

Fundamentos de Imagens Digitais

O aparecimento das imagens digitais é resultante do processo de desenvolvimento das ciências da informação. As primeiras imagens digitais surgiram na área da astronomia e progressivamente foram-se expandindo para outras áreas, tal como a medicina até que finalmente chegaram ao público em geral.

Curiosidades:

Em 1957, Russel Kirsch, no NBS, criou a primeira imagem digital.

Em 1975, a Kodak lançou a primeira máquina fotográfica digital. Esta máquina tirava apenas fotografias a preto e branco com uma resolução aproximadamente de 100×100 pixels (0,01 megapixels), pesava 3,6 quilogramas e demorava 23 segundos para tirar uma única fotografia.



Uma imagem digital é uma imagem bidimensional, que emprega um código binário de modo a permitir o seu processamento, transferência, impressão ou reprodução.

Há dois tipos principais de imagem digital:

Imagens de rastreio (raster) e imagens vectoriais.

As imagens raster são imagens formadas por um conjunto de pontos definidos por valores numéricos, no qual cada ponto representa um pixel. A desvantagem deste tipo de imagem é que ao aumentarmos as medidas da imagem em questão, os pontos irão repartir por uma área maior, tornando a imagem mais indefinida. As imagens raster são geralmente imagens fotográficas.

As imagens vectoriais são originadas a partir de desenhos (retas, pontos, curvas, polígonos simples, etc). A vantagem deste tipo de imagem é que esta pode ser aumentada sem perda de qualidade, ao contrário da raster e o seu tamanho costuma ser menor que as raster. As imagens vectoriais costumam ser usadas para desenhos técnicos de engenharia.

Definição de Pixel

Pixel é elemento mais pequeno que forma uma imagem digital, sendo o conjunto de milhares de pixels que compõem uma imagem. Geralmente, é concedida uma cor a cada pixel.



O pixel não precisa apresentar forçosamente a forma de um pequeno quadrado. A palavra pixel pode significar diversas realidades, conforme o contexto em que o termo se encontra inserido. Por exemplo, o pixel pode:

- Ser representado por valores numéricos: Pixels por polegada, Pixels por linha, Número de pixels existentes numa câmera digital, etc.
- Ser considerado uma medida (“A imagem é de 1024×728 pixels”)

É importante salientar que um megapixel corresponde a um milhão de pixels.

Cores Digitais

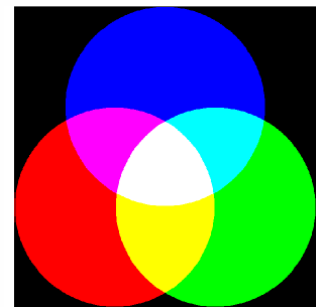
A cor é considerada uma percepção visual provocada pela ação dos feixes de fótons sobre células da nossa retina que transmitem sensações para o nosso sistema nervoso. E todos nós vemos as cores de forma subjetiva.

As cores das imagens digitais são definidas através de diversos modelos de cor.

Os modelos de cor são sistemas usados para organizar e definir cores através de um conjunto de propriedades básicas que são reproduzíveis. Estes foram desenvolvidos de maneira a uniformizar a forma de como as cores são especificadas em formato digital, de modo a produzi-las com rigor em qualquer tipo de hardware. Abaixo alguns modelos de cor.

RGB

O modelo de cores RGB é um sistema de cores aditivo (as cores são formadas adicionando luz a cada uma das cores intervenientes no processo), inspirado na teoria de visão colorida tricromática e tem como base as cores: vermelho, verde e azul. Este modelo pode produzir até 16,7 milhões de cores. Este é o modelo mais usado e conhecido.



