**ArcGIS Pro Shoreline Digitization Tool**

**Robert Elliott, MS in GIS**

**Acknowledgements:**

**Dr. Yichun Xie (Eastern Michigan Univ.)**

**USGS at Woods Hole**

**Table of Contents**

I. Introduction

II. Methods  
 III. Data Sources

IV. Tool Usage Instructions

i. Accessing the Tool in ArcGIS Pro

ii. Running the Tool

V. Tool Parameter Descriptions

VI. Interpreting Tool Output

VII. Tool Usage Considerations, Limitations, and Troubleshooting

VIII. Incorporating Tide and Beach Slope

IX. References

X. Appendix: Using the Shoreline Digitization Tool with Single-Band Imagery

**I. Introduction:**

Rapid and accurate shoreline digitization using remote imagery can greatly facilitate coastal change detection and water surface area estimation, both of which have important implications to human society and to the natural world. From sea level rise exposure risk assessment to rapid disease vector risk (e.g. from water pooling) to natural habitat conservation, it is important to be able to observe and quantify how water body extent (i.e. shoreline) changes over time. Fully automating shoreline digitization allows large areas of shoreline features to be created with significantly less time, expense, and subjectivity than digitizing shorelines manually or delineating shorelines through supervised image classification (or any number of more complex machine learning algorithms). The ArcGIS Pro "script tool" format was chosen because script tools are relatively simple to share and use by coastal professionals and organizations. The script tool is based on the Python language but does not require the end user to know how to code. Rather, the end user will only have to specify a handful of parameters, one of them being the file location of a pre-written Python file (.py). For those with Python experience, the code is thoroughly commented to explain the underlying logic and to allow for customization by the end user. The script tool was created using ArcPro version 2.7 and requires at a minimum the Spatial Analyst extension, though the Image Analyst extension adds substantial functionality. The tool should be compatible with all later versions of ArcPro but may not be compatible with earlier versions. Though powerful, the tool may not be able to automate shoreline digitization 100% of the time. Visual interpretation by an analyst is required to confirm the accuracy of shoreline digitization, and in some cases manual shoreline digitization may still be necessary for at least part of an image. The tool's polygon outputs facilitate water surface area estimation, and the tool's DEM-based contour output can help the user compare the digitized shoreline output to ground reference water levels (e.g. tides) from the time of multi-band imagery collection. This script tool has incorporated feedback from Dr. Yichun Xie and from USGS scientists at Woods Hole. It has also drawn insights from existing shoreline tools such as CoastSat (Vos, 2022). Though this tool is ever-evolving, it can already offer coastal professionals new capabilities in shoreline digitization and great improvements in user-friendliness.

**II. Shoreline Script Tool Methods:**

The shoreline digitization script tool is optimized to analyze high resolution multi-band imagery. The tool's shoreline detection algorithms use bands that are common and inexpensive to collect (Red, Green, Blue, Near Infrared) and also allow for the more expensive but highly useful shortwave infrared (SWIR) band. Using the shoreline digitization tool does not require any specific knowledge of the algorithms used. If the imagery bands required for a given algorithm are provided by the user, then the corresponding algorithm will be used to create output. The tool user's task will simply be to visually select the appropriate shoreline output that most accurately delineated the shoreline. The shoreline detection algorithms used are as follows (in no particular order): i) Binary (Otsu) thresholded (M)NDWI, ii) Unsupervised classification using ISO clusters (machine learning), ii) Band Ratio (("Blue" > "NIR") & ("Blue" > "Red")), iii) Otsu thresholded (M)NDWI plus Band Ratio (("Blue" > "NIR") & ("Blue" > "Red")), iv) Otsu thresholded (M)NDWI plus other Band Ratio (("Blue" > "SWIR") & ("Blue" > "Red")), and v) User-input (M)NDWI binary thresholds.

Combining NDWI with a band ratio approach yields an output that may increase RGBNIR shoreline detection in areas with shoreline impervious surfaces (e.g. houses, concrete). In general, NDWI identifies as water any pixels with a value greater than zero (McFeeters, 1996). However, the classification sometimes works better if the threshold (cut-off) is set to a value other than 0 (say, 0.3). The only algorithm that does not require a NIR or SWIR band is unsupervised classification. Since unsupervised classification does not calculate (M)NDWI and the threshold values parameter is based on (M)NDWI, the tool users should not input threshold values if they do not input NIR or SWIR bands. Though simple in design, the shoreline script tool is customizable to allow the user to manually input NDWI threshold levels that can improve shoreline detection. The shoreline script tool's default approach is to identify the largest water body (by surface area) in an image and delineate its shoreline. If the user enters a value in the MinWaterBody parameter, however, all water bodies larger than the MinWaterBody value will have their shorelines delineated.

