Wildlife Society Bulletin 43(4):726–736; 2019; DOI: 10.1002/wsb.1016

*Tools and Technology*

# Efficacy of Automated Detection of Motion in Wildlife Monitoring Videos

**BRUCE G. MARCOT,**[1] *U.S. Forest Service, Pacific Northwest Research Station, 620 SW Main Street, Portland, OR 97205, USA*
**TERESA J. LORENZ,** *U.S. Forest Service, Pacific Northwest Research Station, 3625 93rd Avenue SW, Olympia, WA 98512, USA*
**PHILIP FISCHER,** *27291 US Highway 12, Naches, WA 98937, USA*
**BEN G. WEINSTEIN,** *Department of Fisheries and Wildlife, Marine Mammal Institute, Oregon State University, 2030 Marine Science Drive, Newport, OR 97365, USA*
**SAMUEL COWELL,** *Department of Biology, Utah State University, 5305 Old Main Hill, Logan, UT 84322, USA*

**ABSTRACT** Documenting biodiversity and wildlife behavior is time-consuming and potentially invasive. Remotely placed cameras often are used to increase sampling and minimize disturbance of focal animals. Such wildlife monitoring programs can entail numerous and lengthy videos requiring massive amounts of time to analyze their content. We evaluated the efficacy of a computer program, MotionMeerkat (http://benweinstein.weebly.com/motionmeerkat.html), for automated detection of motion in 21 continuous, 3-hour-long videos monitoring woodpecker (Picidae) nest cavities during 2016 in Washington, USA. The program produces still-image frame captures from the video where motion was detected, files logging the video time-stamps of each frame, and x,y coordinates within each frame where motion was detected. We compared program results with video reviews conducted by students using a standard protocol for identifying movement events and species identification. The program tagged 60% more motion events of all types, and 38% more motion events of target species (woodpeckers and potential predators), than were found by student reviews. We also conducted sensitivity analyses of the program by varying program input settings; results suggest that optimal settings depend on the video quality, environmental conditions, desired target species, and type of motion. We conclude that automated motion-detection programs such as MotionMeerkat can play important roles in use of continuous videos for wildlife monitoring, but results need to be reviewed by the researcher. 

**KEY WORDS** black-backed woodpecker, hairy woodpecker, motion detection, northern flicker, video monitoring, Washington, white-headed woodpecker, woodpecker breeding behavior.

Wildlife research and monitoring studies often use remote videos (Stewart et al. 1997, Bolton et al. 2007, Cox et al. 2012), commonly including motion-triggered video cameras (e.g., Allen et al. 2015) or still-photograph camera traps (e.g., Kelly 2008, Cotsell and Vernes 2016). Continuous-running or time-lapse video cameras have been usefully employed in a wide array of studies such as for detecting nesting of red tree voles (*Arborimus longicaudus;* Forsman et al. 2009), marine epifaunal invertebrates (Hemery and Henkel 2015), nest predation on passerines (Thompson et al. 1999, Rodewald et al. 2011, Lyons et al. 2015, DeGregorio et al. 2016), and nocturnally foraging burrowing owls (*Athene cunicularia;* Marsh et al. 2014). Recording systems are now inexpensive and widely available, and familiarity with and use of these systems have grown; therefore, the problem of researchers having an overwhelming amount of video of all forms of wildlife subjects to review calls for a tool that can help reliably automate video review.

When motion-triggered video cameras are unavailable or are unreliable due to sensor sensitivity, the researcher often is confronted with the onerous task of viewing continuous videos to locate specific activities that might appear only at long intervals and for brief instances (Evans et al. 2015). The tedium of such long viewing periods can demand many hours of concentrated focus, resulting in errors of missing or misidentifying key activities, and viewers must be trained to recognize target activities and avoid bias. The scale of such studies can be overwhelming, and viewer fatigue could introduce bias such as missing important movement events of target species. For example, in the current woodpecker (Picidae) research project, we monitored 118 nests daily with video, with varying numbers of hours per day, during the nesting season (~40 days) and for 3 years, totaling >42,358 hours of viewing; this would translate to >4,875 8-hour work days, although fatigue would rapidly set in after viewing and analyzing just 2 such 2-hour videos.

Received: 27 March 2018; Accepted: 3 July 2019
Published: 20 October 2019

[1]*E-mail: bmarcot@fs.fed.us*

A solution is to use automated algorithms to detect motion in continuous, prerecorded videos. This technology for wildlife application has advanced in recent years (Stauffer and Grimson 1999, Laube and Purves 2006, Tekalp 2015, Nazir et al. 2017), with, for example, an array of algorithms for identifying motion (Karasulu 2010, Fang et al. 2016), specific shapes such as of target species (Wawerla et al. 2009, Zeppelzauer 2013, Priya and Domnic 2014, Tack et al. 2016, Valletta et al. 2017), face features (Burghardt and Ćalić 2006), 3D movement patterns of insects (Kastberger et al. 2011), and semantic context of motion events (Yong et al. 2013). National Institutes of Health, Research Services Branch, has developed Image Processing and Analysis in Java (ImageJ; https://imagej.nih.gov/ij/) that includes various plug-in codes to assist in automated analysis of micrographs of biological materials, including image comparisons such as used for motion detection. Additionally, many applications are available, on many platforms, for real-time motion detection of live video feeds (e.g., Bolton et al. 2007) including use of smartphones (Steen 2017). We focused here on detecting motion in continuous, prerecorded videos not taken with motion-detection technology.

We tested the efficacy and useful settings of a software system called MotionMeerkat (http://benweinstein.weebly.com/motionmeerkat.html) designed to identify and tag user-specified motion in prerecorded videos to obviate the need for real-time viewing of each video (Weinstein 2015). We initially explored use of other programs designed to detect motion in prerecorded videos, including Safeware Technologies' SafeMotion© (www.safeware.ca), VideoLAN's VLC media player© (www.videolan.org), library toolbox motion-detection extensions to MathWorks' MatLab© (www.mathworks.com), and others including security surveillance systems. However, none had user-defined parameters and flexibility for use with wildlife monitoring projects, particularly the capacity to specify motion sensitivity, object movement speed and size, portion of the frame in which to detect motion, and especially the ability to save video frames and write databases listing frames and time stamps where motion was detected, as MotionMeerkat does.

Our purpose was to determine the efficacy of using this motion-detection program to aid analysis of continuous (not motion-activated) videos used for monitoring wildlife activity, and help determine best settings through a sensitivity analysis. We define efficacy in terms of efficient analysis time and lowest error rates of detecting motion events.

