Image retrieval

李德运 SA18006165

1.实验要求

利用所给的2050（50\*41）图像，设计并实现一个图像检索程序。对已有的图像建立一个library，对于一副输入图像，找到与之最为相近K幅图像，计算检索结果的准确率（precision）、召回率（recall）、F1-value和MRR的值。

2.实验原理

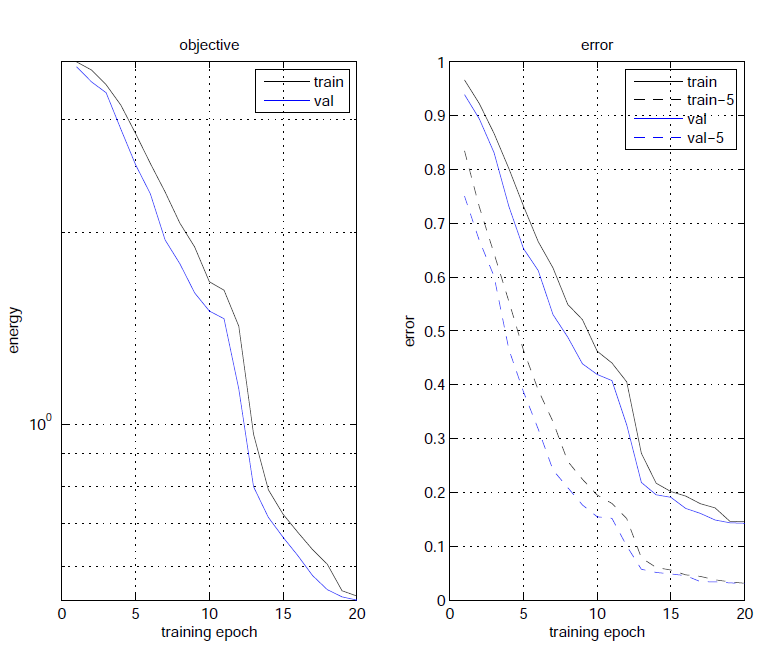
采用卷积神经网（CNN），训练出一个多分类器。CNN网络包含3个卷积层，3个pooling层，3个relu层，两个全连阶层，1个Softmax层。各层的具体参数可以参见my\_cnn.m文件。

以全部图像为训练样本进行训练。对于每幅图像，先进行预处理后将图像降采样至32\*32，再进行训练或者检索；对于黑白图像，将gray图像复制给rgb三个通道，因此所有图像在CNN的输入都为32\*32\*3。

简单的采用Softmax层的输出为图像特征向量，计算不同图像特征向量之间的欧式距离，距离越小的认为越接近。

CNN代码使用了matconvnet工具包[1]。

3.实验结果



如上图，训练的误差和validation（使用训练集合作为alidation集）误差率如图。迭代次数取20次。可以发现在接近20的时候误差下降已经趋于平缓。

最后的evaluation如下表所示：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| K | Precision | Recall | F1 | MRR | t(all image) |
| 10 | 0.8263 | 0.1653 | 0.2754 | 0.5113 | -- |
| 20 | 0.8128 | 0.3251 | 0.4645 | 0.6183 | -- |
| 50 | 0.7184 | 0.7184 | 0.7184 | 0.7418 | -- |
| 100 | 0.4005 | 0.8009 | 0.5340 | 0.7546 | 0.620589 s |

平均每幅图像的test时间为0.62/2000=0.31ms

4.代码说明

本实验代码使用了matconvnet工具包，使用matlab语言编写。

由于训练和测试产生的数据集过于庞大，因此只给出源代码。

代码使用如下：

1.安装matconvnet

2.运行/data/rpoj4\_data/getimdb.m脚本，产生imdb.mat文件

3.运行/code/my\_cnn.m文件，训练cnn网络

4.运行/code/matconvnet-1.0-beta8/example/get\_res.m文件，得到test结果（result.mat）文件。

5.运行evaluation.m脚本。得到结果。

5.实验总结

卷积神经网相对于传统的图像检索方法，准确率更高（使用简单的图像直方图也可以做，使用SIFT算子也可以做）。使用CNN的好处在于，不需要认为的去提取特征。很明显，图像的分类检索涉及到了图像语义信息，而人为定义的图像feature往往是比较底层的（如颜色，坐标，亮度等等），语义层的feature比较难定义。CNN算法很好的解决了这一问题。

此外，本实验对CNN的使用是比较粗略的，因此准确率也不高。对CNN各层的设计、参数的调整还没有很深刻的认识，这些还有待学习。

另外，本实验的准确率不高与图像预处理也有关系，将图像压缩至32\*32，信息丢失过多。但同时训练的速度很快（2050幅图像只用了6分钟不到）。

6.参考

[1]. Vedaldi A, Lenc K. MatConvNet-Convolutional Neural Networks for MATLAB[J]. arXiv preprint arXiv:1412.4564, 2014.

[2]. <http://www.tuicool.com/articles/MzMbEv2>

[3]. L. Fei-Fei, R. Fergus and P. Perona. *One-Shot learning of object ategories.* IEEE Trans. Pattern Recognition and Machine Intelligence. Inpress.