# 密集人数统计应用（Python）

开发者将本应用部署至Atlas 200 DK或者AI加速云服务器上实现对本地mp4文件或者RTSP视频流进行解码，对视频帧中的人群图像进行人头数量的预测，并将预测的结果信息发送至Presenter Server端进行展示。

当前分支中的应用适配**1.3.0.0**与**1.31.0.0**版本的[DDK&RunTime](https://ascend.huawei.com/resources" \o " )。

前提条件

部署此Sample前，需要准备好以下环境：

* 已完成Mind Studio的安装。
* 已完成Atlas 200 DK开发者板与Mind Studio的连接，交叉编译器的安装，SD卡的制作及基本信息的配置等。
* 由于需要配置开发板联网，默认设置为USB连接，开发板地址为192.168.1.2

软件准备

运行此Sample前，需要按照此章节获取源码包，进行相关的环境配置并准备模型文件。

获取源码包。

将[https://gitee.com/Atlas200DK/sample-crowdcounting-python](https://gitee.com/Atlas200DK/sample-fasterrcnndetection-python" \o " )仓中的代码以Mind Studio安装用户下载至Mind Studio所在Ubuntu服务器的任意目录，例如代码存放路径为：/home/ascend/sample-crowdcounting-python。

以Mind Studio安装用户登录Mind Studio所在Ubuntu服务器，确定当前使用的DDK版本号并设置环境变量DDK\_HOME，tools\_version，NPU\_DEVICE\_LIB和LD\_LIBRARY\_PATH。

