

Fundamentos da Imagem Digital

Antonio Escaño Suri

(suri@tecgraf.puc-rio.br)

Tecgraf/PUC-Rio
Setembro de 2002

Sumário

| | | |
|-----------|---|-----------|
| 1. | Introdução | 8 |
| 1.1. | Estruturação do Texto | 9 |
| 2. | Conceitos Básicos..... | 11 |
| 2.1. | Computação Gráfica e Processamento de Imagens | 11 |
| 2.2. | Tópicos de Processamento de Imagens..... | 13 |
| 2.3. | Contínuo x Discreto | 13 |
| 2.4. | O Paradigma dos Quatro Universos..... | 16 |
| 3. | Resolução Espacial | 21 |
| 3.1. | Topologia de um Pixel | 23 |
| 3.2. | Reconstrução | 25 |
| 3.3. | Aliasing | 27 |
| 4. | Cor..... | 30 |
| 4.1. | Processos de Formação de Cor | 32 |
| 4.2. | Amostragem e Reconstrução de Cor..... | 33 |
| 4.3. | Cor no Paradigma dos 4 Universos..... | 35 |
| 4.4. | Padronização para Cor | 36 |
| 4.5. | Diagrama de Cromaticidade e Propriedades de Cor | 39 |
| 4.6. | Sistemas de Cor..... | 43 |
| 4.7. | Sistemas de Gerência de Cor | 49 |
| 4.8. | Quantização com Menos de 24 bits | 51 |
| 5. | Resolução Temporal | 54 |
| 6. | Captura..... | 58 |
| 6.1. | Scanners | 58 |
| 6.2. | Digitalizadores de Sinal de Vídeo..... | 59 |

| | | |
|------------|--|-----------|
| 6.3. | Câmeras Digitais | 60 |
| 6.4. | Parâmetros de Câmeras Digitais | 61 |
| 6.5. | Escolhendo um Sistema de Captura..... | 63 |
| 6.6. | Transferindo Imagens Digitais..... | 63 |
| 7. | Visualização..... | 64 |
| 7.1. | Monitores Analógicos | 66 |
| 7.2. | Monitores Digitais Planos e Projetores | 68 |
| 7.3. | Impressoras | 69 |
| 8. | Armazenamento..... | 71 |
| 8.1. | Compressão | 74 |
| 8.2. | Formatos de Arquivo | 76 |
| 8.3. | Arquivos de Vídeo | 77 |
| 8.4. | Imagens Grandes e Multiresolução..... | 78 |
| 9. | Processamento..... | 80 |
| 9.1. | Restauração/Realce | 81 |
| 9.2. | Segmentação | 81 |
| 9.3. | Extração de Atributos..... | 82 |
| 9.4. | Classificação | 82 |
| 9.5. | Operações Pontuais | 83 |
| 9.6. | Operações Locais | 85 |
| 9.7. | Operações Globais | 86 |
| 10. | Software..... | 87 |
| 11. | Aplicações | 90 |
| 12. | Internet Links | 93 |
| 13. | Bibliografia..... | 95 |

Figuras

| | |
|--|----|
| Figura 1: Áreas da Computação Gráfica..... | 12 |
| Figura 2: Tópicos de Processamento de Imagens | 13 |
| Figura 3: Gráfico de uma Função Unidimensional | 14 |
| Figura 4: O Processo de Amostragem..... | 15 |
| Figura 5: O Processo de Quantização | 15 |
| Figura 6: O Paradigma dos Quatro Universos | 17 |
| Figura 7: O Paradigma para Imagens..... | 18 |
| Figura 8: Gráfico de uma Função Bidimensional | 18 |
| Figura 9: Processo de Discretização de uma Imagem..... | 20 |
| Figura 10: Imagens com Mesmo Tamanho Real e Diferentes Resoluções..... | 22 |
| Figura 11: Imagens com Mesma Resolução e Tamanho Real Diferentes | 22 |
| Figura 12: Vizinhança em um Grid Regular | 23 |
| Figura 13: Vizinhança em um Grid Hexagonal | 24 |
| Figura 14: Sistema de Coordenadas de uma Imagem | 24 |
| Figura 15: Eixos Mais Comuns..... | 25 |
| Figura 16: Interpolação de uma Função Discreta Unidimensional..... | 26 |
| Figura 17: Interpolação Vizinho mais Próximo versus Bilinear..... | 26 |
| Figura 18: <i>Aliasing</i> Unidimensional | 27 |
| Figura 19: <i>Aliasing</i> Bidimensional | 28 |
| Figura 20: Padrão de Moiré | 28 |
| Figura 21: Anti- <i>Aliasing</i> em Desenho de Primitivas | 29 |
| Figura 22: Distribuição Espectral de Potência | 30 |
| Figura 23: Comprimentos de Onda do Espectro Eletromagnético | 31 |
| Figura 24: Cor no Paradigma dos Quatro Universos. | 36 |

| | |
|---|----|
| Figura 25: Experimento para Obtenção das Funções de Reconstrução de Cor | 37 |
| Figura 26: Funções de Reconstrução de Cor no Sistema CIE-XYZ..... | 39 |
| Figura 27: Detalhes do Diagrama de Cromaticidade | 40 |
| Figura 28: Mudanças na Distribuição Espectral de uma Cor..... | 41 |
| Figura 29: Correção Gamma..... | 42 |
| Figura 30: Cubo RGB | 46 |
| Figura 31: Modelo HSB | 47 |
| Figura 32: Modelo de Munsell..... | 49 |
| Figura 33: Espaço de Cor de Dois Monitores | 49 |
| Figura 34: Exemplos de Quantização com 256 Cores | 52 |
| Figura 35: Tempo de Redesenho de Quadros em Cinema..... | 55 |
| Figura 36: Conversão 3-2 de Cinema para Vídeo..... | 57 |
| Figura 37: Parâmetros de uma Câmera Fotográfica..... | 61 |
| Figura 38: Efeito da Vizinhança na Percepção | 65 |
| Figura 39: Comparação entre Razões de Aspectos Tradicionais | 66 |
| Figura 40: Tempo de Redesenho de um CRT versus um LCD | 68 |
| Figura 41: Macro Fotografias de Monitores CRT e LCD respectivamente..... | 69 |
| Figura 42: A Mesma Imagem em Diferentes Resoluções e Número de Cores..... | 72 |
| Figura 43: Efeito Perceptual das Bandas de Mach | 75 |
| Figura 44: Uma Imagem em Multiresolução | 79 |
| Figura 45: Seqüência de Processamento de Imagens..... | 80 |
| Figura 46: Operação Pontual..... | 83 |
| Figura 47: Função de mapeamento de Tonalidades..... | 84 |
| Figura 48: Histograma de uma Imagem Monocromática. | 85 |
| Figura 49: Operação Local..... | 85 |

Equações

| | |
|--|----|
| Equação 1: Definição de uma Função Real Unidimensional..... | 13 |
| Equação 2: Definição | 16 |
| Equação 3: Definição de Imagem Contínua | 18 |
| Equação 4: Modelo para o Receptor | 33 |
| Equação 5: Modelo para o Emissor | 34 |
| Equação 6: Cálculo da componente C_i para um par Receptor-emissor. | 34 |
| Equação 7: Definição da Cor de Referência e Componentes Normalizadas. | 34 |
| Equação 8: Obtenção das Componentes Normalizadas. | 34 |
| Equação 9: Reconstrução de uma Cor para um par Receptor-emissor | 35 |
| Equação 10: Conversão entre CIE-RGB e CIE-XYZ..... | 39 |
| Equação 11: Cálculo de Luminância a partir das Primárias RGB | 43 |
| Equação 12: Cálculo de Luma a partir das Componentes não lineares $R'G'B'$ | 43 |
| Equação 13: Diferença Euclidiana para duas Cores no Sistema RGB | 44 |
| Equação 14: Distância Ótima para Visualização | 64 |
| Equação 15: Resolução Ótima para Visualização..... | 64 |
| Equação 16: Tamanho de uma Imagens em Bytes | 72 |

Tabelas

| | |
|---|----|
| Tabela 1: Valores de Iluminamento em Lux para Determinados Ambientes | 41 |
| Tabela 2: Sistema de Vídeo em vários Países..... | 45 |
| Tabela 3: Cortes no modelo HLS..... | 48 |
| Tabela 4: Matrizes de Dithering para 5 Tons..... | 53 |
| Tabela 5: Iluminação Ambiente e Taxa de Redesenho..... | 55 |
| Tabela 6: Comparação entre Dois Modos de Transmissão em HDTV | 57 |
| Tabela 7: Resoluções Ótimas Aproximadas para Distâncias Comuns | 65 |
| Tabela 8: Mapeamento de Índice na Tabela de Cores | 74 |
| Tabela 9: Formatos de Arquivos e Respectivos Sistemas de Cor e Compressão | 76 |
| Tabela 10: Compressão de Vídeo | 78 |

1. Introdução

Apresentamos aqui os conceitos básicos envolvidos em Processamento de Imagens Digitais e procuramos dar uma visão geral da área. Mas, não serão apresentadas técnicas de processamento em profundidade, nem pelo lado matemático, nem pelo lado computacional.

Por isso mesmo, serve tanto a pessoas da área técnica que estão iniciando seu aprendizado em Processamento de Imagens, ou pessoas de Editoração Eletrônica que queiram aprender os conceitos que estão por traz de programas como o Adobe PhotoShop e o Corel PhotoPaint.

Assim, qualquer um com conhecimentos básicos em computadores, pode compreender os conceitos aqui apresentados. Mas, o conceito matemático de função é largamente utilizado e é fundamental para o entendimento de algumas noções.

Os conceitos apresentados servem também como uma preparação para um texto mais avançado de Processamento de Imagens. Por esse motivo vários conceitos são tratados superficialmente para não tornar o texto ilegível para pessoas que não sejam da área técnica. Mas, uma das maiores dificuldades aqui é definir o que deve realmente ser superficial e o que não deve. Isso só a evolução desse material, através do retorno de pessoas que tenham acesso a ele, poderá dizer.

1.1. Estruturação do Texto

Dividimos a apresentação em 3 partes: tópicos básicos, tópicos complementares e tópicos de técnicas e aplicações. Sugerimos que sejam vistos inicialmente os tópicos básicos para melhor compreensão dos demais.

Conceitos Básicos

Apresentamos alguns dos conceitos básicos necessários para o entendimento dos outros tópicos. Situamos “Processamento de Imagens” e “Computação Gráfica”, apresentamos o “Paradigma dos Quatro Universos”, e mostramos algumas noções de contínuo e discreto. E veremos uma definição do importante conceito de *aliasing*.

Resolução Espacial

Quando realizamos a passagem para o mundo digital, a primeira grande etapa para imagens é a amostragem. Aqui vamos explorar um pouco mais essa questão, assim como seu problema inverso, o de reconstrução da imagem original. Veremos que reconstrução é sinônimo de interpolação.

Cor

O que é cor? Como representar cor no computador? Quais os padrões estabelecidos? Essas e outras perguntas serão abordadas aqui de maneira introdutória. Falaremos também sobre Luminância e Crominância, assim como outras propriedades de cor, e por fim veremos algumas técnicas de quantização.

Resolução Temporal

Quando acrescentamos tempo como mais uma variável a ser amostrada, temos imagens que descrevem um movimento ou uma variação de estado. E temos problemas semelhantes aos de amostragem espacial.

Captura

Apresentamos alguns dos dispositivos usados para passarmos a imagem para o mundo digital. No caso *Scanners*, *Frame/Vídeo Grabbers* e Câmeras Digitais.

Visualização

Uma vez a imagem tendo sido digitalizada, é imediato o desejo de reconstruí-la em alguma superfície de visualização. As mais comuns são monitores e impressoras. Sobre estes mostraremos algumas peculiaridades.

Armazenamento

Embora existam vários formatos de arquivos de imagem (TIFF, PCX, GIF, JPEG) todos armazenam a mesma imagem digital. Para falar sobre isso, mostraremos antes como uma imagem é armazenada na memória do computador.

Processamento

As funções de processamento de imagens digitais podem ser classificadas em duas classes, por escopo ou por resultado. Por escopo, está relacionado com técnicas de processamento. Por resultado, está relacionado com os tipos de resultado de cada técnica.

Software

Aqui procuramos dar uma visão geral de alguns programas de processamento de imagens, procurando situá-los dentro do contexto dos conceitos apresentados. Tais como Adobe PhotoShop, Corel PhotoPaint, Paint Shop Pro, Lview Pro, ACDSee, XV, GIMP.

2. Conceitos Básicos

2.1. Computação Gráfica e Processamento de Imagens

Antes de mais nada, precisamos nos situar dentro do contexto de Computação Gráfica.

Estamos trabalhando basicamente com dois tipos de informação: Visual e Descriitiva. Informação visual é por exemplo a imagem vista na tela do computador. Informação descritiva é referente ao modelo matemático que representa os objetos visualizados. A área de **Processamento de Imagens** em geral abrange operações que são realizadas sobre imagens e que resultam em imagens. A área de **Visualização** abrange as operações de síntese de imagem, ou seja, a geração de uma visualização do modelo. A área de **Visão** abrange as operações de análise dos objetos definidos pela imagem e a geração de modelos matemáticos desses objetos.

Aqui iremos ver várias funções de Processamento de Imagens e um pouco da interface desta área com as áreas de Visão e Visualização.

Os programas de arquitetura, design e simulação gráfica, tais como as utilizadas em filmes com animação por computador abrangem as áreas de Modelagem e Visualização. Programas comerciais que representam muito bem esta área são o AutoCad e o Studio 3D, ambos da AutoDesk, e o Corel Draw da Corel.

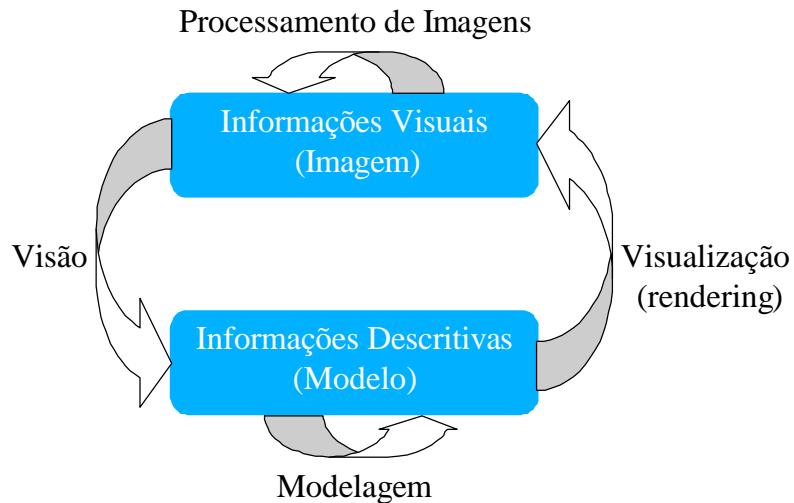


Figura 1: Áreas da Computação Gráfica

Desta maneira, o modo de encararmos os programas das diversas áreas se torna diferente de acordo com o nosso objetivo. Por exemplo, a planta baixa de uma casa pode ser desenhada a partir de suas medidas, sem a necessidade de uma fotografia aérea da casa ou, até mesmo, da existência da casa. Por outro lado, para analisarmos uma amostra de bactérias no microscópio precisamos da imagem da amostra, de onde podemos obter alguma informação, sem a necessidade de conhecermos a posição exata de cada bactéria.

Esta diferença de objetivos vai levar à modelos de trabalho diferentes. Em processamento de Imagens usa-se o modelo de matricial e a computação gráfica geralmente se baseia no modelo de objetos vetoriais. Neste modelo os objetos são armazenados apenas a partir da descrição das coordenadas de seus vértices, sejam elas espaciais ou planares (três ou duas dimensões, respectivamente). Dessa maneira utiliza-se um sistema de coordenadas Cartesiano, onde os objetos podem ser escalados, rotacionados e transladados com maior liberdade para cada objeto. O modelo matricial utiliza uma matriz de dados para armazenar a informação de cor em cada ponto da imagem, onde o sistema de coordenadas é obviamente uma grade de números inteiros que descrevem a posição na matriz. Portanto, no modelo matricial não há distinção dos objetos contidos na imagem. Além disso, armazenar a matriz que contém a imagem geralmente exige muito mais memória que armazenar a descrição vetorial.

A diferença de modelos fica explícita quando apresentamos o paradigma dos 4 universos, que nos ajudará a situarmos todos esses conceitos.

2.2. Tópicos de Processamento de Imagens

Uma vez tendo nos situado no contexto de Processamento de Imagens podemos agora desdobrar essa área nos tópicos que serão abordados. O ponto central de discussão será, é claro, a Imagem Digital. Falaremos sobre dois aspectos importantes que definem a imagem: Resolução e Cor. Feito isso, podemos atacar os pontos de Captura (entrada) e Visualização (saída), Armazenamento e por fim Processamento. Esses outros pontos tem por objetivo esclarecer algumas confusões na área e preparar o aluno para cursos mais avançados.

A figura a seguir ilustra a relação entre os diversos tópicos.

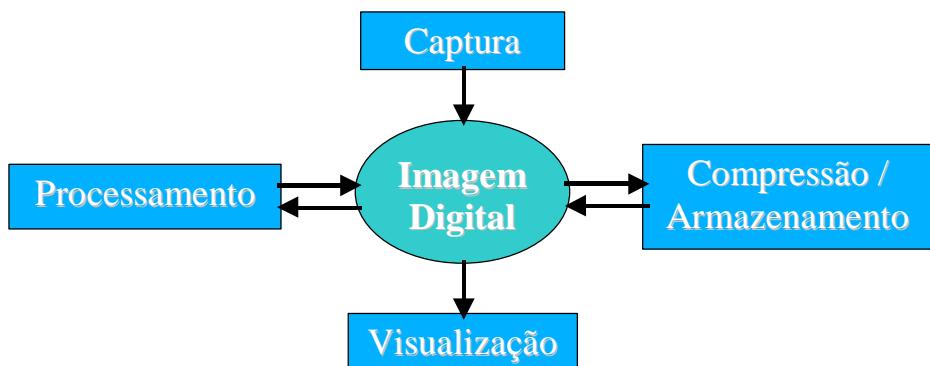


Figura 2: Tópicos de Processamento de Imagens

2.3. Contínuo x Discreto

Para podermos introduzir o conceito de imagem digital é necessário antes falarmos sobre funções contínuas e funções discretas. Podemos pensar em uma função contínua como um corda sem emendas que se estende indefinidamente (ou seja infinita) e por mais que você corte essa corda, sempre poderá cortá-la um pouco menor indefinidamente (ou seja infinitesimal). Matematicamente estamos falando de uma função real definida na reta toda (em \mathbb{R}), onde para cada valor de x fornecido sempre existe um único valor de $f(x)$ obtido.

Formalmente:

$$\begin{aligned} f : \mathbb{R} &\rightarrow \mathbb{R} \\ x &\rightarrow f(x) \end{aligned}$$

Equação 1: Definição de uma Função Real Unidimensional

Colocando isso em um gráfico:

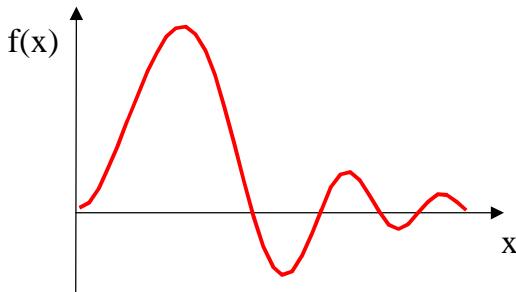


Figura 3: Gráfico de uma Função Unidimensional

O computador só é capaz de armazenar bits, um valor que pode ser 0 ou 1. Para simplificar as coisas juntou-se 8 bits formando uma palavra chamada de byte. Um byte então pode assumir $2^8 = 256$ valores diferentes, variando de 0 a 255. Quando falamos de sistemas operacionais de 16 bits e 32 bits, estamos falando da maior palavra que aquele sistema pode processar de uma vez, por isso sistemas de 32 bits são mais eficientes que os de 16.

Assim, o computador trabalha sempre com números inteiros ou na melhor das hipóteses com uma aproximação de um número real, chamada de ponto flutuante (isso porque o número de bits dedicado para as casas decimais é flexível de acordo com o número). Por consequência, não é possível representar uma função contínua no computador. Podemos apenas simula-la.

O processo para trazer uma função contínua para o computador é discretizando-a (ou digitalizando-a), ou melhor, tomando valores pontuais ao longo de x e guardando o valor de $f(x)$ correspondente (é claro que o eixo $f(x)$ também é contínuo, assim também precisaremos discretizá-lo). O processo de discretização do eixo x (o domínio) é chamado de Amostragem, o do eixo $f(x)$ (o contradomínio) é chamado de Quantização.

Em resumo, a discretização de qualquer sinal contínuo primeiro passa por uma amostragem e depois passa por uma quantização. Além disso, não podemos armazenar um sinal que se estenda indefinidamente, portanto o sinal digital também é limitado a um intervalo do domínio.

A amostragem que vemos na figura é a mais comum de todas e mais popular, implementada na grande maioria de dispositivos de captura. É chamada de uniformemente espaçada, pois cada amostra é tomada em intervalos iguais. Embora existam também técnicas de amostragem que utilizam menos amostras onde a função é monótona e mais amostras onde a função apresenta mais irregularidades.

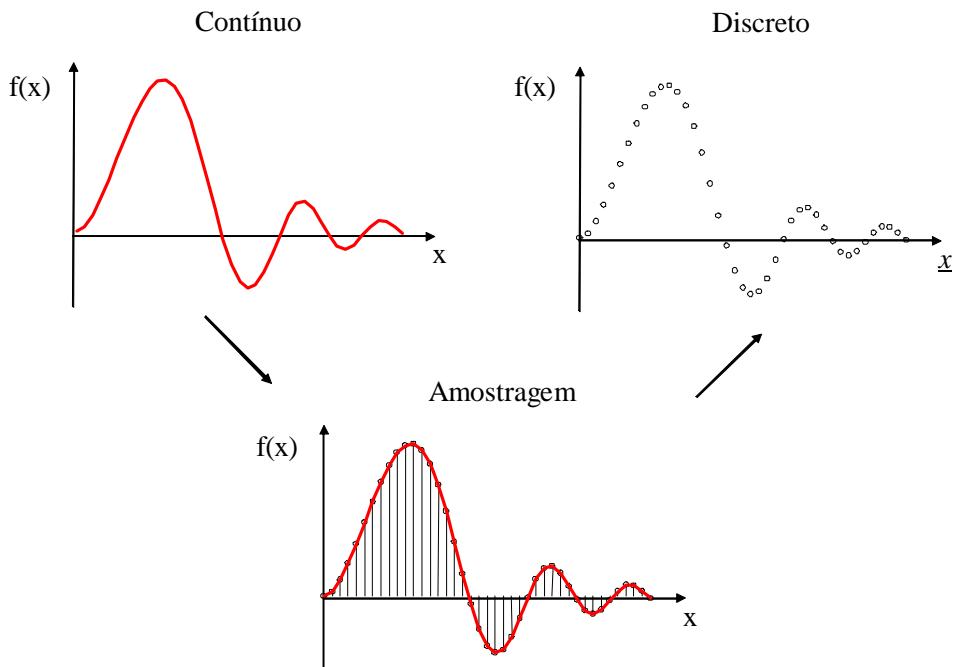


Figura 4: O Processo de Amostragem

A quantização mais comum consiste em tomar o valor máximo e o valor mínimo da função, e dividir este segmento em intervalos iguais de acordo com o número de bits definido para armazenar uma amostra. Assim, o número de valores possíveis será $2^{n\text{bits}}$.

