CLASlite

Sensing Technology For Forest & Agriculture

VERSION 3.3 USER GUIDE

Updated: April 7, 2020
https://claslite.org/

In addition to this User Guide, CLASlite Installation Manual is available in English and Spanish (https://claslite.org/manual/)

i-Cultiver, Inc.
404 Clipper Cove Way
San Francisco, CA 95376

i-CULTIVER
AGRICULTURE FOOD NUTRITION SCIENCE

WELCOME TO THE CLASlite USER GUIDE

The Carnegie Landsat Analysis System – Lite (CLASlite) is a software package designed for highly automated identification of deforestation and forest degradation from remotely sensed satellite imagery. This software package, its algorithms, and any derivatives are fully protected under U.S Patent 8189877-B2.

This guide provides information on CLASlite's scientific background, technical processes, outputs, potential use, and limitations.

We trust that CLASlite will contribute to your organization's forest monitoring efforts. For further information and general inquiries about the CLASlite program, contact us by e-mailing info@i-cultiver.com or visit our website at https://claslite.org/

TABLE OF CONTENTS

WHAT'S NEW IN CLASLITE V3.3

Version 3.3 of CLASlite provides technical advances in a variety of areas described throughout this manual. The new capabilities include:

- Sentinel-2 satellite reflectance support
- Google Earth Engine support
- Reduction of processing time by supporting USGS reflectance images
- Batch processing of forest cover images and full control of forest control criteria
- Opening of all deforestation, disturbance, exclusion pixels, and filter criteria values for the user to change

Plus:

- Ability to process Landsat 8 imagery
- Windows 64-bit support
- New "Prepare Landsat" tool that prepares raw Landsat image files in appropriate format, with template preparation
- Two new masking options; 'No masking' and 'Fmask. Fmask is only available for users running the Windows 64-bit version.
- Extended spectral libraries for all tropical forests, from lowlands to mountain ecosystems
- Improved deforestation and disturbance mapping algorithms
- Faster spectral mixture analysis of satellite imagery supporting forest cover and change mapping
- User-controlled image artifact removal to customize deforestation and disturbance mapping output
- Batch processing of deforestation and disturbance images, up to 1000 entries at a time
- New download manager that initiates *CLASlite Setup*; more automated set-up process

Mission & Vision

Mission

To expand the capacity of governments, non-governmental organizations, and academic institutions to map and monitor tropical forests with highly advanced and scientifically-based forest monitoring technology.

Vision

Following decades of scientific development, forest monitoring with satellites should be an everyday activity for non-experts, thus helping to improve environmental conservation, forest management, and resource policy development.

With CLASlite, governments, non-governmental organizations, and academic institutions of tropical forest nations will:

- Supplement their environmental toolkit with powerful, automated, user-friendly remote sensing technology.
- Map, monitor, and quantify the forest resource from a personal computer.
- Share information, best practices, and feedback within the CLASlite community, collectively and individually increasing forest-monitoring efficacy.

Community feedback based on using CLASlite in a variety of geographies and applications will guide further research and CLASlite technical development, continually advancing this technology to support forest-monitoring efforts of the CLASlite community.

THE CLASLITE TEAM

i-Cultiver, Inc. is working in collaboration with Greg Asner (creator of CLASlite), and with Carnegie Institution for Science (carnegiescience.edu) to manage the CLASlite software.

Gregory Asner, Ph.D.
is the **Creator** of the CLASlite software

Rajnish Khanna, Ph.D.
is the **Founder & CEO** of i-Cultiver, Inc.

ABOUT CLASlite

At a time when awareness of the role of forests in carbon storage, climate change mitigation, and biodiversity protection has dramatically increased, the Department of Global Ecology at the Carnegie Institution for Science was seeking to rapidly advance the science of mapping forests to support internationally policy discussions, and to respond with applied solutions that address on-the-ground needs for forest monitoring.

The Carnegie Landsat Analysis System – Lite (CLASlite) is a software package designed for highly automated identification of deforestation and forest degradation from remotely sensed satellite imagery. Developed by Greg Asner and his team at the Carnegie Institution, CLASlite incorporates state-of-the-art research in remote sensing into a simple, user-friendly yet powerful computer program intended for non-profit institutions and governments in need of technologies for forest monitoring and environmental planning.

CLASlite is the result of more than a decade of biophysical remote sensing research and fieldwork that provides an automated satellite mapping approach to determine one of the most important components of tropical forest structure: *fractional cover* of vegetation canopies, dead vegetation, and bare surfaces. These fractional covers are core determinants of ecosystem composition, physiology, structure, biomass, and biogeochemical processes. Fractional cover analysis sits at the heart of CLASlite, making it a powerful, stable and biophysically-grounded tool that allows for rapid forest monitoring with error tracking.

We at the CLASlite team have expanded our software capabilities and are capacity building for regional- and national-level forest monitoring. We are disseminating the technology through a tailored, demand-driven transference of CLASlite to government, academic and non-government (non-commercial) institutions based on available grant funding.

Currently, CLASlite supports input from nine different satellite sensors: Landsat 4 and 5 Thematic Mapper, Landsat 7 Thematic Mapper Plus, Landsat 8 OLI/TIRS, SPOT 4, SPOT 5, NASA ASTER, NASA Advanced Land Imager (ALI) data, and Sentinel-2.

CLASLITE FUNCTIONS

CLASlite includes core functions to extract land cover information from raw satellite data, generating images and maps to support forest monitoring efforts. These processes include Calibration of raw imagery to apparent surface reflectance, Automated Monte Carlo Spectral Unmixing (AutoMCU) of reflectance data to fractional cover, classification of fractional cover data into a map of forest cover, and change detection with multi-temporal fractional cover data to map deforestation and forest disturbance. These functions are reflected in the steps of CLASlite's User Interface, shown in Fig. 1.

Figure 1: The CLASlite User Interface

CLASLITE 64-BIT SUPPORT

64-bit support for CLASlite depends on the Windows version installed. If the Windows version is 32-bit, CLASlite will run in 32-bit mode and if the Windows version is 64-bit CLASlite will run in 64-bit mode. In other words, the user is not presented with a choice to run CLASlite in 32-bit or 64-bit mode. Running CLASlite in 64bit mode has great benefits that will be discussed below.

Benefits

- Running CLASlite in 64-bit mode will allow users to utilize the full extent of their system RAM, making CLASlite run faster, based on how much RAM is available.
- Running larger images is now possible with CLASlite 64-bit mode and the size of the MOSAIC image is only limited by the hardware resources that are available [described below in more detail].
- Adding Fmask as a masking option that only runs on CLASlite 64-bit mode [described below in more detail].

How do I know if my system will run CLASlite in 32bit or 64-bit mode?

This depends on the Windows version installed. To check your Windows version please refer to the following website:

http://windows.microsoft.com/en-us/windows7/find-out-32-or-64-bit

Caution:

Users should exercise caution when running a MOSAIC image in CLASlite 64-bit mode, and should be aware of their computer's specification before processing MOSAIC images. Because CLASlite creates temporary files, the user is required to have adequate free space in the drive that CLASlite has been installed on. Increasing the size of your input image will cause an increase in the size of temporary files created. Step 4 takes up most space when it creates temporary files. Table 1 contains an example of how much space is needed when you increase the size of the input images that are used in Step 4. The images in case 2 are based on a MOSAIC of 37 adjacent Landsat path/rows.

Table 1: CLASlite step 4 size of temporary files created based on size of input files

	Case 1 [one path/row] Size(MB)	Case 2 [large MOSAIC] Size(MB)
Fractional Cover Image Year 1	750	49000
Fractional Cover Image Year 2	750	49000
Reflectance Image Year 1	650	42000
Reflectance Image Year 2	650	42000
Total size of temporary files created	11500	750717

The deforestation and disturbance calculation [step 4] requires 4 input files, two fractional cover files and two reflectance files. In Table 1, we show two cases. Case 1 is a normal image that consists of a single path/row. Here CLASlite will need 11.5GB of free hard drive space to process the deforestation and disturbance image. Case 2 represents a large MOSAIC of multiple path/rows with an input size of

49GB for one of the fractional cover images and 42GB for one of the reflectance images. Here CLASlite will need 750.8GB of free hard drive space to process the deforestation and disturbance image. That means CLASlite will need more than 65 times the space when processing a single path/row from its default path/row. Also this number is only for a two year change, if you want to create a map compilation of three or more years then you need more free hard drive space. Also, depending on the size of the image, there could be RAM limitations. So, the more RAM you have, the better. In other words, the user must be aware of the free hard drive space available before running a large MOSAIC of adjacent path/rows. A suggestion for freeing up hard drive space would be to put the input image on a different drive from the drive that CLASlite is installed on. That means the user will free up 182GB more space for CLASlite to run safely without any crashes in case 2. The user also should have at least 10GB of free space for Windows to keep it running in an efficient manner.

How do I know if my system is running the correct CLASlite mode [32-bit or 64-bit]?

For troubleshooting purposes, CLASlite will check the mode when launching the software, and if the Windows version is 64-bit, then CLASlite will display a 64-bit mode in the title of the program [as shown in Fig. 1].

If the Windows version is 32-bit, then CLASlite will not display the mode in the title of the program, since this has been the default for all CLASlite versions.

Which input MOSAIC images should I run; raw, reflectance, fractional cover, or deforestation and disturbance?

Now that CLASlite has a 64-bit mode, all four steps can be processed for large images [i.e. MOSAIC]. Although it is recommended to run the calibration step separately for each image and then create a MOSAIC of all the reflectance files. This will be essential because the user now can select between 4 masking options in the calibration step [explained below]. And one large MOSAIC will be forced to use only one masking option for all path/rows. Fmask will require the user to MOSAIC each individual TIF file for each band as a MOSAIC. In addition Fmask is very memory intensive and it deals with the whole image while calculating all the masks. In addition, the default masking option in CLASlite requires a substantial amount of memory.

RECOMMENDATION: The CLASlite team currently runs all images through calibration [step 1] separately, then after evaluating each image independently we create our desired MOSAIC images to run through CLASlite for step 2, 3, and 4.

STEP 1: CALIBRATION TO REFLECTANCE

OVERVIEW

All image sensors, from your eyes to your personal camera to an optical mapping satellite, are remote sensors. All optical imaging sensors are designed to measure the variation in color from pixel to pixel. Raw imagery can be calibrated and atmospherically corrected to reveal valuable surface reflectance information (i.e. color), a critical element in vegetation mapping.

SCIENTIFIC BACKGROUND AND TECHNICAL PROCESS

Radiometric Calibration

When a satellite-based sensor records data, it detects energy reflected from the land surface and the atmosphere between land and the sensor head. These data are collected by the imaging system for on-board digital storage and/or transmission to ground receiving stations.

To use an image quantitatively, however, the data registered in each pixel must be calibrated from units of digital numbers or counts, to units of reflected energy. This process is called *radiometric calibration*. For radiometric calibration, CLASlite uses conversion factors (gains and offsets) made available by the providers of satellite sensors (i.e. NASA, SPOT, etc.). The result of radiometric calibration is an image of in units of radiance (i.e. watts per square meter per unit of solid angle), also known as the energy measured by the satellite-based sensor.

Atmospheric Correction

Radiometric data contain information about both the Earth's surface and its atmosphere. Thus, to work with vegetation (surface data), it is necessary to minimize the contribution of the atmosphere to the values of each pixel in the satellite image. This is accomplished through a process called *atmospheric correction*, which minimizes the effect of water vapor (humidity), aerosols (from dust, volcanoes, etc.), and other factors.

To apply atmospheric correction, CLASlite uses the 6S radiative transfer model (Vermote et al. 1997), which simulates the Earth's atmosphere in each satellite image. Using data from NASA's Moderate Resolution Imaging Spectroradiometer (MODIS) sensor, 6S models the effect of the atmosphere on sunlight as it passes through the atmosphere, interacts with the land surface, and returns through the atmosphere to the satellite sensor. The raw image is then "corrected" by removing the estimated model of the atmosphere, resulting in an image of surface reflectance (i.e. units of percentage, represented by integers 0 to 10000, where 10000 corresponds to 100%).

For every image, CLASlite has fully automated the integration of monthly averages of MODIS data, corresponding to the image's acquisition date, into the 6S code.

Masking

Mapping with optical satellite sensors (Landsat, SPOT, etc.) requires radiance data to determine the reflectance of each pixel, which is the information required to extract information about vegetative cover. No satellite sensors can collect this radiance information on the land surface through clouds, in darkened shadow areas under clouds, or in shadows caused by steep terrain. Thus, clouds and cloud shadows, as well as terrain shadows must be masked, or excluded from the image analysis. These, as well as bodies of water, are automatically masked out of each image during Steps 1 and 2 of CLASlite.

CLASlite has two masking steps: The first is in step 1 for image calibration, and the second is in step 2 for the fractional cover map. The masking that occurs in step 1 eliminates clouds, cloud shadows, topography shadows, and water. The second eliminates MCU errors based on the RMSE values [the seventh band product] in the fractional cover image. Our users have requested the ability to mask images independently using third party software. Thus they have requested the option to produce images without any masking. CLASlite 3.2 gives the user the ability to choose 'No Masking', alongside 'Reduce masking' and 'Fmask', which is discussed below. No Masking will prevent any masking in the calibration step, but the second masking in step 2 is still implemented and is an important factor to reduce MCU errors based on the RMSE band. The 'No Masking' feature is available for all satellites.

CLASlite cloud masking algorithms were optimized for South America in general, with some emphasis on South Africa and East Asia. CLASlite now has a broad user base that includes users from more than 125 countries. So we have decided to add a third party masking option. Fmask [Zhu et al. 2012] is a function that detects clouds, cloud shadows, snow, land and water for Landsat images [8, 7, 5, and 4]. Fmask only runs on Windows 64-bit and needs at least 4GB of RAM to run. In addition the function allows user-driven pixel dilation around clouds, cloud shadows, and snow. For a detailed explanation of the science behind Fmask, we refer our users to the reference provided [Zhu et al. 2012]. The following website is also very useful and informative: https://code.google.com/p/fmask/

The masking options are shown in Fig. 2 below:

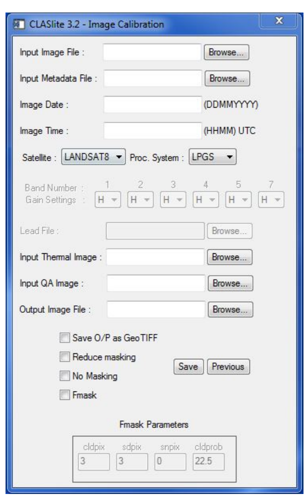


Figure 2: Image calibration interface

Here the user has four masking choices for each image.

1. Normal CLASlite image masking [no option chosen, default masking].

2. Reduce masking [check box].
3. No Masking [check box].
4. Fmask [check box].

The ‘No Masking’ option simply produces reflectance images without any masking performed on the image.

FMASK

Fmask will only be active on Windows 64-bit machines. Users that have Windows 32-bit will notice that Fmask check box is disabled. Choosing the Fmask check box will activate the ‘Fmask Parameters’ that will allow the user to choose the degree of masking intended for each image. There are four ‘Fmask Parameters’ and they are described in the following website (https://code.google.com/p/fmask/):

1) 'cldpix' is dilated number of pixels for cloud with default values of 3. For CLASlite [0-10].
2) 'sdpix' is dilated number of pixels for cloud shadow with default values of 3. For CLASlite [0-10].
3) 'snpix' is dilated number of pixels for snow with default values of 0. For CLASlite [0-10].
4) 'cldprob' is the cloud probability threshold with default values of 22.5 (range from 0~100).

We have used the default parameters recommended by the authors [Zhu et al. 2012].

Note: Calibration in batch mode now supports ‘No Masking’ and ‘Fmask’ options, in addition to the four Fmask parameters.

Caution:

Fmask depends solely on the TIF files and the MTL file, and CLASlite will automatically look for those files in the same folder the _raw, _therm, and _QA [Landsat 8] input files are. We urge our users to keep all input files and raw TIF files in the same folder while running step 1 and step 2 [for auto run purposes]. We also encourage our users to use our naming conventions to minimize possible errors.

Is it possible to run a raw MOSAIC image [multiple adjacent path/rows] through step using Fmask as a masking option?

