

1

CONCEITOS BÁSICOS

1.1. O QUE É COMPUTA- ÇÃO GRÁ- FICA ?

Definição

Conjunto de métodos e técnicas computacionais para a representação de forma gráfica, através de um computador, de objetos de um mundo real (ou virtual). É uma subárea da ciência da computação que é responsável pelo estudo de métodos e técnicas para geração, manipulação, interpretação e armazenamento de modelos e figuras gráficas de objetos utilizando o computador.

Implica:

- Em um modelo interno deste mundo a ser representado
- Em um conjunto de transformações para representar este modelo em um dispositivo de saída de um computador (vídeo, plotter, etc)

Histórico

A representação gráfica detalhada e realística de modelos em um mundo virtual começou com a necessidade de se utilizar o computador para substituir a prancheta de desenho. Surgiu o Projeto Auxiliado por Computador - CAD (Computer-Aided Design). Projeto

Auxiliado por Computador - CAD (Computer-Aided Design): Primeiros dispositivos desenvolvidos para representar objetos gráficos foram os displays vetoriais (random scan displays).

- Não tinham memória de vídeo
- Cada objeto era desenhado separadamente pelo canhão de vídeo (não existia o conceito de pixel)
- Canhão tinha que repetidamente redesenhar toda a cena.

Motivou o Display File: Lista de todos os objetos constantemente redesenhados na tela.

1.2. APLICAÇÕES DA COMPUTAÇÃO GRÁFICA

As áreas de aplicação de CG são muitas além de crescer muito rapidamente, a seguir são listadas apenas algumas:

- CAD/CAM e CAE: sistemas gráficos interativos são utilizados para projetar componentes e peças de sistemas mecânicos, elétricos, eletromecânicos e eletrônicos. Esta lista pode ser acrescida por projetos de edifícios, automóveis, aviões e navios dentre outros projetos. Estes sistemas permitem a visualização e manipulação de representações geométricas com o objetivo de permitir ao usuário interagir com um modelo computacional que está sendo projetado visando à realização de simulações e testes;
- Simulação: a simulação é uma ferramenta que parte do campo científico até o lazer, é uma área com grande crescimento e grandes perspectivas futuras. É possível realizar simulação de experiências perigosas ou de difícil realização. Na área do lazer é observada a grande quantidade de produção de filmes e/ou efeitos especiais para a televisão;
- Controle de processos: sistemas de controle de tráfego aéreo além de controle de usinas nucleares ou sistemas de produção diversos;
- Cartografia: a computação gráfica é usada para produzir representações precisas e esquemáticas de fenômenos naturais e geográficos obtidos a partir da coleta de dados;
- Entretenimento: desenvolvimento de jogos diversos, este sem dúvida um dos mercados de maior crescimento e com maior expectativa para os próximos anos, além de não apresentar possibilidade de saturação do mercado;

- Medicina: nesta área as aplicações são muitas, auxiliam ao diagnóstico médico e possibilitam a visualização e navegação em partes do corpo. O desenvolvimento destes sistemas possibilita desde um planejamento cirúrgico mais detalhado até o uso na área de ensino;

Dados Gráficos podem ser classificados em vetoriais (dados representados por pontos coordenados e linhas, capazes de ligar tais pontos) e matriciais (consiste de uma matriz de pontos no espaço e a cada ponto é associado um atributo de Intensidade). É chamada de representação da imagem uma estrutura de dados que a contém e apresentação da imagem é a relação da mesma com o equipamento que a reproduz.

A entrada pode ser diferenciada coordenadas relativas e coordenadas absolutas.

Vetorial:

- Coordenadas relativas: mouses (ópticos e mecânicos), joysticks, bolota e diais;
- Coordenadas absolutas: teclados, mesas digitalizadoras, caneta óptica e touch screen;

Matricial:

- Digitalizadores de vídeo: a partir de um sinal de televisão gera uma matriz de pontos em um monitor de vídeo;
- Varredores digitais (SCANNER): são baseados na absorção da luz;
- FILM SCANNER: o equivalente para filme;

Parte do hardware responsável pela integração do gerenciamento simultâneo da tela e dos recursos gráficos mínimos residentes no hardware. Necessitam administrar dispositivos de visualização complexa, diferente de processadores que se destinam somente ao armazenamento e manuseio de processamento matemático. Assim, computadores gráficos necessitam, além da unidade de processamento central, de módulos dedicados exclusivamente ao gerenciamento do dispositivo de visualização.

1.3. EQUIPAMENTOS PARA COMPUTAÇÃO GRÁFICA

Equipamentos de entrada

Processadores Gráficos

Equipamentos de saída

Plotters: traçadores gráficos (de mesa e de tambor);

Terminais de vídeo: CRT (Tubos de raios catódicos) princípio da televisão. Um cátodo aquecido emite um feixe eletrônico que é convenientemente focalizado por um conjunto de lentes e acelerado. Ao incidir na superfície de tubo, que é recoberta de fósforo: ponto bombardeado pelos elétrons emite luz e a associação de pontos aceso forma a imagem. O feixe eletrônico é desviado pela ação de campos magnéticos produzidos por bobinas à qual é aplicada uma d.d.p. proporcional ao desvio desejado.

Característica dos terminais de vídeo:

- Resolução: n° de pontos distintos que pode ser aceso, que depende da capa-cidade do sistema computacional que gerencia o display. Propriedade: aspect ratio à aspect ratio = x/y à uma linha vertical de x pontos é de mesmo de uma linha horizontal de y pontos;
- Persistência do fósforo: capacidade do elemento químico reter a luminosidade;
- Sistema de deflexão: controlador do esboço produzido na tela;

Métodos de varredura

São duas as metodologias básicas para geração de imagens na tela gráfica:

- Varredura randômica ou vetorial (random/vector scan): o feixe de elétrons é direcionado somente para as partes da tela onde a figura é desenhada;
- Varredura rastreada ou matricial (raster scan): o feixe de elétrons é disparado sobre todas as partes da tela, sendo a tela criada por um conjunto de pontos (processo da televisão) varre-se a tela de cima para baixo e da esquerda para a direita, marcando os trechos visíveis durante o trajeto;
- Até 1980 prevalecia equipamento de varredura randômica, este método é ideal para representar vetores, típicos das aplicações de CAD. Atualmente predominam equipamentos de varredura matricial.

