A Novel Machine Learning–Based Gap-Filling of Fine-Resolution Remotely Sensed Snow Cover Fraction Data by Combining Downscaling and Regression

SONI YATHEENDRADAS\textsuperscript{a,b} AND SUJAY KUMAR\textsuperscript{b}

\textsuperscript{a} Earth System Science Interdisciplinary Center, University of Maryland, College Park, College Park, Maryland
\textsuperscript{b} Hydrological Sciences Laboratory, NASA Goddard Space Flight Center, Greenbelt, Maryland

(Manuscript received 4 May 2020, in final form 28 July 2021)

ABSTRACT: Satellite-based remotely sensed observations of snow cover fraction (SCF) can have data gaps in spatially distributed coverage from sensor and orbital limitations. We mitigate these limitations in the example fine-resolution Moderate Resolution Imaging Spectroradiometer (MODIS) data by gap-filling using auxiliary 1-km datasets that either aid in downscaling from coarser-resolution (5 km) MODIS SCF wherever not fully covered by clouds, or else by themselves via regression wherever fully cloud covered. This study’s prototype predicts a 1-km version of the 500-m MOD10A1 SCF target. Due to noncollocatedness of spatial gaps even across input and auxiliary datasets, we consider a recent gap-agnostic advancement of partial convolution in computer vision for both training and predictive gap-filling. Partial convolution accommodates spatially consistent gaps across the input images, effectively implementing a two-dimensional masking. To overcome reduced usable data from noncollocated spatial gaps across inputs, we innovate a fully generalized three-dimensional masking in this partial convolution. This enables a valid output value at a pixel even if only a single valid input variable and its value exist in the neighborhood covered by the convolutional filter zone centered around that pixel. Thus, our gap-agnostic technique can use significantly more examples for training (∼67%) and prediction (∼100%), instead of only less than 10% for the previous partial convolution. We train an example simple three-layer legacy super-resolution convolutional neural network (SRCNN) to obtain downscaling and regression component performances that are better than baseline values of either climatology or MOD10C1 SCF as relevant. Our generalized partial convolution can enable multiple Earth science applications like downscaling, regression, classification, and segmentation that were hindered by data gaps.

KEYWORDS: Hydrology; Remote sensing; Satellite observations; Machine learning; Neural networks; Regression

1. Introduction

Spaceborne remote sensing of snow cover fraction (SCF) provides valuable, spatially distributed estimates of snow extent over large geographical and often inaccessible areas. These satellite-based measurements from various sensors cover multiple spatiotemporal scales; however, they might also have significant gaps due to orbital coverage and sensing limitations, especially at finer resolutions of interest. For example, clouds often impact visible and near-infrared frequencies, whereas thin vegetation impedes microwave frequencies.

Some ways of gap-filling satellite-based SCF include the more common spatiotemporal neighborhood persistence-based simple heuristic approaches, and modeling/data assimilation techniques. The temporal persistence techniques rely on the most recent clear sky observation (Hall et al. 2010), while spatial techniques use the information from nearby clear sky pixels. These approaches work best over areas with seasonal snow packs and are insufficient for capturing ephemeral changes in snow from changing weather conditions. Note that the gap-filled portions of the very recently available example gap-filled 500-m MOD10A1F product from the Moderate Resolution Imaging Spectroradiometer (MODIS) sensor aboard the NASA Terra satellite have not been extensively compared against any ground-based “truth” in a formal and rigorous statistical way (Hall et al. 2019). Rather, the MODIS team has visually compared some images to other snow cover products like the IMS snow maps (G. Riggs, NASA, 2020, personal communication).

Modeled estimates that can be used to extend the SCF coverage are subject to uncertainty in their formulations, parameters, and boundary conditions. Physical models with their conceptual approximations of natural processes can be fundamentally deficient in capturing the significant heterogeneity and impacts of human management. In particular, if unmodeled processes are present in the observations, then data assimilation is inefficient in accurately incorporating satellite-based information (Kumar et al. 2015). Alternatively, database regression techniques using multiple inputs can potentially be used to fill example fine-resolution SCF products like the 500-m MOD10A1 from MODIS. Such data-driven approaches will not be impacted by physically based modeling assumptions and could facilitate an effective benchmark that preserves the information content inherent in the observations (Nearing et al. 2016).

In addition to the MOD10A1 product, the MODIS Collection 5 product suite also consists of spatially coarser-resolution SCF products like the 5-km daily MOD10C1. This MOD10C1 with near-complete spatial coverage was originally derived from the MOD10A1 SCF through spatial aggregation...
and represents the areal portion not covered by clouds. This makes it possible to include the MOD10C1 SCF among the set of inputs for gap-filling the MOD10A1 product. For MOD10C1 SCF pixels not completely covered by clouds, such a regression then more specifically becomes a downscaling problem using other inputs as auxiliary information. The SCF values in the MOD10A1 and MOD10C1 products have integer values (that we treat as continuous) denoting the possible percentage coverage. However, in MOD10C1, the SCF, along with its associated cloud cover fraction (CCF) and confidence index (CI) data, together give complementary information on the gaps that existed at the finer MOD10A1 resolution (Riggs et al. 2006). These three variables were actually derived from the MOD10A1 binary data (and not continuous) of snow and cloud cover presence.

This study uses artificial neural networks (ANNs) as the general regression technique. The basic unit of an artificial neural network (ANN) is the neuron that linearly combines inputs through weights and then applies a nonlinear transformation (called an activation function) on the result. A set of neurons having the same set of inputs and outputs is known as a layer, and an ANN is typically composed of a sequence of such layers where the outputs of earlier layers form the inputs to later ones. Legacy shallow ANNs (Cannon 2011) are typically composed of a few processing layers that are dense (or fully connected, with respect to the neurons between adjacent layers).

Recent advances in machine learning called deep learning involve either many layers, or other layer configurations like the example arrangement of inputs or neurons into parallel-stacked grids. A neuron might now be connected to only some neighboring neurons from the previous layer, which along with the gridded arrangement enables operations like convolution on inputs like visual imagery and other gridded data having any neighborhood correlation. These are the convolutional neural networks (CNNs) that originated from the similarly structured “neocognitron” network by Fukushima (1980). A CNN layer is a convolutional filter spanning a small gridded window and operating repeatedly to cover every portion of an image, thereby allowing more efficient characterization of the data features compared to dense ANNs. Downscaling is called “super-resolutionizing” in machine learning parlance. Specifically in computer vision applications, downscaling employs example network architectures like the three-layer super-resolution convolutional neural network (SRCNN) by Dong et al. (2014) that we use in this study for our neural network–based combination of downscaling and regression.

