## NAME

centralscrutinizer - motion detection system for cameras

## SYNOPSIS

**centralscrutinizer** [-d <device>] [-s widxhgt]
**centralscrutinizer** --probe

## DESCRIPTION

Watch a camera for motion and record the frames in which motion occurs. This is a very CPU efficient system built for JPEG cameras which uses partial decoding of the images to feed the motion detector. On a 3GHz x86 platform a 640x480 10fps camera will take about 2% of the CPU. On a tiny embedded platform you can easily support several cameras.

## OPTIONS

**-d**, **--device** Video device name. /dev/video0 is used if not specified.

**-s** WID**x**HGT, **--size** WID**x**HGT
Size of image. It may be different, but it will try to get as close to this as possible.

**-S** directory, **--spool** directory
Where to write the interesting images. The default is the current working directory which is useful if you start with chroot.

**-f** fps, **--fps** fps
The number of frames per second to record. This is only supported for V4L2 devices at the moment. It will get as close as it can.

**-q** num, **--quality** num
JPEG quailty number, 0 to 100. For V4L2 drivers which support it, this lets you specify a JPEG quality.

**-h**, **--help** Print out a summaryof available options.

**-m**, **--mmap** Use memory mapped IO with the V4L2 driver. The old V4L driver only uses memory mapped IO.

**-r**, **--read** Use read() calls with the V4L2 driver. Note: this might run at 100% cpu use if your particular driver will return the same frame more than once.

**-u**, **--userp** Use application allocated buffers with V4L2. This is untested as I have not found a driver that supports it, but it is here for completeness in case you need it.

**-b** device, **--framebuffer** device
Device to use as a framebuffer for displaying the images. e.g. /dev/fb0. This is useful during setup and and when debugging. Only 32bit deep frame buffers will work. This is also a compile time option and may not be present in your build.

**-v**, **--verbose** Print a blinding stream of debugging inforamtion.

**-p**, **--probe** Do not run normally, just probe the camera and print some useful configuration information for V4L2 devices such as available resolutions and frame rates.

**-P** [node:]port, **--port** [node:]port
Specify a port for the HTTP listener. You can also specify an IP address if you don’t want to bind to the default address. You can specify node and port by name or number.

**-e**, **--autoexposure**
Dynamically adjust the brightness of the camera as the software runs. This is most useful for cameras and drivers that have broken or nonexistent autobrightness.

**-k** num, **--skip** num
Number of frames to skip between each motion detection frame. Useful if you are on a tiny processor and can’t afford to check each frame.

## IMAGE FILES

When motion is detected a stream of image files will be written with names like 20071123-223956-0001.jpg where the last 4 digits will increment with each image and the first 14 digits are the YYYYMMDD-HHMMSS timestamp of the event start. When the motion is completed a file named like 20071123-223956-done will be written.

Be aware, to avoid having to cache images it will write a few extra images at the end, then when it decides the motion ended it will delete these back to the last interesting frame. Wait for the "-done" file before you do anything with the images.

## THE HTTP INTERFACE

The HTTP interface is used to monitor and control the program. It responds to a couple of URLs.

/exit Exit the program.

/status Report some JSON encoded status.

/exposure?brightness=xxx&contrast=xxx&colour=xxx
Adjust the brightness, contrast, or colour saturation. You can omit any of these to not change them. The range is 0 to 65535 on each. /anything-else Any other URL will return the next frame received as a JPEG image.

## AUTHOR

Written by Jim Studt, jim@federated.com.