**计算机图形学实验三实验报告**

姓名：王新钰 学号：201701101121

一、实验目的

通过Matlab语言编程理解一些基本图形变换。

二、实验内容及要求

**实验要求**：要求采用Matlab语言进行编程，了解如何利用matlab实现图形的二维、三维变换，并能编写简单的程序。

**实验内容：**编程练习①~⑤

三、MATLAB编程练习

①三维图形绘制

1. plot3（）函数

如：

t=0:pi/50:10\*pi;

plot3(sin(t),cos(t),t)

或

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

2. surf（）函数生成由填充完的面片组成的图形：表面图形

如：

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

figure

surf(x,y,z)

3. mesh（）函数生成由线条构成的图形：网格图

如：

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

figure

mesh(x,y,z)

4. patch()函数：绘制多面体（由顶点和面定义）以下程序使用patch（）函数绘制了3个正方体

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];顶点

fac=[1 2 3 4;2 6 7 3;4 3 7 8;1 5 8 4;1 2 6 5;5 6 7 8];面由顶点序号定义

subplot(1,3,1)

patch('faces',fac,'vertices',vert,'Facecolor','w'); （立方体由顶点和面定义）

view(3);

subplot(1,3,2)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(6),'Facecolor','flat');

view(3);

subplot(1,3,3)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(8),'Facecolor','interp');

view(3);

注：flat为面着色，可以看出面的边缘；interp是插值着色，对边缘进行了插值处理。

②三维图形变换

1.编写一段代码，绘制一个球体，然后平移（利用hold on把变换前后的球体放在同一个figure中）

2.编写一段代码，绘制一个柱体，然后绕x轴旋转60°

3.用sphere生成多面体，然后对其进行错切变换

[x,y,z]=sphere(5);

surf(x,y,z)

T=[1 0 0;

1.2 1 0;

0 0 1];

for i=1:6

for j=1:6

a=x(i,j);

b=y(i,j);

c=z(i,j);

m=[a b c]\*T';

x1(i,j)=m(1);

y1(i,j)=m(2);

z1(i,j)=m(3);

end

end

figure

surf(x1,y1,z1)

要求：先运行以上代码，然后改变球体的多面体数量，错切变换系数，做成小动画

4.三维旋转的实现

for i=-2\*pi:0.5:2\*pi

R=[cos(i) sin(i) 0;-sin(i) cos(i) 0;0 0 1];

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

vert=vert\*R;

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

pause(0.1)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(8),'Facecolor','interp');

view(3);

end

要求：读懂以上程序，程序先绘制一个长方体，然后隔0.1秒又绘制出另一个围绕z轴旋转i角度的长方体，循环中每次（每个顶点）都被旋转矩阵变换成新的顶点。把变换矩阵改成围绕x轴的旋转矩阵、y轴旋转的变换矩阵，再运行程序看看有什么不同。

5.透视投影和平行投影

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

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

subplot(1,2,1)

view(3);

patch('faces',fac,'vertices',vert,'Facecolor','w');

subplot(1,2,2)

view(3);

camproj('perspective')

patch('faces',fac,'vertices',vert,'Facecolor','w');

不同视点下的观察效果

(1)

z=ones(2,5);

subplot(2,2,1);

bar3(z)

subplot(2,2,2);

bar3(z)

view(-30,15)

subplot(2,2,3);

bar3(z)

view(0,0)

subplot(2,2,4);

bar3(z)

view(45,-30)

(2)

campos函数设置摄像头的位置，也就是视点位置

drawnow函数更新figure窗口的图形

surf(peaks)

axis vis3d off

for x = -200:5:200

campos([x,10,10])

drawnow

pause(0.1)

End

(3)

vert=10\*[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

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

view(3);

camproj('perspective')

patch('faces',fac,'vertices',vert,'Facecolor','w');

axis vis3d off

for x = -200:5:200

campos([x,100,50])

drawnow

pause(0.1)

end

四、实验结果

①三维图形绘制

1.plot3()函数

MATLAB代码：

figure

t=0:pi/50:10\*pi;

plot3(sin(t),cos(t),t)

