AUTOMATION OF MULTIPLE IMAGE FUSION ALGORITHMS AND ASSESSMENT OF FUSED PRODUCTS USING STATISTICAL METHODS

Bienvenido G. Carcellar III[1] and Diona Mae Lisniana[2]
[1]University of the Philippines Diliman, Department of Geodetic Engineering, Quezon City, Philippines
Email: bgcarcellar@up.edu.ph
[2]University of the Philpines Diliman, Training Center for Applied Geodesy and Space Technology, Philippines
Email: dmhlisniana@gmail.com

KEY WORDS: Remote Sensing, Python, Pansharpening, Landsat

ABSTRACT: The continuous technological advancement in the field of remote sensing due to the numerous satellites emerging bring us variety in data. Satellite images have different characteristics in terms of temporal, spatial, radiometric and spectral resolutions. To integrate numerous sources of satellite images, the technique called image fusion is soon done to different satellite images. Image fusion or pansharpening is the process of creating a new image which is a combination of information from two source images – one that has a high resolution and another which will be the source of spectral characteristics. There are numerous algorithms to fuse a panchromatic image with its multispectral counterpart. With the increase in the developments of pansharpening algorithms, it is a need now to determine which technique is most suitable for the intended application. The main objective of this research is to automate five existing pansharpening techniques, namely: IHS, Wavelet, IHS-Wavelet, RVS, and Ehlers. After automating these fusion algorithms, the researchers also automated different spectral and spatial statistical tests to release the quality metrics of the produced pansharpened image. For comparing the different fusion methods, we employ different statistical tests to quantify both the spatial and the spectral quality of the fused images. For spectral quality metrics, we compare spectral characteristics of images obtained from the different pansharpening methods with the spectral characteristics of the original multispectral images. While in spatial quality metrics, we are comparing the pansharpened image with the panchromatic band specifically the spatial characteristics. In measuring the spectral quality, we used three indices to measure their quality, namely: Deviation Index, Correlation Coefficient, and Signal to Noise Ratio. For measuring the spatial quality of the images, the only metric used is the high pass deviation index (HPDI).

1. INTRODUCTION

As the field of remote sensing and satellite technology become advanced, satellite data and images are very accessible. This advancement also leads to the increase in variety of spatial data and satellite images. Trying to utilize images from different sources can be very tricky and hard to do since most satellites have different characteristics, such as temporal (how frequent a satellite passes through a certain place), spatial (the ground distance equivalent of a pixel from the image), radiometric (how large a data occupies in the memory, and how deep the variations in greyscale) and spectral resolutions (how wide or narrow the sensor utilizes certain parts of the electromagnetic spectrum). This variety is very beneficial for remote sensing specialists as it give them more resources and more images to compare.

There is no satellite image, however, which has perfect temporal, spatial, radiometric, and spectral characteristics. For most cases, there are existing trade-offs in having a more accurate and finer satellite images. One example is the spatial-spectral trade-off. Images produced through optical sensors most of the time are either of higher spatial resolution or higher spectral resolution. Most satellites provide high-resolution panchromatic images but low-resolution multispectral images.

Images that offer both higher resolutions are produced through higher sensors. Usually, however, these data are not free to use. To come up with a solution to this problem, merging or fusion of two satellite images are done to come up with more detailed satellite images for analysis.

Image fusion or pansharpening is the process of creating a new image which is a combination of information from two source images – one that has a high resolution and another which will be the source of spectral characteristics. This results to a single image with the spatial detail of the panchromatic image and the spectral characteristics of the multispectral image.

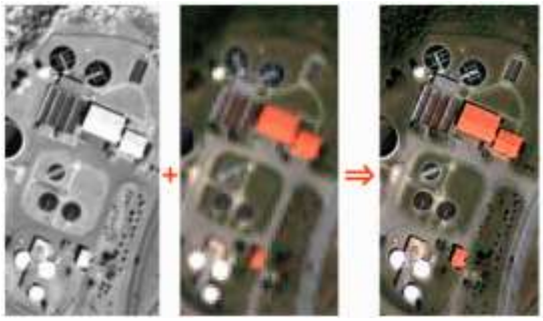

Figure 1. A panchromatic image (Left) with a higher spatial resolution as compared with the multispectral image (middle). Fusing both images results to the pansharpened image (right).

