

Eigenfaces 和 Fisherfaces 的比较与改进

PB17111561-周恩帅 PB17111568-郭雨轩

2020 年 5 月 21 日

摘要

本文给出了两种基于降维的人脸识别算法 Eigenfaces 和 Fisherfaces 的推导过程，同时对两种方法的进行了对比和改进。

对于 Eigenfaces 算法，本文尝试从数据预处理、类内特征聚合和衡量类间差异的 Metric 三个维度对原有方法进行了改进，并在 Yale Faces B 数据集上定量的讨论这些改进各自产生的影响和联合产生的影响。我们发现在单独应用三种方法时，均可以使得算法精度提升；在联合使用时，也可以使得算法精度有较大提升。

本论文的 latex 代码和实验用到的代码见此[github](#)仓库。

目录

1	人脸识别算法的背景介绍	2
2	Eigenfaces 和 Fisherfaces 算法数学推导	2
2.1	Eigenfaces 数学推导	2
2.1.1	Eigenfaces 训练部分数学推导	2
2.1.2	Eigenfaces 推断部分数学推导	3
2.1.3	Eigenfaces 数学推导总结	4
2.2	Fisherfaces 数学推导	4
3	两种算法的对比与改进措施	4
3.1	两种算法的对比	4
3.2	对 Eigenfaces 的改进	4
3.3	对 Fisherfaces 的改进	5
4	实验部分	5
4.1	Eigenface 实验	5
4.2	Fisherfaces 实验	8
5	结论	8
	参考文献	9
A	附录	10

1 人脸识别算法的背景介绍

在这两种算法出现之前，大部分的人脸识别算法主要关注于识别特定的、单一的人脸特征，例如眼睛、鼻子、嘴或者脸的外轮廓。这些方法通过对上述特征出现的位置、形状大小以及相互的关系进行定义，最终得到一个人脸检测模型。但是它们有着非常明显的缺点，这些模型非常的脆弱，难以推广到多人脸、多视角的应用场景上去。

在 1987 年，Sirovich 和 Kirby 两人首次在他们的论文 [2] 中证明了主成分分析（PCA）可以被用在图像集合上并得到一系列的“特征脸”，并且图像集合中的每一张脸都可以由这些“特征脸”线性组合重建出来，重建的误差会随着使用的“特征脸”的数目的增加而减少。

Turk 和 A. Pentland 于 1991 年在他们的论文 [3] 中给出了使用“特征脸”进行人脸识别的方法，这也是今天要介绍的 Eigenfaces 算法。在论文中，他们将人脸集合看成是一个由向量构成的集合，使用 PCA 的方法对人脸降维，再使用得到的“特征脸”计算出每张人脸的一个低维表示，使用这些低维表示，实现对面脸集合的分类。同时他们还介绍了一个快速求解维度很高的协方差矩阵的特征向量的方法，以便于快速得到人脸集合的“特征脸”。

2 Eigenfaces 和 Fisherfaces 算法数学推导

2.1 Eigenfaces 数学推导

2.1.1 Eigenfaces 训练部分数学推导

在进行数学推导之前，需要说明的是，我们的目的是使用 PCA 的方法对输入的人脸数据集合进行降维，对于每张人脸得到一个低维表示，再根据这个低维表示对人脸数据进行分类。首先，对于输入的人脸图像集合把图像组织成列向量形式：

$$\Gamma = \{\Gamma_1, \Gamma_2, \dots, \Gamma_N\}$$

之后，我们需要求出输入向量的平均值，即“平均脸”，可视化效果见附录图8：

$$\Psi = \frac{1}{N} \sum_{i=1}^N \Gamma_i$$

再根据“平均脸”求解出“误差脸”矩阵：

$$\Phi_i = \Gamma_i - \Psi, A = \{\Phi_1, \Phi_2, \dots, \Phi_N\}$$

得到“误差脸”矩阵后，计算出协方差矩阵：

$$C = AA^T$$

该矩阵的 K 个特征向量即为所求的“特征脸”。为了求解这个矩阵的特征值和特征向量，我们需要对其使用 SVD 分解。值得注意的是，假定每张人脸图像的维度是 $[w, h]$ ，那么每个人脸向量的维度也就是 $[w \times h, 1]$ ，则矩阵 C 的维度是 $[w \times h, w \times h]$ 。在通常的图片中， $w \times h$ 的数量级一般在万级别或者十万级别，而对如此之大的一个矩阵做 SVD 分解的时间复杂度是非常大的。为了解决这个问题，我们可以先对这样一个小矩阵做 SVD 分解：

