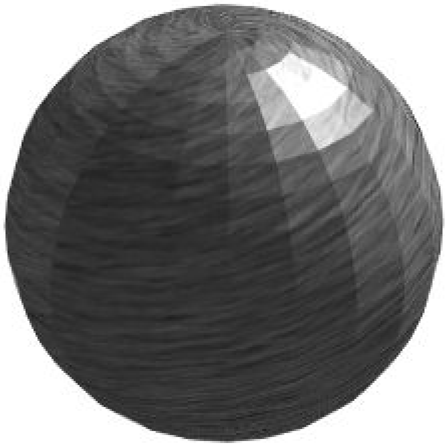
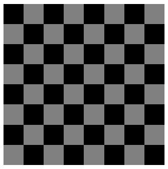
Textury

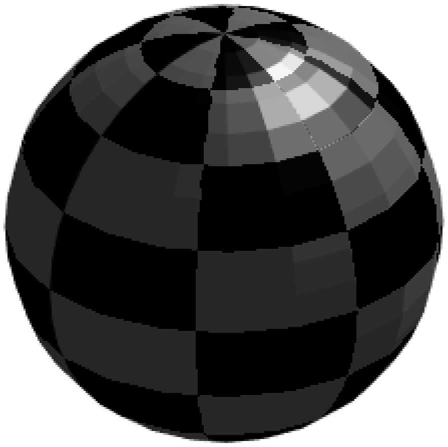
```
[X,Y,Z] = sphere();
f = imread('furr.jpg');

figure,
h = surface(X,Y,Z);
set(h, 'FaceColor','texturemap','EdgeColor','none','Cdata',f);
view(3)
light
axis equal;
axis off;
```



```
t = double(checkerboard > 0.5)/2;
textura(:,:,1) = t;
textura(:,:,2) = t;
textura(:,:,3) = t;
figure,imshow(textura);
```

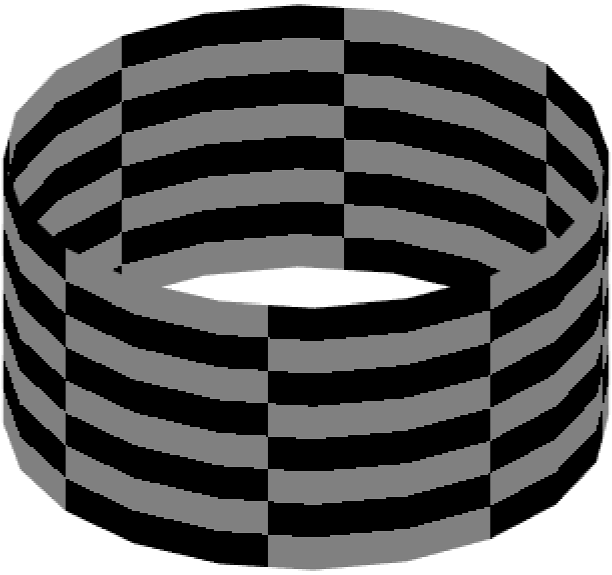


```
[X,Y,Z] = sphere();
figure,
h = surface(X,Y,Z);
set(h, 'FaceColor','texturemap','EdgeColor','none','Cdata',double(textura));
view(3)
axis equal;
axis off;
```



```
[X2,Y2,Z2] = cylinder();
figure,
h = surface(X2,Y2,Z2);
```

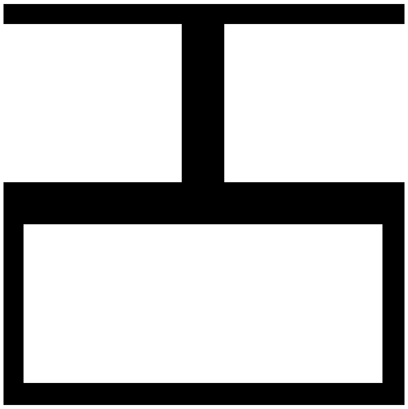
```
set(h, 'FaceColor','texturemap','EdgeColor','none','Cdata',double(textura));
view(3)
axis equal;
axis off;
```



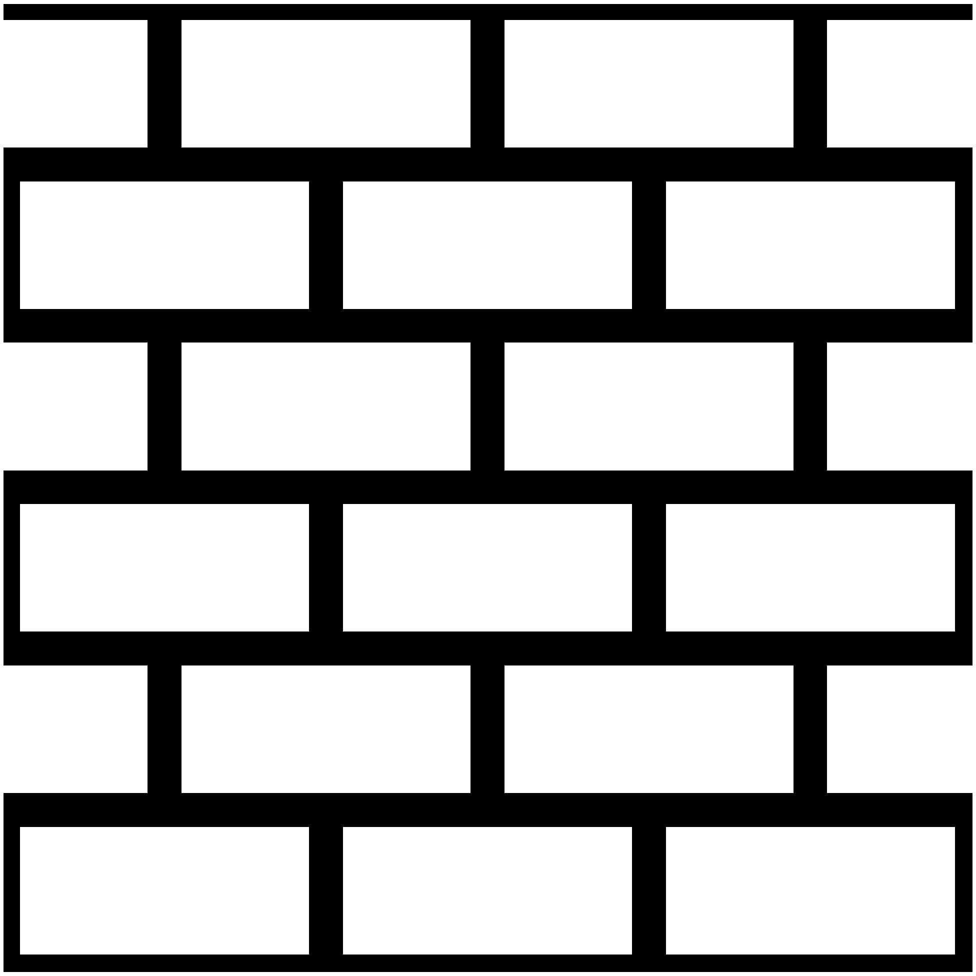
```

```

Textury, které navazují a nenavazují

```
% T1
T1 = ones(200);
T1(90:110,:) = 0;
T1(1 : 90, 90:110) = 0;
T1(110:end,1:10) = 0;
T1(110:end,end-10:end) = 0;
T1(1:10,:) = 0;
T1(end-10:end,:) = 0;

figure,
imshow(T1);
```



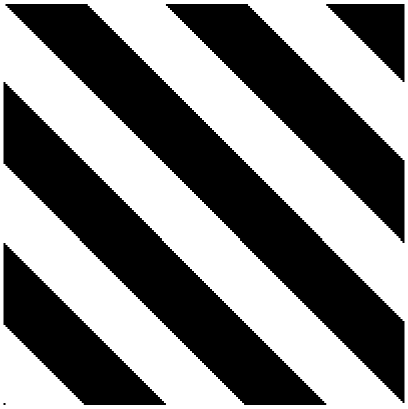
```
%imwrite(T1, "T1a.png");
figure,
imshow([T1 T1 T1; T1 T1 T1; T1 T1 T1]);
```



