```
points_3d = [2,2,2; -2,2,2; -2,2,-2; 2,2,-2; 2,-2,2; -2,-2,2;
 -2,-2,-2; 2,-2,-2];
points_2d = [422,323; 178,323; 118,483; 482,483; 438,73; 162,73;
 78,117; 522,117];


%Question 1
% We draw the image points using small circles corresponding
% to each image point.
plot(points_2d(:,1),points_2d(:,2),'o')
title("Camera image points >> ")


%Question 2
% We write a matlab function that takes the arguments as the
 homogenous
% coordinates of the cube and corresponding homogenous coordinates of
 its
% image and returns the matrix P.
p = P_matrix(points_3d, points_2d);


%Question 3
%We display the matrix P obtained from the above function.
disp("Matrix P");
disp(p);


%Question 4
% Now we need to solve the system pm = 0.
% We first take the singular value decomposition of the matrix P.
% The last column vector V obtained will be the 12 elements in row
 order of
% the projection matrix that transformed the cube corner coordinates
 into
% their images.
% We now print the obtained matrix M.
[U,sigma,V_transpose] = svd(p);

M(1,1:4) = V_transpose(1:4,end)';
M(2,1:4) = V_transpose(5:8,end)';
M(3,1:4) = V_transpose(9:12,end)';

disp("Matrix M");
disp(M);


%Question 5
% Now we need to obtain a translation vector which is a null vector of
 M.
% We first find the SVD of M.
% The 4 elements of the last column of V are the homogeneous
```

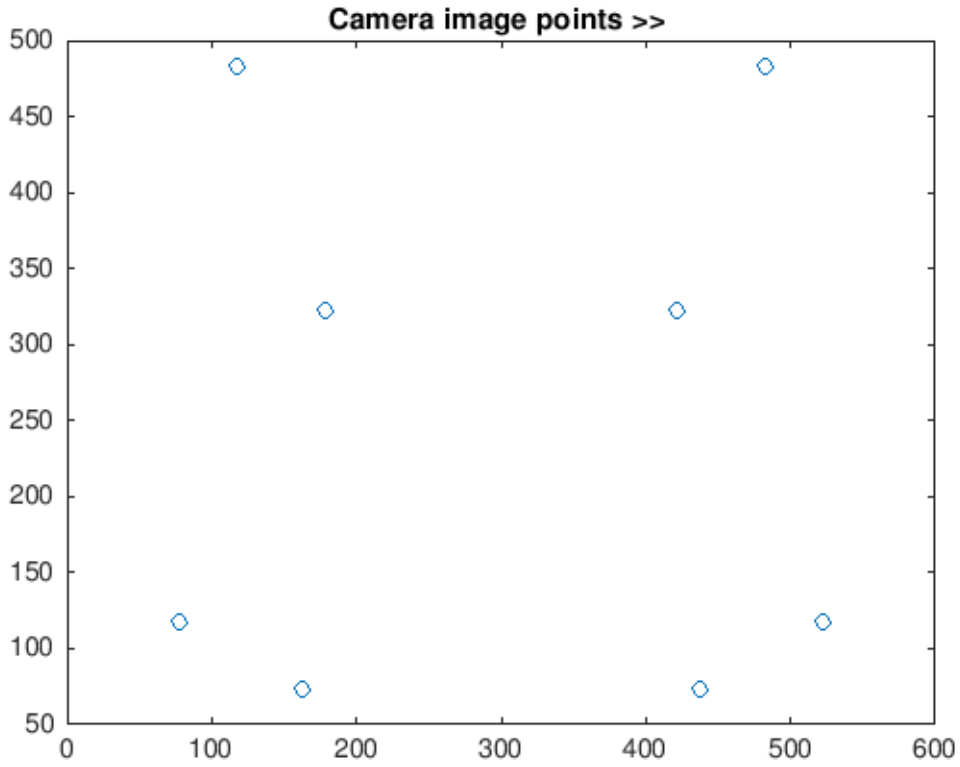
```
% coordinates of the position of the camera center of projection in
 the
% frame of reference of the cube.
[U,sigma,V_transpose] = svd(M);
%M.t = 0, where t = [X_c, Y_c, Z_c, 1]
t = V_transpose(:,end)'/V_transpose(end,end)';
disp("Translation vector: t");
disp(t);


%Question 6
% So now we consider a 3x3 matrix M', which is composed of the first 3
% columns of matrix M.

%Extracts the first 3 columns of M into M'
M_new = M(1:3,1:3);
%Rescale the elements of M' so that the element m_(3,3) becomes equal
 to 1.