**III. Shoreline Script Tool Data Sources:**

The script tool is designed to work with rasters saved to a folder on a hard drive. These rasters can be downloaded manually or through using an API (e.g. Google Earth Engine), but the tool is not currently designed for cloud-based analysis. The Shoreline digitization script tool is designed to accept multi-band rasters/images as input. Some imagery may only be available as single-band (e.g. Landsat Surface Reflectance from Earth Explorer). In such a case, the individual bands must first be "composited" prior to input into the shoreline tool. See "Appendix" for band compositing instructions. Images with high spatial resolution (e.g. 1 meter) will increase the accuracy of shoreline detection. One caution, though, is that images of a very large file size may cause the tool to run very slowly or perhaps even cause it to fail completely. Organizations may have their own sources for high-resolution aerial imagery. If not, aerial imagery is available for public download through the USGS Earth Explorer website (https://earthexplorer.usgs.gov/), the National Map Viewer, and other sources. It is necessary to create an Earth Explorer account and be logged in before downloading data. There are many ways to search for data using Earth Explorer. One relatively simple method is to zoom into the area of interest on the Earth Explorer home page and then under "Polygon" select "Use Map." The area in the map's view will be selected. Next set your "Data Range" for desired imagery dates and select the "Data Sets" button. On the next screen under "Select Your Data Set(s)", expand "Aerial Imagery" and check the box next to NAIP (National Agricultural Imagery Program). Then select the "Results" button (you might have to scroll down to see it). Assuming you have logged in, you will now be able to download the data of interest.

**IV. Shoreline Script Tool Usage Instructions:**

Everything required to install and run the script tool is available on GitHub at:

<https://github.com/TwoPoint34/ShorelineDigitizationArcPro>

The GitHub readme contains a link to a video tutorial demonstrating how to download, install, and run the script tool. In a case where the script tool cannot be downloaded and installed in the methods described by the video, the user can easily create the script tool by following the instructions below:

*i. Accessing or Creating the Script Tool*

Step #1: Open the ArcPro "Catalog Pane"

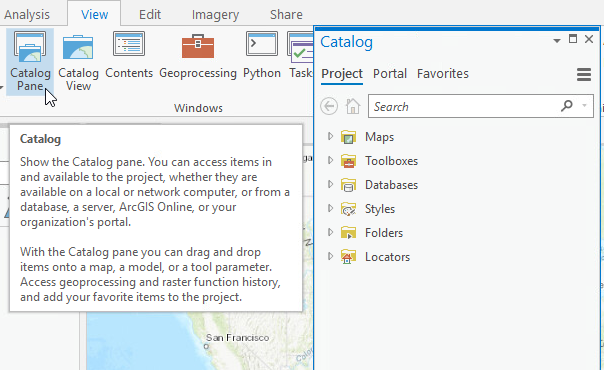


Figure #1

Step #2: To create the script tool from scratch, expand the "Toolboxes" menu and right click on the project's default toolbox then select "New" > "Script"

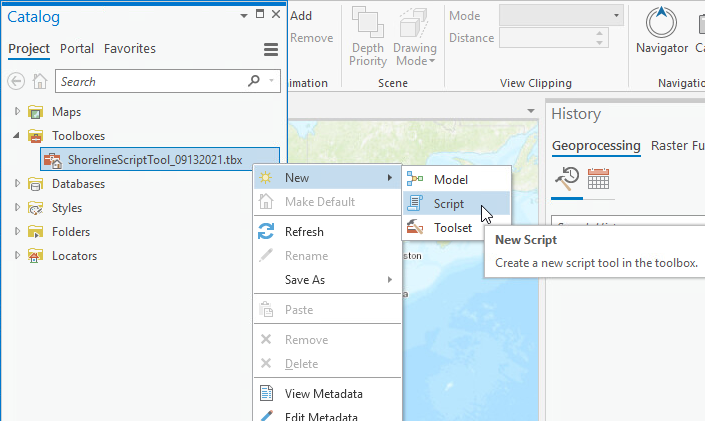


Figure #2

Step #3: In the "New Script" dialog box that opens, you will enter information in the "General," "Parameters," and "Execution" tabs. If the "New Script" dialog box does not open automatically, right click on the script tool and select "Properties."

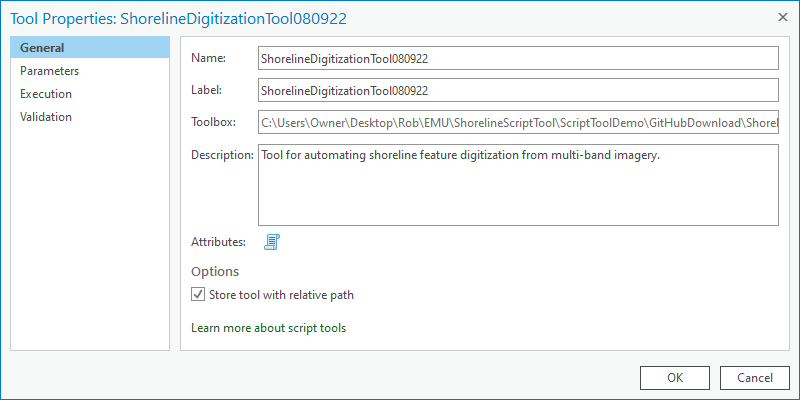


Figure #3

Step #4: In the "General" tab, give your script tool a descriptive name and label. If the script tool was successfully downloaded from GitHub, it should already have the "Parameters" fields filled out. If not, you can alternatively enter them manually. Figure #4 (below) shows the tool's parameters with descriptive names. You can change values in the "Label" and "Name" columns, but the values for "Data Type", "Type" and "Direction" should be kept exactly as shown in the screenshot below. Select "OK" after all the values are filled for the tool parameters. More detailed information on the parameters is available in the "Parameter Descriptions" section.

