问题探究

基于传统方法的人脸识别

陈璟洲

在利用Opencv库实现人脸识别后（直接当黑箱来用，容易导致使用不正确，出问题也不易分析），在实地测试时发现效果不好，是方法本身的问题还是自己实现的问题？

首先剖析自己的实现。自己写的程序主要包含3个部分：

1. 人脸检测
2. 人脸识别
3. 预处理

人脸检测

首先是人脸检测，因为程序在执行时对人脸检测效果良好，所以可以确定这一部分没有问题。

人脸识别

这是与采用识别算法直接相关的模块，因为是采用Opencv自带的库实现的识别算法，所以自己这里可能出现的问题即是调用和参数设置是否正确。

验证自己问题的直接方法即是设置相同的实验的条件，与别人的结果进行参照。最开始的思路是依照CVPR 2013年的一篇文章，在LFW数据库上进行验证。由于LFW数据库存在很多测试标准，因此花了一些时间阅读LFW技术报告来了解这些测试标准。通过阅读技术报告了解了一些人脸识别所面对的一些问题，以及如何设计和建立一个可供大家比较的数据库及其对应的测试标准，以及考虑如何在建立数据库时反映人脸识别问题在真实场景下所面临的问题。但是在利用测试标准进行测试时，考虑到需要编写包括预处理、测试代码等，出于时间的考虑，转换思路，LFW是相对一个比较新的数据库，能否在传统的数据库上，采用比较简单的测试标准进行测试呢？

由于Opencv库是由大神[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html) 实现的，既然要验证自己是否调用正确，最直接的方式莫过于和他自己进行的实验进行比较。[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html) 在AT&T(ORL)数据库和Yale Facedatabase A数据库上对Eigenfaces和Fisherfaces方法进行了测试，由于LBPH的调用方式和Eigenfaces、Fisherfaces方法相同，因此我们采用这两种方法和[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html) 的结果进行比较。

* **AT&T**

**AT&T**是比较简单的一个数据库，没有太多的光照变化，表情和面部姿态有略微地变化，包含40人，每人10张图片，尺寸为的灰度图片。

利用欧几里得距离计算降维后的样本间的距离，采用最近邻匹配的方式进行识别。对数据库采用10折交叉验证的方式进行测试。

* + **Eigenfaces**

降维维数设定为30。我的结果：98.25%，[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html)的结果为96.25%。

参考地址：<https://bytefish.de/blog/eigenfaces/>

注：[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html)的实验中采用了LeaveOneOut的交叉验证方式，因此结果有些许不同。

* + **Fisherfaces**

降维维数设定为39。我的结果：95.25%，[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html)的结果为95.5%。

参考地址：<https://github.com/bytefish/facerec>

* + **LBPH**

圆形LPB算子的半径为1，采样点为8个，将图片划分为的窗格进行采样（默认设置）。降维后的样本间利用不带权重的卡方距离计算距离，不采用等价模式直方图，因此每个区域直方图具有256个bins，64个区域总共的特征维数为16384。我的结果为：97%。

* **Yale Facedatabase A**

**Yale Facedatabase A**较**AT&T**在光照方面多了些变化，同样包括是否戴眼镜，但是多了戴墨镜的图片，除此之外在面部表情上也较**AT&T**更丰富一些（闭眼张嘴等）等。总共包含15人，每人11张图片，尺寸为的灰度图片。

面部姿态方面，不同于**AT&T**的面部轻微转动，作者依据双眼的位置将**Yale Facedatabase A**数据库进行了人脸矫正，并依据双眼的位置对人脸图片进行裁剪，裁剪后的尺寸为。

由于原始的**Yale Facedatabase A**数据库所在官网无法访问，无法获取原版的图片进行同样的矫正裁剪，我的实验中采用的是一套经过类似矫正（矫正后的图像相似度很高，但是裁剪后的尺寸有所不同），裁剪为大小的一套数据。

利用欧几里得距离计算降维后的样本间的距离，采用最近邻匹配的方式进行识别。对数据库采用5折交叉验证的方式进行测试。

* + **Eigenfaces**

降维维数设定为50。我的结果：83.33%，[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html)的结果为80.93%。

为了和之前**AT&T**的结果作以比较，降维维数设定为30，得到结果为83.33%。

参考地址：https://bytefish.de/blog/fisherfaces/

* + **Fisherfaces**

降维维数设定为14。我的结果：90%，[Philipp Wagner](https://bytefish.de/author/philipp-wagner.html)的结果为96.80%。

参考地址：<https://bytefish.de/blog/fisherfaces/>

* + **LBPH**

圆形LPB算子的半径为1，采样点为8个，将图片划分为的窗格进行采样（默认设置）。降维后的样本间利用不带权重的卡方距离计算距离，不采用等价模式直方图，因此每个区域直方图具有256个bins，64个区域总共的特征维数为16384。我的结果为：97.33%。

