Basic Steganography

Hiding a given text in an image is called Steganography. There are many ways to accomplish this task, one of which is presented in this project.

The project starts from reading an image, using **pyplot**. Note that a colored image is a three dimensional array, which is composed of three 2-D arrays (r, g and b).

The next step converts the image into grayscale using the **to_gray** method. Note that the converted image is a 2-D image and has values between 0 and 255.

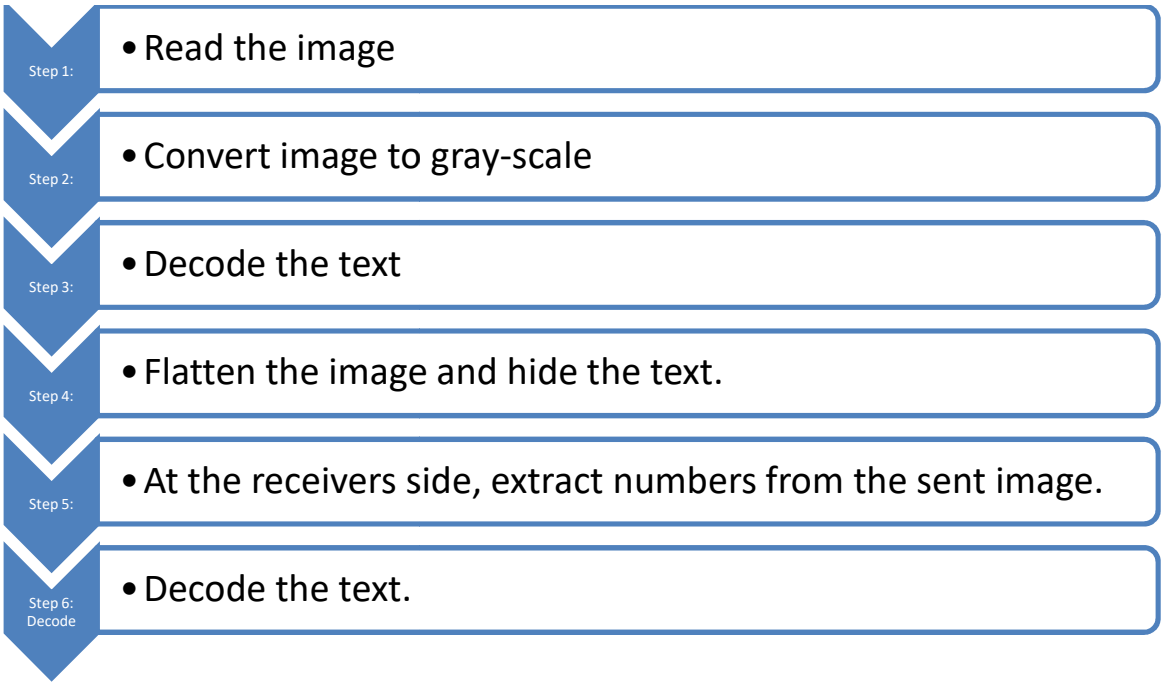
Now, we will ask the user to enter a text and store it in a variable called **Text**.

The further action of course is slightly involved. It requires the following

1. For each word in the text, each character in the word is converted into ASCII value. The key (shared by the sender and the receiver) is then added to the ASCII value. The final number is appended to a list.
2. **Flattening the image**: The two dimensional image is converted into a one dimensional image, using the **reshape** method of **numpy**.
3. Now, random numbers (equal to the number of elements in the list created in the previous step) are generated (**Positions**) and the corresponding values in the flattened image is replaced by the elements of the list.
4. The array so formed is again converted into original shape.
5. The **Positions** array and the image obtained in the previous step are sent to the receiver.
6. At the receiver's side, the received image is flattened.
7. At the locations indicated by the Positions, each value of replaced by the difference of the value and the key. The integers so obtained are converted to characters and appended to the decoded text.

Flow of the project

The following image shows the flow of the project.



Functions used in the project

Function	Input	Output	Purpose
to_gray	Image	Image	Converts the colored image to a gray scale image
encode_text	Text	List of integers	Stated in the abstract
decode	List, key	Text	Stated in the abstract

In built function used in the project

Method	Purpose
matplotlib.pyplot.imread	Reads an image into an array
matplotlib.pyplot.imshow	Displays the image
np.array	Creates a **numpy** array
np.max	Finds maximum value from a **numpy** array
np.max	Finds minimum value from a **numpy** array
ord	Finds the ASCII value of a character
chr	Finds the character corresponding to the ASCII value

References (For the report)

Definition of Stagnography, techniques and applications: https://en.wikipedia.org/wiki/Steganography

1. Detailed study: http://sharif.edu/~kharrazi/pubs/ims04.pdf
2. Imread: https://matplotlib.org/3.3.1/api/_as_gen/matplotlib.pyplot.imread.html
3. Imshow: https://matplotlib.org/3.3.1/api/_as_gen/matplotlib.pyplot.imshow.html
4. np.array: https://numpy.org/doc/stable/reference/generated/numpy.array.html
5. np.max: https://numpy.org/doc/stable/reference/generated/numpy.maximum.html
6. np.min: https://numpy.org/doc/stable/reference/generated/numpy.minimum.html
7. ord: https://www.programiz.com/python-programming/methods/built-in/ord

For details regarding **numpy** refer to **Information Practices** by Harsh Bhasin, New Age International or **Computer Science with Python** by Harsh Bhasin, New Age International.