Todos dispositivos gráficos usam uma grade retangular de localizações endereçáveis. Este retângulo é chamado de "Display Rectangle" ou "Graphics I/O Rectangle". Dispositivos gráficos são fixados de acordo com sua resolução gráfica, o número de horizontais versus posições verticais. Parâmetros importantes: características gráficas básicas;

1. ndh => N° de localizações gráficas endereçáveis horizontalmente;
2. ndv => N° de localizações gráficas endereçáveis verticalmente;
3. width => a largura física de retângulo em milímetros;
4. height => a altura física de retângulo em milímetros.

Geral: monitor PC with 245,0 ndh - 640

height - 186,0 ndv - 200

Baseado nestes conceitos, define-se:

1. Resolução horizontal: Res-horiz = ndh/width;
2. Tamanho de pontos na horizontal: Hor-dot-size = width/ndh;
3. Resolução vertical: res-vert = ndv/height;
4. Tamanho de pontos na vertical: vert-dot-size = height/ndv;
5. Total de pontos endereçáveis: total-nr-dot = ndv*ndh;
6. Graphics aspect ratio: aspect-ratio = vert-dot-size/horiz-dot-size;
7. Physical aspect ratio: Phys-aspect-ratio = height/width;

A Computação Gráfica começou com a exibição de dados em plotters (traçadores) e em telas de tubos de raios catódicos (cathode ray tube - CRT) logo após a introdução dos próprios computadores. Tanto os PCs e Workstations assim como todos os monitores de TV usam essa tecnologia.

Até o início dos anos 80, a Computação Gráfica era um campo pequeno e especializado, principalmente porque o hardware era caro e os programas baseados em gráficos fáceis de usar e eficientes em termos de custo eram poucos. Então, os computadores pessoais

1.4. SISTEMAS DE COORDENADAS

1.5. TECNOLOGIA CRT (CATHODE RAY TUBE)

com monitores gráficos de varredura rastreada (raster graphics display) - tais como o Star, da Xerox, e mais tarde os famosos Apple Macintosh e o IBM PC e seus clones - popularizaram o uso de gráficos de bitmap para interação usuário-computador.

Um bitmap é uma representação de zeros e uns de uma matriz de pontos (pixels, de picture elements) na tela. Assim que os bitmaps tornaram-se mais confortáveis, uma explosão de aplicativos baseados em gráficos fáceis de usar e baratos surgiu. As Interfaces Gráficas com o Usuário (Graphic-based User Interface - GUI) permitiram a milhões de novos usuários controlar aplicativos simples e baratos, tais como planilhas, processadores de texto e programas de desenho.

O entendimento básico da operação interna do CRT ajudará no desenvolvimento de algoritmos para CG. O CRT é um dispositivo analógico que é usado por um dispositivo digital - o computador. Esta seção também abordará a interface entre esses dois dispositivos.

Histórico Dispositivos de traçado e impressoras tais como teletipos e impressoras de linha datam dos primórdios da computação. O computador Whirlwind desenvolvido em 1950 no Massachusetts Institute of Technology (MIT) tinha monitores CRT controláveis como dispositivo de saída, tanto para uso do operador quanto para câmeras produzindo saída impressa. O sistema de defesa aérea SAGE desenvolvido nos meados dos anos 50 foi o primeiro a usar monitores CRT com command and control (comando e controle) nos quais os operadores identificavam alvos com canetas óticas (dispositivos manuais de apontamento que captam a luz emitida por objetos na tela). Os princípios da Computação Gráfica interativa moderna, no entanto, são encontrados na tese de doutorado de Ivan Sutherland sobre o sistema de desenho Sketchpad, de 1963. Ele introduziu estruturas de dados para armazenar hierarquias de símbolos contruídos via replicação de componentes padrão, uma técnica semelhante ao uso de normógrafos para desenhar circuitos. Ele também desenvolveu técnicas de interação que usavam o teclado e a caneta ótica para fazer escolhas, apontar e desenhar, além de muitas outras idéias e técnicas fundamentais, ainda em uso hoje.

Ao mesmo tempo, tornou-se claro para a indústria de computadores, automobilística e aeroespacial que o CAD e a Computer-Aided Manufacturing (CAM, ou Fabricação Assitida por Computador) tin-

ham enorme potencial para automatizar atividades de projeto e desenho. O sistema DAC, da General Motors (1964) e o sistema de projeto de lentes da Itek Digitek foram esforços pioneiros que mostraram a utilidade da interação gráfica nos ciclos de projeto comuns à engenharia. Nos meados dos anos 60, um grande número de projetos de pesquisa e produtos comerciais havia aparecido.

Como naquela época a entrada/saída (I/O) do computador eram feitas principalmente em batch mode (lote) usando cartões perfurados, grandes foram as esperanças de uma explosão da comunicação interativa usuário-computador. Gráficos interativos, como “a janela no computador”, deveriam ser uma parte integrante dos ciclos de projeto interativo mais acelerados. Os resultados não foram tão dramáticos, no entanto, uma vez que gráficos interativos permaneciam acima dos recursos de todos, a não ser as empresas mais tecnologicamente avançadas. Entre as razões para isso estavam:

- O alto custo do hardware gráfico, quando produzido sem os benefícios da produção em massa - numa época em que automóveis custavam poucos milhares de dólares, computadores custavam muitos milhões de dólares, e o primeiro monitor de computador comercial custava mais de cem mil dólares;
- A necessidade de grandes e caros recursos computacionais para suportar enormes bancos de dados de projeto, manipulação interativa de imagens, e os grande programas de pós-processamento cujos dados de entrada vinham da fase de projeto gráfico;
- A dificuldade de escrever grandes programas interativos para os novos ambientes operacionais com time-sharing (compartilhamento de tempo) numa época em que tanto gráficos quanto interação eram novidade para os programadores - FORTRAN, predominantemente - acostumados a fazer programas em batch;
- Os Softwares não-portáteis, únicos, escritos tipicamente para o monitor de um fabricante em particular e produzido sem os benefícios dos princípios modernos de engenharia de software para contruir sistemas modulares e estruturados; quando o software não é portátil, trocar de monitor implica reescrever programas que já funcionam.

