

## Imprimindo Imagens Coloridas em um Dispositivo Monocromático

Cláudio Esperança  
João Luiz Dihl Comba

COPPE-Sistemas, UFRJ  
Caixa Postal 68511  
Rio de Janeiro - RJ - 21945

**RESUMO:** Este trabalho apresenta técnicas de discretização para exibição de imagens originalmente coloridas em dispositivos de impressão monocromáticos. Uma variante do algoritmo de difusão de erros é apresentada com o fim de corrigir eventuais distorções introduzidas pelo mecanismo de impressão.

### 1. Introdução

Uma facilidade frequentemente desejada em um ambiente de síntese de imagens é a de reproduzir uma cópia em papel de uma imagem cromática obtida a partir de um processo qualquer. No nosso caso, por exemplo, as imagens foram obtidas por um sistema de "Ray Tracing".

Obviamente, o meio preferencial para obter cópias em papel é o processo fotográfico. Entretanto, esse recurso é caro e em muitas ocasiões pouco prático. Nesse caso, é costume optar pela utilização de impressoras e, novamente, vemo-nos na situação em que o dispositivo ideal, isto é, impressoras coloridas, não estão normalmente disponíveis. As imagens apresentadas neste trabalho, por exemplo, foram reproduzidas numa impressora *Apple LaserWriter II*. Deparamo-nos então com o seguinte problema: dada uma imagem colorida, gerar uma representação correspondente usando apenas duas cores - branco e preto.

Claramente, a solução do problema envolve duas etapas:

- Transformar as cores dos pontos da imagem original em tons de cinza
- Dada a imagem expressa agora em tons de cinza, empregar um processo de discretização gerando uma imagem composta apenas de pontos brancos e pretos.

### 2. Avaliando tons de cinza

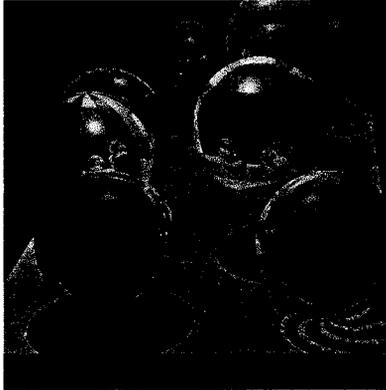
O problema de associar um tom de cinza a uma cor está sujeito a uma série de fatores, uma vez que durante o processo há necessariamente perda de informação. Há casos em que se deseja preservar visualmente o efeito de gradações de cores, ou seja, o efeito de sombreado. Em outras ocasiões, deseja-se preservar a distinção cromática, isto é, fazer por exemplo com que tons de vermelho e verde com o mesmo grau de "luminosidade" sejam distinguíveis na imagem em tons de cinza.

Para obter resultados para todos os gostos, há basicamente dois fatores que devem ser considerados:

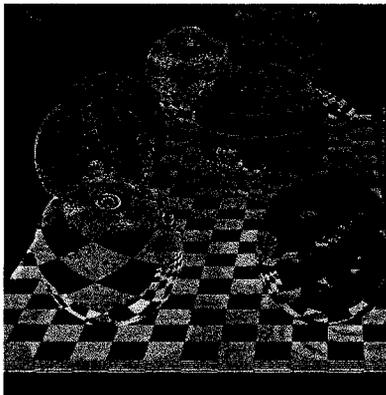
- Qual sistema de cores (RGB, HSV, HLS, etc) deve ser empregado para representar as cores da imagem
- Qual função a ser empregada para, dadas as três componentes de cor primária, avaliar o tom de cinza correspondente.