A recent geophysical application of SRCNNs involved their sequentially stacked pair called DeepSD (for deep statistical downscaling) for precipitation downscaling by Vandal et al. (2017). They individually calibrated each component SRCNN in this stack using core input data of precipitation at the coarse resolution or at a (prepared) intermediate spatial resolution as relevant. DeepSD also used an auxiliary input of Earth science information like terrain elevation. Vandal et al. (2017) showed that the DeepSD predicted both average and extreme values simultaneously, outperforming other statistical downscaling methods. These latter methods included the state-of-the-art bias correction spatial disaggregation (BCSD; Wood et al. 2004) used by the climate and Earth science communities, and a suite of off-the-shelf data mining and machine learning methods for automated-statistical downscaling (ASD: Hessami et al. 2008) compared to which BCSD had been shown to perform better. This ASD method suite included logistic and lasso regression, support vector machine classifier and regression (e.g., Ghosh 2010), and artificial neural network classifier and regression. Specifically at extremes, DeepSD consistently outperformed BCSD, mostly with thinner confidence bounds. Regarding ASD approaches, they are already known to perform inadequately at extremes (Bürger et al. 2012), necessitating capture of only the extreme values using specialized approaches such as those using generalized extreme value theory (Hashmi et al. 2011; Mannshardt-Shamseldin et al. 2010). Finally, the DeepSD (or SRCNN) convolution filter’s basic property of explicitly capturing neighboring spatial dependencies is not present in the other statistical downscaling approaches.

Following machine learning terminology, we hereafter denote each input to a CNN as a data channel (or simply as a channel). A major disadvantage of the regular convolution filter is that it cannot adequately handle data gaps in any input channel. Similar to regression and downscaling requiring the entire set of inputs to have valid values at a pixel, the convolution will work only if all input data within the convolution filter’s gridded window are simultaneously valid. This presents a significant challenge for applications involving remote sensing data. Also, even when inland data gaps are absent, any coastal water pixels operated on by the convolution filter qualify as data gaps. If multiple remote sensing inputs (from different sensors) are used in a CNN-based application, their respective different coverage gaps due to orbits, sensor malfunction, weather conditions, and/or surface feature boundaries can severely limit the amount of usable collocated input data regions. Recent studies such as Karpatne et al. (2018) have identified this issue as a significant data-related challenge in geosciences.

A possible machine learning–based solution to handle data gaps during training and to additionally enable gap filling during prediction is partial convolution (see section 2; Liu et al. 2018) that considers only the valid available pixel values in the convolution filter’s window. This is based on the recent work by Liu et al. (2018) at NVIDIA Corporation on gap filling in computer vision applications. As explained more in section 4a, we generalize this technique to handle data gap variation across inputs, enabling creation of a gap-filled MOD10A1-like SCF product for the region of interest in this study. The downscaling component is then not a strict super-resolutionizing problem in the conventional sense of downscaling of only the coarser-scale SCF channel values. In fact, the SCF, CCF, and CI core channels of MOD10C1 need to be considered together as mutually complementary information for creating the MOD10A1-like SCF product channel. In downscaling, the presence of additional auxiliary input channels besides the core input channels from MOD10C1, avoids the possible nonuniqueness of solutions because of multiple fine-resolution pixels contained in any corresponding coarse-resolution pixel area.
A by-product of this study, and yet a major contribution for data-driven modeling in geosciences, is that our generalized partial convolution approach overcomes the limitations of gaps and their variations across input channels. This means that the Earth science community in general will have a deep learning tool that is gap-agnostic for applications in regression, classification, and segmentation.

The remainder of this paper is organized as follows. Section 2 gives some detailed machine learning explanation of partial convolution. Section 3 describes the study domains and datasets used in this study. Section 4 details the methodology including our implementation of our generalized innovation of partial convolution required for this study. Results are presented in section 5, followed by a discussion of our findings in section 6.

2. Background on partial convolution for gap filling

Figure 1a shows an example $3 \times 3$ convolutional filter or kernel operating on the top-left portion of a monochromatic image to calculate the output pixel value near the top left of the output grid. The filter “depth” is the number of input channels: Fig. 1a illustrates a depth of 1, and Fig. 1b shows the moving window-style movement of a $3 \times 3$ filter block (of depth 3) during application over every portion of a three-channel input (from Saha 2018). (c) An increase in the input receptive field for a $3 \times 3$ filter when CNN layers are serially stacked (from Dertat 2017).
omission of the kernel flipping in cross correlation, since this flipping is not usually an important property of a neural network implementation.

Convolution can be used for gap filling (referred to as image inpainting in computer vision). Historical approaches to convolution-based inpainting have involved assigning some valid substitute value(s) at the input’s gap pixels, typically the mean value from the image. Hence, results have often been dependent on such assigned values, giving visual artifacts instead of a smooth blended semantically meaningful look. The usual postprocessing required to remove such artifacts is computationally expensive and not necessarily successful. Additionally, approaches had traditionally focused on rectangular gap regions located near the image center, disregarding the image edge regions. Note that the edges of even gap-free images subjected to the regular full convolution filter behave like they are affected by data gaps originating from the input zones just outside and adjoining the image edges when the moving filter is centered on such edge pixels (Knutsson and Westin 1993).

Among recent innovations in image inpainting, Liu et al. (2018) at NVIDIA Corporation used a 2D partial convolution technique on trichromatic or RGB-channeled (red, green, and blue) images. They showed improved performance during qualitative and quantitative comparisons against other approaches. Partial convolution involves masking to consider only the valid-value pixel locations, and then normalizing the convolution result by dividing by the number of such valid-value pixels. This partial convolution can be seen as a special case of the normalized convolution introduced by Knutsson and Westin (1993). Since the normalization makes the convolution gap-agnostic, the previously mentioned “edge effects” do not occur during partial convolution.

Liu et al. (2018) showed that this method did robust training on irregular hole patterns instead of only rectangular ones in previous studies, and at anywhere in the image instead of just near the center. They also used it to fill gaps in the buffer zones around valid-value regions during the prediction phase. Their implementation in a complex CNN architecture produced visually and semantically meaningful predictions that blended smoothly with the rest of the image (see our Fig. 2 that is reproduced from their article).

