机器学习第二次作业实验报告

图像分类及语义分割

摘要：本次实验主要解决两个问题：判断一幅图像中是否包含人手；将人手部分标记出来。对第一个问题，我们主要用线性核 SVM 对图像进行分类；对第二个问题，我们用 Unet 来处理。第一问当我们用 75%的数据作为训练集，用 25%的数据作为验证集验证性能时，得到的准确率在 96%~97%左右。第二问当我们用 90%的数据作为训练集，10%的数据作为验证集验证性能时，得到的 dice 系数和 iou 系数分别是 0.04 和 0.06 左右。

1. 导言

图像分类问题由来已久，在现实生活中也有着广泛的应用。常见的图像分类算法一般可分为两种：一是传统机器学习方法，二是深度学习方法。在本次实验中我们采用传统机器学习方法：先用 HOG 特征描述子提取特征，再用线性 SVM 对图片进行分类。HOG 特征子来源于[1]，HOG 特征子是在图像的局部方格单元上提取特征，因此它对图像几何和光学形变能保持较好的不变性。类似地，本次实验我们将 HOG 特征子应用于人手检测问题。

图像语义分割问题的历史则相对较短。在深度学习出现之前，常用的方法是基于随机森林分类器等传统机器学习方法，此类方法的一大弊端是预测效率极低，很难应用于实时场景。深度学习方法的出现极大地推进了图像语义分割问题的发展，全卷积神经网络[2]的出现使得任何大小的输入都可以被接受。之后逐渐出现了 encoder-decoder 架构、空洞架构等新的框架。本次实验中，我们使用基于 encoder-decoder 架构的 Unet。Unet 分为 encoder 和 decoder 两个结构，encoder 结构用于减少空间维度、提取抽象特征，decoder 用于恢复空间维度和细节信息。

2. 实验过程

对第 1 问，我们有如 Fig.1 所示的算法流程：

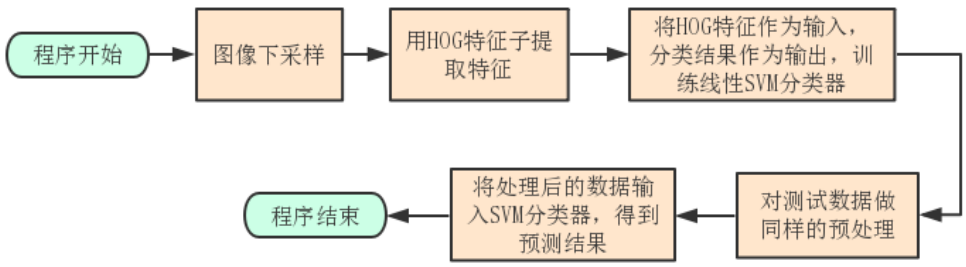


Fig.1 Algorithm flowchart for problem 1

原图像的大小为$480 \cdot 640$，如果直接用来训练模型，需要比较长的时间，如

果先将图像缩小再进行训练，可以有效减少训练时间。由于人眼的关注点一般都在亮区，所以在经过最大下采样之后，人眼仍然能轻松识别图像中的人手。所以，我们有理由认为，即使经过下采样，模型应该也能识别图像中的人手，性能上不应该有太大的差异。

通过观察数据集我们可以发现，数据集中的人手的光照分布极不均匀，并且人手的动作非常丰富。HOG 特征对图像几何和光学形变能保持较好的不变性。因此我们采用 HOG 特征描述子提取图像特征。

在得到用 HOG 特征子提取的图像特征后，我们测试了多种机器学习算法，如随机森林、高斯核 SVM、朴素贝叶斯等等，最后发现线性核 SVM 是比较合适的，因此我们采用了这种方法。

