# Computer Vision Exercise 2

## Spatial Intersection and Resection

Yi Wang 3371561 (resubmit)

# I. Processing Steps

## 1. Compute projection matrix & pixel coordinates

Compute projection matrix:

```
$ImageID___________________________________________________(ORI_Ver_1.0)
        8919
$IntOri_FocalLength_______________________________________________[mm]
          5.98760000
$IntOri_PixelSize______(x|y)______________________________________[mm]
           0.001530         0.001530
$IntOri_SensorSize_____(x|y)___________________________________[pixel]
              3000              4000
$IntOri_PrincipalPoint_(x|y)________________________________[pixel]
        1499.50000000                 1999.50000000
$IntOri_CameraMatrix_____________________________(ImageCoordinateSystem)
        -3913.4640522      -0.00000000                1499.50000000
         -0.00000000   3913.46405229                1999.50000000
          0.00000000       0.00000000                   1.00000000
$ExtOri_RotationMatrix____________________(World->ImageCoordinateSystem)
        0.540702141171              -0.840413805693              -0.036685552114
        0.833792484528               0.541202069167              -0.109043170690
        0.111495682777               0.028371738229               0.993359832685
$ExtOri_TranslationVector________________________(WorldCoordinateSystem)
        513028.292610               5427687.148050                490.883330
```

$$\frac{1}{m_x}, \frac{1}{m_y}$$

$$K = \begin{bmatrix} m_x & & \\ & m_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} c & & p_x \\ & c & p_y \\ & & 1 \end{bmatrix}$$

$$\mathbf{R}$$

$$\tilde{\mathbf{X}}_0$$

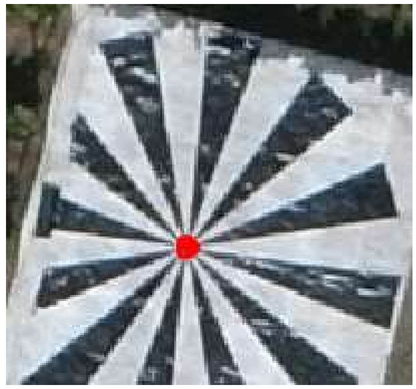
$$[\mathbf{R} \,|\, \mathbf{t}] = [\mathbf{R} \quad -\mathbf{R} \cdot \tilde{\mathbf{X}}_0]$$

$$\mathrm{P} = \mathrm{K}[\mathrm{R} \,|\, \mathrm{t}] = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix}$$

Compute pixel coordinates:

$$\mathbf{x} = \mathbf{P} \cdot \mathbf{X}$$

## 2. Measure one object



## 3. Spatial intersection

For unknown object coordinate $\boldsymbol{X}$ at least two pixel measures $\boldsymbol{x}$ and $\boldsymbol{x'}$ from two cameras with known projection matrix $\boldsymbol{P}$ and $\boldsymbol{P'}$are available:

$$\mathbf{x} = \mathbf{P} \cdot \mathbf{X} \qquad \mathbf{x'} = \mathbf{P'} \cdot \mathbf{X}$$

From which we can build identity equation:

$$\mathbf{x} \times (\mathbf{P}\mathbf{X}) = \mathbf{0} \qquad \mathbf{x'} \times (\mathbf{P'}\mathbf{X}) = \mathbf{0}$$

With

$$\mathrm{P} = \mathrm{K}\left[\mathrm{R} \mid \mathrm{t}\right] = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix}$$

Then we get

$$\begin{aligned} x(\mathbf{p}^{3\mathrm{T}}\mathbf{X}) - (\mathbf{p}^{1\mathrm{T}}\mathbf{X}) &= 0 \\ y(\mathbf{p}^{3\mathrm{T}}\mathbf{X}) - (\mathbf{p}^{2\mathrm{T}}\mathbf{X}) &= 0 \\ x(\mathbf{p}^{2\mathrm{T}}\mathbf{X}) - y(\mathbf{p}^{1\mathrm{T}}\mathbf{X}) &= 0 \end{aligned} \qquad \begin{aligned} x'(\mathbf{p}'^{3\mathrm{T}}\mathbf{X}) - (\mathbf{p}'^{1\mathrm{T}}\mathbf{X}) &= 0 \\ y'(\mathbf{p}'^{3\mathrm{T}}\mathbf{X}) - (\mathbf{p}'^{2\mathrm{T}}\mathbf{X}) &= 0 \\ x'(\mathbf{p}'^{2\mathrm{T}}\mathbf{X}) - y'(\mathbf{p}'^{1\mathrm{T}}\mathbf{X}) &= 0 \end{aligned}$$

