Master in Computer Vision Barcelona

Module: 3D Vision
Lecture 9: Auto-calibration.
Bundle adjustment.

Lecturer: Gloria Haro

Outline

- Auto-calibration
- Bundle adjustment

Auto-calibration

Auto-calibration is the process of **determining the internal camera parameters** directly from multiple uncalibrated images.

There are different methods for autocalibration. We will see the stratified methods which involve two steps:

- Affine reconstruction: it finds the plane at infinity Π_∞
- Metric reconstruction: it finds the image of the absolute conic $\omega = K^{-T}K^{-1}$ (and thus the internal parameters)

Auto-calibration

Problem statement

Assume we already have a projective reconstruction, i.e., we know

- the camera matrices: P_p^i, $i = 1, ..., m$
- the 3D points: $\mathbf{X}_{j,p}$, $j = 1, ..., n$

such that $\mathbf{x_j^i} \equiv P_p^i \mathbf{X}_{j,p}$ $\forall i, j$ (subindex p stands for *projective*)

Auto-calibration

Problem statement

Assume we already have a projective reconstruction, i.e., we know

- the camera matrices: P_p^i, $i = 1, ..., m$
- the 3D points: $\mathbf{X}_{j,p}$, $j = 1, ..., n$

such that $\mathbf{x_j^i} \equiv P_p^i \mathbf{X}_{j,p} \; \forall i, j$ (subindex p stands for *projective*)

There is a **projective ambiguity**:

$$\mathbf{x_j^i} \equiv P_p^i H^{-1} H \mathbf{X}_{j,p} = \hat{P}^i \hat{\mathbf{X}}_j$$

Auto-calibration

Problem statement

Assume we already have a projective reconstruction, i.e., we know

- the camera matrices: P_p^i, $i = 1, ..., m$
- the 3D points: $\mathbf{X}_{j,p}$, $j = 1, ..., n$

such that $\mathbf{x_j^i} \equiv P_p^i \mathbf{X}_{j,p} \; \forall i, j$ (subindex p stands for *projective*)

There is a **projective ambiguity**:

$$\mathbf{x_j^i} \equiv P_p^i H^{-1} H \mathbf{X}_{j,p} = \hat{P}^i \hat{\mathbf{X}}_j$$

Goal: Find a projective mapping $H_{e \leftarrow p}$ that maps the projective recons. to a metric one.

Auto-calibration

Problem statement

Assume we already have a projective reconstruction, i.e., we know

- the camera matrices: P_p^i, $i = 1, ..., m$
- the 3D points: $\mathbf{X}_{j,p}$, $j = 1, ..., n$

such that $\mathbf{x_j^i} \equiv P_p^i \mathbf{X}_{j,p} \; \forall i, j$ (subindex p stands for *projective*)

There is a **projective ambiguity**:

$$\mathbf{x_j^i} \equiv P_p^i H^{-1} H \mathbf{X}_{j,p} = \hat{P}^i \hat{\mathbf{X}}_j$$

Goal: Find a projective mapping $H_{e \leftarrow p}$ that maps the projective recons. to a metric one.

After the mapping, in the Euclidean reference system we have:

- the camera matrices: $P_e^i = P_p^i H_{e \leftarrow p}^{-1}$, $i = 1, ..., m$
- the 3D points: $\mathbf{X}_{j,e} = H_{e \leftarrow p} \mathbf{X}_{j,p}$, $j = 1, ..., n$

Auto-calibration

By choosing the world reference system in the first camera we have:

$$P_e^1 = K[I|0]$$

We also may choose the projective reconstruction so that:

$$P_p^1 = [I|0]$$

Auto-calibration

By choosing the world reference system in the first camera we have:

$$P_e^1 = K[I|0]$$

We also may choose the projective reconstruction so that:

$$P_p^1 = [I|0]$$

Recall that a 4×4 projective matrix can be written as:

$$H_{e \leftarrow p}^{-1} = \begin{pmatrix} A & \tau \\ \mathbf{v}^T & v \end{pmatrix}$$

From $P_e^1 = P_p^1 H_{e \leftarrow p}^{-1}$, we have:

$$K[I|0] = [I|0] \begin{pmatrix} A & \tau \\ \mathbf{v}^T & v \end{pmatrix}$$

then:

$$A = K \text{ and } \tau = \mathbf{0}$$

Auto-calibration

We had:

$$A = K \text{ and } \tau = \mathbf{0}$$

If we fix $v = 1$, we may write:

$$H_{e \leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ \mathbf{v}^T & 1 \end{pmatrix}$$

where K is the calibration matrix of the first camera.

