问题包2

问题1

从图像可知，$MD:z+10x-90=0,\ ND:-z+10x+90=0$。由距离公式：

$$PQ=\frac{|40+10\times 20-90|}{\sqrt{1+10^2}}=\frac{150}{\sqrt{101}}\approx 14.9256$$
$$PR=\frac{|-40+10\times 20+90|}{\sqrt{1+10^2}}=\frac{250}{\sqrt{101}}\approx 24.8759$$

又$PQ:10z-x-380=0,\ PR:10z+x-420=0$，联立方程可解得$Q$，$R$的$z$坐标为

$$z_Q=38.5148,\ z_R=42.4752$$

记在成像平面上，P'，P''的x坐标为x_1，x_2。由三角形相似有

$$\frac{x_1}{PQ}=\frac{MS}{MQ}=\frac{10}{z_Q+10},\ \frac{x_2}{PR}=\frac{NT}{NR}=\frac{10}{z_R+10}$$

解得$x_1=3.0765,\ x_2=4.7405$，

$$d=|x_1-x_2|=1.6640$$

问题2

同问题一，考虑两个成像系统中的三角形相似（x_1，x_2为每个成像系统中各自的坐标），有

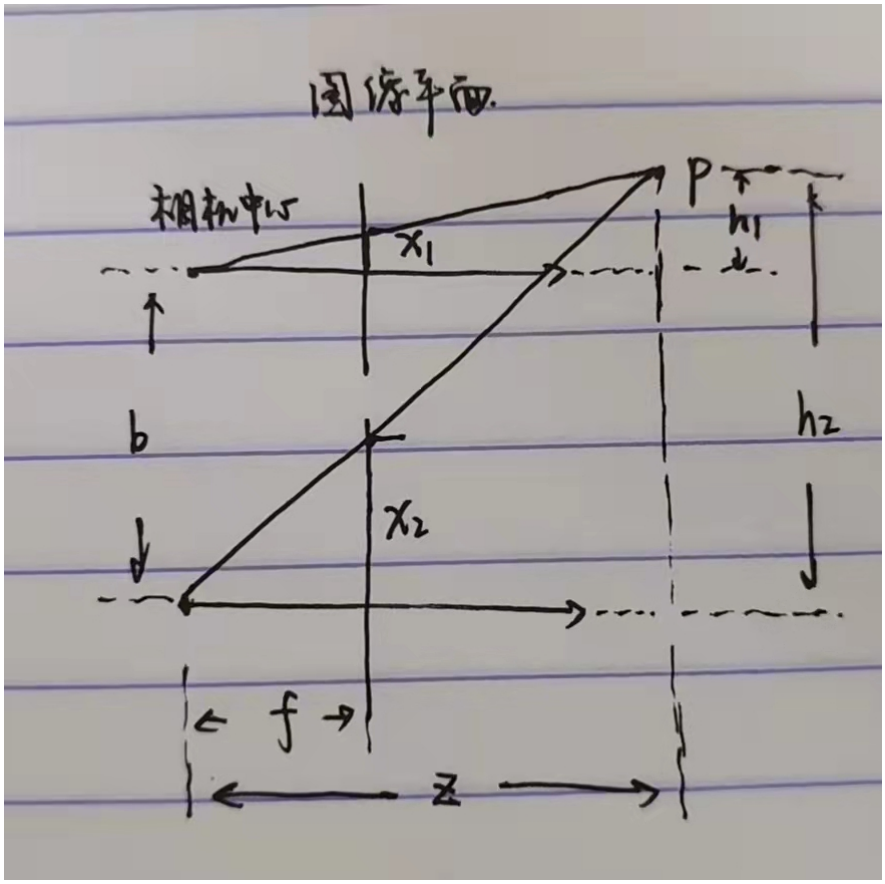
$$\text{Image 1}:\frac{0-x_1}{10-0}=\frac{4-0}{10-4}$$
$$\text{Image 2}:\frac{0-x_2}{10-(-5)}=\frac{4-0}{10-4}$$

因此有$x_1=-\frac{20}{3}$，$x_2=-10$，故

$$x_d=|x_1-x_2|=\frac{10}{3}$$

问题3

记视差为d，基线为b，深度为z，相机焦距为f。如图，$d=x_2-x_1$



由三角形相似

$$\frac{x_1}{h_1} = \frac{x_2}{h_2} = \frac{f}{z}$$

又由于$h_2 - h_1 = b$，故

$$\frac{x_2 - x_1}{h_2 - h_1} = \frac{f}{z} \Rightarrow \frac{d}{b} = \frac{f}{z}$$

因此使用视差d表征立体视觉重建的准确性如下

$$z = b \cdot \frac{f}{d}, \text{ 即} z_{\min} = b \cdot \frac{f}{d_{\max}}, \ z_{\max} = b \cdot \frac{f}{d_{\min}}$$