$$C^* = A^T A$$

这个矩阵的维度为 $[N, N]$ ，其中 N 为人脸数据集的大小，在人脸识别的任务中，可以认为这个数量级是相对较小的。以一个典型的人脸数据集 **Yale Face B** [1] 为例， $N = 165$ ，而 $w = 320, h = 243$ ，维度上的差距可以见得。我们对 C^* 矩阵求出特征向量 \vec{x}_i ：

$$A^T A \vec{x}_i = \lambda_i \vec{x}_i$$

再对上式左乘一个矩阵 A ，适当使用结合律，我们可以得到：

$$(AA^T)(A\vec{x}_i) = C(A\vec{x}_i) = \lambda_i A\vec{x}_i$$

可以看出，我们希望求得的矩阵 C 的特征向量 $\vec{\mu}_i = A\vec{x}_i$ ，因此，我们可以先求出小矩阵 C^* 的特征向量 \vec{x}_i ，再对其特征向量左乘矩阵 A ，即得矩阵 C 的特征向量 $\vec{\mu}_i$ 。至此，我们已经求得所有的“特征脸” $U = \{\mu_1, \mu_2, \dots, \mu_K\}$ ，可视化效果见附录图9。值得注意的是，矩阵 C 并不满秩，因为显然矩阵 A 中的每个列向量满足关系式：

$$\sum_{i=1}^N \Phi_i = \sum_{i=1}^N (\Gamma_i - \Psi) = \sum_{i=1}^N \Gamma_i - N \cdot \frac{1}{N} \sum_{i=1}^N \Gamma_i = 0$$

即所有列向量是线性相关的。因此，矩阵 C 至多有 $K - 1$ 个有用的“特征脸”（对应的特征值不为 0），可视化效果见附录图10。那么，我们求得的这些“特征脸”是否可以作为“脸空间”的标准正交基呢？答案是肯定的，由于矩阵 $C = AA^T$ 是一个实对称矩阵，因而其所有的特征值都是正交的，我们只需要对求出的“特征脸”进行标准化，就可以得到一组“脸空间”的标准正交基，为了叙述方便，我们仍用 U 代指标准化后的“特征脸”。

在得到了一组“脸空间”的基向量，也即“特征脸”之后，我们需要计算每一张已知的脸在这组基下的坐标 $\Omega = [\omega_1, \omega_2, \dots, \omega_k]$ 。我们可以先考察对于一张特定的“误差脸” Φ_i 在一个特定的基 $\vec{\mu}_j$ 上的投影，这就是 ω_j^i ：

$$\omega_j^i = \Phi_i^T \vec{\mu}_j$$

那么一张脸的低维表示可以由以下公式给出：

$$\Omega_i = \Phi_i^T U = [\omega_1^i, \omega_2^i, \dots, \omega_k^i]$$

类似的，所有脸的低维表示可以由以下公式给出：

$$\Omega = A^T U$$

在实际操作中，对于一个类别的人脸，可能有多张人脸数据对应这一类，这时，这一类的低维表示被定义为：所有同属于这一类的人脸图像的低维表示的平均值。

至此，基于 PCA，我们已经对于每张输入的人脸数据都得到了一个低维表示，并且为每一类别生成了一个低维表示。

2.1.2 Eigenfaces 推断部分数学推导

这部分我将介绍如何使用训练部分生成低维表示对输入的图像进行人脸分类。对于每一张新输入的图片 Γ_{new} ，我们首先需要计算出它的“误差脸” Φ_{new} ：

$$\Phi_{new} = \Gamma_{new} - \Psi$$

之后，我们要计算出“误差脸”的低维表示 Ω_{new} ：

$$\Omega_{new} = \Phi_{new}^T U$$

由上节的叙述，我们可以根据这个低维表示重建出输入的人脸 Φ_f ：

$$\Phi_f = \Omega_{new} U^T$$

为了考察输入的图片是否为一张人脸图片，我们计算重建图像与原图像的误差 ϵ_1 ：

$$\epsilon_1 = \|\Phi_{new} - \Phi_f\|^2$$

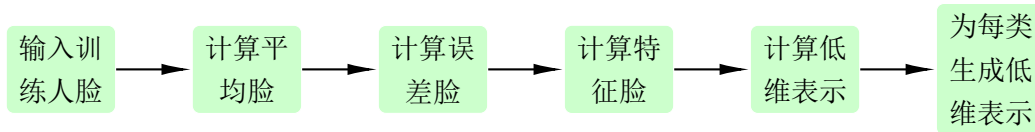
如果误差过大，则认为不是一张人脸。排除掉输入图片是否为一张人脸的因素后，我们再将这个输入生成的低维表示与已知的每一类的低维计算误差，对于第 k 类的误差为 ϵ_2^k ：

