ISPRS Journal of Photogrammetry and Remote Sensing 152 (2019) 132–144

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Data extraction from digital repeat photography using *xROI*: An interactive framework to facilitate the process

Bijan Seyednasrollah[a,b,c,*], Thomas Milliman[d], Andrew D. Richardson[b,c]

[a] Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA 02138, USA
[b] School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86001, USA
[c] Center for Ecosystem Science and Society, Northern Arizona University, Flagstaff, AZ 86001, USA
[d] Earth Systems Research Center, University of New Hampshire, Durham, New Hampshire 03824, USA

ARTICLE INFO

Keywords:
Digital repeat photography
xROI
ROI
Time-series
Phenology
PhenoCam

ABSTRACT

Digital repeat photography and near-surface remote sensing have been used by environmental scientists to study environmental change for nearly a decade. However, a user-friendly, reliable, and robust platform to extract color-based statistics and time series from a large stack of images is still lacking. Here, we present an interactive open-source toolkit, called *xROI*, that facilitates the process of time series extraction and improves the quality of the final data. *xROI* provides a responsive environment for scientists to interactively (a) delineate regions of interest (ROI), (b) handle field of view (FOV) shifts, and (c) extract and export time series data characterizing color-based metrics. The software gives user the opportunity to adjust mask files or draw new masks, every time an FOV shift occurs. Utilizing xROI can significantly facilitate data extraction from digital repeat photography and enhance the accuracy and continuity of extracted data.

1. Introduction

Although the idea of repeat photography to study environmental change goes back a century (Stephens et al., 1987; Turner, 2003), using digital repeat photography has become increasingly popular to monitor and study the environment for a diverse range of applications such as studying plant phenology (Berra et al., 2019; de Moura et al., 2017; Olivera-Guerra et al., 2017; Richardson et al., 2018b; Sonnentag et al., 2012; Watson et al., 2019; Yan et al., 2019), assessing the seasonality of gross primary production (Crimmins and Crimmins, 2008; Migliavacca et al., 2011; Yuan et al., 2018), salt marsh restoration (Knox et al., 2017), monitoring tidal wetlands (O'Connell and Alber, 2016), investigating growth in croplands (Liu and Pattey, 2010; Zhou et al., 2013), and evaluating phenological data products derived from satellite remote sensing (Richardson et al., 2018c; Seyednasrollah et al., 2018). However, extracting "clean" and high quality data from a large set of images often presents three main challenges: (a) delineating region of interests (ROI) (Richardson et al., 2018a), (b) computational costs (Filippa et al., 2016a); and (c) handling expected and unexpected field of view (FOV) shifts (Brown et al., 2016; Moore et al., 2016). All three issues require careful consideration. Currently, these steps are often performed in separate, fully supervised stages. An integrated portable environment with which the user can interactively manage and extract high quality time series would significantly improve data collection for environmental studies.

Obtaining quantitative data from digital repeat photography images is usually performed by defining appropriate ROI's and, for the red (R), green (G) and blue (B) color channels, calculating pixel value (intensity) statistics across the pixels within each ROI. ROI boundaries are delineated by mask files which define which pixels are included and which are excluded from these calculations. User-friendly software libraries to delineate user-defined ROI's interactively are scarce and commonly require commercial licenses (e.g. ENVI, MATLAB Image Processing Toolbox). Additionally, the data extraction process is usually performed in another environment, the process requires adequate familiarity with scripting languages (e.g. R (Team, 2018), MATLAB (MathWorks, 2015), Phenopix R package (Filippa et al., 2016b), Python (Sanner, 1999) and third-party plugins (Sunoj et al., 2018)), or the tools are not suitable for general image datasets (Bradley et al., 2010). Thus, an interactive platform with an easy-to-use graphical user interface that can integrate ROI delineation and time series extraction is highly desired.

Camera field of view shifts will result in pixels or areas outside of the original region of interest falling into the masked area, which can

* Corresponding author at: Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA 02138, USA. School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86001, USA.
E-mail address: seyednasrollah@fas.harvard.edu (B. Seyednasrollah).

https://doi.org/10.1016/j.isprsjprs.2019.04.009
Received 12 July 2018; Received in revised form 16 April 2019; Accepted 16 April 2019

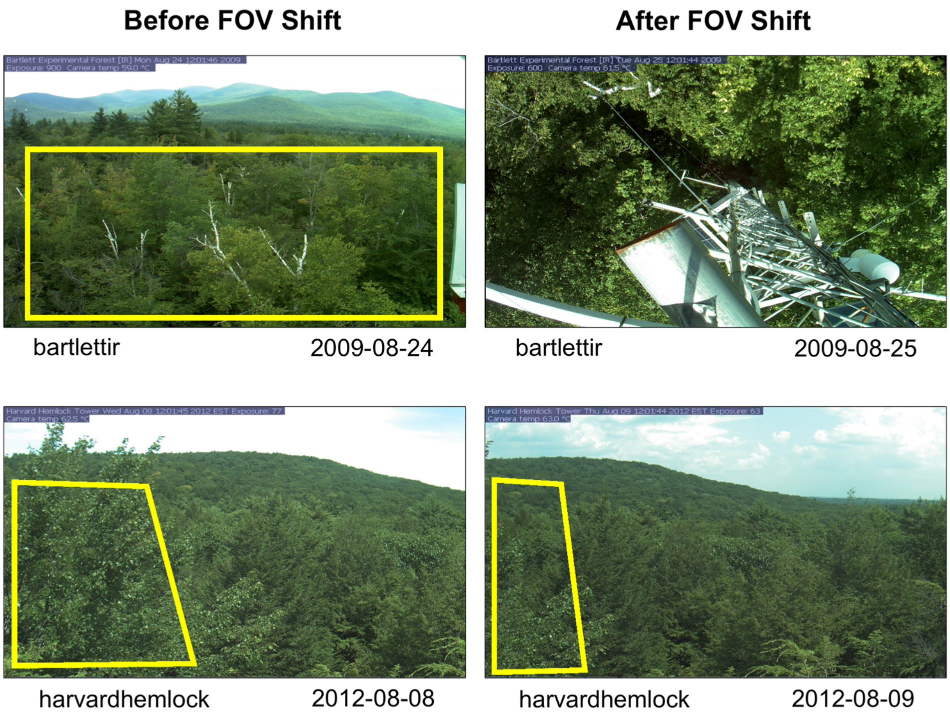

Fig. 1. Two examples of field of view (FOV) shift at the bartlettir and harvardhemlock PhenoCam sites. At bartlettir, the original region of interest after the shift was entirely outside of the field of view and the post-shift FOV was not relevant for the study. At harvardhemlock, FOV shift was minor and redrawing the region of interest fixed the issue.

cause low-quality or even misleading data. Fig. 1 shows two examples of FOV shifts from the PhenoCam network (http://phenocam.sr.unh.edu) that was founded in 2008 to study vegetation phenology across ecosystems of North America using near-surface remote sensing (Richardson, 2018). After a FOV shift occurrence (e.g. Fig. 1), the corresponding ROI and mask files should be adjusted (minor shift: ROI is still in FOV but has moved), redrawn (major shifts: ROI is partially in FOV) or stopped processing (entire FOV has changed). However, detecting FOV shifts is not a trivial task, particularly for large stacks of digital images (e.g., 35 million images of the PhenoCam archive (Seyednasrollah et al., 2019)). Correlation based methods (e.g. phase correlation or binary correlations) (Gottumukkal and Asari, 2004) or distance-based methods (e.g. Manhattan distance) (Dhodapkar and Smith, 2003) has been developed for facial recognition, object detection and tracking techniques, but they often fail to perform a satisfactory job on landscape images (e.g. composition of canopy and sky). Moreover, most of these methods are computationally expensive and require calibration and learning steps (such as site-specific tuning). Therefore, a simple and quick method to detect FOV shifts could further speed-up high quality data extraction and management.

Here, we present an interactive, portable and robust framework, called *xROI*, with a simple graphical user interface (GUI) with which the user can define regions of interest (ROI's), monitor FOV shifts and extract color-based statistical metrics for a stacked set of digital images. *xROI* is an R package that can run on several operating systems. Our toolkit facilitates the entire process by several orders of magnitude, reduces human-based errors, and improves data continuity and reproducibility.

2. Application development

The ***R*** language and ***Shiny*** package (Chang et al., 2017) were used as the main development tools for *xROI*, while ***Markdown*** (Baumer et al., 2014), ***HTML*** (Aronson, 1995), ***CSS*** (Powell, 2010)and ***JavaScript*** (Mikkonen and Taivalsaari, 2007) languages were used to improve interactivity. *R* is an interpreted computer language which is increasingly popular among environmental scientists. *Shiny* is an add-on *R* package that provides a powerful platform for development of web-based applications (*Shiny* apps) in *R*. *Shiny* apps generally include three main elements: (1) the user interface (*UI*), (2) the server-side engine; and (3) the auxiliary functionalities. The *UI* is the element in which the appearance features and graphical user interface are designed. The server element is the engine built to interpret user responses and react accordingly. Most of the processing and computation steps are performed inside the server element, while general set-up and intermediate functions are accommodated inside the auxiliary functionalities. Although *Shiny* apps are primarily used for web-based applications hosted on a web server to be used online, we used *Shiny* for its graphical user interface capabilities. In other words, both UI and server modules are run locally from the same machine and hence no internet connection is required. The *xROI's UI* element presents a side-panel for data entry and three main tab-pages, each responsible for a specific task. The server-side element consists of R and shell scripts. Image processing and geospatial features were performed using the Geospatial Data Abstraction Library (*GDAL*) (Warmerdam, 2008) and the ***rgdal*** (Bivand et al., 2018) and ***raster*** (Hijmans, 2017) R packages.