Fizemos experimentos com os sistemas de cores RGB (*Red-Green-Blue*) e HSV (*Hue-Saturation-Value*). Observamos que no sistema RGB, cores com intensidades próximas tendem a se apresentar em

coordenadas também próximas. Já o sistema HSV tende a preservar mais a cromaticidade, isto é, cores diferentes apresentam-se afastadas no espaço de cores, havendo uma menor dependência da intensidade. Este efeito é bem demonstrado nas Figuras 1(a) - produzida usando o sistema RGB - e 1(b) onde o sistema HSV foi empregado. Pode-se observar na figura 1(a) que o efeito de sombreado foi bem representado, enquanto que na figura 1(b) é possível distinguir as diversas texturas mapeadas nos sólidos. Em ambas as figuras, o tom de cinza foi avaliado pela média aritmética das três componentes.



(a)



(b)

Fig. 1

Experimentamos também três métricas, isto é, funções que avaliam a distância entre duas dadas cores no sistema tridimensional de cores, a saber:

(i) Maior das três componentes

$$G = \max_{i=1..3} C_i$$

(ii) Média aritmética entre as três componentes

$$G = \frac{\sum_{i=1..3} C_i}{3}$$

(iii) Distância euclideana da cor à origem do sistema de cores

$$G = \sqrt{\sum_{i=1..3} \frac{C_i^2}{3}}$$

Na discussão acima, cada  $C_i$  é uma das componentes de cor primária escaladas entre 0 e 1, e  $G$  é o tom de cinza associado. As Figuras 1(a) e (b) foram produzidas usando a métrica (ii). As figuras 2(a) e 2(b) foram obtidas usando o sistema de cores RGB e as métricas (i) e (iii).

3.

### O processo de discretização no espaço de cores ( *Halftoning* )

Uma imagem expressa em tons de cinza normalmente não pode ser reproduzida diretamente numa impressora; o que se faz então é aproximar as diversas gradações de cinza usando uma quantidade de pontos pretos e brancos numa proporção adequada. Trata-se de um problema bastante pesquisado e, dentre as técnicas mais conhecidas para sua solução destacamos os algoritmos de Floyd-Steinberg ([FS 75]), o de difusão de erro (*dot diffusion*) ([Knuth 87],[Pins 89]), o que utiliza uma matriz ordenada de valores (*ordered dither*) ([Newman 81]) e o que utiliza curvas de Peano ([WN 82] e [FB 88]).

Por razões que ficarão aparentes mais adiante, optamos por utilizar o algoritmo de difusão de erro; é conveniente portanto que o apresentemos brevemente:

O algoritmo de difusão de erro é uma variante do algoritmo de Floyd-Steinberg. Ele consiste em procurar compensar em cada ponto sendo convertido o erro cometido no processamento de pontos vizinhos.

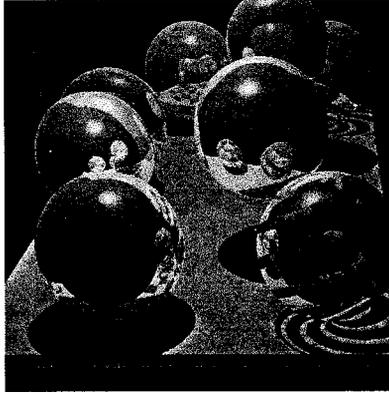
O fator de compensação *Comp* é computado levando-se em conta a distância entre cada vizinho e o ponto de interesse; isto é feito atribuindo um peso proporcional à distância. Em nossa implementação, o fator de compensação computado para um ponto "p" na coluna "c", linha "l" é dado pela média ponderada dos erros cometidos no processamento dos 12 vizinhos correspondentes às 2 linhas e 2 colunas mais próximas. Eis abaixo a matriz de pesos utilizada:

1	2	4	2	1
2	4	8	4	2
4	8	p		

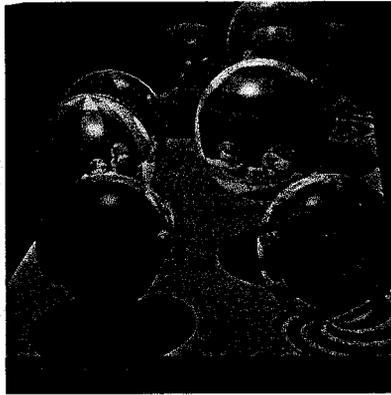
O fator de compensação associado ao ponto "p" é então dado por

$$Comp(p) = \frac{\sum_{i=1..12} Erro(Vizinho(p,i)) \cdot Peso(Vizinho(p,i))}{\sum_{i=1..12} Peso(Vizinho(p,i))}$$

