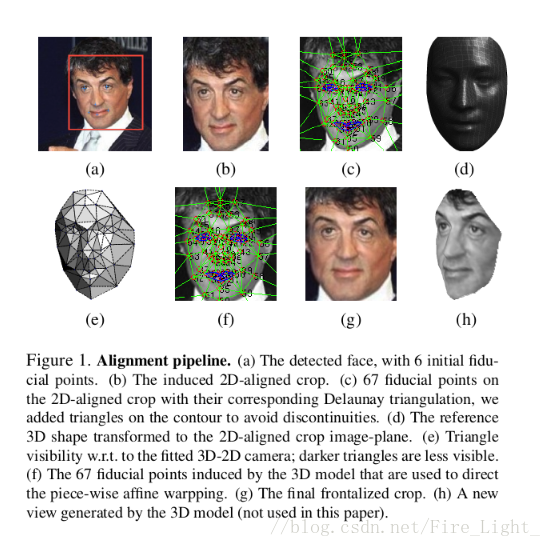
# Face

# Survey

2014-2018

# Deepface

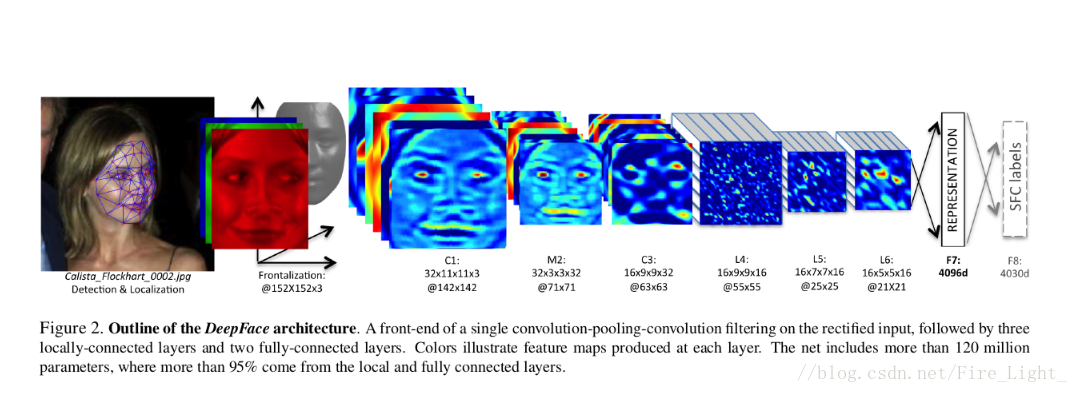
## 3D人脸对齐



## Delaunay三角剖分

对数据处理,视觉分析有帮助.

## Deepface网络结构



Heatpoint#1 **由于对齐好的图像，不同的区域具有不同的局部统计量，卷积的空间平稳性假设不能成立，因此不同的区域需要使用不同的卷积核**

Heatpoint#2 **全连接层，这些层能够捕捉人脸图像中较远部分的特征之间的相关性，比如眼睛的形状与位置和嘴巴的形状与位置**

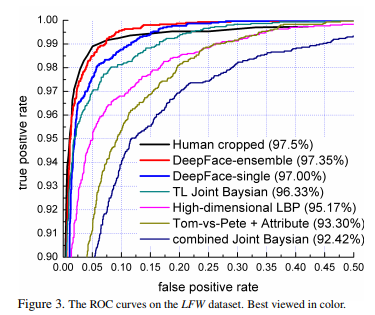
## 特征标准化

1. 特征的每个维度都会除以它在训练集中的最大值
2. L2范数归一化

## 验证方法

1. 简单的求两个特征向量的内积。经过标准化之后，内积越大越相似
   1. 类似余弦距离
2. 在LFW上训练一个感知机，即Siamese network，类似一个全连接层，以全连接层提取出来的特征作比较，计算两张图片距离的方式
   1. 也是特征,求距离.
   2. 只不过这个特征是再训练一个感知机(Siamese network)得到的.

