1. 背景
2. 相关论文的做法及效果
3. 自己想做的点：二进制描述子，可以在马太原学长的基础上做，也可以重新设计整个网络，直接生成二进制描述子，不过可以借鉴其生成匹配特征点的方式
4. 马太原学长的论文里面已经突出的优点在我的论文里面就一笔带过，主要实验就比较二进制，突出实时性，内存占用，帧率，最好能在tx2上跑
5. 马太原学长论文里面有一个创新点：就是slam的匹配机制显示之前深度学习方法，在代价函数上有理论缺陷，导致深度学习描述符在slam里面效果比较差。我就结合一下这个创新点，进行二进制。

看论文的点：用的什么相机，神经网络的架构，损失函数，如何生成用来学习的数据，用了什么数据集，比如用的什么特征点（fast，surf），如何训练，最终达到了什么效果，有什么突出的优点。看的论文只要集中在神经网络描述子一块，开始不必专注于是否二进制展示

Geometric Correspondence Network for Camera Motion Estimation

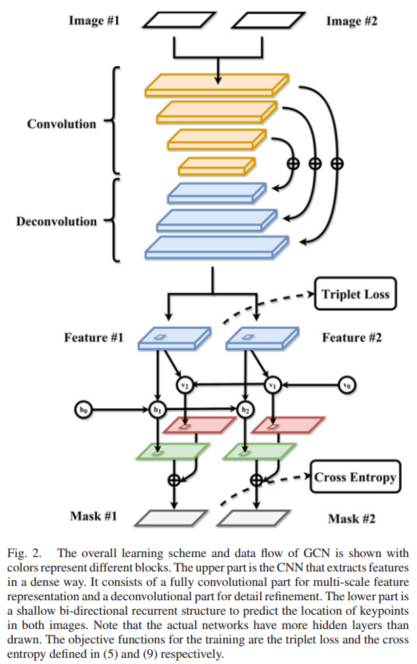
-----IEEE ROBOTICS AND AUTOMATION LETTERS, VOL. 3, NO. 2, APRIL 2018

贡献：提出了一种学习关键点和描述子的检测的统一框架

相机：RGB-D（似乎没用深度信息，为啥不用mono，TUM只是用RGB-D采集的，后面也用了KITTI数据集，所以整个过程与深度信息没有关系）

神经网络架构：结合了RNN的CNN网络进行训练来检测关键点的位置，以及生成结构相同的描述子。网络的优化由带刚体变换的从source frame到reference frame的点的warping。训练主要关注相机的运动，而不是图片内的运动，这样更有利于匹配。

CNN+ shallow bi-directional recurrent convolutional network. 用了ResNet-50和ADAM 优化器。



损失函数：CNN层用triplet loss，可以使相似的samples更近，不相似的samples更远，能够实现最近邻（相似）的匹配

RNN层用的cross entropy，用来做分类

训练的数据：网络的训练是提取高梯度的点（Harris算法生成），并利用相机运动转换到下一张图片，因为有噪声或者标定误差的存在，可能测试得到的真值与实际的值有误差，所以使用KNN（K Nearest Neighbor）算法弥合这个误差。没有使用标准的描述子，而是网络自己去寻找能够完美匹配的特征和描述子。事实上，转换的图片中的点可能用基于梯度的方法都没法检测到

