

Dispositivos de Saída e Formação de Imagens

SCC0250/0650 - Computação Gráfica

Prof. Rosane Minghim

<https://edisciplinas.usp.br/course/view.php?id=61213>

<https://edisciplinas.usp.br/course/view.php?id=61210>

P.A.E. Diego Cintra e Fábio Felix

diegocintra@usp.br, f_diasfabio@usp.br

Instituto de Ciências Matemáticas e de Computação (ICMC)
Universidade de São Paulo (USP)

15 de março de 2018



Sumário

1 Dispositivos de Saída (Exibição)

2 Formação de Imagens

Sistema Computacional

- Um **sistema computacional** normalmente é composto por
 - Processador
 - Memória
 - **Dispositivos de saída (sistema gráfico)**
 - Dispositivos de entrada

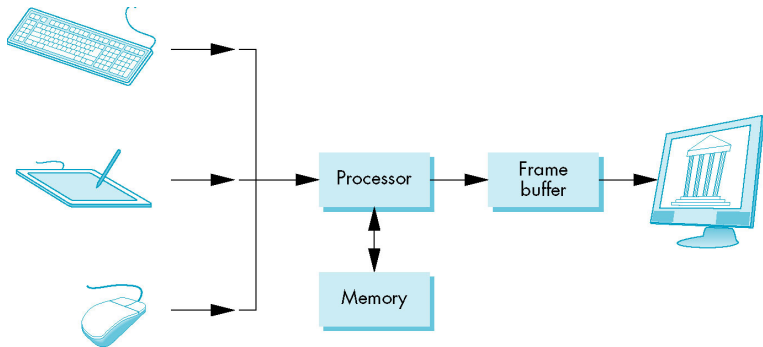
Computação Gráfica: Dispositivos de Exibição

- De forma geral existem dois tipos de **dispositivo de exibição**: dispositivo **vetorial** e dispositivo **matricial**

- Dispositivos vetoriais (“vector”)
 - natureza analógica
 - imagens formadas por segmentos de reta
 - geradas a partir de “display files”

- Dispositivos matriciais (“raster”)
 - natureza digital
 - imagens formadas pelo preenchimento de matriz de “pixels”
 - geradas a partir de “frame-buffers”

Sistema Gráfico Matricial (Raster)



Geração da Imagem em Dispositivo Matricial

- Descrição da **cena** mantida no **frame-buffer**, que contém uma posição associada a cada pixel da tela
- Para cada **pixel**, o valor armazenado na posição correspondente define a **intensidade (ou cor)** com que o pixel será traçado
- Todos os objetos são pixels

Frame Buffer

- Resolução: número de pixels
- Implementado c/ memória VRAM/DRAM/DDR/GDDR
 - Video random-access memory
 - Dynamic random-access memory
 - Acesso rápido para re-exibição e restauro
- frame buffer pode armazenar outras informações além da cor do pixel
 - múltiplas camadas, ou múltiplos buffers

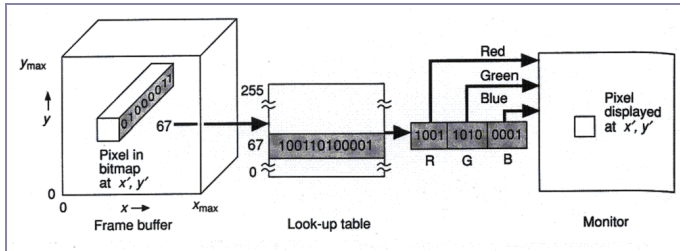
Frame Buffer

- Profundidade do frame buffer (depth):
 - Número de bits p/ cada pixel, determina o número de cores que o sistema consegue exibir
 - 1 bit = 2 cores
 - 8 bits = $2^8 = 256$ cores
 - 24 bits = $2^{24} = 16777216$ cores => sistema true color
 - d bits = 2^d cores possíveis (reais)
- Sistema RGB: grupos de bits associados a cada uma de 3 cores primárias: Red, Green, Blue

Frame Buffer

- A quantidade de **cores** representadas no frame buffer é **limitada**
 - **Paleta de cor** pode ser usada para permitir escolher qual faixa de cor utilizar
- O valor armazenado em uma posição do frame buffer pode ser usado como **índice de uma tabela de cor**
 - Clut: Color Lookup Table

Video Look-up Table



- Define a paleta de cores
 - Nesse exemplo uma paleta contém 256 cores das 4096 possíveis