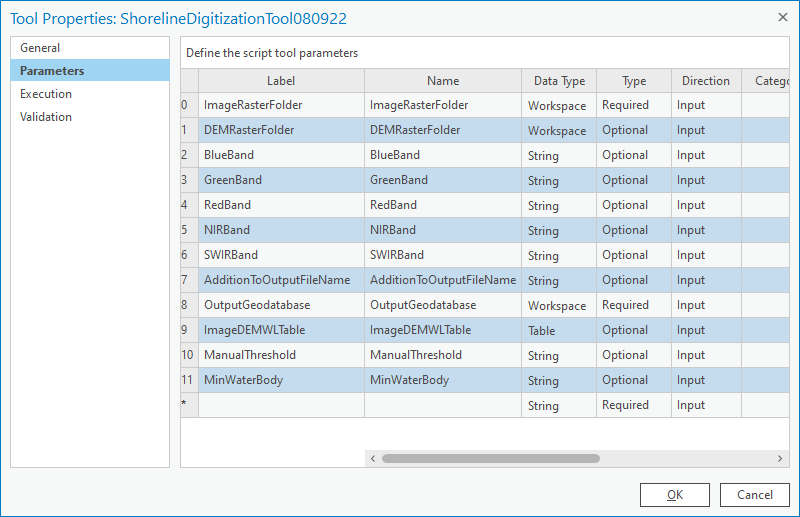


Figure #4

Note: It may seem that the OutputGeodatabase parameter should have its direction as "Output", but for some reason the tool works best with the direction set to "Input."

Step #5: In order to run the script tool, it needs to be linked to the corresponding Python code (.py file). This step is accomplished in the Tool Properties "Execution" tab under "Script File." (Note: There may not be an "Execution" tab in earlier ArcPro versions, and "Script File" will be under the "General" tab). The only action required in the "Execution" tab is to enter the .py file path in the space provided (see below). After doing so, click "OK" on the Tool Properties dialog box, and the tool is created.

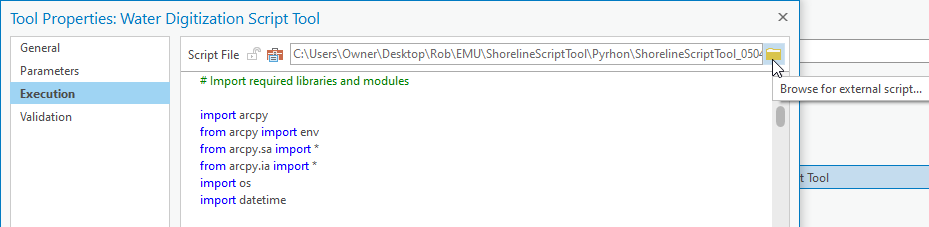


Figure #5

The script tool will be created underneath the project's default toolbox in the Catalog, and it can be accessed from the Catalog Pane at any time.

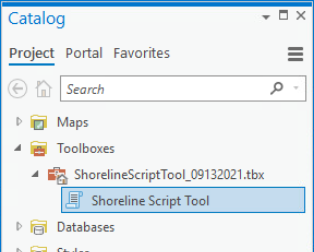


Figure #6

The tool will initially only exist in the Catalog Pane of the project for which it was created, but it can be imported to a new project by selecting Insert > Toolbox > Add Toolbox

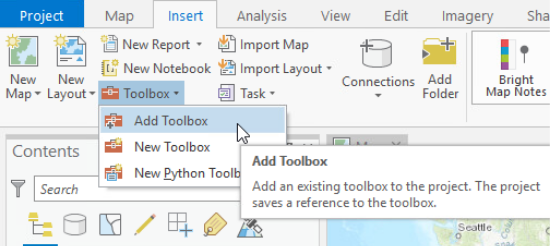


Figure #7

*ii. Running the Script Tool*

To run the script tool, simply open the Catalog Pane and double click the tool. Next, enter the appropriate parameter values (see "Parameter Descriptions" section for more info). The only required parameters are "ImageRasterFolder" and "OutputGeodatabase", though additional parameters may add functionality and increase land/water classification accuracy. Figure #8 shows an example of Shoreline Script Tool Parameter Inputs:

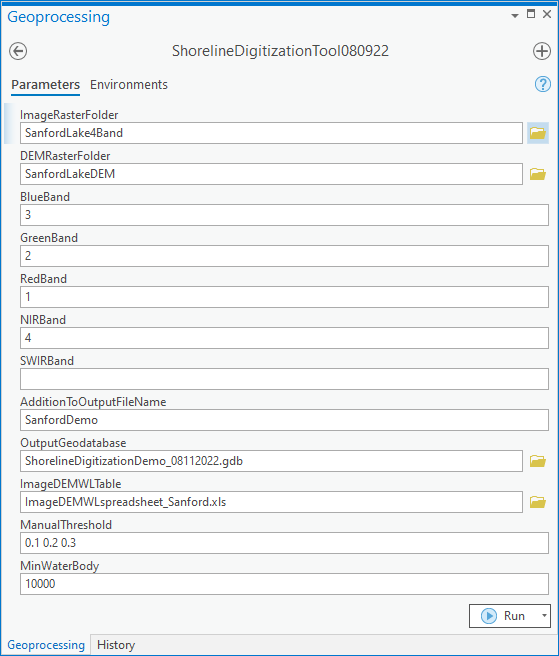


Figure #8

After entering the parameters, select "Run." The time it takes for the tool to complete will depend on the size and number or input rasters. Upon running the script tool, the outputs for the first five input rasters are added to the table of contents, and any remaining outputs will be saved to the output geodatabase but not automatically added to the table of contents. It may be necessary to "refresh" the output geodatabase before the output data appears.

**V. Parameter Descriptions:**

ArcPro begins its parameter count at zero. This discussion likewise begins with Parameter0. For easy reference, you can refer to the screenshot of the parameters in Figure #4 (above).

*Parameter0 (Required): Image Raster Folder*

This parameter is the folder path where the input aerial images are stored. Though the script tool has not been tested with all raster types, any multi-band raster format recognized by ArcPro (e.g. TIFF, GRID, JPEG2000) should be acceptable. Aerial image inputs must be multi-band. Some image types (e.g. Landsat) may require compositing (e.g. with the ArcPro "Composite Bands" tool) prior to usage in the script tool. The script tool should be capable of distinguishing the rasters from other data types in the folder, but the tool will not be able to access rasters that are within additional folders inside the Image Raster Folder path provided. To accurately detect shoreline change, it is important that all the rasters be in the same geographic coordinate system. It is also important to be aware that the tool may not work for rasters of particularly large file size, which would likely be determined by ArcGIS limitations rather than limitations specific to the shoreline script tool.

*Parameter1 (Optional): DEM Raster Folder*

This parameter can be added if the user wants to establish a contour on a DEM at a given water level. For example, the user may find ground reference data (e.g. NOAA) for the water level at the time that the aerial image was taken. That water level data can then be used to create a contour on the DEM, and that contour can then be compared (e.g. for change detection) to the shoreline delineated from the aerial image. DEMs are single-band rasters that include a z-value (elevation). It is important to note that non-shoreline contours may also be created. In DEMs covering large areas and/or hilly terrain, many such additional contours may be created. Assuming the DEM is based on a standardized vertical datum (e.g. NAVD88), the water level is most likely at zero. The tool's DEM-based shoreline contour can be compared to the shoreline delineated from the aerial image in order to identify potential areas of shoreline change. If a DEM is not available for a given area, isolated coastal elevation points and/or contour lines (based on beach slope data) could be interpolated to create a basic DEM surface for input into the shoreline script tool. More information on this interpolation approach is included in the "Incorporating Tide and Beach Slope" section below.

*Parameters 2-6 (Optional): Blue, Green, Red, NIR, and SWIR Bands*

If the script tool's default unsupervised classification algorithm does not produce satisfactory results, the user may specify a multi-band image's band order in an attempt to improve shoreline digitization results. The normalized difference water index (NDWI) has been shown (McFeeters, 1996) to differentiate water from land with a relatively high degree of accuracy. The modified NDWI (MNDWI) has been shown to often improve upon the accuracy of the NDWI (Xu, 2006). My initial observations suggest that when combined with the (M)NDWI technique, the "band ratio" technique (Niya, 2013) can in some cases assist in distinguishing water from impervious surfaces near the shoreline. If the script tool is run with (M)NDWI bands and there is a lot of interference from shoreline impervious surfaces, it may be worth adding the blue and red bands to the analysis. Adding the band ratio method allows all four bands to aid in shoreline delineation for common high resolution imagery sources (e.g. NAIP, Unmanned Aerial Vehicle). There is no harm in inputting values for all five bands (incl. SWIR) when available. The script tool input band number corresponds to the order of the band within the multi-band image. For example, in NAIP multi-band imagery, the 2nd band (green) should be input as 2 and the 4th band (NIR) should be input as 4. When inputting the band number, do not add any characters or punctuation aside from the single digit.

