Automatically drawing vegetation maps using digital time-lapse cameras in alpine ecosystems

A Preprint

Ryotaro Okamoto *
Doctoral Program in Biology
University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8577 Japan.
okamoto.ryotaro.su@alumni.tsukuba.ac.jp

Hiroyuki Oguma
Biodiversity Division
National Institute for Environmental Studies
16-2 Onogawa, Tsukuba, Ibaraki 305-8506 JAPAN
oguma@nies.go.jp

Reiko Ide
Earth System Division
National Institute for Environmental Studies
16-2 Onogawa, Tsukuba, Ibaraki 305-8506 JAPAN
ide.reiko@nies.go.jp

June 6, 2022

Abstract

Enter the text of your abstract here.

Keywords alpine ecology · deep learning · ecosystem monitoring

1 Introduction

The effects of climate change on terrestrial ecosystems are particularly significant in alpine regions ([1]). Alpine vegetation depends on severe conditions such as low temperatures and long snow-covered periods. Thus alpine areas have rare and unique species adapted to the extreme environments. Several studies have reported that recent global climatic changes, e.g., increasing temperatures and reducing snow-covered periods, have accelerated the invasion of non-native species into alpine areas (see [2]). In Japan, dwarf bamboo (*Sasa kurilensis*) has invaded alpine snow meadows, probably driven by the extension of the snow-free period ([3]). Also, climate change has affected the growth and phenologies of native species. For example, the growth of dwarf pine (*Pinus pumila*), a dominant species in Japanese alpine regions, has been affected by climatic conditions such as temperature and snowmelt ([4]). Monitoring and predicting such changes are essential for effective conservation planning. Since the impact of climate change on alpine vegetation varies depending on species and the microhabitats ([5]), spatially high-resolution monitoring with a wide range is required.

Previous studies have mainly depended on field observations, yet it is hard to cover broad areas in alpine regions due to the poor accessibility and severe weather. Satellite, airborne, and Unmanned Aerial Vehicle (UAV) remote sensing methods seem to be alternatives. However, satellite imagery of alpine areas is rarely available due to cloud cover, and the spatial resolution is not enough to observe vegetation changes at the plant community scale. Airborne imagery can obtain high-resolution data, but its cost becomes a bottleneck for frequent monitoring. Although UAV methods have become a cost-effective tool for ecological monitoring ([6]), operating UAVs in alpine regions is challenging due to the strong wind and harsh topology.

On the other hand, researchers have also utilized automated digital time-lapse cameras mounted on the ground for monitoring green-leaf phenologies in forests ([7]), grasslands ([8]), and alpine meadows ([9]). Unlike satellite imagery, such cameras provide images free of clouds and atmospheric effects. Also, they

*https://github.com/0kam

can obtain high-resolution (i.e., sub-meter scale) and frequent (i.e., daily or hourly) images at a meager cost. These studies set some regions of interest (ROI) in the images and calculate the phenology index (e.g., excess greenness, [10]). Also, we can visually interpret the vegetation distribution and its changes from such time-lapse images. However, few studies have tackled this. This lack of studies seems to be because applying such repeat photography in monitoring vegetation distribution has two technical challenges.

First, unlike satellites' multispectral sensors, ordinal digital cameras can only obtain three bands (Red, Green, and Blue), making it harder to classify the vegetation. Second, since digital time-lapse cameras are mounted on the ground, transforming these images into geospatial data (e.g., orthoimage) is challenging. In other words, even if we find a vegetation change, we cannot quantitatively measure and analyze it as geographic data. It is essential to treat ground-based images as geographic data to utilize them in conservation planning.

This study proposes an automated method for drawing vegetation maps and locating and measuring vegetation changes with a digital time-lapse camera by solving these two challenges. We solved the first challenge by utilizing temporal changes in leaf colors in vegetation classification. We also developed an accurate method for transforming a ground-based photograph into geographic data to tackle the second challenge. Finally, we show an example application of our method by detecting and localizing vegetation changes with an 8-years digital time-lapse imagery of the Japanese alps. We aim to use cheap but powerful digital time-lapse cameras in alpine ecosystem conservation.

