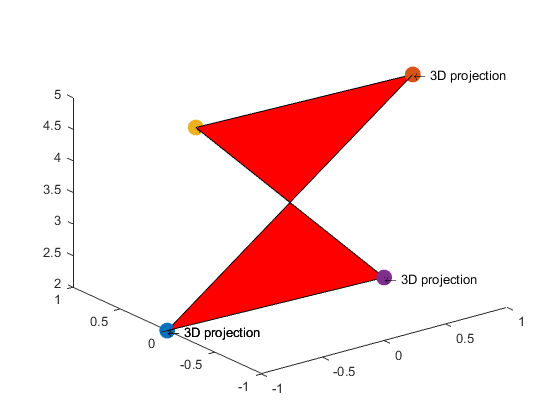
**ASSIGNMENT-3**

Given 3D points are (-1, 0, 2), (1, 0, 5), (0, 1, 4), (0, -1, 3) . when these points are projected the following plot is shown:



Matlab Code:

plot3(-1,0,2,'o','LineWidth',6);

hold on;

plot3(1,0,5,'o','LineWidth',6);

hold on;

plot3(0,1,4,'o','LineWidth',6);

hold on;

plot3(0,-1,3,'o','LineWidth',6);

hold on;

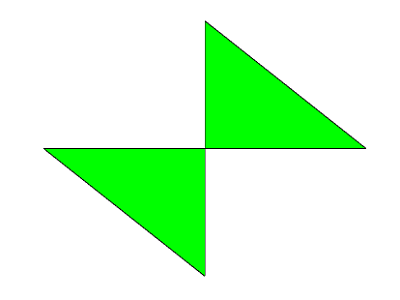
text([-1,1,0,0,-1],[0,0,1,-1,0],[2,5,4,3,2],'\leftarrow 3D projection');

hold on;

fill3([-1,1,0,0],[0,0,1,-1],[2,5,4,3],'r');

hold off;

**a.)** Now by using perspective camera model these 3D points are projected to a 2D space. Plot :



Matlab Code:

function res = projection\_func(X,Z,f)

res=f\*X/Z;

end

V1x=projection\_func(-1,2,1)

V1y=projection\_func(0,2,1)

V2x=projection\_func(1,5,1)

V2y=projection\_func(0,5,1)

V3x=projection\_func(0,4,1)

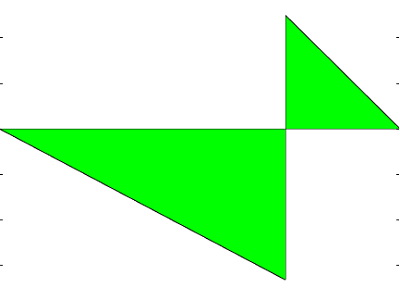
V3y=projection\_func(1,4,1)

V4x=projection\_func(0,3,1)

V4y=projection\_func(-1,3,1)

plot([V1x,V2x,V3x,V4x,V1x],[V1y,V2y,V3y,V4y,V1y]);

**b.)** By using weak projection model if these 3D points are projected the following plot is obtained:



Matlab Code:

%% Weak perspective projection function

function [V1, V2, V3, V4] = weak\_projection\_func(v1, v2, v3, v4)

Zavg=v1(3)+v1(3)+v1(3)+v1(3)/4;

V1(1)=v1(1)/Zavg;

V1(2)=v1(2)/Zavg;

V2(1)=v2(1)/Zavg;

V2(2)=v2(2)/Zavg;

V3(1)=v3(1)/Zavg;

V3(2)=v3(2)/Zavg;

V4(1)=v4(1)/Zavg;

V4(2)=v4(2)/Zavg;

end

Zavg=v1(3)+v1(3)+v1(3)+v1(3)/4;

V1(1)=v1(1)/Zavg;

V1(2)=v1(2)/Zavg;

V2(1)=v2(1)/Zavg;

V2(2)=v2(2)/Zavg;

V3(1)=v3(1)/Zavg;

