COMPLEX ENGINEERING PROBLEM, LINEAR ALGEBRA II (ES-304), SATELLITE IMAGE PROCESSING, PRINCIPAL COMPONENT ANALYSIS

Group Members:

SYED MUHAMMAD ASHHAR SHAH
KHIZAR ALI SHAH
ARBAZ KHAN
ISAAM UL HAQ

CEP Project, Satellite Image Processing

Members:

1. Syed Muhammad Ashhar Shah, 2020478
2. Khizar Ali Shah, 2020196
3. Arbaz Khan, 2020076
4. Isaam Ul Haq, 2020490

This paper deals with the process of analyzing a multispectral Landsat image captured by NASA, obtained from a satellite. In today's time we use high quality satellite images to perform various tasks such as tracking and measuring different types of activities performed by humans or natural sources.

Starting with our analysis, multiple bands of an image, taken individually, are brought together into python objects, which are numpy lists, using libraries so they can be used further. Then they are merged to form a single multispectral image. Next, this image is compared with the original image.

After this merging, we move on to the second part of the paper, that is to study the effects of Principle Component Analysis on the image which can be used for a huge range of things such as image compression, image reconstruction, recognition algorithms or efficient data storage. An image is taken, it is separated into its three red, green, and blue channels. On each one, the PCA is applied that found the eigenvectors and eigenvalues of that image around proper axis and compressed the bits around it. This concluded into having a very small image size with a very small amount of detail loss from the image. Same procedure is applied on all the channels, and with different control variables; and the results for each are showed.

Along with applying PCA, the difference between the quality of information in the original red, green, and blue channel of the image and its after-PCA version is compared, which is called the mean square error, and is shown that the error decreases as the control variables change.

Keywords: principal component analysis, image reconstruction, landstat, satellite image, image processing.

I. INTRODUCTION

In this study we investigate the relationship on how variance in the number of principle components of an image or a dataset affects the overall quality of the image. In addition to PCA we also analyze the overall bands of an image that is captured by a satellite.

We also perform various operations on multispectral images such as scaling, and concatenation of the individual bands obtained to form a high-quality visual satellite image.

Furthermore, we also perform error analysis on the use of principle component analysis and observe the relationship of the amount of error against the number of selected principal components of an image due to which we can find out how many principal components are required to display an image without disrupting its originality to a huge extent.

II. EASE OF USE

A. Planet Activity Analysis

The Landsat images obtained from a satellite are high quality, multi spectral images (contain multiple bands) of the earth's surface. Hence using these images, we can perform analysis on the different activities happening anywhere on the planet such as tracking temperature changes or natural disasters.

B. Disaster Predictions and Prevention

Using these satellite images, we can also use powerful machine learning algorithms to predict any natural or man-made disaster that may occur and hence make the necessary preparation in order to avoid such a disaster to cause high intensity catastrophic damage to life or property.

C. Data Interpretation and Simplicity

We also use the technique of principle component analysis to interpret the data to search for any important patterns in a dataset. This technique also helps to make sense of the data we receive from the multispectral landstat images.

III. ADDITIONAL DATA AND INFORMATION

A. Abbreviations and Acronyms

- PCA: Principal Component Analysis
- CEP: Complex Engineering Problem
- NASA: National Aeronautics and Space Administration

B. Units

- We use "pixels" to measure the resolution of an image as the data on which we perform image analysis operations are stored in a two-dimensional array which represent the (x, y) location of a pixel and hence the overall resolution of an image.

C. Equations

$$S = D / 255 \qquad (1)$$
$$X = X - \mu \qquad (2)$$
$$T = (T.\ C) + \mu \qquad (2)$$

The equation (1) represents the standardization equation as the data we have has different pixel values that range from 0-255 which represent the color density of a pixel. We use this equation (1) to scale all the values from 0-1 so that no component is dominant over the other when we apply PCA.

The equation (2) here is used for application of PCA as we must subtract the mean of the data from each entry of the data so that we can calculate the eigenvalues and eigenvectors.

