MSX Article

MARMSX

Otimização da Paleta de Cores

Resumo

O objetivo deste artigo é apresentar a técnica de otimização da paleta de cores utilizada no MSX Viewer 5 para encontrar a melhor paleta de cores do MSX 2 e do v9990, a partir de imagens de 24 bits do PC.

1- Introdução

Quando reduzimos a quantidade de cores de uma imagem, a perda na qualidade é inevitável. Porém, esta perda de qualidade pode ser minimizada através de alguns métodos de redução de cor [1].

No artigo de Error Diffusion [1], foi visto um método para minimizar o problema das Bandas de Mach em imagens quantizadas. Entretanto, há uma nova questão envolvida na quantização de imagens com paleta de cores – encontrar a paleta de cores ideal para aquela imagem.

A quantização de uma imagem para uma paleta de cor específica é baseada naquele conjunto de cores. Dessa forma, caso a imagem original disponha de diversas combinações de cores não presentes nessa paleta, o resultado é uma imagem com pouca diversidade de cores. Um exemplo disso pode ser visto na figura 1, onde a quantização de uma imagem em tons de cinza para as 16 cores nativas do MSX somente utilizará três cores dessa paleta: o preto, o cinza e o branco. Já quando a quantização é feita a partir de uma paleta otimizada para a imagem, ela terá 16 tons de cinza distintos, conforme visto na figura 1c.

a) Imagem original | b) Quantização para as 16 cores do MSX 1 | c) Quantização para 16 cores de 16 milhões

Figura 1. Quantização de imagem com paleta não otimizada e otimizada.

Com o advento do MSX 2, tornou-se possível controlar a intensidade de RGB de cada componente de cor da tela em 8 níveis cada, podendo ser geradas até 512 cores diferentes. Entretanto, as telas em modo de paleta só permitem exibir 16 dessas 512 cores ao mesmo tempo. Assim, podemos adptar a paleta de cores do MSX 2 para as 16 melhores cores que representam a imagem.

Para se atingir tal fim, não seria simplesmente verificar as cores mais freqüentes na imagem? Não, no caso da imagem possuir grandes áreas e com variedade de tonalidades da mesma cor (ex. céu azul), elas prevalesceriam sobre as outras. Assim, certas cores na imagem poderiam não ter representação, recaindo no problema apresentado na figura 1b. Uma solução para esse problema é a representação de cores por agrupamentos.

A proposta dessa pesquisa é utilizar o classificador K-Means para encontrar a melhor combinação de cores da paleta do MSX 2 para representar uma dada imagem.

2- O Classificador K-Means

K-Means [2] é um algorítimo de classificação iterativo, utilizado em *data mining*. A idéia básica é fazer o seguinte: para um dado conjunto $X=(x_1, x_2, \ldots, x_n)$, criar K sub-conjuntos de X. Cada sub-conjunto é representado por um centróide $C=(c_1, c_2, \ldots, c_k)$.

Em cada iteração do K-Means, ele irá:

1. Associar elementos aos centróides: para cada elemento de X, associe-o ao sub-conjunto do centróide mais próximo.
2. Recalcular os centróides: após as associações, calcular os novos valores de centróides, baseado na média dos elementos de cada sub-conjunto.
3. O algorítimo é interrompido quando todos os centróides não se alterarem entre uma iteração e outra.

O conjunto X é uma massa de dados qualquer. Dessa forma, os valores iniciais dos centróides são desconhecidos. Para a solucionar esse problema, pode-se atribuir valores aleatórios a esses centróides. Outro problema, é que o número K de sub-conjuntos é desconhecido. Assim, o usuário deverá fornecer esse número.

Não importando quantas dimensões tiver cada elemento, ele será associado ao centróide pelo valor da distância até ele. O cálculo da distância é através da Distância Euclidiana:

$$d(x_i, c_j) = \sqrt{\sum_{d=1}^{n} (x_{i,d} - c_{j,d})^2}$$

Onde:

i é o i-ésimo elemento de X.
j é o j-ésimo centróide de C.
d é a d-ésima dimensão de ambos.

O exemplo a seguir irá ilustrar o funcionamento do K-Means para um conjunto X bidimensional.

Seja X = { (7, 1), (5, 1), (4, 8), (7, 8), (7, 3), (3, 5), (6, 1), (2, 2), (4, 10), (10, 8) }.

O conjunto X pode ser visto graficamente na figura 2.