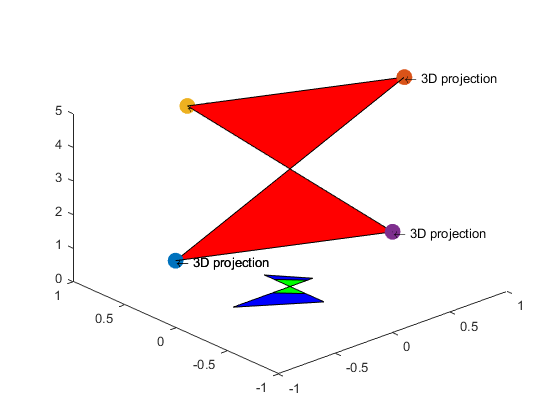
V3(2)=v3(2)/Zavg;

V4(1)=v4(1)/Zavg;

V4(2)=v4(2)/Zavg;

plot([V1(1),V2(1),V3(1),V4(1),V1(1)],[V1(2),V2(2),V3(2),V4(2),V1(2)])

**c.)** Plotting the 3D plot along with plots obtained in perspective camera model and weak projection gives the following plot:



Matlab Code:

plot3(-1,0,2,'o','LineWidth',6);

hold on;

plot3(1,0,5,'o','LineWidth',6);

hold on;

plot3(0,1,4,'o','LineWidth',6);

hold on;

plot3(0,-1,3,'o','LineWidth',6);

hold on;

text([-1,1,0,0,-1],[0,0,1,-1,0],[2,5,4,3,2],'\leftarrow 3D projection');

hold on;

fill3([-1,1,0,0],[0,0,1,-1],[2,5,4,3],'r');

hold on;

plot([V1x,V2x,V3x,V4x,V1x],[V1y,V2y,V3y,V4y,V1y],'k');

fill([V1x,V2x,V3x,V4x,V1x],[V1y,V2y,V3y,V4y,V1y],'b');

hold on;

plot([V1(1),V2(1),V3(1),V4(1),V1(1)],[V1(2),V2(2),V3(2),V4(2),V1(2)],'k');

fill([V1(1),V2(1),V3(1),V4(1),V1(1)],[V1(2),V2(2),V3(2),V4(2),V1(2)],'g')

hold off;

**d.)** Sum of absolute differences(SAD) between the two plots (perspective camera model and weak projection):

SAD value = 0.6678

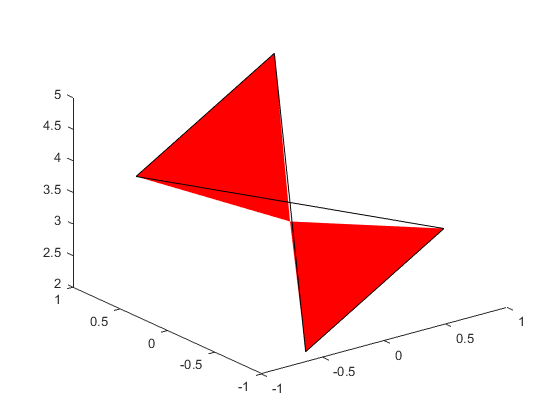
Matlab Code:

function value=SAD\_func(V1,V2,V3,V4,V1a,V2a,V3a,V4a)

value=mod((V1(0)-V1a(0))+(V1(1)-V1a(1))+(V2(0)-V2a(0))+(V2(1)-V2a(1))+(V3(0)-V3a(0))+(V3(1)-V3a(1))+(V4(0)-V4a(0))+(V4(1)-V4a(1)));

end

**c.)** Now rotating the original 3D plot along z-axis by 45 degrees gives the following plot:



Matlab Code:

function rot=rotation\_func(vertex,a)

cosine=cos(a);

sine=sin(a);

R=[cosine -sine 0;sine cosine 0;0 0 1];

rot=mtimes(R,vertex);

end

v1=[-1;0;2]; v2=[1;0;5]; v3=[0;1;4]; v4=[0;-1;3];

v1r=rotation\_func(v1,45);