特征点：高梯度的点，没有使用标准的特征点

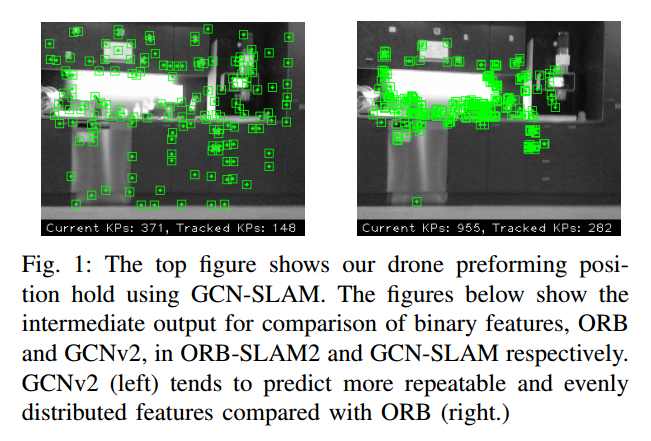
数据集：TUM、KITTI，有位姿的真值，可以生成训练数据

缺点：网络太深，描述子不是二进制。后面可以用浅的网络（Deep Compression，SqueezeNet），描述子二进制化（CNN网络）。

GCNv2: Efficient Correspondence Prediction for Real-Time SLAM

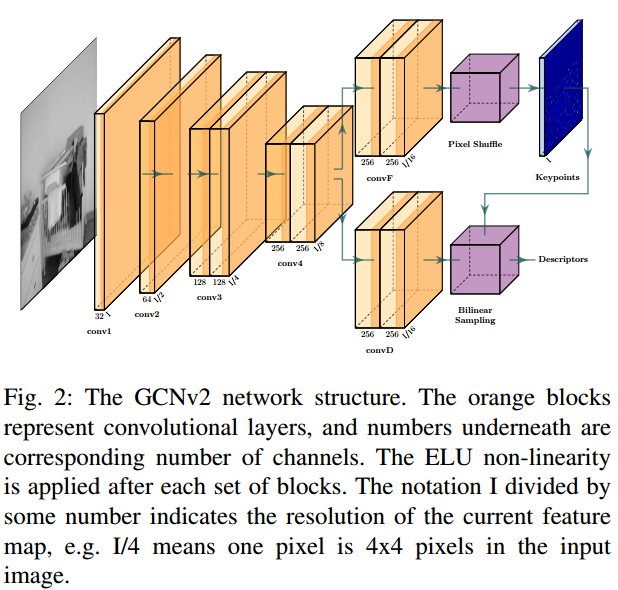
----ICRA2018

贡献：计算效率超过GCN，可以在TX2上面运行，GCNv2 feature可以代替ORB。描述子二进制化（连续帧的匹配时利用Hamming距离（二进制数字异或操作）可以大大加速，生成的词袋也更小，词袋加载可以占用更少的资源）

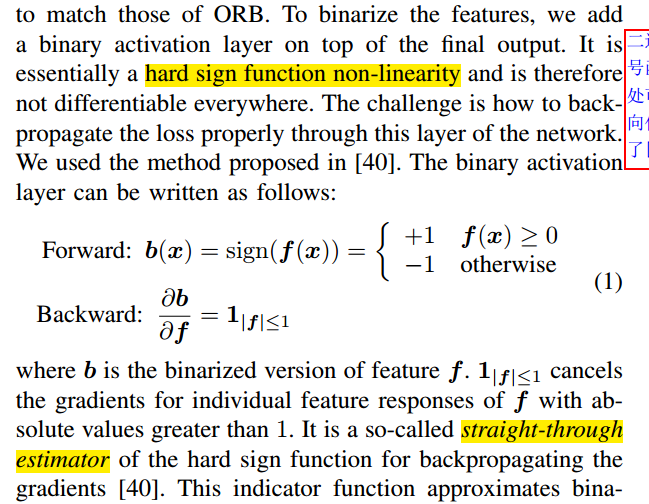


相机：RGB-D

神经网络架构：FCN（加ResNet-50 backbone）+bidirectional recurrent convolutional network，FCN用来提取feature，recurrent network用来寻找关键点的位置



二进制化使用了符号函数，而其不是处处可微，不能反向传播，所以



这种方法可以在反向传播的时候逼近二值化，并且避免对超过边界1的feature response进行过度的惩罚。这种训练方法比强迫网络最小化量化损失来预测一个二进制值更有效

损失函数：triplet loss，cross entropy

训练的数据：

特征点：高梯度的点，没有使用标准的特征点

数据集：

缺点：

ASD-SLAM: A Novel Adaptive-Scale Descriptor Learning for VisualSLAM

----IV2020 Taiyuan Ma

贡献： 提出一种尺度自适应损失函数和基于scale reminder的sampling（采样）策略 来生成更好的局部特征描述子（ASD），并设计了一个SLAM系统叫ASD-SLAM。因为不需要手动设定尺度修正参数，所以训练更容易。ASD对于视角变化大和高重复场景有优势。

相机：RGB-D

神经网络架构：

损失函数：

训练的数据：

特征点：

数据集：

缺点：

贡献：

相机：RGB-D

神经网络架构：

损失函数：

训练的数据：

特征点：

数据集：

缺点：