Auto-calibration

We had:

$$A = K \text{ and } \tau = \mathbf{0}$$

If we fix $v = 1$, we may write:

$$H_{e \leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ \mathbf{v}^T & 1 \end{pmatrix}$$

where K is the calibration matrix of the first camera.

Thus, we look for a matrix $H_{e \leftarrow p}$ with this structure so as to obtain an Euclidean (metric) reconstruction from a projective one.

Auto-calibration

We had:

$$A = K \text{ and } \tau = \mathbf{0}$$

If we fix $v = 1$, we may write:

$$H_{e \leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ \mathbf{v}^T & 1 \end{pmatrix}$$

where K is the calibration matrix of the first camera.

Thus, we look for a matrix $H_{e \leftarrow p}$ with this structure so as to obtain an Euclidean (metric) reconstruction from a projective one.

Notice that

$$H_{e \leftarrow p} = \begin{pmatrix} K^{-1} & \mathbf{0} \\ -\mathbf{v}^T K^{-1} & 1 \end{pmatrix}$$

Auto-calibration

$$H_{e \leftarrow p} = \begin{pmatrix} K^{-1} & \mathbf{0} \\ -\mathbf{v}^T K^{-1} & 1 \end{pmatrix}$$

What is v?

Auto-calibration

$$H_{e\leftarrow p} = \begin{pmatrix} K^{-1} & \mathbf{0} \\ -\mathbf{v}^T K^{-1} & 1 \end{pmatrix}$$

What is v?

Observe that the coordinates of the plane at infinity in an Euclidean frame of $\mathbb{P}^3$ are:

$$\Pi_{\infty,e} = (0,0,0,1)^T$$

Remember how does a homography acts on a plane: $\Pi' = H^{-T}\Pi$. Then,

$$\Pi_{\infty,e} = (0,0,0,1)^T = H_{e\leftarrow p}^{-T}\Pi_{\infty,p}$$

Auto-calibration

$$H_{e\leftarrow p} = \begin{pmatrix} K^{-1} & \mathbf{0} \\ -\mathbf{v}^T K^{-1} & 1 \end{pmatrix}$$

What is v?

Observe that the coordinates of the plane at infinity in an Euclidean frame of $\mathbb{P}^3$ are:

$$\Pi_{\infty,e} = (0,0,0,1)^T$$

Remember how does a homography acts on a plane: $\Pi' = H^{-T}\Pi$.

Then,

$$\Pi_{\infty,e} = (0,0,0,1)^T = H_{e\leftarrow p}^{-T}\Pi_{\infty,p}$$

and

$$\Pi_{\infty,p} = H_{e\leftarrow p}^{T}\Pi_{\infty,e} = \begin{pmatrix} K^{-T} & -K^{-T}\mathbf{v} \\ -\mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -K^{-T}\mathbf{v} \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix}$$

Hence $\mathbf{p} = -K^{-T}\mathbf{v} \longrightarrow \mathbf{v} = -K^T\mathbf{p}$

Auto-calibration

Then, we have

$$H_{e\leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ -\mathbf{p}^T K & 1 \end{pmatrix}$$

Auto-calibration

Then, we have

$$H_{e\leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ -\mathbf{p}^T K & 1 \end{pmatrix} = \underbrace{\begin{pmatrix} I & \mathbf{0} \\ -\mathbf{p}^T & 1 \end{pmatrix}}_{H_{a\leftarrow p}^{-1}} \underbrace{\begin{pmatrix} K & \mathbf{0} \\ -\mathbf{0}^T & 1 \end{pmatrix}}_{H_{e\leftarrow a}^{-1}}$$

where $H_{a\leftarrow p}^{-1}$ is a projective transformation, and
$H_{e\leftarrow a}^{-1}$ is an affine transformation.

