

Computação Gráfica



Prof. Rodrigo Martins

rodrigo.martins@francomontoro.com.br

Alguns slides foram cedidos pelo Prof. Jorge Cavalcanti da UNIVASF
(UNIVERSIDADE FEDERAL DO VALE DO SÃO FRANCISCO)

Dispositivos Gráficos

- Toda imagem criada através de recursos computacionais deve ser representada em algum dispositivo físico que permita a sua visualização.
- Diversas tecnologias e diferentes tipos de dispositivos são utilizados para gerar representações visuais, sendo que o desenvolvimento dessas tecnologias teve um papel fundamental na evolução da CG.
- Tanto para o usuário como para o implementador de sistemas gráficos é importante conhecer as características de cada uma dessas tecnologias para sua melhor utilização.
- Vamos discutir alguns aspectos da arquitetura e organização dos tipos mais comuns dos dispositivos de exibição gráfica, sem entrar em detalhes técnicos.

Dispositivos Gráficos

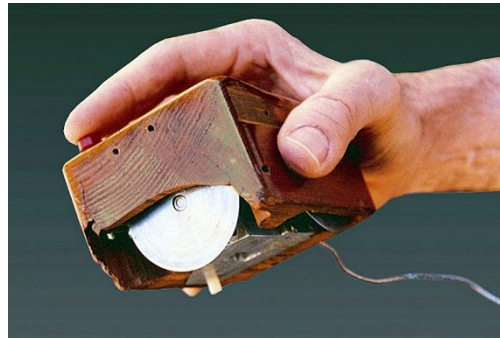
- É possível classificar os dispositivos de exibição (traçadores, impressoras e terminais de vídeo) em duas principais categorias, segundo a forma pela qual as imagens são geradas:
dispositivos vetoriais e dispositivos matriciais.
 - Os dispositivos gráficos vetoriais conseguem traçar segmentos de reta perfeitos entre dois pontos da malha finita de pontos definida por suas superfícies de exibição.
 - Os dispositivos matriciais, por outro lado, apenas conseguem traçar pontos, também em uma malha finita. Assim, segmentos de reta são traçados como sequências de pontos próximos.
- Através dos dispositivos gráficos interagimos com o sistema na busca de uma extensão dos limites do nosso corpo e uma melhor comunicação com a máquina.
- Os dispositivos podem ser classificados em:
 - **Dispositivos gráficos de entrada**
 - **Dispositivos gráficos de saída**

Dispositivos Gráficos de entrada

- Os dispositivos de entrada são componentes eletrônicos que permitem a movimentação e interação com os sistemas.
- A cada dia surge um novo dispositivo com novas propostas ergonômicas, recursos adicionais que agilizam a tarefa de interação ou simplesmente reduzem a quantidade de fios em sua mesa.

Dispositivos gráficos de entrada

- Teclado
- Mouse
- Joystick



Dispositivos gráficos de entrada

- Mesa Digitalizadora

- Consiste em uma mesa e um apontador.
- Ao tocar a mesa o computador recebe as coordenadas do referido ponto.
- Resolução muito fina, medições muito precisas.
- Popular no leste da Ásia como dispositivo para entrada de caracteres.



Dispositivos gráficos de entrada

- Luvas

- Através de sensores detectam e medem as flexões e pressões dos dedos;



Dispositivos gráficos de entrada

- Scanners Tridimensionais
 - Utilizam câmeras digitais acopladas a uma mesa especial que fornece as coordenadas para os sistemas;



Dispositivos gráficos de entrada

- Roupa de RV
 - Permite a interação com o mundo virtual;
 - Usadas para gerar informações de movimento humano e transmitir sensações.

Prodigy
52 Channels



Pioneer
16 Channels



Dispositivos gráficos de saída

- É possível classificar os dispositivos de saída em duas principais categorias, segundo a forma pela qual as imagens são geradas (veja seção anterior de descrição vetorial e matricial de imagens): vetoriais e matriciais.
- Os dispositivos vetoriais conseguem traçar segmentos de reta perfeitos entre dois pontos.
- Os dispositivos matriciais apenas conseguem traçar pontos, ou seja, segmentos de reta são traçados como sequências de pontos próximos, são entretanto, bastante adequados para desenhar áreas cheias e sombras, onde os vetoriais mostram deficiência.