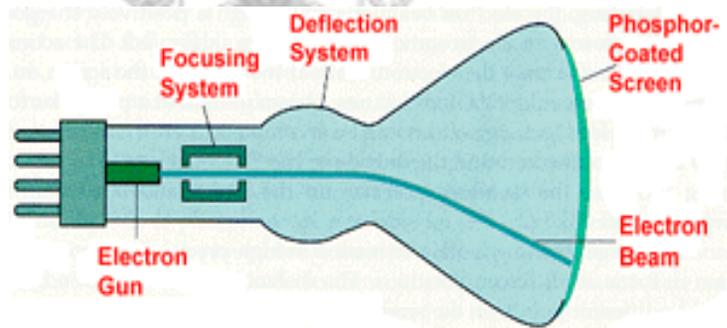
Foi o advento de computadores pessoais baseados em gráficos, tais como o Apple Macintosh e o IBM PC, que finalmente baixaram os

preços do hardware e do software tão drasticamente que milhões de computadores gráficos foram vendidos como “acessórios” para escritórios e residências; quando o campo iniciou no início dos anos 60, seus praticantes nunca sonhavam que os computadores pessoais apresentando interação gráfica fossem se tornar tão comuns tão cedo.

Os principais componentes do CRT são:

- Canhão de elétrons: emite um feixe de elétrons (raios catódicos: elétrons emitidos em movimento rápido pelo cátodo de um tubo em descarga) que passa através do sistema de focagem e deflexão;
- Sistema de focagem: produz uma imagem não embaçada pela focagem de elétrons do raio catódico em um estreito raio de luz;
- Sistema de deflexão: conduz o raio de luz em direção a posição especificada na tela coberta de fósforo;
- Tela coberta de fósforo: emite uma pequena mancha de luz em cada posição contactada pelo raio catódico;

Figura 1.2. Tubo de Raios Catódicos - CRT



Dependendo do tipo do fósforo, diferente luz colorida é gerada na posição da tela pelo feixe de elétrons. A luz desaparece rapidamente, em torno de 10 a 60 microssegundos. Este tempo depende da persistência do fósforo existente no vídeo.

Para manter a imagem no vídeo por mais tempo, a imagem precisará ser desenhada no vídeo. Este processo é chamado de “refreshing” do vídeo.

Pode-se pensar na tela como uma matrix bi-dimensional que é frequentemente percorrida. Levando em consideração que esta matrix

tem L linhas e C colunas. Cada linha da L linha (ou os pontos na linha horizontal da tela) é chamado de scan-line. O número de pontos que pode ser mostrado no vídeo ($L \times C$) é definido como a resolução do vídeo. A resolução é relatada para o tamanho do ponto que emite a luz após o contato com raio de elétrons.

Nesta seção será abordado dispositivos que utilizam tecnologia CRT (onde o feixe de elétrons (raio catódico) é posicionado em um ponto (x,y) na tela recoberta de fósforo para gerar uma mancha de luz).

1.6. DISPOSITIVOS RANDOM SCAN

Um dispositivo random scan usando CRT direciona o feixe de elétrons somente para a parte da tela onde a figura está sendo desenhada. Este tipo de dispositivo conhece como desenhar uma linha na tela dado por dois pontos finais. Isto é porque estes dispositivos são referidos como “vector displays” (ou displais caligráficos).

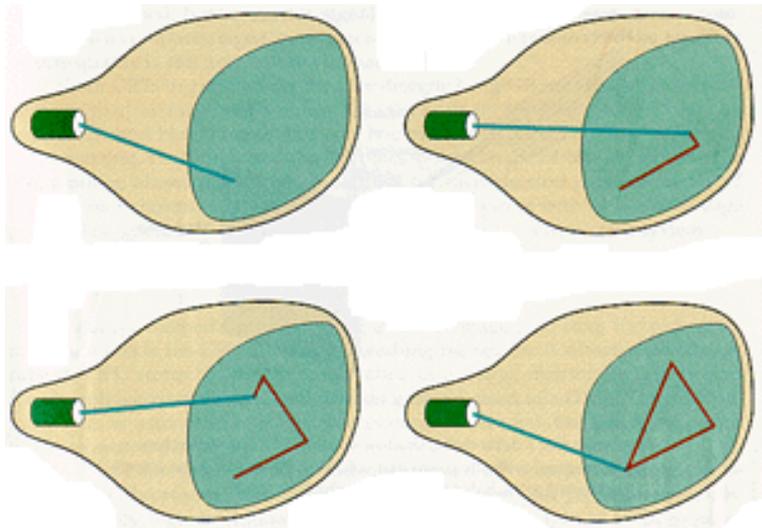
Estes dispositivos de exibição desenvolvidos nos anos 60 foram comuns até meados dos anos 80. O termo vetor é usado aqui como um sinônimo de linha; um stroke (pincelada) é uma linha pequena, e caracteres eram feitos de seqüências desses strokes. Um sistema vetorial típico consiste de um processador de vídeo conectado à CPU, um buffer (área de armazenamento, memória temporária) de vídeo e um CRT. O buffer armazena a display list (lista a ser mostrada); esta lista contém comandos de traçado de pontos e linhas com coordenadas (x,y) ou (x,y,z) , assim como comandos de traçado de caracteres.