Auto-calibration

Then, we have

$$H_{e\leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ -\mathbf{p}^T K & 1 \end{pmatrix} = \underbrace{\begin{pmatrix} I & \mathbf{0} \\ -\mathbf{p}^T & 1 \end{pmatrix}}_{H_{a\leftarrow p}^{-1}} \underbrace{\begin{pmatrix} K & \mathbf{0} \\ -\mathbf{0}^T & 1 \end{pmatrix}}_{H_{e\leftarrow a}^{-1}}$$

where $H_{a\leftarrow p}^{-1}$ is a projective transformation, and
$H_{e\leftarrow a}^{-1}$ is an affine transformation.

Notice that, $H_{e\leftarrow p} = H_{e\leftarrow a} H_{a\leftarrow p}$

Auto-calibration

Then, we have

$$H_{e\leftarrow p}^{-1} = \begin{pmatrix} K & \mathbf{0} \\ -\mathbf{p}^T K & 1 \end{pmatrix} = \underbrace{\begin{pmatrix} I & \mathbf{0} \\ -\mathbf{p}^T & 1 \end{pmatrix}}_{H_{a\leftarrow p}^{-1}} \underbrace{\begin{pmatrix} K & \mathbf{0} \\ -\mathbf{0}^T & 1 \end{pmatrix}}_{H_{e\leftarrow a}^{-1}}$$

where $H_{a\leftarrow p}^{-1}$ is a projective transformation, and
$H_{e\leftarrow a}^{-1}$ is an affine transformation.

Notice that, $H_{e\leftarrow p} = H_{e\leftarrow a} H_{a\leftarrow p}$

- $H_{a\leftarrow p}$ permits to upgrade the projective recons. into an affine one
  Affine reconstruction
- $H_{e\leftarrow a}$ permits to upgrade the affine recons. into a metric one
  Metric reconstruction

Auto-calibration – Affine reconstruction

We need to find the homography:

$$H_{a \leftarrow p} = \begin{pmatrix} I & \mathbf{0} \\ \mathbf{p}^T & 1 \end{pmatrix}$$

The essence of the affine reconstruction is then to locate the plane at infinity in the projective reconstruction frame.

Auto-calibration – Affine reconstruction

We need to find the homography:

$$H_{a \leftarrow p} = \begin{pmatrix} I & \mathbf{0} \\ \mathbf{p}^T & 1 \end{pmatrix}$$

The essence of the affine reconstruction is then to locate the plane at infinity in the projective reconstruction frame.

By applying $H_{a \leftarrow p}$ we will map $\mathbf{p}$ to $\Pi_\infty = (0,0,0,1)^T$ and we will recover te paralellism.

Auto-calibration – Affine reconstruction

How do we identify p?
As every plane, the plane at infinity is determined by **three points on it**.

Procedure:
since $\Pi^T \mathbf{X}_i = 0$, $i = 1, 2, 3$, we have:

$$\underbrace{\begin{pmatrix} \mathbf{X}_1^T \\ \mathbf{X}_2^T \\ \mathbf{X}_3^T \end{pmatrix}}_{A\,(3\times 4 \text{ matrix})} \Pi = \mathbf{0}$$

If the three points are in general position (not on the same line), they provide linearly indep. eq. and the matrix they form is rank 3.

$\longrightarrow$ Then Π is obtained uniquely (up to scale) as the 1-dimensional right null space of A.

Auto-calibration – Affine reconstruction

How do we find the three points on the plane at infinity?

Auto-calibration – Affine reconstruction

How do we find the three points on the plane at infinity?
They are 3D vanishing points, coming from three pairs of parallel lines with different directions.

Auto-calibration – Affine reconstruction

How do we find the three points on the plane at infinity?
They are 3D vanishing points, coming from three pairs of parallel lines with different directions.

