Proiect

Rezultate exercitiul 1

Setup:

* host: imaginea originala color in care se insereaza logo, dimensiune 512x512,
* logo: imagine data 32x32

O imagine care conține desen, tablou, artă, Chip de om

Descriere generată automat

Figure 1 host.png



Figure 2 logo.png

Initial: BER=0.0791, PSNR = 23.3137

PSNR: o valoare mai mare a PSNR indică o calitate mai bună a semnalului după atac

BER: rata de detecție a logo-ului - o valoare mai mică a BER indică o performanță mai bună.

Table 1 Insertia robusta de date

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipul de atac** | **BER – cod current (3,3)-(3,4)** | **PSNR – cod current (3,3)-(3,4)** | **BER – coordonate alese (4,5)-(5,5)** | **PSNR – coordonate alese (4,5)-(5,5)** |
| 1. **Schimbare luminozitate** | 0.1796 | 10.14471 | 0.1796 | 10.14471 |
| 1. **Adaugare zgomot gaussian** | 0.1474 | 15.6331 | 0.1474 | 15.6331 |
| 1. **Adaugare zgomot sare si piper** | 0.2275 | 18.8982 | 0.2275 | 18.8982 |
| 1. **Filtrare cu filtru median** | 0.18554 | 28.5710 | 0.18554 | 28.5710 |
| 1. **Rotire 90 de grade** | 0.5341 | 9.3522 | 0.5341 | 9.3522 |
| 1. **Compresie jpeg** | 0.0791 | 23.2353 | 0.0791 | 23.2353 |

Pentru a lucra cu zgomotul produs in atacurile b) si c) ce presupun generare aleatorie de numere, am utilizat un seed static (42) pentru a putea reproduce rezultatele.

Observam ca pentru atacurile ilustrate valorile BER si PSNR nu se modifica odată ce schimbam coordonatele (atâta timp cat seed-ul pentru zgomot este setat la valoarea statica). Așadar, indiferent de coordonatele alese (3,3)-(3,4) vs. (4,5)-(5,5), rezultatele pentru imaginile host.png si logo.png sunt aceleași.

Analizând rezultatele prezentate în tabelul 1, se pot trage următoarele concluzii:

1. **Performanța la schimbarea luminozității (a)**:
   * BER rămâne constant la 0.1796, iar PSNR scade la 10.14471. Acest lucru indică o rezistență moderată a codului la acest tip de atac, însă calitatea semnalului este afectată semnificativ.
2. **Rezistența la adăugarea de zgomot gaussian (b)**:
   * Cu un BER de 0.1474 și un PSNR de 15.6331, codul se dovedește mai rezistent față de atacul cu zgomot gaussian, menținând o calitate mai bună a semnalului.
3. **Adăugarea zgomotului sare și piper (c)**:
   * BER este cel mai mare în acest caz (0.2275), iar PSNR are o valoare de 18.8982. Aceasta indică faptul că atacul zgomotului sare și piper afectează cel mai mult acuratețea transmiterii informației.
4. **Filtrarea cu filtru median (d)**:
   * Atacul cu filtru median prezintă o performanță bună, cu BER la 0.18554 și un PSNR de 28.5710, cea mai mare valoare din tabel. Aceasta indică o robusteză excelentă a codului la acest tip de atac.
5. **Rotirea imaginii cu 90 de grade (e)**:
   * BER crește semnificativ la 0.5341, iar PSNR scade la 9.3522, ceea ce arată o vulnerabilitate ridicată a codului la această transformare geometrică.
6. **Compresia JPEG (f)**:
   * Codul demonstrează cea mai bună rezistență la compresia JPEG, cu cel mai mic BER (0.0791) și un PSNR relativ ridicat de 23.2353, indicând faptul că informația este păstrată bine în condiții de compresie.

Putem concluziona că identificăm o rezistență bună la atacuri de tip compresie JPEG și adăugarea de zgomot gaussian, dar sunt vulnerabilități la transformări geometrice, cum ar fi rotirea cu 90 de grade. Rezultatele PSNR sugerează că atacurile de tip filtrare mediană și zgomot sare și piper păstrează o calitate relativ bună a semnalului, însă BER indică faptul că informația poate fi mai afectată în aceste cazuri.