Os comandos para traçar pontos, linhas e caracteres são interpretados pelo processador de vídeo. Ele envia coordenadas digitais de pontos para um gerador de vetores que converte os valores digitais das coordenadas em voltagens analógicas para os circuitos defletores que posicionam um feixe de elétrons na cobertura de fósforo do CRT. A essência de um sistema vetorial é que o feixe é dirigido de um ponto ao outro, de acordo com a ordem da lista de comandos; essa técnica se chama random scan (varredura aleatória). Apresentações de laser também usam deflexão random scan do feixe de laser. Uma vez que a emissão de luz do fósforo decai em dezenas ou no máximo centenas de microsegundos, o processador de vídeo deve executar repetidamente a display list para fazer o refresh (reativamento) do fósforo ao menos 30 vezes por segundo (30 Hz) para evitar o flicker (piscada); por isso, o buffer que contém a display list é usualmente chamado de refresh buffer.

Considerando um triângulo que é para ser desenhado em um dispositivo random scan, cada linha pode ser desenhada em qualquer ordem

Tudo o que existe desenhado na tela irá desaparecer em aproximadamente 40 microssegundos dependendo da persistência do fósforo que existe na tela. Lembrando que para permanecer no vídeo a figura precisa ser desenhada sempre (refresh).

Figura 1.3. Random Scan Display



Os objetos no caso de serem muitos são definidos por um conjunto de linhas. Onde o objeto (ex. triângulo) consiste também em muitas linhas, o tempo de refresh pode ser maior que a persistência do fósforo. Isto pode ocasionar “flickering” da imagem na tela

Além disso, isto é quase impossível criar imagens com objetos sombreados ou preencher áreas com uma dada cor.

1.7. DISPOSITIVOS RASTER SCAN

Um dispositivo raster scan usando CRT utiliza outra forma de direcionar a coluna de eletrons na tela através de uma linha por vez de cima para baixo. Em um dispositivo raster scan, a tela é definida como uma matriz de duas dimensões. Lembrando que o termo raster era usado para uma matriz 2D. Cada ponto na tela corresponde a um elemento da matriz que é chamando de pixel (picture element).

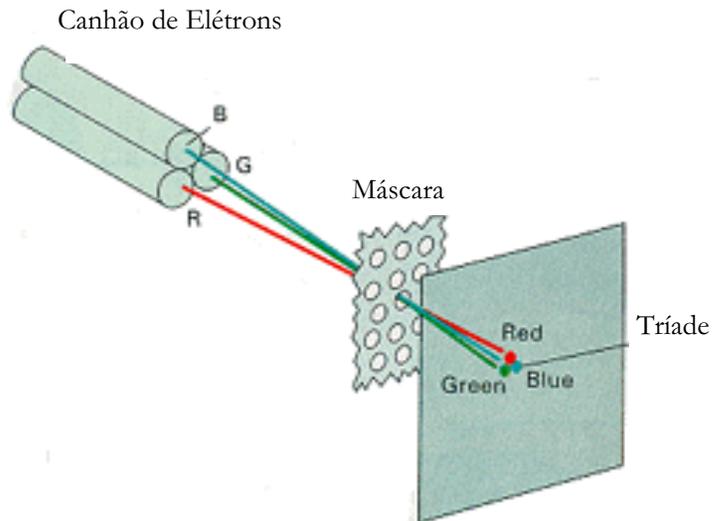
O desenvolvimento no início dos anos 70 de raster graphics (gráficos de rastro), baseados na tecnologia da televisão, contribuíram mais para o crescimento do campo do que qualquer outra tecnologia. Raster displays (monitores de rastro) armazenam as primitivas (linhas, caracteres e áreas preenchidas) num refresh buffer em termos de seus pixels. Em alguns monitores de raster, há um controlador de vídeo em hardware que recebe e interpreta seqüências de comandos de saída similares aos dos monitores vetoriais; em sistemas mais simples (e mais comuns), tais como os dos computadores pessoais, o controlador de vídeo existe apenas como um componente de software da biblioteca gráfica, e o refresh buffer é apenas um pedaço da memória que pode ser lido pelo subsistema de exibição de imagem (frequentemente chamado controlador de vídeo) que produz a verdadeira imagem na tela.

A imagem completa num monitor de raster é formado pelo próprio raster (rastro), que é um conjunto de linhas de rastro horizontais, cada uma sendo uma fileira de pixels individuais; o raster é, portanto, armazenado como uma matriz de pixels representando a área inteira da tela. A imagem inteira é varrida seqüencialmente pelo controlador de vídeo, uma linha de rastro por vez, de cima para baixo e então de volta para cima. A cada pixel, a intensidade do feixe é programada para representar a intensidade do pixel; em sistemas coloridos, três feixes são controlados - um para cada cor primária (RGB - red, green e blue) - de acordo com a especificação das três componentes de cor do valor do pixel.

Nos primeiros modelos, o refresh era feito a taxas de 30 Hz, como a televisão, atualmente uma refresh rate (taxa de refresh) de 60 Hz ou mais é usada para evitar o flicker na imagem. Ao contrário dos sistemas de vetores, onde o refresh buffer armazenava comandos e coordenadas, num sistema de raster a imagem inteira deve ser explicitamente armazenada. O termo bitmap é usado como matriz de pixels. Gráficos de bitmap tem a vantagem sobre gráficos de vetores de que a verdadeira exibição da imagem é manipulada por lógicas simples de varredura: a varredura regular e repetitiva é de longe mais fácil de implementar do que os algoritmos de random scan, que têm que ser altamente precisos para manter a linearidade e a repetibilidade do defletor do feixe.