How do they may be computed from the images?

Auto-calibration – Affine reconstruction

How do we find the three points on the plane at infinity?
They are 3D vanishing points, coming from three pairs of parallel lines with different directions.

How do they may be computed from the images?

a) Pick three vanishing points in two images and find their corresponding 3D points by triangulation.

b) Identify a vanishing point $\mathbf{v}_i$ in one image and one of the lines ℓ'_i (image of one of the parallel lines of the scene leading to $\mathbf{v}'_i$) in the second image. Then, find $\mathbf{V}_i \in \mathbb{P}^3$ by solving:

$$\left.\begin{array}{r} \mathbf{v}_i \times P\mathbf{V}_i = \mathbf{0} \\ \ell_i'^T P'\mathbf{V}_i = 0 \end{array}\right\} \longrightarrow A\mathbf{V}_i = \mathbf{0}$$

$\mathbf{V}_i$ is the right null vector of A.

Auto-calibration

Auto-calibration – Metric reconstruction

The key to metric reconstruction is to find the image of the absolute conic in one of the images $\omega = K^{-T}K^{-1}$

Auto-calibration – Metric reconstruction

The key to metric reconstruction is to find the image of the absolute conic in one of the images $\omega = K^{-T}K^{-1}$

Suppose that the image of the absolute conic is known in some image to be ω, and one has an affine recons. in which the corresponding camera matrix is given by $P = [M|m]$. Then, the affine recons. may be transformed to a metric recons. by applying a 3D transformation of the form:

$$H_{e\leftarrow a} = \begin{pmatrix} A^{-1} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

where A is obtained by Cholesky factorization: $AA^T = (M^T\omega M)^{-1}$

Auto-calibration – Metric reconstruction

The approach relies on identiying ω. There are various ways of doing this.

There are different kinds of constraints on ω:

- Constraints coming from scene orthogonality.
- Constraints coming from known internal parameters.
- Constraints arising from the same cameras (same matrix K) in all images.

Typically, a combination of these constraints is used.

Auto-calibration – Metric reconstruction

Constraints coming from scene orthogonality

If $\mathbf{v}_1$ and $\mathbf{v}_2$ is a pair of vanishing points arising from orthogonal scene lines, then we have a linear constraint on ω:

$$\mathbf{v}_1^T \omega \mathbf{v}_2 = 0$$

Auto-calibration – Metric reconstruction

Constraints coming from known internal parameters

Since

$$\omega = K^{-T}K^{-1} = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{12} & \omega_{22} & \omega_{23} \\ \omega_{13} & \omega_{23} & \omega_{33} \end{pmatrix}$$

knowledge about some restrictions on the internal parameters contained in K may be used to constraint or determine the elements of ω.

Auto-calibration – Metric reconstruction

Constraints coming from known internal parameters

Since

$$\omega = K^{-T}K^{-1} = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{12} & \omega_{22} & \omega_{23} \\ \omega_{13} & \omega_{23} & \omega_{33} \end{pmatrix}$$

knowledge about some restrictions on the internal parameters contained in K may be used to constraint or determine the elements of ω.

In the case where the camera is known to have zero skew, then $\omega_{12} = 0$.

Auto-calibration – Metric reconstruction

Constraints coming from known internal parameters

Since

$$\omega = K^{-T}K^{-1} = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{12} & \omega_{22} & \omega_{23} \\ \omega_{13} & \omega_{23} & \omega_{33} \end{pmatrix}$$

knowledge about some restrictions on the internal parameters contained in K may be used to constraint or determine the elements of ω.

In the case where the camera is known to have zero skew, then $\omega_{12} = 0$.

If the pixels are square, that is, zero skew and $\alpha_x = \alpha_y$, then: $\omega_{11} = \omega_{22}$.