Ao valor original  $Val(p)$  é então acrescentado o fator de compensação  $Comp(p)$  e a discretização é feita pelo seguinte trecho de código:



(a)



(b)

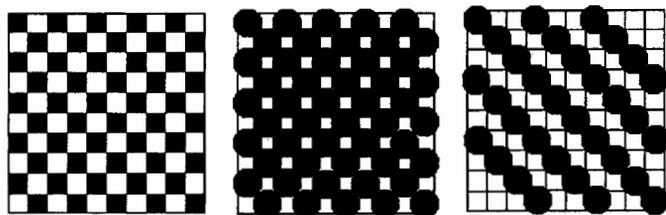
Fig. 2

NovoValor = Val (p) + Comp (p)  
 SE NovoValor  $\geq 0.5$  ENTÃO  
     PoePonto (Preto)  
     Erro (p) = NovoValor - 1  
 SENÃO  
     PoePonto (Branco)  
     Erro (p) = NovoValor  
 FIM-SE

#### 4. Aberrações produzidas pelo mecanismo de impressão

Para obter o tom de cinza desejado, os algoritmos de discretização assumem como premissa que o dispositivo para exibição da imagem é capaz de pintar pontos com forma perfeitamente retangular. Dessa maneira, o processo de integração efetuado pela visão interpretaria gradações de cinza em proporção direta com a razão entre a quantidade de pontos pretos e brancos numa dada área da imagem.

Isto não se verifica experimentalmente. Na prática, os pontos pintados pelos mecanismos de impressão têm forma irregular, mais se aproximando a manchas circulares. Tome como exemplo um tom de cinza médio, isto é, correspondente a uma quantidade igual de pontos pretos e brancos. Tipicamente um algoritmo de discretização produziria um padrão como o representado na figura 3(a). Este padrão ao ser impresso produziria uma relação claro/escuro como a mostrada na figura 3(b). Para obter o tom de cinza desejado nesse mesmo dispositivo seria desejável produzir um padrão como o mostrado na figura 3(c).



(a)

(b)

(c)

Fig. 3

Notamos então que os dispositivos de impressão introduzem aberrações quando pontos pretos e brancos aparecem muito próximos, ou seja, a tinta preta "invade" as áreas brancas vizinhas.

Para amenizar esse efeito, é comum utilizar-se em tipografia um algoritmo de discretização por matriz ordenada onde os pontos pretos tendem a se agregar, gerando uma imagem claramente pontilhada. Nossa imagem de teste foi impressa segundo esse processo e o resultado é apresentado na figura 4.

Uma alternativa a esse processo é computar o erro sendo introduzido sempre que um ponto preto invade uma posição branca. Uma vez que o algoritmo de difusão de erros já mantém uma medida do erro introduzido nas imediações de um ponto, é simples acrescentar mais uma parcela nesse cômputo.

A parcela de erro é acrescentada sempre que um ponto preto é produzido na vizinhança de um ponto branco. Precisamos distinguir dois fatores de compensação correspondentes respectivamente a pontos vizinhos na diagonal ou na mesma linha ou coluna - veja na figura 5.