运行结果：

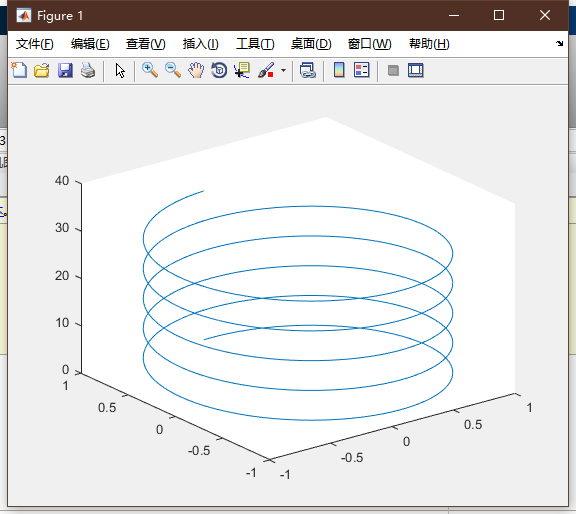


图 1 运行结果1-1-1

MATLAB代码：

figure

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

运行结果

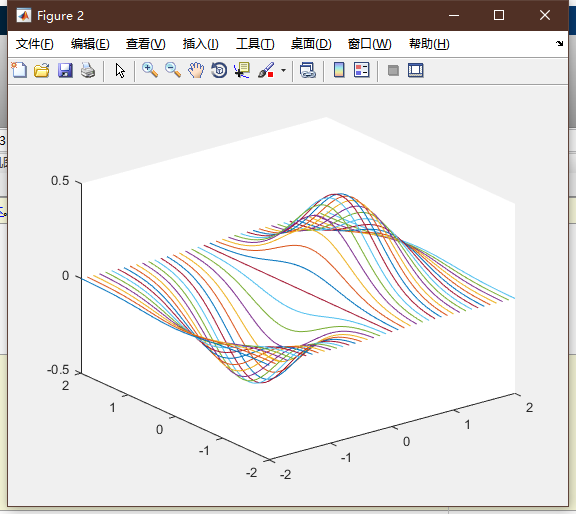


图 2运行结果1-1-2

2.surf()函数

MATLAB代码：

figure

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

figure

    surf(x,y,z)

运行结果

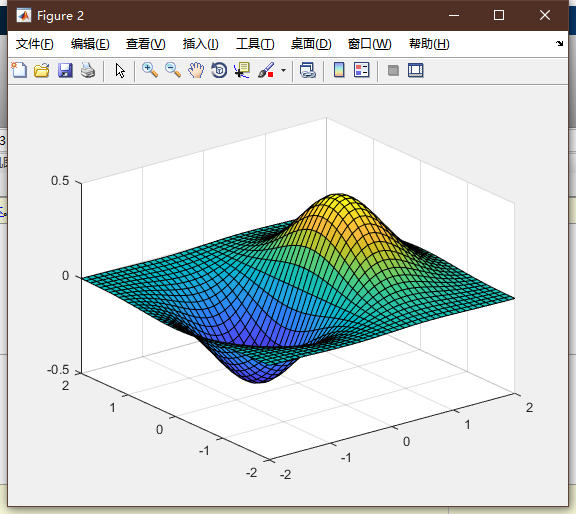


图 3 运行结果1-2-1

3.mesh函数

MATLAB代码：

figure

[x,y]=meshgrid(-2:0.1:2);

z=x.\*exp(-x.^2-y.^2);

plot3(x,y,z)

figure

    mesh(x,y,z)