Auto-calibration – Metric reconstruction

Combination of the previous constraints

Five constraints on ω:

$$\mathbf{u}^T \omega \mathbf{v} = 0$$

$$\mathbf{u}^T \omega \mathbf{z} = 0$$

$$\mathbf{v}^T \omega \mathbf{z} = 0$$

$$\omega_{11} = \omega_{22}$$

$$\omega_{12} = 0$$

In matrix form: $A\omega_V = \mathbf{0}$,
where $\omega_V = (\omega_{11}, \omega_{12}, \omega_{13}, \omega_{22}, \omega_{23}, \omega_{33})^T$

$$A = \begin{pmatrix} u_1 v_1 & u_1 v_2 + u_2 v_1 & u_1 v_3 + u_3 v_1 & u_2 v_2 & u_2 v_3 + u_3 v_2 & u_3 v_3 \\ u_1 z_1 & u_1 z_2 + u_2 z_1 & u_1 z_3 + u_3 z_1 & u_2 z_2 & u_2 z_3 + u_3 z_2 & u_3 z_3 \\ v_1 z_1 & v_1 z_2 + v_2 z_1 & v_1 z_3 + v_3 z_1 & v_2 z_2 & v_2 z_3 + v_3 z_2 & v_3 z_3 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

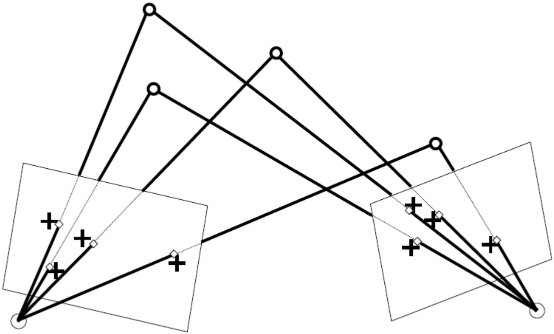
The solution ω_V is the null vector of A.

Outline

- Auto-calibration
- Bundle adjustment

Bundle adjustment

If the image measurements are noisy, then the equations $x_j^i = P^i \mathbf{X}_j$ will not be satisified exactly.



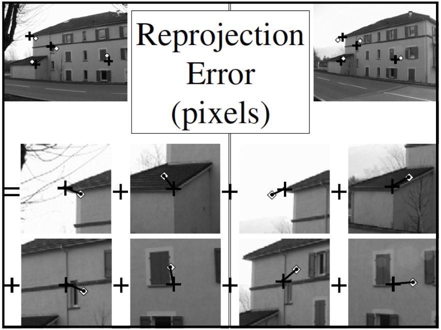
We then seek for matrices $\hat{P}^i$ and 3D points $\hat{\mathbf{X}}_j$ such that $\hat{\mathbf{x}}_j^i = \hat{P}^i \hat{\mathbf{X}}_j$ and $\hat{\mathbf{x}}_j^i$ as much close as possible to $\mathbf{x}_j^i$.

Image source: A. Bartoli

Bundle adjustment

Strategy: Minimize the reprojection error

$$\min_{\hat{P}^i, \hat{\mathbf{X}}_j} \sum_{i,j} d(\hat{P}^i \hat{\mathbf{X}}_j, \mathbf{x}_j^i)^2$$



In the case where the noise has Gaussian distribution, by minimizing this function we are looking for the Maximum Likelihood estimator.

Image source: A. Bartoli

Bundle adjustment

This estimation is known as bundle adjustment since it involves adjusting the bundle of rays between each camera and the set of 3D points.
It is generally used as a final step for any reconstruction algorithm.

PROS:

- It is tolerant to missing data
- It is robust to noise (provides the ML estimate)
- It may be extended to include priors and constraints on camera parameters or point positions.

CONS:

- Non-convex problem, it requires a good initialization
- It can become a large minimization problem because of the number of parameters involved.

Bundle adjustment

Usually the Levenberg-Marquardt method is used to minimize the problem.

There are many variants on how to proceed to reduce the complexity of the probem:

- Do not include all the views or the points, and fill these in later by resectioning or triangulation respectively.
- Partition the data into sets, bundle adjust each set separately and then merge.
- Alternate minimization of P's and $\mathbf{X}$'s.
- Sparse methods.