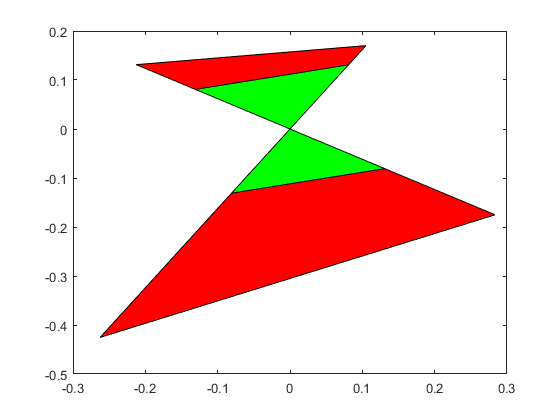
v2r=rotation\_func(v2,45);

v3r=rotation\_func(v3,45);

v4r=rotation\_func(v4,45);

plot3([v1r(1),v2r(1),v3r(1),v4r(1),v1r(1)],[v1r(2),v2r(2),v3r(2),v4r(2),v1r(2)],[v1r(3),v2r(3),v3r(3),v4r(3),v1r(3)]);

Now obtaining plots for weak projection and perspective camera model and plotting them on the same plot:



Matlab Code:

v1rpx=projection\_func(v1r(1),v1r(3),1);

v1rpy=projection\_func(v1r(2),v1r(3),1);

v2rpx=projection\_func(v2r(1),v2r(3),1);

v2rpy=projection\_func(v2r(2),v2r(3),1);

v3rpx=projection\_func(v3r(1),v3r(3),1);

v3rpy=projection\_func(v3r(2),v3r(3),1);

v4rpx=projection\_func(v4r(1),v4r(3),1);

v4rpy=projection\_func(v4r(2),v4r(3),1);

plot([v1rpx,v2rpx,v3rpx,v4rpx,v1rpx],[v1rpy,v2rpy,v3rpy,v4rpy,v1rpy]);

Zwpavg=v1r(3)+v1r(3)+v1r(3)+v1r(3)/4;

V1wp(1)=v1r(1)/Zwpavg;

V1wp(2)=v1r(2)/Zwpavg;

V2wp(1)=v2r(1)/Zwpavg;

V2wp(2)=v2r(2)/Zwpavg;

V3wp(1)=v3r(1)/Zwpavg;

V3wp(2)=v3r(2)/Zwpavg;

V4wp(1)=v4r(1)/Zwpavg;

V4wp(2)=v4r(2)/Zwpavg;

plot([V1wp(1),V2wp(1),V3wp(1),V4wp(1),V1wp(1)],[V1wp(2),V2wp(2),V3wp(2),V4wp(2),V1wp(2)]);

plot([v1rpx,v2rpx,v3rpx,v4rpx,v1rpx],[v1rpy,v2rpy,v3rpy,v4rpy,v1rpy],'k');

fill([v1rpx,v2rpx,v3rpx,v4rpx,v1rpx],[v1rpy,v2rpy,v3rpy,v4rpy,v1rpy],'r');

hold on;

plot([V1wp(1),V2wp(1),V3wp(1),V4wp(1),V1wp(1)],[V1wp(2),V2wp(2),V3wp(2),V4wp(2),V1wp(2)],'k');

fill([V1wp(1),V2wp(1),V3wp(1),V4wp(1),V1wp(1)],[V1wp(2),V2wp(2),V3wp(2),V4wp(2),V1wp(2)],'g');

hold off;

Now obtaining SAD(sum of absolute differences) between the two plot points:

SAD value: 0.629647483672678

Matlab Code:

value\_rot=abs((v1rpx-V1wp(1))+(v1rpy-V1wp(2)))+abs((v2rpx-V2wp(1))+(v2rpy-V2wp(2)))+abs((v3rpx-V3wp(1))+(v3rpy-V3wp(2)))+abs((v4rpx-V4wp(1))+(v4rpy-V4wp(2)));