O trecho de código modificado toma então a seguinte forma

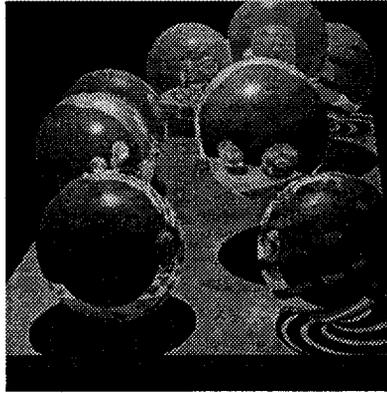


Fig. 4

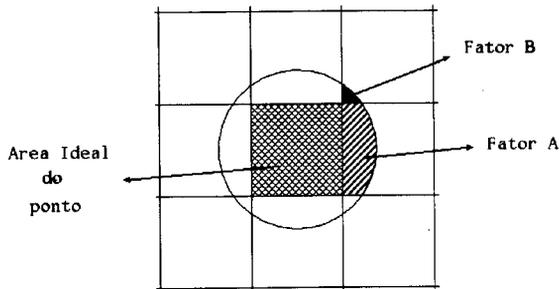


Fig. 5

NovoValor = Val (p) + Comp (p)  
 SE NovoValor >= 0.5 ENTÃO  
   PoePonto (Preto)  
   Erro (p) = NovoValor - 1  
   SE Acima (p) = Branco ENTÃO  
     Erro (Acima (p)) = Erro (Acima (p)) + FatorA  
   FIM-SE  
   SE Esquerda (p) = Branco ENTÃO  
     Erro (Esquerda (p)) = Erro (Acima (p)) + FatorA  
   FIM-SE  
   SE Acima-Esquerda (p) = Branco ENTÃO  
     Erro (Acima-Esquerda (p)) = Erro (Acima-Esquerda (p)) + FatorB  
   FIM-SE  
   SE Acima-Direita (p) = Branco ENTÃO  
     Erro (Acima-Direita (p)) = Erro (Acima-Direita (p)) + FatorB  
   FIM-SE  
 SENÃO  
   PoePonto (Branco)

Erro (p) = NovoValor  
 SE Acima (p) = Preto ENTAAO  
 Erro (p) = Erro (p) - FatorA  
 FIM-SE  
 SE Esquerda (p) = Preto ENTAAO  
 Erro (p) = Erro (p) - FatorA  
 FIM-SE  
 SE Acima-Esquerda (p) = Preto ENTAAO  
 Erro (p) = Erro (p) - FatorB  
 FIM-SE  
 SE Acima-Direita (p) = Preto ENTAAO  
 Erro (p) = Erro (p) - FatorB  
 FIM-SE  
 FIM-SE

## 5. Conclusão

As técnicas apresentadas neste artigo mostraram ser bastante flexíveis para obter reproduções em preto e branco de imagens coloridas preservando no processo suas diversas características. Uma medida mais acurada da fidelidade das imagens produzidas pelo algoritmo para correção de aberrações deveria idealmente ter sido obtida com o auxílio de um fotômetro. Fizemos a calibração do raio do ponto produzido pela impressora através de comparações com as imagens obtidas usando discretização por matriz ordenada. Testes adicionais deveriam ter sido feitos utilizando outros dispositivos de impressão.

## 6. Bibliografia

- [FB 89] B. C. Frías, A. Blumenkrans, "Display de dos niveles de intensidad para imagenes de intensidad continua", *Anais do SIBGRAPI 89*, pp 51-58.
- [FS 75] R. W. Floyd and L. Steinberg, "An adaptative algorithm for spatial gray scale," *Int. Symposium Dig. Tech. Papers*, 36, 1975.
- [Knuth 87] D. E. Knuth, "Digital halftones by dot difusion," *ACM Transactions on Graphics*, 6(4):245-273, October 1987.
- [Newman 81]  
William M. Newman and Robert F. Sproull, *McGraw-Hill Book Company*, 1981.
- [Pins 89] Markus Pins and Hermann Hild, *Proceedings Eurographics'89*, pp. 381-392.
- [Rogers 85] David F. Rogers, "Procedural Elements for Computer Graphics," *McGraw-Hill Book Company*, 1985.
- [WN 82] I. H. Witten and R. M. Neal, "Using Peano curves for bilevel display of continuous-tone images", *IEEE Transactions on Computer Graphics and Applications*, Vol. 2, no 3, May 1982.