SLP (StereoPi Livestream Playground) Raspbian Image

Contents

SLP Raspbian Image Specific

Partition structure

How to prepare MicroSD card with SLP

Network configuration
- Ethernet access
- WiFi access
- Trick
- 4G/LTE Dongle

How do you find StereoPi in your network?
- Connecting to SLP with auto-discovery feature
- Connecting to SLP with other methods

Android application
- Android Application Compatibility
- Android Application Performance Notice
- Download link

HDMI stereoscopic preview (RC FPV usage)

SLP Admin panel options
- Recording
- Max rec duration
- Stream UDP
- UDP clients
- Broadcast
- Audio enabled
- RTMP URL (YouTube etc.)
  - Trick:
  - Tip:
- Video mode
- Decimation
- Upside/Down
- Video resolution
- FPS
- Bitrate
- White balance
- Exposure
- Contrast
- Sharpness
- Digital gain
- Wifi SSID
- WiFi password

Updating software

Advanced users' section

SSH access

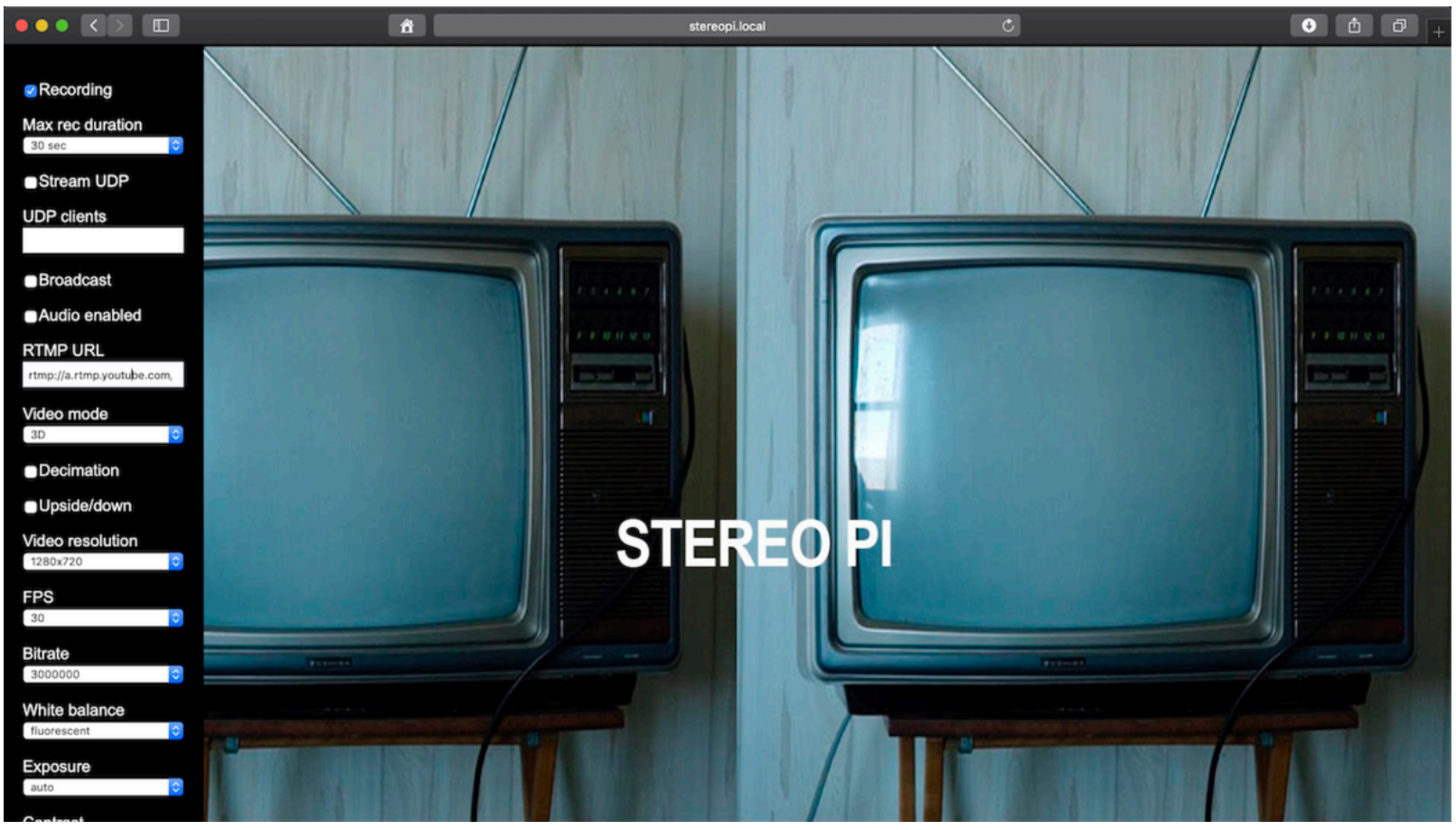
SLP structure details you should know

Receive livestream on Windows

Receive livestream on Mac

Receive livestream on iOS devices

SLP Raspbian Image Specific



This SLP (StereoPi Livestream Playground) image was used for all livestream experiments mentioned on our crowdfunding page

https://www.crowdsupply.com/virt2real/stereopi

and has some additional features. It is not comfortable to use with a monitor, keyboard and mouse. We do it in consumer-like way to save time. It has an Administration panel (like WiFi routers), and is also accessible via SSH.

As this image is compatible with both **StereoPi** and **classic Raspberry Pi**, let us use "**Pi**" name in this article.

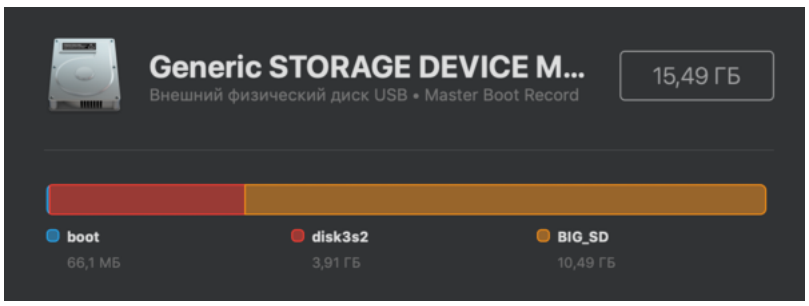
And last notice: by default single-camera mode is enabled for classic Raspberry Pi.

Please remember it is Beta only. If you find some bugs, do not be angry. Please let us know here on Raspberry Pi forum:

```
https://www.raspberrypi.org/forums/viewtopic.php?f=37&t=216940
```

Partition structure

Classic Raspbian has two partitions: BOOT and a Linux filesystem. Our image has both of them (using 4Gb on MicroSD), and also offers 3rd partition in Fat32. It is used for recording captured video. So after recording a video, you can insert MicroSD into your computer and see all your records, as with your car dashcam.