For both x and x’, the third equation is linearly dependent on the other two, therefore we eliminate it and get:

$$\mathbf{AX} = \mathbf{0}$$

where

$$
\mathbf{A} = \begin{pmatrix} x\mathbf{p}^{3\mathrm{T}} - \mathbf{p}^{\mathrm{IT}} \\ y\mathbf{p}^{3\mathrm{T}} - \mathbf{p}^{2\mathrm{T}} \\ x'\mathbf{p}'^{3\mathrm{T}} - \mathbf{p}'^{\mathrm{T}} \\ y'\mathbf{p}'^{3\mathrm{T}} - \mathbf{p}'^{2\mathrm{T}} \end{pmatrix} = \begin{pmatrix} x\mathbf{p}(3;\mathrm{i}) - \mathbf{p}(1,:) \\ y\mathbf{p}(3,:) - \mathbf{p}(2,:) \\ x'\mathbf{p}'(3,:) - \mathbf{p}'(1;) \\ y'\mathbf{p}'(3,:) - \mathbf{p}'(2,:) \end{pmatrix}
$$

And

$$
\mathbf{X} = \begin{pmatrix} X & Y & Z & W \end{pmatrix}^T
$$

Solve this equation using singular vector decomposition, we can get the final object coordinates.

## 4. Back transformation and errors

We apply back transformation with

$$
\mathbf{x} = \mathbf{P} \cdot \mathbf{X} \qquad\qquad \mathbf{x}' = \mathbf{P}' \cdot \mathbf{X}
$$

And calculate the error with

$$
\mathbf{v'v} = \sum \left( \mathbf{x}_{meas} - \mathbf{x}_{trafo} \right)^2 \qquad \sigma_0 = \sqrt{\frac{\mathbf{v'v}}{2 \cdot n_{images} - 3}}
$$

## 5. Direct Linear Transformation

For the direct linear transformation we use the following equation, where P matrix is what we need.

$$
\mathbf{x}_i \times \mathbf{P} \cdot \mathbf{X}_i = \begin{pmatrix} y_i \mathbf{p}^{3T} \mathbf{X}_i - w_i \mathbf{p}^{2T} \mathbf{X}_i \\ w_i \mathbf{p}^{1T} \mathbf{X}_i - x_i \mathbf{p}^{3T} \mathbf{X}_i \\ x_i \mathbf{p}^{3T} \mathbf{X}_i - y_i \mathbf{p}^{2T} \mathbf{X}_i \end{pmatrix} = 0
$$

This can be rewritten into

$$
\begin{pmatrix} \mathbf{0}^T & -w_i \mathbf{X}_i^T & y_i \mathbf{X}_i^T \\ w_i \mathbf{X}_i^T & \mathbf{0}^T & -x_i \mathbf{X}_i^T \\ -y_i \mathbf{X}_i^T & x_i \mathbf{X}_i^T & \mathbf{0}^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = 0
$$

The third row is linear dependent on the first two rows, therefore we can eliminate it:

$$\begin{pmatrix} \mathbf{0}^T & -w_i\mathbf{X}_i^T & y_i\mathbf{X}_i^T \\ w_i\mathbf{X}_i^T & \mathbf{0}^T & -x_i\mathbf{X}_i^T \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = A_i\mathbf{p} = 0$$

To solve this equation we need more than 6 pairs of points:

$$\underset{2n\times 12}{\mathbf{A}} \cdot \mathbf{p} = 0$$

Similar to before, we use singular vector decomposition to calculate the P matrix.

$$[\cup, \mathrm{D}, \mathrm{V}] = \mathrm{svd}(\mathrm{A}, 0)$$
$$\%\ \text{Extract homography } P = \text{reshape}\ (V(:, 12), 4, 3)'$$

## 6. Re-mapping and comparison

Similar to task 1, $\mathbf{x} = \mathbf{P} \cdot \mathbf{X}$ is used to re-compute the mapping

