In this document, we highlight important knowledge in Chapters 11—13.
This will be highly relevant to the final exam.

Chapter 11 Photometric Error and Direct SLAM

Pages 09, 10, 11, 12, 13, 14, 15, 16, 17, 18, 20, 22, 23, 31, 32, 33, 34, 35

Remark: For pages 33—35, students are required to know the main factors considered in photometric calibration. Students should focus on the concepts and can neglect details.

Chapter 12 Bundle Adjustment

Pages 03, 04, 05, 06, 07, 08, 10, 11, 12, 15, 16, 17, 18, 19, 20, 21, 22

Remark: For pages 17, 18, 20, 21, students are required to know the main idea of each algorithm and understand the differences between various algorithms. In the exam, specific derivations will NOT be asked.

Chapter 13 Robust Estimation

Pages 02, 03, 05, 06, 07, 09, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 26, 28, 31, 32

Remark: For pages 31—32, students are only required to understand the role and significance of Huber loss. No details about Huber loss will be asked in the exam.

Chapter 14 SfM and SLAM

Pages 03, 05, 09, 10, 12, 13, 14, 15, 16, 18, 20, 26, 28, 29

Remark: For pages 16 and 18, students are not required to know the details of SLAM methods.

11 _ Photomeric Error and Direct SLAM

P09 Motivation : two-step → one-step ⇒ Relative Pose (RT)

Feature-based : track features → RT (Noise affect)

Direct-based : based on brightness consistancy

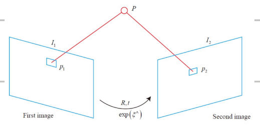
⇒ track features and get RT

P10 Photonetric Error :

$$P_1 = \begin{bmatrix} \bar{u} \\ \bar{v} \\ 1 \end{bmatrix}_1 = \frac{1}{Z_1} P \qquad P_2 = \begin{bmatrix} \bar{u} \\ \bar{v} \\ 1 \end{bmatrix}_2 = K(RP+t)$$

unknown

known



① Given Z_1, R, t of P_1, ⇒ predict P_2

② Assume brightness consistancy assumption

③ Find optimal Z_1, R, t, ⇒ minimize the difference between brightness

④ Get optimal pose and correspondence

$$e = I_1(p_1) - I_2(p_2)$$

$$p_2 = K(RP+t)$$

$$\min_T J(T) = \sum_{i=1}^{N} e_i^T e_i \quad , \quad e_i = I_1(p_{1,i}) - I_2(p_{2,i})$$

⇒ Directly minimize photometric error

$$p^i, R, T = \arg\min_{p^i, R, T} \sum_{i=1}^{N} \left(I_{k-1}(p^i_{k-1}) - I_k\left(\pi(p^i, k, R, T)\right)\right)$$

unknown

\+ All image pixels can be used / reduce computational cost

\- Senstive to frame to frame motion

P14 Depth : Based on depth, back project any pixel in 3D and project in next image

$$\boxed{\lambda} K^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} x_c \\ Y_c \\ Z_c \end{bmatrix}$$

depth

To get depth : RGB-D / Steno Camera / unknown, optimize

Infinite depth problem : $Z \to \infty$ ⇒ numerical stability

↳ $\rho = \frac{1}{\|p - C_0\|}$ $\quad p - C_0 \to \infty$, $\rho \to 0$

P16 Pixel densities: ① Track all pixels = Dense direct method

⇒ Actually, can't ⇒ computational cost

Some points with non-obvious gradient will not be tracked

⇒ but missing points will make 3D Reconstruction harder

② Track pixels with significant gradients = Semi-dense direct method

Only use pixel with high gradient, like edge

Reconstruct Semi-dense structure

③ Track sparse key points = sparse direct method

Don't need to calculate descriptors, just use pixels

Fastest, but sparse reconstruction

P20 Baseline (Relative Pose): Direct SLAM is not suitable for large baseline (motion)

→ Initial pose may be unreliable ⇒ local minimum

→ Photometric consistency assumption is not satisfied

→ Photometric error may be very large

P22

Feature-Based	Direct
can only use & reconstruct corners	can use & reconstruct whole image
faster	slower (but good for parallelism)
flexible: outliers can be removed retroactively.	inflexible: difficult to remove outliers retroactively.
robust to inconsistencies in the model/system (rolling shutter).	not robust to inconsistencies in the model/system (rolling shutter).
decisions (KP detection) based on less complete information.	decision (ordinary point) based on more complete information.
no need for good initialization.	needs good initialization.