$$\epsilon_2^k = \|\Omega_{new} - \Omega_k\|^2$$

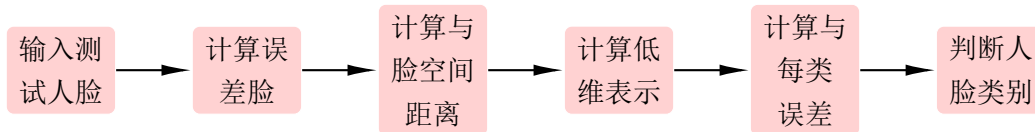
当输入图像与某类的误差最小，且误差不超过预设的阈值时，我们可认为输入的人脸属于这一类，否则输入的人脸属于新的一类。

2.1.3 Eigenfaces 数学推导总结

综上，Eigenfaces 的训练流程为：



Eigenfaces 的推断流程为：



2.2 Fisherfaces 数学推导

3 两种算法的对比与改进措施

3.1 两种算法的对比

3.2 对 Eigenfaces 的改进

对 Eigenfaces 的改进主要从以下三点出发，具体效果在实验部分展示：

- **数据预处理：**我们观察到数据集中同一类人脸的照片中出现了很多不同表情的照片，这些照片最主要的特点为人脸的上半部分基本相同，人脸的下半部分差异较大。因为对人脸分类不需要考虑其表情，所以我尝试去掉了下 1/3 人脸，仅保留剩余部分作为 Eigenfaces 的输入，数据可视化见附录图11。

- **为每类生成低维度表示：**论文中为了生成每一类的低维表示，选择将同属于一类的低维表示取了平均值进行聚合。但是我们认为，取平均值可能会导致某些照片检测出现误差，我们尝试了不做任何聚合和在低维表示上的每个维度去做 abs-max 。
- **计算与每一类的误差：**在论文中使用的是 $L_2 \text{ loss}$ （也即 MSE）作为评价的 metric，它对低维表示中的离群维度比较敏感。但是在这个场景中，可能仅仅需要要求大部分维度的低维表示接近时我们就可以断定输入图像属于这一类，基于此我们尝试了 $L_1 \text{ loss}$ （也即 MAE）作为评价的 metric；还有可能仅仅需要输入图像的低维表示的方向与对应类别的低维表示接近即可，基于这个想法我们也尝试了 cosine loss （也即余弦相似度）作为评价的 metric。

3.3 对 Fisherfaces 的改进

4 实验部分

我们使用 Yale Faces B [1] 数据集对两种方法及改进分别进行了测试，这个数据集包含 165 张人脸图片，一共有 15 个类别，每类有 11 张灰度图片，分辨率为 320×243 ，包含同一个人不同光照条件下和不同表情下的图片，部分数据集图片如下：



图 1: Yale Face B 数据集

4.1 Eigenface 实验

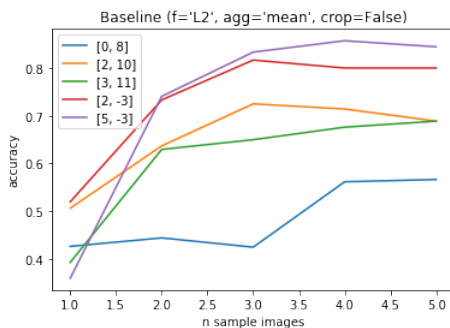


图 2: 基准模型

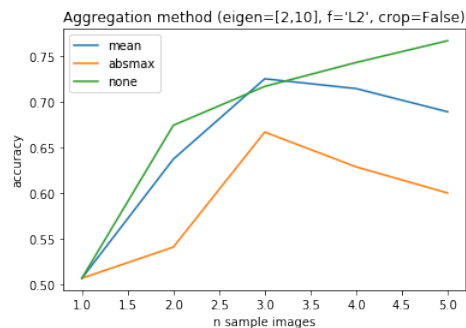


图 3: 改变特征聚合方式的模型

首先我们测试了在不同采样率和不同的“特征脸”选取策略下的精度，见图2，这张图的横坐标为每种类采样的图片数量，纵坐标为准确率，不同的曲线为不同的特征向量的选取策略。例如 $[0, 8]$ 表示选取第 1 大到第 8 大的特征向量作为“脸空间”的基， $[2, -3]$ 代表选取第 3 大到倒数第 3 大的特征向量作为“脸空间”的基。图中可以明显的看出，当使用了特征值比较大的