## 实验结果



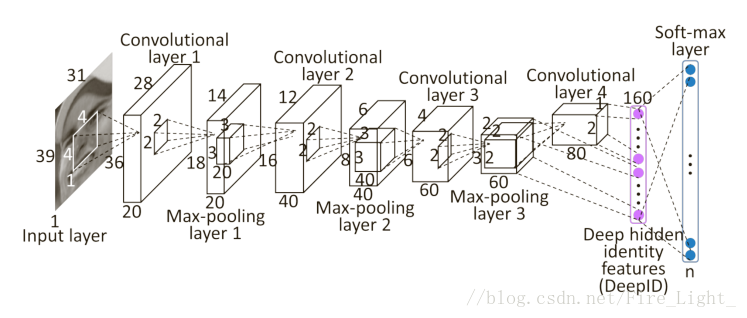
97%@fpr0.025@lfw

# DeepID1

## 网络结构

**DeepID特征就是一个由连接第三层与第四层组成的全连接层特征**

**Heatpoint#3 第四层的特征更加全局化（global），第三层的特征更加细节.因此DeepID连接了两者，以求同时包含全局，细节信息。**

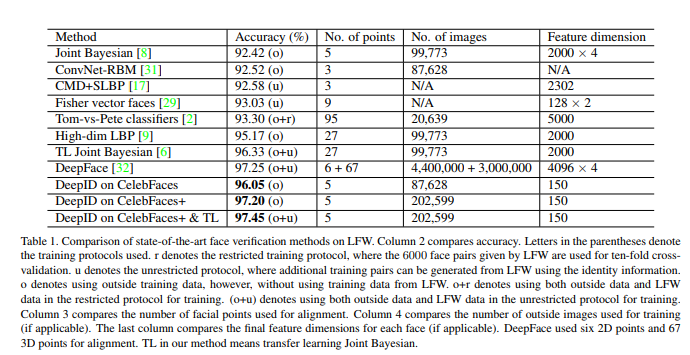


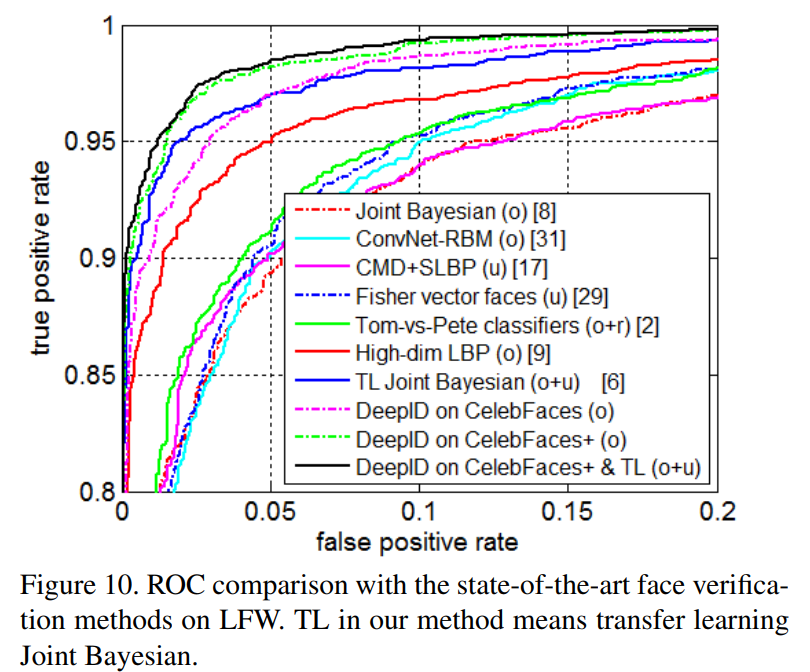
## 网络训练



1. 通过5个landmarks将每张人脸划分成10regions
2. 每张图片提取60patches=10regions\*3scales\*2(RGB or gray)

## 实验结果





# DeepID2

## 文章着力点

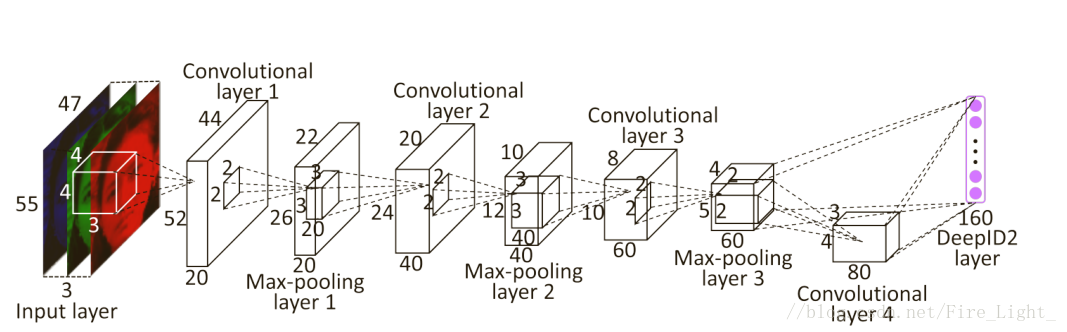
文章着力在扩大类间距离和缩小类内距离

原来的identification signal是处理两个向量距离.