Dispositivos gráficos de saída

- Impressoras



Jato de tinta



Laser



Térmicas

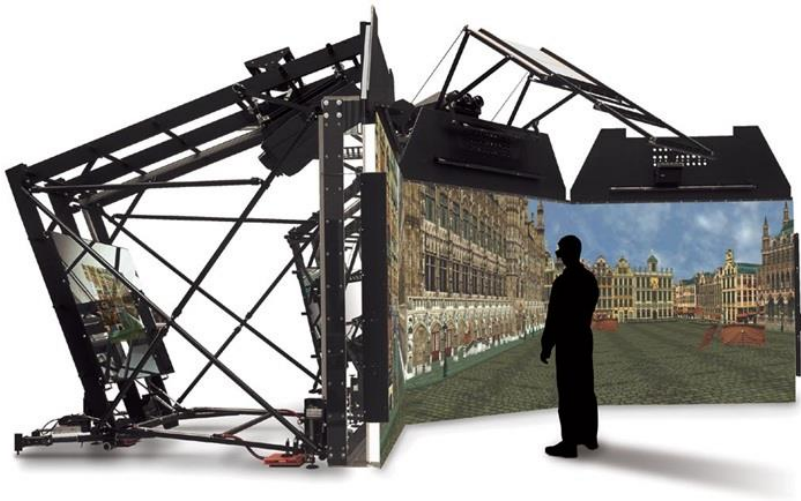
Dispositivos gráficos de saída

- Plotters
 - Produzem desenhos em grandes dimensões, com elevada qualidade.



Dispositivos gráficos saída

- Cave
 - Usam a idéia de colocar o usuário em uma sala com paredes que são na verdade telas para projeção de imagens.



Dispositivos gráficos de saída

- Display de retina – Stereo Glasses



COURTESY: GOOGLE

Dispositivos gráficos de saída

- Head Mounted Displays
 - Conhecidos como “óculos de realidade virtual” ou “capacete de realidade virtual”;
 - Operam exibindo duas telas de imagens de uma cena virtual.



Dispositivos gráficos de saída - Monitores

CRT



LCD



PLASMA



LED



Resolução de Imagens

- A resolução está associada a quantidade e a qualidade de informação que um dispositivo apresenta.
 - A resolução pode ser medida em pixels (*pictures elements*), no caso de monitores ou em DPI (impressoras).
 - O pixel representa uma unidade que pode ser controlada individualmente e que contém informações sobre cores e brilho.
 - O tamanho do pixel vai depender de como a resolução da tela foi configurada.
 - Relação entre o tamanho da tela e a resolução mínima recomendada:

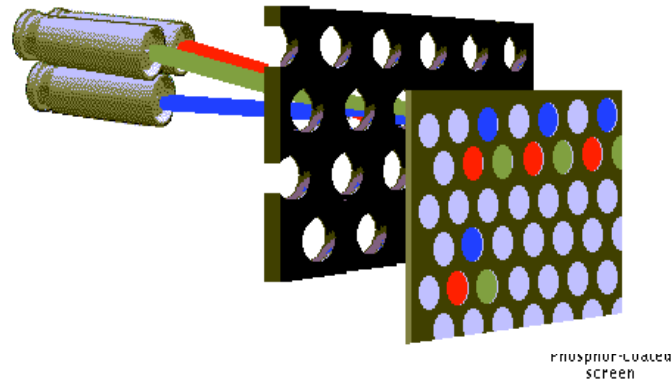
Medida Nominal	Resolução	Medida Nominal	Resolução
15"	800 x 600	19"	1.280 x 1.024
17"	1.024 x 768	21"	1.600 x 1.200

Resolução de Imagens

- Relação pixels X bits

- Cada pixel requer uma quantidade de bits, de acordo com o seu modelo de cores.