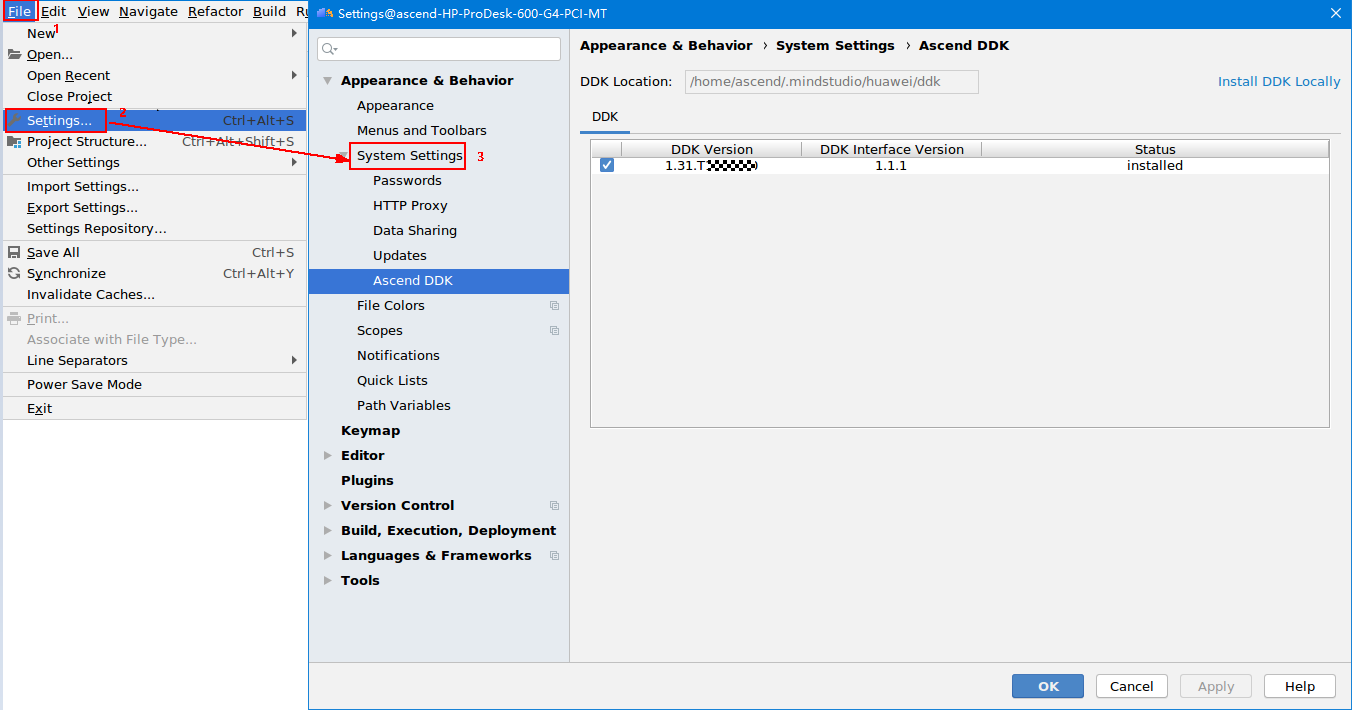
1. 查询当前使用的DDK版本号。

可通过Mind Studio工具查询，也可以通过DDK软件包进行获取。

* 使用Mind Studio工具查询。

在Mind Studio工程界面依次选择“File > Settings > System Settings > Ascend DDK”，弹出如图 DDK版本号查询所示界面。

DDK版本号查询



其中显示的**DDK Version**就是当前使用的DDK版本号，如**1.31.T15.B150**。

* 通过DDK软件包进行查询。

通过安装的DDK的包名获取DDK的版本号。

DDK包的包名格式为：**Ascend\_DDK-{software version}-{interface version}-x86\_64.ubuntu16.04.tar.gz**

其中**software version**就是DDK的软件版本号。

例如：

DDK包的包名为Ascend\_DDK-1.31.T15.B150-1.1.1-x86\_64.ubuntu16.04.tar.gz，则此DDK的版本号为1.31.T15.B150。

1. 设置环境变量。

**vim ~/.bashrc**

执行如下命令在最后一行添加DDK\_HOME及LD\_LIBRARY\_PATH的环境变量。

**export tools\_version=*1.31.X.X***

**export DDK\_HOME=$HOME/.mindstudio/huawei/ddk/$tools\_version/ddk**

**export NPU\_DEVICE\_LIB=$DDK\_HOME/../RC/host-aarch64\_Ubuntu16.04.3/lib**

**export LD\_LIBRARY\_PATH=$DDK\_HOME/lib/x86\_64-linux-gcc5.4**



* ***1.31.X.X***是[1](#li61417158198" \o " )中查询到的DDK版本号，需要根据查询结果对应填写，如**1.31.T15.B150**
* 如果此环境变量已经添加，则此步骤可跳过。

输入**:wq!**保存退出。

执行如下命令使环境变量生效。

**source ~/.bashrc**

从模型仓中下载好原始crowd\_counting.pb模型文件。

1.3.0.0版本的MindStudio模型转换请参考链接：

1.31.0.0版本的MindStudio模型转换请参考链接：

将转换好的模型文件（crowd\_counting.om文件）上传到[步骤1](#d0e163" \o " )中源码所在路径下的“**sample-crowdcounting-python/**”目录下。

----结束

环境配置

配置开发板联网。

1. 不断开usb连接的情况下，开发板通过网线与可以上网的网口相连。
2. 在Mind Studio所在Ubuntu服务器中，以HwHiAiUser用户SSH登录到Host侧。

**ssh HwHiAiUser@192.168.1.2**

切换到root用户，开发板中root用户默认密码为**Mind@123**。

**su root**

打开interfaces配置文件。

**vim /etc/network/interfaces**

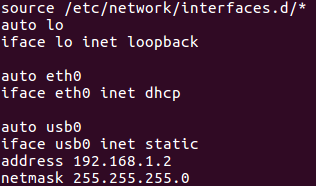
配置dhcp，把eth0的配置修改为如下两行。

**auto eth0**

**iface eth0 inet dhcp**

如图1-2所示，输入:wq!保存退出。

开发板interface文件配置



1. 执行以下命令重启开发板，使配置生效。

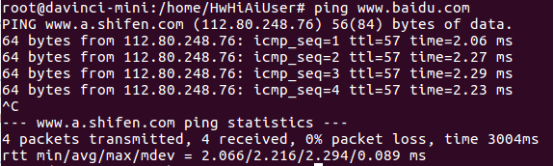
**reboot**

1. 等开发板四个灯常亮时则证明已经完成重启，此时可以ping外网验证网络是否通畅，如果ping通则已经配置成功；如果ping失败，则执行以下命令。成功后如图1-3所示。