A disponibilidade de memórias RAM baratas para uso com bitmaps no início dos anos 70 foi o impulso necessário para fazer dos gráficos de raster a tecnologia dominante de hardware. Monitores CRT

Figura 1.4.



monocromáticos desenhavam em preto-e-branco ou preto-e-verde; alguns usavam laranja e preto. Bitmaps monocromáticos contêm um único bit por pixel, e o bitmap de uma tela inteira com resolução de 1024 x 1024 tem somente 220 bits, ou 128 Kb. Sistemas coloridos baratos têm 8 bits por pixel, permitindo 256 cores simultâneas; sistemas mais caros têm 24 bits por pixel, permitindo 16 milhões de cores. Um sistema colorido típico de resolução 1280 x 1024 com 24 bits por pixel requer aproximadamente 3,75 Mb de RAM.

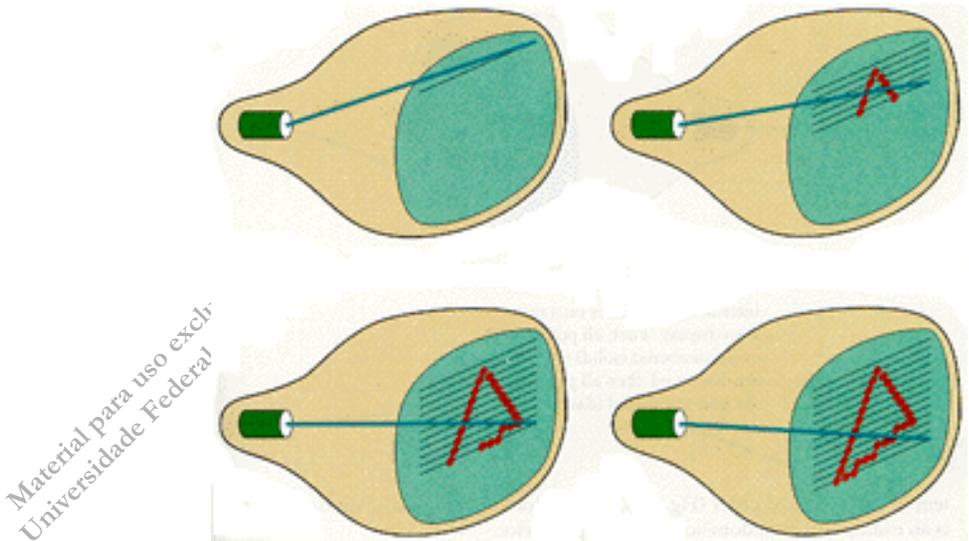
Considerando que um triângulo deve ser desenhado em um dispositivo raster scan. Nesta hora a coluna de elétrons varre toda tela, uma scan-line (linha da matriz) por vez de cima para baixo. Os pixels através do que a linha passa montam as cores desejadas.

As linhas não são desenhadas diretamente como nos dispositivos random scan. As linhas agora consistem de pixels que podem ser desenhados na tela conforme a figura a seguir exemplifica.

O processo de atualização ou *refreshing* pode ser bem melhorado para monitores raster scan. Muitos monitores de TV são dispositivos raster scan: uma linha por vez.

Os sistemas raster-scan são mais utilizados para desenharmos imagens complexas (ex: retratos, paisagens), uma vez que desenharmos a ima-

Figura 1.5. Raster Scan Display moderno



gem, pixel a pixel, não sendo portanto, muito precisos a desenhar linhas (estas ficam com um aspecto "recortado"). Pelo contrário, os sistemas random-scan, desenhavam imagens complexas com uma aparência muito "lisa" (com "contornos" pouco definidos), são portanto, mais utilizados para aplicações que utilizam muito o desenho de linhas (ex: aplicações wireframe de CAD - Computer Aided Design). Um outro aspecto relacionado com estes sistemas é que, uma vez que a informação armazenada com respeito à imagem, é constituída por um conjunto de comandos de desenho de linhas e não por um conjunto de valores referentes a cor e intensidade de cada pixel, por tal motivo, apresentam resoluções mais elevadas que os sistemas raster-scan.

A arquitetura básica de um sistema gráfico pode ser descrita de forma resumida, sem entrarmos em detalhes técnicos, pelos seguintes componentes:

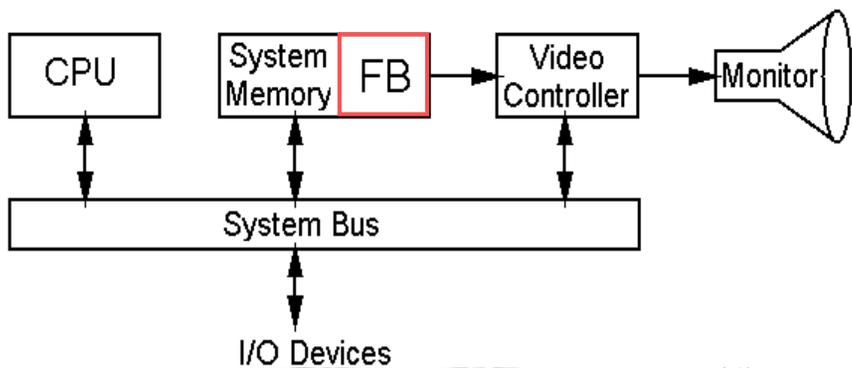
- Software Gráfico - armazena display file
- Controladora/processador de vídeo - desenha linhas
- Vídeos onde o canhão sempre faz a mesma operação de varredura por toda a área de vídeo.

1.8. CARACTERÍSTICAS DO RASTER SCAN DISPLAY

- Imagem discretizada sob a forma de Pixels. O conteúdo do vídeo é representado por uma matrix de pixels. É essa matriz de pixels, denominada *frame buffer* ou **memória de vídeo**, que é redesenhada na tela em um ritmo fixo, independentemente de seu conteúdo.

Esta arquitetura motivou o desenvolvimento de técnicas para representação da parte visível do *display file* no frame buffer, denominadas de técnicas de **recorte** ou *clipping*.