Eigenfaces方法是从基于重构的角度出发，容易受光照变化的影响，导致结果下降；Fisherfaces则是从分类角度的出发，某种程度上学习不同类别间人脸的面部特征，较Eigenfaces受光照影响小一些，但因此也丢失了重构所需要的一些信息；LBPH学习到人脸的局部特征，可以应付光照的单调变化，对光照具有一定的鲁棒性，在两个数据库上的结果整体最优。

**结论：人脸识别部分的对于Opencv库中实现的调用是正确的。**

让我们来进一步探究一下光照变化对这三种算法的影响。

* **The extended Yale B**

**The extended Yale B**是经过裁剪和矫正，仅包含人脸正面面部区域的数据库，面部没有眼镜墨镜等物体，也没有太多的表情变化，为同一个人同时在不同角度的光照下拍摄得到的图片。总共包含38人，每人68张图片，尺寸为的灰度图片。

从数据库中每人随机选取10张图片作为训练集，每人随机选取28张图片作为测试集（两个集合没有交集），距离计算方式和分类器同上。Eigenfaces降维维数设定为30，Fisherfaces取为37，LBPH的半径为1，采样点为8，窗格区域划分为。

* + Eigenfaces: 39.66%
  + Fisherfaces: 61.37%
  + LBPH: 53.1%

我们来改变一下参数，看看对识别效果的影响。

* + Eigenfaces
    - 维度设为20: 35.90%
    - 维度设为50: 43.42%
    - 维度设为70: 44.45%
  + Fisherfaces
    - 维度设为27: 51.97%
    - 维度设为57: 61.37%
    - 维度设为77: 61.37%
  + LBPH
    - 半径为1，采样点为8，的区域：43.14%
    - 半径为1，采样点为8，的区域：60.15%
    - 半径为2，采样点为8，的区域：58.08%
    - 半径为1，采样点为16，的区域：50.09%

通过参数的变化，识别效果有5%-7%的提升。

**The extended Yale B**这套数据库唯一的变量即是光照（人脸矫正良好，没有表情和眼镜等的变化），光照在各个角度变化剧烈时（有些面部区域甚至全黑），三种算法的效果均有较大下降，相较于Eigenfaces的结果，Fisherfaces和LBPH对光照要更加鲁棒一些。另一方面，LBPH算法也难以应对光照的不规则变化。

# 预处理

预处理的作用是提高人脸识别的效果，参考别人的实现，我的程序中包括了人脸矫正、灰度化、尺度归一化、直方图均衡化四部分预处理。在程序执行的过程中，实时显示了人脸矫正的效果是正确的，因此这部分的实现也是正确的。灰度化、尺度归一化、直方图均衡化仅为三行函数语句。因此，预处理这部分的实现是正确的。

我在这里进一步探讨一下预处理对识别效果的影响。因为在**AT&T**数据库上实现的精度很高，因此我选择在**Yale Facedatabase A**和**The extended Yale B**上进行实验。由于这两个数据均为灰度图片，而且经过了较好的人脸矫正，因此我的探究关注在尺度和直方图均衡化两个因素。

实验设置：距离计算方式和分类器同上，采用5折交叉验证，参数方面，Eigenfaces设置为30，Fisherfaces以及LBPH均对应默认设置。

* **尺度变化**
  + **Yale Facedatabase A**

|  |  |  |  |
| --- | --- | --- | --- |
| **尺寸** |  |  |  |
| **Eigenfaces** | **83.33** | **83.33** | **83.33** |
| **Fisherfaces** | **90** | **92** | **91.33** |
| **LBPH** | **97.33** | **84.67** | **91.33** |

* + **The extended Yale B**

|  |  |  |  |
| --- | --- | --- | --- |
| **尺寸** |  |  |  |
| **Eigenfaces** | **39.66** | **39.57** | **39.57** |
| **Fisherfaces** | **61.37** | **61.28** | **60.43** |
| **LBPH** | **53.1** | **50.56** | **45.02** |

这样的结果是比较容易理解的，Eigenfaces和Fisherfaces是基于整体的方法，对尺度变化不敏感，因此结果变化不大；LBPH则是基于局部邻域的特征提取，受尺度变化的影响较大。

* **直方图均衡化**

我们在数据库原始尺寸上进行比较。

* + **Yale Facedatabase A**

|  |  |  |
| --- | --- | --- |
| **是否直方图均衡化** | **是** | **否** |
| **Eigenfaces** | **88** | **83.33** |
| **Fisherfaces** | **94.67** | **90** |
| **LBPH** | **96.67** | **97.33** |