特征向量时，精度反而较低，根据 Eigenfaces 论文 [3] 论文中的说法，Eigenfaces 是一个对光照敏感的模型，而特征值较大的特征向量会包含光照的信息，导致结果准确率较低。我们也可以看到当“脸空间”的特征向量的数目增加时，精确率会有明显的上升，这是因为越多的特征向量会有更好的重建性能，但是同时也会导致更大的计算开销。在我们后续的对比试验中，我们固定特征向量选取策略为 [2, 10]，这是一个有着不错效果，同时又不会导致很大的计算开销的策略，来横向对比不同的改进措施产生的影响。

然后我们测试了在不同采样率和不同的类内特征聚合方式的模型，它与基准模型仅在聚合的方式上有所不同，见图3。可以看出，在不同采样率上，不聚合的方式较于基准模型有着更优的精度，二者都好于使用 abs-max 的方式。这可能是因为，对同属一类的低维表示使用平均值可能会导致信息损失，不进行聚合可以保证更好的多样性，但是也会带来更大的存储开销；abs-max 的方式效果较差，可能是因为在使用 L_2 loss 衡量输入与每一类误差的时候会导致得到的结果与预期偏差更大，从而使得精度下降。

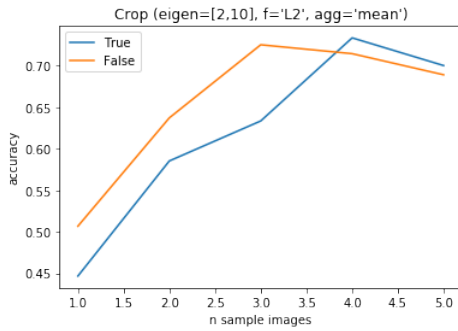


图 4: 对输入数据预处理的模型

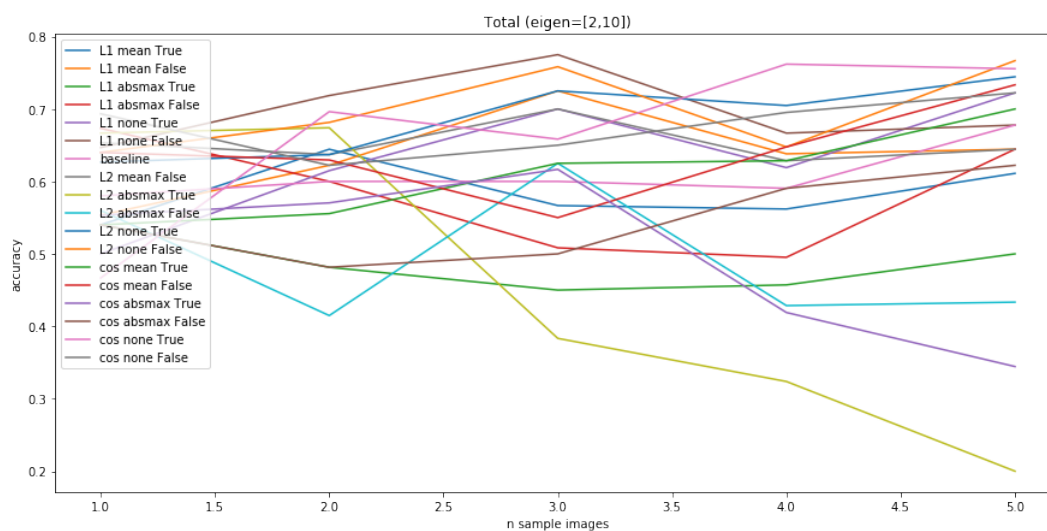
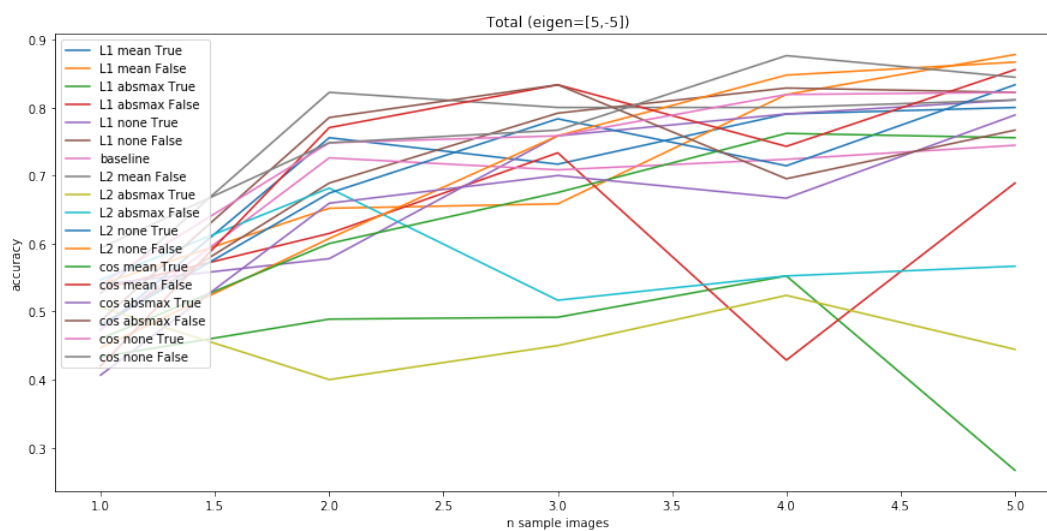