As we have mentioned above, Fmask depends solely on the TIF files and the MTL file for the image required for masking. First users should know that Fmask [Zhu et al. 2012] uses the MTL file to convert raw digital numbers to top of the atmosphere (TOA) reflectance. Then TOA reflectance is used in the Fmask algorithms to estimate the final mask. It would be very hard to create one MTL file that would be accurate to processes multiple images. In addition, the MOSAIC could have a mix of Landsat 8, 7, 5, and 4 images. Second the user has to MOSIAC all the raw TIF files individually, meaning band 1 together, band 2 together, etc. Fmask is very memory intensive and running a MOSAIC could cause the program to crash. We recommend that the user calibrates all the raw images individually, and then creates a MOSAIC based on the reflectance or fractional cover outputs.

OUTPUT

The result of CLASlite image calibration is a reflectance image, providing spectral bands calibrated from raw data to apparent surface reflectance. The number of bands varies by sensor (e.g. Landsat: 6 reflectance bands; SPOT: 4 bands, etc.)

Reflectance Image Band Analysis

After an image has been converted to apparent surface reflectance, the image can be viewed with the ENVI Freelook or similar software to review the reflectance profile for each pixel. In Fig. 3, bands 5, 4, 3 (RGB) are displayed in this Landsat 7 example image.

If you place your cursor over a forested pixel in ENVI Freelook, you can see a spectral profile of that pixel by selecting Options → Z Profile.

In the spectral profile, the X-axis represents the band number and the Y-axis represents reflectance (% times 100). In this example image, there are 6 bands that represent six of the seven spectral bands provided by Landsat 7. (The seventh band, Band 6, is a thermal image.) You can see that the vegetation has a near-infrared (NIR, Band 4) reflectance of more than 30%, while the red reflectance (band 3) is only 2%. This difference between the NIR and red reflectance is characteristic of vibrant, green vegetation. The line between these two bands is often referred to as the “red edge”.

You can also see that in the visible bands (1, 2, and 3), this pixel is brighter in the green (Band 2) than in the blue and red (Bands 1 and 3, respectively). Vegetation absorbs more light in the blue and red parts of the spectrum, leaving green as the color we see with the naked eye.

In a pixel that has little to no vegetation, you can see that the NIR reflectance is lower and the red reflectance is higher than in vegetated pixels. The typical form that we saw in the vegetation with peaks at bands 2 and 4 does not exist in the spectrum of this pixel, indicating that there is less vegetation. In addition, reflectance in bands 5 and 7 (the shortwave-infrared or SWIR) is much higher in pixels with less or no live, green vegetation. This indicates the presence of dead, brown vegetation, soils, and rock.

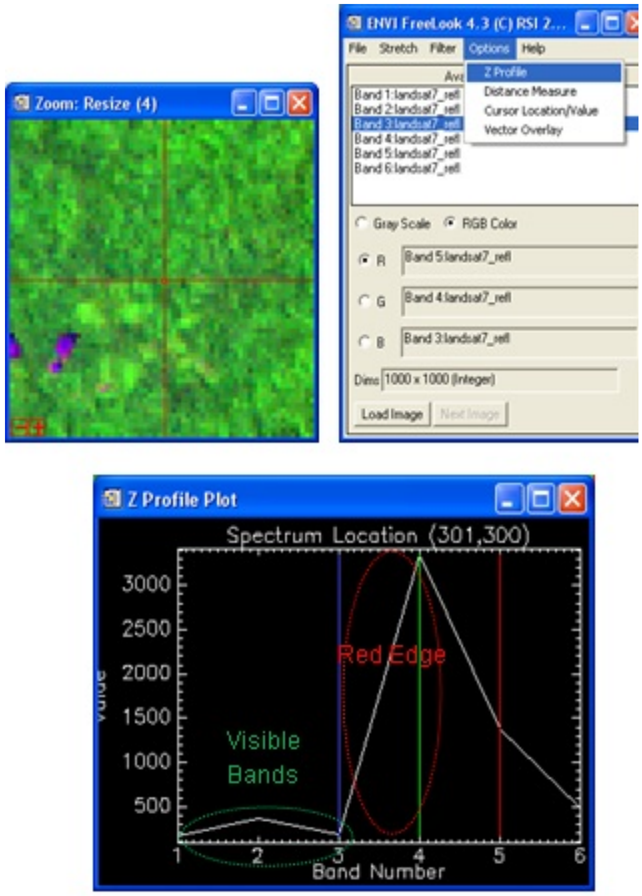


Figure 3: Pixel reflectance profile from a Landsat 7 image

Landsat Climate Data Record (CDR)

USGS has made available Landsat 8, 7, 5, and 4 Surface Reflectance High-Level Data Products on the earth explorer website:

http://earthexplorer.usgs.gov

Under Landsat CDR Category, and we are happy to support these images from preparation to fractional cover conversion. The user can prepare reflectance images for CLASlite for single image preparation and batch image preparation. Also, we have added forest cover batch processing that is supported by template preparation as well. Reflectance and forest cover template preparation support up to 1000 images. Here we will show the new interface that has added support for Reflectance and forest cover template preparation.

STEP 2: AUTOMCU FOR FRACTIONAL COVER

OVERVIEW

Different types of Earth surface covers have different reflectance properties. In other words, each component of the Earth surface has a spectral signature. From these spectral signatures, it is possible to derive information on each pixel in a reflectance image. Since the 1960s, it has been possible to generate maps of land cover from remotely sensed imagery using classification techniques, which assign a whole pixel to a class (i.e., forest, rock) based on the spectral signature in the pixel. This type of thematic classification is useful for land-cover mapping, but it often has reduced sensitivity to small variations and changes in forest cover that occur at the sub-pixel, or within-pixel, scale. Since we want to map deforestation and forest degradation occurring at the sub-pixel scale, we must use a different approach.

CLASlite is the result of more than a decade of biophysical remote sensing research and fieldwork that provides an automated satellite mapping approach to determine arguably the most important characteristic of any forest: *the fractional cover* of live vegetation canopy, dead vegetation, and bare surface within the forest ecosystem. These fractional covers are core determinants of forest composition, structure, biomass, physiology and biogeochemical processes. Fractional cover analysis lies at the heart of CLASlite, providing a powerful, stable and biophysically-grounded algorithm that allows rapid forest monitoring with error tracking.

AUTOMCU – SCIENTIFIC BACKGROUND AND TECHNICAL PROCESS

The AutoMCU, or Automated Monte Carlo Unmixing (Asner 1998, Asner and Heidebrecht 2002, Asner et al. 2004), provides quantitative analysis of the fractional or percentage cover (0-100%) of live and dead vegetation, and bare substrate within each satellite pixel (e.g., within each 30 x 30 m pixel in a Landsat image). Live vegetation is technically referred to as Photosynthetic Vegetation (PV) because live vegetation maintains unique spectral properties associated with leaf photosynthetic pigments, canopy water content, and the amount of foliage in the canopy. The dead or senescent vegetation fraction is termed Non-photosynthetic Vegetation (NPV), which is expressed in the spectrum as bright surface material with spectral features associated with dried carbon compounds in dead leaves and exposed wood. Finally, bare substrate is often dominated by exposed mineral soil, but can also be rocks and human-made infrastructure (e.g. brick).

The AutoMCU was initially developed for forest, savanna, woodland and shrubland ecosystems (Asner 1998, Asner and Lobell 2000, Asner and Heidebrecht 2002), and was later redesigned for tropical forests (Asner et al. 2004, 2005). The method requires "libraries" of spectral endmembers for each of three relevant surface cover types: bare substrate, photosynthetic vegetation, and non-photosynthetic vegetation. Endmembers are reference spectra that are chosen as pure representatives of a given surface material, and they are intended to encompass the spectral variability within that surface material. These libraries, derived from extensive field databases and satellite imagery, are used to decompose each image pixel using the following linear equation:

$$\rho(\lambda)_{pixel} = \Sigma\,[C_e \bullet \rho(\lambda)_e] + \varepsilon = [C_{pv} \bullet \rho(\lambda)_{pv} + C_{npv} \bullet \rho(\lambda)_{npv} + C_{substrate} \bullet \rho(\lambda)_{substratel}] + \varepsilon \quad (1)$$

where $\rho(\lambda)_e$ is the reflectance signature library (e) at wavelength λ and ε is an error term. Solving for each sub-pixel cover fraction (C_e) requires that the satellite observations ($\rho(\lambda)_{pixel}$) contain sufficient spectral information to solve a set of linear equations, each of the form in equation (1) but at different wavelengths (λ).

The tropical forest spectral libraries provide the spectral reflectance signatures required by the AutoMCU sub-model: $\rho_{pv}(\lambda)$, $\rho_{npv}(\lambda)$, and $\rho_{substrate}(\lambda)$. The AutoMCU is a probabilistic approach based on canopy physics (Asner 1998) that reduces each image pixel into the three constituent cover fractions of PV, NPV and bare substrate.

The AutoMCU Spectral Libraries

For the tropical forest spectral library used in CLASlite, both the bare substrate and NPV spectra were collected using ground-based field spectroradiometers (FR and FS-3 Analytical Spectral Devices, Inc., Boulder, Colorado USA). The bare substrate library incorporates a diverse range of mineral soil types, surface organic matter levels and moisture conditions. The NPV spectra library includes surface litter, senescent grass, deforestation residues (slash), and other dry carbon constituents collected from a wide range of species and decomposition states.

In contrast to bare substrate and NPV, the PV spectra of forest canopies require overhead viewing conditions, which is difficult with trees reaching heights of more than 50 meters. Spectral measurements of individual leaves, stacks of foliage, or partial canopies (e.g., branches) introduce major errors in spectral mixture models requiring canopy-level information (Asner 2008). To develop a canopy-level spectral library for CLASlite, PV spectral data were collected using the Earth Observing-1 (EO-1) Hyperion sensor (Ungar et al. 2003), which is the only spaceborne imaging spectrometer launched by NASA for environmental applications. Hyperion data were collected over many tropical forest control sites in Brazil, Peru and elsewhere from 1999 to 2012, providing many millions of spectral observations made at 30-m resolution (Asner et al. 2005, Asner 2008, Asner *unpublished data*). These hyperspectral data were atmospherically corrected to reflectance and convolved to the spectral channels used by the Landsat, ALI, ASTER, and SPOT sensors in CLASlite. As a result, these datasets incorporate the highly variable effects of intra- and inter-crown shadowing, common in tropical forests (Asner and Warner 2003). In total, the spectra represent more than 250,000 field and spaceborne spectrometer observations.

Automated Monte Carlo Unmixing (AutoMCU)

The AutoMCU iteratively selects a PV, NPV and bare substrate spectrum from each library, and unmixes the pixel reflectance into constituent cover fractions using equation (1). CLASlite adopts a Monte Carlo method, whereby the possible combinations of the endmember spectra are pre-computed, and are applied during the AutoMCU run. The process of random selection is repeated up to 50 times or until the solution converges on a mean value for each surface cover fraction. In the original CLAS [20], the iteration was done dynamically until a stable standard deviation between successive fractional cover estimates was reached. Following a series of studies on different tropical forests, we found that 50 iterations per pixel is usually sufficient to achieve a stable solution based on this Monte Carlo approach, and thus this value is fixed in CLASlite (Fig. 4).

An advantage of the Monte Carlo approach is that the per-pixel iterations produce a standard deviation of the estimate for PV, NPV and bare substrate fractions (Fig. 4). These are output from CLASlite as standard deviation images. In addition, a final analysis of the fit of the modeled spectrum (right side of eq. 1) to the input spectrum (left side of eq. 1) is computed for each pixel, leading to a root mean squared error (RMSE) image.

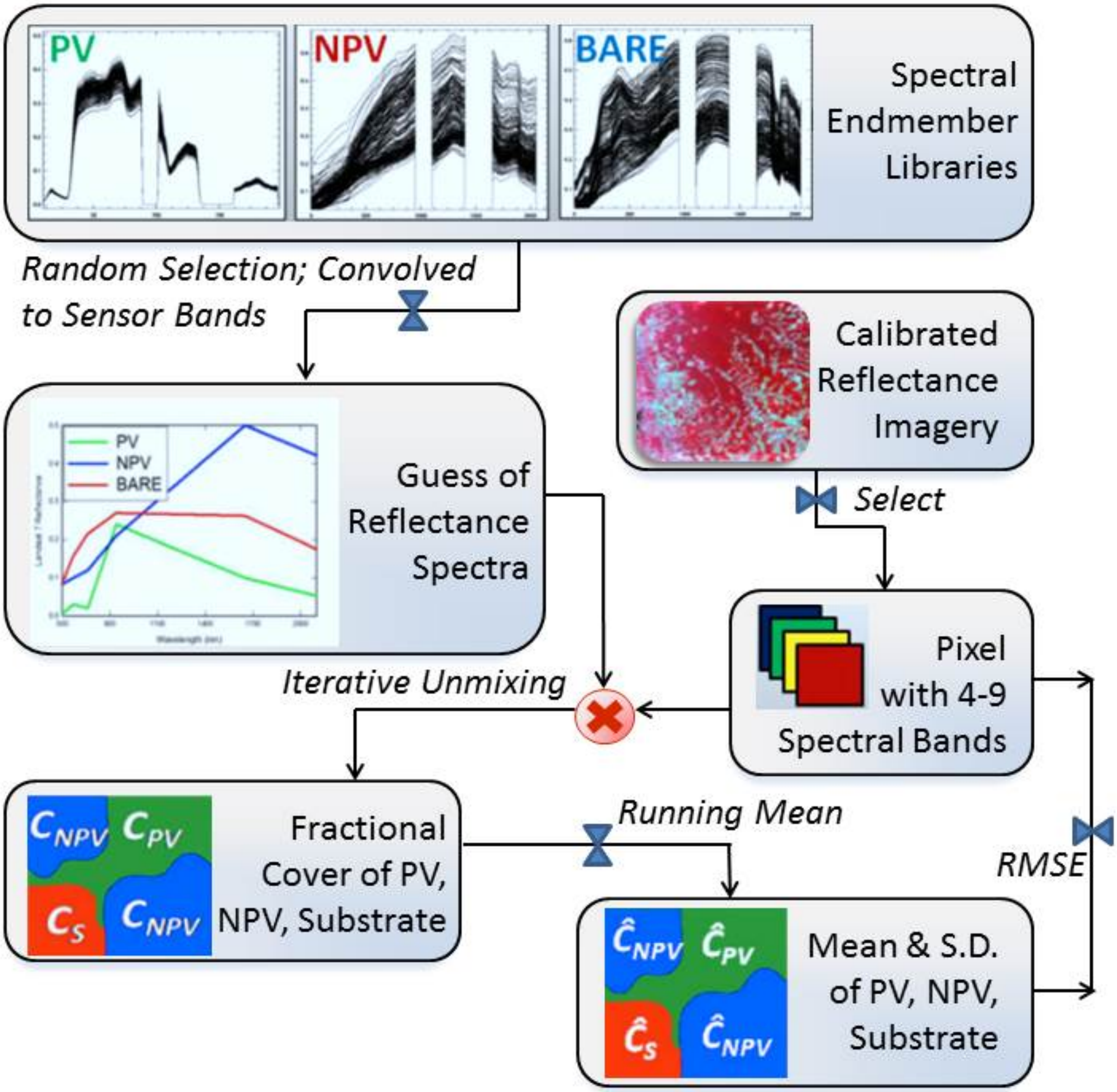


Figure 4: Processing Stream for the Automated Monte Carlo Unmixing (AutoMCU) sub-model within CLASlite

OUTPUT

The output from AutoMCU in CLASlite is a 7-band image containing information about fractional cover of PV, NPV, and bare substrate, uncertainty estimates for each cover fraction, and total error for each pixel in the image.

