Computer Vision
and Geometry Lab

Computer Vision

Exercise Session 1

Camera Calibration

- Intrinsic parameters
  - K
  - Radial distortion coefficients

2D points

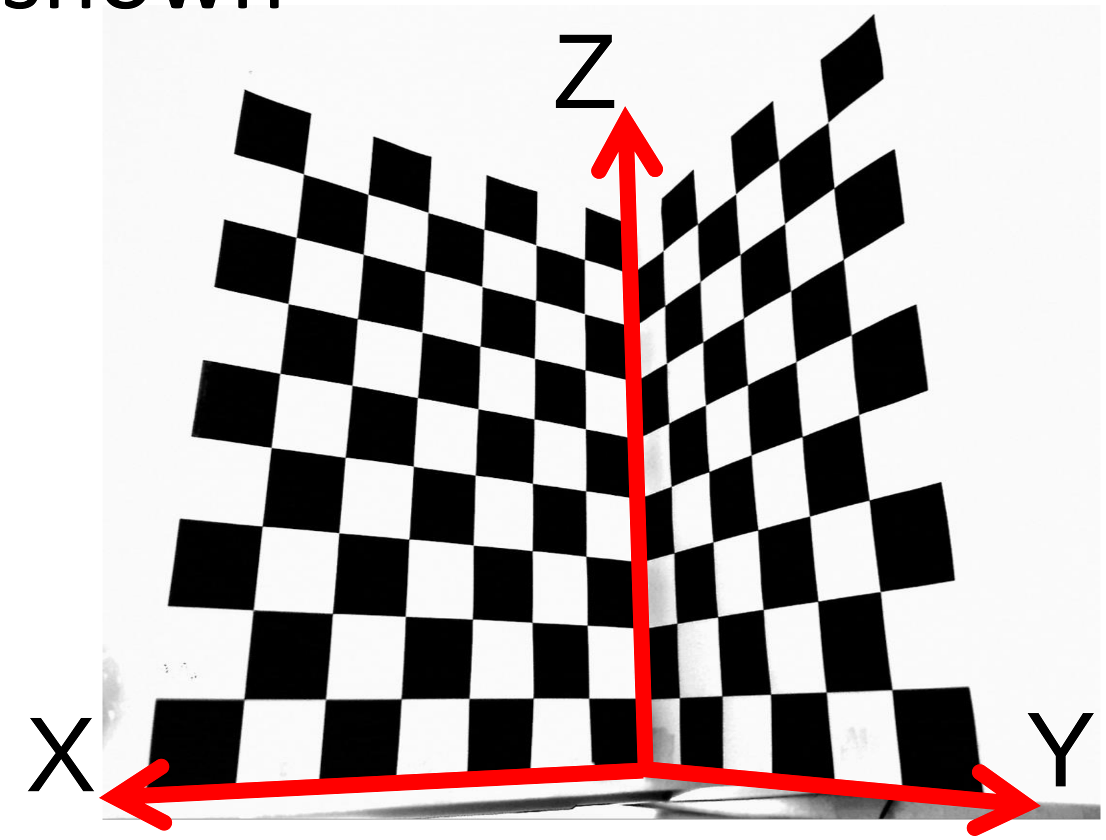
3D points

$$\mathbf{x} \propto \mathbf{P}\mathbf{X}$$

$$\mathbf{x} \propto \mathbf{K}[\mathbf{R} \mid \mathbf{t}]\mathbf{X}$$

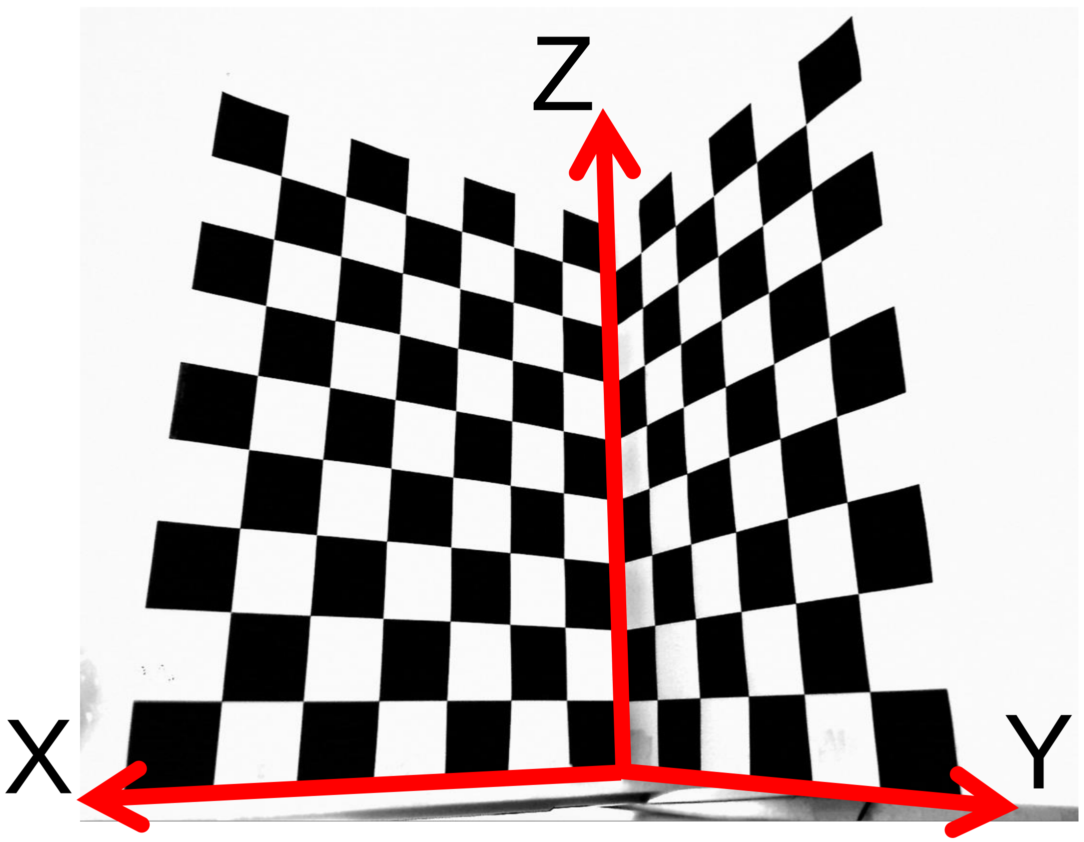
Camera Calibration

- We provide you with an input image
- Need to click points in the image and manually enter the 3D position
- Use the coordinate system as shown



Taking your own pictures (optional)

- Use your own camera
- Build your own calibration object
  - Print checkerboard patterns
  - Stich to two orthogonal planes
- Use constant settings
  - No autofocus
  - Don’t change the zoom



Camera Calibration

- 4 Tasks:
  - Data normalization
  - Direct Linear Transform (DLT)
  - Gold Standard algorithm
  - MATLAB Calibration Toolbox (optional)
- Good reference:
  Multiple View Geometry in computer vision
  (Richard Hartley & Andrew Zisserman)

Data Normalization

- Required for numeric stability
- Shift the centroid of the points to the origin
- Scale the points so that the mean distance to the origin is 1.
- Determine $\widehat{\mathbf{P}}$ using normalized points.
- Determine $\mathbf{P} = \mathbf{T}^{-1}\widehat{\mathbf{P}}\mathbf{U}$

$$\mathbf{T} = \begin{bmatrix} s_{2D} & & c_x \\ & s_{2D} & c_y \\ & & 1 \end{bmatrix}^{-1}$$

$$\mathbf{U} = \begin{bmatrix} s_{3D} & & & c_x \\ & s_{3D} & & c_y \\ & & s_{3D} & c_z \\ & & & 1 \end{bmatrix}^{-1}$$