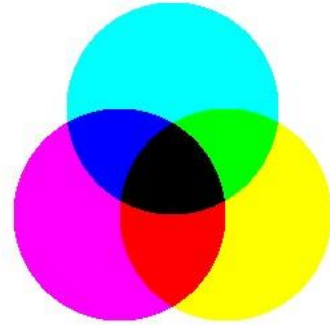
Como funciona? **Uma cor no modelo RGB é formada através da indicação da quantidade de vermelho verde e azul que esta contém.**

Cada uma destas três cores pode variar entre o mínimo e o máximo de uma escala de 0 a 255. Quando todas as cores estão no número zero, o resultado é preto. Se todas estão no máximo, o resultado é branco. Abaixo alguns exemplos:

- Branco – RGB(255,255,255)
- Azul – RGB(0,0,255)
- Vermelho – RGB(255,0,0)
- Verde – RGB(0,255,0)
- Amarelo – RGB(255,255,0)
- Magenta – RGB(255,0,255)
- Ciano – RGB(0,255,255)
- Preto – RGB(0,0,0)

CMYK

O modelo CMYK tem como base as cores primárias (azul ciano, magenta, amarelo) e o preto. É usado geralmente em impressoras e fotocopiadoras. Neste modelo cada cor é descrita com uma porcentagem (de 0% a 100%), sendo que quanto maior for a porcentagem, mais escura a cor será. Este modelo é um modelo subtrativo de cores pois cria cores absorvendo luz.



Razões para que o preto fosse introduzido neste modelo:

- O preto produzido pelas cores primárias não ser puro.
- O pigmento preto é o mais barato de todos
- Usar as três cores primárias para formar o preto faria com que os textos imprimidos demorassem muito tempo a secar

LAB

O modelo LAB é um modelo matemático que foi criado em 1931 pela CIE (La Commission Internationale de L'Eclairage) e foi inspirado na forma como o olho humano percebe as cores. Em 1976, o modelo foi melhorado de forma a garantir cores mais consistentes, independentemente das características do hardware. Este modelo funciona através de um canal de luminosidade e de dois canais de cor (a e b).

