

MC920 - Trabalho 4

Nome: Felipe Escórcio de Sousa - **RA:** 171043

Dezembro 2020

1 Introdução

Este trabalho foi elaborado com o objetivo de implementar um algoritmo de esteganografia em imagens coloridas.

A técnica de esteganografia se resume em ocultar alguma informação nas camadas menos significativas dos pixels de uma imagem, de forma que se possa passar uma mensagem secreta de maneira discreta, pois a alteração desses bits cria mudanças de difícil percepção ao olho humano em relação à imagem sem mensagem.

2 Execução do código

O script foi executado usando-se Python 3.8.5, usando-se as bibliotecas NumPy e OpenCV2, executando-se o arquivo principal lab04.py pelo interpretador, como no exemplo:

```
python3 lab04.py -i input.png -o output.png -t text.txt  
-b 0 -c -s
```

ou também:

```
python3 lab04.py -i input.png -t text.txt -b 0 --decode
```

E os argumentos possíveis são

- input-file - caminho para a imagem png de entrada.
- output-file - caminho da imagem de saída pretendido.

- `textfile` - caminho do arquivo de texto de entrada ou de saída, dependendo da opção escolhida. Caso seja codificar, também é aceito input de texto do terminal. Caso decodificar e não for inserido caminho para arquivo, apenas imprime a mensagem.
- `code` - opção para codificação de texto
- `decode` - opção para extração de texto
- `bit-plan` - o plano de bit pretendido para se codificar ou decodificar mensagem
- `separate` - opção para separação de camada para fim de visualização, apenas para codificação.

como também a primeira letra de cada um dos parâmetros é aceita.

3 Entradas

Foram usados como textos de entradas alguns textos Lorem Ipsum de poucos parágrafos gerados em ferramentas online, como também alguns livros de domínio público como Dom Casmurro, Memórias Póstumas de Brás Cubas, A Metamorfose e Dom Quixote de La Mancha em arquivos planos de texto.

Resultados:

Foi possível com sucesso esconder e reaver mensagens curtas e também alguns capítulos dos livros usados (variando o tamanho dos textos que foram possíveis de esconder dependendo do tamanho da imagem).

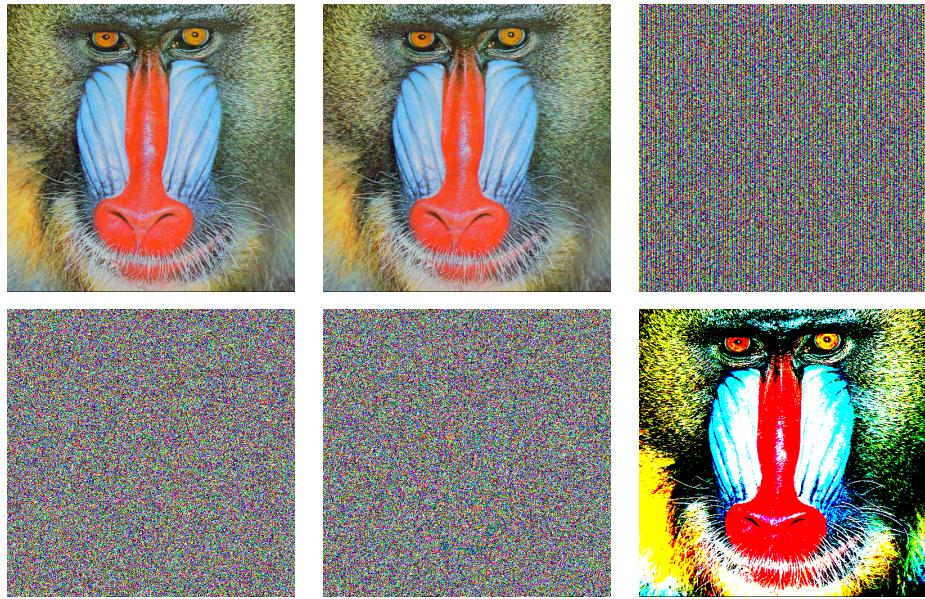


Figura 1: De cima para baixo e da esquerda pra direita, a imagem original do mandril, imagem do mandril com alguns capítulos de Dom Casmurro no bit 0, camada de bit 0 isolada, camada de bit 1 isolada, camada de bit 2 isolada e camada 7 isolada.

