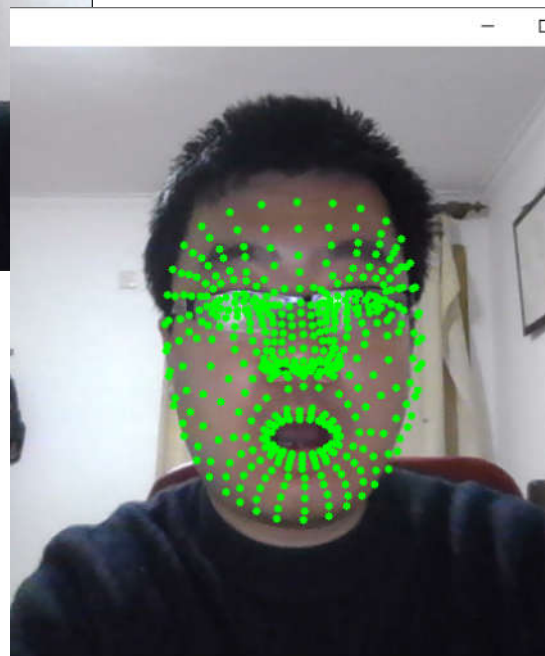


## Python编程与人工智能实践

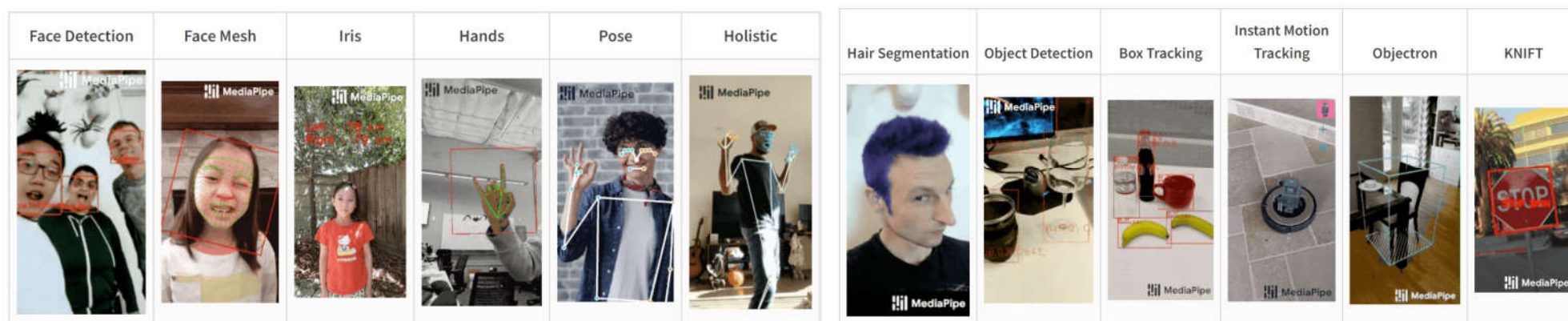


应用篇：  
基于mediapipe的人脸关键点检测

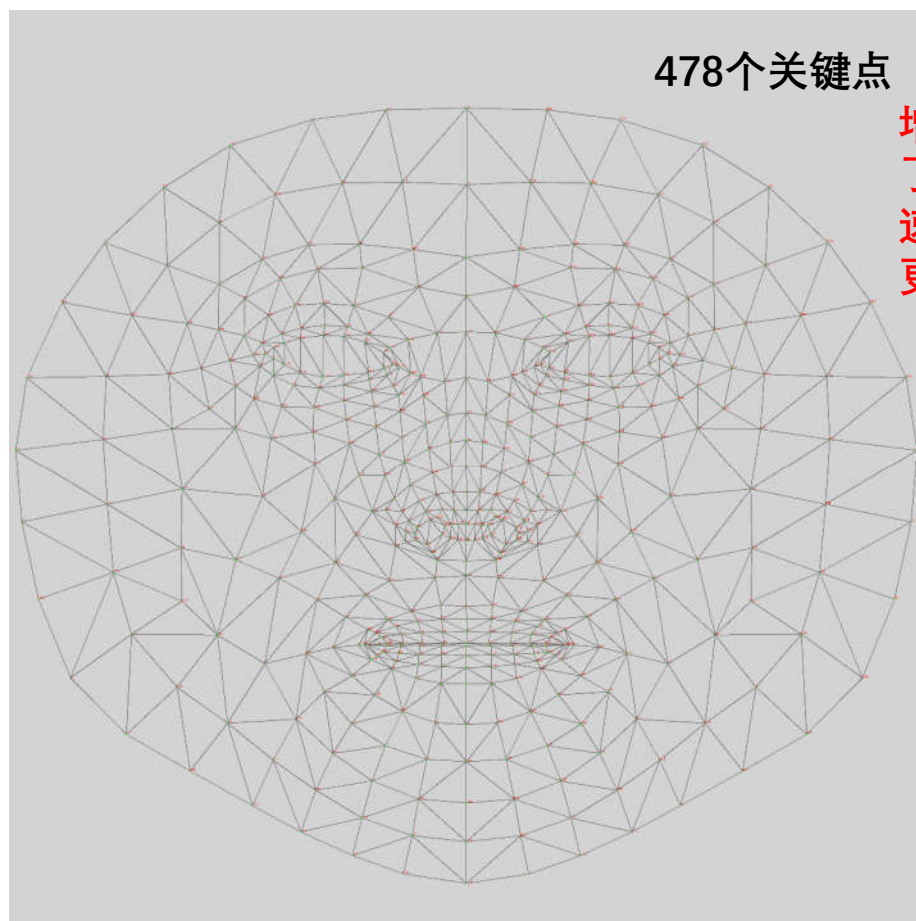
于泓  
鲁东大学  
信息与电气工程学院  
2022.3.23

# MediaPipe: Google Research 开源的跨平台多媒体机器学习模型应用框架

作为一款跨平台框架，MediaPipe 不仅可以被部署在服务器端，更可以在多个移动端（安卓和苹果 iOS）和嵌入式平台（Google Coral 和树莓派）中作为设备端机器学习推理（On-device Machine Learning Inference）框架。