## STUDY AREA

Videos for our analysis were collected during a broader study on woodpecker nest predation at 2 study sites, approximately 2,000 and 3,500 ha in size, in Yakima County, Washington, USA (approximately 46°45′N, 120°58′W). Both sites were on the U.S. Forest Service (USFS) Okanogan–Wenatchee National Forest. We did not specifically intend to restrict our study to USFS land, but it happens that most woodpeckers nest in recently burned forest in this region and USFS has had an active prescribed burn program for >10 years (e.g., USFS 2010, 2013). Forest composition varied based on aspect, slope, elevation, and longitudinal distance from the crest of the Cascade Range. Dominant tree species were ponderosa pine (*Pinus ponderosa*), Douglas-fir (*Pseudotsuga menziesii*), and grand fir (*Abies grandis*). Less common tree species included western larch (*Larix occidentalis*), lodgepole pine (*Pinus contorta*), subalpine fir (*A. lasiocarpa*), quaking aspen (*Populus tremuloides*), and black cottonwood (*Populus trichocarpa*). We estimated that ≥24% of the area within our study sites had been burned with mixed-severity prescribed fire within 10 years of the start of this study. One site was grazed by domestic sheep (*Ovis aries*) in summer and one by domestic cattle (*Bos taurus*). We used climate data from PRISM (Parameter-elevation Regressions on Independent Slopes Model; http://www.prism.oregonstate.edu/normals/) to determine the 30-year mean precipitation level in the study area is 85.8 cm/year, most of which falls as snow in winter, and that the mean January temperature is −1.65° C and the mean July temperature is 16.02° C. For 2016, precipitation totaled 86.5 cm for the year, mean January temperature was −1.42° C, and mean July temperature was 15.87° C. Climate values are based on the center of the study area.

## METHODS

### Video Monitoring Study

Videos used in the current work were derived from our broader woodpecker predation study in which we searched for woodpecker nests from April through July, 2016, using systematic nest searches and behavioral cues by woodpeckers (Jackson 1976). We selected the 4 most common woodpeckers on our sites as target species: black-backed woodpecker (*Picoides arcticus*), hairy woodpecker (*P. villosus*), northern flicker (*Colaptes auratus*), and white-headed woodpecker (*P. albolarvatus*). This broader study provided the videos by which we could test efficacy of the automated motion-detection program.

At 10 nests, we set high-definition video cameras (models HC-V130 and HC-V160; Panasonic Corporation of North America, Newark, NJ, USA) powered with lithium polymer batteries (model 1055275-2 C; Mogen Industrial Limited, Shenzhen, Guangdong, China) for continuous recording. Our intent was to record all daytime activities at nests until the woodpeckers fledged or the nest failed. We excluded night footage because a pilot study indicated most putative predators visited nests during the day. In fact, that study revealed no predation events or nest predator visits at night (Lorenz and Fischer 2018). We set cameras >20 m (typically 30–50 m) from nests to minimize disturbance to woodpeckers. Forests in our study area were open with virtually no shrub or grass cover >15 cm tall. Cameras had an unobstructed view of nest cavities and of the snag bole above and below the cavity, as well as the area behind and on either side of the nest without us having to disturb the surrounding vegetation such as by cutting shrub, grass, or forb cover. We purposefully did not approach nest snags closer than required to set up cameras, with the exception of white-headed and black-backed woodpecker nests in which we captured nestlings within 2 days of fledging for other research objectives.

We elevated cameras 3–5 m above ground in trees and snags to minimize human theft and disturbances from elk (*Cervus canadensis*), which often damage cameras within their reach. We enclosed cameras and batteries in black plastic acrylonitrile butadiene styrene drain pipes to protect them from precipitation.

We recorded video in Advanced Video Coding High Definition (AVCHD; MTS file extension) or Moving Picture Experts Group (MPEG; MP4 file extension) formats. Cameras recorded AVCHD files in approximately 2-hour segments and MPEG files in 20-minute segments. We visited cameras daily to replenish batteries and memory cards. Cameras recorded in all weather and on all days, except for rare cases when a malfunction or low battery caused the camera to stop before we were able to revisit the site.

### Student Processing of Videos

We selected videos for study review using a stratified random sample design based on days from incubation, early nestling, and late nestling stages of the target woodpecker species. We trained students to follow our protocol for video review. Briefly, we instructed students to watch the video at 1–2 × speed. In this protocol, the students recorded on paper data sheets all activities of vertebrate animals observed in the videos, noting the video time stamp when any animal entered the video as the start of a motion event, duration of the visit, species observed, and animal behavior in specified categories. The students then transcribed that information to computer spreadsheets. As needed, students paused videos to record start and end times for behaviors and replayed portions of videos for identifying animals and behaviors. Each video was watched by only one student.

### Automated Motion Detection in Videos

We used MotionMeerkat version 2.0.5 (Weinstein 2015), a freely available automated motion-detection program designed for ecological monitoring (http://benweinstein.weebly.com/motionmeerkat.html). MotionMeerkat operates by scanning an existing video for frame-to-frame differences using several background-subtraction methods (KaewTraKulPong and Bowden 2002; Zivkovic 2004, 2006; Godbehere et al. 2012) as coded and available in OpenCV (Open Source Computer Vision Library, opencv.org). Where differences are found as an indication of movement in the video, the program saves the associated video frames as JPEG format files and compiles all findings into a comma-separated value (CSV format) data file denoting each frame's video time stamp, frame number, and x,y coordinate position on the frame where the difference (i.e., the movement) occurred. The saved video JPEG files can then be viewed as thumbnails or postprocessed into an animation so as to view all motion-captured frames as a continuous sequence itself, which helps interpret the content and context of the motion events. Outputs from the program also include a log file listing various input parameter settings, processing time and frame rate, the number of candidate motion events, and other summary outcomes. The program is run through a graphical user interface by specifying a single video or a folder with multiple videos, input parameters as discussed below, and an output folder location for results.

We defined a motion event as a continuous set of video frames in which some object or objects differed in position, frame to frame, similar to the "hit window" defined by Villette et al. (2016) in their video camera-trap study of small mammals. For example, a motion event could consist of a woodpecker flying into the frame and alighting on the bole of a snag. If the bird then sat motionless for >1 frame, and then moved again, this would mark the start of another motion event. MotionMeerkat would not identify the frames where the bird was motionless yet still visible, but would identify the point where it began moving again. Other motion events tagged by the program could include nontarget birds or insects flying into or through the frame, movement of tree branches in the wind, or other spurious movements. We ran MotionMeerkat on 5 computers using Microsoft Windows® (Microsoft Corporations, Redmond, WA, USA) operating systems and a variety of memory capacities, graphics adapters, and processor speeds (Table 1).

### Comparison with Manual Motion Detection

We tested the efficacy of MotionMeerkat in detecting motion by comparing the program results with the students' manual viewing of each video, as follows. We analyzed 21 videos from the woodpecker study from 9 locations, taken by cameras running unattended in fixed positions to monitor use of cavity trees. All selected videos were recorded in AVCHD format, were approximately 2 hours in length, and recorded at 30 frames/second (fps) and 1,920 × 1,090 pixel resolution. MotionMeerkat initially starts with default parameter settings that we changed as our "normative" input settings for each video (Table 2), including the "draw motion objects" option that bounds moving objects with a red box in the frame-capture image files. We compiled the following output results: each motion event detected by the program, the type of motion event (target wildlife species or other motion), the duration of the motion event, and the specific time stamp on the video of each motion event.