2 Materials and methods

2.1 Digital time-lapse camera imagery

We used repeat photography data owned by National Institute for Environmental Studies, Japan (NIES). All the images are publically available on NIES' webpage (`https://db.cger.nies.go.jp/gem/ja/mountain/station.html?id=2`). In 2010, NIES installed the digital time-lapse camera (EOS 5D MK2, Canon Inc., 21 M pixels) on a mountain lodge Murodo-sanso (about 2350 m a.s.l., above the forest limit), located at the foot of Mt. Tateyama (3015 m a.s.l.), in the Nothern Japanese Alps. The camera takes one photograph per hour, from 6 a.m. to 7 p.m. . The camera's field of view (FOV) includes Mt. Tateyama, which ranges from about 2350 m a.s.l. to 3015 m a.s.l. in elevation. The area has a complex mosaic-like vegetation structure because of its topography, including rocks, cliffs, curls, and moraines. From April to November, the camera has observed the snowmelt and seasonal phenology of evergreen, deciduous dwarf trees (e.g., *Pinus pumila*, *Sorbus* sp.), dwarf bamboos (e.g., *Sasa kurilensis*), and alpine herbaceous plants (e.g., *Geum pentapetalum*, *Nephrophyllidium crista-galli*). We used the images from late summer to late fall of 2012 and 2020 to observe the distribution changes of the dwarf pine (*Pinus pumila*): the dominant species in Japanese alps, and the invasive dwarf bamboo (*Sasa kurilensis*).

2.2 Preprocessing

2.2.1 Selecting images

First, we selected images that are suitable for vegetation classification. We choose four days with good weather from late summer to late fall each year (9/19, 10/1, 10/11, 10/23 in 2010. 9/19, 10/2, 10/11, 10/23 in 2020). We used the images of this season because we can classify vegetation with the patterns of autumn foliage coloration. Also, we chose four images from 11 a.m. to 2 p.m. each day to avoid the effect of temporal noise such as shadows. Finally, we got 16 images for each year.

2.2.2 Automatic image-to-image alignment

Since images are slightly misaligned with each other due to wind, we aligned them before processing. We implemented the program with Python3 programming language and OpenCV4 (`https://opencv.org/`) image processing library. First, we set one image of 2010 as the alignment target. Next, we automatically found matching keypoints between the target and other images using AKAZE local feature extractor ([11]) and K-nearest neighbor matcher. Then, we searched and applied the homography matrix that minimizes the distance between each pair of the matching points, using OpenCV's "findHomography" function.

2.3 Automatic vegetation classification

Next, we classified the pixel time series of repeat photography imagery into vegetation. Because deciduous plants have great differences in autumn phenology among species, researchers have used that information for vegetation classification with satellite imagery (e.g., [12], [13], [14]). In our target area, we can recognize the vegetation type with the pixel time series of autumn (Fig. 2). However, no research has applied this technique to ground-based repeat photography imagery. We developed a deep-learning method to take advantage of high-resolution and frequent repeat photography data. Fig.3 shows the classification process.

2.3.1 Model Architecture

Since ground-based photographs can provide high spatial resolution (sub-meter scale in our dataset), we can use leaf texture as additional information for classification. For example, we can see a glossy surface on dwarf bamboos and a mat surface on dwarf pines. Thus, we used a small patch (9x9 pixels square) rather than a single pixel as an input of the model. Therefore, a model input is a time series of image patches.

Our model has two components to deal with this data structure (see Fig. 3). First, we extracted each patch's features (e.g., texture) using Convolutional Neural Network (CNN) layers (Fig. 4a). CNN is a neural network specialized in recognizing spatial structures of data, such as images. The CNN part outputs a time series of extracted features. Second, we used Recurrent Neural Net (RNN) layers (Fig. 4b) to classify the temporal patterns of the features extracted by the CNN part. RNN is a neural network specialized in recognizing temporal or sequential data dynamics, such as text and speech. Amongst many variants of RNN, we used Long Short Time Memory (LSTM, [15]). Combining CNN and RNN, our model can classify the input image-patch time series considering the colors and textures of each patch and their temporal patterns. This CNN-RNN architecture has also been used for audio classification([16]) and scene text recognition([16]).

Using a Deep Learning method has another positive side effect; mini-batch training. Since we used high-resolution images (16 x 21M pix), the training dataset became so immense that we could not feed them to the model at once: that consumes the computer's RAM too much. However, using Deep Learning methods enables us to provide them in small portions (called mini-batches) to save RAM consumption.

2.3.2 Dataset preparation

We set 5 vegetation classes: dwarf pine, dwarf bamboo, other vegetation, no vegetation, and sky. Using a free image annotation software (Semantic Segmentation Editor, Hitachi, `https://github.com/Hitachi-Automotive-And-Industry-Lab/semantic-segmentation-editor`), an expert drew polygons on an image of 2015 for each class. Since vegetation might be different in 2012, 2015, and 2020, applying this teacher data to the image of 2012 and 2020 may cause misclassification. However, vegetation changes are expected to occur at the edge of the plant community. Thus we shrink each polygon with several pixels before applying it to the training.

