Computer Vision II: Multiple View Geometry (IN2228)

Chapter 12 Bundle Adjustment

Dr. Haoang Li

06 July 2023  11:00-11:45
# Announcement Before Class

## Updated Lecture Schedule

For updates, slides, and additional materials: [https://cvg.cit.tum.de/teaching/ss2023/cv2](https://cvg.cit.tum.de/teaching/ss2023/cv2)

<table>
<thead>
<tr>
<th>Date</th>
<th>Chapter/Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wed 19.04.23</td>
<td>Chapter 00: Introduction</td>
</tr>
<tr>
<td>Thu 20.04.23</td>
<td>Chapter 01: Mathematical Backgrounds</td>
</tr>
<tr>
<td>Wed 26.04.23</td>
<td>Chapter 02: Motion and Scene Representation (Part 1)</td>
</tr>
<tr>
<td>Thu 27.04.23</td>
<td>Chapter 02: Motion and Scene Representation (Part 2)</td>
</tr>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
<tr>
<td>Thu 18.05.23</td>
<td>No lecture (Public Holiday)</td>
</tr>
<tr>
<td>Wed 24.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Thu 25.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Wed 26.04.23</td>
<td>Chapter 02: Motion and Scene Representation (Part 1)</td>
</tr>
<tr>
<td>Thu 27.04.23</td>
<td>Chapter 02: Motion and Scene Representation (Part 2)</td>
</tr>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
<tr>
<td>Thu 18.05.23</td>
<td>No lecture (Public Holiday)</td>
</tr>
<tr>
<td>Wed 24.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Thu 25.05.23</td>
<td>No lecture (Conference)</td>
</tr>
</tbody>
</table>

### Foundation

<table>
<thead>
<tr>
<th>Date</th>
<th>Chapter/Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
</tbody>
</table>

### Core part

<table>
<thead>
<tr>
<th>Date</th>
<th>Chapter/Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
<tr>
<td>Thu 18.05.23</td>
<td>No lecture (Public Holiday)</td>
</tr>
<tr>
<td>Wed 24.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Thu 25.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
</tbody>
</table>

### Advanced topics and high-level tasks

<table>
<thead>
<tr>
<th>Date</th>
<th>Chapter/Part</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
<tr>
<td>Thu 18.05.23</td>
<td>No lecture (Public Holiday)</td>
</tr>
<tr>
<td>Wed 24.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Thu 25.05.23</td>
<td>No lecture (Conference)</td>
</tr>
<tr>
<td>Wed 03.05.23</td>
<td>Chapter 03: Image Formation (Part 1)</td>
</tr>
<tr>
<td>Thu 04.05.23</td>
<td>Chapter 03: Image Formation (Part 2)</td>
</tr>
<tr>
<td>Wed 10.05.23</td>
<td>Chapter 04: Camera Calibration</td>
</tr>
<tr>
<td>Thu 11.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 1)</td>
</tr>
<tr>
<td>Wed 17.05.23</td>
<td>Chapter 05: Correspondence Estimation (Part 2)</td>
</tr>
</tbody>
</table>

Videos and reading materials about the combination of deep learning and multi-view geometry:

- Wed 05.07.23: Chapter 11: Photometric Error and Direct Method
- Thu 06.07.23: Chapter 12: Bundle Adjustment
- Wed 12.07.23: Chapter 13: Robust Estimation
- Thu 13.07.23: Exam Information and Knowledge Review
- Wed 19.07.23: Chapter 14: SLAM and SFM
- Thu 20.07.23: No Onsite Lecture. Alternative: Online Meeting for Question Answering
Today’s Outline

- Error Metrics
- Definition of Bundle Adjustment
- Basic Knowledge of Non-linear Optimization

- Application of Non-linear Optimization to Bundle Adjustment Based on Lie Algebra (skipped due to limited time)
Error Metrics

Overview

✓ The quality of the estimated camera pose can be measured using different error metrics:
  • Algebraic error
  • Epipolar Line Distance (only for 2D-2D)
  • Reprojection Error

✓ By minimizing any of the above error, we can optimize the camera pose.

✓ The above metrics are not limited to 2D-2D. We can also use them to evaluate 3D-2D case. In our class, let us take 2D-2D for example.
Error Metrics

- Algebraic Error

- We consider 8-point algorithm for illustration. It seeks to minimize the algebraic error:

\[
err = \|QE\|^2 = \sum_{i=1}^{N} (\bar{p}_2^i E \bar{p}_1^i)^2
\]

- From the derivation of the epipolar constraint and the property of dot product, we can observe:

\[
\|\bar{p}_2^T E \bar{p}_1\| = \|\bar{p}_2^T \cdot (Ep_1)\| = \|\bar{p}_2\| \|Ep_1\| \cos(\theta)
\]

  - Associative law
  - Property of dot product

\[
= \|\bar{p}_2\| \|[T_x]R \bar{p}_1\| \cos(\theta)
\]

  - Definition of essential matrix (in right camera frame)
Error Metrics

- Algebraic Error

✓ We can see that this product depends on the angle $\theta$ between $\vec{p}_2$ and the normal to the epipolar plane.