How to prepare MicroSD card with SLP

1. Prepare MicroSD card 16 Gb or higher capacity.
2. Download image file and unzip it on your computer. **File size is about 5 Gb.**

   Download mirror 1 (http://wiki.stereopi.com/files/slp-raspbian-190205.zip) (Our Wiki hosting)

   Download mirror 2 (https://drive.google.com/file/d/1JnxccWD9Vjz0Yjj16mewli41-grd7M2Z/view?usp=sharing) (Google Drive)

   Download mirror 3 (https://yadi.sk/d/utvXWrhQ-Q5IrA) (Yandex Disk)

1. Write image to MicroSD. As a Pi user, you might know, that it does not mean “copy-paste” image file, but requires special image writing software. In Windows we prefer to use Win32DiskImager (https://sourceforge.net/projects/win32diskimager/).
2. Do not remove MicroSD right after this procedure, as you might want to set up access to your WiFi hot spot. You can do it right after preparing image.

Tip1: Not all MicroSD marked as 16Gb have the exact same size. Win32DiskImager may warn you that your MicroSD is too small, but you have the option to “continue”. For this case we leave several hundreds of unused spaces at the end of MicroSD, so image will not be broken on such cards.

Tip 2 for advanced users: Main system use only 4 Gb on MicroSD, and other ~12 Gb is used for video records. If you have only 8 Gb card, you may record our SLP image too with Win32DiskImager. But after that you need to manually fix broken Fat32 partition (create it again and format).

Network configuration

Ethernet access

If you connect Pi by Ethernet to your router, you do not need to configure it. IP address will be obtained automatically by DHCP.

WiFi access

If you did not removed prepared MicroSD from your computer, you should now go to “BOOT” disk and find “**stereopi.conf**” file. Open it and write your WiFi access point name and password in these rows:

```
wifi_ssid=my-access-point-name
wifi_psk=mypassword
```

If you use Raspberry Pi WITHOUT built-in WiFi or StereoPi, please do not forget to connect WiFi dongle to your device before powering on. Now your Pi will automatically connect to your router after booting. If it is does not work – please check twice AP name and password. Also system might not like SSID name with spaces in it.