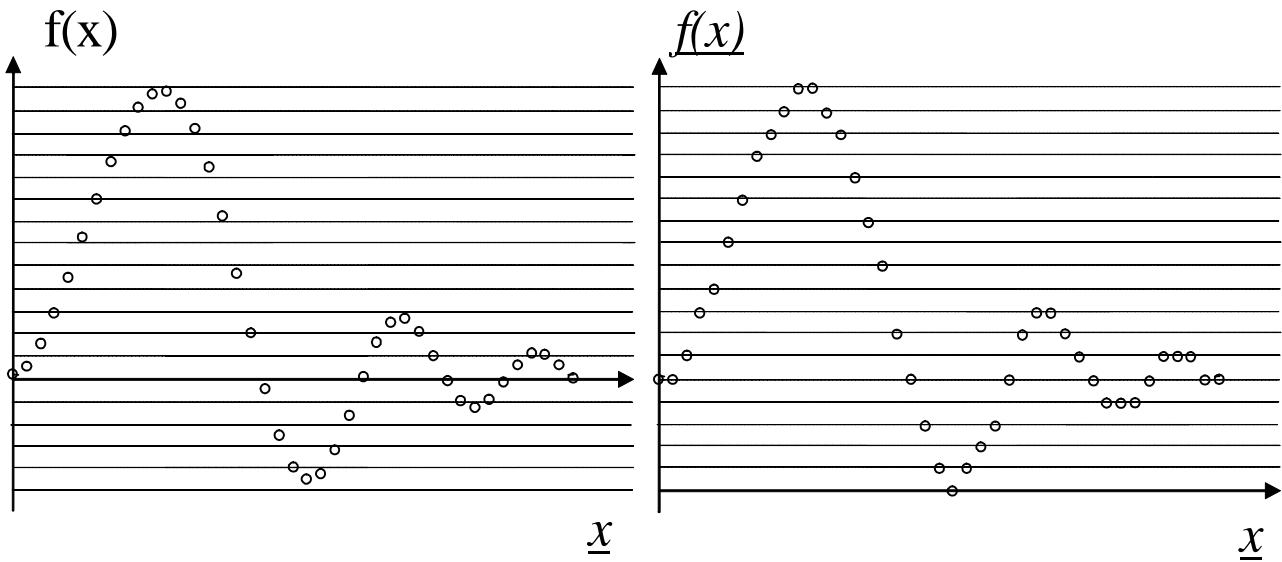


Figura 5: O Processo de Quantização

Os problemas aparecem quando tentamos reconstruir a função contínua a partir da função discretizada. Por volta de 1920, Nyquist mostrou que existe um limite para que amostragem seja

bem sucedida e não introduza erros. Baseados nesse limite, Shannon e Whittaker escreveram o teorema que diz que existe uma reconstrução exata de uma função amostrada uniformemente com uma determinada taxa de amostragem ($\frac{1}{\Delta t}$), se esta taxa for maior que o dobro da maior variação contida na função (2Ω).

Neste caso, a reconstrução ideal é dada por:

$$f(t) = \sum_{k=-\infty}^{k=+\infty} 2\Omega\Delta t f(k\Delta t) \text{sinc}(2p\Omega(t - k\Delta t))$$

Equação 2: Definição

Assim, uma técnica muito comum é aplicar uma transformação na função para que contenha apenas freqüências dentro da taxa de amostragem desejada, na prática um filtro de suavização.

2.4. O Paradigma dos Quatro Universos

Esse paradigma vai nos ajudar a entender esse processo de sair do mundo real, onde os sinais são contínuos, e ir para o mundo do computador onde tudo é discreto. O paradigma especifica 4 universos. O universo Físico, onde estão os objetos do mundo real. O universo Matemático, onde formulamos descrições abstratas desses objetos. O universo de Representação vai permitir trazer essas descrições abstratas para o mundo digital, e é onde se dará a discretização dos sinais contínuos. O universo de Implementação é onde é feita a codificação do sinal discretizado na memória do computador através de uma estrutura de dados.

Assim, para estudar um determinado fenômeno, ou objeto da natureza, no computador, nós precisamos associá-lo a um modelo matemático e então encontrar uma representação discreta para esse modelo que pode ser implementada no computador.

A codificação dos dados pode ser decodificada e depois reconstruída para podermos voltar com o fenômeno estudado para o mundo real.

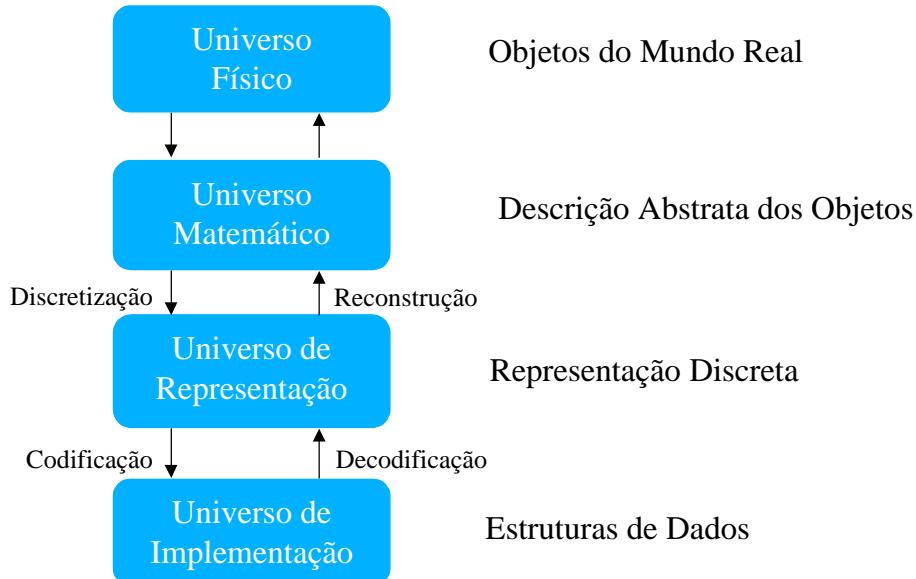


Figura 6: O Paradigma dos Quatro Universos

Para imagens também precisamos fazer esses mesmos passos.

A imagem fotográfica é obtida do mundo real através de câmeras ou sensores que captam luz. A imagem capturada em um filme fotográfico representa bem a imagem real que queremos transportar para o computador. Repare que o filme define um plano limitado por um retângulo, onde cada posição nesse plano contém a informação de cor relativa aquela posição, ou seja, a imagem neste caso é um sinal de cor 2D contínuo, onde o domínio é o plano e o contradomínio é o espaço de cor. A discretização disso é muito simples, segue os mesmos princípios da discretização de sinais 2D contínuos quaisquer.

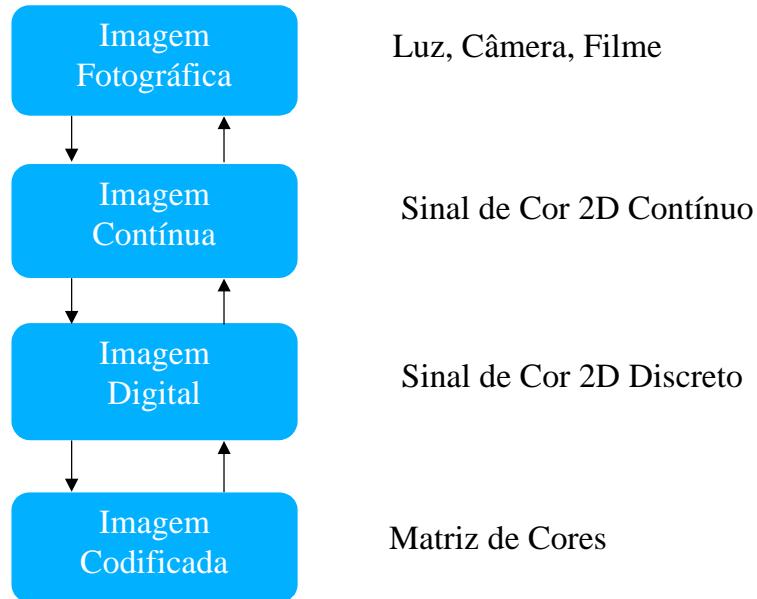


Figura 7: O Paradigma para Imagens

A imagem contínua é modelada matematicamente pela função:

$$I(x, y) = \text{Cor no ponto } (x, y)$$

$$x \in [0, X] \text{ e } y \in [0, Y]$$

Equação 3: Definição de Imagem Contínua

onde x e y são números reais, limitados ao intervalo de 0 a X, e de 0 a Y, respectivamente.

Colocando isso em um gráfico:

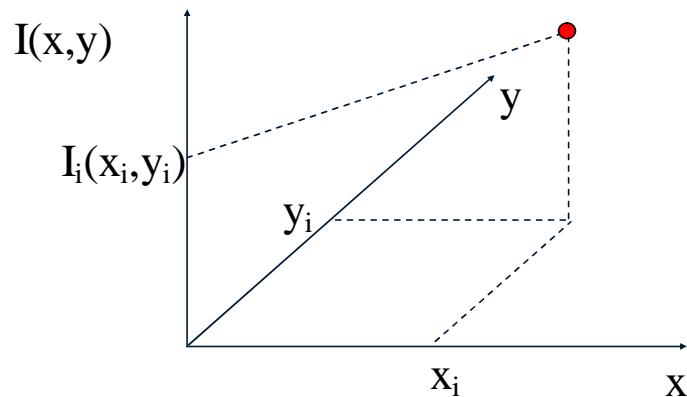


Figura 8: Gráfico de uma Função Bidimensional

A imagem digital é obtida a partir de uma amostragem e de uma quantização dessa função, como já foi visto, e pode ser representada pela mesma função, só que neste caso, x e y são números inteiros. Uma vez amostrado o plano temos uma matriz de valores que representam a cor. Cada elemento dessa matriz é chamado de Pixel (abreviação de Picture Element).

Mas, ainda não falamos nada sobre a quantização da cor. Na realidade cor também é um fenômeno físico para o qual precisamos fazer os mesmos passos que fizemos para o plano da imagem. Precisamos descobrir a definição de cor, suas representações e como trazê-las para o computador. Por enquanto, vamos deixar essas discussões de lado e atacar a questão de amostragem do plano 2D onde as cores estão distribuídas.

A figura a seguir ilustra o processo de discretização da imagem e mostra uma possível codificação da cor para cada elemento da matriz.

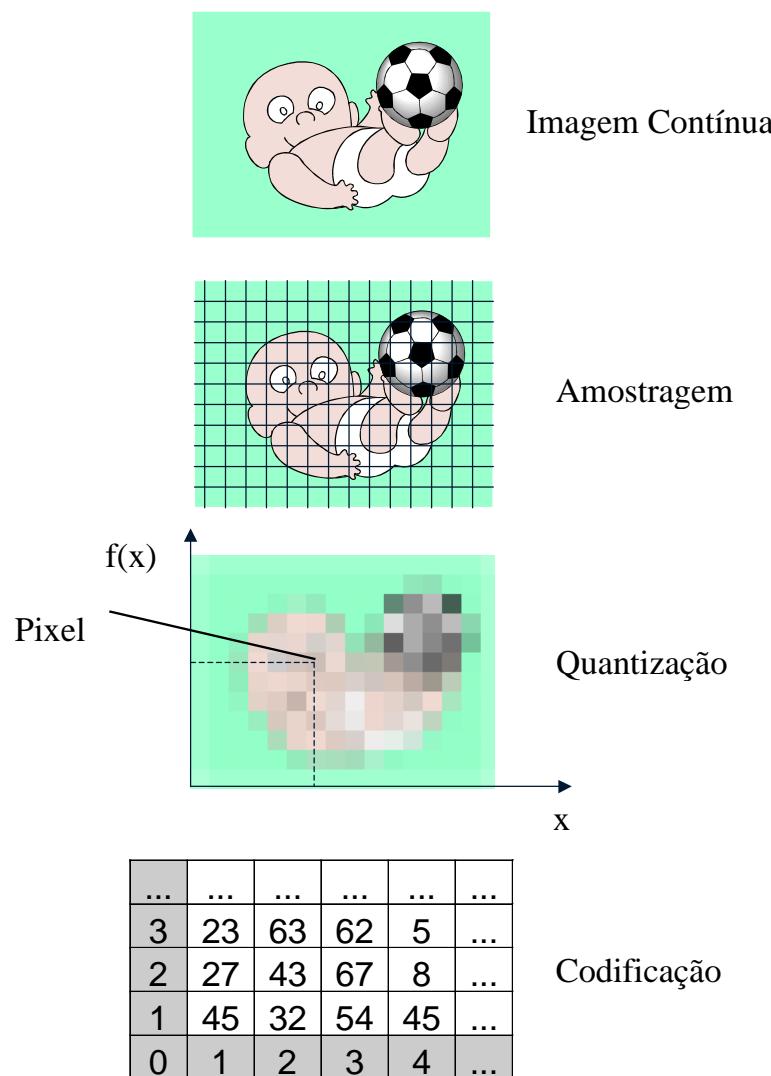


Figura 9: Processo de Discretização de uma Imagem

3. Resolução Espacial

Ao ser digitalizada a imagem assume um tamanho adimensional, 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.

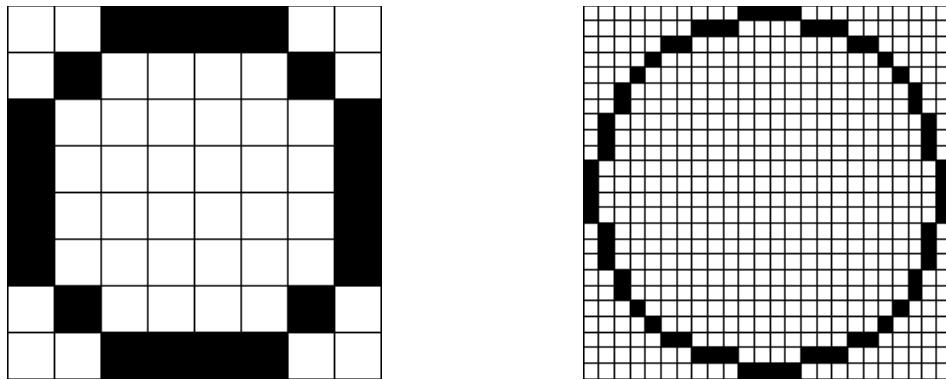


Figura 10: Imagens com Mesmo Tamanho Real e Diferentes Resoluções

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.

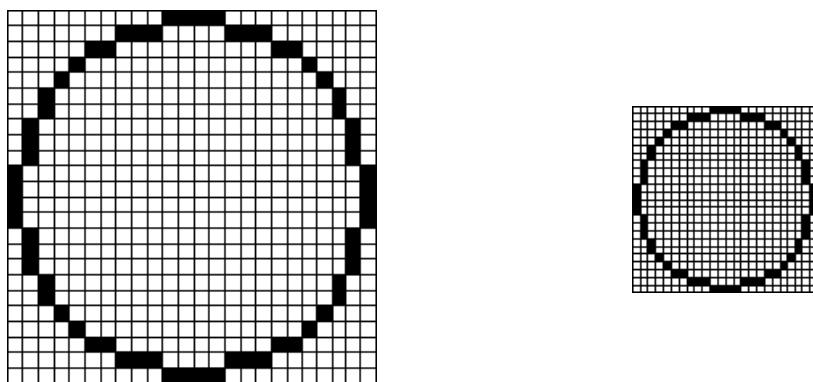


Figura 11: Imagens com Mesma Resolução e Tamanho Real Diferentes

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

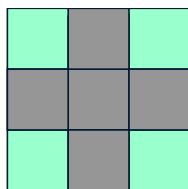
imagem com qualidade aceitável. É claro que se os objetos não estejam em um mesmo plano, as medidas são muito mais complexas e envolvem transformações projetivas.

3.1. Topologia de um Pixel

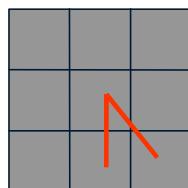
As amostras do plano são pontuais e adimensionais, mas os dispositivos para reconstrução não deixam um espaço entre uma amostra e outra. Assim, muitas vezes iremos representar o pixel, não como um ponto, mas como um quadrado preenchido com a cor do pixel, que está no seu centro. Por sinal, o exemplo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

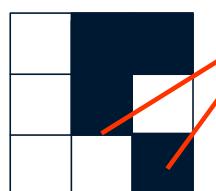
Nesta grade regular podemos definir dois tipos de vizinhança: 4 conectada (onde os vizinhos são eqüidistantes) ou 8 conectada (onde nem todos são eqüidistantes, os vizinhos na diagonal são mais distantes que os vizinhos nas laterais). Normalmente usa-se uma combinação das duas vizinhanças, definindo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



Vizinhança 4 Conectada



Vizinhança 8 Conectada



Vizinhos 8 conectados,
mas não 4 conectados

Figura 12: Vizinhança em um Grid Regular

Para criar uma vizinhança onde todos os vizinhos fossem eqüidistantes 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 converte-la para a grade hexagonal, realizar o processamento e converte-la de volta para a grade regular.

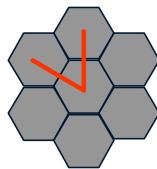


Figura 13: Vizinhança em um Grid Hexagonal

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.

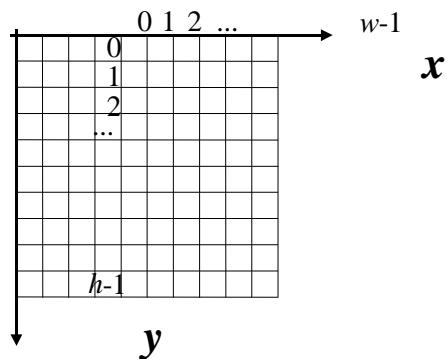


Figura 14: Sistema de Coordenadas de uma Imagem

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.

Mas pode ser orientado de várias formas:

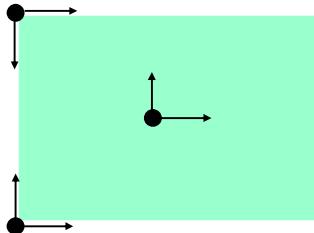


Figura 15: Eixos Mais Comuns

3.2. Reconstrução

Quando falamos em amostragem imediatamente nós temos o problema inverso: Como reconstruir? Ou melhor podemos fazer a pergunta: Dados os valores discretos, qual a função contínua que passa por eles? Quando estamos no ambiente computacional, também estamos querendo resolver um outro problema que é o de re-amostragem. Isto é, reconstruir e amostrar novamente com outra resolução ou com outra grade regular ou não.

Em matemática, reconstrução é sinônimo de interpolação. O uso de interpolação permite que dada uma seqüência de amostras possamos obter o valor da função em uma posição qualquer do eixo.

A técnica mais trivial é simplesmente escolher a amostra mais próxima da posição desejada e tomar o seu valor como valor desta posição. Esta técnica é usada muito para a visualização da imagem em maior ou menor proximidade, ou melhor, fazendo Zoom, simulando uma aproximação ou afastamento da imagem em relação ao observador.

Uma outra técnica muito utilizada para fazer reconstrução é a de interpolação linear (bilinear no caso de 2 dimensões). Essa técnica calcula uma média ponderada das amostras mais próximas da posição desejada de acordo com a distância à mesma.

Pela figura a seguir podemos perceber que a interpolação linear irá obter resultados bem melhores que a de vizinho mais próximo. Mas fica claro que podemos obter resultados ainda melhores usando curvas em vez de retas. Uma técnica com esse enfoque que se tornou popular é a interpolação bi-cúbica, que é semelhante a interpolação linear, mas em vez de usar retas, usa funções cúbicas (x^3).

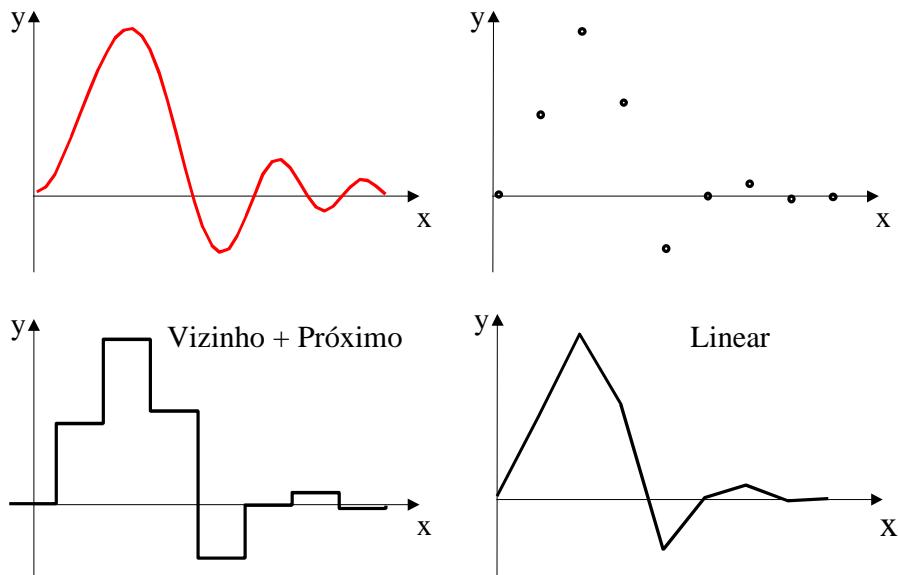


Figura 16: Interpolação de uma Função Discreta Unidimensional

Esta imagem mostra o efeito de uma mudança no tamanho de uma determinada imagem, usando-se a técnica de interpolação linear e a técnica do vizinho mais próximo.

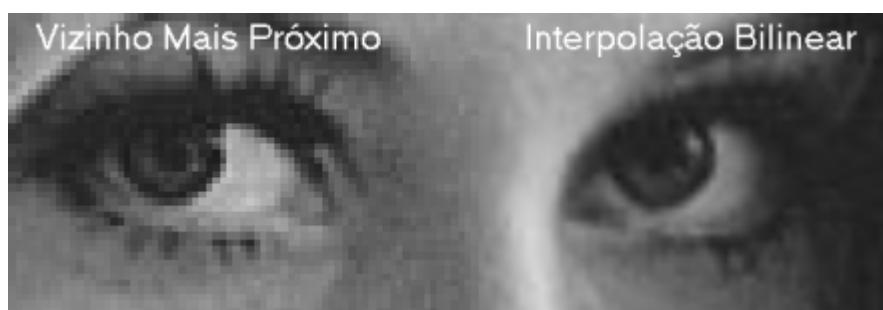


Figura 17: Interpolação Vizinho mais Próximo versus Bilinear

Todas as técnicas de deformação (*warping*) da imagem são implementados usando interpolação. Até mesmo alguns movimentos rígidos, como as rotações diferentes de 90°, utilizam interpolação, por que no fundo estamos reconstruindo, deformando o modelo contínuo e re-

amostrando. Isto também ocorre nas transformações contínuas de uma imagem em outra (*morphing*) que são feitas como uma composição de uma deformação do R^2 (*warping*) combinada com uma composição de cores (*blending*).

3.3. *Aliasing*

Um fenômeno muito comum em amostragem é o fenômeno de *aliasing*. Ele acontece quando realizamos uma amostragem muito pobre da função contínua original e quando vamos reconstruir-a obtemos resultados inconsistentes com o esperado. Matematicamente podemos dizer que a freqüência de amostragem foi aproximadamente inferior à maior freqüência de variação da função contínua.