## 7. Reconstruct the camera parameters

### a) translation vector X0

$\boldsymbol{X}$0 can be computed from Singular Value Decomposition (SVD) of P:

$$[U, D, V] = svd(P, 0)$$

Where X0 is the last column of V

### b) camera matrix K and rotation matrix R

$$\boldsymbol{P} = \boldsymbol{K}\left[\boldsymbol{R} \mid \boldsymbol{t}\right] = \boldsymbol{K}\left[\boldsymbol{R} \mid -\boldsymbol{R}\tilde{\boldsymbol{X}}_{\boldsymbol{0}}\right] = \boldsymbol{K}\boldsymbol{R}\left[\boldsymbol{I}_{\boldsymbol{3}} \mid -\tilde{\boldsymbol{X}}_{\boldsymbol{0}}\right] = \boldsymbol{M}\left[\boldsymbol{I}_{\boldsymbol{3}} \mid -\tilde{\boldsymbol{X}}_{\boldsymbol{0}}\right]$$

Where $\boldsymbol{M} = \boldsymbol{KR}$ is the left 3x3-Sub-Matrix of P, With $M^{-1} = R^T K^{-1}$ matrix $\boldsymbol{M}$ can be decomposed into QR decomposition:

$$[q, r] = qr(M^{-1})$$

And

$$R = q^{-1}$$

$$K = r^{-1}$$

PS: we have to normalize the K and X0 by the scale factor.

# II. Results

## 1. Fundamental matrix & Pixel coordinates

Table1. Fundamental matrix P

| | | | |
|---|---|---|---|
| 3497.48715935772 | 2083.00231625386 | 1869.39205798182 | -13101021647.8206 |
| 2323.73516206219 | -3332.09572565785 | 997.544343618571 | 16893075551.3581 |
| 0.113062965097000 | -0.0513027902360000 | 0.992262460056000 | 219946.269494256 |

Table2. Pixel coordinates

| ID | 15 | 24 | 25 | 32 | 37 | 98 | 99 |
|---|---|---|---|---|---|---|---|
| X (pix) | 3182.8948 | 735.9314 | 626.0374 | 3883.0065 | 4277.4454 | 1950.6201 | 1831.1469 |
| Y (pix) | 2309.9545 | 2663.2930 | 236.2401 | 2654.6698 | 235.6924 | 533.9252 | 502.6307 |

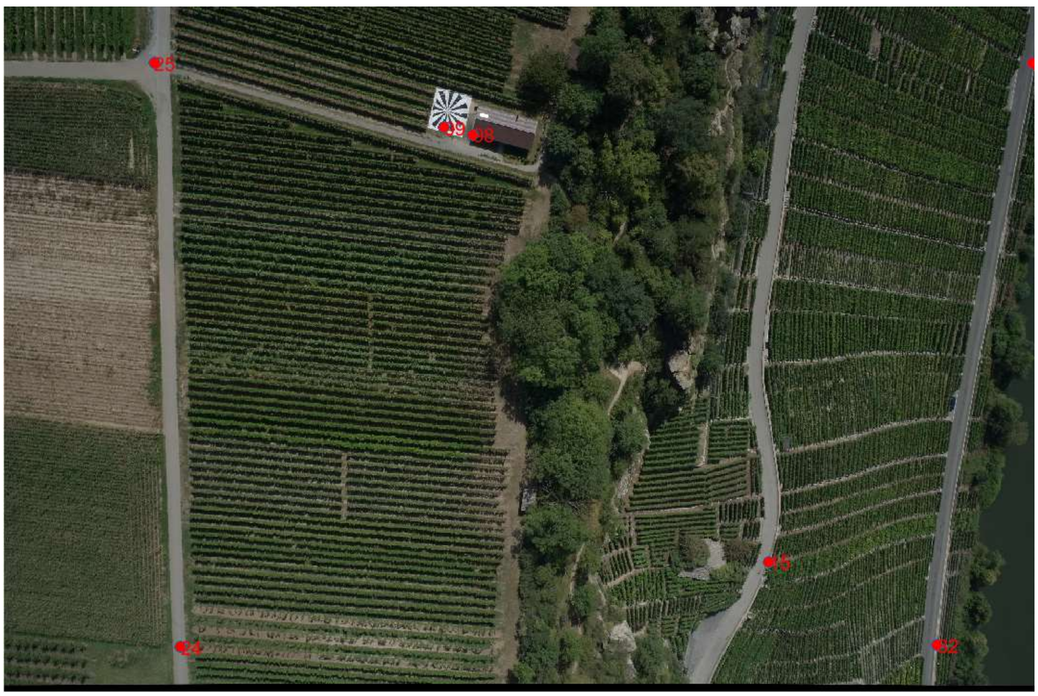



Fig1. Object points in image

## 2. Measure an object

## 3. Object Coordinates

Table3. Calculated object coordinates

| | |
|---|---|
| X (m) | 512997.1910 |
| Y (m) | 5427680.3764 |
| Z (m) | 326.5378 |

