SLOW DOWN TO SPEED UP

A smart, cheap setup for shooting long-duration time-lapse movies. By Ken Murphy

We've all seen time-lapse movies that seem to speed up the world around us. The effect is very compelling, making processes that normally occur at a rate too slow to perceive unfold before our eyes, such as the blooming of a flower.

I'm working on a time-lapse movie that captures the dynamics of weather and clouds and the patterns of sunrise and sunset for an entire year. I need a rsetup to capture a large number of images. I need to collect images without interrupting the image-capture process, and to access the system remotely. I want to do it on the cheap, without sacrificing quality.

Here's the solution I came up with: I'm putting a dusty old PC back into service, installing Linux and gPhoto image capture software, and connecting it to my old 4-megapixel Canon A520 camera via USB. The camera will be mounted in an improvised (yet sturdy) outdoor enclosure. With this setup, the images can be continually captured directly to disk, around the clock, and I'm able to log in remotely to control the camera. I can also compile the images into movies on the same system.

1. Install Linux.

Note that installing Linux will wipe out any data on your PC. You can download the Ubuntu installer for free (ubuntu.com), or buy the DVD from Amazon. The installation tools will walk you through the entire process, asking you to select various application packages. You will likely want an OpenSSH server so you can remotely and securely log in, a web server such as Apache so that you can view your images remotely, and any scripting languages you may find useful, such as Python or Perl.

MATERIALS

Digital camera Free gPhoto software supports hundreds of cameras; check their website (gphoto.org) to see if yours is on the list. Even older digital cameras can take high-quality pictures, and a 4-megapixel camera exceeds the resolution required to make an HD movie.

Wide-angle conversion lens (optional) I want to capture as much of the sky as possible, so I bought a wide-angle conversion lens online for about $30. It fits right over the camera's existing lens and doubles the angle of view.

AC adapter Don't expect the USB connection from your computer to supply power to your compact digital camera! I found a cheap off-brand power supply online that's compatible with my camera.

PC A minimal installation of Linux, along with the gPhoto software, will not place great demands on your system; an old 600MHz Pentium III, for example, has more than enough horsepower.

USB active extension cables (optional) I picked up a couple for about $16 each. These can be daisy-chained together to reach distances of 60–80 feet (according to manufacturers' claims).

Enclosure I'm used a 6"×6"×6" steel enclosure made by Hoffman (hoffmanonline.com), part #A6R66NK.

Picture frame glass for the enclosure window. My local frame shop cut me a 5"×5" piece.

Slotted angle (optional) for mounting the enclosure. I used two 6' lengths. The mounting hardware you require may vary.

Miscellaneous hardware You'll need a handful of ¼-20 machine screws or hex bolts, washers, nuts and/or wing nuts, for putting it all together. If you need to fit your camera with a power-on setscrew (see Step 4, below), you'll also need a 3" aluminum mending brace, a 1" #8 machine screw, and a couple of matching nuts.

Dremel or your favorite metal cutting tool to modify the enclosure. There are many ways to cut sheet metal, but I used a rotary tool with a tungsten carbide cutting bit.

Silicone caulk

5-minute epoxy (optional) for attaching the setscrew bracket (again, see Step 4)

Uninterruptible power supply, aka UPS (optional) I want my system to survive short-term power outages, so I purchased an inexpensive UPS designed for desktops.

Software I used open source gPhoto software (gphoto.org), which runs on Unix-Linux operating systems. My flavor of choice is Ubuntu (ubuntu.com), which is about as easy to install as Linux gets. If you're not quite ready to jump into the Linux universe, there are great image capture and time-lapse programs for other operating systems. For Windows, there's GBTimelapse shareware (granitebaysoftware.com); it works only with Canon cameras.

2. Install additional software.

The easiest way to install software on Ubuntu is using its package manager system. To install gPhoto, log in and enter the command:

```
sudo apt-get install gphoto2
```

Likewise, you can install FFmpeg (which you'll use to assemble your images into a movie) like so:

```
sudo apt-get install ffmpeg
```

It's likely that you'll want a video player to view your creation. For Linux, there is VLC:

```
sudo apt-get install vlc
```

You may also choose to install other programs or libraries. For example, ImageMagick includes a number of line-command tools for manipulating images, and Image::EXIF is a Perl module for extracting EXIF data embedded in your image files.

3. Test your camera.

Before putting your camera in its enclosure, see how it works with the software. Connect your camera to the PC with its USB cable, and power it on. Log into your Linux box and enter the command:

```
gphoto2 --list-config
```

GPhoto will attempt to auto-detect your camera. If successful, it will output a list of configuration parameters. For many supported cameras, you can manipulate all the camera's settings just as if you were operating it by hand. For example, to set the image capture size to the lowest resolution:

```
gphoto2 --set-config imgsize=small
```

Some cameras may need to be identified explicitly, e.g.:

```
gphoto2 --camera "Canon PowerShot A520 (PTP mode)"
    --port usb: --list-config
```

Before capturing an image, Canon cameras must first extract the lens with the command:

```
gphoto2 --set-config capture=on
```

To capture 3 images at 10-second intervals, enter the command:

```
gphoto2 -F 3 -I 10 --capture-image-and-download
```

This will capture and download 3 image files into your current directory: *capt0000.jpg*, *capt0001.jpg*, and *capt0002.jpg*. If you have problems, the gPhoto site has good documentation.