问题4

证明:
基本矩阵$E = R[t]_\times$，其中R为旋转矩阵（旋转矩阵为正交阵），t为平移向量，$[t]_\times$为t的坐标方阵（坐标方阵为反对称矩阵）。记

$$S = [t]_\times = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix}$$

因此$EE^T = RSS^TR^T$。由于旋转矩阵$R^T = R^{-1}$，故$EE^T = RSS^TR^{-1}$，即EE^T的特征值与SS^T的特征值相同。

对SS^T做正交相似对角化操作，有

$$SS^T = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & t_z & -t_y \\ -t_z & 0 & t_x \\ t_y & -t_x & 0 \end{pmatrix}$$

$$\sim \begin{pmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & t_z & 0 \\ -t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} t_z^2 & 0 & 0 \\ 0 & t_z^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

故特征多项式为$(\lambda - t_z^2)^2\lambda = 0$。即$EE^T$的一个特征值为$0$，另两个相等。

由于E的奇异值等于EE^T的特征值，故基本矩阵E的奇异值一个特征值为0，另两个相等。

问题5

找到如下两张经典图像及其初始透射变换矩阵数据进行**stereo pair rectification**

其中原始的透视变换矩阵为

```
Po1 = 3×4
          976.50           53.82         -239.80       387500.00
           98.49          933.30          157.40       242800.00
            0.58            0.11            0.81         1118.00

Po2 = 3×4
          976.70           53.76         -240.00        40030.00
           98.68          931.00          156.71       251700.00
            0.58            0.11            0.81         1174.00
```

参考*Computer Vision: Algorithms and Applications*中关于*Rectification*的部分，编写MATLAB代码计算，得到校正后的转移矩阵和透射变换矩阵为

```
T1 = 3×3
    1.05        0.02     -157.46
    0.03        1.00      -11.20
    0.00        0.00        0.93

T2 = 3×3
    1.05        0.02     -157.31
    0.03        1.00      -10.99
    0.00       -0.00        0.93

Pn1 = 3×4
  932.92       56.12     -375.31    234103.66
  117.52      932.46      141.89    240180.60
    0.69        0.11        0.72      1101.87

Pn2 = 3×4
  932.92       56.12     -375.31   -137987.50
  117.52      932.46      141.89    240180.60
    0.69        0.11        0.72      1101.87
```

对双目视觉图像进行矫正，结果如下

改变图像显示范围参数（shape）可得矫正图像如下

left image after Rectification

right image after Rectification

代码如下（使用MATLAB的实时编译器编写代码，代码在`.mlx`文件中）：

```
img1 = imread('left.png');
img2 = imread('right.png');

im1 = im2double(img1);
im2 = im2double(img2);

Po1 = double([976.5 53.82 -239.8 387500
       98.49 933.3 157.4 242800
       .579 .1108 .8077 1118]);
Po2 = double([976.7 53.76 -240 40030
       98.68 931.0 156.71 251700
       .5766 .1141 .8089 1174]);
[T1, T2, Pn1, Pn2] = rectify(Po1, Po2);

left = imwarp(im1, projective2d(T1'));
right = imwarp(im2, projective2d(T2'));

tiledlayout(2,1)
nexttile
imshowpair(img1, img2,'montage');
title('Original Images');
nexttile
```

```
imshowpair(left, right,'montage');
title('Rectified Stereo Images');

function [T1,T2,Pn1,Pn2] = rectify(Po1,Po2)

    % RECTIFY: compute rectification matrices

    % factorize old PPMs
    [A1,R1,~] = art(Po1);
    [A2,~,~] = art(Po2);

     % optical centers (unchanged)
    c1 = -inv(Po1(:,1:3))*Po1(:,4);
    c2 = -inv(Po2(:,1:3))*Po2(:,4);

    % new x axis (= direction of the baseline)
    v1 = (c1-c2);
    % new y axes (orthogonal to new x and old z)
    v2 = cross(R1(3,:)',v1);
    % new z axes (orthogonal to baseline and y)
    v3 = cross(v1,v2);

    % new extrinsic parameters
    R = [v1'/norm(v1)
         v2'/norm(v2)
         v3'/norm(v3)];
    % translation is left unchanged

    % new intrinsic parameters (arbitrary)
    A = (A1 + A2)./2;
    A(1,2)=0; % no skew
    % new projection matrices
    Pn1 = A * [R (-R*c1)];
    Pn2 = A * [R (-R*c2)];

    % rectifying image transformation
    T1 = Pn1(1:3,1:3)/Po1(1:3,1:3);
    T2 = Pn2(1:3,1:3)/Po2(1:3,1:3);

end

function [A,R,t] = art(P)
    % ART: factorize a PPM as P=A*[R;t]
    Q = inv(P(1:3, 1:3));
    [U,B] = qr(Q);

    R = inv(U);
    t = B*P(1:3,4);
    A = inv(B);
    A = A./A(3,3);
end
```