- Ex: **Sistema RGB**



Cada cor primária - 256 níveis (8 bits)

Cada pixel – 3 cores (RGB) X 8 bits = 24 bits (true color) => 16 M de cores

Cores de 32 bits => 8 bits para o canal alfa, com 256 níveis de transparência

Cores Preto & Branco – 1 bit (P.Ex: 0 Branco, 1 Preto)

Resolução de imagens

dpi (pontos por polegada)

- Uma medida da resolução da impressora em pontos por polegada. Impressoras a laser/ink jet típicas de mesa imprimem em 300 dpi.
- Fotocompositoras imprimem em 1270 ou 2540 dpi. Impressoras com capacidades maiores de dpi produzem uma saída mais suave e mais limpa.
- O termo dpi também é usado para medir resolução de digitalização e para indicar resolução de bitmap.
- Imagens para visualização em monitores não precisam ter mais que 100 dpi.
- Imagens a serem impressas, o mínimo recomendado para uma boa resolução são 300 dpi.

Resolução de imagens

- O número de pixels por polegada de uma imagem em formato bitmap, é medida em ppi (pixels por polegada) ou dpi (pontos por polegada).
- Resoluções baixas podem resultar em aparência granulada na imagem em formato bitmap, e resoluções altas podem produzir imagens mais suaves, mas resultam em arquivos maiores.



Resolução Gráfica

- Virtualmente todos os dispositivos de I/O gráficos usam uma malha retangular de posições endereçáveis - a qual é denominada “retângulo de visualização”.
- A “resolução gráfica” de um dispositivo é o número de posições (ou pontos, ou pixels) horizontais e verticais que ele pode distinguir. Existem 4 parâmetros que definem a resolução.

1. **ndh** - número de posições endereçáveis horizontalmente.
2. **ndv** - número de posições endereçáveis verticalmente.
3. **width** - a largura do retângulo de visualização em mm.
4. **height** - a altura do retângulo de visualização em mm.

- A partir desses 4 parâmetros, vários números interessantes podem ser calculados.

Resolução Gráfica

1. resolução horizontal:
2. tamanho ponto horizontal:
3. resolução vertical:
4. tamanho ponto vertical:
5. total pontos endereçáveis:
6. resolução de área:
7. razão de aspecto (gráfica):
8. razão de aspecto (física):

$$horiz_res = \frac{ndh}{width}$$

$$horiz_dot_size = \frac{width}{ndh}$$

$$vert_res = \frac{ndv}{height}$$

$$vert_dot_size = \frac{height}{ndv}$$

$$total_nr_dots = ndh.ndv$$

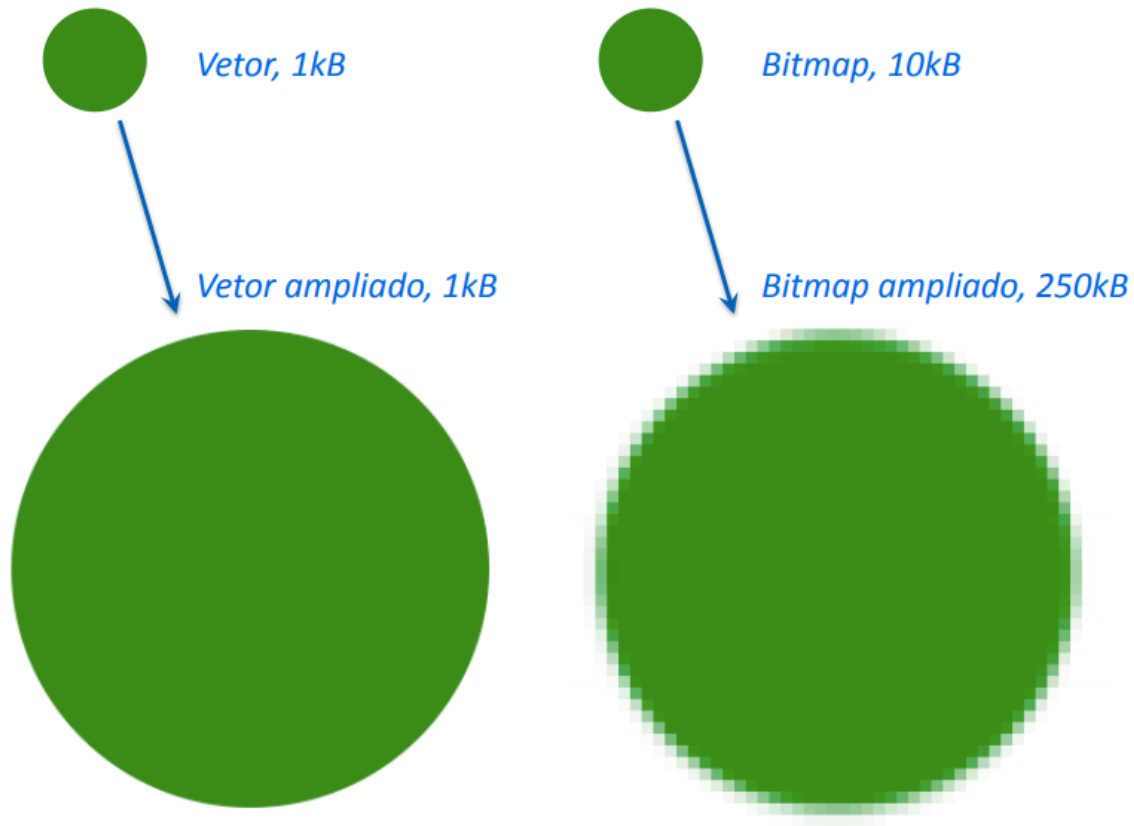
$$area_res = \frac{total_nr_dots}{(width.height)}$$