The *xROI* R package has been published on The Comprehensive R Archive Network (CRAN). The latest tested *xROI* package can be installed from the CRAN packages repository by running the following command in an R environment:

```
utils::install.packages('xROI').
```

Alternatively, the latest beta release of *xROI* can be directly downloaded and installed from the package *GitHub* repository:

```
devtools::install_github('bnasr/xROI'),
```

however, this requires that the necessary *R* packages and *GDAL* library have already been installed on the local system.

xROI depends on many R packages including: ***colourpicker***, ***data.table***, ***jpeg***, ***lubridate***, ***moments***, ***plotly***, ***raster***, ***RCurl***, ***rgdal***, ***rjson***, ***shiny***, ***shinyAce***, ***shinyBS***, ***shinydashboard***, ***shinyFiles***, ***shinyjs***, ***shinyTime***, ***shinythemes***, ***sp***, ***stringr***, and ***tiff***. All the required libraries and packages will be automatically installed with installation of *xROI*. The package offers a fully interactive high-level interface as well as a set of low-level functions for ROI processing. A comprehensive user manual for low-level image processing using *xROI* is available from https://

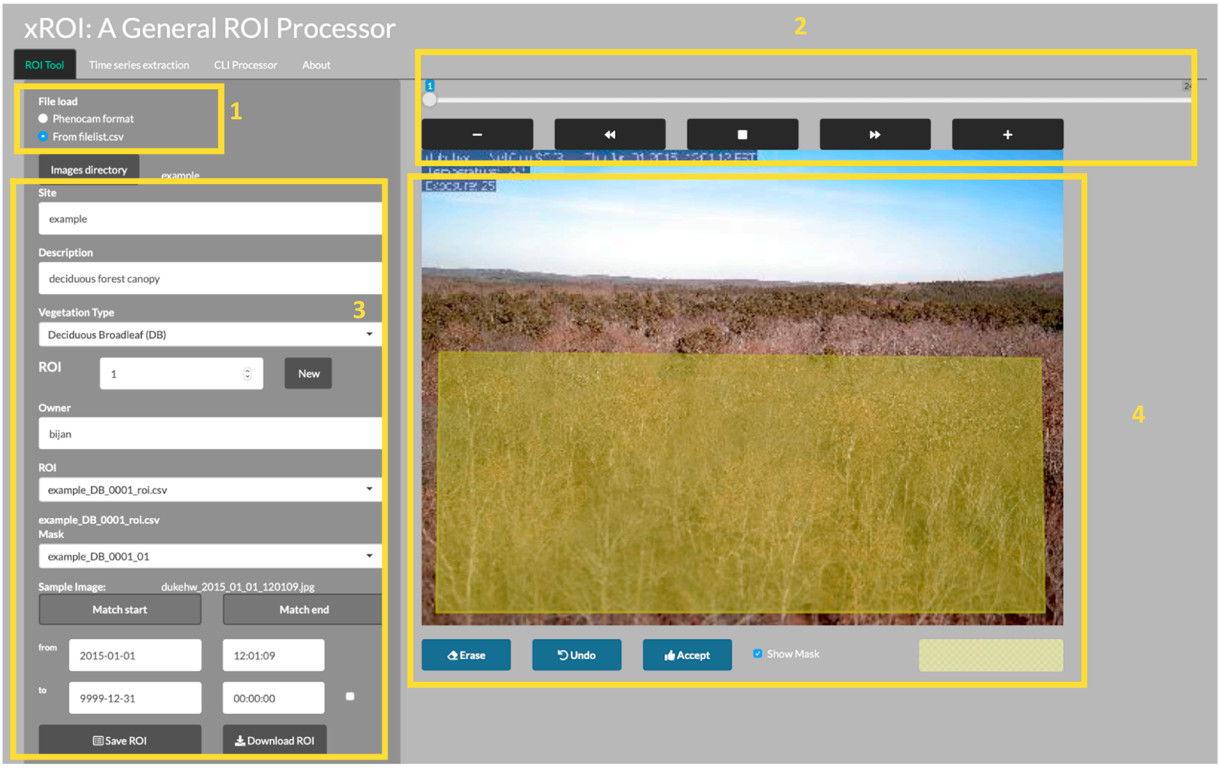

Fig. 2. Arrangement of the user-interface items in the ROI drawing module in xROI. (1) loading images, (2) exploring tool to browse images, (3) entering ROI metadata, and (4) the ROI drawing panel.

cran.r-project.org/package = xROI/xROI.pdf. The user manual includes a set of examples for each function. Here we explain the graphical user interface, which can be launched from an interactive R environment by:

```
library(xROI)
Launch()
```

or from the command line (e.g. *shell* in Linux, *Terminal* in macOS and *Command Prompt* in Windows machines) where an R engine is already installed by:

```
Rscript -e 'xROI::Launch(Interactive = TRUE)'
```

Calling the *Launch* function opens up the app in the system's default web browser, featuring an example dataset to explore different modules or upload a new dataset of images.

3. Design and structure

xROI includes three main modules: (a) ROI drawing module, (b) FOV shift monitoring module; and (c) time series extraction module. Fig. 2 shows the arrangement of each module as separate tab-panels. All the modules have a server-side and a UI-side which are explained in the following sections. The modules and how they are linked to each other are illustrated in Appendix A: Application Flowchart.

3.1. ROI drawing module

The main function of the ROI drawing module is to provide an interactive environment for creating regions of interest (ROI's) and storing associated files on a disk space for a later use. The user can load a set of images using the ***Image directory*** button and browse into the folder containing the data (item 1 in Fig. 2). There are two ways to load images: (1) using the "PhenoCam format", and (2) using the ***filelist.csv*** input file. If the user selects "PhenoCam format", all JPEG images in the image directory that follow the PhenoCam naming convection (Richardson et al., 2018a) will be loaded into the app. Time and date metadata will be automatically assigned to each image based on their filenames (i.e. < sitename > _ < YYYY_MM_DD > _ < hhmmss > .jpg, where MM = 01-12, DD = 01-31, hh = 00-23, mm = 00-59 and ss = 00-59). If the user selects "From ***filelist.csv***", the software looks for a comma separated file named "***filelist.csv***" to obtain information about how to properly load the dataset. In that case, the user is responsible for generating the ***filelist.csv*** file. The ***filelist.csv*** file contains a list of images and their associated timing and is formatted in the comma-separated-values format as follows. Each row includes one column for the filename as character strings and six columns for year, month, day, hour, minute and second of the acquisition date and time, in that order. Two example rows are presented as follows:

"dukehw_2015_01_01_120109.jpg",2015,1,1,12,1,9
or
"IMG2012.jpg",2019,2,5,7,00,00.

The user can explore loaded images using the exploring panel (item 2 in Fig. 2).

After images are loaded, the first step of generating a new ROI is to enter metadata for the ROI, including site name, ROI description, vegetation type (see Table 1), ROI ID number (a user defined number to identify ROI for vegetation type); and start and end date and time of the mask files (item 3 in Fig. 2). The user can create new ROI masks by drawing polygons around the region of interest on the plotted image. This is performed using *Shiny's clickOpts* functionality. The image that is used to draw an ROI is called the "*sample image*". The sample image may be used for later references. The user can add new vertices using single clicks on the image, drawing edges of a polygon. Double clicks are reserved to close a polygon and start a new one. The user can edit an existing polygon using the ***Undo*** and ***Erase*** button (item 4 in Fig. 2).

Table 1
File formats and naming convention of ROI files.

File	Format	Naming convention[*]	Purpose
ROI List	Text/CSV	< sitename > _ < veg_type > _ < ROI_ID_number > _roi.csv	ROI metadata
Mask	TIFF	< sitename > _ < veg_type > _ < ROI_ID_number > _ < mask_ID_number > .csv	Raster-based mask file
Polygon	CSV	< sitename > _ < veg_type > _ < ROI_ID_number > _ < mask_ID_number > _vector.csv	Vector-based region of interest

* < sitename > is a character string including the site name entered by the user. < veg_type > is a two-character value indicating the vegetation type as selected by the user (AG: Agriculture, DB: Deciduous Broadleaf, EB: Evergreen Broadleaf, EN: Evergreen Needleleaf, DN: Deciduous Needleleaf, GR: Grassland, MX: Mixed Forest, NV: Non-vegetated, RF: Reference Panel, SH: Shrub, TN: Tundra, UN: Understory, WL: Wetland, XX: Other/Canopy). < ROI_ID_number > is a four-digit integer number as a unique identification number of the ROI for the corresponding vegetation type. < mask_ID_number > is a two-digit integer number identifying the mask files.

The vector-based polygons are stored as the relative coordinates of vertices. The polygons can be converted to a rasterized mask using the ***Accept*** button. This step is performed using the ***GDAL***, ***rgdal*** and ***raster*** libraries.

To store the generated ROI to disk, we adopted the file structure and formats of the *PhenoCam* network (Richardson et al., 2018a). Each ROI definition consists of an "ROI List" file, one or more mask file(s), and their corresponding vector-based polygon files. An *ROI List* file is a *CSV* file that contains ROI metadata including owner's name, primary vegetation type, description, list of mask files, their associated start and end date and times and the sample image filename. Details about formatting of the ROI List file are discussed in Richardson et al. (2018a) and also presented here in Appendix B: Description of ROI List Files. Vector-based polygon files indicate the relative coordinates of points defining the region of interest. In fact, polygon files are the raw input data created by the user in order to generate mask files. Mask files are raster images containing binary bitmaps of each mask in *TIFF* format: black (1) for "included" and white (0) for "excluded" pixels. The vector-based file, which is size-free, is used to generate a mask file that is the same dimension as the sample image. Although only mask rasters (in TIFF format) and ROI list files are directly used to extract the time series, vector-based information (i.e. coordinates of vertices) are stored for reference; potential uses include generating new mask rasters for files with different image dimensions. The user can save all ROI-related files in the original directory for later reference using the ***Save ROI*** button. The user can also download them as a zipped file using the ***Download ROI*** button. And, because the ROI definitions follow the standard *PhenoCam* format, the downloaded ROI files can be proposed to the *PhenoCam* data management team for incorporation into the routine processing, if image data have already been contributed to the *PhenoCam* network. The file formats and the naming convention of the ROI files are presented in Table 1.