运行结果：

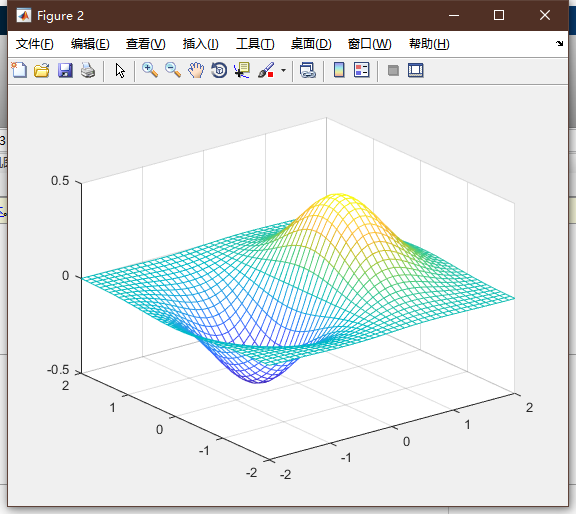


图 4 运行结果 1-3-1

4.patch()函数

MATLAB代码：

figure

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];%顶点

fac=[1 2 3 4;2 6 7 3;4 3 7 8;1 5 8 4;1 2 6 5;5 6 7 8];%面由顶点序号定义

subplot(1,3,1)

patch('faces',fac,'vertices',vert,'Facecolor','w'); %（立方体由顶点和面定义）

view(3);

subplot(1,3,2)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(6),'Facecolor','flat');

view(3);

subplot(1,3,3)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(8),'Facecolor','interp');

view(3);

运行结果：

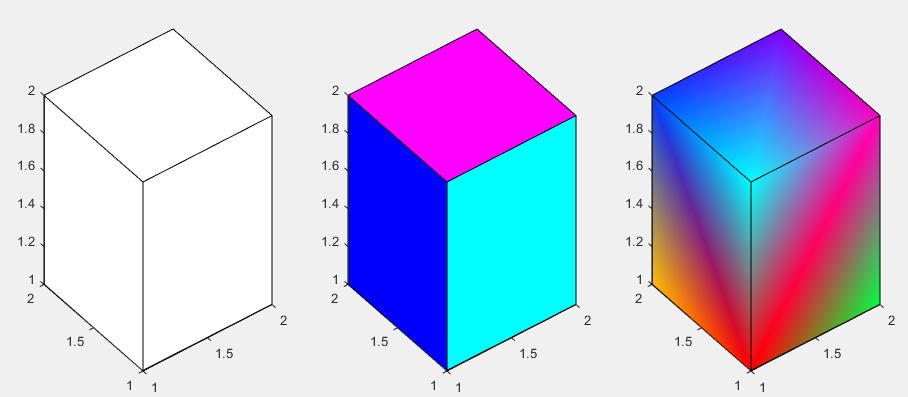


图 5 运行结果1-4-1

②三维图形变换

1.编写一段代码绘制一个球体，然后平移

MATLAB代码：

figure

%画(5 5 5)为圆心，半径为3的球

[x,y,z]  = ellipsoid(5,5,5,3,3,3);

surf(x,y,z)

hold on

%平移到(15 5 5)

[x2,y2,z2] =ellipsoid(15,5,5,3,3,3);

surf(x2,y2,z2)

axis equal

axis([0 20 0 15 0 10]);

运行结果：

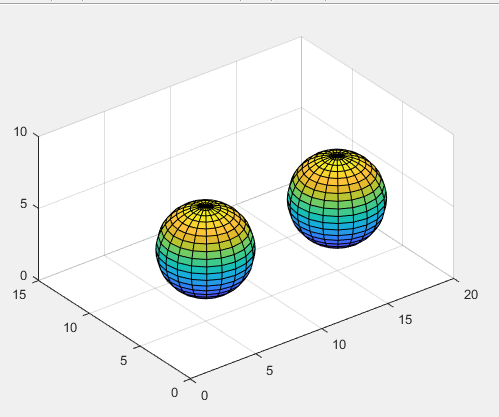


图 6 运行结果2-1-1

2.编写一段代码，绘制一个柱体，然后绕x轴旋转60°

MATLAB代码：

%绘制半径为2，高为4,50分割的圆柱面

figure

[x,y,z] = cylinder(2,50);

z(2,:)=4;

fig(1) = surf(x,y,z,'FaceColor',[0 1 0]);

axis equal

axis([-5 5 -5 5 -5 5]);

hold on

%绘制顶和底

theta = linspace(0,2\*pi,100);

