



视觉SLAM：从理论到实践

第四次课 相机模型 非线性优化



主讲人 高翔

清华大学 自动控制与工程 博士
慕尼黑工业大学计算机视觉组 博士后
Email: gao.xiang.thu@gmail.com

2017年冬



第四讲 相机模型 非线性优化



1. 针孔相机模型与图像
2. 实践：OpenCV/RGBD图像拼接
3. 批量状态估计问题
4. 非线性最小二乘法
5. 实践：Ceres和g2o

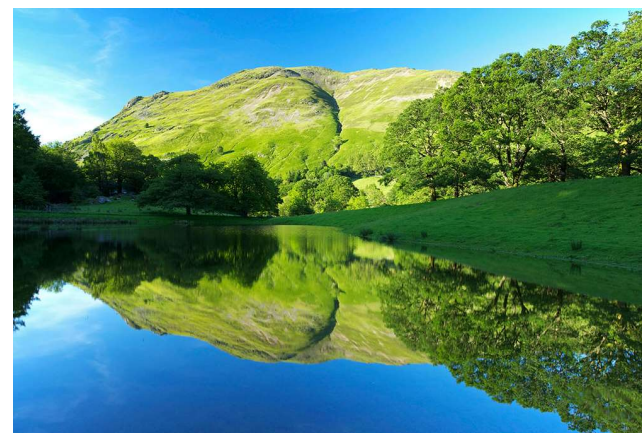
往期内容回顾

1. SLAM的运动与观测模型
2. 观测模型具体形式？

1. 针孔相机模型

1. 针孔相机模型

- 现实生活中存在大量的照片
 - 照片记录了真实世界在成像平面上的投影
 - 这个过程丢弃了“距离”维度上的信息
 - 普通相机可以用针孔模型很好地近似



1. 针孔相机模型

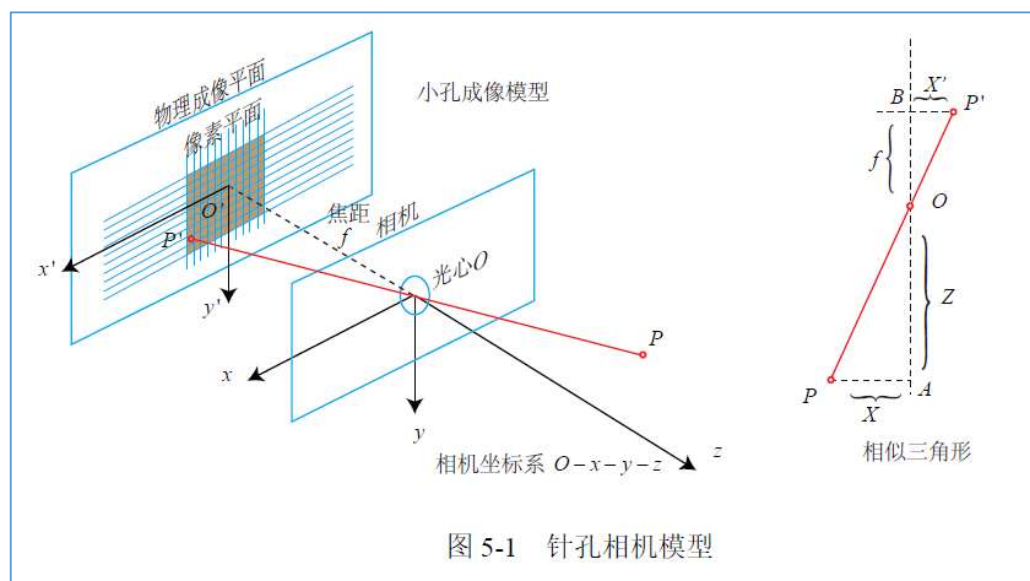
- 小孔成像模型

原始形式 $\frac{Z}{f} = -\frac{X}{X'} = -\frac{Y}{Y'}$

翻转在前面 $\frac{Z}{f} = \frac{X}{X'} = \frac{Y}{Y'}$

整理之：

$$\begin{aligned} X' &= f \frac{X}{Z} \\ Y' &= f \frac{Y}{Z} \end{aligned}$$



1. 针孔相机模型

- 小孔成像模型
- 成像平面到像素坐标

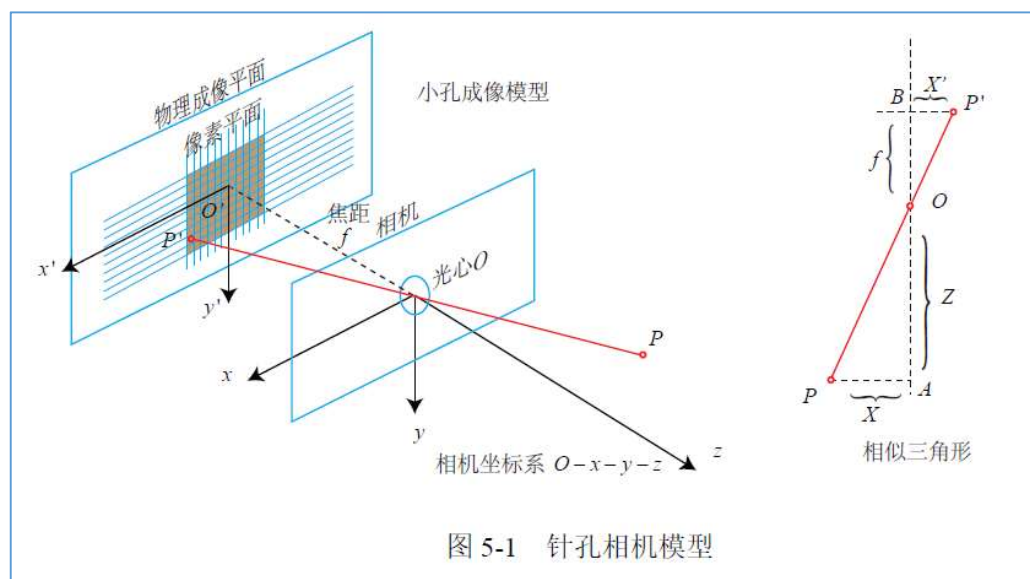
$$\begin{cases} u = \alpha X' + c_x \\ v = \beta Y' + c_y \end{cases}$$