We then compared the MotionMeerkat results with analyses of the same videos viewed in real time by the trained students. How well MotionMeerkat identified all motion events including desired events informed us on the program's efficiency. We tallied the number and duration of all motion events, and of motion events of the target study species only (Table 3), as detected by students and by the program for each video. We developed a confusion table, which is a matrix summarizing classification error rates (Kohavi and Provost 1998), for each video that denoted frequencies of the following: Type I errors or false positives, which are when MotionMeerkat tagged the start of a

**Table 1.** Computers used for running the MotionMeerkat motion-detection program for videos recorded in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016.

| Attribute | Computer 1 | Computer 2 | Computer 3 | Computer 4 | Computer 5 |
|---|---|---|---|---|---|
| Make | Home made | HP | HP | HP | Dell |
| Model | n/a | Window7 x64 SCCM Image 2016 | 7.2 | EliteBook HP 840 G1 | XPS 8700 |
| Operating system | Windows 10 Professional | Windows 7 Enterprise | Windows 7 Enterprise | Windows 7 Enterprise | Windows 7 Professional |
| Central processing unit (CPU) type | Intel Core i7 920 | Intel Core2 Duo | Intel Core2 Duo | Intel Core i5-4300U | Intel Core i7-4790 |
| CPU speed (GHz) | 2.67 | 2.40 | 2.40 | 1.90 | 3.60 |
| Installed RAM (GB) | 12 | 4 | 4 | 16 | 16 |
| System type | 64-bit | 64-bit | 64-bit | 64-bit | 64-bit |
| Graphics adapter type | NVIDIA GeForce 9500 GT | ATI Mobility Radeon X2300 | ATI Mobility Radeon X2301 | AMD Radeon HD 8750 M | NVIDIA GeForce GTX 750 Ti |
| Graphics adapter total available graphics memory (MB) | 4,095 | integrated[a] | integrated[a] | integrated[a] | 4,096 |
| Graphics adapter shared system memory (MB) | 3,071 | 4,096 | 4,096 | integrated[a] | 2,048 |
| How used in this study | Main analysis | Main analysis | Main analysis | Sensitivity analysis[b] | Sensitivity analysis[c] |

[a] Graphics adapter integrated with system board, separate information not available.
[b] Used to conduct all sensitivity analyses.
[c] Used only to cross-check sensitivity analysis Run 01 (see Table 7).

**Table 2.** MotionMeerkat input parameter names, values, and functions; and normative input settings used for videos recorded in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016.

| Parameter name (value range) | Parameter function | Setting used |
|---|---|---|
| Basic settings | | |
| Background variation (0–5) | Sets the tolerance threshold in the mixture of Gaussian detector. The slider controls the mixture of Gaussion learning rate. The true value (hidden from the user) ranges from 0 (no learning) to 1. At 0, the background mask never changes. There is a trade-off between accuracy and efficiency with the learning rate. As the learning rate increases, you run the risk of missing slow-moving objects, and learning too fast, such that little changes become part of the background. If the learning rate is too low, you will return frames for even insignificant movement. | 3 |
| Organism movement (0–5) | The slider changes the speed of background updating. Increasing the slider will make MotionMeerkat less sensitive to abrupt changes in the background and reduce the number of frames returned. It is a companion to the first slider and sets the degree to which new pixels can be different from the background. | 3 |
| Minimum object size [0,1] | The smallest size of the contour box to be considered. Expressed in percentage of the total frame. For example, minSIZE = 10 says disregard any motion contour that has an area smaller than 10% of the frame. This filter helps remove small pixel movements. It should be increased slowly. Higher values remove contour objects that do not meet the minimum size. Users often overdo this parameter; it should be increased slowly. This argument is ignored if 'Draw your object size on screen' is selected. | 0.0003 |
| Crop area of motion (Y/N) | Draw a blue box for a restricted area of motion detection. Anywhere outside the box, the detector will ignore movement. If set, the first window of the video will pop up, and the user can use right click to draw a rectangle around the area to include. After drawing the rectangle, hit ESC to exit the screen. The resulting first frame, with the excluded area indicated with a white box, will flash briefly. The output image will still have the whole frame, but any motion outside the excluded area will be ignored. | N |
| Advanced settings | | |
| Adapt (Y/N) | This is the ratio of frames expected to have motion events for adaptive averaging. Value expressed as a ratio, so 0.1 is 10% of frames. The software will reduce sensitivity if exceeds this level on a 10-min interval. | Y |
| Draw motion objects (Y/N) | The motion objects will be drawn with a red bounding box. | Y |
| Delay motion detection (min) | A delay in starting the motion detection. Often useful to ignore camera setup or deployment. Time is given in minutes. Frames are completely ignored, and are not included in output statistics. | 0 |
| Scan video (frames) | Down-sample frame rate by scanning X frames (0). High-definition video data (often 30 frames/sec and higher) will take a long time to process. Indeed the entire point of MotionMeerkat is that events of interest are rare. For this reason, it is often extremely helpful to scan for motion by jumping frames, and then computing every frame once motion is found. The down-sampling arguments take in an integer for how many frames to skip. For example if I entered 5, MotionMeerkat would skip 5 frames for every 1 it analyzed. Once it found motion, it would analyze the next frame without skipping. This will greatly reduce computational time. | 0 |
| Remove singles (frames) | Removes single video frames from the analysis; if the frames before and after a potential motion event was not returned, do not return the frame to user. This was done to help remove small blips in the background. | 0 |

**Table 3.** Target species of interest in the woodpecker monitoring video study (Supporting Information 1, available online) in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016. All entries in this table were identified in the videos in this study, although only the woodpecker species and their potential nest predators were of focal attention for the woodpecker study.

| Code | Species or event | Target species? |
|---|---|---|
| AMRO | American robin (*Turdus migratorius*) | no |
| BBWO | Black-backed woodpecker (*Picoides arcticus*) | yes |
| CAFI | Cassin's finch (*Haemorhous cassinii*) | no |
| CHIP | Chipmunk species (*Tamias minimus*, *T. amoenus*, or *T.* [*Neotamias*] *townsendii*) | yes |
| COYO | Coyote (*Canis latrans*) | no |
| DEJU | Dark-eyed junco (*Junco hyemalis*) | no |
| ELK | Rocky Mountain elk (*Cervus canadensis nelsoni*) | no |
| GMGS | Golden-mantled ground squirrel (*Callospermophilus lateralis*) | yes |
| HAWO | Hairy woodpecker (*Picoides villosus*) | yes |
| HOWR | House wren (*Troglodytes aedon*) | no |
| HUMA | Human (*Homo sapiens*) | no |
| NOFL | Northern flicker (*Colaptes auratus*) | yes |
| STJA | Steller's jay (*Cyanocitta stelleri*) | no |
| UNBI | Unknown bird species | no |
| UNKN | Unknown species or events (e.g., flying insects, branches moving in the wind) | no |
| UNRO | Unknown rodent species | yes |
| UNSO | Unknown small songbird species | no |
| WEBL | Western bluebird, *Sialia mexicana* | no |
| WETA | Western tanager (*Piranga ludoviciana*) | no |
| WEWP | Western wood pewee (*Contopus sordidulus*) | no |
| WHWO | White-headed woodpecker (*Picoides albolarvatus*) | yes |
| WISA[a] | Williamson's sapsucker (*Sphyrapicus thyroideus*) | no |

