在自动驾驶中，激光雷达是一个不可缺的传感器，但实际自动驾驶车辆上不会单独出现激光和相机，还会有其他传感器的参与，如 IMU、GPS 等。如下图所示，即各传感器的初始参数（内外参）结合各传感器数据，然后计算各传感器的姿态信息，并根据各传感器数据估计的姿态更新传感器数据，并输出地图信息。

相机与激光雷达之间，关键在于准确地传感器姿态（主要是图像姿态），而图像姿态中最主要的一点在于时间同步。不管对于机械式激光雷达还是对于固态激光雷达，其授时机制都比较丰富，但对于相机来说，只能对相机进行触发曝光，无法控制其曝光到成像的时间，所以这里的相机时间，一般是指相机曝光后的时间，通常的解决方法为测量相机的平均延时 d，然后以在主机上获取到图像帧时的主机时刻 t 减去这个延时（t-d）作为该帧图像的时间戳。当然，也可以采用触发时间当作相机时间戳，理论上就可以完美的对上了。

但实际应用中，统计时间包括从触发到 soc 接收到的时间，以及硬件上的误差，其时间戳并不能完整的对应上，所以需要进行专门进行时空同步。这里衍生出单独进行时间同步、空间同步以及同时进行时空同步两种方法。这里分别简单介绍一下。

**时间同步：**

首先声明，这里的时间同步方案只能用在标定校准，无法在车上实时的运行。主要通过对比序列图像估计的姿态信息、激光雷达点云估计的姿态信息以及车体姿态信息，获得图像和对应时间点的相对姿态信息，以及激光和对应时间点的姿态信息。这就涉及到图像姿态曲线或激光点云姿态曲线与车体姿态曲线相似度的求解。以图像姿态为例。假设给定两条时间序列为：x={x0,x1,...xn}和 y={y0,y1,...yn}。其相似性度量就是找到一个合适的度量函数 Sim(x;y)来衡量这两条时间序列之间的相似性。一般采用 Minkowski 距离法和动态时间弯曲距离法作为度量函数。

Minkowski 距离是一种适应于比较简单的相似性度量的距离度量方法，两条等长时间序列之间的 Minkowski 距离：

$$d(X,Y)=\left(\sum_{i=1}^{n}|x_i-y_i|^{p}\right)^{\frac{1}{p}}$$

以相机估计的 pitch 和车体的 pitch 为例，当利用 Minkowski 距离方法计算相机估计的 pitch 序列与车体 pitch 之间的匹配度时，某一个时间点的 Minkowski 距离为最小，那么就可以认为此时的 pitch 与图像序列计算出的 pitch 是对应的。

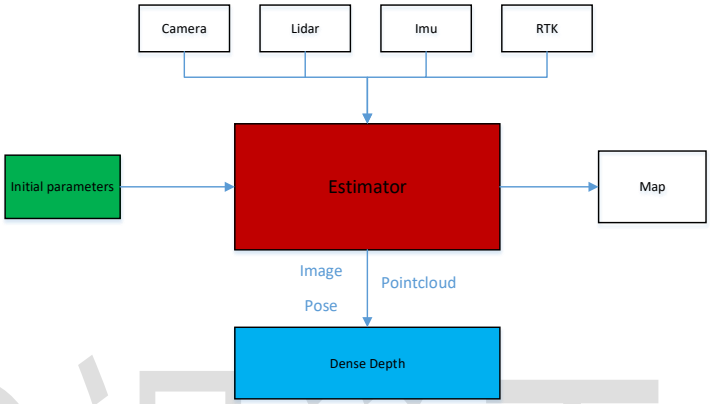
动态时间弯曲距离(Dynamic Time Warping Distance, DWT)，是基于动态规划的思想，通过构建一个邻接矩阵，寻找最短路径和的方法。DTW 是一个典型的优化问题，其用满足一定条件时间的弯曲路径 P 来描述输入模板和参考模板的时间对应关系，求解两个模板匹配时累计距离最小所对应的一条路径。该算法缺点在于计算比较复杂，运行时间较长。

由于完全相同的两条时间序列基本上不存在，而时间间隔的存在和时间序列数据较为复杂，且有线性平移、弯曲等问题，所以需要选择误差最小即最佳相似度来描述两个时间序列间的关系。上述两种方法各自有其不足之处：Minkowski 距离法不能度量时间序列存在线性漂移和时间扭曲的情况，且对不等长的时间序列无能为力；而动态时间弯曲距离法运行时间