The application of partial convolution techniques to our application required some important considerations. The 2D partial convolution in the Liu et al. (2018) study involved gaps that were consistent across the (three) RGB channels of the image. However, our application (like other applications in Earth science, specifically those using remotely sensed and in situ data), is beset with a practical issue that the gaps can differ between the input channels. Hence, a modified and generalized partial convolution procedure is required that can work with the gap patterns that vary across channels. Additionally, the loss functions in the Liu et al. (2018) study were highly complex to provide the required information on the specific spatial patch semantics of RGB channel images in personal photography. Hence, those loss functions will not work effectively for remotely sensed geophysical variables wherein the data are not correlated in the same way both within and
between the input channels. This means that Earth science applications might have to use information from simpler loss functions that are typically used in Earth science, along with additional auxiliary data channels besides the input channels.

Additionally, computer vision super-resolutioning studies involve the same number of channels in both the input and target data. For example, Knutsson and Westin (1993) used a single channel (representing a grayscale image) for both input and output whereas Liu et al. (2018) used three RGB channels. However, super-resolutioning applications (or even general regression applications) in Earth science often involve auxiliary inputs, and the total number of input channels (core, plus auxiliary) will be different from the number of target channels. For instance, Vandal et al. (2017) had the same number (i.e., one) of precipitation channels in the core input and the target, but also had an auxiliary input channel present. Even though our current study application uses a single target SCF channel, the SCF downscaling component uses three core input channels (SCF, CCF, and CI) having complementary information parts of the naturally occurring SCF at coarse resolution, in addition to the other inputs that become auxiliary.

3. Study domain and datasets

a. Study domain and data partitioning

Our study domain of 3° × 3° extent along latitude–longitude is centered over Lake Tahoe near central California in the western United States and includes portions of both California and Nevada (Fig. 3). A significant portion of the domains is occupied by the Sierra Nevada mountain range. Per online elevation information from Wikipedia (2021), the height of the Sierra Nevada increases gradually from north to south. Between the northern and southern edges of our domain, these mountain peaks range from 5000 ft (1500 m) near Fredonyer Pass to almost 14,000 ft (4300 m) at Mount Humphreys near Bishop, California. Near Lake Tahoe, the elevations of the peaks range from roughly 9000 ft (2700 m) to more than 10,881 ft (3317 m).

Wikipedia (2021) also provides information about precipitation in the Sierra Nevada. During the fall, winter, and spring, precipitation ranges from approximately 20 to 80 in. (510–2030 mm), and mostly as snow above around 6000 ft (1800 m). However, most regions eastward of the crest are in a rain shadow, annually receiving less than about 25 in. (655 mm). Summers are dry, with afternoon thunderstorms occurring mostly from North American monsoon in mid- and late summer. This makes Nevada the driest state in the United States. The progression of biotic zones along increasing elevation in the Sierra Nevada is the western foothills of grassland/savanna/woodland, Pinyon pine–juniper woodland, Nevada lower montane forest of pines and Giant Sequoia, upper montane forest of pine/fir, subalpine zone of pines, and the alpine region (Schoenherr 1995). The comparatively mild winter temperature is usually only just low enough to sustain a heavy snowpack. California’s major source of water and significant source of electric power generation is the Sierra Nevada snowpack that is ephemeral in many parts of this spatial domain.

Data used in machine learning are usually partitioned into a training set and an evaluation set (called the validation set) that ensures the absence of overfitting to the training set. When ANNs are specifically used, multiple network sizes and/or architectures are typically experimented with, so that two evaluation sets (validation and test) are required. The test set is used only with the final “best” trained network to confirm similar performance as the validation set. This avoids overfitting to the validation set itself from any possible hidden preferential bias during the experimentation with multiple network architectures and/or sizes. Our study considers only a single network size and architecture (the simple augmented SRCNN), but we use the early stopping criterion that uses information from the validation dataset to stop training; hence, we report the network performance on both the evaluation sets.

The training examples of input-target pairs constitute the training set. For a CNN application in any machine learning framework software, the specification of a training example as being for an entire target image instead of any portion of it (e.g., just an individual pixel) requires developing tailored loss and cost functions. A loss function is defined for a single training example, while a cost function is defined for the entire training set as an average over those loss function values of the examples. Our current study considers an image as target in a training example; however, note that we derive our loss function from the output values of the individual target pixels in that image. Our cost function is then the average of loss function values across images instead of across all images’ pixels.

A minimum of 1 year of data is typically preferable for training to capture the full intra-annual range of behavior. We consider a minimum of one year (2011) of data for training, an additional larger 3-yr training dataset (2009–11) for tackling any possible overfitting to the 1 year of data (i.e., to try
eliminating any variance or any difference in performance between the training and validation data), an example year 2012 for validation data, and finally an example nonconsecutive year 2014 to these as test data. As information about years having potentially different hydrometeorology than regular years, we provide the following information about drought years. While California has had many drought periods, the ones relevant to our study are 2007–09 that saw some of the worst wildfires in Southern California history during the 2007 summer, and the longest and driest 2011–16 since record-keeping began. These droughts were separated by a very wet 2010/11 season that occurred during a strong La Niña phase (Wikipedia 2020).

b. Data channels

In addition to the MODIS SCF products used as the target and the core input channels for the downscaling component, we also consider other products as auxiliary data channels for this component but as the main set of data channels for the regression component:

1) static terrain-related ones, namely, elevation, slope, and aspect;
2) some dynamic inputs and outputs of a land surface model (LSM), such as precipitation, snow water equivalent (SWE), surface radiative temperature and leaf area index (LAI), and SCF; and

1) STATIC TERRAIN

The global high-resolution (30-m resolution) DEM data from the Shuttle Radar Topography Mission (SRTM; Farr et al. 2007) are used to derive terrain maps of elevation, slope, and aspect. Conforming to the 1-km resolution model in this study, we used the SRTM resampled to 1 km.

