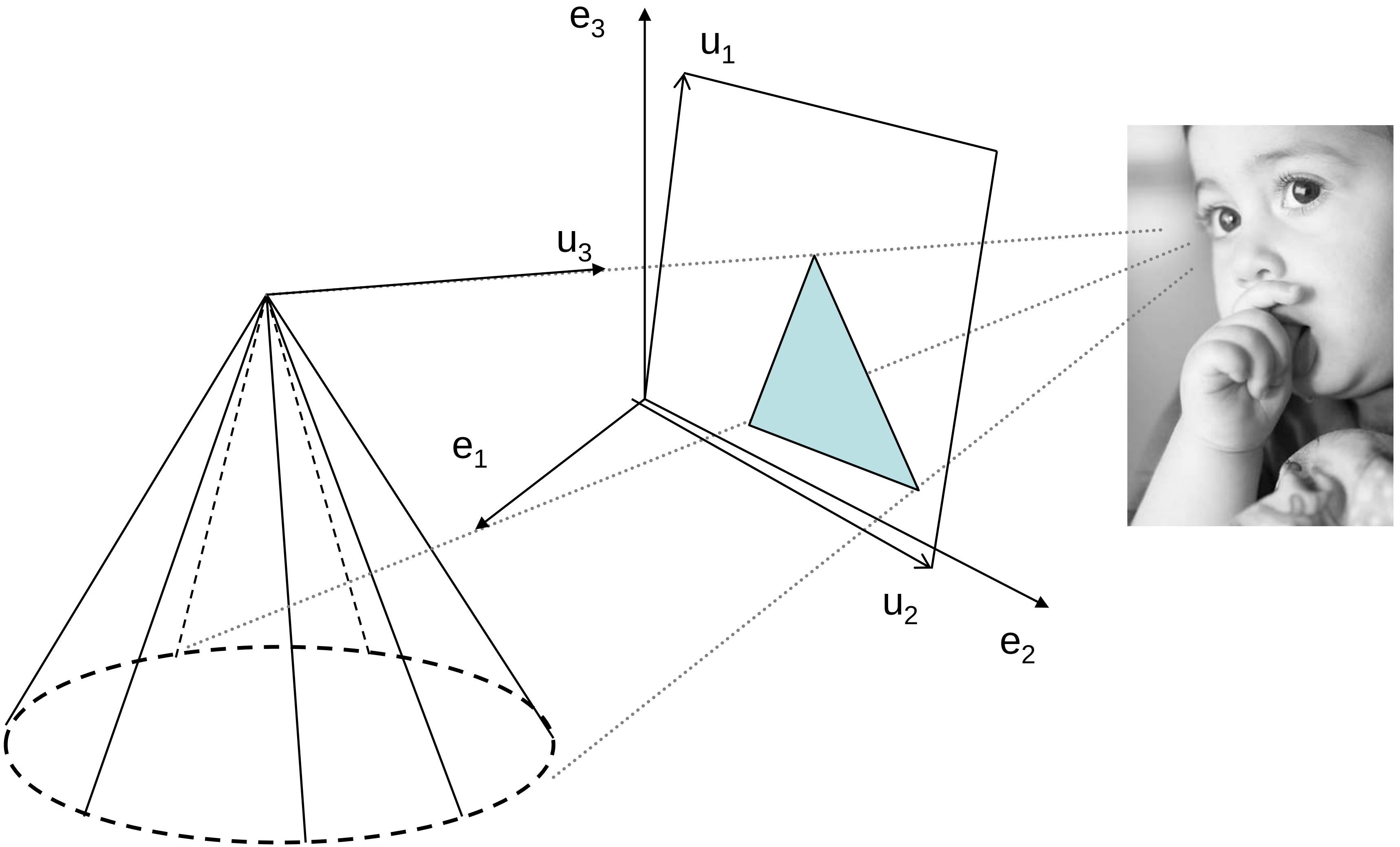
3-mõõtmeline graafika

3-mõõtmelise objekti projekteerimine tasandile



Projektsiooni tüübid

Kui vaatleja on projektsioontasandist lõpmatult kaugel, siis võib vaatlejasilma ja projekteeritava objekti vahelisi kiiri vaadelda paralleelsetena ning sel juhul räägitakse **paralleelprojektsioonist**.