Key point

P23 Direct SLAM Method: LSD-SLAM / DSO / VSO

P31 Photometric Calibration: Brightness consistancy assumption may affected by exposure time, vignetting and other factors

→ Reduce various effect to meet brightness consistency assumption

① Response function

Light energy = irradiance

Use irradience consistency between two images

Response function : energy $\xrightarrow{\text{map}}$ digital signal

② Exposure time

long exposure time ⇒ brighter image

Consistency of irradiance ⇒ calibrate exposure time

③ Vignetting

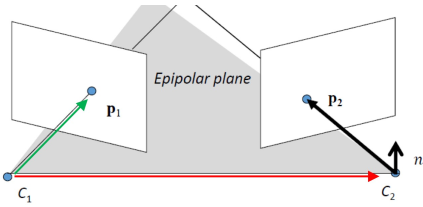
Reduction of an image's brightness toward the periphery compare to image center

corner of image

Manufacturing problem

12 – Bundle Adjustment

P03 Error Metrics : ① Algebraic Error



Essential Matrix

8-point-method : $err = \|QE\|^2 = \sum_{i=1}^{N} (\bar{p}_2^{i\,T} E \bar{p}_1^i)^2$

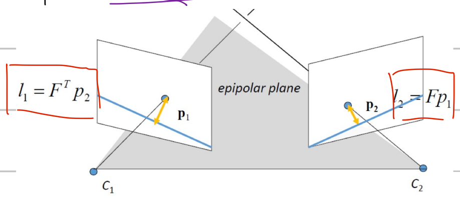
$\|\bar{p}_2^T E \bar{p}_1\| = \|\bar{p}_2^T \cdot (E\bar{p}_1)\| = \|\bar{p}_2\| \, \|E\bar{p}_1\| \cdot \cos\theta = \|\bar{p}_2\| \cdot \boxed{\|[t]_\times R \bar{p}_1\|} \cos\theta$

$= n$ (Nominal of epipolar plane in right camera frame)

⇒ It is nonzero, if p_1, p_2, T are not co-planar ($\theta \neq 90°$)

② Epipolar line distance (2D-2D)

Epipolar-line-to-point distance : $err = \sum_{i=1}^{N} (d(p_1^i, l_1^i))^2 + (d(p_2^i, l_2^i))^2$



Left point

$$\begin{bmatrix} u_2^i \\ v_2^i \\ 1 \end{bmatrix}^T \boxed{F} \begin{bmatrix} u_1^i \\ v_1^i \\ 1 \end{bmatrix} = 0$$

Fundamental Matrix $F = K_2^{-T} E K_1^{-1}$

Point lies on a line: dot(p, l)=0

Epipolar line computed by the right point

05/

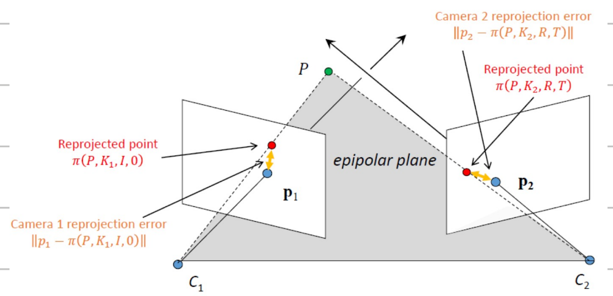
正交 orthogonal

③ Reprojection Error $err = \sum_{i=1}^{N} \|p_1^i - \pi(p^i, K_1, I, 0)\|^2 + \|p_2^i - \pi(p^i, K_2, R, T)\|^2$

Need triangulate the 3D P ⇔ more expensive

⇒ But more accurate ⇔ point-to-point distance

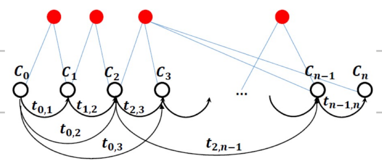
⇒ For Pose and 3D Point optimization / Accuracy evaluation

