# ArcMap Add-In Button:
# Assigning Band Combinations to Imagery Based on Sensor Type & Study Type

*David Lindsey*

***GISC 6387 - GIS Workshop Project***

***for Remote Sensing Certification***

***2018 August 1st***

## **Introduction**

For decades, scientists have utilized various remote sensing platforms in order to study spectral signatures on earth and beyond. While these spectral signatures do not change, the band associations among the differing sensor platforms for these spectral signatures can and do change. Depending on the remote sensing platform utilized, a researcher may be dealing with four spectral bands, five bands, seven bands, or many more. What band combinations do researchers need to keep up with in order to study identical spectral features among varying sensors? Additionally, the user needs to know what the different band combinations create, display, and emphasize (e.g. urban sprawl, vegetation analysis, land/water differentiation, among others).

In order to accomplish the multitude of spectral studies with various sensor band combinations, a user must be relatively knowledgeable of remote sensing principles. Those users must understand the various sensors available, any spectral limitations of each sensor, as well as the necessary band combinations required to produce the desired study type. In the grand scheme of things, there are dozens of sensors, each with potentially dozens of bands. A person could only be faulted as being human for not having all of these sensor types and band combinations memorized! Wouldn't it be easier to have a tool that could load an image with a preset sensor type and study type? No user would then need to manually propagate band combinations.

For any users ingrained with utilizing ArcMap for analysis purposes, they will be disappointed to discover that there is no tool in place to accommodate such a request. ArcMap, as currently designed, does not support the ability to automatically assign raster band combinations for different areas of study, much less identify and assign which sensor captured the image[1]. For example, if a user wishes to load a Landsat-7 image and focus on a vegetation study via the RGB band combination 4, 3, and 2, the user would need to manually assign this combination with prior knowledge of that sensor's band capabilities. This would be done by manually adjusting band combinations within the image layer's properties (see **Appendix – Figure 1**).

In order to alleviate a user's need to memorize and/or research sensor band combinations, this project seeks to create an ArcMap Add-In button that will automatically import an image based on sensor type and study type (i.e. the appropriate band combination per study type per sensor type). The desired outcome is to better streamline the user's imagery research experience, as well as fix a shortcoming within Esri's ArcMap software.

## Literature Review

Numerous research examples exist which clearly warrant a need for this tool. One such research paper is entitled *Comparative Analysis of Landsat-5 TM and SPOT HRV-1 Data for Use in Multiple Sensor Approaches* (Hill and Aifadopoulou, 1990). The paper discusses how the SPOT-1 and Landsat-5 sensors are compared when monitoring various vegetative and non-vegetative targets of interest. To summarize, the paper concludes that both sensors are similarly useful in monitoring agricultural and vegetative processes. Within this research, the authors elaborate on the differences each sensor possesses. They even include a useful table for visual reference, showing how Landsat-5 possesses seven unique bands (including RGB/NIR), while SPOT-1 possesses only three of these seven bands (RG/NIR). This would indicate that a

[1] http://desktop.arcgis.com/en/arcmap/latest/manage-data/raster-and-images/setting-default-raster-display-options.htm

researcher needs to select bands 4,3,2 on Landsat-5 in order to study vegetation, while bands 3,2,1 on SPOT-1 would need to be selected.

Another reference paper is entitled *Remote Sensing of Soil Salinity: Potentials and Constraints* (Metternicht and Zinck, 2003). This paper discusses the need and importance of monitoring soil salinity in various locations. The authors essentially go into a review of numerous sensors and how they could benefit salinity research as well as hinder the process through sensor limitations (ground-based and remote). What stood out with this paper is how many sensors are listed, with differing band combinations representing various methods of spectral research. The authors include an extensive table displaying the various bands available for each sensor. These sensor reviews include Landsat 4-7, SPOT 1-4, LISS 2-3, among others. Due to the evolution of most remote sensing platforms, band combinations have a tendency to change with new sensor releases. Depending on the platform utilized, the band combinations for various studies may vary.

The research paper entitled *Combination of Proximal and Remote Sensing Methods for Rapid Soil Salinity Quantification* (Aldabaa, Weindorf, Chakraborty, and Li, 2015) also discusses soil salinity and differing sensor approaches to monitoring it. These included both land-based sensors as well as Landsat-7 and Landsat-8 imagery. The authors conclude that the Landsat data is least accurate among those sensors tested (as compared to ground-based sensors). However, the authors also spend considerable time comparing the band differences between Landsat-7 and Landsat-8. Due to the evolution of Landsat-7 to Landsat-8, the band combinations have changed. To view natural color with Landsat-7, bands 3, 2, 1 are used. With Landsat-8, this changes to bands 4, 3, 2.

Given the types of sensors that exist for analyzing imagery and their spectral bands, an Add-In tool of this nature could save the user time and effort by properly symbolizing imagery based on application-prescribed band combinations. A user would not be required to research and memorize band combinations per study area, per sensor.

Geospatial software companies like Hexagon Geospatial and Esri both have picked up on this importance, and have incorporated tools designed to perform similar tasks within ERDAS IMAGINE[2] and ArcGIS Pro[3]. However, Esri has yet to retroactively introduce this feature into ArcMap. The fact that so many users still rely on ArcMap is what makes this Add-In tool necessary.

**Study Area & Data Sources:**

In order to build the Add-In tool, several software components were required. For this project's purposes, Microsoft Windows 10 was used. Microsoft Visual Studio 2015, C#, Microsoft .NET Framework 4.7, and ArcObjects SDK 10.5.x were utilized and tested with ArcMap 10.5.1 (on Advanced licensing level). Numerous, non-specific imagery from multiple sensor platforms were obtained and used for testing the application's various phases (E.g. Landsat 7-8 imagery acquired from USGS, Sentinel 2-3 imagery acquired from the European Space Agency, etc.). For research and implementation use, physical reading resources were helpful and, ultimately, necessary to accomplish all desired objectives[4].

[2] https://www.hexagongeospatial.com/products/power-portfolio/erdas-imagine/erdas-imagine-product-release-details
[3] http://pro.arcgis.com/en/pro-app/help/data/imagery/raster-display-ribbon.htm
[4] Please see Reference section for specific sources utilized.

