人工智能之深度学习

人脸识别

主讲人：Vincent Ying

课程要求

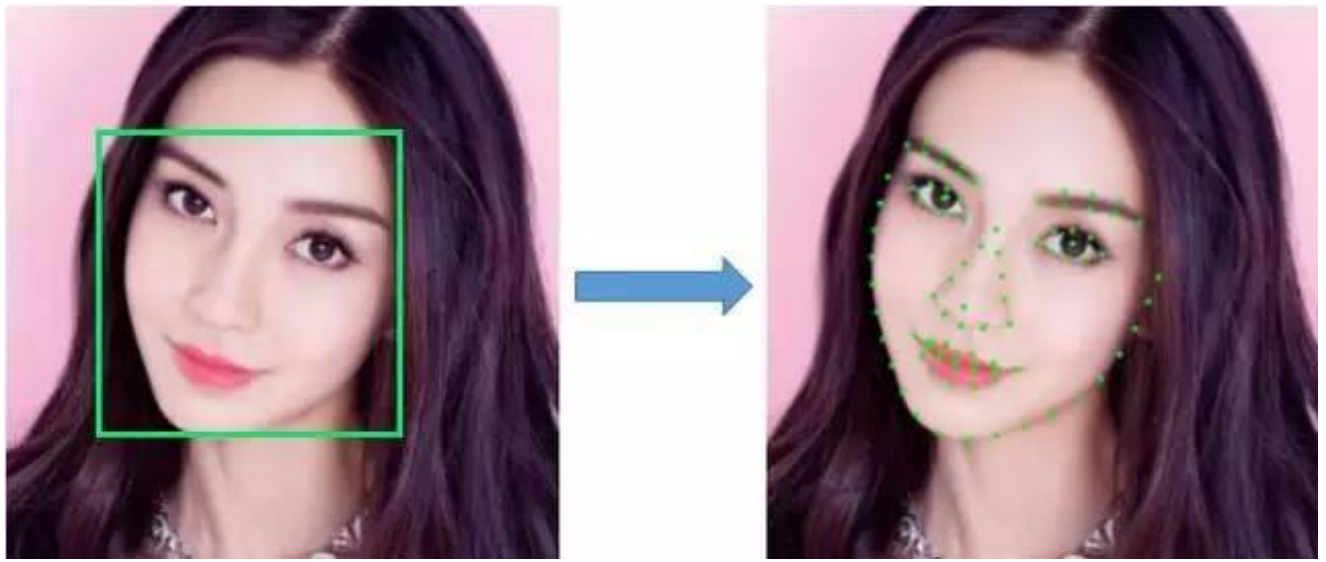
- 课上课下“九字”真言
  - 认真听，**善摘录，勤思考**
  - **多温故，乐实践**，再发散
- 四不原则
  - 不懒散惰性，不迟到早退
  - 不请假旷课，不拖延作业
- 一点注意事项
  - 违反“四不原则”，不推荐就业

课程内容

- 人脸识别概述
- **MT CNN**
- **Face Net**

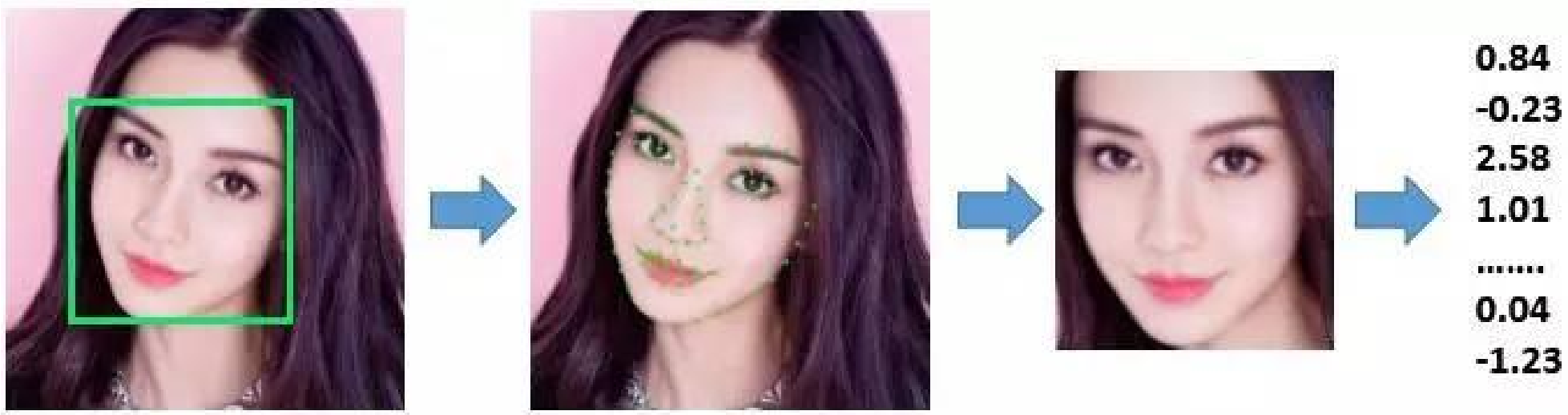
人脸识别概述

- 人脸检测(Face Detection)：从待检测的图像中获取人脸所在位置的一项技术；
- 人脸匹配(Face Alignment)：从定位出的人脸上五官关键点坐标的一项技术；