2) DYNAMIC FIELDS FROM LAND SURFACE MODEL

We include LSM fields in the auxiliary channels to try to incorporate the relevant dynamics and constraints of physically based process variations. The LSM used is Noah-MP version 3.6 run at 1-km resolution inside the Land Information System (LIS) software (Kumar et al. 2006; Peters-Lidard et al. 2007). It used the modified MODIS-IGBP as the land cover map from NCEP (National Centers for Environmental Prediction) and the STATSGO+FAO blended soil texture map from NCAR (National Center for Atmospheric Research). Among our auxiliary channels, precipitation is an input to Noah-MP from the data of phase 2 of NASA’s North American Land Data Assimilation System (NLDAS-2: Xia et al. 2012a,b), while the rest (SWE, surface radiative temperature, LAI, and SCF) are Noah-MP output fields. The NLDAS-2 rainfall input to Noah-MP is a product at a spatial resolution of 0.125° and disaggregated to hourly data from the daily CPC-Unified gauge-only analysis (Xie et al. 2007; Chen et al. 2008) before 2012 or the operational CPC (Climate Prediction Center) product after 2012. We applied the topographic corrections of lapse rate and slope-aspect in LIS to the NLDAS-2 radiation (Cosgrove et al. 2003), and bilinear interpolation from the NLDAS-2 precipitation values to get corresponding values on our LIS output grid.

3) MODIS C5 SATELLITE-BASED PRODUCTS

In addition to the MOD10A1 and MOD10C1 SCF products, the MOD11A1 Land Surface Temperature (LST) product and the MOD10A1 snow albedo product, we also describe here the MOD10L2 swath-based snow cover product. This is because the MOD10L2 snow cover product stage is the earlier and completely relevant step in the standard MODIS products’ production pipeline for MOD10A1 and MOD10C1 snow cover.

The MOD10L2 500-m products include binary snow presence (Riggs et al. 1994, 2006; Hall et al. 1995, 2002; Klein et al. 1998) and SCF. In addition to the reflectance and surface temperature criteria used to produce the binary snow product, the SCF was also calculated using the statistical-linear regression equation of Salomonson and Appel (2004, 2006). The creation process of the MOD10A1 500-m sinusoidal daily products of binary snow and SCF involves selecting the appropriate pixel from the corresponding MOD10L2 products using a scoring algorithm to select an observation for the day.

The MOD10C1 SCF product production at 0.05° resolution entails spatially aggregating the MOD10A1 binary snow presence data (and not the MOD10A1 SCF data). Corresponding generated maps of cloud cover fraction or percentage (CCF) similarly consist of aggregating the cloud pixel information from the MOD10A1 binary snow presence data. Note that the different land pixel type values in the MOD10A1 binary snow data like snow, cloud, snow-free land, and others indicating bad data (e.g., missing data, night, detector saturation) are mutually exclusive. To account for this exclusivity and also to allow for snow presence in reality under clouds for the land pixels designated as cloud in MOD10A1 (resulting in some snow ending up being an inherent part of MOD10C1 CCF instead), MOD10C1 also has an expression product of confidence called the confidence index of the SCF values. If a 0.05° cell contained 12% or greater land, then it was considered land and analyzed; if less than 12% it was considered ocean.

The sum of MOD10A1 land counts was the calculation basis for the MOD10C1 values of SCF, CCF and CI. For example, the MOD10C1 SCF was calculated as percentage snow = 100 × count of binary snow presence observations/ count of land observations. Finally, the CI was also calculated but from the count of cloud-free or clear land observations (i.e., both snow and snow-free land pixels). A high CI indicates cloudless conditions and good data values and that the SCF is an estimate of very good quality. As an example calculation for a 0.05° cell containing 50 MOD10A1 observations split as 20 for snow, 15 for snow-free land, 10 for cloud, and 5...
for other data (but not water), the SCF was 40% \([=100 \times 20/(20 + 15 + 10 + 5)]\), the CCF 20% \([=100 \times 10/(20 + 15 + 10 + 5)]\), and the CI 70% \([=100 \times (20 + 15)/(20 + 15 + 10 + 5)]\). Note that if an entire MOD10C1 cell is covered by clouds, then its SCF and CI values will both be 0%, regardless of how many of the spanned MOD10A1 pixels by that cell are snow or snow-free in reality.

The MOD10A1 snow albedo product is created from the visible and near-infrared bands using the MODIS land surface reflectance as input (Klein and Stroeve 2002). The MOD11A1 LST product was generated by the generalized split-window LST algorithm (Wan and Dozier 1996; Wan 2006). Our study only uses MOD11A1 pixel values having a quality control (QC) value of 0 denoting good quality, and discards pixel values other QC values, for example, like 1 that denotes other quality.

4. Methodology and the neural network architecture

a. Methodology and its implementation

1) PARTIAL CONVOLUTION AND COMPARISON METRICS

The widely used two-dimensional (2D) full convolution procedure on 2D input images actually involves multiple 3D-shaped filters acting on 3D-shaped input data portions (see section 2). Each filter results in each single-depth 2D output channel that are then stacked across-channel to actually give a 3D output shape of depth greater than one. A partial convolution modifies this procedure by masking to use only valid values consistent across the input data block depth and normalizing (or dividing) the result by the number of such valid values. The spatial consistency in data gaps across the input RGB channels in Liu et al. (2018) means that the across-channel axial dimension is redundant for gap information, effectively making this a 2D masking. This means that the fraction of valid values in the input will be the same regardless of whether we consider an individual channel of that input or the full across-channel stacked input. Our solution approach differs from the Liu et al. (2018) study in the following aspects:

1) Data gaps vary across the input channels in our study, meaning that the third depth dimension also now carries gap information. We call our generalization of partial convolution to give truly 3D masking as a depth-included partial convolution, hereafter abbreviated as DIPConv [and we hereafter denote the Liu et al. (2018) partial convolution as PConv]. Our version will work even if just a single valid value exists anywhere (and in any input channel) in the 3D input portion considered by the filter.

2) We use the simple mean squared error (MSE) loss function. Liu et al. (2018) had instead used complex loss functions that consider spatial patch semantics specific to RGB channels. But similar to the loss functions in the Liu et al. (2018) study, our developed version of the loss function also considers only the valid-data pixels in the target image.

3) For downscaling, our study augments the information from the coarse MOD10C1 SCF input using auxiliary inputs, and for downscaling, it uses only the auxiliary inputs.

For any input channel, the percentage of its values that are usable when considered by itself (meaning that if it had been the only channel in the input) would be the same as when considered together with other channels having the same gap pattern, for a multichannel input during PConv. It will also be different from the percentage of its usable values when it is considered together with other channels for a multichannel input during DIPConv (due to the variable gap information along the depth axis in the latter). In PConv, valid pixels are those for which all channels (inputs and target for training, or only inputs for prediction) simultaneously have valid data, forming an across-channel validity intersection operator.