2.3.3 Implementation and model training

We implemented the classifier with Python3 language and PyTorch deep neural network library (`https://pytorch.org/`). All source codes are publically available via GitHub (`https://github.com/0kam/xxxx`).

3 Automatic georectification

Then, we developed a novel method to convert ground-based landscape imagery into GIS-ready geographical data. This process is called georectification. Georectification of ground-based images has been a difficult task, and this causes the underuse of potentially rich information in ground-based imagery. In plain words, georectification means aligning images into Digital Surface Models (DSMs) so that every pixel of an image gets a geographical coordinate. We can consider a camera as a function that transforms 3D geographical coordinates (latitude, longitude, height) into 2D image coordinates (locations of each pixel). So estimating the parameters of this function (such as the camera location, pose, and the field of view), we can map an image onto a DSM. We recommend readers to [17] for a plain mathmatical explanation of this process. Usually, georectification has three steps:

1. Finding Ground Control Points (GCPs) in the image.

2. Estimating camera parameters such as camera poses and field of view using GCPs.

3. Mapping the image onto the DSM using camera parameters.

Recently, researchers have developed some georectification methods to use ground-based photographs in glaciology ([18]) and snow cover studies ([17]). Especially, [17] is worth mentioning for its semi-automatic method using mountain silhouettes as GCPs. However, this silhouette-based method has a drawback in the projection accuracy. It only uses limited areas (silhouettes) of images in the image-to-DSM alignment, and also it ignores lens distortion. Because our target site has a complex vegetation distribution and our camera has considerable lens distortion, we needed a more accurate method.

3.1 Local-feature-based matching of images and DSMs

To get matching points between images and DSMs on a broader area, we used an airborne image already georectified. Combining an airborne image and a DSM, we rendered simulated landscape images (Fig. x). Then we got matching points (GCPs) by applying AKAZE local feature matcher to a target image and this simulated image (Fig. x). GCPs have geographical coordinates (from the DSM) and image coordinates (from the image). Our procedure requires the camera's exact location and initial camera parameters to render the simulated image.

3.2 Modeling and estimating lens distortions

We modeled the lens distortion (1) based on [19] and OpenCV's implementation, where x'' and y'' are the coordinates of each pixels after distortion and x' and y' are the coordinates before distortion. Our model includes radial ($k1$ $k6$), tangental ($p1, p2$), thin prism ($s1$ $s4$) distortion, and unequal pixel aspect ratio ($a1, a2$). See Weng et al. 1992 for the details of lens distortion modeling. Thus, now we can project GCPs' geographical coordinates into image coordinates using lens distortion parameters and other camera parameters (the camera location, pose, and the field of view). We optimized these (except camera location) by minimizing the square projection error of the GCPs using the Covariance Matrix Adaptation Evolution Strategy (CMA-ES, [20]). We could not estimate the camera location because it makes the problem too complicated (e.g., a telephoto taken from a distance and a wide-angle taken from a close look similar).

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} x' \frac{1+k_1r^2+k_2r^4+k_3r^6}{1+k_4r^2+k_5r^4+k_6r^6} + 2p_1x'y' + p_2(r^2+2x'^2) + s_1r^2 + s_2r^4 \\ y' \frac{1+a_1+k_1r^2+k_2r^4+k_3r^6}{1+a_2+k_4r^2+k_5r^4+k_6r^6} + p_1(r^2+2y'^2) + 2p_2x'y' + s_3r^2 + s_4r^4 \end{bmatrix} \tag{1}$$

3.3 Implementation and data set

We implemented the algorithm with Python3 language and published it as an open-source package via GitHub (`https://github.com/0kam/alproj`). You can try it with your data. We used an airborne photograph taken in the November of 201X with a spatial resolution of 1.0 m. Also, we used the 5m resolution Digital Elevation Model provided by the Geospatial Information Authority of Japan (GSI) as a DSM.

4 Results

4.1 Vegetation classification accuracy

We evaluated the performance of the vegetation classifier using a 5-fold cross-validation design. Each fold was stratified with the vegetation categories. We used three standard metrics in machine learning evaluation: precision, recall, and F1-score (2), where TP, FP, FN is true positives, false positives, and false negatives, respectively.