Direct Linear Transform (DLT)

$$[\mathbf{x}_i]_{\times} \mathbf{P}\mathbf{X}_i = \mathbf{0}$$

$$\mathbf{x} = \begin{pmatrix} x \\ y \\ w \end{pmatrix} \qquad \mathbf{X} = \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix}$$

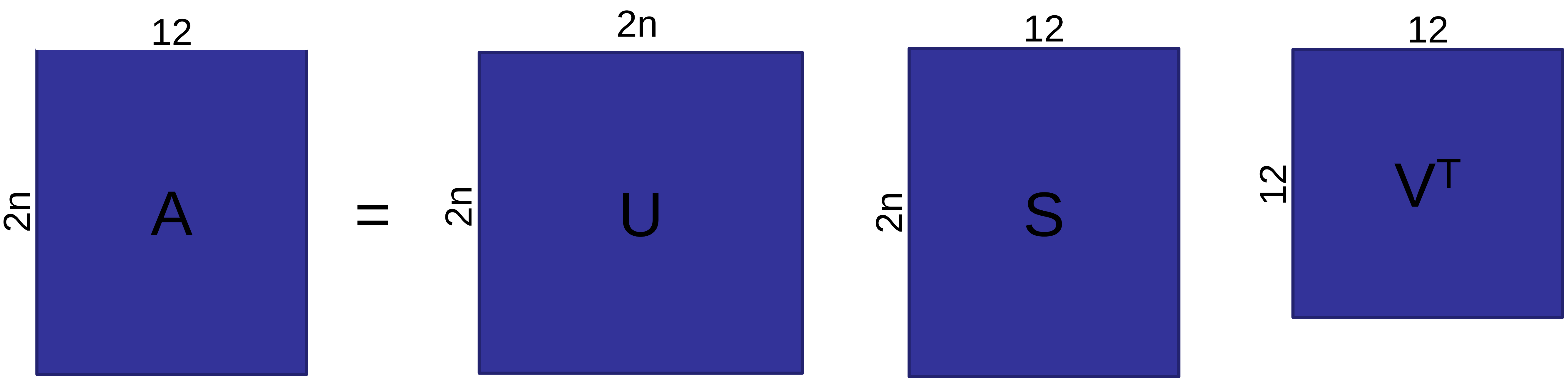
$$\rightarrow \mathbf{A}_i\mathbf{P} = \begin{bmatrix} w_i\boldsymbol{X}_i{}^T & 0^T & -x_i\boldsymbol{X}_i{}^T \\ 0^T & -w_i\boldsymbol{X}_i{}^T & y_i\boldsymbol{X}_i{}^T \end{bmatrix} \begin{pmatrix} \mathbf{P}^1 \\ \mathbf{P}^2 \\ \mathbf{P}^3 \end{pmatrix} = 0$$

$$\downarrow$$

$$\begin{bmatrix} X_{ix} & X_{iy} & X_{iz} & 1 & 0 & 0 & 0 & 0 & -x_iX_{ix} & -x_iX_{iy} & -x_iX_{iz} & -x_i \\ 0 & 0 & 0 & 0 & -X_{ix} & -X_{iy} & -X_{iz} & -1 & y_iX_{ix} & y_iX_{iy} & y_iX_{iz} & y_i \end{bmatrix} \begin{pmatrix} P_{1,1} \\ P_{1,2} \\ \vdots \\ P_{3,3} \\ P_{3,4} \end{pmatrix} = 0$$