DIPConv differs from PConv in applying an across-channel validity union operator instead on the inputs, so that just one valid value in any input channel in the spatial receptive field (refer to Fig. 1a or section 2) of the target pixel is enough to calculate a valid output value at that target pixel (but a minimum of 2 valid input values or channels in the spatial receptive field is perhaps preferable). Figure 4 gives a simple example illustration of the across-channel validity operator for PConv and DIPConv. When the MOD10C1 channels have at least one valid value, this becomes a downscaling component instead of regression. The union operator of DIPConv can be advantageous in obtaining a larger data sample size as compared with PConv. We created Keras API layers for implementing DIPConv and PConv in the TensorFlow/Keras machine learning framework (Abadi et al. 2015). These layers conform to the coding style and user-friendliness of the Keras full-convolution API layers, plus the choice of updating the mask during convolution at the boundary of the valid-pixel regions.

We train the network using MSE as the loss function, but report its root-mean-squared error (RMSE) form. Our subjective goal at the outset in this study was to achieve an initial RMSE in SCF of around 10%, with an attempted refinement to 5% if possible. Another metric relevant for comparison against snow presence/absence products (thus different from our continuous-value target SCF) that we report is the binary accuracy that is calculated by binning the predicted SCF values into a snow versus no-snow using a threshold at 50%. Due to different initial random seeds giving different trained models on the graphical processing units that we conducted these runs on, we did three runs for each training and report the results for the run that has the lowest value of the higher RMSE from among the training and validation set performances of that run. For the 3-yr training periods, the difference in performance between the runs was always less than 1% RMSE in SCF.

To give some context to the predictive performances, we use dynamic climatology as one baseline. We create climatology for each pixel and day of the year using corresponding values from years 2004 to 2015. Our other baseline is simply
1) in the remaining 3 auxiliary channels. Having minimum one valid value, and 3 valid values (minimum is downscaling having the bare-minimum number (one) of valid values in DIPConv being causing a minimum of total 4 valid values. (b) An example of the minimum single location in each channel needs to have a valid value, to have actual dimensions 3 \times 3. Partial convolution pixel location of this 3 \times 3 input portion for an application accepting 4 inputs (shown by 4 MOD10A1 pixels.

![Image](https://via.placeholder.com/150)

**Fig. 4.** Example schematic of valid values (in green) in a 3 \times 3 input portion for an application accepting 4 inputs (shown by 4 grids on each row), that can give a valid output value at the center pixel location of this 3 \times 3 domain portion when subject to a 3 \times 3 partial convolution filter. Hence, this input portion or this single filter have actual dimensions 3 \times 3 \times 4. (a) An example of the bare-minimum number of valid values in PConv where the same minimum single location in each channel needs to have a valid value, causing a minimum of total 4 valid values. (b) An example of the bare-minimum number (one) of valid values in DIPConv being enough to provide a valid output value at the center pixel location. (c) Perhaps more than one valid value is better, and so 2 total valid values shown for DIPConv. (d) This subplot is specifically for downscaling having the first (leftmost) channel as the coarser input having minimum one valid value, and 3 valid values (minimum is 1) in the remaining 3 auxiliary channels.

the MOD10C1 SCF cell value applied on its spanned MOD10A1 pixels.

2) REDUCING BIAS AND VARIANCE

A typical machine learning workflow aims to reduce the bias (the training set performance to be yet achieved) and the variance. Some bias reduction tools include bigger networks, better ANN architectures, better optimization algorithm, longer training, and specific forms of regularization like early stopping during training that we used in this study. Our study is limited to using the sophisticated Adam optimization algorithm (adaptive moment estimation; Kingma and Ba 2014) on a single network size and architecture. Variance reduction tools include using a bigger training set, which we implement by increasing the temporal domain.

3) DATA TRANSFORMATION AND TARGET OBSERVATION MASKING

We normalize every input to the same range of \([-0.5, 0.5]\) and hence the same order of magnitude as is typically done in machine learning. We avoided any pronounced skew in the input distributions by using appropriate mathematical transformations. Logarithmic and exponential transformations are typically applied to distributions skewed toward the lower and higher ends of the range, respectively. Per a visual inspection of such input distributions and to make them closer to a normal distribution, we applied a logarithmic transformation to the MOD10C1 CCF, the LSM total precipitation and the LIS SWE, and an exponential transformation to the MOD10C1 CI.

We temporally split the available data so that we train using 1 or 3 years of data and then evaluate using other years of data. We hereafter refer to such consideration of the entire valid-data region of the target for any day as the regular observation. We also need to consider the special case of MOD10C1 cells with zero CI, specifically that a MOD10C1 cell’s SCF and CI are both zero when it is fully masked by clouds, and its spanned MOD10A1 pixels will have invalid data values, regardless of the amount of actual snow area hidden by clouds but present in reality that we need to predict through gap-filling [see section 3b(3)]. Hence, the training portion regions for zero-CI cells of MOD10C1 need to be created by a synthetic spatial masking of portions of both the nonzero-CI MOD10C1 cell regions and the valid MOD10A1 data pixels spanned by these cells, for a regression mode of application (because zero CI of MOD10C1 essentially denotes absence of MOD10C1 information). This can then be used later predicting MOD10A1 SCF values under MOD10C1 cells having zero CI. Note that the MOD10A1 albedo and MOD11A1 LST data are also set to invalid values in these synthetic spatially masked regions of zero CI for MOD10C1, since they are also affected by the same clouds.

We create the abovementioned synthetic mask by converting a third (33\%) of the nonzero CI MOD10C1 cells to zero-CI ones through random sampling (we also experimented with converting a half or 50\% of those cells but the results are similar and so not shown). We hereafter refer to the valid portion remaining of the MOD10A1 observation after the abovementioned masking as the masked observation, and its complementary portion (i.e., the created zero-CI MOD10C1 region) from within the same MOD10A1 regular observation region as simply the complementary region. In the masked observation region, at least one of the MOD10A1 pixels spanned by any (nonzero CI) MOD10C1 cell will have valid values that will form the target for the downscaling mode during training. The error characteristics of the missing MOD10A1 pixels to be gap-filled in this masked observation region will be the same as those of the valid-value pixels in this region.