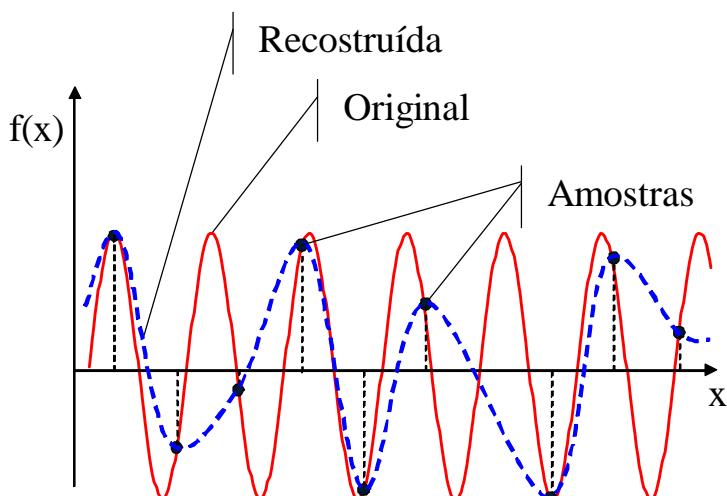
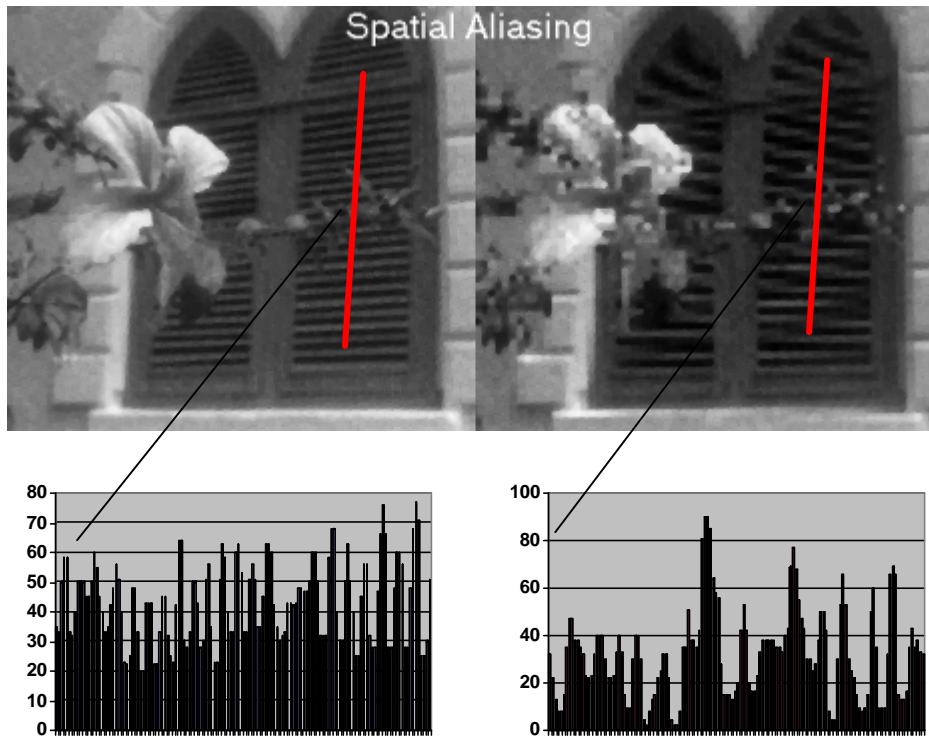


Figura 18: *Aliasing Unidimensional*

Para resolver o problema de *aliasing* temos duas soluções, aumentar a taxa de amostragem, que em geral nem sempre é possível, ou suavizar a função contínua para que ela não tenha freqüências mais altas que a nossa taxa de amostragem.

A imagem a seguir mostra o efeito do *aliasing* espacial em uma imagem que foi amostrada em alta e baixa resolução. Traçando-se uma linha sobre a imagem podemos compreender melhor o conceito de freqüência espacial e o respectivo efeito de aliasing no resultado.

Aliasing também aparece quando estamos reamostrando com menos pontos uma função já discretizada. Neste momento podemos realizar uma média ponderada dos valores que estão sendo descartados.

Figura 19: *Aliasing Bidimensional*

Podemos ver o fenômeno de *aliasing* em outras situações como ao filmarmos uma roda de um carro rodando mais rápido do que a câmera é capaz de capturar, neste caso vemos a roda girar para trás, justamente por que a taxa de amostragem temporal está muito baixa. Da mesma forma que o aliasing espacial, aliasing temporal ocorrerá sempre que o objeto se mover repetitivamente mais rápido que a taxa de amostragem temporal ou número de quadros por segundo.

Um problema muito comum, que é uma forma de *aliasing*, aparece quando sobreponemos dois padrões periódicos que interagem não linearmente produzindo um terceiro padrão. Esse novo padrão é chamado de padrão de Moiré.

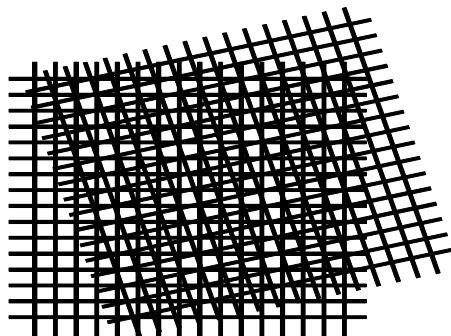


Figura 20: Padrão de Moiré

Em Computação Gráfica *aliasing* é muito comum durante o desenho de primitivas. A memória de vídeo da placa gráfica pode ser vista como uma imagem e uma primitiva a ser desenhada como um objeto real a ser amostrado. Como as primitivas possuem bordas abruptas (alta freqüência) aliasing aparece naturalmente. A solução é suavizar as bordas durante o desenho da primitiva.

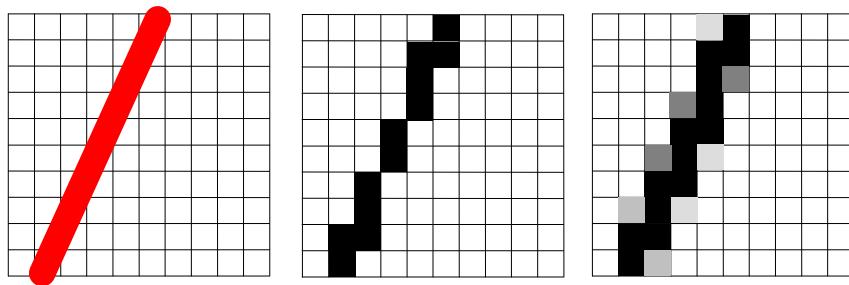


Figura 21: Anti-Aliasing em Desenho de Primitivas

Também podemos evitar *aliasing* no processamento de cenas sintéticas que faz uso do algoritmos de tracado de raios (*ray tracing*). Neste caso para cada *pixel* da imagem a ser gerada, criamos vários *sub-pixels*, ou super-amostramos a cena, para depois calcularmos uma média desses valores que será por fim atribuída ao *pixel*.

4. Cor

Quando falamos de cor de um ponto de vista físico, na realidade estamos falando de Luz. A luz percorre o espaço, ora se comportando como uma onda, ora como uma partícula, isso caracteriza a natureza dual da luz. Para estudarmos cor iremos observar o comportamento da luz como uma onda. Podemos dizer que neste caso a luz é uma radiação eletromagnética que possui uma certa “energia” para cada comprimento de onda. A função que informa o valor dessa “energia” é chamada de Distribuição Espectral de Potência.

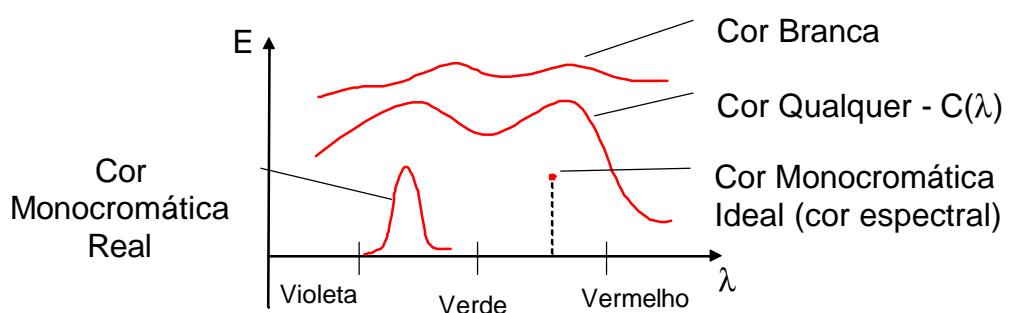


Figura 22: Distribuição Espectral de Potência

Na figura acima podemos ver um exemplo de distribuição, e a diferença entre uma cor monocromática ideal (apenas 1 comprimento de onda) e uma cor monocromática real (onde existe um comprimento de onda predominante, mas não é único).

Como a luz pode ser caracterizada pela sua distribuição espectral, ela pode então ser modelada matematicamente por uma função unidimensional que depende do comprimento de onda. Mas nós estamos interessados em cor, e neste caso estamos falando da sensação humana de diferentes espectros de luz. Ou seja, a definição de cor neste caso envolve uma característica perceptual da espécie humana. Até porque é sabido que cada espécie possui uma percepção de cor diferente. Consequentemente precisamos entender um pouco mais sobre como o olho humano percebe cor e envia essa informação para o cérebro. (A ciência que estuda cor do ponto de vista físico é chamada de Colorimetria. A ciência que estuda cor do ponto de vista perceptual é chamada de Fotometria.)

Nós percebemos luz entre aproximadamente 380nm e 780nm. Luz de 380nm até 500nm parece azul, de 500nm a 600nm parece verde, e de 600nm a 780nm parece vermelho.

É claro que o resto do espectro eletromagnético é ocupado com outros tipos de ondas, que não produzem sensações visuais. A figura a seguir mostra algumas regiões conhecidas e/ou utilizadas por nós.

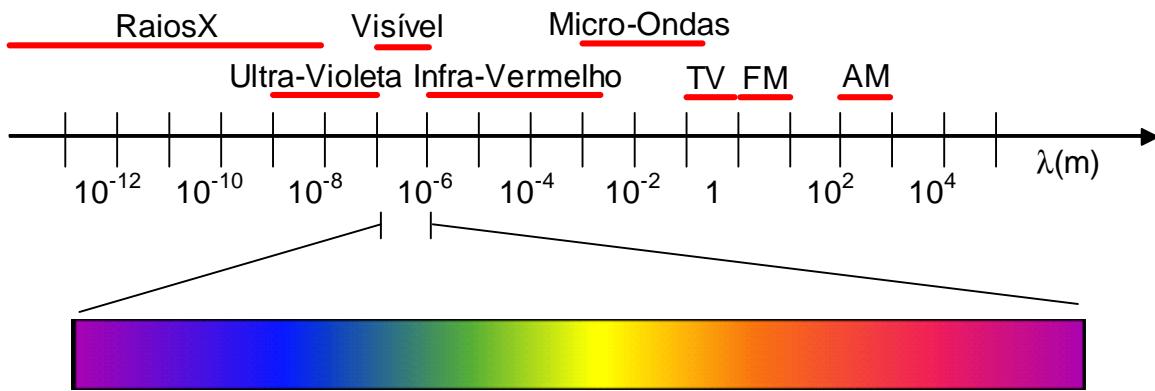


Figura 23: Comprimentos de Onda do Espectro Eletromagnético

A cor monocromática ideal é também chamada de cor espectral, já que possui apenas um único comprimento de onda. A cor púrpura não é uma cor espectral, ela contém comprimentos de onda bem próximos a 380nm e outros bem próximos a 780nm. Veremos esse fenômeno da cor púrpura novamente mais adiante. A luz branca possui todos os comprimentos de onda com energia máxima. A luz branca ideal seria um espectro com uma linha horizontal.

No nosso olho existem dois tipos de sensores, cones e bastonetes. Bastonetes são sensíveis a todos os comprimentos de onda ponderadamente e por isso não são capazes de distinguir cor, possuem melhor resposta à luz de baixa intensidade (visão noturna). Já os Cones são menos

sensíveis à luz (visão direta), mas permitem a distinção de cores porque existem três tipos de cones, os sensíveis a comprimentos de onda próximos a 450nm, a 550nm, e a 600nm ($\text{nm}=10^{-9}\text{m}$). Isso caracteriza o processo de discriminação de cor do olho, chamado de tricromacia.

A distribuição de cones e bastonetes ao longo da retina é responsável por diversos aspectos da nossa visão. Procuraremos nos concentrar apenas naqueles que nos interessam no momento.

A informação de cor é enviada para o cérebro em dois canais, um que codifica a intensidade da luz e outro que codifica as diferenças de cor. A informação provinda dos bastonetes é chamada de luminância e a informação provinda dos cones é combinada em um único canal e chamada de crominância. Isso mostra que o sinal de luminância é muito importante para distinguirmos objetos na imagem nos dando acuidade espacial. Assim como, mostra que a informação de crominância pode ser bastante compactada.

O fato de termos 3 sensores para cor, esta informação ser combinada em um única e termos um sinal separado para informação de intensidade são fatos muito utilizados na captura, representação e codificação do sinal de cor.

4.1. Processos de Formação de Cor

As cores que nós percebemos são geradas a partir de diversos processos psicoquímicos. Em geral podemos classificá-los em três grandes grupos: aditivos, subtrativos e por pigmentação.

No processo aditivo (o mais simples), vários raios de luz são combinados para formar um novo raio. No processo subtrativo um raio de luz passa por um filtro que elimina alguns comprimentos de onda resultando numa nova cor. No processo por pigmentação a luz passa por diversos eventos, sendo refletida, transmitida e absorvida, o resultado dessa interação é a nova cor.

Todos os três processos são comuns na natureza. O processo aditivo é usado nos monitores de computadores e televisões. O processo subtrativo aparece quando usamos um projetor para vermos slides em uma tela. E o processo de pigmentação nos permite ver cores em pinturas e trabalhos impressos. Em alguns processos industriais como impressão offset, mais de um processo pode ser utilizado.

Simplificando o que acontece em uma impressora podemos ver a subtração ocorrendo no papel branco, onde toda a luz é refletida, mas onde existe cor apenas a mesma será refletida e todo o resto será absorvido, ou “subtraído” da luz branca incidindo sobre o papel.

De qualquer forma temos um sistema receptivo de cor baseado em cones sensíveis vermelhos, verdes e azuis. Na escola primária aprendemos que as cores “básicas” são “vermelho”, amarelo e “azul”. Na realidade isso está relacionado com a mistura de tintas que poderia se aproximada por um processo subtrativo e esse vermelho na realidade era magenta, e o azul era ciano, cores complementares do vermelho verde e azul.

Nós iremos concentrar nossos estudos sobre o processo aditivo, pois a luz que chega ao nosso olho se combina aditivamente e todos os estudos de colorimetria também usam esse processo.

4.2. Amostragem e Reconstrução de Cor

Observando o modelo de sensores de cor do olho humano, vemos que podemos representar cor usando um número pequeno e finito de sensores, que são responsáveis por captar uma parte do espectro cada um deles. Isso é possível pelo fato de que o olho humano percebe um determinado conjunto de cores de espectro distinto como sendo a mesma cor. Esse fenômeno é chamado de metamerismo. Graças a ele podemos assistir uma transmissão por TV colorida de um evento, tendo a sensação visual de estarmos vendo a cena real.

Para podermos representar cor no computador precisamos saber como amostrar e reconstruir cor. No universo físico a amostragem de cor está associada aos receptores e a reconstrução está associada aos emissores.

Usando o mesmo modelo do olho o receptor deve ter um número finito de sensores, que poderiam ser 3, mas vamos apresentar de forma genérica usando N sensores. Matematicamente podemos modelar o receptor usando a curva de resposta espectral de cada sensor i, como $s_i(\lambda)$ e escrevendo:

$$C_i = \int C(I) s_i(I) dI$$

Equação 4: Modelo para o Receptor

onde C_i é a cor C percebida pelo sensor i.

Para que o emissor possa agora reconstruir a cor C ele deverá ter N sensores que estimulados corretamente produzirão a cor C . Matematicamente podemos modelar o emissor usando a base de primárias $P_k(\lambda)$ de cada sensor k e escrevendo:

$$C(\mathbf{I}) = \sum_{k=1}^N \mathbf{b}_k \cdot P_k(\mathbf{I})$$

Equação 5: Modelo para o Emissor

onde β_k são as componentes primárias da cor C.

O que gostaríamos de fazer agora é obter os β_k em função dos C_i . O que nos permitiria reconstruir a cor C usando o dado emissor.

Substitui-se C na equação do receptor pela fórmula do emissor para cada um dos sensores do receptor. Isso resulta em:

$$C_i = \sum_{k=1}^N \left(\mathbf{b}_k \cdot \int P_k(\mathbf{I}) s_i(\mathbf{I}) d\mathbf{I} \right)$$

Equação 6: Cálculo da componente C_i para um par Receptor-emissor.

Esta equação nos permitiria obter as componentes β_k necessárias no emissor para reconstruirmos a cor C. Só que na prática, as curvas de resposta espectral $s_i(\lambda)$ do receptor são muito complicadas de se obter, por isso precisamos de uma outra forma de reconstruir cor.

Primeiro, iremos normalizar as componentes β_k por uma cor de referência $W(\lambda)$ com objetivo de calibrar a escala de cores corretamente para cada sensor. O que resulta em:

$$W(\mathbf{I}) = \sum_{k=1}^N w_k \cdot P_k(\mathbf{I}) \quad T_k = \frac{\mathbf{b}_k}{w_k}$$

Equação 7: Definição da Cor de Referência e Componentes Normalizadas.

onde T_k são as componentes primárias normalizadas da cor C.

Agora definimos as funções de reconstrução de cor $C_k(\lambda)$ do emissor em relação ao receptor de maneira que as componentes T_k são obtidas com:

$$T_k = \int C(\mathbf{I}) C_k(\mathbf{I}) d\mathbf{I}$$

Equação 8: Obtenção das Componentes Normalizadas.

E a cor pode ser finalmente reconstruída usando:

$$C(\mathbf{I}) = \sum_{k=1}^N T_k \cdot w_k \cdot P_k(\mathbf{I})$$

Equação 9: Reconstrução de uma Cor para um par Receptor-emissor

Não vamos entrar nos detalhes matemáticos (que já foram muitos até agora) da definição das funções de reconstrução de cor, mas a grande vantagem em usá-las é que em vários casos podem ser facilmente obtidas a partir de experimentos. Podemos interpretar as funções de reconstrução como sendo as coordenadas das cores puras (espectrais) vistas pelo receptor, escritas na base de primárias de cada sensor do emissor.

Em resumo:

- Dadas as primárias do nosso emissor $P_k(\lambda)$;
 - Definimos a cor de referência $W(\lambda)$ em função das $P_k(\lambda)$, obtendo w_k .
 - Obtemos as funções de reconstrução de cor $C_k(\lambda)$ a partir de experimentos usando $W(\lambda)$;
-
- Agora, dada uma cor qualquer $C(\lambda)$, obtemos as componentes T_k a partir das $C_k(\lambda)$;
 - Para reconstruir a cor, usamos a equação 9, pois já temos todas as informações necessárias.

Quando o receptor é o olho humano essas funções são chamadas de “*Color Matching Functions*”. Mais tarde veremos um dos experimentos realizados para obtenção das funções de reconstrução de cor para o olho.

4.3. Cor no Paradigma dos 4 Universos

Uma vez que sabemos amostrar e reconstruir cor, podemos finalmente situar cor no paradigma dos 4 universos. Cor é uma onda eletromagnética que matematicamente pode ser representada por uma função unidimensional, mas que usamos um número finito, pequeno e discreto de sensores como no olho humano. Cada conjunto de sensores é chamado de modelo de representação, e cada sensor no modelo é chamado de componente.

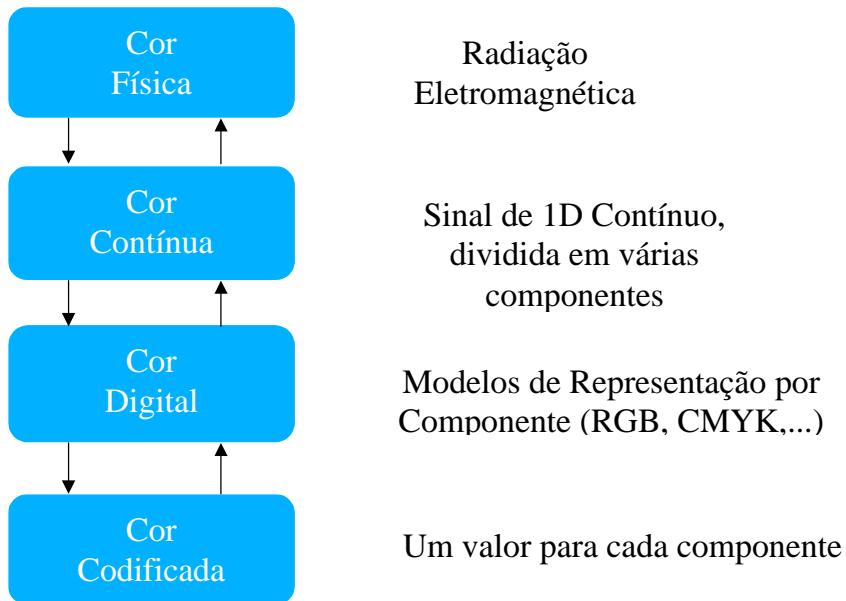


Figura 24: Cor no Paradigma dos Quatro Universos.

Quando codificamos cor é muito comum usarmos 1 byte para cada componente. Um byte permite $2^8 = 256$ possibilidades, assumindo valores de 0 a 255. Por isso esses números aparecem tanto em programas de processamento de imagens e não sabíamos de onde vinham. Mas, repare que essa é apenas uma possível representação, por ser a mais comum induz a algumas pessoas pensarem que é a única, e pior, algumas pessoas pensam que cor no computador se resume aos tais valores de 0 a 255.

Se assumirmos que são 3 componentes isso totalizaria 2^{24} bits, ou 16 milhões de possibilidades. Alguns experimentos mostram que somos capazes de reconhecer aproximadamente 400 mil cores diferentes, o que na prática resulta em um número menor que 2^{19} bits, ou seja 2^{24} está mais do que suficiente levando em conta os diversos metamericismos.

O modelo de representação de cor que nós vimos é completamente genérico e para que os programas de processamento de imagens possam conversar entre si e interagir melhor com o usuário é preciso estabelecer padrões.

4.4. Padronização para Cor

CIE-RGB

O primeiro padrão baseado nesses conceitos foi estabelecido em 1931 pelo comitê CIE (*Comission Internationale de l'Éclairage*). Foi chamado de CIE-RGB pois define as primárias

$P_r(\lambda)$, $P_g(\lambda)$ e $P_b(\lambda)$ do emissor como sendo 3 cores espectrais. O padrão também define o branco como $W(\lambda) = 1$ (chamado ponto branco D₆₅), e com isso foi realizado um experimento de cor para definir o Observador Padrão, ou seja, obter as funções de reconstrução de cor para esse sistema.

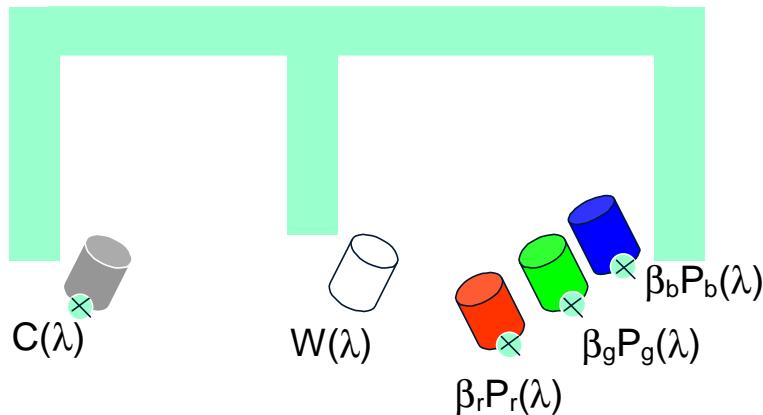


Figura 25: Experimento para Obtenção das Funções de Reconstrução de Cor

Para tanto, colocou-se 3 luzes monocromáticas baseadas nas primárias definidas, acopladas a um mecanismo que possa variar a intensidade de cada uma delas ($\beta_r P_r(\lambda)$, $\beta_g P_g(\lambda)$ e $\beta_b P_b(\lambda)$). Colocou-se então uma luz branca de referência também já especificada. Ajustando as três luzes de maneira que juntas produzam a luz branca de referência, guardou-se os valores (w_r , w_g , w_b) para fazer a normalização dos ajustes de cada uma das luzes.

Agora, colocou-se uma outra luz monocromática $C(\lambda)$ com um mecanismo que permita variarmos seu comprimento de onda. Para um determinado número de comprimentos de onda do espectro visível ajustamos a luz a cada comprimento de onda e ajustamos as três luzes de maneira que juntas fiquem perceptualmente idênticas cada uma dessas luzes. Os valores obtidos com $c_r = \beta_r / w_r$, $c_g = \beta_g / w_g$, $c_b = \beta_b / w_b$ para cada um dos comprimentos de onda formam as funções de reconstrução de cor com o receptor sendo o olho humano.