对第 2 问，我们有如 Fig.2 所示的算法流程：

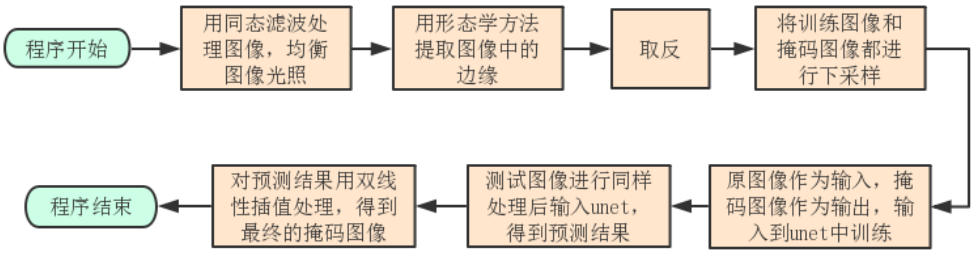


Fig.2 Algorithm flowchart for problem 2

我们注意到数据集中的光照极不均匀，有的图像偏亮，有的图像偏暗，同时训练集又比较小，因此有必要对图像进行预处理，使得图像特征更加鲜明、更易于学习。

将图像进行同态滤波、提取边缘之后，我们对图片做取反操作，这是因为我们在实验过程中发现将图像取反后进行训练能得到更好的结果，但是更深层次的原因有待探索。

基于和第一问类似的理由，在将图像输入 Unet 训练之前，我们同样先对图像进行下采样处理。

将测试图像做同样的处理，再输入 Unet 中，得到预测结果。由于我们的 unet 针对的是经过下采样的图像，因此得到的输出也是经过下采样的图像。为了将图像恢复成原来的大小，我们用双线性插值的方法处理图像。

3. 结果分析

3.1 实验环境

硬件：

处理器 Intel® Core™ i5-7200U CPU @2.50GHz 2.71GHz

内存 8.00GB

软件：

Windows10, Python 3.6.8

3.2 参数说明

3.2.1 Q1

下采样：最大下采样，窗口大小为(2,2)

HOG 特征描述子：cell：(8,8)，block：(2,2)，orientations：9
线性核 SVM：损失函数为 hinge，即

$$L(y_{pred}) = \max(0, 1 - y_{true} \cdot y_{pred})$$

其他参数为 sklearn LinearSVC 默认参数

3.2.2 Q2

Unet 结构如下，我们的模型在参数设置上略有不同，具体参数可以参考 UNet.py。

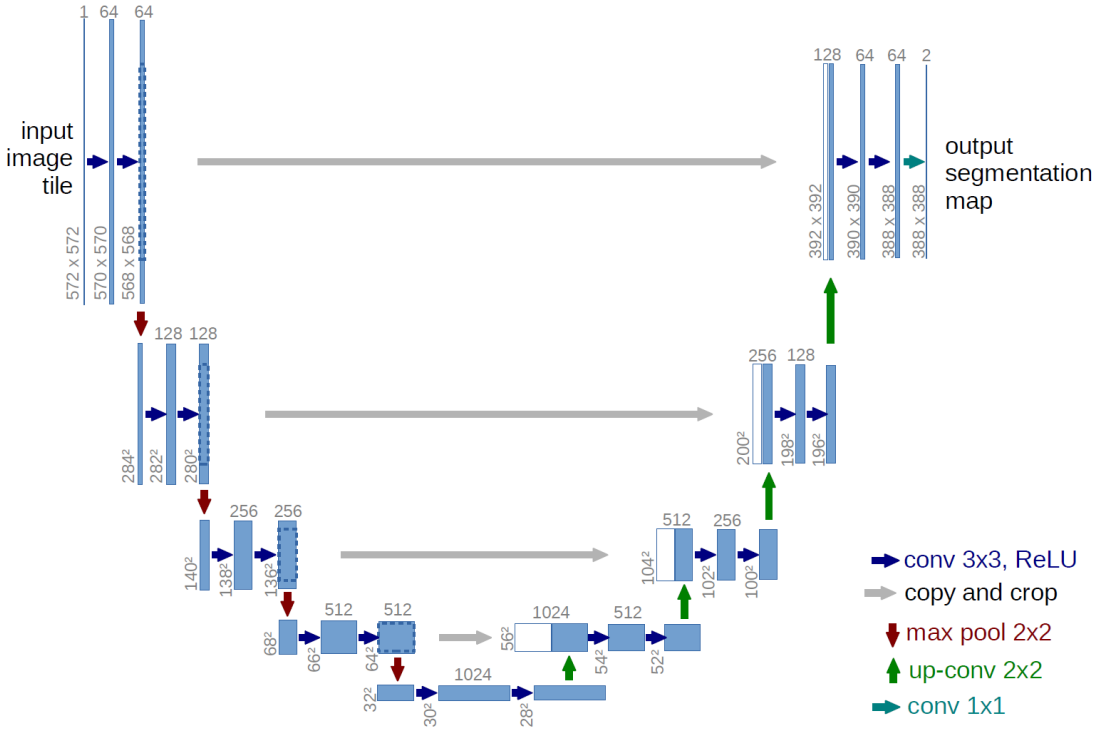


Fig.3 Unet Architecture

3.3 实验结果

3.3.1 Q1

当我们用 75%的数据进行训练，25%的数据进行测试时，得到的结果如 Fig.4 所示。可以看到，和随机森林分类器进行对比时，线性 SVM 分类器的效果要好很多。

