161

AN INEXPENSIVE WEBCAM SYSTEM FOR CAPTURING LIVE SKYSCAPES AND TIME-LAPSE CLOUDSCAPES

Walker Ashley*, Philip Young, and Frederick Schwantes
Meteorology Program, Department of Geography, Northern Illinois University

1. INTRODUCTION

Webcams are a common medium for the visual display of live weather conditions. News media, weather companies, and institutions use webcams to attract viewers, promoting instantaneous atmospheric field trips to their camera sites. Unfortunately, webcam systems offered by commercial venders can cost thousands of dollars, pricing much of this technology out of reach for many educational institutions with tight budgets. The Meteorology Program in the Department of Geography at Northern Illinois University has developed an economical network of webcams delivering live, high-definition, wide-angle perspectives of Midwestern skyscapes. Employing scripts in addition to inexpensive or, in some cases, free software, we are able to generate time-lapse (i.e., a photography technique used to make movie footage using stills from a camera) sequences and daily time-lapse movies that can be used to visualize a range of atmospheric phenomena in classroom settings. We will discuss the requisite components and costs associated with the network, open-source tools and automation techniques for hands-off administration, construction of time-lapse movies using freeware, distribution options to video-sharing websites, system bugs and solutions, and instructional benefits of the system.

2. MOTIVATION

Interactive web-based scientific visualization tools, including time-lapse movies and animated gif images of cloudscapes, provide boundless educational promise. Time-lapse movies transform static visuals ─ whose motion may appear initially imperceptible or subtle to the eye ─ into striking features with discernible fluid motion. Time-lapse can facilitate exploration of complex phenomena found in the atmosphere and enhance student understanding of features that are often presented motionless in customary educational delivery methods, whether textbooks, lab manuals, or standard PowerPoint lectures. We have used time-lapse movies in introductory and upper-level atmospheric science courses to reveal features of meteorology inquiry, including: customary cloud classification and visual explanation of unique cloud type formation such as hole-punch clouds; differential advection; outflow and lake-breeze boundaries; frontal passage; gravity waves and undular bore; thunderstorm development and maturation; arcus clouds associated with convective systems (**Figure 1**); levels of lifting condensation, free convection, and equilibrium; diurnal boundary layer; convective temperature; stratus undulations; etc.

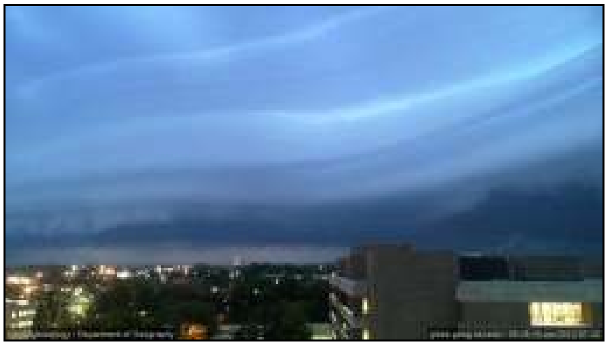


Figure 1. "West" webcam captures approaching shelf cloud associated with early-morning derecho-producing convective system on 24 July 2012. Time-lapse of this setting available at vimeo.com/46293492

We designed the system to: 1) enhance meteorology curriculum by engaging students through reflective observation using unconventional methods and 2) advance outreach efforts, such as spotter training, by improving understanding of complex severe weather phenomena through unique visualizations.

3. SYSTEM HARDWARE

Each constituent of the network cost less than $100; this includes a low-cost USB webcam offered by many online retailers or brick-and-mortar stores ($35-70), a suction-cup mount ($30), and USB extension cable ($10). We have experimented with Microsoft and Logitech webcams and have discovered that Microsoft cameras have superior lens quality and low-light capabilities. We recommend either the Microsoft Lifecam Studio (1080p, or up to 1920 x 1080 resolution) or the Microsoft Lifecam Cinema (720p, or up to 1280 x 720). We currently save images at 1280 x 720 resolution, which provides HD capability when constructing time-lapse movies. It is imperative that the webcam must have a focus mechanism that can be set at or near infinity with autofocusing mechanisms turned off. The "fixed" infinity setting promotes crisp views of skyscapes and removes unnecessary fluctuations in focusing that can diminish time-lapse quality. A suction-cup mount is a flexible solution permitting one to easily 1) secure the webcam to any window and 2) adjust the field of view. We use the GoPro suction cup mount with tripod adapter; however, any suction cup mount with a standard 1/4-20 tripod stud should work. The Microsoft Lifecam Studio has a standard tripod mount; however, the Cinema version does not requiring one to drill a hole through the rubber base to allow tripod mounting. Once the cameras are mounted, we suggest placing black electrical tape over the top of the webcam "on" light to remove any potential reflection in the window. We also suggest taping a black cloth (~18" x 18") to the window such that it falls behind the web cam; this prevents glare

* *Corresponding author address:* Dr. Walker Ashley, #118 Davis Hall, Meteorology Program, Dept. of Geography, Northern Illinois University, DeKalb, IL 60115; e-mail: washley@niu.edu

from hallway and office lights from causing unwanted reflections.