Isso foi feito para um grande número de pessoas escolhidas por sua competência na distinção de cores. Mas durante a experiência descobriram que determinadas cores não conseguiam ser ajustadas pelas três luzes. A solução foi mudar uma das luzes de lugar passando-a para o outro lado, podemos interpretar isso como uma subtração ou melhor um coeficiente β negativo.

Esse conjunto de resultados foi chamado de “Color Matching Functions” ou Funções de Ajuste de Cor. A partir desse conjunto foram definidas 3 funções para o padrão CIE-RGB na recomendação ITU-R BT.709.

CIE-XYZ

O fato de existirem valores negativos no padrão tem uma série de desvantagens, além disso várias outras desvantagens desse sistema fez com que o comitê criasse um outro sistema, baseado nos experimentos do CIE-RGB, mas com algumas características interessantes: todas as componentes das cores visíveis não devem ser negativas, duas primárias devem ter luminância zero (ou seja uma das componentes será a própria luminância), e um maior número de cores espectrais deve possuir pelo menos uma componente zero. A solução foi buscar uma base de primárias fora do visível, ou seja o sistema também representa cores que não são visíveis, mas tudo bem pois basta não considerá-las.

Fazendo uma transformação das funções de reconstrução de cor do sistema CIE-RGB para o CIE-XYZ temos o gráfico da figura a seguir. Nela vemos também um segundo experimento realizado em 1964 que variou um pouco apenas em relação ao primeiro experimento. Também podemos perceber que a curva de luminância (Y) é muito intensa no verde, mostrando que percebemos mais variações de verde do que de outra cor. O “calombo” na curva da componente X vem das componentes negativas que haviam no padrão CIE-RGB. E por fim reparem que a curva do Z é muito estreita, ou seja podemos usar menos memória de computador para guardar essa componente (isso será usado mais tarde quando falarmos de alguns algoritmos de quantização de cor).

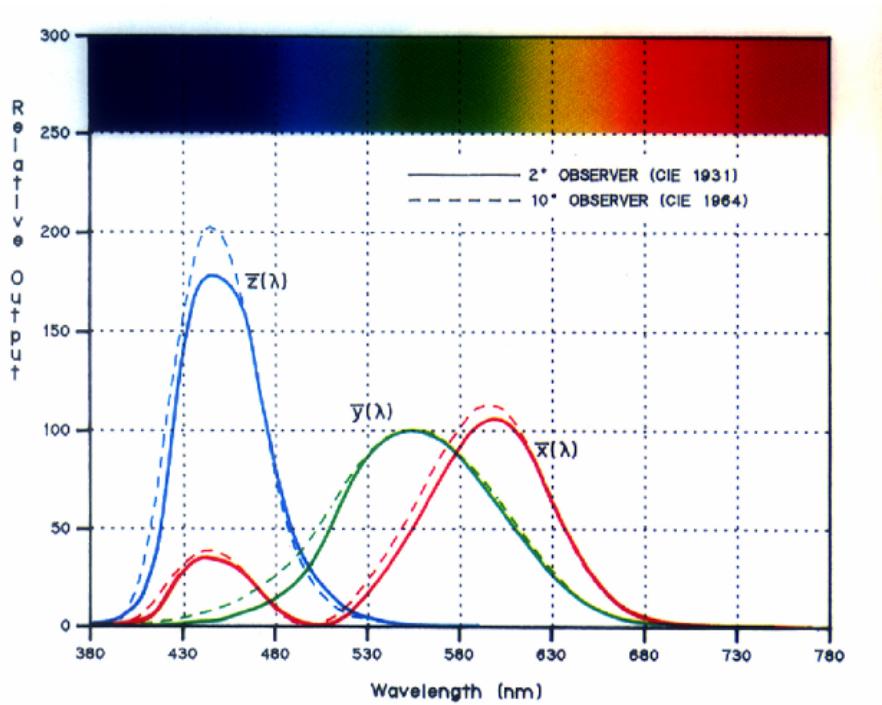


Figura 26: Funções de Reconstrução de Cor no Sistema CIE-XYZ

A conversão entre CIE-RGB para CIE-XYZ é definida como:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \bullet \begin{bmatrix} R_{709} \\ G_{709} \\ B_{709} \end{bmatrix}$$

$$\begin{bmatrix} R_{709} \\ G_{709} \\ B_{709} \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} \bullet \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Equação 10: Conversão entre CIE-RGB e CIE-XYZ

4.5. Diagrama de Cromaticidade e Propriedades de Cor

Já que luminância só é referente a intensidade da cor, criou-se uma visualização das cores visíveis que envolve somente as componentes X e Z de alguma forma. Na realidade é como se pegássemos todas as cores e projetássemos em um plano de luminância constante Y (Observação:

de fato o diagrama é mais fácil de ser entendido por estarmos usando o sistema CIE-XYZ, mas ele foi originalmente definido para o sistema CIE-RGB).

O resultado é a figura a seguir chamada de Diagrama de Cromaticidade. Nele podemos ver uma série de propriedades interessantes de cor.

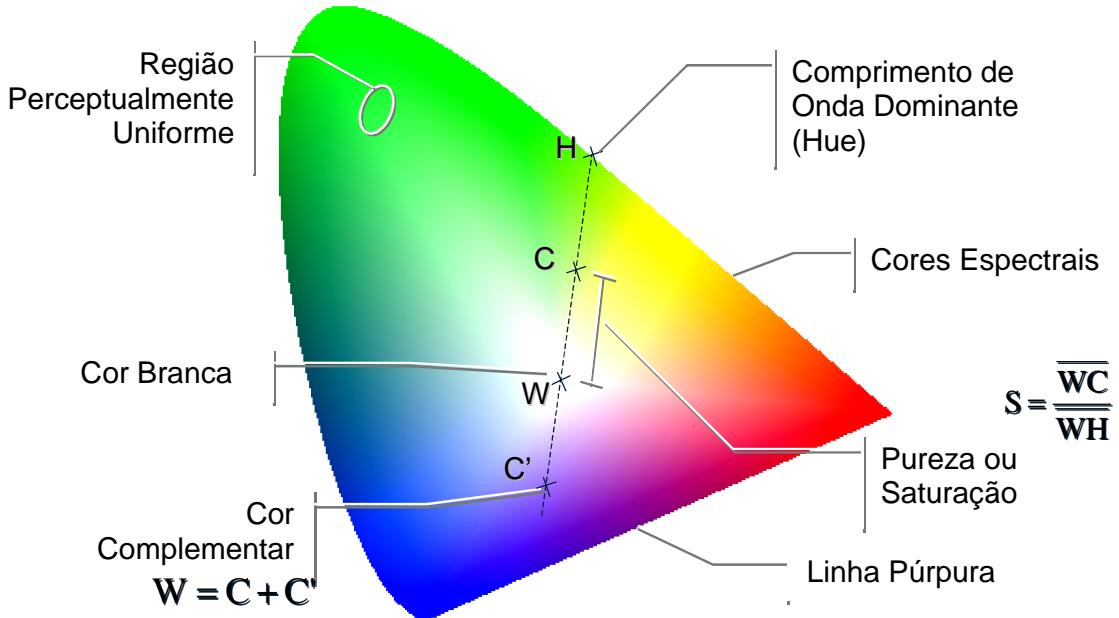


Figura 27: Detalhes do Diagrama de Cromaticidade

As cores na borda do diagrama são todas as cores espectrais, cujas componentes são exatamente as funções de reconstrução de cor.

No centro temos o branco W a partir do qual temos algumas propriedades. Dada uma cor C, a cor pura, ou o comprimento de onda dominante de C, ou melhor o matiz H de C é dado pela projeção de W, passando por C, na borda do diagrama. A saturação é simplesmente a razão entre as distâncias de W à C e de W a H, ou melhor o quanto uma cor se aproxima da cor pura ou se afasta da ausência de cor (se aproxima do branco). A cor C' que somada a C resulta na cor branca é chamada de cor complementar de C.

No diagrama os dois extremos do espectro visível são conectados por uma linha artificial (que não são cores espectrais) chamada de linha púrpura. As cores nessa linha não existem na natureza, pois seriam uma interpolação entre os dois extremos do espectro visível.

No começo dos anos 40 MacAdam realizou uma série de experiências e descobriu que as pequenas variações na cor que não são detectadas pelo olho humano formam elipses no diagrama de cromaticidade. Isso ajudou mais tarde ao CIE a criar sistemas de cor perceptualmente uniformes.

Podemos observar alguns fenômenos diretamente no espectro. Já vimos que uma cor monocromática possui uma forma estreita. Se transladarmos esta forma de lugar estaremos modificando o comprimento de onda dominante ou o matiz. Se aumentarmos sua intensidade estaremos aumentando seu brilho. E se alargarmos sua forma estaremos diminuindo sua saturação (se aproximando do branco, ex: tons pasteis).

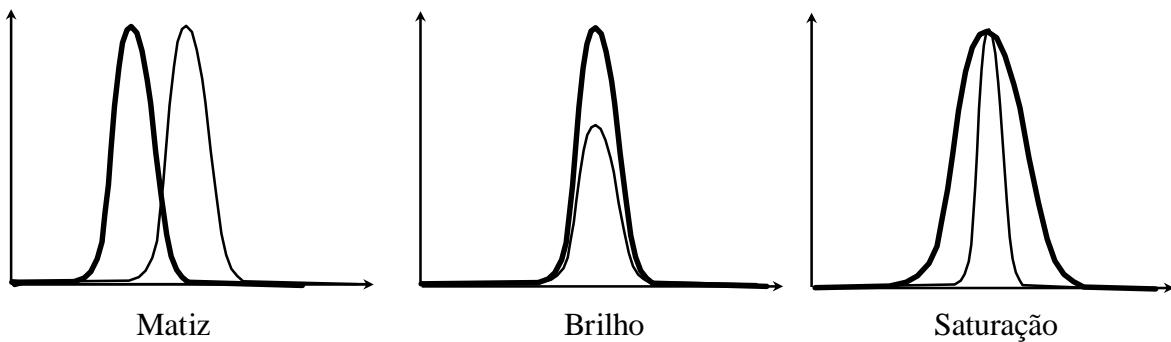


Figura 28: Mudanças na Distribuição Espectral de uma Cor

Adaptação e Correção Gamma

Um outro aspecto interessante sobre o brilho é que embora este varie de acordo com as condições de iluminação, o olho possui uma capacidade enorme de se adaptar a essas variações. É como se ele mudasse a escala de percepção para poder captar o máximo de detalhes com a luminosidade disponível. A tabela abaixo mostra valores típicos de iluminamento em LUX que variam enormemente e mesmo assim conseguimos “enxergar”.

| Ambiente | Valores (lux) |
|---------------------------|---------------|
| Luz do dia (máximo) | 100.000 |
| Luz de dia nublado | 10.000 |
| Interior próximo a janela | 1.000 |
| Mínimo p/ trabalho | 100 |
| Lua cheia | 0,2 |
| Luz das estrelas | 0,0003 |

Tabela 1: Valores de Iluminamento em Lux para Determinados Ambientes

De um outro ponto de vista podemos dizer que o olho está se “adaptando ao branco”, ou seja a percepção de branco é dependente das condições de visualização.

O olho também possui uma resposta não linear à intensidade. Isso pode fazer com que a escala de tons percebida pelo olho seja não linear em determinados casos, ou seja a gama de tons percebida não corresponde a variações equivalentes de intensidade. Para compensar esse problema realiza-se a chamada Correção Gamma, que nada mais faz do que aplicar a função inversa da resposta do olho para obter uma escala percebida linearmente. Podemos ver as duas situações na figura a seguir.

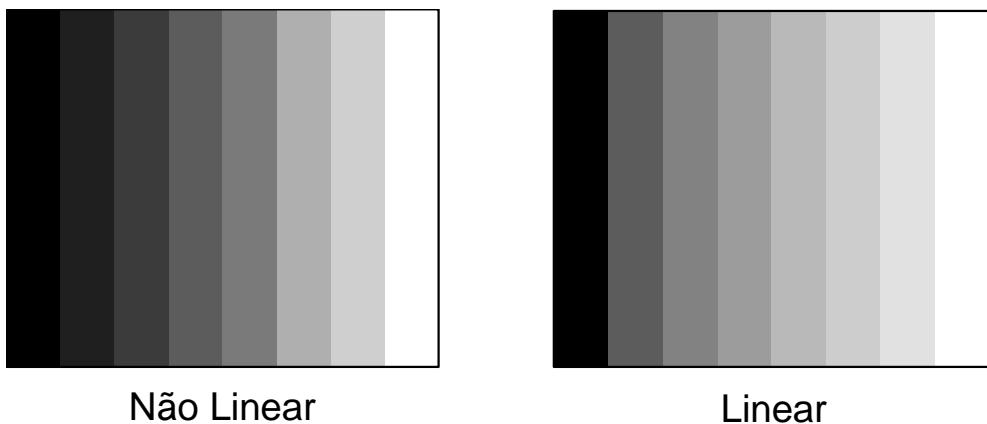


Figura 29: Correção Gamma

Em monitores analógicos mais comuns a função de resposta é praticamente idêntica ao inverso da sensibilidade de *lightness* do nosso sistema de visão. Quando codificamos cores de uma forma perceptual em uma imagem, ao mostrá-la em um monitor CRT as cores são automaticamente invertidas.

Ou seja, com a correção gamma podemos trabalhar em um espaço de cor linearmente perceptual (para intensidades) otimizando o uso do restrito número de bits disponíveis para cada componente. Se isso não fosse possível teríamos que utilizar mais de 8 bits por componente para poder representar uma variação linear de intensidades (aprox. 14 bits por componente).

Luminância x *Lightness*

Luminância é padronizada pelo CIE como intensidade física da luz na região do espectro visível ponderado por uma curva de sensibilidade espectral que corresponde à sensação de brilho do sistema de visão humano. A recomendação ITU-R BT.709 define que Luminância pode ser

computada a partir das primárias RGB para sistemas de vídeo e computação gráfica modernos como:

$$Y_{709} = 0.2125 \cdot R + 0.7154 \cdot G + 0,0721 \cdot B$$

Equação 11: Cálculo de Luminância a partir das Primárias RGB

Não devemos confundir essa definição com a definição de Luma, que se refere a uma quantidade não linear usada para representar *Lightness* em um sistema de vídeo. Neste caso o cálculo se faz aplicando-se uma correção gamma aos coeficientes RGB e de acordo com a recomendação ITU-R BT.601-4 define-se:

$$Y'_{601} = 0.299 \cdot R' + 0.587 \cdot G' + 0,114 \cdot B'$$

Equação 12: Cálculo de Luma a partir das Componentes não lineares R'G'B'

Portanto, cuidado na conversão de imagens coloridas para tons de cinza. O correto seria o programa que realiza a conversão especificar o que exatamente está fazendo. Ou permitir que você escolha qual a conversão.

Observação: sistemas de edição de vídeo analógicos são chamados de lineares, em contrapartida de sistema de edição de vídeo digital que são chamados de não-lineares. Essa linearidade não está relacionada com o sistema de cor e sim com o fato de que em sistemas analógicos as fitas são acessadas linearmente e em sistemas digitais os arquivos de vídeo são acessados em qualquer ordem.

4.6. Sistemas de Cor

Nós vimos até agora 2 sistemas de cor CIE-RGB e CIE-XYZ, mas existem outros sistemas padronizados ou não padronizados.

Sistemas Perceptualmente Uniformes

A diferença entre duas cores no sistema CIE-RGB pode ser calculada usando-se a definição de diferença euclidiana:

$$d = \sqrt{(r_1 - r_2)^2 + (g_1 - g_2)^2 + (b_1 - b_2)^2}$$

Equação 13: Diferença Euclidiana para duas Cores no Sistema RGB

Mas duas cores que são metaméricas podem possuir uma diferença numérica razoável entre si. A idéia é possuir um sistema que usando a mesma equação acima identifique cores metaméricas e não as distingua numericamente. Poderíamos então, deformar o sistema XYZ para que as elipses sobre o diagrama de cromaticidade, mencionadas anteriormente, se tornem círculos.

Assim, foram criados dois sistemas padrão em 1976, o CIE Luv, que estabelece uniformidade no diagrama de cromaticidade, e o CIE Lab que estabelece uniformidade em um espaço de cor tridimensional. L em ambos os casos está relacionado com *Lightness*.

Sistemas de Vídeo Componente

São sistemas baseados em uma decomposição das componentes RGB em luminância e crominância, onde as componentes de crominância são obtidas subtraindo-se a componente Y das componentes RGB e eliminando-se a componente G pois, como vimos, ela é a que mais colabora na luminância. Assim as componentes ficam Y, R-Y e B-Y.

Os sistemas YP_bP_r (Sony Betacam) e YC_bC_r (Vídeo Digital) são baseados nessa decomposição.

Sistemas de Vídeo Composto (NTSC, PAL, ...)

São baseados na mesma decomposição Y, R-Y e B-Y dos sistemas de cor de vídeo componente, mas para serem transmitidos pelas redes de televisão todas as componentes desses sistemas precisam ser combinadas em um único sinal, por isso são chamados de sistemas de vídeo composto.

Transmissões de vídeo usam os sistemas YUV ou YIQ. Nesses sistemas o sinal de luminância compõe 95% do sinal e a informação de crominância apenas 5%. Isso é possível devido a menor sensibilidade do olho a variações de crominância. Além disso, as televisões preto e branco foram criadas antes das televisões coloridas, o uso dessa codificação permitiu que as televisões preto e branco continuassem a receber o mesmo sinal de televisão mesmo sendo incluída a informação de crominância, que nesse caso é ignorada, pois o sinal não aumentou de tamanho. Esse é um dos grandes milagres da indústria de televisão.

Dentro das siglas de sinais de televisão a letra em destaque (M ou G, etc...) informa o padrão de luminância, a abreviação inicial indica o padrão de cor.

| Sistema | Países (dentre outros) |
|-----------------|--|
| NTSC (-M) | EUA, Canadá, Japão, México, Chile, Peru, Colombia, Coréia, ... |
| PAL (-B e -G) | Australia, China, Inglaterra, Holanda, Portugal, Espanha, Suiça, Dinamarca, Suécia, Noruega, ... |
| PAL-M | Brasil |
| PAL-N | Argentina, Paraguai, Uruguai |
| SECAM (-D e -L) | Bulgária, França, Russia, ... |

Tabela 2: Sistema de Vídeo em vários Países

Porque no Brasil a rede elétrica é de 60Hz (30 quadros por segundo) e o padrão de luminância é o mesmo, o sistema americano e o brasileiro só diferem essencialmente na cor, e é possível reproduzir uma fita de vídeo gravada num sistema em um aparelho de vídeo cassete de outro, e vice-versa, ficando apenas a cor prejudicada. Assim como, fitas européias não conseguem ser reproduzidas aqui no Brasil pois diferem na codificação de luminância e pela rede eletrica ser 50Hz.

Sistemas de Dispositivos

São os sistemas usados pelos monitores (mRGB) e scanners (sRGB), cada dispositivo possui o seu próprio sistema. Para permitir a calibração adequada o fabricante deve fornecer os valores das primárias dos fósforos do monitor, por exemplo.

Quando dizemos que uma cor possui componentes RGB=(255, 0, 36), isso não está amarrado a nenhum referencial de cor e perceptualmente pode estimular completamente diferente o olho mudando-se de dispositivo. Para que esses valores tenham significado temos que associá-los a algum sistema de cor padrão, em geral o CIE-XYZ. Por isso a necessidade de calibrarmos o monitor para aplicações que exigem alta qualidade em edição eletrônica.

Esses sistemas definem exatamente um cubo, visto a seguir:

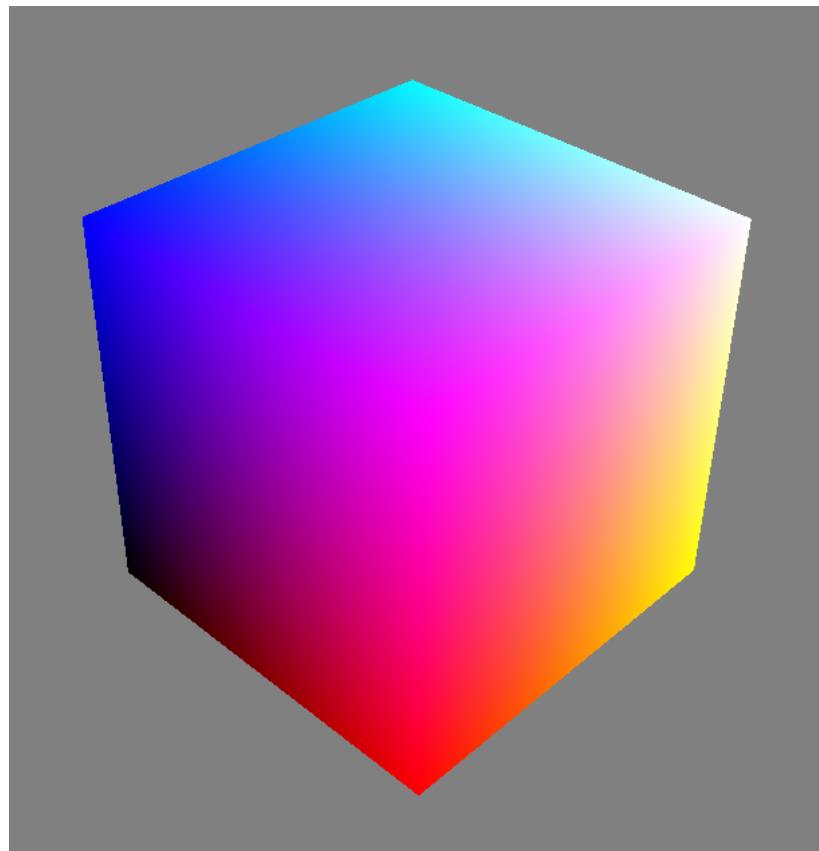


Figura 30: Cubo RGB

Por isso, a projeção deles no diagrama de cromaticidade é um triângulo, veja em Sistemas de Gerência de Cor.

Sistemas Complementares

São sistemas usados para os casos que envolvem os processos de formação de cor de forma subtrativa. Por isso são chamados também de Subtrativos, isso gera uma certa confusão pois na realidade imitam o processo de formação de cor subtrativo de uma forma aditiva.

A combinação de RGB em quantidades iguais leva ao branco. Se mudarmos os eixos de referência de lugar rotacionando-os, temos o sistema CMY. Combinando CMY em quantidades iguais leva ao preto. Por isso simula o processo subtrativo.

No caso específico de impressoras combinar CMY para obter preto tem uma série de desvantagens, pois precisamos das três cores ao mesmo tempo sobre o papel, gastando muita tinta e mesmo assim não conseguindo um preto adequado, além do que torna o processo de secagem da tinta mais demorado, enrugando o papel. Como tinta preta é muito barata, adiciona-se uma quarta componente que é responsável somente pelo preto K (black). O cálculo dessa componente é feito

diretamente a partir das componentes CMY, que posteriormente são alteradas para não conterem informação de preto. Esse cálculo em geral é muito complexo, e é necessário para se obter resultados de alta qualidade gráfica.

Sistemas de Interface (por Coordenada)

São usados para especificar cores em programas. Selecionar valores no sistema RGB é muito pouco intuitivo, embora seja como o olho perceba cor. Utilizando as propriedades de Matiz, Saturação e Brilho (ou alguma grandeza relacionada com a luminância) é muito mais fácil para o usuário especificar um cor e depois obter o valor RGB.

Para isso criou-se uma abstração para o cubo RGB deformando-o em um cone com duas pontas. Essa deformação é extremamente não linear, mas é uma forma simples de representar juntas as três propriedades.

O cone é construído tomando-se o eixo de brilho como o eixo maior do cone, e perpendicular a esse eixo define-se o plano de crominância. Neste plano os matizes possíveis são colocados em um círculo, a linha púrpura não é representada. Quanto mais próximo ao eixo de brilho menor a saturação, mais afastado maior a saturação. No fundo é uma representação simplificada do que está no diagrama de cromaticidade.

