

Programa de tratamento de imagem

MPCP (FEUP/MIEIC)

Abril de 2013

1 Introdução

O programa VIEWER permite visualizar os efeitos de sub-rotinas de tratamento de imagem. Este programa é uma adaptação de um exemplo de demonstração da autoria da Microsoft (WicViewerGdi).

Para utilizar o programa, carrega-se uma imagem (JPG ou PNG); de seguida, podem aplicar-se à imagem sub-rotinas escritas em C++ (menu C++) ou em *assembly* (menu ASM). As alterações são imediatamente visualizadas. (Nota: O redimensionamento da janela faz com que a imagem original seja reposta.)

2 Parâmetros das sub-rotinas

Na versão inicial do programa, apenas se encontra implementada uma sub-rotina em C++ e outra em *assembly*; ambas convertem uma imagem de cor para níveis de cinzento (as implementações são ligeiramente diferentes).

As sub-rotinas de processamento de imagem recebem três parâmetros:

1. **pixels**: apontador para a informação sobre cada ponto da imagem. Essa informação está arranjada por linhas.
2. **largura**: número de pixels por linha.
3. **altura**: número de linhas da imagem.

Uma imagem tem $\text{largura} \times \text{altura}$ pixels.

Em C++, a declaração é:

```
void cfunc1(unsigned char *pixels, long largura, long altura);
```

Em *assembly*, a declaração é:

```
afunc1 PROC pixels: ptr byte, largura:dword, altura:dword
```

As sub-rotinas em C++ estão definidas nos ficheiros `cfunc1.cpp`, `cfunc2.cpp` e `cfunc3.cpp`.

As sub-rotinas em *assembly* estão definidas nos ficheiros `asmfunc1.asm`, `asmfunc2.asm` e `asmfunc3.asm`.

3 Formato de pixels

A cor da imagem é representada no formato RGB (*red, green, blue*). Cada ponto da imagem (pixel) é representado por quatro bytes consecutivos, com o seguinte significado:

1. valor da componente B (azul);
2. valor da componente G (verde);
3. valor da componente R (vermelho);
4. valor da “transparência”.

Os valores de cada componente vão de 0 (mínimo) a 255 (máximo). A cor preta tem o valor RGB={0, 0, 0}; no outro extremo, a cor branca tem o valor RGB={255, 255, 255}. Quando todas as componentes RGB são iguais, obtém-se um nível de cinzento. A transparência é 0 para imagens opacas (o caso mais comum).

As implementações das sub-rotinas `cfunc1` e `asmfunc1` (em C++ e *assembly*, respetivamente) mostram formas de aceder a todos os pontos de uma imagem.