## Universidade Federal de Alfenas - UNIFAL-MG

Bacharelado em Ciˆencia da Computa¸c˜ao Prof. Luiz Eduardo da Silva

Trabalho de Processamento de Imagens

**Atividade 2 - Esteganografia**

# Objetivo

O objetivo desse atividade ´e explorar os conceitos de cores e de codifica¸c˜ao e decodifica¸c˜ao utilizados nos formatos de imagens.

# Problema

O termo **esteganografia**1 deriva da jun¸c˜ao das palavras gregas *estegano* que significa “es- conder ou mascarar”, e *grafia*, que significa “escrita”. Portanto, esteganografia pode ser compre- endida como a arte de esconder informa¸c˜oes, tornando-as ocultas, assim como a criptografia. O objetivo desta t´ecnica ´e que esses dados n˜ao sejam percebidos por terceiros; ou seja, a presen¸ca de mensagens escondidas dentro de arquivos ´e simplesmente desconhecida. Somente o receptor da mensagem tem conhecimento de sua existˆencia, assim como da maneira como extra´ı-la.

Apesar de parecer, a esteganografia e a criptografia s˜ao duas ´areas com objetivos bastante diferentes. Enquanto o segundo tem o prop´osito de impedir que as pessoas saibam o conteu´do de uma mensagem, o primeiro se baseia em evitar que as pessoas saibam que a mensagem escondida existe. Ou seja, na criptografia, os receptores sabem da existˆencia das mensagens, por´em n˜ao conseguem, a princ´ıpio, lˆe-las; a esteganografia tenta fazer com que os receptores n˜ao percebam que h´a uma mensagem naquele meio (imagem, texto, etc.).

Existem v´arias formas de esconder mensagens ou arquivos em imagens. Uma das t´ecnicas poss´ıveis ´e modificar alguns bits (menos relevantes) dos pixels da imagem, com as informa¸c˜oes que se quer esconder. Por exemplo, a modifica¸c˜ao do bit menos significativo de uma banda (red) de um pixel de uma imagem colorida n˜ao ´e percept´ıvel ao olho humano. A esteganoan´alise dessa modifica¸c˜ao, no entanto, n˜ao ´e dif´ıcil de se realizar a partir da compara¸c˜ao do arquivo original e do arquivo modificado. O ru´ıdo, calculado como a diferen¸ca desses dois arquivos ´e a mensagem codificada.

O objetivo desse trabalho ´e completar o c´odigo dispon´ıvel nesta atividade para realizar a decodifica¸c˜ao de um arquivo que foi esteganografado na imagem.

# Descri¸c˜ao

1. Um arquivo est´a escondido na imagem .PPM. O arquivo escondido est´a codificado nos pixels da imagem da seguinte forma:

o bin´ario dos c´odigos ASCII do nome original do arquivo (finalizado pelo caracter zero)

o bin´ario do tamanho do arquivo (em quatro bytes)

e os bytes do arquivo.

1h[ttps://www.gta.ufrj.br/grad/09](http://www.gta.ufrj.br/grad/09) 1/versao-final/stegano/introducao.html

1. Os bits (zero ou um) das representa¸c˜oes bin´arias est˜ao armazenados, modificando as bandas R,G e B dos pixels em sequˆencia. Por exemplo, considerando a imagem colorida:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *r*1 *g*1 *b*1 | *r*2 *g*2 *b*2 | *r*3 *g*3 *b*3 | *r*4 *g*4 *b*4 | *r*5 *g*5 *b*5 | *r*6 *g*6 *b*6 |
| *r*7 *g*7 *b*7 | *r*8 *g*8 *b*8 | *r*9 *g*9 *b*9 | *r*10 *g*10 *b*10 | *r*11 *g*11 *b*11 | *r*12 *g*12 *b*12 |

