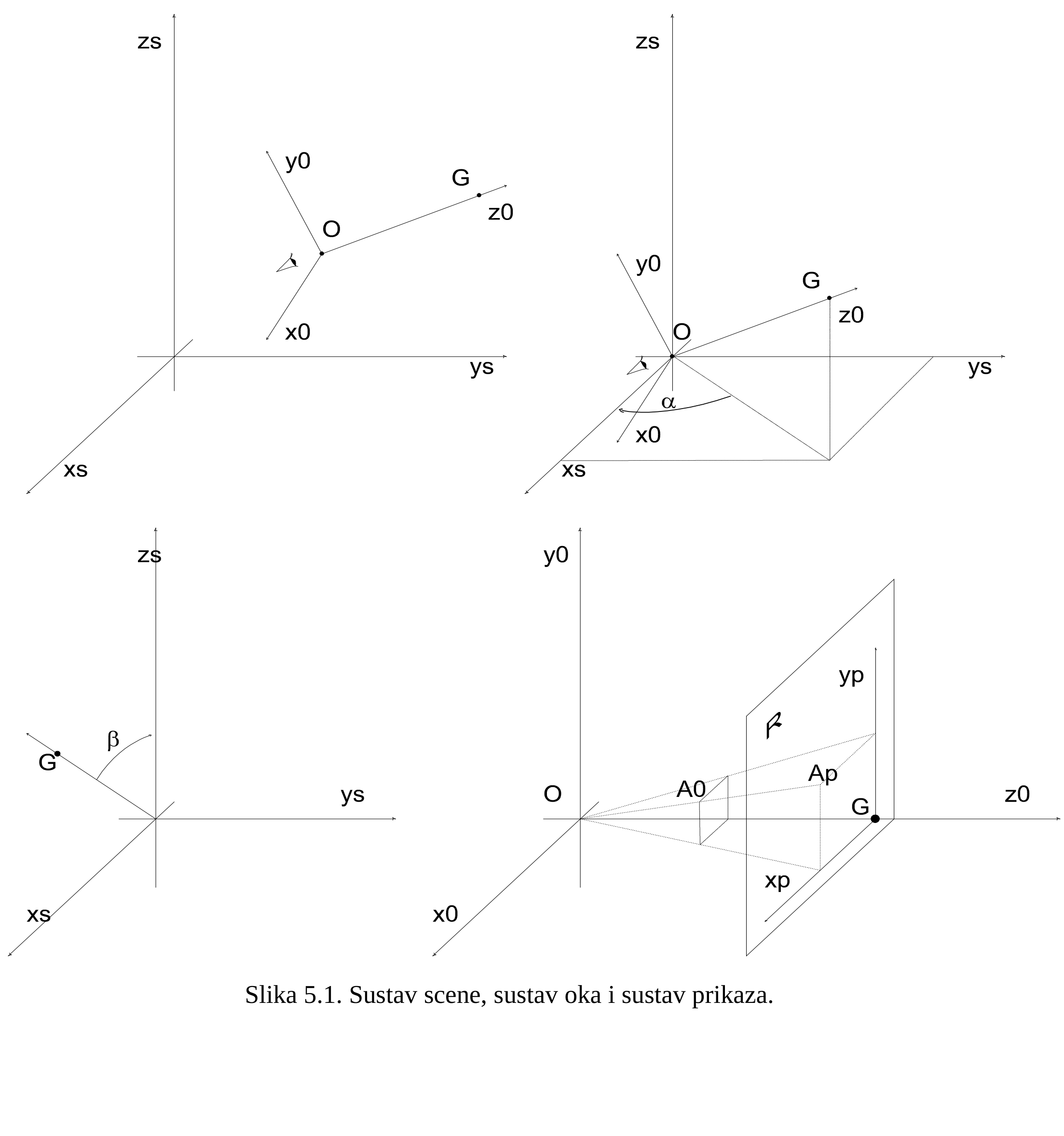
5. Transformacija pogleda i perspektivna projekcija

Sustav scene je trodimenzijski sustav, slika 5.1. Sustav oka je trodimenzijski sustav čija je *z* os upravljena u smjeru pogleda, tako da *z* os predstavlja dubinu slike. Sustav prikaza je dvodimenzijski sustav i smješten je u ravnini projekcije *R*.