X=2\*cos(theta);

Y=2\*sin(theta);

Z=4\*ones(size(X));

fig(2) = fill3(X,Y,Z,[0,1,0]);

fig(3) = fill3(X,Y,Z-4,[0,1,0]);

view(100,20)

hold on

xlabel('x轴');ylabel('y轴');zlabel('z轴');

%%

%绕x轴旋转60度

direction = [1 0 0];

rotate(fig,direction,60)

运行结果：

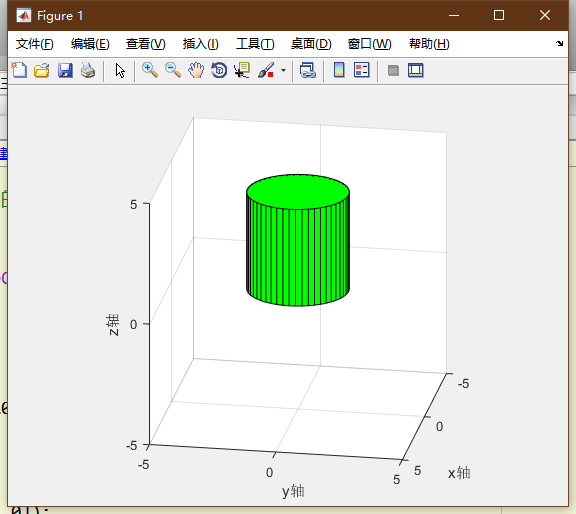


图 7 运行结果2-2-1

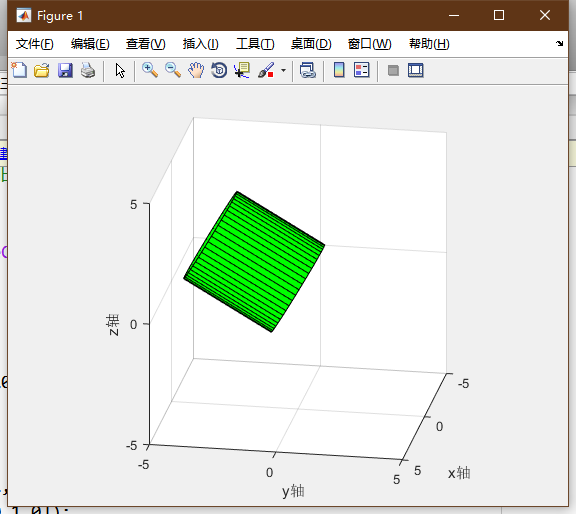


图 8 运行结果 2-2-2

3. 用sphere生成多面体，然后对其进行错切变换

MATLAB代码：

%% sphere

figure

for t = 1:30

    [x,y,z]=sphere(t);

    surf(x,y,z)

    axis equal

    axis([-1 1 -1 1 -1 1]);

    pause(0.2)

end

%%

figure

t = 20;

for e = 1:10

    [x,y,z]=sphere(t);

    %surf(x,y,z)

    T=[1          0   0;

       1+0.1\*e   1   0;

       0          0   1];

    for i=1:t+1

            for j=1:t+1

                a=x(i,j);

                b=y(i,j);

                c=z(i,j);

                m=[a b c]\*T';

                x1(i,j)=m(1);

                y1(i,j)=m(2);

                z1(i,j)=m(3);

            end

    end

    surf(x1,y1,z1)

    axis equal

    axis([-1 1 -2.5 2.5 -1 1]);