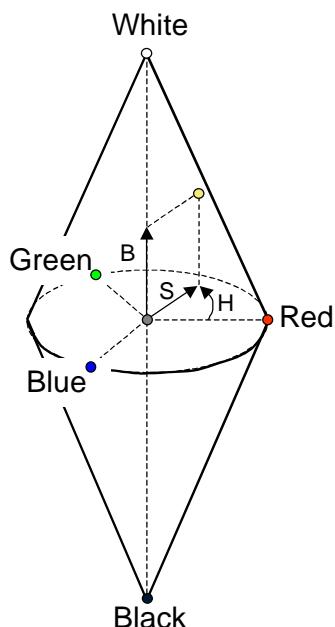


Figura 31: Modelo HSB

Como antigamente precisava-se que a conversão HSB-RGB fosse feita muito rápido, criaram-se aproximações do sistema HSB, onde o cone é substituído por um hexágono (HSL) e pode também possuir apenas uma ponta (HSV).

A tabela abaixo mostra dois cortes no sistema HLS, repare como a intensidade da cor varia para um mesmo valor de L (L é referente à Lightness, mas não tem relação direta com os sistemas CIE-Lab ou CIE-Luv). Pode-se utilizar um desses cortes, e mais um parâmetro escalar para criar a interface com o usuário para selecionar cor. No primeiro caso o usuário escolhe o valor de L e depois escolhe os valores de H e S diretamente no plano de L constante. No segundo caso o usuário escolhe o valor de H e depois os valores de L e S no plano de H constante.

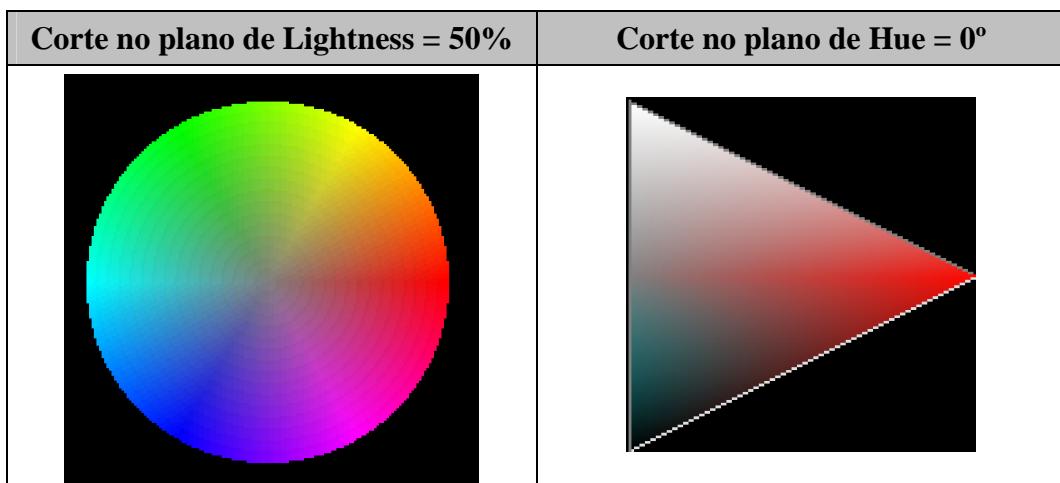


Tabela 3: Cortes no modelo HLS

Sistemas de Interface (por Amostra)

São sistemas usados manualmente por usuário para especificar cor, só que são baseados em conjuntos finitos de amostras de cor, em uma espécie de catálogo.

O primeiro sistema criado foi o de Munsell em 1905, que podemos ver na figura a seguir. Mas existem vários outros: Ostwald, Pantone, Truematch, Focoltone, ...



Figura 32: Modelo de Munsell

4.7. Sistemas de Gerência de Cor

Se nós representássemos todas as cores possíveis em um monitor de computador veríamos um triângulo no diagrama de cromaticidade. Se pegássemos um outro monitor do mesmo fabricante veríamos um outro triângulo numa posição ligeiramente diferente do anterior. Se fosse de um outro fabricante a discrepância pode ser ainda maior. No caso de uma impressora e um monitor a diferença é enorme, já que a região da impressora no diagrama possui até um formato diferente.

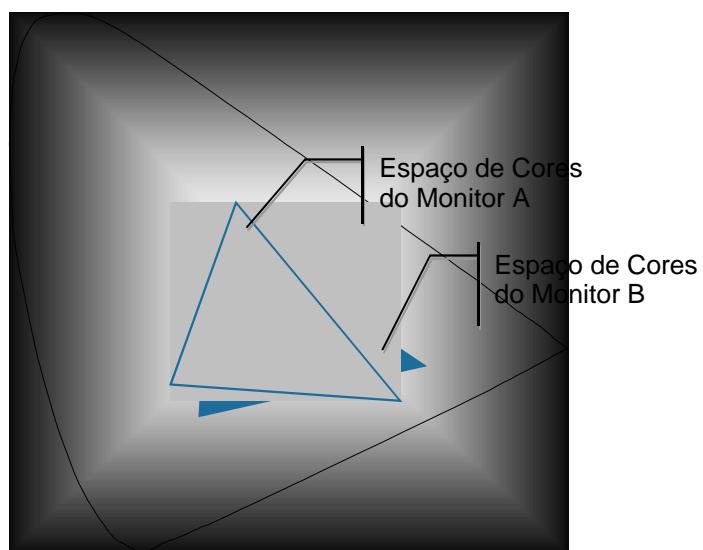


Figura 33: Espaço de Cor de Dois Monitores

Assim vemos que é necessário um mecanismo que nos permita controlar a mudança de um dispositivo para outro, de forma que possamos representar melhor as cores de interesse em cada um desses dispositivos.

Já que existem diversos padrões de cor, ou melhor diversas representações da mesma cor, nada mais natural do que existir um meio de converter de um sistema para outro. Os programas responsáveis por isso são chamados de Sistemas de Gerência de Cor (CMS - *Color Management Systems*) e realizam a conversão entre os diversos sistemas procurando o mínimo de perdas, ou seja o maior número de correspondências possíveis.

Esses programas estão preocupados justamente com a questão de portabilidade de cor, pois quando estamos mudando de um sistema para outro em geral é porque estamos mudando de um dispositivo para outro.

Aqui está um dos grandes pontos da questão de manipulação de cor no computador. Existem diversos desses programas no mercado, mas são caros e precisam de pessoal especializado, além de equipamento adequado para medição de cor. Para que o par Captura-Visualização funcione, todos os dispositivos no seu ambiente de trabalho precisam estar calibrados.

Quando estamos no ambiente de edição eletrônica esse problema fica óbvio. Se o logotipo de uma determinada marca é um tal vermelho. Esse vermelho tem que parecer o mesmo em qualquer mídia que o logo seja impresso ou visualizado.

Poucos sistemas operacionais se preocupam diretamente com isso. O pioneiro na área é o MacOS (roda em computadores Macintosh), que desde sua criação já se preocupava com isso. Recentemente, um esforço conjunto de várias empresas levou a criação do padrão ICM (Image Color Matching) para suporte a gerência de cor em aplicações. Somente a partir do Windows 95 a Microsoft passou a incorporar esse conceito.

O princípio de funcionamento de um sistema de gerência é a criação de perfis de dispositivo que a partir de informações do fabricante e do processo de calibração consigamos obter resultados ótimos, independente do dispositivo usando a informação que está no perfil para realizar a conversão para aquele sistema. De fato você precisa de dois perfis, um para o dispositivo e um para a imagem, que pode ser o perfil do dispositivo de onde a imagem foi capturada.

A criação de um perfil envolve especificar o ponto de branco, as primárias e as funções de transferência para correção gamma.

4.8. Quantização com Menos de 24 bits

Como vimos bastam 24 bits por pixel (bpp) para visualizarmos uma imagem no computador com ótima qualidade. Mas e se o nosso computador não tiver 24 bpp (16 milhões de cores)? A grande maioria das impressoras imprime somente com 4 (quatro) cores... Uma outra motivação para reduzir o número de bpp é simplesmente para economizar memória, como se fosse uma técnica de compressão de dados com perdas.

Mesmo quando existe um número reduzido de cores disponíveis para visualização na tela do computador, a representação de cada cor ainda usa 8 bits por componente. Para reduzir o tamanho de cada pixel, substitui-se sua cor por um índice para uma tabela de cores de 8 bits (veja no tópico de Armazenamento).

Para essa redução de tamanho ser efetiva o número de cores da imagem quantizada deve ser menor que 8 bpp, ou no máximo 256 cores diferentes. Portanto, precisamos encontrar dentre as cores existentes na imagem quais são as 256 mais significativas e dada uma cor da imagem qual das 256 vou escolher para substituí-la.

Existem duas técnicas muito conhecidas. A quantização Uniforme é a mais simples, e usa uma tabela de cores fixa que subdivide o cubo RGB, em 256 partes igualmente espaçadas. Dado que uma cor está em uma dessas partes é trivial encontrar a cor equivalente na tabela. Esse algoritmo é excepcionalmente rápido, mas não possui resultados muito bons quando a imagem possui muitas cores. A quantização Adaptativa é mais complexa e usa uma tabela de cor adaptada as cores presentes na imagem. A técnica mais comum é a do corte mediano (Median Cut) que procura dividir o cubo de maneira que, onde existem mais cores existirão mais divisões, ou melhor procura fazer com que cada divisão tenha aproximadamente o mesmo número de cores. Esse algoritmo é um pouco mais lento e produz resultados muito bons.

A imagem a seguir mostra as duas técnicas de quantização, usando uma palheta de cores uniforme e usando uma palheta adaptativa. Além da imagem original e de uma imagem quantizada com palheta uniforme, mas dessa vez usando também a técnica de dithering vista no próximo tópico.

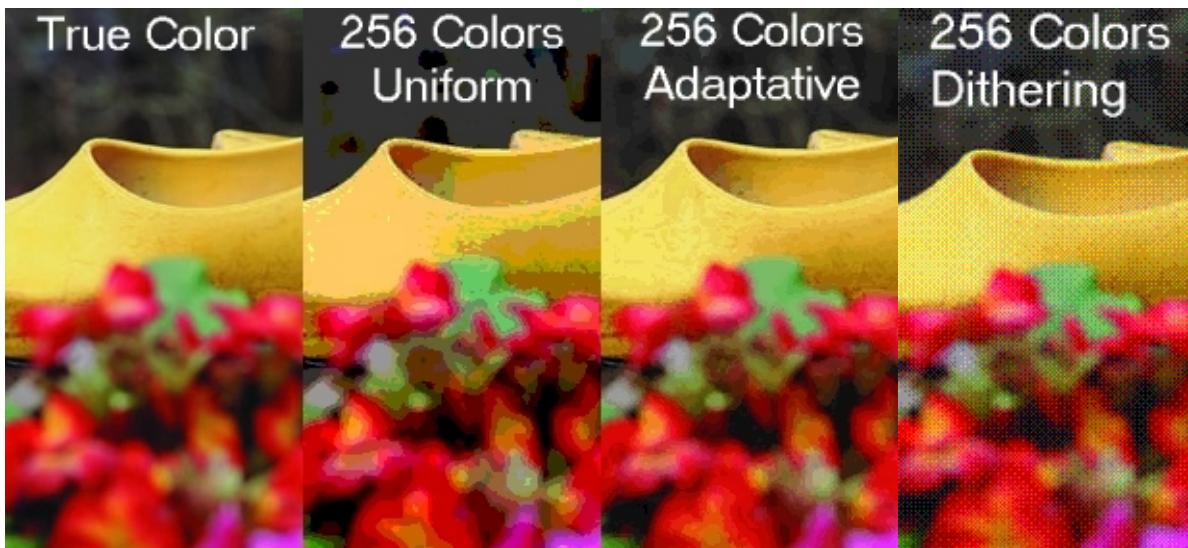


Figura 34: Exemplos de Quantização com 256 Cores

Dithering

No caso de impressoras de 4 cores cada componente CMYK de 8 bits é quantizada para 1 bit (2 cores, onde há tinta ou onde há o fundo branco do papel). Pode se mostrar que nesse caso reduz-se o problema de quatro componentes para imprimir uma componente de cada vez.

A idéia é fazer com que, embora existam apenas 2 cores, o olho perceba meio-tonos. Técnicas de meio-tonos são usados na indústria desde o fim do século passado.

No computador a técnica mais comum, utilizada para esse propósito, é a de *dithering*, cujo objetivo é minimizar perceptualmente os efeitos gerados pela quantização de contornos. Embora, essa técnica possa ser usada em várias situações de quantização (como vimos no exemplo anterior), ela é muito importante no processo de quantização para 1 bit. Inclusive alguns métodos de *dithering* são baseados em técnicas de meio-tonos da indústria, e por isso *dithering* também é conhecido com meio-ton digital.

O funcionamento de um algoritmo de *dithering* consiste em usar resolução espacial para compensar a perda de resolução de cor. De forma geral, calculamos o erro de cor gerado pela quantização de um determinado pixel, e quando existe uma mudança abrupta entre dois níveis de

cores quantizadas, substitui-se essa mudança por uma mistura dos dois níveis, com se estivéssemos espalhando o erro de quantização em uma vizinhança do pixel quantizado.

Existem muitas técnicas de dithering implementadas, tais como Bayer, Floyd-Steinberg, difusão de erro, etc. É muito comum em impressoras o uso de uma matriz de 6x6.

Como exemplo podemos utilizar as matrizes definidas abaixo e substituir os tons de cinza de quatro pixels de uma imagem por uma das matrizes provendo 5 “tons” diferentes. De acordo com a distribuição de tons originais na matriz, podemos rotacionar as matrizes de dithering para melhor uma melhor percepção do resultado.

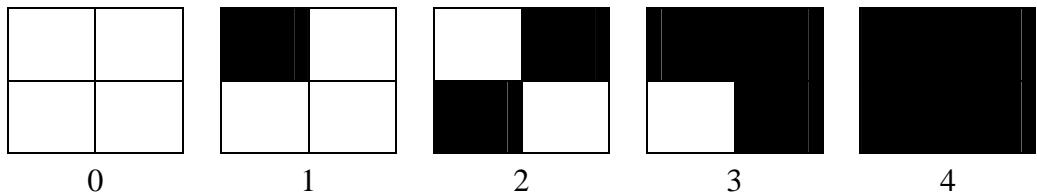


Tabela 4: Matrizes de Dithering para 5 Tons.

5. Resolução Temporal

Uma extensão natural para imagens é pensar que podemos ter uma seqüência de imagens representando uma variação de uma cena ao longo do tempo. De fato o nosso sistema de visão envia constantemente para o nosso cérebro a informação visual captada pelo olho. Portanto o nosso sistema de visão enxerga um continuo temporal de imagens bidimensionais projetadas na retina.

Aplicando os mesmos princípios de amostragem, temos que discretizar o tempo para termos um número finito de imagens em um determinado período. Novamente recaímos no problema de escolher a taxa de amostragem.

Considere uma fonte de luz que permanece acesa por um certo tempo, depois permanece apagada pelo mesmo tempo, repetindo esse procedimento ininterruptamente. Observou-se que a partir de um determinado tempo o olho não percebia mais que a luz estava piscando ou cintilando, mas que isso dependia do ambiente onde estava o observador. Isso é chamado de persistência da visão. É pelo mesmo motivo que quando movemos rapidamente uma fonte luminosa ela parece “riscar” o ar.

A lei de Ferry Porter diz que a freqüência crítica de fusão de cintilação é proporcional ao logaritmo da iluminação ambiente média. Analisando essa lei podemos constatar que quanto maior for a iluminação ambiente maior será a freqüência de percepção da cintilação ou menor será o tempo para que o observador não perceba a cintilação.

Na tabela a seguir podemos comparar alguns ambientes e as taxas de redesenho utilizadas.

| Ambiente | Iluminação | Taxa de Resenho |
|--------------|-------------|-----------------|
| Cinema | Escuro | 48 |
| Tv Européia | Pouco Claro | 50 |
| Tv Americana | Pouco Claro | 59,94 |
| Escritório | Claro | 72 |

Tabela 5: Iluminação Ambiente e Taxa de Redesenho

Mas foi observado também que o nosso sistema de visão não consegue distinguir muito bem objetos em movimento. Assim em vez de exibirmos uma imagem diferente a cada taxa de redesenho podemos exibir a metade desse número que o observador aceitará isso como um movimento contínuo.

Assim, no caso do cinema estipulou-se uma taxa de 24 quadros por segundo. Mas cada quadro é exibido duas vezes.

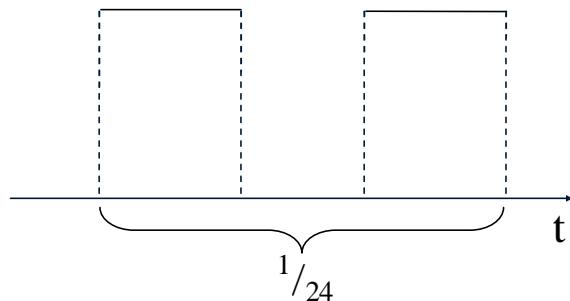


Figura 35: Tempo de Redesenho de Quadros em Cinema

Na transmissão de um sinal de vídeo para Tv, foi necessário uma outra estratégia. Para ocupar uma banda de transmissão menor e levando em consideração que o sinal é transmitido e visualizado simultaneamente decidiu-se transmitir metade da imagem dividindo-a em linhas ímpares e pares. Assim, metade da imagem se formaria na tela mas com a impressão de uma imagem completa, que no próximo instante a outra metade completaria o restante da imagem. Essa divisão de cada imagem em 2 metades de linhas pares e ímpares foi chamado de entrelaçamento de linhas ou quadros entrelaçados. Cada metade é chamada de campo. O número de quadros (inteiros) por segundo cai pela metade da taxa de redesenho.

O modo tradicional de redesenho é chamado de Progressivo. O modo Entrelaçado desenha as linhas ímpares, depois as linhas pares, depois as ímpares do próximo quadro e assim por diante sem apagar as outras linhas visíveis. Com isso temos dois efeitos, um de termos o “dobro” do número de linhas e o “dobro” do número de quadros, mas com “metade” da informação.

O nosso sistema de visão é enganado a várias décadas por esse truque da televisão convencional. De fato esse truque viabilizou a implantação da televisão, mas introduziu uma série de defeitos que com o tempo passamos a aceitar. Se analisarmos campo a campo de um vídeo digital baseado nos sistemas de Tv entrelaçados veremos que objetos em movimento possuem um deslocamento de campo para campo, coisa que não acontece em um quadro de cinema. Ou seja não podemos tratar um vídeo entrelaçado de 60 campos por segundo da mesma forma que um vídeo progressivo de 30 quadros por segundo.

Assim como, na geração de animação modelada por computador para vídeo entrelaçado temos que levar isso em consideração gerando 60 campos, cada campo com a metade do número de linhas e com um deslocamento de 1 linha na vertical alternadamente entre cada campo.

Na realidade no sistema NTSC não são exatamente 60 campos. Em 1953, o número de quadros por segundo foi modificado para que pudesse ser introduzida cor ao padrão. Assim foi reduzido por um fator de $\frac{1000}{1001}$ resultando em 29.97 quadros por segundo ou 59.94 campos por segundo. Esse número também é fundamental ser utilizado, pois somente dessa forma o som estará em sincronia com a imagem. Utilizar 60 quadros por segundo introduz um pequeno erro que acumulado ao longo do vídeo gera uma defasagem entre o som e a imagem.

Uma conversão muito comum é a de filme em vídeo, ou de 24 quadros por segundo para 60 campos por segundo. A forma mais eficiente de realizar isso é utilizando um sistema de redução de 3-2, ou melhor para cada quadro de cinema gera ora 3 campos de vídeo ora 2 campos de vídeo. A figura a seguir ilustra esse processo. No caso da taxa de 59,94 o que é feito é realizar a conversão 3-2 rodando o filme 0.1% mais lento.

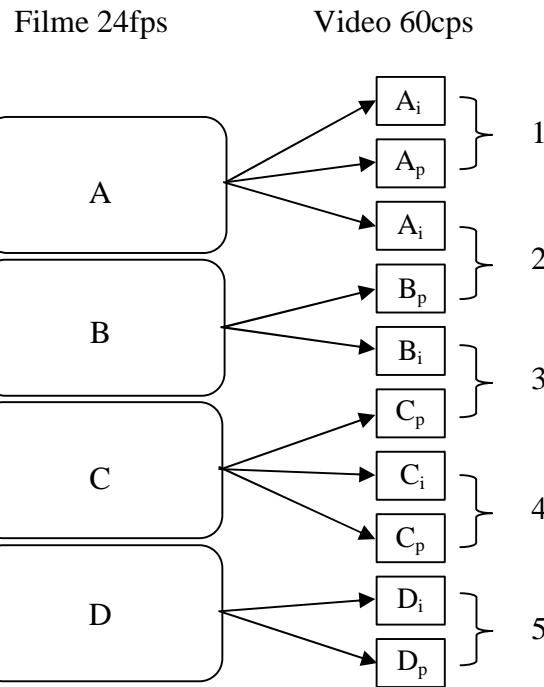


Figura 36: Conversão 3-2 de Cinema para Vídeo

O modo entrelaçado é tão eficiente que foi mantido para o padrão de televisão digital e HDTV. Neste padrão existe a possibilidade de transmissão e recepção em vários modos. O modo progressivo é necessário para aplicações onde a precisão de cada pixel se faz necessária. O modo entrelaçado é usado para a transmissão de vídeo em tempo real com uma qualidade aparentemente superior com praticamente a mesma quantidade de informação do modo progressivo equivalente. Veja na tabela abaixo.

| Notação | Resolução | Modo | Quadros / s | Taxa sem Compressão |
|---------|-----------|-------------|-------------|---------------------|
| 720p | 1280x720 | Progressivo | 60 | 55Mpixels/segundo |
| 1080i | 1920x1080 | Entrelaçado | 30 | 62Mpixels/segundo |

Tabela 6: Comparação entre Dois Modos de Transmissão em HDTV

Convém observar que em HDTV existem formatos com taxa de 24, 30 e 60 quadros por segundo (e taxas modificadas pelo fator $^{1000}/_{1001}$). Assim como, imagem com razão de aspectos 4:3 e 16:9.

6. Captura

Existem diversos mecanismos de captura de imagens. Os mais conhecidos são o Scanner, o Digitalizador de Sinal de Vídeo (Frame/Vídeo Grabber) e a Câmera Digital.

6.1. Scanners

Scanners provêm as imagens de mais alta resolução, mas em compensação são mais caros. Podem capturar diversos tipos de superfície, as mais importantes são papel e filmes fotográficos. Utilizam o sistema de cor RGB e possuem uma calibração fácil de ser feita. O procedimento de calibração requer um modelo impresso, em geral fornecido pelos programas que aceitam calibração. Esse modelo contém uma série de cores, conhecidas ao programa. Assim, quando a imagem é capturada o programa compara os valores obtidos com os valores esperados e pode então gerar um perfil de correção para aquele scanner.

Os scanners são muito semelhantes à máquinas Xerox (fotocopiadoras). Existem basicamente três tipos: manuais, de mesa ou de cilindro. Em qualquer caso o scanner dispõe de uma fonte de luz em forma de uma linha que varre a imagem impressa e mede a quantidade de luz refletida ou transmitida em cada ponto. A luz captada é convertida em um sinal elétrico através de um conjunto de foto-detetores que também formam uma linha. O sinal elétrico é finalmente digitalizado e enviado ao computador.

No caso de scanners manuais a varredura é feita deslizando o aparelho sobre a imagem a ser capturada. No de mesa, em processo análogo ao da Xerox, a varredura é realizada automaticamente pelo aparelho que desloca a fonte de luz e os foto-detetores. No de cilindro tanto a fonte de luz quanto a imagem se movem para fornecer maior qualidade ao processo. Neste caso a fonte de luz se move dentro do cilindro enquanto a imagem é fixada na sua superfície que gira durante o processo.

A escolha da DPI a ser utilizada é o fator mais crítico no uso do scanner. A primeira coisa que se deve ter em mente é "Qual o objetivo ao digitalizar a imagem?". Uma solução é encarar três tipos de resultados: para visualização somente em monitores, para edição e impressão futura, e para análise científica.