$$aspect_ratio = \frac{vert_dot_size}{horiz_dot_size}$$

$$physical_aspect_ratio = \frac{height}{width}$$

Tipos de Imagens

- As imagens podem ser classificadas como **Vetorial** e **Matricial**;



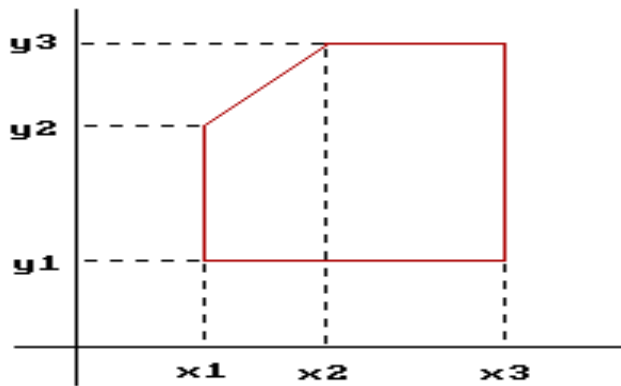
Tipos de Imagens

- A representação vetorial é empregada para a definição e modelagens de objetos que serão representados pela imagem;
- Na representação vetorial são usados como elementos básicos os pontos, as linhas, as curvas, etc.;
- Esses elementos básicos são chamados primitivas gráficas;
- Cada primitiva gráfica possui um conjunto de atributos que define sua aparência e um conjunto de dados que define sua geometria.



Imagens Vetoriais

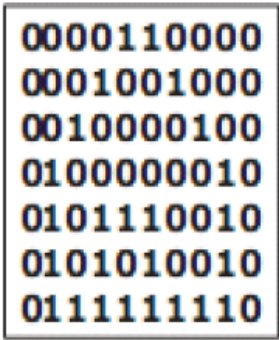
- Vantagens das imagens vetoriais:
 - Facilidade de armazenamento dos elementos geométricos;
 - Facilidade de manipulação (escala, rotação, etc.);
 - Alteração simples;
- Desvantagem das imagens vetoriais
 - Requer dispositivos de saída específicos para ter bons resultados;
 - Reconstrução mais lenta.



Imagens Vetoriais

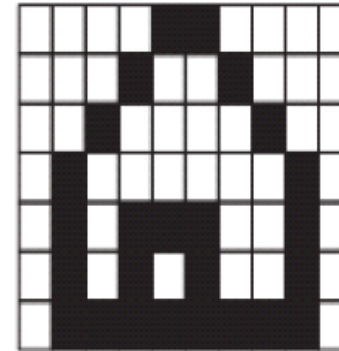
- Ilustração vetorial com e sem preenchimento de cor.