代入

$$\begin{cases} X' = f \frac{X}{Z} \\ Y' = f \frac{Y}{Z} \end{cases}$$

得

$$\begin{cases} u = f_x \frac{X}{Z} + c_x \\ v = f_y \frac{Y}{Z} + c_y \end{cases}$$



1. 针孔相机模型

展开形式

$$\begin{cases} u = f_x \frac{X}{Z} + c_x \\ v = f_y \frac{Y}{Z} + c_y \end{cases}$$

传统习惯

矩阵形式

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \triangleq \frac{1}{Z} KP. \quad Z \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \triangleq KP.$$

左侧是齐次坐标

右侧是非齐次坐标

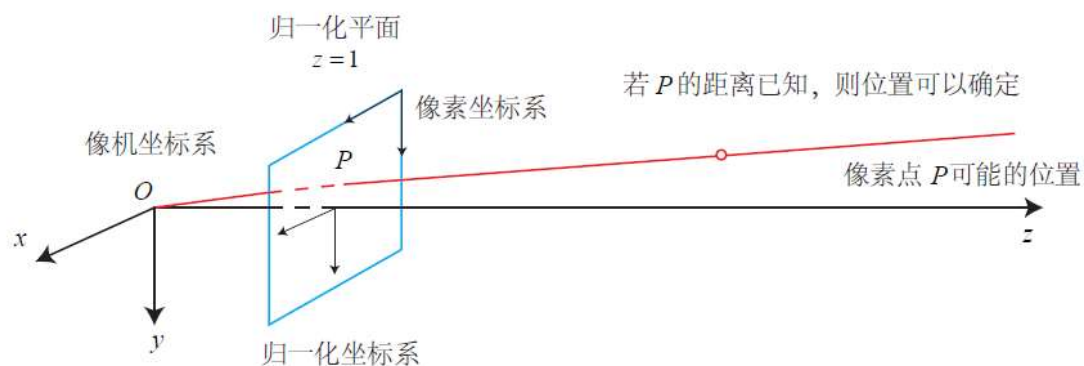
中间矩阵称为内参数

内参通常在相机生产之后就已固定

1. 针孔相机模型

$$Z \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \triangleq KP.$$

同一直线上的投影点仍是同一个



1. 针孔相机模型

- 除内参外，相机坐标系与世界坐标系还相差一个变换：

$$ZP_{uv} = Z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K(RP_w + t) = KTP_w.$$