Para ilustrar como a imagem pode se tornar um pouco mais ruidoza ou mais diferenciada em relação à original, também foi testado com uma imagem maior, mais clara e em bits mais significativos.

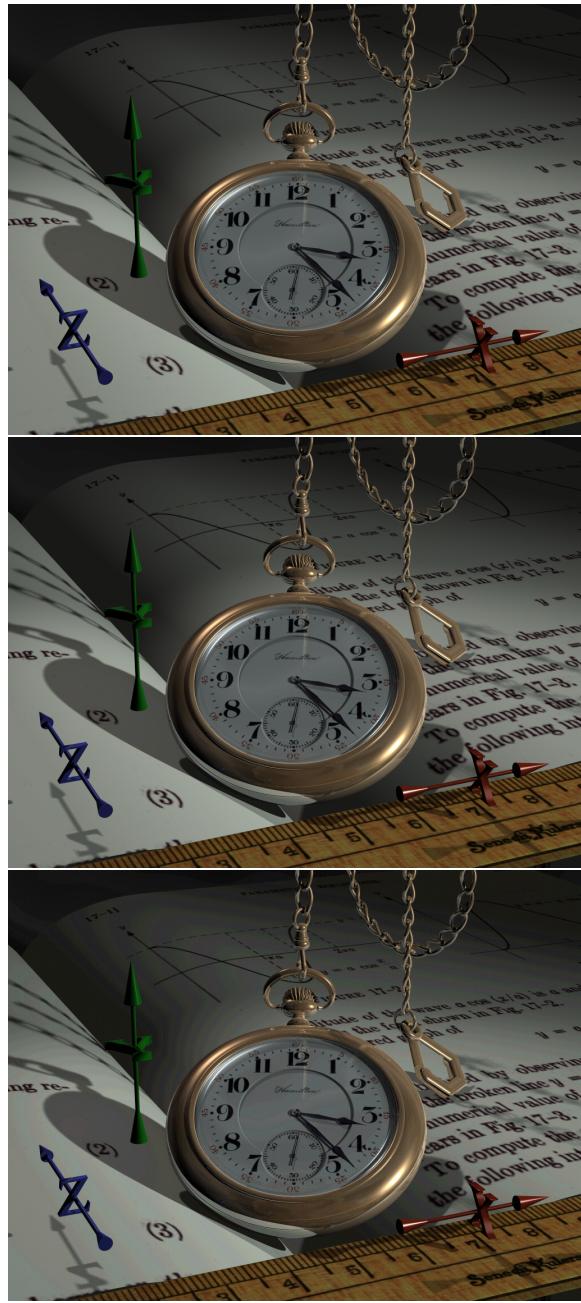


Figura 2: De cima para baixo, a imagem original do relógio, imagem do relógio com alguns capítulos de Dom Casmurro no bit 0 e a mesma imagem com os mesmos capítulos no bit 2.

4 Principais observações:

No geral, observa-se pouquíssimas diferenças em uma imagem e outra, sendo talvez mais notável em imagens de cores mais claras a adição de um pouco de ruído, principalmente em camadas mais significativas. Também, é difícil notar a alteração dos canais mais baixos da imagem, apenas sendo notável a alteração de padrão do ruído da imagem. É notável que, a partir extração das camadas, a maior parte da informação percebida está nas camadas mais significativa.

Ao se tentar converter alguns textos (alguns capítulos de Dom Casmurro e Memórias Póstumas de Brás Cubas) obteve-se erros devido à alguns caractéres não-ascii (travessões por exemplo, mas surpreendentemente não para acentos), tendo sido necessário removê-los (também por simplicidade, em uma implementação mais completa pode ser possível esconder qualquer caractere e até dados mais complexos como imagens e áudios).