Preslikavanje točaka iz sustava scene u sustav oka naziva se transformacija pogleda. Projekcija točaka iz sustava oka u sustav prikaza može se načiniti kao paralelna ili perspektivna projekcija.

Korištene oznake:

$x_S\ y_S\ z_S$ - sustav scene,
$x_O\ y_O\ z_O$ - sustav oka,
$x_P\ y_P$ - sustav prikaza,
O - očište, položaj promatrača u sustavu scene,
G - gledište, točka u sustavu scene u koju je usmjeren pogled,
H - udaljenost ravnine projekcije od očišta, *H*=d(*O*,*G*),
R - ravnina projekcije, točka *G* leži u ravnini projekcije,



Slika 5.1. Sustav scene, sustav oka i sustav prikaza.

5.1. Transformacija pogleda

Za transformaciju pogleda treba odrediti matricu T tako da vrijedi

$$A_O = A_S T \tag{5.1}$$

Matrica T je sastavljena matrica od pet matrica elementarnih transformacija, to su:
T_1 - pomak ishodišta u točku O,
T_2 - rotacija za kut α oko z osi,
T_3 - rotacija za kut β oko y osi,
T_4 - rotacija za kut 90° oko z osi,
T_5 - promjena predznaka na x osi.

Točke $O\begin{pmatrix} x_o & y_o & z_o \end{pmatrix}$ i $G\begin{pmatrix} x_g & y_g & z_g \end{pmatrix}$ mjere se u sustavu scene. Koordinatama točke O određena je matrica T_1

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_o & -y_o & -z_o & 1 \end{bmatrix}. \tag{5.2}$$

Djelovanje T_1 na G daje

$$G_1 = GT_1 \text{ ili } \begin{aligned} x_{g1} &= x_g - x_o \\ y_{g1} &= y_g - y_o \\ z_{g1} &= z_g - z_o \end{aligned}$$

Matrica T_2 glasi

$$T_2 = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5.3}$$

pri čemu je

$$\sin\alpha = \frac{y_{g1}}{\sqrt{x_{g1}^2 + y_{g1}^2}}, \qquad \cos\alpha = \frac{x_{g1}}{\sqrt{x_{g1}^2 + y_{g1}^2}}$$

Djelovanje matrice T_2 na G_1 daje

$$G_2 = G_1 T_2 \textbf{ ili } \begin{aligned} x_{g2} &= \sqrt{x_{g1}^2 + y_{g1}^2} \\ y_{g2} &= 0 \\ z_{g2} &= z_{g1} \end{aligned} .$$

Matrica T_3 glasi

$$T_3 = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5.4}$$

pri čemu je

$$\sin\beta = \frac{x_{g2}}{\sqrt{x_{g2}^2 + z_{g2}^2}}, \qquad \cos\beta = \frac{z_{g2}}{\sqrt{x_{g2}^2 + z_{g2}^2}}.$$

Djelovanje matrice T_3 na G_2 daje

$$G_3 = G_2 T_3 \text{ ili } \begin{aligned} x_{g3} &= 0 \\ y_{g3} &= 0 \\ z_{g3} &= \sqrt{x_{g2}^2 + z_{g2}^2} \end{aligned}$$

Nakon ovih transformacija ostvareno je podudaranje z osi koordinatnog sustava oka z_o sa z osi koordinatnog sustava scene z_s. Potrebno je još ostvariti podudaranja x i y osi sustava oko sa sustavom scene. Koordinatni sustav na zaslonu obično je postavljen tako da je ishodište u gornjem lijevom uglu zaslona, x os postavljena je horizontalno prema desno y os vertikalno prema dolje a z os usmjerena je prema promatraču. Koordinatni sustav oka je prema tome lijevi koordinatni sustav, a sustav scene desni. Potrebne matrice T_4 i T_5 za ostvarivanje podudaranja x i y osi sustava oka sa sustavom scene glase

$$T_4 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad T_5 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.5}$$