Kui vaatleja kaugus projektsioontasandist on sama suurusjärku objekti mõõtmetega, siis räägitakse **tsentraalprojektsioonist e. perspektiivist**.

Plaanide ja kaartide kujutamisel kasutatakse tavaliselt paralleelprojektsiooni, maalimisel ja arvutimängudes tihtipeale aga tsentraalprojektsiooni.

Paralleelprojektsiooni nimetatakse **ortogonaalseks e. ristprojektsiooniks**, kui projekteerivad kiired on projektsioonitasapinnaga risti; vastasel korral räägitakse **kaldprojektsioonist**.

Paralleelprojektsioon

Projektsioonitasandi S määravad vektorid $\boldsymbol{u}_1$ ja $\boldsymbol{u}_2$: Tasandi punktide kohavektoreid võib vaadelda nende vektorite **lineaarse kattena** (kõikvõimalike lineaarkombinatsioonide hulgana):

$$S = span(u_1, u_2).$$

Et projektsioonitasandit määravaid vektoreid oleks hõlbus kasutada baasivektoritena, tuleks nad valida ortonormaalsete vektoritena. Lisades veel projekteerivate kiirte suunalise ühikvektori $\boldsymbol{u}_3$ (mis on risti vektoritega $\boldsymbol{u}_1$ ja $\boldsymbol{u}_2$ üksnes ristprojektsiooni puhul!), saame baasi ruumis $\mathbf{R}^3$.

Baasivektoritest moodustame maatriksi:

$$U = (u_1, u_2, u_3)$$

See maatriks on ortogonaalmaatriks üksnes ristprojektsiooni puhul.

Leiame maatriksi U pöördmaatriksi:

$$U^{-1} = Y^T = (y_1, y_2, y_3)^T$$

Punkti paralleelprojektsioon tasandile

Punkti paralleelprojektsioon tasandile S piki vektorit $\boldsymbol{u}_3$ on määratud **projektsioonimaatriksiga**

$$P = u_1 y_1^T + u_2 y_2^T$$

Punkti

$$z = (z_1,\ z_2,\ z_3)^T$$

projektsiooni leidmine tasandile S:

$$P\mathrm{z} = \mathrm{y}_1^T \mathrm{z}\, \mathrm{u}_1 + \mathrm{y}_2^T \mathrm{z}\, \mathrm{u}_2 = < \mathrm{y}_1^T, \mathrm{z} > \mathrm{u}_1 + < \mathrm{y}_2^T, \mathrm{z} > \mathrm{u}_2$$

Näide MATLAB-i abil (1)

Kolmemõõtmelise objekti (maja) joonelementide otspunktide koordinaadid:

```
x1=[-2,0,0]';
x2=[-2,0,2]';
x3=[-2,1,2]';
x4=[-2,1,0]';
x5=[-4,0,0]';
x6=[-4,0,2]';
x7=[-4,1,2]';
x8=[-4,1,0]';
x9=[-2,0.5 3]';
x10=[-4,0.5 3]';
x11= [-2,0.7,0]';
x12= [-2,0.7,1.2]';
x13=[-2,1,1.2]';
```

Näide MATLAB-i abil (2)

Objekti formeerimine:

```
Viil1=[x2,x3,x4,x1,x2,x9,x3];
Viil2=[x7,x6,x10,x7,x8,x5];
Sein=[x5,x1,x2,x6];
Katus=[x10,x9,x3,x4,x8,x5,x6];
Uks=[x13,x12,x11];
Maja=[Viil1,Uks,x4,x3,Viil2,Sein,Katus];
```

Koordinaatsüsteem:

```
Xaxis=[[0 0 0]',[3 0 0]'];
Yaxis=[[0 0 0]',[0 3 0]'];
Zaxis=[[0 0 0]',[0 0 3]'];
```