2. OBJECTIVES

In response to the problems stated above, there is a need then, to automate different pansharpening algorithms in order for us to compare these algorithms and possibly determine the best pansharpening algorithm for each case. The main objectives of this project are as follows:

- To read .TIF files of panchromatic and RGB bands
- To automate at least three image fusion methods
- To output an image file using the fusion method chosen by the user

3. METHODOLOGY

3.1 Program Requirements and Limitations

The program generates the pansharpened image, given a panchromatic and a multispectral image. The multispectral image may be inputted as one image with multiple bands, or as different tiff files per band, ike The program only automates five pansharpening algorithms that will be discussed below in part B. The following fusion methods are chosen since these methods are not present in more recent image processing software such as ENVI.

Also, the program assumes that the images are in tiff format, as the main focus of pansharpening here in this program are LandSat images. The program also assumes that when the inputted with a single multispectral image, we assume that the blue is the 1st band, green is the 2nd band, and red is the 3rd band.

3.2 Flow Charts and Equations

3.2.1 Fusion Methods

The process of pansharpening images is quite similar in a sense that the first step in the process is to resample the multispectral image to the resolution of the panchromatic image in order to fuse correctly. The main flowchart, as shown in Figure 2, is used before proceeding to the different fusion methods.

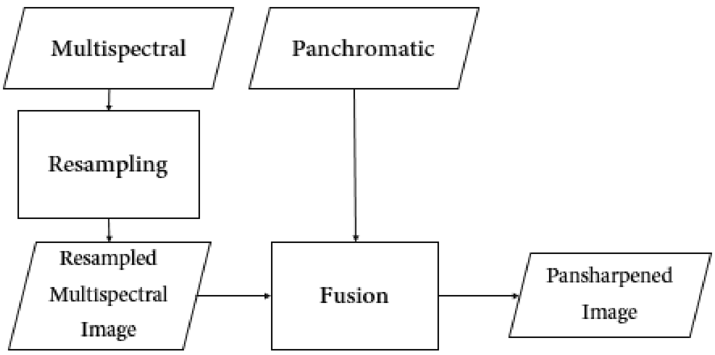

Figure 2. The main flowchart used for pansharpening.

Inside the GUI.py script, we created different functions for resampling the multispectral images. The functions pix2map, map2pix, and resample, are those functions that resamples the multispectral images.

The function openFile, opens the tiff files using the open function of the gdal module. It requires the path of the image to be opened, and returns the array counterparts of these images. Inside the openFile function, it utilizes the resample function so that the returned array from the multispectral band is the resampled version.

The Fusion process box here is where we substitute the different pansharpening algorithms. Here, as stated earlier, we utilized 5 fusion algorithms. The following images show the flowcharts for each fusion algorithm (excluding the Ehlers transform).

The first fusion algorithm employed is the HIS transform. This fusion method is quite popular since it is easily utilized by just employing the formula from the RGB Bands to transform to HIS counterparts. The Intesity band here is important as this band represents the color values. We can say that the intensity band is the merged RGB bands. Because of this, we manipulate the Intensity band and relate it to the panchromatic band. We then acquire a new intensity band which we use to transform to the new RGB. The main flowchart for the HIS transform process is shown below in figure 3.