[a] The Williamson's sapsucker code appeared only once, in the student reviews of the videos, likely as a misidentification of northern flicker.

motion event that was not identified by direct viewing; Type II errors or false negatives, which are when MotionMeerkat failed to tag the start of a motion event that was identified by direct viewing; the total of both error types; and these same errors tallied for any kind of motion versus motion only of the target species or events of interest (Table 3). False negatives can occur when >1 object is moving at the same time; the program does not differentiate the objects and treats it as the same motion event. The other object moving could be another organism, or could be swaying of branches in the wind, etc.

We also compared the duration of individual motion events as identified through student review with those identified through MotionMeerkat. This analysis helped clarify the basic differences in motion event detection and definition between manual and program review. We expected differences where a target species would initially exhibit some movement, such as appearing from a cavity or flying into the frame, and then remain motionless but visible, only to move again. In such situations, the student review would treat this sequence as one motion event, although MotionMeerkat would see it as 2 independent events.

### Tests of Environmental and Video Length Effects

We also tested whether environmental conditions during the video recordings, and whether video length, may have influenced any differences between the students' manual reviews and program's automated reviews. We denoted each video with 2 environmental conditions under which each was taken—lighting and weather. Lighting conditions pertained to videos taken during daytime, dusk or dawn with limited light for at least one-quarter of the video run time, and late dusk with limited light throughout the video. An additional lighting condition that we did not test was when nest trees were severely backlit. Weather conditions pertained to clear, windy, or rainy situations, where we defined windy subjectively when there was obvious and large movement of tree branches and other vegetation. We used analysis of variance with Bonferroni-adjusted *P*-values and Lavene's test based on means, using the program Systat® v13.1 (Systat Software, Inc., Chicago, IL, USA) to compare outcomes to the potential influence of lighting and weather conditions and video runtime length.

### Sensitivity Analysis of Program Operation

We also tested how varying the input parameters of MotionMeerkat influenced its performance and capacity to detect known and desired motion events, in a structured sensitivity analysis. We selected 3 videos each representing calm wind conditions to reduce spurious motion events (Table 4), and ran each video through a set of 20 different combinations of the program's 9 input parameters that group into 4 basic settings and 5 advanced settings (Table 2). We varied input settings individually and in tandem where each combination represented reasonable settings that one might choose (Table 5). For most sensitivity runs, we set minimum object size to 0.0003, which corresponds to the minimal proportion of the video frame size in which woodpeckers or potential predators (e.g., sciurids) would appear. We then mostly varied the basic setting parameters incrementally because they influenced how motion would be detected; the advanced settings largely pertain to how motion-detected frames are captured and catalogued. We limited the test to 3 videos when results converged and to avoid unnecessary computation time that exceeded 326 central processing unit (CPU) hours for the 3 videos and 20 sensitivity runs/video (Table 6).

## RESULTS

For the analyses, we used 21 videos of 2-hour duration each, totaling 42 hours and 4,536,000 individual frames at 30 fps. These videos represented a range of environmental and weather conditions presenting challenges to the automated motion-detection program. This set included a variety of subject matter, such as motion events, including primary target events of woodpeckers, chipmunks, raptors, and other species; and other spurious motion events, such as insects, nontarget birds or mammals, and wind moving tree branches. Specifying the program to draw motion objects resulted in movements highlighted with red bounding boxes in the video frame captures (Table 2; Fig. 1).

**Table 4.** Attributes of 3 video files used in the sensitivity analyses for videos recorded in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016. The codec in all videos was H264 - MPEG-4 AVC (part 10) (h264).

| File attribute | Video file 1 | Video file 2 | Video file 3 |
|---|---|---|---|
| Name | calm_00014.avi | calm_00077.mts | calm_00150.mts |
| Size (GB)[a] | 3.81 | 3.99 | 3.99 |
| Length (hh:mm:ss)[b] | 1:54:46 | 2:02:46 | 2:02:46 |
| Resolution (pixels)[c] | 1,920 × 1,090 | 1,920 × 1,090 | 1,920 × 1,090 |
| Frame rate (frames/sec)[c] | 25.00 | 29.97 | 29.97 |
| Total no. frames, calculated[d] | 172,150 | 220,759 | 220,759 |
| Total no. frames, MotionMeerkat log[e] | 172,132 | 220,770 | 220,557 |

[a] Size as noted in Microsoft Windows Explorer.
[b] Length as noted in the runtime window of the VLC media player (www.videolan.org/vlc/)
[c] Resolution as noted in VLC / Tools / Codec information.
[d] Total no. frames calculated as: [length (sec)] × [frame rate (frames/sec)].
[e] From MotionMeerkat log file resulting from analysis run.

**Table 5.** MotionMeerkat settings used in the sensitivity analyses for videos recorded in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016. See Table 2 for description of parameters. * Indicates settings deviating from their default program values. In run 16, area of motion was cropped to just the nest tree to exclude potentially extraneous motion. Note that run 19 uses all default values, and run 20 pertains to the normative settings used in the main analysis runs. It was determined that the last 3 advanced settings parameters did not pertain to the type of analysis performed, so they were not varied from their default settings.

| | Basic settings | | | | Advanced settings | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Run number | Back-ground variation (0–5) | Organism movement (0–5) | Min. object size [0,1] | Crop area of motion? (Y/N) | Adapt? (Y/N) | Draw motion objects? | Delay motion detection (min) | Scan video (frames) | Remove singles (frames) |
| 1 | 3 | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 2 | 0* | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 3 | 1* | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 4 | 2* | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 5 | 4* | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 6 | 5* | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 7 | 3 | 0* | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 8 | 3 | 1* | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 9 | 3 | 2* | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 10 | 3 | 4* | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 11 | 3 | 5* | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 12 | 3 | 3 | 0.0002* | N | Y | N | 0 | 0 | 0 |
| 13 | 3 | 3 | 0.0002* | N | Y | N | 0 | 0 | 0 |
| 14 | 3 | 3 | 0.0005* | N | Y | N | 0 | 0 | 0 |
| 15 | 3 | 3 | 0.0006* | N | Y | N | 0 | 0 | 0 |
| 16 | 3 | 3 | 0.0003* | Y* | Y | N | 0 | 0 | 0 |
| 17 | 3 | 3 | 0.0003* | N | N* | N | 0 | 0 | 0 |
| 18 | 3 | 3 | 0.0003* | N | Y | N | 0 | 0 | 0 |
| 19 | 3 | 3 | 0.1000 | N | Y | N | 0 | 0 | 0 |
| 20 | 3 | 3 | 0.0003* | N | Y | Y* | 0 | 0 | 0 |