Figura 1.6. Arquitetura simplificada de sistema gráfico



Se incrementarmos este modelo com uma placa controladora de vídeo possuidora de um acelerador gráfico, o que hoje em dia é quase que a regra, essa arquitetura fica como mostra a figura 1.7 adiante.

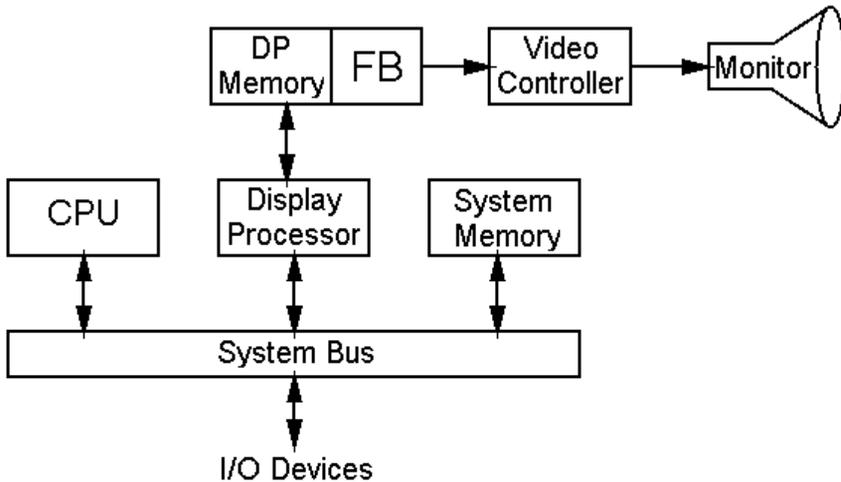
1.9. DISPLAY FILE

Como dissemos anteriormente, o *display file* originou-se da necessidade dos *random scan displays* de se possuir uma lista circular com uma representação vetorial de todos os objetos da cena, percorrida seqüencialmente em um loop infinito. Essa técnica, com algumas modificações, se mostrou útil também para os *raster scan displays*, onde não é o conteúdo do display file e sim o do frame buffer que é desenhado diretamente na tela.

O display file é uma lista composta por todos os objetos do mundo representado no sistema gráfico, independentemente de estarem visíveis em um determinado momento ou não. Características:

- Era percorrida em um loop infinito em random scan displays.

Figura 1.7. Arquitetura de um sistema com acelerador gráfico



- É também usada como estrutura de dados principal para armazenar objetos em sistemas gráficos possuidores de displays mais modernos.

Em sistemas gráficos mais modernos, não existe mais a necessidade de se realizar o percurso do display file em um loop infinito, pois uma vez calculada a representação de um objeto sob a forma de pixels e escritos estes pixels no seu lugar correto no frame buffer, eles permanecem lá até que outra coisa os sobrescreva.

Por isso podemos hoje em dia simplesmente representar o display file como uma lista com as representações vetoriais ou algum tipo de representação compatível com o tipo de mundo que estamos representando, dos objetos deste mundo.

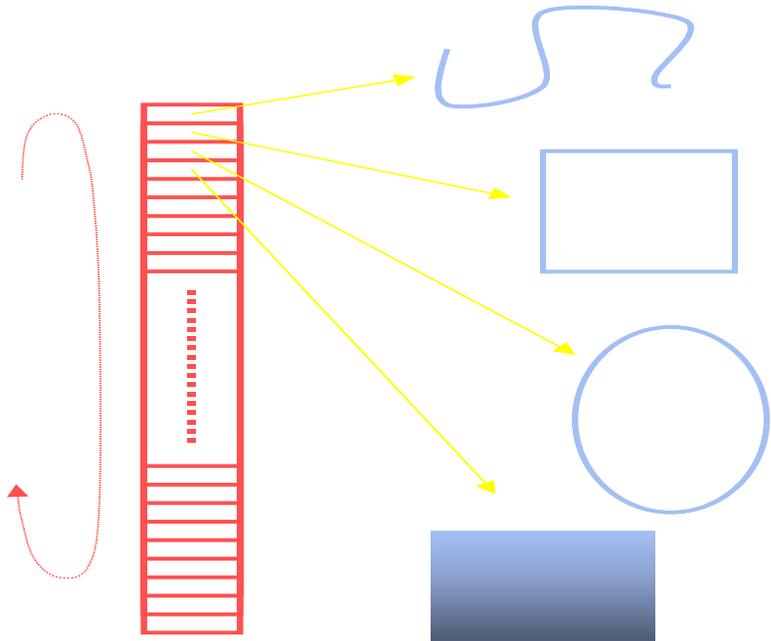
A representação dos objetos do display file no frame buffer é realizada em duas etapas.

Parcialmente realizada pelo sistema operacional e pelo hardware da máquina

- Escrita na memória de vídeo
- Leitura da memória de vídeo
- Aceleração gráfica

1.10. REPRESENTAÇÃO DO DISPLAY FILE NO VÍDEO

Figura 1.8. Display file



- Representação de objetos na forma de pixel

Esta parte não nos interessa e não vamos examinar em detalhes.

A representação dos objetos do display file é também parcialmente realizada pelo programa gráfico através da transformação de viewport ou transformada de porta de visualização. O objetivo da transformada de viewport é realizar uma transformação de sistema de coordenadas: do sistema de coordenadas do mundo, que é o sistema de coordenadas interno da representação dos objetos no sistema para o sistema de coordenadas do vídeo, que é o sistema de coordenadas imposto pela resolução gráfica do vídeo e pela quantidade de memória disponível no framebuffer.

