“Empirical studies show that LSD-SLAM, which relies on depth ﬁlters, performs consistently more powerfully than ORB-SLAM” – Mobile SLAM

However, now there is an orb-slam2 – We should copy both methods in C++ to compare performance.

Inverse Depth Maps - The inverse depth parameterisation represents a landmark's distance, d, from the camera exactly as it says, as proportional to 1/d within the estimation algorithm. The rational behind the approach is that, filtering approaches such as the extended Kalman filter (EKF) make an assumption that the error associated with features is Gaussian.

In a visual odometry setting the depth of a landmark is estimated by tracking the associated features over some series of frames and then using the induced parallax. However for distant features (relative to the displacement of the camera) the resultant parallax will be small, and importantly the error distribution associated with the depth is highly peaked close to the minimum depth with a long tail (i.e. it is not well modelled via a Gaussian distribution). To see an example should refer to Fig. 7 in Civera et al.'s paper (mentioned by @freakpatrol), or Fig. 4 of [Fallon et al. ICRA 2012](http://homepages.inf.ed.ac.uk/mfallon2/publications/12_fallon_icra.pdf).

By representing the inverse depth (i.e. 1/d) this error becomes Gaussian. Furthermore it permits representing very distant points e.g. points at infinity.

The important aspect of the representation used is Civera's paper is explained in Section II B of his paper (see Equation (3)). Here, a landmark is represented relative to the pose (position and orientation) of the first camera from which it is seen. This pose is capture in the first five parameters of Eq (3), whereas the sixth parameter, *ρi*ρi , represents the inverse depth. Eq (4) provides an expression for recovering the world position of the point (i.e. this where the inverse depth gets converted to depth as 1/*ρi*1/ρi )