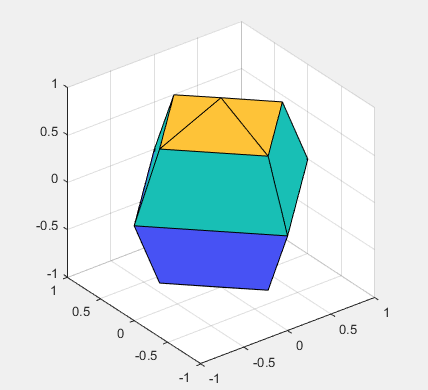
    %view(100,30)

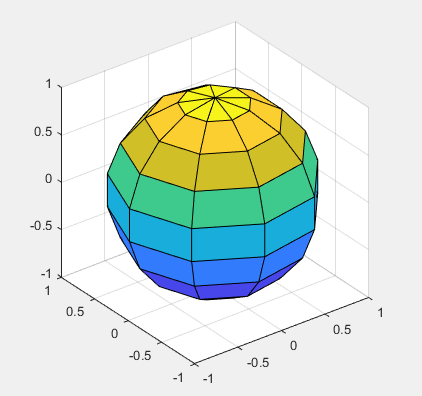
    pause(0.2)

end

运行结果：

动画效果：





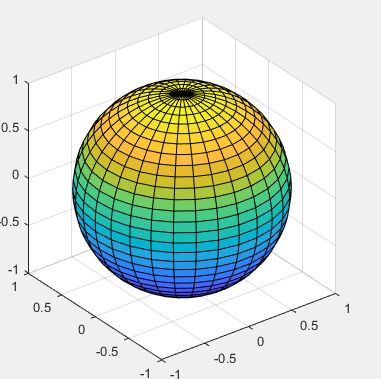
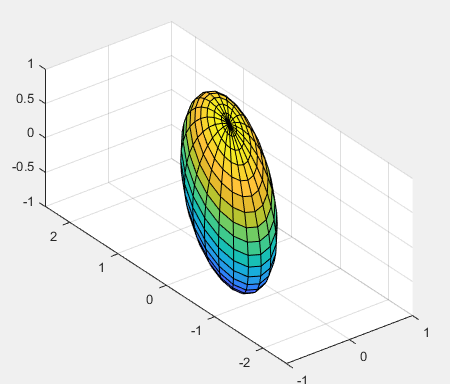


图 9 面的动画



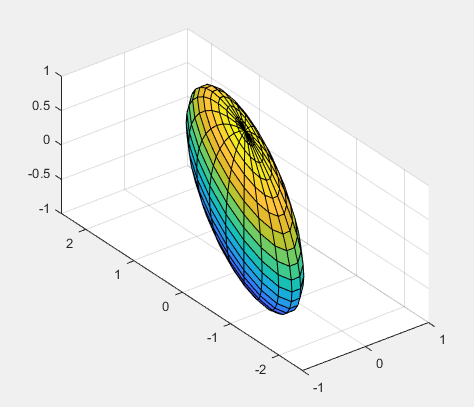


图 10 错切变换的动画

4.三维旋转的实现

MATLAB代码：

for i=-2\*pi:0.5:2\*pi

R=[cos(i) sin(i) 0;-sin(i) cos(i) 0;0 0 1];

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

vert=vert\*R;

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

pause(0.1)

patch('faces',fac,'vertices',vert,'FaceVertexCData',hsv(8),'Facecolor','interp');

view(3);

end

运行结果：

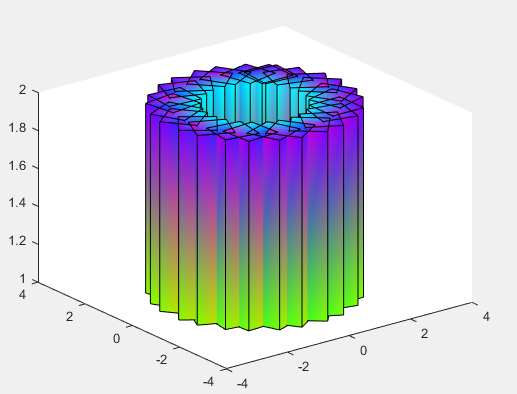


图 11 三维旋转1

更改成围绕X轴旋转:

运行结果：

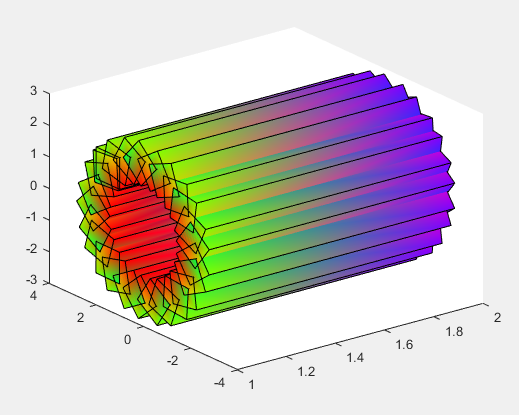


图 12 围绕x轴旋转

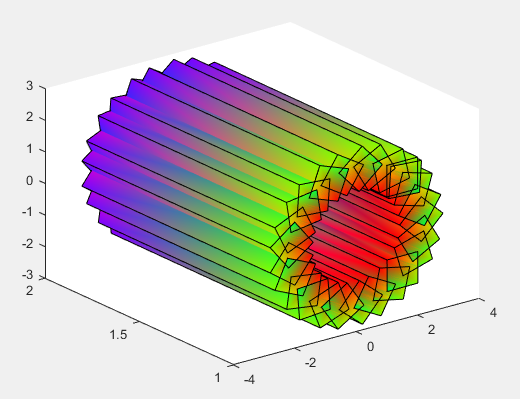


图 13 围绕y轴旋转

5.透视投影和平行投影

MATLAB代码：

figure

vert=[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

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

subplot(1,2,1)

view(3);

patch('faces',fac,'vertices',vert,'Facecolor','w');

axis equal;

subplot(1,2,2)

view(3);

camproj('perspective')

patch('faces',fac,'vertices',vert,'Facecolor','w');

axis equal;

运行结果：

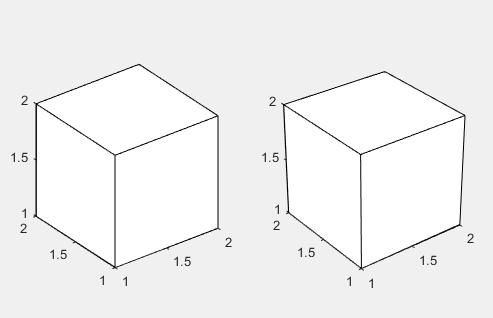


图 14 透视投影和平行投影-1

MATLAB代码：

figure

z=ones(2,5);

subplot(2,2,1);

bar3(z)

subplot(2,2,2);

bar3(z)

view(-30,15)

subplot(2,2,3);

bar3(z)

view(0,0)

subplot(2,2,4);

bar3(z)

view(45,-30)

运行结果：

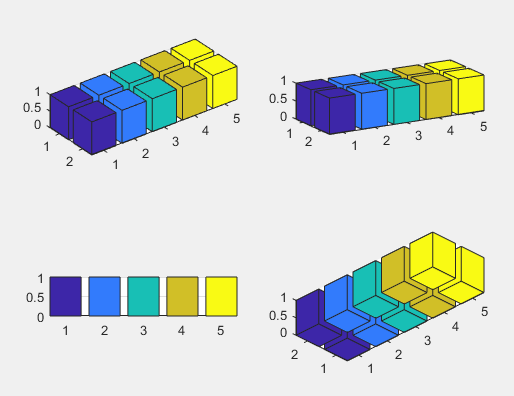


图 15 不同视点下的观察效果

MATLAB代码：

%%

figure

surf(peaks)