1. 这两个向量(DeepID2)都提取于(基于)相同的向量.
2. dissimilar DeepID2特征能够通过一个映射函数g(),映射到相同的实例上(identity).
3. 这就有一个问题, 对于新的任务(tesks)和新实例(identities),g()如果失效了,会导致偏差很大.

为了解决这个问题, 引入了一个额外的face verification signal来确保”类间距离大, 类内距离小”

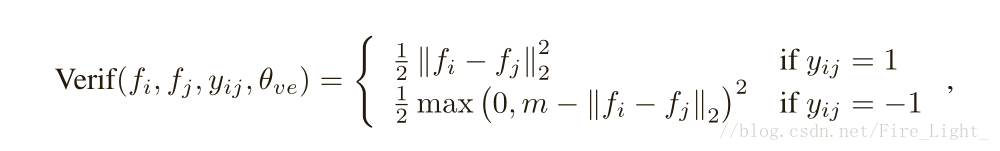
## 网络结构



## 创新点

多损失函数引入.

1. 分类信号,交叉熵
2. 验证信号(contrastive loss)

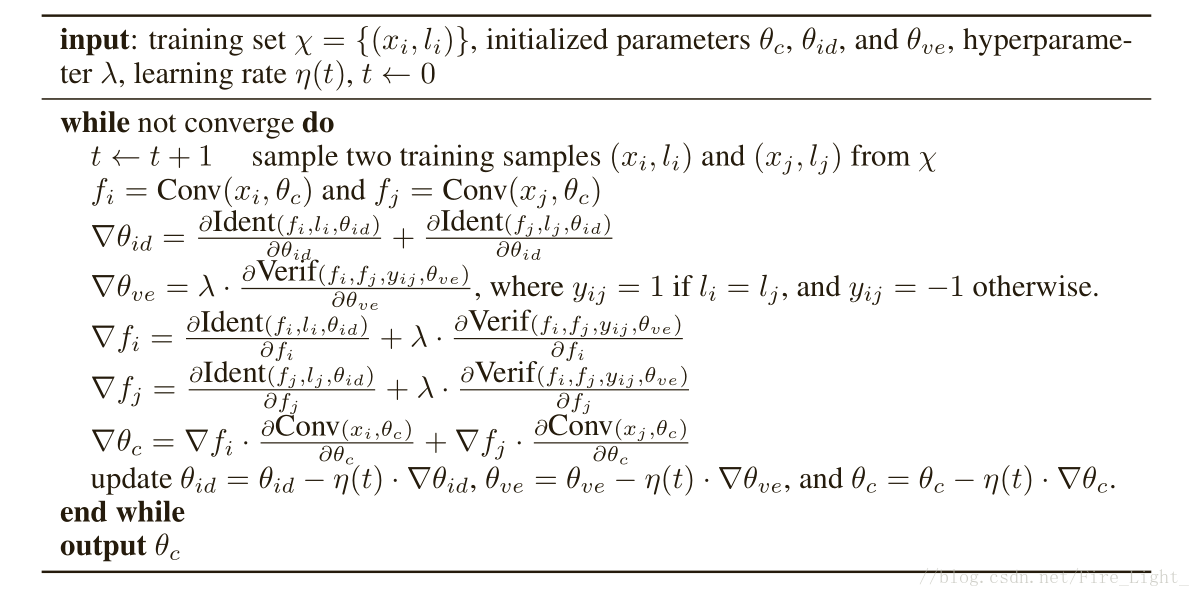


Contrastive loss如下:

https://img-blog.csdn.net/20180214171635734?

contrastive loss,可以有效处理孪生网络paired data关系(什么关系?)

