ADASI Test

Candidate: David Esteban Imbajoa Ruiz

Requirements

The requirements to run the code are:

Gstreamer 1.0

OpenCV 3.4.2

cairo

CMake 3.5.1

C++ 17

VLC >3.0.6

Setup

In order to set up, compile and run the code you have to follow the next steps:

- Install Gstreamer 1.0
- Install OpenCV 3.4.2
- Install CMake 3.5.1
- Install VLC >3.0.6
- Clone the repository:
    - Clone the repository using the next link:

      https://github.com/deivy311/ADASI_Test

  - If you don't have the repository, you can download the zip file, attached to the email named ADASI_Test.zip and unzip it.
- Compile the code.

  Go to the folder where you clone the repository and run the next commands:

```
mkdir build

cd build

cmake ../

make
```

You should see something like the next image.

Uubuntu 22.04

For Linux these libraries must be installed

Follow the steps here: https://gstreamer.freedesktop.org/documentation/installing/on-linux.html?gi-language=c

And in addition, run the following commands:

sudo apt install libgirepository1.0-dev

sudo apt install gir1.2-gst-rtsp-server-1.0

auso apt install libgstreamer-plugins-good1.0-0 libgstreamer-plugins-good1.0-dev gstreamer1.0-plugins-good

sudo apt install gstreamer1.0-rtsp

sudo apt-get install libgstrtspserver-1.0-dev

sudo apt-get install libvdpau-dev vdpauinfo

sudo apt-get install libvdpau-va-gl1

Running the code

Go to the folder where you compile the code and run the executable file, for this a better explanation is in this section.

There are a total of 5 executables files, from de point_0 to point_5, each one of them has a different functionality. The parameters from Point 0 to Point 2_3 are the same, the only difference is the functionality of each one. And the parameters from Point 4 to Point 5 are the same.

In any case you must be placed in the root folder/build/bin in order to run the following commands

For **point_0, point_1 and point_2_3** the parameters are:

Source type: Defines if we want to stream the camera or a file, by default is **autovideosrc** there are only two posible options:

videotestsrc or **autovideosrc**

Examples:

The input example for both are:

Camera case:

./point_1 -s videotestsrc

File case:

./point_0 -s autovideosrc

Host: Defines the host to stream by default is "127.0.0.1"

Examples:

Localhost case:

```
sudo ./point_0 -h 127.0.0.1
```

Port: Defines the port to stream by default is "5002"

Examples:

By default case:

```
sudo ./point_0 -p 5002
```

For **point_4, point_5 and point_2_3** the parameters are same as before but there a couple more:

RTSP port: Defines the RTSP port to stream by default is "5001"

Examples:

By default case:

```
sudo ./point_4 -p_rtsp 5001
```

RTSP video file path: Defines the RTSP video file path to stream by defualt is "**../Files/video_test_2.mp4**", these videos are in the Files folder and also in the build/bin folder.

Examples:

By default case:

```
sudo ./point_5 -f ../Files/video_test_2.mp4
```

Results

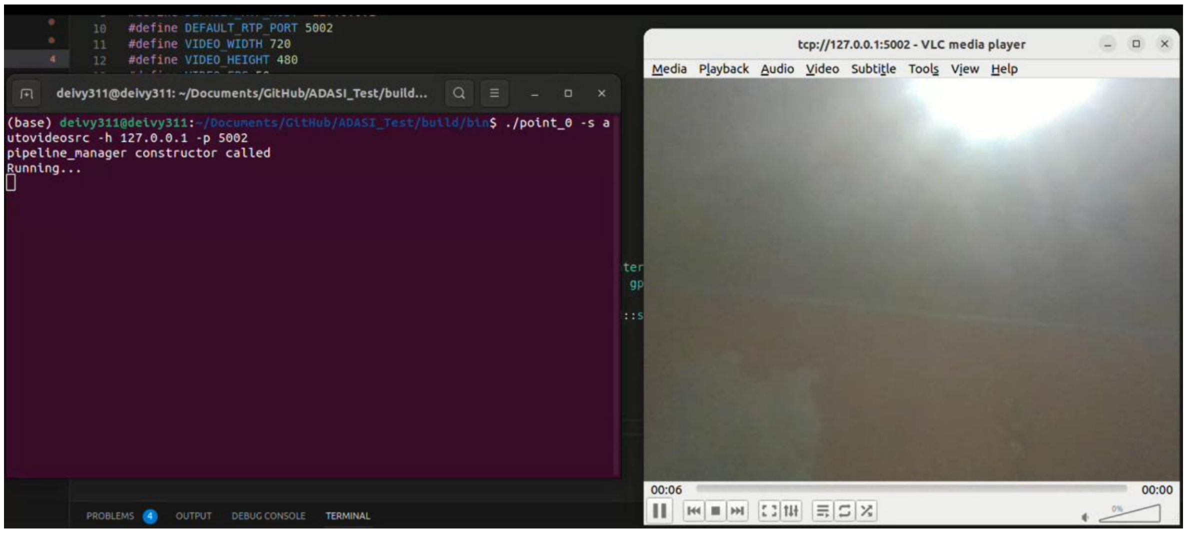
The executables **point_0 and point_1**, can show the results for the literal 1, it is a simple RTP pipeline where can stream a video file or a camera.