3.2. FOV shifts monitoring module

We used a simple, fast and efficient method to detect potential FOV shifts using the center-line image (*CLI*) technique (Seyednasrollah, 2017), and to enable the user, with minimal effort, to validate shift detections. A *CLI* is a single image raster, representing the entire dataset as its vertical columns are composed of the center column of each loaded image. The assembled *CLI* provides a quick, simple and robust way to visualize significant changes in the horizon line or canopy texture. This enables the user to rapidly inspect potential FOV shifts, and adjust the ROI accordingly. The user can move the mouse pointer over the *CLI* to find the date on which an FOV shift has occurred. Clicking on the *CLI* will display the image from the corresponding date on the lower left side of the panel. Fig. 3 shows the built-in *CLI* processor of *xROI* that is used as a shift monitoring module. The user can generate the *CLI* from original images, write the generated *CLI* on the disk space or read a previously saved *CLI* from the disk space. Days without images are shown as black columns (hex code: #000000) in the *CLI* raster. Besides the true color RGB raster of *CLI*, *xROI* also provides monochromic images of individual color channels (red, green and blue) and also brightness, darkness and contrast rasters. Multiple options for visualizing the *CLI* is to assist the user in detecting FOV shifts with choosing an appropriate raster. We performed a quantitative analysis to evaluate the performance of each monochromic images in separating out sky and canopy pixels. We used the bimodality coefficient defined in Zhang et al. (2003) as a proxy for detectability power, where higher bimodality coefficients correspond to better separations of pixels. The results showed that the brightness image and the blue channel were most effective in separating out the pixels, confirming our visual interpretation. The analysis is presented in Appendix D: Bimodality Analysis of the Monochromic CLI.

While the CLI technique can be used to detect most FOV shifts including horizontal, and vertical shifts, the method may fall short in identifying FOV shifts that lacks a strong signal in the CLI image. To address this limitation, we have implemented an additional function ("*detectShifts*") that can be used for detecting other FOV shifts. The function evaluates day-to-day correlation values of the brightness and blue color channels when they are smoothed. It returns a two-column data frame for daily variability of the brightness and the blue channel. A sudden decrease in the correlation values may be interpreted as a potential FOV shift. Note fully automated methods are difficult with outdoor photography due obscured FOV due to rain, clouds, or fog, resulting in false positives when no actual FOV shift has occurred. As the "*detectShifts*" function is computationally expensive, it is only available from the command line and not from the UI.

3.3. Time-series extraction module

The time series extraction module is designed to extract color-based statistics for a selected ROI on the entire image dataset at a specified time interval (see Fig. 4). The time series data can help the user to make decision on selecting appropriate polygons that lead to clearer signals in the final time series. While in the example datasets, we used mid-day images at an interval of 1 day, though in practice data with higher temporal resolution can be obtained using all (sub-daily) images, resulting in higher-quality time-series (Sonnentag et al., 2012).

To perform statistical calculations, each JPEG image is read as three-dimensional array: $[I]_{H \times W \times C}$, where H is the number of vertical pixels (height), W is the number of horizontal pixels (width) and C is the color channel (1: red, 2: green, 3: blue). The third dimension is to store three different color channels (red, green and blue, respectively). Mask rasters, $[M]_{H \times W}$, (in TIFF format) have the same resolution as the sample image, but unlike the sample image, they are in binary (0 and 1) format. M is a binary matrix, 1 for pixels within the ROI and 0 for elsewhere. For a given mask file, the red, green and blue chromatic coordinates (i.e. R_{CC}, G_{CC} and B_{CC}) are defined as (Sonnentag et al., 2012):

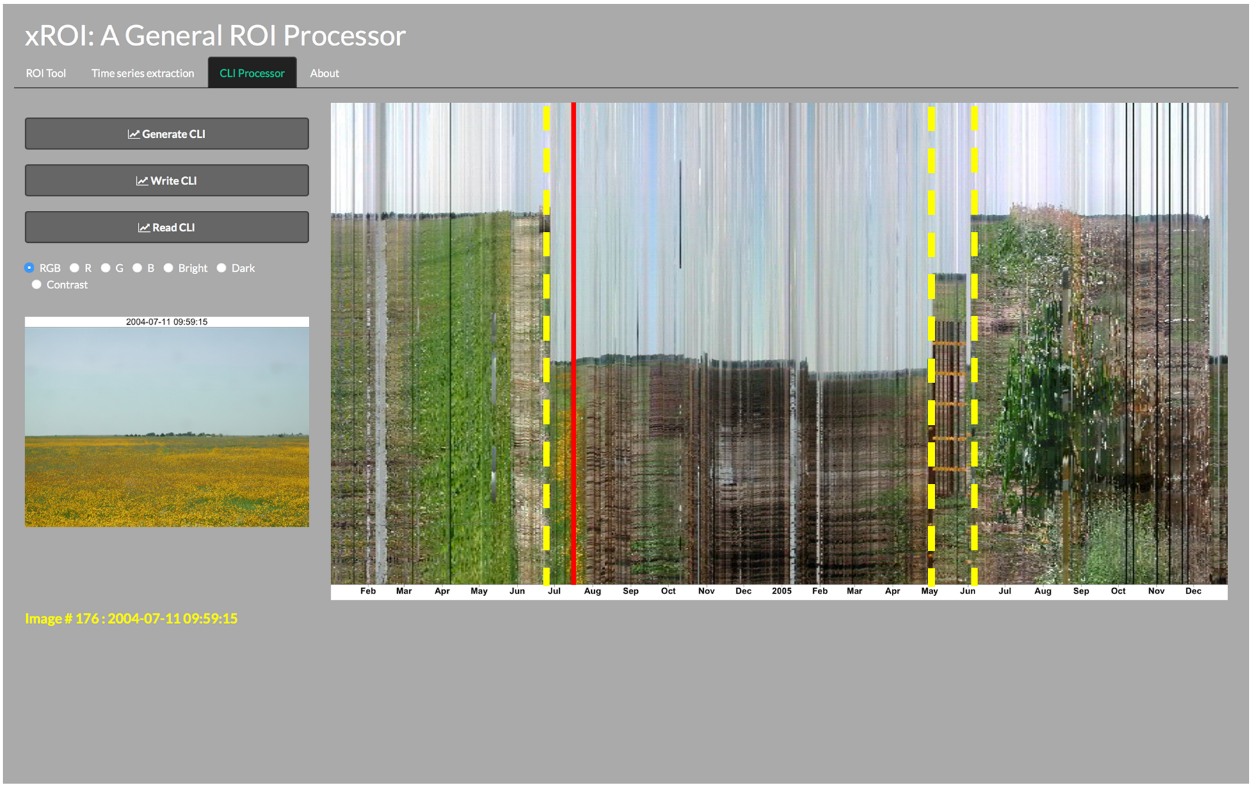

Fig. 3. Detecting field of view shifts using center line images (CLI). The CLI is built by assembling vertical centerlines of all loaded images together. The user can visually detect FOV shifts by monitoring sudden changes in the horizon line and / or canopy texture. The user can plot the corresponding image (small image in the bottom left corner) of each column in the CLI by clicking on the image. The vertical red line indicates the position of the selected image on the CLI. The above example was generated from the PhenoCam site at **armoklahoma.** The vertical yellow dashed lines indicate FOV shifts.

$$R_{CC} = \frac{R_{DN}}{R_{DN} + G_{DN} + B_{DN}}$$

$$G_{CC} = \frac{G_{DN}}{R_{DN} + G_{DN} + B_{DN}}$$

$$B_{CC} = \frac{B_{DN}}{R_{DN} + G_{DN} + B_{DN}}$$

where R_{DN}, G_{DN} and B_{DN} are average red, green and blue digital numbers within the masked areas, respectively. Statistical metrics such as mean, median, 5, 10, 25, 75, 90 and 95 percentiles are derived for each chromatic coordinate across the entire ROI. The chromatic coordinates calculation essentially normalizes each individual color band against the total pixel value of the three channels, normalizing for the total brightness. G_{CC}, in particular is shown to be a reliable metric for monitoring changes in the environment such as leaf out phenology (Klosterman et al., 2018), vegetation identification (Woebbecke et al., 1995), plant health status (Nijland et al., 2014), and biological conservation and restoration (Alberton et al., 2017). In addition to chromatic coordinates, brightness, darkness and contrast rasters are also calculated for the region of interest. Brightness is the maximum value among red, green and blue channel for each pixel. Darkness is the minimum value among red, green and blue channel for each pixel. Contrast is the difference between brightness and darkness. The darkness (D), brightness (B) and contrast (C) rasters (Mao et al., 2014) are calculated as:

$$D[i, j] = \min_{c \in (1,2,3)} I[i, j, c], \quad i \in (1, 2, .., H), j \in (1, 2, .., W)$$

$$B[i, j] = \max_{c \in (1,2,3)} I[i, j, c], \quad i \in (1, 2, .., H), j \in (1, 2, .., W)$$

$$C = B - D$$

The user can change the computation interval at which the time series is extracted. To extract sub-daily time series, the user can simply import a data set consisting of sub-daily images and change the interval value to extract time-series with different temporal resolution. Note that timeseries generated with different interval values are not based on the same concept as the PhenoCam 1-day and 3-day timeseries. Higher interval values in xROI simply mean skipping the images in between each interval, whereas PhenoCam 1-day and 3-day products, explained in Richardson et al. (2018a), are extracted based on statistical metrics within the interval. Higher intervals may only be used only to speed up processing time, but it is ideal to analyze every image in a stack for the highest-quality data. The time series will be plotted as an interactive *Plotly* object (Sievert et al., 2017) with capabilities of zoom, selection and scrolling. The user can hide or show any of the chromatic coordinates and their confidence intervals (i.e. deviation within the ROI) using the checkbox inputs on the side panel. The extracted time series can also be downloaded as a CSV file containing the filenames, time information and the data. In addition to the chromatic coordinates, band ratios (Bradley et al., 2010; Tucker, 1979), excess greenness (Nijland et al., 2014) and the green-red vegetation index (GRVI) (Richardson et al., 2013) are also reported in the output file. Other user defined metrics can be obtained by postprocessing the data included in the output file. A summary of other indices that might be used for monitoring phenology is discussed in Richardson et al. (2013).

