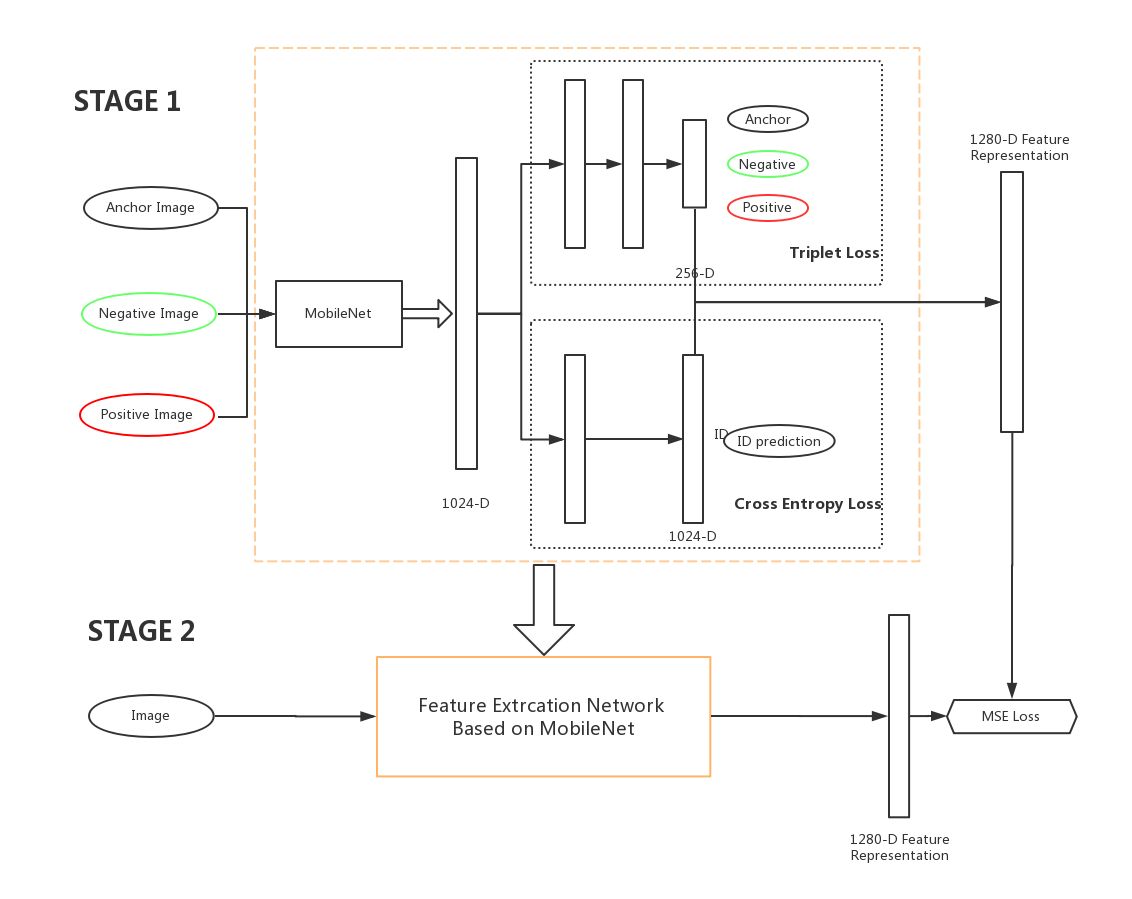
**整体框架**

整体训练结构如下图所示：训练的时候分3个步骤

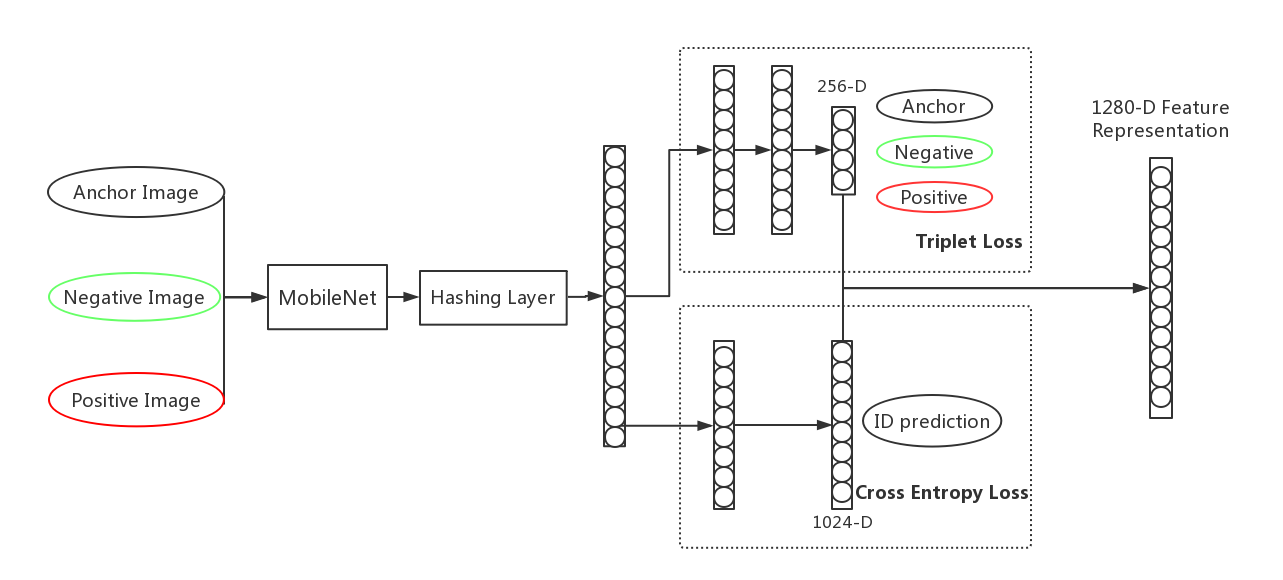
1. 首先训练ID的cross entropy loss分支(**STAGE1**)。这边可以用muti-task训练，添加color和model的分支。性能会提升1-2个点。
2. 1)训练收敛后训练Triplet loss的分支(**STAGE1**)
3. 然后使用Average Feature的方式finetune网络(**STAGE2**)。