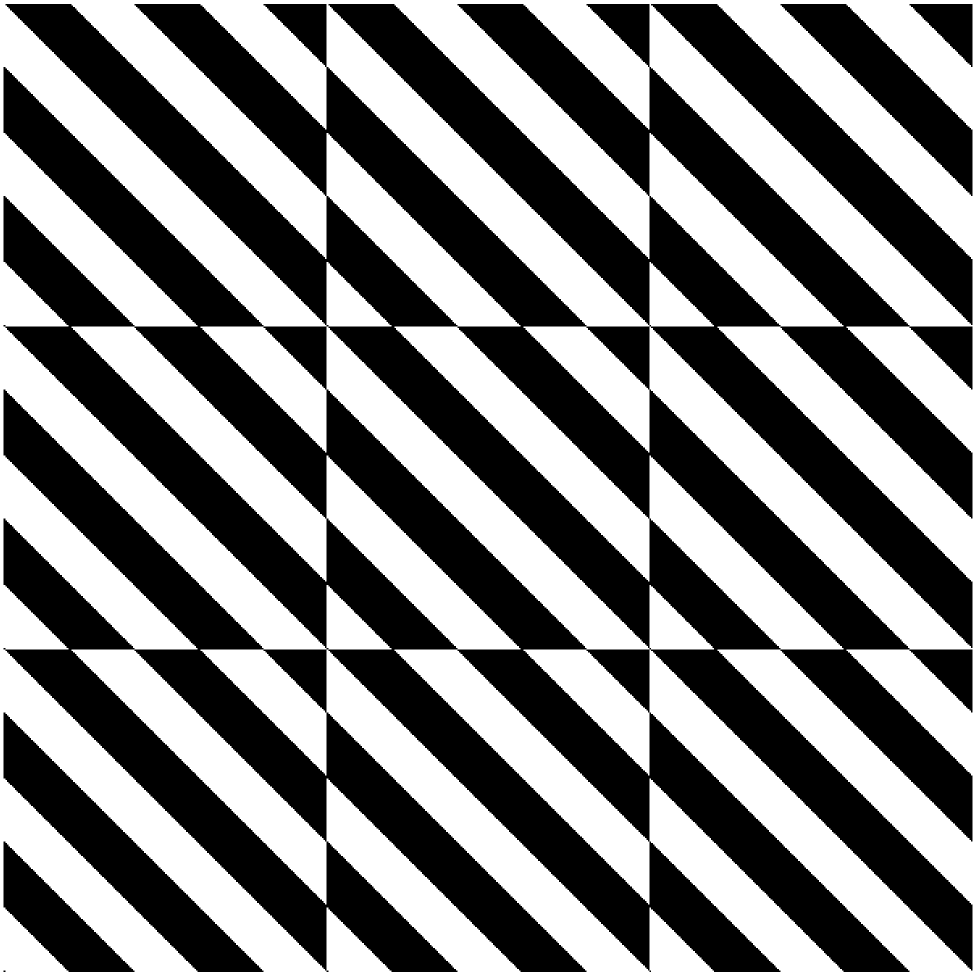
```
%imwrite( [T1 T1 T1; T1 T1 T1; T1 T1 T1], "T1b.png");
```

```
T2 = zeros(200);
for i = 1 : 41
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end

for i = 81 : 121
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end

for i = 161 : 201
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end
```

```
for i = -39 : -1 :  -79
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end

for i = -119 : -1 :  -159
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end

for i = -199 : -1 :  -209
    m = diag(ones([1,200]),i);
    T2 = T2 + m(1:200,1:200);
end

T2 = 1 - T2;

figure,
imshow(T2);
```

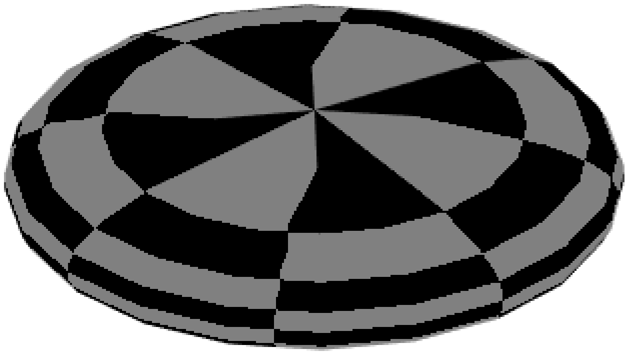


```
%imwrite(T2, "T2a.png");
figure,
imshow([T2 T2 T2; T2 T2 T2; T2 T2 T2]);
```

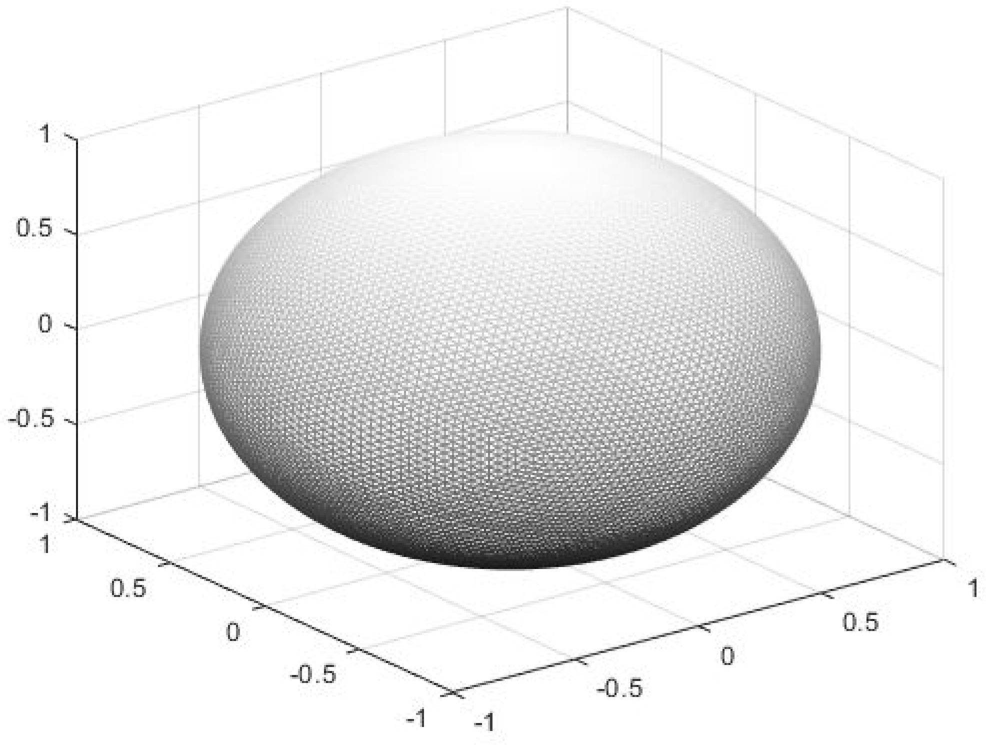


```
%imwrite( [T2 T2 T2; T2 T2 T2; T2 T2 T2], "T2b.png");
```

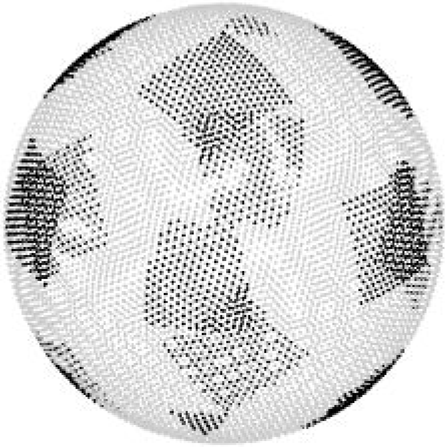
```
h = 0:0.05:2;
R = 2;
r1 = (R^2 - h.^2).^0.5;
r2 = (R^2 - (2-h).^2).^0.5;
r = [r2 r1];
[X,Y,Z] = cylinder(r);

figure,
h = surface(X,Y,Z);
set(h, 'FaceColor','texturemap','EdgeColor','none','Cdata',double(textura));
view(3)
axis equal;
axis off;
```



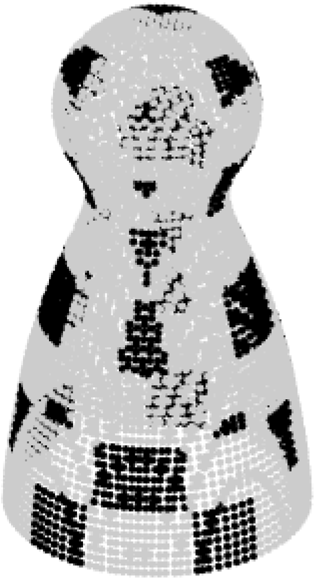
```
koule = icoSphereMesh(8);
trimesh(koule.face, koule.x, koule.y, koule.z);
```