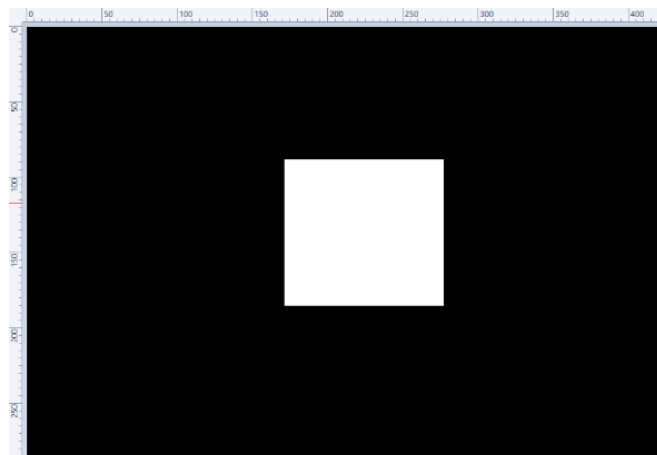
GetColor Python e OpenCV

O algoritmo abaixo denota a função específica para pegar uma cor de determinado pixel da imagem.

```
import cv2
import numpy as np

image = cv2.imread("pretoebranco.png")
print("Pixel 10x10:", image[0,0])
eixox = 224
eixoy = 139
print("Pixel 224x139:", image[eixoy,eixox])

cv2.imshow("Imagem original", image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```



A conversão de cores pode ser realizada utilizando a ferramenta da w3c:

https://www.w3schools.com/colors/colors_rgb.asp

Resolução Espacial

Ao ser digitalizada a imagem assume um tamanho dimensional, em pixels. Mas, pode-se conhecer uma medida de qualidade da amostragem, conhecendo-se a razão entre o número de pixels obtido e o tamanho da imagem real no filme ou equivalente. A isso chama-se de Resolução. Em geral, é medida em pontos por polegada ou DPI (dots per inch), mas pode ser também em pontos por centímetro ou DPC, ou ainda em qualquer outra unidade equivalente.

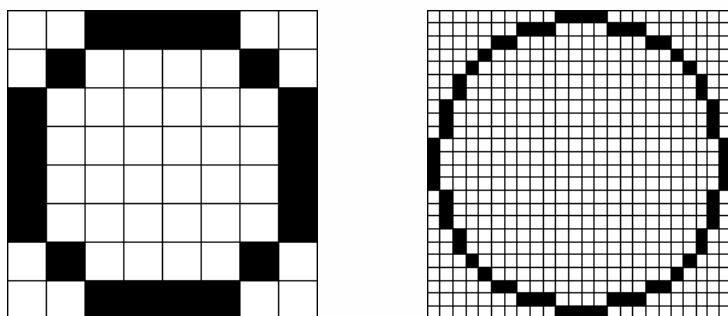
Também é simples estabelecer a relação: **número de pixels = resolução x tamanho real**. Para entender melhor isso, pode-se fazer a seguinte questão: Dado o tamanho da imagem (em cm, in, etc...), quantos valores discretos vão ser tomados? Escolhendo-se a resolução a resposta é dada pela equação acima.

Veja algumas resoluções típicas:

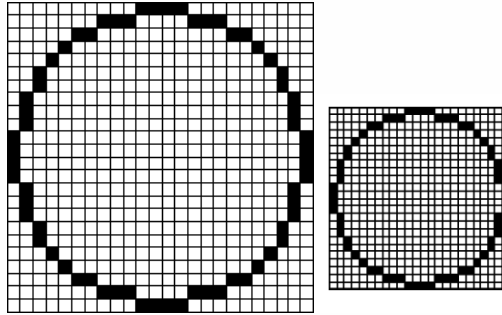
- Monitor Comum - 72 DPI;
- Scanner - 1200 DPI;
- Impressora Jato de Tinta Popular - 600 DPI

Como temos duas dimensões, podemos definir uma resolução horizontal e uma vertical. Quando nada se diz a respeito disso, quer dizer que são iguais (a grande maioria das vezes), caso contrário são necessários dois valores. Uma vez que são diferentes, a razão de aspectos da área entre um pixel e outro não é 1:1 e, portanto, não é um quadrado. Isso gera algumas confusões, pois no monitor a razão é 1:1 e esta imagem quando visualizada aparecerá deformada.

Quando duas imagens com tamanhos reais iguais são capturadas com resoluções diferentes, naturalmente terão número de pixels diferentes e na tela aparecerão com tamanhos diferentes. Na figura a seguir mostramos um círculo amostrado em duas resoluções, mas exibido no seu tamanho original.



Quando duas imagens de tamanhos reais diferentes são capturadas com resoluções diferentes de tal forma que gerem imagens digitais com o mesmo número de pixels, quando visualizadas no monitor aparecerão com o mesmo tamanho na tela. No exemplo a seguir mostramos as imagens proporcionais ao seu tamanho original.



Quando queremos realizar alguma medida sobre os objetos contidos na imagem surge um outro parâmetro, a escala. Pode-se defini-la matematicamente idêntica a resolução só que as unidades serão as mais variadas possíveis. Caso não seja conhecida, é fácil de se obtê-la.

Mede-se algumas distâncias reais na imagem digital e calcula-se a média das razões entre distância e número de pixels, obtendo-se a escala. A partir desse momento outras distâncias podem ser feitas na imagem com qualidade aceitável. É claro que se os objetos não estejam em um mesmo plano, as medidas são muito mais complexas e envolvem transformações projetivas.

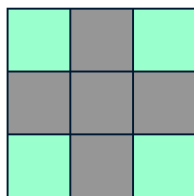