*Parameter7 (Optional): User Addition to Output Filename*

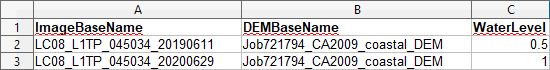
The script tool user may want to include their own addition to the output file names. Doing so may help the user to keep track of various runs of the tool. It is important that this filename addition not include invalid characters (e.g. spaces). The name asked for in this parameter is simply a text string, not a file path. The output will automatically be saved to the Output Geodatabase that is later defined in Parameter8.

*Parameter8 (Required): Output Geodatabase*

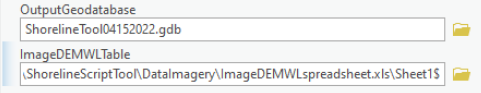
This parameter specifies the location to which the output files will be saved. To avoid potential tool confusion, the output should not be saved to the same folders that contain the aerial images or DEMs. Saving the output to a file geodatabase (as opposed to a folder) avoids tool errors and allows for easy water area measurement through the "shape\_area" field of the output polygons.

*Parameter9 (Optional): ImageDEMWLTable*

Various aerial images may be associated with various DEMs. If users include a DEM input, they must also include an Excel file (.xls or .xlsx) that contains the relation between aerial images and DEMs. The script tool code will convert the Excel file to a File Geodatabase table then loop through each row and access the aerial image and its matching DEM from their respective folder locations. The Excel file should also include water level data to provide the basis for the contour to be drawn on the DEM. The script tool relies upon exact (case sensitive) Excel file field/column titles to access the proper aerial image and DEM files from their folder locations. These field/column titles are: ImageBaseName DEMBaseName WaterLevel



A quirk of Parameter9 is that the Excel file path will sometimes include the sheet title (see below).



In order to run the tool, the sheet title must be manually deleted from the file path such that the path ends with either .xls or .xlsx.

*Parameter10 (Optional): UserThreshold*

The shoreline digitization tool's ability to accurately identify shoreline may sometimes be improved by the use of custom (M)NDWI threshold values for the land/water interface. By default the script tool uses (M)NDWI to produce: a) Otsu binary thresholding on the (M)NDWI raster and b) Otsu (M)NDWI + Band Ratio (if red and blue bands are input). Often these outputs will satisfactorily delineate the shoreline. Again, visual confirmation is required by the user to assess tool accuracy. Sometimes, however, the default outputs do not accurately delineate the shoreline. In such cases, the user may choose to re-run the tool on inadequately delineated images and use custom threshold values. Threshold values can vary substantially from one image to another. It may be possible to improve shoreline detection accuracy by narrowing down the proper threshold values over the course of several tool runs. It is not necessary for the user to understand why a given threshold may be appropriate, but from experience, realistic threshold inputs for Parameter11 are often between 0 and 0.4. They should be input in the precise format:

0.1 0.2 0.3 0.4

The tool outputs a shoreline feature and a water polygon feature for each threshold input value. The output name for the first input value is "NDWIThreshold\_1", the second input value is "NDWIThreshold\_2", etc. If it were to become apparent upon running the tool that the best threshold value lies between, say 0.1 and 0.2, then the user could re-run the tool with a 0.15 (or similar) value.

*Parameter11 (Optional): MinWaterBody*

If the user leaves the MinWaterBody parameter blank, then a shoreline and water body polygon will only be output for the largest contiguous water body in a given image. Since an image may potentially contain many very small water bodies and/or small land areas misidentified as water, these artifacts could significantly slow the tool's performance and/or its water/land classification accuracy. Therefore, if a user desires more than one water body in an image to be analyzed by the script tool, then the user must specify the smallest water body size (surface area) of interest for analysis. The units for this parameter are based on the linear units of the image, which can be found by right-clicking the image in the table of contents then looking under Properties > Source > Spatial Reference > Linear Unit. Since the input for this parameter is surface area, the input value should be the square of the linear unit. Since houses and impervious surfaces can sometimes be incorrectly classified as water, a reasonable value for this parameter would be a surface area that is a bit larger than that of a typical house and its impervious surfaces (e.g. driveway). Or, for example, if you know you have three water bodies of interest in the images being analyzed, you can input a value for this parameter that is slightly smaller than the surface area of the smallest water body of interest. Making the value slightly smaller should compensate for potential land/water classification error in the tool's algorithms.

**VI. Interpreting Shoreline Tool Output:**

The shoreline digitization script tool attempts to simplify its output for straightforward analysis. If only multi-band imagery is input, the tool will output line and polygon features. The line features represent the shoreline, and the polygon features represent the water body surface. The "shape\_area" field of the output polygons allows for a quick calculation of a water body's surface area. If a DEM is available of the water body's bottom (topobathy), this DEM could be used along with the polygon-based surface area to estimate a water body's volume. A potential issue with the 'UnsupClass' polygon output is that (unlike with the other algorithms) the polygons are not automatically classified as either water or land. Therefore, the largest polygon output may actually represent land, not water. In such a case, a simple workaround for water surface area estimation is to create a polygon (CreateConstantRaster > RasterToPolygon) then use the Erase tool with 'UnsupClass' as the Erase Features and the newly created polygon as the Input Features.