**ifdown eth0**

**ifup eth0**

开发板联网配置成功



安装环境依赖。

1. 在开发板的root用户下更换源。

**vim /etc/apt/sources.list**

把原有源更换为arm源，可用的国内arm源有中科大源和清华源等。



arm源可参考[https://bbs.huaweicloud.com/forum/thread-37023-1-1.html](https://bbs.huaweicloud.com/forum/thread-37023-1-1.html" \o " )

源更新后，执行以下命令更新软件列表。

**apt-get update**

1. 安装相关依赖。

apt-get install python-setuptools python-dev build-essential python-pip

pip install numpy==1.11.2 enum34==1.1.6 future==0.17.1 funcsigs==1.0.2 enum future unique enum34 funcsigs protobuf



pip install安装有报错“SSLError”时，请使用：pip install --trusted-host pypi.org --trusted-host files.pythonhosted.org numpy==1.11.2 enum34==1.1.6 future==0.17.1 funcsigs==1.0.2 enum future unique enum34 funcsigs protobuf 安装依赖，表示可信赖的主机解决问题。

1. 安装hiai。

执行脚本安装hiai，hiai的安装脚本可以在码云的tools仓中获取，将脚本内容复制到开发板中。

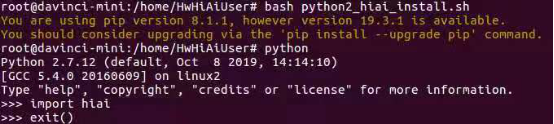
安装脚本路径：**https://gitee.com/HuaweiAscend/tools/blob/master/python2\_hiai\_install.sh**。

在开发板的root用户下执行脚本进行hiai库安装：

**bash python2\_hiai\_install.sh**

安装后如图1-4所示，则为安装成功。

hiai安装成功验证



安装opencv。

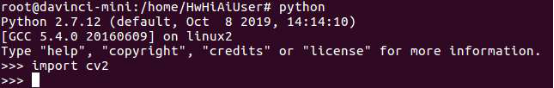
root用户下安装opencv Python库。

命令示例：

**apt-get install python-opencv**

安装后如图1-5所示，则为安装成功。

opencv安装成功验证



----结束

部署

**启动Presenter Server**

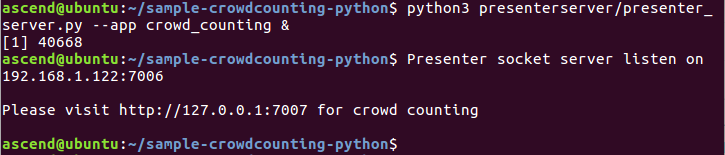
进入工程目录，执行如下命令在后台启动密集人数统计应用的Presenter Server主程序。

python3 presenterserver/presenter\_server.py --app crowd\_counting &

说明：

“presenter\_server.py“在当前目录的“presenterserver“目录下，可以在此目录下执行python3 presenter\_server.py -h或者python3 presenter\_server.py --help查看“presenter\_server.py“的使用方法。

如图4所示，表示presenter\_server的服务启动成功。



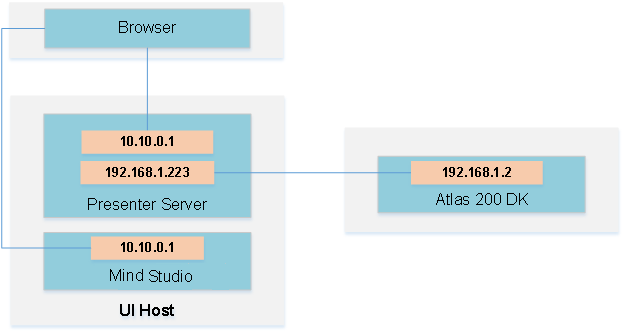
使用上图提示的URL登录Presenter Server，仅支持Chrome浏览器。IP地址为[步骤2](#li9634105881418" \o " )中输入的IP地址，端口号默为7007，如下图所示，表示Presenter Server启动成功。

