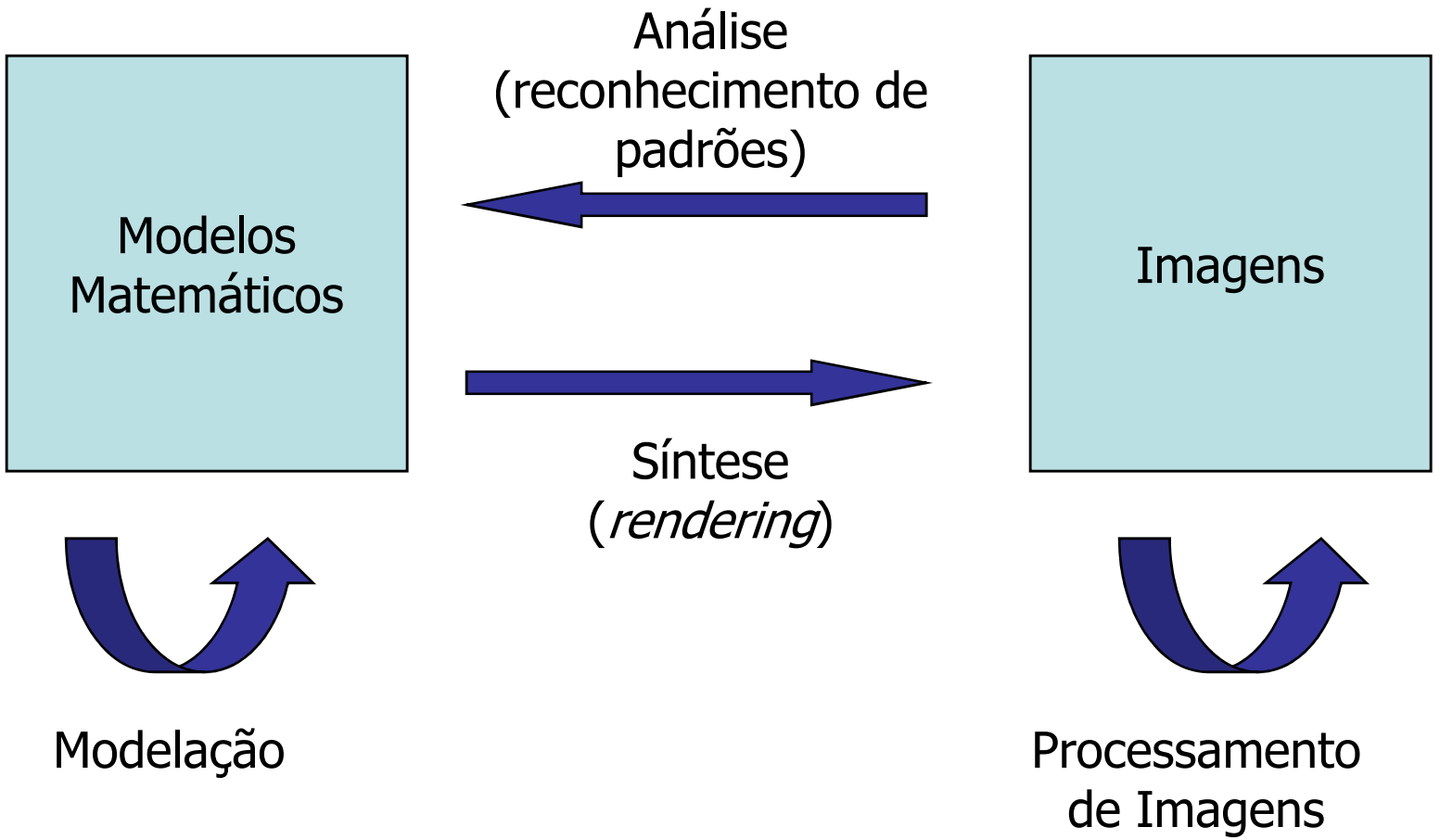
Introdução à Computação Gráfica
Preâmbulo