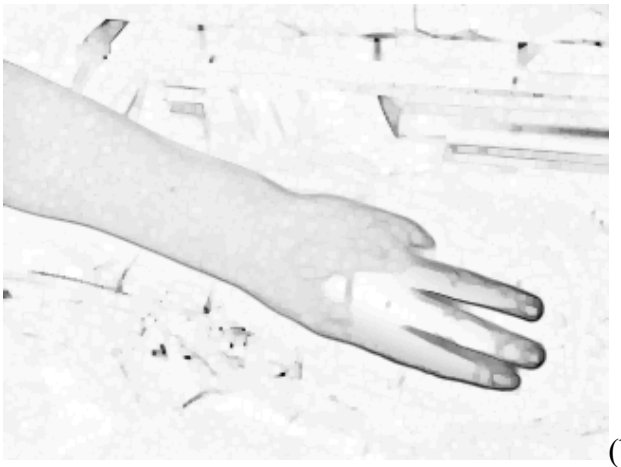
```
Linear SVC Training Time Used: 28.183756351470947
Linear SVC Accuracy on Validation Set: 0.9717153284671532
Linear SVC Predict Time Used: 0.6384809017181396
Randomforest Training Time Used: 6.433605194091797
RandomForestClassifier Accuracy on Validation Set: 0.885036496350365
RandomForestClassifier Predict Time Used: 0.28779029846191406
```

Fig.4 Linear SVM classifier vs. random forest classifier.

3.3.2 Q2

部分测试结果如下所示，(a),(b),(c)分别表示原图像，unet 输入图像，unet 输出图像。

(a)



(b)

(c)

3.4 算法分析

3.4.1 Q1

实验过程中，我们测试了很多不同的预处理方法和机器学习方法，有些计算量太高无法得到模型（如高斯核 SVM 分类器训练了一整天也无法得到结果），而能得到结果的模型在验证集上的准确率都在 90%左右。采用最大下采样+HOG 特征描述子+线性 SVM 的方法，一方面减少了训练时间，另一方面在验证集上也有着不错的准确率。

同时，我们也注意到当把训练出来的 SVM 分类器应用于测试数据集时，却无法达到和在验证集上一样高的准确率。也就是说，我们的模型可能产生了**过拟合**。为了验证这个想法，我们再次训练模型，这次用 70%的数据用于训练集，30%的数据用于验证集，仍然得到了 96.7%的准确率。所以我们推测，如果模型真的过拟合，很有可能是因为整个数据集都有某种特殊的性质，而非我们划分训练集-验证集的比例不恰当。

因此，在进一步的研究中，我们需要搞清楚是什么原因导致模型在验证集和测试集上的性能差异，并针对这一点进行改进。

3.4.2 Q2

实验过程中，我们输入 Unet 的训练集全都是包含人手的图像，这可能导致模型“认知”图像时有一定偏差：每个图像都必定有手。Unet 只学到了什么是手，而没有学到什么不是手。因此，当我们输入没有手的图像时，Unet 仍然有可能将图像的某部分标记为手，在后续的研究中有必要改正这一点。

此外，我们的设备相当受限制（无 GPU），用于训练的数据集也相对较小（一千多张图片），训练回合数也相对较少(15epoch)，因此在进一步的研究中，我们可以考虑将程序部署在集群上，并用图像生成器生成新的图像进行训练，以充分发挥算法的性能。

4．结论

本次实验中我们使用的方法都是基于已有的算法，在其上进行简单的组合以适应问题的需要。

用机器学习方法处理某个问题时，要关注的不仅仅是数据本身，更要关注数据的处理方式，这也是本次实验中我们主要的工作量。合适的数据处理方法不仅能加快训练速度，还能得到比较好的训练结果。在本次实验中，我们测试了同态滤波、拉普拉斯锐化、图像均衡化、同态滤波、形态学处理等等多种预处理方式，最后才得出几个比较适合这个数据集的处理方法。

通过这次实验，我们可以感受到，机器学习方法或者深度学习并不是万能的，这些方法如果直接用在原始数据集上，得到的效果可能很差。而将传统数字图像处理方法和机器学习结合，可以得到更好的效果。

主要参考文献(三五个即可)

[1] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//international Conference on computer vision & Pattern Recognition (CVPR'05). IEEE Computer Society, 2005, 1: 886--893.

[2] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.