Case 1.1: RTP with no overlay

Given the next command input:

```
./point_0 -s autovideosrc -h 127.0.0.1 -p 5002
```

We can see the next results:



```
#define DEFAULT_RTP_PORT 5002
#define VIDEO_WIDTH 720
#define VIDEO_HEIGHT 480
```

```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_0 -s a
utovideosrc -h 127.0.0.1 -p 5002
pipeline_manager constructor called
Running...
```

deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
tcp://127.0.0.1:5002 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
00:06
00:00
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL

In order to see the results by yourself you have to open VLC and go to Media -> Open Network Stream and put the next address:

tcp://127.0.0.1:5002

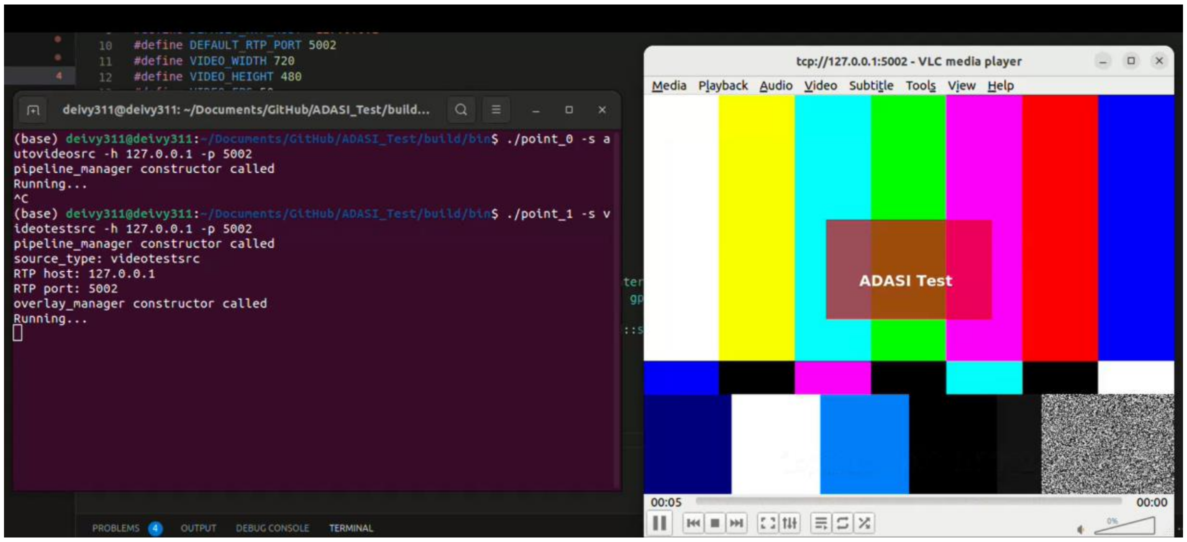
The video related to this is in the Folder Results, named point_1_no_overlay.webm which means it only streams video without any overlay.

Case 1.2: RTP with overlay, demo file

Given the next command input:

```
./point_1 -s videotestsrc -h 127.0.0.1 -p 5002
```

We can see the next results:



```
#define DEFAULT_RTP_PORT 5002
#define VIDEO_WIDTH 720
#define VIDEO_HEIGHT 480
```

```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_0 -s a
utovideosrc -h 127.0.0.1 -p 5002
pipeline_manager constructor called
Running...
^C
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_1 -s v
ideotestsrc -h 127.0.0.1 -p 5002
pipeline_manager constructor called
source_type: videotestsrc
RTP host: 127.0.0.1
RTP port: 5002
overlay_manager constructor called
Running...
```

deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
tcp://127.0.0.1:5002 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
ADASI Test
00:05
00:00
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL

In order to see the results by yourself you have to open VLC and go to Media -> Open Network Stream and put the next address:

tcp://127.0.0.1:5002

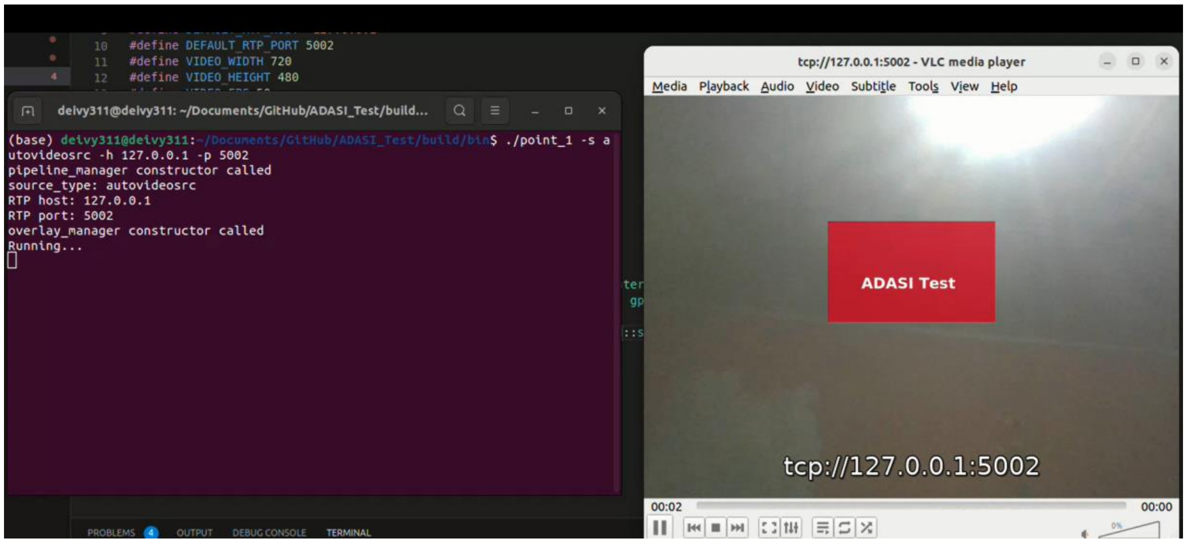
The video related to this is in the Folder Results, named point_1_overlay_file.webm which means it only streams video with overlay, in this case a red box.

