**项目计划书**

项目名称： “智能广告屏”

参赛成员： 代登辉、赵永乐、

指导老师：

目录

[一、 需求分析 1](#_Toc26023348)

[二、 概要设计 1](#_Toc26023349)

[三、 详细设计 1](#_Toc26023350)

[四、 测试报告 5](#_Toc26023351)

[性别年龄识别准确性 5](#_Toc26023352)

[广告投放流畅性 5](#_Toc26023353)

[五、 安装及说明 5](#_Toc26023354)

[Win10所需软件配置： 5](#_Toc26023355)

[树莓派配置： 6](#_Toc26023356)

# 一、 需求分析

如今,广告投放可以主要分为两种形式,第一种是线下广告,如海报、广告机轮播图片、屏幕播放视频广告等。另一种是线上广告,如各大网站通过广告平台投放的广告。各种广告投放方式都有各自的缺点,尤其是线下广告。线下广告一般是一种张贴布置的形式,虽然位置十分灵活,但是缺点是显而易见的,没有目的性,互动性差,不能吸引人的注意。

因为发现目前线下广告投放不能达到理想效果,在投放广告的过程中带有一定的盲目性,我们决定开发一个能够针对特定人群精准投的广告系统。项目初步需求是研发一个广告机实体,广告机上带有摄像头,通过人脸识别技术获取人群数据,根据不同人的属性数据定向投放广告,并且返回广告投放效果,达到更为精准的投放广告的目的。实体机可广泛应用于商场、酒店、展会、小区等场合,有效增强互动性,吸引人流,给用户带来新奇有趣的体验。

# 二、 概要设计

本项目采用opencv开发，OpenCV是一个基于BSD许可（开源）发行的跨平台计算机视觉库，可以运行在Linux、Windows、Android和Mac OS操作系统上。它轻量级而且高效——由一系列 C 函数和少量 C++ 类[构成](https://baike.baidu.com/item/%E6%9E%84%E6%88%90/103686" \t "_blank)，同时提供了Python、Ruby、MATLAB等语言的接口，实现了[图像处理](https://baike.baidu.com/item/%E5%9B%BE%E5%83%8F%E5%A4%84%E7%90%86/294902" \t "_blank)和计算机视觉方面的很多通用算法。本项目使用python语言开发。

项目基于Caffe的预训练模型实现年龄与性别预测，根据预测的结果投放相对应的广告。如检测到小孩则投放玩具类的广告，检测到女士则投放一些化妆品的广告。即根据其性别年龄投放出相应阶段感兴趣的广告，以增强广告投放的效果。

本项目基于树莓派为硬件开发，采用USB摄像头采集人脸数据，普通显示屏作为广告屏。在win10系统设计开发后移植到树莓派（Linux）平台上。

# 三、 详细设计

## 代码设计

（1）人脸区域提取

def getFaceBox(net, frame, conf\_threshold=0.7):

    frameOpencvDnn = frame.copy()

    frameHeight = frameOpencvDnn.shape[0]  # 高就是矩阵有多少行

    frameWidth = frameOpencvDnn.shape[1]  # 宽就是矩阵有多少列

    blob = cv.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123], True, False)

    #  blobFromImage(image[, scalefactor[, size[, mean[, swapRB[, crop[, ddepth]]]]]]) -> retval  返回值   # swapRB是交换第一个和最后一个通道   返回按NCHW尺寸顺序排列的4 Mat值

    net.setInput(blob)

    detections = net.forward()  # 网络进行前向传播，检测人脸

    bboxes = []

    for i in range(detections.shape[2]):

        confidence = detections[0, 0, i, 2]

        if confidence > conf\_threshold:

            x1 = int(detections[0, 0, i, 3] \* frameWidth)

            y1 = int(detections[0, 0, i, 4] \* frameHeight)

            x2 = int(detections[0, 0, i, 5] \* frameWidth)

            y2 = int(detections[0, 0, i, 6] \* frameHeight)

            bboxes.append([x1, y1, x2, y2])  # bounding box 的坐标

            cv.rectangle(frameOpencvDnn, (x1, y1), (x2, y2), (0, 255, 0), int(round(frameHeight / 150)),