Matrica T je umnožak

$$T = T_1 T_2 T_3 T_4 T_5 . \tag{5.6}$$

Drugi način određivanja matrice rotacije

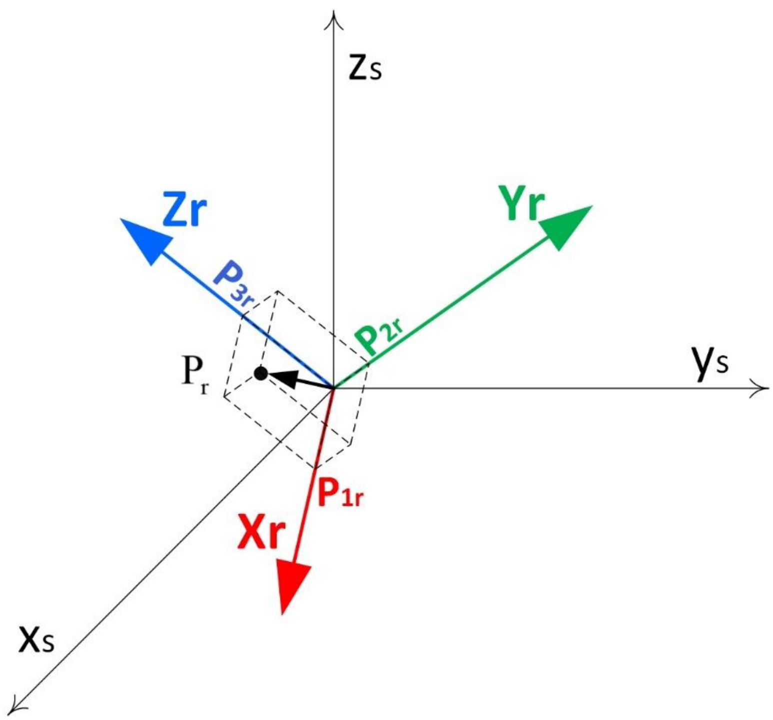
Matricu rotacije možemo izračunati i na drugačiji način. Prisjetimo se da je cilj matrice rotacije jednostavno zarotirati čitav koordinatni sustav, pa tako i sve točke u našem originalnom koordinatnom sustavu. To bi značilo da su se nakon rotacije koordinate svih točaka, gledano iz originalnog koordinatnog sustava, promijenile. Upravo nam matrica rotacije služi kako bismo odredili koordinate rotiranih točaka gledano iz originalnog koordinatnog sustava. No postavlja se pitanje kako jednostavno možemo odrediti tu matricu rotacije. Prisjetimo se da se svaka točka u prostoru može zapisati kao linearna kombinacija koordinatnih osi sustava, pa se tako točka $P_r = (p_{1r}, p_{2r}, p_{3r})$ može prikazati linearnom kombinacijom vektora osi rotiranog sustava $P_r = p_{1r}\overrightarrow{x_r} + p_{2r}\overrightarrow{y_r} + p_{3r}\overrightarrow{z_r}$, kao što je prikazano i na slici 5.2. Upravo ovo svojstvo možemo iskoristiti i za određivanje koordinata neke točke u koordinatnom sustavu nakon rotacije. Neka nam p_{1r}, p_{2r} i p_{3r} predstavljaju koordinate točke u rotiranom koordinatnom sustavu, a $\overrightarrow{x_r}, \overrightarrow{y_r}$ i $\overrightarrow{z_r}$ (označeni crvenom, zelenom i plavom bojom na slici 5.2) vektore koji predstavljaju osi tog rotiranog koordinatnog sustava gledano iz originalnog koordinatnog sustava (pri čemu su primjerice koordinate vektora $\overrightarrow{x_r}$ označene kao x_{1r}, x_{2r} i x_{3r}). Na temelju tih informacija možemo koordinate neke točke u originalnom koordinatnom sustavu odrediti na način da se za svaki od tih vektora pomaknemo za odgovarajuće iznose koji su određeni vrijednostima koordinata te točke gledano iz rotiranog koordinatnog sustava. Dakle, zapravo se radi linearna kombicija osi rotiranog koordinatnog sustava, kao bi odredili koordinate točke u originalnom koordinatnom sustavu, Čitav izraz onda možemo zapisati kao:

$$[p_{1o} \quad p_{2o} \quad p_{3o}] = [p_{1r} \quad p_{2r} \quad p_{3r}] \begin{bmatrix} x_{1r} & x_{2r} & x_{3r} \\ y_{1r} & y_{2r} & y_{3r} \\ z_{1r} & z_{2r} & z_{3r} \end{bmatrix}, \tag{5.7}$$

odnosno kraće kao

$$\mathbf{P}_{\boldsymbol{o}} = \mathbf{P}_{\boldsymbol{r}} * \mathbf{R} \tag{5.8}$$

Pri čemu $\mathbf{P_r}$ označava koordinate točke u našem rotiranom koordinatnom sustavu, $\mathbf{R}$ matricu rotacije, a $\mathbf{P_o}$ koordinate rotirane točke gledano iz originalnog koordinatnog sustava.



Slika 5.2. Prikaz vektora koji čine osi rotiranog koordinatnog sustava gledano iz originalnog koordinatnog sustava

Vidimo da je matrica rotacije zapravo jednaka vektorima koji predstavljaju osi rotiranog koordinatnog sustava. Dakle, ako znamo vektore koji predstavljaju osi rotiranog koordinatnog sustava, onda zapravo veoma lagano možemo odrediti matricu rotacije. No potrebno je prvo odrediti osi rotiranog koordinatnog sustava.

Određivanje osi rotiranog koordinatnog sustava možemo napraviti korištenjem točaka očišta i gledišta. Prethodno smo definirali da se ishodište koordinatnog sustava oka nalazi u očištu. Os *z* definirat ćemo na način da je ta os usmjerena od očišta prema gledištu. Vektor koji će predstavljati *z* os u novom koordinatnom sustavu možemo onda izračunati kao

$$z = G - O \tag{5.9}$$

$$\hat{z} = \frac{z}{\|z\|} \tag{5.10}$$

Na taj način odredili smo vektor koji predstavlja *z* os novog koordinatnog sustava, te smo dodatno taj vektor normirali kako bi imao jediničnu duljinu. Sada je potrebno odrediti i preostale dvije osi. Nažalost, korištenjem samo očišta i gledišta to nije moguće napraviti, jer nam te dvije točke ne daju nikakvu informaciju o tome kako su preostale dvije osi orijentirane. Zbog toga nam je potrebna dodatna informacija koju ćemo koristiti za definiranje preostale dvije osi. U tu svrhu koristit ćemo dodatni vektor, pod nazivom *view-up* (v_{UP}) vektor, koji će određivati *y* os rotiranog koordinatnog sustava. Sada kada imamo dvije osi, treću možemo izračunati jednostavno preko vektorskog produkta preostalih dviju osi, jer znamo da ta os mora biti okomita na prethodne dvije. Dakle, os *x* ćemo izračunati na način da vektor koji predstavlja *z* os vektorski pomnožimo s *view-up* vektorom:

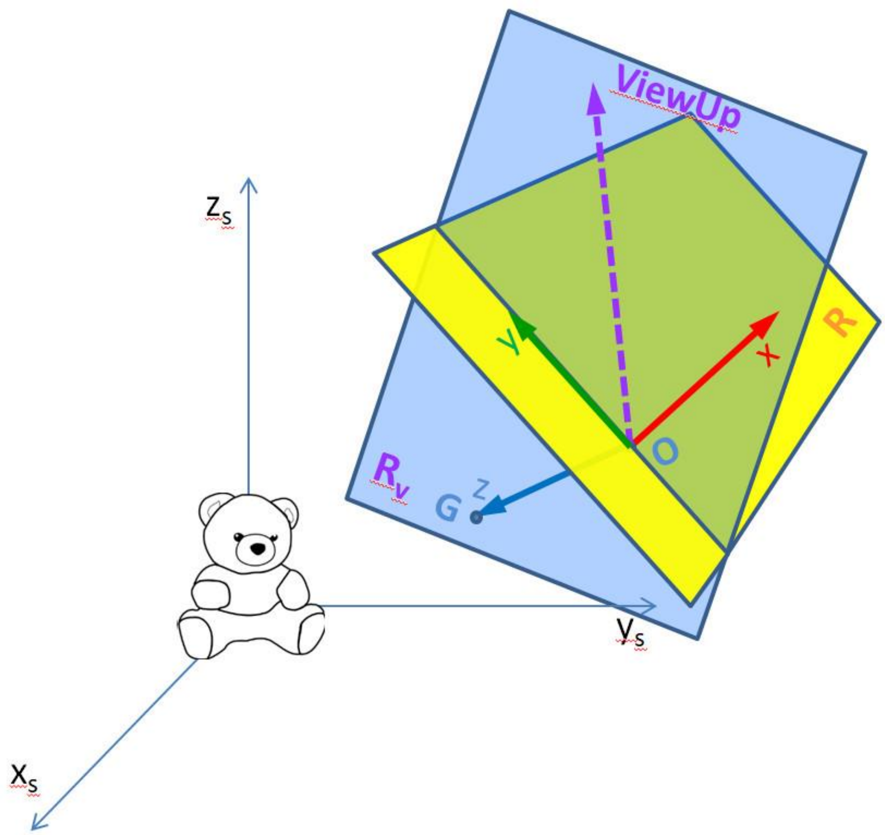
$$\hat{v}_{UP} = \frac{v_{UP}}{\|v_{UP}\|} \tag{5.11}$$

$$\hat{x} = \hat{z} \times \hat{v}_{UP} \tag{5.12}$$

No teško je za očekivati da će korisnik zadati vektor koji će uistinu biti okomit na vektor koji predstavlja *z* os. Zbog toga ćemo pretpostaviti da *view-up* vektor kojeg korisnik zadaje ne mora nužno biti okomit na vektor koji predstavlja *z* os, već da on samo definira smjer u kojem gleda *y* os. To znači da se *view-up* vektor ne mora nužno nalaziti u ravnini projekcije. Takva situacija je prikazana i na slici 5.3, na kojoj se vidi se kako se vektori *z*, *view-up* i *y* nalaze u ravnini R_V, dok se vektori *x* i *y* nalaze u ravnini projekcije *R*. Kako obično želimo da su sve osi našeg koordinatnog sustava međusobno okomite te da se i *y* os nalazi u ravnini projekcije zajedno s *x* osi, potrebno je na temelju zadanog *view-up* vektora odrediti *y* os. Jedan način kako to možemo ostvariti bi bio taj da jednostavno odredimo projekciju *view-up* vektora na ravninu projekcije *R*. No računanje tog reflektiranog vektora možemo izbjeći korištenjem jednog trika. Naime, kako smo već odredili dvije osi koordinatnog sustava, treću možemo opet izračunati preko njihovog vektorskog produkta i na taj način dobiti os koja je okomita na njih dvije, ali zadržava smjer sličan *view-up* vektoru. Dakle, *y* os se onda izračuna kao

$$\hat{y} = \hat{x} \times \hat{z} \tag{5.13}$$

Jedino na što moramo pripaziti jest da *view-up* vektor kojeg korisnik zada nije kolinearan s vektorom koji predstavlja *z* os, jer u tom slučaju ne bi mogli odrediti *x* i *y* osi.



Slika 5.3. Primjer *view-up* vektora koji se ne nalazi u ravnini projekcije

Na taj način izračunali smo sve vektore koji određuju osi novog koordinatnog sustava. Matricu rotacije onda možemo konstruirati pomoću koordinata vektora koji određuju novi koordinatni sustav (u homogenom prostoru):

$$R = \begin{bmatrix} \hat{x}_{1r} & \hat{x}_{2r} & \hat{x}_{3r} & 0 \\ \hat{y}_{1r} & \hat{y}_{2r} & \hat{y}_{3r} & 0 \\ \hat{z}_{1r} & \hat{z}_{2r} & \hat{z}_{3r} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad (5.14)$$