No primeiro caso, a resolução máxima em pixels do monitor vai ditar o tamanho máximo em pixels da imagem. Imagem capturadas para o uso na Internet se encaixam neste caso.

No segundo caso, a maior DPI da impressora a ser utilizada vai ditar a DPI a ser utilizada no scanner durante a captura. Mas, no caso de impressoras a 2 cores, deve-se compensar o tamanho da matriz de dithering. Por exemplo, se for 6x6 em 600 DPI, basta enviar uma imagem de 36 tons de cinza em 100 DPI.

No terceiro caso, onde o objetivo é obter medidas na imagem, o que vai indicar a resolução é a precisão dos valores que se deseja obter no espaço da imagem. Esta precisão indicará a menor distância entre dois pixels, o que por consequência ditará a DPI.

6.2. Digitalizadores de Sinal de Vídeo

Digitalizadores de Sinal de Vídeo dependem do padrão de vídeo utilizado, NTSC, PAL-M, etc. Em geral são de baixa resolução. Alguns permitem capturar um único quadro, chamados de *Frame Grabbers*, outros permitem gravar seqüências inteiras em tempo real, são chamados de *Vídeo Grabbers*. Algumas placas gráficas já incorporam, além da saída para o monitor de vídeo, a saída para vídeo analógico compatível com os padrões conhecidos, e possuem também entrada para vídeo analógico e para RF, ou seja, pode-se ligar o vídeo cassete no computador para capturar e gerar novos vídeos, assim como usar o computador como sintonizador de TV (sinal de Rádio Freqüência).

Uma imagem de vídeo gerada por uma câmera ou um videocassete pode ser digitalizada diretamente a partir do sinal elétrico que a representa, que em geral está disponível como uma saída do equipamento.

O sinal de vídeo é organizado em linhas de imagem, que contém um sinal analógico cada uma. Portanto, o número de linhas da imagem já é praticamente definido pelo próprio sinal de vídeo; para os sistemas NTSC e PAL-M esse valor é 525 e para o sistema PAL-G é 625. Dentre estas apenas aproximadamente 480 são visíveis (na realidade somente de 240 a 360 linhas do sinal de vídeo contém informação relevante).

O número de colunas é normalmente escolhido de acordo com o número de linhas para manter a razão de aspectos da imagem. A escolha do número de colunas também é limitado pela banda do canal de vídeo, que para o sinal padrão NTSC é da ordem de no máximo 750 pixels por linha. Nas televisões comerciais existem cerca de 440 pontos numa linha horizontal.

A calibração em alguns casos é feita ajustando-se o nível de branco e o nível de preto. Além disso, é preciso não esquecer de regular a própria câmera, realizando o que se chama de "bater o branco", ou seja, para uma determinada luz informar para a câmera o nível de branco para a gravação. As câmeras domésticas tentam realizar essa calibração automaticamente.

Quando se está trabalhando diretamente com o conjunto câmera/digitalizador, após a captura da imagem, pode-se tentar melhorar a qualidade da mesma com um método muito simples. Com a câmera desfocada e/ou com um fundo neutro, como um papel cinza, captura-se uma imagem, esta imagem ao ser subtraída da imagem normalmente capturada elimina sujeiras que por ventura estejam na lente ou em um fundo irregular advindo de problemas de alinhamento do equipamento, etc.

6.3. Câmeras Digitais

Câmeras Digitais usam um dispositivo chamado CCD (*Charge Coupled Device*) que atua como se fosse o filme fotográfico. Alcançam resoluções de 640x480 até milhares de pixels. Os Line Scan CCDs são usados para altas resoluções, com alta linearidade, alta banda dinâmica e readout lento e possuem qualidade exceptional. Mais recentemente surgiram algumas câmeras que utilizam CMOS (*Complementary Metal-Oxide Semiconductor*) como sensores. A grande vantagem é a viabilidade industrial pois utilizam a mesma tecnologia de semicondutores tradicionais para produção em baixo custo.

O sistema de cor é RGB. A calibração pode ser feita exatamente como um scanner, mas a qualidade da imagem agora depende de fatores externos, como iluminação do objeto. Algumas câmeras utilizam 3 sensores para capturar a imagem, um para cada componente de cor R,G e B.

As câmeras de vídeo domésticas que conhecemos utilizam esse dispositivo. A limitação da imagem gerada por estas câmeras está associada ao padrão de vídeo analógico utilizado para armazenar o vídeo na fita.

Mais recentemente com a padronização de vídeo digital, já pode-se armazenar a imagem capturada com qualidade muito superior a do vídeo tradicional. Com exceção do Cinema Digital que está em fase final de padronização, continuamos muito limitados a qualidade do padrão de televisão atual, no melhor caso temos uma resolução de 720x480 em modo entrelaçado, o que na prática para imagens em movimento diminue a qualidade para 720x240.

Já as câmeras digitais voltadas apenas para fotografias possuem resoluções muito mais altas, pois não estão limitadas pela padronização do sistema de vídeo.

6.4. Parâmetros de Câmeras Digitais

Câmeras digitais funcionam exatamente como câmeras tradicionais, mas possui alguns parâmetros a mais e outros parâmetros são mapeados de forma diferente no processo de captura.

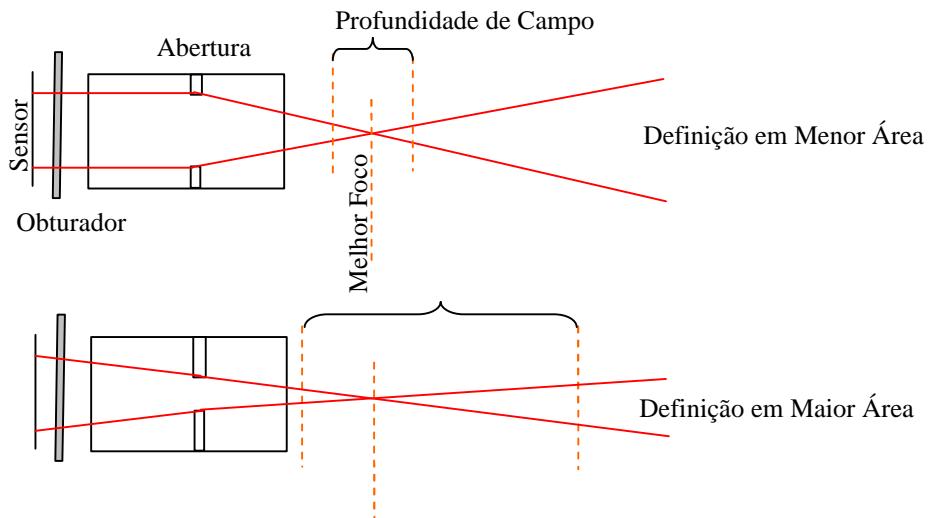


Figura 37: Parâmetros de uma Câmera Fotográfica

Toda câmera possui os seguintes parâmetros:

- Abertura do Diafragma (Aperture - na Lente)
- Distância Focal (Focus - na Lente)
- Fator de Ampliação (Zoom - na Lente)
- Velocidade do Obturador (Shutter – na Câmera)
- Fotômetro (na Câmera)
- Sensor (Filme ou CCD)

Abertura – define a quantidade de luz que chega ao sensor, e está relacionada com a profundidade de campo. A numeração convencional para abertura é diretamente proporcional a profundidade de campo, e inversamente proporcional à abertura. Profundidade de campo é a região onde existe foco.

Obturador – define o tempo que o sensor ficará exposto àquela quantidade de luz. Está relacionado com a velocidade do objeto a ser fotografado. Se muito rápido congelará o movimento, se muito lento borrará o movimento.

Sensor – sendo filme ou CCD possui uma determinada sensibilidade. Se muito sensível, pode incluir muito ruído, se pouco sensível precisará de muita luz. Além disso depende do tipo de iluminação ambiente.

O filme pode ser para luz do dia e para luz de tungstênio. O CCD precisa regular o branco (White Balance). Hoje em dia existem filmes bem versáteis, assim como já existe regulagem automática do branco. Assim como cada um deles possuirá uma referência colorimétrica diferente. O filme funciona aproximadamente com o sistema CMY, o CCD com o sistema RGB. O número ISO do filme define sua sensibilidade e granularidade, no caso do CCD mudar o ISO significa apenas aumentar o ganho do sensor. A sensibilidade da câmera digital vai ser dada pelo número de Lux, e a qualidade da foto fica limitada pelo número de elementos (*pixels*) no CCD.

Em ambas as câmeras precisamos de um **Fotômetro** para que dada a combinação Sensor+Obturador+Abertura nos informe se existe uma quantidade excessiva ou uma falta de luz chegando ao sensor. Na grande maioria das câmeras essa combinação é feita automaticamente baseada no sensor para uma condição média para a profundidade de foco, em geral a maior possível (AE – Auto Exposure). Algumas câmeras contornam isso com programas para várias situações, onde dependendo do sujeito o operador escolhe o programa.

Note que para cada ponto da marcação desses parâmetros o dobro ou a metade da luz chega ao sensor.

Nos CCDs não é possível realizar exposições muito longas.

6.5. Escolhendo um Sistema de Captura

Scanners possuem a vantagem de possuirem altas resoluções e operarem sob condições de luz totalmente sob controle. Mas os dados precisam ser planos e colocados sob sua superfície para aquisição.

Câmeras são mais flexíveis e portáteis, podem adquirir dados em condições diversas, mas são afetadas pelas condições de iluminação e possuem resolução muito menores do que scanners. Além disso será necessário uma preocupação extra com lentes e acessórios fotográficos que possam por ventura auxiliar no processo de captura. Uma outra vantagem do uso de câmeras é que você pode acompanhar o processo em tempo real.

6.6. Transferindo Imagens Digitais

Imagens e vídeo digitais são transferidos para o computador de diversas formas.

Scanners em geral usam o padrão TWAIN para registrar um programa do fabricante que é chamado sempre que se queira capturar uma imagem.

Em câmeras fotográficas digitais o método mais comum é via porta serial, seja USB ou tradicionais COMs. Algumas câmeras possuem uma memória de cartão ou CDs regraváveis que podem ser levados diretamente ao computador.

Em câmeras digitais o padrão mais comum é o IEEE 1394 ou Firewire. É um barramento de alta velocidade que permite transferir o vídeo rapidamente para o computador sem placas adicionais.

Mais recentemente a Microsoft incluiu no seu sistema operacional um padrão para transferência de imagens o WIA (Windows Image Aquisition). Aos poucos os fabricantes estão aderindo a esse padrão que pode funcionar para qualquer tipo de barramento, incluindo USB e IEEE 1394.

7. Visualização

Há duas formas básicas de visualização, a tela e a impressora, embora existam outros dispositivos, como por exemplo gravadores de slides, vamos nos concentrar nestes dois.

Monitores e projetores são sistemas ativos, e impressoras são sistemas passivos. Em ambos os casos a distância do usuário à superfície de visualização é fundamental para a correta percepção da imagem reconstruída. Perceptualmente o nosso olho é capaz de distinguir detalhes em um ângulo de visão de aproximadamente 1/60 graus. Assim a distância ótima para visualização é aproximadamente dada por:

$$Distância \cong \frac{Altura}{Linhas} \left/ \tan\left(\frac{1}{60}\right)\right.$$

Equação 14: Distância Ótima para Visualização

Muito próximo perceberemos artefatos da digitalização, muito distante estaremos subutilizando o dispositivos de visualização.

$$Resolução \cong 1 \left/ \tan\left(\frac{1}{60}\right) \cdot Distância \right.$$

Equação 15: Resolução Ótima para Visualização

Para a visualização em papel a aproximadamente 30cm de uma fotografia precisamos de uma resolução de aproximadamente 300DPI. Observe na tabela abaixo que monitores ainda possuem uma resolução muito baixa para a distância normal com que trabalhamos, já não é o caso de televisões, um motivo para pessoas dizerem que a imagem da Tv é “melhor” que a do monitor.

| Distância | Resolução | Pixels | Exemplo |
|-----------|-----------|-----------------|----------|
| 30cm | 300DPI | A4 – 6300x8100 | Papel |
| 60cm | 150DPI | 15” – 1600x1280 | Laptop |
| 90cm | 100DPI | 17” - 1280x1024 | Monitor |
| 3m | 30DPI | 19” – 720x486 | Tv |
| 7m | 15DPI | 100” – 1024x768 | Projetor |

Tabela 7: Resoluções Ótimas Aproximadas para Distâncias Comuns

Como vimos, o nosso olho se adapta ao ambiente de acordo com as condições de iluminação. Ou seja, a qualidade da visualização também é afetada pelo ambiente onde é realizada. Para verificarmos isso basta observar a figura a seguir para notarmos que o quadrado central, embora tenha a mesma intensidade em ambas as situações, parece ser mais escuro quando num fundo claro e mais claro quando num fundo escuro. Isso nos leva a produzir mecanismos de visualização diferentes para diferentes ambientes.

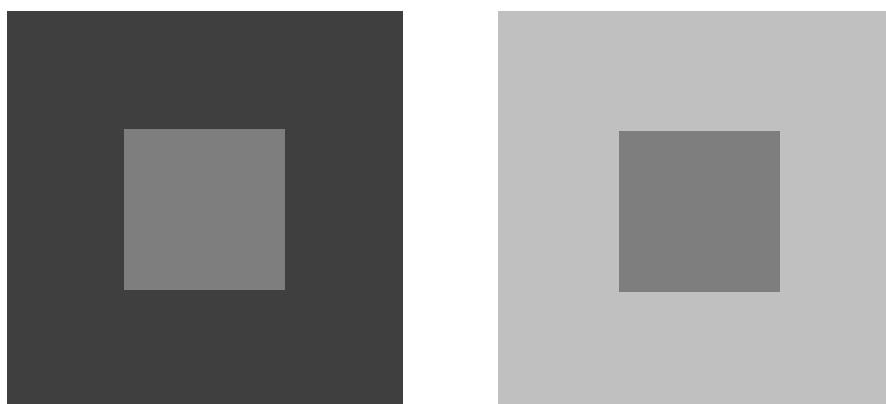


Figura 38: Efeito da Vizinhança na Percepção

No caso de imagens impressas, supõe-se um ambiente muito bem iluminado, sem contar com a ajuda do papel ser normalmente branco. Monitores e projetores (sistemas ativos de visualização) precisam tirar proveito da persistência do nosso sistema de visão redesenhandando a tela

ininterruptamente no tempo necessário para que a imagem apareça sem cintilação, como vimos no tópico de **5. Resolução Temporal**.

A grande maioria dos monitores e projetores possui a razão de aspectos de 4x3, mas já existem monitores com a razão de aspectos de 16x9 chamada de Wide Screen, muito utilizada em DVDs. Veja na figura a seguir uma comparação entre algumas razões mais comuns.

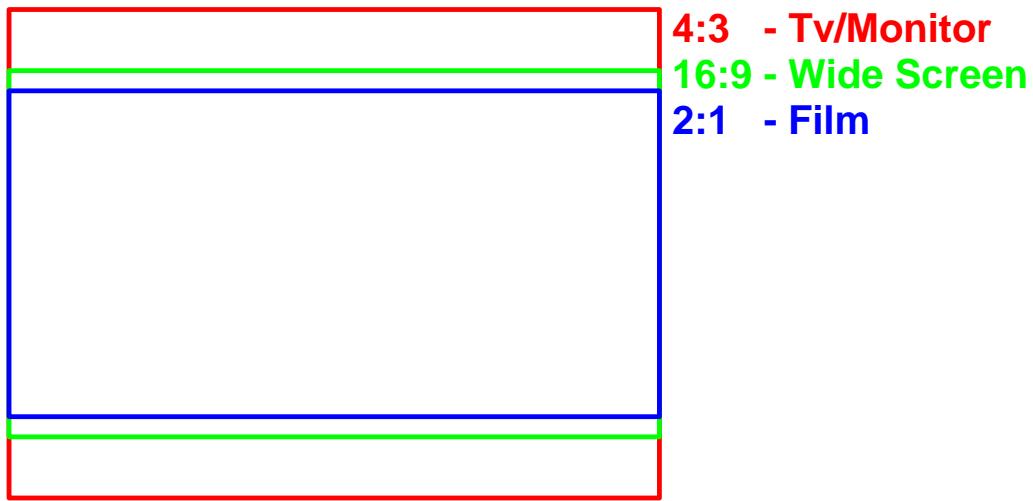


Figura 39: Comparação entre Razões de Aspectos Tradicionais

7.1. Monitores Analógicos

Atualmente existem diversas tecnologias de monitores, os mais comuns funcionam como as televisões comerciais usando tubo de raios catódicos (C.R.T.). Mas a distância entre os centros dos três fósforos RGB do monitor é de 0,21mm e da TV é de 0,60mm (essa distância é chamada de dot pitch). São chamados de dispositivos matriciais de rastreamento, onde a imagem precisa ser reescrita na tela a cada determinado intervalo de tempo (isto é um dos motivos das imagens digitais serem chamadas de *raster images*). Por consequência, a imagem gerada para toda a tela do monitor precisa ser guardada em algum dispositivo; este dispositivo é a placa gráfica.

O monitor utiliza o sistema RGB e possui um número determinado de pixels na horizontal e na vertical. Mas a cor é contínua, depende apenas das cores fornecidas pela placa gráfica. Esta irá dizer se estamos trabalhando em 640x480 com 256 cores ou 1280x1024 com 24 bits, por exemplo. Assim, a visualização no monitor é dependente do melhor ajuste entre ele e a placa de vídeo. Geralmente possuem baixa resolução, aproximadamente 72 DPI.

Vimos que a imagem formada na tela, está sendo redesenhada a cada instante. Se este redesenho não é rápido o suficiente aparece o que chamamos de *flickering* (cintilação). A imagem parece estar piscando. Isso ocorre porque a freqüência de rastreamento é muito baixa para a configuração escolhida. É preciso saber se, tanto o monitor, quanto a placa gráfica, suportam uma freqüência aceitável para a configuração desejada. Em geral, valores acima de 75Hz são excelentes, e 60Hz é aceitável.

É interessante observar que a razão de aspectos de um pixel na tela é 1x1, mas um ajuste errado do monitor pode provocar distorções. As resoluções de tela são múltiplos de 4x3 (640x480, 800x600, 1024x768) para que justamente resultem em *pixels* quadrados. Para verificar isso, veja se a área de visualização no monitor está na razão 4x3. Um outra maneira de fazer a calibração de aspectos é simplesmente desenhar um quadrado em *pixels*, usando por exemplo um programa simples de pintura, e medir esse quadrado com uma régua na tela do monitor, se estiver distorcido então deve ajustar o monitor até que o quadrado esteja correto.

A calibração de cor de monitores requer um dispositivo que é colocado na sua tela e retorna a informação de cor necessária para o programa de calibração. Mas neste caso, a calibração também depende da luz ambiente.

Mesmo não possuindo esse dispositivo pode-se fazer um ajuste mínimo do monitor. Primeiro, coloque na tela uma figura com uma grande área preta e outra branca. Coloque o controle  no mínimo. Reduza o controle  até que o fundo preto da tela fique correto e não com um certo brilho, mas não reduza demais, só o necessário (se aumentando esse controle, logo acima deste nível, a tela ficar com uma cor escura em vez de um cinza escuro, o seu monitor está com um problema de calibração de um dos canhões, deve-se levá-lo à um técnico). Agora, ajuste o controle  de maneira que o branco mostrado fique com uma intensidade adequada para você. Resista à tentação de colocar o brilho do monitor muito alto, isso tem uma série de desvantagens além de piorar a qualidade da imagem.

Os controles  e  são chamados, para monitores modernos, de “Nível de Preto” e “Imagem” (“Black Level” and “Picture”), respectivamente. O controle  ajusta o nível de preto, que faz com que toda a imagem fique mais clara ou mais escura, mas na realidade está maximizando o contraste. O controle  ajusta o nível de branco e por consequência todos os

valores intermediários até o preto, sem modificar o nível de preto, isso está na realidade está ajustando o brilho. Por razões históricas, esses símbolos estão associados aos nomes de “Brilho” e “Contraste”, respectivamente, por causa da sua simbologia.

Uma curiosidade sobre monitores e televisão é que as TVs possuem qualidade muito menor que a dos monitores, mas que visualmente as pessoas ficam satisfeitas com o resultado. Isso também acontece porque, quando um ponto é impresso na tela da TV, ele na realidade sofre um fenômeno de espalhamento fazendo com que a imagem fique com contornos mais suaves e consequentemente agradável para a vista. Além do que, a graduação tonal é muito mais suave por ser contínua, e no caso do monitor pode estar limitada pela placa de vídeo.

7.2. Monitores Digitais Planos e Projetores

Inicialmente usados em Laptops, utilizam em geral de tecnologia LCD (Liquid Crystal Display). Neste caso um cristal líquido é iluminado por trás e controlado usado eletrodos sobre o cristal e luz polarizada de forma a deixar passar apenas as cores desejadas.

Como a luz atravessa o cristal, bloquear totalmente a mesma nem sempre é eficiente e em alguns projetores aparece o problema de que não se consegue um preto absoluto.

Devido a persistência do cristal não é necessário varrer a tela com uma alta taxa de redesenho e a grande maioria trabalha com a freqüência de 60Hz. Por isso mesmo o LCD não consegue ter um tempo de resposta muito alto e por exemplo fica inviável um sistema com estereoscopia. Assim como quando a imagem projetada possui movimentos rápidos pode-se perceber algumas manchas.

Compare o tempo de permanência da imagem em monitores CRT e LCD na figura abaixo.

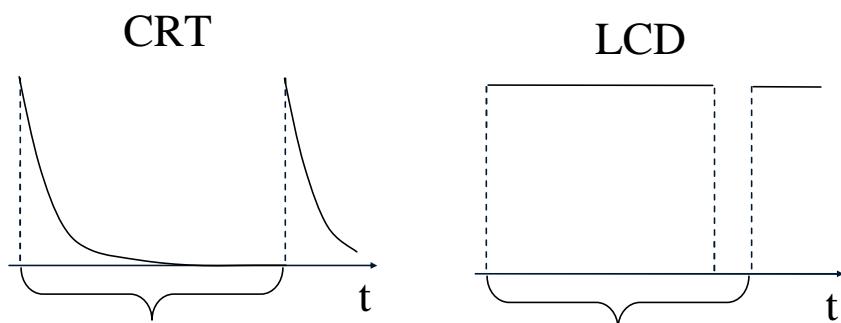


Figura 40: Tempo de Redesenho de um CRT versus um LCD

Compare a distribuição de pontos entre monitores CRT e LCD na figura abaixo. Nos monitores CRT os pontos são distribuídos em tríades RGB que se somam para formar um pixel. Nos monitores LCD são barras laterais, repare que como os cristais filtram cor, esse controle é feito pelo complemento CMY, embora o sinal seja RGB.

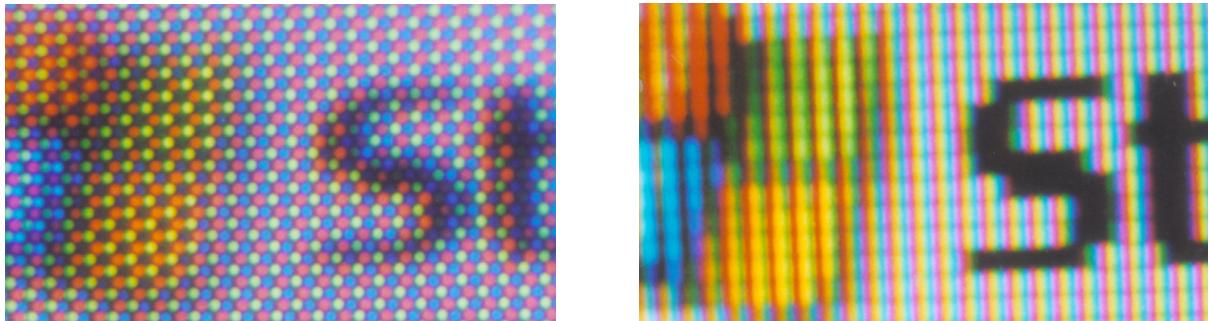


Figura 41: Macro Fotografias de Monitores CRT e LCD respectivamente

A tecnologia LCD facilitou muito a difusão de projetores portáteis. Mas já estão surgindo projetores baseados na tecnologia DLP (usando o chip DMD da *Texas Instruments - Digital Micromirror Device* ou *Deformable Mirror Displays*). Neste caso não existe a necessidade de luz polarizada, e a grande vantagem é também utilizar tecnologia de semicondutores amplamente difundida na indústria. No momento ainda é mais caro do que o LCD e ainda está em evolução. Neste caso eles são capazes de reproduzir apenas 2 tons. A sensação de uma gradação de tons é atingida expondo a luz por mais ou menos tempo.

