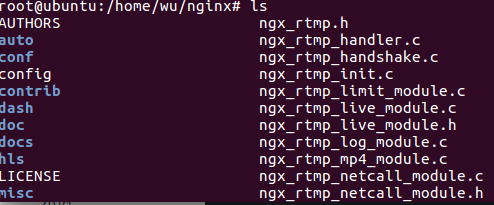
# 从零实现ffmpeg添加去雾filter

-- 摄像头采集图像，并通过web播放去雾后的视频流（一）

1. **配置nginx+rtmp**
   1. 下载nginx并解压 http://nginx.org/



* 1. 利用git clone指令下载nginx-rtmp-module

git clone git@github.com:arut/nginx-rtmp-module.git



* 1. 合并两个文件夹



* 1. 进行编译安装

./configure --add-module=./nginx-rtmp-module

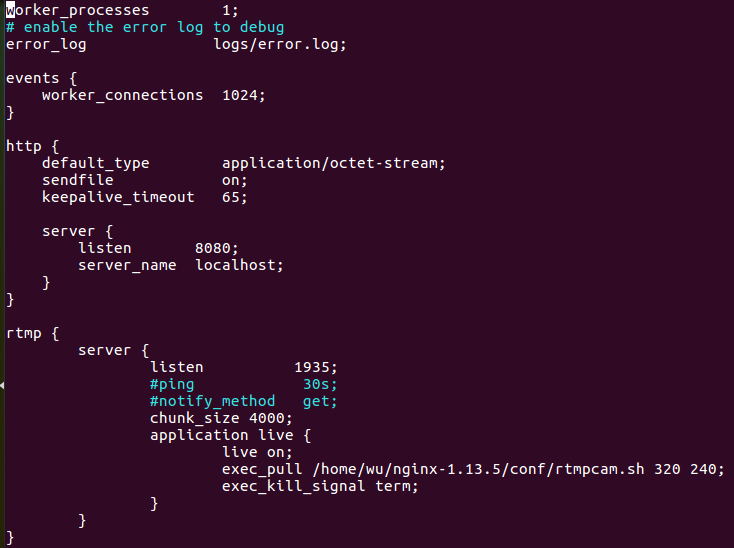
Make –j2

sudo make install

最后编译安装完成。

* 1. rtmp模块的配置

将rtmp的配置加入nginx的.conf文件中<https://github.com/Xingtao/RpiVisi/tree/master/nginx-conf>



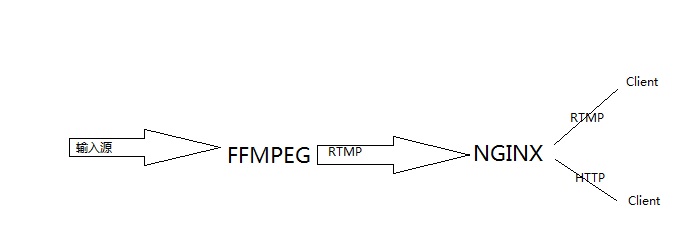
其他脚本作用:

rtmpcam.sh—推送流

startnginx,stopnginx—开始，结束nginx

* 1. rtmp模块的作用

nginx配合ffmpeg做流媒体服务器的原理是: nginx通过rtmp模块提供rtmp服务, ffmpeg推送一个rtmp流到nginx, 然后客户端通过访问nginx来收看实时视频流. HLS也是差不多的原理,只是最终客户端是通过HTTP协议来访问的,但是ffmpeg推送流仍然是rtmp的.



4.7nginx配置文件详解

worker\_processes 2；#nginx的进程数，我的虚拟机为双核可以设置成2

# enable the error log to debug

error\_log logs/error.log;/#在置顶错误日志的存放路径，有[debug|info|notice|warn|error|crit]

#工作模式和连接数上限

events {

#工作进程的最大连接数量，根据硬件调整，和前面工作进程配合起来用，尽量大但是别把CPU跑到100就行。每个进程允许的最多连接数，理论上每台nginx服务器的最大连接数为work\_processes \*worker\_connections keepalive\_timeout 60;;

worker\_connections 1024;

}

#设定http服务器

http {

#默认文件类型 设置文件使用默认的MIME-type类型

default\_type application/octet-stream;

#用来指定nginx是否调用sendfile函数（zero copy方式）来输出文件，对于普通应用，必须设为on.如果用来进行下载等应用磁盘IO重负载应用，可设置为off，一平衡磁盘与网络IO处理速度，降低西永uptime。

sendfile on;

#给客户端分配keep-alive链接超时时间，服务器将在这个超时时间过后关闭链接。

keepalive\_timeout 65;

#虚拟主机的配置

server {

#监听端口

listen 8080;

#于明克以后多个，用空格隔开

server\_name localhost;

}

}

rtmp {#根，保存所有RTMP的块

server {#声明一个RTMP实例

#添加一个监听端口

listen 1935;

#ping 30s;ping间隔，检查活动链接的写应该能，发送一个特殊的包到远程链接，然后再ping\_timeout的指定时间内期待一个回复，如果在这个时间内没收到ping回复则断开。

#notify\_method get;设置HTTP方法通知。默认是带有application/x-www-form-urlencoded的POST内容类型，在一些情况下GET更好，例如想在http的{的部分处理调用}

#刘整合的最大块的大小，默认值为4096，这个值越大CPU负载越小，值不能低于128

chunk\_size 4000;

#创建一个rtmp应用，application的模式并不类似于http location

application live {

live on;

exec\_pull /home/wu/nginx-1.13.5/conf/rtmpcam.sh 320 240;

#设置进程中止信号

exec\_kill\_signal term;

}

}

}