Memória de Imagem

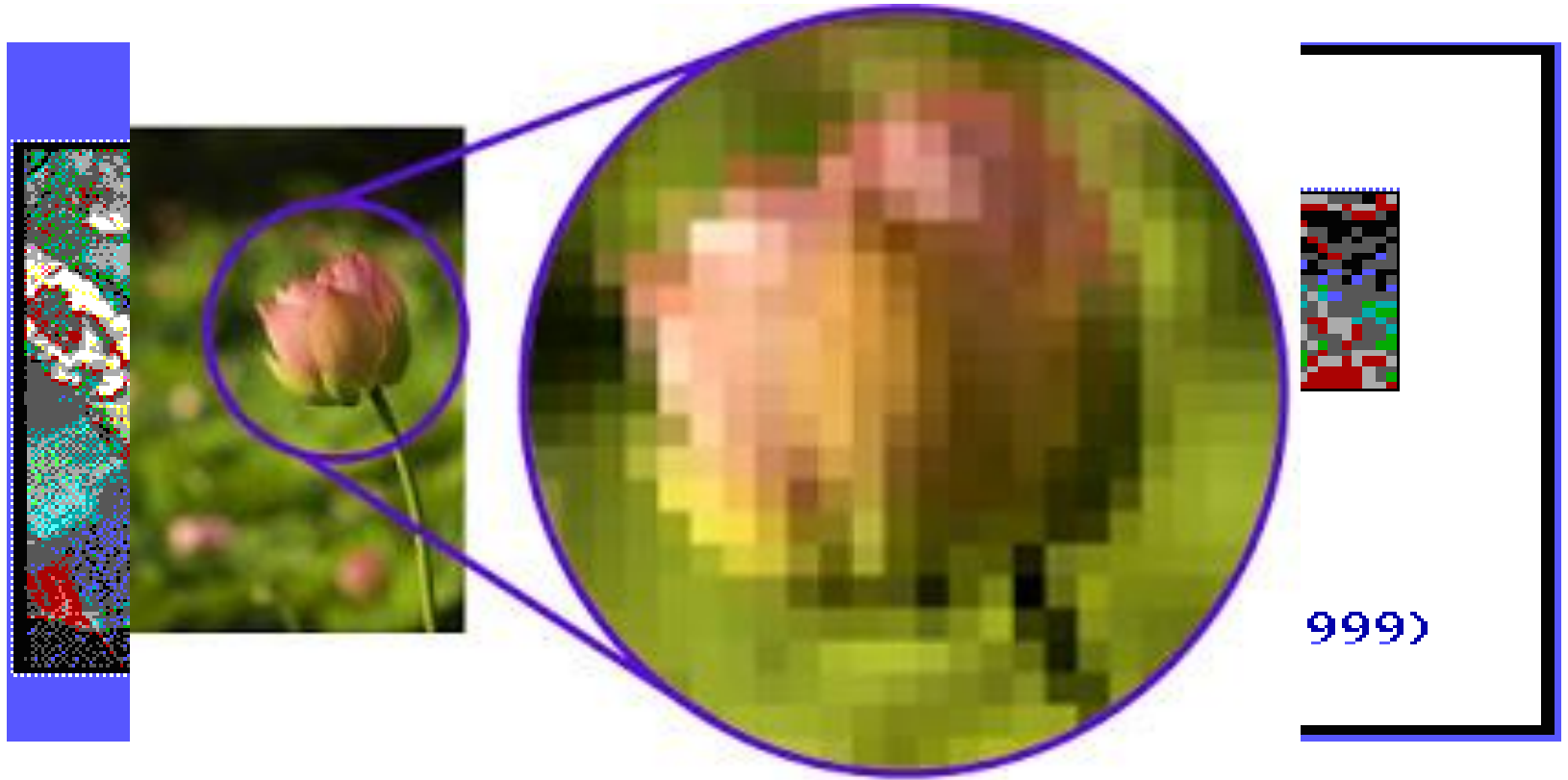
CONTROLADOR DE VÍDEO



Monitor

Imagens Matriciais

- Bitmaps ou Mapa de Bits



Imagens Matriciais

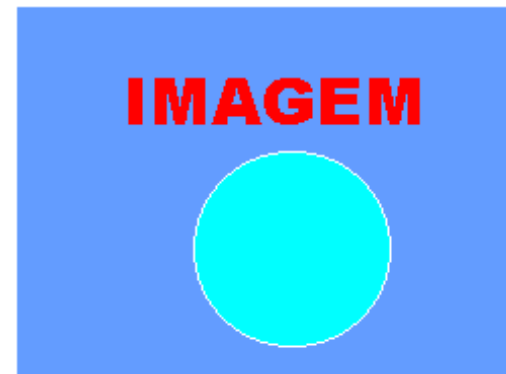
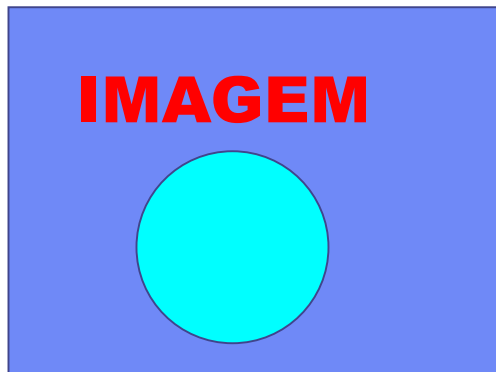
- Vantagens das imagens matriciais
 - Fácil tradução para dispositivos baseados em pontos (monitores, impressoras, etc.);
 - Fácil armazenamento e leitura;
 - Valores dos pixels podem ser alterados individualmente ou em grupo;
- Desvantagens das imagens matriciais
 - Imagens podem ser muito grandes;
 - Dificuldade em realizar operações de escala;

Conversão de formatos

- Bitmap p/ Bitmap
 - Melhores resultados
 - Reajuste na informação de cor
 - Problemas com diferenças no tamanho da paleta de cor
- Vetorial p/ Vetorial
 - Problemas com diferenças entre o número e o tipo de objetos disponíveis
 - Problemas com interpretação de medidas e com a aparência dos elementos de imagem e das primitivas

Conversão de formatos

- Vetorial p/ Bitmap
 - Imagem vetorial é decomposta em pixels e colocada numa matriz
 - Qualidade depende do tamanho da matriz
 - Problemas de serrilhado



Conversão de formatos

- Bitmap para vetorial
 - Conversão mais difícil, com altos índices de falha
 - Algoritmos e heurísticas de detecção de formas
 - Resultados bons para formas geométricas, ruins para imagens reais
 - Normalmente resulta na perda de cores



Conversão de formatos

- Outros fatores que influenciam a conversão