Seja ’A’, a primeira letra do nome do arquivo (cujo c´odigo ASCII bin´ario ´e 01000001). Na codifica¸c˜ao do arquivo, a letra ’A’ modifica as seguintes bandas dos pixels iniciais (identificados em letras maiu´sculas e com cores).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *R*1 *g*1 *b*1 | *r*2 *G*2 *b*2 | *r*3 *g*3 *B*3 | *R*4 *g*4 *b*4 | *r*5 *G*5 *b*5 | *r*6 *g*6 *B*6 |
| *R*7 *g*7 *b*7 | *r*8 *G*8 *b*8 | *r*9 *g*9 *b*9 | *r*10 *g*10 *b*10 | *r*11 *g*11 *b*11 | *r*12 *g*12 *b*12 |

Onde:

*R*1 = *r*1 & 0*xFE*, codifica o bit zero (0)

*G*2 = *g*2 *|* 0*x*01, codifica o bit um (1)

*B*3 = *b*3 & 0*xFE*, codifica o bit zero (0)

*R*4 = *r*4 & 0*xFE*, codifica o bit zero (0)

*G*5 = *g*5 & 0*xFE*, codifica o bit zero (0)

*B*6 = *b*6 & 0*xFE*, codifica o bit zero (0)

*R*7 = *r*7 & 0*xFE*, codifica o bit zero (0)

*G*8 = *g*8 *|* 0*x*01, codifica o bita um (1)

1. A opera¸c˜ao *R*1 = *r*1 & 0*xFE*, que executa um e-bit-a-bit com a m´ascara 0*xFE*, tem o efeito de transformar o u´ltimo bit da banda R para zero.
2. De forma sim´etrica, a opera¸c˜ao *G*2 = *g*2 0*x*01 , que executa um ou-bit-a-bit com a m´ascara 0*x*01, tem o efeito de transformar o u´ltimo bit da banda G para um e assim sucessivamente.

*|*

1. Depois da codifica¸c˜ao do nome do arquivo, os quatro bytes seguintes definem a quantidade de bytes do arquivo, e dessa forma, a quantidade de pixels que est˜ao modificados no restante da imagem.

## Exemplo

Arquivo original:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | P3 |  | |
| 2 | #Created with GIMP |
| 3 | 10 10 |
| 4 | 255 |
| 5 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 6 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 7 | 0 0 0 140 26 26 197 40 40 197 40 40 140 26 26 0 0 0 0 0 0 0 | 0 0 0 | 0 0 0 0 0 |
| 8 | 0 0 0 197 40 40 209 43 43 209 43 43 197 40 40 40 140 26 59 | 197 40 | 59 197 40 40 140 26 0 0 0 |
| 9 | 0 0 0 197 40 40 209 43 43 166 53 144 85 61 197 61 197 85 63 | 209 43 | 63 209 43 59 197 40 0 0 0 |
| 10 | 0 0 0 140 26 26 197 40 40 85 61 197 43 63 209 61 198 90 63 | 209 43 | 63 209 43 59 197 40 0 0 0 |
| 11 | 0 0 0 0 0 0 0 0 0 40 59 197 43 63 209 53 147 166 61 197 85 | 59 197 | 40 40 140 26 0 0 0 |
| 12 | 0 0 0 0 0 0 0 0 0 26 40 140 40 59 197 40 59 197 26 40 140 0 | 0 0 0 | 0 0 0 0 0 |
| 13 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |  |  |
| 14 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |  |  |

Arquivo modificado:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | P3 |  | |
| 2 | # CREATOR: Image P r o c e s s i n g using C=Ansi = ByDu |
| 3 | # |
| 4 | 10 10 |
| 5 | 255 |
| 6 | 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 |
| 7 | 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 |
| 8 | 0 0 0 141 26 26 197 40 40 197 40 40 140 26 26 0 1 0 0 0 1 1 | 0 0 0 | 1 0 0 0 0 |
| 9 | 0 0 0 197 40 40 209 43 42 209 43 43 197 41 40 40 140 27 58 | 197 40 | 59 197 40 40 140 26 0 0 0 |
| 10 | 0 0 0 197 40 40 208 43 43 166 52 144 85 61 196 60 197 85 63 | 208 43 | 63 209 42 58 197 40 0 0 0 |
| 11 | 0 0 0 140 26 26 197 40 40 85 61 196 42 63 209 61 198 90 63 | 209 42 | 62 209 43 59 196 40 0 0 0 |
| 12 | 0 0 0 0 0 0 0 0 0 40 59 197 43 62 209 53 147 166 60 197 85 | 59 196 | 40 40 140 26 0 0 0 |
| 13 | 0 0 0 0 0 0 0 0 0 26 40 140 40 59 196 40 59 197 26 40 140 0 | 0 0 1 | 0 0 0 0 0 |
| 14 | 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 |  |  |
| 15 | 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |  |  |