O conceito é bastante simples, mas vamos nos permitir divagar um pouco para explicá-lo com mais precisão. Quando construímos um sistema gráfico para representar alguma coisa no vídeo do computador, na verdade estamos primeiramente criando uma representação interna de algum mundo, que pode ser um mundo real, na medida que descrevemos um circuito eletrônico em um sistema de projeto de placas de circuito impresso, descrevemos algum objeto arquitetô-

nico em um sistema de projeto civil ou descrevemos uma parte de nossa galáxia em um sistema de simulação e visualização astronômico ou pode ser um mundo virtual em um jogo de computador qualquer.. Para que essa descrição seja a mais fiel possível é natural escolher-se as unidade de medida do mundo que estamos modelando para descrever este modelo deste mundo. Estas unidades podem ser micra em circuitos integrados ou parsecs em um sistema de astronomia. Isto não importa, importa é que o nosso sistema gráfico nos dê a liberdade de representar internamente o nosso mundo da forma mais fiel possível. Da mesma forma também deve o sistema ser capaz de representar o mundo com a quantidade de dimensões que ele possui, por exemplo 2D para um circuito impresso ou uma planta baixa e 3D para um projeto arquitetônico ou um modelo da galáxia. Este é o modelo interno, que estará contido no display file.

Ora, o vídeo de um computador, por outro lado, é uma superfície bidimensional, relativamente pequena e com capacidade bastante limitada de mostrar detalhes. E nós temos que fazer o nosso modelo de mundo interno ser desenhado nesta tela de uma forma satisfatória. Para isso é necessário que haja uma **transformação de visualização**. Essa transformação deverá ser capaz de criar uma representação dos objetos contidos no display file capaz de ser desenhada no vídeo do computador. Além disso, essa representação deve ainda nos satisfazer do ponto de vista estético ou de qualidade de representação dessas informações. Este, no fundo é o objetivo da Computação Gráfica: desenvolver técnicas que possibilitem a implementação dessa transformação.

Nesse processo existem duas operações que são básicas. Podemos até nos arriscar a dizer que todo o resto da Computação Gráfica não passa de refinamentos dessas duas operações básicas:

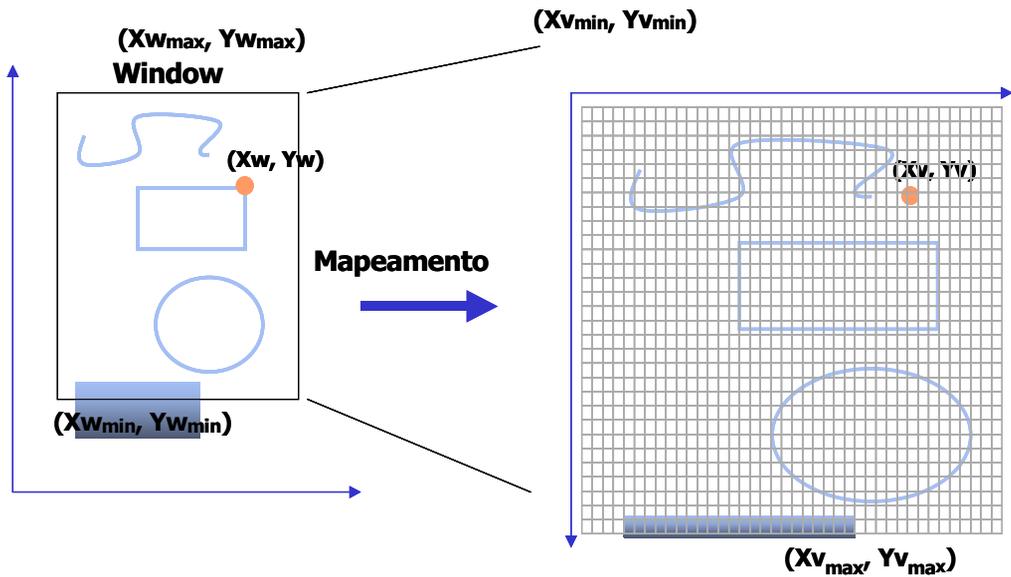
Operação de Seleção de Conteúdo:

Operação de Transformação de Representação:

Essas duas operações são implementadas pela window e pela transformada de viewport. O posicionamento e a determinação do tamanho da window realizam a operação de seleção de conteúdo. Evidentemente que se o mundo representado for um mundo 3D, teremos de projetar a parte visível do mundo sobre a window, mas isto é um detalhe que vamos ignorar no momento. A operação de transformação de representação é implementada pela transformada

de viewport, que transforma dados contínuos de um mundo em dados discretos na resolução e nas coordenadas da viewport.

Figura 1.9. Visualização da transformada de viewport



Resumo Window X
Viewport em 2D

Window: Retângulo que representa o recorte do mundo 2D representado no display file que será mostrado no vídeo.

- Representado no sistema de coordenadas cartesiano.
- Tamanho variável (Zoom).
- Posição variável (navegação)
- Deve ser representado por uma estrutura de dados: Xwmin, Ywmin, Xwmax, Ywmax

Viewport: a Porta de Visualização é a área do vídeo onde será desenhado o que se deseja mostrar.

- Representado num sistema de coordenadas parcialmente invertido, onde o eixo Y aponta para baixo.
- Tamanho fixo ou não controlado pela aplicação.

- deve ser representado por uma estrutura de dados: X_{vpmin} , Y_{vpmin} , X_{vpmax} , Y_{vpmax}

O cálculo da transformada de viewport ou porta de visualização é bastante simples. No fundo é apenas um mapeamento de coordenadas da window para coordenadas da viewport. Para encontrar o valor de x na viewport, x_{vp} , realizamos a transformação linear:

$$x_{vp} = \frac{X_w - X_{wmin}}{X_{wmax} - X_{wmin}} \cdot (X_{vpmax} - X_{vpmin}) \quad (\text{EQ. 1.1})$$

Para encontrar o valor de y realizamos a transformação um pouco diferente, levando em conta a inversão do eixo y :

$$Y_{vp} = \left(1 - \frac{Y_w - Y_{wmin}}{Y_{wmax} - Y_{wmin}} \right) \cdot (Y_{vpmax} - Y_{vpmin}).$$

Neste texto vamos ver uma série de exercícios. O objetivo dos exercícios propostos na Parte I deste Texto é a construção, passo a passo, de um Sistema Gráfico Interativo capaz de representar, em perspectiva realista, objetos em 3D como modelos de arame e também como superfícies bicúbicas renderizadas como malhas de curvas. Os exercícios são progressivos e constroem uns sobre os outros, o que significa que você necessita ter implementado o exercício anterior para poder implementar o atual, pois vai usar o código que produziu como ponto de partida para o novo exercício.