Fractional Cover Image Bands

Band 1	Fractional cover of *bare substrate (S)*, expressed as a percentage (0-100%)
Band 2	Fractional cover of *photosynthetic vegetation (PV)*, expressed as a percentage (0-100%)
Band 3	Fractional cover of *non-photosynthetic vegetation (NPV)*, expressed as a percentage (0-100%)
Band 4	Uncertainty of the *S* fraction, expressed as the standard deviation of AutoMCU iterations
Band 5	Uncertainty of the *PV* fraction, expressed as the standard deviation of AutoMCU iterations
Band 6	Uncertainty of the *NPV* fraction, expressed as the standard deviation of AutoMCU iterations
Band 7	Total error, expressed as the RMSE of the modeled versus observed reflectance signature

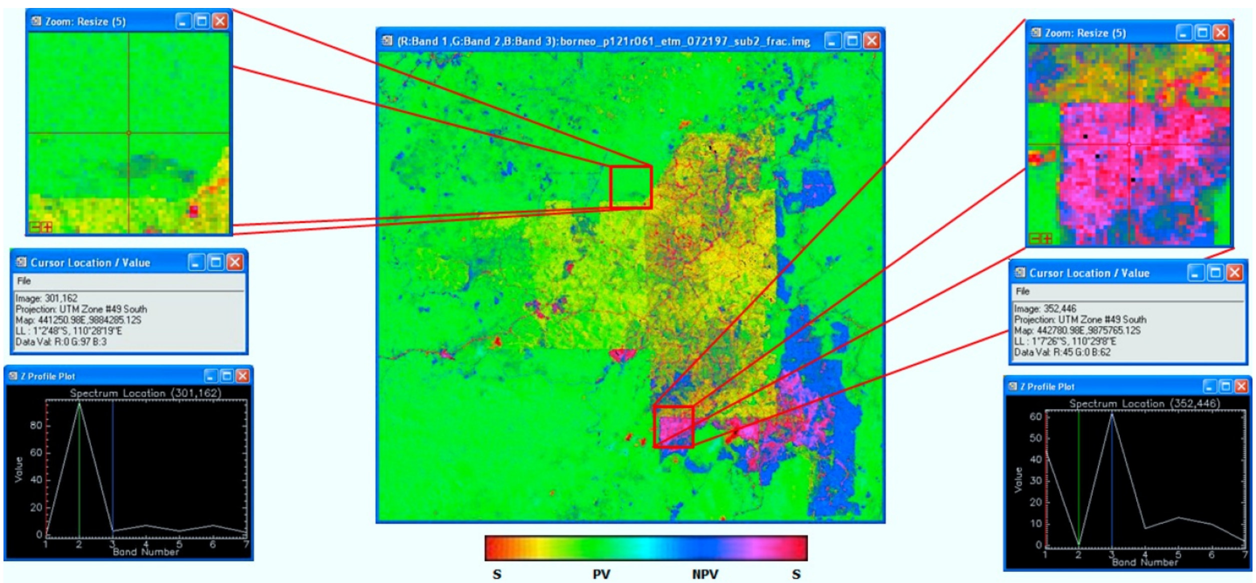


Figure 5: The fractional cover output from CLASlite

The fractional cover image can be analyzed visually, by displaying a color composite of bands 1-3 in an image viewer like ENVI Freelook. In Fig. 5, Band 1 (Fractional cover of **S**) is displayed in **red**, Band 2 (Fractional cover of **PV**) is displayed in **green**, and Band 3 (Fractional cover of **NPV**) is displayed in **blue**. The intensities of each color represent presence of each cover type in each pixel. For example, greener pixels have higher percentage of PV, yellow pixels indicate the presence of both S and PV, while bluer pixels represent higher fractional coverage of NPV.

The image can also be analyzed quantitatively through evaluation of the image band values. Bands 1-3 represent the fractional cover of PV, NPV, and S, expressed in percentages (0-100%). Because these fractions are the output of a probabilistic model, the percentages may not sum exactly to 100%. Bands 4-6 represent uncertainty of each cover fraction, expressed as the standard deviation (SD) of the 50 AutoMCU iterations for each cover type. Higher values indicate increased uncertainty. Band 7 represents the total error, or root mean squared error (RMSE) of the observed versus the modeled reflectance spectra. Combined, the standard deviation and RMSE images provide a way to assess the performance of the AutoMCU on a pixel-by-pixel basis, allowing you to identify areas of concern. Such areas can occur when a vegetation type is not well represented in the spectral libraries, in areas where inorganic materials are present (e.g., infrastructure), or atmospheric disturbances remain unmasked from other CLASlite steps (e.g., edges of clouds, severe haze).

STEP 3: FOREST COVER CLASSIFICATION

OVERVIEW

CLASlite can be used to map *forest canopy cover* from a single satellite image. The patterns and spatial orientation of forest and non-forest cover within these forest cover images often indicate areas of past clear-cutting and disturbance (e.g., an sharply delineated patch of non-forest within an otherwise forested swath), as well as natural non-forested vegetation such as grasslands and shrublands (e.g., an expansive non-forest area with natural borders).

SCIENTIFIC BACKGROUND AND TECHNICAL PROCESS

In version 3.3 of CLASlite, this simple decision tree is used to convert the single-image AutoMCU results to an estimate of forest cover.

Forest: $PV \geq 80$ AND $S < 20$
Non-forest: $PV < 80$ OR $S \geq 20$

where PV is photosynthetic vegetation cover fraction in the pixel, S is the bare substrate fraction in the pixel, and both thresholds can be customized, allowing you to tune your work to the current forest conditions. This S term is included to eliminate non-forest regrowth (successional vegetation, grasses, and some agriculture which may contain high PV fractions) from the forested class. These regrowth cover types usually have higher S levels than are found in neighboring intact forest.

This simple decision tree for forest cover, based on a default S setting of 20, is sufficiently general to allow the algorithm to accommodate a broad range of tropical forests. However, we recommend that you, the user, independently validate these maps. Based on validation, we encourage you to develop more accurate forest cover mapping in your area of interest by improving the forest cover criteria. This can be done by adjusting the Sval threshold value in CLASlite or by applying a user-modified decision tree to the fractional cover image outside of CLASlite. The power of CLASlite thus rests in quickly providing calibrated and corrected reflectance and fractional cover results, at which point you can make an informed modification to the forest cover mapping approach.

Forest Cover Batch Processing

We have added support for batch processing of forest cover images in addition to expanding the forest cover criteria. The user can now choose a value between 0 and 100 for both photosynthetic vegetation (PV) and Soil (S).

The ‘Map Forest Cover’ interface has changed to include Batch Process as shown in Figure 6. The user has to load the prepared ‘step3_template.csv’ template and Load the file to start the processing. The template includes two fields for PV and S values required for each image. The values will follow the same criteria as shown in Figure 6 (i.e. PV>= value and S < value). The default values presented in the template are 80 and 20 for PV and S, respectively.

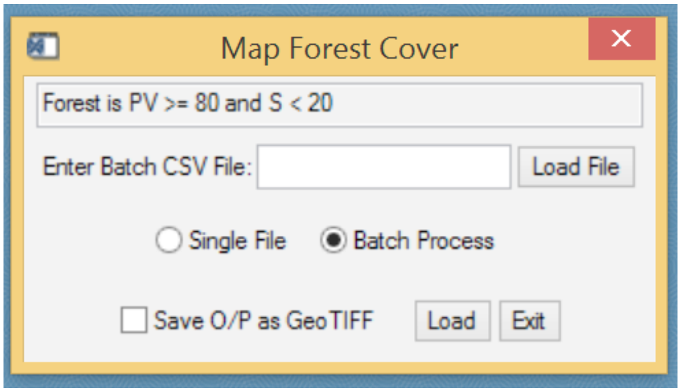


Figure 6 'Map Forest Cover' inteface

Output

The output of Step 3 in CLASlite is classified map of forest cover. The map contains three classes, defined below:

0 – Masked pixels
1 – Forest
2 – Non-forest

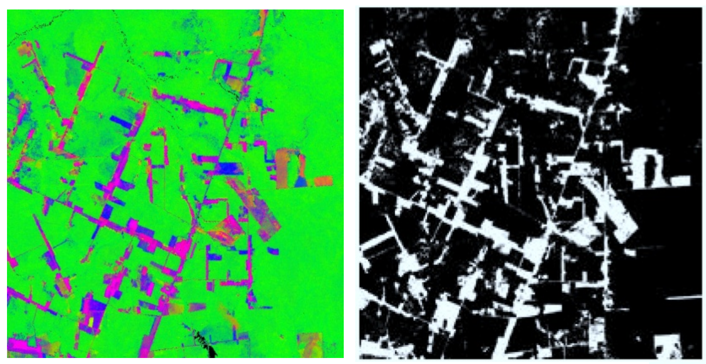


Figure 7: Fractional cover (left) and forest cover (right)

Forest cover maps can be ingested into third party geographic information systems (GIS) for the calculation of spatial statistics or conversion into visual and printable maps. While non-forest areas represented in the forest cover map may be the result of deforestation, it is important to note that neither deforestation nor disturbance can be mapped using a single satellite image. Deforestation and disturbance are forest *change*, and thus require multiple images for detection.

Step 4: Forest Change Detection

Overview

CLASlite includes the fully automated capability to detect *forest change* between a time series of images taken of the same geographic area over time. Multi-image analysis is the most accurate approach for detection of forest loss (deforestation), gain (secondary regrowth), or degradation (areas of persistent forest disturbance).

Scientific Background and Technical Process

To map forest change, CLASlite applies the following decision trees to each pair of images, where the subscripts 1 and 2 indicate images from one year to the next.

Where:

PV_1 = 1st Image photosynthetic vegetation fraction
NPV_1 = 1st Image non-photosynthetic vegetation fraction
S_1 = 1st image bare substrate fraction
$RMSE_1$ = 1st image RMSE

PV_2 = 2nd image photosynthetic vegetation fraction
NPV_2 = 2nd Image non-photosynthetic vegetation fraction
S_2 = 2nd image bare substrate fraction
$RMSE_2$ = 2nd image RMSE

$Refl_{1b1}$ = 1st image reflectance band 1
$Refl_{2b1}$ = 2nd image reflectance band 1
$Refl_{1b4}$ = 1st image reflectance band 4
$Refl_{2b4}$ = 2nd image reflectance band 4

a) Deforestation and disturbance pixels are calculated.

Deforestation:

$(((PV_1 – PV_2) \geq 25)$	[PV decrease captures most deforestation]
OR $((S_1 \leq 5)$ AND $((S_2 – S_1) \geq 15))$	[S increase captures deforestation followed by early regrowth]
OR $((PV_2 < 80)$ AND $((NPV_2 – NPV_1) \geq 20)))$	[NPV increase]

Forest Disturbance:

$((((NPV_2\text{-}NPV_1) \geq 10)$ AND $((PV_1\text{-}PV_2) > 10))$ OR $((S_1 \leq 5)$ AND $((S_2\text{-}S_1) > 10)$ AND $(S_2 \leq 15)))$

b) The following pixels are excluded from the forest change analysis [to eliminate the detection of false positives]**:**

For Landsat sensors:

<u>**For Deforestation:**</u>

((($PV_1 \le 0$) AND ($NPV_1 \le 0$) AND ($S_1 \le 0$))	[masked pixels (image 1)]
OR (($PV_2 \le 0$) AND ($NPV_2 \le 0$) AND ($S_2 \le 0$))	[masked pixels (image 2)]
OR (($PV_1 < 80$) OR ($S_1 \ge 15$))	[non-forest pixels (image 1)]
OR (($PV_1 = -1$)	[MASKED PIXELS IN IMAGE 1]
OR (($PV_2 = -1$)	[MASKED PIXELS IN IMAGE 2]
OR (($PV_1 \ge 80$) AND ($NPV_1 \ge 35$) AND ($RMSE_1 \ge 6$))	[unmasked cloud shadows and water (image 1)]
OR (($PV_2 \ge 80$) AND ($NPV_2 \ge 35$) AND ($RMSE_2 \ge 6$))	[unmasked cloud shadows and water (image 2)]
OR (($S_2 \ge 50$) AND ($S_2 < 100$) AND ($PV_2 > 0$))	[unmasked cloud rings (image 2)]
OR ((($NPV_2 – NPV_1$) < 10) AND (abs($Refl_{1b1} – Refl_{2b1}$)>300))	[unmasked cloud rings, cloud shadows, and topography shadows]

<u>**For Disturbance:**</u>

((($PV_1 \le 0$) AND ($NPV_1 \le 0$) AND ($S_1 \le 0$))	[masked pixels (image 1)]
OR (($PV_2 \le 0$) AND ($NPV_2 \le 0$) AND ($S_2 \le 0$))	[masked pixels (image 2)]
OR (($PV_1 < 80$) OR ($S_1 \ge 15$))	[non-forest pixels (image 1)]
OR (($PV_1 = -1$)	[MASKED PIXELS IN IMAGE 1]
OR (($PV_2 = -1$)	[MASKED PIXELS IN IMAGE 2]
OR (($PV_1 \ge 80$) AND ($NPV_1 \ge 35$) AND ($RMSE_1 \ge 6$))	[unmasked cloud shadows and water (image 1)]
OR (($PV_2 \ge 80$) AND ($NPV_2 \ge 35$) AND ($RMSE_2 \ge 6$))	[unmasked cloud shadows and water (image 2)]
OR (($S_2 \ge 50$) AND ($S_2 < 100$) AND ($PV_2 > 0$))	[unmasked cloud rings (image 2)]
OR ((($NPV_2 – NPV_1$) < 10) AND (abs($Refl_{1b1} – Refl_{2b1}$)>300) AND (abs($Refl_{1b4} – Refl_{2b4}$)<700) AND (abs($Refl_{1b4} – Refl_{2b4}$)>200))	[unmasked cloud rings, cloud shadows, and topography shadows]

For all other, non-Landsat, sensors (SPOT, ALI, and ASTER):

For Deforestation and Disturbance:

((($PV_1 \le 0$) AND ($NPV_1 \le 0$) AND ($S_1 \le 0$))	[masked pixels (image 1)]
OR (($PV_2 \le 0$) AND ($NPV_2 \le 0$) AND ($S_2 \le 0$))	[masked pixels (image 2)]
OR (($PV_1 < 80$) OR ($S_1 \ge 15$))	[non-forest pixels (image 1)]
OR (($PV_1 \ge 80$) AND ($NPV_1 \ge 35$) AND ($RMSE_1 \ge 6$))	[unmasked cloud shadows and water (image 1)]
OR (($PV_2 \ge 80$) AND ($NPV_2 \ge 35$) AND ($RMSE_2 \ge 6$))	[unmasked cloud shadows and water (image 2)]
OR (($S_2 \ge 50$) AND ($S_2 < 100$) AND ($PV_2 > 0$))	[unmasked cloud rings (image 2)]

Pixels that meet the above criteria are excluded from both the deforestation and disturbance images. User customization of artifact removal thresholds is currently only supported for Landsat imagery.

c) Isolated pixels for deforestation and disturbance undergo filtration.

A spatial filter (each deforestation pixel must be surrounded by a minimum of 5 deforestation pixels within a 3X3 pixel kernel) is applied to the raw deforestation result to remove isolated pixels. Pixels that meet the deforestation criteria but do not pass this filter are added to the group of pixels classified as disturbance. An additional spatial filter (each disturbance pixel must be surrounded by a minimum 5 disturbance pixels within a 7X7 pixel kernel) is applied to the raw disturbance result to remove isolated pixels while conserving detected patterns often associated with forest disturbance. These filters are conservative, in that we employ them in CLASlite to reduce your sensitivity to natural tree-fall events and spurious artifacts. As a result, you may under-estimate deforestation and disturbance in some instances. The disturbance filter is less conservative than the deforestation filter because disturbance is more likely to occur in isolated patches.

Artifact Removal Sliders for Landsat imagery

The forest change outputs may include unwanted artifacts (false positives) caused by the influence of clouds, unmasked cloud edges, cloud shadows, topography, and water boundaries. For Landsat imagery only, you can define desired slider settings for artifact removal in both the deforestation and disturbance images. These slider settings, which you can adjust by toggling the two artifact removal sliders in the step 4 window, determine the absolute difference of reflectance bands in the final exclusion decision tree criteria. For a given deforestation or disturbance output, a slider setting of 0 percent means "no removal of artifacts" and a slider setting of 100% means "removal of all possible artifacts." At both extremes, most artifacts are eliminated by the Landsat exclusion decision trees. The difference is that, at 100%, CLASlite eliminates all pixels it recognizes as possible false positives. These pixels are generally unmasked cloud rings, cloud shadows, and topography shadows that are mistaken for forest change. In contrast, at 0%, CLASlite does not eliminate any of these possible false positives.