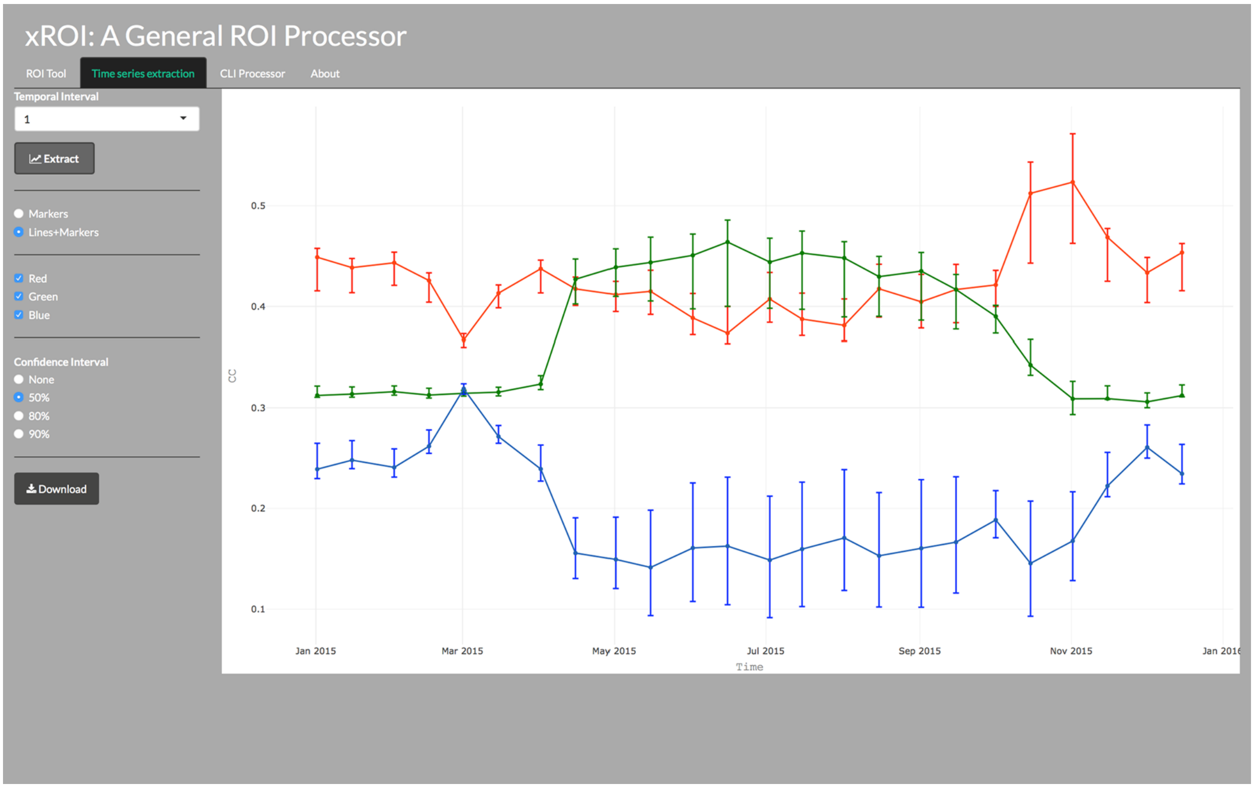

Fig. 4. Time series extraction module. The figure shows an example site and the associated R_{CC}, G_{CC}, and B_{CC} time series data extracted at a 15-day interval. The interactive plot facilitates exploratory analysis throughout the time series. The bars indicate 50, 80 or 90 percentiles of the confidence interval across the entire ROI, depending on the user's selection. The user can also download the time series in CSV format using the Download button. Image datasets with a finer temporal resolution will result in a finer time series.

Table 2
Site specific information and descriptions of regions of interest.

Site name	Lon. , Lat.	Year(s)	FOV Shift Date(s)	ROI description	Vegetation Type
boundarywaters	−91.49, 47.94	2008 2009	4/29/2008 5/11/2009 6/9/2009	Grasses in foreground	GR
pasayten	−119.89, 48.39	2015 2016	12/23/2015	Individual evergreen tree in lower left side of the FOV	EN
proctor	−72.86, 44.52	2016 2017	12/13/2016	Individual maple tree in near background	DB
sherman	−121.75, 38.03	2014 2015	8/18/2014 9/3/2014 9/7/2014	The upper cropland/grassland the foreground	AG

4. Case studies

In the following sections, we use several case studies to explain different features of *xROI*. We quantify how *xROI* performs for handling data management and extraction tasks. Among the sites in the PhenoCam network, the number of FOV shifts varies across sites. For example, the ***monture*** PhenoCam site experienced 62 FOV shifts over the course of 18 years and the ***acadia*** PhenoCam site experienced 9 FOV shifts during the same period. Other PhenoCam sites, such as ***harvard***, with an extremely stable FOV over the period of record (2008-ongoing), and a strong seasonal cycle due to the deciduous canopy, present comparatively smaller challenges. We used four *PhenoCam* sites, including ***boundarywaters***, ***pasayten***, ***proctor*** and ***sherman***, as our case study examples to illustrate how our software works, and how it can be used to extract high quality time series data. Site selection in this document is based on including sites that present various situations, complexities and processing challenges. The ***boundarywaters*** images were collected from a mixed deciduous and evergreen forest at Boundary Waters Canoe Area Wilderness, Superior National Forest, Minnesota, USA. The ***pasayten*** images were taken at a mountainous Ponderosa pine forest at Pasayten Wilderness, Okanogan National Forest, Washington, USA. The ***proctor*** images were taken at a maple-dominated deciduous forest at University of Vermont, Proctor Maple Research Center, Underhill, Vermont, USA. The ***sherman*** images were collected from a grassland at Twitchell Island, Antioch, California, USA. Additional site-specific information is presented in Table 2. We obtained stacks of digital images for the selected sites from the *PhenoCam* dataset available from Richardson et al. (2017). Each set contains daily midday (closest image to the local standard noon) images of the landscape over a two-year period. Note that we have also processed the

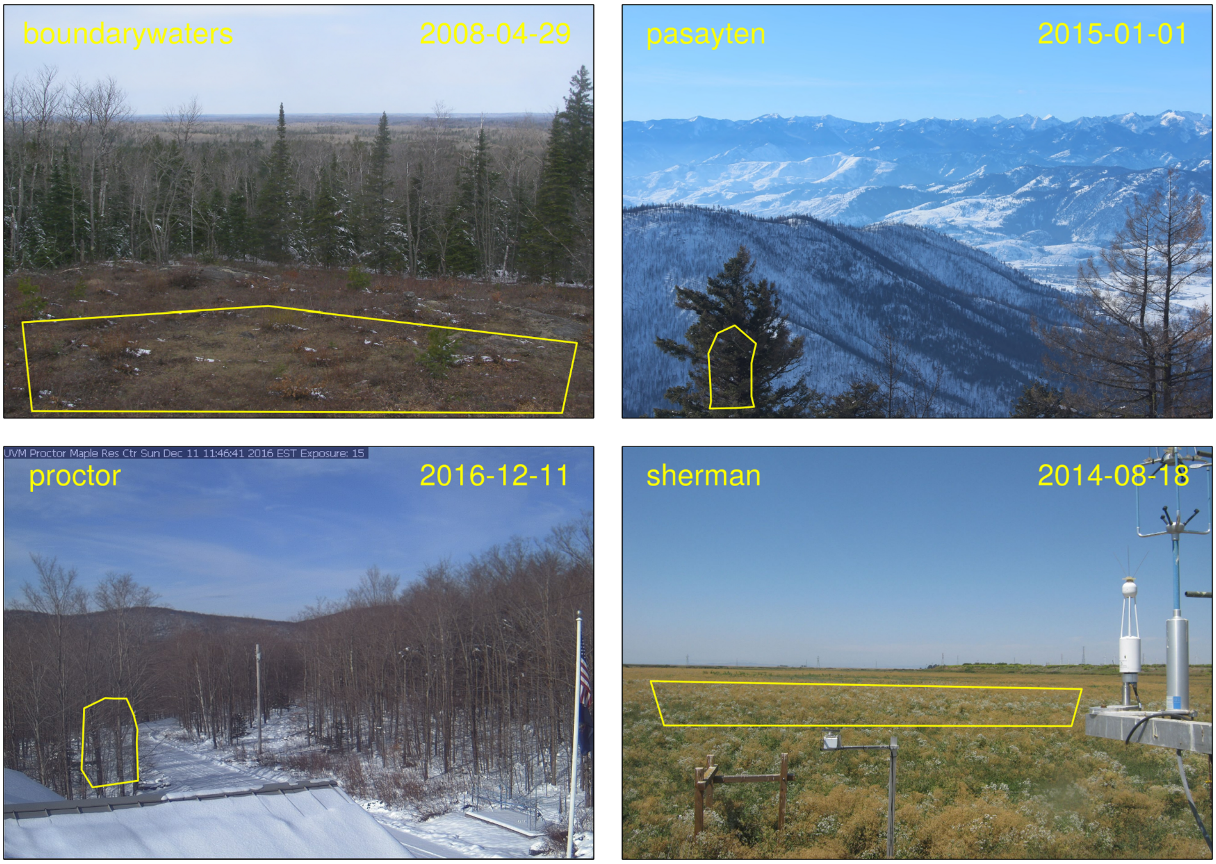

Fig. 5. Original ROI's drawn for case study sites: (a) boundarywaters, (b) pasayten, (c) proctor; and (d) sherman. As field of view has changed for the sites, regions of interest may need to be readjusted or redrawn, otherwise the final data may be incorrect, misleading or in low quality.

higher-frequency data (30-min, Seyednasrollah et al., 2019), but to detect FOV shifts the daily images are sufficient. All selected sites have at least one FOV shift occurrence over the course of the collected data. The case study datasets are available to download from Seyednasrollah (2019) for reproducing the results presented here.

