Universidade Federal de Alfenas - UNIFAL-MG
Bacharelado em Ciência da Computação
Prof. Luiz Eduardo da Silva

Trabalho de Processamento de Imagens
Atividade 1 - Decodificação de uma formato de imagem

Objetivo

O objetivo dessa atividade é explorar os conceitos relacionados a representação de imagem, compactação, codificação de Huffman.

Problema

A fim de diminuir o tamanho dos arquivos PGM, propõe-se um novo formato que explora a codificação dos pixels usando uma árvore de Huffman. Um arquivo de imagem neste novo formato (.PGH) está ilustrado na Figura 1.

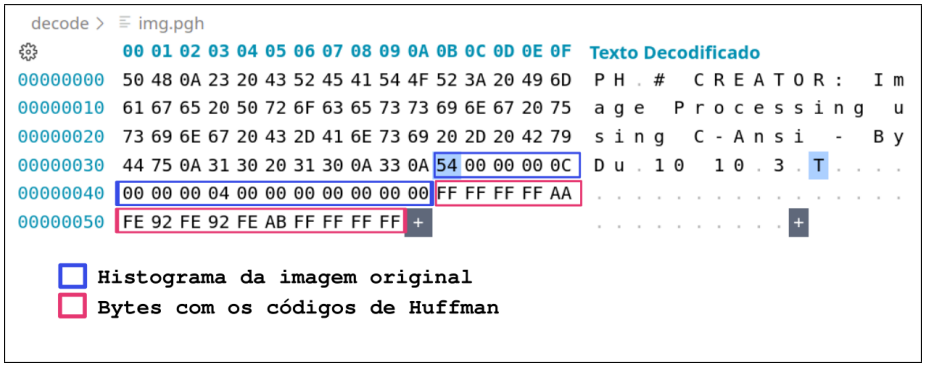

Figura 1: Imagem PGH

Conforme ilustrado na Figura 1, este novo formato de arquivo é composto de um cabeçalho (assim como o arquivo PGM, exceto que o número mágico é PH), em sequência está codificado em bytes o histograma da imagem usando 4 bytes para cada nível de cinza da imagem original e por fim, os bytes que codificam os pixels consideram os códigos de huffman obtidos pela construção da árvore de huffman.

Este arquivo foi obtido a partir do arquivo PGM, conforme está ilustrado na Figura 2.

A árvore de Huffman construída, considerando o histograma dessa imagem está representada na Figura 3.

Descrição

1. Desenvolva um programa, para: (a) fazer a leitura de um arquivo de imagem neste novo formato; (b) converter para PGM e posteriormente; (c) apresentar a visualização do arquivo decodificado. Para tanto, considere a implementação do exemplo de construção de árvore de Huffman apresentada em sala e disponível no repositório GIT da disciplina.

2. Os passos da decodificação são:

encode > img.pgm

```
1   P2
2   # Imagem teste
3   10 10
4   3
5   0 0 0 0 0 0 0 0 0 0
6   0 0 0 0 0 0 0 0 0 0
7   0 0 0 0 0 0 0 0 0 0
8   0 0 0 1 1 1 1 0 0 0
9   0 0 0 1 2 2 1 0 0 0
10  0 0 0 1 2 2 1 0 0 0
11  0 0 0 1 1 1 1 0 0 0
12  0 0 0 0 0 0 0 0 0 0
13  0 0 0 0 0 0 0 0 0 0
14  0 0 0 0 0 0 0 0 0 0
15
```

Figura 2: Imagem original em formato PGM

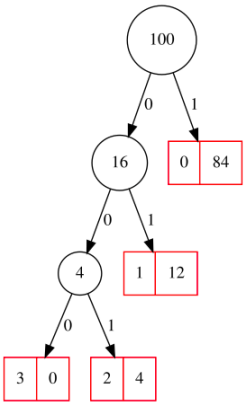

Figura 3: Árvore de Huffman

(a) Abrir o arquivo .PGH obtido em linha de comando

(b) Ler a primeira linha do arquivo que deve conter o "número mágico"PH.

(c) Em sequência, ler um número de linhas de comentários iniciados com hashtag. O arquivo pode não conter nenhuma linha de comentário.

(d) Ler a próxima linha com o número de colunas e o número da linhas da imagem original separados por um espaço

(e) Ler na próxima linha, o máximo nível (mn) de cinza da imagem original.

(f) Ler a tabela do histograma da imagem original, cujo tamanho deve ser quantidade de tons de cinza da imagem original (mn + 1) x quatro bytes (tamanho de um int).

(g) Por último estão os bytes que codificam (usando os códigos de huffman), os pixels da imagem original.

Entrega

1. Incluir um comentário no cabeçalho de cada programa fonte com o seguinte formato:

```
/*-------------------------------------------------------------------
 *            UNIFAL - Universidade Federal de Alfenas.
 *              BACHARELADO EM CIENCIA DA COMPUTACAO.
 * Trabalho..: Decodificador do formato PGH
 * Professor.: Luiz Eduardo da Silva
 * Aluno.....: Fulano da Silva
 * Data......: 99/99/9999
 *-----------------------------------------------------------------*/
```

2. O projeto deverá incluir um arquivo MAKEFILE para construção do executável que deverá ser nomeado de **decode**.

3. O programa deverá ser chamado em linha de comando da seguinte forma (com a imagem de entrada como parâmetro), como por exemplo:

```
./decode <nomedaimagem[.pgh]>
```

4. Enviar num arquivo único (.ZIP), com todos os arquivos fonte do projeto através do Envio de Arquivo do MOODLE.

5. **Não utilizar qualquer biblioteca com os algoritmos. Utilizar as implementações dos algoritmos discutidos em aula.**

Em sequência está uma sugestão para o programa principal do decodificador:

```
/*-------------------------------------------------------------------
* Image Processing using C-Ansi
*   Program: decoding files in PGH format
*-------------------------------------------------------------------*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "../include/imagelib.h"
#include "../include/huffman.h"
#include "../include/node.h"

void msg(char *s)
{
        printf("\nEdecoding files in PGH format");
        printf("\n-------------------------------------");
        printf("\nUsage: %s image-name[.pgh]\n\n", s);
        printf("    image-name[.pgh] is image file in pgh format \n\n");
        exit(1);
}

/*-------------------------------------------------------------------
* main function
*-------------------------------------------------------------------*/
int main(int argc, char *argv[])
{
        char nameIn[100], nameOut[100], cmd[110];
        image In;

        if (argc < 2)
        msg(argv[0]);

        //-- define input/output file name
        img_name(argv[1], nameIn, nameOut, PGH, GRAY);

        //-- read image PGH
        In = read_pgh(nameIn);

        //-- save image PGM
        img_put(In, nameOut, GRAY);

        //-- show image
        sprintf(cmd, "eog %s &", nameOut);
        puts(cmd);
        system(cmd);
        img_free(In);
        return 0;
}
```