Adaptação: João Paulo Pereira
António Costa

Autoria: Claudio Esperança
Paulo Roma Cavalcanti

Computação Gráfica



Áreas relacionadas

- Computação
  - Algoritmos
  - Estruturas de Dados
  - Métodos Numéricos
- Matemática
  - Geometria
  - Álgebra Linear
- Física
  - Óptica
  - Mecânica
- Psicologia
  - Percepção
- Artes

Aplicações

- Desenho Assistido por Computador (CAD)
- Desenho Geométrico Assistido por Computador (CAGD)
- Sistemas de Informação Geográfica (SIG)
- Visualização Científica
- Visualização Médica
- Educação
- Entretenimento / Lazer

Representações Gráficas

- Gráficos "Vectoriais"
  - Representados por colecções de objectos geométricos
    - Pontos
    - Rectas
    - Curvas
    - Planos
    - Polígonos
- Gráficos "Matriciais"
  - Amostragem em grelhas rectangulares
  - Tipicamente imagens digitais
    - Matrizes de "pixels"
    - Cada pixel representa uma cor
  - Dados volumétricos
    - Imagens médicas
    - Cada pixel representa densidade ou intensidade de um campo físico

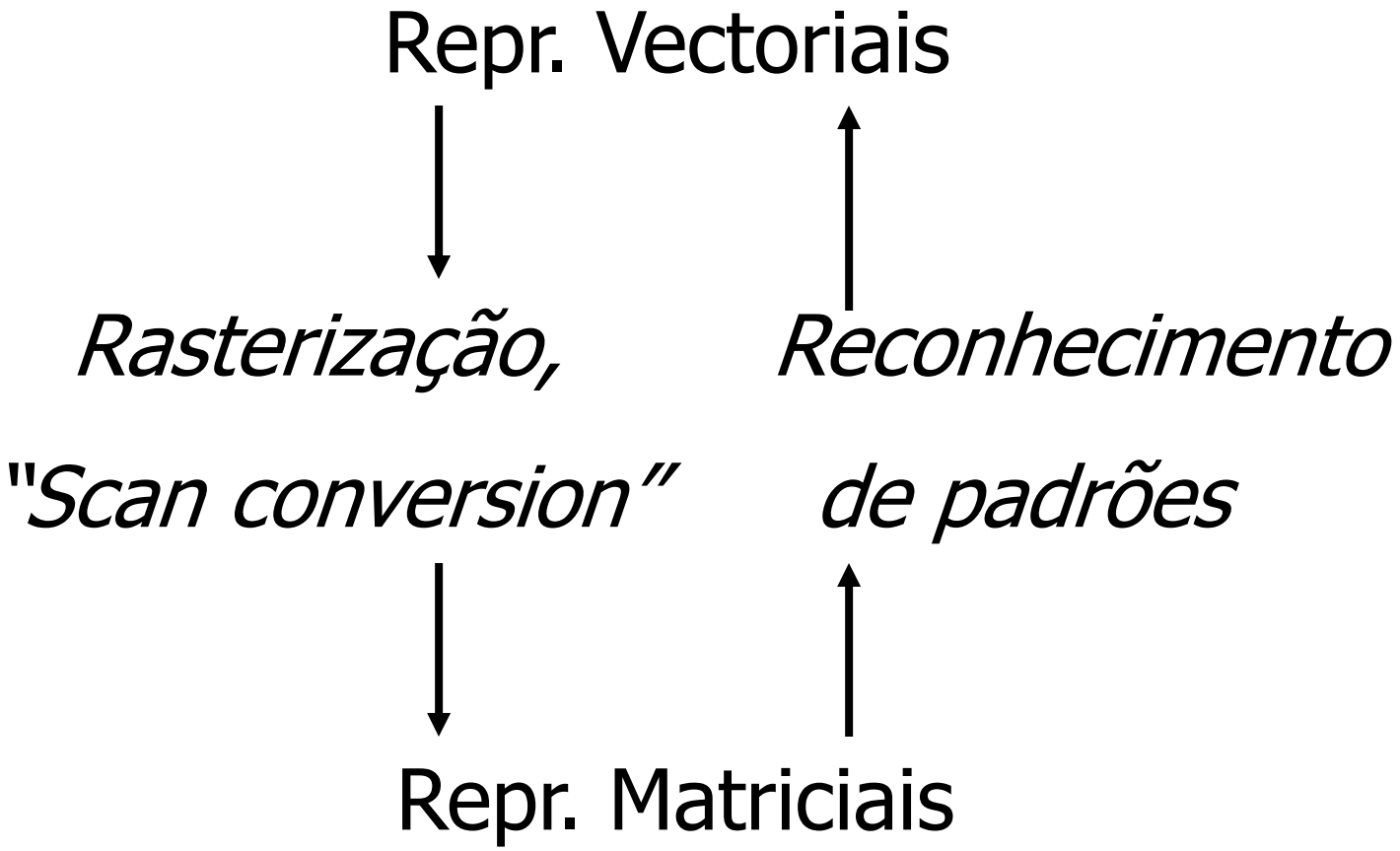
Representações Vectoriais

- Permitem uma série de operações (quase) sem perda de precisão
  - Transformações lineares / afim
  - Deformações
- Por que "quase"? Estruturas de dados utilizam pontos e vectores cujas coordenadas são números reais
  - É necessário usar aproximações
    - Representação em vírgula-flutuante
    - Números racionais
- Complexidade de processamento = *O* (nº vértices / vectores)
- Visualização
  - Dispositivos vectoriais
  - Dispositivos matriciais (requer amostragem, i.e., rasterização)

Representações Matriciais

- Representação flexível e muito comum
- Complexidade de processamento = *O* (nº de pixels)
- Muitas operações implicam em perda de precisão (re-amostragem)
  - Ex.: rotação, escala
  - Técnicas para lidar com o problema
    - Ex.: técnicas anti-discretização (*anti-aliasing*)
- Visualização
  - Dispositivos matriciais
  - Dispositivos vectoriais (requer uso de técnicas de reconhecimento de padrões)