## 训练过程



其中m不参于训练, 过程简单概括为

1.一次使用2个输入，计算了他们的L(Ident)和L(Verif),总损失L为二者通过λ加权求和

2.通过L来执行梯度下降更新卷积参数

3.通过L(Ident)来更新softmax层的参数

## 多Patches操作

每张图片使用了21 facial landmarks

分成200patches（20regions\*5scales\*2RGB&Gray)，水平翻转后变为400patches

使用了200个卷积神经网络，提取400（200\*2）个Deepid2特征

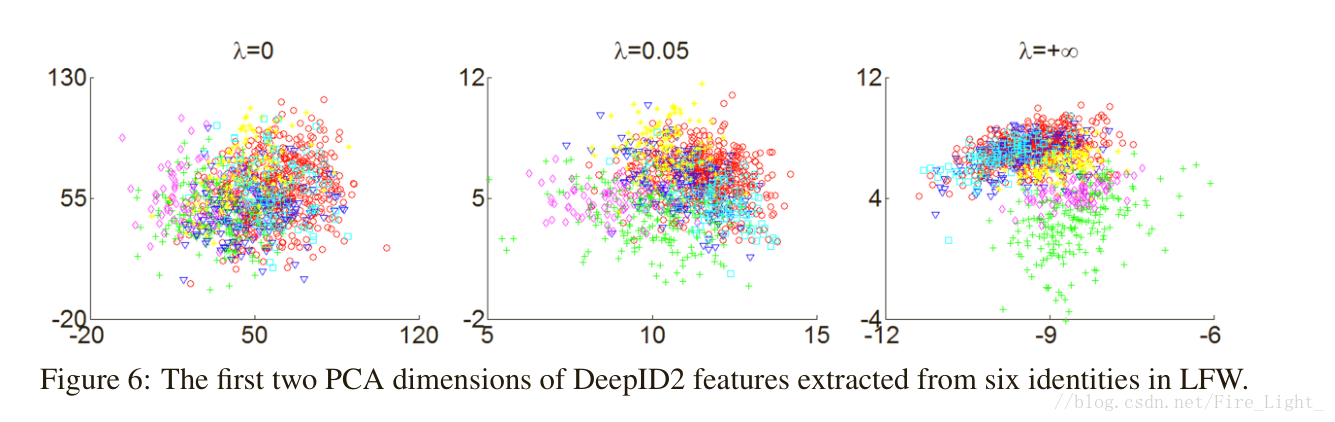
（一个神经网络对应的是一个patch与它的翻转对应的Patch）

使用 Adaptive forward-backward greedy algorithm降为25个Deepid2特征

使用PCA将25\*160Deepid2特征降为180维

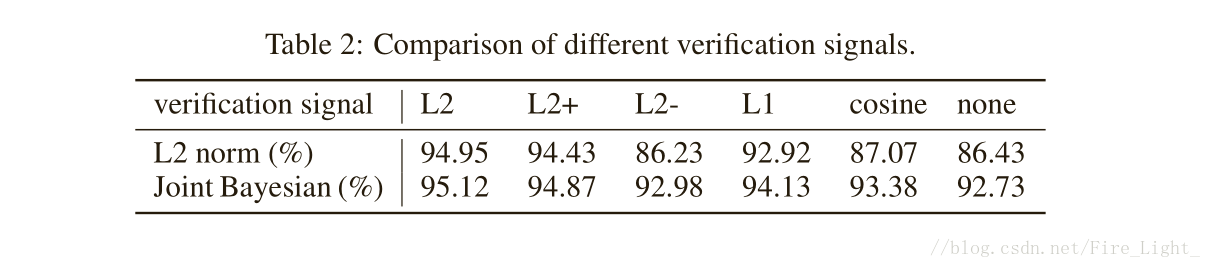
输入联合贝叶斯算法中，进行验证。

## 实验结果



在celebrate+上训练, 在LFW上进行验证。

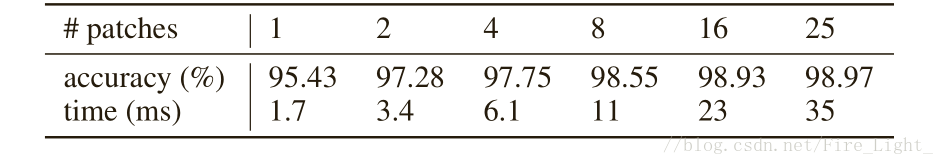
λ对类间与类内特征的影响.



验证信号(verification signal) 可以采用两种算法:

1. L2范数.
2. 联合贝叶斯, 联合贝叶斯效果稍好.

**选择效果最好的25个patches对应的25个网络**，在LFW上得到的最终准确率是98.97%

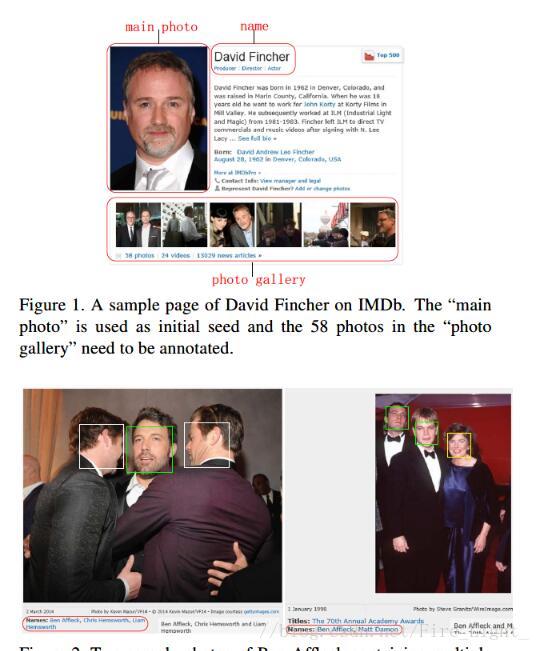


之后，重复使用贪婪的方式寻找其他6组patches（每组25个），得到7组25Paches,用联合贝叶斯算法计算相应的神经网络输出特征可以得到7个得分，使用SVM融合这7个得分来得到最终的准确率是99.15%

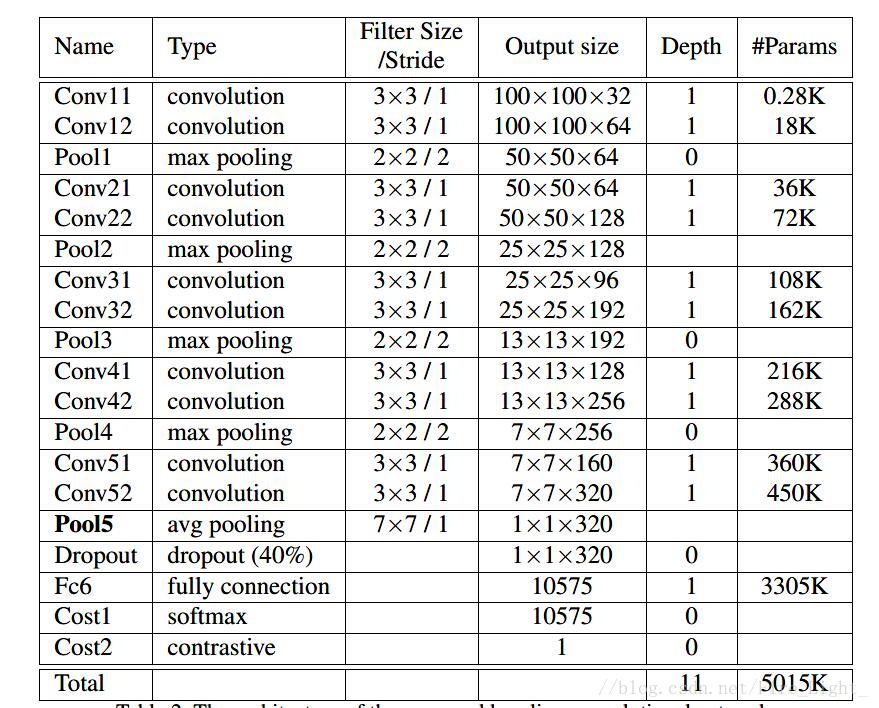
（论文没有提到怎么融合的，猜测具体的融合方法大概是构建一个7维的向量，然后根据该得分向量来分类）

# ##Webface系列1

## Webface数据集半自动生成



## 网络结构



类似(借鉴于)Vgg,inception.