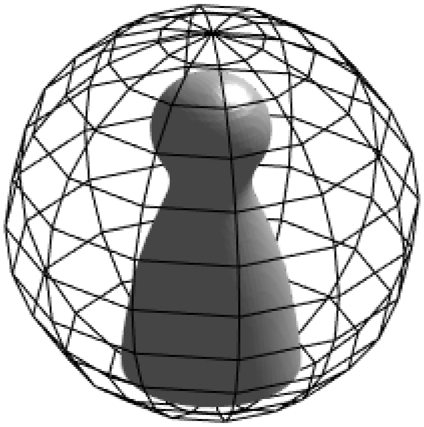
```
koule.barva = 0.8*ones(size(koule.x));

koule.barva((koule.x <=-0.5 | (koule.x <=0.5 & koule.x > 0)) & ...
            (koule.y <=-0.5 | (koule.y <=0.5 & koule.y > 0)) & ...
            (koule.z <=-0.5 | (koule.z <=0.5 & koule.z > 0))) = 0;

koule.barva((koule.x > 0.5 | (koule.x <=0 & koule.x > -0.5)) & ...
            (koule.y > 0.5 | (koule.y <=0 & koule.y > -0.5)) & ...
            (koule.z > 0.5 | (koule.z <=0 & koule.z > -0.5))) = 0;


scatter3(koule.x, koule.y, koule.z, 3, repmat(koule.barva,[1,3]),'filled');


axis equal;
axis off;
```



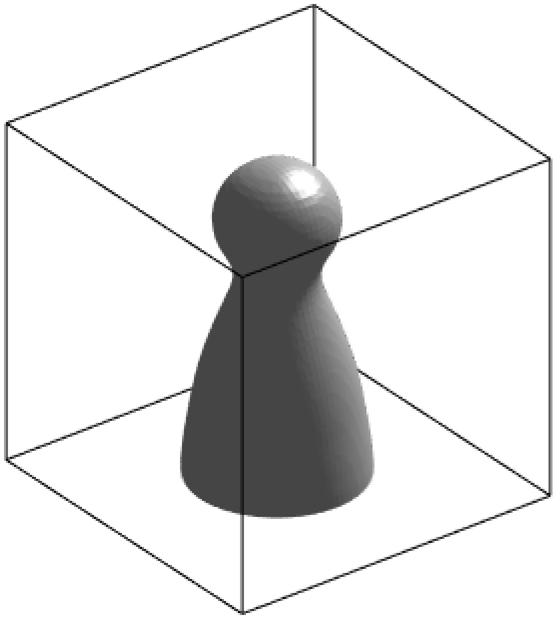
```
pawnStl = stlread("pawn.stl");
pawn.x = pawnStl.Points(:,1);
pawn.y = pawnStl.Points(:,2);
pawn.z = pawnStl.Points(:,3);

pawn.barva = 0.8*ones(size(pawn.x));
pawn.barva((pawn.x <=-6 | (pawn.x <=0 & pawn.x > -3) | (pawn.x <=6 & pawn.x > 3)) & ...
           (pawn.y <=-6 | (pawn.y <=0 & pawn.y > -3) | (pawn.y <=6 & pawn.y > 3)) & ...
           (pawn.z <= 3 | (pawn.z <=9 & pawn.z > 6)| (pawn.z <=15 & pawn.z > 12)| (pawn.z <=21

pawn.barva(((pawn.x <=-3 & pawn.x > -6)|(pawn.x <=3 & pawn.x > 0)|pawn.x > 6) & ...
           ((pawn.y <=-3 & pawn.y > -6)|(pawn.y <=3 & pawn.y > 0)|pawn.y > 6) & ...
           ((pawn.z <=6 & pawn.z > 3)| (pawn.z <=12 & pawn.z > 9)| (pawn.z <=18 & pawn.z > 15)

scatter3(pawn.x, pawn.y, pawn.z, 10, repmat(pawn.barva,[1,3]),'filled');

axis equal;
axis off;
```



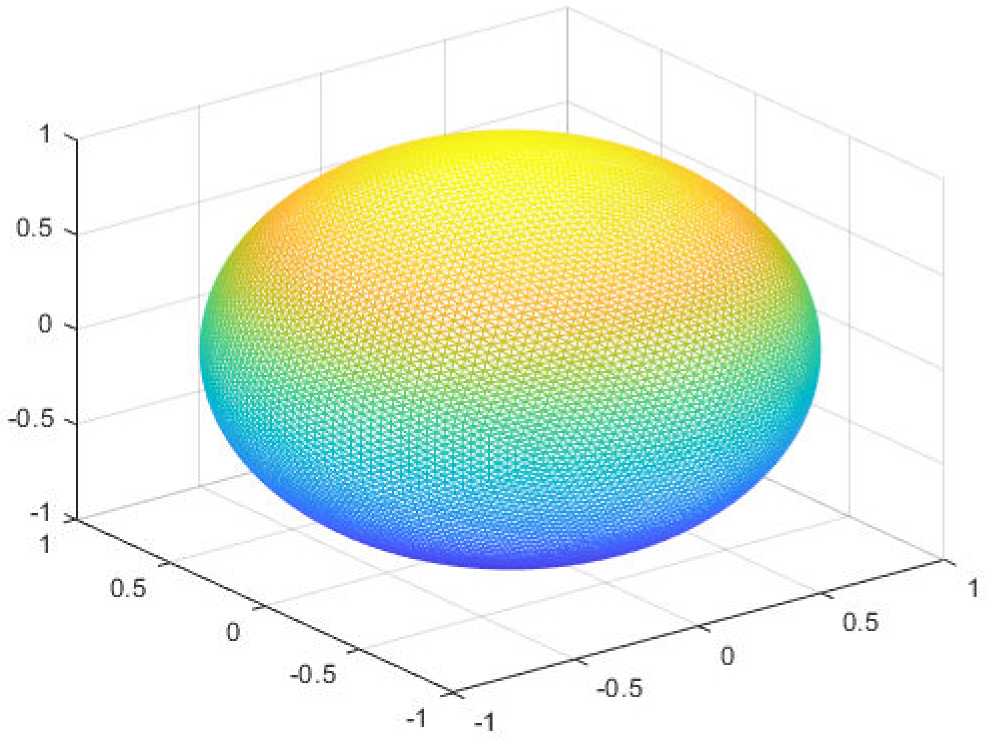
```
[X,Y,Z] = sphere(12);
figure,
hold on
surface(15*X,15*Y,15*Z+12, 'FaceColor',"none");
trisurf(pawnStl, 'EdgeColor', "none", 'FaceColor',[0.9 0.9 0.9]);
light
hold off
axis equal
axis off;
view(3)
```



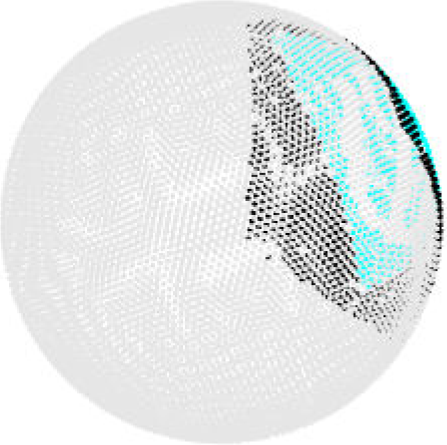
```
vrcholy = [-13 -13 0;   %1
           13 -13 0;  %2
           13 13 0;  %3
           -13 13 0; %4
           -13 -13 26;  %5
           13 -13 26; %6
           13 13 26; %7
           -13 13 26];%8

hrany = [1 2;
         2 3;
         3 4;
         4 1;
         1 5;
         2 6;
         3 7;
         4 8;
         5 6;
         6 7;
         7 8;
         8 5];

plochy = [4 3 2 1;
          1 2 6 5;
          1 5 8 4;
          3 4 8 7;
          2 3 7 6;
          5 6 7 8];

figure,
hold on
for i = 1 : size(hrany,1)
    plot3([vrcholy(hrany(i,1),1),vrcholy(hrany(i,2),1)],[vrcholy(hrany(i,1),2),vrcholy(hrany(i,
end
trisurf(pawnStl, 'EdgeColor', "none", 'FaceColor',[0.9 0.9 0.9]);
light
hold off
axis equal
axis off;
view(3)
```



```
koule = icoSphereMesh(10);
trimesh(koule.face, koule.x, koule.y, koule.z);
```

