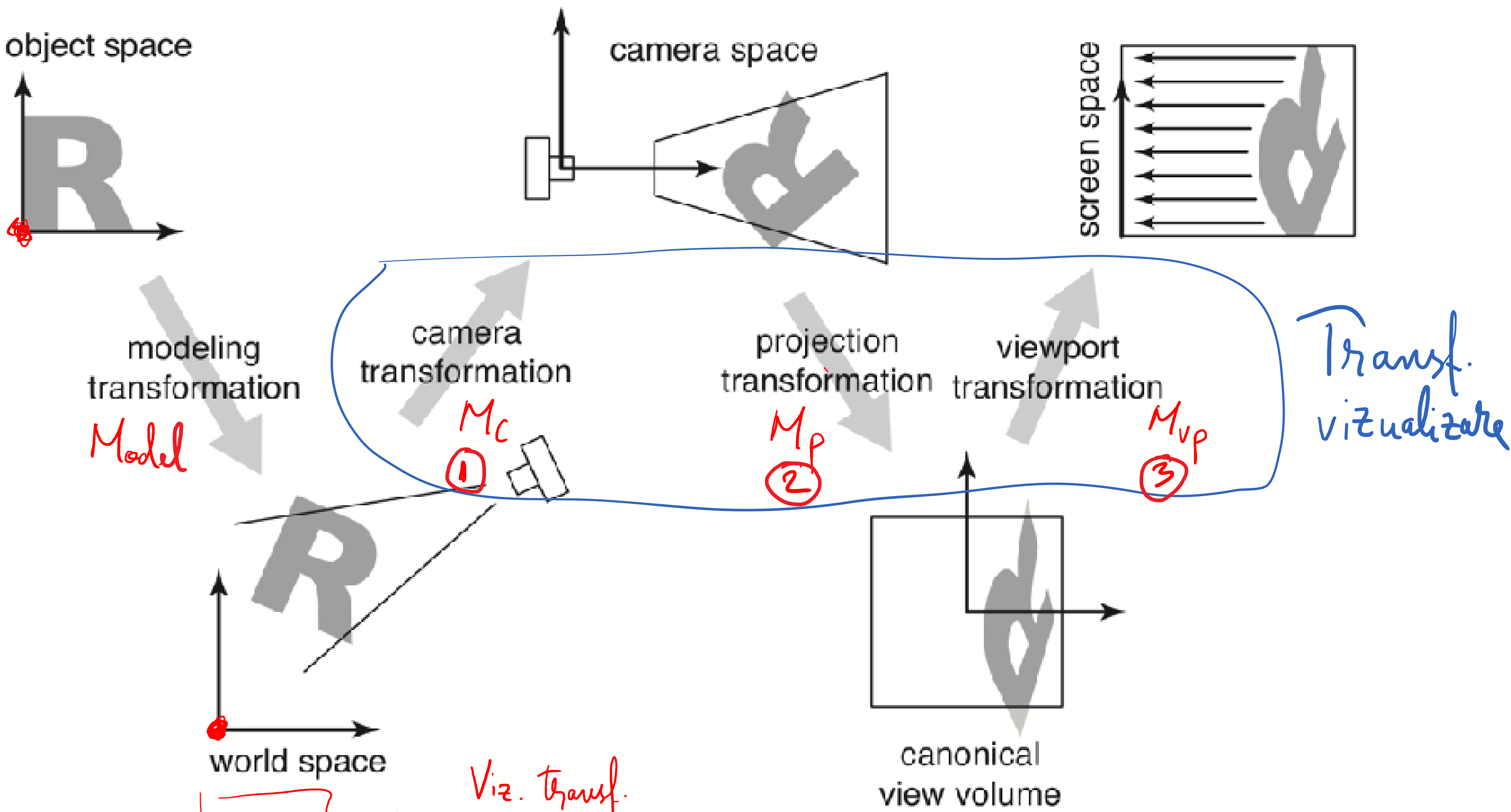


OpenGL



DirectX > 10

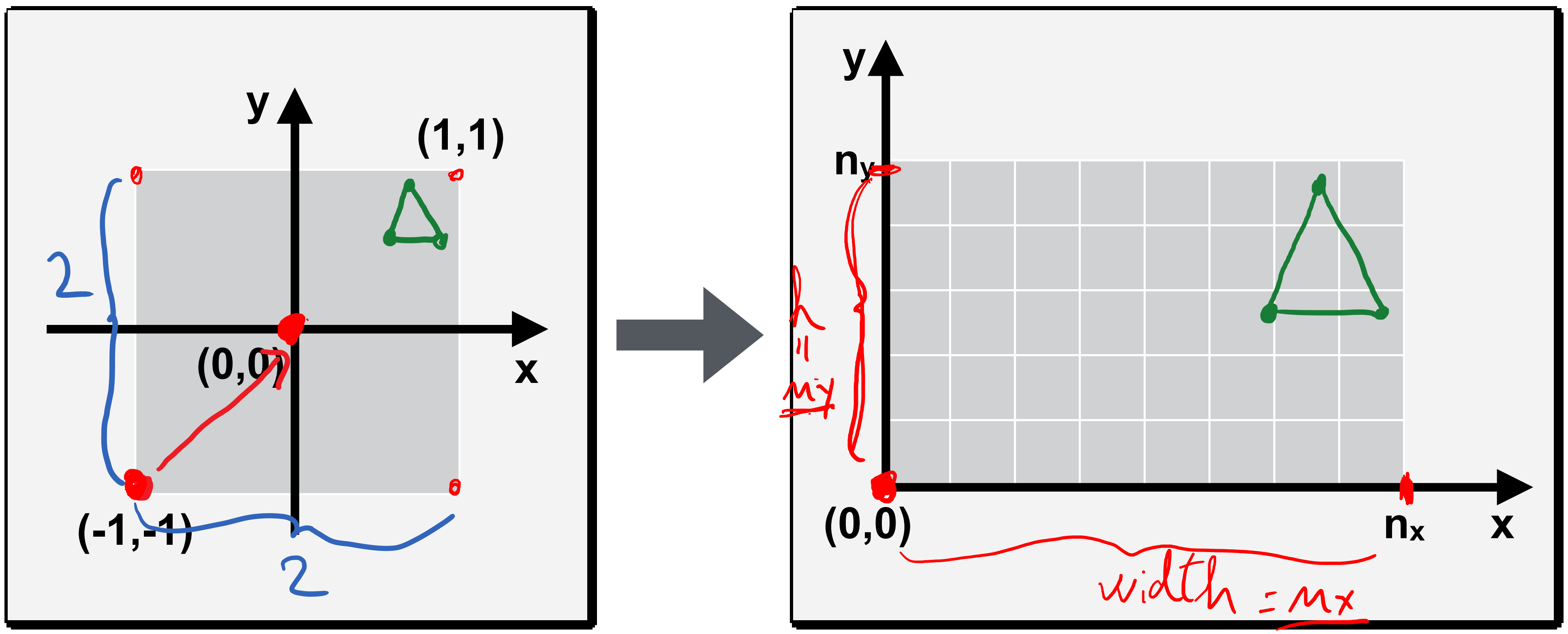
Secvența de tranformare



$$\boxed{M} = \underbrace{\overbrace{M_{vp} \cdot M_p \cdot M_c}^{\text{Viz. transf.}}}_{} \cdot Model$$

[Peter Shirley et al. Fundamentals of Computer Graphics]

Transformarea de viewport



$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & \boxed{1} \\ 0 & 1 & 0 & \boxed{1} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

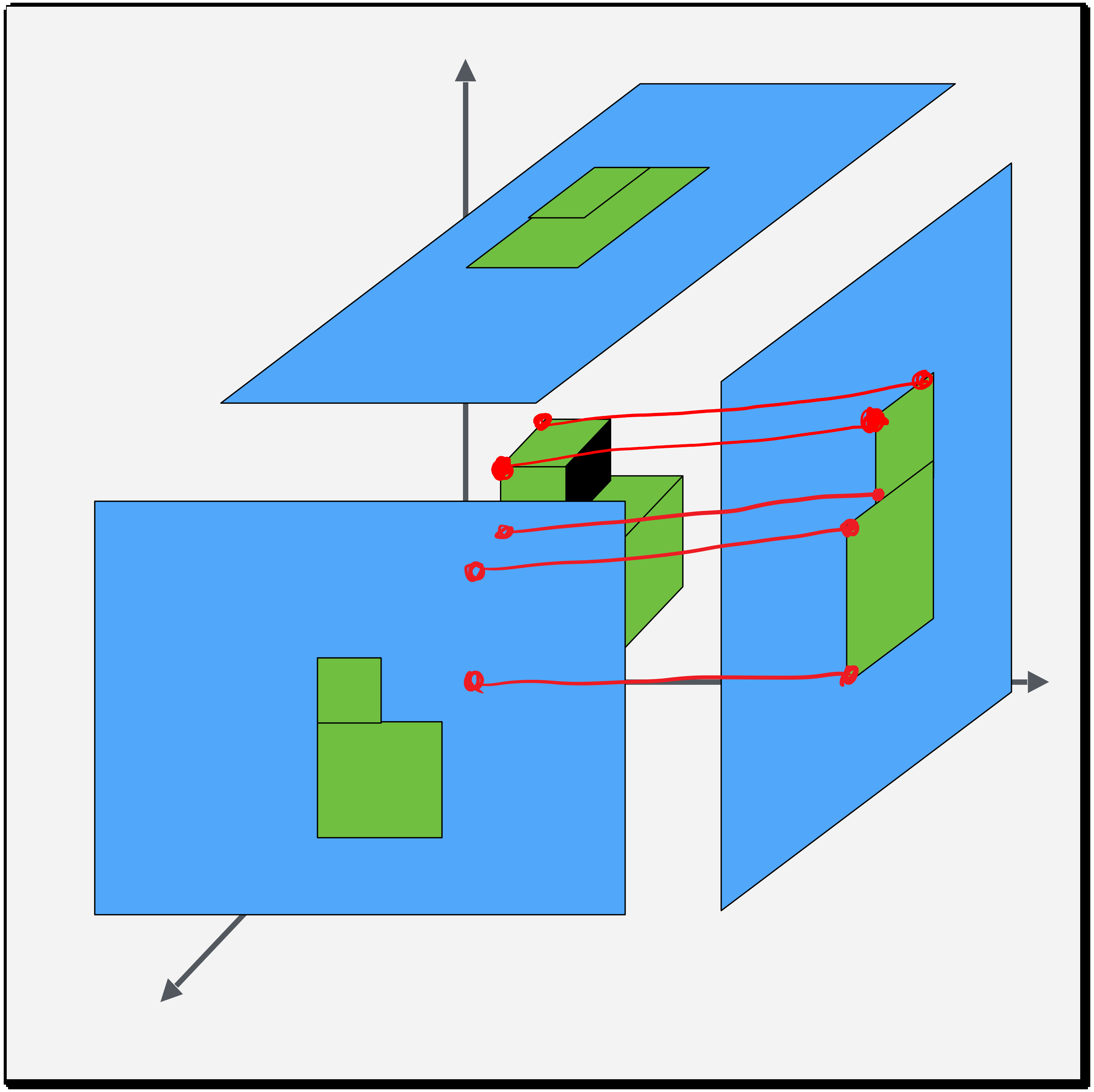
$$\mathbf{S} = \begin{bmatrix} \boxed{\frac{n_x}{2}} & 0 & 0 & 0 \\ 0 & \boxed{\frac{n_y}{2}} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Scalare

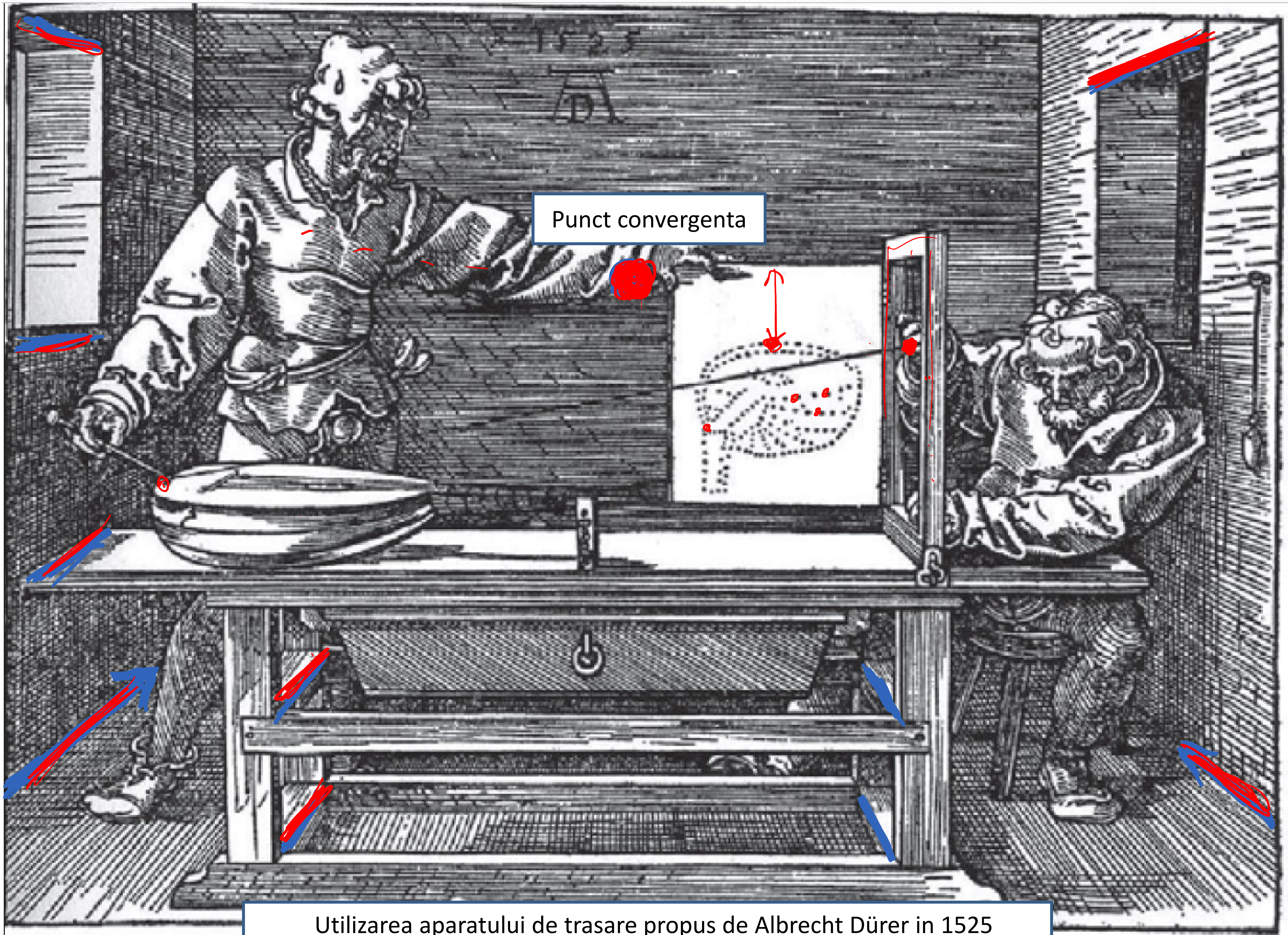
$$\mathbf{M}_{vp} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Trasl.