**Methodology:**

The programming scripts that were constructed to create the overall application were prioritized, structured, and implemented as follows:

1) The Add-In Button relies on coding functionality found within the "Button_AddRaster.cs" file. This code controls the Add-In Button itself, and immediately implements the "GUI_AddRaster.cs" file. This button is designed so that a user cannot click/open the tool twice prior to the tool being exited. In other words, the button icon becomes "disabled" once the application has been executed, and the button will remain this way until the tool has been exited by the user. The code is 51 lines in length.
2) As mentioned before, once the Add-In Button has been clicked, the button click event triggers the "GUI_AddRaster.cs" file. This coding controls the Windows Form that opens for the user. All parameters are identified within this window. It consists of Sensor/Study Type drop-down boxes, as well as a "Browse" button for selecting the raster image for import into ArcMap. This code consists of 1149 lines of code.
3) Within the "GUI_AddRaster.cs" file, other classes and functions are accessed as well. All sensor types are stored as independent class code files with a naming convention similar to "Sensor_{**SensorName**}.cs". These files will import, and be reliant upon, the functions found within the study type codes. Each of the 26 sensor classes range in length from 52 -157 lines of code each
4) Each study type is stored as an independent class code files as well. The naming conventions for the study type codes are "Class_AssignTextBoxRGB_{**StudyType**}.cs". These class files are accessed within the aforementioned sensor class files. These files house the actual band combinations per study type per sensor. If any sensor band

combinations need adjustment, these codes are where the changes need to be made by a user. There are nine class codes total, for each of the nine study types available. Each of those files are 428 lines of code apiece.

5) Within each sensor file, another class file is accessed by the code. There is a code known as “Class_BandCombination.cs” which is responsible for importing the raster image into ArcMap with proper layer naming convention, band assignment, and zoom extent. This code contains 105 lines.
6) Within the “GUI_AddRaster.cs” file, once a user selects a raster image for import, another class file called “Class_CountRasterBands.cs” is executed. This class performs a band count on the user’s selected raster image. This function establishes the total band count for the raster image, and is responsible for comparing the image band count to the expected sensor type’s band count. This file contains 100 lines of code.

All codes and functions (see **Appendix – Figure 2**) introduced above will be expanded upon in the following section.

*Operation-specifics for the ArcMap Add-In Tool:*

The ArcMap Add-In tool has been designed as a button for the user to click. The button’s icon will appear to the user as a red satellite (see **Appendix – Figure 3**). Once the button is clicked, the button itself will become disabled/non-selectable. Additionally, a Windows Form dialog box will appear with multiple parameters (see **Appendix – Figure 4**). Within this Windows Form, the user will first need to select the type of sensor that the imported image will correspond with. For the current release of this application, 36 sensors are available to the user. The following sensors are available within the drop-down list: ASTER, EO-1 ALI, EO-1 Hyperion, FormoSat 1-2, FormoSat-5, GeoEye-1, IKONOS, Landsat 1-5 MSS, Landsat 4-5 TM, Landsat 7-8, MODIS, NAIP (3-Band), NAIP (4-Band), OrbView-3, Pleiades-1, QuickBird, RapidEye, Sentinel 2-3, SPOT 1-7, and WorldView 2-4 (see **Appendix – Figure 5)**. Once the user has specified the sensor type, the sensor's designated band count will populate beneath the drop-down list (see **Appendix – Figure 6**). This number will vary depending on the sensor selected (e.g. Landsat-7 = 8 bands, Landsat-8 = 11 bands, Sentinel-2 =12 bands, etc.).

As soon as the sensor type has been selected, the "Study Type" drop-down list will become selectable[5]. Depending on the sensor selected, the user will be able to choose among the following options: Agriculture, Bare Earth, Burn Scarring, Healthy Vegetation, Land-Water Boundaries, Natural Color, Natural Color-No Atmosphere, Urban, and Vegetation (see **Appendix – Figure 7**). This list of options will fluctuate depending on the sensor selection and its spectral capabilities. The study type designation will cause RGB values to populate beneath the "Study Type" drop-down list (see **Appendix – Figure 8**). These values represent the band combination for that particular sensor's selected study type.

The next task for the user is to click the "Browse" button under the "Select Raster Image" section and navigate to the image that will be loaded into ArcMap (see **Appendix – Figures 9 and 10**). The tool is programmed to only display Esri-designated raster files having appropriate raster file extensions, as defined within ArcMap's Raster Option defaults[6]. Having this restriction will prevent such data as Shapefiles and text-based files from accidentally being selected. After the user selects the image, the file path will populate within the text box and the raster image's band count will populate underneath the file path (see **Appendix – Figure 11**). The application is

[5] Please note: This drop-down list will be inaccessible if the user has not specified a sensor type. This is due to the code being reliant on the sensor type before establishing band combinations affiliated with the study types.
[6] Open ArcMap > Click Customize tab > Open ArcMap Options > Click Raster tab > Click File Formats button

programmed to execute a band count on any image selected by the user, so the band counts will vary by selection.

With both a sensor type and raster image selected, the code then executes a comparison between the sensor type's band count and the raster image's band count. If the two band counts do not match, both count values will become highlighted in red and a red warning message will appear (see **Appendix – Figure 12**). The warning message will state "WARNING: Total band count of sensor selection does not match total band count of image selection. Band combinations may appear incorrectly matched." The appearance of this warning message does not prevent the tool from operating. However, depending on the bands that may be missing from the image, the band combinations may not display correctly once imported into ArcMap. Once the band counts for the sensor type and image are aligned, the red highlights and warning message will disappear.

After the raster image has been selected, the user can choose to check/uncheck the "Zoom to Raster Extent" checkbox option in the lower-left corner of the window (see **Appendix – Figure 13**). Having this option "checked" will cause the application to zoom to the imported image's layer extent within ArcMap. Leaving the option "unchecked" will import the image, but it will not cause ArcMap to deviate from the user's current extent within the map. This may be useful for users who wish to import various imagery, but not lose the location where they are currently performing analysis. By default, the "Zoom to Raster Extent" checkbox is set to "checked" when the tool is launched.

After the user has defined all input parameters, the "OK" button can be clicked. This action will cause the tool to check for any user-defined errors. For example, if a user clicks "OK" without selecting an image or defining a sensor/study type, a pop-up warning message will appear (see **Appendix – Figures 14, 15, and 16**). This message will instruct the user to select all input requirements before the tool can proceed. If all input parameters are properly selected, the tool will then select the image and define the band combinations based on the sensor type and study type. The image will then be imported into ArcMap with appropriate band symbology.

While importing the image into ArcMap, the coding sets a pre-defined naming convention for the new image layer: {**SensorType**}_{**StudyType**}__{**RasterImageName**}. If any spacing exists within the Sensor Type name, Study Type name, or image name, the code will automatically remove the spacing. This standard naming convention will allow the user to keep track of which study types are being loaded into ArcMap, allowing for multiple study types of the same image to be loaded without confusion.

Lastly, there is an “Exit” button that the user can click once they are finished using the tool. This will exit the application and allow the Add-In button (the red satellite) to become selectable once again.

## Results & Discussion:

With the Add-In tool constructed, the envisioned functionality has worked as intended. All tasks perform smoothly and produce time-saving approaches for research conducted with various imagery bands.