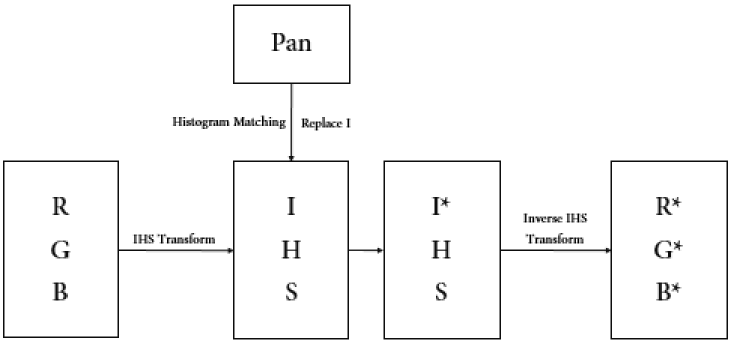

Figure 3. The main flowchart used for IHS Transformation algorithm

The wavelet transform, on the other hand, utilizes the wavelet decomposition method. The panchromatic histogram matched red, green, and blue images are decomposed to form four sub images each, named HH, HL, LH, and LL. Figure 4 below shows the decomposition output from the panchromatic image of the sample image used for this program.

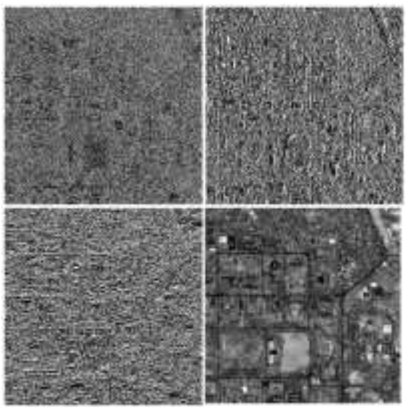

Figure 4. The output decomposed images; HH (upper left), HL (upper right), LH (lower right), LL (lower left) from the panchromatic image of UP Diliman.

These decomposed images have a spatial resolution that is twice lower than the resampled image. As shown in figure 4, we can see that the LL image resembles the original image. In the wavelet transform, we replace these decomposed images with the original unresampled bands, to create a new RGB. The flowchart below, in Figure 5, shows the step by step process of the Wavelet transform algorithm.

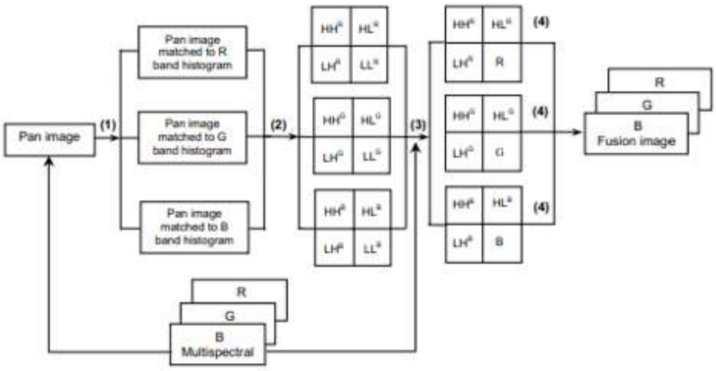

Figure 5. The flowchart used for Wavelet Transform Fusion algorithm.

Another transformation method done in this program is the IHS-Wavelet transformation. This method is classified under the combined fusion algorithms, like the Ehlers transform. Unlike the original Wavelet transform in which we decompose all three bands, here we only decompose the intensity band. We also decompose the panchromatic band, and using a weighting formula used by Al-Wassai, et al. We manipulate the LL of the intensity and the LL of the panchromatic band. The output of this equation is the new LL image to be substituted to the LL of the panchromatic band. We then perform a reverse wavelet transform to arrive at a new intensity band. Then we employ the HIS-RGB transformation to finally arrive at the pansharpened RGB bands.

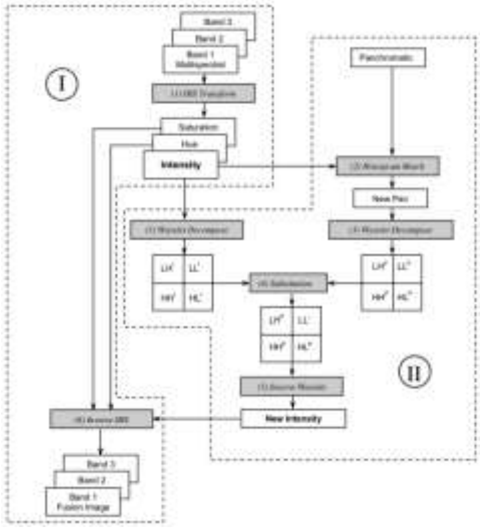

Figure 6. The flowchart for the IHS-Wavelet transform fusion algorithm.