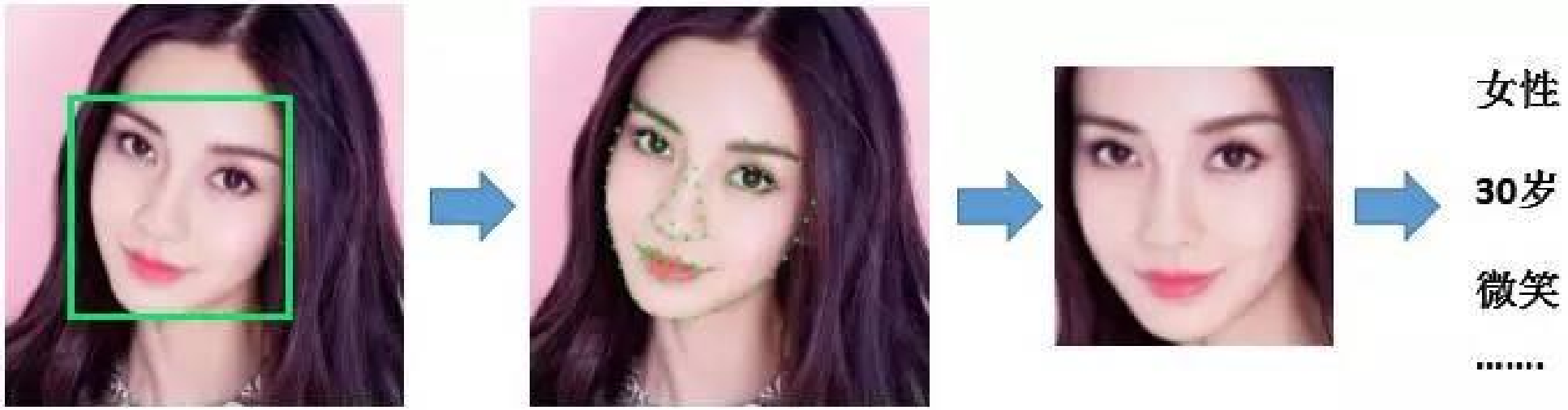
人脸识别概述

- 人脸特征提取(Face Feature Extraction)：将一张人脸图像转换为固定维度的特征向量的过程，这个向量具有表述这个人脸特征的能力。



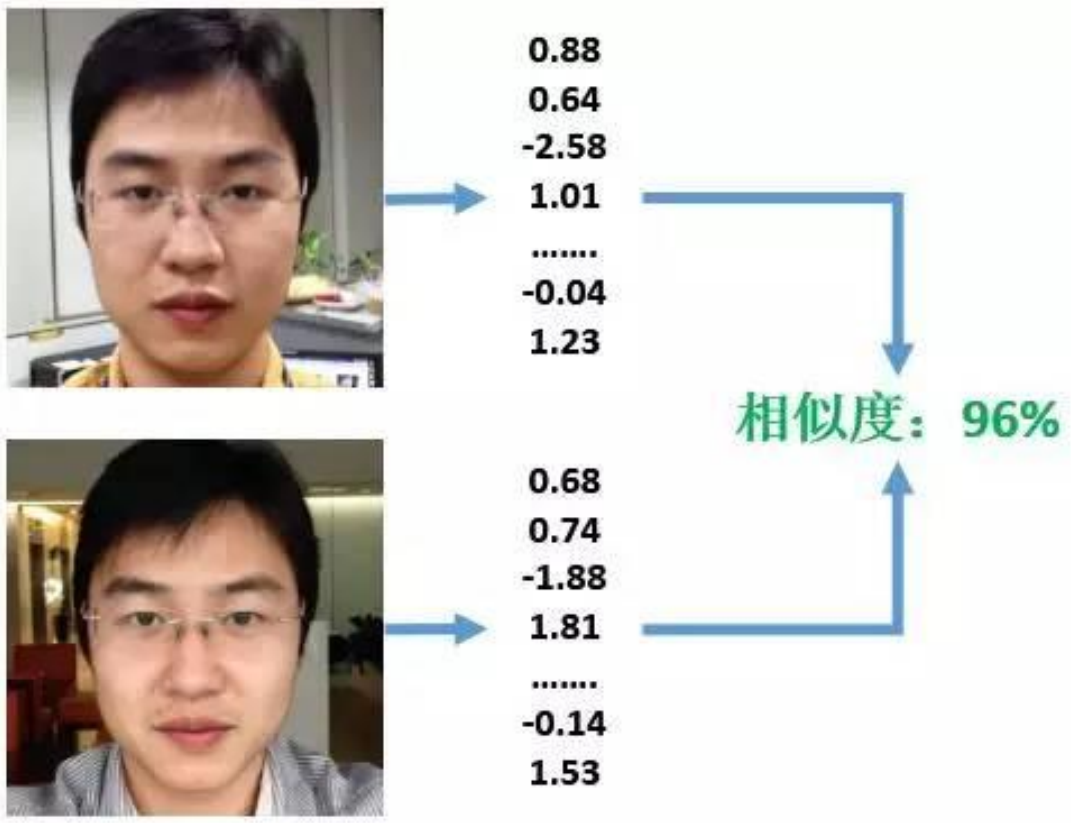
人脸识别概述

- 人脸属性识别(Face Attribute)：是识别出人脸的性别、年龄、姿态、表情等属性值的一种技术。



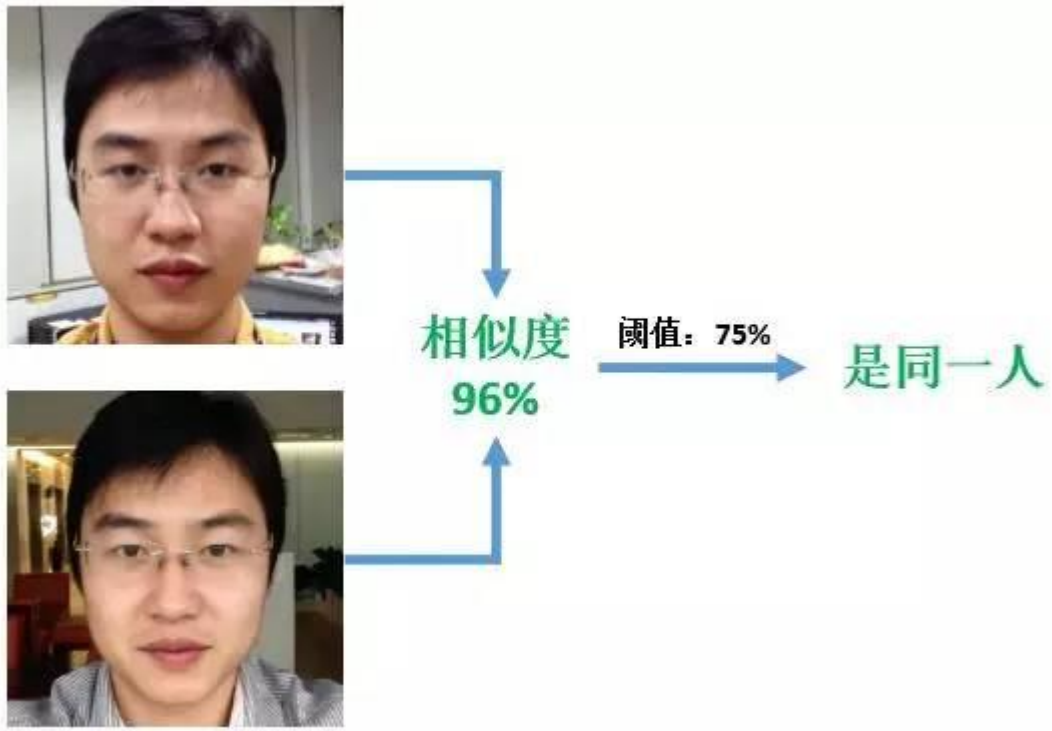
人脸识别概述

- 人脸对比(Face Compare)：衡量两个人脸之间相似度的算法，人脸对别是人脸特征提取出来的人脸特征基础上来进行的。



人脸识别概述

- 人脸验证(Face Verification)：通过输入的两个人脸特征，通过获取两个人脸特征之间的相似度，然后基于预设的阈值来比较来判断这两个人脸特征是否属于同一个人。



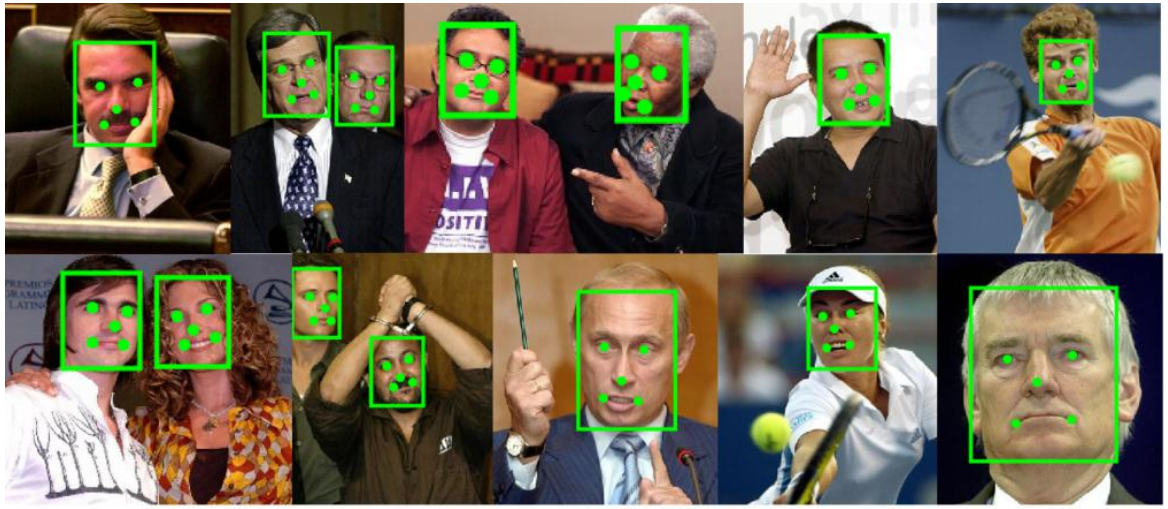
人脸识别概述

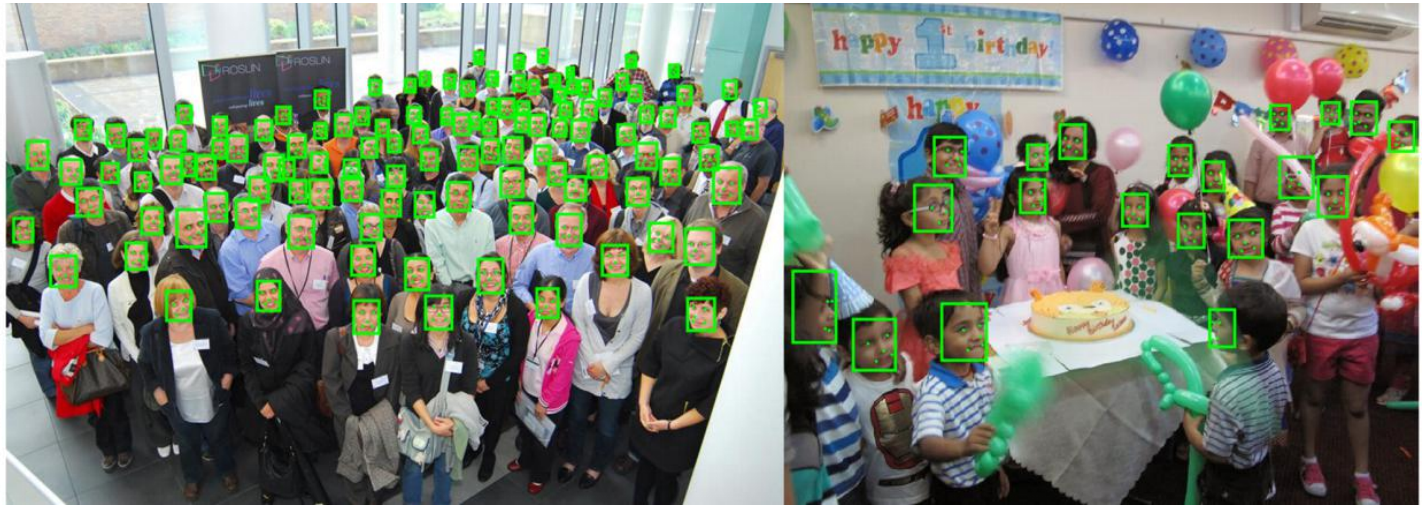
- 人脸识别/人脸检索(Face Recognition)：通过输入的人脸特征，从数据库中找出与输入特征相似度最高的用户，然后进行用户信息判断。

MTCNN

- MTCNN(Multi-task Cascaded Convolutional Networks, 多任务级联卷积神经网络)，将人脸区域检测和人脸关键点(为了做人脸对齐操作)放到一起，主体结构为级联结构，主要由三个子网络P-Net、R-Net以及O-Net构成。
  - P-Net(Proposal Networks)： 候选框快速生成；
  - R-Net(Refine Networks)： 高精度候选框过滤；
  - O-Net(Output Networks)： 边框生成以及人脸关键点检测；
  - 图像金子塔、边框回归、非极大值抑制(NMS)等。
  - **https://arxiv.org/ftp/arxiv/papers/1604/1604.02878.pdf**