4.1. Procedure and workflow

Using *xROI*, we performed two experiments on each image set. In the first experiment, we used the ROI drawing module to generate an ROI representing the dominant vegetation cover based on the first image in the dataset. ROI descriptions for all four sites are explained in Table 2. The originally drawn ROI's are shown in Fig. 5. We intentionally assumed there was no FOV shift in the dataset and generated the G_{CC} time series for two years of data at each site using the time series extraction module.

In the second experiment, we used the FOV shifts monitoring module and the *CLI* processor to visually detect all the FOV shifts per camera and then readjust the ROIs from the first experiment by adding new masks for each shift periods (Table 2). The resulting *CLI* for each camera is presented in Appendix C: Center-line Images of Study Sites.

The *CLI* for the case studies are shown in Fig. S2. Using the *CLI* processor helped us to detect FOV shifts in the image datasets by monitoring the horizon line for each image. This is particularly critical for extracting accurate and meaningful phenological signals from digital repeat photography. The time series obtained in the first and second experiments are plotted in Fig. 6. Dates when FOV shifts occurred are indicated with dashed vertical lines. It is seen that accounting for the FOV shifts is essential and greatly enhances the quality of the derived data, particularly for ***pasayten*** and ***proctor*** in the second year of each set of images. Another example of how unadjusted FOV shifts may reduce data quality is the ***monture*** PhenoCam site as it is discussed in Richardson et al. (2018a).

5. Discussion

We illustrated how different modules of the *xROI* application work together to enable an integrated environment for ROI based image processing. *xROI* enhances time series extraction from digital repeat photography datasets in three ways: (1) interactive generation of user-defined ROI on a sample image, (2) extracting time series data of red, green and blue chromatic coordinates and their corresponding statistics, and (3) providing a simple way to identify FOV shifts using a center line image-based approach.

xROI has integrated several processes in a single framework to produce time series of multiple bands from a stack of digital images. This is a significant contribution, and greatly reduces the workload associated with conducting each task using separate software tools. Importantly, the user can perform the entire image processing workflow without any knowledge of computer programming or image processing, using a simple and user-friendly graphical user interface. This is critical, considering the amount of technical analyses that is being performed in response to the user's clicks. Alternatively, more advanced users can use the low-level *xROI* functions library to carry out customized analyses

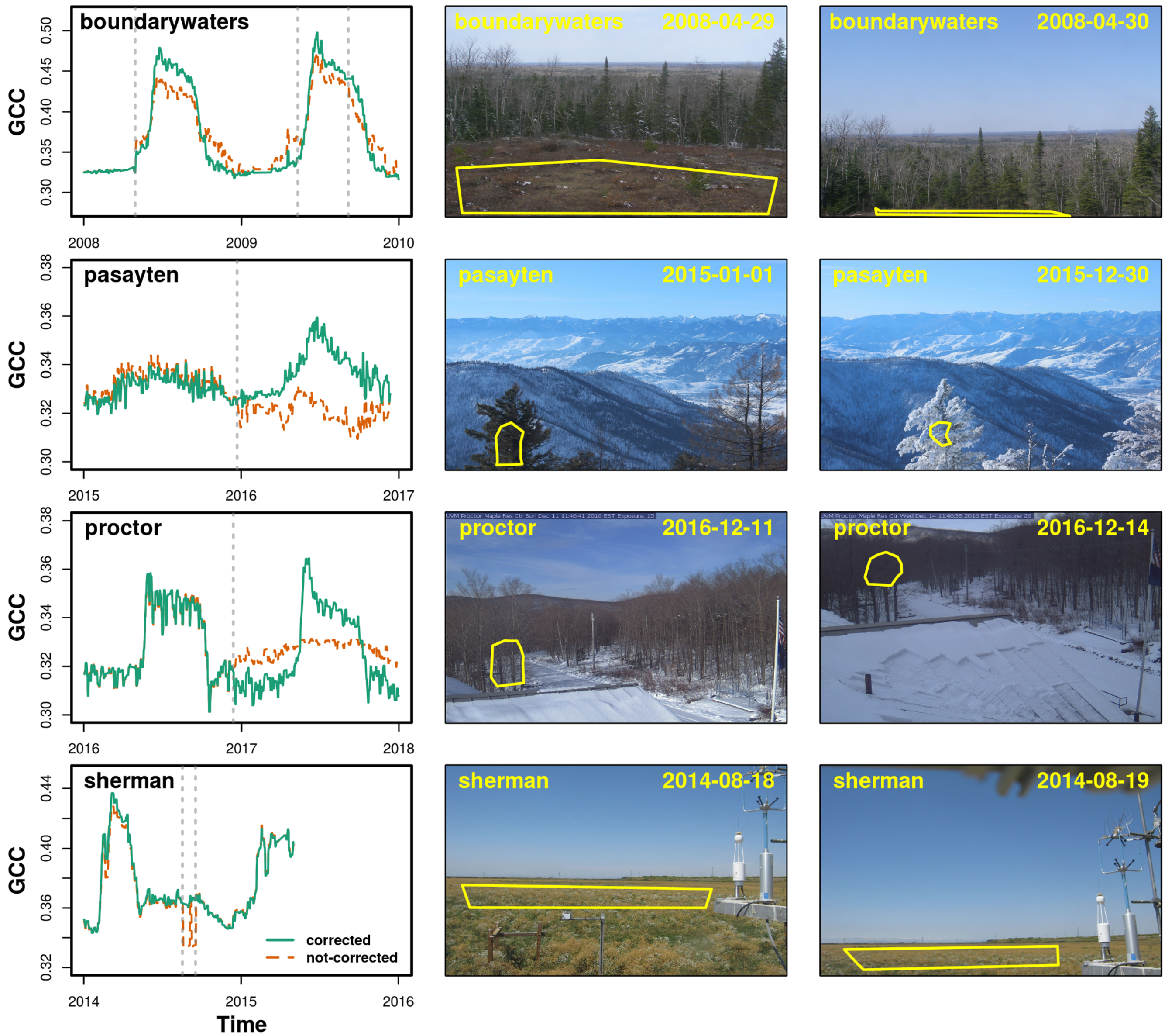

Fig. 6. Time series extraction using xROI for four case study sites. The left panel shows the extracted time series, for each dataset. Red (dotted) and green (solid) lines indicate pre and post mask adjustment using xROI, respectively. Dashed vertical lines indicate FOV shifts. The middle panel shows the cameras' FOV before a shift occures. The right panel shows the FOV's after the shift. ROI's are shown by yellow polygons.

using a command-line interface. We believe both approaches will contribute to advancing the potential for cutting-edge science applications of digital repeat photography.

The built-in interactive time series extraction module can be used not only for generating color-based statistics, but also it can be operated as a real-time evaluation toolkit for drawing the most appropriate ROI. The user can create several ROI's and run the time series extraction module for each one to assess which ROI results in a less noisy time series or more suitable confidence intervals. To mitigate the impacts of FOV shifts, the tool can also be used to test candidate ROI masks and identify which masks are most robust to changes in the camera FOV. This might be particularly useful when FOV shifts are small, and a single, well-chosen ROI might suffice. The "interval" input value can be used to speed-up this process by running the time series extraction module on a subset of the original image set (i.e. by using larger interval values). Additionally, the confidence interval bars show color variation for pixels across the ROI at each time step. Relatively lower ranges of variation indicate more homogeneous color across the ROI. This might be desired in particular for focusing on a specific object such as an individual tree in the image.

To handle FOV shifts – as one of the most important challenges in digital repeat photography – we showed how *xROI* exploits a simple method to visualize camera FOV stability using the *CLI* technique. The FOV shift detection module plays an essentially important role at sites where the camera is subject to intentional or accidental FOV shifts. For example, tower climbers might accidentally kick the camera housing or wind and vibration might loosen the camera mount. Using the case studies, we illustrated how the user can readjust ROI masks to correct for FOV shifts. We also tested detectability of FOV shifts using several monochromic rasters in the *CLI*. Both visual interpretations and the results from the bimodality coefficient (See Appendix D: Bimodality Analysis of the Monochromic CLI) showed brightness and blue channel rasters suggest the highest performance in distinguishing canopy from sky, and hence identifying the horizon line. This was true for all the four case studies. This may be explained by the greater average difference values of blue and brightness between canopy and sky. In other words, sky pixels are usually bright and have high values in the blue channel, while vegetation pixels are dark and low in the blue channel. Using the red and green channels showed the lowest power for detecting FOV shifts. Although this result may not be universal or consistent for a larger set of sites, it can further be evaluated and utilized in developing methods for automatic detection of FOV shifts. Our method using the vertical CLI is not perfect, and one can envision cases where there is only a horizontal shift in camera FOV, which might not be detected from the CLI. However, in our experience, the vast majority of FOV shifts that negatively impact data quality are clearly identified from this analysis, because it is extremely rare that the plane of an FOV shift is exactly parallel to the horizon. For this reason, even FOV shifts with a small vertical component can often be detected through inspection of the CLI.