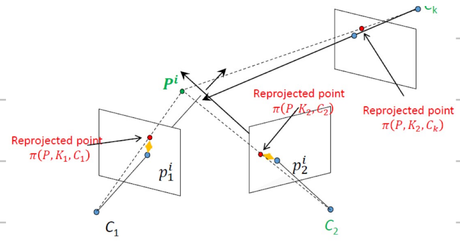


P10 Bundle Adjustment : Two-view reprojection minimization ⇒ Multi-view (Bundle Adjustment)

First camera = World frame

⇒ Graph optimization problem (Node: parameters / Edge: constraints)





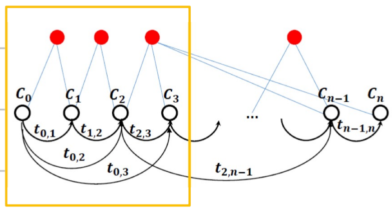
cameras ↘ ↙ points

$$P^1, P^2, \dots, P^n, C_1, C_2, \dots, C_n = \arg\min_{P^1, P^2, \dots P^n, C_1, C_2, \dots, C_n} \sum_{k=1}^{n} \sum_{i=1}^{N} \underbrace{\rho}_{\text{Huber norm}}\left(P_k^i - \pi(P^i, K_k, C_k)\right)$$

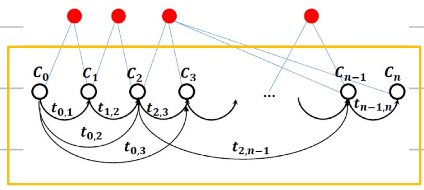
⇑ often use non-linear optimization, e.g. Gauss-Newton

P12 Acceleration : ① Small window limits the parameter

② Fix 3D landmarks and only update camera parameter



A small window



Motion-only BA

P15 Non-linear Optimization : ① First-order optimality condition :

$$\min_x F(x) = \frac{1}{2}\|f(x)\|_2^2 \qquad \frac{dF}{dx} = 0$$

② Gradient Descent for complex problem

iteratively minimize the function

x_k : temporary value / Δx : adjustment for x ⇐ unknown

$$x_{k+1} = x_k + \alpha \cdot \Delta x_k$$

P17 Method : ① Steepest method : First-order Taylor expansion

$$F(x_k + \Delta x_k) = F(x_k) + J(x_k)^T \Delta x_k = F(x_k) - \underbrace{\lambda}_{\text{step}} J(x_k)$$

$\Delta x^* = -J(x_k)$ ⇐ only a direction to show the gradient is along minus

⇒ Zig-zag descending trajectory

② Newton's method

Second-order Taylor

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

$$J + H \Delta x = 0 \Rightarrow H \Delta x = -J$$

⇒ Time consuming due to Hessian matrix

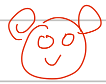
③ Gauss-Newton Method

First-order Taylor

use Jacobian to approximate the Hessian

$$\underbrace{J(x) J^T(x)}_{H(x)} \Delta x = \underbrace{-J(x) f(x)}_{g(x)}$$

$$H \Delta x = g$$



P22 Application to Bundle Adjustment:

General objective function by Gauss-Newton:

$$e(x + \Delta x) \approx e(x) + J \Delta(x)$$

need SO3, SE3 / map Lie Group to Lie Algebra

13 _ Robust Estimation

P02 Problem of outliers : Correspondence ⇐ Feature descriptors or pixel
↑ contaminated by outliers

Reasons : Reptitive patterns / Geometric and photometric changes /
Large image noise / Occlusions / Moving objects / Blur

Goal : Robust Estimation is essential for reliable and accurate localization
Remove outliers and estimate model parameters

Method : Expection - maximization / RANSAC / Robust kernel (M)
⇒ Estimate the parameters of model

P07 EM : Each iteration performs E-step and M-step
E-step : assign weight / outlier ratio to data
M-step : compute model parameters based on weighted fitting