图 5: 改变损失指标的模型

接下来我们测试了仅进行预处理（对图片裁切到下 1/3）带来的影响，见图4。可以看到，当采样率较大的时候，进行预处理的效果要比不进行预处理的效果要好，这可能是因为当采样率增大，加上使用平均作为类内特征聚合的方式，会导致训练集中差异较大的部分被平均化，导致测试集中的每张图片都与正确类别的低维表示差距较大，而进行裁切后，去掉了差异化较大的部分，仅保留了共性的部分，因此图像更容易被判别到正确的类别。

之后我们测试了仅改变衡量类归属的指标带来的影响，见图5。基准模型使用的是 L_2 loss，当采样率增大的时候 L_1 loss 和 \cosine loss 的效果均高于基准模型，这可能是因为， L_2 loss 更加关注低维表示中的离群维度，而随着采样率的增大，训练集数据间的差异也在增大，导致即使属于同一类的低维表示，在这个向量的某一维也可能有较大的差距，而使用 L_2 loss 则放大了这个差距，导致精度下降，另外两种没有这种情况。

最后我们将测试了三种方式联合使用时产生的影响，见图6和图7。两图的主要差别在特征向量的选取策略不同，前者可以认为是在受限的特征向量的选取场景下，后者可认为是在理想情况下。可以看到当联合使用三种改进方式时，有部分排列组合相较于基准模型有了较大的提升，说明所提出的三种模型在改进的方向上是相对较为独立的，都各自解决了部分 Eigenfaces 模型原有的一些不足。

图 6: 选取 $[2, 10]$ 特征向量, 三种策略联合图 7: 选取 $[5, -5]$ 特征向量, 三种策略联合

4.2 Fisherfaces 实验

5 结论

对于两种算法，本文分别给出了它们的数学推导对于 Eigenfaces 算法，本文从三个维度尝试对其改进，独立应用时均有小幅度精度提升，在联合使用时也有明显的精度提升。

参考文献

- [1] A.S. Georgiades, P.N. Belhumeur, and D.J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6):643–660, 2001.
- [2] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A*, 4(3):519–524, Mar 1987.
- [3] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991. PMID: 23964806.

A 附录

在这部分我们给出 Eigenface 的部分中间结果的可视化效果。

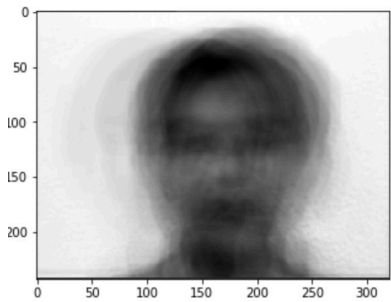


图 8: 平均脸

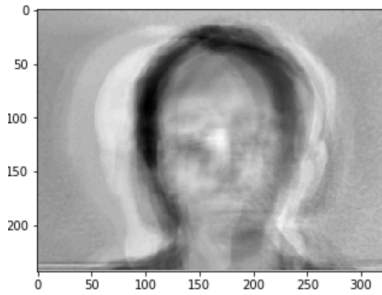


图 9: 特征脸

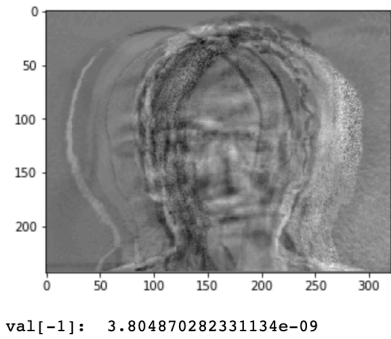


图 10: 最后一个特征脸及其对应的特征值

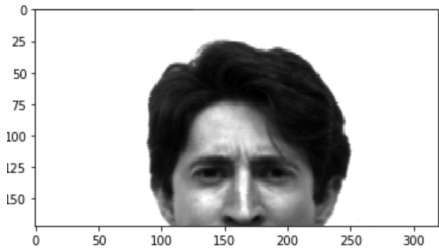


图 11: 裁切脸