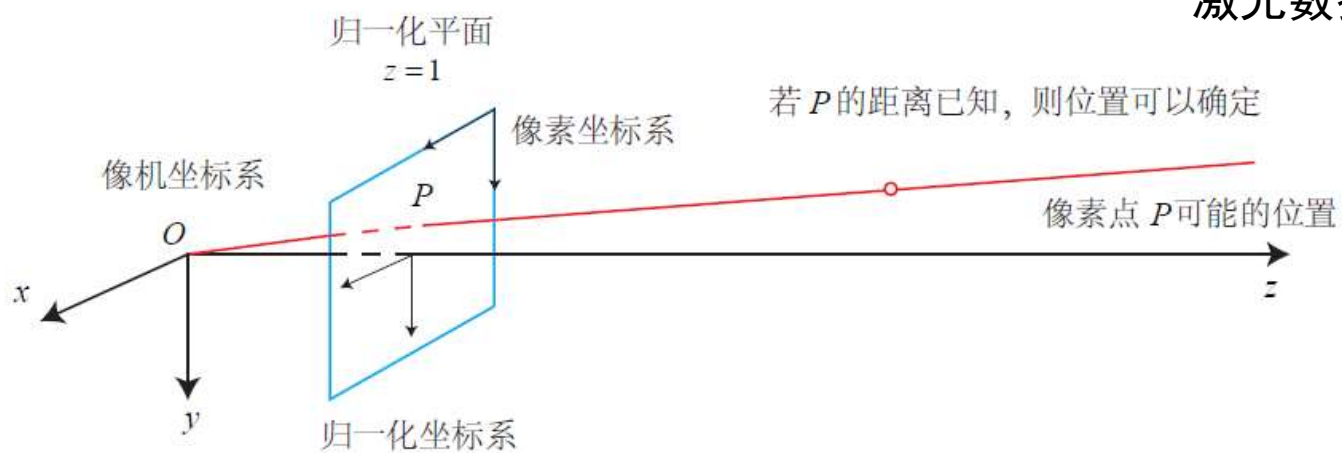
先把P从世界坐标变到
相机坐标系下

- 这里 R, t 或 T 称为外参
 - 注：右侧式子隐含了一次非齐次到齐次的变换（见书）
- 外参是SLAM估计的目标

1. 针孔相机模型

- 投影顺序：世界——相机——归一化平面——像素

激光数据的观测模型更加简单



1. 针孔相机模型

- 畸变
 - 针孔前的镜头会引入畸变



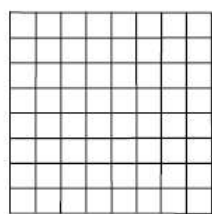
广角镜头畸变



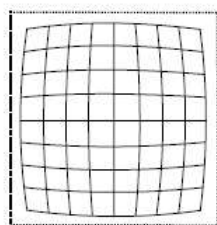
鱼眼镜头畸变

1. 针孔相机模型

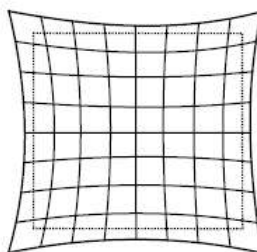
- 主要的畸变类型：径向畸变和切向畸变



正常图像



桶形失真



枕形失真

图 5-3 径向畸变的两种类型。

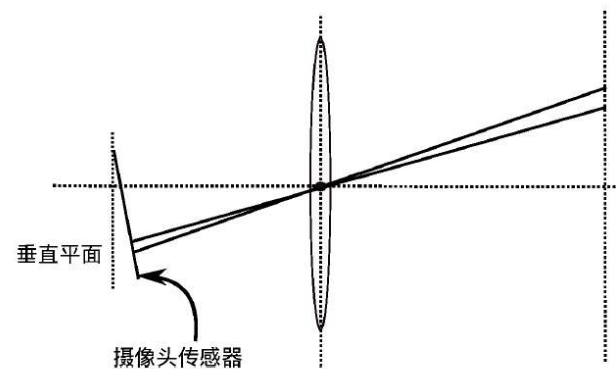


图 5-4 切向畸变来源示意图。

1. 针孔相机模型

- 数学模型
 - 畸变可以用归一化坐标的变换来描述

$$\begin{aligned}x_{distorted} &= x(1 + k_1r^2 + k_2r^4 + k_3r^6) & x_{distorted} &= x + 2p_1xy + p_2(r^2 + 2x^2) \\ y_{distorted} &= y(1 + k_1r^2 + k_2r^4 + k_3r^6) & y_{distorted} &= y + p_1(r^2 + 2y^2) + 2p_2xy\end{aligned}$$

径向畸变：多项式描述

切向畸变：多项式描述

放在一起：
实际当中可灵活保留各项系数

$$\begin{cases} x_{distorted} = x(1 + k_1r^2 + k_2r^4 + k_3r^6) + 2p_1xy + p_2(r^2 + 2x^2) \\ y_{distorted} = y(1 + k_1r^2 + k_2r^4 + k_3r^6) + p_1(r^2 + 2y^2) + 2p_2xy \end{cases}$$

1. 针孔相机模型

- 小结

1. 首先，世界坐标系下有一个固定的点 P ，世界坐标为 P_w ；
2. 由于相机在运动，它的运动由 R, t 或变换矩阵 $T \in SE(3)$ 描述。 P 的相机坐标为：
 $\tilde{P}_c = RP_w + t$ 。
3. 这时的 \tilde{P}_c 仍有 X, Y, Z 三个量，把它们投影到归一化平面 $Z = 1$ 上，得到 P 的归一化相机坐标： $P_c = [X/Z, Y/Z, 1]^T$ ^①。
4. 最后， P 的归一化坐标经过内参后，对应到它的像素坐标： $P_{uv} = KP_c$ 。