Conversão entre representações



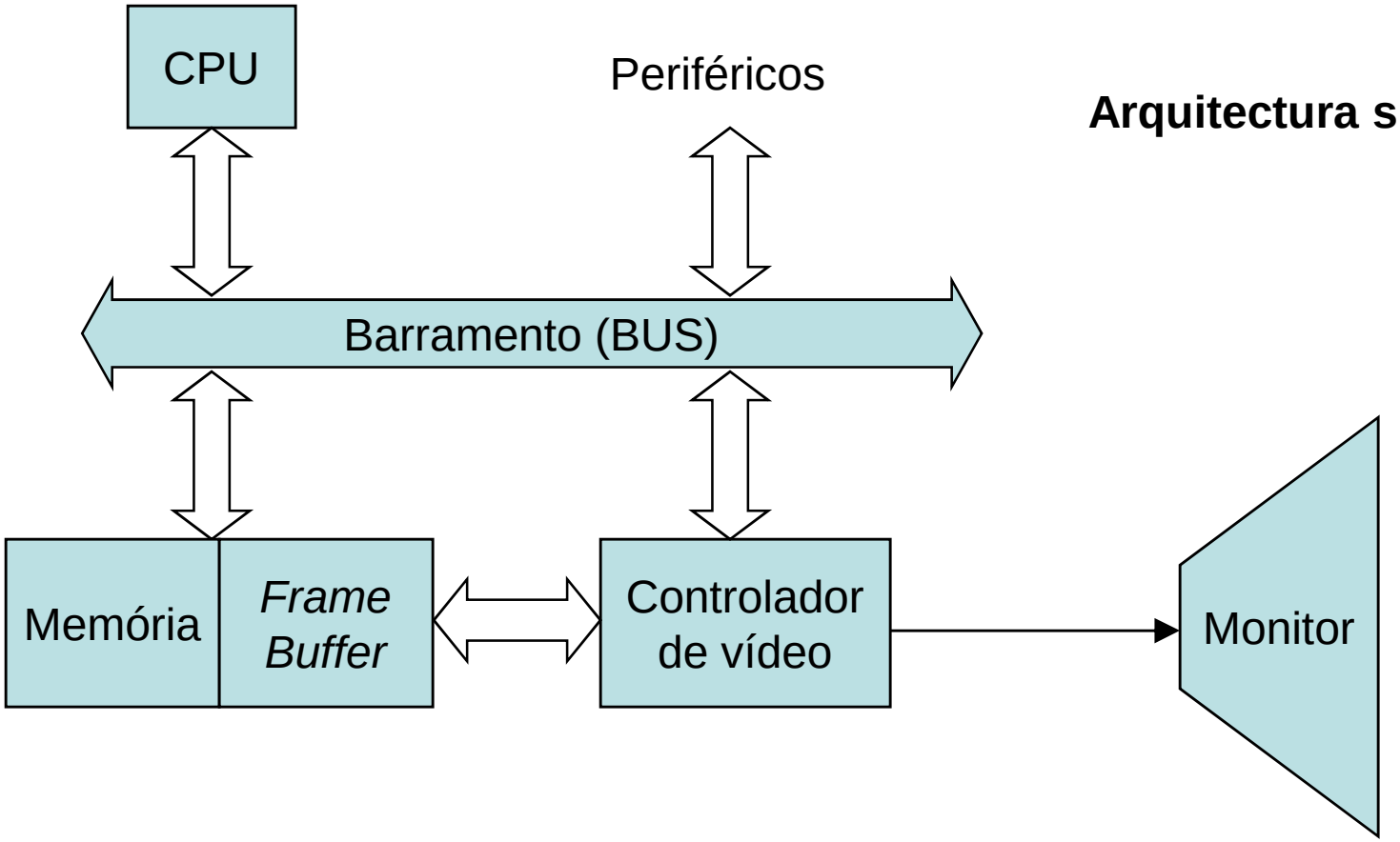
Dispositivos Gráficos

- Dispositivos vectoriais
  - Terminais gráficos vectoriais (obsoletos)
  - Traçadores (*plotters*)
  - Dispositivos virtuais
    - Ex.: Linguagens de descrição de página (HPGL / Postscript)
    - Rasterização implícita
- Dispositivos Matriciais
  - Praticamente sinónimo de dispositivo gráfico
  - Impressoras, *displays*