The last fusion algorithm used in this program, representing the Arithmetic Combination, is the Regression Variable Substitution Transformation, or the RVS. Here, the algorithm utilizes statistical formula to relate the old RGB and the panchromatic image. The process is to solve the coefficients for the main linear formula for solving the new RGB values for the pansharped image. The coefficients, namely a and b as shown in Figure 7, is solved by using covariance, variance, and mean values of each band.

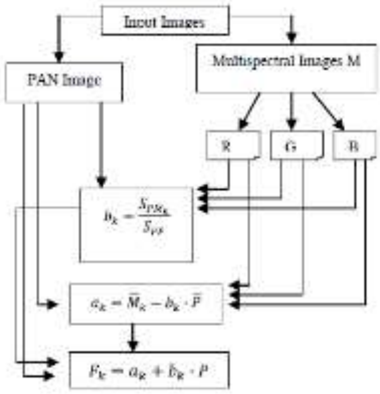

Figure 7. The flowchart utilized in the RVS transform fusion method.

3.2.2 Comparison Methods

For comparing the different fusion methods, we employ different statistical tests to quantify both the spatial and the spectral quality of the fused images. For spectral quality metrics, we compare spectral characteristics of images obtained from the different pansharpening methods with the spectral characteristics of the original multispectral images. While in spatial quality metrics, we are comparing the pansharpened image with the panchromatic band specifically the spatial characteristics.

In measuring the spectral quality, we used three indices to measure their quality. The first index is the deviation index, which measures the normalized global absolute difference of the fused image with the multispectral image (Bethune, Muller, Donnay. 1998). Below is the equation used in solving the DI, and its counterpart function in the program

$$DI_k = \frac{1}{nm}\sum_{i}^{n}\sum_{j}^{m}\frac{|F_k(i,j) - M_k(i,j)|}{M_k(i,j)} \quad (eq.1)$$

```
def DevIndex(newArray, oldArray):
    newArray = newArray.astype(float)
    oldArray = oldArray.astype(float)
    DI = np.sum(abs(newArray-oldArray)/oldArray)/np.size(newArray)
    return DI
```

Figure 8. The function used in solving the deviation index.

The second Spectral Quality metric used is the correlation coefficient (CC) This index measures the closeness or similarity between two images. It can vary between -1 to +1. A value close to +1 indicates the images are very similar, while a value close to -1 indicates that they are highly dissimilar.

$$CC = \sum_{i}^{n}\sum_{j}^{m}\frac{(F_k(i,j) - F_k)(M_k(i,j) - M_k)}{\sqrt{\sum_i^n \sum_j^m (F_k(i,j) - F_k)^2}\sqrt{\sum_i^n \sum_j^m (M_k(i,j) - M_k)^2}} \quad (eq.2)$$

```
def CorrCoeff(newArray, oldArray):
    numerator = np.sum((newArray-np.mean(newArray))*(oldArray-np.mean(oldArray)))
    denominator = m.sqrt(np.sum((newArray-np.mean(newArray))**2)*np.sum((oldArray-np.mean(oldArray))**2))
    cc = numerator/denominator
    return cc
```

Figure 9. The function used in solving the correlation coefficient

The third index used is the signal to noise ratio. The concept in this index is that the original image is considered the signal, and as the image is processed in utilizing different fusion algorithms, noises are added which cause distortion in the spectral quality of the images.

$$SNR_k = \sqrt{\frac{\sum_i^n \sum_j^m (F_k(i,j))^2}{\sum_i^n \sum_j^m (F_k(i,j) - M_k(i,j))^2}} \quad (eq.3)$$

```
def SignalNoise(newArray, oldArray):
    newArray = newArray.astype(float)
    oldArray = oldArray.astype(float)

    numerator = np.sum(newArray**2)
    denominator = np.sum((newArray-oldArray)**2)
    snr = m.sqrt(numerator/denominator)
    return snr
```