Proiecție

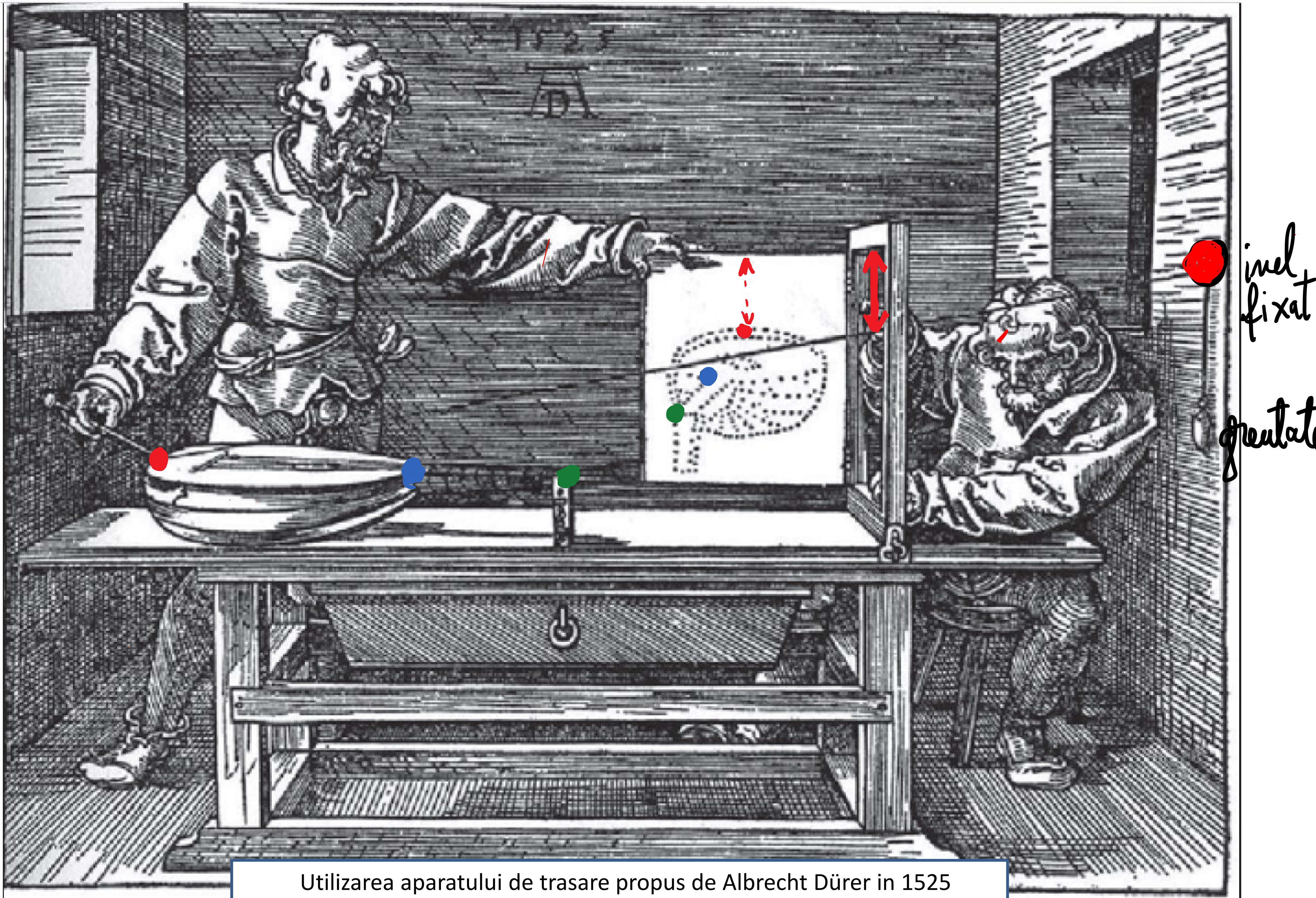


Proiecție



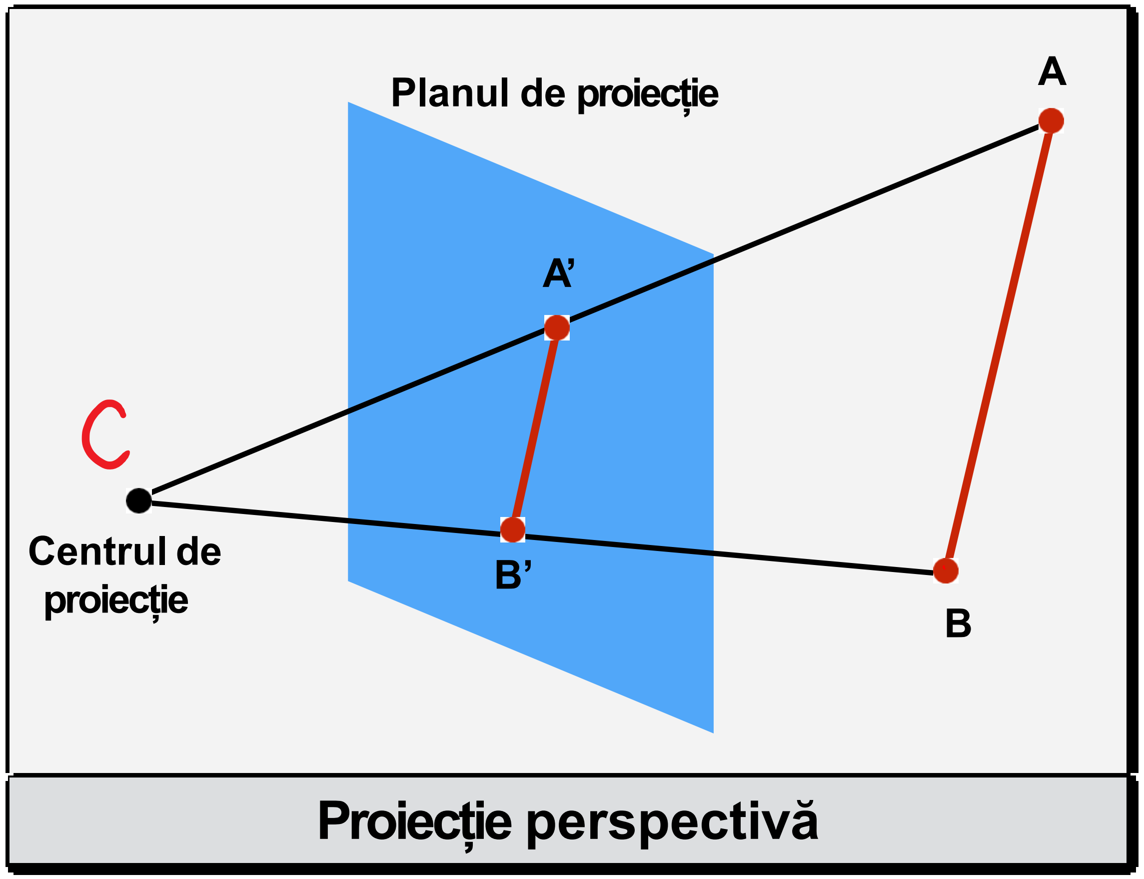
Utilizarea aparatului de trasare propus de Albrecht Dürer in 1525

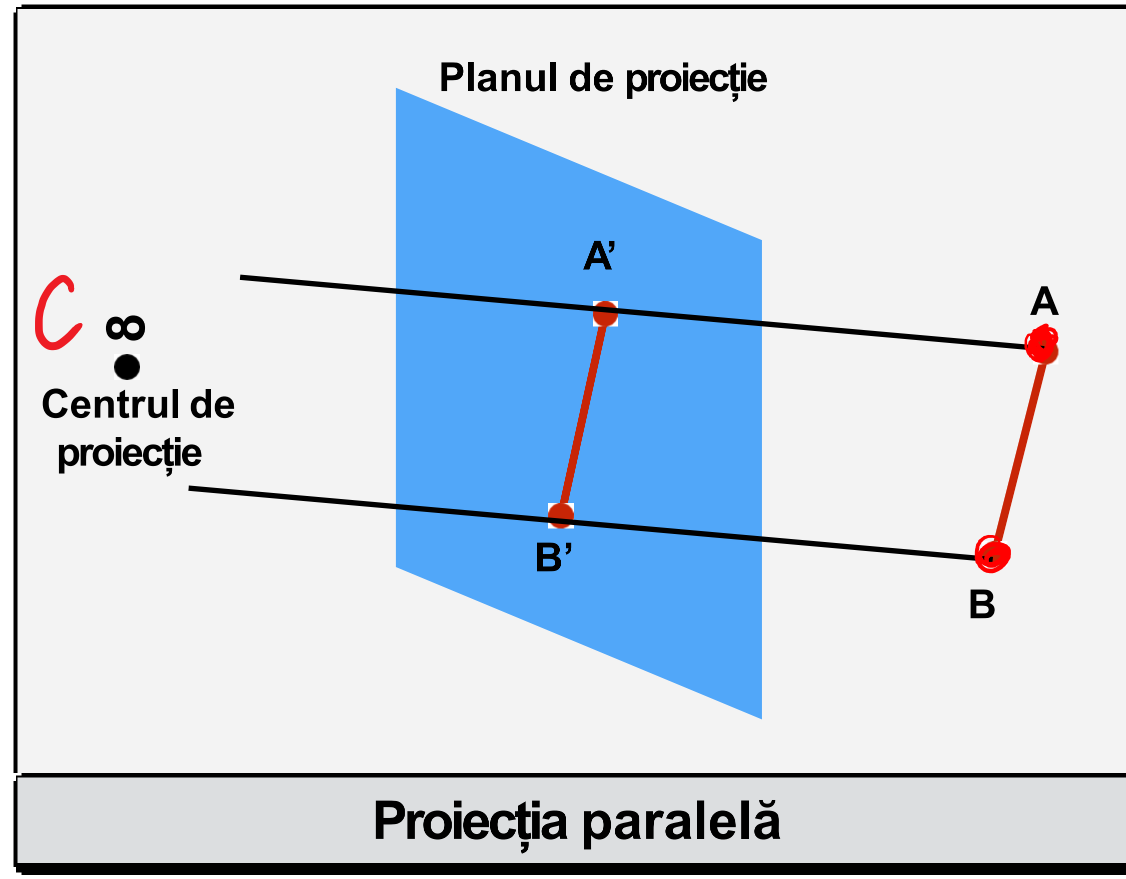
Proiecție



Utilizarea aparatului de trasare propus de Albrecht Dürer in 1525

Proiecție perspectivă vs. paralelă

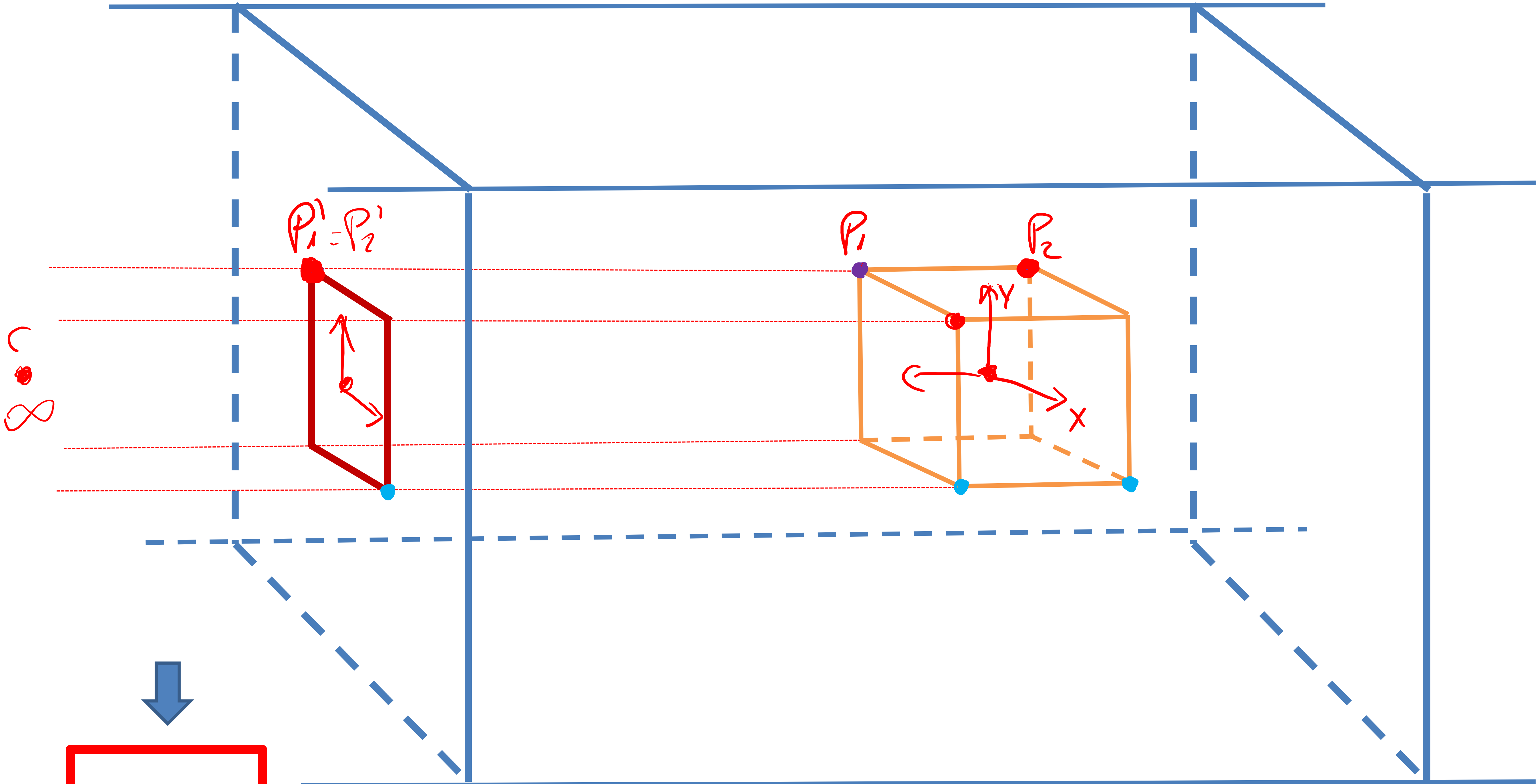


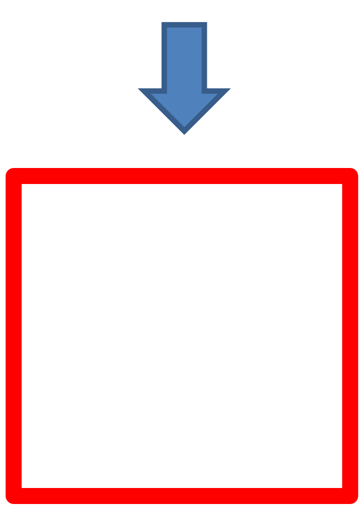


- Razele de proiecție converg către centrul de proiecție (locația observatorului/ camerei de vizualizare)
- Obiectele apar mai mici pe măsură ce crește distanța către centrul de proiecție
- Liniile paralele cu planul de proiecție rămân paralele după aplicarea proiecției
- Liniile care nu sunt paralele cu planul de proiecție vor converge către un punct numit punct de fugă (vanishing point)

- Razele de proiecție sunt paralele Punct
- de convergență la infinit
- Poziția observatorului / camerei de vizualizare la infinit
- Liniile paralele rămân paralele după aplicarea proiecției

Proiecție paralelă





imagine rezultat

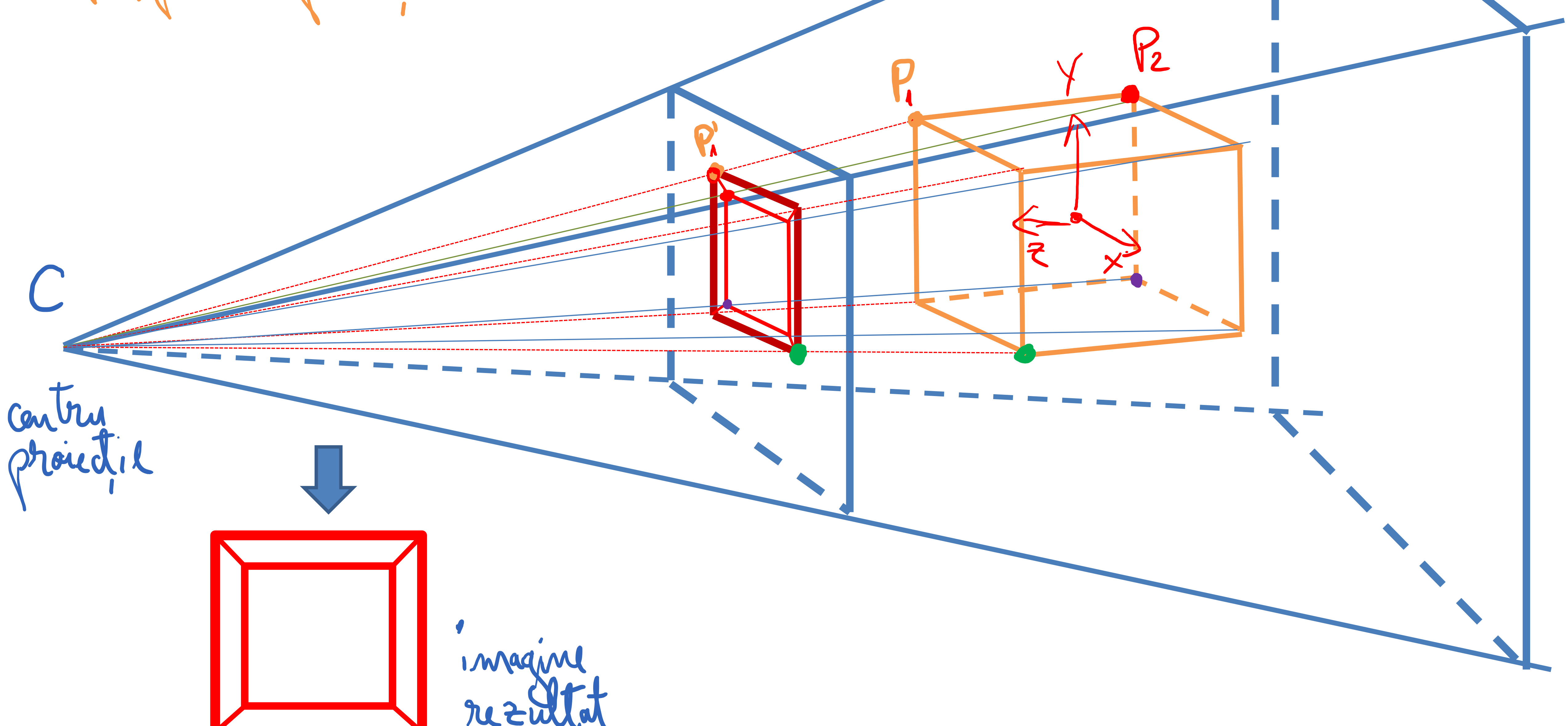
Raze de proiecție paralele
Centru proiecție => ∞

