System：：pubImageData()函数：

trackerData().readImage对每一帧图像进行跟踪，先对图像亮度进行调整；cur表示当前帧，forw表示下一帧，prev表示上一帧。当前传入的图像是forw。在cur不为空（即已经初始化）的情况下，用光流法追踪forw中的关键点。reduceVector根据追踪的status对成员变量（prev\_pts（像素坐标），cur\_pts，forw\_pts， ids，cur\_un\_pts（归一化坐标），track\_cnt）进行更新。ids是个vector，存储点的编号（后续记录ids如何记录点的编号）。只有status是好的，reduceVector才会把点留下。un下标代表经过去畸变的点，存储的是归一化坐标，track\_cnt表示当前点被追踪到的次数。未经过初始化时不进行光流法追踪。

对追踪到的点的track\_cnt自增。

对成员变量进行rejectwithF()验证，用本质矩阵去除outlier，只保留不是outlier的成员变量，同样通过reduceVector函数。由于检测新的特征点在reduceVector()后面进行，因此对prev\_pts和cur\_pts进行保留时能保证保留的是匹配点。

对追踪到的点进行蒙版setMask，不让追踪到的特征点被后续提取特征点重复提取。还会对追踪到的特征点根据追踪次数排序，让追踪次数多的点排在forw\_pts前面，ids和track\_cnt也要改变次序。

如果当前帧追踪的特征点比较少，则进行提取特征点。提取到特征点通过addPoint()添加到forw\_pts中，新点对应的ids push\_back（-1）。

更新成员变量，cur赋值给prev，forw赋值给cur，通过undistortedPoints()得到去畸变后的点的归一化坐标，并在cur\_un\_pts中存储。并计算特征点在图像中的运动速度（看程序是归一化x和y的速度？），保存在pts\_velocity中，用于下一帧光流法跟踪。

trackerData().readImage结束后用trackerData().updateID对新点的编号进行更新；如果新点的ids为-1，则赋值为n\_id，同时对n\_id进行自增，这样保证了新的点的id号越来越大。

为什么轨迹舍弃前几帧的位姿？