**Table 6.** Outcomes of MotionMeerkat sensitivity analyses (Table 5) for each video file (Table 4) and summed over all files for videos recorded in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016; Σ = sum, $\bar{x}$ = mean, CV = coefficient of variation; n/a = not applicable. Values exclude sensitivity analysis Run 2 (Table 5) that incurred internal program errors, and that resulted in nearly all frames being selected with motion.

| | Video file 1 | | | Video file 2 | | | Video file 3 | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| Parameter | Σ | $\bar{x}$ | CV | Σ | $\bar{x}$ | CV | Σ | $\bar{x}$ | CV | Σ |
| Run time (Central processing unit [min]) | 4,920.3 | 233.3 | 2.4 | 7,906.3 | 341.1 | 46.7 | 6,768.9 | 293.9 | 1.24 | 19,596 min (326.6 hr) |
| Frames/sec | n/a | 12.3 | 2.3 | n/a | 11.9 | 21.5 | n/a | 12.5 | 1.2 | n/a |
| Candidate motion events (no. frames) | 107,910 | 871 | 37.7 | 226,511 | 308 | 34.7 | 232,886 | 659 | 28.0 | 567,307 |
| No. frames skipped, insufficient motion | 3,253,204 | 171,219 | 0.16 | 4,188,603 | 220,447 | 0.04 | 4,177,758 | 219,872 | 0.1 | 11,619,565 |
| No. frames analyzed | 3,361,912 | 171,132 | 0.0 | 4,415,400 | 220,770 | 0.0 | 4,411,140 | 220,557 | 0.0 | 12,188,452 |
| No. frames picked | 3,678 | 194 | 130.5 | 4,161 | 219 | 46.2 | 10,187 | 536 | 34.7 | 18,026 |
| No. motion frame segments | 2,469 | 130 | 94.7 | 3,223 | 170 | 40.3 | 6,636 | 349 | 28.9 | 12,328 |


**Figure 1.** Examples of video frames saved by MotionMeerkat with various species and motion events of interest (Table 3) using program settings noted in Table 2 in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016. The program draws red boxes around the detected motions; the inclusion and minimal size of the box are specified by the user. a) Two black-backed woodpeckers at a cavity snag. Here, the program tagged one in motion but not the other perched motionless. b) Example of spurious motion events, in this case flying insects that were not the target organism of interest. The user can adjust the degree of sensitivity of motion to include or exclude such events. c) Northern goshawk approaching a northern flicker nest (not a depredation event). d) Adult northern flicker exiting from a nest cavity. e) Chipmunk entering a vacated white-headed woodpecker nest cavity. Only the tail was in motion in this frame. f) White-headed woodpecker at nest cavity entrance. Only the head was in motion in this frame.

## Efficacy of Automated Motion Detection

Among all 21 videos and all motion event types, student reviews recorded 325 discrete motion events ($\bar{x} = 15.5 \pm 8.0$ SD/video), whereas MotionMeerkat detected 518 events ($\bar{x} = 24.7 \pm 17.7$ SD/video) or 60% more. Among target-species motion event types, student reviews recorded 242 motion events ($\bar{x} = 11.5 \pm 8.4$ SD/video), and MotionMeerkat caught 334 events ($\bar{x} = 15.9 \pm 14.0$ SD/video) or 38% more. MotionMeerkat recorded a greater variation among the videos in the number of motion events as compared with results from the student reviews (Table 7). Differences between student-review and program results

**Table 7.** Analysis of 21 videos comparing student-reviewed results with MotionMeerkat (MM) program results, for all motion event categories and for target study species for videos in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016 (Table 3).

| Analysis parameter | Total | $\bar{x}$ | Min. | Max. | SD[a] | CV[b] | SE[c] |
|---|---|---|---|---|---|---|---|
| No. of motion events detected—All motion event categories | | | | | | | |
| Student review | 325 | 15.5 | 6 | 32 | 8.0 | 52% | 1.7 |
| MotionMeerkat | 518 | 24.7 | 3 | 71 | 17.7 | 72% | 3.9 |
| All | 586 | 27.9 | 6 | 72 | 18.7 | 67% | 4.1 |
| Difference[d] | 193 | 9.2 | −3 | 39 | 9.7 | 105% | 2.1 |
| No. of motion events detected—Target study species only | | | | | | | |
| Student review | 242 | 11.5 | 2 | 32 | 8.4 | 73% | 1.8 |
| MotionMeerkat | 334 | 15.9 | 2 | 54 | 14.0 | 88% | 3.1 |
| All | 373 | 17.8 | 2 | 61 | 15.4 | 87% | 3.4 |
| Difference[d] | 92 | 4.4 | 0 | 22 | 5.6 | 128% | 1.2 |
| Classification error analysis—All motion event categories[e] | | | | | | | |
| Correct classification[f] | 44% | 49% | 21% | 100% | 21% | 44% | 0.88 |
| Type I error[g] | 45% | 35% | 0% | 71% | 23% | 67% | 0.96 |
| Type II error[h] | 11% | 14% | 0% | 50% | 13% | 99% | 0.55 |
| Total error[i] | 56% | 48% | 0% | 79% | 24% | 49% | 0.97 |
| Species mismatch error | 62% | 55% | 0% | 82% | 23% | 42% | 0.96 |
| Classification error analysis—Target study species only[e] | | | | | | | |
| Correct classification[f] | 54% | 62% | 27% | 100% | 28% | 44% | 1.43 |
| Type I error[g] | 35% | 27% | 0% | 67% | 22% | 80% | 1.12 |
| Type II error[h] | 10% | 10% | 0% | 60% | 16% | 149% | 0.81 |
| Total error[i] | 45% | 38% | 0% | 73% | 27% | 73% | 1.42 |
| Species mismatch error | 49% | 40% | 0% | 73% | 29% | 71% | 1.48 |

[a] SD = standard deviation.
[b] CV = coefficient of variation.
[c] SE = standard error of the mean.
[d] Difference = no. of motion events from MotionMeerkat minus the no. of motion events from student review.
[e] Error rates pertain to mismatches of the time stamp for the start of each motion event.
[f] Correct classification rate = motion start events noted by student review and MM with the same video time stamp (±10 sec).
[g] Type I error = motion start events caught by MM but not by student review.
[h] Type II error = motion start events caught by student review but not MM.
[i] Total error = Type I + Type II error rates.

occurred, in part, because of how MotionMeerkat would tag the start of a new motion event even if the same object remained in the field of view but stopped and then started moving again. Other differences occurred because MotionMeerkat could capture spurious motion events such as insects or nontarget birds, whereas they would be ignored or not detected in the student reviews.

