# Estratégia utilizando paleta de cores

Temos o par estéreo. Criamos o anaglífico, o qual contém o canal de cor vermelho da imagem direita e os canais verde e azul da imagem esquerda. Os canais restantes, aqui nomeados “paleta de cores”, devem ser armazenados em alguma estrutura e gravados como um arquivo auxiliar, o que servirá para o processo de reconstrução do par estéreo.

## 1) Abordagem utilizando uma imagem auxiliar

Comando de chamada do programa:

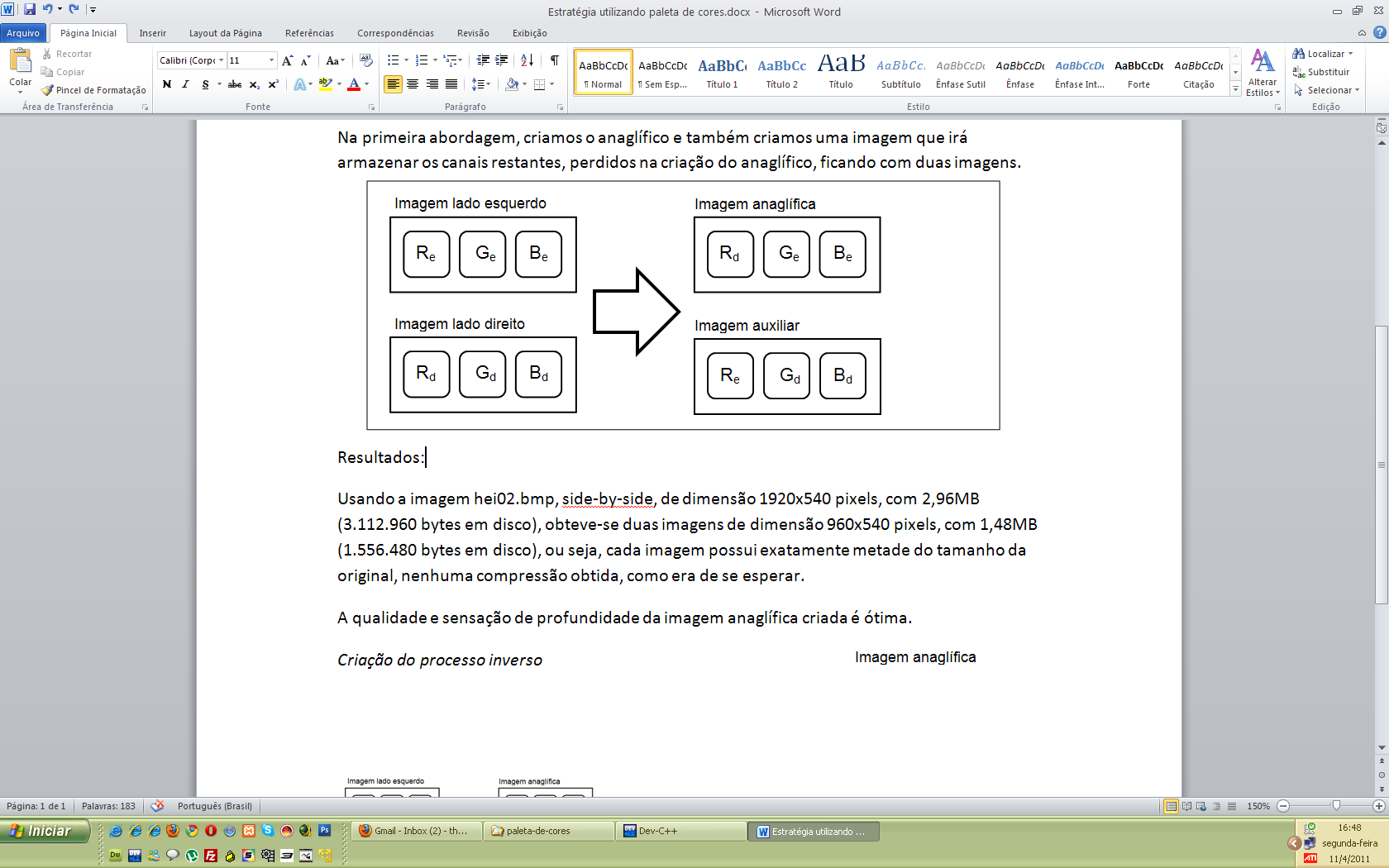
paleta-imagem <image.bmp> {O | 1} {-e | [-d <auxiliary\_image.bmp>]}

0: side-by-side image

1: above/below image

-e: encoding

-d: decoding



*Criação do processo de transformação anaglífica*

Na primeira abordagem, criamos o anaglífico e também criamos uma imagem que irá armazenar os canais restantes, perdidos na criação do anaglífico, ficando com duas imagens.