Table 1: Cross validation result

vegetation	f1-score	precision	recall
Macro Average	0.982	0.977	0.987
	(0.005)	(0.008)	(0.001)
Weighted Average	0.997	0.997	0.997
	(0.001)	(0.001)	(0.001)
Dwarf Pine	0.986	0.977	0.995
	(0.005)	(0.012)	(0.003)
Dwarf Bamboo	0.955	0.951	0.960
	(0.004)	(0.013)	(0.006)
Other vegetation	0.997	0.999	0.996
	(0.001)	(0.000)	(0.002)
Non Vegetation	0.970	0.959	0.984
	(0.034)	(0.066)	(0.004)
Sky	1.000	1.000	1.000
	(0.000)	(0.000)	(0.000)

Note:
Mean (SD) of 5-Fold Cross-Varidation

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F_1 = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{4}$$

$$\tag{5}$$

The vegetation classifier achieved a high recall and precision in all five categories 1. Recall and precision were lowest in dwarf bamboo with the smallest training data. Even in dwarf bamboo, false positives occurred at low frequency (about 5%), and only a tiny amount of true positives were missed by the pipeline (about 4%). Since the weighted average of each metric was higher (more than 0.99), balancing the training dataset with vegetation categories may improve these results.

1 shows the products of vegetation classification. We can observe the distribution of dwarf pine and dwarf bamboo at the plant-community scale.

4.2 Georectification accuracy

4.3 Vegetation maps

4.4 Detection and localization of vegetation changes

5 Discussion

We suggested a fully automated procedure to transform time-lapse imagery into georeferenced vegetation maps at a meager cost. This task is challenging because of 1. the difficulty of classifying vegetation with ordinal digital camera imagery and 2. the difficulty of georectification. We solved these issues by 1. using the temporal information of autumn leaf colors for vegetation classification and 2. developing a novel method for

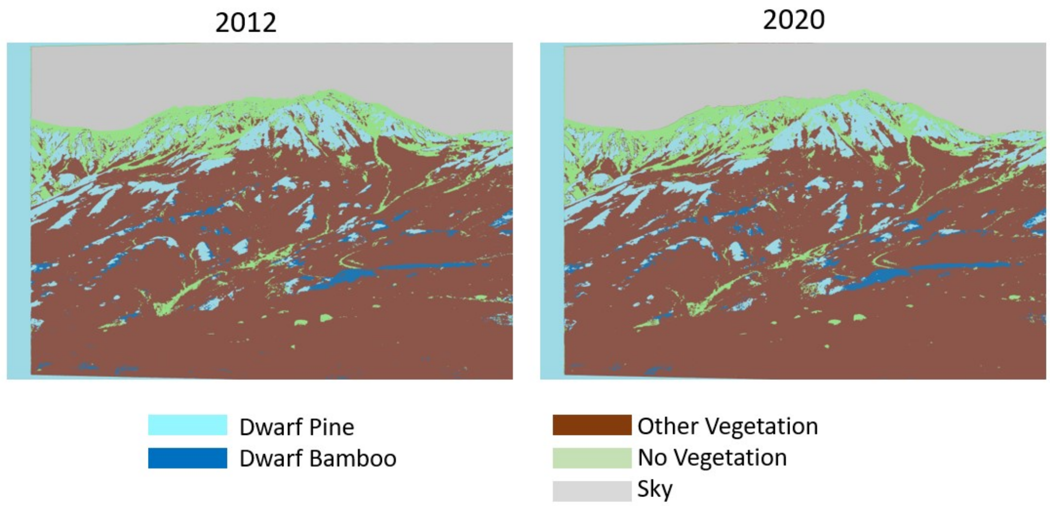

Figure 1: Vegetation classification results

accurate image georectification. Our vegetation classification performance (with an accuracy of 0.87) and georectification methods (with an RMSE of *rmse* m) are practical.

5.1 The benefit of using time-series imagery for vegetation classification

One of the shortcomings of ordinal digital cameras is that they only have three bands (Red, Blue, and Green). We made up for this lack of information by using the rich temporal information that time-lapse cameras can obtain. Many plant species have characteristic phenologies, such as flowering and autumn foliages. Observing this requires long-term (such as year-round) monitoring with a high frequency, and digital time-lapse cameras are suitable. This study only focused on the two species (dwarf bamboo and stone pine) that are easy to classify, even with a single image. However, like the previous studies ([12], [13], [14]), we expect our method to classify more vegetation (e.g., dwarf deciduous trees and alpine herbaceous plants) using the temporal information of leaf colors. We also used hourly images for each day. Fig. x shows that this additional information made our model robust against transient noises, such as shadows. Since alpine ecosystems have rough topologies, shadows are a considerable problem. Time-lapse cameras are also beneficial for this reason.