In order to save on equipment costs, we use second-hand desktop computers running Windows XP or 7 that are 3-6 years old and that have typically outlived their utility in faculty and graduate student offices (**Figure 2**). Because webcams can be somewhat resource intensive, we do suggest systems with at least 4GB of RAM. Since USB signal degrades with extension length, we do not recommend an extension cable longer than 6'. Depending on how far the computer is from the window, the setup may not require an extension cable.

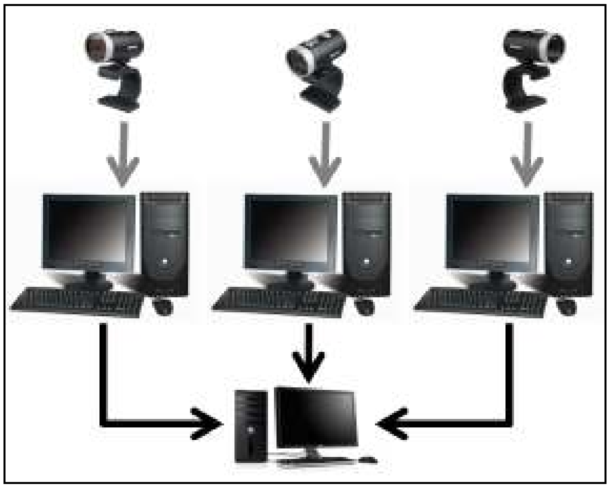
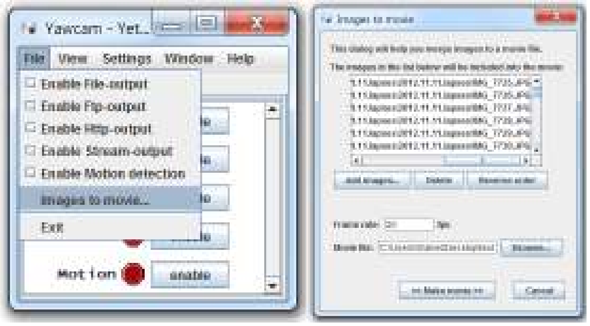


Figure 2. Webcam system configuration that includes three webcams feeding three second-hand desktops that FTP stills every 10 seconds, as well as daily time-lapse movies, to mid-grade desktop running Apache web server (httpd.apache.org).

4. SYSTEM SOFTWARE

Most webcams will require the installation of not only drivers, but supporting software that allows the user to modify settings such as focus, lightning, etc. While this software is compulsory, it typically does not have the option to save images at set intervals, add captions, FTP images, or other customizations necessary for a time-lapse system. Therefore, we recommend the use of third-party software that permits user customizations and image capture automation.

There are a number of third party webcam programs available that have a variety of options (e.g., Blue Iris, iSpy, Webcam Free, Conquer Cam, KABcam, Fwink Free, Willing Webcam, Digi Watcher); however, we have used primarily two software in the construction of our system ─ Tin Cam (http://www.tincam.com; free to try and $19 thereafter) and Yawcam (http://www.yawcam.com; free, but technically donationware). Both Tin Cam and Yawcam interface seamlessly with Microsoft, Logitech, and other modern webcams, allowing the capture of images at set intervals, consecutive image save, FTPing of images, text and timestamp overlay capability, among other options such as live video streaming (we have not been able to deliver live streaming due to department firewall restrictions). After months of testing both software, we have found that Tin Cam is the more stable (i.e., rarely, if ever, crashes) option of the two. However, Yawcam does have two features that Tin Cam does not have: 1) an easy-to-setup scheduler for those who only want to capture images during predetermined hours and days of the week, and 2) an efficient tool to create a time-lapse movies in QuickTime .mov format (**Figure 3**). The time-lapse option in Yawcam is astonishingly fast, permitting the construction of movie from 100s of still images in a matter of a few seconds. We use this feature to create short movie sequences to illustrate specific times of the day or features. We have created a script to automate the creation a daily movies, which we discuss later.



This dialog will help you merge images to a movie file.