Across all 21 videos and all 586 motion events caught by either student or program review, 45% of motion events were caught by MotionMeerkat but not by student review (Type I error) and 11% were caught by student review but not by MotionMeerkat (Type II error), for a total error rate of 56%. Among the 373 target species motion events caught by either student or program review, 35% of motion events were caught by MotionMeerkat but not by student review (Type I error) and 10% were caught by student review but not by MotionMeerkat (Type II error), for a total error rate

**Table 8.** Durations (seconds, s) of individual motion events as viewed by students compared with results of MotionMeerkat, across all 589 motion events collectively identified from both evaluations from 21 videos, for all motion event categories and for target study species in Yakima County, Washington, USA, on the Okanogan–Wenatchee National Forest during 2016 (Table 3). Differences are measured as motion event durations from MotionMeerkat program results minus durations from student reviews, across all motion events across all videos ($n = 589$ motion events) or as noted for each video ($n = 21$ videos).

| | Motion event duration (s) | | | | |
|---|---|---|---|---|---|
| Parameter | Student review | MotionMeerkat | Motion event duration difference (s) | Total event difference (s) | Mean event difference (s) |
| All motion event categories | | | | | |
| Total | 58,842 | 32,852 | −25,990 | −25,990 | −1,488 |
| $\bar{x}$ | 100 | 56 | −44 | −1,238 | −71 |
| Min. | 0 | 0 | −4,592 | −6,453 | −767 |
| Max. | 4,602 | 4,260 | 4,260 | 4,841 | 336 |
| SD | 418.5 | 304.7 | 474.3 | 3,179.5 | 233.7 |
| CV | 416.7 | 543.5 | −1,069.5 | −256.9 | −329.8 |
| SE | 17.3 | 12.6 | 19.6 | 693.8 | 51.0 |
| *N* | 586 | 586 | 586 | 21 | 21 |
| Target study species only | | | | | |
| Total | 56,785 | 27,009 | −29,776 | −29,776 | −3,802 |
| $\bar{x}$ | 152 | 72 | −80 | −1,418 | −181 |
| Min. | 0 | 0 | −4,592 | −6,406 | −2,296 |
| Max. | 4,602 | 4,260 | 4,260 | 3,971 | 336 |
| SD | 515.6 | 358.8 | 581.5 | 2,970.0 | 533.8 |
| CV | 338.7 | 495.5 | −728.4 | −209.5 | −294.8 |
| SE | 26.7 | 18.6 | 30.1 | 648.1 | 116.5 |
| *N* | 373 | 373 | 373 | 21 | 21 |

of 45% (Table 7). Error rates of denoting motion events and of species identification varied widely across all 21 videos (Table 7).

Although the student reviews resulted in fewer numbers of motion events, the total motion event durations noted by students were longer than with MotionMeerkat outcomes (Table 8). For all motion event categories, student reviews resulted in 79% greater motion event durations than did the program, and 110% greater for target species motion events only. Differences occurred because the student reviews of motion events typically included durations when an object, including a target species, would remain in view but were motionless, whereas MotionMeerkat would not tag such conditions as movements.

Use of MotionMeerkat was highly time-efficient including the few minutes for set-up time to run each video, and time to review the results. Review time of the program output for each video was approximately 10 minutes, which included scanning output frame-capture images in thumbnails and scrolling through them quickly in sequence, and checking specific time-stamped points in the original video for further details on selected motion events. In contrast, manually viewing 2-hour videos also entailed recording and transcribing each event, which totaled on average ≥2.5 hours work time. Summed across all 21, 2-hour videos, the difference was approximately 3.5 hours to review program output compared with approximately 52.5 hours to do fully manually processing. Using the program saved >90% of evaluation time.

## Tests of Environmental and Video Length Effects

Twelve videos were taken in daylight, 8 in early- to mid-crepuscular dawn or dusk, and 1 in darker, late-dusk conditions (Pearson $\chi^2_2 = 8.86$, $P = 0.012$). Eight were taken in clear weather, 9 in windy conditions, and 4 in rainy conditions (Pearson $\chi^2_2 = 2.00$, $P = 0.37$). We found no relationship between video comparison attributes of number of motion events detected and error rates, and duration of individual motion events, for all events and for target species (Tables 7, 8), with lighting conditions (all $P > 0.06$) and with weather conditions (all $P > 0.07$). We found no relationship (all $P = 1.000$) of video comparison attributes with video length (Supporting Information, available online).

## Sensitivity Analysis

The computation time for conducting the 20 sensitivity runs among 3 videos totaled 326.6 CPU hours, in which >12 million video frames were analyzed (Table 6). MotionMeerkat ran at approximately 12 fps, identifying >0.5 million frames with candidate motion events and >12,000 motion frame segments. Outcomes (Supporting Information) resulted in only one apparent relationship, where the output parameter of number of frames skipped based on insufficient movement was correlated with the input setting of organism movement ($n = 20$ tests, $r = 0.73$, $P = 0.03$). Although not strong relationships ($P > 0.05$), learning rate and average frames per second were positively related and total run time was negatively related to the input parameter background variation. Also, the number of candidate motion events, hit rate, total number of frames picked, and number of movement frame segments were negatively related to the input settings related to the movement of the organism and minimum object size (see Table 2 for definitions). In general, run duration was longer with greater scene complexity and with slower and more detailed learning rates, which could result in spurious motion events being identified by the program. Learning rate refers to the program's comparison of sequential frame images for detecting differences, which it interprets as movement. High learning rates, specified with values >0.2 in the program's input settings, may help detect quickly moving objects but may miss slowly moving objectives.

# DISCUSSION

There is an increase in the need for, and utility of, automated motion-detection programs for *in situ* wildlife studies (e.g., Kalcounis-Rueppell et al. 2013). Clearly, a program that can supplant many thousands of hours of direct video viewing can save researchers valuable time, effort, and cost. Although motion detection is hardly a new technology having been used for decades in surveillance systems, the application we tested in this study seems unique in its user-friendly graphical user interface, its capacity for tuning settings for specific video content, and producing individual video frame image captures and log files detailing where motion was found within a video stream.

## Lessons from the Analysis

To our knowledge, this is a novel comparison test and sensitivity analysis of a motion-detection program for use in wildlife monitoring. Results suggest that there is no one set of optimal program settings, nor did we attempt to identify them, because different videos taken under various environmental conditions for different target movements would entail a different set of input settings for most efficient results.

Study objectives will be an important determinant of program settings. In our woodpecker study, it was more important to minimize Type II errors (false negative findings from the program) over Type I errors (false positive findings), assuming that results from the program would then be checked and sorted for relevant study targets. For example, one study objective was to identify instances and species of nest predators. For this objective, the error of the program missing potential instances would be more egregious than would be the error of the program finding, and then our reviewing, motion events that may turn out to be nontarget or other spurious sources. Indeed, this analysis resulted in a total Type II error rate of 11%, which we found to be acceptably low because it entailed a minimal amount of additional cross-checking with the videos.