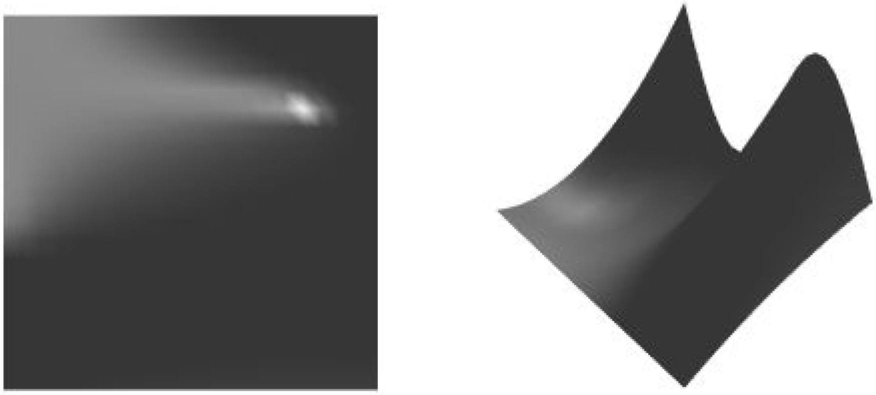


```
t = imread("up.png");

koule.barva = 0.9*ones([size(koule.x,1),3]);

for i = 1 : size(koule.x,1)
    if(abs(koule.x(i)) > abs(koule.y(i)) && abs(koule.x(i)) > abs(koule.z(i)) && koule.x(i) >=
        v = (koule.y(i) + koule.x(i))/(2*koule.x(i));
        u = (-koule.z(i) + koule.x(i))/(2*koule.x(i));
        U = floor(u*299) + 1;
        V = floor(v*299) + 1;
        koule.barva(i,:) = t(U,V,:);
    end
end


scatter3(koule.x, koule.y, koule.z, 5, koule.barva,'filled');


axis equal;
axis off;
```



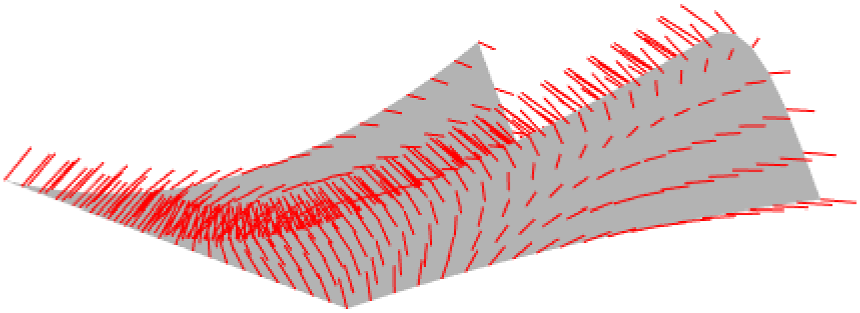
```
v = 0 : 0.05 : 1;
u = 0 : 0.05 : 1;

[U,V] = meshgrid(u,v);

P0 = (2.5-3.3).*V + 3.3;
P1 = 5*sin(V*(2*pi))+2.5;
P2 = 10*(U-0.5).^2;
P3 = -2*(U-0.3).^2+1.88+1.6;


plocha2 = (1-U).*P0 + U.*P1 + (1-V).*P3 + V.*P2 - (1-U).*(1-V).*P0(1,1) - (1-U).*(V).*P0(11,11)

figure,
subplot(1,2,1), surf(U,V,plocha2, "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
lighting gouraud
light
axis equal
axis off
view(2)
subplot(1,2,2), surf(U,V,plocha2, "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
lighting gouraud
light
view([-45 45])
colormap gray;
```

```
axis off
```



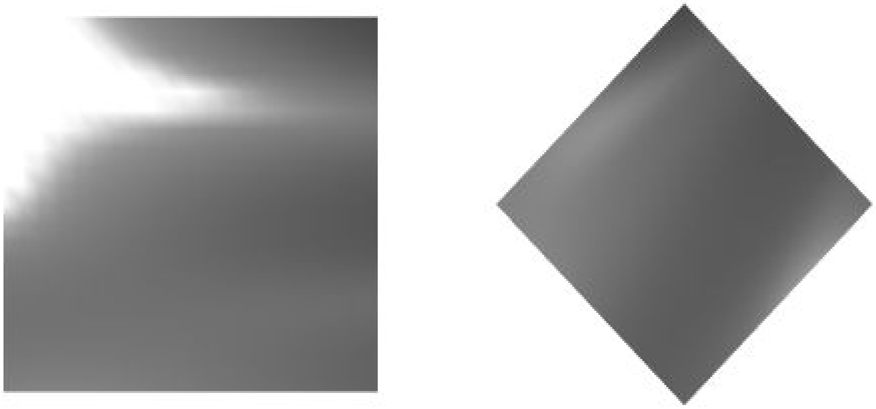
```
v = 0 : 0.05 : 1;
u = 0 : 0.05 : 1;

[U,V] = meshgrid(u,v);


plocha = zeros(size(U));

figure,
subplot(1,2,1), surf(U,V,plocha, "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
lighting gouraud
light
axis equal
axis off
view(2)
subplot(1,2,2), surf(U,V,plocha, "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
lighting gouraud
light
view([-45 45])
colormap gray;

axis off
```

```
v = 0 : 0.05 : 1;
u = 0 : 0.05 : 1;

[U,V] = meshgrid(u,v);

P0 = (2.5-3.3).*V + 3.3;
P1 = 5*sin(V*(2*pi))+2.5;
P2 = 10*(U-0.5).^2;
P3 = -2*(U-0.3).^2+1.88+1.6;


plocha2 = (1-U).*P0 + U.*P1 + (1-V).*P3 + V.*P2 - (1-U).*(1-V).*P0(1,1) - (1-U).*(V).*P0(11,11

figure,
surfnorm(U,V,plocha2, "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);

colormap gray;
%axis equal
axis off
```

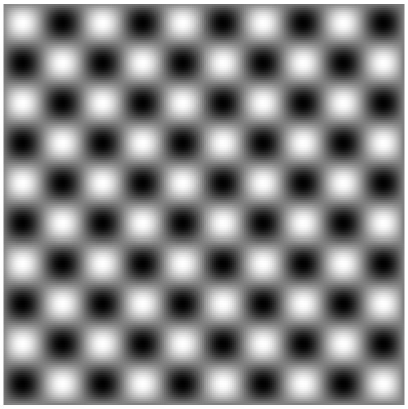


```
[nx, ny, nz] = surfnorm(plocha2);
v = 0 : 0.05 : 1;
u = 0 : 0.05 : 1;

[U,V] = meshgrid(u,v);
b = reshape([nx ny nz], size(U,1),size(U,2),3);
plocha = zeros(size(U));

% figure,
% surf(ones(size(U)), 'VertexNormals',b , "EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
% lighting gouraud
% camlight
% view(2)
% axis equal
% axis off

figure,
subplot(1,2,1), surf(U,V,plocha,'VertexNormals',b , "EdgeColor","none", "FaceColor",[0.7 0.7 0.
lighting gouraud
light
axis equal
axis off
view(2)
subplot(1,2,2), surf(U,V,plocha,'VertexNormals',b , "EdgeColor","none", "FaceColor",[0.7 0.7 0.
lighting gouraud
light
view([-45 45])
```

```
colormap gray;

axis off
```



```
v = 1 :  1 : 200;
u = 1 :  1 : 200;
velikost = size(u,2);

[U,V] = meshgrid(u,v);
textura = sin(10*U*pi/velikost).*sin(10*V*pi/velikost);


figure,
imshow(textura,[]);
```



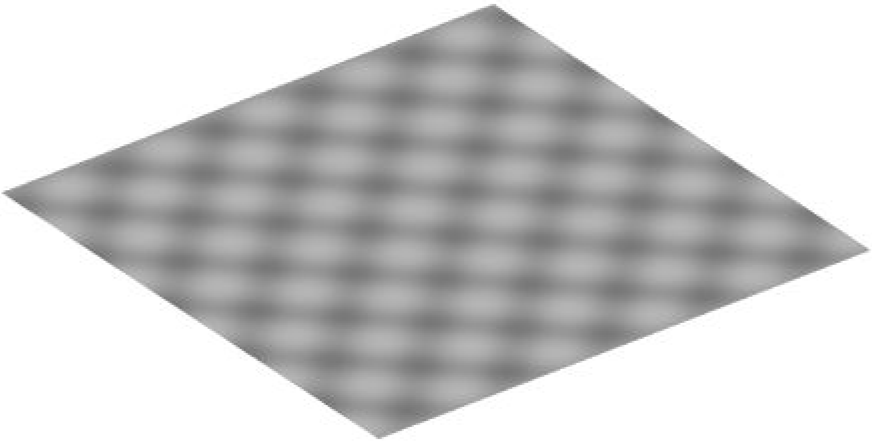
```
uu = 35;
vv = 65;

gu = sin(10*(uu+1)*pi/velikost).*sin(10*vv*pi/velikost) - sin(10*(uu-1)*pi/velikost).*sin(10*vv
```

```
gu = -0.1564
```

```
gv = sin(10*uu*pi/velikost).*sin(10*(vv+1)*pi/velikost) - sin(10*uu*pi/velikost).*sin(10*(vv-1)
```