1. 针孔相机模型

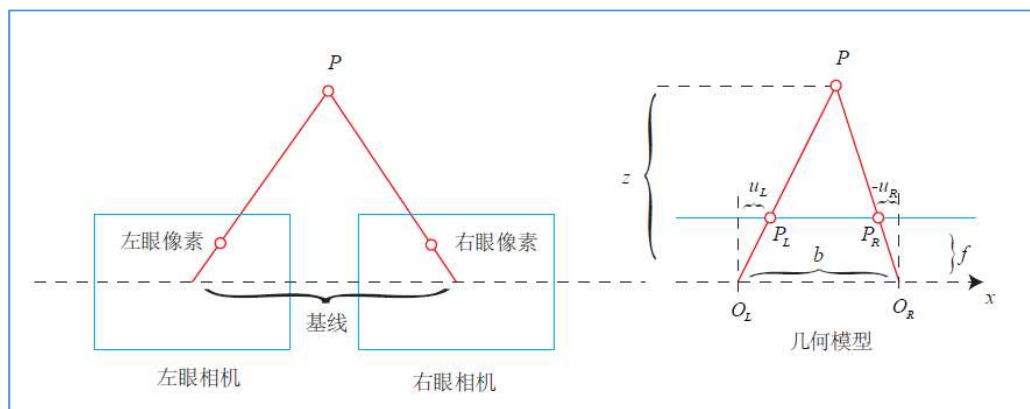
- 双目模型

- 左右相机中心距离称为基线
- 左右像素的几何关系：

$$\frac{z - f}{z} = \frac{b - u_L + u_R}{b}.$$

- 整理得

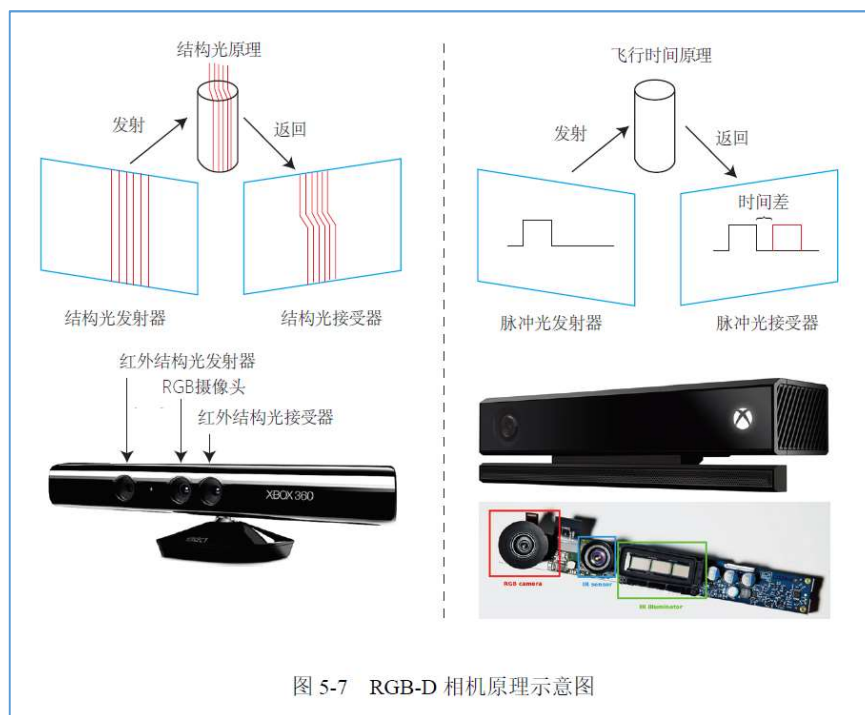
$$z = \frac{fb}{d}, \quad d = u_L - u_R.$$