Proiecție perspectivă

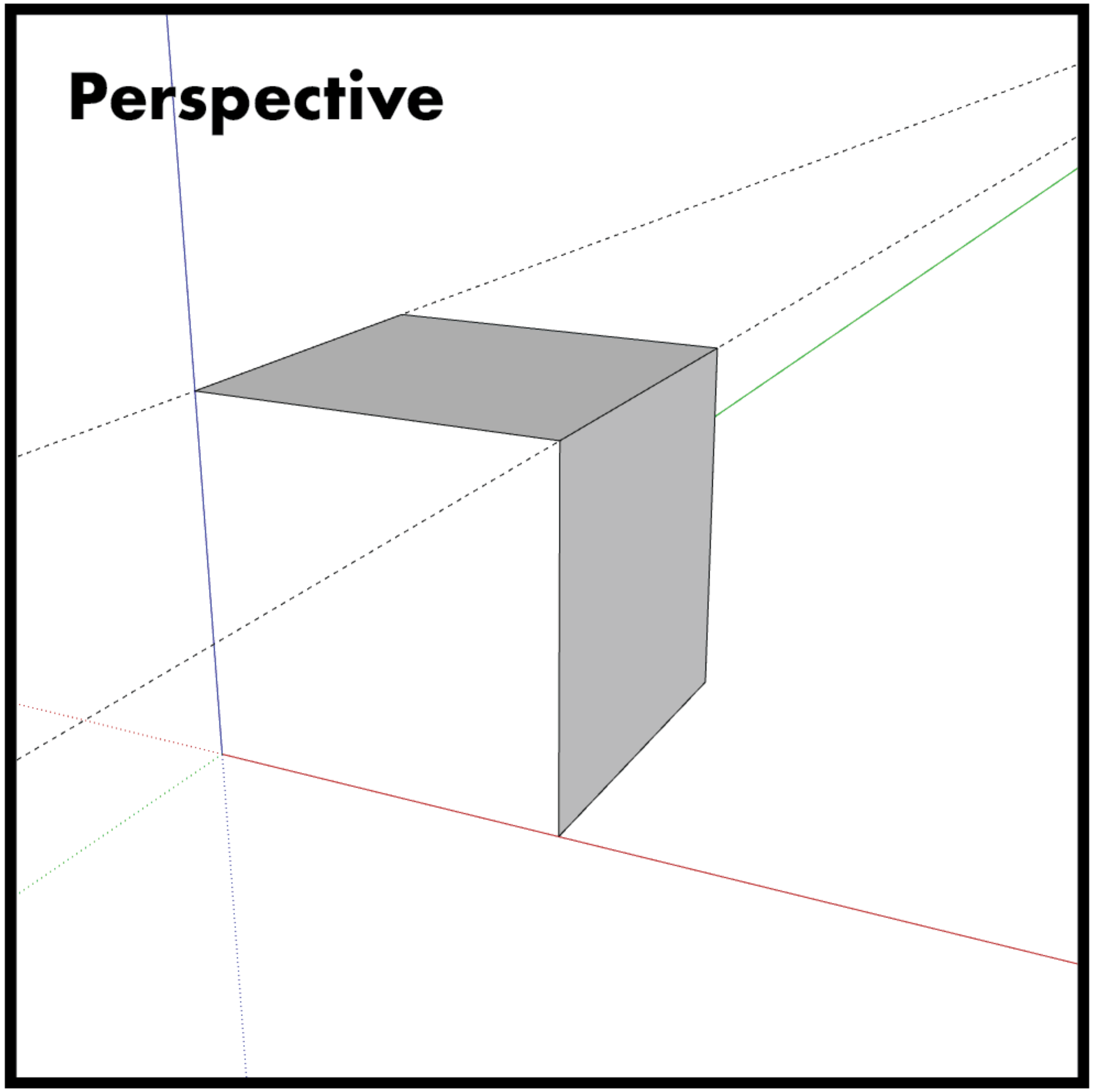
Raze de proiecție convergente în C

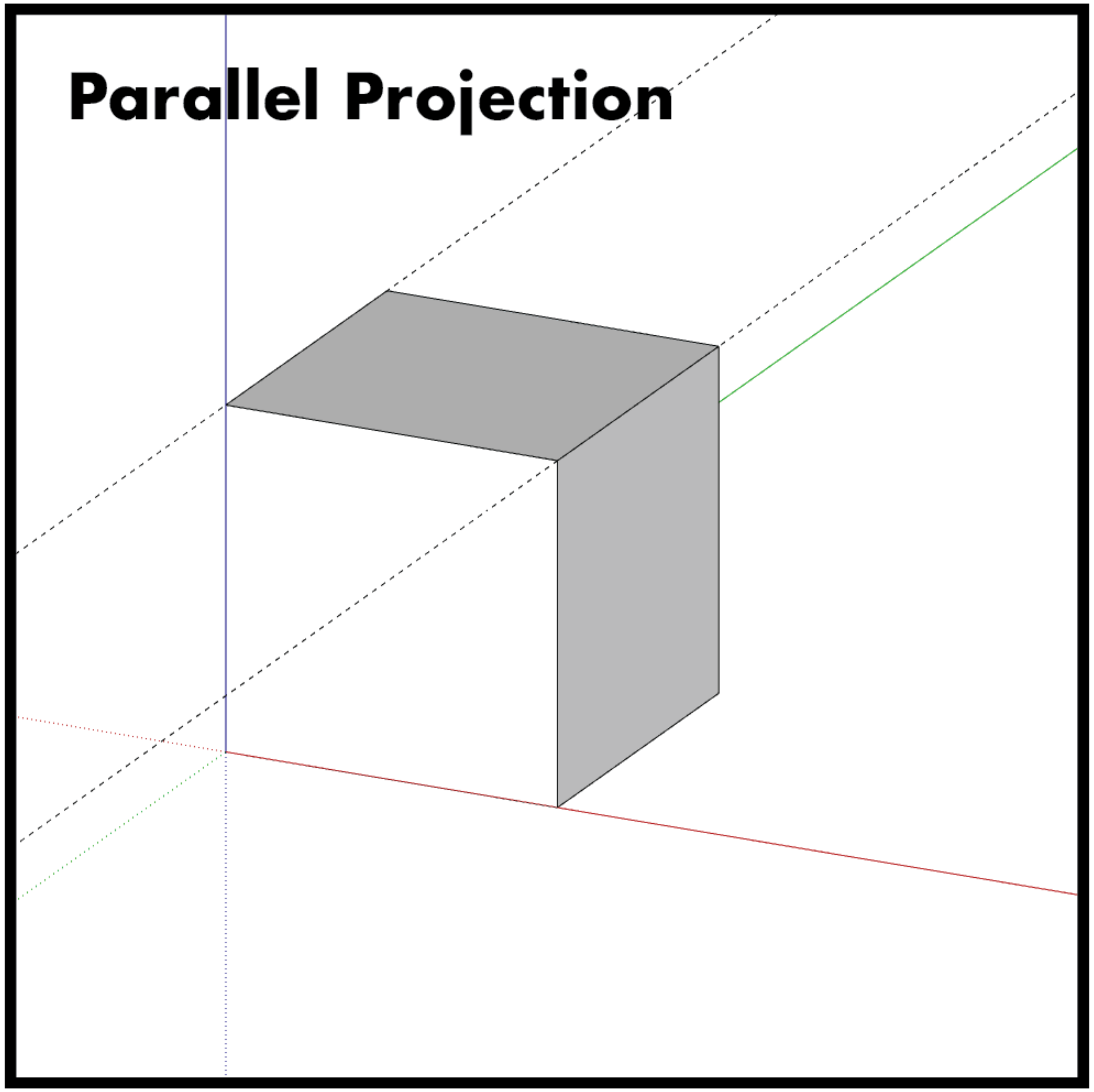
P->punct obiect

P'->punct proiecție

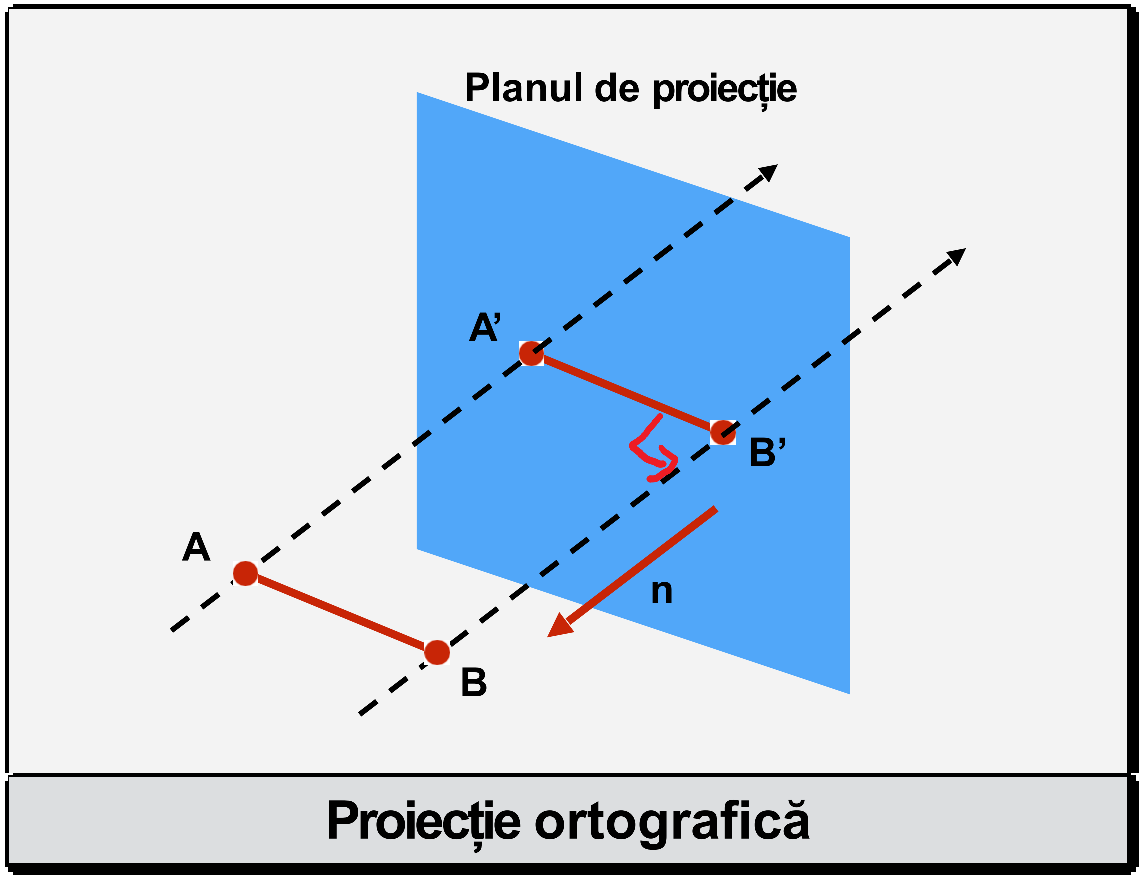


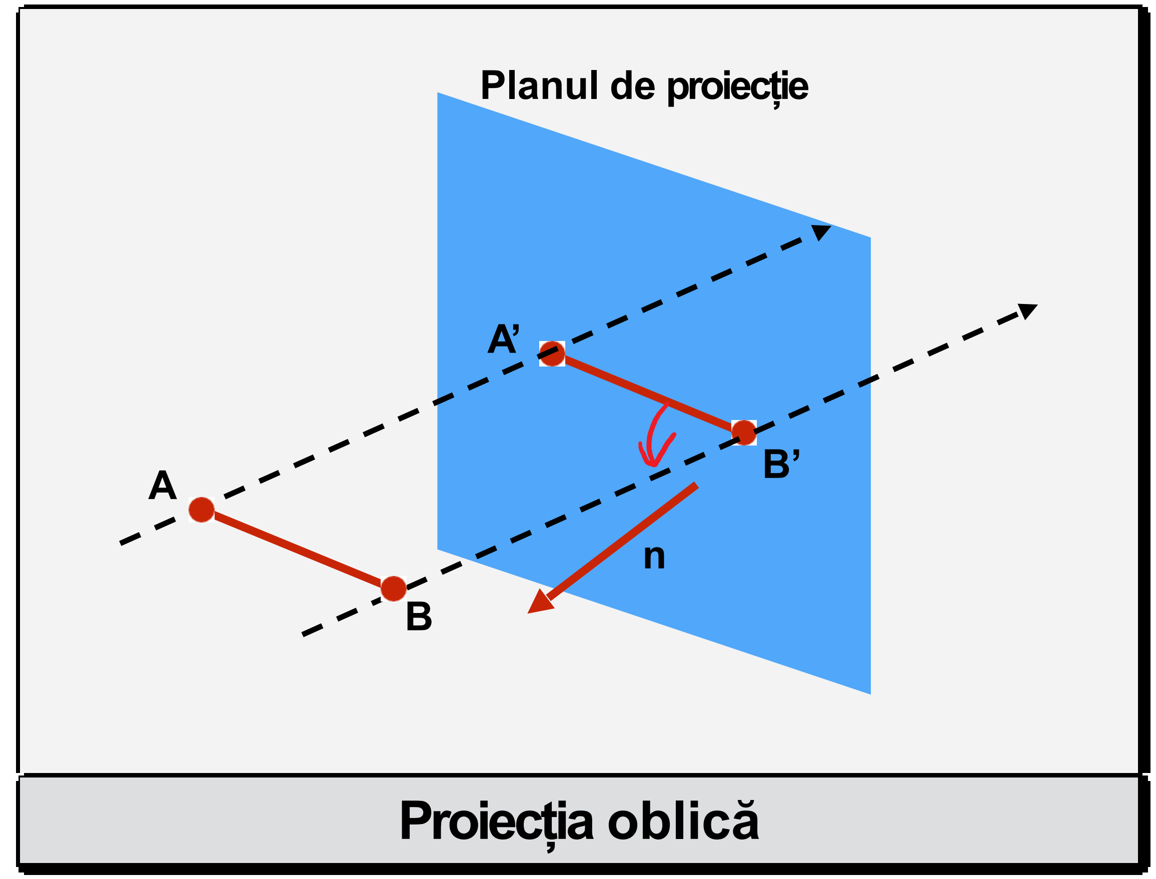
Proiecție perspectivă vs. paralelă





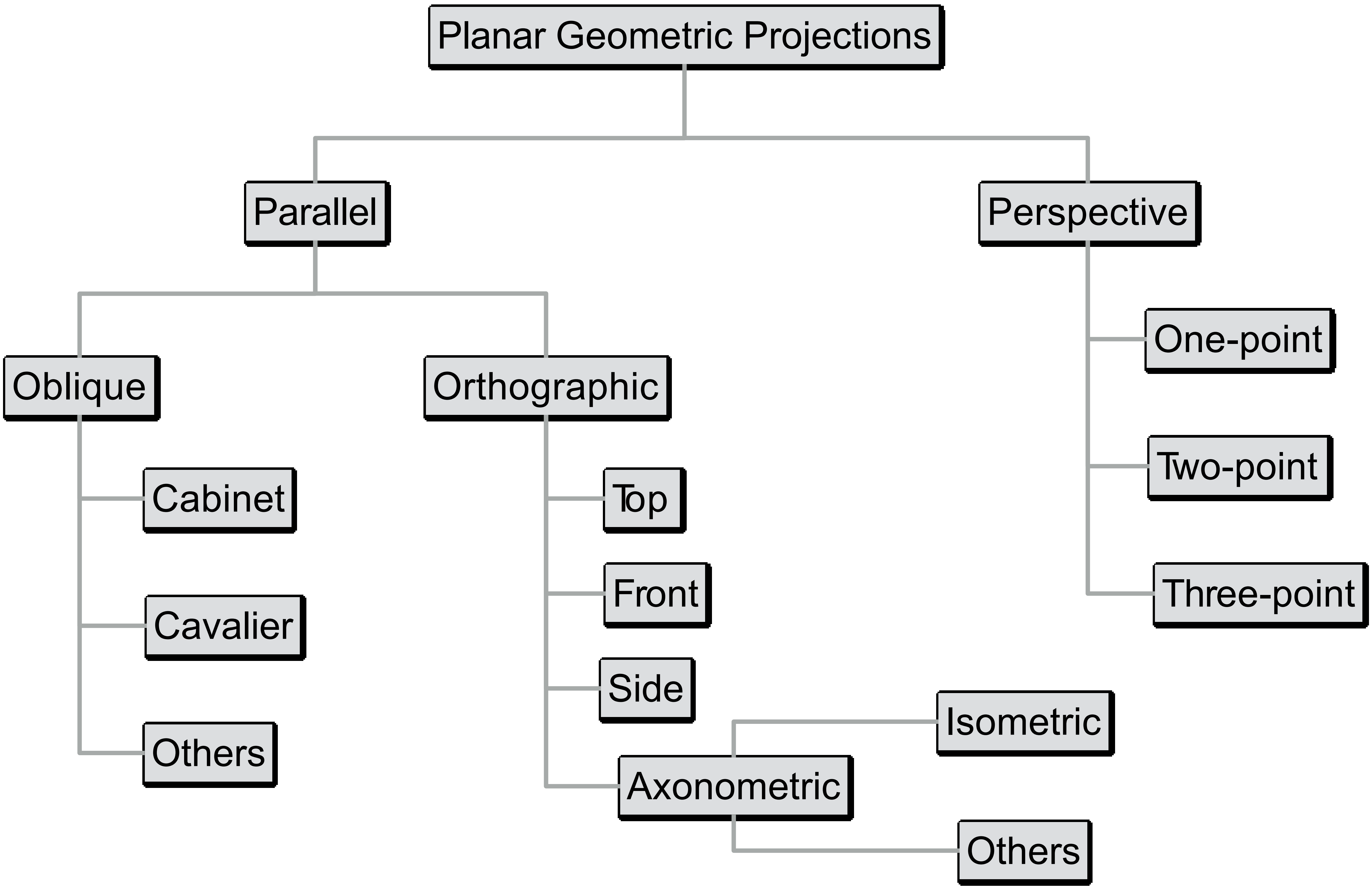
Proiecția paralelă





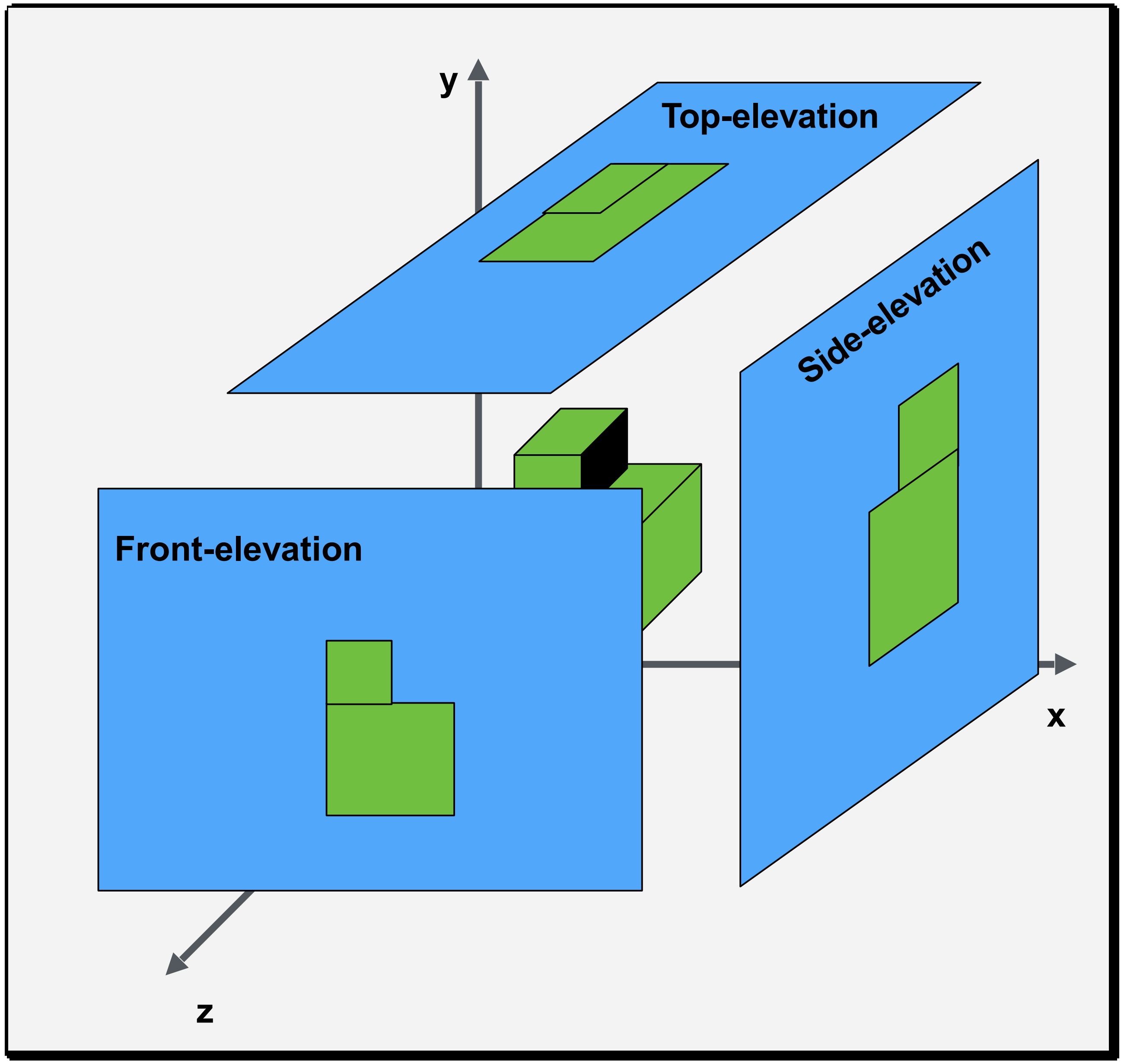
- Direcția de proiecție este perpendiculară pe planul de proiecție
- Direcția de proiecție diferă de vectorul normal la planul de proiecție

Taxonomia proiecțiilor

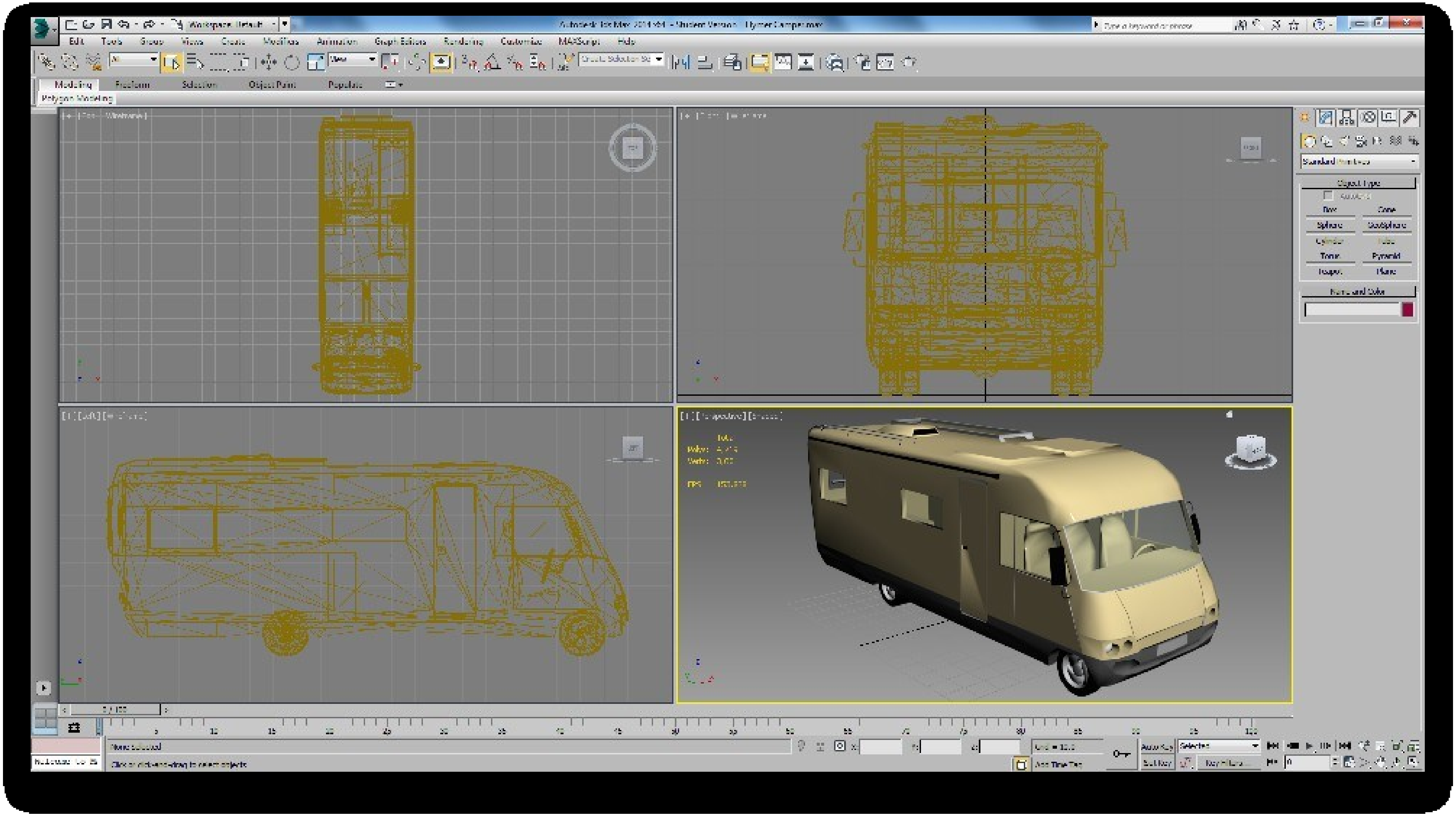


Proiecția ortografică

- **Vedere de sus** (Top-elevation)
  - Planul de proiecție este perpendicular pe axa **y**
- **Vedere frontală** (Front-elevation)
  - Planul de proiecție este perpendicular pe axa **z**
- **Vedere laterală** (Side-elevation)