To allow for the zero-CI MOD10C1 cell portions to train by themselves without including (via the convolution filter) the nonzero-CI MOD10C1 information existing at the cell boundaries of the zero-CI MOD10C1 cell portions, we added additional examples to the training data by keeping only the complementary portion of the inputs and target and masking out the rest. This effectively enabled the filter to capture cluster-type behavior of zero-CI MOD10C1 data that might have
existed if the created mask had mimicked clusters similar to clouds patterns, instead of randomly located cells obtained from a random sampling. After training, the batch performance obtained is a composite value of three separate components: the masked observation portion, the complementary portion that possibly includes the nonzero-CI MOD10C1 information existing at the portion’s boundaries via the convolution filter (these first two components use only the original examples mentioned further above), and again the same complementary portion but it does not include such information (this last component uses only the additional created examples mentioned further above). Hence, we separately further calculate these three separate components of the batch performance. The latter two components can be considered as representing scattered cloud cells and large cloud patches, respectively. Going from the first to the third component also indicates a transition from downscaling to regression.

We also minimally attempt to capture seasonal snow characteristics by separate trainings to individual seasons for the above combinations of 1- and 3-yr training periods with the regular and masked observation targets. Following standard practice, we define 3-month seasons as winter starting in December and ending in February, and so on.

b. The SRCNN architecture

We leverage an existing convolutional network structure typically used for downscaling, for our combined regression–downscaling application. For downscaling, the core input channels in our study are the spatially interpolated versions (to the target fine resolution of 1 km) of the original coarse-resolution MOD10C1 images using the nearest neighbor technique. This enables the same spatial resolution for the data channels throughout the network.

Computer vision studies typically have no auxiliary inputs during super-resolving of three-channel color image data (e.g., RGB) or of one-channel grayscale data in earlier studies. An example is the SRCNN by Dong et al. (2014) for fast and state-of-the-art image restoration quality when compared against other methods. The three layers of their lightweight SRCNN respectively handled overlapping spatial-patch feature extraction/representation from the coarse-resolution image, nonlinear mapping to high-resolution patch representations, and reconstruction by combining the predictions within a spatial neighborhood to generate the final high-resolution image. We adopt this same SRCNN architecture but add auxiliary input channels, similar to Vandal et al. (2017), but where the auxiliary inputs become the main inputs when a regression operation is required instead of downscaling. Figure 5 shows the data flow schematic for our auxiliary channel-augmented network version that we hereafter denote as P-SRCNN915, where the “P” stands for partial convolution, and the numerals in the “915” string stands for the filter sizes. These filter sizes, and the number of filters being 64, 32, and 1, respectively, are the same as those used by Dong et al. (2014).

5. Results
a. Improvement in data usability and statistics

We first examine the percentage of valid and usable values during training. Figure 6 denotes these percentages by circular filled markers when each channel is considered only by itself [sole-channel input; see section 4a(1)], by solid lines for PConv, and by dashed lines for DIPConv [see section 4a(1)]. Inland water bodies like Lake Tahoe show up as data gaps for our study domain, so that no data channel will have complete (100%) valid-data coverage. The satellite-based auxiliary channels have more data gaps due to issues like poor-quality or missing data: the amounts of available data for the MOD10A1 snow albedo, the MOD11A1 LST, and the MOD10A1 SCF target are near zero (the lowest), 40%, and two-thirds, respectively.

![Data flow schematic for 3-layer SRCNN](image-url)
The MOD10A1 snow albedo marker values form the upper bounds for the corresponding intersection-based PConv solid-line percentages in Fig. 6 (an upper bound and not an equality because of reduction by gaps from other channels collocated with the valid-data regions of MOD10A1 snow albedo). However, the DIPConv’s validity union operator on the auxiliary channels means that the near-complete areal coverage of the LSM dynamic and static channels become the prediction coverage also during downsampling. Further intersection with the target mask during training limits the size of the training and validation sets (see MOD10A1 SCF markers coinciding with the dashed lines in Fig. 6). Hence, DIPConv has significantly increased the usable-data percentages from the near-zero values for PConv.

We also examined univariate distribution statistics like the mean and standard deviation. Regardless of considering usable values of each data channel by either itself or PConv or DIPConv, the LSM SWE statistics for training (specifically for the 1-yr training period of year 2011) differs markedly from that for validation in having higher values of these statistics (not shown). For most channels, we also found that the DIPConv was better able than PConv to reproduce statistics that were obtained when those channels were considered by themselves for usable data. For example, DIPConv did not give a large decrease of 15–20 K in the mean of MOD11A1 LST compared to when MOD11A1 LST was considered by itself, that PConv showed. Thus, DIPConv’s validity union operator has allowed the inclusion of much drier no-snow conditions (low SCF values) during the year in addition to the wet (snow) conditions only that was considered by PConv procedure. This allowed better training at both extremes by the SRCNN.

**b. Batch performance**

Figure 7 shows the batch RMSE performances across different training sizes (Figs. 7a,b). For training to the 3-yr period or its seasons, Fig. 7c displays the component performances of the batch, and the corresponding baseline performances for dynamic climatology and MOD10C1 SCF are illustrated in Figs. 7d and 7e, respectively [see section 4a(3) for description of the seasons, components and baselines]. For Figs. 7a and 7b, each x-axis label is either all seasons taken together, or each season. The 3-yr training period (entirely or its seasons) in Fig. 7b shows better closeness and negligible variance (<1% SCF) for any season between the training and validation RMSEs, as against more than 4% SCF variance seen in Fig. 7a for winter when snow is actually dominant.

Note that for the 3-yr training period, there is also an insignificant difference in predictive performances for any season between using the model trained to the same season (shown in Fig. 7b) and using the model trained to all seasons together (not shown). For Fig. 7 and the upcoming spatial performance figures, we show the results from the former (i.e., the model trained to each season). Also note that we found these results to have no significant difference from similar runs that did not have the LSM SCF variable included among the inputs, that is, that variable seems to not provide any additional information for improving the performance.