Trick

If you enable WiFi access point on your phone, Pi will access Internet over it. That is, you can both livestream to YouTube and access Admin panel. So you can livestream to YouTube using your phone and powerbank to power on your Pi.

4G/LTE Dongle

If you connect 4G/LTE dongle with RNDIS support, your Pi will livestream to the Internet. In our aquarium experiment we used Huawei E3372 dongle.

How do you find StereoPi in your network?

We use auto-discovery feature in our SLP image. So here are some approaches how to connect StereoPi using it.

Connecting to SLP with auto-discovery feature

We suggest your Pi and computer/phone/tablet are connected to the same router, and your network has no difficult topology.

1. Open a browser and type:

http://stereopi.local

Admin panel will appear. In some networks it works immediately, but in other networks it might take some time to find and open panel (10-20 seconds). You will see Administration panel:

2. Open our Android application. It will automatically find your Pi, connect to it and start playing livestream video. If you tap on the screen you will see a prompt with an IP address. If you tap on it, then browser will be opened with

Admin panel.

Connecting to SLP with other methods

1. You can use software for network scan to find IP address.

- For Windows you can use Advanced IP Scanner (http://www.advanced-ip-scanner.com)

- For Mac you can use LanScan (https://itunes.apple.com/us/app/lanscan/id472226235?mt=12).

2. You can go to your router Admin panel and look for IP addresses in a list of connected devices.

3. After you've found IP address, type it in your browser.

Android application

This simple application automatically connects to Pi with SLP image, requests video livestream and shows it.

If you use StereoPi and "3D" mode enabled in Admin panel – you can enjoy 3D video by putting your phone to google cardboard-like device. We like ColorCross (https://www.geeektech.com/en/geeek-color-cross-vr-3d-glasses-for-smartphones.html) and Pocket 360 (https://www.gearbrain.com/i-am-cardboard-vr-headset-2349974389.html) solutions.

If you use classic Pi, you can see video livestream from your camera without any additional equipment.

Tap on the screen to show prompt with current IP address of Pi.

Clicking on this prompt will open browser with Admin panel.

You can also use this application on Oculus Go as we did. Put your Oculus Go in developer mode and upload application. At this moment there are no joystick support for Oculus, and we plan to add it later.

Android Application Compatibility

Application was tested on Android 7.1.1, 7.1.2 and 8.0

It is compiled for Androids >=5.0, but was not tested with 5 and 6 versions.

Android Application Performance Notice

With old Android phones (5 years or more) you may obtain high latency of your video (delay up to 0.3-0.5 seconds). With newer models video latency is near 0.1 seconds.

Download link

Download APK file. (http://wiki.stereopi.com/files/StereoPi-2019-02-05.apk)

HDMI stereoscopic preview (RC FPV usage)

Right after start Pi will push image from camera to HDMI output. For 3D view on HDMI we used some trick. By default Raspbian turn HDMI in 3D mode if it needs to show 3D image, for devices like 3D TV. So on a classic monitor you will see an image from one camera and a part of an image from second camera. We have modified `raspivid` and manually turned HDMI in non-3d mode. So if you're RC modelist and want to livestream 3D from StereoPi using your FPV link with HDMI capture – just connect StereoPi to it. In the future we will discuss this option with Raspbian support team to find an easier way to trigger this 3D/non-3D HDMI mode.

SLP Admin panel options

Recording

If you turn on this feature, Pi will record all video to MicroSD card.

Max rec duration

Recorded video will be cut on pieces with a chosen length.

Why do you need it? The answer is simple. We capture “raw” h264 stream from a camera. If you record it directly, you will feel some problems while viewing it in your video player. For example, scroll may not work etc. So we need to convert it to MP4 for comfort use. Our SLP does it on-the-go. But to do it correctly our utility needs to add some information to the file. If you turn off your Pi during this process, you may lose your last record, but all previous records remain safe.

What time to choose? For short on-table experiments it is better to set minimal time (30 seconds). For real-life experiments it is better to set 5-10 minutes.

Stream UDP

Section for advanced users. You may put IP addresses and ports of devices you want to livestream to.