No ne smijemo zaboraviti pretpostavku s kojom smo krenuli, a to je da je globalni sustav bio zadan kao desni koordinatni sustav, dok je sustav scene zadan kao lijevi koordinatni sustav. Dakle, nakon što je obavljena rotacija, postignuta su poklapanja svih osi globalnog koordinatnog sustava i sustava scene, osim osi *z* koje iako leže na istom pravcu pokazuju u suprotnim smjerovima, što je posljedica činjenice da su koordinatni sustavi bili različite orijentacije. Zbog toga je nakon rotacije potrebno još napraviti zamjenu predznaka *z* osi korištenjem matrice transformacije T_Z, koja će jednostavno zrcaliti *z* os:

$$R_Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ukupnu matricu transformacije onda možemo dobiti umnoškom pojedinih matrica transformacija. Dakle, prvo je potrebno napraviti podudaranje ishodišta oba koordinatna sustava korištenjem matrice T_1, nakon toga je potrebno rotirati koordinatni sustav korištenjem matrice R kako bi se sve osi međusobno poklopile, i konačno korištenjem matrice T_Z potrebno je okrenuti predznak x osi iz razloga što su koordinatni sustavi bili zadani u suprotnim orijentacijama. Znaći, ukupna matrica transformacije je onda jednaka:

$$T = T_1 R T_Z \ . \tag{5.15}$$

Potrebno je napomenuti da OpenGL u pozadini radi veoma slično prethodno opisanom postupku. Jedina razlika je ta što OpenGL pretpostavlja da je i sustav oka desni koordinatni sustav, pa zapravo kod transformacije pogleda ne koristi matricu R_Z za okretanje z osi. No onda tijekom projekcije OpenGL okrene z os, što je zapravo ekvivalentno našoj transformaciji R_Z.

5.2. Perspektivna projekcija

Zadaća je odrediti matricu P koja će po zakonu perspektive projicirati točke iz sustava oka u ravninu projekcije, slika 5.1, odnosno u sustavu prikaza,

$$A_P = A_0 P \ . \tag{5.16}$$

Udaljenost ravnine projekcije od očišta je

$$H = \sqrt{\left(x_o - x_g\right)^2 + \left(y_o - y_g\right)^2 + \left(z_o - z_g\right)^2} = z_{g3} \, . \tag{5.17}$$

Iz sličnosti trokuta slijedi

$$x_p = \frac{x_0}{z_0} H, \quad y_p = \frac{y_0}{z_0} H \ . \tag{5.18}$$

Izraz 5.9 napisan u matričnom obliku glasi

$$A_p{}' = A_0 P$$

ili po koordinatama

$$\begin{bmatrix} x_p{}' & y_p{}' & z_p{}' & h_p{}' \end{bmatrix} = \begin{bmatrix} x_0 & y_0 & z_0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{H} \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\begin{aligned} x_p{}' &= x_0 \\ y_p{}' &= y_0 \\ z_p{}' &= 0 \\ h_p{}' &= \frac{z_0}{H} \end{aligned} \tag{5.19}$$

Iz 5.10 slijedi 5.9, što je povratak u nehomogeni prostor tj.

$$x_p = \frac{x_p'}{h_p'} = \frac{x_0}{z_0} H, \qquad y_p = \frac{y_p'}{h_p'} = \frac{y_0}{z_0} H.$$

5.3. Radni zadatak

Zadati poligon te načiniti transformaciju pogleda i perspektivnu projekciju.

1. Iz datoteke učitati koordinate očišta, gledišta i vrhova poligona u sustavu scene.
Gledište se obično zadaje u ishodištu scene G = (0 0 0) ili u središtu tijela (poligona). Očište je točka iz koje gledamo i ovisit će o veličini objekta. Ako su koordinate objekta u rasponu (-1, 1) očište može biti npr. O = (1 1 3). Moramo paziti da očište ne zadamo u unutrašnjosti objekta. Ako se rotacijska matrica računa korištenjem drugog opisanog postupka, onda je potrebno dodatno zadati i *view-up* vektor.

2. Odrediti matricu transformacije pogleda T po formuli 5.6 ili 5.15.

3. Odrediti matricu perspektivne projekcije P.

4. Načiniti transformaciju i projekciju zadanih vrhova poligona.

5. Iscrtati poligon.

6. Ponoviti korake 1-5. za tijelo iz prethodne vježbe.