

Read me

Steganography: Hiding text inside another

To run the .ipynb you need to install Jupyter notebook or can run the code in google collab

https://colab.research.google.com/?utm_source=scs-index

To run the code you need to install

1. OpenCv
2. Matplotlib
3. Numpy

To install OpenCv run “pip install opencv-python”.

To install Matplotlib run “pip install matplotlib”.

To install Numpy run “pip install numpy”.

```
import cv2
from matplotlib import pyplot as plt
from math import log10, sqrt
import numpy as np
```

Import these to the program

Now each cell has to perform particular task as below:

STEPS:

1. **Import all the requires modules**
2. Take input of the image
3. `img1 = cv2.imread("Cover_1.png")`
4. `temp= cv2.imread("Cover_1.png")`

5. **Resize the image to standard size**

```
img1=cv2.resize(img1, (256,256))
temp=cv2.resize(temp, (256,256))
row,col,dim=temp.shape
print(row,col,dim)
```

6. Choose a key and message

```
7. while(1):
8.     key=input("Enter the security key : \n")
9.     if len(key)==0:
10.         print("Key size cannot be 0 .. \nReEnter Key")
11.     else:
12.         break
13.     message=input("\nEnter the message to be hidden : \n")
14.     key_indx=0
15.     message_len=len(message)
```

16. Perform Encryption algorithm on the image and key

```
l=len(message)

for i in range(l):
    img1[r,c,d]=dict1[message[i]]^dict1[key[key_indx]]
    r=(r+1)
    c=(c+1)
    d=(d+1)%3
    key_indx=(key_indx+1)%len(key)
```

17. Save the encrypted image

```
18.     cv2.imwrite("enc_img.png",img1)
19.     print("The data is hided and encrypted image is saved
    successfully")
```

20. Perform Decryption Algorithm of the image and key

```
key1=input("\nReEnter key to decrypt : ")
decrypt=""
if(key==key1):
    for i in range(1): #

        decrypt+=dict2[img1[r,c,d]^dict1[key[key_indx]]]
        r=(r+1)
        c=(c+1)
        d=(d+1)%3
        key_indx=(key_indx+1)%len(key)
    print("Message decrypted Successfully")
    print("Message was: ",decrypt)
else:
    print("Invalid Key")
```

21. If the key entered is correct the message will be retrieved

```
if key==key1:
    print("Encrypted Image")
    x= cv2.imread("enc_img.png")
    plt.imshow(cv2.cvtColor(x, cv2.COLOR_BGR2RGB))
    plt.show()

    print("Decrypted Image")
    plt.imshow(cv2.cvtColor(temp, cv2.COLOR_BGR2RGB))
    plt.show()
else:
    print("INPUT IMAGE")
    plt.imshow(cv2.cvtColor(img1, cv2.COLOR_BGR2RGB))
    plt.show()
    print("Exiting....")
```

22. Finally calculate the PSNR value by comparing original image and saved encrypted image.