① Estimate line parameter based on all data points , $\min \sum r_i^2$ ⇐ r_i : point - to - line
Result affected by outliers

E ② Base on the estimated model line , calculate the residual error r_i
Base on the residual error , assign weights : $w_i = e^{-r_i}$
$w_i \in [0, 1]$ / r larger , e small , weight small

M ③ Using weighted loss : $\min \sum w_i r_i^2$
update model parameters

④ Repeat EM until convergence
⑤ Treat data with weight > threshold (0.8) as inliers

problem : highly sensitive to initialization
$\sum r_i^2$ is strongly affected by outliers

P14 RANSAC : Random sample consensus

⇒ Not sensitive to the initial solution / not-deterministic ⇐ different results

pipeline : simultaneously remove outliers and estimate 2D line parameters

① Sample "minimal case" data at $\boxed{\text{random}}$ = don't care outliers or inliers

② Calculate mode parameters based on sampled points

③ Calculate residual error w.r.t. the estimated model } point - to - line / point - to - epipolar

④ select data support current hypothesis ("inlier points" near the model)

↑ inlier set ↑ not must be true inlier

If the mode is computed by ture inliers ⇒ should have high cardinality

⑤ Repeat from 1→4 , k times

⑥ Select the inlier set with the highest cardinality

↑ still can be improved due to noise

⑦ Calculate the final model parameters using optimal inlier set

Minimal and Quasi-minimal : Minimal case Sample 5 points ⇔ 5-point-method

Quasi-minimal case Sample 8 points ⇔ 8-point-method

Minimal case is besser : should guarantee at one valid sampling

↳ true inlier set

P26

① If not all correspondences are inliers ⇒ use minimal case and RANSAC

↳ select bast model based on K iterations

② If we know all correspondences are inlier ⇒ use all of them , generate over-determined linear system

↳ select least square solution

P28 EM vs RANSAC : RANSAC is not deterministic / Result based on the random points

RANSAC is not sensitive to initial condition

P31 Robust kernel (M) : Photometric Loss

3D point back-projected by unknown depth

$$P^i, R, T = \arg\min_{P^i, R, T} \sum_{i=1}^{N} \rho\left(I_{k-1}(p^i_{k-1}) - I_k\left(\pi(P^i, K, R, T)\right)\right)$$

Known

Unknown pose

M-estimator functions (e.g., L_2, L_1, Huber)

M-Estimator - $\rho(\xi)$

affected by outlier

M-estimator such as L_2, L_1, Huber ..

robust function

Goal : minimize the objective function

L_2 is sensitive to outliers

↳ use Huber loss ⟺ combination of L_1 and L_2 loss

14 _ SFM and SLAM

P03 Structure from Motion : Input : image with feature correspondence (2D - 2D)

Output : Structure : 3D positions of features

Motion : camera parameters RT , (K is optional)

↓

Two type : Hierarchical SFM : Input : a set of disordered image (different camera / K)

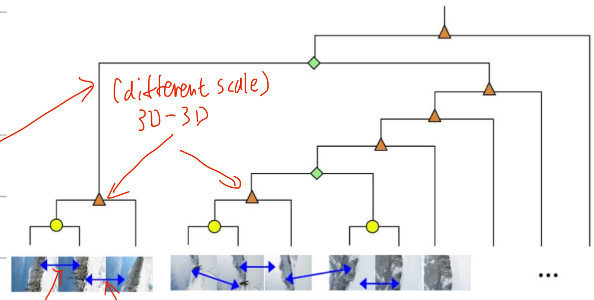
Estimate camera pose and 3D structure in bottom - up

Sequential SFM : Input : a sequential image (same camera / K)

Equivalent to visual odometry (VO)

Incrementally estimate the camera pose and 3D structure

P04 Hierarchial SFM :



7 DoF
sim (3)
For scale

(different scale)
3D - 3D

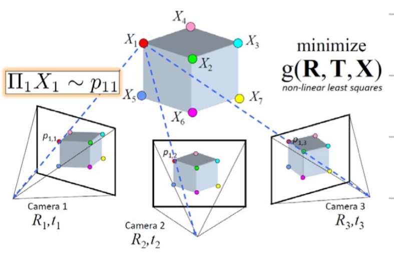
Generate local model
2D - 2D (2D - 3D)

