EPnP 算法整理

EPnP 作为 PnP 算法的一种，与常见的求解参考点深度方案不同，如图 1 所示，用控制点的**加权和**来表示n个参考点，通过求解 4 个非共面的控制点（平面场景下需要 3 个控制点）的深度来减小计算量，进而计算参考点在相机坐标系下的坐标。在求解过程中，为减小计算量，使用一个12×12矩阵的零特征向量的加权和来求解；再通过以求解二次方程的小常数解的方式来选择合适的权重。之后将求得的封闭解作为 Gaussian-Newton 法的初值进行优化，得出更加精确的结果。

在解算过程中，EPnP 算法只需求解控制点的深度，而无需单独计算每个参考点，故而大大地提升了效率。

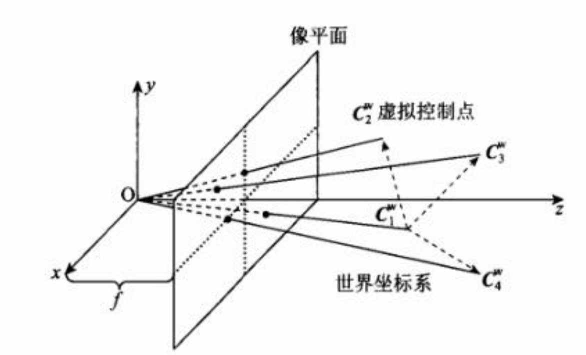

图 1 通过控制点求解 EPnP

一.　　算法步骤（思路）

对于所给定的n个参考点$\boldsymbol{p}_i, i = 1, \dots, n$，已知其 3D 世界坐标$\boldsymbol{p}_i^w$和 2D 图像投影，投影坐标为$\boldsymbol{u}_i$。首先确定合适的控制点$\boldsymbol{c}_j$和权值$\alpha_{ij}$；之后由相机坐标系和像素坐标系之间的转化关系列写线性方程组。

$$s_i \begin{bmatrix} \boldsymbol{u}_i \\ 1 \end{bmatrix} = \boldsymbol{K}\boldsymbol{p}_i^c = \boldsymbol{K}\sum_{j=1}^{4} \alpha_{ij}\boldsymbol{c}_j^c$$

每个参考点可以提供 2 个方程约束，这样就可以得到$2n$个方程组成的方程组，其中未知数数目为 12（每个控制点的 3D 坐标，共 4 个控制点）。将方程组转化为矩阵的形式

$$\boldsymbol{M}\boldsymbol{x} = \boldsymbol{0}$$

其中$\boldsymbol{M}$为$2n \times 12$矩阵，$\boldsymbol{x}$为 12 维向量。为简化求解过程，使用12×12矩阵$\boldsymbol{M}^T\boldsymbol{M}$的零特征向量的加权和来表示解向量$\boldsymbol{x}$，这样得出权值$\beta_i$后即可得到方程组的解。

$$\boldsymbol{x} = \sum_{k=1}^{N} \beta_k \boldsymbol{v}_i$$

其中N为矩阵$\boldsymbol{M}^T\boldsymbol{M}$的零特征值数目，$\boldsymbol{v}_i$为矩阵的零特征向量。利用空间欧氏变换前后不同点之间的距离不变的特性，可对β_k进行求解。

$$\left\| \sum_{k=1}^{N} \beta_k \boldsymbol{v}_k^{[i]} - \sum_{k=1}^{N} \beta_k \boldsymbol{v}_k^{[j]} \right\|^2 = \left\| \boldsymbol{c}_i^w - \boldsymbol{c}_j^w \right\|^2$$

其中$\boldsymbol{v}_k^{[i]}$为特征向量$\boldsymbol{v}_k$中代表控制点$\boldsymbol{c}_i$的子向量。

在得出特征向量权值β_k后即可计算出控制点的相机坐标$\boldsymbol{c}_j^c$，再通过开始时确定的控制点权值

α_{ij}，即可得出参考点的相机坐标$\boldsymbol{p}_i^c$。这样就将 2D-3D 位姿估计问题转化为 3D-3D 位姿估计问题，可通过 ICP 等算法求解位姿。

为减小误差得到更高精度的解，可以将之前求得的封闭解作为初值，以特征向量权值β_k作为优化变量，以不同控制点的间距在欧氏变换前后的变化量作为误差，通过 Gaussian-Newton 等方法，进行迭代优化，就得到了最大精度下的解。