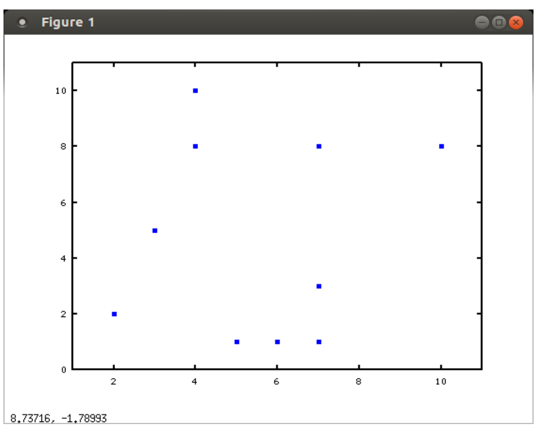


Figura 2. Conjunto X do exemplo.

Suponha que se deseje classificar esse conjunto em dois grupos. Assim, serão utilizados dois centróides para representar os grupos, com valores iniciais aleatórios.

Dado que os valores aleatórios encontrados para *C* sejam:

$C = \{ (2, 4), (9, 9) \}$

O conjunto X (em azul) e o centróides C (em vermelho) podem ser vistos na figura 3.

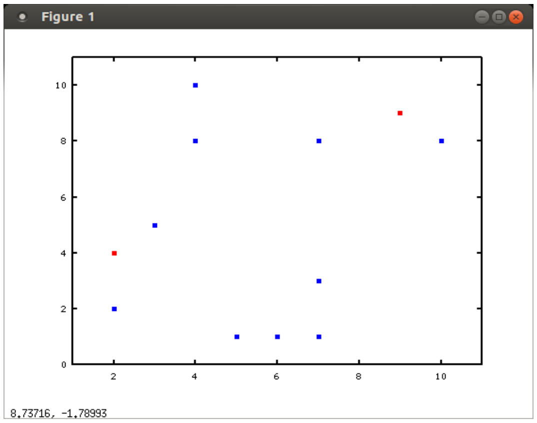


Figura 3. Conjunto *X* e os centróides *C* do exemplo.

A figura 4 apresenta o resultado da classificação dos dados utilizando-se o algorítimo de K-Means, bem como a trajetória dos centróides durante as iterações. O valor acima do centróide indica o número da iteração, onde o valor 0 é a posição inicial do centróide.

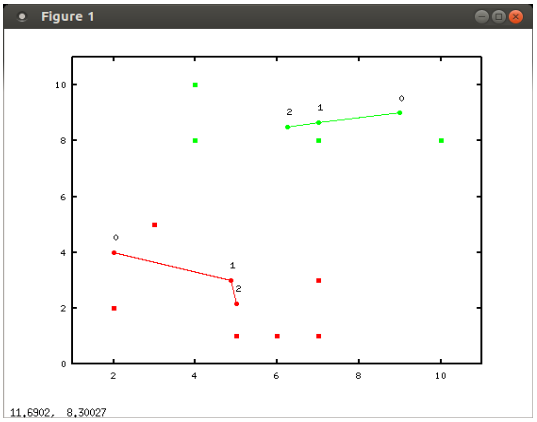


Figura 4. Resultado para a classificação por K-Means.

Foram necessárias duas iterações para que se encontrasse a convergência dos resultados. Os pontos ligados pelas linhas representam a trajetória dos centróides.

Ao final da classificação, dois sub-conjuntos foram encontrados. Eles estão assinalados pelos pontos em verde (grupo 1) e vermelho (grupo 2) na figura 4.

2.1- Os centróides vazios

O K-Means pode apresentar uma situação indesejável, onde a classificação é interrompida e o resultado é inadequado. Trata-se dos centróides vazios. Um centróide vazio é aquele que nenhum elemento do conjunto X está associado a ele.

A causa desse problema pode ser:

- Número K de grupos inadequado
- Má distribuição inicial dos centróides

A solução para o primeiro caso, é modificar o número K de agrupamentos, até que se atinja um número adequado. Já no segundo, a quantidade K é o ideal ou o necessário, como o caso do MSX, que deve ter 16 ou 64 cores. Em tese, apenas reiniciando a classificação solucionaria o problema. Mas, nem sempre essa solução resolve o problema.

Para solucionar o problema de centróides vazios, a proposta é dividir os elementos dos centróides mais "gulosos" com os centróides "famintos". Assim, deve-se:

- Ordenar os centróides de forma descrescente, a partir do número de elementos de X.
- Modificar os valores dos centróides vazios para as coordenadas dos mais populosos, inserindo uma pequena "perturbação", para que os centróides não estejam exatamente na mesma posição (senão, ele continuará vazio).