axis vis3d off

for x = -200:5:200

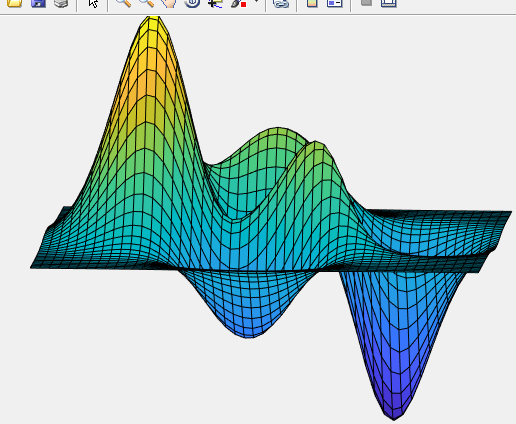
    campos([x,10,10])

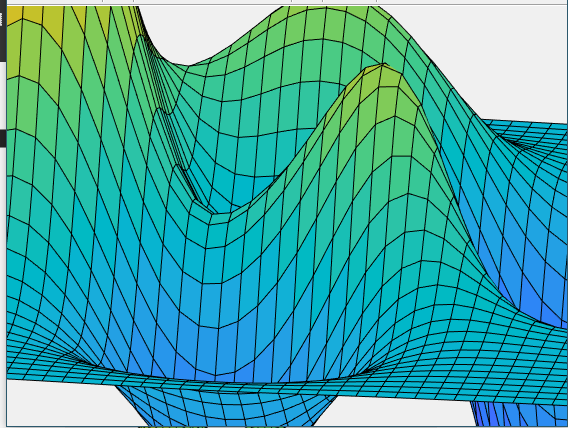
drawnow

pause(0.1)

end

运行结果：





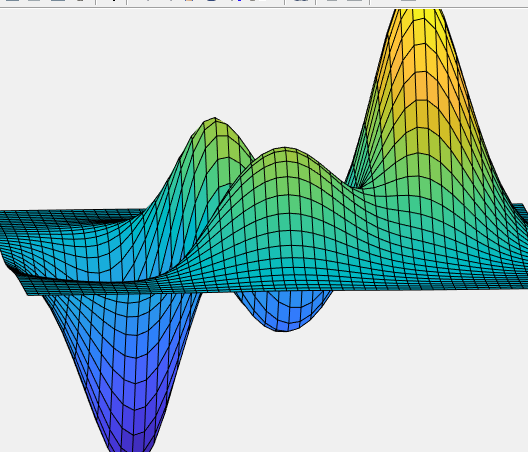


图 16 摄像头位置的改变

MATLAB代码：

figure

vert=10\*[1 1 1;1 2 1;2 2 1;2 1 1;1 1 2;1 2 2;2 2 2;2 1 2];

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

view(3);

camproj('perspective')

patch('faces',fac,'vertices',vert,'Facecolor','w');

axis vis3d off

for x = -200:5:200

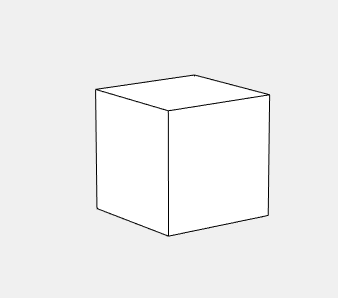
    campos([x,100,50])

drawnow

pause(0.1)

end

运行结果：



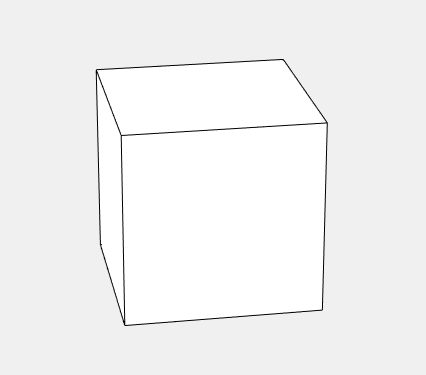


图 17 改变摄像头效果