d称为视差（disparity），描述同一个点在左右目上成像的距离
d最小为1个像素，因此双目能测量的z有最大值：fb
虽然距离公式简单，但d不容易计算

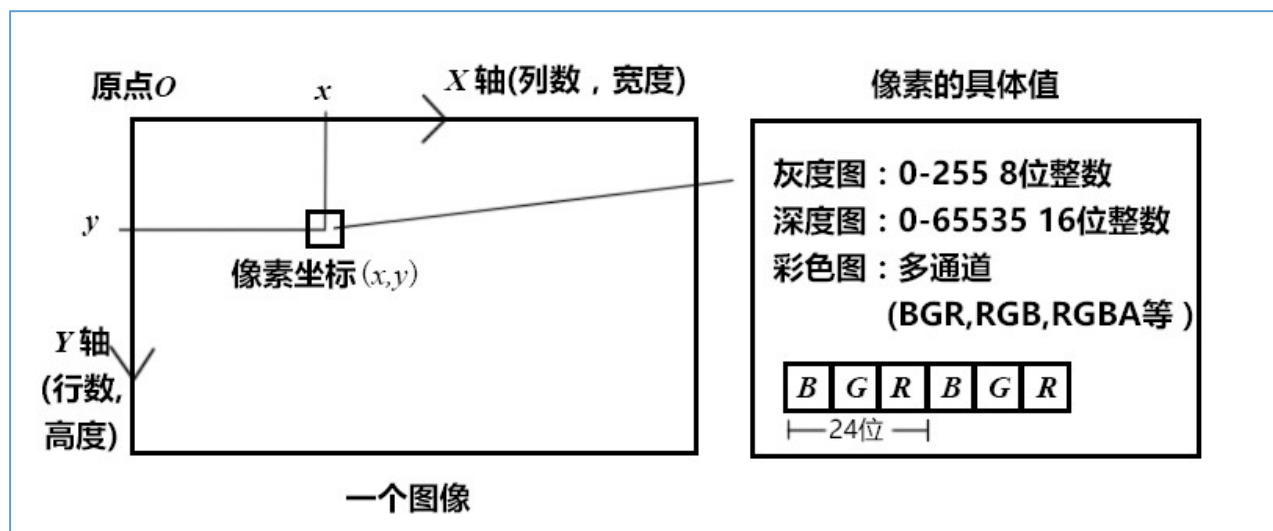
1. 针孔相机模型

- RGB-D相机：物理手段测量深度
 - ToF或结构光两种主要原理
 - 通常能得到与RGB图对应的深度图



1. 针孔相机模型

- 相机成像后，生成了图像
- 图像在计算机中以矩阵形式存储（二维数组）
 - 需要对感光度量化成数值，例如0~255之间的整数（彩色图像还有通道）



2. 实践：OpenCV/图像拼接

3. 批量状态估计问题

3. 批量状态估计问题

现在我们已经探讨了观测模型

x 为旋转+平移
 h 为相机观测模型

$$\begin{cases} x_k = f(x_{k-1}, u_k, w_k) \\ z_{k,j} = h(y_j, x_k, v_{k,j}) \end{cases} .$$

问题：在给定模型和具体观测数据时，
如何估计状态变量？

3. 批量状态估计问题

- 批量式 (batch)
 - 一次给定所有的数据, 估计所有的变量
 - 状态变量:

$$\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_N, \mathbf{y}_1, \dots, \mathbf{y}_M\}.$$

- 求解: $P(\mathbf{x}|\mathbf{z}, \mathbf{u})$.
- 先不考虑运动方程, 仅看观测方程 (类似于SfM)
- 贝叶斯法则:

$$P(\mathbf{x}|\mathbf{z}) = \frac{P(\mathbf{z}|\mathbf{x}) P(\mathbf{x})}{P(\mathbf{z})} \propto P(\mathbf{z}|\mathbf{x}) P(\mathbf{x}).$$

$$\begin{cases} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k) \\ \mathbf{z}_{k,j} = h(\mathbf{y}_j, \mathbf{x}_k, \mathbf{v}_{k,j}) \end{cases}.$$

3. 批量状态估计问题

- $P(x|z)$ 条件分布很难求解，但可以求：

$$P(x|z) = \frac{P(z|x)P(x)}{P(z)} \propto P(z|x)P(x).$$

- a) 最大后验估计 (Maximize a Posterior, MAP)

$$x^*_{MAP} = \arg \max P(x|z) = \arg \max P(z|x)P(x).$$

- b) 最大似然估计 (Maximize Likelihood Estimation, MLE)

$$x^*_{MLE} = \arg \max P(z|x).$$

“在哪种状态下，最容易产生当前的观测”

3. 批量状态估计问题

- 从最大似然到最小二乘

- 例子：某次观测

$$z_{k,j} = h(y_j, x_k) + v_{k,j},$$

- 由于噪声是高斯的：

$$v_k \sim N(0, Q_{k,j})$$

- 于是

$$P(z_{j,k}|x_k, y_j) = N(h(y_j, x_k), Q_{k,j}).$$

- 现在要求x,y的最大似然估计，怎么求？

3. 批量状态估计问题

- 一般的高斯分布：

$$P(x) = \frac{1}{\sqrt{(2\pi)^N \det(\Sigma)}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu)\right).$$

- 负对数形式 $-\ln(P(x)) = \frac{1}{2} \ln\left((2\pi)^N \det(\Sigma)\right) + \frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu).$

马氏距离

最小化x时，只和它有关

- 原问题的最大化，相当于负对数最小化

- 因此，关于原问题的最大似然：

- 相当于最小化： $P(z_{j,k}|x_k, y_j) = N(h(y_j, x_k), Q_{k,j}).$

- 所有量加在一起： $x^* = \arg \min \left((z_{k,j} - h(x_k, y_j))^T Q_{k,j}^{-1} (z_{k,j} - h(x_k, y_j)) \right).$

这货就是所谓的最小二乘

3. 批量状态估计问题

- 我们把状态最大似然估计变成了最小二乘问题
- 对于原问题：

定义误差：

$$\begin{cases} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k) \\ \mathbf{z}_{k,j} = h(\mathbf{y}_j, \mathbf{x}_k, \mathbf{v}_{k,j}) \end{cases}.$$

$$\begin{aligned} \mathbf{e}_{v,k} &= \mathbf{x}_k - f(\mathbf{x}_{k-1}, \mathbf{u}_k) \\ \mathbf{e}_{y,j,k} &= \mathbf{z}_{k,j} - h(\mathbf{x}_k, \mathbf{y}_j), \end{aligned}$$