Results from using *xROI* on the four provided example datasets as case studies showed the final time series were significantly improved by adjusting the ROI and mask files after each FOV shift. Unadjusted ROI result in erroneous time series and misleading —if not simply incorrect —phenological patterns. For instance, unadjusted ROI's at the case study sites resulted in significant errors in both the magnitude and the timing of greenness. At ***boundarywaters***, maximum G_{CC} falsely dropped from 0.48 to 0.44 in 2008 and 0.50 to 0.47 in 2009. The unadjusted ROI at the same site resulted in 55 days (from 225 to 170) and 97 days (from 167 to 264) bias in the length of the growing season in 2008 and 2009, respectively. At ***pasayten*** and ***proctor***, G_{CC}'s from the unadjusted ROI did not exhibit any greenup during the growing season of the second year (2016 at ***pasayten*** and 2017 at ***proctor***). At ***sherman***, the unadjusted ROI resulted in a false G_{CC} drop (from 0.36 to 0.33) for about 33 days in 2014, which does not correspond to any phenologically or ecologically relevant change in the state of the canopy. Previous works have studied ecological drivers that explain the interannual variabilities of G_{CC} (Richardson et al., 2019, 2018c) and the strong agreement between transition dates from G_{CC} and those observed on the ground (Richardson et al., 2018a). An overview of methods and scientific questions and applications is discussed in Richardson (2018).

Although in this paper, we primarily focused on extracting phenological time series from datasets of vegetation phenology, *xROI* can be used to extract different kinds of time series from individual color channels or their combinations. For example, the application can also be utilized for continuous measurements of accumulated snow depth by delineating an ROI around a measuring stick with a contrasting color to snow (e.g., black) in the FOV and converting the chromatic coordinates to the proportion of black and white in the ROI (Farinotti et al., 2010). Similarly, time series of chromatic coordinates can be used for detecting water in tidal salt marshes (O'Connell and Alber, 2016), assessing water saturation status in soil (Silasari et al., 2017), and understanding the geomorphology of sand dunes (Banaszak and Selesko, 2016). As xROI is open-source, we hope that the scientific community can develop other tools built on the present framework to address a wide range of applications.

6. Conclusion

The *xROI* application was introduced for extracting color-based time series for datasets of digital repeat photography. *xROI* is an R package with both low-level image processing toolbox and a responsive graphical user interface. As an open-source software, *xROI* can be run on several platforms including macOS, Microsoft Windows, and many Linux distributions. The ready-to-use binaries of *xROI* are available from the Comprehensive R Archive Network (CRAN) archive. The most up-to-date source code can be accessed from the GitHub repository. We hope that the *xROI* R package will significantly enhance data quality and facilitates data extraction and management tasks, primarily for scientists who use near-surface remote sensing imagery for studying the environment, including—but not limited to—*PhenoCam* network imagery.

Acknowledgements

Support for the development and maintenance of PhenoCam network infrastructure has come from the National Science Foundation, through the Macrosystems Biology program, awards EF-1065029 and EF-1702697. The tools described here were developed to facilitate processing of image data from the National Ecological Observatory Network, under agreement 18-0080 between Battelle Memorial Institute and the Arizona Board of Regents. A.D.R. acknowledges support from NASA through the AIST program (award 80NSSC17K0582). The armoklahoma images were provided courtesy of the Atmospheric Radiation Measurement (ARM) Climate Research Facility, Billings, Oklahoma. The bartlettir images from the Bartlett Experimental Forest tower were supported by the National Science Foundation (grant DEB-1114804) and the USDA Forest Service's Northern Research Station. The boundarywaters images from Superior National Forest were provided courtesy of the USDA Forest Service Air Resources Management Program. The harvardhemlock images from Harvard Forest were partially supported through the National Science Foundation's LTER program (DEB-1237491), and Department of Energy Office of Science (BER). The pasayten images from the Okanogan National Forest were provided courtesy of the USDA Forest Service Air Resources Management Program. The images from proctor site were supported by the Agricultural Experiment Station of the University of Vermont. The sherman images were provided courtesy of the Berkeley Biometeorology Lab at University of California Berkeley. Imagery from the PhenoCam network is made publicly available under a fair use agreement (https://phenocam.sr.unh.edu/webcam/fairuse_statement/). We thank our many site collaborators for their efforts in support of PhenoCam.

Declarations of interest

The authors declared that there is no conflict of interest.

Appendix A. Application flowchart

See Fig. S1.

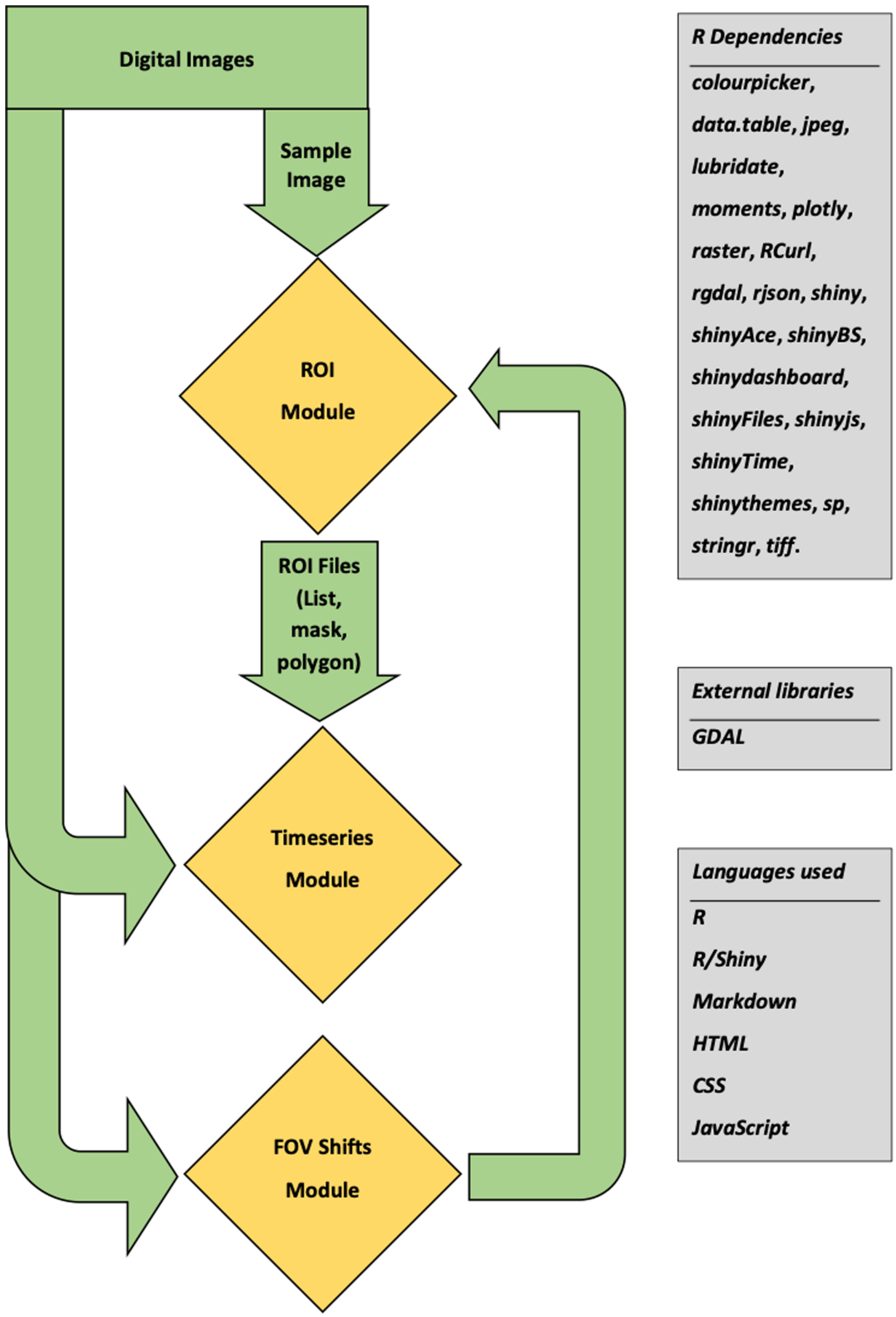

Fig. S1. Application flowchart.

Appendix B. Description of ROI list files

This sections explains the formatting of (1) the "ROI list files", which detail the date and time range over which each binary image mask was applied in processing the image data for a site; (2) the binary "image mask files", which delineate the ROI over which the image analysis was conducted; and (3) sample images for each image mask file. With (1) and (2), which we consider as essential metadata, the time series data sets can be reproduced from the original image files. A more comprehensive data description is explained in (Richardson et al., 2018a).

The naming convention for the ROI list files in is as follows:

- < **sitename** > _ < **veg_type** > _ < **ROI_ID_number** > _roi.csv

where **sitename** is the name of the camera site, as listed in the metadata contained in Data Record 1 (e.g., "coweeta"), **veg_type** is a two-letter abbreviation identifying the dominant vegetation within the ROI, e.g. DB for deciduous broadleaf trees (see Table 1), and **ROI_ID_number** is a numeric code that serves as a unique identifier to distinguish between multiple ROIs of the same vegetation type at a given site (0001 for the first ROI list, 0002 for the second, etc.).

A sample ROI list file (coweeta_DB_0001_roi.csv) is as follows:

```
#
# ROI List for coweeta
#
# Site: coweeta
# Veg Type: DB
# ROI ID Number: 0001
# Owner: mtoomey
# Creation Date: 2012–09-05
# Creation Time: 11:42:00
# Update Date: 2012–09-05
# Update Time: 11:42:00
# Description: full canopy including deciduous and subdominant conifers
#
start_date,start_time,end_date,end_time,maskfile,sample_image
2011–04-14,00:00:00,2012–11-08,14:31:57,coweeta_DB_0001_01.tif,coweeta_2011_04_08_143030.jpg
2012–11-08,15:01:00,9999–12-31,23:59:59,coweeta_DB_0001_02.tif,coweeta_2012_11_09_113132.jpg
```

The first 13 lines (beginning with #), document the provenance of the ROI list, and contain a brief description of the vegetation that is delineated by the associated image masks.