One issue was encountered during the application’s development, and it had to do with the GUI window freezing for a couple of seconds. If a user selects a raster image with dozens upon dozens of bands (E.g. EO-1 Hyperion image contains 242 bands), the window may freeze for one to two seconds while the application counts the total number of bands. An attempt was made to rectify this issue by introducing these counting tasks within a separate “thread”, which alleviated the GUI from freezing. However, this was causing other problems with the appearance

/ disappearance of label messages within the code. Due to time constraints for this project, and the fact that the “freeze” time was no more than two seconds, the threading concept was removed from implementation.

## Conclusions and Future Study

With the creation of this Add-In tool, a user has an efficient means to import raster imagery into ArcMap with appropriate band combinations required for various study types. Although this list only encompasses 36 sensors and nine study types, enhancements to this application would easily be achievable. The way the tool was programmed via code, it would be very simple for another programmer to expand the study type list or sensor type list with minimal invasive programming adjustments. Additionally, if more time were allotted for this application, the threading concept could be reintroduced to prevent the window from freezing momentarily. The application could also be expanded to automatically detect the sensor type based on a user’s raster image selection. This could be accomplished through a combination of pixel size identifications along with the sensor band counts. Even without these enhancements, the application provides users with a helpful tool for importing imagery and performing analysis on various band combinations.

**References:**

Aldabaa, Weindorf, Chakraborty, and Li (2015). Combination of proximal and remote sensing methods for rapid soil salinity quantification. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0016706114003437

Amirian, P. (2013). *Beginning ArcGIS for Desktop Development Using .NET*. West Sussex, UK: John Wiley & Sons, Ltd.

Hill, Joachim and Aifadopoulou, Dorothea (1990). Comparative Analysis of Landsat-5 TM and SPOT HRV-1 Data for Use in Multiple Sensor Approaches. Retrieved from: https://calval.cr.usgs.gov/PDF/Hill.J_RSE.34.1990_Chander.pdf

Metternicht, G.I. and Zinck, J.A. (2003). Remote sensing of soil salinity: potentials and constraints. Retrieved from: https://s3.amazonaws.com/academia.edu.documents/42114460/Remote_sensing_of_soil_salinity_potentia20160204-27585-1s11eyy.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1527366220&Signature=dRqRf1kfxcg%2FPxjCV6Kd37vXWy4%3D&response-content-disposition=inline%3B%20filename%3DRemote_sensing_of_soil_salinity_potentia.pdf

Sharp, John. (2015). *Microsoft Visual C# Step by Step 8th Edition.* Redmon, WA: Microsoft Press.