 - Formatos proprietários ou específicos de uma aplicação
 - Número de cores
 - Tamanho da paleta
 - Formato da compressão



Raster (PNG)



Vector (SVG)

Arquitetura Gráfica

Frame-buffer

- O acesso ao dispositivo gráfico é mais lento que o acesso à memória.
 - Isso acontece devido à grande quantidade de componentes que atuam para que um *pixel* seja mostrado na tela.
 - O acesso ao controlador gráfico é feito via protocolos de hardware e software que são mais complicados que enviar um byte para memória.
- Assim, existem ocasiões que a velocidade é um fator importante:
 - Rasterização on-line, redesenho de imagens, jogos, animações etc.
- É necessário o desenvolvimento de técnicas de construção de imagens em memória.
 - Uma das principais é a criação do "*frame -buffer*".

Arquitetura Gráfica

Frame-buffer

- O frame-buffer é composto por uma região da memória que armazenará a imagem e por um grupo de rotinas de acesso à essa imagem.
- A memória é considerada como um “espaço” monodimensional, onde cada byte é endereçado por um único valor.
- Já uma imagem é um objeto gerado em espaço bidimensional (ou 3D), e isso acarretará:
 - Um pixel da imagem será representado por um número de bytes, dependendo de quantos bits serão necessários para compor a cor correspondente.
 - A quantidade de bytes na memória deverá ser suficiente para comportar todos os pixels a serem representados.
 - Deverá ter uma correspondência aritmética entre a posição (x, y) do pixel e seu endereço no frame-buffer.