The default slider values are 50% for deforestation and 25% for disturbance. These defaults are intended to remove noise as best as possible without removing real forest change, such as pixels of grass cover on previously deforested land.

The sliders work by adjusting threshold values in the exclusion decision trees as follows:

- For deforestation, (abs($Refl_{1b1}$ – $Refl_{2b1}$)>300)) corresponds to the default slider value of 50%. This 300 value can range from 500 at a 0% slider value to 0 at a 100% slider value. More pixels are excluded as the artifact removal percentage approaches 100%, due to the increasingly low threshold value.
- For disturbance, (abs($Refl_{1b1}$ – $Refl_{2b1}$)>300) AND (abs($Refl_{1b4}$ – $Refl_{2b4}$)<700) corresponds to the default slider value of 25%. As with deforestation, this 300 value can range from 500 to 0 as the slider value increases. The second threshold, 700, is constant for all slider settings in the 25-100% range. This 700 value decreases as the artifact removal percentage is lowered from 25% and approaches 0%, at which point the value is 300.

As you adjust the disturbance slider setting, the recommended deforestation slider setting will correspondingly increase or decrease. This relationship is due to a link between the process in step b) and c): because pixels determined by the deforestation filter to be artifacts are passed on to the disturbance filter to be considered as possible disturbance, a high deforestation slider value is generally needed to achieve a high disturbance slider value. This automatic adjustment is a recommendation only; you can and should ignore this suggestion if you know the study area well enough to estimate what percentage of artifact removal is appropriate.

For illustrated examples of the effectiveness and limitations of the sliders, see Appendix III.

Aggregation feature

By default, the deforestation and disturbance pixels undergo one final step in order to generate a deforestation map in which contiguous forest loss areas are clearly depicted. In this step, disturbance pixels in close proximity to a contiguous patch of deforestation pixels are moved from the disturbance output to the deforestation output. To achieve this, a buffer representing a radius of 120 meters is applied to the boundary of contiguous deforestation patches and all disturbance pixels falling within that radius is moved to the deforestation map. The outcome is that, for every deforestation patch, disturbance pixels associated with the same forest loss event will appear in the final deforestation output. The 120 meter buffer may be conservative in some places—that is, it may not capture all disturbance associated with a contiguous deforestation patch—so you should always check your results.

You can choose to map deforestation and disturbance without this Aggregation Feature by unselecting the “Aggregation Feature” box prior to running the change detection. You should recognize that turning off the aggregation feature discards some of the real-life spatial relationships between deforestation and nearby associated disturbance. For example, you might encounter a donut-shaped patch of deforestation with a couple disturbance pixels, in which only one tree was logged, embedded inside.

One situation in which we recommend that you turn the Aggregation Feature off is in areas prone to large-scale, natural disturbance. River corridors, for example, often undergo high levels of natural disturbance due to increases and decreases in river height as well as meandering of the riverbed. Other types of large-scale natural disturbance may be found in certain vegetation types, such as bamboo forests of the southwestern Amazon and Southeast Asia. In such cases, the Aggregation Feature will attempt to reclassify disturbances neighboring an actual deforestation event, thereby over-estimating the footprint of deforestation.

Multi-image analysis

If Step 4 is run with more than two images (up to 10), the results from individual time steps are compiled into a single map for each deforestation and forest disturbance for the entire time series according to one of the following methods, specified by you:

> **First change:** The first detected events of deforestation and disturbance within the time series are displayed in the resulting forest change maps.
>
> > For example, if a pixel meets the criteria of deforestation in both the 2nd time step and the 4th time step, the final map displays deforestation occurring in the 2nd time step.
>
> **Most recent change:** The most recent detected events of deforestation and disturbance within the time series are displayed in the resulting forest change maps.
>
> > For example, if a pixel meets the criteria of deforestation in both the 2nd time step and the 4th time step, the final map displays deforestation occurring in the 4th time step.

The resulting deforestation and disturbance maps are saved with _deforestation and _disturbance appended, respectively, to the base filename you define.

Batch processing

The deforestation and disturbance step supports batch processing, which enables you to process as many as 1000 images at a time. In contrast to calculating deforestation and disturbance on 2-10 input files, batch processing generates deforestation and disturbance results for each sequence of images. It also enables users to calculate deforestation and disturbance on many sets of images, in various locations simultaneously. Therefore you are not limited to processing a specific geographical location. Each line of inputs in the batch file produces a deforestation and disturbance image. This permits you to calculate maps based on different artifact removal thresholds for both deforestation and disturbance for the same geographical area, if you are unsure which threshold would be the best choice for that specific geographical location. The batch processing is initialized using a text file that contains 8 fields; each field should contain the following [from left to right]:

Fractional cover file year 1, path name and file name [input]
Fractional cover file year 2, path name and file name [input]
Reflectance file year 1, path name and file name [input]
Reflectance file year 2, path name and file name [input]
Percentage of deforestation artifact removal
Percentage of disturbance artifact removal
Deforestation and disturbance image, path name and file name [output]
Aggregation feature (on-“1” or off-“0”_)
GEOTIFF file (yes-“1” or no-“0”)

Note that batch processing is also supported for non-Landsat sensors, but requires a text file with only the following fields:

Fractional cover file year 1, path name and file name [input]
Fractional cover file year 2, path name and file name [input]
Deforestation and disturbance image, path name and file name [output]
Aggregation feature (on-“1” or off-“0”_)
GEOTIFF file (yes-“1” or no-“0”)

Advanced Mode

In CLASlite 3.3 the user can change all the deforestation and disturbance criteria values at their leisure for all satellite sensors available in CLASlite. Figure 8 shows the Map Forest Change interface that now includes an ‘Advanced Mode’ check box and ‘Set Criteria’ button in the upper right that is unchecked and inactive by default. The option is intended for advanced users who fully understand or are ready to understand CLASlite deforestation and disturbance criteria. Once the user checks the ‘Advanced Mode’ check box, the ‘Set Criteria’ button will be active for the user.

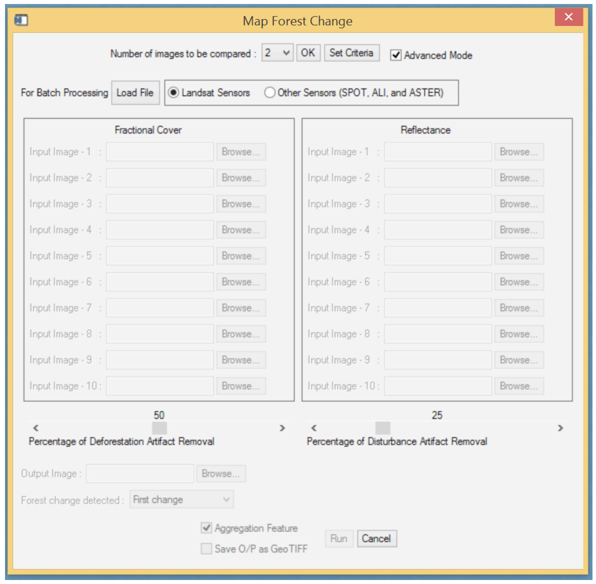


Figure 8 'Map Forest Change' inteface

When the user selects the 'Criteria Set' button the Set Criteria interface appears as shown in Figure 9. The Set Criteria interface includes five sections, Deforestation Criteria, Disturbance Criteria, Deforestation (Excluded Pixels), Disturbance (Excluded Pixels), and the Deforestation and disturbance filter. The five sections are explained in the 'STEP 4: FOREST CHANGE DETECTION: Scientific Background and Technical Process' section.

The five sections are all the processes that are included in Step 4 and are challenging values and parameters to change. The deforestation and disturbance criteria have taken researches years to create and improve, and we do not recommend that the user start from the beginning.
Figure 5 shows that all criteria values are empty and the deforestation and disturbance filter is unchecked (i.e. turned off). To avoid changing or adding all the values we have added the 'Load Parameters' and 'Save Parameters' buttons that allows the user to load previously saved values or save modified criteria values, respectively.

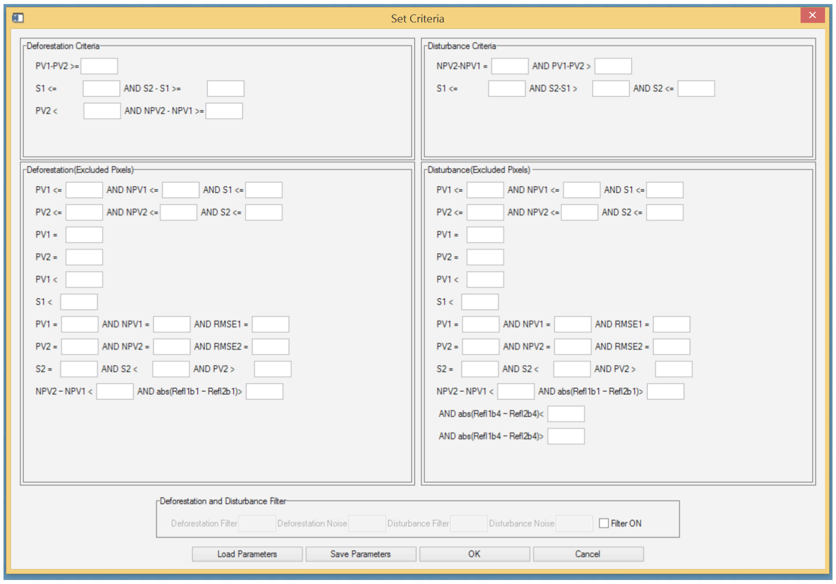


Figure 9 'Set Criteria' inteface – no values

Load Parameters: The button allows the user to upload previously set criteria values using a template. We have added the 'Step_4_Criteria.csv' template in the template folder for the CLASlite directory. Click the 'Load Parameters' button and choose the 'Step_4_Criteria.csv' template and instantly all CLASlite default values will be loaded as shown in Figure 10. The user can choose to change one or multiple values or even turn off the deforestation and disturbance filter. Once the user has completed making the required changes they can click on the 'OK' button. Now they can process one deforestation image or multiple images, or even create a compilation map for all satellite images supported by CLASlite.

Save Parameters: The button allows the user to save any changes made to the CLASlite deforestation and disturbance criteria into a template. The template can be then used again for further processing or shared with other researchers and groups.

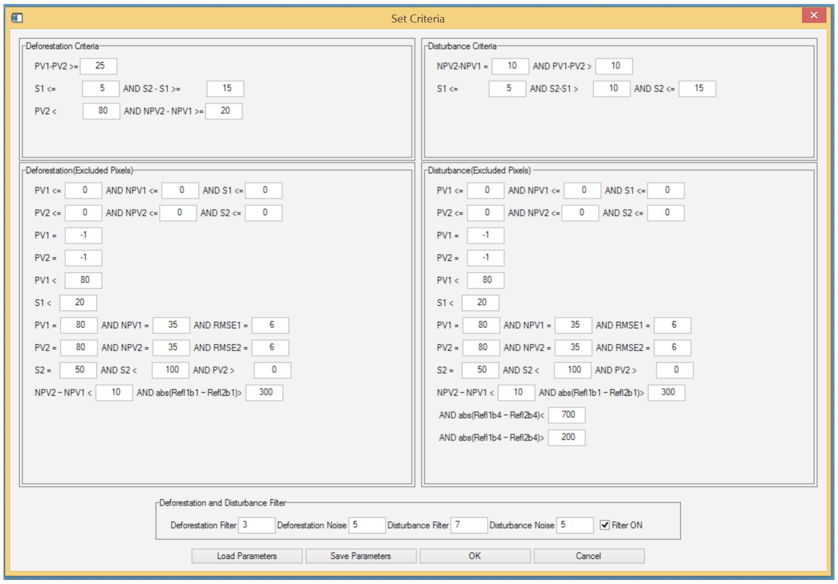


Figure 10 'Set Criteria' inteface – default criteria.

OUTPUT

The output of Step 4 in CLASlite is a pair of classified maps representing deforestation and forest disturbance. If more than two images are used, the output files represent deforestation for all intervals examined over the time series.

A legend file is saved alongside the _deforestation and _disturbance files, indicating the interval during which each change event was detected. Fig. 11 displays an example of this legend file for a 3-image analysis.

0 – No change detected
1 – Change from *image1.tif* to *image2.tif*
2 – Change from *image2.tif* to *image3.tif*

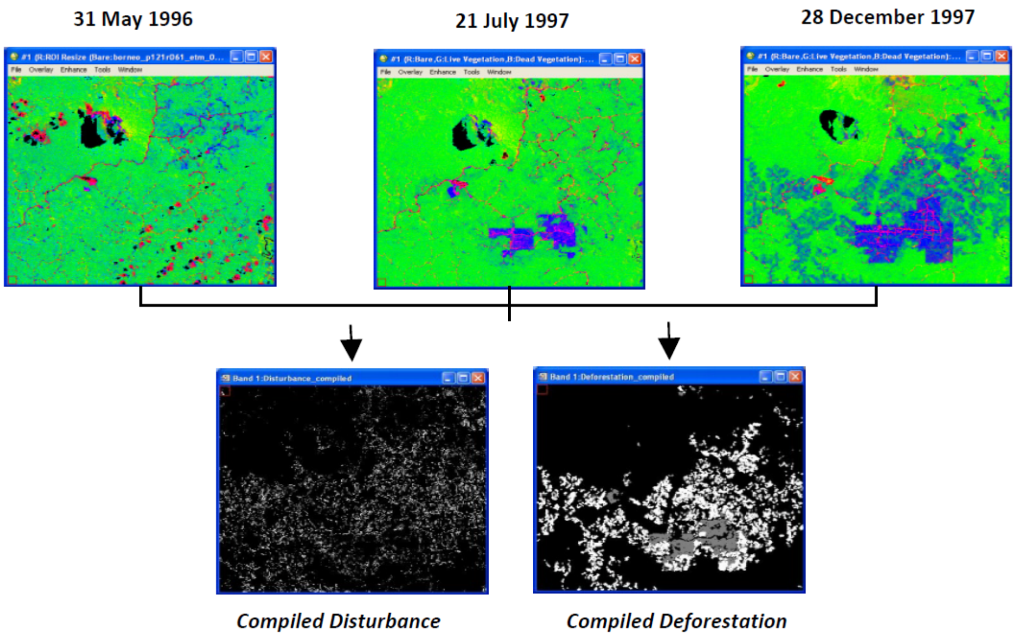


Figure 11: Forest change detection

POTENTIAL & LIMITS OF CLASLITE

The power of CLASlite rests in its unique ability to convert seemingly green “carpets” of dense tropical forest cover found in the basic satellite images into highly detailed maps that can be readily searched for deforestation, logging and other forest disturbance events.

CLASlite User Interaction

CLASlite does not provide a final “map”, but a set of ecologically meaningful images that accurately identify the amount of forest cover, deforestation and disturbance. Although CLASlite is a highly automated process, you need to become familiar with CLASlite output images. All CLASlite output images - from reflectance to fractional cover to forest change images – can be readily incorporated into digital maps via Geographic Information Systems (GIS) and other common mapping and spatial analysis software packages.

Interpretation of CLASlite outputs

CLASlite detects deforestation and forest disturbance as a change in fractional cover of PV, NPV, and S from one point in time to the next. This is a detection of physical change in forest structure and thus, the results do not explicitly indicate the cause of change. As a result, detected forest change can include both natural change (e.g. tree falls) and anthropogenic change (e.g. land conversion for agriculture). However, combined with local knowledge of the study area, maps of land cover change created with CLASlite can be used to understand spatial patterns of land use change. Both deforestation and secondary forest regrowth can be tracked by the CLASlite user. Deforestation is clearly shown as a loss of forest cover, producing bare substrate and NPV. Regrowth can be tracked by a careful account of forest recovery following clearing, which must be previously mapped.

CLASlite is not a tool for direct biodiversity monitoring. It can assist in reaching conclusions regarding biodiversity from forest presence or absence and disturbance, but it has not been designed for the purpose of direct biodiversity (species) monitoring.

Field verification of CLASlite output image is highly recommended if forest monitoring is conducted as basis for on-the-ground project development and execution.