一些技巧的解读：

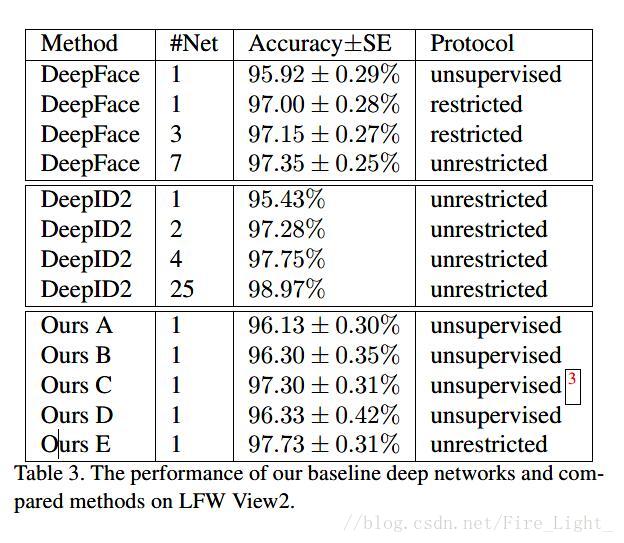
1.使用多个3x3的卷积核代替大卷积核

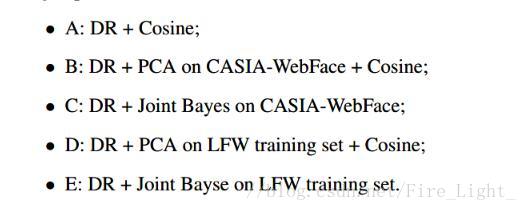
2.只使用一个全连接层，减少参数量

3.使用两种监督信号

可以看出作者在网络构筑上没有提出新的思想，而是借鉴了Inception以及DeepId网络的优秀结构。

## 实验结果





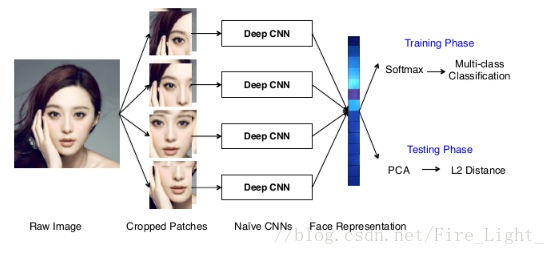
DR是论文的网络,在CASIA上train的.

# ##Face++

## 对现阶段人脸的总结20 Jan 2015

HeatPoint#4**算法的影响在数据量提升中被淡化**

作者同样在MFC上尝试了一系列复杂的算法，如联合贝叶斯算法，聚类算法，以及添加类似deepid网络的分类与验证信号，结果发现这些算法对网络准确率的提高微乎其微，远远比不上数据量的提升来的大。



每张图片切分成4个patch，训练时使用Softmax损失，验证时使用PCA降维后直接计算L2距离，最终在LFW上达到了99:50%的准确率，基本是达到了人类的识别能力的水平

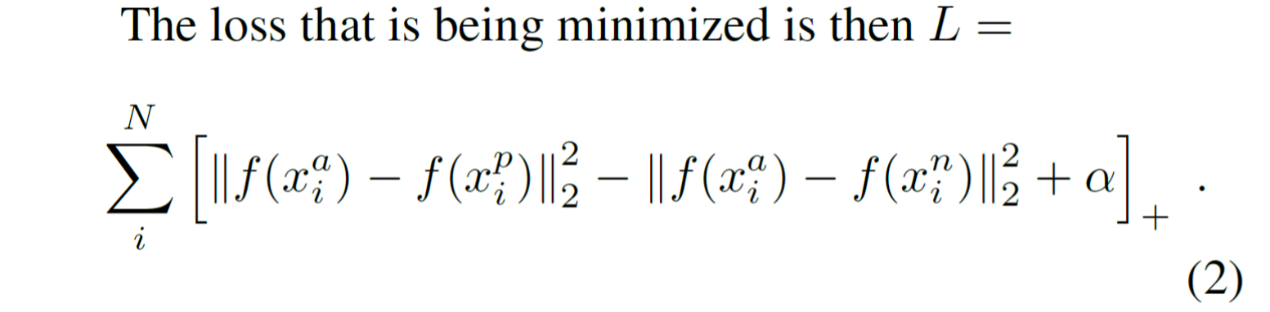
# FaceNet 17 Jun 2015

## 网络结构

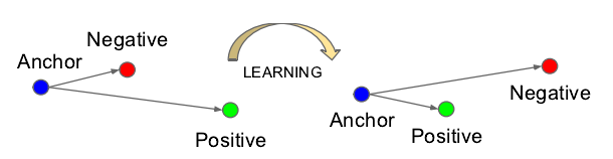


前半部分就是一个普通的卷积神经网络，但是与一般的深度学习架构不一样，Facenet没有使用Softmax作为损失函数，而是先接了一个l2\*\*嵌入\*\*（Embedding）层。

这里的映射关系是将卷积神经网络末端全连接层输出的特征映射到一个超球面上，也就是使其特征的二范数归一化，然后再以Triplet Loss为监督信号，获得网络的损失与梯度。



## Triple Loss



选择最佳三元组.

思路一) 选择hard positive和hard negative. 有一个问题, 网络受困于局部极值.

思路二) 选择所有的positive,保证收敛稳定, 选择semi-hard negative.

何为semi-hard negative?