## **Appendix**

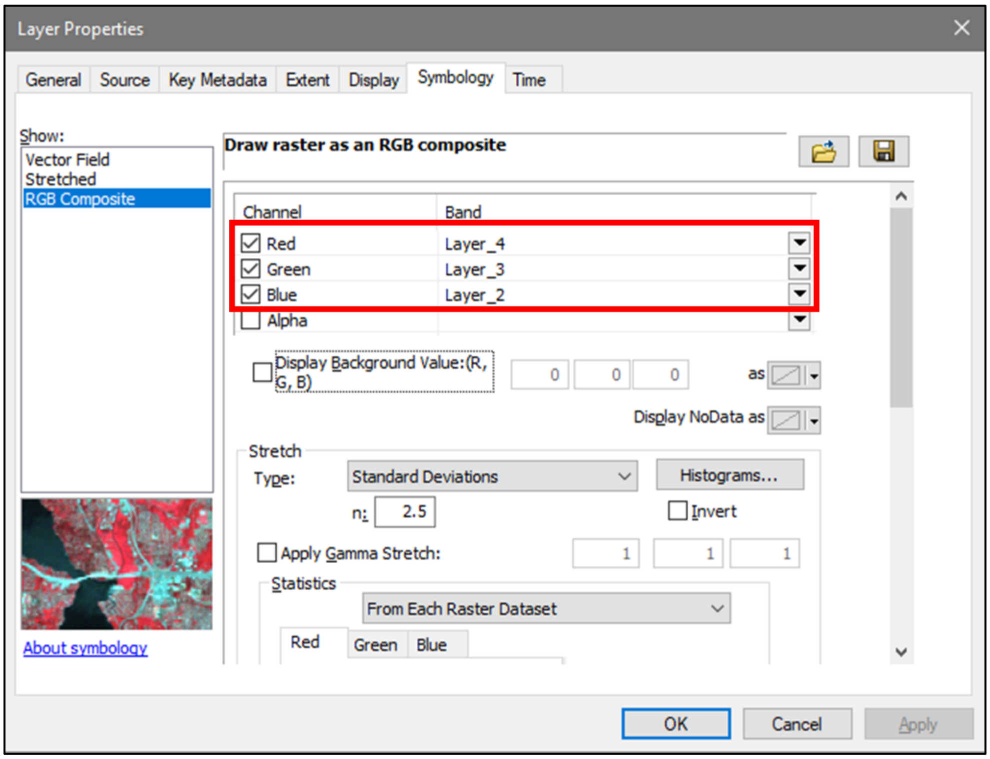


*Figure 1*

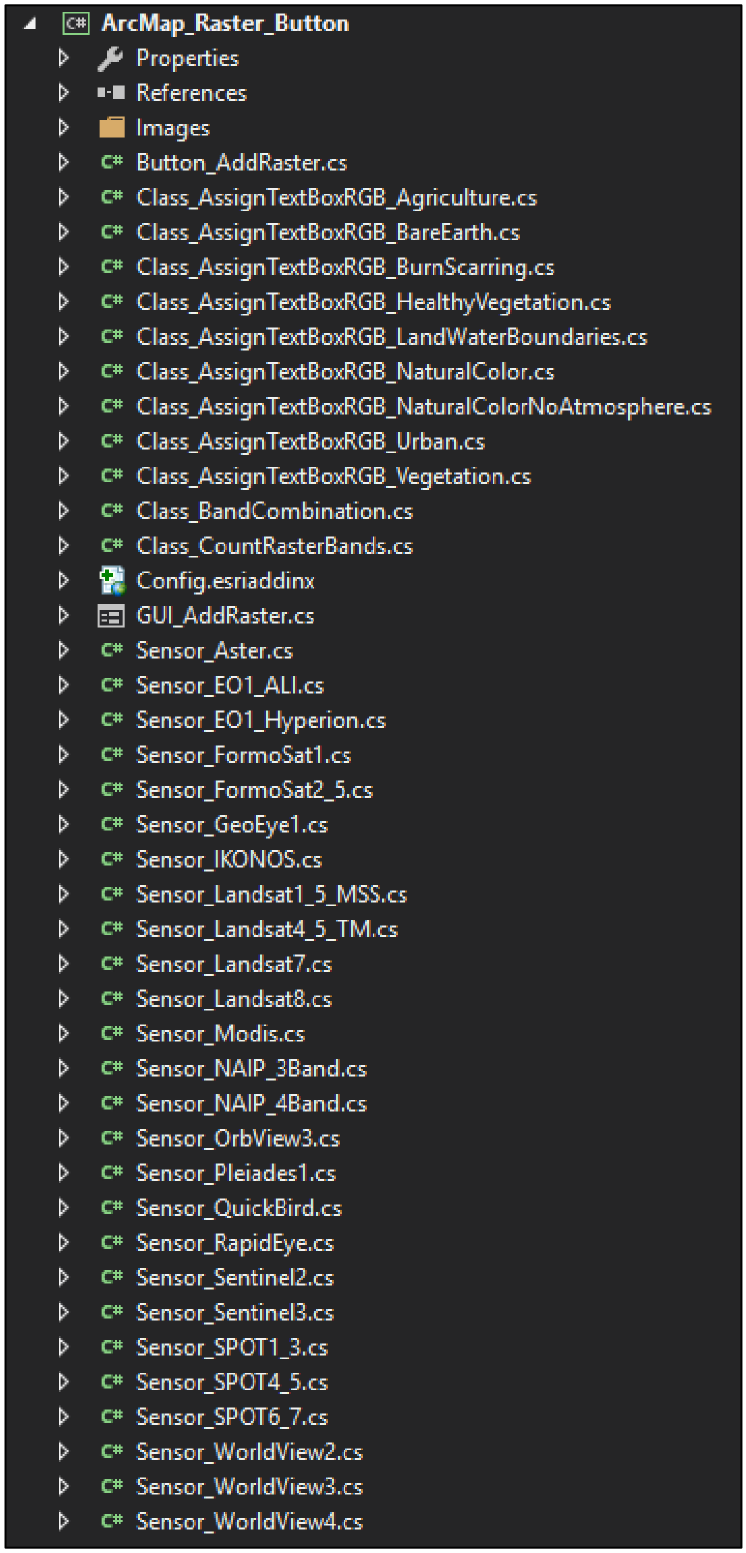
ArcMap_Raster_Button
Properties
References
Images
Button_AddRaster.cs
Class_AssignTextBoxRGB_Agriculture.cs
Class_AssignTextBoxRGB_BareEarth.cs
Class_AssignTextBoxRGB_BurnScarring.cs
Class_AssignTextBoxRGB_HealthyVegetation.cs
Class_AssignTextBoxRGB_LandWaterBoundaries.cs
Class_AssignTextBoxRGB_NaturalColor.cs
Class_AssignTextBoxRGB_NaturalColorNoAtmosphere.cs
Class_AssignTextBoxRGB_Urban.cs
Class_AssignTextBoxRGB_Vegetation.cs
Class_BandCombination.cs
Class_CountRasterBands.cs
Config.esriaddinx
GUI_AddRaster.cs
Sensor_Aster.cs
Sensor_EO1_ALI.cs
Sensor_EO1_Hyperion.cs
Sensor_FormoSat1.cs
Sensor_FormoSat2_5.cs
Sensor_GeoEye1.cs
Sensor_IKONOS.cs
Sensor_Landsat1_5_MSS.cs
Sensor_Landsat4_5_TM.cs
Sensor_Landsat7.cs
Sensor_Landsat8.cs
Sensor_Modis.cs
Sensor_NAIP_3Band.cs
Sensor_NAIP_4Band.cs
Sensor_OrbView3.cs
Sensor_Pleiades1.cs
Sensor_QuickBird.cs
Sensor_RapidEye.cs
Sensor_Sentinel2.cs
Sensor_Sentinel3.cs
Sensor_SPOT1_3.cs
Sensor_SPOT4_5.cs
Sensor_SPOT6_7.cs
Sensor_WorldView2.cs
Sensor_WorldView3.cs
Sensor_WorldView4.cs