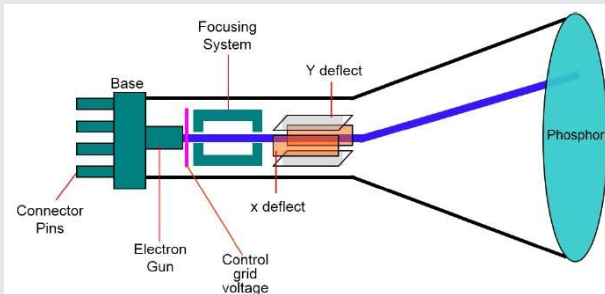
Processador

- O **processador** é responsável por transformar as primitivas gráficas (linhas, círculos, polígonos, etc.) em pixels no f.b.
 - Essa conversão é conhecida como **rasterização** ou **conversão matricial**

Tipos de Monitores

Cathode-Ray Tubes (CRT)

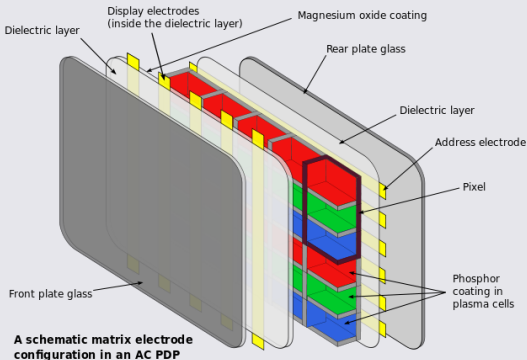
Figura:



Tipos de Monitores

Plasma

Figura:

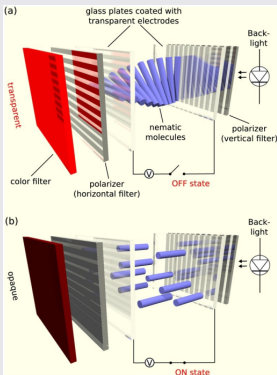


Tipos de Monitores

Liquid Crystal Display (LCD)

Vários tipos de LCD já foram desenvolvidos para melhorar a imagem e o processo de fabricação. Dentre eles podem ser destacados o *Twisted Nematic* (TN), *In-Plane Switching* (IPS), *Advanced Fringe Field Switching* (AFFS), *Vertical Alignment* (VA), *Advanced Super View* (ASV) etc.

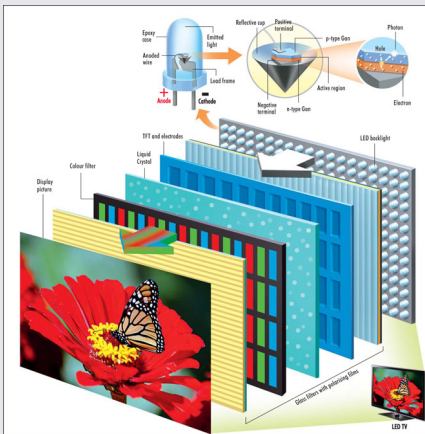
Figura:



Tipos de Monitores

Light-Emitting Diode (LED)

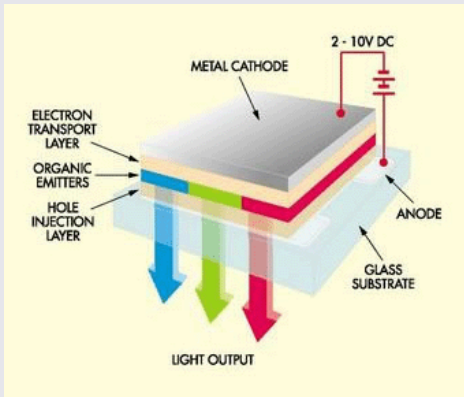
Figura:



Tipos de Monitores

Organic Light-Emitting Diode (OLED) e Active Matrix Organic Light-Emitting Diode (AMOLED)

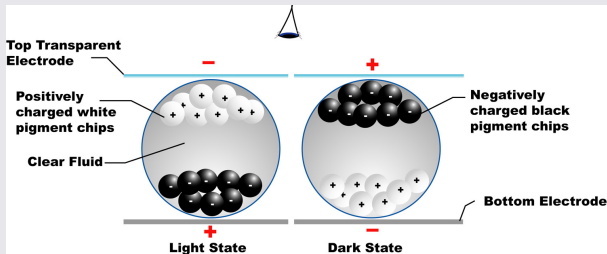
Figura:



Tipos de Monitores

Electronic paper

Figura:



Tipos de Monitores

- CRT

Vantagens e desvantagens

Grande, desajeitado, alto consumo de energia.