* + **The extended Yale B**

|  |  |  |
| --- | --- | --- |
| **是否直方图均衡化** | **是** | **否** |
| **Eigenfaces** | **62.03** | **39.66** |
| **Fisherfaces** | **97.74** | **61.37** |
| **LBPH** | **52.73** | **53.1** |

直方图均衡化依据直方图累计函数，将原先图像像素分布比较集中的图像变换到一个更大区间上的均匀分布，从而增强图像的对比度。直方图均衡化对于基于整体的方法（Eigenfaces以及Fisherfaces）提升较大，但是对于LBPH这种依据中心像素点与邻域像素点进行比较二值化的局部特征提取方式却没有太多变化。

性能评估

以上实验仅是我们在传统数据库上所做的实验，这些数据库年代比较久远，相对来说是在可控的单一环境中，经过良好的人脸矫正处理，控制其他因素而研究某一因素变化（例如光照）的比较简单的数据库。然而实际环境要更加复杂，同时良好地预处理也能帮助识别效果的提升，所以一个更直接的问题是评估传统方法的性能。可以通过阅读其他人的文献，参考其他人在不同数据库上利用传统方法所做的实验，来大致获得一个对传统方法的评估。

以下我们参考的文章中子空间特征提取算法、分类器选取和以上我们的实验设置大体相同，在参数设置上有区别。

* Ahonen T, Hadid A, Pietikainen M, et al. Face Description with Local Binary Patterns: Application to Face Recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(12): 2037-2041.

这篇文章利用LBPH算法在FERET数据库上进行实验，在光照变化较多的实验组上仅有60%多的识别率。

* Huang G B, Mattar M, Berg T, et al. Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments[J]. Month, 2007.

这篇文章利用Eigenfaces算法，在没有经过人脸矫正的LFW数据库上的识别率仅有12%。

* Martinez A M, Kak A C. PCA versus LDA[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 2001, 23(2):228-233.

文章在AR数据库上首先研究对比来LDA和PCA算法在小样本上的表现（训练样本：2张/人，测试样本：5张／人），PCA和LDA表现在50%左右的识别率；随后进一步扩大样本量，PCA的效果达到65%，LDA最高可以达到85%。

* Tan X, Triggs B. Enhanced local texture feature sets for face recognition under difficult lighting conditions.[J]. IEEE Transactions on Image Processing, 2010, 19(6):1635.

文章分析来在严苛光照条件下对LBPH等算法的影响，在Extended Yale B以及CAS-PEAL-R1数据库上进行实验，原始LBPH算法在Extended Yale B上的识别率可以到80%，而在CAS-PEAL-R1上仅有20%。

* Yang B, Chen S. A comparative study on local binary pattern (LBP) based face recognition: LBP histogram versus LBP image[J]. Neurocomputing, 2013, 120(10):365-379.

文章在数据库FERET、Extended Yale B、CMU PIE和AR上对LBPH、Eigenfaces和Fisherfaces算法进行了比较分析。文章首先每人选取一张图片作为训练样本，测试结果三种算法的识别率均为10%左右；然后扩大训练集数量进行实验，我们关注和我们实验中训练集样本量（10张／人）接近的实验结果，LBPH和Eigenfaces为30%到40%，Fisherfaces为70%到80%。

综合分析

首先确定人脸检测部分没有问题后，验证了人脸识别部分使用的正确性，然后探讨了一下预处理对人脸识别效果的影响。在预处理的过程中，我们对训练集和测试集采用同样的处理标准：灰度化、尺度归一化、直方图均衡化以及人脸矫正。

在实际进行测试时，首先是训练库中为我在寝室环境下拍摄的12张图片，然后立即当场进行一对一测试（此时光照背景变化都小），三种算法的效果都不错，能正确识别我拒识其他人；在隔一天之后，到实验室打开摄像头实时测试（考虑到实际应用场景，这种不同地点采集测试的情况是可能的），选取效果较好的LBPH和Fisherfaces算法对我和另外两名同学（简称为A和B同学）进行一对一测试（即是否识别我）：

Fisherfaces：正确识别我，拒识A同学；正确识别我，误识B同学；

LBPH：正确识别我，误识A同学；正确识别我，误识B同学。

实验中训练集和测试集背景变化大，不同于数据库中在单一可控环境下拍摄的图片；采取的预处理效果有限，不同于数据库中经过良好预处理的图片；在实地测试时，我们训练集中有12张图片／人，但是视频一秒读入28-30张图片，测试集数量较多，而论文中在数据库上实验时固定训练集和测试集（样本量的确对识别效果有影响）。论文中三种算法效果不尽如人意，可以预见，实验的结果相比于论文会更差一些。

接下来参阅文献，参考更好的算法和预处理进行系统改进。另外，上述实验中实现的是一个识别问题（Identification），相对于人脸验证（Verification）来说是一个更难的问题，之后可以考虑将系统设计为一个人脸验证系统。