Line 14 lists the column headers for the mask entry rows. The mask entries begin on line 15. For this site there was one minor change in the field of view, so there are two ROI mask entries. Any additional field of view changes would result in additional rows (mask entries) being appended to the file. Note that as described in Methods, if the field of view shift is too large or if there are other exogenous events that necessitate distinguishing between the resulting data sets, a new ROI list (e.g., coweeta_DB_0002_roi.csv) would be created for the site.

For each mask entry, the data fields are:

- **start_date** (format: YYYY-MM-DD, where MM = 01–12 and DD = 01–31)
- **start_time** (format: hh:mm:ss, where hh = 00–23, mm = 00–59, ss = 00–59)
- **end_date** (format: same as for start_date)
- **end_time** (format: same as for start_time)
- **mask_file**: the filename for the 8-bit TIFF mask file with black for the ROI and white for the region to exclude from calculations
- **sample_image**: the filename for a sample image in the date range

Note that only images within the date and time ranges (from **start_date** and **start_time** to **end_date** and **end_time**) listed are included in the processed data set generated from this list. For **end_date**, the date code 9999–12-31 is used to keep the processing open-ended.

The naming convention for the image mask files is:

- < **sitename** > _ < **veg_type** > _ < **ROI_ID_number** > _ < **mask_index** > .tif

Here, the **mask_index** matches the entry number in the list (01 for the first entry, 02 for the second entry, etc.). The image mask files are stored in the TIFF image format (.tif) because of the flexibility that this offers, and because of compatibility with the python PIL library.

Sample images for each mask file have the same naming convention but terminate in a .jpg extension:

- < **sitename** > _ < **veg_type** > _ < **ROI_ID_number** > _ < **mask_index** > .jpg

Appendix C. Center-line images of study sites

See Fig. S2.

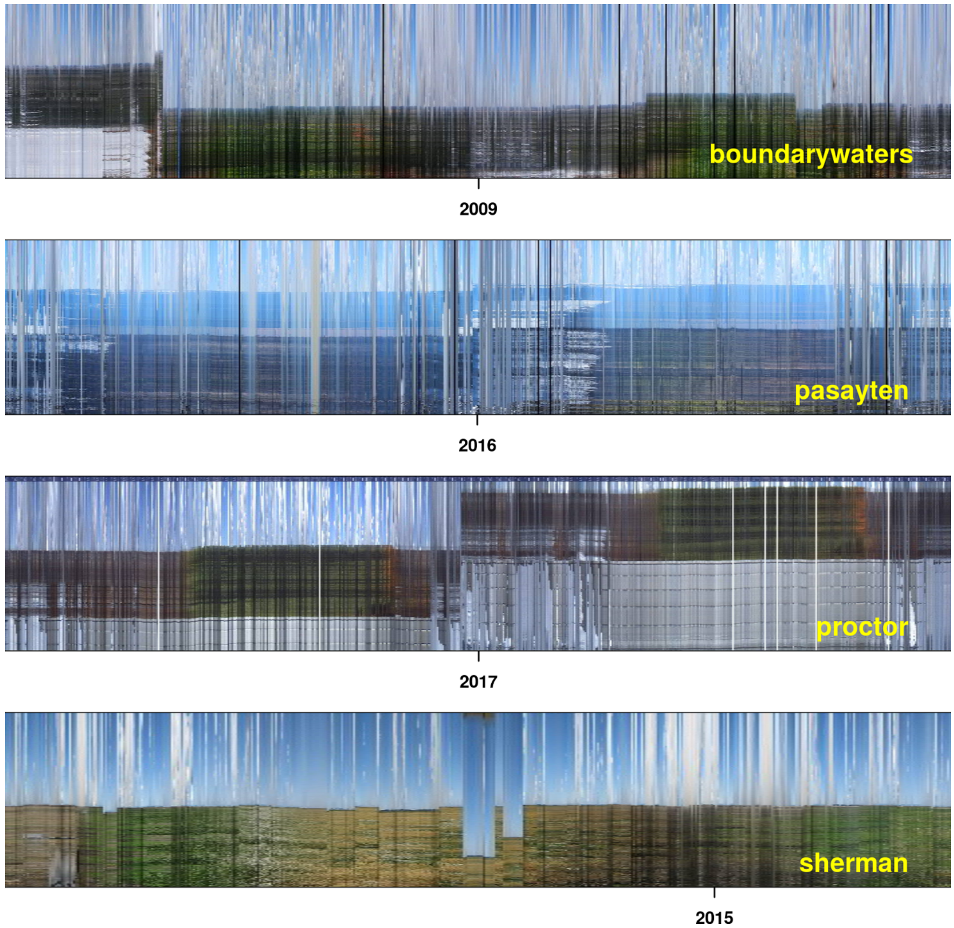

Fig. S2. Center-line images (CLI) are used to detect field of view shifts. Each panel shows the CLI image for case study sites: **boundarywaters**, **pasayten**, **proctor**; and **sherman**. Note that at **sherman** the FOV shifts are generally minor, except for the obvious shifts mid-timeseries.

Appendix D. Bimodality analysis of the monochromic CLI

As we have explained above, the *CLI* raster in the FOV shift monitoring module can be plotted in true RGB color format, as well as in monochromic color channels (R, G, and B), and brightness, and darkness rasters. The individual bands may display different levels of efficiency for separating canopy and sky pixels and so the horizon line. This can be highlighted by comparing the bimodality coefficients of each image. To quantify which binary band performs more reliably for distinguishing between canopy and sky, we ran a bimodality analysis on our datasets. For each *CLI* image we calculated the bimodality coefficient (Zhang et al., 2003) of individual color channels and brightness and darkness rasters (Fig. S3). Higher values of the bimodality coefficient indicate greater difference between the frequency of sky and canopy pixels and therefore bands with high bimodality coefficient are less challenging to visualize FOV shifts. This simple experiment can help us to select the most appropriate bands for which the *CLI* is plotted. We also visually observed visibility of the horizon line in different monochromic rasters. The results suggested that using the brightness and blue channels can better separate two groups of pixels including sky and canopy pixels than the other bands.

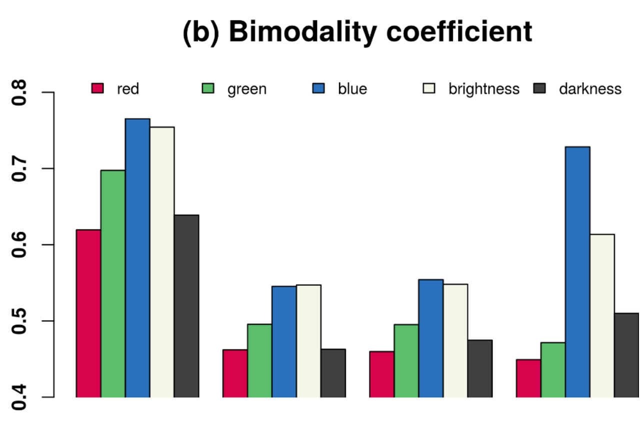

Fig. S3. Bimodality coefficients for monochromic rasters at different sites. Similar to our visual interpretations, blue channel and brightness images showed the highest performance to detect FOV shits. Darkness, red and green channels showed the lowest detectability power among all tested rasters.

References

Alberton, B., Torres, R.D.S., Cancian, L.F., Borges, B.D., Almeida, J., Mariano, G.C., dos Santos, J., Morellato, L.P.C., 2017. Introducing digital cameras to monitor plant phenology in the tropics: applications for conservation. Perspect. Ecol. Conserv. 15, 82–90.

Aronson, L., 1995. HTML 3 Manual of Style. Ziff-Davis Publishing Co.

Banaszak, E., Selesko, M., 2016. Tracing the sand dunes: using a combination of panoramic photography and dune pins to track changes in Michigan's sand dunes over time.

Baumer, B., Cetinkaya-Rundel, M., Bray, A., Loi, L., Horton, N.J., 2014. R Markdown: Integrating a reproducible analysis tool into introductory statistics. arXiv preprint arXiv:1402.1894.

Berra, E.F., Gaulton, R., Barr, S., 2019. Assessing spring phenology of a temperate woodland: a multiscale comparison of ground, unmanned aerial vehicle and Landsat satellite observations. Remote Sens. Environ. 223, 229–242.

Bivand, R., Keitt, T., Rowlingson, B., 2018. rgdal: Bindings for the 'Geospatial'. Data Abstraction Library.

Bradley, E., Roberts, D., Still, C., 2010. Design of an image analysis website for phenological and meteorological monitoring. Environ. Modell. Softw. 25, 107–116.

Brown, T.B., Hultine, K.R., Steltzer, H., Denny, E.G., Denslow, M.W., Granados, J., Henderson, S., Moore, D., Nagai, S., SanClements, M., 2016. Using phenocams to monitor our changing Earth: toward a global phenocam network. Front. Ecol. Environ. 14, 84–93.

Chang, W., Cheng, J., Allaire, J.J., Xie, Y., McPherson, J., 2017. shiny: Web Application Framework for R.

Crimmins, M.A., Crimmins, T.M., 2008. Monitoring plant phenology using digital repeat photography. Environ. Manage. 41, 949–958.

de Moura, Y.M., Galvão, L.S., Hilker, T., Wu, J., Saleska, S., do Amaral, C.H., Nelson, B.W., Lopes, A.P., Wiedeman, K.K., Prohaska, N., 2017. Spectral analysis of amazon canopy phenology during the dry season using a tower hyperspectral camera and modis observations. ISPRS J. Photogramm. Remote Sens. 131, 52–64.