Depending on the image band combinations used, different final products will be output. The simplest workflow entails the user not entering band information, in which case only the IsoCluster based unsupervised classification algorithm is used and the output contains the name 'UnsupClass'. Inputting only the green and infrared bands leads to a binary (Otsu) thresholding product of the (M)NDWI raster. If no SWIR band is provided, the NIR band is used, and the output is named 'OtsuNDWI.' If a SWIR band is provided, its output is named 'OtsuMNDWI.' If the user provides red and blue bands, an additional output is the sum of the Otsu(M)NDWI raster and the Band Ratio raster. This sum does not appear to exist in the remote sensing literature, but my observation suggests it may help to distinguish between water and impervious surfaces. The output name of this sum is "BRNIROtsuNDWI" when used with NIR and "BRSWIROtsuNDWI" when used with SWIR. If the user manually enters (M)NDWI threshold levels, those outputs will be named "NDWIThreshold\_#." If the user inputs a table with DEM rasters and water level data, then the script tool will also create DEM contour lines. These contours will be named "GroundReferenceWaterLevel."

The full file names also include the image raster name, the user's name addition, and a timestamp in the format MonthDayHourMinute. The full output name will be available in the MERGE\_SRC field in the output feature class attribute table. See example below:

LC08\_L1TP\_045034\_20200629\_GroundReferenceWaterLevel\_UserNameAddition\_04291206

In order to facilitate output file management, all of the output features for a given input aerial image are merged into a single feature class. Individual output features can be matched with their input images and algorithms based on the output feature name. A single shoreline output feature can be analyzed simply by selecting the row of interest, and the feature will be highlighted in blue. To create a layer from a single feature alone, simply right click on the feature class in the table of contents, then choose Selection > Make Layer From Selected Features.

In many cases, the shoreline detection will be very accurate for at least one of the default threshold values (i.e. Otsu, BandRatio). In other cases, the user may benefit from entering custom threshold values and re-running the tool. In still other cases, the output features will be mostly accurate but may require some manual manipulation to improve accuracy. The degree of accuracy required will depend on the objectives of the end user. Again, there may be some cases in which full shoreline automation will not be possible, but the tool will generally save the end user substantial effort by at least partially automating shoreline detection.

After selecting a shoreline feature that closely matches the aerial image on which it was based, the tool user/analyst may be able to interpret useful shoreline change information from visual assessment. Comparing the shoreline feature delineated from the aerial images to the DEM-based contours may prove particularly useful. That said, a somewhat more rigorous shoreline change assessment can be conducted with a few additional steps (described in the following paragraph). For a very detailed shoreline change analysis, the shoreline script tool's output can be used as input for the Digital Shoreline Analysis System (DSAS).

Once users select the image-based shoreline feature that best matches their shore section of interest, they can go further by using the feature as a mask in the ArcPro's Extract by Mask tool (again, no user coding required). With the shoreline feature as a mask, the DEM pixels along the shoreline feature can be extracted. There will then be pixels with elevation data in the same horizontal location as the water in the aerial image. This elevation data should be taken with a grain of salt, but it can still be very useful in helping the analyst to identify areas of shoreline change. Adjusting the symbology of the shoreline extracted DEM can help to highlight areas of change. In my experience, the "Temperature" color scheme is a good choice for highlighting differences in pixel values.

**VII. Tool Usage Considerations, Limitations, and Troubleshooting:**

The shoreline digitization tool is best thought of as an *attempt* to fully automate shoreline detection and digitization. While the tool will accurately detect at least part of an image's shoreline in most cases, sometimes the inherent uncertainties of remote sensing will lead to inadequate detection. That notwithstanding, given the time and labor involved in manual shoreline digitization, the script tool has great potential to facilitate and automate shoreline monitoring (especially in applications with large datasets).

*General Usage Tips:*

-The shoreline digitization script tool requires an *Advanced* ArcPro license (incl. Spatial Analyst).

-Make sure that all parameters are entered correctly and are of the format required by the tool. For example, entering a file path for a folder containing a folder that contains the input rasters will result in tool failure. The file path entered must contain the rasters themselves, and the rasters must not be within an additional folder.

-Aerial images and DEMs should be in the same coordinate systems. It may be necessary to reproject image/DEM pairs into matching coordinate systems prior to running the script tool.

-Make sure that the horizontal units (e.g. feet, meters) are the same for both the input aerial images and DEM.

-Sometimes ArcGIS (and the Shoreline Script tool by extension) will behave very strangely (e.g. failing when it previously succeeded with the exact same parameters). It may help to reboot ArcGIS and/or your computer or possibly start a new ArcGIS project.

-Using Jupyter Notebook while ArcPro is also running can sometimes cause issues. Rebooting the program(s) will sometimes fix the issue.