The images in the list below will be included into the movie:

Frame rate: 20 fps

Movie file:

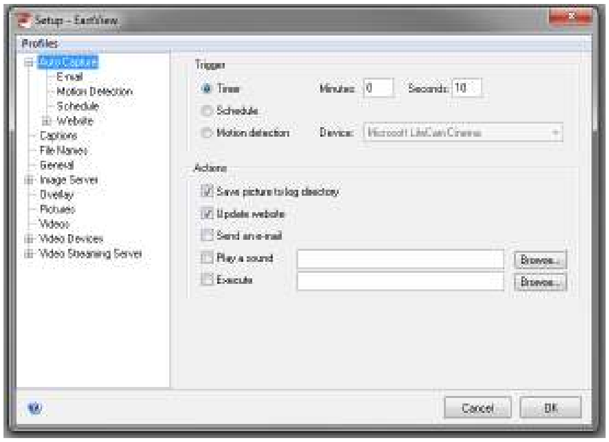
Figure 3. Yawcam interface demonstrating the "Images to movie" option that allows an user to quickly select a sequence of images, add a frame rate (frames per second, of fps), and generate a .mov format ready for playback in players that support that format (e.g., QuickTime, Windows Movie Player). This feature is useful for creating movies of specific features or set periods of the day. Alternatively, this software tool can be used to create lapses of any setting that includes a series of images shot using a DSLR, an intervalometer, and a tripod. The first author uses this method for time-lapse movie creation of cloudscapes acquired during storm chases.

5. TINCAM SOFTWARE

In this section, we discuss the setup and configuration of the Tin Cam v. 1.08 software. First, make sure all webcam drivers and supporting software are installed before setting up Tin Cam. Download and install the free trial of Tin Cam at www.tincam.com. When opening Tin Cam for the first time, a setup wizard will guide you through basic configuration. Once you have these basics implemented, you can customize the webcam and software configuration by selecting "Setup" in the menu (**Figure 4**). Webcam device selection, as well as video formatting and properties (e.g., focusing, brightness, white balance, etc.), is available under the "Video Devices" bullet. Within this menu selection, you can modify your device format; we currently use the 1280 x 760, 16bit option, but higher and lower resolutions are available for most cameras. We use Microsofts' "True Color" option under "Video Settings"; this automatically adjusts brightness, white balance, and saturation, which is a desirable feature for lapse due to continual changes in lighting during the day. Lapses will contain some "flicker" with this automated setting selected, although we find that the flicker is not a aesthetic detriment. There are anti-flicker programs available (e.g., LRTimelapse), but these add-ons and workflows are not for the novice or time constrained. Next, we recommend opening the "Auto Capture" bullet menu and setting the trigger to "timer" and at an update sequence of your choosing. We capture an image each 10 seconds on our system components and have found this interval provides a nice balance between system resource usage, hard drive storage, FTP efficiency, and final time-lapse movie size. Clearly, the shorter (longer) the interval between images, the more fluid and seamless (jerky) the time-lapse movie. Within the "Auto Capture" bullet we select the "save picture to log directory" and "update website" options to promote sequential save and FTP actions. Next, we select the "Website" and ensuing "FTP" sub-bullet. Under this option, we turn on the "Upload to FTP server" option. Under this menu, you should also setup your FTP server information, with port (normally 21), username, password, and upload directory. You can make sure your FTP configuration is working by selecting "Test settings". Subsequently, you can add automatic date and time commands (i.e., date and timestamp), as well as reference to your institution or hyperlink, via the "Captions" menu (note this information along the bottom of Figure 1). Under the "File Names" menu, we select the "Timed filenames" and point to a directory where we plan to store the images. We also use the "Website and Image Server" option to name the images as they would appear on our webserver (e.g., eastcam.jpg). Under the "General" tab, we select the "Load" and "Start auto capture" options to cause system startup upon computer reboot. We advise reducing the JPEG image quality to 90% under the "Pictures" tab to decrease FTP load and image storage size. We do not notice a significant drop in image quality at the 90% selection despite a rather large drop in file size. There are a number of other options and customizations in the program, but the above actions will provide the foundation for automatic FTP upload to a server for live image views and sequential image saves for later lapse creation. Once "setup" options are completed, make sure you select the "auto capture" option under the "Capture" menu button at the top of the program. The program can run hidden in the system tray, providing a clean desktop workspace. Yawcam software has many of the same options and customizations with a slight learning curve.