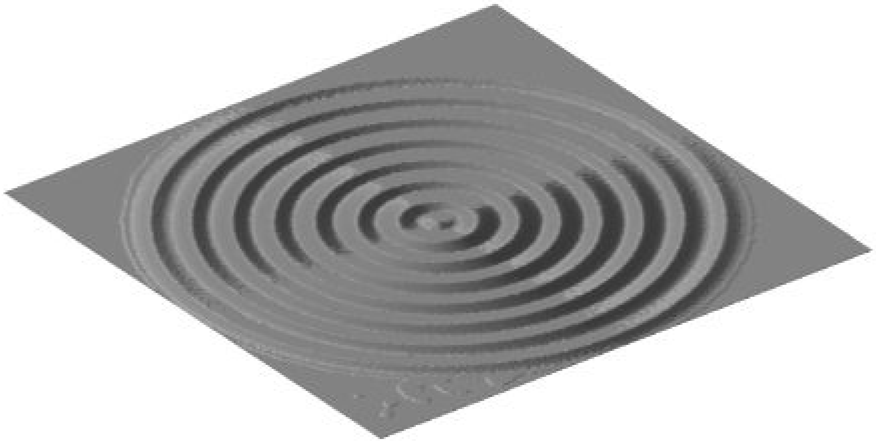
```
gv = 0.1564
```

```
Gu = imfilter(textura, [-1;0;1]);
Gv = imfilter(textura, [-1 0 1]);

Su = [1, 0, 0];
Sv = [0, 1, 0];
N = [0, 0, 1];

Normaly =[];

Normaly(:,:,1) = Gv;
Normaly(:,:,2) = Gu;
Normaly(:,:,3) = ones(size(Gu));




%imwrite((textura+1)/2,"bumpTextura.png");

figure,
surf(U,V,zeros(size(Gu)), 'VertexNormals',Normaly ,"EdgeColor","none", "FaceColor",[0.7 0.7 0.7
lighting gouraud
light
material dull
axis equal
axis off
```



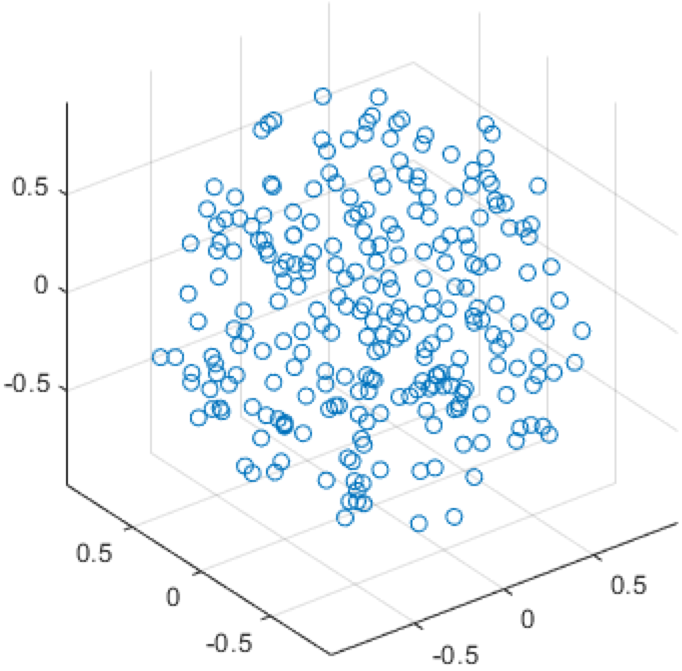
```
textura = rgb2gray(imread("bump6.png"));
%textura = rgb2gray(imread("bump7.jpg"));
Gu = imfilter(textura, [-1;0;1]);
Gv = imfilter(textura, [-1 0 1]);

Su = [1, 0, 0];
Sv = [0, 1, 0];
N = [0, 0, 1];

Normaly = [];
Normaly(:,:,1) = Gv;
Normaly(:,:,2) = Gu;
Normaly(:,:,3) = ones(size(Gu));




%imwrite((textura+1)/2,"bumpTextura.png");
```

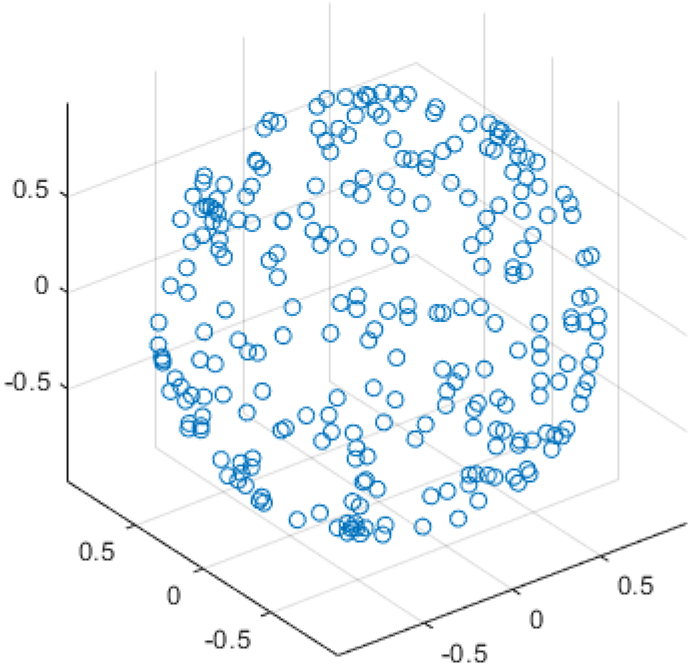
```
figure,
surf(zeros(size(Gu)), 'VertexNormals',Normaly ,"EdgeColor","none", "FaceColor",[0.7 0.7 0.7]);
lighting gouraud
light
material shiny
axis equal
axis off
```