Näide MATLAB-i abil (3)

Baasivektorite, projektsioonimaatriksi ja selle pöördmaatriksi formeerimine:

```
u1=[0,1,0]';
u1=u1/norm(u1);
u2=[0,0,1]';
u2=u2/norm(u2);
u3=[1,1,1]';
u3=u3/norm(u3);
U=[u1,u2,u3];
Uk=[u1,u2];
% Y on maatriksi U pöördmaatriks
Y=inv(U);
% kaks ülemist rida
Yk=[Y(1,:)
Y(2,:)];
```

Näide MATLAB-i abil (4)

Funktsioon, mis leiab projektsiooni ja joonistab selle projektsioontasandile:

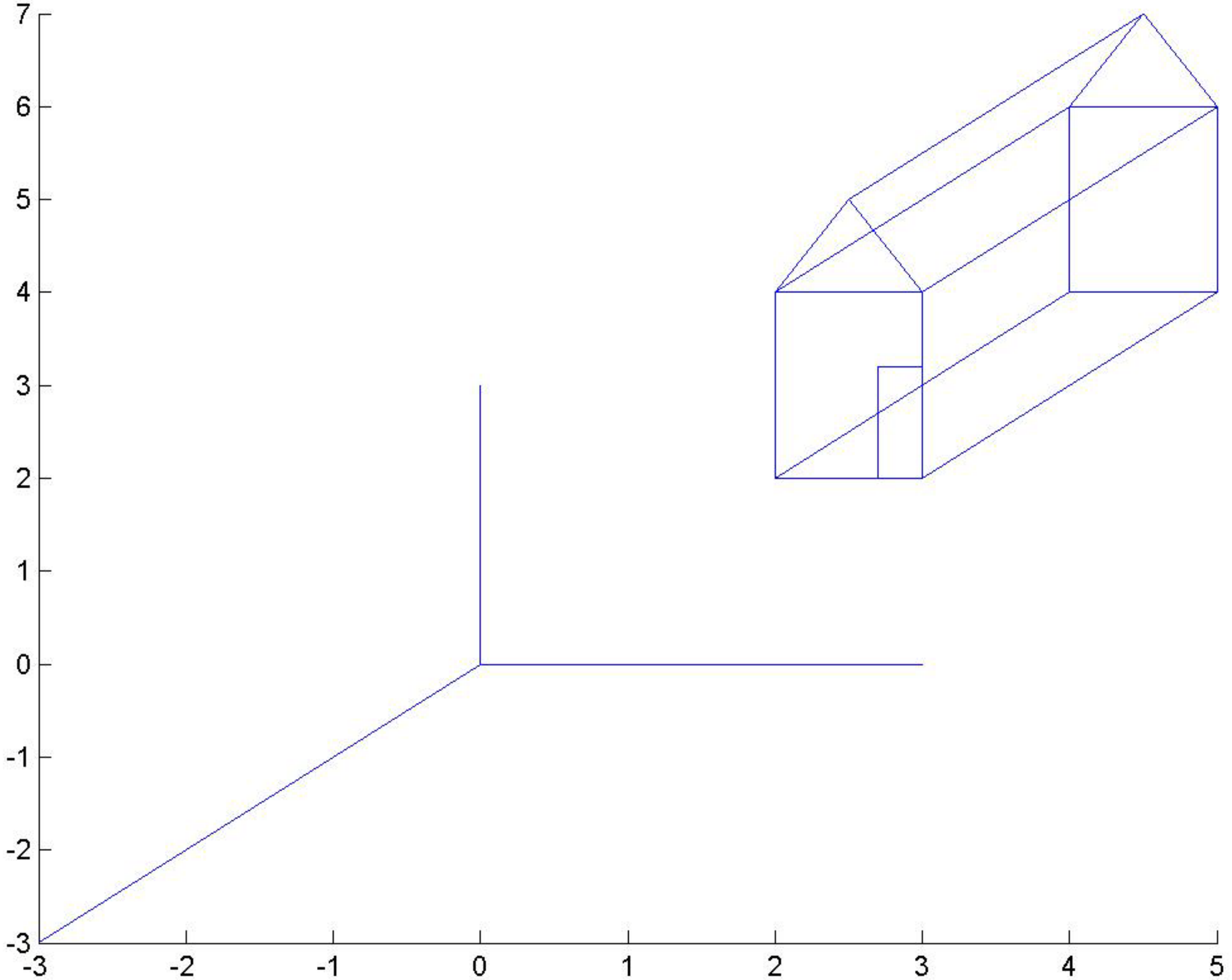
```
function picture3=d3comm(X,Y,type)
% Joonistab murdjoone läbi punktide

%                    X=[x1,x2,...xn]

% paralleelprojektsioonide

Projection=(Y*X)';
plot(Projection(:,1),Projection(:,2), type);
```