二.　　确定控制点坐标

理论上，4 个控制点的坐标可任意选取。但是在 Lepetit 的论文中，作者表示在实践中发现，参考点的重心作为控制点之一时，算法的结果较为稳定。首先，将参考点的重心作为第一个控制点

$$\boldsymbol{c}_1^w = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{p}_i^w$$

之后构建矩阵$\boldsymbol{A}$，$\boldsymbol{A}$为$n\times 3$矩阵。

$$\mathrm{A} = \begin{bmatrix} \boldsymbol{p}_1^{w^T} - \boldsymbol{c}_1^{w^T} \\ \vdots \\ \boldsymbol{p}_n^{w^T} - \boldsymbol{c}_n^{w^T} \end{bmatrix}$$

这样$\boldsymbol{A}^T\boldsymbol{A}$为$3\times 3$矩阵，3 个特征值分别为$\lambda_1$、$\lambda_2$和$\lambda_3$；3 个特征向量为$\boldsymbol{v}_1$、$\boldsymbol{v}_2$和$\boldsymbol{v}_3$。那么控制点$\boldsymbol{c}_2^w$、$\boldsymbol{c}_3^w$和$\boldsymbol{c}_4^w$为

$$\boldsymbol{c}_j^w = \boldsymbol{c}_1^w + \lambda_{j-1}^{\frac{1}{2}}\boldsymbol{v}_{j-1}$$

其中$j = 2,3,4$。确定了 4 个控制点的世界坐标，那么对应的控制点权值α_{ij}就很容易得到。

三.　　求解控制点的相机坐标

在得到控制点的世界坐标后，接下来就要通过矩阵$\boldsymbol{M}$求解控制点的相机坐标。在参考点的 2D 像素坐标已知的情况下，方程的解在矩阵$\boldsymbol{M}$的核空间中。

3.1. 以特征向量的加权和表示控制点坐标

将相机坐标系和像素坐标系之间的转化关系展开为

$$s_i\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}\sum_{j=1}^{4}\alpha_{ij}\begin{bmatrix} x_j^c \\ y_j^c \\ z_j^c \end{bmatrix}$$

在该式中，4 个控制点的 12 个坐标值为未知量。将其转化为方程组的形式

$$s_i u_i = \sum_{j=1}^{4}\alpha_{ij}f_u x_j^c + \alpha_{ij}u_0 z_j^c$$

$$s_i v_i = \sum_{j=1}^{4}\alpha_{ij}f_v y_j^c + \alpha_{ij}v_0 z_j^c$$

$$s_i = \sum_{j=1}^{4}\alpha_{ij}z_j^c$$

将最后一行代入前两行中，可消去s_i

$$\sum_{j=1}^{4} \alpha_{ij} f_u x_j^c + \alpha_{ij}(u_0 - u_i) z_j^c = 0$$
$$\sum_{j=1}^{4} \alpha_{ij} f_v y_j^c + \alpha_{ij}(v_0 - v_i) z_j^c = 0$$
这样，每一个参考点可以提供两个方程作为约束，n个参考点则能够提供$2n$个方程。将方程组以矩阵的形式表示
$$\boldsymbol{Mx} = \boldsymbol{0}$$
其中$\boldsymbol{M}$为$2n \times 12$矩阵，$\boldsymbol{x}$为方程的未知数，为 12 维向量。
$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{c}_1^{c\,T} & \boldsymbol{c}_2^{c\,T} & \boldsymbol{c}_3^{c\,T} & \boldsymbol{c}_4^{c\,T} \end{bmatrix}^T$$
易知，该方程组的解在$\boldsymbol{M}$的零空间（核空间）内，也就是是矩阵$\boldsymbol{M}$的零奇异值所对应的奇异向量构成的线性空间内。而$\boldsymbol{M}$的零奇异向量同时也是矩阵$\boldsymbol{M}^T\boldsymbol{M}$的零特征向量。$\boldsymbol{M}^T\boldsymbol{M}$是$12 \times 12$矩阵，当$n$的值很大时，计算$\boldsymbol{M}^T\boldsymbol{M}$零特征向量的复杂度要远小于计算$\boldsymbol{M}$零奇异向量；而计算$\boldsymbol{M}^T\boldsymbol{M}$的复杂度为$O(n)$，为整个计算过程中最耗时的步骤，同样远小于计算$\boldsymbol{M}$零奇异向量的复杂度。因此，通过将方程组的解表示为$\boldsymbol{M}^T\boldsymbol{M}$零特征向量加权和的形式，可大大降低解算过程的计算量。即
$$\boldsymbol{x} = \sum_{k=1}^{N} \beta_k \boldsymbol{v}_k$$
其中$\boldsymbol{v}_k$为$\boldsymbol{M}^T\boldsymbol{M}$的零特征向量，$N$为$\boldsymbol{M}^T\boldsymbol{M}$的零特征值数目（与参考点数目$n$不同），$\beta_k$为特征向量权值。