MTCNN

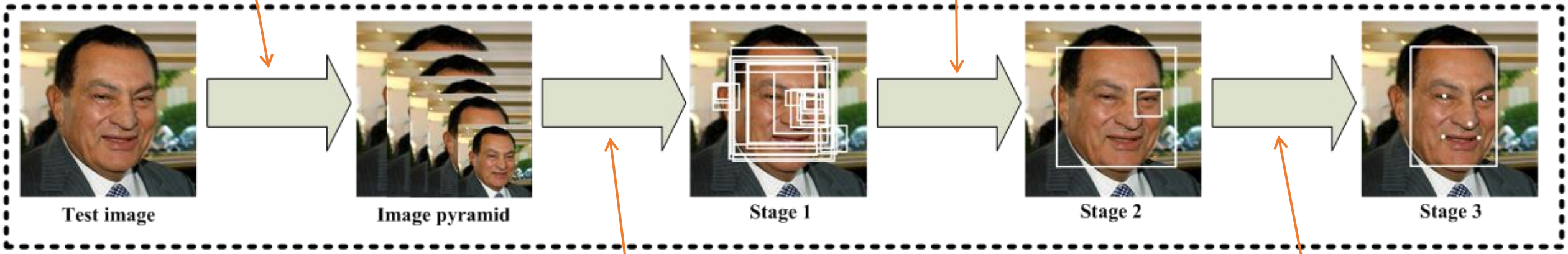




MTCNN

图像金字塔

R-Net、Bounding Box、NMS



P-Net、Bounding Box、NMS

O-Net、Bounding Box、NMS

MTCNN

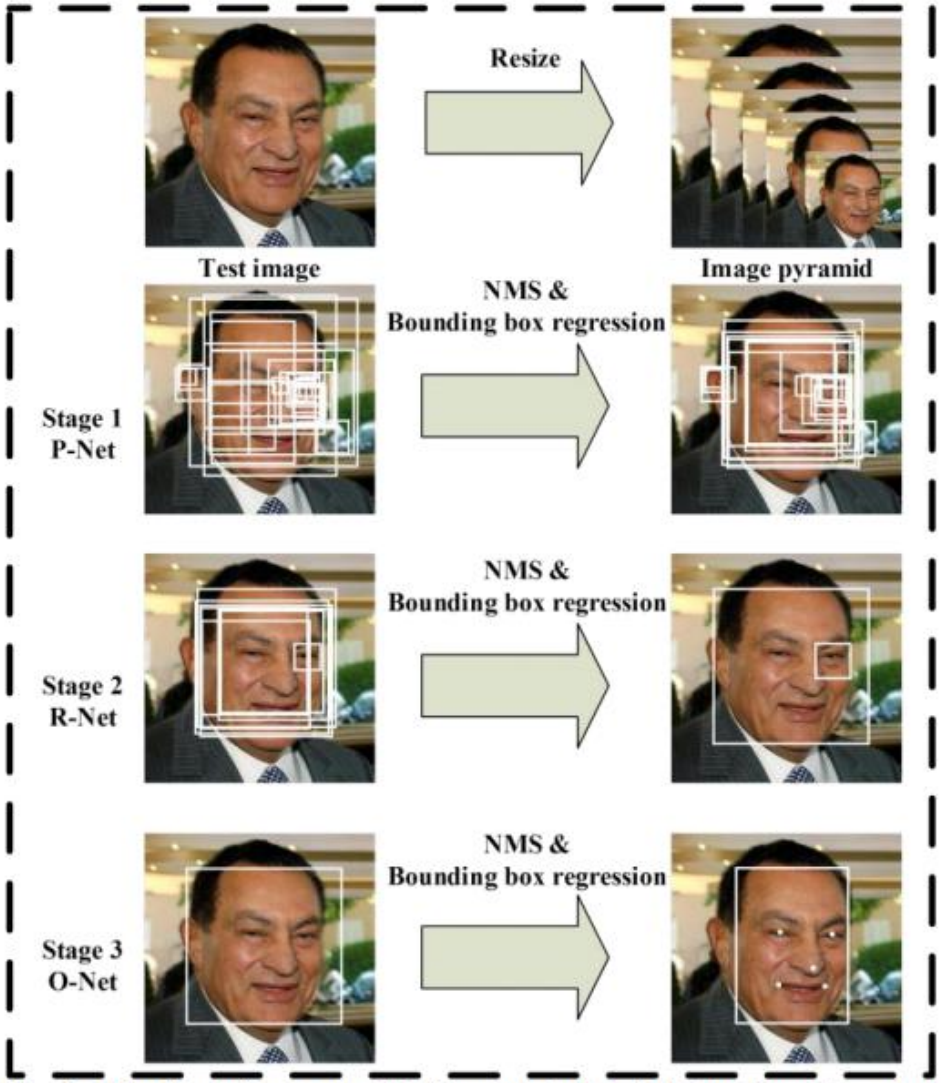
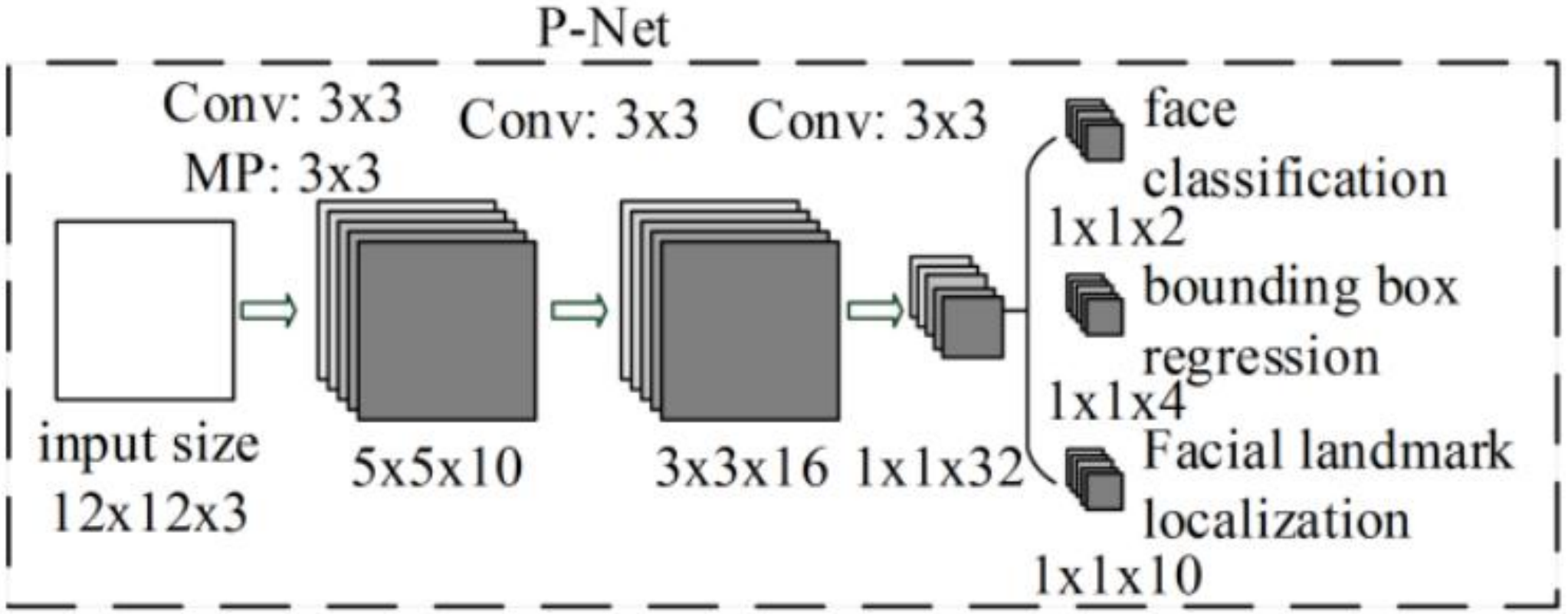


Fig. 1. Pipeline of our cascaded framework that includes three-stage multi-task deep convolutional networks. Firstly, candidate windows are produced through a fast Proposal Network (P-Net). After that, we refine these candidates in the next stage through a Refinement Network (R-Net). In the third stage, The Output Network (O-Net) produces final bounding box and facial landmarks position.

MTCNN

- P-Net：Proposal Network，基本结构就是一个全卷积网络，对于上一步构建的图像金字塔，通过FCN进行初步的特征提取以及边框标定，并进行Bounding Box Regression回归调整边框位置以及NMS进行大部分窗口的过滤；
- P-Net是一个人脸区域的区域建议网络，属于一个浅层网络结构，目的是**为了快速的产生人脸候选框**，该结构最终会输出多张可能存在人脸的人脸区域，并将这些区域输入到R-Net中进行下一步处理。