Perlin

```
G = generateG();
```



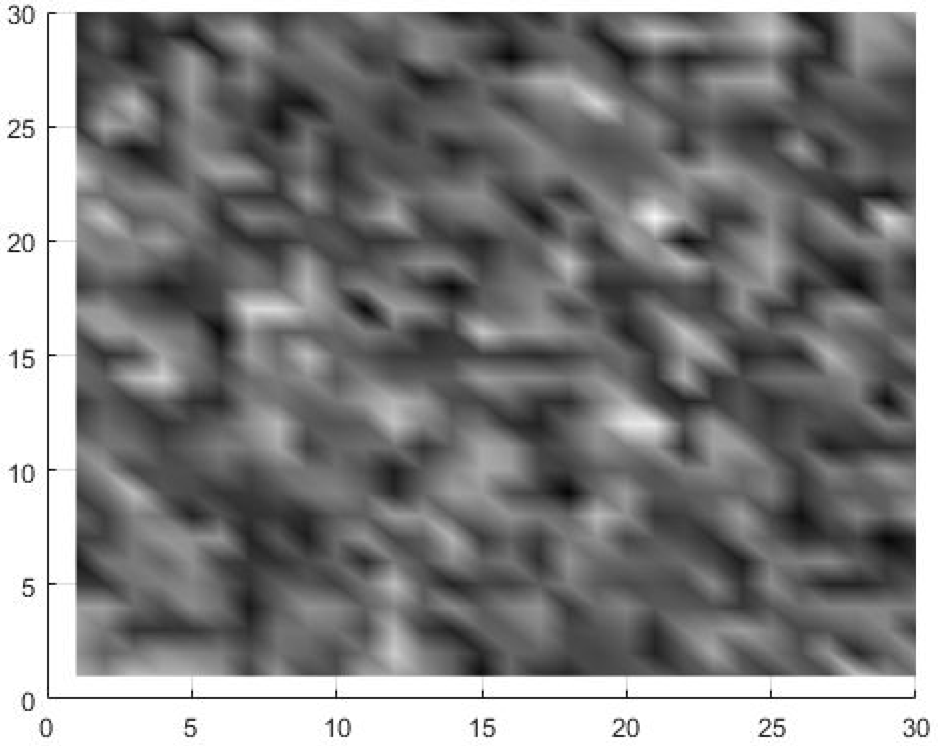


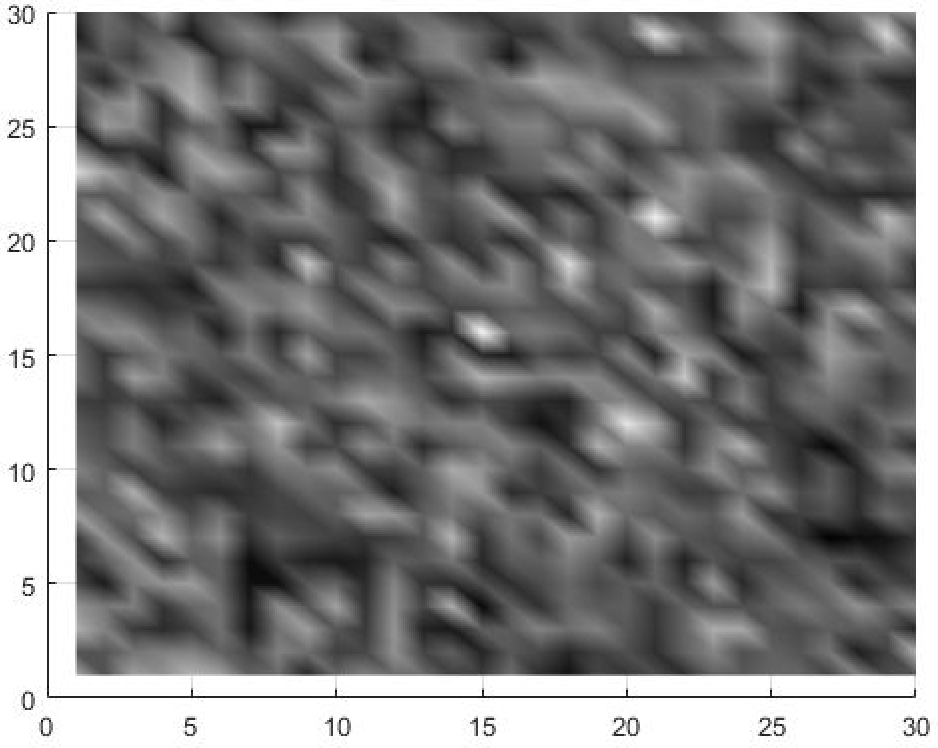
```
P = permutaceIndexu();
```

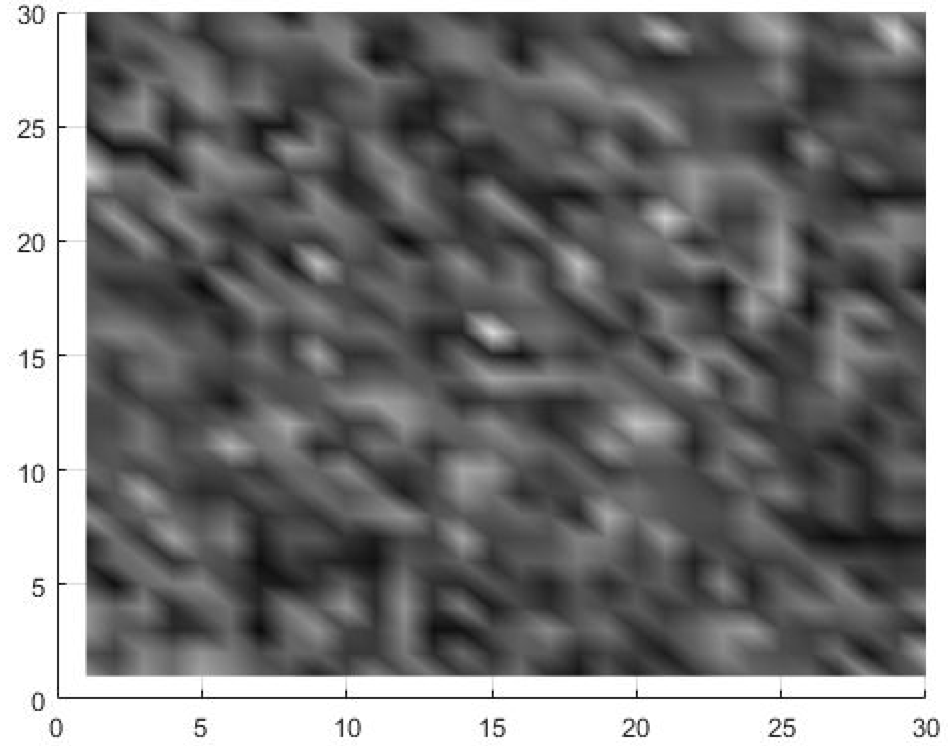
```
P = 1×256
```

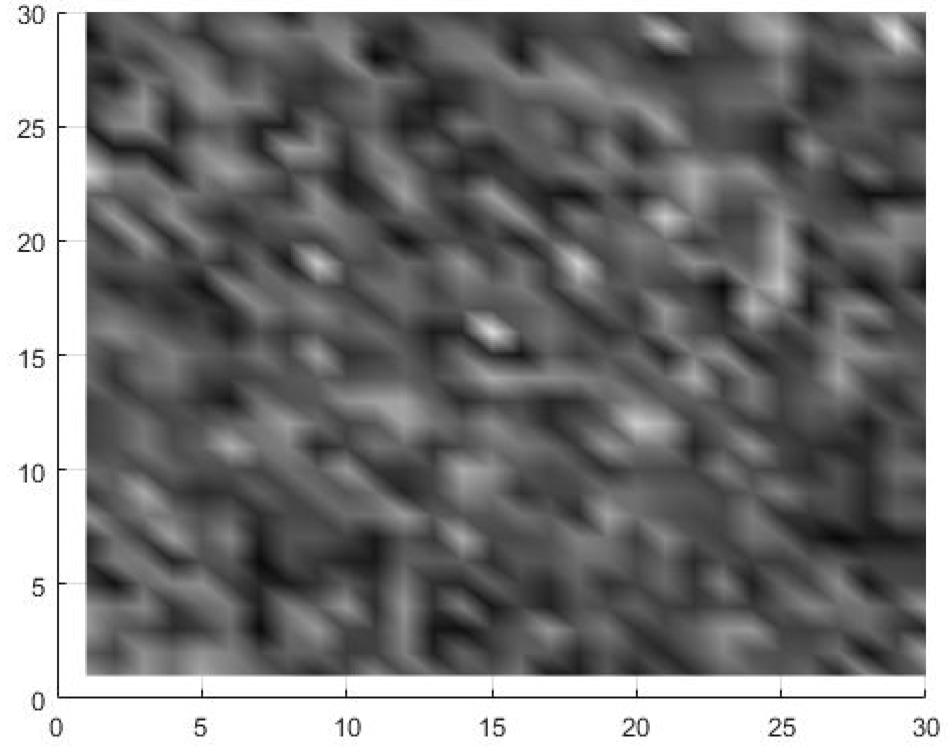
```
    71    70   230   129    98   167    43   229    47    44   233    84    63···
```

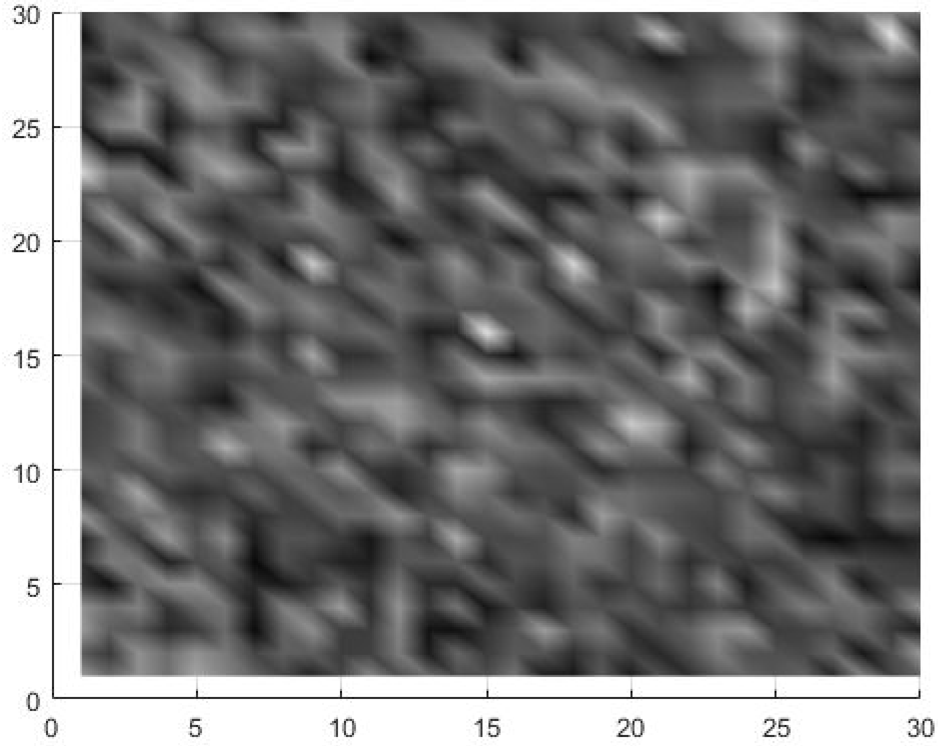
```
X = linspace(1,100,100);
Y = linspace(1,100,100);

[U,V] = meshgrid(X,Y);

I = zeros([100,100,3]);
p=0.5;

snoise = zeros([100,100,3]);
for i = 0 : 7
    I = zeros([10,10,3]);

    for x = 1 : 100
        for y = 1 : 100
            [I(x,y,1), I(x,y,2), I(x,y,3)] = perlin(2^i*x,2^i*y,1,G,P);
        end
    end
    snoise = snoise + p^i * I;

    J = sum(abs(snoise),3);


    figure,
surf(U,V,J, "EdgeColor","none");
xlim([0,30])
ylim([0,30])
shading interp
colormap gray
view(2)

end
```