Figure 10. The function used in solving the signal to noise ratio

For measuring the spatial quality of the images, the only metric used is the high pass deviation index (HPDI). HPDI is a quality metric to measure the amount of edge information from the panchromatic image transferred to the fused image. HPDI wants to extract the high frequency components of the panchromatic band and each fused bands. The deviation index would indicate how much spatial information from the panchromatic image has been incorporated into the fused image. The smaller HPDI, the better image quality.

$$HPDI_k = \frac{1}{nm}\sum_{i}^{n}\sum_{j}^{m}\frac{|F_k(i,j) - P(i,j)|}{P(i,j)} \quad (eq. 4)$$

```
def HPDI(newArray, panArray):
    newArray = newArray.astype(float)
    panArray = panArray.astype(float)

    hpdi = np.sum(abs(newArray - panArray)/panArray)/np.size(newArray)
    return hpdi
```

Figure 11. The function used in solving the high pass deviation index

3.3 Modules and Libraries used

The program is comprised of three scripts. The first script, the pansharpen, contains the functions needed to fuse the panchromatic and the multispectral image. The packages needed for it to run are gdal (for raster processing), os (for reading, writing, and manipulating the path of data needed), numpy (for array processing), math (for mathematical functions), and pywt (for decomposing and reconstructing arrays using wavelet transform).

The second module, which is the comparison.py, contains the functions for solving the spectral and spatial quality metrics in comparing the fused images with the original image. It only needs the numpy library for array operations.

The third module contains the code for GUI construction. It generally needs the built-in GUI library, Tkinter, and its other modules, tkMessageBox (for prompting error messages) and tkFileDialog (for accepting file paths). The GUI.py script is the main script which interacts with the user for fusing images. This script imports the other two scripts earlier mentioned.

4. RESULTS AND DISCUSSION

Initially running the GUI.py script, a window appears to the screen which prompts for the user to choose either to fuse images or to compare a fused image with the original panchromatic and multispectral images.

Figure 12. The main frame

The main frame lets the user choose whether to fuse images or to compare images. The following images are the fusion frame and the comparison frame. Clicking the buttons will prompt a file dialog window. This will let the user to open the files needed to import.

