Exp no: 8

Date: 08/10/2025

Image Steganography

Aim

To hide a secret message within an image in such a way that others cannot discern the presence of the hidden message.

Algorithm:

1. Start
2. Input: cover image, secret message, and secret key
3. convert the secret message into binary/text form
4. Use the secret key to encode each character of the message into image pixels using XOR operation.
5. Save the resulting stego Image
6. To extract the message, use the same key to reverse the XOR process
7. Output: Display the extracted text.
8. End.

Python code:

```
import cv2
import os
d = {}
c = {}
for i in range(255):
    d[chr(i)] = i
    c[i] = chr(i)

x = cv2.imread(r"c:\Users\Joel\Downloads\computer.webp")
i = x.shape[0]
j = x.shape[1]
print("Image Dimensions:", i, "x", j)

key = input("Enter key to edit (security key): ")
text = input("Enter text to hide: ")
kl = 0
z = 0
n = 0
m = 0
l = len(text)
```

```
for i in range(l):
    x[n,m,z] = d[text[i]] ^ d[key[kl]]
    n= (n+1) % x.shape[0]
    m= (m+1) % x.shape[1]
    kl = (kl+1) % len(key)

cv2.imwrite("encrypted_img.jpg",x)
os.startfile("encrypted_img.jpg")
print("Data hiding in image completed successfully")

ch= int(input("\n Enter 1 to extract data from image: "))
if ch==1:
    key1 = input("\n Re-enter key to extract text: ")
    decrypt = ""
    if key==key1:
        kl=0
        z=0
        n=0
        m=0
        for i in range(l):
            decrypt+= c[x[n,m,z] ^ d[key[kl]]]
            n= (n+1) % x.shape[0]
            m= (m+1)% x.shape[1]
            kl = (kl+1) % len(key)
        print("\n Decrypted text is:",decrypt)
    else:
        print(" Key doesn't match. Access denied")
else
    print("Existing program")
```

Result:

Thus, Hided a secret message within an image . that others can't discern the presence of hidden message.