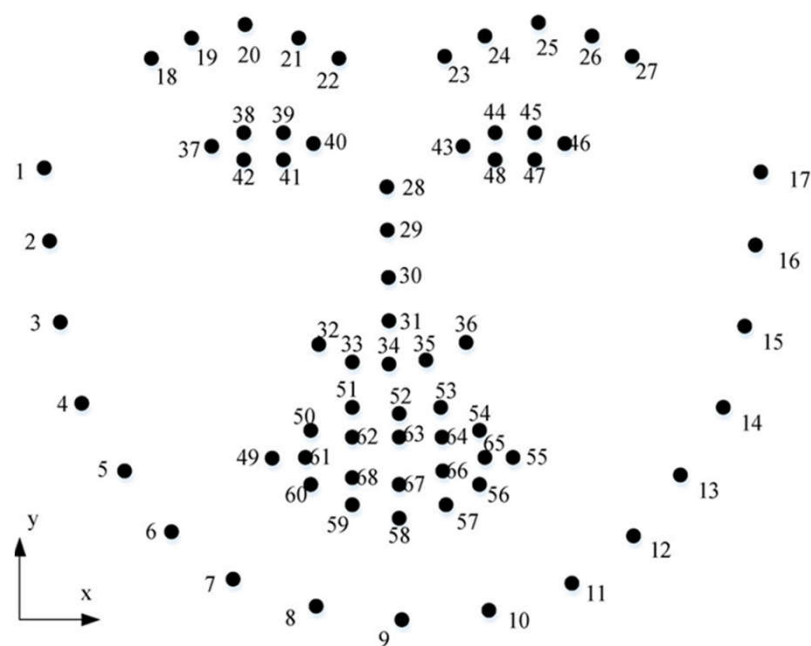
# Face-mesh



增加了跟踪  
速度更快

# DLib

68个关键点



```
import mediapipe as mp
import cv2
import numpy as np

if __name__ == "__main__":
    # 构建脸部特征提取对象
    mp_face_mesh = mp.solutions.face_mesh

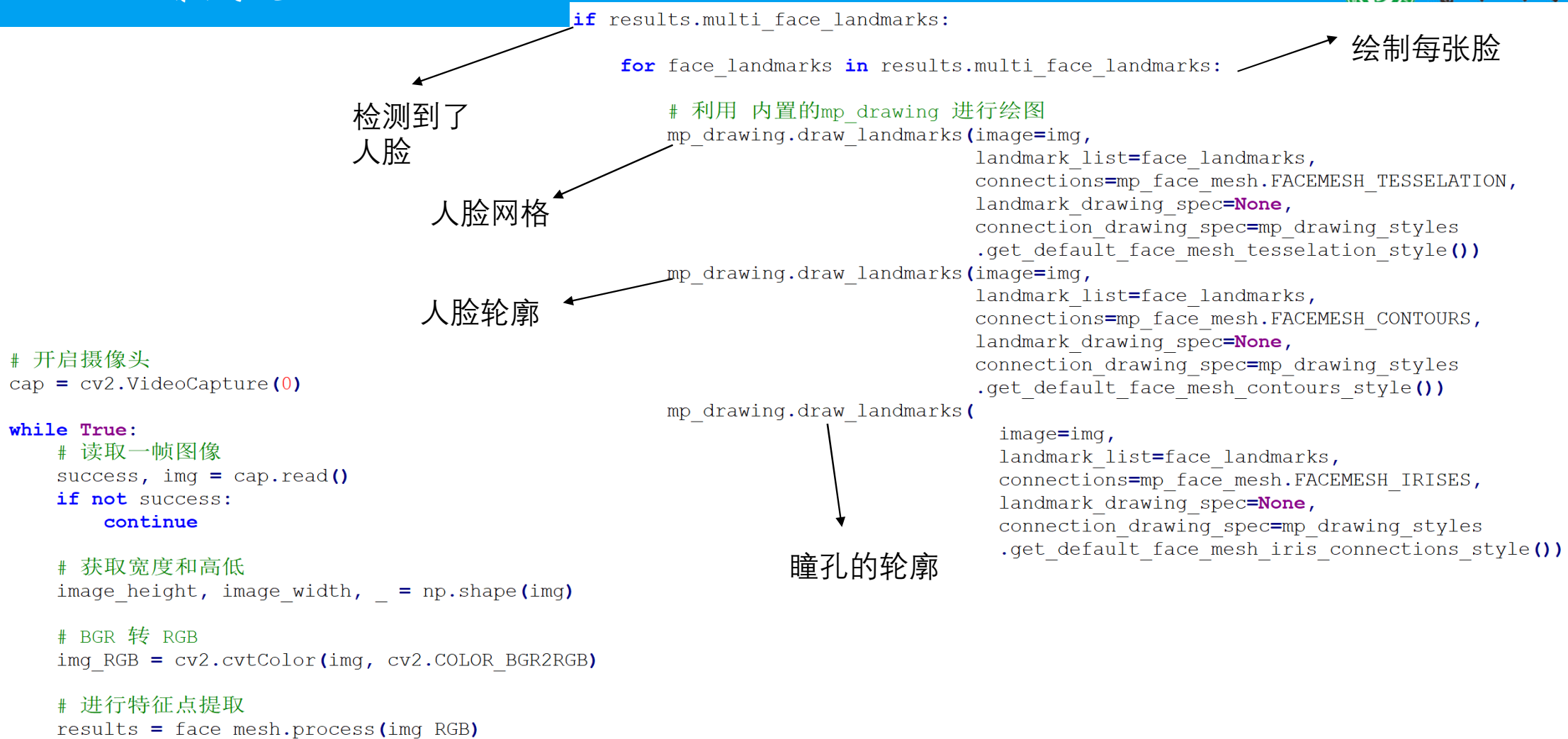
    face_mesh = mp_face_mesh.FaceMesh(static_image_mode=False,
                                       max_num_faces=1,
                                       refine_landmarks=True,
                                       min_detection_confidence=0.5,
                                       min_tracking_confidence=0.5)

    # 构建绘图对象
    mp_drawing = mp.solutions.drawing_utils
    mp_drawing_styles = mp.solutions.drawing_styles
```

False : 处理视频  
True: 处理单张图片

静态图片  
不用设置

内置的绘图对象



```
# 自行计算478个关键点的坐标 并绘制
if face_landmarks:
    # 计算关键点坐标
    for i in range(478):
        pos_x = int(face_landmarks.landmark[i].x * image_width)
        pos_y = int(face_landmarks.landmark[i].y * image_height)

        cv2.circle(img, (pos_x,pos_y), 3, (0,255,0),-1)
```

归一化坐标 x,y,z  
z 相对x坐标的立体坐标

```
cv2.imshow("face-mesh",img)

key = cv2.waitKey(1) & 0xFF

# 按键 "q" 退出
if key == ord('q'):
    break
cap.release()
```