Para incluir o arquivo de nome ’A.txt’, com os seguintes 2 bytes:

1 42

Cuja codifica¸c˜ao (nome, tamanho e os bytes) em bin´ario ´e:

1 A( 41 ) 01000001 . ( 2 E) 00101110

2 t ( 7 4 ) 01110100 x ( 78 ) 01111000 t ( 7 4 ) 01110100 ( 0 ) 00000000 // nome

3 ( 0 ) 00000000 ( 0 ) 00000000 ( 0 ) 00000000 ( 2 ) 00000010 // tamanho

4 4 ( 34 ) 00110100 2 ( 32 ) 00110010 // bytes

1. Pode-se observar ainda que, considerando esse esquema de codifica¸c˜ao, para que um ar- quivo possa ser escondido numa imagem, o nu´mero de pixels da imagem deve ser suficiente para a codifica¸c˜ao do arquivo. Sejam:

TN (Tamanho do Nome) = o nu´mero de letras do nome do arquivo somado de um (caracter zero no fim do nome);

TA (Tamanho do arquivo) = quatro bytes usados para codificar o tamanho do ar- quivo;

NB (Nu´mero de bytes) = nu´mero de bytes do arquivo a ser ocultado;

nlinhas = nu´mero de linhas da imagem;

ncolunas = nu´mero de colunas da imagem.

Assim, o arquivo s´o pode ser esteganografado na imagem se:

8 *×* (*TN* + *TA* + *NA*) *≤ nlinhas × ncolunas*

1. No exemplo anterior, TN = 6, TA = 4, NB = 2, o que soma 12 \* 8 pixels que foram afetados, ou seja, 96 pixels da imagem com 100 pixels (10 linhas e 10 colunas) foram modificados para codificar o arquivo.
2. Considerando esse esquema de codifica¸c˜ao do arquivo na imagem, a sua tarefa ´e desenvol- ver o **decodificador**. O c´odigo a ser completado est´a dispon´ıvel junto com a atividade. Este programa deve receber o nome do arquivo da imagem, extrair e gerar (em disco) o arquivo que est´a escondido.

# Entrega

1. Incluir um comentario no cabe¸calho de cada programa fonte com o seguinte formato:

1 /\*=============================================================

2 \* UNIFAL = Universidade Federal de Alfenas .

3 \* BACHARELADO EM CIENCIA DA COMPUTACAO.

4 \* Trabalho . . : Es te ga no gra f ia

5 \* D i s c i p l i n a : Processamento de Imagens

6 \* Pr o f e s s o r . : Luiz Eduardo da Silva

7 \* Aluno . . . . . : Fulano da Silva

8 \* Data . . . . . . : 99 /99 /9999

9 \*=============================================================\*/

1. O projeto dever´a incluir um arquivo MAKEFILE para constru¸c˜ao da ferramenta de deco- difica¸c˜ao do arquivo que est´a esteganografado (J´a dispon´ıvel nos arquivos da atividade)
2. O programa dever´a ser chamado em linha de comando da seguinte forma:

1 . / decode *<*nome=arquivo=imagem *>* [ .ppm]

2

3 por exemplo :

4 . / decode porto

1. O programa dever´a gerar o arquivo que est´a codificado na imagem, na mesma pasta onde o decodificador for executado.