Fotomosaico

Video deste projeto



Um fotomosaico ou mosaico fotográfico é uma fotografia construída a partir de imagens menores, ou "pastilhas" (do inglês *tiles*). As pastilhas são escolhidas de forma a reproduzir a cor e intensidade luminosa do bloco da imagem original correspondente, gerando um efeito visual muito interessante.

O formato PPM

Para facilitar a leitura e escrita dos arquivos de imagem, neste projeto será adotado o formato de imagem Portable PixMap (PPM), um formato de imagem colorida bem simples e fácil de ler/escrever. Boa parte dos programas de tratamento de imagens reconhece o formato PPM.

A estrutura básica de um arquivo PPM é bem simples:

```
TYPE                    (tipo de imagem: P3 ou P6, vide abaixo)
WIDTH HEIGHT            (largura e altura da imagem em pixels)
MAX                     (valor máximo de cada cor)
R0 G0 B0 R1 G1 B1 ...   (valores Red/Green/Blue de cada pixel, entre 0 e MAX)
```



Existem duas variantes do formato PPM: *PPM ascii* (P3) e *PPM binary* (P6); **ambas** devem ser tratadas neste projeto. Mais detalhes sobre essas variantes podem ser obtidas nesta página.

Um exemplo de imagem PPM ascii (P3):

flower.ppm

```
P3
# this is a comment
10 10
255
255 255 255 255 255 255 255 255 255 255 255 255 255 254 254 255 254 254 255
254 254 255 255 255 255 255 255 255 255 255
```

```
255 255 255 255 255 255 255 255 255 238 215 214 218 145 145 213 120 119 241
224 224 254 253 253 255 255 255 255 255 255
255 255 255 255 255 255 236 209 209 217 60 59 219 52 53 222 56 56 215 56 55
229 184 183 255 254 254 255 255 255
255 255 255 255 254 254 209 103 103 224 54 55 239 96 47 236 86 46 228 58 57
213 126 126 255 254 254 255 255 255
255 255 255 255 255 255 209 93 93 228 55 56 240 120 41 238 99 44 229 59 59 213
123 123 255 254 254 255 255 255
255 255 255 255 255 255 217 151 151 220 53 52 233 60 59 225 53 53 218 57 57
230 191 191 255 254 254 255 255 255
255 255 255 255 255 255 250 246 245 208 183 158 200 94 84 181 123 94 201 165
133 229 234 203 255 255 255 255 255 255
255 255 255 255 255 255 253 253 251 192 210 139 127 169 65 122 166 64 156 186
87 229 236 204 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 254 254 253 172 200 141 201 220 182 254
254 253 255 255 255 255 255 255 255 255 255
255 255 255 255 255 255 255 255 255 255 255 255 250 251 248 253 254 253 255
255 255 255 255 255 255 255 255 255 255 255
```

Você pode usar as imagens que quiser neste projeto. Para perceber melhor o efeito de mosaico, use imagens de grandes dimensões (acima de 3000 pixels de largura). Para gerá-las, pode usar qualquer editor de imagens que ofereça a possibilidade de “salvar como” PPM. No Linux, os softwares Gimp e GThumb oferecem essa opção.

Atividade

Construir um programa em C (C99 ou similar) que leia uma imagem de entrada em formato PPM e gere como saída essa mesma imagem convertida em mosaico, no mesmo formato e dimensões da imagem de entrada, usando as pastilhas disponíveis em um diretório informado.

A estratégia básica de composição do mosaico é simples:

1. Consideram-se pastilhas quadradas, todas com N×N pixels (o programa deve descobrir N e se ajustar a ele)
2. Percorrer o diretório informado e carregar todas as pastilhas.
3. Para cada pastilha, calcular sua cor predominante.
4. Abrir/carregar imagem de entrada (PPM raw ou ascii).
5. Para cada bloco de N×N pixels da imagem de entrada:
    1. Calcular a cor predominante do bloco: [R_{med} G_{med} B_{med}].
    2. Substituir o bloco pela pastilha que tenha a cor predominante mais próxima.
6. Salvar a imagem de saída no mesmo formato da imagem de entrada (PPM raw ou ascii).

A **cor predominante de um bloco** pode ser calculada pela média simples das componentes RGB dos pixels que compõem esse bloco. Alguns autores sugerem, no entanto, usar a raiz quadrada da média dos quadrados das componentes RGB para obter um resultado mais preciso. A mediana das componentes RGB também pode trazer bons resultados.

A **distância entre duas cores** A = [R_A G_A B_A] e B = [R_B G_B B_B] é relativamente complexa de calcular, pois as cores não se comportam de forma linear para os olhos humanos. Uma fórmula simplificada de cálculo no padrão RGB é a distância *Red Mean*, descrita nesta página.



Um dos objetivos deste projeto é capacitar o aluno na leitura e análise de arquivos de dados ASCII e binários, por isso o uso de bibliotecas externas para leitura dos arquivos de imagem



não será aceito (Netpbm, OpenCV e similares).

