# 人脸识别部署文档

# 前言

由于ip摄像头是rtsp协议视频，如果要在web端显示需要安装插件或者转为flash或者http的类型，由于插件在chrome浏览器无法使用，此处使用的是使用的是转为rtmp协议流。

# 部署

## 添加防火墙策略

如果防火墙开启 开启允许1935、8088、8089端口

sudo firewall-cmd --zone=public --add-port=1935/tcp --permanent

sudo firewall-cmd --zone=public --add-port=8088/tcp --permanent

sudo firewall-cmd --zone=public --add-port=8089/tcp --permanent

sudo firewall-cmd --reload

## 部署face\_web

由于face\_web的项目中有些类库在linux安装可以正常使用，故仅可以在linux环境下部署。

登录机器：192.168.0.203 face\_user face\_user

### 创建用户

使用root账号：

useradd face\_user

passwd face\_user

将face\_user加入/etc/sudoers 中

chmod a+w /etc/sudoers

将下面代码追加到文件末尾

%face\_user ALL=(ALL) NOPASSWD: ALL

然后：chmod a-w /etc/sudoers

### 下载代码

从svn中下载项目的代码 face\_web 将代码放入：/home/face\_user 中

### 安装python依赖

1、在项目的根目录执行创建虚拟执行环境

cd /home/face\_user/face\_web

python安装 pip

sudo python get-pip.py

sudo pip install virtualenv

初始化python虚拟机

virtualenv venv

激活虚拟机：

. venv/bin/activate

2、安装系统依赖：

sudo yum install cmake gcc gcc-c++ g++ python-devel boost-devel boost libXrender

注意：

如果系统为centos 7 系统可以直接使用venv.zip解压即可，不用再安装python模块。

3、安装requirements.txt里的python依赖。使用pip安装即可。

pip install scikit-learn flask ConfigParser baidu-aip opencv-python face\_recognition\_models face\_recognition scipy

4、安装python依赖时间有点长，可以接下来安装和启动http视频转换服务：

cd /home/face\_user/face\_web

### 使用nginx+FFMPEG转码

#### Nginx的安装、配置和使用

1、nginx-rtmp-module模块

为了增加对rtmp的支持，下载nginx-rtmp-module

https://github.com/arut/nginx-rtmp-module/archive/v1.2.1.tar.gz

解压到/root/nginx-rtmp-module

配置和编译nginx的时候要用到。

2、安装nginx的依赖

yum -y install openssl openssl-devel

yum -y install pcre-devel

3、下载nginx-1.10.3.tar.gz源码并解压至/root/ http://nginx.org/download/nginx-1.10.3.tar.gz

tar zvxf /root/nginx-1.10.3.tar.gz

mv /root/nginx-1.10.3/ /root/nginx

进入到nginx解压路径后，使用下面的命令配置nginx及其module

cd /root/nginx/

./configure --prefix=/usr/local/nginx --add-module=/root/nginx-rtmp-module --with-http\_ssl\_module

4、编译并安装

make

make install

5、配置nginx对rtmp的支持

vim /usr/local/nginx/conf/nginx.conf

加入下面的配置

rtmp {

server {

listen 1935;

application hls {

live on;

hls on;

hls\_path /tmp/hls;

hls\_fragment 1s;

#hls\_playlist\_length 3s;

}

}

}

注释：

live on; 开启实时

hls on; 开启hls

hls\_path;

          http://xxx/hls這個地址可以用来推流和拉流，其中xxx是你的ip地址或域名。

          它会被映射到/tmp/hls本地路径下。

          ts文件存放路径，使用绝对路径

hls\_fragment 5s; 每个TS文件包含5秒的视频内容

7、nginx开机启动

vi /usr/lib/systemd/system/nginx.service

[Unit]

Description=nginx - high performance web server

Documentation=http://nginx.org/en/docs/

After=network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

PIDFile=/usr/local/nginx/logs/nginx.pid

ExecStartPre=/usr/local/nginx/sbin/nginx -t -c /usr/local/nginx/conf/nginx.conf

ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf

ExecReload=/bin/kill -s HUP $MAINPID

ExecStop=/bin/kill -s QUIT $MAINPID

PrivateTmp=true

[Install]

WantedBy=multi-user.target

启动服务并添加开机启动

systemctl start nginx.service

systemctl enable nginx.service

#### FFMPEG 的安装、配置和使用

1、下載

cd /root/

wget http://ffmpeg.org/releases/ffmpeg-3.4.tar.bz2

解压缩

bzip2 -d ffmpeg-3.4.tar.bz2

tar -xvf ffmpeg-3.4.tar

2、安装依赖

yum install nasm

3、安装&配置

配置

cd ffmpeg-3.4/

./configure --enable-shared --prefix=/usr/local/ffmpeg

make

可能需要几分钟

make install

现在ffmpeg已经被安裝到了/usr/local/ffmpeg路徑下。

添加动态库文件搜索路径

vim /etc/ld.so.conf

把/usr/local/ffmpeg/lib路径添加到ld.so.conf中的最後一行。

使用下面的命令让修改生效

ldconfig

4、使用

确定视频源没问题后使用下面的命令转码

./ffmpeg -f rtsp -rtsp\_transport tcp -i rtsp://show\_face:face\_show@863@192.168.0.215:554/h264/ch1/main/av\_stream -codec copy -f flv -an rtmp://bdata007:1935/hls/test -s 1280\*720

注释：

-f rtsp 输入流是rtsp协议

-rtsp\_transport tcp 输入流是tcp协议

-i rtsp://admin:nuoke123456@192.168.2.226:554/Streaming/Channels/101 指定输入流

-f flv 输出流为flv格式

-an rtmp://192.168.109.129:1935/hls/test 指定推送地址

如果推送成功的话/tmp/hls路径下会生成多个tls文件和test.m3u8文件。

5、编写启动脚本：

在face\_web下

vi start\_proxy.sh

nohup /usr/local/ffmpeg/bin/ffmpeg -f rtsp -rtsp\_transport tcp -i rtsp://show\_face:face\_show@863@192.168.0.215:554/h264/ch1/main/av\_stream -codec copy -f flv -an rtmp://bdata007:1935/hls/test -s 1280\*720 >> ffmpeg.log 2>&1 &

## 启动服务

### 启动准备

1. 由于此系统的人脸库在训练时候会使用综合运营管理平台中的头像，所以为了提高准确率需要用户在863软件部落中上传清晰的本人照片；
2. 目前模型不会定时训练，可以加入定时训练来实时获取最新图片；

3、管理员可以通过人脸添加接口来额外的新增多张照片提高识别率。

### 启动

Python依赖安装完毕后

cd /home/face\_user/face\_web

首先保证摄像头连接网线和电源正常！！！

1. 训练模型：sh train.sh[./venv/bin/python face\_model\_util.py]

2、启动服务： sh start.sh[./venv/bin/python application.py]

3、启动人脸识别服务： sh start\_check.sh[./venv/bin/python face\_check.py]

### 服务运维

#### 开启服务

首先保证摄像头连接网线和电源正常！！！

启动转换服务：start\_proxy.sh

启动web服务：start.sh

启动人脸识别服务：start\_check.sh

或者直接执行:star\_all.sh

另外有相应的stop\_proxy.sh、stop.sh、stop\_check.sh

## 服务访问

首先保证摄像头连接网线和电源正常并服务开启！！！

1、访问http://192.168.0.203:8089/show\_face访问可视化界面。

2、访问http://192.168.0.203:8089/add\_face 来增加人脸照片。