*Figure 2*

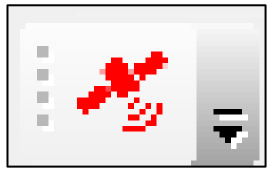

*Figure 3*

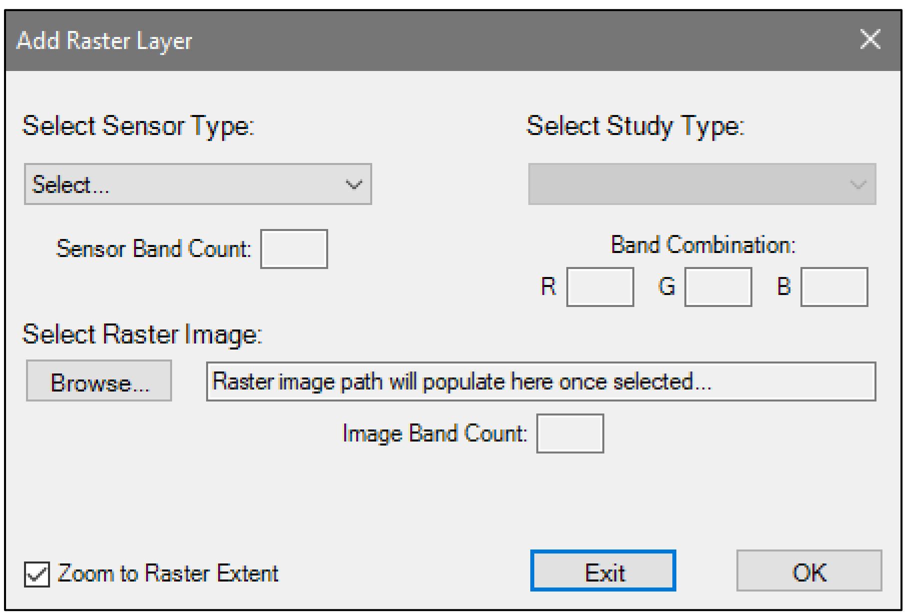


*Figure 4*

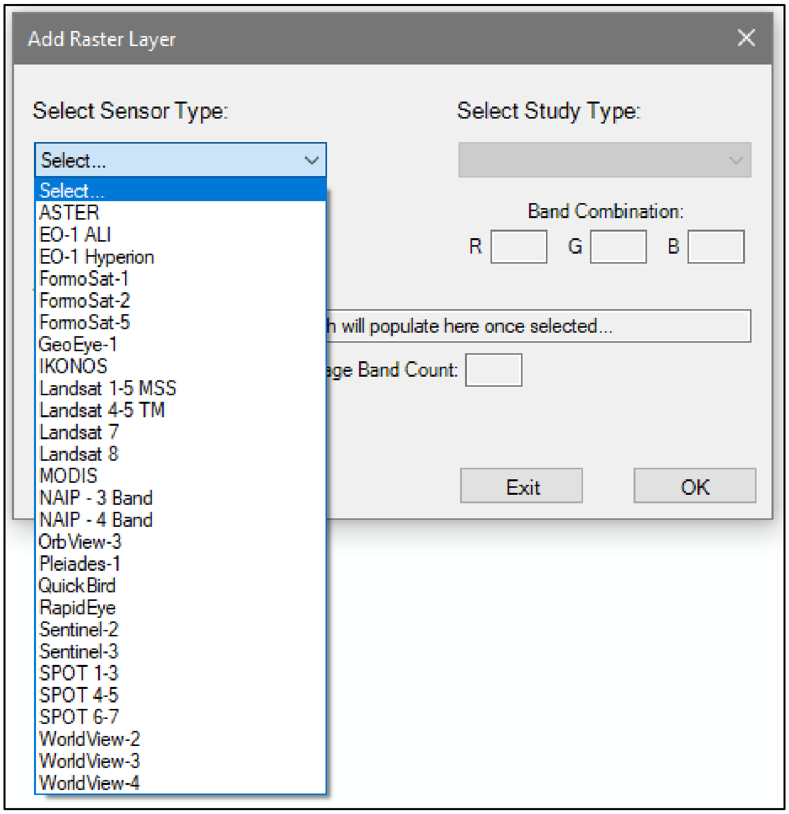


*Figure 5*

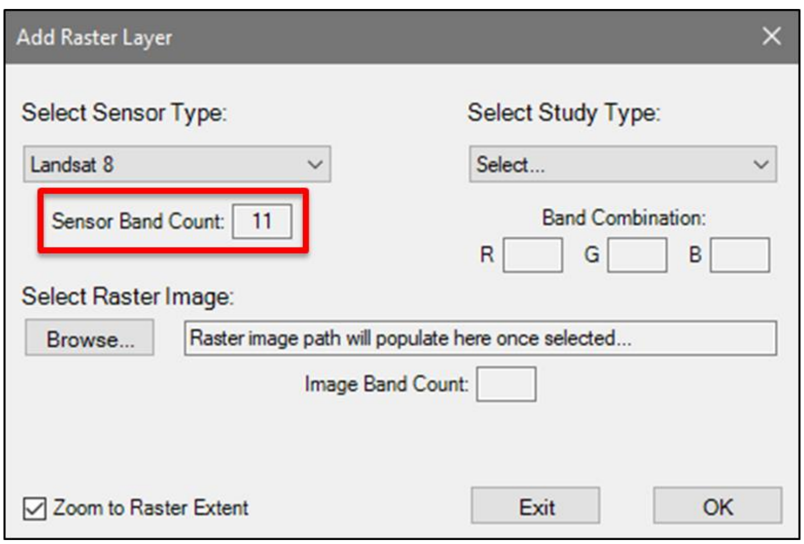


*Figure 6*

