

人脸检测简要综述

评价一个人脸检测算法常用三个指标：

召回率(recall)： $\text{recall} = \text{检测出来的人脸数量} / \text{图像中总人脸数量}$ 。

误检数(false positives)： 与 recall 相对，检测出来的矩形框与任何人工标注框的 IoU 都小于 0.5，则认为这个检测结果是误检，误检越少越好，论文中一般比较 1000 个或 2000 个误检时的召回率情况，工业应用中通常比较 100 或 200 个误检的召回率情况，在误检数一定时，召回率越大越好。

检测速度(FPS)： 检测速度与硬件配置和测试图像尺寸有直接关系，测试图像最常用的尺寸是 VGA(640*480) 图像检测最小人脸 80*80 给出速度；测试环境，CPU 有不同型号和主频，有多核多线程差异，GPU 也有不同型号。

以下对比了近几年内开源的人脸检测算法，主要对比了召回率（500 个误检数情况下）和检测速度 FPS（以标准 VGA 图像作为输入）。其中召回率在 90%以上，CPU-1 指的是单线程 CPU，GPU 是中等性能的（差不多 2080ti）。

Method	recall/ 500	FPS (VGA)		year
		CPU - 1	GPU	
MTCNN	93.54%	16	122	2016
FaceBoxes	95.16%	20	125	2017
DSFD	95.72%	46	380	2018
RetinaFace	96.30%	58	592	2019
Blaze Face	98.61%	—	172	2019

方法 MTCNN 和 RetinaFace 除了画出人脸框之外，还标出了人脸的 5 个特征点，RetinaFace 应该是目前既能保证在获得高召回率的同时，又能在 CPU，GPU，ARM 等平台上实现实时检测的算法。DSFD、FaceBoxes 及 Blaze Face 只画出人脸框，其中 Blaze Face 是一个专门为移动端 GPU 设计的轻量化人脸检测网络，其在 huawei P20 上达到 172FPS，在 Apple iPhone XS 上可达到 1600FPS。

以下为各算法的 github 地址：

MTCNN: <https://github.com/AITTSMD/MTCNN-Tensorflow>

FaceBoxes: <https://github.com/zisianw/FaceBoxes.PyTorch>

DSFD: <https://github.com/TencentYoutuResearch/FaceDetection-DSFD>

RetinaFace: <https://github.com/deepinsight/insightface>

Blaze Face: https://github.com/tkat0/PyTorch_BlazeFace