3.2. 求解合适的特征向量权值

在计算出矩阵$\boldsymbol{M}^T\boldsymbol{M}$后，很容易求出其特征值，及对应的特征向量。在 Lepetit 的论文中提及，N的值与相机焦距有关。

当焦距较小时，适用透视相机模型，需要至少 6 对参考点，此时矩阵$\boldsymbol{M}^T\boldsymbol{M}$的右零空间的维数恰好为 1，即$N = 1$。

焦距较大时，适用正交相机模型，改变控制点深度对参考点的投影没有影响，此时$N = 4$。由于所提供对应的参考点坐标常常带有噪声，因此矩阵的 4 个零特征值在实际运算中并不严格为零，会有微小的值。

Lepetit 的论文中提及，N的取值可以为 1~4 的整数，由参考点的配置、相机焦距、噪声数量等因素决定。为避免特征值过于接近而导致的在选取N的值时出现的错误，Lepetit 在文中建议将$N = 1\sim4$时的权值全部计算出，并在结果中选取重投影误差最小的一项。这一方法在参考点提供的约束不足时仍然适用。
$$\mathrm{res} = \sum_i \mathrm{dist}^2 \left(\boldsymbol{K}[\boldsymbol{R}|\boldsymbol{t}] \begin{bmatrix} \boldsymbol{p}_i^w \\ 1 \end{bmatrix}, \boldsymbol{u}_i \right)$$
相比于计算$\boldsymbol{M}^T\boldsymbol{M}$，计算重投影误差所需的计算量非常小，却大大加强了算法的鲁棒性。在计算特征向量权值时，可利用空间欧氏变换前后不同点之间的距离不变的特性列出方程，对β_i进行求解。
$$\left\| \sum_{k=1}^{N} \beta_k \boldsymbol{v}_k^{[i]} - \sum_{k=1}^{N} \beta_k \boldsymbol{v}_k^{[j]} \right\|^2 = \left\| \boldsymbol{c}_i^w - \boldsymbol{c}_j^w \right\|^2$$
其中$\boldsymbol{v}_k^{[i]}$为特征向量$\boldsymbol{v}_k$中代表控制点$\boldsymbol{c}_i$的子向量。接下来讨论在N的不同取值下，求解β_k的不

同方法。

当$N=1$时，仅有 1 个β。此时可列出方程

$$\left\|\beta\boldsymbol{v}^{[i]}-\beta\boldsymbol{v}^{[j]}\right\|^2=\left\|\boldsymbol{c}_i^w-\boldsymbol{c}_j^w\right\|^2$$

则β的封闭解为

$$\beta=\frac{\sum_{\{i,j\}\in[1;4]}\left\|\boldsymbol{v}^{[i]}-\boldsymbol{v}^{[j]}\right\|\cdot\left\|\boldsymbol{c}_i^w-\boldsymbol{c}_j^w\right\|}{\sum_{\{i,j\}\in[1;4]}\left\|\boldsymbol{v}^{[i]}-\boldsymbol{v}^{[j]}\right\|^2}$$

当$N=2$时，有β_1、β_2

$$\boldsymbol{x}=\beta_1\boldsymbol{v}_1+\beta_2\boldsymbol{v}_2$$

可列出方程

$$\left\|\left(\beta_1\boldsymbol{v}_1^{[i]}+\beta_2\boldsymbol{v}_2^{[i]}\right)-\left(\beta_1\boldsymbol{v}_1^{[j]}+\beta_2\boldsymbol{v}_2^{[j]}\right)\right\|^2=\left\|\boldsymbol{c}_i^w-\boldsymbol{c}_j^w\right\|^2$$