Para exemplificar o problema dos centróides vazios, sejam os seguintes centróides C para o conjunto X do exemplo anterior:

C = { (2, 4), (19, 19) }

Ao utilizar o calssificador K-Means, todos os elementos serão associados a c_1, visto que c_2 está bem mais distante dos pontos.

Outro exemplo, agora com excesso de grupos:

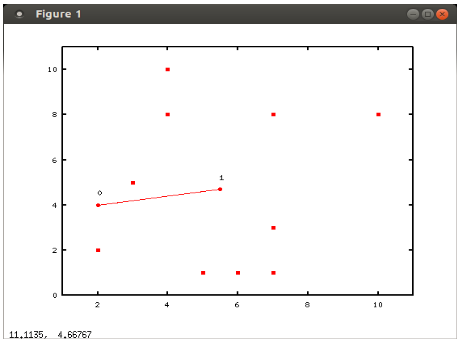
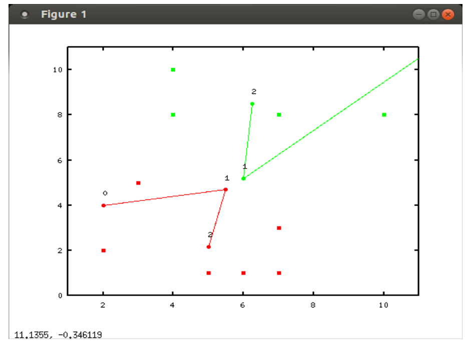
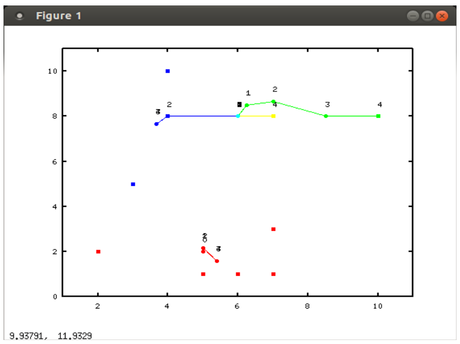
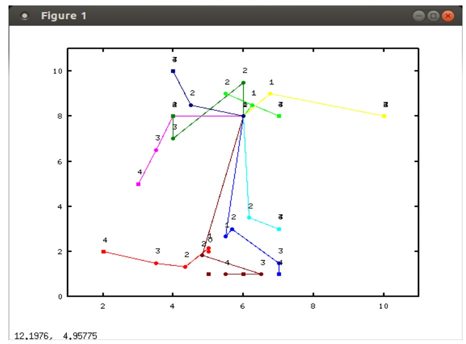
C = { (5, 2), (6, 8), (6, 8), (6, 8), (6, 8), (6, 8), (6, 8), (6, 8), (6, 8) }

Ao classificar X, cada centróide terá os seguintes elementos:

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9
x_1	x_{10}	x_3	x_4					
x_2		x_6						
x_5		x_9						
x_7								
x_8								

Observe que os centróides c_5 até c_9 ficaram vazios, ou seja, aquelas "classes" não tem qualquer representante. Isto representa um desperdício, principalmente quando se trata de cores em um sistema de cores limitado, como o do MSX.

Ao corrigir o problema do centróide vazio, tem-se graficamente para os dois exemplos:

Exemplo	Centróide vazio	Corrigido
1		
2		

Onde a nova classificação do conjunto *X* para o exemplo 2 seria:

C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
x_8	x_4	x_1	x_{10}	x_6	x_5	x_2	x_3	x_9
						x_7		

Agora, todas as 9 classes no exemplo 2 possuem ao menos um elemento de *X*.

3- Otimização da paleta do MSX

O objetivo da aplicação do K-Means em imagens digitais de 24 bits é encontrar os agrupamentos de cor, que melhor representem essa imagem nos sistemas de paleta do MSX 2 e do chip v9990.

A classificação por K-Means da imagem original de 24 bits pode ser feita nesse espaço de cor, ou no espaço de cor correspondente a uma das paletas do MSX. A paleta do MSX 2 possui 512 cores e a do chip v9990 possui 32768 cores. A diferença é que no maior espaço de cor, o ajuste é mais refinado, entretanto, leva mais iterações para convergir.

Em ambos os casos, o ajuste poderá levar a cores repetidas, ou seja, a centróides superpostos. Isto se deve ao processo de quantização ou ao arredondamento dos valores dos centróides, que possuem valores reais. A solução para esse problema será discutida mais adiante.

O conjunto *X* será a imagem de 24 bits do qual será extraída a paleta otimizada. Os centróides de *C* correspondem à paleta de cor. Entretanto, antes de começar a classificação, deve-se modificar a imagem de três dimensões para duas, conforme mostra a figura 5.