4. Fix the power-up problem.

One problem I ran into with my Canon A520 was the fact that when its power supply is momentarily interrupted, restoring the power will not automatically turn the camera back on. Fortunately, I found
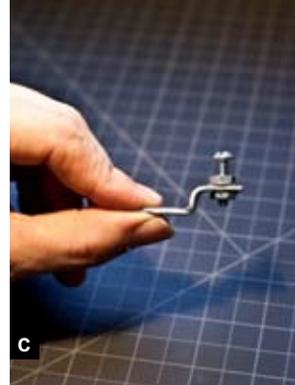
that my camera can be made to switch on on power-up if its power button is held in the depressed position. To do this, I fashioned a simple bracket and setscrew. I bent a 3" aluminum mending brace as shown in Figures A and B.

Now thread one nut partway down the #8 screw, insert it in a bracket hole, and thread the other nut on the other side, so that it looks like Figure C. Then, positioning the screw directly over the power button, glue it in place with 5-minute epoxy as in Figure D. Hold onto the nut between the brace and the camera, and turn the machine screw by hand until it begins to tighten against the power button (Figure E).

You should feel some resistance, and the camera should turn it on. Don't overtighten, as this could damage the camera. Tighten the top nut against the brace to lock the setscrew in place. (Figure F.)

Note that with the power button held down, you may not be able to operate the camera manually, as its buttons will be unresponsive. You will, however, be able to operate it remotely using the gPhoto software.

5. Install the enclosure window.

I removed the front cover of the enclosure, cut a 4"×4" window, and sealed the glass into place with the silicone adhesive (Figures G and H, following page).

6. Assemble and install the camera mount.

Camera tripod sockets accept a ¼-20 bolt, which makes it easy to improvise a mount with a single bolt, mounted through the floor of the enclosure. Use a hex nut to tighten a large washer against the bottom of the camera.

I want my camera pitched upward within the enclosure, to capture more of the sky, so my mount includes an L-bracket bent down to about 45° using a couple of pairs of vise-grips (Figure I). The camera is attached to the bracket using a ¼-20×1" machine screw, a big 1½" washer, and a nut. The other leg of the bracket is mounted to the enclosure floor.

If your enclosure has knockouts, you may want to use them for bringing your power cords and USB cables into the box; I just cut a notch at the front of the box floor for easy access with the enclosure cover door removed (Figure J).

7. Put it all together.

After assembling the enclosure outside, I plug my PC and the extension cord into the UPS, and run the power and USB cables out my window. I make sure the setscrew is properly set to ensure that the power is on, then plug in the AC adapter and stuff it,

BUILDING THE ENCLOSURE: Fig. G: Remove the front cover of the enclosure and cut a 4"×4" window. Fig. H: Seal the glass into place with silicone adhesive. Fig. I: Make a camera mount from a bent L-bracket, washer, screw, and bolt. **Fig. J: If your box doesn't have knock-outs for the UBB cable, cut a notch at the front of the box floor. Fig. K: Adjust the setscrew to ensure power is on, then plug in the AC adapter and stuff it into the enclosure. The enclosure is ready to be mounted.**

along with anything else that might need protection from weather, inside the enclosure (Figure K). Then I mount the enclosure on my roof.

9. Make time-lapse movies!

Hopefully you've pointed your camera at something interesting. To capture an image every 10 seconds for 1 hour, log into your Linux box (or open a terminal window), and enter the following command:

```
gphoto2 --F 360 -I 10 --capture-image
```

When it completes, you'll see that the current directory is filled with image files. Enter the command:

```
ffmpeg -f image2 -i capt%04d.jpg -sameq
   FirstMovie.mpg
```

This will assemble the images into an MPEG movie file. To view it, open it with VLC:

```
vlc FirstMovie.mpg
```

At the default 24 frames per second, this gives you a whopping 15 seconds! Of course with this setup you'll be able to make movies *much* longer than that. To tell gPhoto to run indefinitely until you interrupt it, set the `-F` flag to `0`:

```
gphoto2 -F 0 -I 10 --capture-image
```

This just scratches the surface. The gPhoto and FFmpeg programs are extremely powerful, and the possibilities are endless for creating beautiful time-lapse movies that span days, months, or even years.

Resources

- » **CHDK** — replacement firmware for some Canon cameras: chdk.wikia.com/wiki/CHDK
- » **gPhoto** — library and applications for Unix-like operating systems: gphoto.org
- » **GBTimelapse** — Windows shareware, for Canon cameras only: granitebaysoftware.com
- » **iStopMotion** — Mac shareware: boinx.com/istopmotion
- » **FFmpeg** — versatile image, audio, and video processing software. ffmpeg.mplayerhq.hu

➕ For additional online resources, go to makezine.com/go/TK.

Ken Murphy is a musician, programmer, artist, and tinkerer living in San Francisco. By day, he helps keep the website for his local public broadcasting station humming along. Ken is also the creator of "Blinkybugs," and he's currently working on a children's book and kit featuring these simple little electronic insects. You can follow the progress of his year-long time-lapse movie of the sky at his web site: murphlab.com/hsky.