## 4. Back transformation errors

Table4. Differences between pixel coordinates of origin points and back transformed points

| Photo_ID | 20813 | 20814 | 20815 | 20816 | 20849 | 20850 | 20851 | 20852 |
|---|---|---|---|---|---|---|---|---|
| Diff_X (pix) | 2.8041 | 0.9643 | -6.1380 | 1.6560 | -5.6518 | 2.7380 | 2.8864 | -0.4562 |
| Diff_Y (pix) | 1.0683 | 0.5424 | -0.2438 | -0.8561 | 0.1675 | -2.2750 | 2.8403 | -0.3507 |

Table5. total transformation error

| | |
|---|---|
| $\sigma_x$ (pix) | 2.7342 |
| $\sigma_y$ (pix) | 1.0962 |

## 5. Direct Linear Transformation

Table6. P_DLT (without centralization)

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | -0.6128 |
| 0 | 0 | 0 | 0.7902 |
| 0 | 0 | 0 | 0 |

Table7. P_DLT (after centralization)

| | | | |
|---|---|---|---|
| 0.0052 | 0.0031 | 0.0028 | -0.8836 |
| 0.0035 | -0.0050 | 0.0015 | -0.4681 |
| 0 | 0 | 0 | -0.0003 |

## 6. Remapping Difference & Error

Table8. Remapping differences

| ID | 15 | 24 | 25 | 32 | 37 | 98 | 99 |
|---|---|---|---|---|---|---|---|
| X (pix) | 3.15e-08 | -1.32e-08 | -1.64e-08 | 1.13e-07 | 1.05e-07 | -5.16e-08 | -4.64e-08 |
| Y (pix) | 1.39e-08 | -4.08e-08 | -6.92e-09 | 8.10e-08 | 5.60e-09 | -8.54e-09 | -1.85e-08 |

Table9. Remapping error

| | |
|---|---|
| $\sigma_x$ (pix) | 5.23e-08 |
| $\sigma_y$ (pix) | 2.85e-08 |

# 7. Reconstruct Camera Parameters

## a) reconstructed camera parameters

X0=

$$\begin{pmatrix} 512980.9951 \\ 5427701.5267 \\ 514.7943 \end{pmatrix}$$

K=

$$\begin{pmatrix} -3933.3636 & 1.29e-08 & 2143.5000 \\ 0 & 3933.3636 & 1423.5000 \\ 0 & 0 & 1 \end{pmatrix}$$

R=

$$\begin{pmatrix} -0.827570749775 & -0.557530411593 & 0.065471324026 \\ 0.549857636162 & -0.828569770063 & -0.105492730048 \\ 0.113062965097 & -0.051302790236 & 0.992262460056 \end{pmatrix}$$

## b) Differences

X0_difference=

$$\begin{pmatrix} 1.59e-08 \\ 1.86e-09 \\ -3.23e-08 \end{pmatrix}$$

K_difference=

$$\begin{pmatrix} 5.15e-07 & 1.29e-08 & -3.60e-07 \\ 0 & -5.33e-07 & -2.07e-07 \\ 0 & 0 & 0 \end{pmatrix}$$

R_difference=

$$\begin{pmatrix} -1.58e-12 & 4.85e-13 & -2.06e-11 \\ 2.02e-12 & 5.89e-13 & 7.57e-12 \\ -2.07e-11 & -5.34e-12 & 2.63e-12 \end{pmatrix}$$

III. re-submit remark

The large differences (my first submission) between origin and remap coordinates and camera parameters in task 5,6 and 7 results from the round error of MATLAB, which moreover comes from the large scale difference between object horizon and vertical coordinates (XY has a scale of $10^6$ but Z only $10^3$, this leads to a round error when calculating P matrix). To solve this problem, we could simply centralize the object coordinates at the very beginning by subtracting the mean value (see code file also: four more lines in task 5).

```
X_20851_m = repmat(mean(X_20851'),7,1)';

X_20851_m(4,:) = 0;

X_20851_c = X_20851-X_20851_m;  % Centralization
```