                         8)  # rectangle(img, pt1, pt2, color[, thickness[, lineType[, shift]]]) -> img

    return frameOpencvDnn, bboxes

（2）性别与年龄预测

faceProto = "/home/pi/IAD/models/opencv\_face\_detector.pbtxt"

faceModel = "/home/pi/IAD/models/opencv\_face\_detector\_uint8.pb"

ageProto = "/home/pi/IAD/models/age\_deploy.prototxt"

ageModel = "/home/pi/IAD/models/age\_net.caffemodel"

genderProto = "/home/pi/IAD/models/gender\_deploy.prototxt"

genderModel = "/home/pi/IAD/models/gender\_net.caffemodel"

src = cv.imread("/home/pi/IAD/AD/SSpicture/1.jpg") #屏幕保护，即没人脸时显示

# 模型均值

MODEL\_MEAN\_VALUES = (78.4263377603, 87.7689143744, 114.895847746)

ageList = ['(0-2)', '(4-6)', '(8-12)', '(15-20)', '(25-32)', '(38-43)', '(48-53)', '(60-100)']

genderList = ['Male', 'Female']

# 加载网络

ageNet = cv.dnn.readNet(ageModel, ageProto)

genderNet = cv.dnn.readNet(genderModel, genderProto)

# 人脸检测的网络和模型

faceNet = cv.dnn.readNet(faceModel, faceProto)

 if not bboxes:

            print("No face Detected, Checking next frame")

            cv.imshow("advertising",src)

            continue

        for bbox in bboxes:

            face = frame[max(0, bbox[1] - padding):min(bbox[3] + padding, frame.shape[0] - 1),

                   max(0, bbox[0] - padding):min(bbox[2] + padding, frame.shape[1] - 1)]

            #

            blob = cv.dnn.blobFromImage(face, 1.0, (227, 227), MODEL\_MEAN\_VALUES, swapRB=False)

            genderNet.setInput(blob)   # blob输入网络进行性别的检测

            genderPreds = genderNet.forward()   # 性别检测进行前向传播

            gender = genderList[genderPreds[0].argmax()]   # 分类  返回性别类型

            ageNet.setInput(blob)

            agePreds = ageNet.forward()

            age = ageList[agePreds[0].argmax()]

            label = "{},{}".format(gender, age)

            print(label)

（3）投放广告模块

cv.namedWindow("advertising",cv.WINDOW\_AUTOSIZE)  #定义窗口 /以一个参数是窗口名称 //第二个参数是窗口形式

cv.resizeWindow("advertising",1000,1500)

def read\_file(path\_file\_number):

    if path\_file\_number == []:

      print("未找到图片文件")

    else:

        num =random.randint(0,len(path\_file\_number)-1)

        src = cv.imread(path\_file\_number[num])

        cv.imshow("advertising",src)

        cv.waitKey(3000)

###################################################################################

def video\_input(path\_file\_video\_number):  #定义视频函数

    if path\_file\_video\_number == []:

      print("未找到视频文件")

    else:

        num =random.randint(0,len(path\_file\_video\_number)-1)

        capture = cv.VideoCapture(path\_file\_video\_number[num]) #打开摄像头 /参数是数字表示摄像头 也可以是视频路径

        while(True):  #循环 /无限循环

            ret , frame = capture.read()  #获取视频信息 \第一个返回值表示视频状态 \第二个返回值表示每一帧画面

            if ret == False:

                break

            cv.imshow("advertising",frame)  #展示图片信息

            cv.waitKey(50)

##################################对应女性的广告投放################################

def aaa(path\_file\_video\_number,path\_file\_number,a):

    if a%10:

        video\_input(path\_file\_video\_number)

    if a//10:

        read\_file(path\_file\_number)

## 程序设计流程图

# 四、 测试报告

## 性别年龄识别准确性

## 广告投放流畅性

# 五、 安装及说明

在树莓派上配置好环境没有代码提示编写程序比较麻烦，因此代码开发在win10上进行设计，采用FileZIllaClient这款软件把代码及相关数据上传到树莓派。

## Win10所需软件配置：

IDE ：Anaconda 内置 VS Code，

环境配置软件：Anaconda Navigator (Anaconda)，

终端模拟软件：Xshell，

[FTP](https://baike.baidu.com/item/FTP/13839) 客户端软件：FileZilla Client

SD内存卡修复工具：SD Card Formatter

系统烧录工具：Win32DiskImage

## 树莓派配置：

开启ssh

莓派终端安装xrdp服务器

树莓派安装FTP Server，及其初始配置

树莓派换源及更新

配置摄像头

配置python-opencv环境