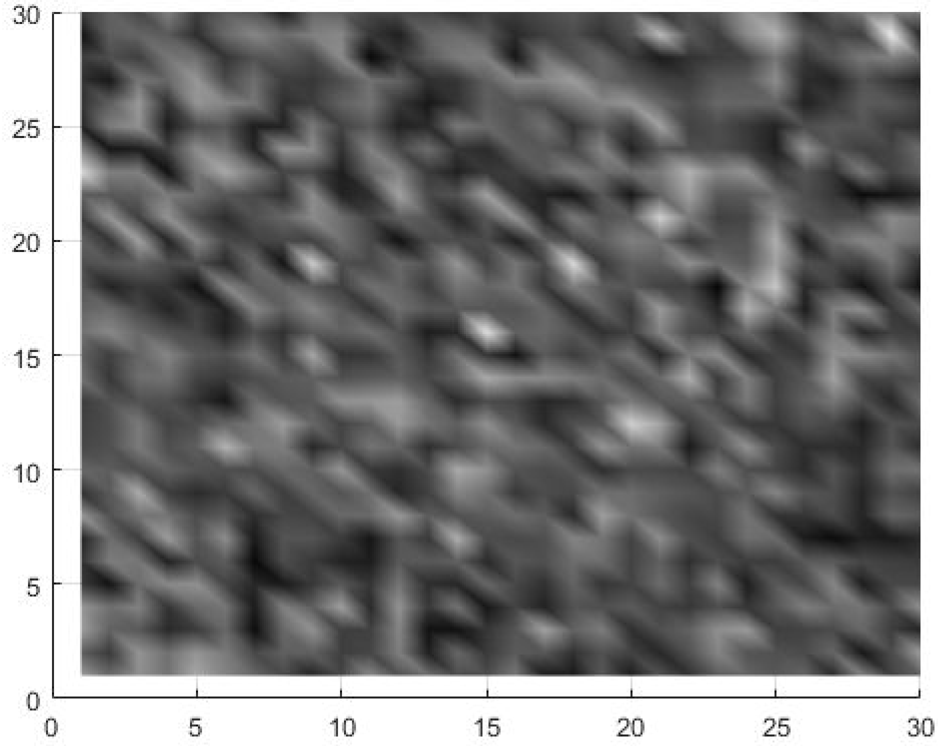

30
25
20
15
10
5
0
0
5
10
15
20
25
30

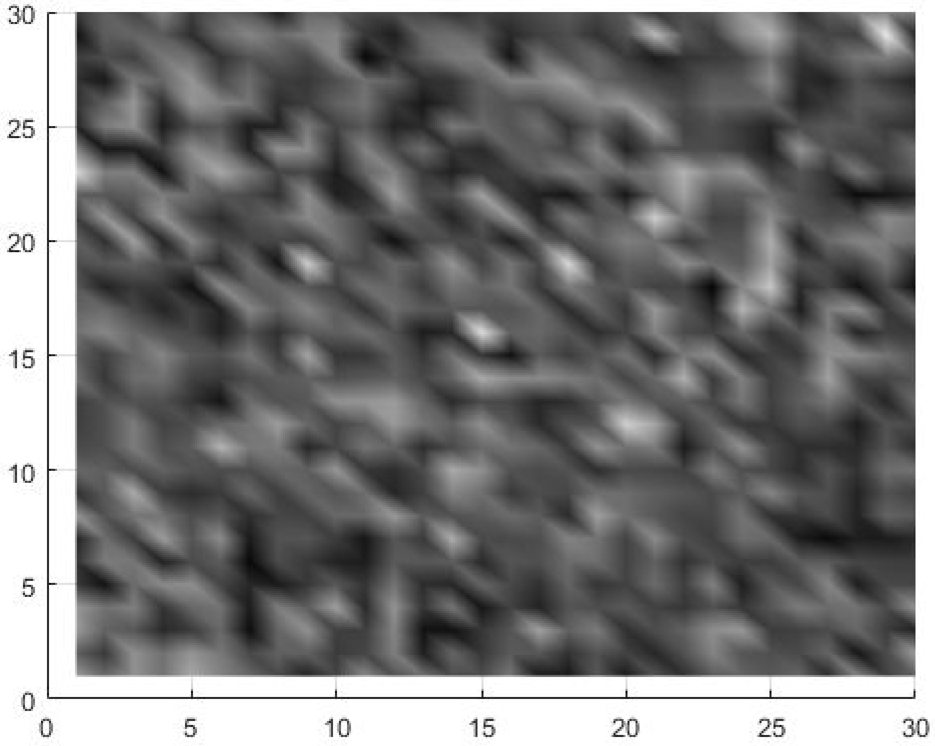

30
25
20
15
10
5
0
0
5
10
15
20
25
30

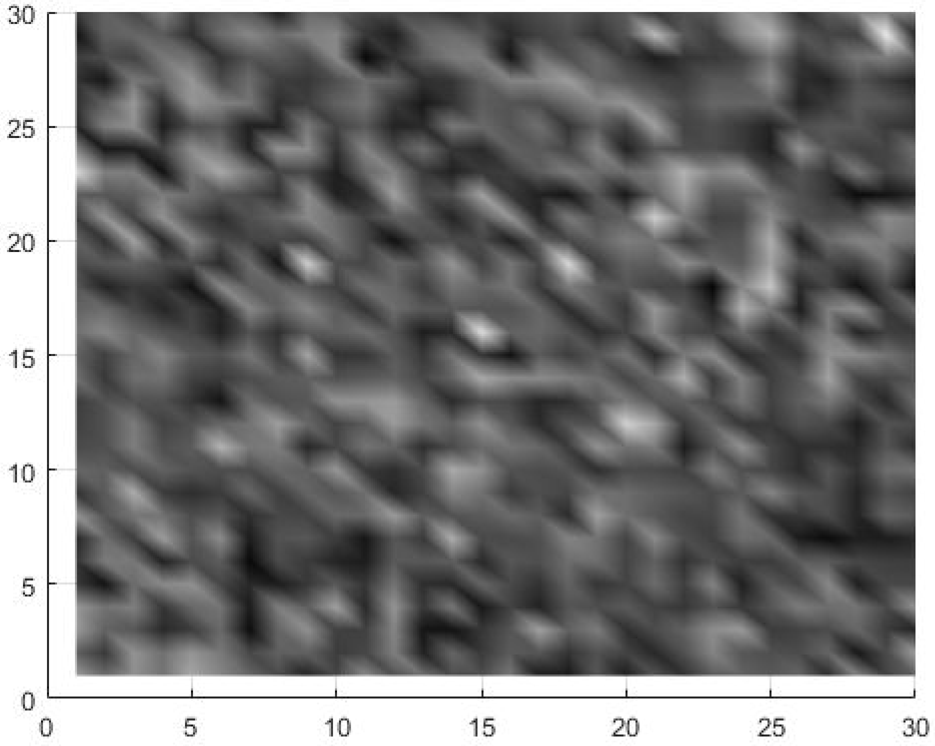

30
25
20
15
10
5
0
0
5
10
15
20
25
30


30
25
20
15
10
5
0
0
5
10
15
20
25
30

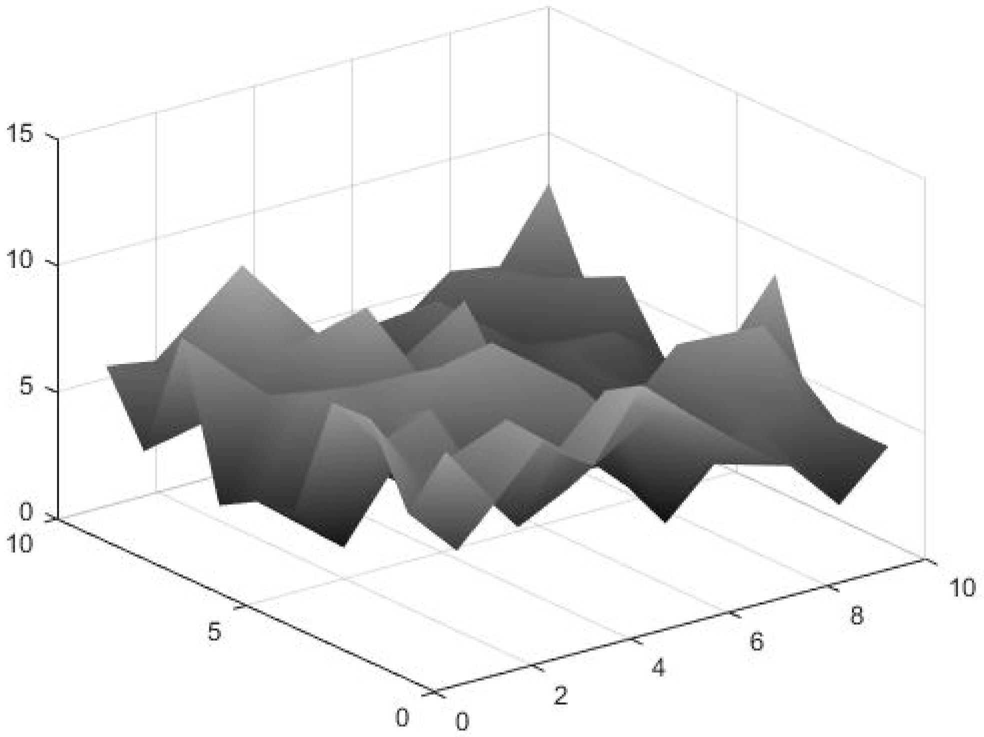

30
25
20
15
10
5
0
0
5
10
15
20
25
30


30
25
20
15
10
5
0
0
5
10
15
20
25
30

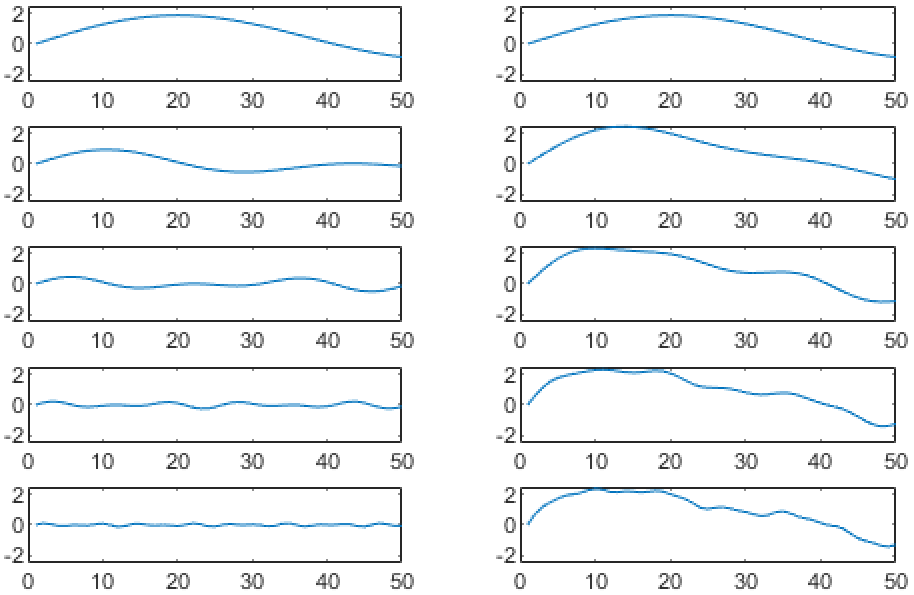


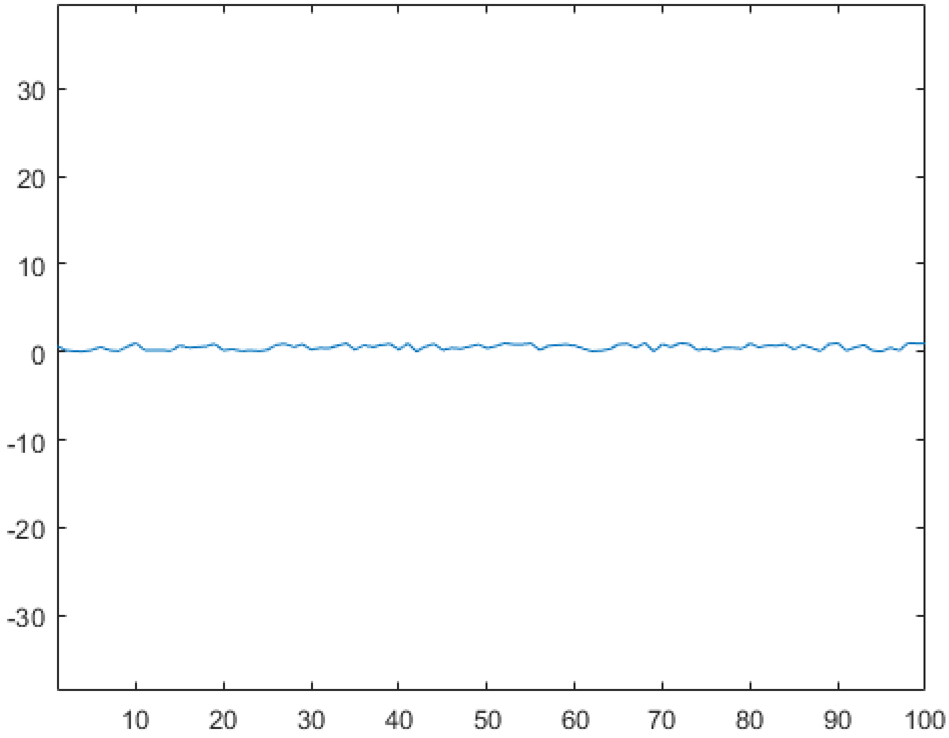


```
J = sum(abs(snoise),3);
figure,
surf(U,V,J, "EdgeColor","none");
xlim([0,10])
ylim([0,10])
shading interp
colormap gray
```



```
% sin (2 * x) + sin(pi * x)

X = linspace(0,1.5,50);
snoise  = zeros([1,50]);
p = 0.5;
index = 1;
figure,
for i = 0 : 4
    x = 2^i * X;
    noise = sin(2 * x) + sin(pi * x);
    x =  p^i * noise;
    subplot(6,2,index), plot(x);
    snoise = snoise + x;
    index = index+1;
    ylim([-2.5,2.5]);
    subplot(6,2,index), plot(snoise);
    ylim([-2.5,2.5]);
    index = index+1;
end
```


2
0
-2
0
10
20
30
40
50


30
20
10
0
-10
-20
-30
10
20
30
40
50
60
70
80
90
100

Pomocné funkce

```
% Vygeneruj náhodné směry
```

```
function [G] = generateG()

    rvals = 2*rand(256,1)-1;
    elevation = asin(rvals);
    azimuth = 2*pi*rand(256,1);

    radii = (rand(256,1).^(1/3));