- 最小化误差的二范数：min

$$J(\mathbf{x}) = \sum_k \mathbf{e}_{v,k}^T \mathbf{R}_k^{-1} \mathbf{e}_{v,k} + \sum_k \sum_j \mathbf{e}_{y,k,j}^T \mathbf{Q}_{k,j}^{-1} \mathbf{e}_{y,k,j}.$$

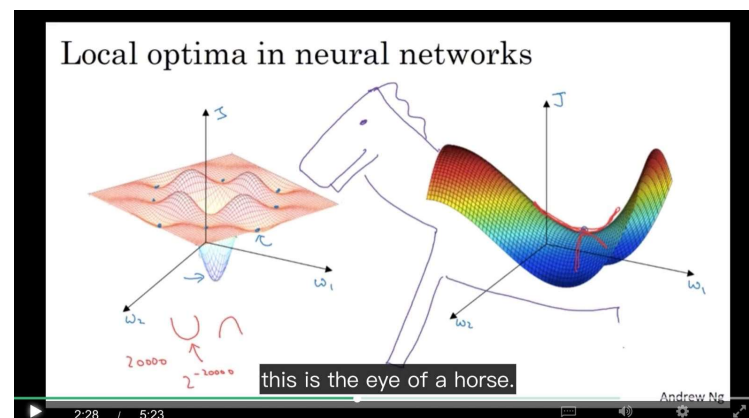
3. 批量状态估计问题

- 直观解释
 - 由于噪声的存在，当我们把估计的轨迹与地图代入SLAM的运动、观测方程中时，它们并不会完美的成立。
 - 此时就调整状态的估计，使得误差最小化
- 该问题有何结构？
 - 由许多个误差的平方和（或Sigma范数和）组成。
 - 虽然总体维度高，但每个项很简单，只关联2个变量。
 - 如果用李代数表达位姿，那么是无约束优化问题。
- 如何求解？
 - 下面先来介绍通用的非线性最小二乘问题。

4. 非线性最小二乘

4. 非线性最小二乘

- 先考虑简单的问题: $\min_x \frac{1}{2} \|f(x)\|_2^2$. 这里 $x \in \mathbb{R}^n$, f 为任意函数
- 当 f 很简单时:
 - 解: $\frac{df}{dx} = 0$. 将得到极值点或鞍点, 比较这些解即可
- 当 f 复杂时:
 - df/dx 难求, 或 $df/dx=0$ 很难解
 - 使用迭代方式求解



4. 非线性最小二乘

- 迭代方式:

1. 给定某个初始值 x_0 。

问题：如何确定这个增量？

2. 对于第 k 次迭代，寻找一个增量 Δx_k ，使得 $\|f(x_k + \Delta x_k)\|_2^2$ 达到极小值。

3. 若 Δx_k 足够小，则停止。

4. 否则，令 $x_{k+1} = x_k + \Delta x_k$ ，返回 2.

4. 非线性最小二乘

- 确定增量的方法（即梯度下降策略）：一阶的或二阶的
- 泰勒展开：
$$\|f(x + \Delta x)\|_2^2 \approx \|f(x)\|_2^2 + J(x) \Delta x + \frac{1}{2} \Delta x^T H \Delta x.$$

- 若只保留一阶梯度：

$$\min_{\Delta x} \|f(x)\|_2^2 + J \Delta x$$

增量的方向： $\Delta x^* = -J^T(x)$. （通常还需要计算步长）

- 称为最速下降法（Steepest Method）

4. 非线性最小二乘

- 若保留二阶梯度：

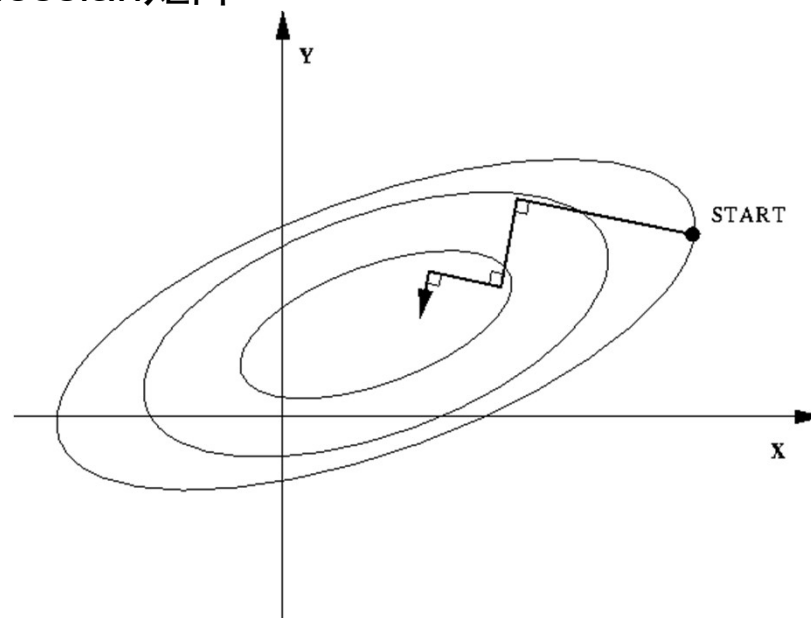
$$\|f(x + \Delta x)\|_2^2 \approx \|f(x)\|_2^2 + J(x) \Delta x + \frac{1}{2} \Delta x^T H \Delta x.$$