-It is possible for geodatabases to become corrupt, and it may sometimes be necessary to repair a geodatabase or create a new one.

-Be sure that input files do not contain invalid characters (e.g. spaces)

-Due to time constraints, in its current form the tool largely lacks validation and error-checking measures. Common sources of error that could cause tool failure include:

Issue: Tool fails and returns an error: 999999

Potential Solution:

-"999999" is a generic ArcGIS error code that may have many causes. One possibility is that the input raster(s) may be too large.

Issue: Shoreline not delineated accurately

Potential Solutions:

-Double-check that the band numbers input in the script tool parameters match the order of bands in the input rasters. It's important to note that different imagery types (e.g. NAIP vs. Landsat) may have different band orders

-If imagery contains a SWIR band but it was not used in the tool's run, try inputting the SWIR band number.

-Inputting the red and blue bands may improve shoreline detection accuracy (esp. if there are many impervious surfaces along the shore).

-Clouds will interfere with analysis. Find images without clouds or remove clouds from the images prior to analysis by the script tool.

-Also, images of smaller areas will likely yield more accurate shoreline delineation, though the tool can often successfully process large areas.

Issue: Failure to acquire a "schema lock" on a table. This error may manifest itself as other (non schema lock) error codes as well

Potential Solution:

-Make sure that the tool is not being accessed from multiple applications (e.g. ArcPro script tool, Jupyter Notebook) simultaneously

Issue: "ArcGIS.Desktop.Core.Geoprocessing.GPItemException" error is thrown.

Potential Solution:

-One step in the script tool involves the conversion of an excel file to a file geodatabase table. This action requires the installation of a driver, discussed here: <https://community.esri.com/t5/arcgis-pro-questions/arcgis-desktop-core-geoprocessing-gpitem/td-p/1130590>

and here:

<https://pro.arcgis.com/en/pro-app/latest/help/data/excel/work-with-excel-in-arcgis-pro.htm>

Issue: Tool ran successfully but data does not appear in "Add Data" pane

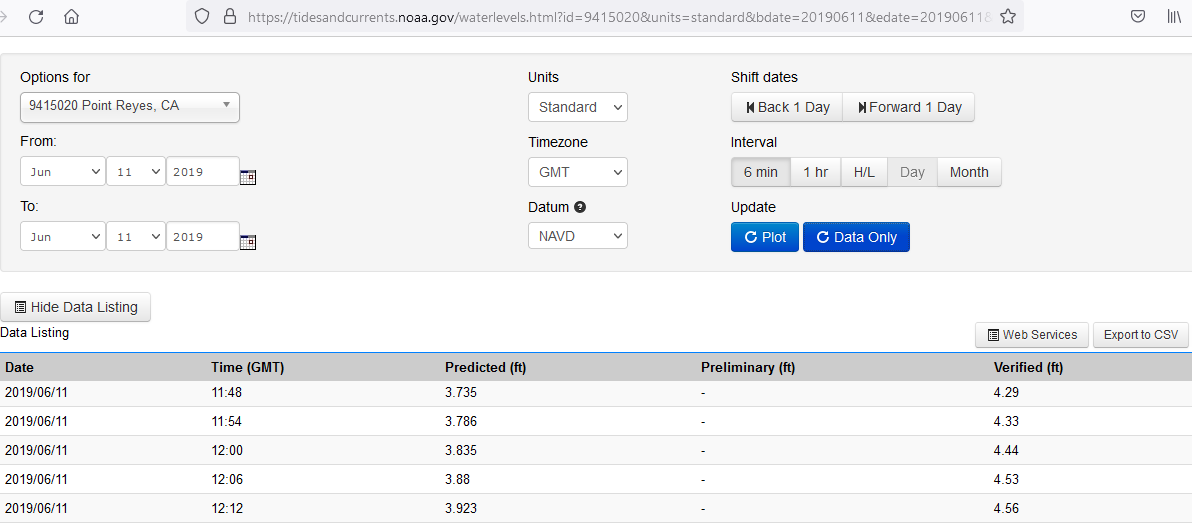
Potential Solution:

-Refresh 'Add Data' pane

**VIII. Incorporating Tide and Beach Slope:**

When analyzing aerial images of large water bodies (e.g. oceans), the tide (water level) at the time/day of imagery capture becomes very significant. For example, an image taken at low tide could lead an analyst to conclude that the shoreline had expanded seaward if compared to an image taken at high tide. The primary purpose of the DEM input into the shoreline digitization script tool is for tide and beach slope considerations, though it could also be useful for flood modeling based on various water level scenarios. If only aerial images and no DEM are input, then the tool can still be useful to delineate the shoreline and estimate water surface area of minimally tide-affected water bodies such as small inland lakes and rivers. In tide-affected water bodies, the script tool does not attempt to *adjust* aerial image-derived shoreline features for tide, per se. For one, simply adjusting the shoreline features horizontally is not likely to accurately represent variations in beach slope, which would essentially require a DEM. Basing shoreline change monitoring on high-resolution DEMs alone would likely be prohibitively expensive and very labor-intensive. Using the shoreline digitization tool to analyze multi-band aerial imagery, on the other hand, can greatly increase the temporal resolution of shoreline change monitoring and greatly decrease the associated labor and cost.