    [x,y,z] = sph2cart(azimuth,elevation,radii);



    figure,
    scatter3(x, y, z);
    axis equal;

    velikost = 1./(x.^2 + y.^2 + z.^2).^(0.5);
    G = [velikost.*x, velikost.*y, velikost.*z];
    figure,
    scatter3(G(:,1), G(:,2), G(:,3));
    axis equal;
end

% Permutace indexů
function P = permutaceIndexu()
    P = 1 : 256;
    for i = 1 : 256
        k = randi([1,256]);
        pom = P(i);
        P(i) = P(k);
        P(k) = pom;
    end
end

% fold funkce
function index = fold(i,j,k,P)
    index = P(mod(P(mod(P(mod(i,256) + 1) + j,256) + 1) + k,256) +1);
end

% drop funkce
function d = drop(t)
    d = 1 - 3*abs(t)^2 + abs(t)^3;
end

% ubytek
function omega = ubytek(u,v,w)
    omega = drop(u) * drop(v) * drop(w);
end
```

```
function [x2,y2,z2] = perlin(x,y,z,G,P)
        x2 = 0;
        y2 = 0;
        z2 = 0;
        for s = 0 : 1
            for t = 0 : 1
                for u = 0 : 1
                    i = floor(x)+s;
                    j = floor(y)+t;
                    k = z+u;

                    vx = x - i;
                    vy = y - j;
                    vz = 1 - k;

                    vlnka = ubytek(vx,vy,vz) * G(fold(i,j,k,P),:);
                    x2 = x2 + vlnka(1);
                    y2 = y2 + vlnka(2);
                    z2 = z2 + vlnka(3);
                end
            end
        end
end
```