7.3. Impressoras

Impressoras usam tecnologias muito variadas. Por isso, o processo de calibração é muito complexo e as vezes não gera bons resultados. Este depende de muitos fatores: papel, tinta, até umidade do ar. E existe uma grande variedade de dispositivos para fazê-lo. Em contra partida, impressoras chegam a altíssimas resoluções.

O sistema de cor na maioria das impressoras é CMYK, mas existem algumas poucas impressoras que usam o modelo RGB. Algumas impressoras operam apenas a 4 cores, apenas 1 ton para cada componente. Neste caso utilizam resolução espacial para compensar a baixa resolução cromática, com por exemplo técnicas de *dithering* vistas no tópico de Quantização de Cor.

Por isto existem duas grandes problemáticas envolvendo impressão de imagens. A primeira é relacionada a fidelidade da cor impressa, pois necessariamente precisamos converter de RGB para

CMYK. A segunda é que a grande maioria das impressoras imprime somente com apenas 4 cores, que não são combinadas. Assim, calibração de cor e as técnicas de dithering são fundamentais para obtermos resultados de alta qualidade.

As impressoras de jato de tinta possuem resoluções espaciais inferiores as das impressoras laser, mas em contrapartida possuem resolução cromática muito melhor, podendo reproduzir até 16 milhões de cores, enquanto que a grande maioria das impressoras laser possuem apenas 2 cores, o branco do papel e o preto. Neste último caso, para imprimir tons de cinza, a impressora usa resolução espacial para simular tons enganando o olho, usando a proximidade entre os pontos desenhados.

Levando esses fatos em consideração, devemos prestar atenção durante a inserção de imagens em documentos para não utilizarmos tamanhos muito grandes que não vão ser aproveitados pela impressora e vão aumentar o tamanho do documento, assim como aumentar o tempo de impressão, desnecessariamente.

8. Armazenamento

Uma vez capturada a imagem, esta é armazenada na memória do computador como uma seqüência de bytes, ou melhor um mapa de bits (Bitmap). Este armazenamento é feito independente da resolução e número de cores do monitor (Device Independent Bitmap - DIB), embora alguns programas simples, como o PaintBrush do Windows 3.1, que usa um bitmap dependente da configuração de tela.

Como vimos anteriormente, a imagem digital é simplesmente uma matriz de pixels, onde cada pixel é um número (ou 3, etc...) que representa o valor da cor naquela posição.

Quanto maior o número de pixels, maior o tamanho da imagem na memória. Mas qual o tamanho de cada pixel?

Isso vai depender de obviamente do sistema de cor e do número de bits usado para representar cada componente. Se a imagem é monocromática (tons de cinza) terá apenas 1 componente, se for colorida no sistema RGB terá 3 componentes, e assim por diante.

Cada componente será quantizada por um determinado número de bits. Em geral, usam-se apenas 8. A partir da versão 4.0 o sistema Photoshop, por exemplo, passou a tratar imagens com 16 bits por componente. Cuidado para não confundir esse número com o valor de 16 bits de cor de algumas resoluções de display, neste caso usam-se 5 bits para cada componente RGB, e um bit é flexível, normalmente usado para dar ao verde 6 bits. Já em sistemas de processamento de imagens

científicas é muito comum tratar imagens com números inteiros com 32 bits que incluem valores negativos, ou até mesmo números de ponto flutuante (valores com várias casas decimais de precisão).

Quando falamos de 8 bits, falamos de 2^8 (256) possibilidades de valores diferentes para cada componente. Assim, se são 3 componentes temos $2^8 \times 2^8 \times 2^8 = 2^{24}$ (também chamada de resolução de cor *True Color*), totalizando aproximadamente 16 milhões de cores possíveis. É claro que imagem alguma usa toda as cores disponíveis ao mesmo tempo.

Esta imagem mostra o efeito de reduzirmos a resolução e/ou reduzirmos o número total de cores de uma determinada imagem.

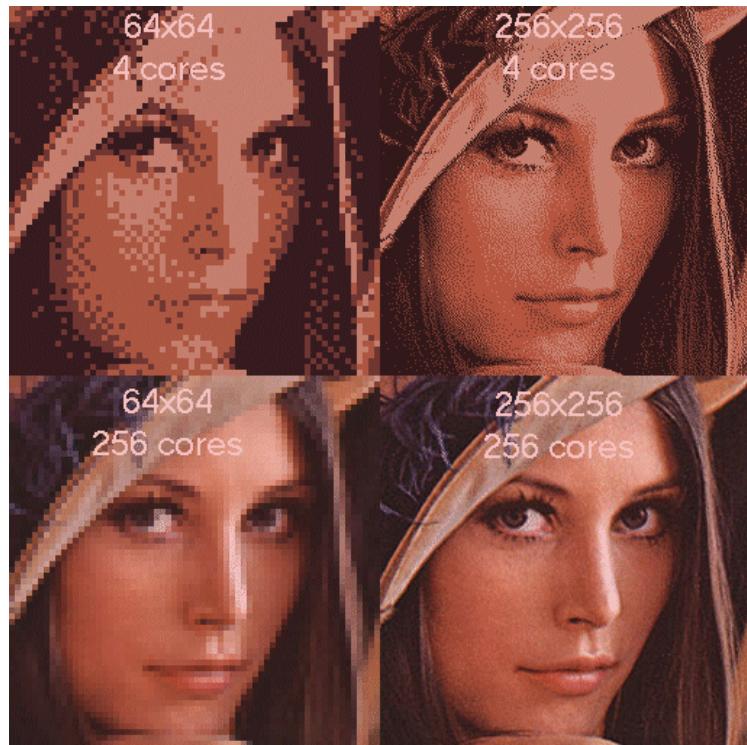


Figura 42: A Mesma Imagem em Diferentes Resoluções e Número de Cores

O tamanho em bytes de uma imagem pode ser calculado com:

$$\text{Size} = \text{Height} \times \text{Width} \times \text{Depth} \times \text{BytesPerComponent}$$

Equação 16: Tamanho de uma Imagens em Bytes

onde largura e altura são o número de pixels na horizontal e vertical, respectivamente; profundidade é o número de componentes do sistema de cor (RGB=3, CMYK=4, Tons de Cinza =1, ...).

O número de bytes por componente geralmente é 1, mas pode ter quaisquer valor dependendo a precisão desejada. São comuns também valores menores que um byte, tais como 4 e 1 bit. Neste caso o cálculo do tamanho depende de como a imagem é estruturada em memória.

Um outro termo utilizado é BPP, bits por pixel que corresponde a Profundidade x BPC, totalizando o número de bits efetivo que cada pixel ocupa.

Vejamos alguns exemplos do cálculo do tamanho ocupado em memória por uma imagem:

Exemplo 1:

Imagen de 1024x768 com 24 Bpp = 2,25 Mb

Exemplo 2:

Imagen 15cm x 10cm capturada com 300 DPI em True Color (24 Bpp)

$$\text{Largura} = 15 \times 300 / 2,54 = 1771$$

$$\text{Altura} = 10 \times 300 / 2,54 = 1181$$

$$\text{Tamanho} = 1771 \times 1181 \times 3 = \sim 6 \text{ Mb}$$

Exemplo 3:

A mesma imagem com 600 DPI

$$\text{Tamanho} = \sim 24 \text{ Mb}$$

Exemplo 4:

Página A4 com 600 DPI = ~100Mb

Historicamente, computadores não possuíam uma interface gráfica tão poderosa quanto temos hoje. Assim, foram definidas simplificações para o modelo RGB, que embora diminuam significativamente o número de cores disponíveis, reduz o tamanho da imagem em memória. Essas imagens são chamadas de indexadas ou coloridas com palheta de cores (*indexed images, palette based images*).

Nessas imagens o valor do *pixel* contém um único número que corresponde a um índice para uma tabela, essa tabela é que conterá o valor da cor em RGB. Imediatamente podemos ver que as 3 componentes são reduzidas para 1, isso já é uma grande economia já que o tamanho da palheta é pequeno perto do tamanho da imagem. As imagens tons de cinza podem ser vistas como um caso particular onde a palheta contém os valores de R, G e B iguais.

| Tabela de Cores | | | |
|-----------------|-----|-----|-----|
| ÍNDICE | R | G | B |
| ... | ... | ... | ... |
| 123 | 255 | 32 | 128 |
| | | | |

Tabela 8: Mapeamento de Índice na Tabela de Cores

8.1. Compressão

Observando os exemplos anteriores vemos que não há computador que resista a tanto pixel. Por isso foram criados, e ainda são muito estudados, métodos de compressão de imagem. Esse métodos podem ser classificados em dois tipos, com perdas em sem perdas. Os métodos com perdas são baseados no fato de que pequenos detalhes podem ser eliminados que perceptualmente não serão notados de imediato.

Dentre os métodos com perdas se destacam o JPEG (Join Photographic Experts Group), criado a partir de uma composição de várias técnicas e adotado amplamente. Um outro método muito bom é chamado de Fractal, embora pouco use da matemática de fractais. Esse método infelizmente é propriedade privada de uma empresa que comercializa seus direitos de uso.

Dentre os métodos sem perdas, podemos enumerar vários: Código de Huffman, RLE (Run Length Encoding), LZW (Lempel-Ziv & Welch), JBIG (Join Bi-level Image Experts Group). Cada um desse métodos utiliza uma técnica diferente que em geral lhe dá o nome.

No caso de vídeo, a grande maioria dos métodos são com perdas. Dentre eles destacam-se Cinepack, Indeo e MPEG,. Muitas placas de vídeo já possuem decodificadores para MPEG implementados em hardware, otimizando muito o desempenho de tocadores de vídeo.

O mais simples de todos e também o mais fácil de entender, é o RLE. Esse método usa o fato de que imagens podem possuir muitos pixels repetidos em seqüência numa mesma linha. Assim, em vez de armazenar todas essas repetições, armazena-se o valor repetido e o número de repetições. Obviamente esse método só funciona bem para imagens com muitas áreas de uma mesma cor.

Na compressão com perdas é importante conhecer como o nosso sistema de visão funciona para descartarmos as informações menos importantes.

Como vimos, grande parte da informação de cor abosrvida por nós está na luminância. Assim podemos separar o sinal de cor em luminância e crominância e descartar alguma informação de crominância. No caso usando menos bits para crominância. A notação “4:2:2” muito vista na especificação de sistemas é referente a essa compressão. Para esse exemplo, a cada 4 amostras usamos 4 valores para armazenar os respectivos valores de luminância, mas utilizamos apenas 2 valores para cada uma das componentes restantes da decomposição Y, R-Y e B-Y. Na realidade os dois últimos números também definem como a quatização será feita espacialmente.

Em áreas com muitos detalhes a diminuição dos mesmos passa despercebida. Asssim podemos aplicar quantizações mais grosseiras em áreas de alta freqüência espacial.

Observando a figura 29, na página 42, podemos observar que entre cada faixa de tom de cinza percebemos um efeito de aumento do contraste. Isso é devido ao efeito das bandas de Mach. Nas bordas abruptas percebemos uma transição não linear que também pode ser usada para aumentar a taxa de compressão nessa região já que o nosso sistema se encarrregará de aumentar a percepção da borda.

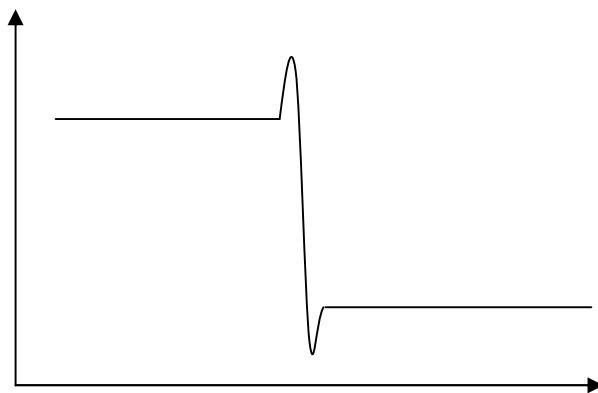


Figura 43: Efeito Perceptual das Bandas de Mach

O nosso sistema de visão detecta movimento muito bem, mas não detecta todos os detalhes de objetos em movimento. Assim, para a compressão de vídeo quanto maior a velocidade de objetos de quadro para quadro quanto mais podemos comprimir essas regiões na imagem.

No caso de compressão com perdas em vídeo (e em som) é comum especificarmos a taxa de compressão como uma função direta do quanto de informação pode ser transmitida ao longo do tempo, ou *Bit Rate*. Por exemplo, se um vídeo de 640x480 com 24 bpp a 30 quadros por segundo

precisaria de 210Mbits por segundo. Se notarmos que conexões Internet domésticas estão a 256Kbits/s nos melhores casos, que redes atuais operam a 100Mbits/s, vemos que a compressão desses dados é fundamental, mas que pode ser variada de acordo com o meio onde o dado trafega.

Inclusive essa é uma medida vital para sistemas de transmissão de vídeo e som sob demanda, *Streaming*. Tais como o *Real Vídeo/Audio* e *Windows Media Services*.

8.2. Formatos de Arquivo

Como a memória RAM do computador é volátil, precisamos de algum mecanismo de persistência para a imagem, ou seja, gravá-la em um arquivo. Os detalhes de como gravar imagens em arquivos não são universais e consequentemente existem vários padrões.

A tabela a seguir mostra os formatos de arquivo padronizados que já estão bem estabelecidos no mercado:

| Formato | Sistema de Cor | Compressão |
|---------|----------------------------|------------------------------|
| GIF | RGB 256 | LZW |
| TIFF | RGB, CMYK, YCbCr, Lab, Luv | RLE, LZW, JPEG, JBIG, Outros |
| JPEG | RGB, YCbCr, CMYK, Gray | JPEG |
| PCX | RGB | RLE |
| BMP | RGB | RLE |
| TGA | RGB | RLE |

Tabela 9: Formatos de Arquivos e Respectivos Sistemas de Cor e Compressão

Repare que o formato TIFF se destaca entre os demais. Atualmente ele é o formato mais recomendado para armazenar imagens que serão usadas em diversos programas, pois praticamente todos já o reconhecem e sua flexibilidade é enorme. Possue suporte para várias informações além da imagem, tais como dados de colorimetria para correta reprodução da mesma.

Dois outros formatos são importantes de serem mencionados, o GIF e o JPEG. Isto porque eles são utilizados como os padrões de imagem na Internet.

O formato GIF armazena apenas imagens RGB com 256 cores ou menos, sendo adequado para imagens de mais baixa qualidade e imagens tons de cinza. Esse formato permite que você defina uma cor que será ignorada pelo Browser, tornando parte da imagem que contém aquela cor,

transparente. Uma outra característica interessante é que as linhas no arquivo podem ser gravadas entrelaçadas, fazendo com que o Browser receba algumas linhas de toda a imagem antes de chegarem todas as linhas, permitindo um efeito de que toda a imagem está sendo recebida em níveis de qualidade crescentes. Esse formato possui duas versões 87a e 89a, somente a segunda que permite transparência. O GIF usa compressão LZW para comprimir a imagem. Essa compressão é muito semelhante a usada no formato de arquivo compactado ZIP e possui ótimos resultados. Além disso, esse formato suporta várias imagens num mesmo arquivo, formando opcionalmente uma animação.

O formato JPEG, usa a compressão JPEG que lhe deu o nome, e permite gravar imagens Tons de Cinza, RGB, CMYK e outros sistemas de cor, mas sempre comprimidas com perdas. Embora tenha perdas é o melhor formato para colocar imagens de alta qualidade. Para Internet o efeito da imagem sendo recebida com níveis de qualidade crescentes é feito através de codificação progressiva (*Progressive Encoding*). Não possui transparência.

Um outro formato que está sendo padronizado para a Internet é o PNG (Portable Network Graphics). Cuja compressão é sem perdas e permite imagens com mais alta qualidade que o GIF. Além de não usar a compressão LZW por causa de um problema de direitos autorais desta compressão usada pelo GIF. Assim, o PNG é o natural sucessor do GIF. Mas para o caso de várias imagens formando uma animação o formato a ser usado é o MNG, criado pelo mesmo grupo do PNG.

O formato TIFF também suporta várias imagens no mesmo arquivo, mas elas não estão associadas a necessariamente uma animação. Podem ser versões de menor resolução da imagem principal.

Um erro muito comum é tratar formatos de arquivos de imagem como tipos de imagem. Mas é preciso ficar bem claro que tipos de imagem não estão intrinsecamente associados à formatos arquivos, estes se prestam apenas ao armazenamento das mesmas. Portanto não faz sentido falar em “uma imagem TIFF” e “uma imagem GIF”, como imagens diferentes.

8.3. Arquivos de Vídeo

No caso de vídeo os formatos são um pouco mais complicados pois na grande maioria também incluem som. Além disso a compressão é com perda de acordo com a aplicação.

Assim como para som existe o CD e o MD, e para imagem existe o PhotoCD. Também existem vários tipos de mídia dedicados ao armazenamento de vídeo, tais como o DVD, VídeoCD e o mais antigo disco de Video Laser.

A tabela a seguir ilustra alguns formatos de vídeo e suas aplicações. A resolução no caso MPEG-2 pode variar dependendo da aplicação.

| Formato | Resolução | Data Rate | Aplicação |
|-------------|-----------|-------------|------------------|
| Motion JPEG | 720x486 | 0.5-25Mbps | General |
| MPEG1 | 352x240 | 150-500Kbps | VCD |
| MPEG2 | 720x486 | 0.1-18Mbps | DVD, SVCD |
| DV | 720x480 | 3.5 Mbps | Cameras Digitais |
| D1 | 720x486 | 25 Mbps | Broadcast |

Tabela 10: Compressão de Vídeo

Os formatos AVI e MOV permitem qualquer forma de compressão através do mecanismo de codecs externos. Ou seja, o formato apenas armazena os dados comprimidos que são repassados pelo codec. Uma situação muito comum é você receber um arquivo no formato AVI que o Media Player não consegue reproduzir pois o codec utilizado para a compressão não está disponível. Ao mesmo tempo em que facilita enormemente o trabalho do tocador, é preciso prestar atenção na disponibilidade dos codecs utilizados.

O formato MPEG foi feito apenas para a compressão MPEG. A compressão MPEG possui diversas evoluções. MPEG-1 em 1992, inclui a definição da compressão dos arquivos de audio MP3, que se refere a MPEG-1 Layer 3. MPEG-2 em 1994, usado em DVD e proposto para Televisão Digital. MPEG-4 em 1998, usado para altas taxas de compressão, a Microsoft lançou a sua própria versão no Windows Media Format 8, assim como a DivX e a Apple com o QuickTime 6.

8.4. *Imagens Grandes e Multiresolução*

Mesmo com os métodos de compressão, para processarmos as imagens, precisamos descomprimi-las para a memória do computador. Quando as imagens são muito grandes comparadas ao tamanho da memória RAM disponível, temos que utilizar de outros recursos para poder processar tal imagem.

O primeiro deles é memória virtual, ou seja, usar disco como se fosse memória RAM. Como não precisamos de toda imagem ao mesmo tempo, deixamos uma parte dela no disco, quando esta parte é requisitada, carregamos a mesma para a memória e descarregamos para o disco partes que não estão sendo utilizadas.

Um outro recurso é utilizar o conceito de multiresolução. O que o próprio nome diz é que teremos a mesma imagem em diferentes níveis de resolução. Dependendo da resolução que o usuário estiver trabalhando, somente um nível é afetado. Quando o usuário muda de nível, as mudanças realizadas naquele nível são propagadas para o novo nível. Somente quando o usuário chega no último nível ou pede para salvar as alterações em disco, é que a propagação é feita para todos os níveis.

Esta imagem mostra os níveis de uma imagem em multiresolução.



Figura 44: Uma Imagem em Multiresolução

Quando possuímos muitas imagens grandes é comum guardarmos uma versão reduzida para consulta rápida ao conteúdo do arquivo, esta versão reduzida é chamada de *thumbnail*.

9. Processamento

As funções de processamento de imagens digitais podem ser classificadas em duas classes, por escopo ou por resultado. Por escopo, está relacionado com técnicas de processamento. Por resultado, está relacionado com os tipos de resultado de cada técnica.

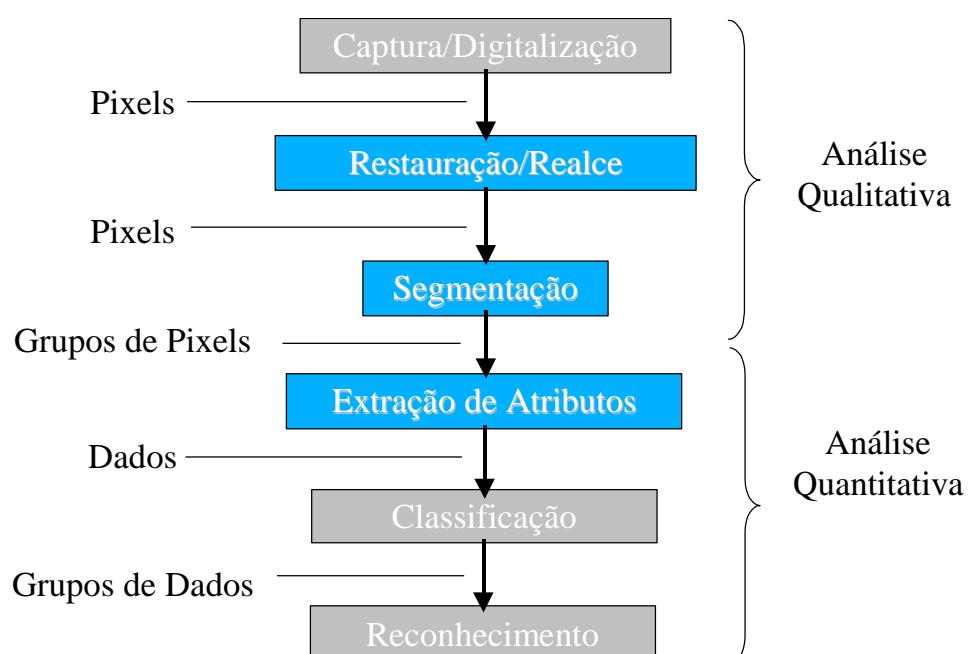


Figura 45: Seqüência de Processamento de Imagens

Através do diagrama mostrando uma seqüência tradicional para processamento da imagem, da captura à classificação, podemos ter uma idéia melhor das classes de processamento por resultado.

Há um fator comum em todas as classes de processamento, este fator é qualidade. Existem duas subdivisões em qualidade de imagem: fidelidade e inteligibilidade. No primeiro caso estamos preocupados em aproximar a imagem processada da imagem original ou de um padrão estipulado que a melhor represente. No segundo caso, nos preocupamos com a informação que conseguimos extrair da imagem, seja pelo olho humano, seja por algum processamento.

Programas para editoração eletrônica precisam ter um cuidado muito grande com a fidelidade na representação da cor da imagem, já programas de processamento de imagens científicas em geral não estão preocupados com a fidelidade da cor e sim a informação contida na imagem.

9.1. Restauração/Realce

Quando estamos no contexto de editoração eletrônica, ficamos normalmente situados apenas no escopo de operações de Realce ou Obscurecimento, e Restauração ou Deformação.

A restauração busca compensar distorções específicas, normalmente geradas no momento de aquisição. Quando se pode identificar experimentalmente a função que representa a deformação ou construir um modelo matemático adequado, é possível buscar a função inversa e aplicá-la sobre a imagem deformada. Por exemplo: Correção de foco, Imagens borradass por movimento. Em todos os casos, a formulação matemática envolvida é extremamente complexa e o custo computacional muito alto. Os resultados podem ter consequências somente qualitativas.