5.2 The performance of the georectification method and its limitation.

5.3 Future application and conclusion

References

[1] Intergovernmental Panel on Climate Change (IPCC). *Climate Change 2007: ThePhysical Science Basis: Contribution of Working Group I to the Fourth AssessmentReport of the IPCC.* Cambridge University Press, 2007.

[2] Jake M. Alexander, Jonas J. Lembrechts, Lohengrin A. Cavieres, Curtis Daehler, Sylvia Haider, Christoph Kueffer, Gang Liu, Keith McDougall, Ann Milbau, Aníbal Pauchard, Lisa J. Rew, and Tim Seipel. Plant invasions into mountains and alpine ecosystems: current status and future challenges. *Alpine Botany*, 126:89–103, 10 2016.

[3] Gaku Kudo, Yukihiro Amagai, Buho Hoshino, and Masami Kaneko. Invasion of dwarf bamboo into alpine snow-meadows in northern japan: pattern of expansion and impact on species diversity. *Ecology and Evolution*, 1:85–96, 9 2011.

[4] Yukihiro Amagai, Masami Kaneko, and Gaku Kudo. Habitat-specific responses of shoot growth and distribution of alpine dwarf-pine (pinus pumila) to climate variation. *Ecological Research*, 30:969–977, 11 2015.

[5] Gaku Kudo, Mitsuhiro Kimura, Tetsuya Kasagi, Yuka Kawai, and Akira S. Hirao. Habitat-specific responses of alpine plants to climatic amelioration: Comparison of fellfield to snowbed communities. *Arctic, Antarctic, and Alpine Research*, 42:438–448, 11 2010.

[6] Susana Baena, Justin Moat, Oliver Whaley, and Doreen S. Boyd. Identifying species from the air: Uavs and the very high resolution challenge for plant conservation. *PLOS ONE*, 12:e0188714, 11 2017.

[7] Andrew D. Richardson, Bobby H. Braswell, David Y. Hollinger, Julian P. Jenkins, and Scott V. Ollinger. Near-surface remote sensing of spatial and temporal variation in canopy phenology. *Ecological Applications*, 19:1417–1428, 9 2009.

[8] Dawn M. Browning, Jason W. Karl, David Morin, Andrew D. Richardson, and Craig E. Tweedie. Phenocams bridge the gap between field and satellite observations in an arid grassland ecosystem. *Remote Sensing 2017, Vol. 9, Page 1071*, 9:1071, 10 2017.

[9] Reiko Ide and Hiroyuki Oguma. A cost-effective monitoring method using digital time-lapse cameras for detecting temporal and spatial variations of snowmelt and vegetation phenology in alpine ecosystems. *Ecological Informatics*, 16:25–34, 7 2013.

[10] D. M. Woebbecke, G. E. Meyer, K. Von Bargen, and D. A. Mortensen. Color indices for weed identification under various soil, residue, and lighting conditions. *Transactions of the ASAE*, 38:259–269, 1 1995.

[11] Pablo F Alcantarilla, Jesús Nuevo, and Adrien Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *Proceedings of the British Machine Vision Conference*, 2013.

[12] Jan Tigges, Tobia Lakes, and Patrick Hostert. Urban vegetation classification: Benefits of multitemporal rapideye satellite data. *Remote Sensing of Environment*, 136:66–75, 9 2013.

[13] Nguyen Thanh Son, Chi Farn Chen, Cheng Ru Chen, Huynh Ngoc Duc, and Ly Yu Chang. A phenology-based classification of time-series modis data for rice crop monitoring in mekong delta, vietnam. *Remote Sensing 2014, Vol. 6, Pages 135-156*, 6:135–156, 12 2013.

[14] Katharina Heupel, Daniel Spengler, and Sibylle Itzerott. A progressive crop-type classification using multitemporal remote sensing data and phenological information. *PFG - Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 86:53–69, 4 2018.

[15] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 11 1997.

[16] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:2298–2304, 7 2015.

[17] Celine Portenier, Fabia Hüsler, Stefan Härer, and Stefan Wunderle. Towards a webcam-based snow cover monitoring network: Methodology and evaluation. *Cryosphere*, 14:1409–1423, 4 2020.

[18] A. Messerli and A. Grinsted. Image georectification and feature tracking toolbox: Imgraft. *Geoscientific Instrumentation, Methods and Data Systems*, 4:23–34, 2 2015.

[19] Juyang Weng, Paul Cohen, and Marc Herniou. I i ieee transactions camera calibration with distortion models and accuracy evaluation. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 14, 1992.

[20] Nikolaus Hansen, Sibylle D. Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary Computation*, 11:1–18, 3 2003.