$$\Delta x^* = \arg \min \|f(x)\|_2^2 + J(x) \Delta x + \frac{1}{2} \Delta x^T H \Delta x.$$

- 则得到（令上式关于 Δx 的导数为零）：
$$H \Delta x = -J^T.$$
- 该方法又称为牛顿法

4. 非线性最小二乘

- 最速下降法和牛顿法虽然直观，但实用当中存在一些缺点
 - 最速下降法会碰到zigzag问题（过于贪婪）
 - 牛顿法迭代次数少，但需要计算复杂的Hessian矩阵
- 能否回避Hessian的计算？
 - Gauss-Newton
 - Levenberg-Marquadt



4. 非线性最小二乘

- Gauss-Newton

- 一阶近似 $f(x)$: $f(x + \Delta x) \approx f(x) + J(x) \Delta x$.
- 平方误差变为:

$$\begin{aligned} \frac{1}{2} \|f(x) + J(x) \Delta x\|^2 &= \frac{1}{2} (f(x) + J(x) \Delta x)^T (f(x) + J(x) \Delta x) \\ &= \frac{1}{2} \left(\|f(x)\|_2^2 + 2f(x)^T J(x) \Delta x + \Delta x^T J(x)^T J(x) \Delta x \right). \end{aligned}$$

- 令关于 Δx 导数为零:

$$2J(x)^T f(x) + 2J(x)^T J(x) \Delta x = 0.$$

$$J(x)^T J(x) \Delta x = -J(x)^T f(x).$$



$$H \Delta x = g.$$

4. 非线性最小二乘



$$J(x)^T J(x) \Delta x = -J(x)^T f(x).$$

$$H \Delta x = g.$$

- G-N用J的表达式近似了H
- 步骤:

1. 给定初始值 x_0 。
2. 对于第 k 次迭代, 求出当前的雅可比矩阵 $J(x_k)$ 和误差 $f(x_k)$ 。
3. 求解增量方程: $H \Delta x_k = g$.
4. 若 Δx_k 足够小, 则停止。否则, 令 $x_{k+1} = x_k + \Delta x_k$, 返回 2.

4. 非线性最小二乘

- Gauss-Newton简单实用，但 $\Delta x_k = H^{-1}g$ 无法保证H可逆
- Levenberg-Marquadt 方法一定程度上改善了它
 - G-N属于线搜索方法：先找到方向，再确定长度
 - L-M属于信赖区域方法（Trust Region），认为近似只在区域内可靠
 - 考虑近似程度的描述

$$\rho = \frac{f(x + \Delta x) - f(x)}{J(x) \Delta x} \quad \text{实际下降/近似下降}$$

- 若太小，则减小近似范围
- 若太大，则增加近似范围

4. 非线性最小二乘

- 改进版的G-N:

1. 给定初始值 \mathbf{x}_0 , 以及初始优化半径 μ 。

2. 对于第 k 次迭代, 求解:

$$\min_{\Delta \mathbf{x}_k} \frac{1}{2} \|f(\mathbf{x}_k) + \mathbf{J}(\mathbf{x}_k) \Delta \mathbf{x}_k\|^2, \quad s.t. \|D \Delta \mathbf{x}_k\|^2 \leq \mu, \quad (6.24)$$

这里 μ 是信赖区域的半径, D 将在后文说明。

3. 计算 ρ 。

4. 若 $\rho > \frac{3}{4}$, 则 $\mu = 2\mu$;

5. 若 $\rho < \frac{1}{4}$, 则 $\mu = 0.5\mu$;

6. 如果 ρ 大于某阈值, 认为近似可行。令 $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}_k$ 。

7. 判断算法是否收敛。如不收敛则返回 2, 否则结束。

4. 非线性最小二乘

- Trust Region内的优化，利用Lagrange乘子转化为无约束：

$$\min_{\Delta x_k} \frac{1}{2} \|f(x_k) + J(x_k) \Delta x_k\|^2 + \frac{\lambda}{2} \|D \Delta x\|^2.$$

- 仍参照G-N展开，增量方程为： $(H + \lambda D^T D) \Delta x = g.$
- 在Levenberg方法中，取 $D=I$ ，则： $(H + \lambda I) \Delta x = g.$