Case 1.3: RTP with overlay, camera

Given the next command input:

```
./point_1 -s autovideosrc -h 127.0.0.1 -p 5002
```

We can see the next results:



```
#define DEFAULT_RTP_PORT 5002
#define VIDEO_WIDTH 720
#define VIDEO_HEIGHT 480
```

```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_1 -s a
utovideosrc -h 127.0.0.1 -p 5002
pipeline_manager constructor called
source_type: autovideosrc
RTP host: 127.0.0.1
RTP port: 5002
overlay_manager constructor called
Running...
```

deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
tcp://127.0.0.1:5002 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
ADASI Test
tcp://127.0.0.1:5002
00:02
00:00
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL

In order to see the results by yourself you have to open VLC and go to Media -> Open Network Stream and put the next address:

tcp://127.0.0.1:5002

The video related to this is in the Folder Results, named point_1_overlay_camera.webm which means it only streams a camera video with overlay, in this case a red box.

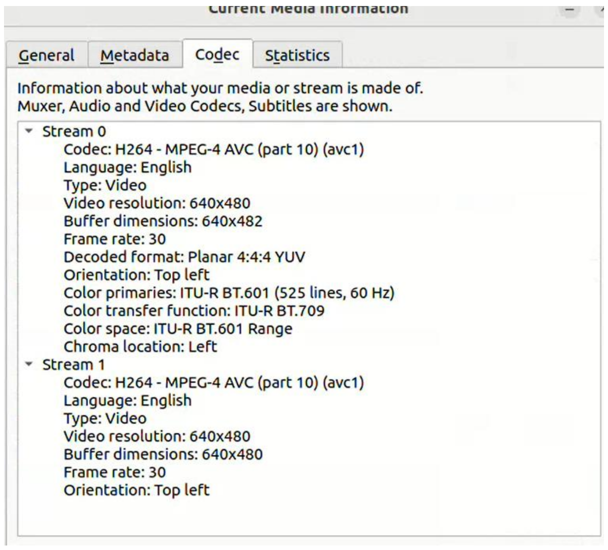
The executable **point_2_3**, can show the results for the literal 2 and 3, it is a RTP pipeline where can stream a video file or a camera and also can stream a video file or a camera with dynamic overlay and output the video in H264.

Case 2.1: RTP with dynamic overlay in H264 format video output, demo file

Given the next command input:

```
./point_2_3 -s videotestsrc -h 127.0.0.1 -p 5002
```

We can see the next results:



Information about what your media or stream is made of.
Muxer, Audio and Video Codecs, Subtitles are shown.

```
Stream 0
    Codec: H264 - MPEG-4 AVC (part 10) (avc1)
    Language: English
    Type: Video
    Video resolution: 640x480
    Buffer dimensions: 640x482
    Frame rate: 30
    Decoded format: Planar 4:4:4 YUV
    Orientation: Top left
    Color primaries: ITU-R BT.601 (525 lines, 60 Hz)
    Color transfer function: ITU-R BT.709
    Color space: ITU-R BT.601 Range
    Chroma location: Left
Stream 1
    Codec: H264 - MPEG-4 AVC (part 10) (avc1)
    Language: English
    Type: Video
    Video resolution: 640x480
    Buffer dimensions: 640x480
    Frame rate: 30
    Orientation: Top left
```

In order to see the results by yourself you have to open VLC and go to Media -> Open Network Stream and put the next address:

tcp://127.0.0.1:5002

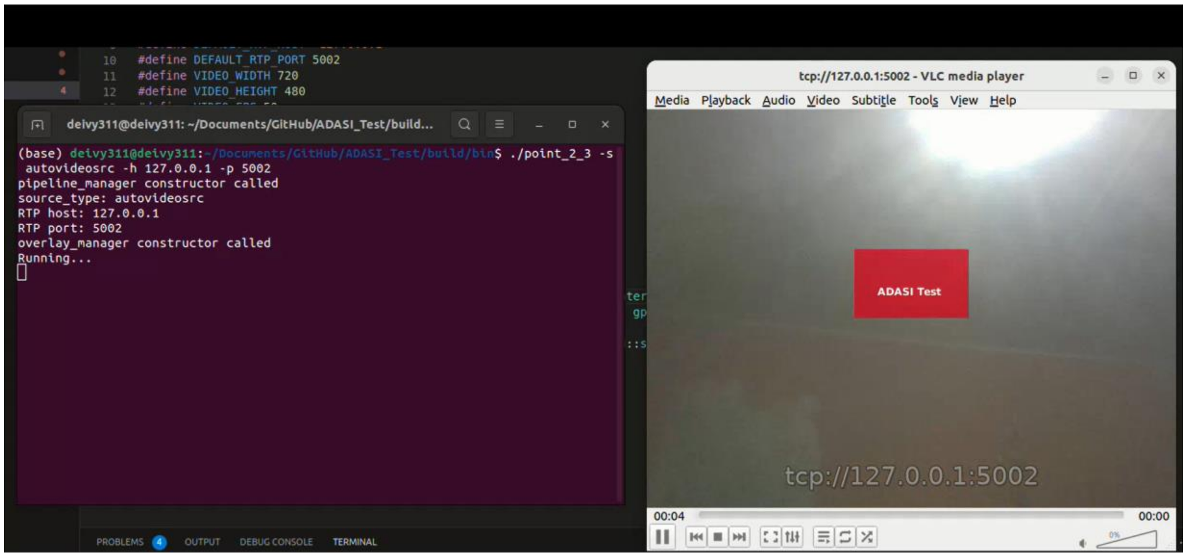
The video related to this is in the Folder Results, named point_2_3_dinamic_overlay_file.webm which means it only streams a video file with dynamic overlay and output the video in H264 format.

Case 2.2: RTP with dynamic overlay in H264 format video output, camera.

Given the next command input:

./point_2_3 -s autovideosrc -h 127.0.0.1 -p 5002

We can see the next results:



```
#define DEFAULT_RTP_PORT 5002
#define VIDEO_WIDTH 720
#define VIDEO_HEIGHT 480
```

```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_2_3 -s
 autovideosrc -h 127.0.0.1 -p 5002
pipeline_manager constructor called
source_type: autovideosrc
RTP host: 127.0.0.1
RTP port: 5002
overlay_manager constructor called
Running...
```

In order to see the results by yourself you have to open VLC and go to Media -> Open Network Stream and put the next address:

tcp://127.0.0.1:5002

The video related to this is in the Folder Results, named point_2_3_dinamic_overlay_camera.mp4 which means it only streams a camera video with dynamic overlay and output the camera video in H264 format.

The Executable **point_4**, can show the results for the literal 4, it is a RTSP pipeline over RTP where can stream a video file or a camera with dinamica overlay and output the video in H264.

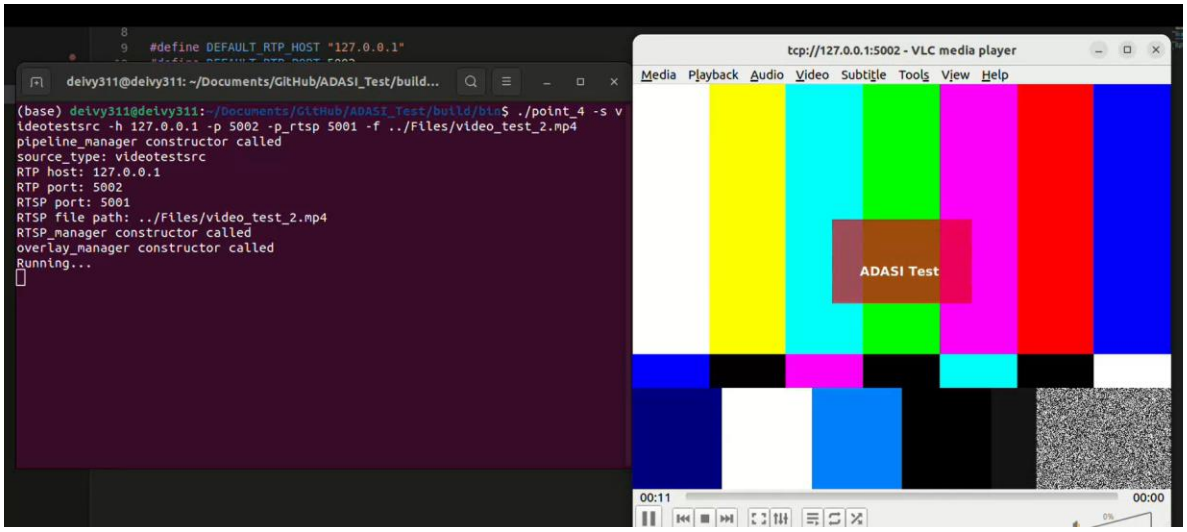
Case 4.1: RTSP with dinamic overlay in H264 format video output, demo file

Given the next command input:

```
./point_4 -s videotestsrc -h 127.0.0.1 -p 5002 -p_rtsp 5001 -f ../Files/video_test_2.mp4
```

We can see the next results:

RTP server



```
#define DEFAULT_RTP_HOST "127.0.0.1"
```

```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_4 -s v
ideotestsrc -h 127.0.0.1 -p 5002 -p_rtsp 5001 -f ../Files/video_test_2.mp4
pipeline_manager constructor called
source_type: videotestsrc
RTP host: 127.0.0.1
RTP port: 5002
RTSP port: 5001
RTSP file path: ../Files/video_test_2.mp4
RTSP_manager constructor called
overlay_manager constructor called
Running...
```