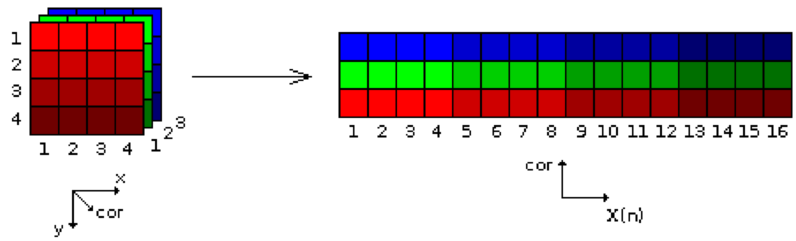


Figura 5: Conversão de matriz para vetor.

No caso da paleta do MSX 2, o número de sub-conjuntos de *X* são 16. Assim, *K*=16. No caso chip v9990, *K*=64. A paleta inicial é composta de valores aleatórios.

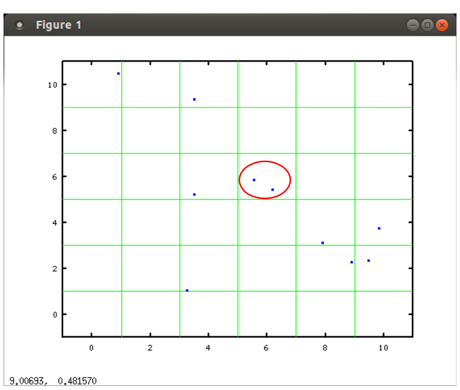
A Distância Euclidiana da cor de um pixel *P* à uma determinada cor da paleta *C* é calculada segundo a equação:

$$d(P,C)=\sqrt{(P_R-C_R)^2+(P_G-C_G)^2+(P_B-C_B)^2}$$

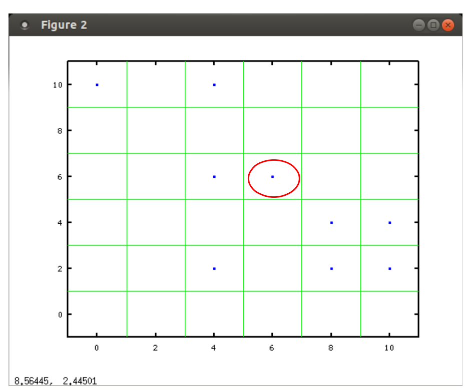
Onde R, G e B são os componentes de cor vermelho, verde e azul, respectivamente.

3.1- Centróides superpostos

A quantização ou o arredondamento dos valores dos centróides podem levar a uma situação indesejada: a de cores repetidas. Isto acontece, quando os centróides possuem valores distintos, mas quando aplicado um desses processos, passam a ter valores iguais. A figura 6 ilustra esse problema.



a) valores originais



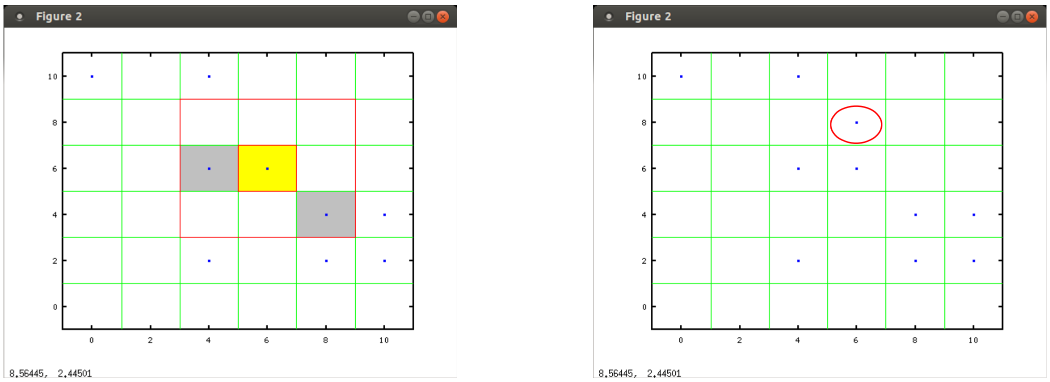
b) quantizado

Figura 6. Superposição de centróides.

Os pontos em azul representam os centróides, enquanto que as linhas verde correspondem à superfície de quantização.

Após a quantização dos valores do centróides, os dois pontos assinalados na figura 6a pelo círculo vermelho ficam superpostos, conforme ilustra o círculo vermelho na figura 6b. Então, o conjunto original de 10 centróides passa a ser, na prática, 9 centróides.