In Figs. 7c–e, each x-axis label has a semicolon separating two abbreviations, where the abbreviation before the semicolon denotes the (spatial) component and that after denotes the RMSE-calculation season (or all seasons taken together): all seasons for the first component are listed together, followed by the same for each succeeding components (with
Similar to Fig. 7b, the Fig. 7c also shows insignificant variance (~2% SCF) between the training, validation, and test data at any x-axis label. This is also manifested by the training RMSEs being slightly higher than the corresponding validation and test ones: the training size seems to have captured well the range of data characteristics across years. Also, for any season (or all seasons together), the RMSEs increase from the first component (representing the masked portion of MOD10A1 observation having nonzero-CI MOD10C1 cells) to the third component (representing its complementary portion in the observation), showing the relative ease of downscaling in filling MOD10A1 gaps compared to regression. Among the seasons, RMSEs for summer and winter are the lowest and highest, respectively, as was seen in Figs. 7a and 7b also. This is understandable since the dataset is mostly dominated by no-snow or negligible snow pixels in the entire dataset of images, resulting in their values showing the least error, and reflected by low RMSEs during the summer season when images are almost completely filled with such values (and vice versa during winter).

The prediction performances in Fig. 7c are noticeably better than the corresponding baseline performances in Figs. 7d and 7e. For the first component, the MOD10C1 SCF values have lower RMSEs than those in dynamic climatology, and so are a tougher baseline to beat than the latter. For the vertical dotted lines separate the components).
remaining two components, there are no MOD10C1 SCF baseline values shown since MOD10C1 CIs and SCFs are supposed to be zero for the areal portion represented by these components. Note that the dynamic climatology baseline values for the second and third component in Fig. 7d are the same because they use the same complementary masking. To summarize, Fig. 7 illustrates that our machine learning-based batch component prediction performances are better than the corresponding baseline ones, with downscaling doing significantly better than regression.

c. Spatial plots of batch error and performance

Figures 8 and 9 illustrate the spatial plots of the RMSEs and binary accuracies, respectively, for the first component denoting the masked observation [see section 4a(3)], with each row of subplots being for each season. The three columns are for the machine learning prediction, dynamic climatology, and MOD10C1 SCF, respectively. Consistent with the order of performances for the batch and its components in section 5b, the predictions are found to perform better than both baselines, with the MOD10C1 RMSE being a tougher baseline to beat than the dynamic climatology. Overall, the higher elevations perform worse than the lower ones, which is understandable because complete or near-complete coverage-based snow pixels usually exist at higher elevations and during winter. Another reason is that the training domination by no-snow or negligible snow pixels (see section 5b) means that the higher-SCF observation pixels perform worse.

Similar to Figs. 8 and 9, Figs. 10 and 11 illustrate the spatial plots of the RMSEs and binary accuracies, respectively, for the second and third components that denote the complementary portion [see section 4a(3)]. While each row of subplots is again for each season, the first two columns show the machine learning predictive performance for the second and third components, respectively, followed by the third column illustrating the performance of the MOD10C1 SCF baseline. Again, consistent with the batch predictions in section 5b, the second component performs better than the third due to possibly including the nonzero-CI MOD10C1 information existing at the convolution filter boundaries, and both these components perform better than dynamic climatology. The lower performance in winter and at higher elevations seen in Figs. 8 and 9 is also reproduced in these figures.

Also note that we had done separate trainings using only the additional data examples mentioned in section 4a(3) that are relevant to the performance of only the third component for checking whether that component’s performance can improve from these current numbers. However, we did not find any significant difference in performance for the third component between such trainings and the trainings for which we have shown results in this and the last section, thus showing that the latter trainings have fully captured the required characteristics of the data of the third component.

d. Individual-day performance

Figure 12 compares the individual-day predictions and performances between the 3-yr training to all seasons together, and to each season separately training (for which we have shown batch and component performances in the previous two sections), for some example days from January 2011 in winter to June 2011 in summer. Note that the trainings were done to the masked observation portion. The first three rows are for the period during January–April in the snow season, and the last row in June is representative of the end of the snow season when only traces of snow remain. The spatial plot patterns match between the middle and rightmost columns, and the RMSEs and binary accuracies differ by an insignificant number (less than 2%) between those columns for each row. Note that these RMSEs and binary accuracies are for the masked observation region (for which coverages are actually 47.5%, 65%, 63.3%, and 65.2% of the image area for subplot rows from first to fourth, respectively), and not for the entire regular observation or prediction regions for which the percent valid data are reported. Hereafter we only show the results from the each-season trainings over 3 years.

Next, Fig. 13 shows individual-day performances for the same example days as Fig. 12. The first column is again the regular observation, while the next two columns are the same as the rightmost column of Fig. 12, but with the synthetic maskings of the masked observation and complementary portion superimposed in black color on the middle and rightmost columns, respectively. These latter two columns are the first and the second performance components, respectively. The middle column shows a low of 2.2% RMSE during the summer day of 24 June, and a high of 11.7% during the winter day of 6 January, while the corresponding numbers in the third column are 5.8% and 16.5%, respectively. As in Fig. 12, the RMSEs and binary accuracies in the middle column are for the masked observation portion (for which coverages are reported in the previous paragraph as 47.5%, 65%, 63.3%, and 65.2% for the shown days), whereas the same performance metrics in the rightmost column are for the complementary mask portion in the regular observation (for which coverages for these days are 22.4%, 32%, 31.9%, and 32%): and so adding these numbers for each row gives the reported percent valid data coverages in the first column (e.g., 47.5% + 22.4% = 69.9% reported in Fig. 13a).

Figures 12 and 13 had the inputs subject to the masked observation mask (and performances reported in Fig. 13 for this same mask and its complementary as the first and second performance components). In contrast, the middle column of Fig. 14 shows the prediction after training to the inputs actually subject to the complementary mask, and the rightmost column shows this same prediction having that same complementary mask superimposed in black color (and this gives the third performance component). This shows a low of 6.2% RMSE during the summer day of 24 June, and a high of 20.4% during the winter day of 6 January.