满足a到n的距离大于a到p的距离的点(a)

è¿éåå¾çæè¿°

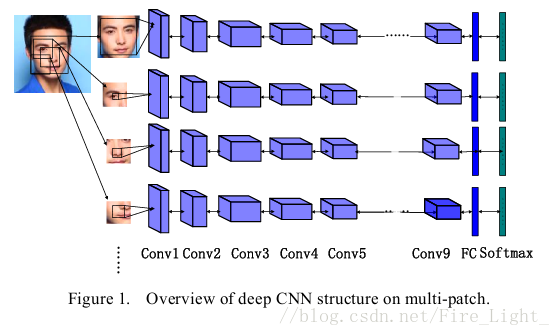
## LFW得分

在LFW上达到了**98.87% +-0.15**的验证准确率

如果预先使用更好的人脸检测算法来对齐人脸，最高可以达到**99.63% +-0.09** 的验证准确率。

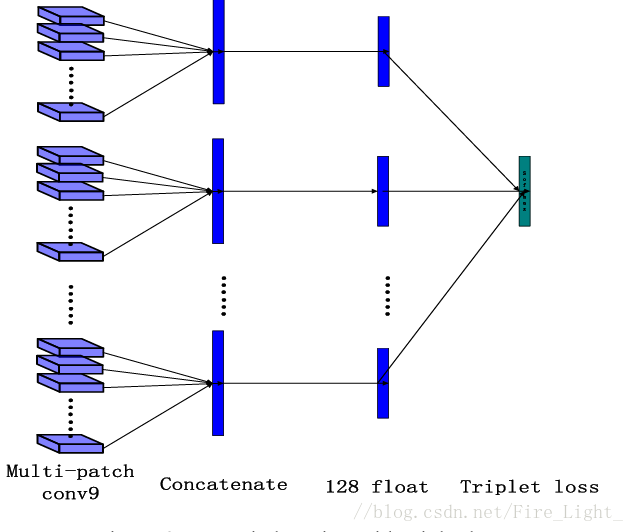
# 百度的人脸识别 23 Jul 2015

## 框架



1. 输入的是对齐好了的人脸图像
2. 将人脸图像通过面部标记点切分为多个相互之间有重叠部分的Patch
3. 分别输入**相同结构的网络**，每个网络都在不同的GPU上训练。
4. 网络结构包含9个卷积层，某些卷积层的后面有池化层，标准化层，论文中与图中都没有详细说明，大致知道网络是什么样子的就可以了。
5. **训练时用softmax训练**，进行**验证时将所有FC层的特征相接**，构成一个高维的面部特征。

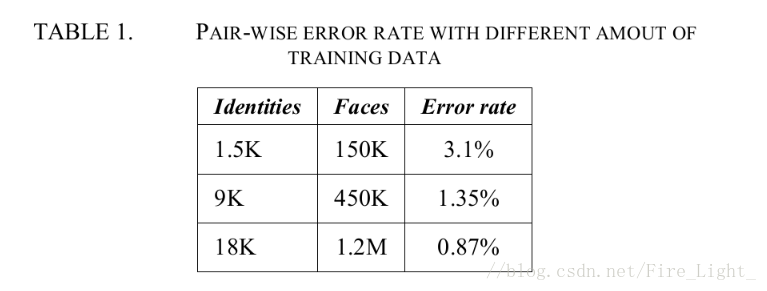
## 验证方法 metrics learning



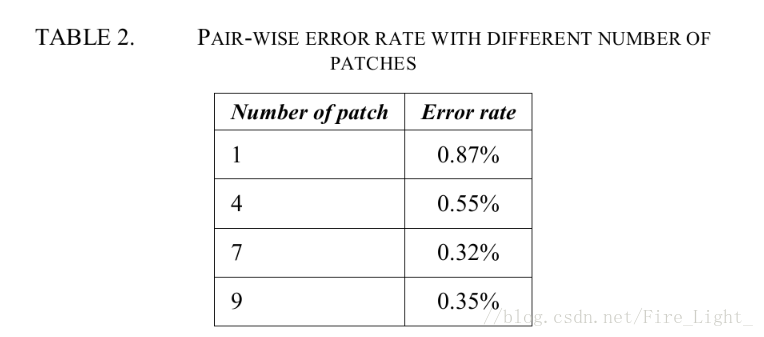
1. 通过triple loss来训练出一组embedding向量(128维的).
2. 采用度量学习(metrics learning)
3. 学习某种度量方法
4. 在特征空间找子空间.
5. 将待测特征(们)映射到(变换到)子空间中.
6. 进而在子空间的帮助下(诠释下)更好(更容易)计算各待测特征的间距.

|  |
| --- |
| *就是在特征空间中学习某种合适的距离度量方法，也就相当于去寻找特征空间中的某一子空间，将特征变换到该子空间上之后，可以很容易的度量各特征之间的距离* |

## 实验



* 1. Face数量对Error率影响, 越多Error率越小.