Figure 13. The fusion frame

Figure 14. The comparison frame

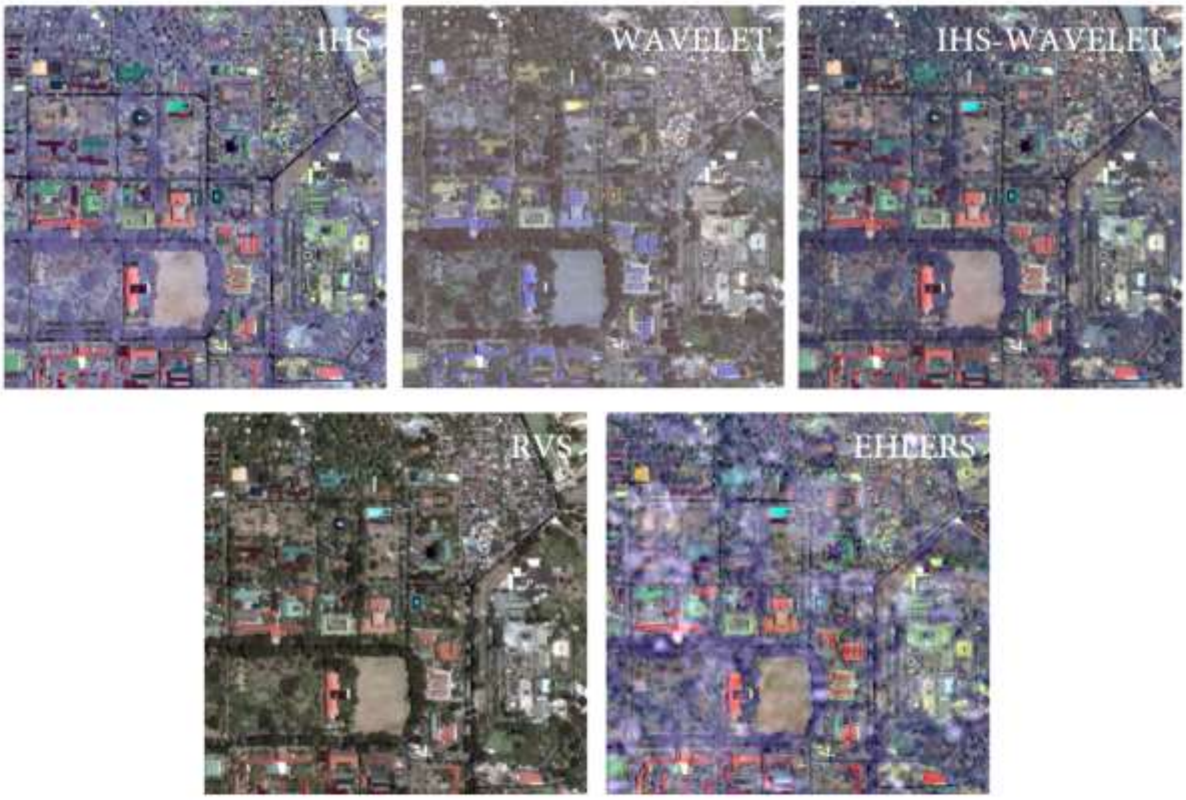

Figure 15. The resulting images from different pansharpening methods

Utilizing the comparison frame, we input the pansharpened image and both the original multispectral and panchromatic images. The metric values are then showed in a message box as shown in Figure 16.

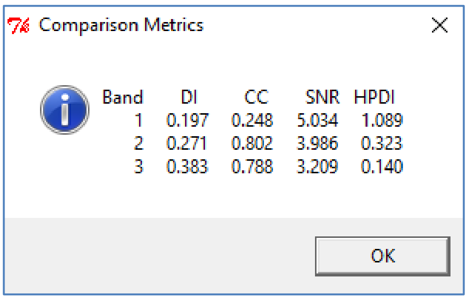

Figure 16. The message box for quality metrics of the IHS-Wavelet Transform

5. CONCLUSION AND RECOMMENDATION

Image fusion is a very important tool in remote sensing but not all recently developed fusion methods are available in image processing software. Each of these techniques offer advantages and disadvantages caused by the different algorithms employed.

The need for automation of these fusion methods can be addressed using Python programming language. Its wide range of libraries, especially those dedicated in scientific computation, allow automation of complex processes. In this project, it was proven that Python can be of great help in image processing particularly in pansharpening which is beneficial for land classification, change detection, and other remote sensing applications.

Since most image processing software do not offer automation of these recently developed fusion methods or those that are yet to be developed, it would very helpful to explore and automate these methods using Python.

6. REFERENCES

References from Journals

[1] Al-Wassai, F.A.; Kalyankar, N.V.; Al-Zaky, A.A. Studying Satellite Image Quality Based on the Fusion Techniques. International Journal of Advanced Research in Computer Science. Vol 2 No. 5, Sept-Oct 2011.

[2] Al-Wassai, F.A.; Kalyankar, N.V.; Al-Zaky, A.A. The Statistical methods of Pixel-based image Fusion Techniques. International Journal of Advanced Research in Computer Science. Vol 1 Issue 3, July 2011.

[3] Gonzales R. C, and R. Woods, 1992. Digital Image Procesing. Addison-Wesley Publishing Company.

[4] De Béthume S., F. Muller, and J. P. Donnay, 1998. “Fusion of multi-spectral and panchromatic images by local mean and variance matching filtering techniques”. In: Proceedings of The Second International Conference: Fusion of Earth Data: Merging Point Measurements, Raster Maps and Remotely Sensed Images, Sophia-Antipolis, France, 1998, pp. 31–36.

[5] Zhang, Y.; Hong, G. 2003. An HIS and wavelet integrated approach to improve pan-sharpening visual quality of natural colour IKONOS and QuickBird images.