Neste seu primeiro trabalho, vamos lançar as bases do seu sistema, iniciando pela implementação de conceitos como window, viewport e display file. Para tanto, implemente o sistema básico de CG 2D contendo:

- SDisplay file para 2D capaz de representar pontos, retas e polígonos (listas de pontos interconectados), onde: Cada objeto possui um nome, cada objeto possui um tipo e cada objeto possui sua lista de coordenadas de tamanho variável dependendo de seu tipo. Para facilitar a sua vida mais tarde, chame o objeto polígono de *wireframe*.
- Transformação de viewport em 2D.
- Funções de navegação 2D (movimentação do window).

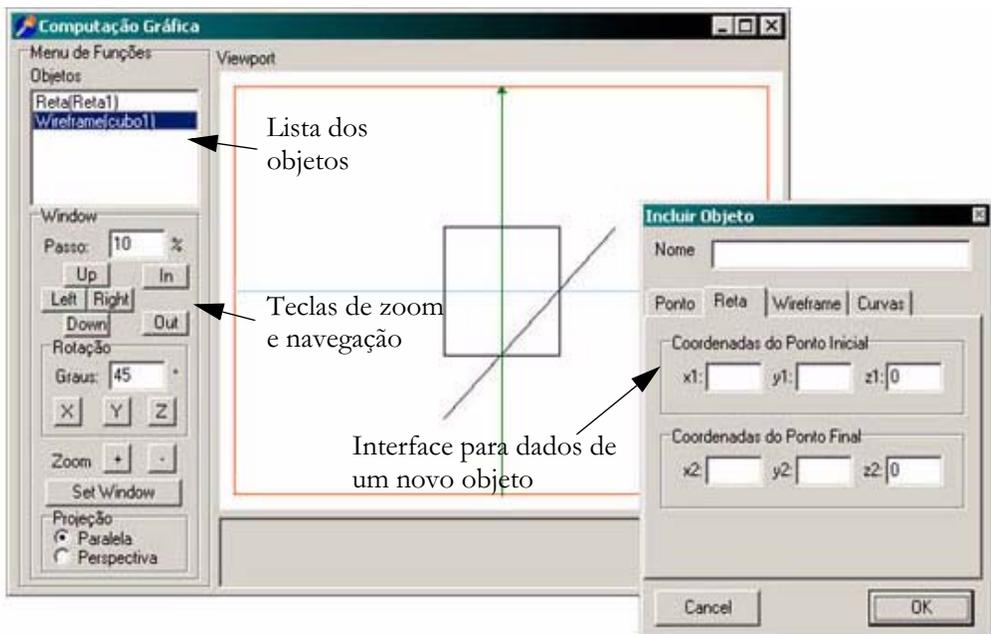
1.11. TRANSFORMADA DE VIEWPORT

1.12. EXERCÍCIO 1.1: SISTEMA GRÁFICO INTERATIVO BÁSICO

- Funções de Zoom (modificação do tamanho do window).

A figura 1.10 abaixo dá uma idéia de como deverá ser a interface de usuário do sistema.

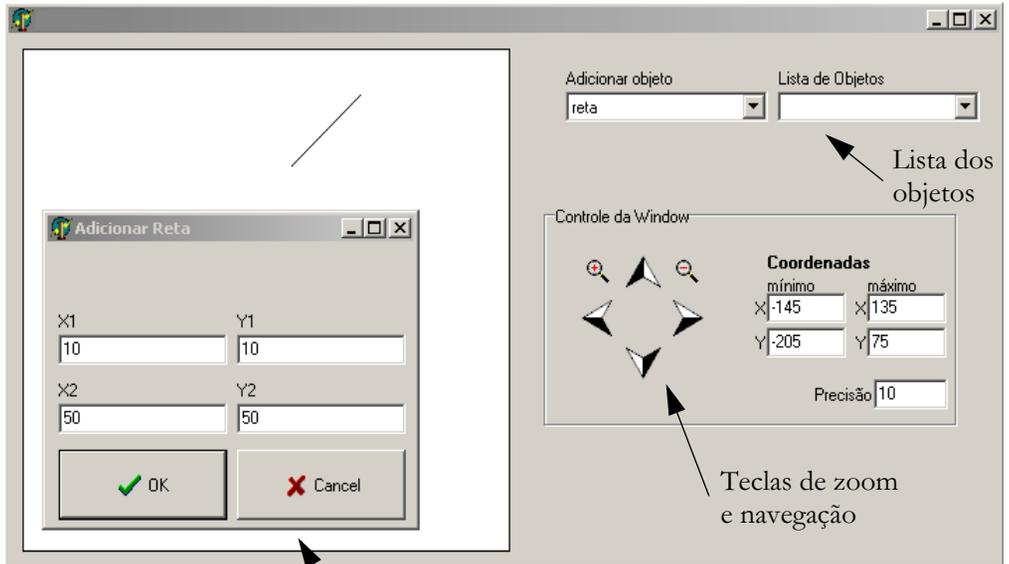
Figura 1.10. Exemplo de interface de usuário de uma realização do exercício proposto: Seu Primeiro Sistema Gráfico Interativo



Uma outra sugestão de interface, mais simples, pode ser vista abaixo.

Material para uso exclusivo na Universidade Federal de Santa Catarina

Figura 1.11. Outra forma de implementar o Exercício 1.1





Material para uso exclusivo na disciplina INE5341
Universidade Federal de Santa Catarina - UFSC