  - Planul de proiecție este perpendicular pe axa **x**



Proiecția ortografică - exemplu

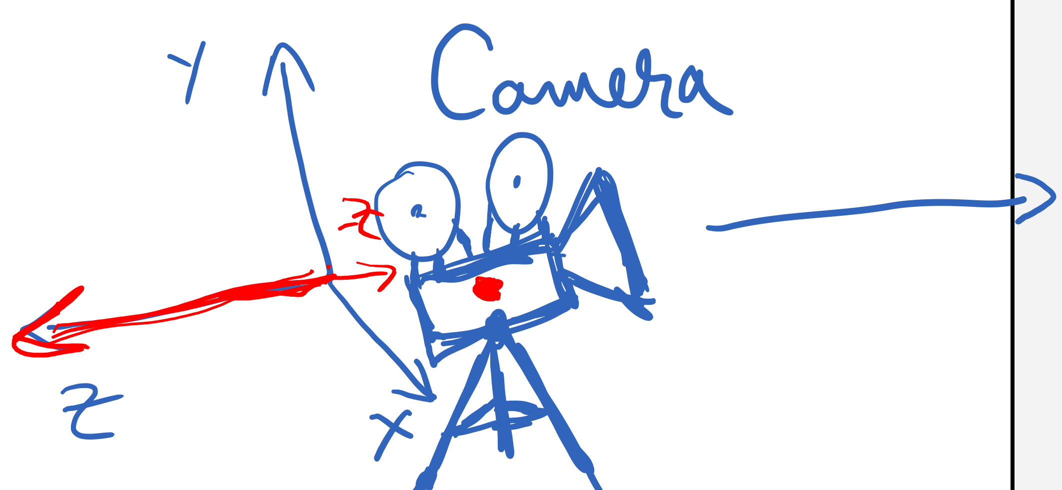


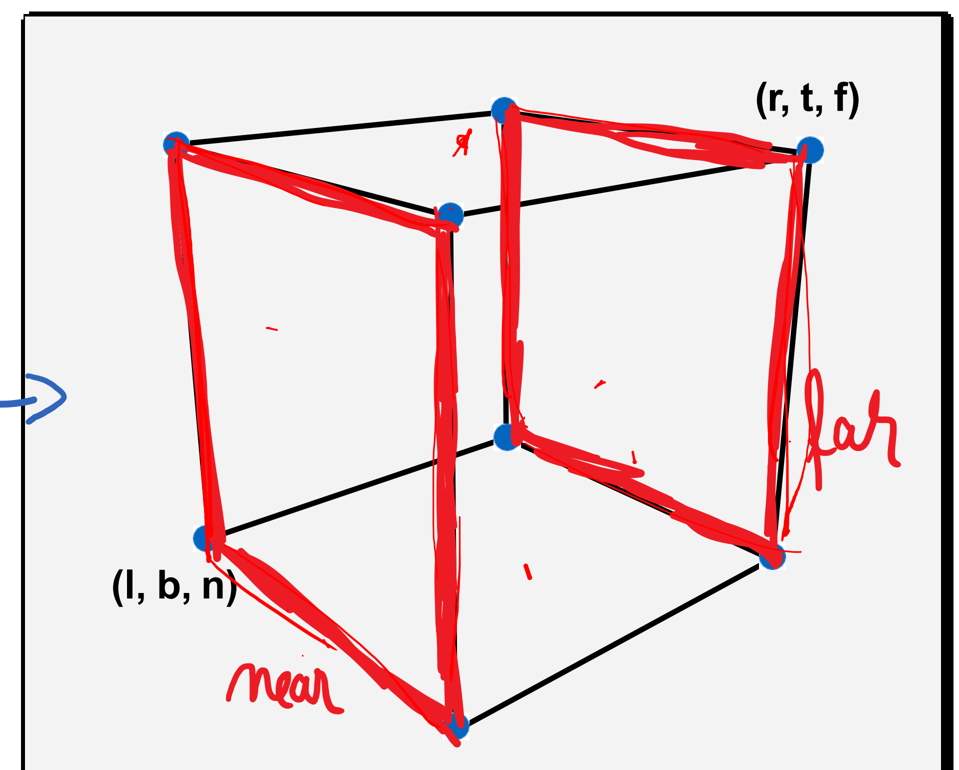
Transformarea proiecție ortografică

Volumul de vizualizare este un paralelipiped având laturile paralele cu axele sistemului de coordonate:

- $\underset{x}{[l, r]} \times \underset{y}{[b, t]} \times \underset{z}{[f, n]} \Rightarrow \boxed{\underset{x}{[-1, 1]} \times \underset{y}{[-1, 1]} \times \underset{z}{[-1, 1]}}$

Observatorul / camera de vizualizare privește de-a lungul axei z negative





Volumul de vizualizare

Transformarea proiecție ortografică

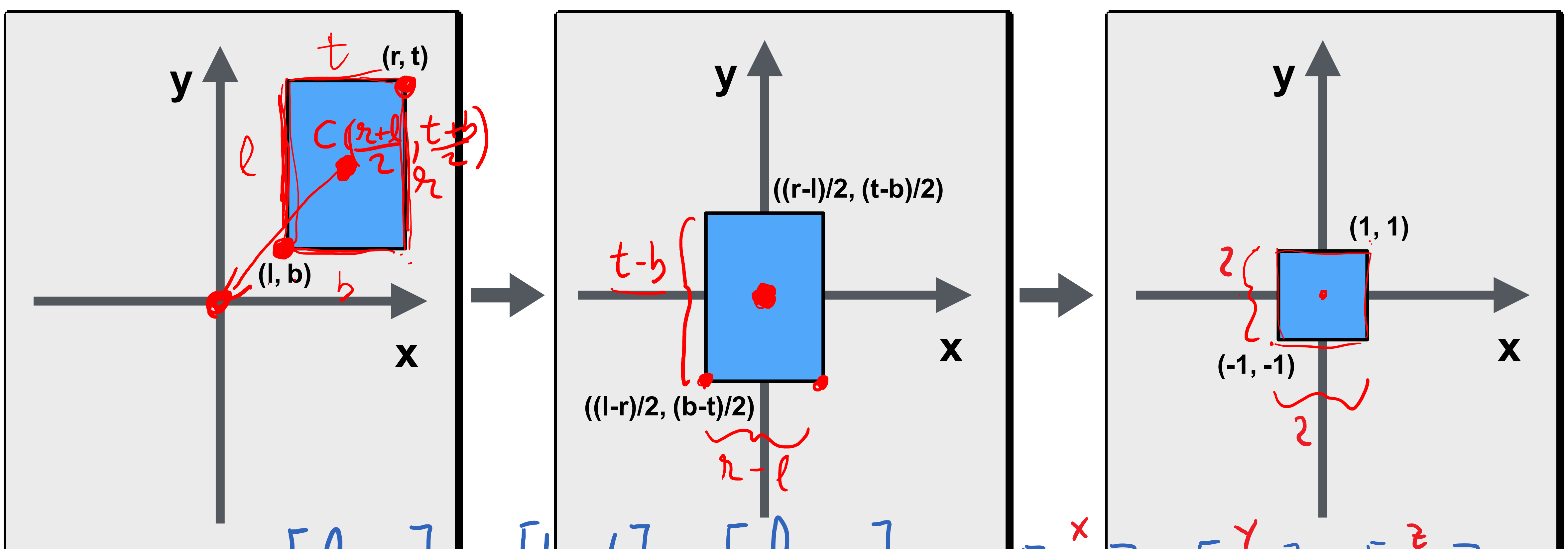
Volum vizualizare : $[l, r] \times [b, t] \times [f, n]$

Transformarea proiecție ortografică

Transformare în volumul canonic de vizualizare

- obs: n > f (direcția de vizualizare este pe direcția axei z negative)
- translație și scalare