MTCNN

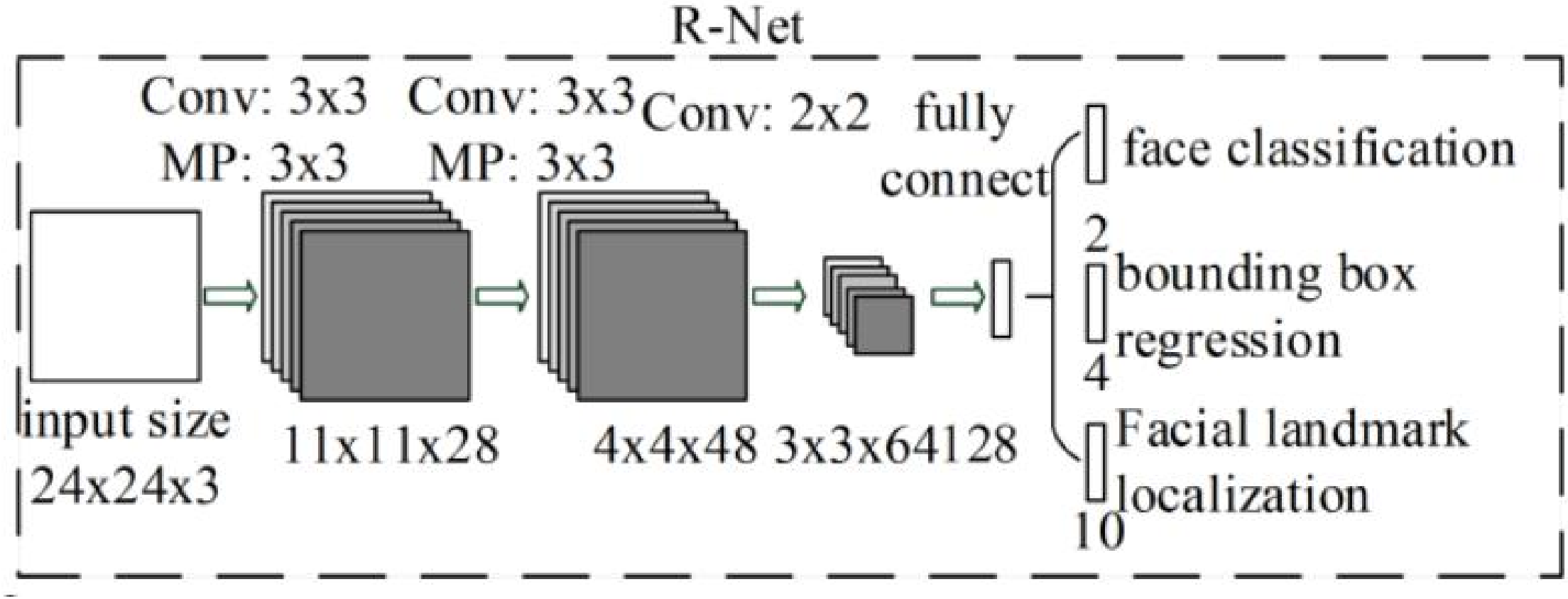


三层简单的卷积网络
三个分支：人脸分类、边框回归、关键点定位

MTCNN

- R-Net: Refine Networks，也属于简单的卷积网络结构，相比于P-Net来讲，增加一个全连接层，因此对于输入数据的筛选会更加严格。在图片经过P-Net后，会留下多个预测窗口，将所有预测窗口输入R-Net中，进一步过滤效果比较差的候选框，最终对选定的候选框使用Bounding Box Regression和NMS进一步优化结果。
- R-Net中输入的是具有一定可信度的人脸区域，输出为可信度相对来讲比较高的人脸区域，并且进行BBR和NMS优化。

MTCNN

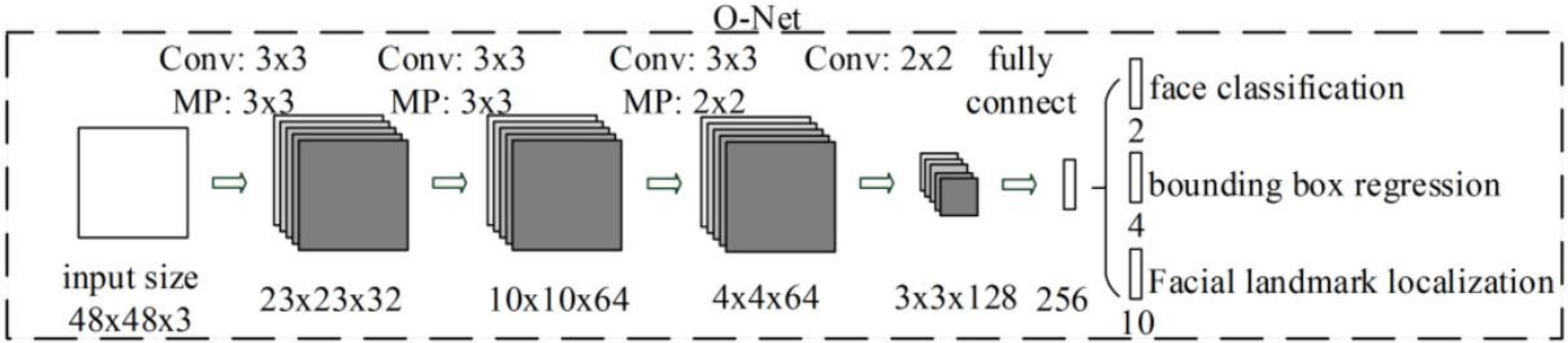


三层卷积+一层全连接的网络
三个分支：人脸分类、边框回归、关键点定位

MTCNN

- O-Net：Output Networks，基本结构是一个比较复杂的卷积神经网络，相比于P-Net和R-Net来讲，使用更加复杂的网络结构来保留更改的特征信息，从而对于得到人脸的区域定位以及最终的关键点坐标信息。
- O-Net和R-Net以及P-Net的结构类似，主要区别是使用更加复杂的网络对模型性能做优化。

MTCNN

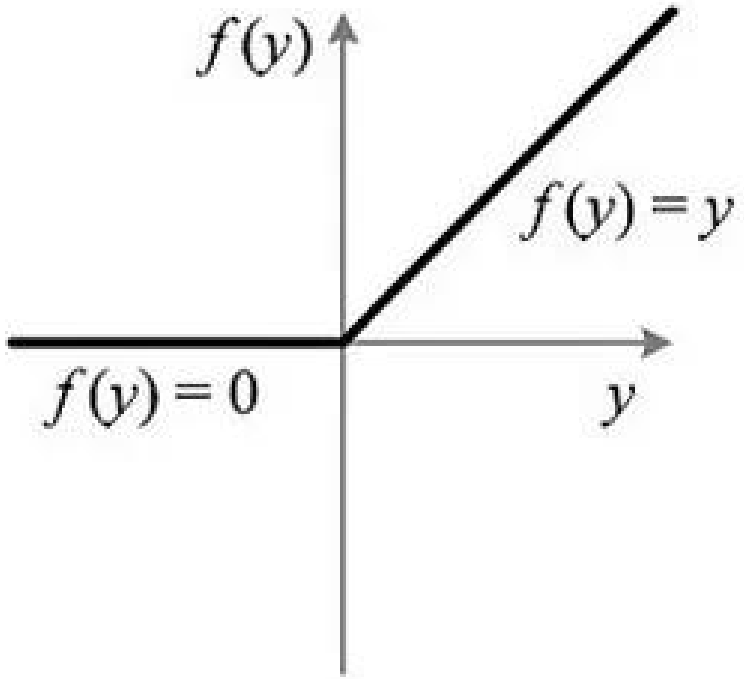


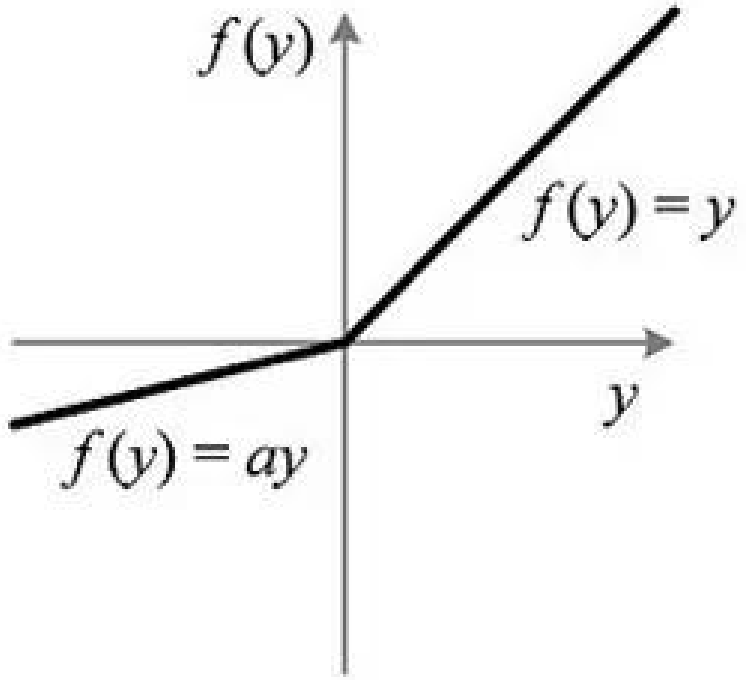
四层卷积+一层全连接的网络
三个分支：人脸分类、边框回归、关键点定位

MTCNN

- 对于CNN网络结构，影响网络性能因素的主要原因有两个：
  - 样本的多样性缺失会影响网络的鉴别能力；
  - 相比于其他的分类检测网络，人脸检测属于一个二分类，每一层不需要太多的filter。
- 结合上述原因，作者设计了每层的filter数量，将5*5的卷积更换为两个3*3的卷积核，这样相比于原始网络主要具有显著减少计算量、提升网络性能的效果，同时在网络中将所有的激活函数更改为PReLU。

MTCNN





$$ReLU(x) = \begin{cases} x & if\, x > 0 \\ 0 & if\, x \le 0 \end{cases}$$

$$PReLU(x_i) = \begin{cases} x_i & if\, x_i > 0 \\ a_i x_i & if\, x_i \le 0 \end{cases}$$

i 表示不同的通道

MTCNN

- Face Classification：二分类问题(是否属于人脸区域)，使用交叉熵损失函数。

$$L_i^{det} = -(y_i^{det} \log(p_i) + (1 - y_i^{det})(1 - \log(p_i)))$$

p_i为网络预测当前区域属于人脸区域的概率值；
y_i_det为当前区域实际上是否包含人脸，
0表示不包含，1表示包含

MTCNN

- Bounding Box Regression：直接预测边框坐标，这里采用直接预测边框坐标的主要目的是为了快速的获取边框，属于一个回归问题，故采用MSE距离损失函数。

$$L_i^{box} = \left\| \hat{y}_i^{box} - y_i^{box} \right\|_2^2$$

预测边框: (x, y, height, width)