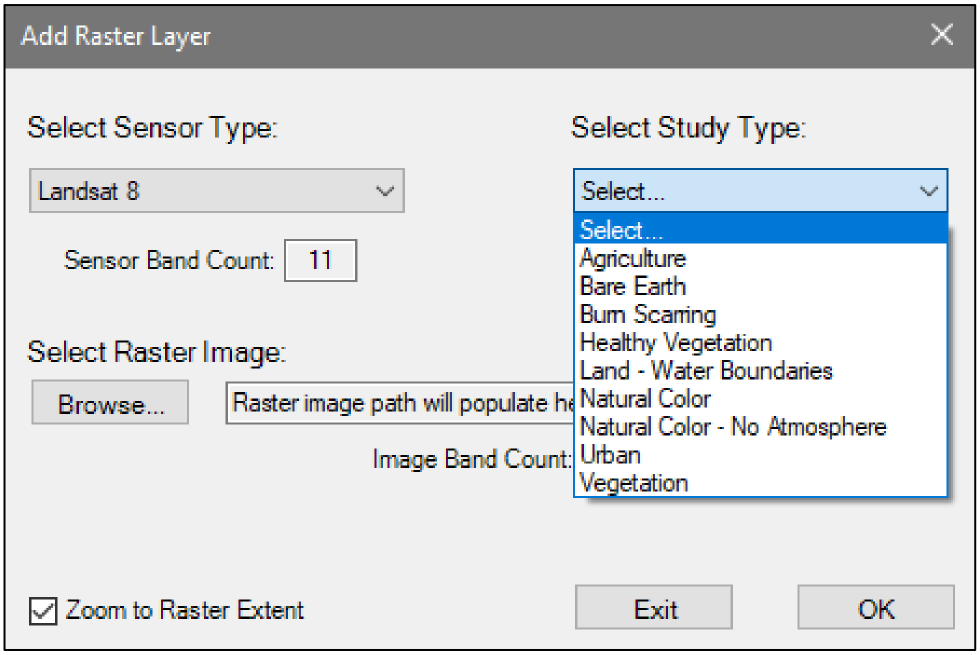


*Figure 7*

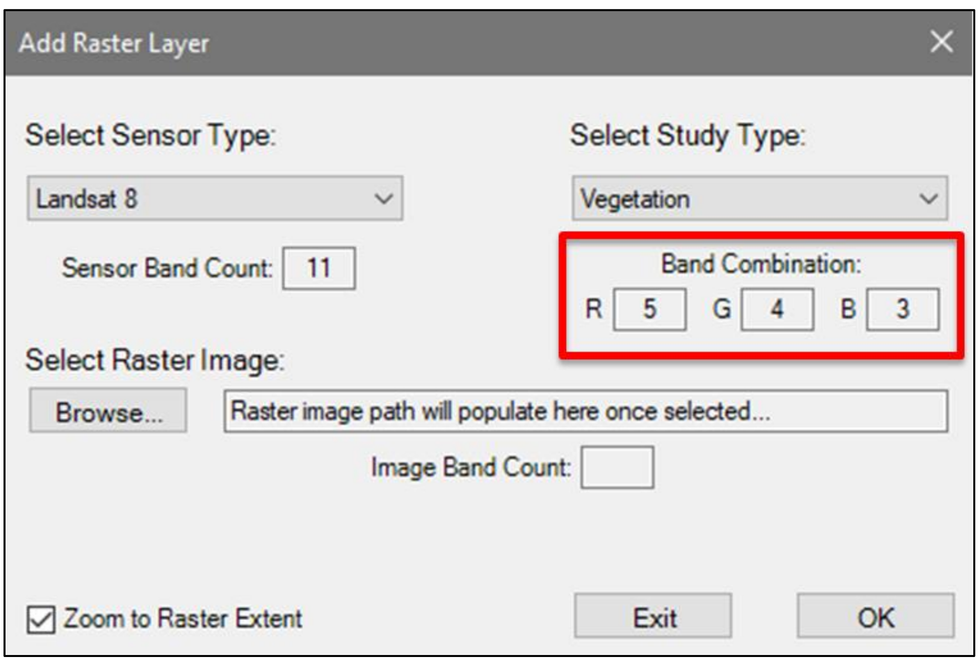


*Figure 8*

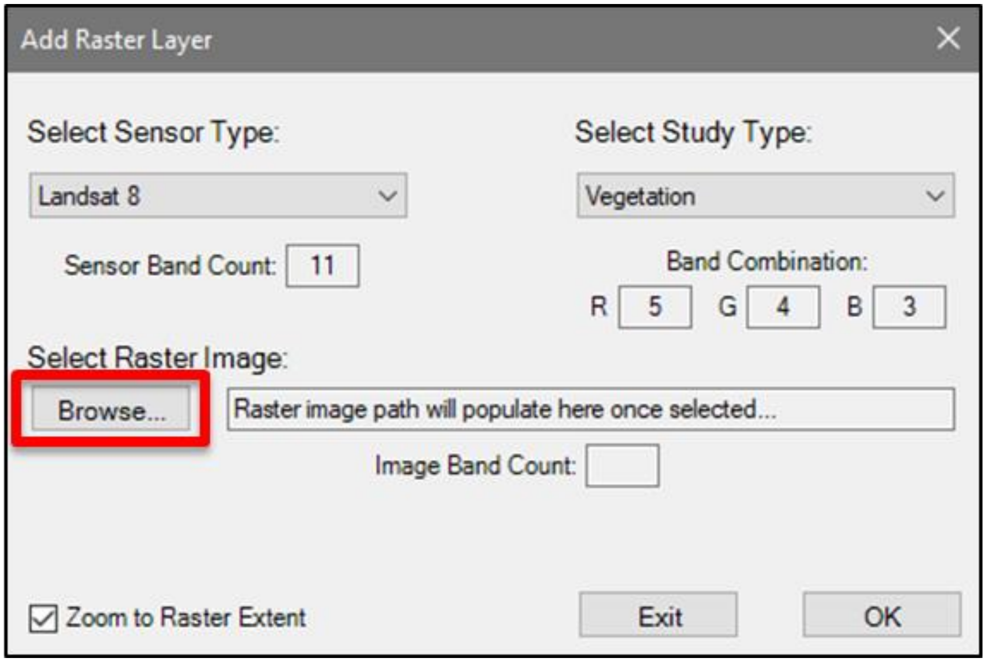


*Figure 9*

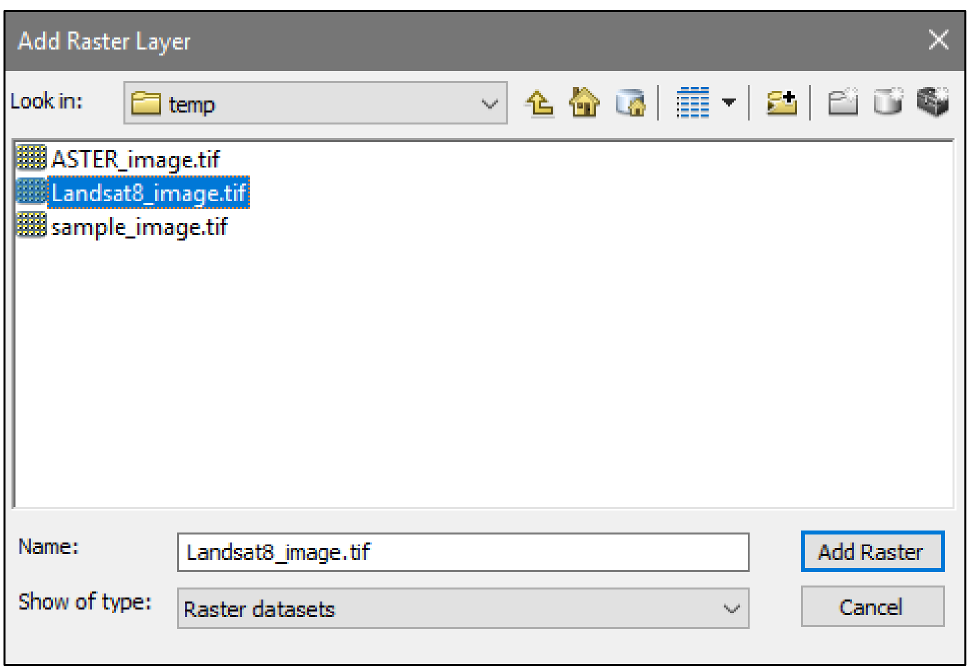


*Figure 10*