Arquitetura Gráfica

Frame-buffer

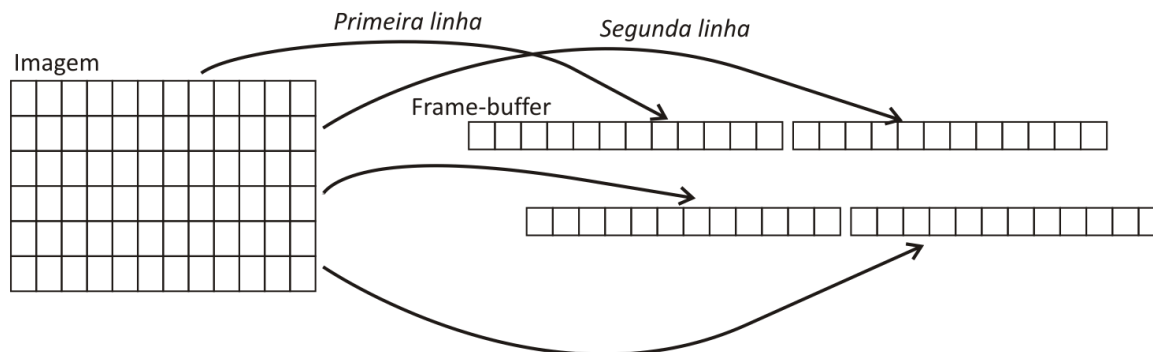
- Essa condição - de correspondência entre posições do pixel e do endereço – pode ser contornada se for utilizada uma linguagem de alto nível que aceite a definição de variáveis do tipo *arrays* ou matrizes bidimensionais.
- Por exemplo, seja uma imagem *true color* de 800 x 600 pixels de tamanho.
 - Cada pixel da imagem é representado por 32 bits (4 bytes).
- O tamanho total do frame-buffer para essa imagem será então 800 x 600 x 4 bytes = 1.920.000 bytes (~1,9 Mb).
- Em C, a declaração dessa área seria:

```
unsigned long int FB [800][600];
```

Arquitetura Gráfica

Frame-buffer

- Ao longo do programa, o acesso ao pixel (x, y) pode ser feito por:
`FB [X][Y] = 0; //O pixel será apagado`
Ou: `FB [X+1][Y+1] = FB [X][Y];`
 - Caso seja usada uma linguagem que não suporte variáveis do tipo matrizes, deve-se implementar uma arquitetura de memória e uma correspondente conversão de endereços.
- Uma boa (e simples) representação do frame-buffer é considerar cada linha como uma sequência de bytes que se agrupam em linhas como sequencias consecutivas.



Arquitetura Gráfica

Frame-buffer

- Seguindo a imagem exemplo (800x600, true color) e estabelecendo que o primeiro byte do frame-buffer é designado pelo endereço FB , o segundo pixel vai está no endereço $FB+4$ (cada pixel ocupa 4 bytes).
 - O 3º pixel está em $FB+2*4$ e o último byte da primeira linha está em $FB+799*4$.
 - O 1º pixel da segunda linha está em $FB+800*4$. O 2º em $FB+800*4+4$ e o 3º em $FB+800*4+2*4$.
- De forma genérica, um pixel qualquer na posição (x,y) na imagem exemplo é dada por : $FB + [800*y + x] * 4$.
- Para um caso qualquer, o endereço de um pixel no frame-buffer é dado por $m = FB + [x + Ly] * b$, onde m é o endereço do início do pixel no frame-buffer, FB é o endereço inicial do frame-buffer, L é o nº de pixels numa linha da imagem e b o tamanho que um pixel ocupa em bytes.

Referências desta aula

- AZEVEDO, Eduardo; CONCI, Aura. 2007. Computação Gráfica: Teoria e Prática. Elsevier, Vol. 2, 2007.
- Aula montada com base no material do Prof. Jorge Cavalcanti - UNIVASF.