Resultados:

paleta-imagem hei02.bmp 0 -e

Usando a imagem hei02.bmp, side-by-side, de dimensão 1920x540 pixels, com 2,96MB (3.112.960 bytes em disco), obteve-se duas imagens de dimensão 960x540 pixels (anaglyph.bmp e auxiliary.bmp), com 1,48MB (1.556.480 bytes em disco), ou seja, cada imagem possui exatamente metade do tamanho da original, nenhuma compressão obtida, como era de se esperar.

A qualidade e sensação de profundidade da imagem anaglífica criada é ótima.

*Criação do processo inverso*

Para o processo inverso, devemos fazer um rearranjo das cores contidas tanto na imagem anaglífica quanto na auxiliar. Criamos um novo container de imagem no OpenCV que, dependendo do parâmetro passado pelo usuário, terá o dobro de largura da imagem anaglífica (para o caso de a imagem original ser do tipo side-by-side) ou o dobro de altura da imagem anaglífica (para o caso de a imagem original ser do tipo above/below).

Considerando o caso de a imagem original ser do tipo side-by-side, temos a seguinte estratégia: A primeira metade da largura do container criado, considerando o sentido da esquerda para a direita, deve receber os valores contidos no canal vermelho da imagem auxiliar e os valores contidos nos canais verde e azul da imagem anaglífica. A segunda metade do container criado receberá o inverso, os valores contidos nos canais verde e azul da imagem auxiliar e os valores do canal vermelho da imagem anaglífica. A estratégia é similar para imagens do tipo above/below, sendo tratado a metade de cima da altura do container criado e depois a metade de baixo.

Resultados:

paleta-imagem anaglyph.bmp 0 -d auxiliary.bmp

Obteve-se uma imagem (Par\_Estereo.bmp), do tipo side-by-side, de dimensão 1920x540px e tamanho 2,96MB (3.112.960 bytes em disco). Nenhum defeito encontrado. Um teste melhor é verificar o PSNR. Com isso, comprovamos que é possível fazer o processo reverso de uma imagem anaglífica. Resta agora estudarmos métodos para que a transformação anaglífica utilizando um arquivo auxiliar seja feita com compressão.

## 2) Abordagem utilizando vetores

A criação de uma imagem auxiliar, abordagem utilizada em 1, carrega dados que não nos interessam, como o cabeçalho da imagem. O realmente necessário é o que está disponível no campo imageData da estrutura IplImage utilizada no OpenCV. Desta forma, uma segunda abordagem é utilizar um vetor que armazene apenas os valores dos pixels de cada um dos canais removidos no processo anaglífico. Tal vetor posteriormente deve passar por algum algoritmo de compressão *lossless*. Novamente, devemos verificar se o processo reverso é possível e reverte a uma imagem sem perda de qualidade.

Comando de chamada ao programa

paleta-vetor <image.bmp> {O | 1} {-e | -d}

0: side-by-side image

1: above/below image

-e: encoding

-d: decoding

Resultados

paleta-vetor hei02.bmp 0 –c

No processo de transformação anaglífica, é criado um arquivo chamado pixelData.dat que contém os valores dos pixels removidos na criação do anaglífico. Podemos dividir este arquivo em três partes. A primeira terça parte contém os dados do canal vermelho da imagem à esquerda, a segunda terça parte contém os dados do canal verde da imagem à direita e por fim, a terceira terça parte contém os dados do canal azul da imagem à direita. O arquivo pode ser lido e armazenado em um vetor, e para acessar cada parte basta acrescentar um offset do tamanho da resolução da imagem (largura\*altura da imagem).

O tamanho do arquivo anaglífico foi de 1,48 MB (1.555.254 bytes) e o do arquivo pixelData.dat foi de 1,48MB (1.555.200 bytes). O arquivo original possui um tamanho de 2.96MB (3.110.456 bytes), ou seja, obteve-se a redução de apenas 2bytes comparado ao arquivo original. Tal redução deve ser maior ao aplicarmos uma compressão *lossless* em pixelData.dat. Fizemos um teste zipando o arquivo pixelData.dat utilizando o programa WinRAR. O tamanho do arquivo ficou em 855KB (875.542 bytes), totalizando 2,31 MB (2.430.796 bytes) com a imagem anaglífica, resultando em 22% de compressão. Isso comprova que é possível obter uma taxa razoável de compressão aplicando algoritmos do tipo *lossless*.

paleta-vetor anaglyph.bmp 0 -d

O resultado da reversão do método gerou novamente o par estéreo com sucesso, nenhum defeito encontrado. Um teste melhor é verificar o PSNR.

Para assegurar o resultado positivo da reversão, o par estéreo obtido foi novamente submetido ao programa para a formação da imagem anaglífica. O resultado foi satisfatório, gerando uma imagem anaglífica com sensação de profundidade.