The equation (3) here represents transformation of a PCA applied dataset back to the original dataset format by performing a dot product of the PCA applied data against the selected eigen vectors and adding back the subtracted mean.

D. Some Common Mistakes

- The word "data" is plural, not singular. When we use the word data, we mean a collection of data of an image.
- Here "transformation" means obtain the original dimension of the image so that we can merge them together to form a dimensionally reduced image. Don't confuse it with the data we get by simply fitting PCA.
- When we mention "Standardization" we mean that we can't directly apply PCA to any data. We must observe the data first to make sure that the data is linearly distributed so that we can recover the original data from the transformed data.
- When we say "reconstructing" we also mean reducing the dimensions of the image data while still maintaining the principal properties of the original image hence reducing the image size without much data loss. Don't confuse it with lossless reconstruction.

IV. Methodology

Satellite image analysis and dimensional reduction using PCA, a growing area of interest among individuals nowadays due to its prediction properties and lossy compression of data for efficient storage. A satellite image is basically a multispectral image that contains different bands of an original image captured by a NASA satellite. We also use dimensional reduction using one of the popular techniques known as PCA.

As we know we satellite images are high quality and are highly expensive to compute and store in huge volumes hence we can use the technique of PCA as a way to detect the different patterns in the large and complex data obtained from the satellite images. Hence, majority of data is transmitted into a handful of principle components which convey meaning to the large and complex data, doing this also reduced the number of dimensions of the original image without discarding any of the important principal components hence no significant data loss takes place and at the same time the overall data is reduced.

All the data that we use in this research is collected from NASA's official servers which are open to the public. We only use a single landstat image of Pakistan as our dataset on which various image operations along with PCA is applied for dimension reduction.

The tool of our choice was **Jupyter Lab** and **Python** as is a powerful programming language and has a wide range of libraries both of mathematical computation along with image analysis.

A. Landstat Image Used

Above is the landstat image that we have used. This is an image consisting of majority of Punjab, Pakistan. The Scene ID corresponding to this image is "". In the project we have applied various operations of band concatenation and principal component analysis on this image.

B. *Individual bands of an image.*

```
['LC08_L1TP_149038_20211218_20211223_01_T1_B1.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B10.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B11.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B2.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B3.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B4.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B5.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B6.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B7.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B8.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B9.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_BQA.TIF']


['LC08_L1TP_149038_20211218_20211223_01_T1_B1.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B2.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B3.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B4.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B5.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B6.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B7.TIF',
 'LC08_L1TP_149038_20211218_20211223_01_T1_B9.TIF']
```

Here is a list of all the bands that we shall use for concatenation in an attempt to form the original-colored image and perform basic image related operation on it. Above is a list of all the bands that we have collected from NASA's website and below is a list of the bands which we shall use for concatenation. We have ignored some of the bands due to the difference in dimensional values of individual bands. The selected bands all have dimensions of 30 whereas the others differ from it.

C. *Principle Component Analysis Function*

```python
def pca(X, comps):
    # get all the dimensions of the matrix
    n_samples, n_features = X.shape
    n_components = min(X.shape)

    # find and subtract the mean
    mu = np.mean(X, axis=0)
    X = X - mu
    # find the eigen values and eigen vectors
    U, S, Vt = linalg.svd(X, full_matrices=False)
    # invert any negative values
    U, Vt = svd_flip(U, Vt)
    # compute the varience
    var = (S**2) / (n_samples - 1)
    # select the components we want to use (pca value)
    components_ = Vt
    components_ = components_[:comps]
    # select the eigen vectors we want to use
    U = U[:, :comps]
    # perform transformation
    U *= S[:comps]
    # inverse transform back into the orignal dimensions
    U = np.dot(U, components_) + mu
    return U
```