✓ It is nonzero when $\vec{p}_1, \vec{p}_2$, and $T$ are not coplanar.
Error Metrics

- Epipolar Line Distance (only for 2D-2D configuration)

  - Sum of squared epipolar-line-to-point distances:
    \[
    \text{err} = \sum_{i=1}^{N} \left( d(p_1^i, l_1^i) \right)^2 + \left( d(p_2^i, l_2^i) \right)^2
    \]

  - Cheaper than reprojection error (introduced later) because does not require point triangulation

- Epipolar line computed by the left point

- Right point

- Fundamental Matrix \( F = K_2^{-T} E K_1^{-1} \)

- Point lies on a line: \( \text{dot}(p, l) = 0 \)

- Epipolar plane

- Epipolar line computed by the left point
Error Metrics

- Reprojection Error

✓ Sum of the Squared Reprojection Errors

\[
err = \sum_{i=1}^{N} \|p_i^1 - \pi(p_i, K_1, I, 0)\|^2 + \|p_i^2 - \pi(p_i, K_2, R, T)\|^2
\]

✓ More expensive than the previous errors because it requires to first triangulate the 3D points.
Error Metrics

- Reprojection Error

- It is the most popular because more accurate. The reason is that the error is computed directly with respect to the original input data, i.e., the image points. It is point-to-point distance.
- Previous algebraic error is with respect to 3D direction; Epipolar line distance is a point-to-line distance.
- Reprojection error is commonly called “golden standard” in our society. For a systematic analysis, please refer to [1].

Link: https://www.robots.ox.ac.uk/~vgg/hzbook/
Error Metrics

➢ Reprojection Error

✓ We often use reprojection error to perform two tasks:
  • Pose and 3D point optimization
  • Accuracy evaluation

Bundle adjustment for optimization  Calibration evaluation
Error Metrics

- Error Minimization

- Let us consider 8-point method. For **more than 8 points**, error will only be 0 if there is **no noise** in the data (if there is image noise, the linear system becomes overdetermined).

- We aim to find the optimal camera pose to minimize the least-squares error.
Bundle Adjustment

Definition

- We extend two-view reprojection minimization to multi-view case, which is called “bundle adjustment”.
- We typically treat the first camera as the world frame.
- We can reformulate the problem as a “graph optimization problem”. Nodes are parameters to optimize, and edges are constraints.
Bundle Adjustment

Definition

- We jointly optimize camera poses of all the cameras and 3D points:

\[
P_i, C_1, \ldots, C_n = \arg\min_{X^i, C_1, \ldots, C_n} \sum_{k=1}^{n} \sum_{i=1}^{N} \rho\left(p_k^i - \pi(P_i, K_k, C_k)\right)
\]

where \(\rho()\) is the Huber norm for robust estimation (introduced next week)

- We often use non-linear optimization, e.g., Gauss-Newton algorithm to minimize the error. Details will be introduced later.
Bundle Adjustment

- Strategies for acceleration

- A small window size limits the number of parameters for the optimization and thus makes real time bundle adjustment possible.

- It is possible to reduce the computational complexity by just optimizing over the camera parameters and keeping the 3D landmarks fixed, e.g., motion-only BA.
Bundle Adjustment

- Photometric Bundle Adjustment

We can extend the photometric error between 1-by-1 frames to 1-by-N frames.

\[
\arg \min_{\lambda, \theta} \sum_{r=1}^{F} \sum_{x \in I_r} \sum_{f \in \text{obs}(x)} \left\| I_r(x) - I_f(\mathcal{W}\{x; \theta_f, \lambda_r(x)\}) \right\|_2^2
\]
Non-linear Optimization

Problem Formulation

✓ A teaser of curve fitting

Input: A set of observed discrete points (no outliers here)
Step 1: Select a suitable model/function with unknown parameters
Step 2: Estimate the parameters by the least-squares method: We define an objective function, i.e., the sum of squared distances.
Non-linear Optimization

Motivation of Gradient Descent Algorithm

To minimize the function, we can employ first-order optimality condition

\[
\min_x F(x) = \frac{1}{2} \| f(x) \|^2_2
\]

\[
\frac{dF}{dx} = 0
\]

If the derivative is simple, we can directly obtain the global minimum of objective function. However, what if the objective function is more complex?
Non-linear Optimization

- Motivation of Gradient Descent Algorithm

Instead of directly obtaining the global minimum, we iteratively minimize the function. $x_k$ is a temporary value. It is known. $\Delta x_k$ is the adjustment of the above temporary value. It is unknown.