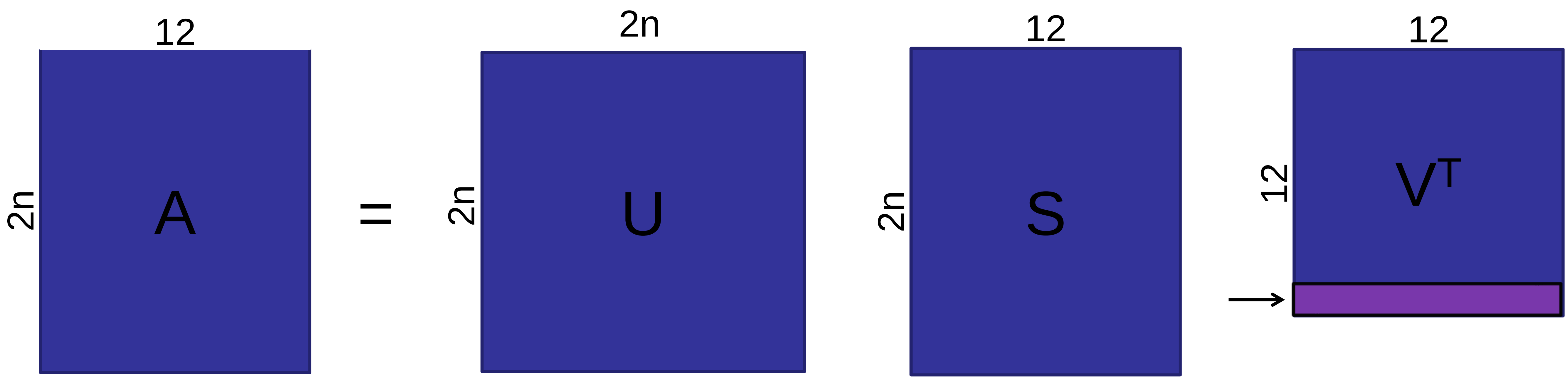
Direct Linear Transform (DLT)

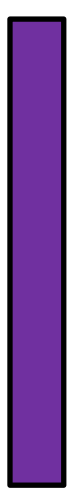
- Singular Value Decomposition



Direct Linear Transform (DLT)

- Singular Value Decomposition





$$= \begin{pmatrix} P_{1,1} \\ P_{1,2} \\ \vdots \\ P_{3,3} \\ P_{3,4} \end{pmatrix} \longrightarrow \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & P_{1,4} \\ P_{2,1} & P_{2,2} & P_{2,3} & P_{2,4} \\ P_{3,1} & P_{3,2} & P_{3,3} & P_{3,4} \end{bmatrix}$$

Camera Matrix Decomposition (K and R)

$$P = K[R|t] = K[R\,|{-RC}] = [KR\,|-KRC]$$

- K is upper triangular
- R is orthonormal
- QR decomposition A = QR
  - Q is orthogonal
  - R is upper triangular

Camera Matrix Decomposition (K and R)

$$P = [KR \mid -KRC]$$

$$M = KR$$

$$M^{-1} = R^{-1}K^{-1}$$

- Run QR-decomposition on the inverse of the left 3x3 part of P
- Invert both result matrices to get K and R

Camera Matrix Decomposition (K and R)

- K should have a positive diagonal
- $\det(R) = 1$

$$\text{e.g. } T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$KR = KTT^{-1}R$$

$$\rightarrow K' = KT \qquad\qquad R' = T^{-1}R$$

Camera Matrix Decomposition (K and R)

- K should have a positive diagonal
- $\det(R) = 1$

$$\text{If } \det(R) = -1 \rightarrow R = -R$$

Camera Matrix Decomposition (C)

- The camera center is the point for which

$$\mathbf{PC} = 0$$

- This is the right null vector of P (→ SVD)

Gold Standard Algorithm

- Normalize data
- Run DLT to get initial values
- Compute optimal $\hat{\mathbf{P}}$ by minimizing the sum of squared reprojection errors

