

SCC251 - Processamento de Imagens
2018

Projeto Final
Esteganografia BPCS

Relatório

Guilherme dos Santos Marcon
NUSP 9293564

Definições

- Imagem recipiente é a que esconderá a outra imagem.
- Imagem alvo é a que será escondida.
- Bloco é uma matriz 8x8 de números de 8 bits da imagem recipiente.
- Plano é uma matriz 8x8 de números binários.
- Complexo é quando um plano possui alternâncias 0-1 suficientes para ser classificado como um plano ruidoso e não um plano informativo.
- Conjugado é quando um plano não é complexo, então é realizada uma operação XOR bit a bit do plano com uma matriz padrão, deixando o plano complexo.

Métodos

O básico do método BPCS é:

- Inserir a imagem alvo nos planos complexos da imagem recipiente.

A partir daí, as melhorias e resoluções de outros problemas que surgem foram:

- Transformar a imagem recipiente de Pure Binary Code para Canonical Gray Code, fazendo com que a inserção da imagem alvo seja ainda menos perceptível.

Uma explicação simples e rápida: Pure Binary Code é a codificação normal de bits, cada bit representando 2 elevado à algo, Canonical Gray Code é uma codificação de mudança, existe um bit inicial de parâmetro e os próximos são comparados com os anteriores, codificando 0 se o bit é igual ou 1 se o bit é diferente.

<http://datahide.org/BPCSe/pbc-vs-cgc-e.html>

- Segmentar e percorrer as imagens de uma maneira mais eficiente, foi decidido que a imagem recipiente não será segmentada previamente em planos complexos/não-complexos, como cada plano só é percorrido uma única vez, essa comparação é realizada quando aquele plano estiver sendo verificado, isso pode fazer com que, caso a imagem recipiente possua muitos planos, ela só verifique o menor número de planos necessários para inserir a imagem alvo.
- Utilizando as Classes de Python para melhor entendimento e clareza do código.
- Inserir nos primeiros 2 blocos complexos o tamanho da imagem alvo.
- Marcar os planos que foram conjugados em um Conjugation Map, que é apenas uma matriz binária, e o inserir depois da imagem alvo também nos planos complexos, tendo cuidado de que cada plano desse mapa também seja complexo.

Resultados preliminares

A inserção se provou complicada, mas os primeiros resultados foram:

- Test1
 - Recipiente: mini-lake.png
 - Dimensões: 1374x917
 - Alvo: mini-lion.png
 - Dimensões: 480x373
 - Resultado:
 - Tempo: 4 segundos
 - RMSE: 2.31
 - Porcentagem inserida: 14%
 - Mudança perceptível visualmente.
- Test2
 - Recipiente: reduced-colorful.smoke.png
 - Dimensões: 625x435
 - Alvo: reduced-lion.png
 - Dimensões: 480x373
 - Resultado:
 - Erro - Espaço insuficiente.
- Test3
 - Recipiente: original-mountain.png
 - Dimensões: 5493x3666
 - Alvo: original-lion.png
 - Dimensões: 1920x1491
 - Resultado:
 - Tempo: 272 segundos
 - RMSE: 1.86
 - Porcentagem inserida: 12%
 - Mudança imperceptível visualmente.
- Test4
 - Recipiente: reduced-lion.png
 - Dimensões: 480x373
 - Alvo: mini-lion.png
 - Dimensões: 192x150
 - Resultado:
 - Tempo: 4 segundos
 - RMSE: 2.7
 - Porcentagem inserida: 18%
 - Mudança perceptível visualmente.
- Test5
 - Recipiente: reduced-lion.png
 - Alvo: halfed-reduced-lion.png
 - Resultado:

- Tempo: 9 segundos
- RMSE: 13.8
- Porcentagem inserida: 40%
- Mudança perceptível visualmente.

Próximos passos

- Implementar o método de recuperar a imagem inserida.
- Melhorar e otimizar partes do código.
- Testar com imagens variadas.