**主要思想：特征匹配很重要，是视觉SLAM的瓶颈。提出了一种深度学习二进制描述子用于SLAM，并且训练了词袋做回环检测，既提高了精度，又保证了效率，有实际意义。精度从FPR95，mAP说明。（不提学长的ASD，直接从hardnet上拿出来，然后直接得到SLAM系统）**

主要就是将神经网络二进制描述符用于视觉SLAM，效果更好。

核心思想就是设计了一种视觉SLAM二进制描述子，提高了精度，保证了效率，有实际意义。

关键点提取，描述符计算，描述子匹配，后端优化

基于深度学习二进制描述符的SLAM系统研究

Abstract：基于特征匹配的SLAM受描述符的影响比较大，CNN对于image patch信息的提取的效果比较好，适用于生成描述符。所以本文设计了一种基于CNN的描述符，效果比较好，在数据集HPatches和Brown dataset上的鲁棒性都比较好，改进的SLAM系统在数据集kitti和tartanair上的结果比传统的ORB-SLAM2要好

介绍二进制描述符的生成，主要是几个损失函数介绍一下，然后就是改进的ORB-SLAM2，进行对比实验，说明我的二进制描述符的FPR95比较好，HPatches数据集的几种模式我的这个表现好，来间接说明我的二进制描述符比一般的更鲁棒，然后说我的这个效率比浮点型的高，回环检测时间比浮点型的短。

系统框架：描述符的生成（几个损失函数介绍），SLAM系统的搭建（ORB-SLAM2）.

实验：Brown Dataset的FPR95；HPatches的图片检索等等（来说明我的描述符比较鲁棒）；Tartanair数据集的SLAM的对比；与浮点型描述子的回环检测时间的对比；

对比实验：Tanh和ReLU的对比实验

BASD-SLAM: A Deep-Learning Visual SLAM System Based On Binary Adaptive-Scale descriptor

Abstract:

The feature quality plays an important role in visual SLAM (Visual Simultaneous Localization and Mapping) based on feature matching, and becomes the bottleneck of positioning accuracy improvement. Now lots of hand-crafted descriptors like BRIEF and ORB don't work very well in complex scenarios. The Convolutional Neural Network is proved to have tremendous advantages on image feature extraction. In this paper, we design a CNN model to extract binary visual feature descriptor from image patches. Based on this deep feature descriptor, we design a monocular SLAM system, named BASD-SLAM, by replacing ORB descriptor in ORB-SLAM2. We also train visual Bag of Words to detect loop closure. Experiments show that our BASD achieves better results on the HPatches dataset and UBC benchmark. In the meantime, the BASD-SLAM system outperforms other current popular SLAM system on KITTI odometry dataset and Tartanair dataset.

描述子的质量在基于关键点的描述子匹配的视觉SLAM中很重要，成为了提高定位精度的瓶颈。现在大量的手工打造的描述子，如ORB，SIFT在复杂场景效果不是很好。而CNN被证明在图片编码方面有很大的优势。所以本文设计了一种基于CNN的二进制视觉描述子，并利用ORB-SLAM2的框架设计了SLAM系统。深度学习二进制描述子可以在提高轨迹估计精度的同时，也能提高特征匹配和后端优化效率。大量在brown dataset，HPatches和tartanair数据集上的实验证明了描述子的鲁棒性和高精度性。

1. INTRODUCTION

背景

发展

现在的问题和研究空白

自己的研究如何填补这个空白

Introduction：

Visual SLAM has got prosperous development in recent years. The result of feature matching in the keypoint-based vSLAM system depends on the descriptor quality. The traditional descriptors rely on the pixel-level

视觉SLAM发展迅速。对于基于特征点的视觉SLAM来说，特征匹配的结果取决于描述子的质量。传统的描述子进行像素级的匹配，匹配误差会累积，最终影响pose estimation result。

近些年，随着深度学习的发展，基于深度学习的图像处理变得越来越火。深度学习对于提取图像的特征，然后进行图像的关联有着巨大的优势。而视觉SLAM的图像处理和数据关联正是深度学习的强项。所以现在有很多的研究者都将深度学习与SLAM结合起来，以解决视觉SLAM中图像数据关联的瓶颈问题。

深度学习在SLAM中的应用提升了SLAM的精度。有许多学者利用深度学习的优势，替换了传统视觉SLAM中的部分模块，比如描述子匹配，重定位等等。也有的使用高级别的语义信息，给系统增加对世界的理解能力，对于特征匹配和特征关联带来了很大的提升。也有端到端的SLAM系统，直接由图片生成位姿估计，在特定场景获得了比较好的效果。

但是当前的深度学习在SLAM中的使用还是有较多问题。比如端到端的SLAM很依赖特定场景，系统泛化能力不强。语义SLAM无法保证场景有相关的语义信息，比如slam++中的椅子。而低级别的描述子信息的提取没有考虑到尺度问题，导致Hardnet等等描述子不适用于SLAM