Forma de chamada

```
$ mosaico [ -p diretório ] [ -i arquivo ] [ -o arquivo ]
```

Opções:

- `-i` : indica a imagem de entrada; se não for informada, assume-se a entrada padrão (*stdin*).
- `-o` : indica a imagem de saída; se não for informada, assume-se a saída padrão (*stdout*).
- `-p` : indica o diretório de pastilhas; se não for informado, assume-se o diretório `./tiles`.
- `-h` : gera uma mensagem de ajuda na saída de erro (*stderr*), explicando o que o programa faz e quais as opções disponíveis.
- Todas as mensagens de erro devem ser enviadas para a saída de erro (*stderr*).

Essas opções podem ser usadas em qualquer ordem:

```
// entrada e saída em arquivos
mosaico -i input.ppm  -o output.ppm
mosaico -o output.ppm -i input.ppm

// entrada em arquivo, saída em stdout, vice-versa ou ambos
mosaico -i input.ppm  > output.ppm
mosaico -o output.ppm < input.ppm
mosaico <  input.ppm  > output.ppm

// as opções podem estar em qualquer ordem
mosaico -p /tmp/tiles -i input.ppm -o output.ppm
mosaico -i input.ppm -p /tmp/tiles -o output.ppm
mosaico -o output.ppm -i input.ppm -p /tmp/tiles
```

O programa deve indicar sua atividade (em `stderr`):

```
Reading tiles from ./tiles
1419 tiles read
Tile size is 20x20
Calculating tiles' average colors
Reading input image
Input image is PPM P3, 3000x4496 pixels
Building mosaic image
Writing output file
```

Requisitos

- O executável deve se chamar `mosaico`.
- O programa deve aceitar como entrada uma imagem no formato PPM (P3 e P6, plain e raw) e gerar como saída uma imagem no mesmo formato e dimensões da entrada: se entrar uma imagem PPM *raw* deve sair uma imagem PPM *raw* e vice-versa.
- As pastilhas a usar para a construção do mosaico são imagens em formato PPM fornecidas pelo professor e colocadas em um diretório separado (por default `./tiles/`).
- Sempre que possível, as informações da imagem necessárias às funções devem ser transferidas como parâmetros (por valor ou por referência, dependendo da situação); o uso de variáveis globais deve ser

minimizado.
- Use `structs` e alocação dinâmica de memória para todas as imagens; só aloque a memória para os pixels após encontrar o tamanho da imagem.
- Construir um Makefile para o projeto:
  - Ao menos os alvos `all` (default), `clean` e `purge`.
  - `CFLAGS = -Wall`
  - Compilar e ligar separadamente (gerar os arquivos-objeto intermediários)
- O que deve ser entregue ao professor:
  - código-fonte
  - `Makefile`
  - Por favor, não envie as imagens de teste! 💪

Leitura de diretório

Para carregar as pastilhas, você terá de percorrer o diretório de pastilhas, encontrar os nomes de todos os arquivos e ler as imagens contidas neles. Existem funções específicas em C para manipular diretórios, que são descritas nesta página.



Não use funções como `system()` e `popen()` para varrer o diretório de pastilhas usando comandos do *shell*. Essas funções lançam processos externos e são **ineficientes** e **inapropriadas** neste caso, pois existem funções adequadas para resolver o problema dentro da própria linguagem C.

Pastilhas

Coleções de pastilhas prontas para uso:

- 20x20 pixels, PPM P3 e P6
- 32x32 pixels, PPM P6
- Tiles League of Legends, contribuição de João Lucas Cordeiro
- Tiles Minecraft, contribuição de Gabriel Dolzan e Dante dos Santos

Para os curiosos, o procedimento que usei para a construção dessas coleções de pastilhas foi o seguinte:

1. Limpar o cache do navegador web (usei o Firefox).
2. Fazer uma ou mais buscas no *Google Images* com o tema de interesse (por exemplo “flower”); deslizar várias páginas na busca, para carregar bem o cache com as miniaturas das imagens.
3. Copiar os arquivos do cache do navegador (em `$HOME/.mozilla/…/cache/`) para um diretório separado e remover os que não forem imagens ou não forem relacionados ao tema de interesse.
4. Converter as imagens para formato PPM (*raw* ou *ascii*), ajustando ao tamanho desejado (32×32 neste exemplo) e cortando os excessos para ficarem quadradas (o comando `mogrify` faz parte do pacote ImageMagick):

   ```
   $ mogrify -format ppm -thumbnail 32x32^ -gravity center -extent 32x32 *
   ```

5. Para obter imagens em formato PPM ASCII, adicionar os flags `-compress none`

Fonte: https://superuser.com/questions/275476/square-thumbnails-with-imagemagick-convert

From:
https://wiki.inf.ufpr.br/maziero/ - **Prof. Carlos Maziero**

Permanent link:
https://wiki.inf.ufpr.br/maziero/doku.php?id=c:fotomosaico

Last update: **2023/08/01 16:20**