实际边框: (x, y, height, width)

MTCNN

- Facial landmark localization：和BBR一样，Facial landmark localization的也是一个回归模型，其主要预测值为5点的坐标，分别为: left eye、right eye、nose、left mouth cornet、right mouth cornet；总共由5个坐标值组成的10维向量。

$$L_i^{landmark} = \left\| \hat{y}_i^{landmark} - y_i^{landmark} \right\|_2^2$$

预测坐标

实际坐标

MTCNN

- MTCNN中模型的损失函数是由分类、回归以及定位三部分损失线性组合而来的。

alpha阈值加大的主要原因是为了让定位越正确。为了获取更加精确的脸部标记点位置点信息。

$$\min \sum_{i=1}^{N} \sum_{j \in \{det, box, landmark\}} \alpha_j \beta_i^j L_i^j \qquad (4)$$

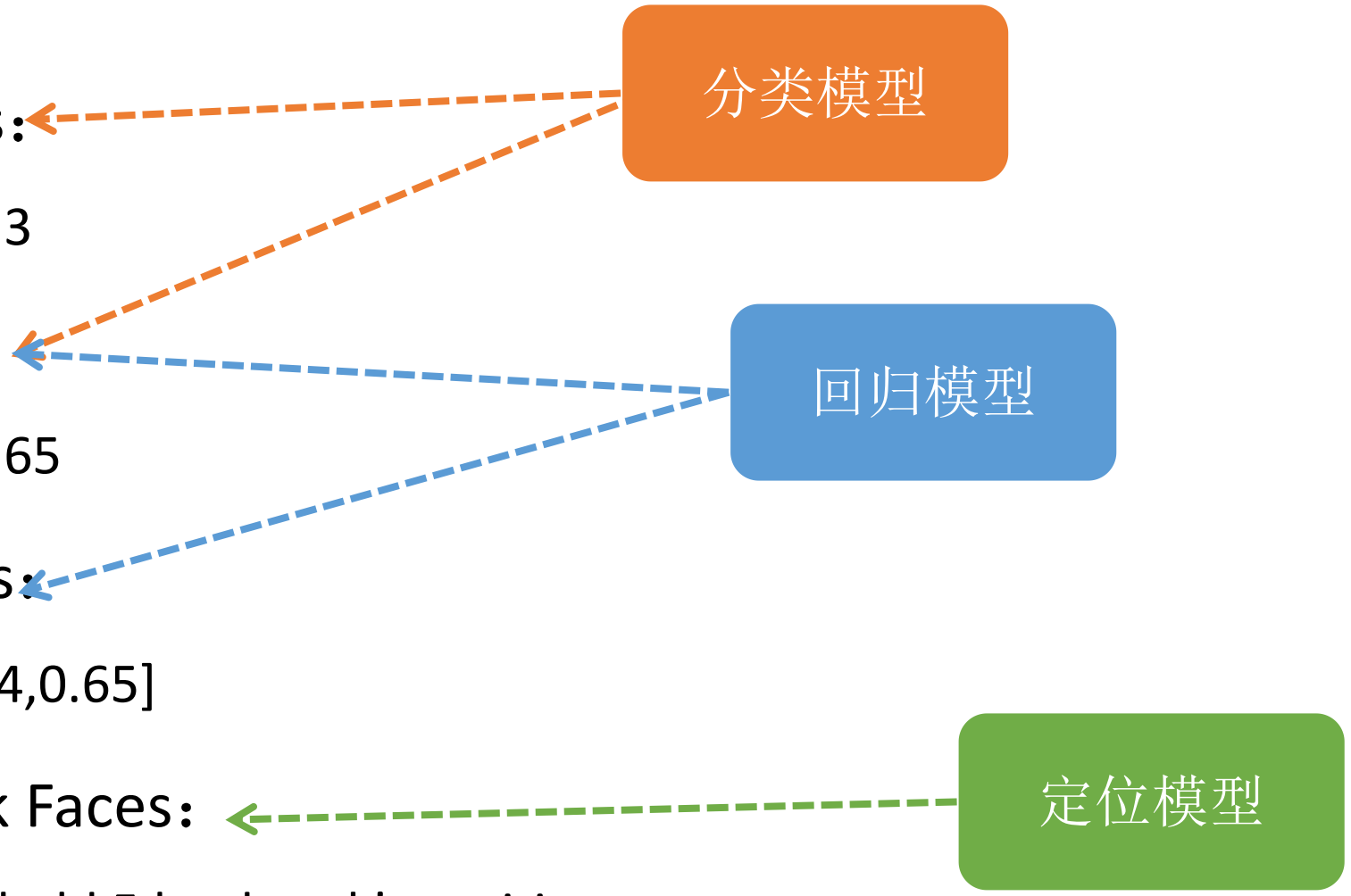
where N is the number of training samples. α_j denotes on the task importance. We use $(\alpha_{det} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 0.5)$ in P-Net and R-Net, while $(\alpha_{det} = 1, \alpha_{box} = 0.5, \alpha_{landmark} = 1)$ in O-Net for more accurate facial landmarks localization. $\beta_i^j \in \{0,1\}$ is the sample type indicator. In this case, it is natural to employ stochastic gradient descent to train the CNNs.

MTCNN

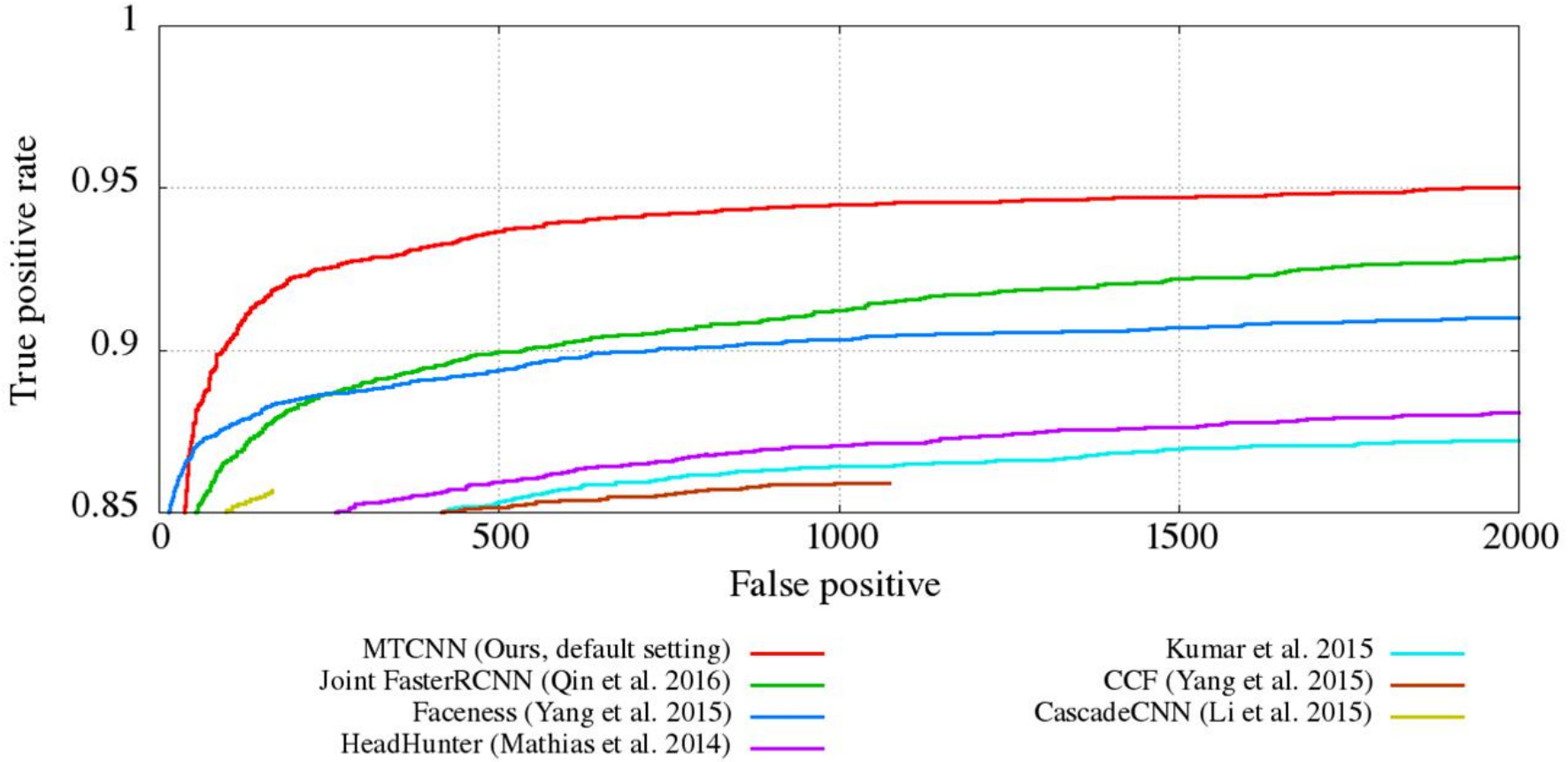
- Online Hard Sample Mining(在线难样本挖掘算法)：
  - 在模型训练的过程中，对于每个批次的样本而言，不是使用所有样本的损失进行模型更新的，而是使用70%的损失函数值最高的样本来反向传播的，原因是：**好的样本对于网络的提升效果有限，只有那些难样本更加有效训练，进行反向传播之后才可以更高的提升网络效果。**