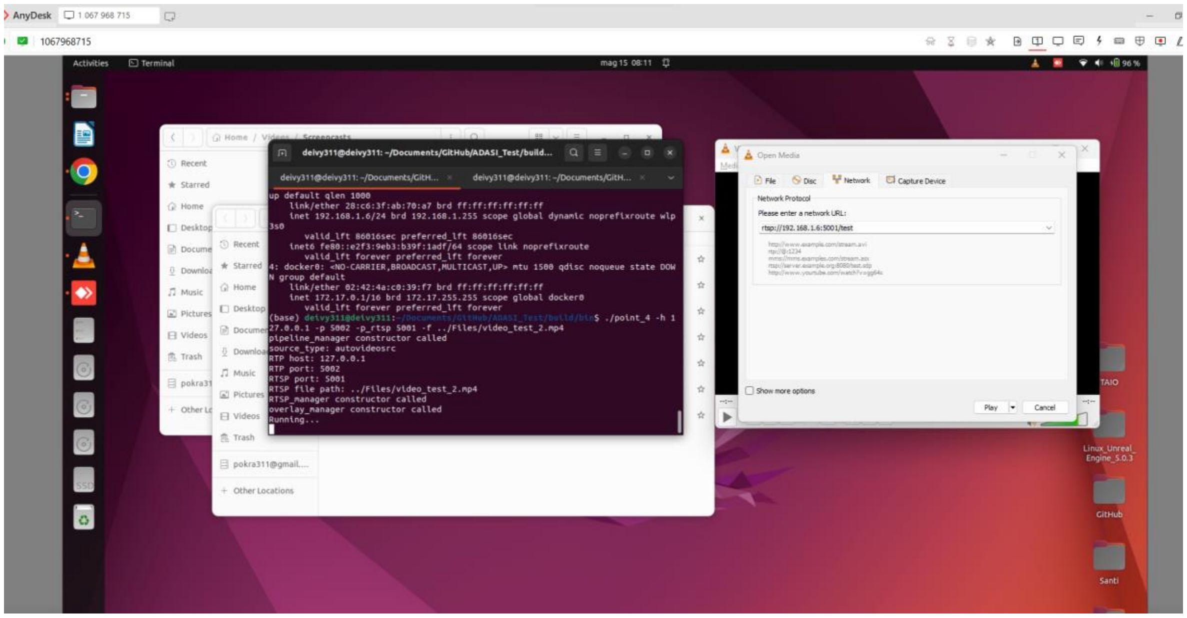
deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
tcp://127.0.0.1:5002 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
ADASI Test
00:11
00:00

The video related to this is in the Folder Results, named point_4_RTP_dinamic_overlay_file.webm which means it streams a video file with dynamic overlay and output the video in H264 format.

The VLC command to see the results is:

tcp://127.0.0.1:5002

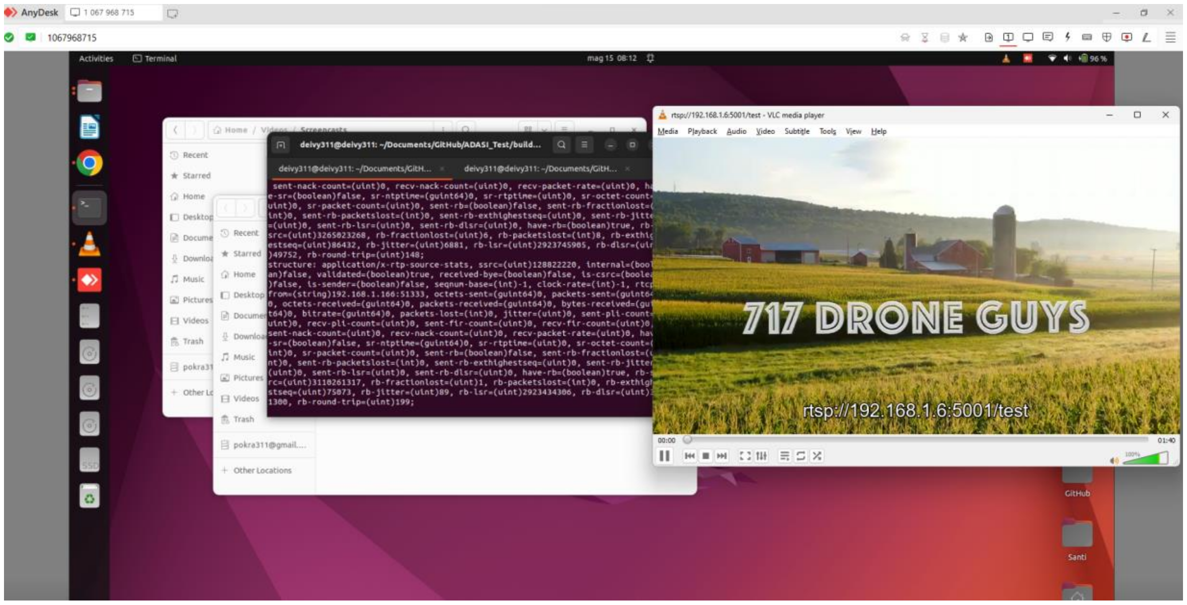
RTSP server



```
deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
up default qlen 1000
    link/ether 28:c6:3f:ab:70:a7 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.6/24 brd 192.168.1.255 scope global dynamic noprefixroute wlp
3s0
       valid_lft 86016sec preferred_lft 86016sec
    inet6 fe80::e2f3:9eb3:b39f:1adf/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOW
N group default
    link/ether 02:42:4a:c0:39:f7 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ ./point_4 -h 1
27.0.0.1 -p 5002 -p_rtsp 5001 -f ../Files/video_test_2.mp4
pipeline_manager constructor called
source_type: autovideosrc
RTP host: 127.0.0.1
RTP port: 5002
RTSP port: 5001
RTSP file path: ../Files/video_test_2.mp4
RTSP_manager constructor called
overlay_manager constructor called
Running...
```