The shoreline digitization tool draws a contour on a DEM based on the Water Level input by the tool user into an Excel spreadsheet ("ImageDEMWLTable" parameter). Water Level values can be accessed from the user's field measurements or through published data. If the day/time of the multi-band aerial image capture is known, then it may be possible to access tide data from the National Oceanic and Atmospheric Administration (NOAA, 2022). The screenshot below shows an example of water level data available from NOAA:



Unfortunately, NAIP data does not include time of capture (USDA, 2022), though Unmanned Aerial Vehicle and high resolution satellite data generally does. When accessing water level data, be sure to match the vertical datum (e.g. NAVD88) of the water level data to the vertical datum used for the DEM.

If a DEM is not available, a user may be able to create a simple DEM from multiple elevation data points and/or a slope raster through interpolation. ArcGIS has various tools for interpolation, such as "Spline" (for points) and "Topo to Raster" (for contour lines). The literature suggests that spline is a common approach for interpolating coastal surfaces, though the script tool user will likely want to conduct their own research if using an interpolation approach.

**IX. References:**

McFeeters, Stuart K. "The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features." International journal of remote sensing 17, no. 7 (1996): 1425-1432.

National Oceanic and Atmospheric Administration (NOAA)- Center for Operational Oceanographic Products and Services. "Tides and Currents." July 15, 2022. https://tidesandcurrents.noaa.gov/.

Niya, Ali Kourosh, Ali Asghar Alesheikh, Mohsen Soltanpor, and Mir Masoud Kheirkhahzarkesh. "Shoreline change mapping using remote sensing and GIS." International Journal of Remote Sensing Applications 3, no. 3 (2013): 102-107.

United States Department of Agriculture (USDA)- Farm Service Agency. "Explanation of the shapefile accompanying the NAIP CCM image, and how to access the image date information (ArcGIS 9.x Version)." July 15, 2022.

https://www.fsa.usda.gov/Internet/FSA\_File/am\_imagedates.pdf

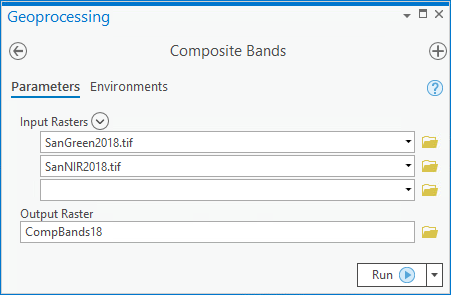
Vos, Kilian. "CoastSat." Github. May 03, 2022.

Xu, Hanqiu. "Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery." International journal of remote sensing 27, no. 14 (2006): 3025-3033.

**X. Appendix** *Using the Shoreline Digitization Tool with Single-Band Imagery*

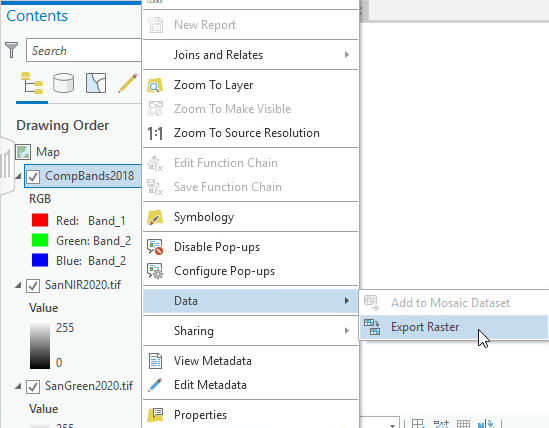
1) Use ArcGIS "Composite Bands" tool

It is only necessary to composite the bands that will actually be used in the script tool. For example, if you are only conducting an NDWI analysis, you will only use the green and NIR bands, and the composite raster will only have two bands.

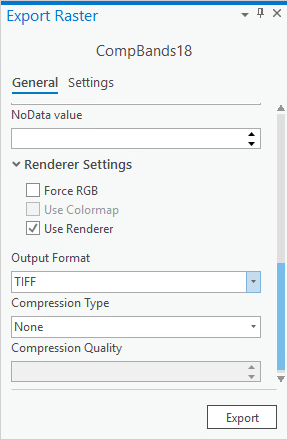


2) Export Composite Raster (ideally directly into the folder that will be used as script tool input)

i)



ii)



3) Execute the Script Tool with the Composite Band Rasters

Input the folder containing the composite band rasters as a script tool parameter and input the corresponding band numbers in their appropriate tool parameters. Generally the first band added to the ArcGIS "Composite Bands" tool will be band #1, the second band added will be band #2, and so forth.