比较复杂，误差不可避免，所以需要权衡考虑，这也是为啥只能适合离线校准。

**空间同步：**

目前，激光雷达与相机配准方法主要有如下几类：一是先提取点、线等特征集，然后通过激光点云与图像特征之间的匹配，求解两者之间的配准参数；或者是先计算激光点云与相机图像之间的互信息损失函数，然后利用优化算法求解相对位姿参数，这类方法比骄适合平直道路上传感器相对位姿移动较小的情况；再或者是采用多个相机或者相机历史帧信息进行对相机进行 SFM 算法，这种方法也是比较适合目前的自动驾驶配置，感兴趣的可以试验一下。下面给出单相机与单激光的流程图：




具体步骤如下：

1）测量计算 IMU 传感器、相机和激光雷达三者之间的空间位置关系，将所述空间位置关系定义为探测系统的空间同步初始值，并定义探测系统的时间同步初始值为 0，将所述空间同步初始值和所述时间同步初始值作为迭代优化求解的初始设置；

2）通过 IMU 传感器获取 IMU 数据，通过相机获取图像数据，通过激光雷达获取点云数据；

3）对所述图像数据进行语义分割及持征点提取，根据语义分割结果对所述特征点进行匹配，构建重投影误差方程，并将 IMU 传感器与相机之间的第一时间偏差引入所述重投影误差方程；

4）引入所述 IMU 传感器与激光雷达之间的第二时间偏差，对所述点云数据中两帧点云进行位姿修正，对修正后的所述两帧点云进行配准，计算所述两帧点云之间的相对位姿；

5）获取两帧图像间的 MU 数据，通过预积分计算两帧图像的第一位姿，获取两帧点云间的 IMU 数据，通过预积分计算两帧点云的第二位姿，计算所述第一位姿和第二位姿之间的位姿偏差；

6）设定滑动窗口，根据所述滑动窗口内所述重投影误差方程、所述相对位姿、所述位姿偏差进行选代优化求解，实现多传感器时间空间标定。

这里的特征提取是成功的关键，可以将上述流程图中的点云与相机分开提特征，也可以将图像与点云结合生成 RGBD 图像，其中图像提供像素坐标系下的 x，y 坐标，激光直接提供相机坐标系下的距离信息。根据 RGBD 图像的信息和相机的内外参，可以计算出任何一个像素点的世界坐标系下的坐标，同样世界坐标系下的点也能依据 RGBD 信息和相机内外

参求解出其在图像中的像素坐标。这样就将标定问题转换为如何获取 RGBD 图像信息的问题了。

首先根据图像提供的像素坐标系下的 x，y 坐标（即下列公式中的 u，v）和相机内参求出相机坐标系下的 X，Y 坐标值。假设目标在相机坐标系下的距离为 Z，所以该目标在相机坐标系下的坐标为

$$P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

那么，相机坐标系 P 和像素坐标系$P_{uv}$的坐标关系公式如下：

$$ZP_{uv} = Z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = KP$$

将上式整理后的具体求解公式如下：

$$X = Z(u - c_x)/f_x$$
$$Y = Z(v - c_y)/f_y$$
$$Z = d$$

那么如何正确获取相机坐标系下的深度信息，根据世界坐标系到像素坐标系下点的坐转换公式：

$$ZP_{uv} = Z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = KP = K\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = K(R\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + t)$$

或者将隐含齐次坐标转非齐次坐标：

$$ZP_{uv} = Z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = KP = K\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = KT\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

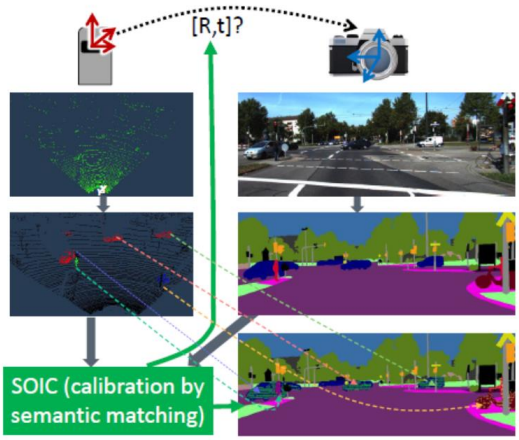
其中：

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

上述就是世界坐标系$P_w$到像素坐标系$P_{uv}$下的点的坐标关系，利用上述关系，依据世界坐标系下的点的坐标$p_w$（激光到 IMU 之间的关系获得），就可以得出像素坐标$[u, v]$和深度 Z，从而建立激光-IMU-相机的关系。如果设定世界坐标系和相机坐标系是重合的，即没有旋转和平移，这时实质就是以相机坐标系建立的点云。这时相机坐标系和世界坐标系的相对位姿