1. Give an initial value $x_0$.
2. For $k$-th iteration, we find an incremental value of $\Delta x_k$, such that the object function $\|f(x_k + \Delta x_k)\|_2^2$ reaches a smaller value.
3. If $\Delta x_k$ is small enough, stop the algorithm.
4. Otherwise, let $x_{k+1} = x_k + \Delta x_k$ and return to step 2.
Non-linear Optimization

- Steepest method

Now consider the $k$-th iteration. Suppose the current solution is at $x_k$ and we want to find the increment $\Delta x_k$. For problem simplification, we use the first-order Taylor expansion to re-write the objective function:

$$F(x_k + \Delta x_k) \approx F(x_k) + J(x_k)^T \Delta x_k$$

Along the negative gradient direction, we can ensure that the function decreases:

$$\Delta x^* = -J(x_k)$$

$\Delta x$ is only a direction. We also manually select another step length parameter (learning rate), say, $\lambda$. The smaller function value is $F(x_k) - J(x_k) \lambda$.
Non-linear Optimization

- Newton’s method

\[
F(x_k + \Delta x_k) \approx F(x_k) + J(x_k)^T \Delta x_k + \frac{1}{2} \Delta x_k^T H(x_k) \Delta x_k
\]

We can also use the second-order Taylor expansion to re-write the objective function:

\[
\Delta x^* = \arg \min \left( F(x) + J(x)^T \Delta x + \frac{1}{2} \Delta x^T H \Delta x \right)
\]

We leverage the first-order optimality condition, i.e., computing the derivative with respect to \( \Delta x \) and setting the result to zero. We thus can obtain

\[
J + H \Delta x = 0 \Rightarrow H \Delta x = -J
\]

Hessian matrix
Non-linear Optimization

- Gauss-Newton Method

✓ Motivation

Steepest method results in the zig-zag descending trajectory

Newton’s method is time consuming due to the computation of Hessian matrix

We need a more effective method: We will introduce a representative method “Gauss-Newton algorithm”.

\[
H_f = \begin{bmatrix}
\frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\
\frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n}
\end{bmatrix}
\]
Non-linear Optimization

- Gauss-Newton Method

Similar to the steepest method, we begin with first-order Taylor expansion

\[
f(x + \Delta x) \approx f(x) + J(x)^T \Delta x
\]

We aim to find the optimal \( \Delta x \) to minimize this function

\[
\Delta x^* = \arg \min_{\Delta x} \frac{1}{2} \left\| f(x) + J(x)^T \Delta x \right\|^2
\]

Let us first expand this function:

\[
\frac{1}{2} \left\| f(x) + J(x)^T \Delta x \right\|^2 = \frac{1}{2} \left( f(x) + J(x)^T \Delta x \right)^T \left( f(x) + J(x)^T \Delta x \right)
\]

Perfect square formula

\[
= \frac{1}{2} \left( \|f(x)\|_2^2 + 2f(x)^T J(x) \Delta x + \Delta x^T J(x) J(x)^T \Delta x \right)
\]
Non-linear Optimization

➢ Gauss-Newton Method

We compute the derivative of the above function with respect to $\Delta x$, and then set the derivative to zero:

\[
\frac{1}{2} \left( \| f(x) \|_2^2 + 2 f(x) J(x)^T \Delta x + \Delta x^T J(x) J(x)^T \Delta x \right)
\]

We transform the above into

\[
J(x) f(x) + J(x) J^T(x) \Delta x = 0
\]

We obtain a linear system to compute $\Delta x$

\[
H(x) \Delta x = g
\]

An approximation to Hessian matrix
Non-linear Optimization

- Application to Bundle Adjustment (A Teaser)

  ✓ General objective function simplification by Gauss-Newton

  \[ e(x + \Delta x) \approx e(x) + J\Delta x. \]

  ✓ We have to compute derivative w.r.t. SO3/SE3. It evolves addition and subtraction operation.

  ✓ Intuitively, R1 is in SO3 and R2 is in SO3, but we cannot guarantee that R1 + R2 is in SO3.

  ✓ To solve this problem, we first map Lie Group to Lie Algebra, and compute the derivative by Lie Algebra. **Due to limited time, we will skip this content this year.**
Summary

- Error Metrics
- Bundle Adjustment
- Non-linear Optimization
Thank you for your listening!
If you have any questions, please come to me :-)