

**NAME**

tinycamd – an efficient HTTP interface to a webcam

**SYNOPSIS**

**tinycamd** [-hv] [-d /dev/video] [-s 640x480] [-p 0.0.0.0:8080]

**DESCRIPTION**

Tinycamd is built to run efficiently, especially on very small hardware. Ideally you will use a camera that can provide an MJPEG or JPEG encoded image stream. tinycamd will reformat it into JPEG images without serious computation. Alternatively you can use a YUYV webcam and let tinycamd do the remainder of the encoding.

Typically, you will use a webserver to proxy to tinycamd for your images. See the --url-prefix option for removing the remaining URL prefix in this case.

**URLS**

tinycamd responds to the following URLs (and some others for internal reasons):

/image.jpg

Return the next frame as a JPEG image. Any URL query string will be ignored, so you can use that to defeat overzealous proxies.

/setup.html

Display a page with the camera controls exposed to HTML-5 compatible browsers. Handy for exploring control functions.

/controls

Return an XML document defining all of the controls, their ranges, and current values.

/set?CID=VALUE

Set a control. CID and VALUE are both integers and will have been concocted by you with reference to the /controls URL.

**OPTIONS**

-h, --help

Display a short help text.

-d, --device PATH

Specify the video device. The default is /dev/video0.

-p, --port [ADDR:]PORT

Specify the TCP port to listen on. The default is 0.0.0.0:8080, this is port 8080 on all available interfaces. You could use 127.0.0.1:8080 if you only wanted to listen on the loopback interface to restrict access to the local host.

-d, --daemon

Detach and run as a daemon.

-s, --size WIDTHxHEIGHT

Specify the dimensions of the video capture. The default is 640x480. Remember that as the capture size goes up, the frame rates possible will go down for most cameras.

-q, --quality QUALITY

Specify the JPEG encoding quality from 1 to 100. Not all cameras which encode to JPEG will respect this, and those that do will probably have discrete levels which they support. Likewise, MJPEG cameras generally ignore this as well. You can use yuyv format and encode in the CPU if you need control, but this will consume many times more CPU time.

-f, --fps NUM

The number of frames per second to capture. This will be used as a suggestion to the camera, but it could be faster or slower depending on what the device supports.

- U, --url-prefix PATH  
If specified, this path will be removed from the front of each URL. This is useful when you are behind a proxy that passes the original path to tinycamd.
- P, --probe  
Probe the camera and print out its capabilities. Most useful for learning a device's supported image formats, resolutions, and frame rates.
- F, --format FMT  
Set the camera's capture format. Supported values are 'jpeg', 'mjpeg', and 'yuyv'.
- M, --monochrome  
Tell the camera to capture monochrome images.
- I, --pid PATH  
A path in which to write our PID. Used for daemon scripts to make stopping easier. This must be writable by the process as invoked, it need not be writable after the setuid and chroot.
- i, --uid NAME  
After starting up and gaining access to the device and network, setuid to this uid.
- C, --chroot PATH  
After starting up, opening the camera, and recording our PID, chroot to this path. All libraries will be loaded, it can be an empty directory.
- password ACCOUNT:PASSWORD  
If specified, then HTTP Basic Authorization will be required to access the image data. The account and password are specified and separated by a colon. They will be hidden in 'ps' listings, but there is a window of time immediately after launch when they will be visible.
- setup-password ACCOUNT:PASSWORD  
If specified, then HTTP Basic Authorization will be required to control the camera. This account will also grant access to the image data.
- m, --mmap  
Use the mmap method to read video frames. Not generally interesting.
- r, --read  
Use the read method to read video frames. Not generally interesting.
- u, --userp  
Use the userp method to read video frames. Not generally interesting.
- v, --verbose  
Display a stream of debugging information.

## EXAMPLES

**tinycamd** --daemon --uid nobody --chroot /var/lib/tinycamd --device /dev/video0 --password watchman:bigeyes

## AUTHOR

Jim Studt, jim@studt.net