We found no influence of video lighting conditions, weather conditions, and video length on the comparison outcomes, although additional testing may be warranted for videos taken in darkness. This suggests that MotionMeerkat provides unbiased results compared with student-reviewed results regardless of environmental conditions or the length of the video recording, within the conditions observed in this test.

## Overall Conclusions and Considerations

We suggest that an automated motion-detection program such as MotionMeerket can play a vital role in review of continuous-motion videos, for several key reasons. First, it is time-efficient. Although MotionMeerkat analyzes a video at approximately 1 × viewing rate, student reviews of the same videos, including time for training, setup, and documenting results, generally took far longer, and the program can run unattended in batch mode to automate the analysis of many multiple files sequentially. Second are problems associated with live viewing of videos, including viewer fatigue, variation among viewers in their attention and accuracy of species identification, the need for developing and implementing viewing protocols, the need for viewer training and proctoring, and other considerations. Prolonged and overuse of computer screens results in Computer Vision Syndrome, which is a real outcome experienced in this project when reviewing the videos (Blehm et al. 2005). Despite asserting viewing protocols and monitoring of student review sessions, errors in species identification and missing entire movement events still occurred and required revisiting some videos. Third is the human bias of focusing on an area of interest within the video frame, particularly the nest cavities. In this study, viewing bias may have resulted in missing some important background activities, whereas the machine-viewing does not discriminate in the way a human would.

Automated motion-detection programs would be exceptionally valuable in studies where researchers are using continuous video recordings to identify cavity-nest predators (Lorenz and Fischer 2018). In such situations there likely would be many hours of no activity, punctuated only by occasional and brief appearances of predators. In the present study, we caught several such instances that would otherwise have gone undetected, by running MotionMeerkat overnight in batch mode on multiple videos. For studies of open-cup nesters, however, there may be considerable motion, both of target species visible in the nest and of superfluous sources such as the wind moving branches. For such conditions, the input parameters of the program can be adjusted to ignore background motion, to focus only on the nest itself, and other adjustments. See Weinstein (2015) for further details on program settings and use.

However, automated motion-detection programs are not panaceas. Motion detection makes no attempt to classify which objects are creating the movement signature (e.g., Weinstein 2018), and as such, may result in extraneous frames due to unwanted background movement as well as unwanted foreground movement such as by nontarget animals or humans. Also, discrete movement events could be, in part, an artifact of how the program clips events when a target organism remains in view but is motionless. Thus, results from motion-detection programs, including MotionMeerkat, still need to be reviewed, sorted for relevant target events, and interpreted with caution, particularly when evaluating their number and duration. On the other hand, although the program was proficient in identifying motion, it tags multiple, simultaneously moving targets as just a single continuous movement event. In general, how automated motion-detection programs identify individual motion events may not correspond to how a human viewer would define and identify events, and thus requires subsequent review.

Another consideration pertains to the extensive CPU time needed for automated review of many and lengthy videos.

This may itself become a constraining factor if researchers have access to limited computational power, particularly CPU and graphics processing unit speeds, and drive storage capacity. However, given adequate computer power and capability, the best use of a motion-detection program is as a valuable time-saving advisory tool producing interim results to be cross-checked by the researcher.

The best program settings for running MotionMeerkat and other motion-detection programs will depend on study objectives, conditions under which videos were taken, attributes of the target organisms of interest, and other factors. In general, the best or recommended field and environmental conditions for such videos if used by an automated motion-detection program would consist of avoiding spurious motion events caused by wind and other weather events, with adequate natural or artificial lighting, and with adequately high video frame rates and shutter speeds so as to effectively identify any movement source. With such considerations accounted for, automated motion-detection programs such as MotionMerkat can play valuable roles in continuous-video wildlife studies.

## ACKNOWLEDGMENTS

We thank the many undergraduate students who reviewed video for this project at Utah State University and Evergreen State College. Funding was provided by U.S. Forest Service Pacific Northwest Research Station. Thanks to J. St. Hilaire, R. Huffman, and G. Mackey for logistical support of field work, and to K. Sullivan and L. Lauren for logistical support in working with the students. Our thanks to P. H. Bloom and F. R. Thompson III for their helpful reviews. Mention of commercial products does not necessarily denote endorsement by the United States government.

## LITERATURE CITED

Allen, M. L., L. M. Elbroch, C. C. Wilmers, and H. U. Wittmer. 2015. The comparative effects of large carnivores on the acquisition of carrion by scavengers. American Naturalist 185:822–833.

Blehm, C., S. Vishnu, A. Khattak, S. Mitra, and R. W. Yee. 2005. Computer vision syndrome: a review. Survey of Ophthalmology 50:253–262.

Bolton, M., N. Butcher, F. Sharpe, D. Stevens, and G. Fisher. 2007. Remote monitoring of nests using digital camera technology. Journal of Field Ornithology 78:213–220.

Burghardt, T., and J. Ćalić. 2006. Analysing animal behaviour in wildlife videos using face detection and tracking. IEE Proceedings—Vision, Image and Signal Processing 153:305–312.

Cotsell, N., and K. Vernes. 2016. Camera traps in the canopy: surveying wildlife at tree hollow entrances. Pacific Conservation Biology 22:48–60.

Cox, W. A., M. S. Pruett, T. J. Benson, S. J. Chiavacci, and F. R. Thompson III. 2012. Development of camera technology for monitoring nests. Pages 185–210 *in* C. A. Ribic, F. R. Thompson III, and P. J. Pietz, editors. Video surveillance of nesting birds. Studies in Avian Biology No. 43. University of California Press, Berkeley and Los Angeles, USA.

DeGregorio, B. A., S. J. Chiavacci, T. J. Benson, J. H. Sperry, and P. J. Weatherhead. 2016. Nest predators of North American birds: continental patterns and implications. BioScience 66:655–665.

Evans, D. R., S. L. McArthur, J. M. Bailey, J. S. Church, and M. W. Reudink. 2015. A high-accuracy, time-saving method for extracting nest watch data from video recordings. Journal of Ornithology 156:1125–1129.

Fang, Y., S. Du, R. Abdoola, K. Djouani, and C. Richards. 2016. Motion based animal detection in aerial videos. Procedia Computer Science 92:13–17.

Forsman, E. D., J. K. Swingle, and N. R. Hatch. 2009. Behavior of red tree voles (*Arborimus longicaudus*) based on continuous video monitoring of nests. Northwest Science 83:262–272.

Godbehere, A. B., A. Matsukawa, and K. Goldberg. 2012. Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation. Pages 4305–4312 *in* 2012 American Control Conference. Institute of Electrical and Electronics Engineers, Montreal, Quebec, Canada.