Tipos de Monitores

- CRT
- Plasma

Vantagens e desvantagens

Melhor brilho, muito contraste, ótima resolução, melhor ângulo de visão, temperatura alta, menor vida útil, tamanhos a partir de 50", ambientes claros podem prejudicar a imagem por causa de reflexos na tela.

Tipos de Monitores

- CRT
- Plasma
- LCD

Vantagens e desvantagens

Menos brilhantes do que o plasma, melhores em ambientes mais claros com menos reflexos, baixo consumo de energia, menor desgaste da tela, telas finas e leves, baixa interferência eletromagnética, menor ângulo de visão, menor contraste, falta de uniformidade da luz traseira que pode deformar a imagem.

Tipos de Monitores

- CRT
- Plasma
- LCD
- LED

Vantagens e desvantagens

Uniformidade da luz trazeira, melhor resolução, menor espessura.

Tipos de Monitores

- CRT
- Plasma
- LCD
- LED
- OLED

Vantagens e desvantagens

Melhor brilho, maior contraste, fidelidade de reprodução de cores, telas com pouca espessura (2cm, 0.5cm), ângulo de visão de 180°, telas podem ser flexíveis e até transparentes, preço muito elevado.

Tipos de Monitores

- CRT
- Plasma
- LCD
- LED
- OLED
- AMOLED

Vantagens e desvantagens

Maior taxa de atualização, melhor brilho, contraste, cores e nitidez.

Tipos de Monitores

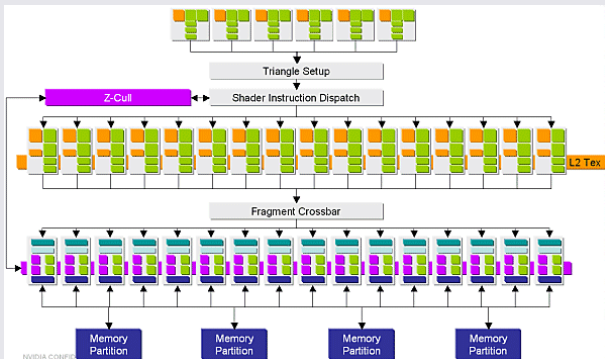
- CRT
- Plasma
- LCD
- LED
- OLED
- AMOLED
- Electronic Paper

Vantagens e desvantagens

Baixo consumo de energia, não pode ser visualizado sem uma fonte de luz externa, mas já existem modelos de aparelhos que possuem leds embutidos.

Exemplo de arquitetura de placa gráfica

Figura: NVIDIA GeForce 6800 Ultra GPU



Sumário

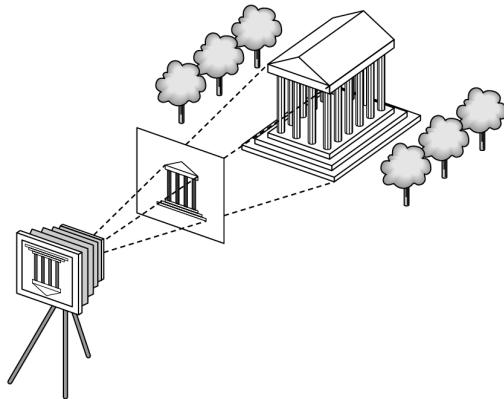
1 Dispositivos de Saída (Exibição)

2 Formação de Imagens

Formação de Imagens

- Em computação gráfica as imagens são formadas usando um **processo análogo** ao aplicado por sistemas físicos de imageamento
 - Câmeras
 - Microscópios
 - Telescópios

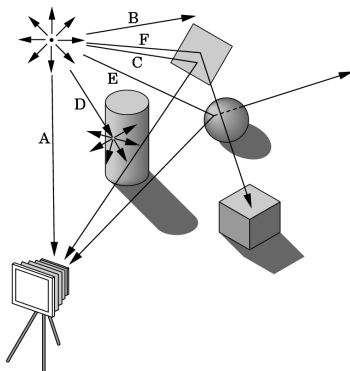
Modelo de Câmera Sintética



- (1) Objetos, (2) observador e (3) fonte(s) de luz

Ray Tracing

- Uma forma de implementação desse modelo de câmera é utilizando a abordagem de *ray tracing*



Pipeline Gráfico

- Outra implementação seria considerando uma arquitetura *Pipeline*, onde os objetos são **processados**, **um de cada vez**, na **ordem** que eles foram gerados pela aplicação
 - Só considera efeitos locais de luz
 - Abordagem empregada pela OpenGL