You may use several comma-separated values in “UDP clients” field, like this:

```
192.168.1.10:3000,192.168.1.20:4000
```

To capture this stream you may use your own application or gstreamer (see Advanced User’s Section).

UDP clients

Field for UDP clients (for Stream UDP option).

Broadcast

Turn on or turn off video livestream.

Audio enabled

If you connect USB sound card with Microphone, or USB microphone, system will try to stream video with sound.

RTMP URL (YouTube etc.)

If you put RTMP string here, Pi will send the livestream to it. For livestream to YouTube you need to copy livestream link from your YouTube studio and put it in this field. Our link looks like this:

```
rtmp://a.rtmp.youtube.com/live2/9k1d-2ue3-xxxx-xxxx
```

Trick:

YouTube does not allow livestream without sound. So if SLP has not detected a USB soundcard and microphone, it adds fake silence sound to this stream, and it works. If you have USB soundcard and mic, you will hear sound on YouTube.

Tip:

If you have no experience on YouTube livestream, you should know that YouTube adds a time buffer of up to 30 seconds for your video. That is, you will not see your video immediately, and when you see it, you can find some delay. It is not a bug, but a feature. It helps viewers to obtain smooth and high-quality video.

Video mode

Choose 3D if you have StereoPi or another Pi-based solution with two cameras connected. Choose 2D if you have classic Raspberry Pi. If you use this option with StereoPi, only one camera will be used for video capture.

Decimation

This option compress each taken image horizontally to 50% of its original width. This feature is useful for FPV and Depth Map, as you save full FOV (field of view) of your camera. For other implementations you may uncheck this option. In this case FOV will be cropped, but you have normal side ratio of your video. For our Oculus experiment we unchecked this option. In fact this option adds “-dec” parameter to raspivid.

Upside/Down

Rotate image to 180 degree. With 1 camera it is simple. With 2 cameras we do flip and also swap left/right images. Otherwise your left and right images appear in the wrong place, and your brain will be stuck on trying to build 3D image from this.

Video resolution

Resolution of video captured.

FPS

Frames Per Second settings for video. Please notice, that if you choose too high FPS for high resolution you may obtain capture problems. For 1280x720 with ONE camera you may use 48 FPS. For 1280x720 and 3D use 42 FPS. You can experiment with these settings to fit your needs.

We want to notice that highest FPS is critical for at least two scopes. First one is livestream to VR helmets like Oculus, as ease of perception depends on this. The second one is RC FPV, as higher FPS decrease video latency.

Bitrate

Number of megabits used for 1 second of video after compression, in megabits (not megabytes). This settings affects the H264 encoder of your Pi. The higher bitrate, the better video quality you have, and the more bandwidth connection required for livestream. For livestream we recommend to use 2-3 Mbit at HD or FullHD resolution. For 640x480 even 1 Mbit is Ok. Higher bitrates are useful for recording video on MicroSD without livestream. Please notice you need a high-speed MicroSD to record at high bitrates.

White balance

Changes white balance mode.

In this option we implemented options available for Raspberry Pi Camera (https://www.raspberrypi.org/documentation/raspbian/applications/camera.md).

Exposure

Changes exposure mode.

In this option we implemented options available for Raspberry Pi Camera (https://www.raspberrypi.org/documentation/raspbian/applications/camera.md).

Contrast

Changes contrast mode.

In this option we implemented options available for Raspberry Pi Camera (https://www.raspberrypi.org/documentation/raspbian/applications/camera.md).

Sharpness

Changes sharpness mode.

Notice: It looks as if this feature has a bug in Raspbian, as it is work with only one camera in stereoscopic mode.

In this option we implemented options available for Raspberry Pi Camera (https://www.raspberrypi.org/documentation/raspbian/applications/camera.md).

Digital gain

Changes digital gain mode. In this option we implemented options available for Raspberry Pi Camera (https://www.raspberrypi.org/documentation/raspbian/applications/camera.md)

Wifi SSID

Name of your WiFi access point

Can be adjusted here or in `stereopi.conf` file in a BOOT partition

WiFi password

Password for your WiFi.

Updating software

If we have an update, we created a file named `stereopi.tar.gz`.

You can put it in “BOOT” part of MicroSD. After start, SLP will automatically find and implement it.

Notice: We have not released any public updates yet.

Advanced users’ section

SSH access

You might want to look inside. When you know the IP address of your Pi, you can access it by SSH.