Displays

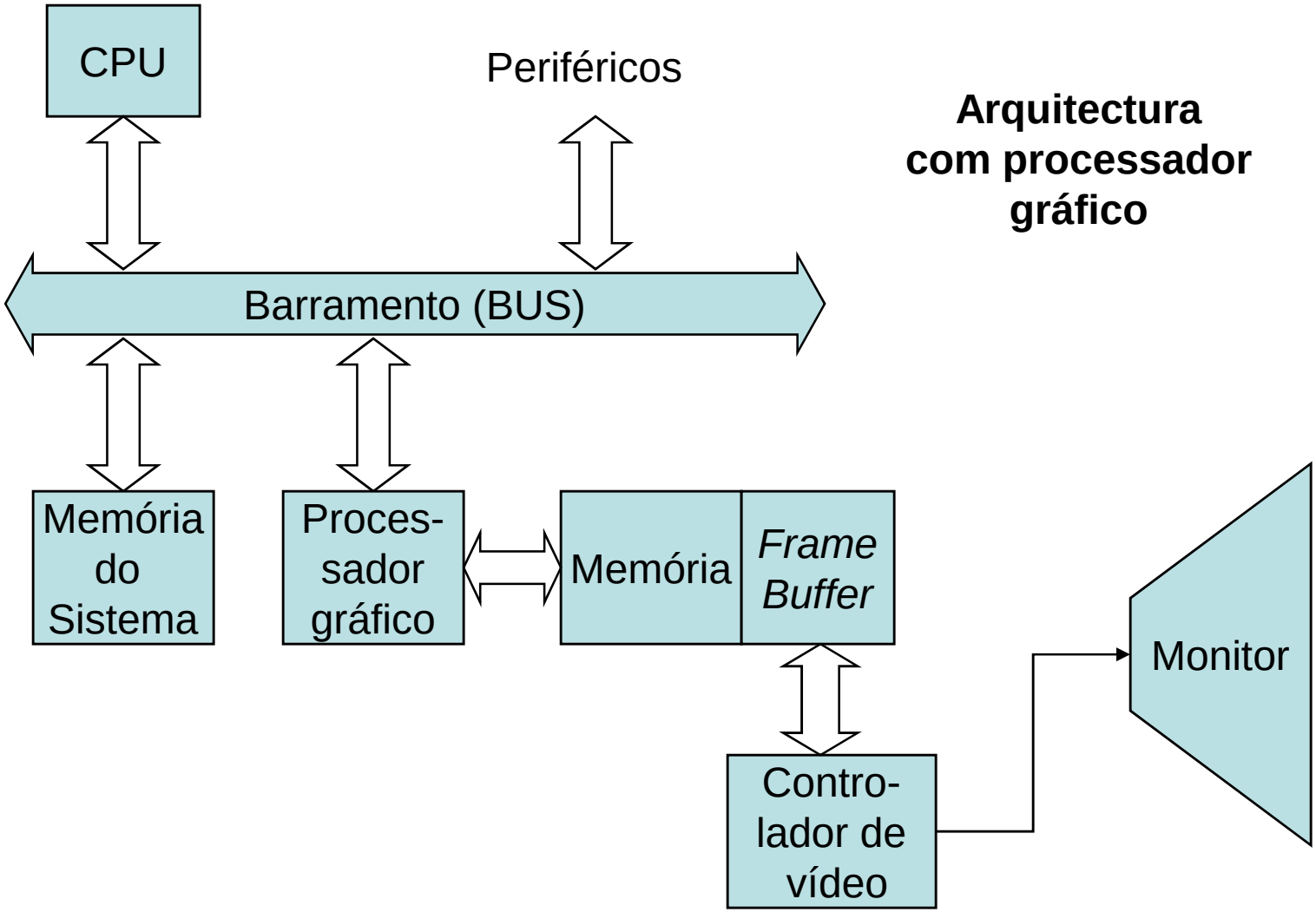
- Resolução espacial
  - Tipicamente de 640x480 até 1600x1200
  - Tendência de aumento
- Resolução no espaço de cor
  - Monocromático (preto e branco)
    - Praticamente restrito a PDAs e equipamentos de baixo custo
  - Tabela de cores
    - Cada pixel é representado por um número (tipicamente 8 bits – de 0 a 255) que indexa uma tabela de cores (tipicamente RGB 24 bits)
    - Poucas (ex.: 256) cores simultâneas mas cada cor pode ser escolhida de um universo grande (ex.: 2^{24})
    - Problema da quantização de cores
  - RGB
    - Cor é expressa por quantidades discretas de vermelho (*red*), verde (*green*) e azul (*blue*)
    - Tipicamente 24 bits (8 bits para cada componente)
    - Quando o número de bits não é divisível por 3, a resolução do azul costuma ser menor que das outras 2 componentes

Arquitectura de Sistemas Gráficos



Arquitectura simples

Arquitectura de Sistemas Gráficos



Processador (acelerador) gráfico

- Hardware especializado
- Uso de paralelismo para atingir alto desempenho
- Alivia o CPU do sistema de algumas tarefas, incluindo:
  - Transformações
    - Rotação, translação, escala, etc.
  - Recorte (clipping)
    - Supressão de elementos fora da janela de visualização
  - Projecção (3D $\rightarrow$2D)
  - Mapeamento de texturas
  - Rasterização
  - Amostragem de curvas e superfícies paramétricas
    - Geração de pontos a partir de formas polinomiais
- Normalmente usa memória separada da do sistema
  - Maior largura de banda no seu uso

Programação Gráfica

- À primeira vista: basta desenhar
  - Uma subrotina para desenhar cada tipo de objeto
- Mas ...
  - Como fazer interacção?
  - Como estruturar a cena?
  - Como controlar os atributos dos objectos?
  - Como resolver problemas de visibilidade?
  - Como suportar diversos dispositivos gráficos?
  - Como fazer programas independentes dos sistemas operativos?
- Ferramentas
  - APIs gráficas (ex.: OpenGL, PHIGS, Java3D)
  - Camadas de interface com o S. O. / sistema de janelas