Pipeline Gráfico

- Muito do trabalho nesse pipeline está em **converter** representações de objetos de um **sistema de coordenada** para outro
 - Coordenadas dos objetos
 - Coordenadas de câmera (olho)
 - Coordenadas da tela



Processamento de Vértices

Processamento de Vértices

- **Projeção** é o processo que combina a visão 3D de um observador com objetos 3D para produzir imagens 2D
 - **Projeções perspectivas**: os raios de projeção se encontram no centro da projeção
 - **Projeção paralela**: os raios de projeção são paralelos, o centro da projeção é substituído por uma direção de projeção
- O processamento de vértices também calcula **cor**



Montagem de Primitivas

Montagem de Primitivas

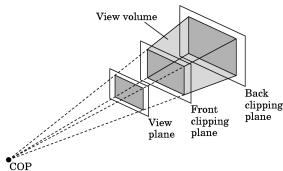
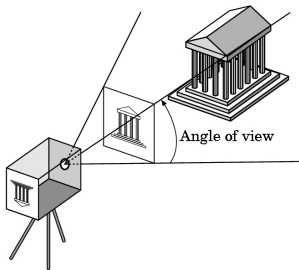
- Vértices devem compor objetos geométricos antes de acontecer o recorte e a rasterização
 - Segmentos de linha
 - Polígonos
 - Superfícies e curvas



Recorte

Recorte

- Assim como nenhuma câmera real pode “ver” completamente o mundo, a **câmera virtual** pode somente **ver parte mundo** ou espaço de objetos
 - Objetos que não estão dentro desse volume são “recortados” da cena



Rasterização

Rasterização

- Os objetos que não foram “recortados” **geram pixels** que são coloridos no frame buffer
 - Vértices e seus atributos são **interpolados** sobre os objetos pelo rasterizador
- A rasterização produz um conjunto de fragmentos (“pixels em potencial”) para cada objeto
 - Tem uma localização no frame buffer
 - Tem atributos de cor e profundidade



Processamento de Fragmentos

Processamento de Fragmentos

- Fragmentos são processados para **determinar a cor** dos pixels correspondentes no frame buffer
 - Mapeamento de textura ou interpolação das cores dos vértices
- **Fragmentos** podem ser **bloqueados** por outros fragmentos mais próximos da câmera
 - Remoção de superfície oculta

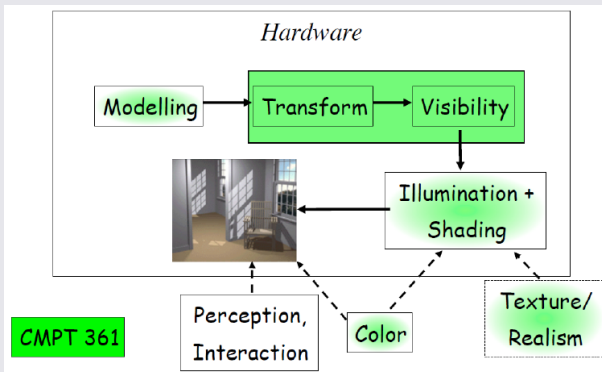


Pipeline Programável

- A maioria das placas gráficas implementam esse Pipeline Gráfico com funcionalidades fixas
 - Processamento dos vértices
 - Processamento dos fragmentos
- Atualmente o processamento dos vértices e fragmentos é programável
 - Muitas técnicas que não podiam ser implementadas em **tempo real agora são possíveis**
 - Isso é *GLSL (OpenGL Shading Language)*

Pipeline Gráfico

Figura:



Créditos

- Slides em inglês e imagens desta aula atribuídos a
 - Torsten Moller
 - <http://www.cs.sfu.ca/~torsten/Teaching/Cmpt361/>
 - <https://techreport.com/review/6572/nvidia-geforce-6800-ultra-gpu>
 - <https://olhardigital.com.br/video/plasma,-lcd,-oled-saiba-mais-sobre-as-telas-de-tv-de-hoje-e-do-futuro/27145>
 - https://openi.nlm.nih.gov/detailedresult.php?img=PMC3439495_pone.0044048.g001&req=4
 - https://www.infowester.com/lcd_plasma_oled.php
 - https://en.wikipedia.org/wiki/Plasma_display
 - <https://www.businesstoday.in/gadget/know-tech/led-tv-lcd-plasma-tv/story/191656.html>
 - http://www.tftcentral.co.uk/articles/coming_2007.htm
 - <https://ayddup.wordpress.com/tag/eink/>