Single- versus multi-image analysis

The capability for detecting disturbance or deforestation from a single image should be used with caution. Disturbance and deforestation are based on a change in condition from one time period to another. Although patterns of deforestation or disturbance can be inferred from a single image, a human interpretation of the results is necessary. Therefore anything detected as disturbance or deforestation from a single image should be used as a guide to further investigation and validation. Using a pair of images to detect disturbance and deforestation is better than relying on a single image.

In detecting forest change, the multi-image analysis should use images from the same time of the years considered, preferably from the same month. Otherwise changes in forest phenology might affect CLASlite’s capability for forest change detection.

Basis of decision trees

CLASlite's decision trees are calculated based on investigative research on hundreds of images throughout the Amazon basin and Andean mountains. Therefore, the decision tree thresholds may not be perfectly suitable for all regions on Earth.

USEFUL REFERENCES

SCIENTIFIC BASIS DOCUMENTS

Asner, G.P. 1998. Biophysical and biochemical sources of variability in canopy reflectance. Remote Sensing of Environment 64:234-253.

Asner, G.P. and D.B. Lobell. 2000. A biogeophysical approach for automated SWIR unmixing of soils and vegetation. Remote Sensing of Environment 74:99-112.

Asner, G.P. and K.B. Heidebrecht. 2002. Spectral unmixing of vegetation, soil and dry carbon in arid regions: Comparing multi-spectral and hyperspectral observations. International Journal of Remote Sensing 23:3,939-3,958.

Asner, G.P. and A.S. Warner. 2003. Canopy shadow in IKONOS satellite observations of tropical forests and savannas. Remote Sensing of Environment 87:521-533.

Asner, G.P., M. Keller, R. Pereira, J.C. Zweede, and J.N.M. Silva. 2004. Canopy damage and recovery following selective logging in an Amazon forest: Integrating field and satellite studies. Ecological Applications 14(4):280-298.

Asner, G.P. 2008. Hyperspectral remote sensing of canopy chemistry, physiology and diversity in tropical rainforests. Chapter 12 in Hyperspectral remote sensing of tropical and subtropical forests. M. Kalacska and G.A. Sanchez-Azofeifa (eds.) Taylor and Francis Group.

Asner, G.P., D.E. Knapp, A. Balaji, G. Páez-Acosta. 2009. Automated mapping of tropical deforestation and forest degradation: CLASlite. Journal of Applied Remote Sensing, vol. 3 (033543), 24pp. [doi:10.1117/1.3223675]

Zhu, Z. and C.E. Woodcock. 2012. Object-based cloud and cloud shadow detection in Landsat imagery, Remote Sensing of Environment doi:10.1016/j.rse.2011.10.028.

EXAMPLE APPLICATIONS

Asner, G.P., M.M.C. Bustamante, and A.R. Townsend. 2003. Scale dependence of biophysical structure in deforested lands bordering the Tapajós National Forest, Central Amazon. Remote Sensing of Environment 87:507-520.

Asner, G.P., D.E. Knapp, E.N. Broadbent, P.J.C. Oliveira, M. Keller, and J.N. Silva. 2005. Selective logging in the Brazilian Amazon. Science 310:480-482.

Asner, G.P., D.E. Knapp, A.N. Cooper, M.C.C. Bustamante, and L.O. Olander. 2005. Ecosystem structure throughout the Brazilian Amazon. Earth Interactions 9(7):1-31.

Elmore, A.J., G.P. Asner, and R.F. Hughes. 2005. Satellite monitoring of vegetation phenology and fire fuel conditions in Hawaiian drylands. Earth Interactions 9:1-21.

Asner, G.P., E.N. Broadbent, P.J.C. Oliveira, D.E. Knapp, M. Keller, and J.N. Silva. 2006. Condition and fate of logged forests in the Brazilian Amazon. Proceedings of the National Academy of Sciences 103(34):12947-12950.

Broadbent, E.N., D.J. Zarin, G.P. Asner, M. Pena-Claros, A. Cooper, and R. Littell. 2006. Forest structure and spectral properties after selective logging in Bolivia. Ecological Applications 16:1148-1163.

Oliveira, P.J., G.P. Asner, D.E. Knapp, A. Almeyda, R. Galvan-Gildemeister, S. Keene, R.F. Raybin, and R.C. Smith. 2007. Land-use allocation protects the Peruvian Amazon. Science 317:1233-1236.

Davidson, E.A., G.P. Asner, T.A. Stone, C. Neill, and R. de O. Figueiredo. 2008. Objective indicators of pasture degradation from spectral mixture analysis of Landsat imagery. Journal of Geophysical Research 133, G00B03, doi: 10.1029/2007JG000622.

Broadbent, E.N., G.P. Asner, P.J.C. Oliveira, D.E. Knapp, M. Keller, and J.N. Silva. 2008. Forest fragmentation from deforestation and selective logging in the Brazilian Amazon. Biological Conservation 141:1745-1757.

Asner, G.P., T.K. Rudel, M. Aide, R. Defries, and R. Emerson. 2009. A contemporary assessment of change in humid tropical forests. Conservation Biology 23:1386-1395.

Almeyda, A.M., E.N. Broadbent, M. Schmink, S.G. Perz, and G.P. Asner. 2010. Deforestation drivers in Southwest Amazonia: comparing smallholder farmers in Inapari, Peru and Assis Brasil, Brazil. Conservation and Society 8:157-170.

Asner, G.P., G.V.N. Powell, J. Mascaro, et al. 2010. High-resolution carbon stocks and emissions in the Amazon. Proceedings of the National Academy of Sciences 107:16738-16742.

IDEAM. 2011. Capacidad Institucional Tecnica Cientifica para apoyar Proyectos REDD: Reduccion de Emisiones por Deforestación en Colombia. Instituto de Hidrologia, Meteorologia y Estudios Ambientales, Bogotá.

Asner, G.P., R.F. Hughes, J. Mascaro, A.L. Uowolo, D.E. Knapp, J. Jacobson, T. Kennedy-Bowdoin, and J.K. Clark. 2011. High-resolution carbon mapping on the million-hectare Island of Hawaii. Frontiers in Ecology and Environment 9:434-439.

Alencar, A., G.P. Asner, D.E. Knapp, and D. Zarin. 2011. Temporal variability of forest fires in the eastern Amazon. Ecological Applications 21:2397-2412.

Carlson, K.M., L.M. Curran, D. Ratnasari, A.M. Pittman, B.S. Soares-Filho, G.P. Asner, S.N. Trigg, D.A. Gaveau, D. Lawrence, and H.O. Rodrigues. 2012. Committed carbon emissions, deforestation, and community land conversation from oil palm plantation expansion in West Kalimantan, Indonesia. Proceedings of the National Academy of Sciences 109:7559-7564.

Reimer, F., J. Borner, and S. Wunder. 2012. Monitoring deforestation for REDD: an overview for options for the Juma Sustainable Development Reserve Project. CIFOR Report. June 2012

Asner, G.P., J.K. Clark, J. Mascaro, G.A. Galindo García, K.D. Chadwick, D.A. Navarrete Encinales, G. Paez-Acosta, E. Cabrera Montenegro, T. Kennedy-Bowdoin, Á. Duque, A. Balaji, P. von Hildebrand, L. Maatoug, J.F. Phillips Bernal, A.P. Yepes Quintero, D.E. Knapp, M.C. García Dávila, J. Jacobson, and M.F. Ordóñez. 2012. High-resolution mapping of forest carbon stocks in the Colombian Amazon. Biogeosciences 9:2683-2696, doi:10.5194/bg-9-2683-2012.

Carlson, K.M., L.M. Curran, G.P. Asner, A. McDonald Pittman, S.N. Trigg, and J.M. Adeney. 2012. Carbon emissions from forest conversion by Kalimantan oil palm plantations. Nature Climate Change doi:10.1038/nclimate1702.

MINAM. 2012. Memoria Técnica de la Cuantificación de Cobertura de Bosque y Deforestación 2000-2011. Ministry of Environment, Peru.

Bryan, J.E., P.L. Shearman, G.P. Asner, D.E. Knapp, G. Aoro, and B. Lokes. 2013. Extreme differences in forest degradation in Borneo: Comparing practices in Sarawak, Sabah, and Brunei. PLOS ONE 8(7):e69679.

APPENDIX I: NON-DISCLOSURE AGREEMENT

24 April 2020
This Non-Disclosure Agreement supersedes previous agreements.

CLAS and CLASlite

i-Cultiver, Inc. (“**i-Cultiver**”, “**CLASlite**”), Carnegie Institution for Science (“**Carnegie**”) and Dr. Gregory P. Asner (“**Dr. Asner**”) hereby (collectively, “**we**,” or “**us**” or “**our**”) are working together to offer Carnegie Landsat Analysis System (“CLAS”) and the Carnegie Landsat Analysis System-lite (“CLASlite”) for forest mapping. CLAS, CLASlite, its subroutines including AutoMCU, and all derivatives are protected under U.S. Patents 8189877, 20090214084 and 20120288159-A1; International Classification G06K9/62. CLASlite has been exclusively licensed to **i-Cultiver, Inc.**

It is the intention of Carnegie and i-Cultiver to work with other parties in the use of CLASlite for the purpose of conducting environmental studies and ecosystem monitoring. It is recognized that such use may require the disclosure by “**us**” of certain information (“Proprietary Information”) to any such party (“**User**”). Proprietary Information includes, but is not limited to, all information pertaining to the basis, background, development or composition of CLAS, CLASlite or any sub-routine. Proprietary Information also includes all source code, executable programs, sample data, manuals or other written documentation. The purpose of this non-disclosure agreement (“**NDA**”) is to protect “**our**” Proprietary Information and to ensure the proper handling and publication of data and information that is developed in studies involving the use of CLAS or CLASlite.

Accordingly, by using CLAS or CLASlite, the User agrees to the following:

1. The CLAS, CLASlite or any sub-routine software, including all source code, executable programs, and supporting digital files and libraries, are the sole property of Carnegie and the licensee (i-Cultiver, Inc.), and shall not be reproduced by the User or provided to any party without written approval. The software module embedded within CLAS and CLASlite, known as the Automated Monte Carlo Unmixing (“AutoMCU”) code, remains the sole property of **Dr. Gregory P. Asner** (“**Dr. Asner**”). Any effort to use CLAS, CLASlite or AutoMCU in an unauthorized way or to reproduce any portion of the CLAS, CLASlite or AutoMCU software is prohibited, and will result in immediate termination of the User’s right to use CLAS, CLASlite or AutoMCU and the taking of any other legal or civil action deemed appropriate.

2. “**We**” make no warranty as to the quality and/or accuracy of any data obtained by the User pursuant to any CLAS or CLASlite study, or the suitability of any such data for any purpose. The User assumes all risks and liabilities in the collection, interpretation and use of any data or results obtained using the CLAS or CLASlite program. The User will indemnify Carnegie and i-Cultiver (us) against any claims or damages resulting from the User’s use of CLAS or CLASlite as provided herein.

3. Upon Carnegie’s or i-Cultiver’s request, the User will return all copies of any Proprietary Information and CLAS/CLASlite Materials which may have been provided to, or used by, the User in connection with its performance of any CLAS or CLASlite studies.

APPENDIX II: IMAGE PREPARATION FOR CLASLITE

Before processing data through CLASlite, you must prepare the data for input to the software. The data supplied to CLASlite must be in the correct format and have the correct number of bands. The expected formats and characteristics of the data vary by sensor. Common steps for preparing data from any satellite are:

1. Geo-referencing the image to a UTM projection (WGS-84) ellipsoid
2. Resampling spectral bands to the same spatial resolution (pixel size), if necessary
3. Reordering bands, if necessary
4. Saving the image to GeoTIFF or ENVI format

The data requirements for each satellite sensor are listed in this section. You can use almost any image processing package to prepare your image, though ENVI and ERDAS are popular options.

1. Landsat 8 OLI/TIRS

Landsat 8 provides two thermal bands instead of one, and a new band for Quality Assessment (QA). As a result, preparing the image files for Step 1 of CLASlite is a more involved process. We have resolved this issue by creating a tool, called "Prepare Landsat," that will do the necessary stacking for you. You simply have to browse and select the directory containing the .TIF files for your image. This folder should not contain any other image files, and the name of your files should begin with "LC8."

Once you've selected a folder, press "Stack" and CLASlite will prepare a _raw file, a _therm file_, a _QA file, and a _MTL.txt file containing metadata. You will need to ingest these outputs in Step 1. The _raw, _therm, and _QA files will be in ENVI format.

If you want to run this process for more than one image at the same time, you will need to paste the directory names into column A (each directory in a separate cell) of the "tools_prepare_landsat.csv" template, which you can find in the "templates" folder of your CLASlite directory. To run, just check "Batch Process" instead of "Single File," then press "Stack".

2. Landsat Thematic Mapper (TM) 4, 5, and Enhanced Thematic Mapper+ (ETM+) 7

CLASlite has the capability to process images from the Landsat 4, 5 and 7 satellites. The image data must be in two files. One file must contain the data for Landsat bands 1-5 and band 7 (ordered from lowest to highest). If you also have the thermal band for masking clouds (band 6, high gain), it must be in a separate file, and must cover the exact same area and have the same pixel size as the other file. Therefore, it may be necessary to resample the thermal band from its original resolution to the pixel size of the other bands, prior to use with CLASlite. When you resample the imagery, use the nearest neighbor resampling kernel. The pixel values in all Landsat imagery used in CLASlite must be 1-byte values and must not have any atmospheric corrections applied to them.

NOTE that, alternatively, you may use the "Prepare Landsat" tool – described above – to prepare raw .TIF files for Landsat 4, 5, and 7. Just be sure that Landsat 7 files begin with "L7" or "LE7," Landsat 5 files begin with "L5" or "LT5," and Landsat 4 files begin with "L4" or "LT4." For these three satellites, the outputs are a _raw file, a _therm file, and a _MTL.txt file only. There is no _QA file because no QA band exists for Landsat 4, 5, or 7.

3. Earth Observing-1 (EO-1) Advanced Land Imager (ALI)

CLASlite can process images from the Earth Observing-1 satellite, which carries the Advanced Land Imager (ALI) sensor. Only ALI level-1G data are supported in CLASlite.

ALI does not have a thermal band. Only the 9 visible, near-infrared and shortwave-infrared bands must be contained in one file with the original 16-bit integer values for each pixel. The bands should be ordered from lowest to highest.

4. Advanced Spaceborne Thermal Emission & Reflection Radiometer (ASTER)

CLASlite can process ASTER Level-1B imagery acquired by the NASA Terra satellite. ASTER images come in different resolutions. The visible/near-infrared (VNIR) imagery has 15-meter pixels, while the shortwave-infrared (SWIR) imagery has 30-meter pixels. Since these bands come in different resolutions, it is recommended that all of the bands be resampled using the nearest neighbor kernel to the lowest (30-meter) resolution of the 9 VNIR and SWIR bands. The thermal bands of ASTER have 90-meter pixels, but they are not used in CLASlite.

One potential problem with ASTER data is that the VNIR and SWIR images are collected from two different telescopes, making it possible for the two sets of bands to be misaligned. Misalignment is more likely to occur in areas with large variations in elevation. When misalignment is a problem, the best thing to do is to geo-reference the images separately, then combine them using image-to-image registration.

CLASlite requires that the radiance conversion coefficients be applied beforehand. The image processing software may take care of this step during resampling, but if not, you will need to do so manually.

5. Système Pour l'Observation de la Terre (SPOT) 4 and 5

The CAP and DIMAP formats are supported for SPOT-4 (HRVIR) and 5 (HRG) in CLASlite. The 4 spectral bands should be organized into a single GeoTIFF or ENVI file with a 20-meter pixel size.

In order for CLASlite to convert the image to reflectance, it will need to read the LEADXX.DAT file for CAP formatted images and the .DIM or .XML for DIMAP formatted images. This file should be included on the original media on which the SPOT image was received and contains gain parameters that are needed to convert the image to radiance and reflectance.

As in the case of Landsat data, the input file should be 1-byte per pixel.