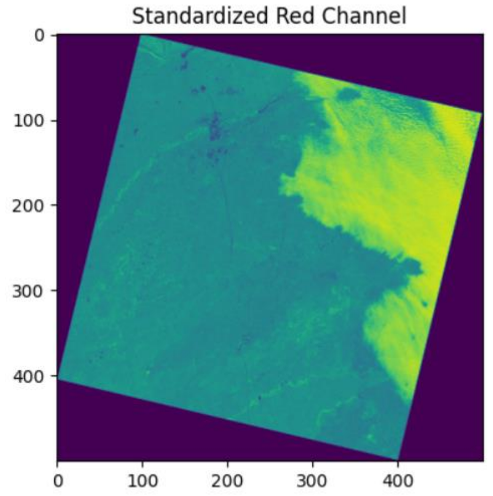
Above we have the python function that we have coded for performing the task of PCA. In the function we have used the scikit-learn SVD and the NumPy library to compute and find the eigen vectors and the eigen values along with performing various matrix related operation to receive a dimensionally reduced image.

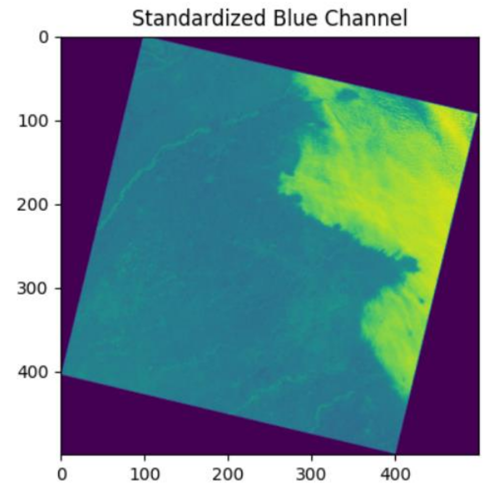
D. *Mean Squared Error Function*

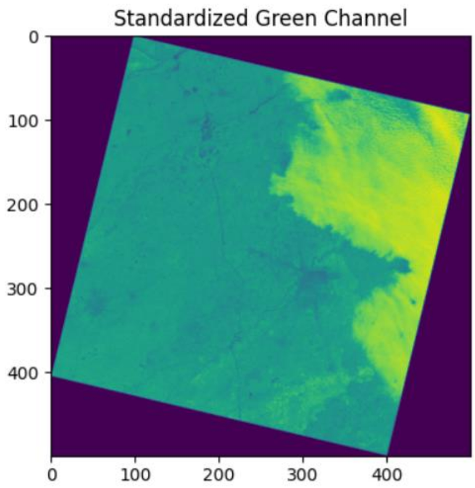
```python
def mse(img1, img2):
    err = np.square(np.subtract(img1,img2)).mean()
    return err
```

This is an illustration of the mean squared error function that we have developed for our project that takes two images in the form of lists/arrays and then uses the NumPy library to calculate the mean squared error between the two image where one of the images is the original image that is untouched, PCA not applied, while the other is the one on which we have applied PCA. Furthermore, we will compare the channels with each other rather than the RGB images to get an accurate approximation of the error between the two.

E. *Channels of the image.*







Here you can see the three channels, Red, Blue and Green, of the colored image on which we shall apply PCA for our selected value of components. We will first apply PCA to these individual channels/bands and then concatenate them together to form a dimensionally reduced image similar to the original image but with the number of principle components according to our choice.

V. Task Distribution

First, Iasaam used the Internet and went to the NASA's website to download the Landsat images. There were twelve bands of an image, and the table showed their resolutions along with other information. Taking a closer look at resolutions showed that four of the bands had a different resolution while the other eight had the same. This way, all these bands cannot be stored in a NumPy array properly, and doing operations on them would have caused issues, so I deleted these four.

The remaining eight images were then loaded onto the python program using the library CV2. After loading or opening them, they are stored in the *bands* list. For visual analysis, all of them were plotted on the screen individually.