MTCNN

- Negatives：
  - IoU < 0.3
- Positives：
  - IoU > 0.65
- Part Faces：
  - IoU: [0.4,0.65]
- Landmark Faces：
  - faces labeld 5 landmark’s positions

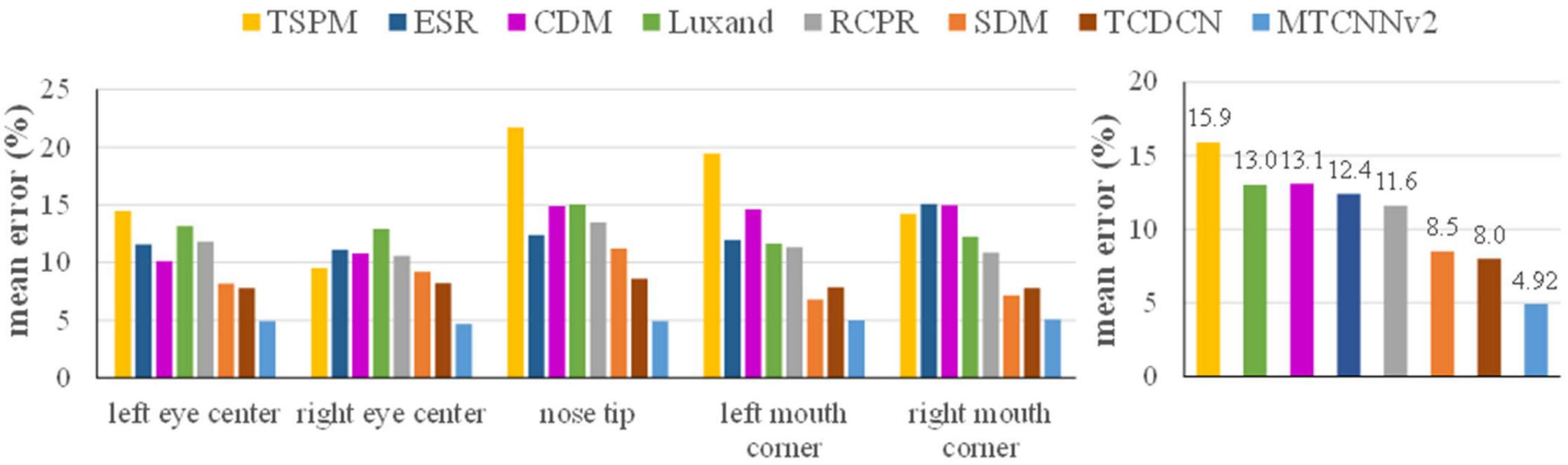


MTCNN



(c) Result on FDDB for face detection

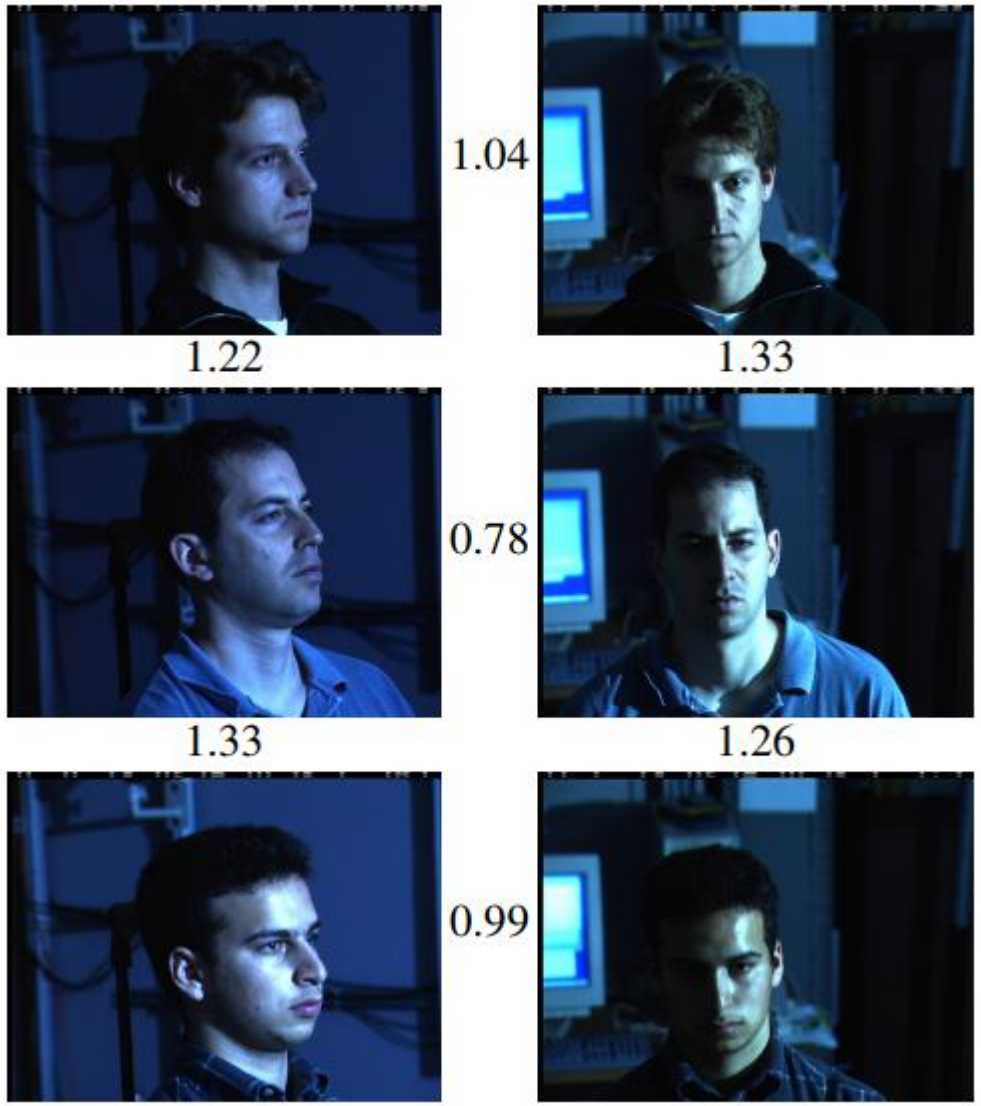
MTCNN



(d) Result on AFLW for face alignment

Face Net

- FaceNet(A Unified Embedding for Face Recognition and Clustering)，https://arxiv.org/pdf/1503.03832.pdf
- 通过 CNN 将人脸映射到欧式空间的特征向量上，计算不同图片人脸特征的距离，通过**相同个体人脸的距离，总是小于不同个体人脸的距离**这一先验知识训练网络。



Face Net

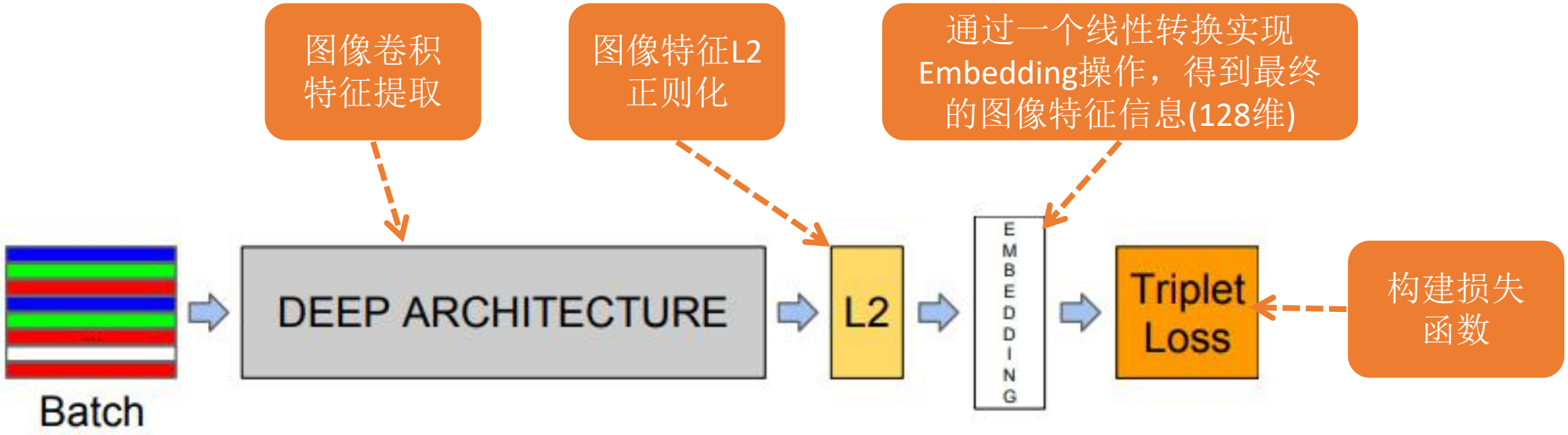


Figure 2. **Model structure.** Our network consists of a batch input layer and a deep CNN followed by L_2 normalization, which results in the face embedding. This is followed by the triplet loss during training.