9.2. Segmentação

Quando estamos no contexto de processamento de imagens científicas, o mais comum é queremos obter dados relacionados com os objetos presentes na imagem. Então são necessárias as operações de Segmentação que procurarão isolar regiões de pixels, e operações de Extração de Atributos que vão olhar para essas regiões e calcular uma série de parâmetros que as descreverão.

A operação de segmentação mais comum é a limiarização por um tom de corte. Tudo que está acima deste tom vira branco, tudo que está abaixo vira preto, obtendo-se uma imagem binária.

A partir desse momento as operações Morfológicas são excepcionalmente úteis, veja em Operações Locais.

9.3. Extração de Atributos

Utilizando imagens binárias é muito simples obter dados relevantes, ou atributos, das regiões segmentadas, tais como:

Número total de objetos.

Propriedades geométricas, do tipo: Área, Perímetro, Centro de gravidade, Largura máxima e mínima

Atributos relacionados à forma, tais como: Circularidade, Concavidade

Propriedades de luminância, tais como: Nível de cinza médio de cada região (1o momento da distribuição), Desvio padrão do nível de cinza (2o momento da distribuição), outros momentos estatísticos da distribuição de cada região (skewness e kurtosis)

Propriedades associadas à textura

As propriedades geométricas são medidas diretamente sobre a imagem binária. As de luminância são obtidas da imagem original, a partir da informação de posição dos diversos objetos, fornecida pela imagem binária.

9.4. Classificação

Uma vez com esses parâmetros coletados, queremos distinguir objetos na imagem agrupando esses parâmetros de acordo com sua semelhança para cada região de pixels encontrada. Feita essa classificação desses parâmetros os objetos estão reconhecidos e podemos agora tomar decisões e relatar fatos relacionados com os objetos do mundo real, ponderando sempre através uma medida de erro da classificação.

Este processo é muito complexo e existem diversos níveis de automação. Os mais simples implicam em processos de agrupamento estatístico, para os quais a decisão humana é fundamental. Os mais sofisticados permitem ao computador reconhecer diferentes objetos através de técnicas de inteligência artificial com pouca ou nenhuma intervenção humana. Os processos que possuem intervenção são chamados de supervisionados.

Aqui, a palavra classificação não denota nenhum juízo de valor, mas apenas o agrupamento em classes dos diversos objetos obtidos na segmentação, cujos atributos já foram medidos.

Em geral, vários atributos são necessários para uma correta classificação. Mas, quanto mais atributos, mais complexo se torna o problema. Desta forma, é muito importante realizar uma seleção adequada dos atributos disponíveis, visando otimizar o processo.

9.5. Operações Pontuais

Operações pontuais são operações em que um pixel da imagem resultante depende apenas do mesmo pixel na imagem original. Neste caso a única informação que nós temos é a cor do pixel, por isso muitas das operações pontuais são operações que alteram características de cor e luminância, tais como: brilho, contraste, nível de branco e nível de preto, saturação, correção gamma, limiar (*threshold*), posterização, solarização, negativo, etc.

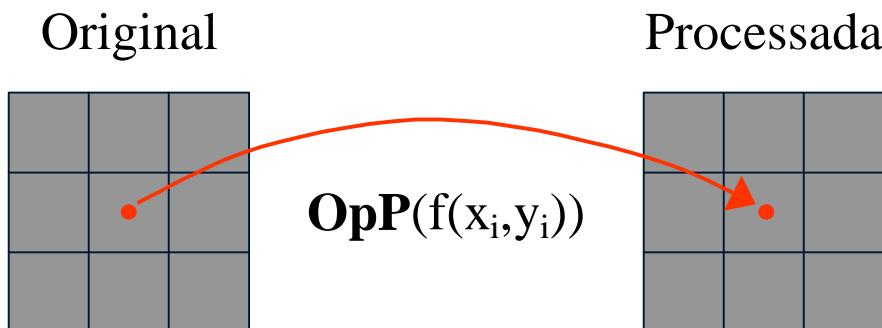


Figura 46: Operação Pontual

Qualquer operação pontual pode ser visualizada como um mapeamento de pixels da imagem original para a imagem processada. E isso é facilmente representado como um gráfico que relaciona as tonalidades disponíveis na imagem original com as tonalidades disponíveis na imagem processada. Inclusive algumas operações bastante conhecidas são visualizadas nesse gráfico como funções muito simples. A operação pontual de identidade é simplesmente uma reta que mapeia todos os tons da imagem original nos mesmos tons na imagem processada. Transladando essa reta estamos simplesmente alterando o brilho na imagem. Alterando sua angulação estamos alterando o contraste, mas existem outras formas de se alterar o contraste como por exemplo usando uma função logarítmica que faria com que tons escuros fosse mapeados em um número maior de tons, realçando esses tons escuros, ou o inverso, uma função exponencial onde tons claros seriam realçados.

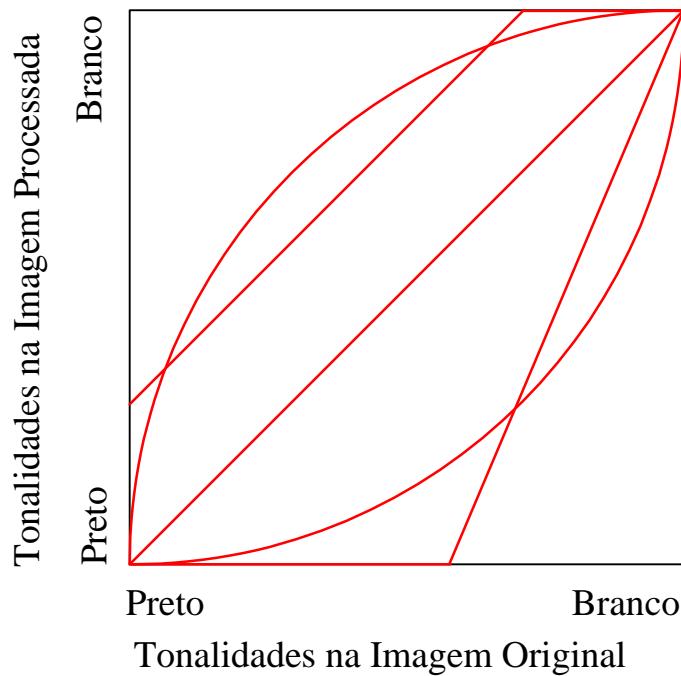


Figura 47: Função de mapeamento de Tonalidades

Até agora estávamos falando de operações pontuais que envolvem apenas uma imagem original. Estas são chamadas de unárias. Mas podemos definir operações que usam mais de uma imagem original, são chamadas de binárias, ternárias, etc. Por exemplo, uma operação de média de várias imagens capturadas na mesma posição, é uma técnica comum de eliminação de ruído. Mas as operações binárias mais comuns são as operações aritméticas de soma, subtração, divisão, etc de duas ou mais imagens.

Embora as funções pontuais estejam restritas somente ao pixel sendo processado, o processamento em si pode levar em consideração dados globais da imagem, como por exemplo o histograma.

O histograma é uma função estatística da imagem que para cada nível de tonalidade, calcula quantos pixels existem naquela tonalidade. Muitas operações pontuais usam o histograma como parâmetro de decisão para fornecer resultados diferentes para o pixel da imagem processada.

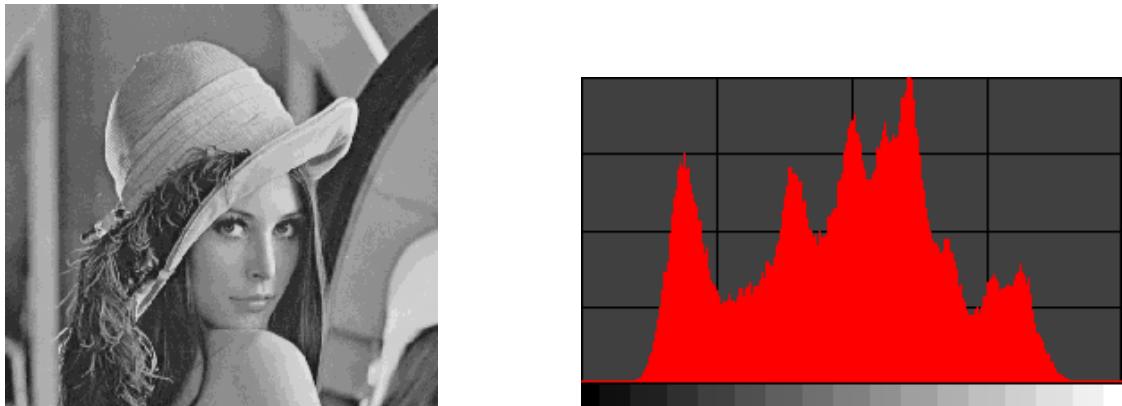


Figura 48: Histograma de uma Imagem Monocromática.

9.6. Operações Locais

Operações pontuais são operações em que um pixel da imagem resultante depende de uma vizinhança do mesmo pixel na imagem original. As operações locais mais comuns são as operações de filtragem que usam uma convolução com um kernel de dimensão $n \times n$, onde n é tipicamente 3, 5, 7 e 9, mas pode assumir qualquer valor. Por exemplo, no caso de $n=3$ é muito fácil de entender. Cada pixel da vizinhança 3×3 do pixel na imagem original é multiplicado por um valor e todos esses valores são somados resultando no pixel novo. Os valores que ponderamos a vizinhança são armazenados em uma matriz chamada, neste caso, de kernel de convolução. Operações de Blur, Sharpen, Detecção de Bordas e muitas outras são implementadas assim.

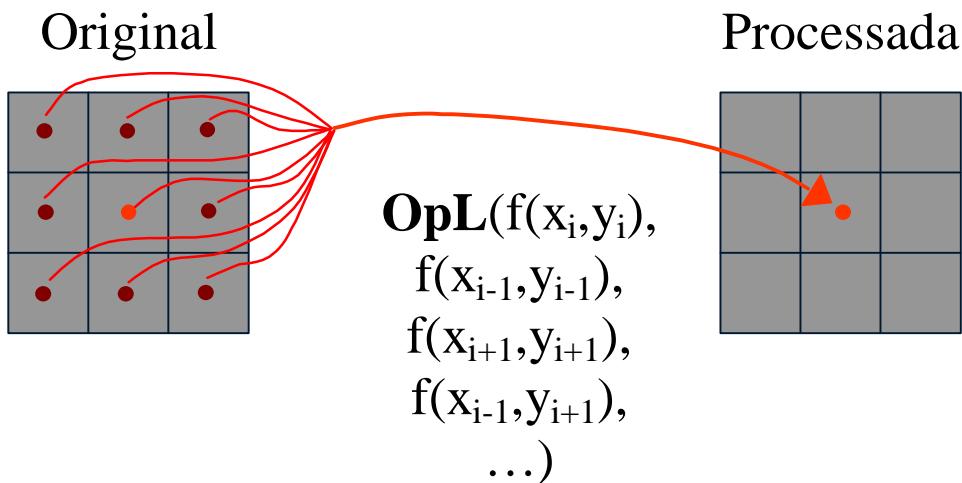


Figura 49: Operação Local

As operações que realizam deformações no grid de pixels utilizam, como vimos, a técnica de interpolação para encontrar o novo ponto. A técnica de interpolação não deixa de ser uma operação local que depende apenas de uma vizinhança, embora algumas técnicas de interpolação muito sofisticadas e em geral muito lentas usam uma vizinhança bem extensa. Dentre as técnicas que usam interpolação podemos citar: Warping, Morphing, Resize, Stretch, Rotação, etc.

Um outro conjunto de operações que também podem ser consideradas como locais são as operações Morfológicas. São muito utilizadas por sistemas de processamento de imagens científicas para a etapa de segmentação, como já mencionado. Embora muito semelhantes a convolução com um kernel, a forma com que os vizinhos são computados é ligeiramente diferente, pois usam operações lógicas, tais como AND e OR.

9.7. Operações Globais

Operações globais são operações em que um pixel da imagem resultante depende de um processamento realizado em todos os pixels da imagem original. Neste grupo de operações estão as transformadas de domínio, tais como a Transformada de Fourier, a Transformada de Wavelets (que na realidade são muitas, dependem de que wavelet está sendo usada) e a Transformada de Hough.

Todas essas transformadas nos dão informações muito interessantes sobre a imagem original. A Transformada de Fourier, por exemplo, é base fundamental para toda teoria de processamento de sinais e com ela pode-se realizar uma série de operações muito importantes com imagens. A Transformada de Wavelets irá decompor a imagem em uma estrutura de multiresolução, que é muito utilizada para compressão de imagens. A Transformada de Hough procura identificar formas geométricas na imagem, tais como retas e círculos.

Esse grupo de operações é visto somente em programas de processamento de imagens científico.

10. Software

Existem diversos sistemas de processamento de imagens disponíveis no mercado. Eles se distinguem pela forma de distribuição (Comercial, *Shareware*, *Freeware*), pela área de aplicação (editoração eletrônica, análise científica, visualização, edição informal, pesquisa, ...), e pelas plataformas em que estão disponíveis (UNIX, Windows, MAC, ...).

O programa mais bem sucedido, não importando a categoria, é o Adobe Photoshop. Atualmente na versão 5.0, é um dos mais robustos e confiáveis sistemas de processamento de imagens. É voltado para editoração eletrônica e não possui funções de análise de imagens. Possui versões para MAC, Windows e UNIX.

Seguindo essa mesma linha temos o Corel PhotoPaint e o Fractal Design Painter, mas estes só rodam no ambiente Windows. O Fractal Design Painter possui mecanismos muito peculiares de pintura, além de permitir a criação de animações usando seqüências de imagens.

Outros programas na mesma linha, mas que ganharam popularidade por serem *Shareware/Freeware* e não possuírem a complexidade dos sistemas acima, são o Paint Shop Pro e o LviewPro. Um programa semelhante a esses, mas que realiza apenas a visualização é o ACDSee, se tornou popular pela facilidade de fazer *Slide Shows*. Todos esses são somente para Windows.

No UNIX temos o GIMP e o XV, este último muito divulgado na *Internet*. Ambos são *Freeware*.

Quando se trata de análise de imagem e processamento científico, muitos dos programas mencionados são úteis, mas outros assumem um nicho não atacado por estes.

Na linha *Freeware* temos o NIH Image (MAC, Windows e Java) e o Image Tool (Windows). No UNIX temos ainda dois grandes sistemas o Khorus e o XITE. Sendo que o Khorus abrange muito mais do que processamento de imagens.

Na linha comercial temos o KS-400, o Image Pro, o Optimas, e o WiT (todos apenas em Windows). O WiT tem uma característica muito interessante, a de permitir a criação de macros usando uma linguagem visual.

Adobe PhotoShop

Como este é muito difundido, convém comentarmos algumas funções do mesmo associando-as aos conceitos aqui apresentados. Vamos estar comentando funções da versão “4.01”.

No menu **Image** temos as funções **Image Size** e **Canvas Size**, ambas relativas a resolução espacial da imagem.

Na função **Image Size** quando **Resample Image** está marcado pode-se mudar o tamanho em *pixels* da imagem, quando está desmarcado pode-se mudar apenas o tamanho real da imagem sem mudar o número de *pixels*. Junto a este controle você também pode escolher o método de interpolação, quando o número de *pixels* for mudar. Este diálogo é bastante versátil permitindo manter a razão de aspectos da imagem original, e permite também mudar o tamanho em *pixels* baseado no tamanho real e na resolução.

A função **Canvas Size** irá mudar o número de *pixels* da imagem, mas apenas acrescentará *pixels* acima, abaixo ou dos lados de acordo com a posição da imagem original em **Anchor**.

Tome cuidado ao mudar o tamanho da imagem, pois ao reduzi-la estará sempre perdendo informação, e ao aumentá-la corre o risco de ocupar muita memória.

Alguns procedimentos relacionados com calibração dos sistemas de cor utilizados são feitos no submenu **Color Settings** do menu **File**.

Embora suporte 16 bits por componente, além dos tradicionais 8, praticamente nenhuma função fica habilitada quando muda-se para 16 bits. Ainda não entendi o objetivo do suporte a 16 bits já que quase nada pode ser feito.

No submenu **Mode** do menu **Image** você normalmente irá, e deve, usar o sistema de cor RGB, junto com suas variações: **Gray Scale** e **Indexed Color**. Em geral é mais conveniente processar a imagem nesses modos e depois converte-la para algum outro, se necessário. O modo **Bitmap** serve para converter a imagem para apenas 1 bit por pixel.

Quando se está trabalhando no modo RGB pode-se visualizar rapidamente como seria sua imagem em CMYK selecionando a opção **CMYK Preview** do menu **View**.

O sistema CMYK é usado para criar uma separação de cores para a impressão. Também se pode usá-lo para trabalhar diretamente sobre imagens CMYK provindas de determinados dispositivos.

O sistema Lab é usado quando se quer trabalhar com imagens de PhotoCD ou quando se quer editar a componente de luminância independentemente dos valores de cromaticidade. O sistema Lab também é usado internamente para converter de um sistema para outro e é o recomendado para o transporte de uma aplicação para outra.

Também tenha cuidado ao utilizar o submenu de **Mode**, pois durante as conversões de um sistema para outro podem ocorrer perdas na imagem. Exceto quando convertendo para **Multichannel** que apenas separa cada uma das componentes para serem trabalhadas independentemente.

Quando a imagem está em tons de cinza, pode-se trabalhar com o modo de **Duotone**, que inclui *monotone*, *duotones*, *tritones* e *quadtones*. Nesses modos a escala de cinzas é substituída por uma escala em um tom, de uma, duas, três ou quatro cores, respectivamente. Este modo é útil para a impressão da imagem quando a tinta preta é substituída por uma outra cor (*monotone*) ou quando usa-se uma segunda cor (*duotone*), por exemplo um cinza, para aumentar a escala tonal da imagem impressa.

Todas as funções do submenu **Adjust** do menu **Image** são operações pontuais e dependem somente da cor do próprio pixel na imagem original. A função **Calculations** do menu **Image** são operações pontuais binárias ou ternárias por precisam de duas ou mais imagens. Quando você junta dois ou mais **Layers** em um único **Layer**, está realizando também uma operação pontual. Já no caso do menu **Filter** a grande maioria das funções são operações locais.

11. Aplicações

Qualquer área onde se possa capturar uma imagem e dela obter resultados é uma área onde pode ser usado um sistema de processamento de imagens. Devido a esta vasta amplitude de áreas de aplicação, sistemas de processamento de imagens são, à priori, interdisciplinares. Seguem alguns exemplos:

- Sensoriamento Remoto

Centenas de fotografias, enviadas por naves espaciais e satélites em torno da terra, precisam ser avaliadas e catalogadas com fidelidade e eficiência. Fotos de países podem ser usadas para a obtenção rápida de mapas cartográficos e estudos geográficos, tais como análise do solo e da safra agrícola.

- Microscopia.

Análise de imagens provindas de microscópios óticos ou eletrônicos, em áreas que variam desde a medicina até a metalurgia, identificando e classificando partículas.

- Medicina.

Imagens radiográficas de baixa qualidade podem ser realçadas para a obtenção de um diagnóstico mais preciso. Com a introdução da tomografia computadorizada e da ressonância

magnética, imagens volumétricas são utilizadas para diversas aplicações, tais como planejamento cirúrgico e diagnóstico.

- Manutenção de obras de arte.

Evitar a degradação com o tempo de obras de arte não é um processo simples, mas possuindo uma imagem da obra original que pode ser processada para ajudar no processo de restauração para se obter um retrato mais fiel o possível. Além disso, existem estudos para a verificação da autenticidade de pinturas através da análise das pinceladas de cada pintor.

- Identificação de impressões digitais.

Através de adequada classificação e armazenamento pode-se identificar uma impressão digital usando um banco de dados associado a um banco de imagens.

- Armazenamento de documentos.

É comum que grandes quantidades de informação fiquem "encalhadas" devido à dificuldade de acesso a quilos de papel que ficam engavetados e deteriorando. Estes textos podem ser transferidos para um sistema de hipertexto que permita uma consulta rápida e dinâmica. Para tanto, é necessário capturar as imagens dos textos e depois realizar um reconhecimento de caracteres, segue então um processo de integração ao hipertexto. Muitas vezes estes textos são documentos oficiais que precisam manter suas características, portanto as próprias imagens são armazenadas.

- Medidas de velocidade de escoamento de fluidos.

Através de partículas suspensas no líquido, pode-se iluminar um corte no escoamento através de um plano de luz, e utilizando uma exposição lenta obtém-se uma fotografia com traços correspondentes às trajetórias das diversas partículas. A partir dos comprimentos dos traços pode-se obter uma medida aproximada da velocidade do fluido.

- Controle de qualidade.

A aplicação de processamento de imagens na área de controle de qualidade é muito vasta. Praticamente qualquer processo industrial que necessite de alguma monitoração ótica ou visual pode ser automatizado com vantagens. O que pode impedir é o grau de dificuldade de interpretar cada imagem, assim como em qualquer outra aplicação de análise de imagens.

Uma aplicação comum é a verificação de placas de circuito impresso, que possuem uma geometria simples e regras fixas.

- Outros: Astronomia, Fotografia, Vídeo, Efeitos Especiais, etc.

12. Internet Links

Cor

Poynton's Colour FAQ - <http://home.inforamp.net/~poynton>

LittleCms - <http://www.littlecms.com/>

Adobe Color and Color Management / Technical Guides -

<http://www.adobe.com/support/techguides/color/main.html>

CIE / International Commission on Illumination -

<http://www.hike.te.chiba-u.ac.jp/ikeda/CIE/home.html>

A Standard Default Color Space for the Internet (sRGB) -

<http://www.w3.org/Graphics/Color/sRGB.html>

efg's Computer Lab and Reference Library - <http://www.efg2.com>

Captura

Twain - <http://www.twain.org>

Windows Image Acquisition Still Image and Video Capture

<http://www.microsoft.com/hwdev/tech/WIA/default.asp>

ScanTips - <http://www.scantips.com>

Armazenamento

LibTIFF - <http://www.libtiff.org>

LibPNG - <http://www.libpng.org>

LibJPEG - <http://www.jpeg.org>

FreeImage - <http://www.freeimage.org>

ImageMagick - <http://wwwimagemagick.org>

QuickTime – <http://www.apple.com/quicktime>

Windows Media – <http://www.microsoft.com/windowsmedia>

MPEG – <http://www.mpeg.org>

Sistemas

Xite - <http://www.ifi.uio.no/~blab/Software/Xite>

GIMP - <http://www.gimp.org>

XV - <http://www.trilon.com/xv>

NIH Image - <http://rsb.info.nih.gov/nih-image>

ImageTool - <http://ddsdx.uthscsa.edu/dig/itdesc.html>

Referências

Kodak Digital Learning Center - <http://www.kodak.com/US/en/digital/dlc/index.jhtml>

HyperGraph - <http://www.siggraph.org/education/materials/HyperGraph/hypergraph.htm>

ShortCourses - <http://www.shortcourses.com>

Scan Tips - <http://www.scantips.com>

Eletronic Photo Imaging - <http://www.epi-centre.com>

Brian Lawler Essays - <http://www.thelawlers.com>

13. Bibliografia

- [BAXES 94] G. A. Baxes; “Digital Image Processing (Principles and Applications)”; John Wiley & Sons, Inc.; 1994.
- [CASTLEMAN 95] K. R. Castleman; “Digital Image Processing”; Prentice-Hall, Inc.; 1995.
- [GOMES 97] J. Gomes e L. Velho; “Image Processing for Computer Graphics”; Springer; 1997.
- [FOLEY 96] J. Foley, A. van Dam, S. Feiner, J. Hughes; “Computer Graphics: Principles and Practice”; Addison-Wesley; 1996.
- [GONZALEZ 92] R. C. Gonzalez, R. Woods; “Digital Image Processing”; Addison-Wesley; 1992.
- [POYNTON 96] C. A. Poynton; “A Technical Introduction to Digital Video”; John Wiley & Sons, Inc.; 1996.