M_new = M_new/M_new(end,end);
% Print matrix M( (M_new)
disp("M'")
disp(M_new)


%Question 7
% We now need to do the RQ factorization of M'
cos_x = M_new(3,3)/sqrt(M_new(3,3)^2 + M_new(3,2)^2);
sin_x = -M_new(3,2)/sqrt(M_new(3,3)^2 + M_new(3,2)^2);

%Take the inverse cosine to get theta_x
theta_x = asin(sin_x);

R_x = [1,0,0; 0, cos_x, -sin_x; 0, sin_x, cos_x];

% Now once we hve the R_x matrix, we can useit to compute matrix N.
% Using N = M' * R_x.
N = M_new*R_x;

%We now print the rotation matrix R_x, theta_x and the matrix N.
disp("R_x = ")
disp(R_x)
disp("theta_x = ")
disp(theta_x)
disp("Matrix N")
disp(N)

%Question 8
% We now compute the rotation matrix R_z using the cosine and sine
 form.
cos_z = N(2,2)/sqrt(N(2,1)^2 + N(2,2)^2);
sin_z = -N(2,1)/sqrt(N(2,1)^2 + N(2,2)^2);
% Take the inverse sine to get theta_z
theta_z = asin(sin_z);
```

```
R_z = [cos_z,-sin_z,0;sin_z,cos_z,0;0,0,1];

%Display the value of theta_z which is very small
disp("theta_z = ")
disp(theta_z)

%Question 9
% R = R_1*R_2*R_3
R = R_x*eye(3)*R_z;
K = M_new * R;

%Rescale so that its element K_(3,3) is set to 1.
K = K/K(3,3);
disp("Matrix K")
disp(K)

%u0 and v0 are the pixel coordinates of the camera center.
u0 = K(1,3);
v0 = K(2,3);
%We can compute the focal length as an average of K(1,1) and K(2,2)
f = (K(1,1)+K(2,2))/2;
disp("Focal length of camera in pixels = ")
disp(f)
disp("Pixel coordinates of Image center = ")
disp([u0,v0])
```

```
Matrix P
  Columns 1 through 6

           2           2           2           1           0
  0
           0           0           0           0           2
  2
          -2           2           2           1           0
  0
           0           0           0           0          -2
  2
          -2           2          -2           1           0
  0
           0           0           0           0          -2
  2
           2           2          -2           1           0
  0
           0           0           0           0           2
  2
           2          -2           2           1           0
  0
           0           0           0           0           2
 -2
          -2          -2           2           1           0
  0
           0           0           0           0          -2
 -2
```

```
          -2          -2          -2           1           0
 0
           0           0           0           0          -2
-2
           2          -2          -2           1           0
 0
           0           0           0           0           2
-2

 Columns 7 through 12

           0           0         844         844         844
422
           2           1         646         646         646
323
           0           0        -356         356         356
178
           2           1        -646         646         646
323
           0           0        -236         236        -236
118
          -2           1        -966         966        -966
483
           0           0         964         964        -964
482
          -2           1         966         966        -966
483
           0           0         876        -876         876
438
           2           1         146        -146         146
 73
           0           0        -324        -324         324
162
           2           1        -146        -146         146
 73
           0           0        -156        -156        -156
 78
          -2           1        -234        -234        -234
117
           0           0        1044       -1044       -1044
522
          -2           1         234        -234        -234
117

Matrix M
   -0.1925   -0.0283   -0.0786   -0.7346
   -0.0000   -0.2044   -0.0001   -0.6120
    0.0000    0.0001    0.0003    0.0024

Translation vector: t
   -0.0000   -2.9912   -8.2695    1.0000

M'
 -734.6289 -107.8955 -299.9999
```

```
   -0.0009 -780.1442   -0.2641
    0.0000    0.3597    1.0000

R_x =
    1.0000         0         0
         0    0.9410    0.3384
         0   -0.3384    0.9410

theta_x =
   -0.3452

Matrix N
 -734.6289    0.0000 -318.8125
   -0.0009 -734.0199 -264.2723
    0.0000         0    1.0627

theta_z =
   1.2602e-06

Matrix K
  691.2797    0.0009 -299.9999
    0.0000  690.7067 -248.6780
   -0.0000   -0.0000    1.0000

Focal length of camera in pixels =
  690.9932

Pixel coordinates of Image center =
 -299.9999 -248.6780
```



Published with MATLAB® R2019a