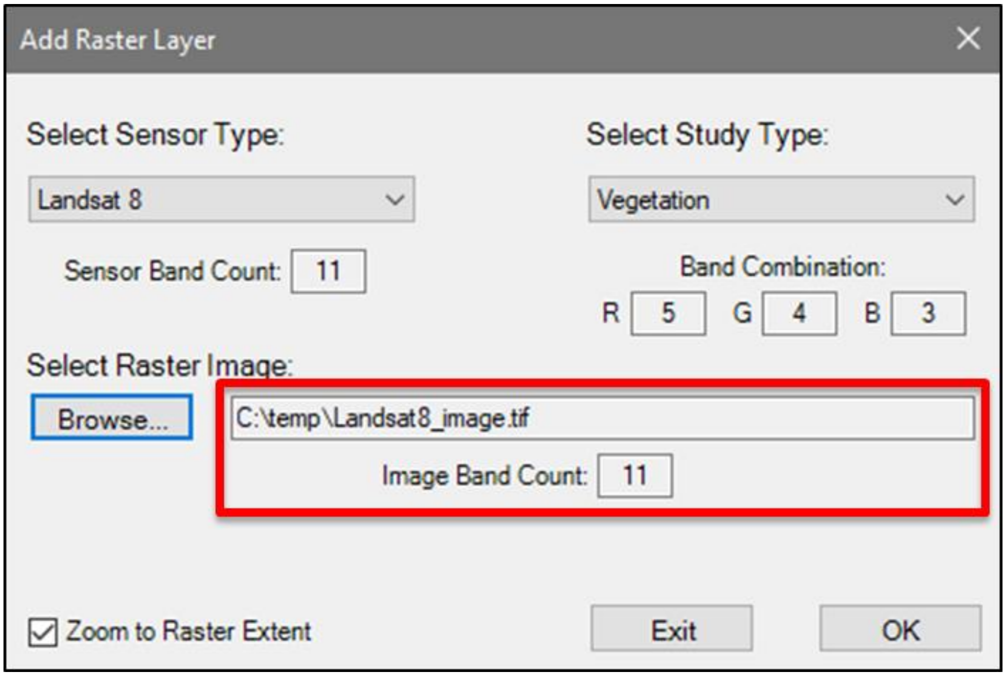


*Figure 11*

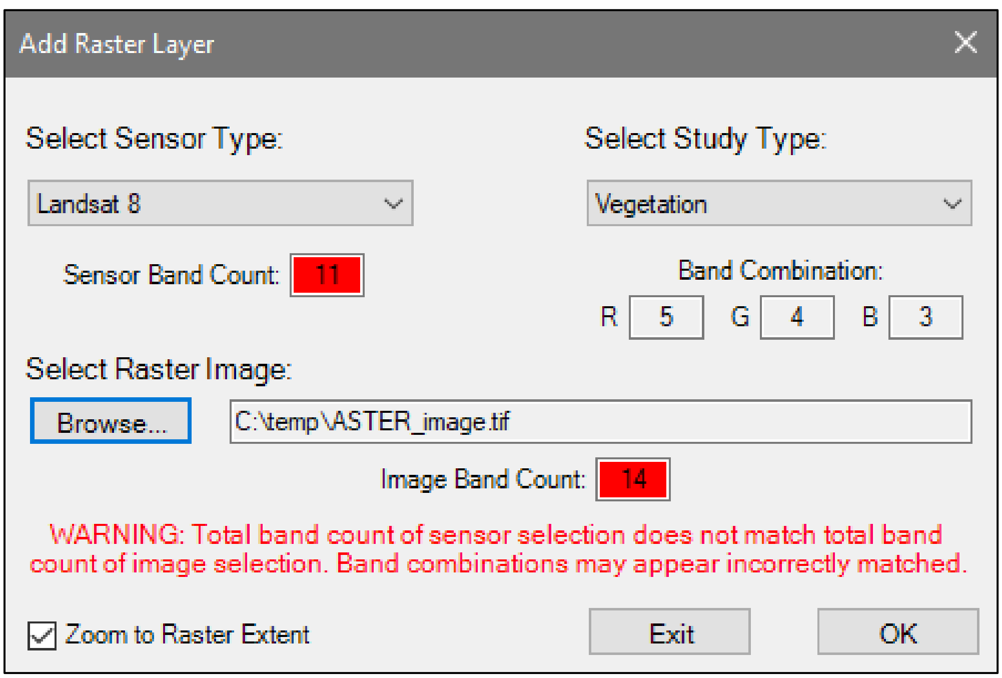


*Figure 12*

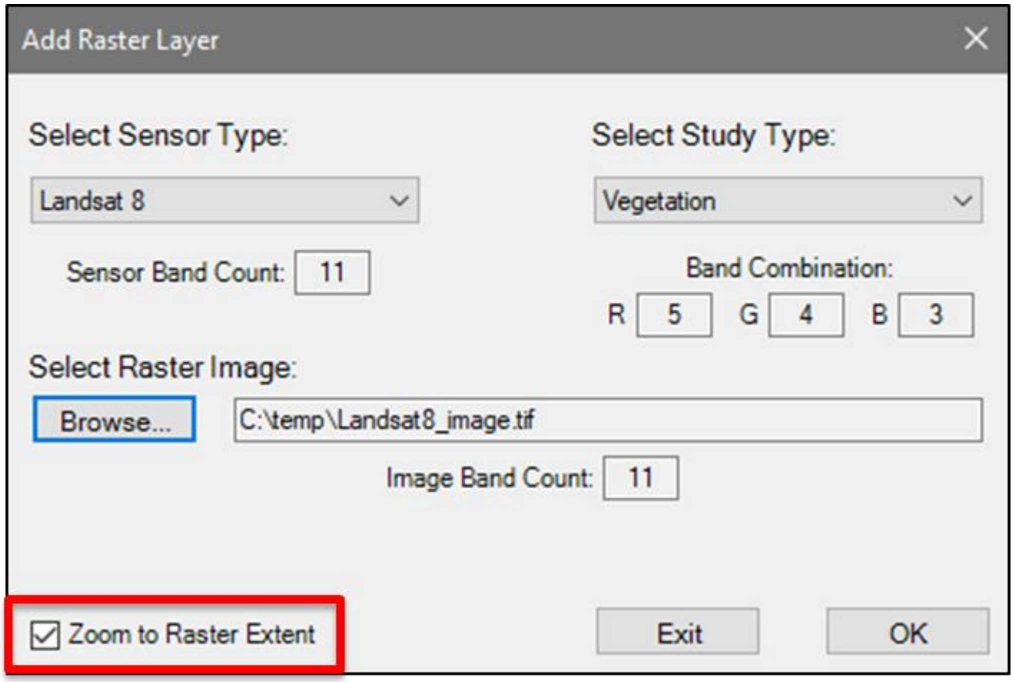


*Figure 13*

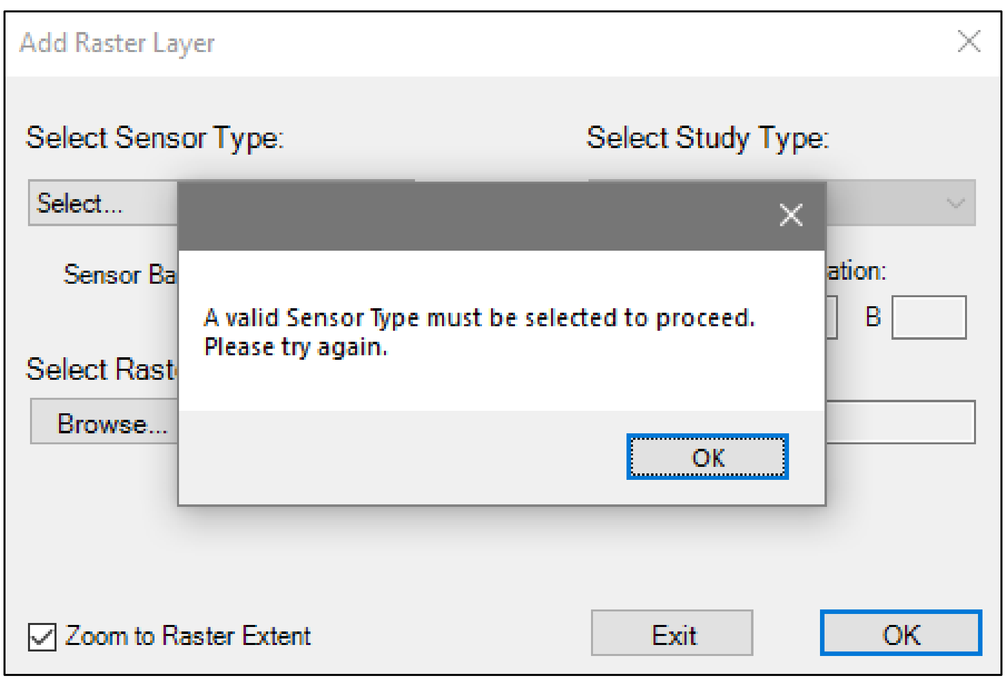


*Figure 14*

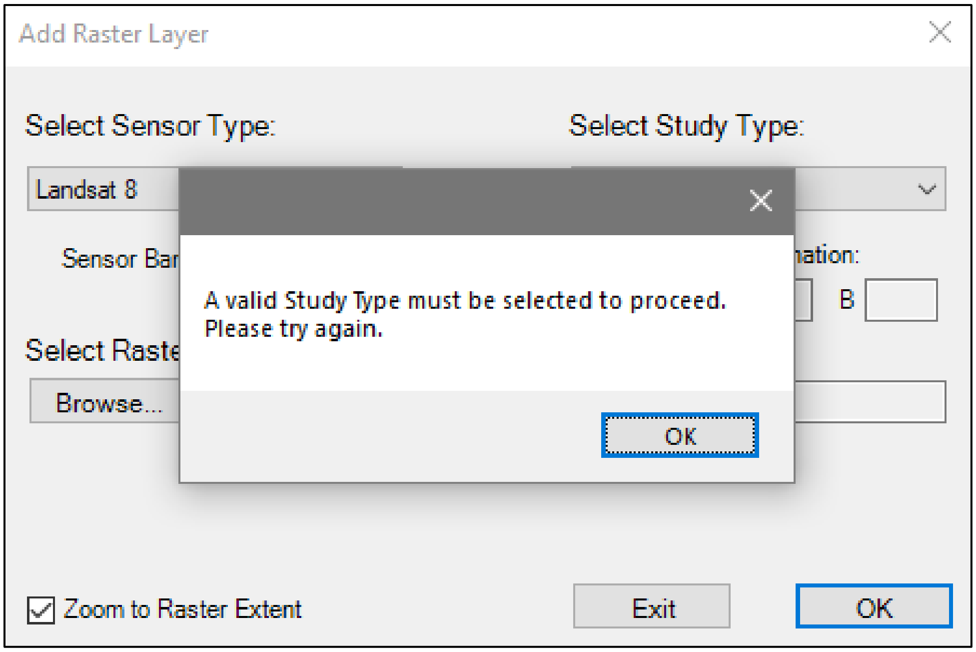