Näide MATLAB-i abil (5)

Projektsiooni leidmine ja joonistamine

```
clf % puhastab joonise
hold on
d3comm(Maja,Yk,'-');
d3comm(Xaxis,Yk,'b');
d3comm(Yaxis,Yk,'b');
d3comm(Zaxis,Yk,'b');
figure(gcf) % Vaata joonist.
```

Joonis



Objekti pööre koordinaattelgede ümber

Punkti koordinaatide teisendus pöördel Θ radiaani võrra ümber x-telje:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_x(\theta)\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Punkti koordinaatide teisendus pöördel Θ radiaani võrra ümber y- või z-telje:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_y(\theta)\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_z(\theta)\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Objekti pööre ümber kinnispunkti

Pöörates objekti ümber kinnispunkti $\mathbf{x}_0$, tuleb

1. Viia läbi kinnispunkti nihutamine koordinaatide alguspunkti (siire): $\mathbf{z} = \mathbf{x} - \mathbf{x}_0$;
2. Korrutada punktide $\mathbf{z}_1$; $\mathbf{z}_2$; ...; $\mathbf{z}_n$ koordinaadid pöördemaatriksitega;
3. Siirduda tagasi esialgsetele koordinaatidele: $\mathbf{x} = \mathbf{z} + \mathbf{x}_0$;

Näide MATLAB-i abil

```
function Z =move(X,x0)
% Punktide siire nii, et kinnispunkt siirduks alguspunkti
[m,n]=size(X);
for i=1:n
Z(:,i)=X(:,i)-x0;
end
```

Näide MATLAB-i abil (2)

```
function Z =move(X,x0)
% Punktide siire nii, et kinnispunkt siirduks alguspunkti
[m,n]=size(X);
for i=1:n
Z(:,i)=X(:,i)-x0;
end
```

```
function Y=xrot(X,f)
% Pööre f radiaani võrra ümber x-telje.
Rx=[1 0 0
0 cos(f) -sin(f)
0 sin(f) cos(f)];
Y=Rx*X;
```

Näide MATLAB-i abil (3)

```
clf;
whitebg('k');
figure(gcf);
hold on
axis([-4 8 -4 8]);
axis(axis);
d3comm(Maja,Yk, 'k');
d3comm(Xaxis,Yk,'b');
d3comm(Yaxis,Yk,'b');
d3comm(Zaxis,Yk,'b');
Xmemory=Maja;
x0=[-2,0,0]';
```

Näide MATLAB-i abil (4)

```
for i=1:1200
      Xs=move(Maja,x0);
      Z=xrot(Xs, 2*pi/600);
      Maja=move(Z,-x0);
      d3comm(Xmemory,Yk,'k') %-- Vana pilt eemaldatakse
      d3comm(Maja,Yk,'-') %--     uus pilt.
      d3comm(Yaxis,Yk,'b')%--    alati uuesti teljed,
      d3comm(Zaxis,Yk,'b')%--    sest osa neist
      d3comm(Xaxis,Yk,'b')%--    joonistatakse üle
      Xmemory=Maja;
end
hold off
```