4. 非线性最小二乘



$$(H + \lambda I) \Delta x = g.$$

- LM相比于GN，能够保证增量方程的正定性
 - 即，认为近似只在一定范围内成立，如果近似不好则缩小范围
- 从增量方程上来看，可以看成一阶和二阶的混合
 - 参数 λ 控制着两边的权重

4. 非线性最小二乘

- 小结
 - 非线性优化是个很大的主题，研究者们为之奋斗多年
 - 主要方法：最速下降、牛顿、G-N、L-M、DogLeg等
 - 与线性规划不同，非线性需要针对具体问题具体分析
 - 问题非凸时，对初值敏感，会陷入局部最优
 - 目前没有非凸问题的通用最优值的寻找办法
 - 问题凸时，二阶方法通常一两步就能收敛

5. 实践: Ceres and g2o

5. 实践

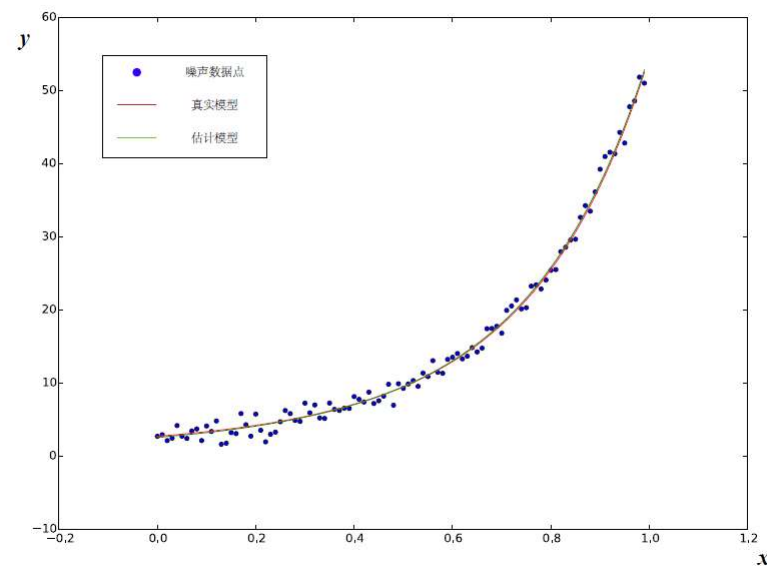
- Google Ceres Solver
 - 通用最小二乘问题求解库
 - 最一般的形式（带边界的核函数最小二乘）

$$\begin{aligned} \min_x \quad & \frac{1}{2} \sum_i \rho_i \left(\|f_i(x_{i_1}, \dots, x_{i_n})\|^2 \right) \\ \text{s.t.} \quad & l_j \leq x_j \leq u_j. \end{aligned}$$

- f 在Ceres中称为代价函数（Cost Function）， x 称为参数块（Parameter Block）

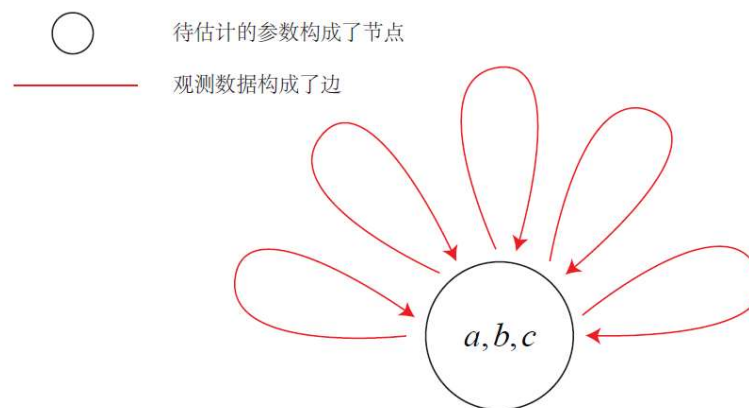
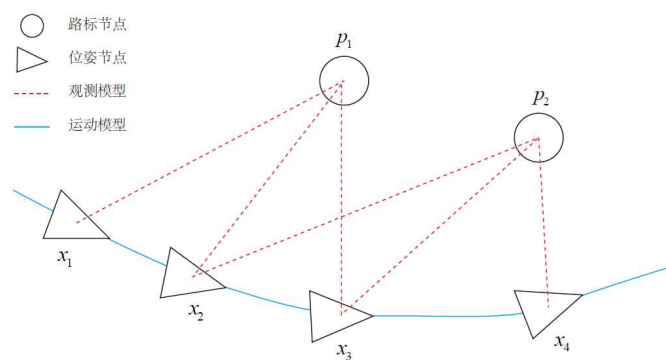
5. 实践

- 实验：曲线拟合
- 设曲线方程： $y = \exp(ax^2 + bx + c) + w$,
- 我们得到一些带噪声的样本数据：x, y
- 希望拟合（回归）曲线参数：a, b, c



5. 实践

- G2O中以图模型表达上述最小二乘问题



曲线拟合实验