	Landsat TM & ETM+	ALI	ASTER	SPOT 4 (HRVIR) and 5 (HRG)
Thermal band used?	Yes	No	No	No
Number of spectral bands used	6	9	9	4
Band order	1,2,3,4,5,7	MS-1', MS-1, MS-2, MS-3, MS-4, MS-4', MS-5', MS-5, MS-7	1,2,3N,4,5,6,7,8,9	1,2,3,4
Input data type	8-bit	16-bit short integer	32-bit floating point	8-bit
Processing level of data supported	n/a	Level 1G	Level 1B	CAP or DIPAM

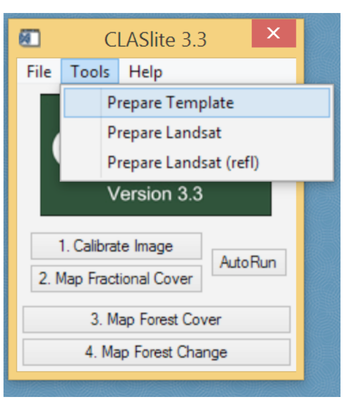
Figure 12: Required data format for imagery from CLASlite-compatible sensors

AUTOMATIC PREPARATION OF CLASLITE LANDSAT IMAGERY

TEMPLATE PREPARATION

This new tool provides an automated step to create the user templates for Landsat images and Landsat reflectance images. With the new 64-bit mode, we think that our users will be running batch images much more often, so we have added a tool that will allow our users to create the tools_prepare_landsat.csv, step1_template.csv, step2_template.csv, and 'step3_template.csv' templates. We did not create a Step-4 template because we cannot assume what two years the user would like to run. Thus we have left the step4_template_ls.csv file to be created by the user.

The 'Prepare Template' tool can be accessed under the Tools menu as shown in Fig. 13. The interface gives the user the ability to choose between 4 options, explained below:



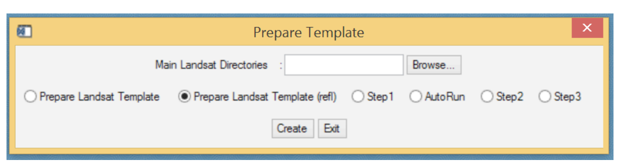


Figure 13: 'Prepare Template' interface

- Prepare Landsat Template: prepares the tools_prepare_landsat.csv file.
- Prepare Landsat Template (refl): prepares the tools_prepare_landsat_refl.csv file.
- Step 1: prepares the step1_template.csv file with output files as reflectance.
- AutoRun: prepares the step1_template.csv file with output files as fractional cover.
- Step 2: prepares the step2_template.csv file.
- Step 3: prepares the step3_template.csv file.

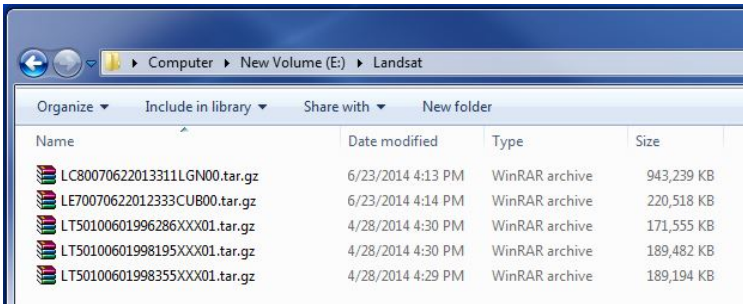
We would now like to direct the user to the correct method of using the 'Prepare Template' tool effectively and without any problems. The code depends on our naming system, so the user must use the following file naming system:

Raw image should end with '_raw'. Example: L8167080_20130719_raw
Thermal image should end with '_therm'. Example: L8167080_20130719_therm
QA image should end with '_QA'. Example: Example: L8167080_20130719_QA
Reflectance image should end with '_refl'. Example: L8167080_20130719_refl
Fractional cover image should end with '_frac'. Example: L8167080_20130719_frac

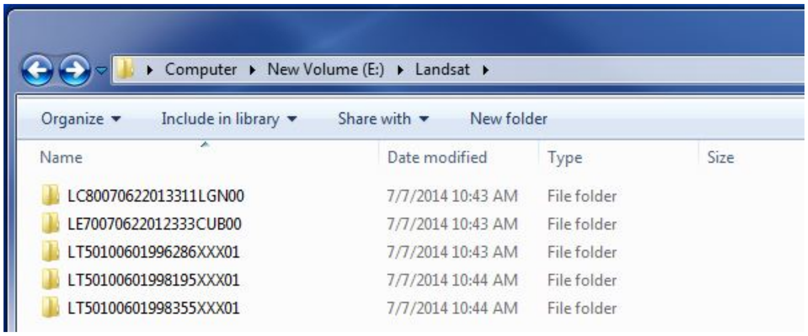
Method

This example assumes that the user has 5 Landsat images that have been downloaded and are required for processing.

- Create a working folder called 'Landsat' that contains the five zipped images downloaded from USGS [See image below].



- Unzip the five images each to their separate folders and remove the original zip files so that the working folder 'Landsat' only has the image folders [See image below].



- Use the ‘Prepare Template’ tool to create the tools_prepare_landsat.csv file, which is used in the ‘Prepare Landsat’ step.
- Use the ‘Prepare Landsat’ step in conjunction with the tools_prepare_landsat.csv file created in the previous step. To create the _raw, _therm, and _QA [Landsat 8] files.
- All the templates are automatically saved in the ‘C:\CLASlite\templates’ folder. The code will overwrite the existing .csv template, so please make a copy of previous templates if needed.
- Now that we have the input files ready, use ‘Prepare Template’ for step 1. It is important to mention that the user cannot run this step unless the _raw, _therm, and _QA [Landsat 8] files already exist, or the code won’t be able to create the templates. The user should choose step1 if they want to only run the calibration step, and AutoRun, if they want to run the AutoRun Fractional Cover [both step1 and step 2]
- The user can skip Step 1 and download Landsat CDR Images. The user can then use the Use the ‘Prepare Template (refl)’ tool to create the tools_prepare_landsat_refl.csv file, which is used in the ‘Prepare Landsat (refl)’ step.
- Use the ‘Prepare Landsat (refl)’ step in conjunction with the tools_prepare_landsat_refl.csv file created in the previous step. To create the _refl files.
- To run ‘Step 2’, the user should have created the reflectance files up to this point. And the names of reflectance outputs for all the images should have ‘_refl’ at the end.
- To run the last option ‘Step 3’, the user should have created the fractional cover files up to this point. And the names of fractional cover outputs for all the images should have ‘_frac’ at the end.

PREPARING A LANDSAT IMAGE

The Prepare Landsat tool is used to prepare Landsat images only for Step 1 (Calibration to reflectance)

Steps to prepare a Landsat image:

- First we unzip the Landsat image, and we verify that our folder contains only files for that image, as shown below in Fig. 14.

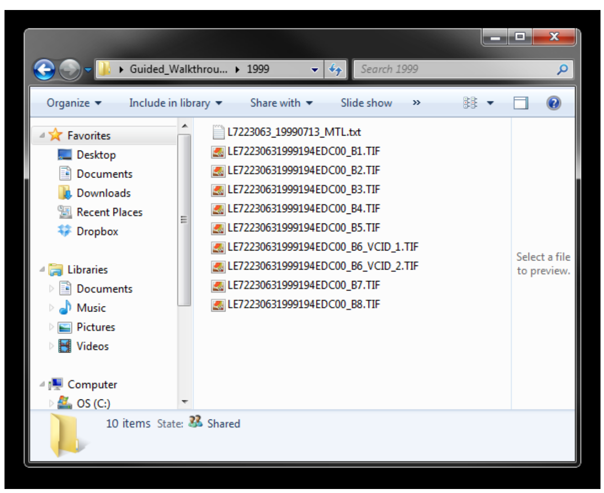


Figure 14: Example of a downloaded Landsat image after unzipping the file

- Navigate to and select the “Prepare Landsat” tool as shown in Fig. 15.

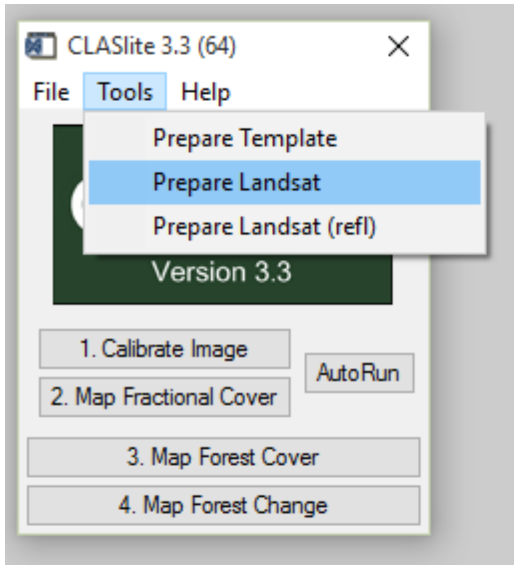


Figure 15: Example of a downloaded Landsat image after unzipping the file

- Search for the folder that contains the .TIF folders for your image as shown below in Fig. 16.

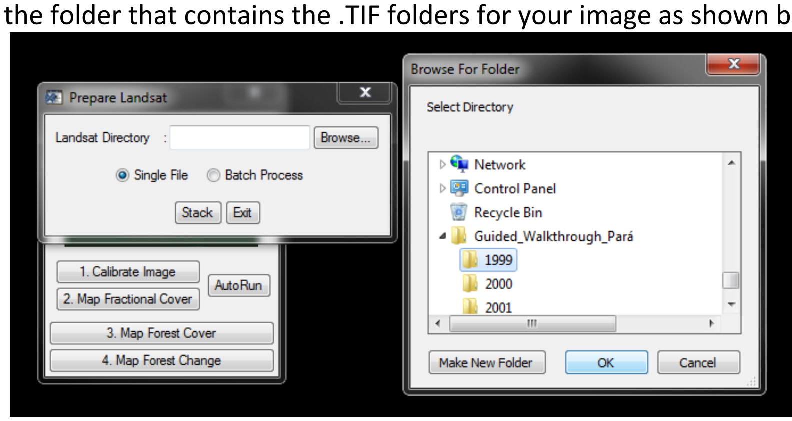


Figure 16: Navigating to the location of the unzipped Landsat .TIF files

- Click “Stack”.
- As described in the previous section, if the user wants to prepare multiple images at a time, you should enter the paths for each scene’s folder (which should contain only the files for that image) into “tools_prepare_landsat.csv” in C:\claslite\templates. Then click “Batch Process” and load the file. You can automatically create the “tools_prepare_landsat.csv” template by using the ‘Prepare Template’ tool.

PREPARING A LANDSAT REFLECTANCE IMAGE

The Prepare Landsat tool is used to prepare Landsat images only for Step 1 (Calibration to reflectance)

Steps to prepare a Landsat image:

- First we unzip the Landsat reflectance image, and we verify that our folder contains only files for that image, as shown below in Fig. 17.

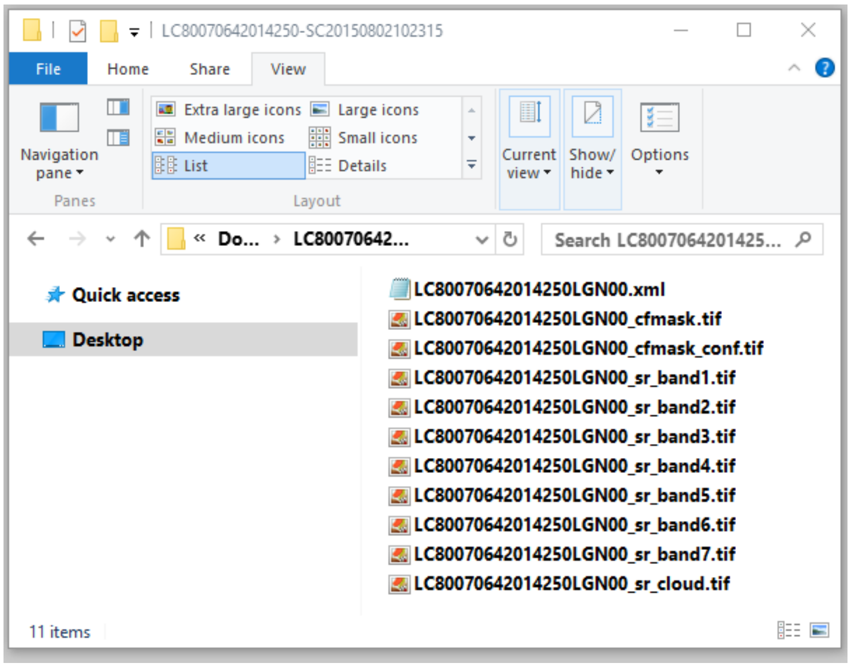


Figure 17: Example of a downloaded Landsat image after unzipping the file

- Navigate to and select the “Prepare Landsat (refl)” tool as shown in Fig. 18.

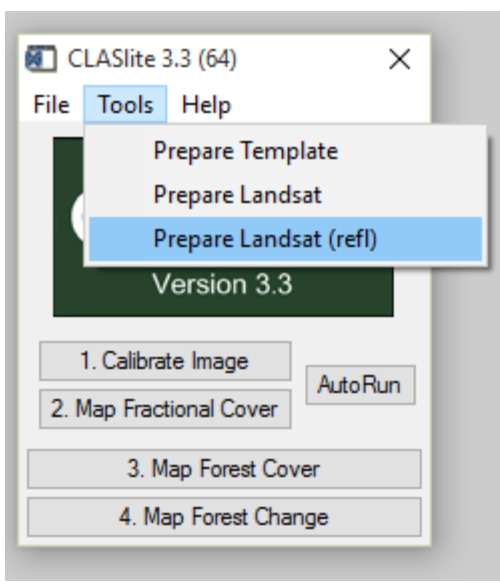


Figure 18: Example of a downloaded Landsat reflectance image after unzipping the file

- Search for the folder that contains the .TIF folders for your image as shown below in Fig. 19.

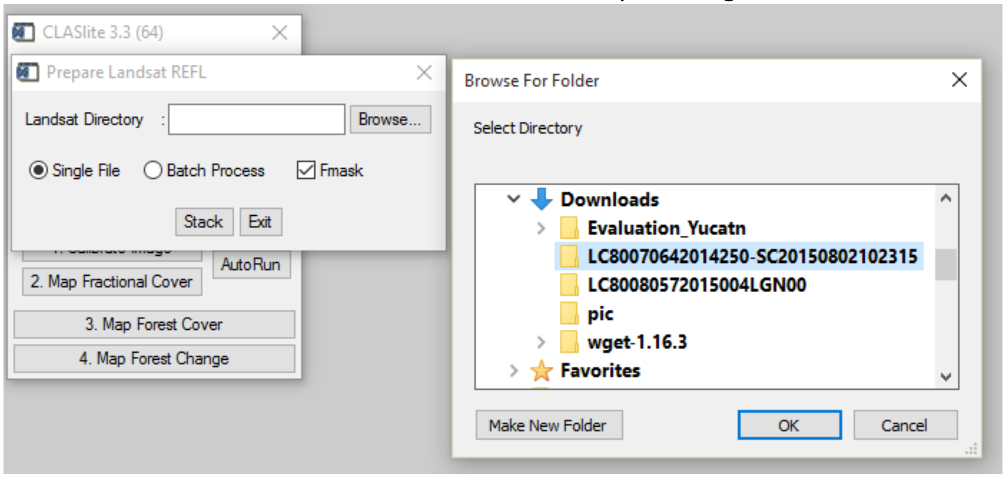


Figure 19: Navigating to the location of the unzipped Landsat .TIF files

- Click “Stack”.
- As described in the previous section, if the user wants to prepare multiple images at a time, you should enter the paths for each scene’s folder (which should contain only the files for that image) into “tools_prepare_landsat_refl.csv” in C:\claslite\templates. Then click “Batch Process” and load the file. You can automatically create the “tools_prepare_landsat_refl.csv” template by using the ‘Prepare Template (refl)’ tool.
- Fmask: This check box works for both single and batch processing, it allows the user to turn on or off cloud masking. The only cloud masking tool available here is Fmask, if the user requires CLASlite cloud masking criteria, then step 1 must be used. CLASlite cloud masking criteria requires the radiance image that is unavailable for the Prepare Landsat Reflectance tool.