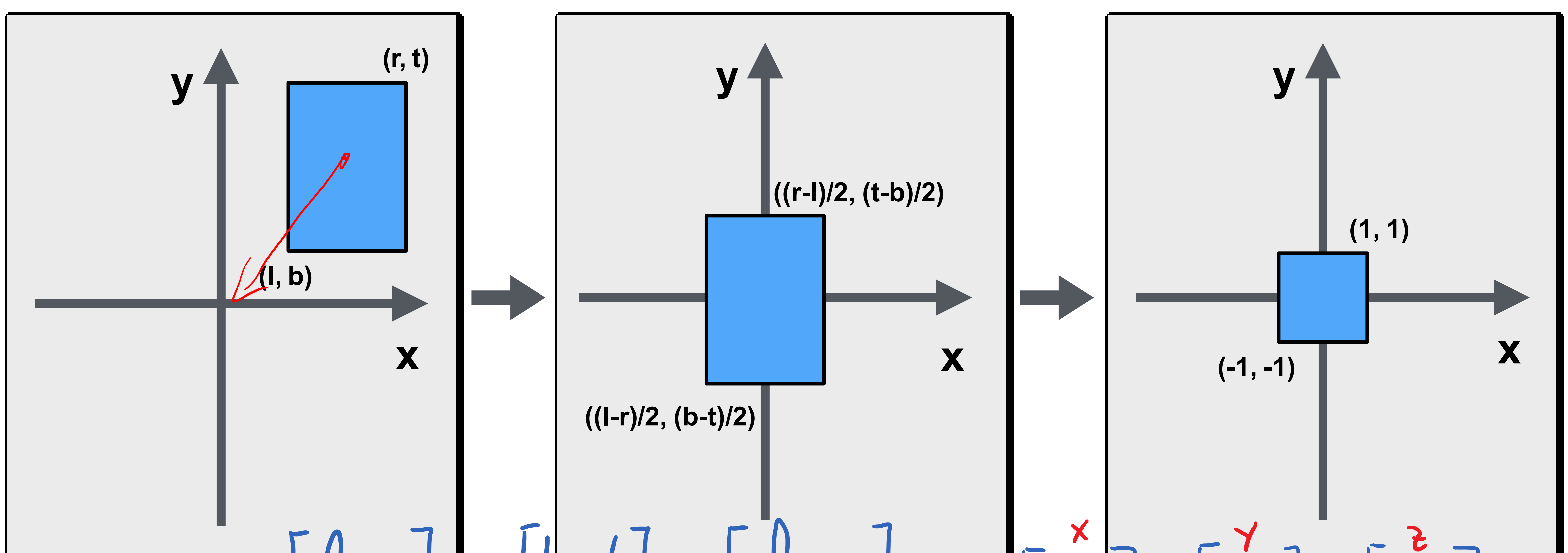
$$T = \begin{array}{cc} \begin{bmatrix} 1 & 0 & 0 & -\frac{r+l}{2} \\ 0 & 1 & 0 & -\frac{t+b}{2} \\ 0 & 0 & 1 & -\frac{n+f}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{array}{l} T_x \\ T_y \\ T_z \\ \end{array} \end{array} \qquad S = \begin{array}{c} \begin{array}{cccc} S_x & S_y & S_z & \end{array} \\ \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{n-f} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$



Conversie: $[l, r] \times [b, t] \times [f, n] \Rightarrow \overset{x}{[-1, 1]} \times \overset{y}{[-1, 1]} \times \overset{z}{[-1, 1]}$

Transformarea proiecție ortografică

$$\mathbf{M}_{orth} = \mathbf{ST} = \begin{array}{c} \begin{matrix} \quad S_x & \quad\quad S_y & \quad\quad S_z & \quad\quad T \end{matrix} \\ \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & -\frac{n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$



Conversie: $[l, r] \times [b, t] \times [f, n] \Rightarrow \overset{x}{[-1, 1]} \times \overset{y}{[-1, 1]} \times \overset{z}{[-1, 1]}$

Transformarea proiecție ortografică

construct $\mathbf{M}_{vp}$

construct $\mathbf{M}_{orth}$

$\mathbf{M} = \mathbf{M}_{vp}\mathbf{M}_{orth}$

for each line segment(a_i, b_i) **do**

  $p = \mathbf{M}a_i$

  $q = \mathbf{M}b_i$

  drawline(x_p, y_p, x_q, y_q)

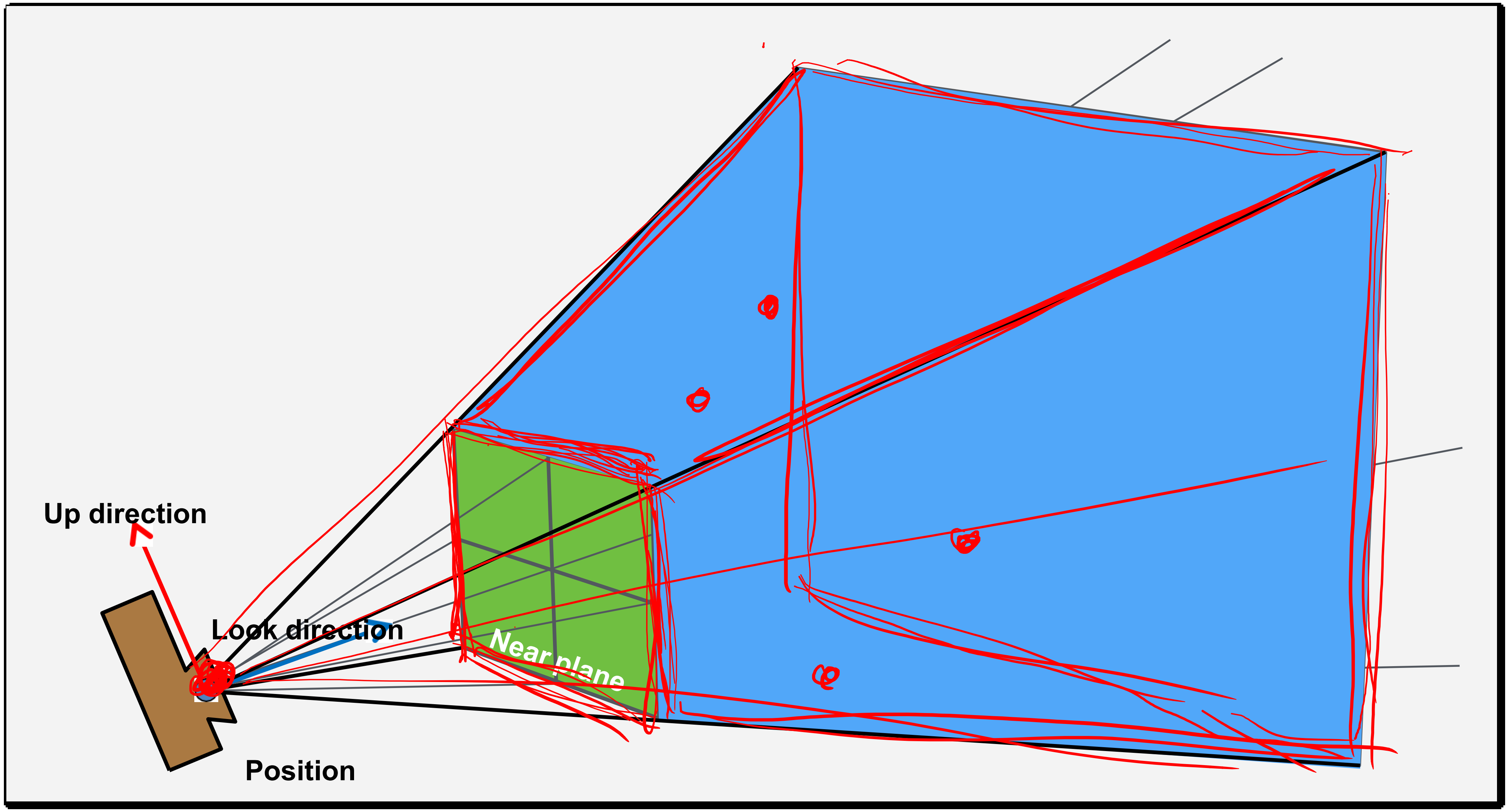
$$\begin{bmatrix} x_{pixel} \\ y_{pixel} \\ z_{canonical} \\ 1 \end{bmatrix} = M_{vp} M_{orth} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathrm{M}_{vp} = \begin{bmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{M}_{orth} = \mathbf{ST} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & -\frac{n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

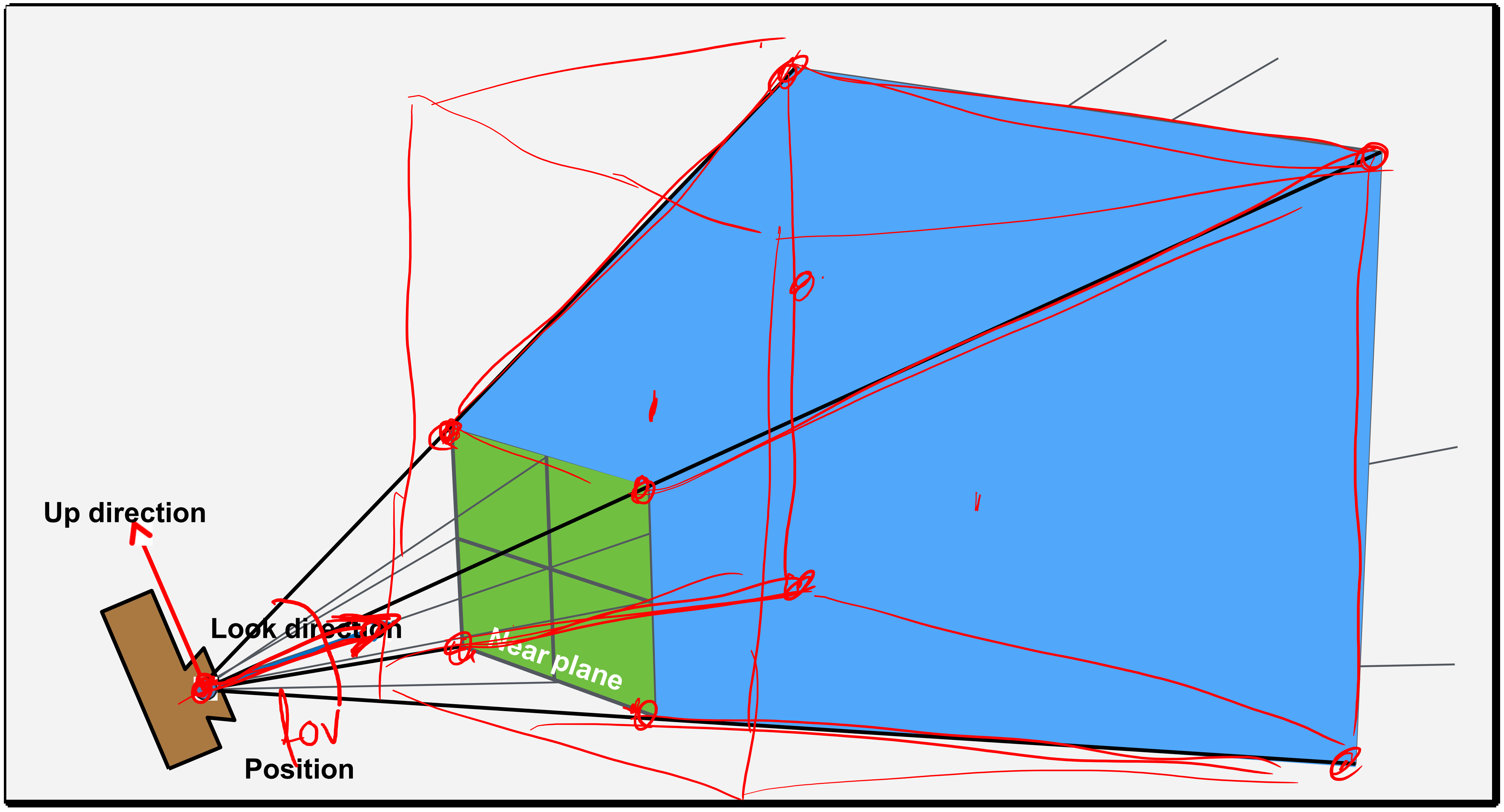
Camera de vizualizare 3D

- Poziție
- Orientare
- Câmp de vizualizare (Field of view)
- Depth of field
- Distanță focală
- Tip de proiecție

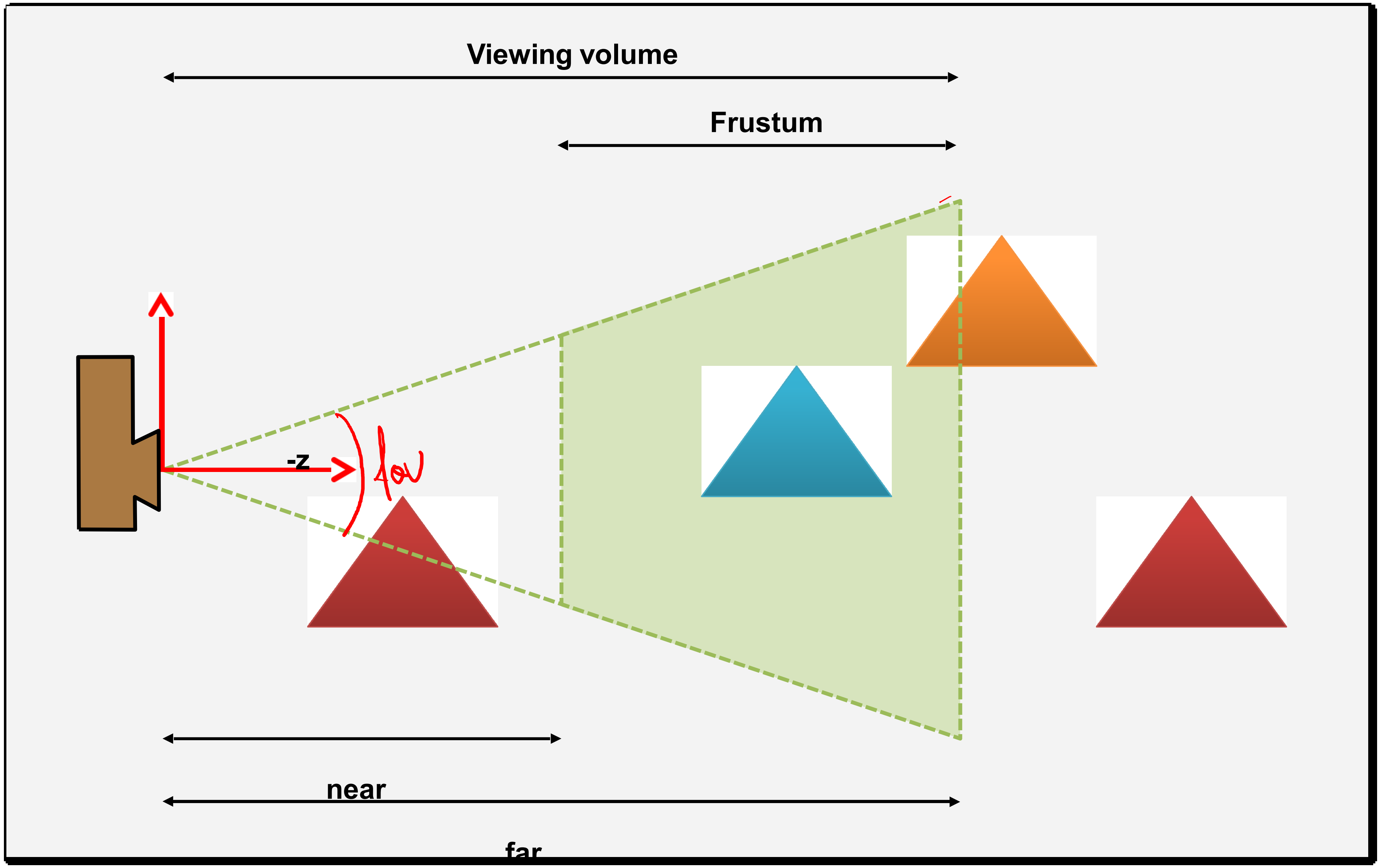


Camera de vizualizare 3D

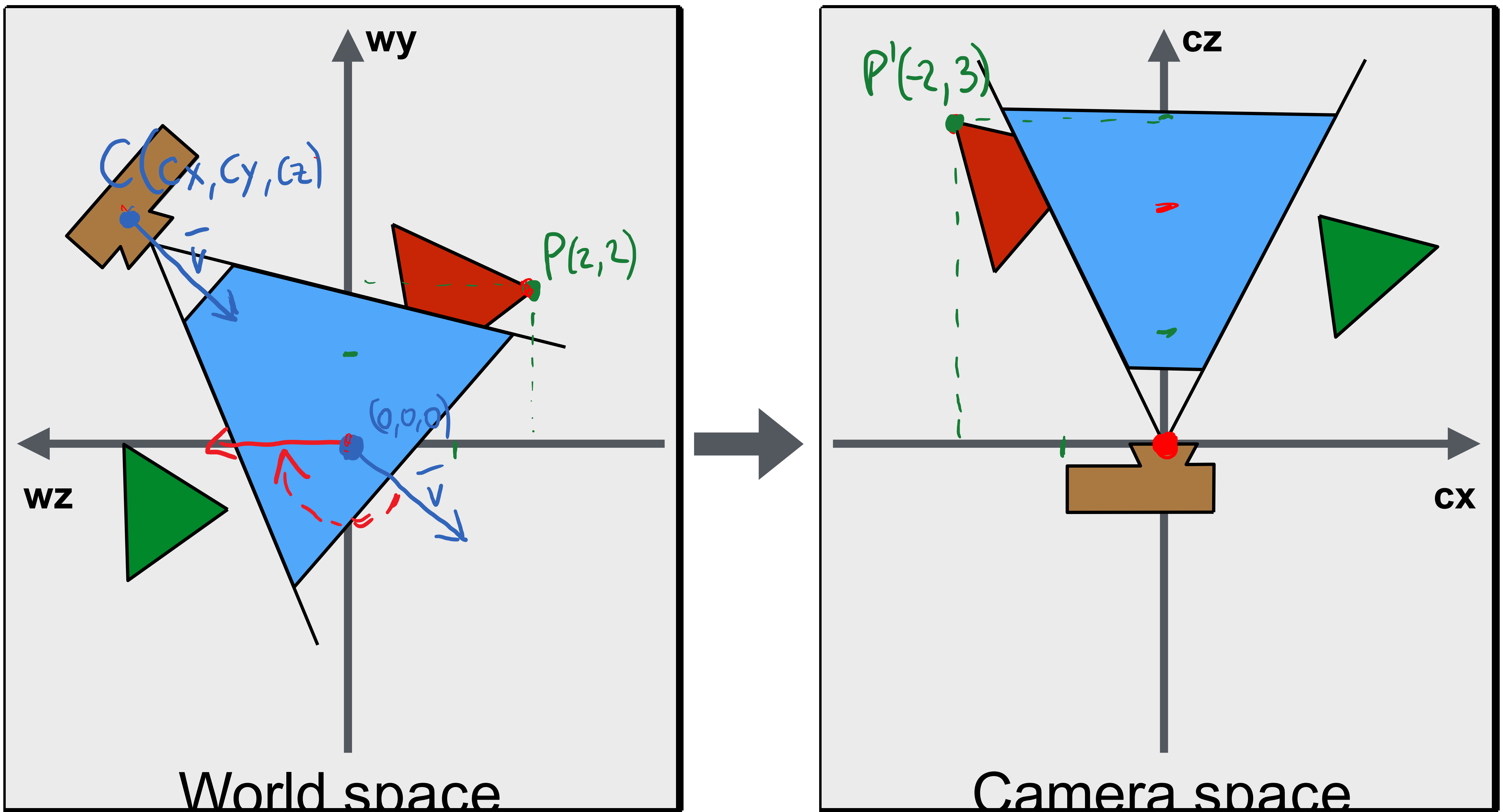
- Poziție
- Orientare
- Câmp de vizualizare (Field of view)
- Depth of field
- Distanță focală
- Tip de proiecție