Hemery, L. G., and S. K. Henkel. 2015. Patterns of benthic mega-invertebrate habitat associations in the Pacific Northwest continental shelf waters. Biodiversity and Conservation 24:1691–1710.

Jackson, J. A. 1976. How to determine the status of a woodpecker nest. Living Bird 15:205–221.

KaewTraKulPong, P., and R. Bowden. 2002. An improved adaptive background mixture model for real-time tracking with shadow detection. Pages 135–144 *in* P. Remagnino, G. A. Jones, N. Paragios, and C. S. Regazzoni, editors. Video-based surveillance systems. Springer, New York, New York, USA.

Kalcounis-Rueppell, M., T. Parrish, and S. Pauli. 2013. Application of object tracking in video recordings to the observation of mice in the wild. Pages 105–115 *in* J. Rychtář, S. Gupta, R. Shivaji, and M. Chhetri, editors. Topics from the 8th Annual UNCG Regional Mathematics and Statistics Conference. Springer Proceedings in Mathematics & Statistics, 64. Springer, New York, New York, USA.

Karasulu, B. 2010. Review and evaluation of well-known methods for moving object detection and tracking in videos. Journal of Aeronautics and Space Technologies 4:11–22.

Kastberger, G., M. Maurer, F. Weihmann, M. Ruether, T. Hoetzl, I. Kranner, and H. Bischof. 2011. Stereoscopic motion analysis in densely packed clusters: 3D analysis of the shimmering behaviour in giant honey bees. Frontiers in Zoology 8(3). https://doi.org/10.1186/1742-9994-8-3

Kelly, M. J. 2008. Design, evaluate, refine: camera trap studies for elusive species. Animal Conservation 11:182–184.

Kohavi, R., and F. Provost. 1998. Glossary of terms. Journal of Machine Learning 30:271–274.

Laube, P., and R. S. Purves. 2006. An approach to evaluating motion pattern detection techniques in spatio-temporal data. Computers, Environment and Urban Systems 30:347–374.

Lorenz, T. J., and P. C. Fischer. 2018. Cameras show sciurids visiting white-headed woodpecker nests without depredating contents. Journal of Fish and Wildlife Management 9(1):e1944–687X.

Lyons, T. P., J. R. Miller, D. M. Debinski, and D. M. Engle. 2015. Predator identity influences the effect of habitat management on nest predation. Ecological Applications 25:1596–1605.

Marsh, A., E. M. Bayne, and T. I. Wellicome. 2014. Using vertebrate prey capture locations to identify cover type selection patterns of nocturnally foraging burrowing owls. Ecological Applications 24:950–959.

Nazir, S., S. Newey, R. J. Irvine, F. Verdicchio, P. Davidson, G. D. Fairhurst, and R. van der Wal. 2017. WiseEye: next generation expandable and programmable camera trap platform for wildlife research. PLoS ONE 12(1):e0169758.

Priya, G. G. L., and S. Domnic. 2014. Shot based keyframe extraction for ecological video indexing and retrieval. Ecological Informatics 23:107–117.

Rodewald, A. D., L. J. Kearns, and D. P. Shustack. 2011. Anthropogenic resource subsidies decouple predator–prey relationships. Ecological Applications 21:936–943.

Stauffer, C., and W. E. L. Grimson. 1999. Adaptive background mixture models for real-time tracking. Pages 246–252 *in* Proceedings of the IEEE Computer Society Technical Committee on Pattern Analysis and Machine Intelligence. Volume 2. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Los Alamitos, California USA.

Steen, R. 2017. Bird monitoring using the smartphone (iOS) application. Videography for motion detection. Bird Study 64:62–69.

Stewart, P. D., S. A. Ellwood, and D. W. Macdonald. 1997. Remote video-surveillance of wildlife—an introduction from experience with the European badger *Meles meles*. Mammal Review 27:185–204.

Tack, J. L. P., B. S. West, C. P. McGowan, S. S. Ditchkoff, S. J. Reeves, A. C. Keever, and J. B. Grand. 2016. AnimalFinder: a semi-automated system for animal detection in time-lapse camera trap images. Ecological Informatics 36:145–151.

Tekalp, A. M. 2015. Digital video processing. Prentice Hall Press, Upper Saddle River, New Jersey, USA.
Thompson, F. R. III, W. D. Dijak, and D. E. Burhans. 1999. Video identification of predators at songbird nests in old fields. Auk 116:259–264.
U.S. Department of Agriculture, Forest Service [USFS]. 2010. Gold Spring restoration project, Naches Ranger District, Okanogan–Wenatchee National Forest environmental assessment. U.S. Forest Service, Wenatchee, Washington, USA.
U.S. Department of Agriculture, Forest Service [USFS]. 2013. Nelli restoration project, Naches Ranger District, Okanogan–Wenatchee National Forest environmental assessment. U.S. Forest Service, Wenatchee, Washington, USA.
Valletta, J. J., C. Torney, M. Kings, A. Thornton, and J. Madden. 2017. Applications of machine learning in animal behavioural studies. Animal Behaviour 124:203–220.
Villette, P., C. J. Krebs, T. S. Jung, and R. Boonstra. 2016. Can camera trapping provide accurate estimates of small mammal (*Myodes rutilus* and *Peromyscus maniculatus*) density in the boreal forest? Journal of Mammalogy 97:32–40.
Wawerla, J., S. Marshall, G. Mori, K. Rothley, and P. Sabzmeydani. 2009. BearCam: automated wildlife monitoring at the Arctic Circle. Machine Vision and Applications 20:303–317.
Weinstein, B. G. 2015. MotionMeerkat: integrating motion video detection and ecological monitoring. Methods in Ecology and Evolution 6:357–362.
Weinstein, B. G. 2018. Scene-specific convolutional neural networks for video-based biodiversity detection. Methods in Ecology and Evolution 9:1435–1441.
Yong, S.-P., J. D. Deng, and M. K. Purvis. 2013. Wildlife video key-frame extraction based on novelty detection in semantic context. Multimedia Tools and Applications 62:359–376.
Zeppelzauer, M. 2013. Automated detection of elephants in wildlife video. EURASIP Journal on Image and Video Processing 2013:46.
Zivkovic, Z. 2004. Improved adaptive Gausian mixture model for background subtraction. Pages 28–31 *in* J. Kittler, M. Petrou, and M. S. Nixon, editors. Proceedings of the 17th international conference on pattern recognition. Volume 2. The Institute of Electrical and Electronics Engineers Computer Society Press, Los Alamitos, California, USA.
Zivkovic, Z. 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters 27:773–780.

*Associate Editor: McRoberts.*

## SUPPORTING INFORMATION

Additional supporting information may be found in the online version of this article at the publisher's web-site. We provide results of the sensitivity analyses of videos used by MotionMeerkat. Also available are examples of full videos used in this study as well as results of their analysis using MotionMeerkat.