Rezultatele obținute din rularea scriptului fără atacurile a),b),c),d),e) sau f) sunt ilustrate in următoarele figuri alb-negru (binare):

O imagine care conține text, schiță, Chip de om, lanț

Descriere generată automat

O imagine care conține diagramă, captură de ecran, Font, pixel

Descriere generată automat

Exemplificarea rezultatelor intermediare pentru fiecare tip de atac sunt ilustrate in Tabelul 2 de mai jos.

Table 2 Ilustrarea rezultatelor a)-f)

|  |  |  |
| --- | --- | --- |
| **Atac** | **Imagine host transformată** | **Imagine logo decodificată** |
| **a)Schimbare luminozitate** |  |  |
| **b)Adaugare zgomot gaussian** |  |  |
| **c)Adaugare zgomot sare si piper** |  |  |
| **d)Filtrare cu filtru median** |  |  |
| **e)Rotire 90 de grade** |  |  |
| **f)Compresie jpeg** |  |  |
| Rezultat Marcat DCT  Atacul f) | O imagine care conține Chip de om, schiță, desen, tablou  Descriere generată automat | |

Rezultate exercitiul 2

**Plan de acțiune:**

1. Am modificat și rulat codul pentru a testa inserția pentru N=1000,2000,3000,4000
2. Am salvat imaginea gazdă și histograma.
3. Am documentat rezultatele într-un tabel.

O imagine care conține captură de ecran, diagramă, text, Interval

Descriere generată automat

Figure 3 Histograma imaginii gazda

Pentru generarea mesajului am modificat range-ul de numere intreg ice pot fi generate, astfel incat sa se poata genera si valori de 1 (se regaseste in scriptul givenCode\_ex2.py, functia obtine\_rezultate()):

*mesaj=np.random.randint(0,2,N)*

Conform histogramei imaginii gazdă, cele mai bune valori pentru l și r se situează în intervalul 50-70, fapt demonstrat și de rezultatele din tabelul 3. Pentru alte valori, precum l=155, r=156 spatiul de codificare a imaginii este insuficient pentru N>=2000, iar rata de decodificare implicit foarte mare.

*Din testele realizate, cei mai buni coeficienti l si r sunt:*

* *Pentru N=1000: L=55,R=56*
* *Pentru N=2000: L=58,R=59*
* *Pentru N=3000: L=60, R=61*
* *Pentru N=4000: L=60, R=61*

Analiza PSNR pentru valori de 1000, 2000, 3000 și 4000 biti inserati în imagine. Rezultatele (inclusiv erorile recuperate) se regasesc in tabelul 3, unde au fost marcate cu portocaliu cele mai bune solutii:

Table 3 Rezultate insertie reversibila de date

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N | L | r | PSNR imagine marcata | PSNR imagine recuperata | Erori mesaj recuperate | Probleme spatiu |
| 1000 | 60 | 61 | 58.1701 | inf | 0.0 | \_ |
| 2000 | 60 | 61 | 55.5432 | inf | 0.0 | \_\_ |
| 3000 | 60 | 61 | 52.5952 | inf | 0.0 | \_\_ |
| 4000 | 60 | 61 | 50.2909 | inf | 0.0 | \_\_ |
| 1000 | 58 | 59 | 59.4140 | inf | 0.0 |  |
| 2000 | 58 | 59 | 56.1436 | inf | 0.0 |  |
| 3000 | 58 | 59 | 51.1250 | inf | 0.0 |  |
| 4000 | 58 | 59 | 49.6613 | inf | 0.0 |  |
| 1000 | 55 | 56 | 60.35084 | inf | 0.0 | \_ |
| 2000 | 55 | 56 | 55.50261 | inf | 0.0 | \_\_ |
| 3000 | 55 | 56 | 52.15619 | inf | 0.0 | \_\_ |
| 4000 | 55 | 56 | 49.87093 | inf | 0.0 | \_ |
| 1000 | 155 | 156 | 50.18753 | inf | 0.0 | \_ |
| 2000 | 155 | 156 | 48.14249 | inf | 297.0 | Spatiu insufficient |
| 3000 | 155 | 156 | 48.14289 | inf | 820.0 | Spatiu insufficient |
| 4000 | 155 | 156 | 48.14288 | inf | 1319.0 | Spatiu insufficient |
| 500 | 155 | 156 | 51.99622 | inf | 0.0 | \_ |