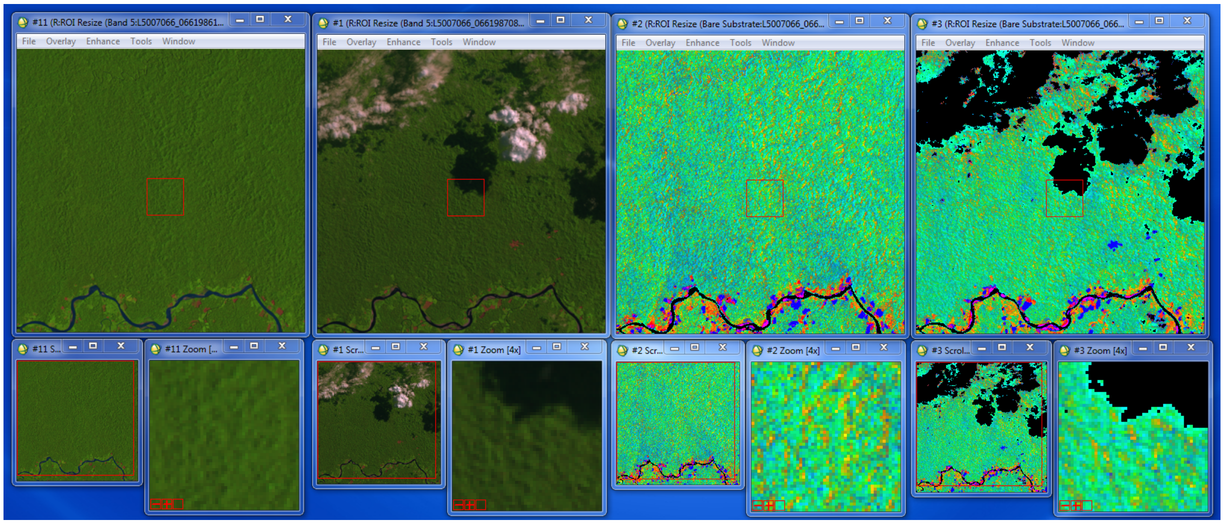
APPENDIX III: ARTIFACT REMOVAL USING SLIDERS

CLASlite v3.1’s artifact removal sliders were designed to improve the quality of forest change detection and to enhance your flexibility to customize the software’s algorithms depending on your knowledge of the land cover in an area. The drawback is that sliders can remove real deforestation and disturbance pixels, or preserve false positives, if not used mindfully.

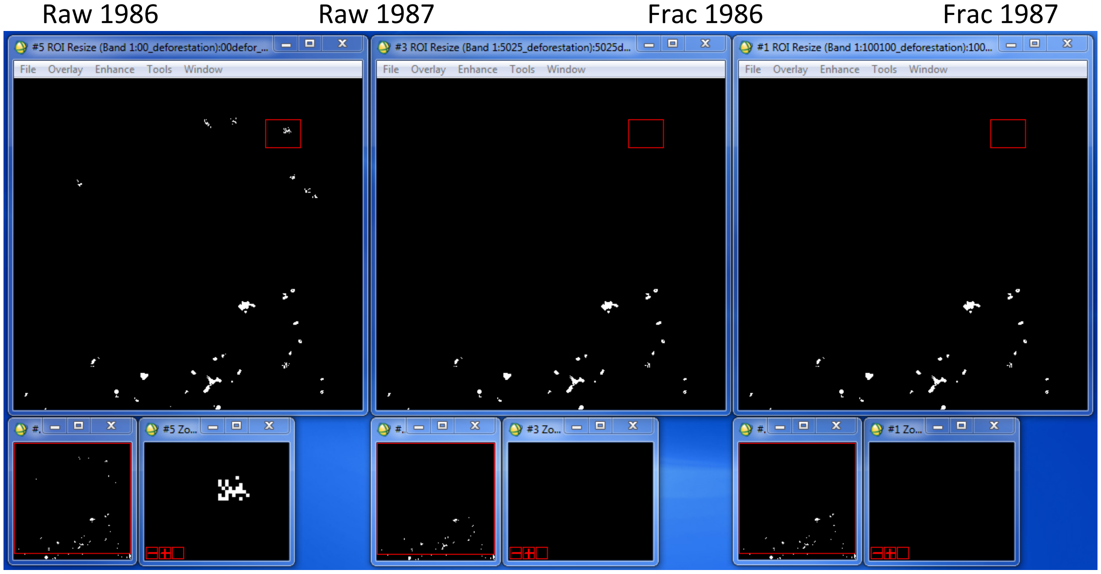
Here, we illustrate examples of where the sliders can work well and where they have shortcomings. There is no prescription slider setting for eliminating a given type of artifact, but the following are common outcomes of using the sliders. Note that the Aggregation Feature was turned on for these examples.

Example 1 depicts a scene that contains both real deforestation and disturbance near a river and artifacts due to clouds. The example illustrates that sliders, even when set to 100%, do not remove real deforestation and disturbance pixels. The deforestation slider removes all artifacts when set to 50%, but even at 100% some disturbance artifacts remain.

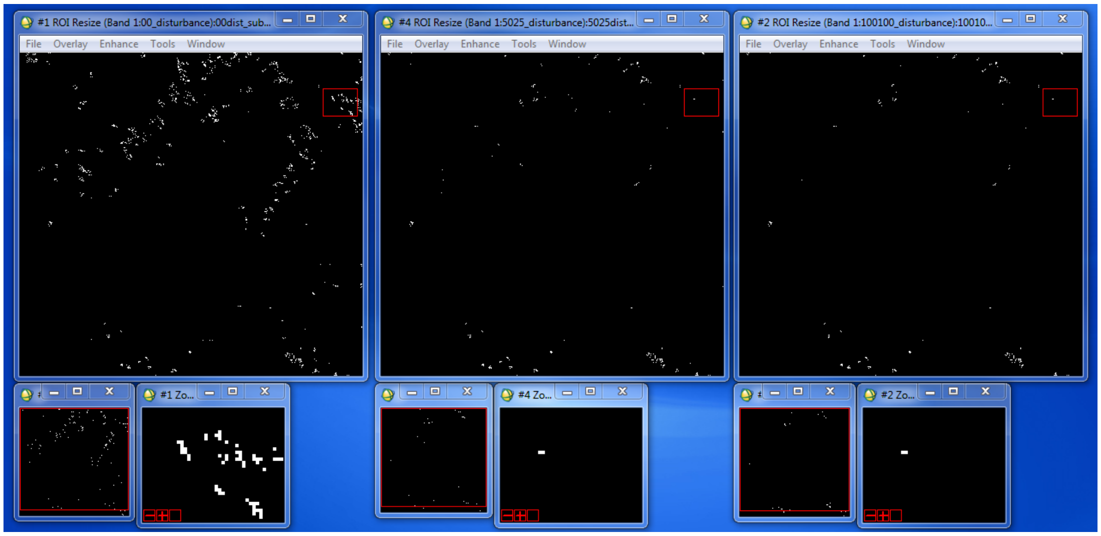
Example 2 illustrates that real deforestation and disturbance pixels in the form of grass cover on cleared land are at risk of being eliminated by slider values exceeding default settings. This problem can also arise if deforestation is kept at the default 50% value while disturbance is raised to a higher value.



Raw 1986 Raw 1987 Frac 1986 Frac 1987

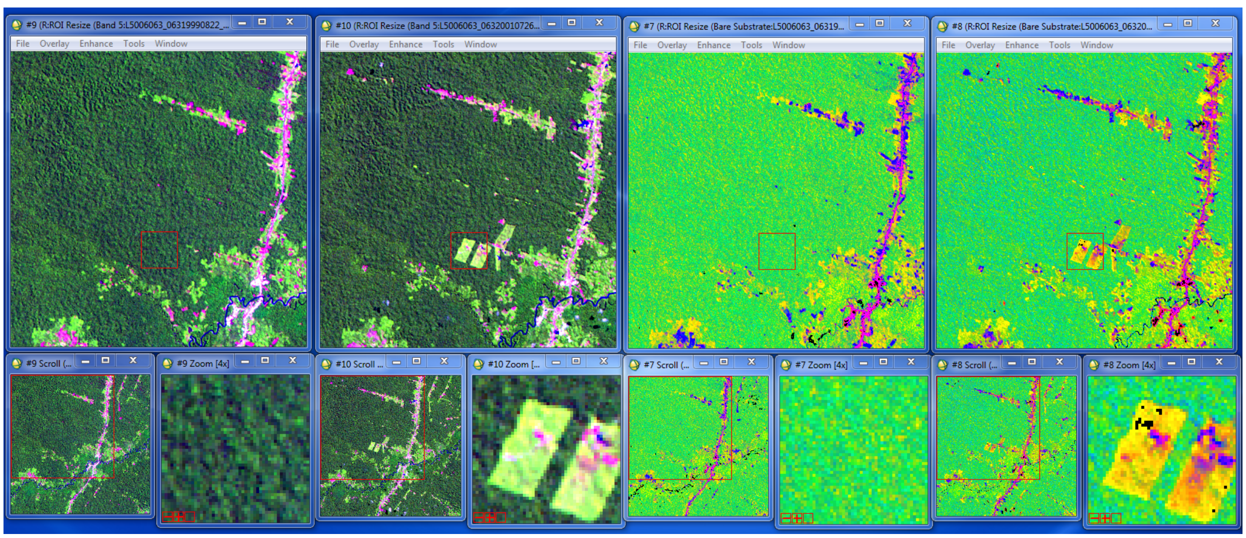


Defor 0% Defor 50% Defor 100%



Dist 0% Dist 25% Dist 100%

Figure 20: Sliders example



Raw 1999 Raw 2001 Frac 1999 Frac 2001

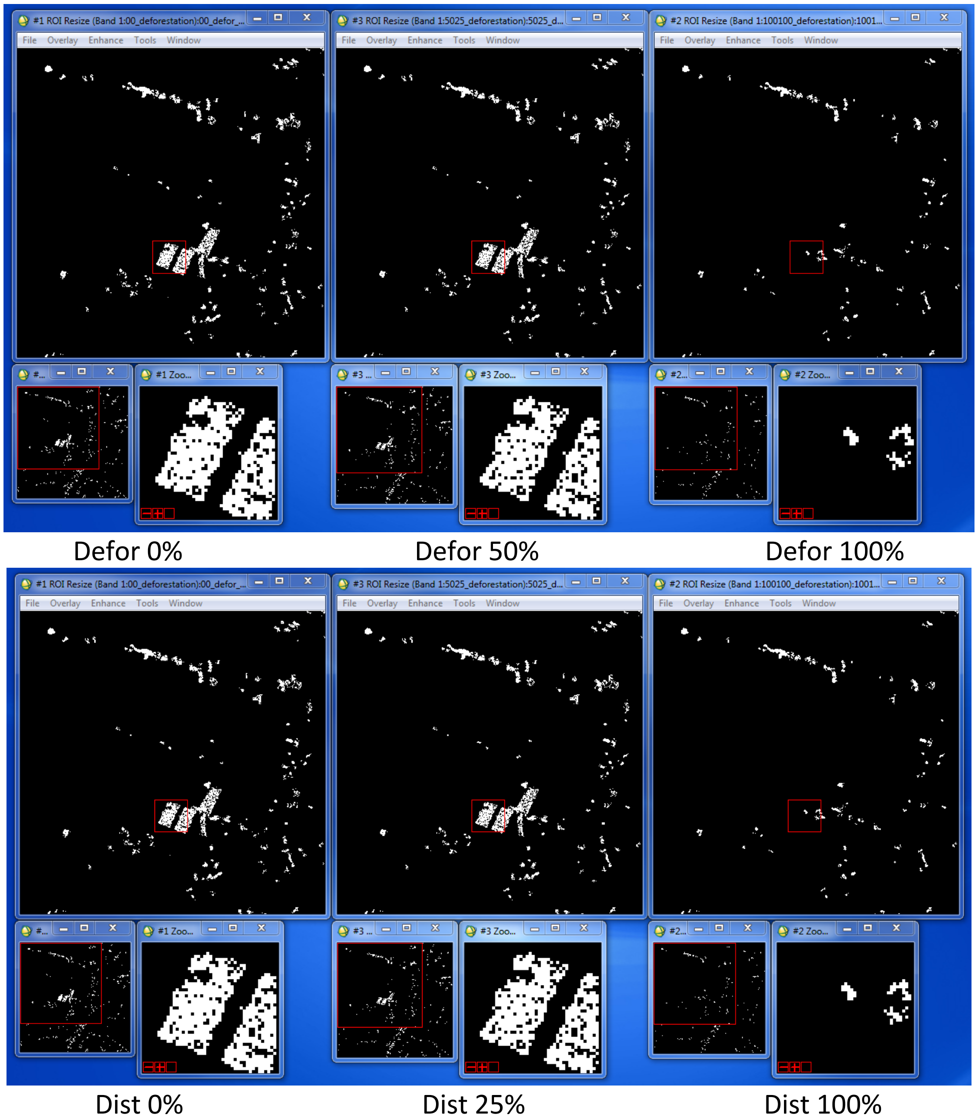


Figure 21: Sliders example 2

APPENDIX IV: SENTINEL-2 SUPPORT

Sentinel-2 reflectance processing can be performed for single and batch image processing. The satellite can be chosen from the satellite drop-down menu in Step 2, shown in Figure 22

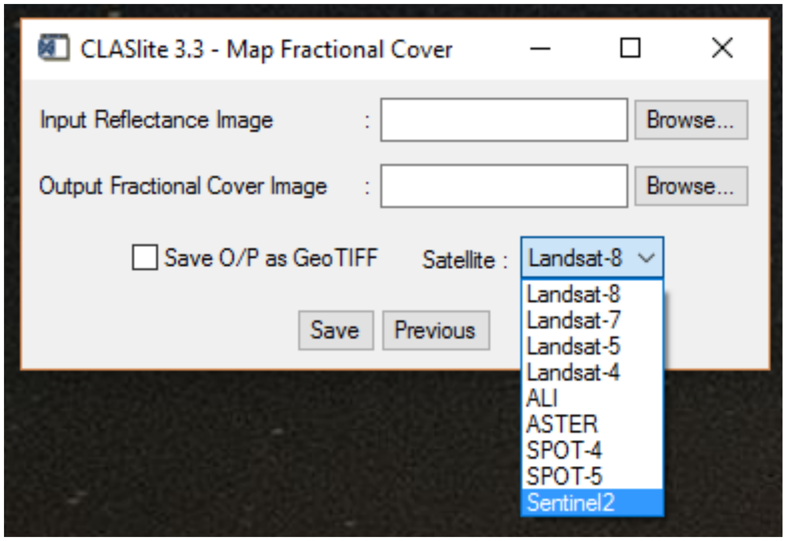


Figure 22: Sentinel-2 Fractional Cover processing in Step 2

In order to prepare the reflectance image, Sentinel-2 must be processed using external tools that are separate from CLASlite. We have prepared a description that will help the user prepare reflectance images below.

Preparing a Sentinel-2 Reflectance image

With the introduction of CLASlite version 3.3, data from the Sentinel-2 MSI sensor can be used in CLASlite. The surface reflectance product from Sentinel-2, which can be prepared for use in AutoMCU, is generated by using software supplied by the European Space Agency (ESA). CLASlite does not create a surface reflectance image from Sentinel-2. Instructions for creating an L2A (surface reflectance) product and preparing it for use in CLASlite are described below.

Getting Sentinel-2 MSI data

Data from the Sentinel-2 MSI sensor can be searched and downloaded here. Please sign-up for an account to download data. The level of data most common is the Level 1C (top-of-atmosphere), which can also be processed to Level 2A. The data are typically distributed as a zip file, with the imagery broken into tiles.

Instructions for Processing data to Level 2A

Please visit this page to get information about the sen2cor software to process Sentinel-2 MSI L1C imagery to L2A. Install and configure the sen2cor software and then process the data to surface reflectance as shown in the following example. The sen2cor software only works for 64-bit Windows, Linux, and Mac operating systems. Please read the release notes carefully before installation, especially section 4 describing installation and setup. For any questions about the installation or usage of Sen2Cor, please follow the STEP forum area dedicated to Sen2Cor that can be accessed here.