在本文中，我们提出了一种提取图片局部描述子的网络。我们提取出的描述子的形式与传统的描述子ORB类似，可以很容易地适用在当前流行的视觉SLAM系统上。并且我们考虑到了尺度问题（引用学长），让其更加适用于SLAM。为了匹配效率，我们对生成的描述子进行了二进制化。我们也为自己描述子单独训练了词袋，回环检测的精度得到提高。

Related work

Local feature descriptor

讲讲传统的描述子，浮点型的，二进制型的。讲讲CNN描述子，浮点型的，二进制型的。这些描述子都只是用于image verification,等等，没有在SLAM和SFM上进一步测试，

Local Feature Descriptor

传统：SIFT，ORB，BRIEF，SURF

DL：Lift，SuperPoint，LF-NET，MatchNet，L2-Net，SuperGlue

1. 深度学习浮点型描述符：Lift，SuperPoint，LF-Net，多分支CNN网络：triplet network，L2-net，hardnet
2. 深度学习二进制描述符：各种二进制方法

Deep learning enhanced SLAM

替换模块的深度学习SLAM

语义SLAM：https://blog.csdn.net/fxr123fxr/article/details/82988485?utm\_medium=distribute.pc\_relevant.none-task-blog-BlogCommendFromBaidu-3&depth\_1-utm\_source=distribute.pc\_relevant.none-task-blog-BlogCommendFromBaidu-3

端到端SLAM：DeepVO

Proposed method

Deep learning enhanced SLAM

Contribution

1. We propose a binary descriptor with CNN model using four loss function, and outperforms other traditional descriptor on accuracy and effectiveness.

2. We design a monocular system with our learned descriptor, and achieve better results than other traditional visual SLAM system on several benchmark datasets.

3 System Overview（SLAM框架）

In our BASD-SLAM system, we still adopt traditional visual SLAM pipeline. ORB-SLAM2 is a classical visual SLAM system. Unlike other end-to-end SLAM system, we just replace the traditional hand-crafted descriptor ORB with our learned descriptor and evaluate the efficiency and effectiveness of our descriptor. This also enables our descriptor suitable to other SLAM system like SFM.

3.1 Local feature design

Compared with hand-crafted methods, learned descriptors has tremendous advantages, such as compact structure, evenly distribution, robust to noise and so on. Moreover, learning-based descriptors are data-adaptive. In order to make our descriptor more effective, we just adopt shallow convolutional neural network to generate our descriptor, and the shallow network has also been proved to be suitable to extract low-level image information [14]. float descriptors sacrifice the effectiveness of feature matching and loop closure. Instead, our shallow network will obtain binary local feature descriptor, and also maintain the high precision.

[15] reveals that triplet network has greater advantages in metric learning than Siamese network, so we also adopt the former to train our descriptor. There are eight convolutional layers, each of which is followed by a Tanh non-linearity and Batch Norm operations. And the output of network is normalized to unit-length. In order to reduce the possibility of overfitting, we add a dropout layer in the last of our network. After lots of tuning step and training process, we set the dropout rate to 0.3.

Loss function plays an important role in descriptor generation. We adopt four loss type to train our descriptor. We will describe in detail below.

1. Adaptive-Scale Triplet Loss

Triplet loss has been proved to have great advantages in descriptor generation. So we also adopt this loss function. [ASD-SLAM] proposed the scale uncertainty influence in triplet loss, and modified the prime triplet loss function to reflect the changes of scale by adding a scale reminder factor. Given three image patches, Pa, Pb and Pc, which represent the anchor, positive and negative image patches. After the reasoning of network, we get descriptors xa, xb and xc respectively. And the adaptive-scale triplet loss function is defined as





Where d- and d+ is the L2 distance of anchor descriptor with negative descriptor, anchor descriptor with positive descriptor, respectively.

Because we set batch size to 1024, so the choice of d- and d+ matters. We also adopt the adaptive-scale sampling strategy to obtain suitable d- and d+.

The native training strategy is too complex and performs not well, so we turn to the hard negative mining strategy proposed in [hardnet], which is proved to be effective and easy to converge in training.

1. Even-Distribution Loss

The distribution of binary bits reflects the encoding quality of neural network. In large dataset, same bit of every descriptor generated by all image patches should have same numbers of -1 and +1 roughly. However, the sign function is not differentiable, so we cannot reduce even-distribution loss by optimizing the numbers of -1 and +1. We just constraint the means of every float descriptor dimension in one batch size descriptors to 0. Even-distribution loss is defined as :



1. Quantization Loss

In quantization step, we use sign function to obtain binarization result of float descriptor. However, the difference between real-value and ±1 can bring a great drop in accuracy. So we minimize the quantization loss to get a better binary descriptor. Quantization loss is defined as:



1. Correlation Loss

In order to make the descriptor contain more information, the bits of every descriptor should have less correlation[L2-net]. So we introduce the correlation loss penalty to get more differentiable descriptors. We use the descriptor generated from anchor image patch, where  is row vector of one image descriptor.