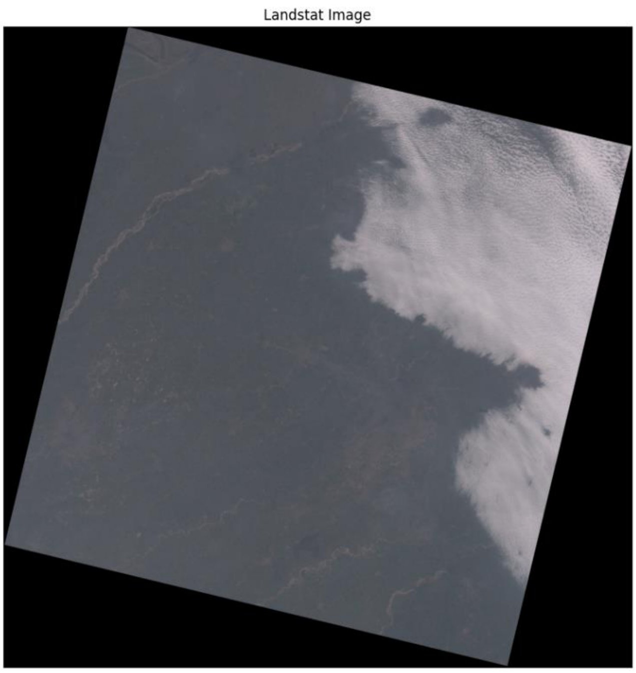
The next task was done by Khizar and was to merge these eight bands and form a single colorful image out of these. For that purpose, the way *bands* list contains the bands is not suitable. It has a RGB coordinate of 3 for every band, and if it is merged, they will become 27. Avoiding this, we used the xarray and earthpy library to create another bands list, but it is squeezed in such a way that it doesn't have the RGB coordinate, and only the dimensions. This new list also contains the labels with the bands as a two-dimensional coordinate of the list. Then this single xarray-object is merged, and the image is plotted onto the screen. This image is then compared with the original one for a visual comparison. Each of the images are then resized, cropped, and then plotted

Next comes the actual Principal Component Analysis. The image to be used is first loaded and converted to a NumPy list which represents a two-dimensional matrix. After plotting, the three channels, red, green, and blue, are separated from the image. The dimensions of each are reduced to 500x500 because PCA will take forever if applied on a large-resolution matrix. Among these three channels, one may be dominant than the other, for which all of them are standardized/normalized to a value between 0 and 1. This task was done by Ashhar who was in charge of coding and applying the PCA functionality.

To these normalized matrices, the PCA function is applied individually, which finds the covariance matrix that has the eigenvectors and the eigenvalues, and then uses those to find new dimensions of original points by transforming them. Same task was done using different PCA values, and against each one, an error, called the mean square error, was calculated. The error analysis was done on each channel, each time with a different PCA value. The function that was developed for calculating the error against the original image and the PCA applied image was done by Arbaz.

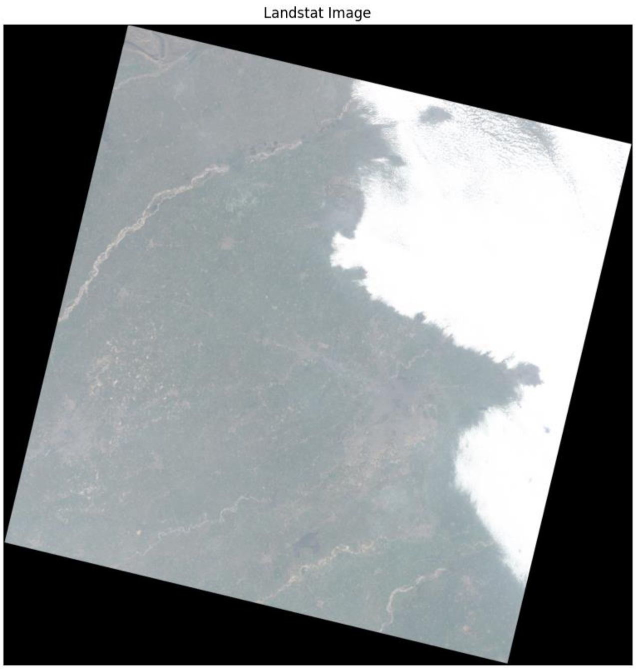
Lastly, we had to write the project report to sum up what we had done, and this was divided among all the group members into multiple portions equally.