User credentials are the same as for default user:

login: `pi`

password: `raspberry`

SLP structure details you should know

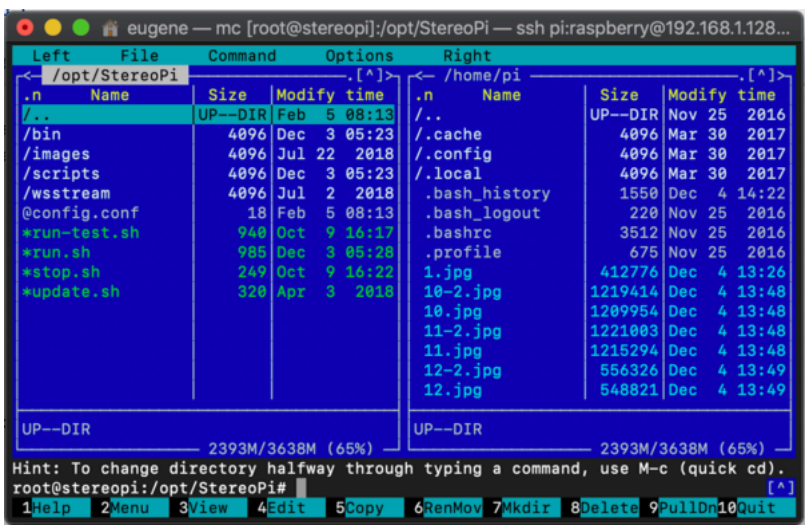
1. We prepared this image for aggressive usage. That's why we want to save the filesystem in a case of occasional poweroffs. So by default, filesystem is read-only and you cannot change any file. To unlock it type:

```
sudo mount -o rw,remount /
```

2. `mc` already installed, you may use it.

3. Our streaming scripts are here:

```
/opt/StereoPi
```



If you will enter this directory, you can stop all scripts by running:

```
./stop.sh
```

To start them again run:

```
./run.sh
```

4. Most scripts have auto-restart feature (start again if something go wrong).

5. A few words about "chunker" utility:

We have a lot of projects when we capture video from one source and need to use it in several ways at the same time (like livestream, record and use for Computer Vision). So we need to send it to several applications at the same time. We use a network approach with this, and subsystems of our projects talk with each other over network infrastructure with 127.0.0.1 and different ports. This approach gives us flexibility and we can easily forward video stream to internal subsystem or to external IP address. For example, if you decide to move your OpenCV part from Pi to nearby NVidia Jetson, you just aim livestream from internal IP and port to Jetson's one. So "in the root" of this system we have special "chunker" utility. It cuts video stream on pieces to fit network bandwidth with maximum efficiency, and try to adapt to maximum MTU available. At this moment this utility is overloaded with a lot of features for our custom projects. We will exclude all unneeded features and open source it after polishing.

Receive livestream on Windows

You need to have gstreamer 1.x installed.

If you already have it, run this script:

```
gst-launch-1.0 -v udpsrc port=3000 buffer-size=300000 ! h264parse ! avdec_h264 ! fpsdisplaysink sync=false
```

Receive livestream on Mac

To receive it on Mac you may run the same line:

```
gst-launch-1.0 -v udpsrc port=3000 buffer-size=300000 ! h264parse ! avdec_h264 ! fpsdisplaysink sync=false
```

For example, my notebook have IP address 192.168.1.139

To obtain livestream I need:

1. In SLP Admin panel I turn on "Stream UDP" option
2. Put "192.168.1.139:3000" in UDP clients field
3. Press "save" at the bottom of Admin panel
4. Run gst-launch line in command line on my Mac

My current GStreamer info:

```
MBP-Evgeny:~ eugene$ gst-launch-1.0 --version
gst-launch-1.0 version 1.14.4
GStreamer 1.14.4
Unknown package origin
```

Gstreamer have been installed with Brew package manager.

Receive livestream on iOS devices

For iOS we used gstreamer-based Fishing FanCam application, described in EZ-broadcast Wiki (https://github.com/rodizio1/EZ-WifiBroadcast/wiki/GCS-~-FishingFanCam).

But looks like this application is not supported now. We plan to do our own application for this, but it might take a lot of time. We will be pleased If you can suggest other gstreamer-based iOS applications or help us with making simple application like our Android one.

Retrieved from "http://wiki.stereopi.com/index.php?title=SLP_(StereoPi_Livestream_Playground)_Raspbian_Image&oldid=160"

This page was last edited on 7 February 2019, at 12:57.