6. SYSTEM CONFIGURATION AND SCRIPTING

We have three computers with affiliated webcam and Ethernet/LAN connections in our system (**Figure 2**); this setup includes east, north, and west perspectives with each camera having a skyview factor of 70-80%. Persistent solar glare and lack of an unobstructed equatorward view restrict a fourth view to the south. The north- and west-facing cams are on our building's 7th floor locked in closet or lab space; the east view is in the first author's 2nd floor office. The webcams are placed on the interior side of windows (**Figure 5**); the arrangement precludes the need for outdoor enclosures and maintains easy camera maneuverability. We run a script shortly after sunset to remove all image files for that previous 24-hr period that were between twilight periods (i.e., nighttime). Set-aside daytime images are moved subsequently to a new folder with the date (i.e., YYYYMMDD_eastlapse) as the standard naming convention. The script then uses the free FFmpeg multimedia framework (http://ffmpeg.org) to convert files, sequence the files, and generate a movie (in our case, .mov format) of that day's images. The script then FTPs the movie file to the first author's server where an archive of all movies is available to viewers. A server is not necessary for lapse creation, but it does provide an easy distribution method such that anyone with internet access and the lapse archive directory URL can access and download movies for playback. Alternative to script lapse creation is the "Images to movie" option available in the Yawcam software (**Figure 3**).



Trigger

● Timer Minutes: 0 Seconds: 10

○ Schedule

○ Motion detection Device: Microsoft LifeCam Cinema

Actions

☑ Save picture to log directory

☑ Update website

□ Send an e-mail

□ Play a sound

□ Execute

Figure 4. Tin Cam "Setup" interface. Menu on left reveals the numerous configuration options; in this example, "auto capture" is set to take an image every 10 seconds, save it to a log directory, and update website (which also pushes FTP).

Those readers who are interested in the script discussed above can contact the first author for usage. For those who would rather not automate the lapse creation process, we recommend using a script that, at the minimum, places images from that day into a new folder. In this "cleanup" script (e.g., **Figure 6**), a new folder is created with the standard YYYYMMDD naming convention, with images from that day moved into that subdirectory. This cleanup script prevents Windows from having to manage thousands and thousands of images within a single directory; based on our experience, Windows simply cannot handle (at least efficiently) a directory that contains tens of thousands of individual files.

7. ALTERNATIVE DISSEMINATION METHODS

Beyond setting up a website with live images and links to movie creations on a server, we have discovered that lapses have more utility and exposure if social media (e.g., Facebook) and course management systems (e.g., Blackboard) are used in dissemination. We therefore advocate setting up either a Youtube or Vimeo account; these video-sharing websites provide the ability to store movies, hotlink movies on course management systems and social networking services, and easily bring up lapses on-the-fly such as in a classroom setting where students often query instructors on unique features they have witnessed. In addition, we have found great utility in selecting a short sequence of images illustrating a feature, creating an animated gif (using, e.g., Ulead GIF Animator), and including it within a PowerPoint lecture. These animated gifs bring lectures on atmospheric features to life by illustrating their naturally dynamic character.