1. 人脸预处理的patches对Error率的影响.
2. Patches越多, Error率越小.

# VGGFace 2015

## 主要内容

* 1. 构建一个人脸数据集. 算法主导, 少量人工参与.
  2. 对比各种CNN网络, 提出简单有效的网络在人脸识别数据集上取得较好的表现.

## VggFace Dataset

构建过程主要分为5个阶段

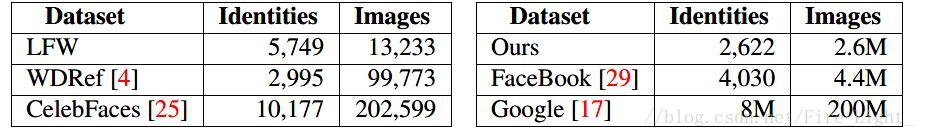
1阶段在IMDB网站上获得名人名字的名单列表，这个和CASIA-Webface有点像

2阶段是通过谷歌来扩大每个人的图片量

3阶段使用一个Fisher Vector Faces descriptor 与SVM清除一些错误的图片

4阶段清除一些一样的照片，因为从不同网站上爬下来的照片，难免会有所重复。

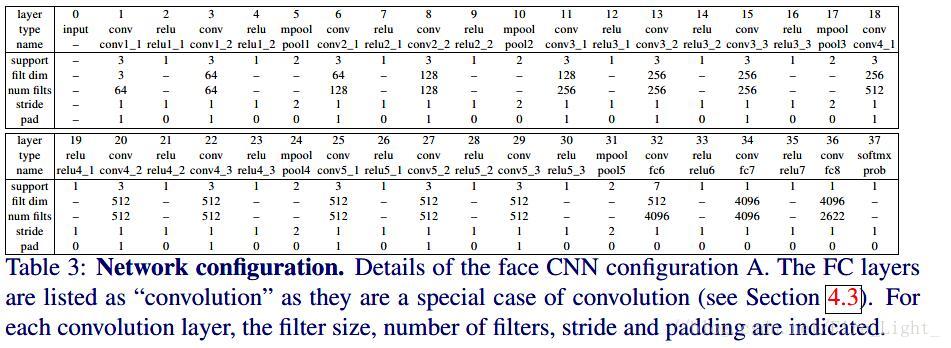
5阶段再人工清除一些照片



数据集比较

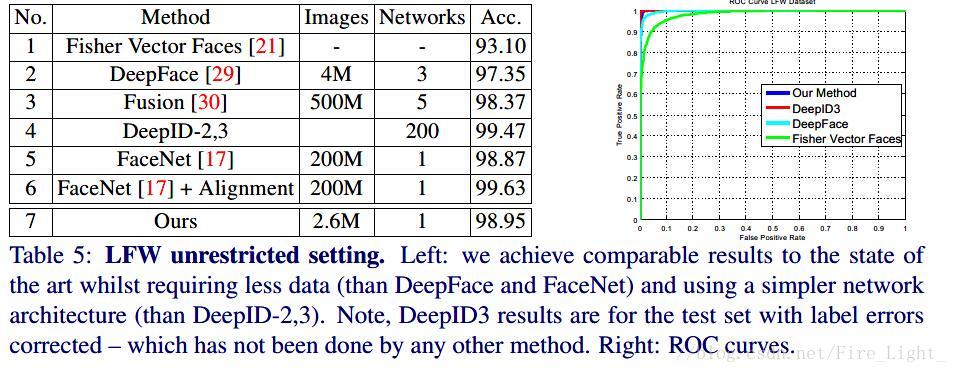
1. 图片个数以及包含的人ID个数.

## 网络结构



1. 采用VGG的结构(D结构?)
2. 使用softmax在vggDataSet上预训练
3. 在验证时,进一步以度量学习.
4. Vgg的最后一层, 2622维度上.(2622个分类上)
5. 在2622维度上,用triple Loss做一个度量学习. (缩小类内距离,扩大类间距离).

## 实验



# FR+FCN 16 Apr 2014

FR, aka, Face Recover

FCN, aka, Face Convolutional Network

为了解决什么问题?

* 1. 正面重构人脸.
  2. 目的是对齐人脸,以便做人脸验证.
  3. 涉及到一个问题
     1. 数据集中人脸很多.
     2. 需要找到”正面”的人脸作为Gt.
     3. 如何做这个筛选?