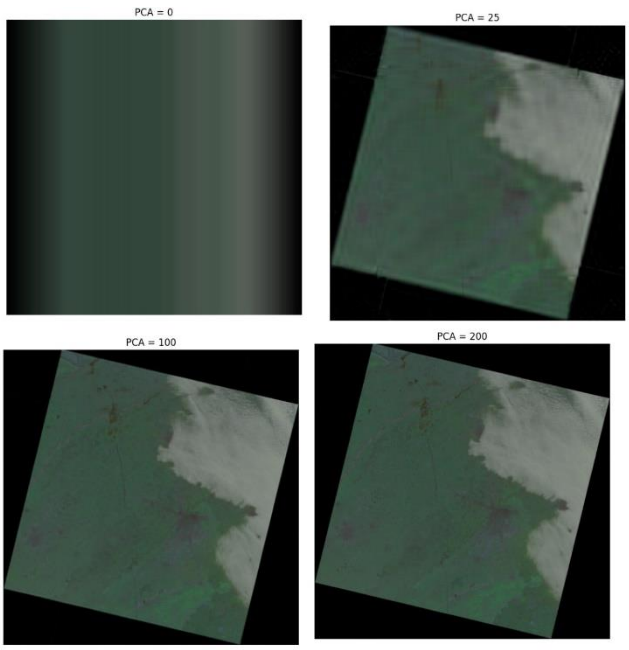
VI. Results



The image above is formed when we concatenate the bands which we have talked about previously. This image has a high contrast as compared to the original image. That is since we have ignored some of the bands due to their difference in dimensions.

The image below shows the concatenated image on which we have applied a linear stretch. By stretching the image, we observe that the contrast of the image is reduced and we can see all the visual aspects of it more clearly and compare it with the original image.





Above we also have a collection of images on which we have applied different values of PCA and have gotten images with different number of principle components. Looking at these pictures we can see that with the greater the number of principal components the clearer image returned.

In the figure attached below we also see that as we increase the number of principle components for each separate image channel and then use the developed MSE function to compute the error between it and the original channel on which there is no PCA applied, the mean squared error reduces on each increment. This shows that by increasing the number of principle components we get an image that is visually closer to the original image.

```
For PCA Value of 0
Error for Red Channel: 0.019760469894594384
Error for Blue Channel: 0.019760469894594384
Error for Green Channel: 0.019760469894594384


For PCA Value of 25
Error for Red Channel: 0.0005981363540817502
Error for Blue Channel: 0.0005981363540817502
Error for Green Channel: 0.0005981363540817502


For PCA Value of 50
Error for Red Channel: 0.0002905208820516286
Error for Blue Channel: 0.0002905208820516286
Error for Green Channel: 0.0002905208820516286


For PCA Value of 75
Error for Red Channel: 0.00018124409973337895
Error for Blue Channel: 0.00018124409973337895
Error for Green Channel: 0.00018124409973337895


For PCA Value of 100
Error for Red Channel: 0.0001231695672214768
Error for Blue Channel: 0.0001231695672214768
Error for Green Channel: 0.0001231695672214768
```

CONCLUSION

From here we see that PCA in its standard form is a widely used form of data compression technique and is beneficial for data analysis as it can be used to describe the data. We also observe that there are also various methods that correspond to data analysis and may only be loosely connected to PCA, but just like PCA they all share a common approach that is factorial decomposition of certain matrices. The use of PCA is vast and can span multiple disciplines other than just being used for satellite image processing and data compression. As time goes on there may be significant improvement and new adaptations built by using basic PCA functionality for complex methods but who can really tell.

REFERENCES

[1] https://numpy.org/doc/stable/reference/
[2] https://matplotlib.org/stable/api/index.html
[3] https://github.com/topics/cv2-library
[4] https://docs.xarray.dev/en/stable/api.html
[5] https://keras.io
[6] https://en.wikipedia.org/wiki/Principal_component_analysis
[7] https://www.nature.com/articles/nmeth.4346
[8] https://en.wikipedia.org/wiki/Mean_squared_error
[9] https://www.nasa.gov/multimedia/imagegallery/index.html
[10] https://en.wikipedia.org/wiki/Multispectral_imaging