这是一个关于β的二次方程，为便于计算，以线性化的方式进行求解。以β_{12}代替$\beta_1\beta_2$，β_{11}代替β_1^2，β_{22}代替β_2^2，这样方程即可改为求解$[\beta_{11}\quad\beta_{12}\quad\beta_{22}]^T$。就 4 个控制点而言，由$i,j$的不同取值，共有$C_4^2=6$个方程，因此用矩阵表示方程组为

$$\boldsymbol{L\beta}=\boldsymbol{\rho}$$

其中$\boldsymbol{L}$为6×3系数矩阵，$\boldsymbol{\beta}$为 3 维未知数向量，$\boldsymbol{\rho}$为 6 维向量，表示控制点间距离。可通过$\boldsymbol{L}$的伪逆来求解β_{ab}的值；之后可通过参考点的深度为正的规律，确定β_a的符号；进而得出β_a的值，求得每个控制点的坐标。

当$N=3$时，有β_1、β_2、β_3。与$N=2$的情况类似，将二次方程线性化后，求解向量$\boldsymbol{\beta}$。此时$\boldsymbol{\beta}$为 6 维向量

$$\boldsymbol{\beta}=[\beta_{11}\quad\beta_{12}\quad\beta_{13}\quad\beta_{22}\quad\beta_{23}\quad\beta_{33}]^T$$

而$\boldsymbol{L}$则为6×6矩阵。此时可通过$\boldsymbol{L}$的逆矩阵求解$\boldsymbol{\beta}$，进而得出每个控制点的坐标。

当$N=4$时，有β_1、β_2、β_3、β_4，线性化后$\boldsymbol{\beta}$为 10 维未知数向量。那么 6 个方程的约束不足以直接解出 10 个未知数。此时考虑到乘法的交换性

$$\beta_{ab}\beta_{cd}=\beta_a\beta_b\beta_c\beta_d=\beta_{a'b'}\beta_{c'd'}$$

其中$\{a',b',c',d'\}$为$\{a,b,c,d\}$的另一种排列方式。这样可以减小未知数的个数，例如

$$\beta_{23}=\frac{\beta_{12}\beta_{13}}{\beta_{11}}$$

如此，可解出$\boldsymbol{\beta}$，进而得出每个控制点的坐标。

3.3. 平面情况下的求解方法

在平面情况下，所有的参考点都位于同一平面上，此时只需要 3 个控制点，$\boldsymbol{M}$为$2\mathrm{n}\times9$矩阵。$\boldsymbol{M}^T\boldsymbol{M}$的特征向量也相应地变为 9 维，但是上文所提及的关于矩阵$\boldsymbol{M}$的约束依然有效，只是方程数目从 6 个减为 3 个。因此，在$N\geq3$时，就需要使用之前非平面形式下在$N=4$时才需要用到的减小未知数个数的方法。

四. Gaussian-Newton 优化

在前文得出$\boldsymbol{\beta}$的封闭解的基础上，以之作为初值，将$\boldsymbol{\beta}=[\beta_1\quad\beta_2\quad\beta_3\quad\beta_4]^T$作为优化变量，以不同控制点的间距在欧氏变换前后的变化量作为误差，可得出目标函数

$$\mathrm{Error}(\boldsymbol{\beta})=\sum_{(i,j)\ s.t.\ i<j}\left(\left\|\boldsymbol{c}_i^c-\boldsymbol{c}_j^c\right\|^2-\left\|\boldsymbol{c}_i^w-\boldsymbol{c}_j^w\right\|^2\right)$$

这是一个无约束的非线性最优化问题，使用 Gaussian-Newton 法求解即可。此时优化变量仅限于$\boldsymbol{\beta}$，最多不超过 4 个，与参考点数目无关，因此优化过程中的计算量极低，可视为常数。一般情况下，这一步骤的优化迭代次数不会超过 10 次。计算封闭解与优化的整体复杂

度可视为与参考点数目呈线性关系。
在优化得出最优解后，即可计算出参考点在相机坐标系下的坐标$\boldsymbol{p}_i^c$，将 2D-3D 位姿估计问题转化为 3D-3D 位姿估计问题。之后求解位姿即可（可通过 ICP 等算法）。

参考

[1] Lepetit V, Moreno-Noguer F, Fua P. EPnP: An Accurate O(n) Solution to the PnP Problem[J]. International Journal of Computer Vision, 2009, 81(2): 155.
[2] JesseChen79.深入 EPnP 算法[EB/OL]. https://blog.csdn.net/jessecw79/article/details/82945918, 2018-10-5.
[3] 哀酱.理解 EPnP 算法[EB/OL]. https://blog.csdn.net/u010821666/article/details/80738359, 2018-8-14.