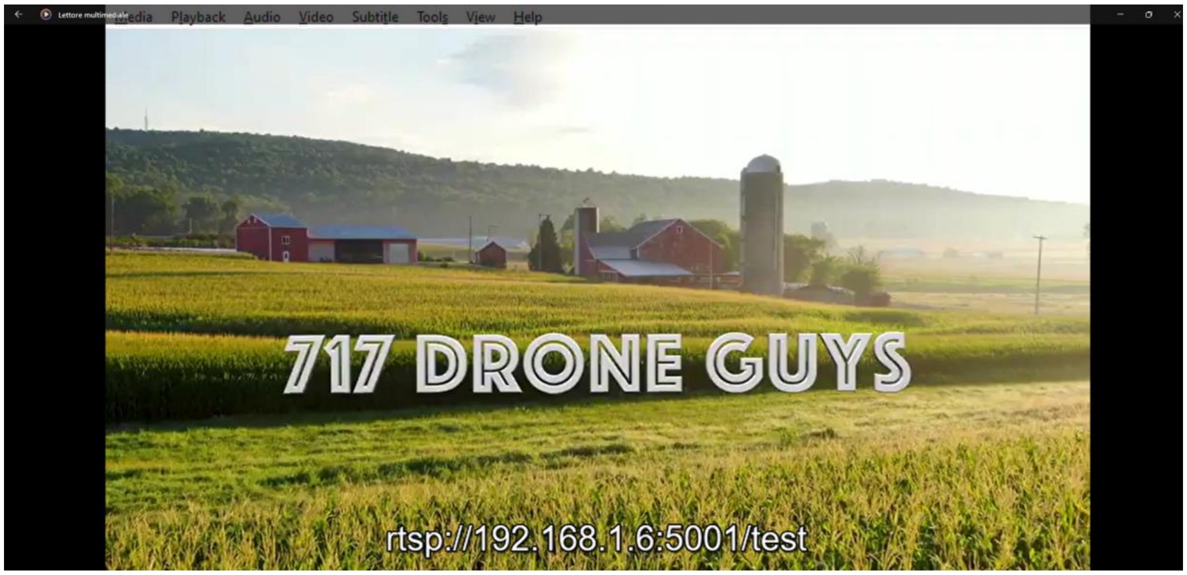
Open Media — Network

Please enter a network URL:
rtsp://192.168.1.6:5001/test



```
deivy311@deivy311: ~/Documents/GitHub/ADASI_Test/build...
 sent-nack-count=(uint)0, recv-nack-count=(uint)0, recv-packet-rate=(uint)0, ha
e-sr=(boolean)false, sr-ntptime=(guint64)0, sr-rtptime=(uint)0, sr-octet-count
uint)0, sr-packet-count=(uint)0, sent-rb=(boolean)false, sent-rb-fractionlost=(
int)0, sent-rb-packetslost=(int)0, sent-rb-exthighestseq=(uint)0, sent-rb-jitte
=(uint)0, sent-rb-lsr=(uint)0, sent-rb-dlsr=(uint)0, have-rb=(boolean)true, rb-
src=(uint)3265023268, rb-fractionlost=(uint)6, rb-packetslost=(int)8, rb-exthig
estseq=(uint)86432, rb-jitter=(uint)6881, rb-lsr=(uint)2923745985, rb-dlsr=(uir
)49752, rb-round-trip=(uint)148;
structure: application/x-rtp-source-stats, ssrc=(uint)128822220, internal=(boo
an)false, validated=(boolean)true, received-bye=(boolean)false, is-csrc=(boole
)false, is-sender=(boolean)false, seqnum-base=(int)-1, clock-rate=(int)-1, rtc
from=(string)192.168.1.166:51333, octets-sent=(guint64)0, packets-sent=(guint6
0, octets-received=(guint64)0, packets-received=(guint64)0, bytes-received=(gu
t64)0, bitrate=(guint64)0, packets-lost=(int)0, jitter=(uint)0, sent-pli-count
uint)0, recv-pli-count=(uint)0, sent-fir-count=(uint)0, recv-fir-count=(uint)0,
sent-nack-count=(uint)0, recv-nack-count=(uint)0, recv-packet-rate=(uint)0, hav
-sr=(boolean)false, sr-ntptime=(guint64)0, sr-rtptime=(uint)0, sr-octet-count=
int)0, sr-packet-count=(uint)0, sent-rb=(boolean)false, sent-rb-fractionlost=(
nt)0, sent-rb-packetslost=(int)0, sent-rb-exthighestseq=(uint)0, sent-rb-jitter
(uint)0, sent-rb-lsr=(uint)0, sent-rb-dlsr=(uint)0, have-rb=(boolean)true, rb-
rc=(uint)3110261317, rb-fractionlost=(uint)1, rb-packetslost=(int)0, rb-exthigh
stseq=(uint)75073, rb-jitter=(uint)89, rb-lsr=(uint)2923434306, rb-dlsr=(uint)
1300, rb-round-trip=(uint)199;
```

rtsp://192.168.1.6:5001/test - VLC media player
717 DRONE GUYS
rtsp://192.168.1.6:5001/test
00:00
01:40


Lettore multimediale
Media Playback Audio Video Subtitle Tools View Help
717 DRONE GUYS
rtsp://192.168.1.6:5001/test

The video related to this is in the Folder Results, named point_4_RTSP_dinamic_overlay_file.mp4 which means it streams a video file with dinamic overlay and output the video in H264 format.