Face Net

- 存在特征向量的维度选择问题，维度约小计算越快，但是太小的话很难区分不同图片；维度越大越容易区分不同图片，但是太大训练模型不易收敛，且测试时计算慢，占用空间大。作者实验证明 128 维的特征能够较好的平衡这个问题。

#dims	VAL
64	86.8% ± 1.7
128	87.9% ± 1.9
256	87.7% ± 1.9
512	85.6% ± 2.0

Table 5. **Embedding Dimensionality.** This Table compares the effect of the embedding dimensionality of our model NN1 on our hold-out set from section 4.1. In addition to the VAL at 10E-3 we also show the standard error of the mean computed across five splits.

Face Net

- 三元损失函数(Triplet Loss)
  - 二元损失函数的目标是把相同个体的人脸特征映射到空间中是距离近的点。
  - 三元损失函数目标是映射到相同的区域，使得同一个体内距离小于不同个体间距离(聚类)。

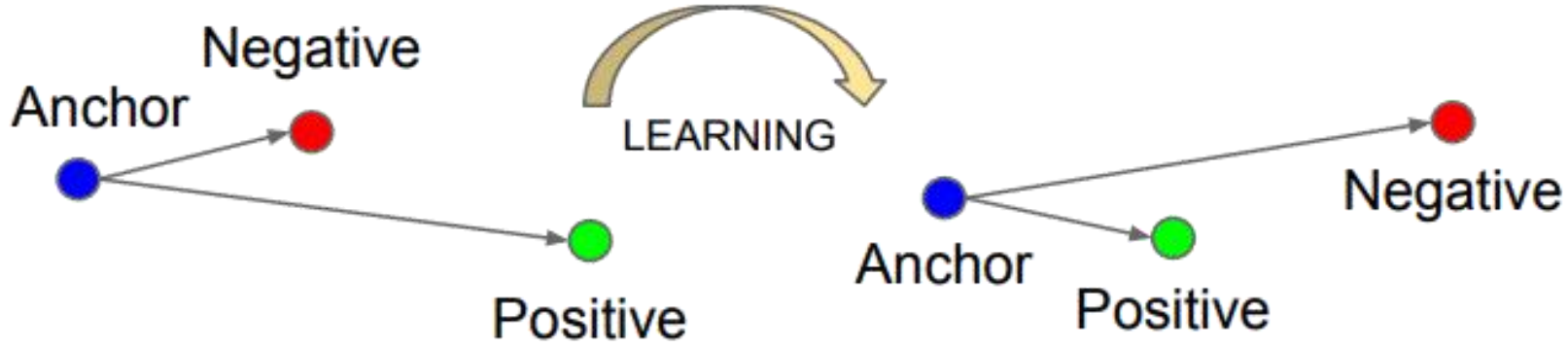
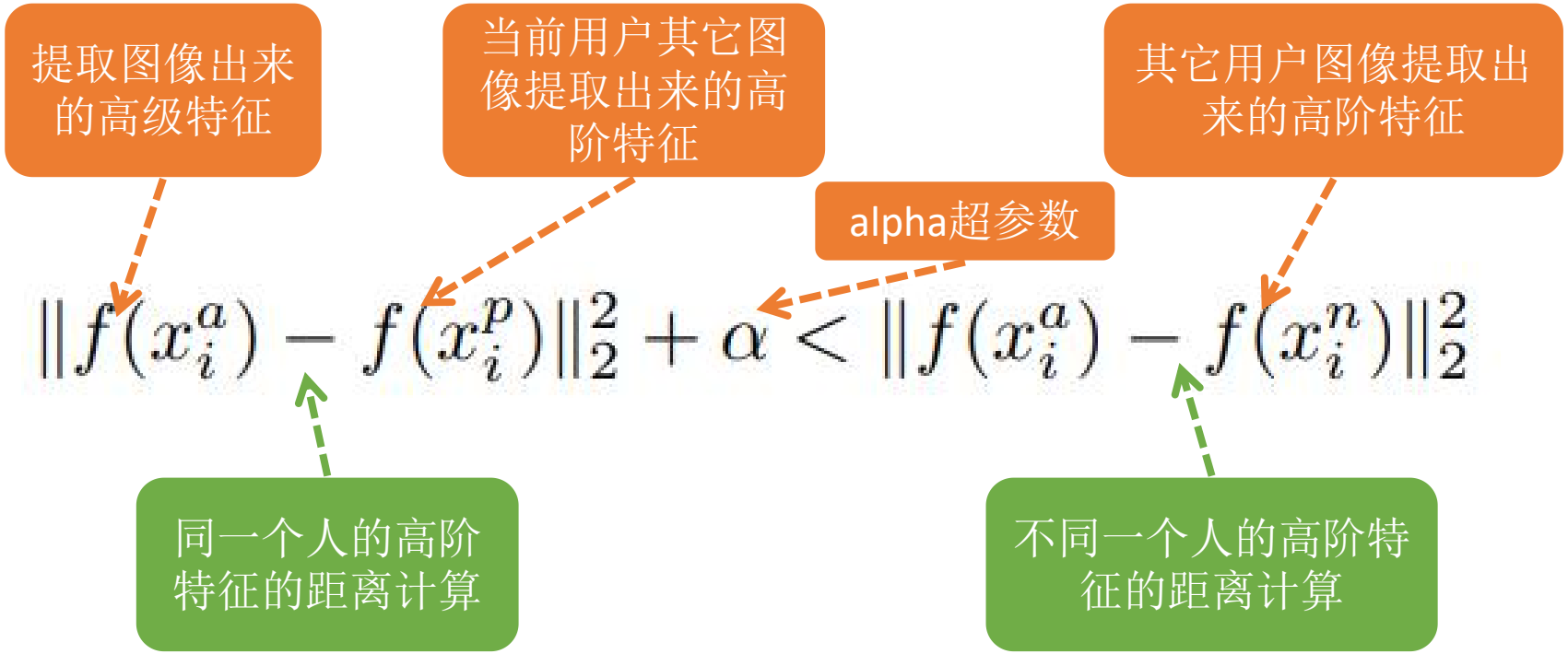
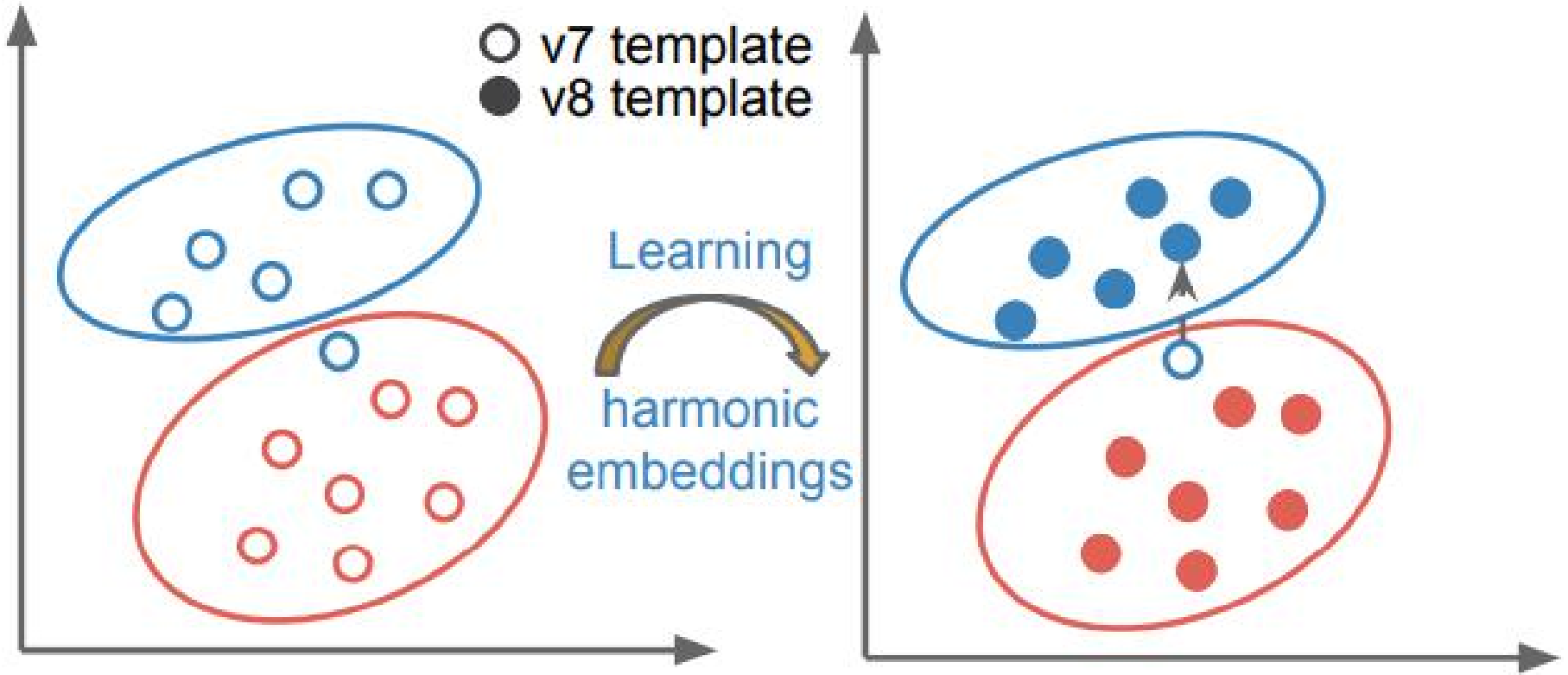


Figure 3. The **Triplet Loss** minimizes the distance between an *anchor* and a *positive*, both of which have the same identity, and maximizes the distance between the *anchor* and a *negative* of a different identity.