Uma possível solução para esse problema é deslocar um dos pontos superpostos para o espaço quantizado vizinho, desde que este não esteja ocupado. A figura 7 ilustra esse processo.



a) possíveis candidatos b) problema corrigido

Figura 7. Correção do problema de superposição de centróides.

Os centróides superpostos estão assinalados com o fundo amarelo na figura 7a. O retângulo vermelho representa a vizinhança de interesse dessa região. As regiões marcadas com o fundo cinza representam as regiões ocupadas e que não irão servir para deslocar o centróide superposto. Então, foi escolhida uma região livre aleatoriamente, assinalada pelo círculo vermelho, conforme mostra a figura 7b.

3.2- Resultados obtidos

De forma a testar a otimização da paleta de cores do MSX, foi utilizado o MSX Viewer 5. Nesse caso, foi aplicada a quantização de cores através da Distância Euclidiana tanto para a paleta do MSX 1, como para as paletas otimizadas do MSX 2 e V9990.

a) Imagem original b) Paleta do MSX 1 c) Paleta otimizada, *K*=16 d) Paleta otimizada, *K*=64

Figura 8. Resultados obtidos para o MSX, utilizando o MSX Viewer 5.

Quando foi aplicada a quantização de cores sobre a imagem a partir da paleta default do MSX 1, foram encontradas 4 cores: preto, verde, cinza e branco (figura 8b).

Após otimizar a paleta de cores do MSX 2 para a imagem de teste, foi aplicada a quantização sobre a imagem original utilizando-se essa paleta. Os resultados obtidos foram sensivelmente melhores do que a experiência anterior (figura 8c). Nesse caso, foram selecionadas 16 cores de um universo de 512 que melhor representam a imagem.

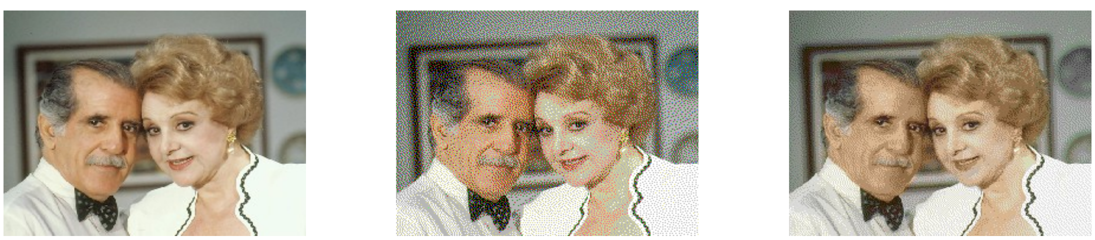
Entretanto, quando foi utilizada a paleta do processador de vídeo V9990 para esse mesmo fim, os resultados foram melhores ainda (figura 8d). Aqui, utilizaram-se 64 cores de um total de 32768.

Uma vez otimizada a paleta de cores, é possível aplicar o método de Error Diffusion sobre a imagem em vez da quantização. Nesse caso, os resultados são melhores ainda, conforme ilustra a figura 9. Para esse teste foi utilizada a paleta do MSX 2.

a) Imagem original | b) Quantização sobre paleta otimizada | c) Error Diffusion sobre paleta otimizada

Figura 9. Comparação entre a quantização e o Error Diffusion aplicados a uma paleta do MSX 2 otimizada.

Quando aplicado o método de Error Diffusion sobre a imagem original, a otimização da paleta de cores também produz melhores resultados do que a utilização da paleta default do MSX 1, conforme mostra a figura 10.



a) Imagem original | b) Error Diffusion com paleta do MSX 1 | c) Error Diffusion com paleta otimizada

Figura 10. Combinação das técnicas de Error Diffusion e otimização de paleta.

A imagem da figura 10c corresponde à screen 5 do MSX 2, com 16 cores de 512.

4- Créditos e Bibliografia

Este artigo foi escrito por Marcelo Silveira, Engenheiro de Sistema e Computação, formado pela Universidade do Estado do Rio de Janeiro.

Escrito em: Julho de 2017, revisado em Fevereiro de 2018.
E-mail: flamar98@hotmail.com
Homepage: http://marmsx.msxall.com

Referências Bibliográficas:

[1] – Error Diffusion, artigo, Marcelo Silveira em MarMSX Development. http://marmsx.msxall.com
[2] – K-Means Clustering, Wikipedia. http://en.wikipedia.org/wiki/K-means_clustering
[3] – MSX Viewer 5 Appendix, http://www.marmsx.msxall.com