Dhodapkar, A.S., Smith, J.E., 2003. Comparing program phase detection techniques. Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture. IEEE Computer Society p. 217.

Farinotti, D., Magnusson, J., Huss, M., Bauder, A., 2010. Snow accumulation distribution inferred from time-lapse photography and simple modelling. Hydrol Process. 24, 2087–2097.

Filippa, G., Cremonese, E., Migliavacca, M., Galvagno, M., Forkel, M., Wingate, L., Tomelleri, E., Di Cella, U.M., Richardson, A.D., 2016a. Phenopix: AR package for image-based vegetation phenology. Agric. For. Meteorol. 220, 141–150.

Filippa, G., Cremonese, E., Migliavacca, M., Richardson, A., Galvagno, M., Forkel, M., 2016b. phenopix: Pixel Based Phenology.

Gottumukkal, R., Asari, V.K., 2004. An improved face recognition technique based on modular PCA approach. Pattern Recogn. Lett. 25, 429–436.

Hijmans, R.J., 2017. raster: Geographic Data Analysis and Modeling.

Klosterman, S., Melaas, E., Wang, J., Martinez, A., Frederick, S., O'Keefe, J., Orwig, D.A., Wang, Z., Sun, Q., Schaaf, C., 2018. Fine-scale perspectives on landscape phenology from unmanned aerial vehicle (UAV) photography. Agric. Forest Meteorol. 248, 397–407.

Knox, S.H., Dronova, I., Sturtevant, C., Oikawa, P.Y., Matthes, J.H., Verfaillie, J., Baldocchi, D., 2017. Using digital camera and Landsat imagery with eddy covariance data to model gross primary production in restored wetlands. Agric. Forest Meteorol. 237, 233–245.

Liu, J., Pattey, E., 2010. Retrieval of leaf area index from top-of-canopy digital photography over agricultural crops. Agric. Forest Meteorol. 150, 1485–1490.

Mao, J., Phommasak, U., Watanabe, S., Shioya, H., 2014. Detecting foggy images and estimating the haze degree factor. Journal of Computer Science & Systems Biology 7, 1.

MathWorks, I., 2015. Matlab Image Processing Toolbox (Version 2015).

Migliavacca, M., Galvagno, M., Cremonese, E., Rossini, M., Meroni, M., Sonnentag, O., Cogliati, S., Manca, G., Diotri, F., Busetto, L., 2011. Using digital repeat photography and eddy covariance data to model grassland phenology and photosynthetic CO_2 uptake. Agric. Forest Meteorol. 151, 1325–1337.

Mikkonen, T., Taivalsaari, A., 2007. Using JavaScript as a real programming language.

Moore, C.E., Brown, T., Keenan, T.F., Duursma, R.A., Van Dijk, A.I., Beringer, J., Culvenor, D., Evans, B., Huete, A., Hutley, L.B., 2016. Reviews and syntheses: Australian vegetation phenology: new insights from satellite remote sensing and digital repeat photography. Biogeosciences 13, 5085.

Nijland, W., De Jong, R., De Jong, S.M., Wulder, M.A., Bater, C.W., Coops, N.C., 2014. Monitoring plant condition and phenology using infrared sensitive consumer grade digital cameras. Agric. Forest Meteorol. 184, 98–106.

O'Connell, J.L., Alber, M., 2016. A smart classifier for extracting environmental data from digital image time-series: Applications for PhenoCam data in a tidal salt marsh. Environ Modell Softw 84, 134–139.

Olivera-Guerra, L., Mattar, C., Merlin, O., Durán-Alarcón, C., Santamaría-Artigas, A., Fuster, R., 2017. An operational method for the disaggregation of land surface temperature to estimate actual evapotranspiration in the arid region of Chile. ISPRS J. Photogramm. Remote Sens. 128, 170–181.

Powell, T.A., 2010. HTML CSS: The Complete Reference, (Complete Reference Series). McGraw-Hill Osborne, New York, NY.

Richardson, A., Hufkens, K., Milliman, T., Aubrecht, D., Chen, M., Gray, J., Johnston, M., Keenan, T., Klosterman, S., Kosmala, M., 2017. PhenoCam Dataset v1. 0: Vegetation Phenology from Digital Camera Imagery, 2000–2015. ORNL DAAC, Oak Ridge, Tennessee, USA.

Richardson, A.D., 2018. Tracking seasonal rhythms of plants in diverse ecosystems with digital camera imagery. New Phytol.

Richardson, A.D., Hufkens, K., Li, X., Ault, T.R., 2019. Testing the Hopkins law of bioclimatics with PhenoCam data. Appl. Plant Sci.

Richardson, A.D., Hufkens, K., Milliman, T., Aubrecht, D.M., Chen, M., Gray, J.M., Johnston, M.R., Keenan, T.F., Klosterman, S.T., Kosmala, M., 2018a. Tracking vegetation phenology across diverse North American biomes using PhenoCam imagery. Sci. Data 5.

Richardson, A.D., Hufkens, K., Milliman, T., Aubrecht, D.M., Furze, M.E., Seyednasrollah, B., Krassovski, M.B., Latimer, J.M., Nettles, W.R., Heiderman, R.R., Warren, J.M., Hanson, P.J., 2018b. Ecosystem warming extends vegetation activity but heightens vulnerability to cold temperatures. Nature 560, 368–371.

Richardson, A.D., Hufkens, K., Milliman, T., Frolking, S., 2018c. Intercomparison of phenological transition dates derived from the PhenoCam Dataset V1. 0 and MODIS satellite remote sensing. Sci. Rep.-UK 8, 5679.

Richardson, A.D., Klosterman, S., Toomey, M., 2013. Near-surface Sensor-derived Phenology, Phenology: An Integrative Environmental Science. Springer, pp. 413–430.

Sanner, M.F., 1999. Python: a programming language for software integration and development. J. Mol. Graph. Model. 17, 57–61.

Seyednasrollah, B., 2017. drawROI: An interactive toolkit to extract phenological time series data from digital repeat photography. Zenodo. https://doi.org/10.5281/zenodo.1066588.

Seyednasrollah, Bijan, Swenson, Jennifer J., Domec, Jean-Christophe, Clark, James S., 2018. Leaf phenology paradox: Why warming matters most where it is already warm. Rem. Sens. Environ. 209, 446–455.

Seyednasrollah, B., 2019. xROI Example Datasets [Data Set]. Zenodo. http://doi.org/10.5281/2560686.

Seyednasrollah, B., Young, A.M., Hufkens, K., Milliman, T., Friedl, M.A., Frolking, S., Richardson, A.D., 2019. Tracking vegetation phenology across diverse biomes using PhenoCam imagery. The PhenoCam Dataset v2.0. Sci. Data.

Sievert, C., Parmer, C., Hocking, T., Chamberlain, S., Ram, K., Corvellec, M., Despouy, P., 2017. plotly: Create Interactive Web Graphics via 'plotly.js'.

Silasari, R., Parajka, J., Ressl, C., Strauss, P., Blöschl, G., 2017. Potential of time-lapse photography for identifying saturation area dynamics on agricultural hillslopes. Hydrol Process.

Sonnentag, O., Hufkens, K., Teshera-Sterne, C., Young, A.M., Friedl, M., Braswell, B.H., Milliman, T., O'Keefe, J., Richardson, A.D., 2012. Digital repeat photography for phenological research in forest ecosystems. Agric. Forest Meteorol. 152, 159–177.

Stephens, H.G., Shoemaker, E.M., Powell, J.W., 1987. In the Footsteps of John Wesley Powell: An Album of Comparative Photographs of the Green and Colorado Rivers, 1871-1872 and 1968. Johnson Books.

Sunoj, S., Igathinathane, C., Saliendra, N., Hendrickson, J., Archer, D., 2018. Color calibration of digital images for agriculture and other applications. ISPRS J. Photogramm. Remote Sens. 146, 221–234.

Team, R.C., 2018. R: A Language and Environment for Statistical Computing.

Tucker, C.J., 1979. Red and photographic infrared linear combinations for monitoring vegetation. Remote Sens. Environ. 8, 127–150.

Turner, R.M., 2003. The Changing Mile Revisited: An Ecological Study of Vegetation Change with Time in the Lower Mile of an Arid and Semiarid Region. University of Arizona Press, Tucson.

Warmerdam, F., 2008. The Geospatial Data Abstraction Library, Open Source Approaches in Spatial Data Handling. Springer, pp. 87–104.

Watson, C.J., Restrepo-Coupe, N., Huete, A.R., 2019. Multi-scale phenology of temperate grasslands: improving monitoring and management with near-surface phenocams. Front. Environ. Sci. 7, 14.

Woebbecke, D.M., Meyer, G.E., Von Bargen, K., Mortensen, D., 1995. Color indices for weed identification under various soil, residue, and lighting conditions. Trans. ASAE 38, 259–269.

Yan, D., Scott, R., Moore, D., Biederman, J., Smith, W., 2019. Understanding the relationship between vegetation greenness and productivity across dryland ecosystems through the integration of PhenoCam, satellite, and eddy covariance data. Remote Sens. Environ. 223, 50–62.

Yuan, H., Wu, C., Lu, L., Wang, X., 2018. A new algorithm predicting the end of growth at five evergreen conifer forests based on nighttime temperature and the enhanced vegetation index. ISPRS J. Photogramm. Remote Sens. 144, 390–399.

Zhang, C., Mapes, B.E., Soden, B.J., 2003. Bimodality in tropical water vapour. Q. J. R. Meteor. Soc. 129, 2847–2866.

Zhou, L., He, H.-L., Sun, X.-M., Zhang, L., Yu, G.-R., Ren, X.-L., Wang, J.-Y., Zhao, F.-H., 2013. Modeling winter wheat phenology and carbon dioxide fluxes at the ecosystem scale based on digital photography and eddy covariance data. Ecol. Inf. 18, 69–78.