The correlation matrix  is defined as:



Where  is mean of ith row of . Obviously, the off-diagonal elements of R should be 0. So the correlation loss is:



3.2 SLAM System (An Approach for CNN-Based Feature Matching Towards Real-Time SLAM)

ORB-SLAM2 is a fantastic visual SLAM work in recent year, which is suitable for monocular camera based on PTAM structure. So we choose ORB-SLAM2 as our SLAM system. we can substitute our learned descriptor for ORB easily, because our learned descriptor has same structure with ORB. The CNN model is embedded in descriptor extractor after FAST keypoint detection using the implementation of pytorch c++ API. We organize the image patches as a single tensor, and transfer to CNN model, so the model can reason all the image patches with one step, which can accelerate the reasoning time.ORB-SLAM2 implement the Bag of Words to detect loop closure, so we also train Bag of Words with the descriptor reasoned by our CNN model. Because the difference of descriptor, we adjust the matching threshold in SLAM system.

4.Experiments

4.1 descriptor evaluation

4.1.1 UBC benchmark dataset

UBC benchmark dataset, consisting of three datasets, Yosemite, Notredame and Liberty, is suitable for training descriptors, whose patches are centered on real interest point detection. So we use it to evaluate our model. We just use one dataset to train our model, and the other two to evaluate the model output. We compare it with other hand-crafted and learned local descriptors with FPR95 standard. The result is listed in TABLE I. We can conclude that our descriptor outperforms others.

4.1.2 HPatches dataset

4.1.3 Matching result in hard scenarios

The performance in dealing with hard scenarios like illumination change and view change is very important to SLAM. So we choose the hand-crafted descriptor, ORB, to make a comparison with our learned descriptor. Although we do RANSAC in SLAM system, the performance will be bad if the mismatch number exceeds the match number. We choose large illumination change and large view change pictures to evaluate descriptors. From the match results, our descriptor outperforms the ORB descriptor, which show the robustness of our descriptor.

|  |  |
| --- | --- |
| C:\senior\SLAM\slam\ASDNet\model_test\图片匹配测试结果\view change\v_charing\BASD_4_6.png | C:\senior\SLAM\slam\ASDNet\model_test\图片匹配测试结果\view change\v_charing\ORB_4_6.png |
| BASD | ORB |

4.2 SLAM system evaluation

4.2.1 Evaluation of KITTI Odometry dataset

我们的描述子能够在kitti上比ORB更快地进行初始化，说明我们的描述子在视角快速切换的时候更鲁棒。

4.2.2 Evaluation of Tartanair dataset

To evaluate the accuracy and robustness of our learned descriptor in BASD-SLAM, we introduce the Tartanair dataset for localization and mapping evaluation. We compared our system with ORB-SLAM2, including the evaluation standard, ATE( absolute trajectory error) and SR(success rate). In order to present the robustness of our descriptor, we respectively choose three contexts in Tartanair, Soul-City, Japanese-Alley, Ocean. The evaluation results are shown in TABLE I. The bold represents the better result. We also choose the evo evaluation tool to estimate and visualize some context trajectory. And the results are shown in TABLE II. From the evaluation results above, we can easily draw the conclusion that our learned descriptor SLAM system outperforms the traditional descriptor SLAM system ORB-SLAM2.

为了评估系统的鲁棒性，我们用Tartanair数据集来进行定位和建图的评估，我们将我们的系统与ORB-SLAM2进行比较，评估了ATE和SR。为了体现出我们的描述子的鲁棒性，我们分别选择了Tartanair数据集中的三个场景：Soul-city，Janpanese-alley，Ocean。ATE和SR的评估结果如下表。相对更好的结果用粗体标识出来了。我们也是用了evo评估工具对几个场景进行了评估和可视化，结果如图。从以上可以看出，我们的SLAM系统比传统的ORB-SLAM2系统更加鲁棒，取得更好的表现

四个损失函数的讲解

Slam系统的搭建，词袋的训练，

Experiments

Brown dataset的fpr95

图片的匹配

Tartanair的实验

Kitti数据集

[14] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, and T. Brox, “Discriminative unsupervised feature learning with exemplar convolutional neural networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 9, pp. 1734–1747, Sep. 2016.

分析一些学长的introduction和related work主要写了啥

研究背景和研究重要性

引出该领域的研究空白

点题-指出本文的研究课题

概述文章的核心方法论和主要发现

提出猜想和研究目的

给出自己的contribution

ASD-SLAM

Introduction

SLAM依赖特征匹配，依赖描述子的质量。过去几十年传统描述子发展起来，介绍一下ORB 、SIFT优缺点。也出现了许多深度学习描述子。比如DeepSec，L2-Net，CS L2-Net，HardNet等等。他们有一些比传统好。但是不适用与SLAM。主要原因就是没考虑到尺度性。大多数人专注于数据增强和更稳定数据集增强鲁棒性，忽视了损失函数的重要性。说一下自己的contribution：