**Deep Supervised Hash**

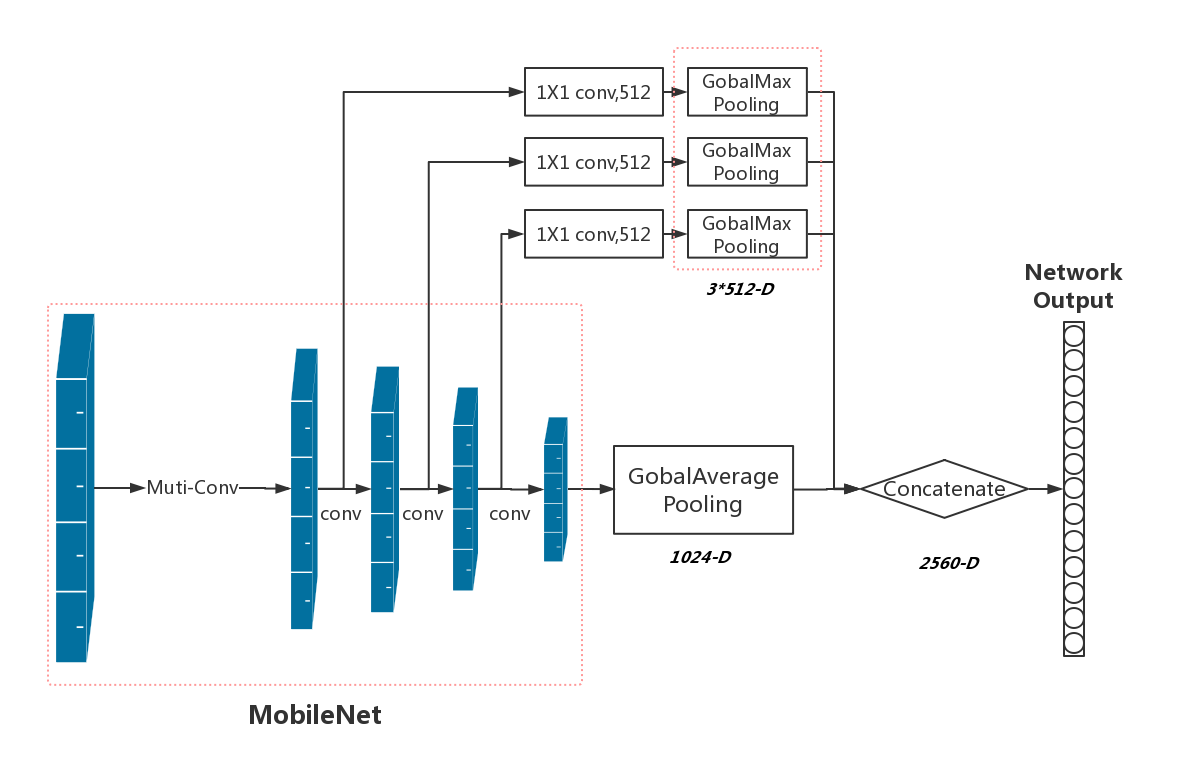
针对以上的模型，提出了改进措施：使用了Deep Supervised Hashing来进行性能优化。

1. 提升检索速度，由之前计算欧式距离或者余弦距离变成计算汉明距离。
2. 降低特征大小，降低了32倍，大大减低存储特征的空间开销。
3. 性能上的提升。



**局部特征学习**

backbone网络(MobileNet)直接学习得到的特征对局部信息没有特别好的描述。我们提出了新的结构(Local Feature Enhancement)来提升整体性能。其中更改的网络结构如下所示：训练的时候只需要将下面的网络



**性能列表**

性能表格如下(MGN和PCB是我们复现版本，用来做对比，其余都是)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **描述** | **mAP** | **Top1-Acc** | **Top5-Acc** | **Top20-Acc** | **Top20-Recall** | **Top50-Recall** |
| **TripleLoss+Color+Model** | 0.554 | 0.628 | 0.8 |  |  |  |
| **Bseline** | 0.766 | 0.846 | 0.917 | 0.943 | 0.566 |  |
| **TripleLoss+ID+Color+Model** | 0.779 | 0.863 | 0.924 | 0.955 |  |  |
| **Bseline +Local feature enhence** | 0.818 | 0.883 | 0.934 | 0.957 | 0.622 | 0.636 |
| **PCB** | 0.787 | 0.867 | 0.921 | 0.951 | 0.603 | 0.624 |
| **MGN** | 0.806 | 0.877 | 0.93 | 0.955 | 0.613 | 0.63 |
| **Bseline + Local feature enhance**+DSH | 0.837 | 0.892 | **0.941** | **0.962** | 0.645 | 0.656 |
| **Bseline +Local feature enhance**+DSH+reranking(200) | **0.867** | 0.893 | 0.932 | 0.957 | **0.67** | **0.671** |

**代码使用介绍**

训练阶段

1. train\_V35\_step1.py使用该代码来训练cross entropy loss，收敛后
2. 使用train\_V35.py该代码来训练triplet loss分支
3. 然后求各个类在特征空间的中心，使用train\_V35\_distill.py来进行第三阶段的训练。
4. 训练完成

测试阶段

1. 先使用generate\_mobilenet\_triplet\_features\_db.py来生成gallery的特征
2. 然后使用query\_triplet\_test\_ballTree\_DSH.py来测试性能