The VLC command to see the results is:

rtsp://192.168.1.6:5001/test # this is the ip of the same computer but viewed from

The Executable **point_5**, can show the results for the literal 5, it is a RTSP pipeline over RTP where can stream a video file or a camera with dynamic overlay and output the video in H264. An as additional feature the video streamed is stored.

Case 5.1: Storing file.

Given the next command input:

```
./point_5 -s autovideosrc
```

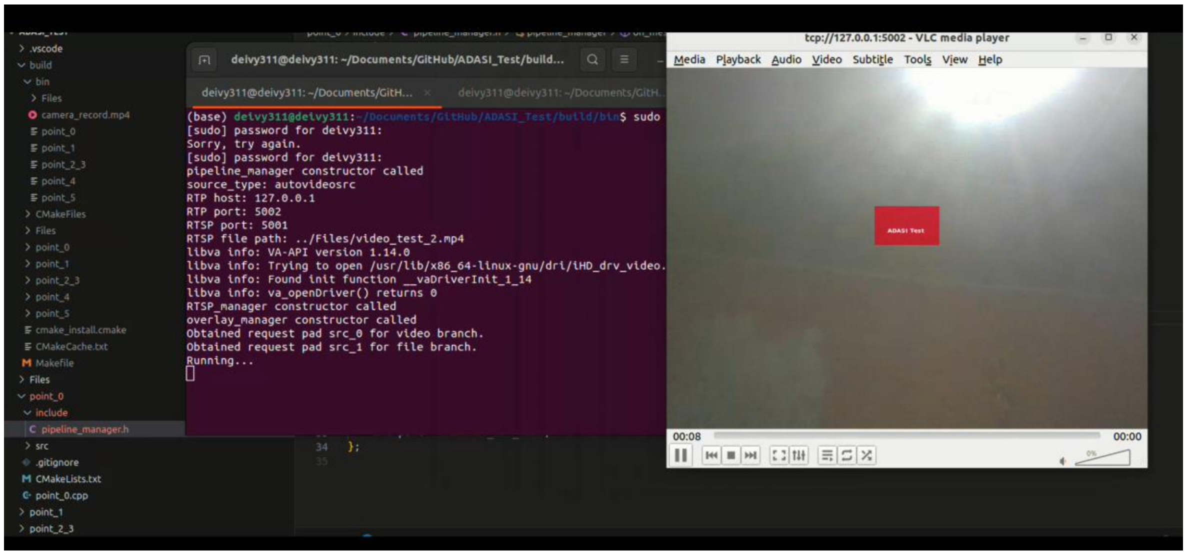
The video related to this is in the Folder Results, named point_4_RTP_dinamic_overlay_camera.mp4 which means it streams a camera video with dinamic overlay and output the video in H264 format.

The VLC command to see the results is:

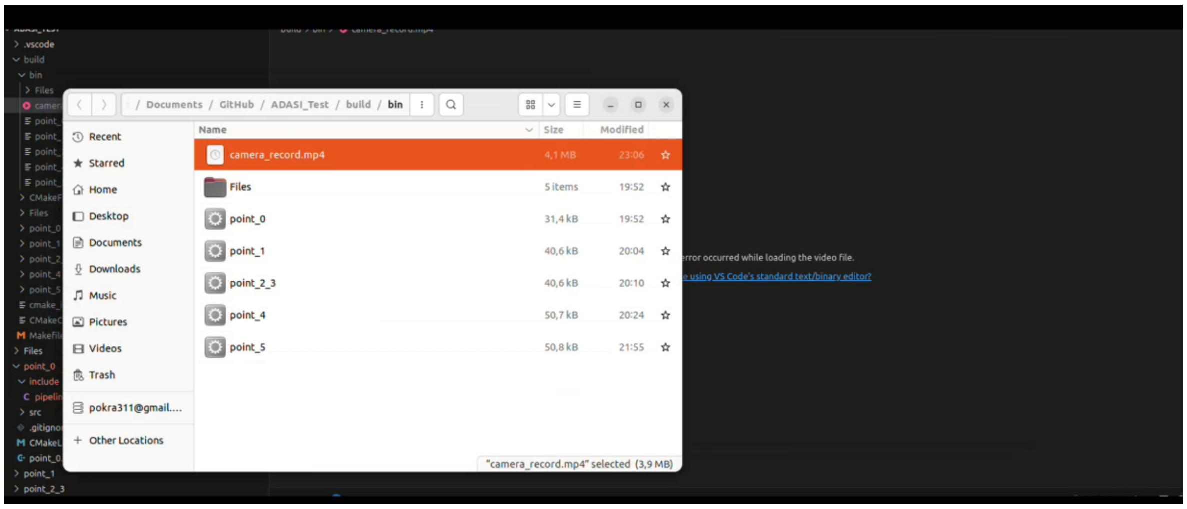
tcp://127.0.0.1:5002

rtsp://127.0.0.1:5001/test another

The video related to this is in the Folder Results, named point_5_RTSP_RTP_dinamic_overlay_camera.mp4 which means it streams a camera video with dynamic overlay and output the video in H264 format.



```
(base) deivy311@deivy311:~/Documents/GitHub/ADASI_Test/build/bin$ sudo
[sudo] password for deivy311:
Sorry, try again.
[sudo] password for deivy311:
pipeline_manager constructor called
source_type: autovideosrc
RTP host: 127.0.0.1
RTP port: 5002
RTSP port: 5001
RTSP file path: ../Files/video_test_2.mp4
libva info: VA-API version 1.14.0
libva info: Trying to open /usr/lib/x86_64-linux-gnu/dri/iHD_drv_video.
libva info: Found init function __vaDriverInit_1_14
libva info: va_openDriver() returns 0
RTSP_manager constructor called
overlay_manager constructor called
Obtained request pad src_0 for video branch.
Obtained request pad src_1 for file branch.
Running...
```

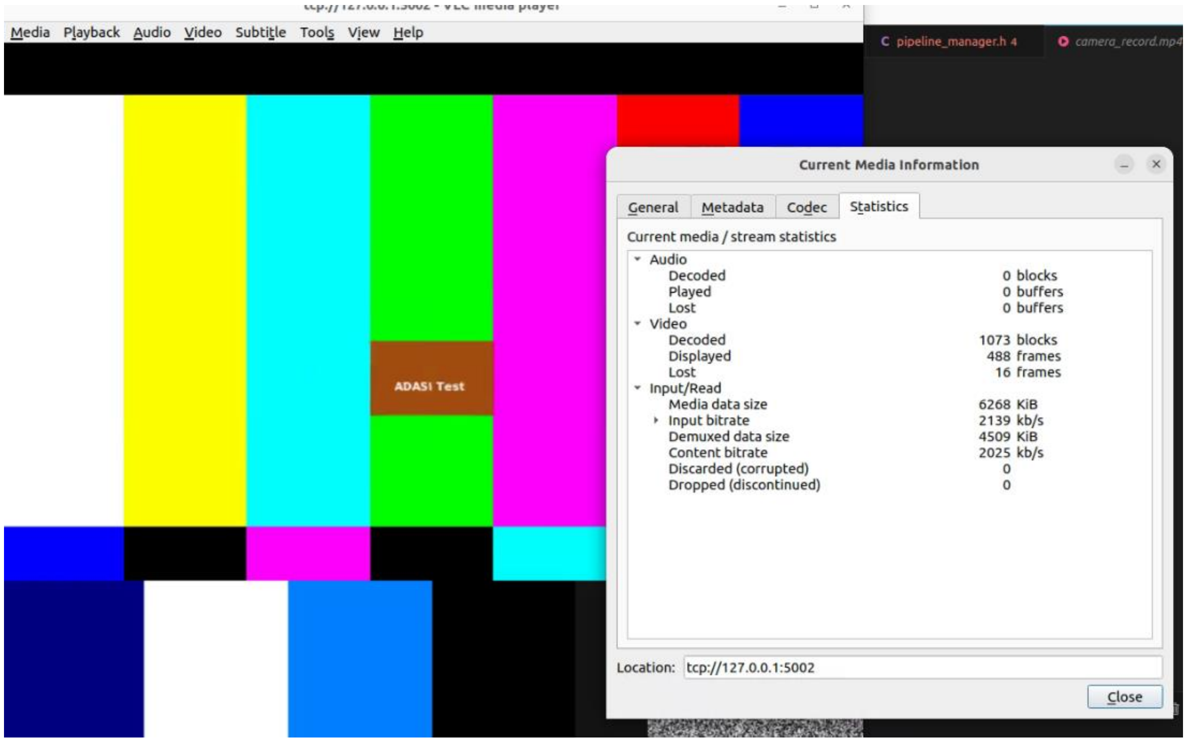


Name	Size	Modified
camera_record.mp4	4,1 MB	23:06
Files	5 items	19:52
point_0	31,4 kB	19:52
point_1	40,6 kB	20:04
point_2_3	40,6 kB	20:10
point_4	50,7 kB	20:24
point_5	50,8 kB	21:55

"camera_record.mp4" selected (3,9 MB)

Metrics:

In the next image we can se the metrics computed for the point_4 and point_5


tcp://127.0.0.1:5002 - VLC media player
Media Playback Audio Video Subtitle Tools View Help
ADASI Test
pipeline_manager.h 4
camera_record.mp4
Current Media Information
General Metadata Codec Statistics
Close

Current media / stream statistics

```
Audio
    Decoded                              0 blocks
    Played                               0 buffers
    Lost                                 0 buffers
Video
    Decoded                           1073 blocks
    Displayed                          488 frames
    Lost                                16 frames
Input/Read
    Media data size                   6268 KiB
    Input bitrate                     2139 kb/s
    Demuxed data size                 4509 KiB
    Content bitrate                   2025 kb/s
    Discarded (corrupted)                0
    Dropped (discontinued)               0
```

Location: tcp://127.0.0.1:5002