的旋转矩阵 R 就为单位矩阵，位移向量 t 就是零向量。如果在对应点的位置再加入彩色信息就能够构成彩色点云。

上面是一种构建 RGBD 图像的方法，依赖于传感器的内外参矩阵。还有一种提取图像点云语义信息的激光与相机标定方法，就是引入语义质心，将标定问题转换为 PnP 问题，通过构建点云和图像语义之间对应关系，最小化代价函数来估计最优的传感器外参。




具体步骤如下：

1）利用现有的方法对图像和点云进行了预训练，得到语义分割结果，基于这些语义分割结果,通过语义质心(SCs)来得到一个初始的位姿估计值

2）在语义对应信息约束下，定义了代价函数

3）以初始值为基准，对代价函数进一步的优化,得到更加精确的参数

假设采用$P^L = \{\mathbf{p}_1^L, \mathbf{p}_2^L, \ldots, \mathbf{p}_n^L\}$来表示点云集，其中$\mathbf{p}_i^L = (x_i, y_i, z_i) \in \mathbb{R}^3$，n 表示点云数量，L 表示激光雷达坐标系，点云标签定义为$\ell_i^{pcd} \in S$。另外假设采用$\ell_{[l,m]}^{img} \in S, S = \{0, 1, 2 \ldots N\}$表示像素的语义标签，$I[l,m]$表示像素类别，其中$l \in [0, W] \text{ and } m \in [0, H]$，由于分辨率的差异，像素的数量远远大于点云的数量。那么从点云 PL 到相机 Pc 坐标系下的转换可以表示为旋转角$\boldsymbol{\theta} = (\theta_x, \theta_y, \theta_z)$和平移向量$\mathbf{t} = (t_x, t_y, t_z)$，所以，激光坐标系下的点可以通过下式转换到相机坐标系下：

$$\mathbf{p}_i^C = \mathcal{R}(\theta) \cdot \mathbf{p}_i^L + \mathbf{t}$$

在结合相机内参 K 和投影函数$\mathcal{P}$，那么可以将相机坐标系的点投影到像素坐标$[u^i, v^i]$下：

$$[u^i, v^i] = \mathcal{P}\left(\mathbf{K}, \mathbf{p}_i^C\right)$$

所以，求解激光雷达与相机之间的相对关系可以转化为求解点云标签$\ell_i^{pcd} \in S$与像素标签$\ell_{[u^i,v^i]}^{img} \in S$一致性最大化的问题。两者之间的代价函数可以定义为：

$$\mathcal{C} = 1 - e^{-\epsilon^{-1}\left(\ell_i^{pcd} - \ell_{[u^i,v^i]}^{img}\right)}$$

对于转换到相机坐标系的点$p_i^C$，如果超出图像或者与像素标签不一致，则通过定义一个距离函数 D 来计算原始激光雷达坐标系下的点$p_i^L$的损失。

$$\mathcal{D}\left(\mathbf{p}_i^L\right) = \min_{\ell_{[l,m]}^{img} \ell_i^{pcd}} \left(\mathcal{M}\left([u,v]^i,[l,m]\right)\right) \left|p_i^L\right|^2$$

因为这个函数计算图像中具有相同标签的曼哈顿距离. 因此上式的 M 定义如下:

$$\mathcal{M}([u,v],[l,m]) = |u-l| + |v-m|$$

由于语义分割会有多个类别, 可以对不同类别进行了加权, 这样可以根据不同类别对损失函数进行计算, 我们定义的类别加权函数$1_A$ 如下:

$$1_A(x) := \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

结合上述的公式, 给出点云和图像的最终的代价函数, 分母表示有效的语义标签的个数

$$\mathcal{L} = \frac{\sum_{s\in S}\sum_i^n 1_{\{s\}}\left(\ell_i^{pcd}\right)\mathcal{C}\left(\mathbf{p}_i^L\right)\mathcal{D}\left(\mathbf{p}_i^L\right)}{\sum_{s\in S}\sum_i^n 1_{\{s\}}\left(\ell_i^{pcd}\right)}$$

对于上式，可以通过最小化代价函数来求解外参$\hat{\boldsymbol{\theta}}$和$\hat{\mathbf{t}}$

$$\hat{\boldsymbol{\theta}}, \hat{\mathbf{t}} = \arg\min_{\boldsymbol{\theta},\mathbf{t}} \mathcal{L}$$