Face Net


提取图像出来的高级特征
当前用户其它图像提取出来的高阶特征
其它用户图像提取出来的高阶特征
alpha超参数
$\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2$
同一个人的高阶特征的距离计算
不同一个人的高阶特征的距离计算

Face Net



Face Net

- 损失函数：
  - 实际上不是使用所有样本来计算损失，在同一个类别中，选择距离最远的样本(hard positive)；在不同类别中，选择距离最近的样本(hard negative)的距离来构造损失函数。

$$\sum_{i}^{N}\left[\|f(x_i^a)-f(x_i^p)\|_2^2-\|f(x_i^a)-f(x_i^n)\|_2^2+\alpha\right]$$

选择同一个人脸的不同图像上距离最远的

选择不同一个人脸图像上距离最近的

Face Net

- 网络结构：基于VGGNet

layer	size-in	size-out	kernel	param	FLPS
conv1	220×220×3	110×110×64	7×7×3, 2	9K	115M
pool1	110×110×64	55×55×64	3×3×64, 2	0	
rnorm1	55×55×64	55×55×64		0	
conv2a	55×55×64	55×55×64	1×1×64, 1	4K	13M
conv2	55×55×64	55×55×192	3×3×64, 1	111K	335M
rnorm2	55×55×192	55×55×192		0	
pool2	55×55×192	28×28×192	3×3×192, 2	0	
conv3a	28×28×192	28×28×192	1×1×192, 1	37K	29M
conv3	28×28×192	28×28×384	3×3×192, 1	664K	521M
pool3	28×28×384	14×14×384	3×3×384, 2	0	
conv4a	14×14×384	14×14×384	1×1×384, 1	148K	29M
conv4	14×14×384	14×14×256	3×3×384, 1	885K	173M
conv5a	14×14×256	14×14×256	1×1×256, 1	66K	13M
conv5	14×14×256	14×14×256	3×3×256, 1	590K	116M
conv6a	14×14×256	14×14×256	1×1×256, 1	66K	13M
conv6	14×14×256	14×14×256	3×3×256, 1	590K	116M
pool4	14×14×256	7×7×256	3×3×256, 2	0	
concat	7×7×256	7×7×256		0	
fc1	7×7×256	1×32×128	maxout p=2	103M	103M
fc2	1×32×128	1×32×128	maxout p=2	34M	34M
fc7128	1×32×128	1×1×128		524K	0.5M
L2	1×1×128	1×1×128		0	
total				140M	1.6B

Table 1. **NN1.** This table show the structure of our Zeiler&Fergus [22] based model with 1×1 convolutions inspired by [9]. The input and output sizes are described in $rows \times cols \times \#filters$. The kernel is specified as $rows \times cols, stride$ and the maxout [6] pooling size as $p = 2$.

Face Net

- 网络结构：基于GoogleNet

type	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj (p)	params	FLOPS
conv1 (7×7×3, 2)	112×112×64	1							9K	119M
max pool + norm	56×56×64	0						m 3×3, 2		
inception (2)	56×56×192	2		64	192				115K	360M
norm + max pool	28×28×192	0						m 3×3, 2		
inception (3a)	28×28×256	2	64	96	128	16	32	m, 32p	164K	128M
inception (3b)	28×28×320	2	64	96	128	32	64	L_2, 64p	228K	179M
inception (3c)	14×14×640	2	0	128	256,2	32	64,2	m 3×3,2	398K	108M
inception (4a)	14×14×640	2	256	96	192	32	64	L_2, 128p	545K	107M
inception (4b)	14×14×640	2	224	112	224	32	64	L_2, 128p	595K	117M
inception (4c)	14×14×640	2	192	128	256	32	64	L_2, 128p	654K	128M
inception (4d)	14×14×640	2	160	144	288	32	64	L_2, 128p	722K	142M
inception (4e)	7×7×1024	2	0	160	256,2	64	128,2	m 3×3,2	717K	56M
inception (5a)	7×7×1024	2	384	192	384	48	128	L_2, 128p	1.6M	78M
inception (5b)	7×7×1024	2	384	192	384	48	128	m, 128p	1.6M	78M
avg pool	1×1×1024	0								
fully conn	1×1×128	1							131K	0.1M
L2 normalization	1×1×128	0								
total									7.5M	1.6B

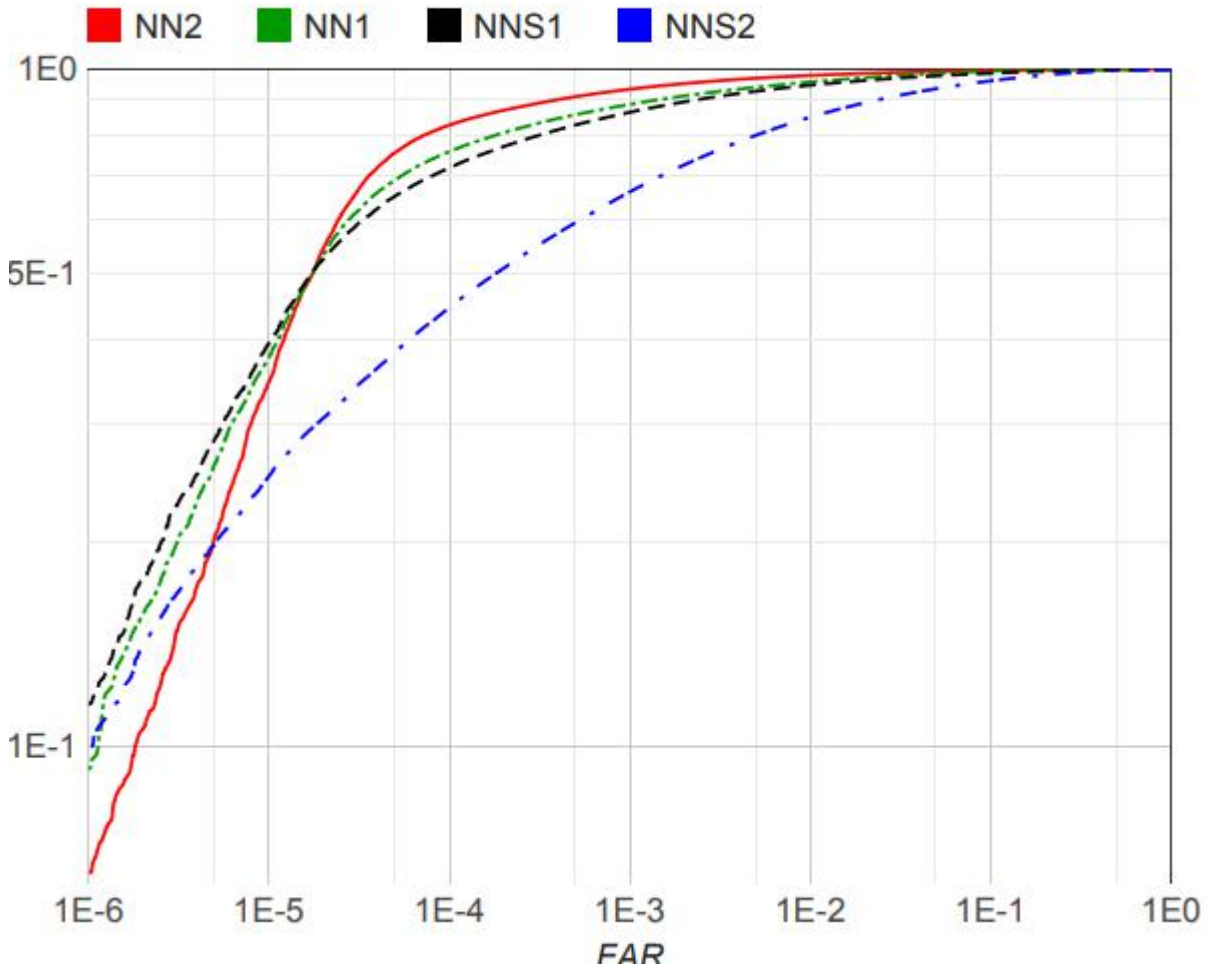
Table 2. **NN2.** Details of the NN2 Inception incarnation. This model is almost identical to the one described in [16]. The two major differences are the use of L_2 pooling instead of max pooling (m), where specified. *I.e.* instead of taking the spatial max the L_2 norm is computed. The pooling is always 3×3 (aside from the final average pooling) and in parallel to the convolutional modules inside each Inception module. If there is a dimensionality reduction after the pooling it is denoted with p. 1×1, 3×3, and 5×5 pooling are then concatenated to get the final output.

Face Net

architecture	VAL
NN1 (Zeiler&Fergus 220×220)	87.9% ± 1.9
NN2 (Inception 224×224)	89.4% ± 1.6
NN3 (Inception 160×160)	88.3% ± 1.7
NN4 (Inception 96×96)	82.0% ± 2.3
NNS1 (mini Inception 165×165)	82.4% ± 2.4
NNS2 (tiny Inception 140×116)	51.9% ± 2.9

Table 3. **Network Architectures.** This table compares the performance of our model architectures on the hold out test set (see section 4.1). Reported is the mean validation rate VAL at 10E-3 false accept rate. Also shown is the standard error of the mean across the five test splits.

Face Net



扩展：人脸识别经典数据集

- LFW： http://vis-www.cs.umass.edu/lfw/
- PubFig： http://www.cs.columbia.edu/CAVE/databases/pubfig/
- YouTube Faces DB： http://www.cs.tau.ac.il/~wolf/ytfaces/
- MTFL： http://mmlab.ie.cuhk.edu.hk/projects/TCDCN.html