6. Discussion

Similar to data gaps present in remotely sensed data in general, the satellite-based SCF observations like MOD10A1 products also have data gaps that pose significant limitations for their use in Earth science applications. Available derived
FIG. 8. Spatial plots of batch RMSEs, for the masked observation component region (i.e., the nonzero-CI MOD10C1 cells). Rows correspond to seasons; columns in order are predictions from 3-yr training to each season, followed by the dynamic climatology and then the MOD10C1 SCF baselines. The no-data regions are the lake areas mask. Domain-wide RMSEs for the subplots are given in Fig. 7 as numbers at the x-axis labels starting with “Comp1” for the seasons in Figs. 7c–e that correspond to the first, second, and third columns in this figure, respectively.
Fig. 9. As in Fig. 8, but showing binary accuracies instead of RMSEs. Domain-wide values not reported.
FIG. 10. Spatial plots of batch RMSEs, for the complementary components region (i.e., the created zero-CI MOD10C1 cells). Rows correspond to seasons; columns denote second and third component predictions, respectively, after 3-yr seasonal trainings, and dynamic climatology. The no-data regions are the lake areas mask. Domain-wide RMSEs for the subplots are given in Fig. 7: the numbers at the x-axis labels starting with “Comp2” for the seasons in Fig. 7c correspond to the leftmost column here, those at the labels starting with “Comp3” for seasons in Fig. 7c correspond to the middle column here, and those at the labels starting with “Comp2” or “Comp3” in Fig. 7d correspond to the rightmost column here.
FIG. 11. As in Fig. 10, but showing binary accuracies instead of RMSEs. Domain-wide values not reported.
FIG. 12. Observed and predicted MOD10A1 SCF patterns for some selected days in 2011. (left) MOD10A1 SCF regular observations, (center) predictions from masked training to all seasons together, and (right) those predictions from training to masked observations from their respective relevant seasons. All subplots show the respective percent valid-data coverages. The center and right columns also report the RMSE and the binary accuracy for the masked (and not regular) observation portion. Each subplot also shows gaps from the LSM’s lake masks.
FIG. 13. Observed and predicted MOD10A1 SCF patterns for the same selected days in 2011 as in Fig. 8. (left) Regular observations, (center) predictions with additional black-colored synthetic mask superimposed (inputs were also subject to this mask during training in the form of masked observation), and (right) the same predictions with an additional complementary black-colored mask superimposed. The statistics reported for the center and right columns consider the combined masking of the lakes (shown as missing data) and the respective black-colored maskings.
FIG. 14. Observed and predicted MOD10A1 SCF patterns for the same selected days in 2011 as in Fig. 8. (left) Regular observations, (center) predictions after training was done to the complementary region of masked observation (with inputs also subject to this same mask during training), and (right) the same predictions with an additional complementary black-colored mask superimposed. The statistic reported for the center column considers the maskings of the lakes (shown as missing data), and for the right column considers additionally the black-colored masking.
coarser-resolution SCF products like MOD10C1 and other auxiliary geophysical products have less or no gaps, hence our study attempts to gap-fill the finer-resolution MOD10A1 by downscaling and regression from these products. Specific to the coarse-resolution MOD10C1 product, any such gaps in its cells are represented as zero values of SCF and CI that together reflect no valid data in the MOD10A1 pixels spanned by those cells. The MOD10A1 pixels that do not receive any input values from nonzero-CI MOD10C1 cells are where the regression happens using auxiliary inputs, instead of downscaling.

Suited to images, the use of new and advanced gap-agnostic techniques such as deep learning–based CNNs has potential for such an application using information from adjacent pixel values and additional auxiliary image data from other satellite-based and physically based model products. However, such input images can also contain data gaps that can impart resultant errors during the downscaling or regression. Recent partial convolution capabilities in machine learning are gap-agnostic and enable convolution calculations even in the presence of data gaps that are consistent across the input channels.

This article presents an innovation to overcome the requirement of gap consistency across input channels by generalizing the partial convolution methods to work even when data gaps vary across the inputs. While the typical partial convolution procedure only allows using less than 10% of the pixels for training or prediction in our downscaling application, the generalized partial convolution approach described in this manuscript enables the use about two-thirds of the data examples for training, and almost all the examples for prediction. This represents a significant increase in the representativeness of our data analysis. Even with the simple three-layer legacy super-resolution CNN (SRCNN) network augmented with auxiliary input channels, we obtained batch performances that beat the baselines of dynamic climatology and MOD10C1 specification, in terms of RMSE and binary accuracy in the SCF. The performance is seen to be better for the downscaled locations than for the locations where neural network-based regression happens. Additionally, the individual-day spatial predictions and performances and spatial performance maps demonstrate the effectiveness of our approach. The CNN-based demonstration presented here provides an effective strategy for gap-filling SCF products, different from the empirical approaches developed in prior studies.

Neural networks are typically implemented using machine learning software frameworks like the TensorFlow/Keras framework that we have used in our study. The user can easily just swap in our partial convolution layers into any network of choice, instead of the regular Keras layer APIs that are widely used. Our achieved performance can obviously be further improved using multiple techniques including experimentation with the network architecture and data. The network architecture can be made deeper and more complex by stacking more layers in series or in parallel.

Our downscaling technique is similar to state-of-the-art statistical downscaling techniques on other geophysical variables, like bias correction spatial disaggregation (BCSD) and SRCCN-based architectures like DeepSD that downscale at averages and extremes simultaneously. Yet the ability of our generalized partial convolution version to function even in the presence of spatial gaps that vary across input channels makes it a very powerful tool. It is truly able to use spatial neighborhood information to downscale or predict in regions where some observational inputs may be sparse or absent or of poor quality. These usually occur in the poorest regions that are most affected by climate change, and where such downscaled data may be needed the most for adaptation.

We expect that the advancements presented in this article will pave the way for gap-filling and extending the information and coverage of other remote sensing datasets. Similar gap-agnostic partial convolution approaches can also be used in data driven modeling applications in Earth science. The advancement to partial convolution-based deep learning from our downscaling-based SCF gap-filling study thus serendipitously overcomes a critical challenge in geosciences of data gaps (Karpatne et al. 2018), enabling the use of CNNs across multiple types of Earth science applications where such data gaps have been a hindrance. The erstwhile restriction of needing data gaps to be consistent across input channels had hampered the widespread usage and adoption of neural networks and CNNs. In general, the advancements presented here provide the Earth science community with a gap-agnostic deep learning infrastructure for applications in regression, classification, and segmentation.

Acknowledgments. Funding for this work was provided by the NASA Goddard Space Flight Center’s Internal Research and Development (IRAD) program. Computing was supported by the Advanced Data Analytics Platforn (ADAPT) at the NASA Center for Climate Simulation.

Data availability statement. All MODIS data used during this study are either openly available from the NASA National Snow and Ice Data Center Distributed Active Archive Center [e.g., Hall et al. (2006a) for MOD10C1], or derived from them using reprojection and nearest-neighbor interpolation when converting from the original 500-m resolution [e.g., Hall et al. (2006a) for MOD10A1] to 1-km resolution. The static terrain data, the dynamic LSM data used by us, and the partial convolution code created by us will be made publicly available at https://doi.org/10.5281/zenodo.6478716 within a week of the final published version of this manuscript.

REFERENCES