Merge different mode
3D - 3D

2D - 2D
⇒ traingulate 3D P ⇔ 2D ⇒ absolute pose ⇒ 3D point

⇒ Re-projection minimization

Jointly optimize camera pose and 3D points



Number of 3D points

Number of cameras

3D → 2D

$$g(\mathbf{X},\mathbf{R},\mathbf{T}) = \sum_{i=1}^{m}\sum_{j=1}^{n} \underbrace{\boxed{w_{ij}}}_{\substack{\textit{indicator variable:}\\ \text{is point } i \text{ visible in image } j\text{ ?}}} \cdot \left\| \underbrace{\mathbf{P}(\mathbf{x}_i,\mathbf{R}_j,\mathbf{t}_j)}_{\substack{\textit{predicted}\\ \text{image location}}} - \underbrace{\begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix}}_{\substack{\textit{observed}\\ \text{image location}}} \right\|^2$$

Joint estimation using non-linear least-squares optimization

P12 Sequential SFM : Equivalent to visual odometry

① Initialize the structure and motion from 2 views ⇒ RT and $P \in \mathbb{R}^3$

② Absolute pose estimation from 3D-2D for new frame

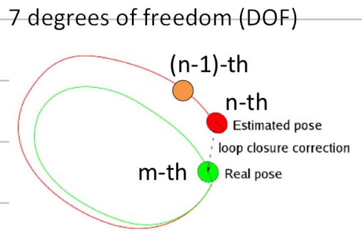
③ Triangulation to increment 3D map

④ Refind structure and motion through bundle adjustment

P14 SLAM : Combination of visual odometry and loop closure

↳ affected by noise ↳ global consistency

Loop closure : Loop detection and correction

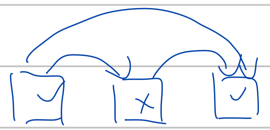


① use m-th pose and Sim(3) to update the n-th path

② use relative pose update n-th pose

m, n 都是预测的点，但是在 m, n 是不同时间同一场景

Method : ① PTAM Parallel Tracking and Mapping



Monocular only

Featured-based (Fast, re-projection error, jointly optimize)

key frame

② ORB-SLAM : advanced PTAM

Monocular and Stereo

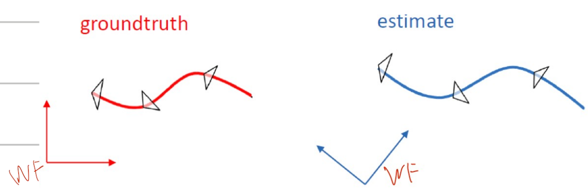
Feature-based (Fast, re-projection error, jointly optimize)

↳ Oriented FAST and Rotated BRIEF

P26 Evaluation Metrics : Compare the estimated trajectory with ground truth trajectory

↳ GPS or motion tracking system

Challenge : Different coordinate systems / Different scales



(ATE)

P28 Absolute trajectory error : ① Align the estimated trajectory to the ground truth from Start to end using Similarity Transformation (R, T, s) by minimizing the sum of square position errors

$$R,T,s = \underset{R,T,s}{\operatorname{argmin}} \sum_{k=0}^{n} \|\hat{C}_k - sRC_k - T\|^2$$

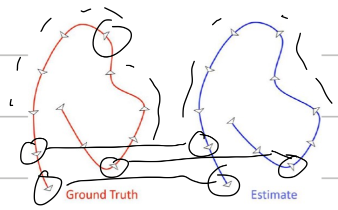
groundtruth positions

estimated positions

7DoF

Parameters of the similarity transformation that we want to find

This can be solved based on **Horn's method** or **Umeyama's method** (we mentioned them in the Chapter of 3D-3D geometry)



3D-3D point correspondences are obtained by **timestamp alignment**

② Compute the root mean square error (RMSE) after alignment

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n} \|\hat{C}_k - sRC_k - T\|^2}{n}}$$

← step

3D-3D point correspondences are obtained by **timestamp alignment**

Pros and cons : + single number metric

+ capture global error

- doesn't encode relative error