主页显示



Presenter Server、Mind Studio与Atlas 200 DK之间通信使用的IP地址示例如下图所示：

IP地址示例



其中：

* Atlas 200 DK开发者板使用的IP地址为192.168.1.2（USB方式连接）。
* Presenter Server与Atlas 200 DK通信的IP地址为UI Host服务器中与Atlas 200 DK在同一网段的IP地址，例如：192.168.1.223。
* 通过浏览器访问Presenter Server的IP地址本示例为：10.10.0.1，由于Presenter Server与Mind Studio部署在同一服务器，此IP地址也为通过浏览器访问Mind Studio的IP。

crowd-counting-python应用支持解析本地视频和RTSP视频流。

* 如果需要解析本地视频，需要将视频文件传到Host侧。

例如将视频文件crowd.mp4上传到Host侧的“/home/HwHiAiUser/sample”目录下。



支持H264与H265格式的MP4文件，如果MP4文件需要剪辑，建议使用开源工具ffmpeg，使用其他工具剪辑的视频文件ffmpeg工具可能不支持解析。

* 如果仅解析RTSP视频流，本步骤可跳过。

将工程拷贝到开发者板上

在Mind Studio所在Ubuntu服务器中，在任一目录下

执行以下命令将工程拷贝到开发者板上的HwHiAiUser用户目录下：

scp -r /home/ascend/sample-crowdcounting-python/ HwHiAiUser@192.168.1.2:/home/HwHiAiUser/

输入密码即可完成拷贝，默认密码为：Mind@123

----结束

运行

在Mind Studio所在Ubuntu服务器中，以HwHiAiUser用户SSH登录到Host侧。

**ssh HwHiAiUser@192.168.1.2**

在HwHiAiUser用户下进入应用代码所在目录。

**cd ~/sample-crowdcounting-python**

执行应用程序。

**python main.py** *channel*

* *channel:* 输入的视频文件名或者RTSP流地址

视频文件运行的命令示例如下所示：

**python main.py /home/HwHiAiUser/sample/crowd.mp4**

RTSP视频流的命令示例如下所示：

**python main.py rtsp://192.168.2.37:554/cam/realmonitor?channel=1&subtype=0**

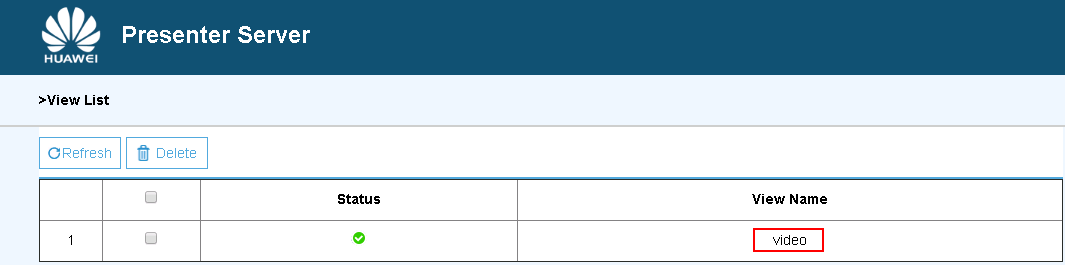


可使用ctrl+c停止程序

使用启动Presenter Server服务时提示的URL登录 Presenter Server 网站，详细可参考[启动Presenter Server](#li156931456596" \o " )。

等待Presenter Agent传输数据给服务端，单击“Refresh”刷新，当有数据时相应的Channel 的Status变成绿色，如图1-10所示。

Presenter Server界面





* Presenter Server最多支持10路Channel同时显示，每个 *presenter\_view\_app\_name* 对应一路Channel。
* 由于硬件的限制，每一路支持的最大帧率是20fps，受限于网络带宽的影响，帧率会自动适配为较低的帧率进行展示。

单击右侧对应的View Name链接，比如上图的“video”，查看结果。

----结束

**停止Presenter Server服务**

Presenter Server服务启动后会一直处于运行状态，若想停止密集人数统计应用对应的Presenter Server服务，可执行如下操作。

以Mind Studio安装用户在Mind Studio所在服务器中执行如下命令查看密集人数统计应用对应的Presenter Server服务的进程。

ps -ef | grep presenter | grep crowd\_counting

ascend@ascend-HP-ProDesk-600-G4-PCI-MT:~/sample-crowdcounting$ ps -ef | grep presenter | grep crowd\_counting

ascend 7701 1615 0 15:10 pts/8 00:00:00 python3 presenterserver/presenter\_server.py --app crowd\_counting

如上所示7701 即为密集人数统计应用对应的Presenter Server服务的进程ID。

若想停止此服务，执行如下命令：

kill -9 7701

----结束