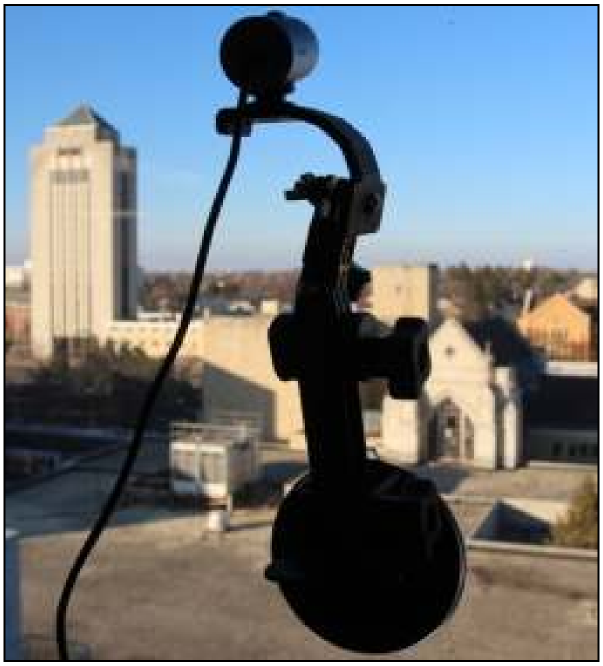


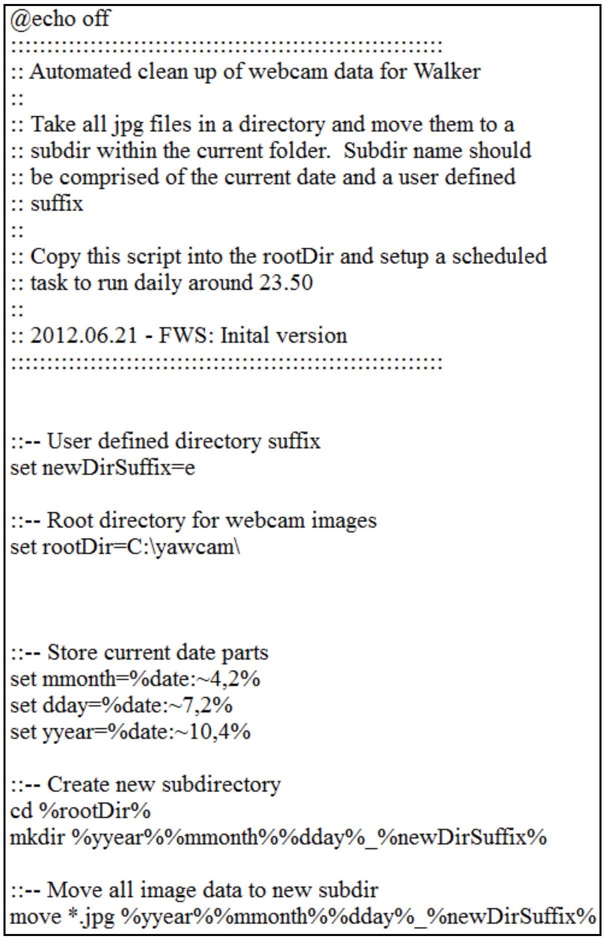
Figure 5. "North" webcam on window suction cup mount providing high-resolution images with 10 second updates.

8. SYSTEM BUGS AND SOLUTIONS

We have come across several system bugs and potential workarounds. First, in our experience, Yawcam would randomly lose the USB connection to the camera; in some cases, the software would work fine for weeks before dropping a connection several days in a row at arbitrary intervals. We never could determine the cause of the connection failure, but have found no such issues with the Tin Cam software. This connection failure associated with Yawcam was the primary reason we shifted to the Tin Cam software. Second, we have noticed that direct sunlight on our west cam can cause the focusing mechanism to change resulting in a blurry image. We believe this is due to the heating of the black outer body of the webcam, which, in turn, causes an adjustment in the focusing mechanism. Our solution has been to tape a piece of aluminum foil (or, alternatively, a split-in-half CD/DVD) atop the webcam to reflect incoming radiation.

9. CONCLUSION

Beyond the initial benefit of allowing students to visually reach outside the classroom walls to see current weather, we have discovered that the real power of this resourceful webcam system is when we take the archived webcam images and compose time-lapse movies. Somewhat mundane, static images of clouds can be collected in a timeline and sped up 2X, 20X, or even 200X to reveal the dramatic fluid motion of the atmospheric "ocean". These cloudscape clips often inspire awe in students (and instructors!), promoting an awareness of the highly dynamical nature of the atmosphere. These elements are normally subtle or elusive in real-time, but come to life when the time-lapse technique is applied. In the future, we will develop formal assessments to evaluate the utility of using time-lapse movies and gif animations in our meteorology curriculum.



```
@echo off
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:: Automated clean up of webcam data for Walker
::
:: Take all jpg files in a directory and move them to a
:: subdir within the current folder.  Subdir name should
:: be comprised of the current date and a user defined
:: suffix
::
:: Copy this script into the rootDir and setup a scheduled
:: task to run daily around 23.50
::
:: 2012.06.21 - FWS: Inital version
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::


::-- User defined directory suffix
set newDirSuffix=e

::-- Root directory for webcam images
set rootDir=C:\yawcam\



::-- Store current date parts
set mmonth=%date:~4,2%
set dday=%date:~7,2%
set yyear=%date:~10,4%

::-- Create new subdirectory
cd %rootDir%
mkdir %yyear%%mmonth%%dday%_%newDirSuffix%

::-- Move all image data to new subdir
move *.jpg %yyear%%mmonth%%dday%_%newDirSuffix%
```

Figure 6. A batch script that runs near midnight daily via the Windows' Scheduler to place images from that day into a new subdirectory. We run a more elaborate script that removes all nocturnal images and creates a timelapse movie using FFmpeg. Interested readers can contact the first author regarding script acquisition/usage.

Live skyscapes from our system may be found at: climate.niu.edu/cams.html or, a four-panel view, at chubasco.niu.edu/cams4p.html. Archive lapses may be found at: http://chubasco.niu.edu/lapse/ Please let us know via email (washley@niu.edu) if you setup a system so we can view your skyscapes!