*Figure 15*

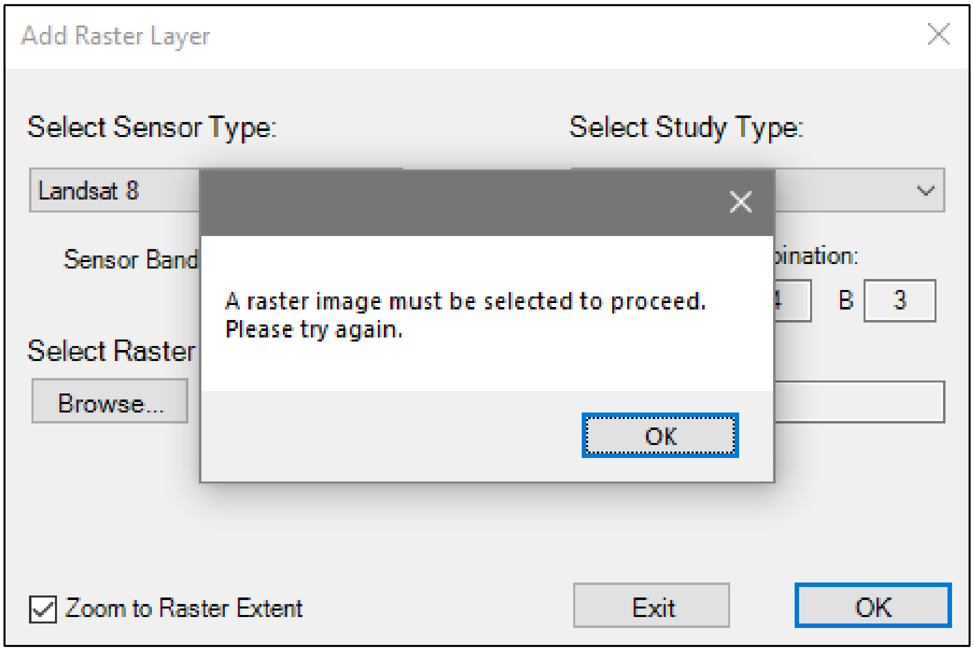


*Figure 16*

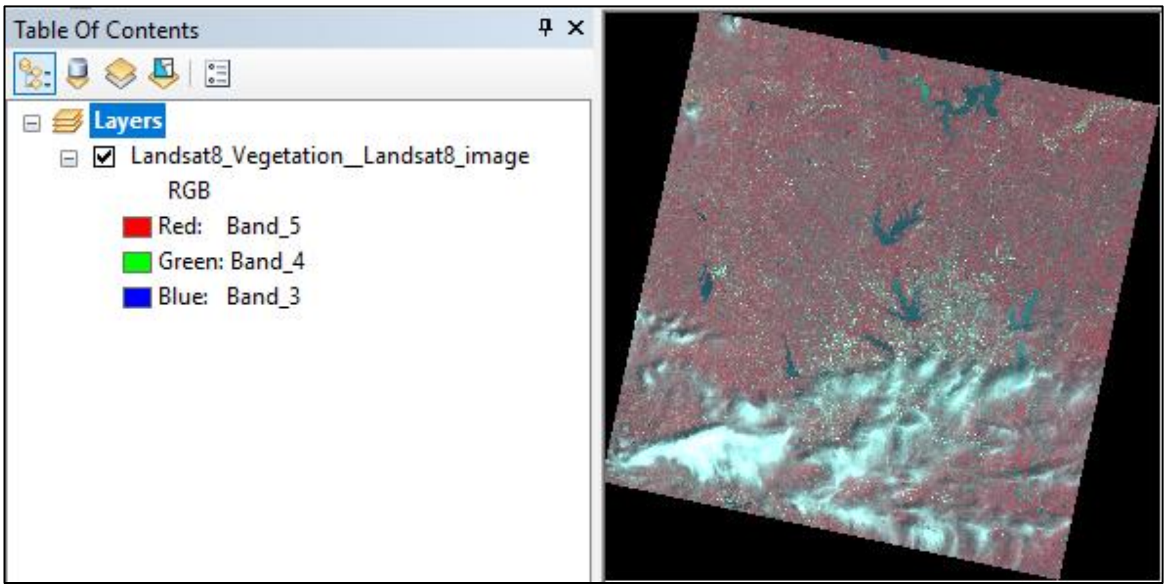
Table Of Contents
Layers
Landsat8_Vegetation__Landsat8_image
RGB
Red: Band_5
Green: Band_4
Blue: Band_3

*Figure 17*