Volumul de vizualizare



Transformarea de vizualizare



Translatie : (Cx, Cy, Cz) ⇒ (0,0,0)
Rotatie : aliniere $\bar{v}$ la axa Oz

} ⇒ Aplicate pe puncte. Ex: M = R · T

M · P = P'

Transformarea de cameră

Specificăm poziția și orientarea camerei prin:

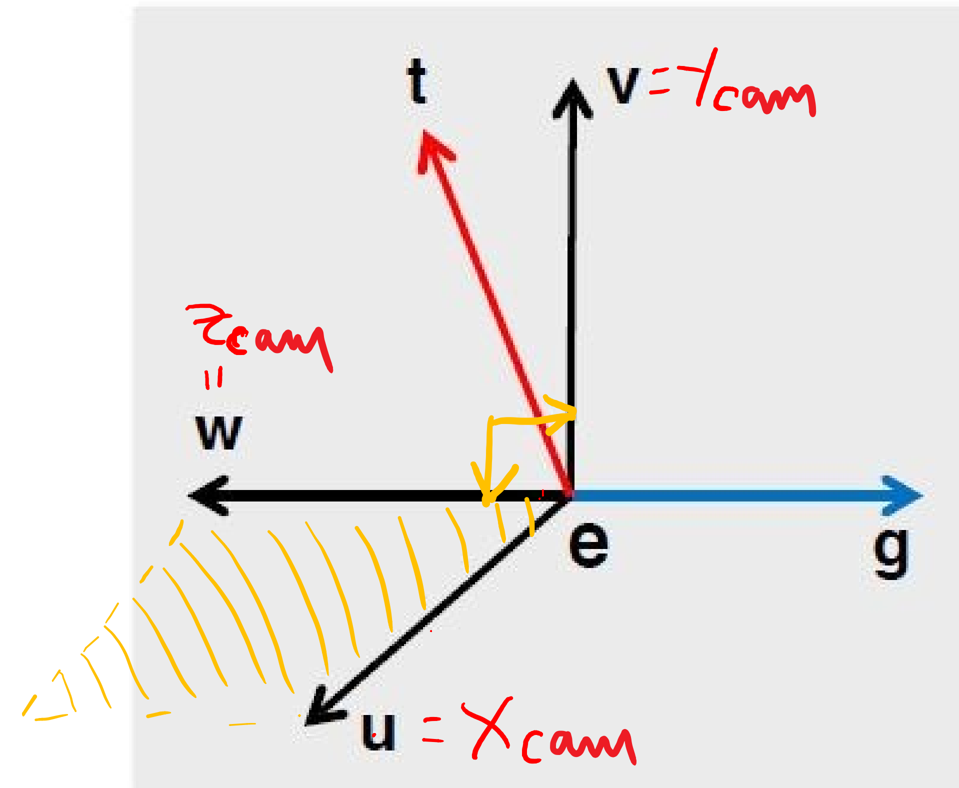
- eye position **e**
- gaze direction **g**
- view-up vector **t**

Calculăm:

$$\mathbf{w} = -\frac{\mathbf{g}}{\|\mathbf{g}\|}$$

$$\mathbf{u} = \frac{\mathbf{t} \times \mathbf{w}}{\|\mathbf{t} \times \mathbf{w}\|}$$

$$\mathbf{v} = \mathbf{w} \times \mathbf{u}$$



Modificarea sistemului de coordonate

Rotație pentru a alinia vectorii **u**, **v**, **w** cu vectorii **x**, **y**, **z**

Considerăm sistemul de vectori bază

$$\mathbf{e}_1 = \overset{x}{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}} \qquad \mathbf{e}_2 = \overset{y}{\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}} \qquad \mathbf{e}_3 = \overset{z}{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}$$

Trebuie să găsim matricea de rotație care rotește **u** peste $\mathbf{e}_1$, **v** peste $\mathbf{e}_2$ și **w** peste $\mathbf{e}_3$

$$\mathbf{M}_r\mathbf{u} = \mathbf{e}_1$$
$$\mathbf{M}_r\mathbf{v} = \mathbf{e}_2$$
$$\mathbf{M}_r\mathbf{w} = \mathbf{e}_3$$

Modificarea sistemului de coordonate

Multiplicăm cu $\mathbf{M_r^{-1}}$

$x, y, z \Rightarrow u, v, w$

$$\mathbf{u} = \mathbf{M}_r^{-1}\mathbf{e}_1$$

$$\mathbf{v} = \mathbf{M}_r^{-1}\mathbf{e}_2$$

$$\mathbf{w} = \mathbf{M}_r^{-1}\mathbf{e}_3$$

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

Probă: $M_r^{-1} \cdot x = u$

Deoarece sistemul generat de vectorii bază **u**, **v**, **w** este ortonormat inversa matricii este transpusa ei (în coordonate omogene)

$u \cdot v = u_x \cdot v_x + u_y \cdot v_y + u_z \cdot v_z = 0$

!!!

$u, v, w \rightarrow$ perpendiculari

$$\mathbf{M}_r = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_x \\ u_y \\ u_z \\ 1 \end{bmatrix} = \begin{bmatrix} u_x^2 + u_y^2 + u_z^2 \\ \cancel{u_x v_x + u_y v_y + u_z v_z} \\ \cancel{w_x u_x + w_y u_y + w_z u_z} \\ 1 \end{bmatrix}$$

Transformarea de cameră

Transformarea dintr-un volum de vizualizare arbitrar în volumul de vizualizare canonic

- translația camerei în origine
- transformarea în sistemul de coordonatele a camerei

R -> aliniere axe

transl. în origine

$$\mathbf{M}_{cam} = \begin{bmatrix} u_x & u_y & u_z & 0 \\ v_x & v_y & v_z & 0 \\ w_x & w_y & w_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -e_x \\ 0 & 1 & 0 & -e_y \\ 0 & 0 & 1 & -e_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

e = eye position = camera position

Transformarea de cameră

construct $\mathbf{M}_{vp}$ -> viewport 3

construct $\mathbf{M}_{orth}$ -> proiecție 2

construct $\mathbf{M}_{cam}$ -> camera 1

$\mathbf{M} = \mathbf{M}_{vp} \mathbf{M}_{orth} \mathbf{M}_{cam}$

for each line segment(a_i,b_i) **do**

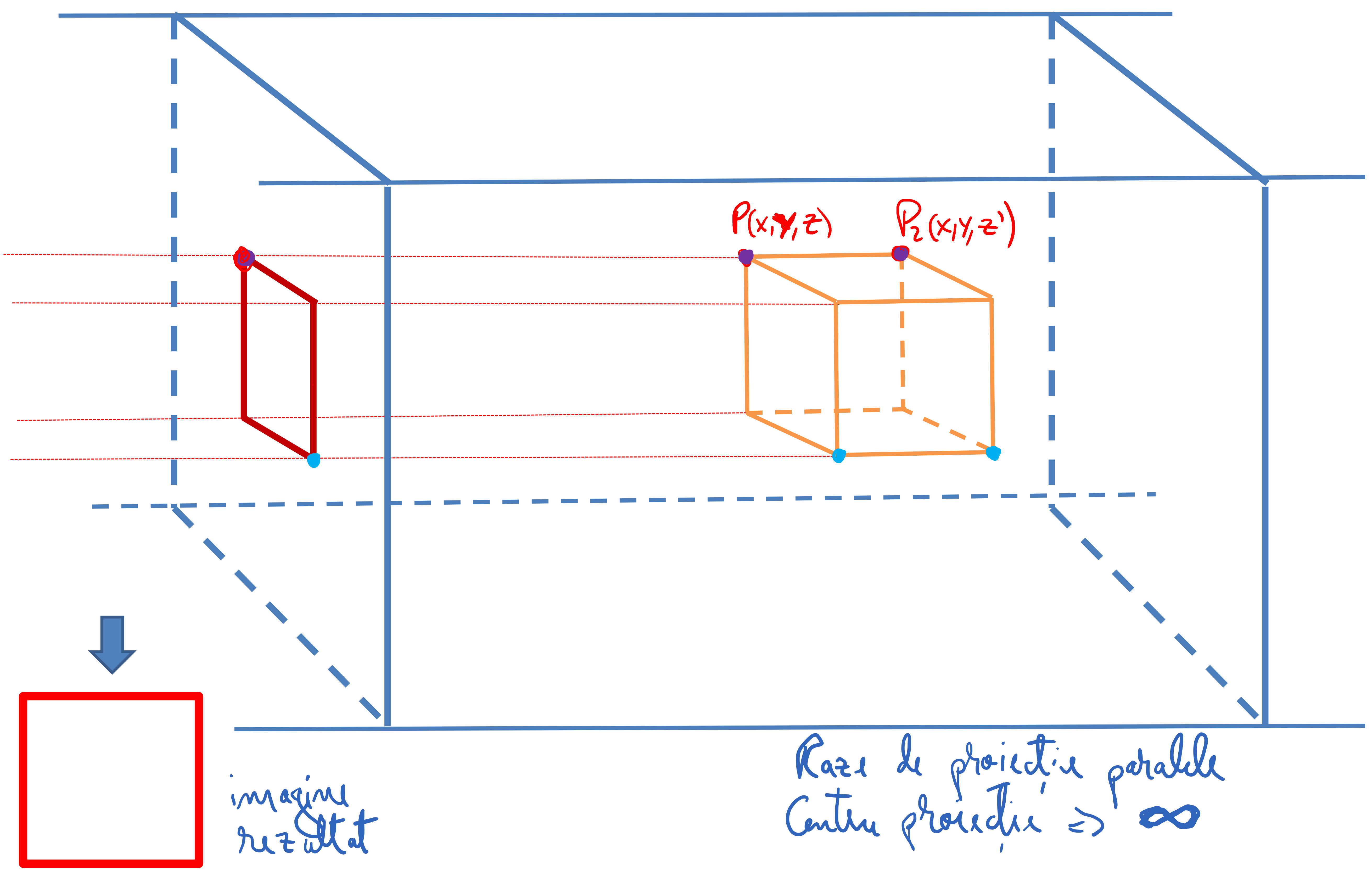
  $p = \mathbf{M} a_i$

  $q = \mathbf{M} b_i$

  drawline(x_p,y_p,x_q,y_q)

$$\begin{bmatrix} x_{pixel} \\ y_{pixel} \\ z_{canonical} \\ 1 \end{bmatrix} = \mathbf{M}_{vp} \mathbf{M}_{orth} \mathbf{M}_{cam} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Proiecție paralelă