Processing L1C to L2A

In this example, an L1C product was downloaded and unzipped in a directory. If the sen2cor software has been properly installed and configured, an L2A product can be generated. Windows command line (CMD) must be used to convert L1C to L2A. Please refer to the following website for different methods to launch CMD in Windows. The screenshot in Figure 23 displays the CMD window with the usage information for the "L2Process" command, which is used to generate the L2A product.

```
Command Prompt
C:\Users\sinan>L2A_process --help
usage: L2A_Process [-h] [--resolution {10,20,60}] [--sc_only] [--cr_only]
                   [--refresh] [--GIP_L2A GIP_L2A] [--GIP_L2A_SC GIP_L2A_SC]
                   [--GIP_L2A_AC GIP_L2A_AC]
                   directory

Sentinel-2 Level 2A Prototype Processor (Sen2Cor). Version: 2.2.1, created:
2016.04.29, supporting Level-1C product version: 13.1.

positional arguments:
  directory             Directory where the Level-1C input files are located

optional arguments:
  -h, --help            show this help message and exit
  --resolution {10,20,60}
                        Target resolution, can be 10, 20 or 60m. If omitted,
                        all resolutions will be processed
  --sc_only             Performs only the scene classification at 60 or 20m
                        resolution
  --cr_only             Performs only the creation of the L2A product tree, no
                        processing
  --refresh             Performs a refresh of the persistent configuration
                        before start
  --GIP_L2A GIP_L2A     Select the user GIPP
  --GIP_L2A_SC GIP_L2A_SC
                        Select the scene classification GIPP
  --GIP_L2A_AC GIP_L2A_AC
                        Select the atmospheric correction GIPP

C:\Users\sinan>
```

Figure 23: L2Process help command displayed in CMD window

The product name for the example input image is:
S2A_OPER_PRD_MSIL1C_PDMC_20160808T014659_R024_V20160807T135257_20160807T135257.zip

When this zip file is extracted, it will create the following directory:
S2A_OPER_PRD_MSIL1C_PDMC_20160808T014659_R024_V20160807T135257_20160807T135257.SAFE

Under this directory is a subdirectory called GRANULES, which contains subdirectories of the various tiles. Information on the tile naming convention (shown in red below) are given at this website. Also, a KML of the whole tile system is available.

```
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MBA_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MBV_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MCA_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MCB_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MCV_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MDA_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MDB_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MDV_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MEA_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MEB_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MEV_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MFA_N02.04
S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MFB_N02.04
```

The following command example shows how to start the processing of the L2A product for the 22MDV tile:

```
L2A_Process
S2A_OPER_PRD_MSIL1C_PDMC_20160808T014659_R024_V20160807T135257_20160807T135257.
SAFE/GRANULE/S2A_OPER_MSI_L1C_TL_MTI__20160807T201612_A005884_T22MDV_N02.04
--resolution=20
```

If you want to process all tiles, the command string would simply include the main directory:

```
L2A_Process
S2A_OPER_PRD_MSIL1C_PDMC_20160808T014659_R024_V20160807T135257_20160807T135257.
SAFE --resolution=20
```

Preparation of Reflectance Image

When the reflectance image is generated by sen2cor, it is written out in a different directory. In this example, the output directory is:
S2A_USER_PRD_MSIL1C_PDMC_20160808T014659_R024_V20160807T135257_20160807T135257.SAFE

This product needs to be written out in ENVI format for use in the AutoMCU process of CLASlite. The best way to prepare these data is with the Sentinel Application Platform (SNAP). This is another software product available from ESA. It can be downloaded here. From this application, open the L2A product that you generated with sen2cor. After the product is opened, go to the Raster->Geometric Operations->Resampling tool menu. The resampling window will appear as shown in Figure 24(a).

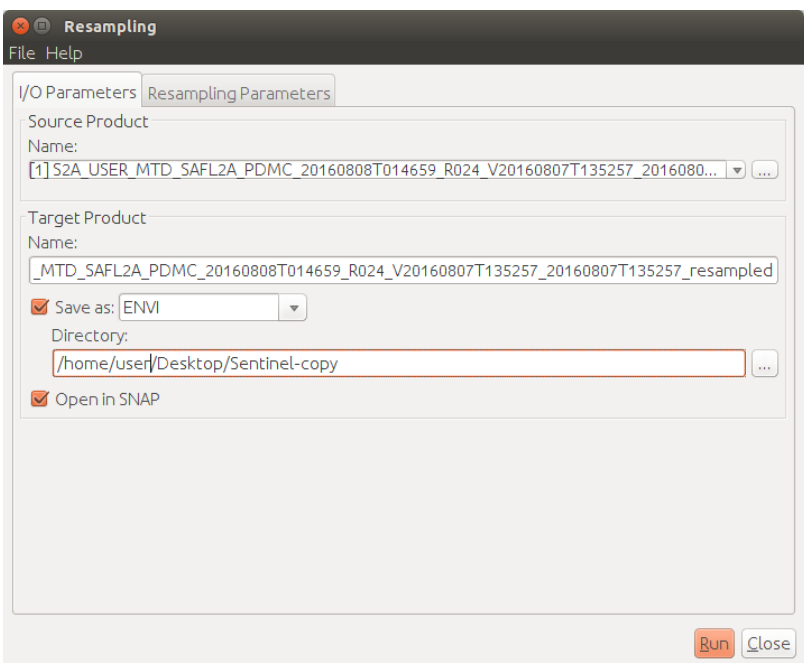


Figure 24(a): Resampling window in SNAP program (I/O Parameters tab)

Select the opened L2A product as the source product and provide a name for the new resampled target product. Save the output in ENVI format. Click on the Resampling Parameters tab. Check the “By pixel resolution” radio button and select 20 (meters). Also, please make sure that you have selected and unselected all the parameters as shown in Figure 24(b) below. Run the resampling operation.

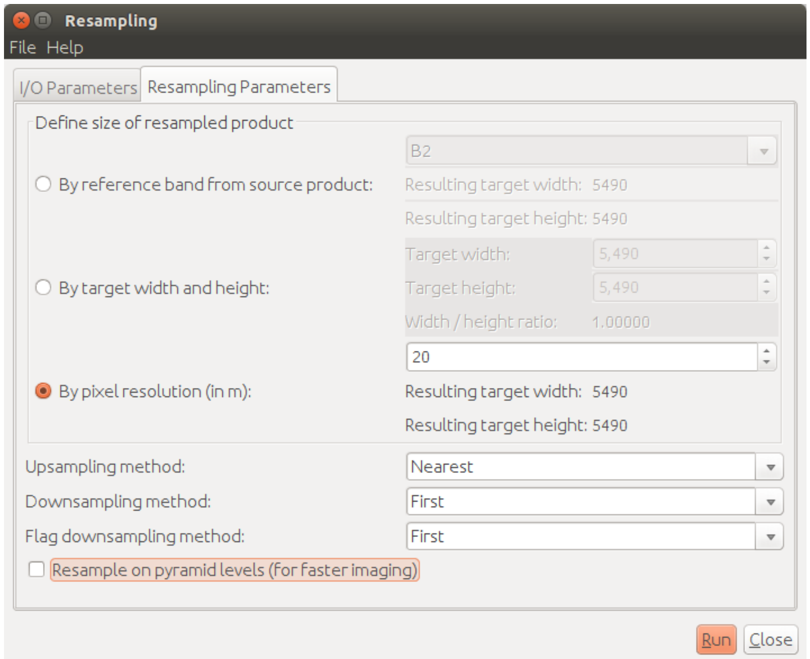


Figure 24(b): Reesampling window in sen2cor program (Resampling Parameters tab)

When the resampling is done, the ENVI files will be written as separate bands in the directory specified as the target product. The next step is to put the 9 bands of 20-meter data into a single ENVI file. These 9 bands are B2, B3, B4, B5, B6, B7, B8A, B11, and B12. This can be done in ENVI by using the New File Builder to put the 9 files together (in numerical order). It can also be done with the GDAL command gdal_merge.py. The following command string shows how to generate the multi-band ENVI image for use in CLASlite.

```
python gdal_merge.py -o sentinel2_reflectance -of ENVI -separate -co
INTERLEAVE=BIL B2.img B3.img B4.img B5.img B6.img B7.img B8A.img B11.img
B12.img
```

Any manual masking of this input image should be done so that the masked pixels have a value of -9999. If masking is not necessary, then the image generate from gdal_merge.py should be ready to use in CLASlite.

Appendix V: Google Earth Engine Support

CLASlite supports Google Earth Engine Landsat images similar to Landsat Level 1 data products available from USGS archive from Earth Explorer (earthexplorer.usgs.gov). After downloading the images from Google Earth Engine, no additional processing is required. The user only needs to provide Landsat images in the same naming conventions available from Earth Explorer. The naming convention for Landsat Level 1 products for different satellites are shown in Figure 25. The XXXXXXXXX characters can be any preferable indications, such as image path/row or date. However, the characters before and after XXXXXXXXX must be exactly as shown in Figure 25. For example, Landsat 8 must start with 'LC8' and end with B1 through BQA.

Landsat 8 (12 bands)	Landsat 7 (9 bands)	Landsat 5 (7 bands)
LC8XXXXXXXXX_B1.TIF	LE7XXXXXXXXX_B1.TIF	LT5XXXXXXXXX_B1.TIF
LC8XXXXXXXXX_B2.TIF	LE7XXXXXXXXX_B2.TIF	LT5XXXXXXXXX_B2.TI
LC8XXXXXXXXX_B3.TIF	LE7XXXXXXXXX_B3.TIF	LT5XXXXXXXXX_B3.TI
LC8XXXXXXXXX_B4.TIF	LE7XXXXXXXXX_B4.TIF	LT5XXXXXXXXX_B4.TI
LC8XXXXXXXXX_B5.TIF	LE7XXXXXXXXX_B5.TIF	LT5XXXXXXXXX_B5.TI
LC8XXXXXXXXX_B6.TIF	LE7XXXXXXXXX_B6_VCID_1.TIF	LT5XXXXXXXXX_B6.TI
LC8XXXXXXXXX_B7.TIF	LE7XXXXXXXXX_B6_VCID_2.TIF	LT5XXXXXXXXX_B7.TI
LC8XXXXXXXXX_B8.TIF	LE7XXXXXXXXX_B7.TIF	
LC8XXXXXXXXX_B9.TIF	LE7XXXXXXXXX_B8.TIF	
LC8XXXXXXXXX_B10.TIF		
LC8XXXXXXXXX_B11.TIF		
LC8XXXXXXXXX_BQA.TIF		

Figure 24: Required naming conventions for Google earth Landsat images

The Google Earth Engine website can be accessed from here. The user must first sign-up for an account then learn Google Earth Engine code. Extensive documentation for Google Earth Engine code can be found here. The user should go through the guides, references, and tutorials provided by Google. The user can change the image names, described in Figure 25, in the Google Earth code, or after downloading the images. We present here a Google Earth code example for downloading Landsat 8, as shown in Figure 26. The purpose of the example is not to teach the user how to write code with Google Earth, rather how to add the required naming conventions directly in the code.

```
var geometry = /* color: ff0000 */ee.Geometry.Polygon(
    [[[113.37890625, 3.574827886692079],
      [116.7626953125, 3.8949463767329],
      [119.3994140625, 3.7634047800344397],
      [120.8056640625, 4.464713550531585],
      [119.7509765625, 7.305805890302946],
      [113.203125, 7.218620330234878]]]);

var sabah = geometry;

// Compute a cloud score.  This expects the input image to have
// STD band names: ["red", "blue", etc], so it can work across sensors.
var cloudScore = function(img) {
  // A helper to apply an expression and linearly rescale the output.
  var rescale = function(img, exp, thresholds) {
    return img.expression(exp, {img: img})
        .subtract(thresholds[0]).divide(thresholds[1] - thresholds[0]);
  };

  var score = ee.Image(1.0);
  // Clouds are reasonably bright in the blue band.
```

```
  score = score.min(rescale(img, 'img.blue', [0.1, 0.3]));

  // Clouds are reasonably bright in all visible bands.
  score = score.min(rescale(img, 'img.red + img.green + img.blue', [0.2, 0.8]));

  // Clouds are reasonably bright in all infrared bands.
  score = score.min(rescale(img, 'img.nir + img.swir1 + img.swir2', [0.3, 0.8]));

  // Clouds are reasonably cool in temperature.
  score = score.min(rescale(img, 'img.temp', [300, 290]));

  // extra RGB filter
  score = score.max(rescale(img, 'img.red + img.green + img.blue', [0.2, 0.8]));

  //note, snow filter ignored!

  return score;
};

function addShadowSum(img){
   return
img.addBands(img.select(shadowSumBands).reduce(ee.Reducer.sum()).select([0],['shadowSum']));
  }

var shadowScore = function(img) {
  var score;
  if (year > 2012) {
    score = cloudScore(img.select(LC8_BANDS,STD8_NAMES));
  }
  else if (year == 2012) {
    score = cloudScore(img.select(LC7_BANDS,STD7_NAMES));
  }
  else {
    score = cloudScore(img.select(LC5_BANDS, STD5_NAMES));
  }
  score = img.reduce(ee.Reducer.sum()).select([0],['shadowSum']);
  return score;
};

var LC5_BANDS = ['B1','B2',   'B3',    'B4',  'B5',  'B6',    'B7'];
var STD5_NAMES = ['blue', 'green', 'red', 'nir', 'swir1', 'temp', 'swir2'];
var LC7_BANDS = ['B1','B2',   'B3',    'B4',  'B5',  'B6_VCID_1','B6_VCID_2', 'B7', 'B8'];
var STD7_NAMES = ['blue', 'green', 'red', 'nir', 'swir1', 'temp_lr','temp', 'swir2','pan'];
var LC8_BANDS = ['B1','B2',   'B3',    'B4',  'B5',  'B6',    'B7',    'B10', 'BQA'];
var STD8_NAMES = ['aerosol','blue', 'green', 'red', 'nir', 'swir1', 'swir2', 'temp','qa'];

var shadowSumNames = ['B4', 'B5', 'B7'];
var shadowSumBands = ['nir','swir1','swir2'];//Bands for shadow masking
```

```
// Load current data - make sure it falls within Landsat 8 time ranges!
var year = 2012;
var start_date = '-04-01';
var end_date = '-08-15';


var landsat = ee.ImageCollection('LANDSAT/LC8_L1T_8DAY_RAW').filterDate(year.toString() +
start_date,year.toString() + end_date)
.filterBounds(sabah);
var toa = ee.ImageCollection('LANDSAT/LC8_L1T_8DAY_TOA').filterDate(year.toString() +
start_date,year.toString() + end_date)
.filterBounds(sabah);


print(landsat);


toa = toa.map(function(img) {
    // Invert the cloudscore so 1 is least cloudy, and rename the band.
    var score = cloudScore(img.select(LC8_BANDS,STD8_NAMES));
    var shadow_score = shadowScore(img.select(shadowSumNames, shadowSumBands));
    score = ee.Image(1).subtract(score).add(shadow_score).select([0], ['cloudscore']);
    return img.addBands(score);
  });

landsat = landsat.combine(toa.select(['cloudscore']));
landsat = landsat.qualityMosaic('cloudscore');



var jsonCoordString = sabah.toGeoJSON();
print(jsonCoordString);
var vizParams = {'bands': ['B6', 'B5', 'B4']};

Map.addLayer(landsat.clip(sabah),vizParams);

for (var i=0; i < 12; i+=1) {

  var tempnum = i+1;
   var tempname;
// Start adding naming convention, ex: LC8Landsat_B1.TIF
    var satname = 'LC8';
    if (tempnum < 12) {
      tempname =satname + 'LANDSAT_' + year.toString() + '_B' + tempnum.toString();
    }
    else {
      tempname =satname + 'LANDSAT_' + year.toString() + '_BQA' ;
```

```
    }


  Export.image.toDrive({image: landsat.select(i),
                    description: tempname,
                    fileNamePrefix: tempname,
                    folder: 'landsat_cloudfilter',
                    scale: 30,
                    maxPixels: 10000000000,
                    region: jsonCoordString,
                    crs: 'EPSG:32650'
  });

}
```

Figure 26: Google Earth Engine code example for Landsat 8