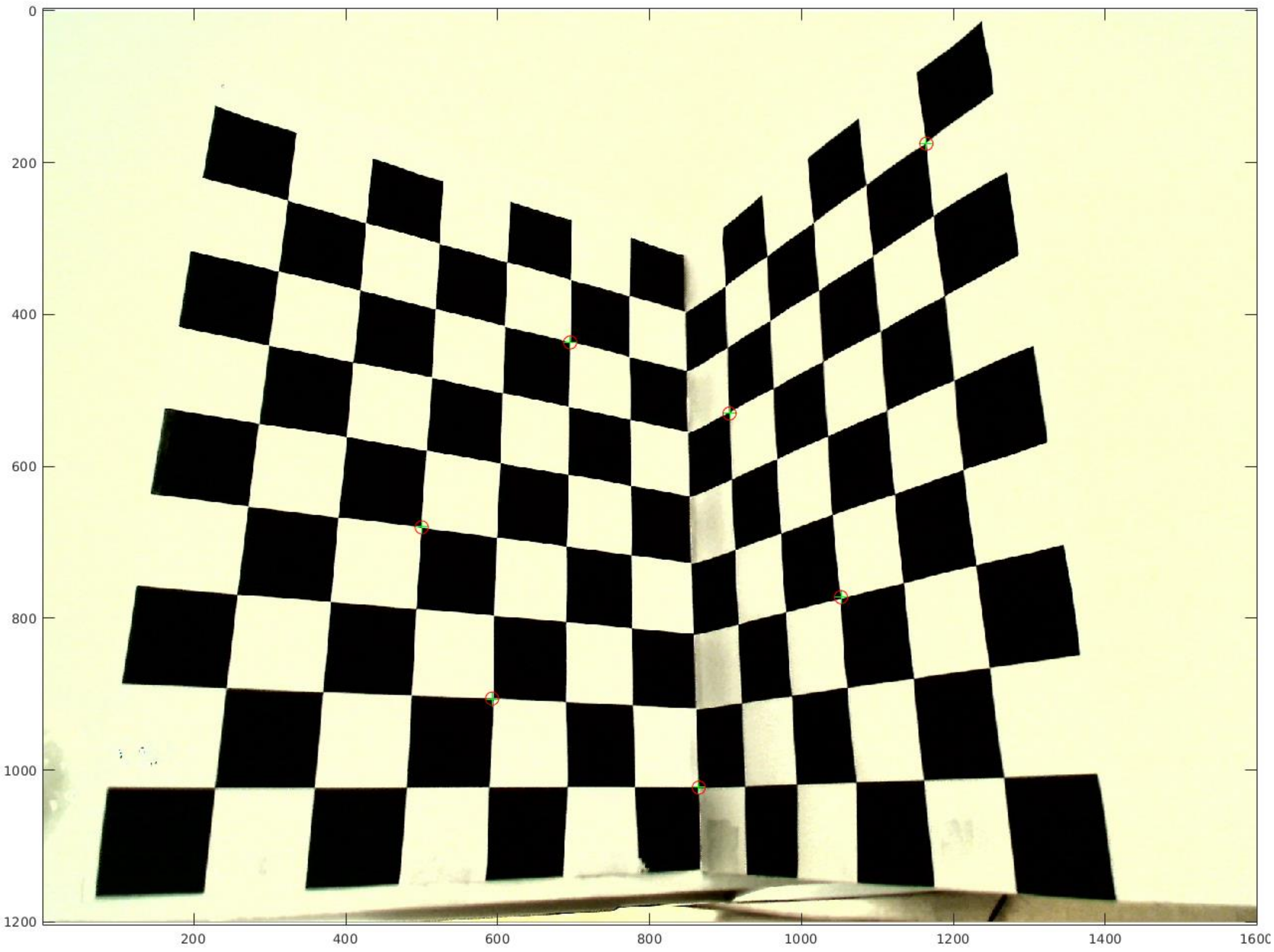
$$\min_{\hat{\mathbf{P}}} \sum_{i=1}^{N} \mathrm{d}(\hat{\mathbf{x}}_i, \hat{\mathbf{P}}\hat{\mathbf{X}}_i)^2$$

- Denormalize $\hat{\mathbf{P}}$

Hand-in

- Source code
- Matlab .mat-file with hand-clicked 3D-2D correspondences
- Image used for calibration (optional)
  - Use the same camera with the same settings for all tasks!
- Visualize hand-clicked points and reprojected 3D points
- Discuss values of intrinsic parameters
- Discuss average reprojection errors of all methods

Hand-in



- Reprojection of the the 3D points