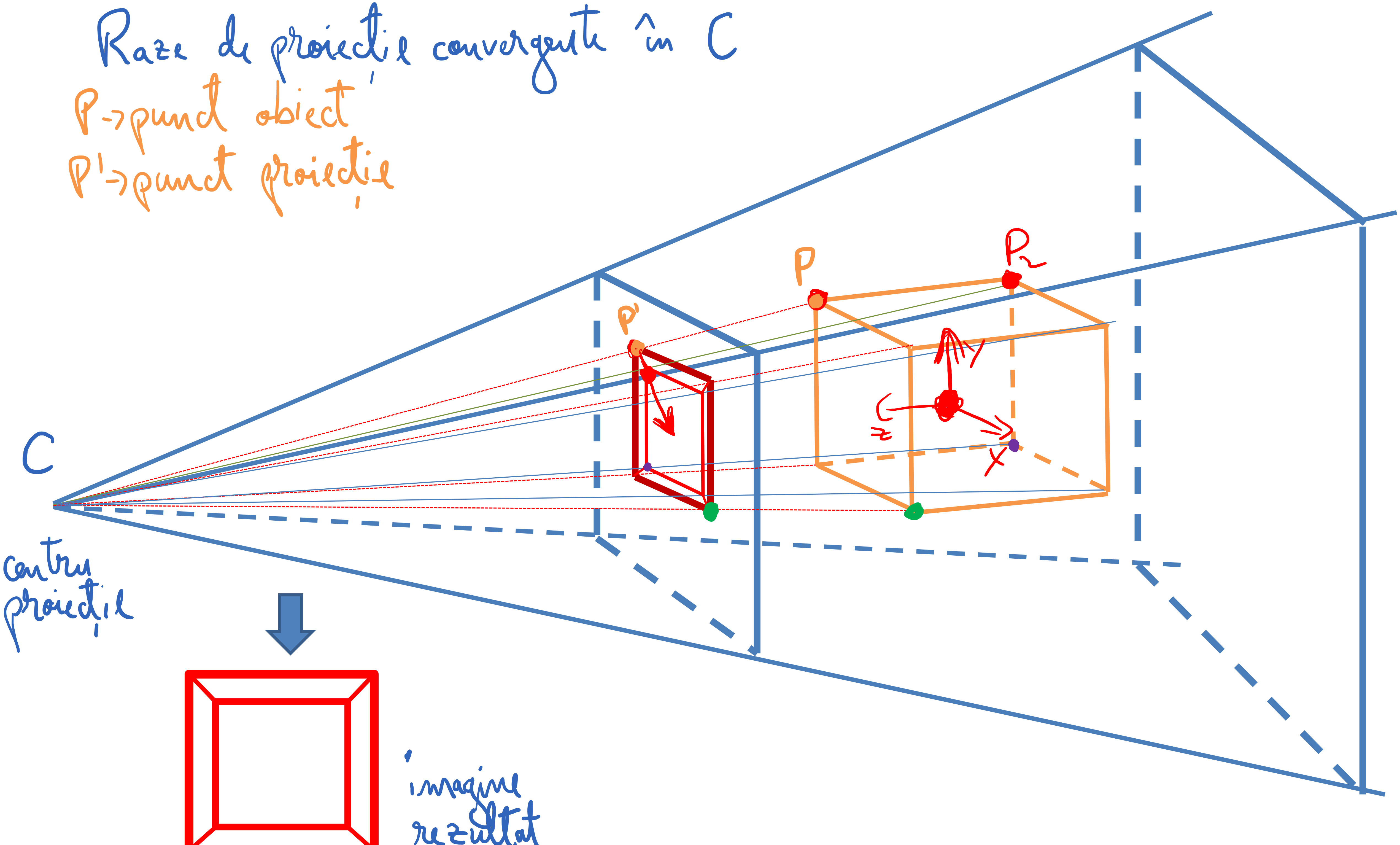


Proiecție perspectivă

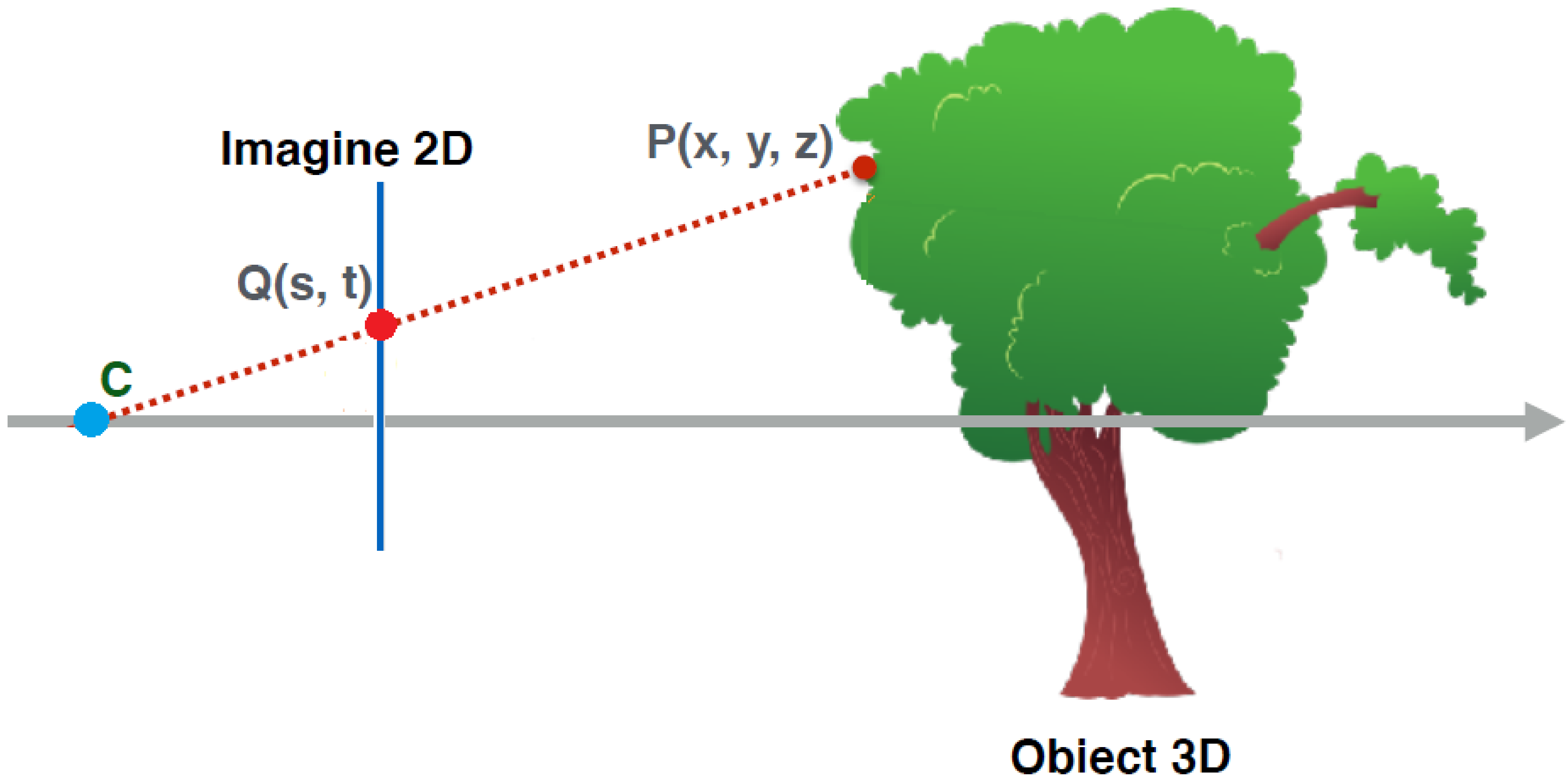
Raze de proiecție convergente în C

P -> punct obiect

P' -> punct proiecție

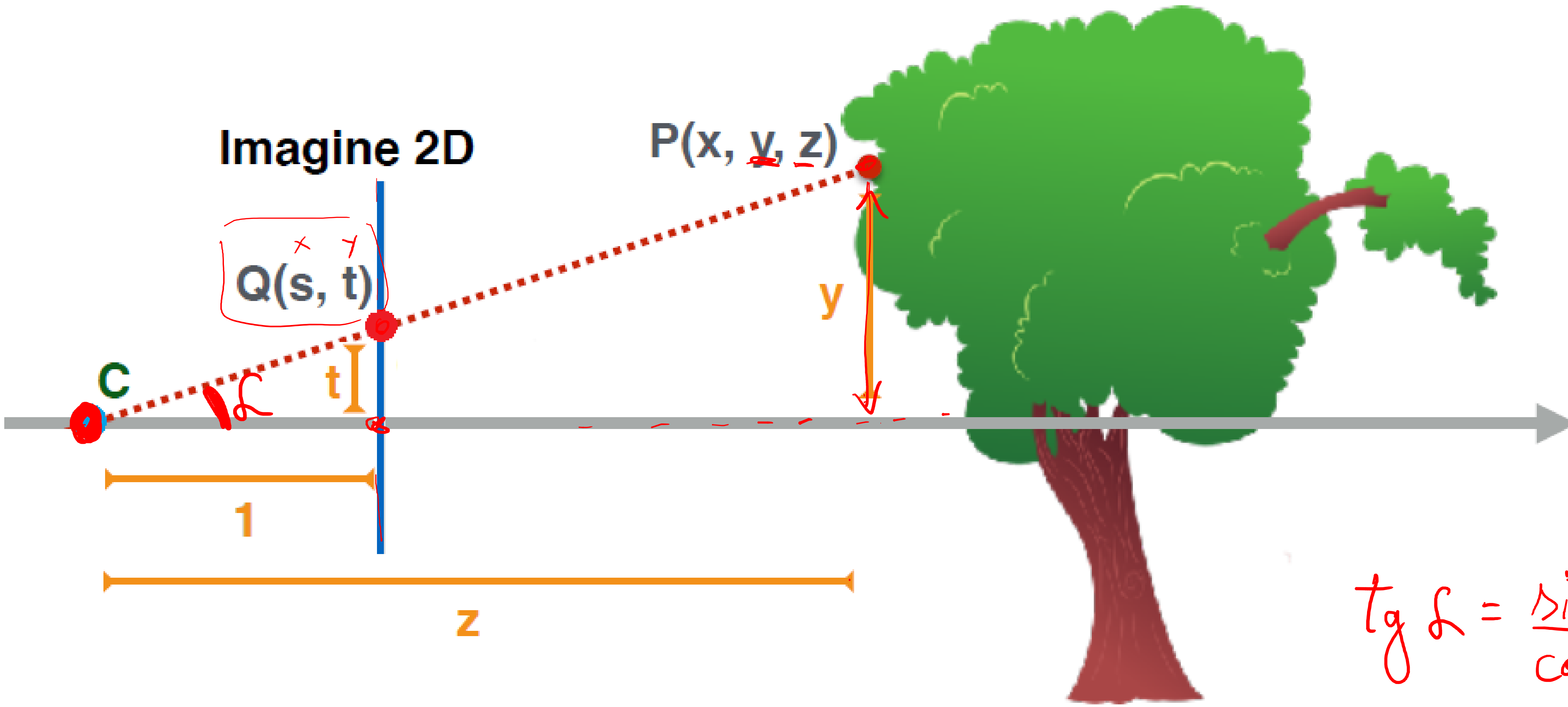


Proiecție perspectivă



Proiecție perspectivă

Consideram distanta pana la planul de proiectie = 1



Datorită **triunghiurilor asemenea** avem

Obiect 3D

$$\boxed{\frac{t}{1} = \frac{y}{z}} \qquad \frac{s}{1} = \frac{x}{z}$$

$\underset{(y_Q)}{t} = \frac{1 \cdot y}{z}$ $\qquad \underset{(x_Q)}{s} = \frac{1 \cdot x}{z}$

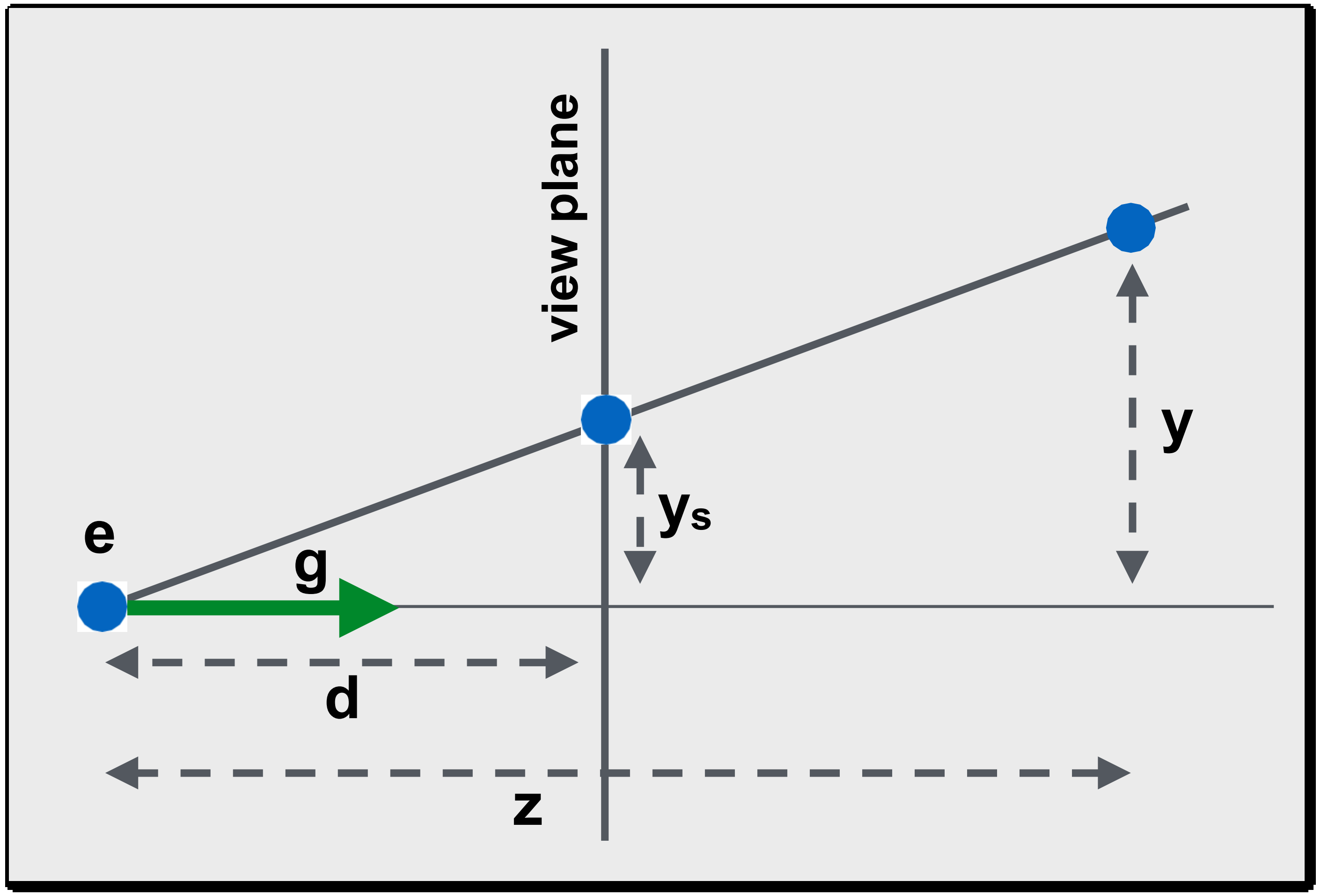
$tg\,\alpha = \frac{\sin}{\cos} = \frac{O}{A}$

$tg\,\alpha = \frac{t}{1}$

$tg\,\alpha = \frac{y}{z}$

Coordonatele punctului Q sunt $\boxed{\textbf{Q(x/z, y/z)}}$

Transformarea de proiecție perspectivă



$$x_s = \frac{d}{z}x \qquad y_s = \frac{d}{z}y$$

Transformări generale

Vectorul **[x y z w]ᵀ** definit în coordonate omogene reprezintă punctul **(x/w, y/w, z/w)**

$$x_{prj} = \frac{d \cdot x}{z}$$
$$y_{prj} = \frac{d \cdot y}{z}$$

Transformările liniare permit calcularea unor expresii de forma:

$$x' = ax + by + cz$$

$$\begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ax+by+cz \\ y \\ z \end{bmatrix}$$

Transformările afine extind expresiile la forma:

$$x' = ax + by + cz + d$$

$$\begin{bmatrix} a & b & c & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax+by+cz+d \\ y \\ z \\ 1 \end{bmatrix}$$

Considerând **w** numitorul unui fracții expresiile posibile se extind la:

$$x' = \frac{ax + by + cz + d}{ex + fy + gz + h}$$

$$\begin{bmatrix} a & b & c & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ e & f & g & h \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax+by+cz+d \\ y \\ z \\ ex+fy+gz+h \end{bmatrix}$$

← împărțire la ultimul termen (w)

Transformare de proiecție

$$\begin{array}{c} \\ \begin{bmatrix} \widetilde{x} \\ \widetilde{y} \\ \widetilde{z} \\ \widetilde{w} \end{bmatrix} \end{array} = \begin{array}{c} \\ \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_3 & b_3 & c_3 & d_3 \\ e & f & g & h \end{bmatrix} \end{array} \begin{array}{c} P \\ \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \end{array}$$

$$\boxed{\begin{aligned} \widetilde{x} &= a_1 x + b_1 \cdot y + c_1 \cdot z + d_1 \\ \widetilde{y} &= \text{———} \quad '' \quad \text{———} \\ \widetilde{z} &= \text{———} \quad '' \quad \text{———} \end{aligned}}$$

$$\widetilde{w} = \underbrace{e}_{0} \cdot x + \underbrace{f}_{0} \cdot y + \underbrace{g}_{\substack{\Downarrow \\ g=1}} z + \underbrace{h}_{0}$$

$$(x', y', z') = (\widetilde{x}/\widetilde{w}, \widetilde{y}/\widetilde{w}, \widetilde{z}/\widetilde{w}) \Rightarrow \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

$$x' = \frac{d \cdot x}{1 \cdot z} \qquad y' = \frac{d \cdot y}{1 \cdot z}$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$w \qquad\qquad w$$

Pentru calculul corect al x', y' avem nevoie de $w = 1 \cdot z$

$$\Downarrow$$

$$w = 0 \cdot x + 0 \cdot y + 1 \cdot z + 0$$

Transformare de proiecție

$$\mathbf{M}_p = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & d & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

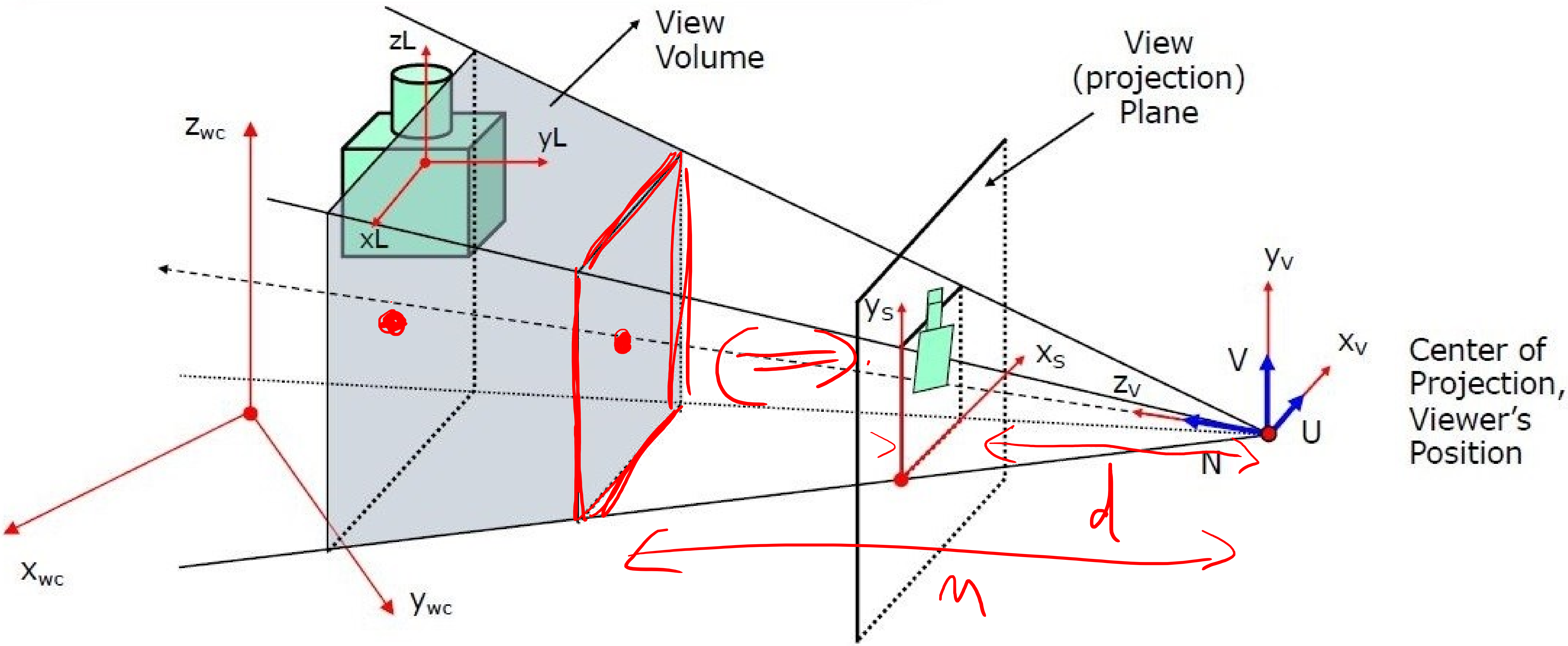
$$\boxed{\begin{aligned} x_s &= \frac{d}{z}x \\ y_s &= \frac{d}{z}y \end{aligned}}$$

$$\mathbf{P} = \mathbf{M}_p \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \underline{xd} \\ \underline{yd} \\ \underline{zd} \\ \boxed{z} \end{bmatrix} \xrightarrow{\text{After perspective divide}} \begin{bmatrix} \frac{d}{z}x \\ \frac{d}{z}y \\ \boxed{d} \\ 1 \end{bmatrix} \begin{array}{l} \tilde{x} \\ \tilde{y} \\ \\ \\ \end{array}$$

$z = ?$

Avem nevoie de valorile originale pentru z pentru a separa obiectele în funcție de adâncime

Transformare de proiecție



Transformare de proiecție

$$\mathbf{M}_p = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & \boxed{a} & \boxed{b} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{P} = \mathbf{M}_p \begin{bmatrix} x \\ y \\ \boxed{z} \\ 1 \end{bmatrix} = \begin{bmatrix} xn \\ yn \\ \boxed{az + b} \\ \boxed{z} \end{bmatrix} \xrightarrow{\text{After perspective divide}} \begin{bmatrix} \frac{n}{z}x \\ \frac{n}{z}y \\ \boxed{a + \frac{b}{z}} \\ 1 \end{bmatrix}$$

z-ul pct rezultat după proiecție

Trebuie determinați **a** și **b** astfel încât:

- pentru **z = n**, **a + b/z = n**
- pentru **z = f**, **a + b/z = f**

a: $n + f$

b: $-fn$

Matricea de proiecție perspectivă

$$\mathbf{P} = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -fn \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$x_{proj} = \frac{n \cdot x}{z}$$

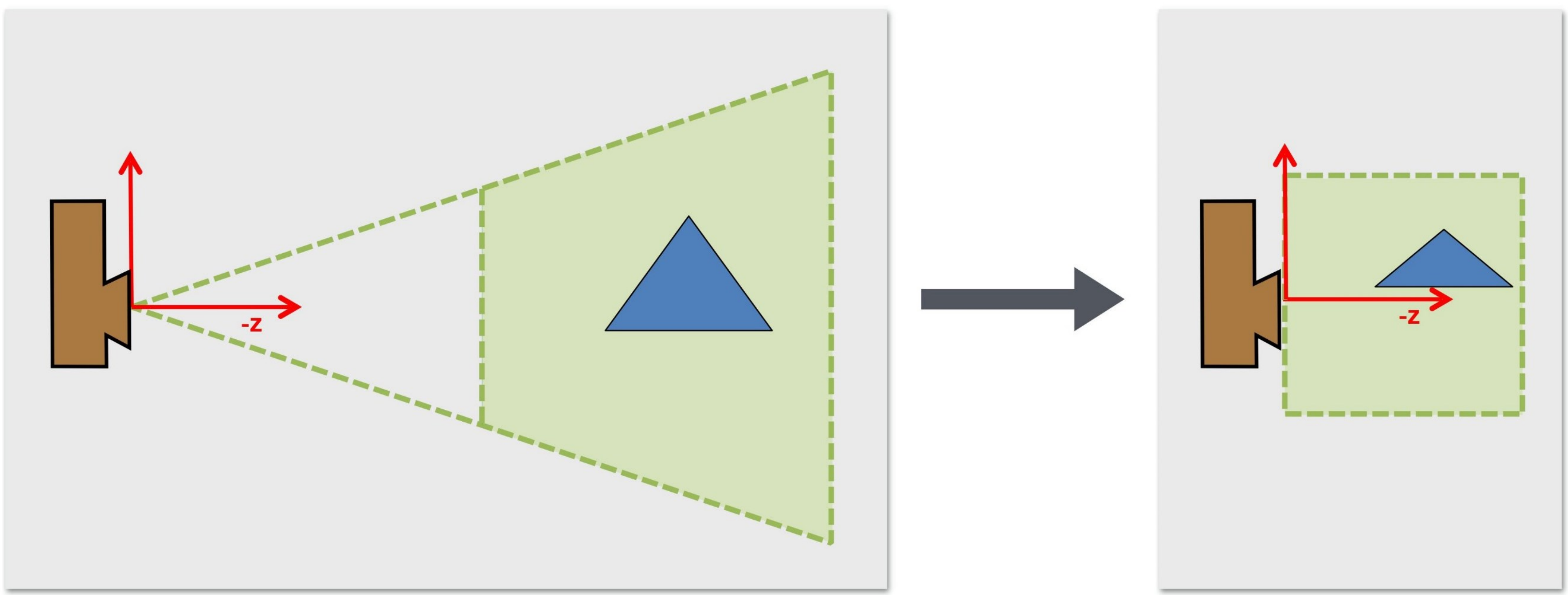
$$y_{proj} = \frac{n \cdot y}{z}$$

$$z_{proj} = z$$

$\Rightarrow$ păstrăm z original

Perspective divide

$$\begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -fn \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} nx \\ ny \\ (n+f)z - nf \\ z \end{bmatrix} \xrightarrow{\text{After perspective divide}} \begin{bmatrix} \frac{n}{z}x \\ \frac{n}{z}y \\ n+f-\frac{nf}{z} \\ 1 \end{bmatrix}$$



Matricea de proiecție perspectivă

Mapează volumul de vizualizare perspectivă (modelat ca un trunchi de piramidă) la volumul de vizualizare ortografic (paralelipiped cu laturile aliniate cu axele sistemului de coordonate)

$$\mathbf{P} = \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -fn \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Proiecția perspectivă

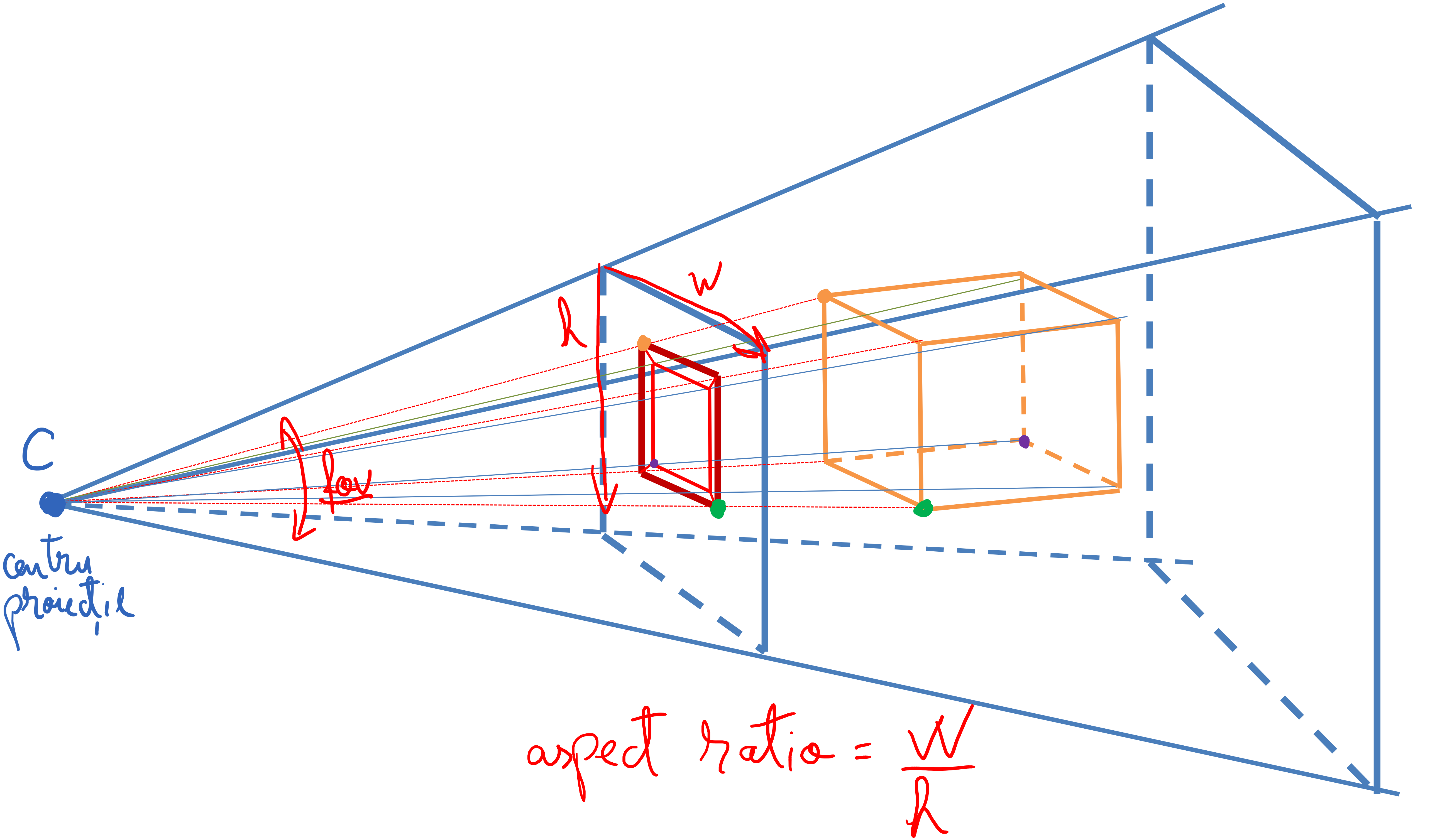
Matricea de proiecție perspectivă

$$\mathbf{M}_{\text{per}} = \mathbf{M}_{\text{orth}}\mathbf{P}$$

$$\mathbf{M}_{per} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{l+r}{l-r} & 0 \\ 0 & \frac{2n}{t-b} & \frac{b+t}{b-t} & 0 \\ 0 & 0 & \frac{f+n}{n-f} & \frac{2fn}{f-n} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

construct $\mathbf{M}_{\text{vp}}$
construct $\mathbf{M}_{\text{per}}$
construct $\mathbf{M}_{\text{cam}}$
$\mathbf{M} = \mathbf{M}_{\text{vp}}\mathbf{M}_{\text{per}}\mathbf{M}_{\text{cam}}$
for each line segment(a_i,b_i) **do**
  $p = \mathbf{M}a_i$
  $q = \mathbf{M}b_i$
  drawline$(x_p/w_p, y_p/w_p, x_q/w_q, y_q/w_q)$

Proiecție perspectivă

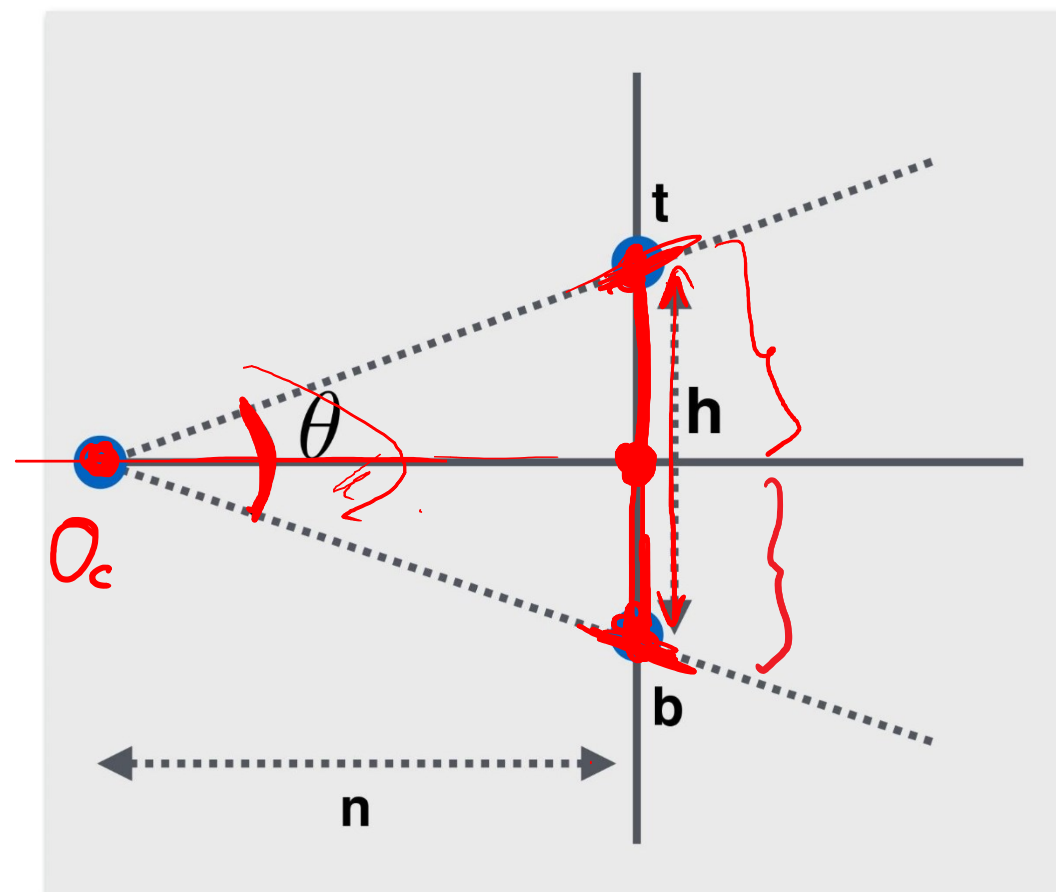


Câmp de vizualizare

Câmp de vizualizare (field of view - FOV): orizontal sau vertical

FOV pentru jocuri pe console aproximativ 60°

FOV pentru jocuri pe PC aproximativ 90-100°



$$\mathbf{M}_{per} = \begin{bmatrix} \boxed{\frac{1}{aspect \cdot \tan\frac{\theta}{2}}} & 0 & 0 & 0 \\ 0 & \boxed{\frac{1}{\tan\frac{\theta}{2}}} & 0 & 0 \\ 0 & 0 & \boxed{\frac{f+n}{n-f}} & \boxed{\frac{2fn}{f-n}} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\boxed{\tan\frac{\theta}{2} = \frac{h}{2n}}$$

$$\boxed{aspect = \frac{width}{height}} = \boxed{\frac{r-l}{t-b}}$$

$$\boxed{\frac{2n}{r-l} = \frac{2n}{aspect\cdot(t-b)} = \frac{2n}{aspect\cdot h} = \frac{\cancel{2n}}{aspect\cdot\tan\frac{\theta}{2}\cdot\cancel{2n}}} \quad \boxed{\frac{2n}{t-b} = \frac{1}{\tan\frac{\theta}{2}}} \quad \ldots$$

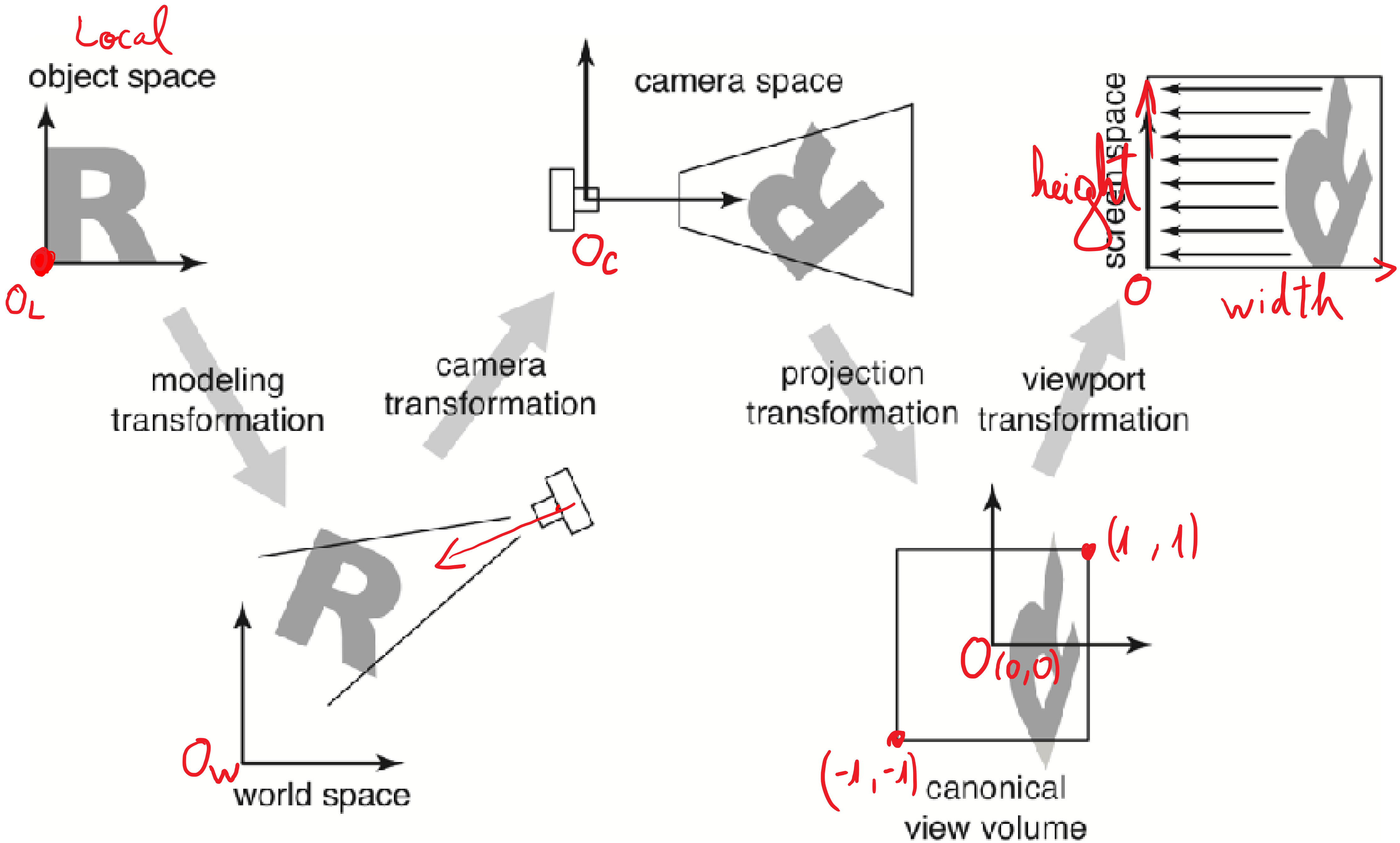
Câmp de vizualizare

FOV: 73.15^{o}

FOV:106.15^{o}

[http://pyroteq.com/wp-content/uploads/2014/07/farcry_3_fov_comparison.jpg]

Secvența de tranformare



[Peter Shirley et al. Fundamentals of Computer Graphics]

Sisteme de coordonate