Topologia de um Pixel

As amostras do plano são pontuais e adimensionais, mas os dispositivos para reconstrução não deixam um espaço entre uma amostra e outra. Assim, muitas vezes iremos representar o pixel, não como um ponto, mas como um quadrado preenchido com a cor do pixel, que está no seu centro.

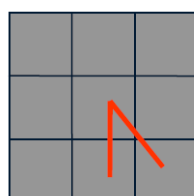
Por sinal, o exemplo anterior já usa essa representação. Como vimos, pixels são normalmente quadrados, gerando uma grade (grid) regular devido à amostragem uniformemente espaçada. Um problema que aparece nesse momento é:

Quais são os vizinhos de um determinado pixel? Essa pergunta é fundamental para algoritmos de preenchimento de áreas e algoritmos de determinação de continuidade de objetos na imagem.

Nesta grade regular podemos definir dois tipos de vizinhança: 4 conectada (onde os vizinhos são equidistantes) ou 8 conectada (onde nem todos são equidistantes, os vizinhos na diagonal são mais distantes que os vizinhos nas laterais). Normalmente se usa uma combinação das duas vizinhanças, definindo 4 conectada para os objetos e 8 conectada para o fundo.

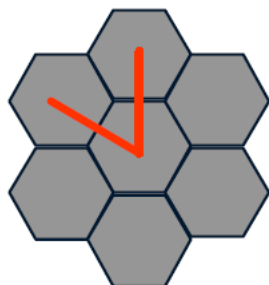


Vizinhança 4 Conectada



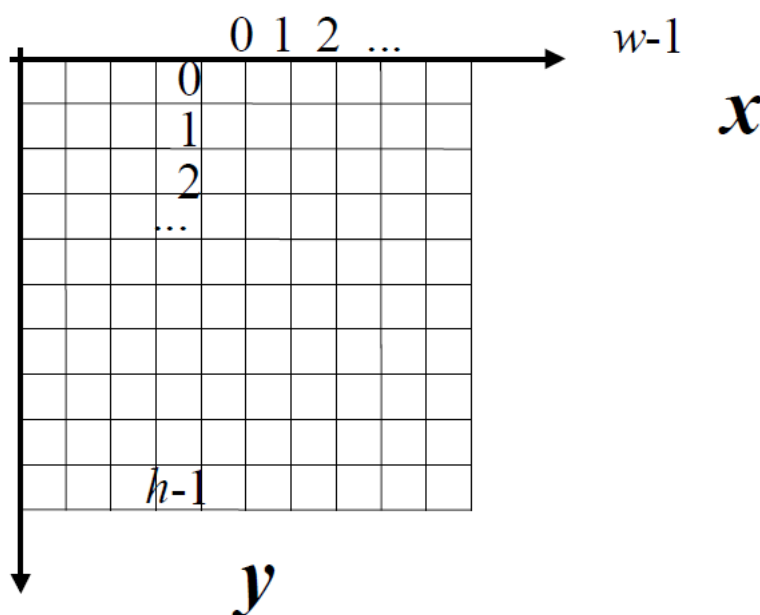
Vizinhança 8 Conectada

Para criar uma vizinhança onde todos os vizinhos fossem equidistantes e só houvesse uma definição, temos que mudar a grade regular para uma grade hexagonal. São poucos algoritmos implementados com essa vizinhança, pois como a imagem está em geral amostrada em uma grade regular, deve-se convertê-la para a grade hexagonal, realizar o processamento e convertê-la de volta para a grade regular.



Vizinhança em um Grid Hexagonal

Já que estamos falando da grade não poderíamos deixar de falar sobre a origem do sistema de coordenadas. Localizada no canto superior esquerdo na grande maioria dos sistemas de processamento de imagem, pode na realidade ser posicionada em qualquer lugar da imagem, até mesmo fora dela.



Essa localização preferencial para o canto superior esquerdo, com eixo y invertido em relação ao padrão da matemática, é herdada dos próprios dispositivos de visualização, onde a primeira linha a ser redesenhada na tela é a do topo da tela, que segue o padrão de escrita ocidental, da esquerda para a direita e de cima para baixo.

GetResolution e UpdateColor Python e OpenCV

```
import cv2

import numpy as np

image = cv2.imread("batman.png")

h,w,bpp = np.shape(image)

print("Altura:", h) #Altura

print("Largura:", w) #Largura

print("Profundidade:", bpp) #profundidade da cor
```

Também é possível criar cores através de diretrizes de array.

```
# o padrão no CV2 é BGR

preto = np.array([0,0,0])

azul = np.array([127,84,38])

image[100,20] = azul
```

UpdateColor Conjunto de Pixels

```
import cv2

import numpy as np

image = cv2.imread("batman.png")

h,w,bpp = np.shape(image)

print("Altura:", h) #Altura

print("Largura:", w) #Largura

print("Profundidade:", bpp) #profundidade da cor

preto = np.array([0,0,0])

azul = np.array([127,84,38])

for i in range(0,h):

    for j in range(0,w):

        if np.array_equal(image[i,j],preto):

            image[i,j] = azul

cv2.imshow("Imagem original", image)

cv2.waitKey(0)

cv2.destroyAllWindows()
```